NETAPP SOLUTION DEPLOYMENT GUIDELINES

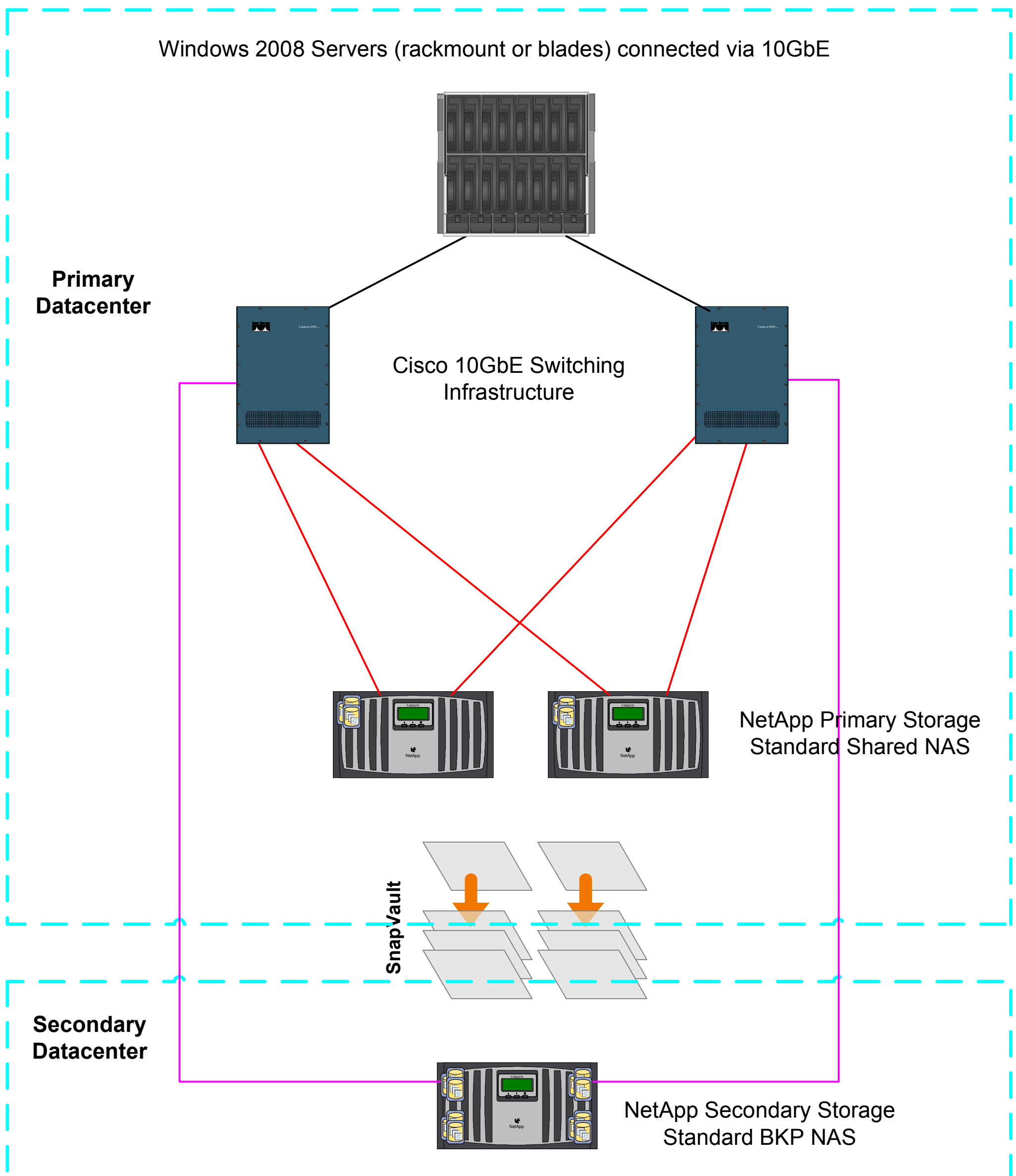
Thomson Reuters – SQL Server and iSCSI on NetApp

Prepared By:
Michael Arndt
arndt@netapp.com
Version 6: February 2014

Table of Contents

1 Executive Summary

This document outlines the deployment guidelines for the SQL Server and NetApp SnapManager for SQL Server (SMSQL) environment in Thomson Reuters Professional using NetApp storage. SQL Server 2008 on Windows 2008 will be deployed with NetApp storage to implement an improved backup and recovery architecture using NetApp snapshots and SnapVault. Storage for the SQL Server environment will be accessed via the iSCSI protocol, and will utilize the standard shared NAS environment already in place within each datacenter at Thomson Reuters Professional. While most of this document is tailored for the SQL environment, it is equally applicable to Windows Servers with iSCSI LUNs that are not running SQL, and differences between a SQL and non-SQL configuration will be called out as appropriate. This document is not a step by step guide for implementation, but rather it will provide environment specific details surrounding each component of the solution and a high level description of the backup / recovery use cases that this configuration covers for the TRP SQL Server environment.



2 NetApp storage system configuration and provisioning

2.1 Storage system requirements

Storage for Microsoft SQL servers should be deployed on the latest standard high tier shared NAS environment at Thomson Reuters Professional. As of the writing of this document, that consists of a FAS6080 storage cluster with 300GB 15k RPM drives, running Data ONTAP 7.3.3P3, and using vFilers for all provisioning.

2.2 vFiler configuration and Data Motion considerations

When configuring vFilers for use with iSCSI LUNs, the following configuration options and limitations must be taken into consideration:

- Each vFiler will be configured with a local account, named *iscsi*, used for SnapDrive communication via HTTP between the SQL Server and the vFiler. A script developed by TRP will be used to automate the creation of this account for each vFiler. This account will have a unique password for each primary vFiler, and a common password for each shared SnapVault secondary vFiler. The account should be created with the a role that has the following capabilities:

    login-http-admin,login-ssh,cli-passwd,api-lun-,api-snapshot-*,api-iscsi-*,api-volume-get-*,api-volume-list-*,api-volume-container,api-volume-size,api-volume-options-list-*,api-volume-set-option,api-volume-autosize-get,api-snapvault-*,api-snapmirror-get-*,api-ems-*,api-igroup-*,api-qtree-*,api-license-*,api-system-*,api-aggr-list-*,api-aggr-options-list-*,api-aggr-get-*,api-file-get-*,api-file-read-*,api-file-list-*,api-options-get,api-snapmirror-list-**

  The above list of capabilities does not include the ability to use FlexClone functionality, since that requires the capability to destroy volumes as well. This will work in the TRP environment because FlexClone is only licensed on SnapVault secondary systems, and SnapDrive uses LUN clones when connecting to LUNs on a SnapVault destination storage system. This list of capabilities would need to be updated if FlexClone was licensed on primary storage systems in the TRP environment.
- Each vFiler, and all SnapVault secondary backup vFilers, must have an account named *iscsisv*. This account will be used for communication during the SnapVault backup process, as well as for SnapShot pruning as described in section 2.6. This account must be configured to allow public key authentication from the *netapp* account on all Operations Manager servers, and should be created with a role that has the following capabilities:

    login-ssh,cli-snapvault,cli-snap**
- The *iscsi.ip_based_tpgroup* option must be enabled in order to support online migrations with Data Motion for vFilers.
- There are limitations on the maximum number of volumes allowed per vFiler when doing a Data Motion for vFilers migration. These limits are as follows:
    - FAS3xxx – maximum of 8 flexvols per vFiler.
    - FAS6xxx – maximum of 20 flexvols per vFiler.

2.3 Volume, Qtree, and LUN configuration

Each SQL Server deployment will require a minimum of 3 flexvols configured on a NetApp primary storage system in the standard high tier shared NAS environment, as show in the following table:

Storage Volume and Qtree Path	LUN Name	LUN Size	Volume Size	Snap Reserve	SnapVault?
/vol/<bu>_wi_<appname>_tsys_nosnap/luns	tmpsys	25% of DB	25% of DB sizes	0%	No
	quorum	1GB			
/vol/<bu>_wi_<appname>_usr_snap/luns	userdb	Custom	(2 * LUN sizes) + 1GB	50%	Yes
	userlog	Custom			
/vol/<bu>_wi_<appname>_info_snap/luns	snapinfo	10GB or 20GB	21GB	50%	Yes

Whether provisioning for SQL or non-SQL data, each volume will have a qtree in it named “luns”, since SnapVault requires the use of qtrees when replicating LUNs. The Volume Size is set slightly larger than the size of all the LUNs in the volume in order to accommodate a small amount of space required for LUN metadata and lun clones based on snapshots in the volumes.

The follow configuration settings apply for each volume in order to implement thin provisioning for the volumes. Note that thin provisioning should only be used when provisioning storage on an aggregate for which all other volumes are also thin provisioned.

- Set the volume guarantee to “none”.
- Set the volume option “fractional_reserve” to “0”
- Set the “try_first” volume option to “snap_delete”.
- Enable snap autodelete and set the snap autodelete trigger to “snap_reserve”.

The following configuration settings apply for each LUN:

- The LUN reservation should be enabled (which is the default setting).

2.4 Snap Reserve sizing and monitoring

For volumes on which snapshots will be taken, the Snap Reserve size listed in section 2.3 is 50%. The reasons for configuring volumes with iSCSI LUNs in this manner are as follows:

- Using a Snap Reserve of 50% instead of a Snap Reserve of 0% allows the storage administrator to limit the amount of space in the volume allocated to LUNs. This guarantees that a Windows or SQL administrator with access to SnapDrive cannot provision more capacity in their volumes than was allocated to them by the TRP storage team.
- Using a Snap Reserve of 50% means that we have an equal amount of space for LUNs and the snapshots of the LUNs. This guarantees that we can always have 1 snapshot of a volume without the possibility of running out of space due to snapshot consumption.

It should also be noted that snap autodelete is used to keep the Snap Reserve area from overflowing into the active filesystem portion of the volume. In the TRP environment, snap autodelete is configured to trigger at 80% of Snap Reserve utilization. In a worst case scenario, retaining a single snapshot could result in Snap Reserve utilization of above 80% (but less than 100%).

In order to manage these scenarios, an Operations Manager alert for the snapshot-full condition will be configured to page the TRP storage team when the Snap Reserve is at 80% utilization for at least 30 minutes (this time delay gives snap autodelete time to bring the utilization back down under 80% if possible). In situations where a single snapshot results in Snap Reserve utilization of over 80%, any newly created SMSQL or SnapDrive snapshots will be immediately removed by snap autodelete. Because of this, the proper response to a page for this condition is to temporarily (or permanently, for repeat offenders) grow the overall size of the volume.

2.5 Snapshot backup configuration on primary storage

Snapshot creation on primary storage will be driven completely by SMSQL for LUNs with SQL data, and by SnapDrive for LUNs with non-SQL data. All primary storage volumes used for LUNs should have a "*snap sched*" of "*0*" and there should not be a "*snapvault snap sched*" schedule in place.

2.6 SnapVault backup configuration on secondary storage

The SnapVault relationship for volumes with iSCSI LUNs will be initialized in a similar manner as any other SnapVault relationship. The only difference will be that we must replicate LUNs at the qtree level when using SnapVault, as opposed to replicating at the volume level. Therefore, the command to initialize a SnapVault relationship for a volume with iSCSI LUNs will be as follows:

> *vfiler run <Secondary_vFiler> snapvault start –S <primary_vFiler>:/vol/<volname>/luns /vol/<volname>/luns*

While most volumes accessed via NFS or CIFS have "*snapvault snap sched*" schedules in the TRP environment, volumes used for iSCSI LUNs have non-snapvault snapshots taken by SMSQL or SnapDrive. There will be no snapvault snapshot schedules put in place on primary or secondary storage volumes hosting iSCSI LUNs.

SnapVault updates are performed by a script that replicates SMSQL or SnapDrive created backup snapshots from primary storage to secondary storage. The script that is replicating the SMSQL or SnapDrive created backup snapshots is named "*lun_snapvault_update.pl"*, and will be run nightly from a cron job on each Operations Manager server. This script will create identically named snapshots on the SnapVault secondary storage system, with the exception that the snapshots will have a "*sv_"* prefix on the SnapVault secondary.

Another script will be used to perform pruning of snapshots on the SnapVault secondary storage systems, as well as pruning of snapshots on primary volumes used for LUNs with non-SQL data (SMSQL manages snapshot retention on primary storage for LUNs with SQL data). This script is named "*lun_snapvault_cleanup.pl*", and will be run nightly from a cron job on each Operations Manager server. On the SnapVault secondary storage system, this script will remove any snapshots for volumes containing vaulted LUNs that are older than the number of days specified in the secondary volume naming convention of "*sv_<num>_volname*". On primary storage systems, this script will operate on volumes used for LUNs with non-SQL data (as denoted by a "*wid_*" or *"wip_"* in the volume name), and any snapshots older than 7 days will be removed.

2.7 Aggregate over-provisioning and capacity alerting

As of the writing of this document, the plan is to over-provision aggregates by a factor of 2 to 1. For example, an aggregate with 10TB of phsyiscal space would have 20TB of thin provisioned volumes allocated on it, with 10TB assigned to the active filesystems for LUNs, and 10TB assigned to volume Snap Reserves for holding snapshot copies of data.

The use of thin provisioned volumes with a 50% Snap Reserve is useful for avoiding out of space conditions on volumes with LUNs that are using snapshots. This configuration also allows us to over-provision the aggregate so that we can drive up overall storage utilization. This configuration, along with a 2 to 1 over-provisioning scheme, also allows us to guarantee that we will always have enough space in an aggregate to hold all of the LUNs, even in a worst case scenario of having to delete all volume snapshots in order to accomplish this. Note that we do not expect to ever hit this worst case scenario, as action should be taken before we get to this point.

Since we are using thin provisioned volumes, and managing space at the aggregate level, we must use different capacity alerting metrics than fully provisioned volumes would use. NetApp Operations Manager should be configured to set the Aggregate Full Threshold to 70% and the Aggregate Nearly Full Threshold to 60% for aggregates used with thin provisioned volumes. These settings can be overridden from the global default values by navigating to **Global -> Member Details -> Aggregates -> <aggr name> -> Aggregate Tools Edit Settings** in the Operations Manager web interface. Operations Manager alarms should then be configured to notify appropriate personnel when these thresholds are breached.

2.8 SQL Server log backups on a NetApp vFiler with compression

The initial implementation of SQL Server on NetApp used non-NetApp storage to store and compress SQL Server log backups. As of Data ONTAP 8.1, NetApp also supports data compression. In environments where NetApp will be used to store and compress SQL Server log backups, the following standards should be followed:

- There will be a shared vFiler in each datacenter module designated for SQL Server log file backups.
- The vFiler will be stored on a low tier shared primary storage system that is part of a HA pair.
- Each SQL Server instance will get it's own qtree on a shared volume in the vFiler.
- The volume(s) storing the SQL Server log backups will use the background compression method, not inline compression.
- The qtrees will be shard out via CIFS, and the CIFS shares will be accessed from the SQL Servers directly in order to perform log backups.
- The TR SQL Server DBA team will be responsible for managing the log backup retention and removing old log backups.
- The log backups will not be replicated via SnapVault or SnapMirror, and SnapShots will not be used on the volume(s) storing SQL Server log backups.

The following commands demonstrate how to configure background compression on a given volume of a vfiler. In this example, we set compression to run at 1am every night:

vfiler run vfilername sis on /vol/volname
vfiler run vfilername sis config –C true /vol/volname
vfiler run vfilername sis config -s sun-sat@1 /vol/volname
vfiler run vfilername sis config

In order to see the compression savings of a given volume, the command "*df –S volname*" can be used.

3 SQL Server Configurations

3.1 LUN layout guidelines

The SQL Server instances will use a number of iSCSI connected LUNs to store their data. The following table describes the standard LUN layout and how it maps to the volume layout on the NetApp storage system.

LUN Description	Mountpoint or Drive Letter	Storage Volume and Qtree Path	LUN Name
SystemDBs and TempDB	C:\<blah>\SystemDBs or H:\	/vol/<bu>_wi_<appname>_tsys_nosnap/luns	tmpsys
Quorom (*)	Q:\		quorum (* MSCS only)
UserDB data files	C:\<blah>\UserDB1 or E:\	/vol/<bu>_wi_<appname>_usr_snap/luns	userdb
UserDB log files	C:\<blah>\UserLog1 or G:\		userlog
SMSQL SnapInfo	C:\<blah>\SnapInfo or S:\	/vol/<bu>_wi_<appname>_info_snap/luns	snapinfo

The following guidelines will be used when multiple pairs of LUNs for UserDB data files and UserDB log files is required due to the number of User databases. One layout consideration that applies to all deployment scenarios is that a larger database (over 200GB in size) is recommended to reside on a dedicated pair of LUNs, and in a dedicated volume, for all data files and log files. This is to ensure that we can perform fast restores of large databases using the "*lun clone split*" methodology. When databases share LUNs, the only restore option available is a file copy based restore. While file copy times can vary, we will assume a file copy speed of 100GB per hour for restore purposes. With this value in mind, a User database that is over 200GB in size would take over 2 hours for a restore if it resided on shared LUNs with other databases. User databases over 200GB in size should only be placed on shared LUNs if the application can tolerate the RTO that comes with a 100GB per hour file restore process.

3.1.1 Legacy environments running SnapDrive 6.3 and SMSQL 5.1

The following details are from the legacy standards when SnapDrive 6.3 and SMSQL 5.1 were in use. These standards are no longer used for new deployments, but are maintained in this document in case there is a need to look at the old standards.

- Only 35 User Databases are allowed per LUN. This requirement is to ensure that the Microsoft VDI snapshot process completes in a timely manner during the SMSQL backup process, and that SMSQL creates only a single snapshot per volume during the backup process.
- Each User Database over 200GB should reside on a dedicated pair of LUNs if a fast RTO is required.
- We will impose a restriction that only 9 pairs of LUNs for UserDB data files and UserDB log files will be allowed per SQL instance. This is to ensure that we keep our LUN counts to a reasonable level to ensure timely SnapDrive LUN enumeration. The result of this restriction is that this design will only accommodate up to 9 * 35 = 315 User databases (all of which are under 200GB in size).

3.1.2 Environments with up to 1,400 databases

For environments that do not have a large number of databases, we can support up to 1,400 User databases per SQL Server when the following guidelines are followed along with using SMSQL 6.0 and SnapDrive 6.5:

- Up to 175 User databases can be placed on a pair of data / log LUNs.
- Each User Database over 200GB should reside on a dedicated pair of LUNs if a fast RTO is required.
- Up to 8 pairs of data / log LUNs can be used per SQL Server. This will accommodate up to 8 * 175 = 1,400 User databases.
- The SnapInfo drive will be 10GB in size.
- One snapshot per volume will be taken during each SMSQL backup process.
- The following registry keys must be set, in order to ensure that SMSQL only quiesces 175 databases at a time during the backup process. The first key is for stand alone SQL instances, and the second key is for SQL Server clusters. The SnapManager service should be restarted after setting this key:
  HKEY_LOCAL_MACHINE\SOFTWARE\NetApp\SnapManager for SQL Server\Server\MaxDbConcurrentBackup=DWORD::175
  HKEY_LOCAL_MACHINE\Cluster\SOFTWARE\Network Appliance\SnapManager for SQL Server\Server\MaxDbConcurrentBackup=DWORD::175

3.1.3 Environments with up to 6,000 databases

For environments that have a large number of small User databases, we can support up to 6,000 User databases per SQL Server when the following guidelines are followed. These guidelines require SMSQL 6.0 and SnapDrive 6.5 to be used, as these versions have scalability improvements that will help with large numbers of User databases.

- A 64bit version of the Windows Operating System and SQL Server must be used.
- Up to 750 User databases can be placed on a pair of data / log LUNs.
- Each User Database over 200GB should reside on a dedicated pair of LUNs if a fast RTO is required.
- Up to 8 pairs of data / log LUNs can be used per SQL Server. This will accommodate up to 8 * 750 = 6,000 User databases.
- The SnapInfo drive will be 20GB in size.
- Five snapshots per volume will be taken during each SMSQL backup process.
- The following registry keys must be set, in order to ensure that SMSQL only quiesces 175 databases at a time during the backup process. The first key is for stand alone SQL instances, and the second key is for SQL Server clusters. The SnapManager service should be restarted after setting this key:
  HKEY_LOCAL_MACHINE\SOFTWARE\NetApp\SnapManager for SQL Server\Server\MaxDbConcurrentBackup=DWORD::175
  HKEY_LOCAL_MACHINE\Cluster\SOFTWARE\Network Appliance\SnapManager for SQL Server\Server\MaxDbConcurrentBackup=DWORD::175

3.1.5 Database layout example

In order to illustrate the above guidelines, we will show a LUN/volume layout table for a SQL Server instance in a MSCS environment with the following configuration:

- 50 User databases under 200GB in size
- 1 User database that is 500GB in size

LUN Description	Mountpoint or Drive Letter	Storage Volume and Qtree Path	LUN Name
SystemDBs and TempDB	C:\<blah>\SystemDBs	/vol/<bu>_wi_<appname>_tsys_nosnap/luns	tmpsys
Quorom	Q:\		quorum
UserDB data files 1-35	C:\<blah>\UserDB1	/vol/<bu>_wi_<appname>_usr_snap/luns	userdb
UserDB log files 1-35	C:\<blah>\UserLog1		userlog
UserDB data files 36-50	C:\<blah>\UserDB2	/vol/<bu>_wi_<appname>_usr_snap2/luns	userdb2
UserDB log files 36-50	C:\<blah>\UserLog2		userlog2
UserDB data files 51	C:\<blah>\UserDB3	/vol/<bu>_wi_<appname>_usr_snap3/luns	userdb3
UserDB log files 51	C:\<blah>\UserLog3		userlog3
SMSQL SnapInfo	C:\<blah>\SnapInfo	/vol/<bu>_wi_<appname>_info_snap/luns	snapinfo

The above table is simply an example, and the layout rules as documented earlier in this section should always be followed.

3.2 Disk timeout settings

There are a number of Windows timeout related settings that must be in place in order to ensure non-disruptive operations with iSCSI connected LUNs during events such as storage system software upgrades and unplanned storage cluster failovers. These settings are typically implemented by installing the NetApp Windows Host Utilities, but in the TRP environment they will be automatically configured as part of the server build process. Given that network redundancy on the SQL Server will be handled by the Windows operating system using LACP, the settings that must be put in place are for non-MPIO configurations, and are as follows:

Property Type	**Property Value**
HKLM\SYSTEM\CurrentControlSet\Services\ClusDisk\Parameters\ ManageDisksOnSystemBuses	1
HKLM\SYSTEM\CurrentControlSet\Control\Class\{*iSCSI_driver_GUID*}\ *instance_ID*\Parameters\MaxRequestHoldTime	120
HKLM\SYSTEM\CurrentControlSet\Control\Class\{*iSCSI_driver_GUID*}\ *instance_ID*\Parameters\LinkDownTime	15
HKLM\SYSTEM\CurrentControlSet\Services\disk\TimeOutValue	120

3.3 Connecting iSCSI LUNs on the Windows server

The TRP storage team will provision the vFiler, volumes, qtrees, and LUNs required for each Windows server. The storage provisioning details are covered in section 2 of this document.

Once the LUNs have been provisioned on the storage system, and mapped to the iSCSI IQN of the Windows Server, the following steps must be taken on the SQL server prior to using the SnapDrive interface.

- Using the Microsoft iSCSI Initiator GUI:
    - On the **Discovery** tab, create and entry for the vFiler to will be connecting to.
    - On the **Targets** tab, log on to the vFiler. Select the **Automatically restore this connection when the system boots** check box when logging on to the target.
- Using Windows Disk Management:
    - Rescan the disks.
    - Format the disk as NTFS, using the quick format option, and assign a drive letter. Be sure to use a partition type of GPT.

3.4 Configuring SnapDrive on the Windows server

Once the iSCSI LUNs are formatted, they can be recognized with SnapDrive. Perform the following steps in the SnapDrive interface within Windows MMC:

- Right click on the SQL Server you are using in the SnapDrive interface and select **Transport Protocol Settings**.
    - On the **Storage Systems** tab, click **Add** and enter the following:
        - The vFiler you are using for the **Storage System**.
        - Set **Connection Type** to HTTP.
        - Enter the **User Name** as iscsi and set the **Password** to the unique password provided by the storage team.
        - Leave the **Port** set to 80.
- Right click on the **Disks** section of the SnapDrive interface and select **Refresh**.

Once these steps are completed, the iSCSI LUNs should be visible in the SnapDrive interface within MMC. SnapDrive authentication information can also be set using the *sdcli transport_protocol set* command. Also note that SnapDrive must have authentication information established for both primary and secondary storage in order to perform Ad-Hoc SMSQL backups with SnapVault replication, or to perform recovery from SnapVault secondary systems.

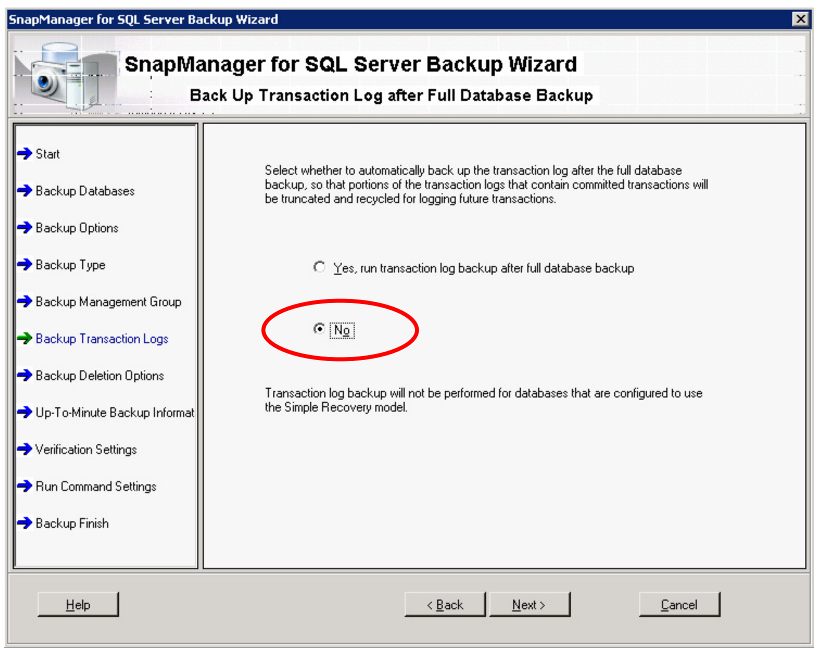
4 SMSQL configuration, backup, and recovery

4.1 SMSQL Configuration Wizard

After SMSQL is installed and the initial SQL databases have been configured on the SQL Server, the SMSQL Configuration Wizard should be used to migrate the database files from local disk to NetApp iSCSI connected LUNs. If the database files are already located on NetApp iSCSI connected LUNs, then the Configuration Wizard still needs to be run at least one time in order to make SMSQL aware of the configuration and to setup the SnapInfo location. After the initial run of the Configuration Wizard, it only needs to be run again if new NetApp iSCSI connected LUNs are added to the SQL Server and SQL database files are placed on the new LUNs.

4.2 SMSQL backup configuration and transaction log management

In the TRP environment, we will use SMSQL only to create full backups once daily. These backups should be configured such that transaction logs are not backed up as part of the full backup, and as such the transaction logs will not be truncated by the SMSQL full backup. The following screenshot shows this section of the SMSQL Backup Wizard:



The truncation of transaction logs will be managed by the TRP SQL Server DBA team. This is required for SQL Server instances that use SQL Log Shipping for replication, since SMSQL does not directly integrate with SQL Log Shipping from a transaction log management perspective. In order to maintain consistent configurations throughout the environment, all SQL Server instances will be managed in this fashion regardless of whether or not they use SQL Log Shipping. The TRP SQL Server DBA team will also be responsible for retaining transaction log backups for the appropriate duration, and making the transaction log backups available for log replay in the event of a database recovery with SMSQL. The transcation log backups will typically be stored on a well defined and common CIFS share in a given datacenter module.

4.3 SMSQL Ad-Hoc backups

In certain use cases, there may be a requirement for a SQL Server to be backed up more than once in a given day, and for these backups to be replicated to the SnapVault secondary storage system. For these scenarios, a program named "*sdcli_snapvault.exe*" has been provided to TRP in order to automate this replication. This program uses "*sdcli*" commands on the Windows server has the following requirements:

- It must be run on the SQL Server immediately after the Ad-Hoc SMSQL backup completes.
- It requires that SnapDrive authentication be configured for the SnapVault secondary storage system.
- It requires no command line arguments, although an optional path to a log file can be provided in order to log messages to this file instead of standard output.
- It will replicate any new snapshots for all volumes with LUNs attached to the Windows server from primary to secondary storage. It will create snapshots with an identical name on secondary storage, with the exception that the snapshots will have a "sv_" prefix.
- It will not finish running until the replication to the SnapVault secondary has completed. For this reason, the Windows server it is running on must not be rebooted during this time.

4.4 SMSQL restore scenarios

4.4.1 Determining the secondary storage system from the Windows server

The storage system being used for SnapVault secondary backup snapshots can be determined by a Windows server administrator or by a SMSQL administrator by running the "*sdcli snapvault relationship_status –D <Drive>*" command on the Windows server, which leverages the SnapDrive CLI. For example, run "*sdcli snapvault relationship_status –D S:*" in order to list the SnapVault relationship details for the volume containing the iSCSI LUN mapped to the *S:* drive.

4.4.2 LUN clones, snapshot dependencies, and busy snapshots

Some SMSQL operations such as backups, database cloning, and certain restore scenarios make use of a snapshot backed LUN clone when an environment is not licensed for FlexClone (as is the case with TRP). A snapshot backed LUN clone will result in a LUN with a "*.rws*" extension being placed in the active filesystem of the volume in order to hold any new writes to the LUN, while all reads for the LUN are serviced by the snapshot copies of the data blocks.

When this *".rws"* LUN exists in the active filesystem, and a new snapshot is taken, that new snapshot captures the "*.rws*" LUN. Since the "*.rws*" LUN actually points to an older snapshot for handling read requests, we now have a situation where the new snapshot has a dependency on the older snapshot. This means that the old snapshot cannot be removed until the LUN clone is disconnected (automatically removing the "*.rws*" LUN) and the new snapshot is also removed. When this condition exists, the old snapshot will be listed as having a "*busy, LUNs*" dependency.

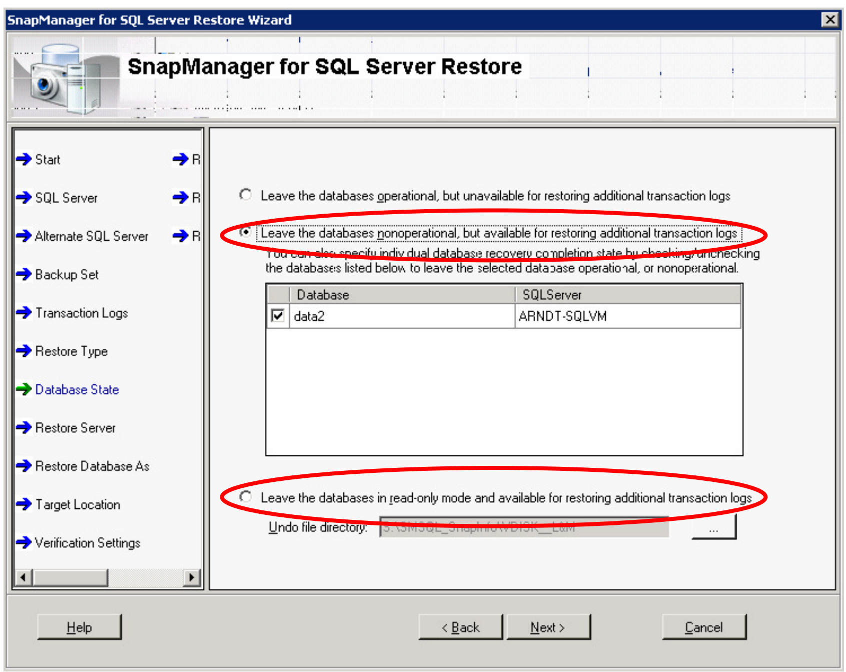
While it is not typically a problem to have snapshots with a "*busy, LUNs*" dependency, it can result in confusion when a newer snapshot must be removed before an older snapshot can be removed. For this reason, it is recommended not to create new snapshots when LUN clones are in existence on a given volume. At a higher level, this means that any database cloning or restore operations should be finished and cleaned up before taking new SMSQL or SnapDrive backups.

4.4.3 Restore from Primary SMSQL snapshot backups

The SMSQL Restore Wizard can be used to restore all databases on a given LUN, or a single database on a LUN with other databases, as long as the SMSQL backup snapshots still exist on primary storage. The SMSQL Restore Wizard can also be used to restore SQL Server system databases, as long as the SnapInfo backup snapshots still exist on primary storage.

When SMSQL restores a user database that is isolated to it's own set of data and log LUNs, a fast *lun clone* and *lun clone split* operation is performed in order to speed recovery of databases that are very large in size. By contrast, when SMSQL restores a user database that shares it's data and log LUNs with other user databases, a clone of the LUN is connected to the SQL Server and the files are simply copied back to their original location. This streaming type of restore is also used whenever restoring system databases from SMSQL backups.

In either restore scenario, the SMSQL Restore Wizard must use one of the available options that leave the user database "*available for restoring additional transaction logs*", if the TRP SQL Server DBA wishes to manually apply transaction logs after the restore is completed. The following screenshot shows this section of the Restore Wizard:



If a simple point in time recovery is desired, then the SMSQL Restore Wizard can be used leave the databases operational, and the database will be brought back online without the option for additional transaction log reply.

4.4.4 Restore all databases on a LUN from secondary snapshot backups

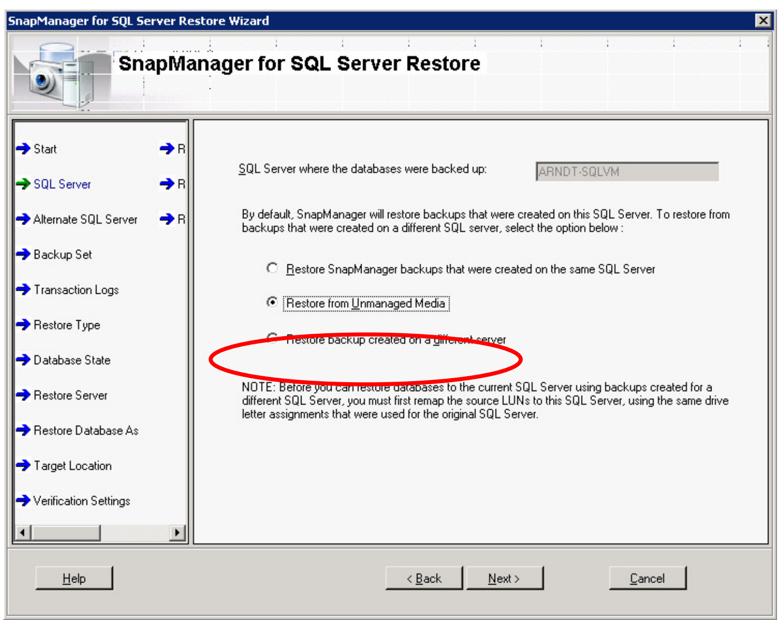
To restore all user databases on a given set of data and log LUNs from a SnapVault Secondary storage system to the primary storage system, the *snapvault restore* command will be used to replicate the entire LUNs from secondary storage back to primary storage. The steps to perform this restore would be as follows:

1. Take all the databases using the user DB data and log LUNs offline.
2. Disconnect the user DB data and log LUNs from the SQL Server using the SnapDrive interface in MMC.
3. Restore these LUNs using the incremental *snapvault restore* command from the Primary_vFiler, as follows:

   vfiler run Primary_vFiler snapvault restore -r –s <snapshot_name> –S Secondary_vFiler:/vol/<volname>/luns /vol/<volname>/luns
4. Wait for the "*snapvault restore"* command to finish. This can be monitored using the "*snapvault status*" command from the Primary_vFiler. When the restore is finished, use the following command to re-establish the SnapVault relationship:

   vfiler run Secondary_vFiler snapvault start -r –S Primary_vFiler:/vol/<volname>/luns /vol/<volname>/luns
5. Connect the LUNs using the SnapDrive interface in MMC.
6. Use the SnapDrive interface in MMC to connect to a copy of the SnapInfo LUN from the appropriate snapshot backup on the Secondary_vFiler. Use the most recent SnapInfo snapshot from the backup date you are restoring from if there are multiple SnapInfo snapshot timestamps to choose from.

   Hint: When navigating the SnapDrive interface to find the LUN on the Secondary_vFiler, click on *<volume_name> -> .snapshot -> <snapshot_name>* and then type in the remainder of the path to your LUN in the **LUN Path** textbox of the SnapDrive interface. The **LUN Path** textbox will list autocomplete options at each level of the directory path.
7. Once the user database data and log file LUNs have been restored to the SQL Server, use the SMSQL Restore Wizard and choose the "*Restore from Unmanaged Media*" option, as shown in the following screenshot:



8. When prompted by SMSQL, navigate to the location of the SnapInfo folder that was connected from the Secondary_vFiler. Restore the database(s) such that they are "*available for restoring additional transaction logs*" if you would like to manually apply transaction logs after the restore is completed.

4.4.5 Restore a single database on a LUN from secondary snapshot backups

To restore a single database on a given LUN (which also contains other user databases) from a SnapVault Secondary storage system to the primary storage system, we must connect to the user database data and log file LUNs on the Secondary_vFiler, as follows:

1. Take the database offline.
2. If there is a single snapshot on the data and log LUNs from the backup process that you are restoring from, simply connect to the LUNs from that snapshot timestamp. If there are enough databases on a single LUN such that multiple snapshots are created as part of a single backup process for those databases, you can determine the timestamp of the snapshot backup that you want to restore from by viewing the SMSQL backup report. You will find the timestamp in the "*Backup Time Stamp*" section of the backup report for the database of interest. For example:

   ```
   [10:10:42.958]  Backup Time Stamp: 11-12-2013_10.10.20
   [10:10:42.958]  **** Backup Group [#1] of Job#1
   [10:10:42.958]  Backup Database List:
   [10:10:42.958]  1 - ARNDT-SQLVM : data2
   [10:10:42.958]  2 - ARNDT-SQLVM : data3

   [10:10:48.614]  Backup Time Stamp: 11-12-2013_10.10.48
   [10:10:48.614]  **** Backup Group [#1] of Job#2
   [10:10:48.614]  Backup Database List:
   [10:10:48.614]  1 - ARNDT-SQLVM : data4
   [10:10:48.614]  2 - ARNDT-SQLVM : data5
   ```

   In the above example section of a SMQL backup report, the data2 and data3 databases were backed up in the snapshot with the `11-12-2013_10.10.20` timestamp, and the data4 and data5 databases were backed up in the snapshot with the `11-12-2013_10.10.48` timestamp.
3. Use the SnapDrive interface in MMC to connect to a copy of the following LUNs from the appropriate snapshot backup on the Secondary_vFiler:
   a. The SnapInfo LUN.
   b. The user database data file LUN.
   c. The user database log file LUN.

   Hint: When navigating the SnapDrive interface to find the LUN on the Secondary_vFiler, click on *<volume_name> -> .snapshot -> <snapshot_name>* and then type in the remainder of the path to your LUN in the **LUN Path** textbox of the SnapDrive interface. The **LUN Path** textbox will list autocomplete options at each level of the directory path.
4. Copy the data file and log file for this database from the backup copies that were attached to the SQL Server in step 2 to the production location for these files.
5. Use the SMSQL Restore Wizard and choose the "*Restore from Unmanaged Media*" option. When prompted, navigate to the location of the SnapInfo folder that was connected from the Secondary_vFiler. Make sure to use the folder with the timestamp that matches what you found in the SMSQL backup report, which should also match the timestamp of the snapshot backup of the LUNs you used to restore the data and log file in step 3. Restore the database(s) such that they are "*available for restoring additional transaction logs*" if you would like to manually apply transaction logs after the restore is completed.
6. Use the SnapDrive interface to disconnect the SnapVault secondary copies of your LUNs.

4.4.6 Restore a system database from secondary snapshot backups

To restore a system database from a SnapVault Secondary storage system to the primary storage system, the procedure is as follows:

1. Use the SnapDrive interface in MMC to connect to a copy of the SnapInfo LUN from the appropriate snapshot backup on the Secondary_vFiler:

   Hint: When navigating the SnapDrive interface to find the LUN on the Secondary_vFiler, click on *<volume_name> -> .snapshot -> <snapshot_name>* and then type in the remainder of the path to your LUN in the **LUN Path** textbox of the SnapDrive interface. The **LUN Path** textbox will list autocomplete options at each level of the directory path.
2. Copy the "*.FBK*" file (in the StreamFullBackup folder) from the secondary snapshot connected LUN of the system database of interest back to the original location on your active SnapInfo LUN. Here is an example path to one of these .FBK files:
   Y:\SMSQL_SnapInfo\SQL__ARNDT-SQLVM\DB__master\FG__\StreamFullBackup\05-09-2011_17.24.27__master.FBK
3. In the SMSQL Restore interface, click on "*Find Backups*" on the right hand *Actions* pane. After this operation completes, you will be able to perform the restore of the system database using the SMSQL Restore Wizard, just as if those backups that were copied to the active SnapInfo LUN had still existed on primary storage.
4. Use the SnapDrive interface to disconnect the SnapVault secondary copy of your SnapInfo LUN.

4.4.7 Restoring a database to an alternate path

Certain use cases may require that a database be restored to an alternate path on the same SQL Server instance, or to an alternate path on a different SQL Server instance.

If the database being restored does not require log recovery to roll forward to a point in time that is different than the available backups, a SQL DBA can simply connect to the snapshot copies of the database and log LUNs on primary or secondary storage and copy the .MDF and .LDF files to the SQL instance and file path of their choice. They can then attach these files as a new database to the SQL instance and bring the database online.

If the database being restored requires log recovery, then SMSQL must be used in this process in order to bring the database online with the ability to perform additional log replay. In this use case, the following procedure can be used:

1. Use the SMSQL *Clone Wizard* to clone a database from an existing backup set.
2. Select the database to be cloned, and give it a new name in the *Clone as Database* screen.
3. Change the target SQL Server instance if required.
4. Leave the database *offline but available for restoring additional transaction logs*.
5. Continue with the rest of the clone wizard. This will create a clone of your database by connecting to snapshot copies of the LUNs so as not to consume additional space on the user database drive.
6. Apply the desired transaction logs via standard processes, outside of SMSQL.
7. Take the database offline, then copy the .MDF and .LDF files to the location that you want them to live long term and attach them to them back to the SQL instance of your choice.
8. Bring the cloned database online.
9. Use the SMSQL *Clone Wizard* to delete the clone you created. This will remove the temporary LUNs that were connected to the snapshot copies when the clone was created.

5 Non-SQL iSCSI LUN backup and recovery

The backup of non-SQL iSCSI LUNs will be performed by a script written by TRP to creat snapshots using the SnapDrive *sdcli* command on primary storage. SnapVault replication of non-SQL iSCSI LUNs will be performed in the same manner as SMSQL created snapshots, as outlined in section 2.6 of this document.

The following list outlines the high level approach to recovery of non-SQL data on iSCSI LUNs:

- Single file recovery is performed using the *sdcli snap restore –flr* command. See the following documentation for more information on the syntax of this command, which will preserve file and folder ACLs when it performs the restore: https://now.netapp.com/NOW/knowledge/docs/snapdrive/relsnap63/html/software/admin/GUID-4D0A718E-D072-47F1-BA60-667E17CDD466.html
- Recovery of an entire LUN from a snapshot on primary storage is performed using the SnapDrive *Restore disk* functionality, which is available in the right click menu of a snapshot in the SnapDrive MMC interface, or via the *sdcli snap restore* command.
- Recovery of an entire LUN from a snapshot on secondary storage is performed by the TRP storage team, using the *snapvault restore* command as documented in section 4.4.4. The LUN being restored must first be disconnected from the Windows server via SnapDrive, then restored using the *snapvault restore* command, and finally reconnected on the Windows server via SnapDrive when the *snapvault restore* is complete.

Appendix A: Resources and Whitepapers

- SnapManager 5.0 for SQL Server: Best Practices Guide
    - http://media.netapp.com/documents/tr-3768.pdf
- NetApp SnapManager 5.1 for Microsoft SQL Server Installation and Administration Guide:
    - https://now.netapp.com/NOW/knowledge/docs/SnapManager/relsmsql51/html/software/install/index.htm
- NetApp SnapManager 5.2 for Microsoft SQL Server Installation and Administration Guide:
    - http://support.netapp.com/documentation/docweb/index.html?productID=61114
- NetApp SnapDrive 6.3 for Windows Installation and Administration Guide:
    - https://now.netapp.com/NOW/knowledge/docs/snapdrive/relsnap63/html/software/admin/index.html
- NetApp SnapDrive 6.4 for Windows Installation and Administration Guide:
    - http://support.netapp.com/documentation/docweb/index.html?productID=60386
- NetApp Windows Host Utilities 5.3 Installation and Setup Guide:
    - https://now.netapp.com/NOW/knowledge/docs/hba/win/relwinhu53/html/software/setup/index.html