ZFS/Lustre History

- 2007
  - Livermore raises ldiskfs scalability/performance concerns
    - Fsck, filesystem size, random IO, data integrity, etc
  - Alternate backend is needed for large lustre filesystems
  - ZFS identified as technically the best solution
    - Addresses all known ldiskfs limitations
    - Proven production quality implementation
    - Licensing concerns can be addressed
    - Must be ported to Linux
  - CFS/Sun start ZFS/Lustre user space implementation
ZFS/Lustre History

- 2008
  - Livermore starts porting ZFS to the kernel
    - Intended to determine viability of a kernel port
    - No unsurmountable technical issues discovered
    - Initial performance results are encouraging
  - Sun Lustre-osd development
    - Shift in strategy, the Livermore kernel port is adopted
    - Brian joins the Sun Lustre-osd development team
    - Continued Lustre-osd development
  - Licensing concerns unresolved... work continues...
ZFS/Lustre History

- 2009
  - Livermore ZFS development
    - Focus on a production quality ZFS port
    - Built quarter scale prototype ZFS/Lustre filesystem
  - Sun/Oracle Lustre-osd development
    - Oracle acquires Sun
    - Lustre-osd development continues unchanged
    - Zerocopy, grants, large dnodes, quotas, utilities, etc
  - Licensing concerns unresolved... work continues...
ZFS/Lustre History

- 2010
  - Livermore ZFS development
    - Linux integration (utilities, udev, zevents, disk failures)
    - Built a full scale ZFS/Lustre filesystem
  - Oracle Lustre-osd development
    - Announced ZFS/Lustre only available for Solaris
    - Lustre-osd development continues on Linux
    - Oracle cancels Lustre... progress is delayed...
  - Licensing concerns unresolved... work continues at LLNL...
ZFS/Lustre History

- 2011
  - Livermore ZFS development
    - ZFS Posix Layer (ZPL) added
    - Lustre-osd development branch publicly available
  - Whamcloud Lustre-osd development
    - Contracted by Livermore to complete Lustre-osd
    - Most of the original Lustre-osd developers are at Whamcloud
  - Licensing concerns unresolved... work continues...

- Late 2011
  - Livermore plans a ZFS/Lustre filesystem for Sequoia
    - 50 PB capacity, 512 GB/s – 1 TB/s bandwidth
ZFS Overview

- Developed by Sun (now Oracle) on Solaris
- Combined filesystem, logical volume manager, RAID
- Copy-on-write
- Built-in data integrity
- Intelligent online scrubbing and resilvering
- Very large filesystem limits
LLNL's Reasons for porting ZFS

- Lustre servers currently use ext4 (ldiskfs)
  - Random writes bound by disk IOPS rate, not disk bandwidth
  - OST size limits
  - fsck time is unacceptable
  - Expensive hardware required to make disks reliable
- Late 2011 requirement:
  - 50PB, 512GB/s – 1 TB/s
  - At a price we can afford
- COW sequentializes random writes
  - No longer bound by drive IOPS
- Single volume size limit of 16 EiB
- Zero fsck time. On-line data integrity and error handling
- Expensive RAID controllers are unnecessary
Licensing Concerns

CDDL = Common Development and Distribution License
GPL = (Gnu) General Public License

CDDL

GPL

GPL

GPL

Lustre

ZFS

SPL

Linux Kernel

CDDL = Common Development and Distribution License
GPL = (Gnu) General Public License
Licensing Concerns

- **Distributing Source**
  - CDDL is an open source license
  - CDDL provides an explicit patent license
  - ZFS changes contributed as CDDL code
  - ZFS sources kept separate from all GPL code

- **Distributing Binaries**
  - Linux kernel allows non-GPL third party modules
    - Nvidia, ATI, etc...
  - Linus views the kernel module interface as LGPL
    - ZFS uses no GPL-only symbols
    - Included headers do not make a derived work
Licensing Concerns

- ZFS is NOT a derived work of Linux
  
  • “It would be rather preposterous to call the Andrew FileSystem a 'derived work' of Linux, for example, so I think it's perfectly OK to have a AFS module, for example.”
    - Linus Torvalds
  
  • “Our view is that just using structure definitions, typedefs, enumeration constants, macros with simple bodies, etc., is NOT enough to make a derivative work. It would take a substantial amount of code (coming from inline functions or macros with substantial bodies) to do that.”
    - Richard Stallman (The FSF's view)
Solaris Porting Layer
Linux/ZFS Glue

- ZFS
- SPL
- Linux Kernel
ZFS and Lustre Components

User

Kernel

Interface Layer

Transactional Object Layer

Pooled Storage Layer

ZPL
ZVOL
/dev/zfs
ZIL
ZAP
Traversal
DMU
DSL
ARC
ZIO
VDEV
Configuration

Lustre

MDT
OST
MDD
OFD
ZFS OSD

ZFS CLI
libzfs

Lawrence Livermore National Laboratory
Ported by LLNL

User

Kernel

Interface Layer

Transactional Object Layer

Pooled Storage Layer

ZFS CLI

libzfs

ZPL

ZVOL

/dev/zfs

ZIL

ZAP

Traversal

DMU

DSL

ARC

ZIO

VDEV

Configuration

MDT

OST

MDD

OFD

ZFS OSD

Lustre

Lawrence Livermore National Laboratory
CFS → Sun → Oracle → Whamcloud
ZFS/Lustre Prototype (Zeno)
OSS SSU (Zeno)

- Component
  - Bandwidth
  - QDR IB: 25.6 GB/s
  - Host SAS: 96.0 GB/s
  - JBOD SAS: 96.0 GB/s
  - Disk: 56.0 GB/s

- 896 TB / SSU
- 25.6 GB/s
- 70 2TB Disks / Host
  - 7 – 8+2 Raid-Z2 groups
  - 1 – 112 TB OST / Host
OSS SSU (Zeno3)

- 960 TB / SSU
- 38.4 GB/s
- 50 2TB Disks / Host
  - 5 – 8+2 Raid-Z2 groups
  - 1 - 80TB OST / Host

Component | Bandwidth
--- | ---
QDR IB | 38.4 GB/s
Host SAS | 38.4 GB/s
JBOD SAS | 96.0 GB/s
Disk | 60.0 GB/s
ZFS Performance Comparison

- Same number of drives
- SATA vs SAS disk
- RAID-Z2 vs RAID-6
- Write Performance is Limited by the ZFS Port
- Read Performance is Limited by Lustre/CPU
- ZFS is unoptimized, this can all be improved!
Single Node Write Performance

- Write performance is consistent with Lustre
- Lustre workload
  - Random 1MiB I/Os
  - 128 thrs to 4096 objs
- 60 MiB/s per disk for small pools (10 disks)
- Limited by taskq when scaled up
- This is fixable
Single Node Read Performance

- Read performance is significantly better than Lustre
- Lustre Workload
  - Random 1MiB I/Os
  - 128 thrs to 4096 objs
- Shows good scaling
- Prefetch disabled
- 50-60 MiB/s per disk even for large pools
- >90% CPU utilization when using 70 disks
- Can be optimized
More Information

- **ZFS & SPL**
  - [http://zfsonlinux.org](http://zfsonlinux.org)
    - Mailing Lists
      - zfs-announce@zfsonlinux.org
      - zfs-discuss@zfsonlinux.org
      - zfs-devel@zfsonlinux.org
    - Download software
    - Documentation

- **Lustre support for ZFS**
  - [http://zfsonlinux.org/lustre.html](http://zfsonlinux.org/lustre.html)

- **Licenses**
  - CDDL - [http://hub.opensolaris.org/bin/view/Main/licensing_faq](http://hub.opensolaris.org/bin/view/Main/licensing_faq)
  - GPLv2 - [http://www.gnu.org/licenses/gpl-2.0.html](http://www.gnu.org/licenses/gpl-2.0.html)
    - RMS - [http://lkml.indiana.edu/hypermail/linux/kernel/0301.1/0362.html](http://lkml.indiana.edu/hypermail/linux/kernel/0301.1/0362.html)