Hybrid Data Guard to Oracle Cloud Infrastructure

Production Database on Premises and Disaster Recovery with DBaaS BM or VM shapes in Oracle Cloud Infrastructure

ORACLE WHITE PAPER | APRIL 2019

Introduction

Oracle’s Maximum Availability Architecture (Oracle MAA) is the best practices blueprint for data protection and availability for Oracle databases deployed on private, public or hybrid clouds. Data Guard and Active Data Guard provide disaster recovery (DR) for databases with recovery time objectives (RTO) that cannot be met by restoring from backup. Customers use these solutions to deploy one or more synchronized replicas (standby databases) of a production database (the primary database) in physically separate locations to provide high availability, comprehensive data protection, and disaster recovery for mission-critical data.

An effective disaster recovery plan can be costly due to the need to establish, equip and manage a remote data center. The Oracle Cloud offers a great alternative for hosting standby databases for customers who do not have a DR site or who prefer not to deal with the cost or complexity of managing a remote data center. Existing production databases remain on-premises and standby databases used for DR are deployed on the Oracle Cloud. This mode of deployment is commonly referred to as a hybrid cloud implementation.

Customers may choose to deploy either a Data Guard or an Active Data Guard standby on the cloud depending upon their requirements. While there are some unique considerations to a hybrid cloud DR configuration, it follows the same Oracle MAA best practices as with any Data Guard deployment. This Oracle MAA blueprint details Oracle MAA Best Practices and provides a procedural overview for deploying DR on the Oracle Cloud using Database as a Service. This paper is intended for a technical audience having knowledge of Oracle Database, Data Guard or Active Data Guard, and Oracle Database backup and recovery. This paper also assumes a basic understanding of services offered by the Oracle Cloud Infrastructure.

Disaster Recovery to the Cloud with Data Guard and Active Data Guard

The Oracle Cloud offers an extensive set of cloud services tailored to specific customer requirements: Platform as a Service (PaaS), Infrastructure as a Service (IaaS), and Software as a Service (SaaS). Disaster Recovery (DR) for on-premises systems is deployed using the Oracle Cloud Infrastructure Database as a Service (OCI DBaaS).

There are three options for DR to the cloud using OCI DBaaS:

- Data Guard utilizing Enterprise Edition Service or High-Performance Service.
- Data Guard utilizing the Extreme Performance Service for BYOL cases where no ADG license exists. (recommended)
- Active Data Guard utilizing the Extreme Performance Service or Exadata Service. (recommended)

Data Guard is included in Oracle Database Enterprise Edition (no separate license is required for on-premises systems) and is supported by all editions of OCI DBaaS (Enterprise, High Performance, and Extreme Performance). Please bear in mind standard edition does not support Data Guard. Please search “Licensing Information” in the Oracle Database Documentation for the selected version to understand the supportability matrix on what is supported in Enterprise Edition versus Standard Edition.

Active Data Guard extends Data Guard capabilities by providing advanced features for data protection and availability as well as offloading read-only workload and fast incremental backups from a production database. Active Data Guard is included in the Extreme Performance Edition and Exadata Service. When used in a hybrid configuration, Active Data Guard must also be licensed for the on-premises system.

MAA recommends

1. Create an OCI standby database target that is symmetric or similar to on-premise primary database to ensure you meet the same performance SLAs after a role transition. Use RAC for RAC, Exadata for Exadata, etc.
2. Ensure network bandwidth is enough to handle peak redo rates
3. Ensure network reliability and security between on-premise to and from OCI cloud
4. Use Active Data Guard for additional auto-block repair, data protection and offloading benefits
5. Use Oracle Transparent Data Encryption (TDE) for both primary and standby databases

Benefits of Hybrid Standby in the Cloud

1. Cloud data center and infrastructure is managed by Oracle
2. Cloud provides basic system life cycle operations including bursting and shrinking resources
3. Oracle Data Guard provides disaster recovery, data protection and ability to offload activity for higher utilization and return on investment.
4. When configured with MAA practices, RTO of seconds with automatic failover when configured with Data Guard Fast-Start failover and RPO less than a second for Data Guard with ASYNC transport or RPO zero for Data Guard in SYNC or FAR SYNC configurations

Data Guard Life Cycle Management e.g. switchover, failover and reinstate is a manual process in Hybrid Data Guard configurations.

Enabling Disaster Recovery on the OCI DBaaS

Enabling DR on the cloud requires instantiation of a Data Guard standby database in the Oracle OCI DBaaS. Once instantiated, Data Guard maintains synchronization between the primary database on-premises and the standby database in the cloud.

- Utilization of the same Oracle MAA best practices as on-premises deployment. Use of additional Oracle MAA best practices specific to hybrid cloud deployments that are specified in this paper.
- Ability to switchover (planned events) or failover (unplanned event) production to the standby database in the cloud during planned maintenance or unplanned outages. Once the failed on-premises database is repaired, the ability to automatically resynchronize it with the new production database in the cloud and then switch production back to the on-premises database.

Service Level Requirements

Hybrid cloud deployments are by definition user-managed environments. The administrator must determine service level expectations for availability, data protection, and performance that are practical for a given configuration and application. Service Levels must be established for each of three dimensions relevant to disaster recovery that are applicable to any Data Guard configuration:

- **Recovery Time Objective (RTO)** describes the maximum acceptable downtime should an outage occur. This includes the time required to detect the outage and to failover both the database and application connections so that service is resumed.
- **Recovery Point Objective (RPO)** describes the maximum amount of data loss that can be tolerated. Achieving the desired RPO depends upon:
  - Available bandwidth relative to network volume.
  - The ability of the network to provide reliable, uninterrupted transmission.
  - The Data Guard transport method used: either asynchronous for near-zero data loss protection or synchronous for zero data loss protection.
- **Data Protection:** With Active Data Guard and MAA configuration, customers can configure the most comprehensive block corruption detection, prevention and auto-repair.
- **Performance:** Database response time may be different after failover if less capacity – compute, memory, I/O, etc, are provisioned at the standby system than in the on-premises production system. This occurs when administrators purposefully under-configure standby resources to reduce cost; accepting reduced service level while in DR mode. MAA best practices recommend configuring symmetrical capacity at both primary and standby so there is no change in response time after failover. Rapid provisioning available with the cloud can enable a middle ground where there is less capacity deployed steady-state, but the new primary is rapidly scaled-up should a failover be required.

Security Requirements

Oracle MAA best practice recommends using Oracle Transparent Data Encryption (TDE) to encrypt both primary and standby databases to ensure all data is encrypted at-rest. Data can be converted during the migration process but it's highly recommended to convert to TDE prior to migration to provide the most secure Data Guard environment. Refer to Oracle Database Tablespace Encryption Behaviour in Oracle Cloud (Doc ID 2359020.1) for more information. VPN connection or Oracle Net encryption is also required for encryption-in-flight for any other database payload (e.g. data file or redo headers) that are not encrypted by TDE.

Using TDE to protect data is an important part of improving the security of the system. Users should, however, be aware of certain considerations when using any encryption solution, including:

- Additional CPU overhead: Encryption requires additional CPU cycles to calculate encrypted and decrypted values. TDE, however, is optimized to minimize the overhead by taking advantage of database caching capabilities and leveraging hardware acceleration within Exadata. Most TDE users see little performance impact on their production systems after enabling TDE. If performance overhead is a concern, please refer to the Oracle Database Advanced Security Guide.
- Lower data compression: Encrypted data compresses poorly because it must reveal no information about the original plaintext data. Thus, any compression applied to TDE encrypted data will have low compression ratios. Hence, when TDE encryption is used, it is not recommended to use with redo transport. However, when TDE is used in conjunction with Oracle databases compression technologies such as Advanced Compression or Hybrid Columnar Compression, compression is performed before the encryption occurs, and the benefits of compression and encryption are both achieved.
- Key management: Encryption is only as strong as the key used to encrypt. Furthermore, the loss of the encryption key is tantamount to losing all data protected by that key. If encryption is enabled on a few databases, keeping track of the key and its lifecycle is relatively easy. As the number of encrypted databases grows, managing keys becomes an increasingly difficult problem. For users with a large number of encrypted databases, it is recommended that Oracle Key Vault be used on-premise to store and manage TDE master keys.

Database, OS Environment and Network Prerequisites

TABLE 1: PREREQUISITES

	On-Premises	Oracle Cloud Infrastructure DBaaS (OCI DBaaS)
Operating System	Linux, Windows or Solaris X86 (My Oracle Support Note 413484.1 for Data Guard cross-platform compatibility)	Oracle Enterprise Linux (64-bit)
Oracle Database*	» *Oracle Database Enterprise Edition 11.2.0.4 (64-bit)* » *Oracle Database Enterprise Edition 12.1.0.2 (64-bit)* » *Oracle Database Enterprise Edition 12.2.0.1 (64-bit)* » *Oracle Database Enterprise Edition 18c (64-bit)* » *Oracle Database Enterprise Edition 19c (64-bit)*	Database as a Service: Enterprise, High Performance and Extreme Performance editions Active Data Guard: Database as a Service (or) Virtual Image: Extreme Performance Edition (or) Exadata Cloud Service
RAC	RAC or non-RAC	RAC or non-RAC
Multitenant	For 12.1 and above, primary database has to be a CDB/PDB database.	Multitenant Database
Physical Vs Virtual	Physical or Virtual	Physical (BM Shapes) or Virtual (VM Shapes)
Database Size	Any Size	Any size. For shape limits please consult documentation
TDE Encryption	Mandatory for both primary and standby	Mandatory for both primary and standby

If On-Premises is not already enabled with TDE, please follow the master note Master Note for Transparent Data Encryption (TDE) (Doc ID 1228046.1) to enable TDE and create wallet files.

** Oracle Database version on primary and standby databases must match during initial instantiation. For database software updates that are standby-first compatible, the primary and standby database Oracle Home software can be different. Refer to Oracle Patch Assurance - Data Guard Standby-First Patch Apply (Doc ID 1265700.1)*

Using Standby Database to reduce downtime during Planned Maintenance

There are several options for utilizing a standby database on the cloud for reducing planned downtime of the primary production database:

Standby-first Patch Apply

Many patches may be applied first to a physical standby for thorough validation. Customers who wish to minimize downtime will frequently patch the standby first, then switch production to the standby database, and then patch the original primary. If the primary and standby are RAC and the software update is RAC rolling, a switchover is not required; however, it is still recommended to update the software on the standby-first for additional validation and protection. Data Guard physical replication is supported between primary and standby running at mixed patch versions for patches that are standby-first eligible (documented in the patch readme). The customer may also choose to run for a period of time with mixed patch versions between primary and standby to enable fast fallback to the unpatched version should there be any unanticipated problems with the patch. See My Oracle Support Note 1265700.1, “Oracle Patch Assurance - Data Guard Standby-First Patch Apply” for more details on patches eligible for the standby-first process.

Database Rolling Upgrade

Another beneficial use case for standby in the OCI DBaaS is for database rolling upgrade to reduce downtime when upgrading to new database Oracle releases that are not standby-first compatible. The transient logical process is used in Oracle 11g and Oracle 12c to temporarily convert a physical standby database to a logical standby, upgrade the logical standby to the new version, validate and when ready execute a Data Guard switchover. After the switchover completes, the original primary database is converted to a synchronized physical standby also operating at the new release. Refer to Oracle 11g Database Rolling Upgrades Made Easy or Oracle 12c DBMS_Rolling for more information.   A more efficient database rolling upgrade process using the standby database exists for Data Guard environments 12.2 and higher.   Refer to Using DBMS_ROLLING to Perform a Rolling Upgrade section in the Data Guard documentation.

Data transfers from on-premises to Oracle Cloud use the public network, VPN and/or the high bandwidth option provided by Oracle FastConnect.

OCI Network Prerequisite

In a Data Guard configuration, the primary and standby must be able to communicate bi-directionally. This requires additional network configuration to allow access to ports between the systems.

Secure Connectivity

There are two options to privately connect your cloud network to the on-premises network, IPSec VPN and FastConnect. Each of these methods require a Dynamic Routing Gateway (DRG) to connect to your Virtual Cloud Network (VCN). Details for creating a DRG can be found in the documentation at this link.

IPSec VPN

IPSec stands for Internet Protocol Security or IP Security. IPSec is a protocol suite that encrypts the entire IP traffic before the packets are transferred from the source to the destination. For an overview of IPSec in OCI review the documentation here.

FastConnect

Oracle Cloud Infrastructure FastConnect provides a method to create a dedicated, private connection between your data center and Oracle Cloud Infrastructure. FastConnect provides higher-bandwidth options and a more reliable and consistent networking experience compared to internet-based connections. More details on FastConnect can be reviewed here.

Public Internet Connectivity

Connectivity between OCI and on-premises can be achieved through the public internet as well. This method is not by default secure and additional steps must be taken to secure transmissions. The remained of this whitepaper assumes public internet connectivity.

By default, cloud security rule for port 1521 is disabled. Also, this default pre-configured port in the cloud VM/BM has open access from the public internet.

1. If Virtual Cloud Network for the standby database doesn't have Internet Gateway, you have to add Internet Gateway. The below document provides how to create internet gateway: https://docs.us-phoenix-1.oraclecloud.com/Content/Network/Tasks/managingIGs.htm
2. Edit Ingress and Egress rule in security list to connect from/to on-premise database. The link below provides additional information. https://docs.us-phoenix-1.oraclecloud.com/Content/Network/Concepts/securitylists.htm

On-Premises Network Configuration

In a Data Guard configuration, information is transmitted in both directions between primary and standby databases. This requires basic configuration, network tuning and opening of ports at both primary and standby databases.

On-Premises Prerequisites

The following prerequisites must be met before instantiating the standby database:

- Name resolution to the OCI DBaaS VM needs to be configured. This can be done either through a static file like /etc/hosts or configuring the on-premises DNS to properly resolve the public IP address of the OCI instance. Also, the on-premises Firewall will need to have Access Control Lists properly configured to allow SSH and Oracle Net to be accessed from the on-premises system to the OCI DBaaS VM.
- Since Data Guard in a DR situation requires access from the cloud instance to the on-premises database, the primary database listener port must be opened with restricted access from the cloud IP addresses using features like iptables etc. Since every corporation has different network security policies, the network administrator will need to perform operations similar to the cloud-side network configuration shown in preceding sections
- Prompt-less SSH from the OCI DBaaS VM to the On-Premises machine. This is configured both for on-premises to the Cloud during the provisioning process and from the Cloud to on-premises.
- The configuration of the On-Premises firewall to allow inbound SSH connectivity from the OCI DBaaS VM to the On-Premises machine.
- The Oracle Home for the on-premises database must be the same Oracle patch set as the standby database. If the OCI environment is on a different bundle patch level, and the on-premise database is on a different bundle patch or PSU, it is recommended to patch the source environment to the same database bundle patch level as the database home in the cloud environment. (The command “$ORACLE_HOME/OPatch/opatch lspatches” can be executed to check the one-off patches installed on both Source and Target environments)
- The steps outlined in this document assume that the on-premises primary database is not already part of an existing Data Guard broker configuration. If there is an existing broker configuration for the on-premises database it is assumed that the administrator has prior knowledge of the broker and knows how to add the new standby database to an existing broker configuration. A value other than ‘NOCONFIG’ for the following query implies an existing broker configuration.

```
SQL> select decode(count(1),0,'NOCONFIG') from v$DG_BROKER_CONFIG;
```

- Use the default listener named LISTENER. The steps outlined in this document assume the default listener name LISTENER is used. To verify run the following command from the on-premises machine. The expected result is shown.

```
$lsnrctl show current_listener | grep ‘Current Listener’ Current Listener is
 LISTENER
```

- Verify the listener port by running the following command from the on-premises machine. The expected result is shown.

```
$lsnrctl status| grep 'Connecting to'
Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=(1521)))
```

Implement MAA Best Practice Parameter Settings on the Primary Database

See Appendix B for a list of best practices. Completing this process on the primary database before instantiation is recommended. Especially configuring the redo logs which will be duplicated during the process.

Validating Connectivity between On-Premises and OCI DBaaS Host

Once all the steps above are implemented successfully, run the command below to validate if the connection looks good from both source to target and vice versa. If telnet is show success, proceed to the next step.

ON ON-PREMISE HOST

```
[root@onpremise1 ~]# telnet xxx.xxx.xxx.xxx 1521
Trying xxx.xxx.xxx.xxx...
Connected to xxx.xxx.xxx.xxx.
Escape character is '^]'.
^C^]q
telnet> q
Connection closed.
```

ON OCI DBAAS HOST

```
[root@oci2 ~]# telnet xxx.xxx.xxx.xxx 1521
Trying xxx.xxx.xxx.xxx...
Connected to xxx.xxx.xxx.xxx.
Escape character is '^]'.
^]q
telnet> q
Connection closed.
```

Deployment Process

This procedure is basically same as migrating the database from on-premise to OCI and the Data Guard setup for a Single Instance (SI) or RAC should be the same. Customers have an option to setup data guard from an SI on-premises to a 2-Node RAC in cloud infrastructure or RAC on-premises to an SI in the cloud infrastructure. Steps to convert an SI to RAC are beyond the scope of this whitepaper.

https://docs.us-phoenix-1.oraclecloud.com/Content/Database/Tasks/mig-rman-duplicate-active-database.htm

PREREQUISITES

The procedure below applies to both Bare Metal and VM DBaaS shapes with the only difference, that the VM service only supports 1 database per VM shape. It is recommended to create the database with the same name as the on-premise database. While the DB Name should be the same, the db_unique_name must be different to the primary system. Patch level of the source database's Oracle home should be the same as the database on VM DBaaS.

TABLE 2: DBAAS EDITIONS

Oracle Database Software Edition of OCI DBaaS	Including License
Standard Edition	Oracle Database Standard Edition 2. *Data Guard does not support Standard Edition
Enterprise Edition	Oracle Database Enterprise Edition, Data Masking and Subsetting Pack, Diagnostics and Tuning Packs, and Real Application Testing.

Enterprise High Performance	Extends the Enterprise Edition with the following options: Multitenant, Partitioning, Advanced Compression, Advanced Security, Label Security, Database Vault, OLAP, Advanced Analytics, Spatial & Graph, Database Lifecycle Management Pack and Cloud Management Pack for Oracle Database.
Enterprise Extreme Performance	Extends the High-Performance package with the following options: In-Memory Database, Active Data Guard, and RAC (requires two VMs of at least two OCPUs each)

Step 1: Create DB System VM or BM

Follow the steps defined in the link below to launch a DB System. Name the database the same as the primary on-premises database.

https://docs.cloud.oracle.com/iaas/Content/Database/Tasks/launchingDB.htm

If using Bare Metal as a standby database, you can create a new database with the same name as the on-premises. Database name and version should be the same as the source database.

Step 2: Manually Delete the Database Created by Tooling

Once the database is provisioned and ready, please perform the below operations to delete the starter database files and we will restore the on-premises database using RMAN. The steps to initiate the RMAN duplicate will be described later in this document.

To delete the starter database, use the manual method of removing the database files from ASM disk groups. Do not use DBCA as this will also remove the srvctl registration as well as the /etc/oratab entries which should be retained for the standby.

To manually delete the database on the OCI DBaaS host, run the steps below

1. Get the current db_unique_name for the OCI DBaaS database. This will be used throughout the remaining steps.

```
$ srvctl config database
```

2.Create a script to remove all database files

```
SQL> set heading off linesize 999 pagesize 0 feedback off trimspool on
SQL> spool /tmp/files.lst
SQL> select 'asmcmd rm '||name from v$datafile
union all
select 'asmcmd rm '||name from v$tempfile
union all
select 'asmcmd rm '||member from v$logfile;
SQL> spool off
```

```
SQL> create pfile='/tmp/<standby DB_UNIQUE_NAME>.pfile' from spfile;  #Backup spfile
```

```
$chmod 777 /tmp/files.lst
```

3. Shutdown the database

First collect the configuration of the database for future reference:

```
$ srvctl config database -d <db_unique_name> /tmp/<standby db_unique_name>.config
```

Then stop the database:

```
$ srvctl stop database -d <db_unique_name> -o immediate
```

4. Remove database files

Remove the existing data files, log files and tempfile(s). The password file will be replaced and the spfile will be reused.

As grid user (sudo from opc user to grid user)
Edit /tmp/files.lst created previously to remove any unneeded lines from sqlplus. Leaving all lines beginning with 'asmcmd'.

Save and execute the script

```
[grid@<host> ~]$ . /tmp/files.lst
```

All files for the starter database have now been removed.

Step 3: Copy the Password File to the OCI DBaaS host.

Copy password file on-Premises to $ORACLE_HOME/dbs/orapw<ORACLE_SID> on OCI DBaaS host.

Check password file location

If Oracle Grid Infrastructure is running on-Premises host, check password file location by "srvctl config database -db <db_name>" If Oracle Grid Infrastructure isn't configured or password file location is null, password file exists in default location($ORACLE_HOME/dbs/orapw<ORACLE_SID>)

```
$ srvctl config database -db testdbname
Database unique name: testdbname
Database name:
Oracle home: /u02/app/oracle/product/12.1.0/dbhome_2
Oracle user: oracle
Spfile: +DATA/testdbname/spfiledbtestdbname.ora
Password file: +DATA/testdbname/PASSWORD/orapw<sid> <======== password file location
Domain: domainname.xxxx.xxxx
```

Copy password file on-premises to OCI DBaaS host

If OCI DBaaS is RAC database, then copy password file to all nodes on OCI DBaaS. If password file location is non-ASM, copy the file as below.

```
$ scp -i <ssh key for OCI DBaaS Host> $ORACLE_HOME/dbs/orapw<SID> opc@<Public-IP-OCI-HOST>:~
```

If password file location is ASM, switch user to "grid" or the ASM owner, source the environment variables and then copy the password file as below.

```
$ sudo su - grid
$ export ORACLE_SID=<ASM ORACLE_SID>
$ export ORACLE_HOME=<GRID_HOME>
$ asmcmd
ASMCMD> cd +<DISKGROUP_NAME>/<DB_UNIQUE_NAME>/PASSWORD
ASMCMD> cp orapw<SID> /tmp
copying +DATA/<DB_UNIQUE_NAME>/PASSWORD/orapw<sid> -> /tmp/orapw<sid>

scp -i <ssh key for OCI DBaaS Host> /tmp/orapw<SID> opc@<Public-IP-OCI-HOST>:/tmp
```

Copy password file to $ORACLE_HOME/dbs/orapw<SID> on OCI DBaaS host

```
As opc user on OCI DBaaS host
$ chmod 777 /tmp/<password file name>
$ sudo su – oracle

For single instance OCI DBaaS place the password file in $ORACLE_HOME/dbs
$ cp /tmp/<password file name> <ORACLE_HOME>/dbs/orapw<SID>

For RAC DBaaS configurations place the password file in ASM:
As grid user:
ASMCMD> pwcopy --dbuniquename <standby DB_UNIQUE_NAME> /tmp/pwdvictor.858.1001423135
+DATA/victor_phx2w5/orapwvictor_phx2w5 –f

NOTE: if you receive ASMCMD-9453: failed to register password file as a CRS resource
then as the oracle user execute the following:
$ srvctl modify database –d <standby DB_UNIQUE_NAME> -pwfile
'+DATA/victor_phx2w5/orapwvictor_phx2w5'

Verify the password file is registered correctly (as oracle):
$ srvctl config database –d <standby DB_UNIQUE_NAME>
```

Step 4: Copying the wallet file to the OCI DBaaS host.

Make sure that $ORACLE_HOME/network/admin/sqlnet.ora contains the following line wallet file location is defined as ENCRYPTION_WALLET_LOCATION parameter in sqlnet.ora

SQLNET.ORA on on-premise host

```
ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)
(METHOD_DATA=(DIRECTORY=/opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME)))
```

The ORACLE_UNQNAME environment variable is set in clusterware and is the same as DB_UNIQUE_NAME of the standby database. Setting this variable in the oracle user environment is recommended.

Copy the ewallet.p12 and cwallet.sso files on-Premises to the above directory on OCI DBaaS host.

ON ON-PREMISE HOST

```
scp -i ~/<ssh_key> ewallet.p12 opc@<Public-IP-OCI-HOST>:/tmp
scp -i ~/<ssh_key> cwallet.sso opc@<Public-IP-OCI-HOST>:/tmp
```

Remove old wallet files in /opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME. If OCI DBaaS is RAC database, do the commands only on node1. (VM RAC nodes shared storage under /opt/oracle/dcs/commonstore using ACFS)

ON OCI DBAAS HOST

```
$ chmod 777 /tmp/ewallet.p12
$ chmod 777 /tmp/cwallet.sso
$ sudo su – oracle
$ cp /tmp/ewallet.p12 /opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME
$ cp /tmp/cwallet.sso /opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME
chmod 600 /opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME/*wallet*
```

Step 5: Configure static listeners

A static listener is needed for initial instantiation of a standby database. The static listener enables remote connection to an instance while the database is down in order to start a given instance. See MOS 1387859.1 for additional details.

As the grid user, add the following entry to the listener.ora on both the cloud DBaaS and on-premise host after replacing the variables. The listener.ora resides in $ORACLE_HOME/network/admin.

LISTENER.ORA

```
SID_LIST_LISTENER =
  (SID_LIST = (SID_DESC =
    (GLOBAL_DBNAME = <DB_UNIQUE_NAME of the OCI database>) (ORACLE_HOME = <Local
Oracle Home>)
    (SID_NAME = <ORACLE SID of the local instance>)
  ))
```

For 11.2 configurations, a static listener is also required for Data Guard Broker. Add the following entry to the listener.ora on premises after replacing the variables. For RAC this must be done on both nodes.

```
SID_LIST_LISTENER =
  (SID_LIST = (SID_DESC =
    (GLOBAL_DBNAME = <DB_UNIQUE_NAME of the OCI database>_DGMGRL)
    (ORACLE_HOME = <Local Oracle Home>) (SID_NAME = <ORACLE SID of the local
instance>)
  ))
```

Finally reload the listener (as grid user):

```
$ORACLE_HOME/bin/lsnrctl reload
```

Start the Standby Instance

srvctl start instance -d <standby DB_UNIQUE_NAME> -i <standby instance name> -o mount

Step 6: Oracle Net Encryption and TNS Entries for Redo Transport

To protect from plaintext or redo from unencrypted tablespaces from being visible on the WAN place the following entries in the sqlnet.ora file on all on-premises and cloud machines which are located in $ORACLE_HOME/network/admin.

SQLNET.ORA ON ON-PREMISE HOST

```
SQLNET.ENCRYPTION_SERVER=REQUIRED
SQLNET.CRYPTO_CHECKSUM_SERVER=REQUIRED
SQLNET.ENCRYPTION_TYPES_SERVER=(AES256,AES192,AES128)
SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER=(SHA1)
SQLNET.ENCRYPTION_CLIENT=REQUIRED
SQLNET.CRYPTO_CHECKSUM_CLIENT=REQUIRED
SQLNET.ENCRYPTION_TYPES_CLIENT=(AES256,AES192,AES128)
SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT=(SHA1)
```

Entries for each database are needed in both primary and standby tnsnames.ora files for proper redo transport. Use the following example, replacing values relevant to the configuration.

***NOTE**: The primary database may already have a TNS entry in the on-premises tnsnames.ora with a server name for the HOST. In this case, simply change the server name in that entry to use the IP address for the host instead.*

***NOTE**: IP addresses are used since there is no DNS between on-premises and cloud environments to resolve server names to IP addresses.*

***NOTE:** RAC configurations cannot resolve the scan listener name thus an address list must be used listing all nodes. Those will be added later, the initial instanton should only list one IP in the tns entries to ensure RMAN is always connecting to the same nodes.*

TNSNAMES.ORA ON ON-PREMISE HOST

```
<standby db_unique_name> = (DESCRIPTION =
  (SDU=65536) (RECV_BUF_SIZE=134217728)
  (SEND_BUF_SIZE=134217728)
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = <standby IP address>)(PORT = {1521|<port#>}))
  )
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = <standby db_unique_name>[.<standby domain name>])
    (UR=A)
  ))
```

TNSNAMES.ORA ON OCI DBAAS HOST

```
<primary db_unique_name> = (DESCRIPTION =
  (SDU=65536) (RECV_BUF_SIZE=134217728)
  (SEND_BUF_SIZE=134217728)
```

```
 (ADDRESS_LIST =
   (ADDRESS = (PROTOCOL = TCP)(HOST = <primary IP address>)(PORT = {1521|<port#>}))
 )
 (CONNECT_DATA =
   (SERVER = DEDICATED)
   (SERVICE_NAME = <primary db_unique_name>[.<primary domain name]>])
   (UR=A)
 ))
```

Set TCP socket size

Check the TCP socket sizes for the on premises system as well as the cloud instance with the following command run as root. (TCP socket sizes for the OCI DBaaS are 128MB)

ON ON-PREMISE HOST

```
# /sbin/sysctl -a | egrep net.core.[w,r]mem_max
net.core.wmem_max = 2097152
net.core.rmem_max = 4194304
# /sbin/sysctl -a | egrep net.core.[w,r]mem_max
net.core.wmem_max = 1048576
net.core.rmem_max = 4194304
```

If necessary, adjust all socket size maximums to 128MB or 134217728. For on premises systems consult your operating system guide for details about how to accomplish this. For the cloud instance edit the /etc/sysctl.conf file settings for net.core.wmem_max and net.core.rmem_max. If the values between on premises and OPC do not match, the network protocol will negotiate the lower of the two values. Therefore, the values between sites is not required to match though that is recommended in order to attain optimal transport performance.

```
net.core.rmem_max = 134217728 net.core.wmem_max = 134217728
```

Step 7: Instantiate the Standby Database

The standby database can be created from the active primary database or from a backup of the primary database. This section describes the method duplicating from the active primary database using Oracle 12.1 or higher feature RMAN ‘RESTORE…FROM SERVICE’.

Backups can also be used to instantiate databases and may be more efficient depending on the size of the database and transfer rate between the systems.

RDBMS 11.2 does not support RMAN RESTORE FROM SERVICE. Backups based duplication or RMAN DUPLICATE must be used

For details see the documentation.

For additional details regarding standby instantiation see MOS 2275154.1 Creating a Physical Standby Database in an 11.2, 12.1, 12.2 or later environment

Start the standby instance (one instance for RAC)

```
$ srvctl stop database -d <standby DB_UNIQUE_NAME> -o immediate
$ rman target /
RMAN> startup nomount

RMAN> restore standby controlfile from service 'primary';

RMAN> alter database mount;

RMAN> restore database from service 'primary' section size 5G;

RMAN> shutdown immediate

Restart the standby database

$ srvctl start database -d <standby DB_UNIQUE_NAME> -o mount
```

Clear all online and standby redo logs

```
$ sqlplus “/ as sysdba”
SQL> alter system set db_create_online_log_dest_1=<DATA Disk group>;
SQL> set pagesize 0 feedback off linesize 120 trimspool on
SQL> spool /tmp/clearlogs.sql
SQL> select distinct 'alter database clear logfile group '||group#||';' from
v$logfile;
SQL> spool off
SQL> @/tmp/clearlogs.sql

SQL> select member from v$logfile;
```

NOTE: All redo logs should be on the DATA disk group in the standby DB_UNIQUE_NAME directory

Step 8: Add RAC instances to tnsnames.ora (RAC only)

Since the scan name cannot be resolved in either direction in a hybrid configuration, and address list must be configured to provide high availability.

NOTE: List the addresses in the address list in a different order on each node to balance redo transport across nodes

TNSNAMES.ORA ON ON-PREMISE HOST

```
<standby db_unique_name> = (DESCRIPTION =
  (DESCRIPTION=
    (ADDRESS_LIST=
     (FAILOVER=on)
     (CONNECT_TIMEOUT=3)(RETRY_COUNT=3)
     (ADDRESS = (PROTOCOL = TCP)(HOST = <standby node1 IP address>)(PORT = {1521|<port#>}))
  )
     (ADDRESS = (PROTOCOL = TCP)(HOST = <standby node2 IP address>)(PORT = {1521|<port#>}))
  )
    )
    (CONNECT_DATA=
      (SERVER=DEDICATED)
      (SERVICE_NAME= standby db_unique_name>[.<standby domain name>])
    )
  )
```

TNSNAMES.ORA ON OCI DBAAS HOST

```
<primary db_unique_name> =
  (DESCRIPTION=
   (ADDRESS_LIST=
    (FAILOVER=on)
    (CONNECT_TIMEOUT=3)(RETRY_COUNT=3)
    (ADDRESS = (PROTOCOL = TCP)(HOST = <primary node1 IP address>)(PORT =
{1521|<port#>}))
  )
    (ADDRESS = (PROTOCOL = TCP)(HOST = <primary node2 IP address>)(PORT =
{1521|<port#>}))
  )
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = <primary db_unique_name>[.<primary domain name>])
  ))
```

Step 9: Configure Data Guard broker.

Enable the dg_broker_config_file parameter on primary and standby database.

NOTE: For ASM, place broker config files on separate disk groups. For RAC, broker config files must be on shared storage.

ON ON-PREMISE AND OCI DBAAS HOST

```
SQL> alter system set dg_broker_config_file1=<broker_config_file location>;
SQL> alter system set dg_broker_config_file2=<broker_config_file location>;
```

Start the Data Guard Broker Process on primary and standby database

ON ON-PREMISE AND OCI DBAAS HOST

```
SQL> alter system set dg_broker_start=true;
SQL> show parameter dg_broker_start
```

```
NAME                     TYPE    VALUE
------------------------------------ ----------- ------------------------------
dg_broker_start              boolean     TRUE

SQL> select pname from v$process where pname like 'DMON%';

PNAME
-----
DMON
```

Register the database via DGMGRL on primary site

ON ON-PREMISE HOST

```
$ dgmgrl sys/<sys password>@<net service name for primary database>
DGMGRL> CREATE CONFIGURATION <configuration_name> AS PRIMARY DATABASE IS <primary
database_name> CONNECT IDENTIFIER IS <Net Service name for primary database>;
DGMGRL> ADD DATABASE <standby database_name> AS CONNECT IDENTIFIER IS <Net Service
name for standby database> MAINTAINED AS PHYSICAL;
DGMGRL> enable configuration;
```

Configuring Client Failover

Automating client failover, the process by which clients are reconnected to the active primary database after a failure, includes relocating database services to the new primary database as part of a Data Guard failover, notifying clients that a failure has occurred in order to break them out of TCP timeout, and redirecting clients to the new primary database. Configuration details are thoroughly covered in the papers MAA Best Practices for Client Failover for Oracle Database 11g and for Oracle Database 12c. Customers using Oracle Database 12c and higher may also choose to configure application continuity for the most comprehensive continuous availability of applications. Please consult these papers and configure your environment appropriately.

Health check and Monitoring

After the standby is instantiated, a health check should be performed to ensure the Data Guard databases (primary and standby) are compliant with Oracle MAA best practices. It is also advisable to perform the health check on a monthly basis as well as before and after database maintenance. There are several methods for checking the health of a Data Guard configuration:

Oracle MAA Scorecard

Oracle provides several automated health check tools that can be downloaded from My Oracle Support specific for the type of hardware platform:

- ORAchk applicable to generic platform (suitable for Database Cloud Service)
- exachk applicable to Exadata Database Machine (suitable for Exadata Cloud Service)

Each of the automated checks include an Oracle MAA Scorecard that reports on a number of key Data Guard configuration best practices in addition to many other checks.

Oracle strongly recommends the use of these automated tools for comprehensive health check of not only the Data Guard configuration but the system as a whole. The health checks are regularly updated with current information. Be sure to download the latest version of the health checks applicable to your platform.

Monitoring

Regular monitoring of the Data Guard configuration is not provided in a Hybrid Data Guard Configuration and must be done manually. Refer to MOS note Monitoring a Data Guard Configuration (Doc ID 2064281.1) for MAA best practice recommendations for monitoring.

Validate DR Readiness

Best practice is to use Active Data Guard to offload read-only workload to the standby database to provide continuous, application-level validation that the standby is ready for production. This provides a level of assurance in addition to continuous Oracle block-level validation performed by Data Guard apply processes. It is also a best practice to periodically place the standby in read/write mode (using Data Guard Snapshot Standby) to validate its readiness to support read-write production workloads. A snapshot standby may also be used for a final level of pre-production functional and performance testing of patches and upgrades since the DR system is sized similarly to the production system. A Snapshot Standby continues to receive redo from the primary database where it is archived for later use, thus providing data protection at all times. Recovery time (RTO), however, will be extended by the amount of time required to convert the Snapshot Standby back to the standby database if a failover is required while testing is in progress. Note that additional storage is required for the fast recovery area when a standby is in snapshot mode (to hold archived redo received from the primary production database for later use and current redo and flashback logs generated by the snapshot standby). Steps for converting a standby to a snapshot standby and back are listed in the section below. Please refer to Oracle documentation for additional details on Data Guard Snapshot Standby. Optionally, you may perform an actual switchover or failover operation to the cloud for a complete end-to-end DR test; for more details see Failover/Switchover to the Cloud.

Converting Standby Database to a Snapshot Standby

A snapshot standby is a fully updatable standby database that is created from a physical standby database. On snapshot standby databases, redo data is received but not applied until the snapshot standby database is converted back to a physical standby database.

The benefits of using a snapshot standby database include the following:

1. It provides an exact replica of a production database for development and testing purposes while maintaining data protection at all times. You can use the Oracle Real Application Testing option to capture primary database workload and then replay it for test purposes on the snapshot standby.
2. It can be easily refreshed to contain current production data by converting to a physical standby and resynchronizing.

Follow the steps below to convert a physical standby database to a snapshot standby

Convert the standby to a snapshot standby and validate

Via Data Guard broker issue the following commands

```
DGMGRL> convert database 'stby' to snapshot standby;
DGMGRL> SHOW CONFIGURATION;
Configuration - DRSolution
Protection Mode: MaxPerformance Databases:
prmy - Primary database stby - Snapshot standby database
Fast-Start Failover: DISABLED
Configuration Status: SUCCESS
```

NOTE: A snapshot standby database cannot be the target of a switchover or failover. A snapshot standby database must first be converted back into a physical standby database before performing a role transition to it.

Convert the snapshot standby back into a physical standby database

Via Data Guard broker issue the following commands

```
DGMGRL> CONVERT DATABASE 'stby' to PHYSICAL STANDBY;
```

Failover/Switchover to the Cloud

At any time, you can manually execute a Data Guard switchover (planned event) or failover (unplanned event). Customers may also choose to automate Data Guard failover by configuring Fast-Start failover. Switchover and failover reverse the roles of the databases in a Data Guard configuration – the standby in the cloud becomes primary and the original on-premises primary becomes a standby database. Refer to Oracle MAA Best Practices for additional information on Data Guard role transitions.

Switchovers are always a planned event that guarantees no data is lost. To execute a switchover, perform the following in Data Guard Broker

```
DGMGRL> validate database stby;
Database Role: Physical standby database Primary Database: pri
Ready for Switchover: Yes
Ready for Failover: Yes (Primary Running)
DGMGRL> switchover to <target standby>;
```

A failover is an unplanned event that assumes the primary database is lost. The standby database is converted to a primary database immediately; after all available redo from the primary has been applied. After a failover the old primary database must be reinstated as a physical standby which is made simpler with flashback database and Data Guard broker enabled. To execute a failover and reinstatement execute the following in Data Guard Broker.

```
DGMGRL> failover to stby;
Performing failover NOW, please wait...
Failover succeeded, new primary is "stby"
Execute startup mount on one instance of the old primary before reinstating.
SQL> shutdown abort
SQL> startup mount
DGMGRL> reinstate database pri
Reinstating database "pri", please wait...
```

For more information on role transitions using the Data Guard Broker see the broker documentation for Oracle Database 11g or 12c.

Switch back to On-Premises

The same role transition procedure mentioned in the failover/switchover process is applied again when you are ready to move production back to the on-premises database.

Conclusion

Hybrid Data Guard using OCI DBaaS systems is an economical method to achieve Disaster Recovery readiness. Utilizing Maximum Availability Architecture best practices ensures the best solution for data protection and availability.

Appendix A: Example Output of RMAN Duplicate:

```
RMAN> duplicate target database for standby from active database ;

Starting Duplicate Db at 11-JAN-18
using target database control file instead of recovery catalog
allocated channel: ORA_AUX_DISK_1
channel ORA_AUX_DISK_1: SID=771 device type=DISK

contents of Memory Script:
{
   backup as copy reuse
   targetfile  '+DATAC1/raccdb/PASSWORD/passwd' auxiliary format
 '/u01/app/oracle/product/12.1.0.2/dbhome_1/dbs/orapwraccdb'   ;
}
executing Memory Script

Starting backup at 11-JAN-18
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=785 instance=raccdb1 device type=DISK
Finished backup at 11-JAN-18

contents of Memory Script:
{
   sql clone "alter system set  control_files =
  ''+RECO/RACCDB_PHX34Z/CONTROLFILE/current.263.965119865'' comment=
 ''Set by RMAN'' scope=spfile";
   backup as copy current controlfile for standby auxiliary
format  '+RECO/RACCDB_PHX34Z/CONTROLFILE/current.264.965119865';
   sql clone "alter system set  control_files =
  ''+RECO/RACCDB_PHX34Z/CONTROLFILE/current.264.965119865'' comment=
 ''Set by RMAN'' scope=spfile";
   shutdown clone immediate;
   startup clone nomount;
}
executing Memory Script

sql statement: alter system set  control_files
=   ''+RECO/RACCDB_PHX34Z/CONTROLFILE/current.263.965119865'' comment= ''Set by
RMAN'' scope=spfile

Starting backup at 11-JAN-18
using channel ORA_DISK_1
channel ORA_DISK_1: starting datafile copy
copying standby control file
```

```
output file name=+RECO/RACCDB_PHX34Z/CONTROLFILE/current.264.965119865
tag=TAG20180111T085105
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:03
Finished backup at 11-JAN-18

sql statement: alter system set  control_files
=   ''+RECO/RACCDB_PHX34Z/CONTROLFILE/current.264.965119865'' comment= ''Set by
RMAN'' scope=spfile

Oracle instance shut down

connected to auxiliary database (not started)
Oracle instance started

Total System Global Area   15032385536 bytes

Fixed Size                     3728304 bytes
Variable Size               2348813392 bytes
Database Buffers           12616466432 bytes
Redo Buffers                  63377408 bytes

contents of Memory Script:
{
   sql clone 'alter database mount standby database';
}
executing Memory Script

sql statement: alter database mount standby database

contents of Memory Script:
{
   set newname for clone tempfile  1 to new;
   set newname for clone tempfile  2 to new;
   set newname for clone tempfile  4 to new;
   switch clone tempfile all;
   set newname for clone datafile  1 to new;
   set newname for clone datafile  2 to new;
   set newname for clone datafile  3 to new;
   set newname for clone datafile  4 to new;
   set newname for clone datafile  5 to new;
   set newname for clone datafile  6 to new;
   set newname for clone datafile  7 to new;
   set newname for clone datafile  8 to new;
   set newname for clone datafile  9 to new;
   set newname for clone datafile  10 to new;
   backup as copy reuse
   datafile  1 auxiliary format new
   datafile  2 auxiliary format new
   datafile  3 auxiliary format new
   datafile  4 auxiliary format new
   datafile  5 auxiliary format new
   datafile  6 auxiliary format new
   datafile  7 auxiliary format new
   datafile  8 auxiliary format new
   datafile  9 auxiliary format new
   datafile  10 auxiliary format new
   ;
   sql 'alter system archive log current';
}
executing Memory Script
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME

renamed tempfile 1 to +DATA in control file
renamed tempfile 2 to +DATA in control file
renamed tempfile 4 to +DATA in control file

executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME

Starting backup at 11-JAN-18
using channel ORA_DISK_1
channel ORA_DISK_1: starting datafile copy
input datafile file number=00009 name=+DATAC1/raccdb/tokyo/tokyo_sysaux.dbf
output file
name=+DATA/RACCDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/sysaux.266.9651199
11 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:45
channel ORA_DISK_1: starting datafile copy
input datafile file number=00001 name=+DATAC1/raccdb/DATAFILE/system.dbf
output file name=+DATA/RACCDB_PHX34Z/DATAFILE/system.265.965119957
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:25
channel ORA_DISK_1: starting datafile copy
input datafile file number=00003 name=+DATAC1/raccdb/DATAFILE/sysaux.dbf
output file name=+DATA/RACCDB_PHX34Z/DATAFILE/sysaux.272.965119983
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:25
channel ORA_DISK_1: starting datafile copy
input datafile file number=00005 name=+DATAC1/raccdb/DATAFILE/undotbs1.dbf
output file name=+DATA/RACCDB_PHX34Z/DATAFILE/undotbs1.263.965120007
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:25
channel ORA_DISK_1: starting datafile copy
input datafile file number=00006 name=+DATAC1/raccdb/DATAFILE/undotbs2.dbf
output file name=+DATA/RACCDB_PHX34Z/DATAFILE/undotbs2.262.965120033
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:25
channel ORA_DISK_1: starting datafile copy
input datafile file number=00007 name=+DATAC1/raccdb/DATAFILE/users.dbf
output file name=+DATA/RACCDB_PHX34Z/DATAFILE/users.261.965120057
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:15
channel ORA_DISK_1: starting datafile copy
input datafile file number=00002 name=+DATAC1/raccdb/PDBSEED/DATAFILE/system.dbf
output file
name=+DATA/RACCDB_PHX34Z/4D8C4492AD829E0DE0530C4F800ADE8D/DATAFILE/system.274.9651200
73 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:07
channel ORA_DISK_1: starting datafile copy
input datafile file number=00004 name=+DATAC1/raccdb/PDBSEED/DATAFILE/sysaux.dbf
output file
name=+DATA/RACCDB_PHX34Z/4D8C4492AD829E0DE0530C4F800ADE8D/DATAFILE/sysaux.275.9651200
79 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:07
channel ORA_DISK_1: starting datafile copy
input datafile file number=00008 name=+DATAC1/raccdb/tokyo/tokyo_system.dbf
output file
name=+DATA/RACCDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/system.276.9651200
87 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:07
channel ORA_DISK_1: starting datafile copy
input datafile file number=00010 name=+DATAC1/raccdb/tokyo/tokyo_users01.dbf
output file
name=+DATA/RACCDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/users.277.96512009
3 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:01
Finished backup at 11-JAN-18
```

```
sql statement: alter system archive log current

contents of Memory Script:
{
   switch clone datafile all;
}
executing Memory Script

datafile 1 switched to datafile copy
input datafile copy RECID=1 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/DATAFILE/system.265.965119957
datafile 2 switched to datafile copy
input datafile copy RECID=2 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/4D8C4492AD829E0DE0530C4F800ADE8D/DATAFILE/system.274.9651200
73
datafile 3 switched to datafile copy
input datafile copy RECID=3 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/DATAFILE/sysaux.272.965119983
datafile 4 switched to datafile copy
input datafile copy RECID=4 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/4D8C4492AD829E0DE0530C4F800ADE8D/DATAFILE/sysaux.275.9651200
79
datafile 5 switched to datafile copy
input datafile copy RECID=5 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/DATAFILE/undotbs1.263.965120007
datafile 6 switched to datafile copy
input datafile copy RECID=6 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/DATAFILE/undotbs2.262.965120033
datafile 7 switched to datafile copy
input datafile copy RECID=7 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/DATAFILE/users.261.965120057
datafile 8 switched to datafile copy
input datafile copy RECID=8 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/system.276.9651200
87
datafile 9 switched to datafile copy
input datafile copy RECID=9 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/sysaux.266.9651199
11
datafile 10 switched to datafile copy
input datafile copy RECID=10 STAMP=965120096 file
name=+DATA/RACCDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/users.277.96512009
3
```

Appendix B: MAA Best Practice Parameter Settings

The following settings are recommended to follow MAA best practices in order to provide maximum availability and protection of the data. These parameters should be set on both the primary and standby databases.

- ARCHIVELOG enabled
- Flashback database on
- FORCE LOGGING enabled
- Use SPFILE
- Use Data Guard Broker
- COMPATIBLE uses 4 decimals and is the same on both databases
- DB_FILES=1024
- Online Redo Log characteristics
  - At least 1G in size for non-Exadata databases, 4G for Exadata
  - Only multiplexed on normal redundancy storage; single member groups when using high redundancy storage
  - Minimum of 3 online log groups per thread
  - Reside on DATA disk group
- Standby Redo Log characteristics
  - Identical size as online redo logs
  - For RAC, assign SRL groups to a thread
  - Single member only
  - Same number of groups per thread as online redo log groups
  - Reside on DATA disk group
- LOG_BUFFER = 128M for 11.2; 256M for 12.1+
- DB_BLOCK_CHECKING=MEDIUM or FULL *Note: this setting could affect performance and should be enabled only after proper testing of the application.*
- DB_BLOCK_CHECKSUM=TYPICAL
- STANDBY_FILE_MANAGEMENT=AUTO
- DB_LOST_WRITE_PROTECT=TYPICAL
- DB_FLASHBACK_RETENTION_TARGET=minimum of 120
- FAST_START_MTTR_TARGET=300

- USE_LARGE_PAGES=ONLY if hugepages are configured and properly sized on the on-premises system
- CLUSTER_INTERCONNECTS set per gv$cluster_interconnects ***# This needs only be set on Exadata***
- PARALLEL_THREADS_PER_CPU=1
- DB_CREATE_ONLINE_LOG_DEST_1= DATA disk group
- DB_CREATE_ONLINE_LOG_DEST_n other than 1 should only be set when DATA is not high redundancy
- DB_CREATE_FILE_DEST uses DATA disk group
- DB_RECOVERY_FILE_DEST uses RECO disk group
- Recyclebin is on

Appendix C: Example output for RMAN RESTORE FROM SERVICE

```
[oracle@host1 ~]$ rman target /

Recovery Manager: Release 18.0.0.0.0 - Production on Sat Mar 2 16:48:14 2019
Version 18.3.0.0.0

Copyright (c) 1982, 2018, Oracle and/or its affiliates.  All rights reserved.

connected to target database (not started)

RMAN> startup nomount

Oracle instance started

Total System Global Area   32212253688 bytes

Fixed Size                    12454904 bytes
Variable Size               4160749568 bytes
Database Buffers           27984396288 bytes
Redo Buffers                  54652928 bytes

RMAN> CONFIGURE DEVICE TYPE DISK PARALLELISM 4;

new RMAN configuration parameters:
CONFIGURE DEVICE TYPE DISK PARALLELISM 4 BACKUP TYPE TO BACKUPSET;
new RMAN configuration parameters are successfully stored

RMAN> restore standby controlfile from service 'victorp';

Starting restore at 02-MAR-19
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=10 instance=victor1 device type=DISK

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: using network backup set from service victorp
channel ORA_DISK_1: restoring control file
channel ORA_DISK_1: restore complete, elapsed time: 00:00:20
output file name=+RECO/VICTOR_PHX2W5/CONTROLFILE/current.256.1001775751
Finished restore at 02-MAR-19

RMAN> alter database mount;

released channel: ORA_DISK_1
Statement processed
```

```
RMAN> restore database from service 'victorp' section size 2G;

Starting restore at 02-MAR-19
Starting implicit crosscheck backup at 02-MAR-19
allocated channel: ORA_DISK_1
allocated channel: ORA_DISK_2
allocated channel: ORA_DISK_3
allocated channel: ORA_DISK_4
Crosschecked 1 objects
Finished implicit crosscheck backup at 02-MAR-19

Starting implicit crosscheck copy at 02-MAR-19
using channel ORA_DISK_1
using channel ORA_DISK_2
using channel ORA_DISK_3
using channel ORA_DISK_4
Finished implicit crosscheck copy at 02-MAR-19

searching for all files in the recovery area
cataloging files...
cataloging done

List of Cataloged Files
=======================
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_1.261.1001776239
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_2.262.1001776491
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_1_seq_1.263.1001776653
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_3.264.1001776653
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_1_seq_2.265.1001776679
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_4.266.1001776721
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_5.267.1001776723
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_1_seq_3.270.1001791759

using channel ORA_DISK_1
using channel ORA_DISK_2
using channel ORA_DISK_3
using channel ORA_DISK_4

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: using network backup set from service victorp
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
channel ORA_DISK_1: restoring datafile 00001 to
+DATA/VICTOR_PHX2W5/DATAFILE/system.263.1001869475
channel ORA_DISK_1: restoring section 1 of 1
channel ORA_DISK_2: starting datafile backup set restore
channel ORA_DISK_2: using network backup set from service victorp
channel ORA_DISK_2: specifying datafile(s) to restore from backup set
channel ORA_DISK_2: restoring datafile 00003 to
+DATA/VICTOR_PHX2W5/DATAFILE/sysaux.274.1001869485
channel ORA_DISK_2: restoring section 1 of 1
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00004 to
+DATA/VICTOR_PHX2W5/DATAFILE/undotbs1.270.1001869497
channel ORA_DISK_3: restoring section 1 of 1
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00005 to
+DATA/VICTOR_PHX2W5/DATAFILE/system.272.1001869507
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_3: restore complete, elapsed time: 00:00:25
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00006 to
+DATA/VICTOR_PHX2W5/DATAFILE/sysaux.271.1001869523
channel ORA_DISK_3: restoring section 1 of 1
channel ORA_DISK_4: restore complete, elapsed time: 00:05:02
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00007 to
+DATA/VICTOR_PHX2W5/DATAFILE/users.268.1001869811
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_4: restore complete, elapsed time: 00:00:19
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00008 to
+DATA/VICTOR_PHX2W5/DATAFILE/undotbs1.266.1001869831
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_4: restore complete, elapsed time: 00:01:07
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00009 to
+DATA/VICTOR_PHX2W5/DATAFILE/undotbs2.264.1001869899
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_3: restore complete, elapsed time: 00:06:41
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00010 to
+DATA/VICTOR_PHX2W5/DATAFILE/system.265.1001869925
channel ORA_DISK_3: restoring section 1 of 1
channel ORA_DISK_4: restore complete, elapsed time: 00:00:39
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00011 to
+DATA/VICTOR_PHX2W5/DATAFILE/sysaux.262.1001869939
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_3: restore complete, elapsed time: 00:05:10
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00012 to
+DATA/VICTOR_PHX2W5/DATAFILE/undotbs1.261.1001870237
channel ORA_DISK_3: restoring section 1 of 1
channel ORA_DISK_2: restore complete, elapsed time: 00:12:44
channel ORA_DISK_2: starting datafile backup set restore
channel ORA_DISK_2: using network backup set from service victorp
channel ORA_DISK_2: specifying datafile(s) to restore from backup set
channel ORA_DISK_2: restoring datafile 00013 to
+DATA/VICTOR_PHX2W5/DATAFILE/undo_2.260.1001870251
channel ORA_DISK_2: restoring section 1 of 1
channel ORA_DISK_3: restore complete, elapsed time: 00:00:27
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00014 to
+DATA/VICTOR_PHX2W5/DATAFILE/users.280.1001870265
channel ORA_DISK_3: restoring section 1 of 1
channel ORA_DISK_2: restore complete, elapsed time: 00:00:23
channel ORA_DISK_3: restore complete, elapsed time: 00:00:20
channel ORA_DISK_4: restore complete, elapsed time: 00:06:25
channel ORA_DISK_1: restore complete, elapsed time: 00:14:13
Finished restore at 02-MAR-19
```

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

- blogs.oracle.com/oracle
- facebook.com/oracle
- twitter.com/oracle
- oracle.com

Integrated Cloud Applications & Platform Services

White Paper Title
April 2019
Author: Kazuhiro Ikeda, Sebastian Solbach

Contributing Authors: Ramachandran Pandrapattahil, Andy Steinorth, Pieter Van Puymbroeck, Lawrence To

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