



Storage Networking, Part 2: Configuration and Planning



an internet.com Storage eBook

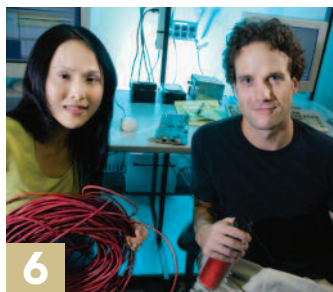
10100101011011010010101101010110110010101001
0100100110011001010100011100101010010
0100101110010010010101001001001
11101110010101001010101101010101
10100101011011010010101101010110110
01001001100110010101000111001010100100001010101

contents

Storage Networking, Part 2: Configuration and Planning



This content was adapted from Internet.com's Enterprise Networking Planet Web site and was written by Charlie Schluting.



2 Configuring Disk Arrays

4 Configuring SAN-Attached Servers



6 Reaping the Benefits of a SAN

10 Planning Your Fabric



12 Understanding Storage Routing

14 Is IP Storage Viable?

Configuring Disk Arrays

By Charlie Schluting

In Part 1, we discussed storage area networks (SANs) and fibre channel. Here at the outset of Part 2, we'll delve into best practices and cover the general concepts you must know before configuring SAN-attached storage. The most critical, sometimes tedious, part of setting up a SAN is configuring each individual disk array.

There are three general steps when configuring a disk array:

- First, you create a RAID set. It can be any type of RAID the array supports, and we'll just assume RAID-5 in this example so we can talk about hot spares.
- You can either slice up the RAID set to present multiple logical unit numbers (LUNs) to a host, or you can create "RAID Groups," as most vendors call it. This is a completely optional step, but it can make your life easier.
- Third, you must assign LUNs to a host.

Create a RAID Set

The first step can be done many ways. Say you have an array that holds 14 disks per tray, and you have four

trays. One option is to create two (or more) RAID-5 volumes on each tray. You can then assign part or all of each RAID-5 volume to various hosts. The advantage to this method is that you will know which hosts use what specific disks. If the array with three additional trays was purchased at the same time, it actually makes more sense to allocate the RAID sets vertically, so that a single tray failure doesn't take out the RAID volume.

With only four trays this means you'll have three disks worth of usable space per 4-disk RAID-5 volume: probably not a good use of space.

More often people will create huge RAID-5 sets on the arrays. There's a balance between performance and resiliency that needs to be found. More disks mean better performance, but it also means that two disk failures at once could take out all

of your data. Surprisingly, multiple disk-at-once failures are quite common. When the array starts rebuilding data onto a previously unused disk, it frequently fails.

Configure RAID Groups

The second step causes quite a bit of confusion. Regardless of how you've configured the RAID sets in the array, you'll need to bind some amount of storage



Jupiterimages

More disks mean better performance, but it also means that two disk failures at once could take out all of your data.

to a LUN before a host can use it. The LUN can be an entire RAID-5 set (not recommended), or it can be a portion. The partitioning method ensures that you aren't giving too large a volume to a host. There are many reasons for this:

- Some file systems cannot handle a 1TB or larger volume
- Your backup system probably won't be able to backup a file system that's larger than a single tape
- The important one: more LUNs presented to the host (seen as individual disks by the OS) means that separate I/O queues will be used

Back to the second step: RAID Groups. A RAID-5 set of 1TB partitioned, for example, into 100GB chunks, will provide 10 LUNs to deal with. If you don't care what nodes use what disks, you can just throw these LUNs into a group with other LUNs. I prefer to keep one RAID group per host, but others see that as limiting flexibility. Some hosts need a dedicated set of disks, where you know that only one host will be accessing the disks. A high-traffic database server, for example, should not have to contend with other servers for I/O bandwidth and disk seeks. If it truly doesn't matter to you, simply create a bunch of LUNs, and assign them to random groups.

It is also important to create and assign "hot spare" coverage. Spare disks that are left inside the array are "hot" spares. They can be "global," so that any RAID volume in the event of a failure uses them, or they can be assigned to specific RAID volumes. Either way, you need to ensure you have a hot spare, if you can afford the lost space. If not, be sure to monitor the array closely — you'll need to replace any failed disk immediately.

This is where it gets tricky. Different storage arrays will have different terminology, and different processes for assigning LUNs or groups of LUNs to a host.

Assign Your LUNS

Step three, "assign LUNs to a host," means that you're going to map WWNs to LUNs on the array. If you didn't, then any host zoned could properly see all the volumes on the array, and pandemonium would ensue. Be

cautious about certain cheaper storage arrays, too. They may not even have this feature by default, until you purchase a license to enable it. While the purveyors of limited-use technology call this feature "WWN Masking" or "SAN-Share," the market leaders in the SAN space realize that it's required functionality.

The most common approach is to create a "storage group," which will contain "hosts" and "LUNs" (or RAID groups with many LUNs). Whatever diverging terminology is used, the universal concept is that you need to create a host entry. This is done by manually entering in a WWN, or connecting the host and zoning it appropriately so that the array can see it. Most arrays will notice the new initiator and ask you to assign it a name. Once your hosts, and all their initiator addresses, are known to the array, it can be configured to present LUNs to the host.

There is one final note about array configuration. You'll be connecting two HBAs to two different fabrics, and the array will have one controller in each fabric. The host needs to be configured for multipathing, so that either target on the array can disappear and everything will continue to function. We'll discuss host configuration later, including multipathing and volume managers, but be aware that the disk array side often needs configuring too. The majority of disk arrays require that you specify what type of host is being connected, and what type of multipathing will be used. Without multipathing, LUNs need to be assigned to specific controllers, so that the appropriate hosts can see them.

Once LUNs are assigned to a host, they should be immediately available to the operating system, viewed as distinct disks.

Think about this for a moment. You've taken individual disks, and combined them into RAID volumes. Then you've probably partitioned them into smaller LUNs, which are handled by the disk array's controllers. Now the host has ownership of a LUN, comprised of possibly 10 different disks, but each LUN is smaller than individual disks. The host OS can choose to stripe together multiple LUNs, or even partition individual LUNs further. It's quite fun to think about. ■

Configuring SAN-Attached Servers

Connecting a host to your shiny new SAN is not the same as connecting a single disk, or even a direct-attached SCSI array. Let's examine the reasoning behind current best practices, and explain how to configure your storage for optimal reliability.

Direct attached storage arrays, if you have used them, offer a great introduction into the world of storage. They have LUNs to configure on the array itself, and then you must deal with them at the host level. As storage sizes have increased, so too have the demands on sysadmins to configure storage in a usable and reliable manner. It may have been acceptable to assign 10 20GB LUNs to 10 different partitions in the past, but 200GB is not much storage any longer.

First, let's define a few steps that should be taken before we start to think about file systems. Before creating file systems, the following must happen:

- Configure the array, as described earlier, to assign LUNs to your host.

- Attach fiber, one from each card, to two switches in distinct fabrics.
- Zone both switches appropriately so that the initiator and target are both visible to each other.

- Verify you see all LUNs.

- Configure multipathing: path failover.

The last step is the tricky part, depending on your operating system and disk array. We'll get to that shortly.

Attaching the fiber is self-explanatory, assuming we understand the concept of keeping each "path" to the storage in separate fabrics. Zoning the switches takes considerably more knowledge, but is very vendor-specific. Brocade, McData, and Cisco switches vary immensely, but the concepts are global. Decide how to zone, and apply the configuration.

At this point, you should be able to "see" the new LUNs on the server. In Windows, opening the Disk Manager should bring the new volumes to light (some report a reboot may be required). Linux, at least recent



Jupiterimages

Connecting a host to your shiny new SAN is not the same as connecting a single disk, or even a direct-attached SCSI array.

versions, should immediately discover the new LUNs. In Solaris you'll need to run 'cfgadm' and possibly 'devfsadm' to see your new LUNs.

If you only have a single path to the storage, you're almost there — it's time to create file systems. The vast majority of SAN-attached hosts, however, will have two paths to the LUNs, so the host will see the same LUNs twice, once per target. Since the storage array has two interfaces, there really are two targets. The host needs to be made aware of the fact that these are really the same volumes.

Multipathing is a host-based driver, combined with array support, which allows redundant connections to your storage array. If you tried to create file systems on all the LUNs you saw, and then decided to try mounting each one individually, your disk array would (hopefully) scream and yell. There's a concept of "primary controller" defined on your array, and if an initiator tries to access the LUN on the non-primary target without first "downing" the preferred path, the array will protect itself. That's hugely simplified, but a good way to think about it.

If you configured your LUNs to be assigned one-per-controller, alternating, as we recommended last time, then your host will be able to successfully use half of the LUNs. It can create file systems and successfully use each LUN, but only via its preferred controller. The only thing this buys you in the event of controller or switch failure is that only half of your volumes will disappear. What we really want to do is abstract our device paths, and mount the abstracted device. Using multipath device nodes means that the underlying "actual" devices can disappear at random, and as long as the driver and storage array get along well, the operating systems will never see a mounted disk device disappear.

Actually configuring multipathing is less than trivial. If you want to make life easier, use Veritas Volume Manager with DMP (Dynamic MultiPathing). It runs on

all operating systems, and it works the same in each. You'll also get the added bonus of using operating system-neutral file systems, in case the need arises to move volumes between platforms.

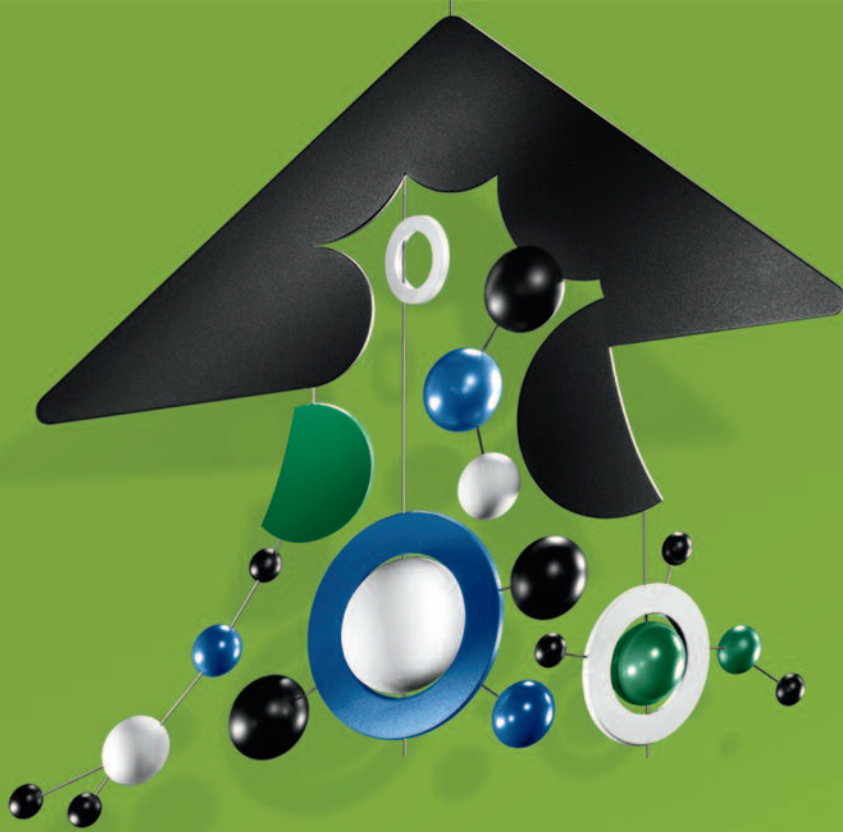
If you're unable to use DMP, you still have two options. The first is to try getting a driver from the storage manufacturer. If the array you purchased was sold with support for your operating system, chances are good that you simply need to install the vendor's driver, and you're off and running. If not, then you get to try to use whatever native multipathing driver your OS includes.

Solaris, for example, has excellent multipathing support. It works very well with storage that Sun has blessed, but may not work at all with some storage. It's a crapshoot; hopefully you did your homework before purchasing the array.

Once multipathing is configured, you'll have one set of devices that you're free to play with. The actual devices are abstracted now, so you want to make sure that you're using the multipath device nodes, not the physical paths.

Now comes the fun part. You get to plan and implement your file system layout. Be extremely careful here, because even with a volume manager as flexible as Veritas or ZFS, you'll still be working yourself into a corner if the wrong decisions are made. The decisions are highly use-specific, so the best advice that can be given is to think carefully. Most people will want to stripe some amount of LUNs together to make larger file systems, but not too large that you can't back it up in a sane amount of time. Too large a file system also means that repairing damage can take excruciatingly long, too.

Of course, don't forget to save your switch and array configurations somewhere safe, and document your multipathing and file system decisions. The best part about multipathing is the testing stage. Go ahead, start copying a huge file and yank the fiber. ■



ALTERNATIVE THINKING ABOUT VIRTUAL STORAGE:

VIRTUALIZE STORAGE NOW.

A powerful business innovation in data storage is now within your reach. The new HP StorageWorks 4400 Enterprise Virtual Array is here. It virtualizes up to 96TB of storage—across numerous storage servers and platforms—simplifying storage management and speeding access. Less limitations. More freedom. Technology for better business outcomes.

HP STORAGEWORKS EVA4400

Up to 96TB virtual storage capacity.

- Enterprise-class performance
- Over 30% better capacity utilization*
- Up to 75% less time needed to configure and manage*
- Easy application integration

Now's the time for virtual storage.
Visit hp.com/go/virtualstorage



Reaping the Benefits of a SAN

We promised to talk about some fancy things you can do with a SAN; things that otherwise would be impossible. These include global file systems and the ability to move storage between servers with virtually no downtime. In fact, the whole way you think about storage, and its hassles, can be thrown out the window.

The benefits of a storage network are seemingly endless. Your storage suddenly becomes fault-tolerant because you can lose a fibre channel switch, a disk array controller or a host HBA and everything should continue working. With the right selection of products, a company can also employ less people to manage storage thanks to the wonderful tools available for dealing with configuration tasks. Data security and storage utilization are also increased with a careful SAN deployment.

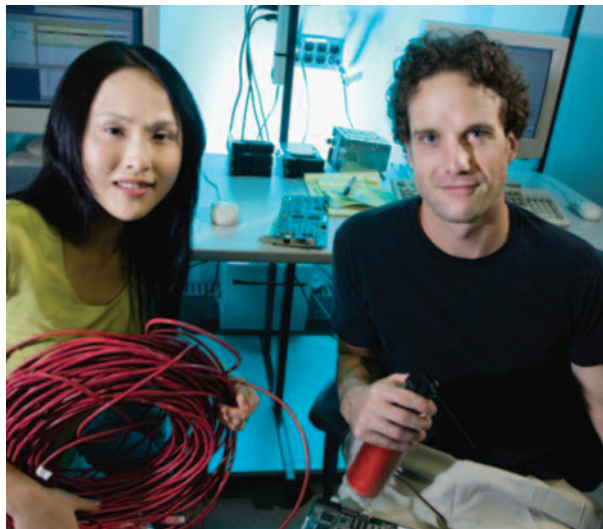
These days it isn't hard to sell the technology to the business leaders, and hopefully the storage administrators are sold as well. Less employee overhead is attractive to both administrators and managers, but the real benefit for the administrators, aside from reli-

ability and all that important stuff, is the neat things you can do with a SAN.

Storage: Here, There, Everywhere

Perhaps the most exciting prospect of moving to a SAN environment is the introduction of easy storage migration. In an old, direct attached storage (DAS) environment, if some storage needed to move to a new server, this would involve a lengthy process. We'd first unmount the file systems using the storage, then physically unplug the unit, move it, connect it to a new host, and bring up the file systems on the new machine. What a hassle! In fact, it was probably more common to need to allocate more storage to just a few LUNs. If the disk array was already full, you'd have no choice but to copy the file systems to a new array.

In a SAN environment, let's take a look at the same process for both scenarios. The first, needing to move "some storage" from one host to another, involves only three steps, none of which include leaving your chair. If we want to move an entire array from one host to



Jupiterimages

Don't forget to save your switch and array configurations somewhere safe, and document your multipathing and file system decisions.

another, all we need to do is unmount file systems, reconfigure the SAN to let the other host see the storage, and then quickly bring it up on the new server. This task can be done in a less than a minute if you're good, but in a DAS world, 15 minutes would be record-breaking.

With DAS environments, it was necessary to reallocate storage by moving the entire array to a new host. If you needed fast RAID storage on a server, it was necessary to connect a new array. Even if the application couldn't actually use all of the storage, it would still get connected to the host that needed some storage. In the SAN world, we can make much better use of our storage.

The second scenario mentioned above is most common. We generally need to grow the size of just a few file systems, rather than replace an entire array. If you have already allocated all LUNs on your DAS unit, you're stuck moving data off the array. In a SAN environment, we simply need to allocate a new LUN to a host.

Generally, we'll have a disk array with two controllers and around 15 FC disk drives. Extra "trays" can be stacked onto and managed by the controller array. If storage starts getting tight, simply add another tray. Amazingly, most arrays also support the ability to grow LUNs, so that taking advantage of added disks doesn't always involve creating new LUNs. You must run a file system that supports this, however.

Even if you do need to add storage from another array to your server, there are still no wires to touch. As long as everything is SAN-attached, you can easily allocate LUNs from multiple arrays to one server. As mentioned, it's a completely different mindset from the DAS days.

Without all the hand waving, we should talk a bit about moving LUNs and adding storage to existing file systems. You really should run some type of volume management soft-

iSCSI Rides Virtualization Wave

By Drew Robb

• SCSI continues to gain market share and momentum. While the IP SAN technology has certainly come in for its share of hype, the sheer number of deployments commands respect, and trends like server virtualization and faster Ethernet connections could accelerate that growth.

"Continued, steady growth has been the watchword for iSCSI over the past year," says John Webster, an analyst at Illuminata.

According to Steve Duplessie, founder and senior analyst at Enterprise Strategy Group, there have been 25,000 iSCSI production implementations to date.

That may seem like a lot, but according to Brad Nisbet, IDC's storage systems program manager, iSCSI commands just a 3 percent share of the total external disk storage systems market. But with growth rates approaching triple digits, IDC expects iSCSI's market share to reach 21 percent by 2010.

LeftHand Networks, for one, a pioneer in the IP SAN market, reports more than 2,000 customers and 5,000 deployed systems running its flagship SAN/iQ product. Such numbers signify the emergence of iSCSI beyond its initial use by small IT shops who couldn't afford a Fibre Channel SAN. While it retains this audience, it has been able to scale upwards successfully into some high-end enterprise deployments.

Commerce Bank and Trust based in Topeka, Kan., is a LeftHand customer. It runs 50 servers at its data center to serve 22 branches. Each branch uses workstations backed up by LeftHand appliance.

"You have to keep things very simple at the branch level, as it's just not humanly possible to manage multiple branches individually," says Steve Haas, Commerce Bank's IT Security Officer. "If you can get a server up and running, you can easily set up an IP SAN."

Big Vendors Take the Lead

But LeftHand isn't the only game in town. Most vendors, even FC SAN stalwarts like EMC and NetApp, are now big in iSCSI. In fact, NetApp and EMC are No. 1 and No. 2 in the market, respectively, according to IDC.

"Some smaller players like EqualLogic are taking off, but NetApp is the major iSCSI vendor at the moment," says Webster.

ware, such as VxFS from Veritas, ZFS from Sun, or even one of the various Linux-native solutions. When we talk about adding a LUN to a file system, what we're really talking about is software-level RAID; generally a RAID 0 stripe. We need some way of stitching together multiple LUNs at the host-level to create file systems. To add more space to the file system, we simply attach a new LUN. This is trivial in VxFS and ZFS.

Now what if we need to move a live file system from one host to another? Again, we ideally want file system help, but this isn't mandatory. The idea is to "export" the file system from one host, reconfigure the SAN (storage array to allow a new host to see the LUN and zoning on the switch, if necessary), and then import the file system on the new server. This is trivial with ZFS and Veritas, but a very manual process if you're using a standard file system.

Clustering

SAN file systems are file systems that can be utilized by more than one server at once. Try this with a standard file system, and in short order, your data will be corrupt. The file system must have the ability to coordinate changes with all others that wish to modify the file system at the same time.

Clustered file systems are extremely useful, especially for highly available file servers or databases. Without a SAN environment, using these SAN file systems is almost impossible. To be fair, there have been a few SCSI DAS arrays that included two ports and allowed two hosts access at the same time, but those were rare and expensive.

There are few options, but the file systems are maturing quite rapidly. They are:

- GFS: Global File System, for Linux.
- Xsan: for OS X.
- OCFS: Oracle Cluster File System, for Oracle databases.
- VMFS: for VMware.

Duplessie's figures show EqualLogic with several thousand customers (and acquiring more at a rate of 400 to 500 a quarter). EqualLogic's PS Series arrays can be joined together to form a 100 terabytes-plus combo.

Patagonia, a manufacturer of outdoor clothing and technical apparel based in Reno, Nev., uses the EqualLogic PS200E. It has a capacity of 5.6 TB.

"Our deployment of EqualLogic has been what was promised: short configuration times, up and running in minutes, and exceptional ease-of-use and management capabilities that help us grow our storage networks as needed," says Tammy Barrett, network engineer at Patagonia.

Virtual Success

While iSCSI has certainly established its own market momentum, it has gained further traction from a couple of high-powered sources. iSCSI is now riding the coattails of two massive storage drivers — VMware and Microsoft.

VMware, of course, is part of the EMC empire. But it has caught the imagination of the server marketplace in a big way.

"iSCSI seems to have an affinity with VMware," says Webster. "Find a solid VMware user and you're likely to find iSCSI."

VMware puts multiple virtual servers on a physical box, and makes it easy to move these virtual servers around. If you have 200 virtual machines across three physical systems, for example, you need to have all physical and virtual machines capable of accessing the same storage. iSCSI is a perfect way to enable that easily.

"iSCSI makes VMware capable of quickly moving from lab environments to production ones — and they are so hot it's pulling a lot of iSCSI SANs along with it right now," says Duplessie.

Accordingly, vendors such as LeftHand are building additional functionality into their products. LeftHand's SAN/iQ storage virtualization and management software can now be combined with VMware server virtualization software to facilitate storage consolidation and virtualization.

"The SAN/iQ and VMware combination is the most sought-after solution by our customers today," says John Spiers, LeftHand's founder and CTO. "We believe the market is headed towards easy-to-manage, virtualized data centers with convergence around IP."

SAN/iQ's feature set includes virtualization, grid-like clustering, pay-as-you-grow, thin provisioning, local and remote replication and snapshots, all managed from one screen.

Expect to see the open source clustered file systems improve in the future. With the widespread adoption of storage networks and higher demands on services, the need for clustered file systems will continue increasing.

We can certainly see that a SAN environment provides for more manageable storage allocations. No more late nights copying data, no more plugging cables to move arrays between hosts, and no more single points of failure. What's not to love? ■

Microsoft, too, is fueling the iSCSI fires. Duplessie points out that Windows now ships with iSCSI initiators included. The iSCSI initiator is the code in the server OS that lets it talk to an iSCSI target (array). Microsoft began giving it away as a free download in 2005. Now it's automatic in Vista and Longhorn. By now, there are an awful lot of machines out there ready for iSCSI. And those numbers are only going to get bigger.

10 Gig on the Way

These major market dynamics could be in for yet another boost once 10 Gbps Ethernet gets going. Analyst firm Dell'Oro Group revealed that the 10 Gigabit Ethernet Switch market surpassed port shipments in excess of 100,000 in the fourth quarter of 2006, with annual revenues topping \$1 billion during 2006. But the 10 Gb market continues to be haunted by high prices. Once the prices come down significantly, there could be no stopping iSCSI.

"We already have a number of customers asking for 10 Gig, and expect to be making announcements around our 10 Gig products in the coming months," says Spiers. "If you look at each Ethernet bandwidth adoption curve, the big inflection in demand comes when prices hit a certain point."

He reckons that point will be reached soon. A new standard has been ratified that introduces copper into the 10 GbE marketplace. That should help considerably with the high price point of current 10 Gb switches.

"We currently don't have any IP SAN initiatives," says Jim Burgard, University of New Orleans' Assistant Vice Chancellor for University Computing and Communications. "When 10Gb Ethernet interfaces become widely available for servers, we will revisit this option."

The potential of iSCSI, then, is staggering. It has faced up to a series of barriers over the last few years, and passed them with flying colors.

"We've proven performance isn't an issue, though the perception still exists," says Duplessie. "It's hard to argue with an interface based on a ubiquitous standard (Ethernet). And with folks like VMware being able to really take advantage of simple, cheap networked block storage, the numbers are only going to accelerate." ■

Planning Your Fabric

Once a SAN fabric grows beyond the initial two SAN switches, it's time to start planning and designing your network. SANs can be easy to slap together, but so are Ethernet networks. Both require some careful planning for scalability. Let's discuss some common SAN design principles and help you plan for expansion.

Experience has shown that migrating to a SAN environment is best done in stages. Perhaps you'll still have many NAS or DAS devices lingering for a while after your initial SAN investment — that's OK. The most dangerous method of migrating to a SAN environment, barring tons of consulting dollars, is to replace everything in one fell swoop. When adding SAN equipment piecemeal, however, it helps to have a big picture end-result in mind.

The end result, for many businesses, is a total SAN environment. All servers should have two paths to the SAN, one to each distinct fabric. All storage, and servers requiring storage, are connected to the SAN. In theory, this is great, but it gets expensive quickly. Then people start to think they don't need to follow best

practices in SAN design, because "that much redundancy is overkill." Sometimes, in really small businesses, it is. More often, it is required. Be extremely careful about dismissing redundant switches, even when it seems like the redundancy is becoming overkill.

The Design

Industry consensus is that a core-edge design is ideal.

A core-edge design means that you have two sets of core switches, one for each fabric that individual nodes will connect to, and then fanned-out switches from there. The model is simple when you can have core devices that are truly redundant and highly available. You connect the core switches to a few edge switches, and the edge switches connect to end-node devices, like servers.



Jupiterimages

A "director class" switch is a SAN switch that has built-in redundancy, performance and scalability features to meet extremely flexible needs. These SAN switches are very, very expensive, but they make great core switches. The advantage to purchasing your "core" in a single (actually, it should be a pair of directors per fabric)

The benefits of a storage network are seemingly endless. Your storage suddenly becomes fault-tolerant because you can lose a fibre channel switch, a disk array controller or a host HBA and everything should continue working.

is that you have fewer devices to manage. The same redundancy and performance requirements can be met with a little creative engineering.

The most common core design, for people lacking the director-class switches, of course, requires more standard SAN switches. Instead of a dual-director core (per fabric), each director can be replaced with a pair of switches. Each switch will connect to the others, meaning you'll have to burn three ports just to configure the core. Luckily, Fibre Channel is smart enough to deal with redundant paths, so it's safe to do this. The cost of initially using four ports on cheaper switches is minuscule compared to the price of a director switch.

With both designs, director and non, the next step is to connect edge switches. The number of edge switches you require is completely dependent on the number of nodes that need to connect. Furthermore, you also need to plan for capacity. Having enough available ports doesn't mean you have the capacity. Edge switch throughput is generally good, internally, but you must be careful not to place an extremely popular set of servers behind a single 2Gb link. Yes, 4Gb links exist now, but the problems don't go away. Thankfully we can use another port between two switches, and combine the available throughput. These aggregated ISLs, or inter-switch links, allow us to continue using edge-class switches in the core, even when we've scaled our usage beyond original planning.

Connecting the Pieces

It helps to think of the "core" as being only two-sided: One set of ports connect to hosts, the others connect to storage, backup, or everything else. Every single device will connect to two fabrics via two different edge switches, each with their own core. To provide the best availability, the edge switches must also con-

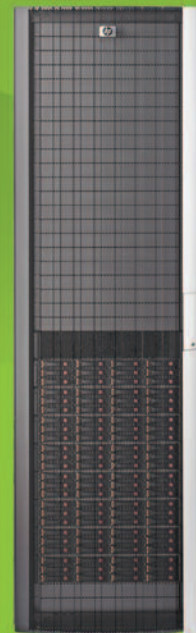
nect to each half of the individual core. If this isn't making sense, search Google for "SAN design" images. There are much prettier images from Brocade and Cisco than I could ever craft.

Do note that ISL links will likely be required throughout the core, depending on a few variables. If your links are all 4Gb, and data transfer rates are generally low, you're probably safe. However, if there is a storage array that serves 20 servers, as well as tons of other traffic, a bottleneck can be difficult to track down. Your core switches definitely need enough throughput to support all aggregate traffic through them, but so do the edge switches. A common mistake is to create a core capable of 10Gb/s, but fail to realize that most of that traffic comes from a single storage array. If the array is connected to the core via a 4Gb link, there's another bottleneck.

New SAN deployments do not usually run into bottlenecks right away, but when it happens, it's always at the most inopportune time. Ideally we would like to plan for and engineer out the possibility of bandwidth contention.

Bandwidth issues aside, we also need to plan for scalability and redundancy. Both aspects are inherent in the edge-core design, but there are, in fact, other schemes you may consider. It's very tempting, and cost effective, to start growing "pairs of switches" everywhere, and then just begin linking them together. That's fine, but pretty soon you'll have a chain, and straight lines are easily broken, even if they're comprised of two parallel lines (remember, pairs of switches).

If you'll never scale past six switches, you'll likely be fine just connecting them together in a circle, but most businesses will need to scale, eventually. ■



ALTERNATIVE THINKING ABOUT STORAGE:

UNIFY STORAGE.

Self-optimizing storage is more powerful and cost-effective. So the new HP StorageWorks 4400 Enterprise Virtual Array unifies viewing and access of up to 96TB of storage through data pooling and automatic capacity allocation, to dramatically simplify managing storage. Bringing storage together saves times and money. Technology for better business outcomes.

HP STORAGEWORKS EVA4400

Up to 96TB virtual storage capacity.

- Enterprise-class performance
- Over 30% better capacity utilization*
- Up to 75% less time needed to configure and manage*
- Easy application integration

Now's the time for virtual storage.
Visit hp.com/go/virtualstorage2



Understanding Storage Routing

Storage networking is not unlike IP networking, most of the time. In the IP world there are numerous routing protocols and standards; you have dozens of options. In the storage world, however, there are no official routing protocols. Routing does exist, though it may not be what you're imagining.

Storage Area Networks can generally be thought of as huge Layer 2 networks. SANs implement a mechanism not unlike Spanning Tree from the Ethernet world to keep themselves loop-free. The big problem with sprawling networks is that one problem can impact the entire network. In a SAN environment this gets more exacerbated due to the nature of the FC protocol itself. One way to combat problems associated with too-large fabrics is to isolate them into distinct networks.

Everyone knows that IT silos should be avoided, but sane network design often requires them. When a SAN sprawls too wide, stability needs often dictate the creation of multiple fabrics. This does not mean you've created a standard "bad" IT silo, just that you've created two separate fabrics. The good news is that in much the same way IP networks operate, we can route traffic between fabrics.



Jupiterimages

In IP networking we must have unique IP addresses, but it doesn't really matter if MAC addresses overlap if they aren't in the same subnet. Fibre Channel has no Layer 3 addresses, so the Layer 2 address, or World Wide Name, must be globally unique. In SAN routing, there are two ways to "route" traffic: by translating, or virtualizing, world wide names (WWNs) into fake ones on the other fabric, or by spoofing the address. The

fact that WWNs must be unique shouldn't be a problem because they are carefully assigned, but it does help to visualize the difference between layering and translating.

Remember, there is no protocol for SAN routing. Everything we're talking about here is vendor specific, unlikely to interoperate with other vendors' products, and subject to interpretation and bias when evaluating the effectiveness of such mech-

anisms.

Routing by Termination on the Fabric

The first method of SAN routing is best thought of as a proxy server, albeit an extremely smart one. McData Corp. developed a mechanism to connect multiple SAN silos together. When configured appropriately, switches can masquerade as both a target and an initia-

The most dangerous method of migrating to a SAN environment, barring tons of consulting dollars, is to replace everything in one fell swoop.

tor, essentially proxying a FC connection between two SAN fabrics. When a port is terminated on a SAN switch, administrators still have the same amount of flexibility when configuring access to LUNs, and in many cases, but not as much security.

In Part 1, we talked about the differences between hard and soft zones. If we need to ensure a certain amount of security, both against attackers and configuration errors, we prefer to use port-based, or hard zones. When configuring these silo connections, translations, or mappings, as we'll call them, the only choice is to configure WWN based restrictions. While unlikely to cause a problem, it is something to be aware of. Part of a procedure for replacing HBAs usually includes updating WWN mappings on switches and storage arrays, and now you need to make sure "routing" configurations get made aware of any changes too.

Terminating SAN connections on each silo switch is nevertheless advantageous. The likelihood of SAN-wide outages is greatly minimized, and isolating problems can be much quicker in a router environment. SAN routing is also heavily used in geographically dispersed networks to increase the reliability and stability of the network as a whole.

Other Routing Methods

SAN routing can also take the form of various other technologies. It turns out that routing in a SAN is just segmentation, and the routing glue to make it work. It's precisely the same in IP networks, except that there are clear mechanisms for dealing with passing packets between domains: layers and routing protocols. SANs need to segment for the same reasons as IP networks, but with the additional and looming stability issue added in.

Many people view protocol encapsulation and translation as routing. The FC over IP (FCIP) and even the iSCSI protocol are in a sense routing protocols. They enable SAN extensions over larger IP networks without

adding to the sprawl of a fabric. These technologies are frequently used in remote SAN replication to a backup site. You certainly wouldn't want to extend an entire fabric to another city, especially when only one device needs connectivity. The iSCSI target is normally hosted on a storage device, so calling it a routing mechanism is a bit of a stretch. Some SAN switches can act as a translator between FC and iSCSI, though, making the role of a router even clearer. In fact, that's the exact same thing an IP node does — take Layer 2 data and add Layer 3, or IP data on top of it.

SAN virtualization gets even hairier. Certain applications of LUN pooling and the subsequent translation into iSCSI are clear-cut candidates for being called routers, but in general, virtualization isn't really routing. Storage virtualization is precisely what the very first method of routing did. When a SAN switch presents its own LUNs, which are in fact hosted on storage array elsewhere, it is creating a virtualized storage device. Virtual LUNs enable some of the more creative ways to go about routing in a SAN fabric, but the concept of virtualization itself isn't really about routing.

On the other hand, if you stick to the simple definition of routing — segmenting and subsequently gluing parts together — then virtualization as described above is routing. Virtualization takes many other forms, however, such as: LUN pools, remote replication, and snapshots. These other uses of virtualization don't facilitate segmentation of a network.

Yes, it is confusing. There's no routing, even though practice has proven it to be necessary. Tricky segmenting, piecemeal glued together with virtualized LUNs, is all we really need. When real routing is needed, for instance, when you need to ship FC packets across the Internet, we simply use IP. It works, so why not just use existing routing infrastructure. Long distances imply high latency anyway, so the main speed benefits of an FC SAN (block-level access and the avoidance of protocol encapsulation) aren't as meaningful. ■

Is IP Storage Viable?

The media loves iSCSI: it is inexpensive to implement, and offers many advantages over DAS or NAS solutions. Instead of focusing on how iSCSI works, we're going to delve into some prevailing myths of IP storage. After understanding the important aspects of the FC and iSCSI technologies, we will be better positioned to make purchasing decisions.

Proponents of iSCSI are quick to point out some attractive facts. These consist of, but are not limited to:

- iSCSI is cheaper because it uses your existing IP infrastructure
- iSCSI is simpler to deploy
- iSCSI is "faster" because you can use 10GbE

These things are true, to varying degrees, depending on the environment and performance needs. Unfortunately, iSCSI cannot meet high-demand enterprise needs.

Performance

Let's first address the performance concerns. The truth is that iSCSI is certainly capable of some impressive throughput. Most any protocol, standard, or device can perform really well in a few niche applications of the technology, and iSCSI does excel in a few areas.

It is no mystery that iSCSI-only devices are targeted to the SMB market. Nobody is claiming you're supposed to put huge transactional databases on iSCSI storage, but the proliferation of iSCSI hype has led to this cost-

saving conclusion for many IT managers. Be extremely careful to avoid this trap.

For many reasons, Fibre Channel (FC) does offer better performance. We often relegate the performance discussion to factors that do not accurately isolate the things being compared. Frequently we see arguments about iSCSI performance turn in to a discussion about how horrible SATA is, simply because many SATA array vendors offer iSCSI support.

Yes, SATA will crumble under a load with lots of random IO, but it has nothing to do with the underlying access protocols.

FC is designed for large block IO and, as such, is heavily optimized to pass storage data around. Ethernet is not, but using jumbo frames (9K instead of 1.5K data units) can alleviate this concern a bit. FC HBA cards are, however, more efficient than Ethernet cards. Without getting into too much detail, FC essentially requires less CPU overhead because Ethernet

usage requires an interrupt for each frame. Ethernet is really designed for small and frequent packet handling, not large data streams. iSCSI also rides on top of TCP, so traffic needs to be passed through many more layers in the OS before actually being sent out on the wire, further increasing latency. TCP checksum offload engines exist, so we exclude the added TCP overhead for practical purposes. In short, FC provides far less latency, and far better throughput. Again, deciding on FC-based SAN versus iSCSI depends on the specific performance needs that you are addressing.



Jupiterimages

Network utilization is another important concern. A primary selling point of iSCSI is that you can utilize your existing network infrastructure; in fact, you can use the same NIC that other IP traffic passes on. This is fine, for casual uses, but a high-traffic (in the TCP/IP sense) servers that also need decent storage access times will clearly suffer with iSCSI deployed over a single network interface. In fact, iSCSI users often find that, because they don't really need good performance, NAS-based solutions (i.e., mounting an NFS or CIFS share) would have worked just as well.

Options exist to help alleviate network congestion at the host. Obviously we can deploy a second network card, and luckily most servers these days ship with two to four Gb network interfaces. When we start talking about the need for 10GbE, we're likely going to be running into other performance issues as was outlined above. Assuming the access characteristics are ideally suited for iSCSI, the 10GbE option does exist.

How will a network handle 10GbE, though? Likely we'd need to upgrade some infrastructure to make this happen. Network congestion in IP storage networks often demands the separation of duties — a completely separate IP network for storage. People might be inclined to directly connect an Ethernet cable to their storage device if 10GbE would put too much strain on the network and there was only one server that needed the performance boost. Unfortunately, that devolves your situation back to DAS (direct-attached) days, and truth be told, a DAS setup could provide much better performance anyway.

Resiliency

The FC world is used to high availability configurations.

Each node connecting to a SAN may use two HBA ports, and "see" each storage LUN twice, but over multiple paths. When configured correctly, these LUNs are accessed from a virtual LUN, and the driver transparently fails over between the actual LUNs in the event of an outage. This is how SAN storage is done, and everyone loves it.

In an FC SAN, we can upgrade or replace storage device controllers and switches with zero downtime. In the iSCSI world, we cannot. Each host is directly connected to a single switch, regardless of whether or not they share the same NIC with TCP/IP traffic. If that switch disappears, there is no failover capability. Some vendors have likely implemented proprietary multipathing solutions for iSCSI, but it is surely only going to work with only one operating system and their own storage device.

The good news in all of this drudgery is that you don't really have to choose one technology over the other. I urge everyone to consider iSCSI as an intermediary between high-performance FC-based SAN and file shares (NAS). Most FC storage array vendors now offer iSCSI support directly on the array itself, enabling businesses to use a hybrid deployment methodology. Your SAN-attached disk array can now be IP-attached, and serve two distinct demands, based on performance and reliability needs.

Don't dismiss iSCSI because of the naysayers, and don't adopt it without careful consideration. ■

This content was adapted from Internet.com's Enterprise Networking Planet Web site and was written by Charlie Schluting.

Internet.com eBooks bring together the best in technical information, ideas and coverage of important IT trends that help technology professionals build their knowledge and shape the future of their IT organizations. For more information and resources on storage, visit any of our category-leading sites:

www.enteprisestorageforum.com
www.internetnews.com/storage
www.linuxtoday.com/storage
www.databasejournal.com
<http://news.earthweb.com/storage>
<http://www.internet.com/storage>

For the latest live and on-demand Webcasts on storage, visit: www.internet.com/storage