



Education

The File Systems Evolution

Christian Bandulet
Principal Engineer, Sun Microsystems

SNIA Legal Notice

- 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.

➤ The File Systems Evolution

- ◆ File Systems impose structure on the address space of one or more physical or virtual devices. Starting with local file systems over time additional file systems appeared focusing on specialized requirements such as data sharing, remote file access, distributed file access, parallel files access, HPC, archiving, security etc.. Due to the dramatic growth of unstructured data files as the basic units for data containers are morphing into file objects providing more semantics and feature-rich capabilities for content processing. This presentation will categorize and explain the basic principles of currently available file systems (e.g. Local FS, Shared FS, SAN FS, Clustered FS, Network FS, Distributed FS, Parallel FS, ...). It will also explain technologies like Scale-Out NAS, NAS Aggregation, NAS Virtualization, NAS Clustering, Global Namespace, Parallel NFS. All of these files system categories and technologies are complementary. They will be enhanced in parallel with additional value added functionality. New file system architectures will be developed and some of them will be blended in the future.

Check Out Other Tutorials



Check out SNIA Tutorial:
Using Classification to Manage
File Servers



Check out SNIA Tutorial:
Exploiting Multi-Tier File
Storage Effectively



Check out SNIA Tutorial:
NAS and iSCSI Technology
Overview



Check out SNIA Tutorial:
Find and Select the Right File
Storage for Your Application



Check out SNIA Tutorial:
Massively Scalable File Storage



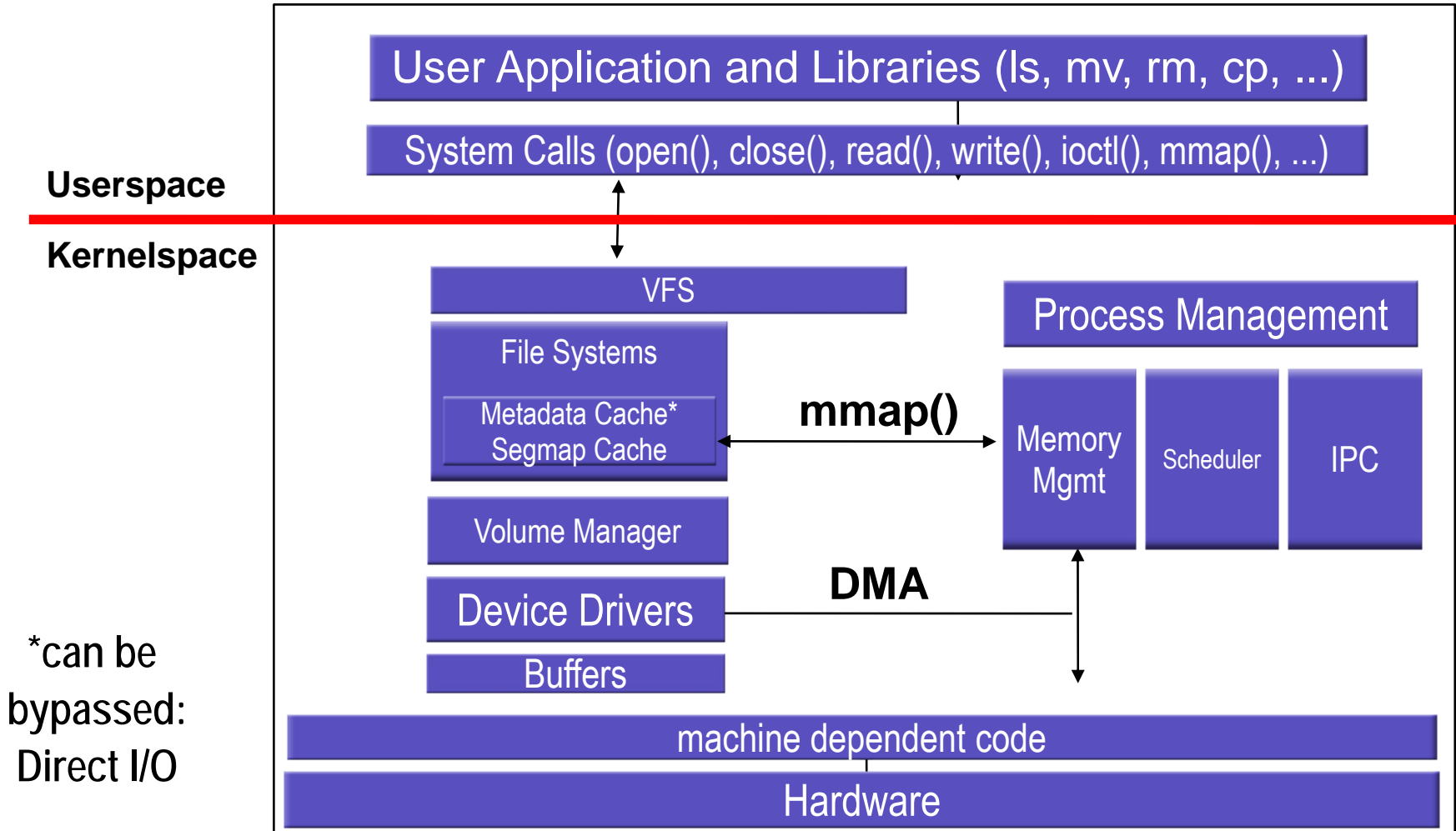
Check out SNIA Tutorial:
pNFS, Parallel Storage for Grid and
Enterprise Computing



Check out SNIA Tutorial:
DFS For Not-So-Dummies

- **File System Basics**
- File Systems Taxonomy
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- Distributed FS
- Distributed Parallel FS
- Scale-Out NAS
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments

File System & Operating System



- File System Basics
- **File Systems Taxonomy**
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- Distributed FS
- Distributed Parallel FS
- Scale-Out NAS
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments

- ADFS – Acorn's Advanced Disc filing system, successor to DFS
- BFS – the Be File System used on BeOS
- EFS – Encrypted filesystem, An extension of NTFS
- EFS (IRIX) – an older block filing system under IRIX
- Ext – Extended filesystem, designed for Linux system
- Ext2 – Second extended filesystem, designed for Linux systems
- Ext3 – Name for the journalled form of ext2
- FAT – Used on DOS and Microsoft Windows, 12, 16 and 32 bit table depths
- FFS (Amiga) – Fast File System, used on Amiga systems. This FS has evolved over time. Now counts FFS1, FFS Intl, FFS DCache, FFS2
- FFS – Fast File System, used on *BSD systems
- Fossil – Plan 9 from Bell Labs snapshot archival file system
- Files-11 – OpenVMS filesystem
- GCR – Group Code Recording, a floppy disk data encoding format used by the Apple II and Commodore Business Machines in the 5¼" disk drives for their 8-bit computers
- HFS – Hierarchical File System, used on older Mac OS systems



- ◆ HFS Plus – Updated version of HFS used on newer Mac OS systems
- ◆ HPFS – High Performance Filesystem, used on OS/2
- ◆ ISO 9660 – Used on CD-ROM and DVD-ROM discs (Rock Ridge and Joliet are extensions to this)
- ◆ JFS – IBM Journaling Filesystem, provided in Linux, OS/2, and AIX
- ◆ LFS – 4.4BSD implementation of a log-structured file system
- ◆ MFS – Macintosh File System, used on early Mac OS systems
- ◆ Minix file system – Used on Minix systems
- ◆ NTFS – Used on Windows NT, Windows 2000, Windows XP and Windows Server 2003 systems
- ◆ NSS – Novell Storage Services. This is a new 64-bit journaling filesystem using a balanced tree algorithm. Used in NetWare versions 5.0-up and recently ported to Linux.
- ◆ OFS – Old File System, on Amiga. Nice for floppies, but fairly useless on hard drives
- ◆ PFS – and PFS2, PFS3, etc. Technically interesting filesystem available for the Amiga, performs very well under a lot of circumstances. Very simple and elegant



- ReiserFS – Filesystem that uses journaling
- Reiser4 – Filesystem that uses journaling, newest version of ReiserFS
- SFS – Smart File System, journaled file system available for the Amiga platforms
- UDF – Packet based filesystem for WORM/RW media such as CD-RW and DVD.
- UDF – Packet based filesystem for WORM/RW media such as CD-RW and DVD
- UFS – Unix Filesystem, used on older BSD systems
- UFS2 – Unix Filesystem, used on newer BSD systems
- UMSDOS – FAT filesystem extended to store permissions and metadata, used for Linux
- VxFS – Veritas file system, first commercial journaling file system; HP-UX, Solaris, Linux, AIX
- VSAM
- WAFL – Used on Network Appliance systems
- XFS – Used on SGI IRIX and Linux systems
- ZFS – Used on Solaris 10



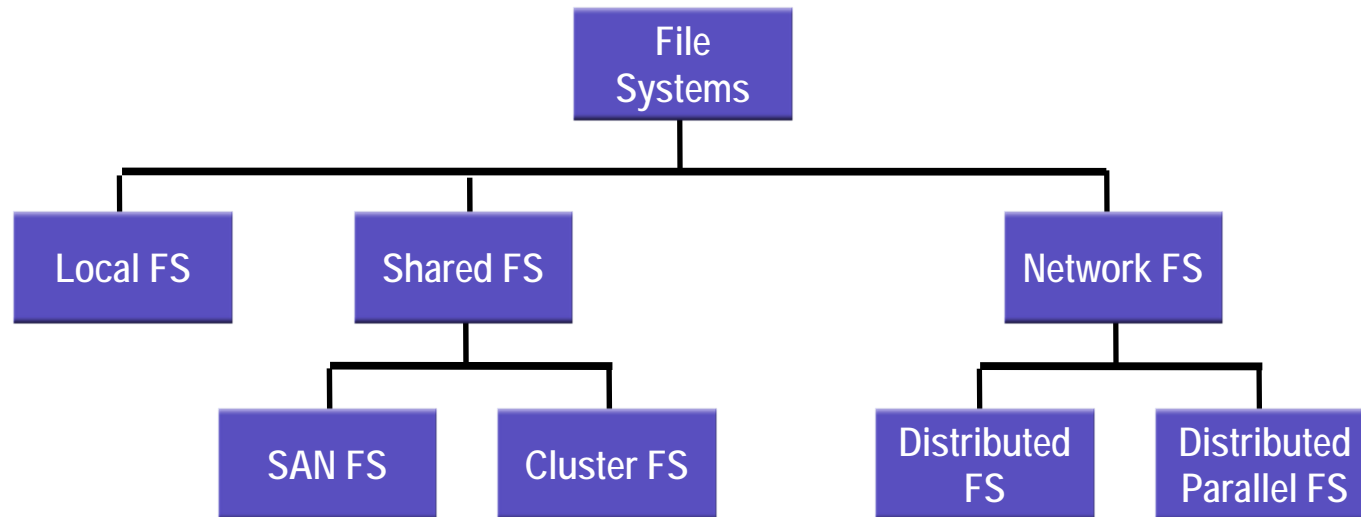
- 9P The Plan 9 and Inferno distributed file system
- AFS (Andrew File System)
- AppleShare
- Arla (file system)
- Coda
- CXFS (Clustered XFS) a distributed networked file system designed by Silicon Graphics (SGI) specifically to be used in a SAN
- Distributed File System (DCE)
- Distributed File System (Microsoft)
- Freenet
- Global File System (GFS)
- Google File System (GFS)
- IBRIX Fusion™
- InterMezzo
- Isilon OneFS™
- Lustre (Sun Microsystems)



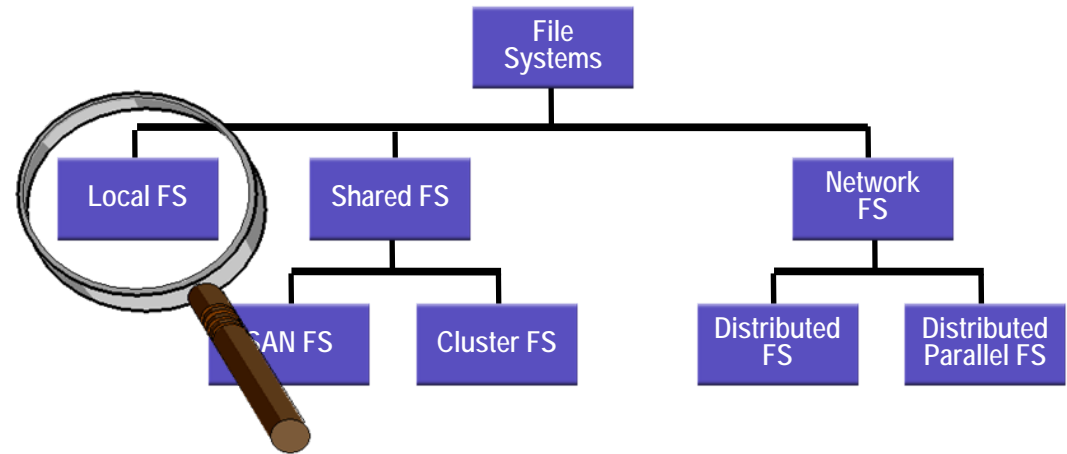
- NFS
- OpenAFS
- Server message block (SMB) (aka Common Internet File System (CIFS) or Samba file system)
- Xsan (a storage area network (SAN) filesystem from Apple Computer, Inc.)
- archfs (archive)
- cdfs (reading and writing of CDs)
- cfs (caching)
- Davfs2 (WebDAV)
- Devfs
- ftpfs (ftp access)
- fuse (filesystem in userspace, like lufs but better maintained)
- GPFS an IBM cluster file system
- JFFS/JFFS2 (filesystems designed specifically for flash devices)
- LUFFS (replace ftpfs, ftp ssh ... access)
- nntps (netnews)
- OCFS (Oracle Cluster File System)



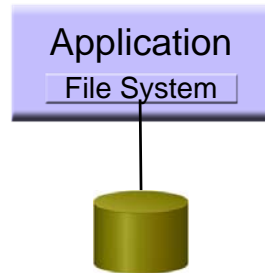
File System Taxonomy



- File System Basics
- File Systems Taxonomy
- **Local FS**
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- Distributed FS
- Distributed Parallel FS
- Scale-Out NAS
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments



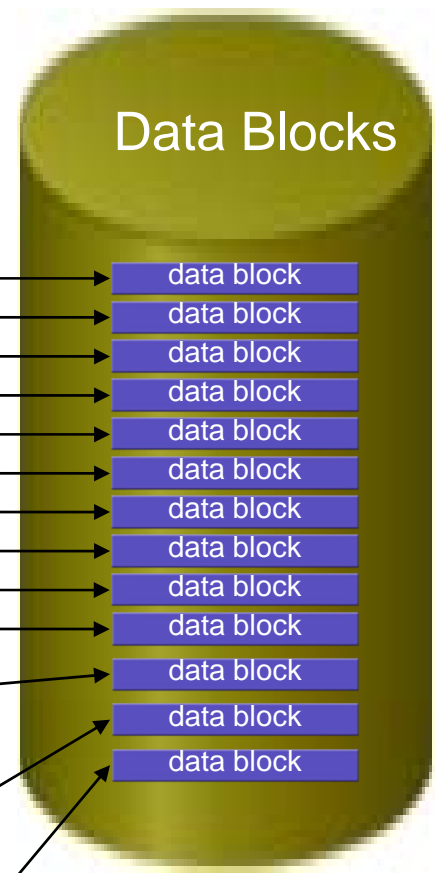
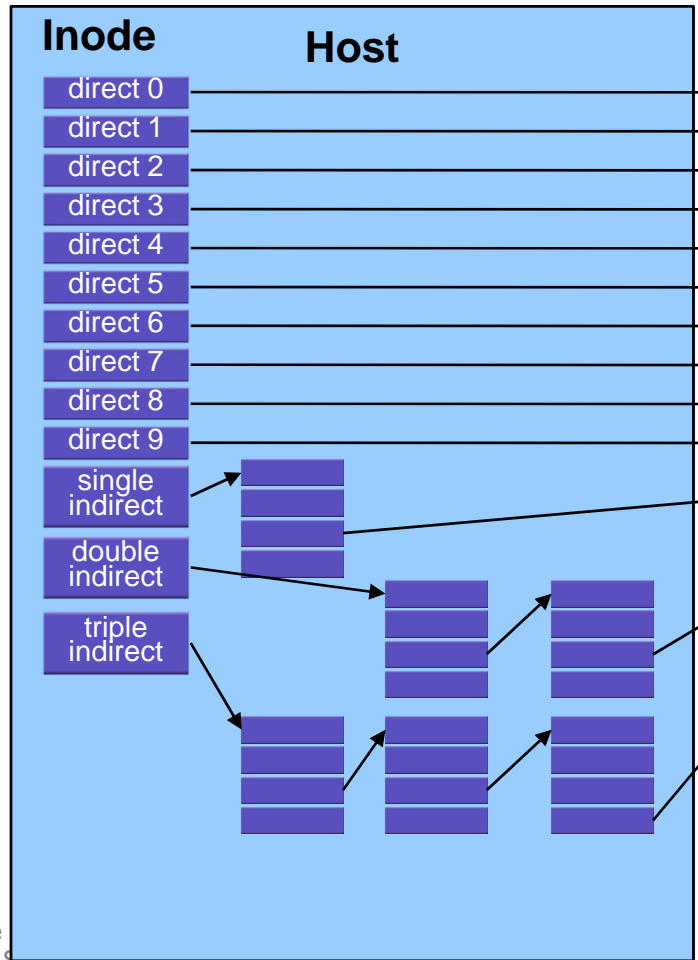
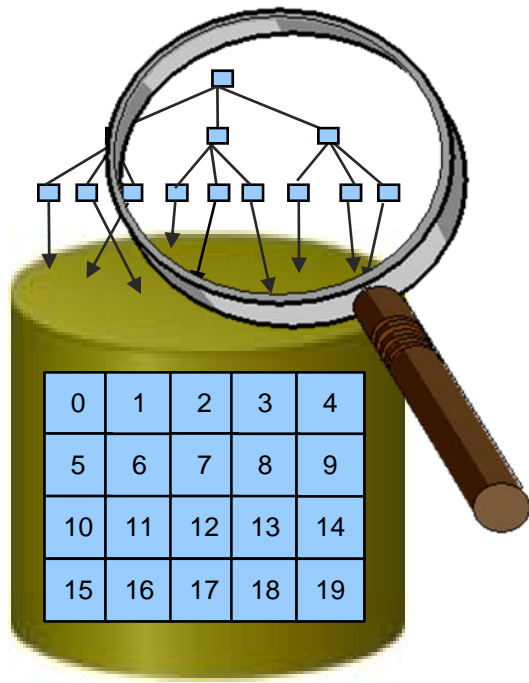
Local FS



➤ FS is **co-located** with application server

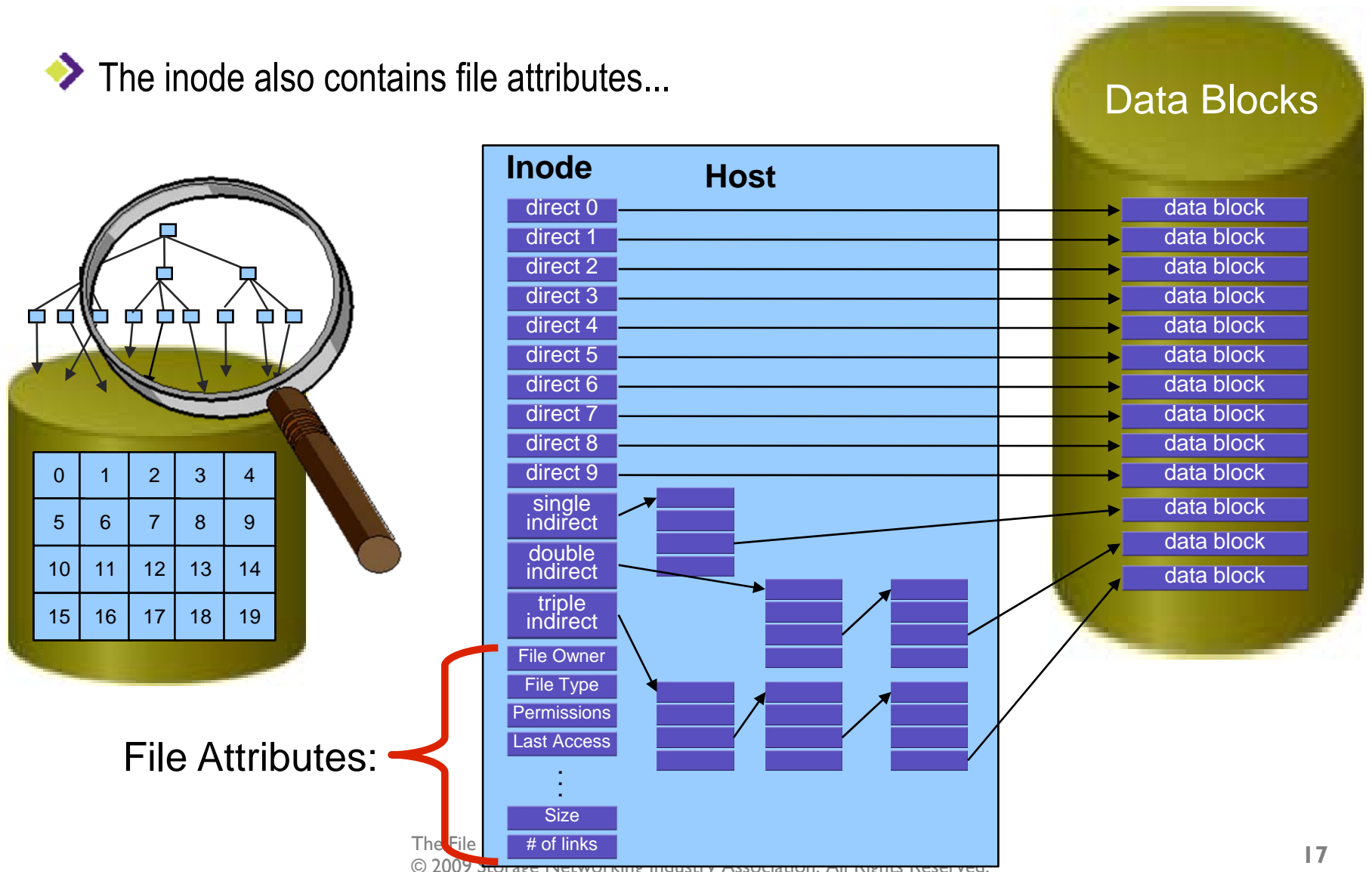
Traditional File System - Inode

➤ When a file system is created, data structures that contain information about files are created. Each file has an inode and is identified by an inode number (often referred to as an "i-number" or "inode") in the file system where it resides.

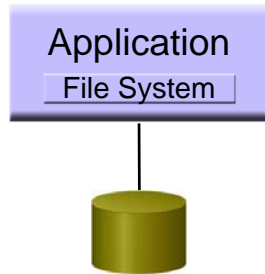


Traditional File System - Inode

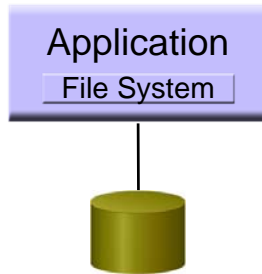
➤ The inode also contains file attributes...



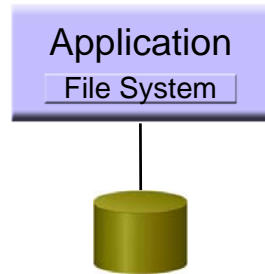
Local FS



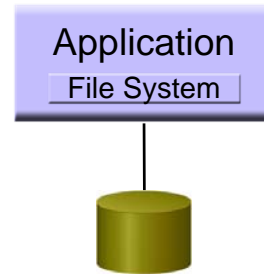
Local FS



Local FS

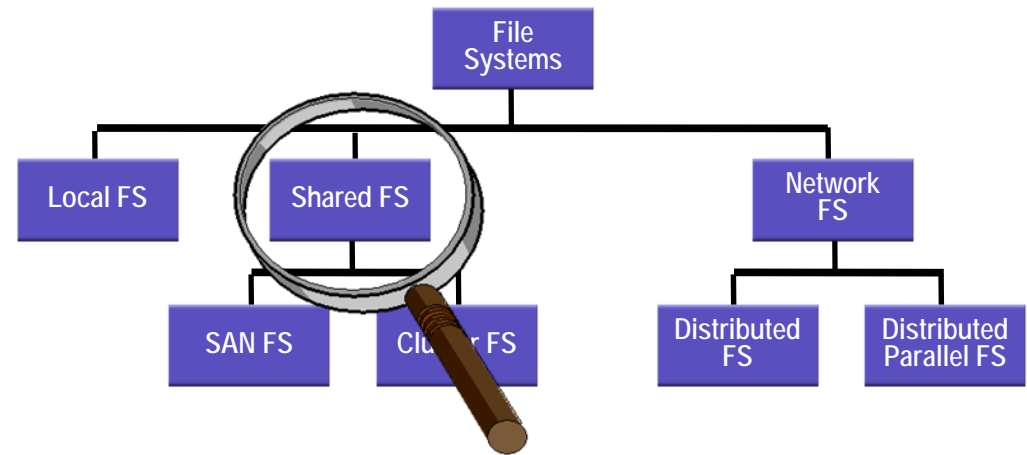


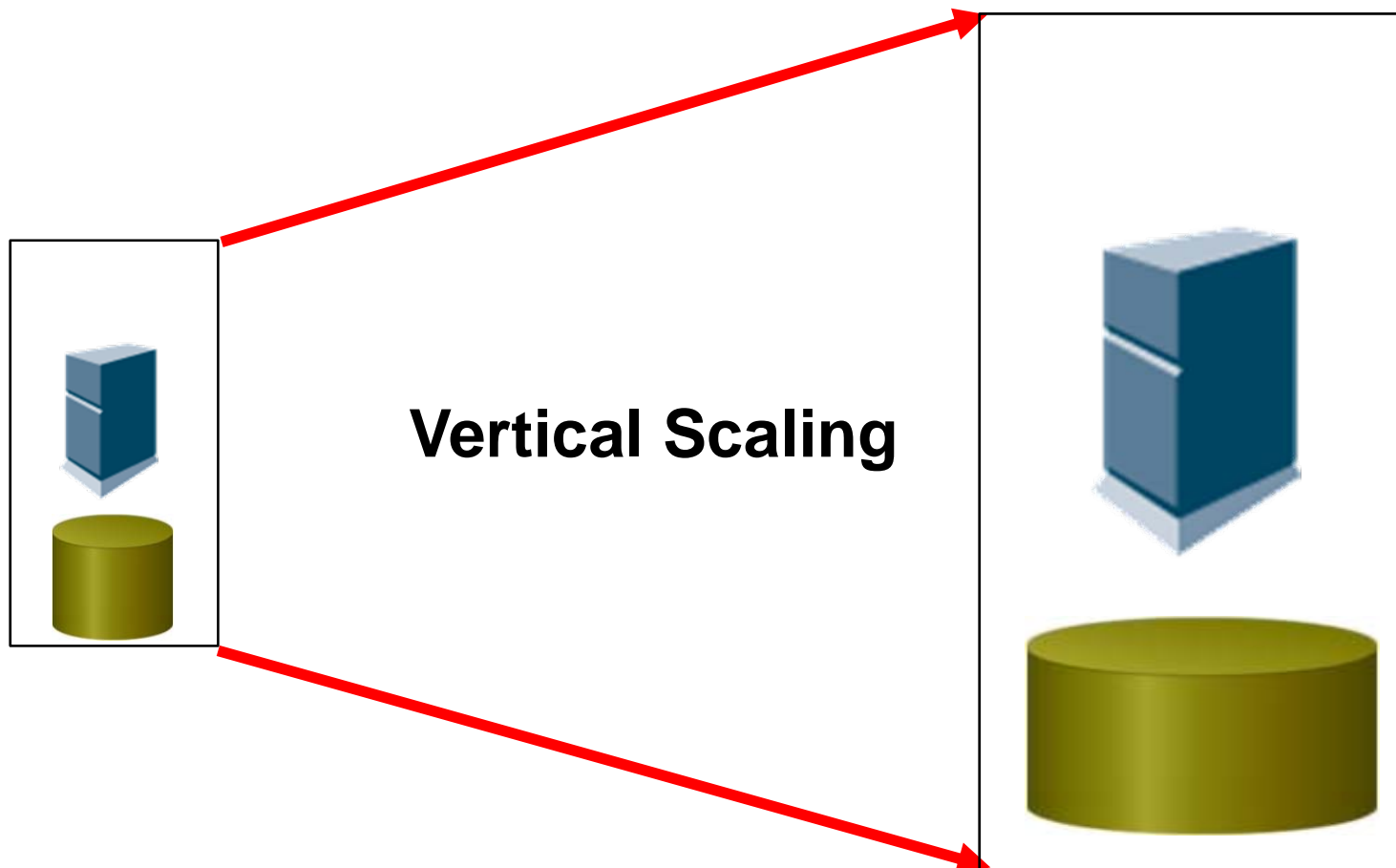
Local FS



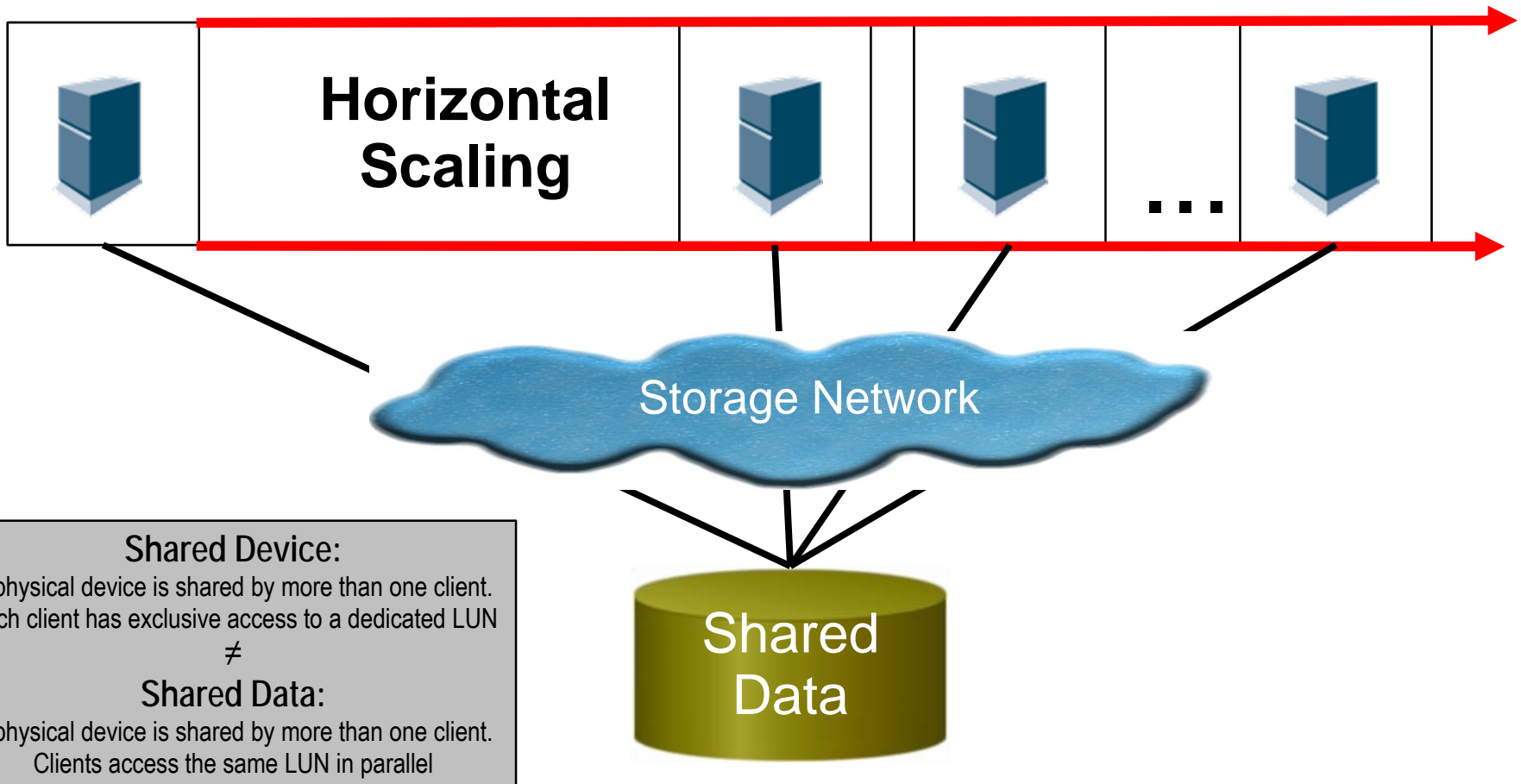
➤ **Islands of storage** (no data sharing)

- File System Basics
- File Systems Taxonomy
- Local FS
- **Shared FS / Global FS**
 - ◆ **SAN FS, Cluster FS**
- Network FS
- Distributed FS
- Distributed Parallel FS
- Scale-Out NAS
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments



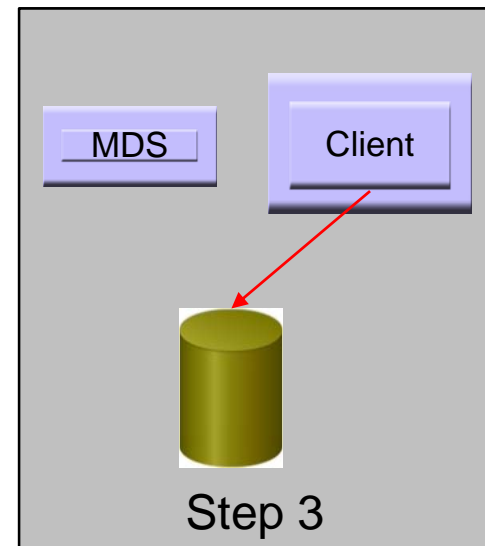
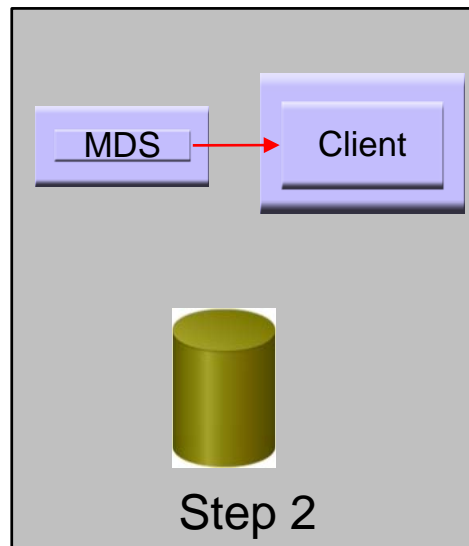
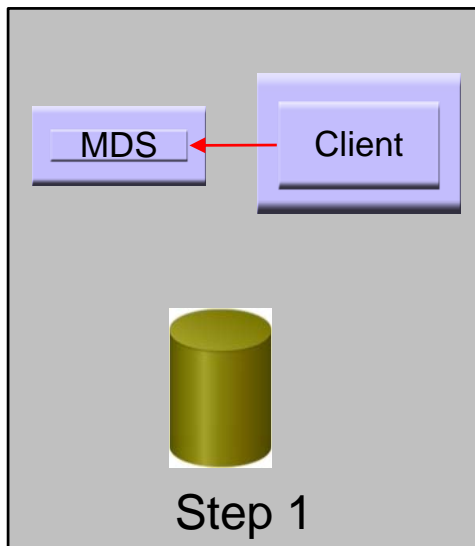


Scale-Out with Shared FS

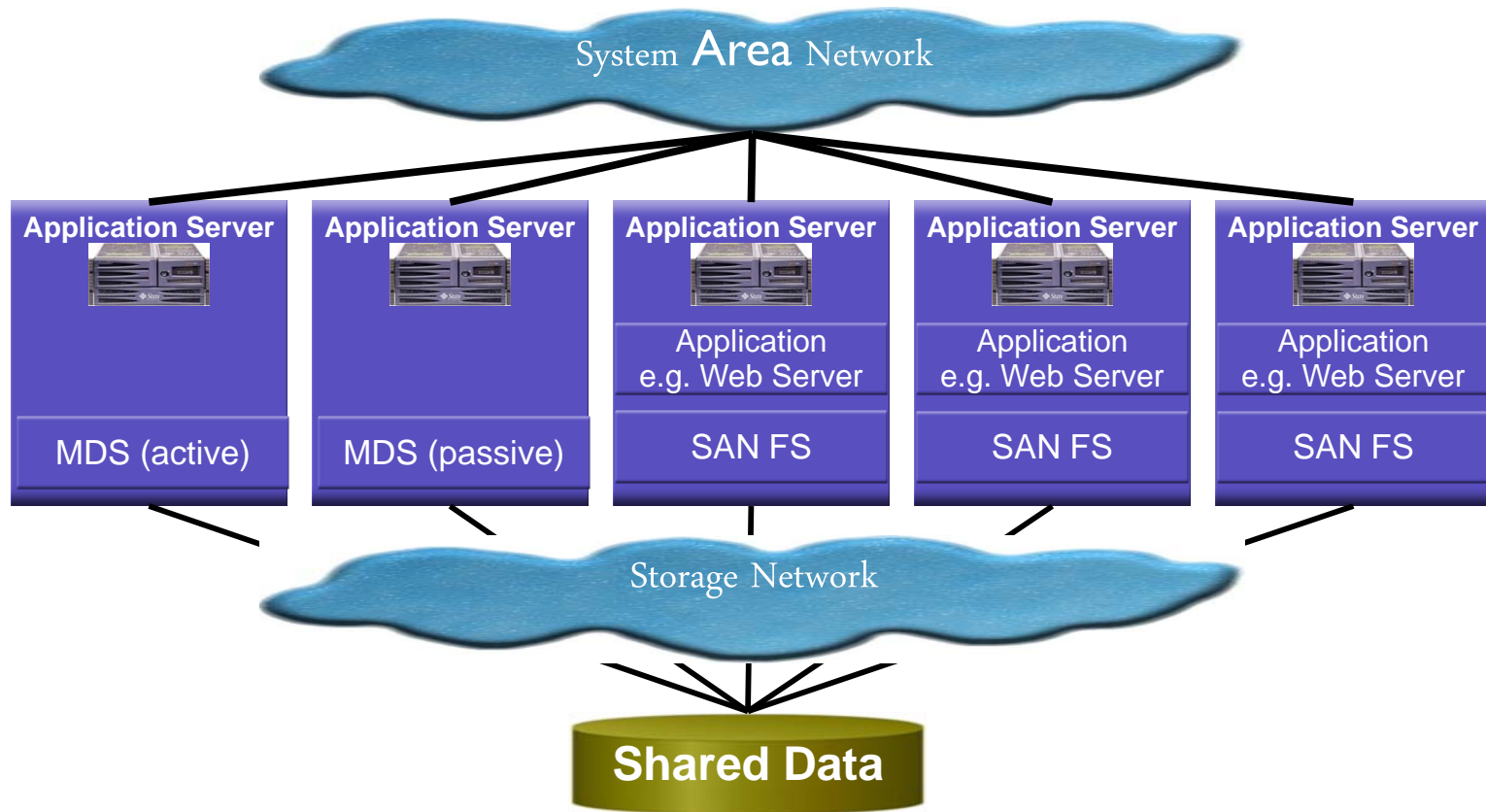


Shared FS/ Global FS Data Access

- Separation between logical and physical placement
- Separate Metadata Server (MDS)
- File access is a three-step transaction...

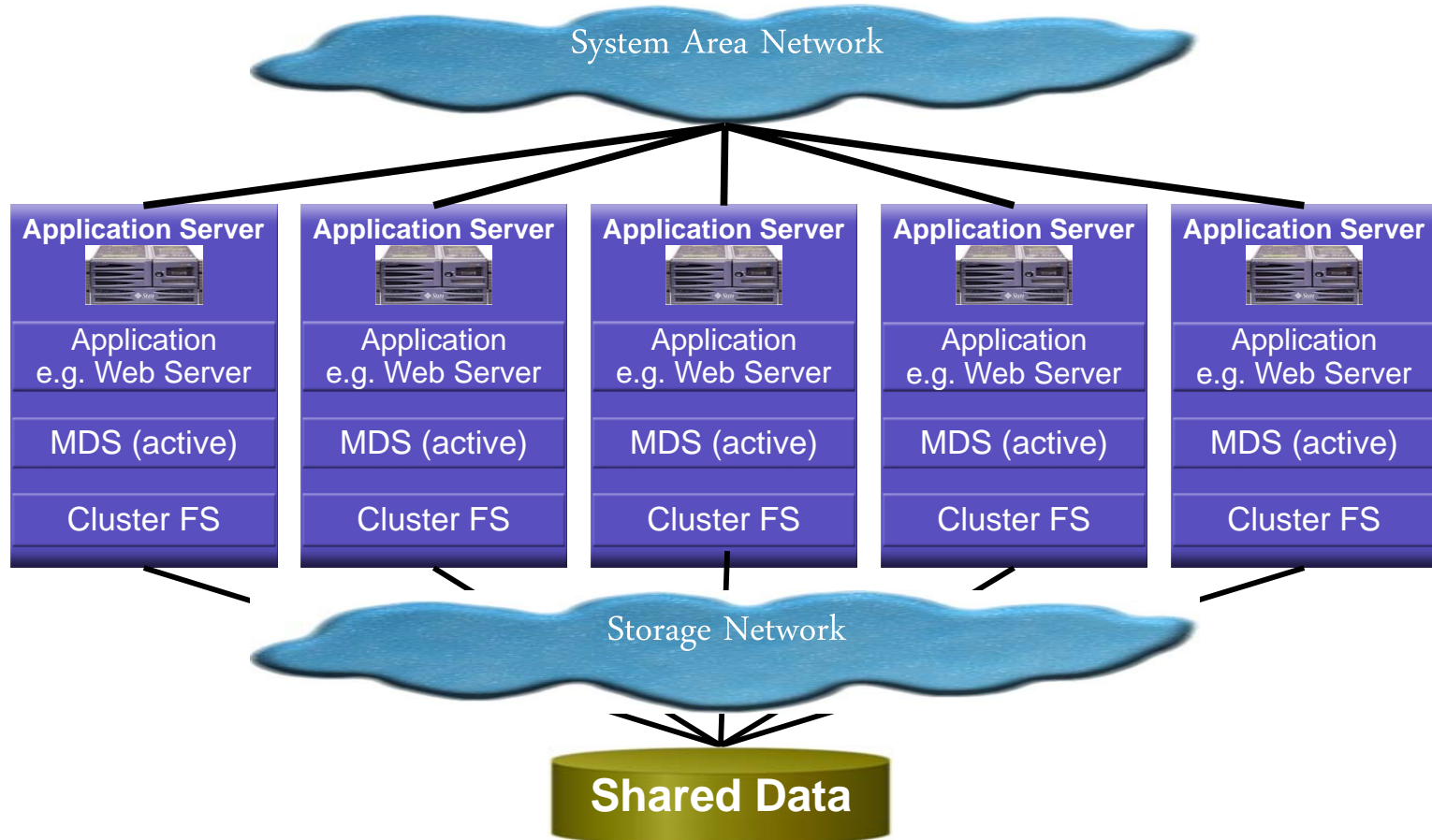


Shared FS / Global FS – SAN FS



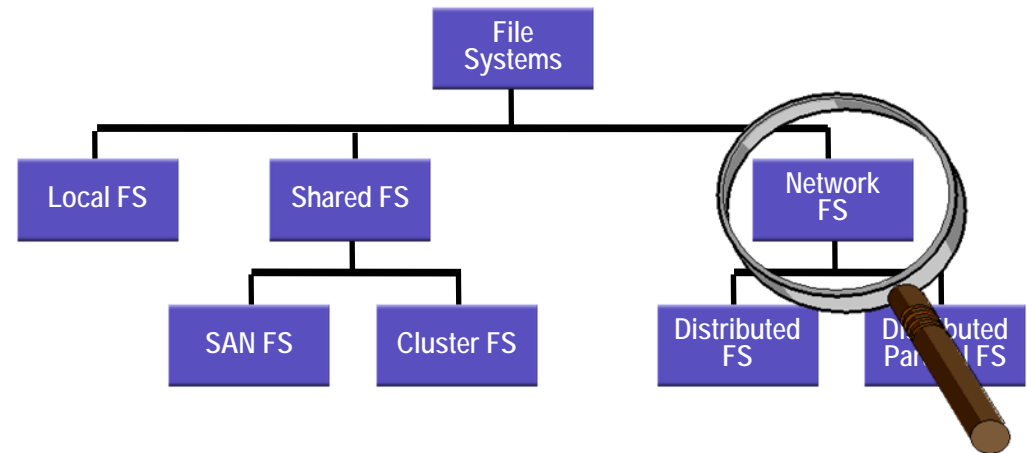
- MDS is not part of each node (i.e. master/slave - asymmetric)
- Heterogeneous with unlimited number of nodes
- Unlimited distance between nodes

Shared FS / Global FS – Cluster FS



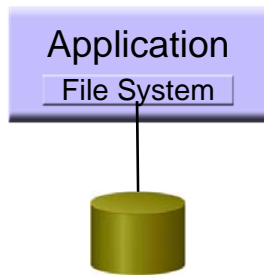
- MDS is part of each (cluster) node (i.e. peer-to-peer - symmetric)
- Homogeneous with limited number of nodes
- Limited distance between (cluster) nodes

- File System Basics
- File Systems Taxonomy
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- **Network FS**
- Distributed FS
- Distributed Parallel FS
- Scale-Out NAS
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments

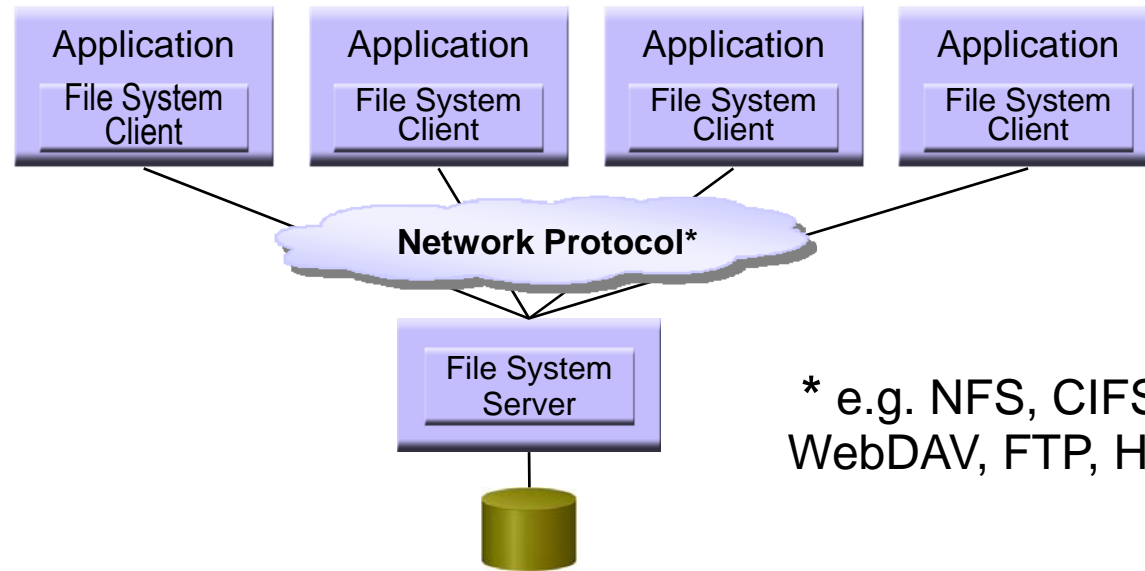


Network Files System- aka Proxy FS

Local FS



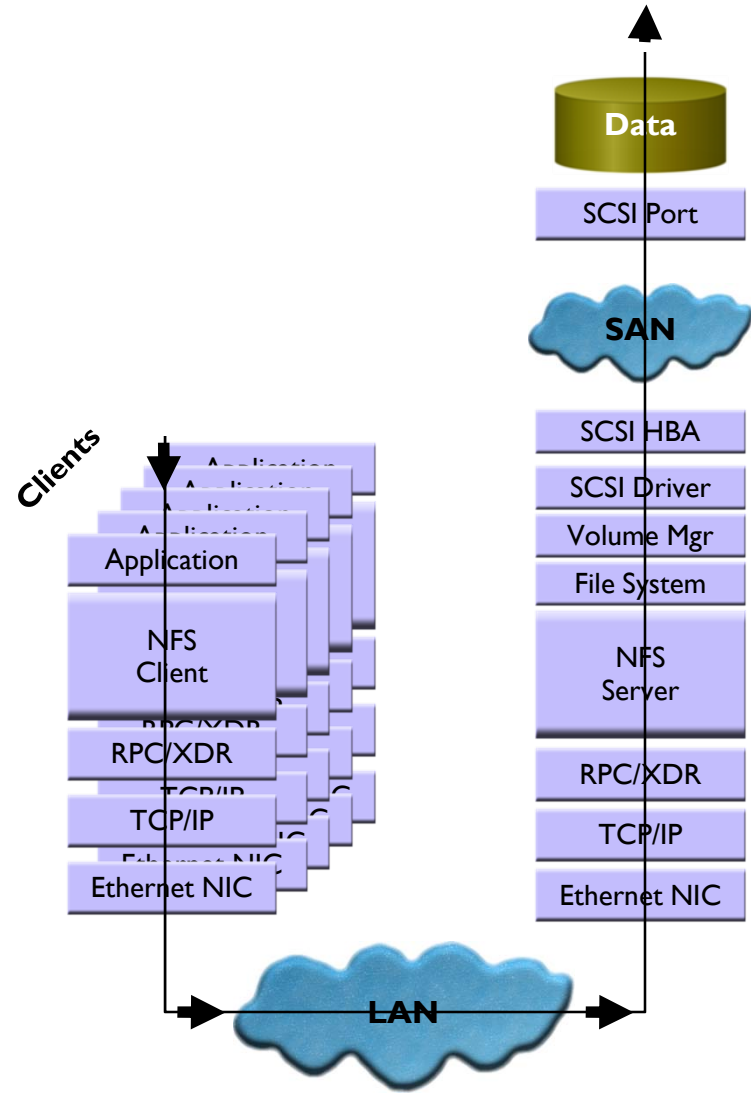
Network FS



* e.g. NFS, CIFS, AFP, WebDAV, FTP, HTTP, ...

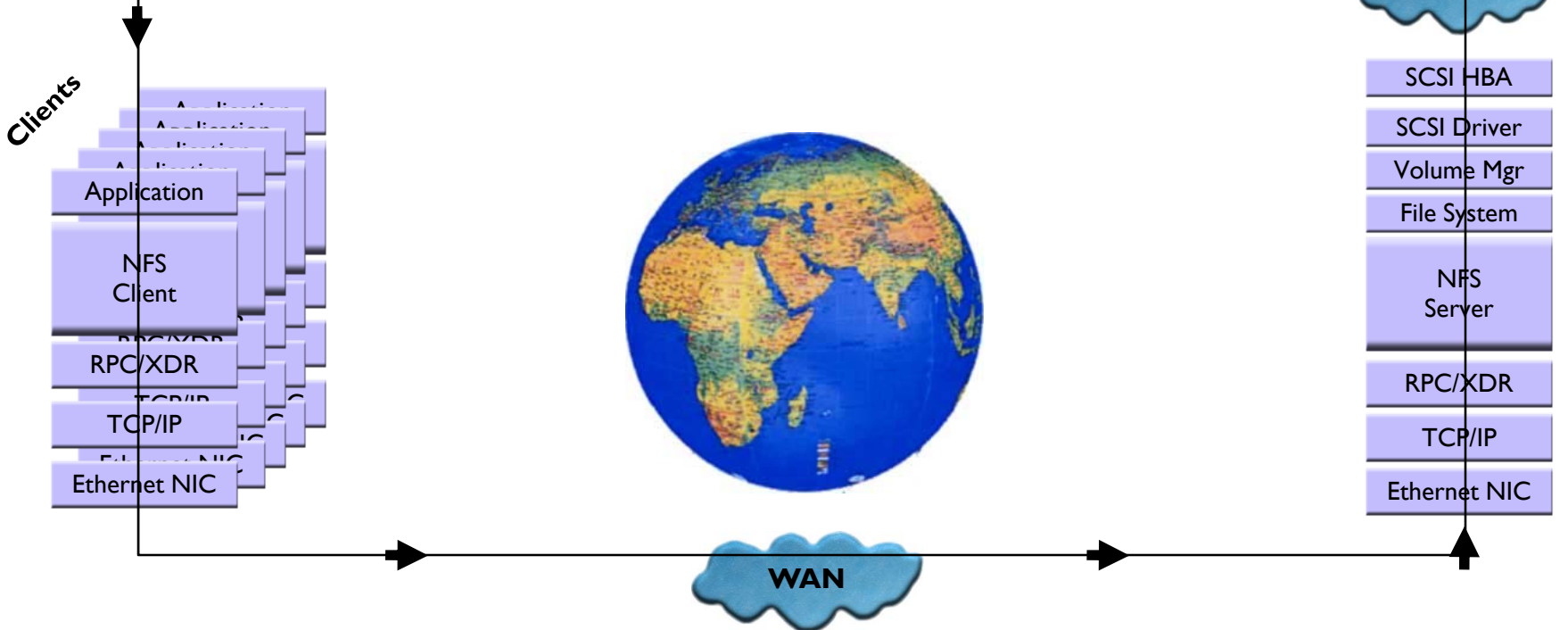
- A network file system is any file system that supports **sharing of files over a computer network protocol** between one or more file systems clients and a file system server

Network FS Stack (e.g. NFS)



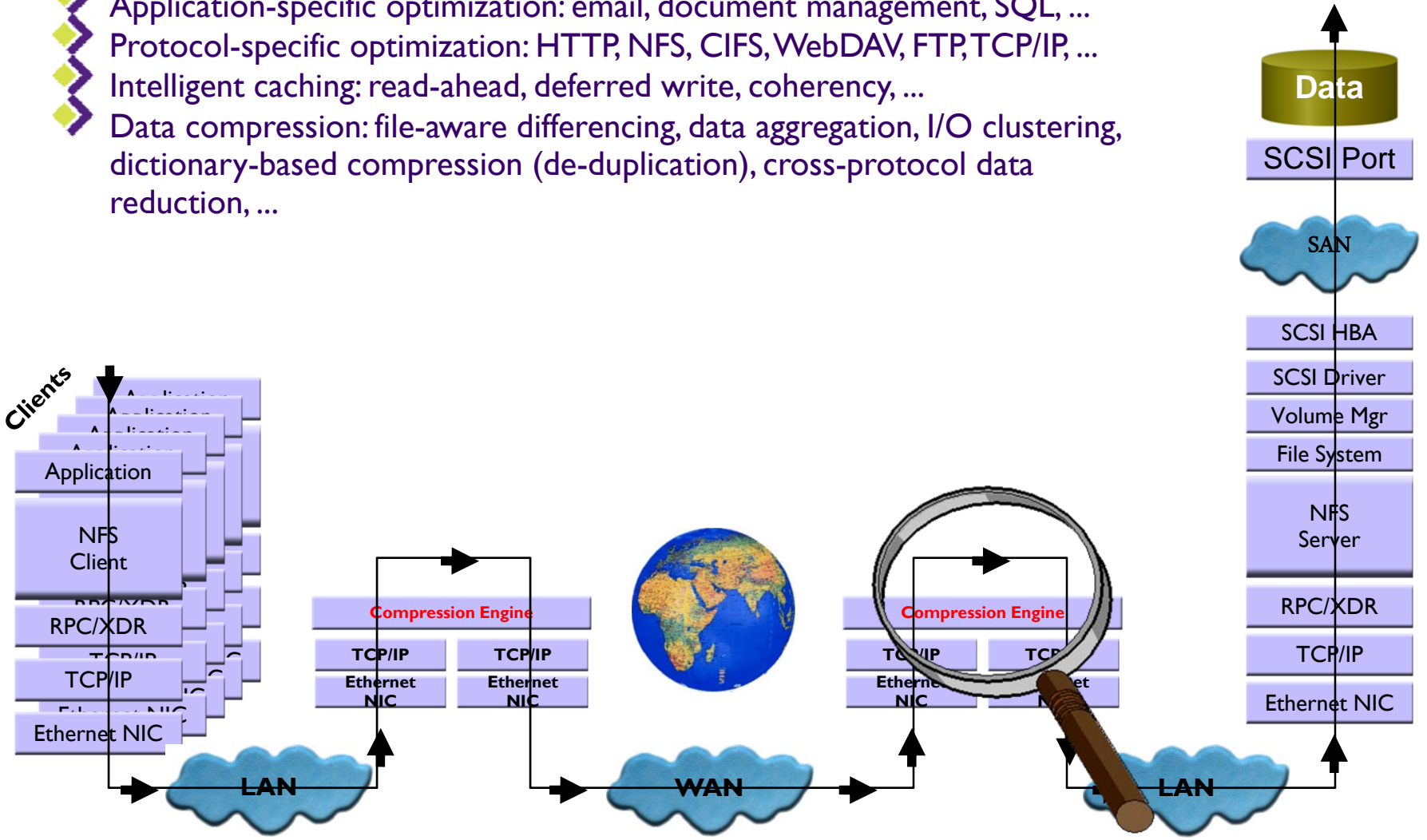
Network FS in a Distributed World

- Consolidating file and storage resources into the data center eases management, administration, cost, and compliance
- Global file sharing and collaboration
- Remote office consolidation and optimization
- **Most application and file access protocols perform poorly over the WAN**



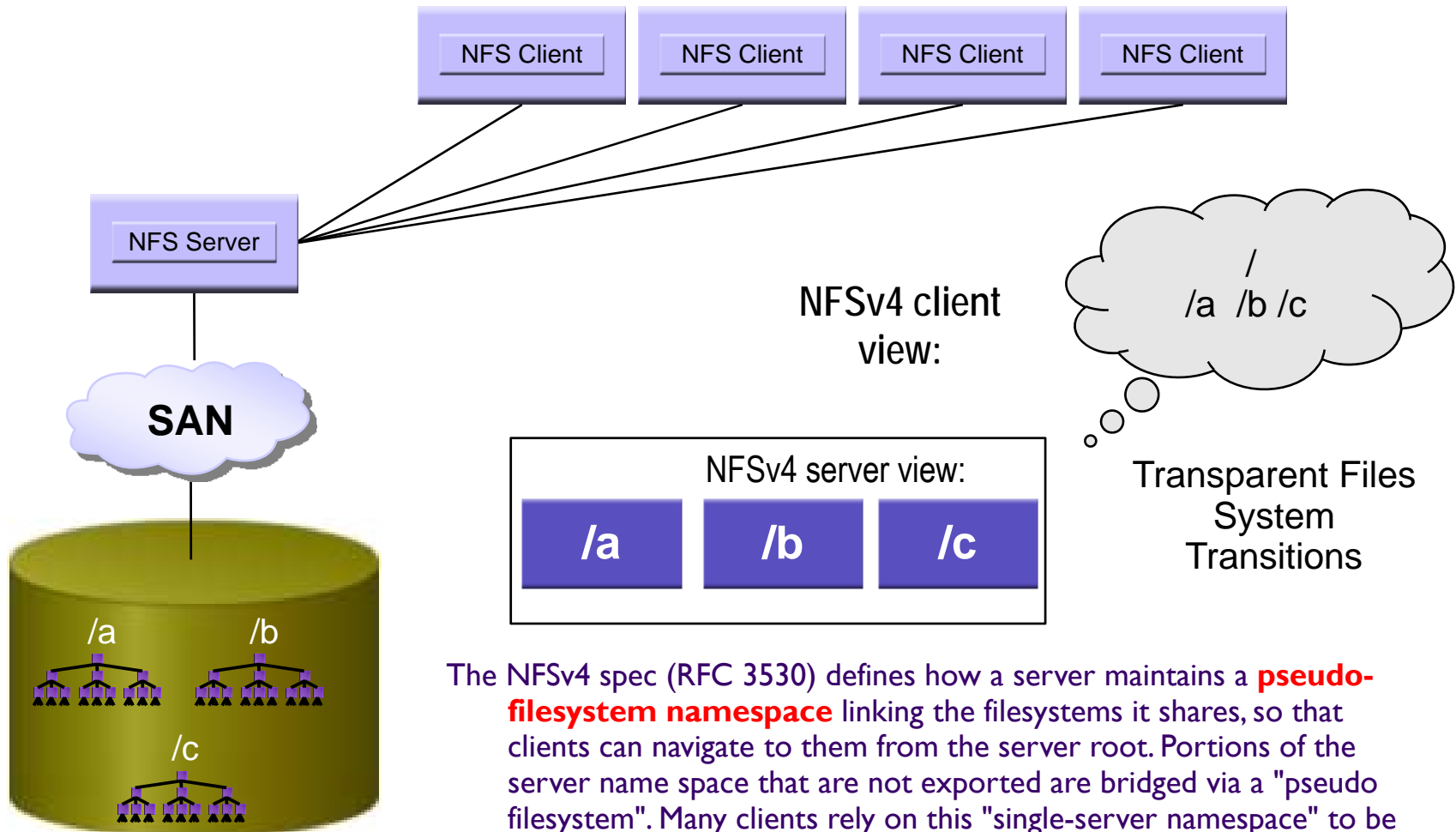
Network Compression

- Application-specific optimization: email, document management, SQL, ...
- Protocol-specific optimization: HTTP, NFS, CIFS, WebDAV, FTP, TCP/IP, ...
- Intelligent caching: read-ahead, deferred write, coherency, ...
- Data compression: file-aware differencing, data aggregation, I/O clustering, dictionary-based compression (de-duplication), cross-protocol data reduction, ...



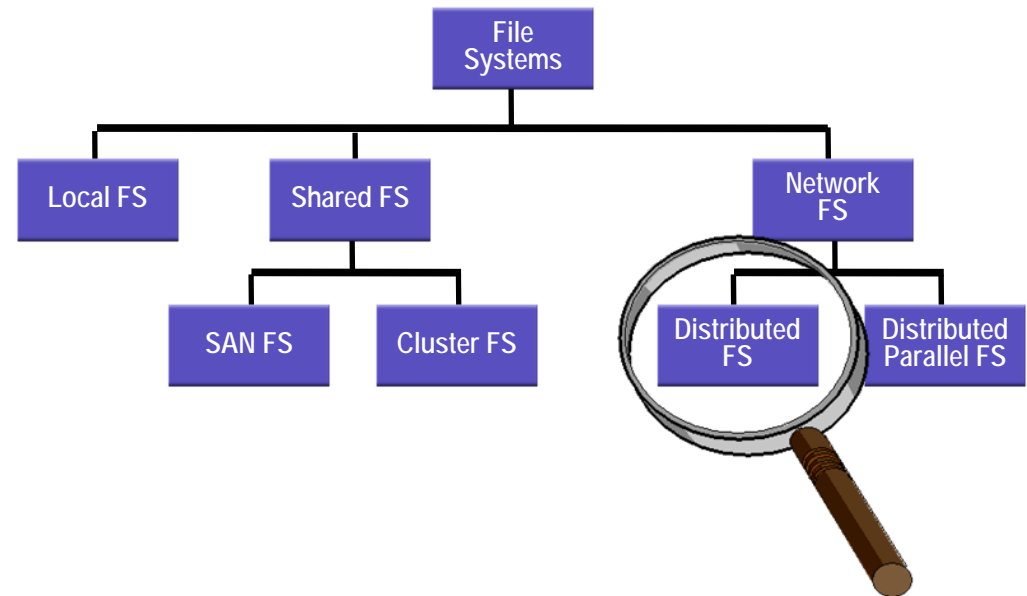
NFSv4 Single-Server Namespace

Server Pseudo FS – aka Shared Name Space

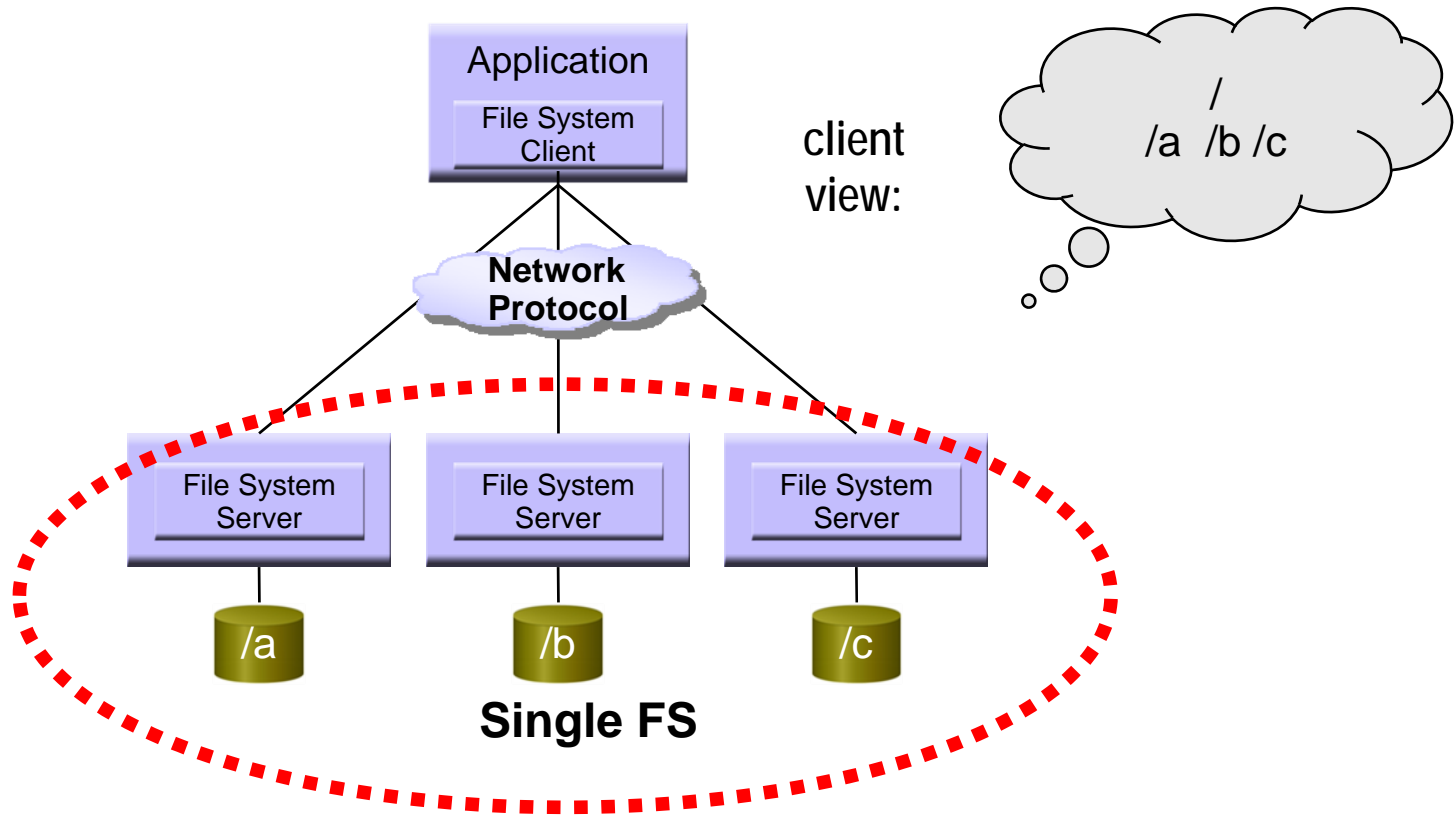


The NFSv4 spec (RFC 3530) defines how a server maintains a **pseudo-file system namespace** linking the filesystems it shares, so that clients can navigate to them from the server root. Portions of the server name space that are not exported are bridged via a "pseudo filesystem". Many clients rely on this "single-server namespace" to be able to access all file systems on the server transparently.

- File System Basics
- File Systems Taxonomy
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- **Distributed FS**
- Distributed Parallel FS
- Scale-Out NAS
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments

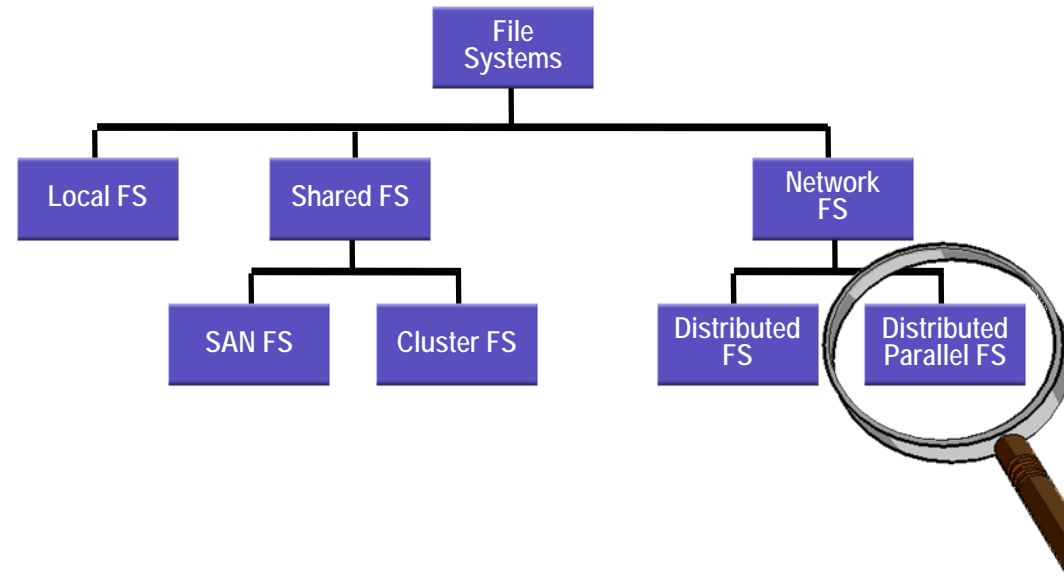


Distributed File System (DFS)



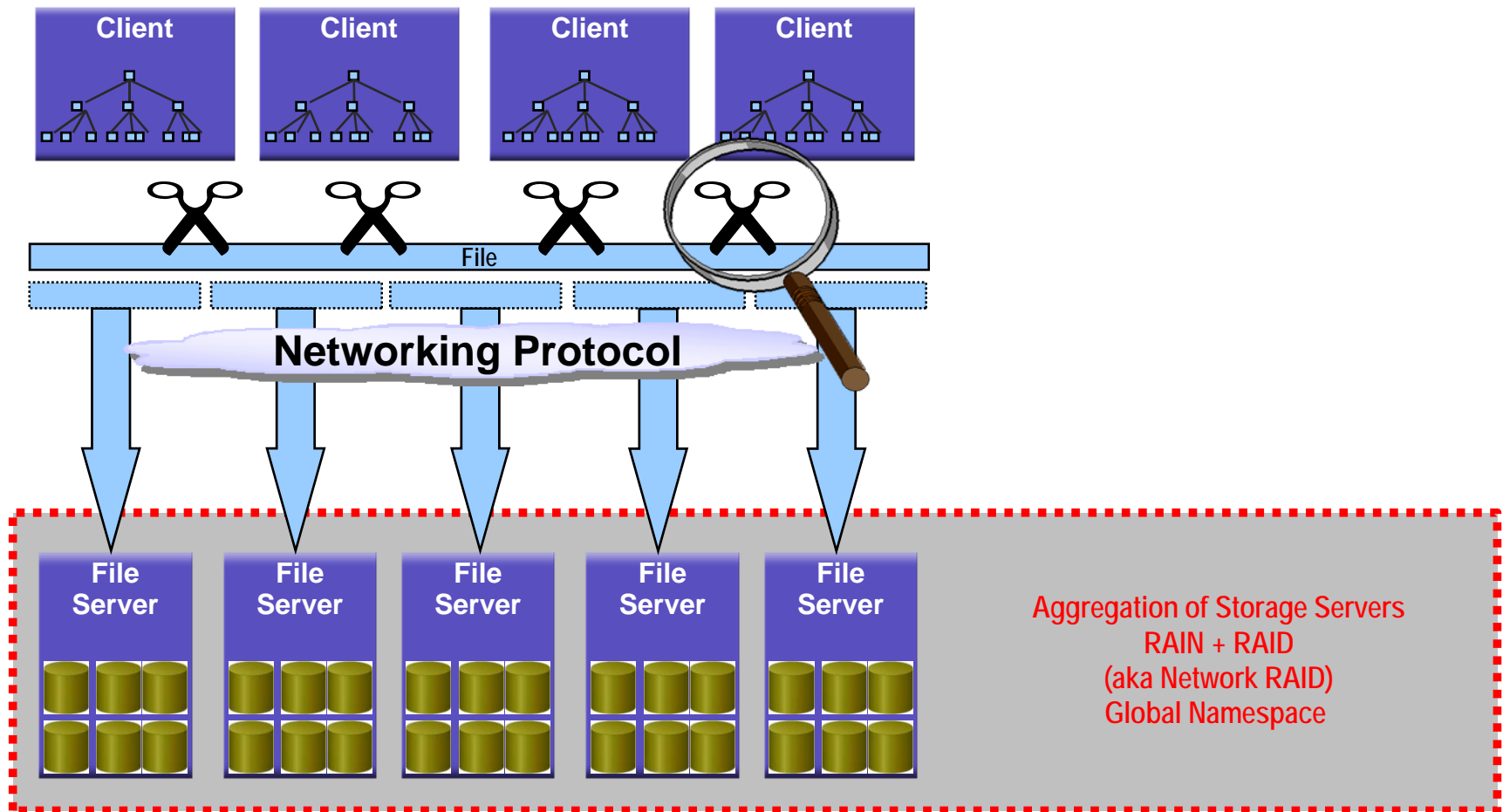
- **A distributed file system is a network file system** whose files are dispersed across file servers (≠ Parallel FS)

- File System Basics
- File Systems Taxonomy
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- Distributed FS
- **Distributed Parallel FS**
- Scale-Out NAS
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments

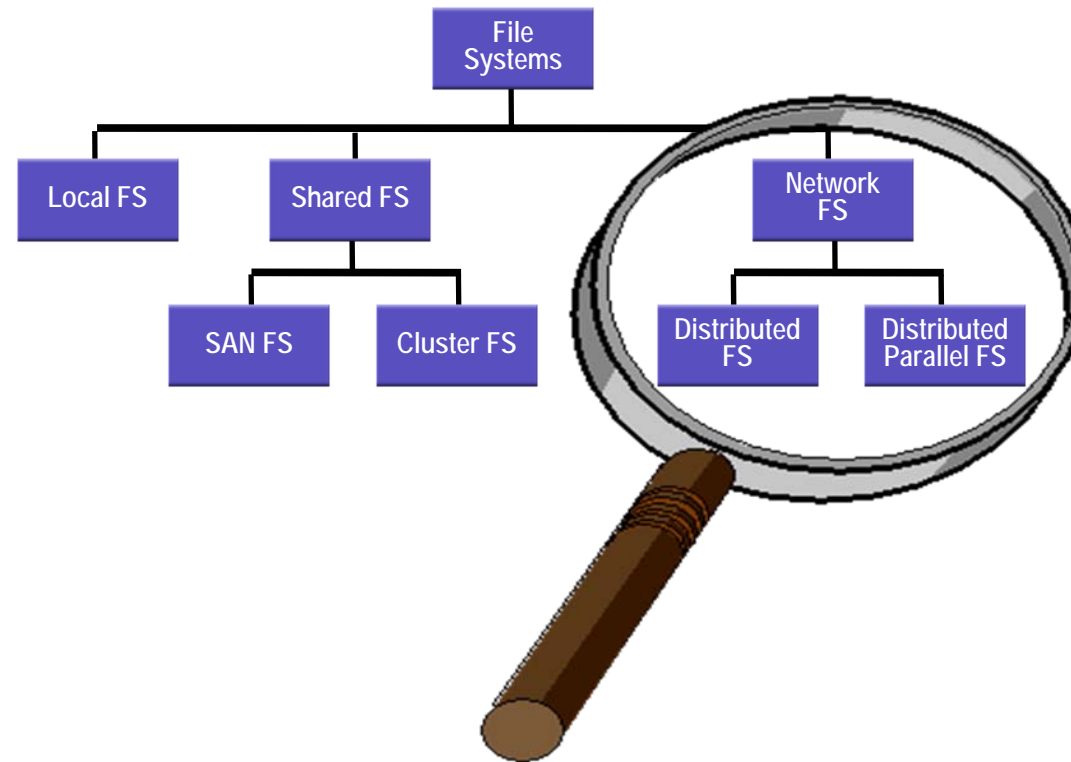


Distributed Parallel File System

- ▶ File Segments distributed across storage nodes – Parallel I/Os

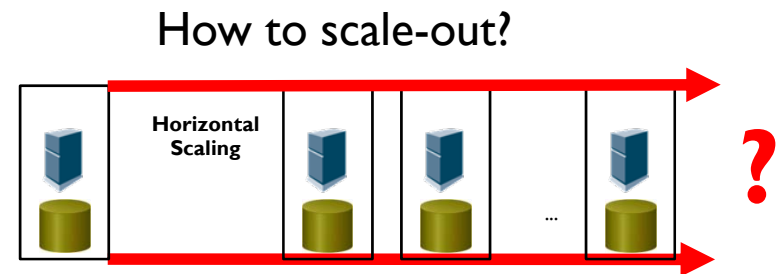
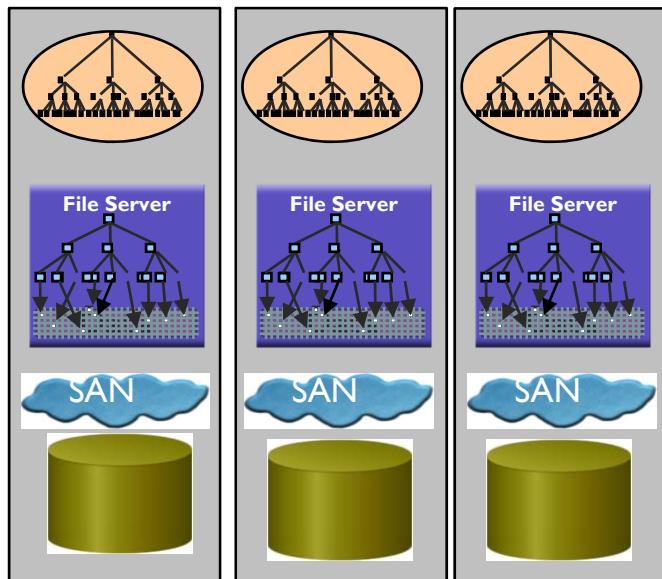
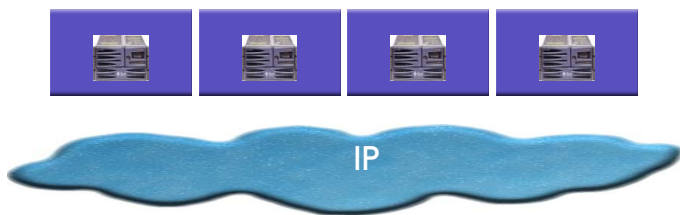


- File System Basics
- File Systems Taxonomy
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- Distributed FS
- Distributed Parallel FS
- **Scale-Out NAS**
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments

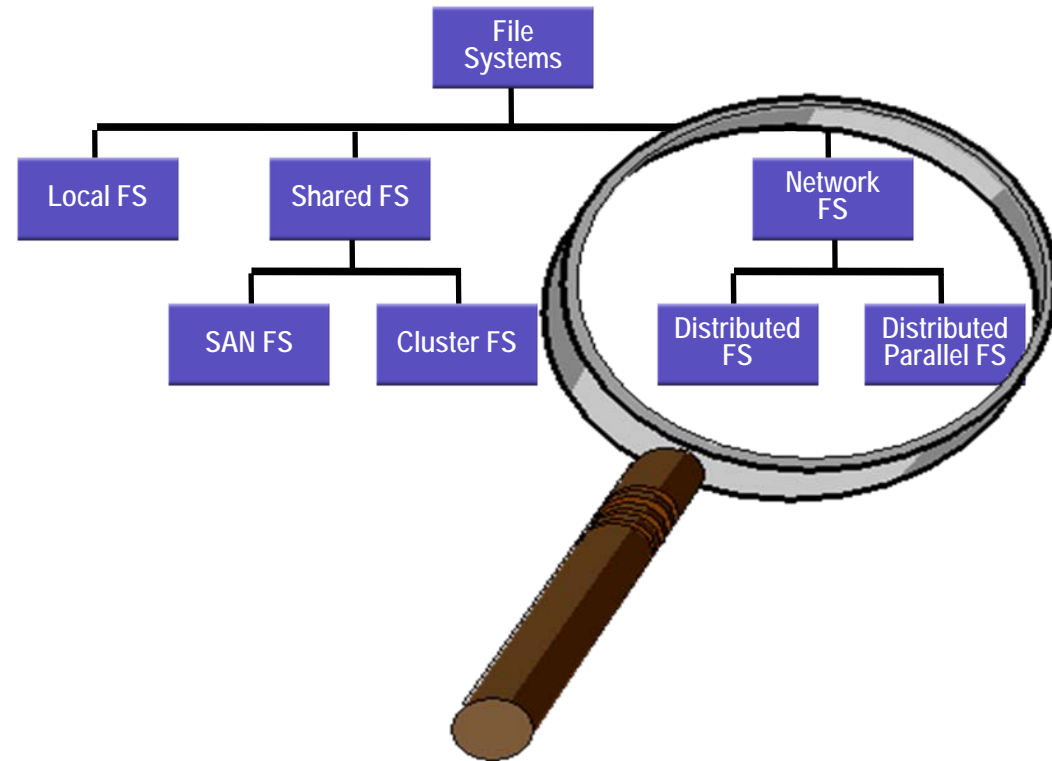


Network Attached Storage (NAS)

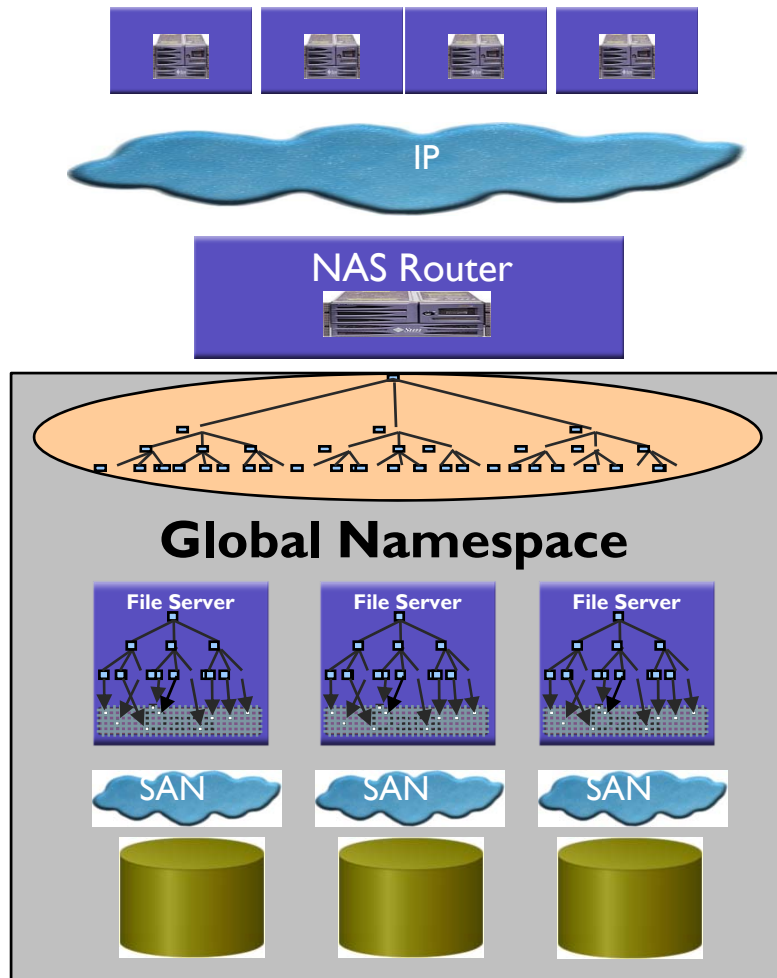
- Storage Islands
- Separated Namespaces
- Multiple mount points



- File System Basics
- File Systems Taxonomy
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- Distributed FS
- Distributed Parallel FS
- **Scale-Out NAS**
 - ◆ **NAS Aggregation**
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments

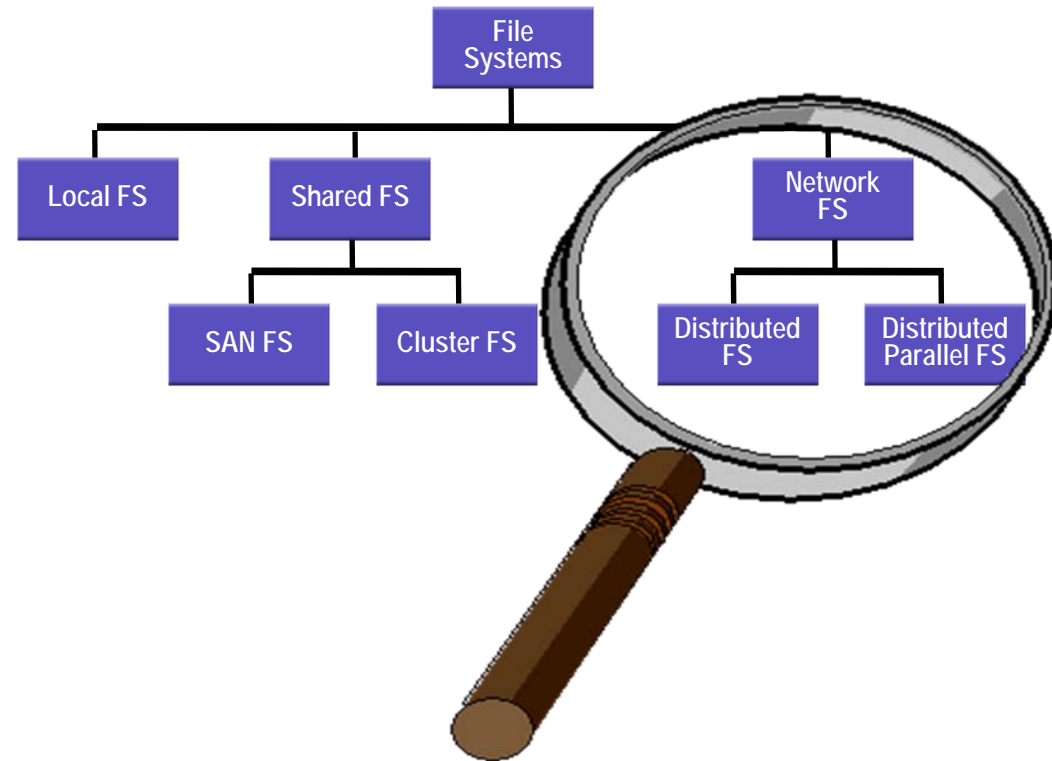


NAS Aggregation & Global Namespace

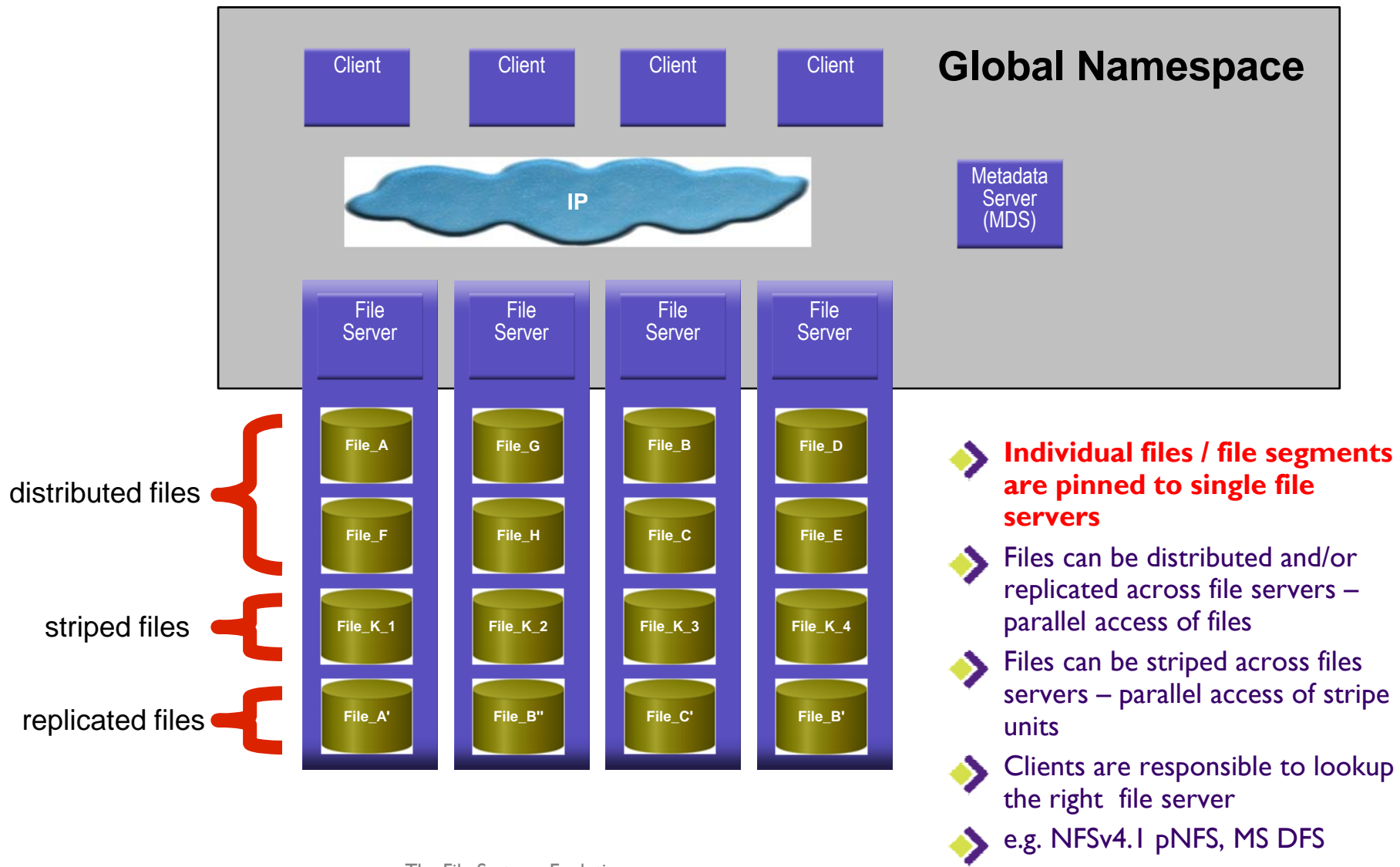


- In-Band Solution
- Aka NAS Router

- File System Basics
- File Systems Taxonomy
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- Distributed FS
- Distributed Parallel FS
- **Scale-Out NAS**
 - ◆ NAS Aggregation
 - ◆ **NAS Virtualization**
 - ◆ NAS Cluster / NAS Grid
- FS Future Developments

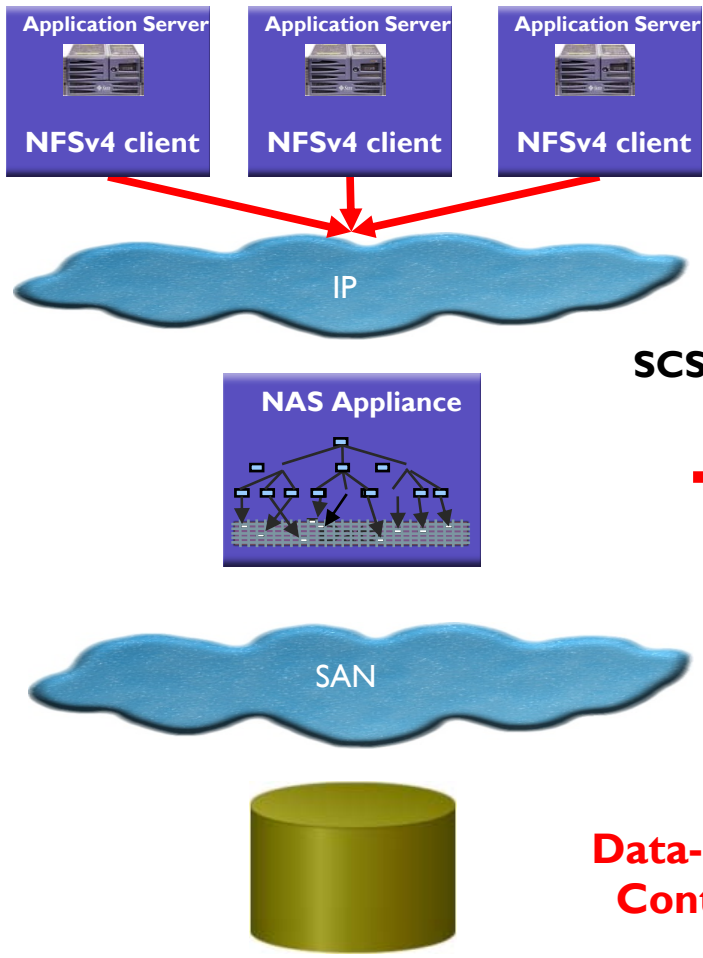


NAS Virtualization - Out-of-Band



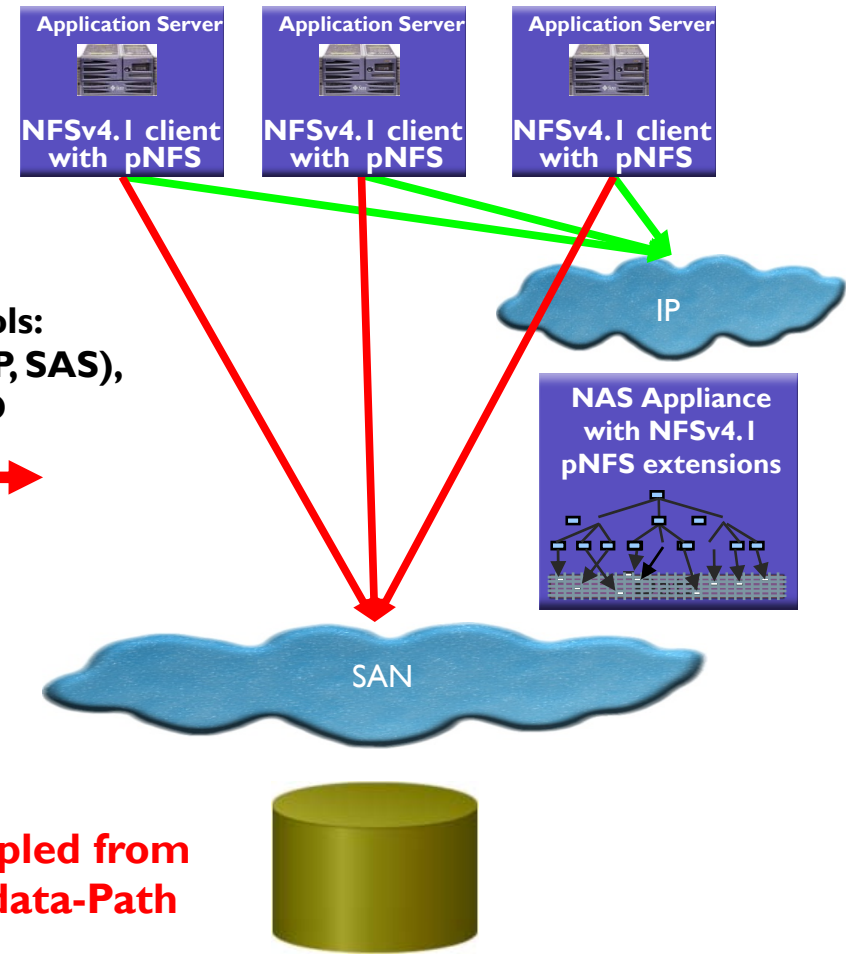
NAS Virtualization – NFS4.1 pNFS

In-Band NAS:



Out-of-Band NAS:

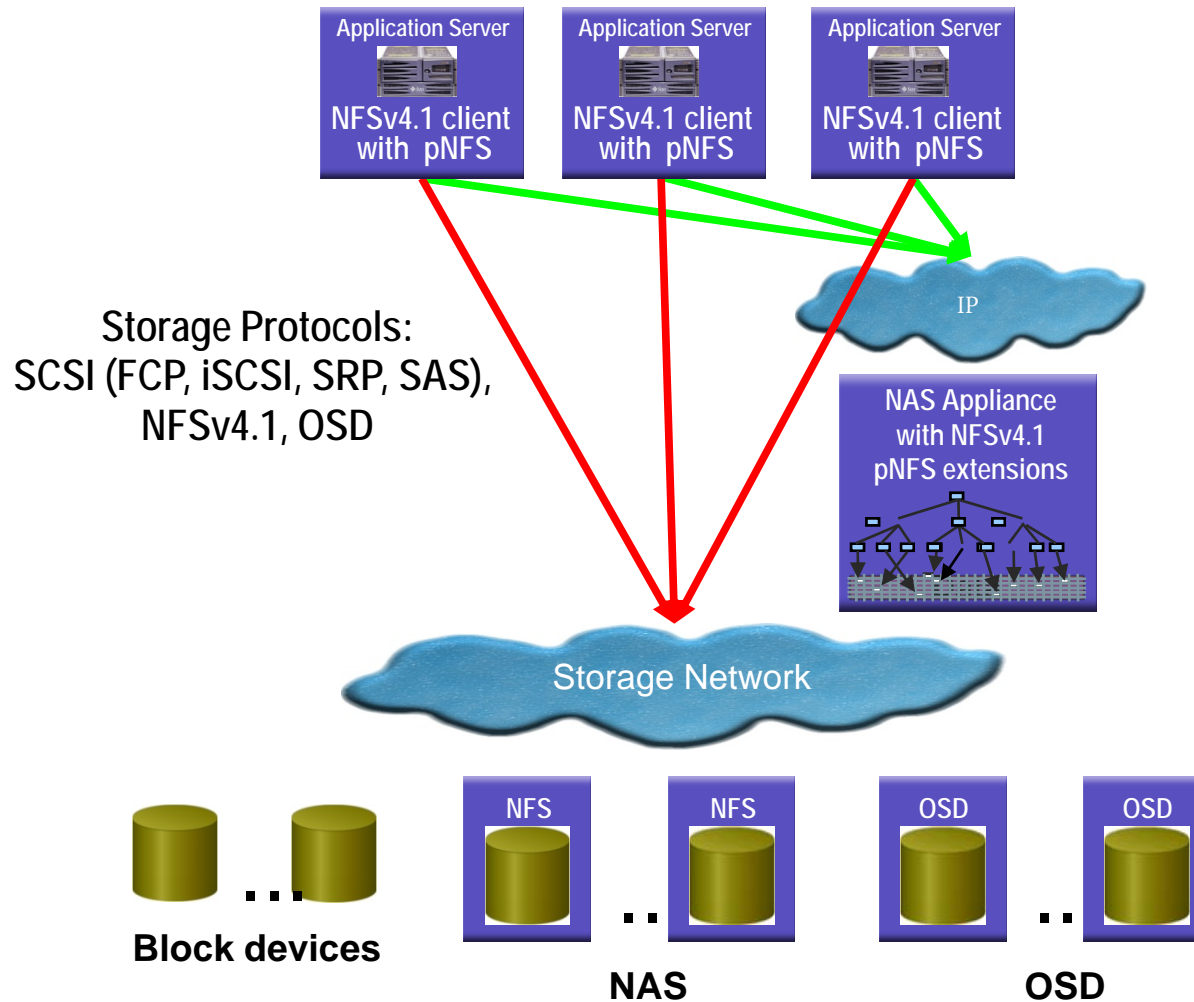
Storage Protocols:
 SCSI (FCP, iSCSI, SRP, SAS),
 NFSv4.1, OSD



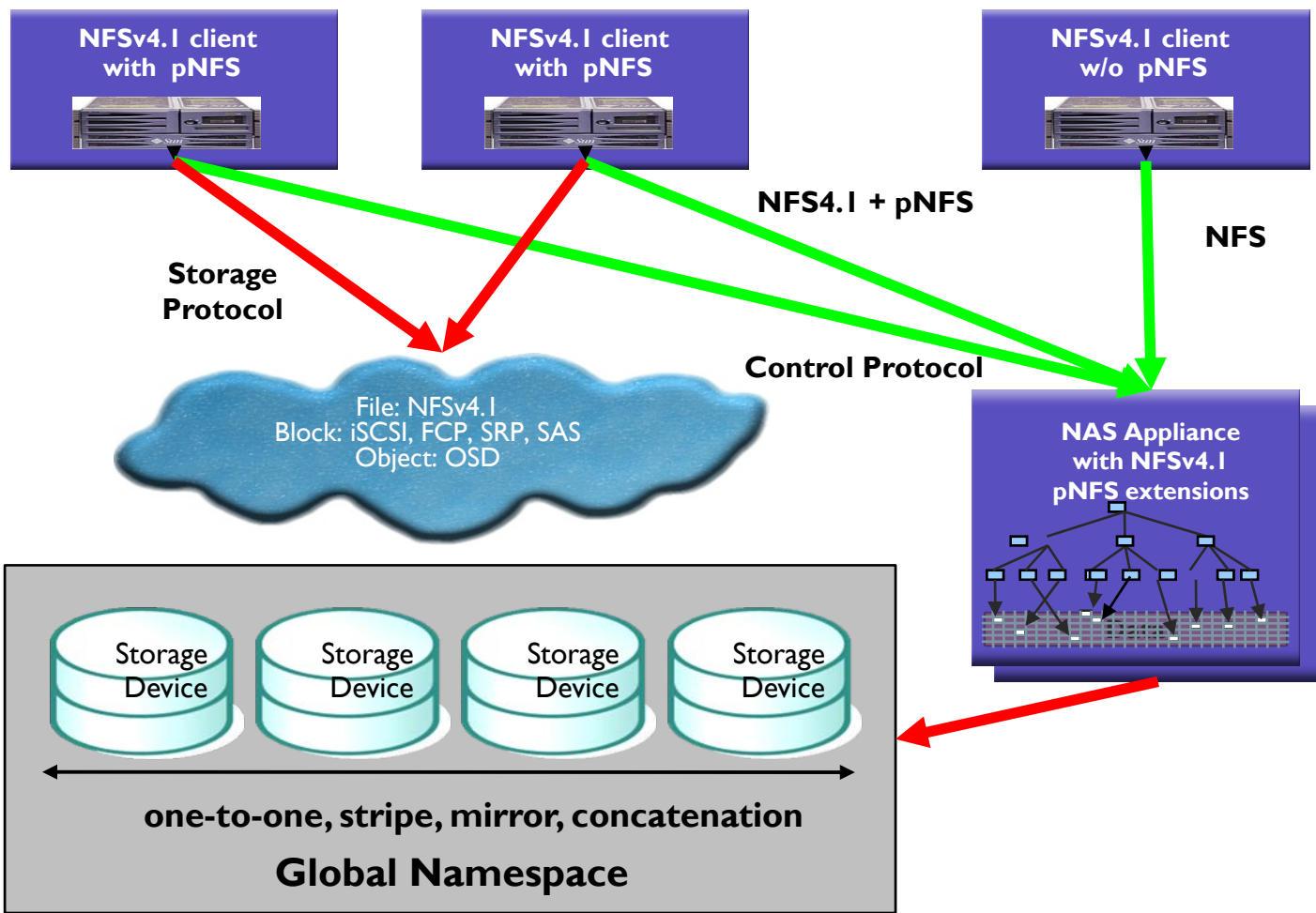
Data-Path is de-coupled from Control- and Metadata-Path

NAS Virtualization – NFS4.1 pNFS

Out-of-Band NAS:



NAS Virtualization – NFS4.1 pNFS

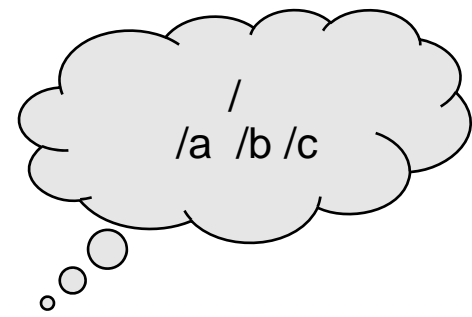
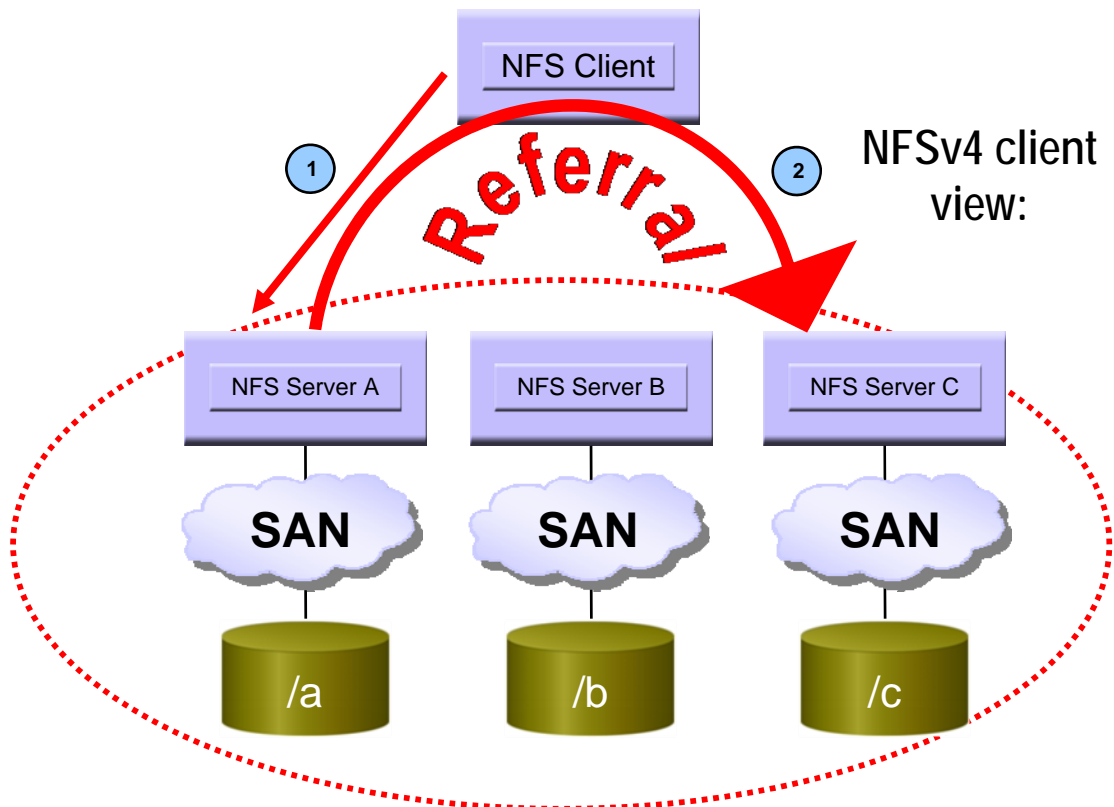


MDS acts as proxy for clients not pNFS enabled

MDS creates Global Namespace

NAS Virtualization – NFS4.1 Referrals SNIA

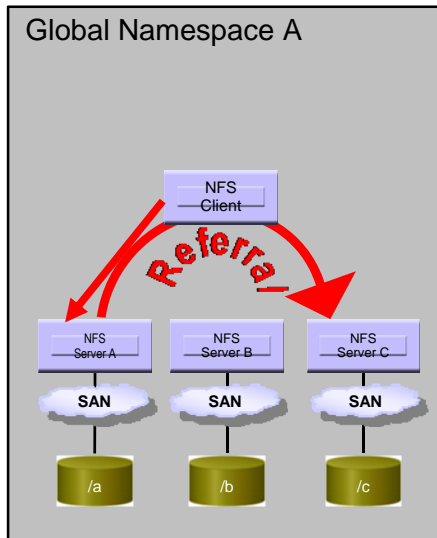
➤ NFSv4.1 – Multi-Server Name space



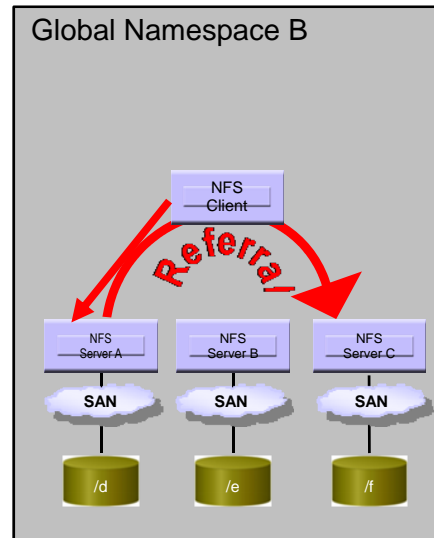
Single FS / Global Namespace

fs_location_info attribute enables:
referral, replicas,
clones, migration

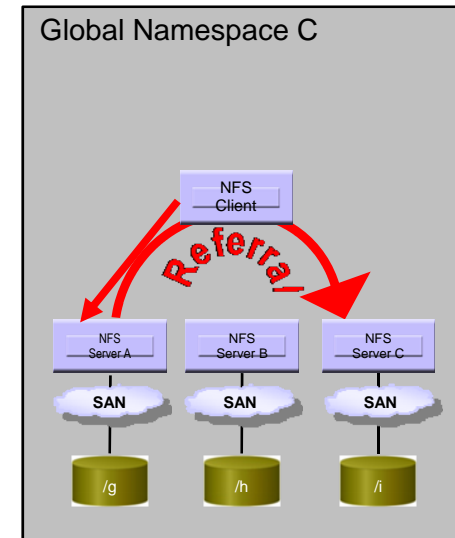
NFSv4.1 supports attributes that allow a namespace to extend beyond the boundaries of a single server through **fs_location_info attribute**. A server can inform a client that data it seeks lives at another location; this is called "referral", and referrals can be used to construct a **Global Namespace**



Collection A of File Servers



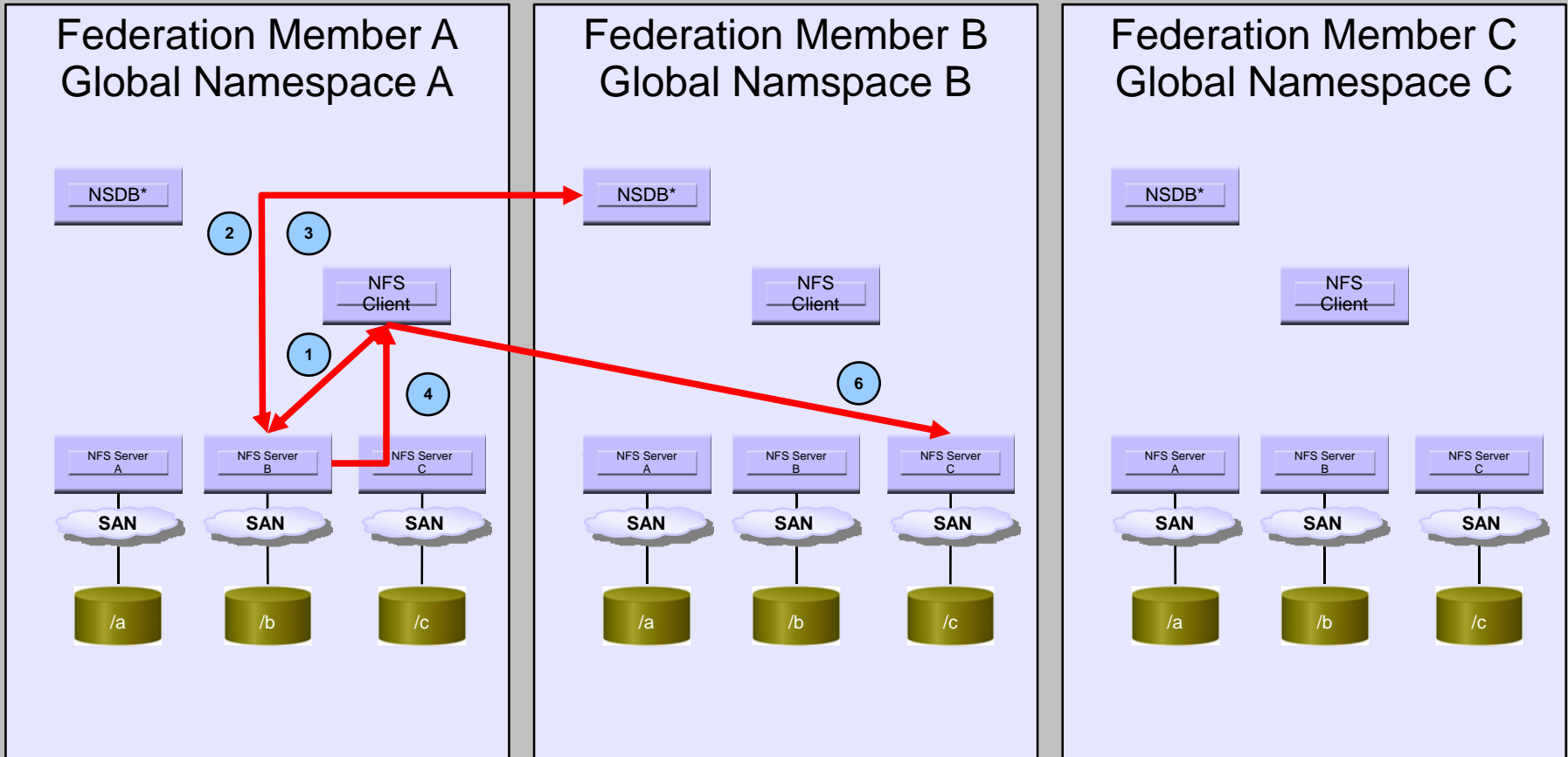
Collection B of File Servers



Collection C of File Servers

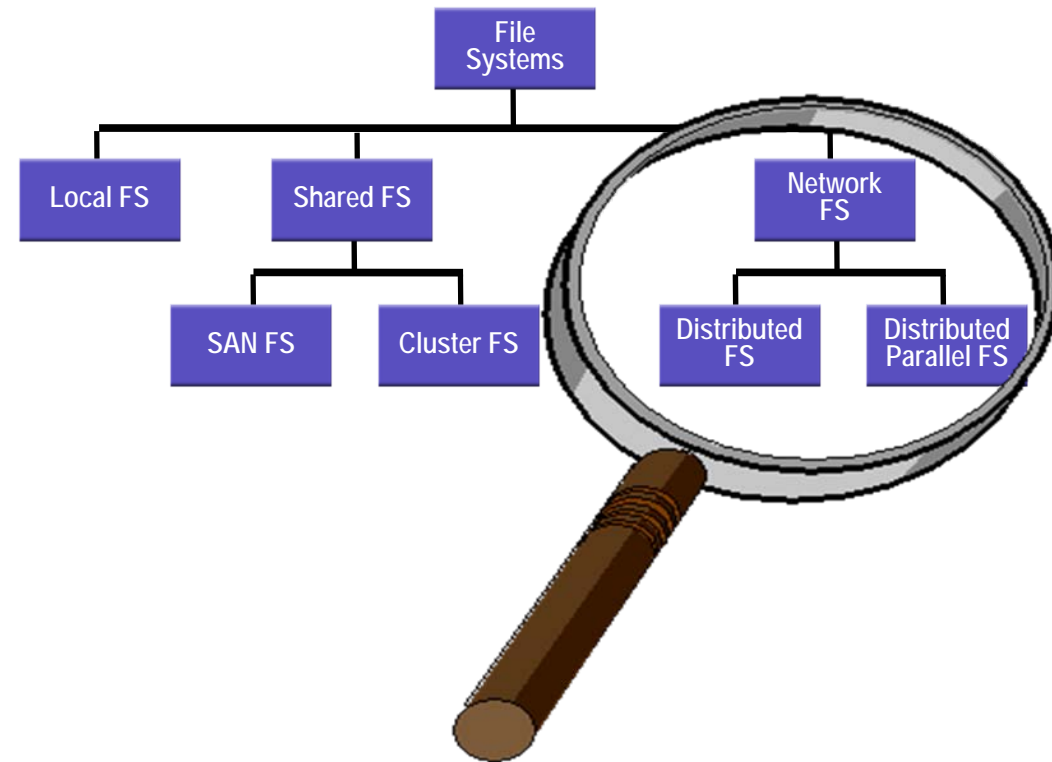
- Current approaches are unsuitable to build a common super namespaces across systems with multiple administrative domains and multiple global namespaces

Federated File System - Global Namespace

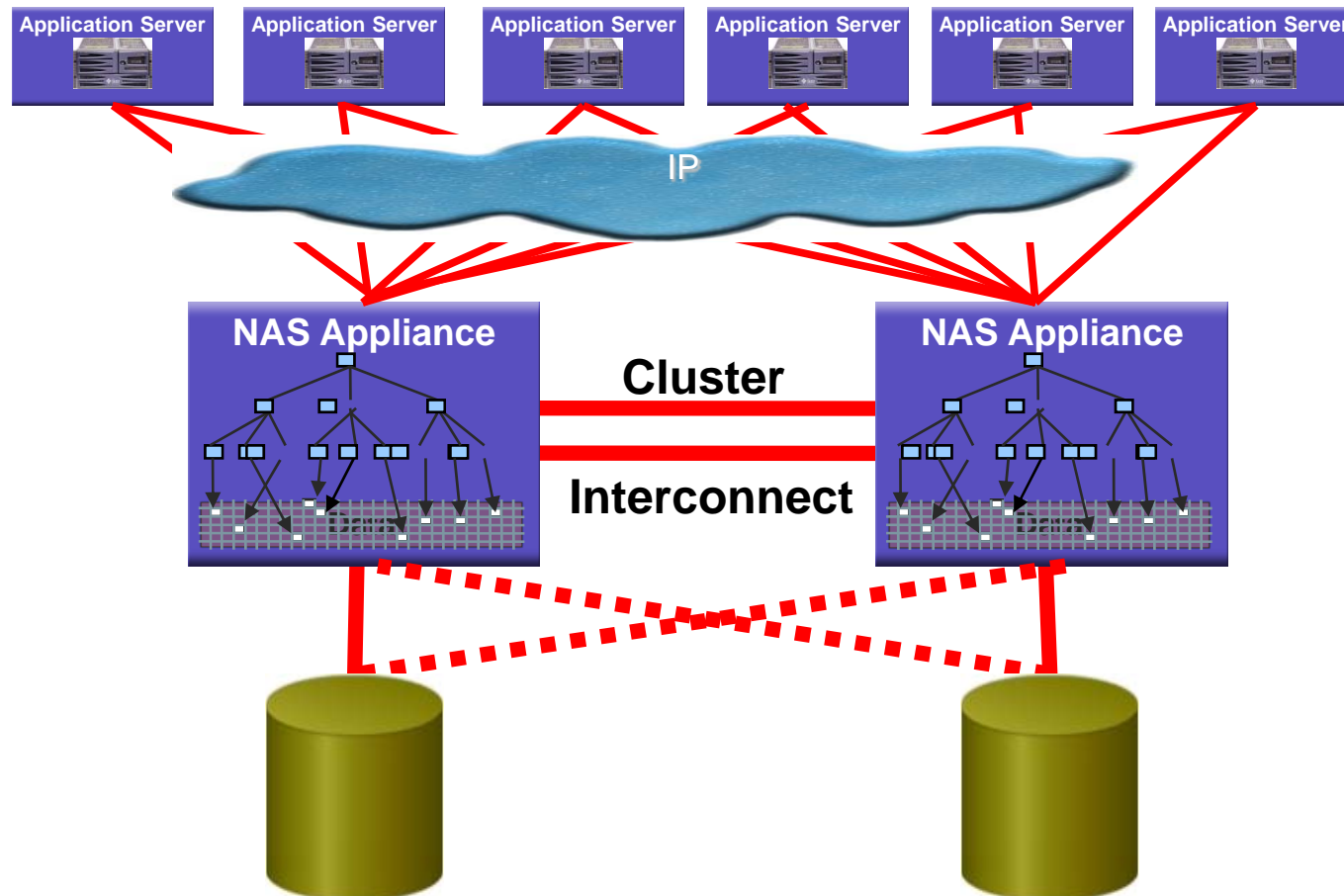


*Namespace Database

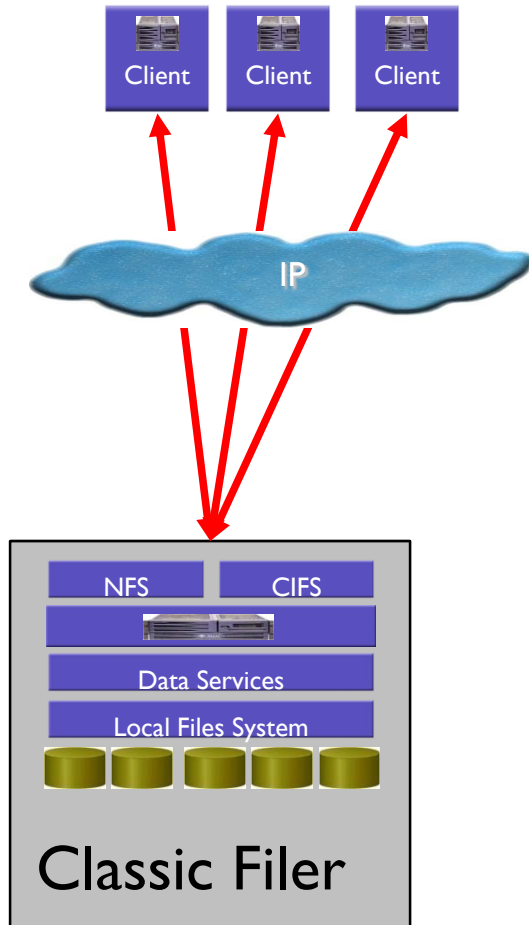
- File System Basics
- File Systems Taxonomy
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- Distributed FS
- Distributed Parallel FS
- **Scale-Out NAS**
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ **NAS Cluster / NAS Grid**
- FS Future Developments



➤ ≠NAS Cluster / NAS Grid

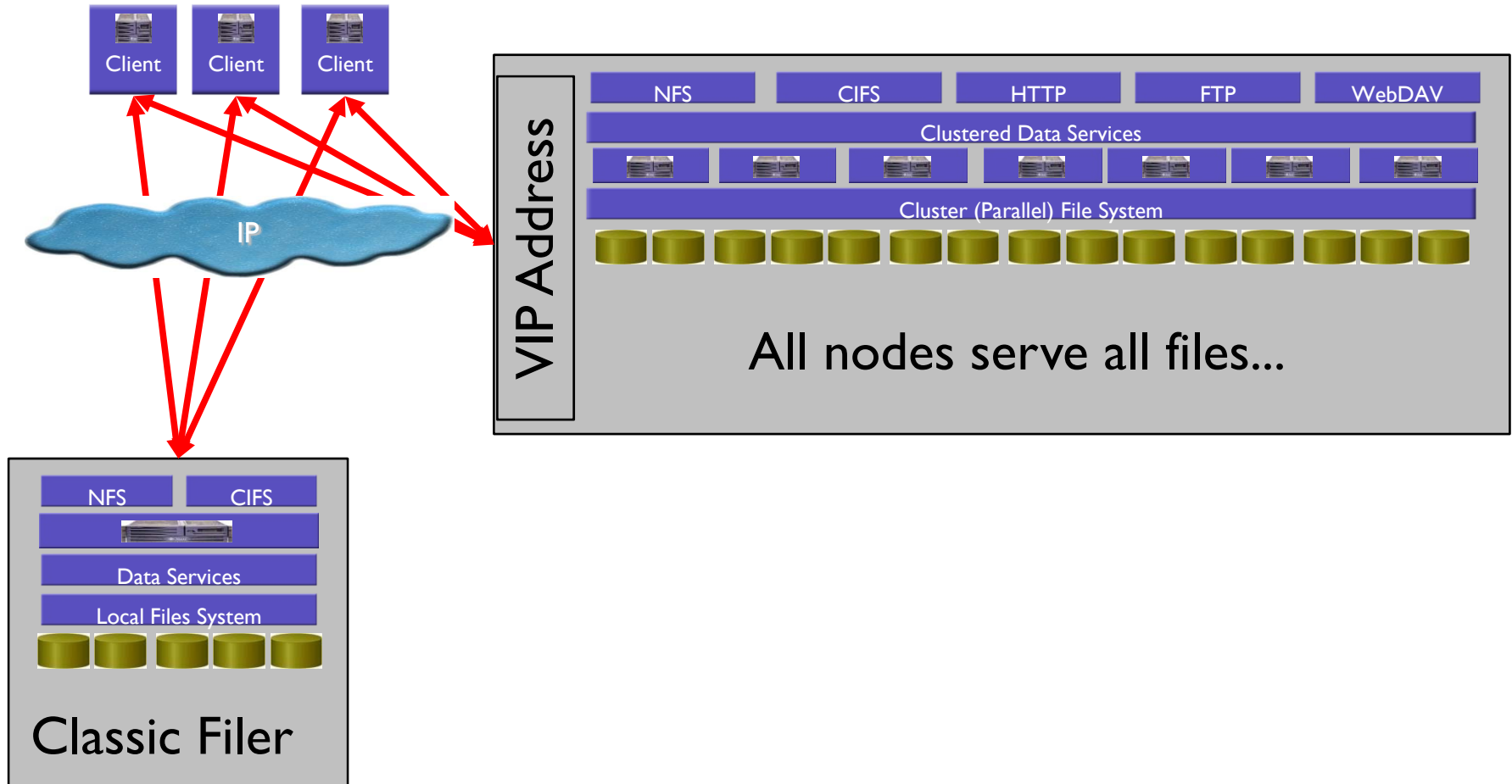


NAS Cluster / NAS Grid (1)



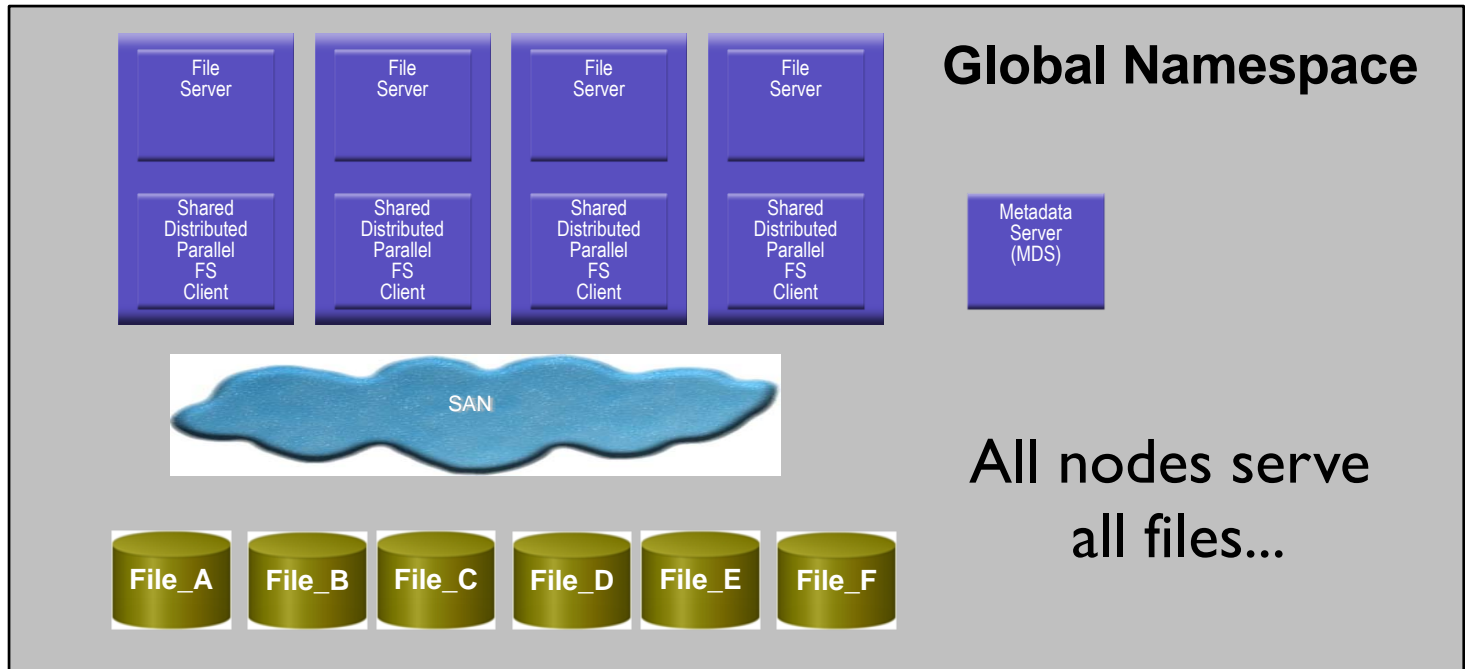
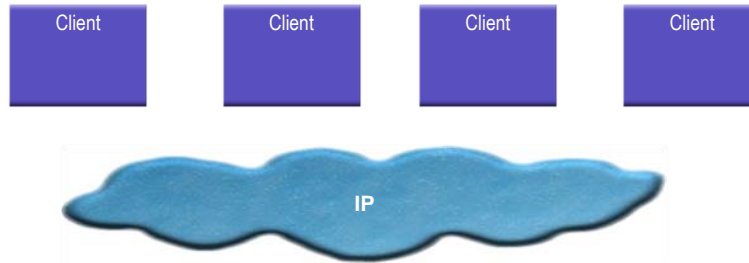
NAS Cluster / NAS Grid (2)

Two flavors:



NAS Cluster / NAS Grid (3)

- File servers are clients in a shared, distributed, distributed-parallel FS
- File servers responsible for file lookup and management
- All file servers serve all files
- Global Lock Manager (GLM) required

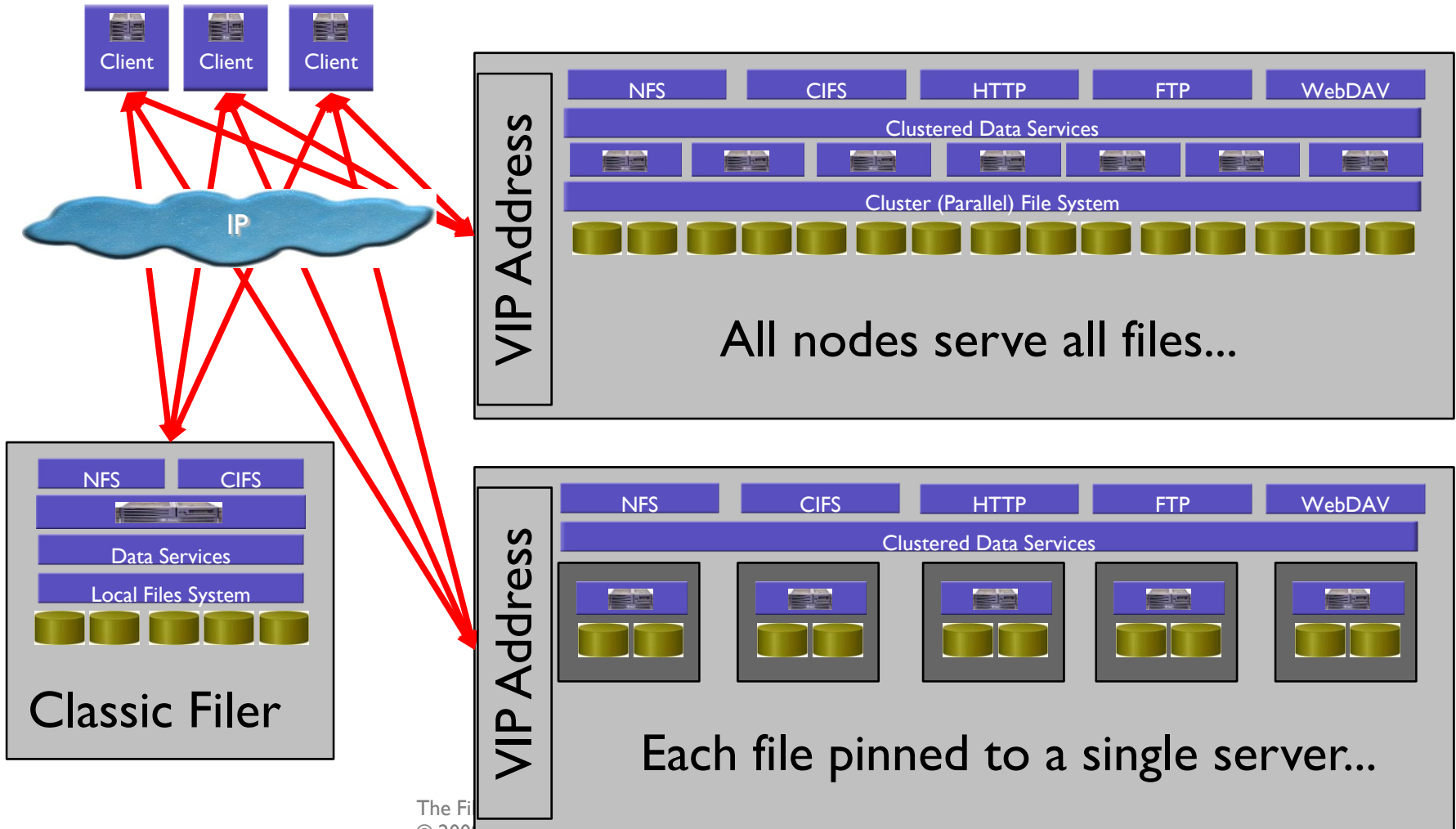


Global Namespace

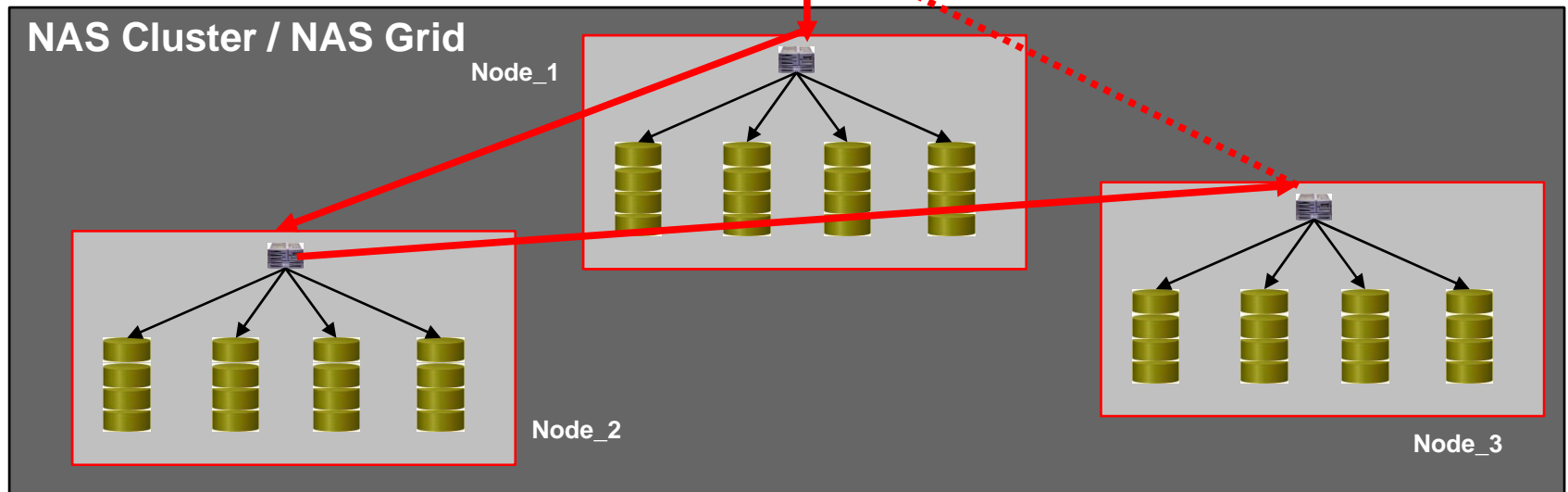
All nodes serve all files...

NAS Cluster / NAS Grid (4)

Two flavors:

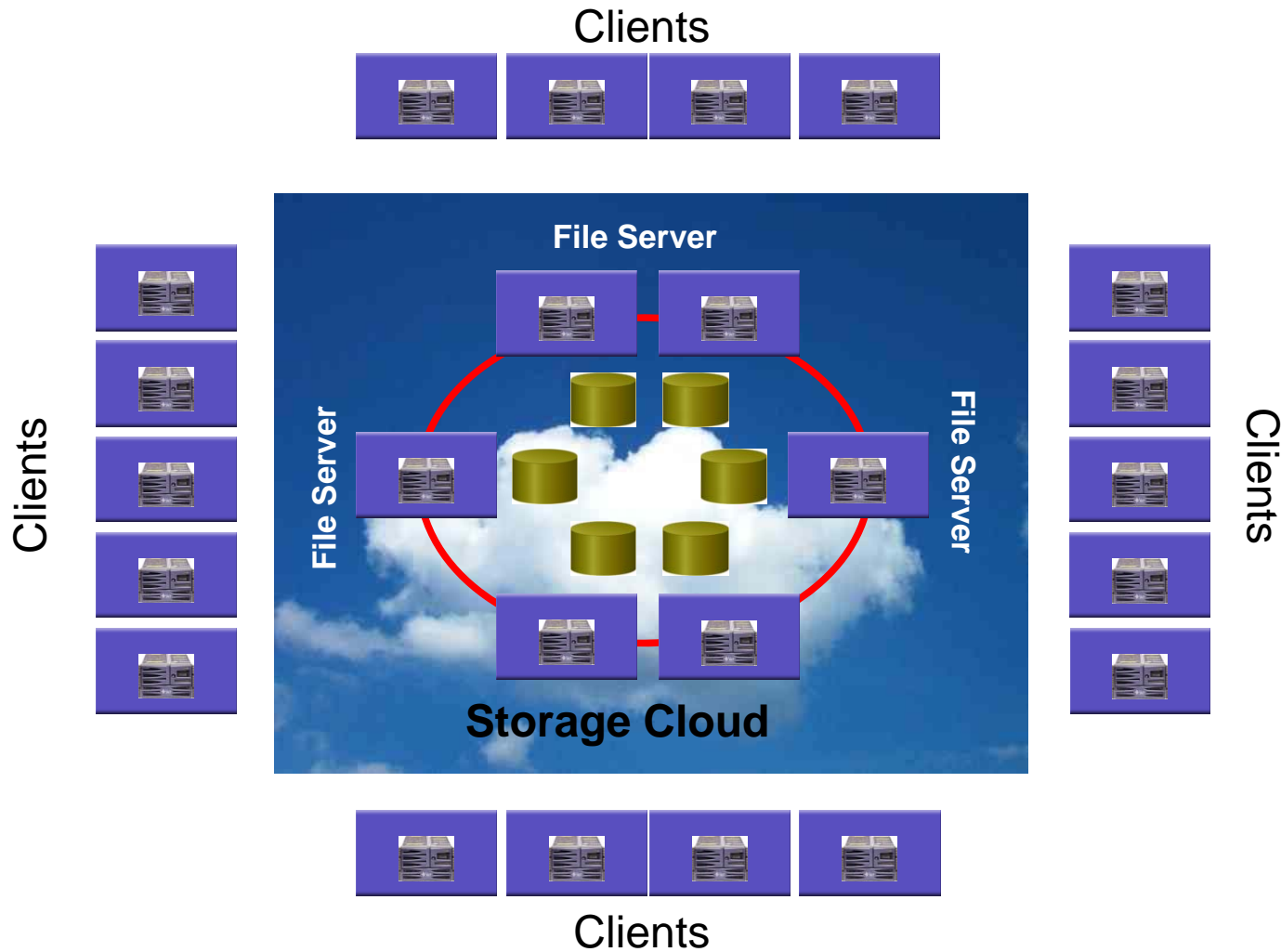


NAS Cluster / NAS Grid (5)

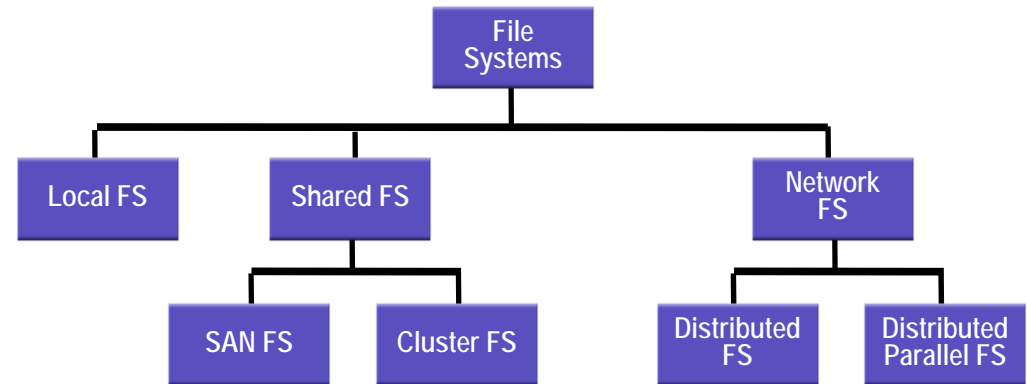


Several NAS systems connected to each other redirecting file I/O requests exporting a single file system

NAS Cluster/Grid & Storage Cloud



- File System Basics
- File Systems Taxonomy
- Local FS
- Shared FS / Global FS
 - ◆ SAN FS, Cluster FS
- Network FS
- Distributed FS
- Distributed Parallel FS
- Scale-Out NAS
 - ◆ NAS Aggregation
 - ◆ NAS Virtualization
 - ◆ NAS Cluster / NAS Grid
- **FS Future Developments**

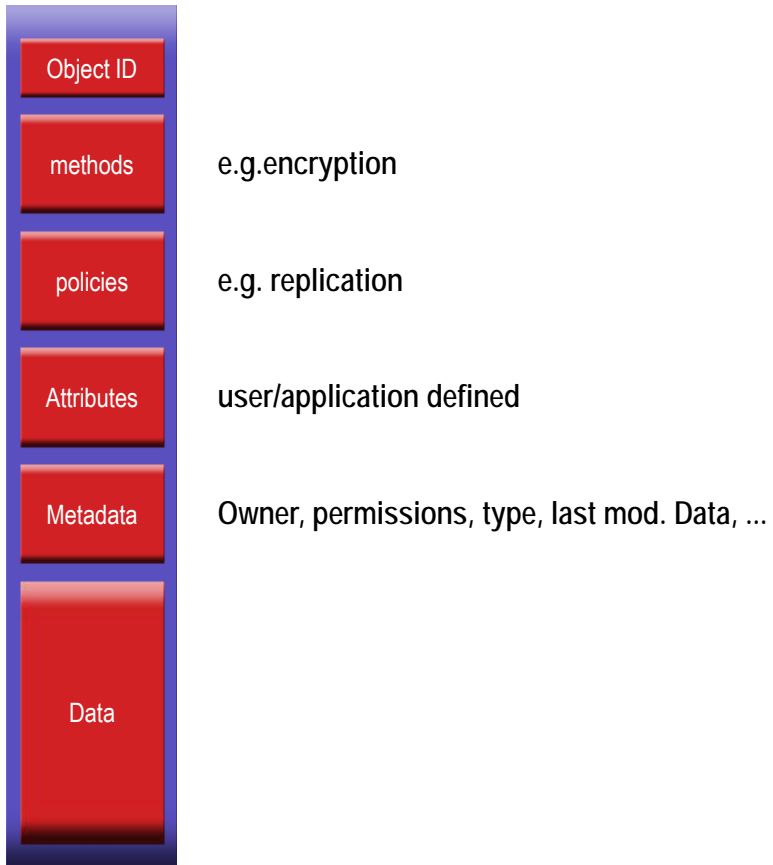


Data For Clouds – File Objects



Owner, permissions, type, last mod. Data, ...

Data For Clouds – File Objects



Data For Clouds – File Objects

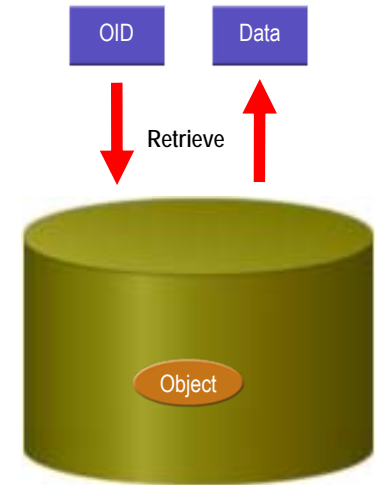
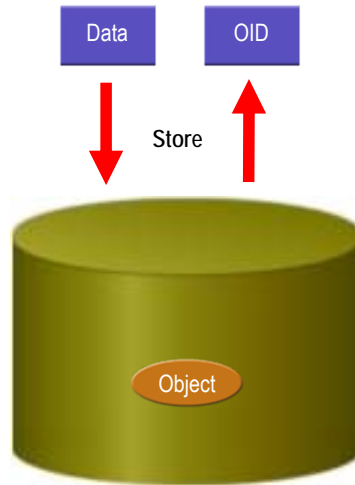


e.g. encryption

e.g. replication

user/application defined

Owner, permissions, type, last mod. Data, ...



Data For Clouds – File Objects

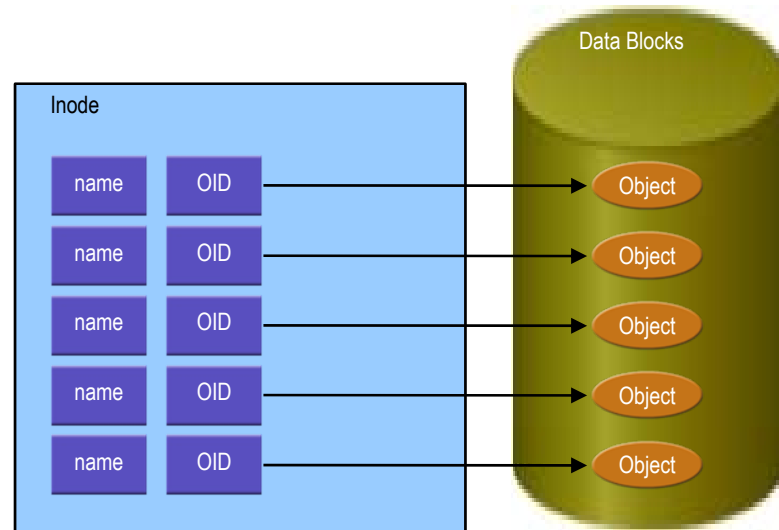
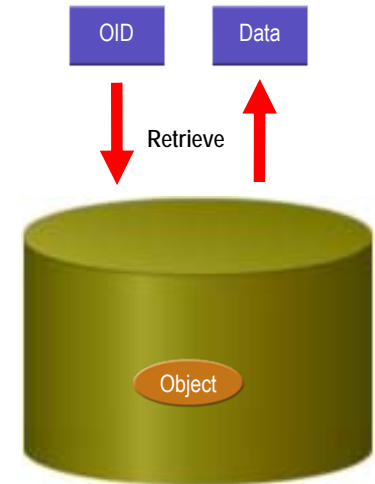
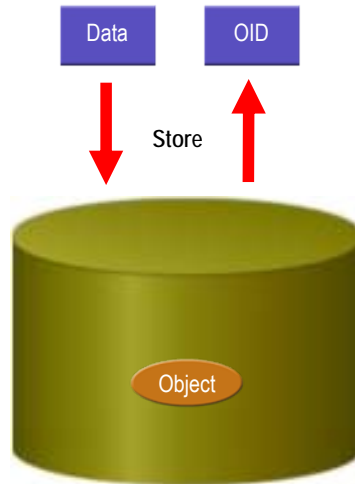


e.g. encryption

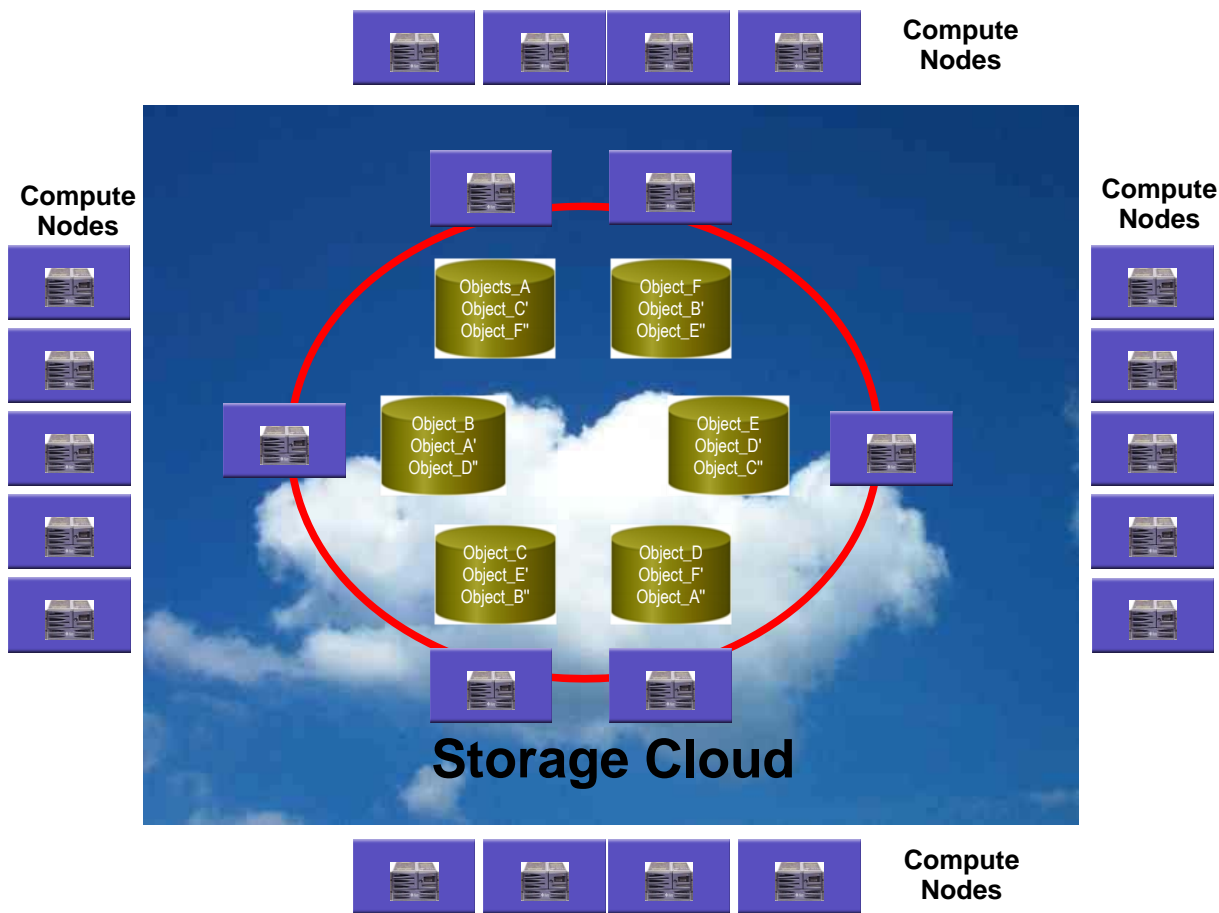
e.g. replication

user/application defined

Owner, permissions, type, last mod. Data, ...



File Systems & Objects for Clouds

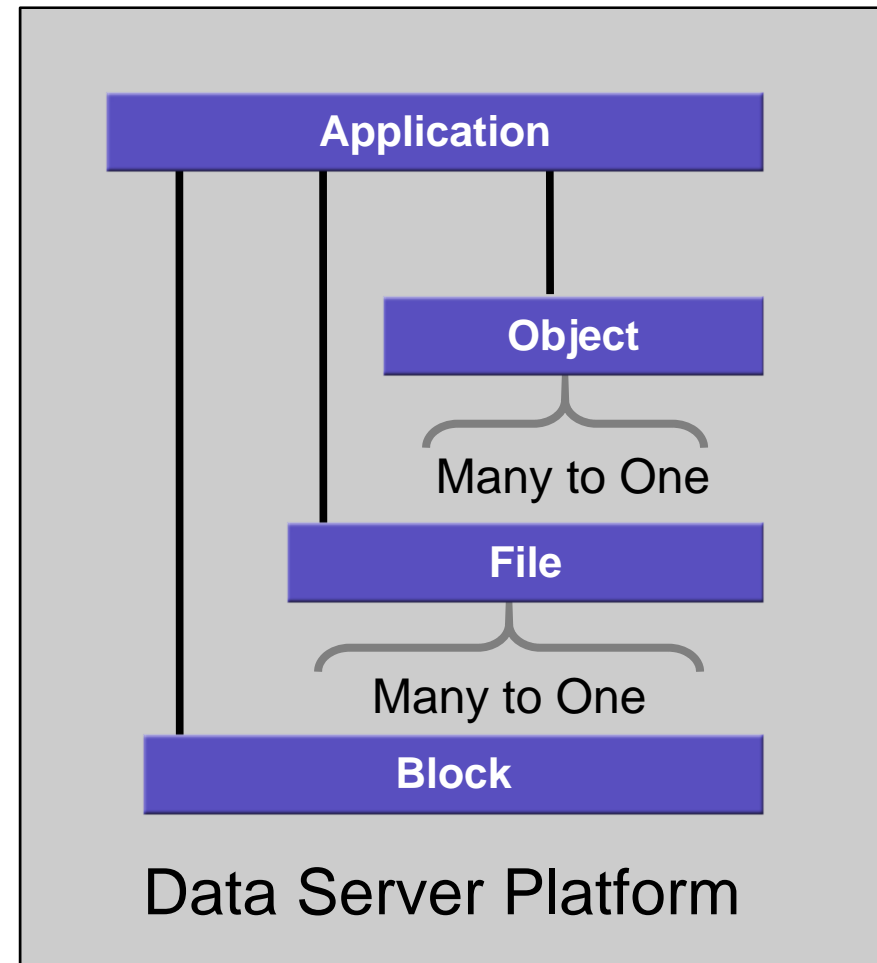


- Availability through file replication
- Sacrifice performance
- Locality of data
- No RAID protection
- Peer-to-peer
- Storage grid
- Mesh topology
- Flat namespace
- Geographically dispersed
- Heterogeneous
- Spontaneous federations

Data Serving Hierarchy

3 Levels of Abstraction

- Application may interface with the storage subsystem in anyone of three layers:
 - ◆ **Block** with highest performance and very little meta data
 - ◆ **File** with medium performance and some meta data
 - ◆ **Object** with medium performance and *rich* meta data



- Please send any questions or comments on this presentation to SNIA: trackfilemgmt@snia.org

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

- SNIA Education Committee

Christian Bandulet, Sun Microsystems