Lustre on ZFS

At The University of Wisconsin

Space Science and Engineering Center

Scott Nolin
September 17, 2013

SSEC

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Why use ZFS for Lustre?

- The University of Wisconsin Space Science and Engineering Center (SSEC) is engaged in atmospheric research, with a focus on satellite remote sensing which has large data needs. As storage has grown, improved data integrity has become increasingly critical.

- The **data integrity features of ZFS** make it an important candidate for many SSEC systems, especially an ongoing satellite data disk archive project. For some background information see:
  - Zhang, Rajimwale, A. Arpaci-Dusseau, and R. Arpaci-Dusseau - *End-to-end Data Integrity for File Systems: A ZFS Case Study* http://research.cs.wisc.edu/wind/Publications/zfs-corruption-fast10.pdf
  - Bernd Panzer-Steindel, CERN/IT - *Data Integrity* http://indico.cern.ch/getFile.py/access?contribId=3&sessionId=0&resId=1&materialId=paper&confId=13797

- Additional ZFS features such as compression and snapshots are also very attractive.

- We have been following the ZFS on Linux project (http://zfsonlinux.org) at Lawrence Livermore National Labs (LLNL). This active project ported ZFS to linux for use as the 55PB filesystem on the supercomputer 'Sequoia'. We feel it is clear that ZFS on Lustre has matured, especially considering the production use on Sequoia.

Why a Test System?

- Proof of concept and design validation
- ZFS as a backend to Lustre is fairly new, released in 2.4
  - Documentation on setup and routine administration is not available in the Lustre manual or man pages yet.
- For the best data integrity we need systems supporting direct JBOD access to disks.
  - Finding appropriate hardware is a challenge.

SSEC Test System: Cove

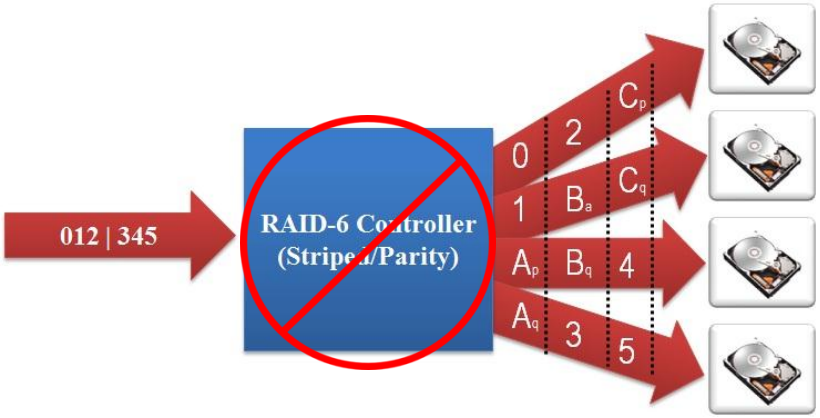
- Grant from Dell to provide testing equipment.
- No high-availability features included in test
  - For many SSEC systems, we have a “5x9 next business day” support level.
  - HA will be important for some future operational systems
- SSEC built the system and tested for performance, data integrity features, and routine administration.
- System design largely based on LUG 2012 “Lustre for Sequoia” presentation by Brian Behlendorf.
  - This allows a reasonable performance comparison to a known system.
  - The Sequoia file system also provides some insight into massive scaling that we cannot test.

Sequoia / Cove Comparison

Sequoia File System *	Cove File System
768 OSS, 68 PB raw	2 OSS, 240 TB raw
High Availability Configuration	Not High Availability
60 disks in 4U per OSS pair	60 Disks in 10U per OSS pair
3TB Near line SAS disks	4TB Near line SAS disks
MDT: Quantity 40 - 1TB OCZ Talos 2 SSDs	MDT: Quantity 4 - 400GB Toshiba MK4001GRZB SSD
OST: • IB Host attached • Dual Raid Controllers • RAID-6 by controller	OST: • Direct SAS attached • JBOD mode • RAID-Z2 by ZFS

** Based on LUG 2012 "Lustre for Sequoia" presentation, Brian Behlendorf*

OST: RAID-6 is not the best answer.



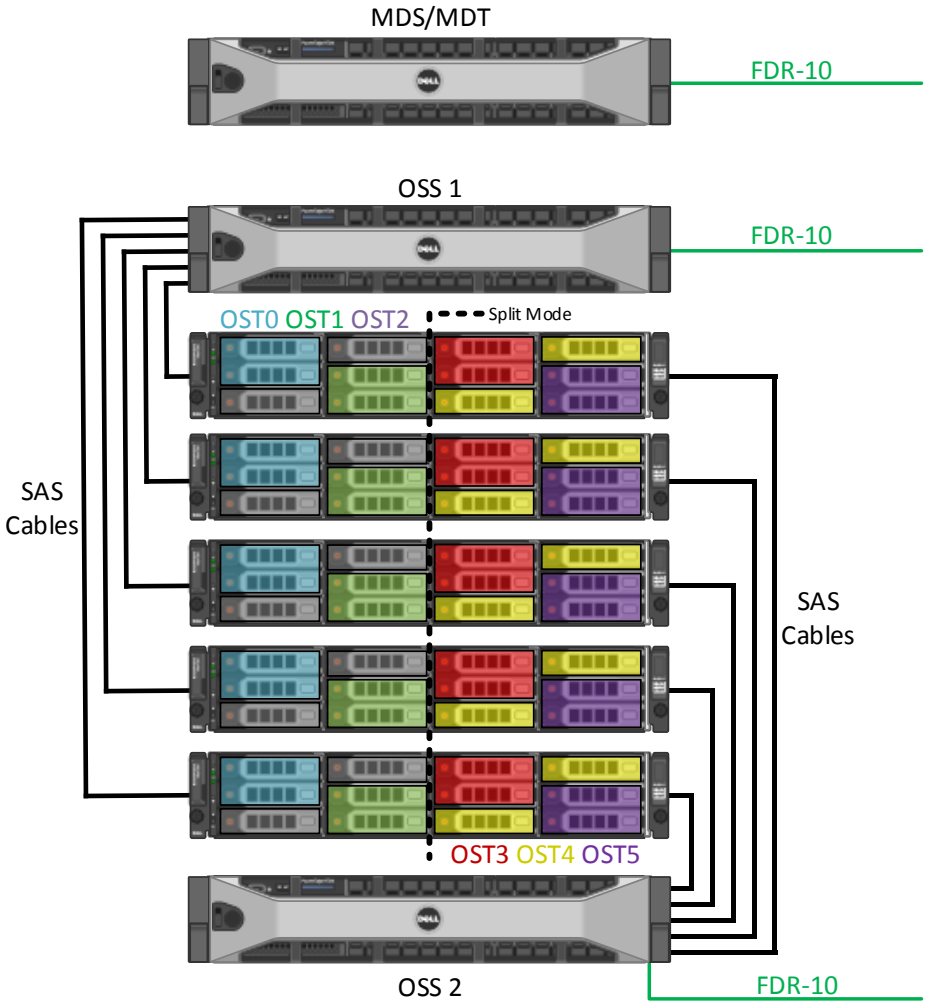
- For the best data integrity, giving ZFS direct disk access is ideal.
  - Cove uses SAS HBA to JBOD.
  - Create a **ZFS RAID-Z2**, which has 2 parity disks like RAID-6
- For our Cove design, this also allows us to stripe the RAID-Z2 across all enclosures, 2 disks per enclosure.
  - So you can lose an entire enclosure and keep operating.

Sequoia: Why No RAID-Z2?

- Brian Behlendorf at Lawrence Livermore National Laboratory (LLNL) explains:
  - LLNL needed something which could be used for ZFS or Ldiskfs.
  - Risk mitigation strategy since this was going to be the first Lustre on ZFS system.
  - Since then have developed confidence in the ZFS implementation and expect future deployments to be JBOD based.

Cove Hardware Configuration

- 3 computers:
  - 1 MDS/MDT
  - 2 OSS
- 5 OST: MD1200 storage arrays
  - **Split Mode** – MD1200 presents itself as 2 independent 6 disk SAS devices
  - Each “half” MD1200 direct SAS attached to OSS via SAS port on LSI 9200-8e adapter
- 1 OST = 10 disk RAIDZ2, 2 disks from each MD1200
- **This configuration can lose an entire MD1200 enclosure and keep operating**
- Disk and OST to OSS ratio same as Sequoia file system



SSEC

WISCONSIN UNIVERSITY OF WISCONSIN-MADISON

Sequoia / Cove MDS-MDT in Detail

MDS/MDT

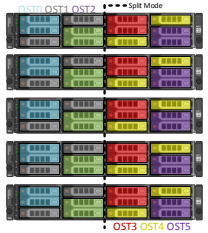
Sequoia	Cove
2 Supermicro X8DTH (HA pair)	1 Dell R720
Xeon 5650, Dual Socket 6 core @ 2.47 GHz	Xeon E5-2643, Dual Socket 8 core @ 3.30GHz
192GB RAM	128GB RAM
QDR Mellanox ConnectX-3 IB	FDR-10 Mellanox ConnectX-3 IB
Dual Port LSI SAS, 6Gbps	Dell H310 SAS, 6Gbps • Internal LSI SAS2008 controller
External JBOD	Internal JBOD (Dell H310 passthrough mode)
Quantity 40 - 1TB OCZ Talos 2 SSDs	Quantity 4 - 400GB Toshiba MK4001GRZB SSD
ZFS Mirror for MDT	ZFS Mirror for MDT

Sequoia / Cove OSS Unit in Detail

OSS

Sequoia	SSEC
Appro Greenblade – Quantity 2 • Sequoia has 378 of these units, for 756 total	Dell R720 – Quantity 2
Intel Xeon E5-2670, Dual Socket, 8 Core @ 2.60GHz	Intel Xeon E5-2670, Dual Socket, 8 Core @ 2.60GHz
64GB RAM	64GB RAM
QDR Mellanox ConnectX-3 IB	FDR-10 Mellanox ConnectX-3 IB
Storage Enclosure Connection • Dual QDR ConnectX-2 IB • HA Pairs	**Storage Enclosure Connection** • LSI SAS 9200-8e (8 ports) • No HA

Sequoia / Cove OST Unit in Detail



Sequoia	Cove
NetApp E5400	Dell MD1200
60 Bay, 4U	12 Bay, 2U x 5 = 60 Bay, 10U
3TB Near line SAS	4TB Near line SAS
IB host attached	SAS host attached
180 TB Raw	240 TB Raw
Volume Management • Dual Raid Controllers • RAID-6 by controller	Volume Management • Direct SAS, JBOD mode • RAID-Z2 by ZFS

SSEC

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Cove Software Configuration

- **Servers**
  - RHEL 6
  - Lustre 2.4 for ZFS
  - ZFS 0.61
  - Lustre 2.3 for ldiskfs (2.4 not available at test time)
  - Mellanox OFED Infiniband Stack

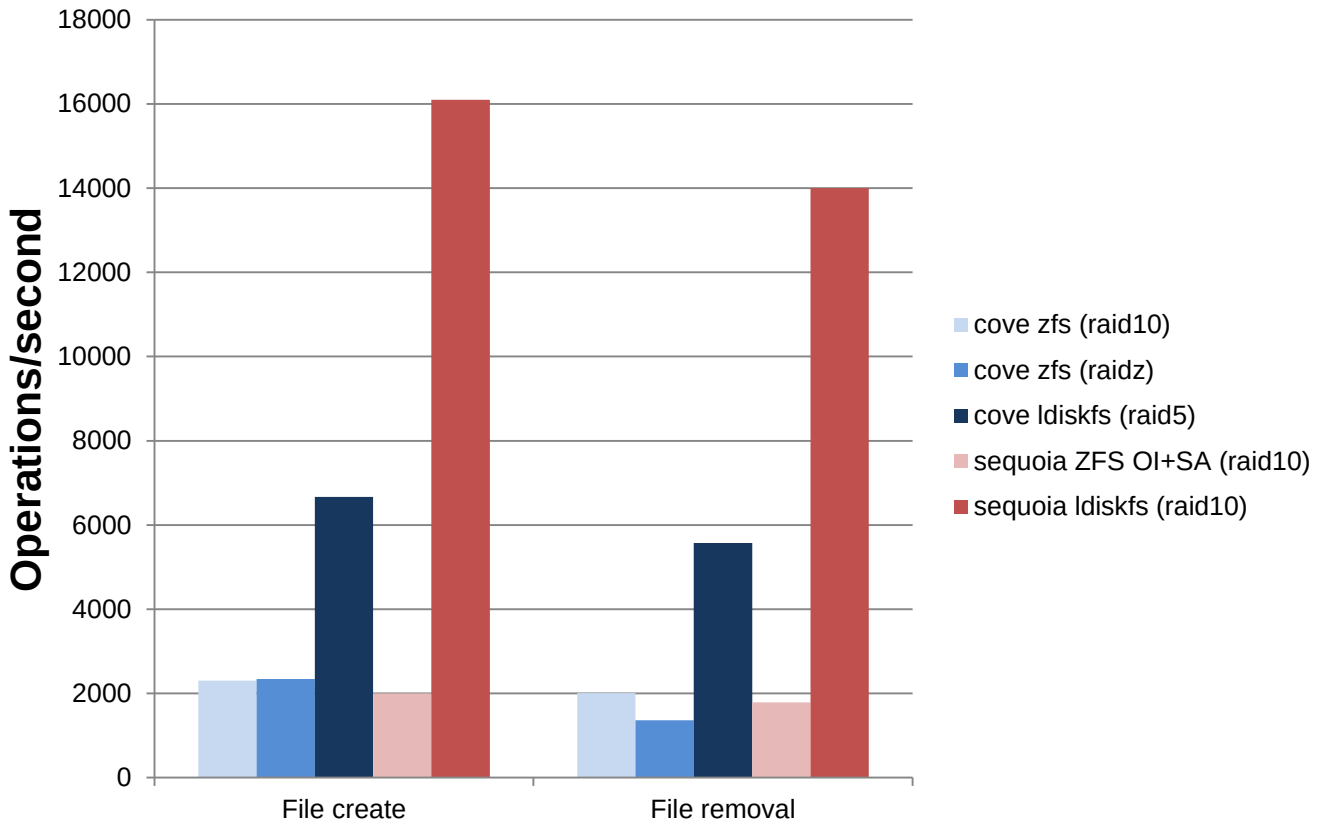
- **Test Clients**
  - 4 - 16 core nodes
  - RHEL 5
  - Lustre 2.15
  - Mellanox OFED Infiniband Stack

Metadata Testing: MDTEST

Sequoia MDTEST *	Cove MDTEST
1. ldiskfs – 40 disk ,mirror 2. ZFS mirror	1. ldiskfs – 4 disk RAID-5 2. ZFS mirror 3. RAID-Z
1,000,000 files Single directory	640,000 files Single directory
52 clients	4 clients, 64 threads total

** Based on LUG 2012 "Lustre for Sequoia" presentation, Brian Behlendorf*

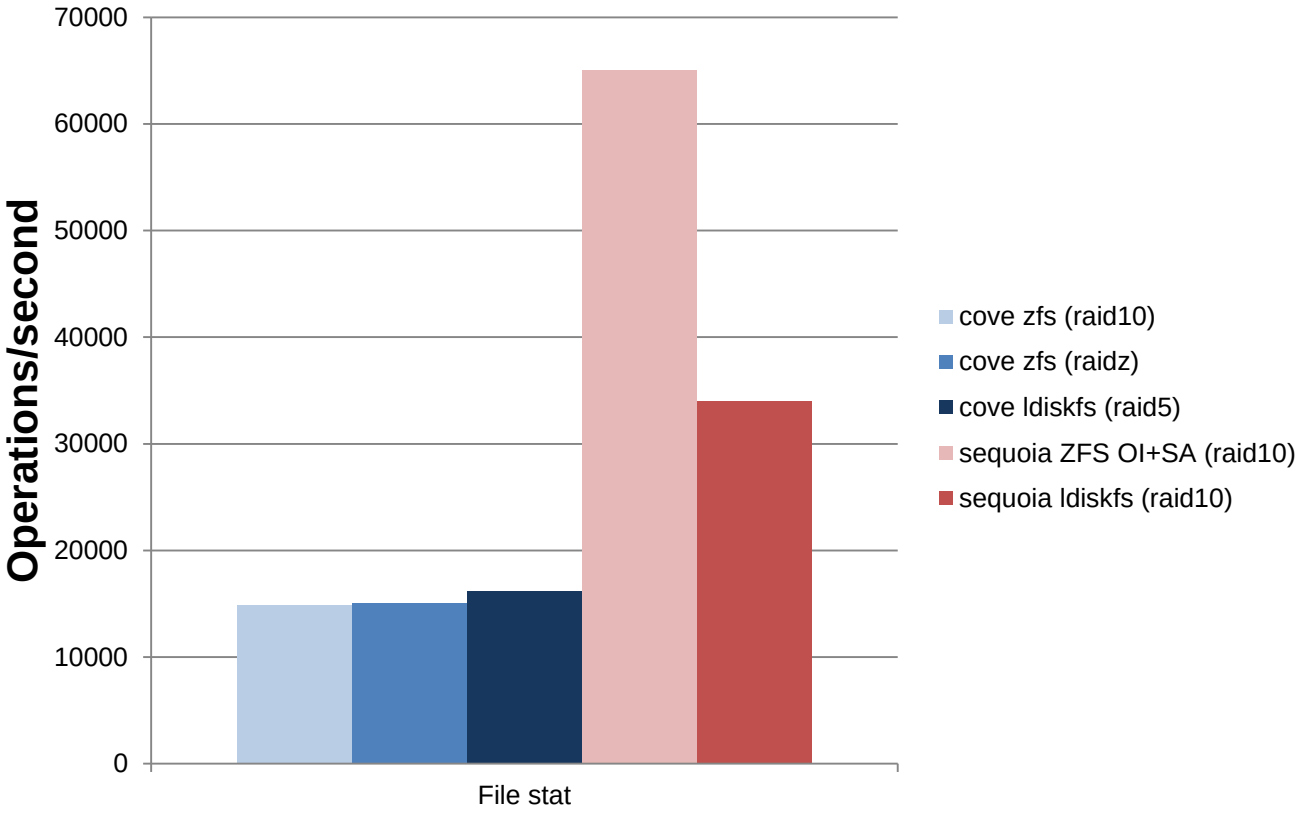
MDTEST: File Create and Remove



SSEC

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

MDTEST: File Stat



SSEC

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

MDTEST Conclusion

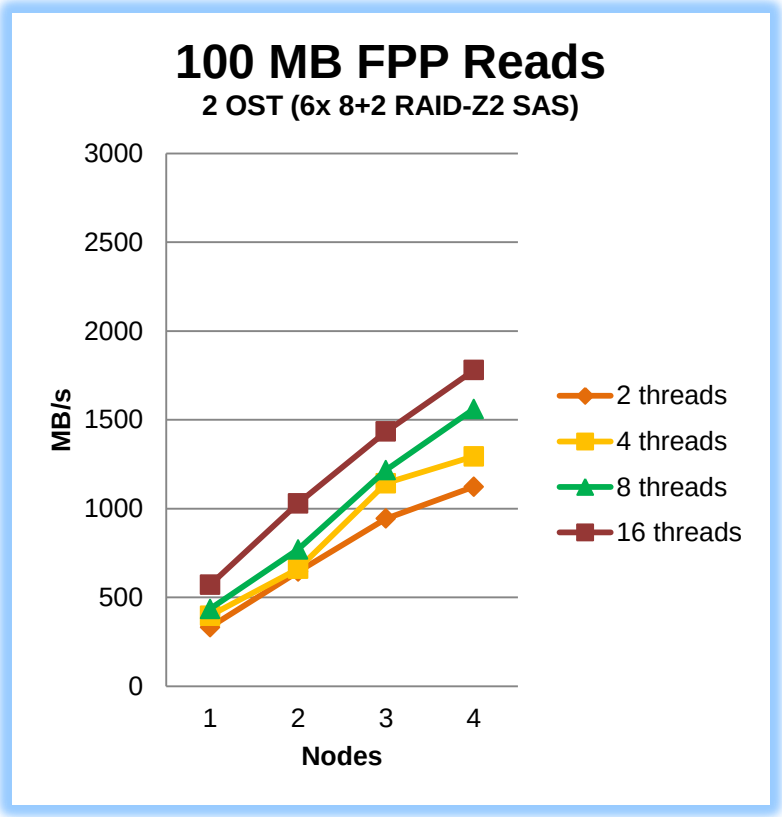
- Cove performance seems consistent with Sequoia performance for MDTEST. File stat is much lower than Sequoia's performance, we think this is due to 40 SSD's vs 4.
- The metadata performance of lustre with ZFS is acceptable for many SSEC needs.

File System IO Performance: IOR

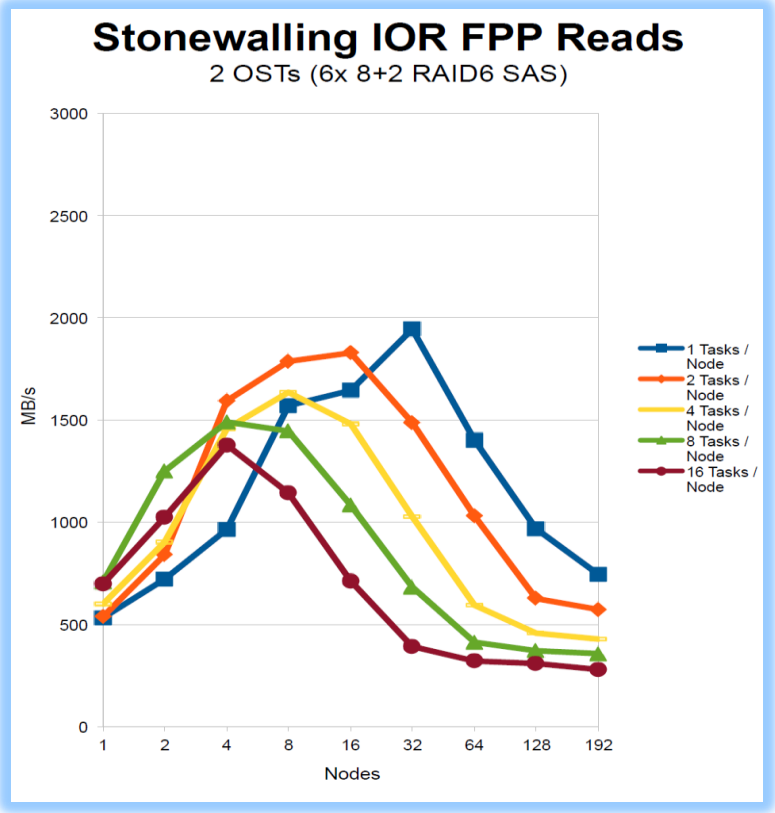
Sequoia IOR Benchmark*	Cove IOR benchmark
2 OSS (1/384th scale)	2 OSS
6 OST: 8+2 RAID-6 SAS	6 OST: 8+2 RAID-Z2 SAS
FPP – One File Per Process	FPP – One File Per Process
1, 2, 4, 8, 16 tasks per node	2, 4, 8, 16 tasks per node
1 to 192 nodes	1 to 4 nodes
Stonewalling	Not Stonewalling
Lustre 2.3	Lustre 2.4
Unknown file size	Showing 100MB file results. (Tested 1MB, 20MB, 100MB, 200MB, 500MB)

* From Andreas Dilger LAD 2012 - “Lustre on ZFS”

IOR Read



Cove

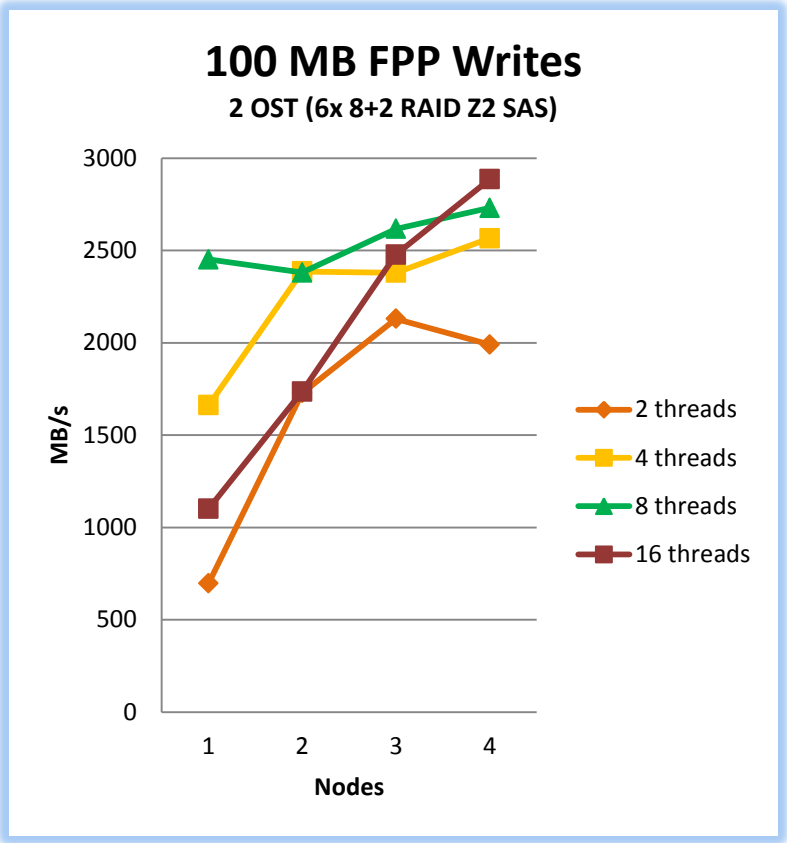


Sequoia

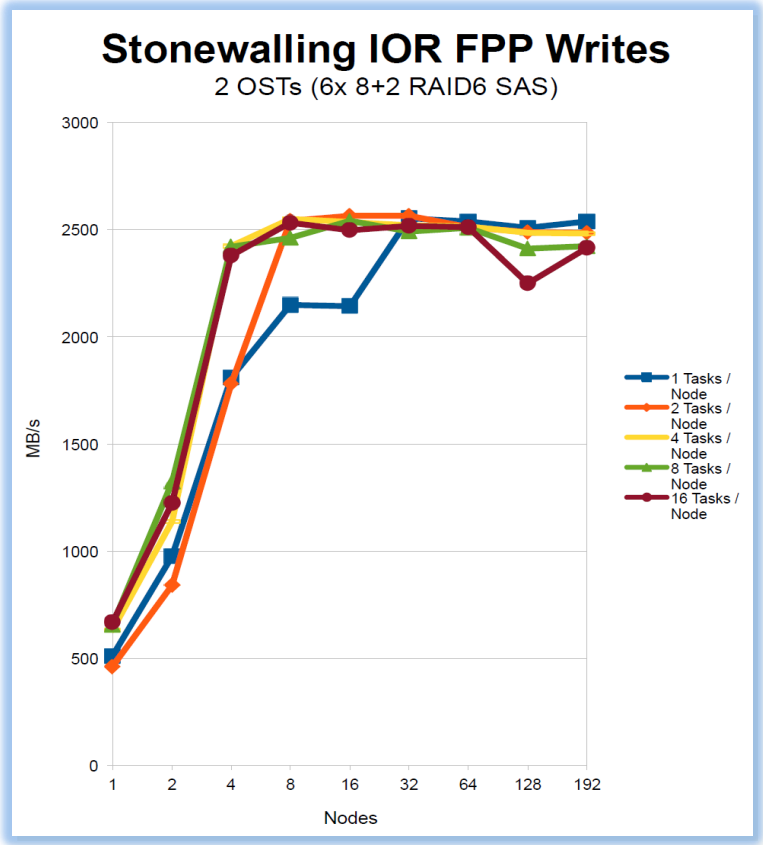
SSEC

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

IOR Write



Cove



Sequoia

SSEC

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

IOR Conclusion

- Cove performance compares well with Sequoia filesystem
- We assume it would scale similarly to Sequoia.
- SSEC will soon have opportunity to test at larger scale.

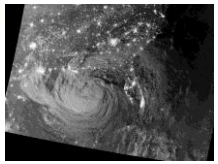
Data Integrity Tests: Routine Failures

- Tested routine things like replacing a single disk, systems going up and down, or an OST offline and back.
- Various “unplanned” tests as I tried things with hardware, moved things, recabled, made mistakes, etc.
- Striping across JBODs means we could test by powering off an entire JBOD enclosure and keep running.
- In all cases, no data was lost or corrupted.

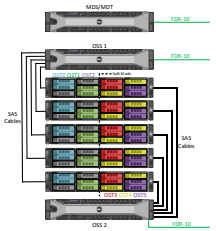
Data Integrity Tests: Corrupting a Disk

For SSEC this is the critical feature of ZFS. The random dd simulates corruption that can happen for a wide variety of reasons, and with most filesystems there is no protection.

No data was lost or corrupted.



22TB VIIRS Data
- With MD5 sums



Store on Cove

Scott Does Damage
- Use ‘dd’ to write random data to a disk.

ZFS Corrects
- Zpool scrub
- ✓ Verify MD5 sums

System Administration

- Giving ZFS direct access to drives is the best choice for ensuring data integrity, but this leads to system administration challenges.
  - ZFS is responsible for volume management, so functions typically provided by the RAID adapter are not available.
- ZFS snapshots are an attractive feature for metadata backups
- We tested various administration tasks, with reasonable solutions for most. Firmware updates are a concern.

System Administration Summary

System Administration Task	SAS HBA / ZFS (OSS/OST)	H310 passthrough mode / ZFS (MDS/MDT)	H810 RAID / ldiskfs (Traditional Method)
Drive Identification	• vdev_id.conf aliases by path • Sas2ircu (LSI command line tool) to enumerate drives	• vdev_id.conf aliases by path • Openmanage to enumerate drives	Dell Openmanage
Drive Firmware Update	• Install H810 • replace cables • flash with Dell Utilities	Dell Utilities	Dell Utilities
Enclosure Firmware Update	*assume same as Drive Methods, not tested	Dell Utilities	Dell Utilities
Drive Failure Notification	Use zpool status results	Dell Openmanage	Dell Openmanage
Predictive Drive Failure Notification	smartctl	Dell Openmanage	Dell Openmanage
Metadata Backup and Recovery	ZFS Snapshot	ZFS Snapshot	dd or tar (some versions of tar and lustre)

System Administration: Firmware

- Drive firmware
  - Dell Redhat utility identifies and will attempt firmware updates, but fails to due to check of “logical drive status”. It expects that to be provided by the Dell H810
  - Bring system down, add H810, re-cable to H810, flash firmware, re-cable to original.
    - Works, doesn’t do anything bad like write a UUID
    - Awkward and slow due to re-cabling
    - Does not scale well
- Enclosure firmware
  - We did not test this, but assume it is a similar situation.

System Administration: Metadata Backup

- ZFS snapshots
  - These are object type backups.
  - No “file-based” backup currently possible
  - Snapshots are very nice for MDT backup.
  - Analogous to the “dd” option required for some lustre versions, but with incrementals and a lot more flexibilty. This is a bonus of zfs.
  - Like many ZFS features, can be useful for other tasks, but metadata backup is most critical for SSEC.

System Administration Conclusion

- While giving ZFS direct access to drives in a JBOD is ideal for data integrity, it introduces some system administration challenges.
  - Vendor tools are focused on intelligent RAID controllers, not direct access JBOD.


Most missing pieces can be directly replaced now with linux tools.


Firmware updates and utilities need more work.

- ZFS snapshots are very convenient for metadata backup and restore.

Next Steps

- Production disk archive system
  - Testing at larger scale (12 OSS, more clients)
  - HA Metadata servers
    - Will try ZFS on infiniband SRP mirrored LUNS
      - Based on Charles Taylor's "High Availability Lustre Using SRP-mirrored LUNs" LUG 2012
- System administration work
  - Pursue firmware utility improvements with Dell
  - Setup, administration, monitoring documentation
    - Will be made public by end of 2013

Thanks to:

- Brian Behlendorf at LLNL
- Jesse Stroik and John Lalande at UW SSEC
- Pat Meyers and Andrew Waxenberg at Dell
- Dell HPC