Storage Basics

Oftentimes, storage isn't given enough attention in system architecture, but it can make or break the service level agreement (SLA) for your application response times. Understanding how to build a cost-effective, high-performance storage system can save you money not only in the storage subsystem, but in the rest of the system as well.

Storage is a huge topic, but this article will give you a high-level look at how it all fits together.

DAS, SAN, and NAS storage subsystems

Direct attached storage (DAS), storage area network (SAN), and network attached storage (NAS) are the three basic types of storage. DAS is the basic building block in a storage system, and it can be employed directly or indirectly when used inside SAN and NAS systems. NAS is the highest layer of storage and can be built on top of a SAN or DAS storage system. SAN is somewhere between a DAS and a NAS.

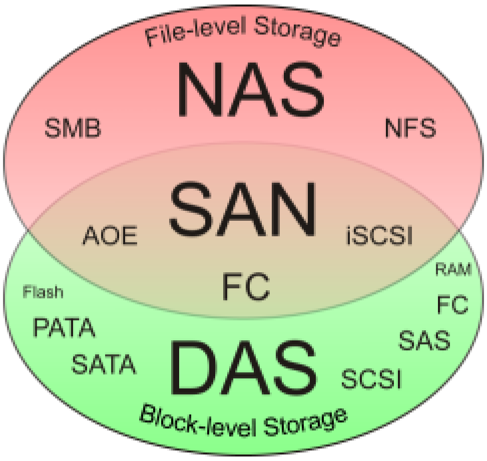

Figure 1 – Overview of storage systems

DAS (Direct Attached Storage)

DAS is the most basic storage subsystem that provides block-level storage, and it's the building block for SAN and NAS. A DAS system is directly attached to a server or a workstation, **without** a storage network in between. The performance of a SAN or NAS is ultimately dictated by the performance of the underlying DAS, and DAS will always offer the highest performance levels because it's directly connected to the host computer's storage interface. DAS is limited to a particular host and can't be used by any other computer unless it's presented to other computers over a specialized network called a SAN or a data network as a NAS server. A DAS controller allows max 4 servers to access the same logic storage unit. Protocols used for communication between computers/servers and DAS storage systems are FC, or SATA, or SCSI, or PATA, or SASA.

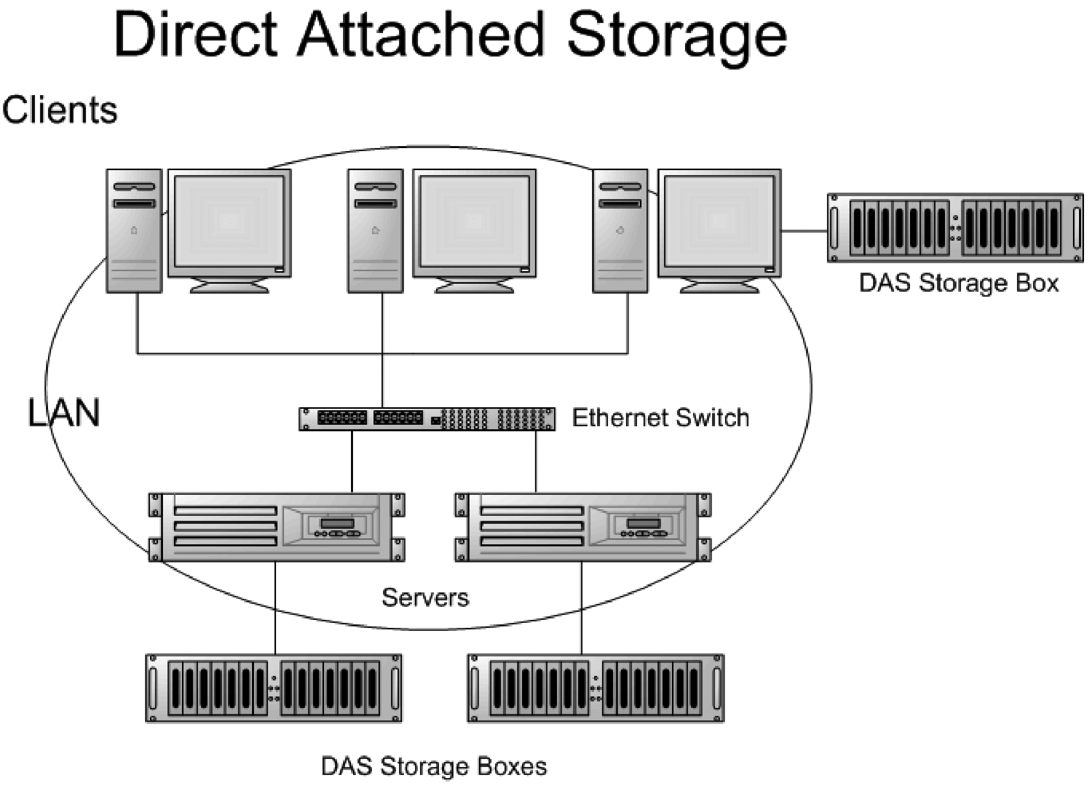

Figure 2 - Example 1 with DAS

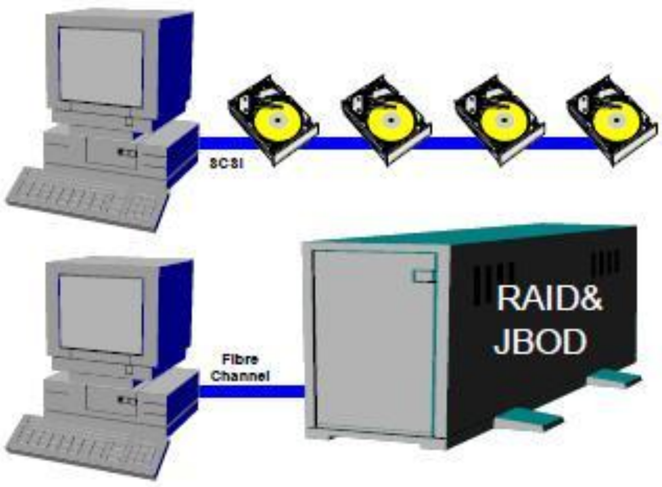

Figure 3 - Example 2 with DAS

The software layers of a DAS system are illustrated in Figure 4. The directly attached storage disk system is managed by the client operating system. Software applications access data via file I/O system calls into the *Operating System*. The file I/O system calls are handled by the *File System*, which manages the directory data structure and mapping from files to disk blocks in an abstract logical disk space. The *Volume Manager* manages the block resources that are located in one or more physical disks in the *Disk System* and maps the accesses to the logical disk block space to the physical volume/cylinder/sector address. The *Disk System Device Driver* ties the *Operating System* to the *Disk controller* or *Host Bus Adapter* hardware that is responsible for the transfer of commands and data between the client computer and the *disk system*. The file level I/O initiated by the client application is mapped into block level I/O transfers that occurred over the interface between the client computer and the disk system.

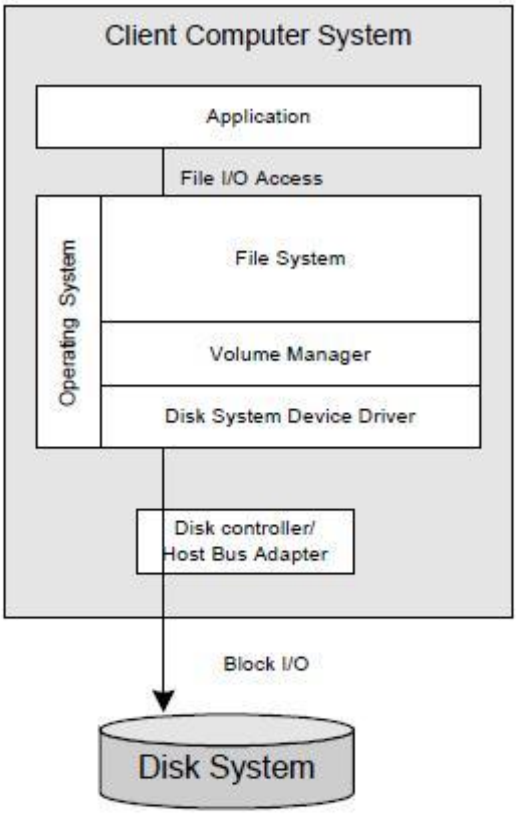

Figure 4 - DAS Software Architecture

Protocols used by a DAS storage subsystem

SCSI - Small computer system interface is one of the oldest forms of storage interfaces traditionally used in server or workstation class computers. It's been through many revisions, from SCSI-1 all the way up to Ultra-320 SCSI, which is the modern SCSI interface. (There is an Ultra-640 standard, but that isn't common.) The 320 and 640 numbers represent MB/s, megabytes per second. SCSI-1 started out 5 MB/s. SCSI is still used in modern servers, but the interface is starting to lose market share to SAS. Most recent versions of SCSI can handle up to 15 hard drives.

While the cable sharing mechanism is relatively efficient, there is a maximum theoretical cap of 320 MB/s, but that limit is reduced further by SCSI overhead. It's theoretically possible that 15 modern SCSI hard drives could have an aggregate throughput of 1350 MB/s, so they would be forced to share a 320 MB/s interface. But in the vast majority of applications, where there will inevitably be some random I/O in the hard drives, the mechanical latency of the hard drives seeking data means it's unlikely that an Ultra-320 interface will be fully saturated.

PATA - Parallel advanced technology attachment (originally called ATA and sometimes known as IDE or ATAPI) was the most dominant desktop computer storage interface from the late 1980s until recently, when the SATA interface took over. PATA hard drives are still being utilized today, especially in external hard drive boxes, but they're becoming rare. Some cheaper high-end server storage devices have also used PATA. Like SCSI, PATA has also gone through many revisions. The most recent version of PATA is UDMA/133 which supports a throughput of 133 MB/s.

Although PATA supports two devices per connector in a master/slave configuration, the performance penalty of sharing a PATA port is severe and not recommended if performance is important to the user. The 40-pin connector and cabling is also extremely wide, which is difficult to use in a high-density environment and tends to block proper airflow. The size of the connector also presents problems for smaller 2.5" hard drives, which require a special shrunken connector.

SATA - Serial advanced technology attachment is the official successor to PATA. So far, there have been two basic versions of SATA, with SATA-150 and SATA-300. The numbers 150 and 300 represent the number of MB/s that the interfaces support. SATA doesn't have any performance problems due to cable/port sharing, but that's because it doesn't permit sharing at all. One SATA port permits one device to connect to it. The downside is that it's much more expensive to buy an eight-port SATA controller than an Ultra-320 SCSI controller that allows 15 devices to connect to it. The upside is that each drive gets a theoretical 300 MB/s. Current SATA hard drives, however, barely get 80 MB/s, so the bus interface is a bit of overkill for now.

SATA uses a small seven-pin connector and a thin cable, which is more conducive to denser installations and airflow. That's important, especially inside a storage array with 15 hard drives, because you'll need one port and one cable for every drive, whereas SCSI lets you hook up one or two ports to the backplane that the drives attach to. SATA drives are used in smaller servers and some less expensive storage arrays.

SAS - Serial attached SCSI is the latest storage interface that's gaining dominance in the server and storage market. SAS can be seen as a merged SCSI and SATA interface, since it still uses SCSI commands yet it is pin-compatible with SATA. That means you can connect SAS hard drives or SATA hard drives or CD/DVD ROM or burner drives. SAS has a signaling rate of 185, 374, 750, and eventually, 1,500 MB/s. But storage controller technology has historically been rated by actual data throughput, which is lower than the signaling rate. To make these numbers comparable to the numbers listed above, the actual data rates are 150, 300, 600, and eventually, 1,200 MB/s. Note how the two lower data rates match up with SATA.

SAS connectors are keyed such that SATA devices can connect to SAS but SAS devices can't connect to SATA ports. The ports and cabling look similar, but SAS cables can be 8 meters long, whereas SATA cabling is limited to 1 meter. The longer cabling support is due to higher signal voltages, but the voltage is dropped to SATA levels whenever a SATA device is connected.

SAS is designed for the high-end server and storage market, whereas SATA is mainly intended for personal computers. Unlike SATA, SAS can be connected to multiple hard drives through expanders, but the protocol used to share a SAS port has lower overhead than SCSI. Coupled with the fact that the ports are faster to begin with, SAS offers the best of SCSI and SATA in addition to superior performance.

FC - Fibre channel is both a direct connect storage interface used on hard drives and a SAN technology. FC offers speeds of 100, 200, and 400 MB/s. Native FC interface hard drives are found in very high-end storage arrays used in SAN and NAS appliances, although the technology may ultimately give way to SAS.

Flash - Flash memory isn't a storage interface, but it is used for very high-end storage applications because it doesn't have the mechanical latency issues of hard drives. Flash memory can be packaged into the shape of a hard drive with any of the above interfaces so that it can be used in a storage array. The benefit of flash memory is that it can offer more than 100 times the read IOPS (input output per second) and 10 times the write IOPS performance of hard drives, which is extremely valuable to database applications.

The downside of flash memory is that it's very expensive per gigabyte (cost proportional to the performance advantage) and it has a limited number of writes and rewrites. Flash memory will begin to fail anywhere between 10,000 and 1,000,000 writes. To deal with this limitation, flash devices use a mechanism called *wear leveling* to spread out the damage so that the device will last longer, but even that has its limits.

Advantages

In a DAS system the storage resource is dedicated, and besides the solution is inexpensive.

Disadvantages

DAS has been referred to as "Islands of Information”. The disadvantages of DAS include its inability to share data or unused resources with other servers. Both NAS and SAN architectures attempt to address this, but introduce some new issues as well, such as higher initial cost, manageability, security, and contention for resources.

SAN (Storage Area Network)

NAS and SAN are two ways of sharing storage over the network. SANs offer a higher level of functionality than DAS because it permits multiple hosts (server computers) to attach to a single storage device at the block level. It does not permit simultaneous access to a single storage volume within the storage device, but it does allow one server to relinquish control of a volume and then another server to take over the volume. This is useful in a clustering environment, where a primary server might fail and a backup server has to take over and connect to the same storage volume. Because a SAN offers block-level storage to the host, it fools the application into believing it's using a DAS storage subsystem, which offers a lot of compatibility advantages. The SAN may use FC, or Ethernet (iSCSI or AoE) to provide connectivity between hosts and storage.

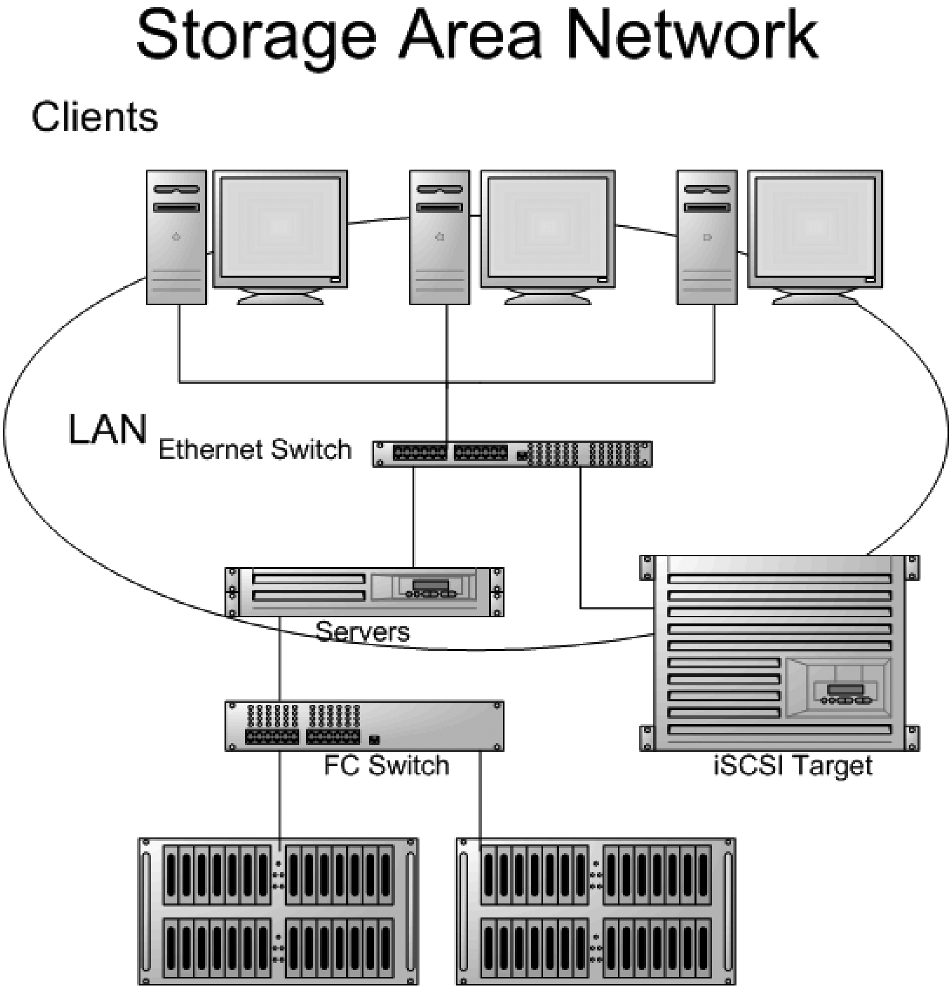

Figure 5 - Example with SAN

Figure 5 gives an example of a typical SAN network. The SAN is often built on a dedicated network fabric that is separated from the LAN network to ensure the latency-sensitive block I/O SAN traffic does not interfere with the traffic on the LAN network. This examples shows an dedicated SAN network connecting servers (application or database servers) on one side, and a number of disk systems and tape drive system on the other. The servers and the storage devices are connected together by the SAN as peers. The SAN fabric ensures a highly reliable, low latency delivery of traffic among the peers.

The SAN software architecture required on the computer systems (servers), shown in Figure 6, is essentially the same as the software architecture of a DAS system. The key difference here is that the disk controller driver is replaced by either the Fibre Channel protocol stack, or the iSCSI/TCP/IP stack that provides the transport function for block I/O commands to the remote disk system across the SAN network. Using Fibre Channel as an example, the block I/O SCSI commands are mapped into Fibre Channel frames at the FC-4 layer (FCP). The FC-2 and FC-1 layer provides the signaling and physical transport of the frames via the HBA driver and the HBA hardware. As the abstraction of storage resources is provided at the block level, the applications that access data at the block level can work in a SAN environment just as they would in a DAS environment. This property is a key benefit of the SAN model over the NAS, as some high-performance applications, such as database management systems, are designed to access data at the block level to improve their performance. Some database management systems even use proprietary file systems that are optimized for database applications. For such environments, it is difficult to use NAS as the storage solution because NAS provides only abstraction of network resources at the file system level for standard file systems that the Database Management System may not be compatible with. However, such applications have no difficulty migrating to a SAN model, where the proprietary file systems can live on top of the block level I/O supported by the SAN network. In the SAN storage model, the operating system views storage resources as SCSI devices. Therefore, the SAN infrastructure can directly replace Direct Attach Storage without significant change to the operating system.

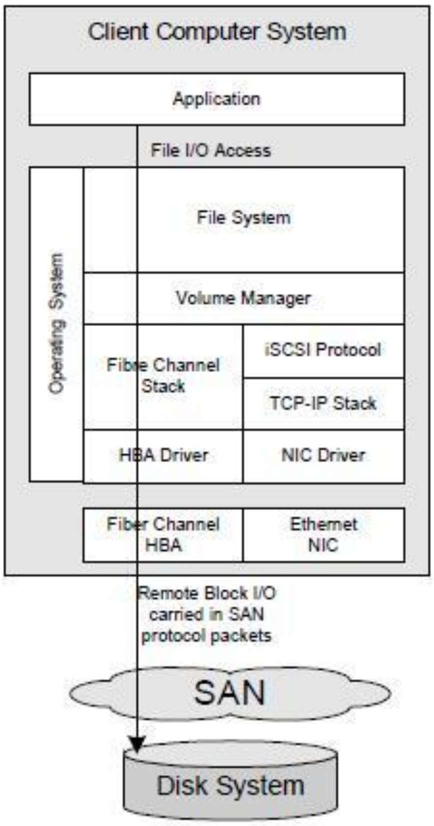

Figure 6 - SAN Software Architecture

SAN technologies

FC - Fibre channel is one of the older, established high-end forms of a SAN. It's common for FC SANs to use native FC hard drives, but they're not limited to it. There are FC SAN implementations that use SCSI or even ATA hard drives. FC SANs typically use 1, 2, or 4 gigabit fiber optic cabling, but less expensive copper cabling and interfaces are used for shorter distances.

FC storage arrays can be directly attached to a server. However, that defeats the ability to reconnect to other servers on the fly if one server fails, so they're typically attached via FC switches. The downside is that FC switches are very expensive per port, especially for the higher-end 4 gigabit variety. It's common for 16-port FC switches to cost tens of thousands of dollars. While the performance is high and the technology is well established, it requires a different knowledge set to manage an FC SAN.

iSCSI - Internet SCSI is a low-cost alternative to FC that's considered easier to manage and connect because it uses the common TCP/IP protocol and common Ethernet switches. Because any network engineer is familiar with TCP/IP and Ethernet switch configuration, and gigabit Ethernet adapters and switches are cheap, the cost advantages over FC SANs are compelling. A 16-port gigabit switch can be anywhere from 10 to 50 times cheaper than an FC switch and is far more familiar to a network engineer. Another benefit to iSCSI is that because it uses TCP/IP, it can be routed over different subnets, which means it can be used over a wide area network for data mirroring and disaster recovery.

Most iSCSI implementations use gigabit Ethernet 1000BASE-T, but speeds can be scaled to 10 gigabits per second with 10GBASE-CX4 and soon with the less expensive 10GBASE-T using twisted pair CAT-6 or CAT-7 copper cabling. It's possible to mix gigabit and 10 gigabit Ethernet such that a high-end storage array uses 10 gigabit Ethernet, but the multiple servers fed by the array connect to the switch using single gigabit Ethernet.

The downside to iSCSI is that it is computationally expensive for high storage throughput because it has to encapsulate the SCSI protocol into TCP packets. This means that it either incurs high CPU utilization (not much of a problem with modern multicore processors) or it requires an expensive network card with TOE (TCP offloading engine) capability in the hardware.

iSCSI targets (iSCSI servers -- the source of the storage) can come in the form of hardware storage arrays that speak the iSCSI protocol or they can come in the form of software added to a server. A server with iSCSI target software loaded is functionally the same as a hardware iSCSI target, but you can build it on top of any major server OS from BSD to Linux to Windows Server. There are open source Linux iSCSI targets and there is commercial iSCSI target software for Windows. Using a software solution allows you to serve a wide variety of devices as iSCSI targets that can be remotely mounted by iSCSI initiators (iSCSI clients) over TCP/IP. Hardware iSCSI targets are merely dedicated servers specifically designed to act as an iSCSI target, and they sometimes simultaneously behave as NAS devices. iSCSI initiator software is natively included in almost every operating system.

AoE - ATA over Ethernet is the most recent SAN technology to emerge, created as an even lower-cost alternative to iSCSI. AoE is a technology that encapsulates ATA commands into low-level Ethernet frames and avoids using TCP/IP. That means it doesn't incur CPU penalty or require high-end TOE-capable Ethernet adapters to support high storage throughput. This makes AoE a high-performance, very low-cost alternative to either FC or iSCSI. Its proponents also boast that the AoE specification fits onto eight pages, compared with the 257-page iSCSI specification.

Because AoE doesn't use TCP/IP, it isn't a routable technology -- but then again, neither are FC SANs. Most SAN implementations don't require routability, and the fact that you might use AoE on a particular initiator or target doesn't prohibit you from using iSCSI. A lot of add-on initiator/target software will support both iSCSI and AoE. Most WAN applications are low-bandwidth, so it won't incur a lot of CPU utilization anyway. This means you can use AoE for the high throughput LAN/SAN environment and use iSCSI for the WAN at the same time without TOE Ethernet adapters.

AoE software initiator support is now native in Linux and BSD, but it isn't natively included in Windows, and you'll have to purchase third-party initiators. Coraid, which is a major supporter/supplier of AoE, provided the original FreeBSD device drivers.

Advantages

Sharing storage usually simplifies storage administration and adds flexibility since cables and storage devices do not have to be physically moved to shift storage from one server to another. Other benefits include the ability to allow servers to boot from the SAN itself. This allows for a quick and easy replacement of faulty servers since the SAN can be reconfigured so that a replacement server can use the LUN of the faulty server. SANs also tend to enable more effective disaster recovery processes. A SAN could span a distant location containing a secondary storage array. This enables storage replication either implemented by disk array controllers, by server software, or by specialized SAN devices. Since IP WANs are often the least costly method of long-distance transport, the FCoIP and iSCSI protocols have been developed to allow SAN extension over IP networks. The traditional physical SCSI layer could only support a few meters of distance - not nearly enough to ensure business continuance in a disaster.

Disadvantages

SANs are very expensive as Fibre channel technology tends to be pricier, and maintenance requires a higher degree of skill. Leveraging of existing technology investments tends to be much difficult. Though SAN facilitates to make use of already existing legacy storage, lack of SAN-building skills has greatly diminished deployment of homegrown SANs. So currently pre-packaged SANs based on Fibre channel technology are being used among the enterprises. Management of SAN systems has proved to be a real tough one due to various reasons. Also for some, having a SAN storage facility seems to be wasteful one. At last, there are a few SAN product vendors due to its very high price and very few mega enterprises need SAN set up.

NAS (Network Attached Storage)

NAS is a file-level storage technology built on top of SAN or DAS technology. It's basically another name for "file server." NAS devices are usually just regular servers with stripped down operating systems that are dedicated to file serving. Although it may technically be possible to run other software on a NAS unit, it is not designed to be a general purpose server. For example, NAS units usually do not have a keyboard or display, and are controlled and configured over the network, often using a browser. A fully-featured operating system is not needed on a NAS device, so often a stripped-down operating system is used. For example, FreeNAS, an open source NAS solution designed for commodity PC hardware, is implemented as a stripped-down version of FreeBSD. NAS systems contain one or more hard disks, often arranged into logical, redundant storage containers or RAID arrays. NAS removes the responsibility of file serving from other servers on the network. NAS devices typically use SMB/CIFS for Microsoft compatibility, NFS for UNIX compatibility, or Samba for both. Many modern NAS appliances will support SAN technologies like iSCSI, and you can basically build the same hybrid storage solution using a general purpose operating system like Linux, BSD, or Windows using your own hardware.

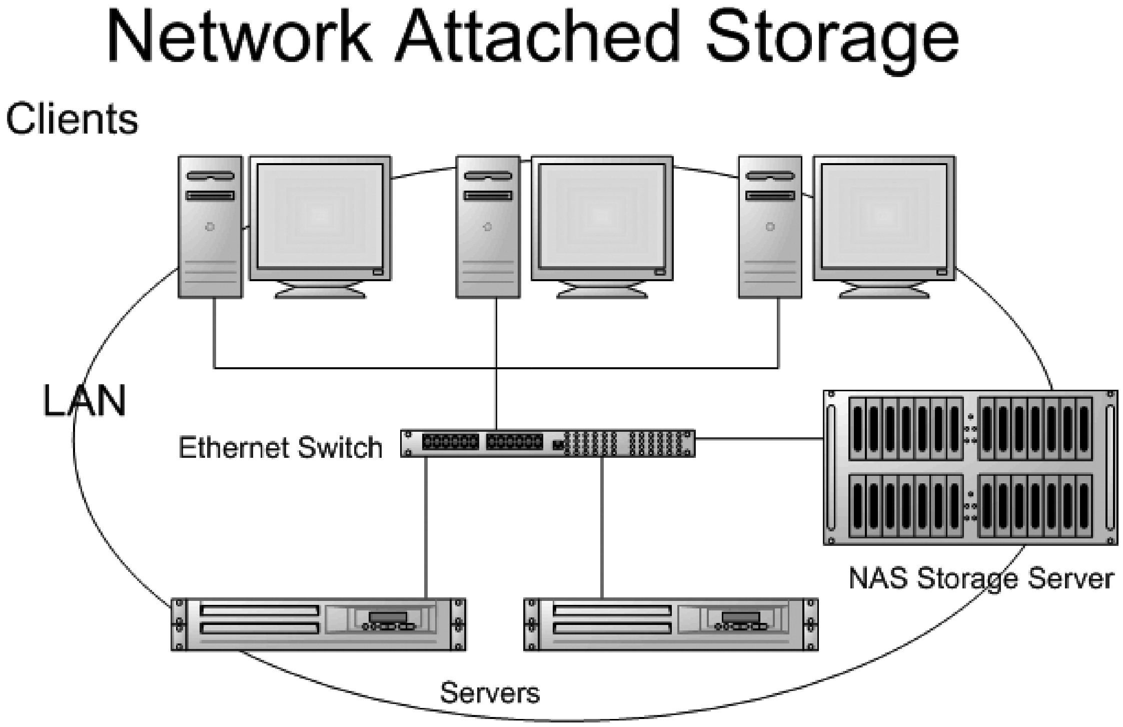

Figure 7 - Example with NAS

The difference between NAS and SAN is that NAS does “file-level I/O” while SAN does “blocklevel I/O” over the network. For practical reasons, the distinction between block level access and file level access is of little importance and can be easily dismissed as implementation details. Network file systems, after all, reside on disk blocks. A file access command referenced by either the file name or file handle is translated into a sequence of block access commands on the physical disks. The difference between NAS and SAN is in whether the data is transferred across the network to the recipient in blocks directly (SAN), or in a file data stream that was processed from the data blocks (NAS). As the file access model is built on a higher abstraction layer, it requires an extra layer of processing both in the host (file system redirector) computer, and in the function of translation between file accesses and block accesses in the NAS box. The NAS processing may result in extra overhead affecting the processing speed, or additional data transfer overhead across the network; both can be easily overcome as technology advances with Moore's law. The one overhead that cannot be eliminated is the extra processing latency, which has direct impact on the performance of I/O throughput in many applications. Block level access can achieve higher performance, as it does not require this extra layer of processing in the operating systems.

The benefit that comes with the higher layer abstraction in NAS is ease-of-use. Many operating systems, such as UNIX and LINUX, have embedded support for NAS protocols such as NFS. Later versions of Windows OS have also introduced support for the CIFS protocol. Setting up a NAS system, then, involves connecting the NAS storage system to the enterprise LAN (e.g. Ethernet) and configuring the OS on the workstations and servers to access the NAS filer. The many benefits of shared storage can then be easily realized in a familiar LAN environment without introducing a new network infrastructure or new switching devices. File-oriented access also makes it easy to implement a heterogeneous network across multiple computer operating system platforms. In this example, there are a number of computers and servers running a mixture of Windows and UNIX OS. The NAS device attaches directly to the LAN and provides shared storage resources.

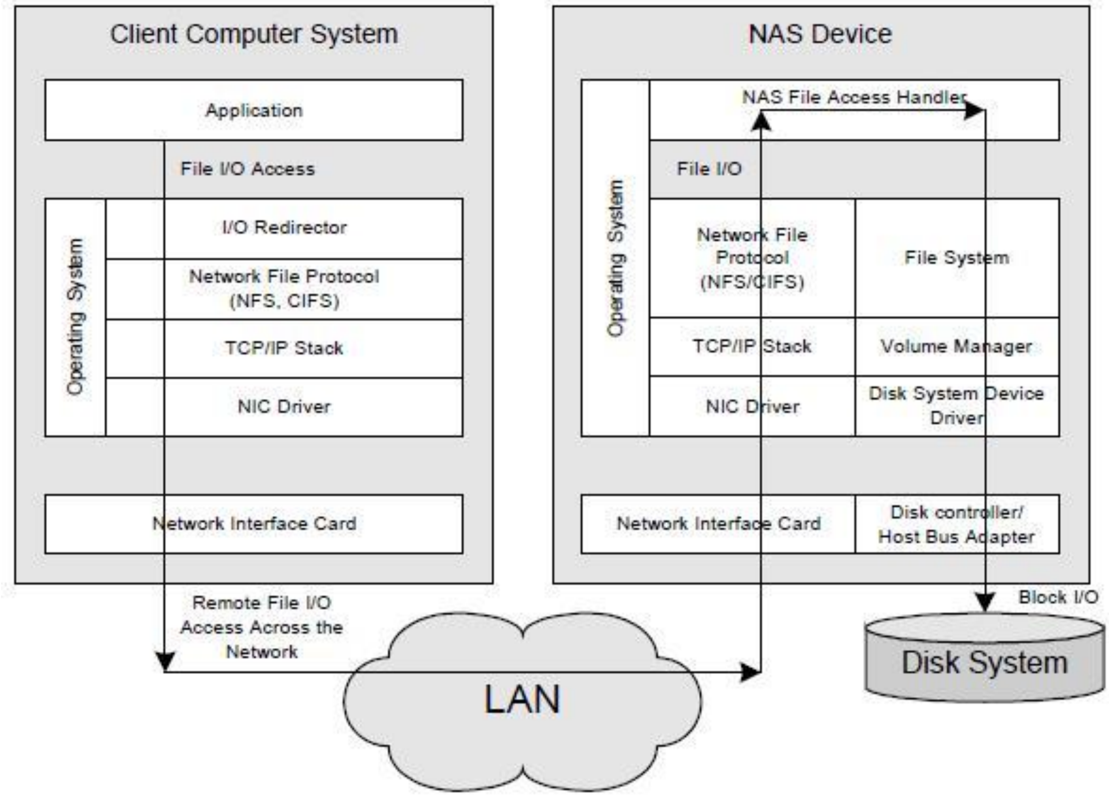

Figure 8 - NAS Software Architecture

The generic software architecture of NAS storage is illustrated in Figure 4. Logically, the NAS storage system involves two types of devices: the client computer systems, and the NAS devices. There can be multiple instances of each type in a NAS network. The NAS devices present storage resources onto the LAN network that are shared by the client computer systems attached to the LAN. The client *Application* accesses the virtual storage resource without knowledge of the whereabouts of the resource. In the client system, the application File I/O access requests are handled by the client *Operating System* in the form of systems calls, identical to the systems calls that would be generated in a DAS system. The difference is in how the systems calls are processed by the *Operating System*. The systems calls are intercepted by an *I/O redirector* layer that determines if the accessed data is part of the remote file system or the local attached file system. If the data is part of the DAS system, the systems calls are handled by the local file system. If the data is part of the remote file system, the file director passes the commands onto the *Network File System* Protocol stack that maps the file access system calls into command messages for accessing the remote file servers in the form of NFS or CIFS messages. These remote file access messages are then passed onto the TCP/IP protocol stack, which ensures reliable transport of the message across the network. The NIC driver ties the TCP/IP stack to the *Ethernet Network Interface card*. The *Ethernet NIC* provides the physical interface and media access control function to the LAN network.

In the NAS device, the Network Interface Card receives the Ethernet frames carrying the remote file access commands. The NIC driver presents the datagrams to the TCP/IP stack. The TCP/IP stack recovers the original NFS or CIFS messages sent by the client system. The NFS file access handler processes the remote file commands from the NFS/CIFS messages and maps the commands into file access system calls to file system of the NAS device. The NAS file system, the volume manager and disk system device driver operate in a similar way as the DAS file system, translating the file I/O commands into block I/O transfers between the *Disk Controller/HBA* and the *Disk System* that is either part of the NAS device or attached to the NAS device externally. It is important to note that the Disk System can be one disk drive, a number of disk drives clustered together in a daisy-chain or a loop, an external storage system rack, or even the storage resources presented to a SAN network that is connected with the HBA of the NAS device. In all cases, the storage resources attached to the NAS device can be accessed via the HBA or Disk controller with block level I/O.

Advantages

The benefit of a NAS over a SAN or DAS is that multiple clients can share a single volume, whereas SAN or DAS volumes can be mounted by only a single client at a time. NAS devices allow administrators to implement simple, low cost load-balancing, and fault-tolerant systems.

Disadvantages

The downside to a NAS is that not all applications will support it because they're expecting a block-level storage device, and most clustering solutions are designed to run on a SAN. Besides the backup solution is more expensive than the storage system. And even, any constrictions in the local area network will slow down the storage access time.