



NetApp®
Go further, faster

Technical Report

Best Practice Guide for Microsoft SQL Server on NetApp Storage

John S. Parker, NetApp
Sourav Chakraborty, NetApp
January 2010 | TR-3821

BEST PRACTICE GUIDE FOR SQL SERVER 2005/2008

This best practice guide is designed to give storage administrators and database administrators the keys to successfully deploying Microsoft® SQL Server® 2005 and 2008 on NetApp® storage. This technical report will discuss Microsoft and NetApp high-availability solutions, best practices for data management, backup and recovery, and best practices for storage layout.

Microsoft®
GOLD CERTIFIED
Partner

TABLE OF CONTENTS

- 1 EXECUTIVE SUMMARY..... 3**
 - 1.1 PURPOSE AND SCOPE 3
 - 1.2 INTENDED AUDIENCE 3
 - 1.3 CAVEATS 4
- 2 NETAPP SIMPLIFIES SQL SERVER DATA MANAGEMENT 4**
 - 2.1 HIGH AVAILABILITY 4
 - 2.2 IMPROVED PERFORMANCE AND ASSET UTILIZATION 6
 - 2.3 ACCELERATED DEVELOPMENT AND TEST WITH RAPID CLONING 7
 - 2.4 Fast backup and restore with SnapManager for SQL Server..... 7
- 3 DATABASE AND STORAGE LOGICAL AND PHYSICAL LAYOUT..... 8**
 - 3.2 MANAGING SPACE..... 9
- 4 BACKING UP AND RECOVERY FOR SNAPMANAGER FOR SQL SERVER 13**
 - 4.1 BACKUP RECOMMENDATIONS..... 14
 - 4.2 ARCHIVING BACKUPS 15
- 5 HOST AND STORAGE SYSTEM INSTALLATION AND CONFIGURATION 15**
 - 5.1 NETAPP STORAGE SYSTEM REQUIREMENTS 15
 - 5.2 INSTALLING AND CONFIGURING SNAPDRIVE 4.1 (OR HIGHER) 16
 - 5.3 STORAGE MIRRORING AND DISASTER RECOVERY 17
- 6 ONGOING MANAGEMENT AND MONITORING..... 19**
 - 6.1 SNAPSHOT RESERVE USAGE MONITORING..... 19
 - 6.2 DISK SPACE USAGE MONITORING 19
 - 6.3 FRACTIONAL SPACE USAGE MONITORING 19
 - 6.4 NETAPP STORAGE SYSTEM EVENT MONITORING..... 21
 - 6.5 TERMINAL SERVER OR REMOTE DESKTOP SERVICES..... 21
- 7 SUMMARY 21**
- APPENDIXES..... 22**
 - APPENDIX A: SMSQL BEST PRACTICES RECAP 22
 - APPENDIX B: CONSOLIDATION METHODOLOGIES 24
 - APPENDIX C: REFERENCES..... 28

1 EXECUTIVE SUMMARY

Today's business applications are more data-centric than in the past, requiring fast and reliable access to intelligent information structures that are often provided by a high-performance relational database system. Over the last decade, many organizations have selected Microsoft SQL Server to provide just such a back-end data store for mission-critical business applications. The latest release, Microsoft SQL Server 2008, builds on the strong base of SQL Server 2005 delivering performance, scalability, availability, and security.

SQL Server implementations have become more complex and require more reliability than before. Database service-level agreements (SLAs) require predictable performance, and outages are disruptive and costly for the database consumers. The underlying physical storage architectures supporting SQL Server are expected to scale in order to meet the growing capacity, but frequently bottlenecks arise, and backup processes get slower as databases get larger. SQL Server databases are growing significantly larger to meet the needs of today's organizations, while business requirements dictate shorter backup and restore windows. When the SQL Server out-of-box streaming backup and recovery solution is used, restore times take the current backup time plus additional time to play back any transaction logs. The larger the database, the harder it is to meet organizational service-level agreements. Failure to meet the organization's SLA can have a direct and devastating effect on revenue and customer goodwill. It is clear that conventional backup mechanisms simply do not scale, and many organizations have sought new and innovative solutions to solve increasingly stringent business requirements.

1.1 PURPOSE AND SCOPE

The success or failure of any software or infrastructure and software deployment hinges on making the right choices and configurations up front. The core goal of this report is to provide best practices for deploying and using SnapManager® for SQL Server (SMSQL) with Microsoft SQL Server 2005 and 2008 with NetApp storage systems and supporting software. Organizations looking to get the most out of their NetApp storage investment with SQL Server will benefit from putting to practice recommendations put forth in this report.

At a high level, the following areas will be covered:

- New features of SMSQL and complementary NetApp technologies
- Best practices for both database and storage logical and physical layout with SQL Server and NetApp storage systems and software
- Backup and recovery (BR) and disaster recovery (DR) considerations and tactics
- Best practices for installation, configuration, and ongoing use of SnapDrive® and SMSQL
- Ongoing management and monitoring best practices

Note: This document will highlight specific best practices sections throughout. All best practices are also available in Appendix A.

1.2 INTENDED AUDIENCE

This report is intended for experienced SQL Server administrators, IT managers, and storage administrators who have reviewed the documents below:

- NetApp SnapDrive Installation and Administration Guide
- SnapManager for SQL Server (SMSQL) Installation and Administration Guide
- Data ONTAP System Administration Guide

Readers should ideally have a solid understanding of the SQL Server storage architecture and administration as well as SQL Server backup and restore concepts. For more information about Microsoft SQL Server architecture, refer to SQL Server Books Online (BOL).

1.3 CAVEATS

- Applications utilizing SQL Server as a back end might have specific requirements dictated by the design characteristics of the application and are beyond the scope of this technical report. For application-specific guidelines and recommendations as they relate to physical and logical database layout, contact your application provider.
- Best practices for managing SQL Server environments will focus exclusively on the latest NetApp storage operating system, Data ONTAP®.

2 NETAPP SIMPLIFIES SQL SERVER DATA MANAGEMENT

Database management in today's business environment is looking for better ways in which to manage database assets. Better management of the database systems provides organizations the ability to meet their SLAs and reduce costs. Meeting SLAs through better management allows for higher efficiency and better use of database assets. This paper will discuss high availability, database mirroring, SnapMirror®, and SQL Server replication.

2.1 HIGH AVAILABILITY

NetApp storage systems are uniquely scalable and highly available with the ability to scale from your small database systems to the very large data warehouses. NetApp fabric-attached storage (FAS) systems offer a rich set of features with inherent redundancy of robust hardware components. Clustered FAS systems are available for even higher levels of availability. Innovations such as double-parity RAID (RAID-DP®) enhance availability by protecting against secondary drive failures that might occur during RAID reconstructs. For more information about RAID-DP, refer to [TR-3298: NetApp Data Protection: Double-Parity RAID for Enhanced Data Protection with RAID-DP](#). The SnapDrive family of products offers multipath I/O (MPIO) solutions for both iSCSI and Fibre Channel protocols, providing redundant paths from hosts to FAS systems. For more information about MPIO, visit the [Microsoft MPIO site](#).

What does database mirroring protect you from versus Microsoft Windows® failover clustering (WFC), and when do you use it over SnapMirror? Database mirroring is great for protecting against computer room or server rack failures. It requires redundant storage and like systems where WFC protects a database from server failure but not so much from an I/O subsystem failure. SnapMirror plays a great part in moving data over very long distances and allows you to move over the entire SQL Server instance rather than individual databases.

These capabilities enhance the key features that allow for highly available systems. When applying these concepts to Microsoft SQL Server, several approaches can be considered. The considerations that this best practice guide will discuss are database mirroring, SnapMirror replication, SQL Server replication, and Microsoft Windows failover clustering. This guide will also discuss what you should consider for business continuance and disaster recovery.

2.1.1 DATABASE MIRRORING

Database mirroring has been a part of Microsoft SQL Server since 2005. It is part of both SQL Server 2005 and 2008. Database mirroring is one of the key tools used by SQL Server to achieve high availability; it is one of those tools that not only provides high availability, but also provides a level of data protection from catastrophic system failure. Management of a mirrored database requires planning on the client and infrastructure side to deal with the unique issues related to mirroring. Database mirroring requires the same storage configuration for both sides of the mirror. It is important to make sure that both the primary and the target have similar LUN structures to maximize SQL Server performance between mirrors. It also requires proper bandwidth between the different sites. Additional considerations concern the database client. If the client is not mirroring-aware, then failover to the mirror requires some type of intervention on the client to switch from the primary database to the mirror.

Database mirroring comes in types: "full safety on" and "full safety off"; basically these are synchronous and asynchronous, respectively. Full safety on, or synchronous, works as a two-phase commit from the client to the primary database to the target database. A fast connection between the primary database and the target (the mirrored copy) is required to make sure that what is committed to the client is written on both the primary and the mirror. Full safety off, or asynchronous, is more of a store and forward approach to committed transactions. It can result in lost transactions under certain circumstances.

To use SnapManager for SQL Server with database mirrors requires that you have it installed on both the primary database as well as the target. The mirrored database will appear to be offline or in recovery mode. This means a consistent Snapshot™ copy is not allowed until the database mirror set is failed over to the target site and the target database is online and open for transactions. If you need to recover, you must recover from the Snapshot copies that were taking place on what was initially the primary site.

Database mirroring can also be used as a migration strategy from one platform to another. Migrating databases from one platform to another has the potential for data outages while the move takes place. By mirroring the database you can move from a FAS system other than NetApp to a NetApp FAS, while minimizing the effect to the production database. Then at a later time the failover can take place, moving the database clients onto the new system.

2.1.2 SQL SERVER AND SNAPMIRROR

The use of SnapMirror elevates the business's ability to recover from major data center disasters and outages. The time to look at SnapMirror in both synchronous and asynchronous mode is when you need to have geographical DR. The best practice for SnapMirror with SQL Server is defined in [TR-3604](#).

Many organizations have requirements for mirroring of SQL Server databases to geographically dispersed locations for data protection and business continuance purposes. Like Snapshot, SnapMirror works at the volume level, so proper grouping of databases by the high-availability index (discussed in further detail in Appendix B: Consolidation Methodologies) can be particularly useful. Locating one database per FlexVol® volume provides the most granular control over SnapMirror frequency as well as the most efficient use of bandwidth between SnapMirror source and destination. This is because each database is sent as an independent unit without the additional baggage of other files such as `tempdb` or databases that might not share the same requirements. While sharing of LUNs and volumes across databases is sometimes required because of drive letter limitations, every effort should be made to group databases of like requirements when specific mirroring and backup policies are implemented.

Note: Data ONTAP 7G can support up to 200 FlexVol volumes per storage system and 100 for clustered configurations. Also, drive letter constraints in WFC configurations might require combining log and data into a single LUN or FlexVol volume.

2.1.3 SQL SERVER REPLICATION

The need to replicate a smaller subset of the database using SQL Server replication is the way to accomplish moving smaller sets of data to multiple locations. When sizing for replication, it is important to keep in mind the size of the distribution database, which is determined by the retention policy for the replication publications. The default is 72 hours, where all transactions that take place on replicated tables are stored in the distribution database. In order to properly size the distribution database, you need to have the delta change rate on the tables that are being replicated for those 72 hours. The retention period can be changed by altering the article publication.

2.1.4 SQL SERVER CLUSTERING

The use of a SQL Server cluster is a proven method to save against host failures. The use of a cluster allows for quick failover of a SQL Server instance level. The cluster does not protect against failure of the storage system. To overcome this challenge, you can use SnapMirror to build a fast way to provide a redundant storage failover with a simple script that uses simple cluster and SnapDrive command line interface (SDCLI) commands to manage both failover of the cluster to additional nodes and also failover of the cluster storage components to a SnapMirror location. The following sample of such a script is provided by John Jones, a NetApp solution architect.

Failover

```
ECHO.  
Cluster res "SQL Server" /offline  
Cluster res "Disk E:\" /offline  
Cluster res "Disk F:\" /offline  
Cluster res "Disk E:\" /delete  
Cluster res "Disk F:\" /delete  
ECHO  
ECHO Connecting to DR Storage.....please stand-by
```

```

SDCLI disk connect -p r200:/vol/sqlonlinedatadr/data1 -d e -dtype shared -I w2k8-
node1 iqn.1991-05.com.microsoft:w2k8-node1.demo.local w2k8-node2 iqn.1991-
05.com.microsoft:w2k8-node2.demo.local w2k8-node3 iqn.1991-05.com.microsoft:w2k8-
node3.demo.local w2k8-node4 iqn.1991-05.com.microsoft:w2k8-node4.demo.local -e
"SQL1"
SDCLI disk connect -p r200:/vol/sqlonlineelogdr/log1 -d f -dtype shared -I w2k8-
node1 iqn.1991-05.com.microsoft:w2k8-node1.demo.local w2k8-node2 iqn.1991-
05.com.microsoft:w2k8-node2.demo.local w2k8-node3 iqn.1991-05.com.microsoft:w2k8-
node3.demo.local w2k8-node4 iqn.1991-05.com.microsoft:w2k8-node4.demo.local -e
"SQL1"
Cls
ECHO.
ECHO Bringing SQL Cluster Online....
Cluster res "Disk E:\" /online
Cluster res "Disk F:\" /online
Cluster res "SQL Server" /online
Cluster res "SQL server agent" /online
Color 2f
ECHO *****
ECHO.
ECHO                                     !!!IMPORTANT!!!
ECHO.
ECHO                                     "The DR storage of the SQLONLINE"
ECHO.
ECHO                                     HAS BEEN REPLICATED TO NETAPP STORAGE SYSTEM (R200) AND
ECHO                                     SUCCESSFULLY MOUNTED
ECHO.
ECHO *****
ECHO.
Pause

```

Failback

```

snapmirror resync -f -S fas3050b:sqlonlinedatal R200:sqlonlinedatadr
snapmirror resync -f -S fas3050b:sqlonlineelog1 R200:sqlonlineelogdr
sdcli disk disconnect -d e
sdcli disk disconnect -d f

SDCLI disk connect -p fas3050b:/vol/sqlonlinedatal/data1 -d e -dtype shared -I
w2k8-node1 iqn.1991-05.com.microsoft:w2k8-node1.demo.local w2k8-node2 iqn.1991-
05.com.microsoft:w2k8-node2.demo.local w2k8-node3 iqn.1991-05.com.microsoft:w2k8-
node3.demo.local w2k8-node4 iqn.1991-05.com.microsoft:w2k8-node4.demo.local -e
"SQL1"
SDCLI disk connect -p fas3050b:/vol/sqlonlineelog1/log1 -d f -dtype shared -I
w2k8-node1 iqn.1991-05.com.microsoft:w2k8-node1.demo.local w2k8-node2 iqn.1991-
05.com.microsoft:w2k8-node2.demo.local w2k8-node3 iqn.1991-05.com.microsoft:w2k8-
node3.demo.local w2k8-node4 iqn.1991-05.com.microsoft:w2k8-node4.demo.local -e
"SQL1"
Cls
ECHO.
ECHO Bringing SQL Cluster Online....
Cluster res "SQL Server" /offline
Cluster res "SQL Server" /online
Cluster res "Sql Server Agent" /online

```

Best practice: Test cluster failover

Make sure that the cluster can fail over and all the resources are available before you initially establish the SQL Server instances.

2.2 IMPROVED PERFORMANCE AND ASSET UTILIZATION

FlexVol technology introduced in Data ONTAP 7G makes it possible to optimize storage utilization by logically partitioning NetApp storage (aggregates) into smaller virtualized volumes (FlexVol volumes). Many SQL Server environments support small databases with very high I/O appetites, often requiring many more disk drives for throughput than required for space. FlexVol makes it possible to create a base aggregate container with many disk drives and logically partition the aggregate into smaller virtualized volume

structures. By combining the same number of spindles previously sliced up across many conventional volumes, much higher throughput and storage asset utilization can be achieved while still maintaining the benefits of discreet dedicated volumes. FlexVol volumes can be created, expanded, or reduced in mere seconds, regardless of size. For more information about FlexVol volumes as they relate to databases, see [TR-3410: Data ONTAP 7G: FlexVol and FlexClone for Microsoft SQL Server](#) and [TR-3373: Data ONTAP 7G: The Ideal Platform for Database Applications](#).

One method for increasing storage efficiency for SQL Server environments is to thin provision the storage for the SQL Server environment. [TR-3653](#) outlines the key procedures for thin provisioning the environment. There must be some considerations taken into account when thin provisioning your storage. First, you must make sure that there is sufficient monitoring for the environment. Secondly, it is important to have the ability to add the needed storage in a timely fashion to make sure of continuing operations of the SQL Server system.

2.3 ACCELERATED DEVELOPMENT AND TEST WITH RAPID CLONING

The NetApp solution for rapidly deploying a development and test database solution uses the functionality of NetApp FlexClone® technology to create read-write clones of any database in a SQL Server instance. The use of a FlexClone copy to create a development and test environment with the minimal disk space creates both an efficient infrastructure but also an environment that has full data sets. The full data set allows for a robust dev/test infrastructure. The solution is outlined in the following documents:

- Development and Test for Microsoft SQL Server (www.netapp.com/us/library/white-papers/wp-7078.html)
 - Rapid Database Development and Deployment (<http://media.netapp.com/documents/wp-7014.pdf>)
- The use of FlexClone allows you to build a SQL Server database on either the existing SQL Server instance or some other instance on a separate host. These databases can be migrated from a FlexClone copy by splitting the clone from the parent. See the Data ONTAP Storage Management Guide for correct syntax for your version of Data ONTAP.

2.4 Fast backup and restore with SnapManager for SQL Server

SnapManager for SQL Server allows the activation of NetApp Snapshot copies as a means to protect data in a very timely fashion. The use of Snapshot technology greatly reduces the time it takes to back up large SQL Server instances. This also enables the optimum use of the storage system by tracking just the changing blocks of data.

Item-level recovery can also be accomplished through the use of FlexClone. Consider the scenario where you have a table that has been deleted or updated incorrectly. Traditional recovery methods required the restoration of the entire database. The NetApp solution to this is to create a FlexClone copy from a Snapshot copy created prior to the event where the table or tables were changed, then copy the tables from the clone into the production database. This requires minimal additional space.

INSTALLING AND CONFIGURING SMSQL 5.0

This section discusses the steps needed to install SMSQL 5.0 on SQL Server 2005 and 2008 hosts.

1. Prerequisites

The following are the prerequisites for installing SMSQL 5.0:

- PowerShell 1.0.
- SnapRestore® should be licensed on the storage controller.
- NetApp Operations Manager 3.7 only if you want to use backup archival feature of SMSQL 5.0.
- Data ONTAP 7.1.3 or above. Data ONTAP 7.3 or above is needed only if you want to use backup archival feature of SMSQL 5.0.
- SnapDrive for Windows 6.0.1 (to enable integration with Protection Manager during install).
- If the host system is running SQL Server 2005, Microsoft Data Access Components (MDAC) 2.8 SP1 must be installed. Windows Server 2003 SP1 and SP2 include MDAC 2.8 SP2.

2. Platform Support

Refer to the platform support documented in the Installation and Administration Guide. Some of the key highlights for platform support are:

- 1) If an SMSQL 5.0 instance is being used for remote administrations of other SMSQL instances, then there is no need to install SnapDrive on that host.
- 2) Upgrading SMSQL from older versions invalidates existing scripts based on the older versions due to changes in the CLI commands.

3. Sizing the SnapInfo Volume

The total size of SnapInfo can be calculated with the following formula:

$$\text{SnapInfo_LUN_Size} = ((\text{DB_Size} * \text{Daily_Change_Rate} * \text{NumberOfOnlineSnapShots}) + \text{TotalSizeOfSystemDatabases} + \text{SnapInfoMetadata})$$

SnapInfoMetadata rate is approximately .01 of the entire SQL Server instance.

This formula is used for each SQL Server instance. If there are multiple SQL Server instances on the host, then this process is required for each SQL Server instance.

4. Installing SMSQL 5.0 on a Microsoft Windows Failover Cluster

Apart from the steps listed in the Installation and Administration Guide for installing SMSQL on a WFC, the following points must be noted:

1. SMSQL 5.0 is *not* a cluster-aware application.
2. SMSQL and all its dependencies must be installed on each cluster node.
3. Each clustered instance must have its own SnapInfo LUN, which should be a shared disk.
4. Place the SnapInfo LUN resource in its appropriate SQL Server cluster group.

3 DATABASE AND STORAGE LOGICAL AND PHYSICAL LAYOUT

The proper layout of a SQL Server database is necessary for proper performance and management of the SQL Server infrastructure. [TR-3696: Microsoft SQL Server Storage Fundamentals for NetApp Storage Systems](#) provides the guidance for the layout of SQL Server on NetApp. This technical report outlines in detail the fundamental storage layout for best performance on NetApp storage systems.

Best practice: Use FlexVol volumes when more granular control is required

When backup requirements are unique to each database, there are two possible paths that can help optimize Snapshot copies and transaction log backups:

- **Most granular control:** Create separate FlexVol volumes for each database for more granular control of backup schedules and policies. The only drawback to this approach is a noticeable increase in administration and setup.
- **Second most granular:** Try to group databases with similar backup and recovery requirements on the same FlexVol volume. This can often provide similar benefits with less administrative overhead.

Best practice: Avoid sharing volumes between hosts

With exception to Microsoft Windows failover clustering (WFC), it is not recommended to let more than a single Windows server create and manage LUNs on the same volume. The main reason is that Snapshot copies are based on volumes, and issues can arise with Snapshot consistency between servers. Sharing volumes also increases chances for busy Snapshot copies.

3.1 Migrating from Drive Letters to Volume Mountpoints

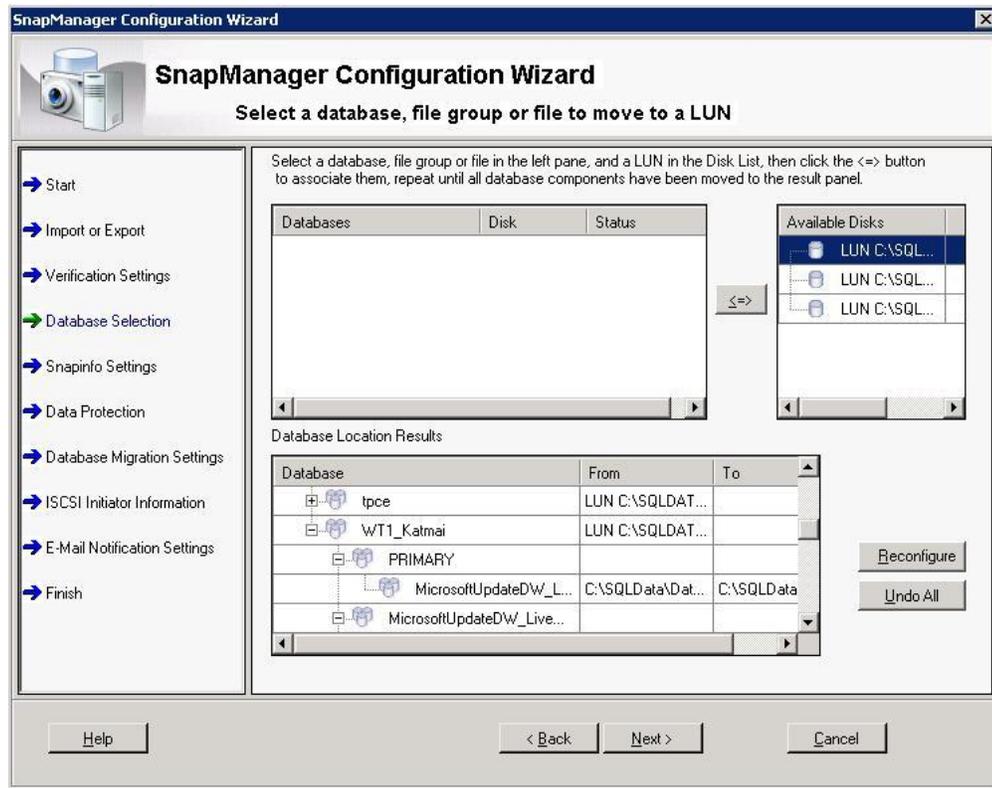
Existing SMSQL configurations have data LUNs mapped to drive letters. Migration of existing system and user databases are handled through the configuration wizard. This can be either drive letters or mountpoints. The affected databases will be taken offline briefly to remap the paths. This process saves time by not having to copy physical data and by minimizing the time that the databases are offline.

When migrating from drive letters to volume mountpoints, you lose the ability to restore those LUNs from previously created backups. It is best practice to create a full backup of all databases both before and after a successful migration. Once the migration is successful, you can discard previous backups. You cannot

restore your databases from these backups. Consult with your NetApp Global Services representative ((888) 4NETAPP) to assist in planning for a migration such as this.

Step 1:

A database is residing and attached to LUN M. Add a reference (C:\Mntp1\db) to existing LUN M using SnapDrive.



Step 2:

Run SMSQL configuration wizard, highlight the database, and click “Reconfigure.” SMSQL Configuration Wizard will list all references to the same LUN on the disk list; each LUN reference will have a label that lists any other reference to the same LUN:

LUN M <C:\Mntp1\db>
LUN C:\Mntp1\db\ <M>

Step 3:

Select C:\Mntp1\db (second reference) and associate it with the database. Press Next to proceed and complete the Configuration wizard. The database will now be attached to “C:\Mntp1\db” instead of “M.”

3.2 MANAGING SPACE

3.2.1 SNAPSHOT COPIES

It is beyond the scope of this paper to explore Snapshot copies in great detail. However, it is necessary to understand the fundamentals of the unique features of NetApp Snapshot functionality and how it relates to Microsoft SQL Server 2005 and 2008. For more information about NetApp Snapshot technology, refer to [TR-3001](#).

NetApp Snapshot backups complete in a matter of seconds, and typically each copy consumes only the amount of data that has changed since the last copy was created. Thus Snapshot copies consume minimal disk space while providing up to 255 online point-in-time images. The amount of disk space consumed by an individual Snapshot copy is determined by two factors:

- The rate data changes within file systems (in megabytes/sec or megabytes/hour, for example)
- The amount of time that elapses between creation of Snapshot copies

The measure of change (in megabytes, gigabytes, and so on) that occurs in the volume between creations of Snapshot copies is often referred to as the delta. The total amount of disk space required by a given Snapshot copy equals the delta changes in the volume along with a small amount of Snapshot metadata.

3.2.2 FRACTIONAL SPACE RESERVATION

Successfully using a fractional reserve policy requires an understanding and correct configuration of the following parameters. This section describes the volume and autodelete settings for proper operation.

Note: Data ONTAP 7.2.4 or later should be used when implementing these settings.

3.2.2.1 VOLUME SETTINGS

When using 0% fractional reserve, the following storage system volume options are recommended:

- `Guarantee = volume`
- `LUN_reservation = on`
- `Fractional_reserve = 0%`
- `Snap_reserve = 0%`
- `Auto_delete = volume`
- `Auto_grow = off` (optional, depending on your environment)
- `Try_first = snap_delete`

VOLUME GUARANTEE

FlexVol volumes, introduced with Data ONTAP 7.0, offer space guarantees, which allow you to determine when space is reserved or allocated from the containing aggregate.

NetApp recommends setting this parameter option to `volume`.

The `volume` option makes sure that the amount of space required by the FlexVol volume is always available from its aggregate. This is the default setting for FlexVol volumes.

With the space guarantee set to `volume`, the space is subtracted, or reserved, from the aggregate's available space at volume creation time. The space is reserved from the aggregate regardless of whether or not it is used for data storage.

FRACTIONAL RESERVE

Fractional reserve is a volume option that determines how much space Data ONTAP reserves for Snapshot overwrite data for LUNs and space-reserved files to be used after all other space in the volume is used. The default value is 100%; however, with the introduction of the autodelete functionality, NetApp recommends setting the fractional reserve to 0 and using the autodelete functionality. When a volume is configured to use fractional reserve space, Data ONTAP removes or reserves this space from the volume as soon as the first Snapshot copy is created and uses this space only when the rest of the volume is full.

The amount of space reserved can be viewed by using the `-r` option in the `df` command. This reserve area is used only when Data ONTAP reports that the volume is full. Unless the volume is full, space for Snapshot overwrites is taken from the volume. This reserved space is used only when the volume is 100% full.

SNAP RESERVE

This variable is set at the volume level as a percentage of the volume. Data ONTAP removes the defined percentage (20% by default) of the volume from being available for configuring LUNs or for file usage with CIFS or NFS. As Snapshot copies need space, they consume space in the snap reserve area. By default, after the snap reserve area is filled, the Snapshot copies start to occupy space from the general volume. For volumes containing LUNs, NetApp recommends setting the snap reserve percentage to 0.

AUTODELETE

Snapshot autodelete allows Snapshot copies to be deleted when certain volume conditions are met. These conditions, also called *triggers*, can be set so that Snapshot copies are automatically deleted under one of the following conditions:

- **volume:** Autodelete is triggered when the volume is nearly full. This information is displayed in the first line output reported for each volume by the `df` command.
 - **Note:** The volume can be full even if there is still space in the `snap_reserve` areas.
- **Snap_reserve:** Autodelete is triggered when the snap reserve space is nearly full.
- **space_reserve:** Autodelete is triggered when the overwrite reserve space is full. This is the space determined by the LUNs with space reservations enabled and the `fractional_reserve` option turned on. The reserve space will never be filled until both the volume and the `snap_reserve` areas are full.

Best practice: Set trigger option to volume

When configuring space reservation policies in Microsoft application environments, NetApp recommends setting the trigger option to `volume`.

3.2.2.2 TRY FIRST

The Try First setting determines how Data ONTAP space management settings are applied. Values for this parameter are:

- `snap_delete`
- `volume_grow`

3.2.2.3 BEST PRACTICE CONFIGURATIONS FOR SQL SERVER ENVIRONMENTS

The recommended autodelete settings are:

- `State = on`
- `Trigger = volume`
- `Delete_order = oldest_first`
- `Defer_delete = prefix`
- `Prefix = exchsnap`
- `Target Free space`

TRIGGER

The trigger setting defines what triggers the autodelete functionality in Data ONTAP. Setting this to `volume` means that autodelete functionality is triggered only when the volume reaches a low-space condition.

Table 1 defines the low-space thresholds in a volume.

Table 1) Autodelete Trigger settings.

Volume Size	Trigger
< 20GB	85%
20 < 100GB	90%
100 < 500GB	92%
500 < 1TB	95%
=> 1TB	98%

TARGET FREE SPACE

The target free space setting is the amount of space that Data ONTAP can free by deleting Snapshot copies with the autodelete function. By default this value is set to 20%. This means that if autodelete is invoked by a low-space condition, autodelete deletes Snapshot copies until the volume has 20% free space. NetApp recommends adjusting this value to 2% greater than the autodelete threshold.

For a volume size of 600GB, the `target_free_space` setting would be adjusted to 7%.

The order in which Snapshot copies are deleted is determined by the following three options:

- **DELETE_ORDER:** This option determines whether the oldest or newest Snapshot copy is deleted first. In the case of Microsoft applications, NetApp generally recommends setting this value to `oldest_first`, which deletes the older Snapshot copies first.
- **DEFER_DELETE:** This option allows the user to define a group of Snapshot copies that should first be deleted when no other Snapshot copies are available. It is possible to defer the deletion of user-created Snapshot copies, scheduled Snapshot copies, or Snapshot copies beginning with a configurable prefix.

Best practice: Use a prefix for Snapshot copies

It is recommended to use the `Prefix` value in SQL Server environments that use SnapManager for SQL Server. This allows the user to define a prefix for Snapshot copies on which deletion should be deferred.

- **PREFIX:** By using the `Prefix` switch with the defer delete option, it is possible to defer the deletion of user-created Snapshot copies, scheduled Snapshot copies, or Snapshot copies whose names have a configurable prefix. When using this option, a prefix of up to 15 characters can be defined; this can be used to defer the deletion of Snapshot copies created by the SnapManager application. When using SnapManager for SQL Server, the prefix should be set to `sqlsnap`, so that autodelete deletes user-created or incomplete Snapshot copies before deleting the oldest SnapManager backup set. This setting should be made on both database and transaction log volumes.

When configuring the prefix for different applications, the prefix should be set to contain the first five to seven letters of the Snapshot copy names.

AUTOSIZE SETTINGS

Volume autosize is a policy-based space management feature in Data ONTAP that allows a volume to grow in defined increments up to a predefined limit if the volume is nearly full.

Table 2 defines the low-space thresholds.

Table 2) Autosize settings.

Volume Size	Trigger
< 20GB	85%
20 < 100GB	90%
100 < 500GB	92%
500 < 1TB	95%
=> 1TB	98%

When using the autosize option, the `-m` option specifies the maximum size to which the volume can be grown. The `-I` option specifies the size increment by which the volume grows. The default value for the growth increment is 5%. This option is available only for flexible volumes. It is possible to define how fast the volume should grow by using the `-i` option. The default growth increment is 5% of the volume size.

When using the volume autosize option, it is important to note factors that might complicate autosizing of volumes—for example, in environments that use SnapMirror to replicate volumes.

Another important factor to keep in mind when using volume autosize is that the potential volume growth and the extent to which the volume can expand. It is critical to incorporate this potential overhead as extra space in the aggregate.

The advantage of this configuration is that it is easy to understand and monitor space utilization. Volumes are independent of each other, which means that each application subset can be tuned independently based on storage needs. This configuration also has the advantage that the oldest Snapshot copies are deleted while the most recent copies are retained.

Best practice: Determine additional space required

Add a safety net percentage to the total volume size. This additional space can help guard against low-volume space conditions.

3.2.3 DISK SPACE CONSUMED BY READ/WRITE SNAPSHOT COPIES

Even though read/write Snapshot files are initially linked to the existing blocks in the Snapshot copy, it is necessary for the NetApp storage system to allocate blocks equal to the entire size of the database files should it be completely overwritten and a copy of it created. There must be enough available disk space to accommodate the entire size of the original LUN while the read/write Snapshot copy is mounted. The space consumed by the read/write Snapshot copy is marked free disk space when it is dismounted using the Disconnect Disk command in SnapDrive. If the space is not available, consider using FlexClone to access Snapshot copies of databases.

3.2.4 SPACE REQUIREMENTS OVERVIEW

Estimating storage requirements can be complicated, especially when a sizing project involves server consolidation. The basic areas of focus consist of several related objects:

- (1) System databases
 - (a) How large does `tempdb` grow throughout the day?
- (2) User databases
 - (a) How frequently will the database and log be backed up?
 - (b) How large is each log dump?
 - (c) How large will the databases grow over the next two to three years?
- (3) SnapInfo
 - (a) How many full backups will be created and retained?
 - (b) How many log backups?
 - (c) How many stream-based backups of system databases?
- (4) Snapshot copy creation and retention
 - (a) What is the approximate delta between Snapshot copies?
 - (b) How many transaction log and Snapshot copies need to be available?
 - (c) When and how often should an archive backup take place?

Work to establish a rough estimate for each of the four areas. Appendix B puts all guidelines to practice in a consolidation project. Refer to this section for more detail.

4 BACKING UP AND RECOVERY FOR SNAPMANAGER FOR SQL SERVER

How to back up and restore SQL Server databases is thoroughly discussed in the SnapManager for SQL Server Installation and Administration Guide. Only areas that have to be well understood will be reiterated here.

SMSQL WILL BACK UP:

- Snapshot copies will be created of all volumes used by the databases being backed up.
- System databases will be backed up using conventional streaming-based backup.
- User databases placed in a LUN that also contains system databases will be backed up using streaming-based backup.
- Backup of transaction logs will always use streaming-based backup to provide point-in-time restores.

SMSQL WILL RESTORE DATABASE(S) BY:

The following steps happen when doing a restore; it uses LUN clone split restore:

- The administrator issues a standard LUN restore operation using SnapDrive.
- SnapDrive sends a snapshot-restore-file Manage ONTAP® command to the storage system to execute.
- Any LUN clone split running for the LUN being restored is aborted.
- User I/O is paused.
- A new LUN clone is created backed by the LUN in the Snapshot copy being restored from. This LUN is initially created without its space reservation to help avoid problems with insufficient free space.
- The original LUN is deleted. The delete is processed piecemeal by the zombie delete mechanism in the background.
- The new LUN clone is renamed to have the same name as the original LUN. Any host connections to the LUN are specially preserved.
- The space reservation is applied to the new LUN clone. Now that the original LUN is deleted, the new LUN clone should be able to regain the same reservation (since the restored LUN is the same size).
- User I/O is resumed. At this point the LUN is fully restored from the host's perspective. The host OS and applications can begin doing reads and writes to the restored data.
- A split of the LUN clone is started. The split runs in the background. This will be a space efficient split. The space reservation for the LUN clone is dropped during the split and reapplied once the split is complete.
- The SnapDrive LUN restore operation completes. This all usually takes about 15 seconds.

BENEFITS

The benefits of the new LUN clone split restores are:

- The SnapDrive restore operation completes in about 15 seconds.
- The LUN remains online and connected to the host throughout the restore. The host does not need to rescan for the LUN after the SnapDrive restore operation completes.
- The LUN is fully restored from the host's perspective once the SnapDrive restore operation completes in about 15 seconds. This is the magic of LUN clones.
- The LUN is available for host I/O (reads and writes) immediately after the SnapDrive restore operation completes in about 15 seconds.
- The split of the LUN clone that was created during the restore runs in the background and does not interfere with the correctness of host I/O to the LUN. Writes done by the host OS and applications are preserved in the clone as you would expect. This is the standard way LUN clones operate.
- The LUN clone split is a new style space efficient split that is usually 90% or more efficient, meaning it uses 10% or less of the fully allocated size of the LUN for the restore operation. The space efficient LUN clone split is also significantly faster than an old style block copy LUN clone split.
- You can determine the progress of the LUN clone split by asking for the LUN clone split status for the LUN. The status will report a percentage complete for the split.
- If the administrator accidentally starts the restore using the wrong Snapshot copy, they can easily issue a new restore operation using the correct Snapshot copy. Data ONTAP will automatically abort any LUN clone split that is running for the prior restore, will create a new LUN clone using the correct Snapshot copy, and will automatically split the LUN clone to complete the restore operation. Previously, the administrator had to wait until the previous single-file SnapRestore (SFSR) was complete before starting the SFSR over with the correct Snapshot copy.
- Multiple LUN clone split restores can be in progress simultaneously.
- No new procedures or commands are needed to use this functionality. The administrator should issue LUN restore operations using SnapDrive as they normally would, and Data ONTAP will handle the details internally.
- If the storage system is halted while the LUN clone split is running, when the storage system is rebooted the LUN clone split will be restarted automatically and will then run to completion.

4.1 BACKUP RECOMMENDATIONS

Full SMSQL backup using Snapshot is quick and does not adversely influence database performance compared to traditional backup with SQL Server BACKUP statements. Transaction log backup is functionally the same using SMSQL as the transaction log backup done using BACKUP log statements and is streamed out to the SnapInfo directory for safekeeping. The length of time required to back up a transaction log depends mainly on the active section of the transaction log that has to be extracted and

copied to the dump directory. When planning the backup and restore process, the methodology or considerations are different from considerations when backup is done by Enterprise Manager or SQL Server Management Studio. The two main considerations are:

- Recovery point objective (RPO): To what point in time does the data need to be recovered?
- Recovery time objective (RTO): How long will it take to get the database back online and rolled forward or backward to the RPO?

Consider the hypothetical customer scenario below:

Customer: ACME Corp.

Database: CommerceDB

Failure type: Database corruption or deletion

RTO: It cannot take longer than 30 minutes to complete a restore.

RPO: Data should be brought back to a time no later than 15 minutes before the failure.

ACME's IT group has shown that a full backup every 30 minutes with a transaction log backup every 15 minutes on the hour makes it possible to restore the database in 20 minutes (without the verification process) to the minimum recovery point objective of 15 minutes. Backup every 30 minutes during a 24-hour time frame will create 48 Snapshot copies and as many as 98 transaction log archives in the volume containing SnapInfo. A NetApp volume can host up to a maximum of 255 Snapshot copies, so the proposed configuration could retain over five days of online Snapshot data.

Thus, it is generally a good idea to categorize all your organization's systems that use SQL Server in terms of RTO and RPO requirements. Once these requirements have been established, SMSQL can be tailored to deliver the appropriate level of protection. Proper testing before deployment can also help take the guesswork out of what can be realistically achieved in production environments.

In situations where business requirements are not entirely clear, refer to the recommended default database backup frequency below.

Best Practice: Minimum Backup Frequency

All backup and recovery processes *should be* created according to each environment's unique requirements. Determine what the RPO and RTO are, then set full backups and transaction log backups appropriately.

4.2 ARCHIVING BACKUPS

Best practice: Always archive or mirror Snapshot copies

Archiving or mirroring backup sets as soon as possible is strongly recommended. Disasters do occur, and if the storage device containing the databases and backup Snapshot images is adversely affected, it will not be possible to restore the databases from the primary storage system. Archive media can be tape, another FAS system, or a device enabled by NearStore®. The archive process is described in further detail in SMSQL 5 Installation and Administration Guide.

5 HOST AND STORAGE SYSTEM INSTALLATION AND CONFIGURATION

Several things need to be done in preparing a NetApp storage system and SQL Server host to create a reliable system with optimal performance. Refer to the latest SnapDrive and SnapManager for SQL Server Administration Guides to make sure the proper licenses and options are enabled on the NetApp storage device.

5.1 NETAPP STORAGE SYSTEM REQUIREMENTS

- Data ONTAP 7G is strongly recommended.
- SnapDrive 4.1 or higher.
- Enable licenses for FCP or iSCSI, SnapManager, and SnapMirror according to requirements.
- At least one gigabit Ethernet NIC for administrative connectivity (copper or fiber).

5.2 INSTALLING AND CONFIGURING SNAPDRIVE 4.1 (OR HIGHER)

5.2.1 PREPARING FOR INSTALLATION

Prior to launching the installation of SnapDrive, use the following checklist to help eliminate the potential for errors or delays during or after the installation.

- Resolve any hardware, cabling, or network issues or other errors.
- Make sure all of the necessary software and printed or electronic documentation (found on <http://now.netapp.com>) is organized and nearby before beginning.
- Configure DNS, hostname, and IP address-related services:
 - Verify that all related systems, including storage systems, servers, and clients, are configured to use an appropriately configured DNS server.
 - Manually add the NetApp storage systems' host names to DNS.
 - Enable, configure, and test RSH access on the storage systems for administrative access (for more information, see Chapter 2 of the Data ONTAP 7.x Storage Management Guide).
- License all of the necessary protocols and software on the storage system.
- Make sure the storage system's date and time are synchronized with the Active Directory domain controllers. This is necessary for Kerberos authentication. A time difference greater than five minutes will result in failed authentication attempts.
- Verify that all of the service packs and hot fixes are applied to Microsoft Windows Server 2003/2008/2008 R2 and Microsoft SQL Server 2005 and 2008.

5.2.2 SNAPDRIVE INSTALLATION TIPS

Select the same account used by the Microsoft SQL Server services when selecting the service account for the SnapDrive and SnapManager for Microsoft SQL Server services.

When creating or configuring the properties for the domain service account, select the "Password never expires" checkbox. Doing so protects the account and service from failing due to user password policies.

Reminder: It's important to make certain the service account has the following permissions:

- Read and write or full control access to the locations in which LUNs will be created (likely if it's already a member of the administrator's group).
- RSH access to the storage systems. For more information on configuring RSH, see the Data ONTAP Administration Guide.

5.2.3 CREATING AND MANAGING LUNS WITH THE SNAPDRIVE GUI

Once installed, SnapDrive can be used to create LUNs on NetApp storage systems for use by Windows Server 2003, 2008, and 2008R2 hosts.

Best practice: Tips for creating LUNs

- When specifying a UNC path to a share of a volume to create a LUN, use IP addresses instead of host names. This is particularly important with iSCSI, as host-to-IP name resolution issues can interfere with the locating and mounting of iSCSI LUNs during the boot process.
- Use SnapDrive to create LUNs for use with Windows to avoid complexity.
- Calculate disk space requirements to accommodate for data growth, Snapshot copies, and space reservations.
- Be sure to leave automatic Snapshot scheduling off as configured by SnapDrive.

5.2.4 SNAPSHOT WRITABLE LUNS OR FLEXCLONE COPIES

Snapshot backups of SQL Server databases are read-only point-in-time images. This protects the integrity of the Snapshot backups. In certain situations, SMSQL will restore a LUN in a Snapshot copy for temporary read/write access (restored to an alternative location using a writable Snapshot copy)³. Writable Snapshot copies are used during a verification process, when a DBA restores a database to an alternative location for recovering from operational mistakes or in some cases when external archive backups take place. A restored writable Snapshot copy is for short-term use only.

³ **Note:** A writable Snapshot copy or Snapshot writable LUN is a LUN that has been restored from a Snapshot copy and enabled for writes. A file with an .rws extension holds all writes to the Snapshot copy.

Best practice: Avoid creating Snapshot copies on volumes with active Snapshot writable LUNs⁴

Creating a Snapshot copy of a volume that contains an active Snapshot copy is not recommended because this will create a “busy Snapshot”^{4,5} issue. Avoid scheduling SMSQL Snapshot copies when:

- Database verifications are running
- Archiving a LUN backed up by Snapshot to tape or other media

⁴ **Note:** There are two ways to determine whether you have a busy Snapshot copy: (1) View your Snapshot copies in FilerView®. (2) Use the following Data ONTAP command to list the busy Snapshot copies:

```
snap list usage VolumeName BusySnapshotName
```

⁵ **Note:** A Snapshot copy is in a busy state if there are any LUNs backed by data in that Snapshot copy. The Snapshot copy contains data that is used by the LUN. These LUNs can exist either in the active file system or in another Snapshot copy. For more information about how a Snapshot copy becomes busy, see the *Block Access Management Guide* for your version of Data ONTAP.

5.3 STORAGE MIRRORING AND DISASTER RECOVERY

NetApp SnapMirror technology mirrors data to one or more NetApp storage volumes over a local area network (LAN) or wide area network (WAN) to another NetApp storage system. Once source and destination relationships are established, a SnapMirror baseline transfer initializes the mirror to create a replica of the source on the destination. SnapMirror maintains the synchronization of the replica through efficient, incremental updates that use bandwidth intelligently to transport deltas across a WAN.

5.3.1 MIRRORING FREQUENCY

The frequency of SnapMirror update events (in most environments) is most often determined by the frequency of SMSQL backups. SnapDrive triggers SnapMirror updates upon completion of a SnapManager backup procedure. A supplemental type of Snapshot copy optimized for use with SnapMirror called “rolling Snapshot copies” can be used to augment standard SQL Server backups. Rolling Snapshot copies are controlled outside of SMSQL by using the scripting executable `sdcli.exe` for SnapDrive. Be mindful of some rolling Snapshot best practices discussed below.

Best practice: When using supplemental rolling Snapshot copies

- Be sure to thoroughly read the SMSQL Installation and Configuration guide section “Minimizing your exposure to loss of data.”
- Be sure to configure rolling Snapshot copies to occur only between SMSQL backup and SnapMirror processes. This will prevent overlapping Snapshot schedules.

Key SnapMirror concepts and tips:

- SnapMirror relationships are based on sources and destinations.
- A destination updates its mirrored copy or a replica of its source by “pulling” data across a LAN or WAN when an update event is triggered by the source. The pull events are triggered and controlled by SnapDrive.
- Consistent with the pulling nature of SnapMirror, relationships are defined by specifying sources from which the destination NetApp storage systems synchronize their replicas.
- Destination systems are identified on source systems by assigning destinations privileges using the `snapmirror.access` protocol access option or by inclusion in the `snapmirror.allow` file.

As discussed earlier, SnapManager is integrated with SnapDrive, which interfaces directly with a NetApp storage system by using either iSCSI or FCP disk access protocols. SnapManager relies on SnapDrive for disk management, controlling Snapshot copies, and SnapMirror transfers. Figure 1 illustrates the integration of SMSQL, SQL Server 2005/2008, SnapDrive, and NetApp storage devices.

5.3.2 CHECKLIST FOR CONFIGURING SNAPMANAGER AND SNAPMIRROR

1. Install (using CIFS setup) both storage systems into the same Windows domain as SQL Server.
2. Configure the SQL Server, SnapDrive, and SnapManager services to use the same logon service account.

3. Make sure the SnapDrive service account has the "Log on as a service" rights in the Windows Server 2003/2008/2008R2 operating system (normally occurs during installation).
4. Verify that RSH commands work between the SQL Server instances and both storage devices using the specified service account.
5. License and enable FCP or iSCSI on both storage devices.
6. License and enable SnapMirror on both storage devices.
7. Establish SnapMirror relationships.
8. Make sure the storage device's SnapMirror schedule is turned off by assigning the "- - -" value (four dashes separated by spaces) in the `custom schedule` field.
9. Initialize the mirror configurations.
10. SnapMirror updates to the specified destinations will occur after the completion of every SMSQL backup.
11. Use SnapDrive and FilerView to verify the successful completion and state of the configured mirrors.

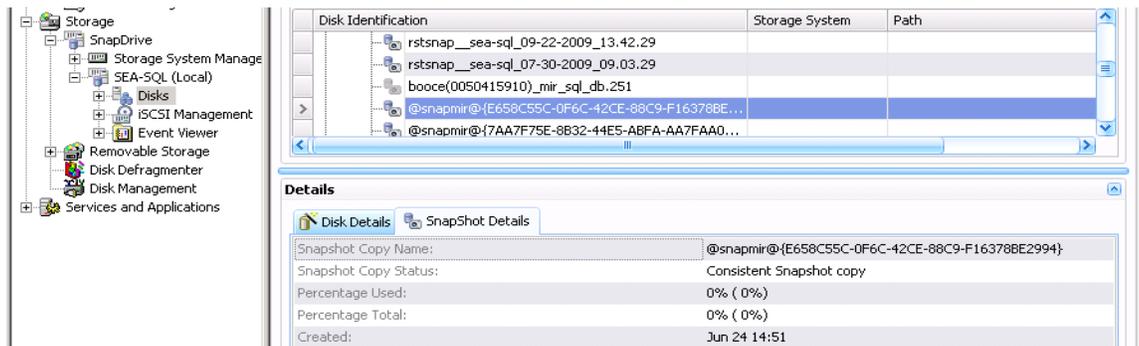


Figure 1) SnapMirror status in SnapDrive.

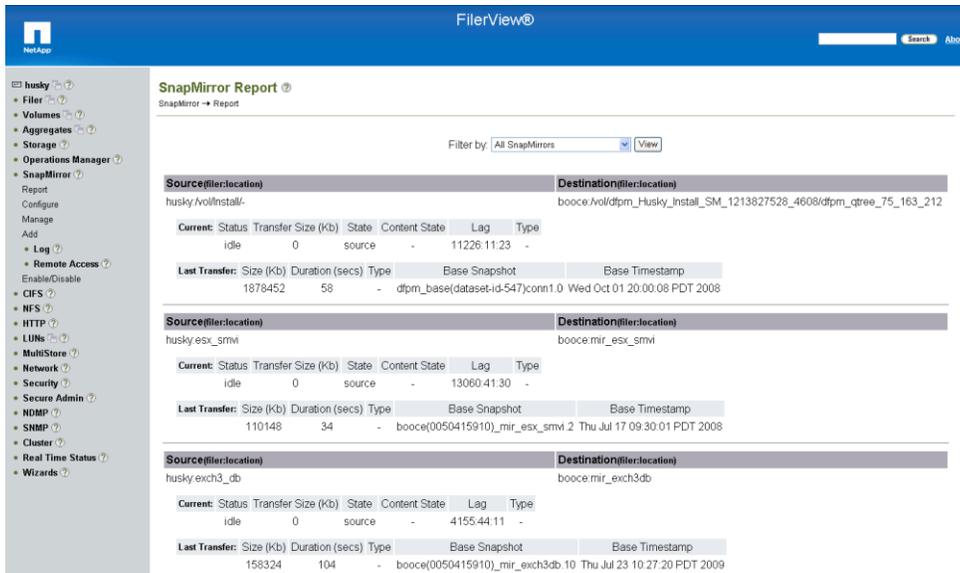


Figure 2) SnapMirror status in FilerView.

6 ONGOING MANAGEMENT AND MONITORING

6.1 SNAPSHOT RESERVE USAGE MONITORING

The task of monitoring the Snapshot reserve is automatically configured at the time of LUN creation. Simply monitor the application event log for warning events generated in the SnapDrive source and Snapshot Monitor event categories. Figure 3 demonstrates that due to Snapshot consumption, the reserve must be expanded to 63%, and there is not enough disk space available to do so. To rectify this situation, simply grow the FlexVol volume, remove some of the oldest SMSQL Snapshot copies, or add new disk resources to the aggregate.



Figure 3) Space alert.

6.2 DISK SPACE USAGE MONITORING

The amount of disk space used by each database managed by SQL Server 2005 and 2008 should be monitored to make sure that the logical drives (LUNs) do not run out of space. The default settings for SQL Server databases will automatically extend its preallocated file space when there is not enough free file space to extend a table (if the database has been defined with `auto extend on`). SQL Server will stop processing transactions when it cannot extend its file space.

Monitoring free space has to be done on two levels. The first level is to free preallocated space that is available for table extents. If Perfmon or a similar monitoring tool such as Microsoft Systems Center Operations Manager (SCOM) is already used for monitoring the system performance; it is easy to add counters and rules to monitor database space. A database's properties will show its current size and how much space is free. When free space gets below or close to 64KB, the preallocated file space will be extended. The second level to monitor is free LUN space, which can be done with Windows Explorer. You might also use the system stored procedure `sp_databases` to show the database files and sizes.

6.3 FRACTIONAL SPACE USAGE MONITORING

In menu click Options->"Fractional Space Reservation Settings" to get the "Current Status" dialog.

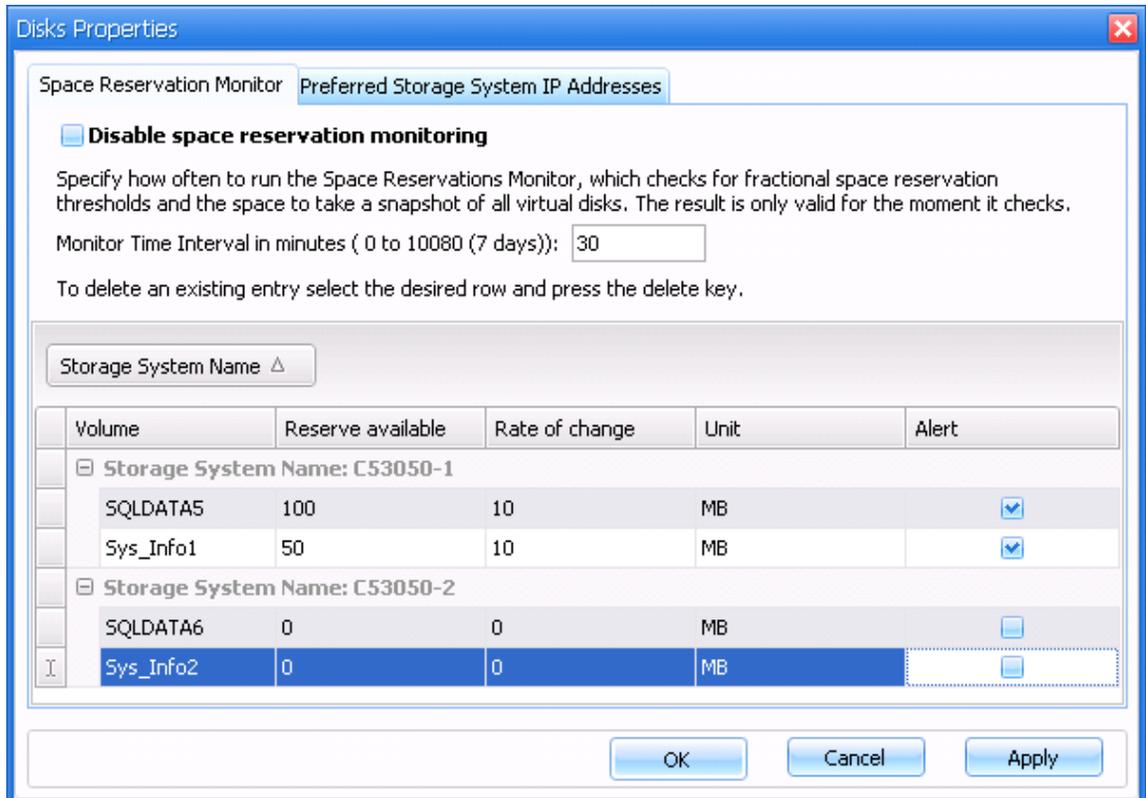


Figure 4) Space Reservation Monitor.

In the Fractional Space Reservation Settings dialog box, use the Current Status tab to view the current space consumption in the storage system volumes that contain LUNs that store SQL Server data or SnapInfo directories. You will see the following information:

DRIVE LETTER OR MOUNTPOINT

The drive letter or NTFS mountpoint on which the LUN is mounted.

FRACTIONAL OVERWRITE RESERVE (%)

The amount of space reserved for overwrites on the storage system volume that contains the LUN. It is expressed as a percentage of the total size of all space-reserved LUNs in the volume.

BACKUP AUTODELETE TRIGGER (%)

It is the setting for the storage system volume that contains the LUN. It is expressed as a percentage of overwrite reserve utilization that triggers automatic deletion of SQL Server backup sets.

DISABLE DATABASE TRIGGER (%)

It is the setting for the storage system volume that contains the LUN. It is expressed as a percentage of overwrite reserve utilization that triggers automatic disabling of SQL Server databases.

USED OVERWRITE RESERVE (%)

The amount of overwrite reserve in use for the storage system volume that contains the LUN. It can be expressed in two ways:

- As a percentage of the total size of all space-reserved LUNs in the volume
- In megabytes of the total size of all space-reserved LUNs in the volume

AVAILABLE RESERVE (MB)

This is amount of overwrite reserve available for the storage system volume that contains the LUN.

6.4 NETAPP STORAGE SYSTEM EVENT MONITORING

Monitor the storage system's event logs to make sure of proper operation. Issues might be discovered in the event logs that require administrative action. One such action might be to replace a disk in a volume if spare disks are not available.

This task can be completed by using the FilerView utility built into Data ONTAP. Start the utility by pointing a Web browser to http://storage system name/na_admin. Next, click the FilerView link. FilerView will start and ask for the credentials of a storage system administrator. Once logged on to FilerView, click the storage system menu option on the left side of the screen. Then choose the Syslog Messages menu option. Review the log on the right side of the screen for any issues that might require administrative action. For more information on FilerView, refer to the Data ONTAP System Administrator's Guide.

If you use Microsoft Systems Center Operations Manager to monitor your IT infrastructure, you should consider NetApp ApplianceWatch™ PRO to monitor your NetApp storage using SCOM.

6.5 TERMINAL SERVER OR REMOTE DESKTOP SERVICES

Microsoft Remote Desktop Services (formerly known as Terminal Server) are not recommended as a way of administering SnapDrive or SnapManager for SQL Server. When you create LUNs from a remote desktop session, the drives can look like they were not created or have been disconnected when in fact they are functioning properly.

Best practice: Use alternative means for remote management

NetApp recommends avoiding Remote Desktop Services for server management when possible. When using Remote Desktop Services with Windows Server 2003/2008/2008R2 (not Windows 2000 Server), you might use a remote desktop session to connect if you use a command line parameter `/console`, shown below:

```
%SystemRoot%\System32\mstsc.exe /console
```

7 SUMMARY

By providing the rich backup and restore features of SMSQL 5, performance and utilization optimizations of FlexVol, and robust data protection capabilities of SnapMirror, NetApp provides a complete data management solution for SQL Server environments. The recommendations and examples presented in this document will help organizations get the most out of NetApp storage and software for SQL Server deployments. For more information about any of the solutions or products covered in this TR, contact NetApp.

APPENDIXES

APPENDIX A: SMSQL BEST PRACTICES RECAP

Best practice: Test cluster failover

Make sure that the cluster can fail over and all the resources are available before you initially establish the SQL Server instances.

Best practice: Use FlexVol volumes when more granular control is required

When backup requirements are unique to each database, there are two possible paths that can help optimize Snapshot copies and transaction log backups:

- Most granular control: Create separate FlexVol volumes for each database for more granular control of backup schedules and policies. The only drawback to this is approach is a noticeable increase in administration and setup.
- Second most granular: Try to group databases with similar backup and recovery requirements on the same FlexVol volume. This can often provide similar benefits with less administration overhead.

Best practice: Avoid sharing volumes between hosts

With exception to Microsoft Windows failover clustering, it is not recommended to let more than a single Windows server create and manage LUNs on the same volume. The main reason is that Snapshot copies are based at the volume level, and issues can arise with Snapshot consistency between servers. Sharing volumes also increases chances for busy Snapshot copies.

Best practice: Set trigger option to volume

When configuring space reservation policies in Microsoft application environments, NetApp recommends setting the trigger option to `volume`.

Best practice: Use a prefix for Snapshot copies

It is recommended to use the `Prefix` value in SQL Server environments that use SnapManager for SQL Server. This allows the user to define a prefix for Snapshot copies on which deletion should be deferred.

Best practice: Determine additional space required

Add a safety net percentage to the total volume size. This additional space can help guard against low-volume space conditions.

Best practice: Minimum backup frequency

All backup and recovery processes *should be* created according to each environment's unique requirements. Determine what the RPO and RTO are, then set full backups and transaction log backups appropriately.

Best practice: Always archive or mirror Snapshot copies

Archiving or mirroring backup sets as soon as possible is strongly recommended. Disasters do occur, and if the storage device containing the databases and backup Snapshot images is adversely affected, it will not be possible to restore the databases from the primary storage system. Archive media can be tape, another FAS system or a device enabled with NearStore. The archive process is described in further detail in SMSQL 5 Installation and Administration Guide.

Best practice: Tips for creating LUNs

- When specifying a UNC path to a share of a volume to create a LUN, use IP addresses instead of host names. This is particularly important with iSCSI, as host-to-IP name resolution issues can interfere with the locating and mounting of iSCSI LUNs during the boot process.
- Use SnapDrive to create LUNs for use with Windows to avoid complexity.
- Calculate disk space requirements to accommodate for data growth, Snapshot copies, and space reservations.
- Be sure to leave automatic Snapshot scheduling off as configured by SnapDrive.

Best practice: Avoid creating Snapshot copies on volumes with active Snapshot writable LUNs⁴

Creating a Snapshot copy of a volume that contains an active Snapshot copy is not recommended because this will create a "busy Snapshot"^{4,5} issue. Avoid scheduling SMSQL Snapshot copies when:

- Database verifications are running
- Archiving a Snapshot backed LUN to tape or other media

⁴ **Note:** There are two ways to determine whether you have a busy Snapshot: (1) View your Snapshot copies in FilerView. (2) Use the following Data ONTAP command to list the busy Snapshot copies:

```
snap list usage VolumeName BusySnapshotName
```

⁵ **Note:** A Snapshot copy is in a busy state if there are any LUNs backed by data in that Snapshot copy. The Snapshot copy contains data that is used by the LUN. These LUNs can exist either in the active file system or in another Snapshot copy. For more information about how a Snapshot copy becomes busy, see the *Block Access Management Guide* for your version of Data ONTAP.

Best practice: When using supplemental rolling Snapshot copies

- Be sure to thoroughly read the SMSQL Installation and Configuration guide section “Minimizing your exposure to loss of data.”
- Be sure to configure rolling Snapshot copies to occur only between SMSQL backup and SnapMirror processes. This will prevent overlapping Snapshot schedules.

Best practice: Use alternative means for remote management

NetApp recommends avoiding Remote Desktop Services for server management when possible. When using Remote Desktop Services with Windows Server 2003/2008/2008R2 (not Windows 2000 Server), you might use a remote desktop session to connect if you use a command line parameter/console, shown below:

```
%SystemRoot%\System32\mstsc.exe /console
```

APPENDIX B: CONSOLIDATION METHODOLOGIES

Consolidation of SQL Server databases, servers, and storage can provide great ROI benefits and optimize hardware, software, and administrative resources. This section will identify general approaches and methodologies in consolidating SQL Server environments. Much of the information presented in preceding sections will be put into practice in a simulated customer deployment scenario.

B.1 PHASE I: ANALYSIS OF EXISTING SQL SERVER INSTANCES

This section will highlight some of the key planning phases that should be undertaken in the beginning stages of a SQL Server consolidation project by focusing on a fictitious company, "ACME Corp.," which has a total of 11 databases running on four dedicated SQL Server instances on direct-attached storage (DAS) and wants to consolidate all instances into a two-node cluster configuration with NetApp storage.

The analysis phase begins with collecting information about existing databases in terms of bottom line criticality to the business. Table 3 illustrates the results of collecting information that is both technical and business related. The high-availability index is a value used to depict the business-critical characterization of each database and will be used as a deciding factor in determining which backup and SnapMirror policies are implemented for each database. The high-availability index works on a five-level designation with an assignment of 1 being the most critical and 5 signifying the lowest level of criticality.

Table 3) ACME's current SQL Server information.

SQL Server	Database	Level
SQL Server 1	DB 1	1
	DB 2	1
SQL Server 2	DB 3	2
	DB 4	1
	DB 5	2
SQL Server 3 DB 7 1	DB 6	2
	DB 7	3
	DB 8	3
SQL Server 4 DB 10 3	DB 9	3
	DB 10	3
	DB 11	1

Table 3 shows a completed table of information describing all databases intended for migration to the new environment along with the ranking of importance to the business. Once the databases have been identified and ranked, the next phase of determining key technical metrics concerned with size, change rate, growth, and throughput can proceed.

It can sometimes be difficult to get data describing I/O throughput during peak loads (holidays, end of quarter, or other peak times), but this data can be very valuable when sizing storage devices correctly for current and future loads at peak times. Table 4 shows the type of data that can be very helpful when sizing the target storage device.

Table 4) ACME's I/O throughputs at peak load.

Database	Database Size	Read per Second (MB)	Write per Second (MB)	Change Rate per Month	Growth Rate per Month
DB 1	270GB	5	1	15%	5%
DB 2	8GB	8	1.5	10%	8%
DB 3	170MB	20	6	12%	6%
DB 4	2GB	0.5	0.1	5%	40%
DB 5	270MB	1.5	0.2	20%	10%
DB 6	43GB	1	0.5	10%	5%
DB 7	500GB	8	2	5%	1%
DB 8	320GB	5	1.2	50%	10%
DB 9	77MB	4	2	5%	5%
DB 10	100MB	2	3	10%	10%
DB 11	3112GB	20	10	10%	5%
Totals	4.872TB	75MB/Sec	27.5MB/sec		

Throughput data can be gathered with several native tools provided by the base OS and the SQL Server database engine. Appendix C uses one such tool by leveraging the dynamic management view `sys.dm_io_virtual_file_stats` available in SQL Server.

```
select db_name(mf.database_id) as databaseName, mf.physical_name,
       num_of_reads, num_of_bytes_read, io_stall_read_ms, num_of_writes,
       num_of_bytes_written, io_stall_write_ms, io_stall,size_on_disk_bytes
from sys.dm_io_virtual_file_stats(null,null) as divfs
join sys.master_files as mf
  on mf.database_id = divfs.database_id
 and mf.file_id = divfs.file_id
```

By understanding the reads and writes you can determine the performance needs of each database. Then determining what databases can be grouped together by function, I/O rates, and recovery times will help to guide the consolidation effort.

Growth rate in terms of space can also be calculated based on projected system growth from a user growth capacity and business perspective. Things such as database groom tasks, "data churn," and Snapshot and transaction log retention should also be taken into consideration.

If unsure how to calculate the total number of disks required, contact NetApp for an official SQL Server and NetApp storage sizing request. This will make sure the right number of spindles is designated for your organization's unique workload characteristics.

B.1.1 ACME'S REQUIREMENTS

New business guidelines for ACME have dictated that all four SQL Server instances be consolidated onto two newly purchased, two-unit servers in an active-active cluster with a storage back end of a FAS3050 using eight full shelves of Fibre Channel disks. A single aggregate model has been selected consisting of a total of 112 144GB drives using RAID-DP with a group size of 16 disks.

The resulting configuration takes the following things into consideration:

- DB1, DB2, DB4, and DB11 have been designated as requiring the highest level of availability and require a full backup every 30 minutes and a transaction log backup at every 15 minutes with immediate SnapMirror replication to a disaster recovery site.

- DB3, DB5, and DB6 do not have as stringent requirements for recovery and can have their transaction logs backed up every 30 minutes with a full backup every hour. Off-site SnapMirror replication is required for DR.
- DB7, DB8, DB9, and DB10 have lower backup and recovery requirements and can be backed up every hour and will not use SnapMirror.
- All databases will need to keep at least three Snapshot copies and additional transaction logs online at all times.
- Management has not budgeted for all databases to be mirrored and has allocated limited space to the DR site.

B.2 PHASE II: DESIGNING AND MIGRATING TO THE NEW ENVIRONMENT

Based on the variety of requirements specified by ACME, along with hard data that was collected and displayed in Tables 3 and 4, a reference diagram was drawn up for the new SQL Server architecture (see Table 5).

All databases were migrated to the FAS3050 with rules applied in the following areas:

- **Size and growth:** All user database LUNs have been sized by a factor of $2x + \text{delta}$ and have been calculated with additional room for growth for the next 12 months. DB11, sized at just over 3TB, has been split into multiple data files and spread evenly over four LUNs. For the critical database, you want to size the volume using the formula $(1 + \text{fractional reserve}) * \text{LUN size} + \text{amount of data in Snapshot copies}$. Also total space reserved for overwrites is calculated using the formula $\text{TOTAL SPACE PROTECTED FOR OVERWRITES} = \text{FRACTIONAL RESERVE SETTING} * \text{SUM OF DATA IN ALL LUNS}$.
- **High availability:** The base aggregate has been constructed using RAID-DP, and dual paths with MPIO have been used for connectivity. WFC has been implemented for server-level failure. The threshold value for the policy “deletion of backup sets” is set less than that of “dismount databases” to prevent databases from going offline.
- **Backup and recovery:** All databases that did not have conflicting requirements from a performance perspective have been logically grouped in FlexVol volumes by backup and recovery requirements. When possible, databases were also grouped in FlexVol volumes for granular control over SnapMirror replication.
- **Performance:** Some databases with more stringent performance requirements have had data and log files separated into different FlexVol volumes for optimal performance.

Table 5) Recommended database placement for ACME migration.

Instance	Database	Size	Database Type	H/A Index	LUN Mount (Device)	FlexVol Volume	SnapMirror
Quorum disk	Quorum data	100MB			Q	1	No
Node 1	System DBs	10GB		1	U	2	No
	DB 1	270GB	OLTP	1	E	3	Yes
	DB 2	8GB	OLTP	2	F	4	Yes
	DB 3	170MB	OLTP	1	G	3	Yes
	DB 4	2GB	OLTP	2	H	5	Yes
	DB 5	270MB	OLTP	2	I, J	5, 6	Yes
	DB 6	43GB	OLTP	1	K, L	7, 8	Yes
	DB 7	500GB	OLTP	3	Y	9	No
	DB 8	320GB	OLTP	3	N	9	No
	DB 9	77MB	OLTP	3	O or C:\mntpt0\db1	9	No
SMSQL	SnapInfo			1	P	10	Yes
Node 2	System DBs	150GB		1	W	11	No
	DB 10	100MB	DSS	2	S	11	No
	DB 11	3112GB	DSS	2	M, R, T, V or C:\Mntpt1\db1 C:\Mntpt2\db2 C:\Mntpt3\db3 C:\Mntpt4\db4	12, 13	Yes
SMSQL	SnapInfo			1	Y	14	Yes

KEY OBSERVATIONS:

- Cluster Server 1 groups databases on FlexVol volumes based on a backup policy of transaction log and database backups every 15 minutes and 30 minutes, respectively. Databases are also grouped by FlexVol volume for different SnapMirror policies and frequency.
- DB 5, DB 6, and DB 11 are all using two FlexVol volumes each to separate log and data files for optimal data locality because performance requirements for these databases are more stringent.
- The system databases for both instances have dedicated FlexVol volumes. The instance on node two is hosting the largest database, DB 11. The system database volume is sized at 150GB to allow ample room for growth of `tempdb` for large query sorting and indexing that take place in the DSS database DB 11.
- DB7, DB8, and DB9 are rated 3 on the high-availability index and are not configured for SnapMirror. They also all reside on the same volume to prevent them from being backed up with other schedules or unnecessarily mirrored.
- Snapinfo has also been designated for mirroring to make sure that both metadata and transaction log backups are transmitted to the SnapMirror site.
- Because the cluster is active-active and must allow for takeover, no drive letter is duplicated on either active node.
- We can use volume mountpoints and not drive letters to break down a huge database into separate LUNs, hence avoiding the issue of running out of drive letters.

The ACME example illustrates a successful implementation using optimal SQL Server file placement, volume, and LUN layout while staying within the confines of drive letter limitations and SMSQL backup requirements. All files, LUNs, aggregates, and FlexVol volumes were implemented to provide just the right level of granular control over performance, backup, and disaster recovery policies.

APPENDIX C: REFERENCES

TR-3373: Data ONTAP 7G: The Ideal Platform for Database Applications
<http://media.netapp.com/documents/tr-3373.pdf>

TR-3402: Microsoft SQL Server 2005 Performance and Scalability Testing Using NetApp FAS920 Storage Systems
<http://media.netapp.com/documents/tr-3402.pdf>

TR-3410: Data ONTAP 7G: FlexVol and FlexClone for Microsoft SQL Server 2000
<http://media.netapp.com/documents/tr-3410.pdf>

TR-3563: NetApp Thin Provisioning
<http://media.netapp.com/documents/tr-3563.pdf>

TR-3604: NetApp Disaster Recovery Solution for Microsoft SQL Server 2005 User Databases
www.netapp.com/us/library/technical-reports/tr-3604.html

TR-3696: Microsoft SQL Server 2005 Relational Engine: Storage Fundamentals for NetApp Storage Systems
www.netapp.com/us/library/technical-reports/tr-3696.html

TR-3734: SAP on Windows and Microsoft SQL Server with NetApp Storage and VMware
<http://media.netapp.com/documents/tr-3734.pdf>

TR-3768: SMSQL Best Practice Guide
<http://media.netapp.com/documents/tr-3768.pdf>

TR-3779: Sizing Guidelines for SQL Server 2005/2008 for NetApp Solutions
<http://media.netapp.com/documents/tr-3779.pdf>

WP-7078: Development and Test Solution for Microsoft SQL Server
<http://media.netapp.com/documents/wp-7078.pdf>

NetApp Solutions for Microsoft SQL Server
<http://media.netapp.com/documents/microsoft-sql-server-overview.pdf>