Oracle Data Guard: Disaster Recovery for Oracle Exadata Database Machine

Oracle Maximum Availability Architecture White Paper
April 2012

Maximum Availability Architecture

Oracle Best Practices For High Availability

ORACLE

Overview

The Oracle Exadata Database Machine provides an optimal solution for all database workloads, ranging from scan-intensive data warehouse applications to highly concurrent OLTP applications. Exadata delivers extreme performance in a highly available and highly secure environment.

Oracle Data Guard is Oracle's disaster recovery solution prescribed by the Maximum Availability Architecture (MAA) to protect mission critical databases residing on Exadata. Data Guard is also used to maintain availability should any outage unexpectedly impact the production database and to minimize downtime during planned maintenance.

Data Guard is included with Oracle Database Enterprise Edition and provides the management, monitoring, and automation software to create and maintain one or more synchronized copies (standby databases), that protect the production database (primary database) from failures, disasters, errors, and corruptions.

Data Guard's native integration with the Oracle database enables the highest level of data protection. A Data Guard physical standby database supports all Oracle datatypes and features while supporting the high transaction volume driven by Exadata. Administrators can use either manual or automatic failover to quickly transition a standby database to the production role. Finally, Data Guard standby databases deliver high return on investment when used for queries, reports, backups, testing, or rolling database upgrades, or other planned maintenance, while also providing disaster protection.

Data Guard and Exadata Database Machine

Oracle Data Guard[1] is the MAA[2] best practice recommendation for Exadata Database Machine for:

- **Disaster recovery (DR).** Data Guard protects against data loss and downtime should the primary site become unavailable. Continuous Oracle validation enables Data Guard to provide the best data protection for the Oracle database. Data Guard physical standby databases transparently support all Oracle data types, database features, and applications, and can meet the demanding performance requirements of Exadata.
- **High Availability (HA).** Data Guard supports up to 30 standby databases in a single configuration. An increasing number of customers use this flexibility to deploy both a local Data Guard standby for HA and a remote Data Guard standby for DR. A local Data Guard standby database complements the internal HA features of Exadata by maintaining availability when unexpected failures or human error make the production database unavailable even though the remainder of the site is still operational. Low network latency between the production database and the local standby makes it easy to use synchronous redo transport and achieve zero data loss if a failover is required. Likewise, the close proximity of the local standby to the application tier also enables fast redirection of application clients to the new primary database. Following a failover to a local standby database, the remote standby database in the Data Guard configuration will recognize that the failover has occurred and automatically begin receiving redo from the new primary database - maintaining disaster protection at all times.
- **Database rolling upgrades.** A Data Guard standby database can also be used to minimize planned downtime when implementing Oracle patch-sets or upgrading to new Oracle releases. Such upgrades are executed using the database rolling upgrade process. Beginning with Oracle Database 11.2.0.2, patches that are certified as Standby-First patches can be implemented first on a physical standby database, tested thoroughly, and then implemented at the primary database. A Data Guard Standby Database is also useful for testing purposes, particularly when using Data Guard Snapshot Standby in conjunction with Oracle Real Application Testing.

[1] http://www.oracle.com/goto/dataguard

[2] http://www.oracle.com/goto/maa

- **Certain migrations to Exadata storage.** Data Guard is one of several approaches available from Oracle to facilitate initial migration to Exadata with minimal downtime. A Data Guard standby database can also be used to reduce downtime for other types of planned maintenance.
- **Offloading read-only workload.** Maintaining a synchronized Data Guard standby database only requires a fraction of the CPU and memory available on a standby system. This leaves considerable capacity on the standby system to address other requirements such as offloading read-only workloads and fast incremental backups from the primary database – increasing Return on Investment (ROI) and reducing the effective cost of DR.

In general, it is transparent to Data Guard whether primary and/or standby databases reside on Exadata or on other hardware. This paper highlights MAA best practices that are particularly relevant to using Data Guard with Exadata.

Best Practices for Disaster Recovery

Data Guard provides the management, monitoring, and automation software infrastructure to create and maintain one or more standby databases to protect Oracle data from failures, disasters, errors, and data corruptions. As users commit transactions at a primary database, Oracle generates redo records and writes them to a local online log file. Data Guard transport services automatically transmit a copy of the redo directly from the primary log buffer to the standby database(s), either synchronously or asynchronously, where it is written to a standby redo log file (SRL).

After the redo has been received by the standby, one of the following methods is used for synchronizing the standby database with the primary, as configured by the administrator:

- Redo Apply (physical standby) uses media recovery to apply changes to a standby database that is open read-only (Active Data Guard). Redo Apply maintains a block for block, exact replica of the primary database, insuring that data is protected at all times.
- SQL Apply (logical standby) uses the SQL Apply process to mine redo data, convert it to SQL transactions and data, and then apply the transactions to a standby database that is open read-write. A logical standby database contains the same logical information as the primary database, although the physical organization and structure of the data can be different.

Data Guard Redo Apply (physical standby) is the MAA best practice for disaster recovery for Exadata. To configure Redo Apply, follow Oracle Data Guard documentation[3], Oracle High Availability Best Practices[4], and the additional best practices described in this white paper.

Data Guard Redo Apply

Redo Apply is the simplest, fastest, and most reliable method of maintaining an independent, synchronized replica of a primary database. A physical standby database applies the redo received from its primary database using the managed recovery process (MRP), a Data Guard aware extension of standard Oracle media recovery that is used by every Oracle database. The MRP controls the highly parallel recovery mechanism native in the Oracle kernel. Oracle has implemented a number of Redo Apply performance enhancements to take specific advantage of the superior I/O characteristics of Exadata. Performance benchmarks conducted by Oracle have achieved standby apply rates of 290 Mbytes/second for OLTP workloads and 640 Mbytes/second for batch workloads (greater than 2 TB/hour).

In general, Redo Apply performance should be sufficient for most workloads using default settings. If, however, the standby database is unable to keep pace with the rate of primary database redo generation, see the MAA best practices for tuning media recovery.[5] Tuning also requires the use of Standby Statspack, see My Oracle Support Note 454848.1 for details.

Standby Instantiation

The simplest method to create a standby database is to use Oracle Recovery Manager (RMAN). There are two options.

The first option uses the `DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE` command[6]. The advantage of active database duplication is that it does not require source database backups or additional disk space for a staging area on the target system. Active duplication copies mounted or online database files over a network directly to an auxiliary instance. There are, however, tradeoffs to this approach. It will impact network performance. It will also impact performance of the primary host because the source database will run processes

[3] http://docs.oracle.com/cd/E11882_01/server.112/e25608/toc.htm

[4] http://docs.oracle.com/cd/E11882_01/server.112/e10803/config_dg.htm#CEGEADFC

[5] http://www.oracle.com/technetwork/database/features/availability/maa-wp-11gr1-activedataguard-1-128199.pdf

[6] http://download.oracle.com/docs/cd/E11882_01/server.112/e17022/rcmbackp.htm#SBYDB4988

required to transfer the files to the target host. Internal MAA performance tests have shown the following results using active duplication:

- 2.9 TB/hour using a single InfiniBand and one RMAN session
- 0.4TB/hour using a single 1 GigE (10 GigE testing is in progress)

The second option uses backup-based duplication and multiple auxiliary instances.[7] Multiple `BACKUP AS COPY` commands with an RMAN session for each Oracle instance will scale throughput beyond what is possible with active duplication. Internal MAA performance tests have shown the following results using this method:

- 6.1 TB/hour over two InfiniBand with two RMAN sessions
- 11.7TB/hour over four InfiniBand with four RMAN sessions
- 3TB/hour across eight 1 GigE and eight RMAN sessions (10GigE testing is in progress)

See My Oracle Support Note 1206603.1 for more information on optimizing RMAN duplicate by using multiple `BACKUP AS COPY` commands.

Corruption Protection

Below are the optimal settings recommended by MAA Best Practices for key parameters and database features that enable corruption detection, prevention, and automatic repair in a Data Guard configuration. For more information on the level of protection provided by each of these settings and performance tradeoffs, see My Oracle Support Note 1302539.1.

On the Primary Database

- `DB_BLOCK_CHECKSUM=FULL`
- `DB_BLOCK_CHECKING=FULL or MEDIUM`
- `DB_LOST_WRITE_PROTECT=TYPICAL`
- Enable Flashback Technologies for fast point-in-time recovery from logical corruptions most often caused by human error and for fast reinstatement of a primary database following failover.

[7] http://download.oracle.com/docs/cd/E11882_01/server.112/e17022/rcmbackp.htm#SBYDB4989

On the Data Guard Physical Standby Database

- `DB_BLOCK_CHECKSUM=FULL`
- `DB_BLOCK_CHECKING=FULL or MEDIUM`
- `DB_LOST_WRITE_PROTECT=TYPICAL`
- Enable Flashback Technologies for fast point-in-time recovery from logical corruptions most often caused by human error and for fast reinstatement of a primary database following failover.
- Use Active Data Guard to enable Automatic Block Repair (Data Guard 11.2 and later releases).

ASM Redundancy

The standard MAA best practice for Exadata is to configure ASM high redundancy for best protection from data and storage failures. ASM high redundancy will maintain availability in the event of: double disk failures, disk failure while another Exadata storage cell is down for maintenance (such as during a rolling upgrade), and disk failure followed by a read error (sector failure) on the redundant disk.

The additional protection level offered by ASM high redundancy, however, can have accompanying tradeoffs for reduced storage capacity and additional I/O. MAA best practice allows for the configuration of ASM normal redundancy when a Data Guard standby database has also been deployed for failover purposes on a separate Exadata system, as an acceptable compromise between protection, capacity, and performance.

Oracle Exadata Hardware Assisted Resilient Data

Exadata implements all Oracle Hardware Assisted Resilient Data (HARD) specifications, providing a unique level of validation for Oracle block data structures. This Oracle-aware validation occurs at the storage level, prior to allowing a write to physical disk. HARD validation complements the protections implemented by the parameters settings listed above. HARD is automatically enabled on Exadata (it does require `DB_BLOCK_CHECKSUM`). HARD checks transparently handle all cases; including Oracle ASM disk rebalance operations and disk failures.

Data Guard and Hybrid Columnar Compression

Data Guard physical standby databases provide transparent support for Oracle Hybrid Columnar Compression (HCC). In addition to improving performance and saving on storage, HCC will decrease the amount of network bandwidth required by Data Guard by placing the post compression image into database redo. Table 1 shows the results of two tests, each loading the

same data into the same table on an Exadata X2-2 quarter rack and Oracle Database 11.2.0.3. HCC compression reduced the volume of redo transported by Data Guard by 78%.

TABLE 1. REDUCTION IN REDO VOLUME DUE TO HCC COMPRESSION

	TEST 1: NO HCC COMPRESSION	TEST 2: HCC COMPRESSION
Data Loaded	174,919 MB	33,889 MB
Total Redo generated	180,939 MB	40,172 MB
Elapsed Time	00:07:15.78	00:06:58.33
Reduction in redo volume using HCC		78% less redo using HCC

Note: Compression ratios will vary with workload, so results will vary from one database to the next.

HCC is a feature of the Enterprise Edition of Oracle Database 11g that is dependent on the underlying storage system. HCC has always been supported by Exadata storage. Sun ZFS and Axiom Pillar storage added support for HCC as of Oracle Database 11.2.0.3. While ZFS and Pillar storage can achieve the same storage savings as Exadata, they are not able to offload database processing and deliver the same high performance as Exadata storage.

Heterogeneous Data Guard Configurations and HCC

Oracle recommends that Data Guard primary and standby database both be deployed on Oracle storage whenever HCC is used (e.g. any combination of Exadata, Sun ZFS, and Pillar Axiom 600 storage). This enables read-only access to HCC compressed tables when using Active Data Guard, and instant access to HCC compressed tables following a switchover or failover.

If high performance is an objective and the primary database is hosted on Exadata storage, Oracle recommends that the standby database also be hosted on Exadata storage. This enables equivalent performance levels when either system operates as primary. In addition to the performance benefits of an Exadata standby, there are numerous approaches described in this paper for high return on investment on a standby Exadata system.

Oracle acknowledges that there are cases when a business decision is made to accept lower service levels after a failover or switchover in return for reducing the acquisition cost of a DR system. For example, an Exadata primary database using HCC can have a standby database deployed on a non-Exadata system that uses SUN ZFS or AXIOM storage (see My Oracle Support Note 413484.1 for limitations to support for heterogeneous Data Guard configurations).

Following a switchover or failover applications would have immediate access to HCC compressed tables as well as being able to load new data in HCC compressed format (HCC support is a function of the underlying Oracle storage).

In a second example where the primary database is on Exadata but HCC is not used, a standby database may be hosted on any third party storage and system (subject to the heterogeneous support limitations defined in My Oracle Support Note 413484.1).

Performance is the only extra consideration for supported heterogeneous configurations described above. Care must be taken to confirm that redo apply on non-Exadata system and/or storage can keep pace applying redo received from an Exadata primary system. Performance testing must also confirm that a standby system on non-Exadata system and/or storage can deliver acceptable production performance when a failover or switchover transitions it to the primary role.

Data Guard Primary using HCC and Standby Database on Third-Party Storage

Oracle is also occasionally asked if it is possible to utilize existing third-party storage to host standby databases for a primary database hosted on Oracle storage where HCC is used. In such a configuration, HCC compressed redo will be transported by the primary and applied at the standby on third party storage in compressed format. However, given that third-party storage is unable to support HCC, the following complications and restrictions arise:

- Active Data Guard cannot read HCC compressed tables on a third party storage. In read-only mode, any attempt to select from an HCC table at the standby will yield the following error:

```
ORA-64307: hybrid columnar compression is only supported in
tablespaces residing on Exadata, ZFSSA, or AXIOM storage
```

- Data Guard Snapshot Standby cannot read HCC compressed tables on third party storage
- Upon switchover or failover to a system using third party storage, HCC compressed tables will need to be decompressed before they can be accessed, lengthening recovery time, requiring substantially more disk space, and delivering less performance than the original primary database.

  To access HCC tables you must failover to the standby and decompress using 'alter table move' for each HCC table. During the alter table move operation the table will be locked.

```
SQL> recover managed standby database finish;
SQL> alter database commit to switchover to primary;
SQL> alter database open;
```

```
SQL> alter table <table_name> move <nocompress>;
```

Note: You may also convert HCC tables using the following compression options on non-Exadata storage: Basic Table Compression <`compress/compress basic`>, or OLTP compression <`compress for oltp`>. OLTP compression requires a license for the Oracle Advanced Compression Option.

- After a failover, you will not be able to easily reinstate the original primary database. Normally, once a failed primary is repaired and the database mounted, Flashback Database is used to flash it back to the SCN before the standby became the new primary. Data Guard can then resynchronize it using redo generated by the new primary database since the failover occurred. The complication with this use-case is that the redo from the `alter table <table_name> move nocompress;` operation will also decompress the compressed tables on the Exadata system. When a subsequent switchover returns the original primary to the primary role, data will need to be converted or reloaded into HCC format to complete the process of returning the primary database to its original state.

In light of the above complications and restrictions, Oracle strongly advises using Oracle storage for both primary and standby systems whenever HCC is used.

Physical Network Configuration

There are two options for configuring a standby database on Exadata: shared network interface (usual configuration) or dedicated network interface for Data Guard.

Shared Network Interface

Standby databases can use either an Oracle RAC VIP interface or a dedicated interface. In a typical configuration, the network interface used by client connections is the same as that used by Data Guard to transmit redo between primary and standby databases. Figure 1 provides an example configuration; with both client connections and Data Guard redo transport using the NET1 (eth1) Gigabit Ethernet interface of an Exadata system. Note that channel bonding is recommended for HA in the event a channel should fail. The additional channel only provides HA, there is no scalability achieved by using channel bonding.

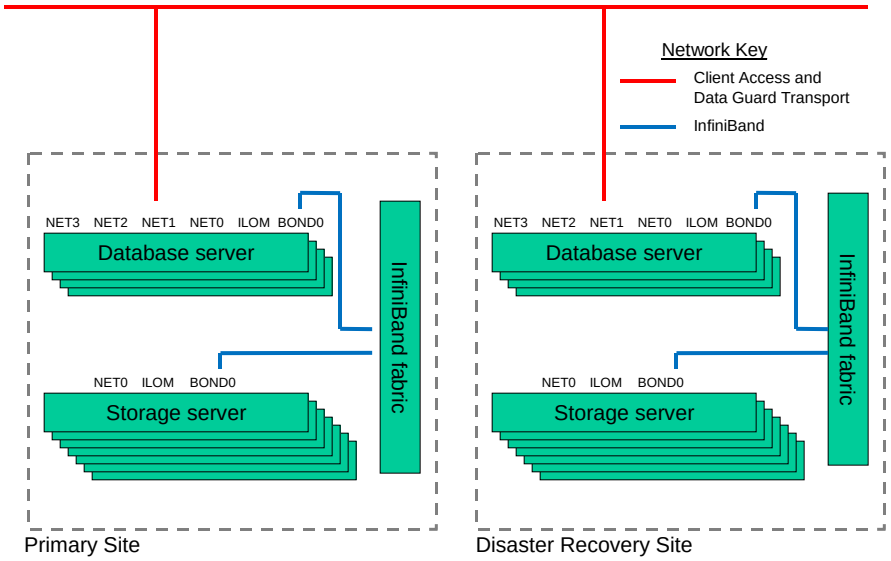

FIGURE 1, DATA GUARD NETWORK CONFIGURATION

Dedicated Network Interface for Data Guard Redo Transport

If you find that a single network interface is unable to handle both client and Data Guard network traffic without impacting primary database performance or the ability of redo transport to keep pace, consider isolating Data Guard traffic by configuring a private network interface. An example configuration using Gigabit Ethernet Interfaces is shown in Figure 2. See My Oracle Support Note 960510.1 for details.

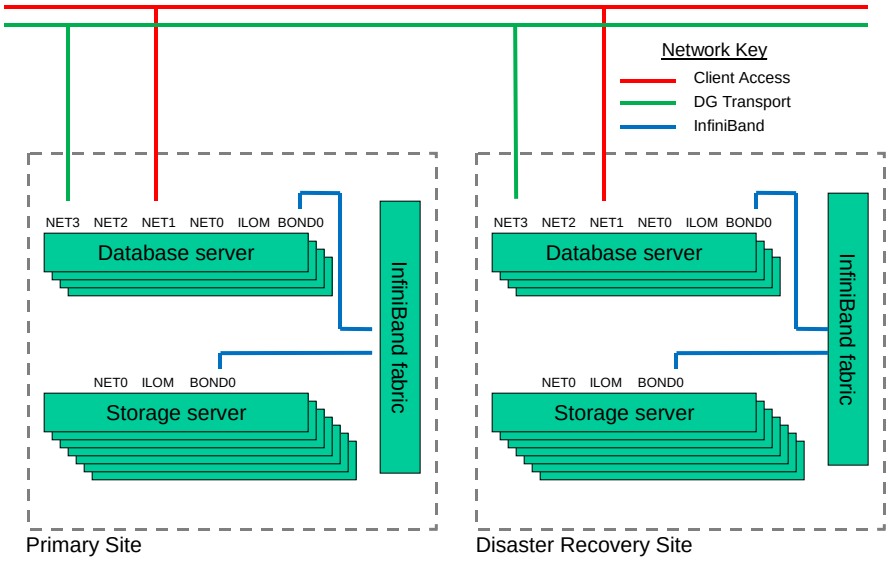

FIGURE 2, ISOLATE DATA GUARD REDO TRANSPORT FROM CLIENT ACCESS

Data Guard Network Best Practices

Standard Data Guard network best practices recommended for any Data Guard configuration are also applicable to Exadata. See Oracle High Availability Best Practices[8] documentation for more details.

Bandwidth Requirements

Network bandwidth between primary and standby systems must be sufficient for Data Guard to transport the volume of redo generated by the primary database. Inadequate network bandwidth will result in a transport lag that will increase exposure to data loss, compromising recovery point objectives (RPO). If it is not possible to procure sufficient bandwidth, you can transmit redo in

[8] http://docs.oracle.com/cd/E11882_01/server.112/e10803/config_dg.htm#CEGHIEIE

compressed format by using the Oracle Advanced Compression Option[9]. This can reduce bandwidth requirements significantly depending on your workload and the degree to which your data is already compressed. Refer to Data Guard 11g Release 2 documentation for information on enabling redo transport compression.[10] Refer to My Oracle Support Note 729551.1 for information on enabling redo transport compression for Data Guard 11g Release 1.

Network Tuning

There is one change to previously published Data Guard best practices that is very important to highlight. The previous recommendation was to set TCP Send/Receive buffer sizes equal to 3 x BDP (Bandwidth Delay Product = network bandwidth x round-trip network latency).

As of Oracle Database 11g, MAA best practice is to set TCP Send/Receive buffer sizes equal to 3 x BDP **or 10 Megabytes, whichever is greater**. This is required to realize maximum benefit from the new streaming protocol implemented for Data Guard 11g redo transport.

Synchronous Redo Transport - Zero Data Loss RPO

Data Guard synchronous redo transport with Maximum Availability protection mode is recommended for applications having a zero data loss RPO. Synchronous transport (SYNC) causes the primary database to wait a maximum of `NET_TIMEOUT` seconds for the standby to confirm that redo for the current transaction has been safely written to disk before acknowledging commit success to the application. If `NET_TIMEOUT` seconds are exceeded, Data Guard will cease attempting to send synchronously until the primary is able to reestablish connection with the standby database.

Data Guard 11g Release 2 synchronous transport will transmit redo to the remote standby in parallel with the local online log file I/O on the primary database (prior releases wrote to the online log file and then shipped in a serial operation). Assuming disk I/O performance is similar at both primary and standby systems, this limits the impact on primary database performance to the time required for the network round-trip (RTT) between the primary and standby.

Maximum Availability with SYNC is always recommended for ideal data protection if round-trip network latency (RTT) between primary and standby databases is less than 5 milliseconds. Higher RTT latency may still be acceptable for applications that are not as sensitive to the impact

[9] http://www.oracle.com/us/products/database/options/advanced-compression/index.html

[10] http://download.oracle.com/docs/cd/E11882_01/server.112/e10700/log_arch_dest_param.htm#SBYDB4902

of SYNC latency. Performance testing is always recommended when deploying Maximum Availability protection mode.

Asynchronous Redo Transport – Near-Zero Data Loss

Data Guard asynchronous redo transport with Maximum Performance protection mode is recommended when there is no zero data loss requirement or if the performance impact of RTT latency is too great to use Maximum Availability. Asynchronous redo transport (ASYNC) allows the primary database to acknowledge commit success to the application without waiting for confirmation that the redo has been received by the standby database. Maximum Performance avoids any impact to primary database response time.

Data Guard 11g ASYNC redo transport also achieves near zero impact on primary database performance. This is accomplished by shipping redo directly from the primary database log buffer instead of the Data Guard 10g practice of shipping from the online redo log file (ORL). To realize the maximum benefit from this enhancement be sure to allocate enough memory to the log buffer to prevent Data Guard from having to read from the ORL (e.g. when the volume of redo generated by the primary database is so high that log buffers are recycled before the redo can be shipped by Data Guard). See My Oracle Support Note 951152.1 for details on how to configure optimal log buffer size. If the ASYNC process must transition to reading from the ORL it will do so automatically, without any disruption to redo transport. Likewise, once it has caught up, it will automatically transition back to reading from the log buffer.

Configuring Automatic Client Failover

Partial Site Failure

A partial-site failure is where the primary database has become unavailable but the application tier at the primary site remains intact. If there is a local Data Guard standby database then all that is required to maintain availability is to redirect the application tier to the new primary database after a Data Guard failover. The same could also hold true when there is a remote Data Guard standby database if the surviving application tier can deliver acceptable performance using a remote connection after a database failover has occurred. Fast Application Notification (FAN) will break connected clients out of TCP timeout, and Transparent Application Failover (OCI

clients) or Fast Connection Failover (JDBC clients) will automatically fail clients over to database services running on the new primary database. See the technical white paper Client Failover Best Practices for Highly Available Oracle Databases: Oracle Database 11g Release 2 for more details[11].

Complete Site Failure

A complete-site failure results in both the application and database tiers being unavailable. To maintain availability users must be redirected to a secondary site that hosts a redundant application tier and a synchronized Data Guard standby database. MAA best practice is to maintain a running application tier at the standby site to avoid startup time and accelerate failover. A WAN traffic manager is used to execute a DNS failover (either manually or automatically) to redirect users to the application tier at standby site while a Data Guard failover transitions of the standby database to the primary production role. See Oracle Database High Availability Best Practices documentation for information on automating complete site failover.[12]

Reduce Overhead and Redo Volume During ETL Operations

Many customers using Exadata will do so for Data Warehouse applications that have high-volume load operations. Often there is extensive processing performed during extract, transform, and load (ETL) operations where interim results are temporarily stored in the database, and then final results are loaded into permanent tables. There is often no need to protect the interim results. If the job fails, it is resubmitted after the problem has been corrected.

The standard best practice for Data Guard configurations is to enable `FORCE LOGGING` at the database level (`ALTER DATABASE FORCE LOGGING;`) to ensure that all transactions are protected. However, placing the primary database in force logging mode for ETL operations can lead to unnecessary database overhead and extra processing by Data Guard to protect interim results that do not require protection.

MAA best practices call for isolating data that does not need to be protected by the standby database into their own tablespaces. Such data would include:

- Data resulting from temporary loads
- Data resulting from transient transformations

[11] http://www.oracle.com/technetwork/database/features/availability/maa-wp-11gr2-client-failover-173305.pdf

[12] http://download.oracle.com/docs/cd/B19306_01/server.102/b25159/outage.htm#BABHCAIA

- Any non critical data

To reduce unnecessary redo generation, do the following:

- Specifiy `FORCE LOGGING` for all tablespaces that you explicitly wish to protect (`ALTER TABLESPACE <name> FORCE LOGGING;`)
- Specify `NO FORCE LOGGING` for those tablespaces that do not need protection (`ALTER TABLESPACE <name> NO FORCE LOGGING;`).
- Disable force logging at the database level (`ALTER DATABASE NO FORCE LOGGING;`) otherwise the database level settings will override the tablespace settings.

Once the above is complete, redo logging will function as follows:

- Explicit no logging operations on objects in the no logging tablespace will not generate the normal redo (a small amount of redo is always generated for no logging operations to signal that a no logging operation was performed).
- All other operations on objects in the no logging tablespace will generate the normal redo.
- Operations performed on objects in the force logging tablespaces always generate normal redo.

Note: Take care when setting `NO FORCE LOGGING` at the tablespace level to avoid unintended nologging operations. Because the database level setting has been removed, new tablespaces added to the database will by default allow no logging operations. You must enable or disable force logging on all new tablespaces as required.

If space is needed at the standby database, then remove and exclude non-logged datafiles from recovery by using the `ALTER DATABASE DATAFILE OFFLINE DROP` statement.

The above procedure has no impact on normal Data Guard operation or Data Guard role transitions. Post role transition, there is minimal maintenance to recreate the objects in the no logging tablespace at the new primary database, if needed.

Best Practices for Planned Maintenance using Data Guard

A Data Guard physical standby database can be used to reduce downtime and risk when performing certain types of planned maintenance on an Exadata system. The general approach is to perform maintenance in rolling fashion, first implementing changes on the standby database, testing the changes, and then switching production from the primary to the standby. The only downtime is the time required to switch database roles and direct client connections to the new primary database.

Examples of such maintenance include:

Database Rolling Upgrades

A Data Guard physical standby database can be used to upgrade to new Oracle Database patch sets and major releases or to change the logical structure of a database, by using the transient logical database rolling upgrade process first introduced in Oracle Database 11g Release 1. This process is as follows:

- Begin with a physical standby database
- Temporarily convert the physical standby to a logical standby (SQL Apply). Note that this will not change the DBID of the standby database; it knows that the conversion is only temporary for the duration of the upgrade.
- Perform the standard process of upgrading the standby database to a new Oracle patchset or major release and then allow Data Guard to resynchronize the standby with the primary database. SQL Apply is used to maintain synchronization while primary and standby operate at different versions. Testing is performed to validate the upgrade was successful. Once complete, perform a switchover to transition the standby database to the primary role. .
- The original primary database is then flashed back (to the point in time when its physical standby was converted to a logical). It is mounted in a new Oracle home and becomes a physical standby of the new primary. A second catalog upgrade is not required because the database is upgraded and resynchronized via Redo Apply using the redo generated by the new primary (beginning with the SCN where it was temporarily converted to a logical standby).

Oracle provides MAA best practices[13] and a script that is downloadable from Oracle support via My Oracle Support Note 949322.1 that automates much of this process.

Patch Assurance using Standby-First Patching

Standby-First Patch Apply enables a physical standby to use Redo Apply to support different software releases between a primary database and its physical standby database for the purpose of applying and validating Oracle patches in rolling fashion (with a 'Standby-First' patch, there is no requirement to temporally convert the standby to a logical using SQL Apply).

Patches eligible for Standby-First patching include:

[13] http://www.oracle.com/technetwork/database/features/availability/maa-wp-11g-upgrades-made-easy-131972.pdf

- Oracle Exadata bundled patch
- Oracle Exadata Storage Server Software patch
- Patch Set Update (PSU)
- Critical Patch Update (CPU)
- Patch Set Exception (PSE)

Standby-First Patch Apply is supported for certified software patches for Oracle Database Enterprise Edition Release 2 (11.2) release 11.2.0.1 and later. Refer to My Oracle Support Note 1265700.1 for more information and the README for each patch to determine if a target patch is certified as being a Standby-First Patch.

Platform Migration and Technology Refresh

There are a number of options for migrating from legacy systems to Exadata. See MAA best practices for Exadata migration[14] to determine the best strategy for optimal tradeoff between application service levels, attributes and performance. Several of the options described in the best practice paper use a Data Guard standby database for migration.

See My Oracle Support Note 413484.1 for information on the heterogeneous primary/standby platform combinations supported in a Data Guard configuration.

An example of using a Data Guard standby database to minimize planned downtime during technology refresh is the upgrading from an HP Oracle Database Machine running Oracle Database 11g Release 1, to a SUN Oracle Exadata Database Machine running Oracle Database 11g Release 2. The upgrade of the Oracle Database and the complete replacement of the hardware platform are executed with less than 5 minutes of total downtime. For details, see My Oracle Support Note 1055938.1.

Data Center Moves

Create a Data Guard standby database in the new Data Center, and then switchover.

[14] http://www.oracle.com/technetwork/database/features/availability/xmigration-11-133466.pdf

Using Data Guard to Clone Test and Development Databases

Clone databases used for test and development are essential to supporting a production environment. These databases routinely need to be refreshed so that they accurately reflect the primary database. There are several options for creating clone databases in a Data Guard configuration with Exadata.

On Exadata Storage

Test and development databases are best deployed on Exadata on an environment that is identical to production. RMAN can be used for one-time creation of a clone database using the RMAN `DUPLICATE` command.[15] Alternatively, Data Guard Snapshot Standby can be used to create a clone database that is easily refreshed[16].

Data Guard Snapshot Standby enables a physical standby database to be open read-write for testing or any activity that requires a read-write replica of production data. A Snapshot Standby continues to receive, but not apply, updates generated by the primary. These updates are applied to the standby database automatically when the Snapshot Standby is converted back to a physical standby database (updates made to the Snapshot Standby while it was open read-write are discarded). Data Guard Snapshot Standby is also an ideal complement to Oracle Real Application Testing.[17]

On Sun ZFSSA Storage Appliance

In cases of insufficient storage capacity on Exadata to clone the production database, a second standby database can be created on a Sun ZFS Storage Appliance (ZFSSA) and added to an existing Data Guard configuration. Unlike the first standby database that resides on Exadata storage for failover purposes, the second standby database is used by ZFSSA as a source to quickly create and maintain multiple space-efficient copies of the production database to support development and test activities.

ZFSSA supports unlimited snapshot capability. A snapshot is a read-only, point-in-time copy of a file system. It is instantaneously created and no space is allocated initially. Blocks are allocated as changes are made to the base file system (copy-on-write). The snapshots are either initiated

[15] http://download.oracle.com/docs/cd/E11882_01/backup.112/e10642/rcmdupdb.htm#i1008564

[16] http://download.oracle.com/docs/cd/E11882_01/server.112/e17022/manage_ps.htm#SBYDB4801

[17] http://www.oracle.com/technetwork/database/features/manageability/index.html

manually or can be automated by scheduling at specific intervals. The snapshot data can be directly accessed for any backup purposes. Any reads to the snapshot blocks are served by the base file system's block. When changes happen to the base file system, the older block is now referenced by the snapshot and the new changed block is referenced by the file system.

ZFSSA also supports an unlimited number of clones. A clone is an instantaneously created read-write copy of a snapshot. One or more clones are created from a single snapshot. All the regular operations are allowed on clones including taking a snapshot from the clone. Clones are typically used in test, development, QA, and backup environments. Similar to snapshots, when the clone is created, no space is allocated. The reads to the clone are served by the base file system's blocks. The changed blocks are allocated only when the blocks are changed in the clone.

As of a patch available for Oracle Database 11.2.0.3, ZFSSA is able to support HCC. ZFSSA does not have the performance advantage of native Exadata Storage, but it is a cost-effective target for backups and for snapshots and clones.

See the MAA Best Practice paper, "Database Cloning using Oracle Sun ZFS Storage Appliance and Oracle Data Guard"[18].

On Third-Party Storage that Does Not Support HCC

There may be cases when migrating to Exadata when users wish to allocate legacy storage that formerly hosted the production database to development and test. As discussed in the HCC section of this paper, this presents challenges when the Exadata primary database is using HCC and the test and development database reside on third-party storage that is unable to support HCC. There are two approaches that can be used with Data Guard.

Option 1: Snapshot Standby on Third Party Storage

This option uses a Data Guard Snapshot Standby database that is open read-write as a development or test database, or to seed other development or test databases. The process begins by creating a physical standby on third party storage. The managed recovery process will apply changes to the HCC tables.

- In read only mode any attempt to select from the table will yield the following error:

[18] http://www.oracle.com/technetwork/database/features/availability/maa-db-clone-szfssa-172997.pdf

```
ORA-64307: hybrid columnar compression is only supported in
tablespaces residing on Exadata, ZFSSA, or AXIOM storage
```

- To access or modify the HCC tables convert the physical standby into a snapshot standby[19] and uncompress. Note that the table will be locked during the alter table move operation.

```
SQL> alter database convert to snapshot standby;

SQL> alter database open;

SQL> alter table <table_name> move <nocompress>;
```

  Note: You may also convert HCC tables using the following compression options on third party storage: Basic Table Compression <`compress/compress basic`>, or OLTP compression <`compress for oltp`>. OLTP compression requires a license for the Oracle Advanced Compression Option.

- After the HCC tables have been decompressed you can shutdown and copy the snapshot standby database to seed either test or development databases. Once you have finished the copying process convert the snapshot standby back into a physical standby. When a snapshot standby is converted back into a physical standby the HCC table that was decompressed in the above step will return to being HCC compressed and managed recovery will resume updating the standby database. Use the following command to convert the snapshot standby back to a physical standby:

```
SQL> alter database convert to physical standby;
```

Note: A Snapshot Standby uses guaranteed restore points to enable the conversion back to a physical standby database. Sufficient space must be allocated to the Fast Recovery Area of the standby database to accommodate the flashback logs that will be generated during the alter table move procedure.

Option 2: Logical Standby on Third Party Storage

This option begins by creating a physical standby on third party storage as in Option 1. The physical standby database is then converted into a logical standby using steps outlined in the Data Guard Concepts and Administration Guide[20]. SQL Apply will maintain and update HCC

[19] http://download.oracle.com/docs/cd/E11882_01/server.112/e17022/manage_ps.htm#SBYDB4801

[20] http://download.oracle.com/docs/cd/E11882_01/server.112/e17022/create_ls.htm#g105412

tables and data. Any attempt to access the data will result in the error described above (ORA-64307)

- To access or update the HCC data you must first decompress using 'alter table move'. Note that the table will be locked during the alter table move operation.

```
SQL> alter database stop logical standby apply;

SQL> alter table <table_name> move <nocompress>;

SQL> alter database start logical standby apply;
```

  Note: You may also convert HCC tables using the following compression options on third party storage: Basic Table Compression <compress/compress basic>, or OLTP compression <compress for oltp>. OLTP compression requires a license for the Oracle Advanced Compression Option..

- After the HCC tables are decompressed you can shutdown the logical standby database and use it to seed either test or development databases. When you restart the logical standby database, SQL apply will continue to maintain updates to the table from the primary and will automatically decompress the HCC redo and convert it to the compression format (none, BASIC, or OLTP) desired at the standby database. When you need to refresh the test or development database simply repeat the process of shutting down the logical standby database and refresh the dev/test databases.

Best Practices for Maximum ROI

Utilize your Data Guard standby databases for productive purposes while they are in standby role in order to maximize return on investment.

Maintaining a synchronized Data Guard standby database only requires a fraction of the CPU and memory available on the standby system. This leaves considerable capacity on the standby system to address other requirements, increasing ROI and effectively reducing the cost of DR. MAA best practices recommend the following methods for effective use of your standby systems.

- Use Active Data Guard to offload read-only workload to an active standby database to improve primary database performance while also providing HA/DR. It is significant to note that read-only queries on an Active Data Guard standby database have the same guarantee of read-consistency as queries executing on the primary database – no other physical replication solution can provide this level of read-consistency. (Active Data Guard is an option license for Oracle Database Enterprise Edition). For more details on Active Data Guard see the technical white paper, Active Data Guard 11g Best Practices[21].

  Any workload that can function with read-only access to the Oracle Database can be offloaded to an Active Data Guard standby. Offload is also supported for reporting purposes by some Oracle Applications and reporting tools. These include:

  - E-Business Suite Reporting (OracleReports)as of E-Business Suite Release 12.1.3 or later[22]
  - Peoplesoft PeopleTools 8.51[23]
  - Oracle Business Intelligence Enterprise Edition Server[24]
  - Oracle TopLink Applications[25]
- Use the standby database of offload backups from the primary database. Use Active Data Guard to offload fast incremental backups from the primary database using RMAN block change tracking.
- Use a single Exadata system as a 'standby hub', where it functions as a shared resource hosting multiple standby databases for primary databases located in two or more data centers. This consolidation strategy reduces DR cost based on the premise that there is a low likelihood of multiple simultaneous failures at different sites.
- Deploy production databases on Exadata systems in both your primary and DR sites, thus transforming all data centers into primary sites. In this configuration, each site hosts standby databases for the primary databases deployed at the other site, increasing system utilization while reducing the overall impact of an individual site failure. Beyond the load-balancing benefits, this strategy insures that all sites are truly capable of running production at failover

[21] http://www.oracle.com/technetwork/database/features/availability/maa-wp-11gr1-activedataguard-1-128199.pdf
[22] http://blogs.oracle.com/stevenChan/2011/01/adg_ebs12.html
[23] http://download.oracle.com/docs/cd/E18083_01/pt851pbr0/eng/psbooks/tadm/book.htm?File=tadm/htm/tadm13.htm#H4064
[24] http://www.oracle.com/technetwork/database/features/availability/maa-wp-11g-biee-activedataguard-1-131999.pdf
[25] http://www.oracle.com/technetwork/database/features/availability/maa-tech-wp-toplinkwithadg-130077.pdf

time given that their normal routine is to function as a production site 24x365. While there may be constraints that inhibit easily adopting this strategy, the payback is significant.

Best Practices for Managing a Data Guard Configuration

Oracle Enterprise Manager Grid Control makes use of the underlying services of the Data Guard Broker[26] to simplify the creation and management of a Data Guard configuration. Wizards are used for easy instantiation of a standby database. A mouse-click invokes database switchover and failover operations, or you can configure automatic failover (Data Guard Fast-Start Failover) if preferred. Enterprise Manager consolidates the reporting of key Data Guard metrics such as apply lag, transport lag, redo rate and configuration status in a single window called the HA Console. Grid Control also enables historical trend analysis on the Data Guard metrics that it monitors - for example, how the metric's performance has been in the last 24 hrs, or the last 5 days, and so on. Also, through Enterprise Manager, it is possible to set up notification alarms such that administrators are notified in case a metric crosses the configured threshold value.

The Data Guard Broker is a distributed management framework that is included with Data Guard and is required by Enterprise Manager Grid Control. Administrators, if they prefer, may also interact directly with the broker using the broker's command line interface (DGMGRL).

Conclusion

Data Guard Redo Apply is the simplest, fastest and most reliable solution for maintaining an independent, synchronized physical replica of the Oracle Database for HA/DR.

Data Guard:

- Supports both High Availability (with Zero Data Loss and/or Automatic Failover) and Disaster Recovery.
- Provides unique levels of data protection and availability and is the only DR technology able to support the very high transaction volumes driven by Exadata.

[26] http://www.oracle.com/pls/db112/to_toc?pathname=server.112/e17023/toc.htm

- Is truly a drop-in solution for Oracle Database. Data Guard Redo Apply is the only Oracle-aware DR technology that natively supports all Oracle datatypes, database features and workloads.
- Is a tool for minimizing planned downtime, and can be used to migrate to Exadata and to execute database upgrades in rolling fashion.
- Has very few special considerations that need to be made when used with Exadata. You can directly leverage past Data Guard knowledge and experience as you upgrade to this high performance platform.
- Offers a number of deployment options, including Active Data Guard, for productively using standby systems while in standby role, yielding high return on investment.

Data Guard is Oracle's HA/DR solution prescribed by the Maximum Availability Architecture (MAA) to protect mission critical databases residing on Exadata Database Machine.

Oracle Data Guard: Disaster Recovery for
Oracle Exadata Database Machine
April 2012
Authors: Joseph Meeks, Larry Carpenter,
Michael Smith
Contributing Authors: MAA team

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
oracle.com

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

0109