



Education

# Massively Scalable File Storage

Philippe Nicolas, KerStor

- 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.**

- **Title: Massively Scalable File Storage**
- **Abstract:** With the fast evolution of the Internet, the explosion of data volume and growing online services, massively scalable file storage is a reality today. We all use P2P applications which finally use workstation private storage space to enable an almost infinite space. Compliance and Archiving put also a high pressure on file storage and many solutions use innovative approaches developed recently, some research work was done around RAIN architecture. High Performance applications must run and use high performance file storage infrastructure and many proprietary developments were made to address these needs. Some standard work was also done by many vendor to propose some new parallel mode such as pNFS. This session covers only the high-end part of network file storage with high performance, reliability and huge capacity and presents P2P, CAS, Distributed, Parallel and Cloud file storage with important number of servers and clients such as Google FS, Apache project named Hadoop, PVFS, Lustre, pNFS and a few proprietary ones to illustrate industry answers.
- **Learning objectives:** This presentation will introduce and review all high-end network file storage technologies to help better understand vendors solutions and standards challenges. It will help understand each design, advantages and drawbacks to align with applications needs and finally participate to the education of the market to simplify technology adoption.
- **Audience:** IT & Storage Architect, IT Manager and Buyers.

- Needs for Massive Scalability
  - ◆ Fundamental drivers
  - ◆ Needs and limitations
  - ◆ Multiple usages
- Implementations and Philosophies
  - ◆ Conventions & Approaches
  - ◆ Industry and offering examples
- Conclusion

# Needs for Massive Scalability

Fundamental drivers, Needs and limitations

Multiple usages

# Fundamental drivers

- **Avalanche of data, increase demand for unstructured data storage**
  - ◆ x6 in 4 years (281 EB in 2007 to 1800 EB in 2011 – IDC March 2008)
  - ◆ 80% of this volume is unstructured information
  - ◆ 80% of file data growth every year, 80% of inactive data after 30 days ! (file size grows as well)
- **Too many discrete data servers silos and point products**
  - ◆ File Server/NAS, Backup & Archiving, Application, Virtual Machines...
  - ◆ Too low Storage utilization, cost of infrastructure management
  - ◆ Limitations to support millions/100s of millions/billions of files, 10s-100s of PBs, thousands of clients and storage servers (IOPS & BW)
  - ◆ Poor automation
- **Protection too complex and too long**
  - ◆ RTO and RPO not aligned to business
  - ◆ Too many tape restore errors (30%)
  - ◆ Policies enforcement, Load balancing, Migration, Replication, Mirroring, ILM/FLM/Tiering...
- **Business Continuity (HA/BC/DR)**
  - ◆ 100% ubiquitous data access
  - ◆ SLA not set, controlled and measured

# File Storage Needs

## ➤ Needs by numbers and features

- ◆ Clients connections: 10000s
- ◆ Storage Servers: 1000s
- ◆ Storage capacity: 10-100-1000 PBs
- ◆ Throughput: 10s-100sGB/s of aggregated bandwidth & millions of IOPS
- ◆ Full embedded resiliency and redundancy
- ◆ Global, permanent and ubiquity access and presence
- ◆ Notion of global/shared namespace
- ◆ Standard or proprietary file access
- ◆ Security and privacy

**Always-on, Everywhere for Everyone**

# File Storage limitations

Technology	Bottleneck
<p><b>Distributed File System</b>                      e.g NAS/File Servers with NFS/CIFS</p>	<p>Server (Meta-Data and Data)</p>
<p><b>Cluster File System</b></p>	<p>Symmetric: DLM/GLM*                      Asymmetric: Master node                      Dozens of nodes</p>
<p><b>SAN File System</b>                      (aka SAN File Sharing System)</p>	<p>Meta-Data Server                      Hundreds of nodes</p>

\*DLM/GLM: Distributed/Global Lock Manager

# Multiple usages

- Common and new file storage usages
  - ◆ File services
  - ◆ “DB on NAS”
  - ◆ Backup
  - ◆ Archiving & Compliance
  - ◆ Data Mining (file warehouse)
  - ◆ Cloud integration for above topics

**Towards an Universal, Unified, Ubiquitous  
File Storage**

# Implementations and Philosophies

Conventions & Approaches  
Industry and offering examples

# Conventions

## ➤ In this tutorial

- ◆ No coverage of local file system
- ◆ No commercial product mentioned but family, approach or model explained
- ◆ Only Distributed File Storage

## ➤ Scalability, a 3 dimensions approach

- ◆ Performance: no degradation, aggregated improvement
- ◆ Availability: data and access redundancy
- ◆ Manageability: no compromise and no complexity

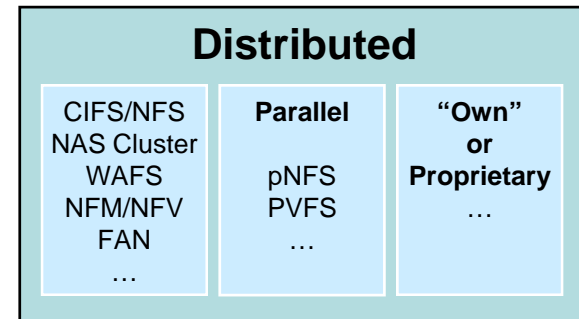
# Implementations and Philosophies

- Industry standard components & servers (COTS\*)
- Open Software (often open-source)
- Standards file sharing protocols (+ API/proprietary)
- Online scalability
- Self Managing, Self-Healing, Self-Aware, Self-Organizing
- Data Management Services
- Scale-out approach with a distributed philosophy based on RAIN, Grid...

\*COTS: Commodity Off The Shelf

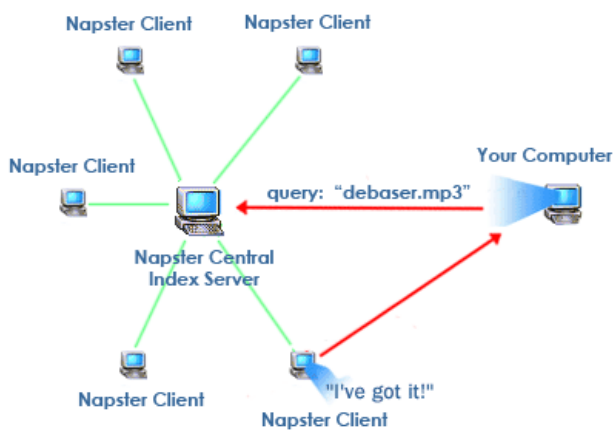
# Distributed implementations

- “Aggregation | Union” of storage servers
- Symmetric, Asymmetric or P2P
  - ◆ Central Authority aka Master node or Masterless
- Parallel vs. non-Parallel
  - ◆ File striped and concurrent access, single node access
- Shared-nothing model
  - ◆ Aggregation of file storage/servers
- User-mode vs. Kernel/System-mode
- File-based vs. object-based
- Notion of Shared/Private Namespace (storage nodes wide)
- RAIN, Grid implementation with embedded data protection and resiliency
  - ◆ No storage nodes are fully trusted and are expected to fail at any time
- Extended features
  - ◆ Policies enforcement for Automation, Snapshot, CDP, Replication, Mirroring, ILM/FLM/Tiering, Migration, Load balancing...

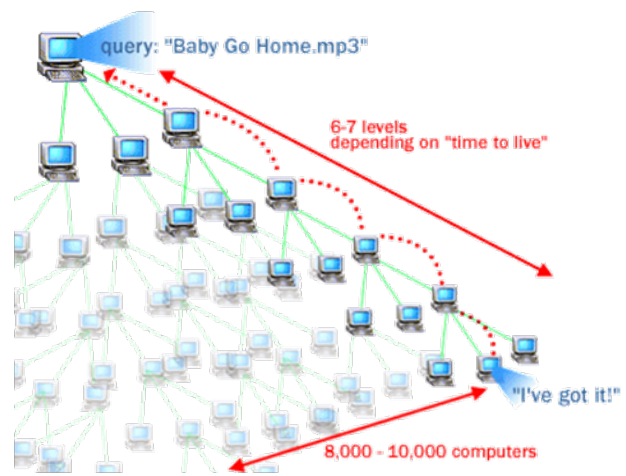


# P2P Implementations

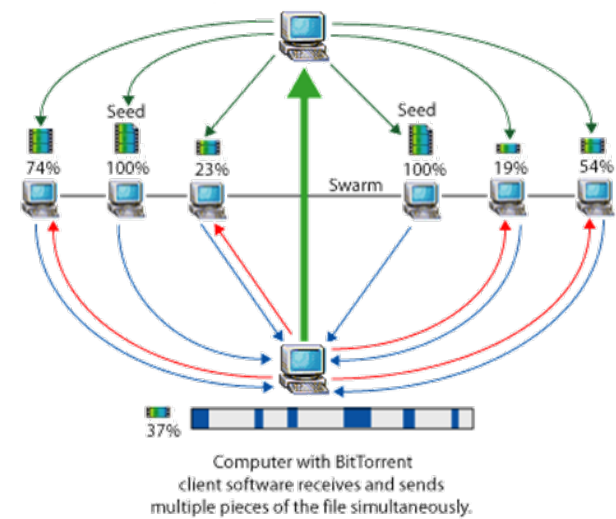
- Interesting implementation with aggregation of other machines storage space with or wo a central server (asym or sym P2P)
  - ◆ Ex: P2P File Sharing System such as music, mp3...



Napster



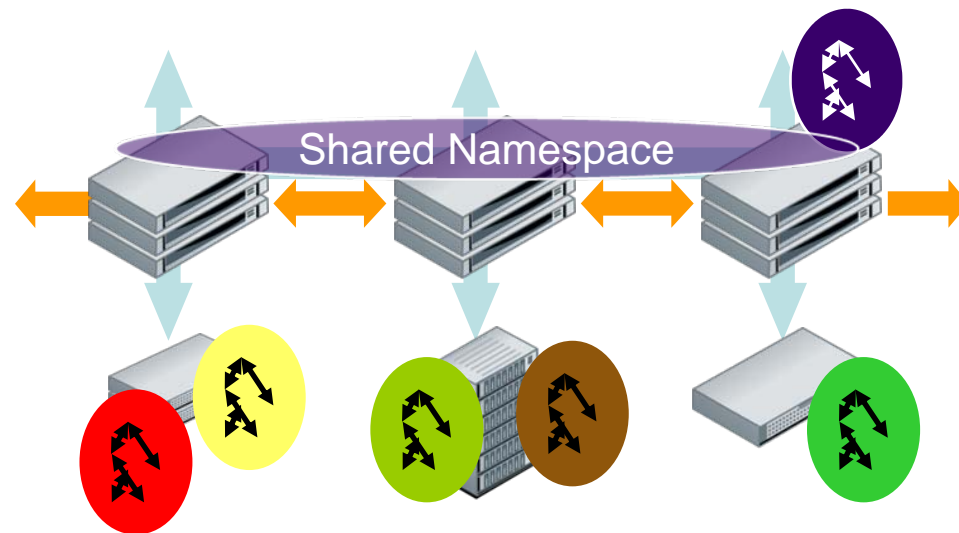
Gnutella



BitTorrent

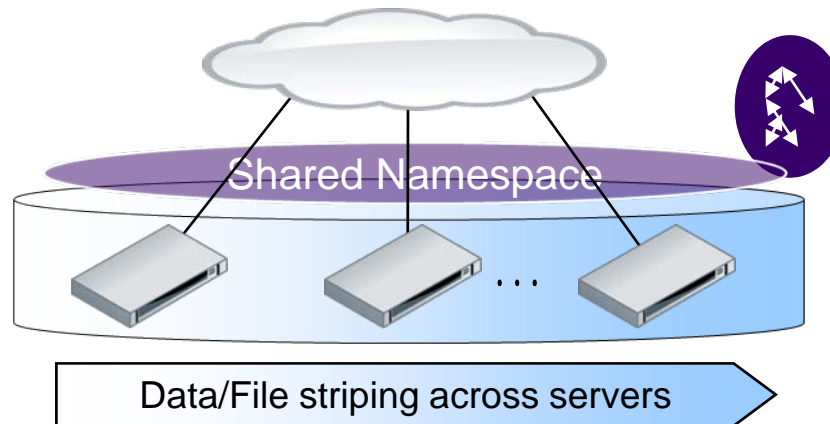
# Industry implementations

- Aggregation of independent storage servers
  - ◆ File entirely stored by 1 storage server (no file striping)
- Symmetric Philosophy



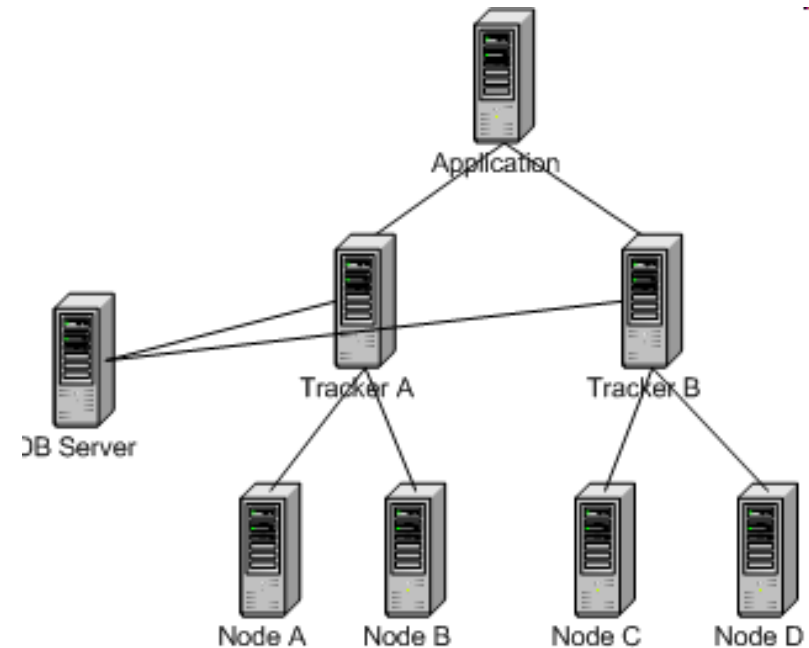
# Industry implementations

- Aggregation of homogeneous storage servers
  - ◆ File is striped across storage server
- Symmetric Philosophy



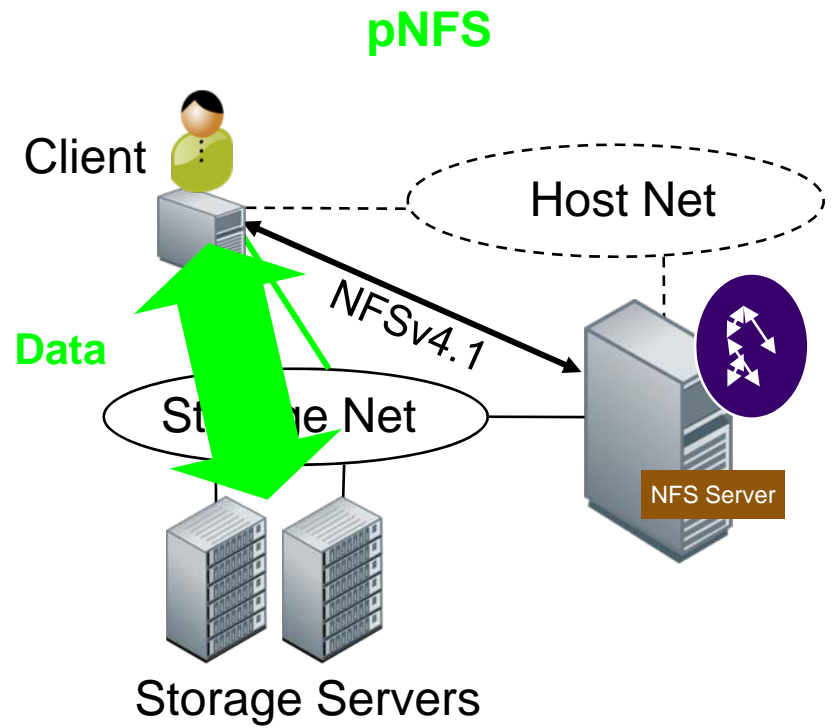
# MogileFS

- Application-level Distributed FS (no kernel module)
- Open-source
- Fully redundant, automatic file replication
- Flat Namespace
- Shared-Nothing
- Local filesystem agnostic
- But not POSIX compliant
- [www.danga.com/mogilefs](http://www.danga.com/mogilefs)



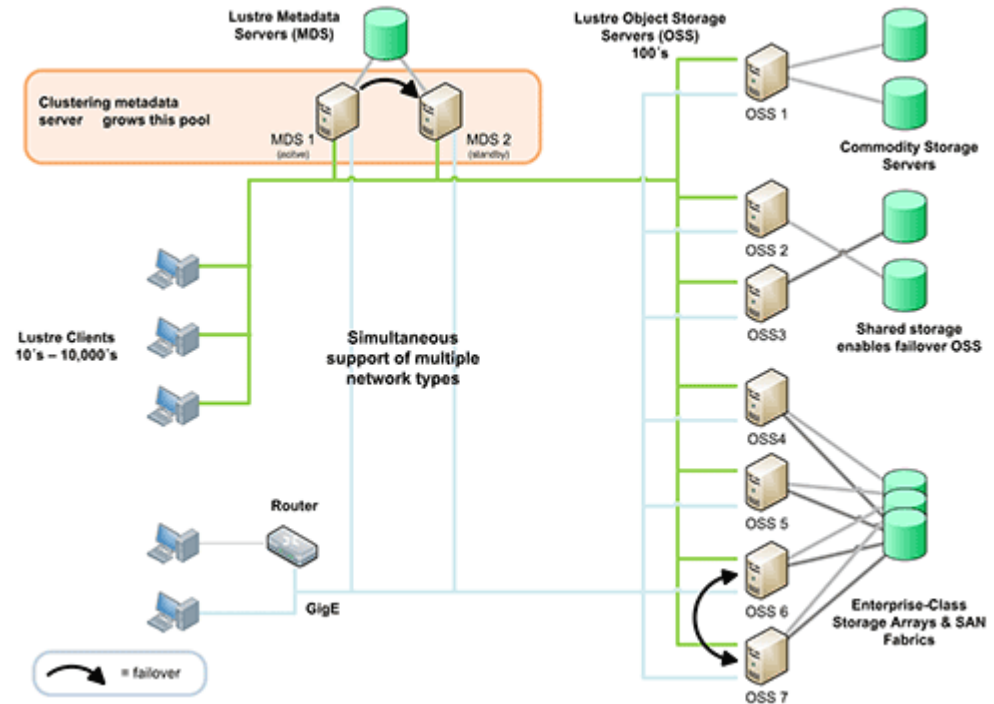
# pNFS (Parallel NFS with NFS v4.1)

- pNFS is about scaling NFS and address file server bottleneck
  - ◆ Same philosophy as SAN FS (master/slave asymmetric philosophy) and data access in parallel
- Allow NFSv4.1 client to bypass NFS server
  - ◆ No application changes, similar management model
- pNFS extensions to NFSv4 communicate data location to clients
  - ◆ Clients access data via Fibre Channel & iSCSI (block), OSD (object) or NFS (file)
- [www.pnfs.com](http://www.pnfs.com)



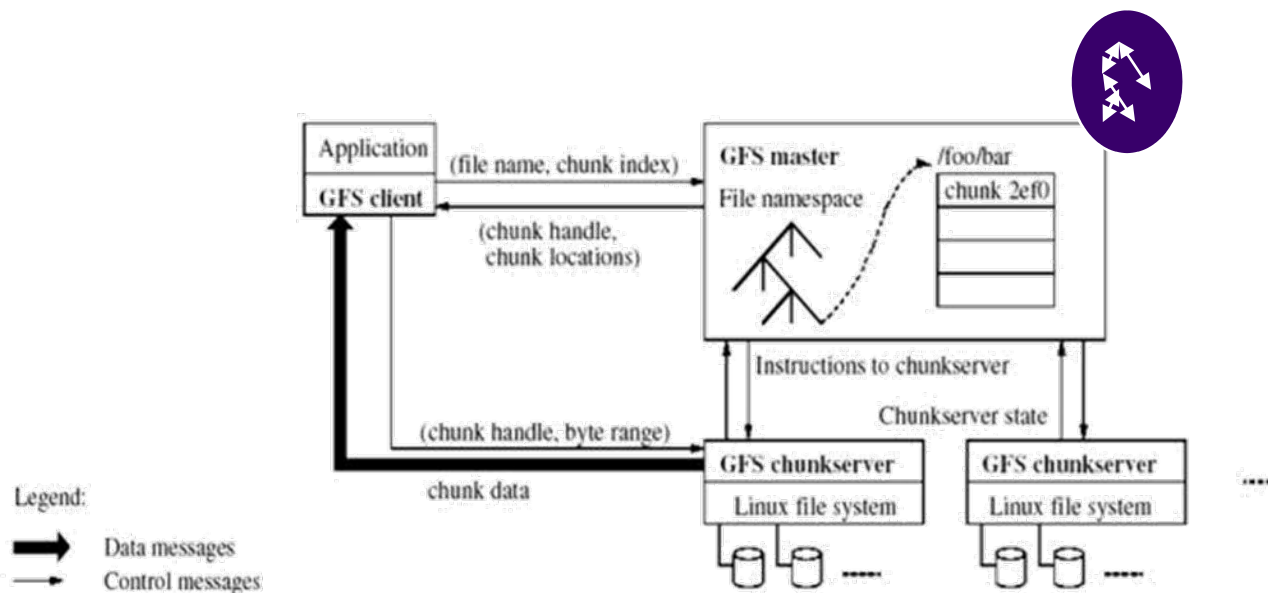
# Lustre

- Open source object-based storage system
  - ◆ Based on NASD study from Carnegie Mellon Univ.
- University project
- Asymmetric Philosophy
- Notion of Object (OST/OSS)
- [www.lustre.org](http://www.lustre.org)



# Google File System

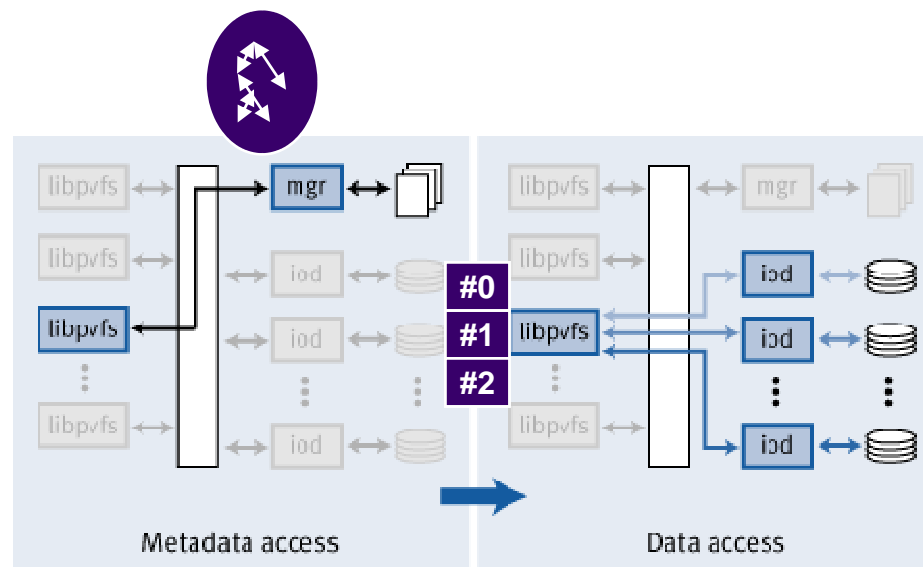
- Internal deployment but used by WW end-users
- Proprietary approach
- Asymmetric philosophy
- Thousands of chunk servers with 64MB chunk size (stripe unite)
- [research.google.com/pubs/papers.html](http://research.google.com/pubs/papers.html)



Source Google

# PVFS (Parallel Virtual File System)

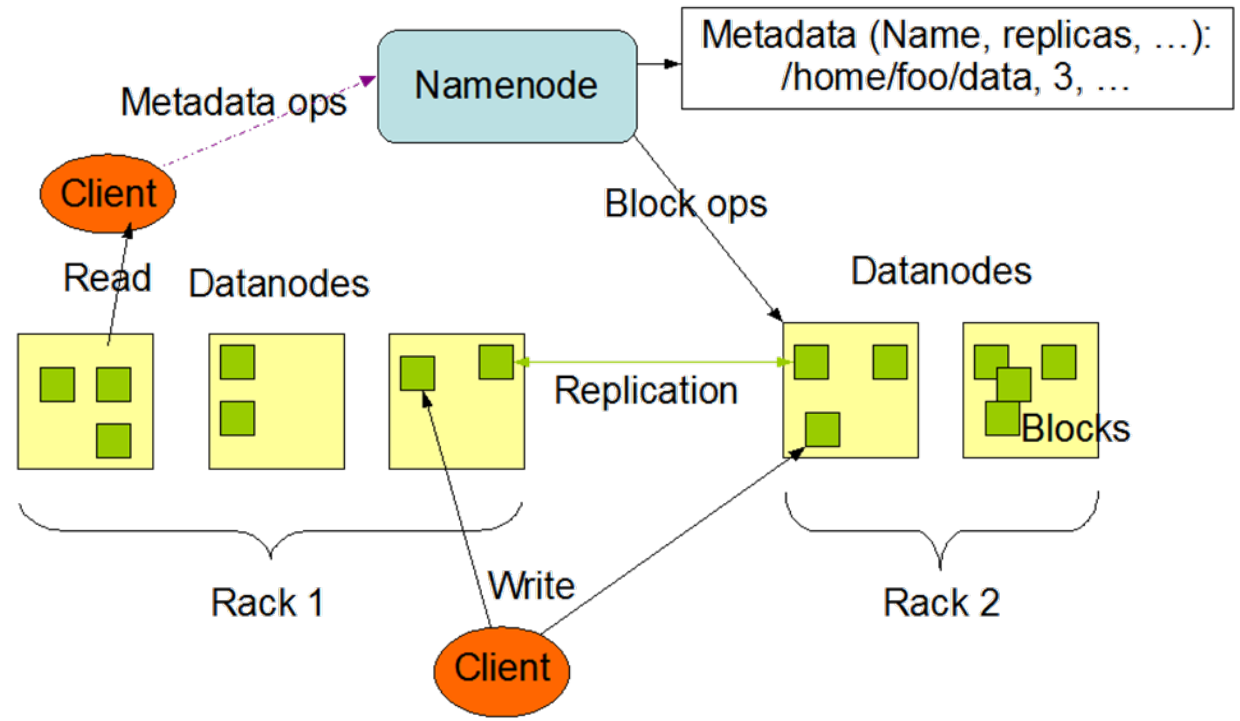
- Now PVFS2
- Project from Clemson University and Argonne National Lab
- Open source and based on Linux
- Asymmetric philosophy
- [www.pvfs.org](http://www.pvfs.org)



# Hadoop DFS

- Apache project
- Highly fault-tolerant built in Java
- Large data sets
- Asymmetric philosophy
- Files striped across DataNodes
- [hadoop.apache.org](http://hadoop.apache.org)

HDFS Architecture



# Conclusion

## ➤ New IT/Technology and Research projects need Scalable (File) Storage

- ◆ Cost effective approach with COTS
- ◆ Acceptance of thousands of machines (clients, servers)
- ◆ Asymmetric (Master/Slave) seems to be a more scalable philosophy
- ◆ Unified, Global Namespace is fundamental
- ◆ Superiority of File/object on block approach
- ◆ Reliability & Security are key (authorization, authentication, privacy...)
- ◆ Standard is needed for broad adoption and cost reduction (pNFS)





Check out SNIA Tutorial:  
**The File Systems Evolution**



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



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



Check out SNIA Tutorial:  
**Virtualization I - What, Why, Where and How?**



Check out SNIA Tutorial:  
**Massively Scalable File Storage**



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



Check out SNIA Tutorial:  
**Cloud Storage: The New Paradigm for Accessing Storage as a Service**

- Please send any questions or comments on this presentation to SNIA
  - ◆ [trackfilemgmt@snia.org](mailto:trackfilemgmt@snia.org) (File Systems & File Management)

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

**- SNIA Education Committee**

**Philippe Nicolas**



Education

# Massively Scalable File Storage

Philippe Nicolas, KerStor