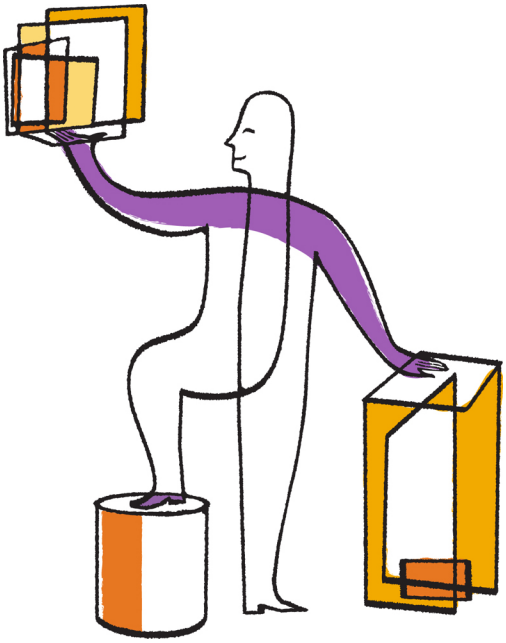
SnapManager® 3.4 for Oracle®

Administration Guide for UNIX®



NetApp, Inc.
495 East Java Drive
Sunnyvale, CA 94089
U.S.

Telephone: +1 (408) 822-6000
Fax: +1 (408) 822-4501
Support telephone: +1 (888) 463-8277
Web: www.netapp.com
Feedback: doccomments@netapp.com

Part number: 215-10242_A0
November 2015

Contents

Product overview 11
Create backups using Snapshot copies 11
Why you should prune archive log files 12
Archive log consolidation 12
Full or partial restoration of databases 12
Verify backup status 13
Database backup clones 13
Track details and produce reports 13
What repositories are 13
What profiles are 14
What SnapManager operation states are 15
How SnapManager maintains security 16
Accessing and printing online Help 17
General layout and configuration 18
Defining the database home with the oratab file 18
Requirements for using RAC databases with SnapManager 19
Requirements for using ASM databases with SnapManager 19
Supported partition devices 20
Support for ASMLib 21
Support for ASM databases without ASMLib 22
Requirements for using databases with NFS and SnapManager 26
Sample database volume layouts 27
Limitations 29
SnapManager limitations for clustered Data ONTAP 33
Limitations related to Oracle Database 34
Volume management restrictions 35
Upgrading SnapManager 36
Preparing to upgrade SnapManager 36
Upgrading the SnapManager hosts 36
Post-upgrade tasks 37
Updating the existing repository 37
Modifying the backup retention class 38
Restore process types 39
Upgrading SnapManager hosts by using rolling upgrade 39
Prerequisites for performing rolling upgrade 40
Performing rolling upgrade on a single host or multiple hosts 41
What a rollback is 43
Configuring SnapManager 47
SnapManager configuration parameters 47
Editing the configuration parameters 53

Configuring SnapDrive for UNIX for an active/active Veritas SFRAC environment ... 53
Configuring SnapManager to support the Veritas SFRAC environment ... 54
Ensuring that ASM discovers imported disks ... 54
Security and credential management ... 56
What user authentication is ... 56
About role-based access control ... 57
Enabling role-based access control ... 58
Setting role-based access control capabilities and roles ... 59
Storing encrypted passwords for custom scripts ... 62
Authorizing access to the repository ... 62
Authorizing access to profiles ... 62
Viewing user credentials ... 63
Clearing user credentials for all hosts, repositories, and profiles ... 63
Setting credentials after clearing the credential cache ... 64
Deleting credentials for individual resources ... 64
Deleting user credentials for repositories ... 65
Deleting user credentials for hosts ... 65
Deleting user credentials for profiles ... 65
Managing profiles for efficient backups ... 66
Creating profiles ... 67
Snapshot copy naming ... 72
Renaming profiles ... 73
Changing profile passwords ... 74
Resetting the profile password ... 74
Authorizing access to profiles ... 75
Verifying profiles ... 75
Updating profiles ... 75
Deleting profiles ... 78
Backing up databases ... 80
What SnapManager database backups are ... 81
What full and partial backups are ... 82
Backup types and the number of Snapshot copies ... 82
Full online backups ... 83
Partial online backups ... 84
Examples of backup, restore, and recover operations ... 84
About control file and archive log file handling ... 86
What database backup scheduling is ... 87
Creating database backups ... 90
Pruning archive log files ... 96
Consolidating archive log backups ... 98
Scheduling archive log file pruning ... 98
Protecting archive log backups ... 99
What AutoSupport is ... 100

Adding storage systems operating in clustered Data ONTAP to the SnapManager server host 100
Enabling AutoSupport in SnapManager 101
Disabling AutoSupport in SnapManager 101
Verifying database backups 102
Changing the backup retention policy 102
Retaining backups forever 102
Assigning backups with a specific retention class 103
Changing the retention policy default behavior 103
Freeing or deleting retention policy exempt backups 103
Viewing a list of backups 104
Viewing backup details 104
Mounting backups 106
Unmounting backups 106
Freeing backups 107
Deleting backups 108
Scheduling database backups 110
Creating backup schedules 110
Updating a backup schedule 112
Viewing a list of scheduled operations 113
Suspending backup schedules 113
Resuming backup schedules 113
Deleting backup schedules 113
Restoring database backup 115
What database restore is 116
When can you use fast restore 118
Fast restore eligibility checks 120
Backup recovery 126
Database state needed for the restore process 126
What restore preview plans are 127
Previewing backup restore information 128
Restoring backups by using fast restore 129
Restoring backups by using Single File SnapRestore 130
Restoring backups on primary storage 131
Performing block-level recovery with Oracle Recovery Manager (RMAN) 135
Restore files from an alternate location 137
Restore backups from an alternate location overview 138
Creating restore specifications 139
Restoring backups from an alternate location 141
Cloning database backup 142
What Cloning is 142
Cloning methods 143
Creating clone specifications 144
Cloning databases and using custom plug-in scripts 148
Cloning databases from backups 148

Cloning databases in the current state 150
Cloning database backups without resetlogs 150
Considerations for cloning a database to an alternate host 151
Cloning a database to an alternate host 152
Viewing a list of clones 153
Viewing detailed clone information 153
Deleting clones 154
Splitting a clone 155
Viewing a clone split estimate 155
Splitting a clone on primary or secondary storage 156
Viewing the status of the clone split process 157
Viewing the result of the clone split process 157
Stopping the clone split process 157
Deleting a profile 158
Destroying a profile 158
Deleting a clone split operation cycle from a repository database 158
Introduction to data protection in SnapManager 159
What protection policies are 159
What protection states are 160
What resource pools are 160
About different protection policies 160
Configuring and enabling policy-driven data protection 161
Configuring DataFabric Manager server and SnapDrive when RBAC is enabled 162
Configuring SnapDrive when RBAC is not enabled 163
Understanding enabling or disabling of data protection in profile 163
How SnapManager retains backups on the local storage 165
Prerequisites for data protection 167
Licences required for data protection in SnapManager 168
Protecting database backups on secondary storage 168
Restoring protected backups from secondary storage 170
Restores of protected backups overview 170
Restoring backups from secondary storage 171
Cloning protected backups 172
SnapManager for Oracle uses Protection Manager to protect a database backup 174
Details of the target database 174
Primary and secondary storage configuration and topology 174
Backup schedule and retention strategy 178
Workflow summary for local and secondary database backup 178
Protected backup configuration and execution 179
Using SnapManager for Oracle to create the database profile for a local backup 179
Using Protection Manager to configure a secondary resource pool 181
Using Protection Manager to configure secondary backup schedules 181

Using Protection Manager to configure a secondary backup protection policy ... 183
Using SnapManager for Oracle to create the database profile and assign a protection policy ... 184
Using Protection Manager to provision the new dataset ... 186
Using SnapManager for Oracle to create a protected backup ... 186
Using SnapManager for Oracle to confirm backup protection ... 187
Database restoration from backup ... 188
Using SnapManager for Oracle to restore a local backup on primary storage ... 188
Using SnapManager for Oracle to restore backups from secondary storage ... 189
Performing management operations ... 190
Viewing a list of operations ... 190
Viewing operation details ... 190
Issuing commands from an alternate host ... 191
Checking the SnapManager software version ... 191
Stopping the SnapManager host server ... 191
Restarting the SnapManager UNIX host server ... 192
Uninstalling the software from a UNIX host ... 192
Configuring notification ... 193
Configuring a mail server for a repository ... 194
Configuring email notification for a new profile ... 195
Customizing the email subject for a new profile ... 196
Configuring email notification for an existing profile ... 197
Customizing the email subject for an existing profile ... 198
Configuring summary email notification for multiple profiles ... 198
Adding a new profile to summary email notifications ... 199
Adding an existing profile to summary email notifications ... 200
Disabling email notification for multiple profiles ... 200
Creating task specification file and scripts for SnapManager operations ... 201
Creating pretask, post-task, and policy scripts ... 202
Operations in task scripts ... 205
Variables available in the task scripts for the backup operation ... 206
Variables available in the task scripts for the restore operation ... 208
Variables available in the task scripts for clone operation ... 209
Error handling in custom scripts ... 210
Viewing sample plug-in scripts ... 210
Creating task scripts ... 212
Storing the task scripts ... 214
Verifying the installation of plug-in scripts ... 215
Creating a task specification file ... 215
Performing backup, restore, and clone operations using prescript and post-scripts ... 216

Updating storage system name and target database host name associated with a profile 219
Updating the storage system name associated with a profile 219
Viewing a list of storage systems associated with a profile 220
Updating the target database host name associated with a profile 221
Maintaining history of SnapManager operations 223
Configuring history for backup operation 223
Viewing a list of SnapManager operation history 224
Viewing the detailed history of a specific operation associated with a profile 224
Deleting history of SnapManager operation 224
Removing history settings associated with a single profile or multiple profiles 225
Viewing SnapManager history configuration details 225
SnapManager for Oracle command reference 226
The smo_server restart command 226
The smo_server start command 227
The smo_server status command 227
The smo_server stop command 228
The smo backup create command 228
The smo backup delete command 232
The smo backup free command 233
The smo backup list command 234
The smo backup mount command 235
The smo backup restore command 237
The smo backup show command 240
The smo backup unmount command 242
The smo backup update command 243
The smo backup verify command 244
The smo clone create command 245
The smo clone delete command 248
The smo clone list command 249
The smo clone show command 250
The smo clone template command 251
The smo clone update command 252
The smo clone split-delete command 253
The smo clone split-estimate command 254
The smo clone split command 254
The smo clone split-result command 258
The smo clone split-stop command 259
The smo clone split-status command 259
The smo clone detach command 260
The smo cmdfile command 261
The smo credential clear command 262
The smo credential delete command 262
The smo credential list command 264
The smo credential set command 264

The smo history list command 266
The smo history operation-show command 267
The smo history purge command 268
The smo history remove command 269
The smo history set command 270
The smo history show command 271
The smo help command 272
The smo notification remove-summary-notification command 273
The smo notification update-summary-notification command 274
The smo notification set command 275
The smo operation dump command 276
The smo operation list command 277
The smo operation show command 278
The smo password reset command 279
The smo plugin check command 280
The smo profile create command 281
The smo profile delete command 285
The smo profile destroy command 286
The smo profile dump command 286
The smo profile list command 287
The smo profile show command 288
The smo profile sync command 289
The smo profile update command 290
The smo profile verify command 294
The smo protection-policy command 295
The smo repository create command 296
The smo repository delete command 297
The smo repository rollback command 298
The smo repository rollingupgrade command 299
The smo repository show command 300
The smo repository update command 301
The smo schedule create command 303
The smo schedule delete command 306
The smo schedule list command 307
The smo schedule resume command 307
The smo schedule suspend command 307
The smo schedule update command 308
The smo storage list command 309
The smo storage rename command 309
The smo system dump command 310
The smo system verify command 311
The smo version command 311
Troubleshooting SnapManager 313
Dump files 318
Creating operation-level dump files 320

Creating profile-level dump files 320
Creating system-level dump files 321
How to locate dump files 321
How to collect dump files 322
Collecting additional log information for easier debugging 322
Troubleshooting clone issues 323
Troubleshooting graphical user interface issues 326
Troubleshooting SnapDrive issues 331
Troubleshooting storage system renaming issue 332
Troubleshooting known issues 333
Mounting a FlexClone volume fails in NFS environment 338
Running multiple parallel operations fails in SnapManager 339
Unable to restore RAC database from one of the RAC nodes where the profile was not created 339
Where to go for more information 340
Error message classifications 341
Error messages 342
Copyright information 364
Trademark information 365
How to send comments about documentation and receive update notifications 366
Index 367

Product overview

SnapManager for Oracle automates and simplifies the complex, manual, and time-consuming processes associated with the backup, recovery, and cloning of Oracle databases. You can use SnapManager with Data ONTAP SnapMirror technology to create copies of backups on another volume and with Data ONTAP SnapVault technology to archive backups efficiently to disk.

SnapManager integrates with native Oracle technologies such as Oracle Real Application Clusters (Oracle RAC), Automatic Storage Management (ASM), and Direct NFS across FC, iSCSI, and NFS protocols. Optionally, backups created by using SnapManager can be cataloged with Oracle Recovery Manager (RMAN) to preserve the backup information; these backups can be used later in block-level restore or tablespace point-in-time recovery operations.

SnapManager highlights

SnapManager features seamless integration with Oracle databases on the UNIX host and with NetApp Snapshot, SnapRestore, and FlexClone technologies on the back end. It offers an easy-to-use user interface (UI) as well as command-line interface (CLI) for administrative functions.

SnapManager enables you to perform the following database operations and manage data efficiently:

- Creating space-efficient backups on primary or secondary storage
  SnapManager enables you to back up the data files and archive log files separately.
- Scheduling backups
- Restoring full or partial databases by using a file-based or volume-based restore operation
- Recovering databases by discovering, mounting, and applying archive log files from backups
- Pruning archive log files from archive log destinations when creating backups of only the archive logs
- Retaining a minimum number of archive log backups automatically by retaining only the backups that contain unique archive log files
- Tracking operation details and generating reports
- Verifying backups to ensure that backups are in a valid block format and that none of the backed-up files are corrupted
- Maintaining a history of operations performed on the database profile
  A profile contains information about the database to be managed by SnapManager.
- Creating space-efficient clones of backups on primary or secondary storage
  SnapManager enables you to split a clone of a database.

Create backups using Snapshot copies

SnapManager enables you to create backups on the primary (local) storage and also on the secondary (remote) storage using protection policies or postprocessing scripts.

Backups created as Snapshot copies are virtual copies of the database and are stored in the same physical medium as the database. Therefore, the backup operation takes less time and requires significantly less space than full, disk-to-disk backups. SnapManager enables you to back up the following:

- All the data files, archive log files, and control files

- Selected data files or tablespaces, all the archive log files, and control files

SnapManager 3.2 or later enables you to optionally back up the following:

- All the data files and the control files
- Selected data files or tablespaces along with the control files
- Archive log files

Note: The data files, archive log files, and control files can be located on different storage systems, storage system volumes, or logical unit numbers (LUNs). You can also use SnapManager to back up a database when there are multiple databases on the same volume or LUN.

Why you should prune archive log files

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

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

Note: You cannot prune the archive log files when Flash Recovery Area (FRA) is enabled for archive log files. If you specify the archive log location in Flash Recovery Area, you must ensure that you also specify the archive log location in the `archive_log_dest` parameter.

Archive log consolidation

SnapManager (3.2 or later) 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 that contain archive logs files that are subsets of other backups.

Full or partial restoration of databases

SnapManager provides the flexibility to restore full databases, specific tablespaces, files, control files, or a combination of these entities. SnapManager enables you to restore data by using a file-based restore process or a faster, volume-based restore process. Database administrators can select the process they want to use or let SnapManager decide which process is appropriate.

SnapManager enables database administrators (DBAs) to preview restore operations. The preview feature enables DBAs to view each restore operation on a file-by-file basis.

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

DBAs can use SnapManager to restore the database and use another tool to recover the information. DBAs are not required to use SnapManager for both operations.

SnapManager (3.2 or later) enables you to restore and recover database backups automatically without DBA intervention. You can use SnapManager to create archive log backups, and then use those archive log backups to restore and recover the database backups. Even if the backup's archive log files are managed in an external archive log location, you can specify that external location so those archive logs can help recover the restored database.

Verify backup status

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

Database administrators (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.

Database backup clones

SnapManager uses the FlexClone technology to create a writable, space-efficient clone of a database backup. You can modify a clone without changing the backup source.

You might want to clone databases to enable testing or upgrades in nonproduction environments. You can clone a database residing on primary or secondary storage. A clone can be located on the same host or on a different host as the database.

FlexClone technology enables SnapManager to use Snapshot copies of the database to avoid creating an entire physical, disk-to-disk copy. Snapshot copies require less creation time and take up significantly less space than physical copies.

See the Data ONTAP documentation for more information about FlexClone technology.

Related information

Data ONTAP documentation: mysupport.netapp.com/documentation/productsatoz/index.html

Track details and produce reports

SnapManager reduces the level of detail database administrators need to track the status of different operations by offering methods to monitor operations from a single interface.

After administrators specify which databases should be backed up, SnapManager automatically identifies the database files for backup. SnapManager displays information about repositories, hosts, profiles, backups, and clones. You can monitor the operations on specific hosts or databases. You can also identify the protected backups and determine whether backups are in process or scheduled to occur.

What repositories are

SnapManager organizes information into profiles, which are then associated with repositories. Profiles contain information about the database that is being managed, while the repository contains data about the operations that are 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 database administrators 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 database cannot exist in the same database and also cannot be a part of the same database that SnapManager is backing up. 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 operations.

If you try to open the graphical user interface (GUI) when the repository database is down, the following error message is logged in the `sm_gui.log` file: `[WARN]: SMO-01106: Error occurred while querying the repository: No more data to read from socket`. Also, SnapManager operations fail when the repository database is down. For more information about the different error messages, see *Troubleshooting known issues*.

You can use any valid host name, service name, or user name to perform operations. 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. 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 contain information about multiple profiles; however, each database is normally associated with only one profile. You can have multiple repositories, with each repository containing multiple profiles.

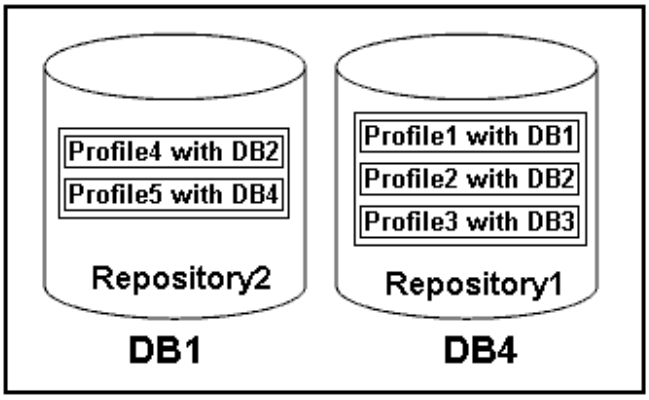
What profiles are

SnapManager uses profiles to store the information necessary to perform operations on a given database. A profile contains the information about the database including its credentials, backups, and clones. By creating a profile, you do not have to specify database details each time you perform an operation on that database.

A profile can reference only one database. The 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 serve as the database backup. 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 also that each profile can define only one database:



In the preceding example, Repository2 is on database DB1 and Repository1 is on the database DB4.

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

user name and password pairs for accessing the host, the repository, the database, and the required connection information if you are using Oracle Recovery Manager (RMAN).

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

Profile for creating full and partial backups

You can create profiles to take full backups or partial backups.

The profiles that you specify to create the full and partial backups contain both the data files and archive log files. SnapManager does not allow such profiles to separate the archive log backups from the data file backups. The full and partial backups are retained based on the existing backup retention policies and protected based on the existing protection policies. You can schedule full and partial backups based on the time and frequency that suits you.

Profiles for creating data files-only backups and archive log-only backups

SnapManager (3.2 or later) allows you to create profiles that take backups of the archive log files separately from the data files. After you use the profile to separate the backup types, you can create either data files-only backups or archive log-only backups of the database. You can also create a backup containing both the data files and archive log files together.

The retention policy applies to all the database backups when the archive log backups are not separated. After you separate the archive log backups, SnapManager allows you to specify different retention durations and protection policies 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 exceeds 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, and eligible backups expire.

Archive log retention duration

After the archive log backups are separated, they are retained based on the archive log retention duration. Archive log backups taken with data file backups are always retained along with those data file backups, regardless of the archive log retention duration.

Related concepts

Managing profiles for efficient backups on page 66

What SnapManager operation states are

SnapManager operations (backup, restore, and clone) can be in different states, with each state indicating the progress of the operation.

Operation state	Description
Succeeded	The operation completed successfully.
Running	The operation has started, but is not finished. For instance, a backup, which takes two minutes, is scheduled to occur at 11:00 a.m.. When you view the **Schedule** tab at 11:01 a.m., the operation appears as Running.

Operation state	Description
No operation found	The schedule has not run or the last run backup was deleted.
Failed	The operation failed. SnapManager has automatically executed the abort process and cleaned the operation. **Note:** You can split the clone that is created. When you stop the clone split operation you started and the operation is stopped successfully, the clone split operation state displays as failed.

Recoverable and unrecoverable events

A recoverable SnapManager event has the following problems:

- The database is not stored on a storage system that runs Data ONTAP.
- An Automatic Storage Management (ASM) database is configured, but the ASM instance is not running.
- SnapDrive for UNIX is not installed or cannot access the storage system.
- SnapManager fails to create a Snapshot copy or provision storage 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 occurs, SnapManager performs an abort process and attempts to return the host, database, and storage system to the starting state. If the abort process fails, SnapManager treats the incident as an unrecoverable event.

An unrecoverable (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 stopped.
- An in-band abort operation fails when the storage system fails, the logical unit number (LUN) or storage volume is offline, or the network fails.

When an unrecoverable event occurs, SnapManager performs an abort process immediately. The host, database, and storage system might not have returned to the initial states. If this is the case, you must perform a cleanup after the SnapManager operation fails by deleting the orphaned Snapshot copy and removing the SnapManager lock file.

If you want to delete the SnapManager lock file, navigate to `$ORACLE_HOME` on the target machine and delete the `sm_lock_`*`TargetDBName`* file. After deleting the file, you must restart the SnapManager for Oracle server.

How SnapManager maintains security

You can perform SnapManager operations only if you have the appropriate credentials. Security in SnapManager is governed by user authentication and role-based access control (RBAC). RBAC enables database administrators to restrict the operations that SnapManager can perform against the volumes and LUNs that hold the data files in a database.

Database administrators enable RBAC for SnapManager by using SnapDrive. Database administrators then assign permissions to SnapManager roles and assign these roles to the users in the Operations Manager graphical user interface (GUI) or command-line interface (CLI). RBAC permission checks happen in the DataFabric Manager server.

In addition to role-based access, SnapManager maintains security by requesting user authentication through password prompts or by setting user credentials. An effective user is authenticated and authorized with the SnapManager server.

SnapManager credentials and user authentication differ significantly from SnapManager 3.0:

- In SnapManager versions earlier than 3.0, you would set an arbitrary server password when you install SnapManager. Anyone who wants to use the SnapManager server would need the SnapManager server password. The SnapManager server password would need to be added to the user credentials by using the `smo credential set -host` command.
- In SnapManager (3.0 and later), the SnapManager server password has been replaced by individual user operating system (OS) authentication. If you are not running the client from the same server as the host, the SnapManager server performs the authentication by using your OS user names and passwords. If you do not want to be prompted for your OS passwords, you can save the data to your SnapManager user credentials cache by using the `smo credential set -host` command.

  Note: The `smo credential set -host` command remembers your credentials when the `host.credentials.persist` property in the `smo.config` file is set to **`true`**.

Example

User1 and User2 share a profile called Prof2. User2 cannot perform a backup of Database1 in Host1 without permission to access Host1. User1 cannot clone a database to Host3 without permission to access Host3.

The following table describes different permissions assigned to the users:

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 the nonprofile commands such as `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. Perform one of the following actions:
   - In the main window, click **Help > Help Contents**.
   - In any window or wizard, click **Help** to display help specific to that window.
2. Use the **Table of Contents** in the left pane to navigate through the topics.
3. Click the Printer icon at the top of the help window to print individual topics.

General layout and configuration

You can find information about the recommended general database layouts and storage configurations to avoid 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.
- SnapManager requires a multiple of 4K block size.
- Include the database system identifier 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 Oracle Recovery Manager (RMAN), you must create RMAN-enabled profiles.

If you want 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 Automatic Storage Management (ASM) disk group.
- A disk group containing data files cannot contain other types of files.
- The logical unit number (LUN) for the data file disk group must be the only object in the storage volume.

The following are some guidelines for volume separation:

- Data files for only one database must be in the volume.
- You must use separate volumes for each of the following file classifications: database binaries, data files, 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.

For more information, see the *SnapManager for Oracle Best Practices*.

Related information

SnapManager for Oracle Best Practices: media.netapp.com/documents/tr-3761.pdf

Defining the database home with the oratab file

SnapManager uses the `oratab` file during operations to determine the Oracle database home directory. 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 as shown in the following table:

Host operating system	File location
Linux	`/etc/oratab`

Host operating system	File location
Solaris	`/var/opt/oracle/oratab`
IBM AIX	`/etc/oratab`

The sample `oratab` file contains the following information:

```
+ASM1:/u01/app/11.2.0/grid:N   # line added by Agent
oelpro:/u01/app/11.2.0/oracle:N         # line added by Agent
# SnapManager generated entry          (DO NOT REMOVE THIS LINE)
smoclone:/u01/app/11.2.0/oracle:N
```

Note: After Oracle is installed, you must ensure that the `oratab` file resides in the location specified in the previous table. If the `oratab` file does not reside in the correct location per your operating system, you must contact technical support for assistance.

Requirements for using RAC databases with SnapManager

You must know the recommendations for using Real Application Clusters (RAC) databases with SnapManager. The recommendations include port numbers, passwords, and authentication mode.

- In database authentication mode, the listener on each node that interacts with an instance of the RAC database must be configured to use the same port number.

  The listener that interacts with the primary database instance must be started prior to initiating a backup.

- In operating system authentication mode or an Automatic Storage Management (ASM) environment, the SnapManager server must be installed and running on each node in the RAC environment.

- The database user password (for example, for a system administrator or a user with the sysdba privilege) must be same for all the Oracle database instances in a RAC environment.

Requirements for using ASM databases with SnapManager

You must know the requirements for using Automatic Storage Management (ASM) databases with SnapManager. Knowing these requirements can help you avoid issues with the ASMLib, partitions, and clone specifications, among other things.

- SnapManager (3.0.3 or later) uses the new sysasm privilege available with Oracle 11gR2 instead of the sysdba privilege to administer an Oracle ASM instance.

  If you use the sysdba privilege to run administrative commands on the ASM instance, an error message is displayed. The database uses the sysdba privilege to access disk groups. If you connect to the ASM instance using the sysasm privilege, you have complete access to all the available Oracle ASM disk groups and management functions.

  Note: If you are using Oracle 10gR2 and 11gR1, you must continue to use the sysdba privilege.

- SnapManager (3.0.3 or later) supports backing up databases that are stored directly on ASM disk groups when the disk group also contains an Automatic Cluster File System (ACFS) volume.

  These files are indirectly protected by SnapManager and might be restored with the remaining contents of an ASM diskgroup, but SnapManager (3.0.3 or later) does not support ACFS.

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

- SnapManager (3.0.3 or later) supports the backup of files that are stored on ASM disk groups when the disk group also contains Oracle Cluster Registry (OCR) files or voting disk files; however, restore operations require slower, host-based or partial-file snap restore (PFSR) method.
  It is best to have OCR and voting disks on disk groups that do not contain database files.
- Each disk used for ASM must contain only one partition.
- The partition hosting the ASM data must be properly aligned to avoid severe performance problems.
  This implies that the LUN must be of the correct type and the partition must have an offset that is a multiple of 4K bytes.

  Note: For details about how to create partitions that are aligned to 4K, see the Knowledge Base article 1010717.
- ASM configuration is not specified as part of the clone specification.
  You must manually remove the ASM configuration information in clone specifications that were created using SnapManager 2.1 before upgrading the host to SnapManager (2.2 or later).
- SnapManager 3.1, 3.1p1, and 3.2 or later support ASMLib 2.1.4.
- SnapManager 3.1p4 or later support ASMLib 2.1.4, 2.1.7, and 2.1.8.

For more information, see the *SnapManager for Oracle Best Practices*.

Related information

SnapManager for Oracle Best Practices: media.netapp.com/documents/tr-3761.pdf
Knowledge Base article 1010717: kb.netapp.com/support/index?page=content&id=1010717
Knowledge Base article 3011226: kb.netapp.com/support/index?page=content&id=3011226

Supported partition devices

You must know the different partition devices that are supported in SnapManager.

The following table provides partition information and how it can be enabled for different operating systems:

Operating system	Single partition	Multiple partition	Non-partition devices	File system or RAW devices
Red Hat Enterprise Linux 5x or Oracle Enterprise Linux 5x	Yes	No	No	ext3*
Red Hat Enterprise Linux 6x or Oracle Enterprise Linux 6x	Yes	No	No	ext3 or ext4*
SUSE Linux Enterprise Server 11	Yes	No	No	ext3*
SUSE Linux Enterprise Server 10	No	No	Yes	ext3***
Red Hat Enterprise Linux 5x or later or Oracle Enterprise Linux 5x or later	Yes	No	Yes	ASM with ASMlib**

Operating system	Single partition	Multiple partition	Non-partition devices	File system or RAW devices
SUSE Linux Enterprise Server 10 SP4 or SUSE Linux Enterprise Server 11	Yes	No	Yes	ASM with ASMlib**
SUSE Linux Enterprise Server 10 SP4 or later or SUSE Linux Enterprise Server 11	Yes	No	No	ASM without ASMlib****

*

For a non-MPIO environment, enter the following command:

`sfdisk -uS -f -L -q /dev/ device_name`

For an MPIO environment, enter the following commands:

- **`sfdisk -uS -f -L -q /dev/ device_name`**
- **`kpartx -a -p p /dev/mapper/ device_name`**

**

For a non-MPIO environment, enter the following command:

`fdisk /dev/device_name`

For an MPIO environment, enter the following commands:

- **`fdisk /dev/mapper/device_name`**
- **`kpartx -a -p p /dev/mapper/device_name`**

***

Not applicable.

****

For an MPIO environment, enter the following command:

- **`kpartx -a -p p /dev/$kernel`**

For more information on the operating system versions supported, refer to the Interoperability Matrix.

Support for ASMLib

SnapManager supports different versions of ASMLib, although there are several factors you must consider when using SnapManager with ASMLib.

SnapManager supports ASMLib 2.1.4, 2.1.7, and 2.1.8. All SnapManager operations can be performed with ASMLib 2.1.4, 2.1.7, and 2.1.8.

If you have upgraded from ASMLib 2.1.4 to ASM 2.1.7, you can use the same profiles and backups created with ASMLib 2.1.4 to restore the backups and create the clones.

You must consider the following when using SnapManager with ASMLib:

- SnapManager 3.1 does not support ASMLib 2.1.7.
  SnapManager 3.1p4 or later support ASMLib 2.1.4, 2.1.7, and 2.1.8.

- After performing a rolling upgrade from SnapManager 3.1 to 3.2, the backups created by using ASMLib 2.1.7 work only if the repository is rolled back to SnapManager 3.1 and ASMLib 2.1.7 is downgraded to ASMLib 2.1.4.
- After performing a rolling upgrade from SnapManager 3.1 to 3.2, backups created by using ASMLib 2.1.7 do not work if the repository is rolled back to SnapManager 3.1 with ASMLib 2.1.7.
  The rollback succeeds, but the profiles and backups cannot be used.

Support for ASM databases without ASMLib

SnapManager supports ASM without ASMLib, by default. The basic requirement is that the devices that are used for ASM disk groups must be partitioned.

When ASMLib is not installed, the device permissions related to ASM disk groups are changed to root:disk when you perform the following operations:

- Restart the host
- Restore a database from the primary storage by using volume-based SnapRestore (VBSR)
- Restore a database from the secondary storage

You can set the proper device permissions by assigning **`true`** to the `oracleasm.support.without.asmlib` configuration variable in `smo.conf`. The devices related to the ASM disk groups are added or removed from the `initasmdisks` file whenever new devices are added or removed from the host. The `initasmdisks` file is located at `/etc/initasmdisks`.

For example, if you set `oracleasm.support.without.asmlib=`**`true`** and then perform a backup mount, new devices are added to `initasmdisks`. When the host is restarted, the device permissions and ownership are maintained by the startup scripts.

Note: The default value for `oracleasm.support.without.asmlib` is **`false`**.

Related references

Supported partition devices on page 20

Supported scripts

The `asmmain.sh` and `asmquerydisk.sh` scripts allow you to change the grid user, group, and the user, all of which are used to query the ASM disks. The scripts must always be executed from the root.

The `asmmain.sh` is the main script file called from any operation that adds or deletes devices. The `asmmain.sh` script calls another script internally, which needs to be executed from the root that has oracle grid credentials. This script queries the ASM disk group's devices, then adds those entries in the `initasmdisk` file with the permission and the ownership of the devices. You can change the permissions and ownership of this file based on your environment and the regex pattern that is used for matching only the `/dev/mapper/*p1`.

The `asmquerydisk.sh` script is used to query the disk list, which is used to create the ASM disk group. You must assign values to `ORACLE_BASE`, `ORACLE_HOME`, and `ORACLE_SID`, depending on your configuration.

The scripts are located at `/opt/NetApp/smo/plugins/examples/noasmlib`. However, these scripts must be moved to `/opt/NetApp/smo/plugins/noasmlib` before starting the SnapManager for Oracle server on the host.

Limitations of using scripts to support an ASM database without ASMLib

You must be aware of certain limitations to using scripts to support an ASM database without ASMLib.

- The scripts provide an alternative solution for any kernel version, but only if ASMLib is not installed.
- The permissions for the scripts must be set in such a way that the scripts can be accessed by root, grid, oracle, or equivalent users.
- The scripts do not support restoration from a secondary location.

Deploying and running the scripts

You can deploy and run the `asmmain.sh` and `asmquerydisk.sh` scripts to support ASM databases without ASMLib.

About this task

These scripts do not follow the pre-scripts or post-scripts syntax and workflow is called when `intitasmdisks` is enabled. You can change anything related to your configuration settings in the scripts. It is recommended to verify if everything in the scripts are working as expected by performing a quick dry run.

Note: These scripts do not harm your system on failures nor will they impact your system. These scripts are executed to update the ASM-related disks to have proper permissions and ownership, so that the disks will always be under ASM instance control.

Steps

1. Create the ASM disk groups with the partitioned disks.
2. Create the Oracle database on the DISK GROUPS.
3. Stop the SnapManager for Oracle server.

   Note: In an RAC environment, you need perform this step on all the RAC nodes.

4. Modify the `smo.conf` to include the following parameters:

   a. `oracleasm.support.without.asmlib` = **`true`**

   b. `oracleasm.support.without.asmlib.ownership` = **`true`**

   c. `oracleasm.support.without.asmlib.username` = *user name of your ASM instance environment*

   d. `oracleasm.support.without.asmlib.groupname` = *group name of your ASM instance environment*

   These modifications set the permissions for the absolute path only, which means instead of partition device, permissions will be set only for dm-* device.

5. Modify the plugins scripts available in `/opt/NetApp/smo/plugins/examples/noasmlib` to include your configuration settings in the scripts.
6. Copy the scripts to `/opt/NetApp/smo/plugins/noasmlib` before starting the SnapManager for Oracle server on the host.
7. Navigate to the `/opt/NetApp/smo` directory and perform a dry run by running the following script:

`sh plugins/noasmlib/asmmain.sh`

The `etc/initasmdisks` file is created, which is the main file that is used.

Example

You can confirm that the `etc/initasmdisks` file contains all the devices related to configured the ASM database, such as:

```
chown -R grid:oinstall /dev/mapper/360a98000316b61396c3f394645776863p1
         chmod 777 /dev/mapper/360a98000316b61396c3f394645776863p1
         chown -R grid:oinstall /dev/mapper/
360a980003754322f7a2b433469714239p1
         chmod 777 /dev/mapper/360a980003754322f7a2b433469714239p1
         chown -R grid:oinstall /dev/mapper/
360a980003754322f7a2b433469714241p1
         chmod 777 /dev/mapper/360a980003754322f7a2b433469714241p1
         chown -R grid:oinstall /dev/mapper/
360a980003754322f7a2b433469714243p1
         chmod 777 /dev/mapper/360a980003754322f7a2b433469714243p1
```

8. Start the SnapManager for Oracle server.
9. Configure SnapDrive for UNIX by adding the following to `snapdrive.conf` file.

   `disconnect-luns-before-vbsr=on`
10. Restart the SnapDrive for UNIX server.

    Note: In an RAC environment, you need perform the Step 3 through Step 10 for all the RAC nodes.

After you finish

The `/etc/initasmdisks` file created, must be executed from either one of the startup scripts or from a script that is newly defined in the rc3.d. The `/etc/initasmdisks` file should always be executed before the oracleha service starts.

Example

```
# ls -ltr *ohasd*
          lrwxrwxrwx 1 root root 17 Aug  7 02:34 S96ohasd -> /etc/init.d/
ohasd
          lrwxrwxrwx 1 root root 17 Aug  7 02:34 K15ohasd -> /etc/init.d/
ohasd
```

In the following example, `sh -x /etc/initasmdisks` will not be available by default, and you need to append it as the first line in the function `start_stack()` in an ohasd script:

```
start_stack()
{
sh -x /etc/initasmdisks
# see init.ohasd.sbs for a full rationale case $PLATFORM in Linux
}
```

Support for Oracle RAC ASM databases without ASMLib

If you are using Oracle RAC databases, the RAC nodes must be updated with the `initasmdisks` file whenever an operation is performed in the master RAC node.

If no authentication is required to log in into the RAC nodes from the master node, the `asmmain.sh` performs a secure copy (SCP) of `initasmdisks` to all the RAC nodes. The master node's

`initasmdisks` file will be called whenever restore happens, and the `asmmain.sh` script can be updated to invoke the same script in all the RAC nodes.

The `/etc/initasmdisks` file created that must be executed from either one of the startup scripts or from a newly defined script in the rc3.d. The `/etc/initasmdisks` file should always be executed before the oracleha service starts.

Support for Oracle 10g ASM databases without ASMLib

If you are using Oracle 10g, the `asmcmd` command is not available for listing disks. You can use the sql query to obtain the disks list.

The `disk_list.sql` script is included in the existing scripts provided in the examples directory to support sql queries. When you execute the`asmquerydisk.sh` script, the `disk_list.sql` script must be executed manually. The example script lines are added with comments in the `asmquerydisk.sh` file. This file can either be placed in the `/home/grid` location or another location of your choice.

Sample scripts to support ASM databases without ASMLib

The sample scripts are available in the `plugins/examples/noasmlib` directory of the SnapManager for Oracle installation directory.

asmmain.sh

```
#!/bin/bash
griduser=grid
gridgroup=oinstall

# Run the script which takes the disklist from the asmcmd
# use appropriate user , here grid user is being used to run
# asmcmd command.
su -c "plugins/noasmlib/asmdiskquery.sh" -s /bin/sh grid
cat /home/grid/disklist

# Construct the final file as .bak file with propre inputs
awk -v guser=$griduser -v ggroup=$gridgroup '/^\/dev\/mapper/ { print
"chown -R "guser":"ggroup" "$1; print "chmod 777 " $1; }' /home/grid/
disklist > /etc/initasmdisks.bak

# move the bak file to the actual file.
mv /etc/initasmdisks.bak /etc/initasmdisks

# Set full full permission for this file to be called while rebooting
and restore
chmod 777 /etc/initasmdisks

# If the /etc/initasmdisks needs to be updated in all the RAC nodes
# or /etc/initasmdisks script has to be executed in the RAC nodes, then
the following
# section needs to be uncommented and used.
#
# Note: To do scp or running scripts in remote RAC node via ssh, it
needs password less login
# for root user with ssh keys shared between the two nodes.
#
# The following 2 lines are used for updating the file in the RAC nodes:
# scp /etc/initasmdisks root@racnode1:/etc/initasmdisks
# scp /etc/initasmdisks root@racnode2:/etc/initasmdisks
#
# In order to execute the /etc/initasmdisks in other RAC nodes
# The following must be added to the master RAC node /etc/initasmdisks
file
# from the asmmain.sh script itself. The above scp transfer will make
sure
# the permissions and mode for the disk list contents are transferred to
```

```
the other RAC nodes
# so now appending any command in the /etc/initasmsdisks will be
retained only in the master RAC node.
# The following lines will add entries to the /etc/initasmsdisks file in
master RAC node only. When this script is executed
#  master RAC node, /etc/initasmdisks in all the RAC nodes will be
executed.
# echo 'ssh racnode1 /etc/initasmdisks' >> /etc/initasmdisks
# echo 'ssh racnode2 /etc/initasmdisks' >> /etc/initasmdisks
```

asmquerydisk.sh

```
#!/bin/bash
export  ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=/u01/app/grid/product/11.2.0.3/grid
export ORACLE_SID=+ASM
export  PATH=$ORACLE_HOME/bin:$PATH

# Get the Disk List and save this in a file called dglist.
asmcmd lsdsk  > /home/grid/disklist

# In oracle 10g the above used command 'asmcmd' is not available so use
SQL
# query can be used to take the disk list. Need to uncomment the
following
# line and comment the above incase oracle 10g is being in use.
# The disk_list.sql script is availbe in this noasmlib examples folder
itself
# which can be modified as per customer needs.
# sqlplus "/as sysdba" @/home/grid/disk_list.sql > /home/grid/disklist
```

disk_list.sql

```
# su - oracle
-bash-4.1$ cat disk_list.sql
select path from v$asm_disk;
exit
-bash-4.1$
```

Requirements for using databases with NFS and SnapManager

You must know the requirements for using databases with Network File System (NFS) and SnapManager. The recommendations include running as root, attribute caching, and symbolic links.

- The file systems must be mounted following the best practice recommendations in the *SnapManager for Oracle Best Practices*.
- You must run SnapManager as root; SnapManager must be able to access the file systems that contain data files, control files, online redo logs, archive logs, and the database home.

  Set either of the following NFS export options to ensure that root can access the file systems:

  - root=*`host name`*
  - rw=*`host name`*, anon=0
- You must disable attribute caching for all the volumes that contain database data files, control files, redo and archive logs, and the database home.

  Export the volumes by using the `noac` (for Solaris and AIX) or `actimeo=0` (for Linux) options.
- You must link the database data files from local storage to NFS to support symbolic links at the mount point-level only.

Related information

SnapManager for Oracle Best Practices: media.netapp.com/documents/tr-3761.pdf

Sample database volume layouts

You can refer to sample database volume layouts for help in configuring your database.

Single-instance databases

The following table shows a sample layout of a single-instance database:

File types	Volume names	Dedicated volume for file types	Automatic Snapshot copies
Oracle binaries	orabin_*host name*	Yes	On
Data files	oradata_*sid*	Yes	Off
Temporary data files	oratemp_*sid*	Yes	Off
Control files	oracntrl01_*sid* (Multiplexed) oracntrl02_*sid* (Multiplexed)	Yes	Off
Redo logs	oralog01_*sid* (Multiplexed) oralog02_*sid* (Multiplexed)	Yes	Off
Archive logs	oraarch_*sid*	Yes	Off

Real Application Clusters (RAC) databases

The following table shows a sample layout of a RAC database:

File types	Volume names	Dedicated volume for file types	Automatic Snapshot copies
Oracle binaries	orabin_*host name*	Yes	On
Data files	oradata_*dbname*	Yes	Off
Temporary data files	oratemp_*dbname*	Yes	Off
Control files	oracntrl01_*dbname* (Multiplexed) oracntrl02_*dbname* (Multiplexed)	Yes	Off
Redo logs	oralog01_*dbname* (Multiplexed) oralog02_*dbname* (Multiplexed)	Yes	Off
Archive logs	oraarch_*dbname*	Yes	Off
Cluster files	oracrs_*clustername*	Yes	On

Single instance of an Automatic Storage Management (ASM) database

The following table shows a sample layout of an ASM database:

File types	Volume names	LUN names	Dedicated volume for file types	Automatic Snapshot copies
Oracle binaries	orabin_*host name*	orabin_*host namelun*	Yes	On
Data files	oradata_*sid*	oradata_*sidlun*	Yes	Off
Temporary data files	oratemp_*sid*	Oratemp_*sidlun*	Yes	Off
Control files	oracntrl01_*sid* (Multiplexed) oracntrl02_*sid* (Multiplexed)	oracntrl01_*sidlun* (Multiplexed) oracntrl02_*sidlun* (Multiplexed)	Yes	Off
Redo logs	oralog01_*dbname* (Multiplexed) oralog02_*dbname* (Multiplexed)	oralog01_*dbnamelun* (Multiplexed) oralog02_*dbnamelun* (Multiplexed)	Yes	Off
Archive logs	oraarch_*sid*	Oraarch_*sidlun*	Yes	Off

ASM RAC databases

The following table shows a sample layout of an ASM RAC database:

File types	Volume names	LUN names	Dedicated volume for file types	Automatic Snapshot copies
Oracle binaries	orabin_*host name*	orabin_*host namelun*	Yes	On
Data files	oradata_*sid*	oradata_*sidlun*	Yes	Off
Temporary data files	oratemp_*sid*	Oratemp_*sidlun*	Yes	Off
Control files	oracntrl01_*sid* (Multiplexed) oracntrl02_*sid* (Multiplexed)	oracntrl01_*sidlun* (Multiplexed) oracntrl02_*sidlun* (Multiplexed)	Yes	Off
Redo logs	oralog01_*dbname* (Multiplexed) oralog02_*dbname* (Multiplexed)	oralog01_*dbname lun* (Multiplexed) oralog02_*dbnamelun* (Multiplexed)	Yes	Off
Archive logs	oraarch_*sid*	Oraarch_*sidlun*	Yes	Off
Cluster files	oracrs_*clustername*	oracrs_*clusternamelun*	Yes	On

Limitations

When working with SnapManager, you must be aware of certain limitations that might affect your environment.

Limitations related to database layouts and platforms

- SnapManager supports control files on a file system or in an ASM disk group and does not support control files on raw devices.
- SnapManager operates in a Microsoft clustering (MSCS) environment but does not recognize the state of the MSCS configuration (active or passive) and does not transfer active management of a repository to a standby server in an MSCS cluster.
- In Red Hat Enterprise Linux (RHEL) and Oracle Enterprise Linux 4.7, 5.0, 5.1, 5.2, and 5.3, the ext3 file system is not supported when deploying Oracle over raw devices by using dynamic multipathing (DMP) in a multipath network I/O (MPIO) environment.

  This issue is noticed in SnapManager only while using SnapDrive 4.1 for UNIX or earlier versions.
- SnapManager on Red Hat Enterprise Linux (RHEL) does not support partitioning of disks using the **parted** utility.

  This is an issue with Red Hat Enterprise Linux (RHEL) **parted** utility.
- In a RAC configuration, when a profile name is updated from RAC node A, the schedule file for the profile is updated only for RAC node A.

  The schedule file for the same profile on RAC node B is not updated and contains the earlier schedule information. When a scheduled backup is triggered from node B, the scheduled backup operation fails because node B contains the earlier schedule file. However, the scheduled backup operation succeeds from node A, on which the profile is renamed. You can restart the SnapManager server so that you receive the latest schedule file for the profile on node B.
- The repository database might exist on a host, which can be accessed by using more than one IP address.

  If the repository is accessed by using more than one IP address, then the schedule file is created for each of the IP addresses. If the schedule backup is created for a profile (for example, profile A) under one of the IP addresses (for example, IP1), then the schedule file for only that IP address gets updated. If profile A is accessed from another IP address (for example, IP2), the scheduled backup is not listed because the schedule file of IP2 does not have an entry for the schedule that was created under IP1.

  You can wait for the schedule to be triggered from that IP address and the schedule file to be updated, or you can restart the server.

Limitations related to SnapManager configuration

- 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.
- 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.
- SnapDrive for UNIX supports more than one type of file system and volume manager on certain platforms.

  The file system and volume manager used for database files must be specified in the SnapDrive configuration file as the default file system and volume manager.

- SnapManager supports databases on MultiStore storage systems with the following requirements:
  - You must configure SnapDrive to set passwords for MultiStore storage systems.
  - SnapDrive 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.
- SnapManager does not support accessing two SnapManager servers running on different ports from a single client (both from the CLI or GUI).
  The port numbers should be the same on the target and remote hosts.
- All LUNs within a volume should reside at the volume level or inside qtrees, but not both.
  This is because if the data is residing on the qtrees and you mount the volume, then the data inside the qtrees is not protected.
- SnapManager operations fail and you cannot access the GUI when the repository database is down.
  You must ensure that the repository database is running when you perform any SnapManager operations.
- SnapManager does not support Live Partition Mobility (LPM) and Live Application Mobility (LAM).
- SnapManager does not support Oracle Wallet Manager and Transparent Data Encryption (TDE).
- SnapManager does not support MetroCluster configuration in raw device mapping (RDM) environments because MetroCluster configuration is yet to be supported by Virtual Storage Console (VSC).

Limitations related to profile management

- If you update the profile to separate the archive log backups, then you cannot perform a rollback operation on the host.
- If you enable a profile from the GUI to create archive log backups, and later try to update the profile by using the Multi Profile Update window or Profile Update window, then you cannot modify that profile to create full backup.
- If you update multiple profiles (wherein some profiles have the **Backup Archivelogs separately** option enabled and other profiles have the option disabled) in the Multi Profile Update window, the **Backup Archivelogs separately** option is disabled.
- If you update multiple non-archive log profiles, the **Backup Archivelogs separately** option in the Multi Profile Update window is disabled.
- If you rename the profile, then you cannot roll back the host.

Limitations related to rolling upgrade or rollback operations

- If you try to install an earlier version of SnapManager for a host, without performing the rollback operation on the host in the repository, you might not be able to do the following:
  - View the profiles that were created in earlier or later versions of SnapManager for the host.
  - Access backups or clones that were created in earlier or later versions of SnapManager.
  - Perform rolling upgrade or rollback operations on the host.
- After you separate the profiles to create archive log backups, you cannot perform a rollback operation on the related host repository.

Limitations related to backup operations

- Backup creation might fail if you run SnapManager operations concurrently on the same host against a different ASM database.
- During recovery, if the backup is already mounted, SnapManager does not mount the backup again and uses the already mounted backup.

  If the backup is mounted by a different user, and if you do not have access to the previously mounted backup, then the other user must provide you the permission.

  All archive log files have read permission for users assigned to a group; you might not have the access permission to the archive log file, if the backup is mounted by a different user group. Users can give permission to the mounted archive log files manually and then retry the restore or recovery operation.
- 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.
- You can use the task specification file for scheduled backup only from SnapManager 3.2 or later.
- When a backup or clone operation is executed simultaneously on both the 10gR2 and 11gR2 RAC databases over ASM, then one of the backup or clone creation operations fails.

  This failure is due to a known Oracle limitation.
- SnapManager integrated with Protection Manager supports the backup of multiple volumes in primary storage to a single volume in secondary storage for SnapVault and qtree SnapMirror.

  Dynamic secondary volume sizing is not supported. See the *Provisioning Manager and Protection Manager Administration Guide For Use with DataFabric Manager Server 3.8* for more information.
- SnapManager does not support vaulting of backup using the post-processing script.
- If the repository database is pointing to more than one IP and each IP has a different host name, the backup scheduling operation is successful for one IP but fails for the other IP.
- After upgrading to SnapManager 3.4, any backups scheduled with post-scripts using SnapManager 3.3.1 cannot be updated.

  You must delete the existing schedule and create a new schedule.
- In clustered Data ONTAP environment, SnapManager does not support multiple secondary destinations for a source volume.

Limitations related to restore operations

- When an indirect method is used for performing a restore operation, and if the archive log files required for recovery are available only in backups from the secondary storage system, SnapManager fails to recover the database, because SnapManager cannot mount the backup of archive log files from the secondary storage system.
- When SnapManager performs a volume restore operation, the archive log backup copies that are made after the corresponding backup is restored are not purged.

  When the data files and archive log file destination exist on the same volume, the data files can be restored through a volume restore operation if there are no archive log files available in the archive log file destination. In such a scenario, the archive log Snapshot copies that are created after the backup of the data files are lost.

  You should not delete all the archive log files from the archive log destination.
- In an ASM environment, if the Oracle Cluster Registry (OCR) and voting disk files coexist on a disk group that has data files, then the fast restore preview operation displays the wrong directory structure for the OCR and voting disk.

Limitations related to clone operations

- You cannot view any numerical values between 0 and 100 for the progress of the clone split operation due to the speed with which the inodes are discovered and processed by the storage system containing the flexible volume.
- SnapManager does not support receiving emails only for the successful clone split operations.
- SnapManager only supports splitting a FlexClone.
- The cloning of online database backup of the RAC database that uses external archive log file location fails due to failure in recovery.

  The cloning fails because Oracle fails to find and apply the archive log files for recovery from the external archive log location. This is an Oracle limitation. For more information, see the Oracle BUG ID 13528007. Oracle does not apply archive log from the non-default location at the *Oracle support site*. Ensure that you have a valid Oracle metalink user name and password.
- SnapManager 3.3 or later does not support using the clone specification XML file created in the releases before SnapManager 3.2.
- If temporary tablespaces are located in a different location other than the datafiles location, a clone operation creates the tablespaces in the datafiles location.

  However, if temporary tablespaces are Oracle Managed Files (OMF) that are located in a different location other than the datafiles location, the clone operation does not create the tablespaces in the datafiles location. The OMFs are not managed by SnapManager.
- SnapManager fails to clone a RAC database if you select `-resetlogs` option.

Limitations related to archive log files and backups

- SnapManager does not support pruning of archive log files from the flash recovery area destination.
- SnapManager does not support pruning of archive log files from the standby destination.
- 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 by using SnapManager CLI or GUI, the modified retention class is not considered for backup because archive log backups are retained based on retention duration.
- If you delete the archive log files from the archive log destinations, the archive log backup does not include archive log files older than the missing archive log file.

  If the latest archive log file is missing, then the archive log backup operation fails.
- If you delete the archive log files from the archive log destinations, the pruning of archive log files fail.
- SnapManager consolidates the archive log backups even when you delete the archive log files from the archive log destinations or when the archive log files are corrupted.

Limitations related to changing of target database host name

The following SnapManager operations are not supported when you change the target database host name:

- Changing the target database host name from the SnapManager GUI.
- Rolling back of the repository database after updating the target database host name of the profile.
- Simultaneously updating multiple profiles for a new target database host name.
- Changing the target database host name when any SnapManager operation is running.

Limitations related to SnapManager CLI or GUI

- The SnapManager CLI commands for the `profile create` operation that are generated from the SnapManager GUI do not have history configuration options.

  You cannot use the profile create command to configure history retention settings from the SnapManager CLI.

- SnapManager does not display the GUI in the Mozilla Firefox when there is no Java Runtime Environment (JRE) available on the UNIX client.

- While updating the target database host name using the SnapManager CLI, if there are one or more open SnapManager GUI sessions, then all the open SnapManager GUI sessions fail to respond.

Limitations related to SnapMirror and SnapVault

- The SnapVault post-script is not supported if you are using Data ONTAP operating in 7-Mode.

- If you are using clustered Data ONTAP, you cannot perform volume-based SnapRestore (VBSR) on the backups that were created in the volumes that have SnapMirror relationships established.

  This is due to an ONTAP limitation, which does not allow you to break the relationship when doing a VBSR. However, you can perform a VBSR on the last or most recently created backup only when the volumes have SnapVault relationship established.

- If you are using Data ONTAP operating in 7-Mode and want to perform VBSR on the backups that were created in the volumes, which have SnapMirror relationship established, you might set `override-vbsr-snapmirror-check=`**`ON`** in SnapDrive for UNIX. For more information see SnapDrive documentation.

- In some scenarios, you cannot delete the last backup associated with the first Snapshot copy when the volume has SnapVault relationship established.

  You can delete the backup only when you break the relationship. This issue is because of Data ONTAP restriction with base Snapshot copies. In a SnapMirror relationship, the base Snapshot copy is created by SnapMirror engine and in a SnapVault relationship the base Snapshot copy is the backup created by using SnapManager. For each and every update, the base Snapshot copy points to the latest backup created by using SnapManager.

Limitations related to Data Guard Standby databases

- SnapManager does not support Logical Data Guard Standby databases.

- SnapManager does not support Active Data Guard Standby databases.

- SnapManager does not allow online backups of Data Guard Standby databases.

- SnapManager does not allow partial backups of Data Guard Standby databases.

- SnapManager does not allow restoring of Data Guard Standby databases.

- SnapManager does not allow pruning of archive log files for Data Guard Standby databases.

Related information

Documentation on the NetApp Support Site: mysupport.netapp.com

SnapManager limitations for clustered Data ONTAP

You must know the limitations for some functionalities and SnapManager operations if you are using clustered Data ONTAP.

The following functionalities are not supported if you are using SnapManager on clustered Data ONTAP:

- Data protection capabilities if SnapManager is integrated with OnCommand Unified Manager

- A database in which one LUN belongs to a system running Data ONTAP operating in 7-Mode and the other LUN belongs to a system running clustered Data ONTAP
- SnapManager for Oracle does not support migration of a Vserver, which is not supported by clustered Data ONTAP
- SnapManager for Oracle does not support the clustered Data ONTAP 8.2.1 functionality to specify different export policies for volumes and qtrees

Limitations related to Oracle Database

Before you start working with SnapManager, you must know the limitations related to Oracle Database.

The limitations are as follows:

- SnapManager supports Oracle versions 10gR2, 11gR1, 11gR2 and 12*c*, but does not support Oracle 10gR1 as the repository or target database.
- SnapManager will not support the use of a SCAN IP address in place of a host name.
  SCAN IP is a new feature in Oracle 11gR2.
- SnapManager does not support Oracle Cluster File System (OCFS).
- Oracle 11g in a Direct NFS (dNFS) environment allows additional mount point configurations in the `oranfstab` file, such as multiple paths for load balancing.
  SnapManager does not modify the `oranfstab` file. You must manually add any additional properties that you want the cloned database to use, in the `oranfstab` file.
- Support for Oracle Database 9i is deprecated from SnapManager 3.2.
- Support for Oracle Database 10gR2 (earlier than 10.2.0.5) is deprecated from SnapManager 3.3.1.

Note: Identify the different versions of Oracle databases supported by referring to the Interoperability Matrix.

Related information

Interoperability Matrix: support.netapp.com/NOW/products/interoperability

Deprecated versions of Oracle database

Oracle database 9i is not supported by SnapManager 3.2 or later, and Oracle database 10gR2 (earlier than 10.2.0.4) is not supported by SnapManager 3.3.1 or later.

If you are using Oracle 9i or 10gR2 (earlier than 10.2.0.4) databases and want to upgrade to SnapManager 3.2 or later, you cannot create new profiles; a warning message is displayed.

If you are using Oracle 9i or 10gR2 (earlier than 10.2.0.4) databases and want to upgrade to SnapManager 3.2 or later, you must perform one of the following:

- Upgrade Oracle 9i or 10gR2 (earlier than 10.2.0.4) databases to either Oracle 10gR2 (10.2.0.5), 11gR1, or 11gR2 databases, and then upgrade to SnapManager 3.2 or 3.3.
  If you are upgrading to Oracle 12*c*, then you must upgrade to SnapManager 3.3.1 or later.

  Note: Oracle database 12*c* is supported only from SnapManager 3.3.1.

- Manage the Oracle 9i databases using a patch version of SnapManager 3.1.
  You can use SnapManager 3.2 or 3.3 if you want to manage Oracle 10gR2, 11gR1, or 11gR2 databases and use SnapManager 3.3.1 or later if you want to manage Oracle 12*c* databases along with the other supported databases.

Volume management restrictions

SnapManager has certain volume management restrictions that might affect your environment.

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

- Disk groups for the database can be managed by only one volume manager.
- Raw devices backed by a logical volume manager are not supported for protection of Oracle data. Raw device storage and Automatic Storage Management (ASM) disk groups must be provisioned directly on physical devices. In some cases, partitioning is required.
- A Linux environment without logical volume management requires a partition.

Upgrading SnapManager

You can upgrade to the latest version of SnapManager for Oracle from any of the earlier versions. You can either upgrade all the SnapManager hosts simultaneously or perform a rolling upgrade, which allows you to upgrade the hosts in a staggered, host-by-host manner.

Preparing to upgrade SnapManager

The environment in which you want to upgrade SnapManager must meet the specific software, hardware, browser, database, and operating system requirements. For the latest information about the requirements, see the Interoperability Matrix.

You must ensure that you perform the following tasks before upgrading:

- Complete the required preinstallation tasks.
- Download the latest SnapManager for Oracle installation package.
- Install and configure the appropriate version of SnapDrive for UNIX on all the target hosts.
- Create a backup of the existing SnapManager for Oracle repository database.

Related information

Interoperability Matrix: support.netapp.com/NOW/products/interoperability

Upgrading the SnapManager hosts

You can upgrade all the existing hosts to use the latest version of SnapManager. All the hosts are upgraded simultaneously. However, be aware that this might result in downtime of all the SnapManager hosts and the scheduled operations during that time.

Steps

1. Log in to the host system as the root user.
2. From the command-line interface (CLI), navigate to the location where you have downloaded the installation file.
3. Optional: If the file is not executable, change the permissions by entering the following command:

   `chmod 544 netapp.smo*`

4. Stop the SnapManager server by entering the following command:

   `smo_server stop`

5. Depending on the UNIX host, install SnapManager by entering the appropriate command:

If the operating system is...	Then run...
Solaris (SPARC64)	**`# ./netapp.smo.sunos-sparc64-`*`version_number`*`.bin`**
Solaris (x86_64)	**`# ./netapp.smo.sunos-x64-`*`version_number`*`.bin`**
AIX (PPC64)	**`# ./netapp.smo.aix-ppc64-`*`version_number`*`.bin`**

If the operating system is...	Then run...
Linux x86	`# ./netapp.smo.linux-x86-`*`version_number`*`.bin`
Linux x64	`# ./netapp.smo.linux-x64-`*`version_number`*`.bin`

6. On the **Introduction** page, press **Enter** to continue.

   The following message is displayed: `Existing SnapManager For Oracle Detected.`

7. Press **Enter**.

8. At the command prompt, perform the following:

   a. Press **Enter** to accept the default value for the operating system user.

   b. Press **Enter** to accept the default value for operating system group.

   c. Press **Enter** to accept the default value for the server startup type.

   The configuration summary is displayed.

9. Press **Enter** to continue.

   The following message is displayed: `Uninstall of Existing SnapManager For Oracle has started.`

   The uninstallation is completed and the latest version of SnapManager is installed.

Post-upgrade tasks

After upgrading to a later version of SnapManager, you must update the existing repository. You might also want to modify the backup retention class assigned to the existing backups and identify which restore process you can use.

Note: After upgrading to SnapManager 3.3 or later, you need to set *`sqlnet.authentication_services`* to **`NONE`** if you want to use database (DB) authentication as the only authentication method. This feature is not supported for RAC databases.

Updating the existing repository

You must update the existing repository so that you can access it by using the upgraded SnapManager.

Before you begin

You must ensure that:

- The upgraded SnapManager server has been started and verified.
- The repository user name and service name consists of alphanumeric characters, a minus sign, an underscore, and a period.
  The repository port can be any valid port number and the repository host name must consist of alphanumeric characters, a minus sign, and a period.
- A backup of the existing repository exists.

Note: If you are upgrading from any version earlier than SnapManager 3.1 to SnapManager 3.3 or later, you must first upgrade to SnapManager 3.2. After upgrading to 3.2, you can then upgrade to SnapManager 3.3 or later.

Step

1. To update the existing repository, enter the following command:

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

   The other options for this command are as follows:

   - `[-force][-noprompt]`
   - `[quiet]` | `[verbose]`

   #### Example

   The following example shows a repository being updated:

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

After you finish

Restart the SnapManager server to restart any associated schedules.

Note: After you update the repository, you cannot use the repository with an earlier version of SnapManager.

Modifying the backup retention class

The upgraded SnapManager assigns the default backup retention class to the existing backups. You can modify the default retention class values to meet your backup requirements.

About this task

The default backup retention class assigned to the existing backups are as follows:

Backup type	Retention class assignment after upgrade
Backups to be retained forever	Unlimited
Other backups	Daily

Note: You can delete the backups that are retained forever without changing the retention class.

If you upgrade to SnapManager 3.0 or later, the greater of the following two values are assigned to the existing profiles:

- Previous retention count for the profile
- Default values for the retention count and duration of daily backups as specified in the `smo.config` file

Step

1. Modify the values assigned to `retain.hourly.count` and `retain.hourly.duration` in the `smo.config` file.

   #### Example

   You can enter the following values:

   - `retain.hourly.count` = **12**

- `retain.hourly.duration` = **2**

Related references

SnapManager configuration parameters on page 47

Restore process types

All restore processes are not supported in all SnapManager for Oracle versions. After upgrading SnapManager, you need to be aware of the restore process that you can use for restoring a backup.

The backups that are created by using SnapManager 3.0 or later can be restored by using both fast restore and file-based restore processes. However, the backups that are created by using a version earlier than SnapManager 3.0 can be restored by using only the file-based restore process.

You can determine the SnapManager version used to create the backup by running the `-backup show` command.

Related concepts

What database restore is on page 116

Upgrading SnapManager hosts by using rolling upgrade

The rolling upgrade approach that enables you to upgrade the hosts in a staggered, host-by-host manner is supported from SnapManager 3.1.

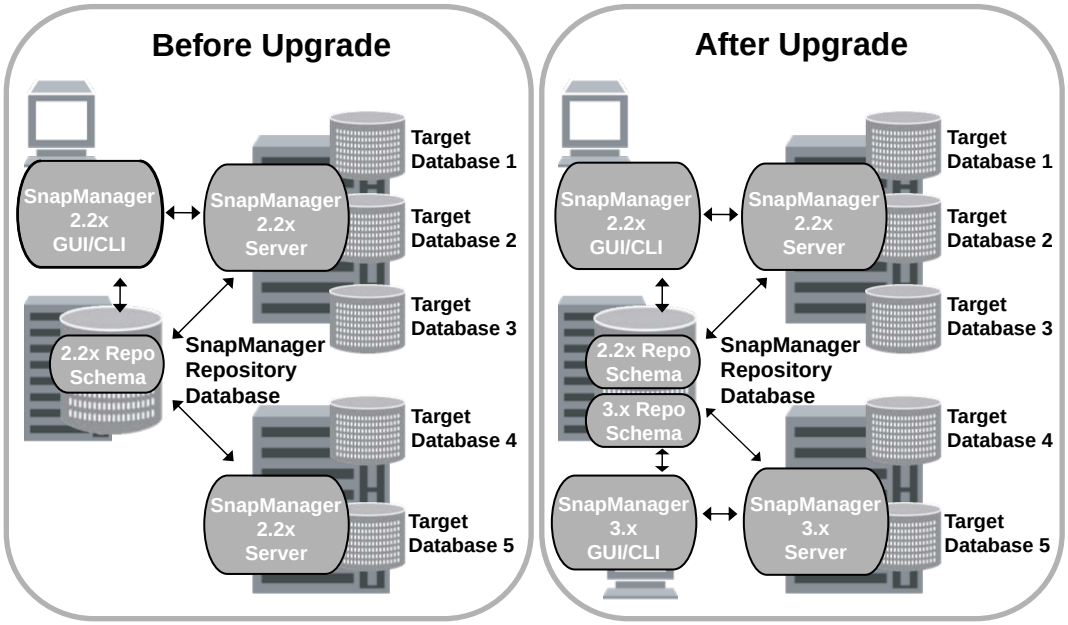
SnapManager 3.0 or earlier only enabled you to upgrade all the hosts simultaneously. This resulted in downtime of all the SnapManager hosts and the scheduled operations during upgrade operation.

Rolling upgrade provides the following benefits:

- Improved SnapManager performance because only one host is upgraded at one time.
- Ability to test the new features in one SnapManager server host before upgrading the other hosts.

Note: You can perform rolling upgrade only by using the command-line interface (CLI).

The following illustration shows the SnapManager architecture before and after performing rolling upgrade on one of the hosts:



After successful completion of rolling upgrade, the SnapManager hosts, profiles, schedules, backups, and clones associated with the profiles of the target databases are migrated from the repository database of the earlier SnapManager version to the repository database of the new version. The details about the operations performed by using the profiles, schedules, backups, and clones that were created in the earlier SnapManager version are now available in the repository database of the new version. You can start the GUI by using the default configuration values of the `user.config` file. The values configured in the `user.config` file of the earlier version of SnapManager are not considered.

The upgraded SnapManager server can now communicate with the upgraded repository database. The hosts that were not upgraded can manage their target databases by using the repository of the earlier version of SnapManager and thereby can use the features available in the earlier version.

Note: Before performing rolling upgrade, you must ensure that all the hosts under the repository database can be resolved. For information about how to resolve the hosts, see *Troubleshooting SnapManager for Oracle*.

Related concepts

What a rollback is on page 43

Related references

Troubleshooting SnapManager on page 313

Prerequisites for performing rolling upgrade

Before performing rolling upgrade, you must ensure that your environment meets certain requirements.

- If you are using any version earlier than SnapManager 3.1 and want to perform a rolling upgrade to SnapManager 3.3 or later, you need to first upgrade to 3.2 and then to the latest version.
  You can directly upgrade from SnapManager 3.2 to SnapManager 3.3 or later.
- External scripts that are used to perform any external data protection or data retention must be backed up.
- The SnapManager version to which you want to upgrade must be installed.

  Note: If you are upgrading from any version earlier than SnapManager 3.1 to SnapManager 3.3 or later, you must first install SnapManager 3.2 and perform a rolling upgrade. After upgrading to 3.2, you can then install SnapManager 3.3 or later and perform another rolling upgrade to SnapManager 3.3 or later.
- The SnapDrive for UNIX version supported with the SnapManager version to which you want to upgrade must be installed.
  For information about installing SnapDrive, see SnapDrive documentation set.
- The repository database must be backed up.
- The amount of SnapManager repository utilization should be minimum.
- If the host to be upgraded is using a repository, SnapManager operations must not be performed on the other hosts that are using the same repository.
  The operations that are scheduled or running on the other hosts will wait for the rolling upgrade to complete.
- 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 involving that repository database fails without warning.

- SnapManager operations must not be performed on the host that is being upgraded.

  Note: The rolling upgrade runs for longer as the number of backups of the hosts being upgraded together increases. The duration of the upgrade can vary depending on the number of profiles and backups associated with a given host.

Related information

Documentation on the NetApp Support Site: mysupport.netapp.com

Performing rolling upgrade on a single host or multiple hosts

You can perform rolling upgrade on a single or multiple SnapManager server hosts by using the command-line interface (CLI). The upgraded SnapManager server host is then managed only with the later version of SnapManager.

Before you begin

You must ensure that all the prerequisites for performing rolling upgrade are completed.

Steps

1. To perform a rolling upgrade on a single 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

   The following command performs the rolling upgrade of all target databases mounted on hostA and a repository database named repoA located on repo_host:

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

2. To perform a rolling upgrade on multiple hosts, 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_database1,host_with_target_database2* -force [-quiet | -verbose]

   Note: For multiple hosts, enter the host names separated by a comma and ensure that you do not include any space between the comma and the next host name. If you are using a Real Application Clusters (RAC) configuration, you must manually upgrade all RAC-associated hosts. You can use `-allhosts` to perform the rolling upgrade of all the hosts.

   #### Example

   The following command performs the rolling upgrade of all the target databases mounted on the hosts, hostA and hostB and a repository database named repoA located on repo_host:

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

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

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

   After successfully upgrading the repository database, you can perform all the SnapManager operations on the target database.

   The upgraded SnapManager for Oracle retains the host-based user credentials, the Oracle software user credentials, and the Oracle Recovery Manager (RMAN) user credentials from the earlier version of SnapManager for Oracle.

   Example

   The following command performs the rolling upgrade of all the target databases available on a repository database named repoA located on repo_host:

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

After you finish

- If the SnapManager server starts automatically, you must restart the server to ensure that you can view the schedules.
- If you upgrade one of the two related hosts, you must upgrade the second host after upgrading the first.

  For example, if you have created a clone from host A to host B or mounted a backup from host A to host B, the hosts A and B are related to each other. When you upgrade host A, a warning message is displayed asking you to upgrade the host B soon after upgrading host A.

  Note: The warning messages are displayed even though the clone is deleted or the backup is unmounted from host B during the rolling upgrade of host A. This is because metadata exists in the repository for the operations performed on the remote host.

Related concepts

Prerequisites for performing rolling upgrade on page 40

What a rollback is

The rollback operation enables you to revert to an earlier version of SnapManager after you perform a rolling upgrade.

Note: Before performing a rollback, you must ensure that all the hosts under the repository database can be resolved.

When you perform a rollback, the following are rolled back:

- Backups that were created, freed, and deleted by using the SnapManager version from which you are rolling back
- Clones created from a backup that was created by using the SnapManager version from which you are rolling back
- Profile credentials modified by using the SnapManager version from which you are rolling back
- Protection status of the backup modified by using the SnapManager version from which you are rolling back

The features that were available in the SnapManager version that you were using but are not available in the version to which you are rolling back, are not supported. For example, when you perform a rollback from SnapManager 3.3 or later to SnapManager 3.1, the history configuration set for profiles in SnapManager 3.3 or later is not rolled back to the profiles in SnapManager 3.1. This is because the history configuration feature was not available in SnapManager 3.1.

Related references

Troubleshooting SnapManager on page 313

Limitations for performing a rollback

You must be aware of the scenarios in which you cannot perform a rollback. However, for some of these scenarios you can perform some additional tasks before performing rollback.

The scenarios in which you cannot perform rollback or have to perform the additional tasks are as follows:

- If you perform one of the following operations after performing a rolling upgrade:
  - Create a new profile.
  - Split a clone.
  - Change the protection status of the profile.
  - Assign protection policy, retention class, or SnapVault and SnapMirror relationships.
    In this scenario, after performing a rollback, you must manually remove the protection policy, retention class, or SnapVault and SnapMirror relationships that were assigned.
  - Change the mount status of the backup.
    In this scenario, you must first change the mount status to its original state and then perform a rollback.
  - Restore a backup.
  - Change the authentication mode from database authentication to operating system (OS) authentication.
    In this scenario, after performing a rollback, you must manually change the authentication mode from OS to database.

- If the host name for the profile is changed
- If profiles are separated to create archive log backups
  In this scenario, you cannot rollback to a version that is earlier than SnapManager 3.2.

Prerequisites for performing a rollback

Before performing a rollback, you must ensure that your environment meets certain requirements.

- If you are using SnapManager 3.3 or later and want to roll back to a version earlier than SnapManager 3.1, you need to roll back to 3.2 and then to the desired version.
- External scripts that are used to perform any external data protection or data retention must be backed up.
- The SnapManager version to which you want to roll back must be installed.

  Note: If you want to perform a rollback from SnapManager 3.3 or later to a version earlier than SnapManager 3.1, you must first install SnapManager 3.2 and perform a rollback. After rolling back to 3.2, you can then install SnapManager 3.1 or earlier and perform another rollback to that version.

- The SnapDrive for UNIX version supported with the SnapManager version to which you want to roll back must be installed.
  For information about installing SnapDrive, see SnapDrive documentation set.
- The repository database must be backed up.
- If the host to be rolled back is using a repository, SnapManager operations must not be performed on the other hosts that are using the same repository.
  The operations that are scheduled or running on the other hosts wait for the rollback to complete.
- 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 rollback operation involving that repository database fails without warning.
- SnapManager operations must not be performed on the host which you want to rollback.
  If there is an operation running, you must wait until that operation completes and before proceeding with the rollback.

Note: The rollback operation runs for a longer time as the cumulative number of backups of the hosts that are being rolled back together increases. The duration of the rollback can vary depending on the number of profiles and backups associated with a given host.

Related information

Documentation on the NetApp Support Site: mysupport.netapp.com

Performing a rollback on a single host or multiple hosts

You can perform a rollback on a single or multiple SnapManager server hosts by using the command-line interface (CLI).

Before you begin

You must ensure that all the prerequisites for performing a rollback are complete.

Steps

1. To perform a rollback on a single host, enter the following 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

The following example shows the command to roll back all the target databases that are mounted on hostA and a repository database named repoA located on the repository host, repo_host:

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

2. To perform a rollback on multiple hosts, enter the following 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*

   Note: For multiple hosts, enter the host names separated by a comma and ensure that there is no space between the comma and the next host name.

   If you are using Real Application Clusters (RAC) configuration, you must manually roll back all RAC-associated hosts. You can use `-allhosts` to perform a rollback of all the hosts.

Example

The following example shows the command to roll back all the target databases that are mounted on the hosts, hostA, hostB, and a repository database named repoA located on the repository host, repo_host:

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

The hosts, profiles, schedules, backups, and clones that are associated with the profiles of the target databases for the host are reverted to the earlier repository.

Related concepts

Prerequisites for performing a rollback on page 44

Post rollback tasks

You must perform some additional steps after you rollback a repository database and downgrade the SnapManager host from SnapManager 3.2 to SnapManager 3.0, to view the schedules created in the earlier version of the repository database.

Steps

1. Navigate to `cd /opt/NetApp/smo/repositories`.

The `repositories` directory might contain two files for each repository. The file name with the number sign (#) is created using SnapManager 3.1 or later and the file name with the hyphen (-) is created using the SnapManager 3.0.

Example

The file names might be as follows:

- `repository#SMO300a#SMOREPO1#10.72.197.141#1521`
- `repository-smo300a-smorepo1-10.72.197.141-1521`

2. Replace the number sign (#) in the file name with the hyphen (-).

Example

The file name that had the number sign (#), now contains hyphen (-): `repository-SMO300a-SMOREPO1-10.72.197.141-1521`.

Configuring SnapManager

After installing SnapManager, you must perform some additional configuration tasks depending on the environment that you are using.

SnapManager configuration parameters

SnapManager provides a list of configuration parameters that you can edit depending on your requirement. The configuration parameters are stored in the `smo.config` file. However, the `smo.config` file might not contain all the supported configuration parameters. You can add the configuration parameters, depending on your requirement.

The following table lists all the supported SnapManager configuration parameters and also explains when to use these parameters:

Parameters	Description
• `retain.hourly.count` • `retain.hourly.duration` • `retain.monthly.count` • `retain.monthly.duration`	These parameters set the retention policy when you create a profile. For example, you can assign the following values: `retain.hourly.count` = **`12`** `retain.hourly.duration` = **`2`** `retain.monthly.count` = **`2`** `retain.monthly.duration` = **`6`**
`restore.secondaryAccessPolicy`	This parameter defines how SnapManager can access data on secondary storage when it cannot be restored directly by using Protection Manager. The different ways to access the data on secondary storage are as follows: • Direct (default) When `restore.secondaryAccessPolicy` is set to **`direct`**, SnapManager clones the 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. • Indirect If you assign **`indirect`** to `restore.secondaryAccessPolicy`, SnapManager copies data to a temporary volume on primary storage, mounts data from the temporary volume to the host, and then copies data out of the temporary volume into the active environment. The indirect method must be used only if the host does not have direct access to the secondary storage system. This method takes twice as long as the direct method because two copies of the data are made. **Note:** In a Storage Area Network (SAN) with Network File System (NFS) as the protocol, SnapManager does not need to connect directly to secondary storage to perform a restore.

Parameters	Description
`restore.temporaryVolumeName`	This parameter assigns a name to the temporary volume. When SnapManager uses the indirect method for restoring data from secondary storage, it requires a scratch volume on the 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 the restore command that uses the indirect method. For example, you can assign the following values: `restore.temporaryVolumeName` = **`smo_temp_volume`**
`retain.alwaysFreeExpiredBackups`	This parameter allows SnapManager to free backups when they expire and when a fast restore is performed, even if data protection is not configured. This parameter frees the protected backups that expire and deletes the unprotected backups that expire. The possible values that you can assign are as follows: • True If you assign **`true`** to `retain.alwaysFreeExpiredBackups`, SnapManager frees the expired backups regardless of whether the backups are protected. The backups are deleted either when they are not protected or if the protected copies on secondary storage have also expired. • False If you assign **`false`** to `retain.alwaysFreeExpiredBackups`, SnapManager frees the expired backups that are protected.
`host.credentials.persist`	This parameter allows SnapManager to store host credentials. By default, the host credentials are not stored. However, host credentials need to be stored if you have a custom script that runs on a remote clone and requires access to a remote server. You can enable storing of host credentials by assigning **`true`** to `host.credentials.persist`. SnapManager encrypts and saves the host credentials.
`restorePlanMaxFilesDisplayed`	This parameter enables you to define the maximum number of files to be displayed in the restore preview. By default, SnapManager displays a maximum of 20 files in the restore preview. However, you can change to a value greater than 0. For example, you can assign the following value: • `restorePlanMaxFilesDisplayed` = **`30`** **Note:** If you specify an invalid value, the default number of files are displayed.

Parameters	Description
`snapshot.list.timeout.min`	This parameter enables you to define the time in minutes for which SnapManager must wait for the `snap list` command to execute when you are performing any SnapManager operations. By default, SnapManager waits for 30 minutes. However, you can change to a value greater than 0. For example, you can assign the following value: • `snapshot.list.timeout.min` = **`40`** **Note:** If you specify an invalid value, the default value is used. For any SnapManager operation, if the `snap list` command execution time exceeds the value assigned to `snapshot.list.timeout.min`, the operation fails with a timeout error message.
`pruneIfFileExistsInOtherDestination`	This pruning parameter enables you to define the destination of the archive logs files. The archive log files are stored in multiple destinations. While pruning archive log files, SnapManager needs to know the destination of the archive log files. The possible values that you can assign are as follows: • When you want to prune the archive log files from a specified destination, you must assign **`false`** to `pruneIfFileExistsInOtherDestination`. • When you want to prune the archive log files from an external destination, you must assign **`true`** to `pruneIfFileExistsInOtherDestination`.
`prune.archivelogs.backedup.from.otherdestination`	This pruning parameter enables you to prune the archive log files backed up from the specified archive log destinations or backed up from external archive log destinations. The possible values that you can assign are as follows: • When you want to prune the archive log files from the specified destinations and if the archive log files are backed up from the specified destinations by using `-prune-dest`, you must assign **`false`** to `prune.archivelogs.backedup.from.otherdestination`. • When you want to prune the archive log files from specified destinations and if the archive log files are backed up at least once from any one of the other destinations, you must assign **`true`** to `prune.archivelogs.backedup.from.otherdestination`.
`maximum.archivelog.files.toprune.atATime`	This pruning parameter enables you to define the maximum number of archive log files that you can prune at a given time. For example, you can assign the following value: `maximum.archivelog.files.toprune.atATime` = **`998`** **Note:** The value that can be assigned to `maximum.archivelog.files.toprune.atATime` must be less than 1000.
`archivelogs.consolidate`	This parameter allows SnapManager to free the duplicate archive log backups if you assign **`true`** to `archivelogs.consolidate`.

Parameters	Description
`suffix.backup.label.with.logs`	This parameter enables you to specify the suffix that you want to add to differentiate the label names of the data backup and the archive log backup. For example, when you assign **`logs`** to `suffix.backup.label.with.logs`, `_logs` is added as a suffix to the archive log backup label. The archive log backup label would then be `arch_logs`.
`backup.archivelogs.beyond.missingfiles`	This parameter allows SnapManager to include the missing archive log files in the backup. The archive log files that do not exist in the active file system are not included in the backup. If you want to include all of the archive log files, even those that do not exist in the active file system, you must assign **`true`** to `backup.archivelogs.beyond.missingfiles`. You can assign **`false`** to ignore the missing archive log files.
`srvctl.timeout`	This parameter enables you to define the timeout value for the `srvctl` command. **Note:** The Server Control (SRVCTL) is a utility to manage RAC instances. When SnapManager takes more time to execute the `srvctl` command than the timeout value, the SnapManager operation fails with this error message: `Error: Timeout occurred while executing command: srvctl status`.
`snapshot.restore.storageNameCheck`	This parameter allows SnapManager to perform the restore operation with Snapshot copies that were created before migrating from Data ONTAP operating in 7-Mode to clustered Data ONTAP. The default value assigned to the parameter is **`false`**. If you have migrated from Data ONTAP operating in 7-Mode to clustered Data ONTAP but want to use the Snapshot copies created before migration, set `snapshot.restore.storageNameCheck=`**`true`**.
`services.common.disableAbort`	This parameter disables cleanup upon failure of long-running operations. You can set `services.common.disableAbort=`**`true`**. For example, if you are performing a clone operation that runs long and then fails because of an Oracle error, you might not want to clean up the clone. If you set `services.common.disableAbort=`**`true`**, the clone will not be deleted. You can fix the Oracle issue and restart the clone operation from the point where it failed.

Parameters	Description
• `backup.sleep.dnfs.layout` • `backup.sleep.dnfs.secs`	These parameters activate the sleep mechanism in the Direct NFS (dNFS) layout. After you create the backup of control files using dNFS or a Network File System (NFS), SnapManager tries to read the control files, but the files might not be found. To enable the sleep mechanism, ensure that `backup.sleep.dnfs.layout=`**`true`**. The default value is **`true`**. When you enable the sleep mechanism, you must assign the sleep time to `backup.sleep.dnfs.secs`. The sleep time assigned is in seconds and the value depends upon your environment. The default value is **`5`** seconds. For example: • `backup.sleep.dnfs.layout=`**`true`** • `backup.sleep.dnfs.secs=`**`2`**
• `override.default.backup.pattern` • `new.default.backup.pattern`	When you do not specify the backup label, SnapManager creates a default backup label. These SnapManager parameters allows you to customize the default backup label. To enable customization of the backup label, ensure that the value of `override.default.backup.pattern` is set to **`true`**. The default value is **`false`**. To assign the new pattern of the backup label, you can assign keywords such as database name, profile name, scope, mode, and host name to `new.default.backup.pattern`. The keywords should be separated using an underscore. For example, `new.default.backup.pattern=`**`dbname_profile_hostname_scope_mode`**. **Note:** The timestamp is included automatically at the end of the generated label.
`allow.underscore.in.clone.sid`	Oracle supports usage of the underscore in clone SID from Oracle 11gR2. This SnapManager parameter enables you to include an underscore in the clone SID name. To include an underscore in the clone SID name, ensure that the value of `allow.underscore.in.clone.sid` is set to **`true`**. The default value is **`true`**. If you are using an Oracle version earlier than Oracle 11gR2 or if you do not want to include an underscore in the clone SID name, set the value to **`false`**.
`oracle.parameters.with.comma`	This parameter enables you to specify all the Oracle parameters that have comma (,) as the value. While performing any operation SnapManager uses `oracle.parameters.with.comma` to check all the Oracle parameters and skip the splitting of the values. For example, if the value of *`nls_numeric_characters=,`*, then specify `oracle.parameters.with.comma=`*`nls_numeric_characters`*. If there are multiple Oracle parameters with comma as the value, you must specify all the parameters in `oracle.parameters.with.comma`.

Parameters	Description
• `archivedLogs.exclude` • `archivedLogs.exclude.fileslike` • `<db-unique-name>.archivedLogs.exclude.fileslike`	These parameters allow SnapManager to exclude the archive log files from the profiles and backups if the database is not on a Snapshot copy-enabled storage system and you want to perform SnapManager operations on that storage system. **Note:** You must include the exclude parameters in the configuration file before creating a profile. The values assigned to these parameters can either be a top-level directory or a mount point where the archive log files are present or a subdirectory. 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 is not included in the dataset that is created in Protection Manager. When there are multiple archive log files to be excluded from the host, you must separate the archive log file paths by using commas. To exclude archive log files from being included in the profile and being backed up, you must include one of the following parameters: • `archivedLogs.exclude` to specify a regular expression for excluding archive log files from all profiles or backups. The archive log files matching the regular expression are excluded from all the profiles and backups. For example, you can set `archivedLogs.exclude` = `/arch/logs/on/local/disk1/.*,/arch/logs/on/local/disk2/.*`. For ASM databases, you can set `archivedLogs.exclude` = `\\+KHDB_ARCH_DEST/khdb/archivelog/.*,\\+KHDB_NONNAARCHTWO/khdb/archivelog/.*`. • `archivedLogs.exclude.fileslike` to specify an SQL expression for excluding archive log files from all profiles or backups. The archive log files matching the SQL expression are excluded from all the profiles and backups. For example, you can set `archivedLogs.exclude.fileslike` = `/arch/logs/on/local/disk1/%,/arch/logs/on/local/disk2/%`.
	• `<db-unique-name>.archivedLogs.exclude.fileslike` to specify an SQL expression for excluding archive log files only from the profile or the backup created for the database with the specified *`db-unique-name`*. The archive log files matching the SQL expression are excluded from the profile and backups. For example, you can set `mydb.archivedLogs.exclude.fileslike` = `/arch/logs/on/local/disk1/%,/arch/logs/on/local/disk2/%`.

Editing the configuration parameters

Depending on your environment, you can change the default values assigned to the configuration parameter.

Steps

1. Open the configuration file from the following default location:

   `default installation location/properties/smo.config`

2. Change the default values of the configuration parameters.

   Note: You can also add supported configuration parameters that are not included in the configuration file, and assign values to them.

3. Restart the SnapManager for Oracle server.

Configuring SnapDrive for UNIX for an active/active Veritas SFRAC environment

If you have included the `host-cluster-sw-restore-warn` parameter in `snapdrive.conf` and have assigned the value **on**, you must change the value to support the restore operation in the active/active Veritas Storage Foundation for Oracle RAC (SFRAC) environment.

About this task

When you are using the active/active Veritas Storage Foundation for Oracle RAC (SFRAC) environment, if the `host-cluster-sw-restore-warn` parameter is set to **on**, a warning message is displayed and the restore operation is stopped. If you want to perform the restore operation in an active/active Veritas SFRAC environment, you must set `host-cluster-sw-restore-warn` to **`off`**.

For information on `snapdrive.conf`, see SnapDrive documentation.

Steps

1. Log in as the root user.
2. Open the `snapdrive.conf` file by using a text editor.
3. Change the value of `host-cluster-sw-restore-warn` to **`off`**.

After you finish

After configuring, restart the SnapDrive for UNIX server.

Related information

Documentation on the NetApp Support Site: mysupport.netapp.com

Configuring SnapManager to support the Veritas SFRAC environment

When SnapManager is installed on Solaris, you can configure SnapManager to support the Veritas Storage Foundation for Oracle RAC (SFRAC) environment.

Before you begin

- The host must have Solaris, host utilities, and Veritas installed.

Steps

1. Create a shared disk group and a file system for SnapManager by using SnapDrive for UNIX so that the file systems are concurrently mounted on both nodes of the Real Application Clusters (RAC).

   For information about how to create a shared disk group and file system, see SnapDrive documentation.

2. Install and configure the Oracle database that is to be mounted on the shared file systems.
3. Start a database instance on any one node of the RAC.

Related information

Documentation on the NetApp Support Site: mysupport.netapp.com

Ensuring that ASM discovers imported disks

If you are using Automatic Storage Management (ASM) in an NFS environment, after installing SnapManager, you must ensure that ASM can discover the disks imported by SnapManager. You can do this by adding the path of the ASM directory to the `ASM_DISKSTRING` parameter.

About this task

You can use Oracle tools to edit the `ASM_DISKSTRING` parameter. For information about editing `ASM_DISKSTRING`, see the Oracle documentation.

The ASM disk path `/opt/NetApp/smo/mnt/*/*/disk*'` must be added to the existing path defined in the `ASM_DISKSTRING` parameter. For example, if the path defined in `ASM_DISKSTRING` was `/mnt/my-asm-disks/dir1/disk*`, after adding the ASM disk path, the updated path will be `'/mnt/my-asm-disks/dir1/disk*,/opt/NetApp/smo/mnt/*/*/disk*'`.

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

- The first asterisk (`*`) indicates the name generated by SnapManager for the root mount point.
- The second `*` indicates the directory within the mount point.
- The third `*` indicates the name of the NFS file.

You must ensure that the `*` matches the topology of your NFS file system, if the disk is mounted in the directories under `/opt/NetApp/smo/mnt/<smo-generated-name>/`.

Steps

1. If you are using ASM disks with NFS in the Network Attached Storage (NAS) environment, edit the `ASM_DISKSTRING` parameter so that it points to the current ASM directory path.

   #### Example

   If the ASM disks mount point is `/mnt/my-asm-disks/*/disk*`, after editing `ASM_DISKSTRING`, the updated path is `/opt/NetApp/smo/mnt/my-asm-disks-20081012/disk1.nfs`. The `ASM_DISKSTRING` parameter is in the form `/opt/NetApp/smo/mnt/*/disk*`.

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

   After editing the `ASM_DISKSTRING` parameter, the results of ASM discovering the disks imported by SnapManager are as follows:

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

   The `ASM_DISKSTRING` parameter is in the form `/opt/NetApp/smo/mnt/*/*/disk*`.

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

2. If you are using ASM disks in the Storage Area Network (SAN) environment, depending on the environment perform one of the following:

If you are using ASM disks with...	Then...
ASMLib over FCP and iSCSI on Linux	Change the permission of the Oracle software owner and primary group of the user by using only the character device. The `ASM_DISKSTRING` path must be `ASM DISKSTRING = ORCL:*`.
FCP and iSCSI on AIX	Add the path name for the `ASM_DISKSTRING` parameter until the ASM directory path. The `ASM_DISKSTRING` path must be `ASM DISKSTRING = /dev/hdsk/*`, where * indicates the ASM disk name.
FCP and iSCSI on Solaris	Add the path name for the `ASM_DISKSTRING` parameter until the ASM directory path. The `ASM_DISKSTRING` path must be `ASM DISKSTRING = /dev/rdsk/*`, where * indicates the ASM disk name.

Related information

Oracle Documentation: www.oracle.com/technetwork/indexes/documentation/index.html

Security and credential management

You can manage security in SnapManager by applying user authentication and role-based access control (RBAC). The user authentication method allows you to access resources such as repositories, hosts, and profiles. RBAC allows you to restrict the operations that SnapManager can perform against the volumes and LUNs containing the data files in your database.

When you perform an operation using either the command-line interface (CLI) or graphical user interface (GUI), SnapManager retrieves the credentials set for repositories and profiles. SnapManager saves credentials from previous installations.

The repository and profiles can be secured with a password. A credential is the password configured for the user for an object, and the password is not 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.
  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, and profiles).

You can manage role-based access by performing the following tasks:

- Enable RBAC for SnapManager by using SnapDrive.
- Assign users to roles and set role capabilities by using the Operations Manager console.
- Optionally, enable SnapManager to store encrypted passwords by editing the `smo.config` file.

If Protection Manager is installed, access to the features is affected in the following ways:

- If Protection Manager is installed, when you create a database profile, SnapManager creates a dataset and populates the dataset with the volumes that contain the database files.
  After a backup operation, SnapManager keeps the dataset contents synchronized with the database files.
- If Protection Manager is not installed, SnapManager cannot create a dataset and you cannot set protection on profiles.

What user authentication is

In addition to using role-based access control (RBAC), SnapManager authenticates the user by using an operating system (OS) login on the host where the SnapManager server is running. You can enable 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, you are authenticated by the OS credentials. You are not prompted for a password because you are already logged in to the host where the SnapManager server is running.
- If the SnapManager client and the SnapManager server are on different hosts, SnapManager needs to authenticate you with both OS credentials. SnapManager prompts you for passwords for

any operation, if you have not saved your OS credentials in your SnapManager user credential cache. If you enter the `smo credential set -host` command, you save the OS credentials in your SnapManager credential cache file and so SnapManager does not prompt for the password for any operation.

If you are authenticated with the SnapManager server, you are 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 you execute a clone operation, you should be able to log in to 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 does not 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 an 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, and profiles).

About role-based access control

Role-based access control (RBAC) lets you control who has access to SnapManager operations. RBAC allows administrators to manage groups of users by defining roles and assigning users to those roles. You might want to use SnapManager RBAC in environments where RBAC is already in place.

RBAC includes the following components:

- Resources: Volumes and LUNs that hold the datafiles that make up your database.
- Capabilities: Types of operations that can be performed on a resource.
- Users: People to whom you grant capabilities.
- Roles: A set of resources and capabilities allowed on resources. You assign a specific role to a user who should perform those capabilities.

You enable RBAC in SnapDrive. You can then configure specific capabilities per role in the Operations Manager Web graphical user interface or command-line interface. RBAC checks occur in the DataFabric Manager server.

The following table lists some roles and their typical tasks, as set in Operations Manager.

Role	Typical tasks
Oracle database administrator	• Creating, maintaining, and monitoring an Oracle database that resides on a host • Scheduling and creating database backups • Ensuring that backups are valid and can be restored • Cloning databases
Server administrator	• Setting up storage systems and aggregates • Monitoring volumes for free space • Provisioning storage on requests from users • Configuring and monitoring disaster recovery mirroring
Storage architect	• Making architectural decisions on storage • Planning storage capacity growth • Planning disaster recovery strategies • Delegating capabilities to members of the team

If RBAC is in use (meaning that Operations Manager is installed and RBAC is enabled in SnapDrive), the storage administrator needs to assign RBAC permissions on all of the volumes and storage systems for the database files.

Enabling role-based access control

SnapManager role-based access control (RBAC) is enabled using SnapDrive. Upon installation of SnapDrive, RBAC is disabled by default. After you enable RBAC in SnapDrive, SnapManager then performs operations with RBAC enabled.

About this task

In SnapDrive the `snapdrive.config` file sets many options, one of which enables RBAC.

For details about SnapDrive, see SnapDrive documentation.

Steps

1. DisplayOpen the `snapdrive.conf` file in an editor.
2. In the snapdrive.conf file, to enable RBAC, change the "rbac-method" parameter from `native` to `dfm`.

   The default value for this parameter is `native`, which disables RBAC.

Related information

Documentation on the NetApp Support Site: mysupport.netapp.com

Setting role-based access control capabilities and roles

After you enable role-based access control (RBAC) for SnapManager using SnapDrive, you can add RBAC capabilities and users to roles to perform SnapManager operations.

Before you begin

You must create a group in the Data Fabric Manager server and add the group to both primary and secondary storage systems. Run the following commands:

- **`dfm group create`** ***`smo_grp`***
- **`dfm group add`** ***`smo_grp primary_storage_system`***
- **`dfm group add`** ***`smo_grp secondary_storage_system`***

About this task

You can use either the Operations Manager web interface or the Data Fabric Manager server command-line interface (CLI) to modify RBAC capabilities and roles.

The table lists the RBAC capabilities required to perform SnapManager operations:

SnapManager operations	**RBAC capabilities required when data protection is not enabled**	**RBAC capabilities required when data protection is enabled**
Profile create or profile update	SD.Storage.Read (smo_grp)	SD.Storage.Read (SMO_*`profile`* dataset)
Profile protection	DFM.Database.Write (smo_grp) SD.Storage.Read (smo_grp) SD.Config.Read (smo_grp) SD.Config.Write (smo_grp) SD.Config.Delete (smo_grp) GlobalDataProtection	None
Backup create	SD.Storage.Read (smo_grp) SD.Snapshot.Write (smo_grp) SD.Snapshot.Read (smo_grp) SD.Snapshot.Delete (smo_grp)	SD.Storage.Read (SMO_*`profile`* dataset) SD.Snapshot.Write (SMO_*`profile`* dataset) SD.Snapshot.Read (SMO_*`profile`* dataset) SD.Snapshot.Delete (SMO_*`profile`* dataset)

SnapManager operations	RBAC capabilities required when data protection is not enabled	RBAC capabilities required when data protection is enabled
Backup create (with DBverify)	SD.Storage.Read (smo_grp) SD.Snapshot.Write (smo_grp) SD.Snapshot.Read (smo_grp) SD.Snapshot.Delete (smo_grp) SD.SnapShot.Clone (smo_grp)	SD.Storage.Read (SMO_*profile* dataset) SD.Snapshot.Write (SMO_*profile* dataset) SD.Snapshot.Read (SMO_*profile* dataset) SD.Snapshot.Delete (SMO_*profile* dataset) SD.SnapShot.Clone (SMO_*profile* dataset)
Backup create (with RMAN)	SD.Storage.Read (smo_grp) SD.Snapshot.Write (smo_grp) SD.Snapshot.Read (smo_grp) SD.Snapshot.Delete (smo_grp) SD.SnapShot.Clone (smo_grp)	SD.Storage.Read (SMO_*profile* dataset) SD.Snapshot.Write (SMO_*profile* dataset) SD.Snapshot.Read (SMO_*profile* dataset) SD.Snapshot.Delete (SMO_*profile* dataset) SD.SnapShot.Clone (SMO_*profile* dataset)
Backup restore	SD.Storage.Read (smo_grp) SD.Snapshot.Write (smo_grp) SD.Snapshot.Read (smo_grp) SD.Snapshot.Delete (smo_grp) SD.SnapShot.Clone (smo_grp) SD.Snapshot.Restore (smo_grp)	SD.Storage.Read (SMO_*profile* dataset) SD.Snapshot.Write (SMO_*profile* dataset) SD.Snapshot.Read (SMO_*profile* dataset) SD.Snapshot.Delete (SMO_*profile* dataset) SD.SnapShot.Clone (SMO_*profile* dataset) SD.Snapshot.Restore (SMO_*profile* dataset)
Backup delete	SD.Snapshot.Delete (smo_grp)	SD.Snapshot.Delete (SMO_*profile* dataset)
Backup verify	SD.Storage.Read (smo_grp) SD.Snapshot.Read (smo_grp) SD.Snapshot.Clone (smo_grp))	SD.Storage.Read (SMO_*profile* dataset) SD.Snapshot.Read (SMO_*profile* dataset) SD.Snapshot.Clone (SMO_*profile* dataset)

SnapManager operations	RBAC capabilities required when data protection is not enabled	RBAC capabilities required when data protection is enabled
Backup mount	SD.Storage.Read (smo_grp) SD.Snapshot.Read (smo_grp) SD.Snapshot.Clone (smo_grp)	SD.Storage.Read (SMO_*profile* dataset) SD.Snapshot.Read (SMO_*profile* dataset) SD.Snapshot.Clone (SMO_*profile* dataset)
Backup unmount	SD.Snapshot.Clone (smo_grp)	SD.Snapshot.Clone (SMO_*profile* dataset)
Clone create	SD.Storage.Read (smo_grp) SD.Snapshot.Read (smo_grp) SD.SnapShot.Clone (smo_grp)	SD.Storage.Read (SMO_*profile* dataset) SD.Snapshot.Read (SMO_*profile* dataset) SD.SnapShot.Clone (SMO_*profile* dataset)
Clone delete	SD.Snapshot.Clone (smo_grp)	SD.Snapshot.Clone (SMO_*profile* dataset)
Clone split	SD.Storage.Read (smo_grp) SD.Snapshot.Read (smo_grp) SD.SnapShot.Clone (smo_grp) SD.Snapshot.Delete (smo_grp) SD.Storage.Write (smo_grp)	SD.Storage.Read (SMO_*profile* dataset) SD.Snapshot.Read (SMO_*profile* dataset) SD.SnapShot.Clone (SMO_*profile* dataset) SD.Snapshot.Delete (SMO_*profile* dataset) SD.Storage.Write (SMO_*profile* dataset)

For details about defining RBAC capabilities, see the *OnCommand Unified Manager Operations Manager Administration Guide*.

Steps

1. Access the Operations Manager console.
2. From the Setup menu, select **Roles**.
3. Select an existing role or create a new one.
4. To assign operations to your database storage resources, click **Add capabilities**.
5. On the Edit Role Settings page, to save your changes to the role, click **Update**.

Related information

OnCommand Unified Manager Operations Manager Administration Guide: mysupport.netapp.com/documentation/productsatoz/index.html

Storing encrypted passwords for custom scripts

By default, SnapManager does not store host credentials in the user credentials cache. However, you can change this. You can edit the `smo.config` file to allow storing of host credentials.

About this task

The `smo.config` file is located at `<default installation location>/properties/smo.config`

Steps

1. Edit the `smo.config` file.
2. Set *`host.credentials.persist`* to **`true`**.

Authorizing access to the repository

In addition to role-based access control (RBAC), SnapManager enables you to set credentials for database users to access the repository. Using credentials, you can restrict or prevent access to the SnapManager hosts, repositories, profiles, and databases.

About this task

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

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

Step

1. Enter the following command:

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

Authorizing access to profiles

In addition to role-based access control (RBAC), SnapManager enables you to set a password for a profile to prevent unauthorized access.

Step

1. Enter the following command:

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

Related references

The smo credential set command on page 264

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 264

Clearing user credentials for all hosts, repositories, and profiles

You can clear the cache of your credentials for resources (hosts, repositories, and profiles). This deletes all of the resource credentials for the user running the command. After clearing the cache, you must authenticate your credentials again to gain access to these secured resources.

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 262

Setting credentials after clearing the credential cache

After clearing the cache to remove the stored user credentials, you can set the credentials for the hosts, repositories, and profiles.

About this task

You must ensure that you set the same user credentials for the repository, profile host, and profile that you had given earlier. An encrypted credentials file is created while setting the user credentials.

The credentials file is located at `/root/.netapp/smo/3.3.0`.

From the SnapManager graphical user interface (GUI), if there is no repository under Repositories, perform the following steps:

Steps

1. Click **Tasks > Add Existing Repository** to add an existing repository.
2. Perform the following steps to set the credentials for repository:
   a. Right-click the repository and select **Open**.
   b. In the **Repository Credentials Authentication** window, enter the user credentials.
3. Perform the following steps to set the credentials for host:
   a. Right-click the host under the repository and select **Open**.
   b. In the **Host Credentials Authentication** window, enter the user credentials.
4. Perform the following steps to set the credentials for profile:
   a. Right-click the profile under the host and select **Open**.
   b. In the **Profile Credentials Authentication** window, enter the user credentials.

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 262

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 -namehost_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

You must create a profile in SnapManager for the database on which you want to perform an operation. You must select the profile and then select the operation that you want to perform.

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.
  You can also create profiles to separate the archive log backups from the data file backups.
- Verify profiles.
- Update profiles.
- Delete profiles.

About profiles and authentication

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

- Oracle authentication with a user name, password, and port
- Operating system (OS) authentication with no user name, password, or port.
  For OS authentication, you must enter the OS account user and group information.

  Note: To use OS authentication for the Real Application Cluster (RAC) databases, the SnapManager server must be running on each node of the RAC environment and the database password must be the same for all Oracle instances in a RAC environment. SnapManager uses the database user name and password to connect to every RAC instance in the profile.

- Database authentication when `sqlnet.authentication_services` is set to **`NONE`**. SnapManager then uses the database user name and password for all the connections to the target database.

  Note: To use database authentication for an Automatic Storage Management (ASM) instance, you must enter the user name and password that you use to log in to the ASM instance.

You can set `sqlnet.authentication_services` to **`NONE`** only in the following environments:

Database layout	Oracle version	Is database authentication supported for the target database	Is database authentication supported for the ASM instance
Any non-ASM and non-RAC database	Oracle 10g and Oracle 11g (lesser than 11.2.0.3)	Yes	No
Stand-alone ASM database on UNIX	Oracle 11.2.0.3 and later	Yes	Yes
ASM instance on RAC database on UNIX	Oracle 11.2.0.3	No	No
RAC database on NFS	Oracle 11.2.0.3	Yes	No

Note: After you disable `sqlnet.authentication_services` and change the authentication method to database authentication, you must set `sqlnet.authentication_services` to **`NONE`**.

If you are accessing a profile for the first time, you must enter your profile password. After you enter your credentials, you can view the database backups within the profile.

Related concepts

What profiles are on page 14

Creating profiles

When creating profiles, you can assign a particular Oracle database user account to the profile. You can set the retention policy for the profile, enable backup protection to secondary storage for all the backups using this profile, and set the retention count and duration for each retention class.

About this task

If you do not provide the values of the `-login`, `-password`, and `-port` parameters of the database, the operating system (OS) authentication mode uses the default credentials.

While creating a profile, SnapManager performs a restore eligibility check to determine the restore mechanism that can be used to restore the database. If the database is on a qtree and the parent volume is not eligible for a fast or volume-based restore, the analysis might be wrong.

SnapManager (3.2 or later) enables you to separate archive log files from the data files while creating a new profile or updating an existing profile. After you have separated the backup using the profile, you can either create only the data files-only backup or archive log-only backup of the database. You can use the new profile or the updated profile to create the backup containing both the data files and archive log files. However, you cannot use the profile to create the full backup or revert the settings.

Profiles for creating full and partial backups

You can create profiles to create the full database backup containing the data files, control files, and archive log files and partial database backup containing specified data files or tablespaces, all the control files, and all the archive log files. SnapManager does not allow you to create separate archive log backups using the profiles created for full and partial backups.

Profiles for creating data files-only backups and archivelogs-only backups

When you create a new profile, you can include `-separate-archivelog-backups` to separate the archive log backup from the data file backup. You can also update the existing profile to separate the archive log backup from the data file backup.

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

- Create an archive log backup
- Delete an archive log backup
- Mount an archive log backup
- Free an archive log backup

While creating the profile to separate archive log backups from the data files 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` is displayed. Even if you create backups for this database, the archive log files are not available in the database backups.

Note: If you encounter an error while creating a profile, use the `smo system dump` command. After you create a profile, if you encounter an error, use the `smo operation dump` and `smo profile dump` commands.

Step

1. To create a profile with a user name, password, and port (Oracle authentication), enter the following 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*][-asminstance -asmusername *asminstance_username* -asmpassword *asminstance_password*]] [-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 Real Application Clusters (RAC) environments, when creating a new profile you must provide the value of the `db_unique_name` parameter as *`db_dbname`*.

   You can also include other options when creating profiles, depending on how you want to access the database.

If...	Then...
You want to use operating system authentication to create the profile	Specify the variables for an operating system account in the DBA group (typically the account used to install Oracle). Instead of adding the user name, password, and port, specify the following: • `-osaccount`*`account_name`* as the name of the operating system account • `-osgroup` *`osgroup`* as the group associated with the operating system account
You want to use Automatic Storage Management (ASM) instance authentication to create the profile	Specify the credentials for ASM instance authentication. • `-asmusernameasmintance_username` is the user name used to log in to the ASM instance. • `-asmpassword` *`asminstance_password`* is the password used to log in to the ASM instance.

If...	Then...
You want to use database authentication to create a profile	Specify the database login details. If the password contains special characters such as exclamation point (!), dollar sign ($), or grave accent (`), then SnapManager does not allow you to create the database authenticated profile from the command-line interface (CLI).
You are using a catalog as the Oracle Recovery Manager (RMAN) repository	Specify the following options and variables: • `-tnsname` *`tnsname`* as the tnsname defined in the `tnsnames.ora` file. • `-login -username` *`username`* as the user name required to connect to the RMAN catalog. If not specified, SnapManager uses the operating system authentication information. You cannot use operating system authentication with RAC databases. • `-password` *`password`* as the RMAN password required to connect to the RMAN catalog.
You are using the control file as the RMAN repository	Specify the `-controlfile` option.
You want to specify a backup retention policy for backups	Specify either the retention count or duration for a retention class, or both. 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 enable backup protection for the profile	Specify the following options and variables: • `-protect` enables backup protection. If you are using Data ONTAP operating in 7-Mode, this option creates an application dataset in the Data Fabric Manager (DFM) server and adds members related to the database, data file, control files, and archive logs. If the dataset already exists, the same dataset is reused when a profile is created. • `-protection-policy` *`policy`* allows you to specify the protection policy. If you are using Data ONTAP operating in 7-Mode and SnapManager is integrated with Protection Manager, you must specify one of the Protection Manager policies. **Note:** To list the possible protection policies, use the `smo protection-policy list` command. If you are using clustered Data ONTAP, you must select either *SnapManager_cDOT_Mirror* or *SnapManager_cDOT_Vault*. **Note:** The profile create operation fails in the following scenarios: ◦ If you are using clustered Data ONTAP but select Protection Manager policy ◦ If you are using Data ONTAP operating in 7-Mode but select either *SnapManager_cDOT_Mirror* or *SnapManager_cDOT_Vault* policy ◦ If you created SnapMirror relationship but selected *SnapManager_cDOT_Vault* policy or created SnapVault relationship but selected *SnapManager_cDOT_Mirror* policy ◦ If you have not created either SnapMirror or SnapVault relationship but selected either *SnapManager_cDOT_Vault* or *SnapManager_cDOT_Mirror* policy • `-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 can be set later by the `smo profile update` command or set by the storage administrator by using Protection Manager console.

If...	Then...
You want to enable email notification for the completion status of the database operations	Specify the following options and variables: • `-summary-notification` enables you to configure a summary email notification for multiple profiles under a repository database. • `-notification` enables you to receive an email notification for the completion status of the database operation for a profile. • `-success -email`*`email_address2`* enables you to receive an email notification on the successful database operation performed by using a new or existing profile. • `-failure -email`*`email_address2`* enables you to receive an email notification on the failed database operation performed by using a new or existing profile. • `-subject`*`subject_text`* specifies the subject text for the email notification while creating a new profile or an existing profile. If the notification settings are not configured for the repository and you try to configure profile or summary notifications by using the CLI, the following message is logged in the console log: `SMO-14577: Notification Settings not configured.` If you have configured the notification settings and you try to configure summary notification by using the CLI without enabling summary notification for the repository, the following message is shown in the console log: `SMO-14575: Summary notification configuration not available for this repository`
You want to backup archive log files separately from data files	Specify the following options and variables: • `-separate-archivelog-backups` enables you to separate the archive log backup from the datafile backup. • `-retain-archivelog-backups` sets the retention duration for archive log backups. You must specify a positive retention duration. The archive log backups are retained based on the archive log retention duration. The data files 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 data files 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 create an online data files backup and archive logs backup together for cloning. When this option is set, whenever you create an online data files backup, the archive logs backups are created along with the data files 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 the `-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 perform a volume-based restore operation on the files specified in the profile.

Related concepts

How to collect dump files on page 322

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

Variable name	Description	Example value
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
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.

Renaming profiles

SnapManager enables you to rename the profile when you update the profile. The SnapManager capabilities that are set on the profile and the operations that can be performed before renaming are retained for the renamed profile.

Before you begin

- You must ensure that there are no SnapManager operations running on the profile while renaming the profile.

About this task

You can rename the profile from both the SnapManager command-line interface (CLI) and graphical user interface (GUI). While updating the profile, SnapManager verifies and updates the profile name in the repository.

Note: SnapManager does not support renaming the profile in the Multi-profile update window.

When you provide a new profile name, the new profile name is added in the client-side credential cache and the earlier profile name is removed. When you rename the profile from a client, the credential cache of only that client is updated. You need to execute the `smo profile sync` command from each of the clients to update the new credential cache with the new profile name.

You can set the password for the profile by using the `smo credential set` command.

If the profile name was included in a Snapshot copy naming pattern, when you rename a profile, the new name for the profile gets updated. All the SnapManager operations that are performed on the

profile use the new profile name. The backups created with earlier profile continue to have the earlier profile name and are used to perform other SnapManager operations.

If you are performing rolling upgrade of SnapManager server hosts, you must ensure that you perform the complete upgrade before renaming the profile.

The new name for the profile is updated only from the SnapManager client from which the request is made. The SnapManager clients that are connected to the SnapManager server are not notified about the change in profile name. You can check the operation log to know about the change in the profile name.

Note: If a scheduled backup operation begins at the time of renaming the profile, then the scheduled operation fails.

Step

1. Enter the following command:

   smo profile update -profile *profile* [-new-profile *new_profile_name*]

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 290

Resetting the profile password

You can reset the profile password if you do not remember the password that you had provided while creating the profile.

Before you begin

- You must ensure that the SnapManager server is running on the repository database.
- You must have the root user credentials of the host on which the repository database is residing.
- You must ensure that the profile is not in use for any operation when the password is being reset for that profile.

About this task

You can reset the password from either the SnapManager CLI or GUI. While resetting the password, SnapManager queries the SnapManager server on the repository host to identify the operating system for the repository host. You must enter the authorized user credentials for connecting to the repository host. The SnapManager server authenticates users with their root credentials on the repository database. When the authentication is successful, SnapManager resets the profile password on the SnapManager server with the new password.

Note: SnapManager does not maintain the history of the password reset operations.

Step

1. Reset the profile password by entering the following command:

   smo password reset -profile *profile* [-profile-password *profile_password*] [-repository-hostadmin-password *admin_password*]

Authorizing access to profiles

In addition to role-based access control (RBAC), SnapManager enables you to set a password for a profile to prevent unauthorized access.

Step

1. Enter the following command:

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

Related references

The smo credential set command on page 264

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 294

Updating profiles

You can update the profiles to modify the profile password, the number of backups to retain, access to the database, the operating system (OS) authentication to database authentication and vice versa, and information about the host. If the Oracle database password information changes, you must also change that information in the profile.

About this task

If protection policy is enabled on the profile, you cannot change the policy by using SnapManager. The storage administrator must change the policy by using the Protection Manager's console.

SnapManager (3.2 or later) enables you to update the profile to separate archive log backups from the data file backups by 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 you create an online data files backup, the archive logs backups are immediately created along with the data files.

Steps

1. Enter the following command:

   smo profile update -profile *profile* [-new-profile *new_profile_name*] [-profile-password *profile_password*][-database -dbname*db_dbname* -host *db_host* [-sid *db_sid*] [-login -username*db_username* -password *db_password*-port *db_port*][-asminstance -asmusername *asminstance_username* -asmpassword *asminstance_password*]] [{-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 you want to...	Then...
Change the profile to use operating system authentication	Instead of adding the user name, password, and port, specify the following: • `-osaccount` *`account_name`* as the name of the operating system account • `-osgroup` *`osgroup`* as the group associated with the operating system account, typically the account used to install Oracle
Use Automatic Storage Management (ASM) instance authentication to create the profile	Specify the credentials for ASM instance authentication. • `-asmusernameasminstance_username` is the user name used to log in to the ASM instance. • `-asmpassword` *`asminstance_password`* is the password used to log in to the ASM instance.
Use a catalog as the Oracle Recovery Manager (RMAN) repository, or you want to remove RMAN	Specify the following options and variables: • `-tnsname` *`tnsname`* as the tnsname defined in the `tnsnames.ora` file. • `-login -username` *`username`* as the user name required to connect to the RMAN catalog. If not specified, SnapManager uses the operating system authentication information. You cannot use operating system authentication with Real Application Clusters (RAC) databases. • `-password` *`password`* as the RMAN password required to connect to the RMAN catalog. • `-controlfile` if you are using the control file as the RMAN repository. • `-remove-rman` to remove RMAN.

If you want to...	Then...
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 to change 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.
Disable backup protection for the profile	Specify `-noprotect` to not protect the database backups created by using the profile. For a profile that had `-protect` enabled, if you want to disable protect, a warning message is displayed stating that this action will delete the dataset and you will not be able to restore or clone backups for this profile.
Enable email notifications for the completion status of the database operations	Specify the following options and variables: • `-summary-notification` enables you to configure a summary email notification for multiple profiles under a repository database. • `-notification` enables you to receive an email notification on the completion status of the database operation for a profile. • `-success -email`*email_address2* enables you to receive an email notification following the completion of a successful database operation performed by using a new or an existing profile. • `-failure -email`*email_address2* enables you to receive an email notification on a failed database operation performed by using a new or an existing profile. • `-subject`*subject_text* specifies subject text for the email notification while creating a new profile or an existing profile. If the notification settings are not configured for the repository and you are trying to configure profile or summary notifications by using the command-line interface (CLI), the following message is logged in the console log: `SMO-14577: Notification Settings not configured.` If you have configured the notification settings and you are trying to configure summary notification by using the CLI without enabling summary notification for the repository, the following message is logged in the console log: `SMO-14575: Summary notification configuration not available for this repository`

If you want to...	Then...
Update the profile to create backup of the archive log files separately	Specify the following options and variables: • `-separate-archivelog-backups` enables you to create a backup of the archive log files separately from the database files. After you specify this option, you can either create a data files-only backup or an archivelogs-only backup. You cannot create a full backup. Also, you cannot revert the profile settings by separating the backup. SnapManager retains the backups based on the retention policy for the backups that were created before taking archivelogs-only backup. • `-retain-archivelog-backups` sets the retention duration for archive log backups. **Note:** If you are updating the profile for the first time, you can separate the archive log backups from the data files backup by using the `-separate-archivelog-backups` option; you must provide the retention duration for the archive log backups by using the `-retain-archivelog-backups` option. Setting the retention duration is optional when you later update the profile. • `-protect` creates an application dataset in the Data Fabric Manager (DFM) server and adds members related to the database, data file, control files, and archive logs. If the dataset exists, it is reused when a profile is created. • `-protection-policy` sets the protection policy to the archive log backups. • `-include-with-online-backups` specifies that the archive log backup is included along with the database backup. • `-no-include-with-online-backups` specifies the archive log file backup is not included along with the database backup.
Change the host name of the target database	Specify `-host` *`new_db_host`* to change the host name of the profile.
Collect the dump files after the profile update operation	Specify the `-dump` option.

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

Related concepts

How to collect dump files on page 322

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 285

Backing up databases

SnapManager enables the backing up of data on local storage resources by using post-processing scripts or by protecting backups on secondary or 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 also enables storage administrators to configure their backups based on policy plans. By using SnapManager, administrators can identify backups that do not conform to policy requirements and rectify those immediately.

SnapManager provides the following options to back up, restore, and recover the 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 data files 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).
- Schedule routine backups.

How SnapManager (3.2 or later) differs from earlier SnapManager versions

SnapManager (3.1 or earlier) enables you to create full database backups that contain data files, control files, and archive log files.

SnapManager (3.1 or earlier) manages only the data files. The archive log files are maintained by using solutions outside SnapManager.

SnapManager (3.1 or earlier) imposes the following constraints in managing database backups:

- Performance impact
  When you perform a full, online database backup (when the database is in the backup mode), the performance of the database reduces for the period of time until the backup is created. In SnapManager (3.2 or later), limited database backups and frequent archive log backups can be taken. Taking frequent archive log backups helps in preventing the database from being placed in backup mode.
- Manual restore and recovery
  When the required archive log files do not exist in the active file system, database administrators have to identify which backup contains the archive log files, mount the database backups, and recover the restored database. This process is time consuming.
- Space constraints
  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. In SnapManager (3.2 or later), the archive log files can be pruned from the active file system to free space periodically.

Why archive log backups are important

Archive log files are required to roll the database forward after a restore operation is performed. Every transaction on an Oracle database is captured in the archive log files (if the database is in the archive log mode). Database administrators can restore the database backups by using the archive log files.

Advantages of archivelog-only backups

- Provides separate retention duration for archivelog-only backups

You can have less retention duration for the archivelog-only backups that are required for recovery.

- Protects the archivelog-only backups based on archive log protection policies
  You can select different protection policies for archivelog-only backups based on their requirement.
- Improves the performance of the database
- Consolidates archive log backups
  SnapManager consolidates the archive log backups every time you take a backup by freeing the duplicate archive log backups.

What SnapManager database backups are

SnapManager enables you to perform different backup tasks. You can assign retention classes to specify how long the backup can be retained; once that time limit is reached, the backup is deleted.

- Create backups on the primary storage
- Create protected backups on the secondary storage resources
- Verify that the backups completed successfully
- View a list of backups
- Schedule backups by using the graphical user interface
- Manage the number of backups retained
- Free backup resources
- Mount and unmount backups
- Delete backups

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

- Hourly
- Daily
- Weekly
- Monthly
- Unlimited

The Protection Manager must be installed to use protection policies for protecting backups. A backup can have one of these protection states: not requested, not protected, or protected.

If new data files are added to the database, you should create a new backup immediately. Also, if you restore a backup taken before the new data files were added and attempt to recover to a point after the new data files were added, the automatic recovery process might fail. See the Oracle documentation to learn more about the process for recovering the data files added after a backup.

What full and partial backups are

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 data files. You can choose to take a separate backup of both tablespaces and data files.

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

Backup type	Advantages	Disadvantages
Full	Minimizes the number of Snapshot copies. For online backups, each tablespace is in backup mode for the entire time of the backup operation. 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	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 create the Snapshot copies. This method of grouping the Snapshot copies minimizes the physical block writes in the log files during an online backup.	The backup can require creating Snapshot copies of multiple tablespaces in the same volume. This method can cause SnapManager to create multiple Snapshot copies of a single volume during the backup operation.

Note: Although you can perform a partial backup, you must 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 creates a Snapshot copy of each volume, while for a partial backup, SnapManager creates 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 ensuring that the maximum limit of Snapshot copies per volume is not reached, you must 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 daily backups created on the previous day.

The following tables show how SnapManager creates Snapshot copies based on the backup type. The example in the tables assumes that database Z 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 different numbers of Snapshot copies.

SnapManager creates Snapshot copies at the volume level instead of the tablespace level, which usually reduces the number of Snapshot copies it must create.

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

Volumes in database	Tablespace TS1 (includes 2 database files)	Tablespace TS2 (includes 2 database files)	Snapshot copies created	Total number of 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	

Volumes in database	Tablespace TS1 (includes 2 database files)	Tablespace TS2 (includes 2 database files)	Snapshot copies created	Total number of 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

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

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

When you specify a full backup, SnapManager performs the following actions:

1. Places the entire database in the online backup mode
2. Creates 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. Creates 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 certain tablespaces or data files that 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.

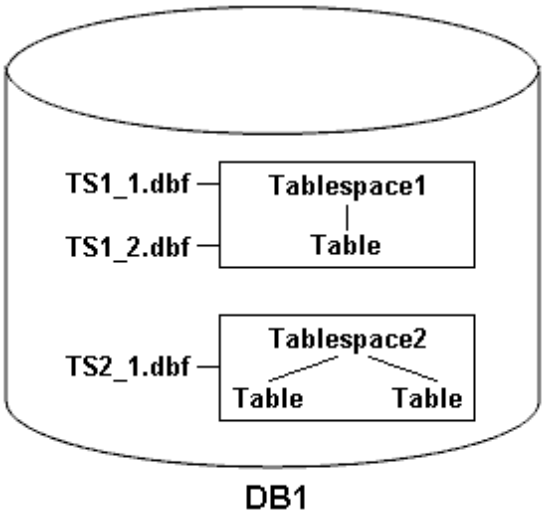
During a partial backup, SnapManager performs these actions:

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

You can find information about some of the backup, restore, and recover scenarios that you can use to accomplish your data protection goals.

The following illustration shows the contents of the tablespace:



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

The following tables describe some full and partial backup, restore, and recover scenarios:

Examples of full backup, restore, and recover operations

Full backup	Restore	Recover
SnapManager makes a backup of everything in 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.	You can 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. • The last transaction made to the database.
	Complete restore without control files SnapManager restores all tablespaces and data files, without the control files.	
	Restore either data files or tablespaces with control files Specify one of the following: • Tablespaces • Data files	SnapManager recovers the data to the last transaction made to the database.
	Restore either data files or tablespaces without control files SnapManager restores one of the following: • Tablespaces • Data files	
	Restore control files only	

Examples of partial backup, restore, and recover operations

Partial backup	Restore	Recover
You can choose one of the following options: • Tablespaces You can specify Tablespace1 and Tablespace2 or only one of them. • Data files You can specify all three database files (TS1_1.dbf, TS1_2.dbf, and TS2_1.dbf), two files, or one file. Regardless of which option you select, the backup includes all the control files. Archive log files are included in the partial backup if the profile is not enabled to create the archive log backups separately.	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.
	Restore either data files or tablespaces with control files SnapManager restores one of the following: • All the data files specified • All the tablespaces specified	
	Restore either data files or tablespaces without control files SnapManager restores one of the following: • 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.	
	Restore control files only	

About control file and archive log file handling

SnapManager includes the control files and optionally includes archive log files with each backup. Archive log files are used for recovery operations.

The database uses control files to identify names, locations, and sizes of the database files. SnapManager includes control files in each backup because control files are used in the restore process.

The changes to a database are tracked by using the online redo logs, which are eventually archived and known as archived redo logs (or archive logs). SnapManager (3.2 or later) enables you to backup data files and archive log files separately with different retentions and frequencies. SnapManager can take backups of only the archive logs or combined backups of data files and archive logs. SnapManager provides complete automated management of archive logs, and does not require any manual intervention for database recovery and also allows pruning of archive logs from one or more archive log destinations after the backup is taken.

Note: To see which tablespaces and data files are included in a backup, use the `backup show` command or the Backup Properties window.

The following table illustrates how SnapManager handles control and archive log files during each operation:

Type of operation	Control files	Archive log files
Backup	Included with each backup	Can be included with each backup
Restore	Can be restored either alone or along with the tablespaces or data files	Can be used for the recovery process

What database backup scheduling is

You can schedule, update, and monitor backups for databases by 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 restarts?	When the SnapManager server restarts, it automatically restarts all the schedules. However, SnapManager does not follow-up on any missed occurrences.

Question	Answer
What happens when two backups are scheduled to occur on two databases at the same time?	SnapManager starts backup operations one at a time and then allows the backups to run in parallel. For example, if a database administrator creates six daily backup schedules for six different database profiles to occur at 1:00 a.m., all six backups run in parallel. If multiple backups are scheduled to occur on a single database profile in a short period of time, the SnapManager server runs only the backup operation with the longest retention duration. Before starting a backup operation, 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. For example, a database administrator might create a daily, weekly, and monthly schedule for a database profile, all of which are scheduled to take backups at 1:00 a.m. On that one day of the month when three backups are scheduled to occur simultaneously at 1:00 a.m., 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 under the user who created the schedule. However, you can change this to your own user ID, if you have valid credentials for both the database profile and host. For instance, by launching Scheduled Backup Properties for the backup schedule created by Avida Davis, Stella Morrow can select her user ID in Perform this operation as user to run the scheduled backup.
How does the SnapManager scheduler interact with the native operating system scheduler?	On the SnapManager server, you cannot view the scheduled backups via the operating system's native scheduler. For instance, after creating a scheduled backup, you do not see any additional entries in cron.

Question	Answer
What happens if the clocks in the graphical user interface and the server are not in sync?	The clocks on the client and server are not synchronized. Therefore, you can schedule a backup in which the start time is in the future on the client 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 p.m. but the current time is 3:30 p.m. on that day, the server performs its first backup at 4:00 p.m. 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 in the graphical interface host is three minutes behind the actual time. • The current time on the client is 8:58 a.m. • You schedule a one-time backup to occur at 9:00 a.m. • You schedule another one-time backup to occur at 8:30 a.m. When the server receives the first request, the time on the server is 9:01 a.m. 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?	SnapManager backup schedules get affected due to Daylight Savings Time or when you change the SnapManager server time. Consider the following implications when the SnapManager server time is changed: • After the backup schedule is triggered, if the SnapManager server time falls back, then the backup schedule does not trigger again. • If Daylight Savings Time begins before the scheduled start time, the backup schedules are triggered automatically. • For example, if you are in the United States and you schedule hourly backups at 4 a.m. that should occur every 4 hours, backups will occur at 4 a.m., 8 a.m., 12 a.m., 4 a.m., 8 p.m., and midnight on the days before and after Daylight Savings Time adjustments in March and November. • Note the following if backups are scheduled for 2:30 a.m. every night: ◦ When the clocks fall back an hour, as the backup is already triggered, the backup does not trigger again. ◦ When the clocks spring forward an hour, the backup triggers immediately. If you are in the United States and want to avoid this issue, you must schedule your backups to start outside the 2:00 a.m. to 3:00 a.m. interval.

Creating database backups

You can create backups of entire databases or portions of databases, including tablespaces, data files, or control files.

About this task

SnapManager provides Snapshot copy capabilities for databases across many host-side storage stacks, including NFS, ASM, Veritas, and others.

> **Note:** For Real Application Clusters (RAC) configurations, SnapManager performs the backup on the host side 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).

From SnapManager 3.4, you can provide your own backup label by overriding the default backup label created by SnapManager. You must set the value of the `override.default.backup.pattern` parameter to **`true`** and specify the new backup label in the `new.default.backup.pattern` parameter. The backup label pattern can contain keywords such as database name, profile name, scope, mode and hostname, which has to be separated by underscore. For example, `new.default.backup.pattern=`*`dbname_profile_hostname_scope_mode`*.

Note: The timestamp will be included automatically at the end of the generated label.

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, you must 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, you should use the `-auto` option.

While creating a backup, if you have enabled pruning and the summary notification was enabled in the profile, two separate emails are triggered. One email is for the backup operation and the other for the pruning. You can correlate these emails by comparing the backup name and backup ID contained in these emails.

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 or later) enables you to back up the archive log files separately from the data files, enabling you to manage the archive log files efficiently.

To create the archive log backups separately, you must create a new profile or update the existing profile to separate the archive log backups by using the `-separate-archivelog-backups` option. Using the profile, you can perform the following SnapManager operations:

- Create an archive log backup.
- Delete an archive log backup.
- Mount an archive log backup.
- Free an 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 allows you to 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 allows you to do the following:

- Create a data files-only backup.
- Create an archivelogs-only backup.
- While creating a data files-only backup, include the archive log backup along with the online data files only backup for cloning.

  If you have included archive log backups along with data files 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 data files 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 archivelogs-only backup:
  - Specify the archive log destination to be backed up along with the archivelogs-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 archivelog-only backup along with an offline data files-only backup.
  - You cannot prune the archive log files when the archive log files are not backed up.
  - You cannot prune the archive log files when Flash Recovery Area (FRA) is enabled for archive log files.

    If you specify the archive log location in Flash Recovery Area, you must ensure that you also specify the archive log location in the `archive_log_dest` parameter.

When you specify the label for online data files backup with included archive log backup, the label is applied for data files 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 you have 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, you must specify the archive log file destinations to be included in the backup, and 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 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. 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`*** **`] [-snapvaultlabel`** ***`SnapVault_label`*** **`][-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]`**

If you want to...	Then...
Create a backup on secondary storage using *SnapManager_cDOT_Vault* protection policy	Specify `-snapvaultlabel`. You must provide the SnapMirror label that you specified in the rules of the SnapMirror policy while setting up the SnapVault relationship as the value.
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.
Specify whether you want 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.

If you want to...	Then...
Specify whether you want 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, and f3 after the option. Example for creating a partial datafile backup on UNIX `smo backup create -profile nosep -data -files /user/user.dbf -online -label partial_datafile_backup -verbose`
Specify whether you want 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, and ts3 after the option. SnapManager supports backing up of read-only tablespaces. While creating the backup, SnapManager changes the read-only table spaces to read-write. After creating the backup, the tablespaces are changed to read-only. Example for creating a partial tablespace backup `smo backup create -profile nosep -data -tablespaces tb2 -online -label partial_tablespace_bkup -verbose`
Specify whether you want to create a unique label for each backup in the following format: full_hot_mybackup_label	For Linux, you might enter this example: `smo backup create -profile targetdb1_prof1 -label full_hot_my_backup_label -online -full  -verbose`
Specify whether you want to create backup of the archive log files separately from the data files	Specify the following options and variables: • `-archivelogs` creates a 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 UNIX `smo backup create -profile nosep -archivelogs -backup-dest /mnt/archive_dest_2/ -label archivelog_bkup -verbose`

If you want to...	Then...
Specify whether you want to create backup of data files and archive log files together	Specify the following options and variables: • `-data` option to specify the data files. • `-archivelogs` option to specify the archive log files. Example for backing up data files and archive log files together on UNIX `smo backup create -profile nosep -data -online -archivelogs -backup-dest  mnt/archive_dest_2 -label data_arch_backup -verbose`
Specify whether you want 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. **Note:** You cannot prune the archive log files when Flash Recovery Area (FRA) is enabled for archive log files. Example for pruning all archive log files while creating a backup on UNIX `smo backup create -profile nosep  -archivelogs -label archive_prunebackup1 -backup-dest /mnt/arc_1,/mnt/arc_2  -prunelogs -all -prune-dest /mnt/arc_1,/mnt/arc_2 -verbose`
Specify whether you want to add a comment about the backup	Specify `-comment` followed by the description string.
Specify whether you want 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.
Specify whether you want to verify the backup at the same time you create it	Specify the `-verify` option.

If you want to...	Then...
Specify whether you want to 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 72

Related tasks

Creating pretask, post-task, and policy scripts on page 202
Creating task scripts on page 212
Storing the task scripts on page 214
Protecting database backups on secondary storage on page 168

Related references

The smo backup create command on page 228

Pruning archive log files

You can prune the archive log files from the archive log locations while creating a backup.

Before you begin

- Archive log files must be 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 are not pruned.
- The database must be in the mounted state.
  If the database is not in mounted state, enter the `-force` option along with backup command.

About this task

While performing a backup operation, you can specify the following:

- Scope of pruning:
  - Delete all the archive log files.
  - Delete the archive log files until the specified System Change Number (SCN).
  - Delete the archive log files until the specified time.
  - Delete the archive log files before the specified time period.
- Destination from where the archive log files must be pruned.

  Note: Even when the archive log file pruning fails in one destination, SnapManager continues to prune the archive log files from the other destinations.

Before deleting the archive log files, SnapManager verifies the following:

- 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.

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. However, if there are any archive log files that are not backed up, not shipped to standby, or not captured by the capture process, SnapManager deletes the archive log files one-by-one. The deletion of archive logs files in a single execution is faster than deleting archive logs one-by-one.

SnapManager can also group the archive log files and delete them batch-by-batch. Each batch will have a maximum of 998 files. This value can be configured below 998 by using the configuration parameter `maximum.archivelog.files.toprune.atATime` in the `smo.config` file.

SnapManager uses Oracle Recovery Manager (RMAN) commands to delete the archive log files. However, SnapManager does not integrate with the RMAN retention policies and deletion policies.

> **Note:** If you delete the archive log files from the archive log destinations, the pruning of archive log files fails.

SnapManager does not support pruning of archive log files in the following scenarios:

- Archive log files are located in the flash recovery area.
- Archive log files are located in the Standby database.
- Archive log files are managed by both SnapManager and RMAN.

Step

1. 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]`

If you want to...	Then...
Prune archive log files	Specify the following options: • `-prunelogs` specifies deleting the archive log files while creating a backup. ◦ `-all` specifies deleting all the archive log files. ◦ `-untilSCN` specifies deleting the archive log files until the specified SCN. ◦ `-until -date` specifies deleting the archive logs including the specified date and time. ◦ `-before` {`-months` | `-days` | `-weeks` | `-hours`} specifies deleting the archive log files before the specified time period.

If you want to...	Then...
Include the destination from where the archive log files are to be pruned	Specify the `-prune-dest` option.

Consolidating archive log backups

SnapManager consolidates the archivelog-only backups every time you take a backup by freeing up the duplicate archivelog-only backups. By default, consolidation is enabled.

About this task

SnapManager identifies the archivelog-only backups which has archive log files in other backups and frees them to maintain minimum number of archivelog-only backups with unique archive log files.

If the archivelog-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 archivelog-only backups is followed only after successful backups and successful pruning operations.

Steps

1. To enable consolidation of the archivelog-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 archivelog-only backups are consolidated.

   If the newly-created archivelog-only backup contains the same archive log files in any of the earlier archivelog-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 archivelog-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

When you create a backup, you can schedule the pruning of archive log files to occur at a specified time.

About this task

SnapManager allows you to prune the archive log files periodically from the active file system.

Step

1. 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]

If you want to...	Then...
Schedule pruning of archive log files	Specify the following options: • `-prunelogs` to schedule pruning of the archive log files • `-prune-dest` to prune archive log files from the archive log destinations
Include a name for the schedule	Specify the `-schedule-name` option.
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 following interval classes: • `-hourly` • `-daily` • `-weekly` • `-monthly` • `-onetimeonly`
Add a comment about the schedule operation	Specify the `-schedule-comment` option followed by the description string.
Specify the start time of the schedule operation	Specify the `-start-time` option in the yyyy-mm-dd hh:mm format.

Protecting archive log backups

While creating profiles, you can enable protection for the archive log backups based on the archive log protection policy.

Step

1. Enter the following 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]**

If...	Then...
You want to backup archive log backups separately and protect the archive log files	Specify the following options: • `-separate-archivelog-backups` enables you to separate the archive log files from the data files. • `-protect` assigns a separate protection policy for the archive log archive log backups. • `-protection-policy` assigns the protection policy for the archive log backups.

What AutoSupport is

The AutoSupport feature enables SnapManager server to send AutoSupport messages to the storage system after the backup operation is complete.

Note: SnapManager sends AutoSupport messages only for the successful backup operations.

You can enable or disable AutoSupport by assigning the following values to the `auto_support.on` configuration parameter in the `smo.config` configuration file:

- **`TRUE`** - Enables AutoSupport
- **`FALSE`** - Disables AutoSupport

Note: By default, AutoSupport is enabled in SnapManager.

Related tasks

Adding storage systems operating in clustered Data ONTAP to the SnapManager server host on page 100
Enabling AutoSupport in SnapManager on page 101
Disabling AutoSupport in SnapManager on page 101

Adding storage systems operating in clustered Data ONTAP to the SnapManager server host

You must add the storage systems operating in clustered Data ONTAP to the SnapManager server host to enable AutoSupport. In SnapManager 3.3 and earlier, AutoSupport was supported only on storage systems operating in 7-Mode.

Step

1. Add storage systems operating in clustered Data ONTAP to the SnapManager server host.

If...	Then run the following command...
Admin Storage Virtual Machine (SVM, formerly known as Vserver) is operating in clustered Data ONTAP	**`snapdrive config set -cserver`** ***`user_name storage_name`***
SVM is operating in clustered Data ONTAP	**`snapdrive config set -vserver`** ***`user_name storage_name`***

Enabling AutoSupport in SnapManager

You must enable AutoSupport, so that storage systems receive messages from the SnapManager server for every successful backup operation.

About this task

AutoSupport can be enabled in two ways:

- By default, the new installation of SnapManager does not contain the `auto_support.on` parameter in the `smo.config` configuration file. This implies that autosupport is enabled.
- You can manually configure the `auto_support.on` parameter.

Steps

1. Stop the SnapManager server.
2. In the `smo.config` configuration file, set the value of the `auto_support.on` parameter to **TRUE**.

   Example

   auto_support.on=TRUE
3. Restart the SnapManager server.

Disabling AutoSupport in SnapManager

You must disable AutoSupport if you do not want the storage system to receive messages from the SnapManager server for every successful backup operation.

About this task

By default, AutoSupport is enabled if the configuration file does not contain the `auto_support.on` parameter. In this scenario, you must add the `auto_support.on` parameter in the configuration file and set the value to **FALSE**.

Steps

1. Stop the SnapManager server.
2. In the `smo.config` configuration file, set the value of the `auto_support.on` parameter to **FALSE**.

   Example

   auto_support.on=FALSE
3. Restart the SnapManager server.

Verifying database backups

You can use the `backup verify` command to verify that the blocks in the database backup are not corrupted. The verify operation invokes the Oracle Database Verify utility for each data file in the backup.

About this task

SnapManager enables you to perform the verify operation at any time that is convenient for you and the users on your system. You can perform the verification immediately after creating the backup. You must specify the profile containing the backup and either the label or the ID of the backup you created.

Note: You can specify `-dump` to collect the dump files after the backup verify operation.

Step

1. Enter the following command:

   `smo backup verify -profile` *`profile_name`* `[-label` *`label`* `| -id` *`id`*`] [-force] [ -dump] [-quiet | -verbose]`

Related references

The smo backup verify command on page 244

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.

Related references

The smo backup update command on page 243

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 243

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 the backup based on the retention settings. Deletion of backups is the default behavior. You can change this default behavior and choose to free the unprotected backups instead.

About this task

By default, SnapManager deletes or frees backups depending on whether they are protected or not 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:

- The backup to secondary storage failed or is in process of being protected.
  This enables the transfer of backups to secondary storage before the retention policy is applied.
- 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. Set the retain.alwaysFreeExpiredBackups property in the smo.config file to **true**.

   For example,

   retain.alwaysFreeExpiredBackups = true

Related references

The smo backup update command on page 243

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]
     ```

Related references

The smo backup update command on page 243

Viewing a list of backups

You can check which backups were created for a profile and the backup state by using the `smo backup list` command. For each profile, the command displays the information about the most recent backup first and then continues until the information for all the backups is displayed.

Step

1. Enter the following command:

   ```
   smo backup list -profile profile_name [-delimiter character] [data | -
   archivelogs] [-quiet | -verbose]
   ```

Related references

The smo backup list command on page 234

Viewing backup details

You can view the detailed information about a particular backup in a profile by using the `smo backup show` command.

About this task

The `smo backup show` command displays the following information for each backup:

- The backup ID
- Whether the backup succeeded or failed
- Backup scope (full, partial, online, or offline)
- Backup mode

- Mount status
- The backup label
- Comment
- The date and time when the operation started and ended
- Information about whether the backup was verified
- The backup retention class
- The database and host name
- The checkpoint System Change Number (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 is displayed:

- The clones made from the backup, if there are any
- Verification information
- If the backup is mounted, SnapManager displays the mount points in use

For the archive log file backup, the same information is displayed as that of the other database backup except for the following information:

- Checkpoint SCN
- End Backup SCN
- Tablespace
- Control files

However, archive log file backup contains the following additional information:

- The first change number of the backup
- The next change number of the backup
- Thread number
- Reset logs ID
- Incarnation

- Log file name

Step

1. Enter the following command:

   smo backup show -profile *profile_name* {-label *label* [data | -archivelogs] | -id *id* [-quiet | -verbose]

Related references

The smo backup show command on page 240

Mounting backups

SnapManager automatically handles mounting a backup to make it available to the host. You can also mount backups in scenarios where you use an external tool, such as Oracle Recovery Manager (RMAN), to access the files in the backup.

About this task

If you are using RMAN, you must 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 consisting of the backup have been mounted.

You can use the `-from-secondary` option to mount the backup from secondary storage. If you do not use this option, SnapManager mounts the backup from primary storage.

You must specify the `-copy-id` option whenever you specify the `-from-secondary` option. If there is more than one backup on the secondary storage system, the `-copy-id` option is used to specify which backup copy on the secondary should be used to mount the backup.

> **Note:** If you are using Data ONTAP operating in 7-Mode, you must specify a valid copy id value. However, if you are using clustered Data ONTAP, you can specify any value (for example 0) for the `-copy-id` option and the value will be discarded by SnapManager.

If you are mounting a database backup to a remote host, you must ensure that the Automatic Storage Management (ASM) credentials are same on both the hosts.

> **Note:** You can optionally collect the dump files after a successful or failed backup mount operation.

Step

1. Enter the following 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 235

Unmounting backups

SnapManager automatically unmounts the backup to make it unavailable to the host server. SnapManager also allows you to unmount if you are using an external tool, such as Oracle Recovery

Manager (RMAN), to access the files in the backup, and to change the state of the backup to remove access.

About this task

If you are unmounting a database backup from a remote host, you must ensure that the Automatic Storage Management (ASM) credentials are same on both the hosts.

You can optionally collect the dump files after a successful or failed unmount backup operation.

The unmount operation might fail sometime with an error message if the mount point is busy, for example, `--[ERROR] FLOW-11019: Failure in Disconnect: SD-10046: You cannot unmount the backup as the mount point is busy with the following mount paths and PID's: /opt/NetApp/smo/mnt/-mnt-neuse_nfsvrdb_arch-20120427052319903_0 with PID 6598`.

You must identify the PID of the session that is resulting in the failure of the unmount operation. Stop the session by running the following command:

`kill pid`

You can then run the unmount operation successfully.

Step

1. Enter the following 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 242

Freeing backups

You can free backups, which deletes the Snapshot copies without deleting the backup metadata. This function frees the space occupied by the backup. You can use the `smo backup free` command to free the backups.

Before you begin

For a backup to be eligible for freeing, you must ensure the following:

- Backup was successful
- Backup is not to be mounted
- Backup does not have clones
- Backup is not to be retained by using an unlimited retention policy
- Backup is not already freed

About this task

If protection is enabled on the profile and the protection policy contains connections from the primary node that use a mirror relationship, then Snapshot copies are deleted on the primary node when a backup is freed. Those Snapshot copies are also deleted from the mirror nodes when the next transfer to secondary storage occurs.

When you free a protected backup, SnapManager requests that Protection Manager 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.

The following table shows the actions taken on both the primary and secondary storage when you free a local backup:

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 the 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 the secondary storage.
	Freed	No action required	No action required	The local backup is already freed.

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. Enter the following command:

   ```
   smo backup free -profile profile_name {-label label [data | -
   archivelogs] | -id id | -all} -force [-dump] [-quiet ] [-force]
   ```

Related references

The smo backup free command on page 233

Deleting backups

You must delete backups when you no longer need them, which frees the space those backups occupy. If you remove backups, you reduce the chance of reaching the limit of 255 Snapshot copies per volume.

Before you begin

- You must ensure that the backup was not used to create a clone.

About this task

When you delete a protected backup, SnapManager deletes the backup from secondary storage and 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. If you delete a freed backup, the backup metadata is removed from the repository.
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. If you delete a freed backup, the backup metadata is removed 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 by secondary storage, the Snapshot copies might be marked for deletion and are deleted later by Protection Manager.

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. 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. Verify that the operations are complete by entering the following command:

   `smo operation list -profile *profile_name* -quiet -verbose`

2. To delete a backup, enter the following 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.

Scheduling database backups

SnapManager (3.2 or later) for Oracle enables you to schedule database backups to occur on a regular basis during off-peak hours to maintain high performance. To schedule a backup, you can create a profile, which includes the database information and retention policy, and then set schedules for the backup.

Note: You must schedule the backups as either a root user or an Oracle user. 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`

The following are some of the schedule-related tasks:

- Schedule a database backup to occur on an hourly, daily, weekly, monthly, or one-time 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: The **Run Now Menu Operation** check box is disabled when a scheduled backup is running for that schedule.

Creating backup schedules

You can schedule a backup to occur at the time and frequency that are suited for your data and environment.

About this task

From SnapManager 3.2 for Oracle, you can schedule the backups of the archive log files separately. However, you must use the profile that you created to separate the archive log files.

If you have scheduled the backups of the data files and archive log files at the same time, then SnapManager creates the data files backup first.

If you select the schedule interval as `-onetimeonly`, then all the pruning options are available. If you select a schedule interval other than `-onetimeonly`, then the pruning options `-until-SCN` and `-until-date` are not supported and the following error message is displayed: `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 High Availability Cluster Multiprocessing (HACMP) environment, you must restart the SnapManager for Oracle server so that the service (virtual) address is mapped to the active host and the SnapManager schedules are adjusted to the active SnapManager host. You can add this information in the preprocessing or post-processing HACMP failover scripts.

Note: If the same profile and schedule name exists in another repository, the backup scheduling operation is not initiated in that repository. The operation will exit with the following message: `operation is already running`.

Step

1. 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]

If you want to...	Then...
Schedule a backup of an online or offline database	Specify `-offline` or `-online` to schedule a backup of the offline or online database. If you specify these, you cannot use `-auto`.
Let SnapManager handle scheduling of a database regardless of whether it is online or offline	Specify `-auto`. If you specify `-auto`, you cannot use `--offline` or `-online`.
Schedule a backup of data files	Specify `-data -files` to list the files separated by commas. For example, use file names f1,f2,f3.
Schedule a partial backup of specific tablespaces	Specify `-tablespaces` to list the tablespaces separated by commas. For example, use ts1,ts2,ts3.
Schedule backup of archive log files	Specify the following: • `-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
Specify the retention class values	Specify `-retain` and indicate whether the backup should be retained according to one of the following retention classes: • `-hourly` • `-daily` • `-weekly` • `-monthly` • `-unlimited` SnapManager defaults to hourly.
Schedule pruning of archive log files	Specify the following: • `-prunelogs` to prune the archive log files while scheduling a backup • `-prune-dest` to specify the archive log destination from which the archive log files are pruned

If you want to...	Then...
Include a name for the schedule	Specify `-schedule-name`.
Schedule backup of the database at a specific time interval	Specify the `interval` option and select the time interval from the following, by which the backups should be created: • `-hourly` • `-daily` • `-weekly` • `-monthly` • `-onetimeonly`
Configure a schedule	Specify `-cronstring` and include the following seven subexpressions that describe the individual option: • 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. • (Optional) 7 refers to the year. **Note:** If you scheduled your backup with different times in `-cronstring` and `-start-time`, then the schedule of the backup is overwritten and triggered by the `-start-time`.
Add a comment about the backup schedule	Specify `-schedule-comment` followed by the description string.
Specify the start time of the schedule operation	Specify `-start-time` in the yyyy-mm-dd hh:mm format.
Change the user of the scheduled backup operation while scheduling the backup	Specify `-runasuser`. The operation runs as the user (root user or Oracle user) who created the schedule. However, you can use your own user ID, if you have valid credentials for both the database profile and host.
Enable a pretask or post-task activity of the backup schedule operation by using the pretask and post-task specification XML file	Specify the `-taskspec` option and provide the absolute path of the task specification XML file for performing a preprocessing 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 allows you to restore a database to the state it was when a Snapshot copy was taken. In addition to the file-based restore process, SnapManager supports volume-based fast restore technology, which reduces the restore time significantly compared to other recovery methods. Because backups are created more frequently, the number of logs that need to be applied is reduced, thus reducing the mean-time-to-recovery (MTTR) for a database.

The following are some of the tasks that you can perform 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 the available logs which stores the last transaction committed to the database.
  The point in time can be an Oracle System Change Number (SCN) or a date and time (yyyy-mm-dd:hh:mm:ss). SnapManager uses the 24-hour clock.
- Restore from backups on primary storage (local backups).
- Restore and recover the backup by using SnapManager, or use SnapManager to restore the backup and use another tool, such as Recovery Manager (RMAN), to recover the data.
- Restore backups from alternate locations.
- Restore protected backups from secondary storage (remote backups) and from an alternate location by using the restore specification file.

For more information, see the *SnapManager for Oracle Best Practices*.

You can restore a backup made by a previous version of SnapManager by using SnapManager 3.0 and later versions.

SnapManager also provides the ability to restore Automatic Storage Management (ASM) databases. An ASM disk group can be shared by multiple databases. Therefore, you cannot revert to an older Snapshot copy of the disk group, because it would revert all the databases. Traditional restore operation solutions go through the host and require all the blocks that constitute the database to 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.

Administrators can perform restore or recovery operations by using the SnapManager graphical user interface (GUI) or by using the command-line interface (CLI).

Related concepts

Backing up databases on page 80

Related references

The smo backup restore command on page 237

Related information

SnapManager for Oracle Best Practices: media.netapp.com/documents/tr-3761.pdf

What database restore is

SnapManager enables you to perform volume-based or file-based backup and restore operations.

The following table describes the restore methods:

Restore process	Details
Volume-based fast restores (from primary storage)	SnapManager restores the data files 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 logical unit number (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 Automatic Storage Management (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 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 the default is the fast restore process, administrators can choose either type. In the fast restore process, SnapManager provides information about the conditions that prevent the fast restore process from completing and those 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 the secondary storage, if the backup also exists on the primary storage.

When the fast restore operation is completed, SnapManager performs the following tasks:

- Frees more recent backups (taken after the backup was restored) in the same profile, because their Snapshot copies no longer exist on the primary storage.
- 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 then Backup_B was created. Each has a Snapshot copy for the data files and one for the archive logs. After SnapManager restores Backup_A by 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 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 the following:
  - Files that were not considered as part of the backup

  - Other files, file systems, or LUNs on the volume

- All the Snapshot copies that were created after the Snapshot copy to which the volume is being reverted are deleted.

  For example, you can no longer restore Tuesday's backup if you volume restored Monday's backup.

- Relationships to secondary storage systems are broken if the restored Snapshot copy is older than the baseline Snapshot copy in the relationship.

Storage-side full file system restore

A storage-side full 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:

- In a SAN environment, all the LUNs used by the file system (and underlying volume group if any) are restored on the storage system.

- In a NAS environment, every file in the file system is 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 restores from primary storage systems, the LUNs (SAN) or files (NAS) are restored in place via SFSR.

- When SnapManager restores from secondary storage systems, the LUNs (SAN) or files (NAS) are copied from secondary storage systems back to the primary storage system over the network.

Because the file system is fully restored, files that are not part of the backup are reverted as well. An override is required if files, which are not part of the restore, exist in the file system that is being restored.

Storage-side file restore

A storage-side file restore is sometimes performed when a storage-side file system restore cannot be performed. In a storage-side file restore, 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 restores NFS files from primary storage systems, the individual files are restored in place by using SFSR.

- When SnapManager restores 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 PFSR.

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
- Disconnecting the clone storage from the host
- Deleting the clone storage

When restoring from the secondary storage, 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 perform host-side file copy restore. SnapManager has two methods of performing a host-side file copy restore from the secondary storage. The method SnapManager selects is configured in the `smo.config` file.

- Direct: SnapManager clones the data on the 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 the 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 take twice as long as the direct secondary access policy because two copies of the data are made.

The decision whether to use the direct or indirect method is controlled by the value of the `restore.secondaryAccessPolicy` parameter in the `smo.config` configuration file. The default is direct.

When can you use fast restore

You can find information about when to perform the fast or volume-based restore process.

To achieve optimal restore performance (volume restore or full disk group restore), you must adhere to the following rules:

- Only complete restores of full backups are eligible for fast restore.
- Only data files are eligible for fast restore.
- Data files must be the only files in a volume to be eligible for fast restore. Although temporary data files can reside in the volume, control files, logs, pfiles, or other files must reside on a separate volume from the data files. (You must set up an Oracle database with data files on a separate volume from control files, archived logs, and online log files.)
- Data files for only one database must be present in the volume.
- Multiple file systems can be used; however, the files in those file systems must be data files for only one database.

- For ASM databases, each database must use its own ASM disk group and the ASM database cannot share storage with any other ASM database.

Note: To check whether a previously created backup is restorable by using fast restore, use the `-preview` option in the `smo backup restore` command.

The fast restore process cannot be used in the following cases:

- On partial backups.
- On backups from the secondary storage if the backup also exists on the primary storage. (You cannot restore these using the file-based or volume-based restore.)
- On backups protected with SnapVault.
  The fast restore process cannot be used for backups that were created earlier than the last protected backup. However, you can use the fast restore process for backups created after the last protected backup. Consider backups A, B, and C. B is the last backup to transfer to secondary storage by using SnapVault. You can fast restore B and C, but cannot fast restore A, because it was created earlier than the last protected backup. SnapVault needs a baseline SnapVault to compute the time difference and send to the secondary storage the next time a backup is transferred to the secondary storage. The last protected backup acts as the baseline Snapshot copy, and so using the fast restore process prevents SnapVault from being able to recognize the baseline.
- FlexClones or LUN clones that use Snapshot copies that were created after the Snapshot copy to which the volume is being reverted. For example, the clones can be the result of a later backup that is being mounted or being cloned by SnapManager.
- LUNs that are not part of the active SnapDrive Snapshot copy.

You cannot perform a fast restore along with other types of restores for the same backup. For example, if one data volume can be restored by using the fast restore process, but another data volume cannot, neither is restored by using the fast restore process. You can choose a file-based restore in this case.

Additionally, you should consider the following points about database restores:

- SnapManager never restores archive logs or redo logs but mounts the backup of archive log files and uses them for recovery.
- SnapManager never restores control files by using volume restore.
- If you want to restore control files and data files, SnapManager performs the restore in two steps.
  SnapManager restores the control files first and then the data files.
- If SnapManager finds temporary files in the same volume as the standard tablespace files, you do not need to issue an override to perform a volume-level restore.
  After a volume restore, the TEMP tablespace is brought back online.

Related concepts

General layout and configuration on page 18

Related information

Documentation on the NetApp Support Site: mysupport.netapp.com

Advantages and disadvantages of using fast restore

DBAs should be aware of the advantages and disadvantages of using volume-based fast restores.

Restoring database backups using fast restores provides the following advantages:

- Volume-based restores reduce the time needed to restore backups.
- SnapManager provides fast restore eligibility checks. SnapManager analyzes the database backup and displays information about whether it can perform the volume-based restore.
- You can preview the restore operation and decide whether to continue with the recommended path or override the recommendation with your selected process.

Restoring database backups using fast restores has the following disadvantages:

- The entire file system is reverted, including files that were not considered part of the backup. Other files, file systems, or LUNs on the volume will also be reverted.
- SnapManager removes all Snapshot copies that were taken after the Snapshot you are reverting to. In effect, you lose the history after the Snapshot copy date. For example, you cannot restore Tuesday's backup if you already restored Monday's backup.

You can avoid the disadvantages by following these recommendations:

- Optimize the database layout according to best practices.
- Protect backups to secondary storage. However, if you delete Snapshot copies from primary storage, you cannot use fast restores to restore them from secondary storage.

Fast restore eligibility checks

When you choose to perform a fast restore of a backup, SnapManager first performs an eligibility check to determine whether the fast restore process can be used.

SnapManager provides the following types of checks:

- Mandatory checks: SnapManager can perform the fast restore process only if all the conditions under this check pass.
- Overridable checks: If the conditions under this check fail, administrators can override the check to force a fast restore process. However, you must override these checks with caution.

The following table lists issues that you might encounter and indicates whether the fast restore eligibility check can be overridden:

Issue	Pass required	Details
ACFS, Voting Disk, or OCR is present on ASM Disk group in 11gR2	Yes	Fast restore cannot be performed. Resolution: None Cannot override.
Only backups created with SnapManager 3.0 or later can be fast restored	Yes	Cannot override.
Only Snapshot copies created with SnapDrive for UNIX 4.0 or later can be fast restored	Yes	Cannot override.

Issue	Pass required	Details
Volume is a root volume	Yes	Volume being restored is a root volume on the storage system. Resolution: Do not use the root volume on the storage system. Cannot override.
Volume restore is not available on Windows	Yes	Volume being restored is a root volume on the storage system. Resolution: None Cannot override.
Volume restore is disabled	Yes	Volume restore has been disabled. Resolution: Enable volume restore by selecting different options when starting the restore. In the command-line interface, do not use `-fast -off`. Cannot override.
Control files and data files on the same volume	Yes	For online backups, control files and data files cannot be on the same volume because SnapManager takes two Snapshot copies of the volume (one in which the data files are consistent in hot backup mode, and one in which the backup control files are consistent after hot backup mode is complete). The volume restore will revert to the first Snapshot copy, which deletes the second Snapshot copy containing the backup control files. When a data file-only restore occurs, the control files are reverted to an inconsistent state, and SnapManager restores the backup control file and then opens the database with the `resetlogs` option, which is not desired behavior. Resolution: Migrate control files and data files onto separate file systems that do not share the same underlying volume. This does not help the restore in which the check failed, but will help future backup restore operations. Cannot override.
Archive logs and data files must not exist on the same volume	Yes	Database archive logs and data files reside in file systems backed by the same storage system volume. If a volume restore was performed, SnapManager cannot open the database after a restore of an online backup because the archived log file that is written after the database is taken out of hot backup mode is not available. Also, you would not be able to roll forward through later transactions that might have been in the archive log files. Resolution: Migrate archive logs and data files onto separate file systems that do not share the same underlying storage system volume. This does not help the restore in which the check failed, but will help future backup restore operations. Cannot override.

Issue	Pass required	Details
Online logs and data files must not exist on the same volume	Yes	Database online redo logs and data files reside in file systems backed by the same storage system volume. If a volume restore was performed, recovery cannot use the online redo logs because they would have been reverted. Resolution: Migrate online redo logs and data files onto separate file systems that do not share the same underlying storage system volume. This does not help the restore in which the check failed, but will help future backup restore operations. Cannot override.
Files in the file system not part of the restore scope are reverted	Yes	Files visible on the host, other than the files being restored, exist in a file system on the volume. If a fast restore or a storage side file system restore was performed, the files visible on the host would be reverted to their original content when the Snapshot copy is created. If SnapManager discovers 20 or less files, they are listed in the eligibility check. Otherwise, SnapManager displays a message that you should investigate the file system. Resolution: Migrate the files not used by the database onto a different file system that uses a different volume. Alternatively, delete the files. If SnapManager cannot determine the file purpose, you can override the check failure. If you override the check, the files not in the restore scope are reverted. Override this check only if you are certain that reverting the files will not adversely affect anything.
File systems in the specified volume group not part of the restore scope are reverted	No	Multiple file systems are in the same volume group, but not all file systems are requested to be restored. Storage side file system restore and fast restore cannot be used to restore individual file systems within a volume group because the LUNs used by the volume group contain data from all file systems. All file systems within a volume group must be restored at the same time to use fast restore or storage side file system restore. If SnapManager discovers 20 or less files, SnapManager lists them in the eligibility check. Otherwise, SnapManager provides a message that you should investigate the file system. Resolution: Migrate the files not used by the database onto a different volume group. Alternatively, delete the file systems in the volume group. Can override.

Issue	Pass required	Details
Host volumes in specified volume group not part of the restore scope are reverted	No	Multiple host volumes (logical volumes) are in the same volume group, but not all host volumes are requested to be restored. This check is similar to `File systems in volume group not part of the restore scope will be reverted` except that the other host volumes in the volume group are not mounted as file systems on the host. Resolution: Migrate host volumes used by the database onto a different volume group. Or, delete the other host volumes in the volume group. If you override the check, all the host volumes in the volume group are restored. Override this check only if you are certain that reverting the other host volumes does not adversely affect anything.
File extents have changed since the last backup	Yes	Cannot override.
Mapped LUNs in volume not part of restore scope are reverted	Yes	LUNs other than those requested to be restored in the volume are currently mapped to a host. A volume restore cannot be performed because other hosts or applications using these LUNs will become unstable. If the LUN names end with an underscore and an integer index (for example, _0 or _1), these LUNs are typically clones of other LUNs within the same volume. It is possible that another backup of the database is mounted, or a clone of another backup exists. Resolution: Migrate LUNs not used by the database onto a different volume. If the mapped LUNs are clones, look for mounted backups of the same database or clones of the database, and unmount the backup or remove the clone. Cannot override.
Unmapped LUNS in volume not part of the restore scope are reverted	No	LUNs other than those requested to be restored in the volume exist. These LUNs are not currently mapped to any host, so restoring them does not disrupt any active processes. However, the LUNs may be temporarily unmapped. Resolution: Migrate LUNs not used by the database onto a different volume, or delete the LUNs. If you override this check, the volume restore will revert these LUNs to the state at which the Snapshot copy was made. If the LUN did not exist when the Snapshot copy was made, the LUN will not exist after the volume restore. Override this check only if you are certain that reverting the LUNs does not adversely affect anything.

Issue	Pass required	Details
LUNs present in Snapshot copy of volume might not be consistent when reverted	No	During Snapshot copy creation, LUNs other than those for which the Snapshot copy was requested, existed in the volume. These other LUNs may not be in a consistent state. Resolution: Migrate LUNs not used by the database onto a different volume, or delete the LUNs. This does not help the restore process in which the check failed, but will help restores of future backups taken after the LUNs are moved or deleted. If you override this check, the LUNs reverts to the inconsistent state at which the Snapshot copy was made. Override this check only if you are certain that reverting the LUNs does not adversely affect anything.
New Snapshot copies have volume clone	Yes	Clones have been created of Snapshot copies that were created after the Snapshot copy is requested to be restored. Because a volume restore will delete later Snapshot copies, and a Snapshot copy cannot be deleted if it has a clone, a volume restore cannot be performed. Resolution: Delete clones of later Snapshot copies. Cannot override.
Newer backups are mounted	Yes	Backups taken after the backup is restored are mounted. Because a volume restore deletes later Snapshot copies, a Snapshot copy cannot be deleted if it has a clone, a backup mount operation creates cloned storage, and a volume restore cannot be performed. Resolution: Unmount the later backup, or restore from a backup taken after the mounted backup. Cannot override.
Clones of newer backups exist	Yes	Backups taken after the backup is restored have been cloned. Because a volume restore deletes later Snapshot copies, and a Snapshot copy cannot be deleted if it has a clone, a volume restore cannot be performed. Resolution: Delete the clone of the newer backup, or restore from a backup taken after the backups that have clones. Cannot override.
New Snapshot copies of volume is lost	No	Performing a volume restore deletes all Snapshot copies created after the Snapshot copy to which the volume is being restored. If SnapManager can map a later Snapshot copy back to a SnapManager backup in the same profile, then the "Newer backups will be freed or deleted" message appears. If SnapManager cannot map a later Snapshot copy back to a SnapManager backup in the same profile, this message does not appear. Resolution: Restore from a later backup, or delete the later Snapshot copies. Can override.

Issue	Pass required	Details
Newer backups will be freed or deleted	No	Performing a volume restore deletes all the Snapshot copies created after the Snapshot copy to which the volume is being restored. Therefore, any backups created after the backup that is being restored are either deleted or freed. Later backups are deleted in the following scenarios: • The backup state is not PROTECTED • `retain.alwaysFreeExpiredBackups` is **`false`** in `smo.config` Later backups are freed in the following scenarios: • The backup state is PROTECTED • `retain.alwaysFreeExpiredBackups` is true **`false`** in `smo.config` Resolution: Restore from a later backup, or free or delete later backups. If you override this check, backups created after the backup that is being restored are deleted or freed.
SnapMirror relationship for volume is lost	Yes (If RBAC is disabled or you do not have RBAC permission)	Restoring a volume to a Snapshot copy earlier than the baseline Snapshot copy in a SnapMirror relationship destroys the relationship. Resolution: Restore from a backup created after the relationship's baseline Snapshot copy. Alternatively, break the storage relationship manually (and then re-create and re-baseline the relationship after the restore is complete). Can override, if RBAC is enabled and you have RBAC permission.
SnapVault relationship for volume is lost if the fast restore process occurred	Yes (If RBAC is disabled or you do not have RBAC permission)	Restoring a volume to a Snapshot copy earlier than the baseline Snapshot copy in a SnapVault relationship destroys the relationship. Resolution: Restore from a backup created after the relationship's baseline Snapshot copy. Alternatively, break the storage relationship manually (and then re-create and re-baseline the relationship after the restore is complete). Cannot override, if RBAC is enabled and you have RBAC permission.
NFS files in volume not part of the restore scope are reverted	No	Files present in the storage system volume, which are not visible on the host, are reverted if a volume restore is performed. Resolution: Migrate files not used by the database onto a different volume or delete the files. Can override. If you override this check failure, the LUNs are deleted.
CIFS shares exist for volume	No	The volume being restored has CIFS shares. Other hosts might be accessing files in the volume during the volume restore. Resolution: Remove unneeded CIFS shares. Can override.

Issue	Pass required	Details
Restoring from alternate location	Yes	A restore specification was provided for the restore operation that specifies that the files be restored from an alternate location. Only host-side copy utilities can be used to restore from an alternate location. Resolution: None. Cannot override.
Storage side file system restore is not supported in a RAC ASM database	Yes	Cannot override.

Backup recovery

In SnapManager, you must perform the restore and recover operations at the same time. You cannot perform a restore operation and then perform a SnapManager recover operation later.

In SnapManager 3.2 or earlier, you can either use SnapManager to restore and recover the backup or 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, you can use RMAN to restore and recover the database at finer granularities such as blocks. This integration combines the benefits of speed and space efficiency of Snapshot copies with the fine level of control for restoring using RMAN.

Note: You must recover a database before you can use it. You can use any tool or script to recover a database.

Starting from SnapManager 3.2 for Oracle, SnapManager enables the restore of database backups automatically by 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 data files are added to the database, Oracle recommends that you take a new backup immediately. Also, if you restore a backup taken before the new data files were added and attempt to recover to a point after the new data files were added, the automatic Oracle recovery process might fail, because it is unable to create data files. See the Oracle documentation for the process for recovering data files added after a backup.

Database state needed for the restore process

The state of the database that is to be restored depends on the type of restore process that you want to perform and the type of files that are to be included.

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

Type of restore	Files included	Database state for this instance	Database state for other instance (RAC only)
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, from Open to Mount) unless the `force` option is specified.

What restore preview plans are

SnapManager provides restore plans before and after a restore operation is completed. The restore plans are used to preview, review, and analyze regarding different restore methods.

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 why some restore mechanisms were not used during the restore operation.

The Preview/Review section

This section shows how each file will be or has been restored. When you view the restore plan before a restore operation, it is called a preview. When you view it after a restore operation is completed, it is called a review.

The following preview example shows that the files are restored using fast volume-based restore, storage-side file system restore, and storage-side system restore methods. To determine why all the files would not be restored by using the same restore method, see the Analysis section.

```
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
The following files will be restored completely via: storage side system restore
+DG2/rac6/undotbs1.dbf
+DG2/rac6/undotbs2.dbf
```

Each restore method has one subsection that contains information about the files that can be restored using that restore method. The subsections are ordered according to decreasing levels of storage method efficiency. In the example above, the fast restore method is more efficient than the storage file system restore method and so is displayed first.

It is possible for one file to be restored by multiple restore methods. Multiple restore methods are used when the underlying logical unit numbers (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 are used to restore the same file, the preview section will be similar to the following:

```
The following files will be restored via a combination of:
[fast restore, storage side file system restore. storage side system restore]
```

The Analysis section

The Analysis section presents the reasons why some restore mechanisms will not be or were not used. You can use this information to determine what is required to enable more efficient restore mechanisms.

The following example shows an Analysis section:

```
Analysis:

The following reasons prevent certain files from being
restored completely via: fast restore
   * LUNs present in snapshot of volume fas960:
        /vol/rac_6_asm_disks may not be consistent when reverted:
        [fas960:/vol/rac6_asm_disks/DG4D1.lun]
     Mapped LUNs in volume fas960:/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.
```

In the example, the first failure is overridable by using `-fast -override` from the command-line interface (CLI), or by selecting **Override** in the graphical user interface (GUI). The second failure about mapped LUNs in the volume is mandatory and not overridable.

You can resolve checks by doing the following:

- To resolve a mandatory check failure, change the environment so that the check will pass.
- To resolve an overridable check failure, you can change the environment, or override the check. However, you must be careful because overriding the check can result in undesired consequences.

Previewing backup restore information

You can preview information about a backup restore process before it occurs 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 provides the following information:

- Which restore mechanism (fast restore, storage-side file system restore, storage-side file restore, or host-side file copy restore) can 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 in the `backup restore` command, SnapManager does not restore anything, but lists the files to be restored and indicates how they will be restored.

Note: You can preview all types of restore mechanisms. The preview shows information about up to 20 files.

Steps

1. Enter the following 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 some files being restored by using the host-side file copy restore process and also explains why some files cannot be restored by 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 file system: +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 other restore processes cannot be used.
3. Begin the restore operation without the `-preview` option, if only reasons that are overridable are displayed.

   You can still override non-mandatory checks.

Restoring backups by using fast restore

You can force SnapManager for Oracle to use the volume-based SnapRestore process rather than other restore processes, if all mandatory fast restore eligibility conditions are met.

About this task

You can use the `backup restore` command with `-fast`:

`backup restore -fast [require | override | fallback | off]`

You can use the `-fast` option only if you want to perform a complete restore of a full backup. The `-fast` option includes the following parameters:

- `require`: Enables you to perform a volume restore, if all mandatory restore eligibility conditions are met and no overridable checks are found.

  If you specify the `-fast` option, but do not specify any parameter for `-fast`, SnapManager uses the `-require` parameter as a default.

- `override`: Enables you to override non-mandatory eligibility checks and perform the volume-based fast restore.
- `fallback`: Enables you to restore the database using any method that SnapManager determines. If you do not specify `-fast`, SnapManager uses the `-fallback` parameter as the default.
- `off`: Enables you to avoid the time required to perform all the eligibility checks, to perform a file-based restore process rather than the fast restore process.

If the backup does not pass the mandatory eligibility checks, the fast restore cannot complete successfully.

SnapManager performs volume-based fast restores in UNIX-based environments only; SnapManager does not perform fast restores in the Windows environment.

While performing VBSR on the data file backup, if the data files and the archive log files are present in the same volume and if the archive log files are not present in the active file system, the restore and recovery of the database succeeds. However, the future archive log Snapshots are deleted as a part of the VBSR resulting in a stale entry of the archive log backup in the repository.

Steps

1. Enter the following command:

   `smo backup restore -profile` *`profile_name`* `-label` *`label`* `-complete -fast require -verbose`

   #### Example

   ```
   smo backup restore -profile targetdb1_prof1
         -label full_bkup_sales_nov_08  -complete -fast require -verbose
   ```

2. Review the fast restore eligibility checks.
3. If the eligibility check determines that no mandatory checks failed, if certain conditions can be overridden, and if you want to continue with the restore process, enter the following command:

   `backup restore -fast override`

Related concepts

Variables available in the task scripts for the restore operation on page 208

Related tasks

Creating pretask, post-task, and policy scripts on page 202
Storing the task scripts on page 214

Restoring backups by using Single File SnapRestore

You can restore the backups by using the Single File SnapRestore (SFSR) method.

Steps

1. Create a profile from the SnapManager graphical user interface (GUI).
2. Back up the database by using the GUI.
3. Unlink the Oracle and Network File System (NFS) service groups from the cluster service groups and freeze them.

4. Ensure that Secure Shell (SSH) is configured between the hosts and SnapDrive for UNIX by setting *`#secure-communication-among-cluster-nodes`* to **on** in the `snapdrive.conf` file.
5. From the SnapManager GUI, perform full backup restore and recovery by using `–alllogs`.
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.

   If one restore operation is followed by another restore operation, then there is a possibility that the creation of the backup Snapshot copy fails. If you run successive restore operations within the specified time in which the SFSR can complete, then SnapManager for Oracle will encounter Snapshot copy creation errors.

   To prevent Snapshot copy creation errors, ensure that restore operations are performed after the time period during which SFSR is in progress.

   To achieve this, check the LUN clone split process status by entering the following command from the storage system command-line interface (CLI):

   `rsh filernname lun clone split status lun-name`

   ```
   Sample Output:
   /vol/delaware_760gb/lun700gb (64% complete)..
   ```

   Note: Volume-based SnapRestore (VBSR) is not supported on Solaris hosts running Veritas stack with SFRAC and VCS environment.

Restoring backups on primary storage

You can use the `backup restore` command to restore a database backup on primary storage.

About this task

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 you are certain that a backup cannot be performed by using a fast restore, you can disable the fast restore eligibility check and perform a file-based restore.

You can use the `backup restore` command options to 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. You can 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:

If you want to restore...	Use...
The entire backup with all tablespaces and data files	`-complete`
The list of specific tablespaces	`-tablespaces`
Specific data files	`-files`
The control files only	`-controlfiles`
Tablespaces, data files, and control files	`-complete -controlfiles`

You can also restore the backup from an alternate location by specifying `-restorespec`.

If you include `-recover`, you can recover the database to:

- 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 date and time recovery and the SCN recovery are point-in-time recoveries.

SnapManager (3.2 or later) provides the ability to recover the restored database backups automatically by 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.

SnapManager provides the external location to Oracle. But, Oracle does not identify the files from the external destination. This behavior is noticed in flash recovery area destination and the Automatic Storage Management (ASM) destination. These are issues with Oracle and the workaround is to always have backup of archive log files in such database layouts.

If any inconsistent SCN or date is provided, then recovery will stop at the last consistent point recovered with the error message `Recovery succeeded, but insufficient`. You have to manually perform 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 log-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 non-Snapshot-capable storage, you should add the destinations for `archivedLogs.exclude` in `smo.config`.

You must 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 a non-Snapshot capable storage on an ASM disk group, and when the database is selected as an archive log destination, SnapManager does not support restoring the backups by using the profile.

If the backup is already mounted, SnapManager 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 Real Application Clusters (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.

The recommended recovery mechanism for the RAC database is as follows:

- 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 date so that the backups can have same 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. Enter the following command:

   `smo backup restore -profile` ***`profile_name`*** **`-label`** ***`label`*** **`-complete -recover -alllogs [-recover-from-location`** ***`path`*** **`[,`****`path2`****`]]-dump-verbose`**

   Example

   ```
   smo backup restore -profile targetdb1_prof1 -label
   full_bkup_sales_nov_08 - complete -recover -alllogs -verbose
   ```

2. To restore data for different scenarios, complete one of the following:

If you want to restore...	Command Example
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`
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`
Complete 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`

If you want to restore...	Command Example
Partial database (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`
Partial database (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`
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`
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.

Example

Enter the following command:

```
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 displays that no mandatory checks failed and if certain conditions can be overridden, and if you want to continue with the restore process, enter the following:

   ```
   backup restore -fast override
   ```

5. Specify external archive log locations by using the `-recover-from-location` option.

Related tasks

Restoring backups by using fast restore on page 129
Restoring backups from an alternate location on page 141

Related references

The smo backup restore command on page 237

Performing block-level recovery with Oracle Recovery Manager (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. RMAN can use either the database's control files or a separate recovery catalog database as its repository.

Steps

1. To perform a full offline backup using SnapManager, enter the following 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 the following 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 the following 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)
```

4. To make the files from the backup accessible on the host and to RMAN, mount the backup by using the following 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 the following 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/NetApp/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 if the blocks have been repaired, use the following command:

   `dbv FILE=filename.dbf`

   Example

   The following output shows that no pages are corrupt:

```
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)
```

   All corrupted blocks were repaired and restored.

Restore files from an alternate location

SnapManager enables you to restore data files and control files from a location other than that of the Snapshot copies in the original volume.

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. When recovering files from external locations, SnapManager uses the `recovery automatic from location` command.

SnapManager also uses Oracle Recovery Manager (RMAN) to recover files. The files to be recovered 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 because it cannot generate the correct file name. Ideally, flash recovery area is a destination that is intended to work with RMAN.

Related tasks

Creating restore specifications on page 139

Restore 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 data from the file system

Before you restore data from an alternate location, you must restore the necessary file system and mount it on the local host.

You can invoke the restore operation from an alternate location to copy the files from alternate file systems to active file systems.

To perform this operation, you must specify the alternate mount points from which to restore the original mount points and the original Snapshot copy names by creating a restore specification file.

> **Note:** The Snapshot copy name is a necessary component because the same file system might be snapped multiple times in a single backup operation (for example, once for the data files and once for the log file).

For Automatic Storage Management (ASM), the disk group name must be same as the disk group that SnapManager cloned to register the backup with Oracle Recovery Manager (RMAN). This name can be obtained by viewing the backup properties.

Related tasks

Creating restore specifications on page 139

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 139

Creating restore specifications

The restore specification file is an XML file that contains the original and alternate locations from which the file can be restored. SnapManager uses this specification file to restore files from the specified location.

About this task

You can create the restore specification file by using any text editor. You must use a .xml extension for the file.

Steps

1. Open a text file.
2. Enter the following:

   `<restore-specification xmlns="http://www.netapp.com">`

3. Enter any file mapping information using the format shown in the following example:

   Example

   ```
   <file-mapping>
       <original-location>/path/dbfilename.dbf</original-location>
       <alternate-location>/path/dbfilename1.dbf</alternate-location>
   </file-mapping>
   ```

   File mapping specifies where a file is 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 where the file is restored.

4. Enter any mounted file system mapping information using the format shown in the example:

   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>
   ```

Mountpoint refers to directory path `/mnt/myfs/)` or an Automatic Storage Management (ASM) disk group mountpoint (for example, `+MY_DG`). The mountpoint mapping specifies the mountpoint from which the files are 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 are restored. The *`snapname`* is the name of the Snapshot copy in which the original files were backed up.

For ASM, the disk group name must be the same 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 Snapshot copy name is a necessary component because the same file system can be used multiple times in a single backup operation (for example, once for the data files and once for the logs).

5. Enter raw device mapping tags and locations using the format shown in the example:

Example

```
<raw-device-mapping>
     <original-location>/path/raw_device_name</original-location>
     <alternate-location>/path/raw_device_name</alternate-location>
</raw-device-mapping>
```

Raw device mapping specifies the location from which a raw device is restored.

6. Enter the following:

   `</restore-specification>`

7. Save the file as a .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.netapp.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 file system(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 file system(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 backups 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 specify the type of restore method you want to use.

About this task

You can 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. Enter the following command:

   `smo backup restore -profile` ***`profile`*** **`-label`** ***`label`*** **`-complete -alllogs -restorespec`** ***`restorespec`***

Related references

The smo backup restore command on page 237

Cloning database backup

If you clone a database, you can 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, which have similar requirements.

You can perform the following tasks related to cloning:

- Clone a database from an existing backup.
- Clone a database in its current state, which enables you to create the backup and the clone in one procedure.
- Clone a protected backup on the 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 on which the database resides.
- Clone a database by 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.

What Cloning is

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.

Some of the advantages of creating a clone by using SnapManager are as follows:

Advantages	Details
Speed	The SnapManager clone operation uses the FlexClone feature available with Data ONTAP. This enables you to quickly clone large data volumes.
Space efficiency	When you create a clone by using SnapManager, space is needed 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 enough space available to copy the entire database.
Virtual copy	You can use the cloned database as if it were 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. Changes to the clone do not affect the original database. After the database is cloned, the cloned database is fully operational.
Simplicity	You can clone a database to any host by using SnapManager commands.

You can clone a backup on the primary (local) storage or a protected backup that is on the secondary (remote) storage. However, you cannot clone a backup if the backup operation is in progress or the backup has been transferred to the secondary storage.

You must ensure that the following prerequisites are met before a database can be cloned:

- Ensure that the `[/etc|/var/opt/oracle]/oratab` directory does not contain an entry pointing to the target system identifier.
- Delete the `spfile<SID>.ora` file from `$ORACLE_HOME/dbs`.
- Delete the `init<SID>.ora` file from `$ORACLE_HOME/dbs`.
- Delete Oracle dump destinations that are specified in the clone specification file.
- Delete the Oracle control files that are specified in the clone specification file.
- Delete the Oracle redo log files that are specified in the clone specification file.

You must give the clone a new system identifier. You cannot simultaneously run two databases with the same system identifier on the same host. You can have a clone on a different host using the same system identifier. You can either give the clone a label or let SnapManager create a label by using the system identifier, date, and time the clone was created.

When you enter a label, you should 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<SID>.ora`.

When you clone a database, SnapManager creates a new `init<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 from the SnapManager CLI. However, from the SnapManager GUI, you can change the directory structure and the metadata of the file system.

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 that 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 nonclustered 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, you must ensure that the ASM instance is running on the remote host.

If the database you cloned was using a `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 <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 XML file, which includes the mappings, options, and parameters for 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 so on.

About this task

You can create the clone specification file by using the SnapManager graphical user interface (GUI), command-line interface (CLI), or a text editor.

When you create the clone specification file by using a text editor, you must save it as a `.xml` file. You can use this XML file for other clone operations.

You can also create a clone specification template and then customize it. You can use the `smo clone template` command or in the GUI, use the Clone wizard.

SnapManager for Oracle adds a version string to any clone specification template that it generates. SnapManager for Oracle assumes the latest version for any clone specification file that lacks a version string.

If you want to perform remote cloning, do not change the default locations of the data files, redo log files, and control files in the clone specification file. If you change the default location, SnapManager fails to create the clone or creates the clone on a database that does not support Snapshot capability. Therefore, the automatic creation of profile fails.

> **Note:** Though mount point and ASM disk group information are editable from the GUI, you can only change the file name and not the file locations.

You can execute a task multiple times, either with the same or different parameter and value combinations.

Steps

1. Open a text file and enter text as shown in the following example:

   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 mount points and raw devices. These items must be mapped from the source to the destination.

Example

The following example displays the data file mount point syntax that you use in the clone specification:

```
<mountpoint>
     <source>/mnt/path/source_data file_mountpoint</source>
     <destination>/mnt/path/target_data file_mountpoint</destination>
</mountpoint>
```

3. Optional: 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 that you use in 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 that 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 that you use in 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

The following example displays the redo log directory structure for cloning:

```
<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 the Oracle parameters that should be set to different values in the cloned database. If you are using Oracle 10, you must specify the following parameters:

   - Background dump

- Core dump
- User dump
- (Optional) Archive logs

  Note: If the parameter values are not set correctly, the clone operation is stopped and you receive an error message.

If you do not specify the location where archive logs are stored, SnapManager creates the clone in `noarchivelog` mode. SnapManager copies this parameter information into the `init.ora` file of the clone.

Example

The following example displays the parameter syntax that you use in clone specification:

```
<parameters>
     <parameter>
          <name>log_archive_dest_1</name>
          <value>LOCATION=/mnt/path/clonename/archive</value>
     </parameter>
</parameters>
```

Example

You can use a default value by using a default element within the parameter element. In the following example, the `os_authentication_prefix` parameter will take the default value because the default element is specified:

```
<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 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>
```

Note: You can use the value from the source database's `init.ora` file for the parameter by not specifying any element.

Example

If a parameter has multiple values, then you can provide the parameter values separated by commas. For example, if you want to move the data files from one location to another, then you can use the `db_file_name_convert` parameter and specify the data file paths separated by commas as seen in the following example:

```
<parameters>
     <parameter>
          <name>db_file_name_convert</name>
          <value>>/mnt/path/clonename/data file1,/mnt/path/clonename/data file2</value>
     </parameter>
</parameters>
```

Example

If you want to move the log files from one location to another, then you can use the `log_file_name_convert` parameter and specify the log file paths separated by commas, as seen in the followowing example:

```
<parameters>
     <parameter>
          <name>log_file_name_convert</name>
          <value>>/mnt/path/clonename/archivle1,/mnt/path/clonename/archivle2</value>
     </parameter>
</parameters>
```

7. Optional: Specify arbitrary SQL statements to execute against the clone when it is online.

   You can use the SQL statements to perform tasks such as re-creating the `temp files` in the cloned database.

   Note: You must ensure that a semicolon is not included at the end of the SQL statement.

Example

The following is a sample SQL statement that you execute 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>
```

Related concepts

Considerations for cloning a database to an alternate host on page 151

Related tasks

Cloning databases and using custom plug-in scripts on page 148
Cloning databases from backups on page 148
Cloning databases in the current state on page 150

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

You can clone a database from a backup by using the `clone create` command.

About this task

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 system identifier (SID). You cannot run two databases with the same SID simultaneously on the same host. You can have a clone on a different host that uses the same SID. To designate a unique name for the clone, use `-label`. 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 to connect to an Oracle instance without having to specify the full database information. SnapManager does not update the `tnsnames.ora` files.

SnapManager always creates a backup including archive log files, if you are using the profile created with `-include-with-online-backups`. SnapManager allows you to clone only the full database backups.

SnapManager (3.2 or later) allows you to clone the backups containing the data files and archive log files.

If the archive log is available from an external location, you can specify the external location during cloning for recovering the cloned database to a consistent state. You must ensure that the external location is accessible by Oracle. Cloning of the archive log-only backups is not supported.

Though the archive log backup is created along with the online partial backup, you cannot create a database clone by using this backup.

You can clone the database backup from the external archive log file location only for a stand-alone database.

The cloning of online database backup of the Real Application Clusters (RAC) database using the external archive log file location fails due to failure in recovery. This is because Oracle database 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.

Cloning datafile backup without archive log backup

When the data files backup does not include the archive log backup, SnapManager for Oracle clones the database based on the System Change Number (SCN) recorded during the backup. If the cloned database cannot be recovered, the `Archived log file for thread <number> and change <SCN> required to complete recovery` error message is displayed, even though SnapManager for Oracle continues to clone the database, and finally succeeds in creating the clone.

When cloning using the data files backup without including the 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 the following 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 151
Variables available in the task scripts for clone operation on page 209

Related tasks

Cloning databases in the current state on page 150
Creating clone specifications on page 144
Creating pretask, post-task, and policy scripts on page 202
Creating task scripts on page 212

Storing the task scripts on page 214

Related references

The smo clone create command on page 245

Cloning databases in the current state

You can create a backup and a clone of the database from the current state of the database by using a single command.

About this task

When you specify the profile with the `-current` option, SnapManager first creates a backup and then a clone from the current state of the database.

In the profile setting, if you have enabled the backup of data files and archive logs together for cloning, whenever you back up the online data files, the archive logs are also backed up. While cloning the database, SnapManager creates the data files backup along with the archive log backup and creates the database clone. If the archive log backup is not included, SnapManager does not create the archive log backup and therefore cannot create the clone of the database.

Step

1. To clone the database in 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 that you want to use.

   Note: You can specify the `-dump` option as an optional parameter to collect the dump files after the successful or failed operations. The dump is collected for both the backup and clone operations.

Cloning database backups without resetlogs

SnapManager enables you to perform flexible cloning so that you can recover the cloned database manually to a desired point in time without opening the database by using resetlogs. You can also manually configure the cloned database as a Data Guard Standby database.

About this task

When you can select the `-no-resetlogs` option while creating the clone, SnapManager performs the following activities to create the cloned database:

1. Executes the preprocessing task activity, if specified, before starting the clone operation
2. Creates the cloned database with the user-specified SID
3. Executes the SQL statements issued against the cloned database.
   Only the SQL statements that can be executed in mount state are successfully executed.
4. Executes the post-processing task activity, if specified.

What tasks you need to do to recover the cloned database manually

1. Mount the archive log backups and recover the cloned database manually by using the archive log files from the mounted path.

2. After performing manual recovery, open the recovered cloned database with `-resetlogs` option.
3. Create temporary tablespaces, if required.
4. Run the DBNEWID utility.
5. Grant sysdba privilege to the credentials of the cloned database.

While cloning the database backups using the `-no-resetlogs` option, SnapManager leaves the cloned database in the mounted state for manual recovery.

Note: The cloned database created with the `-no-resetlogs` option is not a complete database. Therefore you must not perform any SnapManager operations on this database, though SnapManager does not restrict you from performing any operations.

If you do not specify the `-no-resetlogs` option, SnapManager applies the archive log files, and opens the database with resetlogs.

Step

1. Enter the following command:

   `smo clone create -profile` *`profile_name`* `[-backup-label` *`backup_name`* `| -backup-id` *`backup_id`* `| current] -newsid` *`new_sid`* `-clonespec` *`full_path_to_clonespecfile`* `-no-resetlogs`

   If you try to specify both `-no-resetlogs` and `recover-from-location` options, SnapManager does not allow you to specify both these options together, and displays the error message: `SMO-04084: You must specify either one of the options: -no-resetlogs or -recover-from-location`.

Example

```
smo clone create -profile product -backup-label full_offline -
newsid PROD_CLONE -clonespec prod_clonespec.xml -label prod_clone-
reserve -no-reset-logs
```

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 some requirements that must be met.

The following table shows the source and target host setup requirements:

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 the target host.

Prerequisite set up	Requirement
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 of compatible versions

You can also clone an Automatic Storage Management (ASM) database to a remote host. When doing so, you must ensure that the ASM instance is running on the remote host.

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 67
Cloning databases from backups on page 148
Creating clone specifications on page 144

Related references

The smo clone create command on page 245

Cloning with RAC databases

You can clone your RAC database to a non-RAC database or you can clone it to a non-RAC database and then change it to a RAC database.

Steps

1. To clone your RAC database, enter the following command:

   `smo clone create -backup-label backup_label -profile profile_name -newsid new_SID -clonespec full_path_to_clonespec_file -verbose`

2. To change the cloned non-RAC database to a RAC database, rename the file from *`initclone_SID.ora`* to *`initclone_local_instance_SID.ora`*.

3. Edit the renamed file to set the parameter `cluster.database` to *`true`*. RAC database information.
4. Register the cloned RAC database with srvctl.

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 249

Viewing detailed clone information

You can view detailed information about a specific clone by using the `clone show` command.

About this task

The `clone show` command displays the following information:

- Clone system identifier and clone ID
- Clone operation status
- Clone create start and end date or 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 the primary or secondary storage

Step

1. Enter the following command:

   smo clone show -profile *profile_name* [-label *label* | -id *guid*]

Related references

The smo clone show command on page 250

Deleting clones

You can delete the clones when the size of the Snapshot copy reaches between 10% and 20% of the backup. This also guarantees that the clone has the most current data.

About this task

The label is the unique identifier for each clone in a profile. You can use the clone label or ID, but not the system identifier (SID) to delete the clone.

Note: The clone SID and the clone label are not the same.

When you are deleting a clone, the database must be running. Otherwise, many files and directories for the existing clone will not be deleted, resulting in more work before another clone can be created.

The directories specified for certain Oracle parameters in the clone are 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 are not deleted.

Note: You cannot delete a clone when the clone is used in other operations.

You can optionally collect the dump files after a successful or failed clone delete operation.

Step

1. Enter the following command:

   smo clone delete -profile *profile_name* [-label *label* | -id *guid*] [-syspassword *syspassword*]| [login -username *db_username* -password] *db_password* -port *db_port*] [-asminstance -asmusername *asm_username* -asmpassword *asm_password*][-force][-dump][-quiet]|[-verbose]

Example

```
smo clone delete -profile targetdb1_prof1 -label sales0908_clone1
```

Related references

The smo clone delete command on page 248

Splitting a clone

SnapManager enables you to split and manage an existing clone that was created by using the FlexClone technology. In the FlexClone technology, the clone and original database share the same physical data blocks.

Before you perform the clone split operation, you can know that the estimated size of the clone to be split and the required space available on the aggregate.

A new profile is generated by SnapManager if the clone split operation is successful. If SnapManager fails to create the new profile, you can manually create a new profile. You can use the new profile to create database backups, restore data, and create clones. If the clone split operation is successful, irrespective of whether the new profile is created or not, the clone-related metadata is removed from the repository database.

You can perform the following tasks related to splitting clones:

- View the clone split estimate.
- Split a clone on a primary storage.
- Split a clone on a secondary storage.
- View the clone split operation status.
- Stop the clone split operation.
- Destroy the profile along with the underlying storage.
- Delete the profile created for a split clone.

When you split a clone from its parent volume, the Snapshot copies associated with the cloned volume are deleted. The backups created for the cloned database before the clone split process cannot be used because the Snapshot copies of these backups are deleted, and the backups remain as stale entries in the repository.

Viewing a clone split estimate

The clone split estimate helps you know the total free space available on the aggregate, the amount of space shared between the clone and the original database, and the space exclusively used by the clone. In addition, you can view the date and time at which the underlying clone was created and the age of the clone. Based on this estimate, you decide whether to split a clone or not.

About this task

To view the clone split estimate, you must enter the profile name of the original clone and the label or GUID of the clone operation. If the clone is in a different host, you can specify the host name.

Step

1. To view the clone split estimate, enter the following command:

   `smo clone split-estimate -profile` *`profile`* `[-host` *`hostname`*`] [-label` *`clone-label`* `| -id` *`clone-id`*`][-quiet | -verbose]`

The following example shows the command for clone split storage estimate:

```
smo clone split-estimate

-profile p1 -label clone_test_label
```

Splitting a clone on primary or secondary storage

You can use the `clone split` command to split the clone. After the clone split is complete, the clone metadata is removed from the repository database and the backup associated with the clone can be deleted or freed.

About this task

The new profile created after the successful split operation is used for managing the split clone. The new profile will be like any other existing profile in SnapManager. You can use this profile to perform backup, restore, and clone operations.

In addition, you can also configure email notification for the new profile. This enables the database administrator to be notified about the status of the database operation performed using the profile.

Note: SnapManager supports the splitting operation when performed on a FlexClone only.

If the split operation fails, an appropriate error message with the reason for failure is displayed. The status of multiple operations is also displayed in the operation log. For example:

```
--[ INFO] The following operations were completed:
Clone Split : Success
Profile Create : Failed
Clone Detach : Success
```

You can optionally collect the dump files after a successful or failed clone split operation.

Note: After you enter the `clone split` command, you should not stop the SnapManager server until the clone split operation has started.

Note: SnapManager generates the profile even if you do not provide any value for the Oracle account (osaccount and osgroup).

Step

1. Enter the following command:

   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*]][-quiet | -verbose]-dump

Viewing the status of the clone split process

You can view the progress of the split process you started.

Step

1. To view the progress of the clone split process, enter the following command:

   smo clone split-status -profile *profile* [-host *hostname*] [-label *split-label* | -id *split-id*] [-quiet | -verbose]

```
smo clone split-status -profile p1 -id
8abc01ec0e78f3e2010e78f3fdd00001
```

Viewing the result of the clone split process

You can view the result of the clone split process you started.

Step

1. To view the result of the clone split process, enter the following command:

   smo clone split-result -profile *profile* [-host *hostname*] [-label *split-label* | -id *split-id*] [-quiet | -verbose]

```
smo clone split-result -profile p1 -id
8abc01ec0e78f3e2010e78f3fdd00001
```

Stopping the clone split process

You can stop the running clone split process.

About this task

After you stop the split process, you cannot resume it.

Step

1. To stop the clone split operation, enter the following command:

   smo clone split-stop -profile *profile* [-host *hostname*] [-label *split-label* | -id *split-id*] [-quiet | -verbose]

```
smo clone split-stop -profile p1 -id
8abc01ec0e78f3e2010e78f3fdd00001
```

Deleting a profile

You can delete a profile as long as it does not contain successful backups that are currently used in other operations. You can delete profiles that contain freed or deleted backups.

Step

1. Enter the following command:

   `smo profile delete -profile *profile* [-quiet | -verbose]`

   You can delete a new profile created for the clone split. While deleting, the `If you delete the profile, you cannot destroy the profile later` warning message is displayed in the SnapManager command-line interface.

```
smo profile delete -profile AUTO-REVEN
```

Destroying a profile

SnapManager enables you to destroy the profile associated with the split clone (database) along with the underlying storage. Before destroying the profile, ensure you remove the associated backups and clones.

Step

1. To destroy a profile created using the split clone operation as well as the split clone database, enter the following command:

   `smo profile destroy -profile *profile* [-host *hostname*] [-quiet | -verbose]`

```
smo profile destroy -profile AUTO-REVEN
```

Deleting a clone split operation cycle from a repository database

You can delete a clone split operation cycle entry from a repository database.

Step

1. To delete a clone split operation cycle entry from a repository database, enter the following command:

   `smo clone split-delete -profile *profile* [-host *hostname*] [-label *split-label* | -id *split-id*] [-quiet | -verbose]`

```
smo clone split-delete -profile p1 -id
8abc01ec0e78f3e2010e78f3fdd00001
```

Introduction to data protection in SnapManager

SnapManager supports data protection to protect the backups on the secondary or tertiary storage systems. You must set up SnapMirror and SnapVault relationships between the source and the destination volumes.

If you are using Data ONTAP operating in 7-Mode, SnapManager provides policy-driven data protection by integrating with Protection Manager (OnCommand Unified Manager). This automates replicating SnapManager backups on a primary storage system to a secondary storage system or even to a tertiary storage system by using SnapVault or SnapMirror policies created by the storage or backup administrator in Protection Manager. Retention on primary storage is controlled by SnapManager based on the retention defined during profile creation and the retention class tagged during the backup creation. Secondary storage backup retention is controlled by the policy defined in Protection Manager.

If you are using clustered Data ONTAP, SnapManager 3.4 provides *SnapManager_cDOT_Mirror* and *SnapManager_cDOT_Vault* policies for data protection. While creating a profile, you can select these policies depending on the SnapMirror or SnapVault relationship that was established using clustered Data ONTAP CLI or System Manager. When you create a backup selecting the profile for which you enabled protection, the backups are protected to a secondary storage system.

If you were using SnapManager 3.3.1 with clustered Data ONTAP, the backups were protected using post-scripts which were selected while creating profiles. If you want to use those profiles, after upgrading to SnapManager 3.4 you must perform the following operations.

- You must update the profiles to select either *SnapManager_cDOT_Mirror* or *SnapManager_cDOT_Vault* policy and delete the post-script that was used for data protection.
- After updating profile to use *SnapManager_cDOT_Vault* policy, you must delete existing backup schedules and create new schedules to specify the SnapVault label for the backups.
- If the profiles were created in SnapManager 3.3.1 without selecting the post-scripts, you must update the profiles to select either *SnapManager_cDOT_Mirror* or *SnapManager_cDOT_Vault* policy to enable data protection.

Note: If you have backups in the secondary storage system that were mirrored or vaulted using SnapManager 3.3.1 post-scripts, you cannot use those backups for restore or cloning using SnapManager 3.4.

What protection policies are

Protection policies are rules that govern how database backups are protected. You can select the protection policies while creating the profile.

A protection policy defines the following parameters:

- When to transfer copies to secondary storage
- The maximum amount of data that should be transferred at scheduled times
- How long to retain copies for each backup location
- Warning and error thresholds for lag times

When protection is enabled, SnapManager creates a dataset for the database. A dataset consists of a collection of storage sets along with configuration information associated with their data. The storage sets associated with a dataset include a primary storage set used to export data to clients, and the set

of replicas and archives that exist on other storage sets. Datasets represent exportable user data. If the administrator disables protection for a database, SnapManager deletes the dataset.

What protection states are

SnapManager shows the state of each backup. Administrators must know the different states and monitor the state of their backups.

A database backup can have the following protection states:

Status	Definition	Explanation
Protected	Protection was requested and has been enabled.	Protection is enabled for the backup in SnapManager and the Protection Manager successfully copied the backup to another set of physical disks (also referred to as secondary storage). If the 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 completed.	Protection is enabled for the backup, but the backup is not copied to another set of physical disks. The backup is not yet protected, or protection failed, or it was protected earlier 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 the backup is not protected, it becomes protected when it is transferred to secondary storage.
Not requested	Protection was not requested.	Protection is not 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 is always shown as Not requested.

What resource pools are

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 the aggregates on that storage system become available for provisioning.

Storage administrators use the Protection Manager's console to assign a resource pool to the backup and mirror copies. The provision application can then automatically provision volumes out of the physical resources in the resource pool to contain backups and mirror copies.

For protected profiles, SnapManager displays information about a profile and indicates whether a storage resource pool has been assigned to that profile. If not, the profile is considered "non-conformant." After a storage resource pool has been assigned to the corresponding profile's dataset, the profile is considered "conformant".

About different protection policies

You can select different policies to protect the backups on the secondary or tertiary storage systems.

If you are using Data ONTAP operating in 7-Mode and SnapManager is integrated with Protection Manager, you must select one of the following protection policies while creating the profile. The Protection Manager's Management Console provides templates to configure protection policies for the datasets. Even though disaster recovery protection policies are listed in the SnapManager user interface, these policies are not supported.

Policy	Description
Back up	A dataset is backed up locally and also from the primary to secondary storage by using SnapVault or SnapMirror.
Back up, then mirror	A dataset is backed up from the primary to secondary storage by using SnapVault or SnapMirror, and then mirrored to a SnapMirror partner.
Local Snapshot copies only	A dataset uses only local Snapshot copies in the primary storage.
Mirror	A dataset is mirrored from the primary to secondary storage by using SnapMirror.
Mirror and back up	A dataset is mirrored from the primary to secondary storage by using SnapMirror, and then backed up to the secondary storage by using SnapVault or SnapMirror.
Mirror and mirror	A dataset is mirrored from the primary to secondary storage on two different SnapMirror partners.
Mirror, then back up	A dataset is mirrored from the primary to secondary storage by using SnapMirror, and then backed up to tertiary storage by using SnapVault or SnapMirror.
Mirror, then mirror	A dataset is mirrored from the primary to secondary storage by using SnapMirror, and then mirrored to an additional SnapMirror partner.
No protection	A dataset has no Snapshot copies, backups, or mirror-copy protection of any kind.
Remote backup only	Data on a storage system is backed up remotely to secondary storage by using SnapVault or SnapMirror. The licensed application carries out no local backup on the primary storage. This protection policy can be applied to third-party systems with Open Systems SnapVault installed.

If you are using clustered Data ONTAP, you must select one of following protection policies while creating the profile.

Policy	Description
SnapManager_cDOT_Mirror	Mirrors the backup.
SnapManager_cDOT_Vault	Vaults the backup.

Configuring and enabling policy-driven data protection

You must configure SnapDrive and the DataFabric Manager server to enable data protection on the profile to protect backups on the secondary storage systems. You can select the protection policies in the Protection Manager's console to specify how database backups will be protected.

Note: You must ensure that OnCommand Unified Manager is installed on a separate server to enable data protection.

Configuring DataFabric Manager server and SnapDrive when RBAC is enabled

When role-based access control (RBAC) is enabled, you must configure the DataFabric Manager server to include the RBAC capabilities. You must also register the SnapDrive user created in the DataFabric Manager server and root user of the storage system in SnapDrive.

Steps

1. Configure the DataFabric Manager server.
   a. To refresh the DataFabric Manager server to update the changes made directly on the storage system by the target database, enter the following command:

      dfm host discover *storage_system*

   b. Create a new user in the DataFabric Manager server and set the password.
   c. To add the operating system user to the DataFabric Manager server administration list, enter the following command:

      dfm user add *sd-admin*

   d. To create a new role in the DataFabric Manager server, enter the following command:

      dfm role create *sd-admin-role*

   e. To add the DFM.Core.AccessCheck Global capability to the role, enter the following command:

      dfm role add *sd-admin-role* DFM.Core.AccessCheck Global

   f. To add *sd-admin-role* to the operating system user, enter the following command:

      dfm user role set *sd-admin sd-admin-role*

   g. To create another role in the DataFabric Manager server for the SnapDrive root user, enter the following command:

      dfm role create *sd-protect*

   h. To add RBAC capabilities to the role created for the SnapDrive root user or the administrator, enter the following commands:

      dfm role add *sd-protect* SD.Config.Read Global

      dfm role add *sd-protect* SD.Config.Write Global

      dfm role add *sd-protect* SD.Config.Delete Global

      dfm role add *sd-protect* SD.Storage.Read Global

      dfm role add *sd-protect* DFM.Database.Write Global

      dfm role add *sd-protect* GlobalDataProtection

   i. To add the target database oracle user to the list of administrators in the DataFabric Manager server and assign the *sd-protect* role, enter the following command:

      dfm user add -r *sd-protect tardb_host1*\oracle

   j. To add the storage system used by the target database in the DataFabric Manager server, enter the following command:

      dfm host set *storage_system* hostLogin=oracle hostPassword=*password*

   k. To create a new role in the storage system used by the target database in the DataFabric Manager server, enter the following command:

```
dfm host role create -h storage_system-c "api-*,login-*" storage-rbac-role
```

l. To create a new group in the storage system and assign the new role created in the DataFabric Manager server, enter the following command:

```
dfm host usergroup create -h storage_system-r storage-rbac-rolestorage-rbac-group
```

m. To create a new user in the storage system and assign the new role and the group created in the DataFabric Manager server, enter the following command:

```
dfm host user create -h storage_system-r storage-rbac-role -p password -g storage-rbac-group tardb_host1
```

2. Configure SnapDrive.

a. To register the credentials of the *sd-admin* user with SnapDrive, enter the following command:

```
snapdrive config set -dfm sd-admin dfm_host
```

b. To register the root user or the administrator of the storage system with SnapDrive, enter the following command:

```
snapdrive config set tardb_host1 storage_system
```

Configuring SnapDrive when RBAC is not enabled

You must register the root user or the administrator of the DataFabric Manager server and root user of the storage system with SnapDrive to enable data protection.

Steps

1. To refresh the DataFabric Manager server to update the changes made directly on the storage system by the target database, enter the following command:

Example

```
dfm host discover storage_system
```

2. To register the root user or the administrator of the DataFabric Manager server with SnapDrive, enter the following command:

Example

```
snapdrive config set -dfm Administrator dfm_host
```

3. To Register the root user or the administrator of the storage system with SnapDrive, enter the following command:

Example

```
snapdrive config set root storage_system
```

Understanding enabling or disabling of data protection in profile

You can enable or disable data protection while creating or updating a database profile.

To create a protected backup of a database on the secondary storage resources, database administrators and storage administrators perform the following actions.

If you want to...	**Then...**
Create or edit a profile	In the profile, do the following: • Enable backup protection to the secondary storage. • If you are using Data ONTAP operating in 7-Mode and have installed Protection Manager, you can select the policies created by the storage or backup administrator in Protection Manager. If you are using Data ONTAP operating in 7-Mode and protection is enabled, SnapManager creates a dataset for the database. A dataset consists of a collection of storage sets along with configuration information associated with their data. The storage sets associated with a dataset include a primary storage set used to export data to clients, and the set of replicas and archives that exist on other storage sets. Datasets represent exportable user data. If the administrator disables protection for a database, SnapManager deletes the dataset. • If you are using clustered Data ONTAP, you must select either *SnapManager_cDOT_Mirror* or *SnapManager_cDOT_Vault* policy depending on the SnapMirror or SnapVault relationship created. When you disable backup protection, a warning message is displayed stating that the dataset will be deleted and restoring or cloning backups for this profile will not be possible.
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 the `profile show` command output.
Assign storage resources in the Protection Manager Management Console	In the Protection Manager Management Console, 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.
View the conformant profile in SnapManager	In SnapManager, the database administrator sees that the profile has changed to conformant state in both the graphical user interface and in the `profile show` command output, indicating that resources were assigned.

If you want to...	Then...
Create the backup	• Select full backup. • Also, select whether the backup should be protected and select the primary retention class (for example, hourly or daily). • If you are using Data ONTAP operating in 7-Mode and want to protect the backup immediately to secondary storage overriding Protection Manager's protection schedule, specify `-protectnow`.
View the backup	The new backup is shown as scheduled for protection, but not yet protected (in the SnapManager interface and in the `backup show` command output). The status is shown as Not protected.
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.

How SnapManager retains backups on the local storage

SnapManager enables you to create backups that meet retention policies, which specify how many successful backups on local storage should be retained. You can specify the number of successful backups that should be retained in the profile for a given database.

You can create backups for the following:

- 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, you can change the values for the following nonlimited 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 exceeds the retention count. For example, if the backup count is 15 (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 retains the last backup taken.

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

Backups not included in the retention count	**Additional details**
Failed backups	SnapManager retains the information about successful and unsuccessful backups. Although unsuccessful backups require 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	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 retains all the backups that are 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 mirror protection policy	If SnapManager deletes the Snapshot copies for the backup on the primary storage resource and the Snapshot copies are mirrored, the next backup to the secondary storage will 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 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. You can override these defaults and set the values when creating or updating the profile or change the default values for retention count and duration in the `smo.config` file.

Backups on primary storage can be protected by backing up to secondary storage. While SnapManager manages the retention and scheduling of backups on primary storage, the Protection Manager 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 are freed. Their storage resources or Snapshot copies are deleted, but the backups remain in the SnapManager repository and are available for restoration from the secondary storage. You do not have to free backups (for example, with the backup free command). Backups are freed until the backup no longer exists on the secondary storage, and at that point, the backup is deleted.
- If they are not protected, the local backups are deleted.

In an archivelog-only backup operation, SnapManager does not archive the redo log files, unlike in the online database backup process. You must add a pretask script to archive the redo log files before performing the archivelog-only backup operation. The pretask script must run the `alter system switch logfile` command.

The following example shows the actions that SnapManager takes on various types of backups, based on a three-daily-backups retention policy (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	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	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, this backup is deleted.

Related information

Documentation on the NetApp Support Site: mysupport.netapp.com

Prerequisites for data protection

- You must disable the SnapDrive configuration parameter `check-export-permission-nfs-clone` by setting the value to **`off`**.

  For more information about `check-export-permission-nfs-clone`, see SnapDrive for UNIX documentation available at the NetApp Support Site.

- You must configure the SnapMirror relationship for the requested secondary storage volumes in the secondary storage system.

- You must configure the SnapVault relationship for the requested secondary storage qtrees in the secondary storage system for Data ONTAP operating in 7-Mode.

- You must define a policy and rules for the user-defined SnapMirror label if you are using SnapVault post-script for clustered Data ONTAP.

  The SnapVault post-script supports clustered Data ONTAP volumes and the SnapMirror relation types DP and XDP. For information about configuring SnapMirror and SnapVault, see the Data ONTAP documentation available at the NetApp Support Site.

- In NAS environment, you must configure the primary and secondary NAS data path by running the following command:

  `snapdrive config set -mgmtpath` ***`management_path management_path management_path datapath_path`***

  For example, snapdrive config set -mgmtpath f3050-197-91 f3050-197-91 f3050-197-91 f3050-220-91, where f3050-197-91 is the management path and f3050-220-91 is the data path.

Related information

Documentation on the NetApp Support Site: mysupport.netapp.com

Licences required for data protection in SnapManager

You must ensure that licenses required for data protection are installed and enabled on the primary and secondary storage systems.

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 data files, log files, and control files. Secondary storage systems act as remote storage for the protected backups.

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

Note: If you want to enable data protection on the secondary storage systems, you must also install and enable these licenses on the secondary storage systems.

- Either Data ONTAP operating in 7-Mode (7.3.1 or later) or clustered Data ONTAP (8.2 or later)
- SnapVault (depending on the protection policy)
- SnapRestore
- SnapMirror (depending on the protection policy)
- FlexClone is required for Network File System (NFS) and cloning.
  FlexClone is also, required for Storage Area Network (SAN) only if SnapDrive is configured to use FlexClone in SAN environments.
- The appropriate protocol, for example, NFS, Internet Small Computer System Interface (iSCSI), or Fibre Channel (FC)

SnapVault or SnapMirror should be on the 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.

Protecting database backups on secondary storage

You can use SnapManager to protect the backup copies on secondary or tertiary storage systems.

About this task

Step

1. 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] [-snapvaultlabel
SnapVault_label][-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]
```

If you want to...	Then do this...
Create a backup of an online or offline database, rather than allowing SnapManager to manage whether it is online or offline	Specify `-offline` or `-online` to create a backup of the offline database or online database. If you use `-offline` or `-online`, you cannot use `-auto`.
Let SnapManager manage backing up a database, regardless of whether it is online or offline	Specify `-auto`. If you use `-auto`, you cannot use `-offline` or `-online`.
Add a comment about the backup	Specify `-comment`, followed by the description string.
Force the database into the state in which you have specified to back it up, regardless of the state it is currently in	Specify `-force`.
Verify the backup at the same time that you create it	Specify `-verify`.
Create a backup on secondary storage	Specify `-protect`. • If you are using Data ONTAP operating in 7-Mode and want to protect the backup immediately to secondary storage overriding Protection Manager's protection schedule, specify `-protectnow`. • To prevent the backup to secondary storage, specify `-noprotect`. If you are using clustered Data ONTAP and while creating profile you have selected *SnapManager_cDOT_Vault* protection policy, you must specify `-snapvaultlabel`. You must provide the SnapMirror label that you specified in the rules of the SnapMirror policy while setting up the SnapVault relationship as the value.
Specify the retention class values	Specify `-retain` and indicate whether the backup should be retained depending on one of the following retention classes: • `-hourly` • `-daily` • `-weekly` • `-monthly` • `-unlimited` If you do not specify the retain class, SnapManager defaults to hourly.

Examples

The following command protects a database backup:

```
smo backup create -profile PAYDB -protect -retain -daily -full auto -label
full_bkup_sales
```

The following command immediately protects a database backup:

```
smo backup create -profile PAYDB -protectnow -retain -daily -full auto -label
full_bkup_sales
```

Restoring protected backups from secondary storage

You can restore protected backups from secondary storage. However, you cannot restore backups from secondary storage if the backup also exists on primary storage.

Related tasks

Restoring backups from an alternate location on page 141
Creating restore specifications on page 139

Related references

The smo backup restore command on page 237

Restores of protected backups overview

You can choose the restore method that you want to use to restore the backup data from secondary storage to primary storage.

The following table explains the different scenarios and methods that you can use to restore a backup from secondary storage:

Restore target	Explanation
Directly to primary storage	Returns the data from the secondary storage system directly to the original location on the primary storage system over the same network that was used to protect the data. SnapManager uses the direct storage method whenever possible. This method is not possible if the data is in a file system on storage area network (SAN) and if any of the following conditions apply: • Other non-database files are not being restored in the same file system. • Snapshot copies of the control files and data files in a file system being restored were taken at different times. • The logical unit number (LUN) is in a volume group, but other LUNs in the same volume group are not being restored.
Directly to host	Clones the data on the secondary storage system and mounts the cloned data on the host. After the data is cloned and mounted, SnapManager copies it into its original location.

Restore target	Explanation
Indirectly to storage or host	Returns the data from the secondary storage system to a new location on the primary system over the same network that was used to protect the data and to mount the new storage on the host. After the data is returned and mounted, SnapManager copies it into its original location. The indirect storage method might require a long time to return the data. SnapManager first copies data to a scratch volume on the primary host before SnapManager uses it to restore and recover the database. Whether the scratch data is automatically deleted depends on the protocol used. • For SAN, SnapManager deletes the returned data. • For network-attached storage (NAS), SnapManager deletes the contents of the returned qtrees, but does not delete the qtrees themselves. To delete the qtrees, administrators should mount the scratch volume and remove the qtrees using the UNIX `rmdir` command.

If you cannot directly return data to storage, SnapManager can return data either directly to host or indirectly to storage or host. The method depends on the policy governing whether the organization allows connection directly to secondary storage or requires data to be copied over the storage network. You can manage this policy by setting configuration information in the `smo.config` file.

Related references

SnapManager configuration parameters on page 47

Restoring backups from secondary storage

You can restore protected backups from secondary storage and choose how you want to copy the data back to the primary storage.

About this task

You can use the `backup restore` command with the `-from-secondary` option to restore the data from secondary storage. If you do not specify an option, SnapManager restores the data from the Snapshot copies on primary storage.

You cannot use `-from-secondary` option if the backup exists on primary storage; the primary backup must be freed before a backup can be restored from secondary storage. If you use a temporary volume, you must specify the volume by using the `-temp-volume` option.

You must specify the `-copy-id` option whenever you specify the `-from-secondary` option. If there is more than one backup on the secondary storage system, the `-copy-id` option is used to specify which backup copy on the secondary should be used for restore operation.

> **Note:** If you are using Data ONTAP operating in 7-Mode, you must specify a valid copy id value. However, if you are using clustered Data ONTAP, you can specify any value (for example 0) for the `-copy-id` option and the value will be discarded by SnapManager.

When restoring from secondary storage, 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 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.

- If *`restore.secondaryAccessPolicy`* = **`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.
- If *`restore.secondaryAccessPolicy`* = **`indirect`**, SnapManager first copies the data to a temporary volume on primary storage, mounts the data from the temporary volume to the host, and then 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 indirect method will take twice as long as the direct method because two copies of the data are created.

Step

1. Perform one of the following:

Option	Command
To restore a complete database if the selected backup exists on primary storage.	**`smo backup restore -profile`** ***`profile_name`*** **`-label`** ***`label`*** **`-complete -recover -alllogs [-copy-id`** ***`id`*** **`]`**
To restore a complete database if the selected backup does not exist on primary storage.	**`smo backup restore -profile`** ***`profile_name`*** **`-label`** ***`label`*** **`-complete -recover -alllogs -from-secondary [-temp-volume <`*****`temp_volume`*****`>] [-copy-id`** ***`id`*** **`]`**

Example

The following command restores a protected backup from secondary storage system:

```
smo backup restore -profile PAYDB -label daily_monday -complete
-recover alllogs -from-secondary -copy-id 3042 -temp-volume smo_scratch_restore_volume
Operation Id [8abc011215d385920115d38599470001] succeeded.
```

Cloning protected backups

You can use SnapManager to clone a copy of a backup that has been protected.

Before you begin

The host (selected for the clone) must have access to the secondary storage using the same storage protocol (for example, SAN or NAS).

About this task

You can use the `-from-secondary` option to specify that you want to clone from the secondary storage.

You must specify the `-copy-id` option whenever you specify the `-from-secondary` option. If there is more than one backup on the secondary storage system, the `-copy-id` option is used to specify which backup copy on the secondary should be used for cloning.

Note: If you are using Data ONTAP operating in 7-Mode, you must specify a valid copy id value. However, if you are using clustered Data ONTAP, you can specify any value (for example 0) for the `-copy-id` option and the value will be discarded by SnapManager.

Deleting clones of protected backups on secondary storage systems might fail. This issue occurs when the system time of the primary and secondary storage systems are not synchronized.

Step

1. Create a clone of a protected backup copy:

   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 3042 sys_db_finance_sept_08
```

SnapManager for Oracle uses Protection Manager to protect a database backup

SnapManager for Oracle and Protection Manager, when installed on a UNIX host and on the server respectively, give the SnapManager database administrator (DBA) the ability to configure and carry out policy-based Oracle database backups to secondary storage, and to restore, if necessary, the backed up data from secondary to primary storage.

In the following example, a DBA, who is using SnapManager, creates a profile for a local backup on primary storage and another profile for a protected backup to secondary storage. Then this DBA works with his network storage administrator, who is using the Protection Manager's console, to configure a policy-based backup of that database from primary to secondary storage.

Details of the target database

This example of integrated database protection describes the protection of a payroll database. The following data is used in the example.

The database administrator (DBA) at TechCo, a 3000-person company headquartered in Atlanta, must create a consistent backup of the production payroll database, PAYDB. The protection strategy for backing up to primary and secondary storage requires that the DBA and the storage administrator work together to back up the Oracle database both locally on primary storage and also remotely, to secondary storage at a remote location.

Profile information

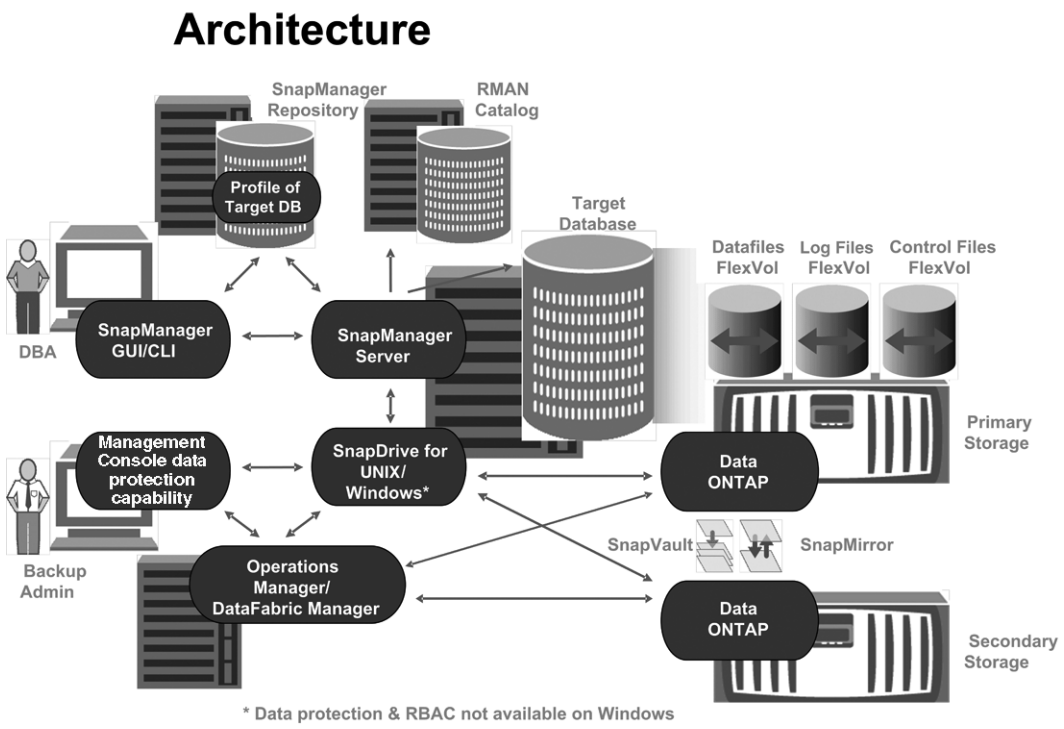
When creating a profile in SnapManager, you need the following data:

- Database name: PAYDB
- Host name: payroll.techco.com
- Database ID: payrolldb
- Profile name: payroll_prod
- Connection mode: Database authentication
- Snapshot naming scheme: *`smo_hostname_dbsid_smoprofile_scope_mode_smid`* (which translates to "smo_payroll.xyz.com_payrolldb_payroll_prod_f_h_x")

Primary and secondary storage configuration and topology

In this example, the TechCo corporation runs its payroll database on a database server that is also a SnapManager for Oracle host and stores its payroll database data and configuration files on primary storage systems at company headquarters. The corporate requirement is to protect that database with daily and weekly backups to local storage as well as backups to storage systems at a secondary storage site fifty miles away.

The following illustration shows the SnapManager for Oracle and the NetApp Management Console data protection capability components required to support local and secondary backup protection.



To manage the payroll database and support its local and secondary backup protection as illustrated in the previous graphic, the following deployment is used.

SnapManager host

The SnapManager host, payroll.techco.com, is located at company headquarters and runs on a UNIX server, which also runs the database program that generates and maintains the payroll database.

Connections

To support local backup and secondary backup protection, the SnapManager host has network connections to the following components:

- SnapManager for Oracle client
- SnapManager repository, which runs the database program, SnapDrive for UNIX, and SnapManager
- Primary storage systems
- Secondary storage systems
- DataFabric Manager server

Installed products

The SnapManager host is installed with the following products for this example:

- SnapManager server
- SnapDrive for UNIX
- Host Utilities

TechCo primary storage systems

The payroll database, including associated data files, log files, and control files, reside on the primary storage systems. These are located at TechCo company headquarters along with the SnapManager host and the network connecting primary storage and the SnapManager host. The latest payroll database transactions and updates are written to the

primary storage systems. Snapshot copies, which provide local backup protection of the payroll database, also reside on the primary storage systems.

Connections

To support secondary backup protection, the primary storage systems have network connections to the following components:

- SnapManager host running the database program, SnapDrive for UNIX, and SnapManager
- Secondary storage systems
- DataFabric Manager server

Installed products

The following licenses must be enabled on these systems for this example:

- Data ONTAP 7.3.1 or later
- SnapVault Data ONTAP Primary
- FlexVol (required for NFS)
- SnapRestore
- NFS protocol

TechCo secondary storage systems

The secondary storage systems, located at a network-connected secondary storage site fifty miles away, are used to store secondary backups of the payroll database.

Connections

To support secondary backup protection, the secondary storage systems have network connections to the following components:

- Primary storage systems
- DataFabric Manager server

Installed products

The following licenses must be enabled on the secondary storage systems for this example:

- Data ONTAP
- SnapVault Data ONTAP Secondary
- SnapRestore
- FlexVol (required for NFS)
- NFS protocol

DataFabric Manager server

The DataFabric Manager server, techco_dfm, is located at company headquarters in a location accessible by the storage administrator. The DataFabric Manager server, among other functions, coordinates the backup tasks between primary and secondary storage.

Connections

To support secondary backup protection, the DataFabric Manager server maintains network connections to the following components:

- NetApp Management Console

- Primary storage systems
- Secondary storage systems

Installed products

The DataFabric Manager server is licensed for the following server products for this example:

- DataFabric Manager

SnapManager repository

The SnapManager repository, located on a dedicated server, stores data about operations performed by SnapManager, for example the time of backups, tablespaces and datafiles backed up, storage systems used, clones made, and Snapshot copies created. When a DBA attempts a full or partial restore, SnapManager queries the repository to identify backups that were created by SnapManager for Oracle for restoration.

Connections

To support secondary backup protection, the secondary storage systems have network connections to the following components:

- SnapManager host
- SnapManager for Oracle client

NetApp Management Console

The NetApp Management Console is the graphical user interface console used by the storage administrator to configure schedules, policies, datasets, and resource pool assignments to enable backup to secondary storage systems, which are accessible to the storage administrator.

Connections

To support secondary backup protection, NetApp Management Console has network connections to the following components:

- Primary storage systems
- Secondary storage systems
- DataFabric Manager server

SnapManager for Oracle client

The SnapManager for Oracle client is the graphical user interface and command line console used by the DBA for the payroll database in this example to configure and carry out local backup and backup to secondary storage.

Connections

To support local backup and secondary backup protection, SnapManager for Oracle client has network connections to the following components:

- SnapManager host
- SnapManager repository, running the database program, SnapDrive for UNIX, and SnapManager
- Database host (if separate from the host running SnapManager)
- DataFabric Manager server

Installed products

To support local backup and secondary backup protection, the SnapManager for Oracle client software must be installed on this component.

Backup schedule and retention strategy

The DBA wants to ensure that backups are available in case of a loss of data, in case of a disaster, and for regulatory reasons. This requires a carefully thought out retention policy for the various databases.

For the production payroll database, the DBA adheres to the following TechCo retention strategy:

Backup frequency	Retention duration	Backup time	Type of storage
Once daily	10 days	7 p.m.	Primary (local)
Once daily	10 days	7 p.m.	Secondary (archive)
Once weekly	52 weeks	Saturdays 1 a.m.	Secondary (archive)

Local backup advantages

Daily local backup provides database protection, which is instantaneous, uses zero network bandwidth, uses a minimum of additional storage space, provides instantaneous restore, and provides finely-grained backup and restore capability.

Because the final weekly backups of the payroll database are retained for a minimum 52 weeks at a secondary storage site, there is no need to retain the daily backups any longer than 10 days.

Protected backup advantages

Daily and weekly backups to secondary storage at a remote location guarantee that if the data at the primary storage site is damaged, the target database is still protected and can be restored from secondary storage.

The daily backups to secondary storage are made to protect against primary storage system damage. Because the final weekly backups of the payroll database are retained for a minimum 52 weeks, there is no need to retain the daily backups any longer than 10 days.

Workflow summary for local and secondary database backup

In this example, the DBA (using SnapManager) and the storage administrator (using the NetApp Management Console data protection capability) coordinate actions to configure local backup and secondary backup (also known as a protected backup) of the target database.

The sequence of actions carried out are summarized as follows:

Secondary resource pool configuration

The storage administrator uses the NetApp Management Console data protection capability to configure a resource pool of storage systems at the secondary site that can be used to store the payroll database backup.

Secondary backup scheduling

The storage administrator uses the NetApp Management Console data protection capability to configure secondary backup schedules.

Protection policy configuration

The storage administrator uses the NetApp Management Console data protection capability to configure a secondary backup protection policy for the target database. The protection policy includes the schedules and specifies the base type of protection to

implement backup protection (backup, mirror, or a combination of both), and names retention policies for primary data, secondary, and sometimes tertiary storage nodes.

Database profile configuration and protection policy assignment

The DBA uses SnapManager to create or edit a profile of the target database that supports secondary backup. While configuring the profile, the DBA:

- Enables backup protection to secondary storage.
- Assigns the new protection policy, which was created in and retrieved from the NetApp Management Console data protection capability, to this profile.

  Assigning the protection policy automatically includes the target database in a partially provisioned, but nonconformant the NetApp Management Console data protection capability dataset. When fully provisioned, the dataset configuration enables backup of the target database to secondary storage.

  The dataset name uses this syntax: *`smo_hostname_databasename`*, which translates to "smo_payroll.techco.com_paydb".

Secondary and tertiary storage provisioning

The storage administrator uses the NetApp Management Console data protection capability to assign resource pools to provision the secondary and sometimes tertiary storage nodes (if the assigned protection policy specifies tertiary storage nodes).

Backup on local storage

The DBA opens the profile with protection enabled in SnapManager and creates a full backup to local storage. The new backup shows in SnapManager as scheduled for protection, but not yet protected.

Secondary backup confirmation

Because the backup was based on a protection-enabled profile, the backup is transferred to secondary according to the protection policy's schedule. The DBA uses SnapManager to confirm the transferral of the backup to secondary storage. After the backup has been copied to secondary storage, SnapManager changes the backup Protection State from "Not protected" to "Protected."

Protected backup configuration and execution

You must configure SnapManager and Protection Manager to support database backup to secondary storage. The database administrator and the storage administrator must coordinate their actions.

Using SnapManager for Oracle to create the database profile for a local backup

The database administrators use SnapManager to create a database profile that will be used to initiate a backup to local storage on a primary storage system. The entire profile creation and backup creation processes are performed entirely in SnapManager; they do not involve Protection Manager.

About this task

A profile contains the information about the database being managed, including its credentials, backup settings, and protection settings for backups. By creating a profile, you do not need to specify database details each time you perform an operation on that database, instead just provide the profile name. A profile can reference only one database. That same database can be referenced by more than one profile.

Steps

1. Go to the SnapManager for Oracle client.
2. From the SnapManager Repositories tree, right-click the host you want associated with this profile, and select **Create Profile**.
3. In the Profile Configuration Information page, enter the following information and click **Next**.
   - Profile name: payroll_prod
   - Profile password: payroll123
   - Comment: Production Payroll database
4. In the Database Configuration Information page, enter the following information and click **Next**.
   - Database name: PAYDB
   - Database SID: payrolldb
   - Database host: Accept the default
     Because you are creating a profile from a host in the repository tree, SnapManager displays the host name.
5. In the second Database Configuration Information page, accept the following database information and click **Next**:
   - Host Account, representing the Oracle user account: oracle
   - Host Group, representing the Oracle group: dba
6. In the Database Connection Information page, select **Use database Authentication** to allow users to authenticate using database information.

   For this example, enter the following information and click **Next**.
   - SYSDBA Privileged User Name, representing the system database administrator who has administrative privileges: sys
   - Password (SYSDBA password): oracle
   - Port to connect to database host: 1521
7. In the RMAN Configuration Information page, select **Do not use RMAN** and click **Next**.

   Oracle Recovery Manager (RMAN) is an Oracle tool that helps you back up and recover Oracle databases using block-level detection.
8. In the Snapshot Naming Information page, specify a naming convention for the Snapshots associated with this profile by selecting variables. The only variable that is required is the **smid** variable, which creates a unique snapshot identifier.

   For this example, do the following:
   a. In the Variable Token list, select the **{usertext}** variable and click **Add**.
   b. Enter "payroll.techco.com_" as the host name and click **OK**.
   c. Click **Left** until the host name appears just after "smo" in the Format box.
   d. Click **Next**.

   The Snapshot naming convention of *`smo_hostname_smoprofile_dbsid_scope_mode_smid`* becomes "smo_payroll.techco.com_payroll_prod2_payrolldb_f_a_x" (where the "f" indicates a full backup, the "a" indicates the automatic mode, and the "x" represents the unique SMID).

9. On the Perform Operation page, verify the information and click **Create**.
10. Click **Operation Details** to see information about the profile create operation and volume-based restore eligibility information.

Using Protection Manager to configure a secondary resource pool

To support backup of the database to secondary storage, the storage administrator uses Protection Manager to organize the secondary storage systems enabled with the SnapVault Secondary license into a resource pool for the backups.

Before you begin

Ideally, storage systems in a resource pool are interchangeable in terms of their acceptability as destinations for backups. For example, when developing the protection strategy for the payroll database, you, as the storage administrator, identified secondary storage systems with similar performance and quality of service levels that would be suitable members of the same resource pool.

You have already created aggregates of unused space on storage systems that you intend to assign to resource pools. This ensures that there is adequate space to contain the backups.

Steps

1. Go to Protection Manager's NetApp Management Console.
2. From the menu bar, click **Data > Resource Pools**.

   The Resource Pools window appears.
3. Click **Add**.

   The Add Resource Pool wizard starts.
4. Complete the steps in the wizard to create the **paydb_backup_resource** resource pool.

   Use the following settings:

   - Name: Use **paydb-backup_resource**
   - Space thresholds (use the defaults):
     - Space utilization thresholds: enabled
     - Nearly Full threshold (for resource pool): 80%
     - Full threshold (for resource pool): 90%

Using Protection Manager to configure secondary backup schedules

To support backup of the database to secondary storage, the storage administrator uses Protection Manager to configure a backup schedule.

Before you begin

Before configuring the schedule for secondary backups, the storage administrator confers with the DBA partner for the following information:

- The schedule that the DBA wants the secondary backups to follow.

  In this case, once-daily backups occur at 7 p.m. and once-weekly backups occur on Saturday at 1 a.m.

Steps

1. Go to the Protection Manager's NetApp Management Console.
2. From the menu bar, click **Policies > Protection > Schedules**.

   The Schedules tab of the Protection Policies window is displayed.
3. Select the Daily schedule **Daily at 8:00 PM** in the list of schedules.
4. Click **Copy**.

   A new Daily schedule, **Copy of Daily at 8:00 PM**, is displayed in the list. It is already selected.
5. Click **Edit**.

   The Edit Daily Schedule property sheet opens to the Schedule tab.
6. Change the schedule name to **Payroll Daily at 7 PM**, update the description, then click **Apply**.

   Your changes are saved.
7. Click the **Daily Events** tab.

   The schedule's current Daily backup time of 8:00 p.m. is displayed.
8. Click **Add** and enter **7:00 PM** in the new time field, then click **Apply**.

   The schedule's current Daily backup time is now 7:00 p.m.
9. Click **OK** to save your changes and exit the property sheet.

   Your new Daily schedule, **Payroll Daily at 7 PM**, is displayed in the list of schedules.
10. Select the Weekly schedule **Sunday at 8:00 PM plus daily** in the list of schedules.
11. Click **Copy**.

    A new Weekly schedule, **Copy of Sunday at 8:00 PM plus daily**, is displayed in the list. It is already selected.
12. Click **Edit**.

    The Edit Weekly Schedule property sheet opens to the Schedule tab.
13. Change the schedule name to **Payroll Saturday at 1 AM plus daily at 7 PM** and update the description.
14. From the **Daily Schedule** drop-down list, select the Daily schedule you just created, **Payroll Daily at 7 PM**.

    Selecting **Payroll Daily at 7 PM** means that this schedule defines when Daily operations occur when the **Payroll Saturday at 1 AM plus daily at 7 PM** schedule is applied to a policy.
15. Click **OK** to save your changes and exit the property sheet.

    Your new Weekly schedule, **Payroll Saturday at 1 AM plus daily at 7 PM**, is displayed in the list of schedules.

Using Protection Manager to configure a secondary backup protection policy

After configuring the backup schedule, the storage administrator configures a protected backup storage policy in which that schedule is to be included.

Before you begin

Before configuring the protection policy, the storage administrator confers with the DBA partner for the following information:

- Retention duration to specify for secondary storage
- Type of secondary storage protection required

About this task

The protection policy that is created, can be listed in SnapManager for Oracle by the DBA partner and assigned to a database profile for the data to be protected.

Steps

1. Go to Protection Manager's NetApp Management Console.
2. From the menu bar, click **Policies** > **Protection** > **Overview**.

   The Overview tab on the Protection Policies window is displayed.
3. Click **Add Policy** to start the **Add Protection Policy** wizard.
4. Complete the wizard with the following steps:

   a. Specify a descriptive policy name.

      For this example, enter **TechCo Payroll Data: Backup** and a description, then click **Next**.

   b. Select a base policy.

      For this example, select **Back up** and click **Next**.

   c. In the **Primary Data** node policy property sheet, accept the default settings and click **Next**.

      Note: In this example, the local backup schedule that was configured in SnapManager is applied. Any local backup schedule that is specified using this method is ignored.

   d. In the **Primary Data to Backup** connection property sheet, select a backup schedule.

      For this example, select **Payroll Saturday at 1 AM plus daily at 7 PM** as your backup schedule, then click **Next**.

      In this example, the schedule that you selected includes both the weekly and daily schedules that you configured earlier.

   e. In the **Backup policy** property sheet, specify the name for the backup node and the retention times for Daily, Weekly, or Monthly backups.

      For this example, specify a Daily backup retention of 10 days and a Weekly backup retention of 52 weeks. After you complete each property sheet, click **Next**.

      After all property sheets are completed, the Add Protection Policy wizard displays a summary sheet for the protection policy that you want to create.
5. Click **Finish** to save your changes.

Result

The **TechCo Payroll Data: Backup** protection policy is listed among the other policies configured for Protection Manager.

After you finish

The DBA partner can now use SnapManager for Oracle to list and assign this policy when creating the database profile for the data to be protected.

Using SnapManager for Oracle to create the database profile and assign a protection policy

You must create a profile in SnapManager for Oracle, enable protection in the profile, and assign a protection policy to create a protected backup.

About this task

A profile contains information about the database being managed, including its credentials, backup settings, and protection settings for backups. After you create a profile, you do not need to specify database details each time you perform an operation. A profile can reference only one database, but that same database can be referenced by more than one profile.

Steps

1. Go to the SnapManager for Oracle client.
2. From the **Repositories** tree, right-click the host, and select **Create Profile**.
3. On the **Profile Configuration Information** page, enter the profile details, and click **Next**.

   #### Example

   You can enter the following information:

   - Profile name: payroll_prod2
   - Profile password: payroll123
   - Comment: Production Payroll database

4. On the **Database Configuration Information** pages, enter the database details, and click **Next**.

   #### Example

   You can enter the following information:

   - Database name: PAYDB
   - Database SID: payrolldb
   - Database host: Accept the default. Because you are creating a profile from a host in the repository tree, SnapManager displays the host name.
   - Host Account, representing the Oracle user account: oracle
   - Host Group, representing the Oracle group: dba

5. On the **Database Connection Information** page, click **Use database Authentication** to allow users to authenticate using database information.
6. Enter the database connection details and click **Next**.

Example

You can enter the following information:

- SYSDBA Privileged User Name, representing the system database administrator who has administrative privileges: sys
- Password (SYSDBA password): oracle
- Port to connect to database host: 1521

7. On the **RMAN Configuration Information** page, click **Do not use RMAN** and click **Next**.

   Oracle Recovery Manager (RMAN) is an Oracle tool that helps you back up and recover Oracle databases using block-level detection.

8. On the **Snapshot Naming Information** page, specify a naming convention for the Snapshots associated with this profile by selecting variables.

   The *smid* variable creates a unique snapshot identifier.

   Perform the following:

   a. In the **Variable Token** list, select *usertext* and click **Add**.
   b. Enter *payroll.techco.com_* as the host name and click **OK**.
   c. Click **Left** until the host name appears just after smo in the Format box.
   d. Click **Next**.

   The Snapshot naming convention of *smo_hostname_smoprofile_dbsid_scope_mode_smid* becomes "smo_payroll.techco.com_payroll_prod2_payrolldb_f_a_x" (where "f" indicates a full backup, "a" indicates the automatic mode, and "x" represents the unique SMID).

9. Select **Protection Manager Protection Policy**.

   The **Protection Manager Protection Policy** enables you to select a protection policy that was configured by using NetApp Management Console.

10. Select **TechCo Payroll Data: Backup** as the protection policy from the protection policies retrieved from NetApp Management Console, and click **Next**.
11. On the **Perform Operation** page, verify the information and click **Create**.
12. Click **Operation Details** to see information about the profile create operation and volume-based restore eligibility information.

Result

- The assignment of a NetApp Management Console protection policy to the database profile automatically creates a nonconformant dataset, visible to the NetApp Management Console operator, with the name convention smo_<hostname>_<profilename>, or in this example: smo_payroll.tech.com_PAYDB.
- If the profile is not eligible for volume restore (also called "fast restore"), the following occurs:
  - The **Results** tab indicates that the profile creation was successful and that warnings occurred during the operation.
  - The **Operation Details** tab includes a WARNING log, which states the profile is not eligible for fast restore and explains why.

Using Protection Manager to provision the new dataset

After the smo_paydb dataset is created, the storage administrator uses Protection Manager to assign storage system resources to provision the dataset's Backup node.

Before you begin

Before provisioning the newly created dataset, the storage administrator confers with the DBA partner for the name of the dataset specified in the profile.

In this case, the dataset name is smo_payroll.tech.com_PAYDB.

Steps

1. Go to Protection Manager's NetApp Management Console.
2. From the menu bar, click **Data** > **Datasets** > **Overview**.

   The Datasets tab of the Datasets window displays a list of datasets that includes the dataset that was just created through SnapManager.
3. Locate and select the **smo_payroll.tech.com_PAYDB** dataset.

   When you select this dataset, the graph area displays the smo_paydb dataset with its backup node unprovisioned. Its conformance status is flagged as nonconformant.
4. With the smo_paydb dataset still highlighted, click **Edit**.

   The Protection Manager's NetApp Management Console displays the Edit Dataset window for the **smo_payroll.tech.com_PAYDB** dataset. The window's navigation pane displays configuration options for the dataset's primary node, backup connection, and backup node.
5. From the navigation pane, locate the options for the dataset's backup node and select **provisioning/resource pools**.

   The Edit Dataset window displays a setting for default provisioning policy and a list of available resource pools.
6. For this example, select the **paydb_backup_resource** resource pool and click >.

   The selected resource pool is listed in the "Resource Pools for this node" field.
7. Click **Finish** to save your changes.

Result

The Protection Manager automatically provisions the secondary backup node with resources from the paydb_backup_resource resource pool.

Using SnapManager for Oracle to create a protected backup

When creating a backup for this example, the DBA selects to create a full backup, sets backup options, and selects protection to secondary storage. Although the backup is initially made on local storage, because this backup is based on a protection-enabled profile, the backup is then transferred to secondary storage according to the protection policy's schedule as defined in Protection Manager.

Steps

1. Go to the SnapManager for Oracle client.
2. From the SnapManager Repository tree, right-click the profile containing the database that you want to back up, and select **Backup**.

The SnapManager for Oracle Backup Wizard starts.

3. Enter

   `Production_payroll`
   as the label.

4. Enter

   `Production payroll Jan 19 backup`
   as the comment.

5. Select **Auto** as the type of backup that you want to create.

   This allows SnapManager to determine whether to perform an online or offline backup.

6. Select **Daily** or **Weekly** as the frequency of the backup.

7. To confirm that the backup is in a valid format for Oracle, check the box next to **Verify backup**.

   This operation uses Oracle DBVerify to check the block format and structure.

8. To force the state of the database into the appropriate mode (for example, from open to mounted), select **Allow startup or shutdown of database, if necessary**, and click **Next**.

9. In the Database, Tablespaces, or Datafiles to Backup page, select **Full Backup** and click **Next**.

10. To protect the backup on secondary storage, check **Protect the Backup** and click **Next**.

11. In the Perform Operation page, verify the information you supplied and click **Backup**.

12. In the progress page, view the progress and results of the backup creation.

13. To view the details of the operation, click **Operation Details**.

Using SnapManager for Oracle to confirm backup protection

Using SnapManager for Oracle, you can view a list of backups associated with a profile, determine whether the backups were enabled for protection, and view the retention class (daily or weekly, in this example).

About this task

At first, the new backup in this example shows as scheduled for protection, but not yet protected (in the SnapManager graphical user interface and in the `backup show` command output). 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.

Steps

1. Go to the SnapManager for Oracle client.

2. In the SnapManager Repository tree, expand the profile to display its backups.

3. Click the **Backups/Clones** tab.

4. In the Reports pane, select **Backup Details**.

5. View the Protection column and ensure that the status is "Protected."

Database restoration from backup

If the active content of the payroll database is accidentally lost or destroyed, SnapManager and the NetApp Management Console data protection capability support restoration of that data from either a local backup or secondary storage.

Using SnapManager for Oracle to restore a local backup on primary storage

You can restore local backups that exist on primary storage. The entire process is performed using SnapManager for Oracle.

About this task

You can also preview information about a backup restore process. You might want to do this to see information about restore eligibility of a backup. SnapManager analyzes data on a backup to determine whether the restore process can be completed by using the volume-based restore or the file-based restore method.

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.

In preview of the restore plan, SnapManager does not restore anything. The preview shows information up to 20 files.

If you want to preview a restore of data files but the database is not mounted, then SnapManager mounts the database. If the database cannot be mounted, then the operation fails and SnapManager returns the database to its original state.

Steps

1. From the **Repository** tree, right-click the backup you want to restore, and select **Restore**.
2. On the **Restore and Recovery Wizard Welcome** page, click **Next**.
3. On the **Restore Configuration Information** page, select **Complete Datafile/Tablespace Restore with Control Files**.
4. Click **Allow shutdown of database if necessary**.

   SnapManager changes the database state, if necessary. For example, if the database is offline and it needs to be online, SnapManager forces it online.
5. On the **Recovery Configuration Information** page, click **All Logs**.

   SnapManager restores and recovers the database to the last transaction and applies all required logs.
6. On the **Restore Source Location Configuration** page, view the information about the backup on primary and click **Next**.

   If the backup exists only on primary storage, SnapManager restores the backup from the primary storage.
7. On the **Volume Restore Configuration Information** page, select **Attempt volume restore** to attempt volume restore method.
8. Click **Fallback to file-based restore**.

This allows SnapManager to use the file-based restore method if the volume restore method cannot be used.

9. Click **Preview** to see the eligibility checks for fast restore and information about mandatory and overridable checks.
10. On the **Perform Operation** page, verify the information you have entered, and click **Restore**.
11. To view details about the process, click **Operation Details**.

Using SnapManager for Oracle to restore backups from secondary storage

Administrators can restore protected backups from secondary storage and can choose how they want to copy the data back to the primary storage.

Before you begin

Before you attempt to restore the backup, check the properties of the backup and ensure that the backup is freed on the primary storage system and is protected on secondary storage.

Steps

1. From the SnapManager for Oracle Repository tree, right-click the backup you want to restore, and select **Restore**.
2. In the Restore and Recovery Wizard Welcome page, click **Next**.
3. In the Restore Configuration Information page, click **Complete Datafile/Tablespace Restore with Control Files**.
4. Click **Allow shutdown of database if necessary**, and then click **Next**.

   SnapManager changes the database state, if necessary. For example, if the database is offline and it needs to be online, SnapManager forces it online.
5. At the Recovery Configuration Information page, click **All Logs**. Then, click **Next**.

   SnapManager restores and recovers the database to the last transaction and applies all required logs.
6. In the Restore Source Location Configuration page, select the ID of the protected backup source and click **Next**.
7. In the Volume Restore Configuration Information page, click **Attempt volume restore** to attempt volume restore.
8. Click **Fallback to file-based restore**.

   This allows SnapManager to use the file-based restore method if the volume restore method cannot be completed.
9. To see the eligibility checks for fast restore and information about mandatory and overridable checks, click **Preview**.
10. At the Perform Operation page, verify the information you have supplied and click **Restore**.
11. To view details about the process, click **Operation Details**.

Performing management operations

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 operations either by using the graphical user interface or command-line 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 277

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 278

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 311

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 228

Restarting the SnapManager UNIX host server

You can restart the server on a UNIX host using the CLI.

Step

1. To restart the server, enter the following command:

   `smo_server restart`

Uninstalling the software from a UNIX host

If you no longer need the SnapManager software, you can uninstall it from the host server.

Steps

1. Log in as root.
2. To stop the server, enter the following command:

   `smo_server stop`
3. To remove the SnapManager software, enter the following command:

   `UninstallSmo`
4. After the introduction text, press **Enter** to continue.

   The uninstallation completes.

Related references

The smo_server stop command on page 228

Configuring an email notification

SnapManager enables you to receive an email notification about the completion status of the profile-executed database operations. SnapManager generates the email and helps you to take appropriate action based on the database operation completion status. Configuring email notification is an optional parameter.

You can configure an email notification for an individual profile as a profile notification and for multiple profiles on a repository database as a summary notification.

Profile notification

For an individual profile, you can receive an email for either or both the successful and failed database operations.

Note: By default, email notification is enabled for failed database operations.

Summary notification

Summary notification enables you to receive a summary email about database operations performed using multiple profiles. You can enable hourly, daily, weekly, or monthly notifications.

Note: From SnapManager 3.3, summary notifications are sent only if you specify the host server that has to send the notification. If you upgrade SnapManager from a version earlier than 3.3, the summary notifications might not be sent if you had not specified the host server in the summary notifications configuration.

Note: If you create a repository in one node of a database that is on a Real Application Clusters (RAC) environment and enable summary notification, later when you add the same repository to another node of the database, the summary notification email is sent twice.

You can use either profile-level notification or summary notification at a time.

SnapManager enables email 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 email notification enabled, you can disable it. If you disable the email notification, you no longer receive email alerts for those profile-executed database operations.

The email 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 of the database
- Start and end time of the database operation
- Status of the database operation

- Error message, if any
- Warning messages, if any

You can configure the following:

- Mail server for a repository
- Email notification for a new profile
- Email notification for an existing profile
- Summary email notification for multiple profiles under a repository

Note: You can configure email notification from both the command-line interface (CLI) and the graphical user interface (GUI).

Configuring a mail server for a repository

SnapManager enables you to specify the mail server details from which the email alerts are sent.

About this task

SnapManager enables you to specify the sender's email server host name or IP address, and the email server port number for a repository database name that requires email 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 email address, you can specify the user name and password.

You must specify name or IP address of the host server that handles the email notification.

Step

1. To configure the mail server to send email 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 email address.	Specify the `-sender-email` option. From SnapManager 3.2 for Oracle, you can include hyphen (-) while specifying the domain name of the email address. For example, you can specify the sender email address as `-sender-email` *`user@org-corp.com`*.
To specify the sender's email server host name or IP address.	Specify the `-mailhost` option.

To do the following...	Then...
To specify the email server port number for a repository database name that requires email 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 email 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 email notification for a new profile

When you are creating a new profile, you can configure to receive an email notification on completion of the database operation.

Before you begin

- You must configure the email address from which the alerts are sent.
- You must use a comma-separated list for multiple email addresses.
  You must ensure that there is no space between the comma and the next email address.

Step

1. 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 email addresses.

When you create a backup of data files and archive log files together using the profile (for creating separate archive log backups), and the data file backup creation fails, the email notification is sent with the data backup as the operation name instead of data backup and archive logs backup. When the data file and archive log file backup operation is successful, you see the output as follows:

```
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     :
```

The following example displays the email 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 email subject for a new profile

You can customize the email subject for the new profile when you create it.

About this task

You can customize the email subject by using the {profile}_{operation-name}_{db-sid}_{db-host}_{start-date}_{end-date}_{status} pattern 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 the mmdd:hh:ss yyyy format	April 27 21:00:45 PST 2012
end-date	End time of the database operation in the mmdd:hh:ss yyyy format	April 27 21:10:45 PST 2012
status	Database operation status	Success

If you do not provide any value for the variables, then SnapManager displays the following error message: `Missing value(s) -subject`.

Step

1. 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*]]

The following is an example showing the email 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 email notification for an existing profile

When you are updating an profile, you can configure to receive an email notification on completion of the database operation.

Before you begin

- You must configure the email address from which the alerts are sent.
- You must enter a single email address or multiple email addresses to which alerts will be sent. You can use a comma-separated list for multiple addresses. You must ensure that there is no space between the comma and the next email address. Optionally, you can add a subject to the email as well.

Step

1. 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*]]

   You can use the `success` option to receive a notification only for successful database operations and the `failure` option to receive a notification only for failed database operations.

Customizing the email subject for an existing profile

SnapManager enables you to customize the email subject pattern for an existing profile by updating that profile. This customized subject pattern is applicable only for the updated profile.

Step

1. 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*]]

The following example shows an email 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 email notification for multiple profiles

SnapManager enables you to configure a summary email notification for multiple profiles under a repository database.

About this task

You can set the SnapManager server host as a notification host from which the summary notification email is sent to the recipients. If the SnapManager server host name or IP address is changed, then the notification host can also be updated.

You can select any one of the schedule times at which you require an email notification:

- Hourly: To receive an email notification every hour
- Daily: To receive an email notification daily
- Weekly: To receive an email notification weekly
- Monthly: To receive an email notification monthly

You need to enter a single email address or a comma-separated list of email addresses to receive notifications for operations performed using those profiles. You must ensure that there is no space between the comma and the next email address when you enter multiple email addresses.

SnapManager allows you to add a customized email 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 the yyyymmdd:hh:ss format
- End time of the database operation in the 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. Enter the following 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* -notification-host *notification-host*

   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 -notification-host wales
```

Adding a new profile to summary email notifications

After you configure a summary email notification for the repository database, you can add a new profile to summary notification by using the `summary notification` command.

Step

1. 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 email notifications

SnapManager enables you to add an existing profile to a summary email notification while updating that profile.

Step

1. 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 email notification for multiple profiles

After you enable the summary email notification for multiple profiles, you can disable them to no longer receive email alerts.

About this task

SnapManager enables you to disable the summary email notification for those profile-executed database operations. From the SnapManager CLI, enter the `notification remove-summary-notification` command to disable the summary email notification for multiple profiles and the name of the repository database for which you do not require email 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 shows summary notification being disabled 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 file and scripts for SnapManager operations

SnapManager for Oracle uses a task specification Extensible Markup Language (XML) file that indicates the pretasks and post-tasks for the backup, restore, and clone operations. You can add the pretask and post-task script names in the XML file for the tasks to be performed before or after the backup, restore, and clone operations.

In SnapManager (3.1 or earlier), you can run the pretask and post-task scripts only for the clone operation. In SnapManager (3.2 or later) for Oracle, you can run the pretask and post-task scripts for the backup, restore, and clone operations.

In SnapManager (3.1 or earlier), the task specification section is part of the clone specification XML file. From SnapManager 3.2 for Oracle, the task specification section is a separate XML file.

> **Note:** SnapManager 3.3 or later does not support the use of the clone specification XML file created in the releases before SnapManager 3.2.

In SnapManager (3.2 or later) for Oracle, you must ensure that the following conditions are met for successful SnapManager operations:

- For backup and restore operations, 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 pretask or post-task activity, you can optionally add the task specification XML file.

You can create the task specification file by using the SnapManager graphical user interface (GUI), command-line interface (CLI), or a text editor. You must use a .xml extension 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 pretasks section includes scripts that could be run before the backup, restore, and clone operations.
- The post-tasks section includes scripts that could be run after the backup, restore, and clone operations.

The values included in the pretasks and post-tasks sections must adhere to the following guidelines:

- Task name: The name of the task must match the name of the script, which is displayed when you run the `plugin.sh -describe` command.

  > **Note:** If there is a mismatch, then you might receive the following 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 is displayed when you run the `plugin.sh -describe` command.

You can create the specification file based on the structure of the following sample task specification file:

```
<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.

From the SnapManager GUI, you can set the parameter value and save the XML file. You can use the Task Enabling page of the Backup Create wizard, Restore or Recovery wizard, and Clone Create wizard, to load the existing task specification XML file, and use the selected file for the pretask 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 in the task specification file for the preprocessing or post-processing activity for the backup, restore, and clone operations irrespective of the file extension of the task specification file.

Creating pretask, post-task, and policy scripts

SnapManager enables you to create the scripts for the preprocessing activity, the post-processing activity, and policy tasks of the backup, restore, and clone operations. You must place the scripts in the correct installation directory to execute the preprocessing activity, post-processing activity, and policy tasks of the SnapManager operation.

About this task

Pretask and post-task script content

All scripts must include the following:

- Specific operations (check, describe, and execute)
- (Optional) Predefined environment variables
- Specific error handling code (return code (rc))

Note: You must include correct error handling code to validate the script.

You can use the pretask 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 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 code.

The policy script is executed before the backup, restore, and clone operations.

Supported format

A shell script file with a .sh extension can be used as the prescript and post-script.

Script installation directory

The directory in which 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 takes 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: You must ensure that the `plugins` directory has the executable permission before using the scripts for the SnapManager operation.

The following table lists the location where you must save the pretask, post-task, and policy scripts for backup, restore, and clone operations:

Activity	Backup	Restore	Clone
Preprocessing	<default_installation_directory>/plugins/backup/create/pre	<default_installation_directory>/plugins/restore/create/pre	<default_installation_directory>/plugins/clone/create/pre
Post-processing	<default_installation_directory>/ plugins/backup/create/post	<default_installation_directory>/plugins/restore/create/post	<default_installation_directory>/plugins/clone/create/post
Policy-based	<default_installation_directory>/plugins/backup/create/policy	<default_installation_directory>/plugins/restore/create/policy	<default_installation_directory>/plugins/clone/create/policy

Sample scripts locations

The following are some samples of the pretask and post-task scripts for the backup and clone operations available in 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 of the script must be declared at the start of the script. You can add new parameter values in `parameter=()` and then use the parameters in the execute function.

Sample script

The following is a sample script with a 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 pretask or post-task scripts that you create must follow a standard SnapManager for Oracle plug-in structure.

The pretask and post-task scripts must include the following operations:

- check
- describe
- execute

If any one of these operations is not specified in the pretask or post-task script, then the script becomes invalid.

When you run the `smo plugin check` command for the pretask or post-task scripts, the returned status of the scripts display error (because the returned status value is not zero).

Operation	Description
check	The SnapManager server runs the `plugin.sh -check` command to ensure that the system has execution permission on the plug-in scripts. You might also include file permission checking on the remote system.

Operation	Description

describe	The SnapManager server runs the `plugin.sh -describe` command to obtain information about your script and match the elements provided by the specification file. Your plug-in script must contain the following description information:
	• `SM_PI_NAME`: Script name. You must provide a value for this parameter.
	• `SM_PI_DESCRIPTION`: Description of the script's purpose. You must provide a value for this parameter.
	• `SM_PI_CONTEXT`: Context in which the script should run-for example, `root` or `oracle`. You must provide a value for this parameter.
	• `SM_PI_TIMEOUT`: The maximum time (in milliseconds) that SnapManager should wait for the script to complete processing and terminate execution. You must provide a value for this parameter.
	• `SM_PI_PARAMETER`: One or more custom parameters necessary for your plug-in script to perform processing. Each parameter should be listed in a new output line and include the name of the parameter and a description. When the script completes processing, the parameter value will be provided to your script by an environment variable.
	The following is the sample output of the Followup_activities script.
	`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 runs the `plugin.sh -execute` command to start your script to execute the script.

Related references

The smo plugin check command on page 280

Variables available in the task scripts for the backup operation

SnapManager provides context information in the form of environment variables related to the backup operation that is 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	Specifies the ID of the current operation	string
SM_PROFILE_NAME	Specifies the name of the profile used	string
SM_SID	Specifies the system identifier of the database	string
SM_HOST	Specifies the host name of the database	string
SM_OS_USER	Specifies the operating system (OS) owner of the database	string

Variables	Description	Format
SM_OS_GROUP	Specifies the OS group of the database	string
SM_BACKUP_TYPE	Specifies the type of the backup (online, offline, or auto)	string
SM_BACKUP_LABEL	Specifies the label of the backup	string
SM_BACKUP_ID	Specifies the ID of the backup	string
SM_BACKUP_RETENTION	Specifies the retention period	string
SM_BACKUP_PROFILE	Specifies the profile used for this backup	string
SM_ALLOW_DATABASE_SHUTDOWN	Specifies if you want to start up or shut down the database If required you can use the `-force` option from the command-line interface.	boolean
SM_BACKUP_SCOPE	Specifies the scope of the backup (full or partial)	string
SM_BACKUP_PROTECTION	Specifies if backup protection is enabled	boolean
SM_TARGET_FILER_NAME	Specifies the 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	Specifies the 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	Specifies the host file system	string
SM_SNAPSHOT_NAMES	Specifies the 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	Specifies the ASM Disk group list	string array
SM_ARCHIVE_LOGS_DIRECTORY	Specifies the 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	Specifies the 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

Variables	Description	Format
SM_CONTROL_FILES_DIRECTORY	Specifies the 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	Specifies the 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	Specifies additional parameters defined by the user. User-defined parameters are not available for plug-ins that are used as policies.	user-defined

Variables available in the task scripts for the restore operation

SnapManager provides context information in the form of environment variables related to the restore operation that is being performed. For example, your script can retrieve the name of the original host and the label of the backup that is restored.

The following table lists the environment variables that you can use in your scripts:

Variables	Description	Format
SM_OPERATION_ID	Specifies the ID of the current operation	string
SM_PROFILE_NAME	Specifies the name of the profile used	string
SM_HOST	Specifies the host name of the database	string
SM_OS_USER	Specifies the operating system (OS) owner of the database	string
SM_OS_GROUP	Specifies the OS group of the database	string
SM_BACKUP_TYPE	Specifies the type of the backup (online, offline, or auto)	string
SM_BACKUP_LABEL	Specifies the backup label	string
SM_BACKUP_ID	Specifies the backup ID	string
SM_BACKUP_PROFILE	Specifies the profile used for the backup	string
SM_RECOVERY_TYPE	Specifies the recovery configuration information	string
SM_VOLUME_RESTORE_MODE	Specifies the volume restore configuration	string
SM_TARGET_FILER_NAME	Specifies the target storage system name **Note:** If more than one storage system is used, then the storage system names must be separated by commas.	string

Variables	Description	Format
SM_TARGET_VOLUME_NAME	Specifies the 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	Specifies the host file system	string
SM_SNAPSHOT_NAMES	Specifies the 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	Specifies the ASM Disk group list	string array
SM_ARCHIVE_LOGS_DIRECTORY	Specifies the 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	Specifies the 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	Specifies the 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	Specifies the 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	Host name associated with the original database	string
SM_ORIGINAL_OS_USER	OS owner of the original database	string

Variables	Description	Format
`SM_ORIGINAL_OS_GROUP`	OS group of the original database	string
`SM_TARGET_SID`	SID of the clone database	string
`SM_TARGET_HOST`	Host name 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
`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 for how to make your own scripts or as a basis for your custom scripts.

About this task

You can find the sample plug-in scripts at the following location:

- `<default_install_directory>/plugins/examples/backup/create`
- `<default_install_directory>/plugins/examples/clone/create`
- `<default_install_directory>/plugins/unix/examples/backup/create/post`

The directory that contains the sample plug-in scripts 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:

Script name	Description	Type of script
`validate_sid.sh`	Contains additional checks to the SID used on the target system. The script checks that the SID has the following characteristics: • Contains three alphanumeric characters • Begins with a letter	Policy
`cleanup.sh`	Cleans the target system so that it is ready to store the newly created clone. Preserves or deletes files and directories depending on the need.	Pretask
`Mirror_the_backup.sh`	Mirrors the volumes after the backup operation occurs in an UNIX environment when you are using either Data ONTAP operating in 7-Mode or clustered Data ONTAP.	Post-task
`Vault_the_backup_cDOT.sh`	Vaults the backup after the backup operation occurs in an UNIX environment when you are using clustered Data ONTAP.	Post-task

Scripts delivered with SnapManager use the BASH shell by default. You must ensure 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 validates database SID names and prevents invalid names from being used in the cloned database. It includes three operations (check, describe, and execute), which are called after you run the script. The script also includes error message handling with codes 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

You can create the pretask, 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 either create a new script or modify one of the SnapManager sample scripts.

Before you begin

Before you start creating the script, ensure that:

- You must structure the script in a particular manner for it to be run in the context of a SnapManager operation.
- You must create the script based on the expected operations, available input parameters, and return code conventions.

- You must include log messages and redirect the messages to user-defined log files.

Steps

1. Create the task script by customizing the sample script.

   Perform the following:

   a. Locate a sample script in the following installation 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 the script with a different name.

2. Modify the functions, variables, and parameters as needed.

3. Save the script in one of the following directories:

   Backup operations scripts

   - <default_install_directory>/plugins/backup/create/pre: Executes the script before the backup operation occurs. Use it optionally when you specify the backup creation.
   - <default_install_directory>/plugins/backup/create/post: Executes the script after the backup operation occurs. Use it optionally 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.

   Restore operation scripts

   - <default_install_directory>/plugins/restore/create/pre: Executes the script before the backup operation occurs. Use it optionally when you specify the backup creation.
   - <default_install_directory>/plugins/restore/create/post: Executes the script after the backup operation occurs. Use it optionally when you specify the backup creation.
   - <default_install_directory>/plugins/restore/create/policy: Always executes the script before the backup operation occurs. SnapManager always uses this script for all the backups in the repository.

   Clone operation scripts

   - <default_install_directory>/plugins/clone/create/pre: Executes the script before the backup operation occurs. Use it optionally when you specify the backup creation.
   - <default_install_directory>/plugins/clone/create/post: Executes the script after the backup operation occurs. Use it optionally when you specify the backup creation.
   - <default_install_directory>/plugins/clone/create/policy: Always executes the script before the backup operation occurs. SnapManager always uses this script for all the backups in the repository.

Storing the task scripts

You must store the pretask, post-task, and policy task scripts in a specified directory on the target server where the backups or clones will be created. For the restore operation, the scripts must be placed in the specified directory on the target server where you want to restore the backup.

Steps

1. Create your script.
2. Save the script in one of the following locations:

For the backup operation

Directory	Description
`<default_install_directory >/plugins/backup/create/policy`	The policy scripts run before the backup operations.
`<default_install_directory >/plugins/backup/create/pre`	The preprocessing scripts run the before backup operations.
`<default_install_directory >/plugins/backup/create/post`	The post-processing scripts run after the backup operations.

For the restore operation

Directory	Description
`<default_install_directory >/plugins/restore/create/policy`	The policy scripts run before the restore operations.
`<default_install_directory >/plugins/restore/create/pre`	The preprocessing scripts run before the restore operations.
`<default_install_directory >/plugins/restore/create/post`	The post-processing scripts run after the restore operations.

For the clone operation

Directory	Description
`<default_install_directory >/plugins/clone/create/policy`	The policy scripts run before the clone operations.
`<default_install_directory >/plugins/clone/create/pre`	The preprocessing scripts run before the clone operations.
`<default_install_directory >/plugins/clone/create/post`	The post-processing scripts run after the clone operations.

Verifying the installation of plug-in scripts

SnapManager enables you to install and use custom scripts to perform various operations. SnapManager provides plugins 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.

Step

1. Enter the following command:

   `smo plugin check -osaccount` ***`os db user name`***

   If you do not provide the `-osaccount` option, verification of the plug-in script installation happens for the root user rather than for a specified user.

   #### Example

   The following output indicates that the policy1, pre-plugin1, and pre-plugin2 scripts have been installed successfully. 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 a task specification file

You can create the task specification files by using graphical user interface (GUI), command-line interface (CLI), or a text editor. These files are used for performing preprocessing or post-processing activity of the backup, restore, or clone operations.

Steps

1. Create a task specification file by using GUI, CLI, or a text editor.

   #### Example

   You can create the specification file based on the structure of the following sample task specification file:

```
<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. Enter the script name.
3. Enter the parameter name and the value assigned to the parameter.
4. Save the XML file in the correct installation directory.

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 prescript and post-scripts

You can use your own 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

- Install the plug-in scripts in the correct SnapManager installation location.

- Verify that the plug-ins are installed correctly by using the `smo plugin check` command.
- Ensure that you are using the BASH shell.

About this task

In the command-line interface (CLI), 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`

   Alternately, you can enter the `which-bash` command at the prompt, and use the command output as the start parameter of the script.

   The BASH shell is operating properly if you do not see an error.

2. For the backup operation, enter the `-taskspec` option and provide the absolute path of the task specification XML file for performing a preprocessing 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]

   If the backup plug-in operation failed, only the plug-in name and return code are displayed. Your plug-in script must include log messages and redirect the messages to the user-defined log files.

3. For the backup restore operation, enter the `-taskspec` option and provide the absolute path of the task specification XML file for performing a preprocessing 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]

   If the restore plug-in operation failed, only the plug-in name and return code are displayed. Your plug-in script must include log messages and redirect the messages to the user-defined log files.

4. For the clone create operation, enter the `-taskspec` option and provide the absolute path of the task specification XML file for performing a preprocessing 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 clone plug-in operation failed, only the plug-in name and return code are displayed. Your plug-in script must include log messages and redirect the messages to the user-defined log files.

Example of creating a backup using the task specification XML file

```
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

SnapManager 3.3 or later allows you to update the storage system host name or storage system address, and the target database host name associated with a SnapManager profile.

Updating the storage system name associated with a profile

SnapManager 3.3 or later provides the ability to update the host name or IP address of a storage system associated with a profile.

Before you begin

You must ensure the following:

- The profile has at least one backup.
  If the profile does not have any backup, then there is no necessity to update the storage system name for that profile.
- No operation is running for the profile.

About this task

You can update the storage system name or IP address by using the SnapManager command-line interface (CLI). While updating the storage system name, the metadata stored in the repository database alone is updated. After renaming the storage system name, you can perform all the SnapManager operations as earlier.

Note: You cannot change the storage system name by using the SnapManager graphical user interface (GUI).

You must ensure that Snapshot copies are available in the new storage system. SnapManager does not verify the existence of the Snapshot copies in the storage system.

However, you must remember the following while performing rolling upgrade and rollback 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 with the new storage system name.
  See *Troubleshooting storage system name issues* for information about how to use the SnapDrive commands for changing the storage system name.
- If you are rolling back the host after renaming the storage system, you must ensure that you change the storage system name back to the earlier storage system name so that you can use the profiles, backups, and clones of the earlier storage system for performing SnapManager operations.

Note: If SnapDrive could not identify the storage system and displays error messages, you can enter the `ipmigrate` command with the earlier and later host names of the storage system. For additional information about storage system name issues, see *Troubleshooting storage system name issues*.

Step

1. Enter the following command:

smo storage rename -profile *profile* -oldname *old_storage_name* -newname *new_storage_name* [quiet | -verbose
]

If you want to...	Then...
Update the storage system name associated with a profile	Specify the `-profile` option.
Update the storage system name or IP address associated with a profile	Specify the following 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.

The following example shows the storage system name being updated:

```
smo storage rename -profile mjullian -oldname lech -newname hudson -verbose
```

Related references

Troubleshooting storage system renaming issue on page 332

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 the target database host name associated with a profile

SnapManager (3.2 or later) for Oracle provides the ability to update the host name of the target database in the SnapManager profile.

Before you begin

- The local user's home directory must be aware of the profile-to-repository mappings.
- The SnapManager graphical user interface (GUI) sessions must be closed.
- In a Real Application Clusters (RAC) environment, the clones or mounted backups available on the host specified in the profile must be deleted and unmounted.

About this task

You can update the profile with the new host name by using only the CLI.

Scenarios not supported for changing the target database host name in profile

The following scenarios not supported for changing the target database host name in the profile:

- Changing the target database host name by using the SnapManager GUI
- Rolling back of the repository database after updating the target database host name of the profile
- Updating multiple profiles for a new target database host name by running a single command
- Changing the target database host name when any SnapManager operation is running
- Changing the target database host name if SnapManager is installed on Solaris and if the database logical unit numbers (LUNs) are created by using host-mounted file system with SVM stack.

Note: After you update the target database host name in the profile, only the target database host name is changed. All the other configuration parameters set on the profile are retained.

After you update the new target database host name in a protection-enabled profile, the same dataset and protection policies are retained for the updated profile.

After you change the host name for the target host, you must ensure that you update the host name for all the existing protected profiles before creating the new protected profiles. To update the host name for a profile, run the `smo profile update` command.

After you update the target database host name, you 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 scenarios, running the SnapManager operations from the new host lead to failure as well as stale entries in the earlier host. To perform SnapManager operations, you must revert to the earlier host name by using `profile update`.

Steps

1. 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]] [-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 you want to...	Then...
Change the target database host name	Specify `-host` *`new_db_host`*

2. To view the target database host name of the profile, enter the following 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. You can maintain the history either from the SnapManager command-line interface (CLI) or graphical user interface (GUI). You can view the history of the operations as a report, and use the report for audit compliance purposes.

You can maintain 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 classes for each of the supported SnapManager operations.

The following are some retention classes that you can assign:

- 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 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 the detailed history of a 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

You can run the `smo_server status` command to view the status of the SnapManager host server.

Syntax

```
smo_server status
[-quiet | -verbose]
```

Parameters

-quiet

Specifies that only error messages are displayed in the console. The default is to display error and warning messages.

-verbose

Specifies that error, warning, and informational messages are displayed in the console.

Example

The following example displays the status of the host server:

```
smo_server status
SMO-17104: SnapManager Server version 3.3.1 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

You can run the `backup create` command to create database backups on one or more storage systems.

Syntax

Note: Before you run this command, you must create a database profile by using the `profile create` command.

Enter the following command to create a database backup:

```
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]
```

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 Real Application Clusters (RAC) database, as long as the primary is in open state, or the primary is mounted and an instance is in open state. You can use `-force` for online backups if the local instance is SHUTDOWN, or no instance is OPEN. The version of Oracle must be 10.2.0.5, else 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 either in 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 `-tablespaces` option.

`-data`

Specifies the data files.

`-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, underscores (_), and hyphens (-). It cannot start with a hyphen.

If you do not specify a label, SnapManager creates a default label in the scope_type_date format, 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 16th January 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 quotation marks (').

Note: Some shells strip the quotation marks off. In this case, you must include the quotation mark 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 in shutdown state, or no instance is in open state.

Note: The version of Oracle must be 10.2.0.5, else the database will hang if any instance in the RAC is mounted.

- If the local instance is in shutdown state and at least one instance is in open state, then using `-force` changes the local instance to mounted state.
- If no instance is in open state, using `-force` changes the local instance to open state.

`-quiet`

Displays only error messages in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in 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` option specifies that the backup be protected 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 System Change Number (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 preprocessing activity or post-processing activity of the backup operation. The complete path of the XML file should be provided while giving 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 -retain -daily
Operation Id [8abc01ec0e79356d010e793581f70001] succeeded.
```

Related concepts

Restoring protected backups from secondary storage on page 170

Related tasks

Creating database backups on page 90

Related references

The smo profile create command on page 281

The smo backup delete command

You can run the `backup delete` command to remove backups that are not automatically removed, such as backups that were used to create a clone or backups that failed. You can delete backups retained on an unlimited basis without changing the retention class.

Syntax

```
smo backup delete
-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 remove. The profile contains the identifier of the database and other database information.

`-id` *`guid`*

Specifies the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. You can use the `smo backup list` command to display the GUID for each backup.

`-label` *`label`*

Specifies the backup with the specified label. Optionally, specify the scope of the backup as data file or archive log.

`-data`

Specifies the data files.

`-archivelogs`

Specifies the archive log files.

`-all`

Specifies 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 in freeing the resources associated with the backup. For example, if the backup was cataloged with Oracle Recovery Manager (RMAN), but the RMAN database no longer exists, including `-force` deletes the backup even though it cannot connect with RMAN.

`-quiet`

Displays only error messages in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in the console.

Example

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 108

Related references

The smo profile create command on page 281
The smo profile update command on page 290

The smo backup free command

You can run the `backup free` command to free 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 profile associated with the backup you want to free. The profile contains the identifier of the database and other database information.

`-id` *`guid`*

Specifies the resources of the backup with the specified GUID. The GUID is generated by SnapManager when you create a backup. You can use the `smo backup list` command to display the GUID for each backup. Include the `-verbose` option to display the backup IDs.

`-label` *`label`*

Specifies the backup with the specified label.

`-data`

Specifies the data files.

`-archivelogs`

Specifies the archive log files.

`-all`

Specifies 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 in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

The following example frees the backup:

```
smo backup free -profile SALES1 -label full_backup_sales_May
Operation Id [8abc01ec0e79004b010e79006da60001] succeeded.
```

Related tasks

Freeing backups on page 107

The smo backup list command

You can run the `backup list` command to display information about 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 profile you want to list backups for. The profile contains the identifier of the database and other database information.

-delimiter *character*

Displays each row on a separate line. The attributes in the row are separated by the character specified.

-data

Specifies the data files.

-archivelogs

Specifies the archive log files.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console. Include the `-verbose` option to display the backup IDs.

Example

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 104

The smo backup mount command

You can run the `backup mount` command to mount a backup in order to perform a recover operation by 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 profile associated with the backup that you want to mount. 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. You can use the `smo backup list` command to display the GUID for each backup.

`-label` *`label`*

Mounts the backup with the specified label.

`-data`

Specifies the data files.

`-archivelogs`

Specifies the archive log files.

`-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.

You must specify the `-copy-id` option whenever you specify the `-from-secondary` option. If there is more than one backup on the secondary storage system, the `-copy-id` option is used to specify which backup copy on the secondary should be used to mount the backup.

Note: If you are using Data ONTAP operating in 7-Mode, you must specify a valid copy id value. However, if you are using clustered Data ONTAP, you can specify any value (for example 0) for the `-copy-id` option and the value will be discarded by SnapManager.

`-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 in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in the console.

Note: You must use this command only if you are using an external tool such as Oracle Recovery Manager (RMAN). SnapManager automatically handles mounting backups if you use the `smo backup restore` command to restore the backup. This command displays a list, which shows the paths where the Snapshot copies have been mounted. This is displayed only when `-verbose` is specified.

Example

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 106

The smo backup restore command

You can run the `backup restore` command to restore backups of a database or a portion of a database and then optionally recover 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. You can 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. You must use spaces to separate the names in the list.
- `-files` *`list`*: Restores only the specified data files from the backup. You must use spaces to separate the names in the list. If the database is running, 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 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.
  You must use the year-month-date: hour: minute: second *(`yyyy-mm-dd:hh:mm:ss`)* format. For hours, use either 12-hour or 24-hour format, depending on the database setting.
- `-until scn`: Rolls forward the data files until it reaches the specified system change number (SCN).
- `-using-backup-controlfile`: Recovers the database using the backup control file.

`-restorespec`

Enables you to 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. If you do not specify an option, SnapManager restores the data from the Snapshot copies on primary storage. You can specify one of the following options:

- `-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 you use a temporary volume, you must specify the volume by using the `-temp-volume` option.
  You must specify the `-copy-id` option whenever you specify the `-from-secondary` option. If there is more than one backup on the secondary storage system, the `-copy-id` option is used to specify which backup copy on the secondary should be used for restore operation.

  Note: If you are using Data ONTAP operating in 7-Mode, you must specify a valid copy id value. However, if you are using clustered Data ONTAP, you can specify any value (for example 0) for the `-copy-id` option and the value will be discarded by SnapManager.

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 file system), then SnapManager will fall back to a host-side file copy restore. SnapManager has two methods for 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, mounts the data from the temporary volume to the host, and then 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 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.

`-preview`

Displays 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 are using the `-preview` option, you must know the following:

- 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.

To preview the restore operation, the database must be mounted. If you want to preview a restore plan, and the database currently is not mounted, then SnapManager mounts the database. If the database cannot be mounted, then the command will fail and SnapManager returns the database to its original state.

The `-preview` option displays up to 20 files. You can configure the maximum number of files to be displayed in the `smo.config` file.

`-fast`

Enables you to choose the process to use in the restore operation. You can force SnapManager to use the volume-based fast restore process rather than other restore processes, if all mandatory restore eligibility conditions are met. If you are aware that a volume restore cannot be performed, you can also use this process to prevent SnapManager from conducting eligibility checks and the restore using the fast restore process.

The `-fast` option includes the following parameters:

- `-require`: Enables you 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 you to override non-mandatory eligibility checks and perform the volume-based fast restore.
- `-fallback`: Enables you to restore the database using any method that SnapManager determines.
  If you do not specify `-fast`, SnapManager uses the default `backup restore -fast fallback`.
- `-off`: Enables you to avoid the time required to perform eligibility checks.

`-recover-from-location`

Specifies the external archive log location of archive log files. SnapManager takes the archive log files from the external location and uses them for the recovery process.

-taskspec

Specifies the task specification XML file for preprocessing activity or post-processing activity of the restore operation. You must provide the complete path of the task specification XML.

-dump

Specifies the collection of dump files after the restore operation.

-force

Changes the database state to a lower state than its current state, if necessary. For Real Application Clusters (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 in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console. Use this option to see why more efficient restore processes could not be used to restore the file.

Example

The following example restores a database 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 115

Related tasks

Restoring backups from an alternate location on page 141
Creating restore specifications on page 139

The smo backup show command

You can use the `backup show` command to display 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 profile for which to show backups. The profile contains the identifier of the database and other database information.

-label *label*

Specifies the label of the backup.

-data

Specifies the data files.

-archivelogs

Specifies the archive log files.

-id *id*

Specifies the backup ID.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console, as well as any clone and verification information.

Example

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_
8abc013111a450480111a45066210001_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 104

The smo backup unmount command

You can run the `backup unmount` command to unmount 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 profile for which you want to unmount a backup. 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. You can use the `smo backup list` command to display the GUID for each backup.

`-label` *`label`*

Unmounts the backup with the specified label.

`-data`

Specifies the data files.

`-archivelogs`

Specifies the archive log files.

`-dump`

Collects the dump files after a successful or failed unmount operation.

`-force`

Unmounts the backup even if there are problems in 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 in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in the console.

Example

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 106

The smo backup update command

You can run the `backup update` command to update the backup retention policy.

Syntax

```
smo backup update
-profile profile_name
[-label label [-data | -archivelogs] | [-id guid]
[-retain {-hourly | -daily | -weekly | -monthly | -unlimited}][-
comment comment_text]
[-quiet | -verbose]
```

Parameters

`-profile` *`profile_name`*

Specifies the profile 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. You can use the `smo backup list` command to display the GUID for each backup.

`-label` *`label`*

Specifies the backup label and scope of the backup as data file or archive log.

`-data`

Specifies the data files.

`-archivelogs`

Specifies the archive log files.

`-comment` *`comment_text`*

Enter text (up to 200 characters) about the backup update. You can include spaces.

`-quiet`

Displays only error messages in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in 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

The following example updates the backup to be set the retention policy to unlimited:

```
smo backup update -profile SALES1 -label full_backup_sales_May
-retain -unlimited -comment save_forever_monthly_backup
```

Related tasks

Changing the backup retention policy on page 102
Retaining backups forever on page 102
Freeing or deleting retention policy exempt backups on page 103

The smo backup verify command

You can run the `backup verify` command 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 profile for which you want to verify a backup. 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. You can 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 in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

The following is an example of verifying 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 102

The smo clone create command

You can run the `clone create` command to create 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
[-asminstance -asmusername asminstance_username  -asmpassword
asminstance_password]]
[-syspassword syspassword][-reserve {yes | no | inherit}]
[-from-secondary {-copy-id id}]
[-no-resetlogs | -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. You can use the `smo backup list -verbose` command to display the GUID for each backup.

-backup-label *backup_label_name*

Specifies to clone the backup with the specified label name.

-current

Specifies to create backup and clone from the current state of the database.

Note: If the database is in `noarchivelog` mode, SnapManager will create an offline backup.

-newsid *new_sid*

Specifies a new, unique Oracle system identifier for the cloned database. The system identifier value is a maximum of eight characters. Oracle does not allow running two databases with the same system identifier 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 within single quotation marks.

Note: Some shells delete the quotation marks. If that is true for your shell, you must escape the quotation 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 an absolute path name.

-asminstance

Specifies the credentials that are used to log in to the ASM instance.

-asmusername *asminstance_username*

Specifies the user name used to log in to the ASM instance.

-asmpassword*asminstance_password*

Specifies the password used to log in to ASM instance.

-syspassword*syspassword*

Specifies the password for the sys privileged user.

Note: You must provide the password for the sys privileged user if the database credentials provided are not the same for the sys privileged user.

-reserve

Setting `-reserve` to **yes** ensures that the volume guarantee space reserve is turned on for the new clone volumes. Setting `-reserve` to **no** ensures that the volume guarantee space reserve is turned off for the new clone volumes. Setting `-reserve` to **inherit** ensures that the new clone inherits the space reservation characteristics of the parent Snapshot copy. 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	Result
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	Result
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]`

Specifies 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.

You must specify the `-copy-id` option whenever you specify the `-from-secondary` option. If there is more than one backup on the secondary storage system, the `-copy-id` option is used to specify which backup copy on the secondary should be used for cloning.

Note: If you are using Data ONTAP operating in 7-Mode, you must specify a valid copy id value. However, if you are using clustered Data ONTAP, you can specify any value (for example 0) for the `-copy-id` option and the value will be discarded by SnapManager.

`-no-resetlogs`

Specifies to skip recovering the database, executing the DBNEWID utility, and not opening the database with the resetlogs while creating the clone.

`-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 preprocessing activity or post-processing activity of the clone operation. Ensure that 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 in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in the console.

Example

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 144
Cloning databases from backups on page 148

The smo clone delete command

You can run the `clone delete` command to delete a clone. You cannot delete a clone if the clone is use by any operation.

Syntax

```
smo clone delete
-profile profile_name
[-id guid | -label clone_name]
[-login
[-username db_username -password db_password -port db_port]
 [-asminstance -asmusername asminstance_username  -asmpassword
asminstance_password]]
[-syspassword syspassword]-force
[-dump][-quiet | -verbose]
```

Parameters

`-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. You can use the `smo clone list` command to display the GUID for each clone.

`-label` *`name`*

Specifies the label for the clone being deleted.

`-asminstance`

Specifies the credentials that are used to log in to the Automatic Storage Management (ASM) instance.

`-asmusername` *`asminstance_username`*

Specifies the user name used to log in to the ASM instance.

`-asmpassword` *`asminstance_password`*

Specifies the password used to log in to ASM instance.

`-syspasswordsyspassword`

Specifies the password for the sys privileged user.

> **Note:** You must provide the password for the sys privileged user if the database credentials provided are not the same for sys privileged user.

`-login`

Allows you to enter the database login details.

-username*db_username*

Specifies the user name required to access the database.

-password*db_password*

Specifies the password required to access the database.

-port*db_port*

Specifies the TCP port number used to access the database that the profile describes.

-dump

Specifies to collect the dump files after the clone delete operation.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

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 153

The smo clone show command

You can run the `clone show` command to display 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. You can 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 in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in the console.

Example

The following example displays information about the clone:

```
smo clone show -profile SALES1 -label full_backup_sales_May -verbose
```

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
```

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: 8abc01ec16514883011651488b580001
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 153

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 144
Cloning databases from backups on page 148

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

You can run the `clone split` command to split a clone. The split clone becomes independent of the original clone. SnapManager generates a new profile after the clone split operation and 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 host name in which the clone exists.

-label *clone-label*

Specifies the label name generated by the clone operation.

-id *clone-id*

Specifies the unique ID generated by the clone operation.

-split-label *split-operation_label*

Specifies the label name generated by the clone operation.

-new-profile *new-profile_name*

Specifies the new profile name that SnapManager will generate after a successful split operation.

-profile-password *new-profile_password*

Specifies the password for the profile.

-repository

Specifies the details of the database for the repository.

-dbname *repo_service_name*

Specifies the name of the database that stores the repository. You can use either the global name or system identifier.

-host *repo_host*

Specifies the name or IP address of the host computer on which the repository database resides.

-port *repo_port*

Specifies the Transmission Control Protocol (TCP) port number used to access the host on which the repository database resides.

-login

Specifies the repository login details. This is optional. If not specified, SnapManager defaults to OS Authentication Connection Mode.

-username*repo_username*

Specifies the user name required to access the host on which the repository database resides.

-database

Specifies the details of the database that will be backed up, restored, or cloned.

-dbname *db_dbname*

Specifies the name of the database that the profile describes. You can use either the global name or system identifier.

-host*db_host*

Specifies the name or IP address of the host computer on which the database resides.

-sid *db_sid*

Specifies the system identifier of the database that the profile describes. By default, SnapManager uses the database name as the system identifier. If the system identifier is different from the database name, you must specify it using the `-sid` option.

For example, if you are using Oracle Real Application Clusters (RAC), you must specify the system identifier of the RAC instance on the RAC node from which SnapManager is executed.

-login

Specifies 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

Specifies the details that SnapManager uses to catalog backups with Oracle Recovery Manager (RMAN).

-controlfile

Specifies the target database control files as the RMAN repository instead of a catalog.

-login

Specifies 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.

For each retention class, either or both the retention count or retention duration might be specified. The duration is in units of the class (for example, hours for hourly, days for daily). For instance, if you specify only a retention duration of 7 for daily backups, then SnapManager will not limit the number of daily backups for the profile (because 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 been created. Snapshot copies that exist retain the previous Snapname pattern. You can use several variables in the pattern text.

-protect -protection-policy ***policy_name***

Specifies 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 the `smo profile update` command or set by the storage administrator by using the Protection Manager's console.

-summary-notification

Specifies the details for configuring summary email notification for multiple profiles under a repository database. SnapManager generates this email.

-notification

Specifies the details for configuring email notification for the new profile. SnapManager generates this email. The email notification enables the database administrator to receive emails on the succeeded or failed status of the database operation that is performed by using this profile.

-success

Specifies that email notification is enabled for a profile for when the SnapManager operation succeeds.

-email*email address 1email address 2*

Specifies the email address of the recipient.

-subject*subject-pattern*

Specifies the email subject.

-failure

Specifies that email notification is enabled for a profile for when the SnapManager operation fails.

-separate-archivelog-backups

Specifies that the archive log backup is separated from the datafile backup. This is an optional parameter, which you can provide while creating the profile. After the backups are separated by using this option, you can either create datafiles-only backup or archive logs-only backup.

-retain-archivelog-backups -hours *hours* | -days *days* | -weeks *weeks*| -months *months*

Specifies that the archive log backups are retained based on the archive log retention duration (hourly, daily, weekly, or monthly).

protect [-protection-policy *policy_name*] | -noprotect

Specifies that the archive log files is protected based on the archive log protection policy.

Specifies that the archive log files are not protected by using the `-noprotect` option.

-include-with-online-backups | -no-include-with-online-backups

Specifies that the archive log backup is included along with the online database backup.

Specifies that the archive log backups are not included along with the online database backup.

-dump

Specifies that the dump files are not collected after the successful profile create operation.

-quiet

Displays only error messages in the console. The default setting displays error and warning messages.

-verbose

Displays error, warning, and informational messages in 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 clone detach command

After splitting a cloned volume from its parent volume in Data ONTAP, you can run the `clone detach` command from SnapManager to let SnapManager know that the volume is no longer a clone.

Syntax

smo clone detach -profile *profile_name* -label *clone_label*

Parameters

-profile *profile_name*

Specifies the profile name from which the clone is created.

-label *clone_label*

Specifies the name generated by the clone operation.

Example

The following command detaches the clone:

```
smo clone detach -profile SALES1 -label sales1_clone
```

The smo cmdfile command

You can use the `cmdfile` command to run any command 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]
```

You can include the command in a text file and use the `smo cmdfile` command to execute the command. You can add only one command in a text file. You must not include smo in the command syntax.

Note: The `smo cmdfile` command replaces the `smo pfile` command. The `smo cmdfile` is not compatible with the `smo pfile` command.

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 in the console. The default is to display error and warning messages.

`-verbose`

Specifies that error, warning, and informational messages are displayed in the console.

Example

This example creates a profile by including the `profile create` command in `command.txt` located at `/tmp`. You can then run the `smo cmdfile` command.

The text file contains the following information:

```
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
```

You can now create the profile by running the `smo cmdfile` command with the `command.txt` 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 63

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 64

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 63

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

How SnapManager maintains security on page 16

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

You can run the `history set` command to configure the operations for which you want to view the history.

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. The name can be up to 30 characters long and must be unique within the host.

-repository

Specifies the details of the database that stores the profile.

-dbname *repo_dbname*

Specifies the name of the database that stores the profile. You can use either the global name or the system identifier.

-host*repo_host*

Specifies the name or IP address of the host where the repository database resides.

-login

Specifies the repository login details.

-username*repo_username*

Specifies the user name required to access the repository database.

-port*repo_port*

Specifies the Transmission Control Protocol (TCP) port number used to access the repository database.

-operation {-operations *operation_name* [*operation_name1*, *operation_name2*] | -all

Specifies the SnapManager operations for which you want to 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, and create and split clone operations. The retention class is set based on the operation count number, number of days, weeks, or months.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example command

The following example displays information about the backup operation:

```
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

You can run the `help` command to display 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

The following are some command names you can use with this command:

- `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

You can run the `notification update-summary-notification` command to enable 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-notification-host notification-host
[-quiet | -verbose]
```

Parameters

-repository

Specifies the details of the repository database.

-port *repo_port*

Specifies the TCP port number used to access the repository database.

-dbname *repo_service_name*

Specifies the name of the repository database. You can use either the global name or the system identifier.

-host *repo_host*

Specifies the name or IP address of the host on which the repository database resides.

-login

Specifies the repository login details. This is optional. If not specified, SnapManager defaults to OS Authentication Connection Mode.

-username *repo_username*

Specifies the user name required to access the repository database.

-email *email-address1,e-mail-address2*

Specifies email addresses of the recipients.

-subject *subject-pattern*

Specifies the email 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 when you want the email notification.

-profiles ***profile1, profile2***

Specifies profile names that require email notification.

-notification-host ***notification-host***

Specifies SnapManager server host from which the summary notification email is sent to the recipients. You can provide host name, or IP address for the notification host. You can also update the host IP or host name.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

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

You can use the `notification set` command 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 sender's email address from which the email alerts are sent. From SnapManager 3.2 for Oracle, you can include a hyphen (-) while specifying the domain name of the email address. For example, you can specify the sender email address as `-sender-email`*`07lbfmdatacenter@continental-corporation.com`*.

-mailhost ***mailhost***

Specifies the name or IP address of the host server that handles email notifications.

-mailport ***mailport***

Specifies the mail server port number.

-authentication -username*username*-password *password*

Specifies authentication details for the email address. You must specify the user name and password.

-repository

Specifies the details of the repository database.

-port*repo_port*

Specifies the Transmission Control Protocol (TCP) port number used to access the repository database.

-dbname*repo_service_name*

Specifies the name of the repository database. You can use either the global name or the system identifier.

-host*repo_host*

Specifies the name or IP address of the host where the repository database resides.

-login

Specifies the repository login details. This is optional. If not specified, SnapManager defaults to OS Authentication Connection Mode.

-username *repo_username*

Specifies the user name required to access the repository database.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

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

You can run the `operation dump` command to create 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 profile 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 in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

The following example creates the dump file for the backup:

```
smo operation dump -profile SALES1
-id 8abc01ec0e78f3e2010e78f3fdd00001
```

```
Dump file created Path:/userhomedirectory/.netapp/smo/3.3/
smo_dump_8abc01ec0e78f3e2010e78f3fdd00001.jar
```

Related concepts

Dump files on page 318

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 190

The smo operation show command

You can run the `operation show` command to list the summary information of all the operations performed against the specified profile. The output lists the client user (the user for the client PC) and the effective user (the user in SnapManager who 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 in the console. The default is to display error and warning messages.

-verbose

Optional: Displays error, warning, and informational messages in the console.

Example

The following command line shows detailed information about 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: 3.3
Repository
   Connection: user1@SMOREPO/hotspur:1521
   Repository version: 3.3
Resources in use
   Volume:
     ssys1:/vol/luke_ES0_0 (FlexClone)
   Filesystems:
     /opt/NetApp/smo/mnt/-mnt_ssys1_luke_ES0_smo_e_es0_f_c_1_8abc0112129b0f81580001_0
```

Related tasks

Viewing operation details on page 190

The smo password reset command

You can run the password `reset command` to reset the password of a profile.

Syntax

```
smo password reset
-profile profile[-profile-password profile_password]
[-repository-hostadmin-password repository_hostadmin_password]
[-quiet | -verbose]
```

Parameters

`-profile` *`profile`*

Specifies the name of the profile for which you want to reset the password.

`-profile-password` *`profile_password`*

Specifies the new password for the profile.

`-repository-hostadmin-password` *`admin_password`*

Specifies the authorized user credential with root privilege for the repository database.

`-quiet`

Displays only error messages in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in the console.

The smo plugin check command

SnapManager enables you to install and use custom scripts for various operations. SnapManager offers backup, restore, and clone plug-ins to automate your custom scripts before and after the backup, restore, and clone operations. Before you use the backup, restore, and clone plug-in, you can run the `plugin check` command to verify the installation of plug-in scripts. 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 preprocessing scripts), and post (for post-processing scripts).

Syntax

```
smo plugin check

-osaccount os_db_user_name
```

Parameter

`-osaccount`

Specifies the operating system (OS) database user name. If you do not enter the `-osaccount` option, SnapManager checks the plug-in scripts as root user rather than for a specific user.

Example

The following example shows that the `plugin check` command found the policy1 custom script stored in the policy directory as an executable. The example also shows that the two other custom scripts stored in the pre directory return no error messages (shown with a status of 0); however, the fourth custom script (post-plug-in1), 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 148

The smo profile create command

You can run the `profile create` command to create a profile of a database in a repository. You must mount the database before you run this command.

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]
 [-asminstance -asmusername asminstance_username  -asmpassword
asminstance_password]]
[-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 system identifier of the database that the profile describes. By default, SnapManager uses the database name as the system identifier. If the system identifier is different from the database name, you must specify it with the `-sid` option.

For example, if you are using Oracle Real Application Clusters (RAC), you must specify the system identifier of the RAC instance on the RAC node from which SnapManager is executed.

-login

Specifies the repository login details.

-username *repo_username*

Specifies the user name needed to access the repository database.

-port *repo_port*

Specifies the TCP port number used to access the repository database.

-database

Specifies 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. You can use either the global name or the system identifier.

-host db_host *db_host*

Specifies the name or IP address of the host computer on which the database runs.

-asminstance

Specifies the credentials that are used to log in to the Automatic Storage Management (ASM) instance.

-asmusernameasminstance_username

Specifies the user name used to log in to the ASM instance.

-asmpasswordasminstance_password

Specifies the password used to log in to ASM instance.

-login

Specifies 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

Specifies the details that SnapManager uses to catalog backups with Oracle Recovery Manager (RMAN).

-controlfile

Specifies the target database control files instead of a catalog as the RMAN repository.

-login

Specifies 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.

-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 (because 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 been created. 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 Protection Manager's console.

`-summary-notification`

Specifies that summary email notification is enabled for the new profile.

`-notification -success -email` *`e-mail_address1,e-mail address2`* `-subject` *`subject_pattern`*

Specifies that email notification is enabled for the new profile so that emails are received by recipients when the SnapManager operation succeeds. You must enter a single email address or multiple email addresses to which email alerts will be sent and an email 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 emails that are not sent. You can use several variables for the email subject.

`-notification -failure -email` *`e-mail_address1,e-mail address2`* `-subject` *`subject_pattern`*

Specifies that enable email notification is enabled for the new profile so that emails are received by recipients when the SnapManager operation fails. You must enter a single email address or multiple email addresses to which email alerts will be sent and an email 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 emails that are not sent. You can use several variables for the email subject.

`-separate-archivelog-backups`

Specifies that the archive log backup is separated from datafile backup. This is an optional parameter you can provide while creating the profile. After you separate the backup using this option, you can either take data files-only backup or archive logs-only backup.

`-retain-archivelog-backups -hours` *`hours`* | `-days` *`days`* | `-weeks` *`weeks`*| `-months` *`months`*

Specifies that the archive log backups are retained 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.

The `-noprotect` option specifies not to protect the archive log files.

`-quiet`

Displays only error messages in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in the console.

`-include-with-online-backups`

Specifies that the archive log backup is included along with the online database backup.

`-no-include-with-online-backups`

Specifies that the archive log backups are not included along with the online database backup.

-dump

Specifies that the dump files are collected after the successful profile create operation.

Example

The following example shows the creation of a profile with hourly retention policy and email notification:

```
smo profile create -profile test_rbac -profile-password netapp -repository -dbname
SMOREP -host hostname.org.com -port 1521 -login -username smorep -database -dbname
RACB -host saal -sid racb1 -login -username sys -password netapp -port 1521 -rman -
controlfile -retain -hourly -count 30 -verbose
Operation Id [8abc01ec0e78ebda010e78ebe6a40005] succeeded.
```

Related concepts

Managing profiles for efficient backups on page 66
Snapshot copy naming on page 72
How SnapManager retains backups on the local storage on page 165

Related references

The smo protection-policy command on page 295

The smo profile delete command

You can run the `profile delete` command to delete a profile of the database.

Syntax

```
smo profile delete
-profile profile
[-quiet | -verbose]
```

Parameters

-profile *profile*

Specifies the profile to be deleted.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

The following example deletes the profile:

```
smo profile delete -profile SALES1
Operation Id [Ncaf00af0242b3e8dba5c68a57a5ae932] succeeded.
```

Related tasks

Deleting profiles on page 78

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

You can run the `profile dump` command to create the `.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 in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

The following example creates a dump for the profile SALES1:

```
smo profile dump -profile SALES1
Dump file created
Path:/userhomedirectory/.netapp/smo/3.3.0/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

You can run the `profile show` command to display the 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 in the console. The default is to display error and warning messages.

`-verbose`

Displays error, warning, and informational messages in the console.

Example

The following example shows the details of the 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

You can run the `profile update` command to update the information for an existing profile.

Syntax

```
smo profile update
-profile profile[-new-profile new_profile_name][-profile-
password profile_password]
[-database
-dbname db_dbname
-host db_host
[-sid db_sid]
[-login
[-username db_username -password db_password -port db_port]
 [-asminstance -asmusername asminstance_username  -asmpassword
asminstance_password]]
[{-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 is set on the profile, you cannot change the policy using SnapManager. You must change the policy using the Protection Manager's console.

-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*

Specifies the password for the profile.

-new-profile*new_profile_name*

Specifies the new name that you can provide for a profile.

-database

Specifies 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. You can use either the global name or the system identifier.

-host *db_host*

Specifies the name or IP address of the host computer on which the database runs.

-sid *db_sid*

Specifies the system identifier of the database that the profile describes. By default, SnapManager uses the database name as the system identifier. If the system identifier is different from the database name, you must specify it using the `-sid` option.

For example, if you are using Oracle Real Application Clusters (RAC), you must specify the SID system identifier of the RAC instance on the RAC node from which SnapManager is executed.

-login

Specifies the repository login details.

-username *repo_username*

Specifies the user name required to access the repository database.

-port *repo_port*

Specifies the TCP port number required to access the repository database.

-database

Specifies 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. You can use either the global name or the system identifier.

-host *db_host*

Specifies the name or IP address of the host computer on which the database runs.

-login

Specifies the database login details.

-username *db_username*

Specifies the user name required to access the database that the profile describes.

-password *db_password*

Specifies the password required to access the database that the profile describes.

-port *db_port*

Specifies the TCP port number required to access the database that the profile describes.

-asminstance

Specifies the credentials that are used to log in to the Automatic Storage Management (ASM) instance.

-asmusernameasminstance_username

Specifies the user name used to log in to the ASM instance.

-asmpassword*asminstance_password*

Specifies the password used to log in to ASM instance.

-rman

Specifies the details that SnapManager uses to catalog backups with Oracle Recovery Manager (RMAN).

-controlfile

Specifies the target database control files instead of a catalog as the RMAN repository.

-login

Specifies 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

Specifies to remove 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.

-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 (because 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 by using the Protection Manager's console .

The `-noprotect` option specifies not to protect the profile to secondary storage.

-summary-notification

Specifies that summary email notification is enabled for the existing profile.

-notification [-success -email *e-mail_address1,e-mail address2* -subject *subject_pattern*]

Enables email notification for the existing profile so that emails are received by recipients when the SnapManager operation succeeds. You must enter a single email address or multiple email addresses to which email alerts will be sent and an email 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 emails that are not sent. You can use several variables for the email subject.

-notification [-failure -email *e-mail_address1,e-mail address2* -subject *subject_pattern*]

Enables email notification for the existing profile so that emails are received by recipients when the SnapManager operation fails. You must enter a single email address or multiple email addresses to which email alerts will be sent and an email 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 emails that are not sent. You can use several variables for the email subject.

-separate-archivelog-backups

Separates the archive log backup from datafile backup. This is an optional parameter you can provide while creating the profile. After you separate the backups are separated using this option, you can create either data files-only backup or archive logs-only backup.

-retain-archivelog-backups -hours *hours* | -days *days* | -weeks *weeks*| -months *months*

Specifies that the archive log backups are retained based on the archive log retention duration (hourly, daily, weekly, monthly).

-protect [-protection-policy *policy_name*] | -noprotect

Specifies that the archive log files are protected based on the archive log protection policy.

Specifies that the archive log files are not protected by using the `-noprotect` option.

-include-with-online-backups | -no-include-with-online-backups

Specifies that the archive log backup is included along with the online database backup.

Specifies that the archive log backups are not included along with the online database backup.

-dump

Specifies that the dump files are collected after the successful profile create operation.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

The following example changes the login information for the database described by the profile and the email notification is configured for this profile:

```
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 retains backups on the local storage on page 165

Related tasks

Changing profile passwords on page 74

The smo profile verify command

You can run the `profile verify` command to verify the profile set up. You must mount the database before running this command.

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 in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

The following example verifies the profile:

```
smo profile verify -profile test_profile  -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_profile 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 75

The smo protection-policy command

You can run the `protection-policy` command to list the protection policies that can be applied to a profile. The protection policy can be applied when a new profile is created or an existing profile is updated. You can also set the protection policy for the profile using the Protection Manager console.

Syntax

```
smo protection-policy list
```

Note: The Protection Manager and SnapDrive must be installed on the server for you to use this command.

Parameters

`list`

Displays the list of protection policies that can be set on a profile.

Example

The following example lists the protection policies that can be set to a profile:

```
smo protection-policy list
```

```
Back up
Back up, then mirror
Chain of two mirrors
DR Back up
DR Back up, then mirror
DR Mirror
DR Mirror and back up
DR Mirror and mirror
DR Mirror, then back up
DR Mirror, then mirror
Local backups only
Mirror
Mirror and back up
Mirror to two destinations
Mirror, then back up
No protection
Partial-volume Mirror
Remote backups only
```

Related concepts

Managing profiles for efficient backups on page 66

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
```

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

You can use the `schedule create` command to schedule a backup to be created at a specific time.

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 that you want to schedule the backup for. 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 create an online backup of a Real Application Clusters (RAC) database, as long as the primary is in open state, or the primary is in mounted state and an instance is open. You can use `-force` for online backups if the local instance is in shutdown state, or no instance is open.

- If the local instance is in shutdown state and at least one instance is open, using `-force` you can change the local instance to mounted.
- If no instance is in open state, using `-force` can change the local instance to open.

`-offline`

Specifies an offline backup while the database is in shutdown state. 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 shutdown 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 `-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 in open state, 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 in open state, 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 in the scope_type_date format, 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 16th January 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 within single quotation marks (').

Note: Some shells strip quotation marks off. If that is true for your shell, you must include the quotation mark 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 in shutdown state, or no instance is open.

Note: The version of Oracle must be 10.2.0.5 , else the database will hang if any instance in the RAC is mounted.

- If the local instance is in shutdown state and at least one instance is open, you can change the local instance to mounted by using `-force`.
- If no instance is open, you can change the local instance to open by using `-force`.

`-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` option specifies that the backup be protected 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 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 system change number (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}`

Specifies the time interval by which the backups are created. You can schedule the backup on an hourly, daily, weekly, monthly, or one time only basis.

`-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 the following subexpressions:

- 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 scheduled operation. The schedule start time should be included in the yyyy-MM-dd HH:mm format.

-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 preprocessing activity or post-processing activity of the backup operation. The complete path of the XML file must be provided with the `-taskspec` option.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in 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 list command

You can run the `storage list` command to display the list of storage systems associated with a particular profile.

Syntax

```
smo storage list
-profile profile
```

Parameters

-profile *profile*

Specifies the name of the profile. The name can be up to 30 characters long and must be unique within the host.

Example

The following example displays the storage systems associated with the profile mjullian:

```
smo storage list -profile mjullian
```

```
Sample Output:
Storage Controllers
--------------------
FAS3020-RTP07OLD
```

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 the storage system is renamed. You must enter the IP address or name of the storage system that is displayed when you run the `smo storage list` command.

-newname ***new_storage_name***

Specifies the IP address or name of the storage system after the storage system is renamed.

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example

The following example uses the `smo storage rename` command to rename the storage system:

```
smo storage rename -profile mjullian -oldname lech -newname hudson -
verbose
```

The smo system dump command

You can run the `system dump` command to create a JAR file that contains diagnostic information about the server environment.

Syntax

```
smo system dump
[-quiet | -verbose]
```

Parameters

-quiet

Displays only error messages in the console. The default is to display error and warning messages.

-verbose

Displays error, warning, and informational messages in the console.

Example of the system dump command

The following example uses the `smo system dump` command to create a JAR file:

```
smo system dump
Path:/userhomedirectory/.netapp/smo/3.3.0/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

You can run the `version` command to determine which version of SnapManager you are running on your local host.

Syntax

```
smo version
[-quiet | -verbose]
```

Parameters

`-quiet`

Displays only error messages in the console. The default is to display error and warning messages.

`-verbose`

Displays the build date and contents of each profile. Also displays error, warning, and informational messages in the console.

Example of the version command

The following example displays the version of the SnapManager:

```
smo version
SnapManager for Oracle Version: 3.3.1
```

Troubleshooting SnapManager

You can find information about some of the most common issues that might occur and how you can resolve them.

The following table describes common issues and possible solutions:

Issue-driven question	Possible solution
Are the target database and listener running?	Run the `lsnrctl status` command. Ensure that the database instance is registered with the listener.
Is the storage visible?	Run the `snapdrive storage show -all` command.
Is the storage writable?	Edit a file in the mountpoint that you just created. Use the `touch filename` command. If the file is created, then your storage is writable. You must ensure that the storage is writable by the user that SnapManager runs as (for example, as root on UNIX).
Is the SnapManager server running?	Run `smo_server status` and try to start the server by using the `smo_server start` command. Before you can use the graphical user interface (GUI) or the command-line interface (CLI) to initiate SnapManager commands related to profiles, the server must be running. You can create or update repositories without starting the server, but to execute all other SnapManager operations, the server must be running. To start the SnapManager server, enter the following command: `smo_server start`.
Are all the components required to run SnapManager set up correctly?	Run the `smo system verify` command to verify that SnapDrive is set up correctly.
Do you have the correct version of SnapManager?	Use the `smo version` command to check the SnapManager version.
Have you looked at the SnapManager log files to determine if the error messages can help isolate the issue?	SnapManager records all log entries into one set of rotating log files. The log files are found at `/var/log/smo`. The log files are found at `C:\program_files\NetApp \SnapManager for Oracle\logs`. It might also be helpful to look at the logs in the following location: `/usr_home/.netapp/smo/3.3.0/log/` Each operation log is written to its own log file of the form `smo_of_date_time.log`.

Issue-driven question	Possible solution
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 UNIX, the files are at the following location: `/opt/NetApp/smo/properties/smo.config` Use the format mentioned in the file to exclude the local archive logs. For additional information, see the “Setting configuration properties” topic. You can also exclude the archive log destinations while creating a backup from the SnapManager CLI. For additional information, see the “Creating database backups” topic. You can also exclude the archive log destinations while creating a backup from the SnapManager GUI.
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)
Were you unable to connect to the repository?	If connecting to a repository fails, run the `lsnrctl status` command 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 set up, this might 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 whether you want to remove the repository. You can remove the repository from your current view so that you can perform operations on other repositories. Also, check whether the repository instance is running by running the `ps -eaf |grep` *`instance - name`* command.
Can system resolve the host name?	Check whether the specified host name is on a different subnet. If you receive an error message that SnapManager cannot resolve the host name, then add the host name in the host file. Add the host name to the file located at `/etc/hosts`: **`xxx.xxx.xxx.xxx hostname`** ***`IP address`***
Is SnapDrive running?	Check whether the SnapDrive daemon is running: **`-snapdrived status`** If the daemon is not running, a message appears indicating that there is a connection error.

Issue-driven question	Possible solution
Which storage systems are configured to be accessed with SnapDrive?	Run the command: **`-snapdrive config list`**
How can SnapManager GUI performance be improved?	• 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 set before for the repository, profile host, and profile. For additional information about setting the user credentials again, see “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. • Check whether you enabled **Open On Startup** in the User Preferences window under the **Admin** menu, from the SnapManager GUI. If this is enabled, then the user configuration (`user.config`) file available at `/root/.netapp/smo/3.3.0/gui/state` is displayed as `openOnStartup=PROFILE`. Because **Open On Startup** is enabled, you must check for recently opened profiles from the SnapManager GUI, using `lastOpenProfiles` in the user configuration (`user.config`) file: `lastOpenProfiles=`*`PROFILE1,PROFILE2,PROFILE3,...`* You can delete the profile names listed and always keep a minimum number of profiles as open. • The protected profile takes more time to refresh than the profile that is not protected. The protected profile is refreshed at a time interval, based on the value specified in the `protectionStatusRefreshRate` parameter of the user configuration (`user.config`) file. You can increase the value from the default value (300 seconds) so that the protected profiles are refreshed only after specified time interval. • Before installing the new version of SnapManager on the UNIX-based environment, delete the SnapManager client-side entries available at the following location: `/root/.netapp`

Issue-driven question	Possible solution
SnapManager GUI takes more time to refresh when there are multiple SnapManager operations started and running simultaneously in the background. When you right-click the backup (that is already deleted but still gets displayed in the SnapManager GUI), the backup options for that backup are not enabled in the Backup or Clone window.	You need to wait until the SnapManager GUI gets refreshed, and then check for the backup status.
What would you do when the Oracle database is not set in English?	SnapManager operations might fail if the language for an Oracle database is not set to English. Set the language of the Oracle database to English: **1.** Add the following under the initial comments in `/etc/init.d/smo_server` • NLS_LANG=American_America • export NLS_LANG **2.** Restart the SnapManager server using the following command: `smo_server restart` **Note:** If the login scripts such as `.bash_profile`, `.bashrc`, and `.cshrc` for the Oracle user is set to **`NLS_LANG`**, you must edit the script to not overwrite **`NLS_LANG`**.
What would you do when the backup scheduling operation fails if the repository database points to more than one IP and each IP has a different host name?	**1.** Stop the SnapManager server. **2.** Delete the schedule files in the repository directory from the hosts where you want to trigger the backup schedule. The schedule file names can be in the following formats: • `repository#repo_username#repository_database_name#repository_host#repo_port` • `repository-repo_usernamerepository_database_name-repository_host-repo_port` **Note:** You must ensure that you delete the schedule file in the format that matches the repository details. **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.

Issue-driven question	Possible solution
What would you do when the SnapManager operation fails with credential file lock error?	SnapManager locks the credential file before updating, and unlocks it after updating. When multiple operations run simultaneously, one of the operations might lock the credential file to update it. If another operation tries to access the locked credential file at the same time, the operation fails with the file lock error. Configure the following parameters in the `smo.config` file depending on the frequency of simultaneous operations: • `fileLock.retryInterval` = 100 milliseconds • `fileLock.timeout` = 5000 milliseconds **Note:** The values assigned to the parameters must be in milliseconds.
What would you do when the backup verify operation's intermediate status shows failed in the Monitor tab even though the backup verify operation is still running?	The error message is logged in the `sm_gui.log` file. You must look in the log file to determine the new values for the `operation.heartbeatInterval` and `operation.heartbeatThreshold` parameters which will resolve this issue. **1.** Add the following parameters in the `smo.config` file: • `operation.heartbeatInterval` = 5000 • `operation.heartbeatThreshold` = 5000 The default value assigned by SnapManager is 5000. **2.** Assign the new values to these parameters. **Note:** The values assigned to the parameters must be in milliseconds. **3.** Restart the SnapManager server and perform the operation again.
What to do when you encounter a heap-space issue?	When you encounter a heap-space issue during SnapManager for Oracle operations, you must perform the following steps: **1.** Navigate to the SnapManager for Oracle installation directory. **2.** Open the `launchjava` file from the *`installationdirectory`*`/bin/launchjava` path. **3.** Increase the value of the `java -Xmx160m` Java heap-space parameter. For example, you can increase the default value of 160m to 200m. **Note:** If you have increased the value of the Java heap-space parameter in the earlier versions of SnapManager for Oracle, you should retain that value.

Issue-driven question	Possible solution
What would you do if you cannot use the protected backups to restore or clone?	This issue is observed if you were using SnapManager 3.3.1 with clustered Data ONTAP and have upgraded to SnapManager 3.4. The backups were protected using post-scripts in SnapManager 3.3.1. From SnapManager 3.4, the backups are protected using either *SnapManager_cDOT_Mirror* or *SnapManager_cDOT_Vault* policies which are selected while creating a profile. After upgrading to SnapManager 3.4, you might still be using the old profiles and thus backups are protected using backup scripts, but you cannot use them for restore or cloning using SnapManager. You must update the profile and select either *SnapManager_cDOT_Mirror* or *SnapManager_cDOT_Vault* policy and delete the post-script that was used for data protection in SnapManager 3.3.1.
What would you do if scheduled backups are not getting protected (SnapVault)?	After upgrading to SnapManager 3.4 and updating the profile to use *SnapManager_cDOT_Vault* policy for protection, you must delete old backup schedules and create new schedules to specify the SnapVault label while creating the schedule.

Dump files

The dump files are compressed log files containing information about SnapManager and its environment. The different types of log files created are operation, profile, and system dump file.

You can use the `dump` command or the **Create Diagnostics** tab in the graphical user interface (GUI) to collect information about an operation, a profile, or the environment. A system dump does not require a profile; however, the profile and operation dumps require profiles.

SnapManager includes the following diagnostic information in the dump file:

- The steps performed
- The length of time for each step to complete
- The outcome of each step
- Error, if any, 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
- Oracle OPatch local inventory details
- Automatic Storage Management (ASM) instance OPatch local inventory details
- Storage system version
- 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 Real Application Clusters (RAC))
- Target database role (primary, physical standby, or logical standby)
- Target database Oracle Recovery Manager (RMAN) setup (no RMAN integration, RMAN with control files, or RMAN with catalog file)
- Target database ASM instance version
- Target database Oracle version
- System identifier (SID) of the target database
- RMAN database name and TNS connection name
- Repository database service name
- Database instances installed on the host
- Profile descriptor
- Shared memory maximum
- Swap space information
- Memory information
- Kernel version
- FSTAB
- Protocol used by Snapdrive
- Multipath environment
- RAC
- Supported volume manager
- Operations Manager version
- Supported file system
- Host utilities version
- Output of the `system verify` command
- Output of the `sdconfcheck` command

SnapManager dump files also contain the SnapDrive data collector file and the Oracle alert log file. You can collect the Oracle alert log file by using the `smo operation dump` and `smo profile dump` commands.

Note: System dump does not contain Oracle alert logs; however, the profile and operation dumps contain the alert logs.

Even if the SnapManager host server is not running, you can access the dump information by using the command-line interface (CLI) or the GUI.

If you encounter a problem that you cannot resolve, you can send these files to NetApp Global Services.

Creating operation-level dump files

You can use the `smo operation dump` command with the name or ID of the failed operation to get log information about a particular operation. You can specify different log levels to gather information about a specific operation, profile, host, or environment.

Step

1. Enter the following 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>
   /.netapp/smo/3.3.0/smo_dump_8abc01c814649ebd0114649ec69d0001.jar
   ```

Creating profile-level dump files

You can find log information about a particular profile by using the `smo profile dump` command with the name of the profile.

Step

1. Enter the following command:

   `smo profile dump -profile` ***`profile_name`***

   Dump file location:

   ```
   Path:/<user-home>
   /.netapp/smo/3.3.0/smo_dump_8abc01c814649ebd0114649ec69d0001.jar
   ```

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

Creating system-level dump files

You can use the smo system dump command to get log information about the SnapManager host and environment. You can specify different log levels to collect information about a specific operation, profile, or host and environment.

Step

1. Enter the following command:

   `smo system dump`

   Resulting dump

   ```
   Path:/<user-home>/.netapp/smo/3.3.0/smo_dump_server_host.jar
   ```

How to locate dump files

The dump file is located at the client system for easy access. These files are helpful if you need to troubleshoot a problem related to profile, system, or any operation.

The dump file is located in the user's home directory on the client system.

- If you are using the graphical user interface (GUI), the dump file is located at:

  ```
  user_home/Application Data/NetApp/smo/3.3.0/smo_dump dump_file_type_name
  server_host.jar
  ```

- If you are using the command-line interface (CLI), the dump file is located at:

  ```
  user_home/.netapp/smo/3.3.0/smo_dump_dump_file_type_name server_host.jar
  ```

The dump file contains the output of the dump command. The name of the file depends on the information supplied. 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	`smo operation dump -profile` *`VH1`* `-verbose`
	The following output is displayed:
	`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/3.3.0/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

You can include `-dump` in the SnapManager command to collect the dump files after a successful or failed SnapManager operation.

You can collect dump files for the following SnapManager operations:

- Creating profiles
- Updating profiles
- Creating backups
- Verifying backups
- Deleting backups
- Freeing backups
- Mounting backups
- Unmounting backups
- Restoring backups
- Creating clones
- Deleting clones
- Splitting clones

Note: When you create a profile, you can collect dump files only if the operation is successful. If you encounter an error while creating a profile, you must use the `smo system dump` command. For successful profiles, you can use the `smo operation dump` and `smo profile dump` commands to collect the dump files.

Example

```
smo  backup create -profile targetdb1_prof1 -auto -full -online -dump
```

Collecting additional log information for easier debugging

If you need additional logs to debug a failed SnapManager operation, you must set an external environment variable `server.log.level`. This variable overrides the default log level and dumps all the log messages in the log file. For example, you can change the log level to DEBUG, which logs additional messages and can assist in debugging issues.

The SnapManager logs can be found at the following locations:

- `/var/log/smo`

To override the default log level, you must perform the following steps:

1. Create a `platform.override` text file in the SnapManager installation directory.
2. Add the `server.log.level` parameter in the `platform.override` text file.
3. Assign a value (*TRACE, DEBUG, INFO, WARN, ERROR, FATAL, or PROGRESS*) to the `server.log.level` parameter.
   For example, to change the log level to *ERROR*, set the value of `server.log.level` to *ERROR*.

   `server.log.level=ERROR`

4. Restart the SnapManager server.

   Note: If the additional log information is not required, you can delete the `server.log.level` parameter from the `platform.override` text file.

SnapManager manages the volume of server log files based on the user-defined values of the following parameters in the `smo.config` file:

- `log.max_log_files`
- `log.max_log_file_size`
- `log.max_rolling_operation_factory_logs`

Troubleshooting clone issues

You can find information about that might occur during clone operations and how you can resolve them.

Symptom	Explanation	Workaround
The clone operation fails when the archive destination is set to *`USE_DB_RECOVERY_FILE_DEST`*.	When the archive destination is referring to USE_DB_RECOVERY_FILE_DEST, Flash recovery area (FRA) actively manages the archive log. SnapManager does not use the FRA location during clone or restore operations and thus the operations fail.	Change the archive destination to actual archive log location instead of the FRA location.

Symptom	Explanation	Workaround
The clone operation fails with the following error message: `Cannot perform operation: Clone Create. Root cause: ORACLE-00001: Error executing SQL: [ALTER DATABASE OPEN RESETLOGS;]. The command returned: ORA-01195: online backup of file 1 needs more recovery to be consistent.`	This issue occurs if Oracle listener fails to connect to the database.	If you are using SnapManager GUI to clone a backup, perform the following actions: **1.** From the Repository tree, click **Repository** > **Host** > **Profile** to display the backups. **2.** Right-click the backup that you want to clone and select **Clone**. **3.** On the Clone Initialization page, enter the mandatory values and select the clone specification method. **4.** On the Clone Specification page, select **Parameters**. **5.** Click the **+Parameter** tab. **6.** In the **Parameter Name** field, enter the name as *`local_listener`* and click **OK**. **7.** Select the **Override Default** check box for the local_listener row. **8.** Click any parameter, then double-click the local_listener parameter, and enter the following value: **`(ADDRESS=(PROTOCOL=TCP)(HOST=<your_host_name>)(PORT=<port#>))`** **9.** Click **Save To File**. **10.** Click **Next** and continue with the clone create wizard. If you are using CLI to clone a backup, you must include the following information in the **<parameters>** tag of the clone specification file: `<parameters> <parameter> <name>local_listener</name> <value>(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=<hostname>)(PORT=<port#>)))</value> </parameter> </parameters>`

Symptom	Explanation	Workaround
The clone operation fails with an error message saying that the mountpoint you are using is already in use.	SnapManager does not let you mount a clone over an existing mountpoint. So an incomplete clone did not remove the mountpoint.	Specify a different mountpoint to be used by the clone, or unmount the problematic mountpoint.
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.	• 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.	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 if a new profile is not created after the clone split operation. Because the prompt is not displayed, you might assume that the profile is created.	From the SnapManager command-line interface (CLI), enter the `clone split-result` command to view the detailed result of the clone split operation.
SnapManager for Oracle fails to clone Oracle 10gR2 (10.2.0.5) physical Oracle Data Guard 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. Because the backup is inconsistent, the cloned database requires recovery, and thus Oracle fails to create the clone successfully.	Upgrade the Oracle database to the Oracle 11gR1 (11.1.0.7 patch).

Symptom	Explanation	Workaround
Cloning a backup to a remote host fails with the following error message `Error: Access is denied`.	While mounting, if the IP address of the host is provided to the snap mount command, the cloning operation might fail. This issue occurs if the host on which the database resides is in workgroup while the remote host is in domain, or vice-versa.	You must ensure that both remote host and the host on which the database resides are in the domain and not in the workgroup.

Troubleshooting graphical user interface issues

You can find information about some common known graphical user interface (GUI) issues that might help you resolve them.

Issue	Explanation	Workaround
While accessing the SnapManager GUI to perform an operation, the following error message might be displayed: `SMO-20111 : Authentication failed for user on host.`	This issue occurs if the password of the user is changed in the host on which the SnapManager server is running. After the password is changed, the credential cache that is created for the user who launched the GUI becomes invalid. SnapManager GUI still uses the credentials in the cache to authenticate and thus the authentication fails.	You must perform one of following tasks: • Delete the credentials of the user whose password was changed and then add the new credentials in the cache by running the following commands: **1.** `smo credential delete` **2.** `smo credential set` • Clear the entire cache by running the `smo credential clear` command. Re-open the GUI and set the credentials, if prompted.
Security warning is displayed while using Java Web Start to access SnapManager GUI.	While accessing SnapManager GUI using Java Web Start, a security warning is displayed. This issue occurs because JNLP jars are self-signed and Java version used by SnapManager does not allow self-signed jars at high security level.	Either change the security settings to medium in the java control panel or add the SnapManager GUI URL to the exception list.

Issue	Explanation	Workaround
The SnapManager web start GUI displays the incorrect version.	After downgrading SnapManager from a later version to an earlier version when you launch the web start GUI, the later version of the SnapManager web start GUI is launched.	You must also clear the cache by performing the following steps: **1.** Start the console. **2.** Enter the following: `javaws -viewer` **3.** On the Java cache viewer screen, right-click the SnapManager application and select **Delete**.
When you restart the GUI and try to check the backups for a certain profile, you see only the names of the profiles.	SnapManager does not display any information about a profile until you open it.	Perform the following steps: **1.** Right-click the profile and select **Open** from the menu. SnapManager displays the Profile Authentication dialog box. **2.** Enter the host user name and password. SnapManager displays the backup list. **Note:** You only need to authenticate the profile once as long as the credentials are valid and remain in the cache.
When you open the first repository in the GUI, an error message similar to the following is displayed: `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 operating system (OS) users or rename the profile by issuing an SQL statement for the repository: `UPDATE SMO_33_PROFILE SET NAME = 'NEW_NAME' WHERE NAME = 'OLD_NAME'`
An error message similar to the following is displayed: `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 GUI 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 GUI view. This removes the reference to the repository in the credentials file and the GUI does not attempt to load the repository again.

Issue	Explanation	Workaround

Profile creation fails because host credentials fail to authenticate in the SUSE Linux Enterprise Server 10 and SUSE Linux Enterprise Server 11 platforms.	SnapManager uses Pluggable Authentication Module (PAM) 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. Thus the host credentials fail.	To successfully log in to the host in the SUSE Linux Enterprise Server 10 and 11 platforms, perform the following steps:
		1. Create a `snapmanager` file in `/etc/pam.d/`.
		2. Add the following content to 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.

SnapManager takes a longer time to load the database tree structure and results in a timeout error message being displayed on the SnapManager GUI.	When you try to perform a partial backup operation from the SnapManager GUI, SnapManager tries to load the credentials for all the profiles, and if there are any invalid entries, SnapManager tries to validate the entry and this results in a timeout error message being displayed.	Delete the credentials of the unused host, repository, and profile by using the `credential delete` command from the SnapManager command-line interface (CLI).
SnapManager fails to generate a new profile after the clone split operation and you do not know if the new profile is created.	SnapManager fails to prompt you if a new profile is not created after the clone split operation. Because no message is displayed for the failed operation, you might assume that the profile is created.	To know if a new profile is created for the clone split operation, perform the following steps: **1.** Click the **Monitor** tab, right-click the clone split operation entry and select **Properties**. **2.** In the Profile Properties window, click the **Logs** tab to view the clone split operation and profile creation logs.

Issue	Explanation	Workaround
The custom scripts for the preprocessing or postprocessing 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 start the respective wizard, the custom scripts are not displayed under the Available Scripts list.	Restart the SnapManager host server and then open the SnapManager GUI.
You cannot use the clone specification XML file created in SnapManager (3.1 or earlier) for the 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 SnapManager 3.2 for Oracle, you must remove the task specification section from the clone specification XML or create a new clone specification XML file. SnapManager 3.3 or later does not support the clone specification XML file created in SnapManager 3.2 or earlier releases.

Issue	Explanation	Workaround
SnapManager operation on the GUI does not proceed after you have cleared user credentials by using the `smo credential clear` command from the SnapManager CLI or by clicking **Admin** > **Credentials** > **Clear** > **Cache** from the SnapManager GUI.	The credentials set for the repositories, hosts, and profiles are cleared. SnapManager verifies user credentials before starting any operation. When user credentials are invalid, SnapManager fails to authenticate. When a host or a profile is deleted from the repository, the user credentials are still available in the cache. These unnecessary credential entries slow down the SnapManager operations from the GUI.	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 the SnapManager GUI. • If you have deleted the encrypted credential file manually, you must restart the SnapManager GUI. Set the credentials that you have given for the repository, profile host, and profile. From the SnapManager GUI, if there is no repository mapped under the Repositories tree, perform the following steps: **1.** Click **Tasks** > **Add Exisiting repository** **2.** Right-click the repository, click **Open**, and enter the user credentials in the **Repository Credentials Authentication** window. **3.** Right-click the host under the repository, click **Open**, and enter the user credentials in **Host Credentials Authentication**. **4.** Right-click the profile under the host, click **Open**, and enter the user credentials in **Profile Credentials Authentication**.

Issue	Explanation	Workaround
The error message `Unable to list the protection policies for the following reason: Protection Manager is temporarily unavailable` is displayed when you select **None** from the **Protection Manager Protection Policy** drop-down menu of the Profile Properties window and the policy settings page of the Profile create wizard.	The Protection Manager is not configured with SnapManager or the Protection Manager is not running.	No action is necessary.
You cannot open the SnapManager GUI by using Java Web Start GUI due to weaker Secure Sockets Layer (SSL) cipher strength of the browser.	SnapManager does not support SSL ciphers weaker 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

Symptom	Explanation
`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
`0001-878 Admin error: HBA assistant not found. Commands involving LUNs should fail.`	LUN discovery error

Troubleshooting storage system renaming issue

You might face issues when renaming a storage system or after you have successfully renamed the storage system.

When you try to rename the storage system, the operation might fail with the following error message: `SMO-05085 No storage controller "fas3020-rtp07New" is found to be associated with the profile`

You must enter the IP address or name of the storage system that is listed when you run the `smo storage list` command.

After you rename the storage system, SnapManager operations might fail if SnapManager fails to recognize the storage system. You must perform some additional steps in the DataFabric Manager server host and the SnapManager server host to resolve this issue.

Perform the following steps in the DataFabric Manager server host:

1. Delete the IP address and host of the earlier storage system in the host file located at `/etc/hosts` in the DataFabric Manager server host.
2. Add the new IP address and host of the new storage system in the host file located at `/etc/hosts` in the DataFabric Manager server host.
3. Change the storage host name by entering the following command:

   `dfm host rename -a` ***`old_host_name new_host_name`***

4. Set the new IP address in the host by entering the following command:

   `dfm host set` ***`old_host_name_or_objId`*** **`hostPrimaryAddress =`** ***`new_storage_controller_ip_address`***

   Note: You must perform this step only if you have specified the IP address as the new storage system name.

5. Update the new storage system name in the DataFabric Manager server host by entering the following command:

   `dfm host diag` ***`old_storage_name`***

You can verify that the earlier storage controller name is replaced with new storage controller name by entering the following command:

dfm host discover *new_storage_name*

Perform the following steps as a root user in the SnapManager server host.

Note: When you enter the new storage controller name, ensure that you use the system alias name and not the fully qualified domain name (FQDN).

1. Delete the earlier storage system name by entering the following command:

   snapdrive config delete *old_storage_name*

   Note: If you do not delete the earlier storage system name, then all the SnapManager operations fail.

2. Delete the IP address and host of the earlier storage system in the host file located at `etc/hosts` in the target database host.
3. Add the new IP address and host of the new storage system in the host file located at `/etc/hosts` in the target database host.
4. Add the new storage system name by entering the following command:

   snapdrive config set root *new_storage_name*

5. Map the earlier and later storage system names by entering the following command:

   snapdrive config migrate set *old_storage_name new_storage_name*

6. Delete the management path of the earlier storage system by entering the following command:

   snapdrive config delete -mgmtpath *old_storage_name*

7. Add the management path for the new storage system by entering the following command:

   snapdrive config set -mgmtpath *new_storage_name*

8. Update the dataset for both data files and archive log files with the new storage system name by entering the following command:

   snapdrive dataset changehostname -dn *dataset_name* -oldname *old_storage_name* -newname *new_storage_name*

9. Update the profile for the new storage system name by entering the following command:

   smo storage rename -profile *profile_name* -oldname *old_storage_name* -newname *new_storage_name*

10. Verify the storage system associated with the profile by entering the following command:

    smo storage list -profile *profile_name*

Troubleshooting known issues

You should be aware of some known issues that might occur when you use SnapManager, and how to work around them

The server fails to start

When starting the server, you might see an error message similar to the following:

```
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
```

This might be because the SnapManager listening ports (27214 and 27215, by default) are currently in use by another application.

This error can also occur if the `smo_server` command is already running, but SnapManager does not detect the existing process.

Workaround

You can reconfigure either SnapManager or the other application to use different ports.

To reconfigure SnapManager, edit the following file: `/opt/NTAP/smo/properties/smo.config`

You assign the following values:

- SMO Server.port=27214
- SMO Server.rmiRegistry.port=27215
- remote.registry.ocijdbc.port= 27215
  The remote.registry.ocijdbc.port must be the same as Server.rmiRegistry.port.

To start the SnapManager server, enter the following command:

```
smo_server start
```

Note: An error message is displayed if the server is already running.

If the server is already running, perform the following steps:

1. Stop the server by entering the following command:

   ```
   smo_server stop
   ```

2. Restart the server by entering the following command:

   ```
   smo_server start
   ```

Terminating a currently running SnapManager operation

If SnapManager server hangs and you cannot execute any operations successfully, you can terminate SnapManager and its operations.

Workaround

SnapManager works with both SnapManager and Protection Manager. You must perform the following steps to list the different processes running and stop the last process running.

1. List all SnapDrive processes that are running:

   ```
   ps
   ```

   Example:

   ```
   ps | grep snapdrive
   ```

2. Stop the SnapDrive process or processes:

   ```
   kill <pid>
   ```

   pid is the list of processes you found using the `ps` command.

   Note: Do not stop all SnapDrive processes. You might want to end only the last process that is running.

3. If one of the operations involves restoring a protected backup from secondary storage, open the Protection Manager console and perform the following:
   a. From the System menu, select **Jobs**.
   b. 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:
   a. Log in as a root user.
   b. List the processes by using the `ps` command.
      Example:

      ```
      ps | grep java
      ```
5. End the SnapManager process:

   ```
   kill <pid>
   ```

Deleting or freeing the last protected backup

When you create the first backup for a profile on secondary storage, SnapManager sends all the information about the backup to Protection Manager. For subsequent backups related to this profile, SnapManager sends only the modified information. If you remove the last protected backup, SnapManager loses the ability to identify the differences between backups and must to find a way to rebaseline these relationships. Therefore, attempting to delete the last protected backup results in an error message being displayed.

Workaround

You can delete the profile or only the profile backup.

To delete the profile, perform the following steps:

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 only the backup, perform the following steps:

1. Create another backup copy of the profile.
2. Transfer that backup copy to secondary storage.
3. Delete the previous backup copy.

Unable to manage archive log file destination names if the destination names are part of other destination names

While creating an archive log backup, if the user excludes a destination that is part of other destination names, then the other destination names are also excluded.

For example, assume that there are three destinations available to be excluded: `/dest`, `/dest1`, and `/dest2`. While creating the archive log file backup, if you exclude `/dest` by using the command

```
smo backup create -profile almsamp1 -data -online -archivelogs  -exclude-
dest /dest
```

, SnapManager for Oracle excludes all the destinations starting with `/dest`.

Workaround

- Add a path separator after destinations are configured in `v$archive_dest`. For example, change the `/dest` to `/dest/`.
- While creating a backup, include destinations instead of excluding any destination.

Restoring control files that are multiplexed on Automatic Storage Management (ASM) and non-ASM storage fails

When the control files are multiplexed on ASM and non-ASM storage, the backup operation is successful. However, when you try to restore control files from that successful backup, the restore operation fails.

SnapManager clone operation fails

When you clone a backup in SnapManager, the DataFabric Manager server might fail to discover volumes, and display the following error message:

```
SMO-13032: Cannot perform operation: Clone Create. Root cause: SMO-11007:
Error cloning from snapshot: FLOW-11019: Failure in ExecuteConnectionSteps:
SD-00018: Error discovering storage for /mnt/datafile_clone3: SD-10016:
Error executing snapdrive command "/usr/sbin/snapdrive storage show -
fs /mnt/datafile_clone3": 0002-719 Warning: Could not check SD.Storage.Read
access on volume filer:/vol/SnapManager_20091122235002515_vol1 for user
user-vm5\oracle on Operations Manager servers x.x.x.x
```

```
Reason: Invalid resource specified. Unable to find its Id on Operations
Manager server 10.x.x.x
```

This occurs if the storage system has large number of volumes.

Workaround

You must perform one of the following:

- From the Data Fabric Manager server, run

  `dfm host discover` ***`storage_system`***

  .

  You can also add the command in a shell script file and schedule a job in the DataFabric Manager server to run the script at a frequent interval.

- Increase the value of *`dfm-rbac-retries`* in the `Snapdrive.conf` file.

  SnapDrive uses the default refresh interval value and default number of retries. The default value of *`dfm-rbac-retry-sleep-secs`* is 15 seconds and *`dfm-rbac-retries`* is 12 iterations.

  > **Note:** The Operations Manager refresh interval depends on the number of storage systems, number of storage objects in the storage system, and the load on the DataFabric Manager server.

  As a recommendation, perform the following:

  1. From the DataFabric Manager server, manually run the following command for all the secondary storage systems associated with the dataset:

     `dfm host discover` ***`storage_system`***

2. Double the time taken to perform the host discovery operation and assign that value to *`dfm-rbac-retry-sleep-secs`*.
   For example, if the operation took 11 seconds, you can set the value of *`dfm-rbac-retry-sleep-secs`* to 22 (11*2).

Repository database size grows with time and not with the number of backups

The repository database size grows with time because SnapManager operations insert or delete data within the schema in the repository database tables, which results in high index space usage.

Workaround

You must monitor and rebuild the indexes according to the Oracle guidelines to control the space consumed by the repository schema.

The SnapManager GUI cannot be accessed and SnapManager operations fail when the repository database is down

SnapManager operations fail and you cannot access the GUI when the repository database is down.

The following table lists the different actions you might want to perform, and their exceptions:

Operations	Exceptions
Opening a closed repository	The following error message is logged in `sm_gui.log`: `[WARN ]: SMO-01106: Error occurred while querying the repository: Closed Connection java.sql.SQLException: Closed Connection`.
Refreshing an opened repository by pressing F5	A repository exception is displayed in the GUI and also logs a NullPointerException in the `sm_gui.log` file.
Refreshing the host server	A NullPointerException is logged in the `sumo_gui.log` file.
Creating a new profile	A NullPointerException is displayed in the Profile Configuration window.
Refreshing a profile	The following SQL exception is logged in `sm_gui.log`: `[WARN ]: SMO-01106: Error occurred while querying the repository: Closed Connection`.
Accessing a backup	The following error message is logged in `sm_gui.log`: `Failed to lazily initialize a collection`.
Viewing clone properties	The following error message is logged in `sm_gui.log` and `sumo_gui.log`: `Failed to lazily initialize a collection`.

Workaround

You must ensure that the repository database is running when you want to access the GUI or want to perform any SnapManager operations.

Unable to create temporary files for the cloned database

When temporary tablespace files of the target database are placed in mount points different from the mount point of the data files, the clone create operation is successful but SnapManager fails to create temporary files for the cloned database.

Workaround

You must perform either of the following:

- Ensure that the target database is laid out so that temporary files are placed in the same mount point as that of the data files.
- Manually create or add temporary files in the cloned database.

Unable to migrate the protocol from NFSv3 to NFSv4

You can migrate the protocol from NFSv3 to NFSv4 by enabling the *`enable-migrate-nfs-version`* parameter in the `snapdrive.conf` file. During the migration, SnapDrive considers only the protocol version, irrespective of the mount point options such as `rw`, `largefiles`, `nosuid`, and so on.

However, after migrating the protocol to NFSv4, when you restore the backup that was created by using NFSv3, the following occurs:

- If NFSv3 and NFSv4 are enabled at the storage level, the restore operation is successful but is mounted with the mount point options that were available during backup.
- If only NFSv4 is enabled at the storage level, the restore operation is successful and only the protocol version (NFSv4) is retained.
  However, the other mount point options such as `rw`, `largefiles`, `nosuid`, and so on are not retained.

Workaround

You must manually shut down the database, unmount the database mount points, and mount with the options available prior to the restore.

Back up of Data Guard Standby database fails

If any archive log location is configured with the service name of the primary database, the back up of Data Guard Standby database fails.

Workaround

In the GUI, you must clear **Specify External Archive Log location** corresponding to the service name of the primary database.

Mounting a FlexClone volume fails in NFS environment

When SnapManager creates a FlexClone of a volume in an 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/NTAPsmo/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 issues 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 the HA pair configuration, ensure that you set the NFS exports option to `on` for both the storage systems.

Running multiple parallel operations fails in SnapManager

When you run multiple parallel operations on separate databases that reside on the same storage system, the igroup for LUNs associated with both the databases might get deleted because of one of the operations. Later, if the other operation attempts to use the deleted igroup, SnapManager displays an error message.

For example, if you are running the backup delete and backup create operations on different databases almost at the same time, the backup create operation fails. The following sequential steps show what happens when you run backup delete and backup create operations on different databases almost at the same time.

1. Run the `backup delete` command.
2. Run the `backup create` command.
3. The `backup create` command identifies the already existing igroup and uses the same igroup for mapping the LUN.
4. The `backup delete` command deletes the backup LUN, which was mapped to the same igroup.
5. The `backup delete` command then deletes the igroup because there are no LUNs associated with the igroup.
6. The `backup create` command creates the backup and tries to map to the igroup that does not exist, and therefore the operation fails.

What to do

You must create igroup for each storage system used by the database and use the following command to update SDU with the igroup information:

```
snapdrive igroup add
```

Unable to restore RAC database from one of the RAC nodes where the profile was not created

In an Oracle RAC environment where both nodes belong to the same cluster, if you attempt a restore operation from a node which is different from the node where the backup was created, the restore operation fails.

For example, if you create a backup in Node A and try to restore from Node B, the restore operation fails.

What to do

Before performing restore operation from node B, perform the following in node B:

1. Add the repository.

2. Sync the profile by running the command `smo profile sync`.
3. Set the credential for the profile to be used for restore operation by running the command `smo credential set`.
4. Update the profile to add the new host name and the corresponding SID by running the command `smo profile update`.

Where to go for more information

You can find information about the basic tasks involved in installing and using SnapManager.

Document	Description
SnapManager description page	This page provides information about SnapManager, pointers to online documentation, and a link to the SnapManager download page, from which you can download the software.
Data ONTAP SAN Configuration Guide for 7-Mode	This document is available at *mysupport.netapp.com*. 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 and SnapDrive Compatibility Matrix	This document is available in the Interoperability section at *mysupport.netapp.com/matrix*. 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 *mysupport.netapp.com*. It contains any last-minute information that you need to get the configuration up and running smoothly.
NetApp host attach and support kits documentation	*mysupport.netapp.com*.
System Configuration Guide	*mysupport.netapp.com*.
Data ONTAP Block Access Management Guide	*mysupport.netapp.com*
Host operating system and database information	These documents provide information about your host operating system and database software.

Error message classifications

You can determine the cause of an error if you know the message classifications.

The following table provides information about the numerical ranges for the different types of messages you might see with SnapManager:

Group	Range	Usage
ENVIRONMENT	1000-1999	Used to log the state or issues with the operating environment of SnapManager. This group includes messages about the systems that SnapManager interacts with, such as the host, storage system, database, and so on.
BACKUP	2000-2999	Associated with the database backup process.
RESTORE	3000-3999	Associated with the database restore process.
CLONE	4000-4999	Associated with the database clone process.
PROFILE	5000-5999	Used for managing profiles.
MANAGE	6000-6999	Used for managing backups.
VIRTUAL DATABASE INTERFACE	7000-7999	Associated with the virtual database interface.
VIRTUAL STORAGE INTERFACE	8000-8999	Associated with the virtual storage interface.
REPOSITORY	9000-9999	Associated with the Repository interface.
METRICS	10000-10999	Associated with the size of the database backup, elapsed time to perform the backup, time to restore the database, the number of times a database has been cloned, and so on.
VIRTUAL HOST INTERFACE	11000-11999	Associated with the virtual host interface. This is the interface to the host operating system.
EXECUTION	12000-12999	Associated with the execution package, including spawning and processing operating system calls.
PROCESS	13000-13999	Associated with the process component of SnapManager.
UTILITIES	14000-14999	Associated with SnapManager utilities, global context, and so on.
DUMP/DIAGNOSTICS	15000-15999	Associated with dump or diagnostic operations.
HELP	16000-16999	Associated with help.
SERVER	17000-17999	Used in the SnapManager server administration.
API	18000-18999	Associated with the API.
AUTH	20000-20999	Associated with the authorization of credentials.

Error messages

You can find information about the 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
`[SMO-05503] You have specified the same name to the profile. Specify a different name to rename the profile.`	Profiles with identical names cannot exist in a repository.	Provide a profile name that is not in use.
`SMO-05505: Unable to update dataset metadata.`	Dataset might have been deleted or does not exist.	Before updating the dataset metadata, verify tjat the dataset exists by using the NetApp Management Console.
`SMO-05506: You cannot update the profile since there are operation(s) running on the profile. You must wait until the operation(s) complete and then update the profile.`	Profile cannot be updated when backup, restore, and cloning operations are in progress.	Update the profile after completion of the current operation.
`SMO-05509: Invalid archivelog primary retention duration - Specify a positive integer value.`	Retention duration of archive log backups cannot be negative.	Specify a positive value for the retention duration of archive log backups.
`SMO-07463: This backup restore requires the database to be in required state. Failed to bring the database to the required state.`	Database is not in the required state for a backup operation.	Ensure that the database is in a relevant state before creating a backup copy. The state of the database that is to be restored depends on the type of restore process that you want to perform and the type of files that are to be included.

Error message	Explanation	Resolution
`SMO-09315: After performing repository upgrade or update operation, you might not receive the summary notification for notifications set in previous version unless you update the summary notification with the notification host details.`	Notification settings are not configured for the repository after a rolling upgrade.	After a rolling upgrade, update summary notification settings to receive notifications.
`SM0-02076: Label name should not contain any special characters other than underscore.`	Label name contains special characters other than the underscore.	The label name must be unique within the profile. The name can contain letters, numbers, an underscore (_), and a hyphen (-) (but cannot start with a hyphen). Ensure that labels do not contain any special characters except the underscore.
`SMO-06308: Exception when attempting to start schedule: java.lang.NullPointerExceptio n`	The fully qualified domain name (FQDN) of the profile host is configured instead of the system's host name and the FQDN of the profile host cannot be resolved.	Ensure that you use the system's host name and not the FQDN.
`Failure in ExecuteRestoreSteps: ORACLE-10003: Error executing SQL "DROP DISKGROUP;control diskgroup name; INCLUDING CONTENTS" against Oracle database +ASM1: ORA-15039: diskgroup not dropped ORA-15027: active use of diskgroup; "control diskgroup name;" precludes its dismount`	The operation to restore a backup with control files fails to drop the control disk group. This issue occurs if there are stale backed up control files in the control disk group.	Identify the stale backed up control files and manually delete them.

Error message	Explanation	Resolution
`RMAN-06004: ORACLE error from recovery catalog database: ORA-01424: missing or illegal character following the escape character`	Backup create operation failed to remove the backup copy from the catalog when SnapManager is integrated with RMAN.	Check if there are any external scripts used for removing the backups from RMAN. Execute the `CROSSCHECK BACKUP` command in RMAN to update the RMAN repository and the `resync catalog` command to synchronize the control file of the target database with the recovery catalog.
`[DEBUG]: Exception while pruning backup. java.lang.IllegalStateException: [Assertion failed] - this state invariant must be true`	Multiple Snapshot copies are created for a single operation ID.	Delete the Snapshot copies manually and use scripts to delete the entries from the repository.
System time and the time displayed by SnapManager in the log files do not match or not synchronized.	A time zone change is not yet supported by Java 7.	Apply the `tzupdater` patch provided by Oracle.
`DISC-00001: Unable to discover storage: The following identifier does not exist or is not of the expected type: ASM File`	Data or control files or redo logs are multiplexed in an ASM database.	Remove the Oracle multiplexing.
`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.	Right-click the **Computer** icon on your desktop and select **Manage** to verify that the user account for the SnapManager service is part of ORA_DBA group. 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.

Error message	Explanation	Resolution
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 file system permissions allow these paths to be discovered.	ASM disks were connected to the host, but the ASM instance is not able to discover them.	If ASM over NFS is being used, ensure that the ASM_DISKSTRING parameter for the ASM instance includes the ASM disk files. For example, if the error states: smo/mnt/<dir_name>/<disk_name>, then add /smo/mnt/*/* to asm_diskstring.
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	After the protection policy of a dataset is set, SnapManager will not allow you to change the protection policy, because it might require realigning the baseline relationships and result in the loss of existing backups on the secondary storage.	Update the protection policy using Protection Manager's Management Console, 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.	The host is not able to discover LUNs created on the storage systems.	Ensure that the transport protocol is properly installed and configured. Ensure that 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_name"	You tried to restore using the indirect storage method and the temporary volume specified does not exist on the primary storage.	Create a temporary volume on the primary storage. Or, specify the correct volume name, if a temporary volume is already created.
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). This issue occurs because 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 that are not backed up as part of this operation (because the database was not open during this backup).

Error message	Explanation	Resolution
`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 the secondary storage of the latest protected backup.	Clone or mount 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 Protection Manager.	Install Protection Manager and configure SnapDrive to use Protection Manager.
`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.`	You tried 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"`	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.`	There are more number of archive log files in the database than the maximum allowed.	**1.** Navigate to the SnapManager installation directory. **2.** Open the `launch-java` file. **3.** Increase the value of the Java heap space parameter `java -Xmx160m`. For example, you can modify the value from the 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 displays the backups as protected even if the protection job in 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/']`	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 the 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.
`SMO-13032: Cannot perform operation: Archive log Prune. Root cause: RMAN Exception: ORACLE-00101: Error executing RMAN command.`	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 operation on the archive log files again.

Error message	Explanation	Resolution

`Unable to parse shell output: (java.util.regex.Matcher[ pattern=Command complete. region=0,18 lastmatch=]) does not match (name:backup_script)`
`Unable to parse shell output: (java.util.regex.Matcher[ pattern=Command complete. region=0,25 lastmatch=]) does not match (description:backup script)`
`Unable to parse shell output: (java.util.regex.Matcher[ pattern=Command complete. region=0,9 lastmatch=]) does not match (timeout: 0)`
	Environment variables are set not set correctly in the pre-task or post-task scripts.	Check if the pre-task or post-task scripts follow the standard SnapManager plug-in structure. For additional information about using the environmental variables in the script, see *Operations in task scripts* on page 205.

`ORA-01450: maximum key length (6398) exceeded.`	When you perform an upgrade from SnapManager 3.2 for Oracle to SnapManager 3.3 for Oracle, the upgrade operation fails with this error message. This issue might occur because of one of the following reasons: • The block size of the tablespace in which the repository exists is less than 8k. • The `nls_length_semantics` parameter is set to `char`.	You must assign the values to the following parameters: • `block_size=`**`8192`** • `nls_length=`**`byte`** After modifying the parameter values, you must restart the database. For more information, see the Knowledge Base article 2017632.

Error messages associated with the database backup process (2000 series)

The following table lists the common errors associated with the database backup 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".`	The archive log backup is taken along with the 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.`	The archive log backup is associated with the database backup and is within 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.`	You exclude some archive log files during the profile create or backup create operation.	No action is necessary.
`SMO-07155: <count> archived log files do not exist in the active file system. These archived log files will not be included in the backup.`	The archive log files do not exist in the active file system during the profile create or backup create operation. These archived log files are not included in the backup.	No action necessary.
`SMO-07148: Archived log files are not available.`	No archive log files are created for the current database during the profile create or backup create operation.	No action necessary.
`SMO-07150: Archived log files are not found.`	All the archive log files are missing from the file system or excluded during the profile create or backup create operation.	No action necessary.

Error message	Explanation	Resolution
`SMO-13032: Cannot perform operation: Backup Create. Root cause: ORACLE-20001: Error trying to change state to OPEN for database instance dfcln1: ORACLE-20004: Expecting to be able to open the database without the RESETLOGS option, but oracle is reporting that the database needs to be opened with the RESETLOGS option. To keep from unexpectedly resetting the logs, the process will not continue. Please ensure that the database can be opened without the RESETLOGS option and try again.`	You try to back up the cloned database that was created with the `-no-resetlogs` option. The cloned database is not a complete database. However, you can perform SnapManager operations such as creating profiles and backups, splitting clones, and so on with the cloned database, but the SnapManager operations fail because the cloned database is not configured as a complete database.	Recover the cloned database or convert the database into a Data Guard Standby database.

Data protection errors

The following table shows the common errors associated with data protection:

Error message	Explanation	Resolution
`Backup protection is 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 try 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 but the Protection Manager is temporarily unavailable. Please try again later.`	You try to delete a profile that has protection enabled; however, Protection Manager is unavailable.	Ensure that appropriate backups are stored in either primary or secondary storage. Disable protection in the profile. When Protection Manager is available again, return to the profile and delete it.

Error message	Explanation	Resolution
`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 Protection Manager Management Console.	Disable protection in the profile temporarily. Continue creating a new profile or updating an existing profile. When Protection Manager is available again, return to the profile.
`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 Protection Manager's Management Console. The Protection Manager is not installed or SnapDrive is not configured.	Install Protection Manager. Configure SnapDrive. Return to the profile, reenable protection, and select the protection policies available in Protection Manager's Management Console.
`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 Protection Manager's Management Console.	Disable protection in the profile temporarily. Continue creating or updating the profile. When Protection Manager's Management Console is available, return to the profile.
`Creating new dataset <dataset_name> for database <dbname> on host <host>.`	You attempted to create a backup profile. SnapManager creates 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 Protection Manager's Management Console. The Protection Manager is not installed.	Install Protection Manager.
`Deleted dataset <dataset_name> for this database.`	You deleted a profile. SnapManager will delete the associated dataset.	No action is necessary.

Error message	Explanation	Resolution
`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, Protection Manager is no longer installed, or no longer configured, or has expired. SnapManager will delete the profile, but not the profile's dataset from Protection Manager's Management Console.	Reinstall or reconfigure Protection Manager. 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.
`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 a 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's Management Console 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 is 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 Protection Manager's Management Console.

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)?`	You tried to disable protection for a protected profile while updating the profile from the 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** check box 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's Management Console, which unregisters all of the secondary and tertiary backup copies associated with that dataset. After a dataset is deleted, all secondary and tertiary backup copies are orphaned. Neither the Protection Manager nor SnapManager for Oracle have the ability to access those backup copies. The backup copies can no longer be restored by 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 is no workaround. The orphaned backups need to be managed manually.

Error messages associated with the restore process (3000 series)

The following table shows the 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.

Error message	Explanation	Resolution
`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.
`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 online redo log files or archive log files cannot 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 do not have sufficient change numbers to apply for recovery. This occurs when the database is online without any transactions. The redo log or archived log files do not have any valid change numbers that can 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 cannot 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, ensure that the file has full access permissions to Oracle. Even if the file has full permissions, and the message continues, the archive log file does 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 the primary storage.	Always restore from the 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.`	During recovery, 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 the 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 clustered Data ONTAP.	SnapManager supports clone split on the FlexClone technology only.
`SMO-04904: No clone split operation running with split-id` *`split_id`*	The operation ID is invalid or no 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 by using the **`clone split-status`** or **`clone split-result`** command.

Error message	Explanation	Resolution
`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 by using RMAN for taking the data files 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.
`[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]`	You did not provide a value for a parameter in the clone specification file.	You must either provide a value for the parameter or delete that parameter if it is not required from the clone specification file.

Error messages associated with the managing profile process (5000 series)

The following table shows the 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.`	The dump operation cannot be performed when profile creation fails.	Use `smo system dump`.

Error messages associated with freeing backup resources (backups 6000 series)

The following table shows the 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 unlimited retention policy. If clones exist, delete them.

Error message	Explanation	Resolution
`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 successful.	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 the common errors associated with virtual storage interface tasks:

Error message	Explanation	Resolution
`SMO-08017 Error discovering storage for /.`	SnapManager attempted to locate storage resources, but found data files, control files, or logs in the `root/` directory. These files should reside in a subdirectory. 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 data files, control files, or redo logs are in the `root` directory. 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 the 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 the SnapManager CLI.

Error message	Explanation	Resolution
`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 the SnapManager CLI.
`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 of the SnapManager host that has mounted backups. These backups are not mounted in the earlier version of the SnapManager host.	Unmount the backups in the later version of the 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 of the SnapManager host that has backups that are being unmounted.	Mount the backups in the later version of the SnapManager host, and then perform the rollback.
`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.`	You attempted to perform 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 the 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 `lsnrctl status` command 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 the 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 creating a profile, you customized the Snapname pattern; however, you included special characters that are not allowed.	Remove special characters other than letters, numbers, underscore, dash, and 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* on page 72.
`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 *`smid`* variable.	Insert the required *`smid`* variable.
`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. • FlexVol volume has restricted Snapshot copies.	Check for the available space in the volume by using the **`clone split-estimate`** command. Confirm that the FlexVol volume has no restricted Snapshot copies.

Error message	Explanation	Resolution
`SMO-13904: Clone Split Result failed.`	This could be due to failure in 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 metadata 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 the common errors associated with SnapManager utilities:

Error message	Explanation	Resolution
`SMO-14501: Mail ID cannot be blank.`	You did not enter the email address.	Enter a valid email address.
`SMO-14502: Mail subject cannot be blank.`	You did not enter the email subject.	Enter the appropriate email subject.
`SMO-14506: Mail server field cannot be blank.`	You did not enter the email 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 email port number.	Enter the email server port number.
`SMO-14508: From Mail ID cannot be blank.`	You did not enter the sender's email address.	Enter a valid sender's email address.
`SMO-14509: Username cannot be blank.`	You enabled authentication and did not provide the user name.	Enter the email authentication user name.
`SMO-14510: Password cannot be blank. Please enter the password.`	You enabled authentication and did not provide the password.	Enter the email authentication password.
`SMO-14550: Email status <success/failure>.`	The port number, mail server, or receiver's email address is invalid.	Provide proper values during email configuration.
`SMO-14559: Sending email notification failed: <error>.`	This could be due to invalid port number, invalid mail server, or invalid receiver's mail address.	Provide proper values during email configuration.
`SMO-14560: Notification failed: Notification configuration is not available.`	Notification sending failed, because notification configuration is not available.	Add notification configuration.

Error message	Explanation	Resolution
`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.
`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 the common 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
`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

Error message	Explanation
`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 72

Copyright information

Copyright © 1994–2015 NetApp, Inc. All rights reserved. Printed in the U.S.

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. 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

NetApp, the NetApp logo, Go Further, Faster, AltaVault, ASUP, AutoSupport, Campaign Express, Cloud ONTAP, Clustered Data ONTAP, Customer Fitness, Data ONTAP, DataMotion, Fitness, Flash Accel, Flash Cache, Flash Pool, FlashRay, FlexArray, FlexCache, FlexClone, FlexPod, FlexScale, FlexShare, FlexVol, FPolicy, GetSuccessful, LockVault, Manage ONTAP, Mars, MetroCluster, MultiStore, NetApp Insight, OnCommand, ONTAP, ONTAPI, RAID DP, RAID-TEC, SANtricity, SecureShare, Simplicity, Simulate ONTAP, Snap Creator, SnapCenter, SnapCopy, SnapDrive, SnapIntegrator, SnapLock, SnapManager, SnapMirror, SnapMover, SnapProtect, SnapRestore, Snapshot, SnapValidator, SnapVault, StorageGRID, Tech OnTap, Unbound Cloud, and WAFL and other names are trademarks or registered trademarks of NetApp, Inc., in the United States, and/or other countries. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such. A current list of NetApp trademarks is available on the web at *http://www.netapp.com/us/legal/netapptmlist.aspx*.

How to send comments about documentation and receive update notifications

You can help us to improve the quality of our documentation by sending us your feedback. You can receive automatic notification when production-level (GA/FCS) documentation is initially released or important changes are made to existing production-level documents.

If you have suggestions for improving this document, send us your comments by email to *doccomments@netapp.com*. To help us direct your comments to the correct division, include in the subject line the product name, version, and operating system.

If you want to be notified automatically when production-level documentation is released or important changes are made to existing production-level documents, follow Twitter account @NetAppDoc.

You can also contact us in the following ways:

- NetApp, Inc., 495 East Java Drive, Sunnyvale, CA 94089 U.S.
- Telephone: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Support telephone: +1 (888) 463-8277

Index

A

active/active SFRAC environment
    configuring SnapDrive *53*
archive log
    pruning *98*
archive log backups
    protecting *99*
archive log file handling *86*
archive log files
    pruning *12*
archive logs
    consolidating *12*
ASM databases
    cloning *142*
    ensuring discovery of imported disks *54*
    requirements with SnapManager *19*
    storage side file restore *116*
    support without ASMLib *22*
ASM databases without ASMLib
    deploying scripts *23*
    limitations *23*
    running scripts *23*
    sample script *25*
    scripts *22*
    support for *22*
ASMLib
    versions that SnapManager supports *21*
authentication
    about *56*
    overview *56*
    profiles *66*
AutoSupport
    clustered Data ONTAP *100*
    description *100*
    disabling *101*
    enabling *100*
    enabling in SnapManager *101*
    understanding *100*

B

backup
    examples *84*
    restore *115*
    restoring *188*
    Snapshot copies *82*
    type *82*
    using prescript and post-scripts *216*
    verifying *102*
backup copies
    protecting *168*
backup create
    command *228*
backup delete
    command *232*
backup free
    command *233*
backup list
    command *234*
backup mount
    command *235*
backup restore
    command *237*
backup retention class
    modifying *38*
backup schedule
    creating *110*
backup schedules
    related tasks *110*
backup show
    command *240*
backup unmount
    command *242*
backup update
    command *243*
backup verify
    command *244*
backups
    about protection policies *159*
    about restoring *116*
    cloning *142*
    cloning protected backups *172*
    control and archive log files *86*
    creating *90*
    enabling protection to secondary storage *163*
    freeing *107*
    full and partial *82*
    local *179*
    mounting *106*
    on primary storage *11*
    on secondary systems *11*
    protection states *160*
    restoring *171*
    restoring from alternate location *137*
    restoring from primary storage *131*
    restoring from secondary storage *170*
    restoring with RMAN *135*
    retention policy *165*
    scheduling *110*
    unmounting *106*
    viewing details *104*
    what you can back up *11*
backups created
    for a profile *104*
backups, databases
    cloning *13*

C

clone
    deleting *154*
    splitting *155*
    using prescript and post-scripts *216*
clone delete
    command *248*
clone information
    viewing *153*

clone show
    command *250*
clone specifications
    creating *144*
clone split
    command *254*
clones, database backups
    creating *13*
cloning
    about *142*
    ASM databases *142*
    backups *142*
    creating clone specifications *144*
    database in current state *150*
    database uses an spfile *142*
    databases from backups *148*
    entering comments *142*
    files created *142*
    in Direct NFS environment *142*
    prerequisites *142*
    protected backups *172*
    RAC databases *142*
    sample plugin scripts *210*
    using custom plug-in scripts *148*
    variables in custom plugin scripts *209*
cloning backup
    without resetlogs *150*
cloning database
    alternate host *151*
    requirements *151*
clustered Data ONTAP
    limitations *33*
cmdfile
    command *261*
collect
    dump files *322*
command
    backup create *228*
    backup delete *232*
    backup free *233*
    backup list *234*
    backup mount *235*
    backup restore *237*
    backup show *240*
    backup unmount *242*
    backup update *243*
    backup verify *244*
    clone create *245*
    clone delete *248*
    clone detach *260*
    clone show *250*
    clone split *254*
    cmdfile *261*
    help *272*
    history set *270*
    notification set *275*
    notification update-summary-notification *274*
    operation dump *276*
    operation show *278*
    password reset *279*
    plugin check *280*
    profile create *281*
    profile delete *285*
    profile dump *286*
    profile show *288*
    profile update *290*
    profile verify *294*
    protection-policy *295*
    schedule create *303*
    smo_server status *227*
    storage list *309*
    storage rename
        command *309*
    system dump *310*
    version *311*
commands
    reference to all *226*
comments
    how to send feedback about documentation *366*
configuration
    for protecting backups *179*
configuration parameters
    editing *53*
    SnapManager *47*
configurations, storage
    finding recommendations for *18*
configure SnapDrive
    data protection *161*
control file handling *86*
create
    policy script *202*
    post-task script *202*
    pretask script *202*
create clone
    command *245*
create detach
    command *260*
creating
    database backups *90*
    profiles *67*, *179*
credentials
    clearing *63*
    deleting *64*
    overview of tasks and components *56*
    setting *64*
    viewing *63*
credentials, user
    role of, security purposes *16*
customizing
    email subject *198*

D

data protection
    configure SnapDrive *161*
    enabling in profile *161*
    introducing *159*
    using protection manager *174*
database
    backing up *80*
    cloning in current state *150*
    sample volume layouts *27*
database backups
    about *81*
database profiles *14*
database requirements

for NFS and SnapManager *26*
database volume layouts
sample *27*
databases
about backing up *81*
about restoring *116*
backing up *90*
block-level restore operations with RMAN *135*
cloning *142*
cloning from backups *148*
cloning using custom plug-in scripts *148*
creating backups *90*
disk group requirements *35*
finding recommended layouts for *18*
Oracle, deprecated versions *34*
RAC *142*
requirements for using with NFS and SnapManager *26*
restoring from alternate location *137*
restoring from primary storage *131*
restoring from secondary storage *170*
setting Oracle home directory *18*
databases, full
restore operations for *12*
databases, partial
restore operations for *12*
delete
backup
deleting *108*
clone *154*
profile *158*
Snapshot copies *107*
Direct NFS environment
cloning *142*
direct storage connection method *116*
disk groups
avoiding layout and configuration issues *18*
requirements *35*
documentation
how to receive automatic notification of changes to *366*
how to send feedback about *366*
where to find it *340*
dump files
collecting *322*
creating *320*
information *318*
locating *321*
operation level *320*
profile level *320*
system-level *321*

E

email notification
configuring *193*
configuring for existing profile *197*
configuring for new profile *195*
disabling *200*
email subject
customizing *196*, *198*
encrypted passwords
storing *62*
error messages
by series *342*
classifications *341*
handling in custom cloning plugin scripts *209*
levels *226*

F

feedback
how to send comments about documentation *366*
FlexClone
splitting a clone *155*
full databases
restore operations for *12*

G

graphical user interface
troubleshooting *326*
GUI
troubleshooting *326*

H

help, online
accessing *17*
history
SnapManager operations *223*
history set
command *270*

I

indirect storage connection method *116*
information
how to send feedback about improving documentation *366*
in dump files *318*
interfaces
CLI *11*
UI *11*
issues, known
troubleshooting *333*

K

known issues
troubleshooting *333*

L

layouts, databases
finding recommendations for *18*
licenses
data protection
licenses required *168*
for data protection *168*
limitations
archive logs *29*
backup *29*
CLI and GUI *29*

cloning *29*
clustered Data ONTAP *33*
Data Guard Standby databases *29*
database layouts and platforms *29*
installation and configuration *29*
profile *29*
restore *29*
rolling upgrade *29*
log space
how to free for archiving *12*
logs, archive
consolidating *12*

M

mounting
backups *106*
multiple hosts
performing rollback on *44*
performing rolling upgrade on *41*
multiple operations
fail *339*

N

NFS
requirements for using databases with *26*
notification emails
configuring for multiple profiles *198*
notification set
command *275*
notification update-summary-notification
command *274*

O

online help
accessing *17*
operation dump
command *276*
operation show
command *278*
operation status
configuring alerts *195*, *197*
operation-level dump files *320*
operations
that SnapManager can perform with ASMLib *21*
operations history
for a profile *224*
maintaining *223*
viewing *224*
Operations Manager
setting up role-based access control *59*
operations, states
list of *15*
what they are *15*
what they mean *15*
operations, status
methods for monitoring *13*
Oracle
limitations *34*
setting home directory *18*
versions supported *34*
Oracle 10g ASM database
without ASMLib *25*
Oracle 9i
deprecated versions *34*
Oracle RAC ASM database
without ASMLib *24*
overview
of SnapManager for Oracle *11*

P

parameters, configuration
SnapManager *47*
partial databases
restore operations for *12*
partition devices, supported
list of *20*
password reset
command *279*
plug-in scripts
samples *210*
verifying installation *215*
plugin check
backup *280*
clone *280*
command *280*
restore *280*
policies
protection *160*
ports
general restrictions *29*
post rollback
tasks *45*
post-scripts
prerequisites *167*
post-upgrade
tasks *37*
preview
restore information *128*
product support
where to find more information *340*
profile
accessing *62*, *75*
deleting *158*
renaming *73*
profile create
command *281*
profile delete
command *285*
profile password
forgot *74*
resetting *74*
setting *62*, *75*
profile show
command *288*
profile update
command *290*
profile verify
command *294*
profile-level dump files *320*
profiles
authentication *66*

creating *67*, *179*
deleting *78*
enabling backup protection *163*
host *179*
managing *66*
overview *66*
protection policy descriptions *160*
Snapshot copy naming patterns and variables *72*
updating properties *75*
verifying *75*
profiles, databases
what they are *14*
properties
updating profiles *75*
viewing backups *104*
protected backups
about protection policies *159*
cloning *172*
enabling protection in the profile *163*
protection policy descriptions *160*
protection states *160*
protecting
archive log backups *99*
protection
using Protection Manager *168*
Protection Manager
assigning storage resources for protected backups *163*
protection-policy
command *295*
prune
archive log
pruning *96*

R

RAC databases
cloning *142*
requirements with SnapManager *19*
RBAC
about authentication *56*
overview of tasks and components *56*
setting up in Operations Manager *59*
See also role-based access control
recover
examples *84*
Recovery Manager (RMAN)
performing block-level restore operations *135*
restrictions *29*
renaming
profile *73*
renaming the storage system
issues *332*
repositories
what they are *13*
repository
accessing *62*
updating *37*
repository credentials
setting *62*
resetting
profile password *74*
resource pools
about *160*
assigning storage resources in Protection Manager *163*
restore
backup from secondary storage *171*
database backup *115*
examples *84*
from alternate location *138*
overview *170*
using prescript and post-scripts *216*
restore data
from file systems *138*
restore operations
of full databases *12*
of partial databases *12*
restore plan
preview
review
analysis *127*
restore process
types *39*
restore specifications
creating *139*
restoring
fails for RAC database *339*
local backup *188*
using SFSR *130*
using Single File SnapRestore *130*
restoring backups
about *116*
about file-based restores *116*
about volume-based restores *116*, *129*
advantages for fast, volume-based restore *119*
connection method setting in configuration file *116*
eligibility checks *120*
from alternate location *137*, *141*
from an alternate location *139*
from primary storage *131*
from secondary storage *170*
partial file snap restore (PFSR) *116*
preview restore information *128*
single file snap restore (SFSR) *116*
specifications *139*
when can you use fast restores *118*
with RMAN *135*
retention
backups to retain *165*
classes *81*
retention classes
about *81*
daily *165*
hourly *165*
monthly *165*
profile *165*
weekly *165*
retention policy
changing
retention policy *103*
example *165*
role-based access control
about *16*, *57*
about authentication *56*
enabling in SnapDrive *58*

overview of tasks and components *56*
setting up in Operations Manager *59*
rollback
limitations *43*
overview *43*
prerequisites *44*
rolling upgrade
overview *39*
prerequisites *40*

S

schedule create
command *303*
scheduling
archive log pruning *98*
secondary resource pool
configuring with Protection Manager *181*
secondary storage
about protection policies *159*
protected backup *163*
protection states *160*
restoring backups from *171*
security
about role-based access control *57*
clearing user credentials *63*
credentials *16*
deleting user credentials *64*
enabling role-based access control in SnapDrive *58*
overview of tasks and components *56*
user authentication *16*
viewing user credentials *63*
SFSR
restoring backups *130*
SIDs
used with clones *142*
Single File SnapRestore
restoring backups *130*
single host
performing rollback on *44*
performing rolling upgrade on *41*
smo_server status
command *227*
SnapDrive
configuring *53*
configuring when RBAC enabled *162*
configuring when RBAC not enabled *163*
enabling role-based access control *58*
troubleshooting *331*
SnapManager
data protection in *159*
debugging *322*
error messages *342*
general information *340*
how it maintains security *16*
limitations *29*
log levels *322*
preparing to upgrade *36*
product support information *340*
requirements for using ASM databases *19*
system requirements for upgrading *36*
troubleshooting *313*
upgrading *36*
with RAC databases *19*
SnapManager for Oracle
configuring *47*
highlights and benefits of *11*
overview of *11*
upgrading *36*
SnapManager host
performing rollback on *44*
performing rolling upgrade on *41*
SnapManager known issues
troubleshooting *333*
SnapManager operations
history *223*
SnapManager server
starting *192*
Snapshot copies
description of *11*
Snapshot copy naming patterns *72*
spfile in database, cloning *142*
splitting
clone *155*, *156*
splitting clone
on primary storage *156*
on secondary storage *156*
storage configurations
finding recommendations for *18*
storage list
command *309*
storage system name
updating *219*
storage systems
adding to the SnapManager server host *100*
storing
encrypted passwords *62*
suggestions
how to send feedback about documentation *366*
summary notification
adding existing profile *200*
adding new profile *199*
configuring *198*
system dump
command *310*
system-level dump files *321*

T

target database host name
updating *221*
task scripts
operations *205*
storing *214*
variables for backup operation *206*
variables for restore operation *208*
task specification file
creating *215*
tasks
post-upgrade *37*
troubleshooting
clones *323*
graphical user interface *326*
GUI *326*
renaming storage system issues *332*
SnapDrive *331*

SnapManager *313*
SnapManager known issues *333*
twitter
how to receive automatic notification of documentation changes *366*

U

unmount
backup *106*
update
database host name *219*
storage system name *219*
updating
target database host name *221*
upgrading
SnapManager *36*
SnapManager for Oracle *36*
user access
authorizing *62*, *75*
user authentication
role of, security purposes *16*
understanding *56*
user credentials
role of, security purposes *16*

V

variables for backup operation
in task scripts *206*
variables for restore operation
in task scripts *208*
verifying
backups *102*
profile setup *75*
veritas SFRAC
configuring *54*
version
command *311*
view
clone information *153*
list of backups *104*
volume requirements *35*
volume-based restore
about *129*
advantages and disadvantages *119*
eligibility checks *120*
from primary storage *131*
from secondary storage *170*
when can you use *118*
Vservers
See SVMs