Sample SAN Configurations For Open Systems

Mark Bruni

1.0 Overview

This paper will cover Storage Area Networking (SAN) configurations currently announced and supported (as of 1/1/2002) by IBM for various disk and tape devices. For the FAStT200, direct fibre channel connections as well as hub or switch connections will be discussed. For the ESS, native FC configurations as well as those that use the SAN Data Gateway (SDG) or the Interim Host Adapter (ESS feature code 3020) are included. Some generic issues with tape are discussed, followed by All the samples in this paper are for Open Systems (non-S/390) disk configurations. It is intended that this paper will be expanded - or perhaps additional separate white papers will be written - in the future to include FICON for the S/390 as well.

NOTE: In all cases, there are specific sets of operating systems, FC adapters, and devices currently supported by IBM. Since this changes as IBM continues its testing, this paper will merely refer to the URLs where this information is kept up to date. *To ensure that a proposed configuration is supported, you must check with these web pages*. These URLs will be mentioned in-line in the document and are also listed for reference in Appendix A.

1.1 SAN concepts

Storage Area Networking is built around the use of Fibre Channel technology as a networking infrastructure to connect storage devices to various host servers. It can be thought of as a replacement for SCSI buses and ESCON channels. A Fibre Channel port is described as being a 1 Gb or one Gigabit port. After all the overhead is taken into account, this 1 Gb capacity results in 100 megabytes per second (100 MBs) of payload capacity. Fibre Channel today mostly runs at 100 megabytes per second, IBM has recently introduced some products with 2 Gb ports (200 megabyte per second payload), and standards are being worked on for higher speeds. Fibre Channel eliminates the distance restrictions of SCSI - 25 meters - allowing single hop distances as high as 10 kilometers standard (some vendors go up to 100km with high output lasers)- dramatically improving the connectivity options allowed.

Standards are in place to run SCSI commands and data over Fibre Channel networks using SCSI-3 protocols. When run over Fibre Channel this is called Fibre Channel Protocol or FCP. Thus, a host server using a Fibre Channel adapter still assumes the role of a SCSI Initiator, and storage devices that are Fibre Channel attached still assume the role of SCSI Targets. It is just that the SCSI commands and data are now flowing over Fibre Channel links rather than SCSI buses.

For the S/390, IBM supports running ESCON-like protocols over Fibre Channel - called FICON. ESCON was in essence a first generation SAN. It used its own physical infrastructure at around 17 Megabytes per second. FICON improves upon ESCON technology, and takes advantage of Fibre Channel's greater capacity.

1.1.1 Fibre Channel components.

From a "SAN edge device standpoint," a host server or storage device will have Fibre Channel adapters for access to a SAN. Typically, a storage device "adapter" is a Fibre Channel port integrated into the device, but from a low level Fibre Channel standpoint it is just another adapter inside a different device. Host adapters - called Host Bus Adapters or HBAs - are typically purchased from a vendor and installed into a host much as ethernet and SCSI adapters are today.

Hosts and storage devices can be directly attached to one another - called a direct or point-to-point connection with nothing but fiber cable in between - or they can be connected to Fibre Channel hubs or switches. These are analogous to LAN hubs and switches in that a hub provides shared bandwidth - 100 megabytes per second for the whole hub - and a switch provides 100 megabytes/second for each port. (hubs are less expensive per port than switches). Some switches - such as the 2109-F16 - provide 2 gigabit ports which can run at either 1 or 2 Gb depending on the port at the other end of the link. All hubs, however, run at 1Gb providing 100 MBs of shared bandwidth. The IBM Managed Hub (3534) as discussed in "Appendix B - Fibre Channel Detail" on page79 allows a switch to look like a hub, but still perform like a switch. (It also provides SNMP management not found in IBM's hub, the 2103-H07.)

NOTE: The IBM Managed Hub and the 2103-H07 hub are generally supported in the same configurations. Two notable exceptions are the FAStT200 disk subsystem or the iSeries (AS/400) server which do not support the 2103.

A collection of one or more interconnected switches is called a "SAN Fabric." While it is customary and intuitive to think of hubs as being part of a SAN - part of the "collage of Fibre Channel" that hosts and storage devices connect to - only the switch-based part of the SAN is considered "the fabric." This is because there are additional higher level services provided by the fabric (e.g. Fabric Login, Name Services, etc.) which are not available from hubs. (Some of these services can be provided elsewhere, but typically, not in a hub.)

The industry also includes a term "director" which is meant to be a fully-redundant switch with no single point of failure except for the ports themselves. This is really like having 2

switches in one box, one of them in hot standby to take over non-disruptively should the active one fail. Typically, all control logic, power, fans, shared memory and the switching mechanism(s) themselves are duplicated, so that only a port or port-card failure requires manual intervention to recover from (if there is a spare port card in the director then only cable(s) need be moved from failing port(s). ***This document normally uses the term "switch" to mean director or switch, except when distinguishing differences between them as in the following paragraphs.***

Directors are more expensive than switches, and typically have more ports as well. Fabrics built with just switches can achieve redundancy through using multiple switches with multiple paths, such that a switch outage does not remove all paths between any two devices. Before discussing how one would decide which to use, let's look at the switches and directors IBM can provide.

IBM provides switches and directors from three different vendors. From Brocade, IBM OEMs the 2109 switch in 3 models. The 2109-S08 and S16 are 8 and 16-port switches with 1Gb ports. These switches can attach to most any kind of device and are the most widely deployed today. IBM also sells the 2109-F16 which is a 16-port switch with 2 Gb ports (There are more differences between the F16 and S16 then just speed, not discussed in this paper.) Brocade has announced a large director, probably available in early 2002, but as of this writing this is not available. Brocade supports FCP, but not FICON.

IBM resells the McDATA product line which includes a 64 port director (2032-064), a 32-port director (2032-001), a 32 and 16-port switch (2031-032, 2031-016) and a "loop switch" (2031-L00) for FC_AL access. McDATA switches and directors do not support loop connection as do the 2109s, so loop-only devices such as 3590 tape drives must connect to a 2031-L00 which then has an "uplink" to a switch or director. McDATA directors can be configured to run FICON and/or FCP protocols. The switches are FCP only.

IBM also resells a 64-port and a 128-port INRANGE director, that can also run FICON and/or FCP. The INRANGE directors can support FCP loop devices as well.

New products are on the horizon. Besides the Brocade director mentioned above, and all vendors are working on higher capacity (more ports) and faster speeds as well as the ability to mix different protocols (iSCSI, FCP, iFCP) onto the same director.

Some customers will be willing to pay for the "industrial strength" reliability of the directors (indeed some demand it.). Others will opt for a switch fabric with sufficient extra switches to provide high availability. In essence, directors take the "ESCON approach" to high availability, and switches take a "switched LAN approach" to high availability.

The main issues in choosing between switches and directors are the size and complexity of the fabric, and the "pain" associated with downtime or outages. For example, can you tolerate the outages associated with microcode upgrades? Can you live with fewer paths until a new switch is acquired and cabled up?

Since directors have more ports than the16 available on the 2109, large fabrics are easier to build with directors. Large complex fabrics using 2109s will need at least twice as many

switches as the same fabric built with directors and depending on how many ports must be dedicated to inter-switch links, and how big the directors are, the difference can be a factor of 3 to 1, 4 to 1, or even higher. This is not necessarily an expense issue (it may be), it does dramatically increase the complexity involved. Every time you add new switches to a fabric you have to figure out how many ISLs are needed to handle the traffic, this design effort is not trivial, and is greatly simplified when using fewer, higher-port directors

Restoring an installation back to a full configuration after an outage will take longer with switches. Even if extra switches are "on the shelf" to replace a failing switch, much physical re-cabling is required to substitute a switch. With directors, the redundant switch in the chassis takes over immediately (with no loss of data), and the failing switch can then be replaced non-disruptively, basically to "replace the spare tire."

In a port failure, there will often be some loss of data, and there is also the need for manual intervention here to move a cable. For an entire SAN to be fault-tolerant, you need extra HBAs in servers and extra ports on devices. You connect these either to different switches or to different port cards on a director (or to different directors of course.)

Microcode upgrades to 2109s cause short outages, and may need to be scheduled in maintenance windows, whereas with directors (and McDATA switches) the new code is downloaded to the backup switch, a non-disruptive failover is done manually, and the director is now running on the new microcode. Fallback to the previous level of code is just as easy. Once the customer is sure of the new code, it can be loaded into the second switch (now the backup switch) so that the new microcode is running even in the event of a failover.

There is no one correct answer regarding switches and directors. It depends on the costs and pain to a customer for various kinds of outages. Some customers can tolerate short outages quite easily and do not really need non-disruptive code upgrades and immediate non-disruptive failover. Others experience so much pain from any outage (or from the size and complexity of their SAN) that the additional cost of the directors is easily justified.

Today, with the McDATA product line IBM can provide a mix of directors and switches, in which the industrial strength directors are in the core of a SAN (where an outage affects many many servers and devices), and the less expensive "edge switches" will be on the boundary. In the future, when switch-to-switch standards are implemented, multi-vendor fabrics will be possible, yielding more options for mixed director and switch networks. ***For the rest of this document, the term "switch" will be used to mean switch or director***.

NOTE: Very large configurations are possible with all vendors because it is not required that all switches be interconnected. By putting multiple adapters in storage devices and servers you can have them connected to multiple different fabrics (separate, not-connected islands of interconnected switches) and still get all the connectivity you want without exceeding the maximum fabric size for a given switch. Fibre Channel architecture allows for up to 239 switches in a single fabric. Vendors typically have smaller maximums that they have tested and certified, although they will often agree to support larger fabrics when asked.

Given a collection of switches/directors comprising a SAN, it is possible to have multiple paths through the fabric between any two switches/directors, giving multiple paths between a single Fibre Channel host adapter and a single Fibre Channel target adapter. IBM switches and directors can use load sharing between these paths, and have the paths back each other up as well. This is all transparent to the host and storage device (target).

Hubs like the IBM 2103-H07 have operational issues due to loop initialization (see Section1.1.2 on page5). A single hub provides what is called an arbitrated loop, in that only one "conversation" is going on at a time, much like old party lines for telephones. If a hub is connected to another hub they become a single arbitrated loop. One arbitrated loop can have up to 126 devices. Since the bandwidth is shared you wouldn't want that many devices on one physical loop, but there are 126 one-byte loop addresses that may be used.

Hubs can also be connected to switches, in which case the switch port has a loop address also, but this is not part of the 126 addresses for servers and storage devices. (Switches use loop address zero on a loop, and no other device may use this address.) Only one switch can be actively connected to any one loop, however. If two switches are connected to a hub (or set of hubs comprising a single loop), the switches will detect this, and one of the switch ports will remain inactive. A loop "ends" at any switch it is connected to. If there are two hubs connected to two different ports on a switch, that is two different loops with two separate independent initializations, and both switch ports use loop address zero on their respective loops. A loop initialization on one loop has no effect on the other loop, and each loop has its own set of 126 addresses to use.

The IBM 2109 switch can support hubs (loops) attached to it, or loop devices can be directly attached to a 2109. (This produces a small two-station loop between the switch and the device.) The McDATA products do not support any kind of loop attachment, except the 2031-L00 described above for device attachment. The 2031-L00 can be attached to any of the other four McDATA switches. For McDATA boxes other than the 2031-L00 only devices or servers that can do "direct fabric attach:" - meaning no loop arbitration - can attach. INRANGE boxes can support loop devices, but not all combinations have been tested so be sure and check the support URLs to be certain.

Other options for connecting loops to switches are discussed in "Appendix B - Fibre Channel Detail" on page79.

1.1.2 Hubs and Loop Initialization

The configurations in this paper are generally not concerned with fabric versus non-fabric connections, except to note which are supported and which are not, and to point out other important differences. The most important difference to remember, is that an arbitrated loop - one or more hubs - must go through a Loop Initialization Process (LIP) - to assign loop addresses - before any I/O can occur. *This process happens every time a host or storage device joins the loop and it disrupts all existing I/O that may be going on.* Furthermore, in almost every instance, you can't just re-drive the interrupted I/O.

This is because, loop addresses are assigned on a first-come first-serve basis. Thus, when LIP occurs on an operational loop, there is no guarantee that, after a LIP is complete, any given FC adapter will get the same loop address that it had before. (If the loop is attached to a switch, there is no guarantee that you are connected to the same switch or even the same fabric as before.) So you cannot just re-drive interrupted I/O as you take the risk of going to the wrong device. A loop initialization forces all adapters to start all over from scratch, much like a SCSI bus reset.

What this means is, if you have more than one host on a loop, if you boot one of the hosts, when it comes back up it will take down the loop as it causes a new initialization to start. (Once again, similar to having two hosts on the same SCSI bus, one system boots and resets the SCSI bus disrupting any occurring I/O.)

For this reason it is normally recommended to only have one host on any loop. Storage devices entering a loop have the same effect, but they are generally not booted very often and thus having more than one of these on a loop is not too much of a problem, although it still must be managed. In most cases, IBM only supports hubs for “distance extension,” meaning that only two “things” can be attached to the hub - usually a device or server and a switch. Support URLs must be checked carefully.

Finally, implementation of Fabric Address Notification can take the sting out of LIP initialization, as described in “Fabric Address Notification (FAN)” on page82. While FAN support is available on several switches and devices, today IBM only supports this function for 3590 tape drives attached to the 2031-L00. (McDATA ES-1000)

1.1.3 SAN zoning

The good news about a Fibre Channel SAN is that it allows any-to-any connectivity. The bad news about a Fibre Channel SAN is that it allows any-to-any connectivity! This was rarely an issue with SCSI because connectivity was difficult, but now with Fibre Channel it is quite easy for servers to access each other’s storage unless something is done to prevent it. If you merely connect a lot of servers and storage devices to a SAN, and do nothing else, then every server will be able to “see” every LUN in the network. Since most platforms are not ready to behave responsibly in such a free-for-all environment (see “Disk Pooling” on page9), something must be done to limit what a server can “see” when it boots up and scans for storage devices.

In essence, we need to be able to partition the SAN so that only the sharing we want to happen will happen. In a SAN fabric this is called zoning. With zoning, I can “carve” up the SAN into different zones. They may overlap to allow some sharing or be completely separate to prevent sharing. Zoning will be necessary in complex SANs with lots of servers and storage devices in order to keep servers from being able to “see” certain devices.

There is a type of zoning that will not be discussed in this paper, and that is Broadcast Zoning. Broadcast zones affect which ports receive a broadcast from another port. FCP and FICON do not use broadcasts, and thus these zones do not apply to these protocols.

Broadcast zones could be used by a broadcast-based protocol such as IP, but IBM does not support IP over Fibre Channel, and it is rarely used.

Zoning as it is currently implemented on the IBM Fibre Channel switch 2109 can be done two different ways.

- Port Zoning - You can merely restrict which switch ports may get to which other switch ports. In the 2109, when using port zoning, every frame is checked to see whether it is allowed to exit the destination port (called hard zoning). If it is not allowed the frame is dropped. A zone is implemented across an entire fabric and only pertains to ports connected to servers or storage devices. (Ports connected to other switches are, in effect, in every zone.) The disadvantage to this method is that if you move a server or device then the zone may need to be re-configured to accommodate the move.
- WWN Zoning - In this case the zoning is done based on the adapter in the server or storage device. There is an 8-byte number called a World Wide Name (WWN) that is unique to each server adapter or device port (see “Appendix B - Fibre Channel Detail” on page79 for more information on WWNs). Collections of WWNs are put into a zone, and thus no matter where a server or storage device is moved the zone holds true. However, this method does not check every frame. When a SCSI initiator enters a fabric, after telling the fabric its own WWN it asks the fabric who else is connected to the fabric. The fabric returns addresses of devices, but only those in the same zone(s) as the host adapter. No enforcement is done after this.

  This is called soft zoning. Hard zoning will also only pass back device addresses that are in the zone, but soft zoning does nothing else. Normally this is enough as the host will not try to talk to anyone else except the devices it has heard about from the fabric. It is possible for a “rogue server” to just start trying any address and talking to it, and soft zoning would not prevent this, but this is a negligible risk and the advantage of having a zone move with a device is compelling.

The McDATA switch can also zone by switch port or by WWN (of the FC adapter or device port), but it is always soft zoning. Both the 2109 and the McDATA switch can have a mix of switch ports and WWNs in a single zone. The 2103 hub cannot zone at all. The IBM Managed Hub has limited zoning as described in “Appendix B - Fibre Channel Detail” on page79.

INRANGE boxes can have port-based zoning or WWN zoning but not both. WWN zoning is basically the same as with McDATA and Brocade. Port-based zones can be Hard zones or Name Server zones. Hard zones restrict all traffic like Brocade hard zoning, but are within a single switch. Name Server zones (specified by switch port) only affect responses from the Name Server when queried for attached devices (like McDATA port zoning. Name Server zones are always completely inside any Hard zones that exist, and Hard zones may not overlap.

All zoning however is based on some kind of port - either ports on a switch (port zoning) or ports on a server or storage device (WWN zoning). If you want two or more servers to access a single device port, but limit each server to only seeing some of the LUNs behind

that port, you must use LUN masking. This is described for various devices later in this paper.

1.1.4 SCSI over Fibre Channel

Fibre Channel Protocol (FCP) is really SCSI-3 commands and data over Fibre Channel. Now when running SCSI over SCSI buses, SCSI IDs do not flow in SCSI commands or data. The IDs of the SCSI device(s) that are currently using a real SCSI bus is determined by the distinct data wires in the SCSI bus. A SCSI Initiator preparing to use a SCSI bus, "chooses" the target (control unit) by changing the voltage on the data wire representing that target ID. (This is why the original SCSI could only have 8 total IDs - typically one server and seven control units - there were only 8 data lines on the bus. Wider SCSI buses literally had more data lines, and thus could distinguish between more IDs.) LUN#s however - which designate separate drives on a target or "behind" a Target ID - are carried in SCSI commands that flow on SCSI buses.

When running SCSI over Fibre Channel - also called Fibre Channel Protocol (FCP) - the same commands are used, and thus you still have no Target ID in the traffic. Neither do you have multiple data lines from which to choose an ID, just a pair of optical fibers. To handle this difference, Fibre Channel adapters in hosts - Host Bus Adapters (HBA) - typically "present" to the file system a Target ID for it to use for a given Fibre Channel control unit found, and then map this ID to the appropriate target Fibre Channel address. The key here is that the only "identifier" a Fiber Channel Target receives over Fibre Channel is a LUN#, it cannot distinguish between different IDs. In effect, you have a single target ID per FC port on a storage device.

NOTE: Architecturally, SCSI-3 over SCSI buses can have 64 LUNs per SCSI Target, but most storage devices only support 32, and most servers will only access 32. With SCSI over Fibre Channel, the architecture allows for over 16 quintillion LUNs per FC port, because the LUN_ID field is 8 bytes long. (A quintillion is a 1 with 18 zeroes following it.) However, all current implementations only use 2 bytes of this field, allowing 64K LUNs per FC port. Various implementations reduce this even further. The ESS supports 2-byte LUN numbers, but it doesn't use every possible LUN#. There are only 4096 LUNs total you can have in the ESS so that is its current limit per FC port. Most servers only support LUN#s 0-255. Dynix/ptx on the NUMA-Q platform and AIX support the 2-byte LUN_IDs the ESS can provide. For other platforms, the ESS ensures that the other platforms never see a LUN # larger than 255. Some older versions of operating systems support less than 256 LUNs per FC target - NT prior to Service Pack 4 only supported 8 - but it is rare today to find a platform that doesn't support at least 256 (LUN#s 0-255).

2.0 Disk Consolidation and Pooling Overview

2.1 Disk Consolidation

The term “disk consolidation” will be used in this paper to mean using a single storage subsystem to provide *separate* disk storage to more than one host. A large storage subsystem, such as the Enterprise Storage Server (ESS), can be partitioned into separate sets of logical devices such that a given host can only get to some subset of devices, and none of the devices can be reached or “seen” by more than one host. (The ESS can also be configured to allow more than one host to see devices, but that is not what is being discussed here. see “Disk Pooling” on page9 below)

If only one host server has access to any given logical volume no provision need be made for concurrent access, and everything works as if the hosts have dedicated storage devices. This allows hosts of multiple types (NT, AIX, etc.) to access the same storage *subsystem*, like an ESS, without interfering with one another’s storage (logical devices inside the ESS). Once this is set up, storage within the partitioned subsystem can be allocated and re-allocated to different hosts as needed, without having to re-cable or move any host or storage device.

Note that a Storage Area Network (SAN) is not needed to do storage consolidation like this. All that is needed is a storage subsystem with enough connectivity options to support the hosts, and the capability of being partitioned into multiple sets of logical devices with appropriate restrictions on the access. What a SAN does, is make the connectivity part of the equation much simpler to achieve by allowing physical access from any server to any device - in effect extending the consolidation available from a single subsystem, to multiple ones. SANs do not have the distance restrictions SCSI has, and there is far less cabling required to connect multiple hosts and devices together in a SAN. A Fibre Channel SAN also has the potential of improving performance - with appropriate design - since Fibre Channel ports run at 100 Megabytes per second today.

2.2 Disk Pooling

With pooling, host servers are actually accessing the same logical devices or volumes. (A subsystem such as the ESS can be configured to allow this if desired.) This always requires some software in the hosts to manage concurrent access to the volumes. Since this also requires use of the same file system, today it means the servers accessing the shared volume(s) are using the same operating system.

NOTE: It is possible in principle to have the same file system on different operating systems - such as the Andrew File System on different UNIX platforms - thus allowing heterogeneous sharing of volumes. This would still require appropriate clustering mechanisms to manage access to the volumes. This sort of thing is not generally seen today, although Tivoli SANergy from IBM allows some file-sharing across platforms to be done over the SAN. SANergy will be discussed in a later version of this paper.

In order to share access to the same volumes, the various UNIX operating systems typically allow a given host to "know" about volumes without accessing them. This allows multiple hosts to "share" the volumes without contention, since really only one host is using the volume at any given time. AIX with HACMP enhances this shared access, by allowing a host to "pick up" use of volumes previously being used by a different host, once it is detected that the host has failed. True simultaneous use of volumes is possible with the with additional software to allow concurrent access. Care should be taken to understand the abilities and restrictions of such software. Such considerations are beyond the scope of this paper.

With Windows NT or 2000, the only shared access is using Microsoft Cluster Services (MSCS). This allows volume sharing between two NT/2000 systems, much like HACMP for AIX. There is however, no simultaneous access to volumes with MSCS. Novell Netware also has clustering services for its servers, similar to MSCS. zOS and OS/390 have facilities inherent in the operating system and I/O Subsystem to allow device sharing. Dynix/ptx, the Unix-based OS that runs on NUMA-Q systems, also has clustering capability that can be added.

In this paper, we will only concern ourselves with whether more than one host can "reach" a given logical device. If more than one host can access a given logical device, then those hosts must be using the same file system, and they must have some sort of ability for shared access, and be appropriately configured. This will be generically referred to in this document as "**shared access capability**", regardless of the level of sharing and failover that is available with the capability.

Some sort of "shared access capability" is always needed when doing disk pooling. It is not needed if you are doing only disk consolidation with partitioning as described above. Note that it is quite possible to do both, that is, you may have multiple clusters of like systems, each using shared access to their own cluster's volumes, but not accessing the other clusters' volumes which are partitioned off in the storage device. (We will show examples of this.)

2.3 Multi-path Host Software

In various high availability configurations, you will want a single host to be able to access the same logical devices through multiple host adapters. (This paper will refer to this as single-host-multi-access) This requires some software in the host to manage this. zOS and zSeries I/O Subsystems can do this already, as can the Dynix/ptx operating system. For Open Systems, IBM provides the Redundant Disk Array Controller (RDAC) driver for the FAStT200, and the IBM Subsystem Device Driver (SDD) - the follow-on to Data Path Optimizer (DPO) - for the Virtual Storage Server (VSS) or the ESS. (There are other examples of single-host-multi-adapter, such as the way AIX supports two SSA adapters, but we will only be discussing single-host-multi-access through Fibre Channel adapters in this paper.)

In any case, *something extra* is needed to allow a single host to access the same logical devices/volumes through more than one Fibre Channel host adapter. (This is a single-host

consideration, and is independent of whether disk pooling is being used or not. Multi-adapter access by a single host can happen with or without disk pooling.)

2.4 Summary of generic disk connectivity

As you read through these sample configurations keep the following in mind:

- **Different file systems must not access the same logical device.** If multiple different host types can access the same storage subsystem, that storage subsystem must be partitioned to avoid any overlapping access between hosts.
- **Even hosts with the same operating system cannot access the same logical devices if they do not have some sort of shared access capability installed and configured to manage the multi-access.** Windows without additional Clustering Services software for instance, would need the same partitioning as different file systems.
- **Hosts with "shared access capability" installed and configured, can take advantage of disk pooling** (shared access to the same logical devices)
- **If a *single* host can access the same devices from two different adapters, there must be some additional host software to manage the multiple access for the single host.**
- **Partitioning, disk pooling, and single-host-multi-access can occur with the same storage subsystem in any combination.**

Now let's look at some examples...

3.0 Disk Configurations with the FAStT200

3.1 FAStT200 Basics

The FasT200 (machine number 3542) is a disk subsystem comprised of one or two RAID controllers that can be attached to IBMN EXP500 drawers of disks. The single controller model only supports up to three drawers (30 disks), whereas the dual controller model can support up to six drawers (60 disks). There is also a FASTt500 and a FASTt700. These are similar in design to the FASTt200 with greater capacity, speed, and number of ports, and some additional function that is disk-related and not SAN related. All the issues and principles presented here for the FASTt200 also apply to the other models.

With the single controller model, the controller accesses one drawer internally (on a Fibre Channel loop), and also has two external Fibre Channel ports. One port allows an attachment to a host, and the other is for loop attachment to additional disk drawers.The host port can be connected either directly to a host server, to a SAN via a 2109 (could be more than one switch in the SAN), or to a fibre channel managed hub (3534).

NOTE: Platform and SAN device support for the FAStT200 can be found at:

http://www.storage.ibm.com/hardsoft/products/fast200/supserver.htm

If a second RAID controller is added to the FAStT200 (or the 2-controller model is ordered), both controllers internally access the first disk drawer, both controllers have their own Fibre Channel port for host attachment, and both controllers have another Fibre Channel port for expansion to additional disk drawers.

NOTE: To be supported, you must connect both controllers to all disk drawers. Each controller can access the drives through its own "drive-side" port or through the port on the other controller, thus there are alternate paths with one controller, as well as redundant controllers.

These drawers - the EXP500 - have four Fibre Channel ports each (2 labeled "In" and 2 labeled "Out") for attachment to the RAID controllers. In essence, there are two loops in one of these drawers, and each disk is "on" both loops. Attaching a controller to one of the ports on a disk drawer puts the controller on one of the drawer's loops. If there is another expansion drawer, then the Out port of one drawer can be connected to the In port of the next one to put them all on the same loop. With two RAID controllers, you can have each controller on different loops of the disk drawer(s) giving redundancy and failover capability in the "back end" of the disk subsystem.

When you configure a FAStT200 you put disk drives into RAID arrays, and then define logical drives within an array. There can be up to 128 logical drives in one FASTt200. These logical drives will appear to the hosts as physical drives (SCSI LUNs.) The FAStT200 can be partitioned into 16 different storage partitions. Logical drives are put into storage partitions, as are hosts or host HBAs. Only the host HBAs that are allowed to a given partition can see the logical drives in that partition. (In effect you can control "sharing" or disk pooling with partitioning, either allowing it or preventing it on a LUN by LUN basis as needed.)

There is a default partition - separate from the 16 - that can contain hosts and LUNs as well, but it is not recommended for production use because unassigned HBAs are automatically put into the default partition. If there are LUNs in the default partition, they are immediately available to any new host that can reach the FAStT200, which could easily be a disaster for production data. Typically, the default partition is used merely as a placeholder for HBAs, not assigned, or for test LUNs where there is no concern for data integrity.

When using the FAStT200 with 2 RAID controllers, each logical drive is associated with a single RAID controller at a time. However, either controller can reach any logical drive, so in the event of a controller failure, the other controller can start managing all of the logical drives. We will see in later examples how other kinds of failures involving Fibre Channel connectivity are handled.

NOTE: There are other failures that can occur inside a FAStT200, such as drive failures handled by a single RAID controller, but these are transparent to the SAN and the hosts, and will not be discussed in this paper.

During normal operations (before a failure) any host connected to (having access to) both FC ports can see all logical drives (in all partitions to which it has been assigned) through both HBAs. That is, we have a single-host multi-access situation, and appropriate host software is required (i.e. the RDAC driver). These kinds of configurations are discussed starting in Figure5 on page17.

3.2 SCSI numbering in the FAStT200

As stated in Section1.1.4 on page8 a fibre channel port "appears" as a single SCSI target ID to the FC hosts. More importantly, different logical drives reachable behind a single destination Fibre Channel port must use unique LUN#s. Each partition in a FAStT200 can support up to 32 LUNs.

In the FAStT200, if you do not use partitioning, all hosts, HBAs, and logical drives are in the "default partition" and there is only one pool of LUN numbers for the whole FAStT200. Each logical drive would have a different LUN# associated with it, regardless of how it is accessed. (This is referred to as the default logical drive-to-LUN mapping, which is what is used in the default partition.)

If you wish to partition storage, explicitly defined mappings will be used, and it is quite possible for the same LUN# to be used for different logical drives in different partitions. The user must insure that any single host only sees unique LUN#s behind a given FC port. Hosts that are in more than one partition must be handled carefully! (Usually a host need only be in one partition. Reasons for a host to be in more than one partition include the requirement for more than 32 LUNs, or a need to share some LUNs with another host while also having un-shared LUNs.)

Also, care must be taken not to exceed restrictions of operating systems on LUN#s allowed. For example, with NT prior to Service Pack 4 you could only use LUN#s 0-7. With Service Pack 4 if you turn on the "Large LUNs" option, LUN#s up to 255 are allowed over Fibre Channel. (Keep in mind that the highest LUN# allowed by an operating system may not be the same for Fibre Channel adapters as it is for SCSI adapters. Most current Fibre Channel implementations can use LUN numbers up to 255, whereas 31 is typically the highest number for real SCSI.)

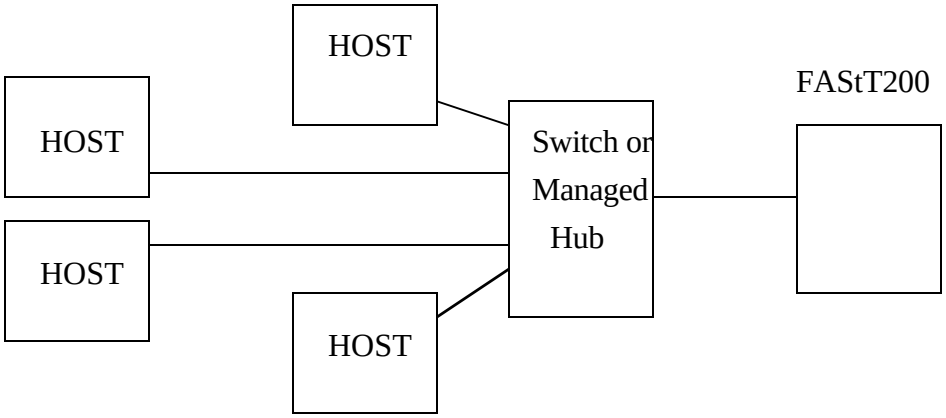
Following are several sample configurations using the FAStT200. Typically, SAN components are not shown unless there is specific reason to do so. In general, a direct connection will behave the same as one with a switch or hub in between. Where this is not true it will be pointed out.

3.3 Single-controller FAStT200 configurations

FIGURE 1. One host, one controller (FAStT200)

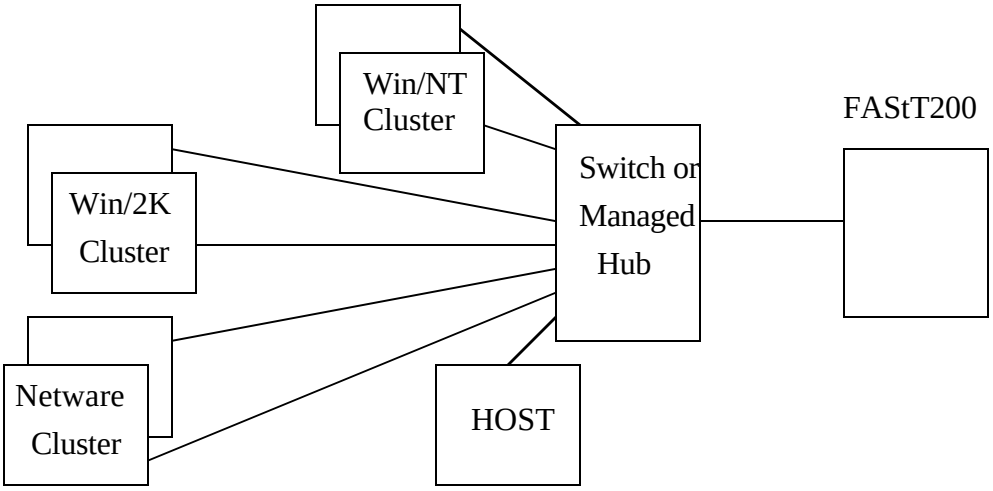
This is a trivial example of a host connected to a single controller FAStT200. There is no partitioning needed as there is only one host.

FIGURE 2. Partitioning a single controller FAStT200



The above figure shows four hosts accessing a single FC port on a FAStT200. As long as the FAStT200 is partitioned, such that each logical drive is only mapped to one of the hosts, the hosts can be any mix of supported operating systems, with none of the hosts having any clustering software added on. In effect, the FAStT200 in the above figure has four non-overlapping sets of logical drives that each host accesses. While one of the hosts could be using default mapping in the default partition, this is not recommended as stated before, since any other host that is plugged into the switch/hub will immediately see all LUNs in the default partition. At least three of the hosts (and preferably all of them) are using specific logical drive-to-LUN mapping in a non-default partition. If there were eight logical drives defined - 2 per host - each host could be using, say, LUNs 0 and 1 to access their 2 drives.

FIGURE 3. Clustering and Partitioning on a single-controller FAStT200



In the figure above, there are still four partitions, but now three of them have two hosts. The fourth partition could be any of the supported operating systems, and could also have two hosts. (If all partitions had two hosts you would need 9 FC ports - 8 for the hosts and one for the FAStT200 - and that would force you to a 16-port switch as the Managed hub only has 8 ports.)

Both hosts in a given cluster are in a host group assigned to a partition for which logical drives are also assigned. The clustering software that controls shared access between hosts in a cluster is required since two hosts in the same partition will see the same logical drives. Once again, using drive-to-LUN mapping, you could have 2 logical drives in each partition, and all 7 hosts could see them as LUN#s 0 and 1. You could map different numbers, but some operating systems (e.g. Windows NT), after checking for LUN 0, quit looking for LUNs if LUN 0 is not present. Typically LUN#s need not be consecutive, but often LUN 0 must exist.

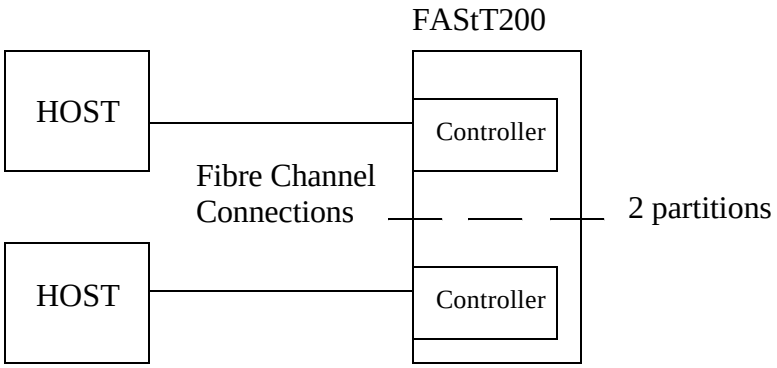
NOTE: In the previous two figures, if a Managed Hub were used, booting one of the servers would drive LIP on all hub ports once the server came back up. (See “Hubs and Loop Initialization” on page5.) Thus it is recommended to use a switch for configurations with more than a couple of servers.

3.4 Configurations for FAStT200 with two controllers

When a FAStT200 has two RAID controllers, each controller has a Fibre Channel port for host access. However, just because you are using two RAID controllers does not necessarily mean that you have a high availability configuration. Let’s look at one very simple configuration in which two hosts are directly attached to the two FC ports on a two-controller FAStT200.

3.4.1 “Non-failover” configuration with 2-controller FAStT200

FIGURE 4. Non-failover FAStT200 configuration



In the configuration shown above, each host is accessing separate logical drives in two separate partitions. These logical volumes can only be reached by one host. For instance, suppose the host at the top can get to three logical drives (i.e. this is what is configured on the FAStT200.) When it boots up, it will query for LUNs. Even if one or more of these LUNs were erroneously assigned to the controller on the bottom - which the host on the top cannot reach - the FAStT200 will respond correctly to the upper host with all three LUNs. The first time the host actually does something with a logical drive on the bottom controller, the FAStT200 will “move” that logical drive to the top controller and the I/O will go on. (There will be some error messages generated, but everything will work out.) At some point, all of the top host’s logical drives will be handled by the top controller, and vice versa, allowing each host to access all of its drives on the one controller it can reach.

In the configuration above, the hosts could be any mix of supported platforms at the appropriate software levels. The appropriate URL should be checked for platform and code-level support.

In the above figure (Figure4 on page16), there is no failover available, not even for controllers. Each host can only reach one controller, and should that controller fail then the host cannot access its logical drives. If something happened to the FC path between a host and the FAStT200, again, that host would lose access to its drives. (These two hosts could be different operating systems, but there is no recovery from an HBA failure, a cable failure, an FC port failure or a controller failure.)

3.4.2 Failover configurations with FAStT200

Something different happens when a single host is connected to both ports of the FAStT200. In order to sort through all the combinations we will look at some very simple configurations that may never be useful to customers, but they will be illustrative of what

is going on with a FAStT200, and thus useful as learning aids. Consider the following example of one host with two HBAs, each connected to a different FC port on a FAStT200.

FIGURE 5. High availability - single host (FAStT200)

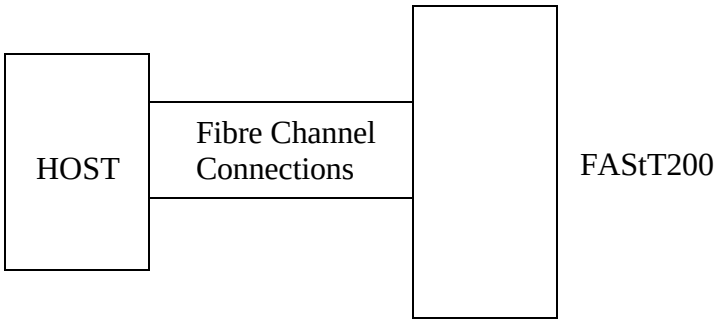


Figure 5 on page17 shows a host that can reach both controllers in a FAStT200. Assume that there is no partitioning. When the host Queries for LUNs, each HBA will “find” all the LUNs in the FAStT200. If nothing else is done, as each HBA attempts to get to the various logical drives, they will be moved back and forth between controllers depending on where the last request came from. Furthermore, (if nothing else is done), the host will not realize that those are the same logical drives out each HBA. The host will see a set of LUNs out one HBA, and another set out the other HBA, and never realize that they are in fact the same logical drive. (Different operating systems will react in different ways to this situation.)

This an example of single-host-multi-access as described in Section2.3, “Multi-path Host Software,” on page10. *There must be software in the host to handle this.* For the FAStT200, this software is provided with FAStT200. There is software available for many platforms as listed in the supported server URL. In essence, this is shim code showing one drive to the host’s file system for any given logical drive, while actually working with two different adapters/drivers (with the logical drive visible out each.) This special software for the FAStT200 is called the Redundant Disk Array Controller (RDAC) driver. Without this software, the operating system would see two drives for every logical drive in the FAStT200. It would not recognize them as the same drive, causing more than a little trouble.

Back to Figure5 on page17, where a single host has access to all logical volumes through both FC adapters. With RDAC installed on the host, when the host comes up, the RDAC driver will access some of the logical volumes through one adapter, and some will be reached through the other adapter (Even though RDAC finds the LUNs out each adapter, it will check to see which controller owns them, and initially will access a given LUN only through the HBA to the controller that owns that LUN. Note that only one path to a LUN is actually in use at any given time, but by assigning different LUNs to different controllers some “manual load balancing” is achieved.)

Failures that do not result in a loss of LUN access - such as a disk failure in a RAID 5 array or something else easily “handled” by the back end of the box - are transparent to the

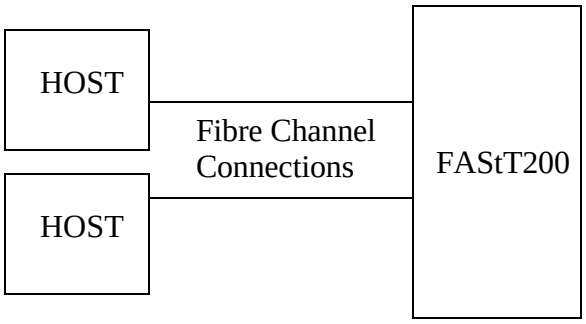
host and RDAC will not detect or handle these. But, when a failure in the current path to a LUN occurs - whether it is a failed RAID controller, or an HBA, or a link, or a host port - the net result is that LUNs can only be reached via one controller. (This is more obvious with a controller failure. since only one controller is left running. With a failure in a Fibre Channel path, then one controller is no longer reachable by the host even though that controller my still be operational.)

For all of these failures, from RDAC's view, there is only one HBA left that is getting valid responses, and RDAC will use that HBA which still can reach a functioning controller for all logical drives. This will move all the logical drives over to the remaining reachable/functional controller, and everything will continue to run. Everything is still the same from the host's point of view - all LUNs are still available.

NOTE: RDAC is *not* the same as the IBM Subsystem Device Driver (SDD) which is the follow-on to Data Path Optimizer (DPO). RDAC only supports the FAStT products and SDD only supports the VSS (over SCSI) and the ESS (over SCSI or Fibre Channel). DPO from Compaq is for the MSS (not covered in this paper), and is different than any of the others, although all provide a similar, (but not the same) multi-path function.

A different kind of high-availability configuration is as follows:

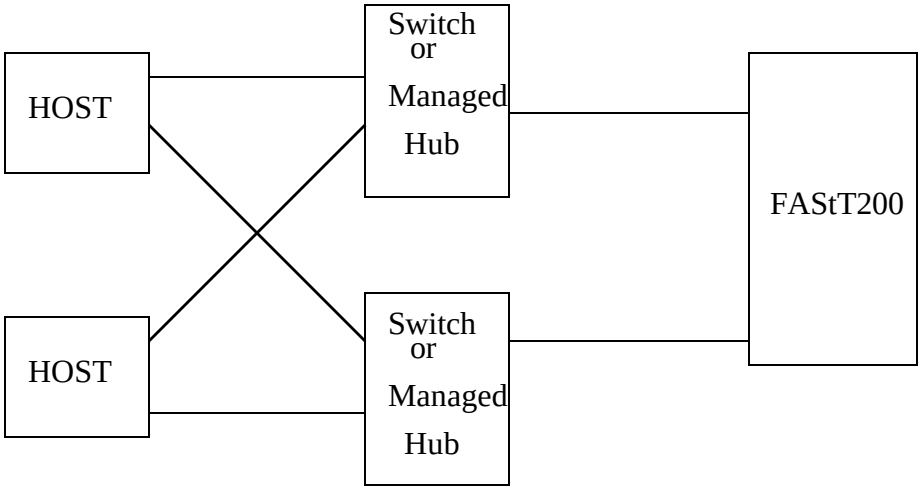
FIGURE 6. Simple failover cluster (FAStT200)



In this case, both hosts can see all of the logical drives and thus require some sort of "shared access capability." Suppose they are two NT boxes using MS Clustering. (All clustering software makes use of LAN connections between hosts not shown in the picture.) In this case you could lose an entire host, and the other host could "pick up" the workload by accessing all the logical drives through the other FC port. *RDAC is not required in the above configuration as any single host is only using one path. However, if any logical drive is accessed by both hosts (concurrent access) it will constantly shift between controllers as the requests come in on both host ports. This is not good unless the concurrently accessed logical drives have very low traffic volumes.*

Let us now combine these two approaches as in the next picture:

FIGURE 7. High availability with shared access (FAStT200)



The configuration in Figure7 on page19 above makes use of both disk pooling (shared access) *and* single-host-multi-access. Assuming no partitioning, each host sees all the logical drives in the FAStT200, and thus "shared access capability" (some kind of clustering) is required. Each host can also reach all the volumes through both of its adapters, thus each host also needs the appropriate RDAC driver.

This configuration can tolerate the failure of an HBA, a link, a switch, a host port, a RAID controller, or even a host, and there would still be at least one host in the cluster accessing every logical drive. Since you can partition the FAStT200 into as many as 16 partitions, you could have 16 different clusters, with all 32 hosts having 2 HBAs - one HBA reaching one FASTt200 port.(This would require two multi-switch fabrics rather than two single-switch fabrics shown here.) All 32 hosts would have the same failover capabilities. You would need to ensure that logical drives and the hosts accessing them were only assigned to one partition.

NOTE: Since each host has RDAC running, which will find out where a logical drive is being handled, initially there could be a logical drive accessed concurrently by two hosts in a cluster without thrashing occurring between controllers. (RDAC in each host would only use the path to the controller initially "owning" the logical drive.) However, in the event of a Fibre path failure, you could easily end up in a thrashing condition as one host started using a different path (controller) for access, and the host without the failure continued to use the original path (controller). Still best to avoid or minimize concurrent access.

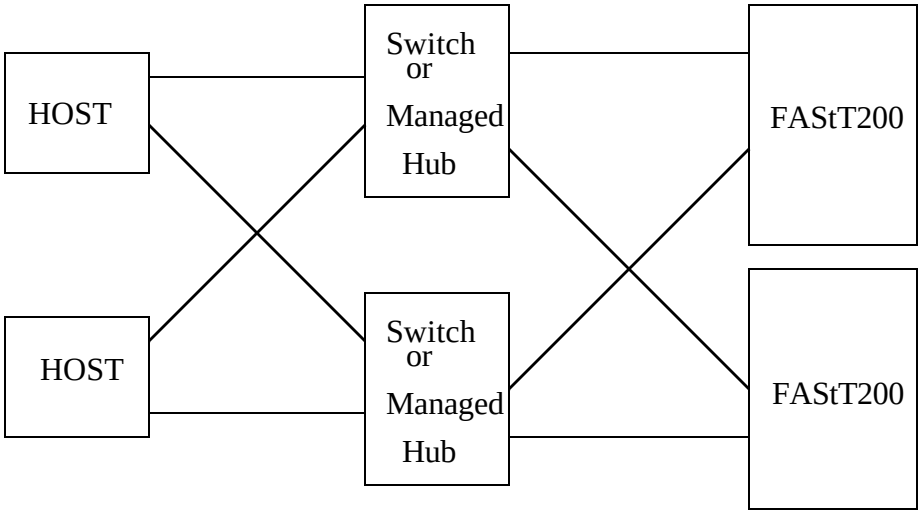
Back to the original 2-host picture at Figure7, "High availability with shared access (FAStT200)," on page19. This configuration could also be done *in principle* with a single managed hub or switch (and some zoning). This would, however, give you a single point of failure - the single hub/switch - thus defeating the purpose of a high-availability configuration. The considerations about multiple servers attached to one hub as discussed in

Section1.1.2, “Hubs and Loop Initialization,” on page5 still apply here so for more than one cluster it is recommended that you use switches rather than hubs.

3.4.3 Configurations with 2 FAStT200s

The configuration in Figure7 on page19 can be expanded to two FAStT200s as shown below.

FIGURE 8. High availability with 2 FAStT200s



At first glance, this configuration merely doubles the amount of storage available in the previous configuration, with all other considerations the same. However it is possible for the two hosts to mirror the storage in one FAStT200 onto the other one. (This would have to be host-based mirroring - duplicate I/O done by the host itself - as neither the SAN infrastructure nor the FAStT200 will do this between two Fast200s.) This mirroring would now allow an entire FAStT200 to fail, and the hosts could then access the data on the second FAStT200 while the first one was getting fixed. Considering all the recovery options (including various RAID levels) already inherent in the FAStT200, this is a lot of extra cost for only a little extra protection. However, if we put the second FAStT200 in a different site we get a different perspective.

Figure 9 on page21 shows the second FAStT200 in a different location using longer distance fibre channel connections. Each location has a configuration like Figure 7 on page19 and the two connected sites can now do remote-mirroring or remote vaulting. Consider the picture below:

FIGURE 9. Remote site FAStT200

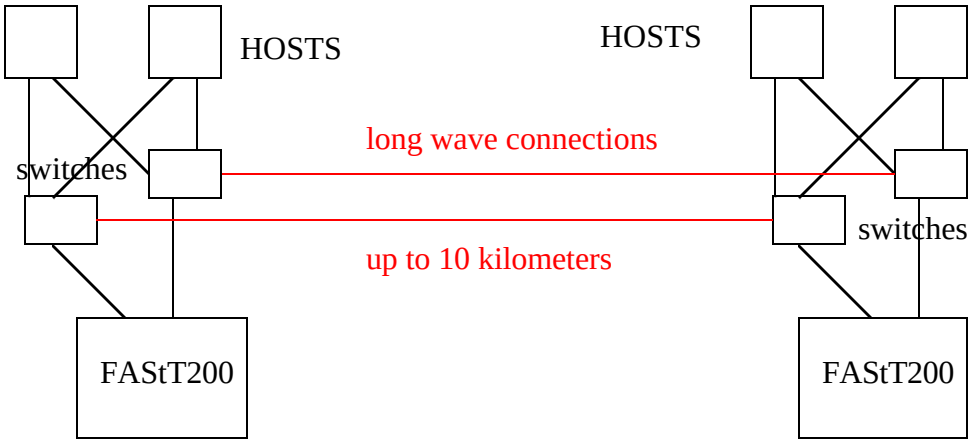


Figure 9 on page21 shows two sites in a single SAN using longwave singlemode fiber connections between switches (these can go up to 10 kilometers). There is no logical difference between these long range connections and any other switch-to-switch connections. They still run at 100 Megabytes/second and operationally everything is just as if the whole SAN is in one room. (The differences are only physical differences in the fiber itself and the lasers driving the fiber. However, there is additional latency in all I/O operations due to the 10 km distance. There will be some performance hit here, for the host doing the mirroring. Data that is not critical could be left un-mirrored.)

This configuration could allow host-based mirroring to a remote site. In the above configuration all four hosts have access to all logical devices through two different adapters so the RDAC driver must be installed on all of the hosts. However, many clusters can only have 2 servers in a cluster, so two of the servers might need to be kept “off-line” (e.g. powered off) at any given time.

Suppose the site on the left were the primary site and the site on the right is a backup site. Normally only the two hosts on the left would be in use, but they have access to both FAStT200s and thus can mirror their data onto the remote FAStT200. The primary site is a high availability configuration, and could tolerate failures of any single SAN component, or a failure of one controller of the FAStT200, and still continue in failover mode. However, if the entire site went down, disaster recovery is possible by using the two hosts in the remote site to access the mirrored data in the remote FAStT200 (There is more work required for *users* to get access to the host servers at the remote site, but that is a network and clustering issue, not a SAN consideration.)

3.5 Other SAN considerations/restrictions with FAStT200

Figure 9 on page21 is the first configuration in which we show more than one hub or switch between a host Fibre Channel adapter and a Fibre Channel port on a FAStT200. In

general wherever we show a switch, you can have multiple switches up to the restrictions of the switch itself. IBM currently supports up to 32 2109 switches in a single SAN, with no more than 8 switches (7 hops) between FC users. With hubs you can have no more than two daisy chained together.

Longer distances are possible by daisy-chaining multiple switches, or using extension devices such as the IBM Fiber Saver, or longer distance GBICs. Currently, a maximum of 70 kilometers total is supported between host adapter and storage device port.

Finally, an IBM Managed Hub can be connected to a switch, but servers attached to an IBM Managed Hub will not find storage devices attached to a switch. A *switch*-attached server can find storage devices on the IBM Managed Hub. (See Section 11.2.1, “The Netfinity Managed Hub,” on page82.)

4.0 Open Systems Disk Configurations with The Enterprise Storage Server (ESS)

The Enterprise Storage Server can be used by both open systems and S/390 or zSeries processors. This portion of the paper only deals with open systems connections.

4.1 ESS Basics

The ESS is a large disk subsystem that can have many different host attachments. 16 adapters can be installed in an ESS in any mix. Current adapters are a 2-port SCSI adapter, a 2-port ESCON adapter, and various 1-port Fibre Channel adapters. (Some Fibre Channel adapters can be used for only FCP - SCSI over Fibre Channel -some Fibre Channel adapters can be used for both FICON and FCP. These “dual-purpose” adapters must be configured to do one or the other - FICON or FCP - and can not be using both protocols simultaneously.)

NOTE: There is another adapter called the “Interim Host Adapter” which is really a SAN Data Gateway (SDG) in combination with a SCSI adapter. This is explained later in the paper.

The ESS uses 8 loops of SSA disk on the back end. While there is an option to use these disks as LUNs themselves - referred to as JBOD or Just a Bunch Of Disks - it is more typical to create LUNs using RAID 5 arrays of 8 disks. (The first two of the arrays on each SSA loop will have an empty disk to be used as a floating spare on the loop, others will use all 8 disks in the array for data and parity.) It is possible to configure Count-Key-Data (CKD) devices on these arrays as well, and these would be used with mainframes through any ESCON or FICON adapters. Details of ESS configuration can be found in ESS manuals and Redbooks dealing with this subject in detail.

4.1.1 LUN usage in the ESS

LUN access and usage in the ESS is different than the FAStT200. There is no concept of partitions in the ESS, rather each LUN is explicitly associated with server HBAs via the HBAs' World-Wide-Names. Thus, LUN masking is achieved by specifying WWNs of HBAs that have access, and disallowing all others.

LUNs that are created on an ESS are presented at either SCSI ports or FC ports. For SCSI access, LUNs can be given affinity to (allowed access from) ESS SCSI ports. Any host coming in on a given SCSI port, can reach the LUNs that have been associated with that port.

With FC however, a LUN is associated with server HBAs. This is done by assigning to a LUN, the WWN of the HBA(s) you want to have access to that LUN. LUNs can be accessed from any FC port, but only by the listed HBAs. (LUNs that have no HBA's WWNs associated with them are treated according to a user-specified default. Either no server can reach a LUN that has no WWNs assigned, or any server can. For security it is best to set the default such that a LUN is not reachable unless specifically allowed via WWN. In test labs or similar non-production environments it may be convenient to allow open access.)

Note that in older levels of microcode, there was no way to restrict access to a LUN to specific FC ports. Once an HBA has access to a LUN it can see that LUN through any FC port that it can reach. If a single HBA can get to two ports of an ESS, it will always see its LUNs through both ports, and thus will require the multi-path support provided by SDD. Almost all ESS configurations end up requiring SDD. With later microcode levels it is possible to assign HBAs to only specific ports, such that the ESS will only talk to an HBA through a specific set of ports.

Finally, the LUN numbers that are presented on the fibre channel ports is different for different HBAs. Most servers only support LUN numbers from 0-255, and this is what they will see, although a single LUN in the ESS might use different LUN numbers with different HBAs. For AIX, and Dynix/ptx (NUMA-Q) a two-byte LUN number is used. For these servers, the ESS volume ID for the LUN is used as the LUN#, and this number is between X'5000' and X'5FFF'.

4.1.2 ESS Failover Architecture

The ESS hardware uses two RS/6K engines (each with multiple processors) to provide advanced functions such as copy services. These controllers have SSA adapters in them that provide the RAID5 function on the back end. Each SSA loop is accessed from each controller, but any given RAID rank is managed at any given time by a single controller.

The host adapters on the front end have connectivity to both RS/6K controllers and can thus reach all LUNs at any time. This is unlike the FAStT200, in which host ports can only really get to a single controller. This means that all possible paths to a LUN (all FC ports a host has access to) can be in use simultaneously, and failover really means just using what-

ever paths are left after a failure. There is never a “shifting” of LUN control from one controller to another, unless a controller fails, because no matter which FC port a request comes in on, it can be satisfied by that FC port merely going to the correct controller for that LUN.

Because for a given LUN multiple paths can be in use simultaneously, very sophisticated load balancing schemes can be used by SDD (involving use of recent I/O statistics to dynamically choose a path). The various failures one can experience merely reduce the number of paths available to the load balancing mechanisms. (Hopefully the number of paths is not reduced to less than one!) Let’s take a look at some sample configurations.

4.2 ESS Configurations - Native FC attach

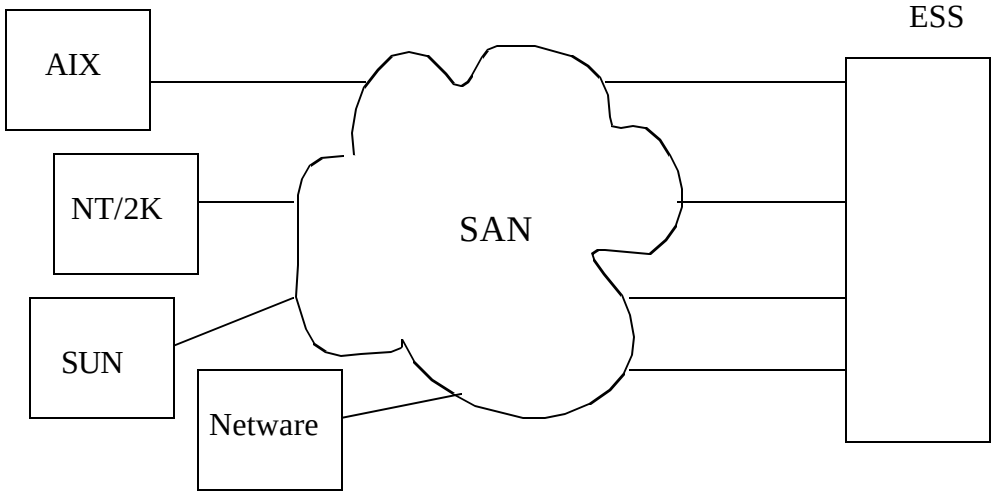
The ESS has great flexibility in how hosts can get to LUNs, making connectivity and SAN design much simpler. Currently the ESS supports up to 128 HBAs per FC port, and up to 512 HBAs per ESS. Due to throughput maximums on FC ports and the ESS itself, this is many more HBAs then anyone would ever actually attach. Supported hosts and required code levels can be found at:

http://www.storage.ibm.com/hardsoft/products/ess/supserver

4.2.1 Basic ESS configurations with Native FC attach

Lets look at a few possible configurations:

FIGURE 10. Basic multi-host ESS configuration (Native FC attach)



The figure above shows 4 different hosts each with one HBA attached to a SAN. The ESS has four FC attachments to this SAN as well. Because these are all different hosts, when defining LUNs in the ESS, you would need to ensure that only the HBA for the host using that LUN is allowed to that LUN. This would prevent one host from using another host’s LUNs.

Also, in Figure10 on page24, if there is no zoning done in the SAN, and there is no restriction configured in the ESS as to which HBAs can get to which ports, then all HBAs would be able to access all four ESS ports. This means that each host would have four different paths to the same LUNs. In effect, each HBA would "see" the same LUN behind multiple SCSI targets. (Each FC port on the ESS behaves as a SCSI target.)

NOTE: Alternate paths within the SAN are not relevant to multi-pathing issues from a SCSI standpoint. It is a unique pair of endpoints - HBA and device port - that comprise a path. Every unique pair of SCSI Initiator Role (HBA) and a SCSI Target Role (device port) that can reach the same LUN represent a different path to that LUN.

Assume in Figure10 on page24, that each host had two LUNs in the ESS assigned to its HBA. Each HBA will see 8 LUNs - 2 LUNs behind each ESS port. If nothing else is done the Operating System will think there are 8 different LUNs it can access even though it is really just 4 copies of the two that it was allowed to access. SDD host software installed on the host would sort this out. (SDD will detect the duplicates, and present only unique LUNs - in our case two of them - to the operating system. Also, unlike RDAC which only uses one path at a time, SDD will use all paths concurrently with dynamic load balancing. SDD can manage up to 32 different paths per LUN.)

It is important to check the website to ensure SDD support. As of 1/1/2002, SDD was not available for Netware. Thus in Figure10 on page24, either the SAN would need to be zoned to restrict the Netware HBA to a single ESS port, or the HBA could be restricted through ESS configuration. The other platforms would need to either use SDD, or be similarly restricted (far preferable to use SDD.)

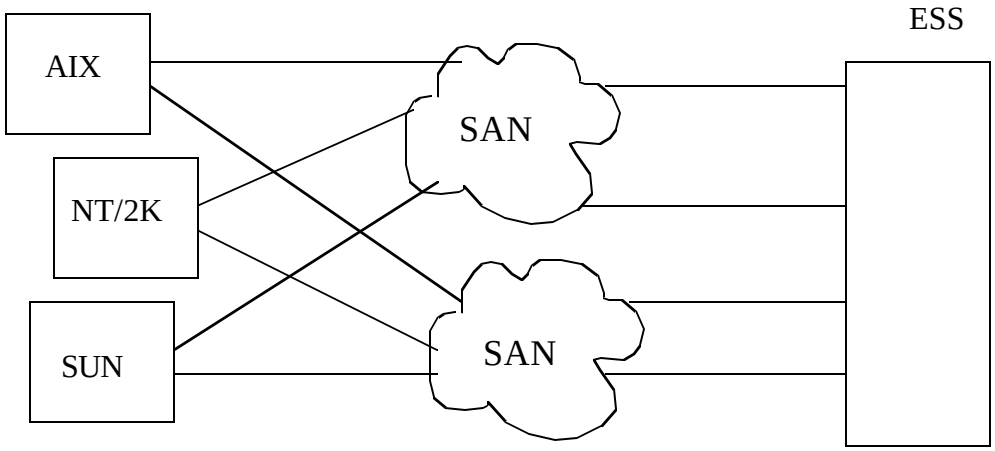
To summarize the issues for the configuration shown in Figure 10 on page24.

1. LUNs in the ESS must be restricted to only the one host. This is done by associating each LUN with only the proper HBA's WWN.
2. 1 HBA x 4 ESS ports = 4 paths. Without further configuration each host will see 4 copies of each LUN and treat them as separate LUNs. This can be resolved for a given host in any one of the following ways:
   - Install SDD in the host to manage the paths (not available for Netware as of 1/1/2002)
   - Zone the SAN such that each HBA only reaches one ESS port
   - Configure the ESS such that the HBA WWN can only use a single port
3. Hosts not using SDD are restricted to a single ESS port, and would not recover from a failure of that port or a failure in the SAN that lost access to that port. Hosts not using SDD would recover from other ESS failures, since a single ESS port can access any LUN from any cluster. Host using SDD and multiple paths (multiple ports on the ESS) could also recover from any ESS port failure or any SAN failure as long as there was still a path available from that host's HBA to any ESS port.
4. HBAs are a single point of failure; there is no recovery from an HBA failure in this configuration. Loss of connectivity from the HBA to the SAN is also an unrecoverable failure.

To eliminate these single points of failure we must add HBAs to the servers.

4.2.2 High Availability Configuration with Native FC-attached ESS

FIGURE 11. High Availability ESS Configurations (Native FC attach) - Servers not clustered

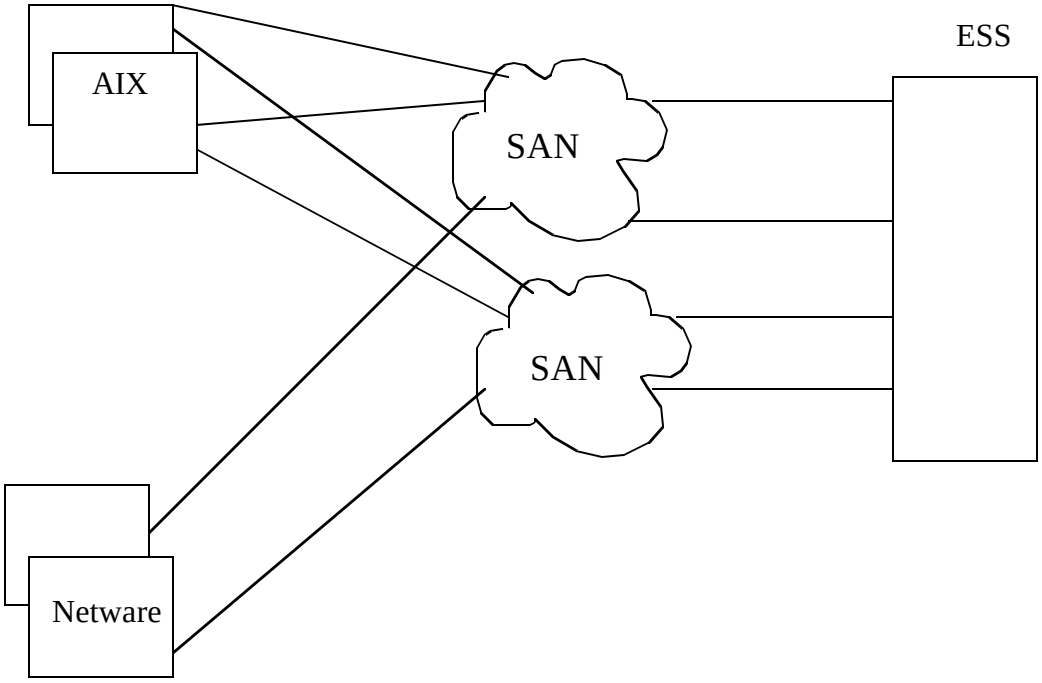


In the figure above, the servers all have high availability configurations. Netware is excluded this time because this configuration requires SDD, and there is no SDD for Netware. Netware could have 2 HBAs as long as the ESS prevented any LUN from being reachable by both HBAs, but then there is no recovery from a lost HBA. In the configuration above, if both HBAs in a server can get to all the server's LUNs (requiring SDD), then an HBA or path failure can occur with no loss of access to LUNs.

The figure shows two separate SANs, common in high availability configurations. It is possible to run this configuration with one large SAN with sufficient alternate pathing, but there is little to gain from doing this. For instance, you could connect the two SANs in the figure together to make one large SAN, but all this would do is give each server 8 paths to each LUN instead the 4 in the figure. (In the figure, each HBA gets to two ESS ports, for two paths each or a total of 4 paths. If the two SANs were connected, each HBA would get to 4 ESS ports yielding 8 paths.) The extra paths do not add much to the configuration.

The only single point of failure in Figure11 on page26 is the server itself. This can be handled with server clusters as show in Figure12 on page27.

FIGURE 12. High Availability ESS Configurations (Native FC attach) - Clustered Servers



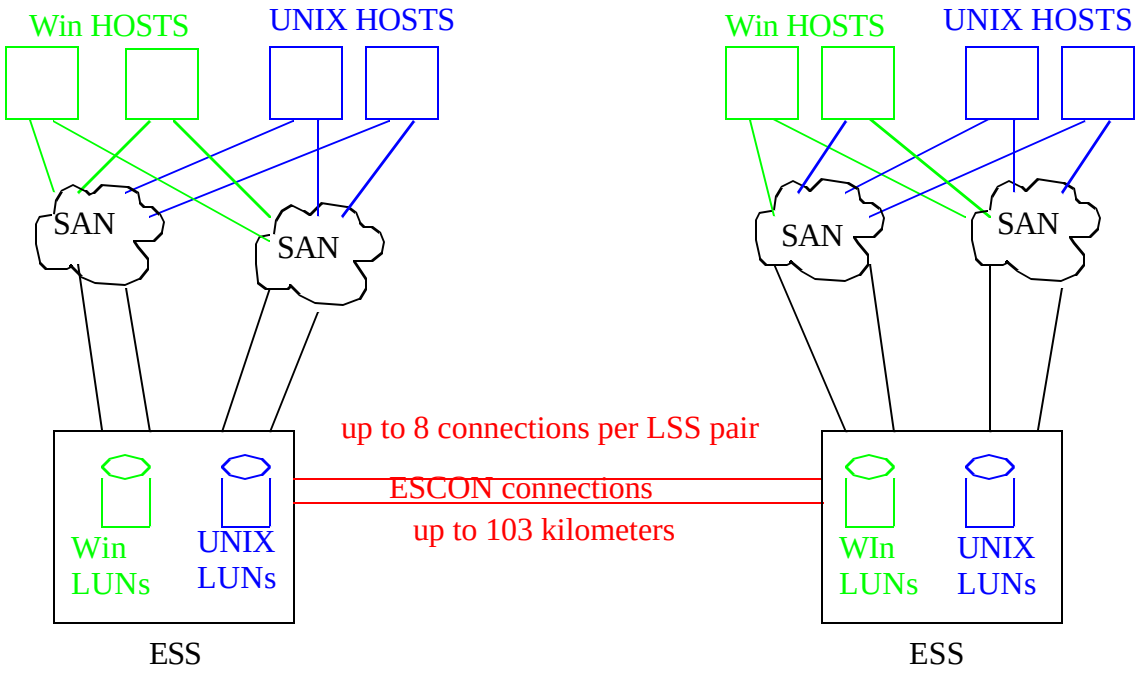
To keep the picture simple, I have just shown an AIX cluster and a Netware Cluster. The AIX cluster has two HBAs per server, and each server is using SDD for multi-pathing to LUNs, as well as clustering software to allow sharing between servers. With the Netware cluster, SDD is unavailable, so each server must be restricted to a single ESS port either by zoning in the SANs or by configuration in the ESS. Netware's cluster services are used to allow LUN sharing between the two Netware servers.

The single HBA and single port in the ESS (for Netware) are single points of failure for the Netware cluster, but should one of these fail, cluster failover mechanisms could be used to move the work to the other Netware server. The AIX servers have no single point of failure, *and*, should one of the servers fail, it can failover to the other server. Windows and SUN can also have clusters with 2 HBAs per server. For full details on which platforms support SDD, and which platforms can use clustering software with the ESS, go to the website for the ESS.

Even though a configuration using host-based mirroring such as that shown in Figure9, "Remote site FAStT200," on page21, is possible with the ESS, there is a better mechanism for ESS-ESS mirroring called Peer-to-Peer Remote Copy (PPRC). PPRC provides mirroring function that does not require the host to be involved. It is a synchronous copy - meaning that the host does not receive the confirmation of a write I/O being complete until both ESSs have the data. Indeed the host at the primary site will not even have access to the mirrored copy.

Consider Figure13 on page28.

FIGURE 13. Remote mirroring with ESS PPRC (Native FC attach)



In the figure above, I have used the term “Win” to refer to NT or Win2K, and “UNIX” to refer to the various flavors of UNIX. This does not imply full sharing of LUNs between any servers - LUN sharing still has all the same requirements it always had, but it is a way of showing that many different platforms could be participating in this configuration. Netware could as well, however without SDD and multi-path.

The ESSs -using PPRC - are mirroring data between themselves, rather than requiring any host to do the mirroring. (This mirroring could be done in both directions if both sites are normally in use rather than one being a hot backup.) Should an entire site fail, the hosts and configuration in the remaining site could be set up to run with the mirrored volumes. (How this switch-over is done is beyond the scope of this paper.) Notice that because it is ESSs that are connected - and not the SANs - the server clusters in the two sites are separate and only access copies of each other’s LUNs. The servers are not involved in the mirroring - indeed they are not even aware it is going on - and thus a server in one site never needs to access a LUN in the other site. (Unlike the host-based mirroring done in Figure9 on page21.)

PPRC requires ESCON attachments between the ESSs. These ESCON attachments are dedicated to the PPRC function, and are uni-directional. If you have two sites backing each other up (mirroring in both directions) then you need at least two ESCON attachments, four for redundancy.

PPRC is set up between two Logical SubSystems (LSSs) one designated as primary and one as secondary. There can be up to 8 ESCON connections between a pair of LSSs. A LSS acting as a primary can mirror to 4 different secondary LSSs, but any one LUN is

only in a single PPRC relationship. Thus, a primary LSS could mirror 4 LUNs, each to a different ESS, but anyone one LUN would have only one secondary copy.

These connections can be up to 103 kilometers. This requires conversion from multi-mode to single- mode fiber, and some kind of “amplifiers” such as ESCON directors or the IBM 9029 or some other similar device. Keep in mind that ESCON attachments run at 17 megabytes/sec as compared to Fibre Channel’s 100 megabytes/second.

NOTE: PPRC requires TCP/IP connectivity between the Ethernet ports of the ESS.

4.3 ESS Configurations - SDG attach

Before the general availability of ESS Fibre Channel adapters, its only SAN participation was through the 2108-G07 SAN Data Gateway (SDG), or with ESS feature code 3020 - the Interim Host Adapter - which included a 2108-G07. While no one is building new configurations this way, there are many still out there and thus they will be described here.

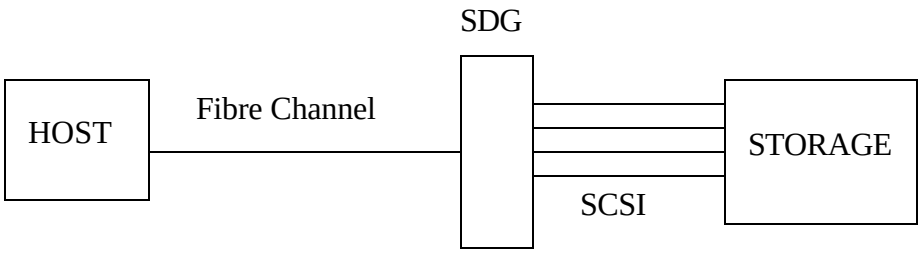
There are multiple support matrices on the web for the SAN Data Gateway. They can be found starting at

http://www.storage.ibm.com/hardsoft/products/sangateway/supserver.htm

4.3.1 SAN Data Gateway Basics

The 2108-G07 SAN Data Gateway is, at the simplest level, a protocol converter between SCSI-3 over Fibre channel (also called FCP or Fibre Channel Protocol) and SCSI over SCSI buses.

FIGURE 14. Usage of SAN Data Gateway



The SDG acts like SCSI Initiators when talking to the real SCSI Targets on the SCSI side, and assumes the role of SCSI targets when talking across Fibre Channel to Fibre Channel hosts (SCSI Initiators.) While the SDG is capable of attaching SCSI Targets (storage devices/subsystems) on the FC side and SCSI Initiators (hosts) on the SCSI side. IBM does not support these kinds of configurations.

The 2108-G07 can have up to 6 fibre channel ports and 4 SCSI ports. These can be left “wide open” - all FC ports capable of using all SCSI ports - in which case all hosts must be the same operating system and be using some mechanism to share volumes. Or zoning in the SDG (similar to, but not the same as zoning in a switch) can be used, in which the fibre channel ports can be restricted to only using certain SCSI ports. The 2108-G07 also supports LUN masking which allows complete control of which hosts can see which LUNs. LUN masking can be used instead of, or in combination with zoning. We will examine zoning first, then we will look at how the SDG maps SCSI LUNs to LUN#s on the Fibre Channel ports, then we will discuss LUN masking.

4.3.2 Zoning in the SDG

In the following examples of zoning we use only 3 Fibre Channel ports to keep the pictures simple. It is a simple extension to more than 3 FC ports.

FIGURE 15. SDG zoning sample 1

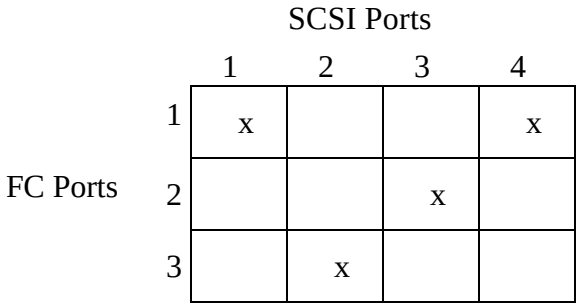
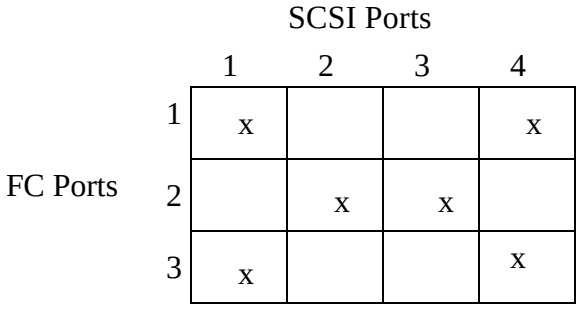


Figure 15 on page30 is one example of zoning. In this particular example, only FC port 1 can use SCSI ports 1 and 4, FC port 2 can use SCSI port 3 and FC port 3 can use SCSI port 2. However, *any zoning combination is configurable - even bizarre, unusable combinations, so care must be taken to ensure that SCSI ports are only shared by FC ports used by hosts that can handle shared access to LUNs.* The example in Figure15 on page30, shows no sharing. An example of zoning to configure some sharing is in the figure below.

FIGURE 16. SDG zoning sample 2



In the figure above, if there were a single host directly attached to FC port 1, and a single host directly attached to FC port 3, and no LUN masking was in effect, then these two hosts would need “shared access capability” as they would be reaching the same LUNs. Note that with zoning, if more than one host can reach the *same* FC port on the SDG, they also must have “shared access capability” since they can reach the same SCSI ports and thus the same LUNs. (LUN masking allows further restriction, see “LUN Masking via VPS (Virtual Private SAN) software” on page33). In the picture above, if a single host was directly attached to FC port 2 it would not need any “shared access capability” because it would be the only host accessing the LUNs on SCSI ports 2 and 3. This host could even be a different operating system from the hosts using FC ports 1 and 3.

NOTE: Fibre channel ports are faster than SCSI ports, therefore a one-to-one correspondence between FC and SCSI ports usually results in an under-utilized FC port. It is usually better to have additional SCSI ports feeding the same FC ports to take advantage of the speed of fibre channel.

4.3.3 Gateway mapping

On the SCSI side of an SDG, the gateway will masquerade as a single Initiator ID for each SCSI port (default uses ID 7). The SDG keeps track of which host is using a SCSI bus at any given time and where incoming SCSI commands and data need to go. This includes honoring SCSI Reserve commands and ensuring that inappropriate commands from a different host don’t flow to a reserved volume. (The downstream storage subsystem thinks everything is coming from the same Initiator ID so some policing must be done by the SDG itself.)

On the Fibre channel side, an SDG Fibre Channel port appears to a host’s file system as a single SCSI Target ID. (Remember SCSI IDs do not flow in the SCSI commands and data as discussed in Section1.1.4 on page8. The HBA driver will present to the host file system a SCSI ID for each FC destination it can reach.) SCSI commands and data do include a LUN#, however. Each real targetID/LUN pair on the SCSI side must appear as a unique LUN on a single target ID on the Fibre Channel side. It is quite possible for there to be the same LUN # (e.g. LUN 1) on two different target IDs on the SCSI side. These would have to be mapped to two different LUN#s on the FC side since in a host all LUNs from the same FC target appear to be at the same SCSI ID. A detailed understanding of how this mapping occurs is valuable for keeping oneself out of trouble!

A single SDG only has a one pool of available LUN numbers - from 0 to 255 - to be used on the Fibre Channel side of the box. This one pool of numbers is for the entire SDG, the same pool for every FC port. Any real device (LUN) on the SCSI side will always be mapped to the same LUN# on every port on the FC side. In some cases the LUN# may be unavailable due to zoning - see “ Zoning in the SDG” on page30 - but where that device is available it will always use the same unique LUN# on the FC side, and no other device will use that LUN# anywhere on the FC side.

LUN 0 on the FC side is reserved for the SDG itself (referred to as the Command and Control LUN. This can be changed to a different LUN# with a simple command.) Thus, an SDG can support at most 255 real LUNs on the SCSI side. The first time the SDG boots it will scan each SCSI port in order, looking for targets and LUNs, and will assign each real target/LUN pair on the SCSI side, to a LUN# on the FC side starting with LUN # 1 and going in numerical order. This mapping is stored in a simple text file in non-volatile storage - called the mapping database -and persists through boots and power cycles unless specifically changed later.

Every time the SDG boots, it re-scans the SCSI ports. Also, you can direct it to re-scan a single SCSI port, or all ports. Any time it scans SCSI ports, the SDG adds to the database if it finds new devices, using the lowest *unassigned* LUN#s first and going forward from there.

If the SDG scans the SCSI ports and finds that some assigned LUN #s no longer have any real devices out there (devices have been removed), those LUN#s are *not* removed from the database, they remain assigned to the real TargetID/LUN it had before. This is because a device may only be temporarily offline such as for maintenance, and when it “comes back” hosts will be expecting to find it at the same LUN#. If the device has been removed permanently, and you want to reuse the LUN# for another device, you can have the SDG re-scan all the SCSI ports and un-assign all LUN#s that don’t have real devices out there on the SCSI buses.This procedure does not affect the original mapping of still-existing devices, so that there could now be holes in the database. (Remember your hosts are expecting devices at certain LUN #s and you don’t want the existing devices to change LUN#s).

You also have the option of completely erasing the mapping database. Then when the SDG reboots, it will re-scan the SCSI ports in order, rebuilding the database from scratch.

There is a command to just list the mapping database, in order to determine what LUN# a given targetID/LUN pair is currently using. Also, the ASCII file that contains the database can be uploaded, edited, and re-downloaded for custom mapping. Editing is risky since the file needs to be in a particular format, but this upload/download process can be used to copy a particular mapping to another SDG.

The reason all of this is important, is that every operating system or file system has its own implementation of what it can do with SCSI over Fibre Channel. Some operating systems cannot use LUN#s greater than 7, others max out at 15 or some other number (most recent systems support up to LUN # 255 for FC HBAs.) Furthermore, some operating systems cannot use a device at a different LUN# from what it started from, without some effort (configuration or even reboot). All of this must be considered when configuring hosts to talk to storage devices through an SDG.

Some basic restrictions come out of this mapping scheme:

- Total maximum LUNs on the SCSI side will be determined by the maximum LUN# supported on the host side.

- A given real targetID/LUN can only be reachable from one SCSI port on the SDG. The SDG will “see” LUNs found on different ports as different devices and will map them to different LUN#s on the Fibre Channel side.
- The LUN# assigned to a given target/LUN on the SCSI side is not used for any other target/LUN anywhere in the SDG. There are configurations with zoning where, if it were allowed by the SDG, one could safely reuse a LUN# on a different FC port to map to a different SCSI target/LUN because the zoning would keep things separate. However, the SDG itself does not allow this, as it only manages a single pool of LUN#s, and never uses a given LUN# for more than one real target/LUN.

4.3.4 LUN Masking via VPS (Virtual Private SAN) software

The newer versions of the 2108-G07 provide LUN masking capability called VPS. Older SDGs can be upgraded to use LUN Masking by purchasing RPQ 8S0511 and upgrading the microcode. LUN masking allows you to specify explicitly which LUN#s on the FC side can be reached by which host adapters. A matrix is presented with LUNs across the top and host WWNs along the side. A check mark in a box means that the host adapter in that row can reach the LUN in that column. Thus, multiple hosts might have access to the same SCSI ports, but will only see certain LUNs in the SDG. If you are using zoning simultaneously with LUN masking in the same SDG, then LUN masking happens only within a given zone. A server will never see a LUN not in its zone, even if LUN masking has allowed it.

NOTE: VPS also allows the host-type to be specified by the WWN of the host adapter. (Without VPS, every host for a given FC port on the SDG must be the same host-type e.g. all NT, or all AIX, etc.) Thus, in order for different host types to use the same FC port on an SDG, VPS must be used. See discussion of other SDG types in Section8.1 on page68

4.3.5 Other SDG restrictions

There are a number of other restrictions on the use of the SDG:

- Platforms that are supported to the ESS through native FC attachment, may not be supported via SDG. Care must be taken to look at the websites to ensure IBM support of any given configuration.
- LUN 0 is reserved for the SDG itself so if an operating system has a special requirement for LUN 0 this is not available by default. The number used can be changed to any other LUN #. Whichever LUN# is used is then reserved on all FC ports.
- SDD through the SDG to the ESS is not supported. This has implications for high availability options (no single-host-multi-adapter option) and is discussed later in this paper.
- AIX’s HACMP without SDD is supported through the SDG, for two-server clusters. Each server must have its own SDG for access to the ESS. You cannot have an HACMP configuration in which one of the servers accesses the shared LUNs via FC through the SDG and one of them accesses the LUNs via direct SCSI attach to the ESS.

- The SDG has restrictions as to how hosts can connect to it. Hubs can only be used to extend the distance of what would otherwise be a direct connection (The hubs are merely being used as repeaters in this case.) Connections using switches can have various restrictions as well, and some combinations are only valid for certain levels of ESS LIC code. Be sure and check both the ESS website and the SDG website carefully.

4.3.6 ESS Interim Host Adapter

When the ESS first announced the native FC adapter, (depending on platform) it announced an Interim Host adapter that is really a SCSI adapter on the ESS and a SAN Data Gateway (plus some cables.) Even though the SDG hardware that ships with an Interim Adapter has two FC ports and 4 SCSI ports, support for the Interim Adapter will be based on using just one FC port and 2 SCSI ports. (When buying an Interim Adapter, the customer only pays the cost of and ESS SCSI adapter. When the native FC adapter is available for the ESS, the customer will only pay the difference in the price of the two adapters and give back the old hardware. The SDG is a free loan.)

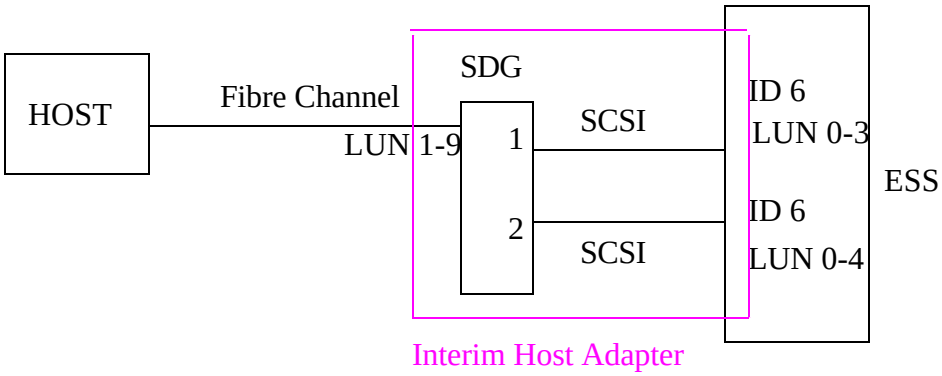
The Interim Host adapter is based on the newer SDG hardware, and comes with the same function and capability as announced for the SDG-G07s on 3/28/00. These new functions include the LUN masking described earlier in this paper, the ability to configure all ports for SCSI Initiator and/or SCSI Target roles, up to 6 shortwave FC ports (previous maximum was three) and the option for a longwave port that uses single mode fiber and can reach up to 10 kilometers. (A longwave port takes the “real estate” of two shortwave ports, thus the maximum FC ports on and SDG is less than six if any longwave ports exist.) The current SDG also provides for non-loop attach, that is, no arbitration from the fibre channel port. The old one was using loop arbitration protocol even if directly attached to a server or attached to a switch, and thus could not be used with a McDATA switch. The new SDG can be attached to the McDATA switch.

4.3.7 Simple Host Connection (ESS Interim Host Adapter)

NOTE: All of the following configurations using the SDG show only direct connections, but everything works the same if a hub is used for distance extension or if switches. Not all combinations are supported for all platforms so check the websites to be sure. If switches are used, and there is a need to restrict certain hosts to only see certain LUNs. some combination of zoning in the fabric, and zoning and/or LUN masking in the SDG would be needed to handle this.

Figure 17 on page35 is a drawing of one host, fibre channel-attached to a SAN Data Gateway, which is SCSI attached to an ESS. This is also representative of an Interim Host Adapter connection. The Interim adapter is basically the portion of the picture in the magenta rectangle.

FIGURE 17. Simple host connection (ESS w/SDG)



In our discussion of using the ESS with the SDG, a logical device in the ESS is a targetID/ LUN# pair on a SCSI bus that is mapped in the SDG to a LUN# on the fibre channel side. In the figure above, the ESS has Target ID 6 with LUN#s 0-3 on SCSI bus 1 and the SDG maps them to LUN#s 1-4. The ESS also has Target ID 6 with LUN#s 0-4 on SCSI bus 2. The SDG maps these to LUN#s 5-9 on the Fibre Channel port. (A Target ID on the FC side does not really exist. Host adapter drivers will map a FC destination address into a SCSI target for the OS to use, but this does not really exists any where on a SAN.

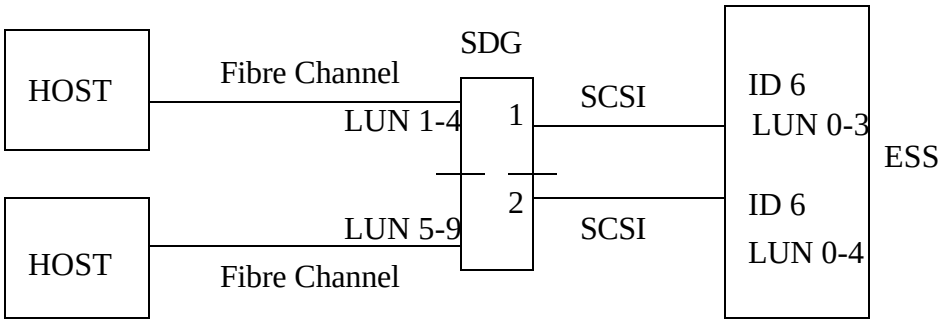
The most important thing to note in the configuration above is that the ESS must be configured such that the two SCSI buses cannot access the same LUNs. This is because the SDG will map the devices found on different SCSI buses onto different LUN#s. (The SDG assumes they are different devices.) Because they have different LUN#s the host will see them as different devices also, *so you need to ensure that they really are different devices by appropriately configuring the ESS.*

This need to keep SCSI ports from the same SDG from seeing the same LUNs in the ESS is true regardless of which configuration we are looking at. In all cases, you must never be able to get to the same LUN on two different SDG SCSI ports on the same SDG, because the SDG will always treat them as separate LUNs. (There is no SDD-like capability inside the SDG, and SDD is not supported for use through the SDG.)

4.3.8 Disk Consolidation - ESS and SDG

Figure 18 on page36 is a picture of two hosts attached to a single SDG and then to a single ESS. This is not an example of the Interim Adapter, as the Interim Adapter only uses a single FC port, but it serves to illustrate some features of the SDG. For this example we are assuming that the hosts are using different file systems and thus cannot have any access to each other’s volumes.

FIGURE 18. Multi-host connection to ESS with SDG zoning



In order to keep everything separate, either the SDG must be zoned such that each FC port (attached to a single host), can only make use of a single SCSI port, *or* LUN masking must be used to keep the hosts from seeing the different LUNs. This configuration does not require LUN masking, so only zoning will be discussed. An example using LUN masking follows later.

The SDG zoning needed for Figure18 on page36 is represented in the following figure (only FC ports 1 and 2 and SCSI ports 1 and 2 are relevant):

FIGURE 19. SDG zoning sample 3

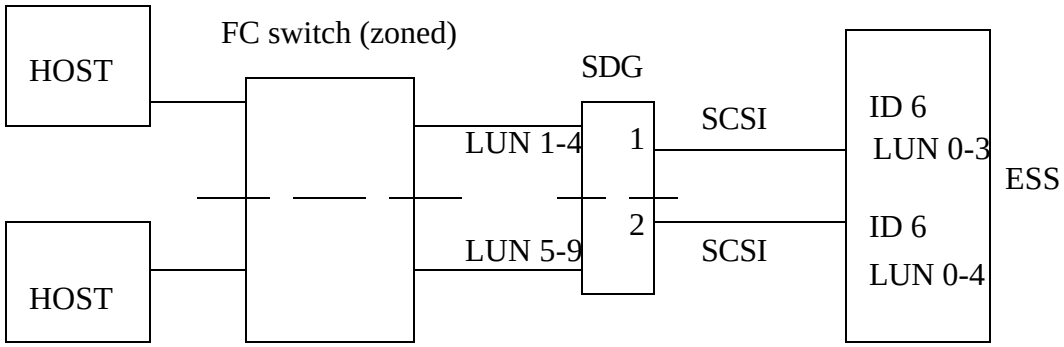
FC Ports \ SCSI Ports	1	2	3	4
1	x		N/A	N/A
2		x	N/A	N/A
3	N/A	N/A	N/A	N/A

Looking at Figure18 on page36, since the SCSI ports will not see the same logical devices (always a requirement with a single SDG) then each host will only see a single set of logical devices with no overlap. If the two hosts were both NT, but did not have Clustering Services installed, you would still need the zoning in the SDG to keep the NT hosts from seeing each other's drives. Note, that even though keeping the LUN access separate means that there would be no confusion in the hosts by re-using LUN#s on the FC side, the SDG always uses a different (unique) LUN# on the FC side for a different (unique) device on the SCSI side (these devices are actually logical devices in the ESS.)

NT hosts or SUN hosts using the JNI adapter can come through a switch into the SDG. If everything is connected to the same SAN, then both of the hosts could get to both FC ports on the SDG. This would allow them each to use both SCSI ports, thereby defeating the

zoning in the SDG. To handle this, the switch(es) in the SAN would also need to implement zoning. A one-switch example of this follows:

FIGURE 20. Multi-host to ESS with SDG zoning and switch zoning



With up to six FC ports and 4 SCSI ports on an SDG, this configuration could be expanded to include four different operating systems coming in four different FC ports on an SDG, zoned to different SCSI ports on the SDG, and then going to four non-overlapping sets of logical devices in the ESS.

Below is the same figure as above, but *without* zoning in the switch.

FIGURE 21. Multi-host to ESS with no zoning in SAN, LUN masking and zoning in SDG

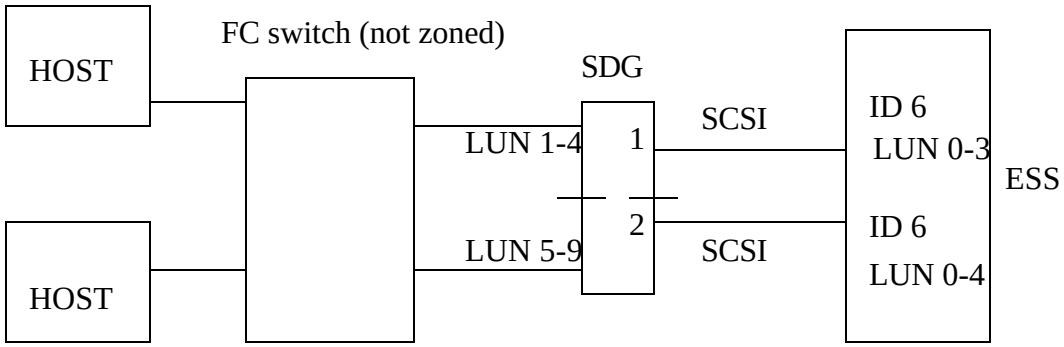


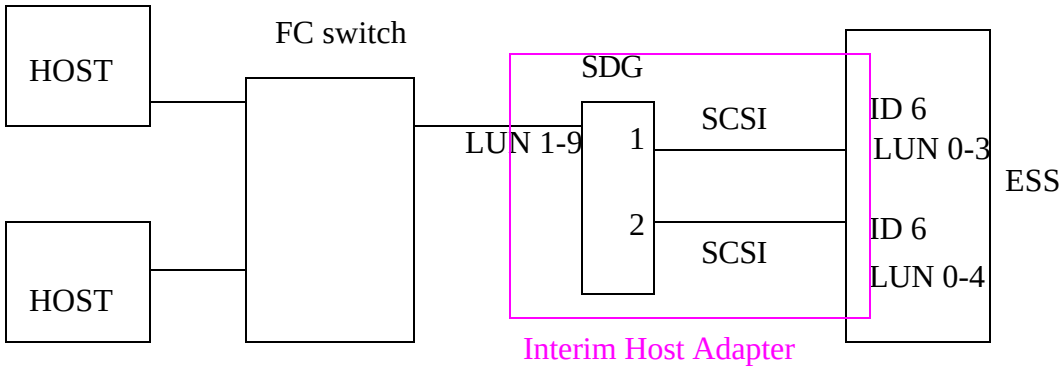
Figure 21 on page37 shows no zoning in the switch but still in the SDG. Using LUN masking in the SDG, you can configure the SDG to only allow the top host's adapter to access LUNs 1-4, and only allow the bottom host's adapter to access LUNs 5-9. In this case the hosts would not even "see" the other LUNs.

However, you still need zoning in the SDG as well! If you only used LUN masking, the top host could reach LUNs 1-4 through either FC port on the SDG. This would look like 8 different LUNs to the host. (The two FC ports on the SDG are the equivalent of two SCSI Targets.) Since SDD is not supported through the SDG you cannot use it to detect that they are the same four LUNs on each port. By zoning the SDG you ensure that only LUNs 5-9

can be seen through the bottom FC port, and the LUN masking then ensures only the bottom host sees those LUNs.

A more common use of LUN masking is as in the following figure. This shows LUN masking in use on an Interim Host Adapter (magenta rectangle)

FIGURE 22. LUN Masking with the ESS Interim Host Adapter



In this configuration both hosts are coming into the SDG via a single FC port. The only way to keep them using separate LUNs is with LUN masking. As long as both hosts are the same operating system you can split up the LUNs anyway you want, and would probably divide them up such that each host is using LUNs on both SCSI buses.

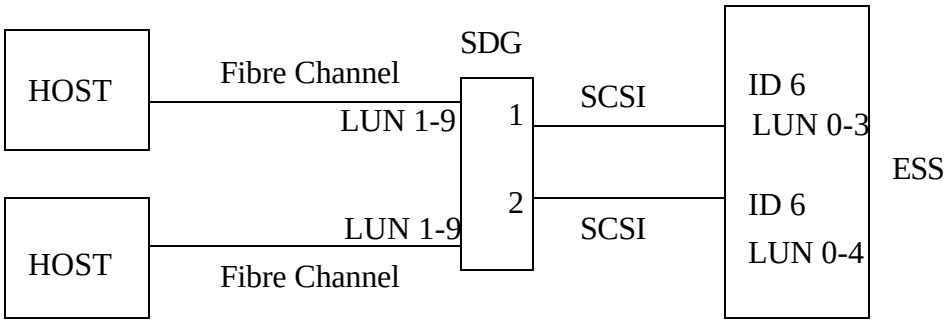
NOTE: If one of the hosts was NT and the other was SUN *you cannot have both hosts accessing LUNs on both SCSI buses as described in the previous paragraph*. This is because, a given SCSI port in an ESS is configured for a particular host type. If you don't segregate the LUNs to different SCSI ports in this case (as done in the previous zoning examples), then you would have a SUN host coming in on a SCSI port defined in the ESS as being for NT hosts, or vice versa, or both. All LUNs in an ESS assigned to a given SCSI port must be for the same host type.

LUN masking is quite useful in situations where the number of desired servers greatly exceeds the "shared access capability" of the server platform. Up to eight different hosts can access the same SDG port, so in principle, up to 48 hosts could come into the same SDG all trying to get through the same 4 SCSI ports. LUN masking can ensure that the only hosts seeing a given LUN are hosts that you want to see that LUN.

4.3.9 Disk Pooling - ESS with SDG

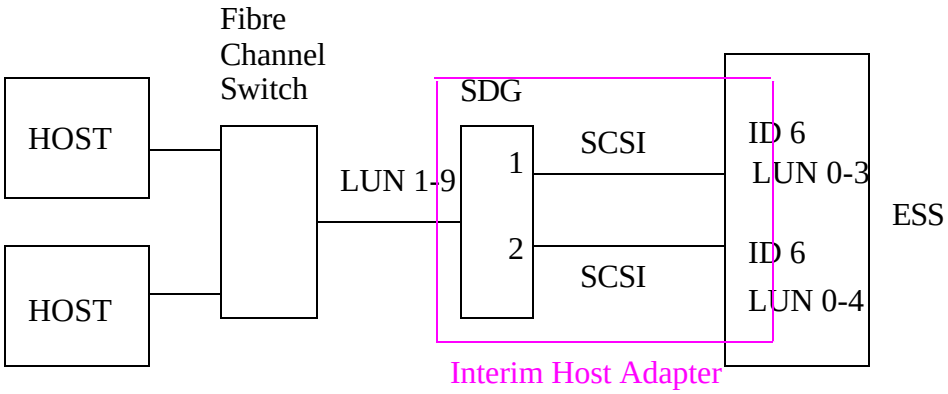
In the figure below, we are using the same configuration as Figure18 on page36, but without the zoning in the SDG, thus allowing both hosts to get to both SCSI ports and all LUNs These hosts would need to be the same platform, and be using some sort of "shared access capability."

FIGURE 23. Shared access to ESS LUNs through the SDG



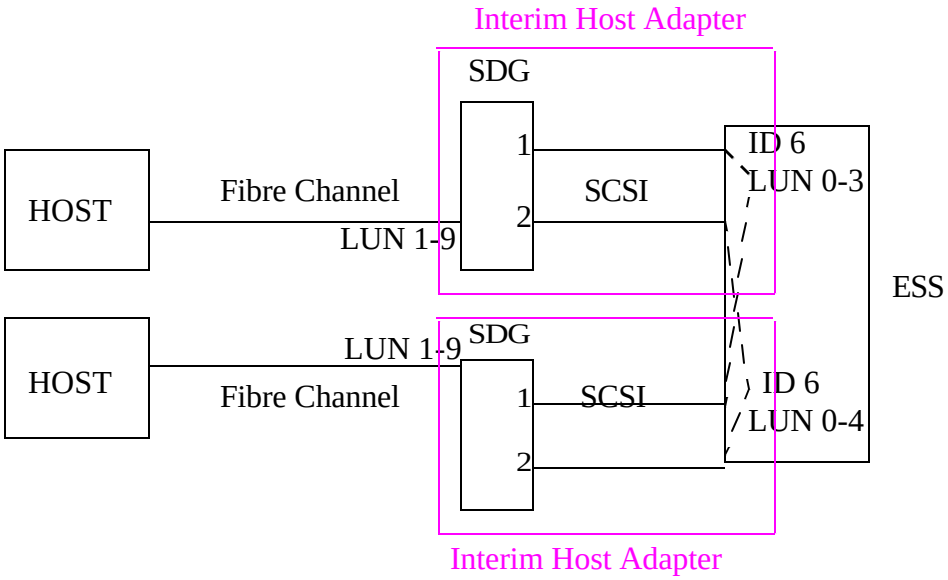
To do this with the Interim adapter I would need either one Interim adapter and a switch, or two separate Interim adapters. The configuration with a switch and one Interim Adapter is as follows:

FIGURE 24. Shared access to ESS with one Interim Adapter



The equivalent picture using two Interim adapters is as follows:

FIGURE 25. Shared access through two Interim Adapters



In the figure above, each LUN in the ESS is associated with two SCSI ports, one SCSI port on each of the SDGs. However, for a single SDG, each of the LUNs on the two SCSI ports are all different.

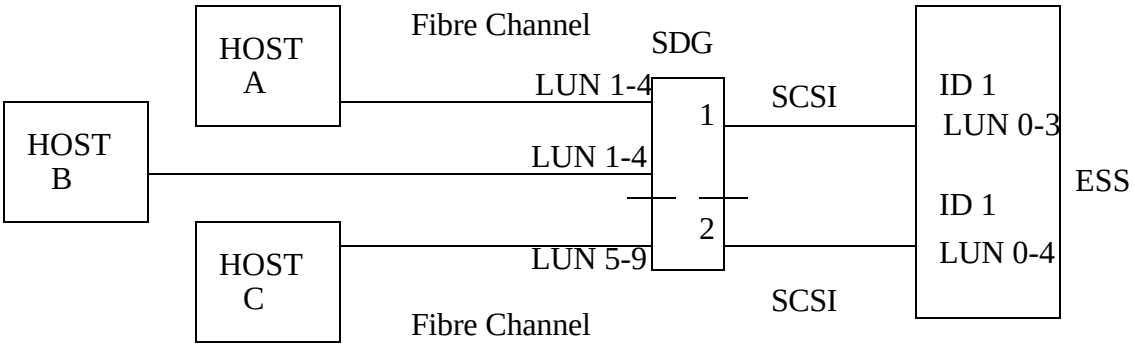
For all three figures, (Figure23 on page 39, Figure24 on page39, and Figure25 on page40) the two hosts must be the same operating system, and some kind of “shared access capability” (clustering) must be used, because both hosts see all LUNs.(In this case you *still* need to keep any given LUN from being seen by two SCSI ports on the same SDG, this is always true, but now each SDG FC port can access all LUNs on two SCSI ports, thus giving each host access to all the logical devices in the ESS.)

A single SDG will map these devices to the same LUN numbers on each of its FC ports, that is, for any given target/LUN - ESS logical device - on the SCSI side, the SDG will use the same LUN# on each FC port on the FC side. In Figure25 on page40, it is possible that the LUN#s used on the FC side of the two different SDGs could be different - if other things had been going on there could be holes in one of the databases or something - but this is OK because in this case it is different servers seeing these LUN numbers, and “shared access” software does not require that they be the same LUN #.

4.3.10 Disk Consolidation and Pooling - ESS with SDG

Below is an example of a configuration that mixes consolidation/partitioning and pooling.

FIGURE 26. Partitioning and pooling in one configuration (ESS w/SDG)



Assume hosts A and B are the same operating system using some sort of “shared access capability”, and host C is a different operating system. The ESS must still be appropriately partitioned such that no logical device is reachable from both SCSI adapters. You could zone the SDG such that the ports to which hosts A and B are attached can only get to SCSI port 1, and the port to which host C is attached can only get to SCSI port 2. Or you could use LUN masking to achieve the same result. Here we are using consolidation with partitioning to keep the unlike hosts from seeing each other’s logical devices, but we are using disk pooling between hosts A and B. This could also be done with three Interim Adapters rather than three FC ports on a single SDG (requires a third SCSI port on the ES), but that picture is too complicated to show here. Also, with more FC ports on an SDG additional combinations are possible, as long as you adhere to the restrictions listed later in “Guiding principles for ESS/SDG configurations” on page48

If just zoning is used in Figure26 on page41, it would look like the following:

FIGURE 27. Zoning for Figure26 on page41

SCSI Ports

FC Ports	1	2	3	4
1	x		N/A	N/A
2	x		N/A	N/A
3		x	N/A	N/A

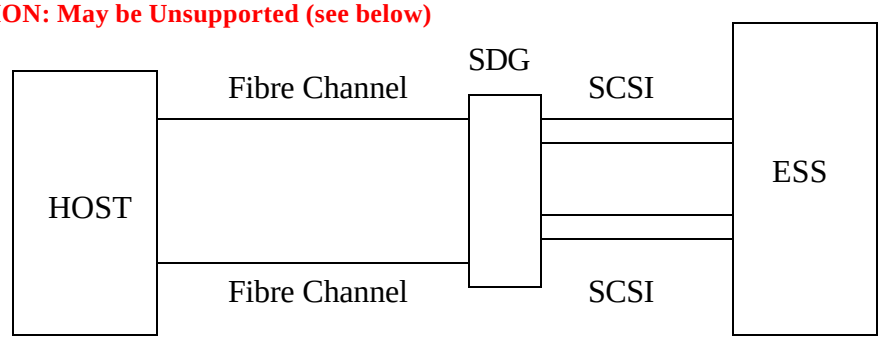
4.3.11 multi-path Single Host Connections - ESS with SDG

While clustering mechanisms (shared access) can certainly work with ESS (see “Disk Pooling - ESS with SDG” on page38), there are no supported high availability configurations through the SDG to the ESS *for a single host*, that is, there is no current support for a single host to access the same logical devices from two different adapters (single-host-

multi-access). Even though a SAN fabric could provide alternate FC paths to an SDG, *these paths would need to go from the same host adapter, to the same destination FC port, for the host to not see duplicate LUNs.* If a single host adapter reaches two different FC ports it sees them as different SCSI Targets. Thus, the only current recovery option on the FC side (when using a SAN Data Gateway) is normal fabric alternate pathing *between a specific host FC port and a specific SDG FC port* (see last paragraph of Section1.1.1, "Fibre Channel components.," on page2 for discussion of alternate pathing within a fabric.*)* Some examples follow.

FIGURE 28. Single host-multi-access (ESS with SDG)

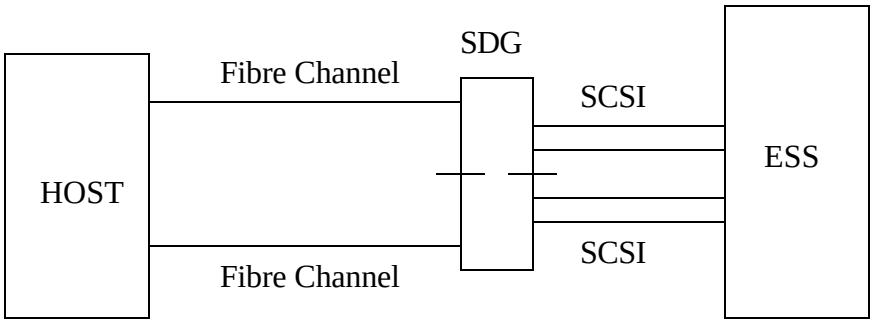
CAUTION: May be Unsupported (see below)



In the figure above, if there is no zoning or LUN masking in the SDG, the host can get to all volumes from either fibre channel port. To make this work, IBM's Subsystem Device Driver (SDD) software would be required to handle the multiple paths, but SDD is not supported through the SDG.

If the SDG were zoned as shown in Figure29 on page 43, then the configuration doesn't require SDD and could work. LUN masking could also be used to ensure that the two HBAs see different LUNs.(This is because LUN masking is done based on the specific host adapter not the host itself.) Note that this is not a high availability configuration, however. Loss of a single component (e.g. HBA, SCSI port, SDG port etc.) loses access to LUNs. There might be some performance benefit in using two FC adapters in the host, but besides keeping the LUNs in the ESS split up so that each one was only associated with one SCSI bus, zoning or LUN masking would also be needed to divide these LUNs up between the two host adapters.

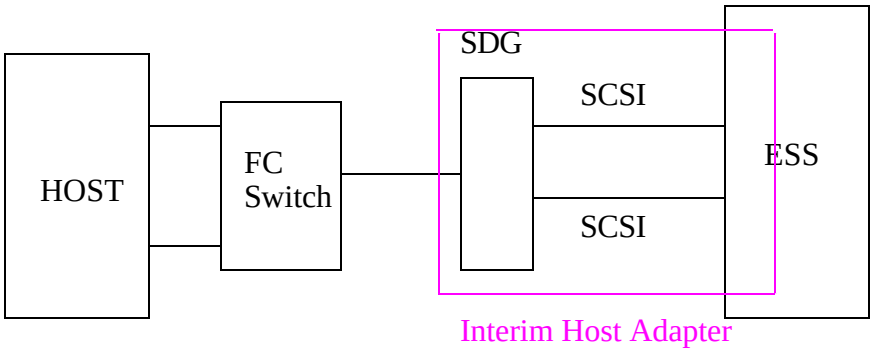
FIGURE 29. Single host, multi-path, no SDD, zoned SDG to ESS



The above figure (Figure29 on page43) is an example of using zoning to defeat single-host-multi-path issues. Each host adapter only sees the LUNs on two of the SCSI buses so no LUN is seen twice. As stated before, this is not a high availability configuration.

A different picture using one Interim Adapter and a switch:

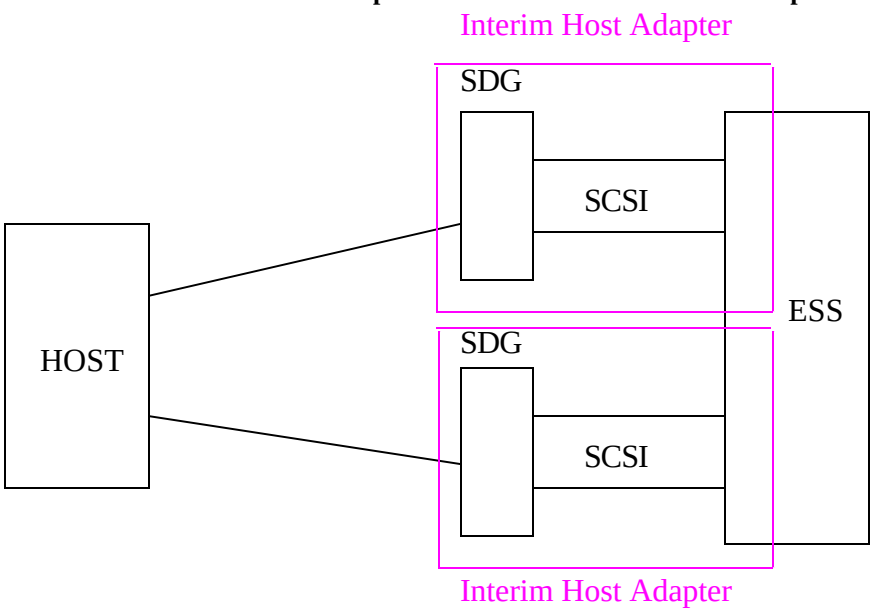
FIGURE 30. Single host, multi-path, no SDD, LUN-masked Interim Adapter to ESS



The figure above requires LUN masking in the SDG to make sure the two host adapters don't see the same LUNs.(Zoning would not work here as there is only one FC port on the SDG in use.) However, this picture is a waste of hardware. It is not high availability (*any* component failure loses access to some LUN) and there is no performance benefit to using two host adapters when all the data is funneled through one SDG port.

If we instead use two Interim Adapters:

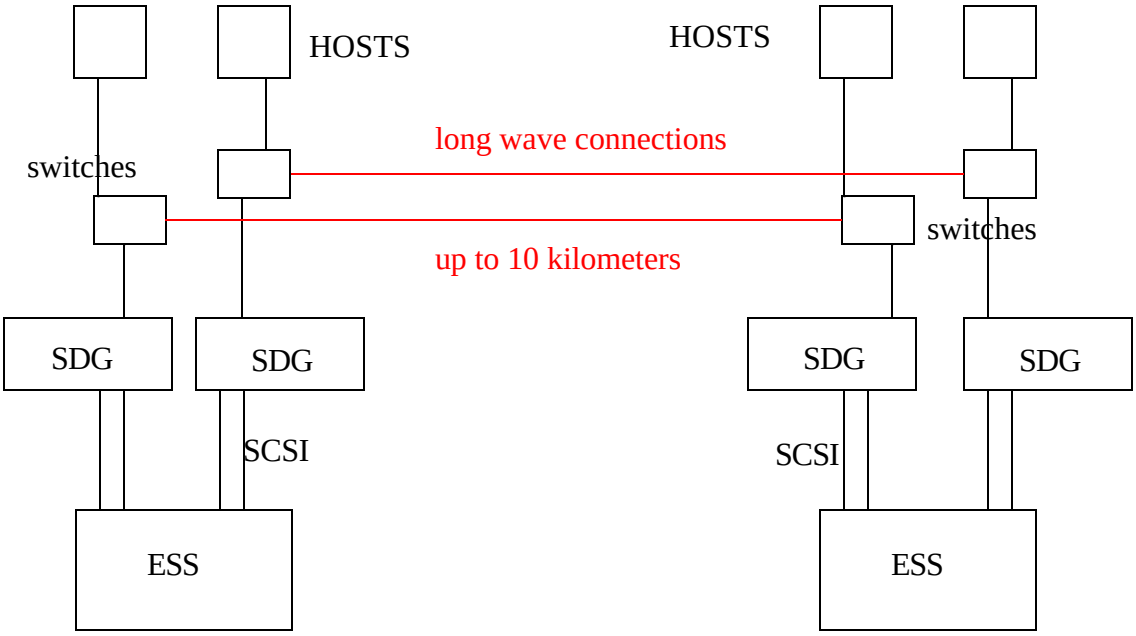
FIGURE 31. One host with two host FC adapters and two ESS Interim Host Adapters



In the figure above (Figure31 on page 44) you still need to have no LUN in the ESS associated with more than one SCSI port. No zoning or LUN masking is needed assuming you have configured the ESS such that every LUN is associated with only one SCSI port. There is still no redundancy. This is not a high availability configuration, but the host has the potential of using more than 100 Megabytes of bandwidth at any point in time, if there is enough simultaneous activity on all four SCSI ports.

Another remote vaulting configuration similar to Figure9 on page21 is available as shown below.

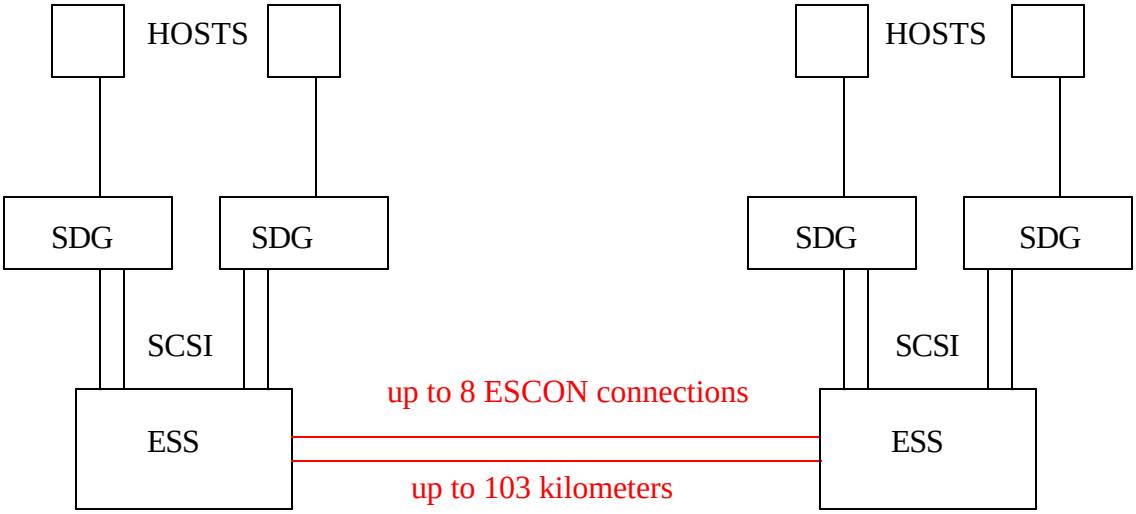
FIGURE 32. Remote Site with SDG and ESS



The basics of Figure32 on page45 - in particular the host-related issues - are the same as for Figure9 on page21. The SDGs in the picture could be Interim adapters since only 2 SCSI ports and one FC port are shown on each SDG. This picture could use SDGs with more ports, or could use additional Interim Adapters and still be valid. Once again, there are lots of issues with how failover occurs between hosts, and how users find the hosts at the remote site. Also this configuration (Figure32 on page45) can only do *host-based* remote mirroring.

Consider the picture below:

FIGURE 33. Remote Copy for ESS with SDG



In the above figure each side of the picture is much like Figure23, “Shared access to ESS LUNs through the SDG,” on page39 (we have shown extra SDGs in this picture). The ESSs themselves are using PPRC to mirror data between them rather than requiring any host to do the mirroring. (This mirroring could be done in both directions if both sites are normally in use rather than one being a hot backup.) Should an entire site fail, the hosts and configuration in the remaining site could be set up to run with the mirrored volumes. (How this switch-over is done is beyond the scope of this paper.)

PPRC requires ESCON attachments between the ESSs. These ESCON attachments are dedicated to the PPRC function, and are uni-directional. If you have two sites backing each other up (mirroring in both directions) then you need at least two ESCON attachments, four for redundancy. The ESS supports up to 8 ESCON connections for PPRC, and the mirroring can be done to up to four backup sites.

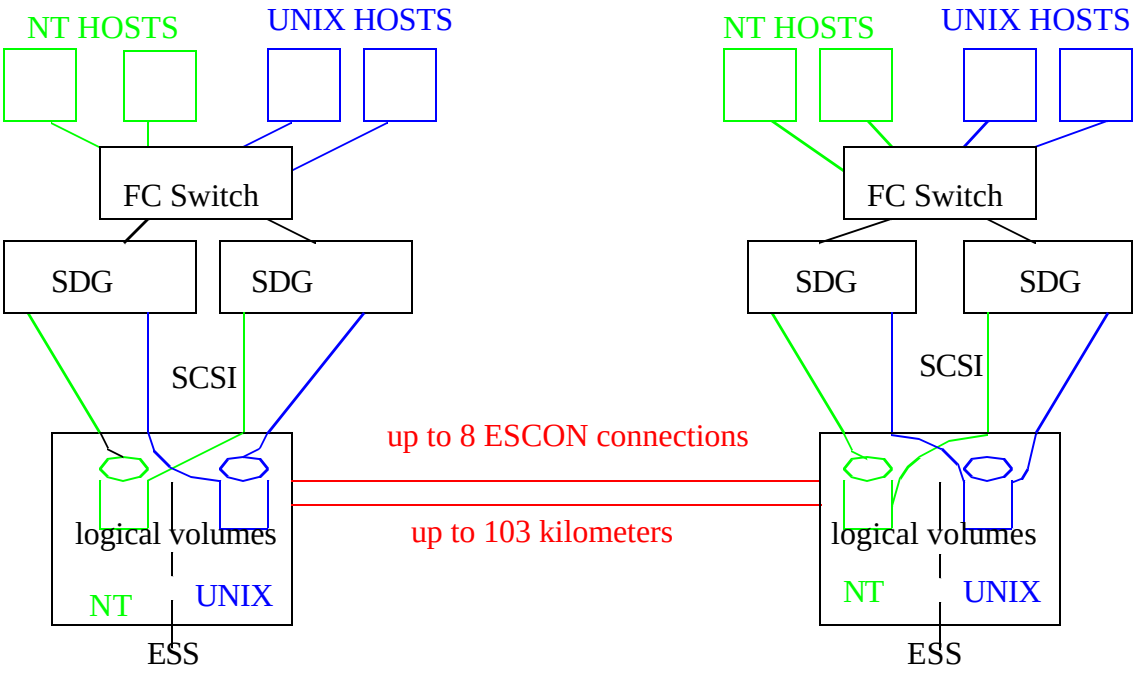
These connections can be up to 103 kilometers. This requires conversion from multi-mode to single- mode fiber, and some kind of “amplifiers” such as ESCON directors or the IBM 9036. Keep in mind that ESCON attachments run at 17 megabytes/sec as compared to Fibre Channel’s 100 megabytes/second.

Another difference between Figure 32, “Remote Site with SDG and ESS,” on page45 and Figure 33, “Remote Copy for ESS with SDG,” on page46 is that there are now two clusters of hosts rather than one large one. This is because there are two SANs now, one in each site. The only connection between sites is the ESCON attachments between the ESSs. Thus, our maximum cluster-size consideration is by site now, not for the whole configuration. (In the case of NT, in Figure32, “Remote Site with SDG and ESS,” on page45 we required two of the hosts to be offline to keep the cluster sized down to two. This meant that the remote site had to be a dormant “hot” backup. In Figure33, “Remote Copy

for ESS with SDG," on page46 we have two clusters of two hosts each and both can be on-line at all times.

By using LUN masking in Interim Adapters this remote backup can be extended to multiple operating systems as in the picture below.

FIGURE 34. PPRC with heterogeneous hosts (ESS w/SDG)



The figure above makes use of the following:

- Clustering Services on the NT boxes to allow shared access to the same LUNs
- "Shared access capability" of some kind in the UNIX boxes. Clustering mechanisms require that the servers in the cluster be running the same OS.
- LUN masking in the SDGs coupled with appropriate ESS configuration to ensure that

  a) a given SCSI bus only has LUNs for the appropriate host type

  b) UNIX hosts can only get to SCSI buses that have their LUNs on them (in the ESS),

  c) NT hosts can only get to SCSI buses that have NT LUNs on them (in the ESS) ***and***

  d) ensure that each host can only see a given LUN from one unique pair of host adapter and SDG FC port. (one path to a LUN from a given host)

Other considerations for Figure34, "PPRC with heterogeneous hosts (ESS w/SDG)," on page47:

- SDGs not used as Interim Adapters can use up to 4 SCSI ports, very useful in a complicated configuration like this
- SDGs not used as Interim Adapters can have up to 6 FC ports so it can be expanded further

- More SDGs, more hosts, more ESSs etc. can be added as long as you adhere to various box constraints and basic rules as spelled out below in “Guiding principles for ESS/ SDG configurations” on page48

4.3.12 Guiding principles for ESS/SDG configurations

The configurations shown are for illustrative purposes and do not show all the possibilities available. The ESS can have up to 32 SCSI ports, allowing very complex configurations, but showing these would make the pictures unusable. Following are guidelines that apply to every configuration you may wish to put together:

- One SDG must not get to the same logical devices through two different SCSI ports.
- If two hosts can get to the same logical devices, they must be the same operating system and use appropriate “shared access capability.”
- Single-host-multi-access through the SDG is not supported. Some combination of switch zoning and/or SDG zoning, and/or SDG LUN masking must be used to ensure that no host has more than one path to a given device. Because of the complexity of large SANs it is possible to inadvertently end up with multiple paths *out a single host adapter* to the same LUN. All that’s required for multiple paths to exist from a single host adapter to a given LUN, is for there to be a different destination port that reaches that same LUN.
- LUN restrictions in the hosts could make otherwise valid configurations unusable by those hosts if the SDG assigns LUN#s that the hosts cannot use. (Rarely an issue now that operating systems allow 255 LUNs on FC adapters.) If a given platform relies on LUN # 0 you may need to change the Command and Control LUN of the SDG to a different value.
- Fibre channel speeds are generally faster than SCSI, thus you normally want to use more SCSI ports than FC ports to make use of the FC bandwidth.
- ESS LUNs associated with a given SCSI port in the ESS must all be used by the same server type, which must match the server type configured for that port in the ESS
- Interim Host Adapters only use 1 FC port and 2 SCSI ports on an SDG

:

4.3.13 S/390 and ESS

The ESS can also participate with a S/390 and a SAN via an ESCON/FICON bridge or with its own native FICON adapter. This will be covered in either a later version of this paper, or a separate paper.

5.0 7140-160 Disk Controller Configurations

5.1 7140-160 Basics

The 7140-160 is an SSA disk controller with a Fibre Channel port for connection to hosts. The physical disks on the SSA loop can be used in various ways. If a single SSA disk is mapped into a LUN on the FC side, this is called a “simple drive.” Multiple disks can be combined into a single LUN, and this is called a “composite drive.” RAID 1 mirroring can be set up for either simple drives or composite drives. (The term “complex drives” is used as a generic term to refer to mirrored, composite, or mirrored composite drives.) Various other SSA disks can be left as spares, either general spares for the whole loop or dedicated spares for a specific mirrored or mirrored composite drive.

The FC port on the 7140-160 is a FC_AL (loop only) port. A second port can be added, but this port is on the same loop - more of a passthru port - and can only be used for loop extension. That is, the 7140 is only one station on a single loop regardless of whether using one or both ports. For instance, if you were to connect two hosts directly to two ports on a single 7140-160, that would be a single loop with 3 stations on it - the two hosts and the 7140 itself.

The 7140 acts as an SSA host on the SSA side of the box, and as a FCP (SCSI) Target on the FC side of the box. Up to 8 7140s may be on the same SSA loop.

5.1.1 LUN access in the 7140-160

Every drive that is configured on the SSA loop - be it simple or complex - is given a unique LUN number on the FC side. These are the LUN#s that a host will discover when it accesses the 7140. Every drive on the SSA loop is also configured to be either public or private. Public means that any 7140 on the SSA loop can use the drive, private means that only 7140s that have been specifically configured to access that drive will do so.

Once a host reaches a 7140, *it will see every LUN that that 7140 uses.* In other words, any host reaching a 7140 sees all public drives and all private drives that that 7140 has access to. If more than one host reaches the same 7140, each host will see the same drives through that 7140. (There is no LUN masking in a single 7140.) The only way to have different hosts see different drives on the same SSA loop, is to have multiple 7140s on the same SSA loop (maximum of eight), and restrict the hosts to specific 7140(s). Some examples of this will follow.

5.2 Configurations using the 7140-160

Supported host platforms and SAN connectivity can be found at

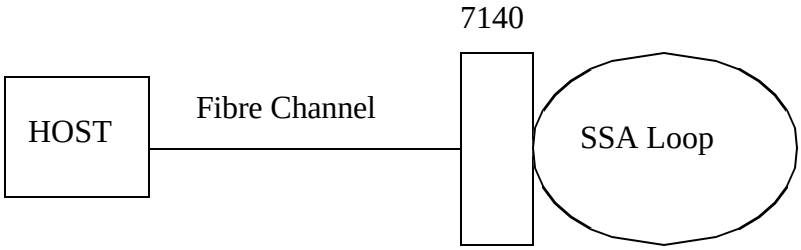
http://www.storage.ibm.com/hardsoft/products/san160/160_support_matrix.html

As of 1/1/2002, only 2109 S Models are supported for connectivity, or 2103 hubs for distance extension only.

5.2.1 Simple 7140 configurations

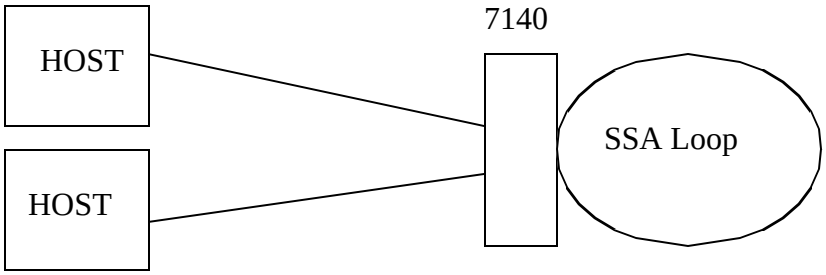
Below is the simplest use of a 7140.

FIGURE 35. Basic 7140 configuration



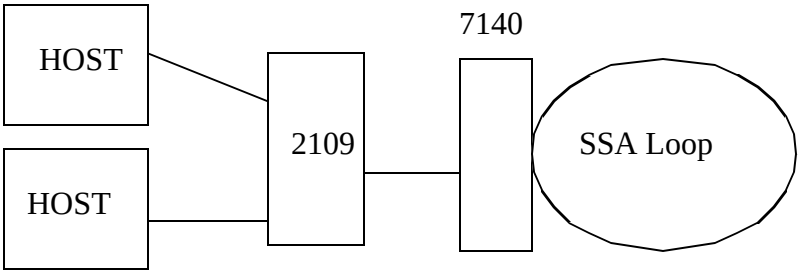
In the figure above, a single host can access SSA disk as if they were FC attached. The host can be allowed to see each SSA disk as a LUN (JBOD), or the drives can be combined as described in “7140-160 Basics” on page49. From the host point of view there are LUNs behind a FC address (SCSI Target). They may be of different sizes depending on how the composite drives are set up, and any mirroring done by the 7140 will be transparent to the host.

FIGURE 36. Two Hosts, one 7140, no switch



In Figure36 on page50, there are two hosts connected to two ports on a single 7140. In this case there is a single Arbitrated Loop with three stations (two hosts and the 7140.) Booting one host would disrupt all traffic as described in “Hubs and Loop Initialization” on page5. The only way to do avoid this is use as a switch as follows:

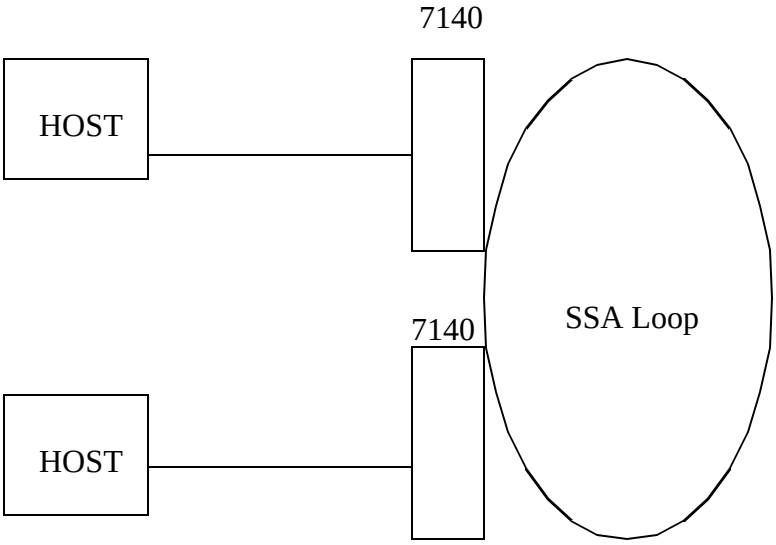
FIGURE 37. Two hosts, with a switch to one 7140.



In the figure above there is a small 2-station loop between the 7140 and switch port it is connected to. Hosts may be using direct fabric attach (no FC_AL used). They could also use FC-AL, but would then be on separate 2-station loops. In either case, booting a host would not affect any I/O flowing to or from the other host.

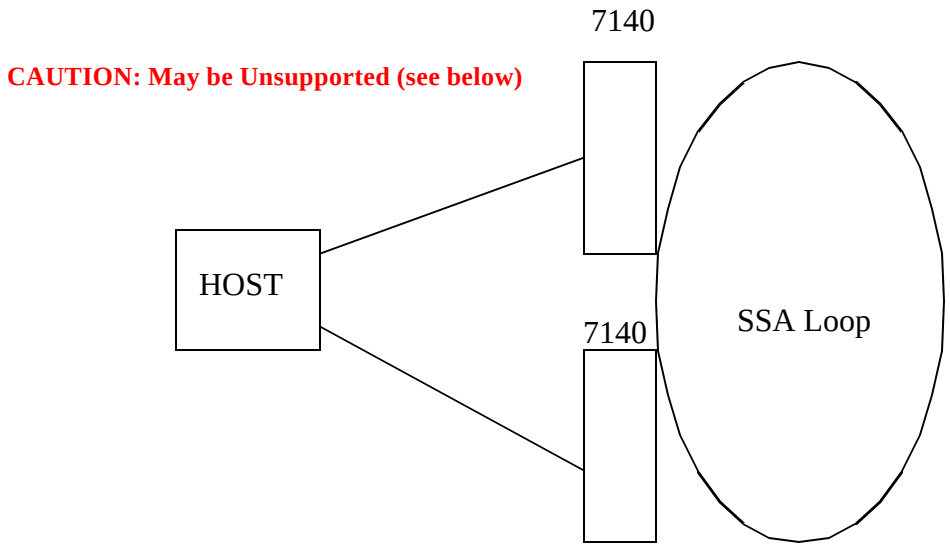
Also, in both Figure 36 on page50 and Figure 37 on page51, since both hosts reach the same 7140, both hosts see all drives that that 7140 sees (either public or private to that 7140), Thus some kind of “shared access” (clustering between like platforms) must be in place. You cannot have hosts of different platforms accessing the same 7140.

FIGURE 38. Unshared access to one SSA loop



In the figure above, it is possible for the two hosts to be different platforms. This is because some drives on the SSA loop can be private to one of the 7140s, while the other drives are private to the other 7140. A maximum of eight 7140s can be on the same SSA loop.

FIGURE 39. Multiple paths to one SSA loop



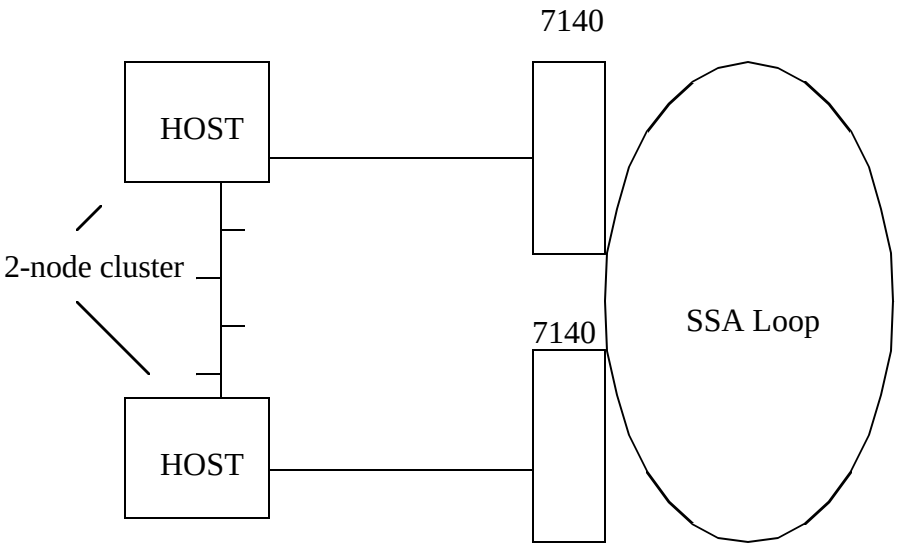
The figure above may or may not be supported. As of 1/1/2002, there is no SDD support for 7140s. This means that if the host sees the same drives through each HBA, it will see them as separate drives. (VICOM supports Veritas DMP in this configuration, but there is no IBM support for multi-pathing through the 7140.) This may or may not change over time.

The figure above could work if all drives are private, and if any given drive is only assigned to one 7140. In this case the host would only see each drive out one HBA.

This is valuable for performance reasons. SSA loops do not have arbitration - all devices on the loop can be transmitting and receiving at the same time (referred to as “spatial re-use.”) When a new host is added to an SSA loop, throughput will typically increase on the loop, because there is now another host driving I/O on the loop. The 7140s act as hosts on the SSA loop, and thus by adding 7140s to a loop we may see an increase in utilization of the loop. We also increase the available bandwidth on the FC side with the additional port on the new 7140.

However, this is not a high availability configuration. Loss of an HBA or a 7140 or a link between them would result in loss of access to some LUNs. High availability can be achieved at the cluster level as shown in the following figure:

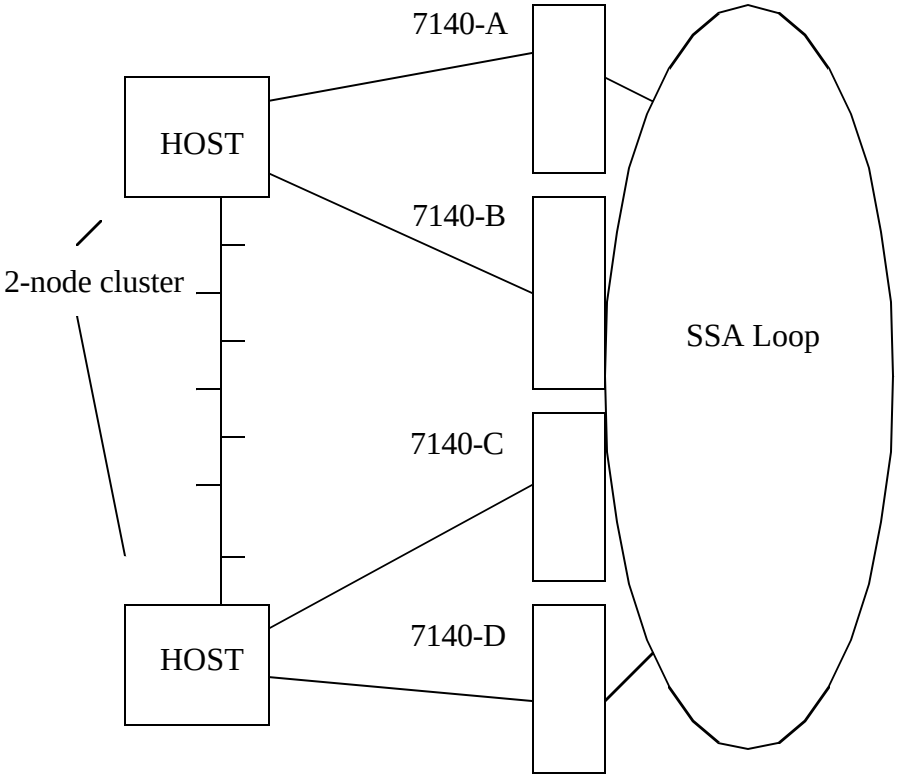
FIGURE 40. High Availability (via clustering) with the 7140



In the figure above we see a typical 2-node cluster attached to two different 7140s on the same SSA loop. Now, any one host only sees each LUN through its 7140. Drives being shared by these two clustered hosts could be either public, or private but allowed to both 7140s. In the figure above, failure of a 7140, or a FC port, or a link or FC path, or an HBA, or an entire server can all be recovered from via normal cluster failover mechanisms.

We can combine both clustering and multiple path options as follows:

FIGURE 41. High performance clustered configuration



In the figure above, 7140s A and C are accessing one set of private drives, and 7140s B and D are accessing a different, non-overlapping set of private drives on the SSA loop. Thus, each host has 2 paths to the SSA loop (using 2 initiators - the 7140s - on the SSA loop), and the hosts themselves are using clustering mechanisms for failover. This yields both the performance benefits of additional paths to additional 7140s, and the high availability benefits of cluster failover mechanisms.

Note: There can be no drives accessed by both 7140-C and D, or by 7140-A and B. If a drive were assigned to both (or say, public so available to all 7140s), then that drive would be seen twice by a host.

6.0 Tape Overview

Many of the issues one encounters with SAN-attached tape drives are similar to those encountered with disk, but there are some issues that are unique to tape. Because of this it is valuable to spend a little time looking at differences between tape and disk.

6.1 Differences between tape and disk

The key differences between tape and disk are

- All current tape drives from IBM are FC_AL only. (As of this writing, there was one vendor that had just announced a tape drive that could attach to a switch without using

Arbitrated Loop protocols. All other drives in the industry use FC_AL. This will change over time, but today it is an issue.) IBM tape drives cannot connect directly to a McDATA switch or director. They must connect to an IBM 2031-L00 (McDATA ES-1000) which is in turn connected to a McDATA switch or director.

IBM tape drives are public loop devices - they can log into a switch and participate fully on a fabric - but they must use Arbitrated Loop protocols on the link to the switch. This means that the switch they connect to must support FC_AL by providing an FL_Port. Not all combinations of drives and switches are necessarily supported, so use of the supported server URLs is crucial.(See “Appendix A - Supported Server URLs” on page77)

- Use of removable media means there is a difference between volumes and drives not found with disk. This is discussed more below.
- Tape drives are not direct access. To “skip around” on a tape volume you must move sequentially forward or backward. Furthermore, tape commands are relative to the current position and status of the tape volume, unlike disk commands that always specify the exact location on a disk that an I/O command is to use.

These last two differences seem relatively straightforward, and yet they have a very large effect on tapes are used with a SAN. Let’s look at this a little more closely.

6.1.1 Removable media

A disk LUN might be an actual physical drive or a logical one. But once a disk LUN is externalized onto a SAN, the volume and the drive are inseparable. You can erase a volume by formatting a drive, thus building a new volume, but you can’t bring the old volume back unless you had originally made a copy of it somewhere. With tape you can merely set a volume aside, use the drive for something else, use the volume on some other drive, and the bring that volume back to the original drive without ever having made a copy of the volume. There is no comparable set of actions with current disk subsystems.

This is important, because it allows for the construction of a tape library, in which you have some number of drives, a larger number of volumes that can be mounted on any of these drives, plus a media changer to move volumes between where they are stored when inactive and whatever tape drive might want to use them. So we have an entirely new device called a media changer that we must allow for with tape. There is nothing comparable with disks. With a disk subsystem, you merely define your LUNs and who can reach them. With a tape subsystem you must consider who will access which drives, which volumes in the library, and who will be able to control the media changer. As we look at various sample configurations, we will discuss both the media changer and the drives in question, and how to handle each.

6.1.2 Sequential access

The simple fact that tape is a sequential medium has surprisingly far-reaching implications.The most important aspect of this is that SCSI commands sent to tape drives refer to

locations that are not absolute but relative to where you are on the volume. For disk I/O, the commands always specify a specific track/sector to go to, and this is unambiguous regardless of when the command is received. However, if a tape gets a command to say, “skip to the next file” or “read the next block” where you end up is completely dependent upon where you started from.

This turns out to be a staggeringly important difference. The reason is, tape commands must arrive at the tape drive ***in the correct order***. Otherwise, something other than what is desired will happen. This is not a requirement with disk because the command will always specify that absolute location for the operation. This simple requirement has the following effects:

- Only one path to a tape drive can be in use at a time - concurrent use of multiple paths could easily cause commands/data to arrive out of order. Single-host multi-path access to tape is possible only in particular circumstances (discussed in Figure7.2.3 on page66)
- Only one application (and thus one server) can be using a tape drive at a time - interleaving commands to tape from different applications is a disaster. If you are running the same application on different servers, that might be a way that tape access could be interleaved - as the application instances could, in principle, work with one another to ensure the right I/O is being done at the right time - but this will not be discussed in this paper.
- “Normal clustering” mechanisms are typically not used with tape. In principle, there is no reason you could not have a server in a cluster pick up usage of tape drives from a failed server. However, the new server would not know anything about the status of the drive and volume and could not continue an application that had started. The application could be restarted from the beginning, but now you are using manual intervention rather than something automatic, and you can do this just as easily without clustering software.

  Tape sharing is done by the applications themselves (e.g. Tivoli Storage Manager) rather than by OS software. Typically, applications which share tape drives put them in a pool that the applications from various servers can get to. In this case, the drives are not “recovered” in any sense, they are merely put in use for a time, and then “returned” to a shared pool. An application that uses a shared drive does not “pick up” where some other operation has left off, but rather starts with a newly mounted volume.

  Now it is possible for an application running on a cluster to journal info about tape data, and thus some clever use of tape sharing could be done that way, but that is beyond the scope of this paper.
- Any given operation must only happen once. Consider a write operation. If you send the exact same write operation a second time to a disk drive, it merely re-writes the same data onto the same location. Everything is fine. If you send the exact same write operation as second time to a tape drive, the second write puts a second copy of the data right after the first one leaving you with corrupt data.

Some of these issues - such as the sharing and pathing issues - affect configuration options and will be discussed in the samples later in this paper. There are also two issues that have to do with support, and these will be discussed next.

6.1.3 FCTAPE and the DDW (Dreaded Double Write)

When a server writes to a tape drive across a Fibre Channel SAN, the tape drive will write the data, and then send a completion response to the server. If that completion response is lost, and the server times out and merely re-drives the I/O, then the data is written twice, and corrupt data is stored with no one realizing that that happened. This never happens with normal SCSI because there is an electrical interlock on the SCSI bus that makes it impossible for the server to not receive an acknowledgement without the tape drive knowing that the server did not receive it. With Fibre Channel this may be unlikely, but it is certainly possible.

A set of standards called FCTAPE was created to address this. When a server that supports FCTAPE first contacts a tape drive, it will indicate that it has additional capability. A tape drive that supports FCTAPE will respond that it also has additional capability. Then if the server times out waiting for a completion response to a write command, rather than re-drive the I/O it will ask the tape drive for status. Basically it is asking “Where do you think we are in this transaction?” (A unit of work in Fibre Channel is called an Exchange, and the server will query the tape drive’s view of the status of a particular Exchange. Exchanges have unique IDs so the tape drive knows exactly which transaction is being queried.) If the tape drive sent the completion out, it can respond that it finished that Exchange, and then the server will ask it to re-transmit from a particular point. The server will receive a new completion response and all is well.

Since many different things can go wrong during all this, it is possible that things cannot be worked out that nicely. The tape drive won’t keep status on a “complete” Exchange forever, and thus may respond that it knows nothing about the queried Exchange. In this case, there is no way to recover, but the server will know that something is wrong and will abort. In this case the particular job will have to be started again from the beginning, but you will never have bad data written (unknowingly) as you would with a double write.

IBM has chosen to only test and support HBAs that support FCTAPE. FCTAPE specifies a specific subset of all possible options - thus improving the likelihood of inter-operability and thereby speeding up the testing process. As long as you stay with supported HBAs you can count on FCTAPE support (usually this has to be turned on when configuring the driver). All IBM fibre channel tape drives support FCTAPE.

6.1.4 Disk and tape on one HBA

Mixing disk and tape on a single SCSI bus was rarely done. The workloads are different, the SCSI profiles are different, and it rarely worked satisfactorily. In some instances, tape drives would use large blocks and “data stream” to maximize performance, tying up the SCSI bus for long periods of time, while disk drives with smaller block sizes appropriate for random I/O would get less than their fair share of the SCSI bus. In other instances, the

tape drives would have trouble getting access to the bus because disk drives would respond faster. It was generally accepted that you kept disk and tape on separate SCSI buses.

With Fibre Channel, it is possible - *in principle* - to do better. I/O can be multiplexed and small blocks can be interleaved in the middle of large blocks. There is no shared bus to keep a slow device from sending whenever it is ready. So it is certainly possible with Fibre Channel to have both disk and tape sending a receiving with little interference. (There is always the issue of having too much data for the bandwidth available, but this is true regardless of whether mixing device types or not.)

Unfortunately, to actually take advantage of these capabilities in Fibre Channel, an HBA driver would have to be carefully written to handle this. Sophisticated multithreading would be needed to interleave the I/O fairly.

Furthermore, older SCSI drivers were written using a single SCSI profile for the whole adapter. This was fine since disk and tape were rarely, if ever, mixed, so a single profile (one for disk or one for tape) could be used for the whole adapter and work fine. (Often this is not under the user's control, the driver picks a profile based on what devices it sees.) HBA drivers could certainly be written to allow different profiles to be used for different devices at the same time on a single adapter, but as yet this is not a widespread practice. Thus, if you do mix disk and tape on a single HBA, you end up either using a disk profile for both disk and tape, or a tape for both disk an tape. Some device will be using a non-optimal profile.

Finally, subtle problems have been discovered whose source has been narrowed down to interference between disk and tape traffic. Often these are intermittent problems that just crop up now and then, but when they do, devices go offline and other bad things happen.

For all of these reasons, IBM's official posture on mixing disk and tape on a single HBA is a little different than other configurations. IBM will support mixing disk and tape on a HBA, and the IBM Support Center will accept problems reported on these configurations. However, if the problem determination process reveals that the problem in question is caused by the mixing of the traffic, IBM may choose to tell the customer that the only fix is to separate the traffic. (This normally means installing additional HBA(s), which can be problematic if a server is short on slots for adapters.) Overtime, more and more will be learned about which combinations seem to work OK, and which combinations have specific problems that will require separation of the traffic.

In essence, the customer is assuming some risk in these configurations, and it is strongly recommended to avoid mixing disk and tape traffic on a single HBA whenever possible.

6.2 Tape Libraries and Media Changers

There is plenty of available documentation about tape libraries and how they work, and the ability to partition them and so on. This paper will not go into detailed discussion of

media changers and tape libraries. From a SAN standpoint the key thing to remember is that the media changer is a separate device that must be separately accessed.

NOTE: In some cases there can be more than one set of robotics to move volumes between drives and offline storage, but the separate robotics are not separately accessed. In this case the "media changer" that accepts library commands is really a "library controller" that routes library commands to the appropriate robotics. There is still one addressable entity to send library commands to, and this paper will refer to this entity as the media changer, regardless of the number of robotic arms actually manipulating media.

This paper will first discuss how media changers are used in a SCSI environment, and then look at examples in a SAN environment.

6.2.1 SCSI access to tape libraries

If you consider a generic library that contains some number of SCSI tape drives and a media changer/library controller (and tape volumes of course), the tape drives will have SCSI ports and be connected to SCSI buses. Each tape drive uses a SCSI Target ID and a LUN (LUN#0) on a SCSI bus. The media changer has multiple options for connectivity.

Some libraries use a separate SCSI connection to the media changer. In this case the media changer uses a separate SCSI Target ID. Some libraries allow internal connections or paths between the tape drives and the media changer. In this case the media changer is merely another LUN behind the tape drive's SCSI Target. These paths are configurable such that not all tape drives have one active, but it is possible to have multiple paths to the media changer this way. Multiple servers can share the library this way, even though the tape drives themselves may be dedicated to a particular server.

Finally, some libraries (e.g. IBM 3494) will use a LAN or serial connection to the media changer. Using LAN connections simplifies the connectivity issues with sharing a media changer.

6.2.2 Fibre Channel access to tape libraries

Fibre channel attachment for the drives in a library is straightforward - use FC ports instead of SCSI ports on the tape drives. But for the media changer the different options discussed above result in different approaches for Fibre Channel connection. These different approaches will be discussed in detail in various configurations later in this paper.

- **Media changer with separate SCSI connection** - thus far this kind of library only participates on a SAN through a SAN Data Gateway of some kind. (see text starting at "SAN Data Gateway Basics" on page29 for more information on this device.) As yet, there are no native FC ports for media changers.
- **Media changer with internal connection from tape drives** - In this case the fibre channel port of the tape drive is sufficient for connectivity for the media changer. With SCSI, the media changer is just another LUN behind the tape drive's SCSI Target. With fibre channel the media changer is another LUN behind the tape drive's FC port.

- **Media changer uses LAN connection** - In this case nothing new is needed for Fibre Channel. The same LAN connection to the media changer can still be used.

7.0 Native Fibre Channel Tape configurations

As of 1/1/02, the only tape drives for which there is native FC attach capability are 3590s - which can be standalone or housed in a 3494 library - and the 3584 LTO library for which you can get FC drives or SCSI drives. The 3583 has a feature to allow FC connection, but this feature is a special type of SAN Data Gateway and will be covered in the section on SDGs and tape.

In the following examples, very little distinction is made between host platforms. However, to ensure support you must check the supported server sites for the different tape products. These URLs are listed in “Appendix A - Supported Server URLs” on page77.

7.1 3584 Native FC configurations

The 3584 is an LTO tape library that can have either SCSI tape drives or FC tape drives. The media changer is accessed via internal paths inside the library. For each tape drive the user configures whether the path is active or not. If it is active, then the media changer appears as an additional LUN behind the FC port of the tape drive. The drive will be LUN 0 and the media changer will be LUN 1 if the path is active.

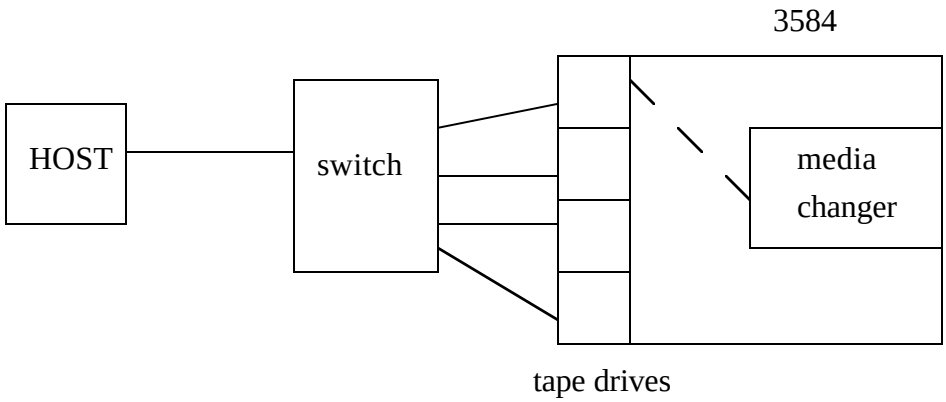
Supported configurations for the 3584 can be found at

http://www.storage.ibm.com/hardsoft/tape/3584/3584opn.html

7.1.1 Simple 3854 FC configurations

Let’s take a look at a simple 3584 configuration...

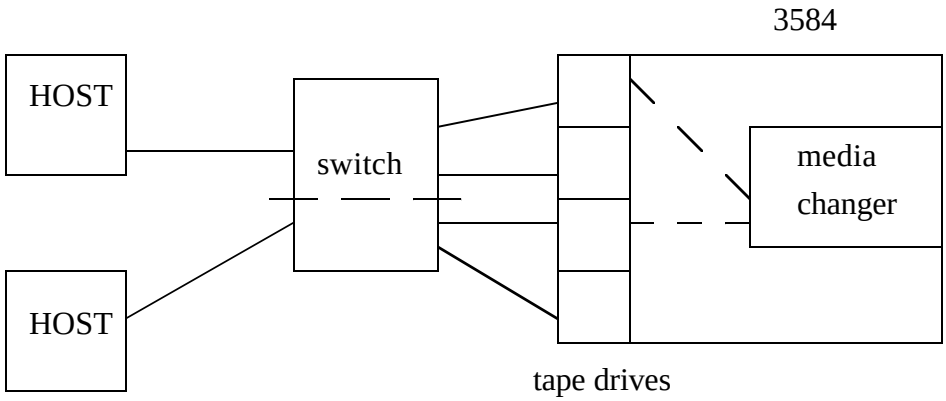
FIGURE 42. Single host 3584 configuration



In the figure above, a single host is connected via a switch to four drives in an 3584. (The 3584 can have many more drives than this, only 4 are shown to keep the picture simple.) Each tape drive is connected to the switch, and one of them has a path to the media changer configured. Now let's add another host....

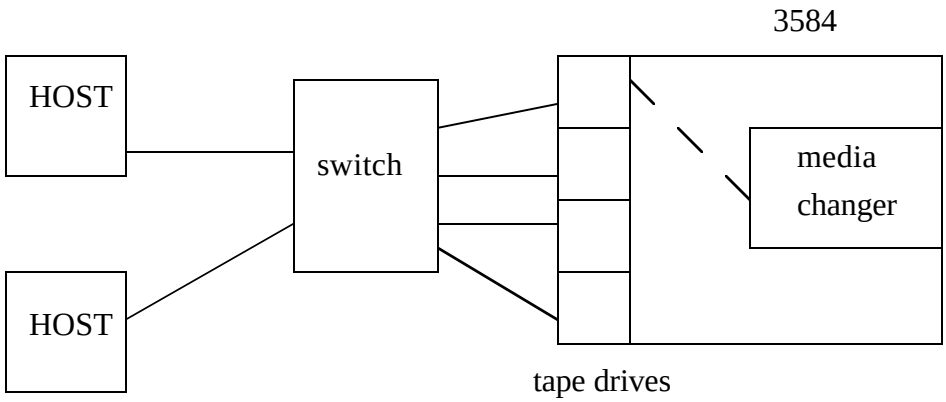
7.1.2 Shared library 3584 configurations

FIGURE 43. Multi-host 3584 configuration



The figure above shows a second host connected to the switch. The switch has been zoned such that the top host only reaches the top two tape drives, and the bottom host only reaches the bottom two tape drives. We have added a path to the media changer for the third tape drive, and thus both hosts can send library commands to the media changer, thus sharing the library. Concurrent access to media changers is not a problem. However, each host has its own two tape drives to use. If the switch had not been zoned, both hosts would see the media changer twice, and both hosts would see all four tape drives - not generally desirable. If hosts are going to actually share the drives themselves (an example of "shared access" as defined in Section2.2 on page 9), then some sort of software must be in place to handle this. Let's look at an example of this:

FIGURE 44. Tape sharing (pooling) with 3584



In the figure above there is no zoning in the switch. Both hosts can see all four drives, and the media changer. To make this work you must use an application that can control access to the drives. One such application is Tivoli Storage Manager (TSM), IBM's backup/ restore product. Other applications may also be capable of this.

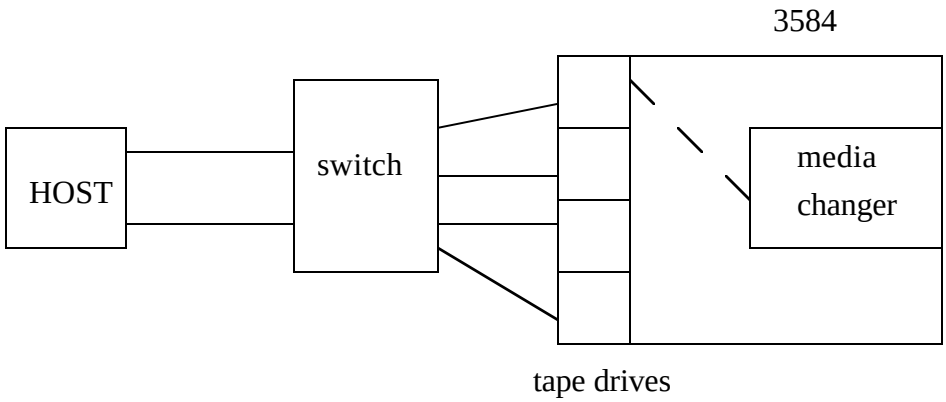
If the appropriate TSM product is installed on the two hosts, the tape drives can be shared even if the two hosts are different platforms. At least one of the clients would need to be a TSM server, the other could either be another TSM Server or a TSM Client capable of LAN-free backup. One host (TSM Server) will be the Library Manager, or "owner" of the tape drives, and the other host (TSM Server or LAN-free-capable Client) will be a Library Client and only use a tape drive if given permission from the Library Manager. Each server sees four tape drives and all four are assigned to the TSM application in each host. On the server running the Library Manager, the media changer is also given to TSM. Then the Library Client always asks for permission to use a tape drive, thus ensuring that only one host is using a tape drive at any given time.

Now suppose we put a second HBA into a single host...

7.1.3 multi-path to 3584

Multiple paths to a single tape drive from a single host are not supported for the 3584. There is no SDD or RDAC-like software for tape. However, multiple paths to a 3584 library are possible, as long as each tape drive is only on one path. Consider Figure45 on page62 below...

FIGURE 45. Invalid single host multi-path to 3584



In the figure above, the host can reach all four tape drives (and the media changer) out both HBAs. The host operating system will see this as 8 tape drives and 2 media changers. This could be managed by only assigning half of the discovered devices to whatever application is using them, or you can zone the switch such that any one tape drive is only reachable via one HBA.

The key point here is that there is not really a high availability option for the 3584 (or tape in general.) 3584s only have a single FC port, so that is a single point of failure anyway. Also, if a host has multiple paths to a tape drive, it will see it as multiple tape drives. Thus,

all but one of these paths must be disabled somehow. This can be done by not assigning the “duplicate” drives to any application, or by zoning the SAN. Either way it is done, a failure requires manual intervention to recover. Either new devices (representing the other paths to the drives) must be assigned to the application (along with any application configuration needed for these “new” devices), or the SAN must be rezoned after a failure. In the zoning case, after rezoning around a failure, the server must find the “new” devices, then you must assign them to the application, and then again do whatever application work is needed. The advantage of using zoning (and all that extra work) is that you don’t have to keep track of which device names are which in the server, you always assign every tape to TSM. Without zoning the user has to figure out what is an appropriate set of devices representing unique drives to assign to the application.

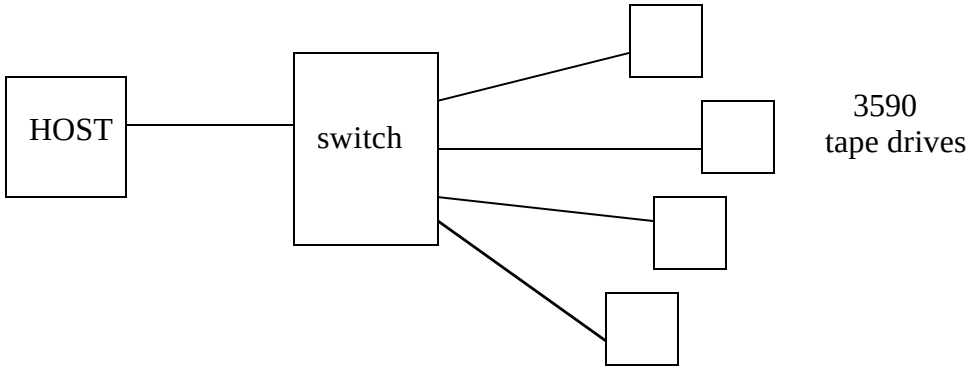
7.2 3590 Native FC configurations

The 3590 is a unique drive in that its FC adapter has 2 FC ports. When used with the 3494 library, a LAN connection is used to access the media changer. (A serial connection is also possible but not often used.) Thus the configurations in this section will only be concerned with access to the drives themselves, as access to the media changer is not a SAN issue. Supported platforms and code levels can be found at

http://www.storage.ibm.com/hardsoft/tape/3590/3590opn.html

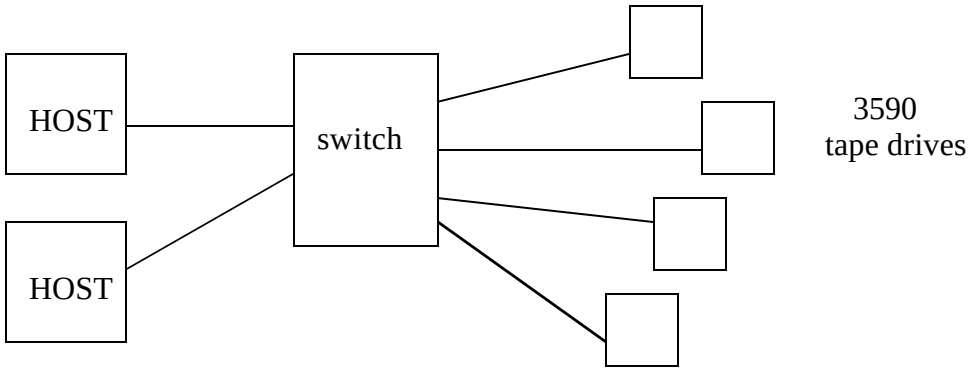
7.2.1 Single-port 3590 configurations

FIGURE 46. Simple 3590 connection (one-port)



In the figure above, the four 3590s are connected to a switch. The “SAN picture” is the same whether the 3590s are standalone or in a 3494 library, as the media changer in a 3494 is reached via LAN connection. Now let’s add another host....

FIGURE 47. Multiple hosts, one 3590 port



In the figure above multiple options are available to handle the different hosts. One option is to zone the switch (similar to Figure43 on page61) such that each host has exclusive access to its tape drives. If the switch is not zoned there are multiple options as follows:

- If the hosts are running an application - such as TSM - that allows tape pooling via a Master/Slave arrangement, this function will ensure that only one host uses the tape at the same time. (This is described in the text following Figure44 on page61.)
- 3590s honor SCSI Reserve/Release protocols. The IBM-supplied device driver will issue SCSI Reserves and Releases on behalf of an application.If you use applications that support the use of these protocols, then the drives may be shared because the Reserve/Release logic will ensure that only one host uses the drive at a time. TSM Servers from Release 3.7 and later support using SCSI Reserve/Release to share 3590 tape drives. Basically, every time a TSM Server is going to use a tape drive, the device driver issues a SCSI Reserve against that drive. If TSM in another server issues an I/O request, the device driver in that host issues a SCSI Reserve against an already reserved drive. It will get back a response indicating that the drive is reserved and will thus not use the drive. Then when the application is done with the drive it issues a SCSI Release so that it may be reserved by TSM running on a different host.

  For Figure47 on page64, if both hosts are TSM Servers they can share the drives (and the 3594 library - if there - via the LAN port) simply by using SCSI Reserve/Release. If a TSM server tries to reserve a 3590 and finds it already in use, it will merely try the next drive in its configured tape pool. This way, two TSM Servers can share a group of 3590s on a first come first server basis.

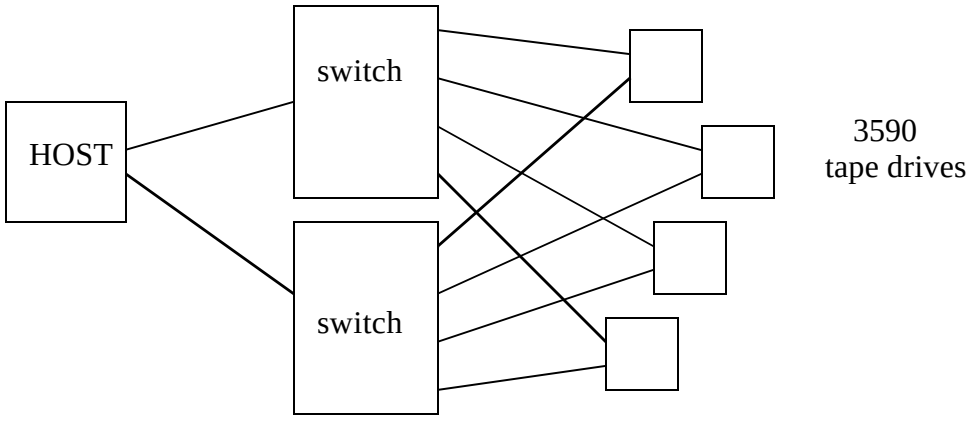
NOTE: There is work to do in TSM to optimize this kind of access. For instance, the default retention period for a tape volume that is no longer in use is 30 minutes. This is useful in the case that the same tape volume is needed again soon - you can avoid the unmount/remount. However, the drive is unavailable during this retention period, so if sharing tapes as in Figure47 on page64, the parameter should be much lower. Such TSM-specific issues are beyond the scope of this paper.

7.2.2 Dual-port 3590 Configurations with one host

The 3590 FC adapter has two separate ports that can be used independently. There is no performance benefit this way, as only one port is ever in use at a time, and the tape drive cannot fill up the one 100MBs port anyway. But the extra port can be combined with an extra path to either a different server or different HBA to facilitate sharing, and in some cases, high availability. There is a lot to account for in these kinds of configurations, and one cannot assume that alternate paths will automatically be used even though they are physically available. (More on this later.)

By default, if a SCSI Reserve is received and accepted on one port, the other port is unavailable until the Release is received *on that same port*. There is one exception to this which will be discussed a little later. But first let's look at few configurations.

FIGURE 48. Single host use of 2 3590 ports



In the figure above we are using two HBAs, two switches, and two ports on each tape drive to provide a complete extra physical path between the host and the tape drives. This results in the same tape drives being seen out each HBA, with the same considerations as discussed in the text following Figure45 on page62. However, there are a few new wrinkles here. Let's look at the various options to make this work...

- Zoning could be used in the switches such that each tape drive is only seen by a single HBA. In this case the host would only see one copy of each tape. By splitting the drives between the two HBAs some performance benefit can be gained, especially if you are using more than the four drives shown in the figure. However, should there be a failure in the SAN - either a link or a switch or a switch port or an HBA, then the user will have to manually rezone the switches to allow all tapes to one HBA, do whatever is necessary at the server for it to see what will appear to be new tape drives - an then assign those new drives to whatever application is using the tape drives.
- SCSI Reserve/Release can be used! In this case, the OS thinks it has 8 tape drives instead of 4 (4 out each HBA), but if the application supports SCSI Reserve/Release then it will only use one path at a time, as the tape drive representing the alternate path of an in-use tape drive will show as unavailable.

Suppose the server is a TSM Server that sees 8 tape drives. You could assign all 8 drives to TSM in a tape pool. As TSM goes looking for drives to use, if some of them show up as unavailable, it will continue looking for available drives in the pool. Should a failure occur, from the OS point of view half of the tape drives disappear, but the others are there to be used.

Note: You would want to configure TSM such that it never tries to use more tape drives than actually exist. (The OS will show more available than actually exist.) Also, if there is a path failure on a drive with an outstanding Reserve, the only way to clear that Reserve is to manually reset the drive.

- If the host is AIX, it is possible to use OS-provided multi-path access to the 3590s. This is only available for AIX connected to native FC-attached 3590s. In this case the host "knows" that there are four drives with two paths each and will handle everything appropriately. This is described in detail in the next section.

7.2.3 3590 multi-pathing with AIX

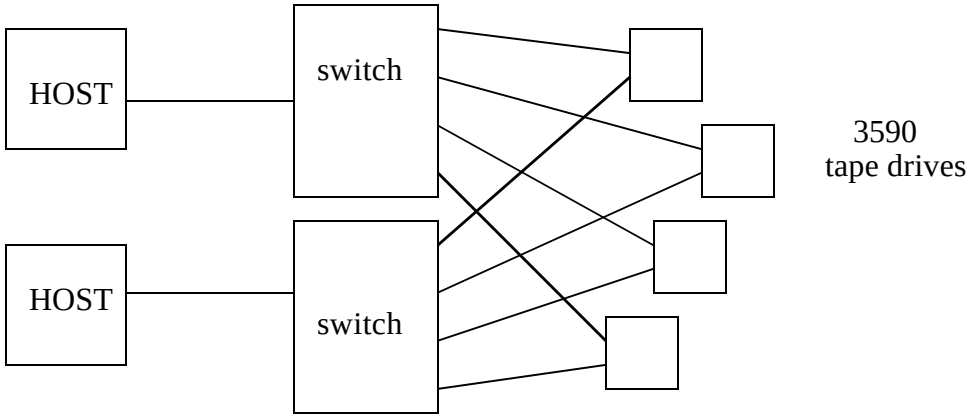
If the host in Figure48 on page65 was AIX then this host would discover 8 tape drives (four out each HBA) and name them rmt0, rmt1,...rmt7. By default they would be treated as 8 separate tape drives as with any other platform (or any other tapes on AIX.) However, a command can be issued to have the atape device driver go out and discover which rmt#s are duplicates of which others, and assign one as Primary and one as Alternate. (There are commands to display this relationship, and other commands to change which rmt# is used as Primary.)

You need only assign one rmt# to an application (best to use the Primary), and the application will have access to both paths - both the primary and alternate rmt# because the device driver is keeping track of all of this on behalf of the application. When an application tries to access any rmt device in a group that has been discovered to be the same drive, the device driver will always use the Primary rmt# if available, and if not it will use an available alternate. (Although our figure only shows two paths, it is possible to have more, and the device driver can keep up with as many as 16 paths to a single drive.)

The device driver is also keeping track of where the tape is positioned, and other tape status information, such that in a failure of one path, the cutover to the alternate path is transparent to the application. In this kind of configuration, AIX can even recover from a tape drive that had an outstanding SCSI Reserve on the failing path.

7.2.4 Dual port 3590 Configurations with multiple hosts

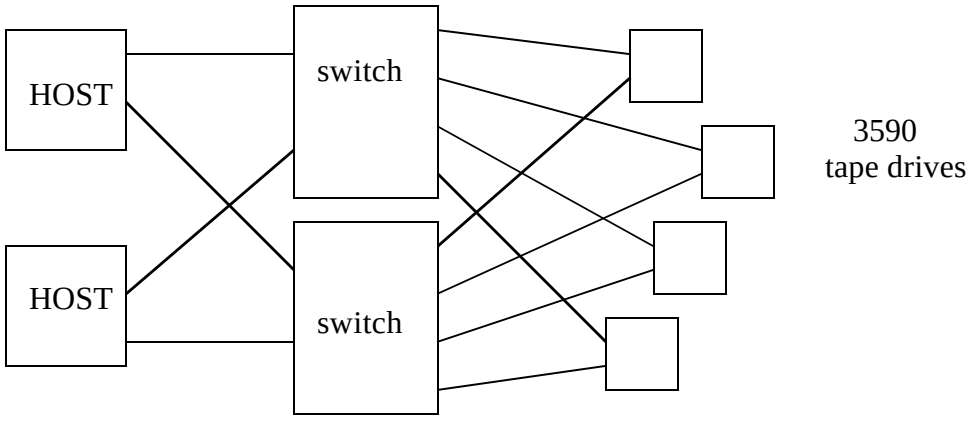
FIGURE 49. 3590 sharing using two ports



In the figure above each host only sees one copy of each drive, as it only has one path to each drive. If the two switches were connected together with an Inter-Switch Link, then each host would access the drives through each drive port, and the same “duplicate drive” issues as discussed for Figure45 on page62 would exist. With the switches separate, the hosts only see one copy of each drive, but each host can get to each drive, so some kind of “shared access” software is required. Either applications using SCSI Reserve/Release, or applications that pool tapes can be used. (TSM can do both.)

Now let’s combine sharing and multi-pathing...

FIGURE 50. Dual port 3590s with multiple paths and multiple hosts



In the figure above, both hosts have two paths to each drive. Thus, the considerations for multi-path and for tape sharing are all in play. There are multiple possibilities here, and the tape sharing issues (between hosts) can be considered separately from the issues of multi-pathing in a single host.

The issue of sharing between hosts does not change. Either the application using the drives must use SCSI Reserve/Release, or there must be some communication between them

such that only one drive is used, such as a Master/Slave relationship described earlier. TSM can do both of these methods for 3590s.

The issue of multi-pathing must be considered for each host. If a host is AIX, the multi-pathing can be handled directly by the device driver. If a host is not AIX, the multi-pathing must be handled either by zoning to limit access, assigning to the application only those OS devices representing unique drives, or via SCSI Reserve/Release.

Thus, use of SCSI Reserve/Release can satisfy both requirements. If the two hosts in Figure 50 on page67 are TSM servers, as long as neither host tries to use more than 4 tape drives at a time this first-come first-served approach can work. If one of the hosts is a LAN-free TSM Client there will be a Master/Slave Relationship between the two TSM applications.

8.0 Tape configurations using the San Data Gateway

8.1 Overview of SDG for tape

A description of the 2108-G07 can be found starting in Section4.3.1 on page29 and continuing through Section4.3.5 on page33 ending on page34. The 2108-R03 and a feature of the 3583 work much the same, and all can be used to connect SCSI tape drives to a SAN. The three different SDGs all have slightly different feeds, speeds and functions. The basic mechanism, however, is the same as that of the 2108-G07 except for the following differences:

- The 2108-G07 has 4 68-pin HVD UltraSCSI ports, can have up to 6 FC ports, and allows LUN masking via VPS. VPS also allows different host types to access the SDG through the same FC port on the 2108. The 2108-G07 has a maximum throughput around 110-120MB/sec. (The G07 is also the only SDG model that supports ESS. All others are tape only.) The G07 can have up to 255 attached LUNs.
- The 2108-R03 has either 2 68-pin HVD UltraSCSI ports, or 2 68-pin LVD UltraSCSI ports (LVD ports can also be used for Single Ended SCSI devices). This model can only use one FC port, and does not support VPS. Thus, all hosts using a given R03 must be the same host-type. The maximum throughput of a 2108-R03 is about 55-60MB/sec. It can have 127 attached LUNs.
- The 3583 LTO Tape Library has an orderable feature (8005) which is an SDG with 4 VHDCI LVD UltraSCSI ports, two FC ports (2Gb-capable), and does not support VPS. Thus this SDG can support two different host-types, one on each port. The maximum throughput of this device is about 140-150MB/sec when using both FC ports. It is dedicated to a specific 3583's drives and media changer.

NOTE: Older versions of the SDG microcode allowed you to set the host-type for an FC-port to "switch." This did *not* allow you to mix host-types on the port, and was the source of some confusion. This option was removed from later versions of code. The only sup-

ported way to mix host-types on a single SDG FC port is with VPS, and this is only available on the G07 model.

All three models can be used with tape. Supported devices and servers are different for each SDG. For the 2108-G07 and R03 the supported server lists can be found at

http://www.storage.ibm.com/hardsoft/products/sangateway/supserver.htm

For the 3583 integrated gateway, the support information is at the 3583 site at

http://www.storage.ibm.com/hardsoft/tape/3583/3583opn.html

There are two other important items relating to SDG support of tape. In Section6.1.3 on page57 we discuss FCTAPE support and IBM's requiring support of same. FCTAPE does not apply to SCSI attached tapes, however, from the point of view of the SAN, the SDG represents one or more FC-attached tapes, and it is desirable for it to support FCTAPE in the same way that our tape drives do. With current releases of microcode the SDG does support FCTAPE on behalf of the tape drives behind it.

Finally, the SDG also supports SCSI Reserve/Release such that everything happens like you would expect it to. The SDG will not allow two hosts to have a Reserve on the same drive, and all response from the SDG to SCSI Reserve/Release logic will be the same as you would expect for actual FC-attached tape drives.

8.2 SDG LUN assignments for tape

SDG LUN assignments for tape work much the same as that for disk described in "Gateway mapping" on page31, with one new wrinkle. Tape drive LUNs are always even-numbered on the FC side of a SDG. The odd-numbered LUNs (the numbers just above the ones used for the tape drive) are reserved for a media changer. Thus, for a 3584 library, the tape LUNs might be numbers 2,4,6, and 8 while LUNs numbered 3,5,7, and 9 are reserved in case there are internal library connections to the media changer. (If the paths are not configured then that SDG LUN# - say 7 in our case - is not used.) For other libraries that use external connections to the media changer (e.g. 3570/3575 or 3583), the media changer is separate and will use some other LUN number. For instance, a 3583 with 6 SCSI drives (and media changer) all attached to an SDG (that is using LUN 0 as the control LUN) would have the media changer at LUN 1, and the drives at LUNs 2, 4, 6, 8, 10 and 12. LUN#s 3, 5, 7, 9, 11 and 13 would be reserved in the SDG but never used.

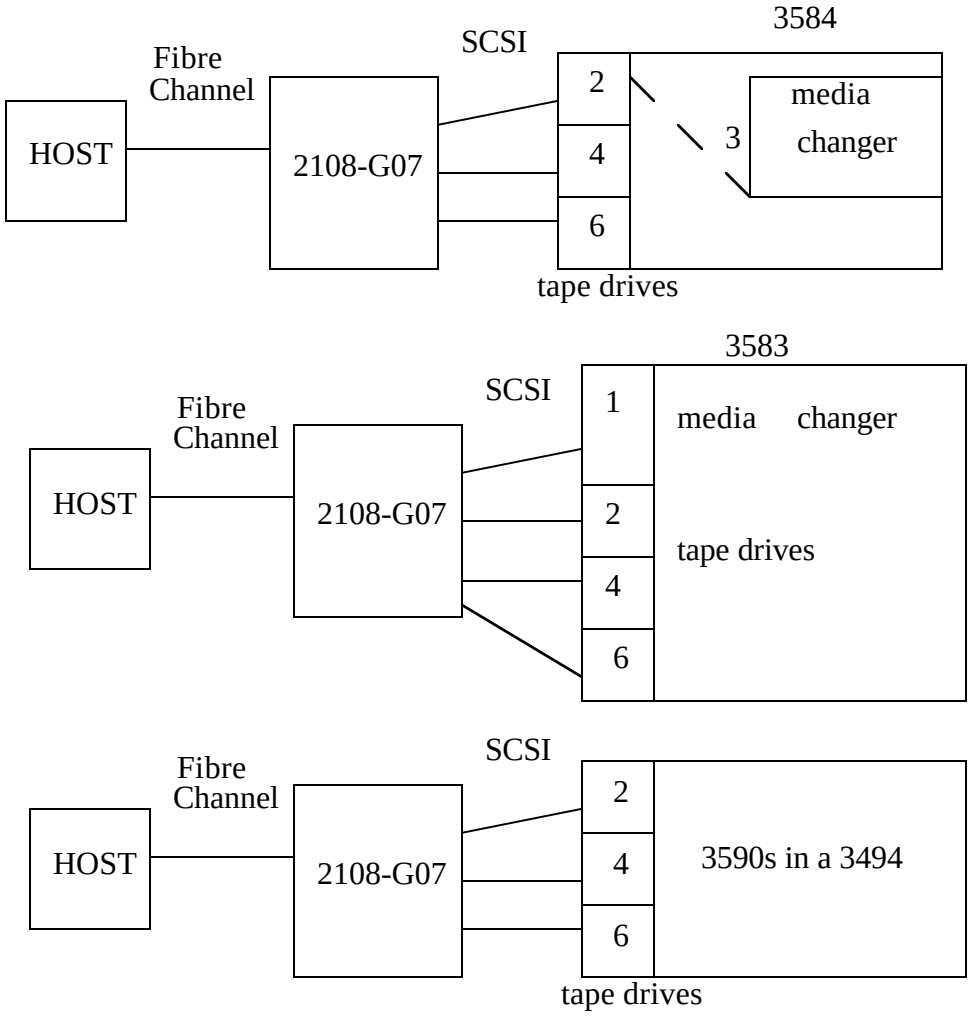
8.3 Tape Configurations using the 2108-G07

Because the workings of the different SDGs are so similar, we will look at tape configurations for the 2108-G07 first, and then just discuss where things differ for the 2108-R03 and 3583 feature.

8.3.1 Tape Configurations with the 2108-G07

Let us first consider some very basic configurations.

FIGURE 51. Simple Tape configurations with 2108-G07

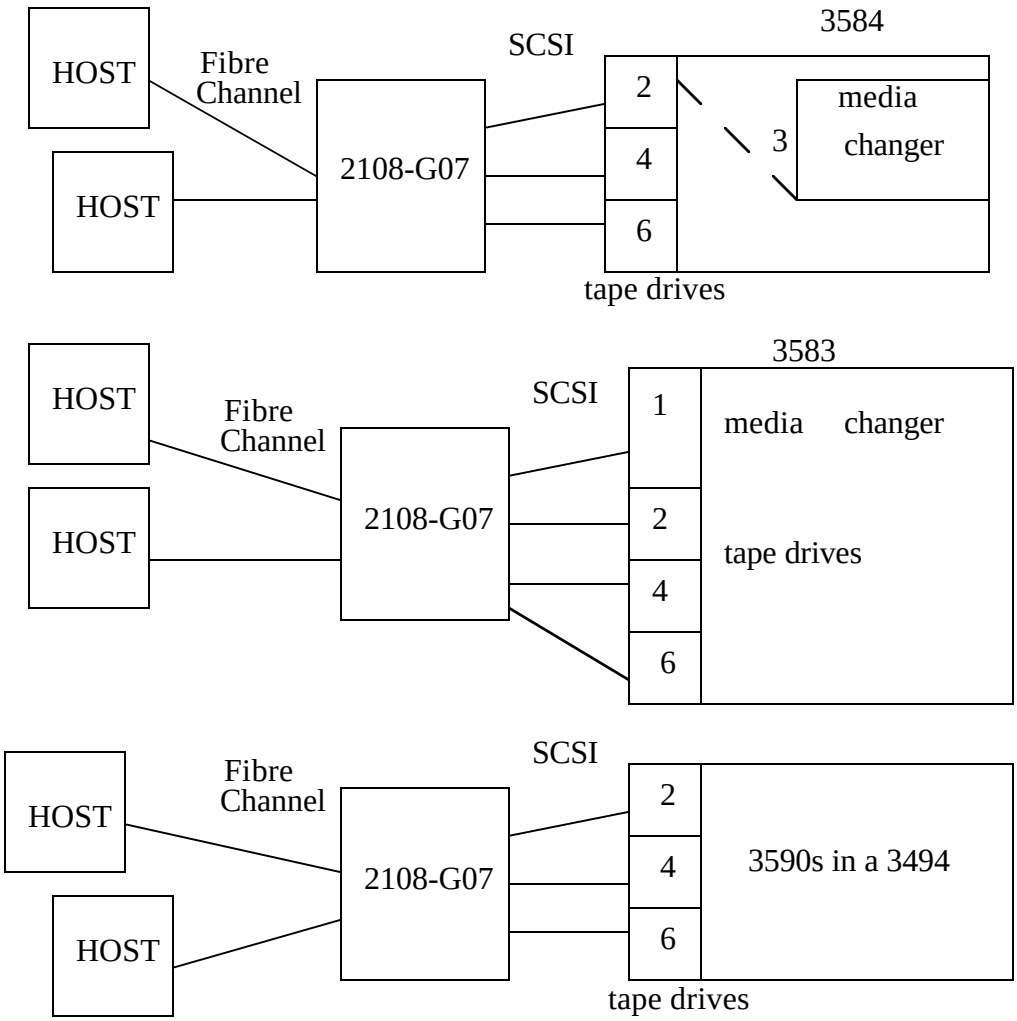


In Figure51 on page70 above, we show three examples of a host directly connected to a 2108-G07. The numbers associated with the tape drives and media changers are actually LUN#s that are used *on the Fibre Channel side of the SDG*, but I show them with the device so you can see which devices they are associated with. From a pure SCSI stand-point, the tape drives would be some SCSI Target ID and LUN#0 as discussed in Section6.2.1 on page59. Also, there could be a SAN in between the host and the SDG, but that would not change things. Since the G07 is HVD SCSI only, the 3584 and 3583 would have to be configured for HVD SCSI (3590 SCSI is HVD). Although the picture shows only one device on each SCSI bus, if there are more than 4 devices to be connected they can be daisy-chained on the SCSI buses just as with any SCSI bus. In general, since the SCSI buses run at 40MB/sec you usually don't want more than 2 tape drives on a single bus.

In this figure - assuming that LUN 0 is used for the SDG itself - the 3 tape drives in all three examples would use LUN#s 2, 4, and 6. Since the 3584 has an internal path to the media changer, it will use the reserved LUN number 3. LUN #3 is not used in the other two examples, much as LUN#s 5 and 7 are not used in any of the examples. With the 3583 the media changer takes the first available LUN which is LUN #1.The media changer for a 3590 library (3494) is not shown as it is reached via LAN.

If we add a second host...

FIGURE 52. Tape, 2108-G07, multiple hosts



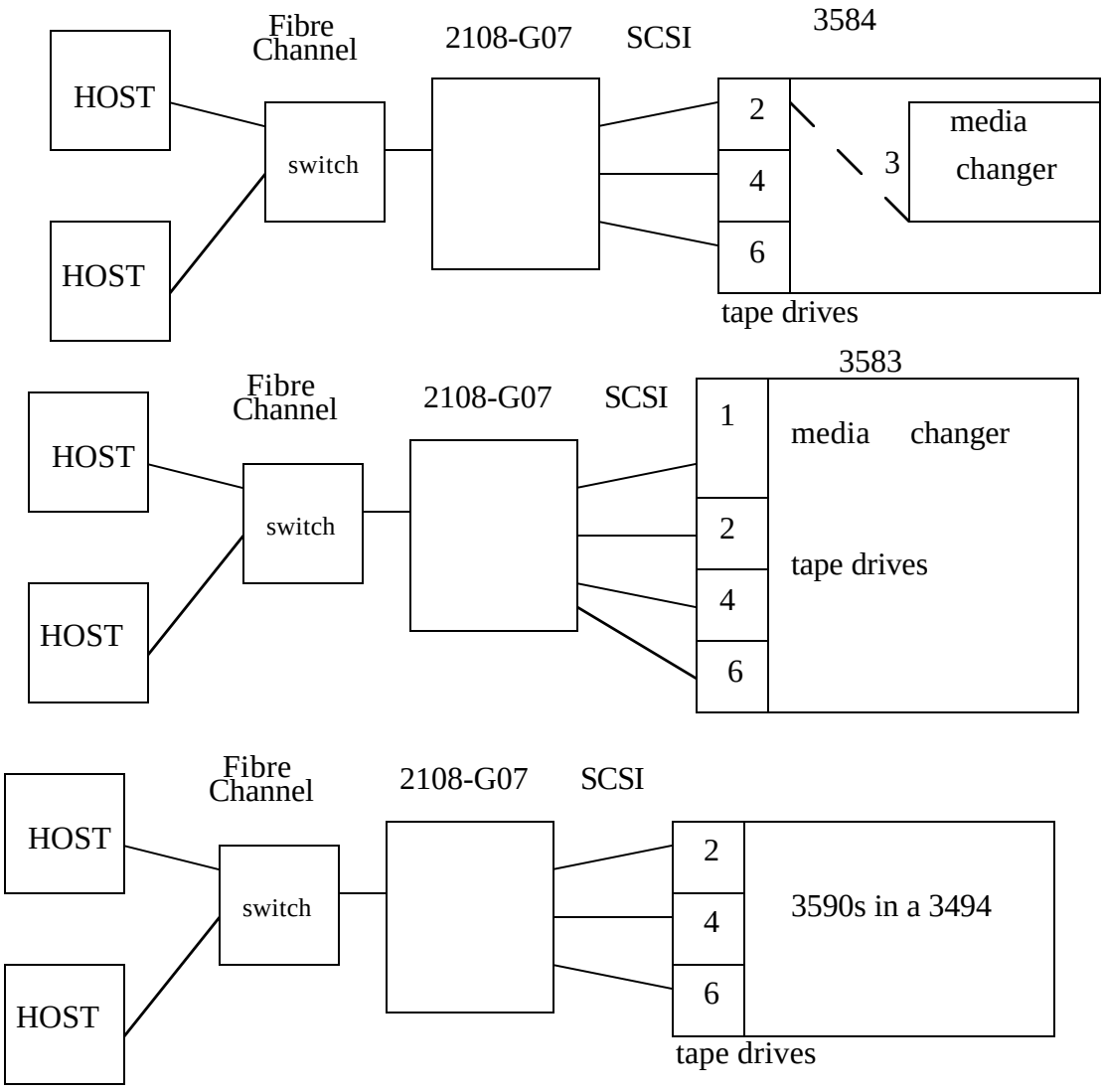
In Figure52 on page71 above, if no zoning (Section4.3.2 on page30) or LUN masking (Section4.3.4 on page33) is done in the SDG, then both hosts will see all three tape drives. For the 3584 and 3583 they will also both see the media changer. In these two examples, if you were using TSM in the hosts to share tape drives, you would assign the media changer to TSM in the host that was the Library Master, and leave the media changer unassigned in the host that had the Library Client or LAN-free Client.

LUN masking could be used to allow only one host to see the media changer. Zoning in the SDG could also accomplish this also, but for the 3584, the zone that includes the media changer also includes the top tape drive (LUN 2) since they are on the same SCSI bus. In other words - in the 3584 example - if the 2108-G07 were merely zoned, a given host would see either both the top tape drive and media changer, or neither. (Access would be to the top SCSI port of the SDG. (See “Zoning in the SDG” on page 30.)

Finally, these hosts could share the tape drives via the same methods as described in the previous section - Section7.0, “Native Fibre Channel Tape configurations,” on page60. Basically, either an application specifically written to share tapes (such as TSM), or SCSI Reserve/Release.

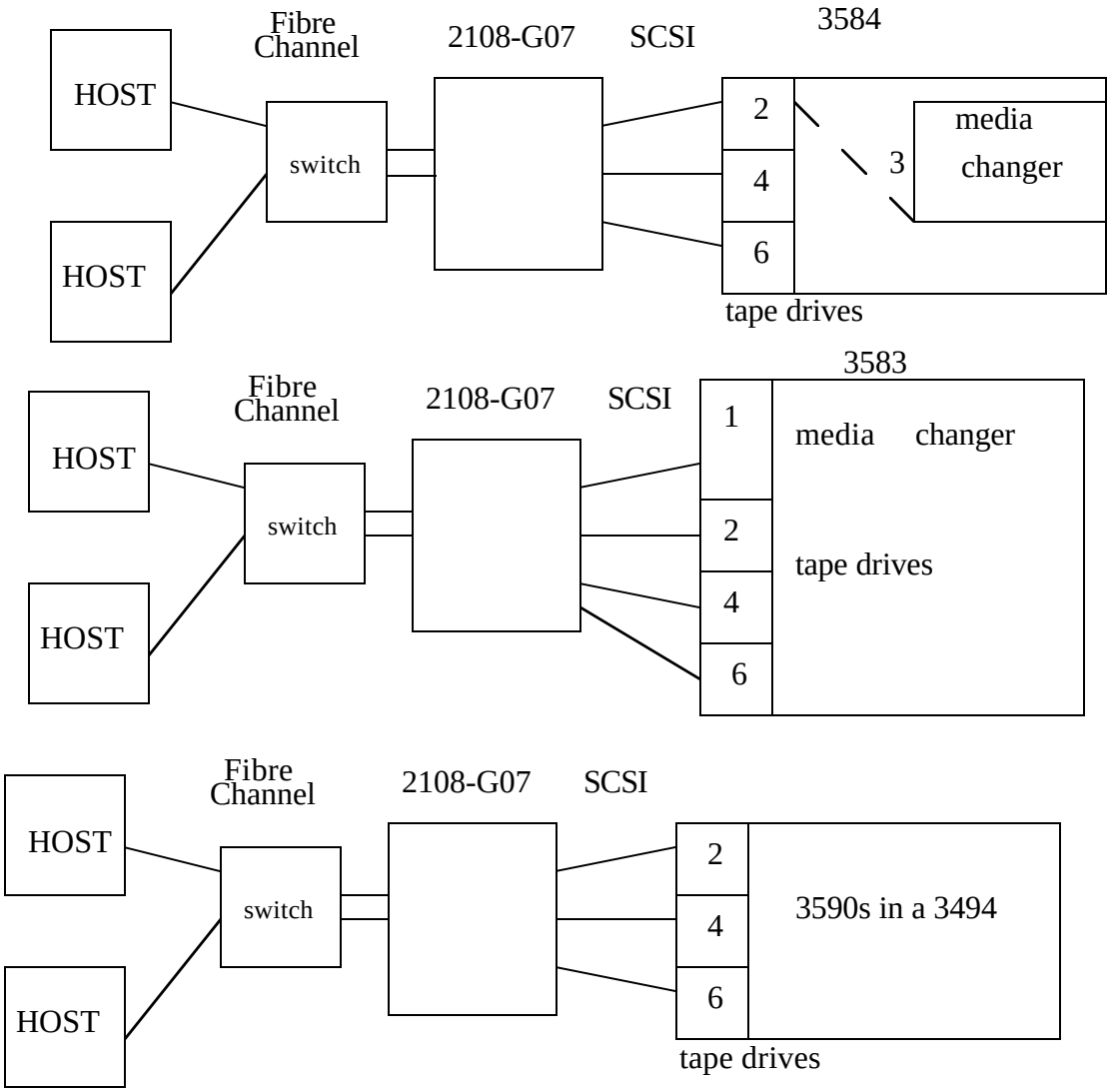
Let’s put a switch in the picture...

FIGURE 53. 3584, 2108-G07, two hosts, and a switch



The figure above works the same as Figure 53 on page72 except that zoning cannot be used to split the tape resources between the hosts. (Both hosts come in the same SDG FC port and thus will be in the same zone.) LUN masking could be used to separate them. If you are using LUN masking or zoning to dedicate tape drives in a library to different hosts, you need to ensure that the media changer is available to all hosts sharing the library. Also, if the two hosts were a different OS, this would represent a different host-type to the SDG, thus VPS would be needed to set host-type by host HBA/WWN. (see "**NOTE:**" in Section4.3.4 on page33) Now a small complication....

FIGURE 54. Tape, 2108-G07, two hosts, SAN



In the figure above we have added one more connection from each switch to each SDG. If the switches were zoned such that each host could only access one SDG port, this would be the same configuration as Section52 on page71. But if there is no zoning in the switches, then each host can get to *both* SDG ports, and will see all the drives (and the media changer for 3584/3) behind each one. This will appear to the hosts as 6 drives (and 2 media changers for 3854/3).

LUN masking in the SDG alone will not help here. A given host may be restricted to certain LUNs, but it will still see those LUNs behind both FC ports on the SDG, and thus see double LUNs. Zoning in the SDG would help! Zoning in the SDG could ensure that any given LUN is only available to one FC port, and thus eliminate the duplicates as seen by the hosts. Also, applications that support SCSI Reserve/Release could be used with the 3490s with “double tape drives.” By use of Reserve/Release they would avoid trying to use the same 3590 through different LUNs.

Finally, Figure54 on page74 also has the potential for both hosts to see the same LUNs. This kind be handled via LUN masking to keep things separate, or by using an application that supports tape sharing, or by using an application that supports use of SCSI Reserve/ Release(3590s only).

8.4 Tape configurations using the 2108-R03

8.4.1 2108-R03 differences

The 2108-R03 behaves almost exactly like the G07 with a few key differences:

- There is only one Fibre Channel port. Because all SDGs only allow 8 HBAs per FC port, this means that only 8 host adapters can use the R03 at the same time.
- There is no VPS, thus no LUN masking, and host-type is by FC port. That means all hosts using a 2108-R03 must be the same host type.
- There are only 2 SCSI buses. Combined with a max throughput of 55-60MB/sec means that you shouldn't have more than 4 tape drives behind an R03 (fewer if using compression at the tape drives.)
- R03 can support HVD, LVD and single ended SCSI, thus can be used with more different drive types. As always, supported-server matrices should be consulted to verify support.

Everything else is the same, including honoring SCSI Reserve/Release logic and providing FCTAPE support on the FC side.

8.5 Tape configurations using the 3583 Internal Gateway Module

8.5.1 3853 Feature 8005

The internal SDG for the 3583 (feature 8005) will behave just as the 2108-G07 in most cases. In the figures used in "Tape Configurations using the 2108-G07" on page69, the middle configuration in the figures uses the 3583 with a 2108-G07. Everything would be the same using the internal gateway with the following differences:

- Only the 3583 is supported, LVD SCSI is used. One feature 8005 can support only one 3583. Supported servers may be different than other SDGs, you must check the URLs.
- The 3583 has 2 FC ports only, meaning a total of 16 HBAs can use the SDG at the same time. These ports can run at 2 Gb.
- There is no VPS, thus no LUN masking, and host-type is by FC port. The 3583 can have at most 2 different host-types connected - one per port.

9.0 Summary

This paper has covered a lot of ground, however a few guidelines can be applied to every situation. Below is a summary of the salient points:

- Fibre Channel Protocol (FCP) is basically SCSI-3 over Fibre Channel. However, care must be taken to ensure that LUN#s presented by a Fibre Channel (SCSI) Target are the numbers the host expects.
- Multiple hosts with access to the same LUNs require software and/or configuration *in the hosts* to handle it.
- Disk LUNs can be shared via clustering mechanisms, tape LUNs usually are shared by function in a specific application - either the application uses a Master/Slave arrangement such that one server allocates tape drives to all the others, or the application supports use of SCSI Reserve/Release logic. Use of SCSI Reserve/Release requires that the tape drives themselves, and any gateway (e.g. SDG) in the middle also honor/support Reserve/Release. (IBM currently supports this on 3590s only.)
- If a single host can get to the same LUN via multiple paths (possible even with only one adapter in the server), software in the host must be there to handle that. (Not all configurations are supported.) For disk LUNs this is normally done by a disk-specific piece of software added to the host. (SDD for ESS, RDAC for FastT). For tape, only AIX with FC-attached 3590s supports multi-path. In this case the additional software is in the atape device driver itself. SCSI Reserve/Release can also be used with 3590s to manage multiple paths.
- For tape, IBM only supports HBAs that support the FCTAPE standard. All IBM FC tape drives as well as the SDG support this standard.
- Many configurations that *should* work are not yet supported. You must always check the URLs listed in Appendix A to verify support.
- Make good use of standard support vehicles such as Techline, ViewBlue Q+A, etc.

10.0 Appendix A - Supported Server URLs

For support matrix for the 2109 Fibre Channel Switch use

http://www.storage.ibm.com/hardsoft/products/fcswitch/supserver.htm

For support matrices for the 2108 SAN Data Gateway, there are multiple pages reachable from

http://www.storage.ibm.com/hardsoft/products/sangateway/supserver.htm

The support matrix for the FAStT200 is at

http://www.storage.ibm.com/hardsoft/products/fast200/supserver.htm

A full list of supported servers for the Enterprise Storage Server (ESS) is at

http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm

The support matrix for the 7140-160 is at

http://www.storage.ibm.com/hardsoft/products/san160/160_support_matrix.html

The support matrix for the 3854 LTO library is at

http://www.storage.ibm.com/hardsoft/tape/3584/3584opn.html

For the 3583 it is at

http://www.storage.ibm.com/hardsoft/tape/3583/3583opn.html

For the 3590 it is at

http://www.storage.ibm.com/hardsoft/tape/3590/3590opn.html

11.0 Appendix B - Fibre Channel Detail

11.1 Fiber Channel Addresses

There are multiple addresses in Fibre Channel networks. The ones used for getting data from source to destination are assigned by the network itself, and thus for a given server or storage these can change if you move that server or device. To be able to “track” a server or device if it is moved about a network, another address- called a World Wide Name - is used, and is associated with an adapter or device port, as does not change due to movement.

11.1.1 World Wide Names

Every Fibre Channel adapter - in a host or imbedded in a storage device - has an 8-byte number in it called a World Wide Name (WWN). They are globally unique. They are administered by a central authority that gives out blocks of addresses to vendors. It is then up to the vendor to ensure in their manufacturing process that the adapters have different addresses. (This is very similar to how Ethernet addresses are kept unique.) Typically, the first 5 bytes are assigned by the central authority, and then the vendor manages the 16 million possibilities in the last three bytes. (If this is not enough they can ask for another block of addresses.)

World Wide Names are not used for moving data around a SAN, but only for identifying ports. (You can also have a WWN for a node, such as one that represents the whole FCSS or whole ESS, but use of these is optional and not really needed.

WWNs are used primarily for zoning purposes. (see “SAN zoning” on page6) By using port WWNs for building zones I ensure that the zone stays the same no matter where I move a device or server.

11.1.2 Loop addresses

On any arbitrated loop, loop addresses are assigned by the Loop Initialization Process (LIP). These are one-byte addresses. There are 126 different non-zero addresses allowed. Address zero is reserved for a switch connected to a loop. Once all the addresses are assigned a list of addresses of all members of the loop is passed around the loop, thus every loop device can learn the loop addresses of everyone else on the loop.

11.1.3 Fabric addresses

In a switched fabric, a 3-byte address is used. The first byte is the switch number, the second byte is the port on that switch, and the third byte is a device number. If there is a loop attached to the switch port, the third byte is the loop address described above. (All the devices on that loop have the same first two bytes in their fabric address.)

All non-loop, fabric-attached servers or storage devices use 3-byte addresses that are assigned to them by the fabric via a Fabric Login process (FLOGI). FLOGI also registers a Fibre Channel adapter's WWN with the switch (used for zoning - see "Soft Zoning" bullet in "SAN zoning" on page6).Fabric-attached servers/devices then have the option of querying the fabric to find out who else is on the SAN. (SCSI Initiators will query the fabric while SCSI Targets typically just wait to hear from an Initiator.) The fabric will return all the "appropriate" 3-byte addresses - ensuring that only addresses in the requestor's zone(s) are returned - and then the server can start to contact these devices.

Loop devices do not necessarily perform FLOGI to get a 3-byte address. *Public* loop devices are "fabric aware" and *will* use FLOGI to get a full 3-byte address from the fabric and can then query the fabric to find other devices anywhere in the fabric. Private loop devices only use the loop address - the last byte - and never use a 3-byte address. Indeed, Private loop devices are not even aware that the fabric is there, they never issue FLOGI, and thus can only reach other devices on the same loop. The 2109 can provide additional capabilities for Private loop devices, as explained below.

11.1.4 Loop to fabric conversations

In the 2109 switch default configuration, if a fabric-aware server asks the fabric for possible devices to talk to, the fabric will return the full 3-byte addresses of all loop devices that are attached to the fabric - both Public and Private loop devices. When the fabric-attached server then opens a conversation (sends) to a Private loop device, the 2109 will map the 3-byte address of the fabric-aware server to an unused loop address on that device's loop, thus fooling the Private loop device into thinking it is talking on a loop rather than across a fabric. This is called Standard Translative Mode and comes with the 2109 as part of the base software.

Note that this only works if the SCSI Initiator - the "calling party" - is fabric aware and the SCSI Target - the "receiving party" is the Private loop device. If a SCSI Initiator is a private loop device (HP servers for example), then it will never query the fabric for attached devices, it will only talk to devices on its own loop using the one-byte loop address, and Standard Translative Mode does not get a chance to work. For Private loop servers like HP, the Quickloop function - software available via RPQ for the 2109 - is required for any "off-loop" conversations.

11.2 Quickloop

The 2109 switch with the Quickloop RPQ (8S0521) can make two or more loops that are attached to it behave as a single loop, by controlling the allowed loop addresses to keep them all unique, and then switching between the ports on behalf of the loop devices. Quickloop is the fundamental principle behind the Netfinity Managed hub as described in "The Netfinity Managed Hub" on page82 below.

To build a Quickloop on a 2109, all you do is define two or more ports as Quickloop ports. Then, as these ports engage in loop initialization, the switch ensures that all loop addresses on all the ports are unique - just as they would be on a single loop. The switch then

switches between the ports as the devices try to reach the various loop addresses. This is important because some servers' HBAs - HP is a prime example - cannot use the full 3-byte fabric addresses, and only use the one-byte loop address. In this case, a Private loop server such as HP (only uses the one-byte loop address) could talk to devices elsewhere in the fabric, as long as their attachment to the fabric is to a port also defined as part of the same Quickloop. Any device attached to a port defined as Quickloop must use loop arbitrations protocol, and cannot login to or query the switch fabric, that is, it is treated as a Private loop device regardless of its capability. (FLOGI attempts are ignored so the device becomes in essence a Private loop device.)

Quickloops are only logically the same loop. They are made up of multiple physical "loop-lets" that all share the same 126 addresses. From a pure Fibre Channel standpoint these are different loops, and a LIP on one loop-let does not affect any other loop-let. The 2109 manages the addresses - and the Loop Initialization Process (LIP) that assigns them - to ensure that the addresses already in use on the other loop-lets do not get used in the new LIP. These already-used addresses are presented as "already taken" to the loop-let that is initializing.

Also, because you are switching between loop-lets. It is possible to have multiple conversations going on simultaneously within one Quickloop, but no more than one at a time per loop-let.

The following restrictions apply to Quickloop function:

- A given Quickloop (one set of 126 loop addresses spread across multiple switch ports) can only span two switches in a fabric. (There can be multiple different Quickloops in a fabric, but there is never communication between different Quickloops.)
- A given switch can only participate in one Quickloop. That Quickloop can be zoned such that it behaves like separate loops, but only one set of 126 addresses is used.
- Servers (SCSI Initiators) attached to Quickloop ports (switch ports that are defined as Quickloop), can only talk to storage devices (SCSI Targets) that are on the same Quickloop. (If the Quickloop is zoned then only those in the same zone.) This is because these servers cannot query the fabric for other devices, they only "know" about devices on their own loop - the Quickloop in this case.
- Servers (SCSI Initiators) that are not attached to Quickloop ports, and are "fabric aware" (can login to the fabric and query it for devices), can reach storage devices (SCSI Targets) on a Quickloop. They use normal Standard Translative Mode to do this, as explained in "Loop to fabric conversations" on page80.
- Zoning comes in two flavors. I can chop up a Quickloop by port - in which case the ports can also be part of a larger fabric zone - or I can chop it up by loop address, in which case it is only a Quickloop zone. Zoning by loop address does not affect how the rest of the fabric accesses the Quickloop, and is useful only in limited situations beyond the scope of this paper. Zoning by WWN - as described in "Soft Zoning" on page6 - is not available for Quickloop devices since these devices never do fabric login (FLOGI) and thus don't register their WWN with the fabric.

11.2.1 The Netfinity Managed Hub

This 8-port managed hub is actually a 2109-like switch with Quickloop turned on for all 8 ports, and all other high level switch function removed (to keep the price down.) The price point is closer to that of a hub than that of a switch, with the intent of connecting only one server or storage device into each port. From a pure Fibre Channel Arbitrated Loop standpoint there are 8 distinct loops (loop-lets), 8 separate LIP procedures, and 8 separate Loop Arbitrations going on simultaneously. This allows switched rather than shared bandwidth - multiple conversations between different pairs of devices can go on simultaneously - but it does not eliminate the penalty for Loop Initialization described in “Hubs and Loop Initialization” on page5. If a device is booted or inserted on the logical loop (comes up on a port of the Managed Hub), the LIP is propagated to all ports in the same loop zone. If the loop zones overlap, all ports in the overlapping zones will also have LIP driven. This is because the only way for a private loop device to know who is on the loop is via the LIP process. (At the end of the LIP process, a bitmap is sent around that shows everyone on the loop.) Thus, in order to find out that someone has joined a loop, a LIP must be performed for everyone such that they re-receive the bitmap.

A managed hub can be extended to one other managed hub creating one logical loop and/or it can be attached to a 2109 switch. But when attached to a 2109 if must adhere to all the Quickloop restrictions. Suppose for instance, that a Managed hub is connected to a 2109 that does *not* have the Quickloop RPQ (or does not have Quickloop turned on any of its ports). Then hosts on the Managed Hub can only talk to storage devices on that hub because those are the only devices on the same Quickloop. However, Fabric-aware servers on non-Quickloop ports could still get to storage devices on the Managed Hub through Standard Translative Mode.

NOTE: The above example is the one instance in which the Managed Hub has less function than the 2103 (see Note on page2). Public loop devices on a 2103 can login to the fabric and reach storage devices attached to the fabric, however, the 2103 has only one conversation going at once.

There is limited zoning with the IBM Managed Hub. You can chop up a Quickloop by switch port or loop address as described above in Section11.2, “Quickloop,” on page80.

11.3 Fabric Address Notification (FAN)

Fabric Address Notification is an additional set of protocols in the Fibre Channel standard that allows public loop devices and SAN fabrics to re-drive I/O after a LIP rather than starting over from scratch. A discussion of “normal” LIP is in order first. Note that FAN only applies to *public* loop devices attached to fabrics. Private devices, or public devices on loops unattached to a fabric (a private loop) cannot take advantage of this function.

11.3.1 LIP basics

For loops attached to switches, the switch port basically runs the LIP process. (There is an election for a temporary Loop Master that is always won by the switch port.) After the

election process, the switch (as Loop Master) sends a bit-map around the loop for devices to choose a loop address. Address zero is shown as already taken in this bit-map, since the switch will use that address. Each device in turn will have a chance to choose an address.

The bit map is sent out four times. Each time, the addresses picked in the previous rounds are shown as already taken. The first time is intended for public devices that had been previously logged into a fabric (via FLOGI as described earlier in this appendix). These devices will attempt to get the same address they had before. The second time is for all other devices previously active to try and get the same address as before. The third time is for devices with "hard" addresses. This is normally a configuration option on a device, meaning that the address configured is required by the device. While the intent was for devices to not participate on the loop if they did not receive that particular hard address, some devices may still be willing to accept a different address even if they can't get the configured hard address. (Although soft addressing should be fine in most circumstances, some platform/HBA combinations require hard addressing.) The final round for the bit map is for "soft" addressing. Meaning if you couldn't get the address you wanted in the first three rounds, you take what you can get on the last round.

The important point to remember here is that these are all best-effort attempts at getting the address a device wants. A switch is guaranteed of getting address zero, but no other device can be sure, because as devices move from loop to loop it is always possible for some device to have access to an address first, taking a loop address that some other device "wanted."

For instance, if device A had gotten loop address 2 before, and device B is a new station on the loop (causing the LIP) and takes address 2 before A does (perhaps B was address 2 on a previous loop), now device A no longer has address 2. The switch cannot send I/O destined for address 2 as it will go to the wrong place. Devices A and B cannot send I/O destined for address 0 (the switch) since they may have been moved to a new fabric, there is no way to tell. What happens is that all devices flush I/O buffers and start from scratch. Public devices re-drive FLOGI and start all over. Thus, the LIP is completely disruptive, much like a SCSI bus reset.

11.3.2 FAN flows

If both a switch and a device support Fabric Address Notification something extra can happen that will allow I/O to be re-driven instead of flushed. In essence, devices that support FAN keep track of their previous status - their full 3-byte address and the switch and fabric they are attached to (all provided by FLOGI response from the switch.). Then, when a LIP occurs, if the switch supports FAN, it will send its information in a special FAN frame to all previous addresses that had done Fabric Login. If those devices are the same as before the LIP, and verify that the fabric/switch information is the same as before, the device will send a FAN Accept to the switch, and I/O can be sent along knowing that it will go to the same place. No new FLOGI is done in this case.

If the FAN device sees anything different than before, it will reject the FAN, re-FLOGI, and all I/O is flushed. If the device does not support FAN it might reject the FAN frame

from the switch, or it might ignore it. In either case, it will re-drive FLOGI and this will flush all I/O as normal.

Note that this only works for FAN-capable devices that have previously FLOGId to a FAN-capable switch. Private devices cannot take advantage of FAN, nor can public devices that did not complete FLOGI. For instance, devices connected to Quickloop ports or a 3534 (see above) cannot use FAN since they never complete FLOGI.

IBM supports use of FAN with Fibre Channel 3590s and 3584 Fibre Channel drives attached to a 2031-L00 (McDATA ES-1000.) The 2031-L00 propagates LIP to all ports when a tape is brought online after maintenance (or a new tape is connected). FAN support allows this to happen without disrupting on-going I/O to other tape drives on the same ES-1000.

