THE WINDOWS MANAGER'S GUIDE TO **iSCSI SANs**

INSIDE:

CHAPTER 2

Gearing up

for SANs on a SQL Server

Gearing up
for SANs on a SQL Server

Using iSCSI SANs on a SQL Server can improve performance, reduce infrastructure costs and facilitate system scalability. Get details on how to configure an iSCSI SAN for optimal SQL Server performance and growth.

BY DENNY CHERRY

ONE MAIN BENEFIT to using iSCSI to attach your SQL Server to a storage area network (SAN) is that it reduces SAN infrastructure set-up costs. There also are no additional fibre network costs to consider when configuring your solution, which means that there are no delicate fibre cables to handle.

iSCSI SANs allow you to attach a few dozen individual hard drives to each logical unit number (LUN). As your system grows, additional hard drives can be added into the RAID group to improve performance capabilities.

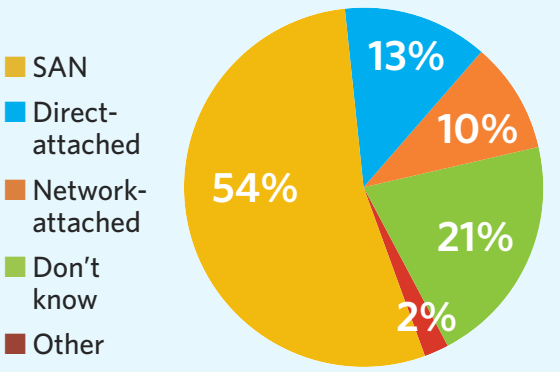
Using an iSCSI SAN solution also facilitates environment scalability, allowing you to create the initial LUN at the required size. However, if storage requirements change, you can extend the LUN and the Windows partition on the drive using the DISKPART tool (or DISKPAR in Windows 2000).

Standard Ethernet networks within

data centers are 1 GB in size, with 10 GB networks being ratified. Typically, however, normal TCP/IP traffic underutilizes these networks within the data center. iSCSI uses this existing network connectivity between the server and the storage.

In a 2007 survey of 182 database administrators, 54% use SAN for SQL. This storage solution dominates over direct-attached solutions, which ranked second at 13% (Figure 1).

FIGURE 1: **SAN is predominant storage architecture for SQL DBAs**



SOURCE: TECHTARGET

A MOVABLE SOLUTION

With other storage solutions (Fibre Channel, JBOD, etc.), the SQL Server and the storage array must be located in the same physical location, often within a short distance of each other. iSCSI solutions allow the server and storage to be located anywhere, as long as there is a high-speed, low-latency network between them. While the systems should be placed fairly close to one another, if the network can support the storage traffic, the systems can be physically located anywhere.

When selecting a physical location for your servers and storage, keep in mind that it takes time to move data throughout these locations. Data takes about 3 microseconds per kilometer of distance traveled over fiber optic cable. If your storage and server are 100 kilometers apart, it will take at least 300 microseconds for each data packet to go from the storage array to the server. That means it will take 600 microseconds for the request to be sent to the storage and return the data packet.

While this may not seem like much of a delay for SQL Server, it can be quite an interruption. In addition to this nearly 1-millisecond delay, you have to take into account the additional time it takes to retrieve the data from the disk and prepare it for transport. While iSCSI can be used to send data great distances, special care must be taken when designing the network to handle this data traffic. For instance, keep distances as short as possible, especially when you must transfer large amounts of data from the SQL Server to the storage. As the distance increases, the amount of data that must be sent in real time must be reduced so as not to degrade performance.

CONFIGURING iSCSI INITIATOR

The beauty of iSCSI solutions is that they don't have any specific hardware requirements. Connectivity between the storage and the server occurs through an iSCSI initiator. iSCSI initiators are either hardware or software initiators.

iSCSI solutions don't have any specific hardware requirements. Connectivity between the storage and the server occurs through an iSCSI initiator.

A hardware initiator is a physical expansion card that's installed in the SQL Server with a dedicated Ethernet connection between the hardware initiator and the network switch. A software iSCSI initiator uses an existing network connection to create the link between the server and the storage.

While a hardware initiator is recommended, it isn't required. There are several solutions in which a software initiator will work just as well. The higher the load on the SQL Server, the more you may want to consider a hardware initiator. Because the software initiator is a software driver running within the server's operating system, all disk access must go through the software. This can zap SQL Server's memory and CPU resources. If you aren't sure if a hardware initiator is required, you can start with a software initiator and switch to a hardware initiator later, as needed.

When designing a redundant infrastructure for your iSCSI environment, two physical paths from the server are recommended to separate Ethernet switches. Additionally, multiple Ethernet paths should be established in your storage device. This will help spread the load between additional initiators and will allow your SQL Server to survive a network switch or network cable failure.

DEPLOYING AND TUNING AN iSCSI SAN

Deploying a SQL Server into an existing iSCSI solution is fairly easy. To do so, follow these steps.

1. Create LUNs on the iSCSI storage and assign them to the new host's initiator. The initiator will detect the LUNs and present them to the operating system as local storage. You may not see their volumes immediately, but you can query for them by right-clicking on *Disk Management* within the Computer Management console and selecting *Rescan Disks*. iSCSI volumes will show up as SCSI disks.

2. Import the volumes into the system as either basic or dynamic disks. You don't need to RAID the disks

within Windows; they're already configured on the storage array. However, basic disks are recommended.

With an iSCSI storage solution, there are a couple of places within the network that require specific tuning: the Ethernet network and the physical storage. With your Ethernet network running at peak performance, turn your attention to the storage.

Avoid creating Windows partitions on drives using the Computer Management console when deploying iSCSI volumes to a host. This can cause an issue within the boot sector's configuration.

3 Ensure that your storage isn't queuing too many requests per spindle. Queued requests should be measured at the RAID-group level, not only the LUN level. The number of acceptable operations varies between each array and vendor. Typically, a formula such as (n*3) is used, where *n* is the number of disks in the RAID group.

If the number of operations being queued is above this value, you may need to add spindles to the RAID group. Some iSCSI solutions treat the entire storage array as a single RAID group. Therefore, if queuing occurs, you must add additional drives to the array.

When deploying iSCSI volumes to a host, don't create Windows partitions on the drives using the Computer Management console. This can cause an issue within the configuration of the boot sector of the Windows volume itself. This issue appears in all storage solutions including the local disk, JBOD, SAN, iSCSI, etc., on all operating systems.

4 After you have rescanned the iSCSI bus and imported the disks into Windows, open a command prompt and launch DISKPART.EXE. Using the **SELECT** command, choose the disk on which you want to create a partition.

5 Use the **CREATE PARTITION** command to create the primary partition with an alignment of 64 using this syntax:

```
CREATE PARTITION PRIMARY
ALIGN=64
```

This should be done for all disks you're adding to the system. After creating the partitions, refresh your Computer Management window, assign drive letters and format your disks as usual. Not creating volumes this way will incur additional overhead for each physical read and write on the storage. For more information,

refer to the article Optimize disk configuration in SQL Server.

ISSUES AND WORKAROUNDS

Like all storage solutions, iSCSI is not a black box that you can simply fill with data and leave alone. It also must be understood by database and Windows administrators. If too much load is placed on a single LUN or RAID group, performance will suffer and will possibly affect other systems that share the RAID group.

When looking into SQL Server performance issues, you must know the I/O profile of both your SQL Server and any other systems that are sharing your LUN's physical disks. Most OLTP systems have a random I/O profile, as do most file systems.

OLAP databases typically have a sequential I/O profile. Systems with random profiles should not share physical hard drives with systems that have sequential profiles, unless the systems aren't used simultaneously. If systems share physical hard drives and attempt to use them at the same time, the random pattern of the OLTP database will randomize the sequential I/O pattern. Proper and

When looking into SQL Server performance issues, you must know the I/O profile of both your SQL Server and any other systems sharing the LUN's physical disks.

WHAT IS... OLTP and OLAP?

► OLTP, or **online transaction processing**, is a class of program that facilitates and manages transaction-oriented applications, typically for data entry and retrieval transactions in a number of industries. New OLTP software uses client/server processing and brokering software that allows transactions to run on different computer platforms in a network.

► OLAP, or **online analytical processing**, enables a user to easily and selectively extract and view data from different points of view. To facilitate this kind of analysis, OLAP data is stored in a multidimensional database, which considers each data attribute as a separate "dimension." OLAP software can locate the intersection of dimensions and display them. Attributes such as time periods can be broken down into subattributes. —WHATIS.COM

regular defragmentation of both the disk and the database can fix this.

The path from the server to the storage is another common storage bottleneck. With Fibre Channel solutions, the cost per port is very high, meaning that adding more fibre ports to the SQL Server is expensive. However, with the relatively low cost of Ethernet ports, adding additional ports to an iSCSI solution is more cost efficient. ■

Data security over the LAN

WHEN WORKING WITH iSCSI, an additional layer of security must be considered as data is being transferred from the database over the Ethernet network. Consider isolating your SQL Servers and storage from the rest of the Ethernet network.

This can be done in a couple of ways. You can physically isolate the servers on a separate network; however, this is a costly configuration. Another solution is to configure virtual LANs (VLANs) within your existing Ethernet network. This will isolate network traffic between the storage and SQL Servers from the rest of the network, preventing others on the network from sniffing your network traffic.

Some initiators and storage arrays support over-the-wire encryption, which, as its name implies, encrypts data as it travels over the wire. This solution taxes the initiator's processor as well as the storage array's processors. It should only be used in the most extreme cases—where traffic must travel over an unsecured network such as the Internet, for example. Systems that need to encrypt data traffic over the wire should always use a hardware initiator. A software initiator isn't feasible for a system that needs to encrypt iSCSI traffic.

Because iSCSI presents storage volumes over an Ethernet network, securing those volumes within the storage array and the operating system is highly recommended. Ensure that the storage array allows only the correct host to mount the volumes, and secure the data within the volumes using NTFS best practices of granting only the minimum rights required.

ABOUT THE AUTHOR:

Denny Cherry *has more than a decade of experience managing SQL Server, including MySpace.com's more than 175-million-user installation, one of the largest in the world. Denny's areas of expertise include system architecture, performance tuning, replication and trouble-shooting. He uses these skills in his role as a senior database administrator and architect at Awareness Technologies. Denny is a longtime member of PASS and Quest Software's Association of SQL Server Experts and has written numerous technical articles on SQL Server management.*

SearchSQLServer.com

The Windows Manager's Guide to iSCSI SANs is part of the Windows Manager's Guide series.

Marilyn Cohodas
Editorial Director
mcohodas@techtarget.com

Michelle Boisvert
Editor
mboisvert@techtarget.com

Martha Moore
Copy Editor
mmoore@techtarget.com

Linda Koury
Art Director of Digital Content
lkoury@techtarget.com

Jonathan Brown
Publisher
jebrown@techtarget.com

Matt Salamone
Senior Director of Sales
msalamone@techtarget.com