Quest® NetVault® Plug-in *for Oracle* 13.0

User's Guide

Legend

■ **WARNING: A WARNING icon indicates a potential for property damage, personal injury, or death.**

❗ **CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.**

ℹ **IMPORTANT NOTE**, **NOTE**, **TIP**, **MOBILE**, or **VIDEO:** An information icon indicates supporting information.

NetVault Plug-in *for Oracle* User's Guide
Updated - October 2020
Software Version - 13.0

Contents

Introducing Quest® NetVault® Plug-in *for Oracle*

- Quest NetVault Plug-in for Oracle: at a glance
- Key benefits
- Feature summary
- Target audience
- Recommended additional reading

Quest NetVault Plug-in *for Oracle*: at a glance

The **Quest NetVault (NetVault) Plug-in *for Oracle* (Plug-in *for Oracle*)** increases confidence in the recoverability of Oracle environments, including Oracle Real Application Clusters (RAC) and Plug-in *for Oracle* Data Guard, while eliminating the need for complex scripting. Through a web-based user interface (WebUI) and automated workflow process, the plug-in offers a centralized way to set up, configure, and define backup and restore policies for your Oracle databases. Support for online backups through user-managed or full-featured Recovery Manager (RMAN)-based backups lets you select the preferred backup method without requiring you to learn Oracle database internals. The plug-in offers a detailed level of control that minimizes downtime by allowing you to back up and restore complete databases, individual tablespaces, or individual datafiles. Through integration with a range of backup devices, your data is protected and stored offsite to meet your disaster-recovery and business-continuity goals.

Plug-in *for Oracle* is designed to speed up restores while maximizing IT staff efficiency through flexible backup and recovery options. In addition, the plug-in supports Oracle features such as Oracle RAC, Data Guard, Automatic Storage Management (ASM), and Flashback Database. The plug-in offers you advanced backup and recovery options in case of hardware failure or data loss.

Key benefits

- **Reduce risk with flexible backup and recovery options**: Plug-in *for Oracle* equips administrators with the tools to simplify backup and recovery of business-critical Oracle databases. The plug-in empowers you to create a comprehensive and flexible backup policy and reduces the need for scripting or manually issuing commands, without the need to understand Oracle database internals. The plug-in lets you choose between simple user-managed or full featured RMAN-based backups while increasing automation. The plug-in reduces risk further by providing RMAN Validate Restore and RMAN Trial Recovery support to help verify the accuracy of the recovery package in advance.

- **Minimize downtime by speeding up restores—restore only what is needed**: Plug-in *for Oracle* ensures that databases remain online and accessible during backup operations, assuring no user downtime. Integration with Oracle's Flashback Database maximizes availability by enabling you to rewind an Oracle database to a previous time to correct problems caused by logical data corruptions or user errors

without restoring physical datafiles. When needed, you can perform full, incremental, and time, SCN, and log sequence number-based, point-in-time (PIT) restores. The plug-in is designed for detailed recoveries, which lets you recover complete databases, individual tablespaces, or individual datafiles. Similarly, you can restore read-only files when they are needed. Through automated options, the plug-in reduces reliance on human interaction, which eliminates syntax errors often caused by manual intervention.

- **Ensure business continuity**: With offsite backups being an important part of the data-protection plan for business-critical applications, the plug-in takes advantage of NetVault's integration with a range of backup devices. NetVault lets you select which backup device to store the backup on. You can store the backup online in a virtual tape library (VTL). You can also duplicate the job to physical tape libraries shared by multiple Oracle databases, other proprietary databases, or even general backup files.

Feature summary

- User Managed full database backups while data is online and accessible

- Protection for single-instance, multi-instance RAC, and Data Guard environments

- RMAN-based Full, Differential, or Cumulative Incremental Backups while data is online and accessible

- Support for Oracle Multitenant Architecture with Oracle Database 12c or later. In a Container Database (CDB), you can create one or more Pluggable Databases (PDBs) or you can opt not to create any. You can also create non-CDB databases, which was the standard prior to Oracle Database 12c. For more information, see the Oracle 12c Multitenant Architecture documentation.

- Backup Parameter, Control File, Archived Redo Log Files, and External Configuration Files

- Duplicate Database backups to create local or remote duplicate databases

- Protection down to the datafile level

- Protect datafiles stored in Oracle's ASM File System or on raw devices

- Protection of backup recovery files in the Flash Recovery Area (FRA)

- Parallel backup support

- Automatic instance configuration

- Flashback Database integration

- Validate and perform trial restores

- Restore complete databases, individual tablespaces, individual datafiles, or only the corrupted data blocks

- Restore read-only files when needed

- Rename of datafiles

- Point-and-click Complete Recovery and PIT Recovery based on time, SCN, and log sequence number

- Trial Recovery

- Disaster recovery of databases

- Integration with a range of backup devices

- Point-and-click WebUI

Target audience

While advanced Oracle DBA skills are not required to create and run routine backup operations, they are required for defining the Oracle database backup-and-recovery strategy.

Recommended additional reading

- **NetVault documentation:**

    - *Quest NetVault Installation Guide*: This guide provides details on installing the NetVault Server and Client software.

    - *Quest NetVault Administrator's Guide*: This guide explains how to use NetVault and describes the functionality common to all plug-ins.

    - *Quest NetVault CLI Reference Guide*: This guide provides a description of the command-line utilities.

    You can download these guides from https://support.quest.com/technical-documents.

- **Oracle database documentation:**

    - *Oracle Database Backup and Recovery User's Guide*

    - *Oracle Database Backup and Recovery Basics*

    - *Oracle Database Backup and Recovery Reference*

    - *Oracle Database Backup and Recovery Advanced User's Guide*

    Use the following link to find documentation for your version of Oracle environment: https://docs.oracle.com/en/database/oracle/oracle-database/index.html

Defining a backup strategy

- Defining a strategy: overview
- Reviewing the available backup methods
- Reviewing the supported Oracle file types
- Defining an RMAN backup strategy
- Oracle Flashback technology
- NetVault Policy Management

Defining a strategy: overview

The primary objective of backing up data is to recover from the damage caused by a failure and resume normal operations as quickly as possible. This recovery can be accomplished only if you have a well-designed backup-and-recovery strategy that maximizes data protection and minimizes data loss.

To formulate an efficient strategy, begin with your recovery strategy, which defines your recovery requirements, and the recovery techniques you use to recover from various failure modes, such as media failure, user error, data corruption, and complete loss of the Oracle Database Server.

Each restore and recovery requirement imposes requirements on your backup strategy. After you have decided on your recovery strategy, you can plan your backup strategy, which will define your backup requirements, including:

- Oracle features you use to take, store, and manage your backups
- Backup types you implement
- Intervals at which you schedule these backups
- Situations in which you back up the data outside the regular schedule

Typically, when defining a backup strategy, you face a trade-off between mean time to recover (MTTR) and backup space usage.

The topics that follow provide information and guidelines that can help you formulate a backup strategy for use with the plug-in.

Reviewing the available backup methods

Plug-in *for Oracle* offers two backup methods: User Managed and RMAN. The plug-in supports implementing a pure User Managed or a pure RMAN-based backup strategy; that is, your backup strategy should include either User Managed backups or RMAN backups, not a combination of the two.

- User Managed backup method
- Recovery Manager (RMAN) backup method

User Managed backup method

Oracle continues to support traditional User Managed backup and recovery. With this method, the files that make up the database are backed up and restored using a combination of host operating system (OS) commands and **SQL*Plus** backup- and recovery-related capabilities.

While the plug-in's User Managed backups offer the simplest form of backups with complete point-and-click functionality for Full Backups, recovering the Oracle database involves more human interaction and DBA expertise. User Managed restores begin with point-and-click restoration of the various datafiles from the backup media to the original or new destination. Restoration of the datafiles is followed by the recovery of the entire database or parts of it. With User Managed recovery, you must determine what needs to be recovered and manually perform the recovery outside the plug-in by running a series of **SQL*Plus** commands.

The User Managed backup method offers the simplest backup strategy at the expense of longer and more complex restores, making it more suitable for less mission-critical databases.

The backup-and-restore strategies available with the User Managed method are discussed in the *Oracle Database Backup and Recovery Advanced User's Guide* and the *Oracle Database Backup and Recovery User's Guide for 11g*.

While User Managed backups are available for all supported Oracle versions, they are not supported in environments where Oracle's ASM or FRA is deployed.

Recovery Manager (RMAN) backup method

Oracle provides RMAN, which automatically integrates with sessions running on the Oracle Database Server to perform a range of backup and recovery activities. RMAN also reduces the administration work associated with your backup strategy by maintaining an extensive record of metadata about backups, archive logs, and its own activities, known as the RMAN repository. In restore operations, RMAN uses this information to eliminate the need for you to identify backup files for use in restores.

RMAN backups provide maximum reliability and flexibility when defining a backup strategy. The plug-in supports a range of backup types and options available with Oracle's RMAN tool together with the ability to handle a number of recovery scenarios with less reliance on human interaction and DBA expertise. You select what to restore, the latest backup, and, if appropriate, the time, SCN, or log sequence number for PIT Recovery, and the plug-in automatically performs the recovery without further interaction.

RMAN is Oracle's preferred solution for backup and recovery, and is the best choice for mission-critical databases. It provides maximum flexibility during the recovery process, allowing you to recover the database up to the point of failure no matter what caused the failure, whether it is a disaster, media failure, user error, or database corruption.

For a feature comparison between User Managed and RMAN backups, see *Feature Comparison of Backup Methods* in the *Oracle Database Backup and Recovery Basics* guide. The same guide also discusses the backup and restore strategies available with the RMAN method.

RMAN backups are available for all supported Oracle versions but are the only supported backup method for multi-instance RAC databases and Data Guard environments. In addition, Flashback Database is only available for the plug-in's RMAN-based backups.

The following figure illustrates the RMAN-based backup process:

Figure 1. RMAN-based backup
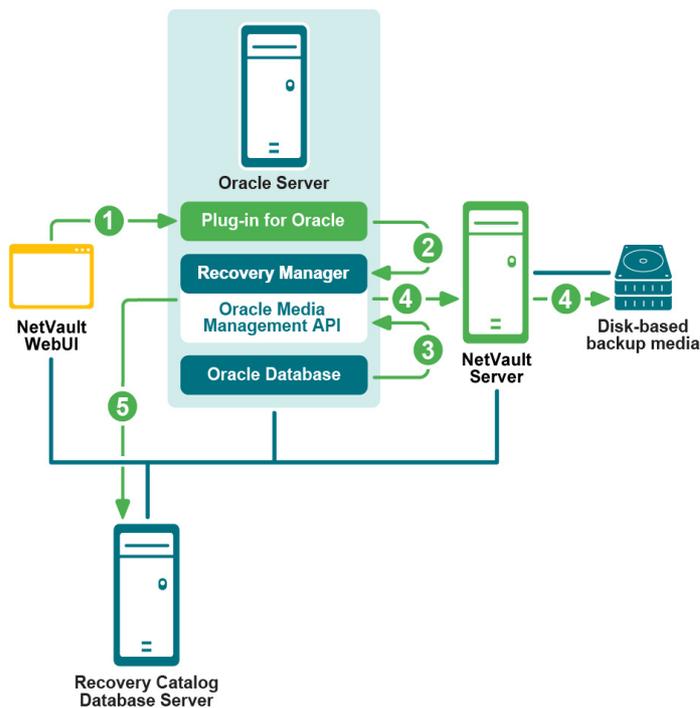
1   You define a backup job in the NetVault WebUI.

2   Plug-in *for Oracle* converts the job definition into the corresponding RMAN backup commands.

3   RMAN runs backup commands that read data from the Oracle database datafiles.

4   RMAN uses the Oracle Media Management application programming interface (API) to store backup files on the backup device managed by the NetVault Server.

5   RMAN stores backup metadata in the Recovery Catalog repository.

The following figure illustrates the RMAN-based restore and recovery process:

Figure 2. RMAN-based restore and recovery
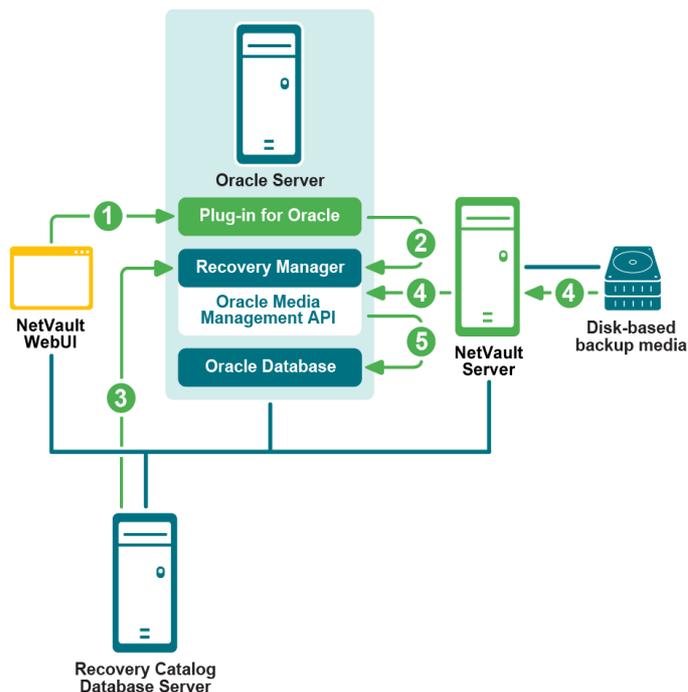
1   You define the restore job in the NetVault WebUI.

2   Plug-in *for Oracle* converts the job definition into corresponding RMAN restore and recovery commands.

3   RMAN reads the backup metadata from the Recovery Catalog repository.

4   RMAN uses the Oracle Media Management API to read the backup from the backup device managed by the NetVault Server.

5   RMAN restores datafiles to the Oracle server and recovers the database.

Reviewing the supported Oracle file types

Plug-in *for Oracle* lets you back up the following types of Oracle database files, which are critical to the successful operation of the Oracle Instance:

*   **Datafiles**: A Datafile is a physical file on disk that was created by the Oracle database and contain data structures such as tables and indexes. A datafile can belong to only one database, and is located either in an OS file system or in an ASM disk group.

*   **Control File**: Control Files are binary files that record the physical structure of the database. The file includes: the database name, names and locations of associated datafiles and online redo log files, timestamps for database creation, current log sequence number, and checkpoint information. Protecting the Control File is critical to many recovery scenarios.

*   **Parameter File**: The client-side initialization parameter file (**PFILE**) or the server-side initialization parameter file (**SPFILE**) for the Oracle database.

*   **Archived Redo Logs**: The Oracle database copies full online redo log groups to one or more archive locations on disk, where they are collectively called the archived redo log. Individual files are called archived redo log files. After a redo log file is archived, it can be backed up to other locations on disk or on tape, for long-term storage and use in future recovery operations. Without archived redo logs, your database backup and recovery options are severely limited. Your database must be taken offline before it

can be backed up. If you must restore your database from backup, the database contents are only available as of the time of the backup. Reconstructing the state of the database to a specific point is not possible without archive logs.

- **External Configuration Files**: The Oracle database depends on other files for operation such as network configuration files, **"tnsnames.ora"** and **"listener.ora,"** and password files. These files need to be protected for corruption or disaster recovery purposes.

It is critical that your backup strategy include all these file types to ensure recoverability from any type of failure, including media failure, data corruption, or a disaster.

Defining an RMAN backup strategy

The first step in defining an RMAN backup strategy is determining the location for the RMAN repository. The second step involves determining the types and frequency of backups that are needed to meet your recovery requirements. These steps are discussed in the following topics:

- Choosing the location for the RMAN repository
- Reviewing the RMAN Tablespace and Datafile backup types

Choosing the location for the RMAN repository

The RMAN repository is the collection of metadata about backup and recovery operations on the target database. The authoritative copy of the RMAN repository is always stored in the Control File of the target database. You can also use a Recovery Catalog, that is, an external database, to store this information.

- **Control File**: RMAN can conduct all necessary backup and recovery operations using just the Control File to store the RMAN repository information. This ability increases the importance of protecting the Control File as part of your backup strategy.

  Storing the RMAN repository in the Control File is especially appropriate for small databases where installation and administration of a separate Recovery Catalog database are burdensome. The only RMAN feature that is not supported in this mode is stored scripts, which are not used by the plug-in.

  When using the Control File to store the RMAN repository, Oracle highly recommends the following:

  - Enable Oracle's **Control File Autobackup** feature. This feature causes RMAN to back up the Control File automatically, and also lets RMAN restore the Control File Autobackup without access to a repository.
  - **Keep a record of your Oracle Database ID (DBID)**, which may be required to recover the database when the Control File is lost.
  - Use a minimum of two multiplexed or mirrored Control Files on **separate disks**.

- **Recovery Catalog**: You can also use an external Oracle database to store the RMAN repository. This external database is known as the Recovery Catalog. While the Control File has finite space for records of backup activities, a Recovery Catalog can store a longer history. The complexity of administering a Recovery Catalog database can be offset by the convenience of having the extended backup history available for a recovery that precedes the history in the Control File.

  RMAN maintains the Recovery Catalog's version of the RMAN repository. The database targeted for backup never accesses it directly because RMAN propagates the information about the database structure, archived redo logs, backup sets, and datafile copies into the Recovery Catalog from the target database's Control File after any operation.

  Oracle recommends that you store the Recovery Catalog in a dedicated database. If you store the Recovery Catalog alongside other data in a database and you lose that other database, you lose your Recovery Catalog as well. This configuration makes recovery more difficult. Moreover, if you store the Recovery Catalog in a database that is targeted for backup, you cannot perform whole database restores for that database. You cannot restore the whole database because the database must be in a mounted

state for restores, and the plug-in does not have access to the Recovery Catalog to perform the recovery in that state.

Reviewing the RMAN Tablespace and Datafile backup types

Plug-in *for Oracle* supports following types of RMAN backups:

- **Full Backup**
- **Incremental Backup**

The **Incremental Backups** are further classified as follows:

- **Level 0 Incremental Backup**
- **Level 1 Incremental Backup**
    - **Differential Incremental Backup**
    - **Cumulative Incremental Backup**

Full Backup

An RMAN **Full Backup** is a backup of a datafile that includes every allocated block in the file being backed up. A Full Backup of a datafile can be an image copy, in which case every data block is backed up. You can also store it in a backup set, in which case datafile blocks not in use may be skipped. An RMAN Full Backup cannot be part of an Incremental Backup strategy, that is, it cannot be the parent or base of a subsequent Incremental Backup.

Incremental Backup

An Incremental Backup of a datafile captures images of blocks in the datafile that have changed since a specific point, usually the time of a previous Incremental Backup. Incremental backups are smaller than full datafile backups, unless every block in the datafile is changed. RMAN Incremental Backups are only available for datafiles.

During media recovery, RMAN uses the block images from Incremental Backups to update changed blocks to their contents at the SCN where the block was created. RMAN completes this process in a single step. Without Incremental Backups, all changes must be applied one at a time from the archived redo logs. To reduce downtime, RMAN always chooses an Incremental Backup over reapplying individual changes stored in archived redo logs.

Multilevel Incremental Backups

RMAN's Incremental Backups are multilevel. Each incremental level is denoted by a value of 0 or 1.

- **Level 0 Incremental Backup**: A Level 0 Incremental Backup, which is the base for subsequent Incremental Backups, copies all blocks containing data, backing up the datafile into a backup set just as a Full Backup would.

    The only difference between a Level 0 Incremental Backup and a Full Backup is that a Full Backup *cannot* be included in an incremental strategy.

- **Level 1 Incremental Backup**: A Level 1 Incremental Backup can be either of the following types:

    - **Cumulative Incremental Backup**: In a Cumulative Incremental Backup, RMAN backs up all the blocks used since the most recent Level 0 Incremental Backup. Cumulative Incremental Backups speed up restores by reducing the number of Incremental Backups you need to include in the recovery process. Cumulative Backups require more space and time than Differential Backups because they duplicate the work done by previous backups at the same level.

Cumulative Backups are preferred over Differential Backups when recovery time is more important than disk space, because fewer Incremental Backups need to be applied during recovery.

▪ **Differential Incremental Backup**: In a Differential Incremental Backup, RMAN backs up all blocks that have changed since the most recent Cumulative or Differential Incremental Backup, whether at Level 1 or Level 0.

RMAN determines which Level 1 backup occurred most recently and backs up all blocks modified since that backup. If no Level 1 is available, RMAN copies all blocks changed since the Level 0 backup.

Differential Incremental Backups lead to longer restores because multiple Incremental Backups must be applied during recovery.

Archive Log backup

Archived redo logs are important for PIT Recovery and must be backed up regularly. You can back up the archive logs separately or together with the datafiles and other supported files. The plug-in also provides the option to force an additional log switch before backing up the archive logs and the option to delete the archive logs from the disk after backing them up to the backup sets.

Reviewing the Control File backup types

Protecting the Control File is critical in many recovery situations. Plug-in *for Oracle* provides two methods for protecting the Control File: Manual Backups and Autobackups. However, when using the RMAN method, you can perform either Control File Autobackups or Manual Backups but not both. When the Control File Autobackups are enabled for a database, Manual Control File Backups are not available.

Manual Backups of Control File

The User Managed and RMAN backup methods both support Manual Backup of the Control File.

• **Manual Backup of Control File with User Managed backup method**: With the User Managed backup method, the plug-in does not use the active Control File for backup and restore. During backup, the plug-in saves a snapshot of the Control File to the user-specified **Control File Save Filename (Full Path)**, and then backs up this copy of the Control File. This process ensures that a consistent copy of the Control File is protected. During restoration, the plug-in restores the Control File to the **Control File Save Filename (Full Path)** to ensure that the active Control File is not inadvertently overwritten. Manually copy this file to the original location as required. To ensure that a recent copy of the Control File is always available, include it with each User Managed backup.

• **Manual Backup of Control File with RMAN backup method**: When Control File Autobackups are disabled and the Control File node is explicitly selected on the **NetVault Selections** page, the plug-in uses RMAN's **BACKUP CURRENT CONTROLFILE** to back up the Control Files. To ensure that a recent copy of the Control File is always available, include it with each RMAN backup.

Autobackups of Control File

An Autobackup of the Control File also includes the SPFILE. The Autobackups are made after every RMAN-based backup or after making structural changes to the database. Structural changes include adding a tablespace, altering the state of a tablespace or datafile, adding an online redo log, renaming a file, adding a redo thread, and so on.

Manual vs. Autobackups

A Manual Control File backup lets you restore a specific copy of a Control File. This option is ideal when data corruption has occurred and you need to restore the version most recently saved before data corruption. You are

responsible for ensuring that the Control File is backed up with every RMAN-based backup and after every database structural change. While Manual Control File backups offer the ability to restore specific Control File backups, they cannot be used during disaster recovery or in multi-instance RAC environments.

Control File Autobackups provide the most reliability to ensure that there is always a recent backup of the Control File. In addition, they are required for disaster recovery and in multi-instance RAC environments.

Reviewing the other backup types

Other backup types include:

- Duplicate Database backup
- Flash Recovery Area (FRA) backups

Duplicate Database backup

A Duplicate Database backup lets you duplicate or clone a source database for special purposes that must not affect the overall backup and restore procedures for the source database. RMAN supports duplicating databases to increase a DBA's flexibility when performing the following tasks:

- Testing backup and recovery procedures.
- Testing an upgrade to a new release of Oracle database.
- Testing the effect of applications on database performance.
- Generating reports.

For example, you can duplicate the production database on *hostA* to *hostB*, and then use the duplicate database on *hostB* to practice restoring and recovering this database without damaging the production database.

The plug-in's support for Oracle's Database Duplication is backup-based duplication. For additional information on Oracle's Duplicate Database functionality, see *Duplicating a Database* in the *Oracle Database Backup and Recovery User's Guide*.

Flash Recovery Area (FRA) backups

Oracle introduced FRA in Oracle 10g. FRA is a disk location used to store recovery-related files such as Control Files, archived redo logs, and RMAN backups. Oracle and RMAN manage the files in the FRA automatically.

An FRA minimizes the need to manage disk space manually for backup-related files and balance the use of space among the different types of files stored in the FRA. In this way, FRA simplifies the ongoing administration of your database. Oracle recommends that you enable a recovery area to simplify backup management.

Enabling FRA can speed up restores by enabling RMAN to restore the needed datafiles from the FRA versus restoring the files from backup media managed by NetVault Server. However, storing backups in the FRA does not provide you with offsite protection for disaster recovery protection. Therefore, the plug-in provides the following Backup Destination options when FRA is enabled for the database:

- **NetVault Media Manager**
- **Disk (Backup to FRA)**
- **Both NetVault Media Manager and Disk (Backup to FRA then Backup FRA to NetVault Media)**

Additionally, the plug-in lets you perform a separate backup of the backup recovery files stored in the FRA-to-backup media managed by NetVault Server. Backup recovery files include: Full and Incremental Backup sets, Control File Autobackups, datafile copies, and archived redo logs. If an archived redo log file is missing or corrupted, RMAN looks outside the recovery area for a good copy of the log that it can use for the backup. Flashback logs, the current Control File, and online redo logs are not backed up.

Developing a Recovery Catalog backup strategy

The Recovery Catalog database is similar to any other database, and must be a key part of your backup-and-recovery strategy.

Oracle recommends that you back up the Recovery Catalog with the same frequency that you back up the target database. For example, if you make a weekly whole-database backup of the target database, back up the Recovery Catalog immediately after all target database backups to protect the record of the whole-database backup. This backup can also help you in a disaster recovery scenario. Even if you have to restore the Recovery Catalog database using a Control File Autobackup, you can then use the full record of backups in your restored Recovery Catalog database to restore the target database without using a Control File Autobackup for the target database.

Oracle recommends the following guidelines for an RMAN backup strategy for the Recovery Catalog database:

*   Run the Recovery Catalog database in **ARCHIVELOG** mode so that PIT Recovery is possible.

*   Back up the database onto two separate media, for example, disk and tape.

*   Include Archived Log files in your backups.

*   Do not use a different Recovery Catalog as the repository for the backups.

*   Configure the **Control File Autobackup** feature to **ON**.

Examples of backup sequences

Following are a few examples of backup sequences that can be implemented to meet the data protection requirements of your Oracle Database Server.

*   **Full Backups only**: When requirements guarantee data protection up to the previous day, performing Full Backups daily should be sufficient. DBAs are only required to recover the database up to the point of the last Full Backup.

*   **Full Backup and Archived Log Backups**: When requirements mandate PIT data protection but recovery time is not important, Full Backups coupled with Archived Log backups should provide the necessary protection.

    For example, Full Backups are performed every Sunday night at 11:00 p.m., while Archived Log backups are performed Monday through Saturday at 11:00 p.m.

    This strategy requires RMAN to restore and apply each Archived Log backup between the last Full Backup and the point-of-failure in succession. This strategy can lead to longer recovery as the week progresses. To speed up restores, you can include Incremental backups, which reduce the number of Archived Log backups that must be restored.

*   **Incremental Level 0 and Differential Incremental Level 1 and Archived Log Backups**: When requirements mandate PIT data protection and backups have to be as fast as possible, Incremental Level 0 and Differential Incremental Backup and Archived Redo Log backups are an ideal strategy.

    For example, Incremental Level 0 backups are performed every Sunday night at 11:00 p.m., while Differential Incremental Level 1 backups are performed Monday through Saturday at 11:00 p.m., and Archived Log backups are performed every 5 to 6 hours. Each Differential Incremental Level 1 backup includes all the changes since the last Level 0 or Level 1 backup, and the Archived Log backups include the transaction logs since the last Differential Incremental Level 1 backup.

    This strategy can lead to longer recovery because RMAN must restore the Incremental Level 0 backup, each Differential Incremental Level 1 backup since the Incremental Level 0 backup, and the Archived Log backups between the last Differential Incremental backup and the point-of-failure.

*   **Incremental Level 0 and Cumulative Incremental Level 1 and Archived Log Backups**: When requirements mandate PIT data protection and recovery time should be as fast as possible, Incremental Level 0 and Cumulative Incremental Level 1 and Archived Log backups are the best combination.

For example, Incremental Level 0 backups are performed every Sunday night at 11:00 p.m., while Cumulative Incremental Level 1 backups are performed Monday through Saturday at 11:00 p.m. and Archived Log backups are performed every 5 to 6 hours. Each Cumulative Incremental Level 1 backup includes all the changes since the last Incremental Level 0 backup, and the Archived Log backups include the Redo Logs since the last Cumulative Incremental Level 1 backup.

This strategy leads to quick recovery because RMAN only has to restore the Incremental Level 0 backup, the last Cumulative backup, and the Archived Log backups created between the last Cumulative backup and the point-of-failure.

Defining a Backup Destination strategy

When FRA is enabled, **Backup Destination** options are enabled on the **Backup Options** tab. These options let you select the destination for the backup. These options let you define a media destination strategy that best meets your needs.

- **NetVault Media Manager**: This option is the default. When selected, RMAN opens an "SBT_TAPE" channel and writes the backup files to the backup media that is managed by NetVault and specified in the **Device options** section of the **Target** tab. NetVault can manage a range of backup devices, including disk-based media, VTLs, and tape drives, which lets you implement a Disk-to-Disk-to-NetVault Media Manager backup strategy. Even though FRA has been enabled for the database, it is not used to store backup recovery files. This option is ideal for users who want to maintain a dedicated FRA for permanent files such as copies of the current Control File and online redo logs or Flashback logs.

- **Disk (Backup to FRA)**: When selected, RMAN opens a DISK channel and only writes the backup files to the FRA. Storing the backup recovery files in the FRA provides the fastest restore possible with RMAN because RMAN can restore the files directly from the FRA without having to read files from backup media. Because the FRA is typically disk storage residing on the same local or shared storage as the datafiles, it does not provide complete protection against complete media failure, corruption, or disaster recovery. To ensure that the backup recovery files are stored in offsite storage, you can use the **Both NetVault Media Manager and Disk** or perform periodic **Flash Recovery Area Backups** to back up the backup recovery files stored in the FRA-to-backup media managed by NetVault.

- **Both NetVault Media Manager and Disk (Backup to FRA then Backup FRA to NetVault Media)**: Implementing a NetVault Media Manager and Disk backup strategy provides the most protection with the fastest restores. When selected, RMAN opens a DISK channel and writes the backup files to the FRA followed by a backup of the backup recovery files stored in the FRA-to-backup media managed by NetVault in the same backup job. This option lets you perform the fastest recovery possible from isolated media failure and data corruption while providing offsite protection for complete media failure or disaster recovery. Selecting the **Both NetVault Media Manager and Disk** option lengthens the duration of the backup job because RMAN must perform two types of backups in the same backup job.

  If you want to store backup recovery files in the FRA for faster restores and you require offsite protection for disaster recovery, but you do not want to lengthen the duration of the daily backup jobs, there is an additional option for performing daily backups to the FRA while performing **Flash Recovery Area Backups** on a different independent schedule. For example, when Full Backups are performed every Sunday at 11:00pm, they are performed to **Disk**. When Incremental Backups are performed Monday through Saturday at 11:00pm, they are also performed to **Disk**. However, a **Flash Recovery Area Backup** to backup media is performed weekly every Sunday at 10:00pm before the Full Backup is performed. Additionally, you can perform the Flash Recovery Area Backup at any interval that meets business requirements.

Oracle Flashback technology

Oracle Database versions 10g and later provides a group of features known as Oracle Flashback Technology. Flashback Technology features include the Oracle Flashback Database, Oracle Flashback Table, and Oracle Flashback Drop. Flashback Technology supports viewing past states of data and winding data back and forth in time without requiring you to restore the database from backup. Depending on the changes to your database, Flashback Technology features can often reverse the unwanted changes more quickly and with less effect on database availability than media recovery.

According to Oracle Corporation, the Oracle Flashback Database lets you rewind an Oracle database to a previous time to correct problems caused by logical data corruptions or user errors. Plug-in *for Oracle* provides support for Flashback Database through the NetVault WebUI, which enables point-and-click Flashback Database options as an alternative to media recovery.

If an FRA is configured and you have enabled the Flashback Database functionality, you can use the **FLASHBACK DATABASE** command to return the database to a prior time. Flashback Database is not true media recovery because it does not involve restoring physical datafiles. Flashback Database is preferable to using the **RESTORE** and **RECOVER** commands in some cases because it is faster and easier and does not require restoring the whole database.

The time it takes to rewind a database with **FLASHBACK DATABASE** is proportional to how far back in time you need to go and the amount of database activity after the target time. The time it would take to restore and recover the whole database could be much longer. The before images in the flashback logs are only used to restore the database to a point in the past; forward recovery is used to bring the database to a consistent state at some time in the past. The Oracle database returns datafiles to the previous point, but not auxiliary files, such as initialization parameter files.

Quest recommends that you enable Flashback Database for Oracle 10g and later databases so that the administrator can flashback the database to correct problems caused by logical data corruptions or user errors. For more information, see *Oracle Flashback Technology* in *Oracle Database Concepts* and to *Configuring Oracle Flashback Database and Restore Points* in the *Oracle Database Backup and Recovery User's Guide*.

NetVault Policy Management

You can use a NetVault Job Policy to submit one or more jobs that target one or more similar clients.

If you intend to use the NetVault Job Policy across your organization's Oracle clients, Quest strongly recommends that you use, whenever possible, a generic name for the Oracle Instances deployed in different Oracle servers. Do not use a name associated with the machine on which the instance resides, or a name that is unique.

Jobs generated for instances with unique names, in general, cannot be ported to other Oracle clients in your organization. Using generic instance names improves portability and policy management across all affected clients. In this way, you can create NetVault Job Policies that can be applied to the different Oracle clients in your organization.

For more information, see the *Managing policies* topic in the *Quest NetVault Administrator's Guide*.

Installing and removing the plug-in

- Prerequisites
- Prerequisites for RMAN backups
- Recommended configuration for a single-instance non-RAC environment
- Installing or upgrading the plug-in
- Removing the plug-in

Prerequisites

Before installing the plug-in, verify that the following prerequisites are met on the machine designated as the Oracle Database Server.

NetVault Server and Client software

At a minimum, the Client version of NetVault must be installed on the machine where the plug-in is installed. In a single-instance non-RAC environment, the plug-in is installed on the server where the Oracle database resides. In a multi-instance RAC environment, the plug-in is installed on one of the nodes that hosts one of the instances for the RAC-enabled database.

Database in ARCHIVELOG mode

Verify that the database is running in ARCHIVELOG mode. For instructions on enabling the archiving of filled groups of redo log files, see the *Oracle Administrator's Guide*.

To verify that the database is properly set in this mode, perform the following steps.

1 Start **SQL*Plus**, and connect with administrator privileges to the database.

```
sqlplus SYS/<password>@<connect_identifier> AS SYSDBA
```

2 From the SQL> prompt, enter:

```
ARCHIVE LOG LIST
```

The output shows the current mode. If the **Automatic archival** property displays an **Enabled** status, the database is properly configured. The following example shows that the **Database log mode** is set to **Archive Mode**.

```
SQL> archive log list
Database log mode              Archive Mode
Automatic archival             Enabled
Archive destination            e:\oraclelogs
Oldest online log sequence     11
Next log sequence to archive   13
Current log sequence           13
SQL>
```

Prerequisites for RMAN backups

The following topics identify the requirements for performing RMAN backups:

- Adding the Oracle user to the local administrator group on Windows
- Determining the Oracle SID and Home directory in a non-RAC environment
- Creating the RMAN repository
- Registering the target database
- Enabling Control File Autobackups
- Enabling Flashback Database
- Enabling Block Change Tracking
- Using DR Series or QoreStor as the storage device for Oracle Database 12.x or later on Linux or UNIX

Adding the Oracle user to the local administrator group on Windows

If you are using Oracle 12.x or later and you created an Oracle administrative user as part of your installation, add this user to the local administrator group on Windows. This step ensures that RMAN backups have access to the NetVault Process Manager (nvpmgr).

The following steps describe how to add the user on Windows Server 2008 R2. For information for your environment, see the applicable system documentation.

1   Click **Start > Administrative Tools > Computer Management**.

2   In the **Computer Management** window, open **System Tools**, select **Local Users and Groups**, and select **Users**.

3   In the right pane, right-click the Oracle user, and select **Properties**.

4   In the **Properties** dialog box, select the **Member Of** tab, and click **Add**.

5   Add the local Administrators group, and click **OK**.

6   Click **OK** again.

Determining the Oracle SID and Home directory in a non-RAC environment

To identify the Oracle system ID (SID) and home directory, use one of the following procedures, based on the OS running on the Oracle Database Server.

Windows

On Windows, the SID and Oracle Home can be determined from the Registry.

1   Start the **Registry Editor** on the Oracle Database Server.

   a   On the taskbar, click **Start**, and then click **Run**.

   b   On the **Run** dialog box, type **regedit**, and then click **OK**.

2   Locate the following registry key:

```
My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE
```

Various directories are displayed. Each folder prefaced with **"KEY_"** represents an installation of Oracle on the target machine, if multiple installations of Oracle exist on the machine.

3   Open the folder entitled **"KEY_<Oracle_Home>"** that represents the target installation of Oracle to list the registry items in the frame on the right.

4   In the **Name** column, locate the following items, and make note of the associated entry in the **Data** column:

   ▪   ORACLE_HOME

   ▪   ORACLE_SID

5   Exit the Registry Editor.

Linux and UNIX

On Linux and UNIX, you can use the **"oratab"** file to identify the SID and Oracle Home.

1   Open the **"oratab"** file by issuing the following command in a terminal window:

```
more /etc/oratab
```

2   In the **"oratab"** file, note the application information.

   The file contains entries for each Oracle database running on the server in the following format:

   <ORACLE_SID>:<ORACLE_HOME>:<Additional Information>

   For example, if **"oratab"** contains the following:

   ```
   PROD:/u03/app/oracle/product/10.2.0/db_1:N
   ```

   It indicates that:

   ▪   **"PROD"** is the **ORACLE_SID**

   ▪   **"/u03/app/oracle/product/10.2.0/db_1"** is the **ORACLE_HOME**

Creating the RMAN repository

As discussed in Choosing the location for the RMAN repository, for an RMAN backup strategy you must first choose a location for the RMAN repository. By default, this repository is maintained in the Control File. However, for larger or more mission-critical databases, create a Recovery Catalog database that can store a much longer history than the Control File.

Creating a Recovery Catalog is a three-step process: configure the database that will contain the Recovery Catalog, create the Recovery Catalog owner, and then create the Recovery Catalog itself.

Configuring the Recovery Catalog database

When you use a Recovery Catalog, RMAN requires that you maintain a Recovery Catalog schema. The Recovery Catalog is stored in the default tablespace of the schema. SYS cannot be the owner of the Recovery Catalog. Decide which database you want to use to install the Recovery Catalog schema. Quest recommends that you create a separate database for the Recovery Catalog and run the catalog database in **ARCHIVELOG** mode.

Next, allocate space to be used by the catalog schema. The size of the Recovery Catalog schema depends on the number of databases managed by the catalog. The schema also grows as the number of archived redo log files and backups for each database increases. For more information on sizing the Recovery Catalog schema, see *Planning the Size of the Recovery Catalog Schema* in the *Oracle Database Backup and Recovery Advanced User's Guide*.

Creating the Recovery Catalog owner

After choosing the Recovery Catalog database and creating the necessary space, perform the following steps to create the owner of the Recovery Catalog and grant this user the necessary privileges:

1 Start **SQL*Plus**, and connect with administrator privileges to the database that will contain the Recovery Catalog.

```
sqlplus SYS/<password>@<connect_identifier> AS SYSDBA
```

2 Create a user and schema for the Recovery Catalog.

For example, enter:

```
CREATE USER rman IDENTIFIED BY <password>
TEMPORARY TABLESPACE temp
DEFAULT TABLESPACE tools
QUOTA UNLIMITED ON tools;
```

3 Grant the **RECOVERY_CATALOG_OWNER** role to the schema owner.

This role provides the user with all privileges required to maintain and query the Recovery Catalog.

```
GRANT RECOVERY_CATALOG_OWNER TO rman;
```

4 Grant the **CONNECT** role to the **rman** user.

```
GRANT CONNECT TO rman;
```

Creating the Recovery Catalog

After creating the catalog owner, create the catalog tables with the RMAN **CREATE CATALOG** command. This command creates the catalog in the default tablespace of the catalog owner.

1 Connect to the database that will contain the catalog as the catalog owner.

For example:

```
> rman CONNECT CATALOG rman/<password>@<connect_identifier_for_catalog>
```

2 Run the **CREATE CATALOG** command to create the catalog.

The creation of the catalog can take several minutes.

If the catalog tablespace is this user's default tablespace, you can run this command:

```
CREATE CATALOG;
```

You can also specify the tablespace name for the catalog in the **CREATE CATALOG** command. For example:

```
CREATE CATALOG TABLESPACE <tablespace_name>;
```

Registering the target database

The first step in using a Recovery Catalog with a target database is registering the database in the Recovery Catalog.

1 After making sure that the Recovery Catalog database is open, connect RMAN to the target database and Recovery Catalog database.

For example, issue the following command to connect to the catalog database.

```
> rman TARGET / CATALOG rman/<password>@<connect_identifier_for_catalog>
```

2 If the target database is not mounted, mount or open it:

```
STARTUP MOUNT;
```

3   Register the target database in the connected Recovery Catalog:

```
REGISTER DATABASE;
```

RMAN creates rows in the catalog tables to contain information about the target database. RMAN then copies all pertinent data about the target database from the Control File into the catalog, synchronizing the Catalog with the Control File.

4   Verify that the registration was successful.

```
REPORT SCHEMA
```

Registering multiple databases in a Recovery Catalog

You can register multiple target databases in a single Recovery Catalog, if they do not have duplicate DBIDs. RMAN uses the DBID to distinguish one database from a different database.

In a Data Guard environment, only the primary database is registered in the Recovery Catalog.

Enabling Control File Autobackups

Control File Autobackups are required for disaster recovery with the plug-in. For more information on Control File Autobackups, see Reviewing the Control File backup types. For more information on multi-instance RAC databases, see Enabling Control File Autobackups in a RAC environment.

You can turn on the Autobackup feature by running the following commands from the RMAN command-line utility:

1   After making sure that the Recovery Catalog database is open, connect RMAN to the target database and Recovery Catalog database.

For example, issue the following commands to connect to the catalog database:

```
> rman TARGET / CATALOG rman/<password>@<connect_identifier for catalog>
```

Optionally, you can use the following commands to change the autobackup format. The default format for Control File Autobackup is **"%F".**

```
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE
    TYPE <device type> TO '<string>'
```

– or –

```
SET CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE <device type> TO '<string>'
```

The **SET CONTROLFILE AUTOBACKUP FORMAT** command overrides the configured autobackup format in the current session only.

If you modify the default format for the autobackups, you must specify the same format during a restore.

2   Configure the default device type to "SBT_TAPE".

```
CONFIGURE DEFAULT DEVICE TYPE TO 'SBT_TAPE';
```

3   Enable Control File Autobackups.

```
CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

4   Confirm that Control File Autobackups are enabled by reviewing the RMAN configuration settings.

```
SHOW ALL;
```

Enabling Flashback Database

Flashback Database lets you rewind an Oracle database to a previous time to correct problems caused by logical data corruptions or user errors as long as the datafiles are intact. For more information on Flashback Database, see Oracle Flashback technology.

Flashback Database uses its own logging mechanism, creating Flashback Logs and storing them in the FRA. You can only use Flashback Database if Flashback Logs are available. To take advantage of this feature, set up your database in advance to create Flashback Logs.

- Determining whether Flashback Database is enabled
- Setting a retention target

Determining whether Flashback Database is enabled

1 Start **SQL*Plus**, and connect with administrator privileges to the database.

```
sqlplus SYS/<password>@<connect_identifier> AS SYSDBA
```

2 From the SQL> prompt, enter:

```
select flashback_on from v$database;
```

The output shows the status of Flashback Database.

To enable Flashback Database, first configure an FRA. For more information, see *Enabling the Flash Recovery Area* in the *Oracle Database Backup and Recovery User's Guide*.

Setting a retention target

After ensuring that FRA is configured, perform the following steps to set a flashback retention target. This retention target specifies how far back you can rewind a database with Flashback Database. From the target time onwards, the database regularly copies images of every changed block in the datafiles to the Flashback Logs.

1 Start **SQL*Plus**, and connect with administrator privileges to the database.

```
sqlplus SYS/<password> AS SYSDBA
```

2 From the SQL> prompt, shut down and mount the database:

```
shutdown immediate;
startup mount;
```

3 Optionally, set the **DB_FLASHBACK_RETENTION_TARGET** to the length of the desired flashback window in minutes.

By default, **DB_FLASHBACK_RETENTION_TARGET** is set to one day—1440 minutes. To set the flashback window, issue the following commands:

```
alter system set db_flashback_retention_target=1440;
alter database flashback on;
```

4 Open the database:

```
alter database open;
```

Flashback Database is now enabled and available for use.

Enabling Block Change Tracking

When this feature is enabled, RMAN uses a change-tracking file to record changed blocks in each datafile. This feature eliminates the need to scan every block in the datafile for an Incremental Backup, thus improving performance of the Incremental Backup jobs.

> **i** | **IMPORTANT:** Because it introduces minimal performance overhead on the database during normal operations, **Block Change Tracking** is disabled by default on Oracle 10g and later. However, the benefits of avoiding full datafile scans during Incremental Backups are considerable, especially if only a small percentage of data blocks is changed between backups. For more information on Incremental Backups and change tracking, see *Improving Incremental Backup Performance Change Tracking* in the *Oracle Database Backup and Recovery Basics* guide.

You can enable change tracking when the database is either open or mounted.

1 Start **SQL*Plus**, and connect with administrator privileges to the target database.

    ```
    sqlplus SYS/<password>@<connect_identifier> AS SYSDBA
    ```

2 Store the Block Change Tracking file in one of the following locations:

    ▪ To store it in the *default location*:

        ▫ Set the **DB_CREATE_FILE_DEST** parameter for the target database.

        ▫ Issue the following SQL statement to enable Block Change Tracking:

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

    ▪ To store it in a *user-defined location*, issue the following SQL statement:

        ```
        ALTER DATABASE ENABLE BLOCK CHANGE TRACKING USING
            FILE '/<path to desired destination>/rman_change_track.dat' REUSE;
        ```

        The **REUSE** option tells Oracle to overwrite an existing file with the specified name.

        i | **IMPORTANT:** In a RAC environment, the change-tracking must be on shared storage that is accessible from all nodes in the cluster.

        The following example stores the Block Change Tracking File in a file located in an ASM File System, which is used for shared storage in a RAC environment.

        ```
        ALTER DATABASE ENABLE BLOCK CHANGE TRACKING USING
            FILE '+DATA/o10grac/block_change_tracking.dat' REUSE;
        ```

Using DR Series or QoreStor as the storage device for Oracle Database 12.x or later on Linux or UNIX

If you are running Oracle Database 12.x or later on Linux or UNIX, backups generated using NetVault might fail if the device you select for **Target Storage** is a software-defined Quest DR Series system or QoreStor storage device. To avoid this issue, run the storage device **sc_manager** command with the **disable** option. Use the root account to complete the following steps on the applicable operating system.

1 Navigate to the NetVault **dynlib** directory.

    For example, in a default Linux installation, enter the following:

    ```
    cd /usr/netvault/dynlib
    ```

2 Run the **sc_manager** command with the **disable** option.

    For example:

    ```
    ./sc_manager disable
    ```

3 To verify the status of sc_manager, enter the following:

    ```
    ./sc_manager status
    ```

Recommended configuration for a single-instance non-RAC environment

While you can set up a single machine as both the NetVault Server and the Oracle Database Server, that is, all software installation and configuration requirements are performed on a single machine, Quest recommends that these two entities exist on separate machines. For more information on installing Plug-in *for Oracle* in a RAC environment, see Installation prerequisites.

Figure 3. Single-instance non-RAC configuration
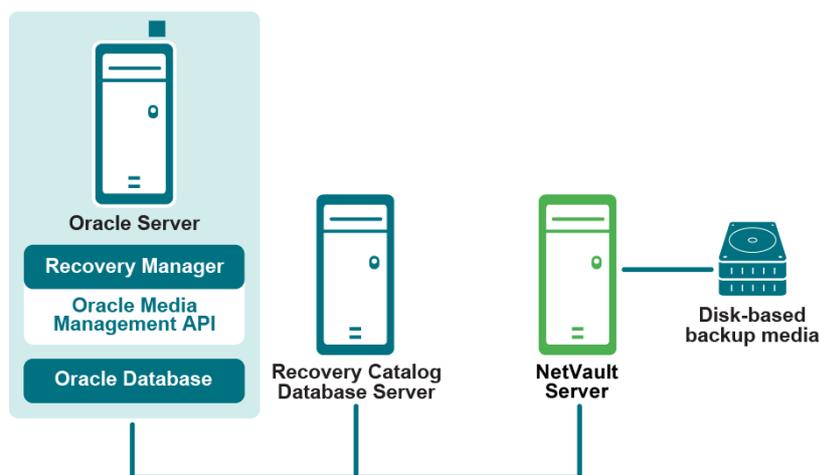
■ Plug-in for Oracle

> **i** | **IMPORTANT:** Regardless of the configured environment, separate NetVault Server and Oracle Database Server vs. a single machine configured as both, the plug-in must be installed on the host on which the Oracle Database Server resides.

Installing or upgrading the plug-in

The following topic describes the process for installing the plug-in on a single client or upgrading an existing one. If you have multiple clients of the same type, you can use the NetVault Configuration Wizard to install the plug-in on multiple clients at the same time. For more information on using push installation to update multiple clients at the same time, see the *Quest NetVault Administrator's Guide.*

1. In the Navigation pane, click **Manage Clients**.

2. On the **Manage Clients** page, select the machine that contains the Oracle Database Server, and click **Manage**.

3. On the **View Client** page, click ✚ .

4. Navigate to the location of the **".npk"** installation file for the plug-in, for example, on the installation CD or the directory to which the file was downloaded from the website.

   Based on the OS in use, the path for this software may vary on the installation CD.

5. Select the file entitled **"ora-x-x-x-x-x.npk,"** where **xxxxx** represents the version number and platform, and click **Open**.

   After the plug-in is successfully installed, a message is displayed.

IMPORTANT: When installing Plug-in *for Oracle* on Windows, the plug-in copies a dynamic-link library (DLL) file, **"ORASBT.DLL,"** to the **"…\system32"** directory. If you are installing the 32-bit version of the plug-in on a 64-bit version of Windows, this directory is named **"…\sysWOW64"**. This DLL is the link between Oracle and the media management utilities used by NetVault, and is required for the proper functioning of the plug-in. Ensure that the **ORASBT.DLL** exists in this directory.

Removing the plug-in

1   In the Navigation pane, click **Manage Clients**.

2   On the **Manage Clients** page, select the applicable client, and click **Manage**.

3   In the **Installed Software** table on the **View Client** page, select **Plug-in *for Oracle***, and click ▬ .

4   In the **Confirm** dialog box, click **Remove**.

Configuring the plug-in

- Configuring a default Target Set for RMAN backups
- Configuring default settings
- Adding a database
- Language support

Configuring a default Target Set for RMAN backups

To designate the target device for Oracle RMAN CLI-initiated backups, you can configure a default backup Target Set. The default backup Target Set is specified in the configuration of the plug-in's default attributes.

1   In the Navigation pane, click **Create Backup Job**.

You can also start the wizard from the Guided Configuration link. In the Navigation pane, click **Guided Configuration**. On the **NetVault Configuration Wizard** page, click **Create backup jobs**.

2   Next to the **Target Storage** list, click ✚ .

3   Click **Device Selection**, and select the **Specify devices** option.

4   Select the device where Control File Autobackups are stored, and click **Set**.

5   On the **Backup Job Wizard - Create Target Set** page, click **Save**.

6   On the **Create New Set** dialog box, specify a user-defined name, and click **Save**.

Configuring default settings

The plug-in lets you set default options for backup and restore jobs. You can override these options on a per-job basis.

1   In the Navigation Pane, click **Create Backup Job**, and click ✚ next to the **Selections** list.

2   In the selection tree, open the applicable client node, click **Plug-in *for Oracle***, and select **Configure** from the context menu.

3   Set the following default options:

- **Use Oracle Password File Authentication**: Oracle supports two methods to authenticate DBAs or SYSDBA users: OS authentication and password-file authentication. Select this option to enable the password file as the authentication method. Oracle's OS authentication takes precedence over password-file authentication; that is, if the requirements for OS authentication are met, even if you use a password file, the user is authenticated by OS authentication.

When this option is selected, the plug-in uses the SYSDBA authentication details from the Oracle password file to connect to the Oracle target instance with RMAN:

```
connect target <SYSDBA User>/<password>@<connect identifier>
```

When this option is cleared, the plug-in uses OS authentication when connecting to the Oracle target instance:

```
connect target /
```

- **Flash Recovery Area Enabled**: Select this option when FRA has been enabled for all the databases that reside on the Oracle Database Server where the plug-in has been installed. When this option is selected, **Flash Recovery Area Backups** and **Backup Destination** options are available. For more information on available FRA features, see Reviewing the other backup types and Defining a Backup Destination strategy.

- **Oracle SYSDBA User Name**: Specify the default Oracle user with SYSDBA privileges that is used by the plug-in to connect to the database.

- **NLS_LANG**: Select the character set that matches the National Language Support (NLS) Language or **NLS_LANGUAGE** parameter for the databases residing on this Oracle Database Server. The **NLS_LANGUAGE** parameter specifies the default language of the database. This language is used for messages, day and month names, symbols for AD, BC, a.m., and p.m., and the default sorting mechanism.

  To determine the current **NLS_LANGUAGE** parameter for the database, issue the following command from **SQL*Plus** as a SYSDBA user:

  ```
  select * from v$NLS_PARAMETERS where PARAMETER = 'NLS_LANGUAGE' or
      PARAMETER = 'NLS_TERRITORY' or PARAMETER = 'NLS_CHARACTERSET';
  ```

  The results are combined in the following format to match the available selections in the list:

  ```
  <NLS_LANGUAGE>_<NLS_TERRITORY>.<NLS_CHARACTERSET>
  ```

  For a list of languages that the plug-in supports and their corresponding **NLS_LANG** value, see Language support.

- **Incomplete backup of ALL items selected**: When multiple items are included in a backup, and the plug-in is unable to back up all the items selected even if RMAN has successfully cataloged some of the selected items, the plug-in lets you specify what action the backup should take. For example, if a job includes multiple tablespaces and archived redo logs, and the backup of archive logs is unsuccessful while the tablespaces are backed up successfully, you can specify what action the backup job should take.

  Plug-in *for Oracle* can do one of the following when this error condition occurs:

  - **Complete with Warnings — Saveset Retained:** The job returns a status of **"Backup Completed with warnings"** and a backup saveset is created that includes the items that were successfully backed up.

  - **Complete without Warnings — Saveset Retained:** The job completes and returns a status of **"Backup Completed."** The errors are logged in the NetVault binary logs and ignored on the **Job Status** page. A backup saveset is created that includes the items that were backed up.

  - **Fail — Saveset Retained:** The job returns a status of **"Backup Failed."** However, a backup saveset is generated that includes the items that were successfully backed up.

  - **Fail — No Saveset Retained:** The job returns a status of **"Backup Failed"** and no saveset of backed-up objects is kept. That is, even if some of the objects were successfully backed up, the saveset is discarded.

- **Use Catalog for RMAN Based Backups**: Select this option to specify that a Recovery Catalog database is used for maintaining a record of all RMAN backup operations performed with this plug-in.

- **Catalog Instance Name**: Enter the Oracle Instance name—which might differ from the Oracle Net Service name—for the default Recovery Catalog database. This instance name must be defined in the **"tnsnames.ora"** file on the Oracle Database Server so that the plug-in can connect to the Recovery Catalog database.

> **NOTE:** If you upgraded from a version of the plug-in before 7.0, the **Catalog Instance Name** field is labeled **Catalog Net Service Name**.

- **Catalog Owner**: Specify the user that was defined as the owner of the Catalog and was granted the **RECOVERY_CATALOG_OWNER** role.

- **User Managed Raw Devices Blocking Factor (KB)**: For User Managed backups only, use this field to set the **Block Read** units for quicker User Managed backups of raw devices. The value can be between 1 kilobyte (KB) and 64KB; for example, if you enter 8 in this field, the plug-in would read 8KB chunks of data at a time while taking backups.

- **NetVault Server** (required for RMAN Backups): Specify the name of the NetVault Server where the Oracle Database Server was added as a NetVault Client.

- **Do Restore from NetVault Server** (required for RMAN Backups): Specify the name of the NetVault Server where the Oracle Database Server was added as a NetVault Client.

- **Restore Backup taken from NetVault Client**: When a *restore* is performed by the plug-in from a terminal session, this field indicates the NetVault Client from which the original backup was performed. If no client name is entered, the restore command defaults to the local NetVault Machine Name performing the restore. **For a complete description of this functionality, see CLI-based RMAN backups and restores: an overview**.

- **CLI Backup/Autobackup Advanced Options Set**: This option requires that you first create a set of desired options on the **Advanced Options** tab, and then save it with a specific name. To have all CLI-based RMAN backups automatically use the Advanced Options specified in this set, enter the name of this set in the text box. If a set name is not specified, all CLI-based RMAN backups use the default Advanced Options.

  When defining the **CLI Backup/Autobackup Advanced Options Set**, the **Discard After Days/Weeks/Years** option must be specified instead of the **Discard After Full Backups** in the **Backup Life** section of the **Advanced Options** tab. CLI-based RMAN backups do not support backup-life generation for Full Backups and backups might expire unexpectedly if the **Discard After Full Backups** option is specified.

- **CLI Backup/Autobackup Target Set** (required for RMAN Backups): Specify the name of the Target Set that was created in Configuring a default Target Set for RMAN backups.

  > **IMPORTANT:** If a Target Set or Advanced Options Set is named in its appropriate field, you override it from the command line by specifying the name of a different set as a variable in the syntax. For more details on naming Sets in the CLI syntax, see CLI-based RMAN backups and restores: an overview.

- **Slave Connection Timeout (minutes - 0 = Never)**: Use this option to set an amount, in minutes, that serves as an inoperability timeout between the plug-in and the Oracle database. If the plug-in senses no activity for this length of time during a backup, it times out and disconnects from the backup. The default entry of zero ("0") indicates that no timeout exists.

- **Generate Script Default Directory**: Enter the full path name of the default directory where RMAN-generated scripts are stored. This directory is used as the default directory for the **Generate RMAN Script** feature available on all Backup Options and Restore Options tabs.

- **Generate Sbt trace to Oracle Dump Directory**: Select this option to send the System Backup to Tape (SBT) trace information to the Oracle dump directory. You can then review the log file to see the sequence of trace commands that were run by the Oracle Server.

- **CLI will not check Quest NetVault Server name**: Select this option if you do not want the RMAN CLI to verify that the specified NetVault Server is available and can be contacted. Do not select this option if you want to ensure that backup jobs do not time out while trying to process a job for an unknown NetVault Server.

- **Preferred Phase for Restores (0 = default)**: The **Advanced Options** section of the **Create Backup Job** page lets you create a Secondary Copy of a backup using either the **Duplicate** or **Data Copy** option. When you restore an RMAN backup, use this field to indicate which backup phase you prefer to use to complete the restore if the applicable backup set from the phase is available. The default entry of **0** indicates that NetVault automatically chooses the copy to which it

has easier access, phase 1 or phase 2. To instruct NetVault to use the original (phase 1) backup if it is available, enter a **1**. To instruct NetVault to use the Secondary Copy (phase 2) for a Data Copy backup if it is available, enter a **2**.

> **i** | **IMPORTANT:** For a Data Copy backup, NetVault always runs the restore from the original (phase 1) backup unless you enter a **2**.

- **Maximum Number of Objects in Contents Courtesy List (x100 - 0 = all)**: Use this field to specify a maximum number of items that the plug-in displays on the **Backup Contents** tab of **Restore Options**. Limiting the number of items reduces the amount of memory used during runtime when the plug-in generates the backup index. The value that you enter is multiplied by a factor of 100. For example, if you enter 50, the maximum is 5000. To specify no limit, that is, display all items, enter 0. Quest recommends that you enter 0 and change the entry only if the plug-in encounters failures during backups while generating the backup index or **Restore Options**.

4   To save the settings, click **OK**.

Adding a database

To begin using the plug-in for backup and restore, first add the target Oracle database to the plug-in. To add a database, perform the following steps for *each* database running on the Oracle server that is protected with the plug-in.

1   In the Navigation Pane, click **Create Backup Job**, and click ➕ next to the **Selections** list.

2   In the selection tree, open the applicable client node, click **Plug-in** *for Oracle*, and select **Add Database** from the context menu.

3   On the **Oracle Instance Details** tab, set the following parameters:

All the parameters pertaining to the Oracle Instance are mandatory for both RMAN and User Managed backup methods.

- **Oracle SID**: Enter the SID for the target Oracle database. For help on determining the Oracle SID, see Determining the Oracle SID and Home directory in a non-RAC environment.

> **i** | **NOTE:** In general, the SID is case-sensitive for Linux and UNIX. The plug-in uses the SID that you enter without modifications to access the Oracle Instance. To determine whether the SID that you enter is case-sensitive, see the documentation for your Oracle installation.

- **Oracle Home**: Enter the complete path to the installation directory of the target database. For help on determining the Oracle Home, see Determining the Oracle SID and Home directory in a non-RAC environment.

> **i** | **IMPORTANT:** To prevent an Oracle error from occurring, do not include a forward slash at the end of the **ORACLE_HOME** directory on Linux and UNIX. Using a backward slash on Windows does not generate an error.

- **Oracle SYSDBA User Name**: Specify an Oracle user with **SYSDBA** privileges that is used by the plug-in to connect to the database.

- **Oracle SYSDBA Password**: Enter the password associated with the name specified in the preceding field.

> **i** | **NOTE:** You can also configure the authentication details for Plug-in *for Oracle* later by accessing the **Set Database Password** functionality. For more information, see Setting the database password.

> **i** | **IMPORTANT:** If the plug-in's authentication details are not configured on the **Add Oracle Database** dialog box or by using **Set Database Password**, a prompt appears each time a connection to the database is required.

- **For Linux and UNIX** *only*: Configure these parameters:

  - **Oracle Software Owner**: Specify the name of the Linux or UNIX user that owns all the Oracle software. This user must have the Oracle Inventory group as its primary group and the **OSDBA** and **OSOPER** groups as secondary groups. The usual name chosen for this user is **oracle**.

  - **Oracle Software Group**: Specify the name of the Linux or UNIX group that owns the Oracle inventory, which is a catalog of all Oracle software installed on the system. The usual name chosen for this group is **oinstall**.

- **NLS_LANG**: Select the character set that matches the NLS Language or **NLS_LANGUAGE** parameter for the databases residing on this Oracle Database Server. The **NLS_LANGUAGE** specifies the default language of the database. This language is used for messages, day and month names, symbols for AD, BC, a.m., and p.m., and the default sorting mechanism.

  To determine the current **NLS_LANGUAGE** parameter for the database, issue the following command from **SQL*Plus** as a SYSDBA user:

  ```
  select * from v$NLS_PARAMETERS where PARAMETER = 'NLS_LANGUAGE' or
      PARAMETER = 'NLS_TERRITORY' or PARAMETER = 'NLS_CHARACTERSET';
  ```

  The results are combined in the following format to match the available selections in the list:

  <NLS_LANGUAGE>_<NLS_TERRITORY>.<NLS_CHARACTERSET>

  For a list of languages that the plug-in supports and their corresponding NLS_Lang value, see Language support.

  If you are unable to find an **NLS_LANG** in this list that exactly matches your current settings from the **SQL*Plus** query, you can enter—or copy and paste—the exact value in the list. Moreover, a perfect match is not required; a compatible match is sufficient. For example, any charset 1252 is compatible with the charset WE8MSWIN1252.

- **Parameter File(s) Path**: When an Oracle Instance is started, the parameters specified within the initialization parameter file establish the characteristics of the Instance. These initialization parameters are either stored in the **PFILE** or **SPFILE**.

  Specifying the path is required when using the User Managed backup method. When using the RMAN backup method, you have three options for backing up the SPFILE. These options include:

  - Enabling Control File Autobackups, which include Autobackups of the SPFILE by selecting the **Control File Autobackup Enabled** option on the **RMAN Details** tab.

  - Disabling Control File Autobackups and using RMAN commands to back up the SPFILE when the **Use RMAN Commands to Backup SPFILE** option on the **RMAN Details** tab is selected.

  - Disabling Control File Autobackups and using File System-based commands to back up the SPFILE by entering the path for the SPFILE in the **Parameter File(s) Path** field.

  If you are using the User Managed or RMAN backup method but you want to use File System-based backups for the SPFILE, specify the complete path; include the directory and filename of the PFILE or SPFILE, depending on whichever is in use. Provide the details for both if you are using **SPFILE** as well as **PFILE**.

  The following table provides the default paths:

Init. file	Platform	Default name	Default location
PFILE	Windows	**initSID.ora**	%ORACLE_HOME%\database\
	Linux and UNIX	**initSID.ora** where the SID represents the ORACLE_SID	$ORACLE_HOME/dbs
SPFILE	Windows	**spfileSID.ora** where SID represents the ORACLE_SID	%ORACLE_HOME%\database\
	Linux and UNIX	**spfileSID.ora** where SID represents the ORACLE_SID	$ORACLE_HOME/dbs

You can enter multiple parameter file paths by using a comma as a delimiter. The default length of this field is **2048** characters.

- **Auto Discover**: To complete the remaining database-configuration fields automatically, click this button after you have completed the **Oracle SID**, **Oracle Home**, **Oracle SYSDBA User Name**, and **Oracle SYSDBA Password** fields.

  > **i** | **IMPORTANT:** When using the **Auto Discover** option, Quest recommends that you review all database-configuration fields for accuracy.

4 On the **RMAN Details** tab, set the following parameters if you intend to use the RMAN backup method.

For more information on Recovery Catalog, see Prerequisites for RMAN backups.

- **Control File Autobackup Enabled (Disables Manual Control File and SPFILE Backups)**: Select this option if the Control File Autobackups have been enabled for this database. Control File Autobackups are required for disaster recovery and multi-instance RAC environments. When this option is selected, Manual Control File backups are disabled and the Control File node is not available on the **NetVault Selections** page. For more information on Control File Autobackups in a multi-instance RAC environment, see Enabling Control File Autobackups in a RAC environment.

- **Use RMAN Commands to Backup SPFILE**: Available only when **Control File Autobackup Enabled** is not selected. When this option is selected, the RMAN commands are used to back up the SPFILE. When this option is not selected, a File System-based backup of the SPFILE is performed of the SPFILE that is specified in the **Parameter File(s) Path** field on the **Instance Details** tab.

- **Use Catalog for RMAN Based Backups**: Select this option if the RMAN repository for this database is stored in a Recovery Catalog database, and then configure the following parameters:

    - **Catalog Instance Name**: Enter the Oracle Instance name, which might differ from the Oracle Net Service name, for the Recovery Catalog database. This instance name must be defined in the **"tnsnames.ora"** file on the Oracle Database Server so that the plug-in can connect to the Recovery Catalog database.

    - **Catalog Owner**: Specify the user that was specified as the owner of the Catalog and was granted the **RECOVERY_CATALOG_OWNER** role.

    - **Catalog Password**: Provide the password associated with the Catalog owner name specified in the preceding field.

    - **CLI Backup/Autobackup Advanced Options Set:** If you are using CLI-based backups, specify the name of the Backup Advanced Options Set that you want to use for this database.

    - **CLI Backup/Autobackup Target Set:** If you are using CLI-based backups, specify the name of the Backup Target Set that you want to use for this database.

    - **CLI Database Name (optional):** If you are using CLI-based backups, enter the database name used by the CLI.

  > **i** | **IMPORTANT:** If the Recovery Catalog is not enabled, the plug-in automatically stores the RMAN Repository information in the database's Control File.

5 On the **User Managed Details** tab, set the following parameters:

This tab is *only* for the User Managed backup method; it is not supported in multi-instance RAC or Data Guard environments.

> **i** | **NOTE:** The User Managed backup method can only back up File System objects; it cannot back up archive logs saved in ASM storage. You can leave the **Archive Log Destination Directory** field on the **User Managed Details** tab blank. Backing up databases in ASM storage is only available using the RMAN backup method.

Provide the following information for making User Managed backups of Control Files and archived redo logs. You can leave these fields blank if your backup-and-recovery strategy includes the RMAN method.

- **Control File Save Filename (Full Path)**: When the plug-in uses the User Managed backup method, backups and restores of the Control File are not performed using the active Control File. A snapshot of the Control File is created in the file named by the **Control File Save Filename (Full Path)**, and the snapshot or copy of the Control File is backed up. This step ensures that a consistent copy of the Control File is protected.

  During a User Managed restore sequence that includes restoring Control Files, the Control Files are restored to the **Control File Save Filename (Full Path)**. This step ensures that active Control Files are not overwritten. During the User Managed recovery processes, copy the file manually from the **Control File Save Filename (Full Path)** to the original location as required.

  Specify the full path name; include a filename, on the Oracle Database Server that is used as the **Control File Save Filename (Full Path)**, and ensure that the plug-in has read/write privileges for the specified location and file.

- **Archive Log Destination Directory**: Enter the complete path to the archive log directory as set in the **LOG_ARCHIVE_DEST** or **LOG_ARCHIVE_DEST_n** parameters in the parameter file. If you archive the redo logs to multiple locations, specify only the primary destination directory in this field.

  > **i** | **IMPORTANT:** The User Managed backup method does not support dynamic path names such as a daily created subfolder in the Archive Log Destination Directory.

- **Archive Log Format**: Specify the extension of the archive log files. The archive file pattern depends on what is established for the **LOG_ARCHIVE_FORMAT** setting in the parameter file. The following list provides some examples:

  - If LOG_ARCHIVE_FORMAT = "%t_%s_%r.dbf", the archive file pattern is "*.dbf".
  - If LOG_ARCHIVE_FORMAT = "%t_%s.arc", the archive file pattern is "*.arc".
  - If LOG_ARCHIVE_FORMAT = "arc%s.%t", the archive file pattern is "arc*.*".

6 On the **External Configuration** tab, set the following parameters:

Although the database depends on other types of files for operation, such as network configuration files and password files, these files cannot be backed up with Oracle RMAN commands. However, they are useful when performing disaster recovery or recovering from user error during Oracle configuration activities.

In the **Oracle Network Configuration Directory, or File(s) Path*** and **Oracle Password Directory, or File(s) Path*** fields, you can enter one or more of the following:

- **Directory Path**: If you enter a directory path, the plug-in scans the directory for filenames that might correspond to an Oracle Network Configuration File or an Oracle Password File. The rest of the files in the directory are ignored.

- **Fully Qualified Path to a File**: The path includes directory and filename.

The **External Configuration** fields are as follows:

- **Oracle Network Configuration Directory or File(s) Path**: Enter the directory where the Oracle Network Configuration files reside, or the list of Oracle Network Configuration files.

- **Oracle Password Directory or File(s) Path**: Enter the directory where the Oracle Password files reside, or the list of Oracle Password files.

NOTE: Oracle lets you install multiple versions of the Oracle Database and Oracle Client software to on the same system. In this type of environment, there might be more than one Oracle Network Configuration Directory, with each one having its own **"tnsnames.ora"** file. Connecting to a database that has multiple "tnsnames.ora" files might display the following error message: **ORA-12154: TNS: could not resolve service name or SQL*NET Unable to resolve service name**

Although the Oracle service is correct, included in the appropriate "tnsnames.ora" file, and located in the appropriate Oracle network configuration directory, this issue occurs because the actual "tnsnames.ora" file that is being used by the Oracle Server is from a different directory; it is not the file that includes the database you are connecting to.

You should use the environment variable **TNS_ADMIN** to indicate the directory where the Oracle network configuration files are located, including the "tnsnames.ora" file that is used to resolve the Oracle service name. Otherwise, the default location might be used, which might result in the preceding connection error.

When you complete the **Oracle Network Configuration Directory or File(s) Path** field, make sure that you include *all* Oracle network configuration directories from the different locations for which you have configured "tnsnames.ora" files. On your system, make sure that you maintain the TNS_ADMIN variable with the proper configured location for the Oracle network configuration files that should be used for connecting to the database.

If you restore the database to an alternate server, make sure that the TNS_ADMIN variable is set with the location of the Oracle network configuration files that include the "tnsnames.ora" file with the Oracle database entry to resolve the Oracle service name that you want to connect to.

7   On the **Optional OS Authentication Details** tab (for Windows-based OS *only*), set the following parameters:

Use the following fields to specify a Windows Administrator user name that allows the Windows installation to "impersonate" an Oracle SYSDBA user. This login is only necessary if backups fail due to **"ORA-01031 insufficient"** errors.

- **Windows Administrator User Name**: Enter the Windows Administrator user name for the account under which the plug-in should run. The name specified in this field must have been already created in the Windows OS.

- **Password**: Enter the password associated with the name specified in the preceding field.

- **Windows Domain**: Specify the domain to which the user belongs. If it is the Local domain, leave this field blank.

8   To save the settings, click **OK**.

9   If you are using Oracle Multitenant Architecture with Oracle Database 12c or later and your CDB includes one or more PDBs, make sure that an entry is included for each PDB in the **"tnsnames.ora"** file and that the status of the PDB is in read-write mode.

When you create a PDB, a default Oracle service with the same name as the PDB name is created automatically. However, an entry to the **"tnsnames.ora"** file might not get added automatically.

Editing database information

1   In the Navigation Pane, click **Create Backup Job**, and click ✚ next to the **Selections** list.

2   In the selection tree, open the applicable client node, and double-click **Plug-in** *for Oracle*.

3   Click the name of the applicable database to highlight it; do *not* select the check box.

4   From the context menu, select **Edit Database**.

The **Edit Oracle Database** dialog box displays configured values for the database instance.

5   Reconfigure the parameters as required.

For information on the fields in this dialog box, see Adding a database.

6   To save the changes, click **OK**.

> **i** | **IMPORTANT:** The password fields on the **Edit Oracle Database** dialog box are blank. Set the passwords again, even if there are no changes. If this step is not done, the existing values are reset to blank when you save the updated information. To update only the authentication details for the plug-in, select the **Set Database Password** option instead of **Edit Database**.

Setting the database password

Plug-in *for Oracle* lets you save the Oracle authentication details that are used to connect to the database automatically. Saving the authentication information eliminates the need to enter the details each time a connection to the database is required. You can save these details either at the time of adding a database or any time later using the **Set Database Password** option.

1   In the Navigation Pane, click **Create Backup Job**, and click ✚ next to the **Selections** list.

2   In the selection tree, open the applicable client node.

3   Double-click **Plug-in *for Oracle***, click the applicable database, and select **Set Database Password** from the context menu.

4   Enter the following information:

   ▪   **Account Name**: Enter an Oracle user name with **SYSDBA** privileges.

   ▪   **Password**: Enter the password associated with the name specified in the preceding field.

5   To save the settings, click **OK**.

> **i** | **IMPORTANT:** If the password for the configured SYSDBA user is changed in the Oracle database, update the password in the plug-in.

Removing a saved password

1   In the Navigation Pane, click **Create Backup Job**, and click ✚ next to the **Selections** list.

2   In the selection tree, open the applicable client node.

3   Double-click **Plug-in *for Oracle***, click the applicable database, and select **Remove Saved Password** from the context menu.

4   When the **Reset Login** dialog box appears, click **Yes** to clear the authentication details.

Removing a database

If a previously configured database is no longer in use, you can remove it from the plug-in.

1   In the Navigation Pane, click **Create Backup Job**, and click ✚ next to the **Selections** list.

2   In the selection tree, open the applicable client node.

3   Double-click **Plug-in *for Oracle***, click the applicable database, and select **Remove Database** from the context menu.

4   When the **Remove Oracle Database** dialog box appears, click **Yes** to remove the database.

Viewing database details

To view database details about a previously configured database, including enabled database options and whether prerequisites for the plug-in have been met, perform the following steps.

1   In the Navigation Pane, click **Create Backup Job**, and click ✚ next to the **Selections** list.

2   In the selection tree, open the applicable client node.

3   Double-click **Plug-in *for Oracle***, click the applicable database, and select **View Details** from the context menu.

The Details dialog box displays the following information:

- ▪ **Database Name**: Name of the selected database as identified in the NAME column of the sys.v$database v$ table.

    > ℹ **NOTE:** This field represents the database name, not the Oracle SID; they are not the same. The database name is the name of the database while the Oracle SID is the name of the instance used to access the database. Although they are frequently set to use the same name, it is not a requirement. Even if the names are the same, they might use different forms of upper- and lowercase, which also means that they are not identical.

- ▪ **DBID**: Oracle DBID for the selected database. DBID is the internal, uniquely generated number that differentiates databases and is required during the disaster recovery process.

- ▪ **Oracle Version**: Oracle version or release of the selected database. When the selected database is the Oracle Enterprise Edition, this information is also displayed. If the selected database is any other edition of Oracle, only the Oracle version is displayed.

- ▪ **NLS_LANG**: Displays the default language of the database. This language is used for messages, day and month names, symbols for AD, BC, a.m., and p.m., and the default sorting mechanism.

- ▪ **Flash Recovery Area Destination**: When FRA is enabled, displays the destination of the FRA.

- ▪ **Flashback Database Enabled**: Indicates **YES** if the Flashback Database feature is enabled or **NO** if not.

- ▪ **Real Application Clusters**: Displays **TRUE** if the instance is part of multi-instance RAC environment or **FALSE** if the instance is a single-instance environment.

- ▪ **Block Media Recovery**: Displays **TRUE** if Block Media Recovery is available for the selected database and is based on the Oracle Edition of the database.

- ▪ **Block Change Tracking**: Displays **ENABLED** if Block Change Tracking has been enabled for the database or **DISABLED** if not.

- ▪ **Control File Autobackup**: Displays **ON** if Control File Autobackups have been enabled for the database or **OFF** if not.

- ▪ **Log Mode**: Displays **ARCHIVELOG** if the database has automatic archiving of the redo logs enabled or **NOARCHIVELOG** if not.

4   To close the dialog box, click **OK**.

Using RMAN Post-Backup Scripts

The plug-in provides a feature to run a script containing RMAN commands on completion of a backup job. For example, you can use this feature to perform an RMAN Repository crosscheck; this crosscheck ensures that data about backups in the RMAN Repository are synchronized with corresponding data in the NetVault Database by running a post-backup script that contains the appropriate RMAN commands. This functionality is designed for Oracle DBAs with advanced RMAN and scripting experience.

- • Creating an RMAN Post-Backup Script

- • Updating a Post-Backup Script

- • Removing a Post-Backup Script

Creating an RMAN Post-Backup Script

i | **IMPORTANT:** Selecting the database when adding an RMAN Post-Backup Script is not supported. If you want to use an RMAN Post-Backup Script for the backup job, you might not be able to select the database node because it selects all subitems by default, including any predefined scripts. If you define more than one RMAN Post-Backup Script and you select the database node in your backup selections, verify that only one post-backup script is selected *before* you save the backup job.

1   In the Navigation Pane, click **Create Backup Job**, and click ╋ next to the **Selections** list.

2   In the selection tree, open the applicable client node.

3   Double-click **Plug-in** *for Oracle*, click the applicable database, and select **Add RMAN Post-Backup Script** from the context menu.

You can also access this command as follows:

a   Open the database node.

b   Double-click the **Post-Backup Scripts** item, and select **Add RMAN Post-Backup Script**.

4   On the **Add RMAN Post-Backup Script** dialog box, enter the script details:

- ▪ **Script Name**: Enter a name for the post-backup script. The script name must not contain the following characters:

    - ▫ Colon (:)

    - ▫ Exclamation point (!)

    - ▫ Pipe (|)

    - ▫ Comma (,)

    - ▫ Semi colon (;)

    - ▫ Equals (=)

    - ▫ Less than (<)

    - ▫ Greater than (>)

    - ▫ Left square bracket (])

    - ▫ Right square bracket ([)

    - ▫ Quotation mark (")

    - ▫ Backslash (\)

    An error message appears if you include these characters.

- ▪ **Script**: Enter the RMAN commands, including the connection strings required for connecting to the target, Catalog, or auxiliary database used in the script.

- **Use 'catalog'/Use 'nocatalog'**: Select whether the script connects to the Catalog with these options.

5   Click **Save** to store the script and close the dialog box.

Updating a Post-Backup Script

1   Open the database node, and double-click the script listed beneath the **Post-Backup Scripts** node.

2   Make the required changes, and click **Save**.

Removing a Post-Backup Script

1   Open the database node, select the script listed beneath the **Post-Backup Scripts** node, and click 🗑.

2   Click **Yes** in the confirmation dialog box.

Language support

The following table lists the default languages available for the plug-in, and their corresponding **NLS_LANG** values.

Table 1. Values

Language	NLS_LANG value
Arabic	ARABIC_UNITED ARAB EMIRATES.AR8MSWIN1256
Brazilian Portuguese	BRAZILIAN PORTUGUESE_BRAZIL.WE8MSWIN1252
	BRAZILIAN PORTUGUESE_BRAZIL.UTF8
	BRAZILIAN PORTUGUESE_BRAZIL.WE8ISO8859P1
	BRAZILIAN PORTUGUESE_BRAZIL.WE8ISO8859P15
Catalan	CATALAN_CATALONIA.WE8ISO8859P1
	CATALAN_CATALONIA.WE8MSWIN1252
	CATALAN_CATALONIA.WE8ISO8859P15
Chinese Simplified	SIMPLIFIED CHINESE_CHINA.ZHS16CGB231280
	SIMPLIFIED CHINESE_CHINA.UTF8
	SIMPLIFIED CHINESE_CHINA.ZHS16GBK
Chinese Traditional	TRADITIONAL CHINESE_TAIWAN.ZHT16BIG5
Croatian	CROATIAN_CROATIA.EE8MSWIN1250
Czech	CZECH_CZECH REPUBLIC.EE8MSWIN1250
Danish	DANISH_DENMARK.WE8ISO8859P1
	DANISH_DENMARK.WE8MSWIN1252
	DANISH_DENMARK.WE8ISO8859P15
Dutch	DUTCH_THE NETHERLANDS.WE8ISO8859P1
	DUTCH_THE NETHERLANDS.WE8MSWIN1252
	DUTCH_THE NETHERLANDS.WE8ISO8859P15

Table 1. Values

Language	NLS_LANG value
English	AMERICAN_AMERICA.AL32UTF8
	AMERICAN_AMERICA.US7ASCII
	AMERICAN_AMERICA.UTF8
	AMERICAN_AMERICA.WE8ISO8859P1
	AMERICAN_AMERICA.WE8ISO8859P15
	AMERICAN_AMERICA.WE8MSWIN1252
	ENGLISH_UNITED KINGDOM.WE8ISO8859P1
	ENGLISH_UNITED KINGDOM.WE8ISO8859P15
	ENGLISH_UNITED KINGDOM.WE8MSWIN1252
Finnish	FINNISH_FINLAND.WE8ISO8859P1
	FINNISH_FINLAND.WE8MSWIN1252
	FINNISH_FINLAND.WE8ISO8859P15
French	FRENCH_FRANCE.UTF8
	FRENCH_FRANCE.WE8ISO8859P1
	FRENCH_FRANCE.WE8ISO8859P15
	FRENCH_FRANCE.WE8MSWIN1252
German	GERMAN_GERMANY.UTF8
	GERMAN_GERMANY.WE8ISO8859P1
	GERMAN_GERMANY.WE8ISO8859P15
	GERMAN_GERMANY.WE8MSWIN1252
Greek	GREEK_GREECE.EL8MSWIN1253
Hebrew	HEBREW_ISRAEL.IW8MSWIN1255
	HEBREW_ISRAEL.UTF8
Hungarian	HUNGARIAN_HUNGARY.EE8MSWIN1250
Italian	ITALIAN_ITALY.UTF8
	ITALIAN_ITALY.WE8ISO8859P1
	ITALIAN_ITALY.WE8ISO8859P15
	ITALIAN_ITALY.WE8MSWIN1252
Japanese	JAPANESE_JAPAN.JA16EUC
	JAPANESE_JAPAN.JA16SJIS
	JAPANESE_JAPAN.UTF8
Korean	KOREAN_KOREA.KO16KSC5601
	KOREAN_KOREA.KO1616KSCCS
	KOREAN_KOREA.UTF8
Norwegian	NORWEGIAN_NORWAY.WE8ISO8859P1
	NORWEGIAN_NORWAY.WE8ISO8859P15
	NORWEGIAN_NORWAY.WE8MSWIN1252
Polish	POLISH_POLAND.EE8MSWIN1250
Portuguese	PORTUGUESE_PORTUGAL.WE8ISO8859P1
	PORTUGUESE_PORTUGAL.WE8ISO8859P15
	PORTUGUESE_PORTUGAL.WE8MSWIN1252
Romanian	ROMANIAN_ROMANIA.EE8MSWIN1250
Russian	RUSSIAN_CIS.CL8MSWIN1251
Slovak	SLOVAK_SLOVAKIA.EE8MSWIN1250

Table 1. Values

Language	NLS_LANG value
Spanish	SPANISH_SPAIN.UTF8
	SPANISH_SPAIN.WE8ISO8859P1
	SPANISH_SPAIN.WE8ISO8859P15
	SPANISH_SPAIN.WE8MSWIN1252
Spanish Latin American	LATIN AMERICAN SPANISH_AMERICA.WE8ISO8859P1
	LATIN AMERICAN SPANISH_AMERICA.WE8ISO8859P15
	LATIN AMERICAN SPANISH_AMERICA.WE8MSWIN1252
Swedish	SWEDISH_SWEDEN.WE8ISO8859P1
	SWEDISH_SWEDEN.WE8ISO8859P15
	SWEDISH_SWEDEN.WE8MSWIN1252
Turkish	TURKISH_TURKEY.WE8ISO8859P9

Backing up data

- Performing User Managed backups
- Performing RMAN backups
- Using Plug-in for FileSystem to back up Oracle ACFS

Performing User Managed backups

The User Managed backup procedure includes the following steps:

- Selecting data for a backup
- Selecting backup options
- Finalizing and submitting the job

Selecting data for a backup

You must use sets—Backup Selection Set, Backup Options Set, Schedule Set, Target Set, and Advanced Options Set—to create a backup job.

Backup Selection Sets are essential for Incremental and Differential Backups. Create the Backup Selection Set during a Full Backup, and use it for Full, Incremental, and Differential Backups. The backup job reports an error if you do not use a Selection Set for the Incremental or Differential Backup. For more information, see the *Quest NetVault Administrator's Guide*.

ℹ | **TIP:** To use an existing set, click **Create Backup Job**, and select the set from the **Selections** list.

1 In the Navigation pane, click **Create Backup Job**.

You can also start the wizard from the Guided Configuration link. In the Navigation pane, click **Guided Configuration**. On the **NetVault Configuration Wizard** page, click **Create backup jobs**.

2 In **Job Name**, specify a name for the job.

Assign a descriptive name that lets you easily identify the job when monitoring its progress or restoring data. The job name can contain alphanumeric and non-alphanumeric characters, but it cannot contain non-Latin characters. On Linux, the name can have a maximum of 200 characters. On Windows, there is no length restriction. However, a maximum of 40 characters is recommended on all platforms.

3 Next to the **Selections** list, click ➕ .

4 In the list of plug-ins, open **Plug-in *for Oracle***, and then open the database you want to back up.

5 If you have not configured the plug-in's authentication details, enter the required information when the prompt appears, and then click **OK**.

- **Account Name**: Specify an Oracle database user with **SYSDBA** privileges.
- **Password**: Enter the password associated with the name specified in the preceding field.

6   After the plug-in connects to the database and lists the supported file types for selection, select the data that you want to back up.

The plug-in supports the following file types for User Managed backups:

- **Parameter File**: Select this check box to include the **PFILE** or the **SPFILE** for the Oracle database. You can open this node and see the filename; however, file selection is not permitted at that level.

- **Control File**: When selected, the plug-in takes a snapshot of the Control File and saves it to the path and file specified in the **Control File Save Filename (Full Path)** field on the **Add**, or **Edit**, **Oracle Database** dialog box. The snapshot copy of the Control File is then backed up to the backup device.

- **Archive Logs**: Select this option to back up all the archived redo logs specified in the **Archive Log Destination Directory** field on the **User Managed Details** tab. The plug-in does not delete the backed-up archived redo logs automatically.

- **All Tablespaces**: Select this node to back up all the tablespaces within the database, or open it and select individual tablespaces to include in the backup. Currently, the plug-in does *not* support backing up individual datafiles for User Managed backup selections, but you can select an individual datafile for restoration.

- **External Configuration Files (for RMAN Backup Method only)**: This option applies to the RMAN backup method. If External Configuration Files are included in the selections for the User Managed backup method, a Warning message is shown in the NetVault binary logs. However, the backup job status is not set to **Backup Completed for Warnings** for this single warning. That is, the backup task ignores the logged warning.

- **Post-Backup Scripts**: This option applies to the RMAN backup method.

> **i** | **IMPORTANT:** Selecting individual datafiles for backup with the User Managed backup method results in a backup error and a message in the NetVault binary logs, **"Individual Datafile Level Backup is not supported by the User Managed Backup Method."**
>
> If the files stored in Oracle's ASM are selected for backup with the User Managed backup method, the backup job fails. This selection includes any datafile, SPFILE, Control File, or any other file stored in ASM. If ASM is enabled for this database, the RMAN backup method must be used.

7   Click **Save**, enter a name in the **Create New Set** dialog box, and click **Save**.

The name can contain alphanumeric and non-alphanumeric characters, but it cannot contain non-Latin characters. On Linux, the name can have a maximum of 200 characters. On Windows, there is no length restriction. However, a maximum of 40 characters is recommended on all platforms.

Inclusive Oracle tablespace backup selection

If a large Oracle database has been split over a number of separate backup jobs, the content for these jobs is typically explicitly defined; that is, tablespaces are explicitly or manually selected with a large green check mark. If new tablespaces are created within the Oracle Instance, they are not included in the existing backup jobs; they are excluded because the tablespaces have been manually selected while the new tablespaces have not been included in the list of tablespaces to back up.

To ensure that new tablespaces are included in the backups, create another inclusive backup job where the **All Tablespaces** node is explicitly or manually selected. This step allows newly created tablespaces to be backed up as part of the inclusive backup job.

Example

The database has the following tablespaces:

- **China_Sales**

- **Japan_Sales**

- **Korea_Sales**

- **LA_Sales**
- **London_Sales**
- **NY_Sales**
- **SD_Sales**
- **UK_Sales**

1. Create an **ASIA_TS** backup job that explicitly selects **China_Sales**, **Japan_Sales**, and **Korea_Sales** tablespaces.

2. Create a **US_TS** backup job that explicitly selects **LA_Sales**, **NY_Sales**, and **SD_Sales** tablespaces.

3. Create an **EMEA_TS** backup job that explicitly selects **London_Sales** and **UK_Sales** tablespaces.

4. Create an **All_TS** backup job that explicitly selects the **All Tablespaces** node and clears all the tablespaces included in the other backup jobs.

Selecting backup options

The next step involves creating the Backup Options Set or selecting an existing one.

> **i** | **TIP:** To use an existing set, in the **Plugin Options** list, select the set that you want to use.

1. Next to the **Plugin Options** list, click ✚ .

2. In the **Backup Method** section, select the **User Managed** option to perform a User Managed backup.

3. If applicable, select the following options:

   - **Include Read-Only and Offline Tablespaces**: By default, the plug-in does not back up read-only and offline tablespaces even if you selected them explicitly on the **NetVault Selections** page. Read-only and offline tablespaces typically do not require backing up unless the read-only data has been updated or the offline tablespaces have been brought online. Eliminating these tablespaces from the Full User Managed backups reduces the backup window. To include read-only and offline tablespace in the backup, select this option.

   - **Delete Archive Logs After Successful Backup**: If you want to delete the archive logs automatically after they are backed up, thus eliminating the need to delete them manually, select this option. This option is useful when you are backing up the archive logs to backup media.

     The plug-in backs up each log sequence number and then delete the file that it has backed up. If you have multiple archive destinations for your redo logs, the other copies of the same log sequence number are not deleted.

     > **i** | **IMPORTANT:** Selecting the **Delete Archive Logs After Successful Backup** option deletes the archive logs even if the backup job fails. Therefore, ensure that the option selected for **Incomplete Backup of ALL Items Selected** includes retaining a backup saveset to ensure that the archive logs can still be restored.

4. From the **Incomplete Backup of ALL Items Selected** list, select the default action for the plug-in.

   When multiple items are included in a backup and the plug-in is unable to back up *all* the items selected, the plug-in lets you specify what action the backup should take. For example, if a job includes multiple tablespaces and archive logs, and the backup of the archive logs is unsuccessful while the tablespaces are backed up successfully, you can specify what action the backup job should take.

   - **Complete with Warnings — Saveset Retained:** The job returns a status of **"Backup Completed with warnings"** and a backup saveset is created that includes the items that were successfully backed up.

   - **Complete without Warnings — Saveset Retained:** The job completes and returns a status of **"Backup Completed."** The errors are logged in the NetVault binary logs and ignored on the **Job Status** page. A backup saveset is created that includes the items that were backed up.

- **Fail — Saveset Retained:** The job returns a status of **"Backup Failed."** However, a backup saveset is generated that includes the items that were successfully backed up.

- **Fail — No Saveset Retained:** The job returns a status of **"Backup Failed"** and no saveset of backed-up objects is kept. That is, even if some of the objects were successfully backed up, the saveset is discarded.

5 In **Set Name**, specify a name for the set, and click **Save**.

The name can contain alphanumeric and non-alphanumeric characters, but it cannot contain non-Latin characters. On Linux, the name can have a maximum of 200 characters. On Windows, there is no length restriction. However, a maximum of 40 characters is recommended on all platforms.

Finalizing and submitting the job

1 Use the **Schedule**, **Target Storage**, and **Advanced Options** lists to configure any additional required options.

2 Click **Save** or **Save & Submit**, whichever is applicable.

> **i** | **TIP:** To run a job that you have already created and saved, select **Manage Job Definitions** in the Navigation pane, select the applicable job, and click **Run Now**.

You can monitor progress on the **Job Status** page and view the logs on the **View Logs** page. For more information, see the *Quest NetVault Administrator's Guide*.

Performing RMAN backups

The RMAN backup procedure includes the following steps:

- Selecting data for a backup

- Setting backup options

- Finalizing and submitting the job

Selecting data for a backup

You must use sets — Backup Selection Set, Backup Options Set, Schedule Set, Target Set, and Advanced Options Set — to create a backup job.

Backup Selection Sets are essential for Incremental and Differential Backups. Create the Backup Selection Set during a Full Backup, and use it for Full, Incremental, and Differential Backups. The backup job reports an error if you do not use a Selection Set for the Incremental or Differential Backup. For more information, see the *Quest NetVault Administrator's Guide*.

> **i** | **TIP:** To use an existing set, click **Create Backup Job**, and select the set from the **Selections** list.

1 In the Navigation pane, click **Create Backup Job**.

You can also start the wizard from the Guided Configuration link. In the Navigation pane, click **Guided Configuration**. On the **NetVault Configuration Wizard** page, click **Create backup jobs**.

2 In **Job Name**, specify a name for the job.

Assign a descriptive name that lets you easily identify the job when monitoring its progress or restoring data. The job name can contain alphanumeric and non-alphanumeric characters, but it cannot contain non-Latin characters. On Linux, the name can have a maximum of 200 characters. On Windows, there is no length restriction. However, a maximum of 40 characters is recommended on all platforms.

3   Next to the **Selections** list, click ✚ .

4   In the list of plug-ins, open **Plug-in *for Oracle***, and then open the database you want to back up.

5   If you have not configured the plug-in's authentication details, enter the required information when the prompt appears, and then click **OK**.

- **Account Name**: Specify an Oracle database user with **SYSDBA** privileges.

- **Password**: Enter the password associated with the name specified in the preceding field.

6   After the plug-in connects to the database and lists the supported file types for selection, select the data that you want to back up.

The plug-in supports the following file types for RMAN backups:

- **Database Node**: When this node is selected and no individual sub-nodes are cleared, the whole Oracle database is backed up with the equivalent RMAN **BACKUP DATABASE PLUS ARCHIVE LOG** command. Unless performing specialized individual backups that only include Parameter File, Control Files, or Archive Logs, Quest recommends that you select this node when performing Full and Incremental Backups.

- **Parameter File**: This node is only available for selection if the **Control File Autobackup Enabled** option on the **RMAN Details** tab of the **Add**, or **Edit**, **Oracle Database** dialog box is *not* selected. Select this node to include the **PFILE** or the **SPFILE** for the Oracle database in the backup. If the **Control File Autobackup Enabled** option or the **Use RMAN Commands to Backup SPFILE** option on the **Add**, or **Edit**, **Oracle Database** dialog box is enabled, the **Parameter File** node does not expand; otherwise, the Parameter File name is displayed when the **Parameter File** node is expanded.

- **Control File**: This node is only available for selection if the **Control File Autobackup Enabled** option on the **RMAN Details** tab of the **Add**, or **Edit**, **Oracle Database** dialog box is *not* selected. Select this node to perform a Manual Backup of the Control File. To ensure that a recent copy of the Control File is always available, include it with each RMAN-based backup. If you are not using a Recovery Catalog, you cannot restore a Manual Control File backup by using the plug-in. For more information on Manual Control File backups versus Control File Autobackups, see Manual Backups of Control File.

- **Archive Logs**: Because archived redo logs are essential for PIT Recovery and should be backed up regularly, Quest recommends that you select this node to include the Archive Logs in a Full or Incremental backup job. Additionally, the Archive Logs can be independently protected with the Archive Log backup type.

- **All Tablespaces**: Select this node to back up all the tablespaces within the database, or open it and select individual tablespaces to include in the backup. In addition, you can expand the individual tablespaces to select individual datafiles. This option is useful for backing up a subset of datafiles because performing whole-database backups takes longer than the allowed backup window.

- **External Configuration Files (for RMAN Backup Method only)**: Select this node to include the External Configuration Files in the backup, or open it and select individual Oracle Network Configuration or Oracle Password files for backup. Plug-in *for Oracle* uses Quest NetVault Plug-in *for FileSystem* (Plug-in *for FileSystem*) functionality to back up and restore the Oracle Network Configuration and Oracle Password files.

- **Post-Backup Scripts**: To run a post-backup RMAN script using the plug-in, open this node and select the individual script to be run. For more information on post-backup scripts, see Using RMAN Post-Backup Scripts.

  > **ⓘ** | **IMPORTANT:** The Database must be in an **OPEN** state to back up the tablespaces and datafiles. However, the Control Files, SPFILE, and Archive Logs can be backed up with the database in a **MOUNT** state.

7   Click **Save**, enter a name in the **Create New Set** dialog box, and click **Save**.

The name can contain alphanumeric and non-alphanumeric characters, but it cannot contain non-Latin characters. On Linux, the name can have a maximum of 200 characters. On Windows, there is no length restriction. However, a maximum of 40 characters is recommended on all platforms.

Setting backup options

The next step involves creating the Backup Options Set or selecting an existing one.

> **i** | **TIP:** To use an existing set, in the **Plugin Options** list, select the set that you want to use.

1 Next to the **Plugin Options** list, click ➕ .

2 In the **Backup Method** section, select the **RMAN** option to perform RMAN backups.

> **i** | **IMPORTANT:** For backups that use the RMAN backup method, do not select the **Ensure this backup is the first on the media** option on the **Target Storage** section. The RMAN backup method generates more than one data stream. If you select the **Ensure this backup is the first on the media** option for such backups, each data stream targets a separate piece of media and assumes that it is the first backup on the media.

3 Select the backup destination.

When **Flash Recovery Area** is enabled on the **Add**, or **Edit**, **Oracle Database** dialog box, the following options let you select the destination for the backup:

- **NetVault Media Manager**: Selecting this option sends the backup to whatever media you designated on the Target tab, such as a physical tape, a NetVault VTL stored on the disk, or a NetVault SmartDisk. When you select this option, the plug-in allocates "SBT_TAPE" channels to perform the backup.

- **Disk (Backup to FRA)**: Selecting this option sends the backup to the Oracle drive instead of the NetVault Media. When you select this option, the plug-in allocates DISK channels to perform the backup.

- **Both NetVault Media Manager and Disk (Backup to FRA then Backup FRA to NetVault Media)**: Selecting this option sends the backup to the Oracle drive and then to whatever media you designated on the Target tab in NetVault.

- **Delete File Copies from Disk After Successful Backup to NetVault Media Manager**: Available only when a FRA backup type is selected. When selected, RMAN deletes the recovery files from the FRA after a successful backup. Selecting this option is equivalent to the following RMAN command:

  ```
  BACKUP BACKUPSET ALL NOT BACKED UP SINCE TIME 'SYSDATE' DELETE INPUT;
  ```

  > **i** | **IMPORTANT:** If you select the **Delete File Copies from Disk After Successful Backup to NetVault Media Manager** option, RMAN deletes the recovery files in the FRA even if the backup job fails. Therefore, ensure that the option selected for **Incomplete Backup of ALL Items Selected** includes retaining a backup saveset to ensure that the backup to disk can still be restored.
  >
  > Selecting the **Backup Destination of Disk (Backup to FRA)** is not supported for **External Configuration Files** or **Parameter Files** that are manually input on the **Add**, or **Edit**, **Oracle Database** dialog box. This is not supported because the FRA is a portion of disk managed by Oracle. **External Configuration Files** and **Parameter Files** that are manually input on the **Add**, or **Edit**, **Oracle Database** dialog box are backed up through Plug-in for FileSystem functionality.

These options let you implement a media destination strategy that balances requirements while speeding up restores. For more information, see Defining a Backup Destination strategy. For instructions on enabling FRA, see the bullet item Flash Recovery Area Enabled in Configuring default settings.

4   Specify the backup type.

- **Full**

- **Incremental Level 0**

- **Level 1 Differential**

- **Level 1 Cumulative**

- **Archive Log**

- **Duplicate Database**

- **Flash Recovery Area** (only available for Oracle 10g and later)

- **Delete Recovery Files from Flash Recovery Area After Successful Backup**: Available only when the FRA backup type is selected. When selected, RMAN will delete the recovery files from the FRA after a successful backup. Selecting this option is equivalent to the following RMAN commands:

  ```
  BACKUP RECOVERY AREA DELETE INPUT

  BACKUP RECOVERY FILES DELETE INPUT
  ```

For more information on these backup types, see Defining an RMAN backup strategy.

> **i** | **IMPORTANT:** FRA backups fail if previous backups have not been performed with a **Backup Destination** of **Disk** or **Both NetVault Media Manager and Disk**. You must use the FRA as a Backup Destination before you can back up the FRA using the **Flash Recovery Area** backup type.
>
> When NetVault instructs backups to Disk (Backup to FRA), **Incremental Level 0** backups (INCR LVL0) are stored in separate backup sets. Additionally, the first **Incremental Level 1** backup (INCR LVL1)—Differential or Cumulative—generates the base **Incremental Level 0** backup copy image, as well as the **Incremental Level 1** backup copy image. Subsequent **Incremental Level 1** backups generate only the necessary Level 1 backup copy image update. For Incremental Backups to Disk (Backup to FRA), NetVault uses the *Oracle-recommended strategy* backup copy image method to maintain incrementally updated copies of the datafiles included in the selection set. For **Incremental Level 0** to Disk (Backup to FRA), NetVault instructs Oracle to perform a backupset backup type.

5   Set the RMAN backup options.

- **Delete Archive Logs After Successful Backup**: If you want to delete the archive logs automatically after they are backed up, thus eliminating the need to delete them manually, select this option. This option is useful when backing up the archive logs from disk to NetVault Media Manager. This option is equivalent to the RMAN command:

  ```
  BACKUP ARCHIVELOG ALL DELETE ALL INPUT;
  ```

  The plug-in backs up each log sequence number and then delete the file that it has backed up. If you have multiple archive destinations for your redo logs, the backed-up archived redo logs are deleted from all log-archiving destinations.

  > **i** | **IMPORTANT:** Selecting the **Delete Archive Logs After Successful Backup** option deletes the archive logs even if the backup job fails. Therefore, ensure that the option selected for **Incomplete Backup of ALL Items Selected** includes retaining a backup saveset to ensure that the archive logs can still be restored.

- **Force Additional Log Switch Before Archive Logs Backup**: Selecting this option ensures that another log switch is performed before the backup of the archived redo logs is started by issuing the SQL statement:

  ```
  ALTER SYSTEM ARCHIVE LOG CURRENT
  ```

- **Skip Offline Tablespaces**: Select this option if you want the plug-in to ignore the offline tablespaces and omit them from the backup.

- **Skip Read Only Tablespaces**: Back up read-only tablespaces after an update is performed, but you can skip them in subsequent backups to reduce the size of the backup. Select this option if you want the plug-in to ignore read-only tablespaces and omit them from the backup.

- **Skip Inaccessible Tablespaces**: Operations within Oracle might render a tablespace inaccessible. For example, if a datafile has been removed at the OS level, but not dropped inside the Oracle database or if a raw device mount point link is dismounted, the tablespace might become inaccessible. Select this option if you want the plug-in to skip inaccessible databases that were selected on the **NetVault Selections** page without causing the backup job to fail.

- **Skip Inaccessible Archive Logs**: Operations within Oracle might render an archive log inaccessible. For example, if an archive log file has been manually moved or renamed, it is not accessible. Select this option if you want the plug-in to skip inaccessible archive logs that were selected on the **NetVault Selections** page without causing the backup job to fail.

- **User Defined Backup Tag**: Select this option if you want to create a tag name for the backup set. If you do not specify a tag, RMAN creates a default tag for backups—except for Control File Autobackups—in the format TAGYYYYMMDDTHHMMSS. You can specify the tag rather than the filename when running the **RESTORE** command. Tags are stored in uppercase, regardless of the case used when entering them. The maximum length of a tag is 30 bytes.

  Additionally, use only characters that are allowed in filenames on the target file system. For example, ASM does not support the use of hyphens (-) in filenames that it uses internally, so you cannot use a hyphen in the tag name, such as, **weekly-incremental**, if you are storing backups in ASM disk groups.

  For more information on using this option in a Data Guard environment, see Creating a User Defined Backup Tag.

- **Files per Backup Set**: Backup information is stored in a logical structure called *backup set*. A backup set contains the data from one or more datafiles, archived redo logs, Control Files, or SPFILE. Datafiles and archive logs cannot be combined together in the same backup set. By default, each backup set contains 4 or fewer datafiles, or 16 or fewer archive logs. Use the **For Tablespaces/Datafiles** and **For Archive Logs** fields to specify the maximum number of files to include in each backup set created by the plug-in.

  If you are using deduplication, the default value of one ("1") requires the plug-in to complete the backup process in a specific order.

  If you change the setting to zero ("0"), the plug-in divides files among backup sets to make optimal use of the available channels. The total number of files to be backed up is divided by the number of channels. If the result is less than 64, that value is the number of files placed in each backup set. Otherwise, 64 files are placed in each backup set.

- **Number of Channels**: You can control the degree of *parallelism* within a backup job by allocating the desired number of channels. Allocating multiple channels simultaneously allows a single job to read or write multiple backup sets in parallel. The number of channels should be less than or equal to the number of drives for the backup devices specified on the **Target** tab. For example, if you are targeting a VTL configured with two drives, the number of channels should be 2 or less. Use the **For Tablespaces/Datafiles** and **For Archive Logs** fields to specify the number of channels.

- **Channel Rate**: Use these settings to specify a job-specific RATE for the channels that are used during backups. RATE is part of the **ALLOCATE CHANNEL** command. While defining this RATE does not determine the actual RATE used, it does establish the maximum level for transfers, which prevents RMAN from consuming excessive bandwidth and then degrading performance. Use the **For Tablespaces/Datafiles** and **For Archive Logs** fields, and their associated unit settings, to specify the upper-limits. By default, the unit field is blank, which represents bytes. You can also select **K** (kilobytes), **M** (megabytes), or **G** (gigabytes).

- **Backup Piece Max Size (0=max)**: Use these settings to specify a job-specific backup piece maximum size (MAXPIECESIZE) for the channels that are used during backups. By default, the backup piece maximum size is unlimited if not otherwise configured. To limit the size of the backup pieces generated by Oracle RMAN, set an upper limit in the Backup Piece Max Size (MAXPIECESIZE) settings. To specify the upper limit, use the **For Tablespaces/Datafiles** and **For Archive Logs** fields and their associated unit settings. In the unit field, select **G** (gigabytes), **M**

(megabytes), **K** (kilobytes), or **B** (bytes). For example, to limit the backup piece size to 2 GB or less, enter value 2 and select unit G in the Backup Piece Max Size option.

> **i** | **NOTE:** To notify Oracle RMAN to use the default configured value, enter **0** in the **Backup Piece Max Size (0=max)** option.
>
> Backup Piece Max Size (MAXPIECESIZE) is a clause of the RMAN **ALLOCATE CHANNEL** command. If you enter a value other than 0, the MAXPIECESIZE clause is included in the A**LLOCATE CHANNEL** command. If a value of 0 is entered, the MAXPIECESIZE clause is excluded in the RMAN **ALLOCATE CHANNEL** command. Here, backup piece maximum size is unlimited if not otherwise configured.

- **Backup as Compressed Backupset**: To add the **"AS COMPRESSED BACKUPSET"** clause to the RMAN **BACKUP** command, select this check box. It allows RMAN to perform binary compression of backup sets. The compressed backup sets do not require any additional steps during recovery.

- **For Archive Logs Use Backup Not Backed up (number of times):** Select this option if you want to skip an Archived Log backup after a specified number of backups have occurred, and then enter the number of backups after which archive logs should be omitted from the backup job.

- **Backup <selections> Not Backed up Since (days ago)**: Select this option if you want to force the plug-in to resume backups automatically after an interruption occurs. Additionally, you can specify the number of days—the default is 1; the range is 0 through 90—to check backward since the backup was last completed. This feature is useful if your backups are large and prone to possible interruptions.

6   From the **Incomplete Backup of ALL Items Selected** list, select the default action for the plug-in.

When multiple items are included in a backup and the plug-in is unable to back up *all* the items selected, even if RMAN successfully cataloged some of the selected items, the plug-in lets you specify what action the backup should take. For example, if a job includes multiple tablespaces and archive logs, and the backup of the archive logs is unsuccessful while the tablespaces are backed up successfully, you can specify what action the backup job should take.

- **Complete with Warnings — Saveset Retained:** The job returns a status of **"Backup Completed with warnings"** and a backup saveset is created that includes the items that were successfully backed up.

- **Complete without Warnings — Saveset Retained:** The job completes and returns a status of **"Backup Completed."** The errors are logged in the NetVault binary logs and ignored on the **Job Status** page. A backup saveset is created that includes the items that were backed up.

- **Fail — Saveset Retained:** The job returns a status of **"Backup Failed."** However, a backup saveset is generated that includes the items that were successfully backed up.

- **Fail — No Saveset Retained:** The job returns a status of **"Backup Failed"** and no saveset of backed-up objects is kept. That is, even if some of the objects were successfully backed up, the saveset is discarded.

7   Set the **Generate RMAN Script** options.

Intended for the advanced DBA, the **Generate RMAN Script** feature lets you generate most of an RMAN script through the WebUI. Later, before using the script, you can edit the script to add more options or parameters. With this option, you do not need to create the script from scratch, which minimizes the risk of syntax errors in the scripts and that backups are not performed correctly.

When the **Output RMAN Script to File** option is selected, the commands that the plug-in automatically sends to RMAN are stored in a file in a directory specified on the Configure dialog box. The Oracle DBA can then edit the script as needed and run the script through the RMAN CLI. Backup jobs that are performed by using the generated RMAN scripts are stored in the NetVault Job database.

You can set the following options for the **Generate RMAN Script** feature:

- **Output RMAN Script to File**: Select this option to use the feature and enable access to the remaining options, and then enter the name of the file to which the RMAN script is written.

- **RMAN Script Directory**: Enter the name of the directory where the RMAN scripts are saved. The **Generate Script Default Directory** field defaults to the directory specified on the plug-in's Configure dialog box. However, you can modify the field on a per-backup-job basis. The full path where the RMAN script is saved is built by concatenating the **RMAN Script Directory** with the filename provided in the **Output RMAN Script to File** field.

- **If Existing, Overwrite Destination File**: Select this option to allow the destination file where the RMAN script is output to be overwritten if the file exists. If the file exists and this option is not selected, **Generate RMAN Script** fails and does not write output to the existing file. Select this option only if you are sure that you want to overwrite an existing file.

- **Generate RMAN Script and Submit Job**: With this option selected, in addition to outputting the RMAN script to the specified file, the NetVault Job is scheduled and submitted. Clearing this option ensures that the plug-in just generates the script, which lets you edit and run the script outside of NetVault.

- **Do not Include Connect Strings:** To exclude connection strings from the file because they might include Oracle passwords, select this option. Selecting this option ensures that Oracle passwords are not included in the RMAN script file.

8   In **Set Name**, specify a name for the set, and click **Save**.

The name can contain alphanumeric and non-alphanumeric characters, but it cannot contain non-Latin characters. On Linux, the name can have a maximum of 200 characters. On Windows, there is no length restriction. However, a maximum of 40 characters is recommended on all platforms.

Finalizing and submitting the job

1   Use the **Schedule**, **Target Storage**, and **Advanced Options** lists to configure any additional required options.

2   Click **Save** or **Save & Submit**, whichever is applicable.

> **i** | **TIP:** To run a job that you have already created and saved, select **Manage Job Definitions** in the Navigation pane, select the applicable job, and click **Run Now**.

You can monitor progress on the **Job Status** page and view the logs on the **View Logs** page. For more information, see the *Quest NetVault Administrator's Guide*.

Using Plug-in *for FileSystem* to back up Oracle ACFS

Oracle Automatic Storage Management (ASM) Cluster File System extends the Oracle ASM functionality to support non-database files. These files include Oracle binaries, trace files, alert logs, report files, and other application data files.

The non-database files cannot be backed up using RMAN, but you can use the Plug-in for FileSystem to back up the data stored in an Oracle ACFS volume.

1   Mount the ACFS volume.

2   In the Navigation pane, click **Create Backup Job**.

You can also start the wizard from the Guided Configuration link. In the Navigation pane, click **Guided Configuration**. On the **NetVault Configuration Wizard** page, click **Create backup jobs**.

3   In **Job Name**, specify a name for the job.

Assign a descriptive name that lets you easily identify the job when monitoring its progress or restoring data. The job name can contain alphanumeric and non-alphanumeric characters, but it cannot contain non-

Latin characters. On Linux, the name can have a maximum of 200 characters. On Windows, there is no length restriction. However, a maximum of 40 characters is recommended on all platforms.

4   Next to the **Selections** list, click ✚ .

5   On the **NetVault Selections** page, enter a name for the set in the **Backup Selection Set** box, and open the NetVault Client that is configured as the Oracle Database Server.

6   In the list of plug-ins, open **Plug-in for FileSystem**.

7   Select the ACFS volume, or open the ACFS node and select the required items.

8   Configure the remaining backup options.

    For more information, see the *Quest NetVault Plug-in for FileSystem User's Guide*.

9   Use the **Schedule**, **Target Storage**, and **Advanced Options** lists to configure any additional required options.

10 Click **Save** or **Save & Submit**, whichever is applicable.

> **i** | **TIP:** To run a job that you have already created and saved, select **Manage Job Definitions** in the Navigation pane, select the applicable job, and click **Run Now**.

You can monitor progress on the **Job Status** page and view the logs on the **View Logs** page. For more information, see the *Quest NetVault Administrator's Guide*.

Using the Oracle Flashback Database

- Flashback Database: an overview
- Viewing Flashback Database status
- Flashback Database options
- Flashback Database restrictions
- Restoring data with the plug-in and Flashback Database

Flashback Database: an overview

Introduced in Oracle 10g, Flashback Database lets you rewind an Oracle database to a previous time to correct problems caused by logical data corruptions or user errors as long as the datafiles are intact. Flashback Database is not true media recovery because it does not involve restoring physical datafiles. Flashback Database is preferable to using the **RESTORE** and **RECOVER** commands in some cases because it is faster and easier and does not require restoring the whole database.

For more information on Flashback Database, see *Configuring Oracle Flashback Database and Restore Points* in the *Oracle Database Backup and Recovery User's Guide*.

Flashback Logs

Flashback Database has its own logging mechanism. Flashback Logs are generated and stored in the FRA. After Flashback Database is enabled, altered datafile data blocks are copied into the Flashback Logs. These data blocks can be used later to reconstruct the datafile contents. Because Flashback Logs are captured at regular time intervals, the data blocks restored from the Flashback Logs are those blocks stored most immediately before the desired target time. After Flashback Logs are applied, the Redo Log is reapplied to Complete Recovery to the target time.

Viewing Flashback Database status

To view database details about a previously configured database, including whether FRA and Flashback Database are enabled, perform the following steps.

1   In the Navigation Pane, click **Create Backup Job**, and click ✚ next to the **Selections** list.

2   In the selection tree, open the applicable client node.

3   Double-click **Plug-in** *for Oracle*, click the applicable database, and select **View Details** from the context menu.

    The Details dialog box includes the following details:

- **Flash Recovery Area Destination**: When FRA is enabled, this option displays the destination of the FRA.

- **Flashback Database Enabled**: Indicates **YES** if the Flashback Database feature is enabled or **NO** if not.

Additionally, backups done with plug-in show a Flashback Database node for the plug-in on the **Create Selection Set** page when you create a restore job. The **Flashback Database** node lets you perform a Flashback Database instead of a restore or recovery.

4 To close the dialog box, click **OK**.

Flashback Database options

Selecting Flashback Database in the plug-in is equivalent to the RMAN or SQL **FLASHBACK DATABASE** command.

Flashback Database has the following options:

- **To Restore Point**: Flashes the database to a specific restore point. A restore point is a named point that can be used as a flashback target. When you create a restore point, you are naming the current System Change Number (SCN). You can create up to 2048 named restore points and these points can either be **normal** or **guaranteed**. A guaranteed restore point is used in a Flashback Database.

  Restore points can be created using the **CREATE RESTORE POINT** command. For more information, see *Creating Normal and Guaranteed Restore Points* in the *Oracle Database Backup and Recovery User's Guide*.

- **To Time**: Reverts the database back to its state at the time specified.

- **To SCN**: Reverts the database back to its state at the specified SCN.

- **Before Time**: Reverts the database back to its state one second before the specified timestamp.

- **Before SCN**: Reverts the database back to its state at the system change number just preceding the specified SCN.

- **Before Reset Logs** (available for Oracle 10.2.x and later versions only): Rewinds the database back to the SCN just before the **RESETLOGS** happened.

Flashback Database restrictions

There are two restrictions on selecting Flashback Database:

- Whole Database and Flashback Database nodes cannot be selected at the same time. If selected, the restore job fails and the NetVault binary logs show an error message:

  Cannot select 'Whole Database' and 'Flashback Database' at the same time.

- On the **Create Selection Set** page, Flashback Database cannot be selected at the same time as other nodes such as Parameter File, Control Files, Tablespaces, and Datafiles. If selected, the restore job fails and the NetVault binary logs show an error message:

  Restore Selections Error. Cannot select 'Flashback Database' and Individual Datafiles at the same time.

Restoring data with the plug-in and Flashback Database

1   In the Navigation pane, click **Create Restore Job**.

2   To filter the items displayed in the saveset table, click **Filter** ▼.

3   In the saveset table, select any RMAN-based backup saveset, and click **Restore**.

4   On the **Create Selection Set** page, select the **Flashback Database** node.

5   Click ✏, and then click the **Restore Options** tab.

6   Select **Whole Database Restore**, **Mount Database Before Restore**, and **Shutdown Abort Database First**.

> **i** | **NOTE:** The database must be in a mounted state to perform a Flashback Database.

7   On the **Post Recovery** tab, select **Open Database after Recovery** and **Read Write and Reset Logs**.

8   On the **Flashback Database** tab, select the applicable **Flashback Point Type**.

   For example, select **To Time**, and input the time to which the database should be flashed back.

9   Click **OK** to save the settings, and then click **Next**.

10 In **Job Name**, specify a name for the job if you do not want to use the default setting.

11 In the **Target Client** list, select the machine on which you want to restore the data.

12 Use the **Schedule**, **Source Options**, and **Advanced Options** lists to configure any additional required options.

13 Click **Save** or **Save & Submit**, whichever is applicable.

Restoring data

- Restoring and recovering data: an overview
- Performing User Managed restores
- Using advanced User Managed restore procedures
- Performing RMAN restores
- Using RMAN types of recovery in a non-RAC environment
- Using advanced procedures with RMAN restores

Restoring and recovering data: an overview

Reconstructing the contents of all or part of a database from a backup typically involves two phases: Restore and Recovery. Restoring is the process of retrieving a copy of the datafile from a backup. Recovery is the process of reapplying changes to the file since the backup from the archived and online redo logs, to bring the database to a desired SCN—typically the present, or the point of failure.

Based on your backup method, follow the instructions in the relevant topic for successfully performing data restoration using Plug-in *for Oracle*.

Understanding User Managed recovery

When a User Managed backup strategy has been implemented, you must manage the recovery process. This process includes the following tasks:

- Determining what to recover.
- Restoring the needed files with the plug-in.
- Performing the recovery process manually outside the plug-in by issuing a series of **SQL*Plus** commands.

The following topics provide an overview of the User Managed recovery process. For more information, see *About User-Managed Restore Operation* in the *Oracle Database Backup and Recovery Advanced User's Guide* or to *Performing User-Managed Backup and Recovery* in the *Oracle Database Backup and Recovery User's Guide*.

- Determining which datafiles require recovery
- Recovering datafiles and archive logs
- Restoring Control Files

Determining which datafiles require recovery

When a media failure or data corruption has occurred, you can use the following **SQL*Plus** query to determine which datafiles need recovery. This command works only when the database is in an **OPEN** state:

```
SELECT FILE#, ERROR, ONLINE_STATUS, CHANGE#, TIME FROM V$RECOVER_FILE;
```

Recovering datafiles and archive logs

After identifying the damaged datafiles, restore the last Full Backup of the datafiles and the backup of all archive logs generated since the last backup until the target recovery time. When you restore the datafiles, the existing files are replaced with the backed-up copy. Conversely, when you restore the archive logs, these files are copied to the original location so that they are available to the database during the recovery process.

The recovery of the damaged datafiles and the archive logs includes the following steps:

1 Connect to the database with SYSDBA privileges by using the **SQL*Plus** utility.

```
SQLPLUS SYS AS SYSDBA
```

2 If the database is open, use the following command to alter all the tablespaces that you want to recover:

```
ALTER TABLESPACE <Tablespace Name> OFFLINE;
```

3 Use the plug-in to restore the damaged datafiles from the latest User Managed Full Backup saveset.

To restore datafiles to a different location, specify the new path during restore.

4 Use the plug-in to restore all the archive logs backups that have been completed since the User Managed Full Backup.

Restore the logs to the primary destination directory. If space is limited, use the **Rename** option to restore the logs to a different directory. For more information, see Using advanced User Managed restore procedures.

5 After the restoration is complete, use **SQL*Plus** to perform the recovery manually by issuing the following command for every tablespace that needs recovering:

```
RECOVER TABLESPACE <Tablespace Name>
```

6 Alter the recovered tablespaces online with the following **SQL*Plus** command:

```
ALTER TABLESPACE <Tablespace Name> ONLINE;
```

Restoring Control Files

You can recover a Control File using one of the following options:

- **Restoring a Lost Copy of a Multiplexed Control File**: Use this procedure to recover a database if a permanent media failure has damaged one or more Control Files of a database, and at least one Control File has not been damaged by the media failure. For more information, see *Restore Lost Copy of a Multiplexed Control File* in the *Oracle Database Backup and Recovery Advanced User's Guide* or to *Responding to the Loss of a Subset of the Current Control Files* in the *Oracle Database Backup and Recovery User's Guide for Oracle 11g*.

- **Restoring a Control File from Backup After Loss of All Current Control Files**: Use this procedure to restore a Backup Control File if a permanent media failure has damaged all Control Files of a database and you have a backup of the Control File.

Restoring a Control File from backup after loss of all current Control Files

When a Control File is inaccessible, you can start the Instance, but not mount the database. If you attempt to mount the database when the Control File is unavailable, you get the following error message:

```
ORA-00205: error in identifying Control File, check alert log for more info
```

You cannot mount and open the database until the Control File is accessible.

The plug-in saves a copy of the snapshot of the Control File to the **Control File Save Filename (Full Path)** specified on the **Add**, or **Edit**, **Oracle Database** dialog box. You can copy the Control File from this location if you lose all current Control Files. If the snapshot is not available in the **Control File Save Filename (Full Path)**, you can restore it from the backup by performing the following steps:

1   Use the plug-in to restore the Control File from the latest Control File backup.

   The plug-in restores it to the **Control File Save Filename (Full Path)**. The restored file has a **".SAV"** extension.

2   Connect to the database with SYSDBA privileges by using the **SQL*Plus** utility.

   ```
   SQLPLUS SYS AS SYSDBA
   ```

3   Shut down the database.

4   Rename and copy this restored file to the Control File name and location specified in the Oracle parameter file.

5   Mount the database.

   ```
   STARTUP MOUNT;
   ```

6   To recover the database to a point equivalent to the current SCN stored in the datafiles, perform the recovery by using the **USING BACKUP CONTROLFILE** command.

   More recovery is needed to recover the database to the point that equals the current SCN stored in the datafiles. For example, when restoring from an older backup, the Control File contains a different SCN from the current datafiles, so it does not know which log sequence is archived and which is not.

   For example:

   ```
   RECOVER DATABASE USING BACKUP CONTROLFILE;
   ```

   For more information on restoring a Backup Control File, see *Restoring a Backup Control File to the Default Location* in the *Oracle Database Backup and Recovery Advanced User's Guide*.

7   After recovery is finished, open the database with the **RESETLOGS** option.

   ```
   ALTER DATABASE OPEN RESETLOGS;
   ```

Understanding RMAN recovery

Review the following topics regarding RMAN recovery:

- How RMAN backups are selected for use in recovery operations
- Choosing between Incremental Backups and Redo Logs
- Choosing the restore source
- Datafile media recovery
- Block Media Recovery

How RMAN backups are selected for use in recovery operations

When the plug-in performs RMAN-based restores, RMAN itself is performing the restores and recovery operations. RMAN uses the record of available backups in the RMAN repository to select the best available backups for use in the restore operation. This step simplifies the recovery process by eliminating the need for less-experienced personnel to determine the order in which backups should be restored.

Choosing between Incremental Backups and Redo Logs

If RMAN has a choice between applying an Incremental Backup or applying redo logs to the restored datafiles to meet a recovery objective, it always chooses an Incremental Backup. If overlapping levels of Incremental Backup are available, RMAN automatically chooses the one covering the longest period.

RMAN automatically restores the datafiles that it needs from available backups, applies Incremental Backups to the datafiles if they are available, and then applies archive logs.

Choosing the restore source

When FRA is enabled, the plug-in lets you define a media destination strategy that balances requirements while speeding up restores by specifying the **Destination Backup Options** on the **Backup Options** tab. These options let you select the destination for the backup. They include:

- **NetVault Media Manager**
- **Disk (Backup to FRA)**
- **Both NetVault Media Manager and Disk (Backup to FRA then Backup FRA to NetVault Media)**

The plug-in provides you with the option during the restore process to define the **Restore Source** or location that RMAN should use as the source for the restore. This option lets you speed up restores by ensuring that RMAN restores from the FRA or from the NetVault media. With this option, you can:

- Restore from the FRA even if the backup is also available on NetVault media such as disk-based media, VTL, or tape.
- Restore from the NetVault media when the FRA is suspected to have a media failure or data corruption and you do not want to risk restarting the restore and recovery process due to RMAN's inability to restore from the FRA.

Additionally, you can use these same **Restore Source** options when you have performed CLI-based backups to Disk by allocating a channel to Disk but you want to use the plug-in to perform the restore.

The following Restore Source options are available:

- **NetVault Media Manager**: When selected, RMAN opens an "SBT_TAPE" channel and reads the backup files from the backup media that is managed by NetVault and specified in the **Device options** section of the **Target** tab. Select this option if an FRA-backup-destination strategy has not been implemented or the Backup Destination of **NetVault Media Manager** was selected for all backups.
- **Disk (Restore from FRA or Disk)**: When selected, RMAN only opens a DISK device type and restores the backup files from the FRA or the OS-specific directory specified when configuring RMAN's DISK device type. This option is the default.
- **Both NetVault Media Manager and Disk (Restore from both NetVault Media and (FRA or Disk))**: When selected, RMAN opens an "SBT_TAPE" channel and a DISK channel, which allows RMAN the flexibility to choose the best source for the recovery files. Select this option if your backup destination strategy includes the **Both NetVault Media Manager and Disk** option or you have chosen to store backup recovery files in the FRA while performing **Flash Recovery Area Backups** on a different, independent schedule.

Datafile media recovery

Datafile media recovery is the application of online logs, archived redo logs, or Incremental Backups to a restored datafile to update it to the current time or some other specified time. Datafile media recovery, often called Recovery, is either Complete Recovery or Point-in-Time (PIT) Recovery, also called Incomplete Recovery. PIT Recovery is one way to respond to a data loss caused by a user error, such as dropping a table, or logical corruption that goes unnoticed for some time.

Complete Recovery vs. Point-in-Time (PIT) Recovery

Complete Recovery recovers the database to the most recent time without the loss of any committed transactions. With PIT Recovery, you can recover the database to a specific point. The plug-in supports PIT Recovery based on

time, log sequence number, or system change number. Currently, the plug-in supports only Database Point-in-Time Recovery (DBPITR).

- **System Change Number (SCN) Based Point-in-Time Recovery**: When an SCN is specified during PIT Recovery, RMAN recovers up to, but not including, the specified SCN. For example, if SCN 1000 is specified, recovery up to SCN 999 is performed.

- **Log Sequence Number (LSN) Based Point-in-Time Recovery**: When the exact time of the data corruption or failure is not known, specifying a Log Sequence Number that contains the target SCN is a viable option. RMAN recovers through the specified log. **V$LOG_HISTORY** can be queried to view the logs that have been archived to identify the appropriate log sequence number and thread.

- **Time Based Point-in-Time Recovery**: Time-based PIT Recovery is useful when the time that the data corruption occurred is known. For example, if a developer dropped a table at 6:00 a.m., PIT Recovery can be performed with a stop time of 5:55 a.m. The plug-in recovers up to, but not including, the specified time.

For more information on PIT Recovery and database incarnations, see *Performing Database Point-in-Time Recovery* in the *Oracle Database Backup and Recovery Basics* guide.

Block Media Recovery

Block Media Recovery is only available in the Oracle Enterprise Edition. This feature reduces downtime by letting you recover only the blocks that were corrupted instead of restoring and recovering the entire datafile. Block Media Recovery is most useful for physical corruption problems that involve a small, well-known number of blocks. Block-level data loss usually results from intermittent, random I/O errors that do not cause widespread data loss, and memory corruptions that get written to disk. Block Media Recovery is not intended for cases where the extent of data loss or corruption is unknown and the entire datafile requires recovery. In such cases, Datafile Media Recovery is the best solution.

You can use Block Media Recovery to recover one or more corrupt data blocks within a datafile. Block Media Recovery provides the following advantages over Datafile Media Recovery:

- It lowers the MTTR because only blocks needing recovery are restored and recovered.

- It lets the affected datafiles remain online during recovery.

Without Block Media Recovery, if even a single block is corrupt, you must take the datafile offline and restore the datafile from a backup. You must apply all redo logs generated for the datafile after the backup was created. The entire file is unavailable until media recovery completes. With Block Media Recovery, only the blocks actually being recovered are unavailable during the recovery.

Prerequisites

In addition to running the Oracle Enterprise Edition, Oracle requires the following prerequisites for Block Media Recovery to be met as defined in *Performing Block Media Recovery* of the *Oracle Database Backup and Recovery User's Guide*.

- The target database must run in **ARCHIVELOG** mode and be open or mounted with a current Control File.

- The target database must not be a standby database.

- The backups of the datafiles containing the corrupt blocks must be Full or Incremental Level 0 backups and not proxy copies.

- RMAN can use only archived redo logs for the recovery. RMAN cannot use Incremental Level 1 Incremental Backups. Block Media Recovery cannot survive a missing or inaccessible archived redo log, although it can sometimes survive missing redo records.

- Flashback Database must be enabled on the target database for RMAN to search the flashback logs for good copies of corrupt blocks. When flashback logging is enabled and contains older, uncorrupted versions of the corrupt blocks, RMAN can use these blocks, possibly speeding up the recovery.

Identifying corrupted blocks

The **V$DATABASE_BLOCK_CORRUPTION** view displays blocks marked corrupt by database components such as RMAN commands, **ANALYZE**, and SQL queries. Physical corruption, sometimes called media corruption,

results in rows being added to this view. For example, the database does not recognize the block: the checksum is invalid, the block contains all zeros, or the block header is fractured.

In addition to being reported in **V$DATABASE_BLOCK_CORRUPTION**, block corruption is also reported in the following locations:

- Results of the **LIST FAILURE**, **VALIDATE**, or **BACKUP … VALIDATE** command

- Error messages in standard output

- Oracle Alert Log

- User trace files

- Results of the SQL commands **ANALYZE TABLE** and **ANALYZE INDEX**

- Results of the **DBVERIFY** utility

- Third-party media management output such as NetVault.

  For example, you may discover the following messages in a user trace file:

  ORA-01578: ORACLE data block corrupted (file # 7, block # 3)

  ORA-01110: data file 7: '/oracle/oradata/trgt/tools01.dbf'

  ORA-01578: ORACLE data block corrupted (file # 2, block # 235)

  ORA-01110: data file 2: '/oracle/oradata/trgt/undotbs01.dbf'

Performing User Managed restores

A standard User Managed restore with Plug-in *for Oracle* includes the following steps:

- Preparing the database for recovery
- Selecting data for a restore
- Finalizing and submitting the job
- Recovering the database
- Opening the database for use

Preparing the database for recovery

You do not have to take a complete Oracle database offline to restore selected tablespaces, but you must take each tablespace that you want to restore offline before performing the restore.

1 Connect to the database with SYSDBA privileges by using the **SQL*Plus** utility.

   ```
   SQLPLUS SYS AS SYSDBA
   ```

2 If the database is open, take all tablespaces that contain damaged datafiles offline by running the following command:

   ```
   ALTER TABLESPACE <Tablespace Name> OFFLINE IMMEDIATE;
   ```

Selecting data for a restore

1   In the Navigation pane, click **Create Restore Job**.

2   To filter the items displayed in the saveset table, click **Filter** ▼.

    The table displays the saveset name (job title and saveset ID), creation date and time, size, and catalog status. By default, the list is sorted alphabetically by creation date.

3   In the saveset table, select the applicable saveset.

    When you select a saveset, the following details are displayed in the **Saveset Information** area: Job ID, job title, server name, client name, plug-in name, saveset date and time, retirement setting, Incremental Backup or not, Archive or not, and saveset size.

4   Select one of the following options:

    ▪   **Restore All Using Defaults:** To restore using the prepopulated defaults, click this button, and proceed to Finalizing and submitting the job.

    ▪   **Restore:** To use the **Create Selection Set** page to select the items that you want to restore, click this button and proceed to the next step.

5   On the **Create Selection Set** page, select the data that you want to restore.

    Depending on the items selected for the backup, the saveset includes:

    ▪   **Parameter File**

    ▪   **All Tablespaces**

    ▪   **Backup Control File**

    ▪   **Archive Logs**

6   Open the applicable items, and select the data that you want to include in the restore.

    > **NOTE:** For User Managed restores, you do not have to set any other options; the **Restore Options** tab does not contain any fields.

Finalizing and submitting the job

The final steps include setting additional options on the Schedule, Source Options, and Advanced Options pages, submitting the job, and monitoring the progress through the Job Status and View Logs pages. These pages and options are common to all NetVault Plug-ins. For more information, see the *Quest NetVault Administrator's Guide*.

1   Click **OK** to save the settings, and then click **Next**.

2   In **Job Name**, specify a name for the job if you do not want to use the default setting.

    Assign a descriptive name that lets you easily identify the job when monitoring its progress. The job name can contain alphanumeric and non-alphanumeric characters, but it cannot contain non-Latin characters. On Linux, the name can have a maximum of 200 characters. On Windows, there is no length restriction. However, a maximum of 40 characters is recommended on all platforms.

3   In the **Target Client** list, select the machine on which you want to restore the data.

    > **TIP:** You can also click **Choose**, and then locate and select the applicable client in the **Choose the Target Client** dialog box.

4   Use the **Schedule**, **Source Options**, and **Advanced Options** lists to configure any additional required options.

5   Click **Save** or **Save & Submit**, whichever is applicable.

    You can monitor progress on the **Job Status** page and view the logs on the **View Logs** page. For more information, see the *Quest NetVault Administrator's Guide*.

Recovering the database

After restoring the datafiles, you can choose to recover the database, tablespace, or datafile.

1 Connect to the database with SYSDBA privileges by using the **SQL*Plus** utility.

```
SQLPLUS SYS AS SYSDBA
```

2 Run one of following commands based on the required result:

- To recover a whole database, enter:

    ```
    RECOVER DATABASE
    ```

- To recover a specific tablespace, enter:

    ```
    RECOVER TABLESPACE <Tablespace Name>
    ```

- To recover a specific datafile, enter:

    ```
    RECOVER DATAFILE '<datafile name with complete path>';
    ```

The database notifies you when media recovery is complete:

```
Media recovery complete.
```

Opening the database for use

After recovery terminates, open the database for use by issuing the following command at the **SQL*Plus** prompt:

```
ALTER DATABASE OPEN;
```

Using advanced User Managed restore procedures

With the User Managed backup method, Plug-in *for Oracle* lets you rename the Control File, Datafiles, and Archive Log Directory or Individual Archived Log files during restore. Renaming files can be useful if you do not want to overwrite the existing version and want to create a copy of the file. You can also relocate the files to a different directory while restoring it with the plug-in.

i | IMPORTANT: Renaming of data is not supported when restoring Users Managed backups to a raw device. This option can only be used if you are restoring a User Managed backup to file system storage.

The following topics describe advanced User Managed restores and provide more information regarding additional features that are available:

- Renaming or relocating Control File, Datafile, or individual Archived Log files
- Renaming or relocating the Archive Log Directory

Renaming or relocating Control File, Datafile, or individual Archived Log files

1 Complete the steps outlined in Preparing the database for recovery.

2 In the Navigation pane, click **Create Restore Job**.

3 In the saveset table, select the applicable saveset, and click **Restore**.

   For more information, see Selecting data for a restore.

4 On the **Create Selection Set** page, click the item that you want to rename or relocate, and select **Rename** from the context menu.

5 In the **Rename/Relocate** dialog box, enter the applicable information, and click **OK**.

   ▪ **Rename:** Enter the new name.

   ▪ **Relocate:** Enter the new path.

   The item is accompanied by the name and location information in parentheses.

6 If you want to overwrite existing files of the same name in the new or current location, complete the following steps:

   a On the **Create Selection Set** page, click ⬜ .

   b Click the **Restore Options** tab, select **Allow Overwrite Existing Files**, and click **OK**.

7 Continue with the restore procedure as explained in Finalizing and submitting the job.

> **i** | **IMPORTANT:** When renaming individual archive log files during a restore job, perform media recovery with **SET AUTORECOVERY OFF**, and enter the archive log filenames manually to indicate the revised filenames.

Renaming or relocating the Archive Log Directory

When renaming an Archive Log Directory, all archive logs are restored to the directory specified.

1 Complete the steps outlined in Preparing the database for recovery.

2 In the Navigation pane, click **Create Restore Job**.

3 In the saveset table, select the applicable saveset, and click **Restore**.

   For more information, see Selecting data for a restore.

4 On the **Create Selection Set** page, click the Archive Log Directory that you want to rename or relocate, and select **Rename** from the context menu.

5 In the **Rename/Relocate** dialog box, enter the applicable information, and click **OK**.

   ▪ **Rename:** Enter the new name.

   ▪ **Relocate:** Enter the new path.

   The item is accompanied by the name and location information in parentheses.

6 If you want to overwrite existing files of the same name in the new or current location, complete the following steps:

   a On the **Create Selection Set** page, click ⬜ .

   b Click the **Restore Options** tab, select **Allow Overwrite Existing Files**, and click **OK**.

7   Continue with the restore procedure as explained in Finalizing and submitting the job.

> **i** | **IMPORTANT:** When restoring some or all of the required archived redo log files to an alternate
> location, specify the location before media recovery by using the **LOGSOURCE** parameter of the
> **SET** statement in **SQL*Plus**.

Performing RMAN restores

A standard RMAN restore with Plug-in *for Oracle* includes the following steps:

- Selecting data for a restore
- Setting restore options
- Finalizing and submitting the job

Selecting data for a restore

1   In the Navigation pane, click **Create Restore Job**.

2   To filter the items displayed in the saveset table, click **Filter** ▼.

The table displays the saveset name (job title and saveset ID), creation date and time, size, and catalog
status. By default, the list is sorted alphabetically by creation date.

The following table outlines the backup-type identifiers:

Backup type	Backup type identifier
RMAN Full Database	RMAN FULL DB
RMAN Incremental Level 0	RMAN INCRMTAL LVL 0
RMAN Cumulative Incremental	RMAN CUML INCRMTAL
RMAN Differential Incremental	RMAN DIFF INCRMTAL
RMAN Archive Log	RMAN ARCHIVE LOG
RMAN Control File and Parameter File	RMAN CTRL PARM
RMAN Flash Recovery Area	RMAN FRA
RMAN Duplicate Database	RMAN DUPL DB
All other RMAN Backups	RMAN Backup

3   In the saveset table, select the applicable saveset.

When you select a saveset, the following details are displayed in the **Saveset Information** area: Job ID,
job title, server name, client name, plug-in name, saveset date and time, retirement setting, Incremental
Backup or not, Archive or not, and saveset size.

4   Select one of the following options:

- **Restore All Using Defaults:** To restore using the prepopulated defaults, click this button, and
  proceed to Setting restore options.

- **Restore:** To use the **Create Selection Set** page to select the items that you want to restore, click
  this button and proceed to the next step.

5   On the **Create Selection Set** page, with the **Whole Database** node and the other nodes based on the
items selected for the backup displayed, select the applicable items:

- **Whole Database**: Select this node if you want to Restore or Recover, or both, the entire database.
  With this option selected, you can perform the DBPITR based on SCN, LSN, or time to reconstruct
  the database up to the required point.

Selecting the **All Tablespaces** node or all the datafiles and tablespaces nodes is *not* equivalent to selecting this node.

- **Parameter File**: Available only if the **Control File Autobackup Enabled** option is cleared on the **Add**, or **Edit**, **Oracle Database** dialog box.

  When restoring a parameter file, select the backup that contains the specific version of the Parameter File that you want. For example, the Parameter File is backed up weekly on Sunday night at 11 p.m. Assuming today is Thursday and you need to restore the Parameter File to the state it was on Tuesday at 6 p.m., select the Parameter File from the last backup that occurred before Tuesday at 6 p.m., which would be Sunday night's backup at 11 p.m.

- **SPFILE**: For restore purposes, a node SPFILE is available in the restore selections if the backup contents include the backup of the SPFILE. If selected, it restores the SPFILE using RMAN commands—in particular the **RESTORE SPFILE** command.

  > **i** | **NOTE:** An SPFILE cannot be restored to overwrite a live SPFILE, for example, if the database was started using the SPFILE. However, it can be restored to a different location. Use the **Restore Rename** feature of NetVault and input the full path —full directory path and filename—of the location to which to restore the SPFILE.

  Unlike the loss of the Control File, the loss of the SPFILE does not cause the Oracle Instance to stop running immediately. The Oracle Instance may continue operating while it is not shut down. You can restore a backed-up SPFILE to an alternate location. After the Oracle Instance is shut down, restart the Oracle Instance using the restored SPFILE, or copy the restored SPFILE to the default location and then restart the Oracle Instance.

- **Control File**: You cannot open this node further. Also, if a Recovery Catalog is not in use, you cannot restore it from the WebUI. The restore fails if you try to include Control File in a restore job, unless **Use Catalog** is enabled on the **Recovery Catalog** tab.

  When restoring the Control File from Manual Backup, select the backup that contains the specific version of the Control File that you want. For example, Manual Backups of the Control File are performed nightly at 11 p.m. Assuming today is Thursday and you need to restore the Control File to the state it was on Tuesday at 6 p.m., select the Control File from the last backup that occurred before Tuesday at 6 p.m., which would be Monday night's backup at 11 p.m.

- **All Tablespaces**: Select this node to perform an **All Tablespace** restore where all the tablespaces are restored consecutively with the RMAN **RESTORE DATAFILE** command. Performing an **All Tablespace** restore requires the database to be in a **MOUNT** state.

- **Individual Tablespaces/Datafiles**: Open the **All Tablespaces** node to reveal the tablespaces available for restoration. You can select individual or multiple tablespaces, or open the tablespace node and select individual datafiles. Restoring individual tablespaces or datafiles requires the tablespace to be OFFLINE during the restore and recovery process.

- **External Configuration Files**: You can expand this node to reveal the Oracle Network Configuration and Oracle Password files available for restore. Select this node to restore all the available External Configuration Files, or open it and select individual files. The plug-in uses the Plug-in for FileSystem functionality to restore the Oracle Network Configuration and Oracle Password files.

Setting restore options

In this step, you specify whether you want to perform complete or PIT Recovery, and set the other restore options.

On the **Create Selection Set** page, click ![edit icon], and then follow the steps outlined in the following topics:

- Viewing the contents of backup pieces
- Setting the Pre-Restore and general Restore Options
- Setting recovery options
- Setting Block Media Recovery options

Viewing the contents of backup pieces

Before proceeding, you can view the contents of the various backup pieces in the selected backup set. You can see what each piece contains, the backup start and end time, datafile names, SCN range (including the low and high SCN), and the log sequence number of the archived redo logs. These details are displayed on the first tab, **Backup Contents**.

Setting the Pre-Restore and general Restore Options

To continue, click the **Restore Options** tab.

- **Pre-Restore Options**: These options let you automatically prepare the database or tablespaces for the restore process during the plug-in's restore job.

    - **Whole Database Restore**: Select this option if a complete database is being restored.

    - **Tablespace/Datafile Restore**: Select this option if individual or multiple tablespaces or datafiles are being restored.

    - **Startup Database Nomount Before Restore**: When restoring a Control File, the database must be in a **NOMOUNT** state. Select this option if you want the plug-in to put the database in a **NOMOUNT** state automatically before issuing the RMAN run block.

    - **Mount Database Before Restore**: When performing **Whole Database** or **All Tablespaces** restores, the database must be in a **MOUNT** state. Select this option if you want the plug-in to mount the database automatically before performing the restore and recovery. If this option is not selected, you must manually put the database in a **MOUNT** state before submitting the restore job to prevent the job from failing.

    - **Shutdown Abort Database First**: Available only when **Mount Database Before Restore** is selected. If the database is in an **OPEN** state, the database must be shut down before it can be opened in a **MOUNT** state. Select this option if you want the plug-in to run a **SHUTDOWN ABORT** command automatically before starting the database in a **MOUNT** state. If this option is not selected and the **Mount Database Before Restore** option is selected, the restore job fails if you do not manually shut down the database before submitting the job.

    - **Alter Tablespaces Offline Before Restore**: When restoring individual tablespaces or datafiles, the tablespace must be offline before the restore is performed. Select this option if you want the plug-in to run the **ALTER TABLESPACE OFFLINE** command automatically for each tablespace that is being restored.

- **Restore Options**: These options pertain to the restore phase of Oracle's restore and recovery process.

    - **Alter Tablespaces Online After Restore**: When restoring individual tablespaces and datafiles, the tablespace must be returned to an online state after the restore and recovery is completed. Select this option if you want the plug-in to run the **ALTER TABLESPACE ONLINE** command automatically after the selected tablespaces are restored and recovered. This option enforces a recovery of the selected tablespace regardless of whether **Perform Recovery** is selected.

    - **Include Recover Tablespace Clause:** If you selected the **Alter Tablespaces Online After Restore** option, this option is selected by default, which ensures that the tablespaces are included in the recovery and brought online. Clear this option if you want to recover the tablespaces manually.

- **Check Read-Only Datafiles and Restore if Necessary**: Select this option to include read-only datafiles in the restore. The read-only files are included only if they are needed to complete recovery. If they are not damaged, RMAN omits them.

- **Number of Channels**: You can control the degree of parallelism within a restore job by allocating the desired number of channels. Allocating multiple channels simultaneously allows a single restore job to restore multiple backup sets in parallel. The number of channels should be less than or equal to the number of channels that was specified for the backup job.

- **Restore/Recover from User Defined Backup Tag**: Select this option and enter a tag name to select backup sets from which to restore. Selecting this option is equivalent to the RMAN restore option **FROM TAG="<*tag_name*>"**. This option overrides the default RMAN restore selection of the most recent backups or file copy available. This option also restricts the automatic selection to backup sets or file copies that were created with the specified tag. If multiple backup sets or file copies have a matching tag, RMAN selects the most recent backup set or file copy. Tag names are not case-sensitive.

  > **i** | **IMPORTANT:** In a Data Guard environment, Quest recommends that you use the **Restore/Recover from User Defined Backup Tag** option to restrict the restore to backups taken from a particular Oracle Database Server. For more information, see Restoring data to an alternate server in a Data Guard environment.

- **Restore Source**: Select the location that RMAN should use as the source for the restore. This option lets you speed up restores by ensuring that RMAN restores from the FRA even if the backup is also available on NetVault media, that is, disk-based media, VTL, or tape.

  Alternately, the **Restore Source** option can ensure that RMAN restores from the NetVault media when the FRA is suspected to have a media failure or data corruption. This option is useful when you do not want to risk restarting the restore and recovery process due to RMAN's inability to restore from the FRA.

  Additionally, you can use these same options even if you originally performed CLI-based backups to disk by allocating a channel to disk.

  The following **Restore Source** options are available:

  - **NetVault Media Manager**

  - **Disk (Restore from FRA or Disk)**

  - **Both NetVault Media Manager and Disk (Restore from both NetVault Media and (FRA or Disk))**

  For more information, see Choosing the restore source.

- **Validate Restore**: Validates the restore without restoring any data. The plug-in tells RMAN to select and confirm that the existing available backup sets are sufficient for the selected operation, such as an entire database restore or a single tablespace restore. This step ensures that the necessary backup sets are not corrupted and are usable. The following **Validate Restore** options are available:

  - **Validate Restore Only**: If this option is selected, only validation is performed.

  - **Perform Restore if Validate Completes Without Errors**: If this option is selected and the **Validate Restore** completes without errors, the restore is also performed.

Setting recovery options

The plug-in supports the following types of Datafile Media Recovery. For more information on complete and incomplete database recovery, see Datafile media recovery.

- **Complete Recovery of individual datafiles and tablespaces**

- **Complete Recovery of whole database**

- **Point-in-Time Recovery of Database (DBPITR)**

Use the **Perform Recovery** tab to select the type of recovery and the recovery options that pertain the recovery phase of the process:

- **Recovery Type**: Use these options to select the type of recovery to perform.

    - **Do Not Perform Recovery**: Select this option if you want to restore the files but do not want to perform recovery.

    - **Perform Complete Recovery**: Select this option to recover an individual tablespace, multiple tablespaces, or the whole database to the most recent point, without the loss of any committed transactions.

    - **Perform Tablespace Point in Time Recovery**: Select this option to restore the datafiles of the tablespaces chosen on the **NetVault Selections** tab to a specific point. Use a control—**System Change Number Based**, **Log Sequence Based**, or **Time Based**—to restore the datafiles to a specific SCN, log sequence, or time. Use the **Auxiliary Destination** control to define the location of the Oracle-based automated auxiliary instance used to perform the recovery of the tablespaces.

    - **Perform Table Level Point in Time Recovery**: Select this option to recover specific tables to a specific point. Use a control (**System Change Number Based**, **Log Sequence Based**, or **Time Based**) to specify the point to recover to. Use the **Auxiliary Destination**, **Recover table**, and **Remap table** controls to provide additional options for the recovery-table task. For more information and examples, see Performing table-level recovery.

    - **Perform Database Point in Time Recovery**: Select this option if you need to return a database to its state at a past point. For example, to reverse the effect of a user error, such as dropping a table, you might want to return the database to its contents before the deletion occurred.

        The plug-in currently supports only DBPITR. You can use this option only with the **Whole Database** node on the **Create Selection Set** page. Enabling this option when restoring one or more datafiles or tablespaces causes the restore job to fail.

        You can restore the database to its state at a previous target SCN, log sequence, or time.

        > **i** | **NOTE:** DBPITR opens the database in **RESETLOGS** mode. When **Perform Database Point in Time Recovery** is selected, the options on the **Post Recovery** tab, described in Setting Post Recovery options, are not available because an **OPEN** mode is already part of the DBPITR.

    - **Point-in-time controls**: You can use one of the following options with the preceding options to specify a point to recover to.

        - **System Change Number Based**: For an SCN-based PIT Recovery, select this option, and specify the SCN until which the transactions must be recovered. RMAN recovers up to, but not including, the specified SCN.

        - **Log Sequence Based**: For a log-sequence-based PIT Recovery, select this option, and specify the terminating log sequence number and the thread to which it belongs. To identify the appropriate log sequence number and thread, you can query **V$LOG_HISTORY** to view the logs that have been archived.

        - **Time Based**: For a time-based PIT Recovery, select this option, and set the date and time in the fields provided. The plug-in recovers up to, but not including, the specified time.

    - **Auxiliary Destination**: Use this field to specify a directory—full path—that an Oracle-based automated auxiliary instance uses as part of the PIT Recovery process. By default, this field lists a directory within the **ORACLE_HOME** directory. You can change this directory to any other location. This directory *must* exist before you run the recovery process, and you must ensure that the Oracle user has access to the full path. For more information, see Performing table-level recovery.

    - **Recover table**: Enter a comma-separated list of tables that you want to include in a table-level recovery. For more information, see Performing table-level recovery.

    - **Remap table**: Enter a comma-separated list of tables that you want to rename as a part of a table-level recovery. For more information, see Performing table-level recovery.

    - **Perform Trial Recovery Only**: Only available for Oracle 10g and later, Enterprise Edition databases. Select this option to perform a Trial Recovery, which you can use to determine whether

a recovery succeeds or encounters problems or whether a previous recovery has encountered a problem. It lets you look ahead into the redo stream to detect possible problems. The Trial Recovery applies redo in a way similar to normal recovery, but it does not write changes to disk and it rolls back its changes at the end of the Trial Recovery. If errors are encountered during the Trial Recovery, they are logged in the Oracle Alert Log as test run errors.

> **i** | **IMPORTANT:** Because Trial Recovery occurs in memory, the Trial Recovery requires enough system resources to complete. Trial Recovery terminates if the database runs out of the maximum number of buffers that Trial Recovery is permitted to use.

- **Recovery Options**: These options pertain to the recovery phase.

  - **Check Read-Only Datafiles and Recover if Not Current**: Used with the **Check Read-Only Datafiles** and **Restore if Necessary Restore Option**. Select this option to include read-only datafiles in the recovery process. The read-only files are included only if they are needed to complete recovery. If they are not damaged, RMAN omits them.

  - **Delete Restored Archive Logs No Longer Needed**: The plug-in copies the archived redo logs to the directory from which they were backed up. Select this option to delete the restored archive logs that Oracle deems unnecessary.

Setting Block Media Recovery options

Use the options on the **Block Media Recovery** tab to specify that you want to perform Block Media Recovery. When Block Media Recovery is performed, the following should be noted:

- Objects selected on the **Create Selection Set** page are ignored.

- **Mount Database Before Restore** and **Shutdown Abort Database First** options on the **Restore Options** tab are honored.

- Options selected on the **Post Recovery** tab are honored.

- All other Restore and Recovery options are ignored.

The Block Media Recovery tab contains the following options:

- **Perform Block Media Recovery**: Select this option to perform Block Media Recovery and recover only the corrupted blocks versus recovering the entire datafile.

- **Recover All Corrupted Blocks**: Select this option to recover all the corrupted blocks that are reported in **V$DATABASE_BLOCK_CORRUPTION**.

- **Specify Corrupted Blocks to Recover**: When you only want to recover a certain subset of the corrupted blocks, select this option. In the corresponding text box, enter the datafile number and block number of each of the corrupted blocks that you want to recover. Enter one datafile-block entry per line in the following format:

```
DATAFILE <num> BLOCK <num>
DATAFILE <num> BLOCK <num>
.
.
.
DATAFILE <num> BLOCK <num>
DATAFILE <num> BLOCK <num>
```

In this example, *<num>* represents the datafile and block number to recover.

The text box accepts up to 4096 characters, which results in approximately 200 datafile-block entries. The syntax is not validated.

Setting Post Recovery options

When performing recovery using the **Whole Database**, **All Tablespace**, or **Disaster Recovery** options, the database is in a **MOUNT** state during the restore and recovery process. You must return the database to the **OPEN** state after completion. The plug-in provides the flexibility of opening the database automatically after the recovery process during the restore.

The **Post Recovery** tab includes the following options:

- **Open Database after Recovery**: Select this option to open the database automatically and make it available for normal use after the recovery process is completed during the restore. Selecting this option makes the following options available:

  - **Read Write**: Opens the database in read/write mode, which lets you generate redo logs. This option is the default if you are opening a primary database.

  - **Read Write and Reset Logs**: Opens the database in read/write mode and resets the current log sequence number to 1. It also archives any un-Archived Log files, including the current log, and discards any redo information that was not applied during recovery so that it is never applied. This option must be selected in the following situations:

    - After performing incomplete media recovery or media recovery using a backup Control File.

    - After a previous **OPEN RESETLOGS** operation did not complete.

    - After a **FLASHBACK DATABASE** operation.

  - **Read Only**: Select this option to restrict users to read-only transactions, which prevents them from generating redo logs. This option is the default setting when you open a physical standby database, so that the physical standby database is available for queries even while archive logs are being copied from the primary database site. Oracle has the following restrictions on opening a database with the **Read Only** option:

    - You cannot open a database in **Read Only** mode if it is opened in **Read Write** mode by a different instance.

    - You cannot open a database in **Read Only** mode if it requires recovery.

    - You cannot take tablespaces offline while the database is open in **Read Only** mode. However, you can take datafiles offline and online, and you can recover offline datafiles and tablespaces while the database is open in **Read Only** mode.

Setting Destination Details

If the SYSDBA user name or password or the Oracle SID for the database has changed since the time of backup, set the following options for the restore job on the **Destination Details** tab:

- **Target Service**: This option is required only if the current SID for the Oracle DBID has been changed since the backup was taken.

- **Oracle SYSDBA User Name**: Specify the new SYSDBA user name if changed.

- **Oracle SYSDBA Password**: Specify the password associated with the name specified in the preceding field.

  > **i** | **IMPORTANT:** The Destination Details requires that the DBID of the backed-up or source database match the DBID of the destination database. When the goal of the restore is to create a duplicate of the source database to a database with a different DBID, using a **Duplicate Database** backup is required. For more information, see Duplicate Database backup.

Setting the Recovery Catalog details

While the Recovery Catalog information is provided for backups on the **Configure** or **Add**, or **Edit**, **Oracle Database** dialog box, set the Recovery Catalog configuration for a particular restore job by configuring the Recovery Catalog information on the **Recovery Catalog** tab.

- **Use Catalog**: Select this option to use the RMAN repository in the Recovery Catalog and store information about this restore operation in a Recovery Catalog. Clear **Use Catalog** when performing a test recovery.

  The following fields are enabled when you select this option:

  - **Catalog Instance Name**: Enter the Oracle Instance name, which might differ from the Oracle Net Service name, for the Recovery Catalog database. This instance name must be defined in the **"tnsnames.ora"** file on the Oracle Database Server so that the plug-in can connect to the Recovery Catalog database. If the Recovery Catalog was used during the backup, the **Instance Name** field defaults to the Oracle Instance name used during the backup job.

  - **Catalog Owner**: Specify the user that was specified as the owner of the Recovery Catalog and was granted the **RECOVERY_CATALOG_OWNER** role.

  - **Catalog Password**: Provide the password associated with the Catalog Owner name specified in the preceding field.

Setting the Clone Database options

The **Clone Database** tab is available only when an **RMAN DUPL DB** backup is selected on the **Create Selection Set** page. You can use it to specify the options for restoring a **Duplicate Database** backup to a destination database. For more information on the prerequisites, see Duplicating a database in a non-RAC environment or Duplicating a database in a RAC environment.

- **Duplicate Database**: Select this option to restore the selected Duplicate Database backup to an alternate destination database.

- **Destination Database** section:

  - **Auxiliary Instance**: Specify the Oracle Net Service name of the auxiliary instance. This field defaults to **auxdb**.

  - **Oracle Home**: Specify the Oracle Home for the auxiliary instance. This field defaults to the Oracle Home of the source database.

  - **Oracle SYSDBA User Name**: Specify the SYSDBA user name that is used to connect to the auxiliary instance. This field defaults to the SYSDBA user name specified in the configuration of the source database.

  - **Oracle SYSDBA Password**: Enter the password for the SYSDBA user name specified in the preceding field.

  - **Use nofilenamecheck Option**: Select this option to prevent RMAN from verifying whether the datafiles and online redo logs files of the source database are in use when the source database files use the same names as the duplicate database files. You are responsible for determining that the duplicate operation does not overwrite useful data.

    This option is required when you restore a duplicate database to an alternate server. Do not select this option if you are restoring a duplicate database to the same server; otherwise, the restore job fails with the following errors:

    - RMAN-10035: exception raised in RPC: ORA-19504: failed to create file

    - ORA-27086: skgfglk: unable to lock file – already in use

    - SVR4 Error: 11: Resource temporarily unavailable

    - RMAN-10031: ORA-19624 occurred during call to DBMS_BACKUP_RESTORE. RESTOREBACKUPPIECE

- **Duplicate From Database** section:

  - **Target Database**: Specify the Oracle SID of the target database. This field defaults to the Oracle SID of the database including in the backup.

  - **Oracle Home**: Specify the Oracle Home for the source database. This field defaults to the Oracle Home of the source database.

  - **Oracle SYSDBA User Name**: Specify the SYSDBA user name that is used to connect to the source database. This field defaults to the SYSDBA user name specified in the configuration of the source database.

  - **Oracle SYSDBA Password**: Enter the password for the SYSDBA user name specified in the preceding field.

Setting Generate RMAN Script options

Intended for the advanced DBA, the **Generate RMAN Script** feature lets you generate most of an RMAN script through the WebUI. Later, before using the script, you can add more options or parameters. With this option, you do not need to create the script from scratch, which minimizes the risk of syntax errors in the scripts and that restores are not performed correctly.

You can set the following options for the **Generate RMAN Script** feature:

- **Output RMAN Script to File**: Select this option to use the feature and enable access to the remaining options, and then enter the name of the file to which the RMAN script is written.

- **RMAN Script Directory**: Enter the name of the directory where the RMAN scripts are saved. The **Generate Script Default Directory** field defaults to the directory specified on the plug-in's Configure dialog box. However, you can modify the field on a per-backup-job basis. The full path where the RMAN script is saved is built by concatenating the **RMAN Script Directory** with the filename provided in the **Output RMAN Script to File** field.

- **If Existing, Overwrite Destination File**: Select this option to allow the destination file where the RMAN script is output to be overwritten if the file exists. If the file exists and this option is not selected, **Generate RMAN Script** fails and does not write output to the existing file. Select this option only if you are sure that you want to overwrite an existing file.

- **Generate RMAN Script and Submit Job**: With this option selected, in addition to outputting the RMAN script to the specified file, the NetVault Job is scheduled and submitted. Clearing this option ensures that the plug-in only generates the script, which lets you edit and run the script outside of NetVault.

- **Do not Include Connect Strings:** To exclude connection strings from the file because they might include Oracle passwords, select this option. Selecting this option ensures that Oracle passwords are not included in the RMAN script file.

Finalizing and submitting the job

Complete the steps outlined in Finalizing and submitting the job under Performing User Managed restores.

Using RMAN types of recovery in a non-RAC environment

The following topics describe the types of recovery that you can perform in a non-RAC environment:

- Restoring the Control File from an Autobackup
- Recovering individual datafiles and tablespaces to the same server
- Recovering Whole Database to same server
- Performing disaster recovery with RMAN

For information on the types of RMAN recoveries available in a RAC environment, see Restoring data in a RAC environment.

Restoring the Control File from an Autobackup

Restoring the Control File should only be performed in the following cases:

- All current copies of the Control File have been lost due to a media failure or accidental user deletion, or have become corrupted and are not usable.
- Oracle Customer Support has advised that a Control File restore is required.

As the Autobackup uses a default format, RMAN can restore it even if it does not have a repository available that lists the available backups. RMAN replicates the Control File to all CONTROL_FILES locations automatically.

1 Start RMAN, and connect to the target database:

```
>rman connect TARGET /
```

2 Start the target instance without mounting the database:

```
STARTUP FORCE NOMOUNT;
```

3 Use **SET DBID** to set the database identifier for the target database.

RMAN displays the DBID whenever you connect to the target. You can also obtain it by inspecting saved RMAN log files, querying the Catalog, or looking at the filenames of Control File Autobackup. For example, run:

```
SET DBID <Source_DBID>;
```

4   Restore the Control File Autobackup:

```
RUN {
ALLOCATE CHANNEL CH1 TYPE 'SBT_TAPE';
RESTORE CONTROLFILE FROM AUTOBACKUP;
ALTER DATABASE MOUNT;
RESTORE DATABASE;
RECOVER DATABASE;
}
```

If the default format was modified for the Control File Autobackup, specify the same format during a restore:

```
RUN {
SET CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE <device type> TO '<string>'

ALLOCATE CHANNEL CH1 TYPE 'SBT_TAPE';
RESTORE CONTROLFILE FROM AUTOBACKUP;
ALTER DATABASE MOUNT;
RESTORE DATABASE;
RECOVER DATABASE;
}
```

5   Open the database, and reset the online logs:

```
ALTER DATABASE OPEN RESETLOGS;
```

Running the **ALTER DATABASE OPEN RESETLOGS** command from RMAN versus **SQL*Plus** resets the target database automatically so that you do not have to run a **RESET DATABASE**. By resetting the database, RMAN considers the new incarnation as the current incarnation of the database. If you run **ALTER DATABASE OPEN RESETLOGS** from **SQL*Plus** and do not follow it with a **RESET DATABASE** at the RMAN prompt, RMAN refuses to access the Recovery Catalog. RMAN refuses because it cannot distinguish between a **RESETLOGS** operation and an accidental restore of an old Control File.

> **i** | **IMPORTANT:** After restoring the Control File from an Autobackup, Quest recommends that you perform a Full or Incremental Level 0 backup with the database node selected on the **NetVault Selections** page.

Recovering individual datafiles and tablespaces to the same server

This type of recovery is performed when the database is open, but some of the datafiles are damaged. This recovery is useful when you want to recover the damaged tablespace, while leaving the database open so that the rest of the database remains available. You can only perform **Complete Recovery** when restoring and recovering individual datafiles or tablespaces. *PIT Recovery is not available for individual datafiles or tablespaces.* Enabling PIT Recovery causes the restore job to fail. Following is an overview of the process to recover individual datafiles or tablespaces:

1   On the **Create Selection Set** page, select the damaged datafiles or tablespaces from the latest available backup of the tablespace, and click ⬚ .

2   Click the **Restore Options** tab, and select the **Alter Tablespaces Offline Before Restore** and the **Alter Tablespaces Online After Restore** options.

3   On the **Perform Recovery** tab, select the applicable options:

   ▪   To recover the tablespace to the current time, that is, last committed transaction available in the archive and online redo logs, select the **Perform Complete Recovery** option.

   ▪   If you do not want to perform any recovery, select the **Do Not Perform Recovery** option.

   > **i** | **NOTE:** If you select the **Perform Database Point in Time Recovery** option, the restore job fails.

- To delete the restored archive logs that Oracle deems unnecessary, select the **Delete Restored Archive Logs No Longer Needed** option.

4 Complete the **Target Client**, **Schedule**, and **Advanced Options** tabs, and submit the job.

RMAN automatically chooses the best available Full, Incremental, or Archive Log backups and restore and recover the tablespace.

Recovering Whole Database to same server

This type of recovery is performed when the current Control File and SPFILE are intact but all the datafiles are damaged or lost. You can perform a Complete Recovery or recover the database to a specific point. Following is an overview of the process to recover the whole database, which must be performed when the database is in a **MOUNT** state:

1 On the **Create Restore Job—Choose Saveset** page, find the latest available backup that included *all* the tablespaces, and click **Next**.

2 On the **Create Selection Set** page, navigate to the source NetVault Client and database, select the **Whole Database** node, and click 📝 .

3 Click the **Restore Options** tab, and select the applicable options:

- To mount the database before the performing the restore, select **Mount Database Before Restore**.
- To shut down the database automatically before mounting the database, select **Shutdown Abort Database First**.
- To include read-only datafiles in the restore, select **Check Read-Only Datafiles and Restore if Necessary**.

4 To perform recovery on the database, that is, to reconstruct the database up to the current or specified point, select one of the following options on the **Perform Recovery** tab:

- **Perform Complete Recovery**: Recovers the database to the current time, that is, to the last committed transaction available in the archived and online redo logs.
- **Perform Point-in-Time Recovery**: Recovers the database to a specific point whether it be a specific time, log sequence number, or SCN.

5 On the **Perform Recovery** tab, select these additional options, if applicable:

- To include read-only datafiles in the recovery process, select **Check Read-Only Datafiles and Recover if Not Current**. The read-only files are included only if they are needed to complete recovery. If they are not damaged, RMAN omits them.
- To delete the restored archive logs that Oracle deems unnecessary, select **Delete Archive Logs No Longer Needed**.

6 On the **Post Recovery** tab, select **Open Database after Recovery and Read Write and Reset Logs** to open the database automatically in read-write mode and reset the online logs.

7 Complete the **Target Client**, **Schedule**, and **Advanced Options** tabs, and submit the job.

RMAN automatically chooses the best available full, incremental, or Archived Log backups and restore and recover the database.

> **i** | **IMPORTANT:** After completing a restore of the Whole Database, Quest recommends that you perform a Full or Incremental Level 0 backup with the database selected on the **Create Selection Set** page.

Performing disaster recovery with RMAN

Disaster recovery includes the restore and recovery of a database after the loss of the entire target database, all current Control Files, all online redo log files, and all parameter files. It is also assumed that the FRA has been lost. To perform a disaster recovery, the minimum required set of backups includes backups of all the datafiles, some archived redo logs generated after the time of the backup, and at least one Autobackup of the Control File.

If your goal is to create a copy of your target database for ongoing use on a new host, do *not* use these procedures. Instead, use the **Duplicate Database** backup type, which is designed for this purpose. For more information, see Duplicate Database backup.

> **!** | **CAUTION: To avoid overwriting the database and losing data, use caution when selecting any of the options described in the following topics and before performing disaster recovery.**

The following methods are available for performing a disaster recovery:

- Using the Stepwise Disaster Recovery process
- Disaster recovery to same server
- Retrieving a Control File manually

Important notes

When performing a disaster recovery of any kind, be aware of the following:

- When performing a test recovery, do not connect to the Recovery Catalog during datafile restoration. Otherwise, RMAN records information about the restored datafiles to the Recovery Catalog. This recording interferes with future attempts to restore and recover the primary database. If you must use a Recovery Catalog because the Control File is not large enough to contain the RMAN repository data on all the backups that you need to restore, export the catalog. After exporting the catalog, import it into a different schema or database, and use the copied Recovery Catalog for the test restore. Otherwise, the catalog considers the restored database as the current target database.

- Using a disaster recovery procedure in an Oracle database environment that is not intended for disaster recovery might result in data loss in the database. Do not use a disaster recovery procedure to restore or recover individual datafiles or archive logs, or if you only intend to restore an Oracle database SPFILE or a Control File.

- Perform a disaster recovery only when it involves the whole database, SPFILE, and Control File. Existing data might be lost in the database, because a disaster recovery procedure overwrites it.

- The Stepwise Disaster Recovery feature issues commands that include a small subset of the options available in an Oracle Database Server to perform a disaster recovery. For more advanced options, or a targeted disaster recovery, Quest recommends that you use the plug-in's documented RMAN CLI procedures. Because the Stepwise Disaster Recovery feature does not use the full range of recovery options, it might not complete successfully. If not, use the RMAN CLI process to perform the recovery.

Using the Stepwise Disaster Recovery process

This process runs a restore of the Control File from an Autobackup and restores the database datafiles using the same restore task. If you use this process, be aware of the following:

- Objects selected on the **Create Selection Set** page are ignored.
- All other restore- and recovery-related options are ignored.
- Only the options selected on the **Stepwise Disaster Recovery** tab are run.

To use Stepwise Disaster Recovery:

1 In the Navigation pane, click **Create Restore Job**.

2 To filter the items displayed in the saveset table, click **Filter** ▼.

The table displays the saveset name (job title and saveset ID), creation date and time, size, and catalog status. By default, the list is sorted alphabetically by creation date.

3   In the saveset table, select the applicable saveset.

When you select a saveset, the following details are displayed in the **Saveset Information** area: Job ID, job title, server name, client name, plug-in name, saveset date and time, retirement setting, Incremental Backup or not, Archive or not, and saveset size.

4   Click **Restore**.

5   On the **Create Selection Set** page, select any item.

As explained earlier, selections on this page are ignored for this process.

6   Click ✎ , and then click the **Stepwise Disaster Recovery** tab.

7   Select the **Perform the following Step-by-Step Disaster Recovery** option, and then select the applicable suboptions:

- **Set Dbid**: If you want to specify the DBID, select this option. The plug-in automatically enters the DBID of the Oracle database for which the backup was taken in this field. The DBID is needed when a parameter file or SPFILE is not available to start the Oracle database in a **NOMOUNT** state.

- **Restore Spfile from Autobackup; Look Back Max Days**: If the Oracle Control File Autobackup option was enabled in your scheduled backups and you want to restore the SPFILE from an Autobackup, select this option. If Oracle Control File Autobackup was *not* enabled, you cannot restore the SPFILE from an Autobackup; for more information, see Enabling Control File Autobackups. By default, this option looks for Autobackups in the last seven days; update it as applicable. This option issues the following RMAN commands:

  ```
  STARTUP NOMOUNT;
  restore spfile from autobackup maxdays <NumberOfDays>;
  ```

- **Restore Controlfile from Autobackup; Look Back Max Days**: If the Oracle Control File Autobackup option was enabled in your scheduled backups and you want to restore the Control File from an Autobackup, select this option. If Oracle Control File Autobackup was *not* enabled, you cannot restore the Control File from an Autobackup; for more information, see Enabling Control File Autobackups. By default, this option looks for Autobackups in the last seven days; update it as applicable. This option issues the following RMAN commands:

  ```
  STARTUP FORCE NOMOUNT;
  restore controlfile from autobackup maxdays <NumberOfDays>;
  ```

- **Restore Database**: If you want to restore all datafiles included in the Oracle database, select this option. A **RESTORE DATABASE** command attempts to restore all the datafiles in the Oracle database. This option issues the following RMAN command:

  ```
  RESTORE DATABASE;
  ```

- **Recover Database**: If you want to recover all datafiles included in the Oracle database, select this option. A **RECOVER DATABASE** command attempts to recover all the datafiles in the Oracle database. This recovery might include automatically restoring and applying archived transaction logs. This option issues the following RMAN command:

  ```
  RECOVER DATABASE;
  ```

- **Alter Database Open Resetlogs**: If you want to open the database with the **RESETLOGS** option, select this option. This option issues the following **SQL*Plus** command:

  ```
  ALTER DATABASE OPEN RESETLOGS;
  ```

  ⓘ | **NOTE:** With the database restored, you have the option of opening an RMAN CLI prompt and issuing the recovery commands from the prompt. This option eliminates the need to select the **Recover Database** and **Alter Database Open Resetlogs** options and lets you issue advanced recovery options.

8   Complete the steps in Finalizing and submitting the job.

Disaster recovery to same server

The following procedure details the steps to perform disaster recovery to the same server where the source database resided. This procedure would be used in the scenario in which a complete media failure occurred and either the damaged host was rebuilt using the same configuration or a new server was configured imitating the original host's configuration, including host name.

> **i** | **NOTE:** You can also use the following procedure to restore the Oracle Server to a different host, one that is not the original production server, while the production server is still active. However, as stated previously, if your goal is create a copy of the target production database for ongoing use on a new host, do not use this procedure. Instead, use the **Duplicate Database** backup type. For more information, see Duplicate Database backup.

The same procedure should be used whether a Recovery Catalog was being used as part of the backup strategy for the RMAN Repository. RMAN always stores information concerning the backups in records in the Control File. The version of this information in the Control File is the authoritative record of RMAN's backups of your database. RMAN can conduct all necessary backup and recovery operations using just the repository information in the Control File.

For variations on the following disaster recovery procedure, see *Performing Disaster Recovery* in the *Oracle Database Backup and Recovery Advanced User's Guide*.

To perform a disaster recovery without a Recovery Catalog, the following requirements must be met:

- **Oracle database software installed**: Install the same version of the Oracle database software on the system as existed on the damaged server. The **ORACLE_HOME** should be identical on both the source and destination Oracle server. You do not have to create the database.

- **Oracle password file created**: Use the **orapwd** utility to create the Oracle password file.

    - On Linux:

        ```
        orapwd file=<Oracle_Home>/dbs/orapw<SID> password=<password>
        ```

    - On Windows:

        ```
        orapwd file=<Oracle_Home>\database\orapw<SID> password=<password>
        ```

- **Oracle service on Windows created and started**: Create the Oracle Service on Windows with Oracle's **oradim** utility. The **Oracle SID** should be identical to the original or source database. This option creates an identical Oracle Service name. For example, from an OS prompt run:

    ```
    oradim -new -sid <sid_name>
    ```

    After the Windows service is created, start the service with the following command from an OS prompt:

    ```
    net start oracleservice<sid_name>
    ```

- **NetVault software and Plug-in *for Oracle* installed**: The same version of NetVault software and the plug-in must be installed and configured as it originally existed on the damaged server. Ensure that the plug-in's default attributes are defined as specified in Configuring default settings.

- **DBID of the source database identified**: Identify the DBID of the damaged database. If the DBID was not recorded as part of the initial Plug-in *for Oracle* installation, look for it in the NetVault binary logs for RMAN-based backups of the source database. The DBID is also available on the **Details** dialog box or the **Backup Contents** tab.

- **Default location for SPFILE must exist**: Verify that the default location for the SPFILE exists. Failure to do so results in an **ORA-17502 Failed to create file** error during the SPFILE restore.

- **Backups of all tablespaces made available**: Verify that you have the latest backups of all the tablespaces.

To perform the recovery, complete the following steps.

1 Restore the Oracle Net Services Configuration Files.

    If the **External Configuration Files** were included in a previous RMAN backup or the Plug-in for FileSystem was used to back up the Oracle Net Services configuration files, such as "listener.ora,"

"sqlnet.ora," and "tnsnames.ora," restore the files to the destination Oracle Database Server. Otherwise, configure the Oracle Net Services on the destination Oracle server.

2 Start the Oracle listener.

For example, from an OS command prompt run:

```
lsnrctl start
```

3 Start RMAN with no catalog, and connect to the target database:

```
RMAN TARGET / NOCATALOG
```

4 Set the DBID to the damaged database DBID.

```
SET DBID <Source_DBID>;
```

5 Start the target instance without mounting the database.

> **i** | **IMPORTANT:** A series of Oracle errors might appear if a PFILE is not available when you run the following command. This process is acceptable, and you can ignore the errors. Oracle uses a default PFILE to ensure that the restore process for the SPFILE continues—"starting Oracle instance without parameter file for retrieval of spfile."

```
STARTUP NOMOUNT;
```

6 Restore the **SPFILE** from Autobackup.

```
RUN {
ALLOCATE CHANNEL C1 TYPE 'SBT_TAPE';
RESTORE SPFILE FROM AUTOBACKUP;
}
```

7 Restart the database with the restored SPFILE.

```
STARTUP FORCE NOMOUNT;
```

8 Restore the Control File from Autobackup.

If you are restoring the Control File from the last Autobackup, run:

```
RUN {
ALLOCATE CHANNEL CH1 TYPE 'SBT_TAPE';
RESTORE CONTROLFILE FROM AUTOBACKUP;
}
```

> **i** | **IMPORTANT:** If you do not have an Autobackup of the Control File, complete the steps in Retrieving a Control File manually before continuing.

9 On the **Create Restore Job—Choose Saveset** page, find the latest available backup that included *all* the tablespaces, and click **Next**.

10 On the **Create Selection Set** page, navigate to the source NetVault Client and database, select the **Whole Database** node, and click 📝 .

11 Click the **Restore Options** tab, and select the applicable options:

- To mount the database before the performing the restore, select **Mount Database Before Restore**.

- To include read-only datafiles in the restore, select **Check Read-Only Datafiles and Restore if Necessary**.

12 Because recovery must be performed in a disaster recovery scenario, on the **Perform Recovery** tab, select **Perform Database Point-in-Time Recovery**.

PIT Recovery recovers the database to a specific time, log sequence number, or SCN. If the specific time or log sequence of the failure is not known, Quest recommends that you use the SCN option.

13 On the **Perform Recovery** tab, select **Check Read-Only Datafiles and Recover if Not Current** to include read-only datafiles in the recovery process.

The read-only files are included only if they are needed to complete recovery. If they are not damaged, RMAN omits them.

14 On the **Recovery Catalog** tab, clear the **Use Catalog** option to ensure that the Recovery Catalog is not used for this restore.

15 Complete the **Schedule** and **Advanced Options** tabs, and submit the job.

RMAN automatically chooses the best available Full, Incremental, or Archive Log backups and restore and recover the database.

16 In RMAN, update the RMAN metadata.

```
CROSSCHECK ARCHIVELOG ALL;
```

17 Exit RMAN.

> **ⁱ | IMPORTANT:** After completing a disaster recovery, Quest recommends that you perform a Full or Incremental Level 0 backup with the database selected on the **Create Selection Set** page.

Retrieving a Control File manually

When Control File Autobackups are not available, the RMAN Catalog is not used, and only Control File backups were performed by selecting the Control File node on the NetVault Selections page, you might be able to use the following procedure to extract the Control File from a backup piece.

1 On the **Create Restore Job—Choose Saveset** page, find the latest available backup that is marked with "Controlfile Included."

2 Recover the Control File.

```
SQL> STARTUP NOMOUNT;

DECLARE
   devtype varchar2(256);
   done    boolean;
BEGIN
   devtype := dbms_backup_restore.deviceallocate('sbt_tape', params=>'');
   dbms_backup_restore.restoresetdatafile;
   dbms_backup_restore.restorecontrolfileto
      ('<destination PathAndNameOfControlFile>');
   dbms_backup_restore.restorebackuppiece
      ('<backupSegment NameIdentifiedEarlier>',done=>done);
END;
/
```

For example:

```
sqlplus /nolog
connect / as sysdba
STARTUP NOMOUNT;
DECLARE
devtype varchar2(256);
done    boolean;
BEGIN
devtype := dbms_backup_restore.deviceallocate('sbt_tape', params=>'');
dbms_backup_restore.restoresetdatafile;
dbms_backup_restore.restorecontrolfileto
   ('C:\oracle\product\10.1.0\oradata\o10gg\control01.ctl');
dbms_backup_restore.restorebackuppiece
   ('ZETK2003:#3293:ZETK2003:274_578391288:O10GG_1135958971',done=>done);
END;
/
```

3 Use the **REPLICATE CONTROLFILE** command to copy the restored Control File.

```
RMAN TARGET / NOCATALOG
```

```
SET DBID <Source_DBID>;
connect TARGET /

RUN {
ALLOCATE CHANNEL C1 TYPE 'SBT_TAPE';

replicate controlfile from
   '<sourcePathAndNameOfControlFile>';

}
```

Using advanced procedures with RMAN restores

This topic contains subtopics that describe other restore operations that can be performed with Plug-in *for Oracle*.

- Renaming or relocating files

- Duplicating a database in a non-RAC environment

- Performing table-level recovery

Renaming or relocating files

With the RMAN backup method, the plug-in lets you rename the Control File and datafiles during restore. Renaming files can be useful if you do not want to overwrite the existing version and want to create a copy of the file. You can also relocate the files to a different directory while restoring it with the plug-in.

1   Complete the steps outlined in Preparing the database for recovery.

2   In the Navigation pane, click **Create Restore Job**.

3   In the saveset table, select the applicable saveset, and click **Restore**.

    For more information, see Selecting data for a restore.

4   On the **Create Selection Set** page, click the datafile or Control File to restore, and select **Rename** from the context menu.

5   In the **Rename/Relocate** dialog box, enter the applicable information.

    - **Rename:** Enter the new name.

    - **Relocate:** Enter the new path.

The plug-in uses the RMAN **SET NEWNAME** command to restore or rename—or both—RMAN backups. The **SET NEWNAME** command has precedence over the **DB_CREATE_FILE_DEST** parameter. If the full path is not specified for a restore or rename, the plug-in behaves as follows:

    - **SPFILE**: The file is restored to the "$ORACLE_HOME/dbs" directory with the new name.

    - **Datafile**:

        □   If you specify a new location when the file is created, the file is restored or renamed to this location with the new name.

        □   If you do not specify a new location when the file is created, the file is restored or renamed to "$ORACLE_HOME/dbs" with the new name.

    > **i** | **NOTE:** A full path can include a path to an ASM location. For example:
    > **+DATA/my_db/my_datafile_renamed.dbf**

When renaming the Control File during restore, the resulting filename should exist in the **CONTROL_FILES** list in the Oracle Parameter File.

6   To apply changes, click **OK**.

The item is accompanied by the name and location information in parentheses.

7   Continue with the restore procedure as explained in Finalizing and submitting the job.

Duplicating a database in a non-RAC environment

A Duplicate Database backup lets you create a duplicate or clone of a source database for special purposes that does not affect the backup and restore procedures for the source database. RMAN supports duplicating databases to increase a DBA's flexibility when performing the following tasks:

- Testing backup and recovery procedures.

- Testing an upgrade to a new release of the Oracle database.

- Testing the effect of applications on database performance.

- Generating reports.

For more information on duplicating a multi-instance RAC database, see Duplicating a database in a RAC environment. For more information on Oracle's Duplicate Database functionality, see *Duplicating a Database* in the *Oracle Database Backup and Recovery User's Guide*.

The plug-in provides the following restore scenarios for the **Duplicate Database** backup:

- Restoring a Duplicate Database to the same server

- Restoring a Duplicate Database to an alternate server with the same directory structure

- Restoring a Duplicate Database to an alternate server with an alternate directory structure

Restoring a Duplicate Database to the same server

The following procedure details the steps to restore a Duplicate Database backup to the local or same server where the source database resides. Because the duplicated database or destination database resides on the same physical server as the source database, the destination database must use a different directory structure.

- Prerequisites

- Restoring the Duplicate Database

Prerequisites

To restore a Duplicate Database backup to the same server, the following prerequisites must be met:

- **Same version of Oracle database software**: The version and edition of Oracle for the destination database must be identical to the version and edition of Oracle being used for the source database. This requirement includes identical patch levels.

- **Duplicate Database backup available**: A Duplicate Database backup must be completed successfully and made available.

- **Source database in open state**: The source database must remain in an **OPEN READ WRITE** state during the entire Duplicate Database restore process.

- **Auxiliary instance prepared**: RMAN's backup-based Duplicate Database process requires the preparation of an auxiliary instance:

  1   **Oracle password file for auxiliary instance created**: Required if you want to use a password file versus OS authentication for the auxiliary connection when duplicating to the same host as the source database. For more information on creating a password file, see *Creating and Maintaining a Password File* in the *Oracle Database Administrator's Guide*.

2 **Oracle Net connectivity established to auxiliary instance**: The auxiliary instance must be available through Oracle Net by adding the instance to the **"tnsnames.ora"** file. Additionally, on Windows platforms, perform the following command to start the instance:

```
oradim -new -sid <sid_name>
```

3 **Auxiliary instance destination directories exist**: The following destination directories for the auxiliary instance must be created on the server where the destination database resides. If the destination database stores its datafiles in ASM, the ASM instance name for the destination database must exist.

   ▫ Control_Files

   ▫ DB_Create_File_Dest

   ▫ Audit_File_Dest

   ▫ Background_Dump_Dest

   ▫ User_Dump_Dest

4 **PFILE created for auxiliary instance**: A client-side PFILE for the Auxiliary Database must be created from the SPFILE of the source database.

   ▫ Start **SQL*Plus**, and connect with administrator privileges to the source database.

```
sqlplus sys/<password>@<source_connect_identifier> AS SYSDBA
```

   ▫ Create a PFILE from the SPFILE.

```
create pfile = '<PFILE_destination_directory>/
   init<auxiliary_sid>.ora' from spfile;
```

5 **PFILE updated with auxiliary values**: The PFILE created for the auxiliary instance must be updated with the parameter values specific to the auxiliary instance.

   ▫ Remove all entries that begin with: ***<source_sid>.__***

   ▫ Update the ***.db_name=** entry to reflect the name of the destination database.

   ▫ Update the following entries to reflect the destination directory for the auxiliary instance. This directory must exist; however, if the source database stores the datafiles in ASM, the ASM instance name for the destination database should be specified for each of the following **control_files** and **db_create_file_dest** parameters:

```
*.audit_file_dest=
*.background_dump_dest=
*.control_files=
*.user_dump_dest=
*.db_create_file_dest=
*.db_recovery_file_dest=
```

   ▫ Comment out the following entry by inserting a "#" (number sign) at the beginning of the following entry:

```
*.dispatchers='(PROTOCOL=TCP) (SERVICE=<source_sid>XDB)'
```

   ▫ Add the following line to specify the pairs of strings for converting the names of datafiles and tempfiles.

```
*.db_file_name_convert='<source_db_create_file_dest>/<source_sid>',
   '<destination_db_create_file_dest>/<destination_sid>'
```

   For example:

```
*.db_file_name_convert=' C:\oracle\oradata\prod',
   'c:\oracle\oradata\test'
```

   ▫ Add the following line to specify the pairs of strings for naming online redo log files:

```
*.log_file_name_convert='<source_db_create_file_dest>/
   <source_sid>', '<destination_db_create_file_dest>/
```

```
              <destination_sid>'
```

For example:

```
*.log_file_name_convert=' C:\oracle\oradata\prod',
    'c:\oracle\oradata\test'
```

6   **Auxiliary instance started in NOMOUNT state**: The auxiliary instance must be started in
    NOMOUNT state with the PFILE that has been updated with the parameter values specific to the
    auxiliary instance.

   ▫   Start **SQL*Plus**, and connect with administrator privileges to the auxiliary instance.

   ```
   sqlplus sys/<password>@<auxiliary_connect_identifier> AS SYSDBA
   ```

   ▫   Start the instance in **NOMOUNT** state.

   ```
   startup nomount pfile = '<PFILE_destination_directory>/
       init<auxiliary_sid>.ora'
   ```

7   **Auxiliary instance created from SPFILE from the PFILE**

   ```
   create spfile from pfile='<PFILE_destination_directory>/
       init<auxiliary_sid>.ora';
   ```

8   Exit **SQL*Plus**.

    You must exit **SQL*Plus** for the Duplicate Database restore to complete successfully.

Restoring the Duplicate Database

To restore the Duplicate Database, perform the following steps.

1   On the **Create Restore Job—Choose Saveset** page, find the applicable **Duplicate Database** backup,
    and click **Next**.

2   On the **Create Selection Set** page, navigate to the source NetVault Client and database, select the **Whole
    Database** node, and click 🖊 .

3   Click the **Clone Database** tab, and select the **Duplicate Database** option.

4   In the **Destination Database** section:

   ▪   Specify the Oracle Net Service name of the auxiliary instance in the **Target Service** field.

   ▪   Specify the **Oracle Home** for the auxiliary instance.

   ▪   Specify the **Oracle SYSDBA User Name** that is used by the plug-in to connect to the auxiliary
       instance.

   ▪   Specify the **Oracle SYSDBA Password**.

5   In the **Duplicate From Database** section:

   ▪   Specify the **Oracle SID** for the source database.

   ▪   Specify the **Oracle Home** for the source database.

   ▪   Specify the **Oracle SYSDBA User Name** that is used by the plug-in to connect to the source
       database.

   ▪   Specify the **Oracle SYSDBA Password**.

6   Continue with the restore procedure, including completion of the **Schedule** and **Advanced Options** lists,
    as explained in Finalizing and submitting the job.

Restoring a Duplicate Database to an alternate server with the same directory structure

The following procedure details the steps to restore a Duplicate Database backup to an alternate or remote server—other than where the source database resides—and use the same directory structure as the source database.

- Prerequisites
- Restoring the Duplicate Database

Prerequisites

To restore a Duplicate Database backup to an alternate or remote server, the following prerequisites must be met:

- **Same version of Oracle database software**: The version and edition of Oracle for the destination database must be identical to the version and edition of Oracle being used for the source database. This requirement includes identical patch levels.

- **Duplicate Database backup available**: A Duplicate Database backup must be completed successfully and made available.

- **Source database in open state**: The source database must remain in an **OPEN READ WRITE** state during the entire Duplicate Database restore process.

- **Auxiliary instance prepared**: RMAN's backup-based Duplicate Database process requires the preparation of an auxiliary instance:

  1 **Oracle password file for auxiliary instance created**: Required if you want to use a password file versus OS authentication for the auxiliary connection when duplicating to the same host as the source database. For more information on creating a password file, see *Creating and Maintaining a Password File* in the *Oracle Database Administrator's Guide*.

  2 **Oracle Net connectivity established to auxiliary instance**: The auxiliary instance must be available through Oracle Net by adding the instance to the **"tnsnames.ora"** file of *both* the source server and the alternate server. Additionally, on Windows platforms, perform the following command to start the instance:

     `oradim -new -sid <sid_name>`

  3 **Auxiliary instance destination directories exist**: The following destination directories for the auxiliary instance must be created on the server where the destination database resides. If the destination database stores its datafiles in ASM, the ASM Instance name for the destination database must exist.

     - Control_Files
     - DB_Create_File_Dest
     - Audit_File_Dest
     - Background_Dump_Dest
     - User_Dump_Dest

  4 **PFILE created for auxiliary instance**: A client-side PFILE for the Auxiliary Database must be created from the SPFILE of the source database.

     - Start **SQL*Plus**, and connect with administrator privileges to the source database.

       `sqlplus sys/<password>@<source_connect_identifier> AS SYSDBA`

     - Create a PFILE from the SPFILE.

       ```
       create pfile = '<PFILE_destination_directory>/
          init<auxiliary_sid>.ora' from spfile;
       ```

  5 **PFILE updated with auxiliary values**: The PFILE created for the auxiliary instance must be updated with the parameter values specific to the auxiliary instance.

- Remove all entries that begin with: ***<source_sid>.___***

- Update the ***.db_name=** entry to reflect the name of the destination database.

- Update the following entries to reflect the destination directory for the auxiliary instance. This directory must exist; however, if the source database stores the datafiles in ASM, the ASM instance name for the destination database should be specified for each of the following **control_files** and **db_create_file_dest** parameters:

```
*.audit_file_dest=
*.background_dump_dest=
*.control_files=
*.user_dump_dest=
*.db_create_file_dest=
*.db_recovery_file_dest=
```

- Comment out the following entry by inserting a "#" (number sign) at the beginning of the following entry

```
*.dispatchers='(PROTOCOL=TCP) (SERVICE=<source_sid>XDB)'
```

6 **Auxiliary instance started in NOMOUNT state**: The auxiliary instance must be started in **NOMOUNT** state with the PFILE that has been updated with the parameter values specific to the auxiliary instance.

- Start **SQL*Plus**, and connect with administrator privileges to the auxiliary instance.

```
sqlplus sys/<password>@<auxiliary_connect_identifier> AS SYSDBA
```

- Start the instance in **NOMOUNT** state.

```
startup nomount pfile = '<PFILE_destination_directory>/
    init<auxiliary_sid>.ora'
```

7 **Auxiliary instance created from SPFILE from the PFILE.**

```
create spfile from pfile='<PFILE_destination_directory>/
    init<auxiliary_sid>.ora';
```

8 Exit **SQL*Plus**.

You must exit **SQL*Plus** for the Duplicate Database restore to complete successfully.

- **NetVault software and the Plug-in *for Oracle* installed**: The same version of NetVault software and the plug-in must be installed and configured on the alternate server where the destination database resides.

  When configuring the plug-in on the alternate server, ensure that the following Default Attributes are defined:

  - **NetVault:** Specify the name of the NetVault Server where the source database server was added as a NetVault Client.

  - **Do Restore From NetVault Server**: Specify the name of the NetVault Server where the source database server was added as a NetVault Client.

  - **Restore Backup taken from NetVault Client**: Specify the NetVault Machine Name for the source database server.

  For more information, see Configuring default settings.

- **Source database added to Plug-in *for Oracle* installed on alternate server**: The source database must be added to the plug-in that is installed on the alternate (standby) server where the destination database resides.

  For example, the production Oracle Server is named **salesdb**. On the alternate server that has the plug-in installed, add a database named **salesdb**. You complete this step even though the existing database has not yet been cloned to the alternate server. This step forces the plug-in to create a placeholder that can then be accessed during the cloning process when the Oracle database is restored to the alternate (standby) server.

For more information on adding a database, see Adding a database. These instructions work for the original database, as well as the placeholder that you create on the alternate server.

Restoring the Duplicate Database

1 On the **Create Restore Job—Choose Saveset** page, find the applicable **Duplicate Database** backup, and click **Next**.

2 On the **Create Selection Set** page, navigate to the source NetVault Client and database, select the **Whole Database** node, and click ✏ .

3 Click the **Clone Database** tab, and select the **Duplicate Database** option.

4 In the **Destination Database** section:

   ▪ Specify the Oracle Net Service name of the auxiliary instance in the **Target Service** field.

   ▪ Specify the **Oracle Home** for the auxiliary instance.

   ▪ Specify the **Oracle SYSDBA User Name** that is used by the plug-in to connect to the auxiliary instance.

   ▪ Specify the **Oracle SYSDBA Password**.

   ▪ Select the **Use nofilenamecheck** option.

5 In the **Duplicate From Database** section:

   ▪ Specify the **Oracle SID** for the source database.

   ▪ Specify the **Oracle Home** for the source database.

   ▪ Specify the **Oracle SYSDBA User Name** that is used by the plug-in to connect to the source database.

   ▪ Specify the **Oracle SYSDBA Password**.

6 In the **Target Client** list, select the NetVault Machine Name of the alternate server where the destination database resides.

7 Continue with the restore procedure, including completion of the **Schedule** and **Advanced Options** lists.

   For more information, see Finalizing and submitting the job.

Restoring a Duplicate Database to an alternate server with an alternate directory structure

The following procedure details the steps to restore a Duplicate Database backup to an alternate or remote server—other than where the source database resides—and use a different directory structure.

• Prerequisites

• Restoring the Duplicate Database

Prerequisites

To restore a Duplicate Database backup to an alternate or remote server with an alternate directory structure, the following prerequisites must be met:

• **Same version of Oracle database software**: The version and edition of Oracle for the destination database must be identical to the version and edition of Oracle being used for the source database. This requirement includes identical patch levels.

• **Duplicate Database backup available**: A Duplicate Database backup must be completed successfully and made available.

• **Source database in open state**: The source database must remain in an **OPEN READ WRITE** state during the entire Duplicate Database restore process.

- **Auxiliary instance prepared**: RMAN's backup-based Duplicate Database process requires the preparation of an auxiliary instance:

  1 **Oracle password file for auxiliary instance created**: Required if you want to use a password file versus OS authentication for the auxiliary connection when duplicating to the same host as the source database. For more information on creating a password file, see *Creating and Maintaining a Password File* in the *Oracle Database Administrator's Guide*.

  2 **Oracle Net connectivity established to auxiliary instance**: The auxiliary instance must be available through Oracle Net by adding the instance to the **"tnsnames.ora"** file of *both* the source server and the alternate server. Additionally, on Windows platforms, perform the following command to start the instance:

  ```
  oradim -new -sid <sid_name>
  ```

  3 **Auxiliary instance destination directories exist**: The following destination directories for the auxiliary instance must be created on the server where the destination database resides. If the destination database stores its datafiles in ASM, the ASM instance name for the destination database must exist.

  - Control_Files

  - DB_Create_File_Dest

  - Audit_File_Dest

  - Background_Dump_Dest

  - User_Dump_Dest

  4 **PFILE created for auxiliary instance**: A client-side PFILE for the Auxiliary Database must be created from the SPFILE of the source database.

  - Start **SQL*Plus**, and connect with administrator privileges to the source database.

  ```
  sqlplus sys/<password>@<source_connect_identifier> AS SYSDBA
  ```

  - Create a PFILE from the SPFILE.

  ```
  create pfile = '<PFILE_destination_directory>/
     init<auxiliary_sid>.ora' from spfile;
  ```

  5 **PFILE updated with auxiliary values**: The PFILE created for the auxiliary instance must be updated with the parameter values specific to the auxiliary instance.

  - Remove all entries that begin with: *<source_sid>.__*

  - Update the ***.db_name=** entry to reflect the name of the destination database.

  - Update the following entries to reflect the destination directory for the auxiliary instance. This directory must exist; however, if the source database stores the datafiles in ASM, the ASM instance name for the destination database should be specified for each of the following **control_files** and **db_create_file_dest** parameters:

  ```
  *.audit_file_dest=
  *.background_dump_dest=
  *.control_files=
  *.user_dump_dest=
  *.db_create_file_dest=
  *.db_recovery_file_dest=
  ```

  - Comment out the following entry by inserting a "#" (number sign) at the beginning of the following entry:

  ```
  *.dispatchers='(PROTOCOL=TCP) (SERVICE=<source_sid>XDB)'
  ```

  - Add the following line to specify the pairs of strings for converting the names of datafiles and tempfiles.

  ```
  *.db_file_name_convert='<source_db_create_file_dest>/
     <source_sid>', '<destination_db_create_file_dest>/
  ```

```
        <destination_sid>'
```

For example:

```
*.db_file_name_convert=' C:\oracle\oradata\prod',
    'c:\oracle\oradata\test'
```

- □ Add the following line to specify the pairs of strings for naming online redo log files:

```
*.log_file_name_convert='<source_db_create_file_dest>/
    <source_sid>', '<destination_db_create_file_dest>/
    <destination_sid>'
```

For example:

```
*.log_file_name_convert=' C:\oracle\oradata\prod',
    'c:\oracle\oradata\test'
```

6 **Auxiliary instance started in NOMOUNT state**: The auxiliary instance must be started in **NOMOUNT** state with the PFILE that has been updated with the parameter values specific to the auxiliary instance.

- □ Start **SQL*Plus**, and connect with administrator privileges to the auxiliary instance.

```
sqlplus sys/<password>@<auxiliary_connect_identifier> AS SYSDBA
```

- □ Start the instance in **NOMOUNT** state.

```
startup nomount pfile = '<PFILE_destination_directory>/
    init<auxiliary_sid>.ora'
```

7 **Auxiliary instance created from SPFILE from the PFILE.**

```
create spfile from pfile='<PFILE_destination_directory>/
    init<auxiliary_sid>.ora';
```

8 Exit **SQL*Plus**.

You must exit **SQL*Plus** for the Duplicate Database restore to complete successfully.

- **NetVault software and the Plug-in *for Oracle* installed**: The same version of NetVault software and the plug-in must be installed and configured on the alternate server where the destination database resides.

  When configuring the plug-in on the alternate server, ensure that the following Default Attributes are defined:

  - ▪ **NetVault:** Specify the name of the NetVault Server where the source database server was added as a NetVault Client.

  - ▪ **Do Restore From NetVault Server**: Specify the name of the NetVault Server where the source database server was added as a NetVault Client.

  - ▪ **Restore Backup taken from NetVault Client**: Specify the NetVault Machine Name for the source database server.

  For more information, see Configuring default settings.

- **Source Database Added to Plug-in *for Oracle* Installed on Alternate Server**: The source database must be added to the plug-in that is installed on the alternate (standby) server where the destination database resides.

  For example, the production Oracle Server is named **salesdb**. On the alternate server that has the plug-in installed, add a database named **salesdb**. Complete this step even though the existing database has not yet been cloned to the alternate server. This step forces the plug-in to create a placeholder that can then be accessed during the cloning process when the Oracle database is restored to the alternate (standby) server.

  For more information on adding a database, see Adding a database. These instructions work for the original database, as well as the placeholder that you create on the alternate server.

Restoring the Duplicate Database

1 On the **Create Restore Job—Choose Saveset** page, find the applicable **Duplicate Database** backup, and click **Next**.

2 On the **Create Selection Set** page, navigate to the source NetVault Client and database, select the **Whole Database** node, and click 🖉 .

3 Click the **Clone Database** tab, and select the **Duplicate Database** option.

4 In the **Destination Database** section:

- Specify the Oracle Net Service name of the auxiliary instance in the **Target Service** field.

- Specify the **Oracle Home** for the auxiliary instance.

- Specify the **Oracle SYSDBA User Name** that is used by the plug-in to connect to the auxiliary instance.

- Specify the **Oracle SYSDBA Password**.

- Select the **Use nofilenamecheck** option.

5 In the **Duplicate From Database** section:

- Specify the **Oracle SID** for the source database.

- Specify the **Oracle Home** for the source database.

- Specify the **Oracle SYSDBA User Name** that is used by the plug-in to connect to the source database.

- Specify the **Oracle SYSDBA Password**.

6 In the **Target Client** list, select the NetVault Machine Name of the alternate server where the destination database resides.

7 Continue with the restore procedure, including completion of the **Schedule** and **Advanced Options** lists.

For more information, see Finalizing and submitting the job.

Performing table-level recovery

Table-level recovery (RECOVER TABLE) is an Oracle-based feature that is available starting with Oracle Database 12.1 Enterprise Edition. If you are using the applicable version of Oracle, you can use the plug-in to recover specific Oracle tables to a specific point.

Oracle Server recovers the tablespaces that contain the tables listed for recovery into an auxiliary instance. Oracle Server then uses data pump to export the table data from the auxiliary instance to the target instance.

> **i** | **IMPORTANT:** The examples shown throughout the following topics are based on using the fully automated version of the auxiliary instance. By using the automated method, the auxiliary instance is fully created and managed by the Oracle Server whenever a destination, that is, a directory in the file system, is provided.

To restore tablespaces to the auxiliary instance, you must provide backups of the system tablespaces, the tablespaces that contain the tables to recover, and the controlfile as it existed at the time of the backup.

The plug-in lets you take advantage of the automated recovery method using an auxiliary instance. For information regarding other methods of performing table-level recovery, see your Oracle documentation. These other methods let you use the RMAN command prompt to recover from backups created by the plug-in.

The following topics provide information for performing a table-level recovery:

- Uses of table-level recovery
- Table-level requirements and limitations
- Configuring channels for table-level recovery
- Performing table-level recovery from the RMAN CLI
- Performing table-level recovery from the NetVault WebUI

Uses of table-level recovery

Table-level recovery lets you recover one or more tables or table partitions to a specific point without affecting other tables or database objects. You can use backups created using the plug-in's RMAN backup method to accomplish this process. In addition to other recovery methods, table-level recovery is useful when you need to recover:

- A few tables to a specific point.
- Tables that were dropped and purged.
- Tables that are logically corrupted.
- A table for which the **FLASHBACK TABLE** command is unavailable; for example, the Flashback Table cannot rewind because of structural changes, or the target point is older than the undo point.

Table-level requirements and limitations

There are Oracle database-specific limitations and requirements for performing this type of recovery. To perform recovery successfully, review the following requirements and limitations. For more information, including a complete list of constraints, see the table-level recovery information in your Oracle documentation.

- Table-level recovery is available in the Enterprise Edition of Oracle Server 12.1 or later.
- To use automated table-level recovery, the directory specified in the **"Auxiliary Destination"** clause must exist before running recovery. You can use an existing empty directory, or create a directory, if the directory exists before starting the process.
- You must use the **CONFIGURE** command to configure the channels in RMAN. The auxiliary database uses the same channels as the target database.
- Full backups of the tablespaces SYSTEM, SYSAUX, undo, that is, UNDOTBS1, and SYSEXT, if the SYSEXT tablespace is present, must exist. The tablespace that contains the table that is the target of the recovery must also exist.

- You cannot use table-level recovery to recover: tables that belong to SYS, tables from the SYSTEM and SYSAUX tablespaces, or tables on standby databases.

- You cannot use the **"REMAP"** clause to recover tables with named NOT NULL constraints.

Configuring channels for table-level recovery

To use the plug-in for recovery, configure SBT channels in RMAN. These channels connect RMAN and the NetVault media management utilities. You can configure a default SBT channel, as well as additional channels of type SBT or DISK. To display the current RMAN configuration, run the following command from an RMAN prompt:

```
RMAN> SHOW ALL;
```

If no SBT channels are configured, that is, only settings for DISK channels exist, use the following command to instruct RMAN to add a set of settings for SBT channels:

```
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO 'SBT_TAPE';
```

When these settings are added for SBT channels and a table-level recovery is run, at least one DISK channel and one SBT channel are available for the recovery process. The auxiliary instance uses the same channels that are available for the target instance.

You can also use the following command to add additional channels, if applicable:

```
RMAN> CONFIGURE CHANNEL <channelNumber> DEVICE TYPE sbt
```

For more information on configuring channels, see your Oracle documentation.

Performing table-level recovery from the RMAN CLI

As stated previously, the examples are based on using the fully automated version of the auxiliary instance. For additional examples on using an automated auxiliary instance or a user-managed auxiliary instance, both of which can be used with the plug-in to perform table-level recovery, see your Oracle documentation. Note the following:

- Table-level recovery requires the use of the **"until"** clause to define the state that you want to recover to.

- The following examples assume that you are connected to the target instance and, if applicable, the RMAN recovery catalog.

Examples

The following example recovers three tables of an Oracle user, **sales**, to a specific SCN using a fully automated auxiliary instance for which the required files are populated in a Linux or UNIX directory, **/oracle/nvbu_pitr_auxiliary_destination**, which you created before running recovery.

```
RMAN> recover table SALES.NORTHSALES, SALES.SOUTHSALES, SALES.CENTRALSALES
until scn 5555638
auxiliary destination '/oracle/nvbu_pitr_auxiliary_destination';
```

Similar example except that it uses a log sequence:

```
RMAN> recover table SALES.NORTHSALES, SALES.SOUTHSALES, SALES.CENTRALSALES
until sequence 38 thread 1
auxiliary destination '/oracle/nvbu_pitr_auxiliary_destination';
```

Similar example except that it uses a specific date and time, and it is based on a Windows Server environment:

```
RMAN> recover table SALES.NORTHSALES, SALES.SOUTHSALES, SALES.CENTRALSALES
until time "to_date('2013/11/23 06:59:00', 'yyyy/mm/dd hh24:mi:ss')"
auxiliary destination
'C:\app\Administrator\product\11.2.0\dbhome_1\nvbu_pitr_auxiliary_destination';
```

Using the **remap table** option, the following example recovers two of the three selected tables to a new table with a different name. The third table is recovered with the same name as the original.

```
RMAN> recover table SALES.NORTHSALES, SALES.SOUTHSALES, SALES.CENTRALSALES
until time "to_date('2013/11/23 06:59:00', 'yyyy/mm/dd hh24:mi:ss')"
```

```
remap table sales.northsales:northsales_recovered,
sales.southsales:southsales_recovered
auxiliary destination
'C:\app\Administrator\product\11.2.0\dbhome_1\nvbu_pitr_auxiliary_destination';
```

Recovers a table from a common user:

```
RMAN> recover table "C##SALES"."NORTHSALES", "C##SALES"."SOUTHSALES"
until scn 5555638
remap table "C##SALES"."NORTHSALES":"NORTHSALES_RECOVERED"
auxiliary destination '/oracle/nvbu_pitr_auxiliary_destination';
```

By default, RMAN imports the recovered tables or table partitions into the target database. Before import, the tables are stored in an export dump file. You can use the **NOTABLEIMPORT** option to prevent RMAN from importing the recovered tables or table partitions. If you use this option, the tables are recovered to the specified point, and the export dump file is created, but the dump file is not imported into the target database. At that point, you can analyze the tables using the auxiliary instance, or manually import the dump file into the target database instance using the Oracle Data Pump Import utility.

```
RMAN> recover table "C##SALES"."NORTHSALES", "C##SALES"."SOUTHSALES"
until time 'SYSDATE-4'
auxiliary destination '/oracle/nvbu_pitr_auxiliary_destination'
datapump destination '/oracle/nvbu_pitr_datapump_destination'
dump file 'northsales_southsales_export_dump.dat'
notableimport;
```

> ℹ️ **NOTE:** If errors occur during recovery, Oracle does not delete the export dump file if the recovered table created one. You can then use the export dump file to complete the recovery process manually.

Performing table-level recovery from the NetVault WebUI

In addition to the steps outlined in Performing RMAN restores, the following options apply specifically to table-level recovery.

1   In the **Recovery Type** section on the **Perform Recovery** tab, select the **Perform Table Level Point in Time Recovery** option.

2   Select the **System Change Number Based**, **Log Sequence Based**, or **Time Based** option.

- If you select the **System Change Number Based** option, the plug-in instructs RMAN to use the **"until scn"** clause during table recovery. For example: **until scn 5555638**

- If you select the **Log Sequence Based** option, the plug-in instructs RMAN to use the **"until sequence <number> thread <number>"** clause during table recovery. For example: **until sequence 38 thread 1**

- If you select the **Time Based** option, the plug-in instructs RMAN to use the **"until time"** clause during table recovery. For example: **until time "to_date('2013/11/23 06:59:00', 'yyyy/mm/dd hh24:mi:ss')"**

3   In the **Auxiliary Destination** field, specify a directory (full path) that the auxiliary instances use to store all the files needed, including copies of the controlfile, archive logs, and datafiles.

Use of this field is equivalent to the **"auxiliary destination '<directoryFullPath>'"** clause.

> ℹ️ **IMPORTANT:** Ensure that the Oracle user has access to the full path.

4   In the **Recover table** field, enter a comma-separated list of tables that you want to include in the recovery table as part of a table-level recovery.

Use the **oracle_user_name.table_name** construct. Oracle 12c includes common users and local users, and the naming convention used for the local users is similar to the Oracle database user names in earlier versions of Oracle.

For containers where common users are in place, the Oracle database supplies the SYS and SYSTEM common users. The name of user-created common users must begin with **C##** or **c#**. For more information, see the Oracle 12c Multitenant Architecture documentation.

> **i** | **IMPORTANT:** When listing a table from a common user, use double quotation marks. While **SQL*Plus** accepts queries on the tables using a string that includes **C##** or **c#** and excludes the double quotation marks, RMAN does not.

Example 1: The following lists three tables for the **sales** user.

```
sales.northsales, sales.southsales, sales.centralsales
```

Example 2: The following example lists two tables for the common **C##SALES** user.

```
"C##SALES"."NORTHSALES", "C##SALES"."SOUTHSALES"
```

Use of the preceding example is equivalent to the **recover table "C##SALES"."NORTHSALES", "C##SALES"."SOUTHSALES"** clause.

5   In the **Remap table** field, enter a comma-separated list of tables that you want to rename, if applicable, as a part of a table-level recovery.

This option lets you recover the table and analyze it for appropriateness to determine whether to overwrite the original table. You do *not* need to provide a mapping for every table listed in the **Recover table** field; you only need to list the mapping for those tables that you want to rename temporarily.

Example 1: The following maps the **sales.northsales** table to **sales.northsales_recovered**.

```
sales.northsales:northsales_recovered
```

Example 2: The following maps the **sales.northsales** table to **sales.northsales_recovered**.

```
sales.northsales:northsales_recovered, sales.southsales:southsales_recovered
```

Example 3: The following maps the **sales.northsales** table to **sales.northsales_recovered**.

```
"C##SALES"."NORTHSALES":"NORTHSALES_RECOVERED"
```

Use of the preceding example is equivalent to the **remap table "C##SALES"."NORTHSALES":"NORTHSALES_RECOVERED"** clause.

Maintaining the Recovery Catalog

- Resynchronizing the Recovery Catalog
- Using CROSSCHECK to update the RMAN repository

Resynchronizing the Recovery Catalog

Plug-in *for Oracle* automatically runs a full or partial resynchronization of the Recovery Catalog when performing RMAN backups as long the Control File is mounted and the Recovery Catalog database is available at command execution.

You can use the RMAN **RESYNC CATALOG** command to perform manual Full Resynchronizations when:

- The Recovery Catalog is unavailable when you issue any of the commands that automatically resynchronize the catalog.

- The database is backed up infrequently and generates a high number of log switches every day. For example, hundreds of archive logs are archived between database backups, and 1000 switches are generated between catalog resynchronizations

- You have changed the physical structure of the target database such as by adding or dropping a tablespace. As with archive operations, the Recovery Catalog is not updated automatically when the physical schema changes.

You should not need to run **RESYNC CATALOG** often. For more information, see *Using CROSSCHECK to Update the RMAN Repository* in the *Oracle Database Backup and Recovery Advanced User's Guide*.

To force a Full Resynchronization of the Recovery Catalog, perform the following steps.

1 Start RMAN, and connect to the target database and Recovery Catalog.

    > rman TARGET / CATALOG rman/<*password*>@<*connect_identifier for catalog*>

2 Mount the target database.

    STARTUP MOUNT;

3 Resynchronize the Recovery Catalog.

    RESYNC CATALOG;

4 Exit RMAN.

Using CROSSCHECK to update the RMAN repository

To ensure that data about backups in the RMAN Repository is synchronized with corresponding data in the NetVault Database, perform a crosscheck. The **CROSSCHECK** command updates outdated RMAN repository information about backups whose repository records do not match their physical status. For example, if a user removes archive logs from disk with an OS command, the repository still indicates that the logs are on disk, when in fact they are not.

For more information on the **CROSSCHECK** command, see the *Oracle Database Backup and Recovery Basics* guide.

1 Start RMAN, and connect to the target database and Recovery Catalog.

```
> rman TARGET / CATALOG rman/<password>@<connect_identifier for catalog>
```

2 Crosscheck the archive logs.

```
CROSSCHECK ARCHIVELOG ALL;
```

3 Exit RMAN.

Using the RMAN CLI

- CLI-based RMAN backups and restores: an overview
- Prerequisites for CLI-based backups and restores
- Performing CLI-based backups
- Performing CLI-based restores
- Using the CROSSCHECK and CATALOG commands with backups

CLI-based RMAN backups and restores: an overview

If you want use advanced RMAN features that are not supported through Plug-in *for Oracle*'s interface, you can perform CLI-based RMAN backups and restores instead of, or with, the plug-in. During CLI-based RMAN backups and restores, the plug-in provides media management functionality for RMAN.

> **i** | **IMPORTANT:** Before using this functionality, Quest recommends that you thoroughly review the *Oracle Database Backup and Recovery Advanced User's Guide* to familiarize yourself with the internals of RMAN.

Before completing any additional or different steps that are outlined in this topic, review the information provided in the previous topics, such as Installing and removing the plug-in and Configuring the plug-in.

Prerequisites for CLI-based backups and restores

The steps described in the following topics must be performed before initiating backups or restores from the command line:

- Verifying that the proper RMAN executable is in use (Linux only)
- Ensuring an "SBT_TAPE" channel is established
- Running the NetVault 'nvpluginaccess' utility

Verifying that the proper RMAN executable is in use (Linux only)

Various installations of Linux include more than one executable named as **rman**. Because of this inclusion, you must verify that the correct Oracle RMAN application is started when you issue the **rman** command at the prompt. This step can be done by making the path to the Oracle RMAN executable **($ORACLE_HOME/bin)** as the first search path in the **PATH** environmental variable.

Ensuring an "SBT_TAPE" channel is established

Plug-in *for Oracle* uses an Oracle **"SBT_TAPE"** channel to conduct backups of an Oracle database. SBT is commonly used to specify a destination for RMAN commands used to back up to tape media. For complete information on establishing this type of channel for use, see *Configuring and Allocating Channels for Use in Backups* in the *Oracle Database Backup and Recovery Advanced User's Guide*.

> **i** | **IMPORTANT:** Jobs that are generated using the RMAN CLI cannot be re-run using the NetVault WebUI. The RMAN CLI command, that is, the RMAN script, is not sent from the RMAN CLI application back to NetVault. So, if you select **Run Now** on the **Job Status** page, the job fails.

Running the NetVault 'nvpluginaccess' utility

You can run the **nvpluginaccess** utility, which is available in the **"util"** directory on the NetVault Server, from the NetVault Server or from a NetVault Client.

If the password for the NetVault default user has been set or a non-default NetVault user is used to run the backup and restore, you must run the **nvpluginaccess** utility on the NetVault Server or you must configure access for the NetVault Client. You must then obtain the necessary security clearance for the plug-in to perform CLI backups and restores.

> **!** | **CAUTION: The password is limited to seven characters or less.**

Configuring access for a NetVault Client

1   Initiate a command-line session, and run the following command:

```
nvpluginaccess -client <NetVaultServerName>
```

2   When the plug-in prompt appears, type **4** to select the **Specify plugin name** option.

> **i** | **IMPORTANT:** Do *not* select option 2 or 3; you must enter **CLI** for the name of the plug-in, which requires selection of option 4.

```
Please select a plugin you wish to allow client '<NetVaultServerName>'
  to access:
1) Informix plugin
2) Oracle RMAN plugin
3) Informix and Oracle RMAN plugins
4) Specify plugin name
```

3   When the name prompt appears, type **CLI**.

```
Please input the plugin name: CLI
```

4   When the user-account prompt appears, type **default**.

```
Please enter the user account: default
```

5   When the password prompt appears, enter the applicable password.

```
Please enter the password for the account 'default': <password>
```

The system indicates that the client was added.

```
Successfully added client '<NetVaultServerName>'.
```

Running the utility from the NetVault Server

1   Initiate a command-line session, and navigate to the **"util"** directory:

- On Linux and UNIX, the **"util"** directory is in **"…/netvault/util"**

- On Windows, the **"util"** directory is in **"…\NetVault\util"**

where **"…"** represents the path to the NetVault Server installation directory.

2 Run the following commands:

```
nvpluginaccess -remove -client <clientMachineName>
nvpluginaccess -client <clientMachineName> -account <userName>
```

where *<clientMachineName>* is the name of the NetVault Client machine on which the plug-in is installed and *<userName>* is the NetVault user who performs the backups and restores.

Example:

```
nvpluginaccess -client DB_Server -account default
```

3 When the list of available plug-ins appears, enter the number that corresponds to the plug-in ("Oracle RMAN Plug-in"), and press **Enter**.

4 When the password prompt appears, enter the password for the specified NetVault account.

A confirmation message appears stating that access has been granted for the specified NetVault Client.

5 Verify that the **nvpluginaccess** utility has been correctly configured by reviewing the contents of the **"hookplugs.cfg"** file located in the **"config"** directory:

- On Linux and UNIX, the **"config"** directory is in **"…/netvault/config"**

- On Windows, the **"config"** directory is in **"…\NetVault\config"**

where **"…"** represents the path to the NetVault Server installation directory.

The following entry in the "hookplugs.cfg" file corresponds to the preceding example.

```
[DB_SERVER]
Plug-ins=Oracle Rman!,Oracle Rman Plug-in
Account=default
Password=%G59j$#
```

> ⓘ **NOTE:** The **Password** field in the "hookplugs.cfg" file contains an encrypted view of the password. The actual password is not displayed.

Performing CLI-based backups

The following topics provide information for performing CLI-based backups:

- Example of a basic CLI backup
- More parameters: the "format" clause
- More parameters: the "send" and "PARMS" clauses
- Querying the NetVault media name

Example of a basic CLI backup

This topic shows how to take a Full Backup of the "USERS" tablespace using RMAN commands:

1 Log on to the Oracle Database Server, and initiate a terminal session.

2 From the prompt, generate a **"Run Block"** as follows:

```
run {
allocate channel c1 device type 'SBT_TAPE';
backup
```

```
      tablespace USERS;
      release channel c1;
      }
```

The preceding commands are the minimum required to perform a CLI-based backup. The plug-in provides several additional parameters that can be used to set different options for a CLI backup job. The topics that follow provide a description of these parameters.

More parameters: the "format" clause

The **"format"** clause specifies the pattern to use in creating a filename for the backup pieces or image copies that it creates. The plug-in provides additional parameters that can be used as variables along with the **"format"** clause. These variables let you specify three parameters for a backup job:

- Server Name
- Target Set
- Advanced Options Set

The **"format"** clause is used within a run block and is specified immediately after the **"backup"** command in the syntax. Use of the additional colon before the *"%d_%u_%p_%c"* variable is required; use of the *"%d_%u_%p_%c"* variable is critical because Oracle usually generates a unique string that you can use to identify the separate backup segments in the NetVault Media Manager.

```
run {
allocate channel c1 type 'SBT_TAPE';
backup
format '<Variable_1>:<Variable_2>:<Variable_3>::%d_%u_%p_%c'
tablespace USERS;
release channel c1;
```

Specific variables are included with this clause, each following a designated sequence, for example, *"<Variable_1>"*, *"<Variable_2>"*, and *"<Variable_3>"*. Each corresponds to specific NetVault and RMAN values, and all are outlined in the following table. Ensure that each variable in the command is separated by a colon (":") and that a second colon is inserted before the *"%d_%u_%p_%c"* variable.

Table 2. Variables

Variable	Description
<Variable_1>	**NetVault Server Name**: The NetVault Machine Name of the NetVault Server from which you run the backup job. This value is *required*.
<Variable_2>	**Target Set Name**: To target a specific device during backup, specify a Target Set created for use with Oracle RMAN backups. Leave this option blank if you want to use NetVault's default setting of **"Any Device."** This option overrides the Target Set specified on the **Configure** dialog box. For more information on Target Sets, see the discussion of **Selection Sets** in the *Quest NetVault Administrator's Guide*.

Table 2. Variables

Variable	Description
<Variable_3>	**Advanced Options Set Name**: Specify an Advanced Options Set created for use with Oracle RMAN backups. Leave this option blank if you want to use NetVault's default Advanced Options settings. This option overrides the Advanced Options Set specified on the **Configure** dialog box.
	When defining the **CLI Backup Advanced Options Set**, the **Discard After Days/Weeks/Years** option must be specified instead of **Discard After Full Backups** in the **Backup Life** section of the **Advanced Options** tab. CLI-based RMAN backups do not support backup-life generation for Full Backups, and backups might expire unexpectedly if the **Discard After Full Backups** option is specified. For more information on Advanced Option Sets, see the discussion of **Selection Sets** in the *Quest NetVault Administrator's Guide*.
"%d_%u_%p_%c"	These values pertain to unique RMAN identifiers; that is, they are Oracle environmental variables. Oracle uses this variable to generate a unique string that can be used to identify the separate backup segments in the NetVault Media Manager.
	• **%d**: Specifies the name of the database.
	• **%u**: Specifies an eight-character name made up of a compressed representation of the backup set or image copy number and the time the set or copy was created.
	• **%p**: Specifies the piece number within the backup set.
	• **%c**: Specifies the copy number of the backup piece within a set of duplicated backup pieces.
	NOTE: As an alternative, you can use *"%d_%s_%t"* or another combination of keywords that generates a unique identifier. For substitution variables that you can use with the **formatSpec** subclause, see your Oracle documentation.

More parameters: the "send" and "PARMS" clauses

Use the parameters described in this topic with either the **"send"** or the **"PARMS"** command to set up the NetVault options for a backup job run from the command line. This topic describes how the **"send"** and the **"PARMS"** clauses work.

- **"send" clause**: The parameters specified with the **"send"** clause are applied to a specific job.

  ```
  send '<Variable>=<Value>';
  ```

  The following shows an example:

  ```
  run { allocate channel ch1 type 'SBT_TAPE'
  send 'NV_TARGET_MEDIA=Mid' }
  run { allocate channel ch2 }
  ```

  In this example, the **"NV_TARGET_MEDIA"** command affects only the "SBT_TAPE" Channel **"ch1"**.

  The **"send"** clause is used within a run block and is specified immediately after the **"allocate"** command in the syntax.

  ```
  run {
  allocate channel c1 type 'SET_TAPE';
  send 'NV_VERIFICATION=TRUE';
  backup
  tablespace USERS;
  release channel c1;
  ```

- **"PARMS" clause**: The parameters specified with the **"PARMS"** clause are applied to all the jobs run from the Oracle Instance.

  ```
  PARMS="ENV=(<Variable>=<Value>)"
  ```

The following shows an example:

```
run { allocate channel ch1 type 'SBT_TAPE' PARMS="ENV=(NV_TARGET_MEDIA=Mid)" }
run { allocate channel ch2 }
```

In this example, the **"NV_TARGET_MEDIA"** command affects the "SBT_TAPE" Channel **"ch1"** and any other channels set in the other run blocks.

The **"PARMS"** clause is used within a run block and is specified immediately after the **"allocate"** command in the syntax.

```
run {
allocate channel c1 type 'SBT_TAPE'
PARMS="ENV=(NV_VERIFICATION=TRUE)";
backup
tablespace USERS;
release channel c1;
```

Variables available with "send" and "PARMS"

The following table describes the variables available for use with the **"send"** and **"PARMS"** clauses:

Table 3. Variables

Variable	Description
NETVAULTCLIACCOUNT NETVAULTCLIPASSWORD	Use these variables to set the NetVault user and password that performs the backup or restore. These variables must be set when a password has been defined for the default NetVault user or a non-default user performs the backup or restore. **Example:** `send 'NETVAULTCLIACCOUNT=default'; send 'NETVAULTCLIPASSWORD=<password>'; PARMS="ENV=(NETVAULTCLIACCOUNT=default, NETVAULTCLIPASSWORD=<password>)";` **NOTE:** Use of the **NETVAULTCLIPASSWORD** variable is a security risk. As an alternative, Quest recommends using the **nvpluginaccess** utility described in Running the NetVault 'nvpluginaccess' utility. If the **NETVAULTCLIPASSWORD** variable is implemented, it displays the NetVault password in plain text.
NV_ADVANCED_OPTIONS_SET	Use this variable to specify a template for an **Advanced Options Set** created for use with Oracle RMAN CLI backups to target a specific set of NetVault Advanced Options during backup. This parameter overrides the Advanced Options Set that you specified on the **Configure** dialog box. **Example:** `send 'NV_ADVANCED_OPTIONS_SET= my_advanced_options_set_name'; PARMS="ENV=(NV_ADVANCED_OPTIONS_SET=my_advanced_options _set_name)";` For more information on Advanced Option Sets, see the discussion of **Selection Sets** in the *Quest NetVault Administrator's Guide*. **NOTE:** NetVault supports specifying a template of an Advanced Options Set while simultaneously specifying individual Advanced Options, for example, NV_VERIFICATION. If Backup Advanced Options are provided using Send/PARMS clauses as well as using a defined Advanced Option Set, both Advanced Options are combined. In case of conflict, Advanced Options provided using Send/PARMS clauses are given priority over the same options defined in the Advanced Option Set.

Table 3. Variables

Variable	Description

NV_AUTO_LABEL_MEDIA	Use this variable to label and use any blank media automatically when no other media is available for the job. The permissible values are:
	
	• **True**
	• **False**
	
	Example:
	
	`send 'NV_AUTO_LABEL_MEDIA=TRUE';`
	`PARMS="ENV= (NV_AUTO_LABEL_MEDIA=TRUE)";`

NV_BACKUP_LIFE	Use this variable to control how long a backup exists on its target media. The permissible values are:
	
	• **count {full | days | weeks | years}**
	
	The parameters within curly braces are optional. **Count** followed by **full** determines the number of generations that the backup is kept. This parameter is the default, that is, if nothing is specified with count, the number of backup generations is considered for the backup life. **Days**, **weeks**, or **years** represents the number of days, weeks, or years that the backup is kept before it is discarded.
	
	Example:
	
	`send 'NV_BACKUP_LIFE=2';`
	`PARMS="ENV=(NV_BACKUP_LIFE=2)";`

NV_BACKUP_SERVER	Use this variable with the "PARMS" clause to establish the default NetVault Server for an entire Oracle Instance. **Example:** `PARMS="ENV=(NV_BACKUP_SERVER=NVSERVER)"`

NV_GROUP_LABEL	Use this variable to select a specific group of media based on its group label. This variable must be used with the **"NV_TARGET_MEDIA"** variable; otherwise, it has no effect.
	
	Example:
	
	`send 'NV_TARGET_MEDIA=Group Label,`
	`NV_GROUP_LABEL=Group_2';`
	`PARMS="ENV=(NV_TARGET_MEDIA=Group`
	`Label,NV_GROUP_LABEL=Group_2)";`

NV_JOB_PRIORITY	Use this variable to set the NetVault Job Priority for the CLI task. The value input can range from 1 (highest priority) to 100 (lowest priority). If this parameter is not specified, the default job priority is 30.
	
	Example:
	
	`send 'NV_JOB_PRIORITY=5';`
	`PARMS="ENV=(NV_JOB_PRIORITY=5)";`

NV_JOB_TITLE	Use this variable to assign a title to the job. Quest recommends that only use this variable with the **"send"** command on a per job basis, that is, to avoid the chance of all future jobs being labeled with this job title, when applied with the **"PARMS"** clause. **Example:** `send 'NV_JOB_TITLE=Backup_1';`

Table 3. Variables

Variable	Description

NV_MID_LABEL	Use this variable to select a specific piece of media based on its media label. This variable must be used with the **"NV_TARGET_MEDIA"** variable; otherwise, it has no effect.
	
	Example:
	
	`send 'NV_TARGET_MEDIA=Mid, NV_MID_LABEL=Media_1';`
	`PARMS="ENV=(NV_TARGET_MEDIA=Mid, NV_MID_LABEL=Media_1)"`

NV_NETWORK_COMPRESSION	Use this variable to compress backup data before it is transferred over the network, that is, in the same manner as the **Advanced Options** tab's **"Network Compression"** option. The permissible values are:
	
	• **True**
	• **False**
	
	Example:
	
	`send 'NV_NETWORK_COMPRESSION=True';`
	`PARMS="ENV=(NV_NETWORK_COMPRESSION=False)";`

NV_ORIGINAL_NV_CLIENT	Use this variable with the **"PARMS"** clause to establish the machine name of the original NetVault Client Machine Name where the backup was performed.
	
	Example:
	`PARMS="ENV=(NV_ORIGINAL_NV_CLIENT=`
	`<nv_client_machine_name>)"`

NV_POSTSCRIPT	Use this variable to run a script after completion of a backup job. The script to run must be saved in the **"…\netvault\scripts"** subdirectory on the Oracle Database Server, where "…" refers to the complete path to the installation of NetVault.
	
	Example:
	
	`send 'NV_POSTSCRIPT=C:\NetVault\scripts\p2.txt';`
	`PARMS="ENV=(NV_POSTSCRIPT=p2.txt)";`

NV_PRESCRIPT	Use this variable to run a script before a backup job. The script to run must be saved in the **"…\netvault\scripts"** subdirectory on the Oracle Database Server, where "…" refers to the complete path to the installation of NetVault.
	
	Example:
	
	`send 'NV_PRESCRIPT=C:\NetVault\scripts\p1.txt';`
	`PARMS="ENV=(NV_PRESCRIPT=p1.txt)";`

NV_RESET_ENV_PARMS	Use this variable to reset the Oracle environment after the **"PARMS"** clause has been used in the run block.
	
	Example:
	
	`send 'NV_RESET_ENV_PARMS=TRUE'`
	`PARMS="ENV=(NV_RESET_ENV_PARMS=TRUE)"`

Table 3. Variables

Variable	Description

NV_REUSE_MEDIA	Use this variable to set a specific media as reusable. The permissible values are:
	
	• **Never**
	• **Any**
	• **With same group label as target media**
	
	If the value is set as **Never**, no media marked for reuse is used for the backup. With the parameter set as **Any**, any media previously marked for reuse is considered for the backup. If the value is set as **With same group label as the target media**, any media containing the specified group label that has been previously marked for reuse is considered for the backup.
	
	Example:
	
	`send 'NV_REUSE_MEDIA=Any';`
	`PARMS="ENV=(NV_REUSE_MEDIA=Any)";`

NV_RESTORE_SOURCE_SET	Use this variable to specify a Restore Device Source Set created for use with Oracle RMAN CLI restores. This variable lets you target a specific NetVault set of devices and device options during a restore. Do *not* use this option if you want to use the default NetVault setting of Any Device.
	
	NOTE: To use this variable, specify a Restore Device Source Set that was created from the **Restore Source** option on the **Restore Options** tab. Do not use NetVault Target Sets. A NetVault Target Set is used to specify target devices during a backup. A NetVault Restore Source Set is used to specify a set of source devices and devices options during a restore.
	
	Set this variable through the "**send**" or **"PARMS"** clause in the RMAN CLI.
	
	• **Using "send"**: The specified parameters apply only to a specific job.
	
	`send '<Variable>=<Value>';`
	Example:
	
	`run {`
	`  allocate channel ch1 type 'sbt_tape';`
	`  send 'NV_RESTORE_SOURCE_SET=`
	`    my_device_options_source_set_name';`
	`  restore tablespace 'TEST1';`
	`}`
	
	• **Using "PARMS"**: The specified parameters apply to all jobs run from the Oracle Instance.
	
	`PARMS="ENV=(<Variable>=<Value>)`
	Example:
	
	`run {`
	`  allocate channel ch1 type 'sbt_tape'`
	`    PARMS="ENV=(NV_RESTORE_SOURCE_SET=`
	`      my_device_options_source_set_name)";`
	`  restore tablespace 'TEST1';`
	`}`

Table 3. Variables

Variable	Description

NV_SCHEDULE_SET	Use this variable to specify a Schedule Set created for use with Oracle RMAN CLI backups to specify or reuse a pre-existing schedule set during backup.
	
	Example:
	
	`send 'NV_SCHEDULE_SET=my_schedule_set';`
	`PARMS="ENV=(NV_SCHEDULE_SET=my_schedule_set)";`
	
	For more information on Schedule Sets, see the *Quest NetVault Administrator's Guide*.

NV_TARGET_MEDIA	Use this variable to set the media in the targeted device. The permissible values are:
	
	• **Any not in a group**: To target any media that does not have a Group Label assigned to it.
	• **Any**: To target any available media.
	• **MID**: To target a specific piece of media based on its media label, in which case the **NV_MID_LABEL** variable has to accompany this variable, and the applicable piece of media must be named.
	• **Group Label**: To target the media within a specific group based on its Group Label, in which case the **NV_GROUP_LABEL** variable has to accompany this variable, and the applicable Group Label must be named.
	
	Example:
	
	`send 'NV_TARGET_MEDIA=Any';`
	`PARMS="ENV=(NV_TARGET_MEDIA=Mid,`
	`NV_MID_LABEL=Media_1)";`

NV_TARGET_SET	Use this variable to specify a Target Set created for use with Oracle RMAN CLI backups to target a specific NetVault device during backup. Ignore this option if you want to use the NetVault default setting of **"Any Device."** This parameter overrides the Target Set specified on the **Configure** dialog box.
	
	Example:
	
	`send 'NV_TARGET_SET=my_target_set_name';`
	`PARMS='ENV=(NV_TARGET_SET=my_target_set_name)";`
	
	For more information on Target Sets, see the discussion of **Selection Sets** in the *Quest NetVault Administrator's Guide*.

NV_VERIFICATION	Use this variable to verify completion of a backup job. The permissible values are:
	
	• **True**
	• **False**
	
	If set to **True**, the data transfer is checked and a job message is added to the NetVault Job log stating **"Backup job has verified successfully."**
	
	Example:
	
	`send 'NV_VERIFICATION=TRUE';`
	`PARMS="ENV=(NV_VERIFICATION=TRUE)";`

Resetting the RMAN environment

After the **"PARMS"** clause has been used, use either of the following lines of syntax to reset the RMAN environment:

```
send 'NV_RESET_ENV_PARMS=TRUE'
PARMS="ENV=(NV_RESET_ENV_PARMS=TRUE)"
```

> **i** | **IMPORTANT:** Close any terminal session or window in which you have set the **NETVAULTCLIACCOUNT** and **NETVAULTCLIPASSWORD** variables.

Querying the NetVault media name

For RMAN CLI backups performed through the plug-in, you can use **SQL*Plus** to determine the physical media on which the RMAN backup pieces are stored. The following example shows how to query the media information using Oracle **SQL*Plus**.

Example

```
SQL> select handle, media from v$backup_piece;
HANDLE
------------------------------------------------------------------
MEDIA
------------------------------------------------------------------
NVBU_Server:#642:NVBU_Server:1_692655825:RMAN INCRMTAL LVL 0 SALESDB_1248060220 MMS
NetVault Tape
NVBU_Server:#642:NVBU_Server:2_692655843:RMAN INCRMTAL LVL 0 SALESDB_1248060220 MMS
NetVault Tape
NVBU_Server:#642:NVBU_Server:3_692655935:RMAN INCRMTAL LVL 0 SALESDB_1248060220 MMS
NetVault Tape
```

SQL*Plus uses the following format to display the media information:

- If the media has a barcode identified by NetVault, the format is:

  ```
  (tape_barcode) tape_media_label
  ```

  Sample Output from RMAN CLI

  ```
  SQL> select handle, media from v$backup_piece;
  [...]
  HANDLE
  --------------------------------
  MEDIA
  --------------------------------
  oml4dtid_1_1
  (05A999L3) medialabel
  ```

- If the media does not have a bar code identified by NetVault, the format is:

  ```
  tape_media_label
  ```

  Sample Output from RMAN CLI

  ```
  SQL> select handle, media from v$backup_piece;
  [...]
  HANDLE
  ----------------------------------------------------
  MEDIA
  ----------------------------------------------------
  okl4dri2_1_1
  (NONE) medialabel
  ```

  Note the following traits regarding the string shown in the **media** field:

  - The **tape_barcode** is shown in parentheses.

  - If there is no **tape_media_label**, the media label defaults to **"MMS NetVault Tape"**.

  - If a backup piece spans multiple tapes, only the information from one tape is presented.

- The string is truncated to 60 bytes—60 characters in English. Barcodes are usually 6 bytes or 8 bytes, but the media label can be of variable length. By keeping the media label to a reasonable length, you can ensure that the entire string is shown.

• If you perform RMAN backups from the NetVault WebUI, the format is:

```
MMS NetVault Tape
```

Sample output:

```
SQL> select handle, media from v$backup_piece;
[…]
HANDLE
-----------------------------------------------------
MEDIA
-----------------------------------------------------
NVSERVER:#376:NVCLIENT:792_709294286:RMAN INCRMTAL LVL 0 ORCL_1264468284
MMS NetVault Tape
```

Performing CLI-based restores

This form of restore is performed using the RMAN CLI Utility. Through this utility, you can perform restores of backups, regardless of how they were created, that is, either through the plug-in or from the command line. For more information on the RMAN commands, see the *Oracle Database Backup and Recovery Reference Guide*.

> **i** | **IMPORTANT:** Although this form of restore is initiated locally from a terminal session on the Oracle Database Server, you can monitor job progress from the **Job Status** page in the NetVault WebUI. However, this option applies only to data restores from the media contained in devices controlled by the NetVault Server.

Using the CROSSCHECK and CATALOG commands with backups

Using the Oracle RMAN CLI, you can use the **CROSSCHECK** and **CATALOG** commands in combination with the NetVault Media Manager repository to check information about the backup pieces stored by the Media Manager. Before you run the CROSSCHECK or CATALOG command, make sure that you have allocated SBT channels to communicate with the NetVault Media Manager. You can allocate SBT channels within a run block for CROSSCHECK or CATALOG commands. For CROSSCHECK, you also can allocate SBT maintenance channels.

Using CROSSCHECK

You can use the **CROSSCHECK** command to verify that the backup pieces recorded in the recovery catalog or control file correspond with the backup pieces stored in the NetVault Media Manager repository. **CROSSCHECK** also updates the RMAN catalog for the backup pieces that are no longer stored, or have expired, in the repository. Be aware that the **CROSSCHECK** command does not expire or remove NetVault savesets, nor does it remove RMAN catalog records for backup pieces that are unavailable when the command is run. **CROSSCHECK** only updates the RMAN catalog records with the status of the backup pieces. To remove records of expired backup pieces from the catalog, use the **DELETE** command.

CROSSCHECK marks backup pieces that are not available in the NetVault repository as EXPIRED. In addition, if the NetVault repository itself is not available or cannot be accessed, **CROSSCHECK** also marks the backup pieces as EXPIRED.

> **i** | **IMPORTANT:** Make sure that you run the **CROSSCHECK** command in its own RMAN CLI session. For example, if you have opened an RMAN CLI session to run CLI backup or restores jobs, exit the session, and start a new session to run the applicable **CROSSCHECK** command.

Examples of running CROSSCHECK

To check all backups:

```
RMAN> run { allocate channel ch1 type sbt; CROSSCHECK backup; }
```

To allocate a maintenance channel and check a backup piece:

```
RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE sbt;
RMAN> CROSSCHECK backuppiece 'auu54p3g_1_1';
```

Using CATALOG

You can use **CATALOG** to register backups pieces, image copies, archived files, and datafile copies in the RMAN Catalog Repository. Whenever you perform a backup using the RMAN backup method, the backup pieces created by the backup job are cataloged. However, there might be occasions for which you need to catalog expired backup pieces to add them back into the RMAN catalog. For example, copies of backup pieces that exceed the retention-policy period become obsolete and are not accessible through the Oracle RMAN catalog. In this case, you can use the **CATALOG** command to catalog them and add them back into to the catalog. You can also use the command to catalog backup pieces that were moved, removed, or renamed, or that pertain to a difference Oracle Server.

If a backup piece is already cataloged, using the **CATALOG** command does not change the Oracle RMAN catalog. Use of the CATALOG command is only intended to register backup pieces that are stored in the NetVault Media Manager repository or on a disk. If you attempt to catalog a piece that no longer exists in the repository or on the disk, the CATALOG command fails.

Example of running CATALOG

To catalog a backup piece:

```
RMAN> run { allocate channel ch1 type sbt; catalog backuppiece 'auu54p3g_1_1'; }
```

> **i** | **IMPORTANT:** You must issue the CATALOG command inside a run block.

Some versions of Oracle require that the catalog of the backup pieces that were created using an SBT device type, such as those created with NetVault, are cataloged outside of a run block with automatic channels. In this case, configure the default device type to SBT, and catalog the backup piece or pieces. After the cataloging is completed, clear the default device type to set it to default. The following example illustrates these steps:

```
RMAN> configure default device type to SBT;
RMAN> catalog device type SBT backuppiece 'auu54p3g_1_';
RMAN> configure default device type clear;
```

Using the plug-in with Oracle RAC

- Oracle RACs: an overview

- Recommended additional reading

- Defining a deployment strategy

- Licensing

- Installing the plug-in in a RAC environment

- Configuring the plug-in

- Backing up data

- Restoring data in a RAC environment

Oracle RACs: an overview

Oracle defines its **Real Application Clusters (RAC)** option as "support for the transparent deployment of a single database across a cluster of servers, providing fault tolerance from hardware failures or planned outages. **Oracle RAC** running on clusters provides Oracle's highest level of capability in terms of availability, scalability, and low-cost computing. **Oracle RAC** supports mainstream business applications of all kinds. Support includes online transaction processing (OLTP), decision support systems (DSS), and Oracle's unique ability to support mixed OLTP/DSS environments. Support also includes packaged products such as SAP, PeopleSoft, Siebel, and Oracle E*Business Suite, as well as custom applications."

Plug-in *for Oracle* provides support for a limited number of Oracle versions and platforms in a RAC environment. The installation, configuration, backup, and restore procedures differ in a multi-instance RAC environment. This topic offers specifics on how the plug-in operates in an Oracle RAC environment. It only points out the *differences* between the setup and usage of the plug-in in a multi-instance RAC environment vs. a single-instance or traditional non-RAC environment.

> **i** | **IMPORTANT:** Unless outlined in the topics that follow, backups and restores performed with the plug-in for multi-instance RAC databases are the same as backups and restores performed with single-instance databases.

For a list of supported Oracle versions and platforms in a RAC environment, see the *Quest NetVault Compatibility Guide*.

Recommended additional reading

Quest recommends that the following Oracle RAC documentation be readily available for reference when setting up and using the plug-in in a RAC environment. Use the following link to find documentation for your version of Oracle environment: https://docs.oracle.com/en/database/oracle/oracle-database/index.html. On the Oracle website, look for the following documents and topics.

- Administration and Deployment Guides for your version of Oracle Clusterware and Oracle Real Application Clusters. Be sure to review topics such as:

  - An overview of Oracle Real Application Clusters management tools

- Starting and stopping instances and Oracle Real Application Clusters Databases

- Oracle Clusterware command-line reference

- Database Administrator's and Automatic Storage Management Guides. Review information on using Automatic Storage Management (ASM) and on using the ASM Command-Line Utility (ASMCMD).

Defining a deployment strategy

Deploying Plug-in *for Oracle* in a multi-instance database environment is almost identical to a single-instance database environment except that RMAN-based backups are the only available backup method for RAC databases. The plug-in is installed on one of the nodes that hosts one of the instances for the RAC-enabled database. All RMAN-based backups and restores are performed from this single installation of the plug-in. The following topics detail how the plug-in performs RMAN backups and restores in a RAC environment.

RMAN backup process for RAC environment

Figure 4. RMAN backup process for RAC environment
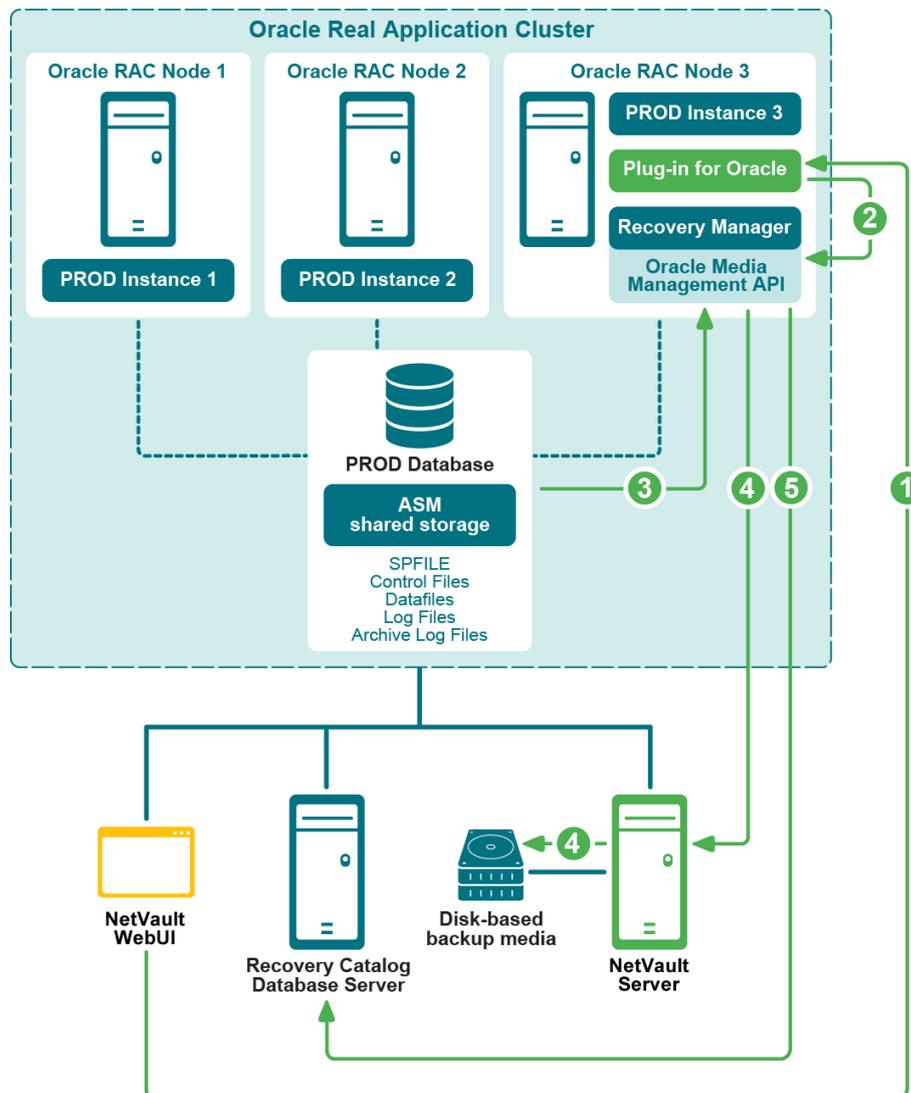
1 You define the backup job in the NetVault WebUI.

2 Plug-in *for Oracle* converts the job definition into the corresponding RMAN backup commands.

3 The RMAN runs the backup commands, which read data from the Oracle database datafiles.

4 The RMAN uses the Oracle Media Management API to store the backup on the backup device managed by the NetVault Server.

5 The RMAN stores the backup metadata in the Recovery Catalog repository.

RMAN restore process for RAC environment

Figure 5. RMAN restore process for RAC environment
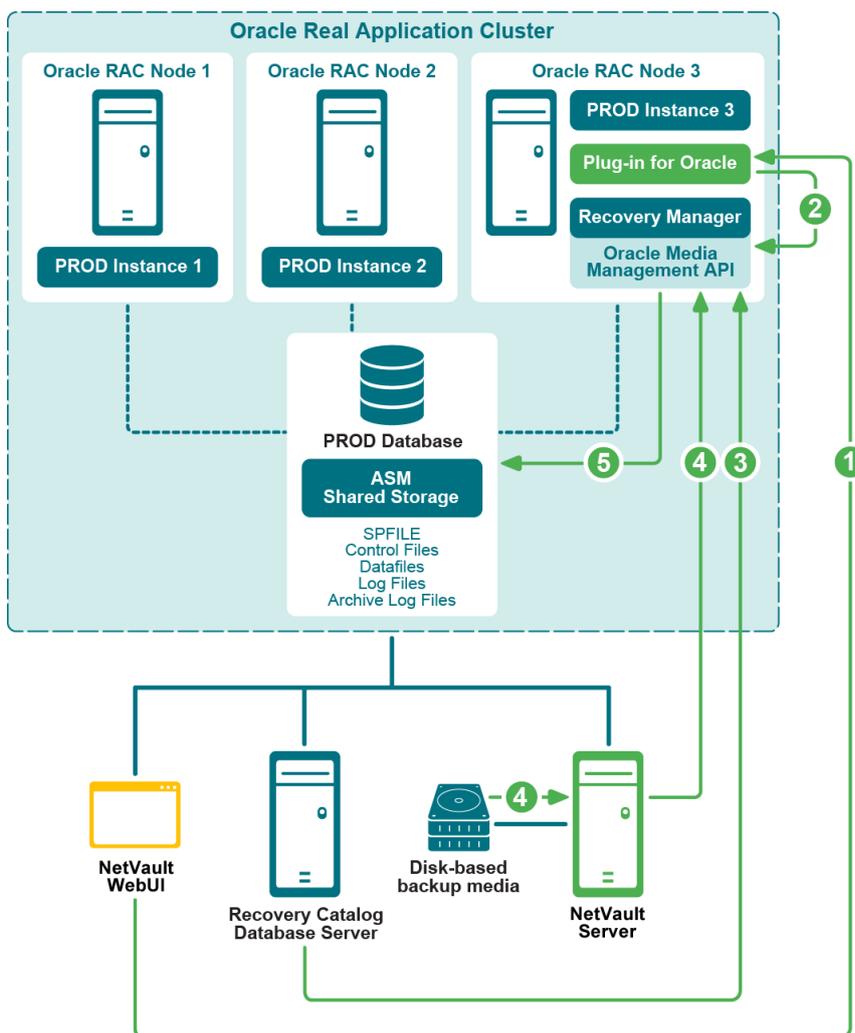
1 You define the restore job in the NetVault WebUI.

2 Plug-in *for Oracle* converts the job definition into corresponding the RMAN restore and recovery commands.

3 The RMAN reads the backup metadata from the Recovery Catalog repository.

4 The RMAN uses the Oracle Media Management API to read the backup from the backup device managed by the NetVault Server.

5 The RMAN restores the datafiles to the Oracle server and recovers the database.

Licensing

Licensing of the plug-in in a single-instance, or non-RAC environment, differs from the licensing of the plug-in in a multi-instance or RAC environment.

- In a single-instance environment, one **Edition-Based Plug-in *for Oracle*** license lets you back up and recover an unlimited number of Oracle databases that reside on a single Database Server.

- In a multi-instance or RAC environment, one **RAC Database-Based Plug-in *for Oracle*** license lets you back up and recover a single database within the single RAC environment. This guideline applies no matter how many instances or nodes reside in the RAC environment. If you have five databases residing in the single RAC environment, purchase five **RAC Database-Based Plug-in *for Oracle*** licenses, and so on.

Each RAC environment needs its own licensing. For example, if the production RAC environment has five databases and the test RAC environment has one database, five licenses are required for the production environment and one license is required for the test environment.

To request permanent **RAC Database-Based Plug-in *for Oracle*** license files, go to https://support.quest.com/contact-us/licensing.

Installing the plug-in in a RAC environment

Before completing any additional or different steps that are outlined in this topic, review the information provided in earlier topics, such as Installing and removing the plug-in and Configuring the plug-in.

- RAC configuration requirements
- Installation prerequisites
- Installing or upgrading the plug-in

RAC configuration requirements

Plug-in *for Oracle*'s support for multi-instance RAC databases is limited to environments that meet the following criteria:

- The version of Oracle and the platform are listed as a supported RAC environment in *Quest NetVault Compatibility Guide*.

- Oracle Clusterware is used as the cluster software.

- Datafiles, Control Files, SPFILE, Log Files, and Archive Log Files are stored on shared storage.

- ASM File System is used for shared storage.

- Control File Autobackups are enabled.

- Recovery Catalog Database is the highly recommended location for the RMAN Repository.

- Single or Multiple RAC databases within the same RAC environment are supported.

Figure 6 and Figure 7 provide an overview of supported single and multiple RAC database deployments.

Figure 6. Supported single-database deployments
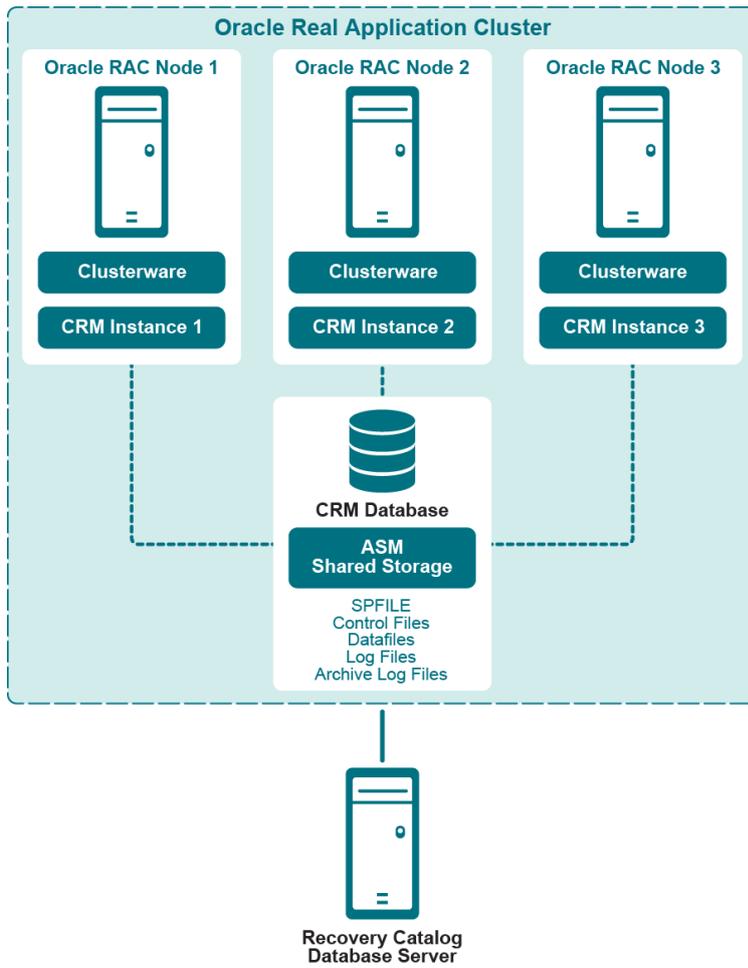
Figure 7. Supported multiple-database deployments
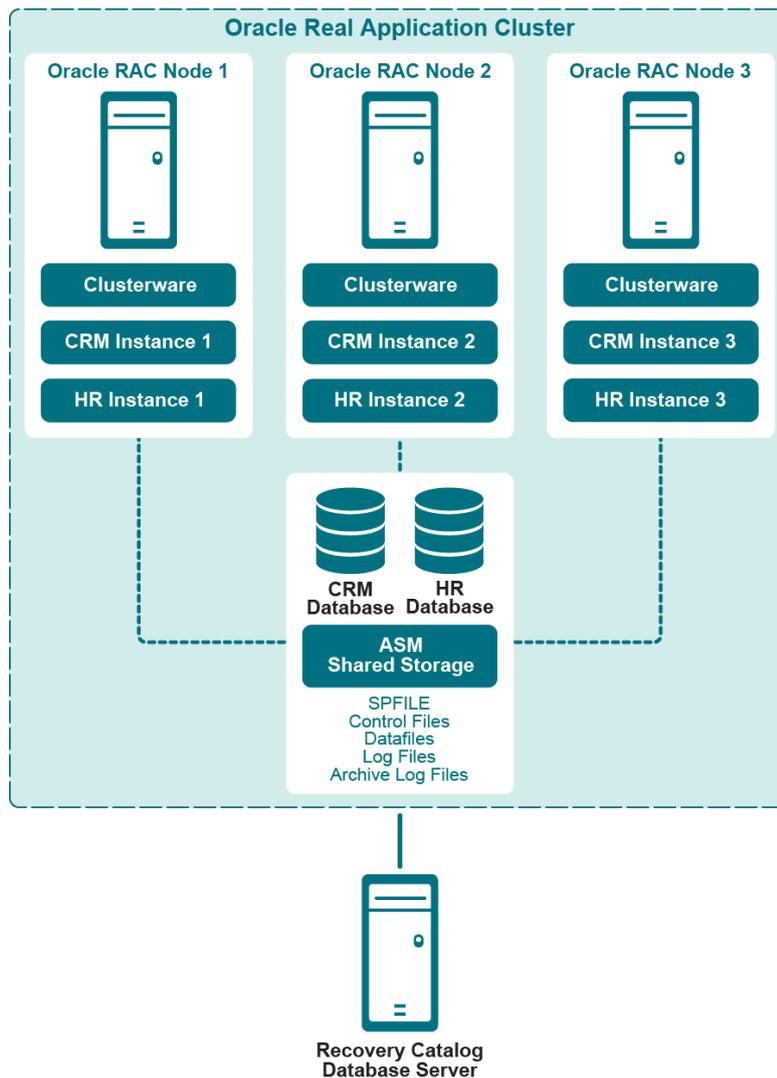
Installation prerequisites

The following prerequisites must be met before you can install Plug-in *for Oracle* in a RAC environment:

- **Oracle RAC environment in place**: You must have a properly configured Oracle RAC environment that matches the configuration as specified in RAC configuration requirements.

- **Separate NetVault Server machine**: The machine that is to serve as the NetVault Server must be properly configured. This machine must exist outside the Oracle RAC environment, but have network connectivity to the nodes and hosts within the RAC environment.

- **Separate Recovery Catalog Server**: Quest recommends that you store the RMAN repository in a Recovery Catalog database on a server outside the Oracle RAC environment. For more information, see Creating the RMAN repository.

- **Additional prerequisites**: In addition to the preceding RAC-specific requirements, meet the requirements that are specified in Prerequisites.

- **Oracle 11g Release 2 and later:** If you are using Oracle 11g Release 2 or later, ensure that the snapshot Control File is shared by all nodes in the RAC database. The snapshot Control File must be saved in a shared location, such as in one ASM instance or on a shared drive. The location of the snapshot Control

File is managed by the snapshot Control File name RMAN parameter. In Oracle 11g Release 2, by default, this parameter refers to a persistent file in the local file system on each of the RAC nodes. If this configuration applies to your environment, ensure that you change the parameter to point to a shared location. Otherwise, RMAN backups that include the Control File might fail with an "ORA-00245: control file backup operation failed" message.

To check the value of the parameter and change it if necessary, perform the following steps:

1 Review the current setting:

```
SHOW ALL;
…
CONFIGURE SNAPSHOT CONTROLFILE NAME TO '/u01/app/oracle/product/11.2.0.2/
   salesdb/dbs/snapcfsalesdb.ora'; # default
…
```

2 If the parameter is set to a persistent local file and you want to change it to a shared location, change it using something similar to the following—the following example assumes that ASM is used and that +FRA is the name of one ASM instance:

```
RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO
   '+FRA/salesdb/snapcfsalesdb.ora';
```

Enabling Control File Autobackups in a RAC environment

Control File Autobackups are required in a RAC environment. For more information, see Reviewing the Control File backup types. For more information on single-instance RAC databases, see Enabling Control File Autobackups.

You can turn the Autobackup feature on for multi-instance RAC databases by running the following commands from the RMAN command-line utility on the node where the plug-in will be installed:

1 After making sure that the Recovery Catalog database is open, connect RMAN to the target database and Recovery Catalog database.

For example, issue the following commands to connect to the catalog database:

```
> rman TARGET / CATALOG rman/<password>@<connect_identifier for catalog>
```

2 Configure the default device type to "SBT_TAPE".

```
CONFIGURE DEFAULT DEVICE TYPE TO 'SBT_TAPE';
```

3 Configure parallelism for the default device type.

```
CONFIGURE DEVICE TYPE 'SBT_TAPE' PARALLELISM 1;
```

4 Assign the channel for Control File Autobackups.

```
CONFIGURE CHANNEL 1 DEVICE TYPE 'SBT_TAPE' CONNECT
   'SYS/<password>@<connect_identifier for local instance>';
```

5 Enable Control File Autobackups.

```
CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

6 Confirm that Control File Autobackups are enabled by reviewing the RMAN configuration settings:

```
SHOW ALL;
```

> **i** | **IMPORTANT:** If the plug-in is relocated to a different node or the SYS password is altered, run this process again from the node where the plug-in is installed. Running the process again ensures that the Control File Autobackups are successfully completed.

Installing or upgrading the plug-in

Installing Plug-in *for Oracle* in a multi-instance database environment is almost identical to installing it in a single-instance database environment. The plug-in is installed on one of the nodes that hosts one of the instances for the RAC-enabled database. All RMAN-based backups and restores are performed from this single installation of the plug-in. Installing the plug-in is a two-step process:

1 **Install the NetVault Client software.**

   During installation of the NetVault Client software on the selected node of the RAC environment, a name that represents the entire RAC environment should be specified as the **NetVault Machine Name** versus the machine name of the selected node. For example, if the machine name for the node where the NetVault Client software is being installed is called RAC_NODE_1, RAC_NODE_1 should *not* be specified as the NetVault Machine Name; it should be a generic or virtual name that represents the entire RAC environment such as PROD_RAC.

2 **Install Plug-in *for Oracle*.**

   After the NetVault Client software has been installed on the selected node of the RAC environment, install the plug-in. The process is identical to installing the plug-in for a single-instance database environment except that plug-in is installed on the NetVault Client that corresponds with the generic or virtual name provided as the NetVault Machine Name during the NetVault Client installation.

Configuring the plug-in

Configuring Plug-in *for Oracle* in a multi-instance RAC environment is almost identical to configuring the plug-in in a single-instance, non-RAC environment except for the items identified in the following topics.

- Configuring default attributes
- Determining the local Oracle SID and Home directory in a RAC environment
- Adding a database to the plug-in

Configuring default attributes

- **NetVault** (required for RMAN backups in a RAC environment): Specify the name of the NetVault Server where the Oracle Database Server was added as a NetVault Client.

- **Do Restores from NetVault Server** (required for RMAN backups in a RAC environment): Specify the name of the NetVault Server where the Oracle Database Server was added as a NetVault Client.

- **CLI Backup/Autobackup Target Set** (required for RMAN backups in a RAC environment): Specify the name of the Backup Target Set that was specified in the Configuring a default Target Set for RMAN backups.

Determining the local Oracle SID and Home directory in a RAC environment

In a multi-instance RAC environment, the local SID and Oracle Home can be determined with the **srvctl config database** commands.

1 From a terminal window, connect as the Oracle Software Owner to the node where plug-in is installed.

2 Run the following command:

```
srvctl config database
```

A list of all databases in the RAC environment is displayed.

3   Run the following command, where *<database_name>* represents the database that is being added to the plug-in:

```
srvctl config database -d <database_name>
```

The name of the node, local Oracle SID, and local Oracle Home are displayed for each node in the RAC environment in the format:

```
<node> <local Oracle SID> <local Oracle Home>
```

In the following example, Plug-in *for Oracle* is being installed on the **"faro"** node, the local Oracle SID is **"o10rac3,"** and the local Oracle Home is **"/home/o10g/db."**

Example

```
faro:/home/o10g => srvctl config database
o10grac
o10rac
faro:/home/o10g => srvctl config database -d o10rac
lisbon o10rac1 /home/o10g/db
evora o10rac2 /home/o10g/db
faro o10rac3 /home/o10g/db
faro:/home/o10g =>
```

Adding a database to the plug-in

Adding a multi-instance database in Plug-in *for Oracle* is almost identical to adding a single-instance, non-RAC database except for the following differences:

*   **Oracle SID**: Enter the local SID for the target RAC database. The local SID is the instance name for the target RAC database on the node where the plug-in is installed.

*   **Parameter File(s) Path**: In a multi-instance RAC environment, the plug-in requires the usage of an SPFILE as a dynamic method of maintaining initialization parameters. The SPFILE must be stored on the shared storage in ASM. Because the SPFILE is included in the Control File Autobackups, this field should be left blank in a RAC environment.

*   **Control File Autobackup enabled**: Select this option because Control File Autobackups are required in a RAC environment.

For more information, see Configuring the plug-in.

Backing up data

RMAN-based backups are the only supported backup method for multi-instance RAC databases. RMAN is RAC-aware in that it automatically identifies that a database is a multi-instance RAC database and backs up the selected data accordingly.

With the plug-in installed on one of the nodes that hosts one of the instances for the RAC-enabled database, all RMAN-based backups and restores are performed from this single installation of the plug-in.

The Parameter File node is not selectable on the **NetVault Selection** page.

For more information on selecting data for backup, setting backup options, and finalizing and submitting the job, see Performing RMAN backups.

Performing backups after node failure

If the node where the plug-in is installed in the RAC environment experiences a node failure and the node is, or will be, unavailable at the time of the next scheduled backup, relocate the plug-in to an alternate node in the RAC environment.

1   **Install the NetVault Client software on an alternate node.**

    Install the NetVault Client on an alternate node in the RAC environment. During the installation process, be sure to use the same **NetVault Machine Name** that was used during the installation of the NetVault Client on the original node. For example, if PROD_RAC was the NetVault Machine Name specified during the installation on the original node, PROD_RAC should be used as the NetVault Machine Name on the alternate node.

2   **Install Plug-in** *for Oracle***.**

    After the NetVault Client has been installed on the selected node of the RAC environment, install the plug-in. The process is identical to installing the plug-in for a single-instance, non-RAC environment except that plug-in is installed on the NetVault Client that corresponds with the generic or virtual name provided as the NetVault Machine Name during the NetVault Client installation.

    > **i** | **IMPORTANT:** If the relocation of the plug-in is expected to be permanent or last longer than 45 days, visit https://support.quest.com/contact-us/licensing and request relocation of the **RAC Database-Based** Plug-in *for Oracle* license files.

3   **With the plug-in installed on the alternate node, update the Oracle SID field on the Edit Oracle Database dialog box.**

    Enter the local SID for the target RAC database. The local SID is the instance name for the target RAC database on the node where the plug-in has been relocated. For more information, see Editing database information.

For more information on selecting data for backup, setting backup options, and finalizing and submitting the job, see Performing RMAN backups.

Restoring data in a RAC environment

Because RMAN is RAC-aware, it restores the selected data accordingly. With Plug-in *for Oracle* installed on one of the nodes that hosts one of the instances for the RAC-enabled database, all RMAN-based restores are performed from this single installation of the plug-in. The restore process is similar to restoring single-instance, non-RAC environments, with the following exceptions:

*   During the Selecting data for a restore phase of performing RMAN restores, you might select a backup set from a node where the **Oracle SID** equals the Oracle SID of the instance on which the plug-in is running. But you might also select a backup set from an Oracle SID of a different RAC node for which backups were performed previously. For more information about performing restores after a node failure, see Performing restores after node failure.

*   During the Setting restore options phase, if authentication for the target database is needed, set the Target Database Authentication Details to the applicable values for the local Oracle instance on which the plug-in that is performing the restore resides. For more information, see Setting Destination Details.

For more information on preparing the database for recovery, selecting data items for a restore, setting restore options, finalizing and submitting the job, and opening the database for use, see Performing RMAN restores.

> **i** | **IMPORTANT:** If the plug-in was relocated to an alternate node and backups were performed from multiple instances, use the current Oracle SID when selecting data on the **Create Selection Set** page.

The following topics outline additional types of RMAN recovery that are available in a RAC environment.

*   Recovering a Whole Database to the same RAC environment

*   Restoring SPFILE in a RAC environment

Recovering a Whole Database to the same RAC environment

This type of recovery is performed when the current Control File and SPFILE are intact but all the datafiles are damaged or lost. You can perform a Complete Recovery or recover the database to a specific point. Following is an overview of the process to recover the whole database in a RAC environment.

1 From a terminal window, connect as the Oracle Software Owner to the node where Plug-in *for Oracle* is installed.

2 Set the **ORACLE_SID** environment variable to the local instance.

```
export ORACLE_SID = <local_SID>
```

3 Start **SQL*Plus**, and connect with administrator privileges to the local instance.

```
SQLPLUS SYS AS SYSDBA
```

4 Shut down the local instance.

```
SHUTDOWN ABORT;
```

5 Start the local instance with the **NOMOUNT** option.

```
STARTUP FORCE NOMOUNT;
```

6 Disable the **CLUSTER_DATABASE** initialization parameter for the local instance.

```
ALTER SYSTEM SET CLUSTER_DATABASE=FALSE SCOPE=SPFILE SID='<local_SID>';
```

7 Shut down the local instance.

```
SHUTDOWN ABORT;
```

8 Exit **SQL*Plus**.

9 Shut down all RAC instances.

```
> srvctl STOP DATABASE -d <database_name> -o abort
```

10 Ensure that all RAC instances are shut down.

```
>srvctl STATUS DATABASE -d <database_name>
```

11 Start **SQL*Plus**, and connect with administrator privileges to the local instance.

```
SQLPLUS SYS AS SYSDBA
```

12 In **SQL*Plus**, start the local instance with the **MOUNT** option.

```
STARTUP MOUNT;
```

13 On the **Create Restore Job—Choose Saveset** page, find the latest available backup that included *all* the tablespaces, and click **Restore**.

14 On the **Create Selection Set** page, select the **Whole Database** node, and click 📝 .

15 Click the **Restore Options** tab, and select the applicable options:

- To include read-only datafiles in the restore and recovery process, select **Check Read-Only Datafiles and Restore if Necessary** and **Check Read-Only Datafiles and Recover if Not**

Current. The read-only files are included only if they are needed to complete recovery. If they are not damaged, RMAN omits them.

- ▪ To delete the restored archive logs that Oracle deems unnecessary, select the **Delete Archive Logs No Longer Needed** option.

16 To perform recovery on the database, that is, to reconstruct the database up to the current or specified point, select one of the following options on the **Perform Recovery** tab:

- ▪ **Perform Complete Recovery**: Recovers the database to the current time, that is, to the last committed transaction available in the archived and online redo logs.

- ▪ **Perform Database Point-in-Time Recovery**: Recovers the database to a specific point whether it be a specific time, log sequence number, or SCN.

17 On the **Recovery Catalog** tab, select the **Use Catalog** option, and specify the Recovery Catalog connection details.

18 Complete the **Target Client**, **Schedule**, and **Advanced Options** tabs, and submit the job.

RMAN automatically chooses the best available full, incremental, or Archived Log backups, and restores and recovers the database.

19 From a terminal window, connect as the Oracle Software Owner to the node where Plug-in *for Oracle* is installed.

20 Start **SQL*Plus**, and connect with administrator privileges to the local instance.

```
SQLPLUS SYS AS SYSDBA
```

21 Enable the **CLUSTER_DATABASE** initialization parameter for the local instance.

```
ALTER SYSTEM SET CLUSTER_DATABASE=TRUE SCOPE=SPFILE SID='<local_SID>';
```

22 Shut down the local instance.

```
SHUTDOWN ABORT;
```

23 Exit **SQL*Plus**.

24 Start all RAC instances.

```
>srvctl START DATABASE -d <database_name>
```

25 Start all RAC Services.

```
>srvctl START SERVICE -d <database_name>
```

26 Ensure that all RAC instances are running.

```
>srvctl STATUS DATABASE -d <database_name>
```

> **ℹ** | **IMPORTANT:** After completing a restore of the Whole Database, Quest recommends that you perform a Full or Incremental Level 0 backup with the entire database selected on the **NetVault Selections** page.

Restoring SPFILE in a RAC environment

Because Control File Autobackups are a prerequisite in a multi-instance RAC environment, the SPFILE is backed up with the Control File whenever an Autobackup is performed. The SPFILE is restored from the Autobackup to the default location; for more information, see the following steps. If the instance is already started with the server parameter file, you cannot overwrite the existing server parameter file. For more information on restoring an SPFILE to a non-default location, see the *Oracle Database Backup and Recovery Basics* guide.

1 From a terminal window, connect as the Oracle Software Owner to the node where Plug-in *for Oracle* is installed.

2 Set the **ORACLE_SID** environment variable to the local instance.

```
export ORACLE_SID = <local_SID>
```

3   Start **SQL*Plus**, and connect with administrator privileges to the local instance.

```
SQLPLUS SYS AS SYSDBA
```

4   Shut down the local instance.

```
SHUTDOWN ABORT;
```

5   Start the local instance with the **NOMOUNT** option.

```
STARTUP FORCE NOMOUNT;
```

6   Disable the **CLUSTER_DATABASE** initialization parameter for the local instance.

```
ALTER SYSTEM SET CLUSTER_DATABASE=FALSE SCOPE=SPFILE SID='<local_SID>';
```

7   Shut down the local instance.

```
SHUTDOWN ABORT;
```

8   Exit **SQL*Plus**.

9   Shut down all RAC instances.

```
>srvctl STOP DATABASE -d <database_name> -o abort
```

10 Ensure that all RAC instances are shut down.

```
>srvctl STATUS DATABASE -d <database_name>
```

11 Start RMAN, and connect to the target database and Recovery Catalog.

```
>rman TARGET / CATALOG rman/<password>@<connect_identifier for catalog>
```

12 Start the local instance with the **FORCE NOMOUNT** option.

```
STARTUP FORCE NOMOUNT;
```

13 Set the database identifier for the target database with **SET DBID**.

RMAN displays the DBID whenever you connect to the target. You can also obtain it by inspecting saved RMAN log files, querying the Catalog, or looking at the filenames of Control File Autobackup. The DBID is also available on the **Details** dialog box or the **Backup Contents** tab.

```
SET DBID <DBID>;
```

14 Restore the SPFILE Autobackup.

If you are restoring the SPFILE to the default location, run:

```
RUN {
ALLOCATE CHANNEL CH1 TYPE 'SBT_TAPE';
RESTORE SPFILE FROM AUTOBACKUP;
}
```

15 Start the instance with the FORCE option.

```
STARTUP FORCE;
```

16 Exit RMAN.

17 Start **SQL*Plus**, and connect with administrator privileges to the local instance.

```
SQLPLUS SYS AS SYSDBA
```

18 Enable the **CLUSTER_DATABASE** initialization parameter for the local instance.

```
ALTER SYSTEM SET CLUSTER_DATABASE=TRUE SCOPE=SPFILE SID='<local_SID>';
```

19 Shut down the local instance.

```
SHUTDOWN ABORT;
```

20 Exit **SQL*Plus**.

21 Start all RAC instances.

```
>srvctl START DATABASE -d <database_name>
```

22 Start all RAC Services.

```
>srvctl START SERVICE -d <database_name>
```

23 Ensure that all RAC instances are running.

```
>srvctl STATUS DATABASE -d <database_name>
```

Restoring Control Files in a RAC environment

Restoring the Control File should only be performed in the following cases:

- All current copies of the Control File have been lost due to a media failure or accidental user deletion, or have become corrupted and are not usable.

- Oracle Customer Support has advised that a Control File restore is required.

Because Control File Autobackups are a prerequisite in a multi-instance RAC environment, perform the following steps to restore the Control Files from the Autobackup.

1 From a terminal window, connect as the Oracle Software Owner to the node where Plug-in *for Oracle* is installed.

2 Set the **ORACLE_SID** environment variable to the local instance.

```
export ORACLE_SID = <local_SID>
```

3 Start **SQL*Plus**, and connect with administrator privileges to the local instance.

```
SQLPLUS SYS AS SYSDBA
```

4 Shut down the local instance.

```
SHUTDOWN ABORT;
```

5 Start the local instance with the **NOMOUNT** option.

```
STARTUP FORCE NOMOUNT;
```

6 Disable the **CLUSTER_DATABASE** initialization parameter for the local instance.

```
ALTER SYSTEM SET CLUSTER_DATABASE=FALSE SCOPE=SPFILE SID='<local_SID>';
```

7 Shut down the local instance.

```
SHUTDOWN ABORT;
```

8 Exit **SQL*Plus**.

9 Shut down all RAC instances.

```
>srvctl STOP DATABASE -d <database_name> -o abort
```

10 Ensure that all RAC instances are shut down.

```
>srvctl STATUS DATABASE -d <database_name>
```

11 Start RMAN, and connect to the target database.

```
>rman TARGET /
```

12 Start the local instance with the **FORCE NOMOUNT** option.

```
STARTUP FORCE NOMOUNT;
```

13 Set the database identifier for the target database with **SET DBID**.

RMAN displays the DBID whenever you connect to the target. You can also obtain it by inspecting saved RMAN log files, querying the Catalog, or looking at the filenames of Control File Autobackup. The DBID is also available on the **Details** dialog box or the **Backup Contents** tab.

```
SET DBID <DBID>;
```

14 Restore the Control File Autobackup.

If you are restoring the Control File from the last Autobackup, run:

```
RUN {
ALLOCATE CHANNEL CH1 TYPE 'SBT_TAPE';
RESTORE CONTROLFILE FROM AUTOBACKUP;
ALTER DATABASE MOUNT;
RESTORE DATABASE;
RECOVER DATABASE;
}
```

15 Open the database with the **RESETLOGS** option.

```
ALTER DATABASE OPEN RESETLOGS;
```

16 Exit RMAN.

17 Start **SQL*Plus**, and connect with administrator privileges to the local instance.

```
SQLPLUS SYS AS SYSDBA
```

18 Enable the **CLUSTER_DATABASE** initialization parameter for the local instance.

```
ALTER SYSTEM SET CLUSTER_DATABASE=TRUE SCOPE=SPFILE SID='<local_SID>';
```

19 Shut down the local instance.

```
SHUTDOWN ABORT;
```

20 Exit **SQL*Plus**.

21 Start all RAC instances.

```
>srvctl START DATABASE -d <database_name>
```

22 Start all RAC Services.

```
>srvctl START SERVICE -d <database_name>
```

23 Ensure that all RAC instances are running.

```
>srvctl STATUS DATABASE -d <database_name>
```

> **i** | **IMPORTANT:** After restoring a Control File from an Autobackup, Quest recommends that you perform a Full or Incremental Level 0 backup with the entire database selected on the **NetVault Selections** page.

Performing disaster recovery in a RAC environment

Disaster recovery includes the restore and recovery of a multi-instance RAC database after a complete media failure of all the shared storage including the current Control Files, all online redo log files, all parameter files, and the Recovery Catalog. To perform a disaster recovery, you need the minimum required set of backups; the minimum includes the backups of all the datafiles, some archived redo logs generated after the time of the backup, and at least one Autobackup of the Control File.

If your goal is to create a copy of your target database for ongoing use on a standby RAC environment or independent RAC environment, do *not* use these procedures. Instead, use the **Duplicate Database** backup type, which is designed for this purpose. For more information, see Duplicating a database in a RAC environment.

i | **IMPORTANT:** When performing a test recovery, ensure that you do not connect to the Recovery Catalog during datafile restoration. Otherwise, RMAN records information about the restored datafiles to the Recovery Catalog. This recording interferes with future attempts to restore and recover the primary database. If you must use a Recovery Catalog because the Control File is not large enough to contain the RMAN repository data on all the backups that you need to restore, export the catalog and import it into a different schema or database, and use the copied Recovery Catalog for the test restore. Otherwise, the catalog considers the restored database as the current target database.

The following procedure details the steps to perform disaster recovery to the same RAC environment where the source database resided. This procedure would be used in the scenario in which a complete media failure of the shared storage occurred and the damaged shared storage was rebuilt using the same configuration or new shared storage was configured imitating the original shared storage's configuration.

For variations on the following disaster recovery procedure, see *Performing Disaster Recovery* in the *Oracle Database Backup and Recovery Advanced User's Guide*.

- Prerequisites
- Performing the disaster recovery

Prerequisites

To perform a disaster recovery without a Recovery Catalog, the following requirements must be met:

- **Oracle RAC environment in place**: You must have a properly configured Oracle RAC environment that matches the configuration as specified Performing User Managed backups. In addition, the same version of the Oracle database software must be installed in the RAC environment as it existed on the damaged RAC environment. The ORACLE_HOME should be identical on both the original and new node. You do not have to create the database, but you do need to configure Oracle Net Services and ASM for shared storage.

- **NetVault software and the Plug-in *for Oracle* installed**: The same version of NetVault software and the plug-in must be installed on one of the nodes in the RAC using the same **NetVault Machine Name** as previously defined for the RAC environment. Ensure that the plug-in default attributes are defined as specified in Configuring default attributes.

- **DBID of the source database identified**: Identify the DBID of the damaged database. If the DBID was not recorded as part of the initial plug-in installation, you can obtain the DBID by examining the NetVault binary logs for RMAN-based backups of the source database. The DBID is also available on the **Details** dialog box or the **Backup Contents** tab.

- **Default Location for SPFILE must exist**: The default location in the ASM shared storage must exist before performing the following steps. Failure to ensure that the default location exists results in an **ORA-17502 Failed to create file** error during the SPFILE restore. Additionally, the SPFILE location should be detailed in the PFILE with the SPFILE parameter.

- **Backups of all tablespaces made available**: Verify that you have the latest backups of all the tablespaces.

Performing the disaster recovery

1 Start RMAN, and connect to the target database.

```
> rman TARGET /
```

2 Set the database identifier for the target database with **SET DBID**.

RMAN displays the DBID whenever you connect to the target. You can also obtain it by inspecting saved RMAN log files, querying the Catalog, or looking at the filenames of Control File Autobackup. The DBID is also available on the **Details** dialog box or the **Backup Contents** tab.

```
SET DBID <DBID>;
```

3 Start the local instance with the **FORCE NOMOUNT** option.

```
STARTUP FORCE NOMOUNT;
```

4 Restore the SPFILE Autobackup.

If you are restoring the SPFILE to the default location, run:

```
RUN {
ALLOCATE CHANNEL CH1 TYPE 'SBT_TAPE';
RESTORE SPFILE FROM AUTOBACKUP;
}
```

5 Start the local instance with the **FORCE NOMOUNT** option.

```
STARTUP FORCE NOMOUNT;
```

6 Restore the Control File Autobackup.

If you are restoring the Control File from the last Autobackup, run:

```
RUN {
ALLOCATE CHANNEL CH1 TYPE 'SBT_TAPE';
RESTORE CONTROLFILE FROM AUTOBACKUP;
}
```

7 Mount the database:

```
ALTER DATABASE MOUNT;
```

8 Disable Block Change Tracking.

```
SQL "ALTER DATABASE DISABLE BLOCK CHANGE TRACKING";
```

9 On the **Create Restore Job—Choose Saveset** page, find the latest available backup that included *all* the tablespaces, and click **Restore**.

10 On the **Create Selection Set** page, navigate to the source NetVault Client and database, select the **Whole Database** node, and click 📝 .

11 Click the **Restore Options** tab, and select the **Check Read-Only Datafiles and Restore if Necessary** and **Check Read-Only Datafiles and Recover if Not Current** options to include read-only datafiles in the restore and recovery process.

12 Because recovery must be performed in a disaster recovery scenario, on the **Perform Recovery** tab, select **Perform Database Point-in-Time Recovery**.

PIT Recovery recovers the database to a specific time, log sequence number, or SCN. If the specific time or log sequence of the failure is not known, Quest recommends that you use the SCN option.

13 On the **Recovery Catalog** tab, clear the **Use Catalog** option.

14 Complete the **Target Client**, **Schedule**, and **Advanced Options** tabs, and submit the job.

RMAN automatically chooses the best available full, incremental, or Archived Log backups and restores and recovers the database.

15 In RMAN, enable Block Change Tracking.

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

16 Update the RMAN metadata.

```
CROSSCHECK ARCHIVELOG ALL;
```

17 Exit RMAN.

18 Start all RAC instances.

```
>srvctl START DATABASE -d <database_name>
```

19 Start all RAC Services.

```
>srvctl START SERVICE -d <database_name>
```

20 Ensure that all RAC instances are running.

```
>srvctl STATUS DATABASE -d <database_name>
```

> **i** | **IMPORTANT:** After completing a disaster recovery, Quest recommends that you perform a Full or
> Incremental Level 0 backup with the entire database selected on the **NetVault Selections** page.

Performing restores after node failure

If the multi-instance RAC database experiences a media failure or data corruption simultaneously with a failure of
the node where the plug-in is installed, relocate the plug-in to an alternate node in the RAC environment to perform
an RMAN-based restore and recovery. Relocating the plug-in is a two-step process:

1   **Install the NetVault Client software on an alternate node.**

     Install the NetVault Client on an alternate node in the RAC environment.

2   **Install Plug-in** *for Oracle*.

     After the NetVault Client has been installed on the selected node of the RAC environment, install the plug-
     in. The process is identical to installing the plug-in for a single-instance database environment except that
     you install the plug-in on the NetVault Client that corresponds with the generic or virtual name provided as
     the NetVault Machine Name during the NetVault Client installation.

     > **i** | **IMPORTANT:** If the relocation of the plug-in is expected to be permanent or last longer than 45 days,
     > visit https://support.quest.com/contact-us/licensing to request relocation of the **RAC Database-
     > Based** Plug-in *for Oracle* license files.

With Plug-in *for Oracle* installed on the alternate node, the restore process is similar to performing restores in a
single-instance, non-RAC environment except for the following differences:

*   During the Selecting data for a restore phase of performing RMAN restores, select the backup set from the
     node on the **Create Selection Set** page where the **Oracle SID** equals the Oracle SID of the instance
     where the plug-in was originally installed before the relocation.

*   During the Setting restore options phase, set the Target Database Authentication Details to the appropriate
     values for the local instance on the alternate node where the plug-in now resides. For more information,
     see Setting Destination Details.

For more information on preparing the database for recovery, selecting data items for a restore, setting restore
options, finalizing and submitting the job, and opening the database for use, see Performing RMAN restores.

Duplicating a database in a RAC environment

A Duplicate Database backup lets you create a duplicate or clone of a source database for special purposes that
must not affect the overall backup and restore procedures for the source database. In a RAC environment, Oracle
does not support duplicating a multi-instance database to a different multi-instance database. However, Oracle
does support duplicating a multi-instance database to a single-instance database followed by converting the
single-instance database to a multi-instance database. For more information, see the *Oracle Real Application
Clusters Installation Guide*.

Restoring a Duplicate Database backup to the same RAC Environment is almost identical to restoring a Duplicate
Database backup in a single-instance environment. The differences are described in the following topics.

Prerequisites

PFILE updated with auxiliary values: In addition to the updates to the PFILE as specified in the Non-RAC
duplication procedures, the following additional updates must be made when the source database is a multi-
instance database:

1   Comment out the RAC-related parameters by inserting a "#" (number sign) at the start of the entry.

     Sample RAC-related parameters include:

- **.cluster_database_instances = N**
- **.cluster_database=TRUE**
- ***<database/instance_name>*.cluster_database=TRUE**
- ***<instance_name>*.instance_number=N**
- ***<instance_name>*.thread = N**
- ***<instance_name>*.undo_tablespace = '*<tablespace_name>*'**

2   Add the following parameters:

***<auxiliary_instance_name>*.undo_tablespace='*<undo_tablespace>*'**

***<auxiliary_instance_name>*.thread = 1**

Post-restore procedure

After completion of the Restore Duplicate Database Procedure as defined in the Duplicating a database in a non-RAC environment, you can convert the resulting single-instance database to a multi-instance database. To do so, use one of the procedures detailed in *Converting to Oracle Real application Clusters from Single-Instance Oracle Databases* in the *Oracle Real Application Clusters Installation Guide*.

Using the plug-in in a failover cluster environment

- Oracle Server Fail Safe Failover Clustering: an overview
- Installing the plug-in
- Licensing the plug-in
- Configuring the plug-in and adding a database
- Backing up data
- Restoring data

Oracle Server Fail Safe Failover Clustering: an overview

Oracle **Fail Safe Failover Clustering** (Active/Passive) is designed to provide high-availability for an entire Oracle Server instance. For example, you can configure an Oracle Server instance on one node of a failover cluster to fail over to any other node in the cluster during a hardware failure, OS failure, or a planned upgrade.

A failover cluster is a combination of one or more nodes (hosts) with one or more shared disks. Various resources hosted by the nodes, such as IP, shared storage, and an application—Oracle in this case—can be grouped to create a **Clustered Service**. A Virtual Service appears on the network as if it were a single computer running an application, but provides failover from one node to a different node if the current node becomes unavailable.

> **i** | **IMPORTANT:** In NetVault terminology, a clustered service is accessed by a **Virtual Client**. The references to **Virtual Client** in Plug-in *for Oracle* are basically references to the **Clustered Service** in the Oracle Server Failover Cluster environment.

Plug-in *for Oracle* provides support for Oracle Server Failover Clustering. Using the failover cluster network name, the plug-in is able to identify the current node that is in control of the Oracle Server Clustered Service and target it for backup.

This topic points out differences between the setup and usage of the plug-in in a Failover Cluster environment vs. a traditional one. It mirrors the topics found in the standard instructions, as follows:

- Installing the plug-in
- Licensing the plug-in
- Configuring the plug-in and adding a database
- Backing up data
- Restoring data

Important considerations

- Unless outlined in the topics that follow, backups and restores performed with the plug-in of clustered data are the same as those procedures performed with traditional Oracle Server data.

- The following topics only offer information on Oracle-specific settings required for the use of this plug-in in a Failover Cluster environment. They do not offer instructions on how to set up NetVault's **Application Cluster Support** to administer backups and restores of non-Oracle Server-related data and files. This process is not plug-in-specific, and you can find complete details in the *Quest NetVault Administrator's Guide*.

- Before you continue, review all cluster-related information provided in the *Quest NetVault Administrator's Guide* to understand how the following information works with Oracle Server Failover Cluster functionality.

Installing the plug-in

Use the following topics to install the plug-in:

- Installation prerequisites

- Installing or upgrading the software

Installation prerequisites

The following prerequisites must be met before you install Plug-in *for Oracle* in a clustered environment:

- **Oracle Failover clustering environment in place**: You must have a properly configured Oracle Cluster environment.

  > **i** | **IMPORTANT:** Support for this feature was tested on Windows Server using the Oracle Fail Safe Clustering feature and employing a two-node Oracle (v6.5) cluster configuration with shared storage containing the database data files and logs. If you intend to use clustering in a different configuration, test backups and restores before deploying it in a production environment.

- **Separate NetVault Server machine**: The machine that is to serve as the NetVault Server must be properly configured. This machine *must exist outside* the Oracle Server cluster, but have network connectivity to the nodes (hosts) within the cluster.

Installing or upgrading the software

Installation of the plug-in for a clustered environment is different than the traditional installation of this plug-in. This process is completed through the creation of a **Virtual Client** on the NetVault Server. A Virtual Client is a group of nodes within the cluster that are seen by the NetVault Server as a *single* client that is created to back up a single clustered service.

Creating a Virtual Client

As noted earlier, the Virtual Client creation process is not plug-in-specific, and you can find complete details in the *Quest NetVault Administrator's Guide*. However, consider the following points during the Virtual Client creation process:

- **Assign a name to the Virtual Client**: Quest recommends that you use the virtual-network name assigned to the Oracle database as the NetVault Virtual Client name; you can also use a name that is easily associated with the Oracle database or cluster environment. This configuration makes it easier to recognize the Oracle database for which the NetVault Virtual Client was created.

- **Only include relevant cluster nodes in the Virtual Client**: The hosts that are to be included in the creation of a Virtual Client should *only* be those nodes within the cluster that are relevant to the Oracle Database Server that is to be backed up and restored.

After the creation of the Virtual Client, the plug-in is transferred to all designated cluster nodes and installed locally. You can use the installed Plug-in *for Oracle* by using the Virtual Client to back up and restore shared data; you can *only* perform backups and restores of data established as shared within the cluster.

Licensing the plug-in

Another difference between using Plug-in *for Oracle* in a clustered environment is how it is licensed for use. The plug-in supports backup and restore of shared data *only*. Hence, an Oracle Database Server Failover Cluster environment requires a clustered application NetVault license for the NetVault Oracle Virtual Client. A NetVault clustered license for NetVault Oracle Virtual Client lets you perform NetVault WebUI-based backups.

To allow RMAN autobackups and RMAN CLI backups, install additional NetVault Oracle non-cluster standalone licenses on the NetVault Server for each NetVault Client that comprises the NetVault Oracle Virtual Client. For example, if you are running a Failover Cluster environment with NetVault Oracle Clients, NetVault Client A and NetVault Client B, obtain NetVault Oracle licenses for NetVault Client A and NetVault Client B, and install the licenses on the NetVault Server, in addition to the clustered application license for the NetVault Oracle Virtual Client.

> **i** | **NOTE:** When creating a NetVault Oracle Virtual Client for the first time, NetVault installer installs a NetVault Oracle Cluster trial license on the NetVault Server that lets you run NetVault WebUI backups for the NetVault Oracle clustered application.
>
> To allow RMAN autobackups and RMAN CLI backups, request and install one NetVault Oracle standalone trial license for each NetVault Client that comprises the NetVault Oracle Virtual Client. Alternatively, install Plug-in *for Oracle* on each NetVault Client. Be sure to request and install additional licenses before the trial licenses expire.

For information on the licensing process, including how to obtain the proper license keys, see the *Quest NetVault Installation Guide*.

Configuring the plug-in and adding a database

Configuring Plug-in *for Oracle* for a clustered failover environment is almost identical to configuring the plug-in for a single-instance, non-clustered environment. Adding a database involves the following exceptions:

- **Oracle SID**: Enter the local SID to the active node in the cluster for the target Oracle database. The local SID is the Oracle instance name on the local node for the target database.

- **Control File Autobackup enabled**: Quest recommends that you select this option.

- **PFILE usage**: If the local instances in the cluster nodes of the targeted Oracle database use PFILE instead of SPFILE, you might want to clear the **Use RMAN Commands to Backup SPFILE** option on the **RMAN Details** tab. Access this tab through **the Add Database** or **Edit Database** options, and enter the applicable information in the **Parameter File(s) Path** box on the **Oracle Instance Details** tab.

In a clustered failover environment to allow NetVault Plug-in *for Oracle* to perform RMAN autobackups and RMAN CLI backups, the plug-in installed in each NetVault Oracle Client comprising the NetVault Oracle Virtual Client must be configured with the name of the NetVault Server. For information about how to enter the NetVault Server name in the inputs **NetVault Server** and **Do Restore from NetVault Server** in each NetVault Oracle Client that is part of the NetVault Oracle Virtual Client, see Configuring default settings.

For more information, see Configuring the plug-in and Adding a database.

i | **IMPORTANT:** If a failover to a different node occurs, use the **Add Database** option to add the Oracle database information to the plug-in on the active node—the one to which the cluster failed over. You only need to enter this information once; in subsequent failovers, the plug-in automatically retrieves the information.

Backing up data

Backing up data using Plug-in *for Oracle* that has been set up for use in a Virtual Client is relatively simple. On the **NetVault Selections** page, open the Virtual Client node, open **Plug-in *for Oracle***, and select the Oracle Server Instances, or the items contained within, for inclusion in the backup.

Restoring data

Restoring data to a Virtual Client is conducted in the same manner as a restore performed to a traditional NetVault Client. All options available for a restore with Plug-in *for Oracle* are also available for Failover Clustering environments, and data selection is also performed in the same way. The difference is that restorable backups of a Virtual Client are displayed on the **Create Selection Set** page under the name of the Virtual Client, not the specific NetVault Client or node that was active during each backup. When a restore job is initiated, NetVault communicates with all member Clients to determine which machine is in control of the failover cluster, and targets this machine for the restore.

i | **IMPORTANT:** The **Restore Backup taken from NetVault Client** field on the **Configure** dialog box must contain the name of the NetVault Virtual Client that performed the backup; otherwise, restores fail. For more information on using the **Configure** dialog box, see Configuring the plug-in.

To restore a Controlfile Autobackup or a backup taken from the CLI in a failover-cluster environment, provide the name of the NetVault Physical Client either in the **Restore Backup taken from NetVault Client** option or in the NV_ORIGINAL_NV_CLIENT option (use this variable with "send").

The Oracle RMAN backup pieces are stored in the NetVault Media Manager, under the name of the NetVault Client that performed the backup. The Client name is listed in the **Create Restore Job** - **Choose Saveset** page, in the **Saveset Information** area.

During a restore, you must have the Plug-in *for Oracle* configured to retrieve the backup pieces stored under the Client name that performed the backup.

Additionally, you can restore a NetVault Virtual Client to a non-clustered (standalone) NetVault Client.

When restoring data to a Virtual Client, Quest recommends that you select the applicable Virtual Client from the Target Client list in the Create Restore Job page.

All the instructions offered about performing a restore can be used in the recovery of a Virtual Client. For more information on restoring a NetVault Virtual Client, see the various topics in Restoring data.

Using the plug-in with Oracle Data Guard

- Oracle Data Guard: an overview
- Defining a deployment strategy
- Deployment
- Installing the plug-in in a Data Guard environment
- Backing up data
- Restoring data

Oracle Data Guard: an overview

Oracle's Oracle Data Guard "ensures high availability, data protection, and disaster recovery for enterprise data. Data Guard provides a comprehensive set of services that create, maintain, manage, and monitor one or more standby databases to enable production Oracle databases to survive disasters and data corruptions. Data Guard maintains these standby databases as copies of the production database. If the production database becomes unavailable because of an outage, Data Guard can switch any standby database to the production role, minimizing the downtime associated with the outage. Data Guard can be used with traditional backup, restoration, and cluster techniques to provide a high level of data protection and data availability."

With Data Guard, administrators can optionally improve production database performance by offloading resource-intensive backup and reporting operations to Standby Database Servers. In addition, backups can be used to recover either the Primary or Standby Database Server, regardless of whether the backups were taken from the Primary or the Standby Database Server.

Plug-in *for Oracle* provides support for a limited number of Oracle versions in a Data Guard environment. The installation, configuration, backup, and restore procedures differ slightly in a Data Guard environment. This topic offers specifics on how the plug-in operates in a Data Guard environment.

For a complete list of supported Oracle versions and platforms in a Data Guard environment, see the *Quest NetVault Compatibility Guide*.

> **i** | **NOTE:** The plug-in's RMAN backup method is the only method supported in Data Guard environments. Unless outlined in the following topics, backups and restores performed with the plug-in for Data Guard physical or standby databases are the same as backups and restores performed with non-Data Guard databases.

Defining a deployment strategy

Determining a strategy for protecting and recovering a Data Guard environment is essential to deploying and using the plug-in in the Data Guard environment. All strategies are valid for any of the supported deployments as detailed in Deployment.

- Back up from Primary only
- Back up from both Primary and Standby
- Back up from Standby only

Back up from Primary only

The simplest strategy is to perform all the backups from the Primary Database Server. You can use these backups to recover the Primary or one of the Standby Database Servers. While this strategy is valid, resource-intensive backup operations affect database performance on the Primary Database Server. With this strategy, the plug-in is installed on the Primary Database Server. Plug-in *for Oracle* is only required to be installed on the Standby Database Server when it is targeted for a restore.

Back up from both Primary and Standby

Offloading resource-intensive backup operations to the Standby Database Server is often one of the drivers for deploying Data Guard. The Oracle version dictates how much of the backup operations can be offloaded to the Standby Database Server:

- **Oracle 10.2.x**: Backing up datafiles and archived redo log files can be offloaded to the Standby Database Server while backups of the Control Files and the SPFILE must be performed on the Primary Database Server. In addition, backups of datafiles and archived redo log files can be used to recover either the Primary or Standby Database Server, regardless of whether the backups were taken from the Primary or Standby Database Server. For more information, see *RMAN to Back Up and Restore Files* in *Oracle 10.2.x Data Guard Concepts and Administration*.

- **Oracle 11.x**: Backups of datafiles, archived redo log files, Control Files, and SPFILE are interchangeable between the Primary and Standby Database Server, which means that all backups can be performed from one of the Standby Database Servers. In addition, you can use backups to recover either the Primary or Standby Database Server, regardless of whether the backups were taken from the Primary or the Standby Database Server. For more information, see *Using RMAN to Back Up and Restore Files* in *Oracle 11.x Data Guard Concepts and Administration*.

While Oracle 11.x does not require all backups to be performed on the Standby Database Server, a valid strategy for an Oracle 11.x environment is to mimic the requirements of a 10.2.x environment. In a 10.2.x environment, only datafiles and archived redo log files are offloaded to the Standby Database Server while the Control Files and SPFILE backups are performed from the Primary Database Server.

Another valid strategy for Oracle 10.2.x and 11.x is to perform Control File and SPFILE backups from the Primary Database Server while alternating the backups of the datafiles and archived redo log files between the Primary Database Server and the Standby Database Server. For example, Incremental Level 0 Backups are performed weekly on the Standby Database Server and the Incremental Level 1 backups are performed daily from the Primary Database Server. In addition, daily backups of the Control File and SPFILE are performed from the Primary Database Server.

Oracle's Maximum Availability Architecture (MAA) best practices recommend that backups be taken at both the primary and the standby databases to reduce MTTR, in case of double outages and to avoid introducing new site practices on switchover and failover.

With this strategy, Plug-in *for Oracle* is installed on both the Primary and at least one of the Standby Database Servers that performs backups.

NOTE: For Oracle versions before Oracle 11.x, backups of SPFILE were assumed to be usable at any other standby database. However, in practice, all standby databases cannot use the same SPFILE. Therefore, RMAN does not allow an SPFILE backup taken at one database site to be used at a different database site. This restriction is in place only when the **COMPATIBLE** initialization parameter is set to 11.0.0.

The standby database lets you offload all backup operations to one specific standby database, except the backups of SPFILE. However, if the **COMPATIBLE** initialization parameter is set to 11.0.0, the SPFILE can be backed up to disk and cataloged manually at standby sites where backups are written to tape or NetVault media. The additional metadata stored in SPFILE backup sets enables RMAN to identify which database SPFILE is contained in which backup set. Thus, the appropriate SPFILE backup is chosen during restore from tape or NetVault media.

Backup destination strategy

When implementing a strategy that backs up from both Primary and Standby Servers for an Oracle 10.2.x Data Guard environment, consider the accompanying backup-destination strategy, which is detailed in Defining a Backup Destination strategy.

Because RMAN stores backup metadata in the Recovery Catalog, a backup performed from the Primary Database Server to the FRA results in the RMAN backup pieces created during backup being registered in the Recovery Catalog. On the following day, if a backup performed from the Standby Database Server is also performed to the FRA, the RMAN backup pieces created during backup are also registered in the Recovery Catalog.

Later in the week, if a backup of the FRA is performed to NetVault Media Manager from the Standby Database Server, the FRA-to-NetVault Media Manager backup attempts to access backup sets that were created by the Primary Database Server's backups to the FRA. Because the primary and standby databases do not share the FRA, the FRA-to-NetVault Media Manager backup fails. RMAN attempts to access backup sets on the Primary Database Server that are not accessible from the Standby Database Server that is performing the FRA-to-NetVault Media Manager backup.

Therefore, for Oracle 10.2.x Data Guard environments, Quest recommends that backups to FRA only be performed from either the Primary Database Server or the Standby Database Server, but not both.

In Oracle 11.x Data Guard environments, the Oracle parameter **db_unique_name** allows the Recovery Catalog to distinguish the origin of backups to the FRA. When selecting FRA-to-NetVault Media Manager backups, RMAN backs up the backup sets in the FRA of the Database Server performing the FRA-to-NetVault Media Manager backup.

For more information on performing backups to NetVault Media Manager, see Setting backup options. For more information on FRA backups, see Flash Recovery Area (FRA) backups.

Back up from Standby only

Starting with Oracle 11.x, backups of datafiles, archived redo log files, Control Files, and SPFILE are interchangeable between the Primary and Standby Database Server. This interchangeability means that all backups can be performed from one of the Standby Database Servers. With this strategy, Plug-in *for Oracle* is installed on both the Primary Database Server and at least one of the Standby Database Servers that performs backups. The plug-in must be installed on the Primary Database Server when it is targeted for a restore; that is, if you want to use the backup of the Standby Database Server to recover the Primary Database Server, you need to install the plug-in on the Primary Database Server.

NOTE: Because Oracle 10.2.x only allows the backing up of the datafiles and the archived redo log files to be offloaded to the Standby Database Server while requiring the backups of the Control Files and the SPFILE to be performed on the Primary Database Server, the Back-up-from-Standby-only strategy is not supported for Oracle 10.2.x Data Guard environments.

Deployment

Plug-in *for Oracle*'s support in Data Guard environments is restricted to physical standby databases. Logical and Snapshot standby databases are not supported. A physical standby database is a transactionally consistent copy of the production database. All references to standby databases see physical standby databases.

The plug-in supports the following types of Data Guard deployments:

Table 4. Supported deployments

Primary Database Server configuration		Physical Standby Server configuration
Single-Instance	-->	Single-Instance
Multi-Instance RAC	-->	Single-Instance
Multi-Instance RAC	-->	Multi-Instance RAC

Deploying the plug-in in a Data Guard environment requires that you install the plug-in on the Primary Database Server and at least one of Physical Standby Servers—based on the backup strategy selected. For more information on selecting a backup strategy, see Defining a deployment strategy.

- Single-instance to single-instance Data Guard deployment
- Multi-instance RAC to single-instance Data Guard deployment
- Multi-instance RAC to multi-instance RAC Data Guard deployment

Single-instance to single-instance Data Guard deployment

In a single-instance to single-instance Data Guard deployment, the plug-in is installed on both the Primary and at least one of the Standby Database Servers.

Figure 8. Single-instance to single-instance Data Guard deployment
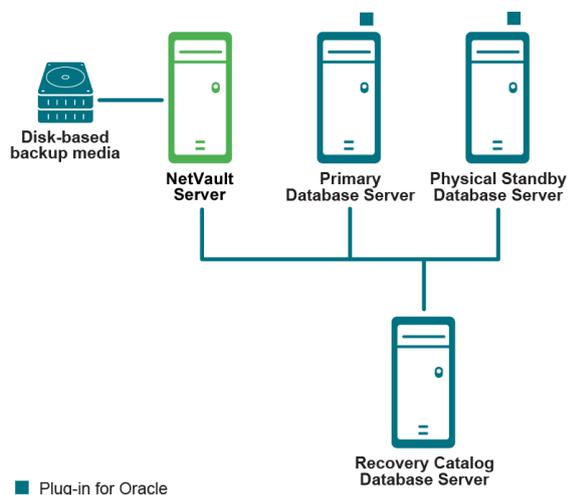
Multi-instance RAC to single-instance Data Guard deployment

In a multi-instance to single-instance Data Guard deployment, the plug-in is installed in the Primary RAC Database environment and at least one of the Standby Database Servers.

Figure 9. Multi-instance RAC to single-instance Data Guard deployment
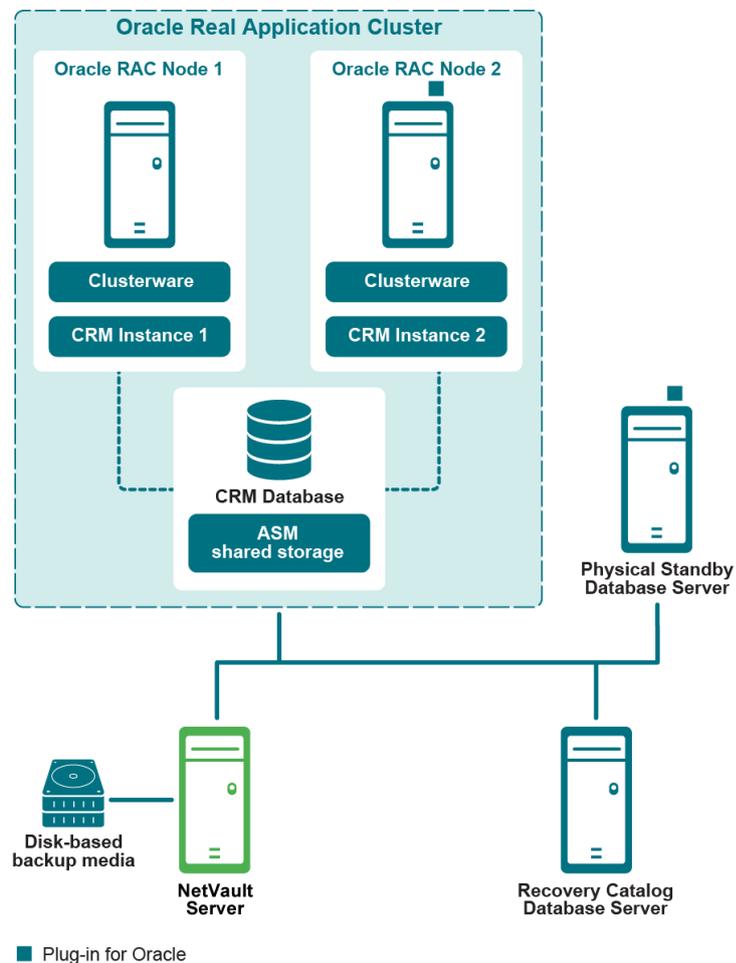
For more information on deploying the plug-in in a multi-instance RAC environment, see Defining a deployment strategy.

Multi-instance RAC to multi-instance RAC Data Guard deployment

In a multi-instance to multi-instance Data Guard deployment, the plug-in is installed in the Primary RAC Database environment and at least one of the Standby Database Servers.

Figure 10. Multi-instance RAC to multi-instance RAC Data Guard deployment
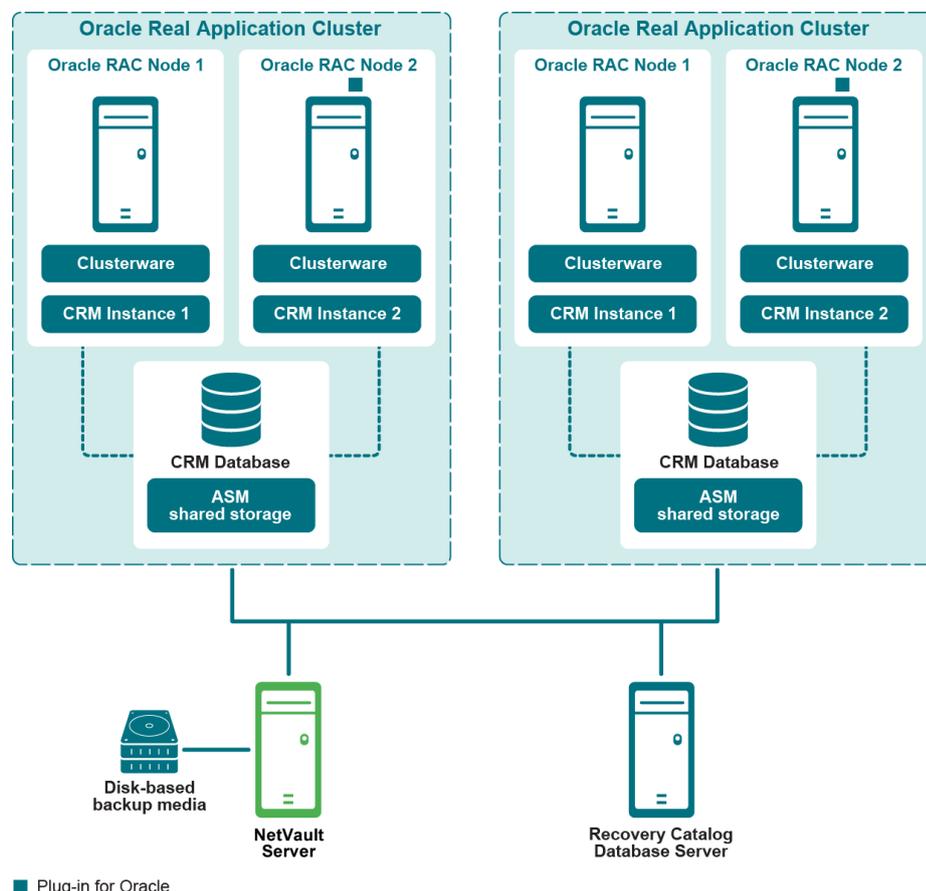
■ Plug-in for Oracle

For more information on deploying the plug-in in a multi-instance RAC environment, see Defining a deployment strategy.

Installing the plug-in in a Data Guard environment

Before completing any additional or different steps that are outlined in the following topics, review the information provided in earlier topics, such as Installing and removing the plug-in and Configuring the plug-in.

* Installation prerequisites

* Installing or upgrading the plug-in

Installation prerequisites

The following prerequisites must be met before you can install Plug-in *for Oracle* in a Data Guard environment:

- **Oracle Data Guard environment in place**: You must have a properly configured Data Guard environment. The version of Oracle is listed as a Supported Data Guard environment in the *Quest NetVault Compatibility Guide*.

- **Oracle RAC environment in place**: For Data Guard environments that include multi-instance Primary or Standby Database Servers, meet the Oracle RAC-required configuration that is specified in RAC configuration requirements.

- **Separate NetVault Server machine**: The machine that is to serve as the NetVault Server must be properly configured. This machine must exist outside the Data Guard environment, but have network connectivity to the nodes and hosts within the Data Guard environment.

- **Separate Recovery Catalog Server**: Quest recommends that you store the RMAN repository in a Recovery Catalog database on a server outside the Data Guard environment. Only the primary database can be registered with the Recovery Catalog. For more information, see Creating the RMAN repository.

- **Control File Autobackups are enabled**: Automatic backups of the Control File and SPFILE must be enabled on the databases that meet the following criteria:

  - **Back up from Primary Only strategy:** Enable Autobackups of the Control File and SPFILE on the Primary Database Server.

  - **Back up from both Primary and Standby strategy:**

    - **Oracle 10.2.x**: Disable Autobackups of the Control File and SPFILE on both the Primary and all the Standby Database Servers where backups are performed. Manual Control File and SPFILE backups are performed on the Primary Database Server.

    - **Oracle 11.x**: Enable Autobackups of the Control File and SPFILE on all the database servers where backups are performed. For example, if Incremental Level 0 Backups are taken from Standby Database Server while Incremental Level 1 Backups are taken from the Primary Database Server, enable Autobackups on both the Primary and Standby Database Server.

  - **Back up from Standby Only strategy:** Enable Autobackups of the Control File and SPFILE on the Standby Database Servers where backups are being performed.

  For more information, see Enabling Control File Autobackups.

- **Flash Recovery Area recommended**: Enabling FRA is recommended for Data Guard environments. For more information, see *Configuring the Flash Recovery Area* in the *Oracle Database Backup and Recovery User's Guide*.

- **Flashback Database recommended**: Enabling the Flashback Database is recommended on Data Guard Standby Database Servers. For more information, see Enabling Flashback Database and the *Using Flashback Database After a Role Transition* section in *Oracle Data Guard Concepts and Administration*.

- **Additional prerequisites:** In addition to the preceding Data Guard-specific requirements, meet the non-Data Guard-specific requirements that are specified in Prerequisites.

Optional RMAN configuration settings

The following settings are optional in a Data Guard environment.

Retention Policy

RMAN's Retention Policy specifies a persistent, ongoing policy for backup sets and copies those sets that RMAN marks as obsolete, that is, not needed and eligible for deletion. As time passes, RMAN marks backup sets and copies as obsolete according to the criteria specified in the retention policy. RMAN automatically deletes obsolete backup sets and copies in the FRA when space is needed. RMAN does not automatically delete obsolete files outside the FRA, such as on NetVault backup media.

Primary database configuration

1 After making sure that the Recovery Catalog database is open, connect RMAN to the primary target database and to the Recovery Catalog database.

For example, issue the following commands to connect to the catalog database:

```
> rman TARGET / CATALOG rman/<password>@<connect_identifier for catalog>
```

2 Configure the retention policy:

```
CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF <n> DAYS
```

Archive Log Deletion Policy

RMAN's Archive Log Deletion Policy specifies that archived redo logs are eligible for deletion if both of the following conditions are met:

- The archived redo logs have been applied to the required standby databases.

- The logs are not needed by the **BACKED UP … TIMES TO DEVICE TYPE** deletion policy. If the **BACKED UP** policy is not set, this condition is always met.

Optional settings

- When backups of archived redo log files are taken on a standby database:

  - Primary database configuration:

    a After making sure that the Recovery Catalog database is open, connect RMAN to the primary target database and to the Recovery Catalog database.

    For example, issue the following commands to connect to the catalog database:

    ```
    > rman TARGET / CATALOG rman/<password>@<connect_identifier
        for catalog>
    ```

    b Configure the deletion policy:

    ```
    CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON STANDBY
    ```

  - Configuration for standby database where backups are performed:

    a After making sure that the Recovery Catalog database is open, connect RMAN to the standby target database performing backups and to the Recovery Catalog database.

    b Configure the deletion policy:

    ```
    CONFIGURE ARCHIVELOG DELETION POLICY TO NONE
    ```

  - Configuration for standby database where backups are *not* performed:

    a After making sure that the Recovery Catalog database is open, connect RMAN to the standby target database where backups will not be performed and to the Recovery Catalog database.

    b Configure the deletion policy:

    ```
    CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON ALL STANDBY
    ```

- When backups of archived redo log files are taken on the primary database:

  - Primary database configuration:

    a After making sure that the Recovery Catalog database is open, connect RMAN to the primary target database performing backups and to the Recovery Catalog database.

    b Configure the deletion policy

    ```
    CONFIGURE ARCHIVELOG DELETION POLICY TO NONE
    ```

  - Standby database configuration:

a   After making sure that the Recovery Catalog database is open, connect RMAN to the standby target database and to the Recovery Catalog database.

b   Configure the deletion policy:

```
CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON STANDBY
```

Backup optimization

RMAN Backup Optimization prevents RMAN from backing up a file to a device type if the identical file is already backed up on the backup media. However, RMAN does not signal an error if backup optimization causes all files to be skipped during a backup.

Configuration for standby database performing backups

1   After making sure that the Recovery Catalog database is open, connect RMAN to the standby target database performing backups and to the Recovery Catalog database.

For example, issue the following commands to connect to the catalog database:

```
> rman TARGET / CATALOG rman/<password>@<connect_identifier for catalog>
```

2   Configure the backup optimization:

```
CONFIGURE BACKUP OPTIMIZATION ON
```

> **i** | **IMPORTANT:** In a switchover or failover where the databases change roles, the corresponding **CONFIGURE** commands must be re-run on the new primary and standby databases.

Installing or upgrading the plug-in

In a Data Guard environment, you must install the plug-in on the Primary Database Server. Optionally, you can also install the plug-in on one or more Physical Standby Database Servers. For more information on the supported Data Guard deployments, see Deployment.

1   **Install the NetVault Client software.**

Install the NetVault Client software on each Oracle Database Server where the plug-in will be installed.

2   **Add the NetVault Clients to the NetVault Server.**

From the WebUI on the NetVault Server, open the **Manage Clients** page. Add the NetVault Clients installed on the primary and standby databases to the NetVault **Clients** list.

3   **Install Plug-in** *for Oracle***.**

With the NetVault Client software installed on each required system of the Data Guard environment, install the plug-in on each system as well. The process is identical to installing the plug-in for a non-Data Guard single or multi-instance database environment.

Backing up data

RMAN-based backups are the only supported backup method for Data Guard environments. RMAN is Data Guard-aware in that it uses its metadata to behave transparently across the primary and physical standby databases in the Data Guard environment.

Performing backups in a Data Guard environment is almost identical to performing backups for a single- or multi-instance database in a non-Data Guard environment. The differences are described in the following topics.

*   Creating a User Defined Backup Tag

*   Performing manual backups of Control File and SPFILE

For more information on selecting data for backup, setting backup options, and finalizing and submitting the job, see Performing RMAN backups.

Creating a User Defined Backup Tag

When a Data Guard strategy of backing up from both the Primary and Standby Database Server is implemented with backups from the Primary and Standby Database Servers being interchangeable, you might want to be able to select the source of the backup during a restore. For example, you might want to restore from the Primary Database Server versus the Standby Database Server. The **User Defined Backup Tag** option lets you assign a tag to the backups that indicates the source of the backup—the Primary Database Server or the Standby Database Server. During the restore, the **User Defined Backup Tag** is specified in the **Restore Options** to ensure that only backups from the applicable server are used for the restore.

1   On the **NetVault Plugin Options** page, click the **RMAN Backup Options** tab.

2   Select the **User Defined Backup Tag** option, and enter a tag name.

     This option creates a tag name for the backup set. Use only characters that are allowed in filenames on the target file system. For example, ASM does not support the use of hyphens (-) in filenames that it uses internally, so you cannot use a hyphen in the tag name, such as, **weekly-incremental**, if you are storing backups in ASM disk groups.

Performing manual backups of Control File and SPFILE

For Oracle 10.2.x Data Guard environments where the back-up-from-both-Primary-and-Secondary strategy has been deployed or Oracle 11.x Data Guard environments where the Control File and SPFILE backups are the only backups that are performed on the Primary Database Servers, Manual Control File and SPFILE backups are performed.

1   Ensure that Control File and SPFILE Autobackups are disabled.

     For more information, see Step 4 under Adding a database.

2   In the selection tree, select the **Parameter File** and **Control File** nodes.

3   On the **Backup Option** tab of the **NetVault Plugin Options** page, select the following options:

     ▪   In the **Backup Method** section, select **RMAN**.

     ▪   In the **Backup Destination** section, select either **NetVault Media Manager** or **Both NetVault Media Manager and Disk (Backup to FRA then Backup FRA to NetVault Media)**. This step ensures that the Control File and SPFILE are stored on backup media for disaster recovery purposes.

     ▪   In the **Backup Method** section, select **Full**.

4   Use the **Schedule**, **Target Storage**, and **Advanced Options** lists to configure any additional required options.

5   Click **Save** or **Save & Submit**, whichever is applicable.

Restoring data

When performing restores in a Data Guard environment where the backup is being restored to the *same* Database Server where the backup was taken, there is no difference between performing the restore in a Data Guard environment or a non-Data Guard environment. For more information on single-instance environments, see Performing RMAN restores. For more information on multi-instance environments, see Restoring data in a RAC environment.

When performing restores in a Data Guard environment where the backup is restored to an *alternate* Database Server in the Data Guard environment, the restore is identical to restoring to the same Database Server except for the differences detailed in the following topics.

- Configuring a restore to an alternate server in a Data Guard environment
- Restoring data to an alternate server in a Data Guard environment
- Recovering Whole Database to Primary Database Server

Configuring a restore to an alternate server in a Data Guard environment

When performing backups from either the Primary or Standby Database Server, the index for the backup is stored in the NetVault Database under the NetVault Client Name of the Database Server where the backup was performed. To restore to an alternate Database Server, the plug-in configuration of the alternate (target) server of the restore must be modified to specify the name of the source NetVault Client.

To configure the backup taken from NetVault Client option, perform the following steps.

1   In the Navigation Pane, click **Create Backup Job**, and click ➕ next to the **Selections** list.

2   In the selection tree, open the NetVault Client that is configured as the alternate or target Oracle Database Server, click **Plug-in *for Oracle***, and select **Configure** from the context menu.

3   On the **Configure** dialog box, complete the **Restore Backup taken from NetVault Client** field by specifying the NetVault Machine Name for the alternate or target Database Server.

    When a restore is performed by the plug-in, this field indicates the NetVault Client from which the original backup was performed.

Restoring data to an alternate server in a Data Guard environment

When performing restores in a Data Guard environment where the backup is restored to an alternate Database Server, the process is nearly identical to restoring to the same Database Server. The differences are described in the following topics.

To restore a backup that used the **User Defined Backup Tag** option, perform the following steps.

1   On the **Restore Options** tab, select the **Restore/Recover from User Defined Backup Tag** check box.

2   Complete the associated field by specifying the tag that corresponds to the backup set for the Database Server from which you want to restore.

3   Ensure that the backup is restored to an alternate Database Server.

    On the **Create Restore Job** page, use the **Target Client** list to select the NetVault Machine Name of the alternate or target Database Server for the restore.

Recovering Whole Database to Primary Database Server

This type of recovery is performed when the current Control File and SPFILE are intact but all the datafiles are damaged on the Primary Database Server, and the backup from the Standby Database Server is used for the recovery. You can perform a Complete Recovery or recover the database to a specific point.

An overview of the process to recover the whole database from a backup of the Standby to the Primary Database Server follows.

1 On the **Create Restore Job—Choose Saveset** page, find the latest available backup on the Standby Database Server that included *all* the tablespaces, and click **Next**.

2 On the **Create Selection Set** page, select the **Whole Database** node, and click 🖊 .

3 Click the **Restore Options** tab, and then select the following:

   ▪ In the **Pre-Restore Options** section, select **Whole Database Restore**.

   ▪ In the **Restore Options** section, select **Restore/Recover from User Defined Backup Tag**, and specify the **User Defined Backup Tag** for the Standby Database Server.

4 To perform recovery on the database, that is, reconstruct the database up to the current or specified point, select one of the following options in the **Recovery Type** section on the **Perform Recovery** tab:

   ▪ **Perform Complete Recovery**: Recovers the database to the current time, that is, to the last committed transaction available in the archived and online redo logs.

   ▪ **Perform Database Point in Time Recovery**: Recovers the database to a specific point-in- time, whether it is a specific time, log sequence number, or SCN.

5 On the **Recovery Catalog** tab, select the **Use Catalog** option, and specify the Recovery Catalog connection details.

6 If the Oracle SID of the Primary Database Server is different from the Oracle SID in the standby database, select the **Destination Details** tab, and complete all fields.

   Enter the **Target Service** of the Primary Database Server, the **Oracle SYSDBA User Name**, and the **Oracle SYSDBA Password**.

   > ⓘ | **IMPORTANT:** In a Data Guard environment, the primary database and standby database have the same database name and the same DBID. However, the Oracle SID might not be the same to differentiate the database servers.

7 In the **Target Client** list, select the NetVault Client Name of the Primary Database Server.

8 Continue with the restore procedure, including completion of the **Schedule** and **Advanced Options** lists.

   For more information, see Finalizing and submitting the job.

   RMAN automatically chooses the best available Full, Incremental, or Archived Log backups and restores and recovers the database.

Using the plug-in with Oracle Container Databases (CDBs) and Pluggable Databases (PDBs)

* Oracle PDBs: an overview
* Enabling connection to a PDB
* Querying CDB, PDB, and non-CDB databases
* Restrictions for backing up and restoring a PDB
* Backing up data
* Restoring data

Oracle PDBs: an overview

With Oracle Multitenant Architecture, you can create an Oracle database that functions as a multitenant Container Database (CDB). You also can create non-CDB databases. In a CDB, you can create one or more Pluggable Databases (PDBs) or you can opt not to create any. A PDB is a portable collection of Oracle schemas, schema objects, and nonschema objects that are presented to an Oracle client as a non-CDB database. Prior to Oracle Database 12c, all Oracle databases were non-CDBs. For more information, see the Oracle 12c Multitenant Architecture documentation.

Enabling connection to a PDB

To connect to a PDB that you have created, you must ensure that an entry is added in the Oracle **"tnsnames.ora"** file and the status of the PDB is in **read-write** mode. When a PDB is created, a default Oracle service with the same name as the Pluggable Database name is created automatically. However, an entry to the **"tnsnames.ora"** file might not have been added.

The following is an example of a **"tnsnames.ora"** file with only an entry of a CDB called **salesdb**:

```
SALESDB =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = salesdb)
    )
  )

LISTENER_SALESDB =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))

ORACLR_CONNECTION_DATA =
```

```
(DESCRIPTION =
  (ADDRESS_LIST =
     (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
  )
  (CONNECT_DATA =
     (SID = CLRExtProc)
     (PRESENTATION = RO)
  )
)
```

To enable connection to a PDB

1   After you create a PDB, ensure that the entry for the PDB has been added to the **"tnsnames.ora"** file.

   If not, add the entry manually. The following example shows a manually edited **"tnsnames.ora"** file after two PDBs, **sales_region200_pdb** and **sales_region300_pdb**, have been created:

```
SALESDB =
  (DESCRIPTION =
     (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
     (CONNECT_DATA =
        (SERVER = DEDICATED)
        (SERVICE_NAME = salesdb)
     )
  )

LISTENER_SALESDB =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))

SALES_REGION200_PDB =
  (DESCRIPTION =
     (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
     (CONNECT_DATA =
        (SERVER = DEDICATED)
        (SERVICE_NAME = sales_region200_pdb)
     )
  )

SALES_REGION300_PDB =
  (DESCRIPTION =
     (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
     (CONNECT_DATA =
        (SERVER = DEDICATED)
        (SERVICE_NAME = sales_region300_pdb)
     )
  )

ORACLR_CONNECTION_DATA =
  (DESCRIPTION =
     (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
     )
     (CONNECT_DATA =
        (SID = CLRExtProc)
        (PRESENTATION = RO)
     )
  )
```

2   Ensure that the status of the PDB is set to **read-write**.

   In the following example, a query is run to list the open-mode status of the PDBs. After that, a query is entered to change the open mode of PDB sales_region200_pdb to read-write.

```
SQLPLUS / AS SYSDBA

SQL> column name format a20
```

```
SQL> select name, open_mode from v$pdbs;

NAME OPEN_MODE
-------------------- ----------
PDB$SEED READ ONLY
SALES_REGION200_PDB   MOUNTED
SALES_REGION300_PDB   MOUNTED

SQL> alter pluggable database sales_region200_pdb open;

   Pluggable database altered.
```

The following example shows how to open all the PDBs with a single query:

```
SQL> alter pluggable database all open;
   Pluggable database altered.

SQL> column name format a20
SQL> select name, open_mode from v$pdbs;

NAME OPEN_MODE
-------------------- ----------
PDB$SEED READ ONLY
SALES_REGION200_PDB   READ WRITE
SALES_REGION300_PDB   READ WRITE
```

Querying CDB, PDB, and non-CDB databases

You can perform various queries of the databases by using **SQL*Plus**.

- Determining whether a database is a CDB
- Listing the containers in a CDB
- Listing the PDBs and their status in a CDB

Determining whether a database is a CDB

The following is an example of the query and response for a non-CDB database:

```
SQLPLUS / AS SYSDBA

SQL> select name, cdb from v$database;

NAME      CDB
--------- ---
PRODDB    NO

SQL>
```

The following is an example of the query and response for a CDB database:

```
SQLPLUS / AS SYSDBA

SQL> select name, cdb from v$database;

NAME      CDB
--------- ---
SALESDB   YES

SQL>
```

Listing the containers in a CDB

You can list the containers in a CDB. For example:

```
SQL> column name format a20

SQL> select name, con_id, dbid, con_uid from v$containers order by con_id;

    NAME                    CON_ID      DBID      CON_UID
    -------------------- ----------   ----------   ----------
    CDB$ROOT                      1   3924627550            1
    PDB$SEED                      2   3283839514   3283839514
    SALES_REGION200_PDB           3   3022227374   3022227374
    SALES_REGION300_PDB           4   1471149548   1471149548
```

Listing the PDBs and their status in a CDB

You can list the PDBs and their status in a CDB. For example:

```
SQL> column pdb_name format a20

SQL> select pdb_id, pdb_name, status from dba_pdbs order by pdb_id;

    PDB_ID     PDB_NAME             STATUS
    ---------- -------------------- ----------
             2 PDB$SEED             NORMAL
             3 SALES_REGION200_PDB  NORMAL
             4 SALES_REGION300_PDB  NORMAL
```

Restrictions for backing up and restoring a PDB

If you intend to target a PDB for backup and restore jobs, be aware that the following operations are *not* available:

- Connecting to a Recovery Catalog. Because the PDB cannot connect to a Recovery Catalog, Quest recommends that you run backups that target the CDB instead of a PDB.

- Backing up, restoring, and deleting archive logs. If necessary, RMAN automatically restores archive logs during media recovery.

- Using the **CONFIGURE** command

- Using DBPITR and Flashback recovery when using **shared undo** mode

- Using Tablespace Point-in-Time Recovery (TSPITR)

- Using table recovery

- Using duplicate database

- Using the **IMPORT CATALOG** command

- Using the **REGISTER DATABASE** command

- Using report/delete obsolete

- Using the **RESET DATABASE** command

Backing up data

You can use the WebUI or CLI to back up the CDB and PDBs.

- Backing up the CDB and PDBs from the WebUI
- Backing up the CDB and PDBs from the CLI

Backing up the CDB and PDBs from the WebUI

To set up the plug-in for backup and restore of a CDB using the WebUI, use the same process outlined in Adding a database.

Performing backups is almost identical to the process described in Performing RMAN backups. You must use sets—Backup Selection Set, Backup Options Set, Schedule Set, Target Set, and Advanced Options Set—to create a backup job. Note the following differences:

- To include *all* the datafiles of the CDB root container, the seed container, and all the datafiles of the plugged-in PDBs, select the **Database node** or the **All Tablespaces node** of a CDB.

- To include only individual tablespaces or individual datafiles of a CDB, select the specific tablespaces or datafiles of the root container. Note that this option excludes the tablespaces or datafiles of PDBs from the backup.

- You cannot add a PDB to the WebUI; PDBs are protected as a part of the CDB to which they are plugged. If applicable, you can use the CLI commands described in Backing up the CDB and PDBs from the CLI to back up only the contents of a PDB.

Backing up the CDB and PDBs from the CLI

When you back up the CDB and PDBs from the CLI, you have the option of backing up everything or backing up only the PDB that is targeted by a specific CDB or a specific PDB.

To back up the CDB and all PDBs

```
rman catalog rcat/rcat@rcatalog

RMAN> connect target /

   connected to target database: SALESDB (DBID=3924627550)
   using target database control file instead of recovery catalog

RMAN> run { allocate channel ch1 type sbt;
2> backup database include current controlfile
3> plus archivelog; }
```

The CLI displays the applicable output regarding the status of the backup.

To back up a PDB that is connected to a CDB as a target

```
RMAN> run { allocate channel ch1 type sbt;

2> backup pluggable database sales_region300_pdb; }
```

The CLI displays the applicable output regarding the status of the backup.

To back up a PDB that is connected to a PDB as a target

```
rman nocatalog

RMAN> connect target pdbadmin/pdbadmin@sales_region200_pdb
   connected to target database: SALESDB:SALES_REGION200_PDB (DBID=3022227374)
   using target database control file instead of recovery catalog
```

```
RMAN> run { allocate channel ch1 type sbt;
2> backup database; }
```

The CLI displays the applicable output regarding the status of the backup.

Restoring data

You can use the WebUI or CLI to back up the CDB and PDBs.

- Restoring the CDB and PDBs from the WebUI
- Restoring the CDB and PDBs from the CLI

Restoring the CDB and PDBs from the WebUI

If you use the WebUI to create and run a backup of a CDB, including its PDBs, you can restore the whole CDB and individual tablespaces or datafiles using the steps provided in Performing RMAN restores. For a more granular restore, you can use the CLI as described in Restoring the CDB and PDBs from the CLI.

Restoring the CDB and PDBs from the CLI

When you restore the CDB and PDBs from the CLI, you have the option of restoring everything or restoring only the PDB that is targeted by a specific CDB.

To restore the CDB

CDBs are restored in a manner similar to restoring non-CDBs. The following example shows the restore of a tablespace from the CDB.

```
rman catalog rcat/rcat@rcatalog

RMAN> connect target /
    connected to target database: SALESDB (DBID=3924627550, not open)
    using target database control file instead of recovery catalog

RMAN> run { allocate channel ch1 type sbt;
2> restore tablespace USERS; }

    allocated channel: ch1
    channel ch1: SID=22 device type=SBT_TAPE
    channel ch1: Quest(R) NetVault(R) Backup Plug-in for Oracle. v.12.2.0.9

    Starting restore at 28-NOV-18

    channel ch1: starting datafile backup set restore
    channel ch1: specifying datafile(s) to restore from backup set
    channel ch1: restoring datafile 00007 to
                 Q:\APP\ORACLEUSER\ORADATA\SALESDB\USERS01.DBF
    channel ch1: reading from backup piece 0rtjacic_1_1
    channel ch1: piece handle=0rtjacic_1_1 tag=TAG20181128T002027
    channel ch1: restored backup piece 1
    channel ch1: restore complete, elapsed time: 00:00:15
    Finished restore at 28-NOV-18
    released channel: ch1

RMAN>
```

To restore a PDB that is connected to a CDB as a target

```
RMAN> run { allocate channel ch1 type sbt;
2> restore pluggable database sales_region300_pdb;
3> recover pluggable database sales_region300_pdb; }
```

The CLI displays the applicable output regarding the status of the restore.

```
sqlplus / as sysdba

SQL> alter pluggable database sales_region300_pdb close;
   Pluggable database altered.
   SQL> exit

rman catalog rcat/rcat@rcatalog

RMAN> connect target /

   connected to target database: SALESDB (DBID=3924627550)
   using target database control file instead of recovery catalog

RMAN> run { allocate channel ch1 type sbt;
2> set until scn=1860018;
3> restore pluggable database sales_region300_pdb;
4> recover pluggable database sales_region300_pdb auxiliary
      destination='C:\oracle_recovery';
5> alter pluggable database sales_region300_pdb open resetlogs; }
```

The CLI displays the applicable output regarding the status of the restore.

```
rman nocatalog

RMAN> connect target pdbadmin/pdbadmin@sales_region300_pdb
   connected to target database: SALESDB:SALES_REGION300_PDB
      (DBID=1471149548, not open)

RMAN> run { allocate channel ch1 type sbt;
2> restore database;
3> recover database; }
```

The CLI displays the applicable output regarding the status of the restore.

Troubleshooting

- Unable to add a database
- Backup fails
- Backup job hangs
- Restore fails
- Oracle RMAN output is not displayed correctly in binary logs

Unable to add a database

Problem

When attempting to add a database to the plug-in, the following error appears:

```
Invalid Credentials.
Unable to configure plugin to connect to this database.
```

Possible Cause

Connection to the database cannot be established due to LDAP failure.

Resolution

Disable LDAP, and try adding the database again.

Backup fails

This topic describes some common errors and their solutions. In cases where an error occurs and is not described in this table, obtain the Oracle database error number from the NetVault Logs, and see the relevant Oracle documentation for resolution.

Table 5. Troubleshooting backup failures

Error	Explanation
ORA-00245: control file backup operation failed	For RAC environments that use Oracle 11g Release 2 or later, ensure that the snapshot Control File is shared by all nodes in the RAC database. Check the CONFIGURE SNAPSHOT CONTROLFILE NAME TO parameter to determine the current setting. For more information, see Installation prerequisites.
ORA-01031: insufficient privileges on Oracle Backup error	User name specified in the **Oracle SYSDBA User Name** field of the **Add**, or **Edit**, **Oracle Database** dialog box must have SYSDBA privileges, be included in the Oracle Password File, or be a member of the ORA_DBA OS group. Additionally, if the SYSDBA user is specified in the Oracle Password File, the **Use Oracle Password File Authentication** option should be selected in the plug-in's **Configure** dialog box.
Failed to Connect to Database or ORA-01034: Oracle is not available	Oracle Instance specified in the **Oracle SID** field of the **Add**, or **Edit**, **Oracle Database** dialog box must be in an **OPEN** state. The instance must be in an **OPEN** state to perform backup.
ORA-01123: Cannot start online backup; media recovery not enabled	Oracle Instance is not running in ARCHIVELOG mode as specified in *Prerequisites*.
RMAN-06062: cannot backup SPFILE because the instance was not started with SPFILE	If either **Control File Autobackup Enabled** or **Use RMAN Commands to backup SPFILE** are selected on the **Add**, or **Edit**, **Oracle Database** dialog box, the database must be started with an SPFILE. Create an SPFILE and restart the instance using the SPFILE or use the **Edit Oracle Database** option to clear these two options and insert the path for the PFILE in the **Parameter File(s) Path** field.
RMAN-06180: incremental backups require Enterprise Edition	For editions other than Oracle Enterprise Edition, **Incremental Cumulative** and **Incremental Differential** backups are not supported. Select either **Full** or **Incremental Level 0** for the **Backup Type**.
RMAN-06181: multiple channels require Enterprise Edition	Ensure that **Number of Channels** is set to **1**.
Failed to back up all data	If the SYSDBA password has been changed on the Oracle Server, it must be updated in the plug-in's Database Configuration. For more information, see Removing a saved password.
Flash Recovery Area Backup Fails with 'specification does not match any datafile copy in the repository' or 'specification does not match any backup set in the repository' errors.	A **Flash Recovery Area** backup was performed; however, the FRA does not contain any recovery files for previous backups that were performed with a **Backup Destination** of **Disk** or **Both NetVault Media Manager and Disk**. You must use the FRA as a **Backup Destination** before you can perform a backup of the FRA with the **Flash Recovery Area** backup type.
Backup fails when using DR Series system or QoreStor as the storage device for Oracle Database 12.x or later on Linux or UNIX	If you are running Oracle Database 12.x or later on Linux or UNIX, backups might fail if the device you select for **Target Storage** is a DR Series system or QoreStor storage device. For more information on avoiding this issue, see Using DR Series or QoreStor as the storage device for Oracle Database 12.x or later on Linux or UNIX.

Backup job hangs

Problem

There is a lock on the Control File and a backup job hangs at log output: "Log Message: Starting Control File and SPFILE Autobackup at *<date>*." Or this job is canceled and a subsequent backup job completes with an error message in the log of the job: "ORA-00230: operation disallowed: snapshot control file enqueue unavailable."

Possible causes

- Plug-in *for Oracle* has not been properly configured; that is, the **NetVault Server** field has not been set on the **Configure** dialog box.

- **Control File Autobackup** has been enabled without proper configuration of the plug-in.

Resolution

1   Cancel the hung backup job.

2   Remove the lock on the Control File.

    a   Start **SQL*Plus**, and connect with administrator privileges to the local instance.

```
SQLPLUS SYS AS SYSDBA
```

    b   Verify that there is a lock on the Control File, and obtain the offending SID with the following query:

```
SELECT s.SID, USERNAME AS "User"
FROM V$SESSION s,V$ENQUEUE_LOCK l WHERE l.SID = s.SID
AND l.TYPE = 'CF' AND l.ID1 = 0 AND l.ID2 = 2;
```

In the following example, the offending SID is 133.

```
SQL> SELECT s.SID, USERNAME AS "User" FROM V$SESSION s,
   V$ENQUEUE_LOCK l WHERE l.SID = s.SID AND l.TYPE =
   'CF' AND L.ID1 = 0 AND l.ID2 = 2;
     SID User
-------- ------------------
     133 SYS
SQL>
```

    c   Obtain the Process ID (SPID) that matches the SID with the following query:

```
SELECT s.sid,p.spid
FROM v$process p, v$session s
WHERE p.addr = s.paddr order by s.sid;
```

In the following example, the offending SPID is 3184.

```
SQL> SELECT s.sid,p.spid FROM v$process p, v$session s
   WHERE p.addr = s.paddr order by s.sid;
     SID SPID
-------- ------------------
     130 12118
     132 5643
     133 3184
     134 3100
     135 12116
```

    d   Exit **SQL*Plus**, and stop the Process in the OS shell:

```
Kill -9 <spid>
```

Example:

```
SQL> exit
```

```
          Disconnected from Oracle Database 10g Enterprise Edition
             10.2.0.1.0 - 64 bit Production
          With the Partitioning, Real Application Clusters, OLAP and
             Data Mining options
          izmir:/export/home/o10g => kill -9 3184
          izmir:/export/home/o10g =>
```

3   Ensure that the **NetVault Server** and **Do Restore from NetVault Server** fields are defined in the default
attributes as detailed in Configuring default settings.

Restore fails

Table 6. Troubleshooting restore failures

Error	Explanation
• ORA-19511: Error received from media manager, error text: SBT error=7011, errno=1, sbtopen: system error • ORA-19511: Error received from media manager layer, error text: SBT error= 7063, errno = 0, sbtread: i/o error • ORA-27007: failed to open file • ORA-27012: Read from file failed	Ensure that **NetVault Server** and **Do Restore from NetVault Server** settings are defined as detailed in *Configuring default settings*.
RMAN-06054: media recovery requesting unknown log: thread *<n>* seq 40 scn *<n>*	This error might occur while performing a **Clone Database** operation. This error results because the old Control File has no record of the new archive log. Running the RMAN command **ALTER DATABASE OPEN RESETLOGS** brings the database online.
Warning (.. ..) Data Plug-in (… …) Failed with exit code 5	This warning appears if Block Media Recovery is selected with Oracle 11g databases. In spite of the "failed" message, the recovery completes with success and full access to the Oracle recovered data is granted. The **code 5** is a warning issued by Oracle 11g to let you know that recovery is necessary. Because the plug-in already includes the RMAN recovery command while performing Block Media Recovery, this message can be ignored.

Oracle RMAN output is not displayed correctly in binary logs

Problem

When running RMAN based jobs, the plug-in redirects the output from Oracle RMAN to the NetVault binary logs. In this manner, the output from Oracle RMAN can be viewed in the NetVault binary logs. However, the output for Oracle RMAN is either incorrectly displayed or not displayed in the NetVault binary logs.

Possible causes

Depending on the NLS_LANG setting, the output might not be displayed correctly or is displayed incorrectly because the encoding of the output generated by Oracle RMAN cannot be displayed by the NetVault binary logs.

Resolution

To resolve this issue, change the NLS_LANG setting to a UTF-8 character set.

For example, if you have set NLS_LANG to JAPANESE_JAPAN.JA16SJISTILDE, and the NetVault binary logs fail to display the output correctly, change the NLS_LANG setting to JAPANESE_JAPAN.UTF-8. For more information, see the **NLS_LANG** option in Adding a database.

Glossary

A

Automatic Storage Management (ASM)

Integrated File System and Volume Manager expressly built for Oracle database files.

B

backup piece

Backup file generated by RMAN. Backup pieces are written in a proprietary format that can only be created or restored by RMAN and are stored in a logical container called a backup set.

backup set

Backup of one or more datafiles, Control Files, SPFILE, and archived redo log files. Each backup set consists of one or more binary files called backup pieces. By default, a backup set consists of only one backup piece.

Block Media Recovery

Recovers only the damaged or corrupted blocks within a datafile using RMAN and leaves the affected datafiles online.

C

Complete Recovery

Recovers the database to the most recent time without the loss of any committed transactions. Typically, you perform Complete Recovery when media failure damages one or more datafiles or Control Files.

Cumulative Incremental Backup

Incremental Backup that backs up all the blocks changed since the most recent backup at Level 0. When recovering with Cumulative Incremental Backups, only the most recent Cumulative Incremental Backup needs to be applied.

D

Database Point-in-Time Recovery (DBPITR)

Recovers an entire database to a specified past target time, SCN, or log sequence number.

Data Guard

Distributed computing system that prevents or minimizes losses due to unplanned events such as human errors, environmental disasters, or data corruption, as well as to planned downtime due to routine maintenance tasks.

database identification (DBID)

An internal, uniquely generated number that differentiates databases. Oracle creates this number automatically when you create the database. The DBID is needed during disaster recovery scenarios.

Differential Incremental Backup

Incremental Backup that backs up all blocks that have changed since the most recent backup at Level 1 or Level 0. Differential Backups are the default type of Incremental Backup. When recovering using Differential Incremental Backups, the plug-in must apply all Differential Incremental Level 1 Backups since the restored tablespace backup.

Duplicate Database

Database created from target database backups using the RMAN duplicate command.

F

Flashback Database

Return of the whole database to a prior consistent SCN. A database flashback is different from traditional media recovery because it does not involve the restore of physical files, instead restoring your current datafiles to past states using saved images of changed data blocks.

Flash Recovery Area (FRA)

Optional disk location that you can use to store recovery-related files such as Control File and online redo log copies, archived redo log files, flashback logs, and RMAN backups. Oracle and RMAN manage the files in the FRA automatically.

I

instance

Combination of background processes and memory buffers.

L

log sequence number (LSN)

Number that uniquely identifies a set of redo records in a redo log file. When Oracle fills one online redo log file and switches to a different one, Oracle automatically assigns the new file a log sequence number.

O

Oracle Clusterware

Portable cluster-management solution that is integrated with the Oracle database.

P

parallelism

Allocating multiple channels for simultaneous RMAN backup and recovery operations. Allows a single job to read or write multiple backup sets in parallel. The number of channels should be less than or equal (<=) to the number of drives for the backup devices.

R

Real Application Clusters (RAC)

Support for the transparent deployment of a single database across a cluster of servers, providing fault tolerance from hardware failures or planned outages.

recovery

Recovering a datafile is taking a restored copy of the datafile and applying to it changes recorded in the database's archived and online redo logs. Term is frequently used to encompass both the restore and recovery processes.

Recovery Manager (RMAN)

Preferred utility for backup and recovery of Oracle databases. RMAN backups provide maximum reliability while providing the most flexibility when defining a backup strategy. Support for a range of backup types and options available with Oracle's RMAN tool are provided. Also included is the ability to perform a number of recovery scenarios with less reliance on human interaction and DBA expertise.

RMAN also reduces the administrative work associated with your backup strategy by maintaining an extensive record of metadata about backups, archive logs, and its own activities. This information is known as the RMAN repository. In restore operations, RMAN uses this information to eliminate the need to identify backup files for use in restores in most situations.

resetlogs

Method for opening a database, which is required after incomplete recovery or recovery with a backup Control File. OPEN RESETLOGS resets the log sequence number to 1, and clears the online redo logs.

Backups from before the OPEN RESETLOGS operation remain valid and can be used along with backups taken after the OPEN RESETLOGS operation to repair any damage to the database.

restore

Retrieve a datafile or Control File onto disk from a backup location on tape, disk, or other media, and make it available to the database server.

S

system change number (SCN)

Stamp that defines a committed version of a database at a specific point. Oracle assigns every committed transaction a unique SCN.

System Global Area (SGA)

Memory region within main memory used to store data for fast access.

U

User Managed backup method

Files that make up the database are fundamentally backed up and restored with a mixture of host OS commands and SQL*Plus backup- and recovery-related capabilities.

About us

Quest provides software solutions for the rapidly-changing world of enterprise IT. We help simplify the challenges caused by data explosion, cloud expansion, hybrid datacenters, security threats, and regulatory requirements. We are a global provider to 130,000 companies across 100 countries, including 95% of the Fortune 500 and 90% of the Global 1000. Since 1987, we have built a portfolio of solutions that now includes database management, data protection, identity and access management, Microsoft platform management, and unified endpoint management. With Quest, organizations spend less time on IT administration and more time on business innovation. For more information, visit www.quest.com.

Technical support resources

Technical support is available to Quest customers with a valid maintenance contract and customers who have trial versions. You can access the Quest Support Portal at https://support.quest.com.

*   The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

*   Submit and manage a Service Request.

*   View Knowledge Base articles.

*   Sign up for product notifications.

*   Download software and technical documentation.

*   View how-to-videos.

*   Engage in community discussions.

*   Chat with support engineers online.

*   View services to assist you with your product.