Technical Report

SnapManager 3.0 for SAP Best Practices

SAP Competence Center, NetApp
March 2010 | TR-3823

LEVERAGING NETAPP DATA ONTAP 7G FOR BACKUP, RECOVERY, AND CLONING OF SAP SYSTEMS RUNNING ON ORACLE DATABASES

Backup, recovery, and creating system copies are all complicated tasks that are synonymous with SAP® systems management. NetApp® SnapManager® 3.0 for SAP simplifies and automates these complex operations by leveraging NetApp Snapshot™, SnapRestore®, and FlexClone® technologies to provide fast, space-efficient, disk-based backups; rapid, granular restore and recovery; and quick, space-efficient system copies of SAP systems running on Oracle® Databases. This document details the best practices for deploying and using SnapManager 3.0 for SAP.

TABLE OF CONTENTS

1 INTRODUCTION

Many leading companies rely on SAP to run their business processes. In such enterprises, business-critical SAP systems running on Oracle Databases must be operational around the clock to facilitate decision making, e-commerce, and a myriad of other internal and external processes. Rapid increases in data volume, application, and database demands make it increasingly difficult to provide availability and protection of valuable data assets. To succeed, SAP administrators need tools that will empower them to create regular backups with minimal impact, perform quick restores and recoveries, and create nondisruptive system copies for development and testing.

NetApp SnapManager for SAP (SMSAP) automates and simplifies the complex, manual, and time-consuming processes associated with the backup, recovery, and creating of system copies of SAP systems running on Oracle Databases. SnapManager leverages NetApp technologies such as Snapshot, SnapRestore, and FlexClone while integrating with SAP BR*Tools and Oracle Databases to enable complete automation of SAP data management. SnapManager for SAP allows IT organizations to scale their storage infrastructure, meet increasingly stringent SLA commitments, and improve the productivity of database and storage administrators across the enterprise.

1.1 PURPOSE AND SCOPE

The objective of this report is to provide best practices for deploying and using SnapManager 3.0 for SAP to back up, recover, and create system copies of SAP systems running on Oracle Databases on NetApp storage systems. The recommendations in this report are generic and not specific to any configuration.

1.2 INTENDED AUDIENCE

This report is intended for SAP and Oracle administrators, storage administrators, and architects implementing a backup, recovery, and cloning solution for SAP systems running on Oracle Databases on NetApp storage. Readers should ideally have a solid understanding of the architecture, administration, and backup and recovery concepts of SAP systems and Oracle Databases and should have reviewed the following:

- Data ONTAP 7.2 or 7.3 System Administration Guide
- SnapManager 3.0 for SAP Installation and Administration Guide
- SnapManager 3.0 for SAP Release Notes
- SnapDrive 4.1 for UNIX Installation and Administration Guide (if Oracle is on UNIX®)
- SnapDrive 5.0.1 or 6.0.1 for Windows Installation and Administration Guide (if Oracle is on Windows®)
- Protection Manager 3.7.1 Installation and Administration Guides
- Operations Manager 3.7.1 Installation and Administration Guides
- SAP on UNIX and Oracle with FCP and NetApp Storage
- SAP on UNIX and Oracle with NFS and NetApp Storage
- SAP on Windows and Oracle with NetApp Storage
- NetApp Best Practice Guidelines for Oracle

2 SNAPMANAGER FOR SAP SIMPLIFIES SAP DATA MANAGEMENT

2.1 NEW IN VERSION 3.0

The following is a brief overview of some of the new features in version 3.0. Refer to the SnapManager 3.0 for SAP Release Notes on the NetApp NOW™ (NetApp on the Web) site for a complete list of all the new features and enhancements in version 3.0.

POLICY-DRIVEN DATA PROTECTION

SnapManager 3.0 for SAP provides policy-driven data protection by integrating with NetApp Protection Manager 3.7.1. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault® or SnapMirror® policies created in Protection Manager by the storage or backup administrator. SnapManager 3.0 for SAP also empowers the SAP administrator to automatically restore such protected backups from the secondary storage system back to the primary storage system. Using SnapManager 3.0 for SAP, SAP administrators can also clone the protected backups on the secondary storage system for test and reporting without affecting the primary storage system. This functionality is optional and is not available when using SnapManager for SAP on Windows. SnapManager requires Protection Manager and NetApp Operations Manager for this functionality. Refer to Appendix B for instructions to configure and enable policy-driven data protection.

ROLE-BASED ACCESS CONTROL (RBAC)

SnapManager 3.0 for SAP now allows administrators to control access based on their roles by leveraging the RBAC capabilities of Operations Manager. This integration of SnapManager with Operations Manager is actually using SnapDrive® for UNIX version 4.1 and later. Older versions of SnapDrive provided limited access control and allowed only the root user to perform SnapDrive operations. SnapDrive 4.1 for UNIX now provides controlled access to even nonroot local or NIS users by using the RBAC infrastructure of Operations Manager 3.7.1. Operations Manager provides granular access to storage objects such as LUNs, qtrees, volumes, and aggregates. Because of this integration, SnapManager for SAP administrators can now control which SnapManager operations each SAP administrator can perform. This functionality is optional and is not available when using SnapManager for SAP on Windows. SnapManager requires Operations Manager for this functionality. Refer to Appendix C for instructions to configure and enable role-based access control.

FAST RESTORES USING VOLUME-LEVEL SNAPRESTORE

Prior versions of SnapManager only performed file-based restores. SnapManager 3.0 for SAP now provides a faster volume-based restore option, which is now the default. This is the fastest possible restore mechanism among all the restore mechanisms that SnapManager offers. SAP administrators can leverage this feature to restore a database in minutes irrespective of the size of the database. This functionality is optional and is not available on Windows.

BUILT-IN SCHEDULER FOR BACKUPS

SnapManager now provides a built-in scheduler for backups. The scheduler can be accessed only from the SnapManager GUI.

CLONING POLICIES AND PRETASKS AND POSTTASKS

You can create custom policies and have SnapManager enforce them. For example, a policy restricting the database system identifier (SID) as per your business rules can be created, and SnapManager will automatically verify the SID you specified in the clone request, based on the rules specified in the policy.

SnapManager can also automate executing custom scripts before and after the clone creation process. This functionality can be used to mask production data, add a temporary tablespace, and so on in the clone database.

CUSTOM SNAPSHOT COPY NAMES

You can now specify a custom naming convention for Snapshot copies created by backups under a SnapManager profile. Custom text or built-in variables such as profile name, database name, or database SID provided by SnapManager can be used to generate the naming convention.

NEW PARAMETERS SUPPORTED FOR THE BACKINT PARAMETER FILE

If you want to use different retention classes with the same SnapManager profile from the BACKINT interface, you can set the `retain` parameter in the BACKINT parameter file, also known as `util` file (for example, `init<SID>.utl`). For example, if you have two BR*Tools configuration files such as `init<SID>_weekly.sap` for weekly backups and `init<SID>_daily.sap` for daily backups, you can use two `util` files, by setting `util_par_file = init<SID>_weekly.utl` in `init<SID>_weekly.sap`, and `util_par_file = init<SID>_daily.utl` in `init<SID>_daily.sap`.

To define the retention classes, set the retain parameter in the `.utl` files in the `init<SID>_weekly.utl` file as follows:

- `retain = weekly`
- `profile_name = <SMSAP profile name>`

To define the retention classes, set the retain parameter in the `.utl` files in the `init<SID>_daily.utl` file as follows:

- `retain = daily`
- `profile_name = <SMSAP profile name>`

From BR*Tools, you can address the two parameter settings by using the option `-p init<SID>_weekly.sap` and `-p init<SID>_daily.sap`.

EXTENDED PLATFORM SUPPORT

SnapManager 3.0 for SAP now also supports backup, recovery, and creating of system copies of SAP systems running on Oracle Databases on Linux® and Windows. Refer to the SnapManager Interoperability Matrix for more details.

ORACLE RAC SUPPORT

SnapManager 3.0 for SAP now also supports SAP systems running on Oracle RAC databases.

2.2 CONCEPTS

REPOSITORY AND PROFILES

SnapManager organizes information into profiles in a repository. The profiles hold the information about the database being managed, including its credentials, backups, and clones. The repository holds data about the operations performed on the profiles.

The SnapManager repository records such information as when a backup was created, which files were backed up, whether the backup was created using the SnapManager or BR*Tools interface, and if a clone was created using the backup. A single repository can hold the information of multiple profiles, as illustrated in Figure 1. The repository can be created using the SnapManager graphical user interface (GUI) or the SnapManager command line interface (CLI) and resides in an Oracle Database.

A profile needs to be created for every database that needs to be managed by SnapManager. Once a profile is created for a database, information specific to that database will be stored in the repository. After a profile is created, database details need not be specified each time an operation is performed on that database. A profile can reference only one database. That same database can also be referenced by more than one profile, as illustrated in Figure 1. But a backup created using one profile cannot be accessed from a different profile, even if both profiles are associated with the same database.

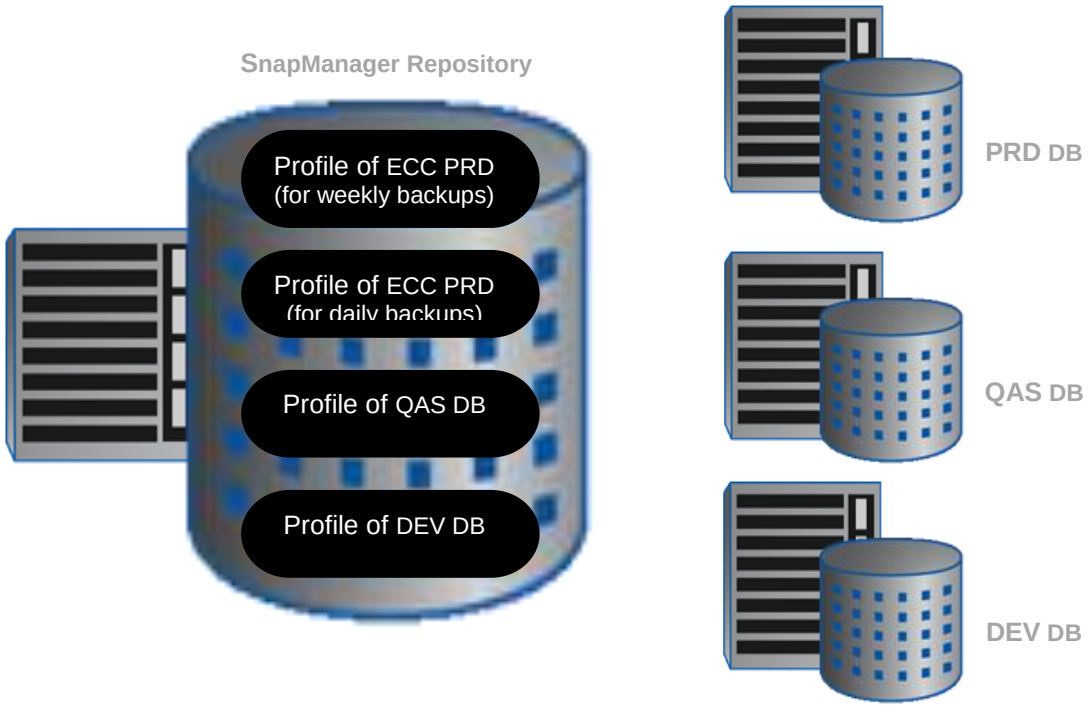


Figure 1) SnapManager repository containing profiles.

BACKUP RETENTION POLICY

You can specify the number of backups to be retained for a database while creating the SnapManager profile for that database. You also have the option to exclude a backup from the retention policy in situations such as creating ad hoc backups. The retention policy is engaged every time a new backup is created. The retention policy applies only to successful backups. Certain backups do not count toward the retention policy. For example, unsuccessful or failed backups and backups used to create a clone are not counted. Refer to the section titled "How SnapManager determines which backups to retain on local storage" in the SnapManager 3.0 for SAP Installation and Administration Guide for more details.

If SnapManager is configured to integrate with Protection Manager and data protection is enabled for a SnapManager profile, backups created under that profile on primary storage are automatically replicated to secondary storage. The database administrator using SnapManager can specify only the retention and scheduling of backups on primary storage. The storage or backup administrator using Protection Manager manages the schedule to replicate the local SnapManager backups on primary storage to secondary storage and the retention of the protected backups on secondary storage.

PROTECTION POLICIES

Protection policies are rules that govern how database backups will be protected. Protection policies are created by the storage or backup administrator using Protection Manager. A protection policy defines when to transfer copies to secondary storage and the maximum amount of data that should be transferred at scheduled times. The protection policy also defines how long to retain copies for each backup location and governs warning and error thresholds for lag time.

Since SnapManager 3.0 for SAP integrates with Protection Manager, SnapManager retrieves available protection policies from Protection Manager and empowers the SAP administrator to associate a specific protection policy with a SnapManager profile.

DATA SETS

When data protection is enabled in SnapManager for SAP by associating a protection policy with a SnapManager profile, SnapManager automatically creates a data set for the target database and registers it

with Protection Manager. A data set is a collection of user data you manage as a single unit and all the replicas of that data. The data is identified by the volume or qtree in which it is located.

RESOURCE POOLS

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

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

Data protection can optionally be enabled for a SnapManager profile. If data protection has been enabled for a profile in SnapManager and a storage resource pool has been assigned to the data set in Protection Manager, then the SnapManager profile's conformance status changes to "Conformant." If not, the profile is considered "Nonconformant."

PROTECTED BACKUPS

SnapManager 3.0 for SAP integrates with Protection Manager to automatically replicate SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. These replicated backups are called protected backups. SnapManager not only automates restoring a protected backup to the primary storage system and recovers the database but also automates cloning the protected backups on the secondary storage system for test and reporting without affecting the primary storage system.

2.3 ARCHITECTURE

Figure 2 illustrates the SnapManager for SAP architecture and the components that work together to provide a comprehensive and powerful backup, recovery, and cloning solution for SAP systems running on Oracle Databases.

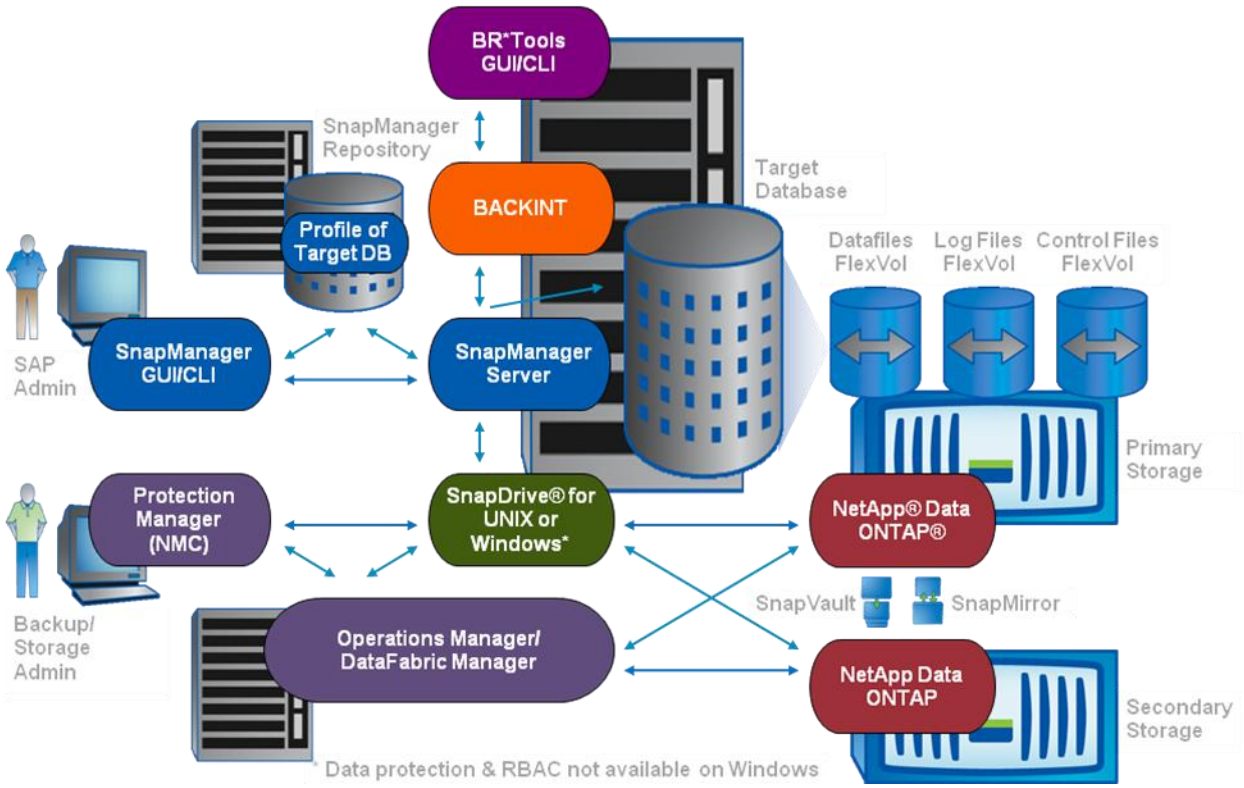


Figure 2) SnapManager for SAP architecture.

2.3.1 COMPONENTS

TARGET DATABASE

The target Oracle Database that needs to be managed by SnapManager can be configured in a variety of ways (standalone or RAC). Refer to the SnapManager Interoperability Matrix for more details about the supported Oracle Database versions, configurations, host operating systems, and protocols.

SNAPMANAGER SERVER

The SnapManager for SAP server must be installed on every host that has a database that needs to be managed by SnapManager.

SNAPDRIVE

SnapManager for SAP requires that NetApp SnapDrive for UNIX or SnapDrive for Windows be installed on the target database host before SnapManager for SAP is installed. SnapDrive simplifies storage management, reduces operational costs, and improves storage efficiency. Key SnapDrive functionality includes error-free application storage provisioning, consistent data Snapshot copies, rapid application recovery, and the ability to easily manage data with its server-centric approach. Refer to the SnapManager Interoperability Matrix to choose the appropriate SnapDrive platform and version based on the operating system running on the target database host. SnapDrive must be installed on every host that has a database that needs to be managed by SnapManager.

SNAPMANAGER REPOSITORY

SnapManager organizes information into profiles. A profile holds information about the database to be managed, including its credentials, backups, and clones. The repository holds data about the operations performed on the profiles. A single repository can hold information on multiple profiles. The repository resides in an Oracle Database.

The repository cannot reside in the database being backed up by SnapManager. Therefore, 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.

SNAPMANAGER GUI/CLI

SnapManager for SAP can be used either from the CLI on any host on which the SnapManager server is running or from the GUI on any host running any of the SnapManager supported operating systems.

OPERATIONS MANAGER

Operations Manager is the Web-based UI of DataFabric® Manager. It is used for day-to-day monitoring, alerting, and reporting on storage and storage system infrastructure. SnapManager integrates with Operations Manager to leverage the RBAC capabilities of Operations Manager. This functionality is optional and is not available on Windows.

PROTECTION MANAGER AND NETAPP MANAGEMENT CONSOLE

Protection Manager provides administrators with an easy-to-use management console for quickly configuring and controlling all SnapMirror and SnapVault operations. The application allows administrators to apply consistent data protection policies across the enterprise, automate complex data protection processes, and pool backup and replication resources for higher utilization. The interface for Protection Manager is the NetApp Management Console.

NetApp Management Console is the client platform for NetApp Management Software applications. NetApp Management Console runs on a Windows or Linux system separate from the server on which DataFabric Manager is installed. NetApp Management Console allows storage, application, and server administrators to perform daily tasks without having to switch between separate user interfaces. Applications that run in the NetApp Management Console are Protection Manager, Provisioning Manager, and Performance Advisor. Protection Manager and Operations Manager are required for leveraging the data protection features of SnapManager for SAP. These functionalities are optional and are not available on Windows.

PRIMARY STORAGE SYSTEM

The target database uses multiple volumes created on the primary NetApp storage system for laying out its datafiles, control files, archive logs, and so on. Snapshot copy backups created by SnapManager of the target database are on the primary storage system. One of the core components of a NetApp storage system is the Data ONTAP® operating system. Data ONTAP 7G is a highly optimized, scalable, and flexible operating system for unified enterprise data management. SnapManager for SAP supports the latest versions of Data ONTAP. The following licenses must be enabled on the primary storage system:

- The correct protocol (FCP, iSCSI, or NFS)
- SnapRestore
- FlexClone (mandatory if using NFS, but optional in FCP and iSCSI environments)
- SnapVault Data ONTAP primary and/or SnapMirror based on the replication policies used (required only if data protection is enabled)

SECONDARY STORAGE SYSTEM

If data protection is enabled for a SnapManager profile, then Snapshot copy backups created by SnapManager on the primary storage system are replicated to a secondary NetApp storage system using SnapVault or SnapMirror, based on the Protection Manager policy specified in the SnapManager profile. The following licenses must be enabled on the secondary storage system:

- The correct protocol (FCP, iSCSI, or NFS)
- FlexClone (mandatory if using NFS, but optional in FCP and iSCSI environments)
- SnapVault Data ONTAP secondary and/or SnapMirror based on the replication policies used (required only if data protection is enabled)

SAP INTEGRATION

SnapManager for SAP integrates with SAP BR*Tools using the BACKINT interface to automate and streamline SAP data management. SnapManager for SAP works with the SAP BRBACKUP tool to automatically identify the backup data set and create Snapshot copies of the appropriate database.

2.4 POLICY-DRIVEN DATA PROTECTION WORKFLOW

Figure 3 illustrates how the SAP administrator using SnapManager 3.0 for SAP and the storage/backup administrator using Protection Manager can enable policy-driven data protection.

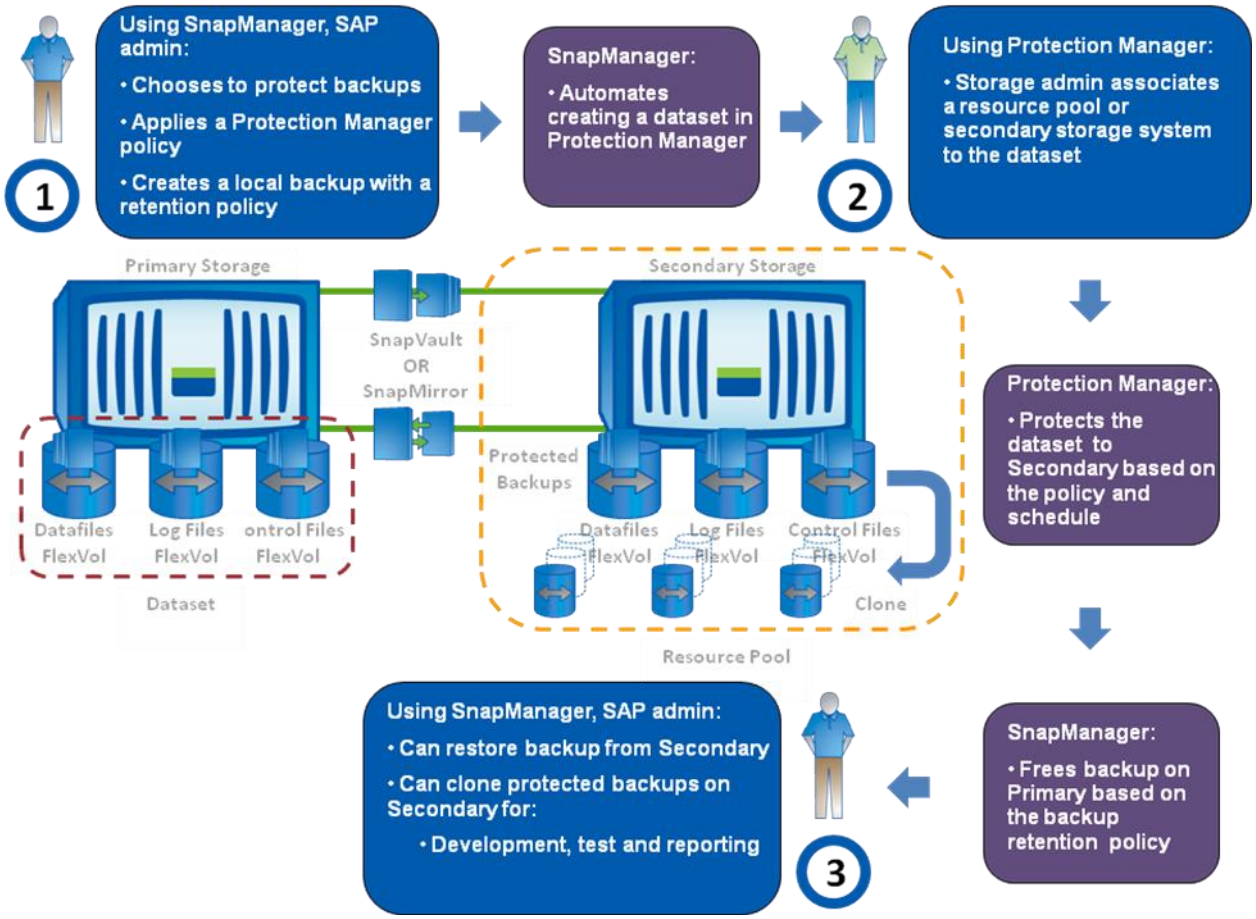


Figure 3) Policy-driven data protection workflow.

3 LANDSCAPE PLANNING

3.1 ORACLE DATABASES LAYOUT

While deploying Oracle Databases to be managed by SnapManager 3.0 for SAP on NetApp storage systems, review the following:

- SAP on UNIX and Oracle with NFS and NetApp Storage
- SAP on UNIX and Oracle with FCP and NetApp Storage
- SAP on Windows and Oracle with NetApp Storage

SnapManager only backs up the following files of an Oracle Database:

- Datafiles

  - SnapManager does not specifically back up temporary datafiles.
- Control files
- Archive logs

SnapManager for SAP currently does not back up any other files such as redo log files, parameter files, Oracle binaries, or files belonging to applications that run on the Oracle Database. The following table illustrates the differences between SnapManager for SAP, BRBACKUP, BRARCHIVE, and BRRESTORE with respect to the files they back up or restore.

Table 1) Comparison between SnapManager for SAP and BR*Tools based on files backed up or restored.

File Types	Backed Up By:			Restored By:	
	SMSAP	BRBACKUP	BRARCHIVE	SMSAP	BRRESTORE
Oracle binaries	No	Yes	No	No	Yes
Datafiles	Yes	Yes	No	Yes	Yes
Temporary datafiles	No	Yes	No	No	Yes
Control files	Yes	Yes	No	Yes	Yes
Redo logs	No	Yes	No	No	Yes
Archive logs	Yes	No	Yes	No	Yes
Parameter files	No	Yes	No	No	Yes
Application files	No	Yes	No	No	Yes

CHANGES IN VERSION 3.0 THAT AFFECT ORACLE DATABASE LAYOUT

- Previous restriction of single file system per volume group removed.
- SnapManager 3.0 for SAP now provides a faster volume-based restore option by default for datafiles. To leverage the new volume-based restore or full disk group restore, consider the following guidelines for:
  - File systems and disk groups:
    - Multiple databases cannot share the same disk group.
    - A disk group containing datafiles cannot contain other types of files. Temporary datafiles can exist on the same disk group as the regular datafiles.
    - The LUNs for the datafile disk group must be the only object in the storage volume.
  - Volume separation:
    - Datafiles for only one database must be in the volume, and the volume cannot contain other types of files. Temporary datafiles can exist on the same volume as the regular datafiles.
      - SAP uses a standard layout for Oracle Database installations. In this layout, SAP places copies of the Oracle control file in the `/oracle/<SID>/origlogA`, `/oracle/<SID>/origlogB`, and `/oracle/<SID>/sapdata1` file systems. The control file in the `sapdata1` file system conflicts with the SnapManager requirements for separating the control files and datafiles into separate volumes and must be adjusted to allow for fast restore capability.

- In the case of a new SAP install, you can adjust the location of the control files using `SAPINST` during the SAP installation process and move the control file normally placed in the `sapdata1` file system to a file system that does not reside in the same volume as the datafiles. (`SAPINST` is the tool that SAP provides for installing SAP systems.)
- In the case of a SAP system that has already been installed, you must move the control file out of that file system to allow for fast restores using SnapManager. You should do this by creating a new file system in a volume that does not contain datafiles and moving the control file to that file system using the following steps:

```
sqlplus / as sysdba
SQL> show parameter control;
SQL> shutdown immediate;
SQL> create pfile from spfile;
SQL> host;

-- Copy the control files from the current location to another existing/new volume. For example:
$ cp /oracle/<SID>/sapdata1/cntrl/cntrl<SID>.dbf
/oracle/<SID>/mirrlogB/cntrl/cntrl<SID>.dbf

-- Edit the init<SID>.ora and change the location of the control files to the new location
-- Save the changes and exit the editor
$ vi $ORACLE_HOME/dbs/init<SID>.ora
$ exit

SQL> create spfile from pfile;
SQL> startup;
SQL> show parameter control;
SQL> host;

-- Delete the control file in the original location
$ rm /oracle/<SID>/sapdata1/cntrl/cntrl<SID>.dbf
$ exit
```

- If temporary datafiles are in the same volume as regular datafiles, then while cloning, SnapManager 3.0 for SAP will create a new temporary datafile in the clone database for each corresponding datafile of the source database. If the temporary datafiles are not in the same volume as the regular datafiles in the source database, then SnapManager will not create any temporary datafiles in the clone. In this case you can specify the creation of a temporary datafile as a postclone SQL statement or script, and SnapManager will automatically create it.

  The following are some pros and cons of including the temporary datafiles in the same volume as the regular datafiles. This will help you determine the best layout for your environment:

  - Pros
    - Since SnapManager does not specifically back up temporary datafiles, they will not be part of any Snapshot copies. Including the temporary datafiles in the same volume as the datafiles will affect the Snapshot copies of that volume and make them grow in size.
    - While cloning a production database with such a layout for development or test, you can specify the creation of a smaller temporary datafile than production, because a postclone SQL statement or script and SnapManager will automatically create it. This way you can control the size of the temporary datafile that will be created in the clone database.
  - Cons
    - If you have many databases using the same storage system, then having a separate volume for temporary datafiles for each database might result in reaching the volumes per storage system limit sooner.
    - While cloning, you will have to write a postclone SQL statement or script to create one or more temporary datafiles in the clone database. SnapManager will automatically execute the SQL statement or script as a post-clone step.

BEST PRACTICES AND REQUIREMENTS FOR DATABASE LAYOUT AND CONFIGURATION

- NetApp recommends a dedicated volume just for datafiles of each database to leverage the new volume-based restore capability of SnapManager for SAP.
- NetApp recommends separating your databases into different flexible volumes.
- A database may not have files in more than one type of SAN file system or volume manager. All files making up a database must reside on the same type of file system.
- NetApp recommends that automatic Snapshot copies be turned off on volumes that are storing datafiles, control files, and archive logs of an Oracle Database.
- All LUNs within a volume should reside at the volume level or reside inside qtrees, not a combination of both.
- The database system identifier (SID) must be included in the `oratab` file. SnapManager relies on the `oratab` file to determine which Oracle home to use.
- SnapManager supports control files on a file system and does not support control files on raw devices.
- You can have multiple disk groups for a database; however, the following rules apply to all disk groups for a given database:
  - Disk groups for the database can be managed by only one volume manager.
  - Raw devices backed by Volume Manager volumes are not supported.
  - A Linux environment without logical volume management (LVM) requires a partition.

BEST PRACTICES AND REQUIREMENTS FOR USING RAC DATABASES WITH SNAPMANAGER

- SnapManager 3.0 for SAP now also supports OS-authenticated database connections for RAC databases. The SnapManager server must be installed and running on each node in the RAC cluster for a RAC database that is using the OS-authenticated connection mode.
- If using the database-authentication connection mode:
  - The listener on each node that services an instance of the RAC database must be configured to use the same port number. Also, the listener that services the primary database instance must be started prior to initiating a backup.
  - The password of the database user that SnapManager uses (typically `sys`) must be the same for all the Oracle instances in a RAC environment.

BEST PRACTICES AND REQUIREMENTS FOR USING DATABASES ON FCP OR ISCSI WITH SNAPMANAGER

- A database may not have files in more than one type of SAN file system or volume manager. All files making up a database must reside on the same type of file system.

BEST PRACTICES AND REQUIREMENTS FOR USING DATABASES ON NFS WITH SNAPMANAGER

- All the volumes that contain Oracle datafiles, control files, redo and archive logs, and the Oracle home must be exported with the `anon=0` or `root=<hostname>` option, which is more secure. SnapManager runs as root and must be able to access the file systems containing Oracle data.
- All the volumes that contain Oracle datafiles, control files, redo and archive logs, and the Oracle home must also have attribute caching disabled and exported with the `noac` (for Solaris™, AIX, HP-UX) or `actimeo=0` (for Linux) option.
- For any storage system volume that uses the NFS protocol and that contains SAP data backed up using BR*Tools, you must disable client access to the Snapshot copies of that volume. If client access is enabled, BR*Tools can attempt to create backups of the hidden .snapshot directories that contain previous backups. Refer to the section titled “Disabling client access to Snapshot copies” in the SnapManager 3.0 for SAP Installation and Administration Guide for more details.
- NetApp recommends not using symbolic links to the location of the database datafiles if linked from local storage to NFS. SnapManager does support symbolic links but only at the mountpoint level.

3.2 SNAPMANAGER REPOSITORY LAYOUT

SnapManager for SAP uses a repository to store metadata about the databases that it manages. This repository must reside in an Oracle Database. Part of the installation process of SnapManager for SAP

involves creating the repository in a schema of a previously installed Oracle Database. Since the repository holds metadata information about all the databases that SnapManager manages, it is critical to follow the best practices mentioned in this section while planning and creating the repository.

WHERE TO INSTALL THE REPOSITORY

Since the SnapManager repository must reside in an Oracle Database, NetApp recommends a dedicated database. NetApp does not recommend reusing an existing SAP Oracle Database for the following reasons:

- SAP Oracle Database licenses do not allow for the installation of standalone Oracle instances. Check with your Oracle representative for the correct license arrangements for the SnapManager repository database.
- SAP discourages storing non SAP application data in a SAP database.
- The Oracle versions and patch levels supported by SnapManager for SAP might be different from those required by SAP applications.
- SAP systems maintenance such as upgrades, kernel patches, or support packages will affect the availability of the SnapManager repository, which will in turn affect SnapManager operations (backups, cloning, and so on) on databases managed by SnapManager for SAP.
- NetApp recommends deploying the databases used for the SnapManager repositories on NetApp storage systems so that cross-repository backups can be performed, as discussed later in this document.

SIZING THE REPOSITORY

The size of the repository will determine how many backup records it can hold. The size of a single backup record in the repository is approximately 100KB. At this size, a repository of 100MB can hold approximately 1,000 backup records.

TABLESPACE AND SCHEMA CONSIDERATIONS

NetApp recommends creating a separate schema with its own tablespace for the SnapManager repository, thus making it easy to back up and restore. SnapManager requires a minimum 4K block size for the tablespace into which it is installed. Check the block size in SQLPlus using the SQL command:

```
select TABLESPACE_NAME, BLOCK_SIZE from dba_tablespaces;
```

Grant only the "`connect`" and "`resource`" roles to the database user who will own the SnapManager repository.

```
grant connect, resource to <smsap_repo_owner>;
```

Refer to Appendix A for a list of steps to quickly install and configure SnapManager for SAP.

HOW MANY REPOSITORIES SHOULD BE INSTALLED?

Being able to back up and restore the SnapManager repository is critical. SnapManager for SAP cannot back up and restore its own repository. Therefore it is a best practice to create at least two repositories so that cross backups can be performed using SnapManager. For example, Repository X has a profile of Repository Y, and vice versa, as shown in Figure 4. You can create more than two repositories, but managing numerous repositories can be complex. The total number of repositories depends on how you choose to organize the target databases into the repositories.

HOW TO DISTRIBUTE SYSTEMS AMONG THE REPOSITORIES

There are several ways to organize the target system profiles among the SnapManager repositories. Two of them are:

- Organizing by application type
- Organizing by usage

If you have multiple Oracle Databases running different applications, such as ECC, CRM, BI, and Portal, then you can create a SnapManager repository for every application type that you have. Each SnapManager repository would have profiles for the databases of a particular application type. All production, development, and testing systems of that application type would be managed by the same SnapManager repository. This

would help group like systems and ease cloning. However, if you have several application types, then you might have to manage several SnapManager repositories, and if you choose to implement another application type you will need to create another SnapManager repository. Since these SnapManager repositories will be managing production instances, each of these repositories will need to be on a server with high availability, which could be expensive. If security is a concern, then managing production systems along with development and test systems of the same type in the same SnapManager repository could be an issue.

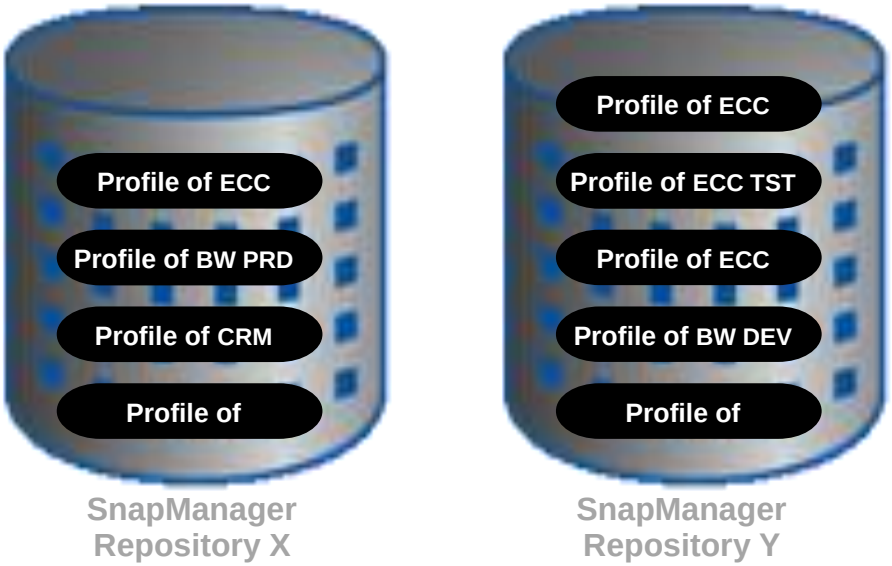


Figure 4) SnapManager for SAP sample repository layout.

Another option is to distribute the databases among the SnapManager repositories based on their usage (for example, development, quality assurance, production, training, and so on). This will limit the number of repositories to the different types of systems that you have. As a result, all production systems will be managed by a single SnapManager repository, and only production SAP administrators can be given access to this repository. Also, if you choose to deploy another SAP application, then you just need to register it in the corresponding SnapManager repository instead of creating a new repository. High availability can be provided only for the SnapManager repository that holds profiles of all the production systems.

A variation of the previous distribution would be to distribute all production systems in a single SnapManager repository and all of the other nonproduction (for example, development, testing, training, and so on) databases in another repository, as depicted in Figure 4. This has the same advantages of the previous distribution along with the added benefit that the number of SnapManager repositories will be reduced to only two.

BEST PRACTICES AND REQUIREMENTS FOR SNAPMANAGER REPOSITORY LAYOUT

- SnapManager requires the Oracle Database used for the SnapManager repository to be 10*g*R2 or higher. Refer to the SnapManager Interoperability Matrix for more details about the supported Oracle Database versions, configurations, host operating systems, and protocols.
- SnapManager cannot back up and restore its own repository. NetApp recommends creating at least two SnapManager repositories so that cross backups (that is, Repository X has a profile of Repository Y, and vice versa) can be performed using SnapManager, as depicted in Figure 4.
- NetApp recommends using a dedicated database for the SnapManager for SAP repository that is not shared with other applications.
- NetApp recommends deploying the database used for the SnapManager repository on a NetApp storage system for quick backup and restore capabilities.
- NetApp recommends creating a separate schema with its own tablespace for the SnapManager repository, thus making it easy to back up and restore.

- SnapManager requires a minimum block size of 4K for the tablespace into which it is installed. Check the block size in SQLPlus using the SQL command:

```
select TABLESPACE_NAME, BLOCK_SIZE from dba_tablespaces;
```

- Grant only the "`connect`" and "`resource`" roles to the database user who will own the SnapManager repository using the SQL command:

```
grant connect, resource to <smsap_repo_owner>;
```

- The repository cannot reside in the database being backed up by SnapManager. Therefore, 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.

4 INSTALLATION AND CONFIGURATION

4.1 LICENSES

Table 2) Required and optional licenses.

License	Type	Required or Optional	
SnapManager for SAP	Host or storage system	On host	Required if using host-based license
		On primary	Required if using storage-based license
		On secondary	Required if using storage-based license and data protection feature
SnapDrive for UNIX and or SnapDrive for Windows	Host or storage system	On host	Required if using host-based license
		On primary	Required if using storage-based license
		On secondary	Required if using storage-based license and data protection feature
Protection Manager	Host	• Required along with Operations Manager for data protection • Not required for RBAC • Not required if data protection is not desired	
Operations Manager	Host	• Required for RBAC • Required (along with Protection Manager) for data protection • Not required if RBAC and data protection features are not desired	
Data ONTAP	Storage system	On primary	Required
		On secondary	Required if using data protection feature
SnapRestore	Storage system	On primary	Required
		On Secondary	Not required
FlexClone	Storage system	On primary	• Required for NFS • Required for SAN if SnapDrive is configured to use FlexClone in SAN environments; optional otherwise

		On secondary	• Required for NFS and if using data protection feature
			• Required for SAN only if all of the following conditions are met:
			  ○ Using data protection feature
			  ○ SnapDrive is configured to use FlexClone in SAN environments
			  ○ Using a protection policy that leverages volume SnapMirror (VSM)
			Note: If using a protection policy that leverages SnapVault or qtree SnapMirror (QSM), SnapDrive will always create LUN clones even if SnapDrive is configured to use FlexClone in SAN environments. In such cases, FlexClone need not be licensed on the secondary storage system.

NFS, iSCSI, or FCP (depending on the protocol used)	Storage system	On primary	Required
		On secondary	Required if using data protection feature
SnapVault	Storage system	On primary	Required only if data protection is enabled and a SnapVault policy in Protection Manager is used
		On secondary	Required only if data protection is enabled and a SnapVault policy in Protection Manager is used
SnapMirror	Storage system	On primary	Required only if data protection is enabled and a SnapMirror policy in Protection Manager is used
		On secondary	Required only if data protection is enabled and a SnapMirror policy in Protection Manager is used

4.2 INSTALLING

SnapManager for SAP is easy to install and configure. Refer to Appendix A for a list of steps to quickly install and configure SnapManager for SAP. For more detailed instructions, follow the steps listed in the chapters titled “SnapManager for SAP deployment considerations,” “Installing or upgrading SnapManager for SAP,” and “SnapManager for SAP workflow quick start” in the SnapManager 3.0 for SAP Installation and Administration Guide.

CHANGES IN VERSION 3.0 THAT AFFECT INSTALLATION

- Prior versions of SnapManager required a server password to be specified during installation of the SnapManager server. This host SnapManager server password has been eliminated in version 3.0.
- SnapManager 3.0 for SAP now also supports OS-authenticated database connections for RAC databases. The SnapManager server must be installed and running on each node in the RAC cluster for a RAC database that is using the OS-authenticated connection mode.

4.3 CHOOSING BETWEEN THE GUI AND CLI

All critical SnapManager operations can be performed using either the GUI or the CLI with the exception of the following mentioned new features in version 3.0, which are available only from the GUI. The command-line interface provides the additional benefit of scripting the SnapManager commands. The SnapManager CLI commands can also be executed by any scheduling software such as CRON.

CHANGES IN VERSION 3.0 THAT AFFECT THE GUI AND CLI

- The SnapManager GUI now provides a built-in scheduler for scheduling backups. The scheduler is available only from the GUI and only for backups. To schedule SnapManager operations using the CLI, any scheduling software such as CRON in UNIX systems can be used.
- Certain fields' common to multiple SnapManager profiles can now be updated at once using the SnapManager GUI. This feature can be leveraged to update frequently changed information stored in SnapManager profiles. For example, if production database passwords are the same and changed regularly, all production profiles can be updated at once. This is also a feature available only in the GUI.
- Most operations requested using the SnapManager GUI can now be run in the background by clicking the new Background button available in most of the wizards for various SnapManager operations such as backups. This allows SAP administrators to execute multiple SnapManager operations for different databases in parallel.
- The SnapManager GUI now has a new Monitor tab to monitor all SnapManager operations and provides the ability to sort rows, rearrange columns, and filter results to create custom reports. If using this tab to monitor operations, SnapManager will automatically refresh every few seconds and even display operations started by other SAP administrators using the SnapManager GUI. This is also a feature available only in the GUI.
- The SnapManager GUI can now be launched on any operating system supported by SnapManager for SAP. Prior SnapManager versions supported launching the GUI only on Windows. Refer to the SnapManager Interoperability Matrix for more details about the supported host operating systems.

4.4 LAUNCHING THE GUI

Starting from version 1.1, the SnapManager GUI is launched from a Web browser on any host running Windows XP and Windows 2003 server operating systems using:

```
https://smsap-server.domain.com:port
```

In this URL, replace:

- `smsap-server` with the name of the host where the SnapManager server was started
- `domain.com` with the domain of SnapManager server host
- `port` with the port number that the SnapManager server is using on the database host; the default port is 27314.

For example, if the SnapManager server is started on the host prd_db1.rtp.netapp.com on port 27314, then you can launch the GUI using the URL https://prd_db1.rtp.netapp.com:27314. Doing so will display the "Launch SnapManager for SAP" link. Click it to launch the GUI.

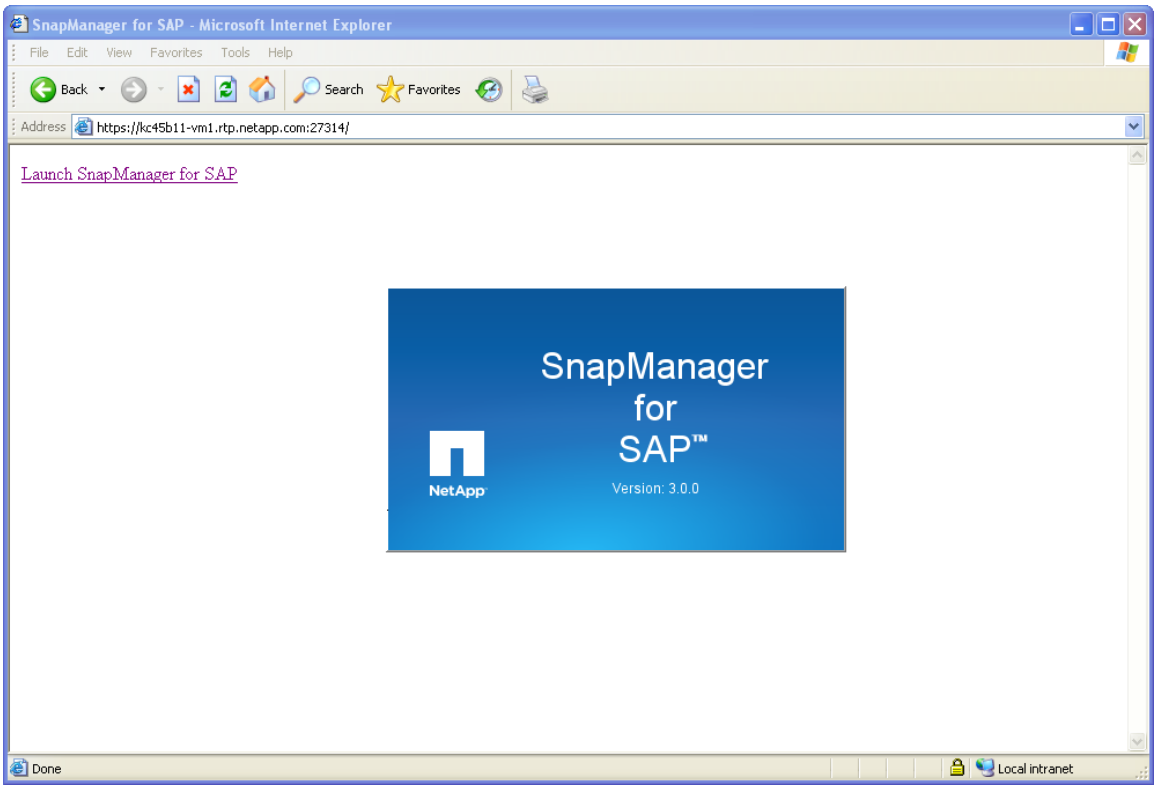


Figure 5) Launching SnapManager for SAP.

While the GUI is being launched you might encounter a few "Hostname Mismatch" warning dialogue windows, which will prompt if you want to proceed. Click "Yes" in all these windows.

If port 27314 is in use or you would like to use another port, then you can reconfigure SnapManager for SAP to use a different port by following these steps:

1. Stop the SnapManager server from the SnapManager CLI by entering `smsap_server stop`.
2. Modify the values of the following parameters in the `<SnapManager_install_directory>/properties/smsap.config` file:

   ```
   SMOServer.port=27314
   SMOServer.rmiRegistry.port=27315
   remote.registry.ocijdbc.port=27315
   ```

   The `remote.registry.ocijdbc.port` must be the same as the `SMOServer.rmiRegistry.port`.
3. Start the SnapManager server from the SnapManager CLI by entering `smsap_server start`.

Once the GUI is launched, you can also customize settings that apply to your use of the SnapManager GUI, as shown in the following screenshot, by clicking the "Admin" menu and then selecting "User Preferences…." These settings are in effect even when you log in using a different computer (if your home directory is shared between both systems).

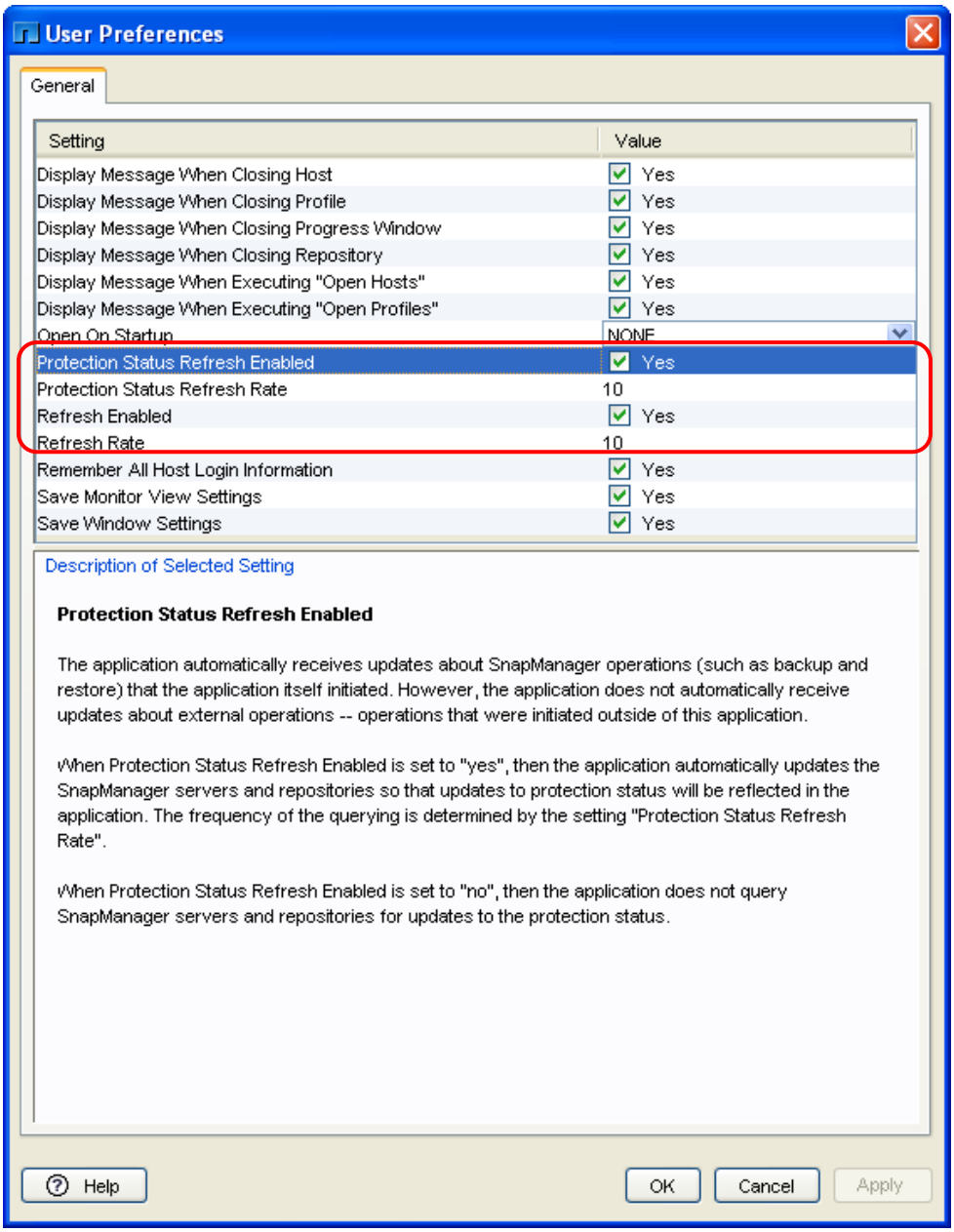


Figure 6) SnapManager for SAP GUI user preferences.

The following table describes some of the important settings listed in the “User Preferences” screen. If data protection is enabled, NetApp recommends enabling “Refresh Enabled” and “Protection Status Refresh Enabled” if not already enabled.

Table 3) User preferences settings.

Setting	Description
Refresh Enabled	Enables SnapManager to poll the repository for updates. When “Refresh Enabled” is enabled, the application queries the SnapManager servers and repositories for updates to operations. The frequency of the querying is determined by the “Refresh Rate” value. “Refresh Enabled” is enabled by default.

Refresh Rate	If “Refresh Enabled” is enabled, the “Refresh Rate” determines how often (in seconds) the application queries the SnapManager servers and repositories for updates to operations. The initial default value is 60 seconds. You can set the value to anything from 10 seconds through 1,000 seconds, inclusive. This field applies only if “Refresh Enabled” option is enabled.
Protection Status Refresh Enabled	While SnapManager automatically updates information about its operations (such as backup, restore, and clone) in the Monitor tab and the Backups/Clones tab, it does not automatically display updates about external operations (such as protection of a backup on secondary storage). When “Protection Status Refresh Enabled” is set to “Yes,” then the application automatically updates the SnapManager servers and repositories so that updates to protection status will be reflected in the application. The frequency of the querying is determined by the setting “Protection Status Refresh Rate.” When “Protection Status Refresh Enabled” is set to “No,” then the application does not query SnapManager servers and repositories for updates to the protection status.
Protection Status Refresh Rate	If “Protection Status Refresh Enabled” is set to “Yes,” then Protection Status Refresh Rate determines how often (in seconds) the application queries the SnapManager servers and repositories for updates to the protection status. The initial default value is 60 seconds. You can set the value to anything from 10 seconds through 1,000 seconds, inclusive.

4.5 CREATING THE REPOSITORY

The repository holds data about the operations performed by SnapManager on the databases it manages. Follow the best practice recommendations mentioned in section 3.2, “SnapManager Repository Layout,” and install and configure an Oracle Database for creating the SnapManager repository.

The easiest way to create the SnapManager repository is from the GUI. From the GUI, select “Operations -> Repository -> Create New Repository…” This will launch the repository wizard and walk you through the repository creation process. Because there is no one-to-one requirement for the SnapManager GUI and the repositories, the GUI can administer multiple SnapManager repositories.

Refer to Appendix A for a list of steps to quickly create the repository.

4.6 CONFIGURING AUTHENTICATION

Credentials are required to authenticate access to secure resources and SnapManager services, such as the database hosts, SnapManager repositories, and SnapManager database profiles. Refer to Appendix A for a list of steps to quickly configure authentication. For more detailed instructions, follow the steps listed in the chapters or sections titled “SnapManager for SAP workflow quick start” and “Managing access and credentials” in the SnapManager 3.0 for SAP Installation and Administration Guide.

4.6.1 Role-Based Access Control

SnapManager 3.0 for SAP now allows SAP administrators to control access based on their roles by leveraging the RBAC capabilities of Operations Manager. This integration of SnapManager with Operations Manager is actually using SnapDrive 4.1 for UNIX. Older versions of SnapDrive provided limited access control and allowed only the root user to perform SnapDrive operations. SnapDrive 4.1 for UNIX now provides controlled access to even nonroot local or NIS users by using the RBAC infrastructure of Operations Manager 3.7.1. Operations Manager provides granular access to storage objects such as LUNs, qtrees, volumes, and aggregates. Due to this integration SnapManager for SAP administrators can now control what SnapManager operations each SAP administrator can perform. This functionality is optional and is not available on Windows. SnapManager requires Operations Manager only if role-based access control is desired. SnapManager does not require Protection Manager for providing role-based access control.

Refer to Appendix C for instructions to configure and enable role-based access control.

CHANGES IN VERSION 3.0 THAT AFFECT AUTHENTICATION

- Prior versions of SnapManager required a server password to be specified during installation of the SnapManager server. This host SnapManager server password has been eliminated in version 3.0 and replaced with individual user operating system (OS) authentication. The SnapManager server now authenticates users with their OS user names and passwords, if those users are not running the client from the same server as the host. If the users are running the client from the same host as the server, the SnapManager server does not have to authenticate them because they are already logged into the server host. If users want to avoid being prompted for their OS user passwords or if using custom scripts that invoke the SnapManager CLI, they can save their data to their SnapManager user credentials cache with the `smsap credential set -host` command. This command saves the encrypted password. The `smsap credential set -host` command remembers the user's credentials when the `host.credentials.persist` property in the `<default installation location>/properties/smsap.config` file is set to `true`.
- If using the CLI, the `smsap credential set` command can now be used without the `-password` option as it will prompt for the password. Previous versions mandated using the `-password` option, which is not secure as the password might be viewable elsewhere in the environment (shell history, process listing, and so on). Although the `-password` option is still available in version 3.0 for backward compatibility, NetApp recommends using the `smsap credential set` command without the `-password` option.

4.7 CREATING PROFILES

To perform any operation on a database using SnapManager for SAP, a profile has to be created in SnapManager for that database. A profile can reference only one database. That same database can be referenced by more than one profile. Backups created using one profile cannot be accessed from a different profile, even if both profiles reference the same database.

You can create profiles using the SnapManager GUI by selecting either “Operations -> Repository -> Create Profile…” or by right-clicking the repository and selecting “Create Profile…” from the drop-down menu. This launches the profile wizard that will guide you through the steps to create a profile.

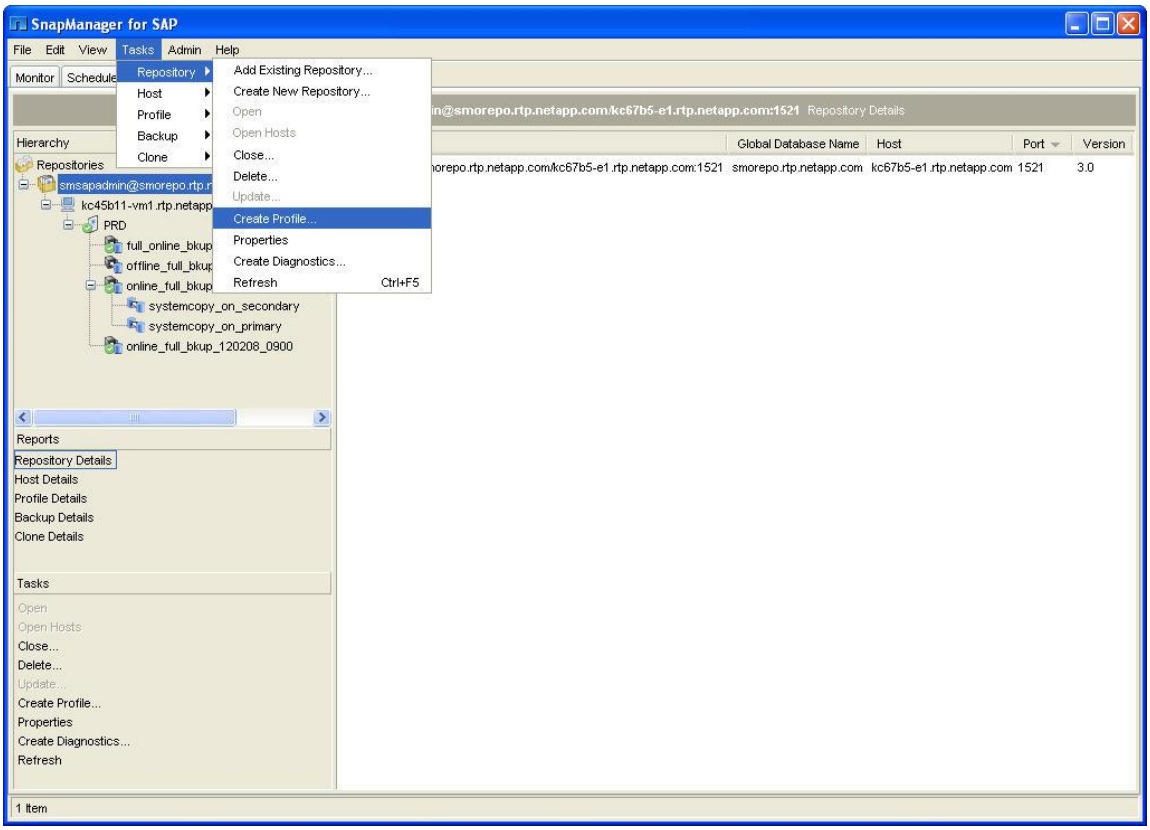


Figure 7) Creating a profile.

While creating a profile, if you choose to use "Database Authentication," then you must provide a database user ID and password with the `sysdba` role. You can choose to create a new database user for this purpose or use an existing Oracle standard user such as "`sys`."

Refer to Appendix A for a list of steps to quickly create a profile in SnapManager for SAP.

SPECIFYING A PROFILE FOR BR*TOOLS COMMANDS

When you run a BR*Tools command that uses the SnapManager provided BACKINT, SnapManager uses a profile from the repository. The repository is determined by the SnapManager credentials of the user running the BR*Tools command. By default, SnapManager uses the profile with the same name as the SAP SID.

As long as the SID is unique for all hosts whose SnapManager profiles are in a given repository, the default profile name is sufficient. Create the SnapManager profile and name it using the database SID value.

However, if the same SID is used on different hosts, or if you want to be able to specify more than one SnapManager profile for use with BR*Tools on a given SAP instance, then you need to define the profile name for BR*Tools commands. For more details on how to specify the profile for BR*Tools commands, follow the steps listed in the SnapManager 3.0 for SAP Installation and Administration Guide under the section titled "Specifying the profile for BR*Tools backups."

4.7.1 CUSTOM SNAPSHOT NAMES

You can now specify a custom naming convention for Snapshot copies created by backups under a SnapManager profile. Custom text or built-in variables such as profile name, database name, or database SID provided by SnapManager can be used to generate the naming convention, as shown in the following screenshot. SnapManager requires the "`{smid}`" variable in the snapname pattern to make every Snapshot copy unique.

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 pattern.

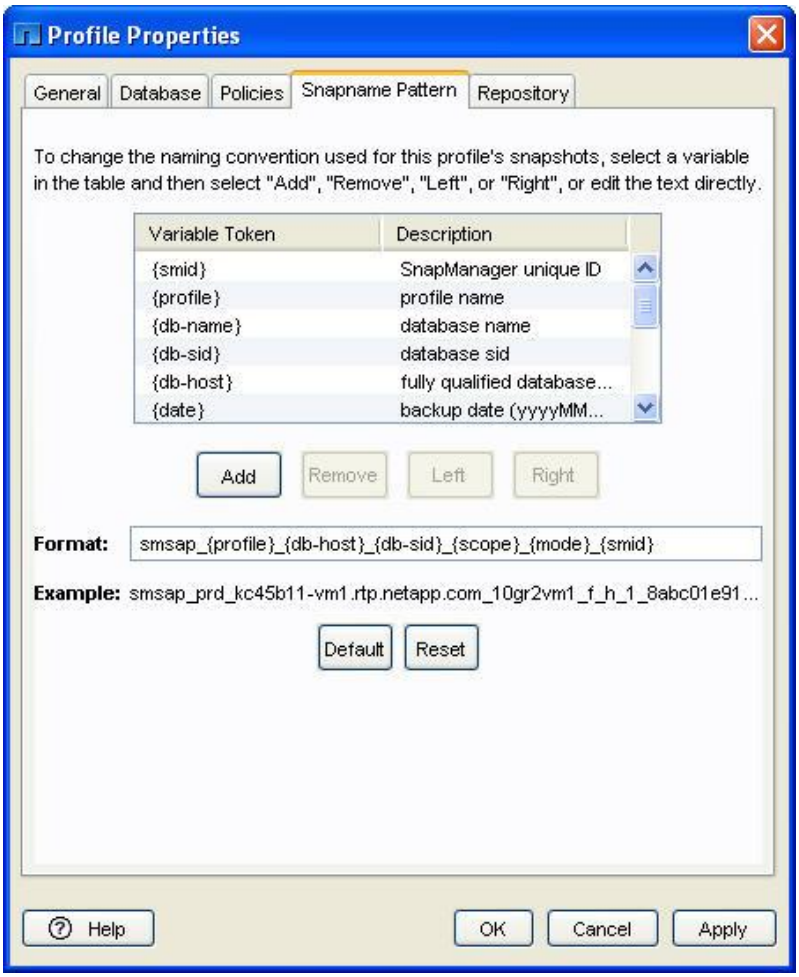


Figure 8) Custom Snapshot copy name.

The following example shows two Snapshot copy names created for the datafiles volume and the control files 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 created 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 created. The "1" Snapshot copy includes the datafiles; the "2" Snapshot copy includes the control files. Because the control file Snapshot copy must be created after the datafile Snapshot copy, two Snapshot copies are required.

Table 4) Sample Snapshot copy names.

Snapname Pattern	`smsap_{profile}_{db-host}_{db-sid}_{scope}_{mode}_{smid}`
Example Snapshot Copy Names	`smsap_prd_kc45b11-vm1.rtp.netapp.com_10gr2vm1_f_h_2_8abce26c17d71adc0117d71ae23b0001_0` `smsap_prd_kc45b11-vm1.rtp.netapp.com_10gr2vm1_f_h_1_8abce26c17d71adc0117d71ae23b0001_0`

4.7.2 BACKUP RETENTION

For every SnapManager profile of a database you can specify a backup retention policy that determines how many successful backups on local storage created under that profile should be retained. The retention policy is engaged every time you create a new backup. While creating a backup using either the SnapManager GUI or CLI, you can specify a retention class of hourly, daily, weekly, monthly, or unlimited for that backup. The number of backups of each retention class that you would like to retain is what is specified in the SnapManager profile. The retention is specified as a:

- Retention count: This determines the minimum number of backups of a particular retention class that should be retained; for example, 10 daily backups.
- Retention duration: This determines the minimum length of time a backup of a particular retention class should be retained; for example, 10 days of daily backups.

For each retention class you can specify only a count or only duration or a combination of count and duration. The following table lists the pros and cons of each and will help guide you to choose the right combination to match your backup retention requirements.

Table 5) Retention classes and their pros and cons.

Retention	Example	Pros	Cons
Count only	Retention class: daily Retention count: 10 Retention duration: 0	SnapManager will enable at least the specified number of backups of that retention class to be retained. For example, if 10 daily backups exist and a new daily backup is created, the oldest backup will be deleted.	Ad-hoc backups of the same retention class also contribute toward the count. For example, a SAP administrator wants to retain daily backups for the last 10 days and so specifies the retention count to be 10. The SAP administrator then creates two ad hoc backups with the retention class “daily” on a particular day, and then those backups will also contribute to the daily retention count, which means the SAP administrator will end up with 10 backups with the retention class “daily” but only daily backups of the last eight days (since two ad hoc daily backups were created on the same day).
Duration only	Retention class: daily Retention count: 0 Retention duration: 10	SnapManager will enable backups to be kept for at least 10 days. For example, if the backup was created on January 1, then it will be deleted on January 11.	If for some reason backups do not get created for 10 days, then all backups will be deleted, leaving no backups from which to restore.
Count and duration	Retention class: daily Retention count: 10 Retention duration: 10	SnapManager will enable at least 10 backups to exist, and all backups less than 10 days old will exist. This keeps ad hoc backups from disrupting the number of days backups are kept and enables 10 backups to exist from which to restore.	None.

A backup expires when its age exceeds the retention duration set for its retention class and the number of backups exceeds the retention count. After a backup expires, SnapManager either frees or deletes the

expired backup. Refer to the section titled “How SnapManager determines which backups to retain on local storage” in the SnapManager 3.0 for SAP Installation and Administration Guide for more details.

If you want to use different retention classes with the same SnapManager profile from the BACKINT interface, you can set the `retain` parameter in the BACKINT parameter file, also known as `util` file (for example, `init<SID>.utl`). For example, if you have two BR*Tools configuration files such as `init<SID>_weekly.sap` for weekly backups and `init<SID>_daily.sap` for daily backups, you can use two `util` files, by setting `util_par_file = init<SID>_weekly.utl` in `init<SID>_weekly.sap`, and `util_par_file = init<SID>_daily.utl` in `init<SID>_daily.sap`.

To define the retention classes, set the retain parameter in the .utl files in the init<SID>_weekly.utl file as follows:

- `retain = weekly`
- `profile_name = <SMSAP profile name>`

To define the retention classes, set the retain parameter in the `.utl` files in the `init<SID>_daily.utl` file as follows:

- `retain = daily`
- `profile_name = <SMSAP profile name>`

From BR*Tools, you can address the two parameter settings by using the option `-p init<SID>_weekly.sap` and `-p init<SID>_daily.sap`.

4.7.3 ENABLING POLICY-DRIVEN DATA PROTECTION

SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. This functionality is optional and is not available on Windows. Protection Manager and Operations Manager licenses are required to use this functionality.

Data protection in SnapManager 3.0 for SAP is enabled at the SnapManager profile level, as shown in the following screenshot. Refer to Appendix B for instructions to configure and enable policy-driven data protection.

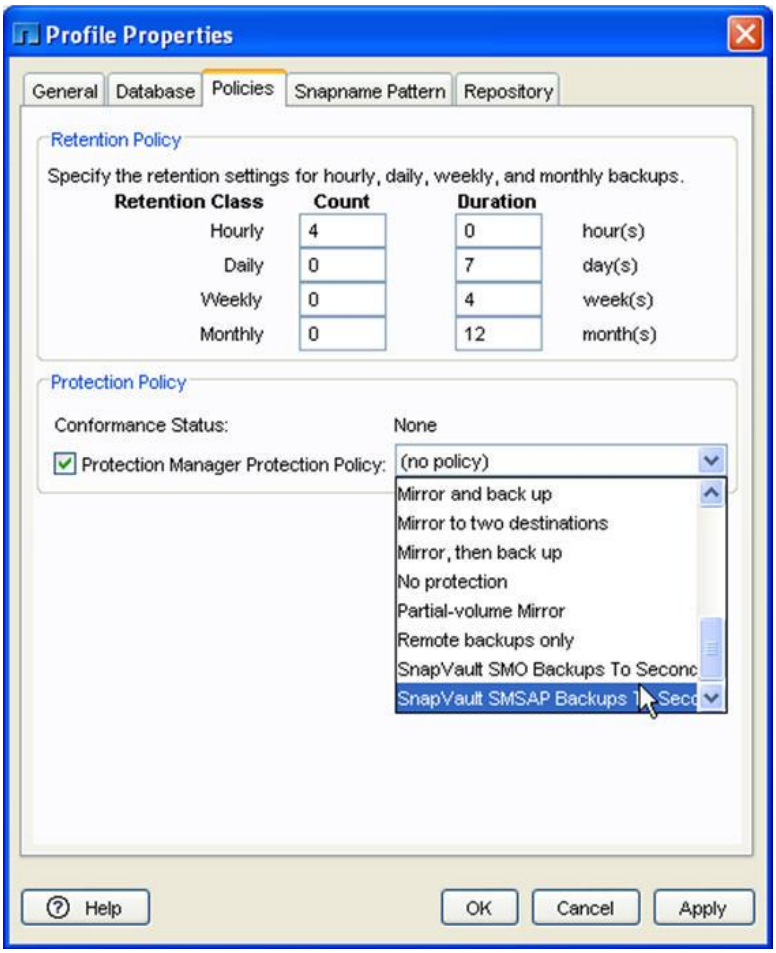


Figure 9) Enabling data protection.

CHANGES IN VERSION 3.0 THAT AFFECT PROFILES

- SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. Data protection is enabled at the SnapManager profile. This functionality is optional and is not available on Windows.
- You can now specify a custom naming convention for Snapshot copies created by backups under a SnapManager profile. Custom text or built-in variables such as profile name, database name, and database system identifier provided by SnapManager can be used to generate the naming convention.
- Certain fields' common to multiple SnapManager profiles can now be updated at once using the SnapManager GUI. This feature can be leveraged to update frequently changed information stored in SnapManager profiles. For example, if production database passwords are the same and changed regularly, all production profiles can be updated at once. This is a feature available only in the GUI.

BEST PRACTICES AND REQUIREMENTS FOR SNAPMANAGER INSTALLATION AND CONFIGURATION

- SnapDrive for UNIX and any associated requirements must be installed before SnapManager for SAP is installed.
- If upgrading to version 3.0, back up all SnapManager repositories before updating them. A failure during the update process could leave the repository in an inconsistent state. Neither SnapDrive nor SnapManager supports reverting to a previous version.

- If upgrading, NetApp recommends upgrading all the SnapManager servers to version 3.0 as the SnapManager GUI and CLI are version specific. Version 3.0 GUI and CLI can only talk to version 3.0 SnapManager servers.
- The target database SID should be included in the `oratab` file. SnapManager relies on the `oratab` file to determine which Oracle home to use. The profile verification operation will fail if SnapManager cannot find the SID in the `oratab` file.
- NetApp recommends allocating sufficient space for the `/opt/NetApp/smsap/tmp` directory. SnapManager uses this directory to hold temporary safe copies while it performs an operation.
- NetApp recommends replicating SnapManager backups of all critical databases to a secondary storage system by leveraging the policy-driven data protection feature in version 3.0. Refer to Appendix B for instructions to configure and enable policy-driven data protection.
- If accessing two or more SnapManager for SAP servers from a single client (GUI or CLI), the default port numbers for all the servers must be the same.
- If using the CLI, the `smsap credential set` command can now be used without the `–password` option as it will prompt for the password. Previous versions mandated using the `–password` option, which is not secure as the password might be viewable elsewhere in the environment (shell history, process listing, and so on). Although the `–password` option is still available in version 3.0 for backward compatibility, NetApp recommends using the `smsap credential set` command without the `–password` option.

5 BACKUP, RESTORE, RECOVERY, AND CLONING

SnapManager for SAP supports the following two different methods for backing up, restore, recovery, and creating of system copies of SAP systems running on Oracle Databases:

- SnapManager for SAP GUI or CLI
- SAP BR*Tools integrated with SnapManager for SAP using the BACKINT interface. The following SAP tools are available for backing up and managing Oracle Database backups:
  - BRBACKUP
    - Backs up datafiles, control files, and online redo logs
  - BRRESTORE
    - Restores datafiles, control files, archive logs, and online redo logs
  - BRRECOVER
    - Performs automatic recovery

Both of these methods can be used either separately or together but should not be mixed. For example, a backup created using BRBACKUP should not be restored using SnapManager for SAP. NetApp recommends using the same tool used for backing up to also be used to perform restore and recovery. This means if you use SnapManager for SAP to create a particular backup, then use SnapManager for SAP to restore and recover from that backup. Similarly, if you have configured SAP BR*Tools to integrate with SnapManager for SAP using the BACKINT interface and you then use BRBACKUP to create a particular backup, then use SAP BR*Tools such as BRRESTORE and BRRECOVER to restore and recover from that backup.

5.1 BACKING UP

SnapManager for SAP leverages NetApp Snapshot technology to create fast and space-efficient backups of Oracle Databases. These backups are point-in-time virtual copies of the database and are stored on the same physical medium of the database. SAP BR*Tools when integrated with SnapManager for SAP using the BACKINT interface also leverages NetApp Snapshot technology to create fast and space-efficient backups.

The following table illustrates the differences between SnapManager for SAP, BRBACKUP, BRRESTORE, and BRRECOVER with respect to the backup, restore, recovery, and cloning of datafiles and control files.

Table 6) Differences between SnapManager for SAP and BR*Tools with respect to backup, restore, recovery and cloning of datafiles and control files.

Method	Local Backup, Restore, and Recovery	Protect Backups to Secondary, Restore from Secondary	Cloning from Local Backups or Cloning from Protected Backups on Secondary
BRBACKUP BRRESTORE BRRECOVER	Yes	No	No
SMSAP GUI/CLI	Yes	Yes (this feature is not available on Windows)	Yes (this feature is not available on Windows)

5.1.1 Backing Up Archive Logs Using BRARCHIVE

Although SnapManager for SAP currently backs up archive logs, along with datafiles and control files, it does not currently remove archive logs that have been backed up or restore archive logs. SnapManager includes archive logs with each backup only to use them for cloning from hot (online) backups. BRARCHIVE can be leveraged along with BRRESTORE and BRRECOVER to implement an archive log management solution.

The following table illustrates the differences between SnapManager for SAP, BRARCHIVE, BRRESTORE, and BRRECOVER with respect to archive log files being backed up and restored.

Table 7) Differences between SnapManager for SAP and BR*Tools with respect to backup and restore of archive logs.

Method	Local Backup, Restore, and Recovery	Protect Backups to Secondary, Restore from Secondary
BRARCHIVE BRRESTORE BRRECOVER	Yes (backup and restore)	No
SMSAP GUI/CLI	Partial (only backup, no restore)	Partial (only backup, no restore)

Using BRARCHIVE integrated with SnapManager for SAP using the BACKINT interface has the following limitations that should be considered:

- Snapshot copy backups in the archive log files volume need disk space based on the retention policy, even when the archive logs are deleted by BRARCHIVE.
- No capability to protect backups to a secondary storage system.

Therefore it is not advisable to use BRARCHIVE integrated with SnapManager for SAP using the BACKINT interface to back up archive logs.

NetApp recommends using BRARCHIVE without the integration with SnapManager for SAP to manage archive log backups, as illustrated in the following figure.

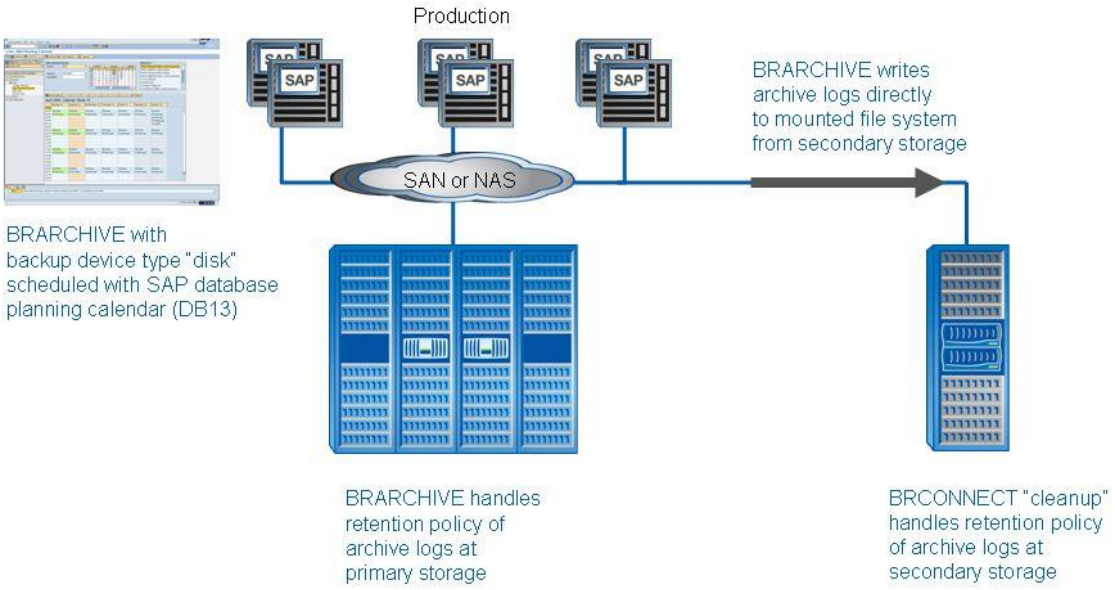


Figure 9) Archive log backup and restore using BR*Tools without BACKINT.

BRARCHIVE should be configured by setting the "`backup_dev_type`" option to "`disk`." The "`archive_copy_dir`" location is a mountpoint from a secondary storage system. The storage network to the secondary storage can be either NFS, iSCSI, or FCP.

The retention policy for archive logs at the primary storage will be controlled by executing BRARCHIVE with the options "`save`," "`delete saved`," or "`save delete`." BRCONNECT will be used with the option "`cleanup`" to handle the retention policy of the archive logs at the secondary storage. It is optional to mirror the archive logs from the secondary storage system to a third location in order to make sure that there are always two copies of the archive logs available.

Recovery using BRRECOVER will also be much faster using this approach, because BRRECOVER will not need to restore the archive logs from the secondary storage. Since the archive logs are accessible from the database host, BRRECOVER will apply the logs directly from the secondary storage without the need of restoring the logs to /`oracle/<SID>/oraarch`.

5.1.2 Backing Up Using SnapManager for SAP and BRARCHIVE

If you do want to use the SAP BRBACKUP tool to create backups and instead would only like to use SnapManager for SAP to create backups, then only a single profile needs to be created in SnapManager for SAP for each database that needs to be managed. Backups created by SnapManager for SAP can optionally be protected to a secondary storage system and can also be used to create system copies either on the local storage system (if the backup exists on primary) or on the secondary storage system (if the backup has been protected to secondary).

SnapManager 3.0 for SAP integrates with Protection Manager to automatically replicate SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator.

As mentioned previously in section 5.1.1 of this document, although SnapManager for SAP currently backs up archive logs, along with datafiles and control files, it does not currently delete archive logs that have been backed up or restore archive logs. SnapManager includes archive logs with each backup only to use them for cloning from hot (online) backups. NetApp therefore recommends using BRARCHIVE without the integration with SnapManager for SAP to manage archive logs. Refer to section 5.1.1 of this document for more details.

The following table describes which tool should be used for each task when using both SnapManager for SAP and BRARCHIVE.

Table 8) Which tool to use for each task when using both SnapManager for SAP and BRARCHIVE.

Task	Tool Used
Local backup	SMSAP GUI or CLI
Restore from local backup	SMSAP GUI or CLI
Archive log backup	BRARCHIVE with "`backup_dev_type = disk`" without SMSAP BACKINT
Archive log restore	BRRESTORE, BRRECOVER with "`backup_dev_type = disk`" without SMSAP BACKINT
Local backups to be protected to secondary	SMSAP GUI or CLI
Restore protected backup from secondary	SMSAP GUI or CLI
Cloning local backup	SMSAP GUI or CLI* * Only backups that have been created using SMSAP GUI or CLI can be cloned.
Cloning protected backup on secondary	SMSAP GUI or CLI* * Only backups that have been created using SMSAP GUI or CLI can be cloned.

To back up a database using the SnapManager GUI, right-click the profile of that database and then select "Backup…" This will launch the backup wizard that will guide you through the steps to create a backup.

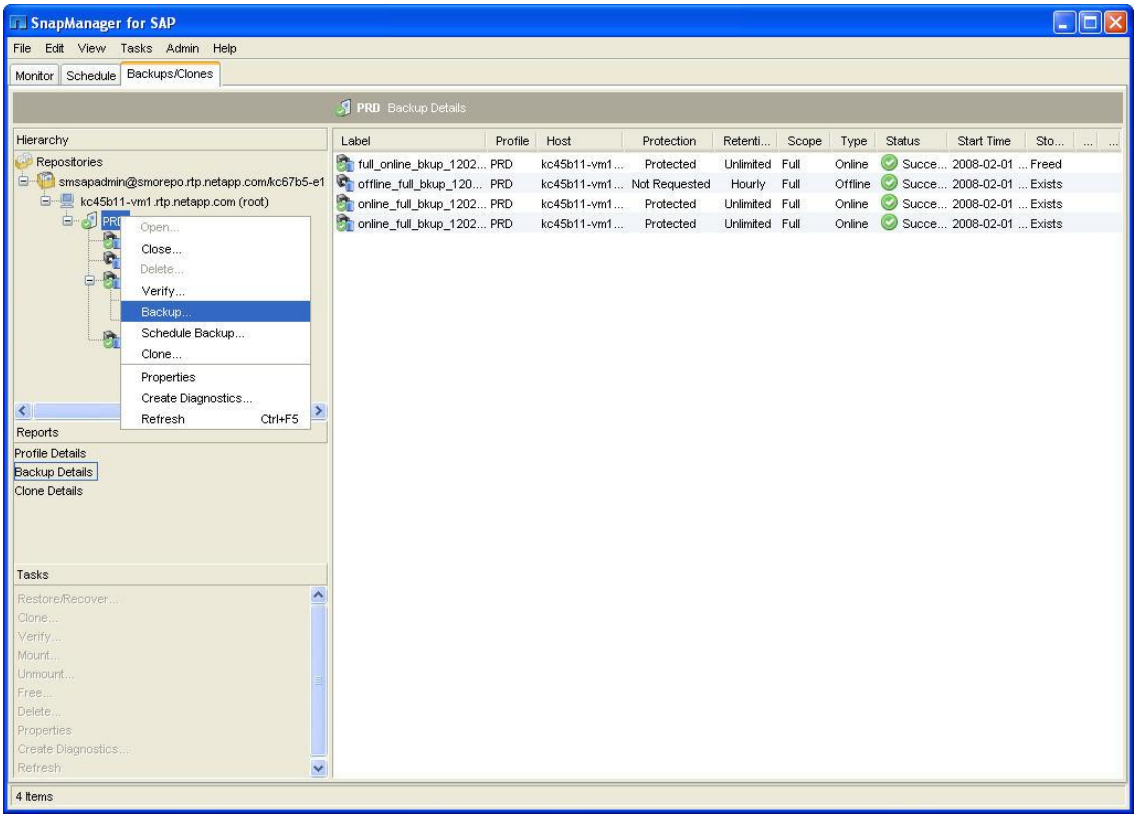


Figure 10) Creating a backup.

Table 9) Example commands to create backups using SnapManager for SAP GUI and CLI.

Creating Backups Using SnapManager for SAP	Example Commands
GUI	Right-click the profile of that database and then select "Backup…"
CLI	To create a full online backup that is exempt from being deleted by the backup retention policy and verifies the backup: `smsap backup create -online -full -profile prd_smsap -label full_bkup_sales_may_08 -verify -retain -unlimited`

PROTECTING BACKUPS CREATED USING SNAPMANAGER FOR SAP

SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager 3.7.1. This automates replicating SnapManager backups on a primary storage system to a secondary storage system or even to a tertiary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. Refer to the chapter titled "Protecting database backups on secondary storage" in the SnapManager 3.0 for SAP Installation and Administration Guide for more details.

Appendix B of this report provides instructions to configure and enable policy-driven data protection.

BACKING UP RAC DATABASES USING SNAPMANAGER FOR SAP

SnapManager 3.0 for SAP now also supports SAP systems running on Oracle RAC databases. For RAC configurations, SnapManager for SAP will perform the backup on the host specified in the SnapManager profile. If that RAC host is down for some reason, then the backup will fail.

SnapManager 3.0 for SAP also supports OS-authenticated database connections for RAC databases. The SnapManager server must be installed and running on each node in the RAC cluster for a RAC database that is using the OS-authenticated connection mode.

If using the database-authentication connection mode:

- The listener on each node that services an instance of the RAC database must be configured to use the same port number. Also, the listener that services the primary database instance must be started prior to initiating a backup.
- The password of the database user that SnapManager uses (typically `sys`) must be the same for all the Oracle instances in a RAC environment.

VERIFYING BACKUPS

SnapManager for SAP can also verify a backup to confirm that physical blocks in the backup have not been corrupted. Verification is done by invoking the Oracle Database verify utility `$ORACLE_HOME/bin/dbv`. Verification can be performed at the time of the backup or later, if desired. SnapManager performs the verification on the database host itself and currently does not provide any options to offload the verification to another host or secondary storage system.

SCHEDULING BACKUPS TO BE CREATED BY BRARCHIVE AND SNAPMANAGER FOR SAP

With the SAP "DBA Cockpit" or "db13" transaction the following tasks can be scheduled:

- Archive log backups with BRARCHIVE
- Deleting of already saved archive logs with BRARCHIVE at primary storage
- Deleting of archive logs with BRCONNECT at secondary storage

Since SnapManager 3.0 for SAP now has a built-in scheduler, local backups created by SnapManager that need to be protected to a secondary storage system can be scheduled using the built-in scheduler in SnapManager for SAP. The schedule to transfer these local backups from primary to a secondary storage system is specified in Protection Manager by the storage or backup admin.

The following table shows an example schedule of backups of a database using SnapManager for SAP, archive log backups using BRARCHIVE, and backup verification using SnapManager for SAP with the following requirements:

- A full online backup with the retention class “hourly” should be created every six hours and retained for two days on the primary storage system.
  - This can be achieved using SnapManager for SAP, and the retention policy is specified in the profile created in SnapManager for SAP.
- A full online backup with the retention class “daily” should be created once a day and retained for five days.
  - This can be achieved using SnapManager for SAP, and the retention policy is specified in the profile created in SnapManager for SAP.
- Archive logs should be backed up every hour and should not be retained for longer than six hours on the primary storage system.
  - This can be achieved by scheduling BRARCHIVE with the following mentioned options:
    - `BRARCHIVE –s` (save option) every hour
    - `BRARCHIVE –ds` (delete saved) every six hours
- Archive logs will be retained for 30 days on the secondary storage system (default value for cleanup in init<SID>.sap).
  - This can be achieved by scheduling BRCONNECT with the following mentioned option:
    - `BRCONNECT –f` cleanup will be scheduled once per week
- A full online backup should be created and protected to the secondary storage system once per day.
  - This can be achieved by scheduling a “daily” backup with SnapManager for SAP and enabling data protection.
- A full online backup with database verification should be created once per week.
  - This can be achieved by scheduling a “weekly” backup with SnapManager for SAP and checking the backup verification option.

Table 10) Example schedule of backups of a database using SnapManager for SAP, archive log backups using BRARCHIVE, and backup verification using SnapManager for SAP.

	00:00	01:00	06:00	07:00	12:00	13:00	18:00	19:00
Mon	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Tue	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Wed	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Thu	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Fri	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Sat	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Sun	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup** **BRCONNECT –f cleanup** init<SID>_Daily.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap SMSAP scheduled online backup with database verification	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup**	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap

The SnapManager for SAP scheduler can be accessed only from the SnapManager GUI and provides options to schedule backups with hourly, daily, weekly, monthly, or unlimited retention classes. Refer to the section titled “About database backup scheduling” in the SnapManager 3.0 for SAP Installation and Administration Guide for more details.

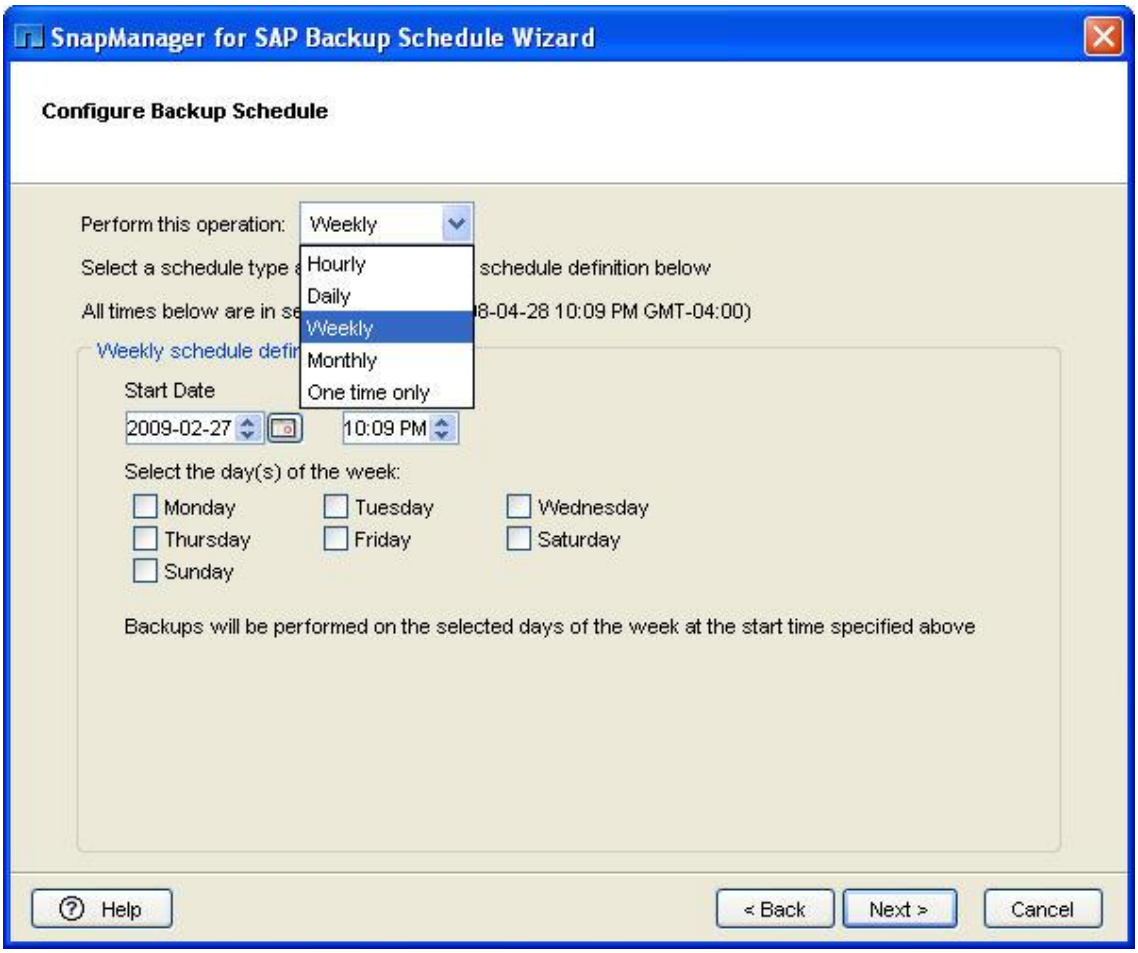


Figure 11) Scheduling a backup.

If using the SnapManager CLI, any scheduling software such as CRON in UNIX systems can be used to schedule SnapManager CLI commands.

While scheduling SnapManager backups using the `smsap backup create` CLI command, you can specify an optional name for the backup by using the `-label` parameter. This name must be unique for all backups created within a particular SnapManager profile. The name can contain letters, numbers, under bar (_), and hyphens (-). It cannot start with a hyphen. If you do not specify a label, SnapManager creates a default label for you.

If you would like to specify a label, while scheduling repetitive SnapManager backups, then you need to make sure that a unique label is specified for every backup created within a profile. You can achieve this by appending operating system environment variables such as the system date and so on to the label name, as illustrated in the following examples.

Table 11) Example SnapManager for SAP backup commands with unique backup labels.

Scheduling Backups		Example Commands
GUI		Right-click the backup and select “Schedule Backup...”

CLI	Windows	• To schedule daily full online backups of an Oracle Database running on Windows and to generate a unique label every time, you can schedule the following SnapManager command:
		`smsap backup create –online –full –profile targetdb1_prof1 –label full_hot_%date:~4,2%%date:~7,2%%date:~10,4%_%time:~0,2%h%time:~3,2%m%time:~6,2%s –retain –daily –verbose`
		• This will create a unique label for each backup in the following format:
		`full_hot_07172008_13h32m34s`

	Linux	• To schedule daily full online protected backups of an Oracle Database running on Linux and to generate a unique label every time, you can schedule the following SnapManager command:
		`smsap backup create –online –full –profile targetdb1_prof1 –label `date +"full_hot_"%m%d%Y"_"%H"h"%M"m"%S"s"` -protect –retain –daily -verbose`
		• This will create a unique label for each backup in the following format:
		`full_hot_07172008_13h32m34s`

BACKUP AND TRANSFER SCHEDULES USING SNAPMANAGER FOR SAP AND PROTECTION MANAGER

While creating a backup using either the SnapManager GUI or CLI, you can specify a retention class of hourly, daily, weekly, monthly, or unlimited for that backup. The number of backups of each retention class that you would like to retain can be specified in the SnapManager profile of a database.

If data protection is enabled for a SnapManager profile, all backups created under that profile will be replicated to secondary storage based on the transfer schedule specified in Protection Manager. Because of a known issue it is critical to synchronize the primary backup schedule in SnapManager for SAP with the transfer schedule in Protection Manager so that the same retention class for each backup is maintained on both primary and secondary storage. Make sure that the transfers of the correct retention class occur after the backups with the same retention class complete.

For example, if you set up the following two schedules in SnapManager for backups on primary:

- Hourly backups on each half hour (for example, at 20:30, 21:30, 22:30, and so on)
- Daily backups at 22:30

Note that when two schedules have a conflict, the one with the less frequent retention class wins. In the above example, since both the schedules need to create a backup at 22:30, SnapManager will create a single backup at 22:30 and assign it a retention class of daily.

Assuming the backups are completed in a few minutes, set up the transfer schedule in Protection Manager as follows:

- Hourly transfers on the hour (for example, at 20:00, 21:00, 22:00)
- 23:00 daily

In this way the transfers of the correct retention class occur after the backups with the same retention class complete, and the backups on primary and secondary storage have the same retention class.

NOTIFYING BACKUP STATUSES USING SNAPMANAGER FOR SAP

Although SnapManager for SAP currently does not provide the capability to notify users the status of various SnapManager operations, you can create scripts that leverage the SnapManager CLI and send out notifications or alerts.

Appendix J of this report has a sample script that will create a backup and send out an e-mail notification with the status and the log of the backup operation. Scripts for other SnapManager operations can be created similarly.

FREEING BACKUPS USING SNAPMANAGER FOR SAP

In version 1.1 of SnapManager for SAP, due to the lack of integration with Protection Manager, SnapManager provided the following features to aid in scripting a solution that leverages NetApp SnapVault and SnapMirror to replicate the SnapManager backups from primary to secondary storage:

- Freeing backups: This option deletes the Snapshot copies associated with a backup without deleting the metadata of that backup in the SnapManager repository. Freeing of backups is recommended only after the associated Snapshot copies are copied to secondary storage. The metadata of the backup is retained in the SnapManager repository to aid restoring and recovery of the database using a copy of the backup in an alternate location. Freeing backups has the following benefits:
  - Frees up space on the primary storage since the associated Snapshot copies are deleted.
  - Enables more frequent backups.
  - Helps avoid reaching the Snapshot copy limit on the volumes.
  - Helps create a scripted disaster recovery solution.
- Restoring from an alternate location

Refer to section 6.1 of this report to understand how these two features can be used to provide a scripted disaster recovery solution.

In version 3.0, SnapManager integrates with Protection Manager to leverage SnapVault and SnapMirror and automates replicating the SnapManager backups from primary to secondary storage. Freeing backups and restoring from an alternate location are still available in version 3.0 as well and can be leveraged in SnapManager environments where Protection Manager is not available and instead a scripted solution is desired to replicate the SnapManager backups from primary to secondary storage.

When you free a protected backup, SnapManager removes the local Snapshot copies for the backup, keeps the remote Snapshot copies, and leaves the backup in the SnapManager repository so it can be restored from secondary storage.

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

To free a backup using the SnapManager GUI, right-click the backup and select “Free...” This will launch the wizard that will guide you through the steps to free a backup.

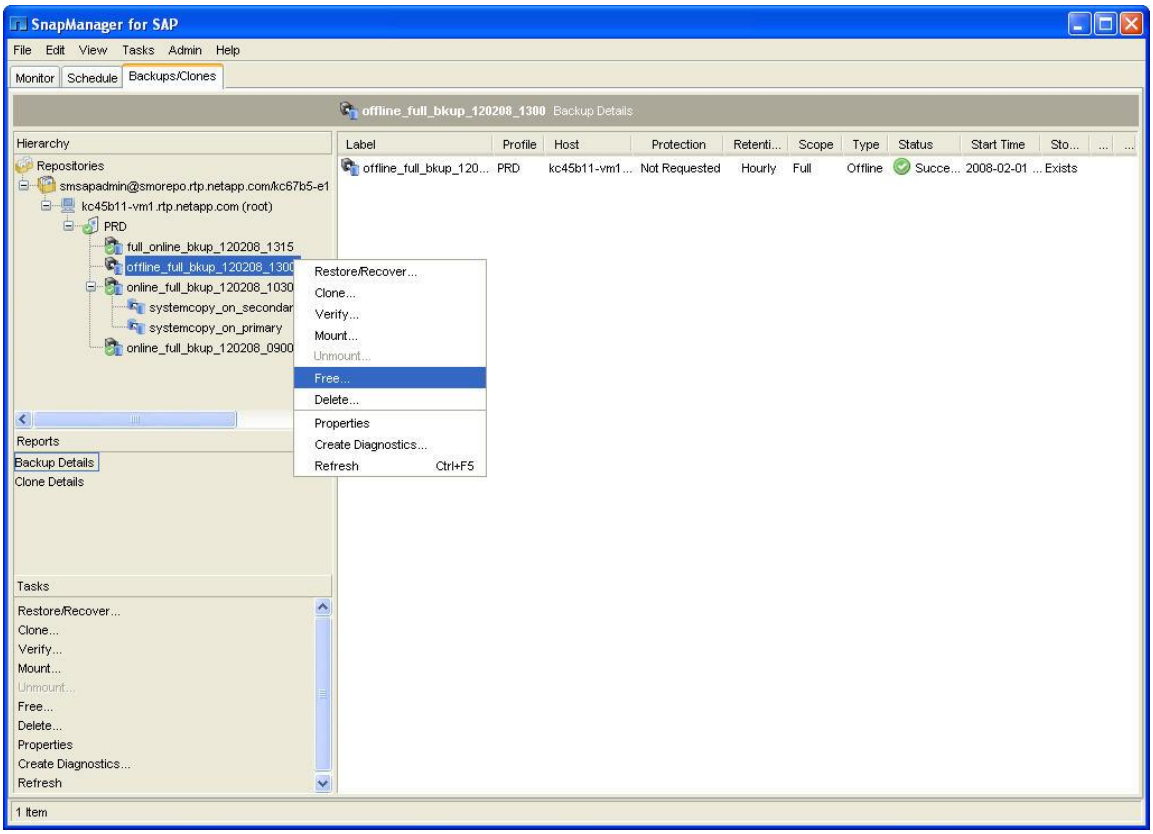


Figure 12) Freeing a backup.

Table 12) Example commands to free backups using SnapManager for SAP.

Freeing Backups	Example Commands
GUI	Right-click the backup and select “Free...”
CLI	`smsap backup free -profile prd -label full_backup_sales_apr_08`

DELETING BACKUPS USING SNAPMANAGER FOR SAP

Deleting a backup will not only delete the associated Snapshot copies but will also delete the metadata of that backup in the SnapManager repository. To delete a backup using the SnapManager GUI, right-click the backup and select “Delete...” This will launch the wizard that will guide you through the steps to delete a backup.

Table 13) Example commands to delete backups using SnapManager for SAP.

Deleting Backups	Example Commands
GUI	Right-click the backup and select “Delete...”
CLI	`smsap backup delete -profile prd -label full_backup_sales_jan_07`

FREEING VERSUS DELETING BACKUPS USING SNAPMANAGER FOR SAP

The following table explains the differences between freeing and deleting a backup.

Table 14) Differences between freeing and deleting backups.

	Freeing a Backup		Deleting a Backup	
GUI navigation	• Right-click the backup • Select “Free…”		• Right-click the backup • Select “Delete…”	
CLI execution	`smsap backup free …`		`smsap backup delete …`	
Are the associated Snapshot copies deleted?	If backup is not protected	Yes	If backup is not protected	Yes
	If backup is protected	Yes on primary but retains the protected Snapshot copies on secondary. **Note:** If using a mirror relationship, then Snapshot copies are deleted on the primary node, and the protected Snapshot copies are also deleted from the mirror nodes when the next transfer to secondary occurs.	If backup is protected	Yes on primary and secondary
Does it free up space occupied by the associated Snapshot copies?	Yes		Yes	
Is the backup record retained in SnapManager after the operation?	Yes		No	
After performing this operation on a backup, can the database be restored from that backup using SnapManager?	Yes, provided you have a protected backup on secondary or a copy of the backup in an alternate location.		No	
When to use?	Use after copying a backup to a secondary location using scripts that leverage SnapVault, SnapMirror, or any other means. For protected backups, SnapManager will automatically free the backups on primary based on the backup retention policy specified in SnapManager.		Use when a backup will never be needed	

5.1.3 Backing Up Using BRBACKUP, SnapManager for SAP, and BRARCHIVE

You can use the SAP BRBACKUP tool to create a backup of your SAP system using the BACKINT interface. But as mentioned previously, since BRBACKUP does not protect local backups to a secondary storage system and since system copies cannot be created from backups created by BRBACKUP, you might also want to use SnapManager for SAP GUI or CLI to create additional backups that can be protected to a secondary storage system, and also if required SnapManager for SAP can be used to create system copies from these protected backups on the secondary storage system.

When using both BR*Tools and SnapManager for SAP together, NetApp recommends using the same tool used for backing up to also be used to perform restore and recovery. This means if you use SnapManager for SAP to create a particular backup, then use SnapManager for SAP to restore and recover from that

backup. Similarly, if you have configured SAP BR*Tools to integrate with SnapManager for SAP using the BACKINT interface and you then use BRBACKUP to create a particular backup, then use SAP BR*Tools such as BRRESTORE and BRRECOVER to restore and recover from that backup.

The following table describes which tool should be used for each task when using both BR*Tools and SnapManager for SAP GUI or CLI.

Table 15) Which tool to use for each task when using both SnapManager for SAP and BR*Tools.

Task	Tool Used
Local backup	BRBACKUP with SMSAP BACKINT
Restore from local backup	BRRESTORE, BRRECOVER with SMSAP BACKINT
Archive log backup	BRARCHIVE with "`backup_dev_type = disk`" without SMSAP BACKINT
Archive log restore	BRRESTORE, BRRECOVER with "`backup_dev_type = disk`" without SMSAP BACKINT
Local backups to be protected to secondary	SMSAP GUI or CLI Additional daily or on-demand backups
Restore protected backup from secondary	SMSAP GUI or CLI
Cloning local backup	SMSAP GUI or CLI* * Only backups that have been created using SMSAP GUI or CLI can be cloned.
Cloning protected backup on secondary	SMSAP GUI or CLI* * Only backups that have been created using SMSAP GUI or CLI can be cloned.

To allow backups to be created by both BRBACKUP and SnapManager for SAP, two profiles will have to be created in SnapManager for SAP for each database that needs to be backed up.

One profile will be used for the backups that will be created by BRBACKUP. The other profile will be used for the backups that will be created by SnapManager for SAP GUI or CLI for data protection and SAP system copies. Both these profiles can be identical except for the profile name, backup retention policy, and the data protection option, as described in the following table. For more details on how to create the profile for BR*Tools commands, follow the steps listed in the SnapManager 3.0 for SAP Installation and Administration Guide under the section titled "Specifying the profile for BR*Tools backups."

Table 16) Profiles to be created for backing up a database using both BRBACKUP and SnapManager for SAP.

Profile Purpose and Name	Used For	Retention Policies for Local Backups	Data Protection Enabled	Comments
Profile for BR*Tools Example name: <SID>_BRTOOLS	Local backup, restore, and recovery using BRBACKUP, BRRESTORE, and BRRECOVER	For example: six hourly, five daily	No	Hourly Snapshot copies every six hours, retained for two days. Daily Snapshot copy once per day, retained for five days.

Profile for SnapManager GUI/CLI Example name: <SID>_SMSAP	Local backups, protecting local backups to secondary leveraging Protection Manager, restore from secondary, cloning on secondary using SnapManager for SAP	For example: one daily	Yes	Daily Snapshot copy once per day and retained for one day. More daily backups are retained on the secondary.

SAP creates a backup profile file named `init<SID>.sap` for each SAP database instance in `$ORACLE_HOME/dbs/`, where SID is the system identifier of the database. You can use this file to specify the default backup utility parameter (`.utl`) file used for BR*Tools commands. In order to be able to use different retention classes with BRBACKUP backups, it is necessary to create different `init<SID>.sap` parameter files, as shown in the following table. For archive log backups with BRARCHIVE, a separate parameter file will be used in order to allow configuring the disk destination for the backups instead of using BACKINT.

Table 17) Details of the `init<SID>.sap` and `init<SID>.utl` files for some sample retention classes.

Sample Retention	InitSID.sap file	init<SID>.utl file
BRBACKUP Retention class hourly	**init<SID>_Hourly.sap** backup_dev_type = util_file util_par_file = init<SID>_Hourly.utl	**init<SID>_Hourly.utl** profile_name = SID_BRTOOLS fast = override retain = hourly protect = no
BRBACKUP Retention class daily	**init<SID>_Daily.sap** backup_dev_type = util_file util_par_file = init<SID>_Daily.utl	**init<SID>_Daily.utl** profile_name = SID_BRTOOLS fast = override retain = daily protect = no
BRARCHIVE	**InitSID_BRARCHIVE.sap** backup_dev_type = disk archive_copy_dir = /mnt/backup2disk	NA

SCHEDULING BACKUPS TO BE CREATED BY BRBACKUP, BRARCHIVE, AND SNAPMANAGER FOR SAP

Because SnapManager for SAP is integrated with SAP BR*Tools, backups can also be scheduled using the SAP database planning calendar. There are some changes that need to be made to the configuration of the system before backups can be run from the SAP database planning calendar. Refer to the chapter titled "Scheduling backups with SAP transaction DB13" in the SnapManager 3.0 for SAP Installation and Administration Guide for more details. Refer to the SAP Help Portal for detailed instructions for using the database planning calendar.

With SAP transaction DB13 the following tasks can also be scheduled:

- Database backups with BRBACKUP and retention class "hourly"
- Database backups with BRBACKUP and retention class "daily"
- Archive log backups with BRARCHIVE
- Deleting of already saved archive logs with BRARCHIVE at primary storage

- Deleting of archive logs with BRCONNECT at secondary storage

Since SnapManager 3.0 for SAP now has a built-in scheduler, local backups created by SnapManager that need to be protected to a secondary storage system can be scheduled using the built-in scheduler in SnapManager for SAP. The schedule to transfer these local backups from primary to a secondary storage system is specified in Protection Manager by the storage or backup admin.

The following table shows an example schedule of backups of a database using BRBACKUP and SnapManager for SAP, archive log backups using BRARCHIVE, and backup verification using SnapManager for SAP with the following requirements:

- A full online backup with the retention class “hourly” should be created every six hours and retained for two days on the primary storage system.
  - This can be achieved using BRBACKUP integrated with SnapManager for SAP using BACKINT, and the retention policy is specified in the BR*Tools profile created in SnapManager for SAP.
- A full online backup with the retention class “daily” should be created once a day and retained for five days.
  - This can be achieved using BRBACKUP integrated with SnapManager for SAP using BACKINT, and the retention policy is specified in the BR*Tools profile created in SnapManager for SAP.
- Archive logs should be backed up every hour and should not be retained for longer than six hours on the primary storage system.
  - This can be achieved by scheduling BRARCHIVE with the following mentioned options:
    - `BRARCHIVE -s` (save option) every hour
    - `BRARCHIVE -ds` (delete saved) every six hours
- Archive logs will be retained for 30 days on the secondary storage system (default value for cleanup in init<SID>.sap).
  - This can be achieved by scheduling BRCONNECT with the following mentioned option:
    - `BRCONNECT -f` cleanup will be scheduled once per week,
- A full online backup should be created and protected to the secondary storage system once per day.
  - This can be achieved by scheduling a “daily” backup with SnapManager for SAP and enabling data protection.
- A full online backup with database verification should be created once per week.
  - This can be achieved by scheduling a “weekly” backup with SnapManager for SAP and checking the backup verification option.

Table 18) Example schedule of backups of a database using BRBACKUP and SnapManager for SAP, archive log backups using BRARCHIVE, and backup verification using SnapManager for SAP.

	00:00	01:00	06:00	07:00	12:00	13:00	18:00	19:00
Mon	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Daily.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Tue	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Daily.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Wed	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Daily.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Thu	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Daily.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Fri	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Daily.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Sat	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Daily.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap **SMSAP scheduled online backup with data protection**	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap
Sun	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Daily.sap **BRCONNECT –f cleanup** init<SID>_Daily.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap SMSAP scheduled online backup with database verification	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap	**BRARCHIVE –s** every hour init<SID>_BRARCHIVE.sap **BRBACKUP online** init<SID>_Hourly.sap	**BRARCHIVE –ds** init<SID>_BRARCHIVE.sap

CHANGES IN VERSION 3.0 THAT AFFECT BACKUPS

- SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. This functionality is optional and is not available on Windows.
- You can now specify a custom naming convention for Snapshot copies created by backups under a SnapManager profile. Custom text or built-in variables such as profile name, database name, and database system identifier (SID) provided by SnapManager can be used to generate the naming convention.
- SnapManager 3.0 for SAP now has a built-in scheduler for backups. The scheduler can be accessed only from the SnapManager GUI and provides options to schedule backups with hourly, daily, weekly, monthly, or unlimited retention classes.
- If creating a backup using the SnapManager GUI, the backup operation can now be run in the background by clicking the new "Background" button available in the backup wizard. This allows SAP administrators to execute multiple SnapManager operations in parallel.

BEST PRACTICES AND REQUIREMENTS FOR BACKING UP

- Although SnapManager can perform partial and full backups, NetApp recommends always performing a full backup of the database when using SnapManager for SAP, as this minimizes the number of Snapshot copies that SnapManager creates. Refer to the section titled "About full and partial backups" in the SnapManager 3.0 for SAP Installation and Administration Guide for more details.
- NetApp recommends replicating backups of all critical databases created using SnapManager for SAP to a secondary storage system by leveraging the policy-driven data protection feature in version 3.0. Refer to Appendix B for instructions to configure and enable policy-driven data protection.
- If Protection Manager is not available and the policy-driven data protection feature in version 3.0 cannot be leveraged, then NetApp recommends that backups created by SnapManager are archived or mirrored off the mainline storage using a scripted solution that leverages NetApp SnapVault or SnapMirror technology. Refer to Appendix I and TR-3437, "Storage Best Practices and Resiliency Guide," for more details.
- If using a scripted solution to replicate backups to a secondary storage system, NetApp recommends freeing up backups on primary only after they have been copied to secondary storage.
- If using SnapVault, NetApp recommends setting the number of backups to be retained to a large number so that vaulting a backup can occur before the backup is pruned from the primary storage system.
- The target database SID should be included in the oratab file. SnapManager relies on the oratab file to determine the Oracle home directory for the managed database.
- For RAC databases, the listener that services the primary database instance must be started prior to initiating a backup.
- If data protection is enabled for a profile and backups are being replicated to secondary storage, then it is critical to synchronize the primary backup schedule in SnapManager for SAP with the transfer schedule in Protection Manager so that the same retention class for each backup is maintained on both primary and secondary storage. Make sure that the transfers of the correct retention class occur after the backups with the same retention class complete. Refer to the "Backup and transfer schedules" section of this report for example schedules. This is a known issue and will be fixed in a future release.
- NetApp recommends using the built-in scheduler in the SnapManager GUI for scheduling backups. This provides SAP administrators using the SnapManager GUI with the added benefit of managing backup schedules and backups from the same interface. Also, backup schedules created and modified by one SAP administrator are visible to all other SAP administrators using the SnapManager GUI.
- NetApp recommends that verification be completed on a regular basis against the completed backups during low activity. However, avoid running verifications while Snapshot copies are being created.
- NetApp recommends freeing up backups only after they have been copied to secondary storage.
- NetApp recommends deleting database backups that are no longer in use to free up the space the backups occupied. Use the "`smsap backup delete …`" command to delete backups created by SnapManager. This will also reduce the chance of reaching the limit of 255 Snapshot copies per volume.

- NetApp recommends deleting backups created by SnapManager using only the SnapManager GUI or CLI. Deleting Snapshot copies created by SnapManager from SnapDrive or Data ONTAP will cause inconsistency between the environment and the SnapManager repository.
- If using SnapVault, NetApp recommends setting the number of backups to be retained to a large number so that vaulting a backup can occur before the backup is pruned from the primary storage system.
- If using archive logs stored on a storage system that is not running Data ONTAP, NetApp recommends excluding such archive logs from consideration for backup with SnapManager. The `smsap.config` file enables you to exclude certain archive log files. The file is located in:

  `/opt/NTAP/smsap/properties/smsap.config` (Solaris)

  `/opt/NetApp/smsap/properties/smsap.config` (for all other platforms)

  Follow the format suggested within the file to exclude the local archive logs.

5.2 RESTORE AND RECOVERY

SnapManager for SAP restores the database to the state it was in at the time the Snapshot copy was created. SnapManager leverages NetApp SnapRestore technology, which shortens the restore time significantly compared to traditional recovery methods. Since backups can now be created more frequently, the amount of logs that need to be applied is drastically reduced, thus reducing the mean time to recovery (MTTR) for a database.

When using both BR*Tools and SnapManager for SAP together, NetApp recommends that the same tool used for backing up also be used to perform restore and recovery. This means if you use SnapManager for SAP to create a particular backup, then use SnapManager for SAP to restore and recover from that backup. Similarly, if you have configured SAP BR*Tools to integrate with SnapManager for SAP using the BACKINT interface and you then use BRBACKUP to create a particular backup, then use SAP BR*Tools such as BRRESTORE and BRRECOVER to restore and recover from that backup.

RESTORING USING BRRESTORE

The process for restoring systems using SnapManager for SAP and BRRESTORE is the same as restoring with any other utility-based backup solution. It is necessary to determine the log file name for the backup you want to restore and then issue the BRRESTORE command as user `ora<sid>`. For example, to restore the online backup with the logfile `bdvlrfxx.anf`, issue the following command as user `ora<sid>`:

```
brrestore -b bdvlrfxx.anf -d util_file -m full
```

In the case that you want to restore only certain datafiles or even a single tablespace, you must specify these options in the BRRESTORE command. This can take quite some time and make the BRRESTORE command very long and cumbersome, which could lead to input errors. It is also necessary to determine the correct command syntax.

If the backup being restored was an online backup, when the restore has completed it will be necessary to perform a database recovery. A database recovery can be performed using the BR*Tools program BRRECOVER or native database tools.

Refer to the SAP Online Help for more information regarding the use of BRRESTORE and BRRECOVER.

PERFORMING RESTORE AND RECOVERY USING THE SNAPMANAGER FOR SAP GUI

To restore and recover from a backup using the SnapManager GUI, right-click the backup and select “Restore/Recover…” This will launch the restore and recovery wizard that will guide you through the steps to restore and recover your database using the backup you selected.

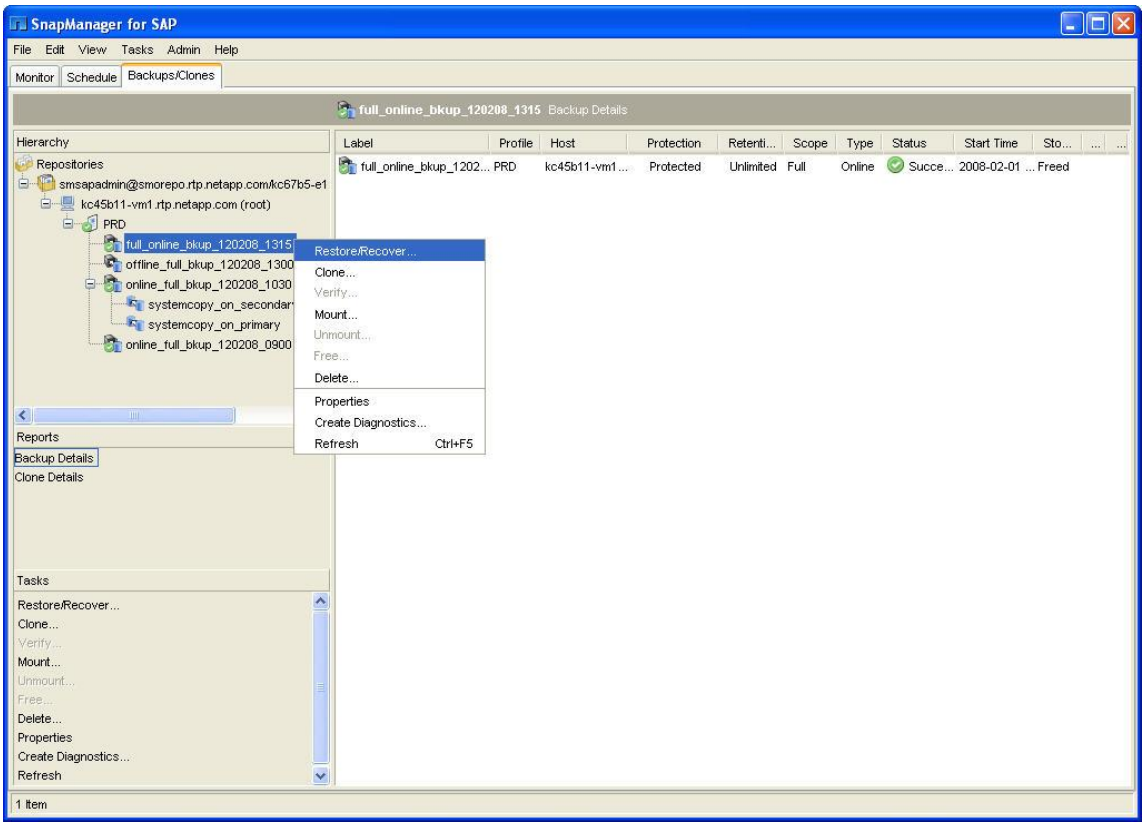


Figure 13) Restore and recovery.

Part of the restore and recovery wizard is selecting the point in time for recovering the database. When the restore is completed, SnapManager for SAP will automatically recover the database to the point specified, reducing the time necessary to make the system available to the end users.

If one or more datafiles or tablespaces need to be restored, this can be easily accomplished using the SnapManager GUI. By choosing the "Selective Restore" option in the restore and recovery wizard, you can quickly and easily select the tablespaces or files you want to restore. These files will then be restored and recovered in one step.

Table 19) Example commands to restore and recover using SnapManager for SAP GUI and CLI.

Restore and Recovery	Example Commands
GUI	Right-click the backup and select "Restore/Recover…"
CLI	To restore an entire backup along with the control files and recover until the last transaction: `smsap backup restore -profile targetdb1_prof1 -label full_bkup_sales_feb_08 -complete -controlfiles -recover -alllogs`

FAST RESTORES

SnapManager 3.0 for SAP now provides a faster volume-based restore option, which is now the default. This is the fastest possible restore mechanism among all the restore mechanisms that SnapManager offers. Prior versions of SnapManager only performed file-based restores. SAP administrators can leverage the fast restore feature to restore a database in minutes irrespective of the size of the database. This functionality is optional and is not available on Windows. Refer to the section titled "About restoring database backups" in the SnapManager 3.0 for SAP Installation and Administration Guide for more details.

Note: Performing a fast restore can have the following negative consequences and therefore must be used with caution:

- The entire storage-side volume is restored, including:
  - Files that were not considered part of the backup
  - Other files, file systems, or LUNs in the volume
- All the Snapshot copies that were created after the Snapshot to which the volume is being restored will be deleted. For example, you can no longer restore Tuesday's backup if you fast restored Monday's backup.
- Relationships to secondary storage systems will be broken if the restored Snapshot copy is older than the baseline Snapshot copy in the relationship.

When you choose to perform a fast restore of a backup, SnapManager first performs mandatory and overridable eligibility checks to determine whether it can use the fast restore process. Refer to the section titled “Fast restore eligibility checks” in the SnapManager 3.0 for SAP Installation and Administration Guide for a complete list of these checks.

SnapManager fast restores only datafiles. To be able to fast restore backups of a database, the following guidelines have to be considered while planning the layout of the database:

- For file systems and disk groups:
  - Multiple databases cannot share the same disk group.
  - A disk group containing datafiles cannot contain other types of files. Temporary datafiles can exist on the same disk group as the regular datafiles.
  - The LUNs for the datafile disk group must be the only objects in the storage volume.
- For volume separation:
  - Datafiles for only one database must be in the volume and the volume cannot contain other types of files. Temporary datafiles can exist on the same volume as the regular datafiles.

PREVIEWING RESTORES

SnapManager 3.0 for SAP also provides a new preview option to review a file-by-file analysis of a restore operation before it takes place. Previewing a restore operation provides 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

NetApp recommends using this option for all fast restores.

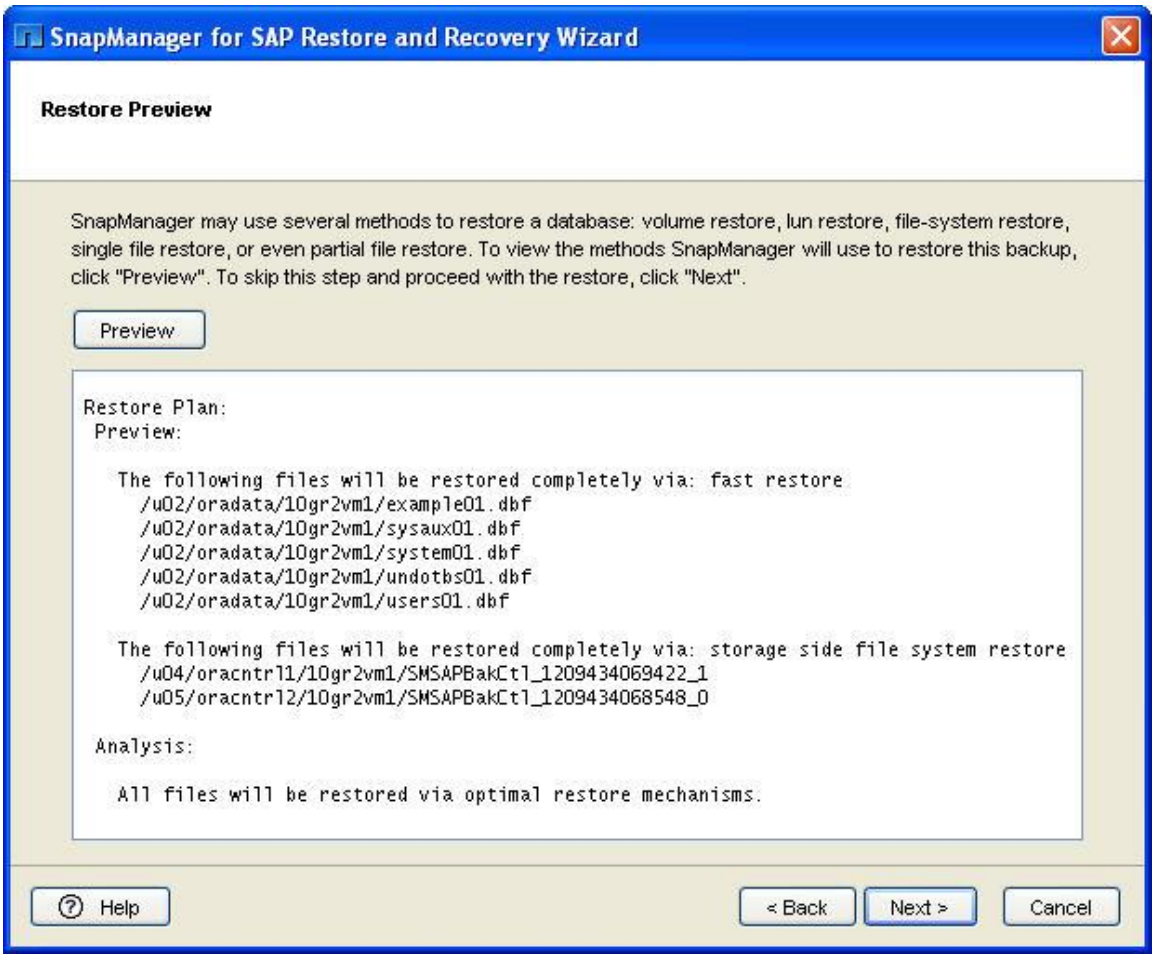


Figure 14) Previewing a restore operation.

The following screenshot illustrates why a fast restore of the datafiles cannot be performed in certain cases. Performing a fast restore can have negative consequences in such cases and therefore must be used with caution.

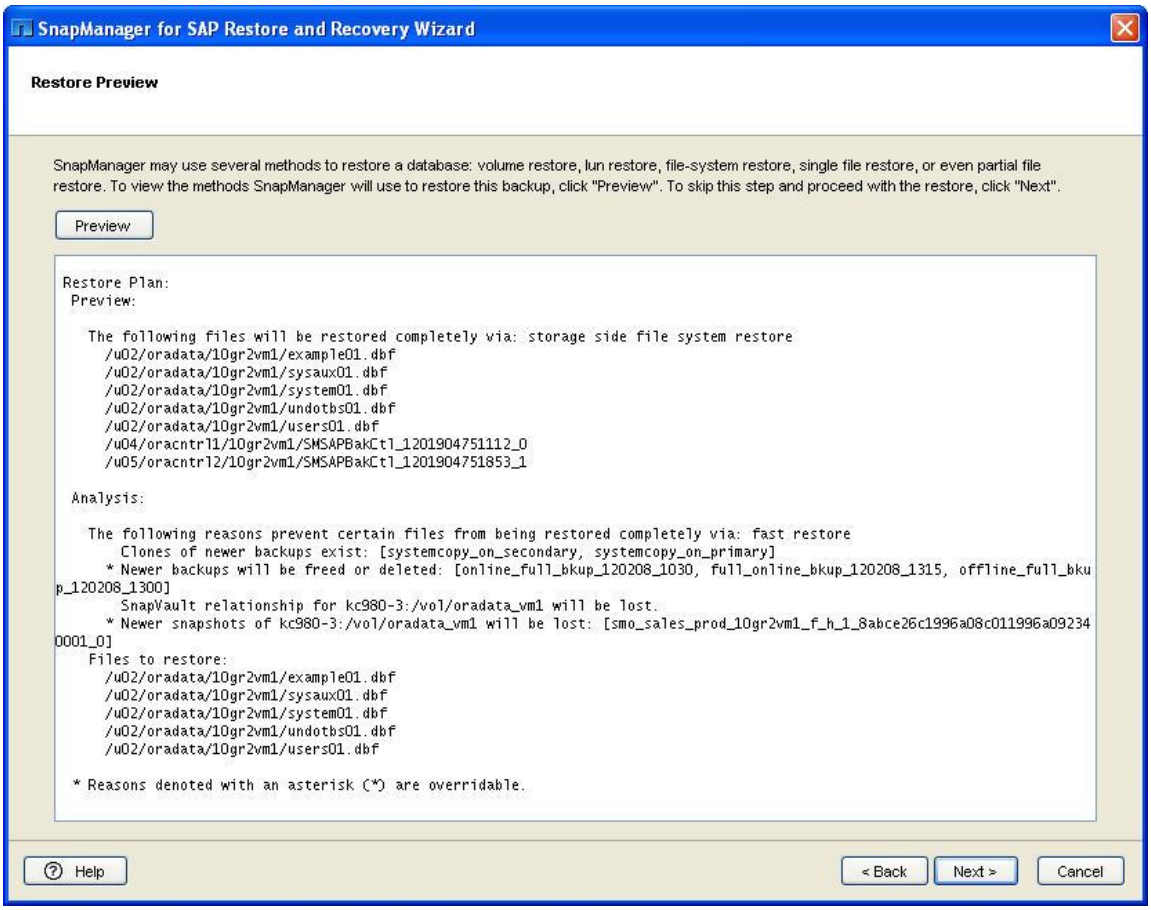


Figure 15) Restore preview screen with impact analysis.

RESTORING PROTECTED BACKUPS

SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager 3.7.1. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. SnapManager 3.0 for SAP also empowers the SAP administrator to automatically restore such protected backups from the secondary storage system back to the primary storage system. This functionality is optional and is not available on Windows. Refer to the section titled “Restoring protected backups from secondary storage” in the SnapManager 3.0 for SAP Installation and Administration Guide for more details.

RESTORING FROM AN ALTERNATE LOCATION

In version 1.1 of SnapManager for SAP, due to the lack of integration with Protection Manager, SnapManager provided the following features to aid in scripting a solution that leverages NetApp SnapVault and SnapMirror to replicate the SnapManager backups from primary to secondary storage:

- Freeing backups
- Restoring from an alternate location

Versions prior to SnapManager 1.1 for SAP could only restore from the original Snapshot copies associated with a backup. Starting with version 1.1 of SnapManager, if the backup on primary storage is freed or the Snapshot copies associated with the backup do not exist on primary, you can restore and recover from a location other than the Snapshot copies in the original volume.

To restore from an alternate location, you must first create a restore specification XML file that specifies the mappings SnapManager requires to restore from (the alternate location) and to the original location. The original location is the location of the file on the active file system at the time of the backup, and the alternate

location is the location from which a file will be restored. An example of the restore specification XML file is provided in Appendix H of this report. For more details about how to restore from an alternate location, refer to the section titled “Restoring backups from an alternate location” in the SnapManager 3.0 for SAP Installation and Administration Guide.

Here are the basic steps to restore and recover from an alternate location:

1. Based on your database layout and what needs to be restored:
    a. Restore the required datafiles from tape, SnapVault, SnapMirror, or any other media to any file system mounted on the database host.
    b. Restore the required file system and mount it on the database host.
    c. Connect to the required raw devices that exist in the local host.
2. Create the restore spec XML file with the original and alternate locations’ mappings and save the file in a location accessible from the SnapManager GUI or CLI.
3. Use the SnapManager GUI or CLI to restore and recover and specify the location of the restore spec XML file.

The screenshot following shows how you can choose to restore from an alternate location using the SnapManager restore and recovery wizard and by selecting the “Use alternative restore specification” option to specify the location of the restore spec XML file. Refer to section 6.1 of this report to understand how this feature can be used to provide a scripted disaster recovery solution.

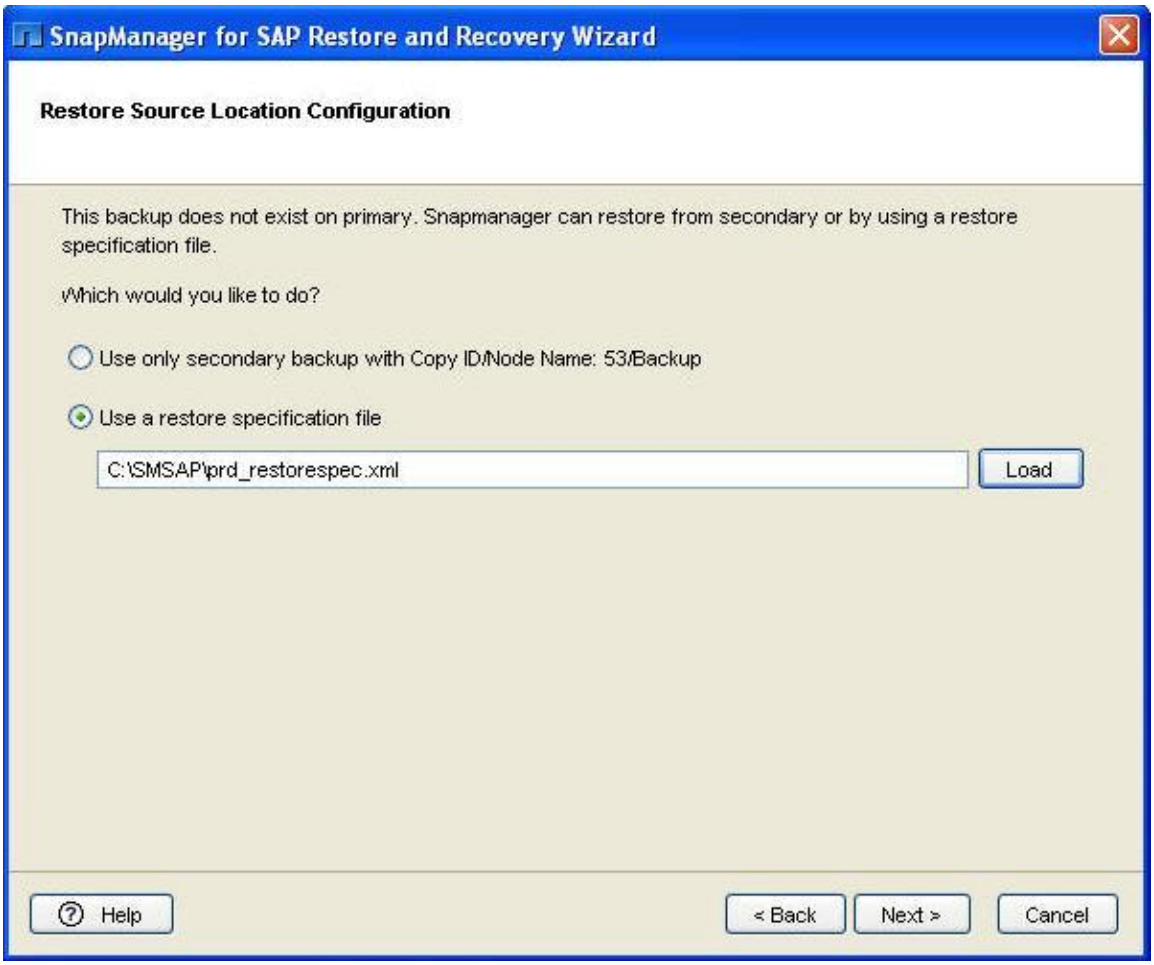


Figure 16) Restoring from an alternate location.

RESTORE AND RECOVERY OF RAC DATABASES

SnapManager for SAP can be invoked from any RAC database node to restore and recover. The RAC node where SnapManager performs the restore and recovery operation need not be same as the node where the backup was performed.

RESTORING ARCHIVE LOGS

Although SnapManager for SAP backs up archive logs, it does not currently restore or manage archive logs. SnapManager uses the archive logs only while cloning the backup. If you need to manage archive logs, then you can either use RMAN or a scripted solution.

SnapManager expects the archive logs to exist in the original location during the recovery process. If the archive logs are accidentally deleted or do not exist in the original location, then you need to first mount the appropriate SnapManager backups that contain the missing archive logs and manually copy the missing archive logs to the original location before initiating the SnapManager restore and recovery operation.

COMMON RESTORE AND RECOVERY SCENARIOS

Some common database restore and recovery scenarios using SnapManager for SAP are presented in Appendix E.

CHANGES IN VERSION 3.0 THAT AFFECT RESTORE AND RECOVERY

- SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. SnapManager 3.0 for SAP also empowers the SAP administrator to automatically restore such protected backups from the secondary storage system back to the primary storage system. This functionality is optional and is not available on Windows.
- SnapManager 3.0 for SAP now provides a faster volume-based restore option, which is now the default. This is the fastest possible restore mechanism among all the restore mechanisms that SnapManager offers. Prior versions of SnapManager only performed file-based restores.
- SnapManager 3.0 for SAP also provides a new preview option to review a file-by-file analysis of a restore operation before it takes place.

BEST PRACTICES AND REQUIREMENTS FOR RESTORE AND RECOVERY

- Performing a fast restore can have the following negative consequences and therefore must be used with caution:
  - The entire storage-side volume is restored, including:
    - Files that were not considered part of the backup
    - Other files, file systems, or LUNs in the volume
  - All the Snapshot copies that were created after the Snapshot copy to which the volume is being restored will be deleted.
  - Relationships to secondary storage systems will be broken if the restored Snapshot copy is older than the baseline Snapshot copy in the relationship.
- NetApp recommends using the new restore preview option first to analyze the effect of the fast restore operation before actually executing the fast restore operation.
- NetApp recommends using the new restore preview option before executing any restore operation.
- NetApp recommends using the SnapManager for SAP GUI to perform database and recovery because of the flexibility, ease of use, and ability to restore and recover the database in a single step provided by the graphical user interface.

5.3 CLONING

System copies are an important part of any SAP environment. A unique feature of SnapManager for SAP is its ability to automate cloning of Oracle Databases. Using NetApp FlexClone technology, SnapManager creates writable clones of the Snapshot copy created during backup. Database clones are created quickly, and clones only consume enough storage to hold modified blocks. Because the clone is based on a Snapshot copy, modifying a clone has no effect on the source database. As a result, each developer or QA engineer can be provided with his or her own personal copy of the database. Developers and QA engineers

can make modifications to these personal copies and even destroy them, if needed, without affecting other users.

Only backups created using SnapManager for SAP can be cloned. Backups created using SAP BR*Tools integrated with SnapManager for SAP using the BACKINT interface cannot be cloned.

CLONING PREPROCESSING

Before a clone can be created, the target system needs to be prepared. SnapManager for SAP only clones the Oracle datafiles. As a result, a SAP system has to be installed on the clone target host before a clone can be attached there. You will need to determine the mountpoints for the cloned file systems and make sure that these mountpoints are available.

SnapManager will create new configuration files during the clone process. The files from the current installation should be renamed or moved. The files `ORACLE_HOME/dbs/init<SID>.ora` and `ORACLE_HOME/dbs/orapwSID` should be moved to another location or renamed. If these files still exist during the clone, the clone will fail.

SnapManager will need a location for trace files to be written. Oracle installations for SAP put trace files in `/oracle/<SID>/saptrace/background` and `/oracle/<SID>/saptrace/usertrace`. SnapManager will create a directory in each of those locations based on the cloning specification. Make sure the directory for the trace files does not already exist, or the clone will fail.

SnapManager will need a location for log archiving. Oracle installations for SAP use `/oracle/<SID>/oraarch` for archived log files. SnapManager will create a directory in that file system based on the cloning specification. Make sure the directory for the archive logs does not already exist, or the clone will fail.

AUTOMATED CLONING PREPROCESSING

SnapManager 3.0 for SAP can now automate executing custom scripts before and after the clone creation process. SnapManager ships with a sample pre processing script `cleanup.sh` in the `<SnapManager_install_directory>/plugins/examples/clone/create/pre` directory that automates the preprocessing tasks mentioned in the previous section. Refer to the section titled “Cloning databases and using custom plug-in scripts” in the SnapManager 3.0 for SAP Installation and Administration Guide for more details on how to create them, where to install them, and how to use them.

CLONING

Once the target system is prepared, the cloning process starts with a backup. Only backups created with the SnapManager interface can be used for creating a clone. BRBACKUP backups cannot be used for clones. Creating a backup using the SnapManager interface is the first step to cloning a system.

The next step is to initiate the cloning process. To clone a database from an existing backup using the SnapManager GUI, right-click the backup and then select “Clone…” This will launch the clone wizard that will guide you through the steps to create a clone. Alternatively, you can choose to perform a new backup and clone from it by selecting a profile and then right-clicking “Clone…”

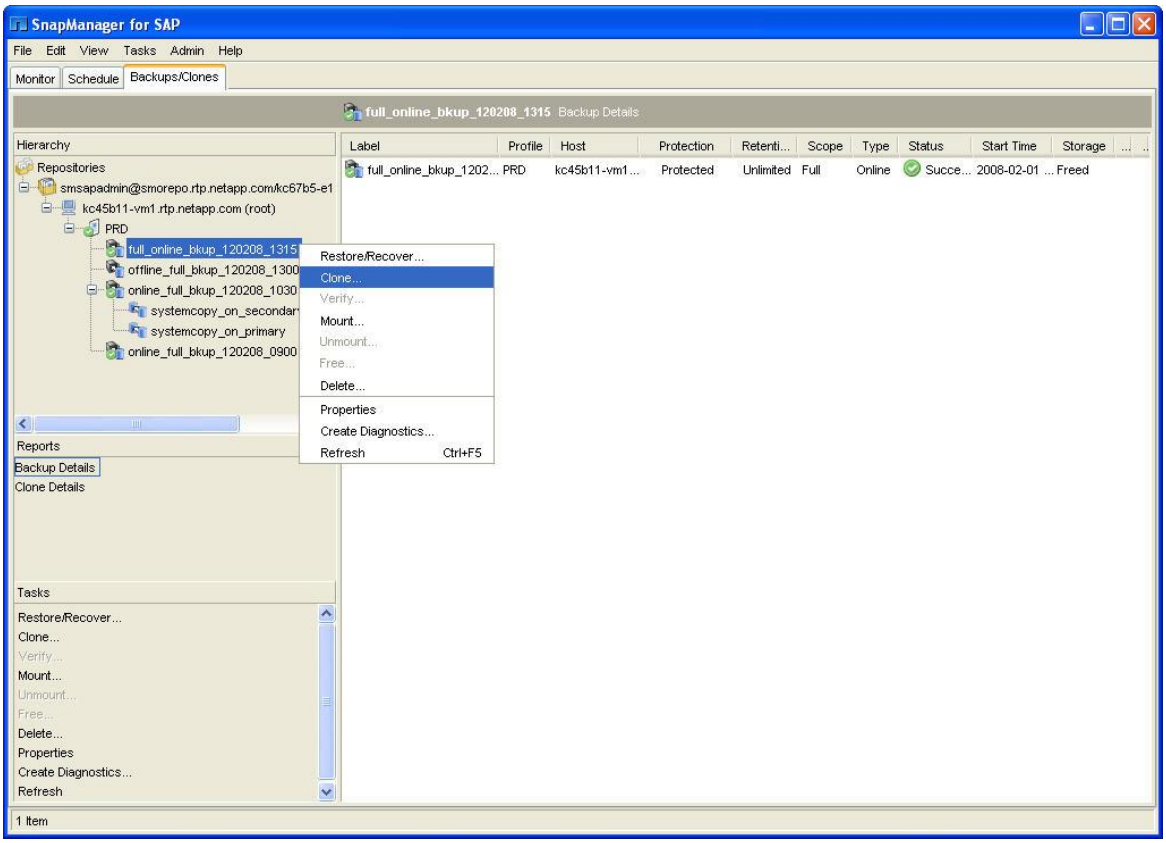


Figure 17) Cloning a backup.

The first time you perform a clone, you will need to proceed through the wizard making the necessary changes to the clone specification. Some changes that you might find useful that are not part of the default clone specification:

- Change the default mountpoints
- Create three copies of the control file in the SAP default locations
- Create the same number of redo log files as the SAP default with the same naming conventions and locations
- Change the log archive destination
- Change the Oracle home location
- Change the Oracle userid information

On the “Clone Specification” screen, there is the potential to save the clone specification in an XML format file, as shown in the following screenshot. It is recommended that this clone specification be saved after you perform any updates to allow for convenient repeating of a clone from one system to another. Appendix G contains the contents of a clone specification XML file. Saving this file can greatly reduce the amount of time necessary to create a system copy the next time it is required.

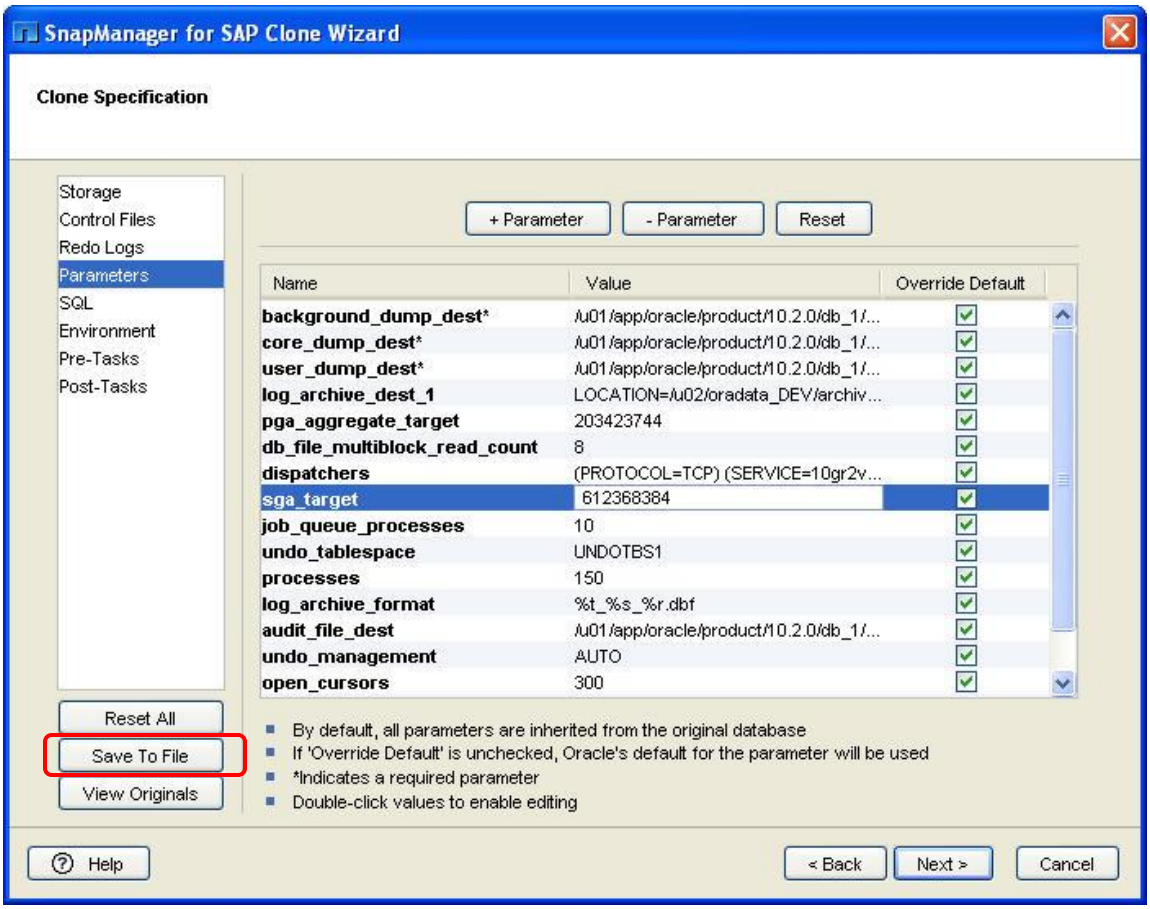


Figure 18) Saving the clone specification.

Table 20) Example commands to clone using SnapManager for SAP GUI and CLI.

Cloning	Example Commands
GUI	• To clone from an existing backup: Right-click the backup and select “Clone...” • To create a backup and clone from it in a single step: Right-click the profile and select “Clone…”
CLI	`smsap clone create -backup-label full_bkup_sales_feb_08 -newsid sls0208 -label sales0208_clone1 -profile targetdb1_prof1 -clonespec /home/oracle/smsap/sales_clonespec.xml`

CLONING POSTPROCESSING

After the clone has been created, before SAP can be started, it is necessary to complete the postprocessing steps. First, the `ops$connect` user has to be updated so that the local `sidadm` and `orasid` userids can connect to the database. This can be done with a script provided by SAP called `ORADBUSR.SQL`. This script is available as part of the SAP installation kit and also on the SAP Service Marketplace. The script is an attachment to note 50088 and can be downloaded directly from the SAP Service Marketplace.

AUTOMATED CLONING POSTPROCESSING

SnapManager 3.0 for SAP can now automate executing custom scripts before and after the clone creation process. SnapManager ships with sample postprocessing scripts in the `<SnapManager_install_directory>/plugins/examples/clone/create/post` directory that

automate the postprocessing tasks mentioned in the previous section. Refer to the section titled “Cloning databases and using custom plug-in scripts” in the SnapManager 3.0 for SAP Installation and Administration Guide for more details on how to create them, where to install them, and how to use them.

CLONING RAC DATABASES

SnapManager for SAP clones a RAC database to a non-RAC database and sets the Oracle parameter `cluster.database` to `false`. You can then change it to a RAC database manually. Detailed steps to perform this conversion are listed in Appendix F.

CLONING PROTECTED BACKUPS

SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. Using SnapManager 3.0 for SAP, database administrators can also clone the protected backups on the secondary storage system for development and test without affecting the primary storage system. This functionality is optional and is not available on Windows. Refer to the section titled “Cloning protected backups” in the SnapManager 3.0 for SAP Installation and Administration Guide for more details.

Note that if cloning a protected backup on secondary storage and a Snapshot copy in the backup happens to be the last Snapshot copy transferred to secondary for that respective qtree or volume, then the clone will fail. An error message appears describing why it failed. In this case, you might want to create another backup and wait for it to be transferred to secondary by Protection Manager during its regular transfer schedule. Alternatively, you might want to contact the storage administrator and ask for the backup to be transferred.

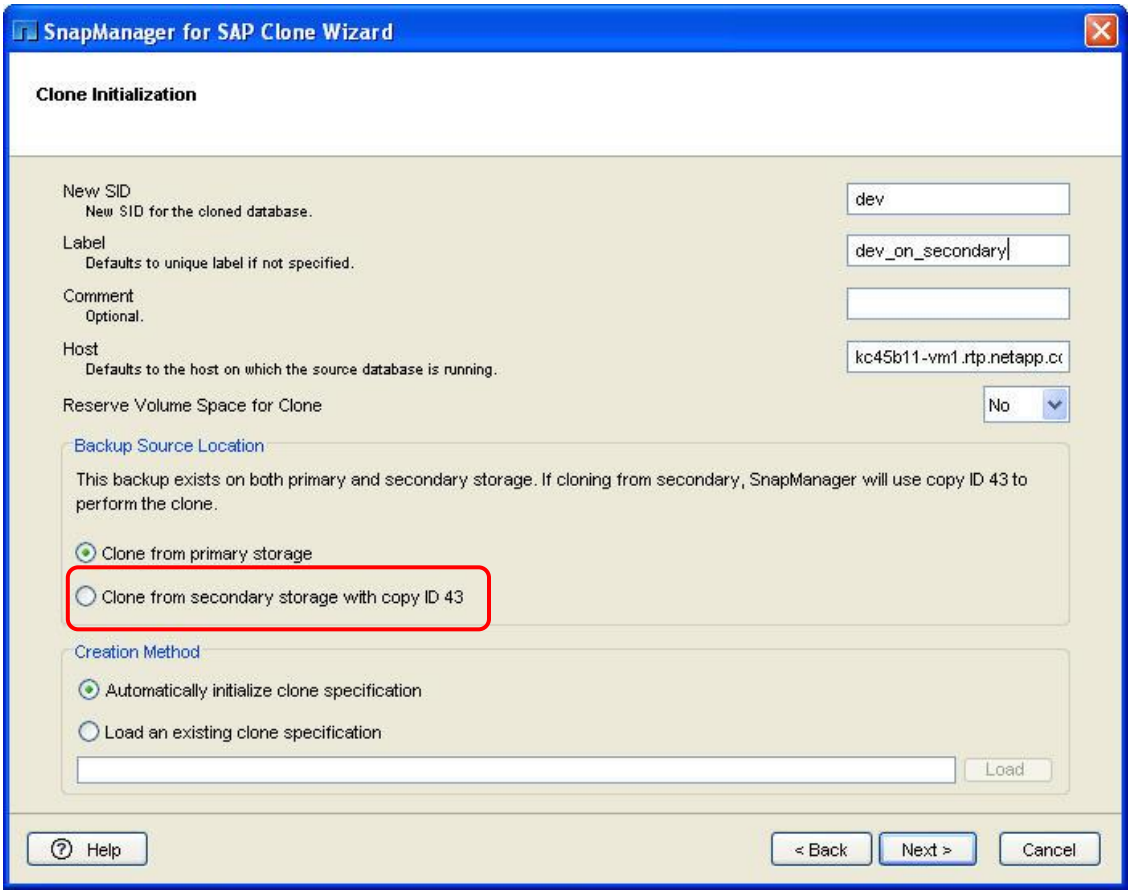


Figure 19) Cloning a protected backup on secondary.

CLONING POLICIES AND PRE/POST-TASKS

SnapManager can enforce custom policies during the cloning process. For example, a policy restricting the database SID as per your business rules can be created, and SnapManager will automatically verify the SID you specified in the clone request, based on the rules specified in the policy.

SnapManager 3.0 for SAP can also automate executing custom scripts before and after the clone creation process. This functionality can be used to mask production data, add a temporary tablespace, and so on in the clone database.

SnapManager ships with some sample scripts in the `<SnapManager_install_directory>/plugins/examples/clone/create` directory that can be leveraged to create custom scripts. Refer to the section titled “Cloning databases and using custom plug-in scripts” in the SnapManager 3.0 for SAP Installation and Administration Guide for more details on how to create them, where to install them, and how to use them.

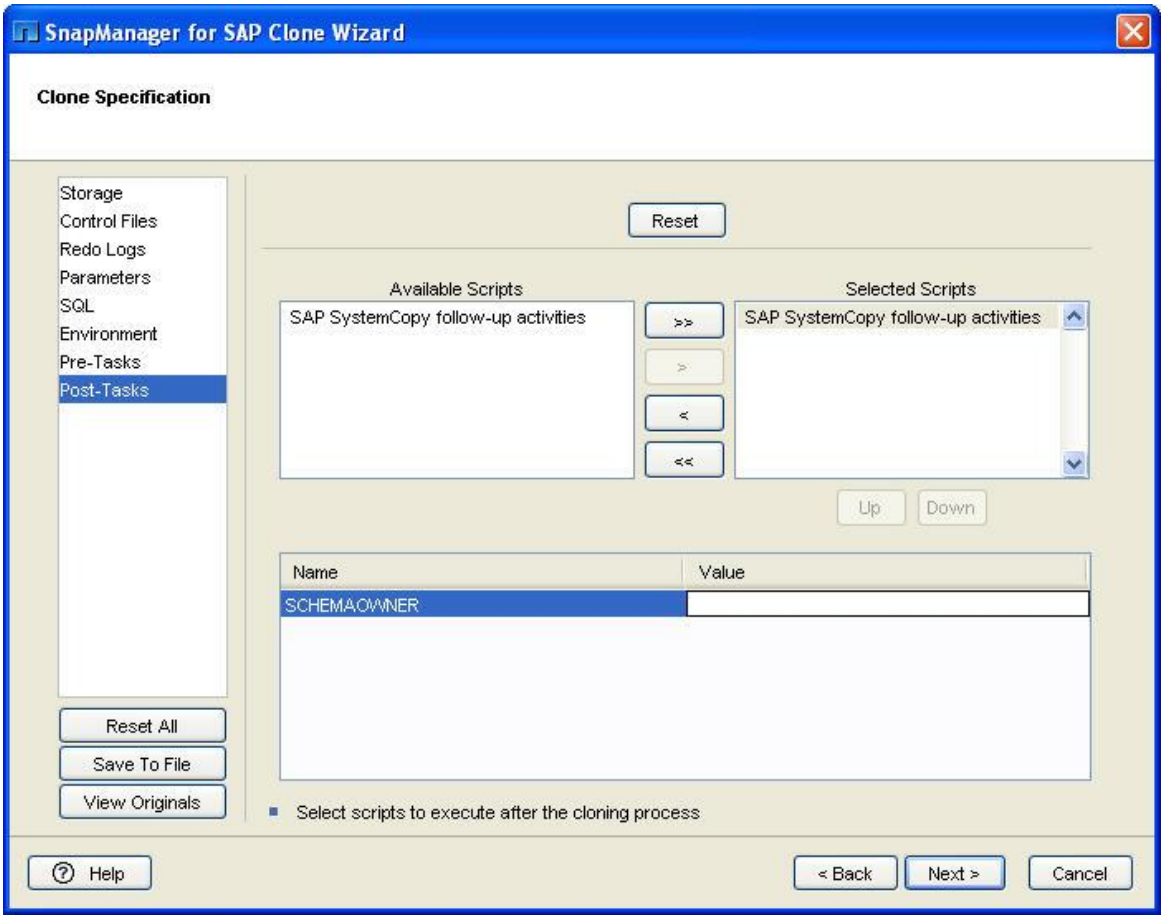


Figure 20) Clone post-tasks.

CLONING CLONES

SnapManager currently does not automatically create a new profile for database clones. If you would like to be able to back up and clone a clone database, then you should create a new profile for the clone database in SnapManager for SAP. Once you create a new profile for the clone database, you will be able to perform regular SnapManager operations such as backup and clone on the clone database.

CLONING TECHNOLOGIES

Prior versions of SnapManager for SAP leveraged FlexClone technology while cloning in NFS environments and LUN clone technology while cloning in SAN environments. In version 3.0, while cloning, SnapManager can leverage optionally FlexClone technology even in SAN environments if SnapDrive is configured to use FlexClone in SAN environments.

SnapDrive for UNIX can be configured to leverage FlexClone technology even in a SAN environment using the `san-clone-method` option in the `snapdrive.conf` file. The following table illustrates the various possible values that can be specified for the `san-clone-method` option in the `snapdrive.conf` file.

Table 21) `san-clone-method` values and their descriptions.

`snapdrive.conf` Variable and Value	Description
`san-clone-method=lunclone`	Create a LUN clone always.
`san-clone-method=unrestricted`	Create an "unrestricted" FlexClone volume or fail if FlexClone is not licensed.
`san-clone-method=optimal`	Create a "restricted" FlexClone volume if FlexClone is licensed on the storage system. Create a LUN clone otherwise.

If using SnapDrive for UNIX, NetApp recommends setting the value of the `san-clone-method` option in the `snapdrive.conf` file to `optimal`. Restart the SnapDrive daemon every time you modify the `snapdrive.conf` file for the changes to take effect.

SnapDrive for Windows will automatically leverage FlexClone technology if it is licensed, so no additional configuration is required.

FLEXCLONE COMPARED TO LUN CLONE

The following table compares FlexClone and LUN clone technologies.

Table 22) FlexClone versus LUN clone.

	FlexClone	LUN Clone
What is it?	A flexible volume clone, FlexClone is a writable point-in-time copy of a parent flexible volume.	A LUN clone is a point-in-time, writable copy of a LUN in a Snapshot copy.
Are changes made to the parent after the clone inherited?	Changes made to the parent flexible volume after the clone is created are not inherited by the FlexClone.	Changes made to the parent LUN after the clone is created are not reflected in the Snapshot copy.
Does it take up space?	Requires no additional disk space until changes are made to the clone or parent.	Requires no additional disk space until changes are made to the clone or parent.
Instantaneous?	Yes	Yes
Licensed?	Yes	No license is required.
Leads to busy Snapshot copies?	No	Yes

Although no separate license is required for creating LUN clones, NetApp recommends licensing FlexClone even in SAN environments (if using a protection policy that leverages volume SnapMirror) and configuring SnapDrive accordingly if required. Doing so will enable SnapManager to leverage FlexClone technology while cloning in both NFS and SAN environments. Also, future versions of SnapManager might provide additional features that will be applicable to only clones created using FlexClone technology.

The following table details the FlexClone licensing requirements on primary and secondary storage systems.

Table 23) FlexClone licensing requirements on primary and secondary.

Storage System	Required or Optional
On primary	• Required for NFS • Required for SAN if SnapDrive is configured to use FlexClone in SAN environments; optional otherwise

On secondary	• Required for NFS and if using data protection feature
	• Required for SAN only if all of the following conditions are met:
	  ○ Using data protection feature
	  ○ SnapDrive is configured to use FlexClone in SAN environments
	  ○ Using a protection policy that leverages volume SnapMirror (VSM)
	Note: In SAN environments if using a protection policy that leverages SnapVault or qtree SnapMirror (QSM), SnapDrive will always create LUN clones even if SnapDrive is configured to use FlexClone in SAN environments. In such cases, FlexClone need not be licensed on the secondary storage system.

SPLITTING CLONES

SnapManager for SAP leverages FlexClone and LUN clone technologies to provide quick, space-efficient cloning of Oracle Databases. NetApp Data ONTAP 7G provides the ability to split a FlexClone volume from its parent. Splitting a FlexClone volume creates a new fully independent FlexVol® volume.

SnapManager for SAP cannot currently split a clone database from its parent. In such cases you can split the clone manually outside of SnapManager for SAP using the Data ONTAP CLI. Since SnapManager is not aware of this manual split operation, if you delete the clone database from SnapManager, SnapManager will delete the clone database and all of its associated volumes even if those volumes have been split from the parent volumes.

To work around this, SnapManager 3.0 for SAP provides an undocumented "`detach`" option that will delete all the metadata of the clone database from the SnapManager repository without deleting the physical volumes associated with the clone database.

Table 24) Example command to detach a clone using SnapManager for SAP.

Detaching Clones	Example Commands
GUI	Not available from the GUI
CLI	`smsap clone detach -profile targetdb1_prof1 -label sales0208_clone1`

After using the `smsap clone detach` command you can create a new profile in SnapManager for the detached clone database. This way you can use SnapManager for SAP to back up the detached database and even clone it, if required.

DELETING CLONES

When the clone is no longer needed, stop the SAP instance, but leave the Oracle Database running. If the database is not running, SnapManager might not perform all the clone deletion steps correctly.

To delete a clone using the SnapManager GUI, right-click the clone and select "Delete…" This will launch the wizard that will guide you through the steps to delete a clone. To delete a backup that has been used to create clones, all the clones must be deleted first.

Table 25) Example commands to delete a clone using SnapManager for SAP.

Deleting Clones	Example Commands
GUI	Right-click the clone and select "Delete..."
CLI	`smsap clone delete -profile targetdb1_prof1 -label sales0208_clone1`

CHANGES IN VERSION 3.0 THAT AFFECT CLONING

- SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. Using SnapManager 3.0 for SAP, database administrators can also clone the protected backups on the secondary storage system for development and test without affecting the primary storage system. This functionality is optional and is not available on Windows.
- If temporary datafiles are in the same volume as regular datafiles, then while cloning, SnapManager 3.0 for SAP will create a new temporary datafile in the clone database for each corresponding datafile of the source database. If the temporary datafiles are not in the same volume as the regular datafiles in the source database, then SnapManager will not create any temporary datafiles in the clone, in which case you can specify the creation of a temporary datafile as a postclone SQL statement or script, and SnapManager will automatically create it. Previous versions never created temporary datafiles in the clone database.
- SnapManager can enforce custom policies during the cloning process. For example, a policy restricting the database SID as per your business rules can be created, and SnapManager will automatically verify the SID you specified in the clone request, based on the rules specified in the policy.
- SnapManager 3.0 for SAP can also automate executing custom scripts before and after the clone creation process. This functionality can be used to mask production data, add a temporary tablespace, and so on in the clone database.
- SnapManager 3.0 now leverages FlexClone technology even in SAN environments if SnapDrive is configured to use FlexClone in SAN environments. Prior versions of SnapManager only supported LUN clones in SAN environments.
- If creating a clone using the SnapManager GUI, the clone operation can now be run in the background by clicking the new “Background” button available in the backup wizard. This allows SAP administrators to execute multiple SnapManager operations in parallel.

BEST PRACTICES AND REQUIREMENTS FOR CLONING

- NetApp recommends replicating SnapManager backups of all critical databases to a secondary storage system by leveraging the policy-driven data protection feature in version 3.0. NetApp also recommends leveraging the protected backups on secondary storage for cloning. Cloning on secondary storage has the following benefits:
  - Provides complete isolation from production
  - Tests DR readiness every time you clone
  - Makes active use of secondary/DR storage for development, test, reporting, and so on
- Although no separate license is required for creating LUN clones, NetApp recommends licensing FlexClone even in SAN environments and configuring SnapDrive accordingly if required. Doing so will enable SnapManager to leverage FlexClone technology while cloning in both NFS and SAN environments. Also, future versions of SnapManager might provide additional features that will be applicable to only clones created using FlexClone technology.
- If cloning a protected backup on secondary storage and a Snapshot copy in the backup happens to be the last Snapshot copy transferred to secondary for that respective qtree or volume, then the clone will fail. An error message appears describing why it failed. In this case, you might want to create another backup and wait for it to be transferred to secondary by Protection Manager during its regular transfer schedule. Alternatively, you might want to contact the storage administrator and ask for the backup to be transferred.
- NetApp recommends adjusting the SGA and other resource-consuming parameters for the clone database, if possible. You can modify these parameters in the clone specification file before initiating clone creation.
- You can also use NetApp FlexShare® to adjust priority between the volume containing the system copy and the volume containing the production database. For more information about FlexShare, refer to the FlexShare Design and Implementation Guide.
- You must give the clone a new Oracle SID. Oracle does not permit you to run two databases with the same SID simultaneously on the same host. You can have a clone on a different host using the same SID. As a result, there is a new label option to designate a unique name for the clone. If you do not use this option, SnapManager creates a unique name for the clone that includes the SID, date, and time.

- If cloning using SnapManager's CLI, you must create a clone specification file for the database. SnapManager creates the clone based on the information in the original database, or you can create the file yourself. Note that directories specified for certain Oracle parameters (archive log destination, background, core, and user dump destinations) in the clone will be destroyed when the clone is deleted and should therefore only contain data for the cloned database.
- NetApp recommends using the `.xml` extension for the clone specification file to enable appropriate editing features.
- If using `TNSNAMES`, add the details of the newly created clone database to the `TNSNAMES` file on the client machines that need to access it.

6 DISASTER RECOVERY

SnapManager for SAP backs up Oracle Databases by creating Snapshot copies on the original primary NetApp storage systems of the databases. This allows fast backups and restores but does not provide security in the event of failure on the primary storage system. While such failures are rare, they can occur, so it is a best practice to always maintain backups of your databases in a secondary location or media.

You can back up data to a secondary location in several ways:

- SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager 3.7.1. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. SnapManager 3.0 for SAP also empowers the database administrator to automatically restore such protected backups from the secondary storage system back to the primary storage system. This functionality is optional and is not available on Windows. Refer to Appendix B for instructions to configure and enable policy-driven data protection.
- If Protection Manager is not available, then a scripted solution that leverages NetApp SnapMirror or SnapVault can be implemented. SnapMirror can replicate data at high speeds over a LAN or a WAN and provides the highest possible data availability and fastest recovery for mission-critical applications. SnapVault is a disk-based backup solution. Both of these solutions only replicate the new and changed blocks incrementally over the network and, by replicating only a subset of the entire storage data, SnapMirror and SnapVault significantly reduce network bandwidth requirements.
- Backup to tape is a common method for creating backups for disaster recovery purposes. Appendix K of this report has a sample script that will identify the last successful backup created by SnapManager and generate the necessary commands to copy the Snapshot copies associated with that backup to a secondary location such as tape.
- Another method for creating a tape backup is to use a dedicated media server to back up the database by attaching a clone of the database. SnapManager for SAP can create a clone from a SnapManager backup onto the media server. The media server will then run the backup software and make a tape copy of the database for disaster recovery purposes. This provides the dual benefits of testing the backup and also having a tape backup without affecting the original database. Some additional cost is related to having a dedicated media server.

6.1 SNAPVAULT AND SNAPMIRROR INTEGRATION

SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager 3.7.1. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator. SnapManager 3.0 for SAP also empowers the database administrator to automatically restore such protected backups from the secondary storage system back to the primary storage system. Using SnapManager 3.0 for SAP, database administrators can also clone the protected backups on the secondary storage system for development and test without affecting the primary storage system. This functionality is optional and is not available on Windows. Refer to Appendix B for instructions to configure and enable policy-driven data protection.

In version 1.1 of SnapManager for SAP, due to the lack of integration with Protection Manager, SnapManager provided the following features to aid in scripting a solution that leverages NetApp SnapVault and SnapMirror to replicate the SnapManager backups from primary to secondary storage:

- Freeing backups
- Restoring from an alternate location

If Protection Manager is not available because of which the data protection features of SnapManager cannot be leveraged, then the following steps provide an overview of how these features can be leveraged to provide a semiscripted disaster recovery solution:

1. Create a backup using the SnapManager GUI or CLI on the primary NetApp storage system.
2. Parse the output of the "`smsap backup show`" command to identify the Snapshot copies associated with the backup.
3. Protect the backup by copying those associated Snapshot copies to a secondary NetApp storage system using either SnapVault or SnapMirror.
4. Optionally, free up the SnapManager backup on the primary storage system using the SnapManager GUI or CLI. This deletes the Snapshot copies associated with the backup on the primary storage, which frees up space on the primary storage system and also allows more frequent backups without reaching the Snapshot copy limit. Freeing up a backup also retains the backup metadata in the SnapManager repository, allowing a restore from an alternate location using SnapManager whenever required.
5. To restore an Oracle Database that has a protected backup in an alternate location, first restore the required datafiles from SnapVault or SnapMirror or any other media to a location on the primary or secondary storage system.
6. Mount the restored location on the database host.
7. Parse the output of the "`smsap backup show`" command to identify the primary storage volumes and the corresponding Snapshot copies.
8. Using the above data, generate a restore specification XML file with the original and alternate locations' mappings and save the file in a location accessible from the SnapManager GUI or CLI. Refer to Appendix H for an example of a restore specification file.
9. Use the SnapManager GUI or CLI to perform restore and recovery from the alternate location by specifying the location of the restore spec XML file.

Refer to Appendix I for an example of how SnapManager for SAP can be integrated with SnapVault to provide a semiscripted disaster recovery solution.

BEST PRACTICES AND REQUIREMENTS FOR DISASTER RECOVERY

- NetApp recommends replicating SnapManager backups of all critical databases to a secondary storage system by leveraging the policy-driven data protection feature in version 3.0. Refer to Appendix B for instructions to configure and enable policy-driven data protection.
- If using a scripted solution to replicate backups to a secondary storage system, NetApp recommends freeing up backups on primary only after they have been copied to secondary storage.
- If using SnapVault, NetApp recommends setting the number of backups to be retained to a large number so that you can vault a backup before the backup is pruned.

7 MAINTENANCE

It is important to properly manage the backups and clones created by SnapManager for SAP.

BACKUPS

While the number of Snapshot copies permitted on a volume is large, it is still necessary to periodically delete old backups. This is especially important for backups created for cloning databases. When such backups are retained for long durations, a large amount of changed data might occupy space in the source volume.

Backups are deleted automatically by SnapManager based on the backup retention number specified in the profile of the database. You can change this value in the profile at any time. The number of backups to be retained usually depends on the business requirements and the database usage. For example, you might

keep a single backup of a sandbox database, one week's backups for a development database, and a few days' backups for a production system. A backup that is being used for a clone cannot be deleted until the clone is deleted.

Backups that have been replicated to a secondary storage system by leveraging the policy-driven data protection feature of SnapManager 3.0 for SAP are automatically freed on the:

- Primary based on the retention policy specified by the SAP administrator in SnapManager for SAP
- Secondary based on the retention policy specified by the backup or storage admin in Protection Manager

If Protection Manager is not available, backups that have been protected by copying the Snapshot copies to a secondary location using other software or using a scripted solution that leverages SnapVault or SnapMirror can be freed on primary. Freeing backups frees up space on the primary storage, since the associated Snapshot copies are deleted, enables more frequent backups, and helps avoid reaching the Snapshot copy limit on the volumes.

SNAPMANAGER SERVER LOGS

SnapManager for SAP is by default installed under:

- `/opt/NetApp/smsap`

The SnapManager server logs are stored under:

- `/var/log/smsap` in UNIX.

SnapManager automatically manages the server log files based on the user-definable values of `log.max_log_files`, `log.max_log_file_size` and `log.max_rolling_operation_factory_logs` parameters in the SnapManager configuration file `smsap.config`. Therefore the directory in which the server logs are written will not grow without bounds. The SnapManager configuration file `smsap.config` is located under the `properties` directory under the directory where SnapManager is installed. For example, in Linux the path of the file would be `/opt/NetApp/smsap/properties/smsap.config`.

BEST PRACTICES AND REQUIREMENTS FOR MAINTENANCE

- NetApp recommends freeing up backups only after they have been copied to secondary storage.
- NetApp recommends deleting backups created by SnapManager using only the SnapManager GUI or CLI. Deleting Snapshot copies created by SnapManager from SnapDrive or Data ONTAP will cause corruption of the SnapManager repository.
- If you use SnapVault, NetApp recommends setting the number of backups to be retained to a large number so that you can vault a backup before the backup is pruned.
- NetApp recommends toning down the SGA and other resource-consuming parameters for the clone database, if possible. You can modify these parameters in the clone specification file before initiating clone creation.

8 CONCLUSION

SnapManager 3.0 for SAP provides a rich feature set that allows IT organizations to take advantage of fast, space-efficient, disk-based backups; rapid, granular restore and recovery; and quick, space-efficient cloning. The recommendations and examples in this report will help get the most out of SnapManager for SAP deployments on NetApp storage. For more information about any of the solutions or products covered in this report, contact NetApp.

APPENDIX A: SNAPMANAGER INSTALLATION AND CONFIGURATION QUICK START GUIDE

The following example commands illustrate how to quickly install and configure SnapManager for SAP. Refer to Appendix D for sample commands to back up, restore, recover, and clone databases using SnapManager.

Prerequisite Tasks	Example Tasks
Verify that all the prerequisites mentioned in the SnapManager 3.0 for SAP Installation and Administration Guide have been met	• Check the SnapManager Interoperability Matrix to make sure that all components in your environment are supported. • Check if all required licenses are enabled.
Repository Host Setup	**Example Commands**
Identify an existing Oracle Database and listener port for the SnapManager repository	• Identify the Oracle SID of the database and make sure the database is open: **# su – oracle** **[oracle@repo_host1** **~]# cat /etc/oratab** • Identify the Oracle listener port for this database and make sure that the listener has been started: **[oracle@repo_host1~]$ LSNRCTL> status**
Create a tablespace to be used by the SnapManager repository in the above database	• Create a new tablespace for the SnapManager repository: **SQL> create tablespace "smsap" datafile '/u01/oradata/smsaprepo/datafile/smsap01.dbf' size 100m autoextend on maxsize 1000M;** • SnapManager requires a minimum 4K block size for the tablespace into which it is installed. Check the block size for the "smsap" tablespace using: **SQL> select tablespace_name, block_size from dba_tablespaces where tablespace_name = 'SMSAP';** TABLESPACE_NAME BLOCK_SIZE ------------------------------ ---------- SMSAP 8192
Create an Oracle user who will own the SnapManager repository in the above database	**SQL> create user smsapadmin identified by adminpw1 temporary tablespace temp default tablespace smsap quota unlimited on smsap;**
Grant only the "connect" and "resource" roles to the above database user	**SQL> grant connect,resource to smsapadmin;**

Target Database Host Setup	Example Commands

Install, configure, and verify SnapDrive on all the target database hosts	• Download the appropriate SnapDrive software file for your host platform from the NetApp NOW site.
	• SnapDrive should be installed on every host that has one or more databases that will be managed by SnapManager.
	• Log in as root and install the SnapDrive software.
	• SnapDrive 4.0.1 for UNIX defaults to use the HTTPS connection. If upgrading to or installing SnapDrive 4.1 for UNIX and if you would like SnapDrive for UNIX to use the HTTP protocol to communicate to the storage system, then:
	- Edit the `snapdrive.conf` file and set
	`use-https-to-filer=off`
	Or if your storage system is using the HTTP protocol and you would like SnapDrive for UNIX to use the HTTPS protocol instead, then:
	- Enable the HTTPS connection on your storage system
	• Restart the SnapDrive daemon every time you modify the `snapdrive.conf` file for the changes to take effect.
	`[root@tardb_host1 snapdrive]# snapdrived restart`
	• Configure SnapDrive and specify which OS user will be used to access the NetApp storage system used by the target databases:
	`[root@tardb_host1 snapdrive]# snapdrive config set root my_netapp_storage_system1`
	• Verify that the above configuration succeeded:
	`[root@tardb_host1 snapdrive]# snapdrive config list`
	`user name filer name`
	`------------------------`
	`root my_netapp_storage_system1`

Install the SnapManager for SAP software on all the target database hosts	• Download the appropriate SnapManager 3.0 for SAP software file for your host platform from the NetApp NOW site. • SnapManager software should be installed on every host that has one or more databases that will be managed by SnapManager. • Log in as root and install the SnapManager for SAP software. • SnapManager 3.0 for SAP now also supports OS-authenticated database connections for RAC databases. The SnapManager server must be installed and running on each node in the RAC cluster for a RAC database that is using the OS-authenticated connection mode. This is a new requirement of SnapManager 3.0 for SAP.
Start the SnapManager server on the target database host	**`[root@tardb_host1 SMSAP]# smsap_server start`** `SMSAP-17100 [INFO ]: SnapManager Server started on secure port` **`27314`** `with PID 8235.` **Note** the port number that the server is started on. This port is used to launch the SnapManager GUI from a Web browser. The default port is 27314.

Verify the SnapManager installation	**[root@tardb_host1 SMSAP]# smsap system verify –verbose** SMSAP-13505 [INFO]: Snapdrive verify passed. SMSAP-13037 [INFO]: Successfully completed operation: System Verify SMSAP-13049 [INFO]: Elapsed Time: 0:00:00.559 Operation Id [N4f4e910004b36cfecee74c710de02e44] succeeded.
If using the database-authentication connection mode you can either use an existing database user with the `sysdba` role (such as `sys`) or create a new database user and grant the `sysdba` role if you are planning to use a new database user: • Create an Oracle user with the sysdba role for the target database that SnapManager will manage	**SQL> create user *smsap_oper* identified by *operpw1;*** To manage a database, SnapManager requires that an Oracle user with the sysdba role connect to that database and perform database operations: **SQL> grant sysdba to *smsap_oper;***
Launching the SnapManager GUI or CLI	**Example Commands**
CLI	• The SnapManager CLI can be accessed from any host where the SnapManager server has been installed. • You could use any target database host where the SnapManager software is already installed to access the SnapManager CLI, or you could use a dedicated host just to issue SnapManager commands using the CLI. The SnapManager software will still need to be installed on this dedicated host to access the CLI. • The SnapManager commands all start with `smsap` (SnapManager for SAP).
GUI	• The SnapManager GUI is launched from a Web browser on any host running an operating system supported by SnapManager for SAP, using: `https://smsap-server.domain.com:port` where: - `smsap-server` is the name of any host where the SnapManager server was installed and started. - `domain.com` is the domain of SnapManager server host. - `port` is the port number that the SnapManager server was started on. The default port is 27314.

<table>
<tr><th>Credentials, Repository, and Profiles Setup</th><th colspan="2">Example Commands</th></tr>
<tr><td>Set repository access credentials for every OS user who will use SnapManager</td><td colspan="2">• Log in as the OS user:
smsap credential set -repository -host repo_host1 -dbname smsaprepo -port 1524 -login -username smsapadmin -password adminpw1
Enter password for database connection smsapdmin@repo_host1:1524/smsaprepo: **********
Note: The smsap credential set command can now be used without the -password option because it will prompt for the password. Although the -password option is still available in version 3.0 for backward compatibility, NetApp recommends using the smsap credential set command without the -password option.
• Execute the above command for all target database hosts this OS user will access using SnapManager.</td></tr>
<tr><td rowspan="2">Create the SnapManager repository using SnapManager</td><td>SMSAP GUI</td><td>Operations -> Repository -> Create New Repository…</td></tr>
<tr><td>SMSAP CLI</td><td>smsap repository create -repository -dbname smsaprepo -host repo_host1 -port 1524 -login –username smsapadmin</td></tr>
<tr><td rowspan="2">Create a profile in SnapManager for every target database that will be managed by SnapManager</td><td>SMSAP GUI</td><td>Operations -> Repository -> Create Profile…</td></tr>
<tr><td>SMSAP CLI</td><td>smsap profile create -profile targetdb1_prof1 -profile-password tardbpw1 -repository -dbname smsaprepo -login -username smsapadmin -host repo_host1 -port 1524 -database -dbname tardb1 -login -username smsap_oper -password operpw1 -host tardb_host1 -port 1521 –sid tardb1 -osaccount oracle -osgroup dba -retain 100 -verbose</td></tr>
</table>

APPENDIX B: CONFIGURING AND ENABLING POLICY-DRIVEN DATA PROTECTION IN SNAPMANAGER 3.0 FOR SAP

ASSUMPTIONS

- Operations Manager and Protection Manager are licensed and installed.
- Operations Manager is installed on a dedicated host.

CONFIGURING OPERATIONS MANAGER AND SNAPDRIVE FOR UNIX

The following example commands illustrate how to configure SnapManager, SnapDrive, Protection Manager, and Operations Manager to enable policy-driven data protection capabilities in SnapManager for SAP.

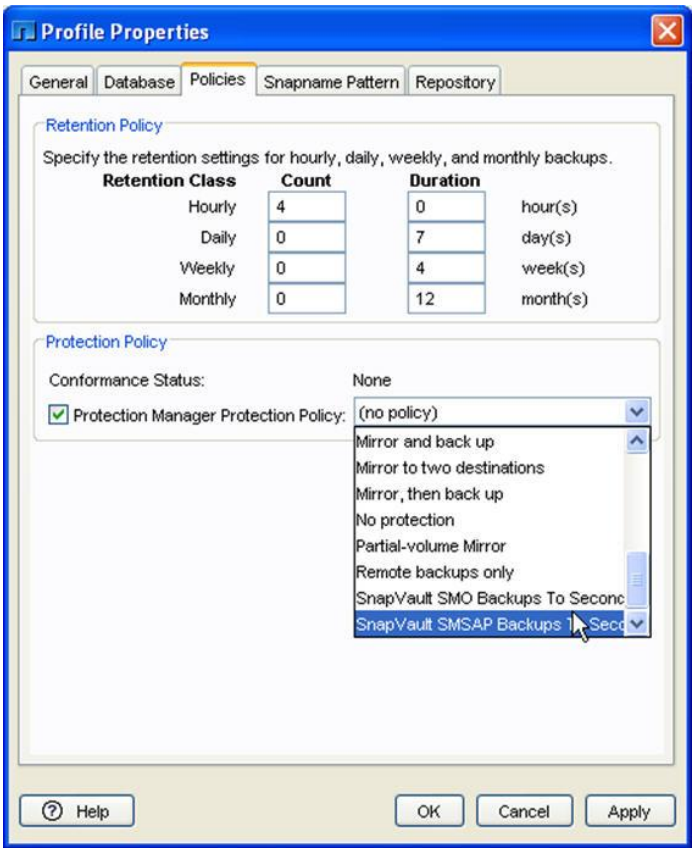
Configuring Operations Manager	Example Commands
Forcefully update DFM with any changes made directly on the storage system used by the target database	**dfm_host> dfm host discover my_netapp_storage_system1**
Create a new OS user called “sd-admin” on the host of the DFM server and set the password for the “sd-admin” user so you know what to tell SnapDrive it is	• Create a new OS user called “sd-admin” on the host of the DFM server. • Set the password of this user so you know what to tell SnapDrive it is.
Add the “sd-admin” OS user to the DFM server administrator list	**dfm_host> dfm user add sd-admin** Added administrator sd-admin.
Create a new role in DFM called “sd-admin-role”	**dfm_host> dfm role create sd-admin-role** Created role sd-admin-role.
Add the DFM.Core.AccessCheck global capability to the new “sd-admin-role” role	**dfm_host> dfm role add sd-admin-role DFM.Core.AccessCheck Global** Added 1 capability to role sd-admin-role.
Add the DFM.Database.Write global capability to the new “sd-admin-role” role Perform this step only when automatic refresh of storage system entities is required on DFM	**dfm_host> dfm role add sd-admin-role DFM.Database.Write Global** Added 1 capability to role sd-admin-role.
Add the “sd-admin-role” role to the “sd-admin” user	**dfm_host> dfm user role set sd-admin sd-admin-role** Set 1 role for administrator sd-admin.

Add the “GlobalDataProtection,” “GlobalRestore” roles to the “sd-admin” user	**dfm_host> dfm user role add sd-admin GlobalDataProtection GlobalRestore** Added 2 roles to administrator sd-admin.
Make DFM aware of the storage system used by the target database	**dfm_host> dfm host set my_netapp_storage_system1 hostLogin=root hostPassword="<password>"**
Using DFM, create a new role called “sd-admin-role” on the storage system used by the target database	**dfm_host> dfm host role create -h my_netapp_storage_system1 -c "api-*,login-*" sd-admin-role** Created role sd-admin-role(9) on my_netapp_storage_system1 (88).
Using DFM, create a new group called “sd-admin-group” on the storage system used by the target database and assign the “sd-admin-role” role to this group	**dfm_host> dfm host usergroup create -h my_netapp_storage_system1 -r sd-admin-role sd-admin-group** Created usergroup sd-admin-group(9) on my_netapp_storage_system1(88).
Using DFM, create a new user called “sd-<hostname>” on the storage system used by the target database and assign the “sd-admin-role” role and “sd-admin-group” group to this user; the “<hostname>” in “sd-<hostname>” is the name of the target database host and so in this example is “sd-tardb_host1”	**dfm_host> dfm host user create -h my_netapp_storage_system1 -r sd-admin-role -p <password> -g sd-admin-group sd-tardb_host1** Created local user sd-tardb_host1(5) on my_netapp_storage_system1(88).
Grant the SnapDrive permissions to the “sd-admin-role” role	**dfm_host> dfm role add sd-admin-role SD.Config.Read Global** **dfm_host> dfm role add sd-admin-role SD.Config.Write Global** **dfm_host> dfm role add sd-admin-role SD.Config.Delete Global** **dfm_host> dfm role add sd-admin-role SD.Storage.Read Global** **dfm_host> dfm role add sd-admin-role SD.Storage.Write Global** **dfm_host> dfm role add sd-admin-role SD.Storage.Delete Global** **dfm_host> dfm role add sd-admin-role SD.Snapshot.Read Global** **dfm_host> dfm role add sd-admin-role SD.Snapshot.Write Global** **dfm_host> dfm role add sd-admin-role SD.Snapshot.Delete Global** **dfm_host> dfm role add sd-admin-role SD.Snapshot.Restore Global** **dfm_host> dfm role add sd-admin-role SD.Snapshot.Clone Global**

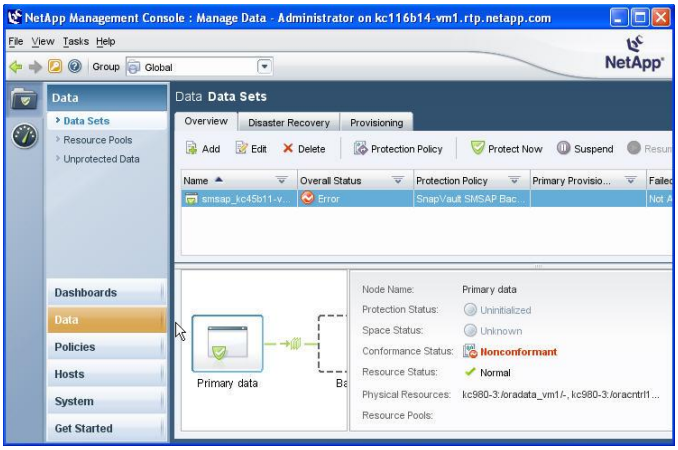
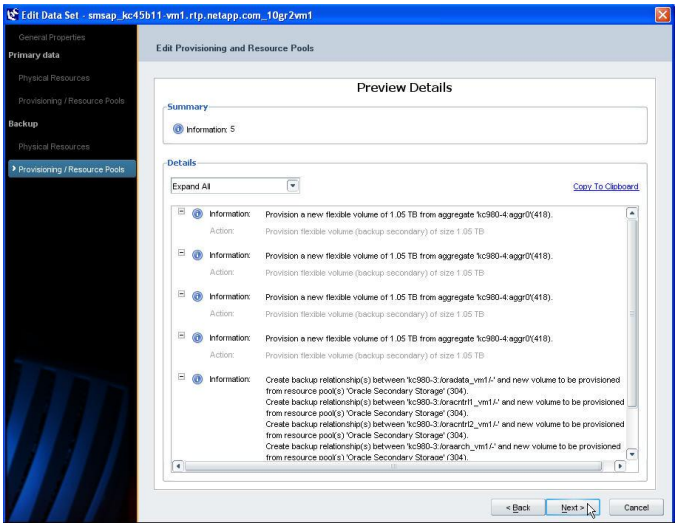
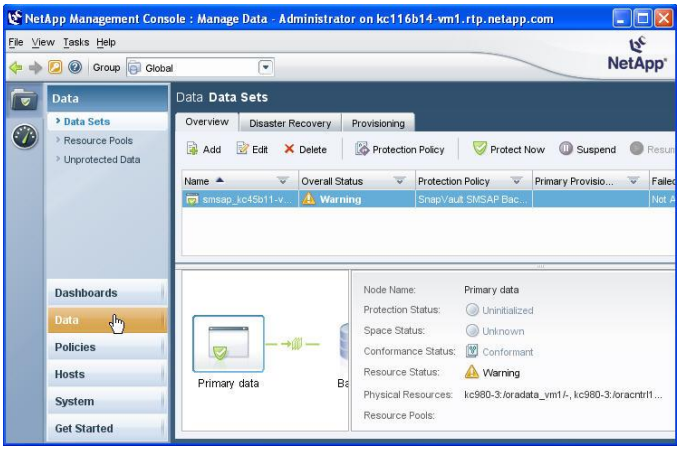
Add the target database host's root user to the list of administrators and assign the "sd-admin-role"	**dfm_host> dfm user add -r sd-admin-role "tardb_host1\root"** Warning: TARDB1_HOST1\root does not exist in the administrator database(s), so login is disabled for this administrator. Added administrator TARDB1_HOST1\root. Added 1 role to administrator TARDB1_HOST1\root.
Configuring SnapDrive	Example Commands
Register the credentials of the "sd-admin" user created on the DFM server host with SnapDrive	**[root@tardb_host1 snapdrive]# snapdrive config set -dfm sd-admin dfm_host** Password for sd-admin: netapp Retype password: netapp
Register the credentials of the "sd-<hostname>" user created on the storage system used by the target database with SnapDrive	**[root@tardb_host1 snapdrive]# snapdrive config set sd-tardb_host1 my_netapp_storage_system1** Password for sd- tardb_host1: netapp Retype password: netapp
Verify the above steps	**[root@tardb_host1 snapdrive]# snapdrive config list** username appliance name appliance type ---------- sd-tardb_host1 my_netapp_storage_system1 StorageSystem sd-admin dfm_host DFM

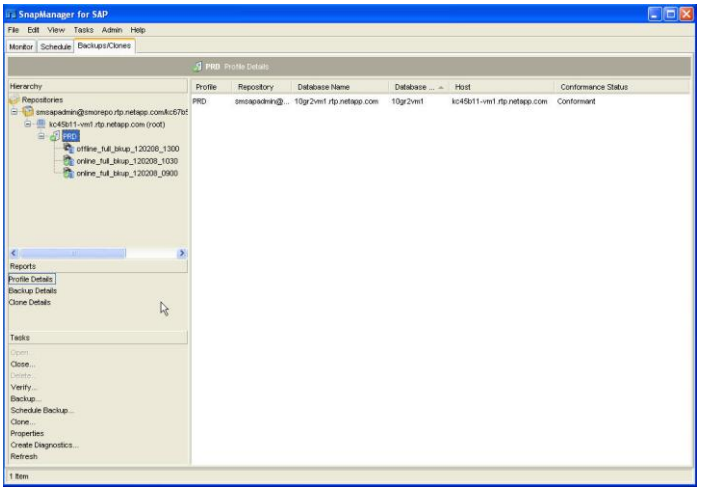
ENABLING DATA PROTECTION IN A SNAPMANAGER PROFILE

Once you complete the above steps to configure Operations Manager and SnapDrive for UNIX, the following have to be done for each SnapManager profile to enable data protection.

Specifying a Retention Policy and Enabling Data Protection in a SnapManager Profile	Examples
• In SnapManager 3.0 for SAP, right-click the profile whose backups you would like to have replicated to secondary. • Click the “Policies” tab. • Specify a “Retention Policy” based on your business requirements for the local backups created by SnapManager on the primary storage system. • Check the “Protection Manager Protection Policy” checkbox, which will immediately populate the list of policies from Protection Manager. • Select one of the policies as advised by your storage or backup administrator; this could be a built-in Protection Manager policy or a custom policy created by the storage or backup administrator for use by SnapManager. • Click the “OK” button. • SnapManager will automatically create a data set for that particular database and register it in Protection Manager. • Contact your storage or backup administrator and request your admin to edit the unprotected data set in Protection Manager and assign a resource pool to it. The “Conformance Status” of the profile is “Nonconformant” until then.	 The protection policy selected in the above screenshot is a custom policy that was created in Protection Manager.

ASSIGNING A RESOURCE POOL IN PROTECTION MANAGER

Assigning a Resource Pool to the Nonconformant Data Set in Protection Manager	Examples
• Using Protection Manager, the storage or backup administrator notices that the data set is unprotected and nonconformant. • To make the data set conformant, the storage or backup administrator edits (select the data set and click the "Edit" button) the data set and assigns a resource pool or secondary storage based on the protection policy selected by the database administrator in SnapManager. • Protection Manager automates provisioning new volumes on secondary and creates the necessary SnapVault or SnapMirror relationship based on the policy specified. The SnapManager backups will be replicated to secondary automatically based on the replication schedule specified by the storage or backup administrator in the Protection Manager policy.	  

Checking Conformance Status for a SnapManager Profile That Has Data Protection Enabled	Examples
• Once the data set is conformant in Protection Manager, the database administrator using SnapManager can also see that the “Conformance Status” is now “Conformant.”	

APPENDIX C: ENABLING RBAC IN SNAPMANAGER 3.0 FOR SAP

ASSUMPTIONS

- Operations Manager is licensed and installed on a dedicated host.

The following example commands illustrate how to configure SnapManager, SnapDrive, and Operations Manager to enable RBAC in SnapManager for SAP.

Configuring Operations Manager and SnapDrive	Example Commands
Follow all the steps detailed in the section titled "Configuring Operations Manager and SnapDrive for UNIX" in Appendix B	Follow all the steps detailed in the section titled "Configuring Operations Manager and SnapDrive for UNIX" in Appendix B.
Configuring SnapDrive	**Example Commands**
Edit the SnapDrive configuration file "`snapdrive.conf`" and set the "`rbac-method`" parameter to "`dfm`"	Edit the `snapdrive.conf` file and set: `rbac-method="dfm"  # Role Based Access Control(RBAC) methods: native or dfm`
Restart the SnapDrive daemon as root	**`[root@tardb_host1 snapdrive]# snapdrived restart`**

APPENDIX D: BACKUP, RESTORE, RECOVERY, AND CLONING QUICK START GUIDE

Creating Backups	Example Commands
GUI	Right-click the profile of that database and then select “Backup…”
CLI	To create a full online backup that is exempt from being deleted by the backup retention policy and verifies the backup: **smsap backup create –online –full –profile targetdb1_prof1 –label full_bkup_sales_may_08 –verify –retain –unlimited**
Freeing Backups	**Example Commands**
GUI	Right-click the backup and select “Free...”
CLI	**smsap backup free -profile targetdb1_prof1 -label full_backup_sales_apr_08**
Deleting Backups	**Example Commands**
GUI	Right-click the backup and select “Delete...”
CLI	**smsap backup delete –profile targetdb1_prof1 -label full_backup_sales_jan_07**
Restore and Recovery	**Example Commands**
GUI	Right-click the backup and select “Restore/Recover….” Refer to Appendix C and Appendix E for more sample restore and recovery scenarios using SnapManager for SAP.
CLI	To restore an entire backup along with the control files and recover until the last transaction: • First preview the restore operation **smsap backup restore –profile targetdb1_prof1 –label full_bkup_sales_feb_08 –complete –controlfiles –recover –alllogs –preview –verbose** • If satisfied with the preview results, then initiate the actual restore operation **smsap backup restore –profile targetdb1_prof1 –label full_bkup_sales_feb_08 –complete –controlfiles –recover –alllogs –verbose** Refer to Appendix E for more sample restore and recovery scenarios using SnapManager for SAP.
Cloning	**Example Commands**
GUI	• To clone from an existing backup: Right-click the backup and select “Clone...” • To create a backup and clone from it in a single step: Right-click the profile and select “Clone…”

CLI	`smsap clone create -backup-label full_bkup_sales_feb_08 -newsid sls0208 -label sales0208_clone1 -profile targetdb1_prof1 -clonespec ./smsap/sales_clonespec.xml`
Deleting Clones	**Example Commands**
GUI	Right-click the clone and select “Delete...”
CLI	`smsap clone delete -profile targetdb1_prof1 -label sales0208_clone1`

APPENDIX E: BASIC DATABASE RESTORE AND RECOVERY SCENARIOS

Some of the most common database restore and recovery scenarios using SnapManager for SAP are presented here.

ASSUMPTIONS

All scenarios described here assume that a full online backup (`full_bkup_sales_jun_08`) of the database has been previously created using SnapManager for SAP.

SCENARIO 1: RESTORE A WHOLE DATABASE WITHOUT CONTROL FILES AND RECOVER USING ALL AVAILABLE LOGS

In this scenario, the current control files exist, but all the datafiles are damaged or lost. Restore and recover the database from an existing full online backup using all available logs.

SnapManager Interface	Example Commands
GUI	• Right-click the backup and select “Restore/Recover…” to launch the restore and recovery wizard. • In the “Restore Configuration Information” screen of the wizard, select the “Complete Datafile/Tablespace Restore without Control Files” option and click the “Next” button. • In the “Recovery Configuration Information” screen of the wizard, select the “All Logs” option and click the “Next” button. • Navigate through the next few screens in the wizard appropriately.
CLI	**`smsap backup restore -profile targetdb1_prof1 -label full_bkup_sales_jun_08 -complete -recover -alllogs -verbose`**

SCENARIO 2: RESTORE A WHOLE DATABASE WITHOUT CONTROL FILES AND RECOVER TO A PARTICULAR SCN

In this scenario, the current control files exist, but all the datafiles are damaged or lost or a logical error occurred after a particular SCN. Restore and recover the database from an existing full online backup to a point immediately before that SCN.

SnapManager Interface	Example Commands
GUI	• Right-click the backup and select “Restore/Recover…” to launch the restore and recovery wizard. • In the “Restore Configuration Information” screen of the wizard, select the “Complete Datafile/Tablespace Restore without Control Files” option and click the “Next” button. • In the “Recovery Configuration Information” screen of the wizard, select the “SCN” option and specify the SCN number (`3794392`) in the text box and then click the “Next” button. • Navigate through the next few screens in the wizard appropriately.
CLI	**`smsap backup restore -profile targetdb1_prof1 -label full_bkup_sales_jun_08 -complete -recover -until 3794392 -verbose`**

SCENARIO 3: RESTORE A WHOLE DATABASE WITHOUT CONTROL FILES AND RECOVER UP TO A DATE AND TIME

In this scenario, the current control files exist, but all the datafiles are damaged or lost or a logical error occurred after a specific time. Restore and recover the database from an existing full online backup up to a date and time immediately before the point of failure.

SnapManager Interface	Example Commands
GUI	• Right-click the backup and select “Restore/Recover…” to launch the restore and recovery wizard. • In the “Restore Configuration Information” screen of the wizard, select the “Complete Datafile/Tablespace Restore without Control Files” option and click the “Next” button. • In the “Recovery Configuration Information” screen of the wizard, select the “Date” option and specify the date and time in the date field and then click the “Next” button. • Navigate through the next few screens in the wizard appropriately.
CLI	`smsap backup restore -profile targetdb1_prof1 -label full_bkup_sales_jun_08 -complete -recover -until "2008-06-29:15:29:23" -verbose`

SCENARIO 4: RESTORE A DATABASE PARTIALLY (ONE OR MORE DATAFILES) WITHOUT CONTROL FILES AND RECOVER USING ALL AVAILABLE LOGS

In this scenario, the current control files exist, but one or more datafiles are damaged or lost. Restore just those datafiles and recover the database from an existing full online backup using all available logs.

SnapManager Interface	Example Commands
GUI	• Right-click the backup and select “Restore/Recover…” to launch the restore and recovery wizard. • In the “Restore Configuration Information” screen of the wizard, select the “Selective Datafile/Tablespace Restore without Control Files” option. Expand the “Datafiles” list and using the “Ctrl” key select one or more datafiles that need to be restored. Then click the “Next” button. • In the “Recovery Configuration Information” screen of the wizard, select the “All Logs” option and click the “Next” button. • Navigate through the next few screens in the wizard appropriately.
CLI	`smsap backup restore -profile targetdb1_prof1 -label full_bkup_sales_jun_08 -files /u02/oradata/sales02.dbf /u02/oradata/sales03.dbf -recover -alllogs -verbose`

SCENARIO 5: RESTORE A DATABASE PARTIALLY (ONE OR MORE TABLESPACES) WITHOUT CONTROL FILES AND RECOVER USING ALL AVAILABLE LOGS

In this scenario, the current control files exist, but one or more tablespaces are dropped or one or more datafiles belonging to a tablespace are damaged or lost. Restore just those tablespaces and recover the database from an existing full online backup using all available logs.

SnapManager Interface	Example Commands
GUI	• Right-click the backup and select “Restore/Recover…” to launch the restore and recovery wizard. • In the “Restore Configuration Information” screen of the wizard, select the “Selective Datafile/Tablespace Restore without Control Files” option. Expand the “Tablespaces” list and using the “Ctrl” key select one or more tablespaces that need to be restored. Then click the “Next” button. • In the “Recovery Configuration Information” screen of the wizard, select the “All Logs” option and click the “Next” button. • Navigate through the next few screens in the wizard appropriately.
CLI	`smsap backup restore -profile targetdb1_prof1 -label full_bkup_sales_jun_08 –tablespaces users -recover -alllogs –verbose`

SCENARIO 6: RESTORE ONLY CONTROL FILES AND RECOVER USING ALL AVAILABLE LOGS

In this scenario, the datafiles 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.

SnapManager Interface	Example Commands
GUI	• Right-click the backup and select “Restore/Recover…” to launch the restore and recovery wizard. • In the “Restore Configuration Information” screen of the wizard, select the “Control Files Restore without Datafile/Tablespace” option and then click the “Next” button. • In the “Recovery Configuration Information” screen of the wizard, select the “All Logs” option and click the “Next” button. • Navigate through the next few screens in the wizard appropriately.
CLI	`smsap backup restore -profile targetdb1_prof1 -label full_bkup_sales_jun_08 –controlfiles -recover -alllogs –verbose`

SCENARIO 7: RESTORE A WHOLE DATABASE WITHOUT CONTROL FILES AND RECOVER USING THE BACKUP CONTROL FILES AND ALL AVAILABLE LOGS

In this scenario, all datafiles and 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.

SnapManager Interface	Example Commands
GUI	• Right-click the backup and select “Restore/Recover…” to launch the restore and recovery wizard. • In the “Restore Configuration Information” screen of the wizard, select the “Complete Datafile/Tablespace Restore without Control Files” option and then click the “Next” button. • In the “Recovery Configuration Information” screen of the wizard, select the “All Logs” and “Using Backup Control Files” options and click the “Next” button. • Navigate through the next few screens in the wizard appropriately.
CLI	**smsap backup restore -profile targetdb1_prof1 -label full_bkup_sales_jun_08 –complete -using-backup-controlfile -recover -alllogs –verbose**

SCENARIO 8: RECREATE A DROPPED TABLE BY EXPORTING IT FROM A CLONE OF A BACKUP

In this scenario, a table is dropped and needs to be imported back from an existing full online backup. To restore just that table, first create a clone from the backup using SnapManager. Then export the table from the clone database and import it back into the target database manually.

SnapManager Interface	Example Commands
GUI	• Right-click the backup and select “Clone...” • Navigate through the next few screens in the wizard appropriately. • Once the clone completes, manually export the table from the clone: **[oracle@tardb_host1][exp1][~]$ exp userid=user/password tables=sales file=sales08.dmp** • Once the export completes, manually import the table into the target database: **[oracle@tardb_host1][tardb1][~]$ imp userid=user/password tables=sales file=sales08.dmp**
CLI	• Create a clone from the backup of the target database. **smsap clone create -profile targetdb1_prof1 –backup-label full_bkup_sales_jun_08 -newsid exp1 -clonespec ./smsap/sales_clonespec.xml -reserve no -label sales0608_exp1 –verbose** • Once the clone completes, manually export the table from the clone. **[oracle@tardb_host1][exp1][~]$ exp userid=user/password tables=sales file=sales08.dmp** • Once the export completes, manually import the table into the target database. **[oracle@tardb_host1][tardb1][~]$ imp userid=user/password tables=sales file=sales08.dmp**

SCENARIO 9: RECREATE A DROPPED TABLE FROM A CLONE OF A BACKUP USING A DATABASE LINK

In this scenario, a table is dropped and needs to be recreated from an existing full online backup. To recreate just that table, first create a clone from the backup using SnapManager. Then manually create a database link from the target database to the clone and recreate the table in the target database using the link.

SnapManager Interface	Example Commands
GUI	• Right-click the backup and select “Clone...” • Navigate through the next few screens in the wizard appropriately. • Once the clone completes, manually add an entry for the clone database (`jun08cln`) in the `tnsnames.ora` file. • Create a database link in the target database to the clone database. **SQL> create public database link jun08_clone** **2 connect to sales identified by salespw** **3 using 'jun08cln';** • Recreate the dropped table in the target database by selecting from the table in the clone database using the database link. **SQL> create table europe_sales as** **2 select * from europe_sales@jun08_clone;**
CLI	• Create a clone from the backup of the target database. **smsap clone create -profile targetdb1_prof1 –backup-label full_bkup_sales_jun_08 -newsid jun08cln -clonespec ./smsap/sales_clonespec.xml -reserve no -label sales0608_exp1 –verbose** • Once the clone completes, manually add an entry for the clone database (`jun08cln`) in the `tnsnames.ora` file. • Create a database link in the target database to the clone database. **SQL> create public database link jun08_clone** **2 connect to sales identified by salespw** **3 using 'jun08cln';** • Recreate the dropped table in the target database by selecting from the table in the clone database using the database link. **SQL> create table europe_sales as** **2 select * from europe_sales@jun08_clone;**

APPENDIX F: CLONING A RAC DATABASE TO NON-RAC AND CONVERTING IT TO A RAC DATABASE

SnapManager for SAP clones a RAC database to a non-RAC database and sets the Oracle parameter `cluster.database` to `false`. The following procedure can be used to convert such a non-RAC database to RAC.

1. Create a clone from a backup using the SnapManager CLI:

```
smsap clone create -backup-label backup_label -profile <profile_name> -
newsid <new SID> -clonespec <full path to clonespec file> -verbose
```

2. When the cloning operation completes, rename the `initclone_SID.ora` file to `initclone_local_instance_SID.ora` and edit it and set `cluster.database` to `true`.
3. Register the cloned RAC database with `srvctl`.

   The following example sets up a cloned database as a one-node RAC instance. Use an editor, such as vi, to change the `.ora` file.

```
$ smsap clone create -backup-label rac_full_backup -profile rac_profile
-newsid CLONE1 -clonespec rac_nfs_clonespec.xml -verbose

$ export ORACLE_SID=CLONE1

$ sqlplus

       shutdown immediate;

$ vi initCLONE11.ora

       *.cluster_database = TRUE

       CLONE11.instance_number=1

       CLONE11.thread=1

       CLONE11.undo_tablespace='UNDOTBS1'

$ srvctl add database -d CLONE1 -o /u02/app/oracle/product/10.2.0/db -m
rtp.company.com -r  primary -y manual

$ srvctl add instance -d CLONE1 -i CLONE11 -n anzio

$ srvctl start instance -d CLONE1 -i CLONE11
```

   Note: You must undo all the steps you performed to turn this clone into a RAC database before attempting to delete the clone.

APPENDIX G: SAMPLE CLONING XML SPECIFICATION

Note: This XML file is provided as a sample only and cannot be used in its current format in a customer environment.

```
<clone-specification>

   <storage-specification>
      <storage-mapping>
         <mountpoint>
            <source>/oracle/<SOURCE SID>_sapdata</source>
            <destination>/oracle/<TARGET SID>_sapdata</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>/oracle/<TARGET SID>/origlogA/cntrl/cntrl<TARGET SID>.dbf</file>
         <file>/oracle/<TARGET SID>/origlogB/cntrl/cntrl<TARGET SID>.dbf</file>
         <file>/oracle/<TARGET SID>/sapdata1/cntrl/cntrl<TARGET SID>.dbf</file>
      </controlfiles>
      <redologs>
         <redogroup>
            <file>/oracle/<TARGET SID>/origlogA/log_g11m1.dbf</file>
            <file>/oracle/<TARGET SID>/mirrlogA/log_g11m2.dbf</file>
            <number>1</number>
            <size unit="M">100</size>
         </redogroup>
         <redogroup>
            <file>/oracle/<TARGET SID>/origlogB/log_g12m1.dbf</file>
```

```
            <file>/oracle/<TARGET SID>/mirrlogB/log_g12m2.dbf</file>
            <number>2</number>
            <size unit="M">100</size>
         </redogroup>
         <redogroup>
            <file>/oracle/<TARGET SID>/origlogA/log_g13m1.dbf</file>
            <file>/oracle/<TARGET SID>/mirrlogA/log_g13m2.dbf</file>
            <number>3</number>
            <size unit="M">100</size>
         </redogroup>
         <redogroup>
            <file>/oracle/<TARGET SID>/origlogB/log_g14m1.dbf</file>
            <file>/oracle/<TARGET SID>/mirrlogB/log_g14m2.dbf</file>
            <number>4</number>
            <size unit="M">100</size>
         </redogroup>
      </redologs>
      <parameters>
         <parameter>
            <name>log_archive_dest</name>
            <value>LOCATION=>/oracle/<TARGET SID>/oraarch</value>
            </parameter>
         <parameter>
            <name>background_dump_dest</name>
            <value>/oracle/<TARGET SID>/saptrace/background</value>
         </parameter>
         <parameter>
            <name>core_dump_dest</name>
            <value>/oracle/<TARGET SID>/saptrace/background</value>
         </parameter>
         <parameter>
            <name>user_dump_dest</name>
            <value>/oracle/<TARGET SID>/saptrace/usertrace</value>
         </parameter>
      </parameters>
   </database-specification>

   <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>


</clone-specification>
```

APPENDIX H: RESTORE SPECIFICATION XML FILE EXAMPLE

The following is an example of the restore specification XML file.

```
<?xml version="1.0" encoding="UTF-8"?>
<restore-specification xmlns="http://www.netapp.com">

   <!-- "Restore from file(s)" -->
   <file-mapping>
      <original-location>/mnt/path/db_10g/NFSDB/user01.dbf</original-
      location>
      <alternate-location>/mnt/vault/user01.dbf</alternate-location>
   </file-mapping>

   <!-- "Restore from host mounted filesystem(s)" -->
   <mountpoint-mapping>
      <original-location>/mnt/path/db_10g/DB</original-location>
      <snapname>D_HSDBR_hsdb1_F_C_1_8abc...0001_0</snapname>
      <alternate-location>/mnt/vault</alternate-location>
   </mountpoint-mapping>

   <!-- "Restore from ASM mounted filesystem(s)" -->
   <mountpoint-mapping>
      <original-location>+DG_1</original-location>
      <snapname>D_HSDBR_hsdb1_F_C_1_8abc...0001_0</snapname>
      <alternate-location>+D_12312341</alternate-location>
   </mountpoint-mapping>

   <!-- "Restore from raw device" -->
   <raw-device-mapping>
      <original-location>/dev/rdsk/c3t4d6s6</original-location>
      <alternate-location>/dev/rdsk/c3t5d2s6</alternate-location>
   </raw-device-mapping>

</restore-specification>
```

APPENDIX I: DISASTER RECOVERY WITH SNAPMANAGER 3.0 FOR SAP AND SNAPVAULT

SnapManager 3.0 for SAP provides policy-driven data protection by integrating with Protection Manager 3.7.1. This automates replicating SnapManager backups on a primary storage system to a secondary storage system using the SnapVault or SnapMirror policies created in Protection Manager by the storage or backup administrator.

In version 1.1 of SnapManager for SAP, because of the lack of integration with Protection Manager, SnapManager provided the following features to aid in scripting a solution that leverages NetApp SnapVault and SnapMirror to replicate the SnapManager backups from primary to secondary storage:

- Freeing backups
- Restoring from an alternate location

If Protection Manager is not available, because of which the data protection features of SnapManager cannot be leveraged, then the following steps provide an overview of how these features can be leveraged to integrate with SnapVault and provide a semiscripted disaster recovery solution.

Note: The commands are provided as an example only.

DATABASE CONFIGURATION

The database managed by SnapManager for SAP in this example was configured as follows:

- The database files reside on a NetApp storage system and are accessed over NFS.
- All files are in qtrees in flexible volumes.
- Qtrees are not shared between databases.
- Archived log files are segregated from datafiles.
- Datafiles and control files reside in different qtrees to make restores from vault more efficient.

SNAPVAULT CONFIGURATION

SnapVault between the primary and secondary storage systems was configured as follows:

- There is one secondary volume for each volume on the primary.
- Each qtree on the primary is vaulted to its corresponding secondary volume. Table 1 shows the SnapVault mappings between the qtrees in the primary storage system and the qtrees in the secondary storage system.

Table 1) Qtree mappings between primary and secondary storage.

Primary Storage System		Secondary Storage System	
Volume	Qtree	Volume	Qtree
oradata	oradata_qtree	oradata_vault	oradata_qtree
oracntrl	oracntrl_qtree	oracntrl_vault	oracntrl_qtree
oralog	oralog_qtree	oralog_vault	oralog_qtree

- The SnapVault relationship between the primary and secondary storage systems was configured by following the best practice recommendations in TR-3369, "NetApp Best Practice Guidelines for Oracle."
- An initial baseline copy of the qtree from the primary to the secondary storage system was executed by running the following command on the secondary storage system:

```
snapvault start -S <pri_filer_name>:/vol/oradata/oradata_qtree
<sec_filer_name>:/vol/oradata_vault/oradata_qtree

snapvault start -S <pri_filer_name>:/vol/oralog/oralog_qtree
<sec_filer_name>:/vol/oralog_vault/oralog_qtree

snapvault start -S <pri_filer_name>:/vol/oracntrl/oracntrl_qtree
<sec_filer_name>:/vol/oracntrl_vault/oracntrl_qtree
```

VAULTING BACKUPS AND RESTORING FROM AN ALTERNATE LOCATION

1. Create a full offline backup of the database using the SnapManager CLI:

```
smsap backup create –profile <profile_name> –full –offline –label
<backup_label_name> -force
```

2. Parse the output of the following CLI command to identify the Snapshot copies associated with the backup for each of the volumes of the database:

```
smsap backup show -profile <profile_name> -label <backup_label_name>
```

3. Protect the backup by vaulting the associated Snapshot copies to the secondary storage system.
    a. Vault the backup using rsh to the secondary storage system and update the secondary volume qtrees to the Snapshot copies created by SnapManager on the corresponding volumes on the primary storage system:

```
rsh <sec_filer_name> snapvault update -w –s <smsap_snapshot_name>
/vol/oradata_vault/oradata_qtree

rsh <sec_filer_name> snapvault update -w –s <smsap_snapshot_name>
/vol/oralog_vault/oralog_qtree

rsh <sec_filer_name> snapvault update -w –s <smsap_snapshot_name>
/vol/oracntrl_vault/oracntrl_qtree
```

    b. Get the name of the last Snapshot copy in each of the volumes on the secondary storage system:

```
rsh <sec_filer_name> snap list -b /vol/oradata_vault/oradata_qtree

rsh <sec_filer_name> snap list -b /vol/oralog_vault/oralog_qtree

rsh <sec_filer_name> snap list -b /vol/oracntrl_vault/oracntrl_qtree
```

    c. Rename the Snapshot copy created on each of the volumes on the secondary storage system to be the same as the Snapshot copies created by SnapManager on the corresponding volumes on the primary storage system:

```
rsh <sec_filer_name> snap rename /vol/oradata_vault/oradata_qtree
<current_snapshot_name> <smsap_snapshot_name>

rsh <sec_filer_name> snap rename /vol/oralog_vault/oralog_qtree
<current_snapshot_name> <smsap_snapshot_name>

rsh <sec_filer_name> snap rename /vol/oracntrl_vault/oracntrl_qtree
<current_snapshot_name> <smsap_snapshot_name>
```

4. Optionally, free up the SnapManager backup on the primary storage system using the SnapManager CLI:

```
smsap backup free -profile <profile_name> -label <backup_label_name>
```

5. In the event of a disaster, you can do one of the following:
   - Restore the required files from the secondary to a qtree on the primary storage system.
   - Use FlexClone to clone the vaulted Snapshot copies on the secondary storage system (this is much faster).

   SnapManager will not restore archive logs even if they are specified in the restore specification. Therefore, restore the archive logs manually or using SnapVault to the original qtree on the primary storage system:

```
rsh <sec_filer_name> vol clone create /vol/oradata_vault_clone –s file –b
/vol/oradata_vault <smsap_snapshot_name>

rsh <sec_filer_name> vol clone create /vol/oracntrl_vault_clone –s file –
b /vol/oracntrl_vault <smsap_snapshot_name>
```

6. Mount the cloned secondary volumes on the database host:

```
mount <sec_filer_name>:/vol/oradata_vault_clone /u01_oradata_scratch

mount <sec_filer_name>:/vol/oracntrl_vault_clone /u01_oracntrl_scratch
```

7. Parse the output of the "`smsap backup show`" command to identify the primary storage volumes and the corresponding Snapshot copies:

```
smsap backup show -profile <profile_name> -label <backup_label_name>
```

8. Using the above data, generate a restore spec XML file with the original and alternate locations' mappings and save the file in a location accessible from the SnapManager CLI:

```
<?xml version="1.0" encoding="UTF-8"?>
<restore-specification xmlns="http://www.netapp.com">
       <!-- "Restore from host mounted filesystem(s)" -->
       <mountpoint-mapping>
              <original-location>/u02/oradata/NA1</original-location>
              <snapname>smsap_NA1_NA1_f_h_1_...0001_0</snapname>
              <alternate-location>/u02/oradata_scratch/NA1</alternate-
              location>
       </mountpoint-mapping>
       <mountpoint-mapping>
              <original-location>/u03/oracntrl/NA1</original-location>
              <snapname>smsap_NA1_NA1_f_h_2_...0001_0</snapname>
              <alternate-location>/u03/oracntrl_scratch/NA1</alternate-
              location>
       </mountpoint-mapping>
</restore-specification>
```

9. Use the SnapManager CLI to perform restore and recovery from the alternate location by specifying the location of the restore spec XML file:

```
smsap backup restore -profile <profile_name> -label <backup_label_name>
–controlfiles –recover –nologs –restorespec <restorespec_file_location>
```

APPENDIX J: SAMPLE SCRIPT TO CREATE A BACKUP AND SEND OUT AN E-MAIL NOTIFICATION

The following sample script illustrates how the SnapManager CLI can be leveraged to create a full online backup and send out an e-mail notification with the status and log of the backup operation. Scripts for other SnapManager operations can be created similarly.

Note: This script is provided as an example only. This script is not considered a NetApp product and is not supported.

```
#!/usr/bin/env bash
#
# This script will do the following:
#
# - Create a full online backup with a unique label
#   using SnapManager for SAP
# - Check the log file if the backup succeeded
# - Email the status and the log of the backup operation
#
#


#
# Generate a unique label for the backup
#
my_bkup_label=$(date +"full_hot_"%m%d%Y"_"%H"h"%M"m"%S"s")
my_smsap_profile=targetdb1_prof1
my_bkup_logfile=/home/oracle/log/smsap_bkup_$my_bkup_label.log
my_from_addr=xyz@abc.com
my_to_addr=dl_dbas@abc.com,tom.jones@abc.com
my_bkup_status=""

#
# SMSAP command to create a full online backup
#
echo -e "\nCreating a SnapManager backup with label $my_bkup_label ..."
smsap backup create -profile $my_smsap_profile -label $my_bkup_label -
full -online -verbose > $my_bkup_logfile 2>&1
echo -e "\nBackup operation completed."

#
# Check the logfile for the status of the backup
#
echo -e "\nChecking the status of the backup operation ..."
grep SUCCESS $my_bkup_logfile > /dev/null  2>&1
```

```
if [ "$?" -ne "0" ]; then
{
    my_bkup_status="Failed"
    echo "SMSAP backup $my_bkup_label failed!"
}
else
{
    my_bkup_status="Succeeded"
    echo "SMSAP backup $my_bkup_label succeeded!"
}
fi

echo "Please check the logfile $my_bkup_logfile for more details."

#
# Send an email notification with the status and the log
#
mail -s "SMSAP Backup $my_bkup_label $my_bkup_status !" $my_to_addr -- -r
$my_from_addr < $my_bkup_logfile
echo -e "\nSent an email notification to $my_to_addr \n"

exit 0
```

Sample output of the above script:

```
[oracle@tardb_host1][~]$ ./smsap_bkup_email.sh

Creating a SnapManager backup with label full_hot_07242008_10h23m25s ...

Backup operation completed.

Checking the status of the backup operation ...
SMSAP backup full_hot_07242008_10h23m25s succeeded!
Please check the logfile
/home/oracle/log/smsap_bkup_full_hot_07242008_10h23m25s.log for more
details.

Sent an email notification to dl_dbas@abc.com,tom.jones@abc.com
```

APPENDIX K: SAMPLE SCRIPT TO COPY THE LAST SUCCESSFUL BACKUP TO TAPE

The following sample script illustrates how the SnapManager CLI can be leveraged to identify the last successful backup created by SnapManager and generate the necessary commands to copy the Snapshot copies associated with that backup to a secondary location such as tape.

Note: This script is provided as an example only. This script is not considered a NetApp product and is not supported.

```
#!/usr/bin/env bash

#
# This script will do the following:
#
# - Parse the 'smsap backup list' command to get the
#   label of the last successful backup
# - Pass that backup label to the 'smsap backup show'
#   command to get all the volumes backed up in that
#   backup and their associated Snapshot names
# - Generate the necessary commands to copy the
#   Snapshot copies associated with that backup
#   to a secondary location like tape etc.
#
# Assumptions:
# - The database used in this example is laid out
#   on a single NetApp storage system
#

my_smsap_profile=targetdb1_prof1
my_storage_system=netapp_prod

#
# Get the label of the last successful backup
#
my_bkup_label=`smsap backup list -profile $my_smsap_profile | grep
"SUCCESS" | head -1 | awk '{ print $7 }'`;
my_bkup_label_date=`smsap backup list -profile prod | grep "SUCCESS" |
head -1 | awk '{ print $1,$2 }'`;
echo -e "\nBackup Label: $my_bkup_label Start Date:
$my_bkup_label_date\n"
```

```
#
# Generate and print the necessary commands to backup to
# a secondary location like tape etc. This command can be modified
# for other tape backup software like Legato Networker, IBM TSM,
# Veritas NetBackup etc.
#

smsap backup show -profile $my_smsap_profile -label $my_bkup_label | sed
-n '/Snapshots:/{n;p;}' | awk 'BEGIN { FS = ":" }; {print "echo -e
/usr/bin/rsh" '$my_storage_system' "-l user:password dump -0uf nrst0a "
$2 "/.snapshot/" $3 " &" "\n"}'
```

Sample output of the above script:

```
[oracle@tardb_host1][~]$ ./smsap_bkup_tape.sh

Backup Label: full_hot_02202009_14h11m54s Backup Start Date: 2009-02-20
15:12:18
echo -e /usr/bin/rsh-l user:password dump -0uf nrst0a
/vol/oradata/.snapshot/smsap_prod_prod_f_h_1_8a9bd4af1f94ac65011f94ac6d00
0001_0 &
echo -e /usr/bin/rsh-l user:password dump -0uf nrst0a
/vol/oralog/.snapshot/smsap_prod_prod_f_h_2_8a9bd4af1f94ac65011f94ac6d000
001_0 &
echo -e /usr/bin/rsh-l user:password dump -0uf nrst0a
/vol/oraarch/.snapshot/smsap_prod_prod_f_h_2_8a9bd4af1f94ac65011f94ac6d00
0001_0 &
```

www.netapp.com