



Storage Systems Division

# Virtualization in Storage Area Networks

Torsten Rothenwaldt  
IBM European Storage Center

# Agenda

- Present storage virtualization models and concepts.
- Compare different architectures.
- Explain implementation details of disk block and file virtualization.
- Discuss current enterprise usage of disk/file virtualization.

# Table of contents

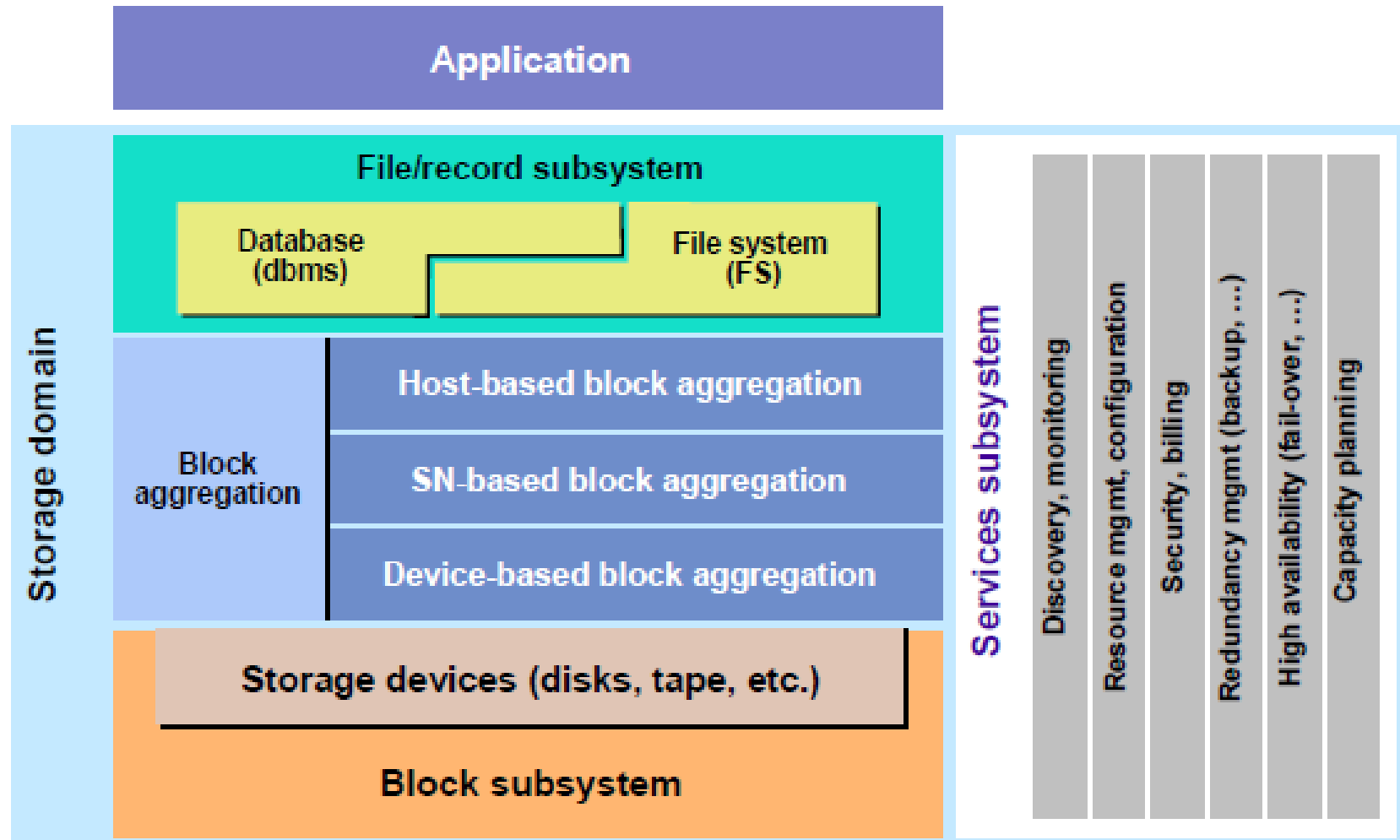
## **Models and concepts**

Disk block virtualization

File and record virtualization

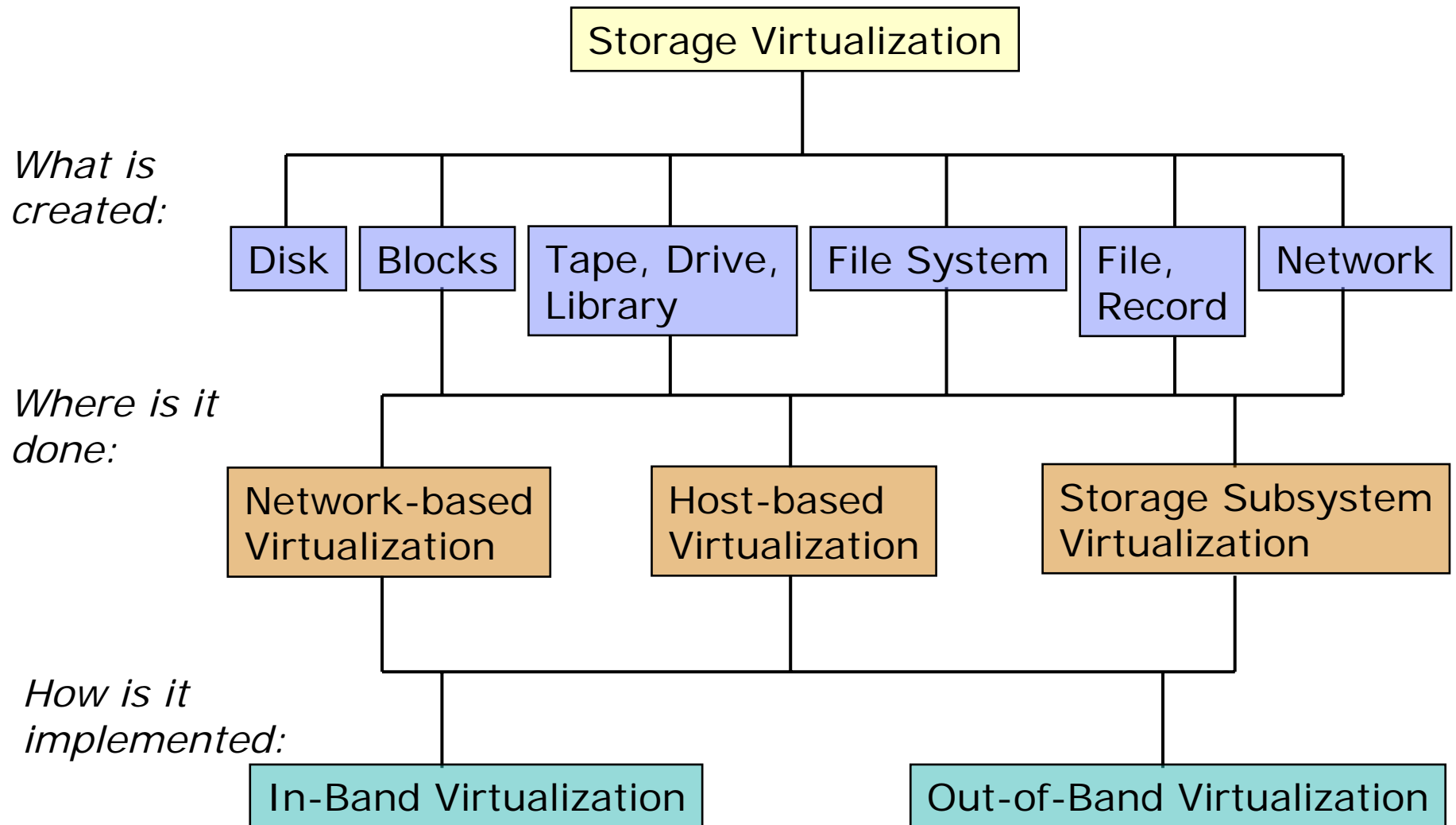
Storage virtualization in enterprises today

# The SNIA Shared Storage Model



© SNIA 2000

# SNIA Storage Virtualization Taxonomy



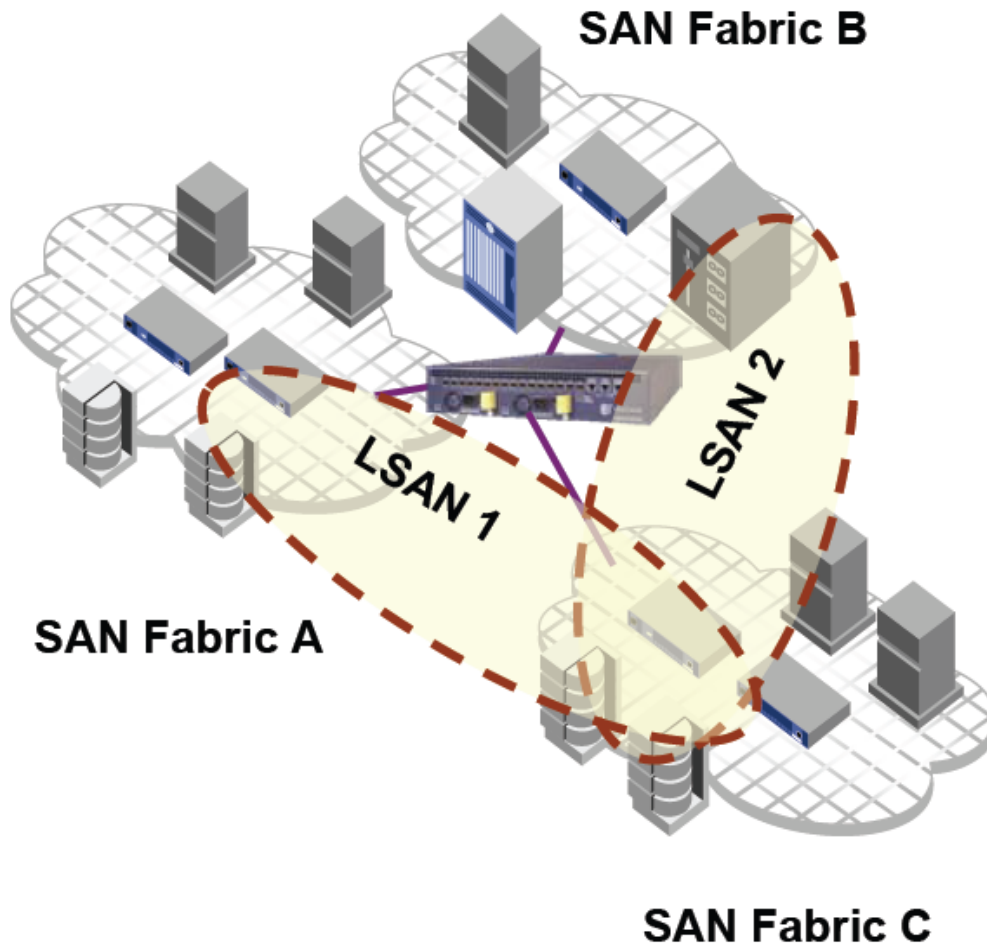
# Tape Storage Virtualization

- See previous presentation.

# Network Virtualization

- Aggregate multiple SAN islands into a unified infrastructure
  - Multiple SANs of the same protocol (native or tunneled)
  - Multiple SANs of different protocols (FCP, iSCSI, FCIP, iFCP)
  - Implemented with bridges, routers, gateways
- Provide multiple SANs on top of a common infrastructure
  - Switch partitioning
  - Virtual fabrics on top of real fabrics

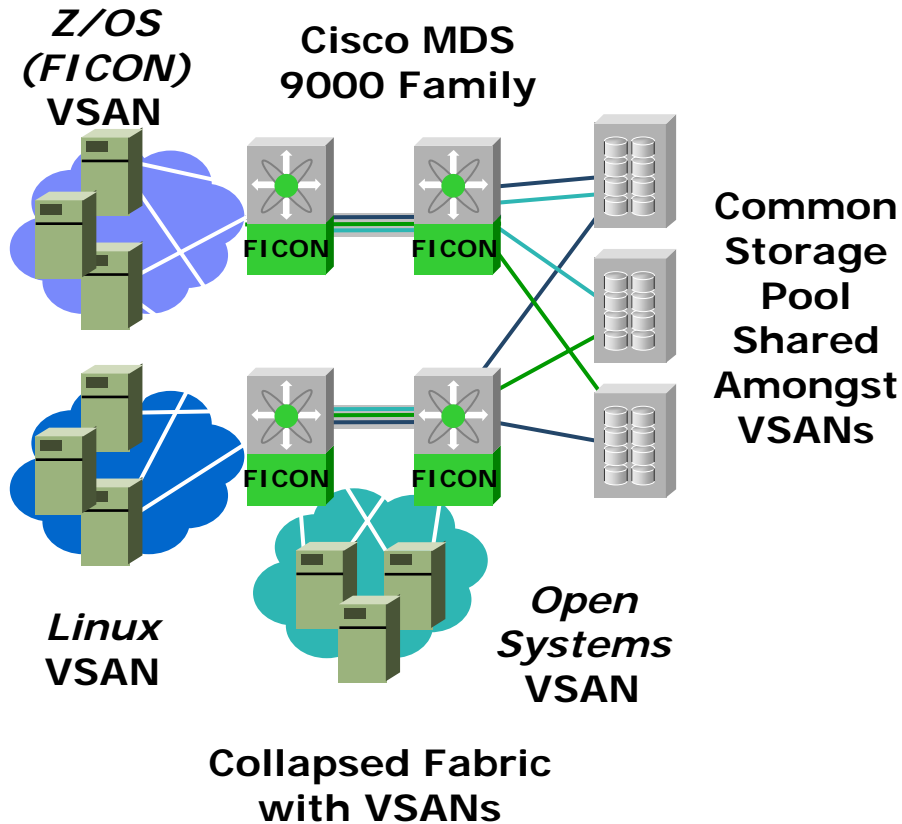
# Network Example: Brocade LSAN



- Two SAN islands (same or different protocols) connected by router
- Selected ports from the remote SAN are zoned into the local SAN

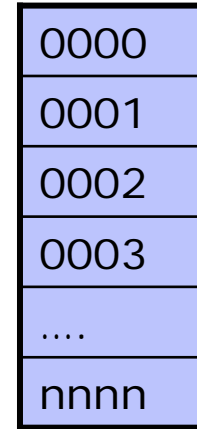
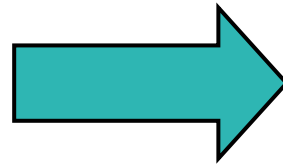
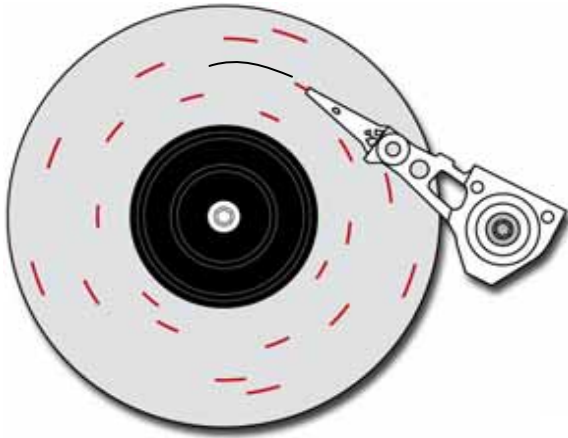


# Network Example: Cisco VSAN



- Partitioning into different operating environments (FICON, Linux-FCP, Open Systems FCP)
- Port-level granularity
- Shared ISLs

# Disk (Drive) Virtualization



- Physical data layout  
Cylinder/Head/Sector addresses
- Media defects

- Virtual data layout  
Logical Block Addresses (LBA)
- "Defect-free"

# Table of contents

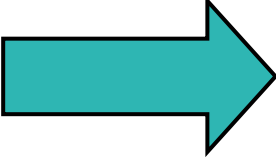
Models and concepts

 **Disk block virtualization**

File and record virtualization

Storage virtualization in enterprises today

# Block-Level Virtualization

- Bound to a physical machine
  - Fixed size and (pseudo-) geometry
  - Limited performance
  - Do break (occasionally)
- 
- Anywhere (can be moved or replaced)
  - Size as needed (can grow, shrink, or morph)
  - Performance scaling
  - Reliable as needed

Where does block-level virtualization reside?

# Virtualization in the Storage Subsystem

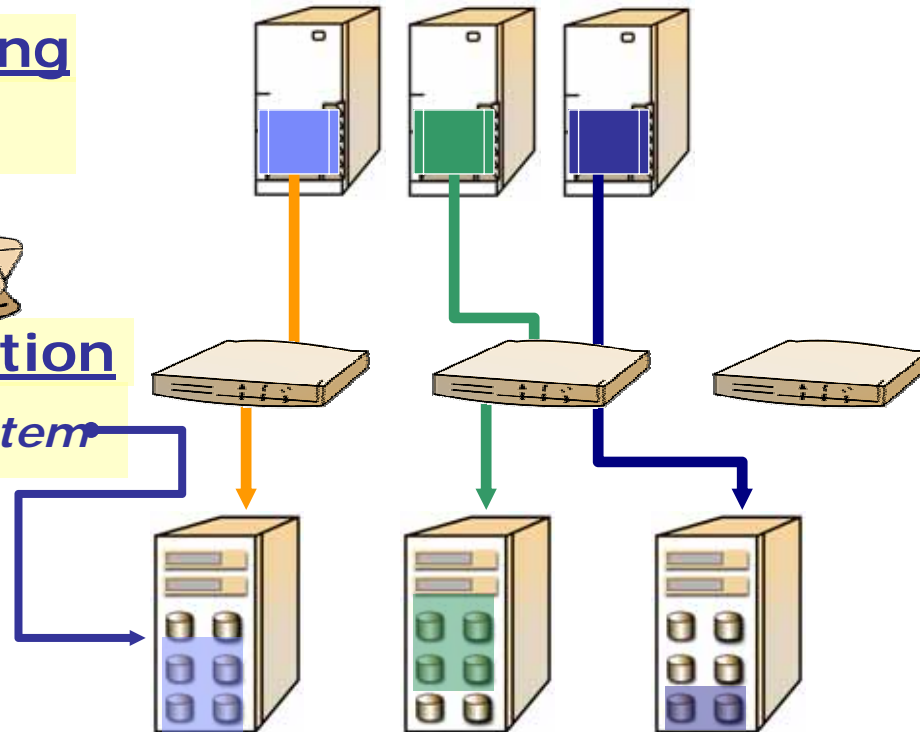
## Provisioning

*Per-host*



## Virtualization

*RAID subsystem*



- + Heterogeneous hosts
- + Mature industry & products
  - Performance
  - Stable & reliable
  - Perceived security

- Homogeneous storage
- Management
  - Device-specific
  - "per-box"

# Virtualization in the Host

## Provisioning

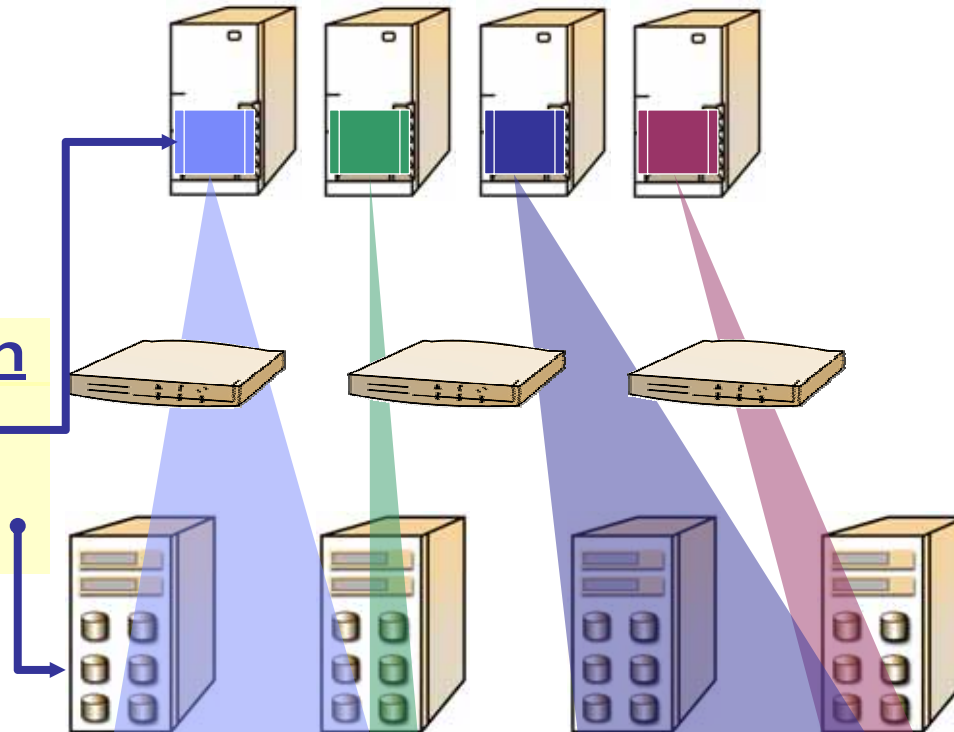
*Per-host...still*



## Virtualization

*Host*

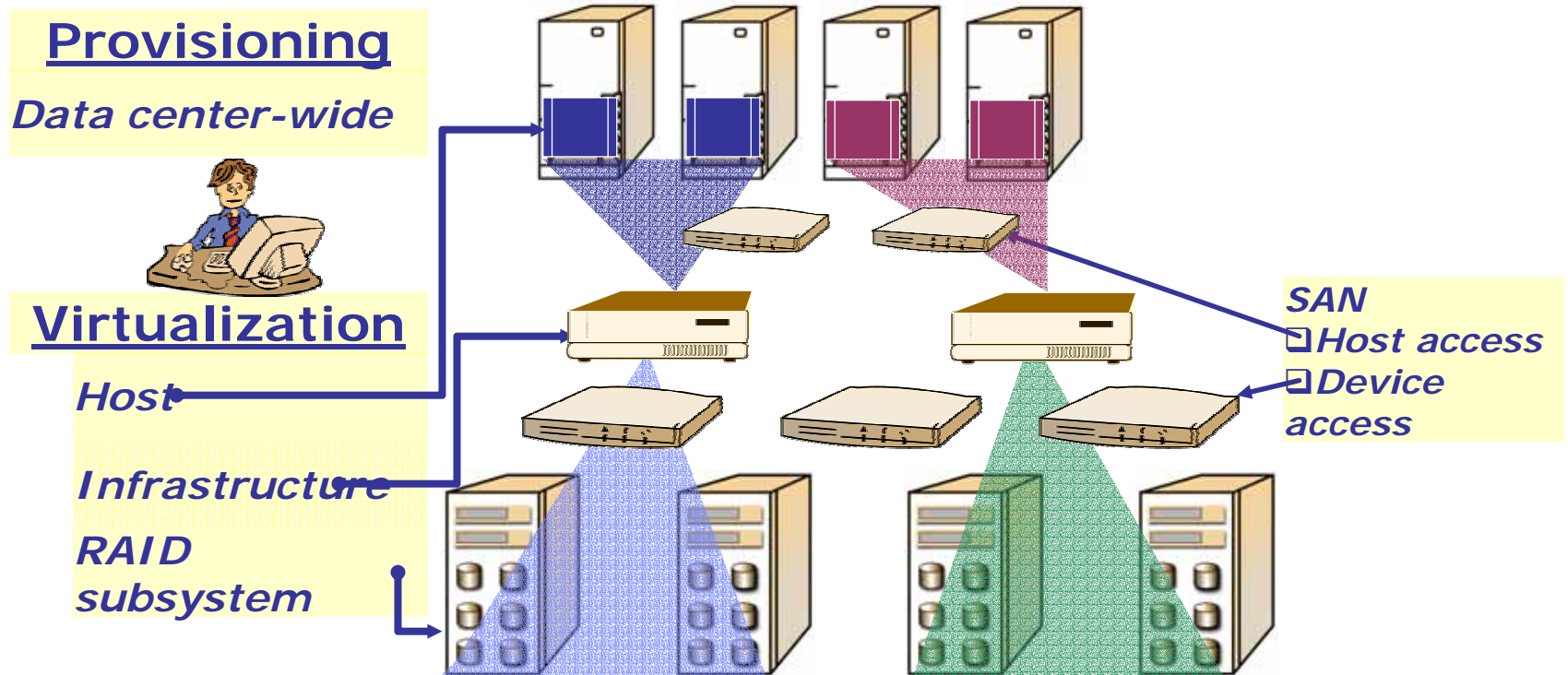
*RAID  
subsystem*



+ Flexibility (any storage)  
+ File system coupling  
(online growth, re-layout,  
movement, snapshots,...)

- Server-centric management  
  of RAID subsystems  
  of volumes  
- Complexity  
  shared data clusters

# Virtualization in the Network: "In-Band"



- + Data center-wide management
  - Heterogeneous storage
  - Heterogeneous hosts

- Complexity
  - User (un)familiarity
  - Integration needs
  - Needs clustering
- Performance perceptions

## Virtualization Devices for In-Band

- Server-based device (appliance)

Virtualizes a variety of physical storage using different HBAs. One pool.

Implements complex storage solutions inexpensively

Adds another layer (managed separately)

Less interoperability issues (no fabric integration)

Appears as standard device.

- Switch-based device (fabric application)

Network optimized

High port counts

Expensive

Elaborated functions not available yet

Fabric integration critical

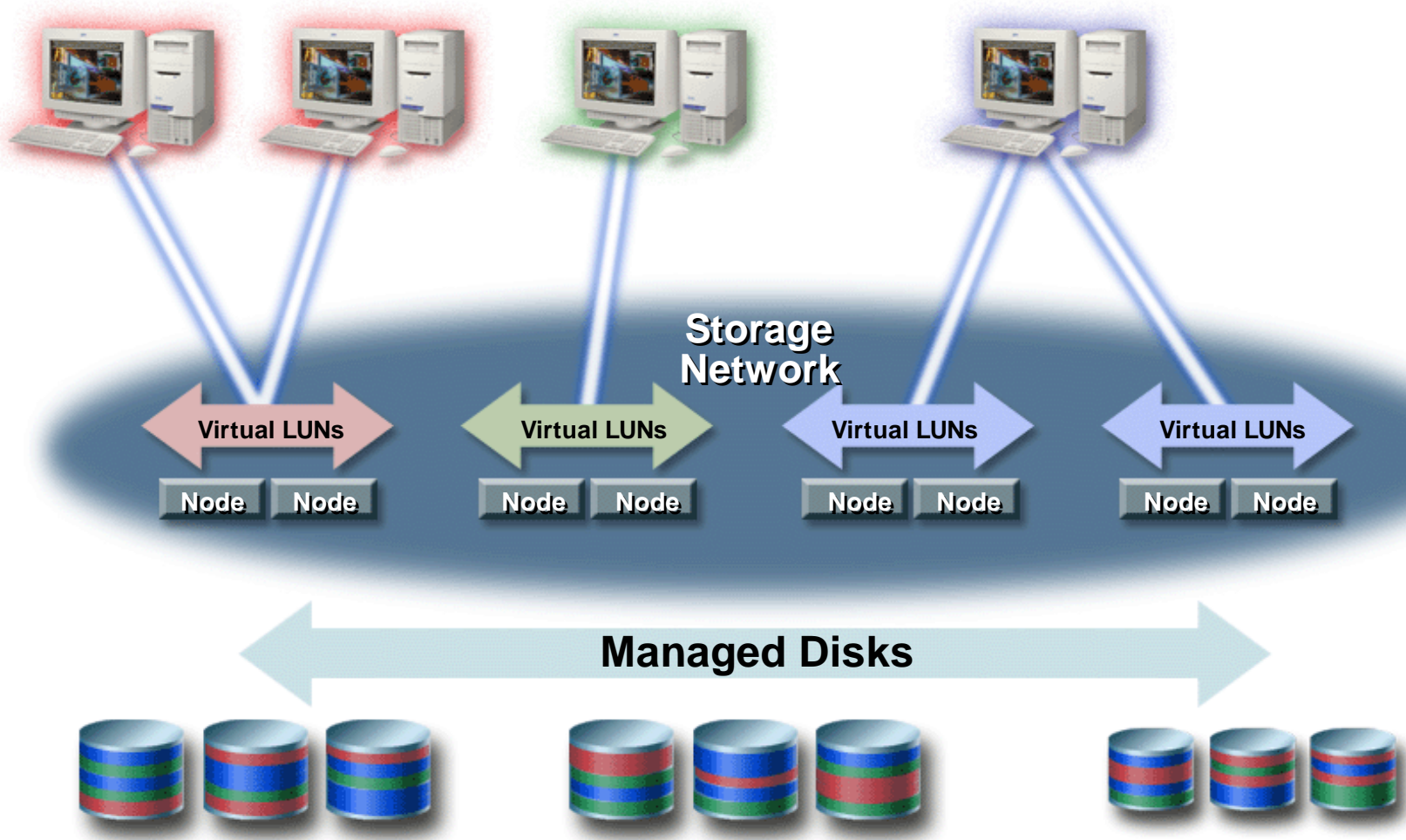
Standardization just beginning (ANSI T11 Fabric Application Interface Standard)



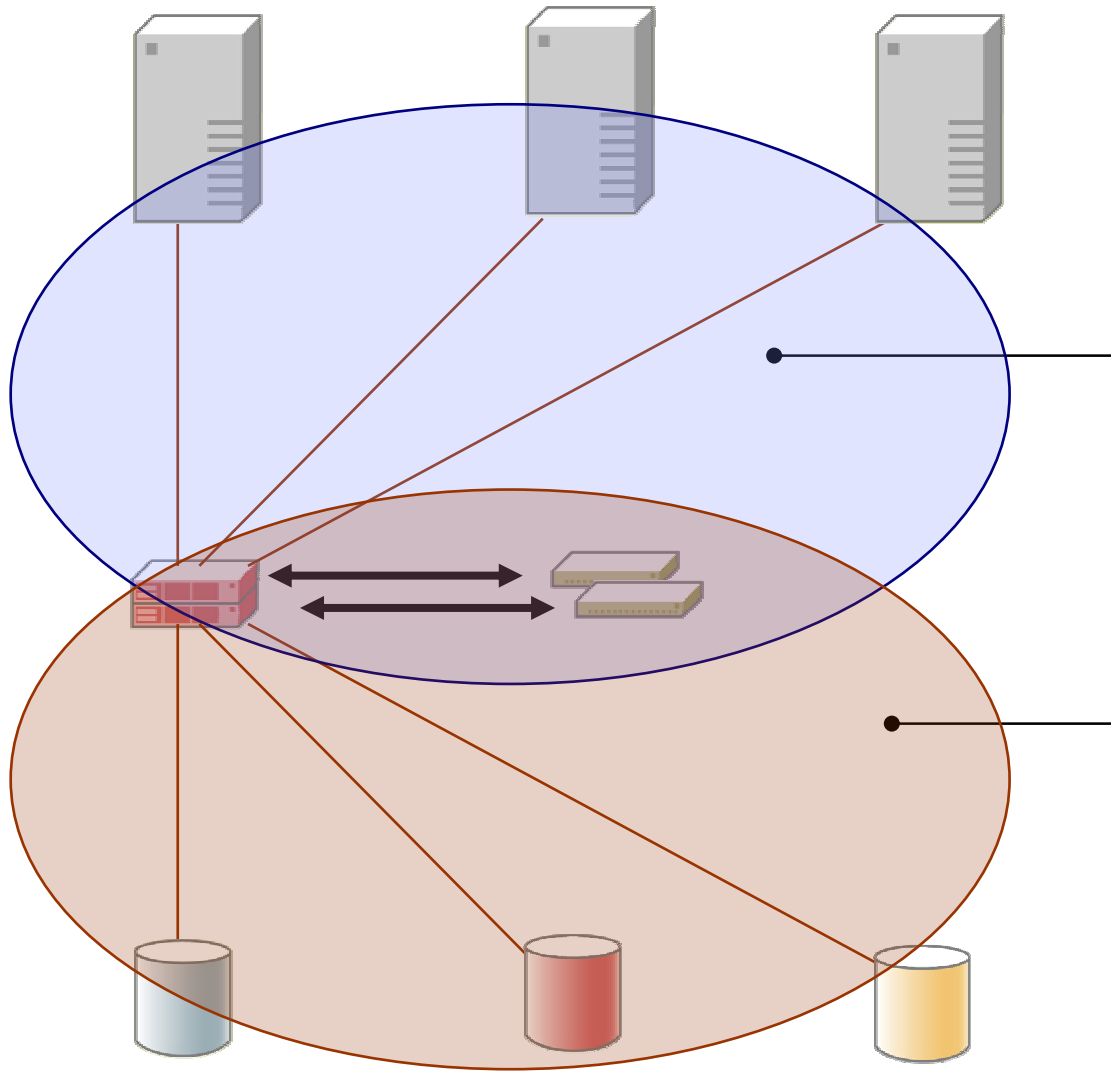
## Example of a Block-Level In-Band Appliance: IBM SAN Volume Controller

- J. S. Glider, C. F. Fuente, W. J. Scales:  
The software architecture of a SAN storage control system.  
IBM Systems Journal, Vol. 42 (2003) Nr. 2, pp. 232-249  
( <http://www.research.ibm.com/journal/> )
- IBM Redbook SG24-6423:  
IBM System Storage SAN Volume Controller.  
( <http://www.redbooks.ibm.com/> )

# IBM SVC: Architecture



# IBM SVC: Zoning



## Host Zone:

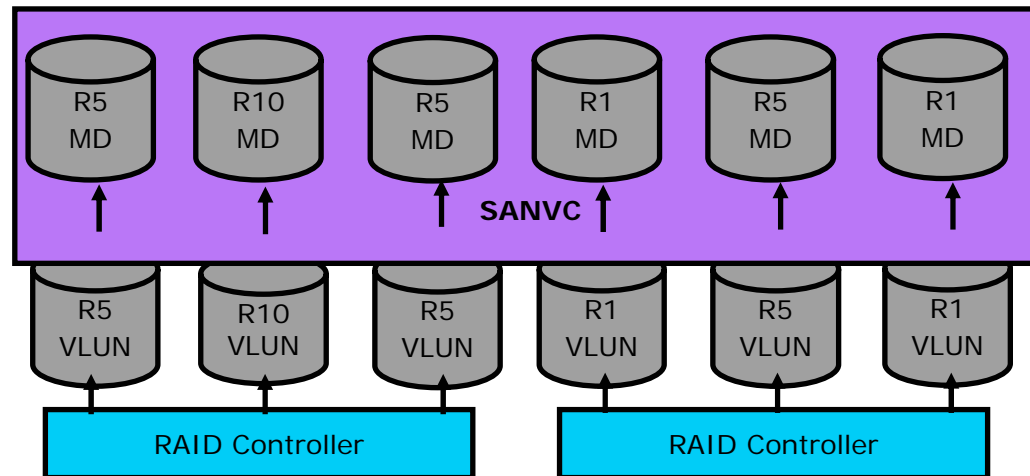
- Hosts zoned only to SVC
- See only Virtual Disks that SVC allows to see

## Device Zone:

- Devices zoned only to SV
- See only SVC nodes as connected hosts

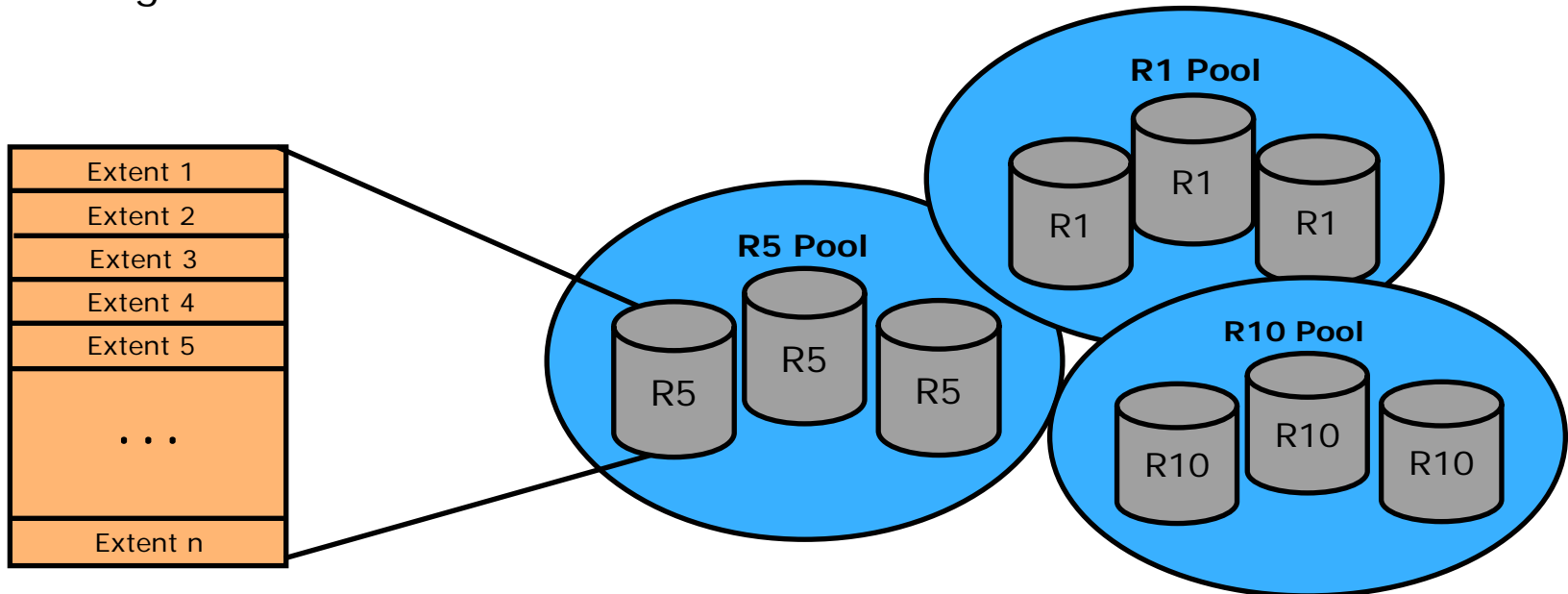
## IBM SVC: Managed Disks

- SVC does not perform RAID functions
- Utilizes RAID capability of backend storage server
- RAID-5, RAID-10, or RAID-1 recommended
- Normally LUNs "surfaced" from storage systems are what the hosts on the SAN see as physical disks
- Disks surfaced by the storage systems are discovered by SVC and referred to as Managed Disks (mdisks)
- Spare capacity on mdisks can be reallocated transparently and dynamically



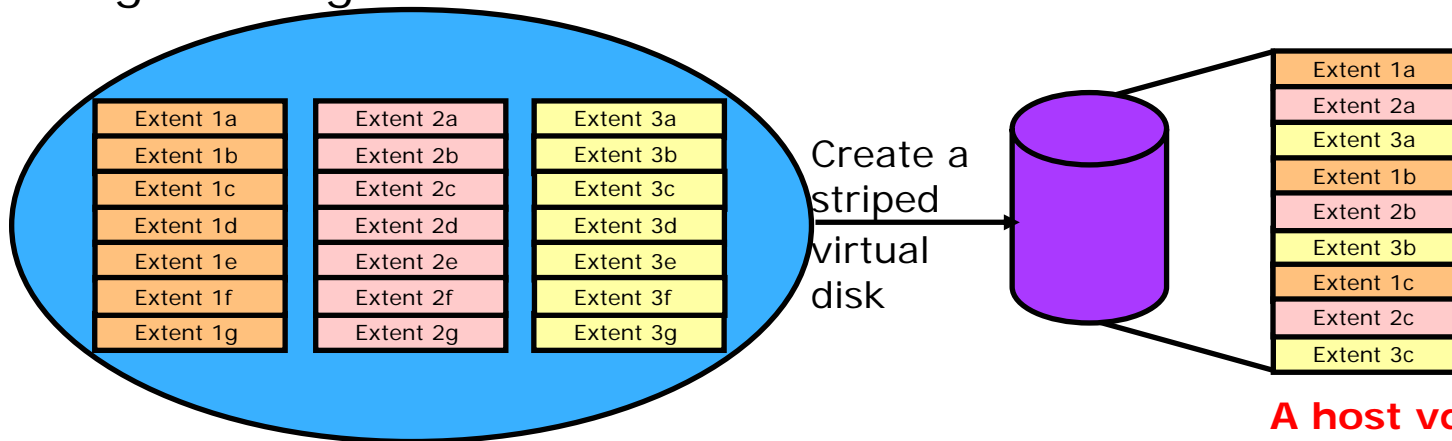
# IBM SVC: Managed Disk Groups

- Once the mdisks are available to SVC, the user assigns them to one or more pools called Managed Disk Groups
- These MDGs are addressed by the SVC in terms of extents:
  - Extent size is determined at MDG creation time, default 16 MB, max 512 MB
  - Extent size determines maximum amount of storage SVC can manage
  - 16-MB extents = 64 TB, 512-MB extents = 2 PB
  - Cannot migrate vdisks between MDGs with different extent sizes
  - Can migrate extents from mdisk to mdisk



## IBM SVC: Virtual Disks

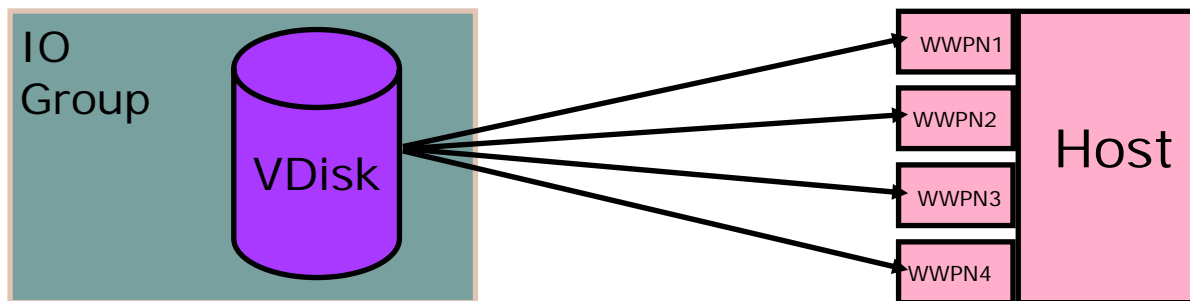
- From these extents, the user can build "virtual disks"
- Various policies can be used to build them:
  - Striped - taking an extent in turn from each disk in the pool or a subset of the disks in the pool (Virtual Disk striped across multiple disks)
  - Sequential - using a single disk in the pool (Virtual Disk mapped sequentially to managed disk)
  - Image - Virtual Disk = Physical LUN
  - SVC Cache - Enabled or Disabled
- Real physical capacity must be available to create a vdisk
- Virtual disks can be expanded, reduced, or deleted
- IO governing can be enabled to limit IO/s or MB/s



**A host vdisk is a collection of Extents - each 16 MB - 512 MB**

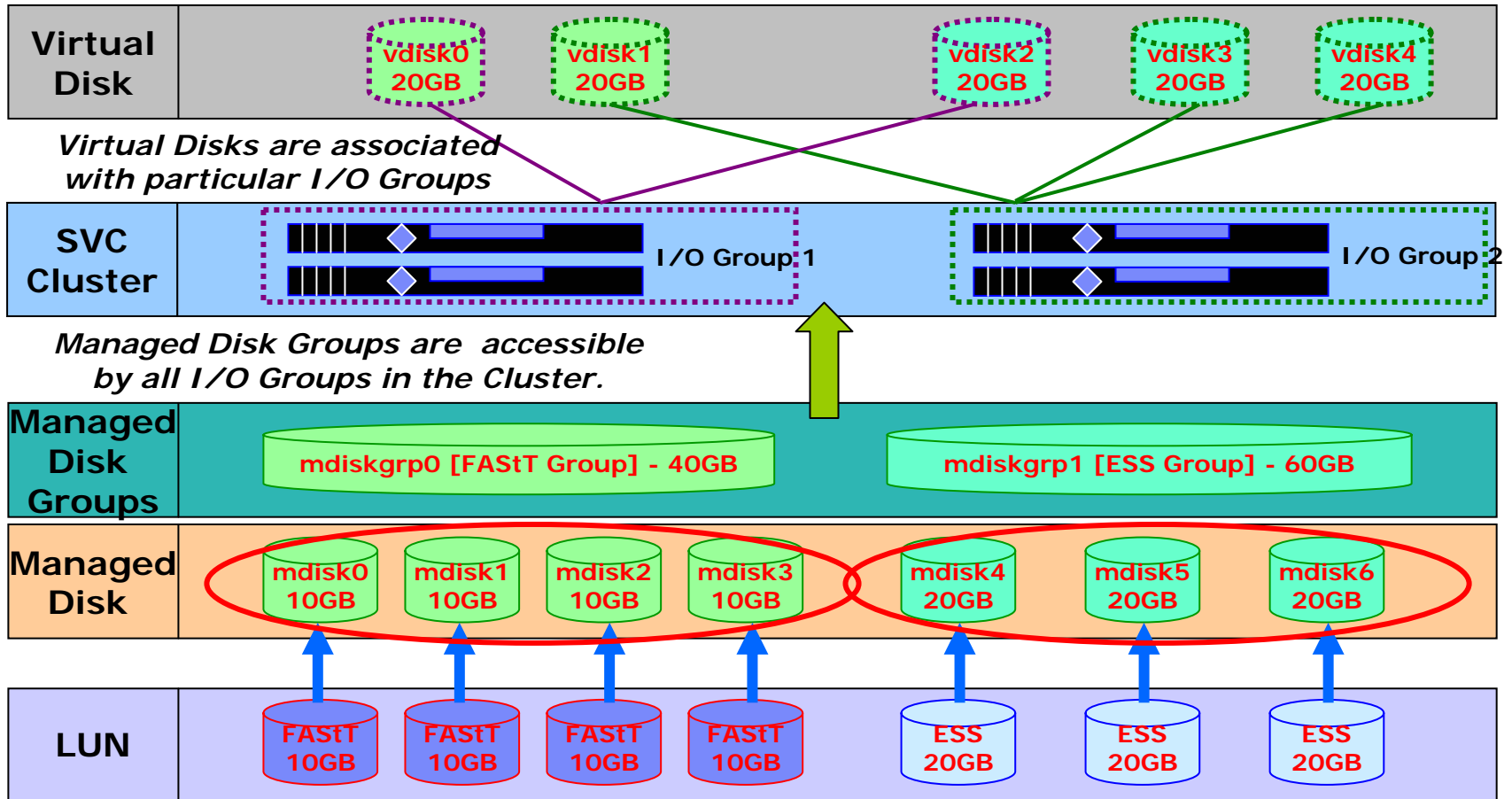
# IBM SVC: IO Groups and Hosts

- Each virtual disk is assigned to a particular IO Group (node pair). Every node in the cluster is aware of the virtual disk, but only owning IO Group services requests.  
All IO is targeted at either of the nodes in the IO Group for purposes of caching and load balancing.
- It is these virtual disks that SVC presents to hosts on the SAN as targets of IO.  
The virtual disks are mapped to hosts (SDD for mutli-path operation). Can be mapped to multiple hosts for use with clustering software. The hosts see these as physical disks (in terms of the OS). SVC knows hosts as groups of HBA WWPNs.



# SVC Combined Physical & Logical View

*Virtual Disks Mapped to Hosts*

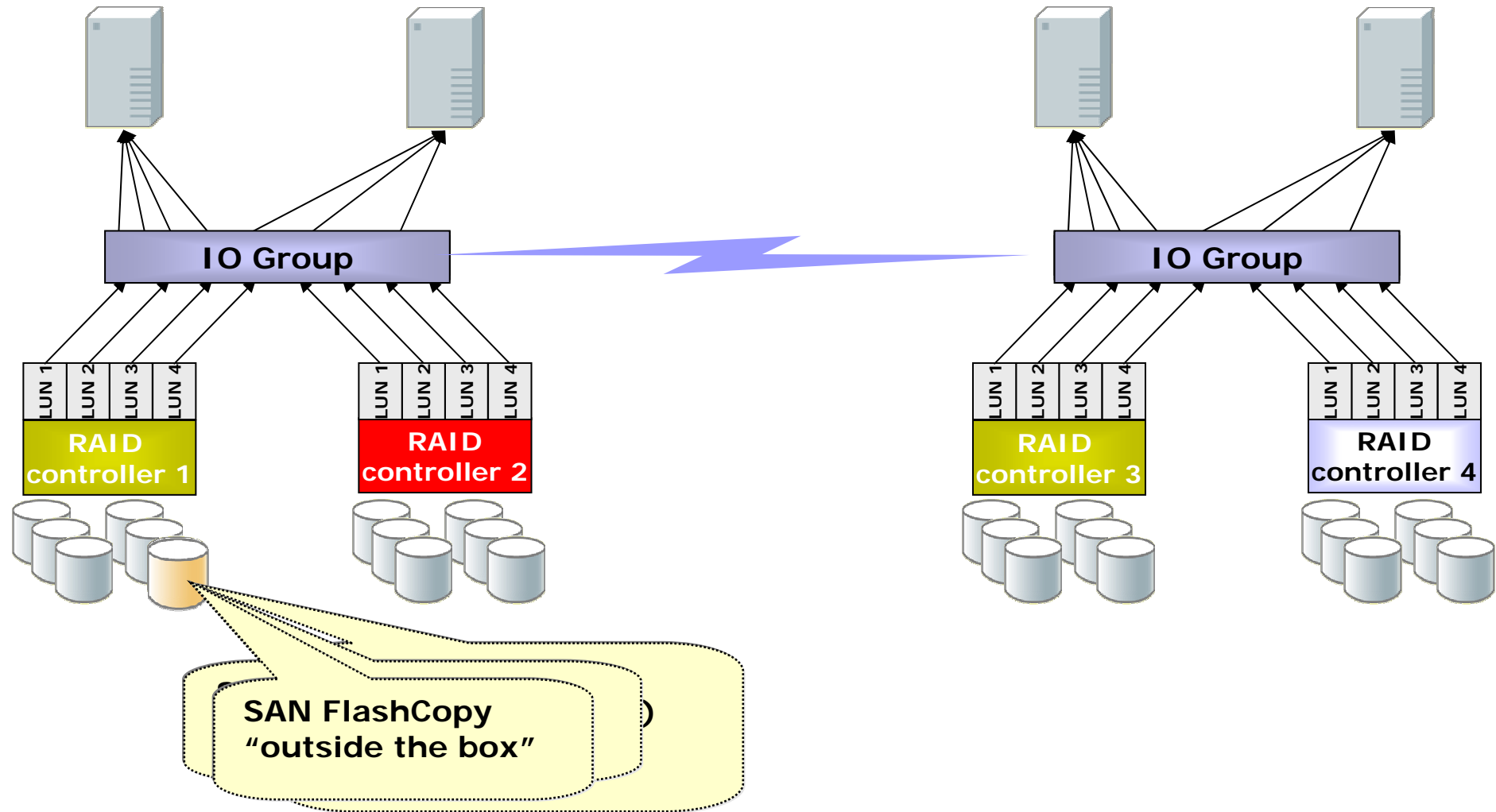




# IBM SVC: Clustering

- Cluster comprised of 2...8 nodes but administered as single image
- No Linux clustering software, failover/failback is function of SVC code
  
- One node automatically designated config/boss node for cluster
  - Assigned cluster IP address
  - Responsible for coordination of node transitions
  - Automatic failover of config node and IP address
- Auto-restart of a node on failure and re-admission to cluster
- Cluster requires majority of nodes remain operating to ensure quorum
- Two nodes is a special case
  - Quorum disk is elected as a tie-breaker
  - Quorum disks are existing mdisks under SVC and use 1 extent
  
- A node stores a write in its own write cache and the write cache of its partner node before acknowledging to the host application
- On node failure, surviving node empties write cache and proceeds in write-through mode

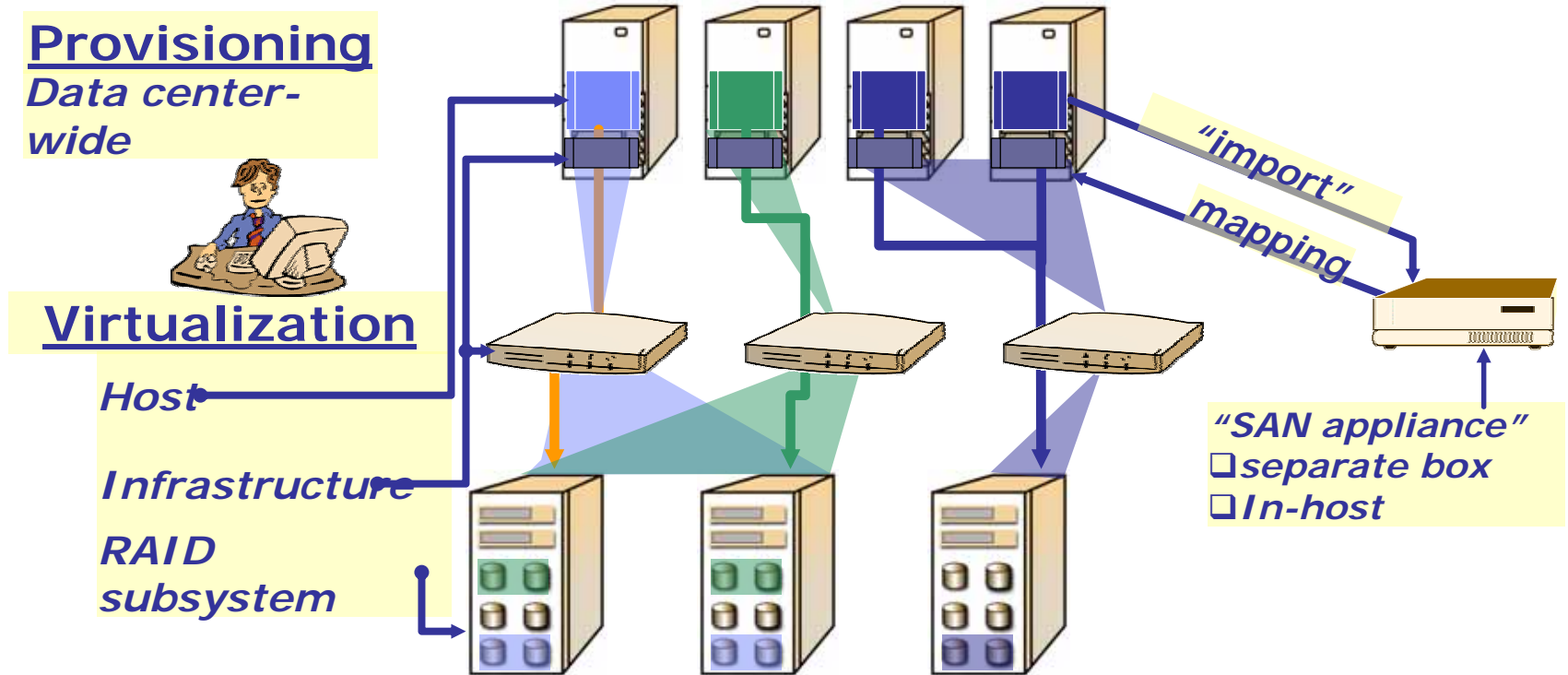
# IBM SVC: Copy Services



## Other Examples of a Block-Level In-Band V.

- DataCore SANSymphony
  - Windows-based appliance (LVM creates virtual disks)
  - Replication synchronous via FC, asynchronous via IP
  - Volume-level snapshots
- HP Continues Access Storage Appliance
  - Fully redundant dual-node active-active replication appliance
  - Replication over FC and IP
  - Volume-level snapshots
- FalconStor IPStor
  - Integrated SAN/NAS infrastructure with FC, NAS and iSCSI
  - Synchronous Remote Mirroring
  - Asynchronous IP based Remote Mirroring
  - Volume Level / File Level Snapshots
  - Application dependent special Functions (e.g. Exchange or Notes Snapshot Agents)

# "Out-of-Band" Virtualization



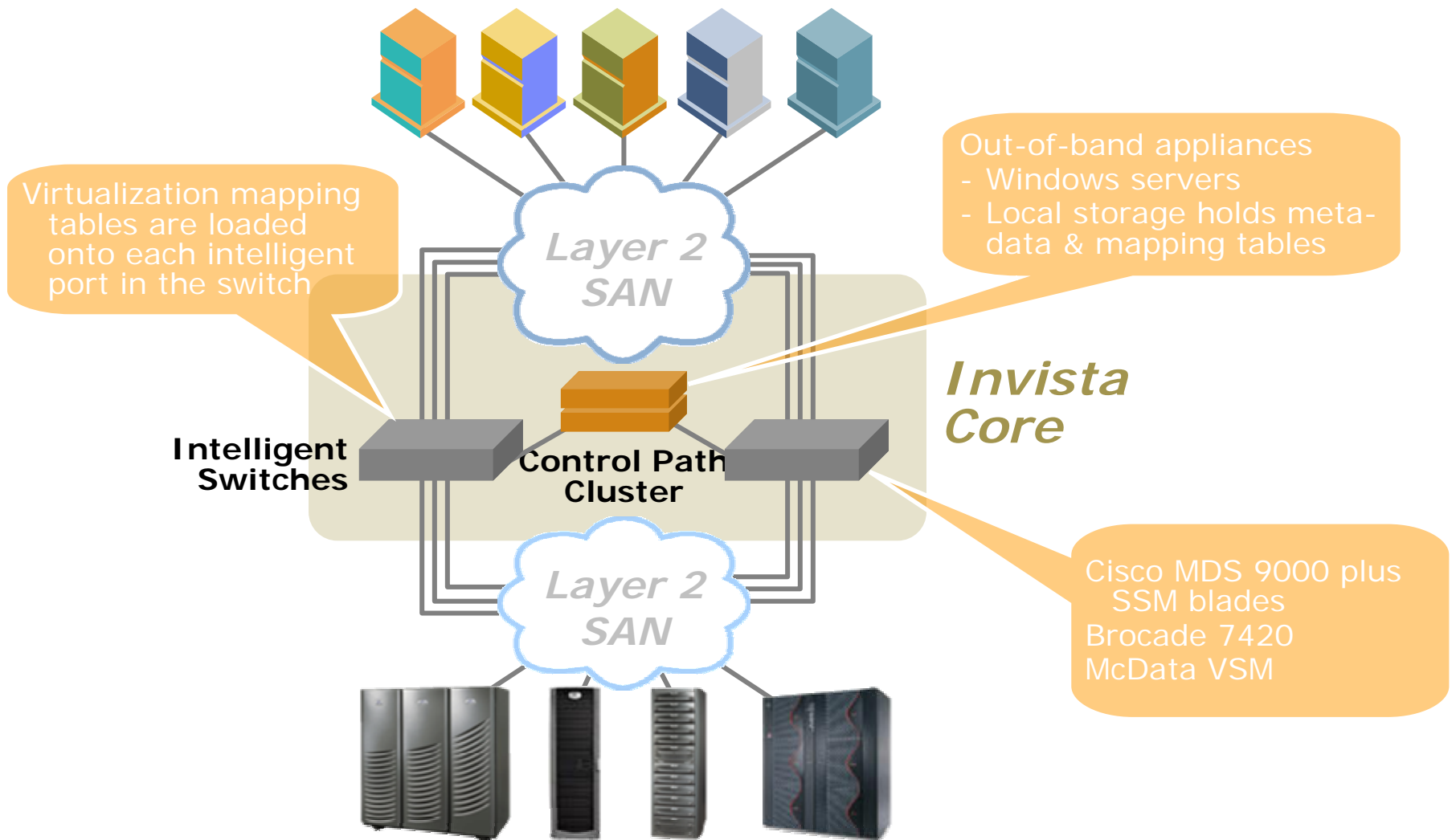
- + Data center-wide management
- + Shorter I/O path
- + Light-weight compared to full volume manager

- May be host-invasive
  - Host-specific
- Appliance availability
- Secure protocols

## Examples of Block-Level Out-of-Band V.

- Compaq project VersaStor (cancelled)
  - Agent in the HBA firmware
  - Maintained there without host OS intervention
- Store Age Virtualization Manager
  - Requires agent and or PCI adapter on host
  - Copy functions

# Block-Level V. Combined: EMC Invista



# Table of contents

Models and concepts

Disk block virtualization

## **File and record virtualization**

Storage virtualization in enterprises today

# File/Record and File System Virtualization

- File system virtualization

  - Aggregates multiple file systems into one large virtual file system

  - Users access data through the virtual file system (underlying file systems transparent to users)

  - Enables additional functionality (special file access protocol on top of one or more existing file systems)

- File/record virtualization:

  - Presents one or more underlying objects (files or directories) as a single composite object

  - Can provide HSM-like properties in storage system

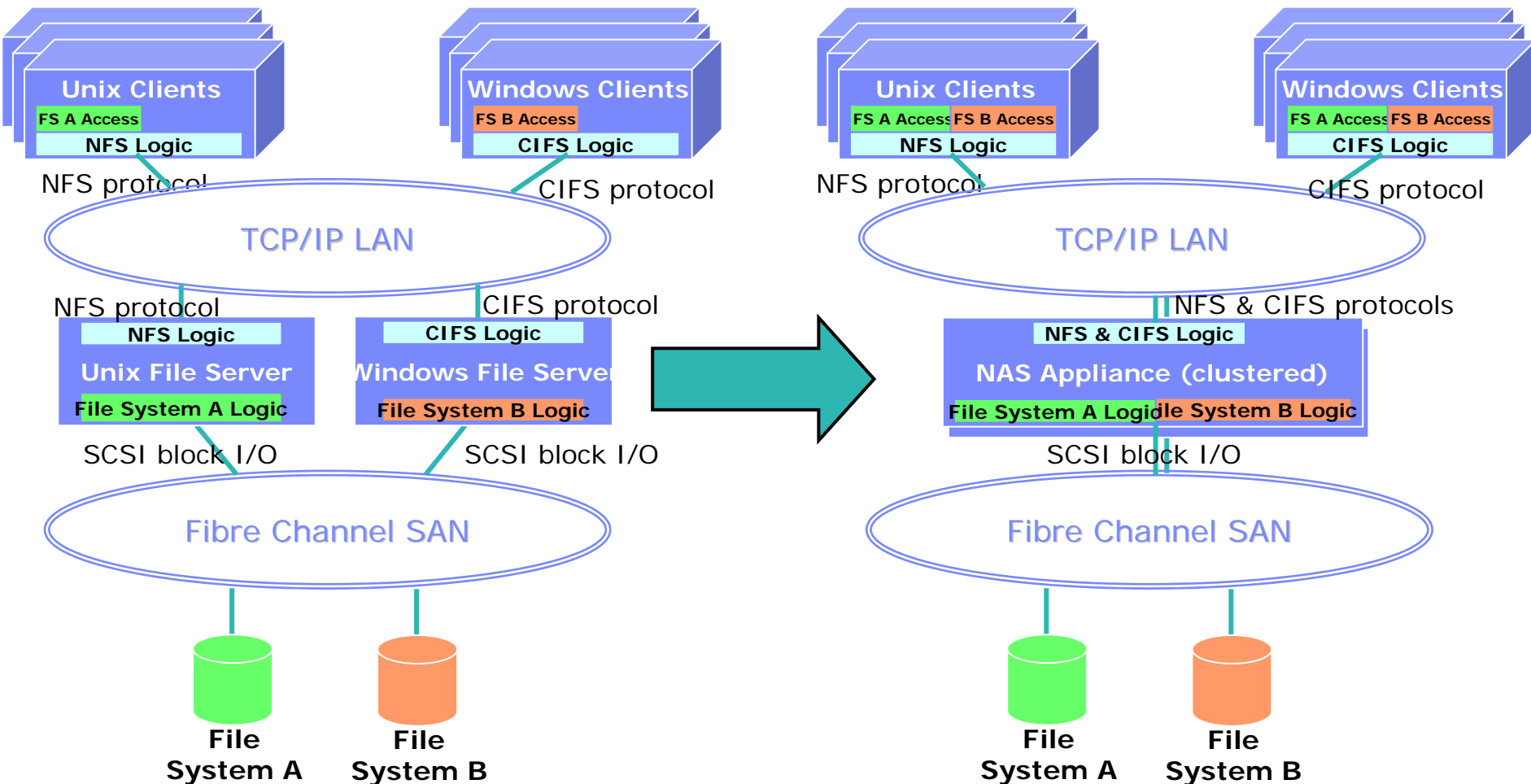
  - Presents an integrated file interface (file data and metadata are managed separately)



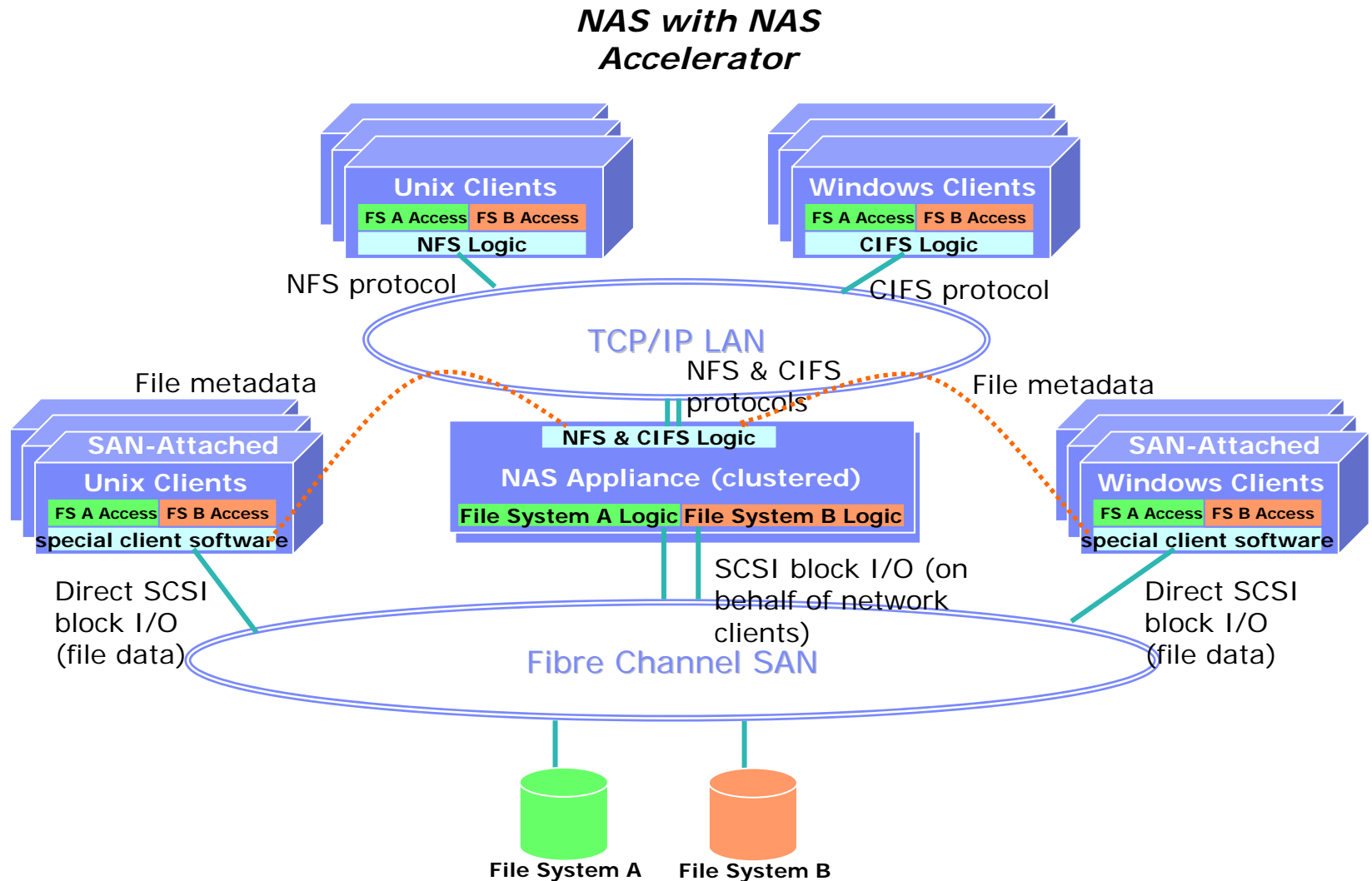
# In-Band File System Virtualization: NAS

**Traditional Network File Server Environment**

**NAS Environment**



# Out-of-Band File System V.: NAS Accelerator

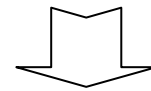
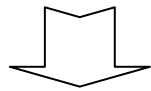


## Example of an Out-of-Band File Virtualization Appliance: IBM SAN File System

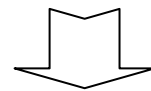
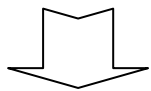
- J. Menon, D. A. Pease, R. Rees, L. Duyanovich, B. Hillsberg:  
IBM Storage Tank — A heterogeneous scalable SAN file system.  
IBM Systems Journal, Vol. 42 (2003) Nr. 2, pp. 250-267  
( <http://www.research.ibm.com/journal/> )
- IBM Redbook SG24-7057:  
IBM SAN File System.  
( <http://www.redbooks.ibm.com/> )

# IBM SFS: Meta Data and User Data

Normal File System  
File Structure

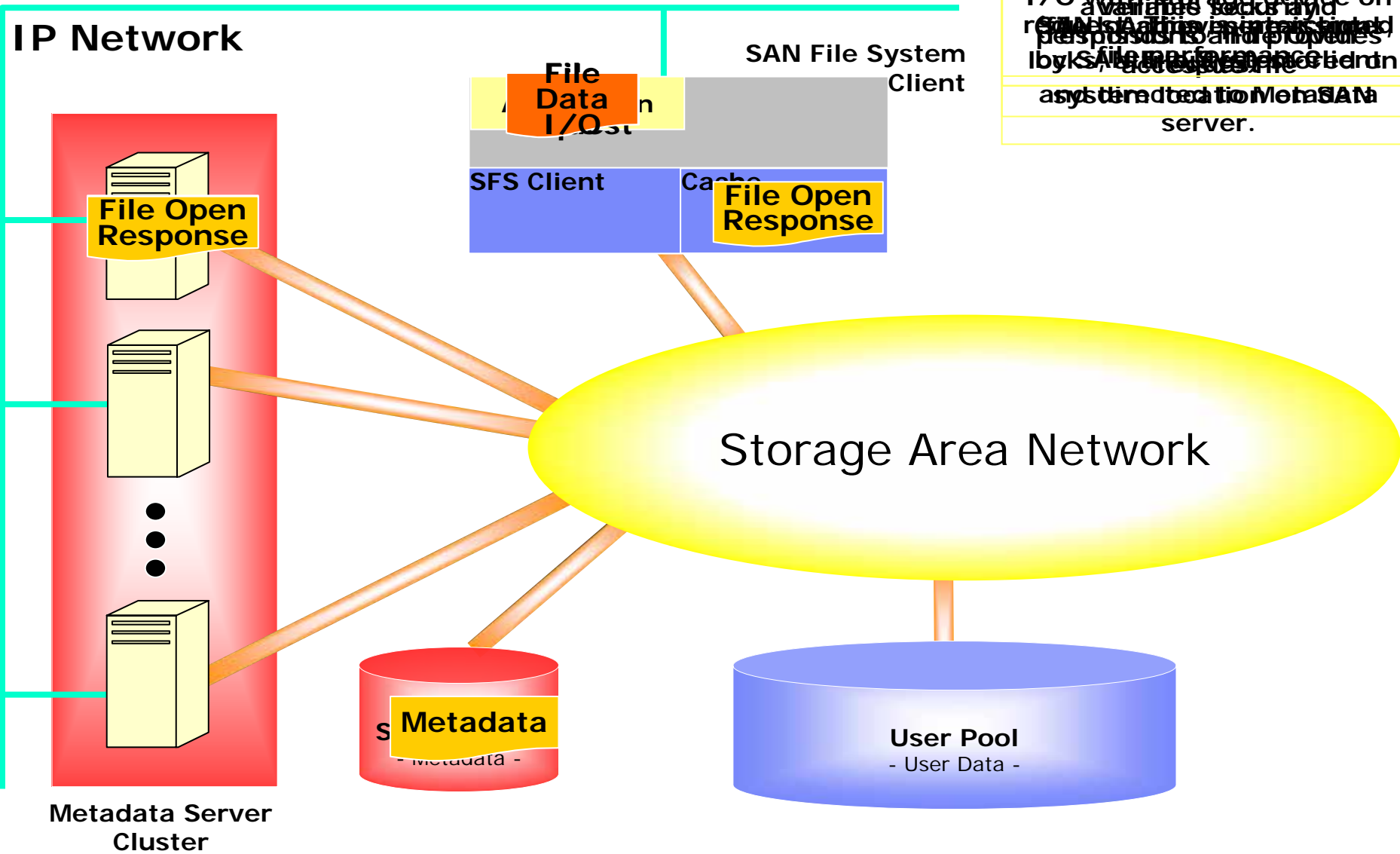


SAN File System  
File Structure

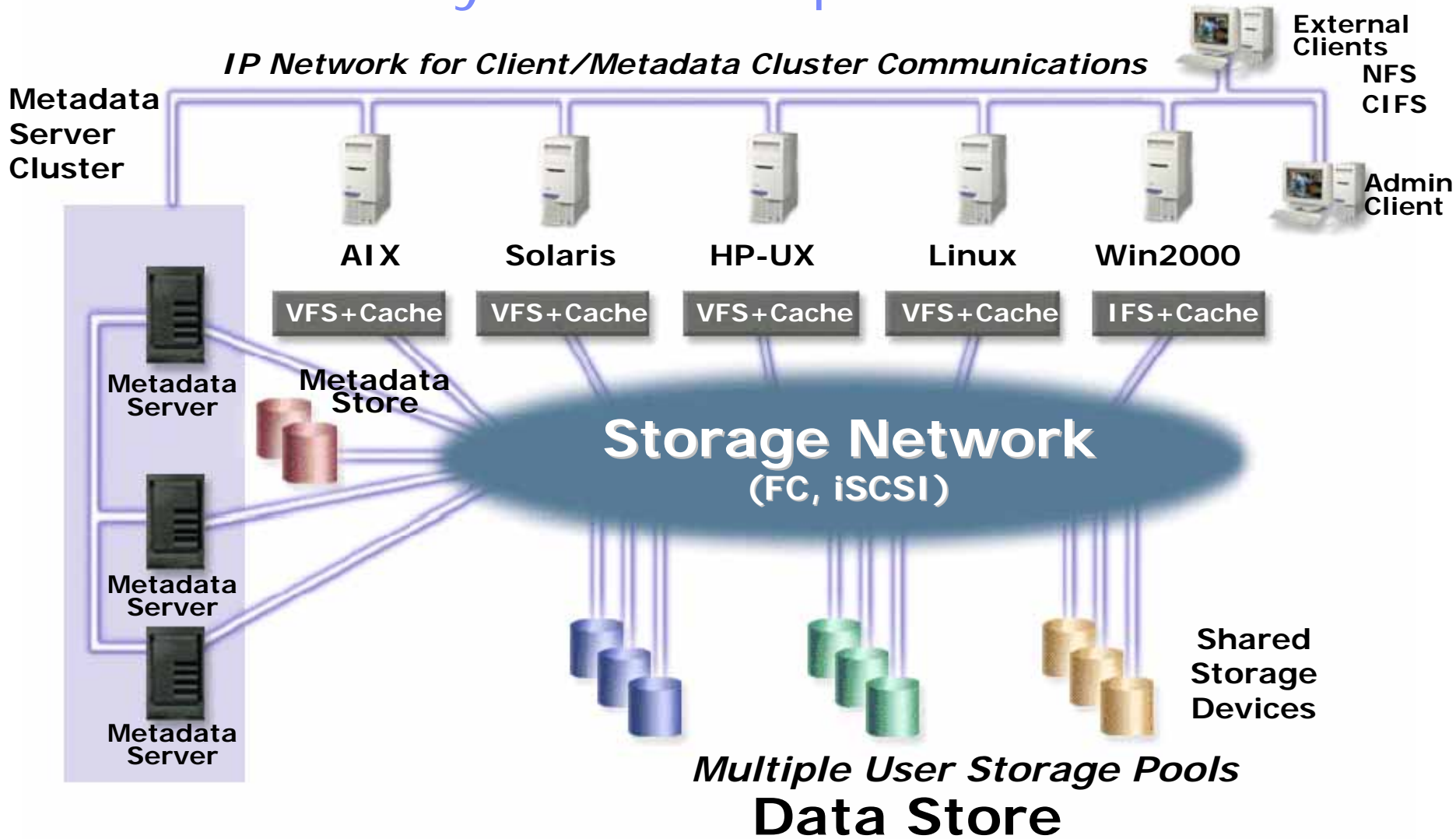


# IBM SFS: Data Flow

An application makes client metadata system requests I/O with the SAN device on a variety of file and response attributes and properties by SFS file performance client system to the MetaSAN server.



# IBM SAN File System: Components



# IBM SFS: Client View of File System

SAN FS appears as a new drive in Windows 2000

Supports Long file names and UNICODE

Supports NTFS Access Control List

SAN FS appears as a new file system in UNIX

"Visible" for native UNIX commands like `df`

does not support `mkfs` or `mkfifo`

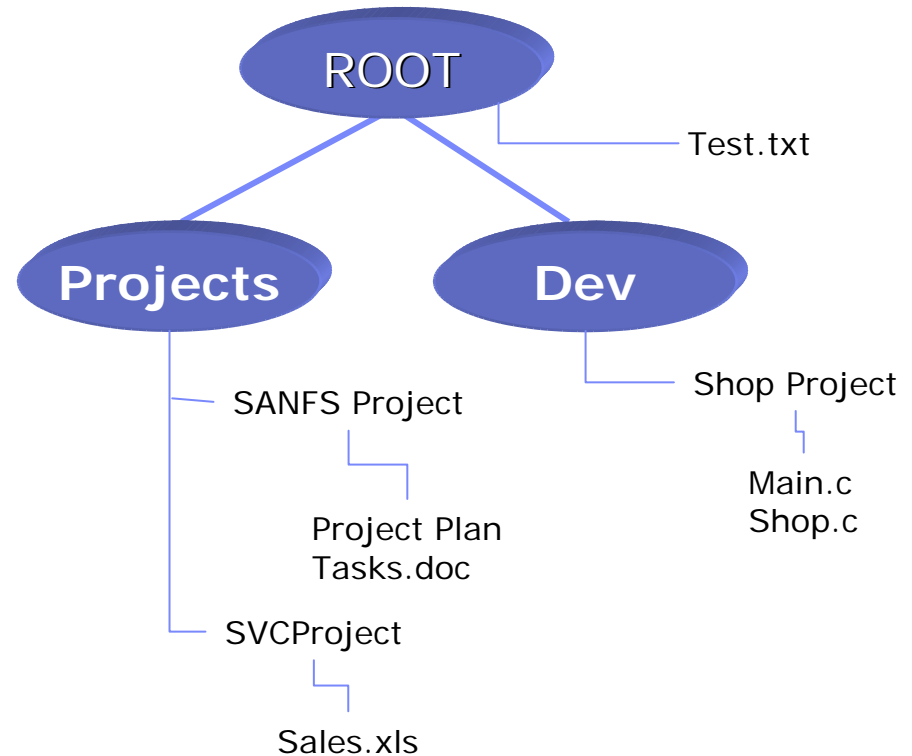


```
# df -k
FileSystem      1024-blocks      Free %Used      Iused %Iused  Mounted on
/dev/hd4         65536           51092   23%       1493    5% /
/dev/hd2        2555904        1275788  51%       29960   5% /usr
/dev/hd9var     65536           54376   18%        679    5% /var
/dev/hd3        65536           63240    4%         48    1% /tmp
/dev/hd1        65536           63404    4%         18    1% /home
/proc           -                -         -          -     - /proc
/dev/hd10opt    65536           55908   15%        387    3% /opt
```

# IBM SFS: Client View of Filesets

Fileset attach points look like normal top-level directories.  
But these directories cannot be deleted or renamed!

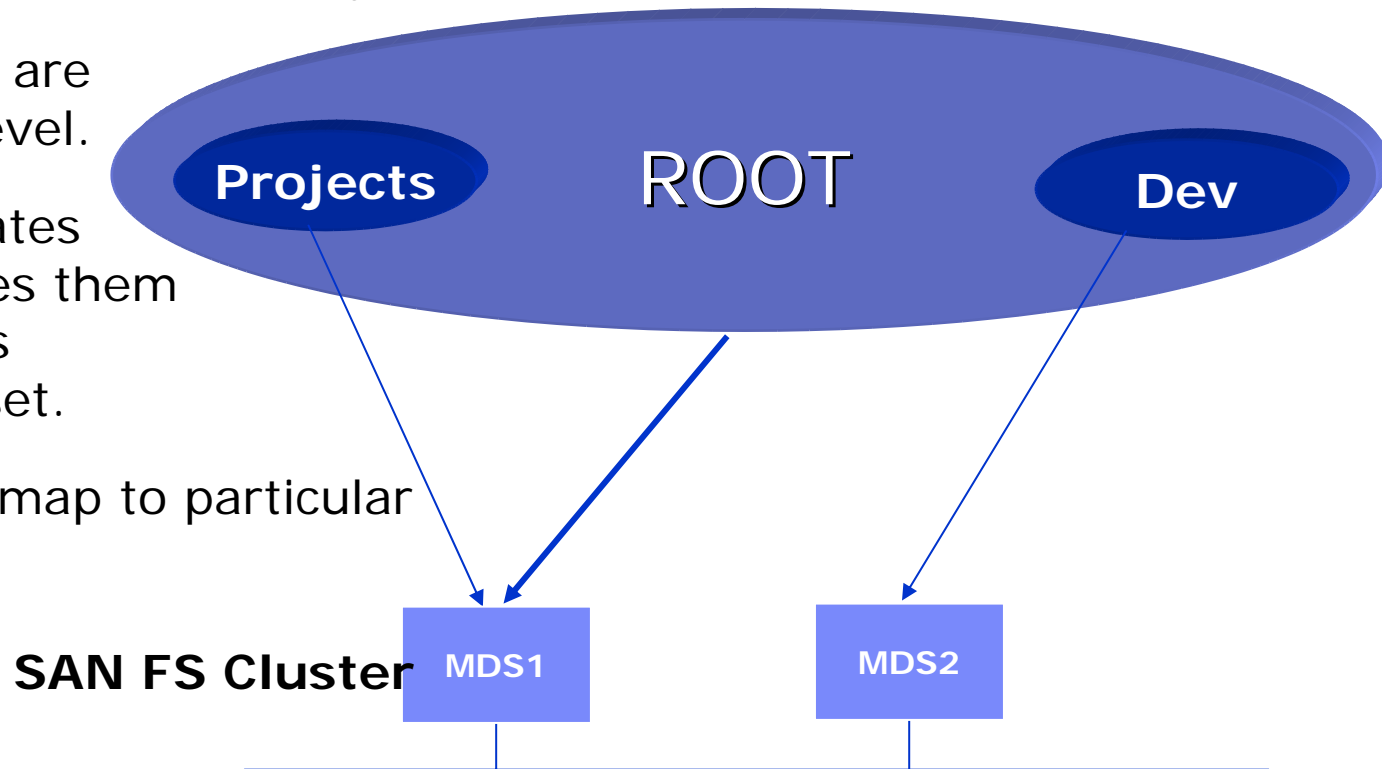
```
T:\>echo hello > test.txt
T:\>dir
Volume in drive T is SANFS
Volume Serial Number is 98E7-6D3C
Directory of T:
10/01/2003 12:26p <DIR> .
10/01/2003 12:26p <DIR> ..
10/01/2003 12:26p <DIR> Projects
10/01/2003 12:31p <DIR> Dev
10/01/2003 12:22p      8 test.txt
          1 File(s)          8 bytes
          4 Dir(s) 7,568,400,384 bytes free
```





## IBM SFS: MDS View of Filesets

- A fileset (except for the root fileset) is a subset of the entire SAN File System global namespace.
- Unit of MDS workload - served by one MDS at a time.
- FlashCopy images are created at fileset level.
- Administrator creates filesets and attaches them at specific locations below the root fileset.
- Do not in general map to particular physical storage.



# IBM SFS: MDS High Availability Features

- MDS failover in case a hardware or software component fails  
automatic workload (fileset) re-distribution included
- Workload re-distribution in case a server is stopped manually
- Automatic exploitation of an N+1 cluster configuration
- Automatic failback when a server is restored
- Non-disruptive manual fileset movement
- Automatic assignment of filesets to servers (optional)
- SFS Quorum Algorithm decides the location of the master.
  - Uses a quorum disk area on a system volume to ensure that all members of newly-formed cluster can access storage.
  - Uses a majority voting method to enable the largest qualified group of servers to form a cluster and elect a master.

## IBM SFS: Storage Pools

- SAN File System uses volumes from one or many storage systems and assigns these volumes to Storage Pools
- Two types of Storage Pools
  - System Pool for SAN FS Config Data and User Metadata (one only)
  - User Pools for data (one by default, typically multiple)
- A set of volumes that provide a desired QoS for a specific use
- Can be expanded or shrunk

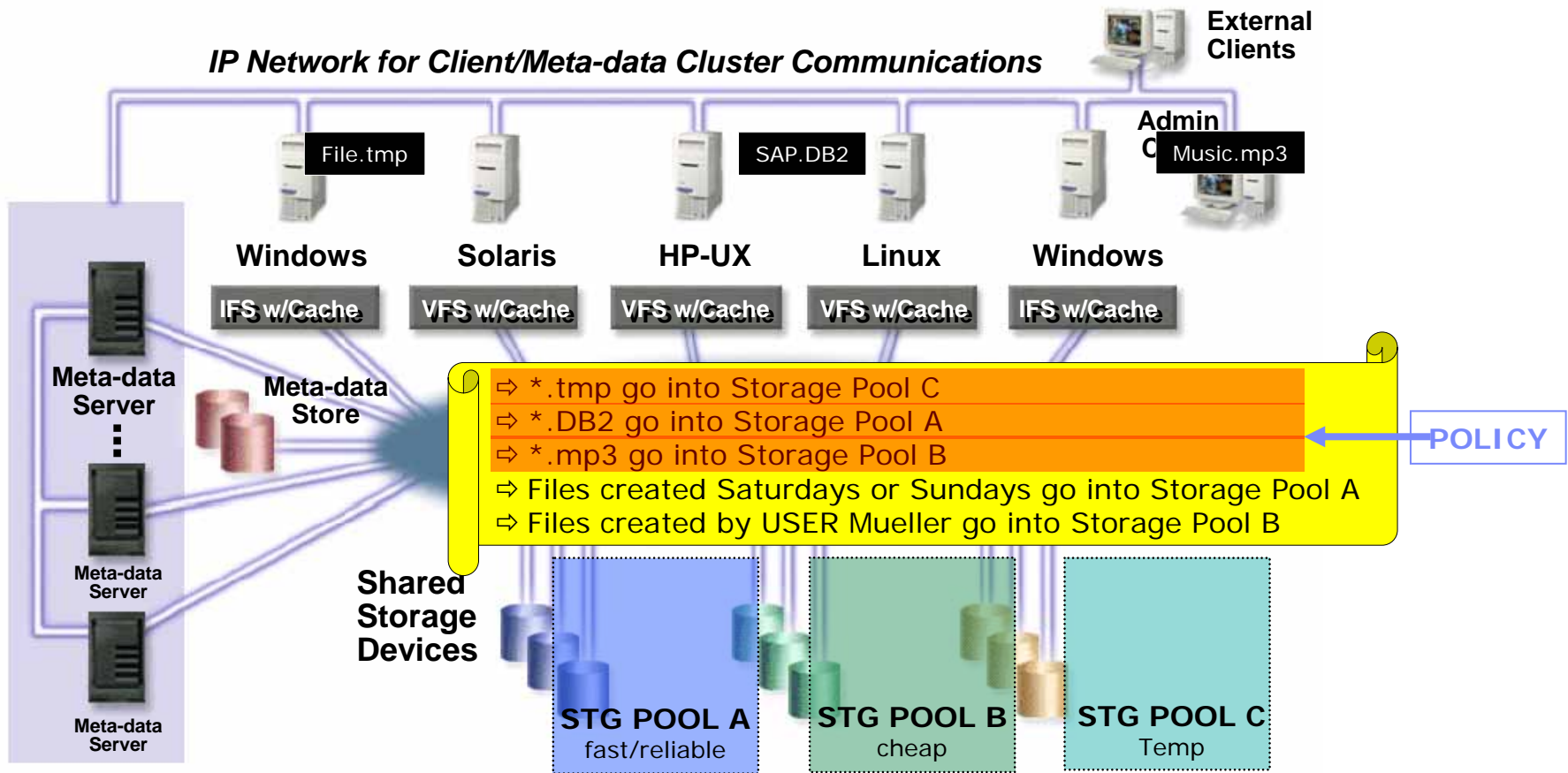


**ONE**



**ONE up to N**

# IBM SFS: Policy-based File Allocation



## Storage Pools

# IBM SFS: Policies and Rules

- Policy-based management uses policy sets and rules.
- A Policy Set is an ordered list of rules.
  - Multiple policy sets can be stored but only one is active.
  - Apply across the MDS cluster.
- Rules specified as conditions on file attributes.
  - Rules use SQL-like language.
  - Files meeting a rule are placed in a pool as specified.
  - No rule applies → file's extents are created in the default pool.
  - Attributes: fileset, file names, extensions, dates, owner
  - Rules evaluated in order of creation.
- Placement only enforced at file creation time.
- Server ensures that the pools and filesets referenced in a policy actually exist when policy is activated.
- Attributes of file at creation time determine placement!

- Applies to migrated data tool, tar extractions, recovery from file-based backup

```
VERSION 1 /* Do not remove or change this line!*/
```

```
RULE 'stgRule1' SET STGPPOOL DBpool WHERE NAME LIKE '%db2%'
RULE 'stgRule2' SET STGPPOOL ImagePool WHERE NAME LIKE '%.jpeg'
RULE 'stgRule4' SET STGPPOOL UNIXUsers FOR FILESET(User1,User2,User3,User4)
RULE 'stgRule3' SET STGPPOOL UNIXSysPool FOR FILESET(UnixSys) WHERE USER_ID <= 100
RULE 'DoW_Sun' SET STGPPOOL Sunday FOR FILESET(fileset1) WHERE DAYOFWEEK(CREATION_DATE)==1
RULE 'DoW_Web' SET STGPPOOL Wednesday FOR FILESET(fileset1) WHERE
DAYOFWEEK(CREATION_DATE)==4
```



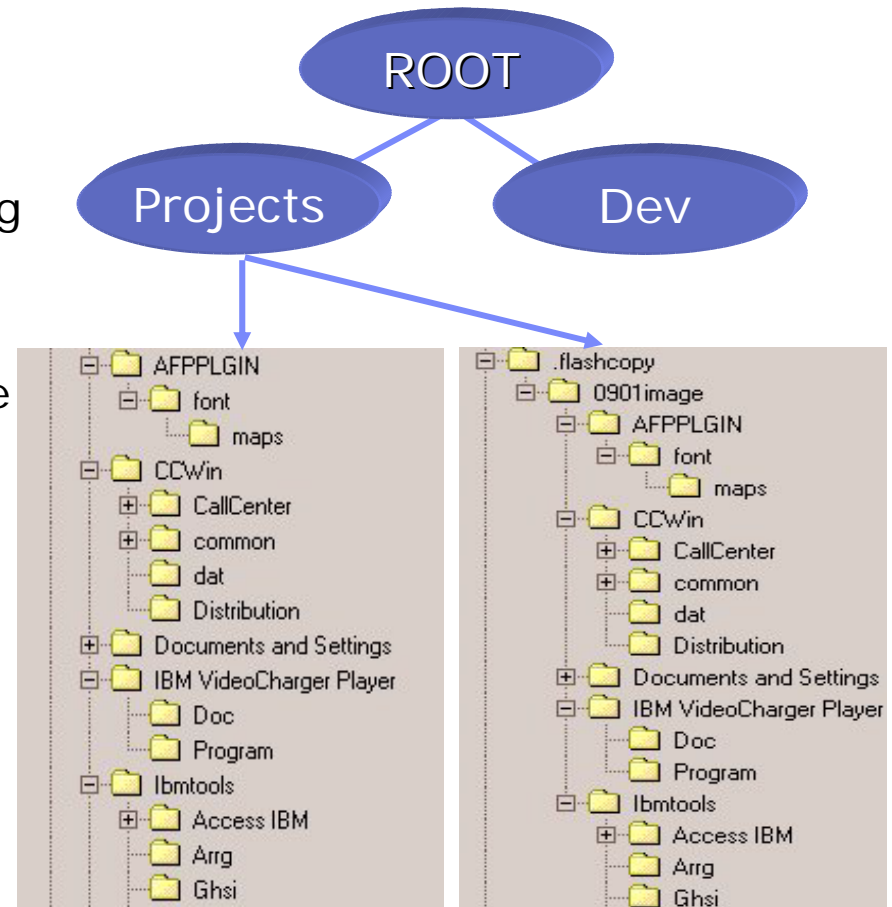
# IBM SFS: Information Lifecycle Management

- Policy-based file placement remains unchanged
- Automated, centralized file management
- Policy-based movement of files between storage pools
- Policy-based deletion of files
- Individually move a file from one storage pool to another
  - No disruption to the application servers accessing the file
  - File is de-fragmented and redistributed
- Policies can be based on pool, fileset, last access date, size criteria

```
RULE ['rulename'] {MIGRATE-FROM-POOL|DELETE-FROM-POOL}
  'sourcepoolname' 'targetpoolname' [FOR FILESET 'filesetname']
  WHERE [AGE operator DAYS] [AND] [SIZE operator {KB|MB|GB}]
```
- Defines these rules in a file management policy, then run a special script to act on the rules
  - Script can be run in a planning mode

# IBM SFS: FlashCopy

- Point-in-time online file-level snapshot taken of a SAN FS Fileset
- Images placed in the special `.flashcopy` directory in the fileset's root
- Images are read-only
- Up to 32 independent images per fileset
- In case of data corruption or loss, image can be reverted to a prior time preventing any system downtime
- Copies use space only for changes
- Each FlashCopy image has a unique name within a fileset (no rename capability)
- FlashCopy images do not support nested filesets (must do each sub-fileset separately)
- Images survive system power loss and reboot
- FlashCopy images inherit the frozen permissions information at the time of the snapshot



# Table of contents

Models and concepts

Disk block virtualization

File and record virtualization

**❖ Storage virtualization in enterprises today**



# Why To Use Storage Virtualization

- The management nightmare
  - Too many servers, operating systems, storage systems, management consoles, ...
  - Too complex policies
  - Too large migration projects
- Availability requirements
- Storage resource utilization

# Enterprise Usage Today

- Widely accepted by customers:
  - Tape drive and library virtualization
  - Disk block level virtualization in LVM and storage subsystem)
  - In-band file system virtualization (NAS)
  - Aggregated networks (iSCSI, long distance connections)
  
- Beginning of the lifecycle:
  - Disk block fabric-layer virtualization (in-band and out-of-band)
  
- Niche markets (this may change):
  - Virtual SANs
  - File and record virtualization

# Why Do Customers Hesitate?

- Reliability
  - SAN and storage failures may affect the whole datacenter.
  - SAN and storage failures may destroy data.
  - Security implications not clear
  
- Complexity
  - Lack of standards
  - Products difficult to understand and evaluate
  - Multi-vendor support
  
- Market situation
  - Religious wars about architectures
  - Solutions from niche players and startups



Storage Systems Division

Closing slide

Thank you