Backup and Recovery Best Practices for the Oracle Database Appliance

Protecting databases and applications running on Oracle Database Appliance

February 3, 2021 | Version 0.90

PURPOSE STATEMENT

This document provides an overview of the best practices for defining optimal backup and recovery strategies to protect mission critical data and file systems in Oracle Database Appliance environments. The Oracle Database provides sophisticated and scalable backup technologies. These technologies work transparently on the Oracle Database Appliance.

INTENDED AUDIENCE

Who manages Oracle Database Appliance environments

DISCLAIMER

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle software license and service agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

TABLE OF CONTENTS

Purpose Statement 1

Intended Audience 1

Disclaimer 1

INTRODUCTION 3

Backup and Recovery Procedures for Oracle Database Appliance 3

Oracle Database Appliance Platform Configuration for Backup and Recovery 3

Restore and Recovery in Oracle Database Appliance (Bare Metal configuration) 5
- Recovery from single server node failure on Oracle Database Appliance Bare Metal configuration 5
- Restore after simultaneous failure of both server nodes in Oracle Database Appliance Bare Metal configuration 6

Backup and recovery in Oracle Database Appliance Virtualized Platform configuration 6
- Backup and Restore of ODA_BASE Domain 6
  - Backup of ODA_BASE Domain 6
  - Restore of ODA_BASE Domain 6
- Recovery from single server node failure on Oracle Database Appliance Virtualized Platform 7
- Restore after simultaneous failure of both server nodes in Oracle Database Appliance Virtualized Platform 7
- Backup and Restore of Guest Virtual Machines (VMs) 7

Backup and recovery using ODA Tooling 7

Backup and recovery in Oracle Cloud 15

Backup and recovery in Zero Data Loss Recovery Appliance 19

Backup and recovery using Tape devices 26
- Media Management Software for Tape Backups 26

Disk backups 27
- RMAN Backups to Local Disks 28
- RMAN Backups to External Storage 28
- Performance Observations 28

Backup and recovery with Network File System (NFS) storage 29
- RMAN backups to the Oracle ZFS Storage Appliance 7120 29
- Performance Numbers for ZFSSA-Based Backup Configurations 31
- Engineered Systems Backup Utility 31

Backup and Recovery for Oracle Database Appliance S|M|L 32

Database Backup and Recovery Best Practices 32

Conclusion 33

Appendix A Configuring load-balanced backups 33

Appendix B: Sample Scripts 34

Appendix C Sample commands for backup and recovery in cloud 37

Appendix D Sample commands for backup and recovery in ZDLRA 42

APPENDIX E Setting Up the Transparent Data Encryption Wallet (TDE) 46

References 48

INTRODUCTION

The Oracle Database Appliance is an Oracle Engineered System consisting of hardware and software that saves customers time and money by simplifying deployment, maintenance, and support of high availability database solutions. It is built using the world's most popular database - Oracle Database. Along with Oracle Real Applications Clusters (Oracle RAC), the Database Appliance offers customers a fully integrated system of software, servers, storage and networking that delivers high availability database services for a wide range of custom and packaged OLTP and Data Warehousing workloads.

The Oracle Database Appliance offers customers capacity-on-demand database software licensing, allowing seamless scalability from 2 to 64 on X8-2 HA processor cores without any hardware upgrades. The appliance also offers the option of deploying a virtualized platform based on Oracle VM. Support for virtualization allows customers and ISVs to build a solution-in-a-box that efficiently utilizes resources and extends capacity-on-demand licensing to both database and application workloads by leveraging Oracle VM hard partitioning.

With built-in redundancy Oracle Database Appliance offers high protection against hardware failures. However, you must backup your databases, operating system, application software, and any other artifacts to ensure recoverability from data loss and corruption type of scenario. This document discusses the different options and considerations for backup and recovery operations in an Oracle Database Appliance environment.

The information in this document applies to Oracle Database Appliance Software versions 2.x to 19.x.

BACKUP AND RECOVERY PROCEDURES FOR ORACLE DATABASE APPLIANCE

This document provides details of core backup and recovery topics and procedures for the Oracle Database Appliance platform include the following:

- Restore and recovery in Oracle Database Appliance Bare Metal configuration
- Backup and recovery in Oracle Database Appliance Virtualized Platform configuration
- Backup and Recovery in Oracle Cloud
- Backup and recovery with Oracle ZFS Storage Appliance (ZFSSA)
- Backup and recovery with Zero Data Loss Recovery Appliance (ZDLRA)
- Backup and recovery using Tape devices
- Backup and recovery with Network File System (NFS) storage

Later sections outlines each of the above procedures that you can test, validate, and use in your Oracle Database Appliance environments. Note that the procedures for Oracle Database Appliance Virtualized Platform only apply when you are using such configurations.

ORACLE DATABASE APPLIANCE PLATFORM CONFIGURATION FOR BACKUP AND RECOVERY

Backup and recovery procedures and processes are one of the key operational aspects of Oracle Database Appliance. In order to protect against data loss and corruption, you must ensure that file systems and databases running on your Oracle Database Appliance system are backed up regularly. These backups must be validated and consistent so that the databases and file systems can be restored and recovered when needed.

Oracle Database Appliance includes high bandwidth bonded 1GbE/ 10GbE / 25GbE (SFP28)) port interfaces. For backup traffic either the regular public network interface or a dedicated backup network interface can be used

The layout and available storage capacity of the Oracle Database Appliance disk groups depends on your selection of the “Backup Type” option and your choice of mirroring options in the Oracle Database Appliance Manager. During the deployment process, the Configurator allows you to select between triple mirroring (high redundancy) and double mirroring (normal redundancy) of available storage. The Configurator also allows you a choice to place the backups either on the Oracle Database Appliance local storage or on external storage. When you choose Local Backup Type option, 40% of the disk is assigned to the DATA area, and 60% of the disk is assigned to the Fast Recovery Area (RECO) area. For the External Backup Type option, 80% of the disk is assigned to the DATA area, and 20% of the disk is assigned to the Fast Recovery Area (RECO) area.

Oracle Database Appliance X8-2 allows you to choose DATA disk group capacity between 10% and 90% of total shared disk storage and remainder is reserved for RECO. For example, if you enter 80, then 80% of the storage for DATA and 20% for RECO. Note that Oracle Database Appliance storage capacity varies for different hardware models.

The following core technologies are the key enablers of efficient backup and recovery operations on the Oracle Database Appliance platform.

Oracle Recovery Manager (RMAN)

Oracle Recovery Manager (RMAN) provides the native backup and recovery infrastructure within Oracle Database, enabling optimized data protection in the Oracle Database Appliance environments. Backup, restore, and recovery operations are performed using standard RMAN commands. RMAN can parallelize backup operations across both Real Application Cluster (RAC) nodes. This allows all disks, all network connections and all CPUs in the system to contribute towards performing backup operations.

RMAN block change tracking allows incremental backups to run very quickly and efficiently. With block change tracking, only the areas of the database that have been modified since the last incremental backup, or full backup, are read from disk.

RMAN stores data in one of two formats – Image Copy or Backup Set. An Image Copy is an exact copy of a single data file, archived redo log file, or control file. Image copies are not stored in an RMAN-specific format. They are identical to the results of copying a file with operating system commands. RMAN can use Image Copies during RMAN restore and recovery operations, and it can use Image Copies with non-RMAN restore and recovery techniques

A backup set contains the data from one or more data files, archived redo log files, control files or server parameter file. The smallest unit of a backup set is a binary file called a backup piece. Backup sets are the only form in which RMAN can write backups to sequential devices such as tape drives. For more information, refer Oracle Recovery Manager (RMAN) documentation.

Following are various supported backup storage locations for the database backup

Oracle Secure Backup (OSB)

Oracle Secure Backup (OSB) is a centralized tape backup management solution for the entire IT environment including file systems and Oracle Databases. With built-in RMAN integration, Oracle Secure Backup delivers the fastest Oracle Database backups to tape. Some important backup optimizations such as the following that provide substantial savings in backup time and tape costs are available only with Oracle Secure Backup and RMAN:

- Unused block compression eliminates the time and space needed to backup blocks that are allocated to tablespaces but are not currently used by tables.
- Undo optimization eliminates the time and space usage needed to back up undo data that is not required to recover using the current backup

Oracle ZFS Storage Appliance (ZFSSA)

Oracle ZFS Storage Appliance can be used as a backup storage location for databases. The high-speed networks support good performance. With the ZFS HA solutions, customers do not have to worry about single point of network failures

Oracle Database Backup Cloud Service

Oracle Database Backup Cloud Service is a secure, scalable, on-demand storage solution for backing up Oracle databases to Oracle Cloud. The service complements your existing backup strategy by providing an off-site storage location in the public cloud. Storage management and data transfer complexities are handled by the service, not by database administrators. Database Administrators use the familiar RMAN interface to perform backup and restore operations and there is no need to learn new tools or commands

Oracle ASM Cloud File System

Oracle Database Appliance provides the ability to create a high availability (HA) clustered file system. It is highly recommended that you store all scripts and configuration files in the shared ACFS file system (identified by default as the **/cloudfs** mount point on Oracle Database Appliance (Virtualized platform) and for Oracle Database Appliance (Bare Metal platform), create dedicated ACFS file system) and take backup of this file system on a regular basis to external storage.

Zero Data Loss Recovery Appliance (ZDLRA)

Oracle’s Zero Data Loss Recovery Appliance is a cloud-scale engineered system designed to dramatically eliminate data loss and reduce data protection overhead for all Oracle databases in the enterprise. Integrated with Recovery Manager (RMAN), it enables a centralized, incremental forever backup strategy for hundreds to thousands of databases in the enterprise, using cloud-scale, fully fault-tolerant hardware and storage. The appliance provides databases with sub-second recovery point objectives and continuously validates backups for assured recoverability of Oracle data. Oracle Enterprise Manager enables "single pane of glass" control of all administrative operations on the appliance, providing complete, end-to-end visibility of the Oracle backup lifecycle.

RESTORE AND RECOVERY IN ORACLE DATABASE APPLIANCE (BARE METAL CONFIGURATION)

This section describes the procedure to restore a failed server node in an Oracle Database Appliance Bare Metal configuration. This procedure may apply to situations such as the following:

- A server node crashed and you are unable to bring-up the server node and Clusterware
- Server OS file system is corrupted and cannot be repaired
- Failure of both local disks on a server node
- You are unable to bring up Clusterware and restore is the only option, etc.

This section covers the following two scenarios

- Restore of a single server node in Bare Metal configuration
- Restore after simultaneous failure of both server nodes in Bare Metal configuration

RECOVERY FROM SINGLE SERVER NODE FAILURE ON ORACLE DATABASE APPLIANCE BARE METAL CONFIGURATION

You should only need to execute this procedure if you cannot recover a Server node/Clusterware using any standard recovery procedures.

To restore a single server in Oracle Database Appliance High Availability Model Bare Metal platform configuration, refer to MOS note titled Oracle Database Appliance: Script to perform Bare Metal Restore of a Single node on ODA 12.1.x (Doc ID 2328555.1). When you execute and complete this procedure successfully, the restored node should be up and running with Grid infrastructure normally operational and all RDBMS HOMEs should be added back to the restored node.

Once the failed server node is restored, if applicable, you may need to perform the following post-restoration steps

- Add the instances back to restore node

  For single instance

  - Bring-up the instance. If the instance is not coming up and if database backup exists, then restore and recovery the instance using standard RMAN procedures from the existing backups

  For RAC instance

  - Delete the database instance corresponding to the restored server node using DBCA. This is to ensure that all the references to the inactive instance are deleted before the instance is adding back to the configuration
    - Invoke DBCA in GUI mode from surviving node
    - Select (Instance Management and delete an instance). Select the instance to delete from the list of instances of cluster database. Then delete the instance
  - Add the instance using DBCA from the surviving node back to restore node
    - Invoke DBCA in GUI mode from surviving node
    - Select Instance Management, Add an instance, continue to the next screens and add the instance

  Note: Depending on your configured RMAN backup destination, refer RMAN best practices section of this document to setup backup procedures appropriately.

- Complete any miscellaneous/missing steps on the restored server node
  - Run the commands “odacli describe-component and $GRID_HOME/OPatch/opatch lsinventory” and compare the patch level between the surviving server node and the restored server node. If there are any patch level mismatches, you should apply

the missing patches on the restored server node. Note that on Oracle Database Appliance, patches are applied in specific order (i.e., SERVER àSTORAGE àDB (database).

- If any cron jobs were previously configured on the restored server node, you may want to add those back at this stage.

RESTORE AFTER SIMULTANEOUS FAILURE OF BOTH SERVER NODES IN ORACLE DATABASE APPLIANCE BARE METAL CONFIGURATION

In case of a simultaneous unrecoverable failure of both server nodes, the two nodes can be re-imaged and re-deployed. The databases can then be recovered from backups. This is akin to a fresh deployment of the platform and a restore of the backups on to the newly re-deployed platform. Refer Release Notes "https://docs.oracle.com/en/engineered-systems/oracle-database-appliance/index.html" for details and links to ISO images and GI/RDBMS software bundles that are used for deployments.

BACKUP AND RECOVERY IN ORACLE DATABASE APPLIANCE VIRTUALIZED PLATFORM CONFIGURATION

This section describes the procedure to restore a failed server node in an Oracle Database Appliance Virtualized Platform configuration. This procedure may apply to situations such as the following:

- A server node crashed and you are unable to bring-up the server node and Clusterware
- Server OS file system is corrupted and cannot be repaired
- Failure of both local disks on a server node
- You are unable to bring up Clusterware and restore is the only option, etc
- Unexpected malfunctioning of ODA_BASE domain post any wrongful patching activities or other changes introduced inappropriately in the environment

Recovery operations may vary depending on the nature of loss of part or whole of the Oracle Database Appliance virtualized platform configuration. This section covers the following three scenarios

- Backup and Restoreof ODA_BASE domain
- Recovery from single server node failure on Oracle Database Appliance VirtualizedPlatform
- Recovery from simultaneous failure of both server nodes on Oracle Database Appliance VirtualizedPlatform

BACKUP AND RESTORE OF ODA_BASE DOMAIN

BACKUP OF ODA_BASE DOMAIN

This sub-section describes the procedure to take backup of ODA_BASE, before applying Oracle Database Appliance Patch Set Bundle, any change management activities to environment and etc. Execute the steps below to take backup of the ODA_BASE

1. Login into Domain-0 as root user
2. Stop the ODA_BASE domain
   ```
   $ oakcli stop ODA_BASE
   ```
3. You can use the "rsync" command as listed below to take a backup of the ODA_BASE domain.
   ```
   $ /usr/bin/rsync -vaz --delete --progress --exclude '<exclude files>' <Source location: ODA_BASE location> <Target location: External NFS storage to move the backup>
   ```
   For example,
   ```
   $ /usr/bin/rsync -vaz --delete --progress --exclude *zip --exclude *gz ' /OVS/Repositories/odabaseRepo /external
   ```
   The above command in effect backs up the /OVS/Repositories/odabaseRepo/VirtualMachines/oakDom1 directory to external NFS directory (e.g., "/external")
4. Start the ODA_BASE domain
   ```
   $ oakcli start ODA_BASE
   ```
5. Now, repeat this procedure (steps 1 to 4) on the ODA_BASE domain on the other server node

RESTORE OF ODA_BASE DOMAIN

This sub-section describes the process of restoring ODA_BASE domain from the backup.

Execute the steps below to restore the ODA_BASE domain from the backup (refer to previous sub-section for details).

1. Login into Domain-0 as root user
2. Stop the ODA_BASE
   ```
   $ oakcli stop ODA_BASE
   ```
3. Create directory, if it does not already exist
   ```
   $ mkdir -p /OVS/Repositories
   ```
4. Use“rsync” command (or any other restore procedure)to copy the ODA_BASE domainfrom the backup
   ```
   $ /usr/bin/rsync -vaz --delete --progress --exclude '<exclude files>' <Backup location> <Restore location>
   ```
   For example,
   ```
   $ /usr/bin/rsync -vaz --delete --progress --exclude *zip --exclude *gz/external /OVS/Repositories/
   ```
5. The above command should restore backup of directorybelow to /OVS/Repositories/ from /external directory
   /OVS/Repositories/odabaseRepo/VirtualMachines/oakDom1
6. Start the ODA_BASEdomain
   ```
   $ oakcli start ODA_BASE
   ```
7. Validate the environment using oakcli validatecommand and ensure ODA_BASE is functioning normally
   ```
   $ oakcli validate –d
   ```

RECOVERY FROM SINGLE SERVER NODE FAILURE ON ORACLE DATABASE APPLIANCE VIRTUALIZED PLATFORM

Refer to My Oracle Support note titled Oracle Database Appliance: How to perform Restoration of a Single node for ODA Virtualized Platform 12.1.x (Doc ID 2289376.1) for restoring a single server node of an Oracle Database Appliance Virtualized Platform implementation. At the end of the process, the restored node (ODA_BASE domain) along with the Grid infrastructure instance will be up and running and all RDBMS HOMEs should be added back to this restored node. If applicable, complete the post-restoration steps as referred in the sections above.

RESTORE AFTER SIMULTANEOUS FAILURE OF BOTH SERVER NODES IN ORACLE DATABASE APPLIANCE VIRTUALIZED PLATFORM

As stated in the previous section, if both server nodes of an Oracle Database Appliance system are not recoverable and assuming you have database backups, then the quickest approach to recovery may be to re-image the server nodes and restore and recover the databases using the backups.

To re-image the server nodes, refer Release Notes “https://docs.oracle.com/en/engineered-systems/oracle-database-appliance/index.html” and choose the correct ISO image to re-deploy the system asa virtualized platform. Then deploy ODA_BASE using the correct end-user bundle and restore and recover the database(s) using your standard RMAN procedures. Further, restore or redeploy any virtual machines (VMs) that you may have previously deployed on the system.

BACKUP AND RESTORE OF GUEST VIRTUAL MACHINES (VMS)

Refer to My Oracle Support note titled Howto backup/restore your VM Guest (Doc ID 1633166.1) for various backup and recovery scenarios applicable to Guest VMs deployed on Oracle Database Appliance Virtualized Platform.

BACKUP AND RECOVERY USING ODA TOOLING

Databases with and without Transparent Data Encryption (TDE) enabled, which are deployed on Bare Metal System and on Oracle Database Appliance Release 19.9 and later version, backup and recovery operations can be performed using ODA tooling (Brower User Interface and CLI). ODA tooling supports backup destination as Oracle Fast Recovery Area (FRA) disk (Internal FRA), Network File System (NFS) location (External FRA) and Oracle Cloud Infrastructure Object Storage (Oracle Object Storage). For Oracle Database Appliance Release 19.8 and below versions, refer section (BACKUP AND RECOVERY IN ORACLE CLOUD) for backup and recovery operations

The high-level procedure to setup backups and recovery options using ODA tooling as follows:

(1) Prerequisites

(2) Create a Database Backup Policy

(3) Attach a Backup Policy to Database

(4) Perform a Backups using Browser User Interface

(5) Perform a Restore and recover operations using Browser User Interface, when necessary

Following Backup and Recovery options are available with tooling

- Backup to and recovery from an Oracle Fast Recovery Area (FRA) disk (Internal FRA)
- Backup to and recovery from a Network File System (NFS) location (External FRA)
- Backup to and recovery from Oracle Cloud Infrastructure Object Storage (Oracle Object Storage)

1. Prerequisites

   Following prerequisites need to complete before creating a Backup Policy depending upon the Backup destination

1.1 Prerequisites for Backup destination (Network File System (NFS) location (External FRA))

- Create a mount point for the NFS location

  The mount point must be accessible from both nodes. The oracle user must have read/write permissions to the NFS location.

  Follow these steps to create a mount point for the NFS backup location

  - Follow these steps on the source server (storage server)
    - Create a sharable location on the source server and give full permissions to this directory

      ```
      # mkdir <shared location>
      ```

      For example,

      ```
      # mkdir /mnt/nfs_storage
      # chmod 774 /mnt/nfs_storage
      ```

    - Add entries in the /etc/exports file in the format

      <shared_location> <destination_IP> (<permissions>).

      In this file, replace <shared_location> with the directory being exported, replace <destination_IP> with the destination host IP address or network to which the export is being shared, and replace <permissions> with the options for that host or network (for example, read/write (rw) and etc)

      For example:

      ```
      # cat /etc/exports
      /mnt/nfs_storage 192.0.2.1(rw,sync)
      /mnt/nfs_storage 192.0.2.2(rw,sync)
      ```

    - Restart the NFS server

      ```
      # service nfs  restart
      ```

    - Check the export list for the entries.

      ```
      # showmount -e
      ```

      For example,

      ```
      Export list for odadbserver
      /mnt/nfs_storage 192.0.2.1,192.0.2.2
      ```

  - Follow these steps on the client machine (ODA database server)
    - Create a client location on the client machine as the `root` user.

      ```
      #mkdir <client location>
      ```

      For example,

      ```
      # mkdir /nfs/oda_backup
      ```

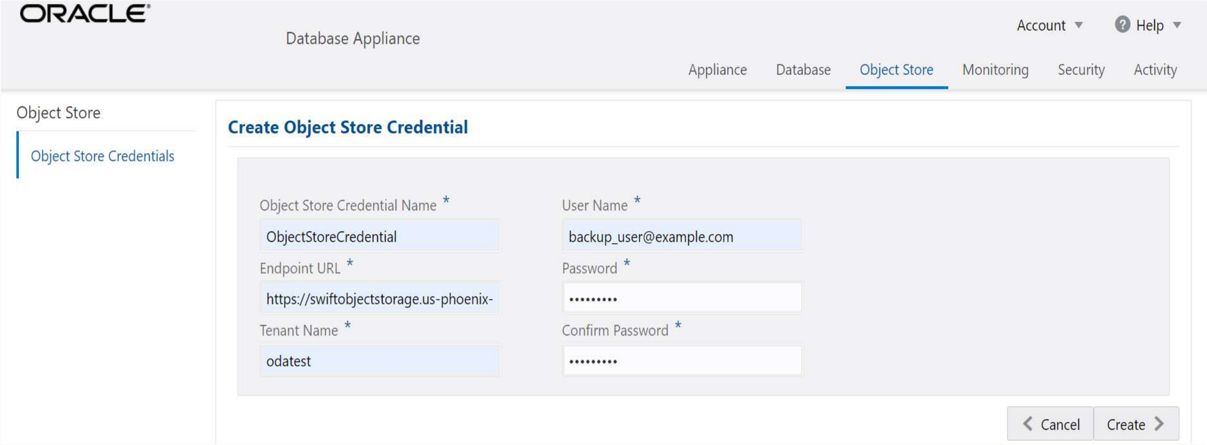
    - Mount this location with the source location in the format

      <storage_server>:<source_folder> <client_location> <mount options>

  For example, add below entries in single line to file /etc/fstab and mount it
  192.0.2.3:/mnt/nfs_storage /nfs/oda_backup nfs
  rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,vers=3,timeo=600
  # mount <client location>
  For example,
  # mount /nfs/oda_backup
  - Check if the mount details are correct
    # mount | grep <client location>
    For example,
    # mount | grep /nfs/oda_backup

1.2 Prerequisites for Backup destination (Oracle Cloud Infrastructure Object Storage (Oracle Object Storage))

- Purchase the Oracle Database Backup Cloud Service subscription
  To get started with the Oracle Database Backup Cloud Service, you need to purchase a service at cloud.oracle.com/database_backup or cloud.oracle.com/database
- Create an Object Store Object with your credentials
  - Log into the Browser User Interface using user oda-admin
    https://host name or ip-address:7093/mgmt/index.html
  - Click the **Object Store** tab in the Browser User interface
  - Click the **Create Object Store Credential** to create a new Object Store credential
    For example, after specifying values for the entries (Object Store Credential Name as ObjectStoreCredential, User Name as backup_user@example.com, Endpoint URL as https://swiftobjectstorage.us-phoenix-1.oraclecloud.com/v1, Tenant Name as odatest and Password as Object Store swift authentication token), user interface screen looks similar to below



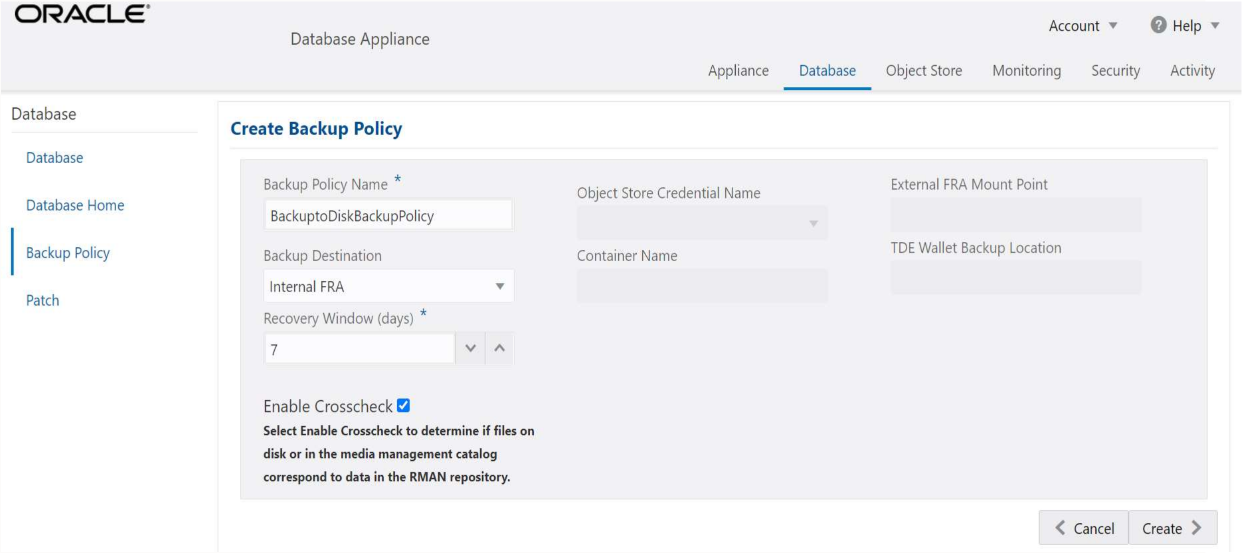
Object Store Credential Name * ObjectStoreCredential
User Name * backup_user@example.com
Endpoint URL * https://swiftobjectstorage.us-phoenix-
Password * •••••••••
Tenant Name * odatest
Confirm Password * •••••••••

  - Click **Create.** Click **Yes** to Confirm that “Are you sure you want to create an Object Store"
    A link to the job appears. When the job completes successfully, the Object Store Credential is ready.
- Create a container (Bucket)
  - Log into the Cloud Console using Tenancy User name and Password
    https://console.us-<region>-1.oraclecloud.com/object-storage/buckets
    For example
    https://console.us-phoenix-1.oraclecloud.com/object-storage/buckets
  - Click the **Create Bucket**, specify BUCKET NAME and Click **Create Bucket** to Create a Bucket
- Configuring Agent Proxy Settings for Object Store Access
  If the Object Store IP address is accessible, only through proxy setup by the Oracle Database Appliance server, then define the proxy setting for the agent, so that the agent can access the Object Store.
  To create a backup policy that uses Object Store location, the agent must be able to access the Object Store URL.
  - Login into ODA database server and switch to root user
  - Define the HttpProxyHost and HttpProxyPort settings in the update-agentconfig-parameters command
    - odacli update-agentconfig-parameters -n HttpProxyHost -v www-proxy.test.com -n HttpProxyPort -v 80 –u
      Specify values for the parameters HttpProxyHost and HttpProxyPort and execute above command
      For more information about the update-agentconfig-parameters command usage, see the Oracle Database Appliance Command-Line Interface.
  - Verify that the update succeeded
    - # odacli describe-job -i <job id from above command>

        - For example,
          # odacli describe-job -i 0b0cbf9b-b0ab-4523-a096-5da4e48fc825
    - Run the list-agentconfigParameters command to view the changes in the proxy settings:
    - # odacli list-agentConfigParameters

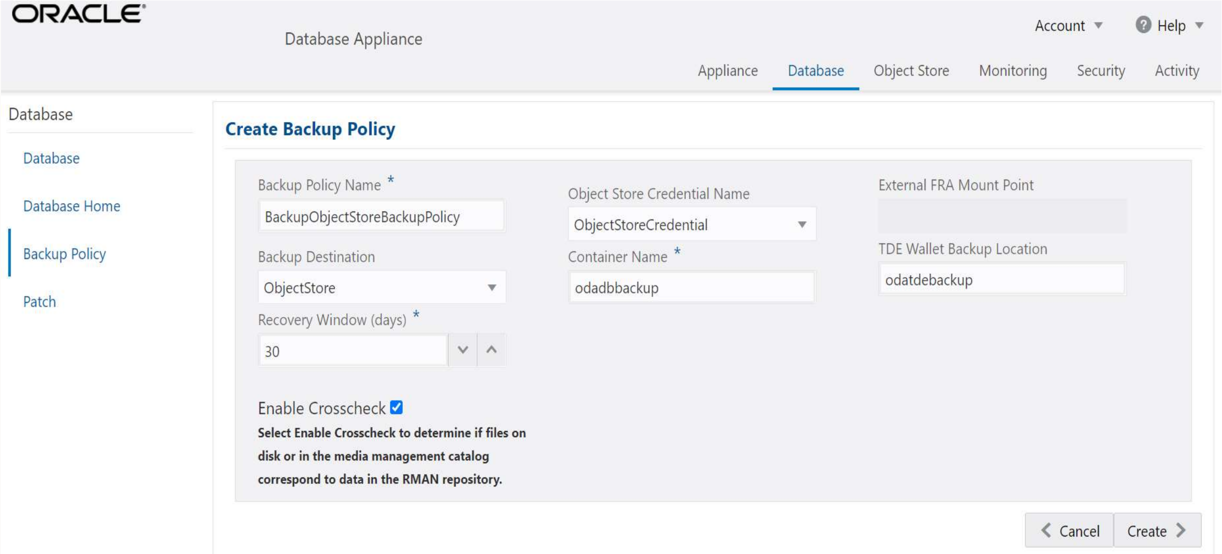
2. Create a Database Backup Policy
   Ensure all **Prequisites** are in place depending upon the Backup Destination before creating a Database Backup Policy
   The Backup Policy defines the backup details. When you create a backup policy, you define the destination for the database backups, either Internal FRA (Disk) or External FRA (NFS location), or Cloud Object Storage, you define the recovery window, enable and disable crosscheck
   Follow these steps to create a backup policy from the Browser User Interface:
   - Log into the Browser User Interface using user oda-admin
     https://host name or ip-address:7093/mgmt/index.html
   - Click the **Database** tab in the Browser User interface
   - Click the **Backup Policy** in the left navigation to display a list of available backup policies
   - Click **Create Backup Policy**
   - Specify name for the **Backup Policy**. Select the number of days for the **Recovery Window**. Select **Enable Crosscheck** to determine if the files on the disk on in the media management catalog correspond to data in the RMAN repository.
   - Select one of the following as the backup destination
     - To backup to disk, select **Internal FRA** as the backup destination.
     - To backup to the cloud, select **Object Store** as the backup destination. If you have more than one Object Store, then select the Object Store Credential Name from the list. Enter a name in the Container Name field.
     - To backup to an NFS location, select **External FRA** as the backup destination, and specify the NFS mount point location.
     - if database is enabled for TDE, then you must specify **TDE Wallet Backup Location,** otherwise not required
       - If database is deployed using ASM storage, then default TDE wallet location is: +DATA/<DBNAME>/tde
       - If database is deployed using ACFS file system, then default TDE wallet location is: /u02/app/oracle/oradata/<DBNAME>/tde

     For example,
     For example, after specifying values **Backup to Disk** to create backup policy for the entries (Backup Policy Name as BackuptoDiskBackupPolicy, Backup Destination as Internal FRA,Recovery Window (Days) as 7 and Enable Crosscheck), user interface screen looks similar to below



Backup Policy Name *: BackuptoDiskBackupPolicy
Backup Destination: Internal FRA
Recovery Window (days) *: 7
Enable Crosscheck ☑
Select Enable Crosscheck to determine if files on disk or in the media management catalog correspond to data in the RMAN repository.
Object Store Credential Name:
Container Name:
External FRA Mount Point:
TDE Wallet Backup Location:

     For example, after specifying values **Backup to Cloud** to create backup policy for the entries (Backup Policy Name as BackupObjectStoreBackupPolicy, Backup Destination as ObjectStore,Recovery Window (Days) as 30, Container as odadbbackup for database backup, TDE Wallet Backup Location as odatdebackup for TDE wallet backup and Enable Crosscheck), user interface screen looks similar to below



Create Backup Policy

Backup Policy Name *: BackupObjectStoreBackupPolicy

Backup Destination: ObjectStore

Recovery Window (days) *: 30

Enable Crosscheck ☑

Select Enable Crosscheck to determine if files on disk or in the media management catalog correspond to data in the RMAN repository.

Object Store Credential Name: ObjectStoreCredential

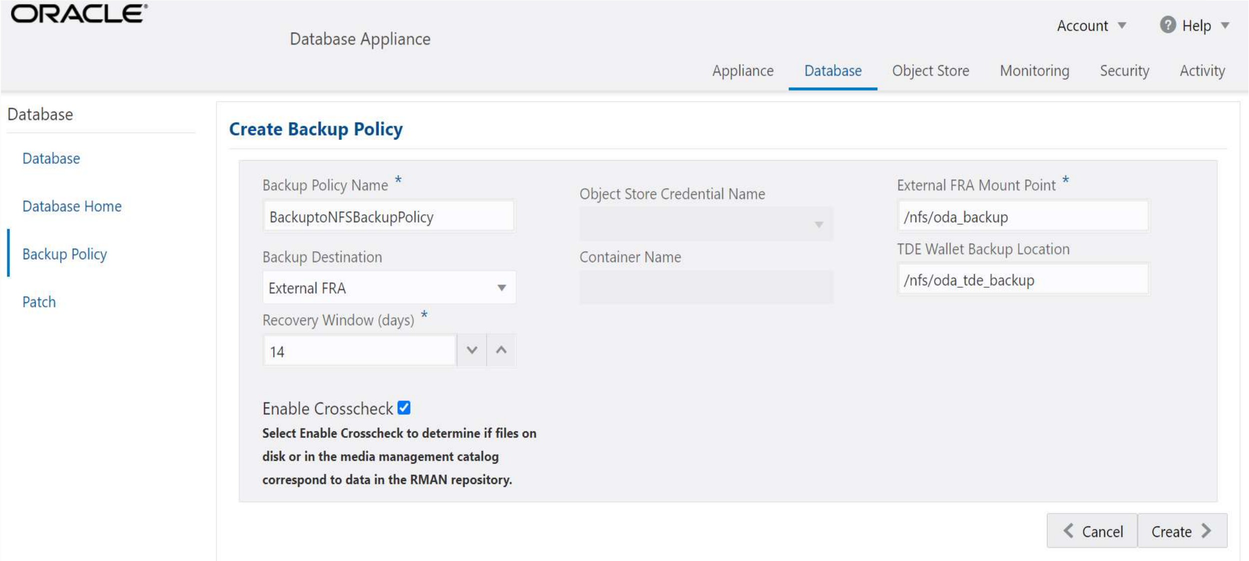
Container Name *: odadbbackup

External FRA Mount Point:

TDE Wallet Backup Location: odatdebackup

Note, container (bucket) should be different for database and TDE wallet backup

For example, after specifying values **Backup to NFS Location** to create backup policy for the entries (Backup Policy Name as BackuptoNFSBackupPolicy, Backup Destination as External FRA,Recovery Window (Days) as 14, External FRA Mount Point as /nfs/oda_backup for database backup, Optionally TDE Wallet Backup Location as /nfs/oda_backup and Enable Crosscheck), user interface screen looks similar to below



Create Backup Policy

Backup Policy Name *: BackuptoNFSBackupPolicy

Backup Destination: External FRA

Recovery Window (days) *: 14

Enable Crosscheck ☑

Select Enable Crosscheck to determine if files on disk or in the media management catalog correspond to data in the RMAN repository.

Object Store Credential Name:

Container Name:

External FRA Mount Point *: /nfs/oda_backup

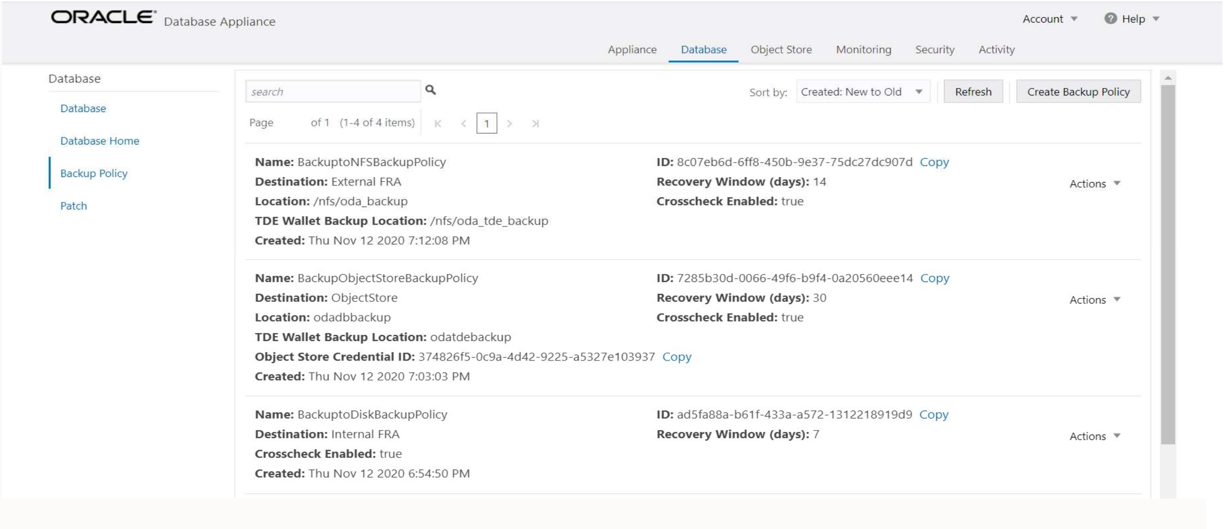
TDE Wallet Backup Location: /nfs/oda_tde_backup

Note, Mount Point location should be different for database and TDE wallet backup

- Click **Create.** Click **Yes** to Confirm that “Are you sure you want to create a backup policy"

A link to the job appears. When the job completes successfully, the backup configuration is ready.

After creating a Backup Policy, you can see all created Backup policies under tab (Database=> Backup Policy)



Name: BackuptoNFSBackupPolicy
Destination: External FRA
Location: /nfs/oda_backup
TDE Wallet Backup Location: /nfs/oda_tde_backup
Created: Thu Nov 12 2020 7:12:08 PM
ID: 8c07eb6d-6ff8-450b-9e37-75dc27dc907d Copy
Recovery Window (days): 14
Crosscheck Enabled: true

Name: BackupObjectStoreBackupPolicy
Destination: ObjectStore
Location: odadbbackup
TDE Wallet Backup Location: odatdebackup
Object Store Credential ID: 374826f5-0c9a-4d42-9225-a5327e103937 Copy
Created: Thu Nov 12 2020 7:03:03 PM
ID: 7285b30d-0066-49f6-b9f4-0a20560eee14 Copy
Recovery Window (days): 30
Crosscheck Enabled: true

Name: BackuptoDiskBackupPolicy
Destination: Internal FRA
Crosscheck Enabled: true
Created: Thu Nov 12 2020 6:54:50 PM
ID: ad5fa88a-b61f-433a-a572-1312218919d9 Copy
Recovery Window (days): 7

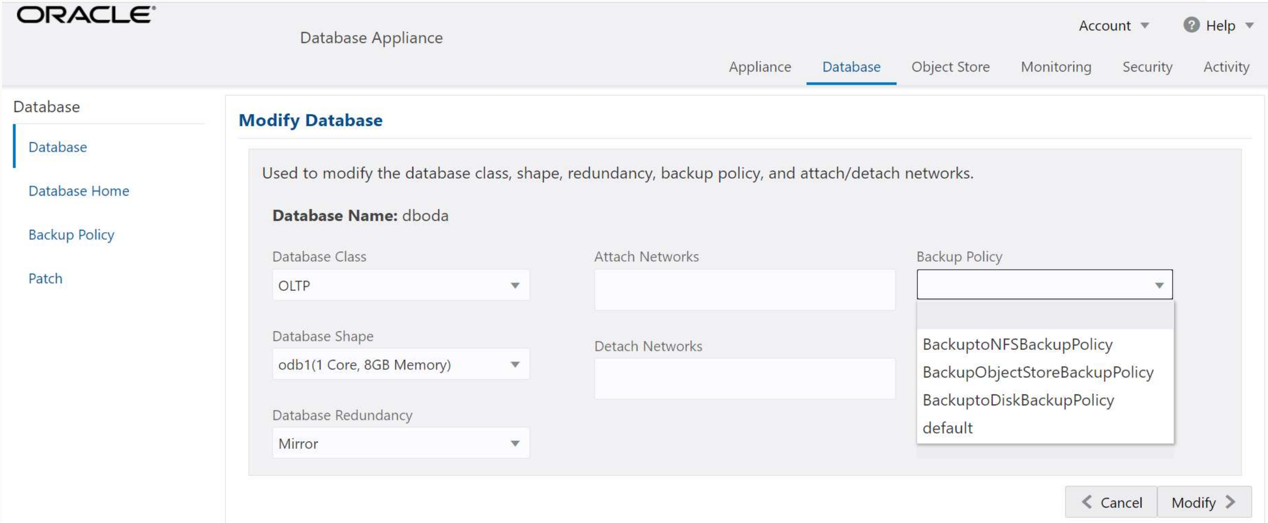
3. Attach a Backup Policy to Database

   Associate the database with this backup policy, either during database creation, or by updating the backup policy for the database Attach a backup policy to a database to define the database backup attributes and destination.

   Follow these steps to attach a backup policy from the Browser User Interface:

- Log into the Browser User Interface using user oda-admin
  https://host name or ip-address:7093/mgmt/index.html
- Click the **Database** tab in the Browser User interface and then select a database from the list.
- For selected database, under **Actions** tab, Click **Modify**
- Select **Backup Policy** from the list of available backup policies (which was created in step 2 above)
- Specify **Backup Encryption Password** depending upon below criteria
  - For TDE-enabled databases
    - Backups are encrypted by default and do not require the RMAN backup encryption password to be specified separately, irrespective of the backup destination.
  - For non-TDE enabled databases
    - When you backup to Disk, no need to specify the RMAN backup encryption password
    - When you backup to NFS location, specifying the RMAN backup encryption password is optional
    - When you backup to Object Storage, specifying the RMAN backup encryption password is mandatory

  Select the Backup Policy from the list and user interface screen looks similar to below



Used to modify the database class, shape, redundancy, backup policy, and attach/detach networks.

Database Name: dboda

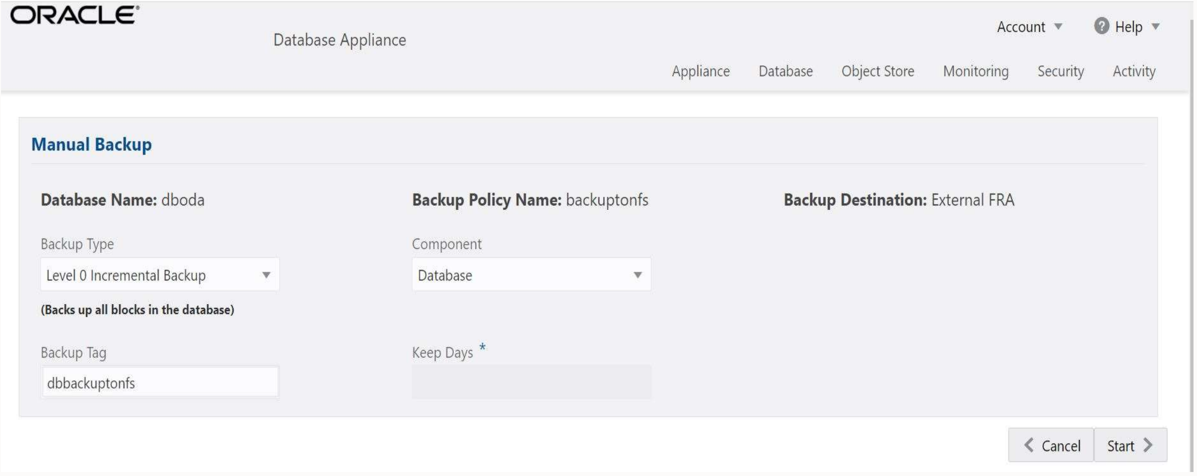
Database Class: OLTP
Database Shape: odb1(1 Core, 8GB Memory)
Database Redundancy: Mirror
Attach Networks
Detach Networks
Backup Policy:
BackuptoNFSBackupPolicy
BackupObjectStoreBackupPolicy
BackuptoDiskBackupPolicy
default

- Click **Modify**, and click **Yes** to Confirm that “"Are you you want modify the database?"
  A link to the job appears. When the job completes successfully, the backup policy is attached to the database.

Once a backup policy is attached to a database, the dcs-agent schedules daily automatic backups for the database. It also schedules archivelog backups for the database. By default, the frequency of the archivelog backup is 30 minutes. The default schedule is a level 0 backup every Sunday and a level 1 backup Monday through Saturday. You can change (Update Database Backup Schedule and Update Archive Log Backup schedule) or disable the schedule.

4. Perform a Backups using Browser User Interface
   After attaching the backup policy, backups are automatically scheduled, and you can also run manual backups. You can specify manual backup options in the Browser User Interface
   - Backing Up a Database Using the Browser User Interface
     Before creating a database backup, you must have a backup policy and must associate a backup policy with the database, otherwise you cannot create backups.
     Follow these steps to attach a backup policy from the Browser User Interface:
     - Log into the Browser User Interface using user oda-admin
       https://host name or ip-address:7093/mgmt/index.html
     - Click the **Database** tab in the Browser User interface and then select a database from the list.
     - Click on Database name, for which you need to take a backup
     - Review the database information, including the backup policy name and destination details and scroll down the screen
     - Click the **Manual Backup** and then
       - Specify **Backup Type** as (Level 0 Incremental Backup or Level 1 Incremental Backup or Archive Log Backup or Lonterm Backup)
         - If you select **Backup Type** as Lonterm Backup, then specify **Keep Days,** otherwise not required
       - Specify **Backup Tag**
       - Select **Component**
         - For Database Backup: Database
         - For TDE Wallet Backup: TDE Wallet

     For example, after specifying values to Back up Database for the entries (Backup Type as Level 0 Incremental Backup, Backup Tag as dbbackuptonfs and Component as Database), user interface screen looks similar to below



Manual Backup

Database Name: dboda

Backup Policy Name: backuptonfs

Backup Destination: External FRA

Backup Type: Level 0 Incremental Backup

(Backs up all blocks in the database)

Component: Database

Backup Tag: dbbackuptonfs

Keep Days *

     - Click **Start** and Click **Yes** to confirm that "Are you sure you want to start a backup?"

     A link to the job appears. When the job completes successfully, the backup is ready. A list of backups appears at the bottom of the page.

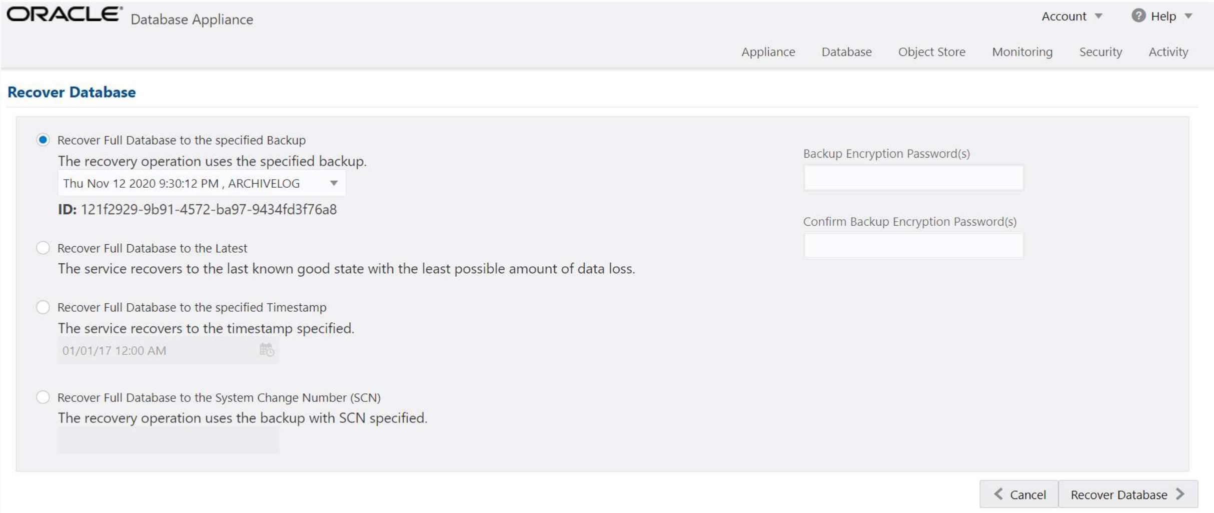
     The dcs-agent generates and saves a backup report for each backup. The backup report contains the metadata required to recover or restore a database.

5. Perform a Restore and recover operations using Browser User Interface
   Recovering a database in Oracle Database Appliance is a full RMAN database recovery.
   - Recovering a Database Using the Browser User Interface
     Follow these steps to recovering a database using Browser User Interface:
     - Log into the Browser User Interface using user oda-admin

https://host name or ip-address:7093/mgmt/index.html

- Click the **Database** tab in the Browser User interface and then select a database from the list.
- Click on Database name and on the Database Information page, click **Recover**

Then User interface screen looks similar to below with Recovery options



Recover Database

● Recover Full Database to the specified Backup
The recovery operation uses the specified backup.
Thu Nov 12 2020 9:30:12 PM , ARCHIVELOG
ID: 121f2929-9b91-4572-ba97-9434fd3f76a8

○ Recover Full Database to the Latest
The service recovers to the last known good state with the least possible amount of data loss.

○ Recover Full Database to the specified Timestamp
The service recovers to the timestamp specified.
01/01/17 12:00 AM

○ Recover Full Database to the System Change Number (SCN)
The recovery operation uses the backup with SCN specified.

Backup Encryption Password(s)

Confirm Backup Encryption Password(s)

The Recovery options are

Recover Full Database to the specified Backup: Select the existing backup from which you want to recover the database

Recover Full Database to the Latest: Select this option to recover the database from the last known good state, with the least possible data loss.

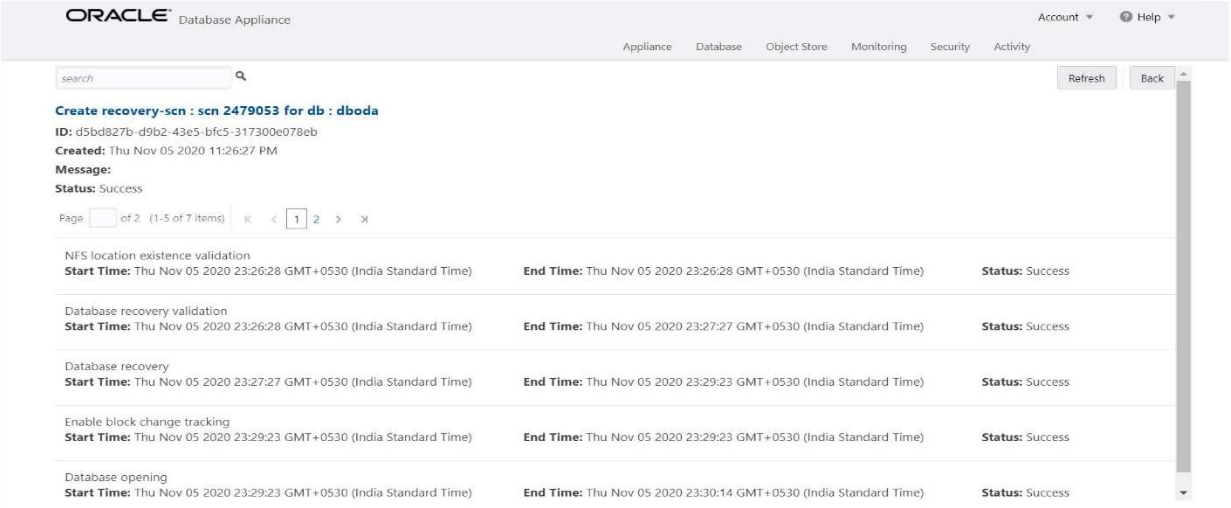
Recover Full Database to the specified Timestamp: Specify the timestamp to recover the database.

Recover Full Database to the System Change Number (SCN): Specify the SCN of the backup from which you want to recover the database.

- On the Recover Database page, depending upon your requirement, select any of the above recovery options
- Sepcify **Backup Encryption Password(s)**
  - For TDE enabled databases, no need to specify RMAN backup encryption password
  - For non-TDE databases, if backup would have taken with RMAN backup encryption password, then only specify same RMAN backup encryption password, otherwise no need to specify any password
- Click **Recover Database** to recover Database
- Click **Yes** to confirm "Are you sure you want to start a recovery"

A link to the job appears. When the job completes successfully, the database is recovered as per the specified recovery options

After recovery is completed, job status screen looks similar to below



Create recovery-scn : scn 2479053 for db : dboda
ID: d5bd827b-d9b2-43e5-bfc5-317300e078eb
Created: Thu Nov 05 2020 11:26:27 PM
Message:
Status: Success

NFS location existence validation
Start Time: Thu Nov 05 2020 23:26:28 GMT+0530 (India Standard Time) **End Time:** Thu Nov 05 2020 23:26:28 GMT+0530 (India Standard Time) **Status:** Success

Database recovery validation
Start Time: Thu Nov 05 2020 23:26:28 GMT+0530 (India Standard Time) **End Time:** Thu Nov 05 2020 23:27:27 GMT+0530 (India Standard Time) **Status:** Success

Database recovery
Start Time: Thu Nov 05 2020 23:27:27 GMT+0530 (India Standard Time) **End Time:** Thu Nov 05 2020 23:29:23 GMT+0530 (India Standard Time) **Status:** Success

Enable block change tracking
Start Time: Thu Nov 05 2020 23:29:23 GMT+0530 (India Standard Time) **End Time:** Thu Nov 05 2020 23:29:23 GMT+0530 (India Standard Time) **Status:** Success

Database opening
Start Time: Thu Nov 05 2020 23:29:23 GMT+0530 (India Standard Time) **End Time:** Thu Nov 05 2020 23:30:14 GMT+0530 (India Standard Time) **Status:** Success

BACKUP AND RECOVERY IN ORACLE CLOUD

This section describes the procedure to backup databases deployed on Oracle Database Appliance to Oracle Database Backup Cloud Service and restore them from those backups. Specifically, this section assumes you are backing up to the Object Storage in Oracle Cloud Infrastructure (OCI).

Note, databases, which are deployed on Bare Metal System and on Oracle Database Appliance Release 19.9 and later, recommendation is to use ODA tooling (Brower User Interface and CLI) for backup and recovery operations.

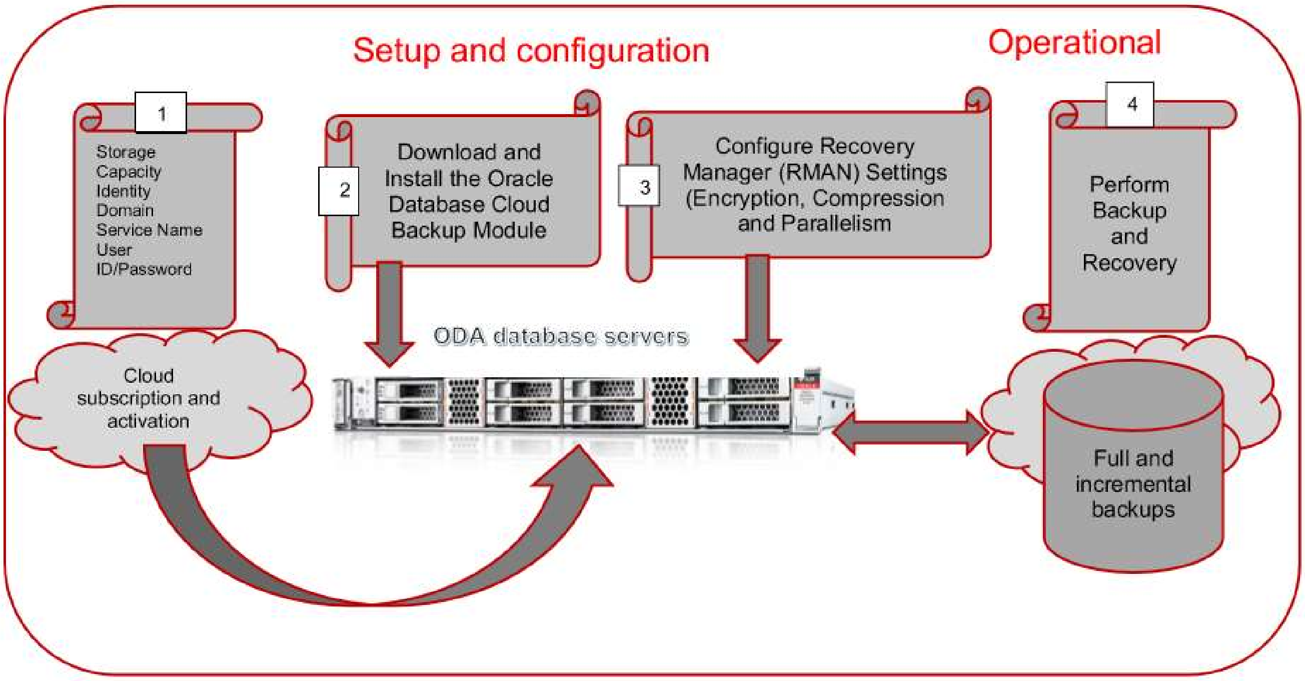

Setup and configuration
Operational
1
Storage
Capacity
Identity
Domain
Service Name
User
ID/Password
2
Download and Install the Oracle Database Cloud Backup Module
3
Configure Recovery Manager (RMAN) Settings (Encryption, Compression and Parallelism
4
Perform Backup and Recovery
ODA database servers
Cloud subscription and activation
Full and incremental backups

FIGURE 1: BACKUP TO ORACLE CLOUD

The high level procedure to setup backups to Oracle Cloud is as follows:

(1) Purchase the Oracle Database Backup Cloud Service subscription (or request a trial)

(2) Download and install the Oracle database cloud backup module

(3) Configure environment with Recovery Manager (RMAN) settings

(4) Use familiar recovery manager (RMAN) commands to perform backups using Oracle Database Backup Cloud Service

(5) Perform restore and recover operations from Oracle Database Backup Cloud Service, when necessary

1. Purchase the Oracle Database Backup Cloud Service subscription (or request a trial)

   To get started with the Oracle Database Backup Cloud Service, you may request a trial subscription or purchase a service at cloud.oracle.com/database_backup or cloud.oracle.com/database

2. Download Oracle Database Cloud Backup Module

Download the Oracle Database Cloud Backup Module from Oracle Technology Network (OTN). The module enables cloud backups and restores. Install the backup module on the system (host(s)) where your Oracle database is running. Download the Oracle Database Cloud Backup Module (opc_installer.zip) from Oracle Technology Network (OTN): https://www.oracle.com/database/technologies/oracle-cloud-backup-downloads.html

3. Install Oracle Database Cloud Backup Module

3.1 Preparing to run the Database Cloud Backup Module installer

Before running the installer, using below url you need to gather some information from your cloud account https://docs.cloud.oracle.com/en-us/iaas/Content/API/Concepts/apisigningkey.htm https://docs.oracle.com/en/cloud/paas/db-backup-cloud/csdbb/installing-backup-module-oracle-database-backup-cloud-service.html

3.2 Running the DB Cloud Backup Module Installer

Assume Oracle Database Cloud Backup Module for OCI is downloaded and unzipped onto oda database server and location is

```
/home/oracle/opc_installer/opc_installer/oci_installer>ls
oci_install.jar  oci_readme.txt
```

Login into database server and switch to oracle user

Set the environment

Run the installer using the parameters that you prepared in advance from above:

```
/home/oracle> java -jar <Location of OCI installer file>

-host https://objectstorage.us-<region>-1.oraclecloud.com

-pvtKeyFile <oci_private_key>

-pubFingerPrint <oci_public_fingerprint>

-uOCID <User OCID for the OCI account>

-tOCID <Tenancy OCID for the OCI account>

-walletDir <Directory in which Oracle Cloud Infrastructure Object Storage credentials are stored>

-libDir <Directory in which the system backup to tape (SBT) library used for backups and restores with OCI is stored>
```

For example,

```
/home/oracle>cd /home/oracle/opc_installer/opc_installer/oci_installer

/home/oracle> java -jar oci_install.jar

-host https://objectstorage.us-phoenix-1.oraclecloud.com

-pvtKeyFile /home/oracle/.oci/oci_api_key.pem

-pubFingerPrint 53:8x:bb:f1:4q:b1:z9:sb:84:99:1q:41:0e:6b:9t:kl

-uOCID ocid1.user.oc1..zjpasmsfyro7yiodp4a24hg3ecikgrlnbk2ts2kq6e6ba

-tOCID ocid1.tenancy.oc1..aaaaaaaawp63anwr2qovebh23qggqt72baxykzm5bsvxm3d53bh67iypsmea

-walletDir /u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/opc_wallet

-libDir /u01/app/oracle/product/12.2.0.1/dbhome_1/lib

-bucket odabackup

Oracle Database Cloud Backup Module Install Tool, build 19.3.0.0.0DBBKPCSBP_2019-10-16

Oracle Database Cloud Backup Module credentials are valid.

Backups would be sent to bucket odabackup.

Oracle Database Cloud Backup Module wallet created in directory

/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/opc_wallet.

Oracle Database Cloud Backup Module initialization file
```

```
/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/opcdboda1.ora created.

Downloading Oracle Database Cloud Backup Module Software Library from Oracle Cloud Infrastructure.

Download complete.
```

Note: Specify -bucket parameter in installer command to store backups in custom location. Otherwise, a default location is created and named as "oracle-data-[first 8 chars of service & tenant]" when you install the Oracle Database Cloud Backup Module for OCI and backups are stored as objects in this location.

After the Oracle Database Backup Cloud Module is installed, validate that the following files exist. These files are used by Oracle Database Backup Cloud Service to perform backups and restores to Oracle Cloud.

File	Location	Purpose
libopc.so	As specified for the -libDir parameter when you run the installer for the Oracle Database Cloud Backup Module for OCI. For example: $ORACLE_HOME/lib	Operating system-specific SBT library that enables cloud backups and restores with the Oracle Cloud Infrastructure
opc$ORACLE_SID.ora	As specified for the -configFile parameter when you run the installer for the Oracle Database Cloud Backup Module for OCI. Default location is under $ORACLE_HOME/dbs	Configuration file that contains the Oracle Cloud Infrastructure Object Storage bucket URL and credential wallet location, where SID is the system identifier of the Oracle database being backed up to Oracle Cloud Infrastructure.
cwallet.sso	As specified for the -walletDir parameter when you run the Oracle Database Cloud Backup Module for OCI installer For example: $ORACLE_HOME/dbs/opc_wallet	Oracle wallet file that securely stores Oracle Cloud Infrastructure Object Storage credentials. This file is used during Recovery Manager (RMAN) backup, restore operations, and is stored in the Oracle Cloud Infrastructure Object Storage wallet directory (for example, opc_wallet)
Wallet for encryption (optional – only needed for TDE)	Wallet location defined in $ORACLE_HOME/network/admin/sqlnet.ora	Used for backup encryption.

TABLE 1: ORACLE DATABASE BACKUP CLOUD MODULE RELATED FILES

Note: If your database server has multiple ORACLE_HOMEs, then the Oracle Database Backup Cloud Module must be installed into each ORACLE_HOME. Alternatively, you can copy the library file (libopc.so) to other ORACLE_HOME library location of each ORACLE_HOME, along with the opc$ORACLE_SID.ora configuration file (assuming you're using the same cloud credentials for backing up all databases in the database server). If using the latter approach, copy and rename the opc$ORACLE_SID.ora file for each database instance you are backing up to the cloud, where ORACLE_SID matches the SID for the database instance

4. Configure RMAN Settings
   Once the Oracle Database Backup Cloud Module is installed, configure Recovery Manager (RMAN) to use Oracle Database Backup Cloud Service as the backup destination.

4.1 Configures RMAN channel

Configure RMAN channel to use the backup module SBT library and the provided configuration file for backup to the cloud

```
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt PARMS='SBT_LIBRARY=location-of-the-SBTlibrary,
SBT_PARMS=(OPC_PFILE=location-of-the-configuration file)';
```

For example:

```
RMAN>CONFIGURE CHANNEL DEVICE TYPE sbt
PARMS='SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libopc.so,SBT_
PARMS=(OPC_PFILE=/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/opcdboda1.ora)';
```

In above example, ORACLE_HOME is /u01/app/oracle/product/12.2.0.1/dbhome_1 and ORACLE_SID is dboda1

4.2 Configure autobackup

Configure RMAN to automatically back up the database control file and server parameter file.

```
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

If you have configured control file autobackup, then the server parameter file is backed up with the control file whenever an autobackup is taken

4.3 Setup PARALLELISM for backup processes

Use multiple RMAN channels for higher parallelism for utilizing maximum available network capacity. You can configure as many RMAN channels as you want. For example, the following configuration uses four channels in parallel to back up to the Cloud.

```
RMAN> CONFIGURE DEVICE TYPE 'SBT_TAPE' PARALLELISM 4 BACKUP TYPE TO BACKUPSET;
```

You can gradually increase parallelism to identify the optimal value for your environment that allows you to achieve an optimal transfer rate without affecting CPU/IO usage and database response time.

4.4 Multi-section backup

Multi-section backups allow you to break up large backup objects (say large data files) into smaller, granular chunks. This helps optimize throughput when parallelism is used for backup processes.

```
RMAN> BACKUP DEVICE TYPE sbt DATABASE SECTION SIZE 1g;
```

4.5 Configuring Compression for Backups

Recovery Manager (RMAN) supports binary compression using HIGH, MEDIUM, BASIC, and LOW compression levels. The recommended compression level for cloud backups is MEDIUM.

The following RMAN commands configure compression using the MEDIUM algorithm:

```
RMAN> CONFIGURE COMPRESSION ALGORITHM 'MEDIUM';

RMAN> CONFIGURE DEVICE TYPE sbt BACKUP TYPE TO COMPRESSED BACKUPSET;
```

4.6 Keep the control file retention the same as the backup retention peroid

Set control_file_record_keep_time same as recovery window

```
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF x DAYS;
```

For example,

```
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 30 DAYS;

/home/oracle> sqlplus "/as sysdba"

SQL> ALTER SYSTEM SET control_file_record_keep_time = 30  SCOPE=both SID='*'
```

4.7 Define your backup strategy

For many situations, you may find a weekly full and daily incremental backup strategy to be an optimal choice. This strategy results in faster backups and could save a significant amount of network bandwidth during backup operations. Use the RMAN fast incremental backup feature (based on block change tracking) to optimize the performance of your daily incremental backups.

4.8 Configure RMAN encryption for backups

Recovery Manager (RMAN) encrypted backups are securely created, transmitted, and stored in the cloud. Use one of the following RMAN encryption modes to encrypt your backups: Note: The customer manages the keys each of the following encryption modes and data is securely transmitted to the Cloud over HTTPS.

4.8.1 Password encryption

RMAN encryption for backups is enforced (mandatory for On-Premise Databases)

Use the RMAN set encryption clause in RMAN run block.

RMAN> SET ENCRYPTION ON IDENTIFIED BY '<Specify password>' ONLY;

For example:

RMAN> SET ENCRYPTION ON IDENTIFIED BY 'odabackup123' ONLY;

Note: The password must be specified each time you backup and restore. If you forget or lose the password, you will not be able to restore your backup.

4.8.2 Transparent Data Encryption (TDE)

To backup using Transparent Data Encryption you need to have a TDE wallet (TDE keystore), which is different from the OPC wallet that stores Oracle Database Backup Cloud Service credentials.

With TDE you don't need to provide a password every time you create or restore a backup.

If TDE wallet is not present, to create a TDE wallet, refer section APPENDIX E SETTING UP THE TRANSPARENT DATA ENCRYPTION WALLET (TDE)

4.8.3 Dual-mode encryption (combination of password and TDE)

You can use dual-mode encryption to back up to Oracle Database Backup Cloud Service. Dual-mode encryption is a combination of password encryption and Transparent Data Encryption (TDE).

4.9 Perform backup / restore and recovery using Oracle Database Backup Cloud Service

You can now connect to the target database and configure an RMAN channel, then issue standard RMAN backup, restore, and recovery commands. Use the online dashboard to monitor your service and storage capacity being used by your backups. If needed, additional storage capacity can be added. Refer to Appendix C for sample commands.

4.10 To diagnose Oracle Cloud Backup Performance, see MOS note 2078576.1

BACKUP AND RECOVERY IN ZERO DATA LOSS RECOVERY APPLIANCE

This section describes the procedure to backup databases deployed on Oracle Database Appliance to Zero Data Loss Reocvery Appliance (ZDLRA) and restore them from those backups.

ZDLRA terminology

- Recovery Appliance administrator
  The administrator who manages a Recovery Appliance. Typical duties include creating and adding databases to protection policies, managing storage space, managing user accounts, configuring tape backups and the Recovery Appliance replication, and monitoring the Recovery Appliance.
- Protected database
  A client database (deployed on ODA) that backs up data to a Recovery Appliance.
- RMAN recovery catalog
  A set of metadata views residing in the Recovery Appliance metadata database.
- Virtual private catalog
  A subset of the metadata in a base RMAN recovery catalog to which a database user account is granted access. Each restricted user account has full read/write access to its own virtual private catalog.
- Protection policy
  A group of attributes (recovery window and estimated space is required for backup) that control how a Recovery Appliance stores and maintains backup data. Each protected database is assigned to exactly one protection policy, which controls all aspects of backup processing for that client.
- Recovery Appliance metadata database

The Oracle database that runs inside of the Recovery Appliance. This database stores configuration data such as user definitions, protection policy definitions, and client database definitions. The metadata database also stores backup metadata, including the contents of the delta store.

- Recovery Appliance user account
  A user account that is authorized to connect to, and request services from, Recovery Appliance. Every Recovery Appliance user account is an Oracle Database user account on the Recovery Appliance metadata database, and the owner of a virtual private catalog. When RMAN backs up a protected database, it connects to the recovery catalog with the Recovery Appliance user account credentials.
- Recovery Appliance Backup Module
  An Oracle-supplied SBT library that RMAN uses to send backups of protected databases over the network to the Recovery Appliance. The library must be installed in each Oracle home used by a protected database.The module functions as an SBT media management library that RMAN references when allocating or configuring a channel for backup to the Recovery Appliance. RMAN performs all backups to the Recovery Appliance, and all restores of complete backup sets, using this module.
- About Backup Encryption and Recovery Appliance
  You can configure protected databases to use backup encryption. If a backup is encrypted during an RMAN backup operation to Recovery Appliance, then the backup remains encrypted on the Recovery Appliance. A subsequent copy of this backup totape will also remain in an encrypted format. However, Oracle recommends that you avoid using RMAN backup encryption when performing backups to Recovery Appliance. Encrypted backups are not ingested by Recovery Appliance and cannot be used to construct virtual full backups or be part of an incremental-forever backup strategy. Backups that are copied to tape from the Recovery Appliance can be encrypted using hardware-based encryption on tape drives or using Oracle Secure Backup
- Tools for Protected Database Operations
  Recovery Appliance provides multiple interfaces to manage backup and recovery operations for protected databases. T
  - Oracle Enterprise Manager Cloud Control (Cloud Control)
    - Cloud Control provides a GUI for administering, managing, and monitoring a Recovery Appliance environment. It also enables you to configure, back up, and recover protected databases.Additional information about using Cloud Control is available in the Cloud Controlonline help
  - RMAN client
    - Recovery Appliance is integrated with RMAN and you can use the RMAN client installed on your protected database to configure, back up, and recover protecteddatabases.
  - SQL*Plus
    - SQL*Plus is a command-line tool that you can use to query the Recovery Appli-ance catalog and run the DBMS_RA PL/SQL package.

The interfaces (RMAN client and SQL*Plus) are used to show examples to perform backup and recovery operations to ZDLRA

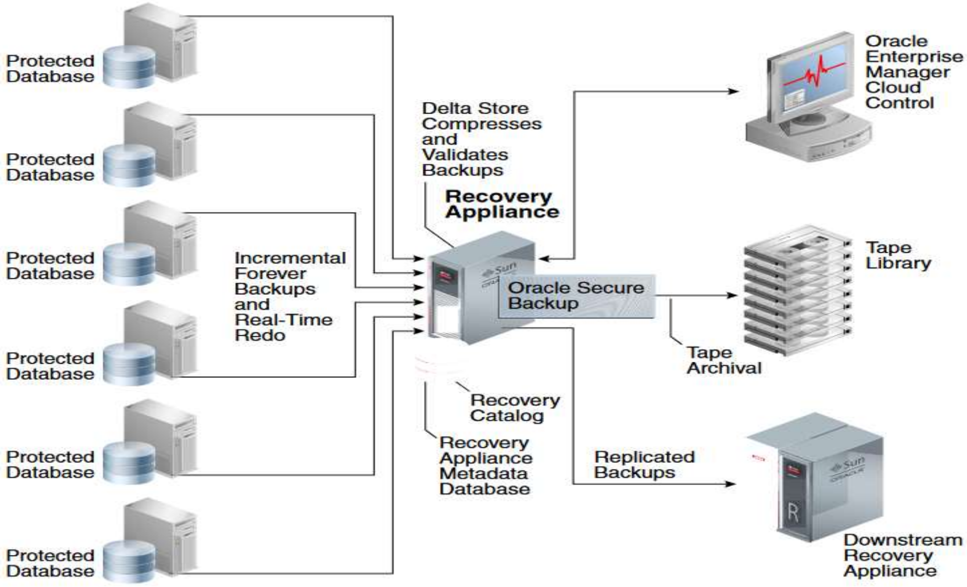


FIGURE 2: Recovery Appliance Architecture and Protected Databases

The high level procedure to setup backups to and recovery from Recovery Appliance is as follows:

1. Configure the Recovery Appliance
2. Enroll the protected database with a Recovery Appliance
3. Configure backup and recovery settings using Recovery Manager (RMAN) for the protected database
4. Use familiar recovery manager (RMAN) commands to perform backup and recovery operations

To use Recovery Appliance as a centralized repository for your protected database backups, configuration is required both on the Recovery Appliance and on the protected database

1. On the Recovery Appliance, Recovery Appliance administrator need to perform following configuration steps
    - Create a protection policy that is assigned to the protected database
    - Create a Recovery Appliance user who owns the virtual private catalog
    - Grant a access for the protected databases to Recovery Appliance user
      Grant the privileges required for performing backup and recovery operations to the Recovery Appliance user that the protected database will use for authentication.This Recovery Appliance user owns the virtual private catalog that stores metadata for the protected database
2. On the protected database, the configuration includes
    - Enroll a protected database with a Recovery Appliance using RMAN
        - Install the Recovery Appliance backup module
          Protected databases communicate with the Recovery Appliance through the Recovery Appliance backup module. You must install the backup module on the protected database host before you enroll the protected database with Recovery Appliance.
          The backup module installation creates the Oracle wallet that stores the credentials required to access or authenticate the protected database with Recovery Appliance and installs the shared library that transfers backup data to the Recovery Appliance.
            - Preparing to Install the Recovery Appliance Backup Module
                - Verify that you have Java version 1.5 or higher
                - Contact the Recovery Appliance administrator and obtain the following information
                    - Recovery Appliance host name and port number
                    - Credentials of the Recovery Appliance user that will be used to authenticatethe protected database with the Recovery Appliance.
                      The permissions required to perform protected database backup and recovery operations need to be assigned to this Recovery Appliance user
            - Obtaining the Installer for the Recovery Appliance Backup Module
                - Download the Recovery Appliance backup module installer from OTN
                    - Access the following URL on OTN
                      http://www.oracle.com/technetwork/database/availability/oracle-zdlra-backup-module-2279224.html
                    - Sign in using your OTN account credentials
                    - Select Accept License Agreement to accept the OTN license agreement
                    - Click All Supported Platforms to download the Recovery Appliance backup module for your platform
                      The Recovery Appliance installer is named ra_installer.zip
                - Install the Recovery Appliance backup module
                    - Unzip the installer downloaded in above step into a local directory
                        - Login into protected database server and switch to oracle user
                        - For example, downloaded location is /home/oracle/zdlra and unzipped into location
                            - /home/oracle/zdlra> unzip ra_installer.zip
                        - After the unzip, the installer contains the files: ra_install.jar and ra_readme.txt
                            - /home/oracle/zdlra> ls
                              ra_install.jar ra_readme.txt
                    - Set the environment and ensure that the ORACLE_HOME environment variable is set to the Oracle home of the protected database
                    - Prepare the parameters required to install Recovery Appliance backup module

Parameter Name	Description
dbUser	User name of the Recovery Appliance user who has the privileges required to connect to, send,and receive backups for the protected database

dbPass	Password for the dbUser user
Host	SCAN host name of the Recovery Appliance
Port	Listener port number of the Recovery Appliance metadata database
serviceName	Service name of the Recovery Appliance metadata database
walletDir	Location of the Oracle wallet that stores the Recovery Appliance user credentials and the proxy information used to connect to the Recovery Appliance. Note: If an Oracle wallet already exists in this directory, then the Recovery Appliance backup module installer overwrites the existing wallet
libDir	Location where the shared library for the Recovery Appliance backup module is stored. This library is used to transfer backup data over thenetwork to the Recovery Appliance. It is recommended that you store the shared library in $ORACLE_HOME/lib Downloading the library is not required only when you regenerate the Oracle wallet and configuration file in an Oracle home where the backup module was previously installed
configFile	Location of the configuration file that stores the configuration parameters for the Recovery Appliance backup module. The default location is $ORACLE_HOME/dbs/ra$ORACLE_SID.ora

TABLE 2: ORACLE DATABASE BACKUP ZDLRA MODULE RELATED FILES

- Execute below step to install the backup module

  -
    ```
    /home/oracle/zdlra>java -jar /home/oracle/zdlra/ra_install.jar
    -dbUser <Recovery Appliance user name>
    -dbPass <Recovery Appliance user password>
    -host <SCAN host name of the Recovery Appliance>
    -serviceName < Service name of the Recovery Appliance metadata database>
    -walletDir < Location of the Oracle wallet that stores the Recovery Appliance user credentials and the proxy information used to connect to the Recovery Appliance>
    -libDir < Location where the shared library for the Recovery Appliance backup module is stored>
    ```
  - For example,
    ```
    /home/oracle/zdlra>java -jar /home/oracle/zdlra/ra_install.jar -dbUser zdlravpc -dbPass ****** -host zdlra-scan.example.com -port 1521  -serviceName zdlra -walletDir $ORACLE_HOME/dbs/ra_wallet -libDir $ORACLE_HOME/lib
    Recovery Appliance Install Tool, build 12.2.0.1.0DBBKPCSBP_2018-06-12
    Recovery Appliance credentials are valid.
    Recovery Appliance wallet created in directory
    /u01/app/oracle/product/12.1.0.2/dbhome_1/dbs/ra_wallet.
    Recovery Appliance initialization file
    /u01/app/oracle/product/12.1.0.2/dbhome_1/dbs/radbzdlra1.ora created.
    Downloading Recovery Appliance Software Library from file ra_linux64.zip.
    Download complete.
    ```
- After the Backup Module is installed, validate that the following files exist. These files are used to perform backup and restore

File	Location	Purpose
libra.so	As specified for the -libDir parameter when you run the installer for the Backup Module For example: $ORACLE_HOME/lib	Operating system-specific SBT library that enables cloud backups and restores with the Oracle Cloud Infrastructure

ra$ORACLE_SID.ora	As specified for the -configFile parameter when you run the installer for the Backup Module Default location is under $ORACLE_HOME/dbs	Configuration file that contains credential wallet location, Recovery Appliance (SCAN host name, Listener port number of the metadata database and Service name) and , where ORACLE_SID is the system identifier of the protected database being backed up
cwallet.sso	As specified for the -walletDir parameter when you run the Backup Module For example: $ORACLE_HOME/dbs/ra_wallet	Oracle wallet file that that stores the Recovery Appliance user credentials and the proxy information used to connect to the Recovery Appliance. Note: If an Oracle wallet already exists in this directory, then the Recovery Appliance backup module installer overwrites the existing wallet

TABLE 3: ORACLE DATABASE BACKUP ZDLRA MODULE RELATED FILES

For example, a single host might have an Oracle Database 11g Oracle home, and an Oracle Database 12c Oracle home. Each Oracle home might support five protected databases, for a total of ten databases running on the host. In this case, only two Recovery Appliance Backup Modules must be installed: one in each Oracle home.

- Registering a Protected Database with the Recovery Appliance Catalog
  All protected databases must use the Recovery Appliance catalog on the target Recovery Appliance to store protected database backup metadata. Registering the protected database with the Recovery Appliance catalog ensures that metadata for the protected database and its backups is stored in the Recovery Appliance catalog. However, any existing backup metadata stored in an RMAN recovery catalog is not available in the Recovery Appliance catalog unless you import the RMAN recovery catalog into the Recovery Appliance catalog
  - Register a protected database with the Recovery Appliance
    - Obtain the name and password of the Recovery Appliance catalog owner that will store backup metadata for this protected database. Contact the Recovery Appliance administrator for these credentials.
    - Connect to the protected database as TARGET and to the Recovery Appliance catalog as CATALOG
      - /home/oracle> rman target / catalog < Recovery Appliance catalog owner >/<Password>@< SCAN host name of the Recovery Appliance>:< Listener port number of the Recovery Appliance metadata database >/< Service name of the Recovery Appliance metadata database>
      - For example,
        ```
        /home/oracle> rman target / catalog zdlravpc/*****@zdlra-scan.example.com:1521/zdlra:dedicated
        ```
    - Register the protected database
      - Register the protected database using the REGISTER DATABASE command.The following command registers the protected database with the Recovery Appliance
        ```
        RMAN> REGISTER DATABASE;
        ```
        For example,
        ```
        RMAN> REGISTER DATABASE;

        database registered in recovery catalog
        starting full resync of recovery catalog
        full resync complete

        RMAN>
        ```
- Configuring Backup and Recovery Settings for Protected Databases
  Once the Backup Module is installed, configure Recovery Manager (RMAN) to use ZDLRA as backup destination.

- Configure RMAN SBT Channels for Recovery Appliance
  Configure RMAN channel to use the backup module SBT library and the provided configuration file for backup to the ZDLRA

  ```
  RMAN> CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE' PARMS 'SBT_LIBRARY= location-of-the-SBTlibrary,ENV=(RA_WALLET=location=file: location-of-the-configuration file credential_alias=< SCAN host name of the Recovery Appliance>:< Listener port number of the Recovery Appliance metadata database >/< Service name of the Recovery Appliance metadata database>)' FORMAT '%U_%d';
  ```

  For example:

  Below allocates an RMAN SBT channel with the SBT_LIBRARY parameter specifying the complete path of the Recovery Appliance backup module. The ENV setting isused to specify the configuration parameters used by the Recovery Appliance backup module. zdlra-scan.example.com is the SCAN of the Recovery Appliance and zdlra is the servicename of the Recovery Appliance metadata database

  ```
  RMAN> CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE' PARMS
  'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-scan.example.com:1521/zdlra:dedicated)' FORMAT '%U_%d';
  ```

  In above example, ORACLE_HOME is /u01/app/oracle/product/12.2.0.1/dbhome_1

- Configure autobackup

  Configure RMAN to automatically back up the database control file and server parameter file.

  ```
  RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
  ```

  If you have configured control file autobackup, then the server parameter file is backed up with the control file whenever an autobackup is taken

- Keep the control file retention the same as the backup retention peroid

  Set control_file_record_keep_time same as recovery window

  ```
  RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF x DAYS;
  ```

  For example,

  ```
  RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 30 DAYS;
  ```

  ```
  /home/oracle> sqlplus "/as sysdba"
  ```

  ```
  SQL> ALTER SYSTEM SET control_file_record_keep_time = 30  SCOPE=both SID='*'
  ```

- Optionally Configure Real Time Redo Transport
  When you configure real-time redo transport, redo data from the protected database is directly transported and stored on the Recovery Appliance. This reduces the window of potential data loss that exists between successive archived log backups.Configuring real-time redo transport for a protected database is a one-time step. After you set it up, the protected database asynchronously transports redo data to the Recovery Appliance.
  To enable real-time redo transport for a protected database
  - Ensure that the Recovery Appliance user that the protected database uses to send backups to the Recovery Appliance is configured. This same user will be used for redo transport. Also ensure that an Oracle wallet is created on the protected database that contains credentials for the Recovery Appliance (and redo transport) user
  - Ensure that the following conditions are met for the protected database:
    - ARCHIVELOG mode is enabled
    - DB_UNIQUE_NAME parameter is set
  - Ensure that, following initialization parameters are set for the protected database:

    ```
    /home/oracle> sqlplus “/as sysdba”
    SQL> ALTER SYSTEM SET REMOTE_LOGIN_PASSWORDFILE=exclusive SCOPE=BOTH;
    SQL> ALTER SYSTEM SET LOG_ARCHIVE_FORMAT='log_%d_%t_%s_%r.arc' SCOPE=BOTH;
    ```

    Note, REMOTE_LOGIN_PASSWORDFILE can be set to exclusive or shared

    ```
    SQL> ALTER SYSTEM SET DB_UNIQUE_NAME=<db_unique_name> SCOPE=BOTH;
    ```

```
    SQL> ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(<db_unique_name_of_ZDLRA>,
    <db_unique_name_of_projected database>)' SCOPE=BOTH;
    ```
    For example,
    ```
    SQL> ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(zdlra, dbzdlra)' SCOPE=BOTH;
    ```
    The DB_NAME and the DB_UNIQUE_NAME of the Recovery Appliance database is zdlra.
    The DB_NAME and the DB_UNIQUE_NAME of the protected database is dbzdlra.
  - Configure and enable an archived log destination that points to the redo staging area on the Recovery Appliance
    The following example configures the protected database to transport redo data synchronously to a Recovery Appliance whose net service name is boston. For example,
    ```
    SQL> ALTER SYSTEM SET LOG_ARCHIVE_DEST_3='SERVICE=boston VALID_FOR=(ALL_LOGFILES,
    ALL_ROLES) ASYNC DB_UNIQUE_NAME=zdlra' SCOPE=BOTH;
    SQL>ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_3='ENABLE' SCOPE=BOTH;
    ```
  - Set the redo transport user to the Recovery Appliance user that was created forthis protected database
    The following example sets the redo transport user to zdlravpc:
    ```
    SQL> ALTER SYSTEM SET REDO_TRANSPORT_USER= zdlravpc  SCOPE=BOTH;
    ```
  - Shut down the protected database and restart it
    Note: Recommendation is use server parameter file for the protected database. If the protected database uses a parameter file instead of a server parameter file,then add the parameters that were set in above steps to the parameter file before you start up the protected database.
- Performing Test Backup and Recovery Operations
  After you enroll the protected database with a Recovery Appliance, it is recommendedthat you perform a test backup and recovery operation. This testing helps confirm thatyour configuration settings are accurate and that the backup to and recovery from theRecovery Appliance are performed successfully. If you encounter any problem with thetest backup or recovery, you may correct your settings and reconfigure your protecteddatabase
  - Run a Test Backup
    To create a test backup of the protected database
    - Connect to the protected database as TARGET and to the Recovery Appliance catalog as CATALOG
      For example,
      ```
      /home/oracle> rman target / catalog zdlravpc/*****@zdlra-scan.example.com:1521/zdlra:dedicated
      ```
    - Configure an RMAN SBT channel for the Recovery Appliance
      A good guideline for choosing the number of channels is to start with the numberof channels that are currently used for incremental backups or a default of 2 or 4channels per node depending on the number of cores or CPUs.
      ```
      RMAN> CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE' PARMS
      'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/
      u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-
      scan.example.com:1521/zdlra:dedicated)' FORMAT '%U_%d';
      ```
    - Use the following RMAN command to perform a full backup
      ```
      RMAN>  BACKUP DEVICE TYPE SBT CUMULATIVE INCREMENTAL LEVEL 1 DATABASE PLUS
      ARCHIVELOG;
      ```
  - Run a Test Recovery
    After creating a test backup of the protected database to Recovery Appliance, you cantest this backup by performing a test recovery.
    To perform a test recovery of the protected database
    - Shutdown and restart the protected database in NOMOUNT mode
    - Connect to the protected database as TARGET and to the Recovery Appliance catalog as CATALOG
      For example,
      ```
      /home/oracle> rman target / catalog zdlravpc/*****@zdlra-scan.example.com:1521/zdlra:dedicated
      ```
    - Use the following RMAN command to perform a full backup
      ```
      RMAN>  BACKUP DEVICE TYPE SBT CUMULATIVE INCREMENTAL LEVEL 1 DATABASE PLUS
      ```
    ```
    ARCHIVELOG;
    ```
    - Use the following RMAN command to restore the previously created test backupfrom the Recovery Appliance. Because the VALIDATE option is used, this can bedone without interfering with the production database.
      For example,
      ```
      /home/oracle> rman target / catalog zdlravpc/*****@zdlra-scan.example.com:1521/zdlra:dedicated
      RMAN> run
      ```

```
{
allocate channel c1 DEVICE TYPE 'SBT_TAPE' PARMS
'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/
u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-
scan.example.com:1521/zdlra:dedicated:dedicated)' FORMAT   '%U_%d';
RESTORE VALIDATE DATABASE;
}
```

If these backup and recovery procedures succeed, then the client (protected) database is ready to perform regular backups to the Recovery Appliance

3. Perform backup / restore and recovery using ZDLRA
   After you configure the protected database, using standard RMAN backup, restore, and recovery commands you can create, schedule protected database backups and perform restore and recovery operations. Recovery Appliance uses the incremental-forever backup strategy for protected database backups. In this strategy, an initial level 0 incremental backupis followed by successive level 1 incremental backups. Refer to Appendix D for sample commands.
4. To determine network throughput for a specific time period, use RMAN network analyzer, see MOS note 2022086.1

BACKUP AND RECOVERY USING TAPE DEVICES

You may choose to store your database backups on tape devices. Some of the key benefits of a tape-based backup strategy include:

- The Oracle Database Appliance and tape-based backups provide fast backup and restore rates
- Tape-only solutions isolate faults from the Oracle Database Appliance
- Oracle Database Appliance storage capacity and network bandwidth are maximized

For a tape-based backup solution, the recommended strategy is as follows:

- Weekly RMAN level 0 (full) backups of the database
- Daily cumulative RMAN incremental level 1 backups of the database
- Daily backups of the Oracle Secure Backup catalog

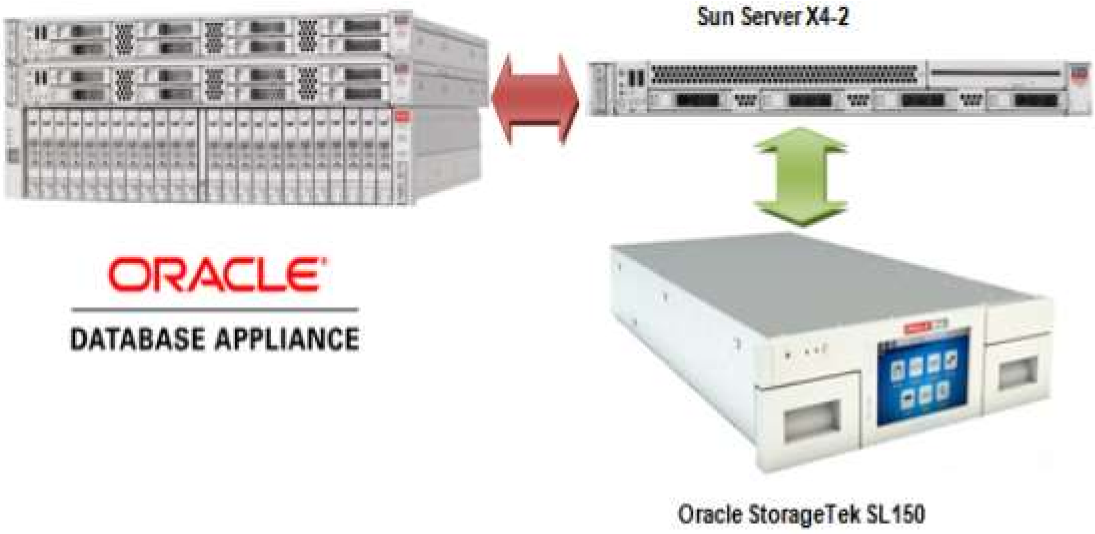


FIGURE 2: ORACLE DATABASE APPLIANCE AND TAPE DEVICES ARCHITECTURE

MEDIA MANAGEMENT SOFTWARE FOR TAPE BACKUPS

In order to perform backups to tape, RMAN is integrated with a media management software, Media management software is the software layer that facilitates RMAN backups to tape. Oracle Secure Backup is the media management software used during the course of the writing of this white paper. It is a highly scalable backup solution with a client/server architecture in which all hosts in the backup domain are centrally managed using a single console and a common management interface across multiple servers and NAS devices. For more information about OSB see the Oracle Secure Backup Documentation

The tape backup performance numbers reported in this white paper were achieved using a single OSB Administrative/Media server with a dedicated 1 Gb active-passive bonded network connected to an Oracle Database Appliance system and the 10 Gb active-passive bonded network connected an Oracle StorageTek SL150 containing two LTO-6 tape drives attached via a 6 Gb/sec SAS connection to the OSB Media Server.

- Target database has 1 TB of data with a data compression of approximately 1.4 to 1. Depending on the composition of the data, compression will vary and so will transfer rates to the tape drive
- There were minimal archive logs to backup and the database was mostly idle during backup. If a significant number of archive logs are present, then it will impact backup times as backing up a large number of small files slows performance. Additionally, if there is heavy load on the database and CPU is fully consumed, backup rates could be affected.
- The performance of the 1 Gb backup was limited by the single active-passive bond on the OSB Media Server. Each node on the Oracle Database Appliance has multiple 1 Gb bonded interfaces. Through any configured interface it can send 120 MB/sec for an aggregate transfer rate of 240 MB/sec, but since there is only one bonded NIC on the Media Server, it can consume a maximum of 120 MB/sec which limits the transfer to 120 MB/sec.
- The performance of the 10 Gb backup was limited by the two LTO-6 tape drives. Average rates were 373 MB/sec per tape drive and additional tape drives could be added up to 1 GB/sec assuming the Database Appliance I/O bandwidth is not exhausted first (single active-passive bond on OSB Media Server).
- Restore tests consisted of restoring the control file and data files from tape, but the recovery operation retrieved archive logs from the local Fast Recovery Area (FRA).
- Backup rates assume tape drives are mounted before the job starts and rates are calculated on data transfer time utilizing OSB recorded start/stop times.

DESCRIPTION	BACKUP RATE (TB/HR)	HOST CPU USAGE
1 Gb Load Balanced Across Both Nodes 2 Cores	0.43TB	2 – 6 %
1 Gb Load Balanced Across Both Nodes 24 Cores	0.43TB	0.5 – 3 %
1 Gb Single Node 2 Cores	0.43TB	2 – 6 %
10 Gb Load Balanced Across Both Nodes 2 Cores	2.7TB	2 – 6 %
10 Gb Load Balanced Across Both Nodes 24 Cores	2.7TB	0.5 – 3 %
10 GB Single Node 2 Cores	1.0TB	2 – 6 %

TABLE 4: TAPE BACKUP PERFORMANCE OBSERVATIONS

DISK BACKUPS

Depending on your backup and recovery requirements and resource availability you may choose to use disk-based backups. You may also choose to use disk-based backups if you require Tablespace Point-in-Time Recovery (TSPITR), switching to a backup copy, or perform incremental merges, as these options are not available with tape-based backups. When you perform disk-based backups on Oracle Database Appliance, the backups are stored in the Fast Recovery Area (FRA) located in the RECO disk group.

Some of the key benefits of a disk-based backup strategy include:

- Faster recovery times during data and logical corruptions
- Ability to perform Tablespace Point-in-Time Recovery (TSPITR)
- Ability to use backups directly with no restore by switching to a copy of the database, tablespace or data file. For disk-based backup solutions, Oracle recommends the following:
    - Use a Fast Recovery Area (FRA)
    - Perform an initial RMAN level 0 (full) backup

  - Perform daily RMAN incremental level 1 backups
  - Roll incremental backups into full backup and delay by 24 hours (see RMAN discussion for details)

RMAN BACKUPS TO LOCAL DISKS

On Oracle Database Appliance Fast Recovery Area (FRA) is created on the RECO ASM diskgroup. Together the Oracle database and RMAN manage the space inside this area, keep track of and manage backups, including deleting old unneeded backups. Oracle RMAN backs up image copies, archived logs, control files, and flashback logs to the FRA. When new backups demand more room, Oracle automatically removes the nonessential backups, freeing the DBA from this chore. The files in the FRA are considered nonessential when they become obsolete according to the backup retention policy, or when they have already been backed up to tape with Oracle RMAN.

RMAN BACKUPS TO EXTERNAL STORAGE

External storage on Network Attached Storage can be made available on Oracle Database Appliance using NFS mounts. This may be useful if you choose to store backups on external storage (or tapes) and want to a larger sized DATA diskgroup. External Backup Type specified as the Backup Type during Oracle Database Appliance deployment allows you to allocate more storage to the DATA diskgroup.

PERFORMANCE OBSERVATIONS

RMAN allows for parallel processing of backup workloads in Oracle Real Application Clusters (RAC) environments. Use the following general guidelines to maximize backup transfer rates to local storage:

- Use both instances and start with one RMAN channel per instance
- Add additional RMAN channels to reach an optimum level.

Optimal backup rates were observed when utilizing both RAC instances and one to four RMAN channels per instance, depending on the system configuration.

Backup operation	Backup Rate (TB/hr)	Restore Rate (TB/hr)	Host Busy (CPU)
Image copy 8 Cores 4 channels	2.2TB	2.35TB	28 – 38 %
Image Copy 32 Cores 8 channels	2.25TB	2.40TB	10 – 20 %

TABLE 5: BACKUP AND RESTORE PERFORMANCE OBSERVATIONS IN A RAC DATABASE ENVIRONMENT

For single instance database deployments, the following performance observations were made.

Backup operation	Backup Rate (TB/hr)	Restore Rate (TB/hr)	Host Busy (CPU)
Image copy 2 Cores 2 channels	1.92TB	2.05TB	50 – 80 %
Image copy 4 Cores 2 channels	1.99TB	2.10TB	25 – 38 %
Image Copy 16 Cores 4 channels	2.16TB	2.24TB	15 – 20 %

TABLE 6: BACKUP AND RESTORE PERFORMANCE OBSERVATIONS IN A SINGLE INSTANCE DATABASE ENVIRONMENT

The performance numbers shown in these tables are for Bare Metal installations, except for the single instance with 2 cores configuration. The same tests performed in a virtualized environment showed backup rates that were 2-5% lower and the hosts were slightly busier.

RMAN Backup sets were created and restored with similar CPU usage. Compression required more CPU depending on the compression algorithm chosen.

BACKUP AND RECOVERY WITH NETWORK FILE SYSTEM (NFS) STORAGE

The Oracle ZFS Appliance 7000 is a unified storage system that provides flexible configuration and attachment options for a wide range of storage demands. The Oracle ZFS 7120 was selected to demonstrate the ability to send RMAN backups over the 10 Gb interfaces on the Oracle Database Appliance to network storage using the Oracle-exclusive dNFS high performance NFS client.

Using network-attached storage for database backups allows isolation of backups from the Oracle Database Appliance internal storage, and opens a range of possibilities for management of the backups including replication to a remote site, snapshots for additional copies of backups, compression of backups by the ZFS Appliance, and sharing of the backups with another database server.

The methodology for network-attached storage is similar to FRA-based backups:

- Use NFS shares and define the NFS Appliance shares to dNFS so that the dNFS client is used
- Perform an initial RMAN level 0 (full) backup
- Perform daily RMAN incremental level 1 backups
- Roll incremental backups into full backup and delay by 24 hours (see see RMAN product documentation (Choose database version (19c|18c|12c|11gR2) =>Database Administration=> Backup and Recovery))

RMAN BACKUPS TO THE ORACLE ZFS STORAGE APPLIANCE 7120

The ZFSSA 7120 is a single-head storage controller with capacity and performance that matches well with the Database Appliance. It can be configured from 3.3 TB to 177 TB of raw capacity and 73 GB of write-optimized flash storage that can be accessed using 1 Gb, 10 Gb, or fiber channel interfaces, using a wide variety of protocols. For use as a Database Appliance backup target, NFS shares accessed over 10 Gb interfaces are recommended. The 7120 comes standard with 4 x 1 Gb network interfaces. For optimal backup performance, the optional 10 Gb interfaces are recommended for the 7120.

The ZFSSA architecture provides flexible configuration options. For this white paper, we chose a configuration that optimizes the RMAN large block, streaming write and read performance over Ethernet interfaces, while maintaining fault-tolerance. Defining NFS shares in a single double parity (RAID-Z2) storage pool provides the necessary performance and availability. We will assume the ZFSSA 7120 has an optional dual 10 Gb network interface card, two dedicated boot drives, and twelve 2 TB or 3 TB HDD for data storage.

The ZFS Storage Appliance 7120 can be configured using the web-based Browser User Interface (BUI) or via CLI commands executed directly on the ZFS Appliance. In all examples below it is assumed the user has logged into the BUI using the root user and password. The usual form of the BUI URL is:

https://<ZFSSA Name or IP Address>:215

The complete documentation for the configuration of all aspects of the Storage Appliance is available in the BUI help screens

- Pools - The ZFS Storage Appliance 7120 stores data in groups of hard disks aggregated into pools. There are several possible pool configurations: Single, double or triple parity and mirrored or striped. Given the emphasis with Oracle Database Appliance on maximum data availability and good performance, choosing double parity (RAID-Z2) is the best balance between performance and availability. While multiple pools can be configured on a dedicated 7120 the best choice is to define a single storage pool
  - Click Configuration->Storage
  - Click the plus sign (+) next to Available Pools
  - Give the pool a name (Pool-0 for example) and click Apply
  - At the “Verify and add devices” screen select all HDDs but do not select the Boot drives
  - Click “Commit”
  - On the next screen choose the Double parity storage profile
  - Click “Commit”
- Shares - The ZFS Storage Appliance 7120 supports NFS, CIFS/SMB and iSCSI network storage protocols, as well as Fiber Channel with an optional interface card. The Oracle Database Appliance has the ability to run a highly-optimized version of the NFS file system client called dNFS, so defining and using NFS shares as targets for Oracle Database Appliance backup is a natural choice. NFS shares can be defined with several options, and for targets for Oracle Database Appliance backup, these are recommended:
  - Database record size: 128 KB
  - Synchronous write bias: Throughput
  - Data Compression: Off for best performance, LZJB for good inflight compression
    The number of NFS shares to define for Oracle Database Appliance backup depends on the number of services and RMAN channels defined to execute the RMAN backups. Generally, one NFS share per RMAN channel provides optimum throughput. As with the FRA based backup configurations, two RMAN channels per server are a good starting point. For a RAC

configuration, a total of four RMAN channels and four shares work well. NFS shares belong to a Project on the appliance, so first we define a Project, then the shares owned by the Project

- In the BUI:
    - Click Shares->Projects
    - In the Projects pane on the left side, click the plus sign (+) next to the word "All"
    - Enter a name for the project and click Apply
    - Click on the new Project name in the Projects pane, then click General
    - Change Synchronous Write Bias to Throughput, Database record size to 128K, and set Data compression to Off or LZJB as desired.
    - Adjust the default permissions for the shares in the project
    - Click Apply – You now have a Project for your ODA backup shares
    - The Filesystems pane will appear, click the plus sign (+) next to the word "Filesystems"
    - Provide a share name
    - Adjust the default permissions given to the share if necessary
    - Click Apply
    - Create three more shares
    - Note the export mount point name shown in the Properties page of each share
- Network Configuration
  We will assume a 7120 configuration with the optional 2 x 10 Gb interface card. The ports can be used independently or can be bound together using the Link Aggregation Protocol (LACP) or IP Multi-Pathing (IPMP). In general, LACP is used for improved performance, while IPMP is used for availability. LACP requires a switch that can use the LACP techniques to load balance between physical ports, while IPMP does not require special switch configuration. Alternatively, the 10 Gb ports on the ZFSSA 7120 can be directly connected, one port to each server on the Oracle Database Appliance, without a switch, using a 192.168.* private non-routable network domain between the Oracle Database Appliance and the 7120. Jumbo frames should be specified.
- Mounting the Shares on the ODA and Configuring dNFS
- The /etc/fstab file on each server should be modified on each ODA server to mount each share created on the ZFSSA 7120 on mount points created on each server
  ```
  mkdir /mnt/backup1 /mnt/backup2 /mnt/backup3 /mnt/backup4
  ```
- Edit /etc/fstab to include an entry for each mount point.
  For example:
  ```
  192.168.2.1:/export/ODA/backup1 on /mnt/backup1 type nfs
  (rw,bg,hard,nointr,rsize=1048576,wsize=1048576,tcp,nfsver=3,timeo=600)
  ```
- Issue the command 'mount –a' to read fstab
- Adjust ownership and permissions, if desired (using chown/chmod commands)
  The Oracle database has a special NFS client called Direct NFS or dNFS. The I/O throughput from an Oracle database to an NFS share is greatly increased if dNFS is used.
- A summary of how to configure dNFS follows:
    - Shut down the Oracle database instance(s) on each server
    - Issue this command from the oracle user on each server:
      ```
      $ make –f $ORACLE_HOME/rdbms/lib/ins_rdbms.mk dnfs_on
      ```
    - Create a file called $ORACLE_HOME/dbs/oranfstab on each server with entries showing the shares defined on the 7120 Appliance
      ```
      server: zfs-server
      path:  192.168.2.1
      export /export/ODA/backup1  mount:  /mnt/backup1
      export /export/ODA/backup2  mount:  /mnt/backup2
      export /export/ODA/backup3  mount:  /mnt/backup3
      export /export/ODA/backup4  mount:  /mnt/backup4
      ```
    - Restart the Oracle database instance(s) on each server
      When executing RMAN, the following SQL queries can verify the use of dNFS:
      ```
      select * from v$dnfs_servers;
      select * from v$dnfs_files;
      ```
      You may also want to review the database alert log and check database startup messages.
- Configuring RMAN to Use the ZFSSA 7120
  In order to efficiently allocate resources across the database nodes during backups, the backup load should be spread evenly between the RAC nodes.
    - Create one service for each RMAN Channel/NFS mount point to run on selected nodes in the cluster:

```
$ srvctl add service –d <dbname> -s <service name1> -r <instance1>  -a<instance2>
$ srvctl add service –d <dbname> -s <service name2> -r <instance2> -a <instance1>
```

For example:

```
$ srvctl add service –d isr –s isrsvc1 –r isr1 –a isr2
$ srvctl add service –d isr –s isrsvc2 –r isr2 –a isr1
```

- Start the services:

  ```
  $ srvctl start service –d <db_unique_name> -s <service_name1>
  $ srvctl start service –d <db_unique_name> -s <service_name2>
  ```

  For example:

  ```
  srvctl start service –d isr –s isrsvc1
  srvctl start service –d isr –s isrsvc2
  ```

  The database backup and recovery strategies when using the ZFSSA 7120 as the target are similar to RMAN commands backing up to the local FRA. The ALLOCATE CHANNEL commands in the RMAN run block need to target the NFS mount points created on the 7120, and they need to connect to the services created to write to each mount. In the example, service isrsvc1 will write to /mnt/backup1 and service isrsvc2 will write to /mnt/backup2. If each service is running on a different server, the resources of both servers will be used to create the RMAN backup set.

- For example:

  ```
  RUN
  {
  allocate channel oem_backup_disk1 type disk format '/mnt/backup1/%U' connect ’@isrsvc1’;
  allocate channel oem_backup_disk2 type disk format '/mnt/backup2/%U' connect ’@isrsvc2’;
  allocate channel oem_backup_disk3 type disk format '/mnt/backup3/%U' connect ’@isrsvc3’;
  allocate channel oem_backup_disk4 type disk format '/mnt/backup4/%U' connect ’@isrsvc4’;
  backup as BACKUPSET tag '%TAG' database;
  backup as BACKUPSET tag '%TAG' archivelog all not backed up;
  release channel oem_backup_disk1;
  release channel oem_backup_disk2;
  release channel oem_backup_disk3;
  release channel oem_backup_disk4;
  }
  ```

PERFORMANCE NUMBERS FOR ZFSSA-BASED BACKUP CONFIGURATIONS

To scale backup rates for disk on Oracle Database Appliance using a RAC configuration:

Use both instances and start with two RMAN channels per instance

Dedicate an NFS mount point to each channel

Continue to add additional RMAN channels for performance per instance

On the Oracle Database Appliance with a 12 TB 7120 configuration, optimal backup rates were achieved with both RAC instances and two RMAN channels per instance.

Configuration	Backup Rate (TB/hr)	Restore Rate (TB/hr)
10 Gb Load Balanced Across Both Nodes	1.2TB	0.6TB

TABLE 7: BACKUP AND RESTORE PERFORMANCE OBSERVATIONS ON ZFS STORAGE APPLIANCE

ENGINEERED SYSTEMS BACKUP UTILITY

RMAN backups of the ODA to an Oracle ZFS Storage Appliance can be configured automatically using the Engineered Systems Backup Utility 2.0 (ESBU), a free utility available on OTN. The ESBU 2.0 User's Guide can guide the user through the setup of the utility. This document illustrates an alternative method of configuring backups to the ZFS Appliance using manual interfaces.

Restore and Recover the database

Use standard RMAN procedures for restore and recovery of the instances for all these scalable destinations. Refer section 5.5 for sample steps (exclude oracle cloud specific configuration steps)

BACKUP AND RECOVERY FOR ORACLE DATABASE APPLIANCE S|M|L

The Oracle Database Appliance S|M|L are single node configurations. Thus, these configurations do not provide high availability that Oracle Database Appliance HA configurations provide. If an Oracle Database Appliance S|M|L server becomes unrecoverable, in most cases, you must re-image, re-deploy, restore, and recover your system from backups.

When re-imaging the system, refer the My Oracle Support note titled "Oracle Database Appliance X6-2S, X6-2M and X6-2 L (Doc ID 2144642.1", to identify and download ISO image for the Operating System. You can then re-image the Oracle Database Appliance S|M|L server using the ISO Image and redeploy the software using the standard process, which includes setting up the operating system, and installing Grid Infrastructure and RDBMS software. Once the system has been redeployed, you can restore and recover the database(s) from your backups using standard RMAN procedures.

DATABASE BACKUP AND RECOVERY BEST PRACTICES

This section outlines some of the core best practices when establishing your backup and recovery configuration in an Oracle Database Appliance environment.

1. Choose backup location based on RTO/RPO requirements
   During the Oracle Database Appliance deployment process, you are required to choose backup location and storage mirroring. These choices determine storage allocation to the DATA and RECO disk groups. The placement of backups on local storage has direct bearing on backup and recovery processes and time requirements. However, local storage on Oracle Database Appliance is premium storage and limited to a maximum capacity and configuration. Choosing the backup location and mirroring options appropriately should allow you to meet your requirements and objectives.
2. Use "weekly full and daily incremental" backup strategy
   Incremental backups allow you to back up only those data blocks that have changed since a previous backup. Incremental backups are thus efficient in terms of time and space requirements. The RMAN block change tracking feature for incremental backups improves incremental backup performance by recording changed blocks in each data file in a change tracking file. If change tracking is enabled, RMAN uses the change tracking file to identify changed blocks for incremental backup, thus avoiding the need to scan every block in the data file at backup time.
   To enable or disable block change tracking refer to the example below. Additional information can also be found in the RMAN Incremental Backup documentation.
   ```
   SQL>ALTER DATABASE ENABLE BLOCK CHANGE TRACKING;
   SQL>ALTER DATABASE DISABLE BLOCK CHANGE TRACKING;
   ```
   However, prior to you should evaluate if your RTO requirements can still be met if you choose to use this approach. Incremental backup typically require substantially less time to execute, giving you the option to backup more frequently and reduce RTO/RPO. By doing incremental backups, you also reduce network usage and network bandwidth requirements when backing up over a network. Further, incremental database backups reduce backup overhead and read I/O volume on the database.
3. Schedule archived log backup more frequently to reduce RPO
   The archived redo logs residing on the system are vulnerable to loss in case of a complete system failure that renders the whole system in an unrecoverable state. For this reason, archived redo logs are backed up to a separate external (often remote) location. Choose a frequency of archived redo log backups that meets your requirements. Many customers use a standby system to transfer redo data to the remote location to ensure minimal redo loss in case of a complete system failure.
4. Validate backups
   Perform RMAN CROSSCHECK operation on the backups to ensure validate backups.
   ```
   RMAN> CROSSCHECK BACKUP;
   ```
5. Validate backups regularly to check for physical and logical corruptions
   After a backup operation, use the RMAN BACKUP VALIDATE command to check the data files for physical corruptions. To check for logical corruptions, include the CHECK LOGICAL clause in the BACKUP VALIDATE command.
   ```
   RMAN> BACKUP VALIDATE CHECK LOGICAL DATABASE ARCHIVELOG ALL;
   ```
6. Perform restore validate weekly
   Use the RMAN RESTORE VALIDATE command to check and verify the integrity of the backups. RESTORE DATABASE VALIDATE command only checks for the datafile backups and not ARCHIVELOG or CONTROLFILE backups. Issue RESTORE

ARCHIVELOG VALIDATE and RESTORE CONTROLFILE VALIDATE commands for the latter. Use RESTORE SPFILE VALIDATE command to check server parameter file backup. For more information, please see Oracle Database documentation. RMAN restore validate command does a block level check of the backups and verifies all needed database files are available, thus ensuring that an actual restore can be performed. It is recommended to validate backups on a regular basis.

```
RMAN> RESTORE DATABASE VALIDATE CHECK LOGICAL;
```

For large backup sets, restore validate command can take longer to complete. For a quick validation to ensure the backup files are available you can leverage restore validate header. This will validate that backups are present but will not perform block-level check.

```
RMAN>RESTORE DATABASE VALIDATE HEADER;
```

7. Quarterly full restore
   Test a full restore of the database in a test environment on a quarterly basis to ensure backups can be reliably used if needed.
8. Use Fast Connect when using backups to Oracle Cloud
   If storing backups in Oracle Cloud, use the Fast Connect facility to leverage greater bandwidth and lower latency and perform backups most efficiently.
9. Update RMAN SBT Module
   If using the RMAN SBT module, update it periodically to ensure you are using a more current version and avoid known issues that may have surfaced and may have been fixed in a latter version

CONCLUSION

Oracle Database Appliance benefits from native Oracle database integration with Oracle Recovery Manager (RMAN). You can choose from a variety of backup destinations depending on your requirements. When Oracle Database Appliance is deployed with the best practices outlined in this white paper, the backup, restore, and recovery operations for your Oracle Databases can be optimized.

Backup placement for Oracle Databases running on Oracle Database Appliance can be either on local storage or external storage. Local backups are placed in the RECO disk group on Oracle Database Appliance storage while external backups can be placed on NFS storage, tape storage, or in the Oracle Cloud. Oracle ZFS Storage Appliance and Oracle StorageTek SL150 Tape device provide a unique value proposition in terms of performance and high availability for hosting external database backups for databases running on Oracle Database Appliance. Oracle Cloud presents a unique opportunity to store backup securely and cost effectively in an offsite location.

During testing, the peak backup performance was observed to be 1.2TB per hour for ZFS Storage Appliance and over 2.7TB per hour for Oracle StorageTek SL150. Backup placed on local Oracle Database Appliance storage executed at the rate of 2.5TB per hour.

Oracle Cloud Database Backup Service offers an effective and low cost alternative to protect your Oracle Appliance databases while at the same time securing your backups in a remote location.

APPENDIX A CONFIGURING LOAD-BALANCED BACKUPS

In order to efficiently allocate resources across the database nodes during backups, the backup load should be spread evenly between the RAC nodes.

Create a service that runs on the selected nodes in the cluster.

```
$ srvctl add service –d <dbname> -s <service name> -r <instance1>,<instance2>

$ srvctl add service –d isr –s isrsvc –r isr1,isr2
```

Start the service

```
$ srvctl start service –d <db_unique_name> -s <service_name>

$ srvctl start service
```

Add a net service name to $ORACLE_HOME/network/admin/tnsnames.ora, which is used for automatic load balancing the connection:

```
ISR =
  (DESCRIPTION =
       (ADDRESS = (PROTOCOL = TCP)(HOST = hamms-scan)(PORT = 1521))
       (CONNECT_DATA =
            (SERVER = DEDICATED)
            (SERVICE_NAME = isr)
        )
    )
```

For specific node connectivity, use net names as shown here

```
ISR1 =
 (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)(HOST = hamms1)(PORT = 1521))
      (CONNECT_DATA =
          (SERVER = DEDICATED)
          (SERVICE_NAME = isr)
          (SID = isr1)
       )
    )
ISR2 =
  (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)(HOST = hamms2)(PORT = 1521))
      (CONNECT_DATA =
            (SERVER = DEDICATED)
            (SERVICE_NAME = isr)
            (SID = isr2)
        )
   )
```

APPENDIX B: SAMPLE SCRIPTS

For all scripts in this section archive logs needed for recovery are available on disk. The scripts do not cover special considerations that may arise when restoring a production database. Customers may use these examples, adjust them to their needs and embed them in shell scripts.

Tape Backup in RAC environments

The script allocates two channels because we have tested with two tape drives and creates a full backup including the archive logs.

```
RUN
 {
   ALLOCATE CHANNEL ch00 TYPE 'SBT_TAPE' CONNECT='@isr';
   ALLOCATE CHANNEL ch01 TYPE 'SBT_TAPE' CONNECT='@isr';
   BACKUP INCREMENTAL LEVEL 0 DATABASE PLUS ARCHIVELOG;
 }
```

Note: Channels are load balanced in RAC

Tape Restore for Single Instance and RAC One Node

For the restore two channels are allocated well and the database is recovered automatically with the available archive logs. Sometimes even the old redo logs were available so that the database could be recovered without open resetlogs.

```
ALTER DATABASE MOUNT
RUN
{
  ALLOCATE CHANNEL ch00 TYPE 'SBT_TAPE';
  ALLOCATE CHANNEL ch01 TYPE 'SBT_TAPE' ;
  RESTORE DATABASE; RECOVER DATABASE;
}
ALTER DATABASE OPEN RESETLOGS;
```

Note: To run parallel restores you must mount the database on the second node and allocate channels using connect strings.

Image copy backup (RAC, RAC One Node and Single Instance)

Before executing the backup as copy operation the configuration details like backup type and parallelism are set.

```
CONFIGURE DEFAULT DEVICE TYPE TO DISK;
CONFIGURE SNAPSHOT CONTROLFILE NAME TO '+RECO/ISR/snap.cf';
CONFIGURE DEVICE TYPE DISK BACKUP TYPE TO COPY;
CONFIGURE DEVICE TYPE disk PARALLELISM 2;
CONFIGURE CONTROLFILE AUTOBACKUP ON;
RUN
{
  BACKUP AS COPY DATABASE;
}
```

Image copy restores on RAC

The channel allocations use the credentials of the user connected to the instance.

```
RUN
{
```

```
ALLOCATE CHANNEL ch1 DEVICE TYPE DISK CONNECT ‘@isr1';
ALLOCATE CHANNEL ch2 DEVICE TYPE DISK CONNECT ‘@isr2';
RESTORE DATABASE;
RECOVER DATABASE;
}
startup;
```

Note: To run parallel restores you must mount the database on the second node and allocate channels using connect strings

Image restore RAC One Node and Single Instance

Restore can also be parallelized and speed up performance.

```
CONFIGURE DEVICE TYPE disk PARALLELISM 2;
RUN
{
RESTORE DATABASE;
RECOVER DATABASE;
}
startup;
```

Backup script for backup set

The configure command sets the backup type for the backup operation.

```
CONFIGURE DEFAULT DEVICE TYPE DISK;
CONFIGURE DEVICE TYPE DISK BACKUP TYPE TO BACKUPSET;
CONFIGURE SNAPSHOT CONTROLFILE NAME TO '+RECO/ISR/snap.cf';
 CONFIGURE DEVICE TYPE disk PARALLELISM 2;
CONFIGURE CONTROLFILE AUTOBACKUP ON;
RUN
{
 BACKUP DATABASE;
}
```

Monitoring disk based backups

When an RMAN job is executed the job transcript is written to stdout by default, but the output can be redirected to a log file that can be analyzed for errors and warnings, as well as to review backup piece names that are written. Additionally, RMAN uses the NLS_DATE_FORMAT environment variable to report times in hours / minutes and seconds, that can be useful to monitor run times.

```
SELECT sid, serial#, context, sofar, totalwork,  round(sofar/totalwork*100,2) "% Complete"   FROM v$session_longops  WHERE
opname LIKE 'RMAN%'   AND opname NOT LIKE '%aggregate%'     AND totalwork != 0  AND sofar <> totalwork        /
```

Use below queries to check RMAN restore progress:

```
SQL> SELECT operation,OBJECT_TYPE,status, mbytes_processed, start_time, end_time FROM v$rman_status  order  by end_time;
SQL> SELECT   operation, status, mbytes_processed, start_time, end_time FROM   v$rman_status where  status ='RUNNING';
SQL> select SID,STAMP,COMMAND_ID,OUTPUT_DEVICE_TYPE,OBJECT_TYPE from   v$rman_status where  status ='RUNNING';
```

```
SQL> select sid,start_time,totalwork, sofar, (sofar/totalwork) * 100 pct_done from v$session_longops where totalwork > sofar AND opname NOT LIKE '%aggregate%' AND  opname like 'RMAN%';
```

APPENDIX C SAMPLE COMMANDS FOR BACKUP AND RECOVERY IN CLOUD

- Backup, restore and recovery using Password Encryption
    - Backing Up (Level 0) to Oracle Database Backup Cloud Service

      ```
      RMAN> SET ENCRYPTION ON IDENTIFIED BY '<Specify password>’ ONLY;
      RMAN> BACKUP DEVICE TYPE SBT AS COMPRESSED BACKUPSET DATABASE PLUS ARCHIVELOG FORMAT '%d_%U';
      ```

      For example

      ```
      RMAN> SET ENCRYPTION ON IDENTIFIED BY 'odabackup123' ONLY;
      RMAN> BACKUP DEVICE TYPE SBT AS COMPRESSED BACKUPSET DATABASE PLUS ARCHIVELOG FORMAT '%d_%U';
      ```

    - Backing Up (Level 1) to Oracle Database Backup Cloud Service

      ```
      RMAN> SET ENCRYPTION ON IDENTIFIED BY 'odabackup123' ONLY;
      RMAN> BACKUP DEVICE TYPE SBT AS COMPRESSED BACKUPSET INCREMENTAL LEVEL 1 DATABASE;
      ```

    - Backup Validation

      ```
      /home/oracle> rman target /
      RMAN> SET DECRYPTION IDENTIFIED BY '<Specify password>’;
      #Get the password from section 4.8.1
      RMAN> RESTORE DATABASE VALIDATE CHECK LOGICAL;
      ```

      For example

      ```
      /home/oracle> rman target /
      RMAN> SET DECRYPTION IDENTIFIED BY 'odabackup123';
      RMAN> RESTORE DATABASE VALIDATE CHECK LOGICAL;
      ```

    - Restore and recover the database

      Connect to Recovery Manager (RMAN), set the decryption password, set the DBID, and restore the SPFILE

      ```
      /home/oracle> rman target /
      RMAN> STARTUP NOMOUNT;
      RMAN> SET DECRYPTION IDENTIFIED BY 'odabackup123';
      RMAN> set DBID=2985052152 #Source database DBID
      RMAN> RUN
      {
      ALLOCATE CHANNEL t1 DEVICE TYPE sbt PARMS 'SBT_LIBRARY=libopc.so ENV=(OPC_FILE=/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/opcodadb1.ora)';
      RESTORE SPFILE TO PFILE '/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/initdboda1.ora' FROM AUTOBACKUP;
      }
      ```

Edit the PFILE to reflect the restore server, changing control file locations (control_files), create (db_create_file_dest), recovery (db_recovery_file_dest) file destinations, audit (audit_file_dest) file destinations, archive log location and db_unique_name (if ORACLE_SID is changing). For example, change the *_dest parameters, so all destinations are correct, change the control_files parameter, and so on. If necessary, create the relevant directories on the restore server.

Restore the control file

```
RMAN> SHUTDOWN IMMEDIATE;

RMAN> STARTUP NOMOUNT;

RMAN> RUN

{

ALLOCATE CHANNEL t1 DEVICE TYPE sbt PARMS 'SBT_LIBRARY=libopc.so
ENV=(OPC_FILE=/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/opcdboda1.ora)';

RESTORE CONTROLFILE FROM AUTOBACKUP;

}
```

Mount the database

```
RMAN> ALTER DATABASE MOUNT;
```

Start the restore and recovery

```
RMAN>

 RUN

{

ALLOCATE CHANNEL t1 DEVICE TYPE sbt PARMS 'SBT_LIBRARY=libopc.so
ENV=(OPC_FILE=<ORACLE_HOME>/dbs/opc<ORACLE_SID>.ora)';

SET NEWNAME FOR DATABASE TO NEW;

RESTORE DATABASE;

SWITCH DATAFILE ALL;

SWITCH TEMPFILE ALL;

SQL "ALTER DATABASE RENAME FILE ''<Specify the location of the source database online redo logfile'' TO ''<Specify
the location of restore database online redo logfile>''";

 ………………..

}
```

If above step fails with below error, then run (RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK;) and repeat above step

```
RMAN-06091: no channel allocated for maintenance (of an appropriate type)
```

Enable block change tracking

```
/home/oracle> sqlplus “/as sysdba”

SQL> ALTER DATABASE ENABLE BLOCK CHANGE TRACKING;
```

Find the system change number (SCN) to make the database consistent

```
RMAN>

RUN

{
```

```
ALLOCATE CHANNEL t1 DEVICE TYPE sbt PARMS 'SBT_LIBRARY=libopc.so
ENV=(OPC_FILE=/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/opcdboda1.ora)';

RESTORE DATABASE PREVIEW;

}
```

Recover the database to that point as per above command output, where scn is the SCN identified in the previous step.

```
RMAN>

RUN

{

 ALLOCATE CHANNEL t1 DEVICE TYPE sbt PARMS 'SBT_LIBRARY=libopc.so
ENV=(OPC_FILE=/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/opcdboda1.ora)';

RECOVER DATABASE UNTIL SCN <SCN from the previous command>;

}
```

Open the database with the RESETLOGS option after restore and recovery is complete

```
RMAN> ALTER DATABASE OPEN RESETLOGS;
```

- Backup, restore and recovery using Transparent Data Encryption (TDE)
  - Ensure TDE is enabled. The TDE WALLET status must be set to OPEN. The WALLET_TYPE can be AUTOLOGIN, for an auto-login wallet (preferred), or PASSWORD, for a password-based wallet. On a multitenant database, make sure that the wallet is open on all PDBs as well as the CDB, and that the master key is set for all PDBs and the CDB.
  - Ensure to set ENCRYPTION_WALLET_LOCATION in the $ORACLE_HOME/network/admin/sqlnet.ora file.
  - Backing Up (Level 0) to Oracle Database Backup Cloud Service

    ```
    RMAN> SET ENCRYPTION ON;

    RMAN> BACKUP DEVICE TYPE SBT AS COMPRESSED BACKUPSET DATABASE PLUS ARCHIVELOG FORMAT
    '%d_%U';
    ```

  - Backing Up (Level 1) to Oracle Database Backup Cloud Service

    ```
    RMAN> SET ENCRYPTION ON;

    RMAN> BACKUP DEVICE TYPE SBT AS COMPRESSED BACKUPSET INCREMENTAL LEVEL 1 DATABASE;
    ```

  - Backup Validation

    ```
    /home/oracle> rman target /

    RMAN> SET ENCRYPTION ON;

    RMAN> RESTORE DATABASE VALIDATE CHECK LOGICAL;
    ```

  - Restore and Recover the database

    Note1: If TDE wallet type is PASSWORD based, then you need to use both below rman clauses for restore and recovery operation

    ```
    RMAN> SET ENCRYPTION ON;
    RMAN> SET DECRYPTION WALLET OPEN IDENTIFIED BY <specifiy TDE password>;
    ```

    Note2: If TDE wallet type is AUTOLOGIN based, then you need to use only below rman clause for restore and recovery operation

    ```
    RMAN> SET ENCRYPTION ON;
    ```

    And use previous section sample steps to complete restore and recovery of the database using TDE

- Point-in-time restore and recovery from the oracle cloud

Depending upon the date and time of the restore, need to identify the control file and use “set until time” clause to do the point-in-time recovery. Use previous section sample steps to complete restore and recovery the database using Password Encryption or TDE

- Backing Up to Oracle Database Backup Cloud Service Using Dual-Mode Encryption

  ```
  RMAN> SET ENCRYPTION ON IDENTIFIED BY '<Password';

  BACKUP DEVICE TYPE SBT AS COMPRESSED BACKUPSET DATABASE PLUS ARCHIVELOG FORMAT '%d_%U';
  ```

  For example

  ```
  RMAN> SET ENCRYPTION ON IDENTIFIED BY ' odabackup123'; BACKUP DEVICE TYPE SBT AS COMPRESSED
  BACKUPSET DATABASE PLUS ARCHIVELOG FORMAT '%d_%U';
  ```

- Convert single instance to two node RAC instance
  - If you have performed, restore and recovery onto same ODA database server, where the backup was taken and if that instance was two node RAC, then use below steps to convert single instance to two node RAC (this is becase, you can only restore and recovery onto single instance, but orginal instance was RAC, hence conversion is required), otherwise no action is required.
  - Below are the sample conversion steps and assumes db_name as dboda, instances name as dboda1 and dboda2
  - Add redo log files to second RAC instance
    - Get the size of the redo log file from instance 1 and add to second instance using below command. Add same number of redo logs of instance 1

      ```
      SQL>  ALTER DATABASE ADD LOGFILE THREAD 2 GROUP 13 SIZE 4194304000;
      Database altered.
      ```
  - Enable thread 2 for second RAC instance

    ```
    SQL>  ALTER DATABASE ENABLE PUBLIC THREAD 2;

    Database altered.
    ```

    Note: You can ignore below message as existing instance is RAC instance

    ```
    ERROR at line 1:

    ORA-01612: instance i2 (thread 2) is already enabled
    ```
  - Add cluster related parameters

    ```
    SQL> ALTER SYSTEM SET CLUSTER_DATABASE_INSTANCES=2 SCOPE=SPFILE;

    System altered.

    SQL>  ALTER SYSTEM SET CLUSTER_DATABASE=true SCOPE=SPFILE;

    System altered.

    SQL> CREATE  UNDO TABLESPACE UNDOTBS2;

    Tablespace created.
    ```

    Note: You can ignore below message as existing instance is RAC instance

    ```
    ERROR at line 1:

    ORA-01543: tablespace 'UNDOTBS2' already exists

    SQL> ALTER SYSTEM SET INSTANCE_NUMBER=1 SID='dboda1' SCOPE=SPFILE;

    System altered.

    SQL> ALTER SYSTEM SET THREAD=1 SID='dboda1' SCOPE=SPFILE;

    System altered.

    SQL> ALTER SYSTEM SET UNDO_TABLESPACE='UNDOTBS1' SID='dboda1' SCOPE=SPFILE;

    System altered.

    SQL> ALTER SYSTEM SET INSTANCE_NUMBER=2 SID='dboda2' SCOPE=SPFILE;
    ```

```
System altered.

SQL> ALTER SYSTEM SET THREAD=2 SID='dboda2' SCOPE=SPFILE;

System altered.

SQL>  ALTER SYSTEM SET UNDO_TABLESPACE='UNDOTBS2' SID='dboda2' SCOPE=SPFILE;

System altered.

SQL> ALTER SYSTEM RESET UNDO_TABLESPACE SID='*' SCOPE=SPFILE;

System altered.
```

Note: You can ignore below message

```
*

ERROR at line 1:

ORA-32010: cannot find entry to delete in SPFILE
```

- Shutdown the database instance 1

```
SQL> shutdown immediate
```

- Create the spfile in ASM

Login into ASM instance

```
[grid@odadbhost1 ~]$ asmcmd

ASMCMD> cp /u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/spfiledboda.ora .

copying /u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/spfiledboda.ora -> +DATA1/testdb/spfiledboda.ora
```

- Prepare the initdboda1.ora and initdboda2.ora files with spfile +DATA1/testdb/spfiledboda.ora

On instance 1 as oracle user

```
[oracle@odadbhost1 ~]$cat /u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/initdboda1.ora

spfile='+DATA1/testdb/spfiletestdb.ora'

[oracle@odadbhost1 ~]$
```

Edit the /etc/oratab file

```
dboda1:/u01/app/oracle/product/12.2.0.1/dbhome_1:N
```

On instance 2 as oracle user

```
[oracle@odadbhost2 ~]$/cat u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/initdboda2.ora

spfile='+DATA1/testdb/spfiletestdb.ora'

[oracle@odadbhost12~]$
```

Edit the /etc/oratab file

```
dboda2:/u01/app/oracle/product/12.2.0.1/dbhome_1:N
```

- Register the RAC instances with CRS

```
[oracle@odadbhost1 ~]$ srvctl add database -d dboda -o /u01/app/oracle/product/12.2.0.1/dbhome_1

[oracle@odadbhost1 ~]$ srvctl add instance -d dboda -i dboda1 -n odadbhost1

[oracle@odadbhost1 ~]$ srvctl add instance -d dboda -i dboda2 -n odadbhost2

[oracle@odadbhost1 ~]$ srvctl status database -d dboda

Instance dboda1 is not running on node odadbhost1

Instance dboda2 is not running on node odadbhost2
```

```
[oracle@odadbhost1 ~]$ srvctl start database -d dboda
[oracle@odadbhost1 ~]$ srvctl status database -d dboda
Instance dboda1 is running on node odadbhost1
Instance dboda2 is running on node odadbhost2
```

  - Modify diskgroup, spfile information.

    You can also modify other information with srvctl command,like password file and etc

    ```
    [oracle@odadbhost1 ~]$ srvctl modify database -d dboda -a DATA,RECO
    [oracle@odadbhost1 ~]$ srvctl modify database -d dboda -p '+DATA/dboda/spfiledboda.ora'
    ```

  - Modify diskgroup, spfile information.

    Use below command and ensure all the configuration is in place

    ```
    [oracle@odadbhost1 ~]$ srvctl config database -d dboda
    Database unique name: dboda
    Database name:
    Oracle home: /u01/app/oracle/product/12.2.0.1/dbhome_1
    Oracle user: oracle
    Spfile: +DATA/dboda/spfiledboda.ora
    Domain: example.com
    Start options: open
    Stop options: immediate
    Database role: PRIMARY
    Management policy: AUTOMATIC
    Server pools: testdb
    Database instances: dboda1,dboda2
    Disk Groups: DATA,RECO
    Mount point paths:
    Services:
    Type: RAC
    Database is administrator managed
    ```

APPENDIX D SAMPLE COMMANDS FOR BACKUP AND RECOVERY IN ZDLRA

- Backing Up the Protected Database

Use regular RMAN commands to create backups of your protected database. To schedule protected database backups, create a script that contains the required backup commands and then use any scheduling utility to schedule backups. You can create full backups, incremental backups, archived redo log backups, control file backups,or backups of specific data files and tablespaces.

To implement the incremental forever backup strategy, you need one level 0 incremental database backup and successive periodic level 1 incremental backups.Because multiple protected databases are backed up to the same Recovery Appliance, backup piece names must be unique across all protected databases. Use the substitution variables %d_%U in the FORMAT string of BACKUP commands to ensure that backup piece names are unique.

- Creating the Initial Full Backup of the Protected Database

  This section describes how to create a one time full backup of the whole protected database that includes archived redo logs. Assume that the protected database is in ARCHIVELOG mode and is configured to automatically back up the control file and server parameter file
  - Connect to the protected database as TARGET and to the Recovery Appliance catalog as CATALOG

    For example,

    ```
    /home/oracle> rman target / catalog zdlravpc/*****@zdlra-scan.example.com:1521/zdlra:dedicated
    ```
  - Ensure that the configuration steps described in section (BACKUP AND RECOVERY USING ZERO DATA LOSS RECOVERY APPLIANCE) are completed
  - Run the following command to allocate three SBT channels for the Recovery Appliance and then create a full backup of the protected database including archivedredo log files

    -
      ```
      RMAN>
      RUN
      {
      ALLOCATE CHANNEL C1 DEVICE TYPE 'SBT_TAPE'  PARMS
      'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-scan.example.com:1521/zdlra:dedicated:dedicated)' FORMAT  '%U_%d';

      ALLOCATE CHANNEL C2 DEVICE TYPE 'SBT_TAPE'  PARMS
      'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-scan.example.com:1521/zdlra:dedicated:dedicated)' FORMAT  '%U_%d';


      ALLOCATE CHANNEL C3 DEVICE TYPE 'SBT_TAPE'  PARMS
      'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-scan.example.com:1521/zdlra:dedicated:dedicated)' FORMAT  '%U_%d';

      BACKUP TAG 'db_full_incr'  CUMULATIVE INCREMENTAL LEVEL 1  DATABASE FORMAT '%d_%U'  PLUS ARCHIVELOG FORMAT '%d_%U' NOT BACKED UP;
      }
      ```

      The BACKUP ... INCREMENTAL LEVEL 1 command automatically creates a level 0 backup if no level 0 backup already exists.

- Creating Incremental Backups of the Protected Database
  - Connect to the protected database as TARGET and to the Recovery Appliance catalog as CATALOG

    For example,

    ```
    /home/oracle> rman target / catalog zdlravpc/*****@zdlra-scan.example.com:1521/zdlra:dedicated
    ```
  - Ensure that the configuration steps described in section (BACKUP AND RECOVERY USING ZERO DATA LOSS RECOVERY APPLIANCE) are completed
  - Run the following command to allocate three SBT channels for the Recovery Appliance and then create a level 1 incremental backup of the protected database including archivedredo log files

    -
      ```
      RMAN>
      RUN
      {
      ```

```
ALLOCATE CHANNEL C1 DEVICE TYPE 'SBT_TAPE'  PARMS
'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/a
pp/oracle/product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-
scan.example.com:1521/zdlra:dedicated:dedicated)' FORMAT  '%U_%d';

ALLOCATE CHANNEL C2 DEVICE TYPE 'SBT_TAPE'  PARMS
'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/a
pp/oracle/product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-
scan.example.com:1521/zdlra:dedicated:dedicated)' FORMAT  '%U_%d';

ALLOCATE CHANNEL C3 DEVICE TYPE 'SBT_TAPE'  PARMS
'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/a
pp/oracle/product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-
scan.example.com:1521/zdlra:dedicated:dedicated)' FORMAT  '%U_%d';

BACKUP TAG 'db_full_incr' CUMULATIVE INCREMENTAL LEVEL 1 DATABASE FORMAT %d_%U' PLUS
ARCHIVELOG FORMAT '%d_%U' NOT BACKED UP;
}
```

- Recovering the Protected Database
  The recovery procedures using Recovery Appliance are identical to those used to recover a database within a conventional RMAN environment. The major difference is the use of a Recovery Appliance as the source for recovery data by configuring or allocating an RMAN channel that corresponds to the Recovery Appliance backup module.
  - Connect to the protected database as TARGET and to the Recovery Appliance catalog as CATALOG
    For example,
    ```
    /home/oracle> rman target / catalog zdlravpc/*****@zdlra-scan.example.com:1521/zdlra:dedicated
    ```
  - Ensure that the configuration steps described in section (BACKUP AND RECOVERY USING ZERO DATA LOSS RECOVERY APPLIANCE) are completed
  - Restoring and Recovering an Entire Database With the Existing Current Control File
    Run the following command to allocate three SBT channels for the Recovery Appliance and then restore and recover all the data files. This example assumes that some or all the data files in the protected database are lost or damaged. However, the control file is available.

  -
    ```
    /home/oracle> sqlplus “/as sysdba”

    SQL> STARTUP MOUNT;
    ```

  -
    ```
    RMAN>
    RUN
    {
    ALLOCATE CHANNEL C1 DEVICE TYPE 'SBT_TAPE'  PARMS
    'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/app/oracle/
    product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-scan.example.com:1521/zdlra:dedicated:dedicated)'
    FORMAT  '%U_%d';

    ALLOCATE CHANNEL C2 DEVICE TYPE 'SBT_TAPE'  PARMS
    'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/app/oracle/
    product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-scan.example.com:1521/zdlra:dedicated:dedicated)'
    FORMAT  '%U_%d';


    ALLOCATE CHANNEL C3 DEVICE TYPE 'SBT_TAPE'  PARMS
    'SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,ENV=(RA_WALLET=location=file:/u01/app/oracle/
    product/12.2.0.1/dbhome_1/dbs/ra_wallet credential_alias=zdlra-scan.example.com:1521/zdlra:dedicated:dedicated)'
    FORMAT  '%U_%d';

    RESTORE DATABASE;
    RECOVER DATABASE;
    ALTER DATABASE OPEN;
    }
    ```

- Point-in-time restore and recovery from the oracle cloud
  Depending upon the date and time of the restore, need to identify the control file and use “set until time” clause to do the point-in-time recovery.
- Restoring and Recovering Tablespaces in the Protected Database
  This example demonstrates how to restore and recover one or more tablespaces in the protected database after they are accidentally dropped or corrupted. The example assumes that the database is up and running and that you will restore only the affected tablespaces.
  Run the following command to allocate SBT channels for the Recovery Appliance and then restore and recover affected tablespaces. The following command restores and recovers the USERS tablespace:
  - RMAN>
    ```
    RUN
    {
    <Allocate the channels, like in above step>
    SQL 'ALTER TABLESPACE users OFFLINE';
    RESTORE TABLESPACE users;
    RECOVER TABLESPACE users;
    SQL 'ALTER TABLESPACE users ONLINE';
    }
    ```

- Restoring and Recovering whole PDB
  This example demonstrates how to perform complete recovery for a PDB in the protected database.
  Run the following command to allocate SBT channels for the Recovery Appliance and then restore and recover whole PDB.
  - RMAN>
    ```
    RUN
    {
    <Allocate the channels, like in above step>
    ALTER PLUGGABLE DATABASE "mypdb" CLOSE IMMEDIATE;
    RESTORE PLUGGABLE DATABASE 'mypdb';
    RECOVER PLUGGABLE DATABASE 'mypdb';
    ALTER PLUGGABLE DATABASE "mypdb" OPEN;
    }
    ```
- Restoring and Recovering a whole PDB in an Oracle RAC environment
  This example demonstrates how to perform complete recovery for a PDB in the protected database.
  Run the following command to allocate SBT channels for the Recovery Appliance and then restore and recover whole PDB.
  - Ensure that all instances of the affected PDB are closed.
  - The following command closes all instances of the PDB mypdb.
  - `SQL> ALTER PLUGGABLE DATABASE "mypdb" CLOSE IMMEDIATE INSTANCES=all;`
  - RMAN>
    ```
    RUN
    {
    <Allocate the channels, like in above step>
    ALTER PLUGGABLE DATABASE "mypdb" CLOSE IMMEDIATE;
    RESTORE PLUGGABLE DATABASE 'mypdb';
    RECOVER PLUGGABLE DATABASE 'mypdb';
    ALTER PLUGGABLE DATABASE "mypdb" OPEN RESETLOGS;
    ALTER PLUGGABLE DATABASE "mypdb" OPEN INSTANCES=all;
    }
    ```
- Restoring and Recovering a control file and database, when real-time Redo Transport is configured
  This example recovers a protected database that is configured to use real-time redotransport from the loss of all database files. Since the control file too is lost, you needto first restore the control file and then perform recovery of the protected database.
  - Determine the SCN to which the protected database must be recovered by querying the RC_DATABASE view. This SCN is the highest SCN at the time the database crashed.
    ```
    SELECT final_change# FROM rc_database WHERE name='<DBNAME>';
    ```
  - This example assumes that the control file is available. If the control file is lost,then you need to first recover the control file before performing the steps listedhere.
  - `STARTUP FORCE NOMOUNT;`
  - `SET DBID=<dbid of protected database>;`

- `RESTORE CONTROLFILE;`
- `ALTER DATABASE MOUNT;`
- RMAN>

  ```
  RUN
  {
  <Allocate the channels, like in above step>
  SET UNTIL SCN <specify SCN value from above query;
   RESTORE DATABASE;
   RECOVER DATABASE;
  }
  ALTER DATABASE OPEN RESETLOGS;
  ```

APPENDIX E SETTING UP THE TRANSPARENT DATA ENCRYPTION WALLET (TDE)

After TDE enabled, the TDE WALLET status must be set to OPEN. The WALLET_TYPE can be AUTOLOGIN, for an auto-login wallet (preferred), or PASSWORD, for a password-based wallet. On a multitenant database, make sure that the wallet is open on all PDBs as well as the CDB, and that the master key is set for all PDBs and the CDB.

Use the following instructions to set up the TDE keystore wallet.

1. Set ENCRYPTION_WALLET_LOCATION in the $ORACLE_HOME/network/admin/sqlnet.ora file.

   ```
   /home/oracle>cat /u01/app/oracle/product/12.2.0.1/dbhome_1/network/admin/sqlnet.ora
   ```

   ```
   ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)(METHOD_DATA=(DIRECTORY=/u01/app/oracle/product/12.2.0.1/dbhome_1/network/admin/)))
   ```

   In case of RAC instance, set above ENCRYPTION_WALLET_LOCATION value in second RAC node $ORACLE_HOME/network/admin/sqlnet.ora file

2. Use below steps to configure Password-based keystore
   1. Connect to the database

      ```
      /home/oracle>sqlplus "/as sysdba"
      ```
   2. Create the keystore

      ```
      SQL> ADMINISTER KEY MANAGEMENT CREATE KEYSTORE
      /u01/app/oracle/product/12.2.0.1/dbhome_1/network/admin' identified by <Specify tde password>;
      ```
      For example

      ```
      SQL> ADMINISTER KEY MANAGEMENT CREATE KEYSTORE
      /u01/app/oracle/product/12.2.0.1/dbhome_1/network/admin' identified by Cloud_12##;
      ```
   3. Open the keystore

      For a non-CDB environment, run the following command

      ```
      SQL> ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY CLoud_12##;
      keystore altered.
      ```

      For a CDB environment, run the following command

      ```
      SQL> ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY CLoud_12## container = ALL;
      keystore altered.
      ```
   4. Create and activate the master encryption key.

      For a non-CDB environment, run the following command.

      ```
      SQL> ADMINISTER KEY MANAGEMENT SET KEY IDENTIFIED BY CLoud_12## with backup;
      keystore altered.
      ```

      For a CDB environment, run the following command.

      ```
      SQL> ADMINISTER KEY MANAGEMENT SET KEY IDENTIFIED BY CLoud_12## with backup container = ALL;
      keystore altered.
      ```

e. Query V$ENCRYPTION_WALLET to get the wallet status, wallet type and wallet location

```
SQL> SELECT * FROM v$encryption_wallet;
 WRL_TYPE          WRL_PARAMETER
 --------------------         --------------------------------------------------------------------------------
 STATUS                      WALLET_TYPE         WALLET_OR FULLY_BAC    CON_ID
 ------------------------------ -------------------- --------- ---------                                    ----------
 FILE                    /u01/app/oracle/product/12.2.0.1/dbhome_1/network/admin/
 OPEN                        PASSWORD            SINGLE    NO        0
```

At this stage, the password-based wallet is enabled, with status "OPEN", and WALLET_TYPE shown as "PASSWORD" in the query output above.

With this configuration of WALLET_TYPE as PASSWORD is completed. Then continue below step (3) only, if you have requirement to configure WALLET_TYPE as AUTOLOGIN, otherwise go to step 4 below

3. Use below steps to configure an auto-login keystore
   a. Execute step 1
   b. Execute step 2
   c. Create the auto-login keystore.

```
SQL> ADMINISTER KEY MANAGEMENT CREATE AUTO_LOGIN KEYSTORE FROM KEYSTORE

 '/u01/app/oracle/product/12.2.0.1/dbhome_1/network/admin/' IDENTIFIED BY CLoud_12##;

             keystore altered.
```

   d. Close the password-based wallet.

```
SQL> ADMINISTER KEY MANAGEMENT SET KEYSTORE CLOSE IDENTIFIED BY CLoud_12##;

 keystore altered.
```

   e. Query V$ENCRYPTION_WALLET to get the wallet status, wallet type and wallet location

```
SQL> SELECT * FROM v$encryption_wallet;

    WRL_TYPE WRL_PARAMETER

    -------------------- --------------------------------------------------------------------------------

            STATUS WALLET_TYPE WALLET_OR FULLY_BAC CON_ID

  ------------------------------ -------------------- --------- --------- ---------

  FILE /u01/app/oracle/product/12.2.0.1/dbhome_2/network/admin/

  OPEN AUTOLOGIN SINGLE NO
```

In the query output, verify that the TDE wallet STATUS is "OPEN" and WALLET_TYPE set to "AUTOLOGIN", otherwise the auto-login wallet is not set up correctly.

With this configuration of WALLET_TYPE as AUTOLOGIN is completed.

4. Copy wallet files
   a. If you are enabling TDE for single instance, no action is required
   b. If you are enabling TDE for Oracle RAC database, copy the below files to the same location on second RAC node,or if you confiugured wallet in a shared file system, then no action is required

```
/u01/app/oracle/product/12.2.0.1/dbhome_1/network/admin/ew*

/u01/app/oracle/product/12.2.0.1/dbhome_1/network/admin/cw*
```

REFERENCES

For details on ODA Backup and Recovery with latest releases, refer below link under section (Backup and Recover)

https://docs.oracle.com/en/engineered-systems/oracle-database-appliance/

Oracle StorageTek SL150 Modular Tape Library

https://www.oracle.com/storage/tape-storage/sl150-modular-tape-library/index.html

Using Oracle Database Backup Cloud Service

https://docs.oracle.com/en/cloud/paas/db-backup-cloud/csdbb/getting-started-oracle-database-backup-cloud-service.html

ODA (Oracle Database Appliance): ODABR a System Backup/Restore Utility (Doc ID 2466177.1)

ODA (Oracle Database Appliance): ODARescue Live Disk (Doc ID 2495272.1)

Bck2Cloud - “1-Button” Cloud Backup/Restore Automation Utility (Doc ID 2363679.1)

CONNECT WITH US

Call +1.800.ORACLE1 or visit oracle.com.
Outside North America, find your local office at oracle.com/contact.

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

Backup and Recovery Best Practices for the Oracle Database Appliance
February, 2021
Author: Giridhar Reddy Sangala
Contributing Authors: Oracle RAC Pack Cloud Innovation and Solution Engineering Team, Oracle Database Product Development