IBM System Storage N series

SnapManager 3.2 for Oracle Installation and Administration Guide for Windows

GA32-2209-00

Contents

Preface

Supported features

IBM System Storage N series storage systems are driven by NetApp Data ONTAP software. Some features described in the product software documentation are neither offered nor supported by IBM. Please contact your local IBM representative or reseller for further details.

Information about supported features can also be found on the N series support website (accessed and navigated as described in *Websites* on page 13).

Websites

IBM maintains pages on the World Wide Web where you can get the latest technical information and download device drivers and updates. The following web pages provide N series information:

- A listing of currently available N series products and features can be found at the following web page:
  www.ibm.com/storage/nas/
- The IBM System Storage N series support website requires users to register in order to obtain access to N series support content on the web. To understand how the N series support web content is organized and navigated, and to access the N series support website, refer to the following publicly accessible web page:
  www.ibm.com/storage/support/nseries/
  This web page also provides links to AutoSupport information as well as other important N series product resources.
- IBM System Storage N series products attach to a variety of servers and operating systems. To determine the latest supported attachments, go to the IBM N series interoperability matrix at the following web page:
  www.ibm.com/systems/storage/network/interophome.html
- For the latest N series hardware product documentation, including planning, installation and setup, and hardware monitoring, service and diagnostics, see the IBM N series Information Center at the following web page:
  publib.boulder.ibm.com/infocenter/nasinfo/nseries/index.jsp

Getting information, help, and service

If you need help, service, or technical assistance or just want more information about IBM products, you will find a wide variety of sources available from IBM to assist you. This section contains

information about where to go for additional information about IBM and IBM products, what to do if you experience a problem with your IBM N series product, and whom to call for service, if it is necessary.

Before you call

Before you call, make sure you have taken these steps to try to solve the problem yourself:

- Check all cables to make sure they are connected.
- Check the power switches to make sure the system is turned on.
- Use the troubleshooting information in your system documentation and use the diagnostic tools that come with your system.
- Refer to the N series support website (accessed and navigated as described in *Websites* on page 13) for information on known problems and limitations.

Using the documentation

The latest versions of N series software documentation, including Data ONTAP and other software products, are available on the N series support website (accessed and navigated as described in *Websites* on page 13).

Current N series hardware product documentation is shipped with your hardware product in printed documents or as PDF files on a documentation CD. For the latest N series hardware product documentation PDFs, go to the N series support website.

Hardware documentation, including planning, installation and setup, and hardware monitoring, service, and diagnostics, is also provided in an IBM N series Information Center at the following web page:

publib.boulder.ibm.com/infocenter/nasinfo/nseries/index.jsp

Hardware service and support

You can receive hardware service through IBM Integrated Technology Services. Visit the following web page for support telephone numbers:

www.ibm.com/planetwide/

Firmware updates

IBM N series product firmware is embedded in Data ONTAP. As with all devices, ensure that you run the latest level of firmware. Any firmware updates are posted to the N series support website (accessed and navigated as described in *Websites* on page 13).

Note: If you do not see new firmware updates on the N series support website, you are running the latest level of firmware.

Verify that the latest level of firmware is installed on your machine before contacting IBM for technical support.

How to send your comments

Your feedback helps us to provide the most accurate and high-quality information. If you have comments or suggestions for improving this document, please send them by e-mail to *starpubs@us.ibm.com*.

Be sure to include the following:

- Exact publication title
- Publication form number (for example, GC26-1234-02)
- Page, table, or illustration numbers
- A detailed description of any information that should be changed

Introduction to SnapManager for Oracle

SnapManager empowers database administrators with the tools necessary to perform policy-driven data management, schedule and create regular database backups with minimal impact, restore data from these backups in the event of data loss or disaster, and create database clones for non-disruptive testing. With SnapManager, you can create backups on primary storage and create protected backups on secondary storage.

SnapManager leverages technologies while integrating with the latest database releases. SnapManager is integrated with the following applications and technologies:

- Protection Manager leverages resource pools, datasets, and protection policies to provide policy-based automation for SnapVault and SnapMirror capabilities.
- Operations Manager provides role-based access control of storage features for enhanced security.
- SnapDrive automates storage provisioning tasks and simplifies the process of creating error-free, host-consistent Snapshot copies of the storage.
- Snapshot (a feature of Data ONTAP) creates point-in-time copies of the database.
- SnapVault (a licensed feature of Data ONTAP) leverages disk-based backups for reliable, low-overhead backup and recovery of databases.
- SnapMirror (a licensed feature of Data ONTAP) replicates database data across a global network at high speeds in a simple, reliable, and cost-effective manner.
- SnapRestore (a licensed feature of Data ONTAP) recovers an entire database in seconds, regardless of capacity or number of files.
- FlexClone (a licensed feature of Data ONTAP) helps to create fast, space-efficient clones of databases from the Snapshot backups.

SnapManager operates across SAN (FC and iSCSI) protocols.

SnapManager also integrates with native Oracle technology, such as Oracle Real Application Clusters (RAC), Oracle Recovery Manager (RMAN), Oracle Automatic Storage Management (ASM), and Oracle Direct NFS (DNFS).

SnapManager for Oracle Overview

SnapManager for Oracle simplifies and automates database backup, recovery, and cloning by leveraging the Snapshot, SnapRestore, and FlexClone technologies.

SnapManager provides the following benefits to database administrators (DBAs), who can perform these tasks:

- Database profiles
    - Organize and retain host and database information in profiles.

When DBAs initiate a backup based on a profile, they can reuse the information rather than having to reenter it for every backup. SnapManager also lets DBAs monitor operations quickly using profiles.

- In the profile, define the Snapshot copy naming patterns and enter custom (prefix or suffix) text, so that all the Snapshot copies can use the same naming convention that meets business policies.
- Map database files to the associated storage automatically, so DBAs no longer need to know the underlying storage system name.
- From SnapManager 3.2 for Oracle, you can specify in the profile, to separate the archive log backup from the datafile backup.

  Also you can update the existing profile to separate the archive log backup from the datafile backup. Once you have chosen to separate the archive log backup, you cannot take the backup of datafiles and archive logs together.
- You cannot take a full backup comprising all the datafiles and archive log files together.
- You can specify the option to separate the archive log backups while creating the profile, as an optional parameter.

- Database backups
  - Backups of full and partial databases
    - Create a full or partial backup quickly in a space-efficient way, which lets administrators perform backups more frequently.

      The full database backup contains all the datafiles, control files, and archive log files in a single backup.

      The partial database backup contains specified datafiles or tablespaces, all the control files, and all the archive log files.
    - Protect backups to secondary storage using SnapVault and SnapMirror via integration with the N series Management Console data protection capability to yield an additional layer of data protection.
    - Schedule backups on an hourly, weekly, daily, monthly, or unlimited basis.
  - Backups of datafiles and archive log files separately
    - From SnapManager 3.2 for Oracle, you can backup the datafiles and archive log files separately. To perform this, you have to specify the option to separate the archive log files while creating or updating the profile.
    - SnapManager 3.2 for Oracle provides a way to take minimum number of datafile backup and frequent archive log backups.
    - You can specify the count and duration for the datafiles backup to be retained in the retention policy.
    - You can specify the duration for the archive log file backups to be retained in archive log retention duration.
    - You can specify datafiles protection policy for the datafiles backups based on which SnapManager protects the datafiles backups.

    - You can specify archive log protection policy for the archive log backups based on which the SnapManager protects the archive log backups.
    - SnapManager 3.2 for Oracle also consolidates the archive log backups to have minimum number of backups by freeing the archive log backups with duplicate archive log files and retaining only the archive log backups with unique archive log files.

      Though this consolidation can be optionally disabled by the user.
- Archive log management
    - From SnapManager 3.2 for Oracle, you can prune the archive log files from the archive log destinations.

      Though the space occupied by the pruned archive log files will be freed when the archive log backups containing these archive log files are purged.
    - SnapManager ensures the archive log files are backed up before pruning them from the archive log destinations. The archive log files which are not backed up will not be pruned.
    - SnapManager ensures the archive log files are shipped to Dataguard Standby while pruning archive log files from a Dataguard Primary.
    - SnapManager ensures the archive log files are captured by Oracle Streams captures process, if any.
    - Recommendations
        - To effectively manage archive log destination space, create the archive log backups, and prune the archive log files along with it.
        - When the SnapManager is integrated with the Protection Manager, as soon as the backup is created, protected, and deleted or freed, the space utilized by archive log files in the archive log destination is freed.
    - SnapManager consolidates the archive log backups to have minimum number of backups by freeing the archive log backups with duplicate archive log files and retaining only the archive log backups with unique archive log files.

      Though this consolidation can be optionally disabled by the user. The archive log backups which contain duplicate archive log files will be freed and a single backup with unique archive logs will be retained.
- Database restore operations
    - Perform file-based restore operations or volume-based fast restore operations. DBAs can also preview restore operations and obtain a file-by-file analysis of restore operations before they perform the operation.
    - Reduce the mean time to restore a database by using SnapRestore.
    - Until SnapManager 3.1 for Oracle, SnapManager recovers the database only if all the archive log files are available in the archive log destination. The DBAs manually mount the archive log backups and use them for recovery.
    - SnapManager 3.2 for Oracle recovers the database automatically using the archive log files from the backup even if the archive log files are not available in the archive log destination.

      SnapManager 3.2 for Oracle also provides a way to recover the database using the archive log files from the external location to a certain extent.

- Database cloning for testing and development
  - Create a clone of a database so that the database can be set up outside of the production environment, for example, in development and test environments for testing upgrades to vital systems.
  - Clone a database on a primary or secondary storage.
  - From SnapManager 3.2 for Oracle, you can clone the datafiles backup with archive log files available in the backup.
    - You can clone the datafiles backup only when the archive log backup is taken along with it.
    - You can also clone the datafiles backup if the archive log files are available in the archive log backups taken separately to a certain extent.
    - You can also clone the datafiles backup of a standalone database to a certain extent with archive log files from any external location accessible by Oracle.
    - If you do not have the archive log in the backups but available from an external location, you can specify the external location during cloning for recovering the cloned database to a consistent state.
  - Cloning of the archive logs-only backups is not supported.
- General
  - Integrate with Operations Manager to maintain security by controlling access to SnapManager features through the use of role-based access control (RBAC).
  - Integrate with existing Oracle tools, such as Recovery Manager (RMAN) and Automatic Storage Management (ASM).
  - Work with Oracle products, which enable DBAs to continue using their current tool sets.

SnapManager provides the following benefits to storage administrators:

- Supports different SAN and NAS protocols (FCP, iSCSI, and NFS).
- Creates backups on secondary (remote) storage using the N series Management Console data protection capability protection policies.
- Lets administrators optimize backups based on the type of backup (full or partial) that works best in their environments.
- Creates space-efficient database backups.
- Creates space-efficient clones.
- Works with host volume managers.

SnapManager also works with the following Oracle features:

- SnapManager provides an integration point with Automatic Storage Management (ASM).
- SnapManager can catalog its backups with Oracle's Recovery Manager (RMAN). If using RMAN, a DBA can make use of SnapManager backups and preserve the value of all RMAN functions, such as block-level restore. SnapManager lets RMAN use the Snapshot copies when it performs recovery or restore. For example, you can use RMAN to restore a table within a tablespace as well as to perform full database and tablespace restores and recoveries from

Snapshot copies made by SnapManager. (The RMAN recovery catalog should not be in the database that is being backed up.)

- SnapManager integrates with Real Application Clusters (RAC). Invoke SnapManager from a RAC database node to create a backup, restore and recover the database, and clone the database.

Integration with other products

SnapManager for Oracle is a standalone product that integrates features from other products to produce fast backups that require only a small amount of space.

SnapManager works with these products:

Products	Explanation
SnapDrive	SnapManager uses SnapDrive to create Snapshot copies of the storage. Using Snapshot copies ensures that backups are space-efficient and faster to create than disk-to-disk backups.
The N series Management Console data protection capability	SnapManager integrates with the N series Management Console data protection capability to protect your database backups to a secondary storage system based on protection policies and to enable the use of datasets. The N series Management Console data protection capability uses SnapVault and SnapMirror to protect the data. The N series Management Console data protection capability is required if you plan to use backup protection to secondary storage. SnapVault or SnapMirror should be on primary and secondary storage systems based on the protection policies used.
Operations Manager	SnapManager integrates role-based access control (RBAC) with Operations Manager. Operations Manager is required for RBAC. It is also required along with the N series Management Console data protection capability for data protection.
FlexClone (a licensed feature of Data ONTAP)	SnapManager uses the FlexClone feature to create fast, space-efficient clones of backups. With FlexClone, you can accomplish these tasks: • Mount backups of NFS databases • Verify backups of NFS databases • Register backups of NFS databases with RMAN (if using RMAN) • Clone NFS databases. SnapManager leverages FlexClone technology to create clones in both NAS and SAN environments.
Snapshot (a feature of Data ONTAP)	Snapshot technology creates point-in-time copies of the database.

Products	Explanation
SnapRestore (a licensed feature of Data ONTAP)	SnapManager reduces the mean time to recover a database by using SnapRestore. SnapRestore software can recover individual files to a multi-terabyte volume so that operations can resume quickly.
SnapVault (a licensed feature of Data ONTAP)	SnapVault leverages disk-based backups for reliable, low-overhead backup and recovery of databases.

Advantages of using SnapManager

SnapManager for Oracle has several advantages for managing data and databases over other products.

SnapManager for Oracle works with storage systems running Data ONTAP and enables you to perform the following tasks, each further described:

- Create space-efficient backups to primary or secondary storage and schedule backups to occur on a regular basis.

  You can create full and partial database backups, apply retention duration, and protection policies to the backups. From SnapManager 3.2 for Oracle, you can create the datafiles-only backups and archive logs-only backups.

  - From SnapManager 3.2 for Oracle, you can protect the backups immediately to the secondary storage when SnapManager is integrated with the Protection Manager.
  - From SnapManager 3.2 for Oracle, you can perform a pre-processing activity or post-processing activity to occur before or after the backup and restore operations.
  - From SnapManager 3.2 for Oracle, you can vault or mirror the backups using the post-processing scripts.
- Restore full or partial databases using file-based or volume-based restore operations and preview restore operations before they occur from primary or secondary storage.
- Automatic restore and recovery of database backups.

  From SnapManager 3.2 for Oracle, you can restore as well as recover the database backups automatically. SnapManager 3.2 for Oracle automatically recovers the restored database by discovering, mounting, and applying the archive log files from the backups.
- Prune archive log files from the archive log destinations while creating the archive logs-only backups.
- Automatically retain minimum number of archive log backups by retaining only the backups with unique archive log files.
- Track operation details and produce reports by host, profile, backup, or clone.
- Verify backup status.
- Maintain history of the SnapManager operations associated with a profile.
- Create space-efficient clones of backups on primary or secondary storage.

For example, you can use the clone for testing updates in non-production environments.

Create backups using Snapshot copies

Because the SnapManager local backup is a virtual copy of the database and is stored on the same physical medium as the database, backups using Snapshot copies take much less time to create and require significantly less space than full, disk-to-disk backups. With SnapManager, DBAs can create backups on primary (local) storage and also on secondary (remote) storage using storage administrator-governed protection policies.

DBAs can create backups of an entire database, one or more tablespaces, or one or more data files, and archive log files. Control files can be backed up along with data files. The database files (data files, archive log files, and control files) can be spread across different storage systems, storage system volumes, or LUNs (logical unit numbers). DBAs can also use SnapManager to back up a database when there are multiple databases on the same volume or LUN.

SnapManager for Oracle enables you to create the following backups:

- The full backup comprising all the data files, archive log files, and control files.
- The partial backup comprising selected data files, or tablespaces, all the archive log files, and the control files.

Starting from SnapManager 3.2 for Oracle, you can optionally create the following backups:

- The datafiles-only backup with all the data files along with the control files.
- The partial datafiles-only backup with the selected data files or tablespaces along with the control files.
- The archive logs-only backup.

Prune archive log files

SnapManager for Oracle enables you to delete the archive log files from the active file system that are already backed up.

Pruning enables SnapManager to take backup of distinct archive log files. Pruning along with the backup retention policy frees the archive log space when the backups are purged.

Archive log consolidation

SnapManager 3.2 for Oracle consolidates the archive log backups to maintain a minimum number of backups for archive log files. SnapManager for Oracle identifies and frees the backups containing archive logs files that are subset of another backup.

Recover the full database or a portion of the database

SnapManager provides the flexibility to restore a full database, specific tablespaces, specific files, control files, or a combination of these entities. SnapManager offers two methods of restoring data: a

file-based restore process and a faster, volume-based restore process. DBAs can choose one or the other or let SnapManager decide which process is appropriate.

DBAs can preview a restore operation before completing it. This gives DBAs the opportunity to view on a file-by-file basis which restore methods apply to which files.

DBAs can specify the level to which SnapManager restores and recovers information when performing the restore operation. For example, DBAs can restore and recover data to a point in time. This point in time can target a date and time or an Oracle System Change Number (SCN).

Additionally, DBAs can use SnapManager to restore the database and use another tool to recover the information; administrators are not restricted to using SnapManager for both operations.

SnapManager 3.2 for Oracle enables you to restore and recover the database backups automatically without the intervention of DBA. You can use SnapManager to create the archive log backups and use the archive log backups for the restore and recovery of the database backups. Even if the archive log files of the backup is managed in the external archive log location, you can specify the external location of the archive logs that are necessary for the recovery of the restored database.

Verify backup status

SnapManager can confirm the integrity of the backup using standard Oracle backup verification operations.

DBAs can perform the verification as part of the backup operation, or at another time. DBAs can set the verify operation to occur during an off-peak time when the load on the host servers is less, or during a scheduled maintenance window.

Clone database backups

SnapManager uses FlexClone technology to create a writable, space-efficient clone of a database backup. The clone is separate from the actual backup, so DBAs can modify it without changing the backup source. Administrators might want to clone databases to enable testing or upgrades in nonproduction environments. DBAs can clone a database residing on primary or secondary storage.

A clone can be located on the same host as the database or on a different host.

SnapManager uses the FlexClone feature, which is part of Data ONTAP, to create a clone. The FlexClone feature enables SnapManager to use Snapshot copies of the database to avoid having to create an entire physical, disk-to-disk copy. The Snapshot copies require less time to create and take up significantly less space than a physical copy.

See the Data ONTAP documentation for more information on FlexClone.

Related information

The IBM support site - www.ibm.com/storage/support/nseries

Track details and produce reports

SnapManager reduces the level of detail DBAs need to track manually by offering methods to monitor operations from a single interface.

After administrators specify which database should be backed up, SnapManager automatically identifies the database files for the backup. DBAs do not need to worry about the underlying database, host file systems, or host volumes.

SnapManager displays information about repositories, hosts, profiles, backups, and clones. DBAs can monitor operations on specific hosts or databases. DBAs can determine whether backups are in process or scheduled to occur. Administrators can also identify which backups are protected or warrant attention.

New features added in SnapManager 3.2 for Oracle

SnapManager 3.2 for Oracle supports the following new features:

- Ability to manage archive log files and backups
  - Back up datafiles and archive log files separately.
  - Recover database backups automatically by discovering the archive log files from the backup and applying them for recovery.
  - Delete the archive log files while creating the backup.
- Capability to execute the pre-processing and post-processing task scripts before and after the backup and restore operations.
- Capability to protect the database backup immediately to secondary system after the backup is created:
  - Protecting database backups when SnapManager is integrated with Protection Manager from SnapManager CLI or GUI.
  - Protecting database backups when SnapManager is not integrated with Protection Manager but when using post-backup scripts to update the SnapMirror or SnapVault relationship.
- Capability to change the database host name and database storage system name for a profile from the SnapManager CLI.
- Ability to generate equivalent CLI commands automatically when a SnapManager operation is performed from the SnapManager GUI.
- Ability to collect and store historical information for SnapManager operations.
- Ability to take an offline backup by completely shutting down the database.
- Ability to create dump files immediately after a SnapManager operation is completed from the SnapManager CLI or GUI.
- Support for Data ONTAP Cluster-Mode, which is applicable only for Linux platform.
  Refer to SnapManager 3.2 for Oracle limitations for Data ONTAP Cluster-Mode.
- Support for Java Runtime Environment (JRE) 1.6.

- Graphical user interface support for Windows 7 platform.
- Separate 64-bit installer on Solaris for 11gR2.

Related concepts

Protecting database backups to secondary storage on page 159
Scheduling database backups on page 169
Updating storage system name and target database host name associated with a profile on page 251
Maintaining history of SnapManager operations on page 257

Related tasks

Creating database backups on page 135
Creating task specification for SnapManager operations on page 246

SnapManager for Oracle architecture overview

The SnapManager for Oracle architecture includes many components, such as the SnapManager for Oracle host, client, and repository. Other components include the primary and secondary storage systems and other N series products.

The SnapManager for Oracle architecture can include the following architectural components:

- SnapManager host
- SnapManager graphical user interface or command line interface
- SnapManager repository
- Primary storage system
- Secondary storage systems
- SnapDrive for UNIX or SnapDrive for Windows

SnapManager can be integrated with the following applications:

- Operations Manager
- Management Console
- The N series Management Console data protection capability

The following image shows the architecture of SnapManager for Oracle and related components:

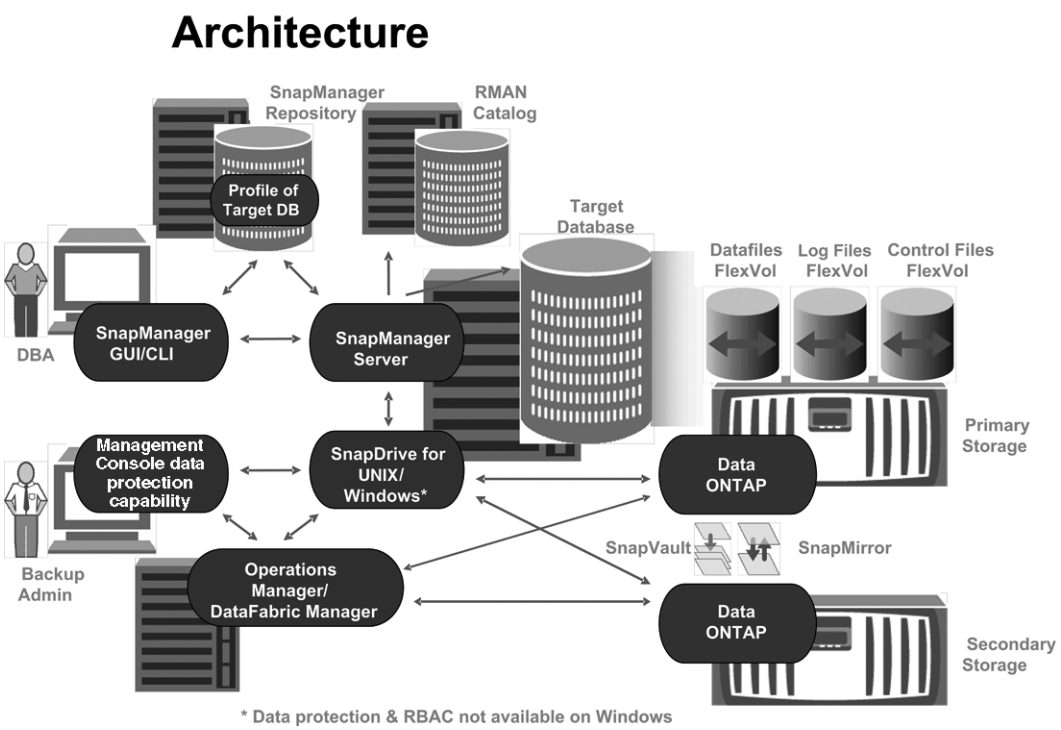

Figure 1: SnapManager for Oracle architecture

SnapManager host

The SnapManager host is a UNIX or Windows server, which also runs other products.

The SnapManager host is installed with the following products:

- SnapDrive for UNIX or SnapDrive for Windows
- Host Utilities

In UNIX, the SnapManager server operates as a daemon. On Windows, SnapManager runs as a service.

SnapManager graphical user and command-line interfaces

The SnapManager client includes both a graphical user interface (GUI) and a command-line interface (CLI).

SnapManager repository

The repository stores data about the operations performed by SnapManager, for example the time of backups, tablespaces and datafiles backed up, storage systems used, clones made, and Snapshot copies created.

Because the repository stores the names of the database Snapshot copies created during backup operations, the repository cannot exist in the same database associated with one of its profiles and

also cannot be a part of the same database that SnapManager is backing up. An administrator must create the repository in a different database than the database being backed up. This means that you must have at least two databases: the SnapManager repository database and the target database being managed by SnapManager up and running when you execute SnapManager.

Primary storage system with SnapManager

Primary storage systems receive the latest transaction updates for the Oracle database, store the data, and provide local backup protection of the database. The primary storage system also maintains database datafiles, log files, and control files.

For availing data protection, the following licenses must be enabled on primary storage systems:

- Data ONTAP 7.3.1 or later
- SnapVault Data ONTAP Primary (depending on the policy)
- SnapRestore
- SnapMirror (depending on the policy)
- FlexClone (required for NFS and cloning. Also, required for SAN if SnapDrive is configured to use FlexClone in SAN environments. Optional otherwise.)
- The appropriate protocol, for example, NFS, iSCSI, or FCP

SnapVault or SnapMirror should be on primary and secondary storage systems based on the protection policies used. The basic backup protection policies require only SnapVault installed on the supporting systems. The policies that include mirror protection require SnapMirror installed on the supporting systems. The backup and mirror disaster recovery policies require SnapMirror installed on the supporting systems.

Secondary storage system with SnapManager

Secondary storage systems act as remote storage for protected backups.

For availing data protection, the following licenses must be enabled on secondary storage systems:

- Data ONTAP 7.3.1 or later
- SnapVault Data ONTAP Secondary (depending on policy)
- SnapRestore
- SnapMirror (depending on policy)
- FlexClone (required for NFS and cloning. Also, required for SAN if SnapDrive is configured to use FlexClone in SAN environments. Optional otherwise.)
- The appropriate protocol, for example, NFS, iSCSI, or FCP

SnapVault or SnapMirror should be on primary and secondary storage systems based on the protection policies used. The basic backup protection policies require only SnapVault installed on the supporting systems. The policies that include mirror protection require SnapMirror installed on the supporting systems. The backup and mirror disaster recovery policies require SnapMirror installed on the supporting systems.

SnapDrive on SnapManager server

SnapManager uses SnapDrive for UNIX or SnapDrive for Windows to create Snapshot copies of the storage. Using Snapshot copies ensures that backups are more space-efficient and faster to create than traditional disk or tape-based backups.

SnapDrive resides on the same server as SnapManager.

Operations Manager server

The Operations Manager server provides infrastructure services (such as discovery, monitoring, role-based access control (RBAC), auditing, and logging) for various applications (for example, Performance Advisor, the N series Management Console data protection capability, and the N series Management Console provisioning capability).

The Operations Manager software runs on a separate server from the applications it supports. It does not run on the storage systems.

Operations Manager can reside on the same server as the SnapManager server; however, typically Operations Manager exists on a dedicated host.

To support local and secondary database backup protection, the Operations Manager server is licensed for the following products:

- The N series Management Console data protection capability
- The N series Management Console provisioning capability

Management Console

SnapManager integrates its data into Management Console, which is an application that displays the N series Management Console data protection capability data. SnapManager works with the N series Management Console data protection capability to protect database backups to a secondary storage system based on protection policies.

The N series Management Console data protection capability is the client platform for Java-based IBM Management Software applications. The N series Management Console data protection capability client runs on a Windows or Linux system, typically separate from the system on which Operations Manager is installed.

The N series Management Console data protection capability uses SnapVault and SnapMirror to provide data protection.

The following applications reside with the N-series Management Console client:

- The N series Management Console data protection capability client
- The N series Management Console provisioning capability client

The N series Management Console data protection capability and Operations Manager are required for data protection.

What repositories are

SnapManager organizes information into profiles, which are then associated with repositories. Profiles hold information about a database being managed, while the repository holds data about operations performed on profiles.

The repository records when a backup took place, which files were backed up, and whether a clone was created from the backup. When DBAs restore a database or recover a portion of it, SnapManager queries the repository to determine what was backed up.

Because the repository stores the names of the database Snapshot copies created during backup operations, the repository cannot exist in the same database associated with one of its profiles and also cannot be a part of the same database that SnapManager is backing up. An administrator must create the repository in a different database than the database being backed up. This means that you must have at least two databases (the SnapManager repository database and the target database being managed by SnapManager) up and running when you execute SnapManager.

You can use any valid host name, service name, or user name. For a repository to support SnapManager operations, the repository user name and service name must consist of only the following characters: alphabetic characters (A-Z), digits (0-9), minus sign ("-"), underscore ("_"), and period (".").

The repository port can be any valid port number and the repository host name can be any valid host name. In other words, the host name must consist of alphabetic characters (A-Z), digits (0-9), minus sign ("-"), and period (".), but not an underscore ("_").

The repository must be created in an Oracle database. The database that SnapManager uses should be set up in accordance with Oracle procedures for database configuration.

A single repository can hold information on multiple profiles; however, each database is normally associated with only one profile. You can have multiple repositories, where each repository holds multiple profiles.

What profiles are

SnapManager uses profiles to store the information necessary to perform operations. A profile holds the information about the database being managed, including its credentials, backups, and clones, while a repository holds data about the operations performed on the profiles. By creating a profile, you do not need to specify database details each time you perform an operation on that database. You simply supply the profile name.

A profile can reference only one database. That same database can be referenced by more than one profile. Backups created using one profile cannot be accessed from a different profile, even if both the profiles reference the same database.

Profile information is stored in a repository. The repository contains both the profile information for the database and information about the Snapshot copies that provided the backup of the database. (The actual Snapshot copies are stored on the storage system.) The Snapshot copy names are stored in the repository containing the profile for that database. When you perform an operation on a database, you must select the profile from the repository.

The following figure illustrates how repositories can hold multiple profiles, but each profile can define only one database. In this example, Repository2 is on database DB1 and Repository1 is on the database DB4.

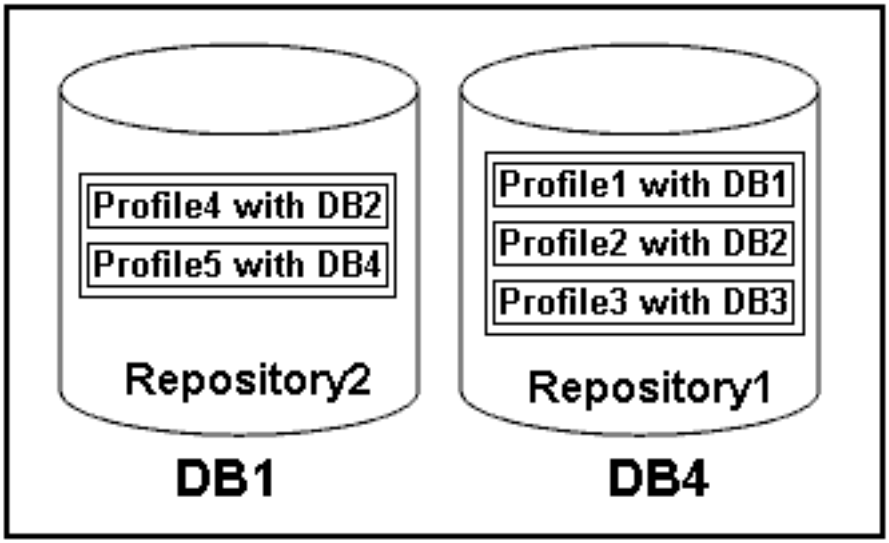

Figure 2: Multiple profiles in separate repositories

Each profile contains the credentials for the database associated with the profile. The credentials enable SnapManager to connect to and work with the database. The stored credentials include the username and password pairs for accessing the host, the repository, the database, and the required connection information if using RMAN.

You cannot access a backup created using one profile from a different profile, even if both profiles are associated with the same database. SnapManager places a lock on the database during an operation to prevent two incompatible operations from being performed simultaneously.

You can create the profiles to take full backups or partial backups. Starting from SnapManager 3.2 for Oracle, you can create profiles that enable you to take backups of the archive log files separately from the data files.

Profile for creating full and partial backups

The profiles that you specify to create the full and partial backups, contain the datafiles and archive log files together. SnapManager does not allow such profiles to separate the archive log backups from the datafile backups. The full and partial backups are retained based on the existing backup

retention policies, and protected based on the existing protection policies. The full and partial backups can be scheduled based on the time and frequency that suits you.

Profiles for creating datafiles-only backups and archive logs-only backups

Starting from SnapManager 3.2 for Oracle enables to separate archive log backups from the datafiles backup. Once you have separated the backup using the profile, you can either create the datafiles-only backups or archive logs-only backups of the database. You can also create a backup containing both the datafiles and archive log files together.

The retention policy applies to all the database backups when the archive log backups are not separated. On separating the archive log backups, SnapManager allows to specify different retention duration and protection policy for the archive log backups.

Retention policy

SnapManager determines whether a backup should be retained by considering both the retention count (for example, 15 backups) and the retention duration (for example, 10 days of daily backups). A backup expires when its age exceeds the retention duration set for its retention class and the number of backups exceed the retention count. For example, if the backup count is 15 (meaning that SnapManager has taken 15 successful backups) and the duration requirement is set for 10 days of daily backups, the five oldest successful eligible backups expire.

Archive log retention duration

When the archive log backups are separated, the archive log backups are retained based on the archive log retention duration. Archive log backups taken along with datafiles backup are always retained along with datafiles backup irrespective of the archive log retention duration.

About protected backups

All organizations have requirements that specify how frequently you back up data and how long you keep backup copies of data. SnapManager offers the ability to back up data on local storage (on the volume where the datafiles reside) or enable data protection by replicating local backups to secondary storage resources.

Backing up data to secondary storage provides these benefits to database administrators:

- Preserves the data in the case of a disaster
- Increases the limit of the number of potential backups. If you back up data only to the primary storage, the number of backups is limited by the number of Snapshot copies that can be created on a single volume.
- Provides database clones on separate storage

SnapManager also enables storage administrators to configure backups based upon carefully thought out protection policies. With SnapManager, storage administrators can quickly see backups that are not conforming to policy requirements and rectify those immediately using the N series Management Console data protection capability. SnapManager policy-based protection also provides backup consistency and policy conformance predictability.

Database administrators can perform the following tasks related to protected backups:

- Create a protected backup of an Oracle database to secondary or even tertiary storage.
- Select a protection policy, which applies data protection to backups.
- View the status of protected backups.
- Schedule backups to primary storage and protected backups to secondary storage.
- Restore data files from a protected backup.
- Clone protected backups.
- Free protected backups. You can free a protected backup on primary storage only if it has been successfully copied to secondary storage.
- Delete protected backups. You can delete a protected backup only if it has been successfully copied to secondary storage and freed from the primary storage. If the protected backup is deleted, SnapManager deletes the backup from secondary storage.

Protection policies

Protection policies are rules that govern how database backups will be protected. SnapManager retrieves available protection policies from Protection Manager and enables you to choose from a set of policies.

When backup protection is enabled, SnapManager creates a dataset for the database. A dataset is a collection of user data that SnapManager manages as a single unit, plus all the replicas of that data. The data is identified by the volume, qtree, or directory in which the dataset is located. If the administrator disables protection for a database, SnapManager deletes the dataset.

You can choose from several protection policies, such as the following:

- Back up, then mirror the data: A dataset is backed up from primary storage to secondary storage on a SnapVault or SnapMirror storage system and from there mirrored to a SnapMirror partner.
- Chain two mirrors together: A dataset is mirrored from primary storage to secondary storage on a SnapMirror partner and from there mirrored to an additional SnapMirror partner.
- Remote backup only: Data on a storage system is backed up remotely to secondary storage on a SnapVault or SnapMirror storage system. The licensed application carries out no local backup on the primary storage. This protection policy applies to third party systems with Open Systems SnapVault installed.

A protection policy specifies the intended management of dataset members. The same policy can be applied to multiple datasets, leveraging configuration of the policy across the datasets. If a policy is updated, the update is propagated across all the datasets to which the policy is applied.

A protection policy defines when data copies (used for backups) and mirror copies should be created on primary storage, when to transfer copies to secondary storage, and the maximum amount of data that should be transferred at scheduled times. The protection policy also defines how long to retain copies for each backup location and governs warning and error thresholds.

In SnapManager, you can select a protection policy for each profile in the repository if Protection Manager is installed. If Protection Manager is not installed, protection policies are not available.

Protection states

Administrators constantly monitor the state of their backups and need to know that they are consistently and successfully maintained. SnapManager shows the state of each backup.

A database backup can have the following protected states:

Status	Definition	Explanation
Protected	Protection was requested and has been enabled.	Protection has been enabled for the backup in SnapManager and Protection Manager successfully copied the backup to another set of physical disks (also referred to as "secondary storage"). If Protection Manager removes a backup from secondary storage due to a retention policy, the backup can return to a not protected state.
Not protected	Protection was requested, but not yet completed.	Protection has been enabled for the backup, but the backup is not copied to another set of physical disks. The backup has not yet been protected, protection failed, or it was once protected but is no longer protected. When you create a backup, the initial protection state of the backup is either "Not requested" or "Not protected." If it is not protected, when the backup eventually gets transferred to secondary storage, it becomes protected.
Not requested	Protection was not requested.	Protection has not been enabled for the backup. A logical copy of the data exists on the same physical disks (also referred to as a "local backup"). If protection is not requested when the backup was created, protection on the backup always shows as Not requested.

About resource pools

A resource pool is a collection of unused physical storage (such as storage systems or aggregates) from which new volumes or LUNs can be provisioned to contain data. If you assign a storage system to a resource pool, all aggregates on that storage system become available for provisioning.

Using Protection Manager, storage administrators assign a resource pool to the backup and mirror destinations of a dataset. The protection application can then automatically provision volumes out of the physical resources in the resource pool to contain backups and mirror copies.

For protected backups, SnapManager displays information about the backup and indicates whether a storage resource pool has been assigned to the backup. If not, the backup is considered "non-conformant." After a storage resource pool has been assigned, the backup changes to a "conformant" backup.

SnapManager operation states

SnapManager operations (backup, restore, and clone) can be in one of the following states, indicating the progress of the operation:

Operation state	Description
Succeeded	The operation completed successfully.
Running	The operation has started, but has not yet finished. For instance, assume a backup is scheduled to occur at 11:00 AM and the backup takes two minutes. When you view the Schedule tab at 11:01 AM, this operation appears as "Running."
No operation found	The schedule has not run or the last run backup was deleted.
Failed	The operation failed. SnapManager automatically executed the abort process and cleaned up the operation. **Note:** You can split the clone created. When you stop the clone split operation you have started and the operation is stopped successfully, the clone split operation state displays as failed.

Recoverable and unrecoverable events

A recoverable SnapManager environment issue includes the following problems:

- The database is not stored on a storage system running Data ONTAP.
- An ASM database is configured, but the ASM instance is not running.

- SnapDrive for UNIX or SnapDrive for Windows is not installed or cannot access the storage system.
- SnapManager fails to create a Snapshot copy or provision storage. This might happen if the volume is out of space, the maximum number of Snapshot copies has been reached, or an unanticipated exception occurs.

When a recoverable event happens, SnapManager performs an abort process and attempts to return the host, database, and storage system to its starting state. If the abort process fails, SnapManager treats the incident as an unrecoverable event.

An unrecoverable (or out-of-band) event occurs when any of the following happens:

- A system issue occurs, such as when a host fails.
- The SnapManager process is ended.
- An in-band abort operation fails because the storage system fails, the LUN or storage volume is offline, or the network fails.

When an out-of-band event occurs, SnapManager ends the process immediately. This means that the host, database, and storage system may not have been returned to their initial states (meaning that all cleanup actions may have not been performed). If this is the case, manually clean up after the failed SnapManager operation (for example, by deleting orphaned Snapshot copies and removing the SnapManager lock file).

SnapManager security

Security in SnapManager is governed by a combination of user authentication and role-based access control (RBAC). RBAC allows DBAs to restrict the operations that SnapManager can perform against the volumes and LUNs that hold the datafiles in a database.

DBAs enable RBAC for SnapManager using SnapDrive. They can then enable permissions to SnapManager on specific user roles and assign users to roles in the Operations Manager Web or command-line interface. RBAC permission checks occur in the DataFabric Manager server.

In addition to role-based access to SnapManager, SnapManager maintains security by requesting user authentication via password prompts on operations or by setting user credentials.

The user who is authenticated and authorized with the SnapManager server is considered the effective user.

SnapManager credentials and user authentication differ significantly from previous versions since 3.0:

- With previous versions, the person who installed SnapManager would set an arbitrary server password. All users who wanted to use the SnapManager server would need the SnapManager server password in their user credentials. They added the SnapManager server password to their user credentials with the `smo credential set -host` command.
- With SnapManager 3.0 and later versions, the SnapManager server password has been replaced with individual user operating system (OS) authentication. The SnapManager server now authenticates users with their OS user names and passwords, if those users are not running the

client from the same server as the host. If the users are running the client from the same host as the server, the SnapManager server does not have to authenticate them because they are already logged into the server host. If users want to avoid being prompted for their OS user passwords, they can save their data to their SnapManager user credentials cache with the `smo credential set -host` command. This command saves the encrypted password.

Note: The `smo credential set -host` command remembers the user's credentials when the following property in the `smo.config` file is set to true: host.credentials.persist.

Consider the case where User1 and User2 share a profile called Prof2. In this case, User2 cannot backup Database1 in Host1 without permissions to access Host1. User1 cannot clone a database to Host3 without permissions to access Host3.

Permission Type	User1	User2
Host Password	Host1, Host2	Host2, Host3
Repository Password	Repo1	Repo1
Profile Password	Prof1, Prof2	Prof2

In the case where User1 and User2 do not have any shared profiles, assume User1 has permissions for the hosts named Host1 and Host2, and User2 has permissions for the host named Host2. User2 cannot run even non-profile commands (for example, dump and system verify) on Host1.

Accessing and printing online Help

The online Help provides instructions for the tasks that you can perform using the SnapManager graphical user interface. The online Help also provides descriptions of fields on the windows and wizards.

Steps

1. Do one of the following:
   - In a main window, click **Help > Help Contents**.
   - In any window or wizard, click **Help** to display help specific to that window.
2. To navigate through the topics, use the Table of Contents in the left pane.
3. To print individual topics, click the Printer icon at the top of the help window.

SnapManager for Oracle deployment considerations

Before deploying SnapManager in your environment, you should be aware that there are important product requirements and prerequisites, in addition to protocol, volume management, and Oracle requirements.

Depending on your needs, install the following in your environment:

Requirement	Details
Data ONTAP	Data ONTAP must be installed. SnapManager takes advantage of ONTAP tools and technologies, including Snapshot copies.
SnapDrive for UNIX or SnapDrive for Windows	SnapManager uses the features of the SnapDrive products, so one of these must be installed before SnapManager can run. SnapManager handles all the interactions with the SnapDrive products. SnapDrive for UNIX or SnapDrive for Windows must be configured correctly for your storage system and protocol choices.
SnapRestore license	Must exist for each storage system.
FlexClone license	FlexClone is a licensed feature in the Data ONTAP version supported by SnapManager. SnapManager handles all of the interactions with the FlexClone feature. SnapManager works with FlexClones in both NAS and SAN environments. • A FlexClone license is required to take full advantage of SnapManager with NFS databases. • If SnapDrive for UNIX is configured to use FlexClone for SAN environments, a FlexClone license is required.
The N series Management Console data protection capability	To use the data protection feature of SnapManager, you must install the N series Management Console data protection capability. The N series Management Console data protection capability leverages resource pools, datasets, and protection policies to provide policy-based protection.

Requirement	Details
Operations Manager	To enable role-based access control (RBAC) of storage features for enhanced security, you must install Operations Manager.
FC, iSCSI, NFS protocols	The appropriate protocols must be installed and licensed.

SnapManager requirements

Before deploying SnapManager in your environment, you should be aware of its requirements and considerations.

Before installing or using SnapManager, review the compatibility matrices for all required products.

- Check the publication matrix page at *www.ibm.com/storage/support/nseries* for important alerts, news, interoperability details, and other information about the product before beginning the installation.
- For SnapDrive for UNIX requirements, see the online SnapDrive for UNIX Compatibility Matrix. For SnapDrive for Windows requirements, see the online SnapDrive for Windows Compatibility Matrix.

Note: SnapManager requires specific minimum Oracle versions on some platforms.

See the kit documentation for more information about the recommended configurations for the host and storage systems.

Note: Contact your sales representative if you need a SnapManager configuration that is not mentioned in the kit documentation.

Related information

The IBM support site - www.ibm.com/storage/support/nseries

Supported host software

While setting up the host system, consider the host environment and operating system requirements.

When preparing to install SnapManager, make sure that you follow these host requirements:

- Configure SnapManager for Oracle and SnapManager for SAP on different hosts.
  They cannot run concurrently on the same host.
- Install SnapDrive for Windows on the host platform, including the products required, such as the Host Utilities.
  Follow the instructions provided with the kit to set up the storage systems to work with the host.

To use the SnapManager graphical user interface, you must have a host running one of the following platforms. See the Compatibility Matrix for details about supported protocols for each.

- Windows Server 2008
- Windows Server 2003

SnapManager also operates in VMware ESX virtualized environment. Check the technical reports for additional information at *www.ibm.com/storage/support/nseries*.

The graphical user interface also requires that the Java runtime environment (JRE) version 1.6 be installed on the host.

Supported host hardware

Consider the memory, disk space, and CPU requirements.

SnapManager requires the following configuration:

Hardware function	Hardware requirements
Memory	The SnapManager server requires 128 MB of memory. The graphical user interfaces requires a minimum 512 MB RAM to run. Each operation run by the SnapManager server requires 48 MB of additional memory while it is running.
Disk space	128 MB available disk space for the graphical user application (minimum).
CPU speed	1.0 GHz processor speed (minimum).

Supported general configurations

Before installing SnapManager, adhere to these general configuration requirements.

- A non-clustered configuration where a single host is connected to a single storage system
- One SnapManager server instance per host
- Any of the topologies involving storage systems running Data ONTAP controller failover

Always review the SnapManager & SnapDrive Compatibility Matrix for the latest detailed information for all storage and volume manager types and versions supported by SnapManager.

Clustered configurations

SnapManager operates in cluster configurations.

SnapManager supports the same host cluster and HA pair configurations that the SnapDrive product and Host Utilities Kit support.

SnapManager also supports non-clustered configurations where a single host is connected to a single storage system, supported host clusters, and storage systems running Data ONTAP controller failover.

Database version support and configuration overview

Perform basic database layout and configuration setup to ensure successful SnapManager operation, including the correct requirements for the `oratab` file, use with RAC databases, and use with NFS.

SnapManager for Oracle integrates with Oracle versions 10gR2 (10.2.0.3, 10.2.0.4, and 10.2.0.5), 11gR1, and 11gR2 (11.2.0.1 and 11.2.0.2); with native Oracle technology (such as RAC, RMAN, ASM, and Direct NFS); and across FCP, iSCSI, and NFS protocols.

Note: Support for Oracle 9i database is deprecated in the release of SnapManager 3.2 for Oracle.

Follow these guidelines for database layout and configuration:

- General database layout and configuration
- Database home setup using the `oratab` file
- Requirements for using RAC databases with SnapManager
- Requirements for using ASM databases with SnapManager
- Requirements for using databases with NFS and SnapManager

General layout and configuration

Administrators should adhere to the recommended general database layout and storage configuration, which cites issues related to disk groups, file types, and tablespaces.

- Do not include files from more than one type of SAN file system or volume manager in your database. All files making up a database must reside on the same type of file system.
- All LUNs within a volume should reside at the volume level or reside inside qtrees, not a combination of both.
- SnapManager requires a multiple of 4K block size.
- Include the database system identifier (SID) in the `oratab` file. Include an entry in the `oratab` file for each database to be managed. SnapManager relies on the `oratab` file to determine which Oracle home to use.
- If you want to register SnapManager backups with RMAN, you must create RMAN-enabled profiles.

To leverage the new volume-based restore or full disk group restore, consider the following guidelines related to file systems and disk groups:

- Multiple databases cannot share the same ASM disk group.
- A disk group containing datafiles cannot contain other types of files.
- The LUNs for the datafile disk group must be the only object in the storage volume.

Follow these guidelines for volume separation:

- Datafiles for only one database must be in the volume.

- Use separate volumes for each of the following file classifications: database binaries, datafiles, online redo log files, archived redo log files, and control files.
- You do not need to create a separate volume for temporary database files, because SnapManager does not back up temporary database files.

Related information

Technical Report 3761 - SnapManager for Oracle Best Practices

Defining the database home with the oratab file

SnapManager uses the `oratab` file to determine the Oracle database home directory during operations. An entry for your Oracle database must be in the `oratab` file for SnapManager to work correctly. The `oratab` file is created during the Oracle software installation.

About this task

The `oratab` file resides in different locations based on the host operating system:

Host operating system	`oratab` file location
Linux	`/etc/oratab`
Windows Server 2003	N/A (Home information stored in Windows registry)
Solaris	`/var/opt/oracle/oratab`
HP-UX	`/etc/oratab`
IBM AIX	`/etc/oratab`

Steps

1. After Oracle is installed, ensure that the `oratab` file resides in the location specified in the table above.
2. If the `oratab` file does not reside in the correct location per your operating system, contact Technical Support for assistance.

Requirements for using RAC databases with SnapManager

Use the following recommendations for using RAC databases with SnapManager. The recommendations include port numbers, passwords, and authentication connection mode.

- For a database-authentication connection mode, the listener on each node that services an instance of the RAC database must be configured to use the same port number. Also, the listener that services the primary database instance must be started prior to initiating a backup.

- For an OS-authenticated connection mode or ASM environment, the SnapManager server must be installed and running on each node in the RAC environment.
- The password of the database user that SnapManager uses (typically `sys` or a user with `sysdba` privilege) must be the same for all the Oracle instances in a RAC environment.

Requirements for using ASM databases with SnapManager

Use the following recommendations for using ASM databases with SnapManager for Oracle. The recommendations include issues with the ASMLib, partitions, and clone specifications.

- SnapManager 3.0.3 uses the new SYSASM system privilege available with Oracle 11gR2 instead of SYSDBA to administer an Oracle ASM instance. If you use the SYSDBA privilege to run administrative commands on an Oracle ASM instance, the operation results in an error. The SYSDBA privilege is intended to be used by the database to access disk groups. Connecting to an Oracle ASM instance as SYSASM grants you full access to all of the available Oracle ASM disk groups and management functions.

  Note: 10gR2 and 11gR1 continue to use SYSDBA.

- SnapManager 3.0.3 supports backing up files that are stored directly on ASM disk groups when the disk group also contains an ACFS volume. SnapManager 3.0.3 does not support backing up files in an Automatic Storage Management Cluster File System (Oracle ACFS).

  Note: Oracle ACFS is a multi-platform, scalable file system, storage management technology available with Oracle 11gR2. Oracle ACFS extends Oracle ASM functionality to support customer files maintained outside of the Oracle database.

- SnapManager 3.0.3 supports backing up and restoring files that are stored directly on ASM disk groups when the disk group also contains Oracle cluster registry (OCR) files or voting disk files. Oracle strictly recommends having OCR and voting disks on diskgroups that do not contain database files.
- As recommended by Oracle, exactly one partition is required for each disk that contains the entire disk.
- SnapManager 3.0.2 for Oracle and later versions supports ASM on raw disks on the Red Hat Enterprise Linux and SUSE Linux Enterprise servers. You can upgrade the server from Red Hat Enterprise Linux 4 Update X to Red Hat Enterprise Linux 5 Update X over nonpartitioned devices.
- SnapManager 3.0.2 for Oracle or later versions supports ASM diskgroups with partitioned devices on Red Hat Enterprise Linux 5 Update X or later versions. However, any existing deployments on partition devices with Red Hat Enterprise Linux 4 Update X will be supported.

  Note: SnapManager does not support partitioned devices with SUSE Linux Enterprise Server 10 SP2 for ASM.

- The start of the ASM disk must be aligned to a 4K WAFL file segment. This implies that the device partition on which the ASM disk is created must be 4K-aligned relative to the device itself and that the "multiprotocol type" for the LUN must be set accurately for the operating system.

Note: For details about how to create partitions that are aligned to 4K, refer to the Knowledge Base article 8190.

- When laying out a database, follow the recommendations in the technical report TR 3329. This report provide information on how to lay out the LUNs for an ASM disk group.
- ASM configuration is not a required part of the clone specification. You must manually remove the ASM configuration information in clone specifications created using the SnapManager 2.1 for Oracle version before upgrading the host to SnapManager 2.2 for Oracle and later version.
- SnapManager 3.1 for Oracle, a patch version (3.1p1) of SnapManager 3.1 for Oracle, and SnapManager 3.2 for Oracle support ASMLib 2.1.4.
- SnapManager 3.2 for Oracle and a patch version (3.1p4) of SnapManager 3.1 for Oracle support ASMLib 2.1.4 and 2.1.7

Related information

Technical Report 3329 - Using Oracle Database 10g Automatic Storage Management
Technical Report 3761 - SnapManager for Oracle Best Practices

Requirements for using databases with NFS and SnapManager

Use the following recommendations for using databases with NFS and SnapManager for Oracle. The recommendations include running as root, attribute caching, and symbolic links.

- SnapManager runs as root and must be able to access the file systems containing data files, control files, online redo logs, archive log, and the database home. To ensure that "root" can access the file systems, either of the following NFS export options must be set:
  - "root=<hostname>", or
  - "rw=<hostname>,anon=0"
- Disable attribute caching for all the volumes that contain database data files, control files, redo and archive logs, and the database home. Export these with the `noac` (for Solaris, AIX, HP-UX) or `actimeo=0` (for Linux) option.
- If database data files are linked from local storage to NFS, SnapManager supports symbolic links at the mountpoint level only.

General restrictions

SnapManager for Oracle has certain restrictions that might affect your environment.

- SnapManager supports control files on a file system or in ASM and does not support control files on raw devices.
- SnapManager supports databases on MultiStore storage systems with the following requirements:
  - You must configure SnapDrive to set passwords for MultiStore storage systems as is the current requirement with other storage systems.

    - The SnapDrive product cannot create a Snapshot copy of a LUN or file residing in a qtree in a MultiStore storage system if the underlying volume is not in the same MultiStore storage system.
- SnapDrive and SnapManager do not support reverting to a previous version.
- SnapManager can be configured to catalog database backups with RMAN. If an RMAN recovery catalog is being used, the recovery catalog must be in a different database than the database that is being backed up.
- The SnapDrive for UNIX product supports more than one type of file system and/or volume manager on certain platforms. The file system and/or volume manager used for database files must be specified in the SnapDrive configuration file as the default file system/volume manager. See the *SnapDrive for UNIX Installation and Administration Guide* for more information about using this file.
- SnapManager for Oracle does not support accessing two SnapManager for Oracle servers running on different ports from a single client (both for the graphical user interface or when using commands). The default port number must be the same for all port numbers involved in remote operations.
- If a user tries to clone a database from secondary storage and a Snapshot copy in the backup happens to be the last Snapshot copy transferred to secondary storage for that qtree or volume, then the clone will fail. An error message appears describing why it failed. Protection Manager controls the secondary transfer schedules. In this case, you might want to take another backup and wait for it to be transferred to secondary storage by Protection Manager during its regular transfer schedule. Alternatively, you might want to contact the storage administrator and ask for the backup to be transferred.
- All LUNs within a volume should reside at the volume level or inside qtrees, but not in both. This is because if the data is residing on the qtrees and the user mounts the volume, then the data inside the qtrees is not protected.
- SnapManager operates within a Microsoft Clustering (MSCS) environment. However, SnapManager does not recognize an MSCS active/passive configuration and will not transfer active management of a repository to a standby server in an MSCS cluster.
- Backup creation might fail if SnapManager operations are run concurrently on the same host against a different ASM database.
- With regard to restoring the database backups from secondary storage system, in the `smo.config` file, the `restore.secondaryAccessPolicy` option can be set only to direct. The `restore.secondaryAccessPolicy` cannot be set to `-indirect` until the release of Protection Manager for use with DataFabric Manager 3.8 is available.
- SnapManager fails to delete a clone after the host name is changed for the host on which the clone is running. To fix this issue, rename the old host name to the same IP address in the client and server hosts file, and try to delete the clone again.
- The clone split progress reported in terms of percentage is calculated based on the number of inodes processed, divided by the total number of inodes present for a given flexible volume. It has been generally observed that the users cannot view any numerical values between 0 and 100 due to the speed with which the inodes are discovered and processed by the storage system containing the flexible volume.

- SnapManager enables you to receive e-mails only for the failed clone split operations and not for the successful clone split operations.
- If you try to install the previous version of SnapManager for a host, ensure that you rollback the repository for that host. If you install the previous version of SnapManager, without performing the rollback of the host in that repository, you might encounter the following issues:
  - Could not view the profiles created in the previous or later version of SnapManager for that host.
  - Could not access backups or clones created in the previous or later version of SnapManager.
  - Can create new profiles and perform database operations using the new profiles, but could not perform rolling upgrade or roll back of that host without deleting the new profiles.
- SnapManager for Oracle does not support ASM and RAC configurations on all supported Windows platforms (Windows 2003 and Windows 2008).
- The supported hardware platforms for Windows OS are 32-bit and 64-bit (Windows x86 and Windows x86_64). IA-64 is not a supported hardware platform.
- SnapManager for Oracle does not support splitting a LUN clone on the Data ONTAP 7-Mode and Cluster-Mode.
- Pruning of archive log files from the flash recovery area destination is not supported.
- Pruning of archive log files from the standby database is not supported.
- The archive log backups are retained based on the retention duration and default hourly retention class. When the archive log backup retention class is modified from default value using the sm`backup update` command, the modified retention class is not considered for the backup as the archive log backups are retained based on the retention duration.
- Once you have separated the profiles to create archive log backups, user you cannot rollback the related host repository to the earlier version of the SnapManager.
- When an indirect method is used for restore, SnapManager cannot mount the backups from the secondary storage system. When SnapManager restores the database from the secondary backup, the recovery fails if the archive log files are available only in the secondary storage system as they cannot be mounted and used for recovery.
- SnapManager for Oracle cannot clone, restore, or mount the last protected backup that is freed from the primary storage system. As a workaround, create another protected backup, and mount or clone, or restore the backup from the previous backup.
- History configuration is a part of the profile creation process in the SnapManager GUI where as it is provided as a separate command from the SnapManager CLI. The equivalent SnapManager CLI commands generated for the profile create operation from the SnapManager GUI does not have history configuration options. You cannot use this command to configure history retention settings from the SnapManager CLI.
- The cloning of online database backup of the RAC database using the external archive log file location fails due to failure in recovery. This is due to an Oracle issue as Oracle fails to find and apply the archive log files for recovery from the external archive log location while cloning the database backup.
- SnapManager does not support using the task specification file for the scheduled backups that are created prior to SnapManager 3.2 for Oracle. You can start using the task specification file for the scheduled backup from SnapManager 3.2 for Oracle.

- If the backup is already mounted, SnapManager for Oracle does not mount the backup again and uses the already mounted backup.

  If the backup is mounted by a different user, and if the current user does not have access to the previously mounted backup, other users have to provide the permissions.

  All the archive log files have read permissions for the groups owners; the current user might not get the permissions, if the backup is mounted by a different user group. The users can give permissions to the mounted archive log files manually and then retry the restore or recovery.
- Starting the SnapManager graphical user interface with Mozilla Firefox web browser when there is no JRE available in the Windows and UNIX client, is not supported.
- Starting the SnapManager graphical user interface with Internet Explorer 6 is not supported on Windows Server 2008 and Windows 7.
- Java Web Start (javaws) is not available for JRE 1.5 AMD 64-bit platforms. Hence, a blank screen is displayed when SnapManager for Oracle is launched.
- While creating a backup from the SnapManager GUI, SnapManager fails to load the database structure in the **Backup Create** wizard when the number of datafiles is high (more than 3000).
- When a backup or clone operation is executed simultaneously on both the 10gR2 and 11gR2 RAC databases over ASM, then any one of the backup or clone creation operation fails. This is due to a known Oracle issue.
- When you update multiple profiles containing a combination of profiles that are enabled to separate archive logs backups and profiles that are disabled to separate archive logs backup with the Multi Profile Update window from the SnapManager GUI, the **Backup Archivelogs separately** option in the Multi Profile Update window is disabled.

  User cannot update the archive log retention duration and protection policies of the archive log backups related to those profiles.
- SnapManager sets the backup state as PROTECTED even when one of the Snapshot copies of the database backup is transferred to the secondary storage system.
- While updating the target database hostname associated with the profile from the SnapManager CLI, if there are one or more SnapManager GUI sessions opened, then all the SnapManager GUI sessions that are opened do not respond.

  Kill all the SnapManager GUI sessions and then restart the SnapManager GUI session.
- When SnapManager performs volume restore, the archive log backups that are taken after the current backup being restored, will not be purged.

  The archive log Snapshot copies will be lost if the datafiles backup is restored through the volume restore mechanism even though the archive log files exist in the same volume of datafiles and the archive log files are not available in the archive log file destination.

  Always have few archive log files in the active file system for SnapManager to avoid volume restore.
- When multiple backup schedules are getting triggered related to a profile, from the multiple hosts or same host having more than one IP that are pointing to the same repository, perform the following steps:

  1. Stop the SnapManager server.

2. Delete the schedule file in the repositories directory available at the SnapManager installation directory from the host(s) where you do not want to trigger the schedule.
   The schedule file name will be in the format of: `repository#repo_username#repository_database_name#repository_host#repo_port`
   Ensure that you delete the schedule file in the exact format that matches the repository details.
   If you find the schedule file name in the format of: `repository-repo_username-repository_database_name-repository_host-repo_port`, you can delete this file too.
3. Restart the SnapManager server.
4. Open other profiles under the same repository from the SnapManager GUI to ensure that you do not miss any schedule information of those profiles.

SnapManager for Oracle features and Oracle configurations not supported on Windows platform

SnapManager for Oracle does not support the following features and platforms on the Windows platform:

- Policy-based data protection as integration with Protection Manager is not available on the Windows platform.
- RBAC as integration with Operations Manager is not available on the Windows platform.
- Fast restore or volume-based SnapRestore (VBSR).
- Split a clone.
- Oracle RAC using any protocol.
- Oracle ASM using any protocol.
- Oracle Direct NFS (dNFS).
- Itanium64 platforms.

SnapManager 3.2 for Oracle limitations for Data ONTAP Cluster-Mode

SnapManager 3.2 for Oracle has the following limitations for Data ONTAP Cluster-Mode.

- AutoSupport notification is not supported.
- Database residing on mixed mode storage, where in one LUN belonging to 7-mode storage system and another LUN belonging to Cluster-Mode storage system, is not supported.

Oracle limitations

SnapManager has the following limitations with Oracle:

- SnapManager for Oracle supports Oracle versions 10gR2, 11gR1 and 11gR2, but does not support Oracle 10gR1 as a repository or target database.
- SnapManager will not support use of a SCAN IP address in place of a hostname. SCAN IP is a new feature in Oracle 11gR2.
- SnapManager does not support Oracle Cluster File System (OCFS).
- There may be certain version and patch limitations. Verify support with the appropriate support matrix at *www.ibm.com/storage/support/nseries*.
- Oracle 11g in a Direct NFS (DNFS) environment allows additional mountpoint configuration, such as multiple paths for load balancing, in the `oranfstab` file. SnapManager does not modify this file, so any additional properties you want a clone to use must be manually added to the `oranfstab` file after cloning with SnapManager.
- Support for Oracle 9i database is deprecated in the release of SnapManager 3.2 for Oracle.

Oracle 9i database support deprecation and use of Oracle 9i database option in SnapManager 3.2 for Oracle

Oracle 9i Database is no longer supported in SnapManager 3.2 for Oracle. SnapManager 2.x, 3.0.x and 3.1.x for Oracle versions support Oracle 9i databases. SnapManager 3.2 for Oracle supports only Oracle 10gR2, 11gR1, and 11gR2 databases.

With the Oracle 9i databases, if you are upgrading to SnapManager 3.2 for Oracle, a warning message displays, and you cannot create new profiles.

What to do if you have Oracle 9i database

- Upgrade the Oracle 9i databases to either Oracle 10gR2, 11gR1, or 11gR2 databases, and then upgrade to SnapManager 3.2 for Oracle.
- Manage the Oracle 9i databases using a patch version of SnapManager 3.1 for Oracle, and use SnapManager 3.2 for Oracle to manage Oracle 10gR2, 11gR1 or 11gR2 databases.

Volume management restrictions

SnapManager for Oracle has certain volume management restrictions that may affect your environment.

You can have multiple disk groups for a database; however, the following rules apply to all disk groups for a given database.

- Disk groups for the database can be managed by only one volume manager.

- Use of raw devices backed by a logical volume manager is not supported for protection of any Oracle data. Raw device storage and ASM diskgroups must be provisioned directly on physical devices. In some cases, partitioning is required.
- A Linux environment without logical volume management (LVM) requires a partition.

Installing or upgrading SnapManager for Oracle

To install or upgrade the software, ensure you have completed the prerequisites. Then download the software and install or upgrade it.

Before you begin

Before you can install or upgrade SnapManager, the host system must be set up correctly:

- The operating system and appropriate patches must be installed and licensed.
- SnapDrive and prerequisite products, such as the Host Utilities must be installed.
  Instructions are included with the kit provided for setting up the storage systems to work with the host system.
- Oracle and any needed Oracle components, such as RMAN or ASM, must be installed and configured.
  The N series support website (accessed and navigated as described in *Websites* on page 13) lists supported Oracle versions.

Preparing to install or upgrade SnapManager for Oracle

Before installing or upgrading SnapManager for Oracle, ensure you follow the considerations section and perform the other tasks.

General considerations to install or upgrade SnapManager for Oracle

Before you can install or upgrade SnapManager, you can check for the following considerations:

- The host system must meet the requirements for SnapManager as specified in the publication matrix page at *www.ibm.com/systems/storage/network/interophome.html* for important alerts, news, interoperability details, and other information about the product.
- Each storage system must have Data ONTAP installed, a FlexClone license (if using NFS with SnapManager operations), a SnapRestore license, and the supported protocol (FC, iSCSI, or NFS) installed and licensed.
- Back up of existing SnapManager repositories must be performed.

Note: If using NFS, ensure the file systems are mounted as recommended in the *Best Practice Guidelines for Oracle* and *SAP with Oracle on UNIX and NFS and Storage*.

Note: The technical reports contain information about products that IBM licenses and in some cases customizes. Technical reports might contain information about models and features that are not supported by IBM.

Related information

The IBM support site - www.ibm.com/storage/support/nseries

Downloading SnapManager software

About this task

The SnapManager installation includes both the host server software and the graphical user interface client software for UNIX and Windows operating systems.

You can install the product software either from the physical media kit or from software updates available for download. Downloads are available only to entitled IBM N series customers who have completed the registration process on the N series support website (accessed and navigated as described in *Websites* on page 13).

Installing or upgrading SnapManager on a Windows host

You can install or upgrade SnapManager software on any approved Windows host. Install only one SnapManager server instance per host.

Steps

1. For upgrades, stop the server by completing the following:
   a. In the Windows Services window, select **Ontap SnapManager 3.1 for Oracle**.
   b. In the left panel, click **Stop**.
2. Double-click the downloaded executable file.

   For Windows x86, use `ontap.smo.windows-x86-3.1.exe`. For Windows x64, use `ontap.smo.windows-x64-3.1.exe`.
3. A dialog box might appear with this message:
   `The publisher could not be verified. Are you sure you want to run this software?`

   Click **OK**.
4. In the Introduction window, click **Next.**
5. If this is an upgrade, a pop-up window appears. To continue, click **OK**.
6. For new installations (not upgrades) at the next Choose Install Folder window, either click **Next** to accept the default location for the installation folder, or choose a new location.

   The default location is: `C:\Program Files\Ontap\SnapManager for Oracle`.
7. On the Menu Availability window, click **Next**.

8. On the Specify Service Properties window, enter the account and password for the Windows service for SnapManager.

   The specified account must be a member of the following groups:

   - The storage system's local administration group
   - The local administrator's group of this system
   - The ORA_DBA group

   Specify whether the service starts up automatically after reboot or if you must manually start the service.

   Click **Next**.

9. On the Pre-Installation Summary window, click **Install.**

10. At the Install Complete window, click **Next**.

11. At the Important Information window, click **Done** to exit the installer.

12. Start the SnapManager server by completing the following:

    a. In the Windows Services window, select **Ontap SnapManager 3.1 for Oracle**.

    b. In the left panel, click **Start**.

13. Verify that the SnapManager system is running correctly by following these steps:

    a. Open the SnapManager Command Line Interface (CLI) command prompt window by selecting **Start > Programs > Ontap > SnapManager for Oracle > Start SMO Command Line Interface (CLI)**

    b. In the CLI window, enter the following command:

       ```
       smo system verify
       ```

    SnapManager displays a message stating that the operation succeeded.

14. For upgrades, upgrade each SnapManager repository by following these steps:

    a. Open the SnapManager Command Line Interface (CLI) command prompt window by selecting **Start > Programs > Ontap > SnapManager for Oracle > Start SMO Command Line Interface (CLI)**

    b. In the CLI window, enter the following command:

       ```
       smo repository update -repository -dbname repo_service_name  -host
       repo_host
       -login -username repo_username   -port repo_port
       ```

    SnapManager displays a message stating that the operation succeeded.

Related tasks

Starting the SnapManager Windows host server on page 82

Stopping the SnapManager host server on page 215

Rolling upgrade

SnapManager provides the ability to upgrade a single or multiple SnapManager server hosts that are communicating with a SnapManager repository database in a host-by-host approach.

What a rolling upgrade is

SnapManager for Oracle versions prior to 3.1 provided the ability only to upgrade all the SnapManager server hosts together to a later version of SnapManager. This resulted in downtime of all the SnapManager server hosts and the scheduled operations, during that time. To overcome these limitations, SnapManager 3.1 for Oracle leverages the rolling upgrade approach.

Rolling upgrades enable you to upgrade one or more SnapManager server hosts that are communicating with a SnapManager repository database in a staggered, host-by-host approach with the following benefits:

- You can use new features available in SnapManager 3.1 for Oracle.
- You can work with new features in one SnapManager server host before upgrading the rest of the hosts in your migration.
- Product performance is enhanced.

Note: SnapManager supports rolling upgrade only from the SnapManager CLI and not from the GUI.

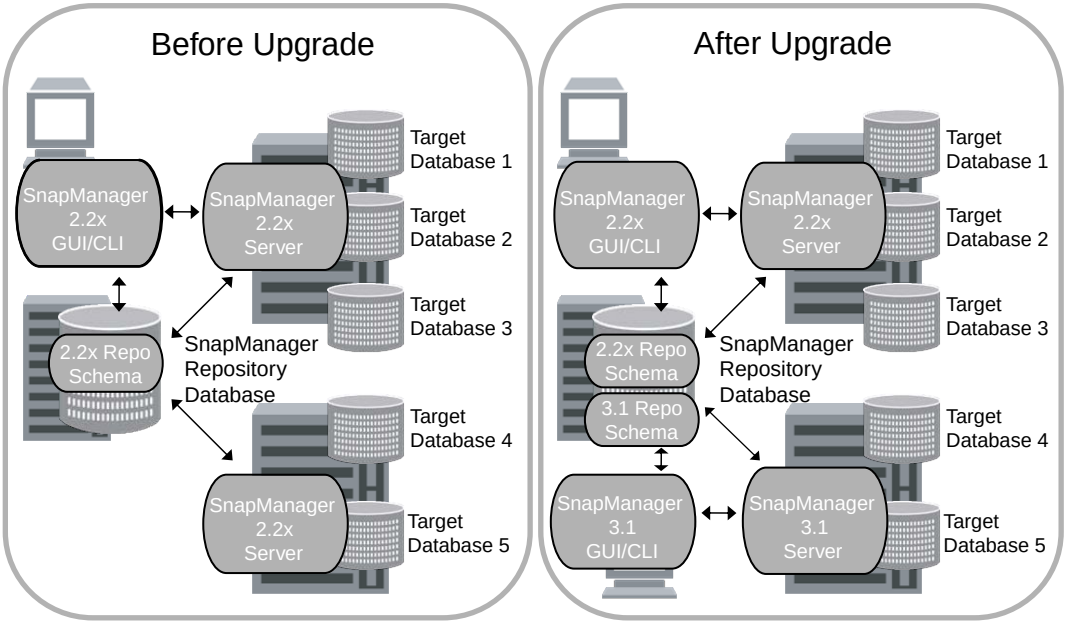

Figure 3: Scenarios before and after a rolling upgrade

After successful completion of a rolling upgrade on the SnapManager server host, the hosts, profiles, schedules, backups, and clones associated with the profiles of the target databases are migrated from the repository database of the previous SnapManager version to the repository database of the new version. The details on the operations performed using the profiles, schedules, backups, and clones that were created in the previous SnapManager version are now available in the repository database of the new version. The profile information is not available in the previous SnapManager version and you cannot view the profile information in the previous version of the SnapManager GUI or CLI.

The upgraded SnapManager for Oracle server can now communicate with the upgraded SnapManager for Oracle repository database. The hosts that were not upgraded can manage their target databases using the features available in the previous version of SnapManager and continue to be managed by the GUI or CLI of those versions of SnapManager for Oracle.

If you want to revert to the previous version of SnapManager, you can roll back to that original version, except in certain scenarios that are not supported. The profile information associated with the previous SnapManager version is not available in the GUI or CLI of the new SnapManager for Oracle version.

Scenarios where the rollback is not supported

If you successfully complete any of the following operations after upgrading the SnapManager server hosts, you cannot roll back the hosts to their original SnapManager versions, and the target databases become a permanent entity of the latest SnapManager version:

- New profile creation
- Clone split
- Protection status change of the profile
- Mount status change of the backup
- Restore

Note: If you want to roll back changing the mount status following a rolling upgrade, you must first change the mount status to its original state and then perform the roll back operation.

Rolling upgrade is supported from SnapManager 2.1.1 for Oracle or later versions only.

Note: Before a rolling upgrade or a rollback, ensure that all hosts under the repository database are resolvable. For details about how to make the system resolvable, see "Troubleshooting SnapManager for Oracle".

Considerations for performing a rolling upgrade

Before performing a rolling upgrade, you need to keep a number of considerations in mind:

- External scripts that are used to perform any external data protection or data retention must be backed up.
- The SnapManager server hosts that are being upgraded must be running SnapDrive 5.0 for UNIX or SnapDrive 6.3 for Windows.
- SnapManager 3.2 for Oracle must be installed in the SnapManager server hosts that are being upgraded.

- The repository database must be backed up before you perform the rolling upgrade.
- All database operations running on the existing version of SnapManager must be complete before you begin the rolling upgrade.

While performing a rolling upgrade, you need to keep a number of considerations in mind:

- If you perform the rolling upgrade or rollback operation on a host using a repository, ensure that you do not perform any SnapManager operations on the other SnapManager hosts under that repository.

  If there are any SnapManager operations scheduled and running on other hosts, these operations wait until the rolling upgrade or rollback operation is complete and then continues.
- Profiles that point to the same repository database must be created with different names in the SnapManager server hosts.

  If you use profiles with the same name, the rolling upgrade and rollback of the hosts involving that repository database fail without sending a warning message.
- Perform the rolling upgrade or rollback of the SnapManager server hosts when the repository utilization is at a minimum or there are no database operations that are running using the repository.
- Ensure that you do not perform any database operations on the host that is being rolling upgraded or rolled back.

After you perform the rolling upgrade, you need to keep a number of considerations in mind:

- If the SnapManager server starts automatically after the install or upgrade process, you must restart the SnapManager server after performing the rolling upgrade. Otherwise, you cannot view the SnapManager schedules.

To improve the rolling upgrade performance on the SnapManager server hosts, check for the following considerations:

- Ensure that you perform the rolling upgrade or rollback of the SnapManager server hosts one-by-one.
- If you want to perform the rolling upgrade or rollback of multiple SnapManager server hosts, ensure that you have a minimum number of profiles and backups.

  The rolling upgrade or rollback operation runs for a longer time as the cumulative number of backups of the hosts that are being upgraded together increases.
- If you have SnapManager hosts with a minimum number of backups, then you can perform a rolling upgrade or rollback of these hosts together.

Rolling upgrade considerations for related host(s)

When any SnapManager operation is performed, for example, a clone is created from the host A to the host B or a backup is mounted from the host A to the host B, and when the host A is rolling upgraded or rolled back, the following warning messages are displayed:

- The following host(s) are related to host(s), upgrade these host(s) immediately after this upgrade.
- The following host(s) are related to host(s), rollback these host(s) immediately after this rollback.

This warning message is displayed even though the clone is deleted or the backup is unmounted for the host B during the rollback or the rolling upgrade of the host A since the metadata exists in the repository for the operations performed on the remote host.

Rolling upgrade scenarios

The following table lists the different rolling upgrade scenarios and the effect of the rolling upgrade:

Rolling upgrade scenario	Effect on SnapManager 3.2 for Oracle
Rolling upgrade from SnapManager 3.0 for Oracle to SnapManager 3.2 for Oracle Rolling upgrade from SnapManager 3.0.1 for Oracle, SnapManager 3.0.2 for Oracle, or SnapManager 3.0.3 for Oracle to SnapManager 3.2 for Oracle Rolling upgrade from SnapManager 3.1 for Oracle to SnapManager 3.2 for Oracle	• Operations manager and RBAC capabilities available in SnapManager 3.0 for Oracle are applied to SnapManager 3.2 for Oracle. Protection policies are available in SnapManager 3.2 for Oracle. • Schedules associated with the target database are upgraded and are available in SnapManager 3.2 for Oracle. • Backups taken in the earlier version of SnapManager for Oracle are available in SnapManager 3.2 for Oracle. • Backups mounted using the earlier version of SnapManager are available as mounted in SnapManager 3.2 for Oracle. • Clones created in the earlier version of SnapManager for Oracle are available after the rolling upgrade.

Performing a rolling upgrade on a single host or multiple hosts

To perform a rolling upgrade on a single or multiple SnapManager server hosts, you use the `repository rollingupgrade` command. The upgraded SnapManager server host is managed only with the later version of SnapManager for Oracle.

About this task

After you perform a rolling upgrade on the SnapManager server host, you can perform normal database operations on the target database.

If you use a RAC configuration, you must manually upgrade all RAC-associated hosts with the later version of SnapManager for Oracle and SnapDrive for UNIX. You can use the rolling upgrade with multiple hosts option to specify host names of all the RAC-associated hosts.

The later version of SnapManager for Oracle retains the host-based user credentials, the Oracle software user credentials, and the Oracle RMAN user credentials.

Note: SnapManager 2.2 or 2.1.1 for Oracle supports database authentication mode only for RAC configuration.

Note: If you change the authentication mode from database authentication to OS authentication after the rolling upgrade on the host, and then perform a rollback, you must manually change the authentication mode from OS to database to perform further operations using the SnapManager 2.2 or 2.1.1 versions.

For details about considerations during rolling upgrade, see "Considerations for performing a rolling upgrade".

The following table lists the versions of SnapManager for Oracle, SnapDrive for UNIX and SnapDrive for Windows supported for the rolling upgrade:

Applications	Versions		
SnapManager for Oracle	3.0 to 3.2	3.0.1 to 3.2 3.0.2 to 3.2 3.0.3 to 3.2	3.1 to 3.2
SnapDrive for UNIX	4.1 to 5.0	4.1 to 5.0	4.2 to 5.0
SnapDrive for Windows	6.0.1 to 6.3	6.1 to 6.3	6.2 to 6.3

Steps

1. To perform a rolling upgrade on a single SnapManager server host, enter the following command:

   smo repository rollingupgrade -repository-dbname *repo_service_name* -host *repo_host* -login -username *repo_username* -port *repo_port* -upgradehost *host_with_target_database* -force [-quiet | -verbose]

   Example: Migrating all target databases mounted on hostA and a repository database named repoA located on a repo_host are rolling upgraded.

   ```
   smo repository rollingupgrade
       -repository
         -dbname repoA
         -host repo_host
         -login
          -username repouser
          -port 1521
        -upgradehost hostA
   ```

2. To perform a rolling upgrade on multiple SnapManager server hosts, enter this command:

   smo repository rollingupgrade -repository -dbname *repo_service_name* -host *repo_host* -login -username *repo_username* -port *repo_port* -upgradehost *host_with_target_database1,host_with_target_database2* -force [-quiet | -verbose]

For multiple hosts, enter the hostnames separated by a comma and ensure you do not provide any space between the comma and the next hostname.

Note: On Windows platform, ensure you enter the whole set of multiple hostnames within double quotes.

Note: In a RAC environment, you must perform rolling upgrade of all the SnapManager server hosts. You can use the `-allhosts` option to perform rolling upgrade of all the hosts in one shot.

Example: Migrating all target databases mounted on the hosts, hostA and hostB and a repository database named repoA located on a repo_host are rolling upgraded.

```
smo repository rollingupgrade
    -repository
      -dbname repoA
      -host repo_host
      -login
       -username repouser
       -port 1521
     -upgradehost hostA,hostB
```

3. To perform rolling upgrade on all the SnapManager server hosts on a repository database, enter this command:

smo repository rollingupgrade -repository -dbname *repo_service_name* -host *repo_host* -login -username *repo_username* -port *repo_port* -allhosts -force [-quiet | -verbose]

Example: Migrating all target databases available on a repository database named repoA located on a repo_host are rolling upgraded.

```
smo repository rollingupgrade
    -repository
      -dbname repoA
      -host repo_host
      -login
       -username repouser
       -port 1521
      -allhosts
```

Considerations for performing a rollback

Before you initiate a roll back, you need to keep a number of considerations in mind:

- External scripts that are used to perform any external data protection or data retention must be restored.
- SnapDrive 5.0 for UNIX or SnapDrive 6.3 for Windows in the SnapManager server host must be uninstalled.
- Original earlier version of SnapManager and SnapDrive for UNIX or SnapDrive for Windows from which you have performed rolling upgrade on the SnapManager server host must be installed.

- The repository database must be backed up before you perform the rollback.

  Note: Once you have separated the profiles to create archive log backups, you cannot roll back SnapManager to earlier versions.

After you perform the roll back of a repository database and downgrade the SnapManager host from SnapManager 3.2 for Oracle to SnapManager 3.0 for Oracle on the AIX platform, you can check for the following considerations to view the schedules created in the earlier version of the repository database.

You can perform the following steps to view the schedules:

1. Check the files in the directory by entering the following command:
   ```
   cd /opt/Ontap/smo/repositories
   ```
2. List the files. The directory might have two files for each repository with filenames as:
   - `repository#SMO300a#SMOREPO1#10.72.197.141#1521`
   - `repository-smo300a-smorepo1-10.72.197.141-1521`

   The file with the hash symbol (#) is created using the SnapManager 3.1 or 3.2 for Oracle version. The file with the hyphen symbol (-) is created using the SnapManager 3.0 for Oracle version.
3. Rename the file. You can do this by changing the hash symbol to the hyphen symbol. Enter the following command:
   - `mv repository#SMO300A#SMOREPO1#10.72.197.141#1521`
   - `repository-SMO300a-SMOREPO1-10.72.197.141-1521`

Rollback scenarios

The following table lists the operations that are supported and the operations that are not supported after a rollback of one or more SnapManager server hosts:

Rollback scenarios	Effects of rollback	What will not be available after roll back?

Rollback from SnapManager 3.2 for Oracle to SnapManager 3.0.1 for Oracle, SnapManager 3.0.2 for Oracle, or SnapManager 3.0.3 for Oracle	• Backups created in SnapManager 3.2 for Oracle are rolled back. • Backups freed in SnapManager 3.2 for Oracle are rolled back. • Backups deleted in SnapManager 3.2 for Oracle are rolled back. • Clones created from a backup using SnapManager 3.2 for Oracle are rolled back. • Change in protection status of the backup in SnapManager 3.2 for Oracle is rolled back. • Profile credentials changed for the upgraded profile using SnapManager 3.2 for Oracle, are applied to the profile in the rolled back version.	The e-mail notification capability added to the profile is not rolled back. History configuration set for the profile are not rolled back.

Rollback from SnapManager 3.2 for Oracle to SnapManager 3.1 for Oracle	• Backups created in SnapManager 3.2 for Oracle are rolled back. • Backups freed in SnapManager 3.2 for Oracle are rolled back. • Clones created from a backup using SnapManager 3.2 for Oracle are rolled back. • Change in protection status of the backup in SnapManager 3.2 for Oracle is rolled back. • Profile credentials changed for the upgraded profile using SnapManager 3.2 for Oracle, are applied to the profile in the rolled back version. • The e-mail notification capability added to the profile is rolled back.	History configuration set for the profile are not rolled back.

Performing a roll back on a single host or multiple hosts

SnapManager enables you to roll back or revert from a higher version of the product to the original version from which you upgraded.

About this task

The hosts, profiles, schedules, backup, and clones, associated with the profiles of target databases for the chosen host, are migrated to the previous repository version except the scenarios where the roll back is not supported. You can now use these rolled back profiles in the original version of the product

If you use RAC configuration, you must manually roll back all RAC associated hosts to the original lower version of SnapManager for Oracle and SnapDrive for UNIX.

The higher version SnapManager retains the host-based user credentials, the Oracle software user credentials and the Oracle RMAN user credentials.

For details about considerations before rolling back, see "Specific considerations before roll back".

Note: If there is a database operation running on the upgraded version of SnapManager, you must wait until that operation completes and only then proceed with rolling back.

Note: If you have set protection policy, retention class and applied SnapVault and SnapMirror relationships for the databases using a higher version of SnapManager for Oracle, you must manually roll back all of these capabilities to the original lower version.

Steps

1. To roll back from a higher version of SnapManager to the original version on a single SnapManager server host, enter this command:

   smo repository rollback -repository -dbname *repo_service_name* -host *repo_host* -login -username *repo_username* -port *repo_port* -rollbackhost *host_with_target_database*

   Example: Migrating all target databases mounted on hostA and a repository database named repoA located on a repo_host are rolled back.

   ```
   smo repository rollback
       -repository
         -dbname repoA
         -host repo_host
         -login
          -username repouser
          -port 1521
        -rollbackhost hostA
   ```

2. To roll back from a higher version of SnapManager to their original versions on multiple SnapManager server hosts, enter this command:

   smo repository rollback -repository-dbname *repo_service_name* -host *repo_host* -login -username *repo_username* -port *repo_port* -rollbackhost *host_with_target_database1,host_with_target_database2*

   For multiple hosts, enter the hostnames separated by a comma and ensure you do not provide any space between the comma and the next hostname.

   Note: On Windows platform, ensure you enter the whole set of multiple hostnames within double quotes.

   Example: Migrating all target databases mounted on the hosts, hostA, hostB and a repository database named repoA located on a repo_host are rolled back.

   ```
   smo repository rollback
       -repository
         -dbname repoA
         -host repo_host
         -login
          -username repouser
          -port 1521
        -rollbackhost hostA,hostB
   ```

Post-upgrade considerations

After a software upgrade, consider the issues related to repository, backup retention, and restore.

Post-upgrade repository considerations

After a SnapManager software upgrade, consider the following repository issues:

The installation and upgrade procedure instructs you to update your existing repositories when upgrading to this version of SnapManager. This updates the repository database schema and objects and you can access them using SnapManager for Oracle.

When you update existing repositories, the existing clones are also updated.

After you upgrade a repository, you must reboot the SnapManager server to restart any associated schedules.

Note: After you update the repository, you cannot use the repository with an older version of SnapManager for Oracle

Post-upgrade backup retention considerations

Upon upgrade to the current version, SnapManager assigns backup retention defaults to existing backups. You might want to adjust these backup defaults to meet your backup requirements.

Backup type	Retention class assignment after upgrade
Backups marked to be retained forever	Unlimited
All other backups	Daily

You can delete backups retained on an unlimited basis without changing the retention class.

After an upgrade to SnapManager 3.0 or later versions, SnapManager sets the larger of the following two values to existing profiles:

- The previous retention count for the profile
- The overall default values for the retention count and duration of daily backups as specified in the SnapManager `smo.config` file.

You can change these defaults by editing the `smo.config` file. For example, enter these new defaults:

```
retain.hourly.count = 12
retain.hourly.duration = 2
```

Related tasks

Setting configuration properties on page 69

Post-upgrade restore considerations

Only backups created in a SnapManager 3.X can be restored using the fast restore process. Backups created in previous versions are restored using the file-based restore process.

Related concepts

Database restore overview on page 176

Updating existing repositories

You can update an existing repository so that you can access it using the upgraded SnapManager software.

About this task

If you are upgrading to SnapManager 3.0.1 or later versions, you must perform the following steps in this order:

- Install SnapManager.
- Upgrade the SnapManager repository.
- Start the SnapManager server.
- Verify the SnapManager system.

Note: However, if you are upgrading to SnapManager 3.1, you must start and verify the SnapManager server prior to upgrading the repository.

You can use any valid host name, service name, or user name. For a repository to support SnapManager scheduling, the repository user name and service name must consist of only the following characters: alphabetic characters (A-Z), digits (0-9), minus sign ("-"), underscore ("_"), and period (".").

The repository port can be any valid port number and the repository host name can be any valid host name. In other words, the host name must consist of alphabetic characters (A-Z), digits (0-9), minus sign ("-"), and period (".") , but not an underscore ("_").

Step

1. Enter this command:

   ```
   smo repository update -repository -dbname repo_service_name -host
   repo_host -login -username repo_username -port repo_port
   ```

   Other options for this command are as follows:

   ```
   [-force] [-noprompt]

   [quiet | -verbose]
   ```

Updating an existing repository

The following command updates an existing repository:

```
smo repository update -repository -dbname SALESDB
-host server1 -login -username admin -port 1521
```

Configuring SnapManager for Oracle

After installing the software, you can configure SnapManager to work in your environment.

Setting configuration properties

Administrators can set several configuration properties, such as maximum number of log files, timeouts for commands, backup retention frequency defaults, methods that SnapManager uses to restore data from secondary storage when it cannot restore data directly by Protection Manager, and a default temporary volume name if SnapManager needs to create a temporary volume.

About this task

Administrators can set the following properties in the `smo.config` file.

From SnapManager 3.2 for Oracle, you can set configuration parameters in the SnapManager configuration file (smo.config) for managing archive log files.

Property	Explanation and example

Retention class defaults	Sets the defaults for the retention policy when you create a profile.
	`retain.hourly.count = 12`
	`retain.hourly.duration = 2`
	`retain.monthly.count = 2`
	`retain.monthly.duration = 6`

Access to secondary storage	Defines how SnapManager accesses data on secondary storage when it cannot be restored directly by Protection Manager: • Direct: Clones data on secondary storage, mounts the cloned data from the secondary storage to the host, and then copies data out of the clone into the active environment. (Default) • Indirect: Copies data to a temporary volume on primary storage, mounts data from the temporary volume to the host, and copies data out of the temporary volume into the active environment. This policy should be used only if the host does not have direct access to the secondary storage system. Restores using this method take twice as long as the "direct" access because two copies of the data are made. This affects only SAN environments; for NFS SnapManager does not need to connect directly to secondary storage to perform a restore. `restore.secondaryAccessPolicy = direct`

Property	Explanation and example
Temporary volume name	Sets the default name of a temporary volume. When SnapManager uses the indirect policy for restoring data from secondary, it requires a scratch volume on primary storage to hold a temporary copy of data until it is copied into the database files and the database is recovered. There is no default value. If you do not specify a value, you must enter a name in each restore command that uses the indirect method. `restore.temporaryVolumeName = smo_temp_volume`
Always free expired backups	Directs SnapManager to free backups when they expire (per the backup retention policy) and when a fast restore of an earlier backup is performed, even if data protection is not configured. • Frees protected backups that expire. • Deletes unprotected backups that expire. Set to one of the following: • True: Free (not delete) expired backups regardless of whether backups are protected. • False: Free expired backups that are protected. Delete them if not protected or if the protected copies on secondary storage have also expired. `retain.alwaysFreeExpiredBackups = true`
Persist host credentials	By default, SnapManager does not store a user's credentials on the host. However, you can change this. For example, if you have custom scripts that require access on a remote server, as when a script runs a remote clone. To store host credentials, set the host.credentials.persist property to "True." SnapManager encrypts and saves the user's credentials for the host. `host.credentials.persist = true`
Maximum number of files displayed in the restore preview	By default, SnapManager displays a maximum of 20 files in the restore preview. However, you can change this. Enter a value greater than 0. If you specify an invalid value, the default is used instead. `restorePlanMaxFilesDisplayed = 30`

Setting up configuration parameters for archive log management

Property	**Explanation and example**

Pruning parameter for pruning the archive log files from the specified archive log destinations or other (external) archive log destinations	If there are backups of archive logs present in the multiple archive log destinations, check for the pruning-related SnapManager configuration parameter (`pruneIfFileExistsInOtherDestination`) in the SnapManager configuration file (smo.config):
	• Set the parameter value as `false` to prune the archive log files from specified destinations, if the archive log files are backed up from the specified destinations (provided using the `-prune-dest` option).
	`pruneIfFileExistsInOtherDestination = false`
	• Set the parameter value as `true` to prune the archive log files from specified destinations, if the archive log files are backed up at least once from any one of the other destinations.
	`pruneIfFileExistsInOtherDestination = true`

Pruning parameter for pruning the archive log files backed up from the specified archive log destinations or other archive log destinations	To prune the archive log files from the specified archive log destinations or other archive log destinations, include the pruning-related SnapManager configuration parameter `prune.archivelogs.backedup.from.otherdestination` in the SnapManager configuration file (smo.config):
	• Set the parameter value as false to prune the archive log files from the specified destinations, if the archive log files are backed up from the specified destinations (provided using the option `-prune-dest`).
	`prune.archivelogs.backedup.from.otherdestination = false`
	• Set the parameter value as true to prune the archive log files from specified destinations, if the archive log files are backed up at least from any one of the other destinations.
	`prune.archivelogs.backedup.from.otherdestination = true`

Pruning parameter to prune maximum number of archive log files at a time	The maximum number of archive log files that you can prune at a time must be less than 1000. `maximum.archivelog.files.toprune.atATime = 998`
Archive log consolidation parameter	Set the value as true to free the duplicate archive log backups `archivelogs.consolidate=true`

Property	Explanation and example
Exclude parameters for not a Snapshot capable storage	If the database is on not a Snapshot capable storage and you want to perform SnapManager operations on that storage, you should add the exclude parameter (archivedLogs.exclude) in the SnapManager configuration file (smo.config). The exclude parameter is used to exclude the archived log files from not a Snapshot capable storage from the profiles and backups. Ensure that you set the exclude parameter before creating a profile. Only after setting the exclude parameter in the SnapManager configuration file, the profile creation is successful. You can add the exclude parameter in the SnapManager configuration file in three ways; however, the latter two ways are recommended configurations for excluding large number of archived log files and for better performance. To exclude archived log files from being included in the profile and being backed up, use one of the below parameters: **1.** Exclude archived log files from all profiles or backups on the SnapManager host using a regular expression (`archivedLogs.exclude`). **2.** Exclude archived log files from all profiles or backups on SnapManager host using a SQL expression (`archivedLogs.exclude.fileslike`). **3.** Exclude archived log files only from a single profile or database using a SQL expression (`<db-unique-name>.archivedLogs.exclude.fileslike`). The detailed note on how to exclude archived log files using the regular expression and SQL expressions are given in the following rows.

Property	Explanation and example
Exclude parameters for not a Snapshot capable storage using regular expression	Exclude archived log files from all profiles or backups on the SnapManager host using the regular expression (`archivedLogs.exclude`). This parameter if set would apply to all profiles or backups created on this host. The values of this parameter can either be a top level directory or a mount point where the archived log files are present or a sub directory. The archived log files matching the regular expression are excluded from the profile and backups. If a top level directory or a mount point is specified and if data protection is enabled for a profile on the host then that mount point or directory would not be included in the dataset that is created in the Protection Manager. When there are multiple archived log files to be excluded from the host, separate the archive log file paths using commas. (This option is deprecated; use option 2 or 3 for better performance). Example of using the regular expression `archivedLogs.exclude` on Windows platform: `archivedLogs.exclude=J:\\ARCH\\.*` **Note:** On Windows platform, if the destination has a file separator, then an additional slash symbol (\) must be added in the pattern and the pattern must end with a double-slash pattern (\\.*). For example, if you want to exclude the destination J:\arch1, then you must provide the value as J:\\arch1\\.*.
Exclude parameters for not a Snapshot capable storage using a SQL expression (`archivedLogs.exclude.fileslike`	Exclude archived log files from all profiles or backups on SnapManager host using the SQL expression (`archivedLogs.exclude.fileslike`). This parameter if set would apply to all profiles or backups created on the host. The values of the parameter can either be a top level directory or a mount point where archived log files are present or a sub directory. The archived log files matching the SQL expression are excluded from the profile and backups. If a top level directory or mount point is specified and if data protection is enabled for a profile on the host, then that mount point or directory would not be included in the dataset that is created in Protection Manager. When there are multiple archived log files to be excluded from the host, separate the archive log file paths using commas. Example of using SQL expression on Windows platform: `archivedLogs.exclude.fileslike=J:\\ARCH2\\%` **Note:** On Windows platform, if the destination has a file separator, then an additional slash symbol (\) must be added in the pattern and the pattern must end with a double-slash pattern (\\%). For example, if you want to exclude the destination J:\arch1, then you must provide the value as J:\\arch1\\%.

Property	Explanation and example
Exclude parameters for not a Snapshot capable storage using a SQL expression (`<db-unique-name>.archivedLogs.exclude.fileslike`)	Exclude archived log files only from a single profile or database using the SQL expression (`<db-unique-name>.archivedLogs.exclude.fileslike`). This parameter if set would apply to only the profile or backups created for the database with the specified `db-unique-name`. The values of this parameter can either be a top level directory or mount point where the archived log files are present or a sub directory. The archived log files matching the SQL Expression are excluded from the profile and backups. If a top level directory or mount point is specified and if data protection is enabled for a profile on the host then that mount point or directory would not be included in the dataset that is created in Protection Manager. When there are multiple archived log files to be excluded from the host, separate the archive log file paths using commas. Example of using SQL expression on Windows platform: `mydb.archivedLogs.exclude.fileslike=J:\\ARCH2\\%` **Note:** On Windows platform, if the destination has a file separator, then an additional slash symbol (\) must be added in the pattern and the pattern must end with a double-slash pattern (\\%). For example, if you want to exclude the destination J:\arch1, then you must provide the value as J:\\arch1\\%.
Parameter to always include the archive log files beyond the missing files in the backup.	The archive log files which do not exist in the active file system are not included in the backup. The archive log files created before the missing files are included in the backup. `backup.archivelogs.beyond.missingfiles=true` Set the value to false to ignore all the archive log files created beyond the missing files to be included in the backup.
Suffix parameter to differentiate the label names of the data backup and the archive log backup	When a user specifies the label for a backup (containing datafiles and archive log file together) created using the profiles that are separated for taking archive log backups, the label names for the data backup and the archive log backup are differentiated with a suffix. `suffix.backup.label.with.logs=logs` By default, an underscore parameter `(_logs)` is added to the archive log backup label, for example `arch_logs`. To edit the parameter `(_logs)`, edit the suffix parameter: `suffix.backup.label.with.logs`. For example, you can specify the value as `suffix.backup.label.with.logs=arc` so that the `_logs` default value is changed to `_arc`.

Steps

1. Access the configuration file in the following default location:

   `<default installation location>/properties/smo.config`

2. Edit the `smo.config` file.
3. After changing any settings, restart the SnapManager server.

Related tasks

Starting SnapManager for Oracle on page 79

What to do when you encounter heap space issue

About this task

When you encounter heap space related issues during SnapManager operation, perform the following steps:

Steps

1. Navigate to SnapManager installation directory.
2. Open the launch-java file from the path: `<installationdirectory>/bin/launchjava`.
3. Increase the value of the java heap space parameter `java -Xmx160m` to a higher value.

   For example, you can modify the value from a default value of `160m` to `200m` as `java -Xmx200m`.

4. If you have increased the value of the java heap space parameter in the earlier versions of SnapManager for Oracle, for example `512m`, you should retain that value in SnapManager 3.2 for Oracle.

Ensuring that ASM can discover imported disks

After you install SnapManager for Oracle, ensure that ASM can discover disks imported by SnapManager. You do this by setting the path in the ASM_DISKSTRING parameter and this path should be set until the ASM disk directory.

About this task

Considerations for ASM on NFS disks

Before you install SnapManager for Oracle, your environment paths might appear similar to the following:

- ASM on NFS disk1: `/mnt/my-asm-disks/dir1/disk1.nfs`

- ASM on NFS disk2: `/mnt/my-asm-disks/dir1/disk2.nfs`

In this case, the ASM_DISKSTRING would show this: `/mnt/my-asm-disks/dir1/*`

However, if this path remains, ASM in an NFS environment cannot discover disks imported by SnapManager.

To ensure that ASM can discover disks imported by SnapManager, after you install SnapManager, provide path name for the ASM_DISKSTRING parameter to the maximum until the ASM directory path. Ensure that you give the exact NFS disk file names following the ASM directory path:

```
ASM_DISKSTRING = '/mnt/my-asm-disks/dir1/disk*,/opt/Ontap/smo/mnt/*/*/
disk*'
```

The ASM_DISKSTRING parameter must match only the ASM disk files and not any other files.

Ensure that the wildcards in ASM_DISKSTRING match the topology of your NFS file system when mounted in a subdirectory associated with the following location:

```
 /opt/Ontap/smo/mnt/<smo-generated-name>/
```

The wildcards in the ASM_DISKSTRING represent the following:

- The first * matches the name (generated by SnapManager) for the root mountpoint.
- The second * matches the directory within the mountpoint.
- The third disk* matches the name of the NFS file itself.

Steps

1. Using Oracle tools, access the ASM_DISKSTRING parameter.

   For information about editing the ASM_DISKSTRING parameter, see the Oracle documentation.

2. Edit the ASM_DISKSTRING parameter so that it points to the current ASM directory path.

   Example

   Example showing how SnapManager connects the ASM disks:

   Consider the following ASM disks mount point:

   ```
    /mnt/my-asm-disks/disk1.nfs
   ```

   SnapManager connects the ASM disks as:

   ```
   /opt/Ontap/smo/mnt/my-asm-disks-20081012/disk1.nfs
   ```

   In this case, the ASM_DISKSTRING is in the form:

   ```
   /opt/Ontap/smo/mnt/*/disk*
   ```

   The wildcards in the ASM_DISKSTRING represent the following:

   - The first * matches my-asm-disks-20081012.
   - The disk* matches disk1.nfs.

Example

Example showing ASM to discover disks imported by SnapManager:

- Clone of ASM on NFS disk1: `/opt/Ontap/smo/mnt/-mnt-my-asm-disks-20081012/dir1/disk1.nfs`
- Clone of ASM on NFS disk2: `/opt/Ontap/smo/mnt/-mnt-my-asm-disks-20081012/dir1/disk2.nfs`

In this case, the ASM_DISKSTRING is in the form:

```
/opt/Ontap/smo/mnt/*/*/disk*
```

The wildcards in the ASM_DISKSTRING represent the following:

- The first * matches -mnt-my-asm-disks-20081012.
- The second * matches dir1.
- The third disk* matches disk1.nfs and disk2.nfs.

Considerations for ASM with FCP and iSCSI for LINUX

Using SnapManager, you can provision ASM disks with or without ASMLib over FCP and iSCSI on LINUX.

To ensure that ASM can discover disks imported by SnapManager on LINUX, after you install SnapManager, change the permission of the Oracle software owner and primary group of the user. Ensure that you use only the character device.

With ASMLib, the ASM DISKSTRING path will appear as:

```
 ASM DISKSTRING = ORCL:*
```

Without ASMLib, the ASM DISKSTRING path will appear as:

```
ASM DISKSTRING = /dev/sd*
```

Without ASMLib and with multipathing, the ASM DISKSTRING path will appear as:

```
ASM DISKSTRING = /dev/mapper/*
```

Without ASMLib and without multipathing, the ASM DISKSTRING path will appear as:

```
ASM DISKSTRING = /dev/sd*
```

The wildcard matches with the ASM disk name.

For raw devices, the ASM DISKSTRING path will appear as:

```
ASM DISKSTRING = /dev/raw/*
```

Considerations for ASM with FCP and iSCSI for HPUX

Using SnapManager, you can provision ASM disks with ASMLib over FCP and iSCSI on HPUX.

To ensure that ASM can discover disks imported by SnapManager on HPUX, after you install SnapManager, provide path name for the ASM_DISKSTRING parameter to the maximum until the ASM directory path.

```
ASM DISKSTRING = /dev/rdisk/*
```

The wildcard matches with the ASM disk name.

Example showing ASM disks mount point:

```
 /dev/rhdisk1
```

SnapManager connects the ASM disks as:

```
 / dev/rhdisk1
```

In this case, the ASM_DISKSTRING is in the form:

```
 /dev/*
```

The wildcard in the ASM_DISKSTRING represent ASM disk name.

Considerations for ASM with FCP and iSCSI for AIX

To ensure that ASM can discover disks imported by SnapManager on AIX, after you install SnapManager, provide path name for the ASM_DISKSTRING parameter to the maximum until the ASM directory path.

```
ASM DISKSTRING = /dev/ *
```

The wildcard matches with the ASM disk name.

Considerations for ASM with FCP and iSCSI for SOLARIS

To ensure that ASM can discover disks imported by SnapManager on SOLARIS, after you install SnapManager, provide path name for the ASM_DISKSTRING parameter to the maximum until the ASM directory path.

```
ASM DISKSTRING = /dev/rdsk/*
```

The wildcard matches with the ASM disk name.

Starting SnapManager for Oracle

The SnapManager startup section lists the tasks that you perform when you start SnapManager. Use this section also if you are just learning about SnapManager.

Before you begin

Before using SnapManager, you should have performed the following actions:

- Downloaded and installed the SnapManager software.
- Determined whether you will use the graphical user interface or the command-line interface.

Steps

1. *Identifying an existing database to backup* on page 79
2. *Verifying the Oracle listener status* on page 80
3. *Creating Oracle users for the repository database* on page 80
4. *Creating an Oracle user for the target database* on page 81
5. *Starting SnapManager* on page 82
6. *Verifying the environment* on page 88
7. *Creating repositories* on page 89
8. *Following the order of operations* on page 91

Identifying an existing database to backup

Identify the Oracle SID of the SnapManager database you will use to create a profile.

About this task

The standard Oracle user ID for non-SAP systems is "oracle".

Steps

1. On the SnapManager for Oracle host server, as root, enter this command:

   ```
   su - oracle
   ```

2. To find the ORACLE SID, at the `[oracle@server1~]#` prompt, enter this command:

   ```
   cat /etc/oratab
   ```

   Output:

   ```
   # This file is used by ORACLE utilities...
   # Multiple entries with the same $ORACLE_SID are not allowed.
   dedb:/mnt/dibbert/server1_orahome:N
   ```

Result

The database ORACLE_SID is dedb.

Verifying the Oracle listener status

Check the Oracle listener status and make note of the listener port. A standard Oracle installation sets the listener port on the database to 1521.

Step

1. To verify the status of the listener, connect to it and make note of the listener port (`1524` in this example).

   At the `[oracle@server1~]$` prompt, enter this command:

   `lsnrctl status`

   Example

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC)))
STATUS of the LISTENER
------------------------
Alias LISTENER
Version TNSLSNR for Linux: Version 9.2.0.6.0 - Production
Start Date 16-MAY-2008 15:52:43
Uptime 40 days 21 hr. 27 min. 0 sec
Trace Level off
Security OFF
SNMP OFF
Listener Parameter File /etc/listener.ora
Listener Log File /home/oracle/product/9i2nd/network/log/listener.log
Listening Endpoints Summary...
(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC)))
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=server1.vmware)
(PORT=1524)))
Services Summary...
Service "dedb" has 1 instance(s).
Instance "dedb", status UNKNOWN, has 1 handler(s) for this service...
Service "ORCL" has 1 instance(s).
Instance "ORCL", status UNKNOWN, has 1 handler(s) for this service...
...
The command completed successfully.
```

Creating Oracle users for the repository database

Create an Oracle user on the SnapManager database for the repository that has not yet been created and assign specific privileges.

About this task

You should grant the "connect" and "resource" privileges to this user. You do not have to create a user for the repository database with sysdba privileges.

Note: This is different from the need to create an Oracle user with the sysdba role for the target database that SnapManager will manage. To manage a database, SnapManager requires that an Oracle user with the sysdba role connect to that database and perform database operations.

Steps

1. Log on to SQLPlus as SYSDBA.

   At the `[oracle@server1 ]` prompt, enter the command:

   `sqlplus '/ as sysdba'`

   Example

   ```
   SQL*Plus: Release 11.2.0.1.0 Production on Wed Jun 1 06:01:26 2011
   Copyright (c) 1982, 2009, Oracle.  All rights reserved.
   Connected to:
   Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
   With the Partitioning, Automatic Storage Management, OLAP, Data Mining
   and Real Application Testing options
   ```

2. To create a user, for example *repo1_user*, for the repository with the administrator password, for example, *adminpw1*, enter this command at the SQL prompt:

   `SQL> create user repo1_user identified by adminpw1;`

3. To grant connect and resource privileges to the user, enter this command:

   `grant connect, resource to repo1_user;`

Creating an Oracle user for the target database

To manage a database, SnapManager requires that an Oracle user with the sysdba role connects to that database and performs database operations.

About this task

SnapManager can use any Oracle user with sysdba privileges that exists in the target database, for example, the built-in "sys" user. You can create a user in the target database to be used exclusively by SnapManager.

Steps

1. Log on to SQLPlus as SYSDBA.

   At the `[oracle@server1 ]` prompt, enter the command:

   `sqlplus '/ as sysdba'`

2. To create a user, for example *smo_oper* with the administrator password, for example, *adminpw1*, enter this command at the SQL prompt:

   `SQL> create user smo_oper identified by adminpw1;`

3. Grant sysdba privileges to this user by entering the following command:

   ```
   SQL> grant sysdba to
   smo_oper;
   ```

Starting SnapManager

To use SnapManager, you can either issue commands from the command-line interface or use a graphical user interface on a host system in the same network as your database host.

You can use SnapManager for Oracle in the following ways:

- By entering commands in a command-line interface (CLI) on a UNIX host that is in the same network as your database host. Use a single command to perform each operation. The CLI provides the ability to invoke SnapManager for Oracle from scripts and from alternate hosts.
  - To access Windows CLI, **Start > All Programs > Ontap > SnapManager for Oracle > > Start SMO command-line interface (CLI)**.

  The commands all start with smo (SnapManager for Oracle). For example, you have commands such as `smo repository create` or `smo backup create`.
  For a list of all the commands and an explanation of their options and arguments, see the Command Reference.
- By accessing the graphical user interface on a host in the same network as your database host. The graphical user interface provides a simple tool with easy-to-use wizards to help you perform SnapManager for Oracle tasks.

Related concepts

SnapManager for Oracle command reference on page 261

Starting the SnapManager Windows host server

Steps

1. Go to **Start > Control Panel > Administrative Tools > Services**.
2. With the Services window open, select Ontap.
3. You can start the server in one of three ways:
   - If you are running SnapManager on a Windows host server, you must start the server before you can initiate any SnapManager operations.
     In the left panel, click **Start**.
   - Right-click Ontap and select **Start** from the drop-down menu.
   - Double-click Ontap, and then in the Properties window, click **Start**.

Verifying the SnapManager Windows host server status

The server must be running for you to execute commands or initiate SnapManager operations. If you are running SnapManager on a Windows host server, you might want to verify the status of the server first.

Steps

1. With the **Services** window open, select **Ontap**.
2. View the status in the Status column.

Using SnapManager commands

After you start the SnapManager host server, you can use SnapManager by entering commands at the prompt on your host.

Step

1. To perform an operation:
   - In case of a Windows host, go to **Start > All Programs > Ontap > SnapManager for Oracle > Start SMO Command Line Interface (CLI)**

Starting the SnapManager graphical user interface

If SnapManager is installed on the host, start the graphical user interface for SnapManager by using a command (UNIX) or by selecting the program from a list of programs (Windows).

Before you begin

Before starting the SnapManager graphical user interface, ensure that the SnapManager server is already started.

About this task

You can start the SnapManager graphical user interface in two ways:

1. From the SnapManager server host on a Windows host,
   select **Start > All Programs > Ontap > SnapManager for Oracle > Start SMO GUI**.
2. If SnapManager is not installed on the host, use Java Web Start, which downloads SnapManager components and starts the graphical user interface.

Related tasks

Downloading and starting the graphical user interface using Java Web Start (Windows) on page 84

Downloading and starting the graphical user interface using Java Web Start (Windows)

If SnapManager is not installed on the host, use Java Web Start, which downloads SnapManager components and starts the graphical user interface. This procedure explains this method on Windows.

Before you begin

Before downloading the graphical user interface, ensure you have any one of the following Windows OS version and web browser version:

Windows OS version

- Windows Server 2008
- Windows Server 2003
- Windows 7

Note: For additional details on the supported Windows edition and service pack versions, check the Compatibility Matrix.

Web browser version

- Microsoft Internet Explorer 6 is supported only on Windows Server 2003
- Microsoft Internet Explorer 8
- Mozilla Firefox 3.5

Before starting the graphical user interface, complete the following tasks:

- Start the SnapManager server.
- Open a Microsoft Internet Explorer or Mozilla Firefox Web browser window.

About this task

Starting the graphical user interface using Java Web Start when there is no JRE available in the Windows client

Note: If you are using Mozilla Firefox Web browser, you must manually download and install JRE 1.6.

Steps

1. In the **Microsoft Internet Explorer** Web browser window, enter the following:

   `https://`*`smo-server.domain.com`*`:port` where *`smo-server.domain.com`* is the fully qualified host name and domain on which you installed SnapManager and port is the listening port for the SnapManager server (27214, by default).

   Note: Ensure that you enter `https` in the browser window.

A dialog box with the message "There is a problem with the site's security certificate...Do you want to proceed?" is displayed.

2. Click **Yes** or **Continue**.
3. Click on the link labeled "Click here to download and install JRE 6.0 and the application".

   A link labeled "Download Java Web Start" with the message "This site might require the following ActiveX control: Java Plug-in 1.6"... "Click here to install" is displayed.

4. In the Install window, perform the following steps:
   a. Click the message labeled "Click here to install...".

      An **Install ActiveX Control** menu is displayed.

   b. Select **Install ActiveX Control...**.

      The message "Internet Explorer - Security Warning" containing the following text: "Do you want to install this software? Name: Java Plug-in 1.6" is displayed.

   c. Click **Install**.

      A "Java Plug-in 1.6." window for the installer for J2SE Runtime Environment 1.6 is displayed.

   d. Click **Install**.

   A window requesting you to install J2SE Runtime Environment 1.6 is displayed.

5. In the Install window, perform the following steps:
   a. On the **License Agreement** page, select **I accept the terms in the license agreement** and click **Next**.

   b. On the **Setup Type** page, select **Typical** and click **Next**. The installation begins.

   c. On the **Installation Completed** window, click **Finish**. SnapManager starts to download.

   A File Download dialog box with the message "Do you want to save this file? application.jnlp" is displayed.

6. In the file download window, perform the following steps:
   a. Install the latest version of JRE 1.6 on the Windows client.

   b. Verify that Java is installed by running the following command: `java -version`. The output should indicate Java version "1.6.0_24" (which is Java 1.6) or later.

   c. Change your Windows configuration settings to always open files with extension "jnlp" with the program Java Web Start Launcher. The exact mechanism to do this varies based on the version of Windows you are using.

   d. Enter the SnapManager URL that you have specified in the step 1.

   The SnapManager download starts on the Windows client and a "Warning - security" dialog box is displayed.

7. Perform the following steps. The message contents and button labels vary based on platform.
   a. In the **"Warning - Security"** dialog box with a message about the certificate of the SnapManager Web site, click **Yes**.

      A dialog box with a title host name mismatch is displayed.

   b. In the hostname mismatch dialog box, click **Run**.

      The "Warning - Security" dialog box with a message about the signature of the SnapManager application is displayed.

   c. Click **Run**.

      A dialog box with the title "Java Installer - Security Warning" and the message "Warning Security - the application's digital signature has an error. Do you want run the application," is displayed.

   d. Click **Run**.

      The browser downloads and starts the SnapManager for Oracle graphical user interface.

About this task

Starting the graphical user interface using Java Web Start when there is JRE 1.5 available in the Windows client

Steps

1. In the Microsoft Internet Explorer or Mozilla Firefox Web browser window, enter the following:

   `https://`*`smo-server.domain.com`*`:port` where *`smo-server.domain.com`* is the fully qualified host name and domain on which you installed SnapManager and port is the listening port for the SnapManager server (27214, by default).

   Note: Ensure that you enter `https` in the browser window.

   A dialog box with the message "There is a problem with the site's security certificate...Do you want to proceed?" is displayed.

2. Click **Yes** or **Continue**.
3. Click the link labeled **Launch SnapManager for Oracle**.

   The SnapManager download starts on the Windows client and a "Warning - security" dialog box is displayed.

4. Perform the following steps. The message contents and button labels vary based on platform.
   a. In the **"Warning - Security"** dialog box with a message about the certificate of the SnapManager Web site, click **Yes**.

A dialog box with a title host name mismatch is displayed.

b. In the host name mismatch dialog box, click **Run**.

The "Warning - Security" dialog box with a message about the signature of the SnapManager application is displayed.

c. Click **Run**.

A dialog box with the title "Java Installer - Security Warning" and the message "Warning Security - the application's digital signature has an error. Do you want run the application," is displayed.

d. Click **Run**.

The SnapManager graphical user interface is launched.

Note: On a Windows 7 platform with JRE 1.5, the error message Unable to launch file displays. To check for the detailed error message, click **Details**. The application has requested a version of the JRE (version 1.6) that is not locally installed. Java Web Start is unable to download and install the requested version automatically. The JRE 1.6 must be installed manually.

About this task

Starting the graphical user interface using Java Web Start when there is JRE 1.6 available in the Windows client

Steps

1. In the Microsoft Internet Explorer or Mozilla Firefox web browser window, enter the following:

   `https://`*`smo-server.domain.com`*`:port` where *`smo-server.domain.com`* is the fully qualified host name and domain on which you installed SnapManager and port is the listening port for the SnapManager server (27214, by default).

   Note: Ensure that you enter `https` in the browser window.

   A dialog box with the message "There is a problem with the site's security certificate...Do you want to proceed?" is displayed.

2. Click **Yes** or **Continue**.
3. Click the link labeled **Launch SnapManager for Oracle**.

   The SnapManager download starts on the Windows client and a "Warning - security" dialog box is displayed.

4. Perform the following steps. The message contents and button labels vary based on platform.

a. In the **"Warning - Security"** dialog box with a message about the certificate of the SnapManager web site, click **Yes**.

   A dialog box with a title hostname mismatch is displayed.

b. In the hostname mismatch dialog box, click **Run**.

   The "Warning - Security" dialog box with a message about the signature of the SnapManager application is displayed.

c. Click **Run**.

   A dialog box with the title "Java Installer - Security Warning" and the message "Warning Security - the application's digital signature has an error. Do you want run the application," is displayed.

d. Click **Run**.

   The SnapManager graphical user interface is launched.

Verifying the environment

You can verify the environment to make sure SnapDrive and SnapManager are set up correctly.

Before you begin

Download, install, and set up the required prerequisites. Make sure SnapManager is installed and the host server is running.

Step

1. To verify that SnapDrive is installed and can be run from the root account, run the following command:

   ```
   smo system verify
   ```

Related references

The smo system verify command on page 359

Verifying SnapDrive for Windows

If you are using SnapDrive for Windows, verify that you can create a Snapshot copy before using SnapManager.

Steps

1. From the Start menu, right-click on **My Computer** and select **Manage**.
2. In the Computer Management window, select **Storage > SnapDrive**.

3. Select a disk.

   If you successfully found disk information for the SnapDrive product, SnapDrive is working correctly. See the *SnapDrive Installation and Administration Guide* for more information about using SnapDrive.

Creating repositories

SnapManager requires a repository on a host to hold data about the operations you perform.

Before you begin

Ensure that the following tasks are completed:

1. Create an Oracle user and password in the repository database.
2. Authorize user access to the repository.

For a repository, SnapManager for Oracle requires a minimum 4K block size for the tablespace into which it is installed. You can check the block size using the following SQL command:

```
select a.username, a.default_tablespace, b.block_size
from dba_users a, dba_tablespaces b
a.username = repo_user
```

where

- `a.default_tablespace` = `b.tablespace_name`
- `a.username` = the user name on the repository

About this task

If you are upgrading repositories, you must reboot the SnapManager server to restart any associated schedules.

Step

1. To create the repository, enter the `repository create` command, using the following general format:

   `smo repository create -repository -dbname repo_service_name -host repo_host -login -username repo_username -port repo_port -force] [-noprompt] [-quiet | -verbose]`

   Where:

   - `-repository -dbname` is the name of the repository database.
   - `-host` is the name of the host for the repository.
   - `-username` is the name of the database user who has access to the repository.

- `-port` is the port for the host.

Other options for this command are as follows:

```
[-force] [-noprompt]

[quiet | -verbose]
```

Note: If you have an existing repository with the same name and you use the `-force` option, all data within an existing repository schema will be overwritten.

Creating a repository

The following command line creates a repository:

```
smo repository create -repository -dbname HRDP
-host server1 -login -username admin -port 1521
```

About organizing repositories

Organize the SnapManager repositories to meet your business needs. You can organize them in several ways, including by application type and by usage.

You can organize repositories in several ways. Here are two suggestions:

- By application type
  If you have multiple Oracle databases running different applications, you can create a SnapManager repository for every application type that you have. Each SnapManager repository would have profiles for the databases of a particular application type. All production, development, and testing databases of that application type would be managed by the same SnapManager repository. This would help group similar databases and ease cloning. However, if you have several application types, then you might have to manage several SnapManager repositories, and if you choose to implement another application type, you will need to create another SnapManager repository. Since these SnapManager repositories will be managing production databases, each of these repositories must be on a server with high availability, which could be expensive. Also, managing production databases along with development and test databases of the same type in the same SnapManager repository could be a security issue.
- By usage
  Another option is to distribute the databases among the SnapManager repositories based on their usage (for example, production, development, testing, and training). This limits the number of repositories to the different types of databases that you have. Because all production databases would be managed by a single SnapManager repository, only production DBAs can be given access to this repository. Also, if you choose to deploy another database for a new application type, then you just need to register it in the corresponding SnapManager repository instead of creating a new repository. High availability can be provided only for the SnapManager repository that holds profiles of all the production databases.

SnapManager for Oracle and SnapManager for SAP should not share the same repository. For SnapManager for Oracle and SnapManager for SAP, it is recommended that you use a different repository (meaning a different Oracle database user) for each product if you have both in your

environment. Using a different repository, either in the same or different databases, provides more flexibility by allowing independent upgrade cycles for each product.

Following the order of operations

SnapManager allows you to perform various operations such as creating profiles, performing backups, and cloning backups. Some operations must be done in a specific order.

About this task

You should perform operations in the following order:

1. Create a profile.
2. Perform a backup.
3. Restore the backup.
4. Create a clone specification file.
5. Clone a database.

Steps

1. Create a profile: You can create a profile on an existing repository using the `smo profile create` command.

   Note: The Oracle user specified for the target database must have sysdba privileges.

   Example

   ```
   smo profile create -profile prof1 -profile-password prof1cred
   -repository -dbname HR1 -login -username admin -host server1 -port 1521
   -database -dbname dedb -login -username db_oper2
   -password dbpw1 -host server1 -port 1521 -osaccount oracle
   -osgroup dba
   ```

2. Create a backup: You can create a backup on an existing profile using the `smo backup create` command.

   Example

   ```
   smo backup create -profile prof1 -full -offline -label full_backup_prof1 -force
   ```

3. Restore a database: You can restore and recover a database backup on primary storage with SnapManager using `smo backup restore` command.

   Example

   ```
   smo backup restore -profile prof1 -label full_backup_prof1
   -complete -recover -alllogs
   ```

4. Create a clone specification: You can create a template clone specification using the `smo clone template` command or in the graphical user interface, use the Clone wizard to create a template

clone specification. You can also create the clone specification file using a text editor. Refer to "Creating clone specifications."

5. Clone a database: You can clone a database using an existing backup using the `smo clone create` command. Before that, you must have an existing clone specification or create one to specify the storage and database specifications for the clone.

   Example

   ```
   smo clone create -profile prof1 -backup-label full_backup_prof1
   -newsid clone1 -label prof1_clone -clonespec /opt/<path>/smo/clonespecs/prof1_clonespec.xml
   ```

Managing security and credentials

Security in SnapManager is governed by a combination of user authentication and role-based access control (RBAC). Authentication allows access to resources, such as repositories, hosts, and profiles. RBAC allows you to restrict the operations that SnapManager can perform against the volumes and LUNs that hold the datafiles in your database. Additionally, data protection features are available only if the N series Management Console data protection capability is installed.

When you first run a command from either the command line or graphical user interface, SnapManager retrieves the credentials set for repositories and profiles. SnapManager saves credentials from previous installations.

Objects in SnapManager, such as the repository and profile, can be secured with a password. In contrast, a credential is the password configured for the user for an object, not the password configured on the object itself.

You can manage authentication and credentials by performing the following tasks:

- Manage user authentication either through password prompts on operations or by using the `smo credential set` command. You can set credentials for a repository, host, or profile.
- View the credentials that govern the resources to which you have access.
- Clear a user's credentials for all resources (hosts, repositories, and profiles).
- Delete a user's credentials for individual resources (hosts, repositories, or profiles).

You can manage role-based access to SnapManager operations by performing these tasks:

- Using SnapDrive, enable role-based access control for SnapManager.
- Using Operations Manager (formerly DataFabric Manager), assign users to roles and set role capabilities.
- Optionally, enable SnapManager to store encrypted passwords by editing the `smo.config` file.

Access to features is also affected by whether the N series Management Console data protection capability is installed.

- If the N series Management Console data protection capability is installed, when the user creates a database profile, SnapManager creates a dataset in Operations Manager and populates the dataset with the volumes that contain the database files. SnapManager keeps the dataset contents synchronized with the files comprising the database upon each backup.
- If the N series Management Console data protection capability is not installed, SnapManager cannot create a dataset. Users cannot set protection on profiles.

Administrators can perform tasks with the SnapManager graphical user interface or by using the command line interface. The *SnapManager for Oracle Installation and Administration Guide* explains how to complete these tasks using commands. The SnapManager online Help explains how to complete the tasks using the graphical user interface.

About user authentication

In addition to using role-based access control, SnapManager maintains security by authenticating the user using an OS login on the host where the SnapManager server is running. Enable and manage user authentication either through password prompts on operations or by using the `smo credential set` command.

User authentication requirements depend on where the operation is performed.

- If the SnapManager client is on the same server as the SnapManager host, which is typically the case, the user's OS login authenticates the user. Users are not prompted for their passwords because they are already logged into the host where the SnapManager server is running.
- If the SnapManager client is connected to the SnapManager server on a different host, SnapManager needs to authenticate users with both their OS logins and passwords. SnapManager prompts users for passwords for any operation, if users have not saved their OS login credentials in their SnapManager user credential cache. If you enter `smo credential set -host` command, you save the OS logins and passwords in each user's SnapManager credential cache file and SnapManager will not prompt for the password for any operation.

The user who is authenticated with the SnapManager server is considered the effective user. The effective user for any operation must be a valid user account on the host on which the operation executes. For example, if a user is executing a clone operation, the user should be able to log into the destination host for the clone.

You can manage credentials by performing the following tasks:

- Optionally, configure SnapManager to store user credentials in the SnapManager user credentials file. By default, SnapManager doesn't store host credentials. You might want to change this, for example, if you have custom scripts that require access on a remote host. The remote clone operation is one example of a SnapManager operation that needs the login credentials of a user for a remote host. To have SnapManager remember user host login credentials in the SnapManager user credentials cache, set the host.credentials.persist property to "true" in the `smo.config` file.
- Authorize user access to the repository.
- Authorize user access to profiles.
- View all user credentials.
- Clear a user's credentials for all resources (hosts, repositories, and profiles).
- Delete credentials for individual resources (hosts, repositories, or profiles).

Storing encrypted passwords for custom scripts

By default, SnapManager does not store host credentials in the SnapManager user credentials cache. However, you can change this. For example, if you have custom scripts that require access on a

remote server, as when a script runs a remote clone. To store host credentials, edit the `smo.config` file.

Steps

1. Access the following default location:

   `<default installation location>/properties/smo.config`

2. Edit the `smo.config` file.
3. To store host credentials, set the `host.credentials.persist` property to "true."

Authorizing user access to the repository

In addition to using role-based access control in SnapManager, you can set user credentials for database users to provide them access to the repository. Using credentials, administrators can restrict or prevent access to the SnapManager hosts, repositories, profiles, and databases. Using credentials, administrators can give User1 access to a particular repository and profile on a particular host, but restrict User2 from accessing that host, repository, and profile.

About this task

If you set credentials using the `credential set` command, SnapManager does not prompt you for a password.

You can set user credentials when you install SnapManager, or you can set them later.

Step

1. Set the credentials for database users to access the repository from the SnapManager host server by using the command:

   `smo credential set -repository -dbname repo_service_name -host repo_host -login -username repo_username [-password repo_password] -port repo_port`

Authorizing user access to profiles

In addition to using role-based access control in SnapManager, you can set a password on a profile to prevent others from accessing your profiles.

Step

1. To set a password on a profile, enter the following command:

   `smo credential set -profile -name profile_name [-password password]`

Related references

The smo credential set command on page 306

Viewing user credentials

You can list the hosts, profiles, and repositories to which you have access.

Step

1. To list the resources to which you have access, enter this command:

   ```
   smo credential list
   ```

Example of viewing user credentials

This example displays the resources to which you have access.

```
smo credential list

Credential cache for OS user "user1":
Repositories:
Host1_test_user@SMOREPO/hotspur:1521
Host2_test_user@SMOREPO/hotspur:1521
user1_1@SMOREPO/hotspur:1521
Profiles:
HSDBR (Repository: user1_2_1@SMOREPO/hotspur:1521)
PBCASM (Repository: user1_2_1@SMOREPO/hotspur:1521)
HSDB (Repository: Host1_test_user@SMOREPO/hotspur:1521) [PASSWORD NOT SET]
Hosts:
Host2
Host5
```

Related references

The smo credential list command on page 305

Clearing user credentials for all hosts, repositories, and profiles

About this task

You can clear the cache of your credentials for resources (hosts, repositories, and profiles). When you try to access a repository, host, or profile, you will be required to authenticate with your credentials again to gain access to these secured resources.

This deletes all of the resource credentials for the user running the command.

Steps

1. To clear your credentials, enter the `smo credential clear` command from the SnapManager CLI or select **Admin** > **Credentials** > **Clear Cache** from the SnapManager GUI.
2. Exit the SnapManager GUI.

   Note:

   - If you have cleared the credential cache from the SnapManager GUI, you do not need to exit the SnapManager GUI.
   - If you have cleared the credential cache from the SnapManager CLI, you must restart SnapManager GUI.
   - If you have deleted the encrypted credential file manually, you must restart the SnapManager GUI again.

3. To set the credentials again, repeat the process to set credentials for the repository, profile host, and profile. For additional information on setting the user credentials again, refer to "Setting credentials after clearing credential cache".

Related references

The smo credential clear command on page 302

Setting credentials after clearing credential cache

SnapManager enables you to set the credentials for resources (hosts, repositories, and profiles) after clearing the cache of your credentials as a root user using the `smo credential clear` command from the SnapManager CLI or by selecting the **Clear Cache** option from the SnapManager GUI.

About this task

You have to restart the SnapManager GUI depending on how the cache is cleared.

Note:

- If you have cleared the credential cache from the SnapManager GUI, you do not need to exit the SnapManager GUI.
- If you have cleared the credential cache from the SnapManager CLI, you must restart SnapManager GUI.
- If you have deleted the encrypted credential file manually, you must restart the SnapManager GUI again.

Ensure that you set the same user credentials that you have given before for the repository, profile host, and the profile.

An encrypted credentials file created while setting the user credentials is deleted when you clear the user credentials (from the SnapManager CLI or GUI).

The credentials file is located at file path for the Windows environment: `C:\Documents and Settings\Administrator\Application Data\Ontap\smo\3.2.0`

From the SnapManager GUI, if there is no repository mapped under the **Repositories** tree, perform the following steps:

Steps

1. Add an existing repository using the **Add Existing Repository** option under the **Tasks** section or menu.
2. Right-click the repository, select **Open**, enter the user credentials in the **Repository Credentials Authentication** window.
3. Right-click the host under the repository, select **Open**, enter the user credentials in the **Host Credentials Authentication** window.
4. Right-click the profile under the host, select **Open**, enter the user credentials in the **Profile Credentials Authentication** window.

Deleting credentials for individual resources

You can delete the credentials for any one of the secured resources, such as a profile, repository, or host. This enables you to remove the credentials for just one resource, rather than clearing the user's credentials for all resources.

Related references

The smo credential delete command on page 303

Deleting user credentials for repositories

You can delete the credentials so a user can no longer access a particular repository. This command enables you to remove the credentials for just one resource, rather than clearing the user's credentials for all resources.

Step

1. To delete repository credentials for a user, enter this command:

   `smo credential delete -repository -dbname repo_service_name-host repo_host -login -username repo_username -port repo_port`

Deleting user credentials for hosts

You can delete the credentials for a host so a user can no longer access it. This command enables you to remove the credentials for just one resource, rather than clearing all the user's credentials for all resources.

Step

1. To delete host credentials for a user, enter this command:

   smo credential delete -host -name*host_name* -username-*username*

Deleting user credentials for profiles

You can delete the user credentials for a profile so a user can no longer access it.

Step

1. To delete profile credentials for a user, enter this command:

   smo credential delete -profile -name *profile_name*

Managing profiles for efficient backups

To perform any operation on a database using SnapManager, you must create a profile in SnapManager for that database. When you want to perform an operation on a database, you simply choose the profile and then choose the operation.

Tasks related to profiles

You can perform the following tasks:

- Create profiles to enable full or partial backups and backups to primary, secondary, or even tertiary storage.
  Create profiles to separate the archive log backups from the datafile backups.
- Verify profiles
- Update profiles
- Delete profiles

About profiles and authentication

When you create a profile, you specify a database and can choose one of the following the methods that lets you connect to the database:

- Oracle authentication with a username, password, and port
- Operating system (OS) authentication with no username, password, or port. For OS authentication, you enter the OS account user and group information. To use OS authentication for the RAC databases, the SnapManager server must be running on each node of the RAC environment.

Note: If Oracle authentication is used, the database password must be the same for all Oracle instances in a RAC environment. SnapManager uses the database username and password to connect to every RAC instance in the profile.

If this is the first time you are accessing a profile, you must enter your profile password. Once you've entered credentials, you can view the database backups within the profile.

Administrators can perform tasks with the SnapManager graphical user interface or by using the command-line interface. The *SnapManager for Oracle Installation and Administration Guide* explains how to complete these tasks using commands. The SnapManager online Help explains how to complete the tasks using the graphical user interface.

What profiles are

SnapManager uses profiles to store the information necessary to perform operations. A profile holds the information about the database being managed, including its credentials, backups, and clones, while a repository holds data about the operations performed on the profiles. By creating a profile,

you do not need to specify database details each time you perform an operation on that database. You simply supply the profile name.

A profile can reference only one database. That same database can be referenced by more than one profile. Backups created using one profile cannot be accessed from a different profile, even if both the profiles reference the same database.

Profile information is stored in a repository. The repository contains both the profile information for the database and information about the Snapshot copies that provided the backup of the database. (The actual Snapshot copies are stored on the storage system.) The Snapshot copy names are stored in the repository containing the profile for that database. When you perform an operation on a database, you must select the profile from the repository.

The following figure illustrates how repositories can hold multiple profiles, but each profile can define only one database. In this example, Repository2 is on database DB1 and Repository1 is on the database DB4.

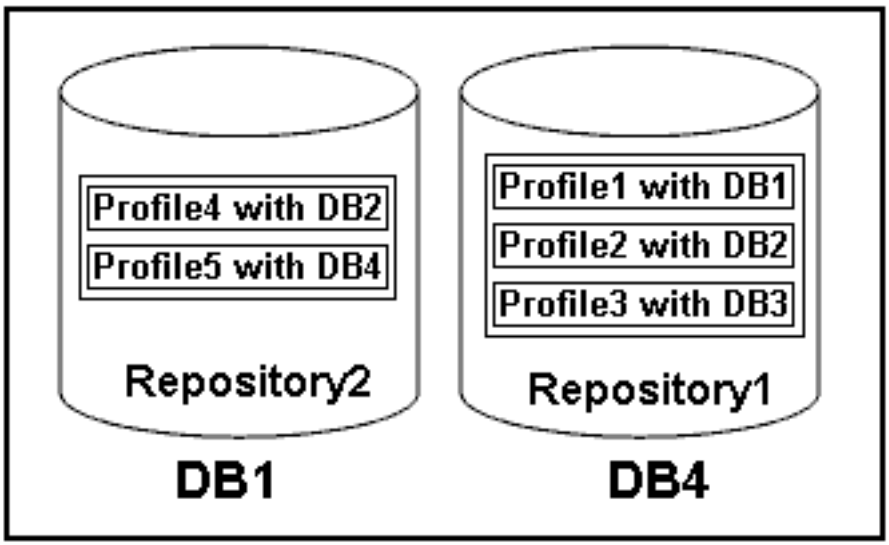

Figure 4: Multiple profiles in separate repositories

Each profile contains the credentials for the database associated with the profile. The credentials enable SnapManager to connect to and work with the database. The stored credentials include the username and password pairs for accessing the host, the repository, the database, and the required connection information if using RMAN.

You cannot access a backup created using one profile from a different profile, even if both profiles are associated with the same database. SnapManager places a lock on the database during an operation to prevent two incompatible operations from being performed simultaneously.

You can create the profiles to take full backups or partial backups. Starting from SnapManager 3.2 for Oracle, you can create profiles that enable you to take backups of the archive log files separately from the data files.

Profile for creating full and partial backups

The profiles that you specify to create the full and partial backups, contain the datafiles and archive log files together. SnapManager does not allow such profiles to separate the archive log backups from the datafile backups. The full and partial backups are retained based on the existing backup retention policies, and protected based on the existing protection policies. The full and partial backups can be scheduled based on the time and frequency that suits you.

Profiles for creating datafiles-only backups and archive logs-only backups

Starting from SnapManager 3.2 for Oracle enables to separate archive log backups from the datafiles backup. Once you have separated the backup using the profile, you can either create the datafiles-only backups or archive logs-only backups of the database. You can also create a backup containing both the datafiles and archive log files together.

The retention policy applies to all the database backups when the archive log backups are not separated. On separating the archive log backups, SnapManager allows to specify different retention duration and protection policy for the archive log backups.

Retention policy

SnapManager determines whether a backup should be retained by considering both the retention count (for example, 15 backups) and the retention duration (for example, 10 days of daily backups). A backup expires when its age exceeds the retention duration set for its retention class and the number of backups exceed the retention count. For example, if the backup count is 15 (meaning that SnapManager has taken 15 successful backups) and the duration requirement is set for 10 days of daily backups, the five oldest successful eligible backups expire.

Archive log retention duration

When the archive log backups are separated, the archive log backups are retained based on the archive log retention duration. Archive log backups taken along with datafiles backup are always retained along with datafiles backup irrespective of the archive log retention duration.

How SnapManager determines which backups to retain on local storage

Using SnapManager, database administrators create backups that meet retention policies, which specify how many successful backups on local storage should be retained to support business requirements. The retention strategy involves several scenarios to handle regulatory requirements, loss of data, and disasters. SnapManager lets you specify how many successful backups it should retain in the profile for a given database. The retention policy is engaged every time you create a new backup.

An administrator might require the following backups for a production payroll database:

- 10 days of daily backups on primary storage
- 2 months of monthly backups on primary storage

- 7 days of daily backups on secondary storage
- 4 weeks of weekly backups on secondary storage
- 6 months of monthly backups on secondary storage

For each profile in SnapManager, administrators can change any value for any non-limited retention classes:

- Hourly
- Daily
- Weekly
- Monthly

SnapManager determines whether a backup should be retained by considering both the retention count (for example, 15 backups) and the retention duration (for example, 10 days of daily backups). A backup expires when its age exceeds the retention duration set for its retention class or the number of backups exceed the retention count. For example, if the backup count is 15 (meaning that SnapManager has taken 15 successful backups) and the duration requirement is set for 10 days of daily backups, the five oldest successful eligible backups expire.

After a backup expires, SnapManager either frees or deletes the expired backup. SnapManager always keeps the last backup taken.

SnapManager counts only the number of successful backups as eligible for the retention count and does not consider the following in that count:

Backups not included in the retention count	**Additional details**
Failed backups	SnapManager keeps information on both successful and unsuccessful backups. Although unsuccessful backups take up only minimal space in the repository, you might want to delete them. Unsuccessful backups remain in the repository until you delete them.
Backups designated to be retained on an unlimited basis or backups for a different retention class than the one for this count	SnapManager does not delete backups designated to be retained on an unlimited basis. Additionally, SnapManager considers only those backups in the same retention class (for example, SnapManager considers only the hourly backups for the hourly retention count).
Backups mounted from local storage	When Snapshot copies are mounted, they are also cloned and so are not considered eligible for retention. SnapManager cannot delete the Snapshot copies if they are cloned.
Backups that are used to create a clone on local storage	SnapManager keeps all backups used to create clones, but does not consider them for the backup retention count.

Backups not included in the retention count	Additional details
Backups that are cloned or mounted on secondary storage and that use the protection policy of "mirror"	If SnapManager did delete the Snapshot copies for the backup on the primary storage resource and it is mirrored, the next backup to secondary storage would fail.

When you free a backup from its primary storage resources, the primary resources (Snapshot copies) used by the backup are destroyed, but the backup metadata is still available. SnapManager does not consider freed backups eligible for deletion, so freed backups *are not* considered in the backup retention count.

SnapManager provides a default retention count and duration for each retention class. For example, for the hourly retention class count, SnapManager by default retains four hourly backups. Administrators can override these defaults and set the values when creating or updating the profile or they can change the default values in a SnapManager configuration file (`smo.config` file).

The default values for retention count and duration are stored in the `smo.config` file.

Backups on primary storage can be protected to secondary storage. While SnapManager manages the retention and scheduling of backups on primary storage, the N series Management Console data protection capability manages the retention and scheduling of backups on secondary storage.

When local backups expire based on their retention policy, they are either deleted or freed, depending on whether they are protected:

- If they are protected, the local backups will be freed, meaning that their storage resources or Snapshot copies will be deleted, but the backups will remain in the SnapManager repository and will be available for restoration from secondary storage. You do not have to manually free backups (for example, with the backup free command). Backups remain freed until the backup no longer exists on secondary, at which point the backup is deleted.
- If they are not protected, the local backups will be deleted.

The following example shows the actions that SnapManager takes on various types of backups based on a retention policy set to retain three daily backups (with the count set to retain 3).

Backup date	Status	Retention policy action taken	Explanation
5/10	Successful	Keep	This is the most recent successful backup, so it will be kept.
5/9	Successful, cloned	Skip	Because SnapManager does not consider backups used for cloning in the retention policy count, this backup is omitted from the count of successful backups.

Backup date	Status	Retention policy action taken	Explanation
5/8	Successful, mounted	Skip	Because SnapManager does not consider mounted backups in the retention policy count, this backup is omitted from the count of successful backups.
5/7	Failed	Skip	Failed backups are not counted.
5/5	Successful	Keep	SnapManager keeps this second successful daily backup.
5/3	Successful	Keep	SnapManager keeps this third successful daily backup.
5/2	Successful	Delete	SnapManager counts this successful backup, but after SnapManager reaches three successful daily backups, SnapManager deletes this backup.

Snapshot copy naming

You can specify a naming convention or pattern to describe the Snapshot copies related to the profile you create or update. You can also include custom text in all Snapshot copy names.

You can change the Snapshot copy naming pattern when you create a profile or after the profile has been created. The updated pattern applies only to Snapshot copies that have not yet occurred; Snapshot copies that exist retain the previous snapname pattern.

The following examples show the two Snapshot copy names taken for a volume. The second Snapshot copy listed has _F_H_1_ in the middle of its name. The "1" indicates that it is the first Snapshot copy taken in the backup set. The first Snapshot copy listed is the most recent and has a "2," which means it is the second Snapshot copy taken. The "1" Snapshot copy includes the datafiles; the "2" Snapshot copy includes the control files. Because the control file Snapshot copies must be taken after the data file Snapshot copy, two Snapshot copies are required.

```
smo_profile_sid_f_h_2_8ae482831ad14311011ad14328b80001_0
smo_profile_sid_f_h_1_8ae482831ad14311011ad14328b80001_0
```

The default pattern includes the required smid, as shown in the following:

- Default pattern: smo_{profile}_{db-sid}_{scope}_{mode}_{smid}
- Example: smo_my_profile_rac51_f_h_2_8abc01e915a55ac50115a55acc8d0001_0

You can use the following variables in the Snapshot copy name:

Variable name	Description	Example value
smid (Required)	The SnapManager unique ID is the only required element when creating a name for the Snapshot copy. This ID ensures that you create a unique Snapshot name.	8abc01e915a55ac50115a55acc8d0001_0
class (Optional)	Retention class associated with the backup for the profile and indicated by hourly (h), daily (d), weekly (w), monthly (m), or unlimited (u).	d
comment (Optional)	Comment associated with the backup for the profile. Spaces in this field will be converted to underscores when the Snapshot copy name is complete.	sample_comment_spaces_replaced
date (Optional)	Date that the backup occurs for the profile. Date values are padded with zeros if necessary. (yyyymmdd)	20070218
db-host (Optional)	Database host name associated with the profile being created or updated.	my_host
db-name (Optional)	Database name associated with the Snapshot copy you create.	rac5
db-sid (Optional)	Database sid associated with the Snapshot copy you create.	rac51
label (Optional)	Label associated with the backup for the profile.	sample_label
mode (Optional)	Specifies whether the backup is completed online (h) or offline (c).	h
profile (Optional)	Profile name associated with the backup you create.	my_profile
scope (Optional)	Specifies whether the backup is either full (f) or partial (p).	f
time (Optional)	Time that the backup occurs for the profile. Time values for this variable use the 24-hour clock and are padded with zeros if necessary. For example, 5:32 and 8 seconds appears as 053208. (hhmmss)	170530
time-zone (Optional)	Time zone specified for the target database host.	EST

Variable name	Description	Example value
usertext (Optional)	Custom text that you can enter.	prod

Note: SnapManager for Oracle does not support the colon (:) symbol in the long forms of the names for Snapshot copies.

Creating profiles

You have various options when you create a profile, including the way you will access the profile. You can choose how you access the profile, select the database, and select the host associated with the profile. When creating profiles, you can assign a particular Oracle database user account. You can set the retention policy for the profile, enable backup protection to secondary storage for all backups using this profile, and set the retention count and duration for each retention class.

About this task

If you do not specify the -login, -password and the -port parameters of the database, SnapManager defaults to the OS authentication mode.

Starting from SnapManager 3.2 for Oracle, you can specify to separate archive log files from the datafiles while creating a new profile or updating an existing profile. Once you have separated the backup using the profile, you can either take only the datafiles-only backup or archive logs-only backup of the database. Using the new profile or the updated profile, you can also create the backup containing both the datafiles and archive log files. You can neither use the profile to create the full backup nor revert the settings from separating the backup.

Profile for creating full and partial backups

Using the profiles, you can create the full database backup containing the datafiles, control files and archive log files and partial database backup containing specified datafiles or tablespaces, all the control files, and all the archive log files.

SnapManager does not allow to take separate archive log backups using such profiles.

Profiles for creating datafiles-only backups and archive logs-only backups

SnapManager 3.2 for Oracle enables you to take archive log files backup and datafiles backup separately. To take separate backups, you must create or update the existing profile selecting the option to separate the archive log file backups from the datafile backups. Using the profile, you cannot take full backup but you can take archive log files backup along with online datafiles backup.

Using the new profile options to separate the archive log backups, you can perform the following SnapManager operations:

- Create archive log backup
- Delete archive log backup

- Mount archive backup
- Free archive log backup

While creating the profile to separate archive log backups from the datafiles backup, if the archive log files do not exist in the database for which the profile is created, then a warning message `Archived log file does not exist in the active file system. The archived log file versions earlier than the <archive log thread version> log file will not be included in the backup` displays. Even if you create backups for this database, the archive log files will not be available in the database backups.

Note: If you encounter an error while creating a profile, use the `smo system dump` command. Once you have successfully created a profile, use the `smo operation dump` and `smo profile dump` commands.

Step

1. To create a profile with a username, password, and port (Oracle authentication), enter this command:

   smo profile create -profile *profile* [-profile-password *profile_password*] -repository -dbname *repo_dbname* -host *repo_host* -port *repo_port* -login -username *repo_username* -database -dbname *db_dbname* -host *db_host* [-sid *db_sid*] [-login -username*db_username* -password *db_password* -port *db_port*] [-rman {-controlfile | {-login -username *rman_username* -password *rman_password* -tnsname *rman_tnsname*} }] -osaccount *osaccount* -osgroup *osgroup* [-retain [-hourly [-count *n*] [-duration *m*]] [-daily [-count *n*] [-duration *m*]] [-weekly [-count *n*] [-duration *m*]] [-monthly [-count *n*] [-duration *m*]]] [-comment *comment*][-snapname-pattern *pattern*][-protect [-protection-policy *policy_name*]] [-summary-notification] [-notification [-success -email *email_address1, email_address2* -subject *subject_pattern*] [-failure -email *email_address1, email_address2* -subject *subject_pattern*]][-separate-archivelog-backups -retain-archivelog-backups -hours *hours* | -days *days* | -weeks *weeks*| -months *months* [-protect [-protection-policy *policy_name*] | -noprotect] [-include-with-online-backups | -no-include-with-online-backups]] [-dump]

   Other options for this command are as follows:

   `[-force] [-noprompt]`

   `[quiet | -verbose]`

   Note: For RAC environments, ensure that you provide the value of the parameter `db_unique_name` as `db_ dbname` for the `-database -dbname` option when you create a new profile.

   You have other options when creating profiles, depending on how you want to access the database, whether you are using RMAN, and if you want to set a specific retention policy.

If...	Then...
You want to use OS authentication to create the profile	Specify the variables for an OS account in the DBA group (typically the account used to install oracle). Instead of adding the username, password and port, specify: • `-osaccount`*`account_name`* is the name of the OS account. • `-osgroup` *`osgroup`* is the group associated with the OS account. **Note:** These OS variables are required for UNIX but not allowed for databases running on Windows.
You want to specify a backup retention policy for backups of this profile database	Specify either the retention count or the retention duration for a retention class, or both to specify the backup retention policy. The duration is in units of the class (for example, hours for hourly, days for daily). • `-hourly` is the hourly retention class, for which [`-count` *`n`*] [`-duration` *`m`*] are the retention count and retention duration, respectively • `-daily` is the daily retention class, for which [`-count` *`n`*] [`-duration` *`m`*] are the retention count and retention duration, respectively. • `-weekly` is the weekly retention class, for which [`-count` *`n`*] [`-duration` *`m`*] are the retention count and retention duration, respectively. • `-monthly` is the monthly retention class, for which [`-count` *`n`*] [`-duration` *`m`*] are the retention count and retention duration, respectively.
You want to enable backup protection for the profile	Specify these options and variables: • `-protect` creates a dataset • `-protection-policy` *`policy`* is the name of the protection policy to use for the dataset. **Note:** To list the possible protection policies, use the `smo protection-policy list` command. • `-noprotect` indicates not to protect the database backups created using the profile. **Note:** If `-protect` is specified without `-protection-policy`, then the dataset will not have a protection policy. If `-protect` is specified and `-protection-policy` is not set when the profile is created, then it may be set later by `smo profile update` command or set by the storage administrator through the N series Management Console data protection capability.

If...	Then...
You want to enable e-mail notification on the completion status of the database operations.	Specify these options and variables: • `-summary-notification` enables you to configure a summary e-mail notification for multiple profiles under a repository database. • `-notification` enables you to receive an e-mail notification on the completion status of the database operation for a profile. • `-success -email`*`email_address2`* enables you to receive an e-mail notification on the successful database operation performed using a new or an existing profile. • `-failure -email`*`email_address2`* enables you to receive an e-mail notification on the failed database operation performed using a new or an existing profile. • `-subject`*`subject_text`* specifies subject text for the e-mail notification while creating a new profile or an existing profile.
You want to backup archive log files separately from datafiles	Specify these options and variables: • `-separate-archivelog-backups` enables to separate the archive log backup from the datafile backup. Once this option is specified, you can either take datafiles-only backup or archive logs-only backup. You cannot take a full backup. Also, you cannot revert the profile settings from separating the backup. • `-retain-archivelog-backups` sets the retention duration for archive log backups. The archive log backups are retained based on the archive log retention duration. The datafiles backups are retained based on the existing retention policies. • `-protect` enables protection to the archive log backups. • `-protection-policy` sets the protection policy to the archive log backups. The archive log backups are protected based on the archive log protection policy. The datafiles backups are protected based on the existing protection policies. • `-include-with-online-backups` includes the archive log backup along with the online database backup. This option enables you to take an online datafiles backup and archive logs backup together for cloning. When this option is set, whenever you take an online datafiles backup, the archive logs backups are taken along with the datafiles immediately. • `-no-include-with-online-backups` does not include the archive log backup along with database backup.
You can collect the dump files after the successful profile create operation.	Specify `-dump` option at the end of the profile create command.

When you create a profile, SnapManager analyzes the files in case you later want to use a volume-based restore on the files specified in the profile.

Related concepts

How to collect dump files on page 367

Changing profile passwords

To protect the existing profiles in the repository, you should update the passwords for the profiles. You can apply this updated password when creating a backup using this profile.

Step

1. To update the profile password for an existing profile, enter this command:

   ```
   smo profile update -profile profile_name -profile-password password
   ```

Related references

The smo profile update command on page 335

Authorizing user access to profiles

In addition to using role-based access control in SnapManager, you can set a password on a profile to prevent others from accessing your profiles.

Step

1. To set a password on a profile, enter the following command:

   ```
   smo credential set -profile -name profile_name [-password password]
   ```

Related references

The smo credential set command on page 306

Verifying profiles

You can verify that an existing profile is set up correctly. When you verify a profile, SnapManager checks the environment for the profile you specify and verifies that the profile is set up and the database in this profile is accessible.

Step

1. To verify if the profile is set up correctly, enter this command:

   ```
   smo profile verify -profile profile_name
   ```

Related references

The smo profile verify command on page 340

Updating profiles

About this task

You can modify the profile password, the number of backups to retain, the access to the database and information about the host.

If the Oracle database password information changes, you must change this information in the profile.

If protection policy was set on the profile, you cannot change the policy using SnapManager. The storage administrator must change the policy using the N series Management Console data protection capability.

SnapManager 3.2 for Oracle enables you can update the profile to separate archive log backups from the datafile backups using the `-separate-archivelog-backups` option. You can specify separate retention duration and protection policy for the archive log backup. SnapManager enables you to include the archive log backup along with online database backup. You can also create an online datafile backup and archive log backup together for cloning. When this option is set, whenever you take an online datafiles backup, the archive logs backups are taken along with the datafiles immediately.

Steps

1. To update a profile, enter this command:

   ```
   smo profile update -profile profile [-profile-password profile_password]
   [-database -dbnamedb_dbname -host db_host [-sid db_sid] [-login -
   usernamedb_username -password db_password-port db_port]] [{-rman{-
   controlfile | {-login  -username rman_username -password  rman_password
   ```

-tnsname *rman_tnsname*}}} | -remove-rman]-osaccount *osaccount* -osgroup *osgroup* [-retain [-hourly [-count *n*] [-duration *m*]] [-daily [-count *n*] [-duration *m*]] [-weekly [-count *n*] [-duration *m*]] [-monthly [-count *n*] [-duration *m*]]] [-comment *comment*][-snapname-pattern *pattern*][[-protect [-protection-policy *policy_name*]]| [[-noprotect]] [-summary-notification] [-notification [-success -email *email_address1, email_address2* -subject *subject_pattern*] [-failure -email *email_address1, email_address2* -subject *subject_pattern*]] [-separate-archivelog-backups -retain-archivelog-backups -hours *hours* | -days *days* | -weeks *weeks*| -months *months* [-protect [-protection-policy *policy_name*] | -noprotect] [-include-with-online-backups | -no-include-with-online-backups]] [-dump]

Other options for this command are as follows:

`[-force] [-noprompt]`

`[quiet | -verbose]`

If...	Then...
You want to change the profile to use OS authentication	Instead of adding the username, password and port, specify: • `-osaccount`*account_name* is the name of the OS account • `-osgroup` *osgroup* is the group associated with the OS account, typically the account used to install oracle **Note:** These OS variables are required for UNIX but not allowed for databases running on Windows.
You want to change the backup retention policy for backups of the database in the profile	Specify either the retention count or retention duration for a retention class, or both for changing the retention policy. The duration is in units of the class (for example, hours for hourly, days for daily). • `-hourly` is the hourly retention class, for which [`-count` *n*] [`-duration` *m*]] are the retention count and retention duration respectively • `-daily` is the daily retention class, for which [`-count` *n*] [`-duration` *m*]] are the retention count and retention duration respectively. • `-weekly` is the weekly retention class, for which [`-count` *n*] [`-duration` *m*]] are the retention count and retention duration respectively. • `-monthly` is the monthly retention class, for which [`-count` *n*] [`-duration` *m*]] are the retention count and retention duration respectively.

If...	Then...
You want to update the profile to take backup of the archive log files separately	Specify the following options and variables: • `-separate-archivelog-backups` enables you to separate the archive log files from the database files. Once this option is specified, you can either take datafiles-only backup or archive logs-only backup. You cannot take a full backup. Also, you cannot revert the profile settings from separating the backup. SnapManager retains the backups based on retention policy for the backups taken before updating to separate the archive log backup. • `-retain-archivelog-backups` sets the retention duration for archive log backups. **Note:** If you are updating the profile for the first time to separate the archive log backups from the datafiles backup using the `-separate-archivelog-backups` option, you must provide the retention duration for the archive log backups using the `-retain-archivelog-backups` option. While updating the profile subsequently, setting the retention duration is optional. • `-protect` enables protection to the archive log backups. • `-protection-policy` sets the protection policy to the archive log backups. • `-include-with-online-backups` specifies to include the archive log backup along with the database backup. • `-no-include-with-online-backups` specifies not to include the archive log file backup along with database backup.
You want to change the hostname of the target database	Specify `-host` *`new_db_host`* to change the hostname of the profile.
You can collect the dump files after the profile update operation.	Specify `-dump` option at the end of the `profile update` command.

2. To view the updated profile, enter this command: `smo profile show`

Related concepts

How to collect dump files on page 367

Deleting profiles

You can delete a profile anytime, as long as it does not contain successful or incomplete backups. You can delete profiles that contain freed or deleted backups.

Step

1. To delete a profile, enter this command:

   ```
   smo profile delete -profile profile_name
   ```

Related references

The smo profile delete command on page 330

Backing up databases

All organizations have requirements that specify how frequently you must back up data and how long you must keep backup copies of data. SnapManager offers the ability to back up data on local storage resources (on the volume where the datafiles reside) or enable protected backups on secondary or even tertiary storage resources. The choice to back up to secondary storage provides an additional layer that preserves data in the case of a disaster. SnapManager provides several options for backing up, restoring, and recovering data.

SnapManager also enables storage administrators to configure their backups based on carefully thought out policy plans. With SnapManager, administrators can quickly see backups that are not conforming to policy requirements and rectify those immediately. SnapManager policy-based protection enables backup consistency and policy conformance predictability.

SnapManager provides the following options to back up, restore, and recover data in your database:

- Back up the entire database or a portion of it. If you back up a portion of it, specify a group of tablespaces or a group of data files.
- Back up the datafiles and archive log files separately.
- Back up databases to primary storage (also called local storage) and protect them by backing them up to secondary or tertiary storage (also called remote storage). (Data protection is not available on Windows.)
- Schedule routine backups.

How SnapManager 3.2 for Oracle differs from prior versions SnapManager

SnapManager versions prior to 3.2 version provides the ability to create the full database backups that contain the datafiles, control files, and all the archive log files together.

SnapManager versions prior to 3.2 version manages the datafiles alone and there are situations where the archive log files are maintained using solutions outside SnapManager for Oracle.

Moreover, managing database backups imposes several constraints to DBAs in prior version of SnapManager 3.2 for Oracle:

- Performance impact of database
  Backing up an online full database backups (when database in the backup mode) reduces the performance of the database for specific period of time until the backup is created. Now with SnapManager 3.2 for Oracle, few database backups can be taken and frequent archive log backups can be taken to avoid placing database in backup mode for each database backup.
- Manual restore and recovery of database
  Restoring and recovering the full database backups when the required archive log files do not exist in the active file system was a difficult task for DBAs as they have to manually identify which backup contains the archive log files, mount the database backups, and recover the restored database. Seamlessly being a time-consuming and a long-running process to recover the critical database, DBAs found it as a tough task to recover database.

- Space constraints due to storage destinations becoming full
  When a database backup is created, the archive log destinations become full causing the database not to respond until sufficient space is created on the storage. Now with SnapManager 3.2 for Oracle, the archive log files can be pruned from the active file system to free up the space periodically.

Why archive log backups are important and how SnapManager 3.2 for Oracle provides solution to create archive log backups

Archive log files are necessary files required to roll the database forward once a restore is performed. Every transaction on an Oracle database is captured in the archive log files (if the database is in the archive log mode). Using the archive log backups, DBAs can restore the database backups.

Advantages of archive logs-only backups

Creating the archive logs-only backups provides several benefits:

- Separate retention duration for archive logs-only backups.
  Some organizations can retain backups for longer duration for which recovery is not required. Archive log files for such backups may not be required. So, users can prefer to have less retention duration for the archive logs-only backups that are actually required for recovery.
- Protect the archive logs-only backups based on archive log protection policies.
  Users can select different protection policies for archive logs-only backups based on their requirement.
- Improves the performance of the database
  Taking more archive logs-only backups and less datafile backups increases the performance of the database.
- Consolidates archive log backups
  SnapManager consolidates the archive log backups every time you take a backup by freeing up the duplicate archive log backups.

Administrators can perform tasks with the SnapManager graphical user interface or by using the command line interface. The *SnapManager for Oracle Installation and Administration Guide* explains how to complete these tasks using commands. The SnapManager online Help explains how to complete the tasks using the graphical user interface.

About database backups

Administrators can perform the following backup tasks:

- Create backups on primary storage.
- Create protected backups on secondary storage resources.
- Verify that backups completed successfully.
- View a list of backups.
- Schedule backups using the graphical user interface.
- Manage the number of backups retained.

- Free backup resources.
- Mount and unmount backups.
- Delete backups.

SnapManager creates backups using one of the following backup retention classes:

- Hourly
- Daily
- Weekly
- Monthly
- Unlimited

The N series Management Console data protection capability must be installed to use backup protection to secondary storage.

A backup can have one of these protection states: not requested, not protected, or protected.

If new datafiles are added to the database, Oracle recommends that you make a new backup immediately. Also, if you restore a backup taken before the new datafiles were added and attempt to recover to a point after the new datafiles were added, the automatic Oracle recovery process may fail, because it is unable to create datafiles. See the Oracle documentation for the process for recovering datafiles added after a backup.

Related concepts

Protection states on page 34

About protected backups on secondary storage

By default, SnapManager creates backups using Snapshot copies on the primary storage system. You can optionally enable data protection on the profile to protect backups on secondary storage systems. When using data protection on backups, administrators can see backups that are not conforming to their policy requirements and rectify those immediately.

Using SnapManager, DBAs can manage retention and scheduling of backups on primary storage, initiate restores from secondary storage, and create a clone of a backup on secondary storage. In contrast, storage administrators manage protected backups on secondary storage using the N series Management Console data protection capability.

Backing up data to secondary storage provides these benefits:

- Preserves the data in case of a disaster.
- Increases the limit on the number of potential backups.
  If you back up only to primary storage, the number of backups is limited by the number of Snapshot copies that can be taken on a single volume.
- Enables database clones on separate storage.

Note: SnapManager supports data protection on Linux, Solaris, HP-UX, and AIX platforms but not on Windows platform.

If protection is enabled on the profile, all full backups will be protected automatically (partial backups will not be protected).

You can mount and clone a database from secondary storage if the backup has been freed; however, you cannot verify a backup from secondary storage.

About enabling backup protection in the profile

Within the database profile, you can enable backup protection to secondary storage resources.

To create a protected backup of a database on secondary storage resources, DBAs and storage administrators perform the following steps:

Task	Details
Create or edit a profile.	In the profile, do the following: • Enable backup protection to secondary storage. • If the N series Management Console data protection capability is installed, protection policies appear in the graphical user interface. Select the policy for the profile. SnapManager creates a dataset associated with the profile. When backup protection is enabled, SnapManager creates a dataset for the database. A dataset is a collection of user data (plus the replicas of that data) that SnapManager manages as a single unit. The data is identified by the volume, qtree, or directory in which the dataset is located. If the administrator disables protection for a database, SnapManager deletes the dataset.
View the profile.	Because the storage administrator has not yet assigned storage resources to implement the protection policy, the profile shows up as "non-conformant" in both the SnapManager graphical user interface and in the `profile show` command output.
Assign storage resources in the N series Management Console data protection capability.	In the N series Management Console data protection capability, the storage administrator views the unprotected dataset and assigns a resource pool for each node of the dataset that is associated with the profile. The storage administrator then ensures that secondary volumes are provisioned and protection relationships are initialized by the N series Management Console data protection capability.

Task	Details
View the conformant profile in SnapManager.	In SnapManager, the database administrator then sees that the profile has changed to "conformant" in both the graphical user interface and in the `profile show` command output indicating that resources were assigned.
Create the backup.	• Select full backup. Protection is not allowed on partial backups. • Also select whether the backup should be protected and select the retention class (for example, hourly or daily).
View the backup.	The new backup shows as scheduled for protection, but not yet protected (in the SnapManager interface and in the `backup show` command output).
View the backup list.	After the storage administrator ensures that the backup has been copied to secondary storage, SnapManager changes the backup Protection State from "Not protected" to "Protected" in both the graphical user interface and with the `backup list` command.

How SnapManager determines which backups to retain on local storage

Using SnapManager, database administrators create backups that meet retention policies, which specify how many successful backups on local storage should be retained to support business requirements. The retention strategy involves several scenarios to handle regulatory requirements, loss of data, and disasters. SnapManager lets you specify how many successful backups it should retain in the profile for a given database. The retention policy is engaged every time you create a new backup.

An administrator might require the following backups for a production payroll database:

- 10 days of daily backups on primary storage
- 2 months of monthly backups on primary storage
- 7 days of daily backups on secondary storage
- 4 weeks of weekly backups on secondary storage
- 6 months of monthly backups on secondary storage

For each profile in SnapManager, administrators can change any value for any non-limited retention classes:

- Hourly
- Daily
- Weekly
- Monthly

SnapManager determines whether a backup should be retained by considering both the retention count (for example, 15 backups) and the retention duration (for example, 10 days of daily backups). A backup expires when its age exceeds the retention duration set for its retention class or the number of backups exceed the retention count. For example, if the backup count is 15 (meaning that SnapManager has taken 15 successful backups) and the duration requirement is set for 10 days of daily backups, the five oldest successful eligible backups expire.

After a backup expires, SnapManager either frees or deletes the expired backup. SnapManager always keeps the last backup taken.

SnapManager counts only the number of successful backups as eligible for the retention count and does not consider the following in that count:

Backups not included in the retention count	**Additional details**
Failed backups	SnapManager keeps information on both successful and unsuccessful backups. Although unsuccessful backups take up only minimal space in the repository, you might want to delete them. Unsuccessful backups remain in the repository until you delete them.
Backups designated to be retained on an unlimited basis or backups for a different retention class than the one for this count	SnapManager does not delete backups designated to be retained on an unlimited basis. Additionally, SnapManager considers only those backups in the same retention class (for example, SnapManager considers only the hourly backups for the hourly retention count).
Backups mounted from local storage	When Snapshot copies are mounted, they are also cloned and so are not considered eligible for retention. SnapManager cannot delete the Snapshot copies if they are cloned.
Backups that are used to create a clone on local storage	SnapManager keeps all backups used to create clones, but does not consider them for the backup retention count.
Backups that are cloned or mounted on secondary storage and that use the protection policy of "mirror"	If SnapManager did delete the Snapshot copies for the backup on the primary storage resource and it is mirrored, the next backup to secondary storage would fail.

When you free a backup from its primary storage resources, the primary resources (Snapshot copies) used by the backup are destroyed, but the backup metadata is still available. SnapManager does not consider freed backups eligible for deletion, so freed backups *are not* considered in the backup retention count.

SnapManager provides a default retention count and duration for each retention class. For example, for the hourly retention class count, SnapManager by default retains four hourly backups. Administrators can override these defaults and set the values when creating or updating the profile or they can change the default values in a SnapManager configuration file (`smo.config` file).

The default values for retention count and duration are stored in the `smo.config` file.

Backups on primary storage can be protected to secondary storage. While SnapManager manages the retention and scheduling of backups on primary storage, the N series Management Console data protection capability manages the retention and scheduling of backups on secondary storage.

When local backups expire based on their retention policy, they are either deleted or freed, depending on whether they are protected:

- If they are protected, the local backups will be freed, meaning that their storage resources or Snapshot copies will be deleted, but the backups will remain in the SnapManager repository and will be available for restoration from secondary storage. You do not have to manually free backups (for example, with the backup free command). Backups remain freed until the backup no longer exists on secondary, at which point the backup is deleted.
- If they are not protected, the local backups will be deleted.

The following example shows the actions that SnapManager takes on various types of backups based on a retention policy set to retain three daily backups (with the count set to retain 3):

Backup date	**Status**	**Retention policy action taken**	**Explanation**
5/10	Successful	Keep	This is the most recent successful backup, so it will be kept.
5/9	Successful, cloned	Skip	Because SnapManager does not consider backups used for cloning in the retention policy count, this backup is omitted from the count of successful backups.
5/8	Successful, mounted	Skip	Because SnapManager does not consider mounted backups in the retention policy count, this backup is omitted from the count of successful backups.
5/7	Failed	Skip	Failed backups are not counted.
5/5	Successful	Keep	SnapManager keeps this second successful daily backup.
5/3	Successful	Keep	SnapManager keeps this third successful daily backup.
5/2	Successful	Delete	SnapManager counts this successful backup, but after SnapManager reaches three successful daily backups, SnapManager deletes this backup.

About full and partial backups

You can choose to back up the entire database or just a portion of it. If you choose to back up a portion of the database, you can choose to back up a group of tablespaces or a group of data files. You can choose to take a separate backup of both tablespaces and datafiles. Likewise, you can choose to restore the entire database or just a portion of it.

The following table lists the benefits and consequences of each type of backup:

Backup type	Advantages	Disadvantages
Full backup	Minimizes the number of Snapshot copies. SnapManager takes one Snapshot copy for each volume that the database uses, plus one Snapshot copy for each volume that the log files occupy.	For online backups, each tablespace is in backup mode for the entire time of the backup operation.
Partial backup	Minimizes the amount of time each tablespace spends in backup mode. SnapManager groups the Snapshot copies it takes by tablespace. Each tablespace is in backup mode only long enough to take the Snapshot copies. This method of grouping the Snapshot copies minimizes the physical block writes in the log files during an online backup.	Because the backup could require taking Snapshot copies of multiple tablespaces in the same volume, this method could cause SnapManager to take multiple Snapshot copies of a single volume during the backup operation. As a result, SnapManager might take more Snapshot copies.

Note: Although you can perform a partial backup, it is recommended that you always perform a full backup of the entire database.

Backup types and the number of Snapshot copies

The backup type (full or partial) affects the number of Snapshot copies that SnapManager creates. For a full backup, SnapManager takes a Snapshot copy of each volume, while for a partial backup, SnapManager takes a Snapshot copy of each tablespace file.

Note: Data ONTAP limits the maximum number of Snapshot copies to 255 per volume. You might reach this maximum only if you configure SnapManager to retain a large number of backups where each backup consists of numerous Snapshot copies.

To keep an adequate pool of backups available while reducing the risk of reaching the maximum limit of Snapshot copies per volume, remove backups when they are no longer needed. You can

configure the SnapManager retention policy to remove successful backups after reaching a specific threshold for a specific backup frequency. For example, after SnapManager creates four successful daily backups, SnapManager removes the previous daily backups.

The following tables show how SnapManager creates Snapshot copies based on the backup type. The example in both tables assumes that databaseZ includes two volumes, each volume includes two tablespaces (TS1 and TS2), and each tablespace includes two database files (ts1_1.dbf, ts1_2.dbf, ts2_1.dbf, and ts2_2.dbf).

These tables show how the two types of backups produce a different number of Snapshot copies.

For a full backup, SnapManager takes a Snapshot copy of each volume, resulting in a total of two Snapshot copies. SnapManager takes Snapshot copies at the volume level instead of the tablespace level, which usually reduces the number of Snapshot copies it must take.

Note: Both backups would also take Snapshot copies of the log files.

Table 1: Full backup using SnapManager for Oracle

Volumes in database	Tablespace TS1 (includes 2 database files)	Tablespace TS2 (includes 2 database files)	Snapshot copies taken	Total Snapshot copies
/vol/volA	TS1_1.dbf	TS2_1.dbf	1 per volume	2
/vol/volB	TS1_2.dbf	TS2_2.dbf	1 per volume	

For a partial backup of the same number of database objects, SnapManager takes a Snapshot copy of each tablespace file, resulting in four Snapshot copies.

Table 2: Partial backup using SnapManager for Oracle

Volumes in database	Tablespace TS1 (includes 2 database files)	Tablespace TS2 (includes 2 database files)	Snapshot copies taken	Total Snapshot copies
/vol/volA	TS1_1.dbf	TS2_1.dbf	2 per file	4
/vol/volB	TS1_2.dbf	TS2_2.dbf	2 per file	

Full online backups

For a full backup, SnapManager backs up the entire database and takes Snapshot copies at the volume level (not at the tablespace level).

SnapManager creates two Snapshot copies for each backup. If all of the files needed by the database are in a single volume, then both Snapshot copies show in that volume.

When you specify a full backup, SnapManager performs these actions:

1. Places the entire database in the online backup mode.
2. Takes Snapshot copies of all the volumes containing database files.
3. Takes the database out of the online backup mode.
4. Forces a log switch and then archives the log files. This also flushes the redo information to disk.
5. Generates backup control files.
6. Takes a Snapshot copy of the log files and the backup control files.

When performing a full backup, SnapManager places the entire database in the online backup mode. An individual tablespace (for example, `/vol/vola/ts1_1.dbf`) is in the online backup mode longer than if certain tablespaces or data files were specified.

When a database goes into backup mode, Oracle writes entire blocks to the logs and does not merely write the delta between backups. Because databases do more work in online backup mode, choosing a full backup places a greater load on the host.

Although performing full backups places a greater load on the host, full backups require fewer Snapshot copies, resulting in fewer storage requirements.

Partial online backups

Instead of a full backup, you can choose to perform a partial backup of the tablespaces in a database. While SnapManager takes a Snapshot copy of volumes for *full* backups, SnapManager takes a Snapshot copy of each specified tablespace for *partial* backups.

Because the tablespace level is the lowest level that Oracle allows into backup mode, SnapManager processes backups at the tablespace level, even if you specify a data file in a tablespace.

With a partial backup, each tablespace exists in backup mode for a shorter amount of time compared to a full backup. During an online backup, the database is always available to users; however, the database must perform more work and the host must perform more physical I/O. In addition, because it is taking Snapshot copies of each tablespace specified or each tablespace containing a specified data file instead of the entire volume, SnapManager takes more Snapshot copies.

SnapManager takes Snapshot copies of specific tablespaces or data files. The partial backup algorithm is a loop that SnapManager repeats until it has taken a Snapshot copy of each specified tablespace or data file.

Note: Although you can perform a partial backup, it is recommended that you always perform a full backup of the entire database.

During a partial backup, SnapManager performs these actions:

1. Places the tablespace containing the data files into backup mode.
2. Takes a Snapshot copy of all the volumes used by the tablespace.
3. Takes the tablespace out of backup mode.
4. Continues this process, until it has taken a Snapshot copy of all the tablespaces or files.

5. Forces a log switch and then archives the log files.
6. Generates backup control files.
7. Takes a Snapshot copy of the log files and the backup control files.

Examples of backup, restore and recover operations

This example describes some of the backup, restore, and recover scenarios that you can use to accomplish your data protection goals.

In the following example, the database has two tablespaces. Tablespace1 has one table and two database files associated with it. Tablespace2 has two tables and one database file associated with it.

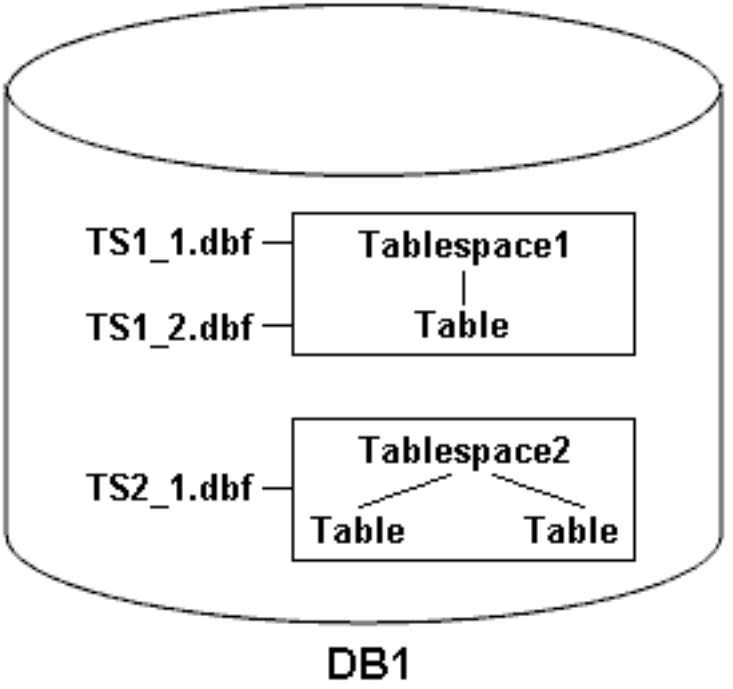

Figure 5: Backup of tablespaces and database files

There are several options for creating backups of this data and then restoring and recovering the data. The following table describes some backup, restore, and recover scenarios:

Examples of full backup, restore, and recover operations

Backup	Restore	Recover
Full backup SnapManager makes a backup of database DB1, including the data files, archive logs, and control files.	**Complete restore with control files** SnapManager restores all data files, tablespaces, and control files in the backup.	**Specify one of the following:** • SCN. Enter an SCN, such as 384641. • Date/Time. Enter a date and time of the backup, such as, 2005-11-25:19:06:22. • To the last transaction made to the database.
Full backup SnapManager makes a backup of everything in database DB1, including the data files, archive logs, and control files.	**Complete restore without control files** SnapManager restores all tablespaces and data files, without the control files.	**Specify one of the following:** • SCN. Enter an SCN, such as 384641. • Date/Time. Enter a date and time of the backup, such as, 2005-11-25:19:06:22. • To the last transaction made to the database.
Full backup SnapManager makes a backup of everything in database DB1, including the data files, archive logs, and control files.	**Restore either data files or tablespaces with control files** Specify either: • Tablespaces. Specify any of the tablespaces. SnapManager restores the tablespaces specified and the control files. • Data files. Specify any of the data files. SnapManager restores the data files specified and the control files.	SnapManager recovers the data to the last transaction made to the database

Backup	Restore	Recover
Full backup SnapManager makes a backup of everything in database DB1, including the data files, archive logs, and control files.	**Restore either data files or tablespaces without control files** SnapManager restores either: • Tablespaces. Specify any of the tablespaces. SnapManager restores only the tablespaces specified. • Data files. Specify any of the database files. SnapManager restores only the data files specified.	SnapManager recovers the data to the last transaction made to the database.
Full backup SnapManager makes a backup of everything in database DB1, including the data files, archive logs, and control files.	**Restore control files only** Restore only the control files.	SnapManager recovers the data to the last transaction made to the database.

Examples of partial backup, restore, and recover operations

Backup	Restore	Recover
Partial backup Specify the data that SnapManager puts in the backup. Choose one of the following options: • Tablespaces. Specify Tablespace1 and Tablespace2 or only one of them. • Data files. Specify all three database files (TS1_1.dbf, TS1_2.dbf, and TS2_1.dbf), two of these files, or one of these files. Regardless of which option you choose, the backup includes the archive logs and control files.	**Complete restore** SnapManager restores all data files, tablespaces, and control files specified in the partial backup.	SnapManager recovers the data to the last transaction made to the database instance.

Backup	Restore	Recover
Partial backup Specify the data SnapManager puts in the backup. Choose one of the following options: • Tablespaces. Specify both Tablespace1 and Tablespace2 or only one of them. • Data files. Specify all three database files (TS1_1.dbf, TS1_2.dbf, and TS2_1.dbf), two of these files, or one of these files. Regardless of which option you choose, the backup includes the archive logs and the control files.	**Restore either data files or tablespaces with control files** SnapManager restores either: • All of the data files specified. SnapManager restores only the data files specified and the control files. • All of the tablespaces specified. SnapManager restores the tablespaces specified and the control files.	SnapManager recovers the data to the last transaction made to the database instance.
Partial backup: Specify the data SnapManager puts in the backup. Choose one of the following options: • Tablespaces. Specify both Tablespace1 and Tablespace2 or only one of them. • Data files. Specify all three database files (TS1_1.dbf, TS1_2.dbf, and TS2_1.dbf), two of these files, or one of these files. Regardless of which option you choose, the backup includes the archive logs and the control files.	**Restore either data files or tablespaces without control files** SnapManager restores either: • Tablespaces. Specify any of the tablespaces. SnapManager restores only the tablespaces specified. If the backup contains Tablespace1, SnapManager restores only that tablespace. • Data files. Specify any of the database files. SnapManager restores only the data files specified. If the backup contains database files (TS1_1.dbf and TS1_2.dbf), SnapManager restores only those files.	SnapManager recovers the data to the last transaction made to the database instance.

Backup	Restore	Recover
Partial backup Specify the data SnapManager puts in the backup. Choose one of the following options: • Tablespaces. Specify both Tablespace1 and Tablespace2 or only one of them. • Data files. Specify all three database files (TS1_1.dbf, TS1_2.dbf, and TS2_1.dbf), two of these files, or one of these files. Regardless of which option you choose, the backup includes the archive logs and control files.	**Restore control files only** Specify to restore only the control files.	SnapManager recovers the data to the last transaction made to the database instance.

About control file and archive log file handling

The database uses control files to identify names, locations, and sizes of the database files. Because control files assist in the restore process, SnapManager includes control files with each backup.

Oracle tracks changes to a database using the online redo logs, which are eventually archived and then known as archived redo logs (or archive logs). Although SnapManager does not currently manage or restore archive logs, SnapManager includes them with each backup and uses them for cloning from online backups. SnapManager requires that archiving be enabled before creating an online backup of the database. SnapManager does not remove archive logs that have been backed up and does not restore them. If you need to manage archive logs, you can use RMAN.

Note: Before restoring a database backup, if archive logs are missing, copy them from the backup back to the original location.

Moving locations of Oracle control files:

```
select * from v$controlfile;
create pfile from spfile;
shutdown;
**move control files**
**Using VI Editor, edit files init<SID>.ora to reflect new location of
control files**
startup;
mount;
create spfile from pfile;
```

```
shutdown;
startup show parameter spfile;
```

Note: SnapManager includes the control files and archive log files with each backup. To see which tablespaces and data files are included in a backup, use the Backup Properties window or the `backup show` command.

The following table illustrates how SnapManager handles control and log files during each operation:

Type of operation	Control files	Archived log files
Backup	Included with each backup	Included with each backup
Restore	You can restore control files along with the restoration of the database, tablespaces, or data files. Or, you can restore only the control files.	Are not restored

About database backup scheduling

You can schedule, update, and monitor backups for databases using the Schedule tab of the graphical user interface. The following table addresses some common scheduling questions:

Question	Answer
What happens to the scheduled backups when the SnapManager server stops and restarts?	When the SnapManager server restarts, it automatically restarts all the schedules. However, SnapManager does not attempt to catch up on any missed occurrences.

Question	Answer
If two backups are scheduled to occur on two databases at the same time, what happens?	SnapManager starts each backup operation one at a time and then allows the backups to run in parallel. For example, if a DBA creates six daily backup schedules for six different database profiles all to occur at 1:00 AM, all six backups will be running in parallel within two minutes. If multiple backups are scheduled to occur on a single database profile in a short period of time, the SnapManager server will run only one backup operation--the one with the longest retention duration. Before starting a backup operation that was created by a schedule, SnapManager first determines the following: • Within the last 30 minutes, has another schedule successfully created a backup, with greater retention, for the same profile? • Within the next 30 minutes, will another schedule attempt to create a backup, with greater retention, for the same profile? If the answer to either question is yes, SnapManager skips the backup. (Backups created on demand are not considered.) For example, a DBA might create a daily, weekly, and a monthly schedule for a database profile, all of which are scheduled to take backups at 1:00 AM. On that one day of the month when three backups are scheduled to occur simultaneously at 1:00 AM, SnapManager runs only the backup operation based on the monthly schedule. The time window of 30 minutes can be changed in a SnapManager properties file.
Under which user does the backup operation run?	The operation runs as the user who created the schedule. However, you can change the "run as user" for a scheduled backup to be your own user ID, if you have valid credentials for both the database profile and database's host. For instance, by launching the Scheduled Backup Properties for the backup schedule created by Avida Davis, Stella Morrow can select her user ID in the "Perform this operation as user" box to run the scheduled backup herself.
How does the SnapManager scheduler interact with the native operating system scheduler?	On the SnapManager server, the user will not be able to view the scheduled backups via the operating system's native scheduler. For instance, after creating a scheduled backup, the Windows user will not see a new entry in the Scheduled Tasks window.

Question	Answer
What happens if the clocks on the graphical user interface and the server are not in sync?	The clocks on the client and server are not synchronized. Therefore, you could schedule a backup in which the start time is in the future on the client in the graphical user interface but in the past on the server. For recurring backups, the server still fulfills the request. For instance, if the server receives a request to perform hourly backups starting an 01/30/08 at 3:00 PM but the current time is 3:30 PM on that day, the server performs its first backup at 4:00 PM and continues to perform backups every hour. However, for one-time only backups, the server handles the request as follows: • If the start time is within the last five minutes of the current server time, SnapManager immediately begins the backup. • If the start time is greater than five minutes, SnapManager does not initiate the backup. For instance, consider the following scenario: • The clock on the graphical interface host is three minutes behind. • The current time on the client is 8:58 AM. • You schedule a one-time backup to occur at 9:00 AM. • You schedule another one-time backup to occur at 8:30 AM. When the server receives the first request, the time on the server is 9:01 AM. Although the start time of the backup is in the past, SnapManager immediately performs the backup. When the server receives the second request, the start time of the backup is more than five minutes in the past. You will receive a message that the schedule request failed because the start time is in the past. You can change the time of five minutes in a SnapManager properties file.
What happens to the scheduled backups for a profile when the profile is deleted?	When a database profile is deleted, the SnapManager server deletes scheduled backups defined for that profile.

Question	Answer

How do scheduled backups behave during Daylight Savings Time or when you change the SnapManager server time manually?	SnapManager backup schedules get affected due to Daylight Savings Time or when you have manually changed the SnapManager server time.
	Consider the following implications when the SnapManager server time is changed:
	• After the backup schedule is triggered, if the SnapManager sever time falls back, then the backup schedule will not trigger again.
	• If Daylight Saving Time spring forward before the schedule start time, the backup schedules are triggered automatically.
	• For example, if you are in the United States and you schedule hourly backups at 4 AM that should occur every 4 hours, backups will occur at 4 AM, 8 AM, noon, 4 PM, 8 PM, and midnight on the days before and after Daylight Savings Time adjustments in March and November.
	• If backups are scheduled for 2:30 AM every night, then:
	  • When the clocks fall back an hour, as the backup is already triggered, the backup will not trigger again.
	  • When the clocks spring forward an hour, the backup will trigger immediately.
	If you are in the United States and want to avoid this issue, schedule your backups to start outside the 2:00 AM to 3:00 AM interval.

Creating database backups

About this task

SnapManager leverages Snapshot technology to create fast and space-efficient backups of databases. These backups are point-in-time virtual copies of the database and are stored on the same physical medium of the database (local backups) or on secondary storage (remote backups). You can back up entire databases or portions of databases, including tablespaces, data files, or control files.

SnapManager provides these capabilities for databases across many host side storage stacks, including NFS, ASM, Veritas, and others.

Note: For RAC configurations, SnapManager performs the backup on the host set in the profile.

Administrators can optionally register backups with Oracle RMAN, which facilitates the use of RMAN to restore and recover the database at finer granularities such as blocks.

While defining the profile, you can customize the names of the Snapshot copies created by backups of that profile. For example, you might insert a prefix string of "HOPS" to denote High Operations backups.

In addition to defining unique names for Snapshot copies created by backups, you can also create unique labels for the backups themselves. When you create a backup, it is a good practice to supply a name for the backup so you have an easy way to identify it by using the -label parameter. This name must be unique for all backups created within a particular profile. The name can contain letters, numbers, underscore(_), and hyphen(-). It cannot start with a hyphen. Labels are case-sensitive. You might want to append information such as operating system environment variables, system date, and backup type.

If you do not supply a label, SnapManager creates a default label name in the form *scope_mode_datestring*, where scope is full or partial and mode is offline, online, or automatic (the letter *c* for cold, *h* for hot, or *a* for automatic).

When you enter a comment, you can include spaces and special characters. In contrast, when you enter a label, do not include spaces or special characters.

For each backup, SnapManager automatically generates a GUID, which is a 32-character HEX string. To determine the GUID, run the `backup list` command with the `-verbose` option.

You can create a full backup of a database while it is online or offline. To let SnapManager handle backing up a database regardless of whether it is online or offline, use the `-auto` option.

You can create a cold backup when the database is in the shutdown state. If the database is in a mounted state, change it to a shutdown state and perform the offline backup (cold backup).

SnapManager 3.2 for Oracle enables you to back up the archive logs files separately from the datafiles, thus helps you to manage the archive log files efficiently.

For creating the archive log backups separately, you must create a new profile or update the existing profile to separate the archive log backups using the `-separate-archivelog-backups` option. Using the profile, you can perform the following SnapManager operations:

- Create archive log backup
- Delete archive log backup
- Mount archive log backup
- Free archive log backup

The backup options vary depending on the profile settings:

- Using a profile that is not separated to take archive log backups separately, you can do the following:
  - Create a full backup
  - Create a partial backup
  - Specify archive log destinations to be backed up for archive log files
  - Specify archive log destinations to be excluded from the backup
  - Specify the pruning options for deleting the archive log files from the archive log destinations

- Using a profile that is separated to take archive log backups, you can do the following:
  - Create datafiles-only backup
  - Create archive logs-only backup
  - While creating a datafiles-only backup:
    - Include the archive log backup along with the online datafiles only backup for cloning.

    If you have included archive log backups along with datafiles in the **Profile Settings** page of the **Profile Create** wizard from the SnapManager GUI, and if you have not selected the **Archivelogs** option in the **Backup Create** wizard, SnapManager always creates the archive log backup along with datafiles for all online backups.
    In such a situation, from the SnapManager CLI, you can consider all the archive log destinations for backup except for the exclude destinations specified in the SnapManager configuration file. But you cannot prune these archive log files. However, you can still use the `-archivelogs` option to specify the archive log file destination and prune the archive log files from the SnapManager CLI.
    If you are creating the backup using the `-auto` option and specify the `-archivelogs` option, SnapManager creates either an online or offline backup based on the current status of the backup.
    - SnapManager creates an offline backup when the database is offline and does not include the archive log files in the backup.
    - SnapManager creates an online backup including archive log files when the database is online.
  - While creating the archive logs-only backup:
    - Specify the archive log destination to be backed up along with the archive logs-only backup
    - Specify the archive log destinations to be excluded from the archive logs-only backup
    - Specify the pruning options for deleting the archive log files from the archive log destinations
- **Scenarios not supported**
  - You cannot create the archive logs-only backup along with an offline datafiles-only backup.
  - You cannot prune the archive log files when the archive log files are not backed up.

  Note: While creating archive log backups on Windows platform, you must enter the full archive log destinations paths within double quotes and the destination paths separated by commas. The path separator should be given as two backslashes (\\) instead of one. For example, each destination path should be provided as: "J:\\mnt\\oradata_nfs_share\\oradata\\, E:\\mnt\\archive_dest_1\\".

When you specify the label for online datafiles backup with included archive log backup, the label is applied for datafiles backup, and the archive log backup will be suffixed with (`_logs`). This suffix can be configured by changing the parameter `suffix.backup.label.with.logs` parameter in the SnapManager configuration file.

For example, you can specify the value as `suffix.backup.label.with.logs=arc` so that the _logs default value is changed to `_arc`.

If the user has not specified any archive log destinations to be included in the backup, then SnapManager includes all the archive log destinations configured in the database.

If any archive log files are missing in any one of the destinations, SnapManager skips all these archive log files created before the missing archive log files even if these files are available in other archive log destination.

While creating archive log backups, specify the archive log file destinations to be included in the backup, and you can set the configuration parameter to include the archive log files always beyond the missing files in the backup.

Note: By default, this configuration parameter is set to true, to include all the archive log files, beyond missing files. If you are using your own archive log pruning scripts or manually deleting archive log files from the archive log destinations, you can disable this parameter, so that SnapManager can skip the archive log files and proceed further with the backup.

SnapManager 3.2 for Oracle does not support the following SnapManager operations for archive log backups:

- Clone the archive log backup
- Restore archive log backup
- Verify archive log backup

SnapManager also supports backing up the archive log files from the flash recovery area destinations.

Step

1. To create a backup of a database, enter this command:

```
smo backup create -profile profile_name {[-full {-online | -offline | -auto} [-retain {-hourly | -daily | -weekly | -monthly | -unlimited}] [-verify] | [-data [[-files files [files]] | [-tablespaces -tablespaces [-tablespaces]] [-datalabel label] {-online | -offline | -auto} [-retain {-hourly | [-daily | -weekly | -monthly | -unlimited]} [-verify] | [-archivelogs [-label label] [-comment comment] [-protect | -noprotect | -protectnow] [-backup-dest path1 [,[path2]]] [-exclude-dest path1 [,path2]]] [-prunelogs {-all | -untilSCN untilSCN | -until-date yyyy-MM-dd:HH:mm:ss | -before {-months | -days | -weeks | -hours}} -prune-dest prune_dest1,[prune_dest2]] [-taskspec taskspec]} [-dump] [-force] [-quiet | -verbose]
```

To do the following...	Then...
To specify whether you want to take a backup of an online or offline database, rather than allowing SnapManager to handle whether it is online or offline	Specify `-offline` to take a backup of the offline database. Specify `-online` to take a backup of the online database. If you use these options, you cannot use the `-auto` option.

To do the following...	Then...
To let SnapManager handle backing up a database regardless of whether it is online or offline	Specify the `-auto` option. If you use this option, you cannot use the `--offline` or `-online` option.
To perform a partial backup of specific files	Specify the `-data -files` option and then list the *`files`*, separated by commas. For example, list the file names f1,f2,f3 after the option. Example for creating a partial datafile backup on Windows `smo backup create -profile nosep -data -files "J:\\mnt\user\user.dbf" -online -label partial_datafile_backup -verbose`
To perform a partial backup of specific tablespaces	Specify the `-data -tablespaces` option and then list the *`tablespaces`*, separated by commas. For example, use ts1,ts2,ts3 after the option. Example for creating a partial tablespace backup `smo backup create -profile nosep -data -tablespaces tb2 -online -label partial_tablespace_bkup -verbose`
To create a unique label for each backup in the following format: full_hot_mybackup_label	For Windows, you might enter this example: `smo backup create -online -full -profile targetdb1_prof1 -label full_hot_my_backup_label   -verbose`

To do the following...	Then...
To create backup of the archive log files separately from the datafiles	Specify the following options and variables: • `-archivelogs` option to create backup of the archive log files. • `-backup-dest` specifies the archive log file destinations to be backed up. • `-exclude-dest` specifies the archive log destinations to be excluded. • `-label` specifies the label for the archive log file backup. • `-protect` enables protection to the archive log backups. **Note:** You must provide either the `-backup-dest` option or the `-exclude-dest` option. Providing both these options together along with the backup displays error message `You have specified an invalid backup option. Specify any one of the options: -backup-dest, or exclude-dest`. Example for creating archive log file backups separately on Windows `smo backup create -profile nosep -archivelogs -backup-dest "J:\\mnt \archive_dest_2\" -label archivelog_backup -verbose`
To create backup of datafiles and archive log files together	Specify the following options and variables: • `-data` option to specify the datafiles. • `-archivelogs` option to specify the archive log files. Example for backing up datafiles and archive log files together on Windows `smo backup create -profile nosep -data -online -archivelogs -backup-dest "J:\\mnt \archive_dest_2\" -label data_arch_backup -verbose`

To do the following...	Then...

To prune the archive log files while creating a backup	Specify the following options and variables:
	• `-prunelogs` specifies to delete the archive log files from the archive log destinations.
	  • `-all` specifies to delete all the archive log files from the archive log destinations.
	  • `-until-scn` *`until-scn`* specifies to delete the archive log files until a specified SCN.
	  • `-until-date` *`yyyy-MM-dd:HH:mm:ss`* specifies to delete the archive log files until the specified time period.
	  • `-before` option specifies to delete the archive log files before the specified time period (days, months, weeks, hours).
	  • `-prune-dest` *`prune_dest1,[prune_dest2`* specifies to delete the archive log files from the archive log destinations while creating the backup.
	Example for pruning all archive log files while creating a backup on Windows
	`smo backup create -profile nosep -archivelogs -label archive_prunebackup1 -backup-dest "E:\\oracle\\MDV\\oraarch\\MDVarch,J:\\" -prunelogs -all -prune-dest "E:\\oracle\\MDV\\oraarch\\MDVarch,J:\\" -verbose`

To add a comment about the backup	Specify `-comment` followed by the description string.
To force the database into the state you have specified to back it up, regardless of the state it is currently in	Specify the `-force` option.
To verify the backup at the same time you create it	Specify the `-verify` option.
You can collect the dump files after the database backup operation.	Specify `-dump` option at the end of the backup create command.

Example

```
smo backup create -profile targetdb1_prof1 -full -online -force  -verify
```

Related concepts

Snapshot copy naming on page 106

Related tasks

Creating pre-task, post-task, and policy scripts for SnapManager operations on page 230
Creating task scripts for SnapManager operation on page 243
Installing the task scripts on page 244
Protecting database backups to secondary storage when SnapManager is not integrated with Protection Manager on page 159

Related references

The smo backup create command on page 264

Pruning archive log files

SnapManager 3.2 for Oracle supports deleting the archive log files from the archive log destinations while taking a backup.

Before you begin

- Archive log files should be backed up along with the current backup.
- The database should be in the `MOUNTED` state, if not provide the `-force` option along with backup.

About this task

SnapManager provides the pruning options along with backup:

- Specify prune archive logs along with the full backup or archive logs-only backup:
  - Pruning is performed only when the archive log files are backed up by the current backup operation. If pruning is specified along with other backups that do not contain archive log files, the archive log files will not be pruned.
- Specify scope for pruning–either all the logs, until a SCN, or until a specified time period. Using these options, you can perform the following tasks:
  - Delete all archive log files.
  - Delete the archive log files until the specified SCN.
  - Delete the archive log files until the specified time.
  - Delete the archive log files before the specified time period.
- Specify destination to prune – from where the archive log files should be deleted.

SnapManager checks for the following before deleting the archive log files:

- Archive log files are backed up at least once.
- Archive log files are shipped to Oracle Dataguard Standby database, if any.
- Archive log files are captured by Oracle streams capture process, if any.

SnapManager ignores the files that are not backed up, not shipped to standby, and not captured by capture process, and deletes the other files.

- If there are any archive log files that are not backed up or not shipped to standby, or not captured by the capture process, SnapManager deletes the archive log files one-by-one, and this process consumes more time. SnapManager will group the archive log files and deletes them batch-by-batch. Each batch will have a maximum of 998 files. This value can be configured below 998 using a configuration parameter `maximum.archivelog.files.toprune.atATime` in the SnapManager configuration file (`smo.config`).
- If the archive log files are backed up, shipped to standby, and captured by the capture process, SnapManager deletes all the archive log files in a single execution and this process is faster.

SnapManager uses RMAN commands to delete the archive log files. But SnapManager does not integrate with the RMAN retention policies and deletion policies.

Even when the archive log file pruning fails in one destination, SnapManager continues to prunes the archive log files from the other destinations

Note: If user manually deletes the archive log files from the archive log destinations, the pruning of archive log files fail.

Scenarios not supported for pruning

- Pruning of archive log files from the flash recovery area destination is not supported.
- Pruning of archive log files from the standby database is not supported.
- Using both SnapManager and RMAN for managing archive log files is not a supported scenario.

Step

1. To prune the archive log backups while creating the backup, enter the following command:

```
smo backup create -profile profile_name {[-full {-online | -offline | -auto} [-retain {-hourly | [-daily | -weekly | -monthly | -unlimited}] [-verify] | [-data [[-files files [files]] | [-tablespaces -tablespaces [-tablespaces]] [-datalabel label] {-online | -offline | -auto} [-retain {-hourly | [-daily | -weekly | -monthly | -unlimited]} [-verify] | [-archivelogs [-label label] [-comment comment] [-protect | -noprotect | -protectnow] [-backup-dest path1 [,[path2]]] [-exclude-dest path1 [,path2]]] [-prunelogs {-all | -untilSCN untilSCN | -until -date yyyy-MM-dd:HH:mm:ss | -before {-months | -days | -weeks | -hours}} -prune-dest prune_dest1,[prune_dest2]] [-taskspec taskspec]} -dump [-force] [-quiet | -verbose]
```

To do the following...	Then...

To prune archive log files	Specify the following options and variables:
	• `-prunelogs` specifies to delete the archive log files while creating a backup.
	  • `-all` specifies to delete all the archive log files.
	  • `--untilSCN` specifies to delete the archive log files until the specified SCN.
	  • `-until -date` specifies to delete the archive logs including the specified date and time.
	  • `-before {-months | -days | -weeks | -hours}` specifies to delete archive log files before the specified time period.

To include destination from which the archive log files are pruned	Specify the `-prune-dest` option.

Consolidating archive log backups

SnapManager consolidates the archive logs-only backups every time you take a backup by freeing up the duplicate archive logs-only backups. By default, consolidation is enabled.

About this task

SnapManager identifies the archive logs-only backups which has archive log files in other backups and frees them to maintain minimum number of archive logs-only backups with unique archive log files.

If the archive logs-only backups are freed by consolidation, then these backups are deleted based on the archive log retention duration.

When the database is in the shutdown or nomount state during archive log consolidation, SnapManager changes the database to the mount state.

If the backup or pruning of archive log files fails, then consolidation will not be done. Consolidation of archive logs-only backups is followed only after successful backups and successful pruning operations.

Steps

1. To enable consolidation of the archive logs-only backups, modify the configuration parameter `consolidation` and set the value as `true` in the SnapManager configuration file (smo.config).

   Once the parameter is set, the archive logs-only backups are consolidated.

   If the newly-created archive logs-only backup contains the same archive log files in any of the earlier archive logs-only backups, then the earlier archive-log only backups are freed.

   Note: SnapManager does not consolidate the archive log backup taken along with the datafiles backup. SnapManager consolidates the archive logs-only backup.

Note: SnapManager consolidates the archive log backups even when user manually deletes the archive log files from the archive log destinations or when the archive log files are corrupted and might be included the backup.

2. To disable consolidation of the archive log backups, modify the configuration parameter `consolidation` and set the value as `false` in the SnapManager configuration file (smo.config).

Scheduling archive log file pruning

SnapManager enables you to schedule the pruning of archive log files from the archive log destinations to occur in the specified time and frequency while creating a backup.

About this task

SnapManager allows you to prune the archive log files periodically from the active file system.

Step

1. To schedule the pruning of archive log files, enter the following command:

```
smo schedule create -profile profile_name {[-full {-online | -offline |
-auto}[-retain [-hourly | -daily | -weekly | -monthly | -unlimited] [-
verify]] | [-data [-files files [files]] | [-tablespaces -tablespaces [-
tablespaces]] {-online | -offline | -auto}[-retain [-hourly | -daily | -
weekly | -monthly | -unlimited] [-verify]] | [-archivelogs]} [-comment
comment] [-protect | -protectnow | -noprotect] [-backup-dest path1 [,
[path2]] [-exclude-dest path1 [,path2]] [-prunelogs{-all | -untilSCN
untilSCN | -before {-date yyyy-MM-dd HH:mm:ss | -months months | -weeks
weeks | -days days | -hours hours}} -prune-dest
prune_dest1,,prune_dest2] -schedule-name schedule_name [-schedule-
comment schedule_comment] -interval {-hourly | -daily | -weekly | -
monthly | -onetimeonly} -cronstring cronstring -start-time {start-time
start_time <yyyy-MM-dd HH:mm>} -runasuser -runasuser [-force] [-quiet |
-verbose]
```

To do the following...	Then...
To schedule pruning of archive log files	Specify the following options and variables: • `-prunelogs` to schedule pruning of the archive log files. • `-prune-dest` to prune archive log files from the archive log destinations.
To include a name for the schedule	Specify the `-schedule-name` option.

To do the following...	Then...
To schedule pruning of archive log files at specific time interval	Specify the `interval` option and indicate whether the archive log files should be pruned based on the interval classes: • -hourly • -daily • -weekly • -monthly • -onetimeonly
To add a comment about the schedule operation	Specify the `-schedule-comment` option followed by the description string.
To specify the start time of the schedule operation	Specify the `-start-time` option in yyyy-MM-dd HH:mm format.

Protecting archive log backups

SnapManager enables you to protect the archive log backups while creating profiles. You can protect the archive log backups based on the archive log protection policy.

Step

1. To protect archive log backups while creating a profile, enter this command:

```
smo profile create -profile profile [-profile-password profile_password]
-repository -dbname repo_dbname -host repo_host -port repo_port -login -
username repo_username -database -dbname db_dbname -host db_host [-sid
db_sid] [-login -usernamedb_username -password db_password -port
db_port] [-rman {-controlfile | {-login -username rman_username -
password rman_password -tnsname rman_tnsname} } ] -osaccount osaccount -
osgroup osgroup [-retain [-hourly [-count n] [-duration m]] [-daily [-
count n] [-duration m]] [-weekly [-count n] [-duration m]] [-monthly [-
count n] [-duration m]]] [-comment comment][-snapname-pattern pattern][-
protect [-protection-policy policy_name]] [-summary-notification] [-
notification [-success -email email_address1, email_address2 -subject
subject_pattern] [-failure -email email_address1, email_address2 -
subject subject_pattern]][-separate-archivelog-backups -retain-
archivelog-backups -hours hours | -days days | -weeks weeks| -months
months [-protect [-protection-policy policy_name] | -noprotect] [-
include-with-online-backups | -no-include-with-online-backups]] [-dump]
```

If...	Then...
You want to backup archive log backups separately, and protect the archive log files	Specify these options and variables: • `-separate-archivelog-backups` enables you to separate the archive log files from the datafile files. • `-protect` sets separate protection policy for the archive log archive log backups. • `-protection-policy` sets the protection policy for the archive log backups.

Verifying database backups

You can use the verify feature to confirm that physical blocks in the backup of the database have not been corrupted. This feature invokes the Oracle Database Verify utility for each data file in the backup.

About this task

SnapManager lets you perform the verify operation at any time that is convenient for you and the users on your system. Perform the verification when you create the backup, immediately after making the backup, or verify the operation later.

Specify the profile containing the backup and either the label or ID of the backup you created.

You can specify to collect the dump files after the backup verify operation.

Step

1. To verify a backup that you have already created, enter this command format:

```
smo backup verify -profile profile_name [-label label | -id id] [-force]
[ -dump] [-quiet | -verbose]
```

Related references

The smo backup verify command on page 283

Changing the backup retention policy

You can change properties of a backup so it is eligible or ineligible for deletion according to the retention policy.

About this task

When you create a backup, you can set its retention policy. You can later choose to either keep that backup for a longer period than the retention policy allows or specify that you no longer need the backup and want the retention policy to manage it.

Retaining backups forever

You can specify that a backup should be ineligible for deletion by the retention policy to keep the backup indefinitely.

Step

1. To specify that a backup be retained on an unlimited basis, enter this command:

```
smo backup update -profile profile_name {-label label [data | -
archivelogs] | -id id} -retain -unlimited
```

Related references

The smo backup update command on page 281

Assigning backups with a specific retention class

DBAs can assign a specific retention class of hourly, daily, weekly, or monthly to backups. Assigning a specific retention class makes the backups performed under this change eligible for deletion.

Step

1. To assign a specific backup retention class, enter this command:

```
smo backup update -profile profile_name {-label label [data | -
archivelogs] | -id id | all} -retain [-hourly | -daily | -weekly | -
monthly]
```

Changing the retention policy default behavior

When a backup expires based on the retention policy, SnapManager determines whether to delete or free a backup based on whether it is protected. You can change the default behavior of deleting unprotected backups when they expire to freeing them instead.

About this task

By default, SnapManager deletes or frees backups depending on whether they are protected as follows:

- For protected backups, SnapManager frees the local backups when they expire.
- For unprotected backups, SnapManager deletes the local backups when they expire. You can change this default behavior.

For protected backups, SnapManager does not consider the following in determining whether to delete the local copy:

- Whether the backup to secondary storage failed or is in process of being protected. This enables the transfer of backups to secondary storage to occur before the application of the retention policy.
- Whether the copy was subsequently deleted from secondary storage.

Steps

1. Access the following default location:

   `<default smo installation location>/properties/smo.config`

2. Edit the `smo.config` file.
3. Change the default behavior so that SnapManager frees unprotected backups that expired by setting the `retain.alwaysFreeExpiredBackups` property in the `smo.config` file to `true`:

   `retain.alwaysFreeExpiredBackups = true`

Freeing or deleting retention policy exempt backups

Backups with the retention class of "unlimited" cannot be deleted or freed directly. To delete or free these backups, you must first assign another retention class, such as hourly, daily, weekly, or monthly. To delete or free a backup that is exempt from the retention policy, you must first update the backup to make it eligible for deletion or free it.

Steps

1. To update the backup to make it eligible for deletion by the retention policy, enter this command:

   `smo backup update -profile profile_name {-label label [data | -archivelogs] | -id id} -retain [-hourly | -daily | -weekly | -monthly]`

2. After updating the backup so it is eligible for deletion, you can either delete the backup or free backup resources.
   - To delete the backup, enter this command:

     ```
     smo backup delete -profile profile_name {-label label [data | -
     archivelogs] | -id id | -all}
     ```

   - To free the backup resources, rather than delete the backup, enter this command:

     ```
     smo backup free -profile profile_name {-label label [data | -
     archivelogs] | -id id | -all} [-force] [ -dump] [-quiet | -verbose]
     ```

Viewing a list of backups

You can quickly check on which backups were made for a profile and the backup state using the `smo backup list` command. For each profile, the command displays the information on the most recent backup first and then continues until information on all backups is displayed.

About this task

The list displays information about all backups for a specified profile. It lists the following information for each backup:

- Start date and time of the backup
- Status of the backup (success or failure)
- Scope of the backup (full, partial, data, logs)
- Mode of the backup (online or offline)
- Storage that indicates the storage resources status of the backup (exists or freed)
- Label of the backup

If the `-verbose` option is specified, it also displays the following information:

- Retention class for the backup
- ID for the backup
- Comment entered when creating the backup
- Whether the backup is protected to secondary storage
- Whether the backup exists on primary storage
- Checkpoint SCN of the database backup
- End point of the online database backup
- Starting SCN of the earlier archive log file in the backup
- Ending SCN of the latest archive log file in the backup

Step

1. To display information about backups in a particular profile, enter this command:

```
smo backup list -profile profile_name [-delimiter character] [data | -
archivelogs] [-quiet | -verbose]
```

Related references

The smo backup list command on page 270

Viewing backup details

You can display detailed information about a particular backup in a profile by using the `smo backup show` command.

About this task

This operation displays the following information for each backup:

- The backup ID
- Whether the backup succeeded or failed
- Backup scope - full, partial, online, offline
- Backup mode
- Mount status
- The backup label
- Comment
- The date and time the operation started and ended
- Information about whether the backup was verified
- The backup retention class
- The database and host name
- The checkpoint SCN
- The end backup SCN (for online backups only)
- The tablespaces and data files from the database backed up
- The control files from the database backed up
- The archive logs from the database backed up
- The storage system and volumes where the files are located
- The Snapshot copies made and their location
- The status of the primary storage resources
- The backup protection status
- A list of copies on secondary storage, in the form of backup_copy ID - node name
- Backup mode

If you specify the `-verbose` option, the following additional information appears:

- The clones made from the backup, if there are any
- Verification information

- If the backup is mounted, SnapManager displays the mountpoints in use

For the archive log file backup, the same information is displayed as that of the other database backup except an additional archive log files section that contains the following information:

- The first change number of the backup
- The next change number of the backup
- Thread number
- Reset logs ID
- Incarnation
- Log file name

SnapManager does not provide the following information for the archive log file backups:

- Checkpoint SCN
- End Backup SCN
- Tablespace
- Control Files

Step

1. To display information about a backup in a specific profile, enter this command:

   ```
   smo backup show -profile profile_name {-label label [data | -
   archivelogs] | -id  id [-quiet | -verbose]
   ```

Related references

The smo backup show command on page 279

Mounting backups

SnapManager automatically handles mounting a backup to make it available to the host. You can mount backups from either primary or secondary storage. SnapManager enables you to mount backups for situations where you are using an external tool, such as RMAN, to access the files in the backup.

About this task

If you are using RMAN, use the mount operation to change the state of a backup (which allows access) and the unmount operation to change the state of a backup (which removes access).

The `smo backup mount` command displays a list of paths where the Snapshot copies comprising the backup have been mounted.

To mount the backup from secondary storage, use the `-from-secondary` option. If you do not use this option, SnapManager mounts the backup from primary storage.

SnapManager enables you to mount the archive log backups from either the primary or secondary storage.

You can optionally collect the dump files after the successful or failed backup mount operation.

Step

1. To mount a backup, enter this command:

```
smo backup mount -profile profile_name {label label [data | -
archivelogs] | -id id} [-host -host] [-from-secondary [-copy-id id]] [-
dump] [-quiet | -verbose]
```

Related references

The smo backup mount command on page 271

Unmounting backups

SnapManager automatically unmounts the backup to make it unavailable to the host. SnapManager provides the unmount operation for situations where you are using an external tool, such as RMAN, to access the files in the backup, and you need to change the state of a backup to remove access.

About this task

You can optionally collect the dump files after the successful or failed backup unmount operation.

Step

1. To unmount a backup, enter this command:

```
smo backup unmount -profile profile_name {label label [data | -
archivelogs] | -id id} [-quiet | -verbose] -dump-force -verbose
```

Related references

The smo backup unmount command on page 280

Freeing backups

You can free eligible backups, which deletes the Snapshot copies without deleting the backup metadata. This frees the space the backup occupied. If you remove backups you no longer need, you

reduce the chance of reaching the limit of 255 Snapshot copies per volume. Use the `smo backup free` command to free the backup resources by deleting just the Snapshot copies.

About this task

For a backup to be eligible for freeing, the backup must:

- Have been successful
- Not be mounted
- Not have clones
- Not be retained using an unlimited retention policy
- Not already be freed

If protection is enabled on the profile and the protection policy contains connections from the primary node that use a mirror relationship, then when Snapshot copies are deleted on the primary node by freeing a backup, those Snapshot copies are also deleted from the mirror nodes when the next transfer to secondary occurs.

When you free a protected backup, SnapManager requests the Protection Manager to remove the local Snapshot copies for the backup. If the backup free operation is successful for the protected backups, the Snapshot copies are deleted by Protection Manager in an asynchronous manner. Protection Manager marks these Snapshot copies for deletion and deletes the Snapshot copies.

Protection state	Local status	Action on primary storage	Action on secondary storage	Explanation
Not requested (to be protected)	Exists	Frees the backup	No action required	SnapManager frees the local backup.
	Freed	No action required	No action required	The local backup is already freed.
Not protected	Exists	Frees the backup	No action required	SnapManager frees the local backup even though no copies exist on secondary storage.
	Freed	No action required	No action required	The local backup is already freed.
Protected	Exists	Frees the backup	No action required; the backup on secondary remains	SnapManager frees the local backup. Copies remain on secondary storage.
	Freed	No action required	No action required	The local backup is already freed.

SnapManager enables you to free the eligible archive log backups if the backups are within the archive log backup retention duration and the archive log files are required for the recovery of the database.

You can specify the `-dump` option as an optional parameter to collect the dump files after the successful or failed backup free operation.

Step

1. To free the backup resources, enter this command:

```
smo backup free -profile profile_name {-label label [data | -
archivelogs] | -id id | -all} -force [-dump] [-quiet ] [-force]
```

Related concepts

Protection states on page 34

Related references

The smo backup free command on page 269

Deleting backups

You should delete eligible backups when you no longer need them, which frees the space that the backup occupied. If you remove backups you no longer need, you reduce the chance of reaching the limit of 255 Snapshot copies per volume.

About this task

For a backup to be eligible for deletion, the backup must not have been used to create a clone.

When you delete a protected backup, SnapManager deletes the backup from secondary storage, and deletes the backup from the SnapManager repository. The following table shows the actions taken on both the primary and secondary storage when you delete a local backup:

Protection state	Local status	Action on primary storage	Action on secondary storage	Explanation
Not requested (to be protected)	Exists	Deletes the Snapshot copies	No action required	SnapManager deletes the local backup.
	Freed	No action required	No action required	The local backup is already freed. Deleting a freed backup removes the backup metadata from the repository.

Protection state	Local status	Action on primary storage	Action on secondary storage	Explanation
Not protected	Exists	Deletes the Snapshot copies	No action required	SnapManager deletes the local backup whether or not it has been protected.
	Freed	No action required	No action required	The local backup is already freed. Deleting a freed backup removes the backup metadata from the repository.
Protected	Exists	Deletes the Snapshot copies	SnapManager deletes the backup on secondary storage	SnapManager deletes the local backup and secondary copies.
	Freed	No action required	SnapManager frees the backup on secondary storage	SnapManager deletes the local backup and secondary copies.

If you attempt to delete a backup that is protected to secondary storage, the Snapshot copies might be marked for deletion and are deleted later by Protection Manager though SnapManager deletes the backup from the repository. If you cannot immediately delete the backup, it is recommended that you wait until another backup occurs before you delete that backup.

You can delete backups retained on an unlimited basis without changing the retention class.

You can optionally collect the dump files after the successful or failed backup delete operation.

If you want to delete the archive log backups, you need to check for the retention duration set for the archive log backup. The archive log backups are controlled by the archive log retention duration. If the archive log backup is within the retention duration and the archive log files are required for recovery of a restored database, you cannot delete the archive log backup.

Steps

1. Before deleting the backup, verify that operations are complete by entering this command:

   ```
   smo operation list -profile profile_name -quiet -verbose
   ```

2. To delete a backup, enter this command:

   ```
   smo backup delete -profile profile_name [-label label [data | -
   archivelogs] | -id id | -all] [-force] [ -dump] [-quiet | -verbose]
   ```

   Use the `-force` option to force the removal of the backup. Forcing the removal of a backup that has incomplete operations might leave the backup in an inconsistent state.

Managing AutoSupport messages

In SnapManager 3.2 for Oracle, AutoSupport (ASUP) messages are enabled by default.

About this task

AutoSupport enables the SnapManager server to initiate the ASUP messages from the storage system. ASUP messages are sent from storage system for every successful backup operation.

If you no longer need the ASUP messages, perform the following steps:

Steps

1. Stop the SnapManager server.
2. Browse to the SnapManager installation directory and locate the properties directory that contains the SnapManager configuration file (smo.config).
3. Add or set the value of `auto_support.on` variable to *`false`* in the smo.config file:

   `auto_support.on=false`
4. Restart the SnapManager server.

Protecting database backups to secondary storage

By default, SnapManager creates backups using Snapshot copies on the primary storage system. You can optionally enable data protection on the profile to protect backups on secondary storage systems. When using data protection on backups, administrators can see backups that are not conforming to their policy requirements and rectify those immediately.

Starting from SnapManager 3.2 for Oracle, you can protect the backups in two ways:

- SnapManager integrated with Protection Manager: In this way, you can protect backups from both the CLI and GUI on the UNIX platforms. Protecting backups to secondary storage when SnapManager is integrated with Protection Manager is not supported on Windows platforms.
- SnapManager is not integrated with Protection Manager: When there are critical backups that you need to protect, you can use the post-backup scripts to protect those backups. These post-backup scripts are used for post-processing activity of the backup operation. Using this option, you can protect the backups from both CLI and GUI on the UNIX and Windows platforms.

Using SnapManager, DBAs can manage retention and scheduling of backups on primary storage, initiate restores from secondary storage, and create a clone of a backup on secondary storage. In contrast, storage administrators manage protected backups to secondary storage using the N series Management Console data protection capability.

Backing up data to secondary storage provides these benefits:

- Preserves the data in case of a disaster.
- Increases the limit on the number of potential backups. If you back up only to primary storage, the number of backups is limited by the number of Snapshot copies that can be taken on a single volume.
- Enables database clones on separate storage.

Note: SnapManager supports data protection on Linux, Solaris, HP-UX, and AIX platforms but not on Windows platform.

If protection is enabled on the profile, all backups (full, archive logs-only, datafiles-only, and archive log backups taken along with the datafile backup) will be protected automatically (partial backups will not be protected).

You can mount and clone a database from secondary storage if the backup has been freed; however, you cannot verify a backup from secondary storage.

Protecting database backups to secondary storage when SnapManager is not integrated with Protection Manager

SnapManager 3.2 for Oracle has the ability to protect database backups from primary storage system to secondary storage system using scripts when SnapManager is not integrated with Protection

Manager and there is no SnapMirror or SnapVault relationship established between the primary and secondary storage systems. These scripts are built-in with SnapManager and you can use these scripts for post-processing activity of the backup operation from both the SnapManager CLI and GUI on the Windows and UNIX-based environments.

Before you begin

Before using the scripts, ensure you establish the SnapMirror and SnapVault relationships between the primary and secondary storage systems:

- The SnapMirror relationship for the requested secondary storage volumes must be configured in the secondary storage system.
- The SnapVault relationship for the requested secondary storage qtrees must be configured in secondary storage system.

About this task

SnapManager for Oracle supported scripts

SnapManager 3.2 for Oracle supports two post-processing scripts to protect backups from primary storage system to secondary storage system after the backup operation occurs:

- Mirror the backup Mirror_the_backup.cmd
- Vault the backup Mirror_the_backup.cmd

SnapManager for Oracle versions prior to 3.2 provided the ability to use the pre-processing or post-processing scripts only for clone operation. From SnapManager 3.2 for Oracle, the pre-processing and post-processing scripts are provided for the backup and the restore operations. You can use these scripts to run before or after the backup or restore operations occur.

Note: The mirror the backup and vault the backup scripts are provided for reference only. They have been tested with SnapDrive 6.2 or 6.3 for Windows but may not work in all environments. Please review and then customize based on your secondary protection requirements. These scripts will not work in the SnapDrive versions lower than SnapDrive 6.2 or 6.3 for Windows.

How to use scripts for post-processing activity of backup operation

To use the scripts for the post-processing activity of the backup operation, perform the following steps:

1. Create a new script or use the available script.
2. Add the script name and required inputs in the post-processing task specification XML file.

For additional information about creating task specification, refer to "Creating task specification and scripts for SnapManager operations".

Mirror the backup script

In the UNIX-based environment, you must provide two input parameters in the task specification XML file.

- Secondary storage name
- Secondary volume name

If the data is spread across different storage systems, then you must enter the storage system names and volume names separated by a comma. Ensure you do not provide any space between the comma and the next storage system name and volume name.

In Windows environment, there is no necessity to provide any inputs in the task specification XML file.

Vault the backup script

In the UNIX-based environment, you must provide two input parameters in the task specification XML file.

- Secondary storage name
- Secondary qtree name

If the data is spread across different storage systems, then you must enter the storage system names and qtree names separated by a comma. Ensure you do not provide any space between the comma and the next storage system name and qtree name.

In Windows environment, there is no necessity to provide any inputs in the task specification XML file.

Sample Scripts

The following sample script mirrors the backup on a Windows environment. It includes the three operations (check, describe, and execute) and calls them at the end of the script. The script also includes error message handling with codes of 0 to 4 and >4.:

```
@echo off
REM $Id: //depot/prod/capstan/main/src/plugins/windows/examples/
backup/create/post/Mirror_the_backup.cmd#1 $
REM
REM Copyright (c) 2011 Org, Inc.
REM All rights reserved.
REM
REM
REM This is a sample post-task script to mirror the volumes to the
secondary storage after successful backup operation.
REM|-------------------------------------------------------------------
-------------------------------------------|
REM| Pre-requsite/
Assumption:
                            |
REM|   SnapMirror relationship for the requested secondary storage
volumes must be configured in Secondary storage. |
REM|-------------------------------------------------------------------
-------------------------------------------|
REM
REM
REM This script can be used from the SnapManager graphical user
interface (GUI) and command line interface (CLI).
```

```
REM
REM To execute the post-task script for the backup operation from
SnapManager GUI, follow these steps:
REM  1. From the Backup wizard > Task Specification page > Post-Tasks
tab > select the post-task scripts from the Available Scripts section.
REM
REM
REM To execute the post-task script for the backup operation from
SnapManager CLI, follow these steps:
REM  1. create a task specification XML file.
REM  2. Add the post-script name in the <post-tasks> tag of the XML
file.
REM
REM Example:
REM              <post-tasks>
REM                 <task>
REM                    <name>Mirror the backup</name>
REM                      <description>Mirror the backup</description>
REM               </task>
REM              <post-tasks>
REM
REM
REM
REM IMPORTANT NOTE: This script is provided for reference only. It
has been tested with SnapDrive 6.3.0 for Windows but may not work in
all environments.  Please review and then customize based on your
secondary protection requirements.
REM
set /a EXIT=0
set name="Mirror the backup"
set description="Mirror the backup"
set parameter=()


if /i "%1" == "-check" goto :check
if /i "%1" == "-execute" goto :execute
if /i "%1" == "-describe" goto :describe

:usage
        echo usage: %0 ^{ -check ^| -describe ^| -execute ^}
        set /a EXIT=99
        goto :exit

:check
        set /a EXIT=0
        goto :exit

:describe
        echo SM_PI_NAME:%name%
        echo SM_PI_DESCRIPTION:%description%
    echo SM_PRIMARY_MOUNT_POINTS : %SM_PRIMARY_MOUNT_POINTS%
        set /a EXIT=0
        goto :exit

REM - Split the comma-separated PRIMARY_MOUNT_POINTS and Mirror the
PRIMARY_MOUNT_POINTS one-by-one.
```

```
:execute
        set /a EXIT=0

    echo "execution started"

    REM FOR %%G IN (%SM_PRIMARY_MOUNT_POINTS%) DO echo %%G

    FOR %%V IN (%SM_PRIMARY_MOUNT_POINTS%) DO sdcli snap
update_mirror -d %%V

    if "%ERRORLEVEL%" NEQ "0" (
           set /a EXIT=4
        )

    echo "execution ended"

        goto :exit

:exit
        echo Command complete.
        exit /b %EXIT%
```

The following sample script vaults the backup on a Windows environment. It includes the three operations (check, describe, and execute) and calls them at the end of the script. The script also includes error message handling with codes of 0 to 4 and >4.:

```
@echo off
REM $Id: //depot/prod/capstan/main/src/plugins/windows/examples/
backup/create/post/Vault_the_backup.cmd#1 $
REM
REM Copyright (c) 2011 Org, Inc.
REM All rights reserved.
REM
REM
REM This is a sample post-task script to vault the qtrees to the
secondary storage after successful backup operation.
REM|-------------------------------------------------------------------
-------------------------------------------|
REM| Pre-requsite/
Assumption:
                            |
REM|   SnapVault relationship for the requested secondary storage
qtrees must be configured in Secondary storage. |
REM|-------------------------------------------------------------------
-------------------------------------------|
REM
REM
REM This script can be used from the SnapManager graphical user
interface (GUI) and command line interface (CLI).
REM
REM To execute the post-task script for the backup operation from
SnapManager GUI, follow these steps:
REM  1. From the Backup wizard > Task Specification page > Post-Tasks
tab > select the post-task scripts from the Available Scripts section.
REM
```

```
REM
REM To execute the post-task script for the backup operation from
SnapManager CLI, follow these steps:
REM  1. create a task specification XML file.
REM  2. Add the post-script name in the <post-tasks> tag of the XML
file.
REM
REM Example:
REM              <post-tasks>
REM                 <task>
REM                    <name>Vault the backup</name>
REM                    <description>Vault the backup</description>
REM               </task>
REM              <post-tasks>
REM
REM IMPORTANT NOTE: This script is provided for reference only. It
has been tested with SnapDrive 6.3.0 for Windows but may not work in
all environments.  Please review and then customize based on your
secondary protection requirements.
REM
REM
REM
REM
REM
set /a EXIT=0
set name="Vault the backup"
set description="Vault the backup"
set parameter=()



if /i "%1" == "-check" goto :check
if /i "%1" == "-execute" goto :execute
if /i "%1" == "-describe" goto :describe

:usage
        echo usage: %0 ^{ -check ^| -describe ^| -execute ^}
        set /a EXIT=99
        goto :exit

:check
        set /a EXIT=0
        goto :exit

:describe
        echo SM_PI_NAME:%name%
        echo SM_PI_DESCRIPTION:%description%
        echo SM_PRIMARY_SNAPSHOTS_AND_MOUNT_POINTS :
%SM_PRIMARY_SNAPSHOTS_AND_MOUNT_POINTS%
        set /a EXIT=0
        goto :exit

REM Split the colon-separated SM_PRIMARY_SNAPSHOTS_AND_MOUNT_POINTS
And SnapVault the mountpoints one-by-one

:execute
        set /a EXIT=0
```

```
    echo "execution started"

        FOR %%A IN (%SM_PRIMARY_SNAPSHOTS_AND_MOUNT_POINTS%) DO
FOR /F "tokens=1,2 delims=:" %%B IN ("%%A") DO sdcli snapvault
archive  -a %%B -DS %%C %%B

        if "%ERRORLEVEL%" NEQ "0" (
           set /a EXIT=4
        )
     echo "execution ended"

        goto :exit

:exit
        echo Command complete.
        exit /b %EXIT%
```

Creating a script for protecting database backups on secondary storage

You can use the scripts as examples to learn how to make your own or use as a basis for your new scripts. You can start a script from scratch or modify one of the SnapManager sample scripts.

About this task

SnapManager expects your script to be structured in a particular manner to support being executed within the context of a SnapManager operation. Create the script based on the expected operations, available input parameters, and return code conventions stated in this document.

Steps

1. To customize a sample script, do the following:
   a. Locate a sample script in the following SnapManager install directory:

      `<default_install_directory>/plugins/backup/create/post`

   b. Open the script in your script editor.
   c. Save it as your own custom script.
2. Modify the functions, variables, and parameters as needed.
3. Save your custom script in one of the following directory locations:

   `<default_install_directory>/plugins/backup/create/post`

   The custom script is executed after the backup operation occurs. Use it on an optional basis when you specify the backup creation.

Creating post-processing task specification for protecting database backups to secondary storage

SnapManager enables you to use scripts in the post-processing activity of the backup operation. You can include the script in the post-processing task specification XML file of the backup operation. Using the scripts, you can mirror or vault the backup to secondary storage.

Before you begin

Before using the scripts, ensure you establish the SnapMirror and SnapVault relationships between the primary and secondary storage systems:

- The SnapMirror relationship for the requested secondary storage volumes must be configured in the secondary storage system.
- The SnapVault relationship for the requested secondary storage qtrees must be configured in secondary storage system.

About this task

To execute the post-processing task specification file for the backup operation from the SnapManager CLI, perform the following steps:

Steps

1. Create a task specification XML file.
2. In the XML file, enter the secondary storage details as input parameters.
3. Save the task specification XML file.

Using post-processing task specification to mirror volumes on Windows

SnapManager for Oracle enables you to use the script to mirror the volumes after the backup operation occurs on a Windows environment.

About this task

To execute the post-processing task for the backup operation from the SnapManager CLI, perform the following steps:

Steps

1. Create a task specification XML file.
2. In the XML file, enter the script name as an input parameter.
3. Save the task specification XML file.

4. Create a protected backup of a database to secondary storage using the following command. While creating the protected backup, you must provide the complete path of the saved task specification XML file after the `-taskspec` option.

   Example: `smo backup create -profile test_profile -full -online -taskspec "C:\\mirror\\snapmirror.xml"`

The following example indicates post-processing task specification structure on Windows environment:

```
    <post-tasks>
                <task>
                   <name>Mirror the backup</name>
                    <description>Mirror the backup</description>
                    <parameter>
                </task>
            <post-tasks>
```

Using post-processing task specification to vault qtrees on Windows

SnapManager for Oracle enables you to use the script to vault the qtrees after the backup operation takes place on Windows environment.

About this task

To execute the post-task script for the backup operation from SnapManager CLI, follow these steps:

Steps

1. Create a task specification XML file.
2. In the XML file, enter the script name as an input parameter.
3. Save the task specification XML file.
4. Create a protected backup of a database to secondary storage using the following command. While creating the protected backup, you must provide the complete path of the saved task specification XML file after the `-taskspec` option.

   Example: `smo backup create -profile test_profile -full -online -taskspec "C:\\mirror\\snapvault.xml"`

The following example indicates post-processing task specification structure including script name on Windows environment:

```
          <post-tasks>
                <task>
                   <name>Vault the backup</name>
                    <description>Vault the backup</description>
                    <parameter>
```

```
                    </task>
                <post-tasks>
```

Scheduling database backups

SnapManager 3.2 for Oracle enables database administrators to schedule database backups to occur on a regular basis during off-peak time to maintain high performance. To schedule a backup, database administrators create a profile that indicates the database information and retention policy and then provide schedules for the backup.

Administrators can perform the following schedule-related tasks:

- Schedule a database backup to occur on an hourly, daily, weekly, monthly, or one-time only basis.
- View a list of scheduled backups associated with a profile.
- Update a scheduled backup.
- Suspend a schedule temporarily.
- Resume the suspended schedule.
- Delete a schedule.

Note: You must schedule the backups as an administrator in the Windows environment. If you try to schedule the backups as a non-existing user, SnapManager displays an error message: `Invalid user: username: Cannot create schedule backup for a given user`.

Creating backup schedules

You can schedule a backup to occur in the specified time and frequency that works best for your data and your environment.

About this task

From SnapManager 3.2 for Oracle, you can schedule the backups of the archive log files separately. But you must use the profile that have created to separate the archive log files.

If you have scheduled the backups of the datafiles and archive log files at the same time, then SnapManager creates the datafiles backup first followed by the archive log file backups.

While creating a scheduled backup, if you have selected the schedule interval as `-onetimeonly`, then all the pruning options are available. If you have selected schedule interval other than the `-onetimeonly` option, then the pruning options `-untilSCN` and `-until-date` are not supported and displays the error message: `"The archive log pruning option you have specified, -until-scn or -until-date for the schedule interval hourly is invalid. Specify either the -onetimeonly option for the schedule interval, or prune the archive logs using any one of the option all, or -before {-months | -days | -weeks| -hours}"`.

When a failover happens in a HACMP environment, you must restart the SnapManager for Oracle server so that the service (virtual) address will be mapped to the active host and the SnapManager

schedules are adjusted to the active SnapManager host. You can add this info in the pre-preprocessing or post-processing HACMP failover scripts.

Step

1. To schedule the backup, enter the following command:

```
smo schedule create -profile profile_name {[-full {-online | -offline | -auto}[-retain [-hourly | -daily | -weekly | -monthly | -unlimited] [-verify]] | [-data [-files files [files]] | [-tablespaces -tablespaces [-tablespaces]] {-online | -offline | -auto}[-retain [-hourly | -daily | -weekly | -monthly | -unlimited] [-verify]] | [-archivelogs]} [-comment comment] [-protect | -protectnow | -noprotect] [-backup-dest path1 [,[path2]] [-exclude-dest path1 [,path2]] [-prunelogs{-all | -untilSCN untilSCN | -until-date yyyy-MM-dd HH:mm:ss | -before {-months | -weeks | -days | -hours}} -prune-dest prune_dest1,prune_dest2] -schedule-name schedule_name [-schedule-comment schedule_comment] -interval {-hourly | -daily | -weekly | -monthly | -onetimeonly} -cronstring cronstring -start-time {start-time start_time <yyyy-MM-dd HH:mm>} -runasuser -runasuser [-force] [-taskspec -taskspec] [-quiet | -verbose]
```

To do the following...	Then...
To schedule a backup of an online or offline database	Specify the `-offline` option to schedule a backup of the offline database. Specify the `-online` option to schedule a backup of the online database. If you use these options, you cannot use the `-auto` option.
To let SnapManager handle scheduling of a database regardless of whether it is online or offline	Specify the `-auto` option. If you use this option, you cannot use the `--offline` or `-online` option.
To schedule a backup of datafiles	Specify the `-data -files` option and list the *`files`*, separated by commas. For example, list the file names f1,f2,f3 after the option.
To schedule a partial backup of specific tablespaces	Specify the `-tablespaces` option and list the *`tablespaces`*, separated by commas. For example, use ts1,ts2,ts3 after the option.
To schedule backup of archive log files	Specify the following options and variables: • `-archivelogs` to schedule backup of the archive log files. • `-backup-dest` to schedule archive log file destinations to be included in the backup. • `-exclude-dest` to schedule the archive log destinations to be excluded from the backup.

To do the following...	Then...
To specify the retention class values	Specify the `-retain` option and indicate whether the backup should be retained according to one of the following retention classes: • -hourly • -daily • -weekly • -monthly • -unlimited If you do not specify a retain option, SnapManager defaults to hourly.
To schedule pruning of archive log files	Specify the following options and variables: • `-prunelogs` to prune the archive log files while scheduling a backup. • `-prune-dest` specifies the archive log destination from which the archive log files are pruned.
To include a name for the schedule	Specify the `-schedule-name` option.
To schedule backup of the database at specific time interval	Specify the `interval` option and indicate the time interval by which the backups should be taken: • -hourly • -daily • -weekly • -monthly • -onetimeonly
To configure schedule using cronstring	Specify the `-cronstring` option and include the following seven sub-expressions that describe the individual option. Use cronstring to schedule the backup. • 1 refers to seconds • 2 refers to minutes • 3 refers to hours • 4 refers to a day in a month • 5 refers to the month • 6 refers to a day in a week • 7 refers to the year (optional) **Note:** If you have scheduled your backup with different time in the `-cronstring` and the `-start-time` options, then the schedule of the backup is overwritten and triggered by the `-start-time` option.
To add a comment about the backup schedule	Specify the `-schedule-comment` option followed by the description string.

To do the following...	Then...
To specify the start time of the schedule operation	Specify the `-start-time` option in yyyy-MM-dd HH:mm format.
To change the user of the scheduled backup operation while scheduling the backup	Specify the `-runasuser` option. The operation runs as the user (root user or Oracle user) who created the schedule. However, you can change the "run as user" for a scheduled backup to be your own user ID, if you have valid credentials for both the database profile and database's host.
To enable a pre-task or a post-activity of the backup schedule operation using the pre-task and post-task specification XML file	Specify the `-taskspec` option and provide the absolute path of the task specification XML file for performing a pre-processing or a post-processing activity to occur before or after the backup schedule operation.

Updating a backup schedule

You can view a list of scheduled operations and update them if necessary. You can update the scheduling frequency, the start time of the schedule, cronstring expression, and the user who scheduled the backup.

Step

1. To update the schedule for a backup, enter this command:

```
smo schedule update -profile profile_name -schedule-name schedulename [-
schedule-comment schedule comment] -interval {-hourly | -daily | -weekly
| -monthly | -onetimeonly} -start-time starttime -cronstring cronstring
-runasuser runasuser [-quiet | -verbose]
```

Viewing a list of scheduled operations

You can view a list of scheduled operations for a profile.

Step

1. To display information about scheduled operation, enter this command:

```
smo schedule list -profile profile_name [-quiet | -verbose]
```

Suspending backup schedules

SnapManager enables you to suspend a backup schedule until the backup schedule is resumed.

About this task

You can suspend the active schedules. If you try to suspend the backup schedule that is already suspended, you might encounter error message "`Cannot suspend: schedule <schedulename> already in suspend state`".

Step

1. To suspend the backup schedule temporarily, enter this command:

   `smo schedule suspend -profile` ***`profile_name`*** **`-schedule-name`** ***`schedulename`*** **`[-quiet | -verbose]`**

Resuming backup schedules

Administrators have the option to resume the suspended backup schedule.

About this task

If you try to resume the active schedules, you might encounter the error message: "`Cannot resume: schedule <schedulename> already in resume state`".

Step

1. To resume the suspended backup schedule, enter this command:

   `smo schedule resume -profile` ***`profile_name`*** **`-schedule-name`** ***`schedulename`*** **`[-quiet | -verbose]`**

Deleting backup schedules

You can delete backup schedules when they are no longer necessary.

Step

1. To delete the backup schedule, enter this command:

   `smo schedule delete -profile` ***`profile_name`*** **`-schedule-name`** ***`schedulename`*** **`[-quiet | -verbose]`**

Restoring database backup

SnapManager for Oracle provides the ability to restore a database to the state it was in at the time a Snapshot copy was created. In addition to its file-based restore process, SnapManager leverages volume-based fast restore technology, which shortens the restore time significantly compared to traditional recovery methods. Since backups can now be created more frequently, the number of logs that need to be applied is drastically reduced, thus reducing the mean-time-to-recovery (MTTR) for a database.

DBAs can perform several tasks related to restoring and recovering data in databases:

- Perform a file-based restore or a volume-based restore, which is the fastest method of restoring database backups and is the default that SnapManager uses.
- Restore the entire backup or a portion of it. If you restore a portion of it, specify a group of tablespaces or a group of data files. You can also restore the control files along with the data or just the control files themselves.
- Recover the data based on either a point in time or on all available logs (meaning, the last transaction committed to the database). The point in time can be an Oracle SCN or a date and time (yyyy-mm-dd:HH:MM:SS). SnapManager uses the 24-hour clock.
- Restore from backups on primary storage (also called local backups).
- Restore protected backups (also called remote backups) from secondary storage and choose how to copy the data back to the primary storage.
- Use SnapManager to both restore and recover the backup, or use SnapManager to restore the backup and use another tool, such as RMAN, to recover the data.
- Restore backups from alternate locations.

You can restore backup data taken by a previous version of SnapManager using SnapManager 3.0 and later versions.

SnapManager also provides the ability to restore ASM-based databases. An ASM disk group can be shared by multiple databases. As a result, you cannot simply revert to an older Snapshot copy of the disk group, because it would revert all the databases. Traditional restore solutions go through the host and require that all the blocks that constitute the database be moved from the storage system to the host and then back to the storage system. SnapManager relieves this overhead by providing the ability to restore just the required data within the ASM disk group without going through the host.

Note: SnapManager provides data protection and volume-based fast restores in UNIX-based environments only; SnapManager does not provide data protection or fast restores in a Windows environment.

Administrators can perform tasks with the SnapManager graphical user interface or by using the command-line interface. The *SnapManager for Oracle Installation and Administration Guide* explains how to complete these tasks using commands. The SnapManager online Help explains how to complete the tasks using the graphical user interface.

Related concepts

Backing up databases on page 117

Related references

The smo backup restore command on page 273

Database restore overview

SnapManager provides the ability to perform volume-based or file-based backup restore operations (which include three file-based methods).

The following table describes the restore methods:

Restore process	Details
Volume-based fast restores (from primary storage)	SnapManager restores the datafiles of a database by restoring a full volume. This default process is the fastest method for restoring your database.
File-based restores	Storage side full file system restore (from primary or secondary): SnapManager performs a full LUN restore.
	Storage side file restore: SnapManager performs a single file snap restore (SFSR) in a NAS environment or a partial file snap restore (PFSR) in an ASM environment. In an SFSR, the files or LUNS that represent the protected objects are restored. A PFSR is performed from the local backup if the file system details are understood and the file system layout has not changed since the previous backup was taken.
	Host side file copy restore (from primary or secondary): SnapManager clones the local backup using either a LUN or a FlexClone. The clone is mounted and then SnapManager copies the host files from the clone into the active file system.

Although SnapManager chooses the fast restore process by default, administrators can choose either type. If a fast restore is selected, SnapManager provides information about conditions that prevent the fast restore process from completing and about conditions that might affect the fast restore but which administrators can ignore if they choose to continue with the process.

Note: You cannot restore a backup from secondary storage, if the backup also exists on primary storage.

Upon completion of a fast restore, the following occurs:

- SnapManager frees more recent backups (taken after the backup was restored) in the same profile, because their Snapshot copies no longer exist on primary storage.
- SnapManager deletes all Snapshot copies for backups in the same profile that had any Snapshot copies automatically deleted by the fast restore process. This prevents backups from being

partially freed. For example, Backup_A was created first and Backup_B created next. Each has a Snapshot copy for the data files and one for the archive logs. After SnapManager restores Backup_A using the fast restore process, SnapManager automatically deletes the data file Snapshot copy from Backup_B. Because the archive log is not restored in the fast restore process, SnapManager must delete Backup_B's Snapshot copy of the archive logs after the fast restore process completes.

Fast restore

Fast restore or volume-based restore is so named because it is the fastest possible restore method. The entire storage system volume is reverted back to a Snapshot copy. At the storage level, this restore is almost instantaneous. However, performing a volume restore can have the following negative consequences, and therefore must be used with caution:

- The entire storage side volume is reverted, including
  - Files that were not considered part of the backup
  - Other files, file systems, or LUNs on the volume
- All the Snapshot copies that were taken after the Snapshot copy to which the volume is being reverted will be deleted. For example, you can no longer restore Tuesday's backup if you volume restored Monday's backup.
- Relationships to secondary storage systems will be broken if the restored Snapshot copy is older than the baseline Snapshot copy in the relationship.

Storage side file system restore

A storage side file system restore is performed when a volume restore cannot be performed, but the entire files system can be restored on the storage system.

When a storage side file system restore is performed, the following occurs, depending on the environment:

- In a SAN environment, all of the LUNs used by the file system (and underlying volume group if any) will be restored on the storage system.
- In a NAS environment, every file in the file system will be restored on the storage system.
  For NAS environments, this restore mechanism does not provide additional benefit over storage side file restore.

When a storage side file system restore is performed, the following occurs, depending on the storage location:

- When SnapManager is restoring from primary storage systems, the LUNs (SAN) or files (NAS) will be restored in place via single file snap restore.
- When SnapManager is restoring from secondary storage systems, the LUNs (SAN) or files (NAS) will be copied from secondary storage systems back to the primary storage system over the network.

Since the file system is fully restored, any files not part of the backup will be reverted as well. An override is required if files, which are not part of the restore, exist in the file system being restored.

Storage side file restore

A storage side file restore is sometimes performed when a storage side file system restore cannot be performed. A storage side file restore is when individual files within a file system are restored directly on the storage systems.

This type of restore can be performed only in NFS environments.

For ASM environments, storage side file restore can be performed only if the following conditions apply:

- Underlying file extents have not changed since the backup was taken (for example, the file was not resized and disk rebalancing has not occurred).
- You are restoring from primary storage systems. (It is not supported when restoring from secondary storage systems.)

When a storage side file restore is performed, the following occurs:

- When SnapManager is restoring NFS files from primary storage systems, the individual files are restored in place using single file snap restore.
- When SnapManager is restoring NFS files from secondary storage systems, the individual files are copied back to the primary storage system over the storage network.
- When restoring ASM files from primary storage systems, the individual files are restored in place by restoring only the bytes in the underlying LUNs associated with the files being restored (the rest of the bytes in the LUNs remain intact). The storage system technology used for restoring LUNs partially is called "partial file snap restore."

Host side file restore

A host side file copy restore is used as a last resort in SAN environments when fast restore, storage side file system restore, and storage side file restore cannot be performed.

A host side file copy restore involves the following tasks:

- Cloning the storage
- Connecting the cloned storage to the host
- Copying files out of the clone file systems back into the active file systems using host copy utilities (for example, the 'cp' command)
- Disconnecting the clone storage from the host
- Deleting the clone storage

When restoring from secondary, SnapManager first attempts to restore data directly from the secondary storage system to the primary storage system (without involving the host). If SnapManager cannot perform this type of restore (for example, if files not part of the restore exist in a file system), then SnapManager will fall back to a host side file copy restore. SnapManager has two

methods of performing a host side file copy restore from secondary. The method SnapManager selects is configured in the `smo.config` file.

- Direct: SnapManager clones the data on secondary storage, mounts the cloned data from the secondary storage system to the host, and then copies data out of the clone into the active environment. This is the default secondary access policy.
- Indirect: SnapManager first copies the data to a temporary volume on primary storage, then mounts the data from the temporary volume to the host, and then copies data out of the temporary volume into the active environment. This secondary access policy should be used only if the host does not have direct access to the secondary storage system. Restores using this method will take twice as long as the "direct" secondary access policy because two copies of the data are made.

The decision whether to use direct or indirect is controlled by the value of the `restore.secondaryAccessPolicy` parameter in the `smo.config` configuration file. The default is "direct."

Backup recovery

The SnapManager recover option is part of the restore operation, which means you must perform the restore and recover at the same time if you are using SnapManager. You cannot perform a restore operation and then later perform a SnapManager recover operation.

Using SnapManager versions prior to 3.2 version, you can use SnapManager to restore and recover the backup, or you can use SnapManager to restore the backup and use another tool, such as Oracle Recovery Manager (RMAN), to recover the data. Because SnapManager can register its backups with RMAN, DBAs can use RMAN to restore and recover the database at finer granularities such as blocks. This integration combines benefits of the speed and space efficiency of Snapshot copies with the fine level of control for restoring RMAN.

Note: You can use any tool or script to recover a database. You must recover the database before you can use it.

Starting from SnapManager 3.2 for Oracle, SnapManager provides the ability to restore the database backups automatically using the archive log backups. Even when the archive log backups are available in the external location, SnapManager uses the archive log backups from the external location to restore the database backups.

If new datafiles are added to the database, Oracle recommends that you take a new backup immediately. Also, if you restore a backup taken before the new datafiles were added and attempt to recover to a point after the new datafiles were added, the automatic Oracle recovery process may fail, because it is unable to create datafiles. See the Oracle documentation for the process for recovering datafiles added after a backup.

Database state needed for restore process

The state that the database needs to be in to be restored depends on the following factors:

- Whether you are doing a restore only or a restore with recovery

- The type of files that should be included in the restore process, specifically control files, data files containing the SYSTEM tablespace, or any other data files not containing the SYSTEM tablespace

The following table lists the state in which the database should be depending on the restore option selected and the type of files you want to include in the restore:

Type of restore	Files included	Database state for this instance	Database state for other instance (RAC only)
Restore only	Control files	Shutdown	Shutdown
	System files	Mount or Shutdown	Mount or Shutdown
	No system files	Any state	Any state
Restore and recovery	Control files	Shutdown	Shutdown
	System files	Mount	Mount or Shutdown
	No system files	Mount or Open	Any

The database state required by SnapManager for a restore operation depends on the type of restore being performed (complete, partial, or control files). SnapManager does not transition the database to a lower state (for example, OPEN to MOUNTED) unless the option is specified.

Restore preview plans

SnapManager for Oracle presents a restore plan when you are previewing a restore operation and after a restore operation is completed.

This topic provides information about the following:

- The structure of the restore plan
- The restore methods that SnapManager can use
- The restore checks required for a restore method to be used
- Ways to fix the environment so that the checks pass

Structure of the restore plan

The restore plan consists of the following two sections:

- Preview/Review: This section describes how SnapManager will restore (or has restored) each file.
- Analysis: This section describes reasons why more efficient restore mechanisms will not be used (or were not used) during the restore.

The Preview/Review section

The first section in the restore plan is the preview or review section. This section presents how each file will be or has been restored. When you are previewing a restore operation, the section is titled "Preview." After a restore operation completes, the section is titled "Review."

The following example shows a Preview section when you are previewing a restore of a simple ASM database:

```
Preview:
The following files will be restored completely via: fast restore
+DG1/rac6/users.dbf

The following files will be restored completely via: storage side file system restore
+DG2/rac6/sysaux.dbf
+DG2/rac6/system.dbf
+DG2/rac6/undotbs1.dbf
+DG2/rac6/undotbs2.dbf
```

This preview shows that +DG1/rac6/users.dbf will be restored using the fast, volume-based restore method and the other files will be restored using the storage side file system restore. To determine why all files would not be restored via the fast restore method, look at the Analysis section.

One subsection exists for each restore method that will be used during the restore. The subsections are ordered by decreasing levels of storage method efficiency. In this example, the fast restore method is more efficient than storage file system restore and so appears first.

It is possible for one file to be restored by multiple restore methods. This can be the case when the underlying LUNs used for a file system are spread among different storage system volumes and some volumes are eligible for volume restore, while others are not. If multiple restore methods will be used to restore the same file, the subsection will be similar to the following:

```
The following files will be restored via a combination of:
[fast restore, storage side file system restore]
```

The Analysis section

The second section in a restore plan is the Analysis section. This section presents the reasons why more efficient restore mechanisms will not be or were not used. Using the information in the Analysis section, you can determine what is required to enable more efficient restore mechanisms, thus speeding up restores.

The following example shows an Analysis section when you are previewing a restore of a simple ASM database (continued from the example in the Preview/Review section).

```
Analysis:

The following reasons prevent certain files from being
restored completely via: fast restore
   * LUNs present in snapshot of volume n3700:
        /vol/rac_6_asm_disks may not be consistent when reverted:
        [n3700:/vol/rac6_asm_disks/DG4D1.lun]
     Mapped LUNs in volume n3700:/vol/rac_6_asm_disks
        not part of the restore scope will be reverted: [DG4D1.lun]

Files to restore:
     +DG2/rac6/sysaux.dbf
     +DG2/rac6/system.dbf
     +DG2/rac6/undotbs1.dbf
     +DG2/rac6/undotbs2.dbf
```

```
* Reasons denoted with an asterisk (*) are overridable.
```

This analysis shows two failed checks that prevent +DG2 from being restored using the fast restore method.

The first check failure is overridable (by using `-fast -override` from the command-line interface, or by selecting the override option in the graphical user interface). The second check failure about mapped LUNs in the volume is mandatory (not overridable).

You can resolve checks by doing the following:

- The only way to resolve a mandatory check failure is to change the environment so that the check will pass.
- To resolve an overridable check failure, you can change the environment, or override the check. Be careful though, because overriding the check could have undesired consequences.

Previewing backup restore information

You can preview information about a backup restore process before it occurs. You might want to do this to see information about restore eligibility that SnapManager for Oracle found on your backup. SnapManager analyzes data on your backup to determine whether the restore process can be completed successfully.

About this task

The restore preview shows the following information:

- Which restore mechanism (fast restore, storage-side file system restore, storage-side file restore, or host-side file copy restore) will be used to restore each file
- Why more efficient mechanisms were not used to restore each file, when you specify the `-verbose` option

If you specify the `-preview` option to the `backup restore` command, SnapManager does not restore anything, but simply produces the list of files to be restored and indicates how they will be restored.

You can preview all types of restores. The preview shows information on up to 20 files.

Steps

1. To preview a backup restore, enter this command:

   `smo backup restore -profile` *`profile_name`* `-label` *`label`* `-complete -preview -verbose`

   Example

   For example, enter:

```
smo backup restore -profile targetdb1_prof1
      -label full_bkup_sales_nov_08  -complete -preview -verbose
```

The following example shows files ready to be restored and lists the different methods used for each.

```
The following files will be restored via storage side file system
restore:
+DG1/sid/datafile06.dbf
+DG1/sid/datafile07.dbf

The following files will be restored via storage side file restore:
/mnt/systemB/volume1/datafile01.dbf
/mnt/systemB/volume1/datafile02.dbf
/mnt/systemB/volume1/datafile03.dbf
/mnt/systemB/volume2/datafile04.dbf
/mnt/systemB/volume2/datafile05.dbf
+DG2/sid/datafile08.dbf
+DG2/sid/datafile09.dbf

The following files will be restored via host side file copy restore:
+DG2/sid/datafile10.dbf
+DG2/sid/datafile11.dbf
```

The following example shows some files being restored using the host-side file copy restore process and also explains why some files cannot be restored using the fast restore option. If you specify the `-verbose` option, SnapManager displays a preview section and an analysis section that explains why each file cannot be restored via the fast restore process.

```
PREVIEW:
The following files will be restored via host side file copy restore:
+DG2/sid/datafile10.dbf
+DG2/sid/datafile11.dbf

ANALYSIS:
The following reasons prevent certain files from being restored via fast restore:
Reasons:
 Newer snapshots of /vol/volume2 have volume clones: SNAP_1
*Newer backups will be freed: nightly2, nightly3
Files to Restore:
/mnt/systemB/volume2/system.dbf
/mnt/systemB/volume2/users.dbf
/mnt/systemB/volume2/sysaux.dbf
/mnt/systemB/volume2/datafile04.dbf
/mnt/systemB/volume2/datafile05.dbf

The following reasons prevent certain files from being restored via fast restore:
Reasons:
* Newer snapshots of /vol/adm_disks will be lost: ADM_SNAP_5
* Luns present which were created after snapshot SNAP_0 was created: /vol/adm_disks/
disk5.lun
* Files not part of the restore scope will be reverted in filesystem: +DG2

Files Not in Restore Scope: +DG2/someothersid/data01.dbf
+DG2/someothersid/data02.dbf
Files to Restore:
+DG2/sid/datafile08.dbf +DG2/sid/datafile09.dbf
+DG2/sid/datafile10.dbf +DG2/sid/datafile11.dbf

* Reasons denoted with an asterisk (*) are overridable.
```

2. Review any reasons why the fast restore process cannot be used.
3. If SnapManager displays only reasons that are overridable, begin the restore by entering the `-backup restore` command without the `-preview` option.

   You will still have the opportunity to override non-mandatory checks.

Restoring backups using Single File SnapRestore

You can restore the backups using Single File SnapRestore (SFSR) method on the hosts.

About this task

Note: VBSR is not supported on Solaris hosts running Veritas stack with SFRAC and VCS environment.

Steps

1. Create a profile from the SnapManager for Oracle GUI.
2. Back up the database using the GUI.
3. Unlink the ORACLE and NFS service groups from the cluster service groups and freeze them.
4. Ensure that `ssh` is configured between the hosts and SnapDrive for UNIX is configured for `ssh` by setting the following parameter in the `snapdrive.conf` file of each host to: `#secure-communication-among-cluster-nodes=on`
5. From the SnapManager for Oracle GUI, perform full backup restore and recovery with the `–alllogs` option by checking the file-based restore as SFSR.
6. Unfreeze the service groups and link them back to the cluster service group.

   Note: This configuration is applicable only when you use SnapDrive 4.1.1 D2 for UNIX and SnapDrive 4.2 for UNIX versions.

   If one restore is followed by another restore then there is a possibility that the creation of the backup snapshot fails. If you run successive restores within the specified time in which the SFSR can complete, then SnapManager for Oracle will encounter snapshot creation errors.

   To prevent snapshot creation errors, ensure that restore operations are performed after the time period in which SFSR is in progress.

   To achieve this, check the LUN clone split process status by entering the following command from the storage system CLI.

   `rsh` *`filernname`* `lun clone split status` *`lun-name`*

   ```
   Sample Output:
   /vol/delaware_760gb/lun700gb (64% complete)..
   ```

Restoring backups on primary storage

You can restore and recover a database backup on primary storage with SnapManager. SnapManager attempts to perform a volume-based, fast restore by default and provides eligibility check

information. You can override some eligibility checks, if needed. If any condition prevents a fast restore, SnapManager performs a file-based restore instead.

About this task

If you are sure that backup files and conditions do not negatively affect the restore using a fast restore, you can force SnapManager to perform a fast restore. Alternatively, if you are certain that a backup cannot be performed using a fast restore, you can disable the fast restore eligibility checks and perform a file-based restore.

Using the `backup restore` command options, you can specify whether SnapManager should restore all or part of the backup. SnapManager also allows you to restore control files along with the data files or tablespaces from the backups in a single user operation. Include `-controlfiles` with `-complete` to restore control files along with tablespaces and data files.

You can select one of the following options to restore the backup. The following table lists the restore choices and the associated command options:

Restore choice	Command options
Restore the entire backup with all tablespaces and data files	`-complete`
Restore the list of specific tablespaces	`-tablespaces`
Restore specific data files	`-files`
Restore the control files only	`-controlfiles`
Restore tablespaces, data files, and control files	`-complete -controlfiles`

You can also restore the backup from an alternate location by specifying the `-restorespec` option.

If you include the `-recover` option, you can recover the database to any one of these:

- The last transaction that occurred in the database (all logs)
- A specific date and time
- A specific Oracle System Change Number (SCN)
- The time of the backup (no logs)
- Restore only

Note: Both the date and time recovery and the SCN recovery are point-in-time recoveries.

SnapManager 3.2 for Oracle provides the ability to recover the restored database backups automatically using the archive log files. Even if the archive log files are available in the external location, if you specify the `-recover-from-location` option, SnapManager uses the archive log files from the external location to recover the restored database backups.

On a Windows environment, when you specify the external archive log locations for the recovery of the restored backups, ensure you include the external location names completely in the upper case. In the file system, all the folders and subfolders names must be in the uppercase, as Oracle translates the

destination path to upper case and expects the external destination paths, folder names, and subfolder names to be in the upper case.

If you specify the external archive log destination paths in the lower case, Oracle might not be able to identify the specified path, and fails to recover the database.

SnapManager provides the external location to Oracle to recover. But, Oracle does not identify the files from the external destination. This behavior is noticed in both the flash recovery area destination and the ASM destination. These are issues with Oracle. Workaround for these issues is to always have backup of archive log files in such database layouts.

If any inconsistent SCN or date is provided for until SCN or until date recovery, then recovery will stop at the last consistent point recovered with the error message `Recovery succeeded, but insufficient`. Users have to manually perform their recovery to a consistent state.

For recovery when no logs are applied, SnapManager recovers until the last SCN of the last archive log file created during the backup. If the database is consistent until this SCN, then the database will be opened successfully. If the database is not consistent at this point, SnapManager still attempts to open the database, which will be opened successfully, if the database is already consistent.

Note: SnapManager does not support recovering the archive logs-only backups.

If the archive log destination on an NFS mount point, is not a Snapshot capable storage, SnapManager enables you to recover the restored database backups using the profile. Before performing SnapManager operations on not a Snapshot capable storage, you should add the destinations in the exclude parameter (`archivedLogs.exclude`) of the SnapManager configuration file (smo.config).

Ensure that you set the exclude parameter before creating a profile. Only after setting the exclude parameter in the SnapManager configuration file, the profile creation is successful.

Note: If the database is not a Snapshot capable storage on an ASM disk group, and when the database is selected as an archive log destination, SnapManager does not support to restore the backups using the profile.

If the backup is already mounted, SnapManager for Oracle does not mount the backup again and uses the already mounted backup.

If the backup is mounted by a different user, and if the current user does not have access to the previously mounted backup, other users have to provide the permissions. All the archive log files have read permissions for the groups owners; the current user might not get the permissions, if the backup is mounted by a different user group. The users can give permissions to the mounted archive log files manually and then retry the restore or recovery.

Recovering database backups in a RAC environment

During recovery of the database backups in a RAC environment, when the required archive log file is not found, Oracle requests for archive log files, and switches between different thread and change number in the RAC database. SnapManager for Oracle tries to recover the database as a best effort. The successful recovery of the database backups in the RAC environment depends on the availability of the archive log files in the backups.

Recommended recovery mechanism for the RAC database

- Ensure that all the archive log files are available in the backups or all the archive log files are available in the one external archive log destination.
- If multiple external archive log destinations are provided, you can provide overlap of the archive log files while specifying the external archive log destinations for all the threads.

  For example, the external archive log location - I can have 1 to 100 archive log files, the external archive log location - II can have 98 to 200 archive log files, and the external archive log location - III can have 198 to 300 archive log files.
- While pruning the archive log files, instead of deleting all the archive log files, you can delete the archive log files until SCN or until date so that the backups can have overlap of the archive log files.

You can specify the `-dump` option as an optional parameter to collect the dump files after the successful or failed restore operation.

Steps

1. To restore a complete database backup and recover all logs, enter this command:

   `smo backup restore -profile` *`profile_name`* `-label` *`label`* `-complete -recover -alllogs [-recover-from-location` *`path`* `[,`*`path2`*`]]-dump-verbose`

   For example, enter: `smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 - complete -recover -alllogs -verbose`

2. To restore data given other scenarios, complete one of the following:

Restore scenario	Sample command
Restore a complete database without control files and recover to a particular SCN number (3794392). In this case, the current control files exist, but all the data files are damaged or lost. Restore and recover the database from an existing full online backup to a point immediately before that SCN.	`smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 -complete -recover -until 3794392 -verbose`
Restore a complete database without control files and recover up to a date and time.	`smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 -complete -recover -until 2008-09-15:15:29:23 -verbose`

Restore scenario	Sample command
Restore a whole database without control files and recover up to a data and time. In this case, the current control files exist, but all of the data files are damaged or lost or a logical error occurred after a specific time. Restore and recover the database from an existing full online backup to a date and time immediately before the point of failure.	`smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 -complete -recover -until "2008-09-15:15:29:23" -verbose`
Restore a database partially (one or more data files) without control files and recover using all available logs. In this case, the current control files exist, but one or more data files are damaged or lost. Restore those data files and recover the database from an existing full online backup using all available logs.	`smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 -files /u02/oradata/sales02.dbf /u02/oradata/sales03.dbf /u02/oradata/sales04.dbf -recover -alllogs -verbose`
Restore a database partially (one or more tablespaces) without control files and recover using all available logs. In this case, the current control files exist, but one or more tablespaces are dropped or one of more data files belonging to the tablespace are damaged or lost. Restore those tablespaces and recover the database from an existing full online backup using all available logs.	`smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 -tablespaces users -recover -alllogs -verbose`
Restore only control files and recover using all available logs. In this case, the data files exist, but all control files are damaged or lost. Restore just the control files and recover the database from an existing full online backup using all available logs.	`smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 -controlfiles -recover -alllogs -verbose`
Restore a complete database without control files and recover using the backup control files and all available logs. In this case, all data files are damaged or lost. Restore just the control files and recover the database from an existing full online backup using all available logs.	`smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 -complete -using-backup-controlfile -recover -alllogs -verbose`
Recover the restored database using the archive log files from the external archive log location.	`smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 -complete -using-backup-controlfile -recover -alllogs -recover-from-location /user1/archive -verbose`

3. Review the fast restore eligibility checks.

For example, enter: `smo backup restore -profile targetdb1_prof1 -label full_bkup_sales_nov_08 -complete -recover -alllogs -recover-from-location /user1/archive -verbose`

4. If the eligibility check determined that no mandatory checks failed, if certain conditions can be overridden (messages are denoted with an asterisk), and if you want to continue with the restore process, enter the following to continue:

   `backup restore -fast override`

5. Specify external archive log locations using the `-recover-from-location` option. With the archive log files, the restored database backups are recovered.

Related tasks

Restoring backups from an alternate location on page 196

Related references

The smo backup restore command on page 273

Performing block-level restore operations with RMAN

You can configure SnapManager to catalog its backups in Recovery Manager (RMAN), an Oracle tool, so that you can perform a block-level recovery using RMAN. SnapManager enables you to catalog backups in the RMAN repository. RMAN can use either the database's control files or a separate recovery catalog database as its repository.

About this task

The following lists the major steps needed to perform a block-level restore using RMAN:

- Perform a full offline backup of the database using SnapManager for Oracle.
- Verify that the backup is cataloged with RMAN.
- Verify the backup to determine if any blocks have been corrupted.
- Mount the backup using SnapManager to make it accessible to RMAN.
- Using RMAN, perform the block-level restore.
- Verify that the corrupted blocks have been repaired.

Steps

1. To perform a full offline backup using SnapManager, enter this command:

   `smo backup create -offline -full -profile profile_name -label backup_label_name -verbose`

Where:

- *profile_name* is the name of the profile associated with the backup
- *backup_label_name* is the name of the backup label

```
smo backup create -offline -full -profile profile_monthly
-label full_backup -verbose

SMO-07109 [INFO ]: Cataloguing all files in backup set with RMAN
TAG=SMC_full_backup_1158773581857, RMAN=ES0/controlfile.
...
SMO-13037 [INFO ]: Successfully completed operation: Backup
SMO-13048 [INFO ]: Operation Status: SUCCESS
SMO-13049 [INFO ]: Elapsed Time: 0:02:20.506
Operation Id [ff8080810dcc47e3010dcc47eb7a0001] succeeded.
```

2. To verify that the backup is cataloged with RMAN, from the database host, enter this command at the RMAN prompt:

   list datafilecopy tag *tag_name*;

```
RMAN> list datafilecopy tag SMO_full_backup_1158773581857;

Recovery Manager: Release 10.2.0.1.0 - Production on Wed Sep 20 10:33:41 2008
Copyright (c) 1982, 2008, Oracle. All rights reserved.
using target database control file instead of recovery catalog
 List of Datafile Copies
 Key File S Completion Time Ckp SCN Ckp Time Name
 ------- ---- - --------------- ---------- --------------- ---
 335 1 A 20-SEP-08 1347825 20-SEP-08
 /opt/<path>/smo/mnt/Host4_ES0_SMO_E_ES0_F_C_0_ff8080810dcc47e3010dcc47eb7a0001
 /system01.dbf
 336 2 A 20-SEP-08 1347825 20-SEP-08
 /opt/<path>/smo/mnt/Host4_ES0_SMO_E_ES0_F_C_0_ff8080810dcc47e3010dcc47eb7a0001
 /undotbs01.dbf
 334 3 A 20-SEP-08 1347825 20-SEP-08
 /opt/<path>/smo/mnt/Host4_ES0_SMO_E_ES0_F_C_0_ff8080810dcc47e3010dcc47eb7a0001
 /sysaux01.dbf
 333 4 A 20-SEP-08 1347825 20-SEP-08
 /opt/<path>/smo/mnt/Host4_ES0_SMO_E_ES0_F_C_0_ff8080810dcc47e3010dcc47eb7a0001
 /user01.dbf
 337 5 A 20-SEP-08 1347825 20-SEP-08
 RMAN>
```

3. To verify the database and determine if any blocks are corrupted, enter this dbverify command:

   dbv FILE=user01.dbf

   Example

   The following output shows that two pages are corrupt:

```
DBVERIFY: Release 10.2.0.1.0 - Production on Wed Sep 20 13:35:44 2006
Copyright (c) 1982, 2005, Oracle. All rights reserved.
DBVERIFY - Verification starting : FILE = user01.dbf
Page 625 is marked corrupt
Corrupt block relative dba: 0x01400271 (file 5, block 625)
Bad header found during dbv:
Data in bad block:
type: 240 format: 6 rdba: 0xed323b81
last change scn: 0x6f07.faa74628 seq: 0x87 flg: 0x02
spare1: 0x60 spare2: 0x5 spare3: 0xef7d
consistency value in tail: 0xa210fe71
check value in block header: 0x13c7
block checksum disabled...
Page 627 is marked corrupt
Corrupt block relative dba: 0x01400273 (file 5, block 627)
Bad header found during dbv:
Data in bad block:
```

```
type: 158 format: 7 rdba: 0x2101e16d
last change scn: 0xe828.42414628 seq: 0xb4 flg: 0xff
spare1: 0xcc spare2: 0x81 spare3: 0x8665
consistency value in tail: 0x46d20601
check value in block header: 0x1a84
computed block checksum: 0x6c30
DBVERIFY - Verification complete
Total Pages Examined : 1280
Total Pages Processed (Data) : 1123
Total Pages Failing (Data) : 0
Total Pages Processed (Index): 0
Total Pages Failing (Index): 0
Total Pages Processed (Other): 34
Total Pages Processed (Seg) : 0
Total Pages Failing (Seg) : 0
Total Pages Empty : 120
Total Pages Marked Corrupt: 2
Total Pages Influx : 0
Highest block SCN : 1337349 (0.1337349)
[oracle@Host4 Host4_ES0]$
```

4. To make the files from the backup accessible on the host and to RMAN, mount the backup using this command:

   smo backup mount -profile *profile_name* -label *label* -verbose

   Example

```
smo backup mount -profile SALES1 -label full_backup -verbose

SMO-13046 [INFO ]: Operation GUID 8abc013111b9088e0111b908a7560001 starting on Profile SALES1
SMO-08052 [INFO ]: Beginning to connect mount(s) [/mnt/ssys1/logs, /mnt/ssys1/data] from
logical snapshot SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001.
SMO-08025 [INFO ]: Beginning to connect mount /mnt/ssys1/logs from snapshot
SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001_0 of volume hs_logs.
SMO-08027 [INFO ]: Finished connecting mount /mnt/ssys1/logs from snapshot
SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001_0 of volume hs_logs.
SMO-08025 [INFO ]: Beginning to connect mount /mnt/ssys1/data from snapshot
SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001_0 of volume hs_data.
SMO-08027 [INFO ]: Finished connecting mount /mnt/ssys1/data from snapshot
SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001_0 of volume hs_data.
SMO-08053 [INFO ]: Finished connecting mount(s) [/mnt/ssys1/logs, /mnt/ssys1/data] from
logical snapshot SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001.
SMO-13037 [INFO ]: Successfully completed operation: Backup Mount
SMO-13048 [INFO ]: Operation Status: SUCCESS
SMO-13049 [INFO ]: Elapsed Time: 0:01:00.981
Operation Id [8abc013111b9088e0111b908a7560001] succeeded.
```

5. To recover the blocks, in RMAN, enter this command:

   blockrecover datafile '/mountpoint/path/file.dbf' block *block_id*, from tag *backup_rman_tag*

   Example

```
RMAN> blockrecover datafile
'/mnt/ssys1/Host4_ES0/file01.dbf' block 625, 626, 627
from tag SMO_full_backup_1158773581857;

Starting blockrecover at 20-SEP-08 using target database control file instead of recovery
catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: sid=153 devtype=DISK
channel ORA_DISK_1: restoring block(s) from datafile copy
 /opt/Ontap/smo/mnt/_mnt_ssys1_Host4_ES0_SMO_E_ES0_F_C_0_ff8080810dcc47e3010dcc47eb7a0001/
user01.dbf
starting media recovery
media recovery complete, elapsed time: 00:00:01
Finished blockrecover at 20-SEP-08
```

6. To verify the blocks have been repaired, use the following command:

   dbv FILE=filename.dbf

   Example

   The following output shows that no pages are corrupt. All corrupted blocks were repaired and restored.

```
dbv FILE=user01.dbf

DBVERIFY: Release 10.2.0.1.0 - Production on Wed Sep 20 13:40:01 2008
Copyright (c) 1982, 2008, Oracle. All rights reserved.
DBVERIFY - Verification starting : FILE = user01.dbf
DBVERIFY - Verification complete
Total Pages Examined : 1280
Total Pages Processed (Data) : 1126
Total Pages Failing (Data) : 0
Total Pages Processed (Index): 0
Total Pages Failing (Index): 0
Total Pages Processed (Other): 34
Total Pages Processed (Seg) : 0
Total Pages Failing (Seg) : 0
Total Pages Empty : 120
Total Pages Marked Corrupt : 0
Total Pages Influx : 0
Highest block SCN : 1337349 (0.1337349)
```

Restores of backups from an alternate location

SnapManager provides the capability to restore data files and control files from a location other than that of the Snapshot copies in the original volume.

SnapManager requires that you restore from the alternate location to the original location. The original location is the location of the file on the active file system at the time of the backup. The alternate location is the location from which a file will be restored.

You can restore the following data from an alternate location:

- The data files from an intermediate file system to an active file system
- The blocks of data from an intermediate raw device into an active raw device

Recovery is automated by SnapManager though, recovery is done by Oracle. When recovering from external locations, SnapManager use the `RECOVERY AUTOMATIC FROM <location>` command.

Oracle recovers the database using this location. The files in this location should be recognizable by Oracle. The file names should be in the default format. When recovering from Flash Recovery Area, SnapManager provides the translated path to Oracle. Oracle though, does not recover from the flash recovery area as it could not generate the correct file name. Ideally, Flash Recovery Area is a destination that is intended to work with RMAN. SnapManager treat the Flash Recovery Area as yet another destination.

Related tasks

Creating restore specifications on page 194

Restores of backups from an alternate location overview

To restore a database backup from an alternate location, use the following major steps, each of which is further described in this section.

- Do one of the following, depending on your database layout and what needs to be restored:
  - Restore the required data files from tape, SnapVault, SnapMirror, or any other media to any file system mounted on the database host.
  - Restore the required file system and mount it on the database host.
  - Connect to the required raw devices that exist in the local host.
- Create a restore specification Extensible Markup Language (XML) file that includes the mappings that SnapManager requires to restore from the alternate location to the original location. Save the file in a location that SnapManager can access.
- Use SnapManager to restore and recover the data using the restore specification XML file.

Restoration of the data from files

Before you restore from an alternate location, you need to restore the necessary files from any storage media and restore the files from applications like SnapVault or SnapMirror to a file system mounted on the local host.

You can use the restore from an alternate location operation to copy the files from an alternate file system to an active file system.

You need to specify the alternate locations from which to restore the original files by creating a restore specification.

Restoration of the data from file systems

Before you restore from an alternate location, you need to restore the necessary file system and mount it on the local host.

You can invoke the restore from an alternate location operation to copy the files from an alternate file systems to active file systems.

To perform this operation, you need to specify the alternate mountpoints from which to restore the original mountpoints and specify the original snapshot names by creating a restore specification.

Note: The physical Snapshot copy name is a necessary component since the same file system could be snapped multiple times in a single backup operation (for example, once for the data files and once for the logs).

For ASM, the disk group name must be the same name as the disk group that SnapManager cloned to register the backup with RMAN. This name can be obtained by viewing the backup properties.

Related tasks

Creating restore specifications on page 194

Restoration of the data from raw devices

Before you restore from an alternate location, you need to connect to the necessary raw devices that exist on the local host.

You can invoke the restore from an alternate location operation to copy the blocks of data from alternate raw devices to active raw devices. To perform this operation, you need to specify the alternate raw device from which to restore the original raw device by creating a restore specification.

Related tasks

Creating restore specifications on page 194

Creating restore specifications

The restore specification is an Extensible Markup Language (XML) file that sets the mappings SnapManager requires to restore from and to the appropriate location. Use the restore specification file when you restore a backup from an alternate location.

About this task

Create the restore specification file with any text editor. Use an extension of .xml for the file to enable appropriate editing features.

Steps

1. Open a text file.
2. Enter any file mapping information using the format in the following example.

   File mapping specifies where an individual file will be restored from. The original location is the location of the file on the active file system at the time of backup. The alternate location is the location from which the file will be restored.

   Example

```
<file-mapping>
    <original-location>/path/dbfilename.dbf</original-location>
    <alternate-location>/path/dbfilename1.dbf</alternate-location>
</file-mapping>
```

3. Enter any mounted file system mapping information using the format in the following example.

   Mountpoint refers to a host mountpoint(for example, `/mnt/myfs/)` or an ASM diskgroup mountpoint (for example, `+MY_DG`. The mountpoint mapping specifies the mountpoint from which files will be restored. The original location is the location of the mountpoint in the active file system at the time of backup. The alternate location is the mountpoint from which the files in the original location will be restored. The `snapname` is the name of the snapshot in which the original files were backed up.

For ASM, the disk group name must be the same as the name as the disk group that SnapManager cloned to register the backup with RMAN. This name can be obtained by viewing the backup properties.

Note: The physical Snapshot copy name is a necessary component since the same file system could be snapped multiple times in a single backup operation (for example, once for the data files and once for the logs).

Example

```
<mountpoint-mapping>
     <original-location>/path/db_name</original-location>
     <snapname>snapname</snapname>
     <alternate-location>/path/vaultlocation</alternate-location>
</mountpoint-mapping>
<mountpoint-mapping>
     <original-location>+DiskGroup_1</original-location>
     <snapname>snapname</snapname>
     <alternate-location>+DiskGroup_2</alternate-location>
</mountpoint-mapping>
```

4. Enter any raw device mapping tags and locations using the format in the following example.

   Raw device mapping specifies the location from which a raw device will be restored.

   Example

```
<raw-device-mapping>
     <original-location>/path/raw_device_name</original-location>
     <alternate-location>/path/raw_device_name</alternate-location>
</raw-device-mapping>
```

5. Enter the following:

   `</restore-specification>`

6. Save the file as an .xml file and close the specification.

Restore specification example

The following example shows the restore specification structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<restore-specification xmlns="http://www.<namespace>.com">
<!-- "Restore from file(s)" -->
     <file-mapping>
          <original-location>/
mnt/pathname/dbname/users01.dbf</original-location>
          <alternate-location>/mnt/vault/users01.dbf</alternate-location>
     </file-mapping>
<!-- "Restore from host mounted filesystem(s)" -->
     <mountpoint-mapping>
          <original-location>/mnt/pathname/dbname/fs</original-location>
          <snapname>Snapshotname</snapname>
          <alternate-location>/mnt/vaultlocation</alternate-location>
     </mountpoint-mapping>
<!-- "Restore from ASM mounted filesystem(s)" -->
     <mountpoint-mapping>
          <original-location>+DISKGROUP_1</original-location>
          <snapname>snapshotname</snapname>
          <alternate-location>+DISKGROUP_2</alternate-location>
     </mountpoint-mapping>
<!-- "Restore from raw device" -->
     <raw-device-mapping>
          <original-location>/pathname/devicename</original-location>
          <alternate-location>/pathname/devicename</alternate-location>
```

```
    </raw-device-mapping>
</restore-specification>
```

Restoring backups from an alternate location

You can restore from an alternate location to restore the data files from an intermediate file system to an active file system, or to restore the blocks of data from an intermediate raw device into an active raw device.

Before you begin

Create a restore specification XML file and be sure to specify the appropriate information for the type of backup restore you choose.

About this task

Use the `smo backup restore` command and specify the restore specification XML file you created to restore the backup from an alternate location.

Step

1. To restore a complete backup from an alternate location, enter this command:

   `smo backup restore -profile profile -label label -complete -alllogs -restorespec restorespec`

Related references

The smo backup restore command on page 273

Cloning database backup

By cloning, you can create a copy of a database that is an exact replica of the original. You could do this to perform tasks such as test an upgrade to a database without affecting the database in production, duplicate a master installation to several training systems, or duplicate a master installation as a base installation to other servers having similar requirements.

You can perform the following tasks related to clones:

- Clone a database from an existing backup.
- Clone a database in its current state, which enables you to make the backup and the clone in one procedure.
- Clone a protected backup on secondary or even tertiary storage.
- Clone a database and use custom plug-in scripts, which run before or after the clone operation.
- Clone a database to the same host where the database resides.
- Clone a database using archive log files from the external archive log location.
- Clone a database to an alternate host.
- Clone a RAC database.
- View a list of clones.
- View detailed clone information.
- Delete clones.

Administrators can perform tasks with the SnapManager graphical user interface or by using the command-line interface. The *SnapManager for Oracle Installation and Administration Guide* explains how to complete these tasks using commands. The SnapManager online Help explains how to complete the tasks using the graphical interface.

Cloning overview

You can clone a database to create an exact replica of the original database. You can create the clone from a full backup or from the current state of the database.

Using SnapManager to create a clone provides the following advantages:

Advantages	Details
Speed	SnapManager can clone data volumes. The SnapManager clone operation uses the FlexClone feature available with Data ONTAP.

Advantages	Details
Space efficiency	Creating a clone using SnapManager requires space only for the changes between the backup and the clone. A SnapManager clone is a writable Snapshot copy of the original database and can grow as needed. In contrast, a physical clone of the database requires that you have sufficient space available to copy the entire database.
Virtual copy	The clone is a virtual copy of the original database. For example, you can use a clone for testing, platform and update checks, multiple simulations against a large data set, and remote office testing and staging. People can work with the cloned database as if it were the original database. Changes to the clone do not affect the original database. After the database is cloned, the cloned database is fully operational.
Simplicity	Using SnapManager commands, you can clone a database to the same host or to a different host.

You can clone a backup on primary (local) storage or a protected backup that is on secondary (remote) storage. However, you cannot clone the last backup copy that is in progress or has been transferred to secondary storage.

The following prerequisites must be met before a database can be cloned:

- The server parameter file `spfile <SID>.ora` must not exist.
- The `init <SID>.ora` must not exist.
- The directory `[/etc|/var/opt/oracle]/oratab` must not contain an entry pointing to the target SID.
- The Oracle dump destinations that are specified in the clone spec must not exist.
- The Oracle control files that are specified in the clone specification must not exist.
- The Oracle redo log files that are specified in the clone specification must not exist.

You must give the clone a new Oracle System Identifier (SID). Oracle does not permit you to run two databases with the same SID simultaneously on the same host. You can have a clone on a different host using the same SID. You can give the clone a label, or let SnapManager create a label using the SID, date, and time the clone was made.

When you enter a comment, you can include spaces and special characters. In contrast, when you enter a label, do not include spaces or special characters.

As part of the cloning process, SnapManager creates the necessary Oracle files and parameters for the cloned database. An example of a necessary Oracle file is `init<clone_SID>.ora`. When you clone a database, SnapManager creates a new `init <clone_SID>.ora` file for the database in the `$ORACLE_HOME/dbs` directory.

When SnapManager clones the storage for a database, it also creates a new file system mountpoint, but does not change the directory structure under the mountpoint.

Oracle 11g in a Direct NFS (DNFS) environment allows additional mountpoint configuration, such as multiple paths for load balancing, in the oranfstab file. SnapManager does not modify this file, so

any additional properties you want a clone to use must be manually added to the oranfstab file after cloning with SnapManager

You can clone a Real Application Cluster (RAC) database as well as a non-clustered database. A RAC clone starts as a single database.

You can clone a database backup to the host in which the database resides or to an alternate host.

You can also clone an ASM database to a remote host. When doing so, make sure the ASM instance is running on the remote host.

If the database you cloned was using an spfile, SnapManager creates an spfile for the clone. It places this file in the `$ORACLE_HOME/dbs` directory and creates the directory structure for the diagnostic files. The file name is `spfile<clone_SID>.ora`.

Cloning methods

You can clone a database using one of two methods. The method you choose affects the `clone create` operation.

The following table describes the cloning methods and their effect on the `clone create` operation and its `-reserve` option. A LUN can be cloned using either method.

Cloning method	Description	`clone create -reserve`
LUN cloning	A new clone LUN is created within the same volume.	When `-reserve` for a LUN is set to `yes`, space is reserved for the full LUN size within the volume.
Volume cloning	A new FlexClone is created and the clone LUN exists within the new clone volume. Uses FlexClone technology.	When`-reserve` for a volume is set to `yes`, space is reserved for the full volume size within the aggregate.

Creating clone specifications

SnapManager for Oracle uses a clone specification Extensible Markup Language (XML) file that indicates the mappings, options, and parameters that you want to use in the clone operation. SnapManager uses this information to determine where to place the files it clones and how to handle diagnostic information, control files, parameters, and other information.

About this task

You can create the clone specification file by using the SnapManager graphical user interface, command-line interface, or a text editor. Use an extension of .XML for the file to enable appropriate editing features. You might want to save this file so that you can use it for other clone operations.

You can also create a template clone specification and then customize it to fit your needs. Use the `smo clone template` command or in the graphical user interface, use the clone wizard to create a template clone specification.

SnapManager for Oracle adds a version string to any clone specification templates that it generates. SnapManager for Oracle assumes the latest version if it encounters a clone specification file lacking a version string.

A task can be executed multiple times, either with the same or different parameter and value combinations. For example, you could use a "Save" task to save multiple files.

Steps

1. To create a clone specification, open a text file and use the following file structure.

   Example

   ```
   <clone-specification xmlns="http://www.example.com>
        <storage-specification/>
        <database-specification/>
    </clone-specification>
   ```

2. In the storage specification component, enter the mount points for the data files.

   The storage specification lists the locations for the new storage created for the clone, such as, data file mountpoints and raw devices. These are the items that must be mapped from the source to the destination.

   Example

   The following example displays the data file mountpoint syntax to use in the clone specification:

   ```
   <mountpoint>
        <source>/mnt/path/source_datafile_mountpoint</source>
        <destination>/mnt/path/target_datafile_mountpoint</destination>
   </mountpoint>
   ```

3. If you have a raw device on the source, you must specify the path for the raw device on the source, and then specify

   `destination auto-generate="true"`
   for the destination.

   Unlike in the clone mapping file from previous versions of SnapManager for Oracle, you cannot specify a location for the raw device on the destination. SnapManager for Oracle will choose the next available device name for the cloned raw device.

   Example

   The following example displays the raw device syntax to use in the clone specification:

   ```
   <raw-device>
        <source>/dev/raw/raw1</source>
        <destination auto-generate="true"/>
   </raw-device>
   ```

4. In the database specification component, identify the control file information as a list of the control files you want created for the clone.

The database specification specifies the database options for the clone, such as, control files, redo logs, archive logs, and Oracle parameters.

Example

The following example displays the control file syntax to use in the clone specification:

```
<controlfiles>
     <file>/mnt/path/clonename/control/control01.ctl</file>
     <file>/mnt/path/clonename/control/control02.ctl</file>
</controlfiles>
```

5. Specify the redo log structure for the clone.

Example

```
<redologs>
     <redogroup>
          <file>/mnt/path/clonename/redo/redo01.log</file>
          <number>1</number>
          <size unit="M">100</size>
     </redogroup>
     <redogroup>
         <file>/mnt/path/clonename/redo/redo02.log</file>
         <number>2</number>
         <size unit="M">100</size>
     </redogroup>
</redologs>
```

6. Specify any Oracle parameters that should be set to a different value in the cloned database than in the source database. If you are using Oracle 9 or 10, you must specify the first three parameters.

- Background dump
- Core dump
- User dump
- (Optional) Archive logs

If you do not specify archive logs for the destination, SnapManager creates the clone in NOARCHIVELOG mode. SnapManager copies this parameter information into the clone's init.ora file.

Note: If a parameter entered conflicts with a parameter that SnapManager for Oracle must change to create the clone, the clone operation is aborted and you receive an error message.

Example

The following example displays the parameter syntax to use in the clone specification:

```
<parameters>
     <parameter>
          <name>log_archive_dest_1</name>
          <value>LOCATION=/mnt/path/clonename/archive</value>
     </parameter>
</parameters>
```

Example

You can use an Oracle default value by using a default element within the parameter element. In the following example, the OS authentication prefix will take Oracle's default value because the default element was used.

```
<parameters>
     <parameter>
          <name>os_authent_prefix</name>
          <default></default>
     </parameter>
</parameters>
```

Example

You can specify an empty string as the value for a parameter by using an empty parameter value element. In the following example, the OS authentication prefix will be set to an empty string.

```
<parameters>
     <parameter>
          <name>os_authent_prefix</name>
          <value></value>
     </parameter>
</parameters>
```

You can use the value from the source database's init.ora file for a parameter by not specifying a parameter element in the clone specification for that parameter.

7. (Optional) Specify arbitrary SQL statements to execute against the clone once it is online.

   You can use this to do tasks such as recreating the `temp files` in the cloned database.

   Example

   The following is an example of a SQL statement that you may want to have executed as part of the clone operation:

```
<sql-statements>
   <sql-statement>
     ALTER TABLESPACE TEMP ADD
     TEMPFILE '/mnt/path/clonename/temp_user01.dbf'
     SIZE 41943040 REUSE AUTOEXTEND ON NEXT 655360
     MAXSIZE 32767M
   </sql-statement>
</sql-statements>
```

Clone specification example

The following example displays the clone specification structure, including both the storage and database specification components:

```
<clone-specification xmlns="http://www.example.com>

   <storage-specification>
     <storage-mapping>
        <mountpoint>
          <source>/mnt/path/source_mountpoint</source>
          <destination>/mnt/path/target_mountpoint</destination>
        </mountpoint>
        <raw-device>
          <source>/dev/raw/raw1</source>
          <destination auto-generate="true"/>
        </raw-device>
        <raw-device>
          <source>/dev/raw/raw2</source>
          <destination auto-generate="true"/>
        </raw-device>
     </storage-mapping>
   </storage-specification>

   <database-specification>
     <controlfiles>
       <file>/mnt/path/clonename/control/control01.ctl</file>
       <file>/mnt/path/clonename/control/control02.ctl</file>
```

```
     </controlfiles>
     <redologs>
         <redogroup>
           <file>/mnt/path/clonename/redo/redo01.log</file>
           <number>1</number>
           <size unit="M">100</size>
        </redogroup>
        <redogroup>
           <file>/mnt/path/clonename/redo/redo02.log</file>
           <number>2</number>
           <size unit="M">100</size>
        </redogroup>
    </redologs>
    <parameters>
      <parameter>
        <name>log_archive_dest_1</name>
        <value>LOCATION=/mnt/path/clonename/archive</value>
      </parameter>
      <parameter>
        <name>background_dump_dest</name>
        <value>/mnt/path/clonename/admin/bdump</value>
      </parameter>
      <parameter>
        <name>core_dump_dest</name>
        <value>/mnt/path/clonename/admin/cdump</value>
      </parameter>
      <parameter>
        <name>user_dump_dest</name>
        <value>/mnt/path/clonename/admin/udump</value>
      </parameter>
    </parameters>
   </database-specification>
</clone-specification>
```

The following example displays the clone specification structure, including both the storage and database specification components, for a Windows environment.

```
<clone-specification xmlns="http://www.example.com>

<storage-specification>
    <storage-mapping>
        <mountpoint>
            <source>D:\oracle\<SOURCE SID>_sapdata</source>
            <destination>D:\oracle\<TARGET SID>_sapdata</destination>
        </mountpoint>
    </storage-mapping>
</storage-specification>

<database-specification>
    <controlfiles>
        <file>D:\oracle\<TARGET SID>\origlogA\cntrl\cntrl<TARGET SID>.dbf</file>
        <file>D:\oracle\<TARGET SID>\origlogB\cntrl\cntrl<TARGET SID>.dbf</file>
        <file>D:\oracle\<TARGET SID>\sapdata1\cntrl\cntrl<TARGET SID>.dbf</file>
     </controlfiles>

     <redologs>
        <redogroup>
            <file>D:\oracle\<TARGET SID>\origlogA\log_g11m1.dbf</file>
            <file>D:\oracle\<TARGET SID>\mirrlogA\log_g11m2.dbf</file>
            <number>1</number>
            <size unit="M">100</size>
        </redogroup>
        <redogroup>
            <file>D:\oracle\<TARGET SID>\origlogB\log_g12m1.dbf</file>
            <file>D:\oracle\<TARGET SID>\mirrlogB\log_g12m2.dbf</file>
            <number>2</number>
            <size unit="M">100</size>
        </redogroup>
        <redogroup>
            <file>D:\oracle\<TARGET SID>\origlogA\log_g13m1.dbf</file>
            <file>D:\oracle\<TARGET SID>\mirrlogA\log_g13m2.dbf</file>
            <number>3</number>
```

```
            <size unit="M">100</size>
        </redogroup>
        <redogroup>
            <file>D:\oracle\<TARGET SID>\origlogB\log_g14m1.dbf</file>
            <file>D:\oracle\<TARGET SID>\mirrlogB\log_g14m2.dbf</file>
            <number>4</number>
            <size unit="M">100</size>
       </redogroup>
    </redologs>

    <parameters>
        <parameter>
            <name>log_archive_dest</name>
            <value>LOCATION=>D:\oracle\<TARGET SID>\oraarch</value>
        </parameter>
        <parameter>
            <name>background_dump_dest</name>
            <value>D:\oracle\<TARGET SID>\saptrace\background</value>
        </parameter>
        <parameter>
            <name>core_dump_dest</name>
            <value>D:\oracle\<TARGET SID>\saptrace\background</value>
        </parameter>
        <parameter>
            <name>user_dump_dest</name>
            <value>D:\oracle\<TARGET SID>\saptrace\usertrace</value>
        </parameter>
    </parameters>
   </database-specification>
</clone-specification>
```

Related concepts

Considerations for cloning a database to an alternate host on page 208

Related tasks

Cloning databases and using custom plug-in scripts on page 204
Cloning databases from backups on page 205
Cloning databases in the current state on page 207

Cloning databases and using custom plug-in scripts

SnapManager provides a method for using your custom scripts before and after a clone operation occurs. For example, you might have created a custom script that validates a clone database SID and ensures the SID is allowed by your naming policy. Using the SnapManager clone plug-in, you can include your custom scripts and have them run automatically before or after a SnapManager clone operation.

Steps

1. View sample plug-in scripts.
2. Create a script from scratch or modify one of the sample plug-in scripts.

   Create your custom script according to SnapManager plug-in script guidelines.
3. Place your custom script in a specified directory location.

4. Update the clone specification XML file and include information about your custom script that should be used during the cloning process.
5. Using a SnapManager command, verify that the custom scripts are operational.
6. When you initiate the clone operation, include the script name and optional parameters.

Cloning databases from backups

About this task

You can clone a database using an existing backup using the `clone create` command.

You must first create a clone specification file for the database. SnapManager creates the clone based on the information in this specification file.

You must give the clone a new Oracle SID. Oracle does not permit you to run two databases with the same SID simultaneously on the same host. You can have a clone on a different host using the same SID. To designate a unique name for the clone, use the -label option. If you do not use this option, SnapManager creates a unique name for the clone that includes the SID, date, and time.

After you clone a database, you might want to update your `tnsnames.ora` files on your client machines with the new cloned database connection information. The `tnsnames.ora` files are used for clients to connect to an Oracle instance without having to specify the full database information when they connect. SnapManager does not update `tnsnames.ora` files.

SnapManager always creates a backup including archive log files, if you are using the profile created with the `-include-with-online-backups` option. SnapManager enables you to clone only the full database backups.

From SnapManager 3.2 for Oracle you can clone the backups containing the datafiles and archive log files.

If you do not have the archive log in the backups but available from an external location, you can specify the external location during cloning for recovering the cloned database to a consistent state. Ensure the external location is accessible by Oracle. Cloning of the archive logs-only backups is not supported.

Though archive log backup is taken along with the online partial backup, user cannot create a database clone using this backup.

On a Windows environment, when you specify the external archive log locations for recovering the cloned database to a consistent state, ensure you include the external location names completely in the upper case. In the file system, all the folders and subfolders names must be in the uppercase, as Oracle translates the destination path to upper case and expects the external destination paths, folder names, and subfolder names to be in the upper case.

If you specify the external archive log destination paths in the lower case, Oracle might not be able to identify the specified path, and fails to recover the cloned database.

You can clone the database backup from the external archive log file location only for a standalone database.

The cloning of online database backup of the RAC database using the external archive log file location fails due to failure in recovery. This is due an Oracle issue as Oracle fails to find and apply the archive log files for recovery from the external archive log location while cloning the database backup.

You can specify the `-dump` option as an optional parameter to collect the dump files after the successful or failed clone create operation.

From SnapManager 3.2 for Oracle you can clone the backup containing the read-only or offline tablespaces.

Cloning datafile backup without archive log backup

When the datafiles backup does not have the archive log backup included in it, SnapManager for Oracle clones the database based on the SCN recorded during the backup. If the cloned database cannot be recovered until the SCN, the error message `Archived log file for thread <number> and change <SCN> required to complete recovery` is displayed, even though SnapManager for Oracle continues to clone the database, and finally succeeds in creating the clone.

While cloning using the datafiles backup without included archive log backup, SnapManager recovers the cloned database until the last archive log SCN which is recorded during the backup.

Steps

1. Create a clone specification file.
2. To create a clone, enter this command:

```
smo clone create -backup-label backup_name -newsid new_sid -label
clone_label -profile profile_name -clonespec full_path_to_clonespecfile
[-taskspec taskspec] [-recover-from-location] path1 [,<path2>...]][-
dump]
```

Related concepts

Considerations for cloning a database to an alternate host on page 208
Variables available in the task scripts for clone operation on page 239

Related tasks

Cloning databases in the current state on page 207
Creating clone specifications on page 199
Creating pre-task, post-task, and policy scripts for SnapManager operations on page 230
Creating task scripts for SnapManager operation on page 243
Installing the task scripts on page 244

Related references

The smo clone create command on page 284

Cloning databases in the current state

You can create a backup and a clone in one procedure from the current state of the database.

About this task

Specifying the profile with the flag `-current` tells SnapManager to clone from the current state of the database, creating a backup and then the clone with one command.

In the profile, if you have enabled to take an online datafiles backup and archive logs backup together for cloning, whenever you take an online datafiles backup, the archive logs backups are taken along with the datafiles immediately. While cloning the database, SnapManager creates the datafiles backup along with the archive log backup and creates the clone applying the archive log files from the archive logs backup. When the archive log backup is not included with the datafiles backup in the profile settings, SnapManager does not create the archive log backup and so could not create the clone of the database.

Step

1. To clone the database from its current state, enter the following command:

   `smo clone create -profile profile_name -current -label clone_name -clonespec ./clonespec_filename.xml`

   This command takes a full automatic backup (generating the backup label) and immediately makes a clone from that backup, using an existing clone specification which you identify.

Cloning protected backups

Use SnapManager to clone a copy of a backup that has been protected to secondary storage.

About this task

Cloning can be done on the available backup copies present on secondary storage. Use the `-from-secondary` option to specify that you want to clone from secondary storage. If more than one copy exists, an arbitrary copy is selected.

Successful cloning of a protected backup requires that the host (selected for the clone) has access to the secondary storage over the same storage protocol (for example, SAN or NAS).

You can mount and clone a database from secondary storage if the backup has been freed; however, you cannot verify a backup from secondary storage.

Step

1. To create a clone of a protected backup, enter this command:

   smo clone create -backup-label *backup_name* -newsid *new_sid* -label *clone_label* -profile *profile_name* -clonespec *full_path_to_clonespecfile* -from-secondary -copy-id *id*

```
smo clone create -backup-label     backup_name
    -newsid     new_sid
    -label      clone_label
    -profile      profile_name
    -clonespec     full_path_to_clonespecfile
 -from-secondary -copy-id  id
```

Example

```
smo clone create -label testdb_clone_clstest
-profile sys_db_finance -from-secondary -copy-id sys_db_finance_sept_08
```

Considerations for cloning a database to an alternate host

Before you can clone to a host other than the one on which the database resides, there are environment considerations to think about and some prerequisite tasks to perform.

The following table shows the requirements for the source and target host setup:

Prerequisite set up	Requirement
Architecture	Must be the same on both the source and target hosts
Operating system and version	Must be the same on both the source and target hosts
SnapManager for Oracle	Must be installed and running on both the source and target hosts
Credentials	Must be set for the user to access the target host
Oracle	The same software version must be installed on both the source and target hosts. The Oracle Listener must be running on target host.
Compatible storage stack	Must be the same on both the source and target hosts
Protocol used to access data files	Must be the same on both the source and target hosts
Volume managers	Must be configured on both the source and target hosts and must be compatible versions

You can also clone an ASM database to a remote host. When doing so, make sure the ASM instance is running on the remote host.

Related information

The IBM support site - www.ibm.com/storage/support/nseries

Cloning a database to an alternate host

You can use the `clone create` command to clone a database backup on an alternate host.

Before you begin

- Create a profile or have an existing profile.
- Create a full backup or have an existing database backup.
- Create a clone specification or have an existing clone specification.

Step

1. To clone a database to an alternate host, enter the following command:

   ```
   smo clone create -backup-label backup_label_name -newsid new_sid -
   host target_host -label clone_label -comment comment_text -profile
   profile_name -clonespec full_path_to_clonespecfile
   ```

   Oracle does not let you run two databases with the same SID simultaneously on the same host. Because of this, you must supply a new SID for each clone. However, you can have a database on another host with the same SID.

Related tasks

Creating profiles on page 108
Cloning databases from backups on page 205
Creating clone specifications on page 199

Related references

The smo clone create command on page 284

Viewing a list of clones

You can view a list of clones associated with a particular profile.

About this task

The list includes the following information about the clones in a profile:

- The ID for the clone

- Status of the clone operation
- Oracle SID for the clone
- Host on which the clone resides
- Label for the clone

If you specify the `-verbose` option, the output also shows the comments entered for the clone.

Step

1. To display a list of all clones for a profile, enter the following command:

   `smo clone list -profile profile_name [-quiet | -verbose]`

Related references

The smo clone list command on page 288

Viewing detailed clone information

You can view detailed information about a specific clone.

About this task

The following information can be viewed using the `clone show` command:

- Clone SID and clone ID
- Clone operation status
- Clone create start and end date/time
- Clone label
- Clone comment
- Backup label and ID
- Source database
- Backup start and end time
- Database name, tablespaces, and data files
- Host name and file systems containing data files
- Storage system volumes and Snapshot copies backing the clone
- Whether the clone was created using the backup on primary or secondary storage

Step

1. To display the details of a specific clone, enter the following command:

   `smo clone show -profile profile_name [-label label | -id guid]`

Related references

The smo clone show command on page 289

Deleting clones

When you are finished working with a clone, you can delete it, using the `clone delete` command.

About this task

You might want to delete clones and clone the backup again when the size of the Snapshot copy reaches between 10% and 20% of the backup. Updating the clone guarantees that the clone has the most current data.

The clone SID and the clone label are not the same in SnapManager.

The label is the unique identifier for each clone in a profile. You can use the clone label or ID, but not the clone SID to delete the clone.

When you are deleting a clone, the database needs to be running. Otherwise, many files and directories for the existing clone will not be deleted, resulting in more work before the next clone can be created.

Note: Directories specified for certain Oracle parameters in the clone will be destroyed when the clone is deleted, and should only contain data for the cloned database: Archive Log Destinations, Background, Core, and User Dump Destinations. The audit files will not be deleted.

You cannot delete a running clone operation.

You can optionally collect the dump files after a successful or failed clone delete operation.

Step

1. To delete a clone, enter the following command:

   `smo clone delete -profile profile_name [-label label | -id guid] [-dump]`

Example

The following example shows the command to delete a cloned database:

```
smo clone delete -profile targetdb1_prof1 -label sales0908_clone1
```

Related references

The smo clone delete command on page 287

Performing management operations for SnapManager for Oracle

You can perform management tasks after you have set up and configured SnapManager. These tasks enable you to manage normal operations beyond backing up, restoring, and cloning.

Administrators can perform tasks with the SnapManager graphical user interface or by using the command-line interface. The *SnapManager for Oracle Installation and Administration Guide* provides instructions on how to complete these tasks using commands. The SnapManager online Help provides instructions on how to complete the tasks using the graphical user interface.

Viewing a list of operations

You can view a summary listing of all the operations performed against a profile.

About this task

You can view the following information when you list operations associated with a particular profile:

- Start and end date when the operation ran
- Operation status
- Operation ID
- Type of operation
- Host that it ran upon

Step

1. To list the summary information of all the operations, use the following command:

   `smo operation list profile -profile profile_name -delimiter character [-quiet | -verbose]`

   When the `-delimiter` option is specified, the command lists each row on a separate line and the attributes in that row are separated by the character specified.

Related references

The smo operation list command on page 321

Viewing operation details

You can view detailed information about a particular profile to verify the success or failure of an operation. It can also help you determine the storage resources in use for a particular operation.

About this task

You can view the following details about a particular operation:

- Operation ID
- Type of operation
- Whether the operation was forced
- Runtime information, including status, start and end date of the operation
- The host on which the operation ran, including the Process ID and SnapManager version
- Repository information
- Storage resources in use

Step

1. To view the detailed information for a specific operation ID, enter the following command:

   ```
   smo operation show -profile profile_name [-label label | -id id] [-quiet
   | -verbose]
   ```

Related references

The smo operation show command on page 322

Issuing commands from an alternate host

You can issue CLI commands from a host other than the database host and SnapManager will route the commands you enter to the appropriate host.

About this task

For the system to dispatch an operation to the correct host, it must first know where to find the profile for the operation. In this procedure the system keeps the profile to repository mapping information for a file in the user's home directory on the local host.

Step

1. To make the local user's home directory aware of the profile-to-repository mappings so it can route the operation request, enter the following command:

```
smo profile sync -repository -dbname repo_dbname -host repo_host -
port repo_port -login -username repo_username [-quiet | -verbose]
```

Checking the SnapManager software version

You can determine which version of the product you are running on your local host by running the `version` command.

Step

1. To check the SnapManager version, enter this command:

   ```
   smo version
   ```

Related references

The smo version command on page 360

Stopping the SnapManager host server

When you have finished using SnapManager, you might want to stop the server.

Step

1. To stop the server, enter the following command, as a root user:

   ```
   smo_server stop
   ```

Related references

The smo_server stop command on page 263

Restarting the SnapManager Windows host server

You can restart the server on a Windows host using Windows services.

Steps

1. Go to **Start > Control Panel > Administrative Tools > Services**.
2. With the Services window open, select Ontap SnapManager 3.1 for Oracle.
3. You can restart the server in one of the following ways:
   a. In the left panel, click **Restart**.
   b. Right-click Ontap SnapManager 3.1 for Oracle and select **Restart** from the drop-down menu.

c. Double-click Ontap SnapManager 3.1 for Oracle and in the properties window that opens, click **Restart**.

Uninstalling the software from a Windows host

You may need to uninstall your SnapManager software from the host server.

About this task

If the server is currently running, stop the server before uninstalling the GUI software.

Steps

1. Go to **Start > Control Panel**.
2. Choose to remove **SnapManager for Oracle**.
3. Choose **Uninstall.**

Configuring e-mail notification

SnapManager enables you to receive an e-mail notification about the completion status of the profile-executed database operations. SnapManager generates the e-mail and helps you to take appropriate action based on the database operation completion status. Configuring e-mail notification is an optional parameter.

You can configure e-mail notification in two ways: for an individual profile as profile notification and for multiple profiles on a repository database as summary notification.

Profile notification

For an individual profile, you can receive an e-mail for either or both the successful and failed database operations.

Note: By default, e-mail notification is enabled for failed database operations.

Summary notification

Summary notification enables you to receive a summary e-mail on database operations performed using multiple profiles by enabling summary notification options such as hourly, daily, weekly, or monthly.

You can use either profile-level notification or summary notification at a time.

SnapManager enables e-mail notification for the following profile-executed database operations:

- Create backup on primary storage
- Restore backups
- Create clones
- Split clones
- Verify backups

After you create or update profiles with the e-mail notification enabled, you can disable it. If you disable the e-mail notification, you no longer receive e-mail alerts for those profile-executed database operations.

The e-mail that you receive contains the following details:

- Name of the database operation, for example, backup, restore, or clone.
- Profile name used for the database operation.
- Name of the host server.
- System identifier (SID) of the database.
- Start and end time of the database operation.
- Success or failed status of the database operation.
- Error message, if any.

You can perform the following tasks related to e-mail notification:

- Configure the mail server for a repository.
- Configure e-mail notification for a new profile.
- Configure e-mail notification for an existing profile.
- Configure summary e-mail notification for multiple profiles under a repository.

Note: You can configure e-mail notification from both CLI and GUI.

Configuring a mail server for a repository

SnapManager enables you to specify the mail server details from which the e-mail alerts are sent.

About this task

SnapManager enables you to specify the sender's e-mail server host name or IP address, and the e-mail server port number for a repository database name that requires e-mail notification. You can configure the mail server port number in a range from 0 through 65535; the default value is 25. If you require authentication for the e-mail address, you can specify the user name and password.

You must specify name or IP address of the host server that handles the e-mail notification.

Step

1. To configure the mail server to send e-mail alerts, enter the following command:

   `smo notification set -sender-email email_address -mailhost mailhost -mailport mailport [-authentication -username username -password password] -repository -port repo_port -dbname repo_service_name -host repo_host -login -username repo_username`

   Other options for this command are as follows:

   `[-force]`

   `[quiet | -verbose]`

To do the following...	Then...
To specify the sender's e-mail address.	Specify the `-sender-email` option. From SnapManager 3.2 for Oracle, you can include hyphen (-) while specifying the domain name of the e-mail address. For example, you can specify the sender e-mail address as `-sender-email user@org-corp.com`.
To specify the sender's e-mail server host name or IP address.	Specify the `-mailhost` option.

To do the following...	Then...
To specify the e-mail server port number for a repository database name that requires e-mail notification. You can configure the mail server port number in a range from zero through 65535; the default value is 25.	Specify the `-mailport` option.
Specify the user name and password if you require authentication for the e-mail address.	Specify `-authentication` option followed by the user name and password.

The following example configures the mail server.

```
smo notification set -sender-email admin1@org.com -mailhost hostname.org.com -mailport 25
authentication -username admin1 -password admin1 -repository -port 1521 -dbname SMOREPO -
host hotspur -login -username grabal21 -verbose
```

Configuring e-mail notification for a new profile

SnapManager enables you to receive an e-mail notification on the completion status of the database operation while creating a new profile.

Before you begin

Before configuring an e-mail notification, ensure you configure the e-mail address from which the e-mail alerts are sent.

About this task

For multiple e-mail addresses, enter the e-mail addresses separated by commas and ensure you do not provide any space between the comma and the next e-mail address.

Note: On Windows platform, ensure you enter the whole set of e-mail addresses within double quotes.

When a backup of datafiles and archive log files are taken together using the profile (for creating separate archive log backups), and the datafile itself creation fails, the e-mail notification is sent with the `operation name` as `Data Backup` instead

`Data Backup and Archive Logs Backup`.

Sample profile notification result when datafile and archive log file backup operation is successful

```
Profile Name    : PROF_31
Operation Name     : Data Backup and Archive Logs Backup
Database SID       : TENDB1
Database Host      : repo1.rtp.org.com
Start Date         : Fri Sep 23 13:37:21 EDT 2011
End Date           : Fri Sep 23 13:45:24 EDT 2011
```

```
Status              : SUCCESS
Error messages      :
```

Steps

1. To add an e-mail notification for a new profile from the SnapManager CLI, enter the following command:

   smo profile create -profile *profile* [-profile-password *profile_password*] -repository -dbname *repo_service_name* -host *repo_host* -port *repo_port* -login -username *repo_username* -database -dbname *db_dbname* -host *db_host* [-sid *db_sid*] [-login -username *db_username* -password *db_password* -port *db_port*] [-rman {-controlfile | {-login -username *rman_username* -password *rman_password* -tnsname *rman_tnsname*} }] -osaccount *osaccount* -osgroup *osgroup* [-retain [-hourly [-count *n*] [-duration *m*]] [-daily [-count *n*] [-duration *m*]] [-weekly [-count *n*] [-duration *m*]] [-monthly [-count *n*] [-duration *m*]]] [-comment *comment*][-snapname-pattern *pattern*][-protect [-protection-policy *policy_name*]][-notification [-success -email *email_address1,email_address2* -subject *subject_pattern*] [-failure -email *email_address1,email_address2* -subject *subject_pattern*]]

   Other options for this command are as follows:

   `[-force]`

   `[quiet | -verbose]`

   Note: SnapManager supports up to 1000 characters for e-mail addresses from the SnapManager CLI.

The following example displays the e-mail notification configured while creating a new profile:

```
smo profile create -profile sales1 -profile-password sales1 -
repository -dbname repo2 -host 10.72.197.133 -port 1521 -login -
username oba5 -database -dbname DB1 -host 10.72.197.142 -sid DB1 -
osaccount oracle
-osgroup dba -notification -success -email admin1@org.com -subject
```

```
{profile}_{operation-name}_{db-sid}_{db-host}_{start-date}_{end-
date}_{status}
```

Customizing the e-mail subject for a new profile

SnapManager enables you to customize an e-mail subject pattern for the profile when you create a new profile.

About this task

You can customize the e-mail subject using the pattern: {profile}_{operation-name}_{db-sid}_{db-host}_{start-date}_{end-date}_{status} or enter your own text.

Variable name	Description	Example value
profile	Profile name used for the database operation.	PROF1
operation-name	Database operation name.	Backup, Data Backup, Data and Archive Logs Backup.
db-sid	SID of the database.	DB1
db-host	Name of the host server.	hostA
start-date	Start time of the database operation in MMdd:HH:SS yyyy format.	Apr 27 21:00:45 PST 2010
end-date	End time of the database operation in MMdd:HH:SS yyyy format.	Apr 27 21:10:45 PST 2010
status	Database operation status.	Success

If you do not provide any variables, then SnapManager displays an error message "Missing value(s) -subject".

Step

1. To customize the e-mail subject, enter the following command:

```
smo profile create -profile profile [-profile-password profile_password]
-repository -dbname repo_service_name -host repo_host -port repo_port -
login -username repo_username -database -dbname db_dbname -host db_host
[-sid db_sid] [-login -username db_username -password db_password -port
db_port] [-rman {-controlfile | {-login -username rman_username -
password rman_password -tnsname rman_tnsname} } ] -osaccount osaccount -
osgroup osgroup [-retain [-hourly [-count n] [-duration m]] [-daily [-
count n] [-duration m]] [-weekly [-count n] [-duration m]] [-monthly [-
count n] [-duration m]]] [-comment comment][-snapname-pattern pattern][-
```

```
protect [-protection-policy policy_name]] [-notification [-success -
email email_address1,email_address2 -subject subject_pattern] [-failure
-email email_address1,email_address2 -subject subject_pattern]]
```

Example showing the e-mail subject pattern

```
smo profile create -profile sales1 -profile-password admin1 -
repository -dbname repo2 -host 10.72.197.133 -port 1521 -login -
username admin2 -database -dbname DB1 -host 10.72.197.142 -sid DB1
-osaccount oracle -osgroup dba -profile-notification -success -email
admin@org.com -subject {profile}_{operation-name}_{db-sid}_{db-
host}_{start-date}_{end-date}_{status}
```

Configuring e-mail notification for an existing profile

SnapManager enables you to receive an e-mail notification on the completion status of the database operation while updating a new profile.

Before you begin

Before configuring an e-mail notification, ensure you configure the e-mail address from which the e-mail alerts are sent.

Step

1. To add an e-mail notification for an existing profile from the SnapManager CLI, enter the `profile-notification` command:

   ```
   smo profile update -profile profile [-profile-password profile_password]
   [-database -dbname db_dbname -host db_host [-sid db_sid] [-login -
   username db_username -password db_password -port db_port]] [{-rman{-
   controlfile | {-login  -username rman_username -password  rman_password
   -tnsname  rman_tnsname}}} | -remove-rman]-osaccount osaccount -osgroup
   osgroup [-retain [-hourly [-count n] [-duration m]] [-daily [-count n]
   [-duration m]] [-weekly [-count n] [-duration m]] [-monthly [-count n]
   [-duration m]]] [-comment comment][-snapname-pattern pattern][[-protect
   [-protection-policy policy_name]]| [[-noprotect]] [-notification [-
   success -email email_address1,email_address2 -subject subject_pattern]
   [-failure -email email_address1,email_address2 -subject
   subject_pattern]]
   ```

   - Use the `success` option to receive an e-mail notification only for successful database operations and the `failure` option to receive an e-mail notification only for failed database operations.
   - You can use both success and failure options to receive the e-mail notification for successful and failed database operations.

You must enter a single e-mail address or multiple e-mail addresses to which e-mail alerts will be sent. For multiple e-mail addresses, enter the e-mail addresses separated by commas and ensure you do not provide any space between the comma and the next e-mail address. You can add a subject to the e-mail as well.

Note: On Windows platform, ensure you enter the whole set of e-mail addresses within double quotes.

Customizing the e-mail subject for an existing profile

SnapManager enables you to customize the e-mail subject pattern for an existing profile by updating that profile. This customized subject pattern is applicable only for the updated profile.

Step

1. To customize the e-mail subject for an existing profile, enter the following command:

```
smo profile update -profile profile [-profile-password profile_password]
[-database -dbname db_dbname -host db_host [-sid db_sid] [-login -
username db_username -password db_password-port db_port]] [{-rman{-
controlfile | {-login  -username rman_username -password  rman_password
-tnsname  rman_tnsname}}} | -remove-rman]-osaccount osaccount -osgroup
osgroup [-retain [-hourly [-count n] [-duration m]] [-daily [-count n]
[-duration m]] [-weekly [-count n] [-duration m]] [-monthly [-count n]
[-duration m]]] [-comment comment][-snapname-pattern pattern][[-protect
[-protection-policy policy_name]]| [[-noprotect]] [-notification [-
success -email email_address1,email_address2 -subject subject_pattern]
[-failure -email email_address1,email_address2 -subject
subject_pattern]]
```

Example showing the e-mail subject pattern.

```
smo profile update -profile sales1 -profile-password sales1 -
repository -dbname repo2 -host 10.72.197.133 -port 1521 -login -
username admin2 -database -dbname DB1 -host 10.72.197.142 -sid DB1
-osaccount oracle -osgroup dba -profile-notification -success -email
```

```
admin@org.com -subject {profile}_{operation-name}_{db-sid}_{db-
host}_{start-date}_{end-date}_{status}
```

Configuring summary e-mail notification for multiple profiles

SnapManager enables you to configure a summary e-mail notification for multiple profiles under a repository database.

About this task

From the SnapManager CLI, enter the `notification update-summary-notification` command to enable a summary level e-mail notification. For this, you must enter the repository database name and the profile names under the repository database. The e-mail notification is enabled only for the profiles selected. In addition, enter the name or IP address of mail server.

You can select any one of the schedule time at which you require an e-mail notification:

- Hourly: Select the time from when you will receive an hourly e-mail.
- Daily: Select the specified time you will receive a daily e-mail notification.
- Weekly: Select any day and time, you will receive weekly e-mail notification.
- Monthly: Select date in a month and time, when you will receive monthly e-mail notification.

You need to enter a single e-mail address or multiple e-mail addresses of recipients separated by commas to receive e-mail notification for operations performed using those profiles. Ensure that there is no space between the comma and the next e-mail address when you enter multiple e-mail addresses.

Note: On Windows platform, ensure you enter the whole set of multiple e-mail addresses within double quotes.

SnapManager allows you to add a customized e-mail subject using the following variables:

- Profile name used for the database operation.
- Database name.
- SID of the database.
- Name of the host server.
- Start time of the database operation in yyyyMMdd:HH:SS format.
- End time of the database operation in yyyyMMdd:HH:SS format.
- Database operation status.

If you select not to add a customized subject, SnapManager displays an error message "Missing value -subject".

Step

1. To enable summary notification for a repository database, enter this command:

   smo notification update-summary-notification -repository -port *repo_port* -dbname *repo_service_name* -host *repo_host* -login -username *repo_username* -email *email_address1,email_address2* -subject *subject-pattern* -frequency {-daily -time *daily_time* | -hourly -time *hourly_time* | -monthly -time *monthly_time* -date {1|2...|31} | -weekly -time *weekly_time* -day {1|2|3|4|5|6|7}} -profiles *profile1 profile2*

   Other options for this command are as follows:

   `[-force] [-noprompt]`

   `[quiet | -verbose]`

```
smo notification update-summary-notification –repository –port 1521 –
dbname repo2 –host 10.72.197.133 –login –username oba5 –email-address
admin@org.com –subject success –frequency -daily -time 19:30:45 –
profiles sales1
```

Adding a new profile to summary e-mail notifications

After you configure a summary e-mail notification for the repository database, you can add a new profile to summary notification using the `summary notification` command.

Step

1. To add a new profile for a summary level e-mail notification, enter the following command:

   smo profile create -profile *profile_name* [-profile-password *profile_password*] -repository -dbname *repo_service_name* -host *repo_host* -port *repo_port* -login -username *repo_username* -database -dbname *db_dbname* -host *db_host* [-sid *db_sid*] [-login -username *db_username* -password *db_password* -port *db_port*] [-rman {-controlfile | {-login -username *rman_username* -password *rman_password* -tnsname *rman_tnsname*} }] -osaccount *osaccount* -osgroup *osgroup* [-retain [-hourly -count *n*] [-duration *m*]] [-daily -count *n*] [-duration *m*]] [-weekly -count *n*] [-duration *m*]] [-monthly -count *n*] [-duration *m*]]] [-comment *comment*][-snapname-pattern *pattern*][-protect [-protection-policy *policy_name*]] [-summary-notification]

   Other options for this command are as follows:

   `[-force]`

   `[quiet | -verbose]`

Adding an existing profile to summary e-mail notification

SnapManager enables you to add an existing profile to a summary e-mail notification while updating that profile.

Step

1. To add an existing profile for a summary level e-mail notification, enter the following command:

   smo profile update -profile *profile_name* [-profile-password *profile_password*] -repository -dbname *repo_service_name* -host *repo_host* -port *repo_port* -login -username *repo_username* -database -dbname *db_dbname* -host *db_host* [-sid *db_sid*] [-login -username *db_username* -password *db_password* -port *db_port*] [-rman {-controlfile | {-login -username *rman_username* -password *rman_password* -tnsname *rman_tnsname*} }] -osaccount *osaccount* -osgroup *osgroup* [-retain [-hourly -count *n*] [-duration *m*]] [-daily -count *n*] [-duration *m*]] [-weekly -count *n*] [-duration *m*]] [-monthly -count *n*] [-duration *m*]]] [-comment *comment*][-snapname-pattern *pattern*][-protect [-protection-policy *policy_name*]] [-summary-notification]

Disabling e-mail notification for multiple profiles

After you enable the summary e-mail notification for multiple profiles, you can disable them to no longer receive e-mail alerts.

About this task

SnapManager enables you to disable the summary e-mail notification for those profile-executed database operations. From the SnapManager CLI, enter the `notification remove-summary-notification` command to disable the summary e-mail notification for multiple profiles and the repository database name to which you do not require any e-mail notification.

Step

1. To disable summary notification for multiple profiles on a repository database, enter the following command:

   smo notification remove-summary-notification -repository -port *repo_port* -dbname *repo_service_name* -host *repo_host* -login -username *repo_username*

The following example disables summary notification for multiple profiles on a repository database:

```
smo notification remove-summary-notification –repository –port 1521 –
dbname repo2 –host 10.72.197.133 –login –username oba5
```

Creating task specification and scripts for SnapManager operations

SnapManager for Oracle uses a task specification Extensible Markup Language (XML) file that indicates the pre-tasks and post-tasks of the backup, restore, and the clone operations. You can add the pre-task and post-task script names in the XML file to perform the task before or after the backup, restore, and clone operations take place.

SnapManager versions prior to SnapManager 3.2 version have the ability to run the pre-task and post-task scripts only for the clone operation. As of SnapManager 3.2 for Oracle, one can run the pre-task and post-task scripts for the backup, restore, and the clone operations.

The clone specification XML file prior to the SnapManager 3.2 version contains the task specification section as a part of the clone specification XML file. As of SnapManager 3.2 for Oracle version, the task specification section is provided as a separate task specification XML file. If you are using the earlier clone specification XML file in the SnapManager 3.2 for Oracle, you must remove the task specification tag (`<task-specification>`) from the clone specification XML or create a new clone specification XML file.

The SnapManager 3.2 for Oracle requires the following tasks to be accomplished for the SnapManager operations:

- For the backup operation, use the task specification XML file.
- For the restore operation, use the task specification XML file.
- For the clone operation, provide two specification files: a clone specification XML file and a task specification XML file. If you want to enable pre-task or post-task activity, you can add the task specification XML file on an optional basis.

You can create the task specification file by using the SnapManager graphical user interface, command-line interface, or a text editor. Use an extension of .XML for the file to enable appropriate editing features. You might want to save this file so that you can use it for future backup, restore, and clone operations.

The task specification XML file includes two sections:

- The pre-tasks section includes scripts that could be run before the backup, restore, and operations.
- The post-tasks section includes scripts that could be run after the backup, restore, and clone operations.

The values included in the pre-tasks and post-tasks sections must adhere to the following guidelines:

- Task name: The name of the task (in `<task><name>`) must match name of the script, which shows in response to the `plugin.sh -describe` command.

  Note: If there is a mismatch between the name in the `<task><name>` option and the pre-task or post-task script name, then you might receive an error message: `the file not found`.

- Parameter name: The name of the parameter must be a string that can be used as an environment variable setting. The string must match the parameter name in the custom script, which shows in response to the `plugin.sh -describe` command.

```
<task-specification>
  <pre-tasks>
<task>
      <name>name</name>
      <parameter>
          <name>name</name>
          <value>value</value>
       </parameter>
    </task>
  </pre-tasks>
  <post-tasks>
    <task>
      <name>name</name>
      <parameter>
          <name>name</name>
          <value>value</value>
       </parameter>
    </task>
  </post-tasks>
</task-specification>
```

Note: The task specification XML file should not contain any policy task.

From the SnapManager GUI, you can set the parameter value and save the XML file. Using the **Task Enabling** page of the **Backup Create** wizard, **Restore or Recovery** wizard, and **Clone Create** wizard, you can load the existing task specification XML file, and use the selected file for the pre-task or post-task activity.

A task can be executed multiple times, either with the same or different parameter and value combinations. For example, you could use a "Save" task to save multiple files.

Note: SnapManager uses the XML tags provided within the task specification file for the pre-processing or post-processing activity for the backup, restore, and the clone operations irrespective of the file extension of the task specification file.

Creating pre-task, post-task, and policy scripts for SnapManager operations

SnapManager enables you to create the scripts for the pre-processing activity, the post-processing activity, and policy tasks of the backup, restore, and the clone operations. You must place the scripts

in the correct installation directory to execute the pre-processing activity, post-processing activity, and policy tasks of the SnapManager operation.

About this task

Pre-task and post-task script content

Scripts include the following:

- Specific operations (check, describe, and execute)
- Optional, predefined environment variables
- Specific error handling (return code (rc))

Note: You must perform proper error handling to validate the script. The return error handling should be correct.

You can use the pre-task scripts for many purposes, for example, cleaning up a disk space before the SnapManager operation starts. You can also use the post-task scripts, for instance, to estimate whether SnapManager for Oracle has enough disk space to complete the operation.

Policy task script content

You can execute the policy script without using specific operations such as check, describe, and execute. The script includes the predefined environmental variables (optional) and specific error handling.

Policy task is executed before the backup, restore, and clone operations.

Script formats supported on Windows and UNIX

Use a command file with an extension of .cmd for the Windows platform and a shell script file with an extension of .sh for the UNIX platform as the pre-script and post-script.

Note: If you select the shell script file for the Windows platform, the SnapManager wizard operation you have started fails to respond. To resolve this, you must close the wizard, provide the command file in the plug-in directory, and perform the SnapManager operation again.

Script installation directory

The directory where you install the script affects how it is used. You can place the scripts in the directory and execute the script before or after the backup, restore, or clone operation take place. You must place the script in the directory specified in the table and use it on an optional basis when you specify the backup, restore, or clone operation.

Note: Ensure that the `plugins` directory has the executable permission before using the scripts for the SnapManager operation.

Table 3: Pre-task, post-task, policy task script installation path

Activities	Backup	Restore	Clone

Pre-processing activities	<default_installation_directory>/plugins/backup/create/pre	<default_installation_directory>/plugins/restore/create/pre	<default_installation_directory>/plugins/clone/create/pre
Post-processing activities	<default_installation_directory>/ plugins/backup/create/post	<default_installation_directory>/plugins/restore/create/post	<default_installation_directory>/plugins/clone/create/post
Policy-based activities	<default_installation_directory>/plugins/backup/create/policy	<default_installation_directory>/plugins/restore/create/policy	<default_installation_directory>/plugins/clone/create/policy

Sample scripts

You can find some samples of the pre-task and the post-task scripts for the backup and clone operations available at the installation directory path:

- <default_installation_directory>/plugins/examples/backup/create/pre
- <default_installation_directory>/plugins/examples/backup/create/post
- <default_installation_directory>/plugins/examples/clone/create/pre
- <default_installation_directory>/plugins/examples/clone/create/post

What you can change in the script

If you are creating a new script, you can change only the describe and execute operations.

Each script must contain the following variables: context, timeout, and parameter.

The variables you have described in the describe function must be declared at the start of the script. You can add new parameter values in the `parameter=()` and then use the parameters in the execute function.

Sample script

Sample script with user specified return code for estimating the space in the SnapManager host.

```
    #!/bin/bash
    # $Id: //depot/prod/capstan/main/src/plugins/unix/examples/backup/
create/pre/disk_space_estimate.sh#5 $
    name="disk space estimation ($(basename $0))"
    description="pre tasks for estimating the space on the target
system"
    context=
    timeout="0"
    parameter=()
    EXIT=0
    PRESERVE_DIR="/tmp/preserve/$(date +%Y%m%d%H%M%S)"
    function _exit {
       rc=$1
       echo "Command complete."
       exit $rc
```

```
    }
    function usage {
       echo "usage: $(basename $0) { -check | -describe | -execute }"
       _exit 99
    }
    function describe {
       echo "SM_PI_NAME:$name"
       echo "SM_PI_DESCRIPTION:$description"
       echo "SM_PI_CONTEXT:$context"
       echo "SM_PI_TIMEOUT:$timeout"
       IFS=^
       for entry in ${parameter[@]}; do
           echo "SM_PI_PARAMETER:$entry"
       done
       _exit 0
    }
    function check {
       _exit 0
    }
    function execute {
       echo "estimating the space on the target system"
       # Shell script to monitor or watch the disk space
       # It will display alert message, if the (free available)
percentage
       # of space is >= 90%
       #
--------------------------------------------------------------------------
       # Linux shell script to watch disk space (should work on other
UNIX oses )
       # set alert level 90% is default
       ALERT=90
       df -H | grep -vE '^Filesystem|tmpfs|cdrom' | awk '{ print $5 "
" $1 }' | while read output;
       do
         #echo $output
         usep=$(echo $output | awk '{ print $1}' | cut -d'%' -f1  )
         partition=$(echo $output | awk '{ print $2 }' )
       if [ $usep -ge $ALERT ]; then
           echo "Running out of space \"$partition ($usep%)\" on $
(hostname) as on $(date)" |
       fi
       done
      _exit 0
     }
    function preserve {
        [ $# -ne 2 ] && return 1
        file=$1
        save=$(echo ${2:0:1} | tr [a-z] [A-Z])
        [ "$save" == "Y" ] || return 0
        if [ ! -d "$PRESERVE_DIR" ] ; then
           mkdir -p "$PRESERVE_DIR"
           if [ $? -ne 0 ] ; then
               echo "could not create directory [$PRESERVE_DIR]"
               return 1
           fi
        fi
        if [ -e "$file" ] ; then
```

```
            mv "$file" "$PRESERVE_DIR/."
        fi
        return $?
    }
    case $(echo $1 | tr [A-Z] [a-z]) in
        -check)    check
                   ;;
        -execute)  execute
                   ;;
        -describe) describe
                   ;;
     *)          echo "unknown option $1"
                 usage
                 ;;
     esac
```

Operations in task scripts

The pre-task and post-task scripts must include the following operations:

Operation	Description
check	The SnapManager server uses this operation to ensure it has execute permission on the plug-in scripts. You might also include file permission checking on the remote system.

Operation	Description

describe	The SnapManager server uses this operation to obtain information about your script and to match the elements provided by the clone specification file. Your plug-in script should echo the following description information to standard output:
	• `SM_PI_NAME`: Script name
	• `SM_PI_DESCRIPTION`: Description of the script's purpose
	• `SM_PI_CONTEXT`: User context in which the script should run, for example, `root` or `oracle`.
	• `SM_PI_TIMEOUT`: The maximum time (in milliseconds) that SnapManager should wait for the script to complete processing and return before terminating its execution.
	• `SM_PI_PARAMETER`: One or more custom user parameters necessary for your plug-in script to perform its processing. Each parameter should be listed in a new output line and include the name of the parameter and a description. Upon execution, the parameter value will be provided to your script via an environment variable of the same name as the parameter. Parameter names and descriptions appear to users.
	The following shows a sample of the -describe output. The script name is "Followup_activities."
	`plugin.sh - describe`
	`SM_PI_NAME:Followup_activities`
	`SM_PI_DESCRIPTION:this script contains follow-up activities to be executed after the clone create operation.`
	`SM_PI_CONTEXT:root`
	`SM_PI_TIMEOUT:60000`
	`SM_PI_PARAMETER:SCHEMAOWNER:Name of the database schema owner.`
	`Command complete.`

execute	The SnapManager server uses this operation to invoke your script to execute the function.

SnapManager requests these operations via the following command line invocations of your script:

- `plugin.sh -check`
- `plugin.sh -describe`
- `plugin.sh -execute`

Variables available in the task scripts for backup operation

SnapManager provides context information in the form of environment variables related to the backup operation being performed. For example, your script can retrieve the name of the original host, the name of the retention policy, and the label of the backup.

The following table lists the environment variables that you can use in your scripts:

Variables	Description	Format
SM_OPERATION_ID	ID of the current operation	string
SM_PROFILE_NAME	Name of the profile used	string
SM_SID	SID of the database	string
SM_HOST	Hostname of the database	string
SM_OS_USER	OS owner of the database	string
SM_OS_GROUP	OS group of the database	string
SM_BACKUP_TYPE	Type of the backup (online/offline/auto)	string
SM_BACKUP_LABEL	Label of the backup	string
SM_BACKUP_ID	ID of the backup	string
SM_BACKUP_RETENTION	Retention period	string
SM_BACKUP_PROFILE	Profile used for this backup	string
SM_ALLOW_DATABASE_SHUTDOWN	Allow database startup or shutdown, if necessary use the -force option from the command-line interface.	boolean
SM_BACKUP_SCOPE	Scope of the backup, full or partial	string
SM_BACKUP_PROTECTION	Backup protection	boolean
SM_TARGET_FILER_NAME	Target storage system name **Note:** If more than one storage system is used, then the storage system names must be separated by commas.	string
SM_TARGET_VOLUME_NAME	Target volume name **Note:** The target volume name must be prefixed with storage device name, for example, SM_TARGET_FILER_NAME/ SM_TARGET_VOLUME_NAME.	string
SM_HOST_FILE_SYSTEM	Host file system	string

Variables	Description	Format
SM_SNAPSHOT_NAMES	Snapshot list **Note:** Name of the Snapshot copies must be prefixed with the storage system name and volume name. Names of the Snapshot copies are separated by commas.	string array
SM_ASM_DISK_GROUPS	ASM Disk group list	string array
SM_ARCHIVE_LOGS_DIRECTORY	Archive logs directory **Note:** If the archive logs are located in more than one directory, then the names of those directories are separated by commas.	string array
SM_REDO_LOGS_DIRECTORY	Redo logs directory **Note:** If the redo logs are located in more than one directory, then the names of those directories are separated by commas.	string array
SM_CONTROL_FILES_DIRECTORY	Control files directory **Note:** If the control files are located in more than one directory, then the names of those directories are separated by commas.	string array
SM_DATA_FILES_DIRECTORY	Data files directorySM_DATA_FILES_DIRECTORY **Note:** If the data files are located in more than one directory, then the names of those directories are separated by commas.	string array
<user defined>	Additional parameters defined by the user in the input definition mechanism for the plug-in is passed directly to the plug-in. Plug-ins used as policies should not expect any user defined parameters to be available.	user defined

Variables available in the task scripts for restore operation

SnapManager provides context information in the form of environment variables related to the backup operation being performed. For example, your script can retrieve the name of the original host, the name of the retention policy, and the label of the backup.

The following table lists the environment variables that you can use in your scripts:

Variables	Description	Format
SM_OPERATION_ID	ID of the current operation	string
SM_PROFILE_NAME	Name of the profile used	string
SM_HOST	Hostname of the database	string
SM_OS_USER	OS owner of the database	string
SM_OS_GROUP	OS group of the database	string
SM_BACKUP_TYPE	Type of the backup (online/offline/auto)	string
SM_BACKUP_LABEL	Backup label	string
SM_BACKUP_ID	Backup ID	string
SM_BACKUP_PROFILE	Profile used for this backup	string
SM_RECOVERY_TYPE	Recovery configuration information	string
SM_VOLUME_RESTORE_MODE	Volume restore configuration	string
SM_TARGET_FILER_NAME	Target Filer name **Note:** If more than one storage system is used, then the storage system names must be separated by commas.	string
SM_TARGET_VOLUME_NAME	Target Volume name **Note:** The target volume name must be prefixed with storage device name, for example, SM_TARGET_FILER_NAME/ SM_TARGET_VOLUME_NAME.	string
SM_HOST_FILE_SYSTEM	Host file system	string
SM_SNAPSHOT_NAMES	Snapshot list **Note:** Name of the Snapshot copies must be prefixed with the storage system name and volume name. Names of the Snapshot copies are separated by commas.	string array
SM_ASM_DISK_GROUPS	ASM Disk group list	string array

Variables	Description	Format
SM_ARCHIVE_LOGS_DIRECTORY	Archive logs directory **Note:** If the archive logs are located in more than one directory, then the names of those directories are separated by commas.	string array
SM_REDO_LOGS_DIRECTORY	Redo logs directory **Note:** If the redo logs are located in more than one directory, then the names of those directories are separated by commas.	string array
SM_CONTROL_FILES_DIRECTORY	Control files directory **Note:** If the control files are located in more than one directory, then the names of those directories are separated by commas.	string array
SM_DATA_FILES_DIRECTORY	Data files directory **Note:** If the data files are located in more than one directory, then the names of those directories are separated by commas.	string array

Variables available in the task scripts for clone operation

SnapManager provides context information in the form of environment variables related to the clone operation being performed. For example, your script can retrieve the name of the original host, the name of the clone database, and the label of the backup.

The following table lists the environment variables that you can use in your scripts:

Variables	Description	Format
SM_ORIGINAL_SID	SID of the original database	string
SM_ORIGINAL_HOST	Hostname associated with the original database	string
SM_ORIGINAL_OS_USER	OS owner of the original database	string
SM_ORIGINAL_OS_GROUP	OS group of the original database	string
SM_TARGET_SID	SID of the clone database	string
SM_TARGET_HOST	Hostname associated with the clone database	string
SM_TARGET_OS_USER	OS owner of the clone database	string
SM_TARGET_OS_GROUP	OS group of the clone database	string

Variables	Description	Format
SM_TARGET_DB_PORT	Port of the target database	integer
SM_TARGET_GLOBAL_DB_NAME	Global database name of the target database	string
SM_BACKUP_LABEL	Label of the backup used for the clone	string

Error handling in custom scripts

SnapManager processes the custom script based on the specific return codes. For example, if your custom script returns a value of 0, 1, 2, or 3, SnapManager continues with the clone process. The return code also influences how SnapManager processes and returns the standard output of your script execution.

Return code	Description	Continue processing the operation
0	The script completed successfully.	Yes
1	The script completed successfully, with informational messages.	Yes
2	The script completed, but includes warnings	Yes
3	The script fails, but the operation continues.	Yes
4 or >4	The script fails and the operation stops.	No

Viewing sample plug-in scripts

SnapManager includes scripts that you can use as examples to learn how to make your own or use as a basis for your new custom scripts.

About this task

The plug-in sample scripts are stored in the following directory:

```
<default_install_directory>/plugins/examples/backup/create
<default_install_directory>/plugins/examples/clone/create
<default_install_directory>/plugins/unix/examples/backup/create/post
<default_install_directory>/plugins/windows/examples/backup/create/post
```

This directory includes the following subdirectories:

- policy: Contains scripts that, when configured, always run on the clone operation.
- pre: Contains scripts that, when configured, run before the clone operation.
- post: Contains scripts that, when configured, run after the clone operation.

The following table describes the sample scripts and lists their locations:

Script name	Description	Type of script
validate_sid.sh	This script contains additional checks to the SID used on the target system. The script checks that the SID has the following: • Contains exactly three alphanumeric characters • Begins with a letter • Does not include reserved SAP SIDs	Policy
cleanup.sh	Cleans up the target system so that it is ready to take a clone. Preserves files and directories, if wanted, or deletes them.	Pre-task
sap_follow_up_activities.sh	Performs follow-up activity tasks as described in SAP System Copy Guide and TR-3442. For example, this script deletes or modifies table entries in the SAP schema.	Post-task
Mirror_the_backup.sh	Mirror the volumes after the backup operation occurs on an UNIX-based environment.	Post-task
Mirror_the_backup.cmd	Mirror the volumes after the backup operation occurs on a Windows environment.	Post-task
Vault_the_backup.sh	Vault the qtrees after the backup operation occurs on the UNIX environment.	Post-task
Vault_the_backup.cmd	Vault the qtrees after the backup operation occurs on the Windows environment.	Post-task

Scripts delivered with SnapManager use the BASH shell by default. Check that support for the BASH shell is installed on your operating system before attempting to run any of the sample scripts.

Steps

1. To verify that you are using the BASH shell, enter the following command at the command prompt: `bash`

   If you do not see an error, the BASH shell is operating properly.

   Alternately, you can enter the `which-bash` command at the command prompt.

2. Locate the script in the following directory:

   `<installdir>/plugins/examples/clone/create`

3. Open the script in a script editor such as vi.

Sample script

The following sample custom script (validate_sid.sh) validates database SID names and prevents invalid names from being used in the cloned database. It includes the three operations (check, describe, and execute) and calls them at the end of the script. The script also includes error message handling with codes of 0, 4 and >4.

```
EXIT=0
name="Validate SID"
description="Validate SID used on the target system"
parameter=()

# reserved system IDs
INVALID_SIDS=("ADD" "ALL" "AND" "ANY" "ASC"
              "COM" "DBA" "END" "EPS" "FOR"
              "GID" "IBM" "INT" "KEY" "LOG"
              "MON" "NIX" "NOT" "OFF" "OMS"
              "RAW" "ROW" "SAP" "SET" "SGA"
              "SHG" "SID" "SQL" "SYS" "TMP"
              "UID" "USR" "VAR")
function _exit {
     rc=$1
     echo "Command complete."
        return $rc}

function usage {
     echo "usage: $(basename $0) { -check | -describe | -execute }"
     _exit 99}

function describe {
        echo "SM_PI_NAME:$name"
        echo "SM_PI_DESCRIPTION:$description"
        _exit 0}

function check {
        _exit 0}

function execute {
        IFS=\$ myEnv=$(env)
     for a in ${paramteter[@]}; do
              key=$(echo ${$a} | awk -F':' '{ print $1 }')
         val=$(echo $myEnv | grep -i -w $key 2>/dev/null | awk -F'=' '{ print $2 }')

              if [ -n "$val" ] ; then
             state="set to $val"
              else
                      state="not set"
                      #indicate a FATAL error, do not continue processing
                      ((EXIT=+4))
              fi
         echo "parameter $key is $state"
```

```
     done

    ########################################################################
    # additional checks
 # Use SnapManager environment variable of SM_TARGET_SID

    if [ -n "$SM_TARGET_SID" ] ; then
          if [ ${#SM_TARGET_SID} -ne 3 ] ; then
        echo "SID is defined as a 3 digit value, [$SM_TARGET_SID] is not valid."
        EXIT=4
    else
        echo "${INVALID_SIDS[@]}" | grep -i -w $SM_TARGET_SID >/dev/null 2>&1
                if [ $? -eq 0 ] ; then
            echo "The usage of SID [$SM_TARGET_SID] is not supported by SAP."
                        ((EXIT+=4))
          fi
    fi
    else
          echo "SM_TARGET_SID not set"
     EXIT=4
    fi  _exit $EXIT}

# Include the 3 required operations for clone plugin
case $(echo "$1" | tr [A-Z] [a-z]) in
  -check )        check     ;;
  -describe )     describe  ;;
  -execute )      execute   ;;         * )
         echo "unknown option $1"    usage         ;;
esac
```

Creating task scripts for SnapManager operation

To create the pre-task, post-task, and policy task scripts for backup, restore, and clone operations, write your script, and include the predefined environment variables in your parameters. You can start a script from scratch or modify one of the SnapManager sample scripts.

About this task

SnapManager expects your script to be structured in a particular manner to support being executed within the context of a SnapManager operation. Create the script based on the expected operations, available input parameters, and return code conventions stated in this document.

Your plug-in script should also include log messages and re-direct the messages into user-defined log files.

Steps

1. To customize a sample script, do the following:
   a. Locate a sample script in the following SnapManager install directory:

      `<default_install_directory>/plugins/examples/backup/create`

      `<default_install_directory>/plugins/examples/clone/create`

   b. Open the script in your script editor.

   c. Save it as your own script.

2. Modify the functions, variables, and parameters as needed.
3. Save your script in one of the following directory locations:

   For the backup operation:

   - <default_install_directory>/plugins/backup/create/pre: Executes the script before the backup operation occurs. Use it on an optional basis when you specify the backup creation.
   - <default_install_directory>/plugins/backup/create/post: Executes the script after the backup operation occurs. Use it on an optional basis when you specify the backup creation.
   - <default_install_directory>/plugins/backup/create/policy: Always executes the script before the backup operation occurs. SnapManager always uses this script for all the backups in the repository.

   For the restore operation:

   - <default_install_directory>/plugins/restore/create/pre: Executes the script before the restore operation occurs. Use it on an optional basis when you specify to restore the backup.
   - <default_install_directory>/plugins/restore/create/post: Executes the script after the restore operation occurs. Use it on an optional basis when you specify to restore the backup.
   - <default_install_directory>/plugins/restore/create/policy: Always executes the script before the restore operation occurs. SnapManager always uses this script for all the restore operations.

   For the clone operation:

   - <default_install_directory>/plugins/clone/create/pre: Executes the script before the clone operation occurs. Use it on an optional basis when you specify the clone creation.
   - <default_install_directory>/plugins/clone/create/post: Executes the script after the clone operation occurs. Use it on an optional basis when you specify the clone creation.
   - <default_install_directory>/plugins/clone/create/policy: Always executes the script before the clone operation occurs. SnapManager always uses this script on all clones in the repository.

Installing the task scripts

To deploy the pre-task, post-task, and policy task scripts in SnapManager, you must place them in a specified directory on the target server where the backups or clones will be created, not the host of the profile. For the restore operation, the scripts must be placed in the specified directory on the target server where you would like to restore the backup.

About this task

Store the scripts in one of the following subdirectories:

- `<default_install_directory >/plugins/backup/create`
- `<default_install_directory >/plugins/restore/create`
- `<default_install_directory >/plugins/clone/create`

where `<default_install_directory>` reflects the directory where you installed SnapManager.

Steps

1. Create your script.
2. Place your script in one of the following directories:

For backup operation

Directory	Description
/plugins/backup/create/policy	Store policy-based scripts here. These will always run before backup operations.
/plugins/backup/create/pre	Store pre-processing scripts here. These will run before backup operations run when you select them.
/plugins/backup/create/post	Store post-processing scripts here. These will run after backup operations occur when you select them.

For restore operation

Directory	Description
/plugins/restore/create/policy	Store policy-based scripts here. These will always run before restore operations.
/plugins/restore/create/pre	Store pre-processing scripts here. These will run before restore operations run when you select them.
/plugins/restore/create/post	Store post-processing scripts here. These will run after restore operations occur when you select them.

For clone operation

Directory	Description
/plugins/clone/create/policy	Store policy-based scripts here. These will always run before clone operations.
/plugins/clone/create/pre	Store pre-processing scripts here. These will run before clone operations run when you select them.
/plugins/clone/create/post	Store post-processing scripts here. These will run after clone operations occur when you select them.

Verifying installation of plug-in scripts

SnapManager provides the ability for you to install and use custom scripts to perform various operations. SnapManager offers plug-ins for the backup, restore, and clone operations, which you can use to automate your custom scripts before and after the backup, restore, and clone operations.

About this task

Before you use the plug-in for the backup, restore, and clone operations, verify the installation of plug-in scripts on the SnapManager server.

Step

1. To verify the installation of SnapManager plug-in scripts, enter this command on the SnapManager server. If you omit the `-osaccount` option, checks the plug-in scripts as root rather than for a specified user.

   `smo plugin check -osaccount` ***`os db user name`***

   Example

   The following output indicates that the policy1, the pre-plugin1, and pre-plugin2 scripts are fine. However, the post-plugin1 script is not operational.

```
smo plugin check
Checking plugin directory structure ...
<installdir>/plugins/clone/policy
   OK: 'policy1' is executable

<installdir>/plugins/clone/pre
   OK: 'pre-plugin1' is executable and returned status 0
   OK: 'pre-plugin2' is executable and returned status 0

<installdir>/plugins/clone/post
   ERROR: 'post-plugin1' is executable and returned status 3
Command complete.
```

Creating task specification for SnapManager operations

You can create the task specification file from the SnapManager graphical user interface, command line interface, or a text editor.

Before you begin

The pre-task, post-task, and the policy task scripts must be placed in the correct installation directory.

About this task

To create the task specification XML file, perform the following steps:

Steps

1. Create a task specification file using following file structure:

```
<task-specification>
  <pre-tasks>
                <task>
      <name>name</name>
      <parameter>
          <name>name</name>
          <value>value</value>
       </parameter>
    </task>
  </pre-tasks>
  <post-tasks>
    <task>
      <name>name</name>
      <parameter>
          <name>name</name>
          <value>value</value>
       </parameter>
    </task>
  </post-tasks>
</task-specification>
```

2. Add script name in the `name` option.
3. If you have added any new parameter in the script, you must add a new parameter name `<parameter> <name>` option in the XML file and provide values.
4. Save the XML file in the correct installation directory.
5. Use the task specification file for the pre-processing activity or the post-processing activity of the backup, restore, or the clone operations.

   Note: You must provide the absolute path of the task specification XML file in the script.

Task specification example

```
  <task-specification>
    <pre-tasks>
      <task>
        <name>clone cleanup</name>
        <description>pre tasks for cleaning up the target system</
description>
        </task>
    </pre-tasks>
    <post-tasks>
      <task>
        <name>SystemCopy follow-up activities</name>
        <description>SystemCopy follow-up activities</description>
        <parameter>
          <name>SCHEMAOWNER</name>
          <value>SAMSR3</value>
        </parameter>
```

```
      </task>
      <task>
        <name>Oracle Users for OS based DB authentication</name>
        <description>Oracle Users for OS based DB authentication</
description>
        <parameter>
          <name>SCHEMAOWNER</name>
          <value>SAMSR3</value>
        </parameter>
        <parameter>
          <name>ORADBUSR_FILE</name>
          <value>/mnt/sam/oradbusr.sql</value>
        </parameter>
      </task>
      </post-tasks>
  </task-specification>
```

Performing backup, restore, and clone operations using pre-script and post-scripts

You can use your script while initiating a backup, restore, or clone operation. SnapManager displays a **Task-enabling** page in the **Backup Create** wizard, **Restore or Recover** wizard, or **Clone Create** wizard, where you can select the script and provide values for any parameters required by the script.

Before you begin

You must install the plug-in scripts in the correct SnapManager installation location. Verify that the plug-ins are installed correctly using the command.

Ensure that you are using the BASH shell.

About this task

In the command-line interface, list the script name, select the parameters, and set the values.

Steps

1. To verify that you are using the BASH shell, enter the following command at the command prompt: `bash`

   The BASH shell is operating properly, if you do not see an error.

   Alternately, you can enter the `which-bash` command at the prompt, and use the command output as the start parameter of the script.

2. You can create a backup, restore a backup, and create a clone using the scripts:

- For the backup operation, enter the `-taskspec` option and provide the absolute path of the task specification XML file for performing a pre-processing or a post-processing activity to occur before or after the backup operation:

  smo backup create -profile *profile_name* {[-full {-online | -offline | -auto} [-retain {-hourly | [-daily | -weekly | -monthly | -unlimited}] [-verify] | [-data [[-files *files* [*files*]] | [-tablespaces -*tablespaces* [-*tablespaces*]] [-datalabel *label*] {-online | -offline | -auto} [-retain {-hourly | [-daily | -weekly | -monthly | -unlimited]} [-verify] | [-archivelogs [-label *label*] [-comment *comment*] [-protect | -noprotect | -protectnow] [-backup-dest *path1* [,[*path2*]]] [-exclude-dest *path1* [,*path2*]]] [-prunelogs {-all | -untilSCN *untilSCN* | -before {-date *yyyy-MM-dd HH:mm:ss* | -months | -days | -weeks | -hours}} -prune-dest *prune_dest1*,[*prune_dest2*]] [-taskspec *taskspec*] [-include-with-online-backups | -no-include-with-online-backups]} -dump [-force] [-quiet | -verbose]

- For the backup restore operation, enter the `-taskspec` option and provide the absolute path of the task specification XML file for performing a pre-processing or a post-processing activity to occur before or after the restore operation:

  smo backup restore -profile *profile_name* {-label <*label*> | -id <*id*>} {-files<*files*> |-tablespaces <*tablespaces*> | -complete | -controlfiles} [-recover {*-alllogs | -nologs | -until* <*until*>}][-restorespec <*restorespec*>] | -from-secondary [*-temp-volume* <*temp_volume*>] [-copy-id *id*]][-taskspec <*taskspec*>] [-verify][-force] backup restore -fast [require | override | fallback | off] [-preview] -dump [-quiet | -verbose]

- For the clone create operation, enter the `-taskspec` option and provide the absolute path of the task specification XML file for performing a pre-processing or a post-processing activity to occur before or after the clone operation:

  smo clone create -profile *profile_name* {-backup-label *backup_name* | -backup-id <*backup-id*>| -current} -newsid *new_sid* -clonespec *full_path_to_clonespecfile* [-reserve <*yes, no, inherit*>] [-host <*host*>] [-label <*label*>] [-comment <*comment*>] [-from-secondary [-copy-id <*id*>]] {-taskspec <*taskspec*>] -dump [-quiet | -verbose]

If the backup, restore, or clone plug-in operation failed, only the plug-in name and return code display. Your plug-in script should include log messages and re-direct the messages into user-defined log files.

Example of creating a backup using the task specification XML file

The following example creates a backups with a task specification XML file `sales1_taskspec.xml` using a profile called SALES1.

```
smo backup create -profile SALES1 -full -online -taskspec sales1_taskspec.xml -force -
verify
```

Updating storage system name and target database host name associated with a profile

SnapManager3.2 for Oracle provides the ability to update the storage system host name or storage system address, and the target database host name associated with a SnapManager profile.

Updating storage system name associated with a profile

SnapManager3.2 for Oracle provide the ability to update the host name or IP address of a storage system associated with a SnapManager profile.

Before you begin

Ensure that the profile has at least one backup, and then update the storage system name. If the profile does not have any backup, then there is no necessity to update the storage system name for that profile.

About this task

SnapManager for Oracle enables you to update the storage system name or IP address associated with the SnapManager profile from the SnapManager CLI. While updating the storage system name, the metadata stored in the repository database alone is updated. Ensure that the Snapshot copies are available in the new storage system.

SnapManager for Oracle does not verify the existence of the Snapshot copies in the storage system.

After renaming the storage system name, you can perform all the SnapManager operations as earlier.

There are few things you must keep in mind while performing rolling upgrade and roll back of the host after renaming the storage system name:

- If you are performing rolling upgrade of the host after renaming the storage system name, you must update the profile for the new storage system name.
  Refer to "Troubleshooting storage system name issues" topic on how to use the SnapDrive commands for changing the storage system name.
- If you are rolling back the host after renaming the storage system, ensure that you change the storage system name back to the earlier storage system name so that you can use those profiles, backups, and clones for performing SnapManager operation.

Scenarios not supported for changing the storage system name in the SnapManager profile

- Updating the storage system name from the SnapManager GUI is not supported.
- SnapManager operations after roll back of the host are not supported.

- When there is any database operation running on the profile, SnapManager does not support changing the storage system host name or IP address for that profile.

Note: If SnapDrive could not identify the storage system and displays error messages, you can enter the `ipmigrate` command with earlier and later host names of the storage system from SnapDrive. For additional information on storage system name issues, refer to "Troubleshooting storage system name issues".

Step

1. To update the host name or IP address of the storage system, enter the following command:

   `smo storage rename -profile` *`profile`* `-oldname` *`old_storage_name`* `-newname` *`new_storage_name`* `[quiet | -verbose]`

If...	Then...
You want to update the storage system name associated with a profile	Specify the `-profile` option.
You want to update the storage system name or IP address	Specify these options and variables: • `-oldname` *`old_storage_name`* is the host name or IP address of the storage system. • `-newname` *`new_storage_name`* is the host name or IP address of the storage system.

```
smo storage rename -profile mjullian -oldname lech -newname hudson -verbose
```

Related references

Troubleshooting storage system name issues on page 376

Viewing a list of storage systems associated with a profile

You can view a list of the storage systems associated with a particular profile.

About this task

The list displays the storage system names associated with the particular profile.

Note: If there are no backups available for the profile, then you cannot view the storage system name associated with the profile.

Step

1. To display information about storage systems associated with a particular profile, enter this command:

   smo storage list -profile *profile* [-quiet | -verbose]

Example

```
smo storage list -profile mjubllian

Sample Output:
Storage Controllers
--------------------
STCO1110-RTP07OLD
```

Updating target database host name associated with a profile

SnapManager 3.2 for Oracle provides the ability to update the hostname of the target database in the SnapManager profile.

Before you begin

- Before updating the target database host name in the profile, enter the smo `profile sync` command to make the local user's home directory aware of the profile-to-repository mappings.
- Before updating the target database hostname from the SnapManager CLI, if there are one or more SnapManager GUI session opened, then close all the SnapManager GUI sessions.
- Before updating the target database host name on a RAC environment, if there are any clones or mounted backups available on the host referred in the profile, delete all the clones and unmount the backups from the host.

  Otherwise, after updating the target database hostname, executing these SnapManager operations from the new host lead to failure as well as stale entries in the earlier host.

About this task

You can update the profile with the new host name from the SnapManager CLI.

After updating the target database host name in the profile, only the target database host name is changed and all the other configuration parameters set on the profile are retained.

After updating the new target database host name in a protection-enabled profile, the same dataset and protection policy are retained for the updated profile.

After changing the host name for the target host, ensure that you update the host name for all the existing protected profiles before creating the new protected profiles. To update the hostname for a profile, use the `smo profile update` command.

After updating the target database hostname, user cannot delete or split the clone, or unmount the backup if the clone or mounted backup is not available in the new host. In such scenario, executing these SnapManager operations from the new host lead to failure as well as stale entries in the earlier host. To perform the above specified SnapManager operation, user must revert back to the earlier hostname through profile update, and perform the SnapManager operation.

Scenarios not supported for changing the target database host name in SnapManager profile

The SnapManager operations that are not supported for changing the target database hostname in the profile are:

- Changing the target database host name from the SnapManager GUI.
- Rolling back of the repository database is not supported after updating the target database hostname of the profile.
- Updating multiple profiles for a new target database hostname in one shot.
- When any SnapManager operation is running, SnapManager does not allow you to change the target database host name.

Steps

1. To update the target database host name of a profile, enter this command:

   smo profile update -profile *profile* [-profile-password *profile_password*] [-database -dbname*db_dbname* -host *db_host* [-sid *db_sid*] [-login -username*db_username* -password *db_password*-port *db_port*]] [{-rman{-controlfile | {-login -username *rman_username* -password *rman_password* -tnsname *rman_tnsname*}}} | -remove-rman]-osaccount *osaccount* -osgroup *osgroup* [-retain [-hourly [-count *n*] [-duration *m*]] [-daily [-count *n*] [-duration *m*]] [-weekly [-count *n*] [-duration *m*]] [-monthly [-count *n*] [-duration *m*]]] [-comment *comment*][-snapname-pattern *pattern*][[-protect [-protection-policy *policy_name*]]| [[-noprotect]] [-summary-notification] [-notification [-success -email *email_address1, email_address2* -subject *subject_pattern*] [-failure -email *email_address1, email_address2* -subject *subject_pattern*]] [-separate-archivelog-backups -retain-archivelog-backups -hours *hours* | -days *days* | -weeks *weeks*| -months *months* [-protect [-protection-policy *policy_name*] | -noprotect] [-include-with-online-backups | -no-include-with-online-backups]] [-dump]

   Other options for this command are as follows:

   `[-force] [-noprompt]`

   `[quiet | -verbose]`

If...	Then...
You want to change the target database host name	Specify `-host new_db_host` to change the target database host name of the profile.

2. To update the target database host name of the profile, enter this command: `smo profile show`.

Maintaining history of SnapManager operations

SnapManager for Oracle enables you to maintain the history of SnapManager operations associated with a single profile or multiple profiles, from the SnapManager CLI or GUI. You can view the history of the operations as a report, and use the report for audit compliance purposes.

SnapManager enables you to maintain the history for the following SnapManager operations:

- Backup create
- Backup verify
- Backup restore
- Clone create
- Clone split

The history information for the SnapManager operations is maintained based on the retention. You can configure different retention class for each of the supported SnapManager operations.

The retention classes that you can assign are:

- Number of days
- Number of weeks
- Number of months
- Number of operations

Based on the retention, SnapManager purges the history automatically. You can also manually purge the history of the SnapManager operation.

The history is maintained for SnapManager-initiated (scheduled backup) and user-initiated operations. If you delete or destroy the profile, all the history information associated with the profile is deleted.

Note: After rollback of the host, you cannot view the history details or perform any history-related operations associated with the profile that has been configured for history maintenance.

Configuring history for SnapManager operation

SnapManager for Oracle enables you to maintain the history of SnapManager operation from the SnapManager CLI or GUI. You can view the history of the SnapManager operation as a report.

Step

1. To configure the history of SnapManager operation, enter the following command:

```
smo history set -profile {-name, profile_name [profile_name1,
profile_name2] | -all -repository -login [-password repo_password] -
username repo_username -dbname repo_dbname -host repo_host -port
```

***repo_port*} -operation {-operations *operation_name* [*operation_name1, operation_name2*] | -all} -retain {-count *retain_count* | -daily *retain_daily* | -weekly *retain_weekly* | -monthly *retain_monthly*} [-quiet | -verbose]**

```
smo
history set -profile -name PROFILE1 -operation -operations backup -
retain -daily 6 -verbose
```

```
smo
history set -profile -name PROFILE1 -operation -all -retain -weekly 3
-verbose
```

Viewing a list of SnapManager operation history

You can view the history of a specific or all SnapManager operations as a report based on the retention settings.

Step

1. To view a list of SnapManager history operations, enter the following command:

   smo history list -profile {*-name, profile_name* [*profile_name1,profile_name2*] | *-all* -repository -login [-password *repo_password*] -username *repo_username* -dbname *repo_dbname* -host *repo_host* -port *repo_port*} -operation {-operations *operation_name* [*operation_name1, operation_name2*] | -all} [-delimiter *delimiter*] [-quiet | -verbose]

Viewing history details of specific operation associated with a profile

You can view the detailed history of a specific SnapManager operation associated with a profile.

Step

1. To display detailed history information about a specific SnapManager operation associated with a profile, enter the following command:

   smo history operation-show -profile *profile_name* {-label *label* | -id *id*} [-quiet | -verbose]

Deleting history of SnapManager operation

You can delete the history of the SnapManager operation, if you no longer require the history details.

Step

1. To delete the history of the SnapManager operation, enter the following command:

```
smo history purge -profile {-name, profile_name profile_name1,
profile_name2] | all -repository -login [-password repo_password] -
username repo_username -dbname repo_dbname -host repo_host -port
repo_port} -operation {-operations operation_name [operation_name1,
operation_name2] | -all} [-quiet | -verbose]
```

Removing history settings associated with a single profile or multiple profiles

SnapManager enables you to remove the history settings of a SnapManager operation. This operation purges all the history information associated with a single profile or multiple profiles.

Step

1. To remove the history of SnapManager operations associated with a single profile or multiple profiles, enter the following command:

```
smo history remove -profile {-name, profile_name [profile_name1,
profile_name2] | all -repository -login [-password repo_password] -
username repo_username -dbname repo_dbname -host repo_host -port
repo_port} -operation {-operations operation_name [operation_name1,
operation_name2] | -all} [-quiet | -verbose]
```

Viewing SnapManager history configuration details

You can view the history settings for a single profile.

About this task

The SnapManager history operation displays the following information for each profile:

- Operation name
- Retention class
- Retention count

Step

1. To display information about the SnapManager history operation for a specific profile, enter the following command:

   ```
   smo history show -profile profile_name
   ```

SnapManager for Oracle command reference

The SnapManager command reference includes the valid usage syntax, options, parameters, and arguments you should supply with the commands, along with examples.

The following issues apply to command usage:

- Commands are case-sensitive.
- SnapManager accepts up to 200 characters and labels up to 80 characters.
- If the shell on your host limits the number of characters that can appear on a command line, you can use the `cmdfile` command.
- Do not use spaces in profile names or label names.
- In the clone specification, do not use spaces in the clone location.

SnapManager can display three levels of messages to the console:

- Error messages
- Warning messages
- Informational messages

You can specify how you want messages displayed. If you specify nothing, SnapManager displays only error messages and warnings to the console. To control the amount of output that SnapManager displays on the console, use one of the following command line options:

- `-quiet`: Displays only error messages to the console.
- `-verbose`: Displays error, warning, and informational messages to the console.

Note: Regardless of the default behavior, or the level of detail you specify for the display, SnapManager always writes all message types to the log files.

The smo_server restart command

This command restarts the SnapManager host server and is entered as root.

Syntax

```
smo_server restart
[-quiet | -verbose]
```

Parameters

`-quiet`

Specifies that only error messages are displayed on the console. The default is to display error and warning messages.

-verbose

Specifies that error, warning, and informational messages are displayed on the console.

Example command

The following example restarts the host server:

```
smo_server restart
```

The smo_server start command

This command starts the host server running the SnapManager for Oracle software.

Syntax

```
smo_server start
[-quiet | -verbose]
```

Parameters

-quiet

Specifies that only error messages are displayed on the console. The default is to display error and warning messages.

-verbose

Specifies that error, warning, and informational messages are displayed on the console.

Example command

The following example starts the host server:

```
smo_server start
SMO-17100: SnapManager Server started on secure port 25204 with PID 11250
```

The smo_server status command

This command verifies the status of the SnapManager host server and is entered at the root.

Syntax

```
smo_server status
[-quiet | -verbose]
```

Parameters

-quiet

Specifies that only error messages are displayed on the console. The default is to display error and warning messages.

-verbose

Specifies that error, warning, and informational messages are displayed on the console.

Example command

The following example requests the status of the host server:

```
smo_server status
SMO-17104: SnapManager Server version 3.2 is running on secure port 25204 with PID 11250
  and has 0 operations in progress.
```

The smo_server stop command

This command stops the SnapManager host server and is entered at the root.

Syntax

```
smo_server stop
[-quiet | -verbose]
```

Parameters

-quiet

Specifies that only error messages are displayed on the console. The default is to display error and warning messages.

-verbose

Specifies that error, warning, and informational messages are displayed on the console.

Example command

The following example uses the `smo_server stop` command:

```
smo_server stop
```

The smo backup create command

This command creates backups of databases on one or more storage system volumes.

Syntax

```
smo backup create-profile profile_name
{[-full{-auto | -online | -offline}[-retain {-hourly | -daily | -weekly
| -monthly | -unlimited} [-verify]  |
[-data [[-files files [files]] |
[-tablespaces tablespaces [tablespaces]] [-label label] {-auto | -
online | -offline} [-retain {-hourly | -daily | -weekly | -monthly |
-unlimited} [-verify] |
[-archivelogs [-label label]] [-comment comment]}
[-protect | -noprotect | -protectnow][-backup-dest path1 [ , path2]]
[-exclude-dest path1 [ , path2]] [-prunelogs {-all | -until-scn until-
scn | -until-date yyyy-MM-dd:HH:mm:ss] | -before {-months | -days | -
weeks | -hours}}
-prune-dest prune_dest1,[prune_dest2]][-taskspec taskspec][-dump]-
force
[-quiet | -verbose]
```

Before you run this command, you must create a database profile using the `profile create` command.

Parameters

`-profile` *`profile_name`*

Specifies the name of the profile related to the database you want to back up. The profile contains the identifier of the database and other database information.

`-auto`

If the database is in a mounted or offline state, SnapManager performs an offline backup. If the database is in an open or online state, SnapManager performs an online backup. If you use the `-force` option with the `-offline` option, SnapManager forces an offline backup even if the database is currently online.

`-online`

Specifies an online database backup.

You can take an online backup of a RAC database, as long as the primary is OPEN, or the primary is MOUNTED and an instance is OPEN. Use `-force` for online backups if the local instance is SHUTDOWN, or no instance is OPEN. The version of Oracle must be 10.2.0.3 or later or the database will hang if any instance in the RAC is mounted.

- If the local instance is SHUTDOWN and at least one instance is OPEN, using -force changes the local instance to MOUNTED.
- If no instance is OPEN, using `-force` changes the local instance to OPEN.

`-offline`

Specifies an offline backup while the database is shut down. If the database is in either the OPEN or MOUNTED state, the backup fails. If the `-force` option is used, SnapManager attempts to alter the database state to shut down the database for an offline backup.

`-full`

Backs up the entire database. This includes all the data, archived log and control files. The archived redo logs and control files are backed up no matter what type of backup you perform. If you want to back up only a portion of the database, use the `-files` or the `-tablespaces` option.

`-files` ***`list`***

Backs up only the specified data files plus the archived log and control files. Separate the list of file names with spaces. If the database is OPEN, SnapManager ensures that the appropriate tablespaces are in online backup mode.

`-tablespaces` ***`tablespaces`***

Backs up only the specified database tablespaces plus the archived log and control files. Separate the tablespace names with spaces. If the database is OPEN, SnapManager ensures that the appropriate tablespaces are in online backup mode.

`-label` ***`label`***

Specifies an optional name for this backup. This name must be unique within the profile. The name can contain letters, numbers, underscore (_), and hyphen (-). It cannot start with a hyphen.

If you do not specify a label, SnapManager creates a default label that has the format scope_type_date where:

- scope is either F to indicate a full backup or P to indicate a partial backup.
- type is C to indicate an offline (cold) backup, H to indicate an online (hot) backup, or A to indicate auto backup, for example, P_A_20081010060037IST.
- date is the year, month, day, and time of the backup. SnapManager uses a 24-hour clock.

For example, if you performed a full backup with the database offline on Jan. 16, 2007, at 5:45:16 p.m. Eastern standard time, SnapManager would create the label F_C_20070116174516EST.

`-comment` ***`string`***

Specifies an optional comment to describe this backup. Enclose the string in single quotes (').

Note: Some shells strip off quote marks. If that is true for your shell, you must escape the quote with a backslash (\). For example, you might need to enter: \' this is a comment\'.

-verify

Verifies that the files in the backup are not corrupt by running the Oracle dbv utility.

Note: If you specify the `-verify` option, the backup operation does not complete until the verify operation completes.

-force

Forces a state change if the database is not in the correct state. For example, SnapManager might change the state of the database from online to offline, based on the type of backup you specify and the state that the database is in.

With an online RAC database backup, use `-force` if the local instance is SHUTDOWN, or no instance is OPEN.

Note: The version of Oracle must be 10.2.0.3 or later or the database will hang if any instance in the RAC is mounted.

- If the local instance is SHUTDOWN and at least one instance is OPEN, then using -force changes the local instance to MOUNTED.
- If no instance is OPEN, using -force changes the local instance to OPEN.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

-protect | **-noprotect** | **-protectnow**

Indicates whether the backup should be protected to secondary storage. The `-noprotect` option specifies that the backup should not be protected to secondary storage. Only full backups are protected. If neither option is specified, SnapManager protects the backup as the default, if the backup is a full backup and the profile specifies a protection policy. The `-protectnow` specifies to protect the backup immediately to secondary storage.

-retain { -hourly | -daily | -weekly | -monthly | -unlimited}

Specifies whether the backup should be retained on an hourly, daily, weekly, monthly, or unlimited basis. If `-retain` is not specified, the retention class defaults to `-hourly`. To retain backups forever, use the `-unlimited` option. The

`-unlimited` option makes the backup ineligible for deletion by the retention policy.

-archivelogs

Creates archive log backup.

-backup-dest *path1*, [*, [path2]*]

Specifies the archive log destinations to be backed up for archive log backup.

-exclude-dest *path1*, [*, [path2]*]

Specifies the archive log destinations to be excluded from the backup.

-prunelogs {-all | -until-scn *until-scn* | -until-date *yyyy-MM-dd:HH:mm:ss* | -before {-months | -days | -weeks | -hours}

Deletes the archive log files from the archive log destinations based on options provided while creating a backup. The `-all` option deletes all the archive log files from the archive log destinations. The `-until-scn` option deletes the archive log files until a specified SCN. The `-until-date` option deletes the archive log files until the specified time period. The `-before` option deletes the archive log files before the specified time period (days, months, weeks, hours).

-prune-dest *prune_dest1,prune_dest2*

Deletes the archive log files from the archive log destinations while creating the backup.

-taskspec *taskspec*

Specifies the task specification XML file that can be used for pre-processing activity or post-processing activity of the backup operation. The complete path of the XML file should be provided which give the `-taskspec` option.

-dump

Collects the dump files after the successful or failed database backup operation.

Example command

The following example creates a full online backup, creates a backup to secondary storage, and sets the retention policy to daily:

```
smo backup create -profile SALES1 -full -online
-label full_backup_sales_May -profile SALESDB -force -protect -retain -daily
Operation Id [8abc01ec0e79356d010e793581f70001] succeeded.
```

Related tasks

Creating database backups on page 135

Related references

The smo profile create command on page 324

The smo backup delete command

This command enables you to remove backups that SnapManager does not automatically remove, such as backups that were used to create a clone or backups that failed.

Syntax

```
smo backup delete
-profile profile_name
[-label label [-data | -archivelogs] | [-id guid | -all]
-force
[-dump] [-quiet | -verbose]
```

Parameters

You can delete backups retained on an unlimited basis without changing the retention class.

-profile *profile_name*

Specifies the database associated with the backup you want to remove. The profile contains the identifier of the database and other database information.

-id *guid*

Deletes the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. Use the `smo backup list` command to display the GUID for each backup.

-label *name*

Deletes the backup with the specified label. Optionally specify the scope of the backup as datafile or archivelog.

-all

Deletes all backups. To delete only specified backups instead, use the `-id` or `-label` option.

-dump

Collects the dump files after a successful or failed backup delete operation.

-force

Forces the removal of the backup. SnapManager removes the backup even if there are problems freeing the resources associated with the backup. For example, if the backup was cataloged with RMAN, but the RMAN database no longer exists,

including `-force` on the command line lets SnapManager delete the backup even though it cannot connect with RMAN.

`-quiet`

Displays only error messages on the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages on the console.

Example command

The following example deletes the backup:

```
smo backup delete -profile SALES1 -label full_backup_sales_May
Operation Id [8abc01ec0e79004b010e79006da60001] succeeded.
```

Related tasks

Deleting backups on page 155

Related references

The smo profile create command on page 324
The smo profile update command on page 335

The smo backup free command

This command enables you to delete the Snapshot copies of the backups without removing the backup metadata from the repository.

Syntax

```
smo backup free
-profile profile_name
[-label label [data | -archivelogs] | [-id guid | -all]
-force
[-dump] [-quiet | -verbose]
```

Parameters

`-profile` *`profile_name`*

Specifies the database associated with the backup you want to free. The profile contains the identifier of the database and other database information.

`-id` *`guid`*

Deletes the resources of the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. Use the `smo backup list`

command to display the GUID for each backup. Include the `-verbose` option to display the backup IDs.

`-label` *`label_name`*

Frees the backup resources with the specified label.

`-all`

Frees all backups. To delete specified backups instead, use the `-id` or `-label` option.

`-force`

Forces the removal of the Snapshot copies.

`-quiet`

Displays only error messages on the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages on the console.

Example command

The following example frees the backup resources:

```
smo backup free -profile SALES1 -label full_backup_sales_May
Operation Id [8abc01ec0e79004b010e79006da60001] succeeded.
```

Related tasks

Freeing backups on page 153

The smo backup list command

This command displays information about all of the backups in a profile, including information about the retention class and protection status.

Syntax

```
smo backup list
-profile profile_name
-delimiter character
[-data | -archivelogs | -all]
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the database you want to list backups for. The profile contains the identifier of the database and other database information.

-delimiter *character*

When this parameter is specified, the command lists each row on a separate line and the attributes in that row are separated by the character specified.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console. Include the `-verbose` option to display the backup IDs.

Example command

The following example lists the backups for the SALES1 profile:

```
smo backup list -profile SALES1 -verbose
Start Date          Status  Scope  Mode    Primary  Label      Retention   Protection
------------------- ------- ------ ------- -------- -----      --------    ----------
2007-08-10 14:31:27 SUCCESS FULL   ONLINE  EXISTS   backup1    DAILY       PROTECTED
2007-08-10 14:12:31 SUCCESS FULL   ONLINE  EXISTS   backup2    HOURLY      NOT PROTECTED
2007-08-10 10:52:06 SUCCESS FULL   ONLINE  EXISTS   backup3    HOURLY      PROTECTED
2007-08-05 12:08:37 SUCCESS FULL   ONLINE  EXISTS   backup4    UNLIMITED   NOT PROTECTED
2007-08-05 09:22:08 SUCCESS FULL   OFFLINE EXISTS   backup5    HOURLY      PROTECTED
2007-08-04 22:03:09 SUCCESS FULL   ONLINE  EXISTS   backup6    UNLIMITED   NOT REQUESTED
2007-07-30 18:31:05 SUCCESS FULL   OFFLINE EXISTS   backup7    HOURLY      PROTECTED
```

Related tasks

Viewing a list of backups on page 150

The smo backup mount command

This command mounts a backup in order to perform a recover operation using an external tool.

Syntax

```
smo backup mount
-profile profile_name
[-label label [-data | -archivelogs] | [-id id]
[-host  host]
[-from-secondary {-copy-id id}]
[-dump]
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the database you want to mount a backup of. The profile contains the identifier of the database and other database information.

-id *guid*

Mounts the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. Use the `smo backup list` command to display the GUID for each backup.

-label *label*

Mounts the backup with the specified label.

-from-secondary -copy-id *id*

Mounts the backup from secondary storage. If this option is not specified, SnapManager mounts the backup from primary storage. You can use this option if the backup is freed. Use the `copy-id` option to differentiate the backups between the secondary and tertiary storage systems.

If there is more than one copy on the secondary or tertiary storage systems, use the `-copy-id` option to specify which copy on the secondary or tertiary storage systems should be used to mount the backup.

-host *host*

Specifies the host on which you want to mount the backup.

-dump

Collects the dump files after the successful or failed mount operation.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Note: You should use this command only if you are using an external tool such as RMAN. SnapManager automatically handles mounting backups if you use the `smo backup restore` command to restore the backup. This command displays a list of INFO logs which shows the paths where the Snapshot copies in the backup have been mounted. These INFO logs are displayed only when `-verbose` is specified.

Example command

The following example mounts the backup:

```
smo backup mount -profile SALES1 -label full_backup_sales_May -verbose
SMO-13046 [INFO ]: Operation GUID 8abc013111b9088e0111b908a7560001 starting on Profile
SALES1
SMO-08052 [INFO ]: Beginning to connect mount(s) [/mnt/ssys1/logs, /mnt/ssys1/data] from
logical snapshot SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001.
SMO-08025 [INFO ]: Beginning to connect mount /mnt/ssys1/logs from snapshot
SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001_0 of volume hs_logs.
SMO-08027 [INFO ]: Finished connecting mount /mnt/ssys1/logs from snapshot
SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001_0 of volume hs_logs.
SMO-08025 [INFO ]: Beginning to connect mount /mnt/ssys1/data from snapshot
SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001_0 of volume hs_data.
SMO-08027 [INFO ]: Finished connecting mount /mnt/ssys1/data from snapshot
SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001_0 of volume hs_data.
SMO-08053 [INFO ]: Finished connecting mount(s) [/mnt/ssys1/logs, /mnt/ssys1/data] from
logical snapshot SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001.
SMO-13037 [INFO ]: Successfully completed operation: Backup Mount
SMO-13048 [INFO ]: Operation Status: SUCCESS
SMO-13049 [INFO ]: Elapsed Time: 0:01:00.981
Operation Id [8abc013111b9088e0111b908a7560001] succeeded.
```

Related tasks

Mounting backups on page 152

The smo backup restore command

This command restores backups of a database or a portion of a database and then optionally recovers the database information.

Syntax

```
smo backup restore
-profile profile_name
[-label label | -id id]
[-files files [files...] |
-tablespaces tablespaces [tablespaces...]] |
-complete | -controlfiles]
[-recover {-alllogs | -nologs | -until until} [-using-backup-
controlfile] ]
[-restorespec restorespec | -from-secondary [-temp-volume temp_volume]
[-copy-id id]]
[-preview]
[-fast {-require | -override | -fallback | -off}]
[-recover-from-location path1 [, path2]][-taskspec taskspec][-dump][-
force]
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the database you want to restore. The profile contains the identifier of the database and other database information.

-label *name*

Restores the backup with the specified label.

-id *guid*

Restores the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. Use the `smo backup list` command to display the GUID for each backup.

Choose all or specified files

Optionally, use one of the following:

- `-complete`: Restores all the data files in the backup.
- `-tablespaces` *`list`*: Restores only the specified tablespaces from the backup. Use spaces to separate the names in the list.
- `-files` *`list`*: Restores only the specified data files from the backup. Use spaces to separate the names in the list. If the database is up, SnapManager ensures that the tablespace containing the files is offline.

-controlfiles

Restores the control files. SnapManager allows you to restore control files along with the data files from the backups in a single user operation. The `-controlfiles` option is independent of other restore scope parameters such as `-complete`, `-tablespaces` and `-files`.

-recover

Recovers the database after restoring it. You must also specify the point to which you want SnapManager to recover the database using one of the following options:

- `-nologs`: Recovers the database to the time of the backup and applies no logs. You can use this parameter for online or offline backups.
- `-alllogs`: Recovers the database to the last transaction or commit, and applies all required logs.
- `-until date`: Recovers the database up to the date and time specified. Use the format year-month-date: hour: minute: second `(yyyy-mm-dd:HH:MM:SS)`. This is the database time, not necessarily the time. For hours, use either 12- or 24-hour format, depending on the database setting.
- `-until scn`: Rolls forward the data files until it reaches the specified SCN number.
- `-using-backup-controlfile`: Recovers using the backup control file.

-restorespec

Enables you to manually restore the data to an active file system and restore from the specified data by providing a mapping of each original Snapshot copy to its active file system. You can specify one of the following options. If you don't

specify an option, SnapManager restores the data from the Snapshot copies on primary storage:

- `-restorespec`: Specifies the data to restore and the restore format.
- `-from-secondary`: Restores the data from secondary storage. You cannot use this option if the backup exists on primary storage; the primary backup must be freed before a backup can be restored from secondary storage. If there is more than one backup copy, you can specify which backup copy to use with the `-copy-id` option. If you use a temporary volume, specify the volume using the `-temp-volume` option.

When restoring from secondary, SnapManager first attempts to restore data directly from the secondary storage system to the primary storage system (without involving the host). If SnapManager cannot perform this type of restore (for example, if files are not part of the restore exist in a file system), then SnapManager will fall back to a host side file copy restore. SnapManager has two methods of performing a host side file copy restore from secondary. The method SnapManager selects is configured in the `smo.config` file.

- Direct: SnapManager clones the data on secondary storage, mounts the cloned data from the secondary storage system to the host, and then copies data out of the clone into the active environment. This is the default secondary access policy.
- Indirect: SnapManager first copies the data to a temporary volume on primary storage, then mounts the data from the temporary volume to the host, and then copies data out of the temporary volume into the active environment. This secondary access policy should be used only if the host does not have direct access to the secondary storage system. Restores using this method will take twice as long as the "direct" secondary access policy because two copies of the data are made.

The decision whether to use direct or indirect is controlled by the value of the `restore.secondaryAccessPolicy` parameter in the smo.config configuration file. The default is "direct."

`-preview`

Shows the following information:

- Which restore mechanism (fast restore, storage-side file system restore, storage-side file restore, or host-side file copy restore) will be used to restore each file
- Why more efficient mechanisms were not used to restore each file, when you specify the `-verbose` option

Using the `-preview` option, the following conditions apply:

- The `-force` option has no impact on the command.
- The `-recover` option has no impact on the command.

- The `-fast` option (`-require`, `-override`, `-fallback`, or `-off`) has significant impact on the output.

If you want to preview a restore of data files, and the database currently is not mounted, then SnapManager mounts the database. If the database cannot be mounted, then the command will fail. SnapManager returns the database to its original state before the command completes.

The `-preview` option shows up to 20 files. You can configure the maximum number of files using the `smo.config` file.

To preview the restore operation, the database must be able to be mounted.

`-fast`

Enables the administrator to choose the process to use in the restore operation. Database administrators can force SnapManager to use the volume-based fast restore process rather than other restore processes, if all mandatory restore eligibility conditions are met. Administrators might want to do this if they know for certain that the fast restore process is the process that they want to use. When administrators know that a volume restore cannot be performed on their backup, they can also use this process to prevent SnapManager from conducting its eligibility checks and prevent the restore from using the fast restore process.

The `-fast` option includes the following parameters:

- `-require`: Enables the database administrator to force SnapManager to perform a volume restore, if all restore eligibility conditions are met. If you specify the `-fast` option, but do not specify any parameter for `-fast`, SnapManager uses the `-require` parameter as a default.
- `-override`: Enables the database administrator (who knows that all overridable conditions can be overridden without negative consequences) to override non-mandatory eligibility checks and perform the volume-based fast restore.
- `-fallback`: Enables the database administrator to restore the database using any method that SnapManager determines and allows the administrator to provide as little information as necessary on the command line. If you do not specify `-fast`, SnapManager uses the default `backup restore -fast fallback`.
- `-off`: Enables the database administrator who knows that a volume restore cannot be performed on the database backup and wants to avoid the time required to perform all the eligibility checks. This option prevents the restore from using the fast restore process.

  Note: Data protection and volume restore cannot be performed on Windows. However, the restore command parameter which allows you to preview a restore operation is still available from the CLI on Windows. The restore preview is not attached to the volume-based fast restore.

-recover-from-location

Specifies the external archive log location of archive log files where SnapManager takes the archive log files from the external location and use them for the recovery process.

-taskspec

Specifies the task specification XML file for pre-processing activity or post-processing activity of the restore operation. Make sure you provide the complete path of the task specification XML.

-dump

Specifies to collect the dump files after the restore operation.

-force

Changes the database state to a lower state than its current state if necessary. For RAC, include the -force option if SnapManager needs to change the state of any RAC instance to a lower state.

By default, SnapManager can change the database state to a higher state during an operation. You do not need to enter this option for SnapManager to change the database to a higher state.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console. Use this option to see why more efficient restore processes could not be used to restore the file.

Example command: Preview the restore process

The following shows an example of using the `-preview` option. SnapManager lists all files ready to be restored and lists the different methods that will be used for each.

```
smo backup restore -profile SALES1 -label full_backup_sales_May
-complete -controlfiles -preview
```

```
The following files will be restored via storage side file system
restore:
+DG1/sid/datafile06.dbf
+DG1/sid/datafile07.dbf

The following files will be restored via storage side file restore:
/mnt/filer/volume1/datafile01.dbf /mnt/filer/volume1/datafile02.dbf
/mnt/filer/volume1/datafile03.dbf /mnt/filer/volume2/datafile04.dbf
/mnt/filer/volume2/datafile05.dbf
+DG2/sid/datafile08.dbf
+DG2/sid/datafile09.dbf

The following files will be restored via host side file copy restore:
```

```
+DG2/sid/datafile10.dbf
+DG2/sid/datafile11.dbf
```

Example preview: Host-side file copy restore

The following shows an example of some files being restored using the host-side file copy restore process and also why some files cannot be restored using the fast restore option. If you specify the -verbose option, SnapManager displays a preview section and an analysis section with reasons that explain why each file cannot be restored via the fast restore process.

```
PREVIEW:
The following files will be restored via host side file copy restore:
+DG2/sid/datafile10.dbf
+DG2/sid/datafile11.dbf

ANALYSIS:
The following reasons prevent certain files from being restored via fast restore:
Reasons:
 Newer snapshots of filer: /vol/volume2 have volume clones: SNAP_1
*Newer backups will be freed: nightly2, nightly3
Files to Restore:
/mnt/filer/volume2/system.dbf
/mnt/filer/volume2/users.dbf
/mnt/filer/volume2/sysaux.dbf
/mnt/filer/volume2/datafile04.dbf
/mnt/filer/volume2/datafile05.dbf

The following reasons prevent certain files from being restored via fast restore:
Reasons:
* Newer snapshots of filer:/vol/adm_disks will be lost: ADM_SNAP_5
* LUNs present which were created after snapshot SNAP_0 was created: filer:/vol/adm_disks/
disk5.lun
* Files not part of the restore scope will be reverted in filesystem: +DG2

Files Not in Restore Scope: +DG2/someothersid/data01.dbf
+DG2/someothersid/data02.dbf
Files to Restore:
+DG2/sid/datafile08.dbf +DG2/sid/datafile09.dbf
+DG2/sid/datafile10.dbf +DG2/sid/datafile11.dbf

* Reasons denoted with an asterisk (*) are overridable.
```

Example command: Restore database and control files

The following example completely restores a database SALES1 along with the control files.

```
smo backup restore -profile SALES1 -label full_backup_sales_May
-complete -controlfiles -force
```

Related concepts

Restoring database backup on page 175

Related tasks

Restoring backups from an alternate location on page 196
Creating restore specifications on page 194

The smo backup show command

This command displays detailed information about a backup, including its protection status, backup retention class, and backups on primary and secondary storage.

Syntax

```
smo backup show
-profile profile_name
[-label label [-data | -archivelogs] | [-id id]
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the database for which to show backups. The profile contains the identifier of the database and other database information.

-label *label*

Specifies the name of the backup.

-id *id*

Specifies the backup ID.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console, as well as any clone and verification information.

Example command

The following example shows detailed information about the backup.

```
smo backup show -profile SALES1 -label BTNFS -verbose
Backup id: 8abc013111a450480111a45066210001
Backup status: SUCCESS
Primary storage resources: EXISTS
Protection sate: PROTECTED
Retention class: DAILY
Backup scope: FULL
Backup mode: OFFLINE
Mount status: NOT MOUNTED
Backup label: BTNFS
Backup comment:
RMAN Tag: SMO_BTNFS_1175283108815
Backup start time: 2007-03-30 15:26:30
Backup end time: 2007-03-30 15:34:13
Verification status: OK
Backup Retention Policy: NORMAL
```

```
Backup database: hsdb1
Checkpoint: 2700620
Tablespace: SYSAUX
Datafile: /mnt/ssys1/data/hsdb/sysaux01.dbf [ONLINE]
...
Control Files:
File: /mnt/ssys1/data/control03.ctl
...
Archive Logs:
File: /mnt/ssys1/data/archive_logs/2_131_626174106.dbf
...
Host: Host1
Filesystem: /mnt/ssys1/data
File: /mnt/ssys1/data/hsdb/SMOBakCtl_1175283005231_0
...
Volume: hs_data
Snapshot: SMO_HSDBR_hsdb1_F_C_1_
8abc013111a45048011la45066210001_0
File: /mnt/ssys1/data/hsdb/SMOBakCtl_1175283005231_0
...
Protected copies on Secondary Storage:
  14448939 - manow
  88309228 - graffe
```

Related tasks

Viewing backup details on page 151

The smo backup unmount command

This command unmounts a backup.

Syntax

```
smo backup unmount
-profile profile_name
[-label label [-data | -archivelogs] | [-id id]
[-force]
[-dump] [-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the database you want to unmount a backup of. The profile contains the identifier of the database and other database information.

-id *id*

Unmounts the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. Use the `smo backup list` command to display the GUID for each backup.

-label *name*

Unmounts the backup with the specified label.

-dump

Collects the dump files after a successful or failed unmount operation.

-force

Unmounts the backup even if there are problems freeing the resources associated with the backup. SnapManager tries to unmount the backup and clean up any associated resources. The log shows the unmount operation as successful, but you may have to manually clean up resources if there are errors in the log.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following is an example of an unmount operation.

```
# smo backup unmount -label test -profile SALES1 -verbose

SMO-13046 [INFO ]: Operation GUID 8abc013111b909eb0111b90a02f50001 starting on Profile
SALES1
SMO-08028 [INFO ]: Beginning to disconnect connected mount(s)
[/u/user1/mnt/_mnt_ssys1_logs_SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001,
 /u/user1/mnt/_mnt_ssys1_data_SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001].
SMO-08030 [INFO ]: Done disconnecting connected mount(s)
[/u/user1/mnt/_mnt_ssys1_logs_SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001,
 /u/user1/mnt/_mnt_ssys1_data_SMO_SALES1_hsdb1_F_C_1_8abc013111a450480111a45066210001].
SMO-13037 [INFO ]: Successfully completed operation: Backup Unmount
SMO-13048 [INFO ]: Operation Status: SUCCESS
SMO-13049 [INFO ]: Elapsed Time: 0:00:33.715
Operation Id [8abc013111b909eb0111b90a02f50001] succeeded.
```

Related tasks

Unmounting backups on page 153

The smo backup update command

This command sets the retention policy for an existing backup.

Syntax

```
smo backup update
-profile profile_name
[-label backup_name [-data | -archivelogs] | [-id guid]
[-retain {-hourly | -daily | -weekly | -monthly | -unlimited}][-
comment comment_text]
```

```
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the database for which to update backups. The profile contains the identifier of the database and other database information.

-id *guid*

Verifies the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. Use the `smo backup list` command to display the GUID for each backup.

-label *label_name* [-data | -archivelogs]

Enter the backup label and specify the scope of the backup as datafile or archive log.

-comment *comment_text*

Enter text (up to 200 characters) about the backup update. You can include spaces.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

-retain {-hourly | -daily | -weekly | -monthly | -unlimited}

Specifies whether the backup should be retained on an hourly, daily, weekly, monthly, or unlimited basis. If `-retain` is not specified, the retention class defaults to `-hourly`. To retain backups forever, use the `-unlimited` option. The `-unlimited` option makes the backup ineligible for deletion.

Example command

The following example updates the backup labeled full_backup_sales_May to never be automatically deleted and sets the retention policy to unlimited.

```
smo backup update -profile SALES1 -label full_backup_sales_May
-retain -unlimited -comment save_forever_monthly_backup
```

Related tasks

Retaining backups forever on page 148

The smo backup verify command

This command lets you check to see if the backup is in a valid format for Oracle.

Syntax

```
smo backup verify
-profile profile_name
[-label backup_name | [-id guid]
[-retain {-hourly | -daily | -weekly | -monthly | -unlimited}][-force] [-
dump][-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the database you want to verify a backup of. The profile contains the identifier of the database and other database information.

-id *guid*

Verifies the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. Use the `smo backup list` command to display the GUID for each backup.

-label *label_name*

Verifies the backup with the specified label.

-dump

Collects the dump files after the successful or failed backup verify operation.

-force

Forces the database into the necessary state to perform the verify operation.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example verifies the backup.

```
smo backup verify -profile SALES1 -label full_backup_sales_May -quiet

DBVERIFY - Verification starting : FILE = +SMO_1_1161675083835/smo/datafile/data.
277.582482539 ...
```

Related tasks

Verifying database backups on page 147

The smo clone create command

This command creates a clone of a backed up database. You can clone a backup from primary or secondary storage.

Syntax

```
smo clone create
-profile profile_name
[-backup-id backup_guid | -backup-label backup_label_name | -current]
-newsid new_sid
[-host target_host]
[-label clone_label]
[-comment string]
-clonespec full_path_to_clonespec_file
[-reserve {yes | no | inherit}]
[-from-secondary {-copy-id id}]
[-recover-from-location path1 [, path2]] [-taskspec taskspec] [-dump] [-
quiet | -verbose]
```

Parameters

-profile *name*

Specifies the database you want to clone. The profile contains the identifier of the database and other database information.

-backup-id *guid*

Clones the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. Use the `smo backup list-verbose` command to display the GUID for each backup.

-backup-label *backup_label_name*

Clones the backup with the specified label name.

-current

Back up and clone from the current state of the database.

Note: If the database is in `noarchivelog` mode, SnapManager will take an offline backup.

-newsid *new_sid*

Specifies a new, unique Oracle SID for the cloned database. The SID value is a maximum of eight characters. Oracle does not allow running two databases with the same SID on the same host simultaneously.

-host *target_host*

Specifies the host on which the clone should be created.

-label *clone_label*

Specifies a label for the clone.

-comment *string*

Specifies an optional comment to describe this clone. Enclose the string in single quotes (').

Note: Some shells strip off quote marks. If that is true for your shell, you must escape the quote with a backslash (\). For example, you might need to enter: \' this is a comment\'.

-clonespec *full_path_to_clonespec_file*

Specifies the path to the clone specification XML file. This can be a relative or absolute pathname. For information about creating a clone specification, refer to "Creating clone specifications."

-reserve

Setting `-reserve` to `yes` assures the volume guarantee space reserve is turned on for the new clone volumes. Setting `-reserve` to `no` assures the volume guarantee space reserve is turned off for the new clone volumes. Setting `-reserve` to `inherit` assures the new clone inherits the space reservation characteristics of the parent Snapshot copy. If nothing is specified, the default setting is *no*.

The following table describes the cloning methods and their effect on the `clone create` operation and its `-reserve` option. A LUN can be cloned using either method.

Cloning method	Description	`clone create -reserve`
LUN cloning	A new clone LUN is created within the same volume.	When `-reserve` for a LUN is set to `yes`, space is reserved for the full LUN size within the volume.

Cloning method	Description	`clone create -reserve`
Volume cloning	A new FlexClone is created and the clone LUN exists within the new clone volume. Uses the FlexClone technology.	When `-reserve` for a volume is set to `yes`, space is reserved for the full volume size within the aggregate.

`-from-secondary` [`-copy-id` *`copy_id`*]

Indicates that SnapManager should clone a copy of a backup that has been protected to secondary storage. If this option is not specified, SnapManager clones the copy from primary storage.

Use the `-copy-id` option to specify which protected backup to use, if more than one copy exists.

`-recover-from-location`

Specifies the external archive log location of archive log backups where SnapManager takes the archive log files from the external location and uses them for cloning.

`-taskspec`

Specifies the task specification XML file for pre-processing activity or post-processing activity of the clone operation. Make sure you provide the complete path of the task specification XML.

`-dump`

Specifies to collect the dump files after the clone create operation.

`-quiet`

Displays only error messages on the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages on the console.

Example command

The following example clones the backup using a clone specification created for this clone.

```
smo clone create -profile SALES1 -backup-label full_backup_sales_May -newsid
CLONE -label sales1_clone -clonespec /opt/<path>/smo/clonespecs/sales1_clonespec.xml

Operation Id [8abc01ec0e794e3f010e794e6e9b0001] succeeded.
```

Related tasks

Creating clone specifications on page 199
Cloning databases from backups on page 205

The smo clone delete command

This command deletes the specified clone.

Syntax

```
smo clone delete
-profile profile_name
[-id guid | -label clone_name]
-force
[-dump][-quiet | -verbose]
```

Parameters

You cannot delete a running clone operation.

-profile *profile_name*

Specifies the name of the profile containing the clone being deleted. The profile contains the identifier of the database and other database information.

-force

Deletes the clone even if there are resources associated with the clone.

-id *guid*

Specifies the GUID for the clone being deleted. The GUID is generated by SnapManager when you create a clone. Use the `smo clone list` command to display the GUID for each clone.

-label *name*

Specifies the label for the clone being deleted.

-dump

Specifies to collect the dump files after the clone delete operation.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example deletes the clone.

```
smo clone delete -profile SALES1 -label SALES_May
Operation Id [8abc01ec0e79004b010e79006da60001] succeeded.
```

The smo clone list command

This command lists the clones of the database for a given profile.

Syntax

```
smo clone list
-profile profile_name
-delimiter character
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the list of clones associated with the profile. The profile contains the identifier of the database and other database information.

-delimiter *character*

When this parameter is specified, the command lists the attributes in each row separated by the character specified.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example lists the database clones in the SALES1 profile.

```
smo clone list -profile SALES1 -verbose

ID Status SID Host Label Comment
-------- ------- ----- ------- ---------- -------
8ab...01 SUCCESS hsdbc server1 back1clone test comment
```

Related tasks

Viewing a list of clones on page 209

The smo clone show command

This command displays information about the database clones for the specified profile.

Syntax

```
smo clone show
-profile profile_name
[-id  guid | -label clone_name]
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the list of clones associated with the profile. The profile contains the identifier of the database and other database information.

-id *guid*

Shows information about the clone with the specified GUID. The GUID is generated by SnapManager when you create a clone. Use the `smo clone show` command to display the GUID for each clone.

-label *label_name*

Shows information about the clone with the specified label.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example displays information about the clone.

```
smo clone show -profile SALES1 -label full_backup_sales_May -verbose
```

Example output: Clone of a backup on primary storage

The following output shows information about a clone of a backup on primary storage.

```
Clone id: 8abc013111b916e30111b916ffb40001
Clone status: SUCCESS
Clone SID: hsdbc
Clone label: hsdbc
Clone comment: null
Clone start time: 2007-04-03 16:15:50
Clone end time: 2007-04-03 16:18:17
Clone host: Host1
Filesystem: /mnt/ssys1/data_clone
File: /mnt/ssys1/data_clone/hsdb/sysaux01.dbf
File: /mnt/ssys1/data_clone/hsdb/undotbs01.dbf
File: /mnt/ssys1/data_clone/hsdb/users01.dbf
File: /mnt/ssys1/data_clone/hsdb/system01.dbf
File: /mnt/ssys1/data_clone/hsdb/undotbs02.dbf
Backup id: 8abc013111a450480111a45066210001
Backup label: full_backup_sales_May
Backup SID: hsdb1
Backup comment:
Backup start time: 2007-03-30 15:26:30
Backup end time: 2007-03-30 15:34:13
Backup host: server1
```

Example output: Clone of a protected backup on secondary storage

The following output shows information about a clone of a protected backup on secondary storage.

```
clone show -label clone_CLSTEST -profile
TEST_USER_NFSTEST_DIRMAC
Clone id:8abc01ec16514aec0116514af52f0001
Clone status: SUCCESS
Clone SID: CLSTEST
Clone label: clone_CLSTEST
Clone comment:comment_for_clone_CLSTEST
Clone start time: 2007-11-18 00:46:10
Clone end time: 2007-11-18 00:47:54
Clone host: dirmac
Filesystem: /ant/fish/bt_dirmac_nfs_clone
File: /ant/fish/bt_dirmac_nfs_clone/datafiles/sysaux01.dbf
File: /ant/fish/bt_dirmac_nfs_clone/datafiles/system01.dbf
File: /ant/fish/bt_dirmac_nfs_clone/datafiles/undotbs01.dbf
File: /ant/fish/bt_dirmac_nfs_clone/datafiles/users01.dbf
Backup id: 8abc01ec165148830116514888b580001
Backup label:full_backup
Backup SID: NFSTEST
Backup comment:
Backup start time: 2007-11-18 00:43:32
Backup end time: 2007-11-18 00:45:30
Backup host: dirmac
Storage System: fish (Secondary storage)
Volume: bt_dirmac_nfs
Snapshot:smo_user_nfstest_b_nfstest_f_c_1_8abc01ec16511d6a0116511d73590001_0
File: /ant/fish/bt_dirmac_nfs/archlogs/1_14_638851420.dbf
File: /ant/fish/bt_dirmac_nfs/datafiles/sysaux01.dbf
File: /ant/fish/bt_dirmac_nfs/datafiles/undotbs01.dbf
File: /ant/fish/bt_dirmac_nfs/archlogs/1_13_638851420.dbf
File: /ant/fish/bt_dirmac_nfs/archlogs_2/1_16_638851420.dbf
File: /ant/fish/bt_dirmac_nfs/datafiles/users01.dbf
```

```
File: /ant/fish/bt_dirmac_nfs/controlfiles/SMBakCtl_1195361899651_2
File: /ant/fish/bt_dirmac_nfs/datafiles/system01.dbf
```

Related tasks

Viewing detailed clone information on page 210

The smo clone template command

This command lets you create a clone specification template.

Syntax

```
smo clone template
-profile name
[-backup-id guid  | -backup-label backup_name]
[-quiet | -verbose]
```

Parameters

-profile *name*

Specifies the database you want to create a clone specification of. The profile contains the identifier of the database and other database information.

-backup-id *guid*

Creates a clone specification from the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. Use the `smo backup list` command to display the GUID for each backup.

-backup-label *backup_label_name*

Creates a clone specification from the backup with the specified backup label.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example creates a clone specification template from the backup with the label full_backup_sales_May. Once the `smo clone template` command completes, the clone specification template is complete.

```
smo clone template -profile SALES1 -backup-label full_backup_sales_May
Operation Id [8abc01ec0e79004b010e79006da60001] succeeded.
```

Related tasks

Creating clone specifications on page 199
Cloning databases from backups on page 205

The smo clone update command

This command updates information about the clone. You can update the comment.

Syntax

```
smo clone update
-profile profile_name
[-label label | -id id]
-comment  commment_text [-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the name of the profile containing the clone you want to update. The profile contains the identifier of the database and other database information.

-id *id*

Specifies the ID for the clone. The ID is generated by SnapManager when you create a clone. Use the `smo clone list` command to display the ID for each clone.

-label *label*

Specifies the label for the clone.

-comment

Shows the comment entered in the clone creation. This is an optional parameter.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example updates the clone comment.

```
smo clone update -profile anson.pcrac5
-label clone_pcrac51_20080820141624EDT -comment See updated clone
```

The smo clone split-delete command

This command lets you delete a clone split operation cycle entry from a repository database.

Syntax

```
smo clone split-delete
-profile profile [-host hostname]
[-label split-label | -id split-id]
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the profile name of the clone.

-host *hostname*

Specifies the hostname in which the clone exists.

-label *split-label*

Specifies the label name generated by clone split start process.

-id *split-id*

Specifies the unique ID generated by clone split start process.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

The smo clone split-estimate command

This command enables you to view the clone split amount of storage consumed estimate.

Syntax

```
smo clone split-estimate
-profile profile
[-host hostname]
```

```
[-label clone-label | -id clone-id]
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the profile name of the clone.

-host *hostname*

Specifies the hostname in which the clone exists.

-label *clone-label*

Specifies the label name generated by clone process.

-id *clone-id*

Specifies the unique ID generated by clone process.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

The smo clone split command

This command splits a clone and the split clone becomes independent of the original clone. SnapManager generates new profile after the split process. You can use this profile to manage the split clone.

Syntax

```
smo clone split
-profile clone-profile
[-host hostname]
{-label clone-label | -id clone-id} [-split-label split-
operation_label][-comment comment]
-new-profile new-profile-name [-profile-password new-
profile_password]
-repository -dbname repo_service_name
-host repo_host
-port repo_port
-login -username repo_username
-database -dbname db_dbname
-host db_host [-sid db_sid] [-login -username db_username -
password db_password
-port db_port]
[-rman {{-controlfile | {-login -username rman_username
-password rman_password} -tnsname rman_tnsname}}]
```

```
-osaccount osaccount
-osgroup osgroup
[-retain
[-hourly [-count n] [-duration  m]]
[-daily [-count n] [-duration m]]
[-weekly  [-count n] [-duration m]]
[-monthly [-count n] [-duration m]] ]
[-profile-comment profile-comment]
[-snapname-pattern pattern]
[-protect [-protection-policy policy_name]] | [-noprotect]]
[-summary-notification
[-notification
[-success -email email_address1,email_address2
-subject subject-pattern]
[failure -email email_address1,email_address2
-subject subject-pattern] ]
[-separate-archivelog-backups -retain-archivelog-backups  -hours  hours |
-days days |
-weeks weeks |
-months months
[-protect [-protection-policy policy_name | -noprotect]
[-include-with-online-backups | -no-include-with-online-backups]]
[-dump]
[-quiet | -verbose]
```

Parameters

-profile *clone-profile*

Specifies the profile name from which the clone is created.

-host *hostname*

Specifies the hostname in which the clone exists.

-label *clone-label*

Specifies the label name generated by clone process.

-id *clone-id*

Specifies the unique ID generated by clone process.

-split-label *split-operation_label*

Specifies the label name generated by clone process.

-new-profile *new-profile_name*

Specifies the new profile name that SnapManager will generate after successful split operation.

-profile-password *new-profile_password*

Specifies the password for the profile.

-repository

The options that follow *-repository* specify the details of the database for the repository

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-login

Starts the repository login details. The `-login -username` *repo_username* are optional. If not specified, SnapManager defaults to OS Authentication Connection Mode.

-username*repo_username*

Specifies the user name needed to access the database that stores the repository.

-database

The options that follow `-database` specify the details of the database that the profile describes. This is the database that will be backed up, restored, or cloned.

-dbname *db_dbname*

Specifies the name of the database that the profile describes. Use either the global name or the SID.

-host*db_host*

Specifies the name or IP address of the host computer on which the database runs.

-sid *db_sid*

Specifies the SID of the database that the profile describes. By default, SnapManager uses the database name as the SID. If the SID is different from the database name, you must specify it with the `-sid` option.

For example, if you are using Oracle RAC, you must specify the SID of the RAC instance on the RAC node from which SnapManager is executed.

-login

Starts the database login details.

-username *db_username*

Specifies the user name needed to access the database that the profile describes.

-password *db_password*

Specifies the password needed to access the database that the profile describes.

-rman

The options that follow `-rman` specify the details that SnapManager uses to catalog backups with RMAN.

-controlfile

Uses the target database control files instead of a catalog as the RMAN repository.

-login

Starts the RMAN login details.

-password *rman_password*

Specifies the password used to log in to the RMAN catalog.

-username *rman_username*

Specifies the user name used to log in to the RMAN catalog.

-tnsname *tnsname*

Specifies the tnsname connection name (this is defined in the `tsname.ora` file).

-osaccount *osaccount*

Specifies the name of the Oracle database user account. SnapManager uses this account to perform the Oracle operations such as startup and shutdown. It is typically the user who owns the Oracle software on the host, for example, `oracle`.

-osgroup *osgroup*

Specifies the name of the Oracle database group name associated with the `oracle` account.

Note: The `-osaccount` and `-osgroup` variables are required for UNIX but not allowed for databases running on Windows.

-retain [-hourly [-count *n*] [-duration *m*]] [-daily [-count *n*] [-duration *m*]] [-weekly [-count *n*] [-duration *m*]] [-monthly [-count *n*] [-duration *m*]]

Specifies the retention policy for a backup where either or both of a retention count along with a retention duration for a retention class (hourly, daily, weekly, monthly).

For each retention class, either or both of a retention count or a retention duration may be specified. The duration is in units of the class (for example, hours for hourly, days for daily). For instance, if the user specifies only a retention duration of 7 for daily backups, then SnapManager will not limit the number of daily backups for the profile (since the retention count is 0), but SnapManager will automatically delete daily backups created over 7 days ago.

-profile-comment *profile-comment*

Specifies the comment for a profile describing the profile domain.

-snapname-pattern *pattern*

Specifies the naming pattern for Snapshot copies. You can also include custom text, for example, "HAOPS" for highly available operations, in all Snapshot copy names. You can change the Snapshot copy naming pattern when you create a profile or after the profile has been created. The updated pattern applies only to Snapshot copies that have not yet occurred. Snapshot copies that exist retain the previous snapname pattern. You can use several variables in the pattern text.

-protect -protection-policy *policy_name*

Indicates whether the backup should be protected to secondary storage.

Note: If `-protect` is specified without `-protection-policy`, then the dataset will not have a protection policy. If `-protect` is specified and `-protection-policy` is not set when the profile is created, then it may be set later by `smo profile update` command or set by the storage administrator through the N series Management Console data protection capability.

-summary-notification

The option that follows `-notification`. Specify details to configure summary e-mail notification for multiple profiles under a repository database. SnapManager generates this e-mail.

-notification

Specify details to configure e-mail notification for the new profile. SnapManager generates this e-mail. The e-mail notification enables the database administrator to receive e-mails on the succeeded or failed status of the database operation that is performed using this profile.

-success

Specifies to enable e-mail notification for the profile so that e-mails received by recipients when the SnapManager operation succeeds.

-email*e-mail address 1e-mail address 2*

Specifies the e-mail address of the recipient.

-subject*subject-pattern*

Specifies the e-mail subject.

-failure

Specifies to enable e-mail notification for the profile so that e-mails are sent to the recipients when the SnapManager operation fails.

-separate-archivelog-backups

Specifies to separate the archive log backup from datafile backup. This is an optional parameter you can provide while creating the profile. Once the backups are separated using this option, you can either take datafiles-only backup or archive logs-only backup.

-retain-archivelog-backups -hours *hours* | -days *days* | -weeks *weeks*| -months *months*

Specifies to retain the archive log backups based on the archive log retention duration (hourly, daily, weekly, monthly).

protect [-protection-policy *policy_name*] | -noprotect

Specifies to protect the archive log files based on the archive log protection policy.

Specifies not to protect the archive log files using the `-noprotect` option.

-include-with-online-backups | -no-include-with-online-backups

Specifies to include the archive log backup along with the online database backup.

Specifies not to include the archive log backups along with the online database backup.

-dump

Specifies to collect the dump files after the successful profile create operation.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

The smo clone split-result command

This command lets you view the result of the clone split process.

Syntax

```
smo clone split-result
-profile profile
[-host hostname]
[-label split-label | -id split-id]
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the profile name of the clone.

-host *hostname*

Specifies the hostname in which the clone exists.

-label *split-label*

Specifies label name generated by clone split start process.

-id *split-id*

Specifies unique ID generated by clone split start process.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

The smo clone split-stop command

This command stops the running clone split process.

Syntax

```
smo clone split-stop
-profile profile
[-host hostname]
[-label split-label | -id split-id]
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the profile name of the clone.

-host *hostname*

Specifies the hostname in which the clone exists.

-label *split-label*

Specifies the label name generated by clone process.

-id *split-id*

Specifies the unique ID generated by clone process.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

The smo clone split-status command

This command lets you know the progress of running split process.

Syntax

```
smo clone split-status
-profile profile
[-host hostname]
[-label split-label | -id split-id]
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the profile name of the clone.

-host *hostname*

Specifies the hostname in which the clone exists.

-label *split-label*

Specifies the label name generated by clone process.

-id *split-id*

Specifies the unique ID generated by clone process.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

The smo cmdfile command

This command is used if the shell on your host limits the number of characters that can appear on a command line.

Syntax

```
smo cmdfile
-file file_name
[-quiet | -verbose]
```

Put a single command in a text file and use the `smo cmdfile` command to execute the command in the text file.

Note: The `smo cmdfile` command replaces the `smo pfile` command, which is deprecated in this release and will not work. The `smo cmdfile` is not backwards compatible with the `smo pfile` command.

You can put only one smo command in a text file. Within the text file, drop the smo from the command syntax.

Parameters

`-file` ***`file_name`***

Specifies the path to text file containing the command you want to execute.

`-quiet`

Specifies that only error messages are displayed on the console. The default is to display error and warning messages.

`-verbose`

Specifies that error, warning, and informational messages are displayed on the console.

Example command

This example creates a profile by putting the `profile` command in a text file and then calling the `smo cmdfile` command. A text file called command.txt in `/tmp` has the following command in it:

```
profile create -profile SALES1 -repository -dbname SNAPMGRR
-login -username server1_user -password ontap -port 1521 -host server1
-database -dbname SMO -sid SMO -login -username sys -password oracle -port 1521
-host Host2 -osaccount oracle -osgroup db2
```

```
Now you can create the profile by entering the smo cmdfile
 command with that file.
```

```
smo cmdfile -file /tmp/command.txt
```

The smo credential clear command

This command clears the cache of the user credentials for all secured resources.

Syntax

```
smo credential clear
[-quiet | -verbose]
```

Parameters

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

This example clears all of the credentials for the user running the command.

```
smo credential clear -verbose

SMO-20024 [INFO ]: Cleared credentials for user "user1".
```

Related tasks

Clearing user credentials for all hosts, repositories, and profiles on page 96

The smo credential delete command

This command deletes the user credentials for a particular secured resource.

Syntax

```
smo credential delete
[-host -name host_name
-username username] |
[-repository
-dbname repo_service_name
-host repo_host
-login -username repo_username
-port repo_port] |
[-profile
-name profile_name]
[-quiet | -verbose]
```

Parameters

-host *hostname*

Specifies the name of the host server on which SnapManager is running.

The `-host` parameter includes the following options:

- `-name` *`host_name`*: Specifies the name of the host for which you will delete the password.

- `-username` *`user_name`*: Specifies the user name on the host.

`-repository-dbname`

Specifies the name of the database that stores the profile. Use either the global name or the SID.

The `-repository` parameter includes the following options:

- `-dbname` *`repo_service_name`*: Specifies the name of the database that stores the profile. Use either the global name or the SID.
- `-host` *`repo_host`*: Specifies the name or IP address of the host server the repository database runs on.
- `-login -username` *`repo_username`*: Specifies the user name needed to access the database that stores the repository.
- `-port` *`repo_port`*: Specifies the TCP port number used to access the database that stores the repository.

`-profile-name` ***`profile_name`***

Specifies the profile with which the database is associated.

The `-profile` parameter includes the following option:

- `-name` *`profilename`*: Specifies the name of the profile for which you will delete the password.

`-quiet`

Displays only error messages on the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages on the console.

Example command

This example deletes the credentials of the profile.

```
smo credential delete -profile -name user1 -verbose

SMO-20022 [INFO ]: Deleted credentials and repository mapping
 for profile "user1" in user credentials for "user1".
```

This example deletes the credentials of the repository.

```
smo credential delete -repository -dbname SMOREPO -host Host2
-login -username user1 -port 1521

SMO-20023 [INFO ]: Deleted repository credentials for "user1@SMOREPO/wasp:1521"
and associated profile mappings in user credentials for "user1".
```

This example deletes the credentials of the host.

```
smo credential delete -host -name Host2

SMO-20033 [INFO ]: Deleted host credentials for "Host2" in user credentials for "user1".
```

Related tasks

Deleting credentials for individual resources on page 98

The smo credential list command

This command lists all credentials of a user.

Syntax

```
smo credential list
[-quiet | -verbose]
```

Parameters

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

This example displays all of the credentials for the user running the command.

```
smo credential list

Credential cache for OS user "user1":
Repositories:
Host1_test_user@SMOREPO/hotspur:1521
Host2_test_user@SMOREPO/hotspur:1521
user1_1@SMOREPO/hotspur:1521
Profiles:
HSDBR (Repository: user1_2_1@SMOREPO/hotspur:1521)
PBCASM (Repository: user1_2_1@SMOREPO/hotspur:1521)
HSDB (Repository: Host1_test_user@SMOREPO/hotspur:1521) [PASSWORD NOT SET]
Hosts:
Host2
Host5
Host4
Host1
```

Related tasks

Viewing user credentials on page 96

The smo credential set command

This command lets you set the credentials for users to access secure resources, such as hosts, repositories, and database profiles. The host password is the user's password on the host on which SnapManager is running. The repository password is the password of the Oracle user that contains the SnapManager repository schema. The profile password is a password that is made up by the person who creates the profile. For the host and repository options, if the optional `-password` option is not included, you will be prompted to enter a password of the type specified in the command arguments.

Syntax

```
smo credential set
[-host
-name host_name
-username username]
[-password password] ] |
[-repository
-dbname repo_service_name
-host repo_host
-login -username repo_username] [-password repo_password] ]
-port repo_port |
[-profile
-name profile_name]
[-password password] ]
[-quiet | -verbose]
```

Parameters

`-host` *`hostname`*

Specifies the name or IP address of the host server on which SnapManager is running.

The `-host` parameter includes the following options:

- `-name` *`host_name`*: Specifies the name of the host for which you will set the password.
- `-username` *`user_name`*: Specifies the user name on the host.
- `-password` *`password`*: Specifies the password of the user on the host.

`-repository -dbname`

Specifies the name of the database that stores the profile. Use either the global name or the SID.

The `-repository` parameter includes the following options:

- `-dbname` *repo_service_name*: Specifies the name of the database that stores the profile. Use either the global name or the SID.
- `-host` *repo_host*: Specifies the name or IP address of the host server the repository database runs on.
- `-login -username` *repo_username*: Specifies the user name needed to access the database that stores the repository.
- `-password` *password*: Specifies the password needed to access the database that stores the repository.
- `-port` *repo_port*: Specifies the TCP port number used to access the database that stores the repository.

`-profile -name` *profile_name*

Specifies the profile with which the database is associated.

The `-profile` parameter includes the following option:

- `-name` *profilename*: Specifies the name of the profile for which you will set the password.
- `-password` *password*: Specifies the password needed to access the profile.

`-quiet`

Displays only error messages on the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages on the console.

Example command for setting repository credentials

The following example sets credentials for a repository:

```
smo credential set -repository -dbname SMOREPO -host hotspur -port 1521 -login -username
chris
Password for chris@hotspur:1521/SMOREPO : ********
Confirm password for chris@hotspur:1521/SMOREPO : ********

SMO-12345 [INFO ]: Updating credential cache for OS user "admin1"
SMO-12345 [INFO ]: Set repository credential for user "user1" on repo1@Host2.
Operation Id [Nff8080810da9018f010da901a0170001] succeeded.
```

Example command for setting host credentials

Because a host credential represents an actual operating system credential, it must include the username in addition to the password.

```
smo credential set -host -name bismarck -username avida
Password for avida@bismarck : ********
Confirm password for avida@bismarck : ********
```

Related concepts

SnapManager security on page 36

The smo history list command

This command enables you to view a list of history details of the SnapManager operation.

Syntax

```
smo history list
-profile {-name profile_name [profile_name1, profile_name2] | -all-
repository -login [-password repo_password] -username repo_username-
host repo_host
-dbname repo_dbname
-port repo_port}
-operation {-operations operation_name [operation_name1,
operation_name2] | -all}
[-delimiter character][-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-repository

The options that follow -repository specify the details of the database that stores the profile.

-dbname *repo_dbname*

Specifies the name of the database that stores the profile. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-operation {-operations *operation_name* |*operation_name1*, *operation_name2*] | -all

Specifies the SnapManager operation for which you configure the history.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

```
smo  history list -profile -name PROFILE1 -operation -operations
backup -verbose
```

The smo history operation-show command

This command enables you to view the history of a specific SnapManager operation associated with a profile.

Syntax

```
smo history operation-show
-profile profile{-label label | -id id}[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-label *label* | -id *id*

Specifies the SnapManager operation ID or label for which you want to view the history.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

```
smo  history operation-show -profile PROFILE1 -label backup1 -verbose
```

The smo history purge command

This command enables you to delete the history of SnapManager operation.

Syntax

```
smo history purge
-profile {-name profile_name [profile_name1, profile_name2] | -all-
repository -login [-password repo_password] -username repo_username-
host repo_host
-dbname repo_dbname
-port repo_port}
-operation {-operations operation_name [operation_name1,
operation_name2] | -all}
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-repository

The options that follow -repository specify the details of the database that stores the profile.

-dbname *repo_dbname*

Specifies the name of the database that stores the profile. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-operation {-operations *operation_name* |*operation_name1*, *operation_name2*] |
-all

Specifies the SnapManager operation for which you configure the history.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

```
smo  history purge -profile -name PROFILE1 -operation -operations
backup
 -verbose
```

The smo history remove command

This command enables you to remove the history of SnapManager operations associated with a single profile, multiple profiles, or all profiles under a repository.

Syntax

```
smo history remove
-profile {-name profile_name [profile_name1, profile_name2] | -all-
repository -login [-password repo_password] -username repo_username-
host repo_host
-dbname repo_dbname
-port repo_port}
-operation {-operations operation_name [operation_name,
operation_name2] | -all}
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-repository

The options that follow -repository specify the details of the database that stores the profile.

-dbname *repo_dbname*

Specifies the name of the database that stores the profile. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-operation {-operations *operation_name* [*operation_name1*, *operation_name2*] | -all

Specifies the SnapManager operation for which you configure the history.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

```
smo  history purge -profile -name PROFILE1 -operation -operations
backup
 -verbose
```

The smo history set command

This command enables you to configure the history of SnapManager operation performed using a single profile, multiple profiles, or all profiles under a repository.

Syntax

```
smo history set
-profile {-name profile_name [profile_name1, profile_name2] | -all-
repository -login [password repo_password] -username repo_username-
host repo_host
-dbname repo_dbname
```

```
-port repo_port}
-operation {-operations operation_name [operation_name1,
operation_name2] | -all}
-retain
{-count retain_count | -daily daily_count | -monthly monthly_count |
-weekly weekly_count}
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-repository

The options that follow -repository specify the details of the database that stores the profile.

-dbname *repo_dbname*

Specifies the name of the database that stores the profile. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-operation {-operations *operation_name* [*operation_name1*, *operation_name2*] | -all

Specifies the SnapManager operation for which you configure the history.

-retain {-count *retain_count* | -daily *daily_count* | -monthly *-monthly_count* | -weekly *weekly_count*}

Specifies the retention class of the create backup, verify backup, restore and recover, create clone, and split clone operations. The retention class is set based on the number of operation count, number of days, weeks, or months.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

```
smo  history set -profile -name PROFILE1 -operation -operations
backup -retain -daily 6
 -verbose
```

The smo history show command

This command enables you to view a detailed history information for a specific profile.

Syntax

```
smo history show
-profile profile
```

Parameters

-profile *profile*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

```
smo  history show -profile -name PROFILE1
 -verbose
```

The smo help command

This command displays information about the SnapManager commands and their options. If you do not supply a command name, it displays a list of valid commands. If you supply a command name, it displays the syntax for that command.

Syntax

```
smo help
[backup|cmdfile|clone|credential|help|operation|profile|protection-policy|
repository|system|version|plugin|diag|history|schedule|notification|
storage|get]
[-quiet | -verbose]
```

Parameters

Valid command names you can use with this command are:

- backup
- clone
- cmdfile
- credential
- diag
- get
- notification
- help
- history
- operation
- plugin
- profile
- protection-policy
- repository
- schedule
- storage
- system
- version

The smo notification remove-summary-notification command

This command disables summary notification for multiple profiles on a repository database.

Syntax

```
smo notification remove-summary-notification
-repository
-dbname repo_service_name
-port repo_port
-host repo_host
-login -username repo_username
[-quiet | -verbose]
```

Parameters

-repository

The options that follow `-repository` specify the details of the database for the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login *repo_username*

Specifies the login name needed to access the database that stores the repository.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

The following example disables summary notification for multiple profiles on a repository database:

```
smo notification remove-summary-notification -repository -port 1521 -
dbname repo2 -host 10.72.197.133 -login -username oba5
```

The smo notification update-summary-notification command

This command enables summary notification for a repository database.

Syntax

```
smo notification update-summary-notification
-repository
-port repo_port
-dbname  repo_service_name
-host repo_host
-login -username repo_username
-email email-address1,email-address2
-subject subject-pattern
-frequency
[-daily -time daily_time |
-hourly -time hourly_time |
-monthly -time monthly_time -date [1|2|3|...|31] |
-weekly  -time weekly_time -day [1|2|3|4|5|6|7]] -
profiles profile1,profile2
[-quiet | -verbose]
```

Parameters

-repository

The options that follow `-repository` specify the details of the database for the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details. The `-login -username` *db_username* `-port` *db_port* are optional. If not specified, SnapManager defaults to OS Authentication Connection Mode.

-username ***repo_username***

Specifies the user name needed to access the database that stores the repository.

-email ***email-address1,e-mail-address2***

Specifies e-mail addresses of the recipients.

-subject ***subject-pattern***

Specifies the e-mail subject pattern.

-frequency {
-daily -time ***daily_time*** **| -hourly -time** ***hourly_time*** **| -monthly -time** ***monthly_time*** **-date {*1|2|3...|31*} | -weekly -time** ***weekly_time*** **-day {*1|2|3|4|5|6|7*} }**

Specifies schedule type and schedule time at which you require e-mail notification.

-profiles ***profile1, profile2***

Specifies profile names that require e-mail notification.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

The following example enables summary notification for a repository database:

```
smo notification update-summary-notification -repository -port 1521 -
dbname repo2 -host 10.72.197.133 -login -username oba5 -email
admin@org.com -subject success -frequency -daily -time 19:30:45 -
profiles sales1
```

The smo notification set command

This command helps you to configure the mail server.

Syntax

```
smo notification set
-sender-email email_address
-mailhost mailhost
-mailport mailport
```

```
[-authentication
-username username
-password password]
-repository
-dbname repo_service_name
-port repo_port]
-host repo_host
-login -username repo_username
[-quiet | -verbose]
```

Parameters

-sender-email *email_address*

Specifies the senders e-mail address from which the e-mail alerts are sent. From SnapManager 3.2 for Oracle, you can include hyphen (-) while specifying the domain name of the e-mail address. For example, you can specify the sender e-mail address as `-sender-email` *07lbfmdatacenter@continental-corporation.com*.

-mailhost *mailhost*

Specifies the name or IP address of the host server that handles e-mail notification.

-mailport *mailport*

Specifies the mail server port number.

-authentication -username *username* -password *password*

Specifies authentication for the e-mail address. For this, you must specify the username and password.

-repository

The options that follow `-repository` specify the details of the database for the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details. The `-login -username` *db_username* `-port` *db_port* are optional. If not specified, SnapManager defaults to OS Authentication Connection Mode.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Command example

The following example configures the mail server:

```
smo notification set -sender-email admin@org.com -mailhost hostname.org.com -mailport 25
authentication -username davis -password davis -repository -port 1521 -dbname SMOREPO -
host hotspur
-login -username grabal21 -verbose
```

The smo operation dump command

This command creates a jar file that contains diagnostic information about an operation.

Syntax

```
smo operation dump
-profile profile_name
[-label label_name | -id guid]
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the database for which you want to create the dump files. The profile contains the identifier of the database and other database information.

-label *label_name*

Creates dump files for the operation and assigns the specified label.

-id *guid*

Creates dump files for the operation with the specified GUID. The GUID is generated by SnapManager when the operation begins.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example creates dump file for the backup with the GUID:

```
smo operation dump -profile SALES1
-id 8abc01ec0e78f3e2010e78f3fdd00001

Dump file created Path:/userhomedirectory/.ibm/smo/3.0/
smo_dump_8abc01ec0e78f3e2010e78f3fdd00001.jar
```

Related concepts

Dump files on page 364

The smo operation list command

This command lists the summary information of all operations recorded against a specified profile.

Syntax

```
smo operation list
-profile profile_name
[-delimiter character]
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-delimiter *character*

(Optional) When this parameter is specified, the command lists each row on a separate line and the attributes in that row are separated by the character specified.

-quiet

(Optional) Displays only error messages on the console. The default is to display error and warning messages.

-verbose

(Optional) Displays error, warning, and informational messages on the console.

Example command

The following example lists the summary information of all the operations logged against the specified profile:

```
smo operation list -profile myprofile
```

```
Start Date Status Operation ID Type Host
------------------- ---------- -------------------------------- ------------------ --
2007-07-16 16:03:57 SUCCESS 8abc01c813d0a1530113d0a15c5f0005 Profile Create Host3
2007-07-16 16:04:55 FAILED 8abc01c813d0a2370113d0a241230001 Backup Host3
2007-07-16 16:50:56 SUCCESS 8abc01c813d0cc580113d0cc60ad0001 Profile Update Host3
2007-07-30 15:44:30 SUCCESS 8abc01c81418a88e011418a8973e0001 Remove Backup Host3
2007-08-10 14:31:27 SUCCESS 8abc01c814510ba20114510bac320001 Backup Host3
2007-08-10 14:34:43 SUCCESS 8abc01c814510e9f0114510ea98f0001 Mount Host3
2007-08-10 14:51:59 SUCCESS 8abc01c814511e6e0114511e78d40001 Unmount Host3
```

Related tasks

Viewing a list of operations on page 213

The smo operation show command

This command lists the summary information of all operations recorded against the specified profile. The output lists the client user (the user for the client PC) and the effective user (the user in SnapManager that is valid on the selected host).

Syntax

```
smo operation show
-profile profile_name
[-label label | -id id] [-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-label *label*

Specifies the label for the operation.

-id *id*

Specifies the identifier for the operation.

-quiet

(Optional) Displays only error messages on the console. The default is to display error and warning messages.

-verbose

(Optional) Displays error, warning, and informational messages on the console.

Show properties example of an operation

The following command line shows detailed information on an operation:

```
# smo operation show -profile myprofile -id ff8080811295eb1c011295eb28230001

Operation Attempted
   Operation ID: ff8080811295eb1c011295eb28230001
   Type:RestoreFor profile: myprofile
   With Force: No
   Performed on backup
   Operation ID: ff8080811295eb1c011296eb23290001
   Label: mylabel
Operation Runtime Information
   Status: SUCCESS
   Start date: 2007-07-16 13:24:09 IST
   End date: 2007-07-16 14:10:10 IST
   Client user: amorrow
   Effective user: amorrow
Host
   Host Run upon: Host3
   Process ID: 3122
   SnapManager version: 3.1
Repository
   Connection: user1@SMOREPO/hotspur:1521
   Repository version: 3.1
Resources in use
   Volume:
     ssys1:/vol/luke_ES0_0 (FlexClone)
   Filesystems:
     /opt/ibm/smo/mnt/-mnt_ssys1_luke_ES0_smo_e_es0_f_c_1_8abc0112129b0f81580001_0
```

Related tasks

Viewing operation details on page 214

The smo plugin check command

SnapManager provides the ability for you to install and use custom scripts to perform various operations. SnapManager offers backup, restore, and clone plug-ins, which you can use to automate your custom scripts before and after backup, restore, and clone operations. Before you use the backup, restore, and clone plug-in, use the plugin check command to verify the installation of plug-in scripts on the SnapManager server. Custom scripts are stored in three directories: policy (for scripts that should always be run before the backup, restore, or clone operation occurs), pre (for pre-processing scripts), and post (for post-processing scripts). This command applies to the server side installation of SnapManager.

Syntax

```
smo plugin check
 -osaccount os_db_user_name
```

Parameter

`- osaccount`

Specifies the OS database user name. If you omit the `-osaccount` option, SnapManager checks the plug-in scripts as root rather than for a specific user.

Example command

The following example illustrates that the plugin check command found "policy1" custom script stored in the policy directory is executable. The example also shows that the two other custom scripts stored in the pre directory return no error messages (show with a status of "0"); however, the fourth custom script ("post-plugin1"), which was found in the post directory, contains errors (shown with a status of "3").

```
smo plugin check
Checking plugin directory structure ...
<installdir>/plugins/clone/policy
OK: 'policy1' is executable
<installdir>/plugins/clone/pre
OK: 'pre-plugin1' is executable and returned status 0
OK: 'pre-plugin2' is executable and returned status 0
<installdir>/plugins/clone/post
ERROR: 'post-plugin1' is executable and returned status 3
<installdir>/plugins/backup/policy
OK: 'policy1' is executable
<installdir>/plugins/backup/pre
OK: 'pre-plugin1' is executable and returned status 0
OK: 'pre-plugin2' is executable and returned status 0
<installdir>/plugins/backup/post
ERROR: 'post-plugin1' is executable and returned status 3
<installdir>/plugins/restore/policy
OK: 'policy1' is executable
<installdir>/plugins/restore/pre
OK: 'pre-plugin1' is executable and returned status 0
OK: 'pre-plugin2' is executable and returned status 0
<installdir>/plugins/restore/post
ERROR: 'post-plugin1' is executable and returned status 3
Command complete.
```

Related tasks

Cloning databases and using custom plug-in scripts on page 204

The smo profile create command

This command creates a profile of a database in a repository. To create the profile, the database must be able to be mounted.

Syntax

```
smo profile create
-profile profile[-profile-password profile_password]
-repository
-dbname repo_service_name
-host repo_host
```

```
-port repo_port
-login -username repo_username
-database
-dbname db_dbname
-host db_host
[-sid  db_sid]
[-login
-username db_username
-password db_password
-port db_port]
[-rman {-controlfile | {-login
-username  rman_username -password  rman_password}
-tnsname  rman_tnsname}}]
-osaccount osaccount
-osgroup osgroup
[-retain
[-hourly [-count n] [-duration m]]
[-daily [-count n] [-duration m]]
[-weekly [-count n] [-duration m]]
[-monthly [-count n] [-duration m]]]]
-comment comment
-snapname-pattern pattern
[-protect [-protection-policy policy]]
[-summary-notification]
[-notification
[-success
-email email_address1,email_address2
-subject subject_pattern]
[-failure
-email email_address1,email_address2
-subject subject_pattern]
[-separate-archivelog-backups -retain-archivelog-backups     -hours hours
|
-days days |
-weeks weeks |
-months months
[-protect [-protection-policy policy_name | -noprotect]
[-include-with-online-backups | -no-include-with-online-backups]]
[-dump]
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-profile-password *profile_password*

Specify the password for the profile.

-repository

The options that follow `-repository` specify the details of the database that stores the profile.

-dbname *repo_service_name*

Specifies the name of the database that stores the profile. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-sid *db_sid*

Specifies the SID of the database that the profile describes. By default, SnapManager uses the database name as the SID. If the SID is different from the database name, you must specify it with the `-sid` option.

For example, if you are using Oracle RAC, you must specify the SID of the RAC instance on the RAC node from which SnapManager is executed.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-database

The options that follow `-database` specify the details of the database that the profile describes. This is the database that will be backed up, restored or cloned.

-dbname *db_dbname*

Specifies the name of the database that the profile describes. Use either the global name or the SID.

-host db_host *db_host*

Specifies the name or IP address of the host computer on which the database runs.

-login

Starts the database login details.

-username *db_username*

Specifies the user name needed to access the database that the profile describes.

-password *db_password*

Specifies the password needed to access the database that the profile describes.

-port *db_port*

Specifies the TCP port number used to access the database that the profile describes.

-rman

The options that follow `-rman` specify the details that SnapManager uses to catalog backups with RMAN.

-controlfile

Uses the target database control files instead of a catalog as the RMAN repository.

-login

Starts the RMAN login details.

-password *rman_password*

Specifies the password used to log in to the RMAN catalog.

-username *rman_username*

Specifies the user name used to log in to the RMAN catalog.

-tnsname *tnsname*

Specifies the tnsname connection name (this is defined in the tsname.ora file).

-osaccount *osaccount*

Specifies the name of the Oracle database user account. SnapManager uses this account to perform the Oracle operations such as startup and shutdown. It is typically the user who owns the Oracle software on the host, for example, `oracle`.

-osgroup *osgroup*

Specifies the name of the Oracle database group name associated with the `oracle` account.

Note: The `-osaccount` and `-osgroup` variables are required for UNIX but not allowed for databases running on Windows.

-retain [-hourly [-count *n*] [-duration *m*]] [-daily [-count *n*] [-duration *m*]] [-weekly [-count *n*] [-duration *m*]] [-monthly [-count *n*] [-duration *m*]]

Specifies retention policy for a backup where either or both of a retention count along with a retention duration for a retention class (hourly, daily, weekly, monthly).

For each retention class, either or both of a retention count or a retention duration may be specified. The duration is in units of the class (for example, hours for hourly, days for daily). For instance, if the user specifies only a retention duration of 7 for daily backups, then SnapManager will not limit the number of daily

backups for the profile (since the retention count is 0), but SnapManager will automatically delete daily backups created over 7 days ago.

-comment *comment*

Specifies the comment for a profile describing the profile domain.

-snapname-pattern *pattern*

Specifies the naming pattern for Snapshot copies. You can also include custom text, for example, "HAOPS" for highly available operations, in all Snapshot copy names. You can change the Snapshot copy naming pattern when you create a profile or after the profile has been created. The updated pattern applies only to Snapshot copies that have not yet occurred. Snapshot copies that exist retain the previous snapname pattern. You can use several variables in the pattern text.

-protect -protection-policy *policy*

Indicates whether the backup should be protected to secondary storage.

Note: If `-protect` is specified without `-protection-policy`, then the dataset will not have a protection policy. If `-protect` is specified and `-protection-policy` is not set when the profile is created, then it may be set later by `smo profile update` command or set by the storage administrator through the N series Management Console data protection capability.

-summary-notification

Specifies to enable summary e-mail notification for the new profile.

-notification -success -email *e-mail_address1,e-mail address2* -subject *subject_pattern*

Specifies to enable e-mail notification for the new profile so that e-mails are received by recipients when the SnapManager operation succeeds. You must enter a single e-mail address or multiple e-mail addresses to which e-mail alerts will be sent and an e-mail subject pattern for the new profile.

You can also include custom subject text for the new profile. You can change the subject text when you create a profile or after the profile has been created. The updated subject applies only to the e-mails that are not sent. You can use several variables for the e-mail subject.

-notification -failure -email *e-mail_address1,e-mail address2* -subject *subject_pattern*

Specifies to enable e-mail notification for the new profile so that e-mails are received by recipients when the SnapManager operation fails. You must enter a single e-mail address or multiple e-mail addresses to which e-mail alerts will be sent and an e-mail subject pattern for the new profile.

You can also include custom subject text for the new profile. You can change the subject text when you create a profile or after the profile has been created. The

updated subject applies only to the e-mails that are not sent. You can use several variables for the e-mail subject.

-separate-archivelog-backups

Specifies to separate the archive log backup from datafile backup. This is an optional parameter you can provide while creating the profile. Once the backups are separated using this option, you can either take datafiles-only backup or archive logs-only backup.

-retain-archivelog-backups -hours *hours* | -days *days* | -weeks *weeks*| -months *months*

Specifies to retain the archive log backups based on the archive log retention duration (hourly, daily, weekly, monthly).

protect [-protection-policy *policy_name*] | -noprotect

Specifies to protect the archive log files based on the archive log protection policy.

Specifies not to protect the archive log files using the `-noprotect` option.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

-include-with-online-backups | -no-include-with-online-backups

Specifies to include the archive log backup along with the online database backup.

Specifies not to include the archive log backups along with the online database backup.

-dump

Specifies to collect the dump files after the successful profile create operation.

Example command

The following example creates a profile named test_rbac with hourly retention policy and e-mail notification:

```
smo profile create -profile test_rbac -profile-password test123 -repository -dbname
SMOREP -host hostname.org.com -port 1521 -login -username smorep -database -dbname RACB -
host saal -sid racb1 -login -username sys -password test123 -port 1521 -osaccount oracle -
osgroup dba -rman -controlfile -retain -hourly -count 30 -verbose
Operation Id [8abc01ec0e78ebda010e78ebe6a40005] succeeded.
```

Related concepts

Managing profiles for efficient backups on page 101
Snapshot copy naming on page 106

How SnapManager determines which backups to retain on local storage on page 103

The smo profile delete command

This command deletes a profile of a database.

Syntax

```
smo profile delete
-profile                        profile
-force
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the profile to be deleted.

-force

Attempts to forcefully delete the profile.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example deletes the profile named SALES1:

```
smo profile delete -profile SALES1
Operation Id [Ncaf00af0242b3e8dba5c68a57a5ae932] succeeded.
```

Related tasks

Deleting profiles on page 116

The smo profile destroy command

This command deletes the split clone (database) along with the profile generated by SnapManager during the clone split process.

Syntax

```
smo profile destroy
-profile profile
```

```
[-host hostname]
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the profile that SnapManager generates after a successful clone split process.

-host *hostname*

Specifies the hostname in which the split clone exists.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example deletes the profile named SALES1:

```
smo profile destroy -profile SALES1
```

The smo profile dump command

This command creates a jar file that contains diagnostic information about a profile.

Syntax

```
smo profile dump
-profile profile_name
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the profile for which you want to create the dump files. The profile contains the identifier of the database and other database information.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example creates a dump for the profile SALES1:

```
smo profile dump -profile SALES1
Dump file created
Path:/userhomedirectory/.ontap/smo/3.1.0/smo_dump_SALES1_hostname.jar
```

The smo profile list command

This command displays a list of the current profiles.

Syntax

```
smo profile list
[-quiet | -verbose]
```

Parameters

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example displays existing profiles with their details:

```
smo profile list -verbose
Profile name: FGTER
Repository:
  Database name: SMOREPO
  SID: SMOREPO
  Host: hotspur
  Port: 1521
  Username: swagrahn
  Password: ********
Profile name: TEST_RBAC
Repository:
  Database name: smorep
  SID: smorep
  Host: elbe.rtp.org.com
  Port: 1521
  Username: smosaal
  Password: ********
Profile name: TEST_RBAC_DP_PROTECT
Repository:
  Database name: smorep
  SID: smorep
  Host: elbe.rtp.org.com
  Port: 1521
  Username: smosaal
  Password: ********
Profile name: TEST_HOSTCREDEN_OFF
Repository:
  Database name: smorep
```

```
  SID: smorep
  Host: elbe.rtp.org.com
  Port: 1521
  Username: smosaal
  Password: ********
Profile name: SMK_PRF
Repository:
  Database name: smorep
  SID: smorep
  Host: elbe.rtp.org.com
  Port: 1521
  Username: smosaal
  Password: ********
Profile name: FGLEX
Repository:
  Database name: SMOREPO
  SID: SMOREPO
  Host: hotspur
  Port: 1521
  Username: swagrahn
  Password: ********
```

The smo profile show command

This command displays information about a profile.

Syntax

```
smo profile show
-profile profile_name
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example shows the details of the TEST_RBAC_DP_PROTECT profile:

```
smo profile show -profile TEST_RBAC_DP_PROTECT -verbose
Profile name: TEST_RBAC_DP_PROTECT
Comment:
Target database:
  Database name: racb
  SID: racb1
  Host: saal
  Port: 1521
```

```
  Username: sys
  Password: ********
Repository:
  Database name: smorep
  SID: smorep
  Host: elbe.rtp.org.com
  Port: 1521
  Username: smosaal
  Password: ********
RMAN:
  Use RMAN via control file
Oracle user account: oracle
Oracle user group: dba
Snapshot Naming:
  Pattern: smo_{profile}_{db-sid}_{scope}_{mode}_{smid}
  Example: smo_test_rbac_dp_protect_racb1_f_h_1_8abc01e915a55ac50115a55acc8d0001_0
Protection:
  Dataset: smo_saal_racb
  Protection policy: Back up
  Conformance status: CONFORMANT
Local backups to retain:
  Hourly: 4 copies
  Daily: 7 day(s)
  Weekly: 4 week(s)
  Monthly: 12 month(s)
```

The smo profile sync command

This command loads the profile-to-repository mappings for that repository to a file in your home directory on the local host.

Syntax

```
smo profile sync
-repository
-dbname repo_service_name
-host repo_host
-port repo_port
-login
-username repo_username                    [-quiet | -verbose]
```

Parameters

-repository

The options that follow `-repository` specify the details of the database for the repository.

-dbname *repo_service_name*

Specifies the repository database for the profile to synchronize.

-host

Specifies the database host.

-port

Specifies the port for the host.

-login

Specifies the log in process for the host user.

-username

Specifies the username for the host.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example shows the result of the command to synchronize the profile-to-repository mappings for the database:

```
smo profile sync -repository -dbname smrepo -host Host2 -port 1521 -login -username user2
SMO-12345 [INFO ]: Loading profile mappings for repository "user2@Host2:smrepo" into
cache for OS User "admin".
Operation Id [Nff8080810da9018f010da901a0170001] succeeded.
```

The smo profile update command

This command modifies the information in an existing profile.

Syntax

```
smo profile update
-profile profile [-profile-password profile_password]
[-database
-dbname db_dbname
-host db_host
[-sid db_sid]
[-login -username  db_username -password  db_password
-port db_port ]]
[{-rman  {-controlfile | {{-login
-username  rman_username
-password  rman_password }
[-tnsname  tnsname]}}} |
-remove-rman]
-osaccount osaccount
-osgroup osgroup
[-retain
[-hourly [-count n] [-duration m]]
[-daily [-count n] [-duration m]]
[-weekly [-count n] [-duration m]]
[-monthly [-count n] [-duration m]]]]
```

```
-comment comment
-snapname-patternpattern
[-protect [-protection-policy policy_name] | [-noprotect]]
[-summary-notification]
[-notification
[-success
-email email_address1,email_address2
-subject subject_pattern]
[-failure
-email email_address1,email_address2
-subject subject_pattern]
[-separate-archivelog-backups -retain-archivelog-backups-hours  hours |
-days days |
-weeks weeks |
-months months
[-protect [-protection-policy policy_name] | [-noprotect]]
[-include-with-online-backups | -no-include-with-online-backups]]
[-dump]
[-quiet | -verbose]
```

Parameters

If protection policy was set on the profile, you cannot change the policy using SnapManager. The storage administrator must change the policy using the N series Management Console data protection capability.

-profile *profile*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-profile-password *profile_password*

Specify the password for the profile

-database

The options that follow `-database` specify the details of the database that the profile describes. This is the database that will be backed up, restored, and so on.

-dbname *db_dbname*

Specifies the name of the database that the profile describes. Use either the global name or the SID.

-host *db_host*

Specifies the name or IP address of the host computer the database runs on.

-sid *db_sid*

Specifies the SID of the database that the profile describes. By default, SnapManager uses the database name as the SID. If the SID is different from the database name, you must specify it with the `-sid` option.

For example, if you are using Oracle RAC, you must specify the SID of the RAC instance on the RAC node from which SnapManager is executed.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-database

The options that follow `-database` specify the details of the database that the profile describes. This is the database that will be backed up, restored, or cloned.

-dbname *db_dbname*

Specifies the name of the database that the profile describes. Use either the global name or the SID.

-host *db_host*

Specifies the name or IP address of the host computer on which the database runs.

-login

Starts the database login details.

-username *db_username*

Specifies the user name needed to access the database that the profile describes.

-password *db_password*

Specifies the password needed to access the database that the profile describes.

-port *db_port*

Specifies the TCP port number used to access the database that the profile describes.

-rman

The options that follow `-rman` specify the details that SnapManager uses to catalog backups with RMAN.

-controlfile

Specifies that SnapManager uses the target database control files instead of a catalog as the RMAN repository.

-login

Starts the RMAN login details.

-password *rman_password*

Specifies the password used to log in to the RMAN catalog.

-username *rman_username*

Specifies the user name used to log in to the RMAN catalog.

-tnsname *tnsname*

Specifies the tnsname connection name (this is defined in the tsname.ora file).

-remove-rman

This option removes RMAN on the profile.

-osaccount *osaccount*

Specifies the name of the Oracle database user account. SnapManager uses this account to perform the Oracle operations such as startup and shutdown. It is typically the user who owns the Oracle software on the host, for example, `oracle`.

-osgroup *osgroup*

Specifies the name of the Oracle database group name associated with the `oracle` account.

Note: The `-osaccount` and `-osgroup` variables are required for UNIX but not allowed for databases running on Windows.

-retain [-hourly [-count *n*] [-duration *m*]] [-daily [-count *n*] [-duration *m*]] [-weekly [-count *n*][-duration *m*]] [-monthly [-count *n*][-duration *m*]]

Specifies the retention class (hourly, daily, weekly, monthly) for a backup.

For each retention class, a retention count or a retention duration or both can be specified. The duration is in units of the class (for example, hours for hourly or days for daily). For instance, if the user specifies only a retention duration of 7 for daily backups, then SnapManager will not limit the number of daily backups for the profile (since the retention count is 0), but SnapManager will automatically delete daily backups created over 7 days ago.

-comment *comment*

Specifies the comment for a profile.

-snapname-pattern *pattern*

Specifies the naming pattern for Snapshot copies. You can also include custom text, for example, "HAOPS" for highly available operations, in all Snapshot copy names. You can change the Snapshot copy naming pattern when you create a profile or after the profile has been created. The updated pattern applies only to Snapshot copies that have not yet occurred. Snapshot copies that exist retain the previous snapname pattern. You can use several variables in the pattern text.

-protect [-protection-policy *policy_name*] | [-noprotect]

Indicates whether the backup should be protected to secondary storage or not.

Note: If `-protect` is specified without `-protection-policy`, then the dataset will not have a protection policy. If `-protect` is specified and `-protection-policy` is not set when the profile is created, then it may be set later by `smo profile update` command or set by the storage administrator through the N series Management Console data protection capability.

The `-noprotect` option specifies not to protect the profile to secondary storage.

-summary-notification

Specifies to enable summary e-mail notification for the existing profile.

-notification [-success -email *e-mail_address1,e-mail address2* -subject *subject_pattern*]

Enables e-mail notification for the existing profile so that e-mails are received by recipients when the SnapManager operation succeeds. You must enter a single e-mail address or multiple e-mail addresses to which e-mail alerts will be sent and an e-mail subject pattern for the existing profile.

You can change the subject text while updating the profile or include custom subject text. The updated subject applies only to the e-mails that are not sent. You can use several variables for the e-mail subject.

-notification [-failure -email *e-mail_address1,e-mail address2* -subject *subject_pattern*]

Enables e-mail notification for the existing profile so that e-mails are received by recipients when the SnapManager operation fails. You must enter a single e-mail address or multiple e-mail addresses to which e-mail alerts will be sent and an e-mail subject pattern for the existing profile.

You can change the subject text while updating the profile or include custom subject text. The updated subject applies only to the e-mails that are not sent. You can use several variables for the e-mail subject.

-separate-archivelog-backups

Separates the archive log backup from datafile backup. This is an optional parameter you can provide while creating the profile. Once the backups are separated using this option, you can either take datafiles-only backup or archive logs-only backup.

-retain-archivelog-backups -hours *hours* | -days *days* | -weeks *weeks*| -months *months*

Specifies to retain the archive log backups based on the archive log retention duration (hourly, daily, weekly, monthly).

-protect [-protection-policy *policy_name*] | -noprotect

Specifies to protect the archive log files based on the archive log protection policy.

Specifies not to protect the archive log files using the `-noprotect` option.

-include-with-online-backups | -no-include-with-online-backups

Specifies to include the archive log backup along with the online database backup.

Specifies not to include the archive log backups along with the online database backup.

-dump

Specifies to collect the dump files after the successful profile create operation.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example changes the login information for the database described by the SALES1 profile and the e-mail notification is configured for this profile. You would run this command after changing the login information on the database itself.

```
smo profile update -profile SALES1 -database -dbname SALESDB
 -sid SALESDB -login -username admin2 -password d4jPe7bw -port 1521
-host server1 -profile-notification -success -e-mail Preston.Davis@org.com -subject
success
Operation Id [8abc01ec0e78ec33010e78ec3b410001] succeeded.
```

Related concepts

How SnapManager determines which backups to retain on local storage on page 103

Related tasks

Changing profile passwords on page 112

The smo profile verify command

This command confirms the profile set up. To verify the profile, the database must be able to be mounted.

Syntax

```
smo profile verify
-profile profile_name
[-quiet | -verbose]
```

Parameters

-profile

Specifies the profile to verify. The profile contains the identifier of the database and other database information.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example verifies the test_rbac_dp_protect profile:

```
 smo profile verify -profile test_rbac_dp_protect  -verbose
[ INFO] SMO-07431: Saving starting state of the database: racb1(OPEN).
[ INFO] SMO-07431: Saving starting state of the database: racb2(SHUTDOWN), racb1(OPEN).
[ INFO] SD-00019: Discovering storage resources for all system devices.
[ INFO] SD-00020: Finished storage discovery for all system devices.
[ INFO] SD-00019: Discovering storage resources for all system devices.
[ INFO] SD-00020: Finished storage discovery for all system devices.
[ INFO] SD-00019: Discovering storage resources for all system devices.
[ INFO] SD-00020: Finished storage discovery for all system devices.
[ INFO] SMO-05070: Database profile TEST_RBAC_DP_PROTECT is eligible for fast restore.
[ INFO] SMO-07433: Returning the database to its initial state: racb2(SHUTDOWN),
racb1(OPEN).
[ INFO] SMO-13048: Profile Verify Operation Status: SUCCESS
[ INFO] SMO-13049: Elapsed Time: 0:04:14.919
Operation Id [Nfffde14ac88cd1a21597c37e8d21fe90] succeeded.
```

Related tasks

Verifying profiles on page 113

The smo repository create command

This command creates a repository in which to store database profiles and associated credentials. This command also checks to see that the block size is adequate.

Syntax

```
smo repository create
-repository
-port repo_port
-dbname repo_service_name
-host repo_host
-login -username repo_username
[-force] [-noprompt]
[-quiet | -verbose]
```

Parameters

-repository

The options that follow *-repository* specify the details of the database for the repository

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-force

Attempts to force the creation of the repository. Using this option results in SnapManager prompting you to backup the repository before creating the repository.

-noprompt

Does not display the prompt to backup the repository before creating it if you use the `-force` option. Using the `-noprompt` option ensures the prompt does not appear, making it easier to create repositories using a script.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Command example

The following example creates a repository in the database SMOREPO on the host hotspur:

```
smo repository create -repository -port 1521 -dbname SMOREPO -host hotspur -login -
username grabal21 -verbose
SMO-09202 [INFO ]: Creating new schema as grabal21 on jdbc:oracle:thin:@//hotspur:1521/
SMOREPO.
SMO-09205 [INFO ]: Schema generation complete.
SMO-09209 [INFO ]: Performing repository version INSERT.
SMO-09210 [INFO ]: Repository created with version: 30
```

```
SMO-13037 [INFO ]: Successfully completed operation: Repository Create
SMO-13049 [INFO ]: Elapsed Time: 0:00:08.844
```

Related tasks

Creating repositories on page 89

The smo repository delete command

This command deletes a repository used to store database profiles and associated credentials. You can delete a repository only if there are no profiles in the repository.

Syntax

```
smo repository delete
-repository
-port repo_port
-dbname repo_service_name
-host repo_host
-login -username repo_username
[-force] [-noprompt]
[-quiet | -verbose]
```

Parameters

-repository

The options that follow `-repository` specify the details of the database for the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-force

Attempts to force the deletion of the repository, even if there are incomplete operations. SnapManager issues a prompt if there are incomplete operations, asking if you are sure you want to delete the repository.

-noprompt

Does not prompt you before deleting the repository. Using the `-noprompt` option ensures the prompt does not appear, making it easier to delete repositories using a script.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Command example

The following example deletes the repository in the SALESDB database:

```
smo repository delete -repository -port 1521 -dbname smorep
-host nila -login -username smofresno -force -verbose
This command will delete repository "smofresno@smorep/nila".
Any resources maintained by the repository must be cleaned up manually.
This may include snapshots, mounted backups, and clones.
Are you sure you wish to proceed (Y/N)?Y
[ INFO] SMO-09201: Dropping existing schema as smofresno
  on jdbc:oracle:thin:@//nila:1521/smorep.
[ INFO] SMO-13048: Repository Delete Operation Status: SUCCESS
[ INFO] SMO-13049: Elapsed Time: 0:00:06.372
[ INFO] SMO-20010: Synchronizing mapping for profiles in
   repository "smofresno@smorep/nila:1521".
[ WARN] SMO-20029: No repository schema exists in "smofresno@smorep/nila:1521".
 Deleting all profile mappings for this repository.
[ INFO] SMO-20012: Deleted stale mapping for profile "TESTPASS".
```

The smo repository rollback command

This command enables you to roll back or revert from a higher version of SnapManager to the original version from which you upgraded.

Syntax

```
smo repository rollback
-repository
-dbname repo_service_name
-host repo_host
-login -username repo_username
-port repo_port
-rollbackhost host_with_target_database
[-force]
[-quiet | -verbose]
```

Parameters

-repository

The options that follow `-repository` specify the details of the database for the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-rollbackhost *host_with_target_database*

Specifies the name of the host which will be rolled back from a higher version of SnapManager to the original lower version.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-force

Attempts to force the update of the repository. SnapManager prompts you to make a backup of the current repository before updating.

-noprompt

Does not display the prompt before updating the repository database. Using the `-noprompt` option ensures the prompt does not appear, making it easier to update repositories using a script.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example updates the repository in the SALESDB database:

```
smo repository rollback -repository -dbname SALESDB
-host server1 -login -username admin -port 1521 -rollbackhost hostA
```

The smo repository rollingupgrade command

This command performs rolling upgrade on a single host or multiple hosts and their associated target databases from a lower version of SnapManager to a higher version. The upgraded host is managed only with the higher version of SnapManager.

Syntax

```
smo repository rollingupgrade
-repository
-dbname repo_service_name
-host repo_host
-login -username repo_username
-port repo_port
-upgradehost host_with_target_database
[-force] [-noprompt]
[-quiet | -verbose]
```

Parameters

-repository

The options that follow `-repository` specify the details of the database for the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-upgradehost *host_with_target_database*

Specifies the name of the host which will be rolling upgraded from a lower version of SnapManager to a higher version.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-force

Attempts to force the update of the repository. SnapManager prompts you to make a backup of the current repository before updating.

-noprompt

Does not display the prompt before updating the repository database. Using the `-noprompt` option ensures the prompt does not appear, making it easier to update repositories using a script.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example updates the repository in the SALESDB database:

```
smo repository rollingupgrade -repository -dbname SALESDB
-host server1 -login -username admin -port 1521 -upgradehost hostA
```

The smo repository show command

This command displays information about the repository.

Syntax

```
smo repository show
-repository
-dbname repo_service_name
-host repo_host
-port repo_port
-login -username repo_username
[-quiet | -verbose]
```

Parameters

-repository

The options that follow `-repository` specify the details of the database for the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Command example

The following example shows details about the repository in the SALESDB database:

```
smo repository show -repository -dbname SALESDB -host server1
-port 1521 -login -username admin
Repository Definition:
User Name: admin
Host Name: server1
Database Name: SALESDB
Database Port: 1521
Version: 28
Hosts that have run operations using this repository: 2
server2
server3
Profiles defined in this repository: 2
GSF5A
GSF3A
Incomplete Operations: 0
```

The smo repository update command

This command updates the repository that stores database profiles and associated credentials when you upgrade SnapManager. Any time you install a new version of SnapManager, you must run the

repository update command before you can use the new version. You are able to use this command only if there are no incomplete commands in the repository.

Syntax

```
smo repository update
-repository
-dbname repo_service_name
-host repo_host
-login -username repo_username
-port repo_port
[-force] [-noprompt]
[-quiet | -verbose]
```

Parameters

-repository

The options that follow `-repository` specify the details of the database for the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. Use either the global name or the SID.

-host *repo_host*

Specifies the name or IP address of the host computer the repository database runs on.

-login

Starts the repository login details.

-username *repo_username*

Specifies the user name needed to access the database that stores the repository.

-port *repo_port*

Specifies the TCP port number used to access the database that stores the repository.

-force

Attempts to force the update of the repository. SnapManager prompts you to make a backup of the current repository before updating.

-noprompt

Does not display the prompt before updating the repository database. Using the `-noprompt` option ensures the prompt does not appear, making it easier to update repositories using a script.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example updates the repository in the SALESDB database:

```
smo repository update -repository -dbname SALESDB
-host server1 -login -username admin -port 1521
```

The smo schedule create command

This command schedules a backup to occur in the specified time and frequency.

Syntax

```
smo schedule create-profile profile_name
[-full{-auto | -online | -offline}
[-retain -hourly | -daily | -weekly | -monthly | -unlimited][-
verify]] |
[-data [[-files files [files]] |
[-tablespaces tablespaces [tablespaces]] {-auto | -online | -offline}
[-retain -hourly | -daily | -weekly | -monthly | -unlimited][-
verify]] |
[-archivelogs]}
[-label label]
[-comment comment]
[-protect | -noprotect | -protectnow][-backup-dest path1 [ , path2]]
[-exclude-dest path1 [ , path2]] [-prunelogs {-all | -until-scn until-
scn | -until -date  yyyy-MM-dd:HH:mm:ss] | -before {-months | -days |
-weeks | -hours}}
-prune-dest prune_dest1,[prune_dest2]]-schedule-name schedule_name[-
schedule-comment schedule_comment]-interval {-hourly | -daily | -
weekly | -monthly | -onetimeonly}-cronstring cron_string-start-time
{start_time <yyyy-MM-dd HH:mm>}-runasuser runasuser[-taskspec
taskspec]-force
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the name of the profile related to the database you want to schedule the back up. The profile contains the identifier of the database and other database information.

-auto

If the database is in a mounted or offline state, SnapManager performs an offline backup. If the database is in an open or online state, SnapManager performs an online backup. If you use the -force option with the -offline option, SnapManager forces an offline backup even if the database is currently online.

-online

Specifies an online database backup.

You can take an online backup of a RAC database, as long as the primary is OPEN, or the primary is MOUNTED and an instance is OPEN. Use -force for online backups if the local instance is SHUTDOWN, or no instance is OPEN. The version of Oracle must be 10.2.0.3 or later or the database will hang if any instance in the RAC is mounted.

- If the local instance is SHUTDOWN and at least one instance is OPEN, using -force changes the local instance to MOUNTED.
- If no instance is OPEN, using -force changes the local instance to OPEN.

-offline

Specifies an offline backup while the database is shut down. If the database is in either the OPEN or MOUNTED state, the backup fails. If the `-force` option is used, SnapManager attempts to alter the database state to shut down the database for an offline backup.

-full

Backs up the entire database. This includes all the data, archived log and control files. The archived redo logs and control files are backed up no matter what type of backup you perform. If you want to back up only a portion of the database, use the `-files` or the `-tablespaces` option.

-files *list*

Backs up only the specified data files plus the archived log and control files. Separate the list of file names with spaces. If the database is OPEN, SnapManager ensures that the appropriate tablespaces are in online backup mode.

-tablespaces *tablespaces*

Backs up only the specified database tablespaces plus the archived log and control files. Separate the tablespace names with spaces. If the database is OPEN, SnapManager ensures that the appropriate tablespaces are in online backup mode.

-label *name*

Specifies an optional name for this backup. This name must be unique within the profile. The name can contain letters, numbers, underscore (_), and hyphen (-). It cannot start with a hyphen.

If you do not specify a label, SnapManager creates a default label that has the format scope_type_date where:

- scope is either F to indicate a full backup or P to indicate a partial backup.
- type is C to indicate an offline (cold) backup, H to indicate an online (hot) backup, or A to indicate auto backup, for example, P_A_20081010060037IST.
- date is the year, month, day, and time of the backup. SnapManager uses a 24-hour clock.

For example, if you performed a full backup with the database offline on Jan. 16, 2007, at 5:45:16 p.m. Eastern standard time, SnapManager would create the label F_C_20070116174516EST.

-comment ***string***

Specifies an optional comment to describe this backup. Enclose the string in single quotes (').

Note: Some shells strip off quote marks. If that is true for your shell, you must escape the quote with a backslash (\). For example, you might need to enter: \' this is a comment\'.

-verify

Verifies that the files in the backup are not corrupt by running the Oracle dbv utility.

Note: If you specify the -verify option, the backup operation does not complete until the verify operation completes.

-force

Forces a state change if the database is not in the correct state. For example, SnapManager might change the state of the database from online to offline, based on the type of backup you specify and the state that the database is in.

With an online RAC database backup, use -force if the local instance is SHUTDOWN, or no instance is OPEN.

Note: The version of Oracle must be 10.2.0.3 or later or the database will hang if any instance in the RAC is mounted.

- If the local instance is SHUTDOWN and at least one instance is OPEN, then using -force changes the local instance to MOUNTED.
- If no instance is OPEN, using -force changes the local instance to OPEN.

-protect | **-noprotect** | **-protectnow**

Indicates whether the backup should be protected to secondary storage. The `-noprotect` option specifies that the backup should not be protected to secondary storage. Only full backups are protected. If neither option is specified, SnapManager protects the backup as the default, if the backup is a full backup and

the profile specifies a protection policy. The `-protectnow` specifies to protect the backup immediately to secondary storage.

-retain { -hourly | -daily | -weekly | -monthly | -unlimited}

Specifies whether the backup should be retained on an hourly, daily, weekly, monthly, or unlimited basis. If `-retain` is not specified, the retention class defaults to `-hourly`. To retain backups forever, use the `-unlimited` option. The `-unlimited` option makes the backup ineligible for deletion by the retention policy.

-archivelogs

Specifies to create archive log backup.

-backup-dest *path1*, [*, [path2]*]

Specifies the archive log destinations to be backed up for archive log backup.

-exclude-dest *path1*, [*, [path2]*]

Specifies the archive log destinations to be excluded from the backup.

-prunelogs {-all | -until-scn *until-scn* | -until-date *yyyy-MM-dd:HH:mm:ss* | -before {-months | -days | -weeks | -hours}

Specifies whether to delete the archive log files from the archive log destinations based on options provided while creating a backup. The `-all` option deletes all the archive log files from the archive log destinations. The `-until-scn` option deletes the archive log files until a specified SCN. The `-until-date` option deletes the archive log files until the specified time period. The `-before` option deletes the archive log files before the specified time period (days, months, weeks, hours).

-schedule-name *schedule_name*

Specifies the name that you provide for the schedule.

-schedule-comment *schedule_comment*

Specifies an optional comment to describe about scheduling the backup.

-interval { -hourly | -daily | -weekly | -monthly | -onetimeonly}

Indicates the time interval by which the backups are created. You can schedule the backup on an hourly, daily, weekly, monthly, or one time only.

-cronstring *cron_string*

Specifies to schedule the backup using cronstring. Cron expressions are used to configure instances of CronTrigger. Cron expressions are strings that are actually made up of seven sub-expressions:

- 1 refers to seconds
- 2 refers to minutes
- 3 refers to hours

- 4 refers to a day in a month
- 5 refers to the month
- 6 refers to a day in a week
- 7 refers to the year (optional)

-start-time ***yyyy-MM-dd HH:mm***

Specifies the start time of the schedule operation. The schedule start time should be included in the format of yyyy-MM-dd HH:mm.

-runasuser ***runasuser***

Specifies to change the user (root user or Oracle user) of the scheduled backup operation while scheduling the backup.

-taskspec ***taskspec***

Specifies the task specification XML file that can be used for pre-processing activity or post-processing activity of the backup operation. The complete path of the XML file should be provided which give the `-taskspec` option.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

The smo schedule delete command

This command deletes a backup schedule when it is are no longer necessary.

Syntax

```
smo schedule delete-profile profile_name
-schedule-name schedule_name[-quiet | -verbose]
```

Parameters

-profile ***profile_name***

Specifies the name of the profile related to the database you want to delete a backup schedule. The profile contains the identifier of the database and other database information.

-schedule-name ***schedule_name***

Specifies the schedule name you provided while creating a backup schedule.

The smo schedule list command

This command lists the scheduled operations associated with a profile.

Syntax

```
smo schedule list-profile profile_name
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the name of the profile related to the database, using which you can view a list of scheduled operations. The profile contains the identifier of the database and other database information.

The smo schedule resume command

This command resumes the suspended backup schedule.

Syntax

```
smo schedule resume-profile profile_name
-schedule-name schedule_name[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the name of the profile related to the database you want to resume the suspended backup schedule. The profile contains the identifier of the database and other database information.

-schedule-name *schedule_name*

Specifies the schedule name you provided while creating a backup schedule.

The smo schedule suspend command

This command suspends a backup schedule until the backup schedule is resumed.

Syntax

```
smo schedule suspend-profile profile_name
-schedule-name schedule_name[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the name of the profile related to the database you want to suspend a backup schedule. The profile contains the identifier of the database and other database information.

-schedule-name *schedule_name*

Specifies the schedule name you provided while creating a backup schedule.

The smo schedule update command

This command updates the schedule for a backup.

Syntax

```
smo schedule update-profile profile_name
-schedule-name schedule_name[-schedule-comment schedule_comment]-
interval {-hourly | -daily | -weekly | -monthly | -onetimeonly}-
cronstring cron_string    -start-time {start_time <yyyy-MM-dd HH:mm>}-
runasuser runasuser[-taskspec taskspec]-force
[-quiet | -verbose]
```

Parameters

-profile *profile_name*

Specifies the name of the profile related to the database you want to schedule the back up. The profile contains the identifier of the database and other database information.

-schedule-name *schedule_name*

Specifies the name that you provide for the schedule.

-schedule-comment *schedule_comment*

Specifies an optional comment to describe about scheduling the backup.

-interval { -hourly | -daily | -weekly | -monthly | -onetimeonly}

Indicates the time interval by which the backups are created. You can schedule the backup on an hourly, daily, weekly, monthly, or one time only.

-cronstring *cron_string*

Specifies to schedule the backup using cronstring. Cron expressions are used to configure instances of CronTrigger. Cron expressions are strings that are actually made up of seven sub-expressions:

- 1 refers to seconds
- 2 refers to minutes
- 3 refers to hours
- 4 refers to a day in a month
- 5 refers to the month
- 6 refers to a day in a week
- 7 refers to the year (optional)

-start-time *yyyy-MM-dd HH:mm*

Specifies the start time of the schedule operation. The schedule start time should be included in the format of yyyy-MM-dd HH:mm.

-runasuser *runasuser*

Specifies to change the user of the scheduled backup operation while scheduling the backup.

-taskspec *taskspec*

Specifies the task specification XML file that can be used for pre-processing activity or post-processing activity of the backup operation. The complete path of the XML file should be provided which give the `-taskspec` option.

The smo storage rename command

This command updates the name or IP address of the storage system.

Syntax

```
smo storage rename
-profile profile-oldname old_storage_name  -newname new_storage_name[-
quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

-oldname ***old_storage_name***

Specifies the IP address or name of the storage system before storage renaming.

-newname ***new_storage_name***

Specifies the IP address or name of the storage system after storage renaming.

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example command

The following example updates the name of the storage system:

```
smo storage rename -profile mjullian -oldname lech -newname hudson -
verbose
```

The smo storage list command

This command displays the list of storage controllers associated with a particular profile.

Syntax

```
smo storage list
-profile profile
```

Parameters

-profile ***profile***

Specifies the name of the profile. This name can be up to 30 characters long and must be unique within the host.

Example command

The following example displays the storage controllers associated with the profile mjullian:

```
smo storage list -profile mjullian


Sample Output:
Storage Controllers
```

```
--------------------
N5200-RTP07OLD
```

The smo system dump command

This command creates a jar file that contains diagnostic information about the server environment.

Syntax

```
smo system dump
[-quiet | -verbose]
```

Parameters

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example of the system dump command

The following example uses the command:

```
smo system dump
Path:/userhomedirectory/.ontap/smo/2.2.01.1/smo_dump_hostname.jar
```

The smo system verify command

This command confirms that all the components of the environment required to run SnapManager are set up correctly.

Syntax

```
smo system verify
[-quiet | -verbose]
```

Parameters

-quiet

Displays only error messages on the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages on the console.

Example of the system verify command

The following example uses the `smo system verify` command:

```
smo system verify
SMO-13505 [INFO ]: Snapdrive verify passed.
SMO-13037 [INFO ]: Successfully completed operation: System Verify
SMO-13049 [INFO ]: Elapsed Time: 0:00:00.559
Operation Id [N4f4e910004b36cfecee74c710de02e44] succeeded.
```

The smo version command

This command lets you determine which version of SnapManager you are running on your local host.

Syntax

```
smo version
[-quiet | -verbose]
```

Parameters

`-quiet`

Displays only error messages on the console. The default is to display error and warning messages.

`-verbose`

Displays the build date and the contents of each profile. Also displays error, warning, and informational messages on the console.

Example of the version command

The following example displays the installed version of SnapManager:

```
smo version
SnapManager for Oracle Version: 3.1
```

Troubleshooting SnapManager for Oracle

Reviewing some of the most common issues that might occur could help you resolve them.

The following table describes common issues and possible solutions:

Issue-driven question	Possible Solution
Are the target database and the listener running?	To check this, run the command `lsnrctl status`. Ensure that the database instance is registered with the listener.
Is the storage visible?	To verify that the storage is visible, use the command `snapdrive storage show -all`.
Is the storage writable?	Make sure that you can write to the storage system by attempting to touch a file within the mountpoint you just created. Use the command `touch filename`. If the file is created, then you've verified that the storage is writable. You want to verify that storage is writable by the user that SnapManager runs as (for example, as 'root' on UNIX).
Is the SnapManager server running?	On Windows, check the status, and start the server via the Service Configuration. Before you can use the graphical user interface or the command-line interface to initiate SnapManager commands related to profiles, the server must be running. You can create or update repositories without starting the sever, but to execute any other SnapManager operations, the server must be running. If not already running, start the SnapManager server by entering this command: `smo_server start`.
Are all the components required to run SnapManager set up correctly?	Run the `smo system verify` command to make sure that SnapDrive is set up correctly for SnapManager to work correctly.
Do you have the correct version of SnapManager?	Use the `smo version` command to check your version of SnapManager.

Issue-driven question	Possible Solution
Have you looked at the SnapManager log files to determine if the error messages can help to isolate the issue?	SnapManager records all log entries and places them into one set of rotating log files. The log files are found in `/var/log/smo` It might also be helpful to look at the logs in this location: `/usr_home/.ontap/smo/3.1/log/` Each operation goes into its own log file of the form smo_of_date_time.log

If you have archive logs stored on a storage system that is not running Data ONTAP, have you excluded them from consideration for backup with SnapManager?	The smo.config file enables you to exclude certain archive log files.
	For Solaris, the file is located in:
	`/opt/Ontap/smo/properties/ smo.config`
	For all other platforms, the files are located in:
	`/opt/Ontap/smo/properties/smo.config`
	Follow the format suggested within the file to exclude the local archive logs.

Do you have a FlexClone license if you are using SnapManager with NFS databases?	A FlexClone license is required to take full advantage of SnapManager with NFS databases. SnapManager uses the FlexClone feature to accomplish these tasks: • Mount backups of NFS databases • Verify backups of NFS databases • Clone NFS databases • Register backups of NFS databases with RMAN (if using RMAN)
Do you have an MS-DOS window open in the directory where you are attempting to install or upgrade SnapManager on Windows?	You will see an error message similar to this: `Directory C:\Program Files\Ontap\SnapManager for Oracle\bin is currently in use by another program. Any window, opened by you or another user, that is currently referencing this directory must be closed before installation can proceed.` Close the window and attempt the installation or upgrade again.

Issue-driven question	Possible Solution
Were you unable to connect to the repository?	If connecting to a repository fails, run `lsnrctl status` on the repository database and check the active service names. When SnapManager connects to the repository database, it uses the service name of the database. Depending on how the listener is setup, this may be the short service name or the fully qualified service name. When SnapManager connects to a database for a backup, restore or other operation, it uses the host name and the SID. If the repository does not initialize correctly because it is currently unreachable, you receive an error message asking if you would like to remove the repository. This allows you to remove the repository from your current view so you can perform operations on other repositories. Also check if the repository instance is running or not with the command `ps -eaf |grep <instance -name >` on UNIX and check if the corresponding service is running on Windows.
The system cannot resolve the host name?	Check if the specified host name is on a different subnet. If you receive an error message that SnapManager is unable to resolve the host name, then add the host name in the host file. • For Windows host, add the host name to the file located at the file path, `C:\WINDOWS \system32\drivers\etc\hosts`, as follows: **`xxx.xxx.xxx.xxx hostname <IP address>`**
Is SnapDrive running?	Check to see if the SnapDrive daemon is running. Enter this command: `-snapdrived status` If the daemon is not running, a message appears indicating that there is a connection error.
Which storage systems are configured to be accessed with SnapDrive?	Enter this command: `-snapdrive config list`

Issue-driven question	Possible Solution
How to improve SnapManager GUI performance?	• Ensure that you have valid user credentials for the repository, profile host, and profile. If your credential is invalid, then clear the user credentials for the repository, profile host, and profile. Reset the same user credentials that you have set before for the repository, profile host, and the profile. For additional information on setting the user credentials again, refer to "Setting credentials after clearing credential cache". • Close the unused profiles. If the number of profiles that you have opened is more, the SnapManager GUI performance slows down.

Dump files

SnapManager can create compressed log files containing information about SnapManager and its environment. You can create an operation, profile, or system dump file.

Using the `dump` commands or the **Create Diagnostics** option in the graphical user interface allows you to capture information about an operation, a profile, or the environment. A system dump does not require a profile; however, the profile and operation dumps do require profiles.

SnapManager includes the following diagnostic information in the file:

- The steps performed
- How long they took
- What the outcome of each step was
- Any messages that occurred during the operation

Note: SnapManager log files or dump files enable read and write permissions only for the root users and the other users who belong to root user group.

SnapManager also includes the following information in the file:

- Operating system version and architecture
- Environment variables
- Java version
- SnapManager version and architecture
- SnapManager preferences
- SnapManager messages
- log4j properties

- SnapDrive version and architecture
- SnapDrive log files
- Oracle version
- Storage system version for the storage system
- Oracle `oratab` file
- Oracle listener status
- Oracle network configuration files (listener.ora and tnsnames.ora)
- Repository database Oracle version
- Target database type (stand-alone or RAC)
- Target database role (primary, physical standby, or logical standby)
- Target database RMAN setup (no RMAN integration, RMAN with control files, or RMAN with catalog file)
- Target database ASM instance version
- Profile descriptor
- Shared memory maximum
- Swap space information
- Memory information
- Kernel parameters
- FSTAB
- Protocol (iSCSI, FC, or NFS)
- Multipath environment
- RAC
- Supported volume manager
- Supported file system
- Host utilities version
- Microsoft iSCSI software initiator version for Windows

Even if the SnapManager host server is not running, the dump information is still available by using commands or the graphical user interface.

Note: SnapManager dump files also contain the SnapDrive data collector file.

If you encounter a problem that you cannot resolve, you can send these files to technical support for help.

Creating operation-level dump files

To get log information about a particular operation, use the `smo operation dump` command with the name or ID of the failed operation. Using various dump commands, you can gather information about a specific operation, profile, host, or environment.

Step

1. To get information about a particular failed operation, enter this command:

smo operation dump -id *guid*

Note: The `smo operation dump` command provides a super-set of the information provided by the `smo profile dump` command, which in turn provides a super-set of the information provided by the `smo system dump` command.

Dump file location:

```
Path:/<user-home>
/.ontap/smo/3.1.0/smo_dump_8abc01c814649ebd0114649ec69d0001.jar
```

Creating profile-level dump files

To get log information about a particular profile, use the `smo profile dump` command with the name of the profile.

Step

1. To get information about a particular profile, enter this command:

   smo profile dump -profile *profile_name*

   Dump file location:

   ```
   Path:/<user-home>
   /.ontap/smo/3.1.0/smo_dump_8abc01c814649ebd0114649ec69d0001.jar
   ```

Creating system-level dump files

To get log information about the SnapManager host and environment, use the smo system dump command. No options are necessary. Using various dump commands, you can gather information about a specific operation, profile, or host and environment. The system-level dump command gathers a stack trace of SnapManager server operations currently running.

Step

1. To gather log information about the SnapManager host, enter this command:

   smo system dump

   Resulting dump

   ```
   Path:/<user-home>/.ontap/smo/3.1.0/smo_dump_server_host.jar
   ```

How to locate dump files

SnapManager gathers information about the operation and your environment and places it in the dump result file.

Each of the dump commands creates an unified client/server dump file containing information about the profile, failed operations, and system operation. The dump file is located at the client system for easy access. The command returns the paths to the dump file. These files can be helpful if you need to troubleshoot a problem with the profile or operation.

The resulting file is located in the user's home directory on the client, according to the following:

- If using the graphical user interface:

```
user_home/Application Data/Ontap/smo/3.1.0/smo_dump dump_file_type_name
server_host.jar
```

- If using the command-line interface:

```
user_home/.ontap/smo/3.1.0/smo_dump_dump_file_type_name server_host.jar
```

The resulting dump file contains the output of the dump request. The name of the file depends on the information supplied. There is one jar file that contains both the client and the server information.

The following table shows the types of dump operations and the resulting file names:

Type of dump operation	Resulting file name
Operation dump command with operation ID	`smo_dump_`*`operation-id`*`.jar`

Operation dump command with no operation ID (GUID or a label)	`smo operation dump -profile` *`VH1`* `-verbose`
	The following output appears:
	`smo operation dump -profile VH1 -verbose`
	`[ INFO] SMO-13048: Dump Operation Status: SUCCESS`
	`[ INFO] SMO-13049: Elapsed Time: 0:00:01.404`
	`Dump file created.`
	`Path: /oracle/VH1/<path>/smo/3.1.0/`
	`smo_dump_VH1_kaw.rtp.foo.com.jar`

System dump command	`smo_dump_`*`host-name`*`.jar`
Profile dump command	`smo_dump_`*`profile-name_host-name`*`.jar`

How to collect dump files

SnapManager enables you to collect the dump files immediately after a SnapManager operation irrespective of whether the operation is a successful or failed one.

To collect the dump files after specific SnapManager operation, you need to add the `-dump` option at the end of the SnapManager command. The `-dump` option is an optional parameter.

You can collect dump files for the following SnapManager operations:

- Create profiles
- Update profiles
- Create backups
- Verify backups
- Delete backups
- Free backups
- Mount backups
- Unmount backups
- Restore backups

- Create clones
- Delete clones
- Split clones

Note: For creating profile, you can collect dump files only when the operation is successful. If you encounter an error while creating a profile, use the `smo system dump` command. For successful profiles, use the `smo operation dump` and `smo profile dump` commands.

Example

```
smo  backup create -profile targetdb1_prof1 -auto -full -online -dump
```

Collecting additional log information for easier debugging

When any SnapManager operation fails, if you would require any additional logs to be added to the existing log files to debug the issue, you have to set an external environment variable (`server.log.level`). This variable overrides the default log level from debug to trace and dumps all the log messages in the log file.

To set the external environment variable, perform the following steps:

1. Create a `platform.override` text file in the SnapManager installation directory.
2. Add a `server.log.level` key in the `platform.override` text file.
3. Assign the value as `TRACE` to the `server.log.level` key:

   `server.log.level=TRACE`

4. Restart the SnapManager server.

After restarting the SnapManager server, SnapManager dumps additional debug information into the logs each time when any SnapManager operation is performed.

Note: If the additional log information is not required, you can delete the value `TRACE` or the entire `server.log.level` key from the `platform.override` text file.

Troubleshooting clone issues

Reviewing some of the clone issues that might occur could help you resolve them.

Symptom	Explanation	What to do
The clone operation fails with a message saying that the mount path you are using is already in use.	SnapManager does not let you mount a clone over an existing mountpoint. This means it is possible that an incomplete clone did not remove the mountpoint.	Specify a different mountpoint to be used by the clone, or unmount the problem mountpoint.

Symptom	Explanation	What to do
The clone operation fails with an error message about data files not having a .dbf extension.	Some versions of the Oracle NID utility do not work with data files unless the files use a .dbf extension.	If you encounter an error during a clone operation, and you have a data file that does not end in a .dbf extension: • Rename the data file to give it a .dbf extension. • Repeat the backup operation. • Clone the new backup.
The clone operation fails due to unmet requirements.	You are attempting to create a clone; however, some of the prerequisites have not been met.	If you encounter an error during a clone operation and there are unmet requirements, proceed as described in "Creating a clone" to meet the prerequisites.
SnapManager fails to generate a new profile after the clone split operation and the user does not know if the new profile is created.	SnapManager fails to prompt the user if a new profile is not created after the clone split operation. Since the failed operation is not prompted, there is a possibility that the user assume that the profile is created.	From the SnapManager CLI, enter the `clone split-result` command to view the detailed result of the clone split process.
SnapManager for Oracle fails to clone Oracle 10gR2 (10.2.0.5) physical Oracle Dataguard Standby databases.	SnapManager for Oracle does not disable the managed recovery mode while performing an offline backup of the Oracle 10gR2 (10.2.0.5) physical standby databases created using Oracle Data Guard services. Due to this issue, the offline backup taken is inconsistent. When SnapManager for Oracle tries to clone the offline backup, it does not even try to perform any recovery on the cloned database. Since the backup is inconsistent, the cloned database requires required recovery, and thus Oracle fails to create the clone successfully.	Upgrade the Oracle database to the Oracle 11gR1 (11.1.0.7 patchset).

Troubleshooting graphical user interface issues

Reviewing the more common known graphical user interface issues might help you resolve the issue.

Symptom	Explanation	What to do
When you restart the graphical user interface and try to check the backups for a certain profile, it looks as if all the backups have disappeared. All you see are the names of the profiles.	Each time you start a graphical user interface session, you must open the repository and any profiles. SnapManager does not display any information about a profile until you open it.	Right-click the profile and select the **Open** option from the menu. SnapManager displays the Profile Authentication dialog box. Enter the host user name and password. Once you do this, SnapManager displays the backup list. You only need to authenticate the profile once as long as the credentials are valid and remain in cache.
Graphical user interface installation on Windows succeeds with errors.	A Windows user installs the graphical user interface. The installation succeeds, but you get an error message. Furthermore, the user does not have a desktop icon to start the graphical user interface In this scenario, the user account used to install the graphical user interface has enough permissions to install the graphical user interface, but not enough permissions to setup the icons and shortcuts for all users of the PC. The user account must be able to modify directory: `C:\Documents and Settings\All Users`	Perform one of the following: • Reinstall the graphical user interface with different settings. On the “Choose Icon Availability” step, deselect “Make these shortcuts available to all users of this PC.” This time the installation should succeed completely. • Login using a user account that is not restricted and reinstall the graphical user interface.

Symptom	Explanation	What to do
When you open the first repository in the graphical user interface, it displays all the profiles. You receive an error message similar to: `The Profile name XXXX clashes with previously loaded repository.`	Identically named profiles cannot exist in a repository. Also, you can open only one repository at a time.	Reference the conflicting profiles from two different OS users or rename the profile by issuing a SQL statement against the repository: `UPDATE SMO_30_PROFILE SET NAME = 'NEW_NAME' WHERE NAME = 'OLD_NAME'`

You receive an error message similar to:
`SMO-01092: Unable to initialize repository repo1@ does not exist:repo1`
`SMO-11006: Cannot resolve host does not exist`
	The repository is inaccessible, perhaps because it no longer exists. The graphical user interface initializes the list of repositories from the credentials file.	The error message asks if you would like to remove this repository so that no attempt is made to load it in the future. If you do not need to access this repository, click **Delete** to remove it from the graphical user interface view. This removes the reference to the repository in the credentials file and the graphical user interface will not attempt to load the repository again.

Symptom	Explanation	What to do

Profile creation fails as host credentials fails to authenticate in the SUSE Linux Enterprise Server 10 and SUSE Linux Enterprise Server 11 platforms.	SnapManager uses PAM security module to authenticate users. In the SUSE Linux Enterprise Server versions 10 and 11 platforms, there is no `snapmanager` file by default in the `/etc/pam.d` directory that provides the required authentication details. Hence, SnapManager consults other files in the `/etc/pam.d/` location and results in the host credentials failure.	To successfully login to the host in the SUSE Linux Enterprise Server 10 and 11 platforms, perform the following steps:
		1. Create a file `snapmanager` in the file path `/etc/pam.d/`.
		2. Add the following content in the `snapmanager` file located at `/etc/pam.d/snapmanager`:
		`#%PAM-1.0`
		`auth include common-auth`
		`account include common-account`
		`password include common-password`
		`session include common-session`
		3. Save the file and retry the profile creation operation. This will resolve the host credential authentication problem.

SnapManager takes a longer time to load the database tree structure and results in a timeout error from the SnapManager GUI.	When you try to perform a partial backup operation from the SnapManager GUI, SnapManager tries to load the credential details for all the profiles, and if there are any invalid entries, SnapManager tries to validate the entry and this makes SnapManager take a longer time and thus result in a timeout error.	To avoid the timeout error while performing a database operation, perform the following task: Delete the credentials of the unused host, repository, and profile using the `credential delete` command from the SnapManager CLI.

Symptom	Explanation	What to do
SnapManager fails to generate a new profile after the clone split operation and the user does not know if the new profile is created.	SnapManager fails to prompt the user if a new profile is not created after the clone split operation. Since the failed operation is not prompted, there is a possibility that the user assume that the profile is created.	To know if a new profile is created for the clone split operation: 1. Select the **Monitor** tab, right-click the clone split operation entry, and select the **Properties** option from the menu. 2. On the **Profile Properties** window, select the **Logs** tab to view the clone split operation and profile creation logs.
The custom scripts for the pre-processing or post-processing activity to occur before or after the backup, restore, or clone operations, are not visible from the SnapManager GUI	When you add custom scripts in the custom backup, restore, or clone script location after you launch the respective wizard, the custom scripts are not displayed under the Available Scripts list.	To utilize the newly-added custom scripts for the pre-processing or post-processing activity to occur, you must restart the SnapManager host server and then launch the SnapManager GUI.
Unable to use earlier clone specification XML file in the SnapManager 3.2 for Oracle for clone operation	From SnapManager 3.2 for Oracle, the task specification section (`<task-specification>`) is provided as a separate task specification XML file.	If you are using the earlier clone specification XML file in the SnapManager 3.2 for Oracle, you must remove the task specification section (`<task-specification>`) from the clone specification XML or create a new clone specification XML file.

Symptom	Explanation	What to do
Unable to proceed SnapManager operation in SnapManager GUI after you have cleared user credentials using the `smo credential clear` command from the SnapManager CLI or by selecting **Admin** > **Credentials** > **Clear Cache** from the SnapManager GUI. When user credentials become invalid, SnapManager tries the validate the credentials for a long time and finally fails to respond. When a host or a profile is deleted from the repository, the user credentials are still available in the cache. These unnecessary credentials entries slows down the SnapManager operations from the SnapManager GUI.	The credentials set for the repositories, hosts, and profiles are cleared. SnapManager verifies user credentials before starting any SnapManager operation.	You have to restart the SnapManager GUI depending on how the cache is cleared. **Note:** • If you have cleared the credential cache from the SnapManager GUI, you do not need to exit the SnapManager GUI. • If you have cleared the credential cache from the SnapManager CLI, you must restart SnapManager GUI. • If you have deleted the encrypted credential file manually, you must restart the SnapManager GUI again. Set the credentials that you have given before for the repository, profile host, and the profile. From the SnapManager GUI, if there is no repository mapped under the **Repositories** tree: • Add an existing repository using the **Add Existing Repository** option under the **Tasks** section or menu. • Right-click the repository, select **Open**, enter the user credentials in the **Repository Credentials Authentication** window. • Right-click the host under the repository, select **Open**, enter the user credentials in the **Host Credentials Authentication** window. • Right-click the profile under the host, select **Open**, enter the user credentials in the **Profile Credentials Authentication** window.

Symptom	Explanation	What to do
The error message `Unable to list the protection policies for the following reason: Protection Manager is temporarily unavailable` is displayed when **None** is clicked in the **Protection Manager Protection Policy** drop down menu of the **Profile Properites** window and the policy settings page of **Profile create** wizard.	The Protection Manager is not configured with SnapManager or the Protection Manager server is not running.	No action is necessary.
Unable to launch SnapManager graphical user interface using Java Web Start GUI due to weaker secure socket layer (SSL) cipher strength of the browser.	SnapManager does not support weaker SSL ciphers less than 128 bits.	Upgrade the browser version and check the cipher strength.

Troubleshooting SnapDrive issues

There are a few common issues you might run into when using SnapManager with SnapDrive products.

First, you must determine if the issue is related to SnapManager for Oracle or SnapDrive. If the issue is a SnapDrive error, SnapManager for Oracle gives an error message similar to:

```
SMO-12111: Error executing snapdrive command "<snapdrive command>": <snapdrive error>
```

The following is an example of a SnapDrive error message where `SMO-12111` is the SnapManager error number. The `0001-770` numbering scheme represents SnapDrive for UNIX errors.

```
SMO-12111: Error executing snapdrive command
"/usr/sbin/snapdrive snap restore -file
/mnt/pathname/ar_anzio_name_10gR2_arrac1/data/undotbs02.dbf
-snapname pathname.company.com:
/vol/ar_anzio_name_10gR2_arrac1:
TEST_ARRAC1_YORKTOW_arrac12_F_C_0_8abc01b20f9ec03d010f9ec06bee0001_0": 0001-770
Admin error: Inconsistent number of files returned when listing contents of
/vol/ar_anzio_name_10gR2_arrac1/.snapshot/
```

```
TEST_ARRAC1_YORKTOW_arrac12_F_C_0_8abc01b20f9ec03d010f9ec06bee0001_0/data
on filer pathname.
```

The following are the most common SnapDrive for UNIX error messages related to LUN discovery, configuration issues, and space. If you receive any of these errors, see the Troubleshooting chapter of the *SnapDrive Installation and Administration Guide*.

Symptom	Explanation
`0001-136 Admin error: Unable to log on to filer: <filer> Please set user name and/or password for <filer>`	Initial SnapDrive configuration
`0001-382 Admin error: Multipathing rescan failed`	LUN discovery error
`0001-462 Admin error: Failed to unconfigure multipathing for <LUN>: spd5: cannot stop device. Device busy.`	LUN discovery error
`0001-476 Admin error: Unable to discover the device associated with ... 0001-710 Admin error: OS refresh of LUN failed ...`	LUN discovery error
`0001-680 Admin error: Host OS requires an update to internal data to allow LUN creation or connection. Use 'snapdrive config prepare luns' or update this information manually...`	LUN discovery error
`0001-817 Admin error: Failed to create volume clone ... : FlexClone not licensed`	Initial SnapDrive configuration
`0001-817 Admin error: Failed to create volume clone ... : Request failed as space cannot be guaranteed for the clone.`	
Space issue `0001-878 Admin error: HBA assistant not found. Commands involving LUNs should fail.`	LUN discovery error

Troubleshooting storage system name issues

After you rename the storage system, you might encounter errors such as `cannot perform operation` while creating backups using the new storage system name. The issue occurs when SnapManager fails to recognize the new storage system.

Reviewing the following steps could help you resolve the issue.

Perform the following steps as a root user from the SnapManager host server CLI:

1. Delete the earlier storage system name using the command:

   `snapdrive config delete filername` ***`filername`***

Note: If you do not delete the earlier storage system name, then all theSnapManager operations fail.

2. Delete the IP address and host of the earlier storage system in the host file located at the file path: `etc/hosts`.
3. Add a new storage system name using the command:

   `snapdrive config set` ***`user_name`*** **`filername`** ***`filername`***
4. Map the earlier and later storage system names using the command:

   `snapdrive config migrate set filername` ***`old_storage_system_name new_storage_system_name`***
5. Add the IP address and host of the new storage system in the host file located at the file path: `etc/hosts`.
6. Update the profile for the new storage system name using the command:

   `smo storage rename -profile` ***`profile-name`*** **`-oldname`** ***`oldfilername`*** **`-newname`** ***`newfilername`***
7. Perform the SnapManager operation either using the old backup created before renaming the storage system name or create a new backup using the later storage system.
8. Verify the storage system associated with the profile using the command:

   `smo storage list -profile` ***`profile-name`***

Troubleshooting known issues

There are some known issues that might occur when you are using SnapManager. Reviewing these could help you resolve the issue.

The server fails to start

When starting the server, you might see an error message similar to this:

```
SMO-01104: Error invoking command: SMO-17107: SnapManager Server failed to
start on port 8074 because of the following errors: java.net.BindException:
Address already in use
```

The most likely cause of this problem is that the SnapManager listening ports (27214 and 27215, by default) are currently in use. In the log file, the text
`java.net.BindException: Address already in use`
is displayed. This error occurs if another application is currently using the ports.

What to do

In this case, reconfigure either SnapManager for Oracle or the other application to use different ports. This error can also occur if the `smo_server` command is already running, but SnapManager for Oracle did not detect the existing process.

To reconfigure SnapManager, edit the following file: `/opt/Ontap/smo/properties/smo.config`

Solaris:

```
/opt/Ontap/smo/properties/smo.config
```

For all other platforms:

```
/opt/Ontap/smo/properties/smo.config
```

Locate the following lines and change the listening port numbers to unique values in your environment:

1. SMO Server.port=27214
2. SMO Server.rmiRegistry.port=27215
3. remote.registry.ocijdbc.port= 27215
   The remote.registry.ocijdbc.port must be the same as the Server.rmiRegistry.port.

Then run the following sequence of commands to restart the SnapManager server:

1. To start the server, enter: smo_server start
   You will get an error message if the server is already running.
2. To stop the server, enter: smo_server stop
3. To restart the server, enter: smo_server start

To start the SnapManager server on a Windows platform, follow these steps:

1. Go to **Start > Control Panel > Administrative Tools > Services**.
2. You can start the server in one of three ways:
   - In the left panel, click **Start**.
   - Right-click Ontap SnapManager 3.2 for Oracle and select **Start** from the drop-down menu.
   - Double-click Ontap SnapManager 3.2 for Oracle and in the Properties window that opens, click **Start**.

Terminating a currently running SnapManager operation

If SnapManager freezes and you cannot execute any operations successfully, you can terminate SnapManager and its operations.

Because SnapManager works with SnapDrive and Protection Manager, consider operations in both of these applications. If one of the operations involves restoring a protected backup from secondary storage, look for this operation in Protection Manager, as well.

1. In UNIX, execute the following command to list all SnapDrive processes that are running: `ps`
   Example: `ps | grep snapdrive`
2. End the offending SnapDrive process or processes by entering this command `kill <pid>` in UNIX.
   where <pid> is the list of processes you found using the ps command.
   Do not stop all SnapDrive processes. You might want to end just the last process that is running.
3. If one of the operations involves restoring a protected backup from secondary storage, open the Protection Manager graphical user interface and do the following:
   a. From the System menu, select **Jobs**.
   b. In the type of job, select **Restore**.
   c. Check for the name of the dataset that matches the one in the SnapManager profile.
   d. Right-click and select **Cancel**.
4. List the SnapManager processes by doing the following:
   a. Log in as root.
   b. List the processes with the `ps` command.
      Example: `ps | grep java`
5. End the offending SnapManager process by doing the following:
   - Record the time that the offending process started.
   - Stop that one process with the `kill <pid>` command.

Deleting or freeing the last protected backup

When a DBA creates the first backup for a profile on secondary storage, SnapManager sends all information about the backup to Protection Manager. For subsequent backups related to this profile, SnapManager sends only the modified information. If you removed the last protected backup, SnapManager would lose the ability to discern the differences between backups and SnapManager would have to find a way to "rebaseline" these relationships. Therefore, attempting to delete the last protected backup would result in an error.

However, there might be a time when you need to delete or free the last protected backup. The method you use depends on whether you need to delete the profile or just the profile backup.

To delete the profile:

1. Delete the profile's backups.
2. Update the profile and disable protection in the profile.
   This deletes the dataset.
3. Delete the last protected backup.
4. Delete the profile.

To delete just the backup:

1. Take another backup for the profile.
2. Transfer that backup to secondary storage.
3. Delete the previous backup.

How to handle archive log file destination names if one or more destination names are part of other destination names

While creating an archive log backup, if user excludes a destination which is part of other destination names then the other destination names are also excluded.

For example, if there are three destinations available to be excluded, `/dest`, `/dest1`, and `/dest2`.

When you provide `/dest` to be excluded while creating the archive log file backup from the SnapManager CLI:

```
smo backup create -profile almsamp1 -data -online -archivelogs  -exclude-
dest /dest
```

SnapManager for Oracle excludes all the archive log files starting with `/dest`.

What to do

You can perform any one of these tasks:

- Add a path separator (/ or \) at the end of destinations configured in `v$archive_dest`. For example, change the `/dest` as `/dest/`.
- While creating a backup, include other destinations instead of excluding the destination.

Mounting a FlexClone volume fails in NFS environment

When SnapManager creates a FlexClone of a volume in NFS environment, an entry is added in the `/etc/exports` file. The clone or backup fails to mount on a SnapManager host with an error message.

The error message is:

```
0001-034 Command error: mount failed: mount: filer1:/vol/
SnapManager_20090914112850837_vol14 on /opt/ONTAPsmo/mnt/-
ora_data02-20090914112850735_1 - WARNING unknown option "zone=vol14" nfs
mount: filer1:/vol/SnapManager_20090914112850837_vol14: Permission denied.
```

At the same time, the following message is generated at the storage system console:

```
Mon Sep 14 23:58:37 PDT [filer1: export.auto.update.disabled:
warning]: /etc/exports was not updated for vol14 when the vol clone create
command was run. Please either manually update /etc/exports or copy /etc/
exports.new to it.
```

This message might not be captured in the AutoSupport messages.

Note: You might encounter similar issue while cloning FlexVol volumes on NFS. You can follow the same steps to enable the `nfs.export.auto-update` option.

What to do

1. Set the `nfs.export.auto-update` option *on* so that the `/etc/exports` file is updated automatically.

   ```
   options nfs.export.auto-update on
   ```

   Note: In an HA pair configuration, ensure you set the NFS exports option `on` for both the storage systems.

Where to go for more information

The information provided is about the basic tasks involved in installing SnapManager. For more information, review the following documents:

Document	Where it is
SnapManager description page	This page contains information about SnapManager, pointers to online documentation, and a link to the SnapManager Download page, where you can download the software.
Compatibility and Configuration Guide for FCP and iSCSI Products	This document is available at *www.ibm.com/storage/support/nseries/*. It is a dynamic, online document that contains the most up-to-date information about the requirements for setting up a system in a SAN environment. It provides the current details about storage systems and host platforms, cabling issues, switch issues, and configurations.
SnapManager SnapDrive Compatibility Matrix	This document is available in the Interoperability section at *www.ibm.com/systems/storage/network/interophome.html*. It is a dynamic, online document that contains the most up-to-date information specific to SnapManager and its platform requirements.
SnapManager Release Notes	This document comes with SnapManager. You can also download a copy from *www.ibm.com/storage/support/nseries/*. It contains any last-minute information that you need to get the configuration up and running smoothly.
Host attach and support kits documentation	*www.ibm.com/storage/support/nseries/*
N series Introduction and Planning Guide	*www.ibm.com/storage/support/nseries/*

Document	Where it is
Technical Reports	The technical reports contain information about products that IBM licenses and in some cases customizes. Technical reports might contain information about models and features that are not supported by IBM.
Host operating system and database information	See the readme files and other documentation that you received with your host operating system and database software.

If you want...	Go to...
General product information	The N series support website (accessed and navigated as described in *Websites* on page 13).
Product support information	The N series support website (accessed and navigated as described in *Websites* on page 13).

Error message classifications

SnapManager assigns error messages to the following numerical ranges. Although this document does not include information for every component or error condition, understanding the message classification can help you determine the cause of an error. The following table provides information about the numerical ranges for the different types of messages you might see with SnapManager. Reviewing the general common error message numerical ranges might help you determine to which operation or object an error message pertains.

Group	Range	Usage
ENVIRONMENT	1000-1999	Message used to log the state or issues with the operating environment of SnapManager. These include messages about systems that SnapManager interacts with, such as the host, the storage system, the database, SnapDrive for UNIX, and so on.
BACKUP	2000-2999	Messages associated with the database backup process.
RESTORE	3000-3999	Messages associated with the database restore process.
CLONE	4000-4999	Messages associated with the database clone process.
PROFILE	5000-5999	Messages associated with managing profiles.
MANAGE	6000-6999	Messages associated with managing backups.
VIRTUAL DATABASE INTERFACE	7000-7999	Messages associated with the Virtual Database Interface. These messages deal with the interface from the product to the various Oracle versions.
VIRTUAL STORAGE INTERFACE	8000-8999	Messages associated with the Virtual Storage Interface. These messages deal with the interface from the product to the various Data ONTAP and SnapDrive for UNIX versions.
REPOSITORY	9000-9999	Messages associated with the Repository interface.
METRICS	10000-10999	Messages associated with the size of the database backup, the elapsed time to perform the backup, the time to restore the database, the number of times a database has been cloned, and so on.
VIRTUAL HOST INTERFACE	11000-11999	Messages associated with the Virtual Host Interface. This is the interface to the host operating system. It hides the differences between host platforms.

Group	Range	Usage
EXECUTION	12000-12999	Messages associated with the execution package, including spawning and processing operating system calls.
PROCESS	13000-13999	Messages associated with the process component of SnapManager.
UTILITIES	14000-14999	Messages associated with SnapManager utilities, global context, and so on.
DUMP/DIAGNOSTICS	15000-15999	Messages associated with dump or diagnostic operations.
HELP	16000-16999	Messages associated with help.
SERVER	17000-17999	Messages associated with the SnapManager server administration.
API	18000-18999	Messages associated with the API.
AUTH	20000-20999	Messages associated with credentials authorization.

Error messages

You can review the possible error messages associated with different SnapManager operations.

Most common error messages

The following table lists some of the most common and important errors associated with SnapManager for Oracle:

Error message	Explanation	Resolution
`ORA-01031: insufficient privileges. Verify that the SnapManager Windows service is set up to run as a user with the correct privileges and that the user is included in the ORA_DBA group.`	You have insufficient privileges in SnapManager. The SnapManager service account is not part of the ORA_DBA group.	Check that the user account for the SnapManager service is part of ORA_DBA group. Do this through the Manage option in the computer icon on the desktop. Check local users and groups and ensure that the account is part of the ORA_DBA group. If the user is the local administrator, ensure that the user is in the group rather than the domain administrator.
`0001-CON-10002: Connected ASM disks with paths <paths> were not discovered by the ASM instance <asm_instance_sid>. Please verify that the ASM_DISKSTRING parameter and filesystem permissions allow these paths to be discovered.`	ASM disks were connected to the host, but the ASM instance was not able to discover them.	If ASM over NFS is being used, ensure the asm_diskstring parameter for the ASM instance includes the ASM disk files. For example, if the error states it cannot discover `smo/mnt/<dir_name>/<disk_name>`, then add `/smo/mnt/*/*` to `asm_diskstring`.

Error message	Explanation	Resolution
0001-DS-10021: Unable to set protection policy of dataset <dataset-name> to <new-protection-policy> because the protection policy is already set to <old-protection-policy>. Please use Protection Manager to change the protection policy	Once the protection policy of a dataset is set, SnapManager will not allow you to change the protection policy, because it may require re-aligning the baseline relationships and it may result in the loss of existing backups on secondary storage.	Update the protection policy using the N series Management Console data protection capability, which provides more options on migrating from one protection policy to another.
0001-SD-10028: SnapDrive Error (id:2618 code:102) Unable to discover the device associated with "lun_path". If multipathing in use, possible multipathing configuration error. Please verify configuration and retry.	This error can be seen if the host is not able to discover LUNs created on storage systems. Discovery problems can occur due to various reasons.	Make sure transport service used is properly installed and configured. Ensure SnapDrive can create and discover a LUN on the storage system.
0001-SD-10028: SnapDrive Error (id:2836 code:110) Failed to acquire dataset lock on volume "storage name":"temp_volume_nam"	You tried to restore using the indirect storage method and the temporary volume specified does not exist on primary storage.	Create a temporary volume on primary storage. Or, specify the correct volume name, if a temporary volume is already created.

Error message	Explanation	Resolution
`0001-SMO-02016: There may have been external tables in the database not backed up as part of this backup operation (since the database was not OPEN during this backup ALL_EXTERNAL_LOCATIONS could not be queried to determine whether or not external tables exist).`	SnapManager does not backup "external tables" (for example, tables that are not stored in .dbf files). Since the database was not open during the backup, SnapManager cannot determine if any external tables are being used.	There may have been external tables in the database not backed up as part of this backup operation (since the database was not open during this backup), ALL_EXTERNAL_LOCATIONS could not be queried to determine whether or not external tables exist.
`0001-SMO-11027: Cannot clone or mount snapshots from secondary storage because the snapshots are busy. Try cloning or mounting from an older backup.`	You tried to create a clone or mount Snapshot copies from secondary storage of the latest protected backup.	Try cloning or mounting from an older backup.
`0001-SMO-12346: Cannot list protection policies because Protection Manager product is not installed or SnapDrive is not configured to use it. Please install Protection Manager and/or configure SnapDrive...`	You tried to list protection policies on a system where Snapdrive is not configured to use the N series Management Console data protection capability.	Install the N series Management Console data protection capability and configure SnapDrive to use the N series Management Console data protection capability.

Error message	Explanation	Resolution
`0001-SMO-13032: Cannot perform operation: Backup Delete. Root cause: 0001-SMO-02039: Unable to delete backup of dataset: SD-10028: SnapDrive Error (id:2406 code:102) Failed to delete backup id: "backup_id" for dataset, error(23410):Snapshot "snapshot_name" on volume "volume_name" is busy.`	This error occurs when you try to free or delete the latest protected backup or a backup containing Snapshot copies that are baselines in a mirror relationship.	Free or delete the protected backup.
`0002-332 Admin error: Could not check SD.SnapShot.Clone access on volume "volume_name" for user username on Operations Manager server(s) "dfm_server". Reason: Invalid resource specified. Unable to find its ID on Operations Manager server "dfm_server"`	This occurs when proper access privileges and roles are not set.	Set access privileges or roles for the users who are trying to execute the command.

Error message	Explanation	Resolution
`[WARN] FLOW-11011: Operation aborted [ERROR] FLOW-11008: Operation failed: Java heap space.`	This issue occurs during the backup create, backup mount, and pruning the archive log files operations when there are more number of archive log files in a database.	1. Navigate to SnapManager installation directory. 2. Open the `launch-java` file from the path: `<installation directory>/bin/launch-java`. 3. Increase the value of the java heap space parameter `java -Xmx160m` to a higher value. For example, you can modify the value from a default value of 160m to 200m as `java -Xmx200m`.
`SD-10028: SnapDrive Error (id:2868 code:102) Could not locate remote snapshot or remote qtree.`	SnapManager for Oracle displays the backups as protected even if the protection job at the Protection Manager is only partially successful. This condition occurs when dataset conformance is in progress (when the baseline Snapshots are getting mirrored).	Take a new backup after the dataset is conformant.
`SMO-21019: The archive log pruning failed for the destination: "/mnt/destination_name/" with the reason: "ORACLE-00101: Error executing RMAN command: [DELETE NOPROMPT ARCHIVELOG '/mnt/destination_name/']`	This issue occurs when the archive log pruning fails in one of the destinations. In such a scenario, SnapManager continues to prune the archive log files from the other destinations. If any files are manually deleted from active file system, the RMAN fails to prune the archive log files from that destination.	Connect to RMAN from the SnapManager host. Run the RMAN `CROSSCHECK ARCHIVELOG ALL` command.

Error message	Explanation	Resolution
`SMO-13032: Cannot perform operation: Archive log Prune. Root cause: RMAN Exception: ORACLE-00101: Error executing RMAN command.`	This issue occurs when the archive log files are manually deleted from the archive log destinations.	Connect to RMAN from the SnapManager host. Run the RMAN `CROSSCHECK ARCHIVELOG ALL` command and perform the pruning the archive log files again.

Error message	Explanation	Resolution

`SQLPlus not found error displayed when SnapManager operation is performed on a HP-UX platform.`	When you perform a SnapManager operation on the HP-UX platform, and encountered an error SQLPlus not found, you need to verify the path for the Oracle user.
	If the path is set as: `PATH=/sbin:/usr/sbin:/usr/bin`, you need to change the path.
	To verify the path, enter the following command:
	`# su oracle -c env | grep ^PATH`
	`PATH=/sbin:/usr/sbin:/usr/bin`
	The specified path is insufficient for SnapManager to execute operations, you must change the default security settings in the security file for HP-UX platform.
		To modify the default security settings in the HP-UX security file, perform the following steps:
		1. Login as root user and provide write permission to user using the following command:`chmod 644 /etc/default/security`
		2. Open the security file using an editor (example vi), locate the following line, and remove the line from the security file: `SU_DEFAULT_PATH=/sbin:/usr/sbin:/bin:/usr/bin`
		3. If the security file contains the SU_KEEP_ENV_VARS variable, then add HOME and SHLIB_PATH values to the variable, and save the security file: `SU_KEEP_ENV_VARS=HOME,SHLIB_PATH`
		4. To verify the updates made to the security file, enter the following command as a root user: `# su oracle -c env | grep ^PATH`
		Now the complete path is displayed.
		`PATH=/usr/sbin:/usr/bin:/usr/ccs/bin:/usr/contrib/bin ...`
		You can also check for the necessary environment variables in the security file.

Error messages associated with database backups process (2000 series)

The following table shows common errors associated with the database backups process:

Error message	Explanation	Resolution
`SMO-02066: You cannot delete or free the archive log backup "data-logs" as the backup is associated with data backup "data-logs".`	This error message is displayed when the archive log backup is taken along with data files backup, and you tried to delete the archive log backup.	Use the `-force` option to delete or free the backup.
`SMO-02067: You cannot delete, or free the archive log backup "data-logs" as the backup is associated with data backup "data-logs" and is within the assigned retention duration.`	This error message is displayed when the archive log backup is associated with the database backup and is with in the retention period, and you tried to delete the archive log backup.	Use the `-force` option to delete or free the backup.
`SMO-07142: Archived Logs excluded due to exclusion pattern <exclusion> pattern.`	This error message is displayed during the profile create or backup create operation if you have excluded some archive log files.	No action is necessary.
`SMO-07155: <count> archived log files do not exist in the active filesystem. These archived log files will not be included in the backup.`	This error message is displayed during the profile create or backup create operation when the archive log files do not exist in the active file system. These archived log files will not be included in the backup.	No action necessary.
`SMO-07148: Archived log files are not available.`	This error message is displayed during the profile create or backup create operation when there are no archive log files created for the current database incarnation.	No action necessary.

Error message	Explanation	Resolution
`SMO-07150: Archived log files are not found.`	This error message is displayed during the profile create or backup create operation if all the archive log files are missing from the file system or excluded.	No action necessary.

Data protection errors

The following table shows common errors associated with data protection. The following list is sorted alphabetically.

Error message	Explanation	Resolution
`Backup protection requested but the database profile does not have a protection policy. Please update the protection policy in the database profile or do not use the 'protect' option when creating backups.`	You attempted to create a backup with protection to secondary storage; however, the profile associated with this backup does not have a protection policy specified.	Edit the profile and select a protection policy. Re-create the backup.
`Cannot delete profile because data protection is enabled and Protection Manager is temporarily unavailable. Please try again later.`	You attempted to delete a profile that has protection enabled; however, the N series Management Console data protection capability is unavailable.	Ensure that appropriate backups are stored in either primary or secondary storage. Disable protection in the profile. When the N series Management Console data protection capability is again available, return to the profile and delete it.
`Cannot list protection policies because Protection Manager is temporarily unavailable. Please try again later.`	While setting up the backup profile, you enabled protection on the backup so that the backup would be stored on secondary storage. However, SnapManager cannot retrieve the protection policies from the N series Management Console data protection capability.	Disable protection in the profile temporarily. Continue creating a new profile or updating an existing profile. When the N series Management Console data protection capability is again available, return to the profile.

Error message	Explanation	Resolution
`Cannot list protection policies because Protection Manager product is not installed or SnapDrive is not configured to use it. Please install Protection Manager and/or configure SnapDrive.`	While setting up the backup profile, you enabled protection on the backup so that the backup would be stored on secondary storage. However, SnapManager cannot retrieve the protection policies from the N series Management Console data protection capability. The Management Console data protection capability is not installed or SnapDrive is not configured.	Install the N series Management Console data protection capability. Configure SnapDrive. Return to the profile, re-enable protection, and select the protection policies available from the N series Management Console data protection capability.
`Cannot set protection policy because Protection Manager is temporarily unavailable. Please try again later.`	While setting up the backup profile, you enabled protection on the backup so that the backup would be stored on secondary storage. However, SnapManager cannot retrieve the protection policies from the N series Management Console data protection capability.	Disable protection in the profile temporarily. Continue creating or updating the profile. When the N series Management Console data protection capability is again available, return to the profile.
`Creating new dataset <dataset_name> for database <dbname> on host <host>.`	You attempted to create a backup profile. SnapManager is creating a dataset for this profile.	No action necessary.
`Data protection is not available because Protection Manager is not installed.`	While setting up the backup profile, you attempted to enable protection on the backup so that the backup would be stored on secondary storage. However, SnapManager cannot access protection policies from the N series Management Console data protection capability. The Management Console data protection capability is not installed.	Install the N series Management Console data protection capability.

Error message	Explanation	Resolution
`Deleted dataset <dataset_name> for this database.`	You deleted a profile. SnapManager will delete the associated dataset.	No action is necessary.
`Deleting profile with protection enabled and Protection Manager is no longer configured. Deleting profile from SnapManager but not cleaning up dataset in Protection Manager.`	You attempted to delete a profile that has protection enabled; however, the N series Management Console data protection capability is no longer installed, is no longer configured, or has expired. SnapManager will delete the profile, but not the profile's dataset from the N series Management Console data protection capability.	Reinstall or reconfigure the N series Management Console data protection capability. Return to the profile and delete it.
`Invalid retention class. Use "smo help backup" to see a list of available retention classes.`	When setting up the retention policy, you attempted to use an invalid retention class.	Create a list of valid retention classes by entering this command: **`smo help backup`** Update the retention policy with one of the available classes.
`Specified protection policy is not available. Use "smo protection-policy list" to see a list of available protection policies.`	While setting up the profile, you enabled protection and entered a protection policy that is not available.	Identify available protection policies, by entering the following command: **`smo protection-policy list`**
`Using existing dataset <dataset_name> for database <dbname> on host <host> since the dataset already existed.`	You attempted to create a profile; however, the dataset for the same database profile already exists.	Check the options from the existing profile and ensure that they match what you need in the new profile.

Error message	Explanation	Resolution
`Using existing dataset <dataset_name> for RAC database <dbname> since profile <profile_name> for the same RAC database already exists for instance <SID> on host <hostname>.`	You attempted to create a profile for an RAC database; however, the dataset for the same RAC database profile already exists.	Check the options from the existing profile and ensure that they match what you need in the new profile.
`The dataset <dataset_name> with protection policy <existing_policy_name> already exists for this database. You have specified protection policy <new_policy_name>. The dataset's protection policy will be changed to <new_policy_name>. You can change the protection policy by updating the profile.`	You attempted to create a profile with protection enabled and a protection policy selected. However, the dataset for the same database profile already exists, but has a different protection policy. SnapManager will use the newly-specified policy for the existing dataset.	Review this protection policy and determine if this is the policy you want to use for the dataset. If not, edit the profile and change the policy.

Error message	Explanation	Resolution
`Protection Manager deletes the local backups created by SnapManager for Oracle`	The Protection Manager deletes or frees the local backups created by SnapManager based on the retention policy defined in the Protection Manager. The retention class set for the local backups are not considered while deleting or freeing the local backups. When the local backups are transferred to a secondary storage system, the retention class set for the local backups on the primary storage system are not considered. The retention class specified in the transfer schedule is assigned to the remote backup.	Run the `dfpm dataset fix_smo` command from the Protection Manager server every time a new dataset is created. Now the backups are not deleted based on the retention policy set in the Protection Manager.

Error message	Explanation	Resolution
`You have selected to disable protection for this profile. This could potentially delete the associated dataset in Protection Manager and destroy the replication relationships created for that dataset. You will also not be able to perform SnapManager operations such as restoring or cloning the secondary or tertiary backups for this profile. Do you wish to continue (Y/N)?`	This warning message is displayed when you try to disable protection for a protected profile while updating the profile from SnapManager CLI or GUI. You can disable protection for the profile using the `-noprotect` option from the SnapManager CLI or clearing the **Protection Manager Protection Policy** checkbox in the Policies properties window from the SnapManager GUI. When you disable protection for the profile, SnapManager for Oracle deletes the dataset from Protection Manager, which unregisters all of the secondary and tertiary backup copies associated with that dataset within the Protection Manager database. Once a dataset is deleted, all secondary and tertiary backup copies are orphaned within the Protection Manager. Neither the Protection Manager nor the SnapManager for Oracle have the ability to access those backup copies. The backup copies can no longer be restored using SnapManager for Oracle. **Note:** The same warning message is displayed even when the profile is not protected.	This is a known issue in SnapManager for Oracle and expected behavior within Protection Manager when destroying a dataset, there are no workaround to prevent this from occurring. The orphaned backups need to be managed manually.

Error messages associated with restore process (3000 series)

The following table shows common errors associated with the restore process:

Error message	Explanation	Resolution
SMO-03031:Restore specification is required to restore backup <variable> because the storage resources for the backup has already been freed.	You attempted to restore a backup that has its storage resources freed without specifying a restore specification.	Specify a restore specification.
SMO-03032:Restore specification must contain mappings for the files to restore because the storage resources for the backup has already been freed. The files that need mappings are: <variable> from Snapshots: <variable>	You attempted to restore a backup that has its storage resources freed along with a restore specification that does not contain mapping for all the files to be restored.	Correct the restore specification file so that the mappings match the files to be restored.

Error message	Explanation	Resolution
`ORACLE-30028: Unable to dump log file <filename>. The file may be missing/ inaccessible/ corrupted. This log file will not be used for recovery.`	The error message is displayed when the online redo log files or archive log files could not be used for recovery. This error occurs due to following reasons: • The online redo log files or archived log files mentioned in the error message does not have sufficient change numbers to apply for recovery. This occurs when the DB is online with no transactions. The redo log or archived log files will not have any valid change numbers that could be applied for recovery. • The online redo log file or archived log file mentioned in the error message does not have sufficient access privileges for Oracle. • The online redo log file or archived log file mentioned in the error message is corrupted and could not be read by Oracle. • The online redo log file or archived log file mentioned in the error message is not found in the path mentioned.	If the file mentioned in the error message is an archived log file and if you have manually provided for recovery, make sure the file has full access permissions to Oracle. Even if the file has full permissions, and the message continues, the archive log file do not have any change numbers to be applied for recovery, and this message can be ignored.
`SMO-03038: Cannot restore from secondary because the storage resources still exist on primary. Please restore from primary instead.`	You tried to restore from secondary storage, but Snapshot copies exist on primary.	Always restore from primary if the backup has not been freed.

Error message	Explanation	Resolution
`SM0-03054: Mounting backup archbkp1 to feed archivelogs. DS-10001: Connecting mountpoints. [ERROR] FLOW-11019: Failure in ExecuteConnectionSteps: SD-10028: SnapDrive Error (id:2618 code:305). The following files could not be deleted. The corresponding volumes might be read-only. Retry the command with older snapshots. [ERROR] FLOW-11010: Operation transitioning to abort due to prior failure.`	These error messages occur during recovery, when SnapManager tries to mount the latest backup from secondary to feed the archive log files from secondary. Though, if there are any other backups, the recovery can succeed. But, if there are no other backups, the recovery might fail.	Do not delete the latest backups from primary. So that SnapManager can use the primary backup for recovery.

Error messages associated with the clone process (4000 series)

The following table shows common errors associated with the clone process:

Error message	Explanation	Resolution
`SMO-04133: Dump destination must not exist`	You are using SnapManager to create new clones; however, the dump destinations to be used by the new clone already exist. SnapManager cannot create a clone if the dump destinations exist.	Remove or rename the old dump destinations before you create a clone.
`SMO-04908: Not a FlexClone.`	The clone is a LUN clone. This applies for Data ONTAP 8.1 7-mode as well as Data ONTAP operating in Cluster-Mode.	SnapManager supports clone split on the FlexClone only.

Error message	Explanation	Resolution
`SMO-04904: No clone split operation running with split-id` *`split-id`*	This is due to an invalid operation ID or no such clone split operation is in progress.	Provide a valid split ID or split label for the clone split status, result, and stop operations.
`SMO-04906: Stop clone split operation failed with split-id` *`split-id`*	The split operation is complete.	Check whether the split process is in progress using the **`clone split-status`** or **`clone split-result`** command.
`SMO-13032:Cannot perform operation: Clone Create. Root cause: ORACLE-00001: Error executing SQL: [ALTER DATABASE OPEN RESETLOGS;]. The command returned: ORA-38856: cannot mark instance UNNAMED_INSTANCE_2 (redo thread 2) as enabled.`	The clone creation fails when you create the clone from the standby database using the following setup: • The primary database is a RAC setup and the standby database is standalone. • The standby is created using RMAN for taking the datafiles backup.	Add the `_no_recovery_through_resetlogs=TRUE` parameter in the clone specification file before creating the clone. See Oracle documentation (ID 334899.1) for additional information. Ensure that you have your Oracle metalink user name and password.

Error message	Explanation	Resolution
`[INFO] Operation failed. Syntax errors in clone specification: [error: cvc-complex-type.2.4c: Expected elements 'value@http://www.example.com default@http://www.example.com' before the end of the content in element parameter@http://www.example.com]`	The clone creation fails with this error message if you have not provided a value for a parameter in the clone specification file.	You must either provide a value for the parameter or delete that parameter if not required from the clone specification file.

Error messages associated with managing profile process (5000 series)

The following table shows common errors associated with the clone process:

Error message	Explanation	Resolution
`SMO-20600: Profile "profile1" not found in repository "repo_name". Please run "profile sync" to update your profile-to-repository mappings.`	Dump operation cannot be performed when a profile creation fails. The dump operation is dependent on profiles. When the profile creation itself fails, the dump operation cannot be performed.	Use the `smo system dump`.

Error messages associated with freeing backup resources (Backups 6000 series)

The following table shows common errors associated with backup tasks:

Error message	Explanation	Resolution
`SMO-06030: Cannot remove backup because it is in use: <variable>`	You attempted to perform the backup free operation using commands, when the backup is mounted, or has clones, or is marked to be retained on an unlimited basis.	Unmount the backup or change the retention policy to something other than on an unlimited basis. If clones exist, delete them.
`SMO-06045: Cannot free backup <variable> because the storage resources for the backup have already been freed`	You attempted to perform the backup free operation using commands, when the backup has been already freed.	You cannot free the backup if it is already freed.
`SMO-06047: Only successful backups can be freed. The status of backup <ID> is <status>.`	You attempted to perform the backup free operation using commands, when the backup status is not SUCCESS.	Try again after a successful backup.
`SMO-13082: Cannot perform operation <variable> on backup <ID> because the storage resources for the backup have been freed.`	Using commands, you attempted to mount a backup that has its storage resources freed.	You cannot mount, clone, or verify a backup that has its storage resources freed.

Virtual storage interface errors (Virtual storage interface 8000 series)

The following table shows common errors associated with virtual storage interface tasks:

Error message	Explanation	Resolution
`SMO-08017 ERROR Error discovering storage for /.`	SnapManager attempted to locate storage resources, but found datafiles, control files, or logs on the root / directory. These files should reside in a subdirectory, but not in the root. The root file system might be a hard drive in your local machine. SnapDrive cannot take Snapshot copies at this location and SnapManager cannot perform operations on these files.	Check to see if datafiles, control files, or redo logs are on the root. If so, move them to their correct locations or re-create control files or redo logs in their correct locations. For example: Move /redo.log to /data/oracle/redo.log, where /data/oracle is the mount point.

Error messages associated with the rolling upgrade process (9000 series)

The following table shows common errors associated with the rolling upgrade process:

Error message	Explanation	Resolution
`SMO-09234:Following hosts does not exist in the old repository. <hostnames>.`	You tried to perform rolling upgrade of a host, which does not exist in the previous repository version.	Check whether the host exists in the previous repository using the `repository show-repository` command from the earlier version of SnapManager CLI.
`SMO-09255:Following hosts does not exist in the new repository. <hostnames>.`	You tried to perform roll back of a host, which does not exist in the new repository version.	Check whether the host exists in the new repository using the `repository show-repository` command from the later version of SnapManager CLI.

Error message	Explanation	Resolution
SMO-09256:Rollback not supported, since there exists new profiles <profilenames>.for the specified hosts <hostnames>.	You tried to roll back a host that contains new profiles existing in the repository. However, these profiles did not exist in the host of the earlier SnapManager version.	Delete new profiles in the later or upgraded version of SnapManager before the rollback.
SMO-09257:Rollback not supported, since the backups <backupid> are mounted in the new hosts.	You tried to roll back a later version SnapManager host that has backups being mounted. These backups are not mounted in the earlier version SnapManager host.	Unmount the backups in the later version SnapManager host, and then perform the rollback.
SMO-09258:Rollback not supported, since the backups <backupid> are unmounted in the new hosts.	You tried to roll back a later version SnapManager host that has backups being unmounted.	Mount the backups in the later version SnapManager host, and then perform the rollback.

Error message	Explanation	Resolution
`SMO-09298:Cannot update this repository since it already has other hosts in the higher version. Please perform rollingupgrade for all hosts instead.`	You performed a rolling upgrade on a single host and then updated the repository for that host.	Perform a rolling upgrade on all the hosts.
`SMO-09297: Error occurred while enabling constraints. The repository might be in inconsistent state. It is recommended to restore the backup of repository you have taken before the current operation.`	This message gets displayed while performing a rolling upgrade or rollback operation if the repository database is left in an inconsistent state.	Restore the repository that you backed up earlier.

Execution of operations (12,000 series)

The following table shows common errors associated with operations:

Error message	Explanation	Resolution
`SMO-12347 [ERROR]: SnapManager server not running on host <host> and port <port>. Please run this command on a host running the SnapManager server.`	While setting up the profile, you entered information about the host and port. However, SnapManager cannot perform these operations because the SnapManager server is not running on the specified host and port.	Enter the command on a host running the SnapManager server. You can check the port with the command `lsnrctl status` and see the port on which the database is running. Change the port in the backup command, if needed.

Execution of process components (13,000 series)

The following table shows common errors associated with the process component of SnapManager:

Error message	Explanation	Resolution
`SMO-13083: Snapname pattern with value "x" contains characters other than letters, numbers, underscore, dash, and curly braces.`	When you were creating a profile, you customized the Snapname pattern; however, you included special characters not allowed.	Remove special characters other than letters, numbers, underscore, dash, and curly braces.
`SMO-13084: Snapname pattern with value "x" does not contain the same number of left and right braces.`	When you were creating a profile, you customized the Snapname pattern; however, the left and right braces do not match.	Enter matching opening and closing brackets in the Snapname pattern.
`SMO-13085: Snapname pattern with value "x" contains an invalid variable name of "y".`	When you were creating a profile, you customized the Snapname pattern; however, you included a variable that is not allowed.	Remove the offending variable. To see a list of acceptable variables, see "Snapshot copy naming."
`SMO-13086 Snapname pattern with value "x" must contain variable "smid".`	When you were creating a profile, you customized the Snapname pattern; however, you omitted the required variable of smid.	Insert the required smid variable.

Error message	Explanation	Resolution
`SMO-13902: Clone Split Start failed.`	There could be multiple reasons for this error: • No space in the volume. • SnapDrive is not running. • Clone could be a LUN clone. • Flex volume has restricted Snapshot copies.	Check for the available space in the volume using the command. Check for the available space in the volume using the **`clone split-estimate`** command. Confirm that the Flex volume has no restricted Snapshot copies.
`SMO-13904: Clone Split Result failed.`	This could be due to failure from the SnapDrive or storage system.	Try working on a new clone.
`SMO-13906: Split operation already running for clone label` *`clone-label`* `or ID` *`clone-id`*`.`	You are trying to split a clone that is already split.	The clone is already split and the clone related meta data will be removed.
`SMO-13907: Split operation already running for clone label` *`clone-label`* `or ID` *`clone-id`*`.`	You are trying to split a clone that is undergoing the split process.	You must wait until the split operation completes.

Error messages associated with SnapManager Utilities (14,000 series)

The following table shows common errors associated with SnapManager utilities:

Error message	Explanation	Resolution
`SMO-14501: Mail ID cannot be blank.`	You did not enter the e-mail address.	Enter a valid e-mail address.
`SMO-14502: Mail subject cannot be blank.`	You did not enter the e-mail subject.	Enter the appropriate e-mail subject.
`SMO-14506: Mail server field cannot be blank.`	You did not enter the e-mail server host name or IP address.	Enter the valid mail server host name or IP address.
`SMO-14507: Mail Port field cannot be blank.`	You did not enter the mail port number.	Enter the mail server port number.

Error message	Explanation	Resolution
`SMO-14508: From Mail ID cannot be blank.`	You did not enter the sender's e-mail address.	Enter a valid sender's e-mail address.
`SMO-14509: Username cannot be blank.`	This message is displayed when you enable authentication and did not provide the username.	Enter the e-mail authentication username.
`SMO-14510: Password cannot be blank. Please enter the password.`	This message is displayed when you enable authentication and did not provide the password.	Enter the mail authentication password.
`SMO-14550: Email status <success/ failure>.`	This could be due to invalid port number, invalid mail server, or invalid receiver's mail address.	Provide proper values during e-mail configuration.
`SMO-14559: Sending E-Mail notification failed: <error>.`	This could be due to invalid port number, invalid mail server, or invalid receiver's mail address.	Provide proper values during e-mail configuration.
`SMO-14560: Notification failed: Notification configuration is not available.`	Notification sending failed, since notification configuration not available.	Add notification configuration.
`SMO-14565: Invalid time format. Please enter time format in HH:mm.`	You have entered time in an incorrect format.	Enter the time in the format: hh:mm.
`SMO-14566: Invalid date value. Valid date range is 1-31.`	The date configured is incorrect.	Date should be in the range from 1 through 31.
`SMO-14567: Invalid day value. Valid day range is 1-7.`	The day configured is incorrect.	Enter the day range from 1 through 7.
`SMO-14569: Server failed to start Summary Notification schedule.`	The SnapManager server got shut down due to unknown reasons.	Start the SnapManager server.

Error message	Explanation	Resolution
`SMO-14570: Summary Notification not available.`	You have not configured summary notification.	Configure the summary notification.
`SMO-14571: Both profile and summary notification cannot be enable.`	You have selected both the profile and summary notification options.	Enable either the profile notification or summary notification.
`SMO-14572: Provide success or failure option for notification.`	You have not enabled the success or failure options.	You must select either success or failure option or both.

Common SnapDrive for UNIX error messages

The following table shows errors related to SnapDrive for UNIX:

Error message	Explanation
`0001-136 Admin error: Unable to log on to filer: <filer> Please set user name and/or password for <filer>`	Initial configuration error
`0001-382 Admin error: Multipathing rescan failed`	LUN discovery error
`0001-462 Admin error: Failed to unconfigure multipathing for <LUN>: spd5: cannot stop device. Device busy.`	LUN discovery error
`0001-476 Admin error: Unable to discover the device associated with...`	LUN discovery error
`0001-680 Admin error: Host OS requires an update to internal data to allow LUN creation or connection. Use 'snapdrive config prepare luns' or update this information manually...`	LUN discovery error
`0001-710 Admin error: OS refresh of LUN failed...`	LUN discovery error

Error message	Explanation
`0001-817 Admin error: Failed to create volume clone... : FlexClone not licensed`	Initial configuration error
`0001-817 Admin error: Failed to create volume clone... : Request failed as space cannot be guaranteed for the clone.`	Space issue
`0001-878 Admin error: HBA assistant not found. Commands involving LUNs should fail.`	LUN discovery error
`SMO-12111: Error executing snapdrive command "<snapdrive command>": <snapdrive error>`	SnapDrive for UNIX generic error

Related concepts

Snapshot copy naming on page 106

Copyright and trademark information

References in this documentation to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's or NetApp's intellectual property rights may be used instead of the IBM or NetApp product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM and NetApp, are the user's responsibility.

No part of this document covered by copyright may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S.A. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark information

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. A complete and current list of other IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

NetApp, the NetApp logo, Network Appliance, the Network Appliance logo, Akorri, ApplianceWatch, ASUP, AutoSupport, BalancePoint, BalancePoint Predictor, Bycast, Campaign Express, ComplianceClock, Cryptainer, CryptoShred, Data ONTAP, DataFabric, DataFort, Decru, Decru DataFort, DenseStak, Engenio, Engenio logo, E-Stack, FAServer, FastStak, FilerView, FlexCache, FlexClone, FlexPod, FlexScale, FlexShare, FlexSuite, FlexVol, FPolicy, GetSuccessful, gFiler, Go further, faster, Imagine Virtually Anything, Lifetime Key Management, LockVault, Manage ONTAP, MetroCluster, MultiStore, NearStore, NetCache, NOW (NetApp on the Web), Onaro, OnCommand, ONTAPI, OpenKey, PerformanceStak, RAID-DP, ReplicatorX, SANscreen, SANshare, SANtricity, SecureAdmin, SecureShare, Select, Service

Builder, Shadow Tape, Simplicity, Simulate ONTAP, SnapCopy, SnapDirector, SnapDrive, SnapFilter, SnapLock, SnapManager, SnapMigrator, SnapMirror, SnapMover, SnapProtect, SnapRestore, Snapshot, SnapSuite, SnapValidator, SnapVault, StorageGRID, StoreVault, the StoreVault logo, SyncMirror, Tech OnTap, The evolution of storage, Topio, vFiler, VFM, Virtual File Manager, VPolicy, WAFL, Web Filer, and XBB are trademarks or registered trademarks of NetApp, Inc. in the United States, other countries, or both.

All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

NetApp, Inc. is a licensee of the CompactFlash and CF Logo trademarks.

NetApp, Inc. NetCache is certified RealSystem compatible.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe on any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, N.Y. 10504-1785
U.S.A.

For additional information, visit the web at:
http://www.ibm.com/ibm/licensing/contact/

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM web sites are provided for convenience only and do not in any manner serve as an endorsement of those web sites. The materials at those web sites are not part of the materials for this IBM product and use of those web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

If you are viewing this information in softcopy, the photographs and color illustrations may not appear.

Index

A

B

C

D

R

S

T

U

V

W

NA 210-05123_A0, Printed in USA

GA32-2209-00