



Go further, faster™

# Requirements for NFSv4.2

## IETF-76

draft-eisler-nfsv4-minorversion-2-  
requirements-02

### 2009-11-12

Mike Eisler  
Senior Technical Director





# Contents

- Disclaimer
- Motivation for NFSv4.2
- Proposed Next Steps



## Disclaimer

- draft-eisler-nfsv4-minorversion-2-requirements-02 is not an official work item
- There is no commitment from IESG/IETF to charter the NFSv4 WG (or any other WG) to charter NFSv4.2
  - I am seeking that commitment



# Motivation for NFSv4.2

- Storage in general and file access particular needs to react to IT trends
  - NFSv4.1 was a reaction to the trends toward
    - scale out (aka grid, aka cloud) computing: pNFS
    - high speed networking: sessions and exactly once semantics
      - All NFSv3/NFSv4.0 vendors/customers continue to struggle with limitations of XID-based reply cache
- Storage is now faced with these major trends:
  - space/efficiency demands
    - drivers are cost of energy and backup times
  - flash
    - flash is now the least expensive storage medium compared to disk and DRAM when measuring cost/IOPS.
  - compliance
    - There are laws that regulate management of customer data
    - Lots more data, lots more devices increases the probability of mismanagement
      - automation might help
- NFSv4.1 was a big change relative to NFSv4.0
  - we certainly will find we didn't get every thing right



# Space

- Disk capacities are doubling on 1-2 year cycles
- Disk access times are not
- Neither are allotments for data management operations
  - e.g. Data Backups
- Energy price spikes are compounding problem
  - data centers are not going to expand
  - new data centers in expensive regions are not going to be built
- Storage industry has responded with “De-Duplication”
  - NFS needs to catch up:
    - space reporting, hole punching, de-dupe mapping on read/write, ...



# Efficiency

- Peer-to-peer networking has been proven
  - For some workloads, NFS clients and servers would benefit from this model
    - see draft-myklebust-nfsv4-pnfs-backend
- Today pNFS allows I/O offload, but not meta-data offload
  - This doesn't have to be the case, see draft-eisler-nfsv4-pnfs-metastripe
- File copy is more efficient if NFS servers take care of it
  - We now have APIs on some NFS clients for performing file copy
  - draft-lentini-nfsv4-server-side-copy has reached WG consensus



# Flash

- Adding flash to storage arrays is goodness
  - doesn't require changes to storage protocols
- However, the value of flash is best realized on the client-side
  - We could cede this ground to Direct-Attached-Storage
    - Traumatic for data centers oriented toward network storage
  - Or we could embrace use cases that leverage client-side flash for network storage
    - caching
      - Sub-file caching is needed



# Compliance

- Data continues to expand rapidly
- The rules for managing this data and the penalties for mismanagement seem to be expanding nearly as rapidly
- The manual approach does not scale
- An immutable compliance attribute needs to be settable on a file when it is created
- Security labeling is a framework for reducing mistakes, and making malicious misuse harder
- See [draft-quigley-nfsv4-sec-label](#)





## Bug fixes/minor enhancements in protocol

- Examples include:
  - pNFS connectivity problem reporting
  - trunking discovery
  - hints of I/O pattern (much harder to discern sequential when pNFS is in use)



## Proposed Next Steps

- Make draft-eisler-nfsv4-minorversion-2-requirements-02 a work item of NFSv4
  - When: November 16, 2009
- Drive to WG Consensus
  - When: January, 2010
- WG Last Call
  - When: February, 2010
- Re-charter WG based on final requirements
  - When: March, 2010 (before Anaheim IETF meeting)