



Education

# **Storage Virtualization II Effective Use of Virtualization**

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 individuals may use this material in presentations and literature under the following conditions:
  - ◆ Any slide or slides used must be reproduced without modification
  - ◆ The SNIA must be acknowledged as 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 nor should be construed as legal advice or opinion. If you need legal advice or legal opinion please contact an attorney.
- The information presented herein represents the Author's personal opinion and current understanding of the 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.**

- *Virtualization Checklist – Background info*
- **Implementing Virtualization Step-by-Step**
- **Achieving .... Through Virtualization**
  - ◆ Capacity
  - ◆ Performance
  - ◆ High Availability
- **Storage Virtualization and the SNIA SMI-S**
- **Policy-based Service Level Management**
- **Q&A**

# Virtualization Checklist

- Before purchasing and implementing any product the user should be aware that Storage Virtualization is an enabling technology and is a part of the solution
- Storage Virtualization is a tool for the IT administrator to simplify the management of the storage resources and reduce the complexity of the overall IT infrastructure

# So, What's the Problem?

- Storage problems in specific areas, such as:
  - ◆ Capacity/Provisioning
  - ◆ Availability
  - ◆ Performance
  - ◆ Flexibility/Change of Attributes
  - ◆ Manageability
- Goals to achieve through adoption of virtualization
  - ◆ Align the storage infrastructure with the Business and IT Objectives of the end user
  - ◆ Measure with Service Level Agreements (SLAs)
    - › internally and externally defined
  - ◆ Implement business plans such as D/R, B/C and H/A

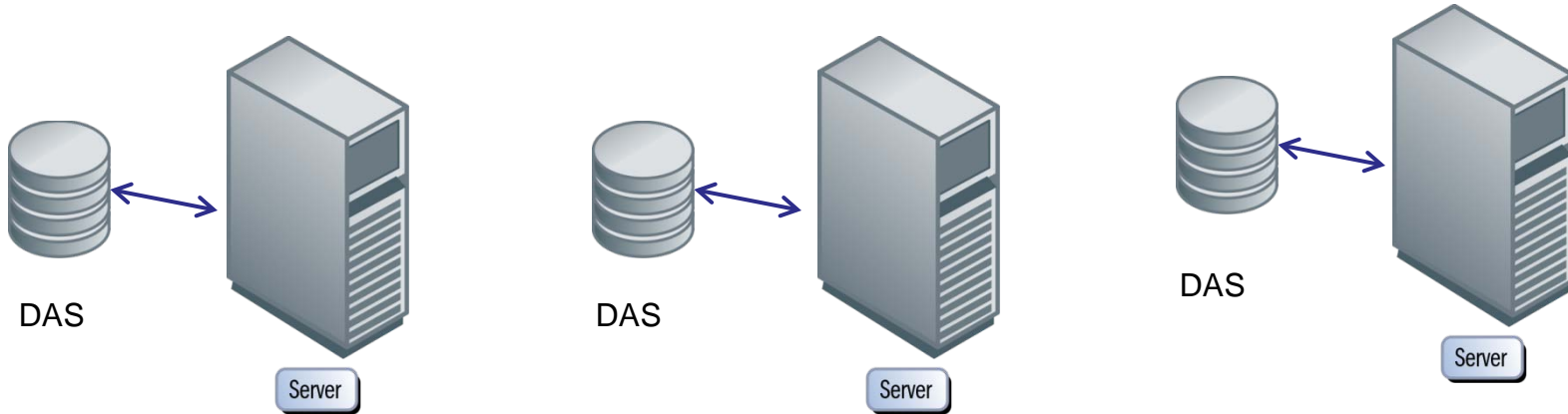
# Implementing Storage Virtualization step-by-step

- Step 1: Plan and Start
- Step 2: Add SAN infrastructure
- Step 3: Add Virtualization infrastructure
  - » Out-of-Band example
  - » In-Band example
- Step 4: Move DAS volumes to SAN
- Step 5: Change Primary/Secondary relationship
- Step 6: Establish HA environment
- Step 7: Create and Use Single Storage Pool
- Step 8: Establish Load Balancing/Multi-pathing
  
- *Sequence may change from project to project*

# Implementing Storage Virtualization

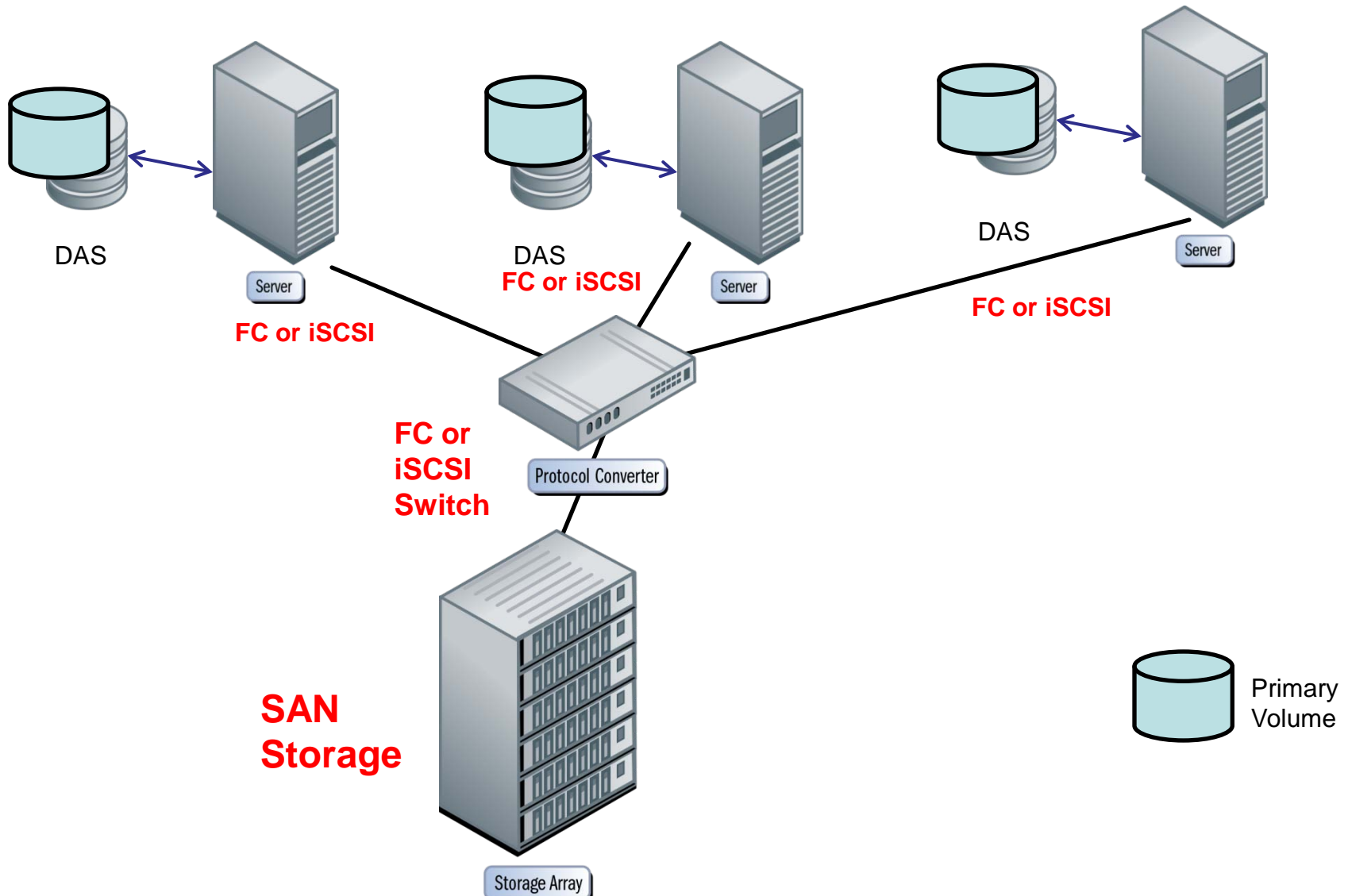
## 1. Start - DAS environment

DAS = either internally or externally direct connected storage devices



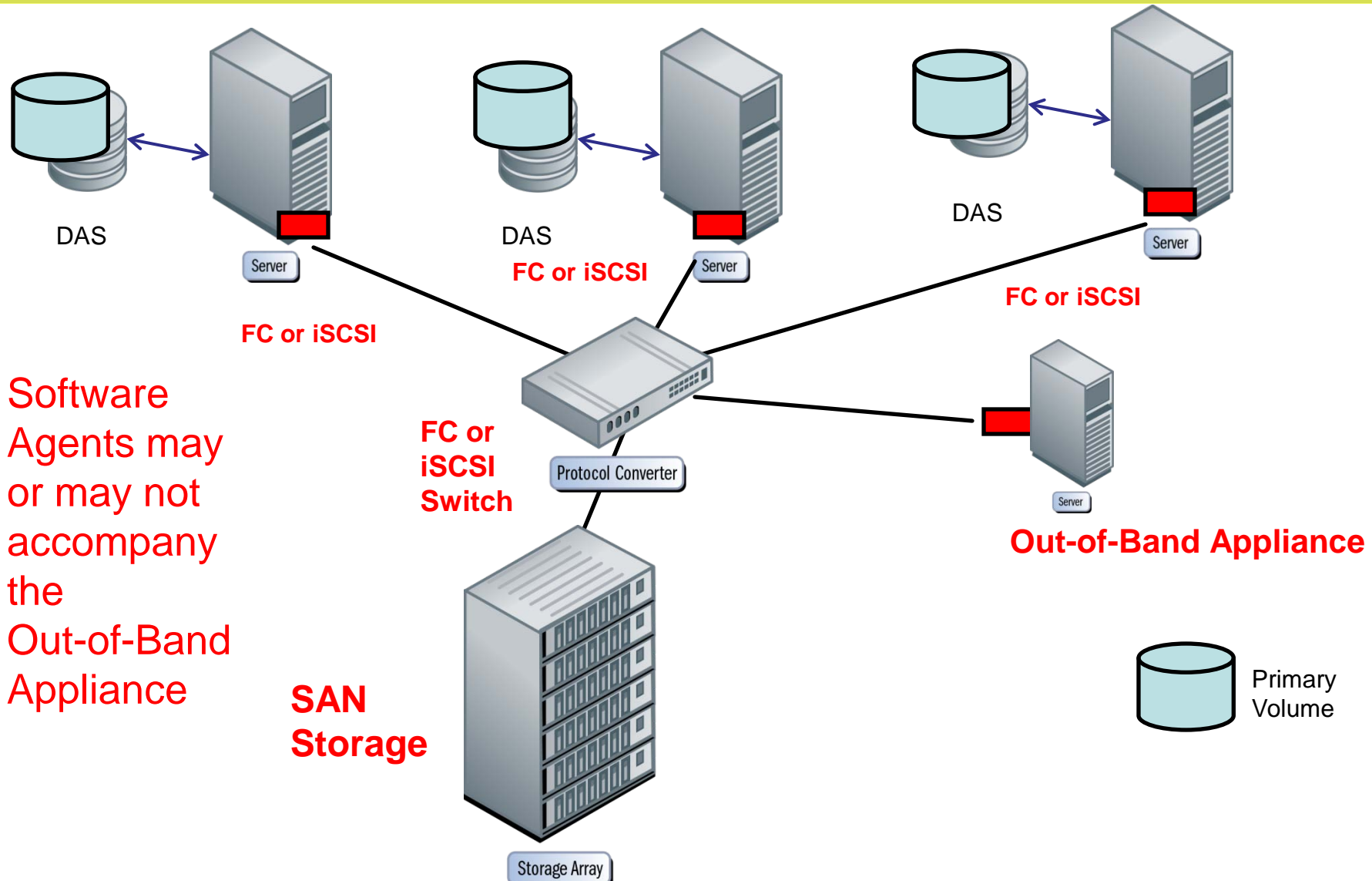
# Implementing Storage Virtualization

## 2. Create SAN infrastructure



# Implementing Storage Virtualization

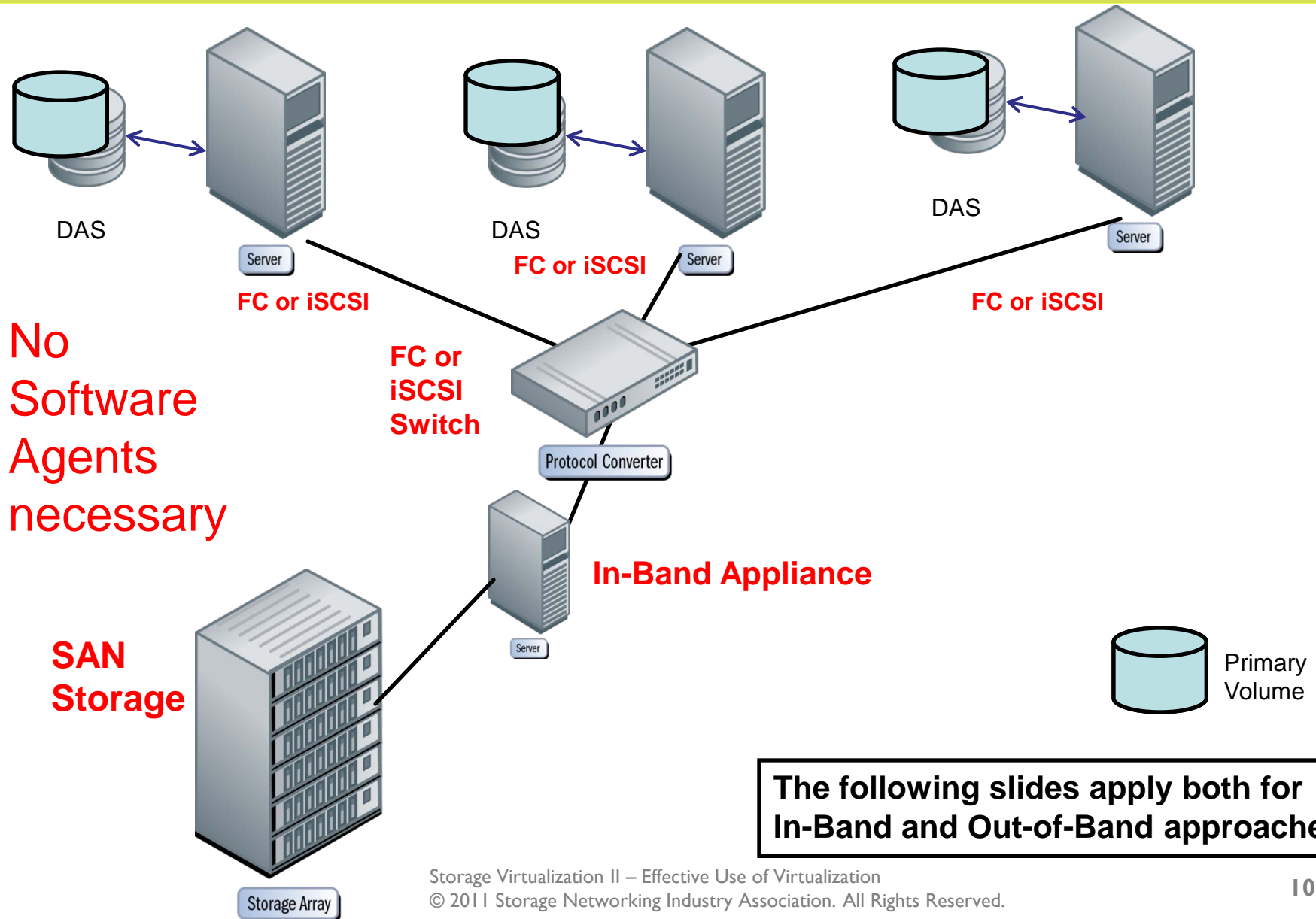
## 3a. Add Virtualization – Network OOB



Software Agents may or may not accompany the Out-of-Band Appliance

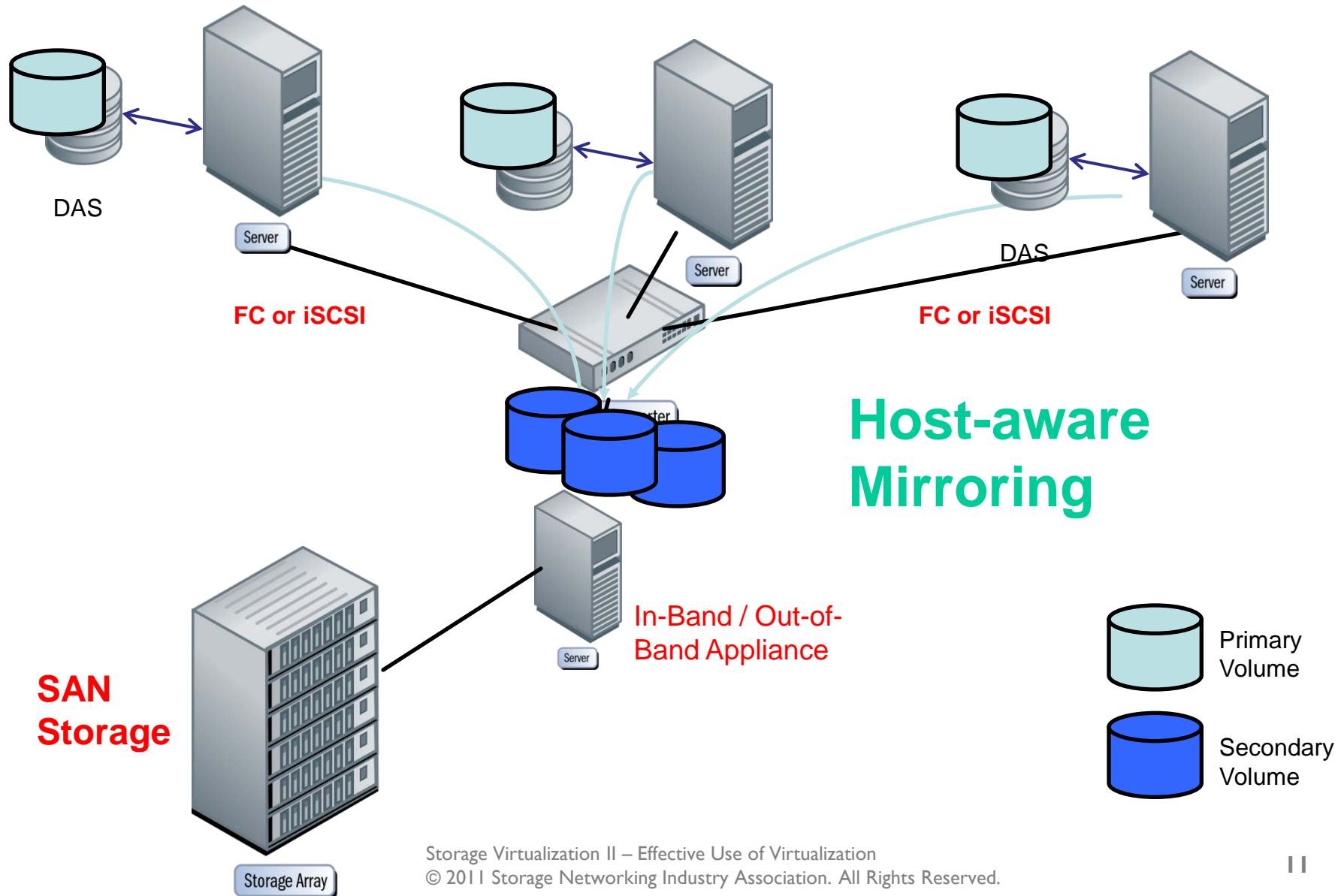
# Implementing Storage Virtualization

## 3b. Add Virtualization – Network InBand



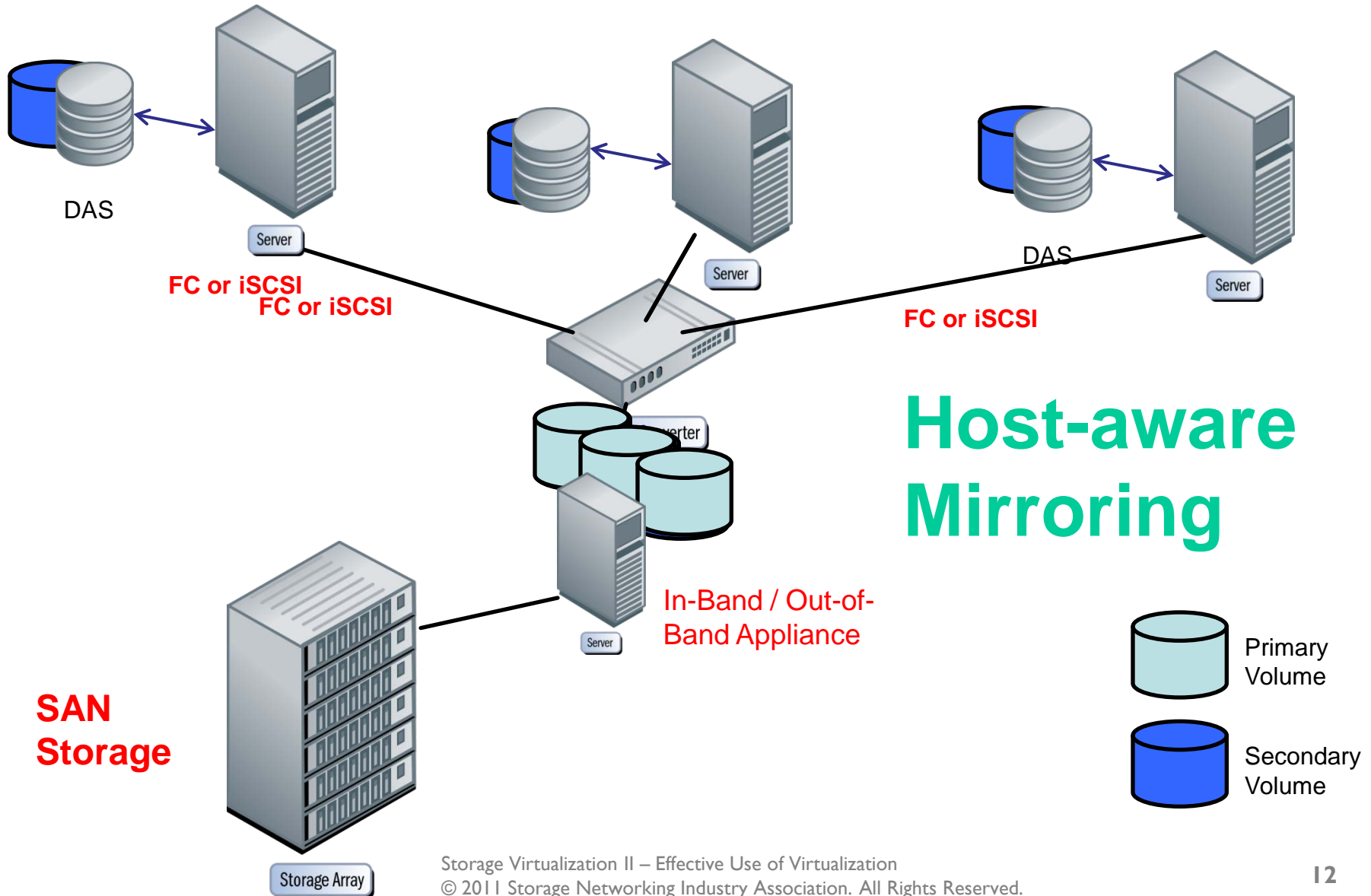
# Implementing Storage Virtualization

## 4. Mirror DAS volumes to SAN



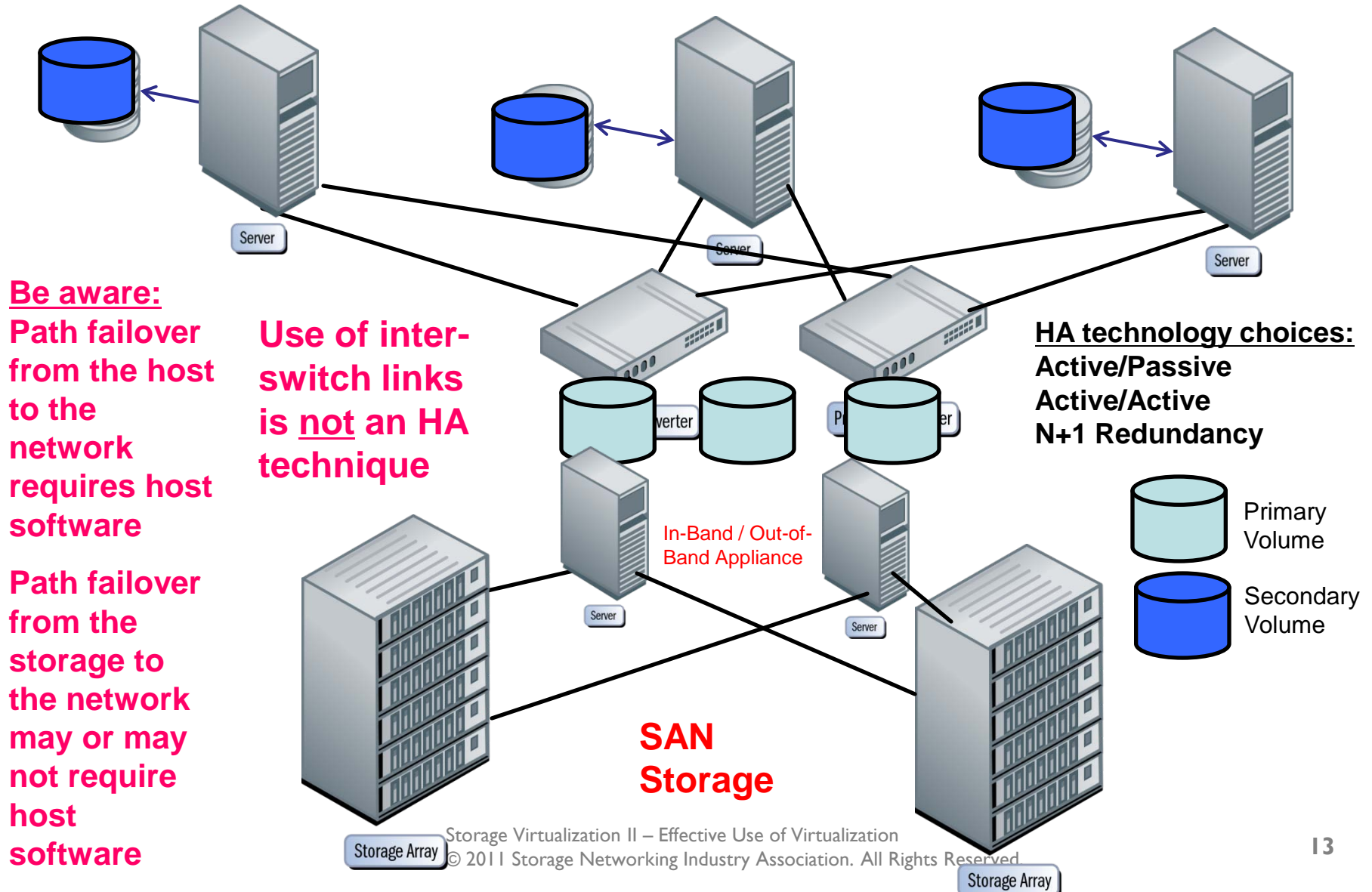
# Implementing Storage Virtualization

## 5. Change Primary/Secondary



# Implementing Storage Virtualization

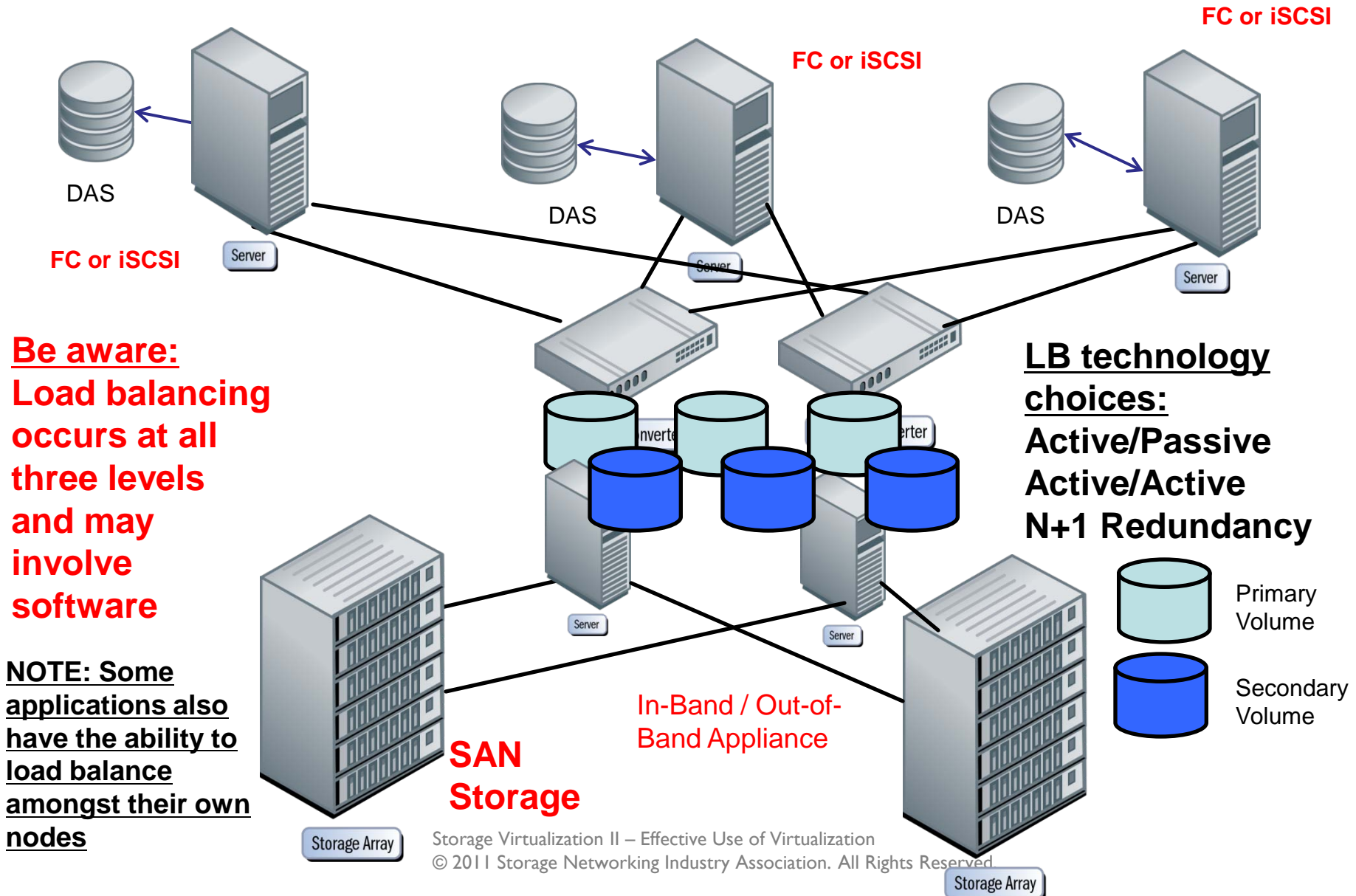
## 6. Establish HA Environment





# Implementing Storage Virtualization

## 8. Create Load Balancing

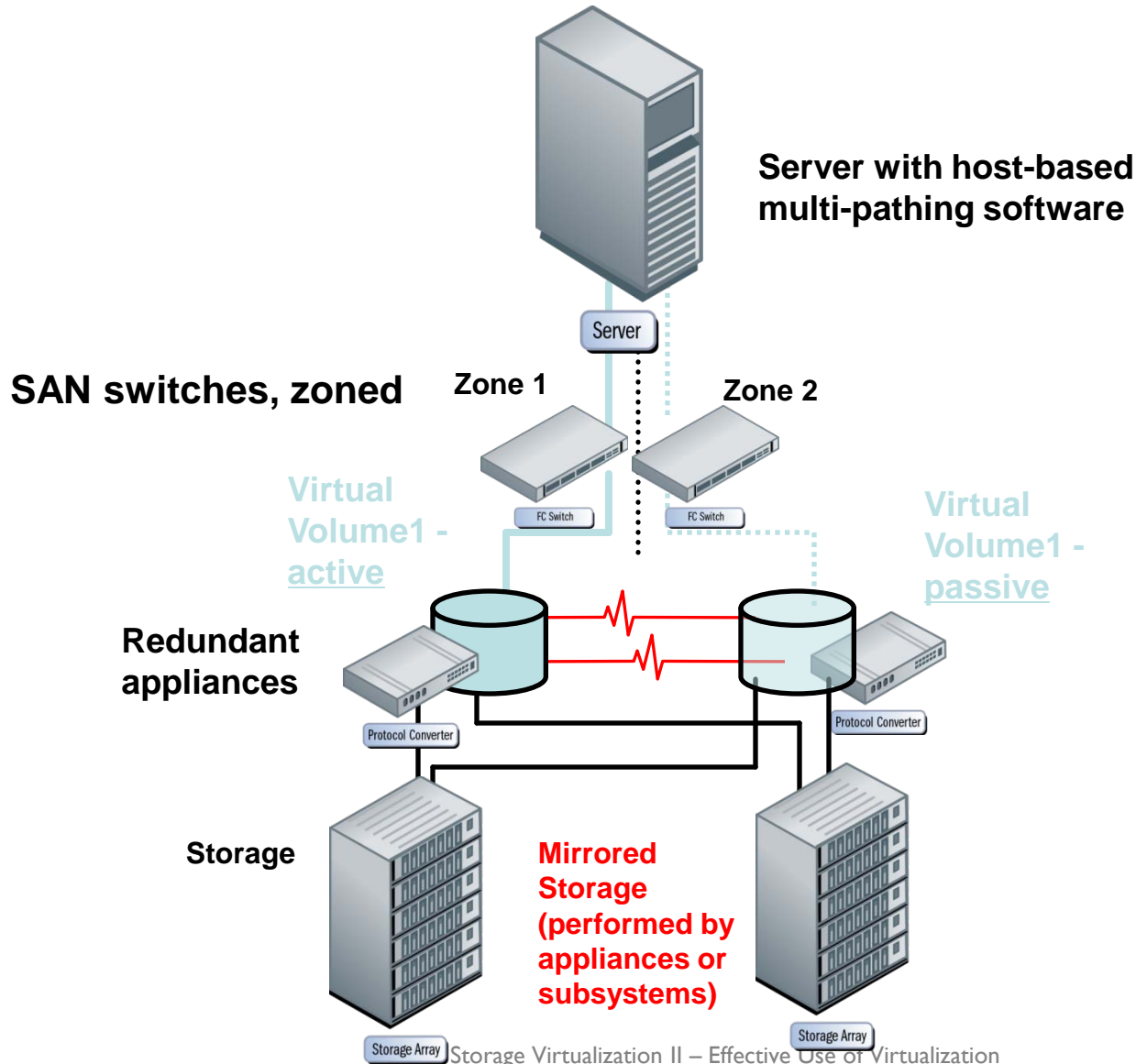


## Network-based Virtualization

- Do not rely on only one network-based appliance or intelligent switch only
  - ◆ Same rule for in-band as for out-of-band
- There are different methods to protect the engines
  - ◆ Active / Passive
  - ◆ Active / Active
  - ◆ N+1 redundancy
  - ◆ N-way distributed clustering
- The technique used is vendor-specific

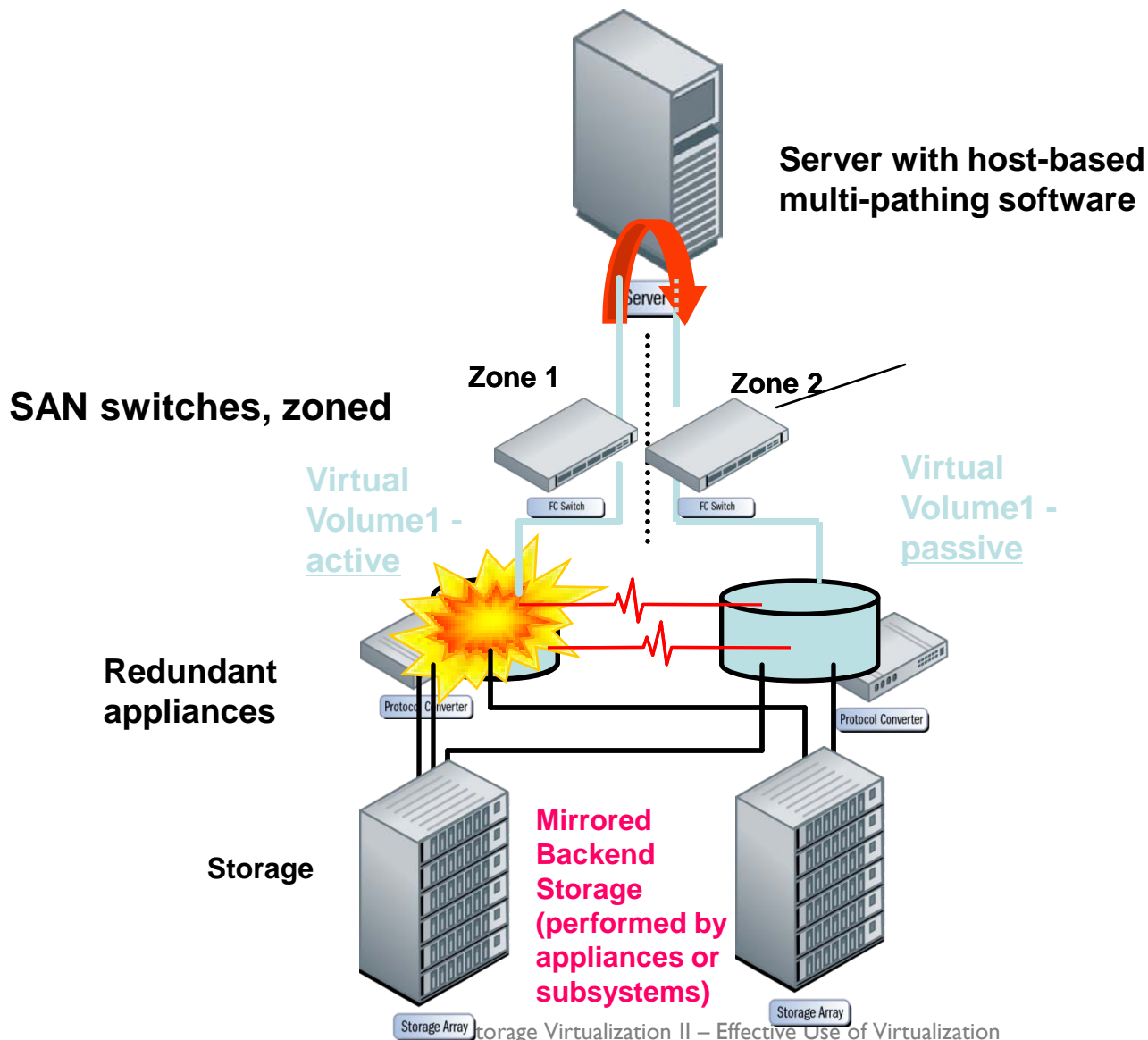
# Achieving High Availability

Example: active/passive appliances or subsystems



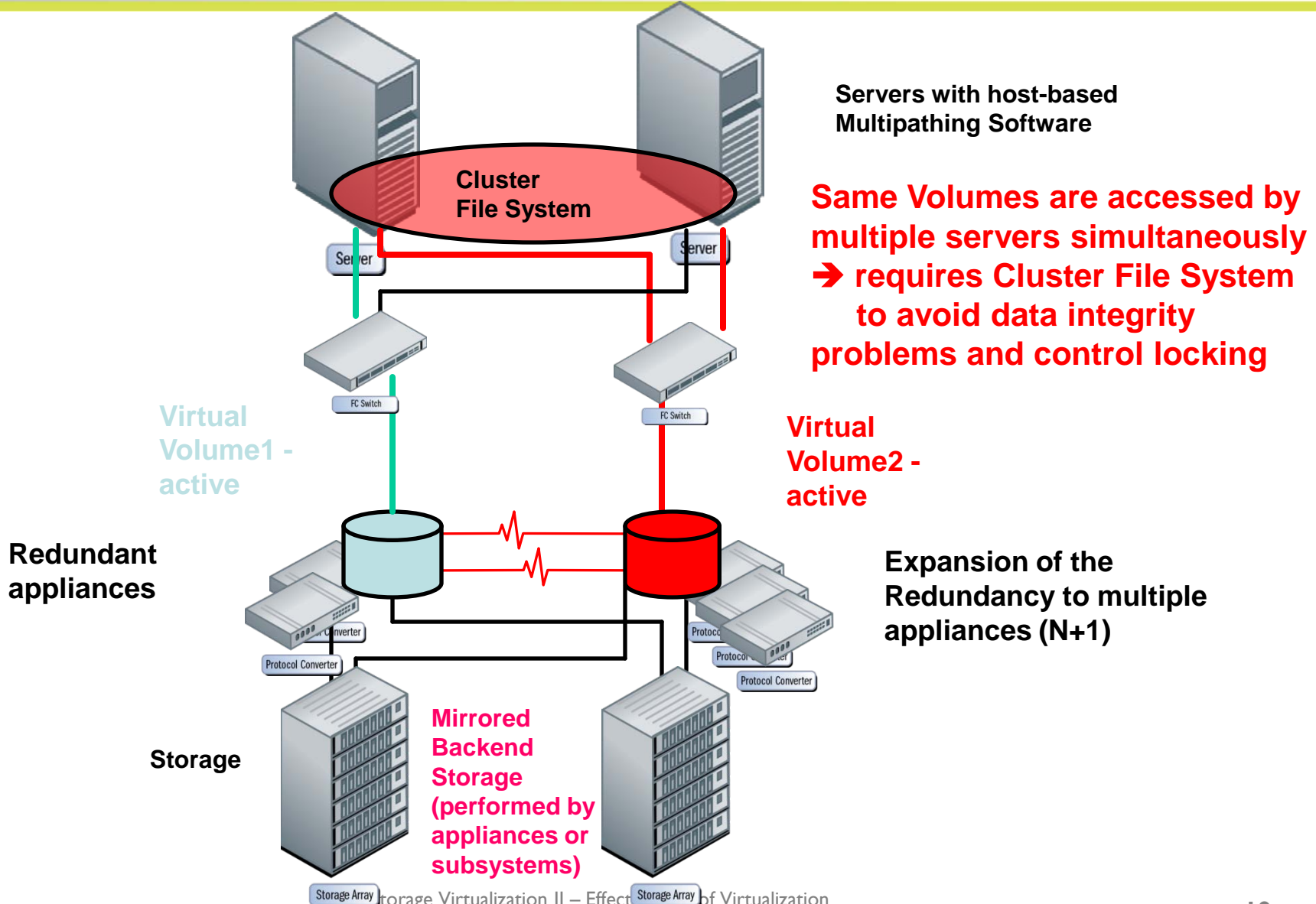
# Achieving High Availability

## Example: active/passive appliances or subsystems



# Achieving High Availability

Example: Multiple access of same volume



Servers with host-based Multipathing Software

Same Volumes are accessed by multiple servers simultaneously  
→ requires Cluster File System to avoid data integrity problems and control locking

Virtual Volume1 - active

Virtual Volume2 - active

Redundant appliances

Expansion of the Redundancy to multiple appliances (N+1)

Storage

Mirrored Backend Storage (performed by appliances or subsystems)

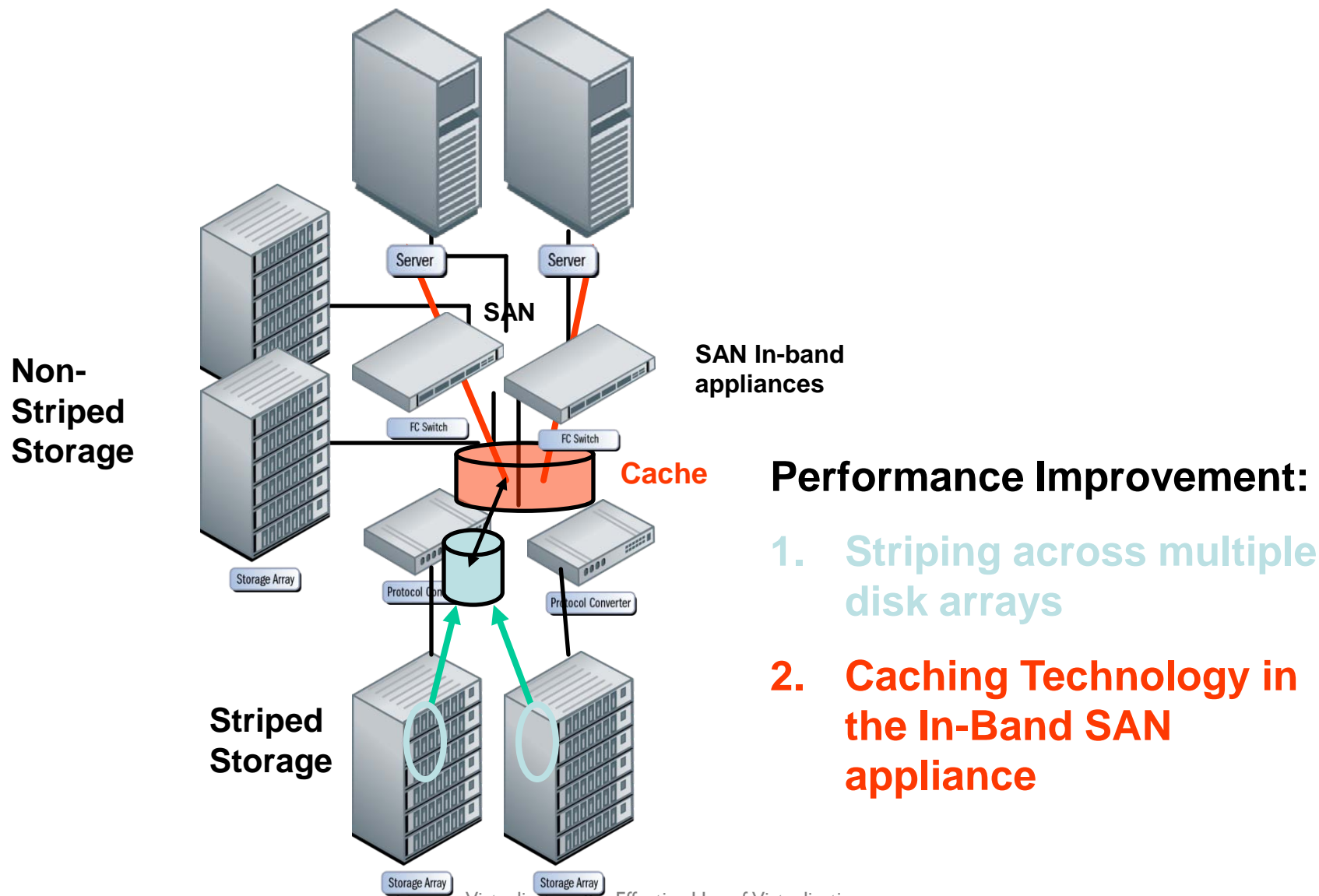
# High Availability Considerations for 'intelligent' switches

- To support high-availability configurations, virtual storage management must be distributed across two or more switches.
  - ◆ The switches present a virtual volume/LUN to the host(s) for a given LUN presented by the storage array(s) – and that LUN may be a virtual disk
  - ◆ Host-based multi-pathing software allows active-passive or active-active access to the virtual volumes presented to the host(s)
  - ◆ Allows hosts to access virtual volumes in the presence of a switch failure

- Striping / Mirroring
  - ◆ Simultaneous reads and/or writes
- Load Sharing
- Load Balancing
- Multipathing
- Off-loading host systems
- Caching (where possible)

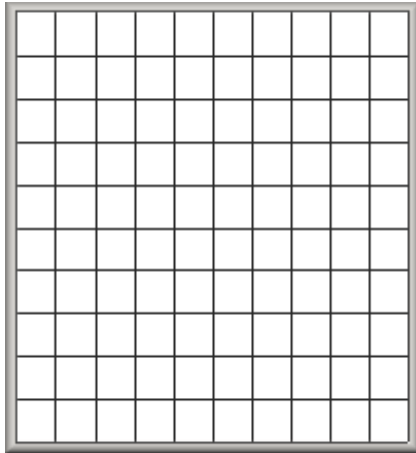
# Achieving Performance

## Example: In-band appliance

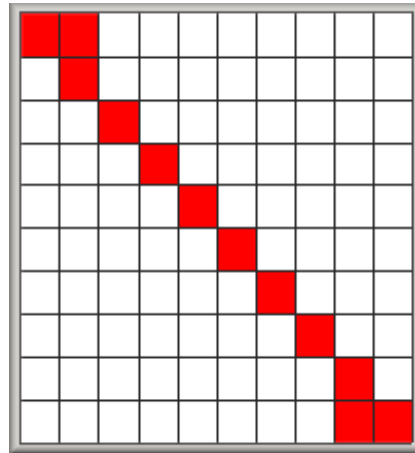


- Eliminate fixed-size LUNs
- Create dynamic virtual LUNs and expand/shrink the LUNs as necessary
  - ◆ Requires dynamic volume /LVM on the host(s)
- Create large virtual LUNs and pools and assign backing (physical) storage to it as the host writes data (Sparse Allocation)
- Data reduction techniques (dedup, compress, incrementalize)
- Dynamic growth of Volumes and File Systems simultaneously

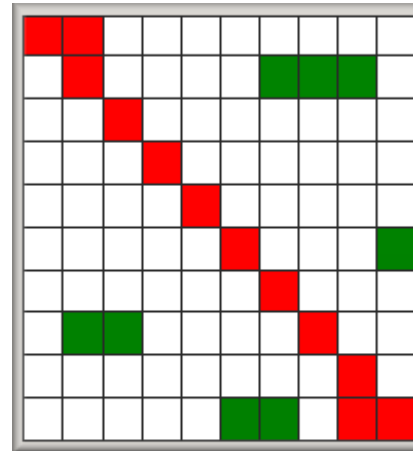
# Sparse Allocation a.k.a. Thin Provisioning



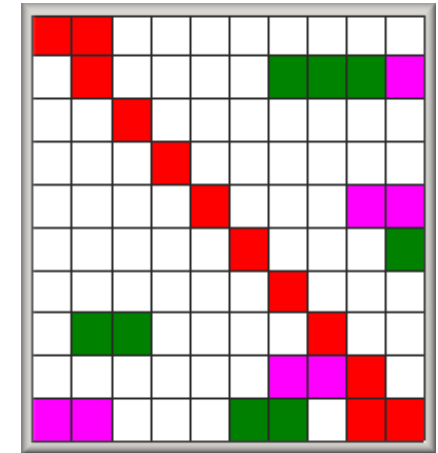
1. New Volume



2. Initial Filesystem allocation



3. Data Written



3. More Data Written

□ Unallocated Logical Block  
■ Filesystem Metadata

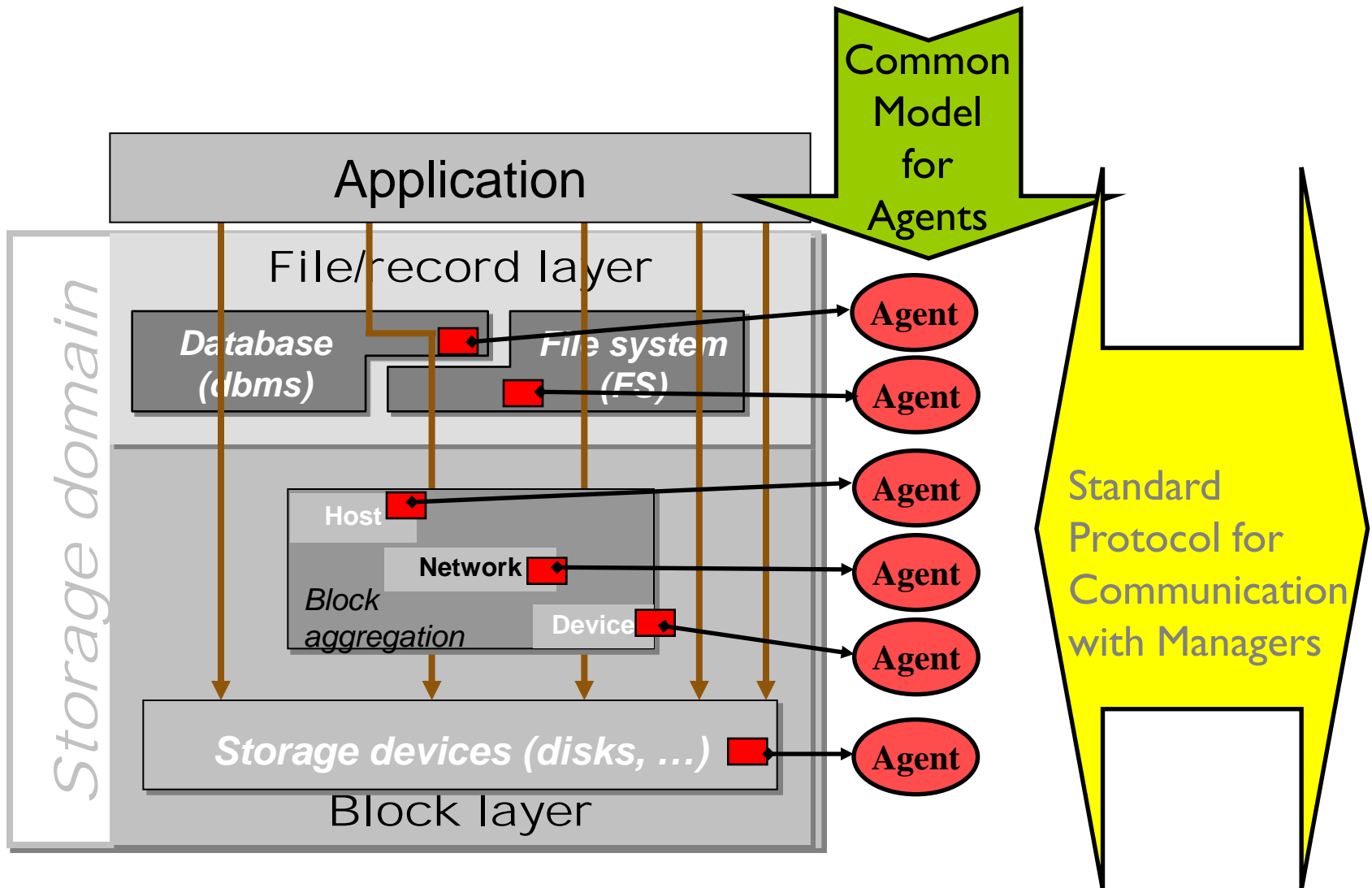
■ Application Data  
■ Application Data

**Saves space – BUT...**

- Beware of Thin “runaway” (array over-subscription)
- Economic advantage varies - reserved space %
- Large thin LUN versus small growing LUN
- Read thin LUN may present problems
  - ◆ Backup, Copy, Duplicate, Replica
  - ◆ Vendor-specific treatment when reading unwritten blocks
- Over-allocate LUN versus over-subscribe system
- Consider data reduction instead of thin provisioning
- Consider using app/OS/HV thin instead of array thin
- Be careful what you ask for – you may get it!

- SNIA's strategic initiative to solve end-user operational challenges for Storage Management
  - ◆ Passive (Discovery and Monitoring)
  - ◆ Active (Storage Configuration; manually and policy-based)
- Based on Standards
  - ◆ WBEM (Web Based Enterprise Management)
  - ◆ CIM (Common Management Model)
- Includes Block Virtualization in its first version
- Now up to version 1.5; 1.6 in the works

# Control Path: SNIA Shared Storage Model

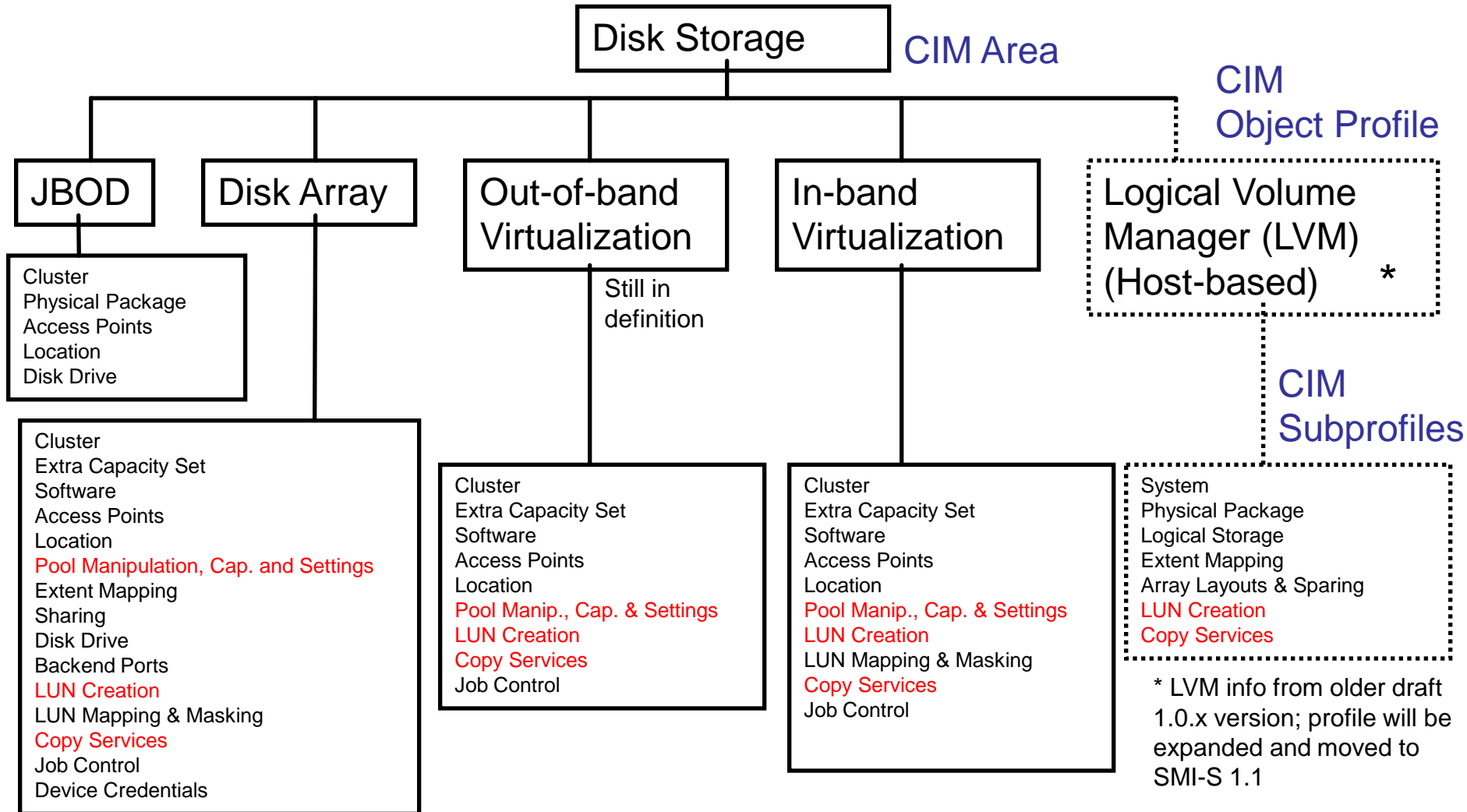


# Some SMI-S Capabilities

1. Identify key resources in a SAN
2. Identify interconnects between key resources in a SAN
3. Receive asynchronous notification that the configuration in a SAN has changed
4. Identify the health of key resources in a SAN
5. Receive asynchronous notification that the health of a SAN has changed
6. Identify the available performance of interconnects in a SAN
7. Receive asynchronous notification that the performance of a SAN interconnect has changed
8. Identify the zones being forced in as SAN
9. Create/Delete and enable/disable zones in a SAN
- 10. Identify the storage volumes in a SAN**
- 11. Create/delete/modify storage volumes in a SAN**
- 12. Identify the connectivity and access rights to Storage Volumes in a SAN**
- 13. Create/delete and enable/disable connectivity and access rights to storage volumes in a SAN**
14. Allow a site to require the use of authenticated clients

**Block Virtualization**

# Disk Storage in the SMI-S



- SMI-S is not a virtualization of storage per se
- It is a “virtualization” of the management APIs for the different vendor’s components.
- The long term impact of SMI-S on virtualization products is profound!
  - ◆ Eliminates the need for proprietary APIs to perform common management tasks such as creating LUNs, manage snapshots or data replication
  - ◆ Avoids reverse engineering and its corresponding problems
- SNIA SMI-S VI.1.x already delivers standard interfaces that use virtualization technologies
- Will help to establish Policy-based Service Level Management and Automated Storage Resource Management (SRM)

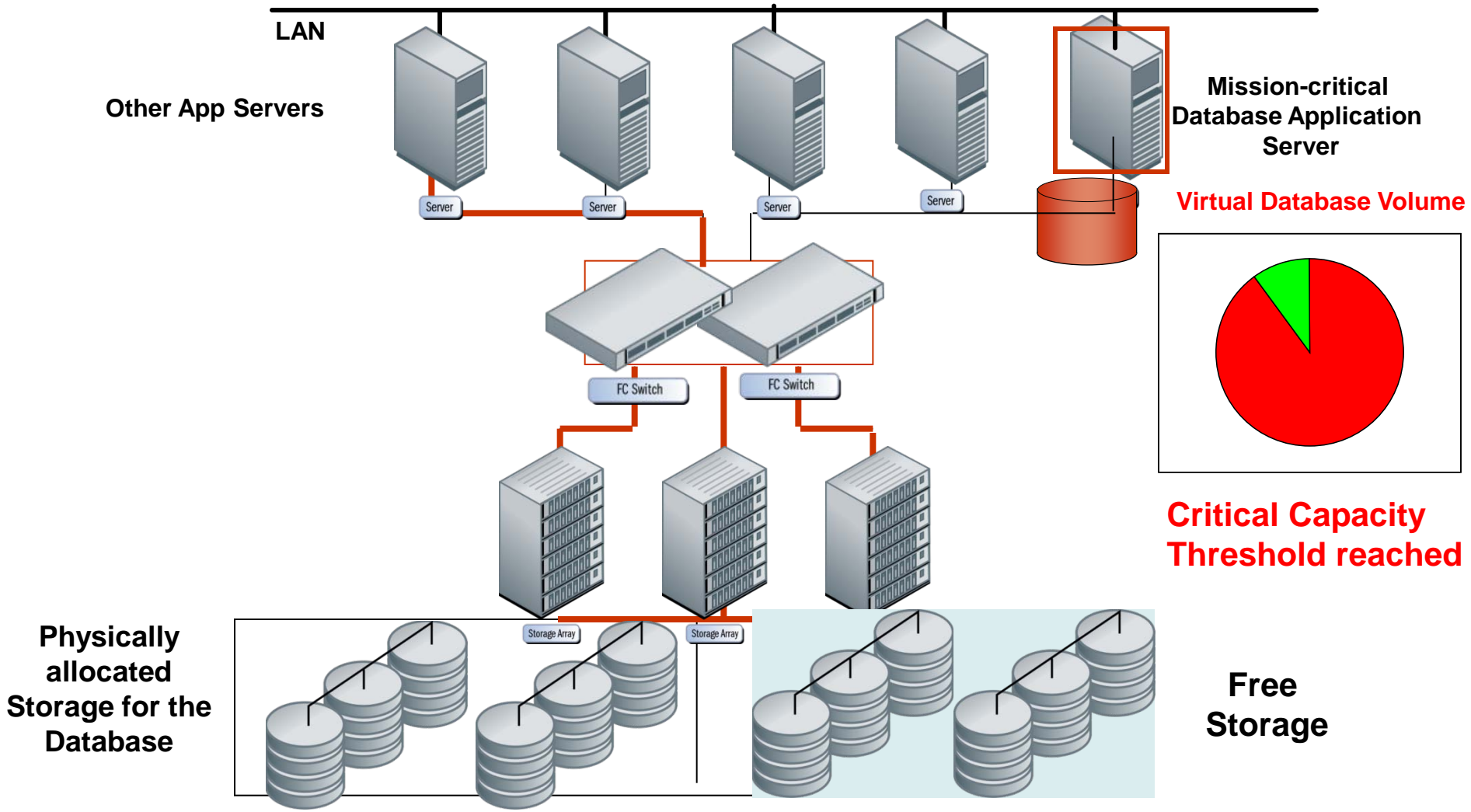
# Policy-based Service Level Management

What's the role of storage virtualization?

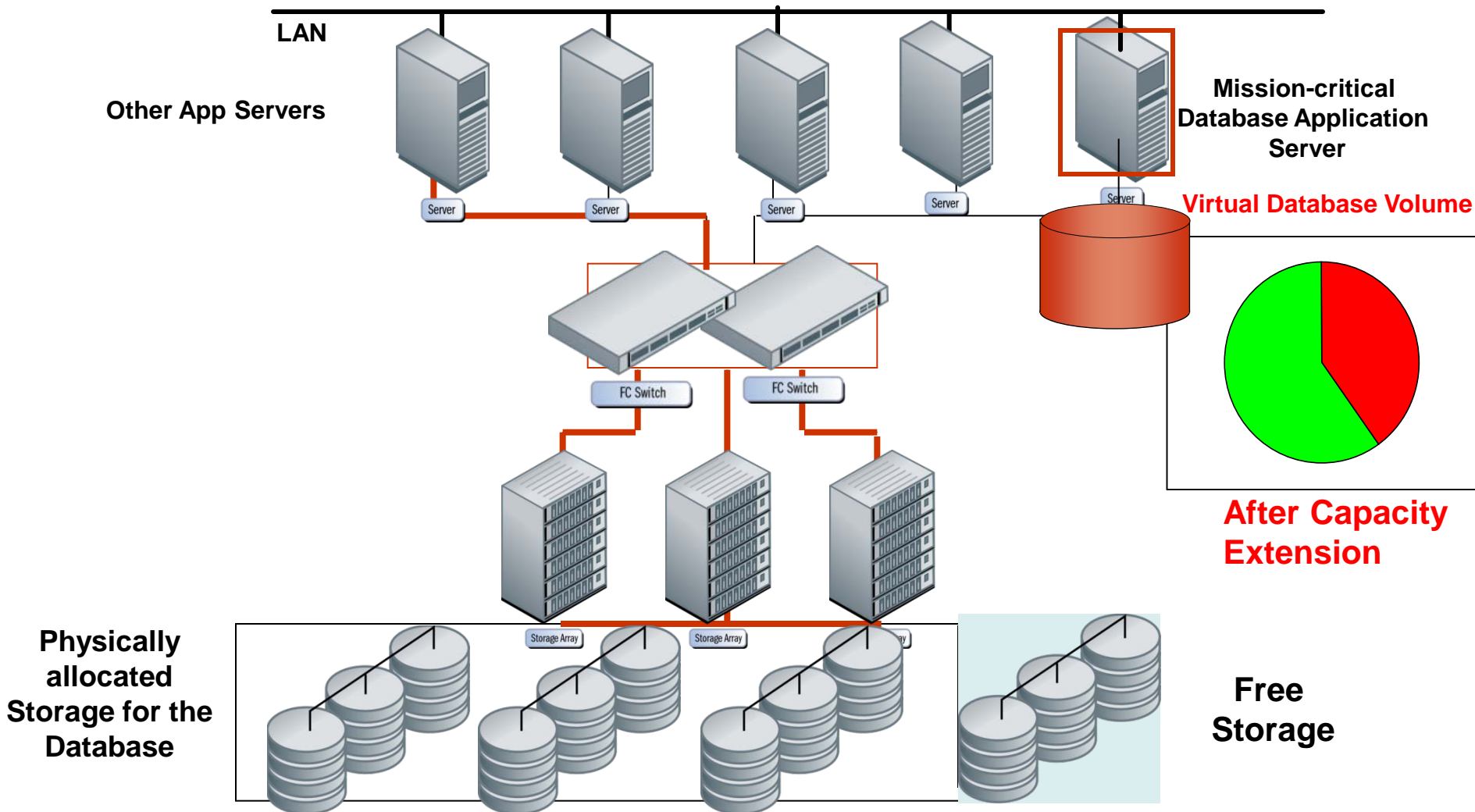
- Handles error-prone administrator tasks (such as storage provisioning) automatically
- Use of Web Services or REST to communicate
- Cloud Data Management Interface (CDMI)
  - ◆ Management especially for Cloud Storage
- Pre-defined rules (policies) must be set
- One critical Service Level Management outcome is efficient Storage Capacity Planning
  - ◆ Dynamic Provisioning
  - ◆ Automated Capacity Plan Execution

- **Monitoring of Storage Capacity**
  - ◆ Application-centric – but can be done in fabric or subsystem
- **Threshold Management**
  - ◆ Definition of rules: What to do, when, to whom...
- **Discover free capacity with desired storage attributes**
- **Assign new storage into the server zone(s)**
  - ◆ Switch zoning via REST or SNIA SMI-S
- **Grant specific server(s) access to the storage**
  - ◆ LUN Masking via REST or SNIA SMI-S
- **Map storage to the server volume(s) (Online !!)**
  - ◆ Resize / Re-layout the volume (REST or SNIA SMI-S)
- **Make larger volume aware to the application**
  - ◆ Including automatic, dynamic growth of file system

# Policy-based Service Level Management - big picture (I)



# Policy-based Service Level Management - big picture (II)



- Policy-based Service Level Management must be performed (executed) without any user interaction
- Once new free storage capacity is discovered, the existing volumes must be resized online without any impact to the application
- Only Storage Virtualization techniques can assure these requirements – coupled with OS involvement

- Please send any questions or comments on this presentation to the SNIA: [tracktutorials@snia.org](mailto: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**

**Nik Simpson**

**Wolfgang Singer**

**David Thiel**

**Joshua Tseng**



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