SOFTEK™

Simplifying Storage Management




# Characterizing Storage Platforms For Effective ILM

Nick Tabellion

CTO – Softek Storage Solutions

# Agenda

- Elements of an ILM solution
- Dilemma of distributed systems
- How to classify storage

# Historical View

- IBM defined the problem of storage management with the help of customers
  - GUIDE White Paper around 1980
    - 30% utilization, 11 GB/storage administrator
- IBM management convinced of importance
  - Future growth rates for storage and servers
- Design and development groups formed
  - Jupiter Joint Study group formed in 1983
- ‘DFSMS’ 1st release – Dec 1988

# Elements of an ILM Solution

- Understand the logical requirements of the applications that access the data
  - How often, fast, security, life span, etc
- Control where data is moved to
  - Find the storage, allocate, update meta data, how to access at new location
- Move the data to new storage
  - Including the ability to go up or down the hierarchy

# Mainframe Approach

- Classify data (logical requirements)
  - Data class
  - Management class
  - Storage class
- Classify storage (physical)
  - Storage group
- Key Concept
  - Separate the logical application requirements from the underlying physical storage
- Management class examples
  - Migrating to ML1 and/or ML2
    - How long to keep data on primary storage
  - When to migrate up to high performance
  - How long to retain active copy/backup copies/archive copies, etc.

# Mainframe Approach

- Where is data moved to (DFP)?
  - DFP provides the Class definitions, directly manages the storage and allocations, accesses the data directly for applications and keeps track of all the meta data
  - Shared file system across all disks
    - Data shared across applications
    - Disks shared across multiple systems
    - This is a major requirement if you really want automated storage management

# Mainframe Approach

- Moving the data to different storage
  - Migrate data using class characteristics
    - DFHSM – Hierarchical Storage Manager
    - OAM – Object Access Method
  - This is an area that still needs improvement
    - Moving data completely transparent to applications, even while in use is a major goal
    - Softek's TDMF (Transparent Data Movement Facility) goes a long way in this direction

# Advantages

- Optimize space utilization
  - 75% utilization
  - data on the right media (disk, tape, optical, robotic)
- Improve application performance
  - Paths, cache, device optimization
- Improve availability
  - Local/remote mirroring, backup/recovery, snapshots
- Transparent installation
  - Generic specifications, *TDMF*
- For IBM:  sell more DASD and Servers
- For customers: lower costs, higher productivity

# How was it implemented?

- IBM evangelism
- Long period of time
- Comprehensive migration and co-existence capabilities
- Customers varied in the granularity of their classification schemes and speed of implementation
- The greater the detail of the class scheme, the more value derived from SMS

# The Rise of Distributed Systems

- No equivalent to SMS in open world
- HW vendor competition created barriers to cross-platform data management
- OS and File Systems introduce semantics differences
  - No wide spread shared file system
- Even the way that data is written to disk varies from one manufacturer to another
  - I don't care unless it effects my logical processing requirements and/or cost structure

***Threw the baby out with the bath water when we moved apps off mainframes to distributed platforms…***

# At the Component Level

- Different products aimed at different targets, according to vendor literature…
  - Core versus Edge array products
  - Enterprise versus desktop disk drives
  - Different RAID levels for different data
  - Interconnects differ in terms of scalability, performance, cost, etc.
  - WORM for legislative compliance

# Evolution to “Networked Storage”

- Difficulties managing DAS
  - *Of course – it’s growing like crazy*
- Bifurcation of storage (c. 1995)
  - NAS for files (cheap and slow)
  - FC fabrics for blocks (expensive and fast)
  - *Marketing speak – since all access and recording on disk is by blocks*
- Virtualization (c. 1999)
  - “Storage Pools” (aggregations of LUNs)
  - Political issues (are LUNs all the same?)
  - *FUD while major vendors struggled to get their own while preserving their base*

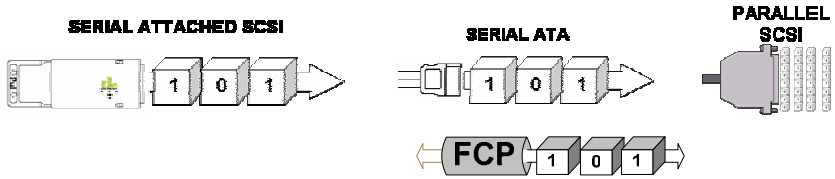
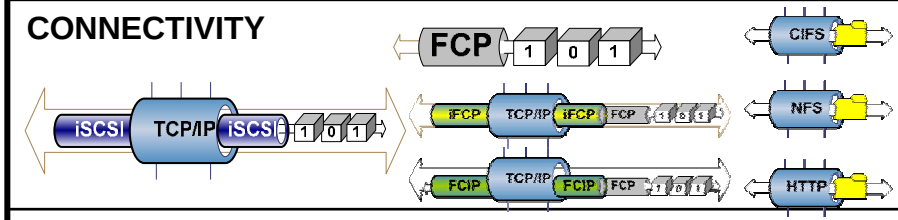
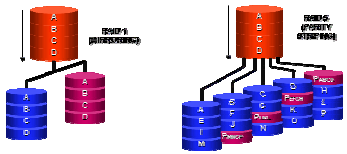
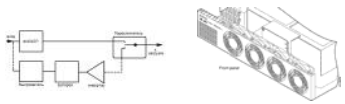
# SANs don’t change FACTS

- Can’t dumb down storage simply through application of FC and Virtualization
  - But these two enable fantastic benefits
- All components are not the same
- Storage architecture increasingly multi-tier
  - different tiers supporting different needs
- No one size fits all
  - storage infrastructure should be built to meet needs of specific applications

# Differentiators Exist

- Component technologies
  - Different drive types offering different capabilities
  - Different RAID levels
  - Different interconnects to support different architectures and services
- Significant ***price, performance and availability*** differences for a given capacity
  - You must have different classes of storage

# Technical Differentiators

| | |
|---|---|
| **DISK**<br>**TIER-ONE**<br>**TIER-TWO**<br>  | Disk drives have different capabilities and price points mapped as much to performance and availability characteristics as to capacity. Getting through self-serving vendor characterizations can be a challenge |
| **CONTROLLER, INTERFACE & BACKPLANE**<br>  | Controllers provide the connectivity inside and outside the array. Interface options are going serial with the advent of Serial ATA, Serial Attached SCSI and Fibre Channel, but parallel SCSI will live until cost becomes issue. Controllers are major performance and availability providers. Serial came into MF's in early 80's. |
| **CONNECTIVITY**<br>  | Will the array stand alone, or will it be networked either as NAS or in a SAN or fabric? External connectivity protocols are exploding and the resulting alphabet soup of protocols may seem daunting at first. But it is really pretty simple. |
| **RAID LEVEL**<br>  | Introduced in 1977, Redundant Arrays of Independent Disks (RAID) have become a typical feature of storage arrays. There are five levels of RAID (six, if you count RAID 0 - no RAID at all) but only a few enjoy widespread use. |
| **RESILIENCY FEATURES**<br>  | The right power plant is very important. Redundant fans, power supplies and battery backup should be considered based on data criticality. |
| **DATA PROVISIONING**<br>  | How smart do you want the array to be? Should it virtualize raw capacity into volumes or do you do this with external software? |
| **DATA PROTECTION**<br>  | Do you want the array to manage protection? Should it be smart enough to do its own mirroring or backups? Going external to the box can save big bucks. |

SOFTEK
Simplifying Storage Management

# Storage HW Classes reflect

- Operational attributes of hardware:
  - Paths, ports, LUN sizes and configurability
  - maintenance updates
- Speeds and feeds of arrays
  - Read/write caching, rotational delay, rotation speed
  - seek time, response time, read/write throughput rate
  - sequential and random rates
  - time to first access (mountable characteristic – disk/tape/optical)
- RAID levels and other low level data protection capabilities
- Other services
  - mirroring, snapshots
  - inclusion in backup, compression
  - encryption, WORM

# Significant New Products

- Hitachi TagmaStor - Full blown array and virtualization
  - Controller for all storage with Mirroring & SnapShots
- Avamar Backup – breakthrough algorithms with storage
- Revivio – breakthrough recovery with virtualized storage
- IBM DMS Storage – the controller is really a server (who would have guessed)
  - Lets see what they do with it

# Cost Implications

- Lifecycle management
  - Matching data to the right platform in terms of platform capabilities
    - Suitability to data usage requirements
- Storing data on the platform
  - Balancing importance of data
    - Frequency of update, access requirements, etc., with cost per GB of platform itself

# Storage Selection

- Application logical requirements drive storage class selection
  - High performance transaction processing systems
    - May require data hosting on high-cost/high-performance arrays featuring dual port drives, RAID, mirroring, etc.
  - E-mail
    - Data might be hosted on less expensive/lower performance arrays with single port drives, RAID, etc.
  - Many variations between the two
- Technology Upgrade
  - When I buy a new Wiz-Bang 30TB array, how do I get my data onto it in the right classification while my applications are running my business?
    - Installation management

# How do you classify storage?

- Capabilities
- Services
- Costs
- Create storage platform profiles with a view of data requirements

# Storage Class is Critical

- Storage classification scheme
  - ILM/DML requirement
    - Impractical otherwise
- Physical class
  - Defines target storage for data management
- Policy-based data mover
  - Uses storage classification scheme, data classification scheme and access frequency counter
    - Determines what, where and when to move
  - Softek Storage Manager example:
    - Disk > 80% full AND File last accessed => 6 months ago Archive to robotic tape system and delete it from disk

# Final Thoughts

- There is a wide variety of storage with significant capability and price differences
- It's impossible for humans to efficiently manage thousands of disks and millions of files
  - (SSM has found 650 million files at 1 account)
- Classifying the logical requirements of applications and physical capabilities of storage
  - can lead to automated storage management solutions that can save you HUGE amounts of $$$$$
  - Thousands of Z/OS MF users enjoying benefits now

SOFTEK™

Simplifying Storage Management




# Thank You