# History & Plumbing: What Happens under the Hood in DAS, NAS, and SAN

# OVERVIEW

Typically, a desire to *utilize assets* … i.e. save money … pushes organizations to consolidate storage. Scattering a hundred file servers around the company -- and then filling up their disks half-way -- wastes space as well as sys admin time. Putting all that disk on one file server allows for more efficient utilization of space and more efficient use of sys admin time. Storage Consolidation is driven by *asset utilization*.

Here at the Hutch, we can see this effect in our current effort to provide consolidated backup services -- an enterprise backup system makes more efficient use of space (space on tape), i.e. increases asset utilization.

While asset utilization generally drives an organization to migrate toward Storage Consolidation, Storage Consolidation provides the foundation for additional applications -- it becomes part of the larger field of Storage Applications which, in addition to facilitating asset utilization, also facilitates a range of applications such as Backup & Recovery (including Snapshotting), Continuity Management (aka Disaster Recovery), High Availability, and Performance Scalability.

# HISTORY OF STORAGE

## Terminology

NETWORKS

Q: What was the original purpose of the network?

A: To share printers. Printers cost a lot, and sharing one printer amongst many people was and is a way to reduce costs. More generally, the original purpose of networks was to share resources, leveraging the value of a capital investment.

Q: What is the purpose of the network today?

A: Primarily sharing information, though for our purposes today, talking about storage, we will focus on the older purpose of sharing resources.

BLOCK-LEVEL ACCESS VS FILE-LEVEL ACCESS

Some applications want block-level access to disk. For example, high-end database management systems (Oracle, Sybase, SQL Server) prefer to read and write database records via SCSI (or ATA) block commands, rather than by going through the OS' file system.

Some applications want file-level access to disk. For example, Microsoft Word doesn't want to think about SCSI … it wants to issue a FILE OPEN command and just receive the entire document back.

Generalization: most applications want file-level access to disk. Operating systems and database managers are notable exceptions.

## Chronology of Block Access

Aka 'A History of Confusing Naming Schemes'. See the *Block Level Protocol Stacks* hand-out.

### ESCON

In the beginning, there was IBM. And life was good. Mainframes used a physical transport protocol called "byte and tag" to communicate with disks. From the primordial soup of "byte and tag" was born ESCON (Enterprise Systems Connection), an optical-only physical transport protocol interconnecting IBM mainframes with each other and with a variety of peripherals. Regrettably, the protocol implemented on top of ESCON was also called ESCON … leading to a confusion of terminology which has continued to this day. (This is kind of like calling both Ethernet and TCP/IP "Ethernet" … ) ESCON (the upper layer protocol) contains exactly five commands.

### SCSI

In the 1979, Larry Boucher (founder of Adaptec) lead the team which developed SCSI (Small Computer System Interface)[1], a physical transport protocol aimed at connecting hosts and peripherals … an 'ESCON for the rest of us', if you like. Regrettably, the SCSI people called the protocol which rode on top of this physical transport 'SCSI' also … thus perpetuating the naming problem which ESCON has started. SCSI is way more complex than ESCON … but then again, it works with all sorts of hosts and all sorts of peripherals from all sorts of vendors … not just with IBM big iron. The SCSI spec contains upwards of 2000 commands … ANSI ended up owning the SCSI development process.

### ATA

In the 1980s, IBM developed ATA (Advanced Technology Attachment), a subset of SCSI intended to facilitate the production of cheap disk, for their new PC AT computer. Controllers implementing ATA tend to be cheaper than controllers implementing the full SCSI command set. Disks whose controllers speak ATA (later called "IDE disks", once the ATA controller was glued onto the disk itself) tend to be produced using manufacturing processes which are less rigorous than those used when making SCSI disks … with the result being disk which costs less and fails sooner.[2] Regrettably, the ATA people perpetuated the naming confusion: they call the physical transport network 'ATA' (a newer version of this transport network is called 'Serial ATA') … *and* they call the protocol riding on top of this (the subset of SCSI), 'ATA' also. ANSI ended up owning the ATA development process.

---

[1] "Small" because it addressed the needs of little machines … machines which didn't require an entire room complete with tons of AC and filtered power in which to live.

[2] An audience member claimed that vendors are moving toward producing all disks using the same manufacturing processes and only at the end gluing an ATA or SCSI controller chip onto them.

FIBRE CHANNEL

In 1994, ANSI ratified the Fibre Channel protocol. This is an optical-only physical transport protocol designed to interconnect hosts and peripherals. Its development was influenced heavily by ex-mainframe storage weenies from IBM. In some sense, it looks and feels like ESCON version 2. However, it disentangles the physical transport protocol from the upper layer protocol riding on top of it: one can run ESCON (the storage control protocol, not the physical transport protocol) on top of Fibre Channel and one can run SCSI (the storage control protocol, not the physical transport protocol) on top of Fibre Channel.[3] For that matter, one can run IP on top of Fibre Channel, too.[4] In this sense, Fibre Channel functions as a competitor to Ethernet/IP, in the same way that ATM competed against Ethernet/IP.

At last, we have a chance to achieve a coherent naming scheme … notice, however, that we are stumbling here, that common parlance perpetuates the past confusion … we tend to talk about "Fibre Channel disks" … realize that there is no such thing as a "Fibre Channel disk" … disks are glued to controllers which speak ESCON, SCSI, or ATA … that's it. Now, some hosts have Fibre Channel NICs installed in them, which, on the back-end, are attached to SCSI disks … the Fibre Channel NICs grab the SCSI frames, encapsulate them into Fibre Channel frames and spit them out onto a Fibre Channel network … but there are no Fibre Channel disks.[5]

## Chronology of File Access

File-oriented protocols developed in the 1980s -- by 1986 (AFP), all these protocols were in place and widely deployed. See the *File Level Protocol Stacks* hand-out.

NFS

In the beginning of networked microcomputers, there was Sun. And life was good. Sun invented NFS (Network File System), and microcomputers could then share files with each other.

NCP

Novell developed NCP (Netware Core Protocol), and those geeks whose parents didn't let them play with Unix could now share files.

PCLAN/SMB

IBM and Microsoft developed PCLAN, which evolved into SMB (Server Message Block), so now mainframe geeks whose parents made them mess with these dumb little microcomputers could share files.

---

[3] I do not believe that any vendor has produced products which run ATA on top of Fibre Channel; but in theory, this would be possible.

[4] And, in theory, other typically LAN-based protocols as well … like IPX and AppleTalk … but, no one does.

[5] In the same way that some hosts have Ethernet NICs in them, which grab TCP/IP frames, encapsulate them in Ethernet frames, and spit them out onto an Ethernet network. But we don't call the disks in these machines 'Ethernet disks'.

LANTASTIC

Artisoft promoted peer-to-peer file sharing (no centralized server -- every workstation shared its disk), so penny-pinchers could share files.

AFP

Apple followed with AFP (Apple Filing Protocol), because the rest of us wanted to share files, too.

## Chronology of Storage Management

- In the beginning (of open systems), we had lots of microcomputers with their own disks … connected to a network to share printers.

- We added a file server. The microcomputers still had their own disks … but we tried to persuade people to save their files to the server's disk.

- We interconnected these departmental LANs into a big corporate LAN … accelerating many-to-many connectivity.

- We centralized the administration of these departmental file servers … and centralized IT began to stress about maintaining all these little servers.[6]

- So centralized IT started consolidating them, first into small clumps and later into bigger clumps (server farms, data centers). This step simplified management and administration and backups … eliminated duplicate resources (keyboards, monitors) … but storage was still dedicated to each server. Centralized IT spends more and more of its time taking servers down to add disk or shuffling groups of users from one server to another as they outgrow the storage located there.

- That brings us to today. Data resides on enterprise servers, on departmental servers, and on clients. It may not be growing exponentially, but it is growing rapidly. The cost of managing this distributed storage starts to approach the cost of the devices themselves (some folks claim that the cost of managing the storage exceeds the cost of the storage itself …)

- This brings us to the modern era of the Storage Area Network. Multiple servers share the same storage device, which the administrator hacks up into separate chunks and then dedicates each chunk to each server. The administrator can perform the following tasks, without taking down servers:
  - add or delete storage for a given server without touching physical disks

[6] In grant-funded institutions, we tend to develop lots of centralized IT departments, reflecting the siloization of resources which our funding model creates. However, conceptually, the same pattern reoccurs -- each of these mini-centralized IT departments finds itself with more and more servers to manage.

- allow servers to automatically allocate additional space as their needs increase (from a pool of unallocated space)
- share additional storage devices like tape and optical
- add additional storage devices as space needs increase

# STORAGE ARCHITECTURES

These are the big three storage acronyms, from a plumbing perspective. See the *DAS & NAS Architecture* and *SAN Flavors* hand-outs.

## Direct Attached Storage

This solution provides applications loaded on *one and only one* server *block-level* access to storage.

Definition

- One server running a general purpose operating system attached via a physical SCSI cable to one disk. Generally, adding disk or performing some disk/volume mappings requires down time.

Variations

- Multiple SCSI cables going to multiple disk arrays … but still attached to a single server.
- Multiple (well … a maximum of two!) servers attached to the storage devices via the magic of dual-ported SCSI -- popular in clustered or highly-available solutions.

Ignoring InfiniBand for the moment, this is the current performance leader in the storage world: dual-attached servers to a RAID 10 disk array equipped with 320MB/s disks can outperform even high-end Fibre Channel/iSCSI SAN solutions.

## Network Attached Storage

This solution provides applications loaded on *multiple* servers *file-level* access to storage.

Definition

- A server equipped with a special-purpose operating system providing NFS and/or SMB file services to other servers and directly to clients … and nothing else. The internal disk is typically attached via SCSI or Fibre Channel cables. Generally, adding disk or performing some disk/volume mappings does not require downtime.

Variations

- Low-end solutions ship as appliances, often consisting of general purpose operating systems hacked to permit administrative access only via a custom Web interface and lacking ‘zero downtime’ features.

### Storage Area Network

This solution provides applications loaded on *multiple* servers *block-level* access to storage.

Definition

- Multiple servers and multiple storage devices attached to a dedicated network carrying block-level traffic (SCSI).

Variations

- The advent of iSCSI allows the previously dedicated network to be overlaid on top of the enterprise's commodity IP network.

### NAS Head + SAN

This solution provides applications loaded on *multiple* servers *block-level* and/or *file-level* access to storage.

Definition

- A NAS box plugged into a SAN back-end.

## SUMMARY

### The Driver for Storage Management

Storage Management exists because customers want more tools to manage space … to better utilize this asset.

### The Key to Storage Management Success

Virtualization, aka indirection. When one host is attached to one disk, our choices are limited. But once we abstract the connection between host and disk, all sorts of capabilities arise:

- Transparent growth and shrinkage of space
- Transparent improvement in performance
- Transparent reduction in MTBF
- Transparent increase in number of clients

This is where storage management technologies spend their energy and their complexity: implementing one (or more!) layers of indirection between the application and the disk.

This indirection can be implemented in the disk, in the host, in the network out-of-band, in the network-in-band … there are many ways to do this.

## FURTHER READING

http://www.skendric.com/san