



Education

Storage Virtualization I What, Why, Where and How?

Presenter: Walt Hubis, Storage Architect at Large

Author: Rob Peglar, Isilon / EMC

- The material contained in this tutorial is copyrighted by the SNIA.
- Member companies and individual members may use this material in presentations and literature under the following conditions:
 - ◆ Any slide or slides used must be reproduced in their entirety without modification
 - ◆ The SNIA must be acknowledged as the source of any material used in the body of any document containing material from these presentations.
- This presentation is a project of the SNIA Education Committee.
- Neither the author nor the presenter is an attorney and nothing in this presentation is intended to be, or should be construed as legal advice or an opinion of counsel. If you need legal advice or a legal opinion please contact your attorney.
- The information presented herein represents the author's personal opinion and current understanding of the relevant issues involved. The author, the presenter, and the SNIA do not assume any responsibility or liability for damages arising out of any reliance on or use of this information.

NO WARRANTIES, EXPRESS OR IMPLIED. USE AT YOUR OWN RISK.

➤ Goals of this tutorial:

- ◆ What is storage virtualization?
- ◆ Why do end users need it?
- ◆ Where is it performed?
- ◆ How does it work?

➤ A link to the SNIA Shared Storage Model

➤ The SNIA Storage Virtualization Taxonomy

➤ A survey through various virtualization approaches

➤ Enhanced storage and data services

➤ Q&A

SNIA Shared Storage Model *A Layered View*

IV. Application

(e.g. NAS, CIFS/NFS)

III. File/record layer

IIIa. Database

IIIb. File system

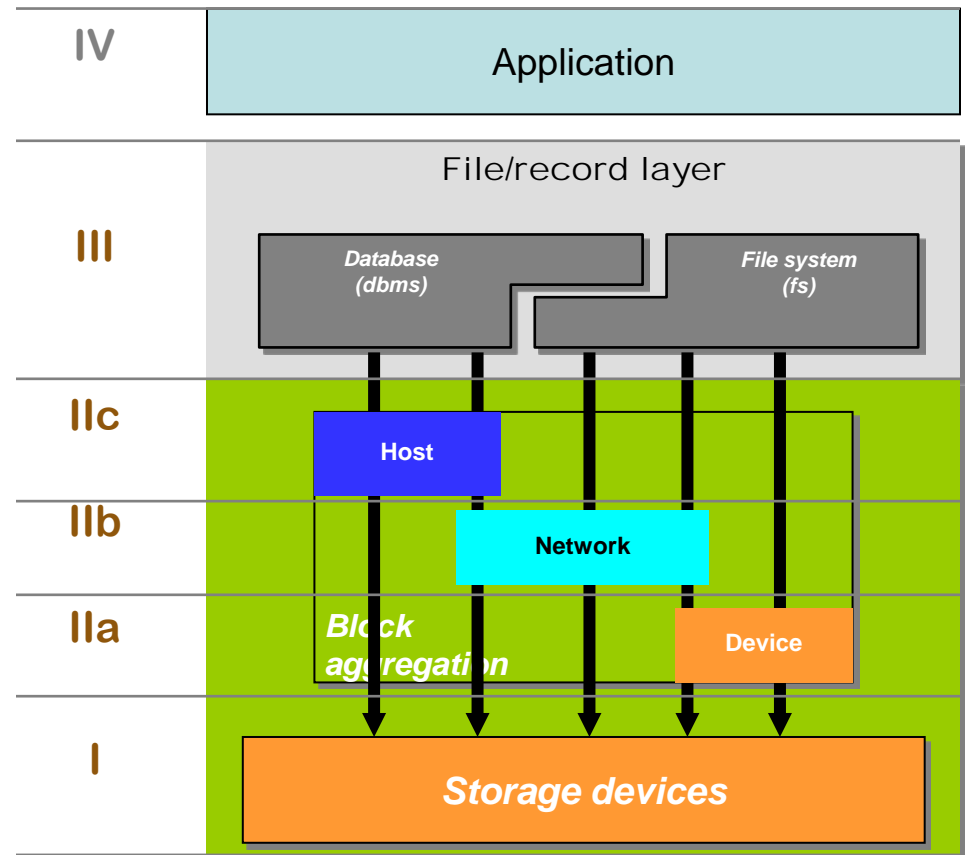
II. Block aggregation

IIa. Host

IIb. Network

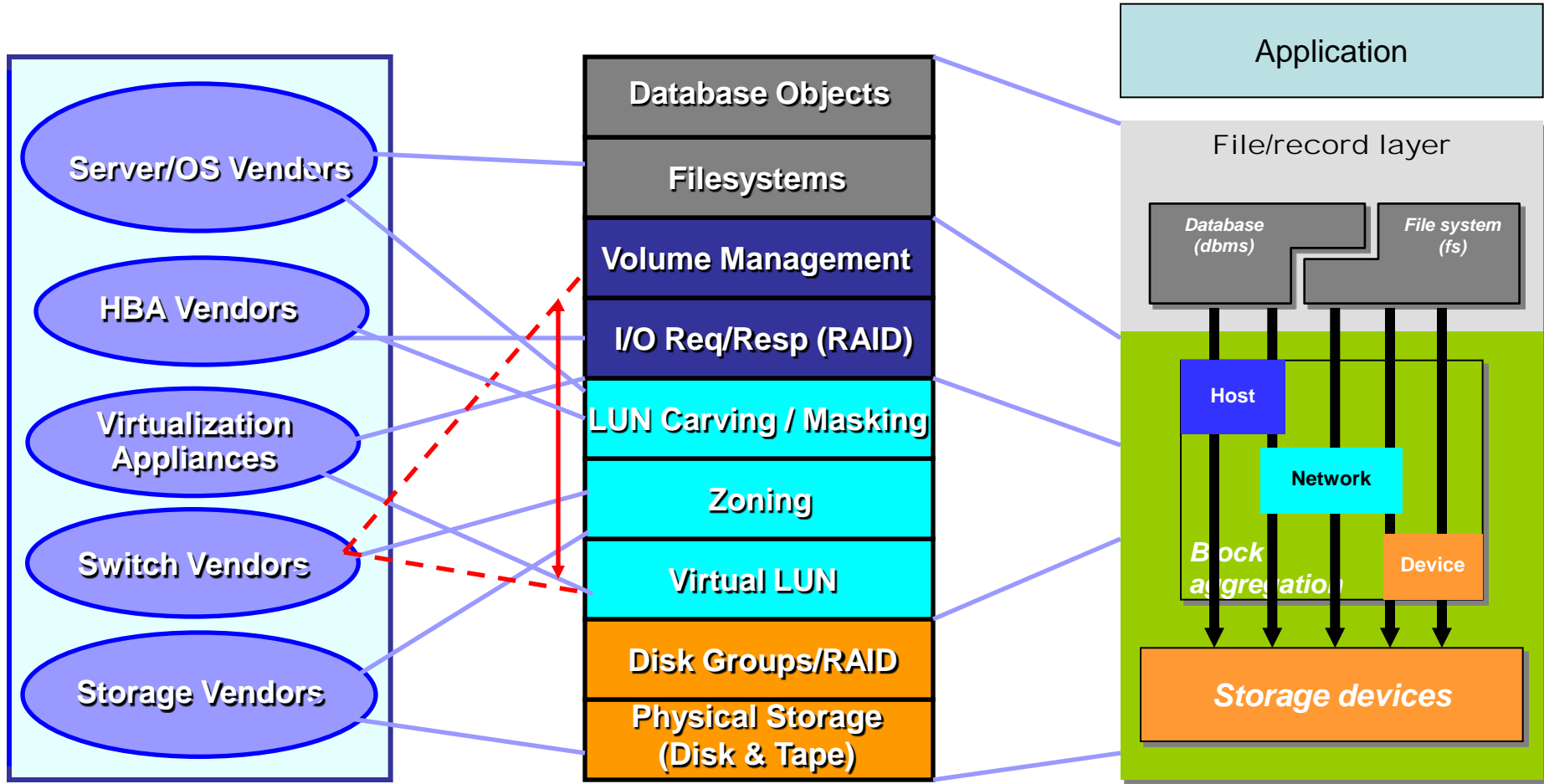
IIc. Device

I. Storage devices



The SNIA Shared Storage Model uses the term “aggregation” instead of “virtualization”

Differentiation *Virtualizing the Storage Stack*



Stack Coverage Expansion – Everybody wants a piece of the pie!

So What's the Problem?

- **The MANAGEMENT nightmare**
 - ◆ Too many different
 - › Servers – now both physical and virtual
 - › Operating systems/Hypervisors
 - › Switching systems
 - › Storage systems and protocols
 - › Management consoles
 - ◆ IT staff skill levels and budget (the lack thereof)
- **Availability requirements driven by e-business**
 - ◆ 24x7 for applications when needed (some 24x7xforever)
 - ◆ Zero tolerance for downtime – planned or unplanned
- **Typical (non-virtualized) storage utilization**
 - ◆ Disk: 30 - 50%
 - ◆ Tape: 20 - 40%

- Storage is physical
 - ◆ Connections & Presentation
 - ◆ Power & Cooling
 - ◆ Access and Configuration
 - ◆ Results in: Complexity, Reboots, Downtime, \$\$\$
- Multiple management systems - complex
 - ◆ Inconsistent
 - ◆ Incompatible
 - ◆ Incomplete
- Result: ever-increasing storage management costs
- Can't support today's rapid data growth

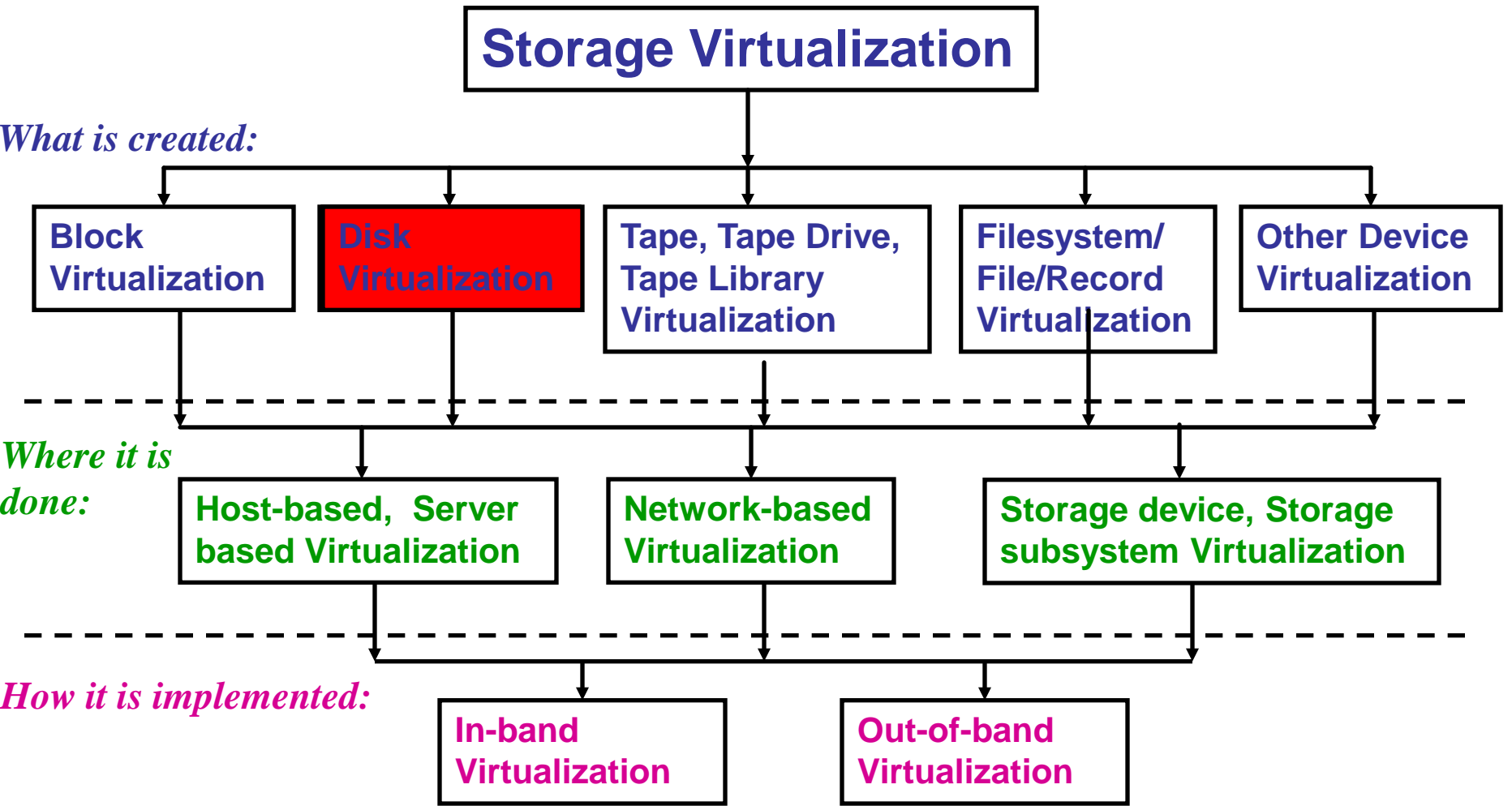
What is Storage Virtualization?

- An **abstraction of detail** that **separates layers**
 - ◆ Host implementation (Application, OS, HBA)
 - ◆ Network implementation (Switch, Router, Gateway)
 - ◆ Storage implementation (Array, Library, Device)
- **Makes invisible to host:**
 - ◆ physical pathing
 - ◆ device characteristics
 - ◆ physical data location
- **Provides Location and Implementation Transparency**
- **Enables Dynamic Operations**
 - ◆ Enables transparent “on the fly” reconfiguration
 - ◆ Allow data location to change transparently to host environment
- *There are many different types, approaches and degrees of storage virtualization*

Benefits of Storage Virtualization

- Openness to new server, network and storage technology
 - ◆ Especially virtual server/hypervisor/metaOS technologies
- Significantly reduced downtime – planned and unplanned
- Increased storage asset utilization
 - ◆ Reduced **power/cooling/space** inputs
 - ◆ Reduced storage capital cost
 - ◆ Reduced management complexity
- (Potentially) Improved performance
 - ◆ Load spreading, balancing, multi-pathing, heuristic shifting
- Dynamic provisioning (on-demand, elastic, cloud)
- Must-Have Architecture – now and into the future
 - ◆ Increased Scalability, Security, Flexibility
 - ◆ Managed file systems and volume managers
- Simplify definition of storage policies and procedures
- Improve delivery and quality of Storage Services

SNIA Storage Virtualization Taxonomy



Disc (Drive) Virtualization

Physical disc drive



Disc Drive

Disk Virtualization

LBA

| |
|--------|
| 000 |
| 001 |
| 002 |
| 003 |
| 004 |
| 005 |
| 006 |
| .. nnn |

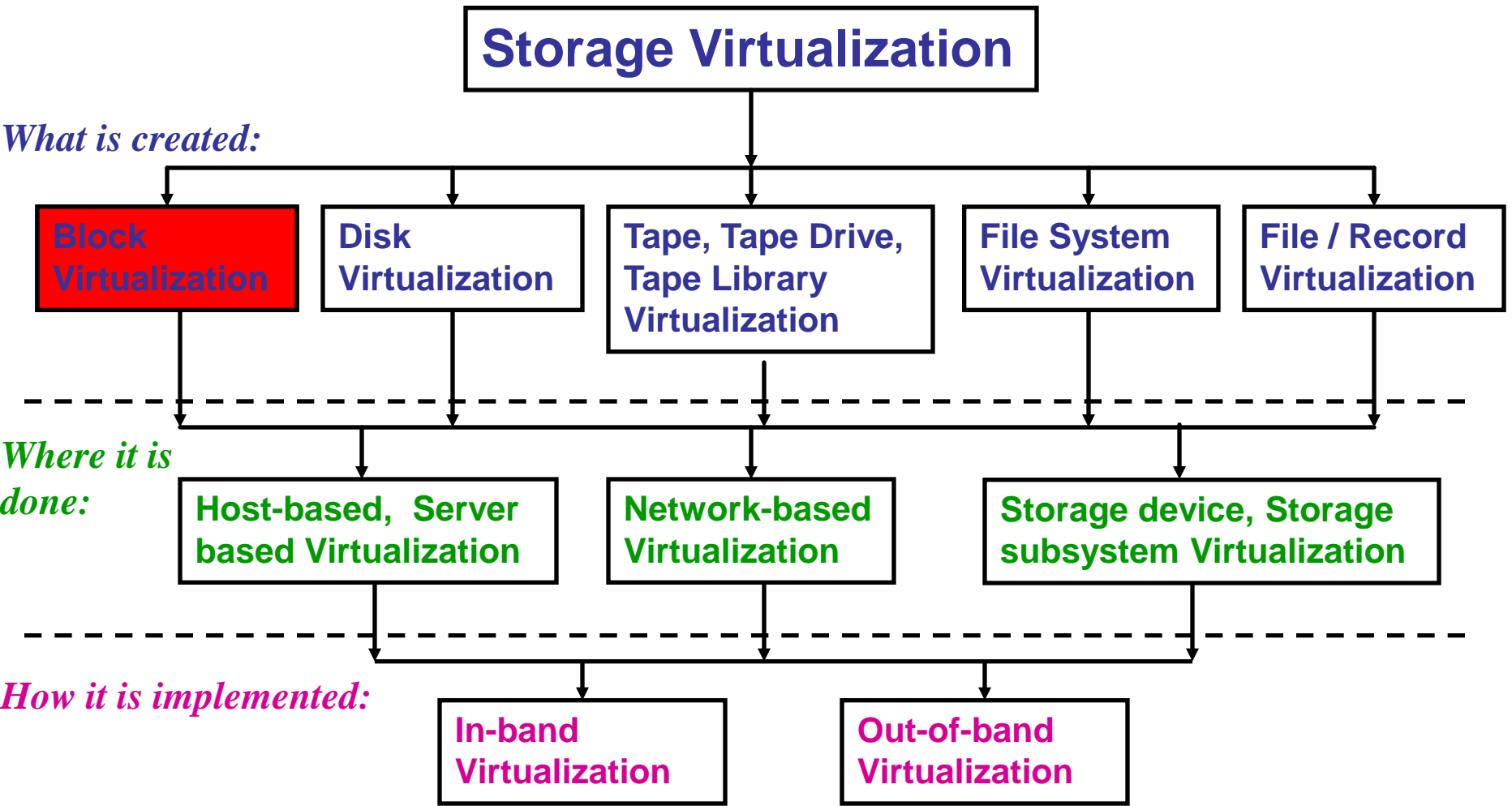
Physical data layout

- C-H-S Addresses
- Media defects

Logical data layout

- Logical Block Addresses (LBA)
- 'Defect-Free'

SNIA Storage Virtualization Taxonomy



What functionality do users need?

Application aspects of storage

- **Capacity**
 - ◆ Application requirements
 - ◆ Structured / unstructured
 - ◆ Growth potential
- **Performance**
 - ◆ Throughput / IOPS
 - ◆ Responsiveness
- **Availability**
 - ◆ Failure resistance
 - ◆ Recovery time/point
 - › **RTO/RPO**
 - ◆ Simplification of change

Physical aspects of storage

- **Capacity**
 - ◆ Disk or Tape Size
 - ◆ Number of disks/channel
 - ◆ Number of tape devices
- **Performance**
 - ◆ Disk latency & seek time
 - ◆ Cache util %, size & hit rate
 - ◆ Media rotation rate (RPM)
 - ◆ Responsiveness
- **Availability**
 - ◆ **MTBF/MTTR (Rebuild time)**
 - ◆ Path redundancy
 - ◆ Path bandwidth

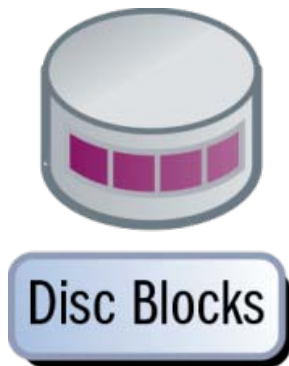
➤ Physical disks



- ◆ Fixed size
- ◆ Bounded performance
- ◆ Do break (occasionally)

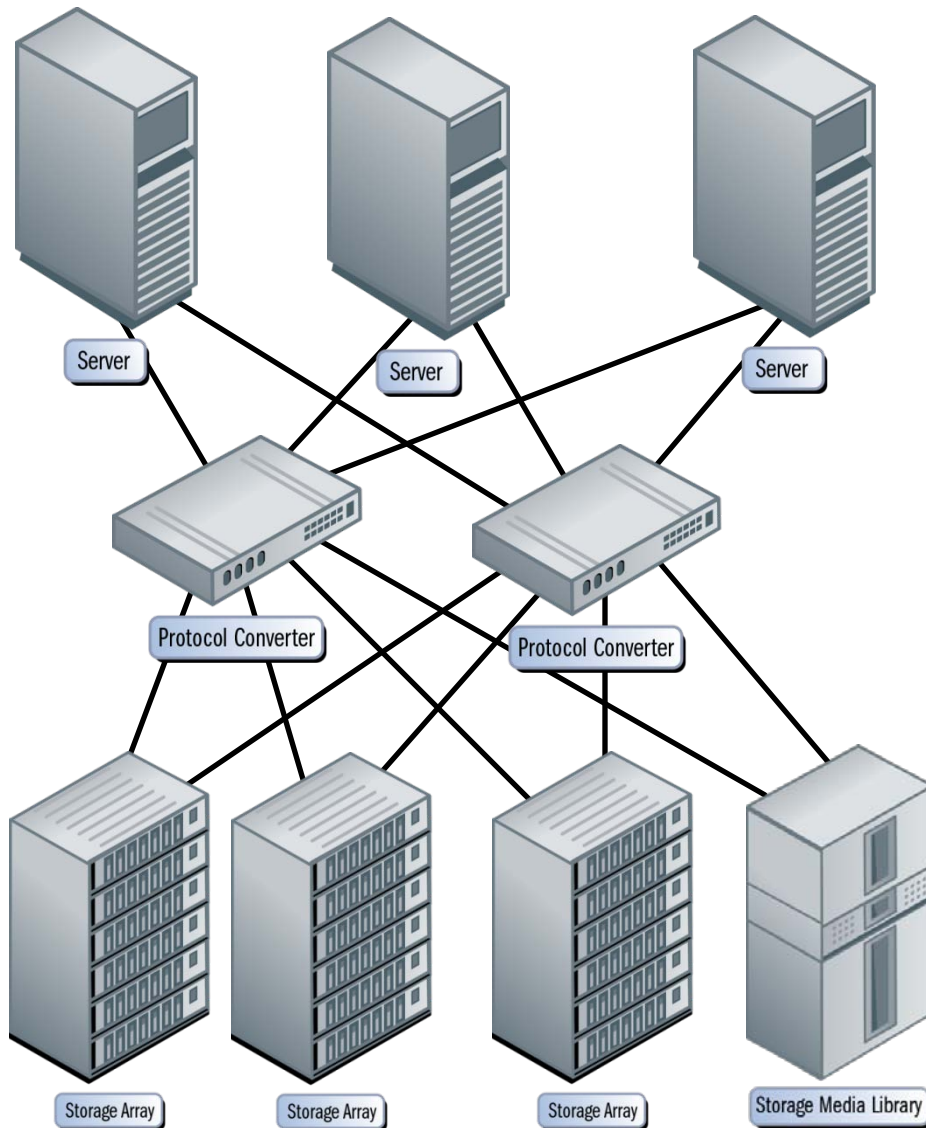
**Block-level
Virtualization**

➤ Virtual disks



- ◆ As large, small or as many as users need
- ◆ Performance scaling up or down
- ◆ As reliable as users and applications need
- ◆ Can grow, shrink or morph

Where Does Virtualization Reside?

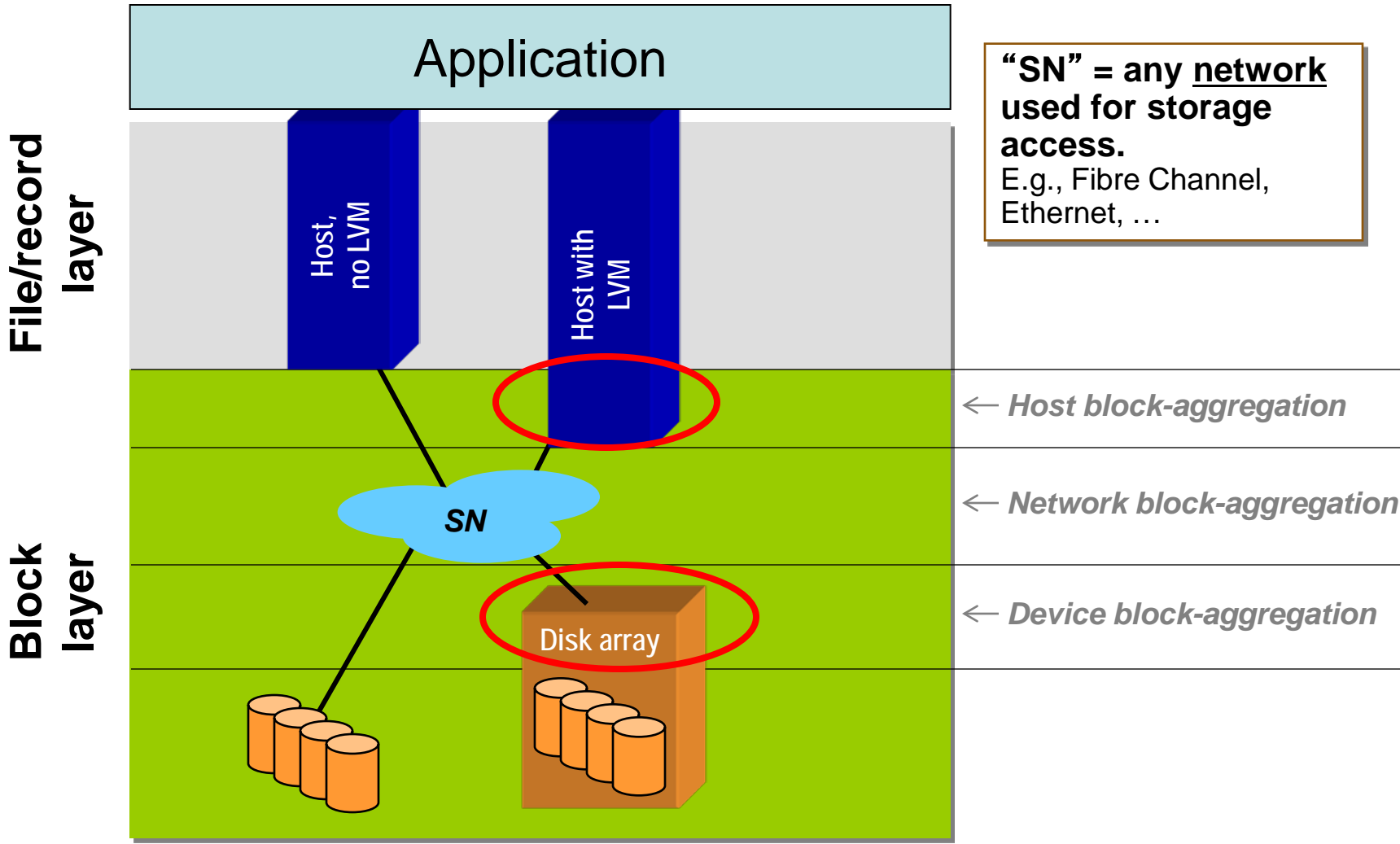


Host?

Network?

**Storage
Device?**

SNIA Shared Storage Model: SAN block storage



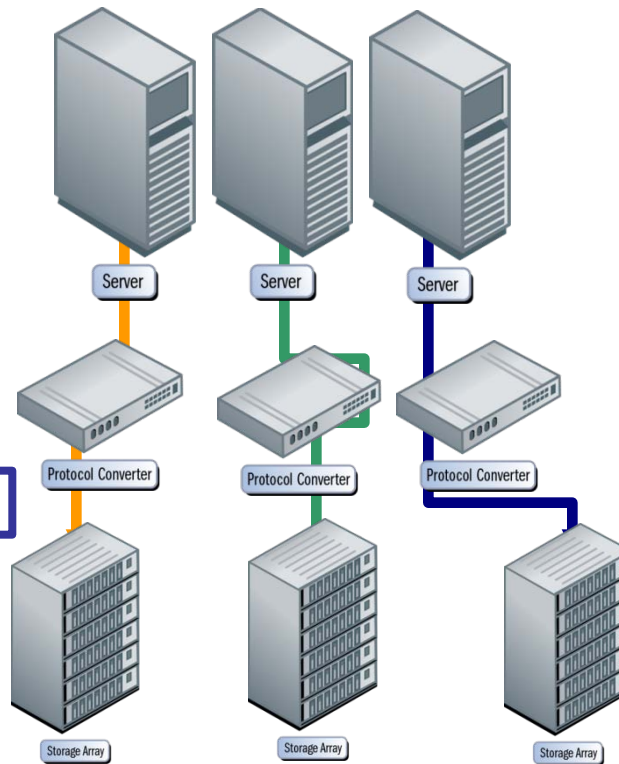
Subsystem-based Virtualization

Provisioning
*Per-host &
storage subsystem*



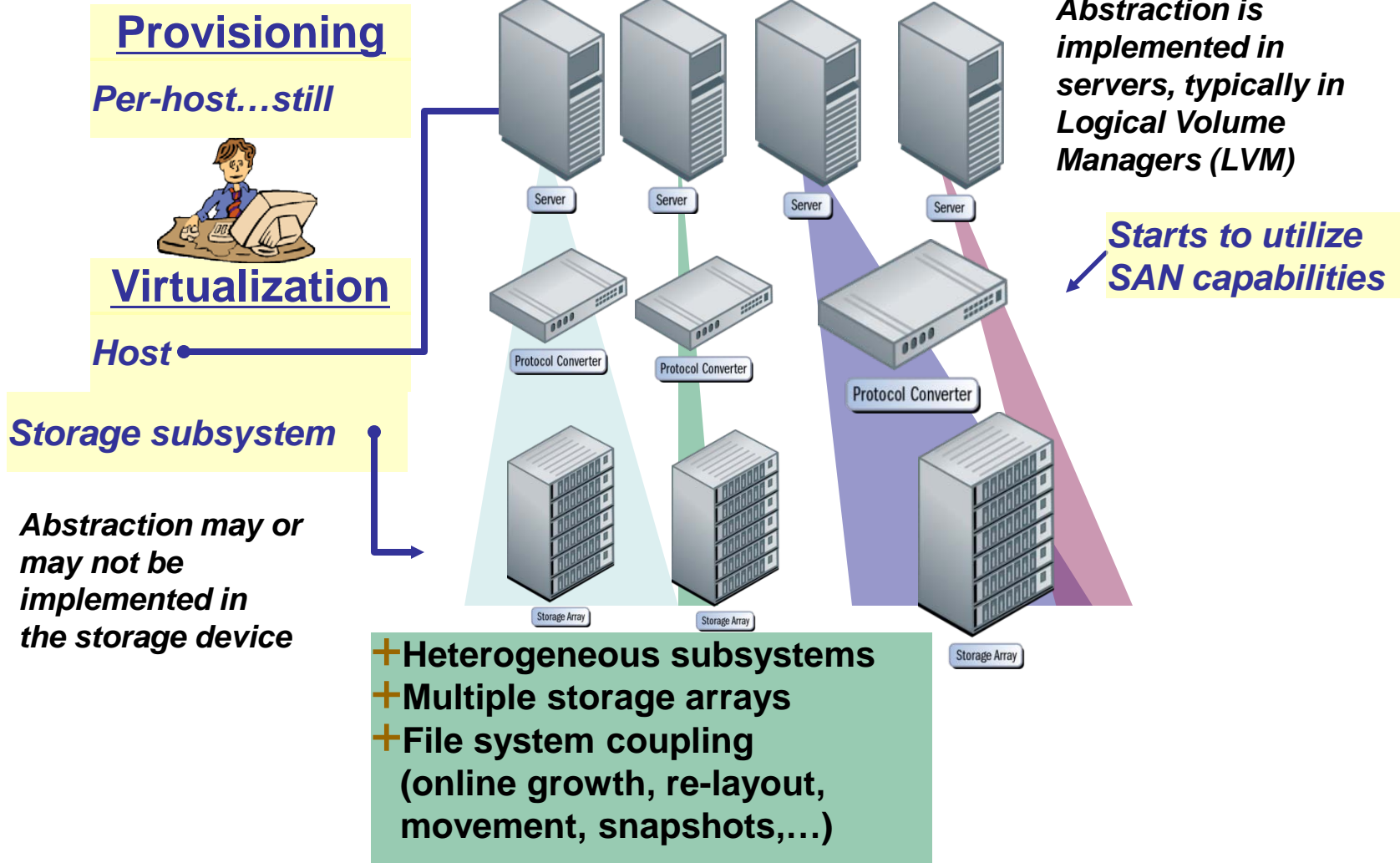
Virtualization
Storage subsystem

*Abstraction is
implemented in
the storage device*

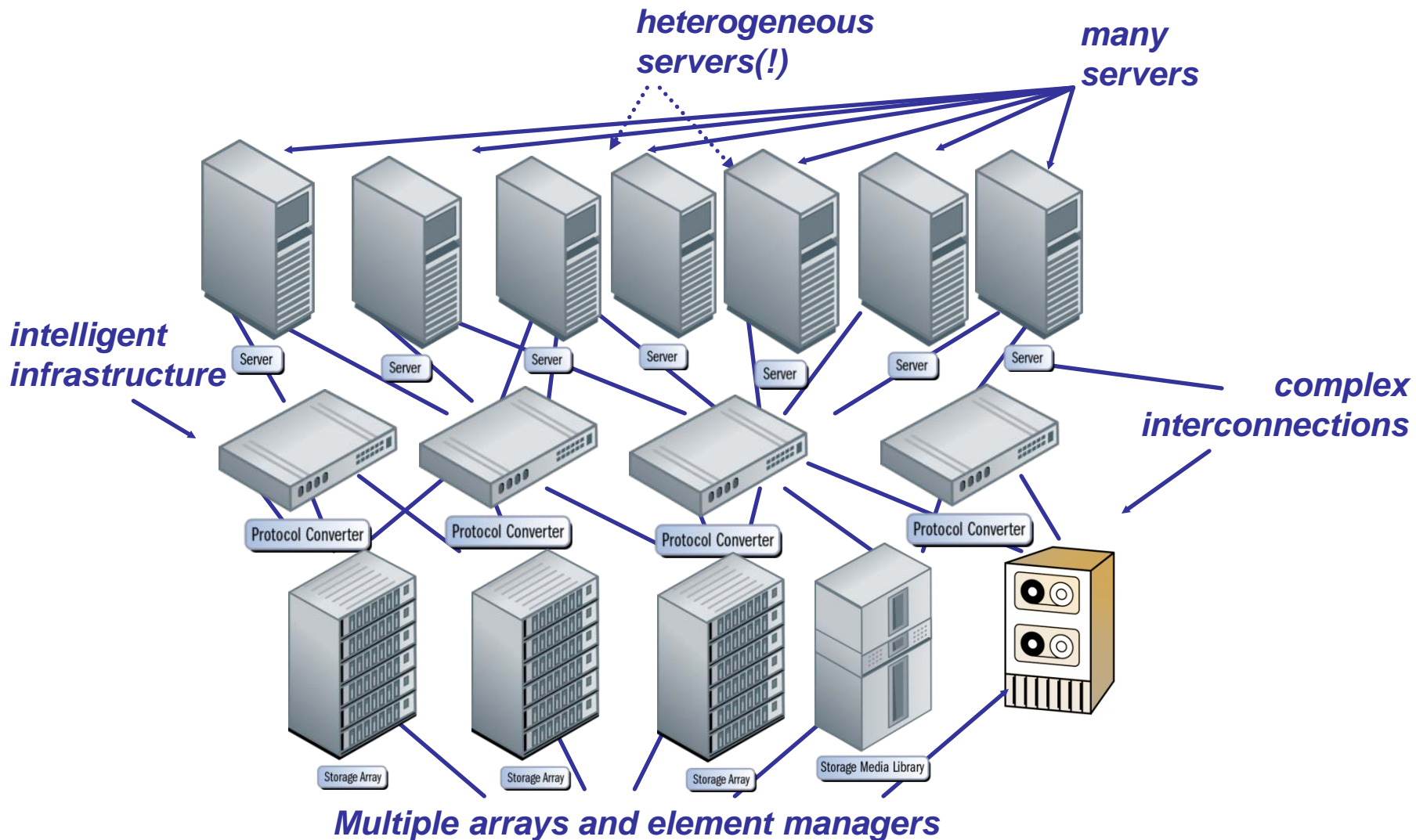


- + Heterogeneous hosts
- + Tiered Storage
- + Mature industry & products
 - Performance
 - Stable & reliable
 - Security less of a concern

Host-based Virtualization



SANs provide a complex infrastructure

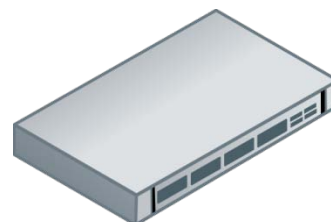




Server

Server-based Device (Appliance)

- + Virtualize a variety of physical storage using various HBAs
- + Implement complex storage services inexpensively
- + FC N_Port functionality
- + iSCSI port functionality



FC Switch

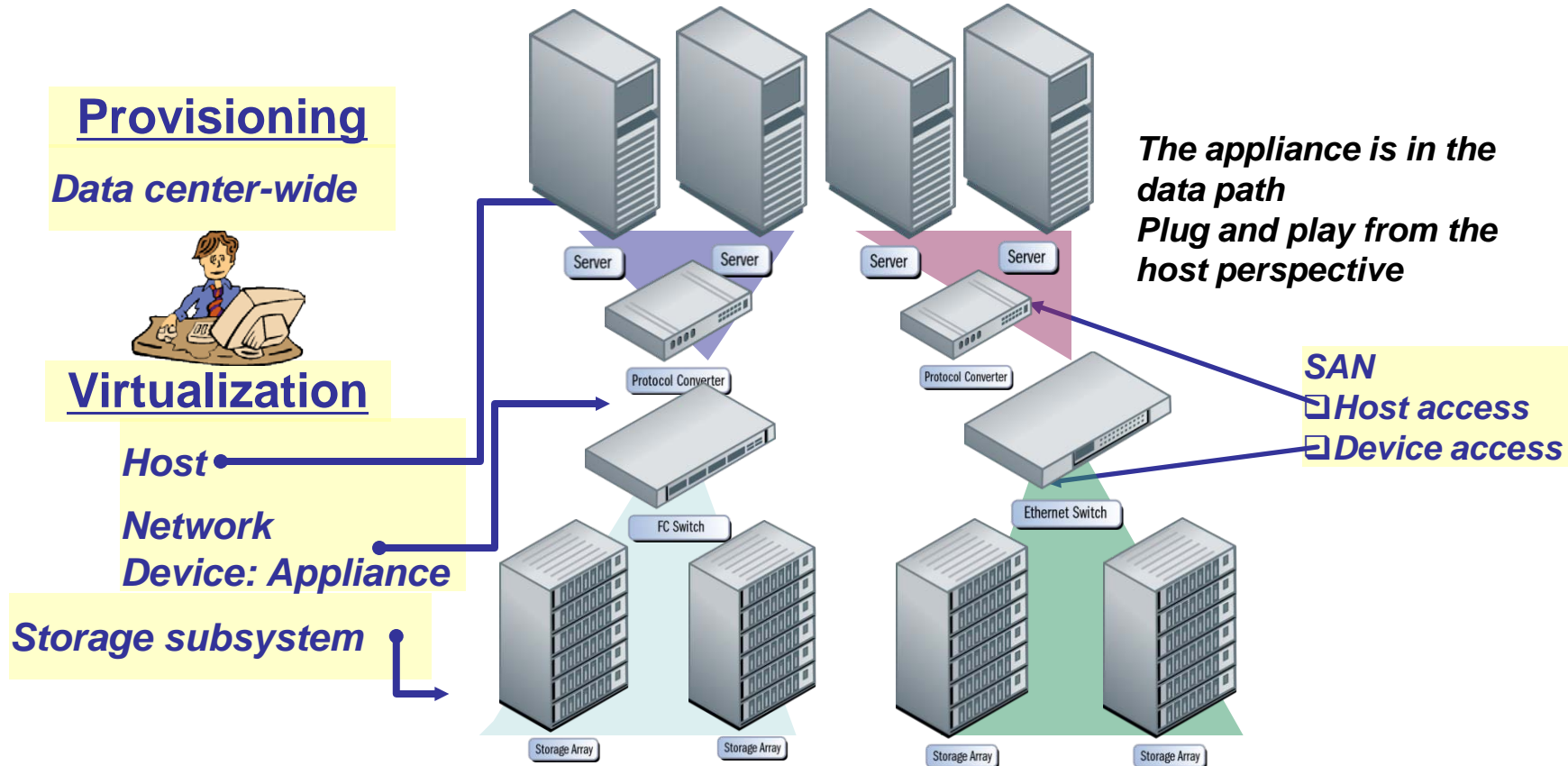


Ethernet Switch

Switch-based Device

- + Network optimized
- + High port counts
- + FC N_Port, FL_port, F_Port or E_Port functionality
- + iSCSI port functionality

Virtualization in the network: *In-band with appliances*



- + Data center-wide management**
 - Heterogeneous storage
 - Heterogeneous hosts
 - One pool per storage media
 - Caching potential in the network

Virtualization in the network: *In-band with switches*

Provisioning

Data center-wide



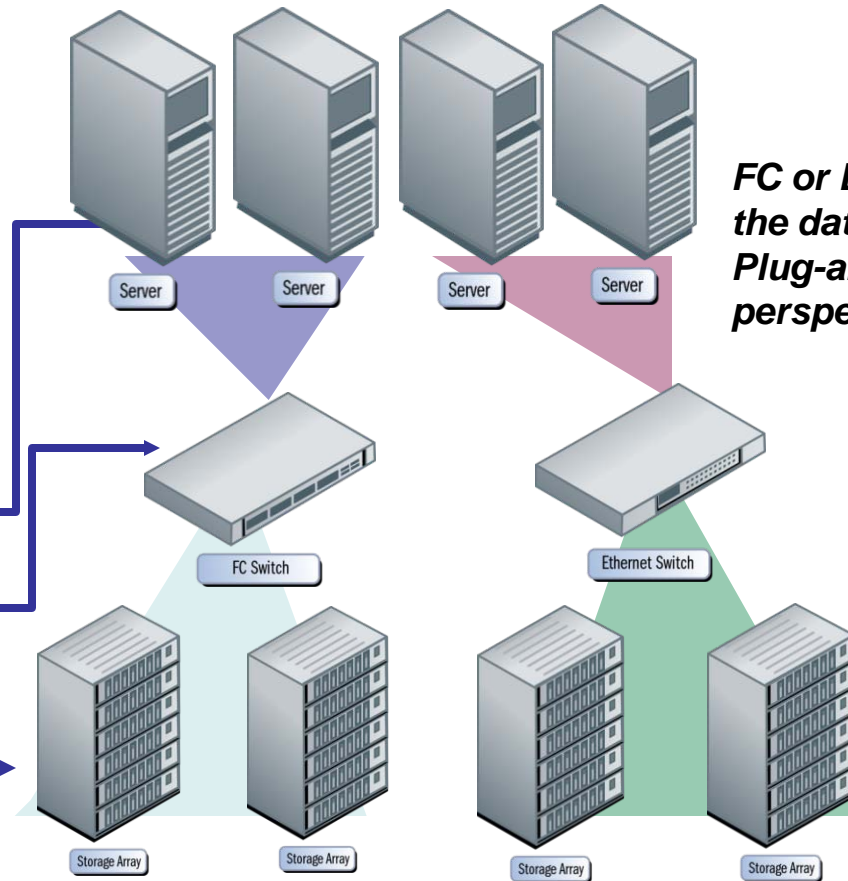
Virtualization

Host

Network

Device: Switch

RAID subsystem




*FC or Ethernet switch in the data path
Plug-and-play from host perspective*

+ Data center-wide management

- Heterogeneous storage
- Heterogeneous hosts

Virtualization in the network: *Out-of-band with appliances*

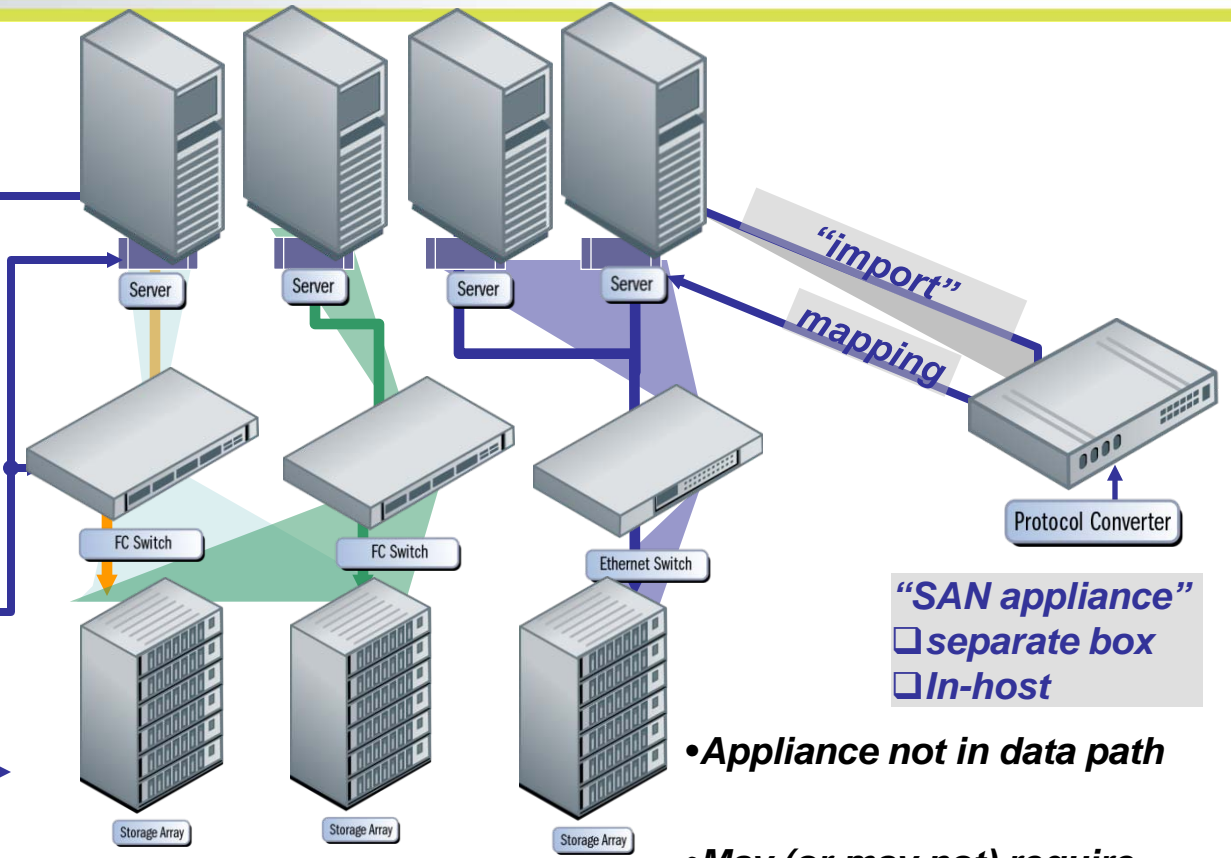
Provisioning
Data center-wide



Virtualization
Host

Network

RAID subsystem



"SAN appliance"
 separate box
 In-host

- + Data center-wide management
- + Shorter data I/O path, but more complex interactions
- + Light-weight compared to full volume manager

- Appliance not in data path
- May (or may not) require agent software on each host
- Separates the data from the control path

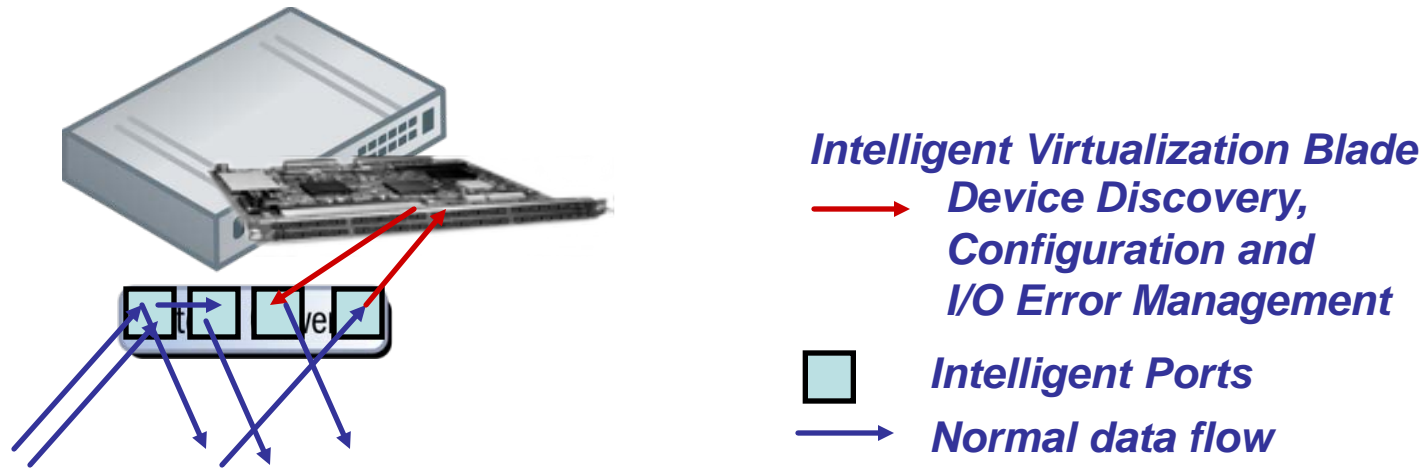
| Comparison | Appliance-based | Switch-based |
|---------------------|---------------------------|-----------------------------------|
| Multi-vendor fabric | Independent functionality | Interoperability mode |
| Switching | Separate ¹ | Integrated |
| Performance | Read and write caching | No store-and-forward ² |
| Functionality | Rich feature set possible | Cost & footprint limits |
| Availability | Fail-over mechanisms | Fabric topology |
| Connectivity | Usually HBA / NIC ports | High density switch ports |
| Scalability | Implementation specific | Implementation specific |
| Storage ROI | Leverage legacy storage | SAN-attached storage |
| Maturity | Stable since 2002 | Stable since 2005 |

1: Some in-band appliances can also perform the switching function.

2: Some intelligent switches actually use a store-and-forward approach, where virtualization is not integrated directly with the data switching.

Switch-based Virtualization: A Closer Look

- A closer look inside the “smart switch”:



- “Smart switch” has the components of a hybrid approach
 - **Metadata Controller** = Virtualization engine for device discovery, volume configuration and I/O error management (“bad path”)
 - **Data Controller** = Intelligent Ports (based on ASICs) provide the virtual/physical I/O translation and forwarding of data to the proper targets (“good path”)

➤ **Problem:**

- ◆ Complex architecture within intelligent switches and other intelligent platforms
- ◆ May lower the implementation speed of management applications
- ◆ Several proprietary approaches by several different vendors

➤ **Solution:**

- ◆ ANSI T11 FAIS (Fabric Application Interface Standard)
- ◆ A set of APIs with a library of managed objects
- ◆ “*easily migrate*” host-or array-based services to intelligent networking platforms

➤ **Functionality of FAIS:**

- ◆ Split data and control path
- ◆ Provide Volume Management
 - › Virtual to physical I/O translation
- ◆ Copy Services such as Snapshots, Mirroring and Data Replication

➤ **T11 FAIS and SNIA SMI-S are complimentary standards**

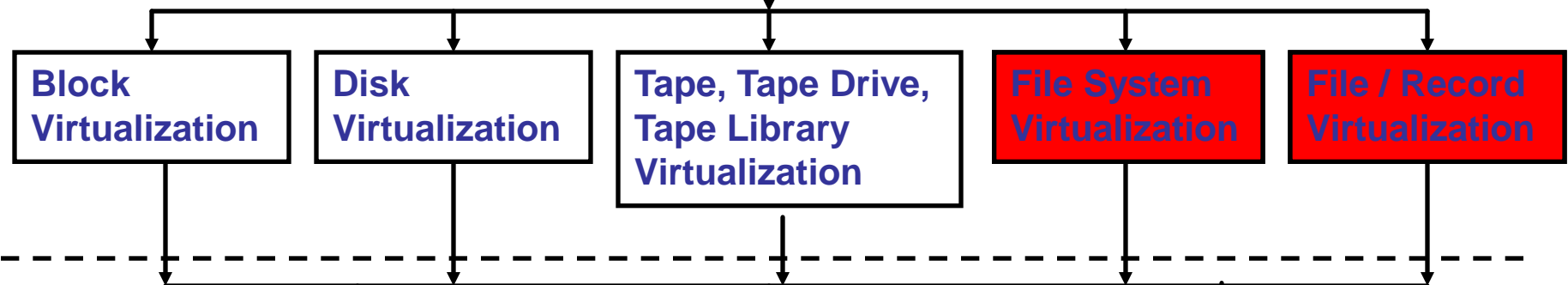
- ◆ FAIS - API on switching platform for services to exploit switch-based capabilities
- ◆ SMI-S - API for managing storage (including services that are switch-based)

Quick Virtualization Comparison

| Virtualization Level | Pros | Cons |
|----------------------|--|---|
| Host-Based | Subsystem independence Close to the Filesystem Use OS-built-in tools No array controller cycles | OS dependence HW dependence (maybe) Use OS-built-in-tools Use host CPU cycles |
| Network-Based | Subsystem independence Host independence No host CPU cycles Choice of band (in,out) | Switch dependence (maybe) Uses switch cycles Choice of band (in,out) |
| Subsystem-Based | Host independence Close to the devices No host CPU cycles Mature technology | Array dependence Far from the filesystem Uses controller cycles Specialized training (maybe) |

Storage Virtualization

What is created:



Where it is done:



How it is implemented:



➤ File / Record Virtualization

- ◆ Presents one or more underlying objects as a single composite object
 - › Objects can be files or directories
- ◆ Can provide HSM like properties in a storage system
- ◆ Presents an integrated file interface
 - › file data and metadata are managed separately in the storage system

➤ File System Virtualization

- ◆ Aggregates multiple file systems into one large “virtual file system”
- ◆ Virtual file systems may be implemented in addition to physical file systems
- ◆ Users access data through the virtual file system
- ◆ Underlying file systems transparent to users
- ◆ Enables additional functionality
 - › different file access protocol
 - › on top of one or more existing file systems



Check out Tutorial: File Systems for Object Storage

Storage Virtualization

What is created:



Where it is done:



How it is implemented:



➤ Tape Media Virtualization

- ◆ Resolves the problem of underutilized tape media
- ◆ Data written to tape at disk cache speed, reduces mounts
- ◆ Saves tapes, tape libraries and floor space

➤ Tape Drive & Library Virtualization (VTL)

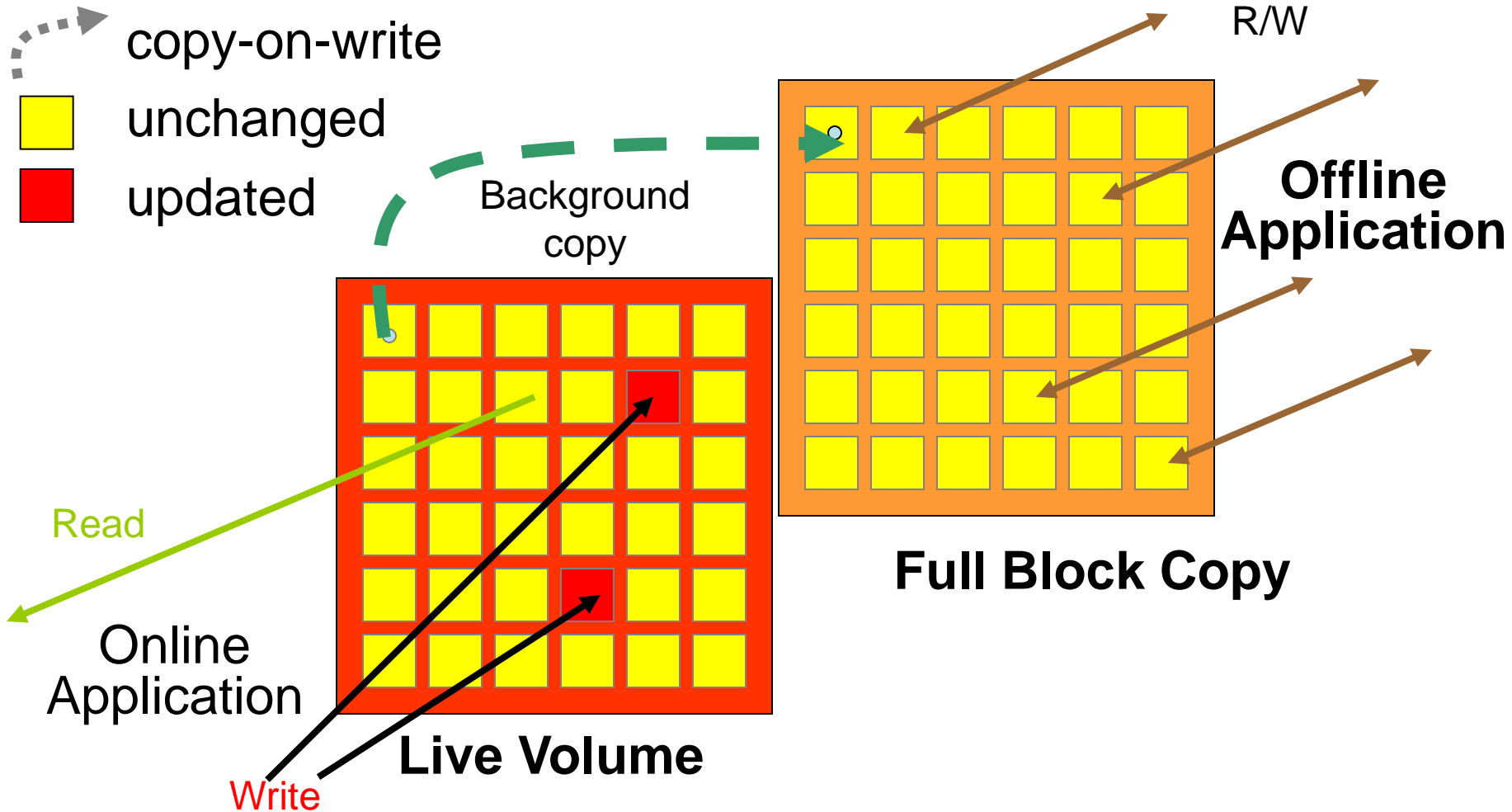
- ◆ Shares tape drives and libraries among a number of servers
- ◆ Less tape drives/libraries required
- ◆ Help to justify use of enterprise-class tape drives
- ◆ Improved error handling
- ◆ Reduced complexity
- ◆ No change to backup application or IT processes
- ◆ Potential for data reduction
 - › Dedup, compression, incrementalization



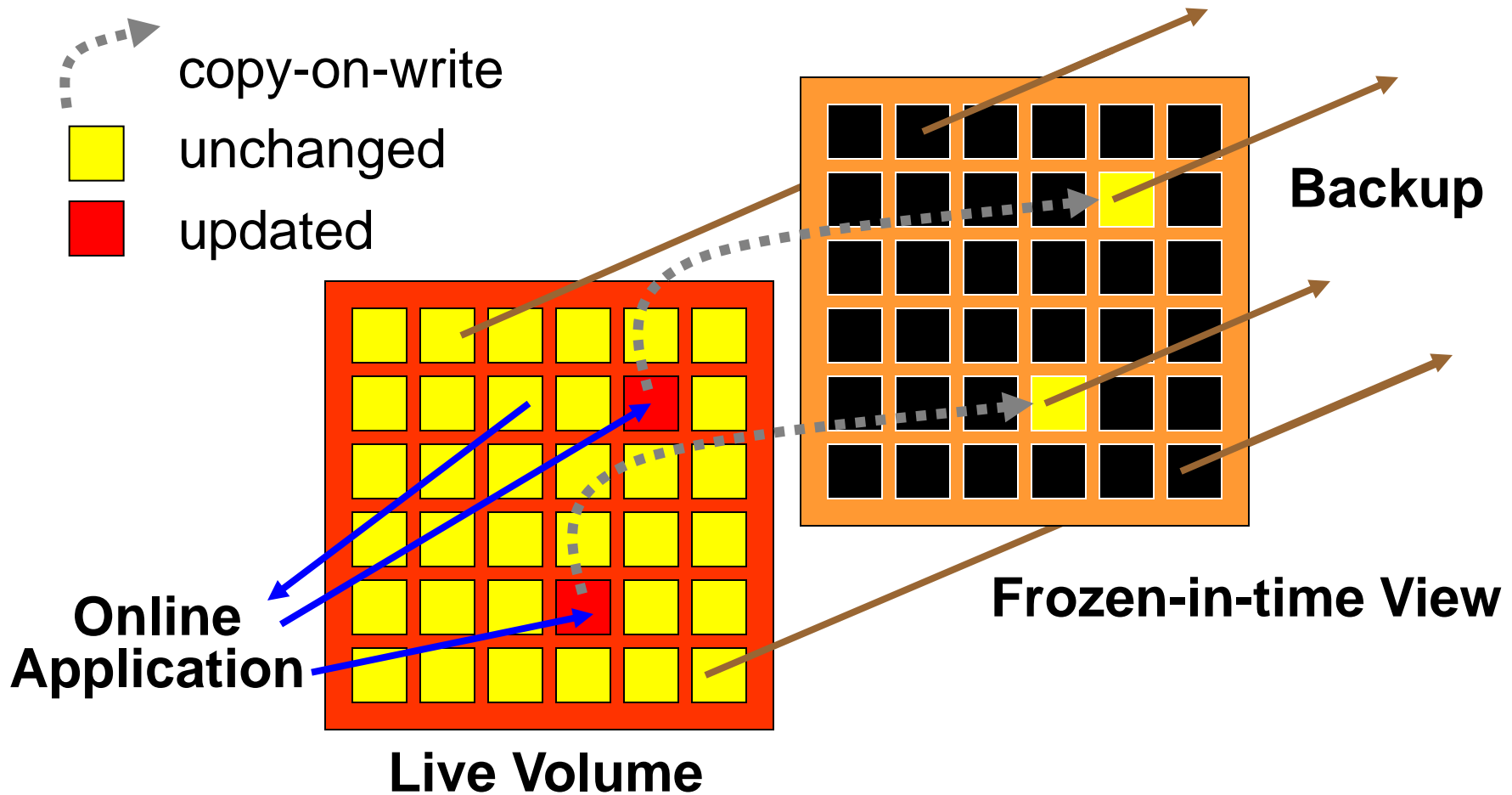
**Check out SNIA Tutorial:
Introduction to Data
Protection**

- ◆ Enhanced Storage & Data Services
 - ◆ Expose/extend the value of virtualization
- ◆ These services become significantly less complex when virtualization technology is implemented:
 - ◆ Backup & Restore
 - ◆ Clustering
 - ◆ Point In Time Copy / Snapshots
 - ◆ Replication
 - ◆ Migration
 - ◆ Transformation
 - ◆ Caching
 - ◆ Security
 - ◆ Quality of Storage Services & Policies
 - ◆ Pooling

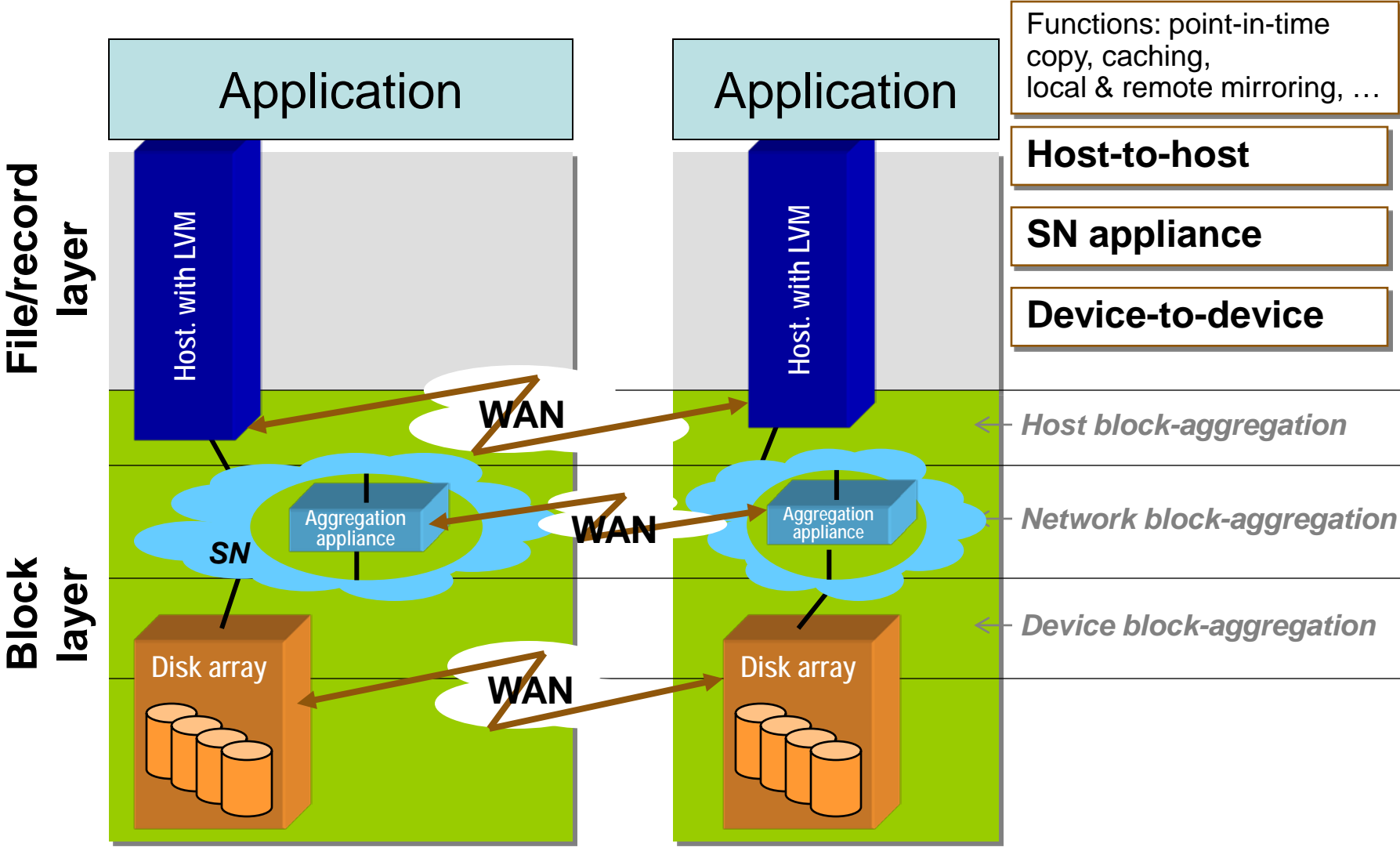
Full Block Copy Snapshot



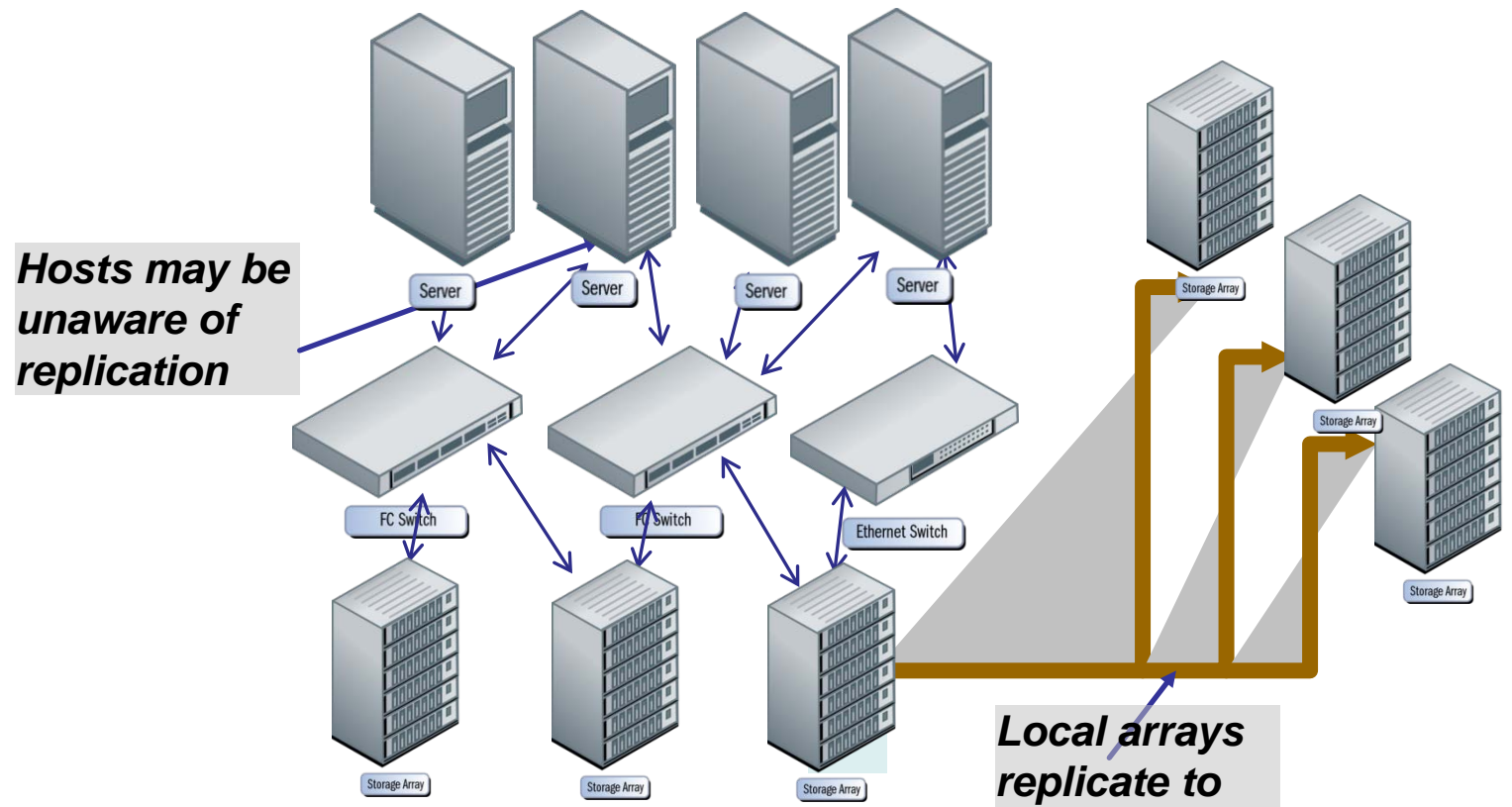
Copy-on-Write (CoW) Snapshot



Data Replication Multi-site block storage



Using Virtualization: Storage-based Data Replication



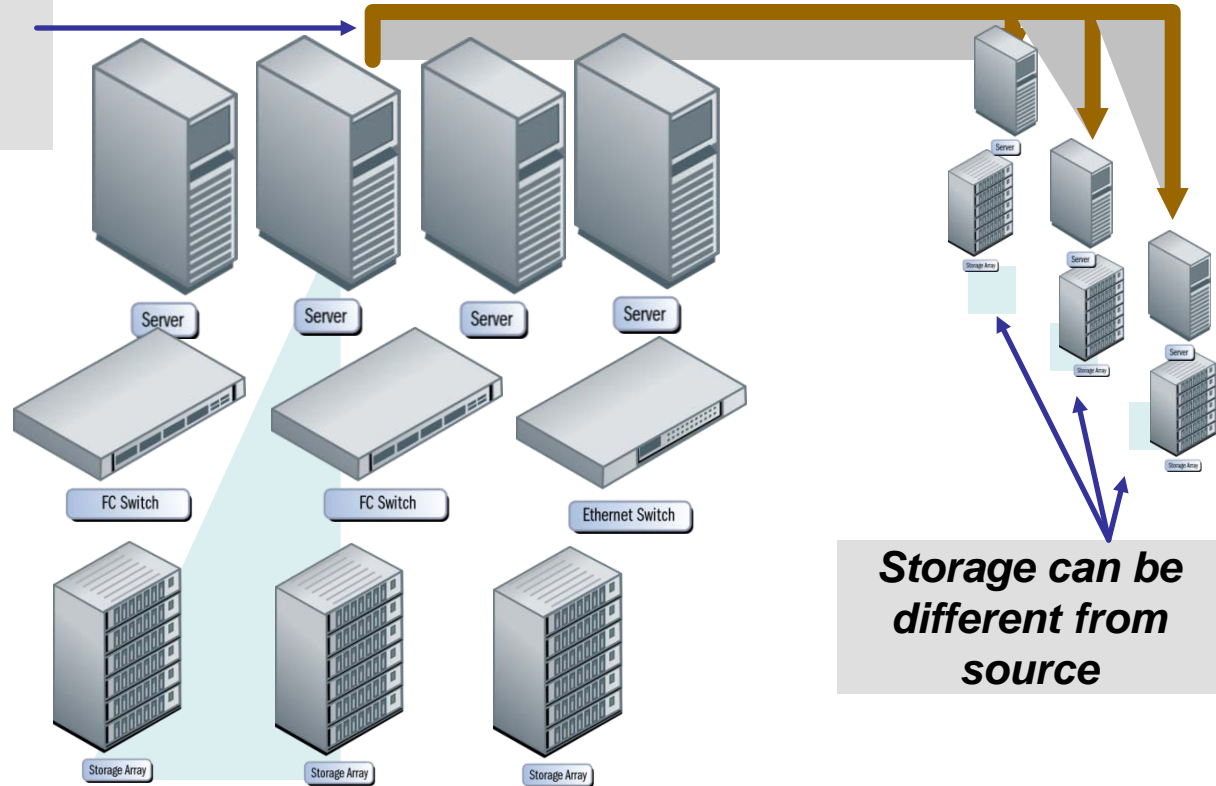
Hosts may be unaware of replication

Local arrays replicate to remote arrays

- + Minimal or no host load
- + Minimal client network load
- + Host platform independent
- + Network independent

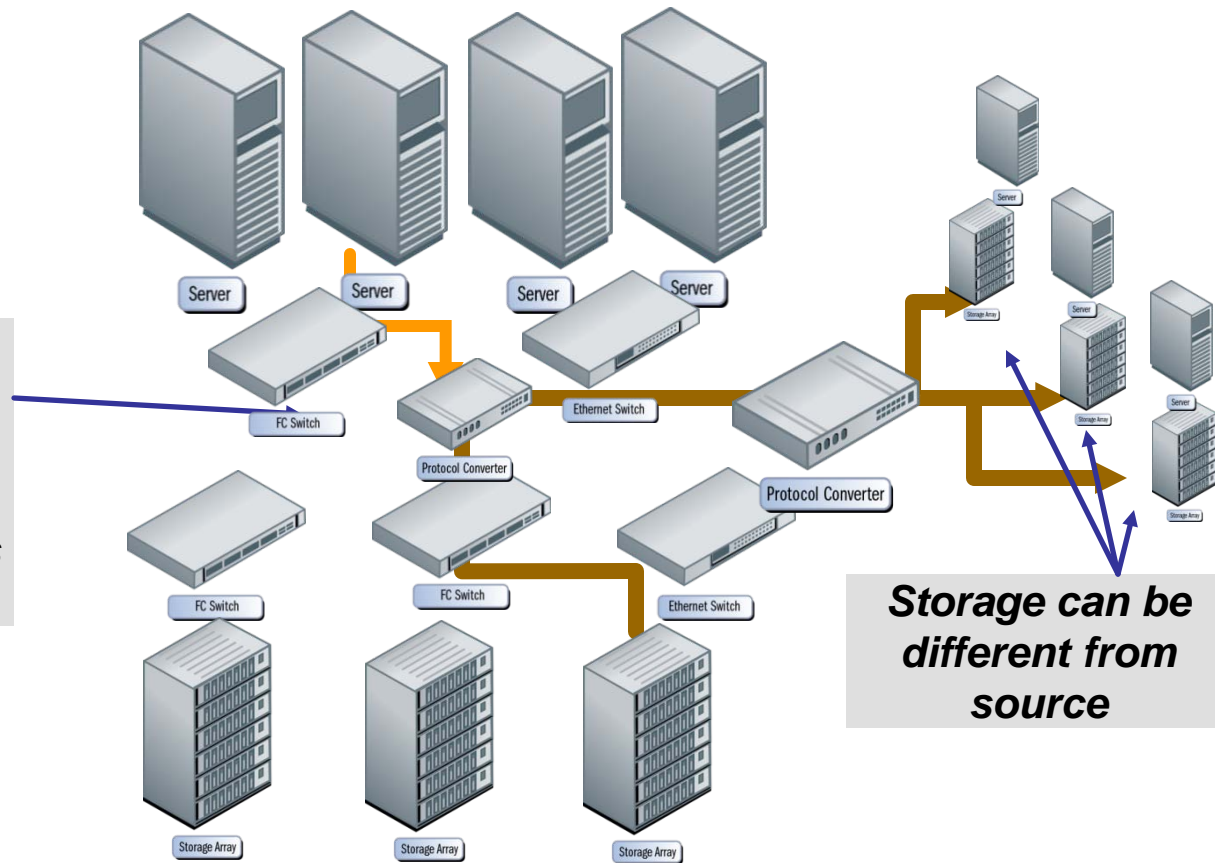
Using Virtualization: Host-based Data Replication

Volume updates replicated to remote servers



- + Recovers from
 - Network & target outages
 - Application load peaks
- + Storage device independent & Application transparent
- + Uses existing network

***SAN appliance
(In-band /
Out-of-Band) or
'Intelligent'
Switch' controls
the replication***



***Storage can be
different from
source***

- + No host load
- + Heterogeneous hosts and storage devices

- Unified Management
 - ◆ Virtualization plus Automation to deliver on SLAs
 - › Standardization (SNIA SMI-S) becomes very important
 - › TII creation of Fabric API Intelligence Standard (FAIS)
- Automatic and Intelligent Storage Provisioning
- Autonomic Data Migration Services
 - ◆ Based on policy, not merely time of last access
 - ◆ File-based and/or block-based
 - ◆ Data Lifecycle Management
- Data center-wide Volumes and File Systems

- SANs provide excellent storage connectivity
- Management is the challenge
 - ◆ Many non-cooperating servers
 - ◆ Hundreds to thousands of heterogeneous devices
- Virtualization to the rescue
 - ◆ The only way to cost-effectively reduce complexity
- Stand by for:
 - ◆ Storage Virtualization II
 - › ‘Effective use of Virtualization’

- Please send any questions or comments on this presentation to the SNIA at this address:
tracktutorials@snia.org

**Many thanks to the following individuals
for their contributions to this tutorial.**

SNIA Education Committee

**Frank Bunn
Curt Kolovson
Ben Kuo
John Logan
Gene Nagle
Russell Warren**

**Rob Peglar
Abbott Schindler
Wolfgang Singer
David Thiel
Joshua Tseng**



For More Information, See the *Storage Virtualization Hands-On Lab*