Windows HPC Server 2008

Microsoft

Native pNFS Client for Windows HPC Server 2008

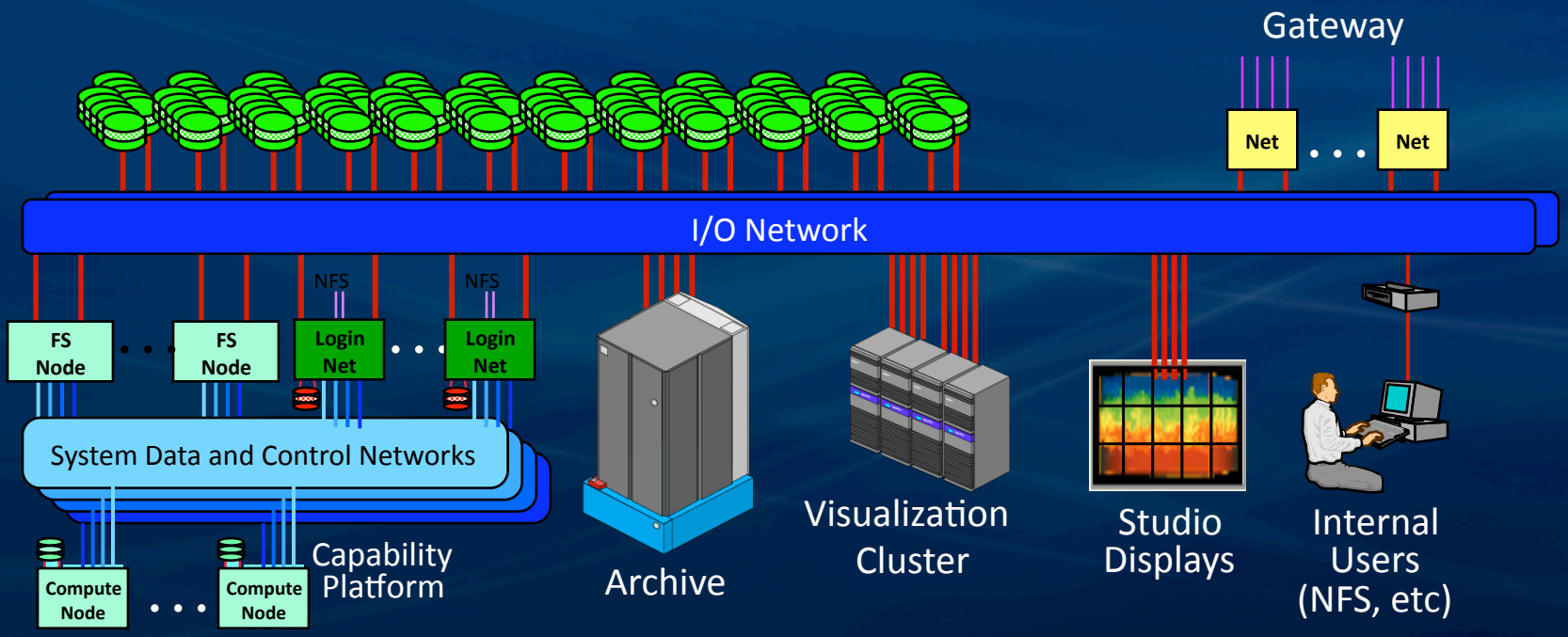
NFS origins

- NFSv2 and NFSv3
  - Proprietary (Sun Microsystems) client/server protocols for distributed filing
  - "Open system"
    - Protocol published
    - Interoperability promoted
  - Stateless, usually UDP-based
  - Other protocols for mount, locks, quotas

NFSv4

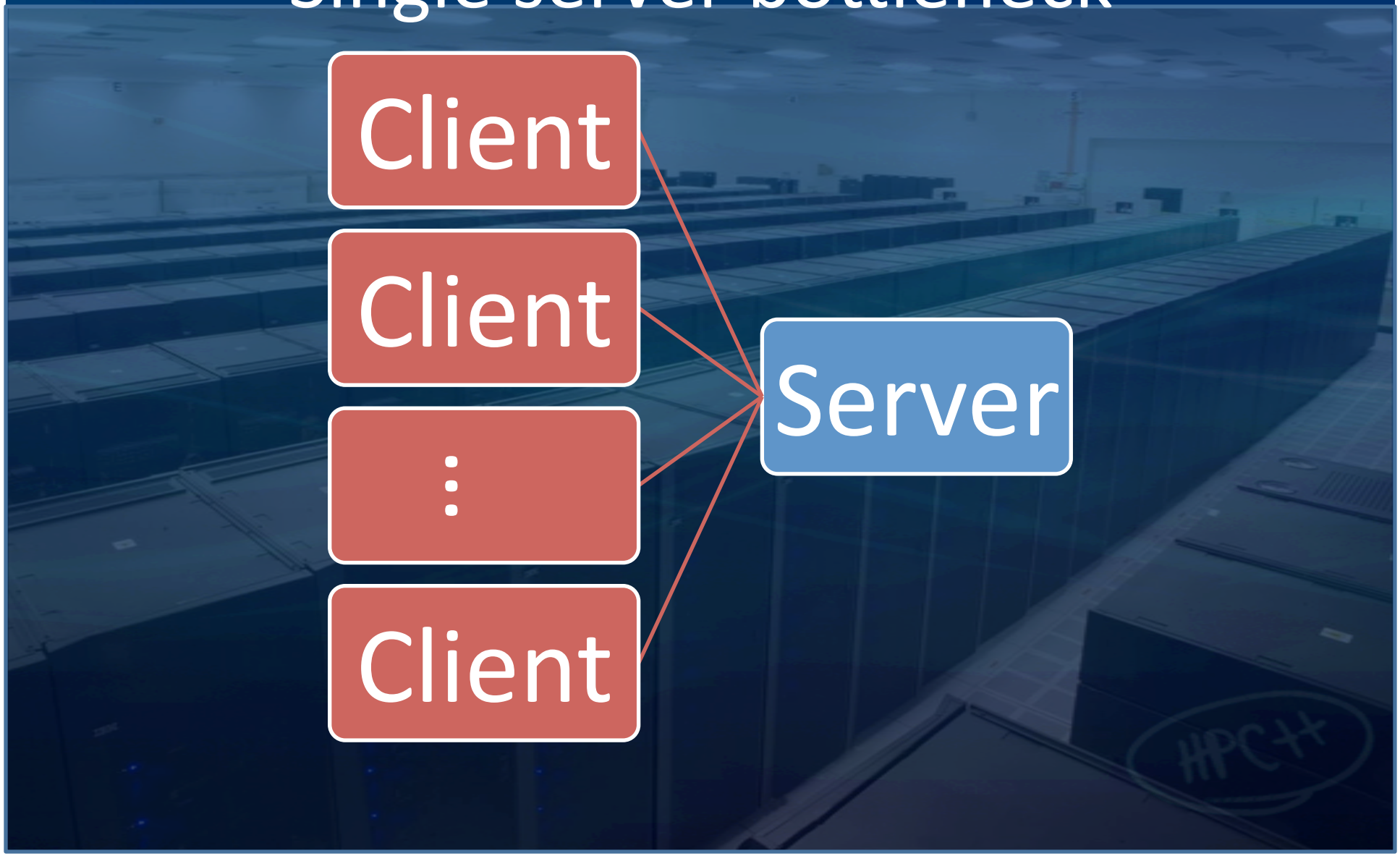
- Control ceded to IETF
- Goals
  - Internet performance
  - Interoperability, internationalization
  - Security, reliability, availability
  - Extensible
- Stateful protocol
  - Open, locks, oplocks, secure channels, callbacks
- RFC 3010, December 2000; RFC 3530, April 2003

NFSv4 and HPC



from ASCI Technology Prospectus, July 2001

Single server bottleneck



Meeting HPC storage needs

- HPC demands for scalable storage are met through innovative, proprietary, non-interoperable solutions
  - Lustre, GPFS, PanFS, PVFS2 dominate
  - Investment in proprietary solution has high risk of lock-in or loss
- pNFS insulates storage architects from these risks
  - Neutral ground through standardization
  - Continues to admit vendor innovation
  - Pools customer investment
  - Spreads investment across more vendors

Parallel file systems

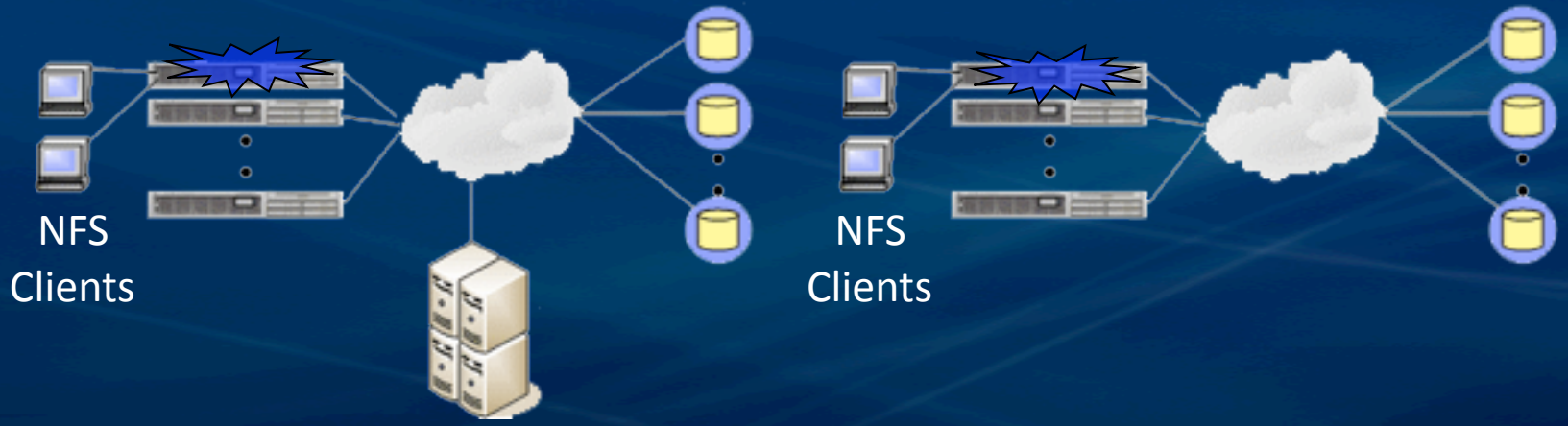
Asymmetric

- Direct access to storage
- Separate metadata servers
- File, object, or block access

Symmetric

- Direct access to storage
- Each node is a fully capable client and metadata server
- File access

NFS advantages and obstacles



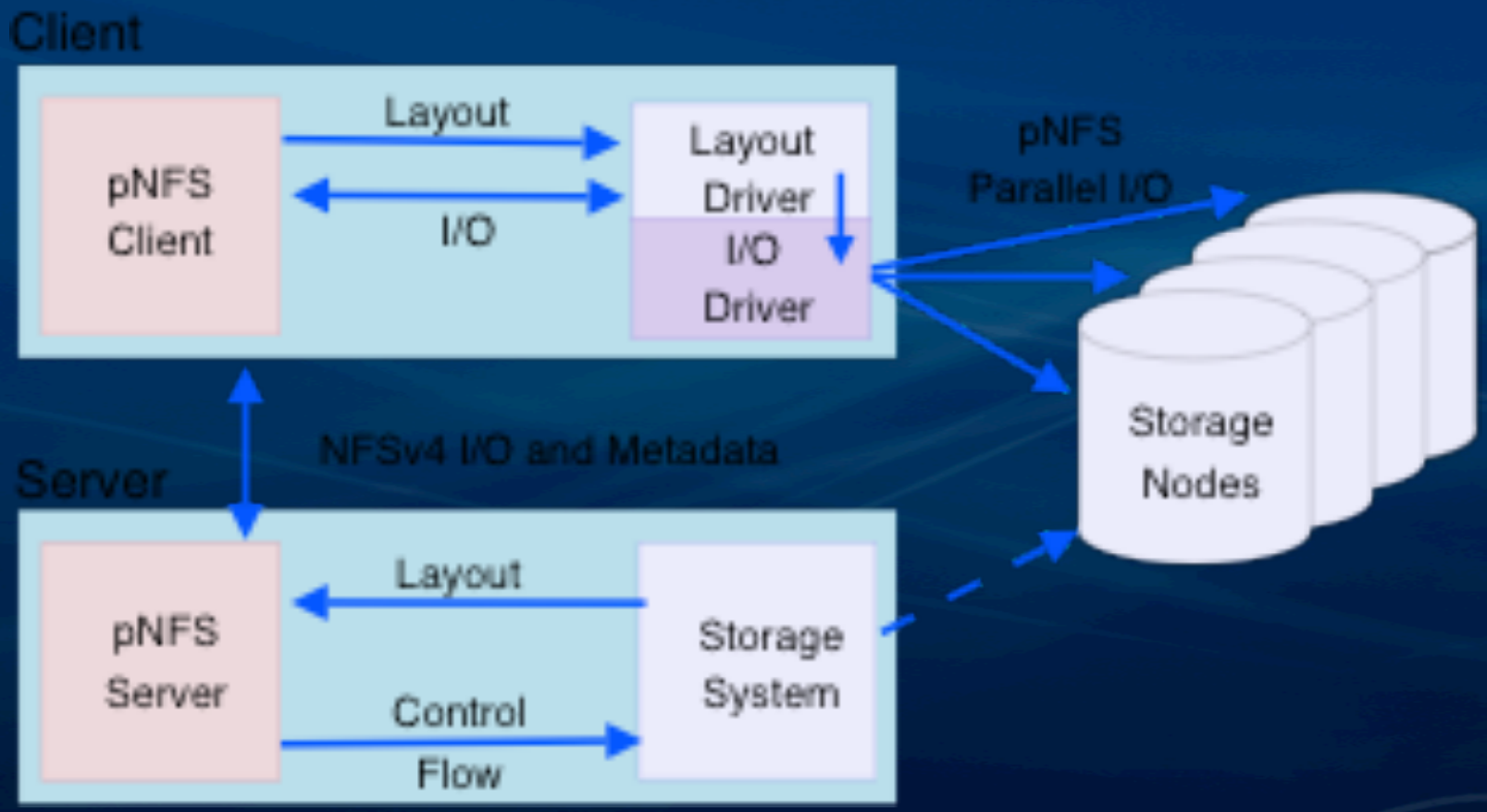
- ✓ Security
- ✓ Heterogeneity
- ✓ Transparency

- ✗ Performance
- ✗ Scalability

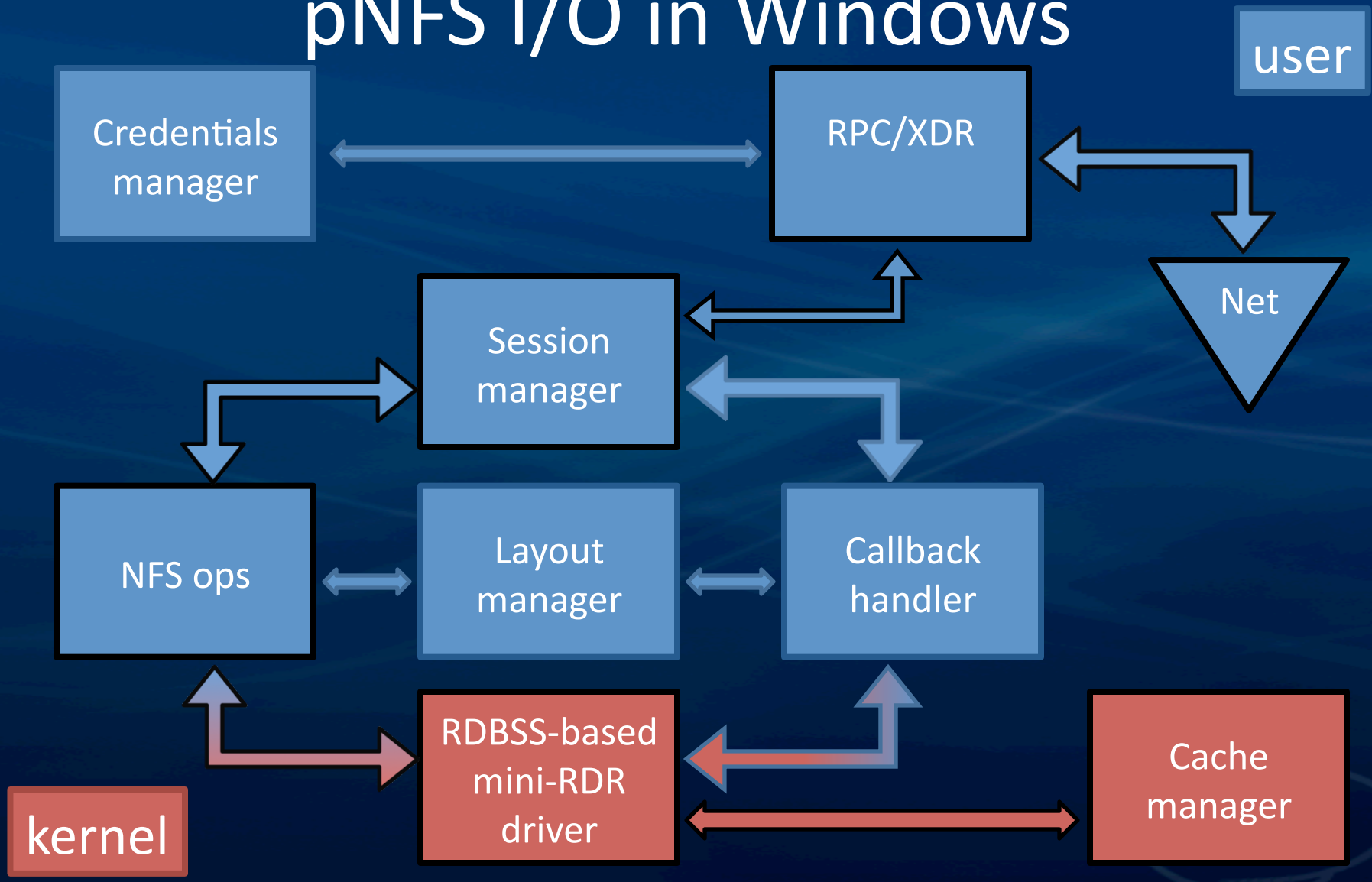
pNFS worldview

- pNFS extends NFSv4
  - parallel, multi-path transfers
  - complex topologies
- A layout associates a file with a device ID
- LAYOUTGET returns the device ID for a given file
  - The handle for a specific storage device topology
- GETDEVICEINFO returns the storage device topology for a given device ID

pNFS


Client
pNFS Client
Layout
I/O
Layout Driver
I/O Driver
pNFS Parallel I/O
Storage Nodes
NFSv4 I/O and Metadata
Server
pNFS Server
Layout
Control Flow
Storage System

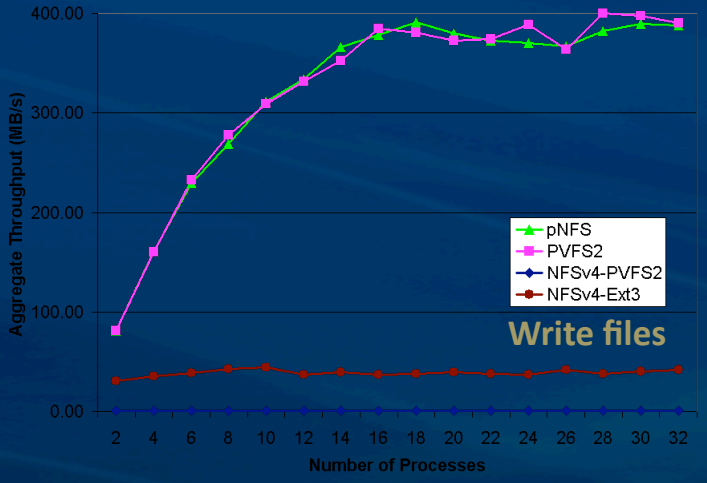
pNFS I/O in Windows

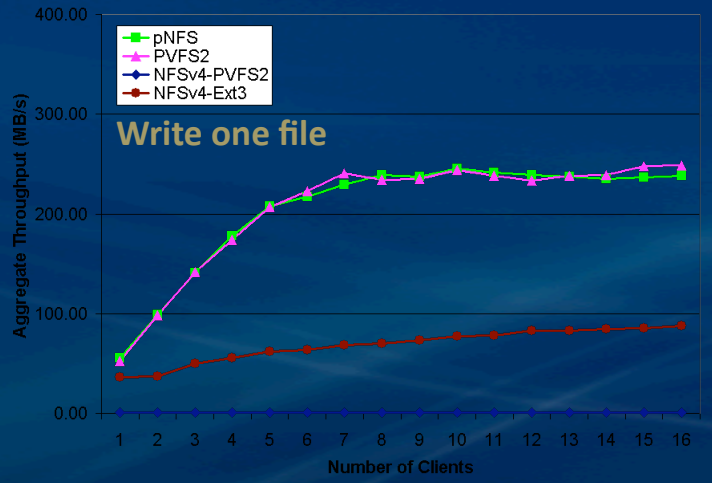


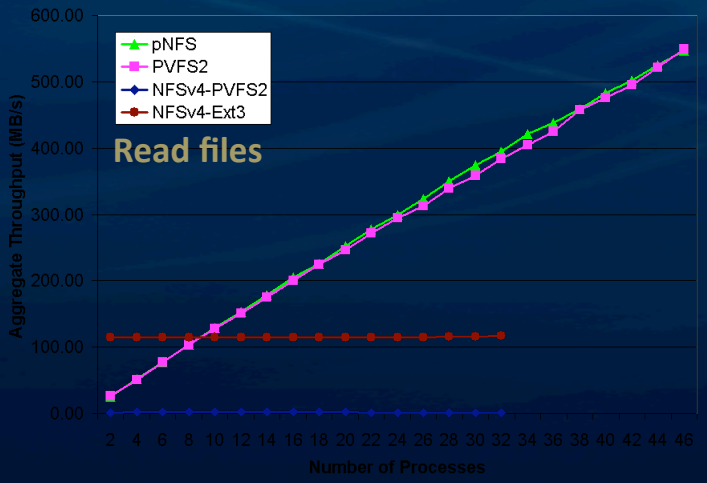
The path to pNFS in Linux

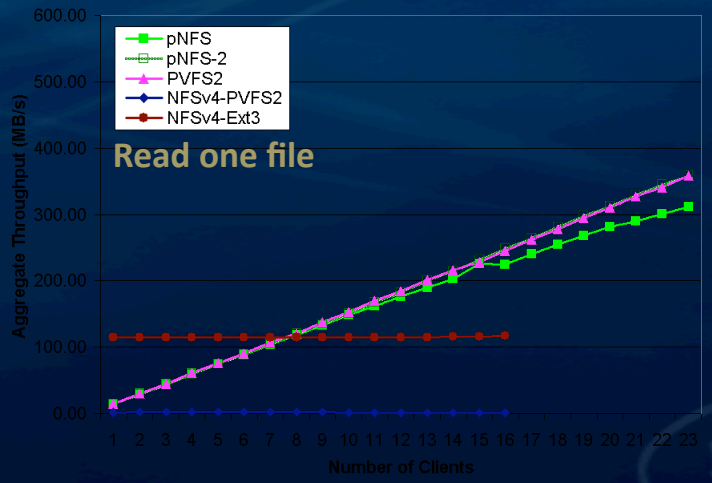
- ✓ Convince people it will work
- ✓ Get on the IETF agenda
- ✓ Draft a protocol standard
  - ✓ Make sure it addresses HPC issues
- ✓ Convince people to fund implementation
- ✓ Gather frequently to test interoperability
- Convince Linux maintainers to accept patches
- Convince Linux distributions to support pNFS

Building the Case









Roadmap

- Standardization
  - Target: 2008
- Implementation
  - Target: 2009
- Distribution
  - Target: 2010 (HPC, other early adopters)
- Enterprise distribution
  - Target: 2011

Implementations

- CITI, Sun, StorSpeed, Seagate, Panasas, Ohio Supercomputing Center, NetApp, LSI, IBM, EMC, Carnegie Mellon, DESY, BlueArc
- Frequent interoperability testing
  - Connectathon, Bake-a-thons
- Functionally correct and interoperable
  - Linux, Solaris clients
  - DESY, EMC, IBM, Linux, NetApp, Panasas, Solaris servers

Windows client status

- Passing most “Basic Connectathon” tests
  - Interoperability testing began at last Bake-a-thon
- Layout implementation begins in 4Q09
- Open source distribution (more hands, more eyes) to begin 1H10
- Functional completeness 4Q10
  - With continuing development and tuning by open source developers

Linux implementation status

- Maintainers work with developers to engineer kernel patches
  - Linux kernel version increments approximately quarterly
  - Ultimately Linus Torvalds decides
- NFSv4.1 is more than pNFS
  - Sessions communication layer, required for pNFS
  - Directory delegation
- Client and server fore and back sessions channel in Linux 2.6.32 kernel

There is much left to do

- Administration tools
  - Metadata server management
  - Volume management
- Performance at scale
  - Instrumentation, measurement, tuning
  - Small-scale file striping performance under way at CITI (fewer than 20 nodes)
- Metadata striping
- Windows HPC Server 2008 metadata server