Technical Report

Automating SAP System Copies
Using the SnapCenter 4.0 SAP HANA Plug-In

Nils Bauer, NetApp
March 2018 | TR-4667

Abstract

This document describes the workflows for SAP system copy, refresh, and clone using the NetApp® SnapCenter® 4.0 SAP HANA plug-in.

SnapCenter’s storage cloning and the option to flexible define precloning and postcloning operations allows SAP Basis administrators to accelerate and automate SAP system copy, clone, or refresh operations. The option to choose any SnapCenter Snapshot® backup at any primary or secondary storage allows you to address your most important use cases, including logical corruption, disaster recovery testing, or the refresh of an SAP QA system.
TABLE OF CONTENTS

1 Introduction .......................................................................................................................... 4
  1.1 The Foundation: Application-Consistent NetApp Snapshot Backups ................................. 4
  1.2 Options: Data Protection: Off-Site Backups and/or Disaster Recovery ......................... 4
  1.3 Flexibility: Use Any Production Snapshot Copy for SAP System Provisioning .............. 5
  1.4 Integration: Data Protection and Efficient SAP System Provisioning ........................... 5
  1.5 Technology: Rapid, Space-Efficient Provisioning Based on Storage Cloning ................. 5

2 SAP System Copy, Refresh, and Clone Scenarios ................................................................. 5

3 Overview of SAP System Refresh Workflows with SnapCenter ........................................ 6
  3.1 SAP System Refresh without Clone Split Workflow .................................................... 7
  3.2 SAP System Refresh with Clone Split Workflow ....................................................... 9

4 Overview of SAP System Copy Workflow with SnapCenter .............................................. 10

5 Overview of SAP System Clone Workflow with SnapCenter ............................................. 11

6 Example Automation Scripts .............................................................................................. 12
  6.1 Example Script sc-system-refresh.sh ....................................................................... 12
  6.2 Example Script sc-mount-volume.sh ...................................................................... 15

7 SAP System Refresh Workflow Descriptions ....................................................................... 15
  7.1 Lab Setup Used for SAP System Refresh Operations .................................................. 15
  7.2 SnapCenter Clone Create Workflow using SAP HANA Studio for Recovery .................. 16
  7.3 SnapCenter Clone Create Workflow Using a Recovery Script .................................... 44
  7.4 SnapCenter Clone Delete Workflow Using External Scripts ....................................... 49
  7.5 SnapCenter Clone Split Workflow .......................................................................... 51
  7.6 SnapCenter Clone Create Workflow When the Cloned Volume Has Been Split .............. 54
  7.7 Automation with PowerShell Scripts ...................................................................... 62

8 SAP System Clone Workflow Description .......................................................................... 64
  8.1 Lab Setup used for SAP System Clone Operation ...................................................... 64
  8.2 Preparation of Target Server .................................................................................... 65
  8.3 Cloning and Mount Operations .............................................................................. 67
  8.4 Fencing, Recovery and Start Operations .................................................................. 75

9 Conclusion .......................................................................................................................... 81

Appendix ................................................................................................................................ 82
  Example Script sc-system-refresh.sh ............................................................................. 82
  Configuration file example sc-system-refresh-QP1.cfg .................................................. 86

© 2018 NetApp, Inc. All Rights Reserved.
1 Introduction

In today’s dynamic business environment, companies must provide ongoing innovation and react quickly to changing markets. Under these competitive circumstances, companies that have implemented greater flexibility in their work processes can adapt to market demands more effectively.

Changing market demands also affect a company’s SAP environments such that they require regular integrations, changes, and updates. IT departments must implement these changes with fewer resources and over shorter time periods. Minimizing risk when deploying those changes requires thorough testing with quality assurance (QA) or test systems with actual production data.

Traditional SAP lifecycle-management approaches to QA and test-system provisioning are primarily based on manual processes. These manual processes are often error-prone and time-consuming, delaying innovation and the response to business requirements.

NetApp solutions for optimizing SAP lifecycle management are integrated into SAP HANA database and lifecycle management tools, combining efficient application data protection with the flexible provisioning of SAP test systems, as is shown in Figure 1.

Figure 1) Optimizing SAP lifecycle management by product integration.

1.1 The Foundation: Application-Consistent NetApp Snapshot Backups

The ability to create application-consistent NetApp Snapshot® backups on the storage layer is the foundation for all operations described in this document. Storage-based Snapshot backups are created by using the NetApp SnapCenter Plug-In for SAP HANA and interfaces provided by the SAP HANA database. SnapCenter registers Snapshot backups in the SAP HANA backup catalog so that the backups are visible within SAP HANA Studio or Cockpit and can be selected for restore and recovery operations.

Storage-based Snapshot backups provide significant advantages when compared with traditional backup approaches, including rapid backup and restore processes (less than a minute), no performance drain on HANA database servers, and no network load during the backup process.

1.2 Options: Data Protection: Off-Site Backups and/or Disaster Recovery

Application-consistent Snapshot backups can be replicated on the storage layer to an off-site backup site or a disaster recovery site controlled by SnapCenter. Replication is based on block changes and is therefore space and bandwidth efficient. In addition, different backup retention policies can be defined for backups on the primary and off-site backup sites.
1.3 Flexibility: Use Any Production Snapshot Copy for SAP System Provisioning

NetApp technology and software integration allows you to use any existing Snapshot copy of the production system as a source for an SAP system copy. This storage can be either the same storage that is used for the SAP production systems, the storage that is used for off-site backups, or the storage at the disaster recovery site. This flexibility allows you to separate development and test systems from production if required and also covers other situations, such as the testing of disaster recovery scenarios at the disaster recovery site.

1.4 Integration: Data Protection and Efficient SAP System Provisioning

There are various scenarios and use cases for the provisioning of SAP test systems and you might also have different requirements for the level of automation. NetApp software products for SAP integrate into database and lifecycle management products from SAP to support different scenarios and levels of automation.

NetApp SnapCenter with the plug-in for SAP HANA is used to provision the required storage volumes based on an application-consistent Snapshot backup. SAP HANA Studio, SAP HANA Cockpit, or SAP Landscape Management (LaMa) is used to perform the required changes on the target system to which the provisioned storage volumes have been attached. Depending on the use case, including SAP system copy, system clone, or system refresh, additional manual steps, such as SAP postprocessing, might be required. More details are covered in the section “SAP System Copy, Refresh, and Clone Scenarios.”

A fully automated, end-to-end provision of SAP test systems can be performed by using SAP LaMa. NetApp Storage Services Connector integrates into SAP LaMa and provides the required operations for SAP LaMa at the storage layer. More details can be found at Integrating NetApp ONTAP Systems with SAP Landscape Management.

1.5 Technology: Rapid, Space-Efficient Provisioning Based on Storage Cloning

NetApp FlexClone® copies provide space-efficient volume clones directly at the storage level in a manner completely transparent to the user. FlexClone copies are based on Snapshot copies and can be created in a matter of seconds without interrupting operations at the source volume. Because data is not copied but rather is referenced in place, the amount of storage required is limited to data that is changed at the source and the target system.

2 SAP System Copy, Refresh, and Clone Scenarios

The phrase “SAP system copy” is often used as a synonym for three different operations: SAP system refresh, SAP system copy, or SAP system clone operations. On the other hand, it is important to distinguish between the different operations because the workflows and use cases differ for each one.

- **SAP System Refresh.** An SAP system refresh is a refresh of an existing target SAP system with data from a source SAP system. The target system is typically part of an SAP transport landscape, for example a quality assurance system, that is refreshed with data from the production system. The hostname, instance number, and SID are different for the source and target systems.

- **SAP System Copy.** An SAP system copy is a setup of a new target SAP system with data from a source SAP system. The new target system could be for example an additional test system with data from the production system. The hostname, instance number, and SID are different for the source and target systems.

- **SAP System Clone.** An SAP system clone is an identical clone of a source SAP system. SAP system clones are typically used to address logical corruption or to test disaster recovery scenarios. With a system clone operation, the hostname, instance number, and SID remain the
same. It is therefore important to establish proper network fencing for the target system to make sure that there is no communication with the production environment.

Figure 2 illustrates the main steps that must be performed during a system refresh, system copy, or system clone operation. The blue boxes indicate steps that can be automated with SnapCenter, while the green boxes indicate steps that must be performed outside of SnapCenter, either manually or by using third-party tools.

All three operations can be fully automated by using SAP LaMa and the NetApp Storage Services Connector. More details can be found at Integrating NetApp ONTAP Systems with SAP Landscape Management.

Figure 2) SAP system refresh, copy, and clone operations.

3 Overview of SAP System Refresh Workflows with SnapCenter

The following chapter provides an overview of the steps in an SAP system refresh workflow with SnapCenter. Figure 3 summarizes the required steps.
Before a first SAP system refresh workflow can be performed with SnapCenter, some initial one-time preparations must be performed.

- The SAP target system, the SAP application, and the SAP HANA database must be installed at the target host.
- After installation, the SAP application and the HANA database must be stopped, and the SAP HANA data volume must be unmounted.

After the initial preparation steps, the SnapCenter clone create workflow when clone has not been split must be executed. After this workflow, the SAP HANA database is started. You can now choose if the storage FlexClone copy should be split with the SnapCenter clone split workflow. In general, it makes sense to split the storage FlexClone if the target system is used for a longer period of time. More details can be found in the chapter “SnapCenter Clone Split”.

SAP post-processing must now be performed either manually or using third-party tools.

For the next SAP system refresh operation, the workflow steps are different depending on whether the FlexClone copy has already been split. If the FlexClone copy has not been split, then the SnapCenter clone delete workflow is used to stop the SAP HANA database and delete the FlexClone copy at the storage layer. If the FlexClone copy has been split, the SnapCenter clone create workflow when the cloned volume has been split must to be performed.

### 3.1 SAP System Refresh without Clone Split Workflow

As shown in Figure 3, two SnapCenter workflows must be performed for an SAP system refresh without clone split operation.

- The SnapCenter clone create workflow when the cloned volume has not been split
- The SnapCenter clone delete workflow

The following chapters explain these two workflows in more detail.

**SnapCenter Clone Create Workflow When the Cloned Volume Has Not Been Split**

Figure 4 shows the different options for performing a clone create workflow with SnapCenter.
1. The SnapCenter cloning operation can be performed at either primary or secondary storage (a SnapVault or SnapMirror destination). A clone create operation is always based on a Snapshot backup that has been created by SnapCenter.

2. SnapCenter creates a FlexClone copy based on a selected backup, mounts the volume to the namespace, and then exports the volume to the provided IP address.

3. SnapCenter creates a new SnapCenter resource for the target system.

4. Mounting the cloned volume (FlexClone copy) can be performed either manually or by using a mount script added to the SnapCenter workflow.

5. The recovery of the SAP HANA database can be performed either manually with SAP HANA Studio or Cockpit or by using a post-clone script added to the SnapCenter workflow.

6. At the end of the workflow, the SAP HANA database is up and running.

Figure 4) SnapCenter clone create workflow when the cloned volume has not been split.

The workflows are described in the chapters “SnapCenter Clone Create Workflow using SAP HANA Studio for Recovery” and “SnapCenter Clone Create Workflow Using a Recovery Script.SnapCenter Clone Create Workflow Using a Recovery Script”

SnapCenter Clone Delete Workflow

Figure 5 shows the different options for executing a clone delete operation with SnapCenter. Before the storage resources can be deleted, the SAP HANA database must be stopped, and the storage volume must be unmounted.

1. Stopping the SAP HANA database and unmounting the storage volume can be performed manually before the SnapCenter clone delete operation is started or it can be executed within the SnapCenter workflow by adding scripts.

2. The SnapCenter clone delete operation deletes the FlexClone copy at the storage layer and deletes the SnapCenter resource that has been created for the target system.
Automating SAP System Copies using the SnapCenter 4.0 SAP HANA Plug-In - © 2018 NetApp, Inc. All Rights Reserved.

Figure 5) SnapCenter clone delete workflow.

The workflow using a pre script and an umount script is described in the chapter “SnapCenter Clone Delete Workflow Using External Scripts.”

3.2 SAP System Refresh with Clone Split Workflow

As shown in Figure 3, two SnapCenter workflows must be performed when performing an SAP system refresh with clone split operation.

- SnapCenter clone create workflow when the cloned volume has been split
- SnapCenter clone split workflow

The following chapters describe the two workflows in more detail.

SnapCenter Clone Create Workflow when the Cloned Volume Has Been Split

Figure 6 shows a clone create operation with SnapCenter after a clone split operation has been executed with automation scripts.

1. The SnapCenter resource on the target system must be deleted before the clone operation can be performed.
2. The SnapCenter cloning operation can be performed at either primary or secondary storage (a SnapVault or SnapMirror destination). A clone create operation is always based on a Snapshot backup created by SnapCenter.
3. SnapCenter pre-scripts are used to stop the SAP HANA database at the target host and to unmount the SAP HANA data volume.
4. SnapCenter creates a FlexClone copy based on a selected backup, mounts the volume to the namespace, and exports the volume to the provided IP address.
5. Mounting the cloned volume (a FlexClone copy) is performed using a SnapCenter mount script.
6. The recovery of the SAP HANA database is performed using a SnapCenter post clone script.
7. At the end of the workflow, the SAP HANA database is up and running.
Automating SAP System Copies using the SnapCenter 4.0 SAP HANA Plug-In

© 2018 NetApp, Inc. All Rights Reserved.

Figure 6) SnapCenter clone create workflow when the cloned volume has been split.

![Diagram of SnapCenter clone create workflow](image)

**Note:** If a clone split operation has been performed using SnapCenter, the clone delete operation is not available. In this case, the split storage volume must be deleted directly on the storage system. See the chapter “SnapCenter Clone Split”.

This workflow is described in the chapter “SnapCenter Clone Create Workflow When the Cloned Volume Has Been.”

SnapCenter Clone Split Workflow

Figure 7 shows the SnapCenter clone split workflow.

1. The FlexClone copy is split on the storage layer.
2. The target resource within SnapCenter becomes an independent resource. The relation to the source system backup is removed.

This workflow is described in the chapter “SnapCenter Clone Split Workflow”.

Figure 7) SnapCenter clone split workflow.

![Diagram of SnapCenter clone split workflow](image)

4 Overview of SAP System Copy Workflow with SnapCenter

The following chapter provides an overview of the different steps for an SAP system copy workflow with SnapCenter. Figure 8 summarizes the required steps.

The required steps are a portion of the steps described in the chapter “Overview of SAP System Refresh Workflows with SnapCenter.”
The SnapCenter workflows (SnapCenter clone create and SnapCenter clone split) are identical to the workflows in the chapters “SnapCenter Clone Create Workflow When the Cloned Volume Has Not Been ” and “SnapCenter Clone Split Workflow.”

Figure 8) Overview of SAP system copy workflow with SnapCenter.

5 Overview of SAP System Clone Workflow with SnapCenter

The following chapter provides an overview of the different steps of an SAP system clone workflow with SnapCenter. Figure 9 summarizes the required steps.

The workflow can be divided in three main parts.

1. Target server preparations
2. Clone and mount operations
3. Fencing, recovery, and start operations

A detailed description of the SAP System Clone Workflow is covered in the chapter “SAP System Clone Workflow Description.”
6 Example Automation Scripts

SnapCenter allows you to add external scripts at different steps of its workflows to provide further automation. To use automation scripts with SnapCenter, the SAP HANA plug-in must be deployed at the target host. More details on the deployment options of the SAP HANA plug-in can be found at SAP HANA Backup and Recovery with SnapCenter.

Typical clone workflow tasks that are automated using external scripts are:

- **Clone create**
  - Mount operation at the target host
  - Recovery of the SAP HANA database at the target host
- **Clone delete**
  - Shutdown of the SAP HANA database at the target host
  - Unmount operation at the target host

6.1 Example Script sc-system-refresh.sh

The example script `sc-system-refresh.sh` can be used to execute mount and unmount operations for SAP HANA data volumes as well as recovery and shutdown operations. The script is called with specific command-line options within the SnapCenter workflows “clone create” and “clone delete” as shown in Figure 10.

The script is generic and is configured using a SID-specific configuration file and must be made available to the target host of the system refresh or copy operation. If the script is used for multiple target hosts, it makes sense to provide an NFS share from which the script is made available to all target hosts.

The script supports the recovery operation for SAP HANA single-container systems as well as for SAP HANA MDC single tenant systems.

**Note:** The script currently only supports SAP HANA systems using NFS as a storage protocol and does not support SAP HANA multiple-host systems.
Note: The example script (found in the Appendix) is provided as is and is not a NetApp-supported product.

Figure 10) System refresh script operations.

Script Command-Line Parameters

The script must be called using two primary command-line parameters.

The operation to be executed requires the following information:

- Either mount, recover, shutdown, or umount
- The SID of the target database

For a mount operation, a third command-line parameter provides the junction path of the cloned volume. The command line has the following format:

- Cloning from primary storage

  \[\text{sc-system-refresh.sh} \text{ mount } \text{SID} \%<\text{primary-volume-name}>_\text{Clone}\]

  The SnapCenter variable \(<\text{primary-volume-name}>_\text{Clone}\) provides the junction path of the cloned volume. In our example with \text{SID}=QP1 and \text{primary-volume-name}=SP1_data_mnt00001, the script CLI parameter is as follows:

  \[\text{sc-system-refresh.sh} \text{ mount } QP1 \%SP1\_data_mnt00001\_Clone\]

- Cloning from secondary storage (a SnapVault or SnapMirror target)

  \[\text{sc-system-refresh.sh} \text{ mount } \text{SID} \%<\text{primary-svm}>\_<\text{secondary-volume-name}>_\text{Clone}\]

  The SnapCenter variable \(<\text{primary-svm}>\_<\text{secondary-volume-name}>_\text{Clone}\) provides the junction path of the cloned volume. In our example with \text{SID}=QP1, \text{primary-svm}=hana and \text{secondary-volume-name}=SP1_data_mnt00001_vault, the script CLI parameter is as follows:

  \[\text{sc-system-refresh.sh} \text{ mount } QP1 \%hana\_SP1\_data_mnt00001\_vault\_Clone\]

  For SnapMirror secondary storage with a \text{secondary-volume-name}=SP1_data_mnt00001_mirror, the CLI parameter is as follows:

  \[\text{sc-system-refresh.sh} \text{ mount } QP1 \%hana\_SP1\_data_mnt00001\_mirror\_Clone\]
Script Configuration File

This script uses a configuration file to configure target system-specific parameters. The configuration file must have a SID-specific file name `sc-system-refresh-SID.cfg`. The SID is a command line parameter that must be added when calling the script `sc-system-refresh.sh`.

**Note:** The database user, which is configured with the hdbuserstore key for the target system, must exist in the source database and must have the correct rights to allow database recovery.

The configuration file parameters are shown in Table 1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HANA_ARCHITECTURE</strong></td>
<td>Can be either &quot;single_container&quot; or &quot;MDC_single_tenant&quot;</td>
</tr>
<tr>
<td><strong>KEY</strong></td>
<td>For example: QP1KEY</td>
</tr>
<tr>
<td><strong>TIME_OUT_START</strong></td>
<td>For example: 18 (18 x 10sec = 3min)</td>
</tr>
<tr>
<td><strong>TIME_OUT_STOP</strong></td>
<td>For example: 18 (18 x 10sec = 3min)</td>
</tr>
<tr>
<td><strong>INSTANCENO</strong></td>
<td>For example: 00</td>
</tr>
<tr>
<td><strong>STORAGE</strong></td>
<td>For example: hanaprimary</td>
</tr>
</tbody>
</table>

The following output shows an example configuration file for an SAP HANA MDC single tenant system with SID=QP1.

```bash
stlrx300s8-7:/mnt/hwval/System-Copy/SnapCenter # cat sc-system-refresh-QP1.cfg
# Target database specific parameters
# HANA_ARCHITECTURE "single_container" or "MDC_single_tenant"
HANA_ARCHITECTURE="MDC_single_tenant"
# hdbuserstore key, which should be used to connect to the target database
KEY="QP1KEY"
# Timeout to start the database in 10sec intervals
# E.g. 3min = 180sec, TIME_OUT_START=18
TIME_OUT_START=18
# Timeout to stop the database in 10sec intervals
# E.g. 3min = 180sec, TIME_OUT_STOP=18
TIME_OUT_STOP=18
# Instance number of the target database
INSTANCENO="00"
# Data LIF of the SVM, which should be used to mount the volume
STORAGE="192.168.173.102"
```
6.2 Example Script sc-mount-volume.sh

The example script `sc-mount-volume.sh` can be used to execute mount operations of nondatabase volumes. The script has the following command-line syntax:

```
sc-mount-volume.sh <SnapCenter-Junction-Path> <Mount Point at target server>
```

The SnapCenter junction path parameter has the same syntax as described in the chapter "Script Command-Line Parameter" for the `sc-system-refresh.sh` script.

Note: The current version of the script uses a fixed IP address/host name to mount the volume from the storage system. This IP address/host name must be adapted within the script, or the script can be changed to get the IP address hostname as an additional command-line parameter.

Note: The script currently only supports nondatabase volumes mounted with NFS.

Note: The example script (found in the Appendix) is provided as is and is not a NetApp supported product.

7 SAP System Refresh Workflow Descriptions

7.1 Lab Setup Used for SAP System Refresh Operations

The following software release have been used in the lab setup:

- SAP HANA 2.0 SPS1 MDC single tenant
- SLES 12 SP1
- SnapCenter 4.0
- ONTAP 9.1

Figure 11 shows the components and configuration of the lab setup.

The SAP HANA database SP1 (MDC single tenant) is used as the source for the system refresh operations. Local backups at the primary storage as well as SnapVault replication to offsite backup storage have been configured for the SP1 system within SnapCenter.

The SAP HANA database QP1 (MDC single tenant) is used as the target for the system refresh operations. An initial SAP HANA installation has been performed for the QP1 system. The SnapCenter SAP HANA plug-in has been deployed on the target server. Cloning from the primary storage as well as from the secondary storage is described in the document.

The SAP HANA systems are connected with NFS to the storage systems. The setup has been performed according to the configuration guide [SAP HANA on NetApp All Flash FAS Systems with NFS](#).
7.2 SnapCenter Clone Create Workflow using SAP HANA Studio for Recovery

This chapter covers the system refresh workflow using a Snapshot backup, either at the primary or the secondary storage. The file system mount operation at the target host is performed by using a mount script that is added to the SnapCenter workflow.

SAP HANA database recovery is performed with SAP HANA Studio. The recovery process is described for a recovery to the latest save point and a recovery using the log backups from the source system.

Note: The recovery can also be performed with SAP HANA Cockpit.

Figure 12 highlights the workflow components.

Clone and Mount Operation with SnapCenter

Clone from Primary Storage

The script configuration file for the target SID (in our example QP1) must include the data LIF IP address or hostname of the SVM that should be used to mount the cloned volume. For more information, see the chapter “Script Configuration File.”
1. Log on to SnapCenter and go to the resource topology view.
2. Select Local Copies and the backup that should be used.
3. Click Clone from Backup.

4. Select the clone target server and provide the target server IP address to which the volume should be exported.
5. Enter the mount command.
   In our example, the SID = QP1, and the primary volume name = SP1_data_mnt00001.
   Use the command:
   ```bash
   sc-system-refresh.sh mount QP1 %SP1_data_mnt00001_Clone
   ```

6. Click Next.
7. Click Finish to start the cloning workflow.

8. The Job Details show the different steps that were executed.
On the target host, the cloned volume is now mounted.

```
stlrx300s8-7:~ # df -h | grep QP1
192.168.173.104:/QP1_log_mnt00001  150G  7.2G  143G   5%  /hana/log/QP1/mnt00001
192.168.173.102:/QP1_shared/shared  150G  36G  115G  24%  /hana/shared
192.168.173.102:/QP1_shared/usr-sap  150G  36G  115G  24%  /usr/sap/QP1
192.168.173.102:/Sc264e52b0-dea0-4f88-bfa0-65346c7f1709  150G  7.9G  142G  6%  /hana/data/QP1/mnt00001
stlrx300s8-7:~ #
```

The topology view of the source system shows the clone at the primary storage.
9. SnapCenter automatically creates a new resource for the clone target.

**Note:** The resource configuration and protection of the new resource must be adapted if you plan to perform backup operations.
Clone from Secondary Storage

The script configuration file for the target SID, in our example QP1, must include the data LIF IP address or the hostname of the SVM that should be used to mount the cloned volume. For more information, see the chapter “Script Configuration File.”

1. Log on to SnapCenter and go to the resource topology view.
2. Select Vault Copies and the backup that should be used.
3. Click Clone from Backup.

4. Select the clone target server and provide the target server IP address to which the volume should be exported.
5. Enter the mount command. In our example, the SID = QP1, the source SVM = hana, and the secondary volume name = SP1_data_mnt00001_vault.

```
sc-system-refresh.sh mount QP1 %hana_SP1_data_mnt00001_vault_Clone
```

6. Click Finish to start the cloning workflow.
7. The Job Details show the different steps that were executed.
8. On the target host, the cloned volume is now mounted.

```
stlrx300s8-7: # df -h | grep QP1
192.168.173.104:/QP1_log_mnt00001 150G  7.2G  143G  5% /hana/log/QP1/mnt00001
192.168.173.102:/QP1_shared/shared 150G  36G  115G  24% /hana/shared
192.168.173.102:/QP1_shared/usr-sap 150G  36G  115G  24% /usr/sap/QP1
192.168.173.107:/Sc94feefdf-6fa1-4bd3-922d-aba9755feae6 150G  76G  74G  52% /hana/data/QP1/mnt00001
stlrx300s8-7: #
```

9. The topology view of the source system shows the clone at the secondary storage.

10. SnapCenter automatically creates a new resource for the clone target.

**Note:** The resource configuration and protection of the new resource must be adapted if you plan to perform backup operations.

**Recovery to Last SAP HANA Save Point**

The recovery of an SAP HANA MDC single tenant system with SAP HANA Studio is done in two steps:

- Recovery of the system database
- Recovery of the tenant database

**Note:** The recovery of an SAP HANA single container system is performed in a single step. The parameters in HANA Studio are identical to the recovery of the system database.
Recovery of the System Database

1. Within SAP HANA Studio, select the system database and select Recover System Database.

2. Select Recover the Database to a Specific Backup.

3. Click OK.

4. Select Recover Without the Backup Catalog.
5. Select Snapshot for the destination type.

6. Click Next.
7. Click Finish to start the recovery.

8. Recovery of the system database finishes.
Recovery of the Tenant Database

1. Within SAP HANA Studio, select the system database and select Recover Tenant Database.

2. Select the tenant database.
3. Select Recover the database to a specific backup.

4. Select Recover Without the Backup Catalog.
5. Select Snapshot for the destination type.

6. Click Next.
7. Click Finish to start the recovery.

8. Recovery of the tenant database finishes.
Recovery Using Log Backups from the Source System

The recovery of an SAP HANA MDC single tenant system using log backups from the source system consist of three steps:

1. Provide access to the log backups of the source system
2. Recover the system database
3. Recover the tenant database

Note: The recovery of an SAP HANA single container system is done in a single step. The parameters in HANA Studio are identical to the recovery of the system database.

Providing Access to the Log Backups of the Source System

In our example, the log backups of the source system SP1 are written to the default location /hana/shared/SP1/HDB00/backup/log. To make the volume accessible from the target system, a FlexClone volume of the SP1_shared volume is created and mounted at the target server.

2. Go to the volume view of the SVM where the SP1_shared volume is located.
3. Select the volume and select Create FlexClone Volume.
4. Provide a name for the FlexClone volume and click Clone.

5. Mount the FlexClone volume to the namespace.
6. Log on to the target server and mount the FlexClone volume.

   stlrx300s8-7: # mount -t nfs 192.168.173.102:/SP1_shared_clone /tmp/SP1-shared-clone

Recovery of the System Database

1. Within SAP HANA Studio, select the system database and select Recover System Database.

   Select Recover the Database to Its Most Recent State.
2. Select Recovery Using Backup Catalog.

3. Enter the backup catalog location of the system database.

Note: In our example, the mounted FlexClone volume of the source system is /tmp/SP1-shared-clone/shared/SP1/HDB00/backup/log/SYSTEMDB/

4. Select the backup catalog entry with the green icon.
5. Enter the log backup location.

**Note:** In our example, the mounted FlexClone volume of the source system is `/tmp/SP1-shared-clone/shared/SP1/HDB00/backup/log/SYSTEMDB`.

7. Click Finish to start the recovery of the system database.

8. The system database is recovered.
Recovery of the Tenant Database

1. Within SAP HANA Studio, select the system database and select Recover Tenant Database.

2. Select the tenant database.
3. **Select Recover the Database to Its Most Recent State.**

4. **Select Recovery Using Backup Catalog.**

5. **Input the backup catalog location of the system database.**

   **Note:** In our example, the mounted FlexClone volume of the source system is
   `/tmp/SP1-shared-clone/shared/SP1/HDB00/backup/log/DB_SP1/`
6. Click OK.

7. Select the backup catalog entry with the green icon.
8. Input the log backup location.

**Note:** In our example, the mounted FlexClone volume of the source system is /tmp/SP1-shared-clone/shared/SP1/HDB00/backup/log/DB-SP1/

10. Click Finish to start the recovery of the tenant database.

11. The tenant database is recovered.
7.3 SnapCenter Clone Create Workflow Using a Recovery Script

This chapter covers a save point. Figure 13 highlights the used workflow components.

Figure 13) SnapCenter clone operation using a recovery script.

1. Log on to SnapCenter and go to the resource topology view.
2. Select Local Copies and the backup that should be used.
   
   **Note:** The clone operation can also be performed from secondary storage, as described in the chapter “Clone from Secondary Storage.”

3. Click Clone from Backup.
4. Select the clone target server and provide the target server IP address to which the volume should be exported.

5. Enter the mount and post clone command.
   In our example, the SID = QP1, and the primary volume name = SP1_data_mnt00001.
sc-system-refresh.sh mount QP1 %SP1_data_mnt00001.Clone
sc-system-refresh.sh recover QP1

6. Click Next.

7. Click Finish to start the cloning workflow.
8. The Job Details screen shows the different steps that were executed.
Script Log File Output

20180108055215###stlrx300s8-7###sc-system-refresh.sh: Adding entry in /etc/fstab.
20180108055215###stlrx300s8-7###sc-system-refresh.sh: 192.168.173.102:/Sc277f998f-e1b2-4129-b09e-e0edd12a10 /hana/data/QP1/mnt00001 nfs
tw,vers=3,hard,timeo=600,rsize=1048576,wsize=1048576,intr,noatime,nolock 0 0
20180108055215###stlrx300s8-7###sc-system-refresh.sh: Mounting data volume.
20180108055215###stlrx300s8-7###sc-system-refresh.sh: mount /hana/data/QP1/mnt00001
20180108055215###stlrx300s8-7###sc-system-refresh.sh: Data volume mounted successfully.
20180108055215###stlrx300s8-7###sc-system-refresh.sh: Recover system database.
20180108055225###stlrx300s8-7###sc-system-refresh.sh: /usr/sap/QP1/HDB00/exe/python/bin/python
/usr/sap/QP1/HDB00/exe/python_support/recoverSys.py --command "RECOVER DATA USING SNAPSHOT CLEAR LOG"

[14049183022080, 0.081] >> starting recoverSys (at Mon Jan  8 05:52:35 2018)
[14049183022080, 0.081] keys: {'command': 'RECOVER DATA USING SNAPSHOT CLEAR LOG'}
own pid: 21783
recoverSys started: 2018-01-08 05:52:36
testing master: stlrx300s8-7
stlrx300s8-7 is master
shutdown database, timeout is 120
stop system
stop system: stlrx300s8-7
stopping system: 2018-01-08 05:52:36
stopped system: 2018-01-08 05:52:37
creating file recoverInstance.sql
restart database
restart master nameserver: 2018-01-08 05:52:42
start system: stlrx300s8-7
7.4 SnapCenter Clone Delete Workflow Using External Scripts

Figure 14) SnapCenter clone delete workflow.

1. Log on to SnapCenter and go to the resource topology view.
2. Select the clone.
3. Click Delete.
4. Enter the Pre Clone Delete command and the Unmount command.  
   In our example the SID = QP1:  
   sc-system-refresh.sh shutdown QP1  
   sc-system-refresh.sh unmount QP1  
   Click OK to start the deletion workflow.

5. The Job Details screen shows the different steps that were executed.
7.5 SnapCenter Clone Split Workflow

A Snapshot copy that is used as a source for a FlexClone copy is marked as busy on the storage layer and cannot be deleted. Therefore, SnapCenter cannot delete this Snapshot copy based on the defined
retention. Depending how long the SAP HANA target system is used, which is based on the FlexClone Copy, additional storage capacity is required to cover the block changes in the source Snapshot copy.

In addition, SnapCenter also does not delete the SAP HANA log backups that would be required to recover from this Snapshot backup. Therefore, additional space is also required in the log backup file system.

The SnapCenter clone split operation allows you to split the FlexClone from the source Snapshot copy. After the split operation, the Snapshot copy can be deleted again by SnapCenter, based on the defined retention.

**Note:** The SAP HANA system can be used during the clone split operation. There is no need to wait until the operation is finished.

1. Log on to SnapCenter and go to the resource topology view.
2. Select the clone.
3. Click Clone Split.

4. The next screen shows the required space for the volume split operation within the storage aggregate. Click Start.
5. The SnapCenter job monitor shows the progress of the clone split operation.
6. When the clone operation is finished, the clone is removed from the topology view of the source system.

Note: After a clone split operation, the storage volume is no longer managed by SnapCenter and therefore a delete operation using SnapCenter is not available.

7.6 SnapCenter Clone Create Workflow When the Cloned Volume Has Been Split

Figure 15 shows the workflow of the clone create operation using automation scripts for shutdown, unmount mount, and recovery.

Figure 15) SnapCenter clone create operation after a clone split operation.

After a clone split operation, the target system resource must be deleted within SnapCenter before a clone create operation can be executed.

1. Log on to SnapCenter and go to the resource view.
2. Select the clone resource.
3. Click the Delete button.

4. Click Yes to delete the resource.

Delete SP1 MDC single tenant -- NFS

The resource will be permanently deleted from SnapCenter Server if it is not protected and if there are no backups and resource groups associated with this resource.

Do you wish to proceed?

No  Yes

5. The resource is now deleted, and the clone create operation can be executed.
6. Go to the resource topology view.
7. Select Local Copies or Vault Copies and the backup that should be used.
8. Click Clone from Backup.

9. Select the clone target server and provide the target server IP address to which the volume should be exported.
10. Enter the required pre clone, mount, and post clone commands in their respective fields. The pre clone, shutdown, and unmount command must be separated by semicolons.

- The pre clone command is as follows:

  ```bash
  /mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh shutdown QP1; /mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh umount QP1
  ```

- The mount command is as follows:

  ```bash
  sc-system-refresh.sh mount QP1 %SP1_data_mnt00001_Clone
  ```

- The post clone command (recovery) is as follows:

  ```bash
  sc-system-refresh.sh recover QP1
  ```

11. Click Next.
12. Click Finish.

13. The progress of the operation can be observed in the Job Details window.
Job Details

Clone from backup 'SnapCenter_LocalSnap_Hourly_01-19-2018_05.00.02.6059'

✓ Clone from backup 'SnapCenter_LocalSnap_Hourly_01-19-2018_05.00.02.6059'
✓ sfdx3000-7.stl.netapp.com
✓ Clone
  ✓ Pre Clone Application
  ✓ Pre Clone Create Commands
  ✓ Storage Clone
  ✓ Mount Commands
  ✓ Post Clone Application
  ✓ Post Clone Create Commands
  ✓ Register Clone Metadata
  ✓ Application Clean-Up
  ✓ Data Collection
  ✓ Agent Finalize Workflow

Task Name: Clone Start Time: 1/19/2018 7:10:54 AM End Time: 1/19/2018 7:17:15 AM

View logs Close
**Note:** The original storage volume of the target system must be deleted manually on the storage layer.
20180119071053###stlrx300s8-7###sc-system-refresh.sh: Stopping HANA database.
20180119071053###stlrx300s8-7###sc-system-refresh.sh: sapcontrol -nr 00 -function StopSystem HDB

19.01.2018 07:10:54
StopSystem
OK
20180119071054###stlrx300s8-7###sc-system-refresh.sh: Wait until SAP HANA database is stopped

....
20180119071054###stlrx300s8-7###sc-system-refresh.sh: Status: GREEN
20180119071105###stlrx300s8-7###sc-system-refresh.sh: Status: GREEN
20180119071116###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071127###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071137###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071148###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071159###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071209###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071220###stlrx300s8-7###sc-system-refresh.sh: Status: GRAY
20180119071220###stlrx300s8-7###sc-system-refresh.sh: SAP HANA database is stopped.
20180119071223###stlrx300s8-7###sc-system-refresh.sh: Unmounting data volume.
20180119071223###stlrx300s8-7###sc-system-refresh.sh: Junction path: Scc6d7e583-54cc-4ab5-91d7-b92586478fc4
20180119071223###stlrx300s8-7###sc-system-refresh.sh: umount /hana/data/QPI/mnt00001.nfs
20180119071223###stlrx300s8-7###sc-system-refresh.sh: Deleting /etc/fstab entry.
20180119071223###stlrx300s8-7###sc-system-refresh.sh: Data volume unmounted successfully.
20180119071235###stlrx300s8-7###sc-system-refresh.sh: Adding entry in /etc/fstab.
20180119071235###stlrx300s8-7###sc-system-refresh.sh: Data volume mounted successfully.
20180119071235###stlrx300s8-7###sc-system-refresh.sh: Recover system database.
20180119071245###stlrx300s8-7###sc-system-refresh.sh: /usr/sap/QPI/HDB00/exe/Python/bin/python /usr/sap/QPI/HDB00/exe/python_support/recoverSys.py --command "RECOVER DATA USING SNAPSHOT CLEAR LOG"

[140446360319744, 0.075] >> starting recoverSys (at Fri Jan 19 07:12:55 2018)
[140446360319744, 0.075] args: ()
[140446360319744, 0.075] keys: {'command': 'RECOVER DATA USING SNAPSHOT CLEAR LOG'}

own pid: 4235
recoverSys started: 2018-01-19 07:12:56
testing master: stlrx300s8-7
stlrx300s8-7 is master
shutdown database, timeout is 120
stop system
stop system: stlrx300s8-7
stopping system: 2018-01-19 07:12:57
stopped system: 2018-01-19 07:12:57
creating file recoverInstance.sql
restart database
restart master nameserver: 2018-01-19 07:13:02
start system: stlrx300s8-7
2018-01-19T07:13:47-05:00 P004427 1610e5688ce INFO RECOVERY state of service: nameserver, stlrx300s8-7:30001, volume: 1, RecoveryPrepared

[140446360319744, 52.462] 0
[140446360319744, 52.462] << ending recoverSys, rc = 0 (RC_TEST_OK), after 52.386 secs
20180119071348###stlrx300s8-7###sc-system-refresh.sh: Wait until SAP HANA database is started

....
20180119071349###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071359###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071410###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071421###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071432###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071443###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071454###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071505###stlrx300s8-7###sc-system-refresh.sh: Status: YELLOW
20180119071515###stlrx300s8-7###sc-system-refresh.sh: Status: GREEN
### 7.7 Automation with PowerShell Scripts

Within the previous chapters, the different workflows have been executed using the SnapCenter UI. All the workflows can also be executed by using PowerShell scripts at the SnapCenter server, allowing further automation. The following chapters describe the basic PowerShell script examples for the following workflows.

- Create clone
- Delete clone
- Split clone
- Create clone after split

**Note:** The example scripts are provided as is and are not NetApp-supported products.

All scripts must be executed in a PowerShell command window. Before the scripts can be run, a connection to the SnapCenter server must be established using the `Open-SmConnection` command.

#### Create Clone Script

The following script can be used to execute the clone create workflow when the clone has not been split before. The parameters highlighted in bold must be adapted to your environment.

```powershell
$BackupName='SnapCenter_LocalSnap_Hourly_02-14-2018_09.00.02.2619'
$JobInfo=New-SmClone -AppPluginCode hana -BackupName $BackupName -Resources @{"Host"="SC30-V2.sapcc.stl.netapp.com";"UID"="SP1\SP1"} -CloneToInstance stlx300s8-7.stl.netapp.com -suffix "clone_to_QP1" -mountcommand '/mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh mount QP1 %SP1_data_mnt00001_Clone' -postclonecreatecommands '/mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh recover QP1' -NFSExportIPs 192.168.173.77
# Get JobID of clone create job
$Job=Get-SmJobSummaryReport | ?{$_._JobType -eq "Clone" } | ?{$_._JobName -Match $BackupName} | ?{$_._Status -eq "Running"}
$JobId=$Job.SmJobId
Get-SmJobSummaryReport -JobId $JobId
# Wait until job is finished
do { $Job=Get-SmJobSummaryReport -JobId $JobId; write-host $Job.Status; sleep 20 } while ( $Job.Status -Match "Running" )
Write-Host " Clone create job has been finished."
```

#### Delete Clone Script

The following script can be used to execute the delete workflow. The parameters highlighted in bold must be adapted to your environment.

```powershell
$CloneInfo=Get-SmClone |?{$_._CloneName -Match "clone_to_QP1" }
```
Automating SAP System Copies using the SnapCenter 4.0 SAP HANA Plug-In

$JobInfo=Remove-SmClone -CloneName $CloneInfo.CloneName -PluginCode hana -PreCloneDeleteCommands '/mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh shutdown QP1' -UnmountCommands '/mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh umount QP1' -Confirm: $False

Get-SmJobSummaryReport -JobId $JobInfo.Id

# Wait until job is finished
do { $Job=Get-SmJobSummaryReport -JobId $JobInfo.Id; write-host $Job.Status; sleep 20 } while ( $Job.Status -Match "Running" )
Write-Host " Clone delete job has been finished."

**Split Clone Script**

The following script can be used to execute the clone split workflow. The parameters highlighted in bold must be adapted to your environment.

```powershell
$CloneInfo=Get-SmClone |?{$_.CloneName -Match "clone_to_QP1" } Invoke-SmResourceSplit -CloneName $CloneInfo.CloneName -confirm:$false -Start

# Get JobID of clone split job
$Job=Get-SmJobSummaryReport | ?{$_.JobType -eq "CloneSplit" } | ?{$_.JobName -Match "clone_to_QP1" } | ?{$_.Status -eq "Running"};
Get-SmJobSummaryReport -JobId $JobId

# Wait until job is finished
do { $Job=Get-SmJobSummaryReport -JobId $JobId; write-host $Job.Status; sleep 20 } while ( $Job.Status -Match "Running" )
Write-Host " Clone split job has been finished."
```

**Create Clone After Split Script**

The following script can be used to execute the clone create workflow, when the clone has been split before. The parameters highlighted in bold must be adapted to your environment.

```powershell
$BackupName='SnapCenter_LocalSnap_Hourly_02-14-2018_09.00.02.2619'

Remove-SmResource -HostName stlrx300s8-7.stl.netapp.com -PluginCode hana -ResourceType MultipleContainers -SID SP1 -TenantDatabaseName SP1_clone_to_QP1 -Confirm:$false

New-SmClone -AppPluginCode hana -BackupName $BackupName -Resources @{"Host"="SC30-V2.sapcc.stl.netapp.com";"UID"="SP1\SP1"} -CloneToInstance stlrx300s8-7.stl.netapp.com -suffix 'clone_to_QP1' -PrecloneCreateCommands '/mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh shutdown QP1;/mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh umount QP1' -MountCommand '/mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh mount QP1 %SP1_data_mnt00001_Clone' - PostCloneCreateCommands '/mnt/hwval/System-Copy/SnapCenter/sc-system-refresh.sh recover QP1' - NFSExportIPs 192.168.173.77

# Get JobID of clone create job
$Job=Get-SmJobSummaryReport | ?{$_.JobType -eq "Clone" } | ?{$_.JobName -Match $BackupName} | ?{$_.Status -eq "Running"};
Get-SmJobSummaryReport -JobId $JobId

# Wait until job is finished
do { $Job=Get-SmJobSummaryReport -JobId $JobId; write-host $Job.Status; sleep 20 } while ( $Job.Status -Match "Running" )
Write-Host " Clone create job has been finshed."
```
8 SAP System Clone Workflow Description

Figure 16 summarizes the steps required to perform an SAP system clone workflow. This chapter covers a detailed description of the three main areas: the target server preparation; the clone and mount operations; and the fencing, recovery, and start operations.

8.1 Lab Setup used for SAP System Clone Operation

The following software release have been used in the lab setup.

- SAP HANA 2.0 SPS0 single container
  - SAP NetWeaver 7.4
  - SLES 12 SP1
  - SnapCenter 4.0
  - ONTAP 9.1

Figure 11 shows the components and configuration of the lab setup.

The SAP HANA database P01 (single container) and the SAP NetWeaver PNW system were used as the source for the system clone operation. The SnapCenter SAP HANA Plug-In was deployed on the target server. SnapCenter backups were configured for the P01 SAP HANA database and for the non-database volumes of the SAP HANA database and the SAP NetWeaver system.

In this chapter, the SAP system clone workflow is described based on clones at the primary storage. Cloning from secondary storage is also possible. The SAP HANA system was connected through NFS to the storage systems. The setup has been performed according to the configuration guide SAP HANA on NetApp All Flash FAS Systems with NFS.
8.2 Preparation of Target Server

This section describes the preparation steps required at the target server that is used for the SAP system clone operation.

SnapCenter HANA Plug-In Installation

To use automation scripts with SnapCenter, the SAP HANA plug-in must be deployed at the target host. More details on the deployment options of the SAP HANA plug-in can be found at SAP HANA Backup and Recovery with SnapCenter.

Installation of SAP Host Agent Software

The SAP host agent software must be installed at the target server. For more information, see the SAP help portal SAP Host Agent.

Configuration of Users, Ports, and SAP Services

The required users and groups for the SAP HANA database and the SAP system must be available at the target server. Typically, central user management is used; therefore, no configuration steps are necessary at the target server. The required ports for the SAP system must be configured at the target hosts. The configuration could be copied from the source system by copying the /etc/services file to the target server.

The required SAP services entries must be available at the target host. The configuration could be copied from the source system by copying the /usr/sap/sapservice file to the target server. The following output shows the required entries for the SAP HANA database and the SAP NetWeaver system used in the lab setup:

```
stirx300s8-5:/etc # cat /usr/sap/sapservices
#!/bin/sh
```

© 2018 NetApp, Inc. All Rights Reserved.
LD_LIBRARY_PATH=/usr/sap/P01/HDB02/exe:$LD_LIBRARY_PATH;export
LD_LIBRARY_PATH=/usr/sap/P01/HDB02/exe/sapstartsrv
pf=/usr/sap/P01/SYS/profile/P01_HDB02_stlrx300s8-3 -D -u p01adm
LD_LIBRARY_PATH=/usr/sap/PNW/ASCS04/exe:$LD_LIBRARY_PATH;export
LD_LIBRARY_PATH=/usr/sap/PNW/ASCS04/exe/sapstartsrv
pf=/usr/sap/PNW/SYS/profile/PNW_ASCS04_stlrx300s8-3 -D -u pnwadm
LD_LIBRARY_PATH=/usr/sap/PNW/DVEBMGS03/exe:$LD_LIBRARY_PATH;export
LD_LIBRARY_PATH=/usr/sap/PNW/DVEBMGS03/exe/sapstartsrv
pf=/usr/sap/PNW/SYS/profile/PNW_DVEBMGS03_stlrx300s8-3 -D -u pnwadm

Note: The services are started in a later step described in the chapter “Start SAP Services”.

Creation of SAP HANA Log Volume
To create an SAP HANA log volume, complete the following steps:

1. Create a log volume at the storage SVM for the SAP HANA database and mount it at the target server.

   ```
stlrx300s8-5:/hana/log/P01/mnt00001 
   # df
   Filesystem 1K-blocks Used Available Use% Mounted on
   /devmapper/360030057013a54201aa3727d258af07f-part2 285630984 61409096 223060620 22% /
   devtmpfs 66033204 8 66033196 1% /dev
   tmpfs 66042616 0 66042616 0% /dev/shm
   tmpfs 66042616 10364 66032252 1% /run
   tmpfs 66042616 0 66042616 0% /sys/fs/cgroup
   192.168.173.103:/hwval 498073600 469960896 28112704 95% /mnt/hwval
   192.168.173.104:/System.Clone_P01_log_mnt00001 134217728 192 134217536 1%
   /hana/log/P01/mnt00001
   stlrx300s8-5:/hana/log/P01/mnt00001 
   #
   ```

2. Create the required subdirectories within the mounted log volume.

   ```
stlrx300s8-3:~ # cd /hana/log/P01/mnt00001/
   stlrx300s8-3:~ # mkdir hdb00001
   stlrx300s8-3:~ # mkdir hdb00002
   stlrx300s8-3:~ # mkdir hdb00003
   stlrx300s8-3:~ # mkdir hdb00004
   stlrx300s8-3:~ # cd ..
   ```

Note: For an SAP HANA MDC single-tenant system, the file system structure is different than for this example of a single-container system. Therefore, you must adapt the commands accordingly. The required file system structure must be checked at the source system.

Preparation of Mount Points
Table 2 shows the required mount points for the lab setup used. The mount points must be created before the clone workflow is executed with SnapCenter.

Table 2) Required mount points.

<table>
<thead>
<tr>
<th>Description</th>
<th>Mount Point</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP HANA P01 data volume</td>
<td>/hana/data/P01/mnt00001</td>
</tr>
<tr>
<td>SAP HANA P01 user home directory</td>
<td>/usr/sap/P01</td>
</tr>
<tr>
<td>SAP HANA P01 binaries</td>
<td>/hana/shared/P01</td>
</tr>
<tr>
<td>SAP HANA P01 log backup</td>
<td>/mnt/log_backup</td>
</tr>
<tr>
<td>SAP NetWeaver PNW binaries</td>
<td>/usr/sap/PNW</td>
</tr>
<tr>
<td>SAP NetWeaver PNW binaries</td>
<td>/sapmnt/PNW</td>
</tr>
</tbody>
</table>
### Description

| SAP NetWeaver PNW binaries | /usr/sap/trans |

### 8.3 Cloning and Mount Operations

#### SnapCenter Cloning of Nondatabase Volumes

To clone nondatabase volumes, complete the following steps:

1. Log on to SnapCenter and go to the resource topology view for the nondata volumes.
2. Select Local Copies and the backup to use.

**Note:** The clone operation can also be performed from a secondary storage, as described in the chapter “Clone from Secondary Storage.”

3. Click Clone from Backup.

4. Select the clone target server and provide the target server IP address to which the volume should be exported.
5. Enter the mount command. In our example, the primary volume name = hana_shared.

   `sc-mount-volume.sh %hana_shared_Clone`

6. Click Finish.
7. When the operation is finished, the clone is shown in the topology view.

8. On the target server, the FlexClone copy of the hana/shared volume is now mounted.

```
stlr300s8-5:$ df -h
Filesystem  Size  Used  Avail Use% Mounted on
/dev/mapper/360030057013a54201aa3727d258af07f-part2  273G  59G  213G  22% /
devtmpfs  63G  8.0K  63G  1% /dev
```
9. Perform the same steps for all other nondatabase volumes. The mount command syntax for the other volumes with our example is as follows:

- `sc-mount-volume.sh %P01_usr_sap_Clone /usr/sap/P01`
- `sc-mount-volume.sh %PNW_sapmnt_Clone /sapmnt/PNW`
- `sc-mount-volume.sh %trans_Clone /usr/sap/trans`
- `sc-mount-volume.sh %PNW_usr_sap_Clone /usr/sap/PNW`

10. After the execution of the SnapCenter clone workflow, all nondatabase volumes are mounted at the target server.

11. SnapCenter creates new resources for all cloned nondata volumes.
SnapCenter Cloning of Database Data Volume

1. Log on to SnapCenter and go to the resource topology view.
2. Select Local Copies and the backup to use.

   **Note:** The clone operation can also be performed from a secondary storage, as described in the chapter “Clone from Secondary Storage”.
3. Click Clone from Backup.

4. Select the clone target server and provide the target server IP address to which the volume should be exported.
5. Enter the mount command. In our example, the SID = P01 and the primary volume name = P01_data_mnt00001.

   `sc-system-refresh.sh mount QP1 %P01_data_mnt00001_Clone`

6. Click Finish to start the cloning workflow.
When the clone workflow is finished, the SAP HANA data volume is mounted at the target server.

Cloning Log Backup Volume

With our example, the log backups of the source system P01 are written to a storage volume with the name log_backup. To make the volume accessible from the target system, a FlexClone volume of the log_backup volume is created and mounted at the target server.

To clone a log backup volume, complete the following steps:

2. Go to the volume view of the SVM where the log_backup volume is located.
3. Right-click the volume, and then select Clone > Create > Volume.
4. Provide a name for the FlexClone volume and click Clone.

5. Mount the FlexClone volume to the namespace.
6. The volume can now be mounted at the target server.

```
stlrx300s8-5:~ # mount -t nfs 192.168.173.102:/log_backup_clone /mnt/log_backup
stlrx300s8-3:/hana/data/P01/mnt00001/hdb00001 # df -h
```

```
<table>
<thead>
<tr>
<th>Filesystem</th>
<th>Size</th>
<th>Used</th>
<th>Avail</th>
<th>Use%</th>
<th>Mounted on</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/mapper/360030057013a54201aa3727d258af07f-part2</td>
<td>273G</td>
<td>59G</td>
<td>213G</td>
<td>22%</td>
<td>/</td>
</tr>
<tr>
<td>devtmpfs</td>
<td>63G</td>
<td>8.0K</td>
<td>63G</td>
<td>1%</td>
<td>/dev</td>
</tr>
<tr>
<td>tmpfs</td>
<td>63G</td>
<td>0</td>
<td>63G</td>
<td>0%</td>
<td>/dev/shm</td>
</tr>
<tr>
<td>tmpfs</td>
<td>63G</td>
<td>11M</td>
<td>63G</td>
<td>1%</td>
<td>/run</td>
</tr>
<tr>
<td>tmpfs</td>
<td>63G</td>
<td>0</td>
<td>63G</td>
<td>0%</td>
<td>/sys/fs/cgroup</td>
</tr>
<tr>
<td>192.168.173.102:/hwval</td>
<td>475G</td>
<td>449G</td>
<td>27G</td>
<td>95%</td>
<td>/mnt/hwval</td>
</tr>
<tr>
<td>192.168.173.102:/System_Clone_P01_log_mnt00001</td>
<td>128G</td>
<td>2.2G</td>
<td>126G</td>
<td>2%</td>
<td>/hana/log/P01/mnt00001</td>
</tr>
<tr>
<td>192.168.173.102:/Scac7db9e6-e5df-435c-bdc3-2b380fb91282</td>
<td>1.0T</td>
<td>451G</td>
<td>574G</td>
<td>44%</td>
<td>/hana/shared</td>
</tr>
<tr>
<td>192.168.173.102:/Sc5e793682-5862-413a-81c4-1134a2306c76</td>
<td>50G</td>
<td>4.0M</td>
<td>50G</td>
<td>1%</td>
<td>/usr/sap/P01</td>
</tr>
<tr>
<td>192.168.173.102:/Scac7db9e6-e5df-435c-bdc3-2b380fb91282</td>
<td>50G</td>
<td>845M</td>
<td>50G</td>
<td>2%</td>
<td>/tmpfs/PNW</td>
</tr>
<tr>
<td>192.168.173.102:/Sc4ff5e73d-0cba-4056-86e9-7da069a3c73f</td>
<td>50G</td>
<td>4.0M</td>
<td>50G</td>
<td>1%</td>
<td>/usr/sap/trans</td>
</tr>
<tr>
<td>192.168.173.102:/Sc715e1e0e-3cd5-4e6a-b539-3ee0df77375c</td>
<td>150G</td>
<td>28G</td>
<td>123G</td>
<td>19%</td>
<td>/usr/sap/PNW</td>
</tr>
<tr>
<td>192.168.173.102:/Scbf07b59f-b930-49f2-90a3-2790dacc832e</td>
<td>2.0T</td>
<td>33G</td>
<td>2.0T</td>
<td>2%</td>
<td>/mnt/log_backup</td>
</tr>
</tbody>
</table>
```

8.4 Fencing, Recovery and Start Operations

**Fencing of the Target System**

Before you perform any recovery or start operations, you must establish proper fencing of the target server so that it cannot communicate with other systems. If proper fencing is not in place, then the cloned production system might exchange data with other production systems, resulting in logically corrupted data.
Target Server Host Name and IP Address
The host name of the target server must be identical to the host name of the source system. The IP address can be different, however. If the source SAP system and the HANA database have been installed enabled for adaptive computing, then the virtual host names for the SAP and SAP HANA services must be identical to the virtual host names of the source production system.

Start SAP Services
The SAP services are started using the command `service sapinit start` at the target server.

Database Recovery with SAP HANA Studio
The SAP HANA database can now be recovered with SAP HANA Studio or Cockpit. The following chapter shows the recovery process using SAP HANA Studio.

**Note:** The following screenshots show the recovery for the single container database P01. An SAP HANA MDC single-tenant database can be recovered in two steps: recovery of the system database followed by recovery of the tenant database.

To recover a database, complete the following steps:

1. Within SAP HANA Studio, select the database and select Recover System.

2. Depending on the use case for the SAP system clone, different recovery options are available. In our example, we perform a point-in-time recovery.
3. Provide the path to the backup catalog, which, in our example, is the path to the mounted FlexClone copy of the log backup volume from the source system. Click Next.

4. The Snapshot backup image, which has been made available with the mounted FlexClone copy, is shown with a green icon. Select the backup and click Next.
5. Provide the path to the log backups, which, in our example, is the path to the mounted FlexClone copy of the log backup volume from the source system. Click Next.

6. Select Initialize Log Area, because the recovery is only performed using log backups.
7. Click Finish to start the recovery process.

8. The recovery is finished, and the SAP HANA database is started.
Starting SAP Application

1. The SAP application can now be started using the `startsap` command at the target server.

```
stlrx300s8-3:/hana/data/P01/mnt00001/hdb00001 # su - pnwadm
stlrx300s8-3:pnwadm 52> startsap
Checking HDB Database
Database is running

Starting Startup Agent sapstartsrv
OK
Instance Service on host stlrx300s8-3 started

starting SAP Instance ASCS04
Startup-Log is written to /home/pnwadm/startsap ASCS04.log

/usr/sap/PNW/ASCS04/exe/sapcontrol -prot N1_HTTP -nr 04 -function Start
Instance on host stlrx300s8-3 started
Starting Startup Agent sapstartsrv
OK
Instance Service on host stlrx300s8-3 started

starting SAP Instance DVEBMGS03
Startup-Log is written to /home/pnwadm/startsap DVEBMGS03.log

/usr/sap/PNW/DVEBMGS03/exe/sapcontrol -prot N1_HTTP -nr 03 -function Start
Instance on host stlrx300s8-3 started
```

2. Log on to the SAP system.
9 Conclusion

SnapCenter storage cloning and the ability to flexibly define precloning and postcloning operations allows SAP Basis administrators to accelerate and automate SAP system copy, clone, or refresh operations. You can choose any SnapCenter Snapshot backup in either primary or secondary storage locations, which allows you to address your most important use cases, such as logical corruption, DR testing, or the refresh of an SAP QA system.
Appendix

The scripts sc-system.sh and sc-mount-volume.sh are described in the chapter “Example Automation Scripts” and are used within the different workflows in this document.

Example Script

```bash
#!/bin/bash
VERBOSE=NO
MY_NAME="basename $0"
BASE_SCRIPT_DIR="dirname $0"

# Generic parameters
MOUNT_OPTIONS="rw,vers=3,hard,timeo=600,rsize=1048576,wsize=1048576,intr,noatime,nolock"

# log file writer
#################################
write2log(){
  TEXT=$1
  echo -n \`date +%Y%m%d%H%M%S` >> $LOGFILE
  echo -n "###" >> $LOGFILE
  echo -n `hostname` >> $LOGFILE
  echo "###$MY_NAME: $TEXT" >> $LOGFILE
  if [ $VERBOSE = YES ]; then
    echo -n \`date +%Y%m%d%H%M%S`
    echo -n "###"
    echo -n `hostname`
    echo "###$MY_NAME: $TEXT"
  fi
}

# Usage
#################################
usage(){
  echo ""
  echo "Usage:"
  echo "Operation mount : $MY_NAME mount <SID> <SnapCenter_Junction_Path_Parameter>"
  echo "Operation umount : $MY_NAME umount <SID>"
  echo "Operation recover : $MY_NAME recover <SID>"
  echo "Operation shutdown: $MY_NAME shutdown <SID>"
}

# Get config file
#################################
get_config_file(){
  CONFIG_FILE="$BASE_SCRIPT_DIR/sc-system-refresh-$SID.cfg"
  if [ ! -e $CONFIG_FILE ]; then
    echo "Config file $CONFIG_FILE does not exist."
    RET=1
  else
    source $CONFIG_FILE
    MOUNT_POINT="/hana/data/$SID/mnt00001"
    HDBNO="HDB$INSTANCENO"
    SIDLOW=`echo $SID | tr [:upper:] [:lower:]`
    SIDADM="$SIDLOW"adm"
    LOGFILE="$BASE_SCRIPT_DIR/sc-system-refresh-$SID.log"
    if [ ! -e $LOGFILE ]; then
      touch $LOGFILE
      chmod 777 $LOGFILE
      fi
      RET=0
      fi
    fi
  fi
```
# Umount operation

# Umount operation

```bash
umount_data_volume()
{
    write2log "Unmounting data volume."
    JUNCTION_PATH=`df | grep /hana/data/"$SID"/mnt00001 | awk '{print $1}' | awk -F / '{print $2}'`
    write2log "Junction path: $JUNCTION_PATH"
    $UMOUNT_CMD=
    write2log "$UMOUNT_CMD"
    $UMOUNT_CMD
    RET=$?
    if [ $RET -gt 0 ]
    then
        write2log "Unmount operation failed."
    else
        write2log "Deleting /etc/fstab entry."
        sed -i "/$JUNCTION_PATH/d" /etc/fstab
        write2log "Data volume unmounted successfully."
    fi
}

# Get status of HANA database

get_status_of_HANA_database

wait_for_status()
{
    EXPECTED_STATUS=$1
    TIME_OUT=$2
    CHECK=1
    COUNT=0
    while [ $CHECK = 1 ]; do
        CMD="sapcontrol -nr $INSTANCENO -function GetSystemInstanceList | grep HDB | awk -F , {'print \"$7\"'}"
        STATUS=`su - $SIDADM -c "$CMD"`
        write2log "Status: $STATUS"
        if [ $STATUS = $EXPECTED_STATUS ]
        then
            CHECK=0
            RET=0
        else
            sleep 10
            COUNT=`expr $COUNT + 1`
            if [ $COUNT -gt $TIME_OUT ]
            then
                RET=1
                break
            fi
        fi
    done
}

# Stop HANA database

stop_hana_db()

wait_for_stop()
{
    CMD="sapcontrol -nr $INSTANCENO -function StopSystem HDB"
    write2log "$CMD"
    su - $SIDADM -c "$CMD >> $LOGFILE 2>&1"
    write2log "Wait until SAP HANA database is stopped ...."
    waiting_for_status GRAY $TIME_OUT_STOP
    if [ $RET -gt 0 ]
    then
        write2log "Timeout error: SAP HANA database not stopped configured timeout."
        echo "Timeout error: SAP HANA database not stopped within configured timeout."
    RET=1
    else
        write2log "SAP HANA database is stopped."
    RET=0
    fi
}
```
Mount operation

```
mount_data_volume() {
  write2log "Adding entry in /etc/fstab."  
  LINE="$STORAGE":"$JUNCTION_PATH $MOUNT_POINT nfs $MOUNT_OPTIONS 0 0"
  write2log "$LINE"
  echo "$LINE" >> /etc/fstab
  write2log "Mounting data volume."  
  MOUNT_CMD="mount $MOUNT_POINT"
  write2log "$MOUNT_CMD"
  $MOUNT_CMD
  RET=$?
  if [ $RET -gt 0 ]
    then
    write2log "Mount operation failed."  
  else
    write2log "Data volume mounted successfully."  
  fi
}
```

Recovery operation

```
recover_database() {
  if [ $HANA_ARCHITECTURE="MDC_single_tenant" ]
    then
    write2log "Recover system database."  
  else
    write2log "Recover single container database."  
  fi

  SQL="RECOVER DATA USING SNAPSHOT CLEAR LOG"
  RECOVER_CMD="/usr/sap/$SID/$HDBNO/exe/Python/bin/python/usr/sap/$SID/$HDBNO/exe/python_support/recoverSys.py --command "$SQL"
  write2log "$RECOVER_CMD"
  su - $SIDADM -c "$RECOVER_CMD >> $LOGFILE 2>&1"
  write2log "Wait until SAP HANA database is started ...."
  wait_for_status GREEN $TIME_OUT_START
  if [ $RET -gt 0 ]
    then
    write2log "Timeout error: SAP HANA database not started within configured timeout."  
    echo "Timeout error: SAP HANA database not started within configured timeout."  
    RET=1
  else
    write2log "SAP HANA database is started."  
    RET=0
  fi

  if [ $HANA_ARCHITECTURE="MDC_single_tenant" -a $RET -eq 0 ]
    then
    write2log "Recover tenant database "$SID"."  
    SQL="RECOVER DATA FOR $SID USING SNAPSHOT CLEAR LOG"
    RECOVER_CMD="/usr/sap/$SID/$HDBNO/exe/hdb/hdbsql -U $KEY $SQL"
    write2log "$RECOVER_CMD"
    su - $SIDADM -c "$RECOVER_CMD >> $LOGFILE 2>&1"
    write2log "Checking availability of Indexserver for tenant "$SID"."  
    CMD="sapcontrol -nr $INSTANCENO -function GetProcessList | grep Indexserver-$SID | awk -F , '{print $3}"
  RETSTR=`su - $SIDADM -c "$CMD"
  RETSTR=`echo $RETSTR | sed 's/" */"g'`
  if [ "$RETSTR" = "GREEN" ]
    then
    write2log "Recovery of tenant database $SID failed."  
    echo "Recovery of tenant database $SID failed."  
    write2log "Status: $RETSTR"
    RET=1
  fi

```

else
    write2log "Recovery of tenant database $SID succesfully finished."
    write2log "Status: $RETSTR"
    RET=0
    fi
fi

# Main
#################################################
if [ "$1" ]
then TASK=$1
    if [ $TASK != "mount" -a $TASK != "umount" -a $TASK != "recover" -a $TASK != "shutdown" ]
    then
        echo "Wrong operation parameter. Must be mount, umount, recover or shutdown."
        usage
        exit 1
    fi
else
    echo "Wrong operation parameter."
    usage
    exit 1
fi

if [ "$2" ]
then
    SID=$2
    get_config_file
    if [ $RET -gt 0 ]
    then
        exit 1
    fi
else
    echo "Wrong Parameter: SID must be provided."
    usage
    exit 1
fi

case $TASK in
    "shutdown")
        stop_hana_db
        exit $RET
    ;;
    "umount")
        umount_data_volume
        exit $RET
    ;;
    "recover")
        recover_database
        exit $RET
    ;;
    "mount")
        if [ "$3" ]
        then
            JUNCTION_PATH=$3
            mount_data_volume
            exit $RET
        else
            echo "Wrong parameter: Junction path must be provided for mount operation."
            usage
            exit 1
        fi
    ;;
esac
Configuration file example sc-system-refresh-QP1.cfg

```bash
# Configuration file example sc-system-refresh-QP1.cfg

# Target database specific parameters
# -----------------------------------------------

# HANA_ARCHITECTURE "single_container" or "MDC_single_tenant"
HANA_ARCHITECTURE="MDC_single_tenant"

# hdbuserstore key, which should be used to connect to the target database
KEY="QP1KEY"

# Timeout to start the database in 10sec intervals
# E.g. 3min = 180sec, TIME_OUT_START=18
TIME_OUT_START=18

# Timeout to stop the database in 10sec intervals
# E.g. 3min = 180sec, TIME_OUT_STOP=18
TIME_OUT_STOP=18

# Instance number of the target database
INSTANCENO="00"

# Data LIF of the SVM, which should be used to mount the volume
STORAGE="192.168.173.102"
```

Example Script sc-mount-volume.sh

**Note:** The current version of the script uses a fixed IP address/host name to mount the volume from the storage system. This IP address/host name must be adapted within the script, or the script can be changed to get the IP address hostname as an additional command line parameter.

```bash
#!/bin/bash

VERBOSE=NO
MY_NAME="`basename $0`"
BASE_SCRIPT_DIR="`dirname $0`"
LOGFILE="$BASE_SCRIPT_DIR/sc-mount-volume.log"

# Generic parameters
MOUNT_OPTIONS="rw,vers=3,hard,timeo=600,rsize=1048576,wsize=1048576,intr,noatime,nolock"

# log file writer
write2log() {
  TEXT=$1
  echo -n `date +%Y%m%d%H%M%S` >> $LOGFILE
  echo -n "###" >> $LOGFILE
  echo -n `hostname` >> $LOGFILE
  echo "###$MY_NAME: $TEXT" >> $LOGFILE
  if [ $VERBOSE = YES ];
    then
      echo -n `date +%Y%m%d%H%M%S`
      echo -n "###"
      echo -n `hostname`
      echo "###$MY_NAME: $TEXT"
  fi
}

# Usage
usage() {
  echo "" 
  echo "Usage:" 
  echo "$MY_NAME <SnapCenter_Junction_Path_Parameter> <Mount point>"
}
```
# Mount operation

```
mount_volume()
{
    MOUNT_OPTIONS="-t nfs -o rw,vers=3,hard,timeo=600,rsize=1048576,wsize=1048576,intr,noatime,nolock"
    MOUNT_CMD="mount $MOUNT_OPTIONS $STORAGE":"$JUNCTION_PATH $MOUNT_POINT"
    write2log "Mounting volume $JUNCTION_PATH to $MOUNT_POINT"
    $MOUNT_CMD
    RET=$?
    if [ $RET -gt 0 ]
    then
        write2log "Mount operation failed."
        exit 1
    else
        write2log "Volume mounted successfully."
        exit 0
    fi
}
if [ "$1" ]
then
    JUNCTION_PATH=$1
else
    echo "Wrong parameter."
    usage
    exit 1
fi
if [ "$2" ]
then
    MOUNT_POINT=$2
else
    echo "Wrong Parameter."
    usage
    exit 1
fi
```

STORAGE="192.168.173.102"
mount_volume
exit $?
Where to Find Additional Information

To learn more about the information described in this document, refer to the following documents and/or websites:

- NetApp SnapCenter Software Resource page:
  http://mysupport.netapp.com/snapcenter/resources
- SAP Software Solutions product page:

Version History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Document Version History</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version 1.0</td>
<td>February 2018</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>
Refer to the Interoperability Matrix Tool (IMT) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer’s installation in accordance with published specifications.

Copyright Information

Copyright © 1994–2018 NetApp, Inc. All rights reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark Information

NETAPP, the NETAPP logo, and the marks listed at http://www.netapp.com/TM are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.

TR-4667-0318