

Design and Installation of Sequoia's 55PB Lustre+ZFS File System

HPC User Forum, Dearborn, MI
September 19th, 2012

Marc Stearman
Parallel File Systems Operations Lead

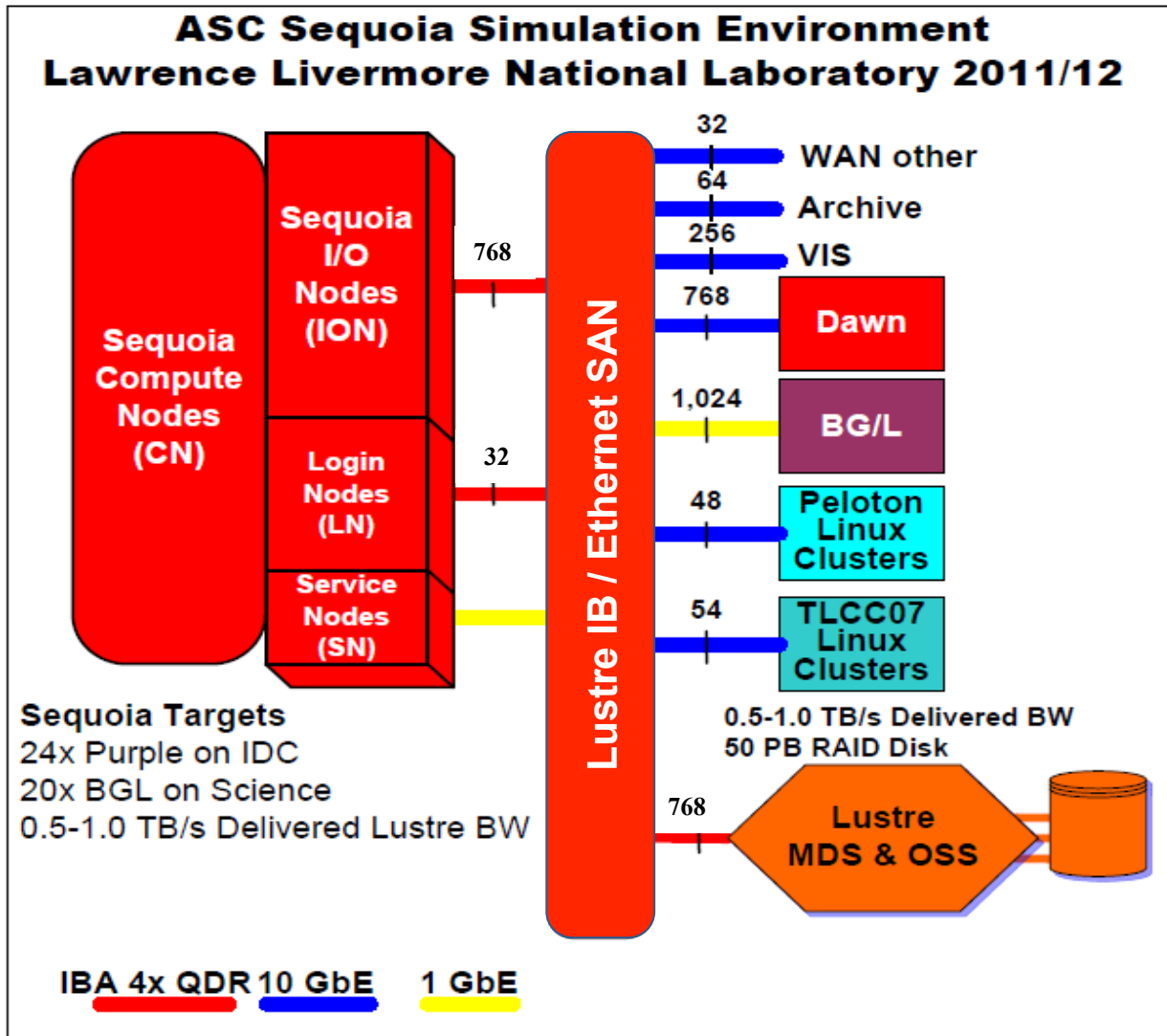
 Lawrence Livermore
National Laboratory

LLNL-PRES-582221

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC



Sequoia Compute Platform



▪ **Sequoia Stats:**

- 20PF Compute Platform
- 96 Racks
- 1.5 Million Cores
- 1.5 PB Memory
- 768 I/O Nodes - QDR IB
- Liquid Cooled



Sequoia I/O Infrastructure

■ Requirements

- 50PB file system
- 500GB/s minimum, 1TB/s stretch goal
- QDR InfiniBand SAN connection to Sequoia
- Must integrate with existing Ethernet infrastructure

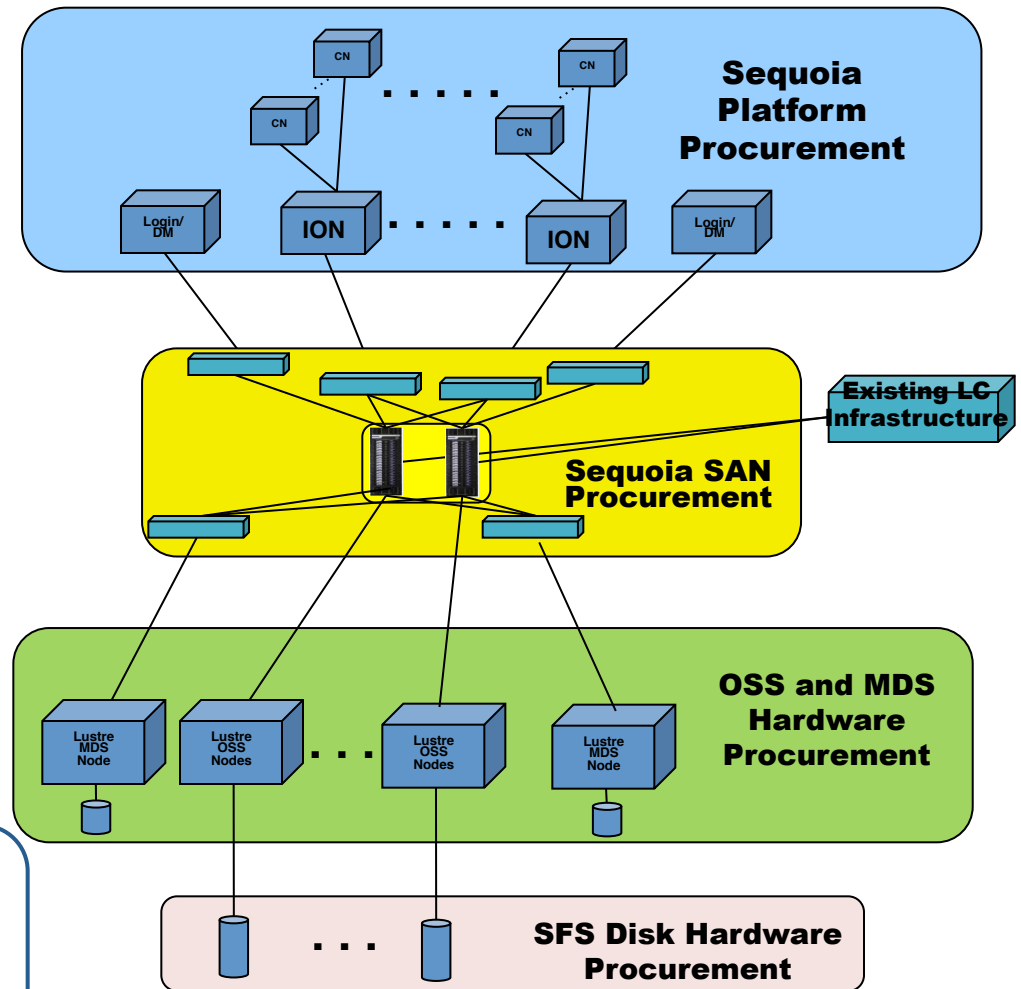
■ >\$20M Budget



Across five procurements dominated by RAID file system and IB SAN hardware procurements

Phased Bandwidth Delivery

- Phase 1: 10% Oct 2011
- Phase 2: 50% Dec 2011
- Phase 3: 100% Feb 2012



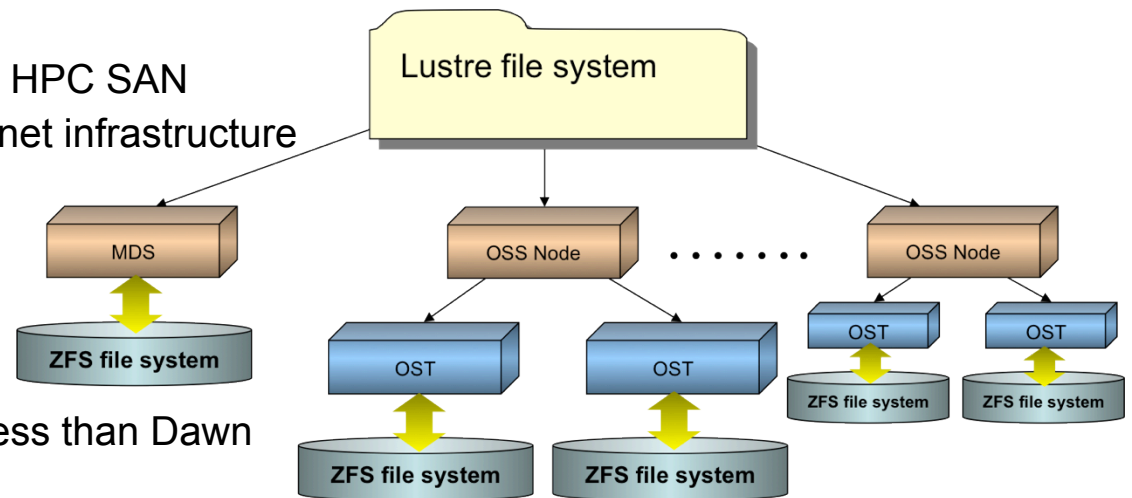
Sequoia I/O Challenges Abound

- **ZFS-based Lustre**

- Has never been done before
- Is dependent on ongoing local development and D&E contract investments
- Is the pioneering implementation of new backend fs for Lustre community
- Will be buggy, does not meet 1TB/s stretch goal without performance improvements

- **InfiniBand SAN**

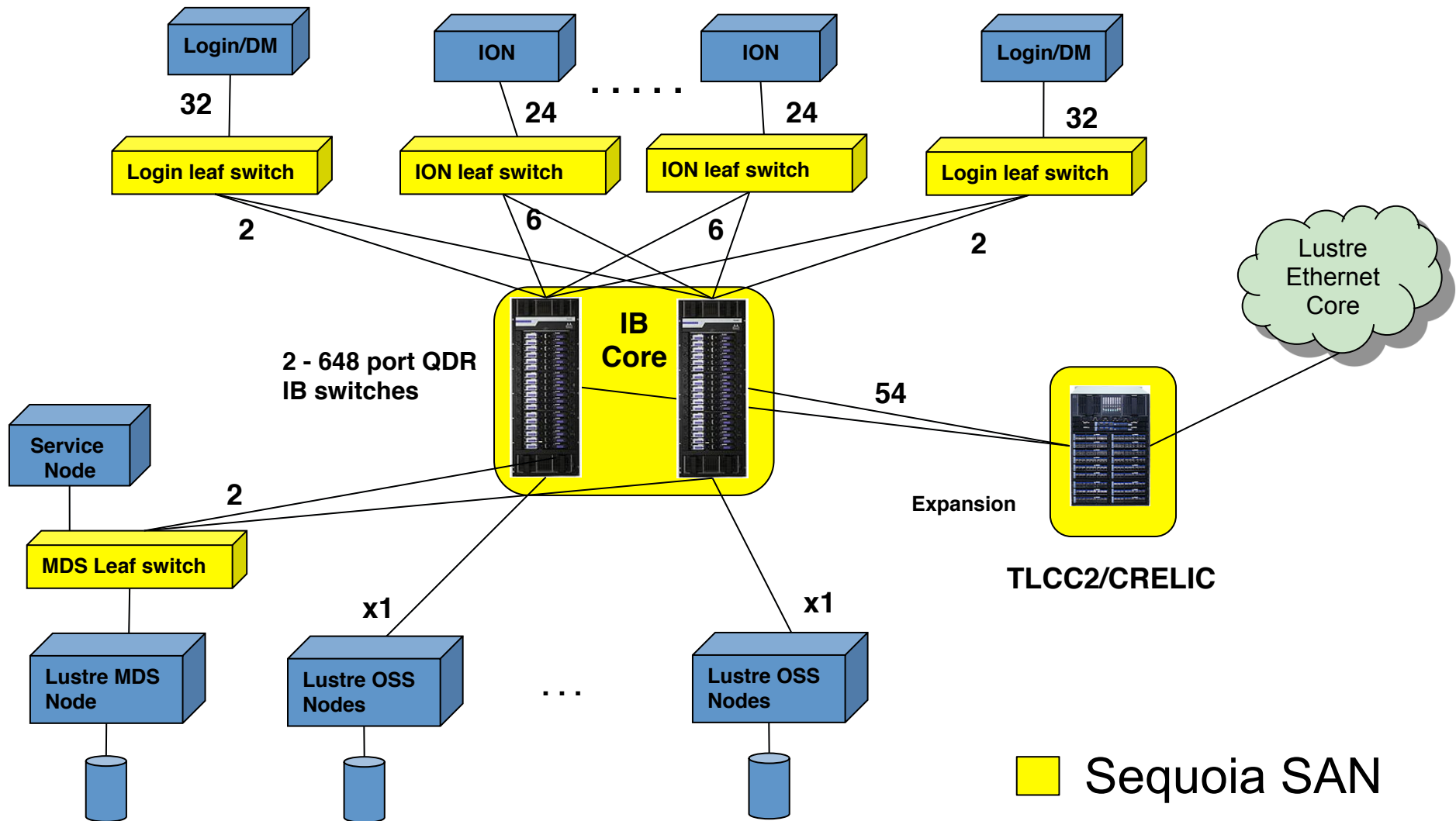
- Lack of tools and experience as HPC SAN
- Need to bridge to existing Ethernet infrastructure



- **Sequoia IONs**

- ION/CN ratio is a factor of two less than Dawn and BG/L
- Lustre client performance previously an issue on BG IONs

Sequoia SAN Architecture

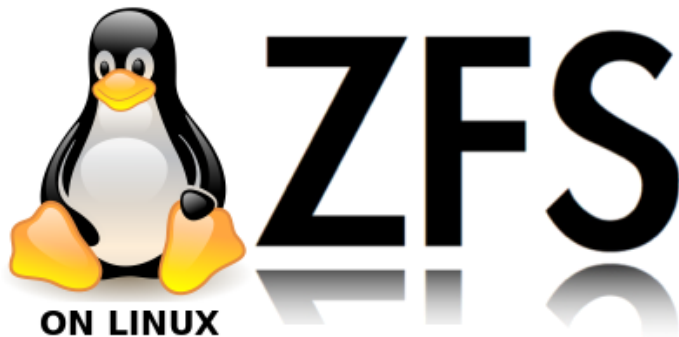


Why ZFS

- Scalability
 - Massive storage capacity
 - 2^{64} bytes per object
 - 2^{78} bytes per pool
 - Dynamic striping
 - Single OST per OSS
- Cost
 - Combined RAID+LVM+FS
 - Built for inexpensive disk
 - No vendor lock in
 - All OpenSource
- Data Integrity
 - Copy-on-Write
 - Checksums
 - Metadata and block data
 - Checksums verified on read
 - Automatically repairs damage
 - Multiple copies of metadata
 - Small amount of storage
 - Spread over different disks
 - Ditto Blocks
 - Redundancy – Stripes, Mirrors, RAIDZ, RAIDZ2, RAIDZ3

Why ZFS

- Manageability
 - Online everything
 - Scrubbing
 - Resilvering
 - Pool expansion
 - Configuration changes
 - Fast filesystem creation
 - High quality utilities
- Features
 - Snapshots
 - Clones
 - Compression
 - Deduplication
 - Dataset send/receive
 - Advanced Caching
 - ZFS Intent Log (ZIL)
 - L2ARC
 - Adaptive Endianness
 - Quotas



D&E Efforts

■ Lustre OSD work



- Abstracts all backend storage access into a single portable API
- Enables Lustre to use any backend file system with a minimum amount of “glue” logic

- Idiskfs
- ZFS
- btrfs (future work)



■ SMP Checksum Performance

- Parallelize checksums across multiple threads

■ ZFS Optimization

■ Quotas



Procurement Status

- **RAID Hardware**

- Contract Awarded to IAS/NetApp

- **SAN Infiniband Hardware**

- Contract Awarded to Advanced HPC/Mellanox

- **OSS**

- OSS Contract Awarded to Appro

- **MDS**

- Supermicro Westmere nodes, with RAID Inc. JBOD and OCZ Talos2 SSDs

- **Sequoia Platform**

- All 96 racks at LLNL, #1 on June 2012 Top 500 list



Sequoia Storage Hardware (OSS)

- **NetApp E5400**

- 60-bay 4U Enclosure with 2 RAID controllers
- 3TB SAS drives
- 180TB RAW capacity
- IB Host Attached



OSS uses TLCC2 design

Appro GreenBlade

Intel Xeon E5-2670 @ 2.60GHz

Dual Socket, 8 core

64GB RAM

QDR Mellanox ConnectX-3 IB down (LNET)

Dual Port QDR ConnectX-2 HCA (SRP to Disk)

Sequoia Storage Hardware (MDS)



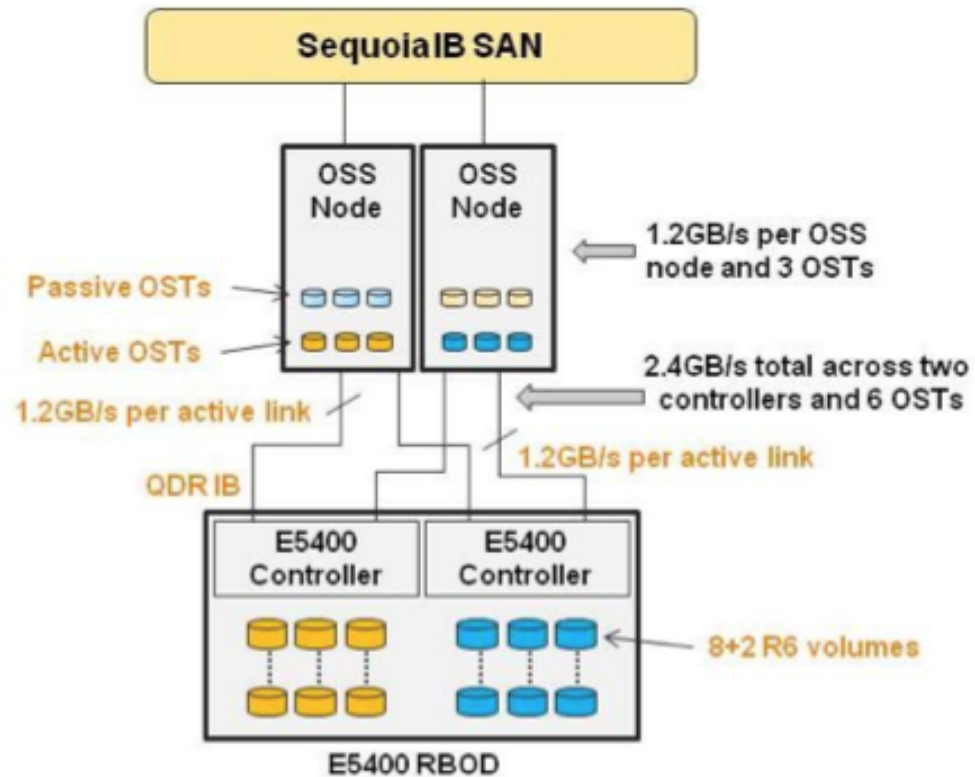
- JBOD
 - Two 24-bay Enclosures
 - Dual SAS controllers
 - 40 – 1TB OCZ Talos2 SSDs
 - 40TB RAW Capacity
 - Configured as RAID10
 - Partitioned smaller to extend MTBF
 - SAS 6Gb/s Host Attached

- Supermicro X8DTH
 - Intel Xeon X5690 @ 3.47GHz
 - Dual Socket, 6 core (24 cpus with Hyperthreading)
 - 192GB RAM
 - JBODS with OCZ Talos2 SDDs (40 Drives, SAS connected using ZFS RAID10)
 - Configure as a failover pair (active/passive) for reliability



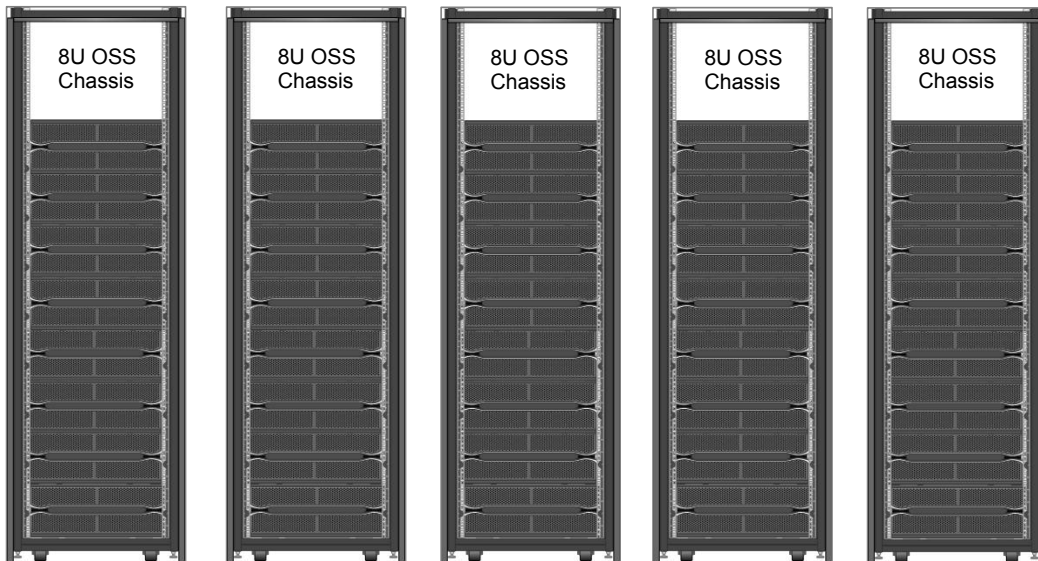
OSS Connectivity

- 2 OSS nodes per E5400 as a failover pair
- Using RHEL6 multipath rdac drivers

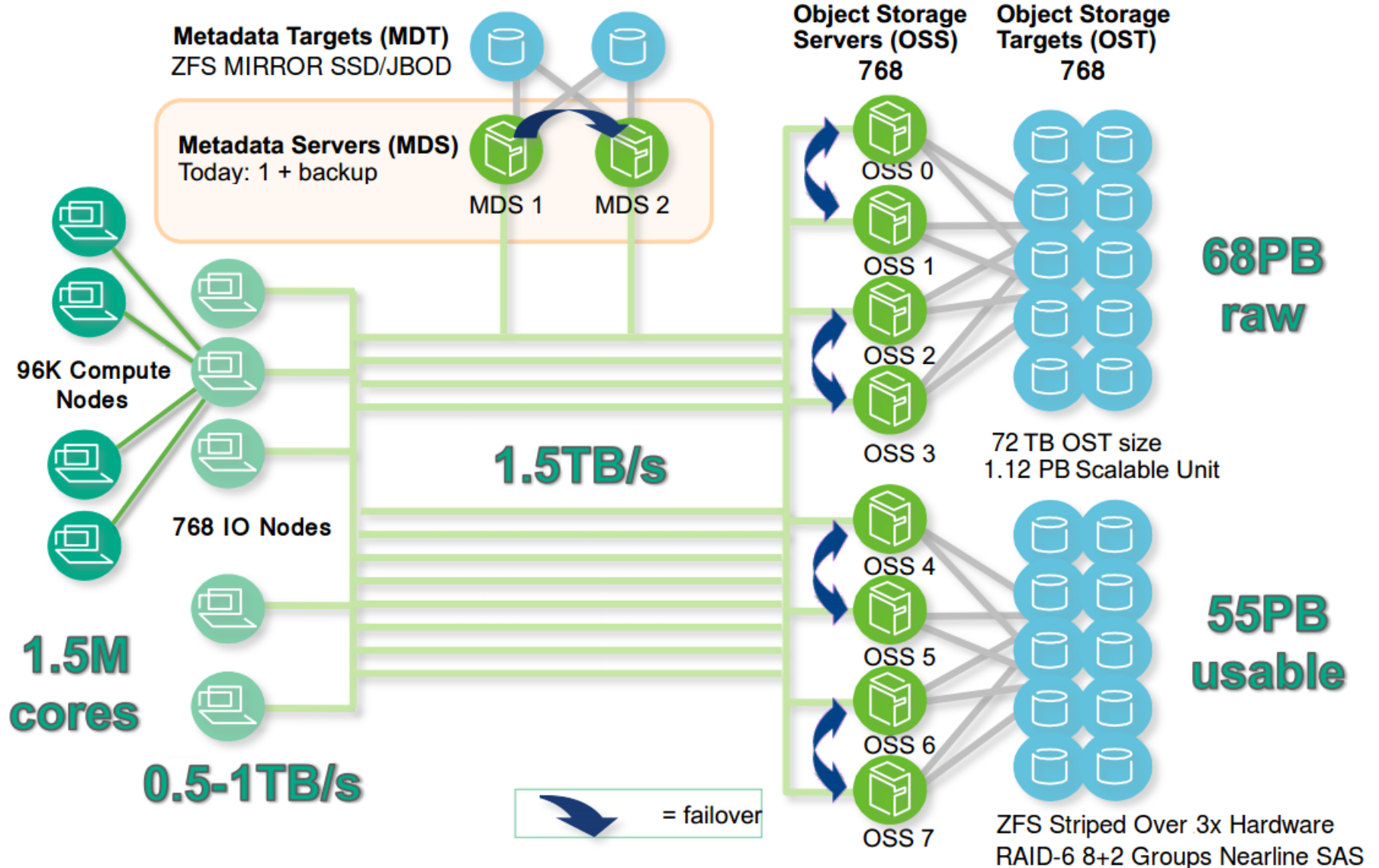


Rack Layout

- 8 E5400s per RSSU (Rack Storage Scalable Unit)
- 16 OSS nodes at the top of the rack
- 48 Racks total: 384 E5400s, 768 OSS nodes
- 55PB Capacity, aiming for 1TB/s



LLNL Sequoia Lustre Architecture



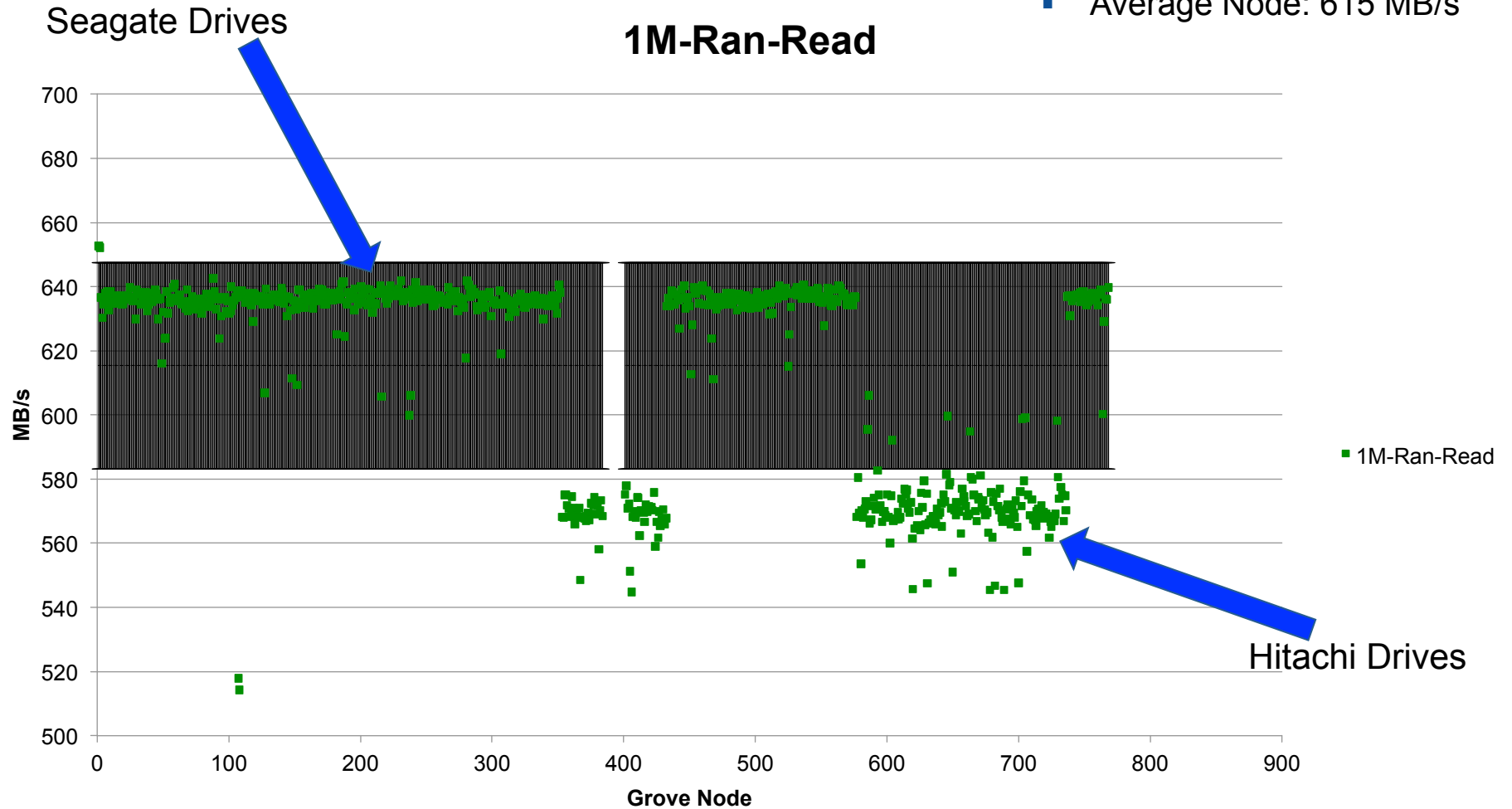
Acceptance Criteria

- **Stability**
 - 30 days with no loss of access to data
- **Integrity**
 - 30 days without occurrence of corruption
- **Performance**
 - Combination of XDD and ZPIOS
 - Couldn't use IOR to Lustre since the code was under development



Performance – XDD

- Projected 472 GB/s
- Average Node: 615 MB/s

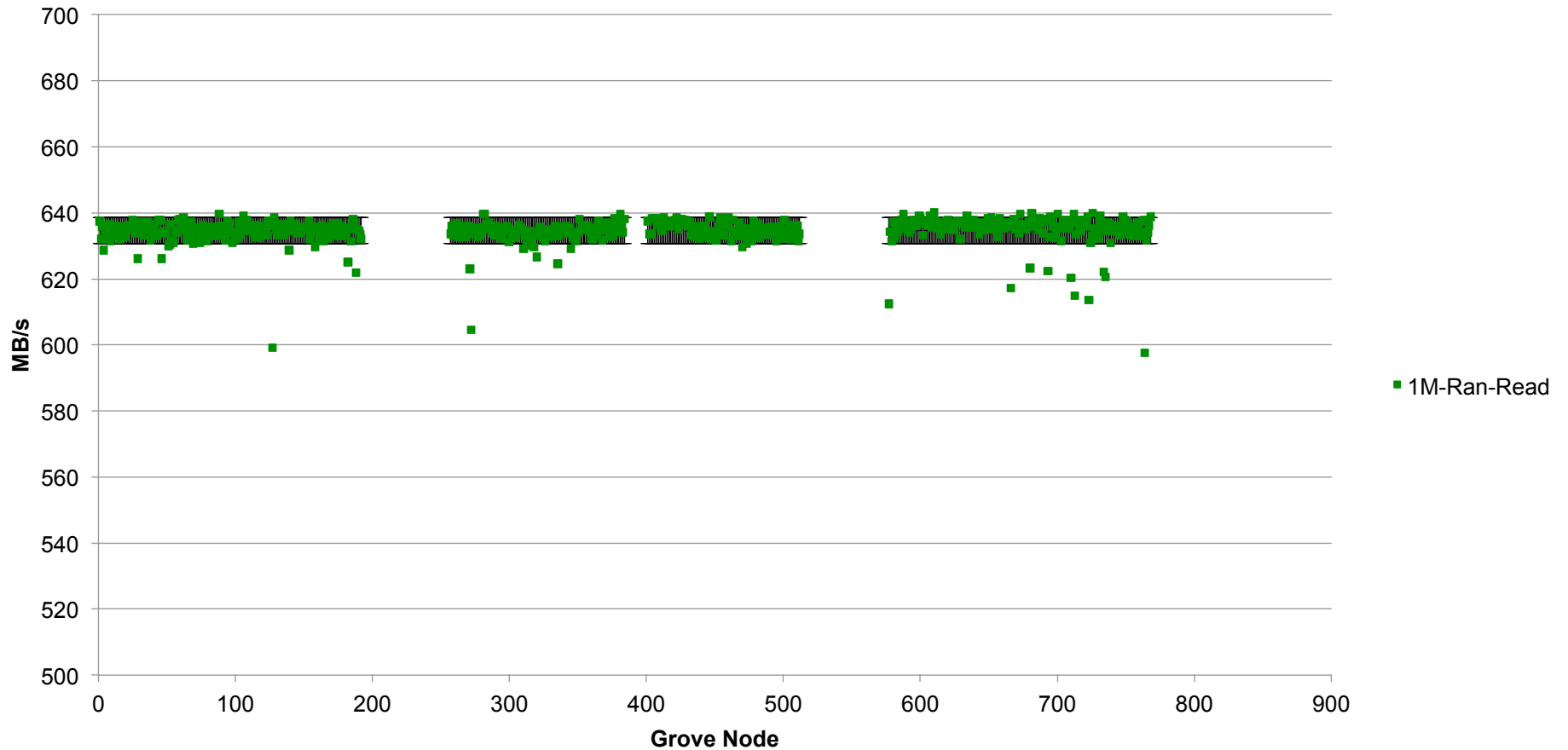


Performance – XDD

Post Drive Swap

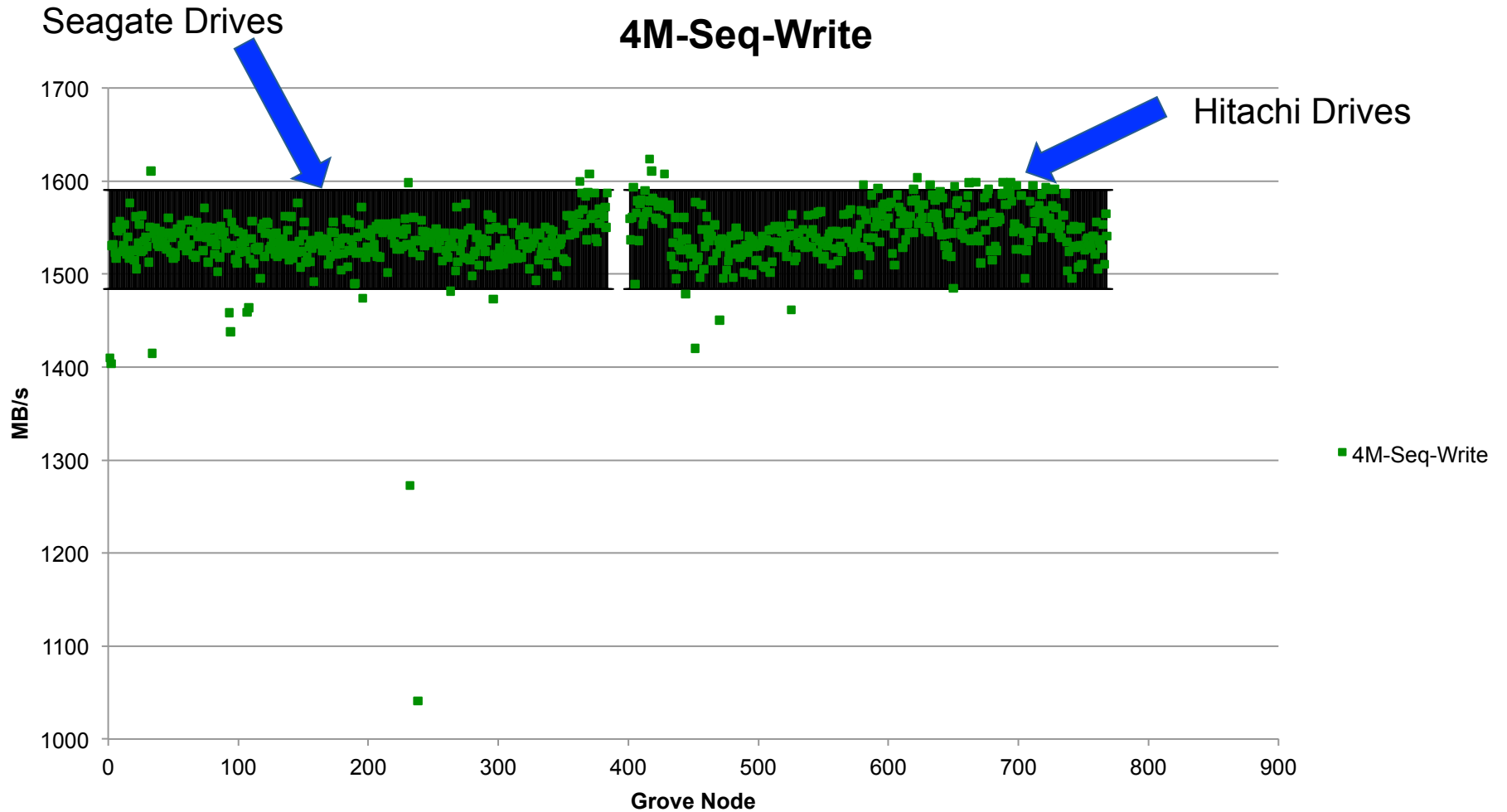
- Projected 487 GB/s
- Average Node: 635 MB/s

1M-Ran-Read



Performance – XDD

- Projected 1.184 TB/s
- Average Node: 1.542 GB/s

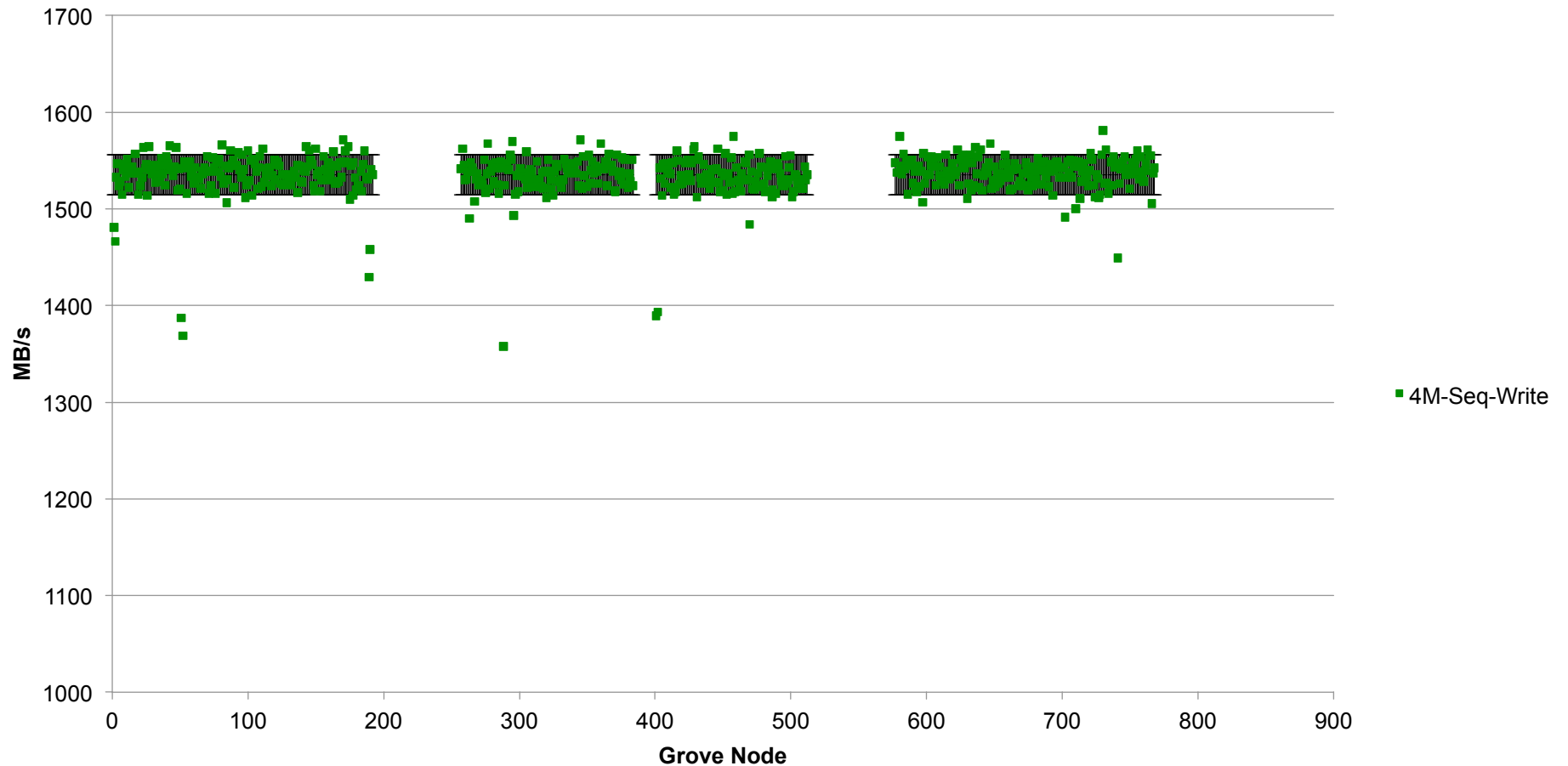


Performance – XDD

Post Drive Swap

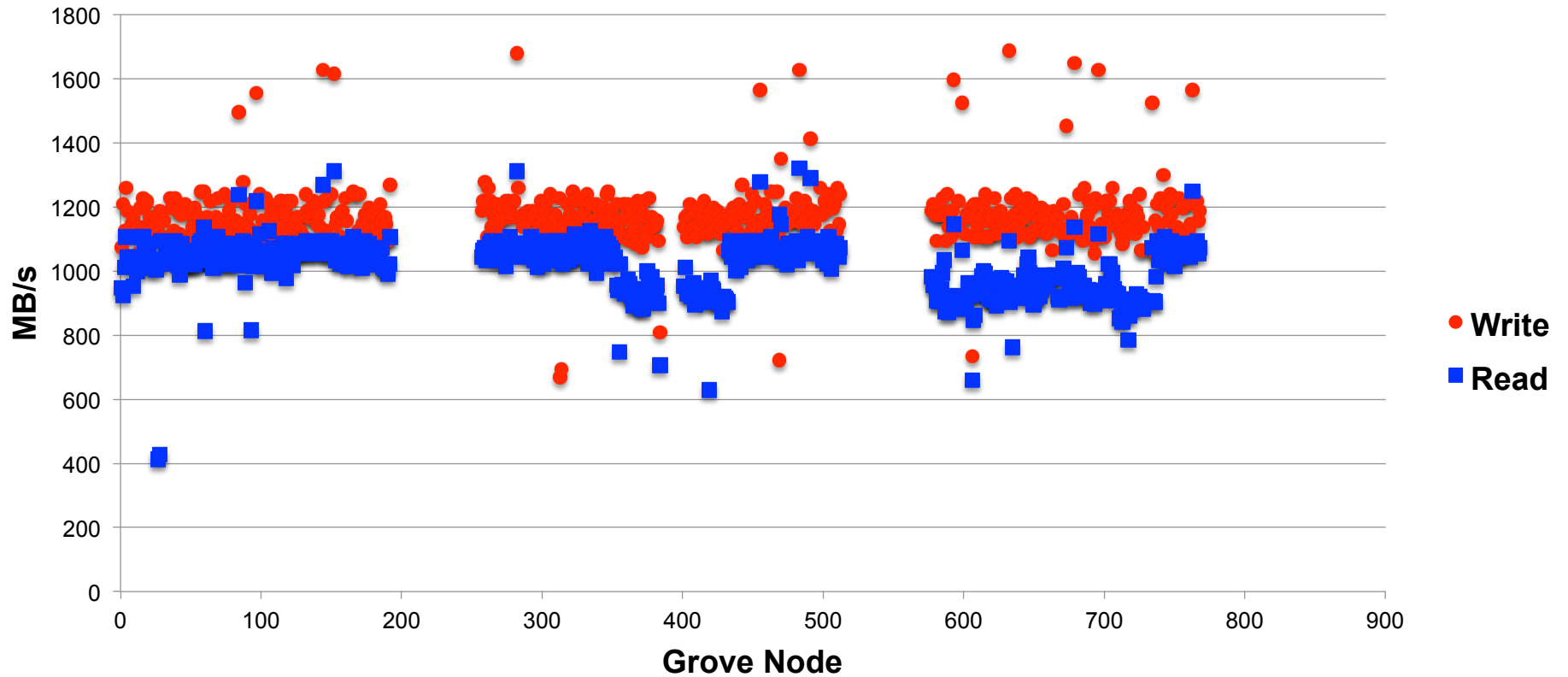
- Projected 1.178 TB/s
- Average Node: 1.535 GB/s

4M-Seq-Write



Performance – ZPIOS

ZPIOS Results 3/13/2012

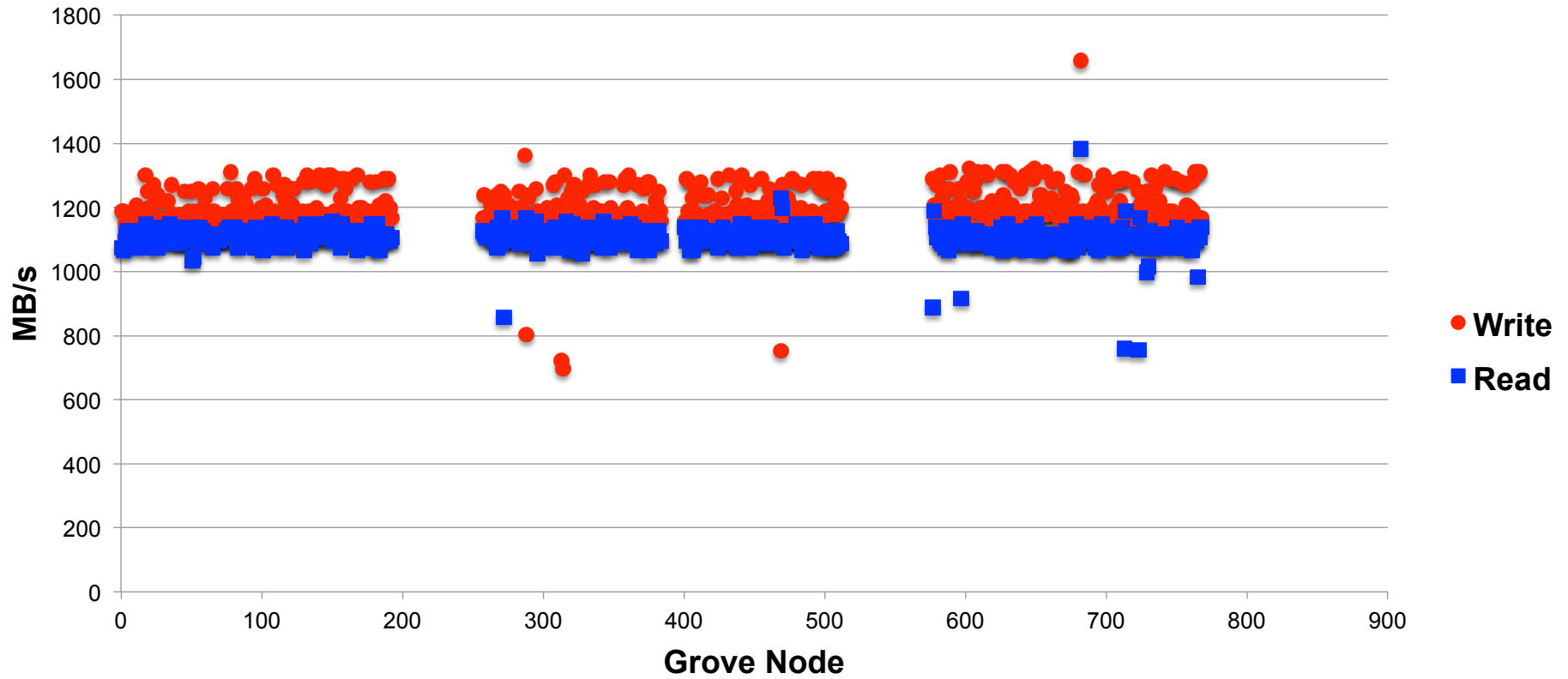


- Aggregate: Write 693 GB/s, Read 603 GB/s (616 nodes)
- Projected: Write 864 GB/s, Read 752 GB/s (768 nodes)
- Average Node: Write 1,125 MB/s, Read 979 MB/s

Performance – ZPIOS

Post Drive Swap

ZPIOS Results 4/16/2012

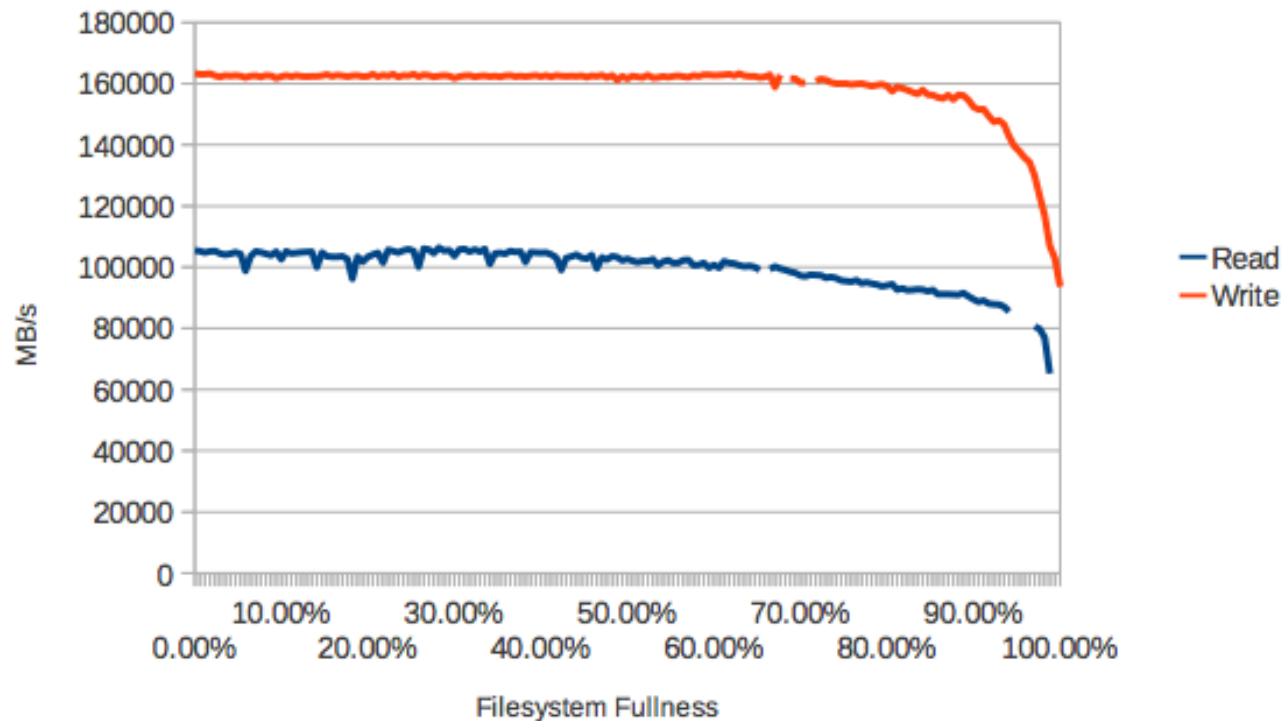


- Aggregate: Write 747 GB/s, Read 689 GB/s (624 nodes)
- Projected: Write 919 GB/s, Read 848 GB/s (768 nodes)
- Average Node: Write 1197 MB/s, Read 1104 MB/s

ZFS In Action

I/O Performance vs Fullness

Stonewalling IOR File per Process with 64 RBODs (128 OSS/OSTs)



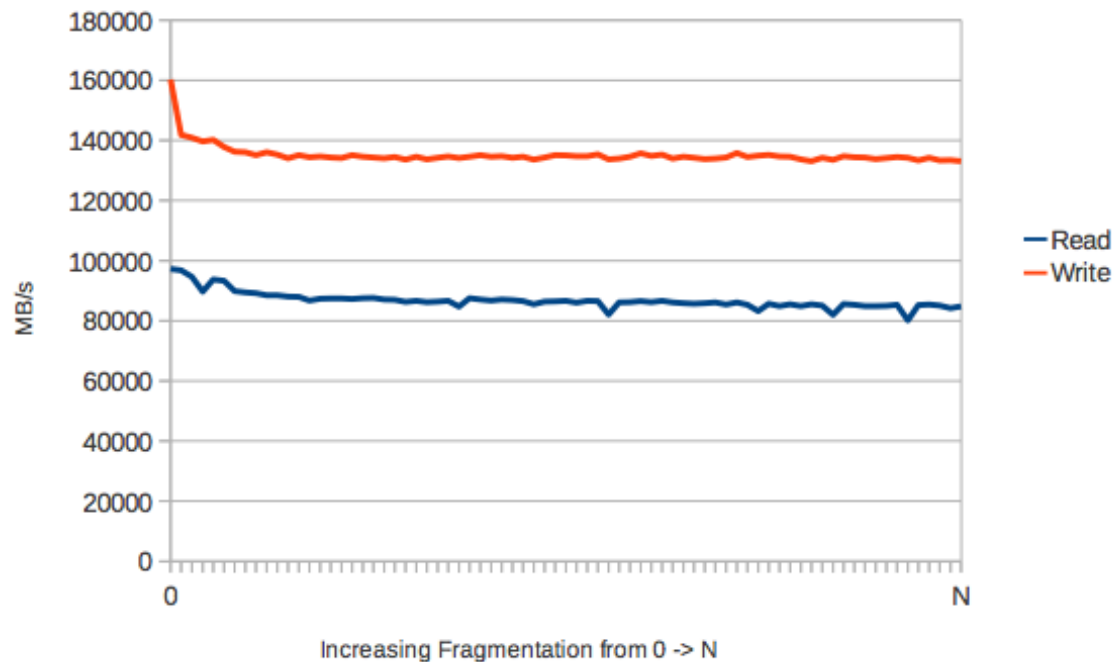
NOTES:

- 1) Started with pristine newly formatted file system
- 2) Files system was filled by running consecutive stonewalling IORs and leaving the files
- 3) Write performance measured by 5 minute stonewalling IOR write phase
- 4) Read performance measured by 2 minute stonewalling IOR read phase
- 5) Minimal fragmentation expected

ZFS In Action

I/O Performance vs Fragmentation

Stonewalling IOR File per Process with 64 RBODs (128 OSS/OSTs)



NOTES:

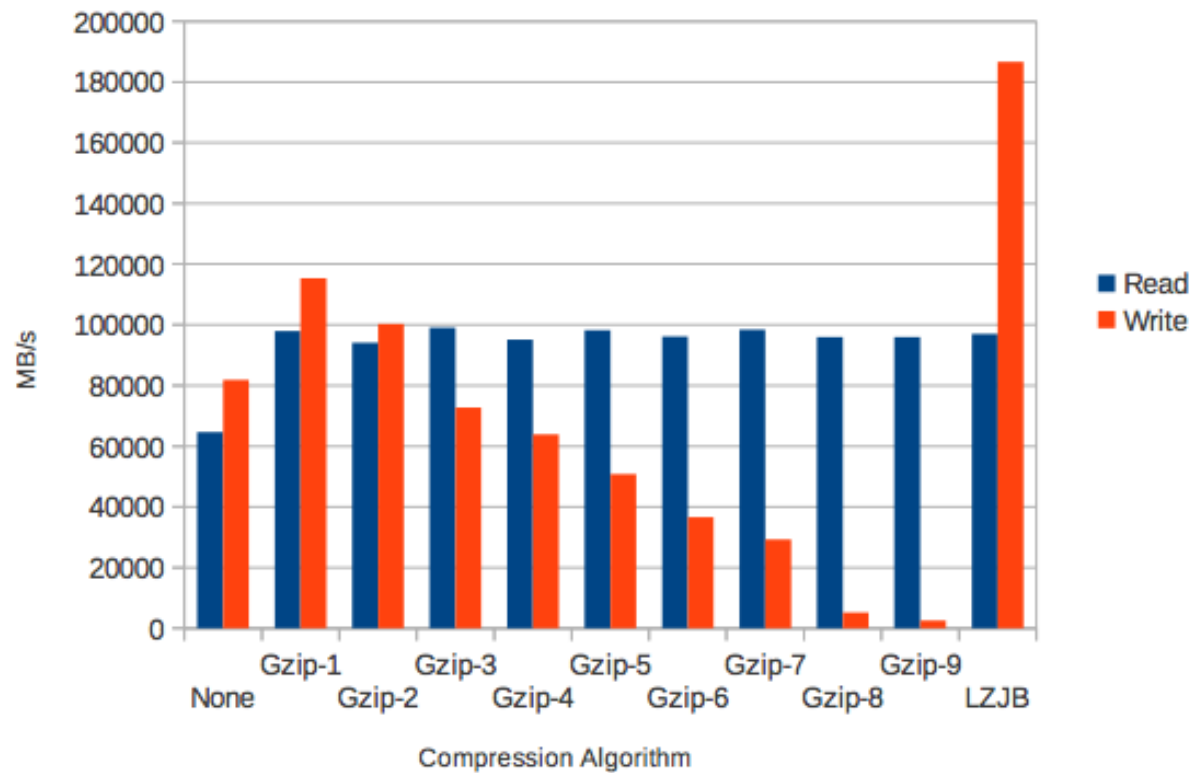
- 1) Started with pristine newly formatted file system
- 2) Files system was filled by running consecutive stonewalling IORs and leaving the files
- 3) Files were randomly removed until the total file system fullness was 70%
- 4a) Each run consists of a stonewalling IOR which creates N files
- 4b) After each run N files are randomly removed from all IOR runs keeping the fullness at 70%
- 5) Write performance measured by 5 minute stonewalling IOR write phase
- 6) Read performance measured by 2 minute stonewalling IOR read phase
- 7) Fragmentation is expected to increase over subsequent runs

ZFS In Action

Compression

I/O Performance with Compression

Stonewalling IOR File per Process with 32 RBODs (64 OSS/OSTs)



Status Update

■ **Functionality**

- Critical Components Complete
 - OSD API
 - ldiskfs OSD
 - zfs OSD
 - LLOG restructuring
 - MGS/MDT/OST over OSD
 - Patchless ZFS Servers
 - Quota over OSD
 - Changelog over OSD
- Remaining
 - ZIL Integration
 - Linux Drive Management

■ **Stability**

- Passes most of the Lustre Test Suite
- Stable under moderate test loads
 - Fixed Memory pressure issues
 - Fixed Lustre and ZFS deadlocks
 - Fixed issues with recovery and failover
- Many weeks without a server panic
- Running our SWL 24/7 to uncover issues
- Working to stabilize Sequoia client
- Troubleshooting IB issues on Sequoia

Status Update

- Two File Systems
 - Split Grove, the file system cluster, into two halves, development and production
 - 24 rack production system, 27PB, 384 OSTs
 - 24 rack development system, (8 racks file system, 16 racks clients today)
 - Easily reconfigured to test various numbers of clients/servers
 - Can update the development code quickly for testing new features/patches
 - When development branch is stable, roll out to production half
 - Will expand production half when development is complete
- Contract with Whamcloud/Intel
 - Development scheduled to be complete in September 2012
 - Available in the Lustre 2.4 release (March 2013)
 - We will run the Lustre Master Branch until 2.4 is GA

Questions?



Website

<http://zfsonlinux.org>

<http://zfsonlinux.org/lustre-configure-single.html>

Mailing Lists

zfs-discuss@zfsonlinux.org

zfs-devel@zfsonlinux.org

Marc Stearman

Parallel File Systems Operations Lead

stearman2@llnl.gov