Education

pNFS, parallel storage for grid and enterprise computing

Joshua Konkle, NetApp, Inc.

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.

Abstract

- pNFS, parallel storage for grid and enterprise computing
  - This session will appeal to Data Center Managers, Cluster/Grid/Compute Server administrators, and those that are seeking a fundamental understanding pNFS. This session will cover the four key reasons to start working with NFSv4 today. Explain why pNFS is a great parallel storage option for grids and enterprise computing, unlocking hidden potential in clustered storage. We'll conclude the session with an overview of the storage layouts for parallel NFS; NFSv4.1 Files, Blocks and T10 OSD Objects.

Tutorial Agenda

- Introduction to NFS
- NFS v4 – Security, High Availability, Internationalization and Performance (SHIP)
- NFSv4.1 – Status and Overview (pNFS)
- pNFS – Layout Overview
  - Files based access
  - Block based access
  - Object based access
- pNFS – OpenSource Client Status

Background Information

- Network File System
  - Protocol to make data stored on file servers available to any computer on a network
  - NFS clients are included in all common Operating Systems, e.g. Linux, Solaris, AIX, Windows etc…..
  - Application and OSI layers (remote procedure calls)
- NFS Server; Inspiration to NAS and appliances
  - Commodity Operating Systems have NFS servers
  - NAS Appliance – Control, Consistency and Cadence
  - Vendors offer commodity hardware, w/ management software

NFSv4 SHIP is sailing

Education SNIA

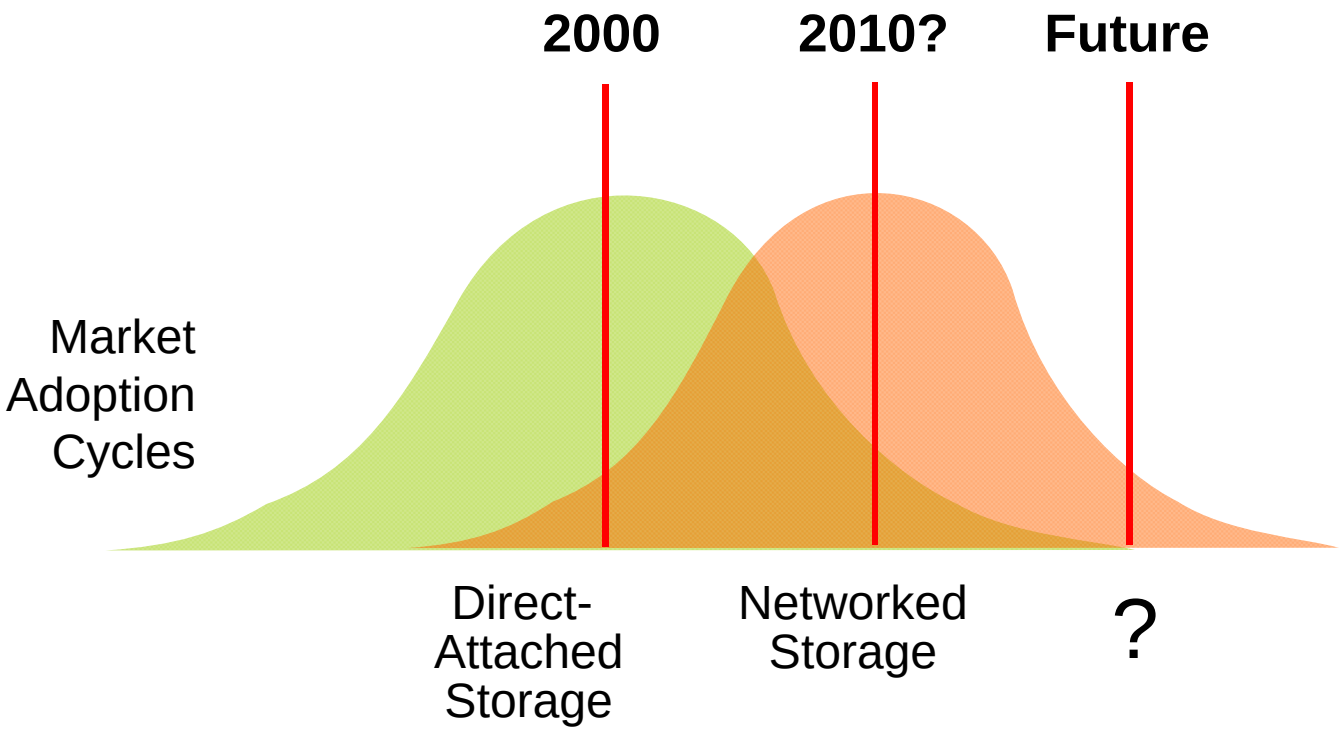
	Functional	Business Benefit
Security	ACLs for authorization Kerberos for authentication	Compliance, improved access, storage efficiency
High availability	Client and server lease management with fail over	High Availability, Operations simplicity, cost containment
International characters	Unicode support for utf8 characters	Global file system for multi-national organizations
Performance	Multiple read, write, delete operations per RPC call	Better network utilization for all NFS clients
	Delegate locks, read and write procedures to clients	Leverage NFS client hardware for better I/O

Lower costs and increase productivity with NFSv4

NFSv4 - HA and Performance

- High Availability via Leased Lock
  - Client renews lease on server file lock @ n Seconds
  - Client fails, lock is not renewed, server releases lock
  - Server fails, on reboot all files locked for n Seconds
    - Gives clients an n Second grace period to reclaim locks
- Performance via Delegations
  - File Delegations allow client workloads for single writer and multiple reader
  - Clients can perform all reads/writes in local client cache
  - Delegations are leased and must be renewed
  - Delegations reduce lease lock renewal traffic

The Evolution of Storage

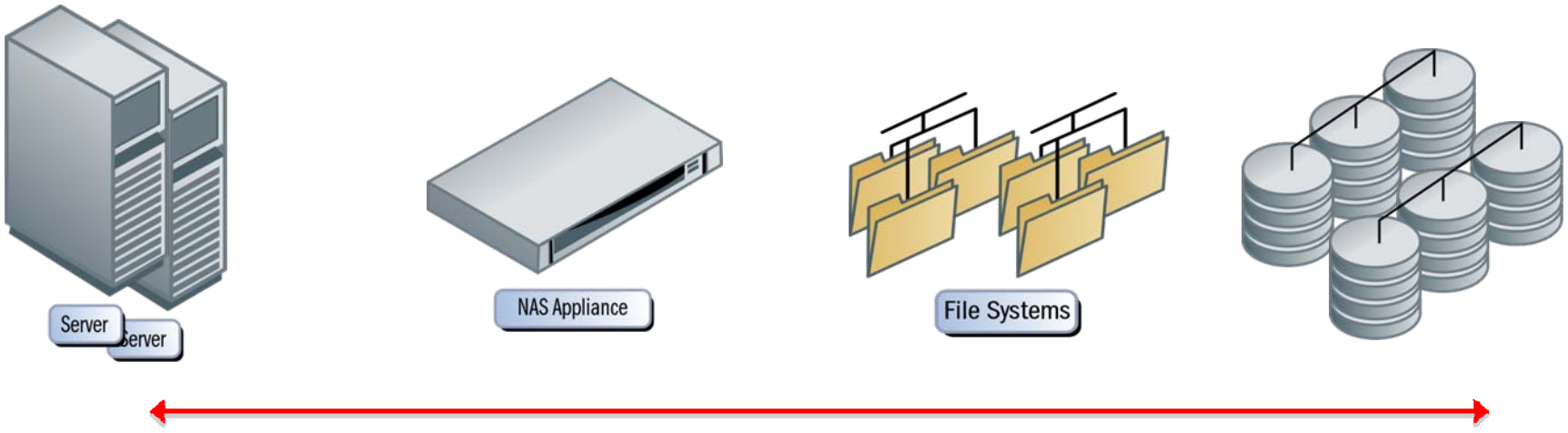


Evolving Requirements

- Economic Trends
  - Cheap and fast computing clusters
  - Cheap and fast network (GigE to 10GigE)
- Performance
  - Exposes single threaded bottlenecks in applications
  - Evolution of computing models
  - Reduced time to market, Response Time
  - Competitive edge or IOPS
- Powerful compute systems
  - more data at exponential rates

NFS – What's the problem?

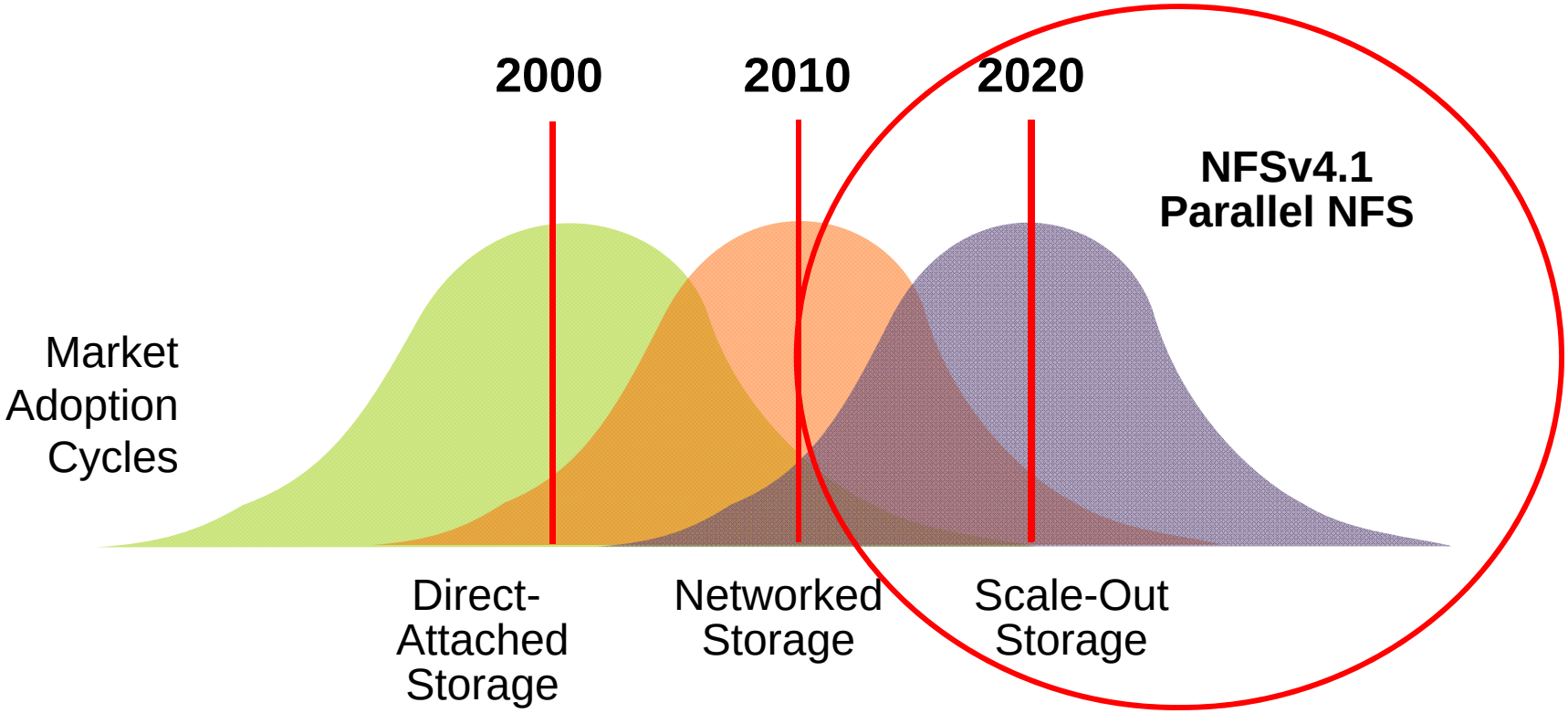
- In-band data access model
  - Easy to build, Limited in scale
  - Well-defined failure modes
  - Limited load balancing options
- Results in Limitations
  - Islands of storage
  - Server and Appliance HW
  - Networking and I/O



Performance, Management and Reliability

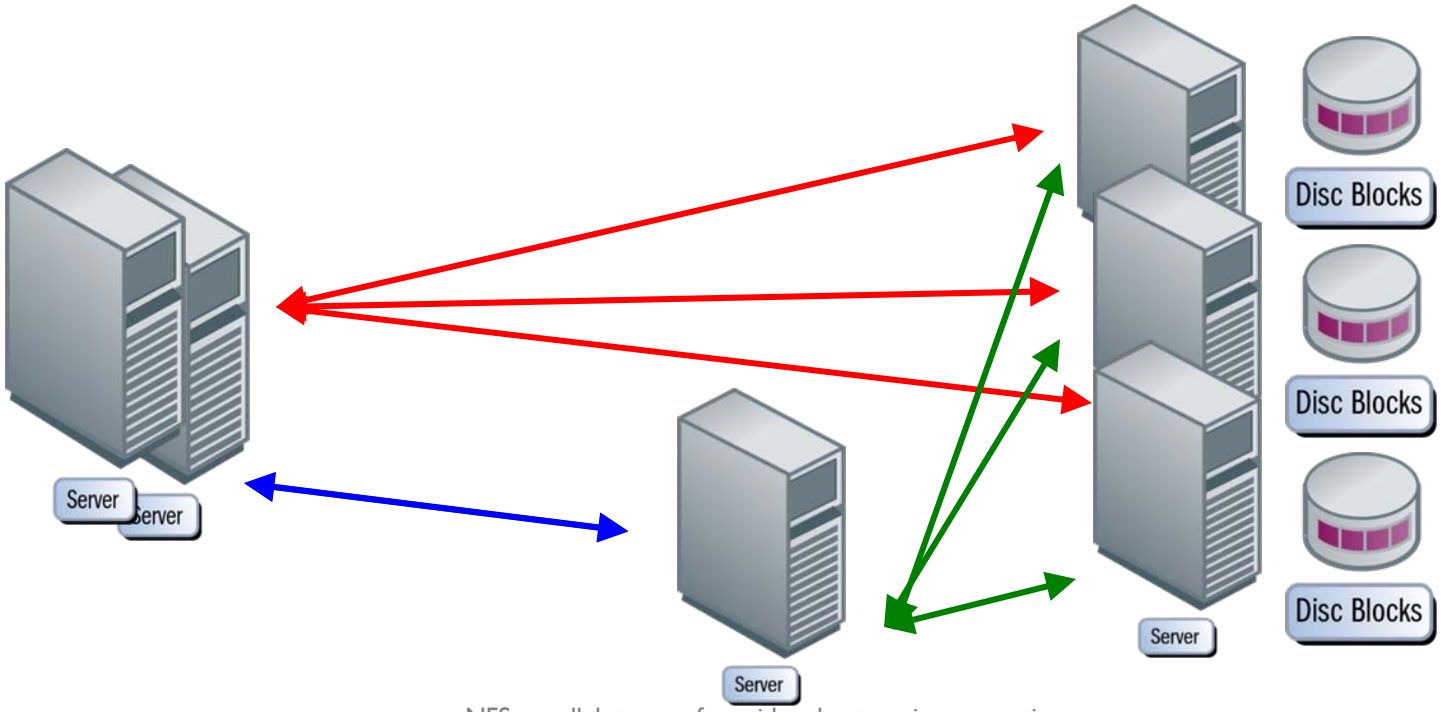
- Random I/O and Metadata intensive workloads
  - Memory and CPU are hot spots
  - Load balancing limited to pair of NFS heads
  - Limited to dual-head configuration
- Compute farms are growing larger in size
  - NFS head can handle a 1000+ NFS clients
  - NFS head hardware comparable to client CPU, I/O, Memory
  - NFS head requires more spindles to distribute the I/O
- Reliability and availability are challenging
  - Data striping limited to single head and disks
  - Non-disruptive upgrades affect dual-head configurations
  - Access and load balancing are typically limited to a pair of NFS server heads

What is the Solution?



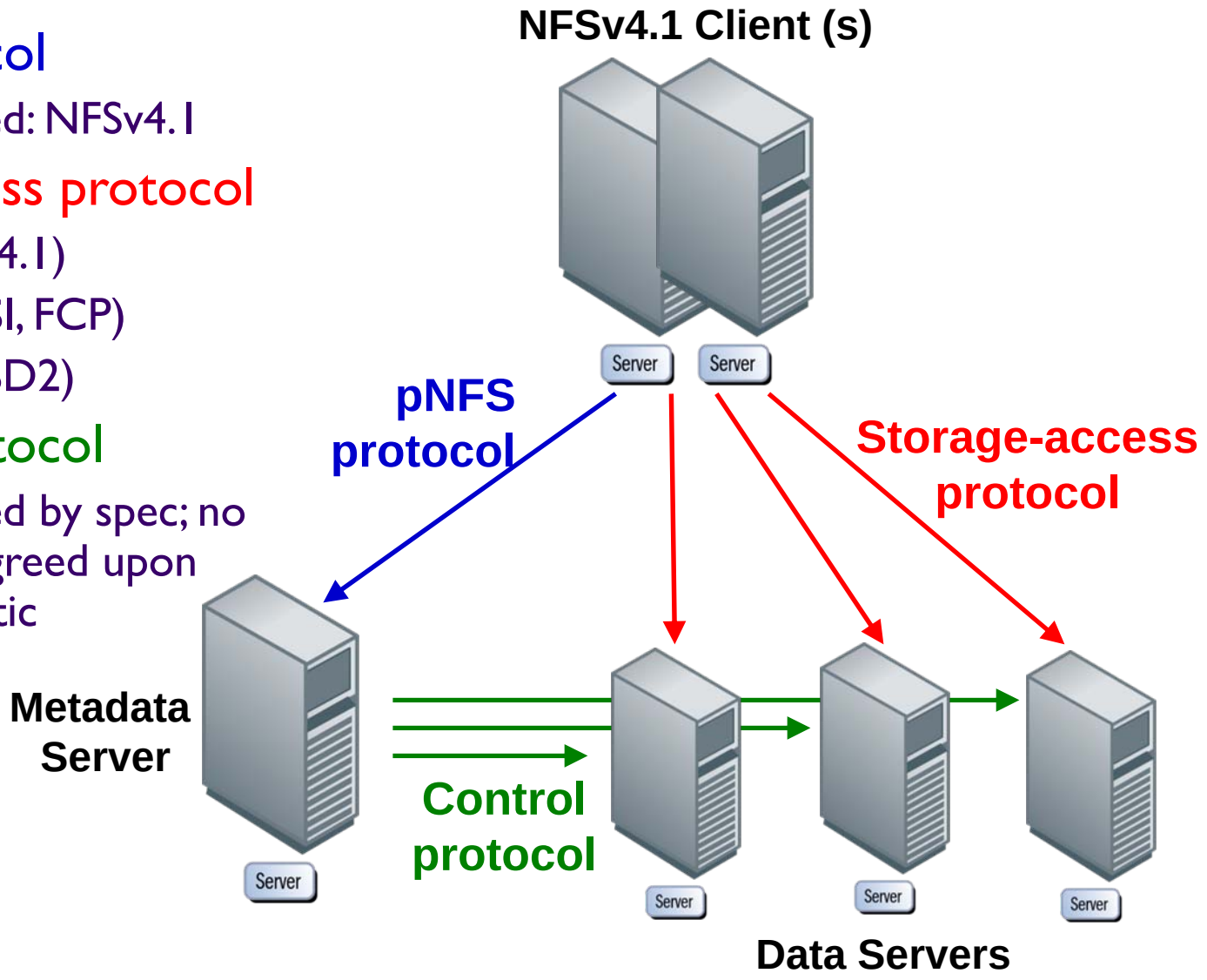
NFSv4.1 – Parallel Data Storage

- NFSv4.1 – Three Storage Types
  - Files – NFSv4.1
  - Blocks – SCSI
  - Objects – OSD T10
- Results in Improvements
  - Global Name Space
  - Head and Storage scaling
  - Non disruptive upgrades while maintaining performance



NFSv4.1 - Parallel NFS 101

- pNFS protocol
  - Standardized: NFSv4.1
- Storage-access protocol
  - Files (NFSv4.1)
  - Block (iSCSI, FCP)
  - Object (OSD2)
- Control protocol
  - Not covered by spec; no generally agreed upon characteristic



NFSv4.1 – Status and Overview

- 2004 – CMU, NetApp and Panasas draft pNFS problem and requirement statements
- 2005 – CITI, EMC, NetApp and Panasas draft pNFS extensions to NFS
- 2005 – NetApp and Sun demonstrate pNFS at Connectathon
- 2005 – pNFS added to NFSv4.1 draft
- 2006 - 2008 – specification baked
  - Bake/Connect a thons; 29 iterations of NFSv4.1/pNFS spec
- 2008 – NFSv4.1/pNFS reaches IETF Approval (December)

pNFS Standards Status

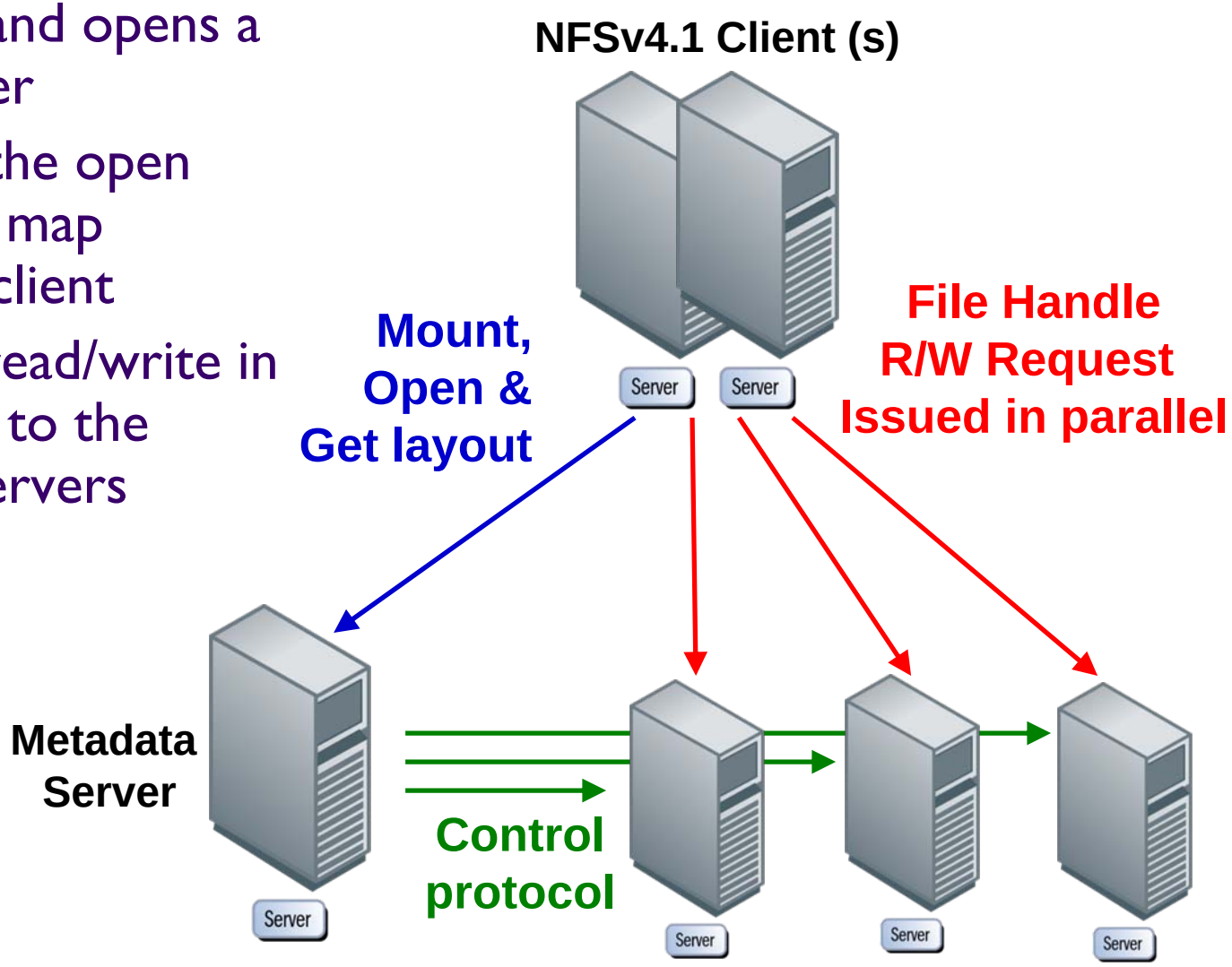
- NFSv4.1/pNFS were standardized at IETF
  - NFSv4 working group (WG)
- All done except for RFCs:
  - WG last call (DONE)
  - Area Director review (DONE)
  - IETF last call (DONE)
  - IESG approval for publication (DONE)
  - IANA review (TBD)
  - RFC publication (Expected 2009)
- Will consist of several documents:
  - NFSv4.1/pNFS/file layout
  - NFSv4.1 protocol description for IDL (rpcgen) compiler
  - blocks layout
  - objects layout
  - netid specification for transport protocol independence (IPv4, IPv6, RDMA)

pNFS Operations

- LAYOUTGET
  - Obtains the data server map from the meta-data server
- LAYOUTCOMMIT
  - Servers commit the layout and update the meta-data maps
- LAYOUTRETURN
  - Returns the layout; Or the new layout, if the data is modified
- GETDEVICEINFO
  - Client gets updated information on a data server in the storage cluster
- GETDEVICELIST
  - Clients requests the list of all data servers participating in the storage cluster
- CB_LAYOUT
  - Server recalls the data layout from a client; if conflicts are detected

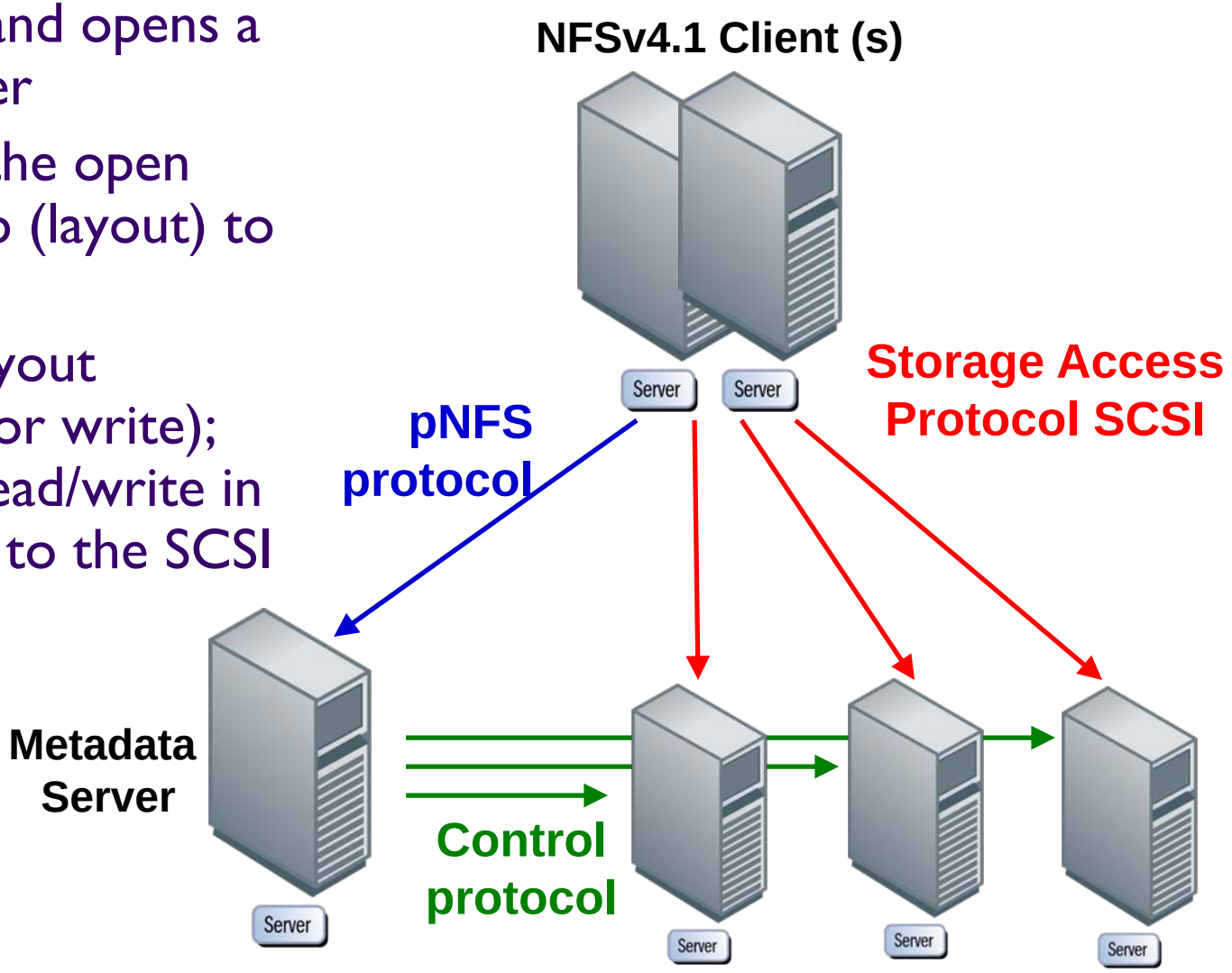
pNFS – NFSv4.1 files access

- Client mounts and opens a file on the server
- Servers grants the open and a file stripe map (layout) to the client
- The client can read/write in parallel directly to the NFSv4.1 data servers



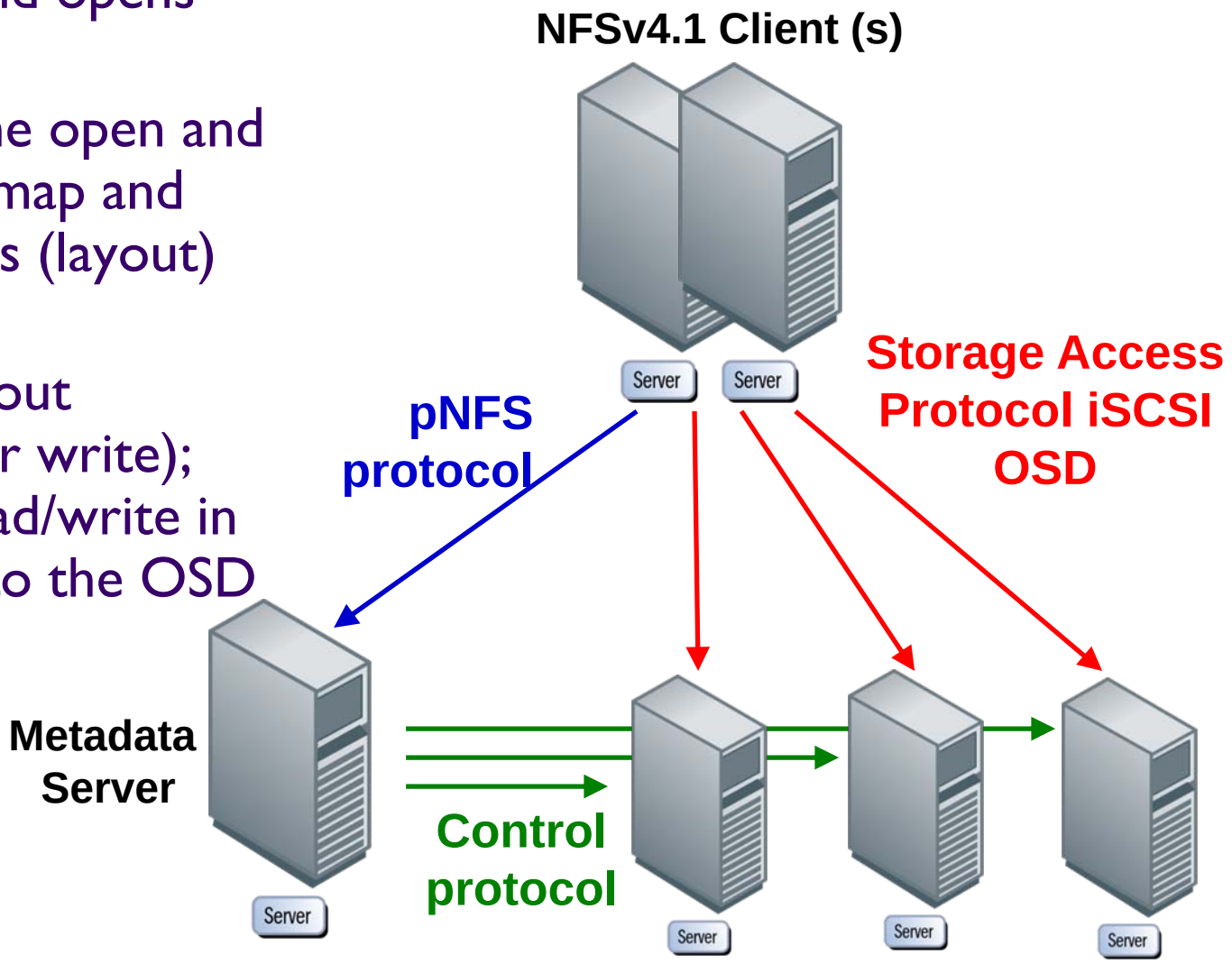
pNFS Blocks Access Model

- Client mounts and opens a file on the server
- Servers grants the open and a block map (layout) to the client
- Based on the layout obtained (read or write); the client can read/write in parallel directly to the SCSI target's



pNFS Objects Access Model

- Client mounts and opens Object
- Servers grants the open and an object stripe map and object capabilities (layout) to the client
- Based on the layout obtained (read or write); the client can read/write in parallel directly to the OSD targets



NFSv4.1 – OpenSource Client Status

- Client and Server
  - Support files (NFSv4.1)
  - Support in progress blocks (SCSI), objects (OSD T10)
  - Client consists of generic pNFS client and "plug ins" for "layout drivers"
- Predicted timeline:
  - Basic NFSv4.1 features 1H2009
  - NFSv4.1 pNFS and layout drivers by 2H2009
  - Linux distributions shipping supported pNFS in 2010

Q&A / Feedback

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

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

- SNIA Education Committee

Mike Eisler
Brian "Beepy" Pawloski
Howard Goldstein
David Black
Omer Asad
Jason Blosil
Mark Carlson
Rob Peglar
Dave Hitz
Ricardo Labiaga
Joshua Konkle
J. Bruce Fields
Joe White
Brent Welch
Ken Gibson
Sachin Chheda
Piyush Shivam
Sorin Faibash
Andy Adamson
Pranoop Ersani
Dave Noveck