

NFSv4 Protocol Development

**Tom Haynes Ph.D.,
Senior Engineer, NetApp**

Contributors;

Alex McDonald, NetApp

Brian Pawlowski, NetApp

Trond Myklebust, Linux

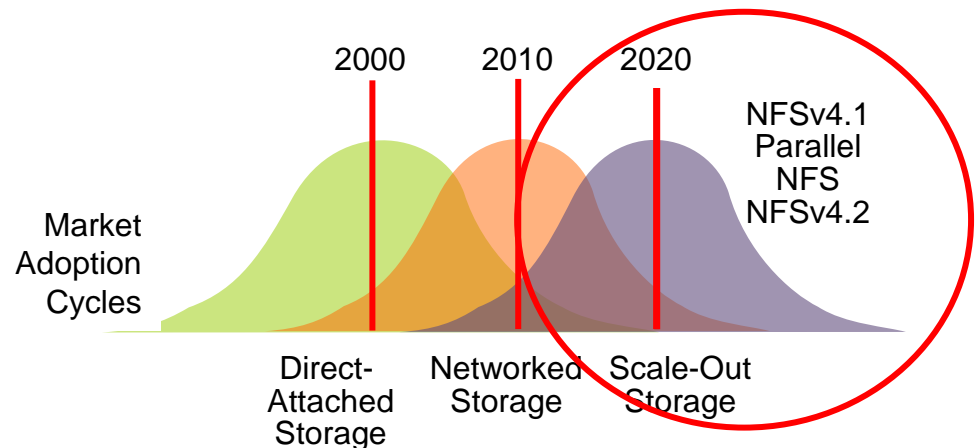
Session Agenda

- ❑ The NFSv4 protocol undergoes a repeated lifecycle of definition and implementation. The presentation will be based on years of experience implementing server-side NFS solutions up to NFSv4.1.
- ❑ We'll examine the lifecycle from a commercial implementation perspective; what goes into the selection of new features, the work with the IETF NFS standards body, the development process and how these features are delivered, and the impact these features have on end users.
- ❑ We'll also cover the work of Linux NFS developers and provide suggestions for file system developers based on these and vendor experiences; and finally, we'll discuss how implementation and end-user experience feeds back into the protocol definition.

- ❑ NFS; You're Soaking In It Now
- ❑ Features of NFSv4, NFSv4.1 and pNFS
- ❑ New features in NFSv4.2
- ❑ Linux developments
- ❑ The Virtualized Datacenter
- ❑ Filesystem Implications
- ❑ Conclusions

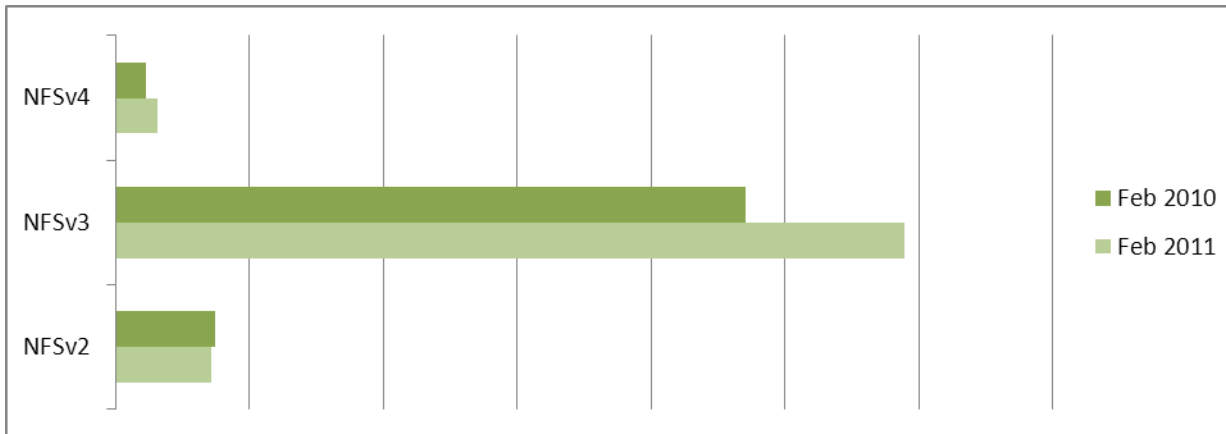
NFS; You're Soaking In It Now

- ❑ NFS is ubiquitous and everywhere
- ❑ NFS doesn't stand still
 - ❑ NFSv2 in 1983, through NFSv4.1 in 2010
 - ❑ NFSv4.2 to be agreed at IETF shortly
 - ❑ Faster pace for minor revisions
- ❑ NFSv3 very successful
 - ❑ Protocol adoption is over time, and there have been no big incentives to change



NFSv4 is forward looking

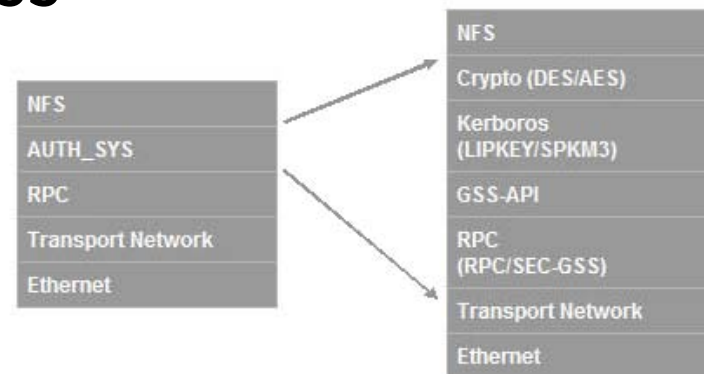
- NFSv4 is forward looking
 - Reflects today's expressed needs & tomorrow's requirements
 - Demand currently small but growing
 - Reflects server view; but client view should match



- User requirements drive developments in NFS

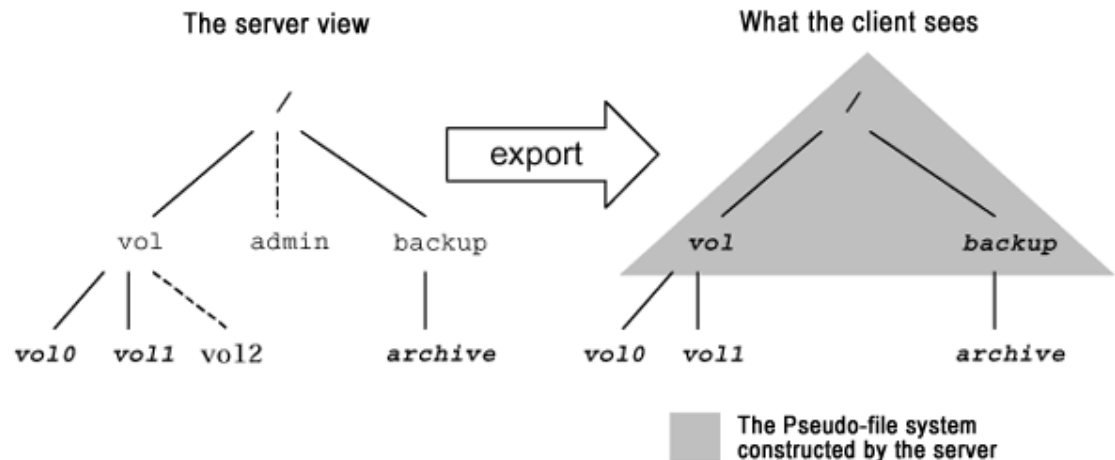
NFSv4 Major Features; Security

- ❑ Strong security framework
- ❑ Access control lists (ACLs) for security and Windows® compatibility
- ❑ Mandatory security with Kerberos
 - ❑ Negotiated RPC security that depends on cryptography, RPCSEC_GSS



NFSv4 Major Features; Namespace

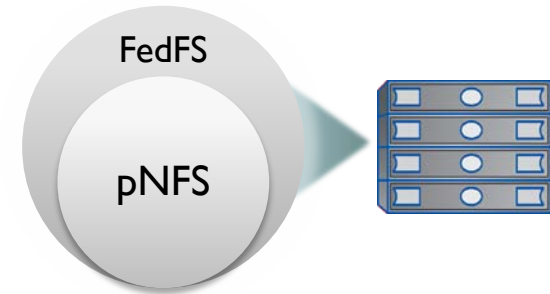
- Uniform and “infinite” namespace
 - Moving from user/home directories to datacenter & corporate use
 - Meets demands for “large scale” protocol
- No automounter required
 - Simplifies administration



NFSv4 Major Features; FedFS

❑ Federated File System

- ❑ Uniform namespace that has local and geographically global referral infrastructure
 - ❑ Accessible to unmodified NFSv4 clients
 - ❑ Addresses directories, referrals, nesting, and namespace relationships
- ## ❑ Client finds namespace via DNS lookup
- ❑ Sees junctions (directories) and follows them as NFSv4 referrals



NFSv4 Major Features; Client Independence

- ❑ NFSv4 gives client independence
 - ❑ Previous model had “dumb” stateless client
 - ❑ Server had the smarts
- ❑ Pushes work out to client through delegations & caching
- ❑ Why?
 - ❑ Compute nodes work best with local data
 - ❑ NFSv4 eliminates the need for local storage
 - ❑ Exposes more of the backend storage functionality
 - ❑ Client can help make server smarter by providing hints

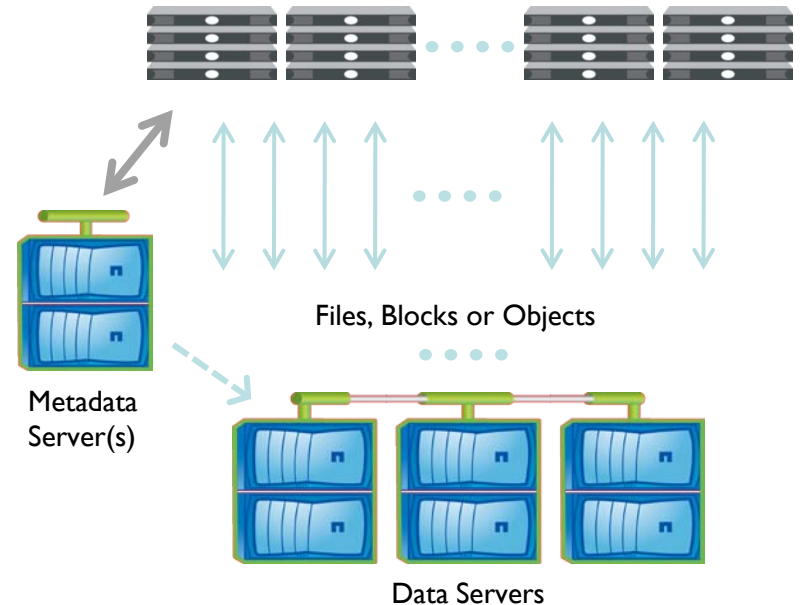
NFSv4 Major Features; Layouts

- ❑ Layouts
 - ❑ Files, objects and block layouts
 - ❑ Provides flexibility for storage that underpins it
 - ❑ Location transparent
 - ❑ Striping and clustering
- ❑ Examples
 - ❑ EMC does blocks
 - ❑ Panasas does objects
 - ❑ Oracle Solaris does files

NFSv4.1 Major Features; Performance

- NFSv4.1 (pNFS) can aggregate bandwidth
 - Modern approach; relieves issues associated with point-to-point connections

- pNFS Client
 - Client read/write a file
 - Server grants permission
 - File layout (stripe map) is given to the client
 - Client parallel R/W directly to data servers
- Removes IO Bottlenecks
 - No single storage node is a bottleneck
 - Improves large file performance
- Improves Management
 - Data and clients are load balanced
 - Single Namespace



NFSv4.1 Major Features; Sessions

- ❑ NFSv3 server never knows if client got reply msg
- ❑ NFSv4.1 introduces Sessions
 - ❑ Major protocol infrastructure change
 - ❑ Exactly Once Semantics (EOS)
 - ❑ Bounded size of reply cache
 - ❑ Unlimited parallelism
- ❑ A session maintains the server's state relative to the connections belonging to a client.

- ❑ Upstream (Linus) Linux NFSv4.1 client support
 - ❑ Basic client in Kernel 2.6.32
 - ❑ pNFS support (files layout type) in Kernel 2.6.39
 - ❑ Support for the 'objects' and 'blocks' layouts was merged in Kernel 3.0 and 3.1 respectively
- ❑ Full read and write support for all three layout types in the upstream kernel,
 - ❑ O_DIRECT reads and writes are not yet supported.



Linux Client and NFSv4.1

- ❑ pNFS client support in distributions
 - ❑ Fedora 15 was first for pNFS files
 - ❑ Kernel 2.6.40 (released August 2011)
- ❑ Red Hat Enterprise Linux version 6.2
 - ❑ “Technical preview” support for NFSv4.1 and for the pNFS files layout type
- ❑ Other Open Source
 - ❑ Microsoft NFSv4.1 Windows client from CITI

- New direction for protocol changes

“Instead of server vendors putting in new features that might attract application developers and vendors, they’re approaching server vendors requesting features that are available on local storage, but that you can’t get to currently via NFS”

New Features in NFSv4.2

- ❑ Server-Side Copy (SSC)
 - ❑ Removes one leg of the copy
 - ❑ Destination reads directly from the source
- ❑ Application Data Blocks
 - ❑ Allows definition of the format of file
 - ❑ Examples: database or a VM image.
 - ❑ INITIALIZE blocks with a single compound operation
 - ❑ Initializing a 30G database takes a single over the wire operation instead of 30G of traffic.

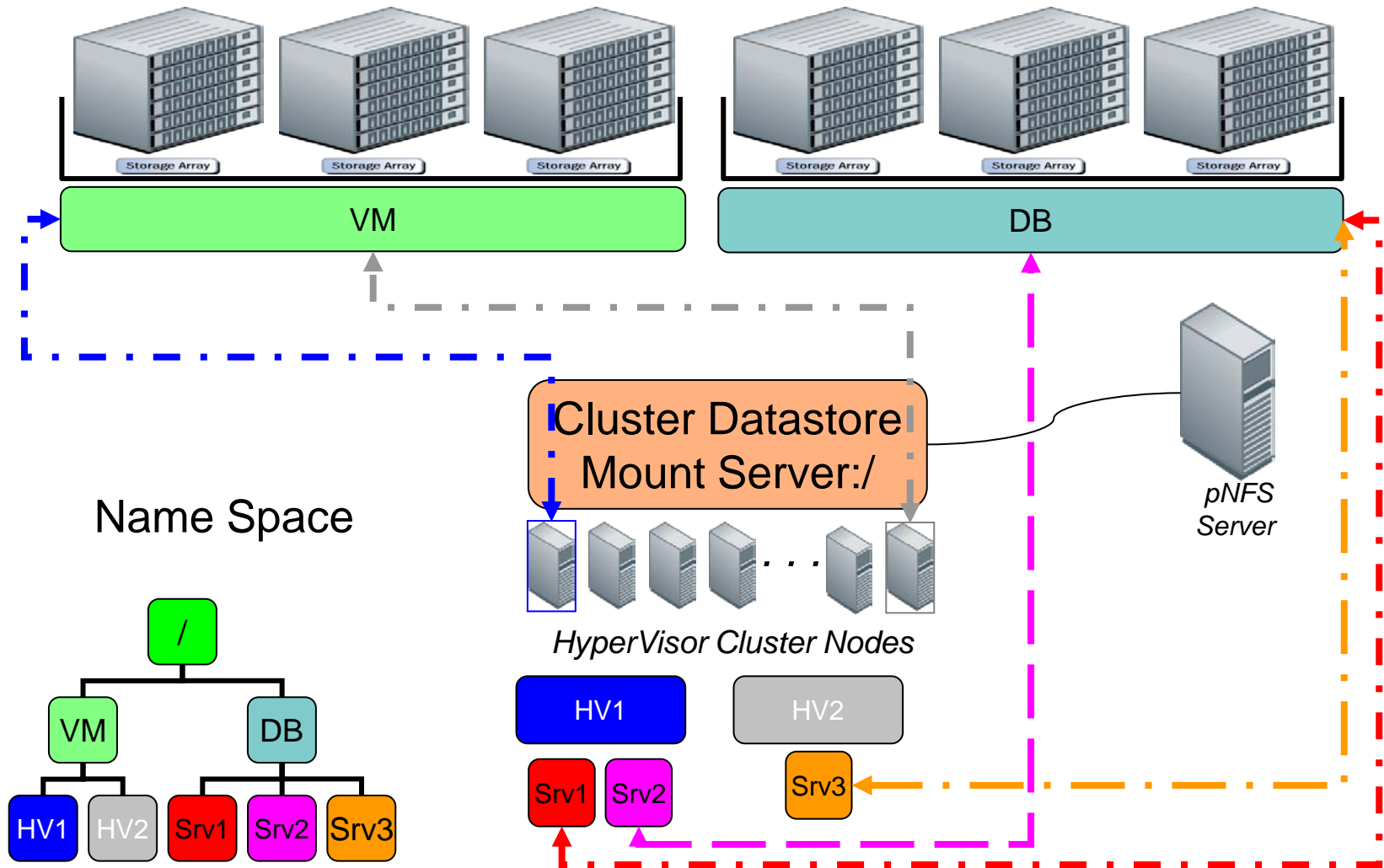
New Features in NFSv4.2

- ❑ Space reservation
 - ❑ Ensure a file will have storage available
- ❑ Sparse file support
 - ❑ “Hole punching” and the reading of sparse files
- ❑ Labeled NFS (LNFS)
 - ❑ MAC checks on files
- ❑ IO_ADVISE
 - ❑ Client or application can inform the server caching requirements of the file

NFSv4.2 Protocol Work

- ❑ Protocol goals
 - ❑ Driven by user & application requirements
 - ❑ Support sophisticated storage
 - ❑ Minimalistic
 - ❑ Keep shared info between client & server to the minimum needed
 - ❑ Gives more flexibility to storage system
- ❑ Run through the IETF
 - ❑ Major industry vendors involved
 - ❑ Protocol moves from proposed to draft to approved
- ❑ Spencer Shepler talk later this week
 - ❑ IETF NFSv4 Working Group; What's Next?

The Virtualized Datacenter



Virtualization; The Game Changer

- ❑ Server virtualization a major area of use
 - ❑ VMware, Citrix Xen
- ❑ Demands of 1000s of images on 100s of servers
- ❑ Requirements from a storage system
 - ❑ Single system image, Resiliency, Load balanced, Transparent & non-disruptive upgrades...
- ❑ NFS a better fit to requirements than SANs
 - ❑ Use cases much wider & broader
 - ❑ Ubiquitous like Linux; available everywhere

- ❑ Files, blocks, objects can co-exist in the same storage network
 - ❑ Can access the same filesystem; even the same file
- ❑ NFS flexible enough to support unlimited number of storage layout types
 - ❑ Three IETF standards, files, blocks, objects
 - ❑ Others evaluated experimentally
- ❑ NAS vs SAN; no-one cares any more
 - ❑ IETF process defines how you get to storage, not what your storage looks like
 - ❑ NetApp pNFS implemented differently from Panasas or BlueArc or EMC or...

- ❑ NFS has more relevance today for commercial, HPC and other use cases than it ever did
 - ❑ Features for a virtualized data centers
- ❑ Developments driven by application requirements
- ❑ Adoption slow, but will continue to increase
 - ❑ NFSv4 support widely available
 - ❑ New NFSv4.1 with client & server support
- ❑ NFS defines how you get to storage, not what your storage looks like

Thank you!

Questions and Answers

Contact Info:
alexmc@netapp.com