OpenZFS on Linux

HEPiX 2014

October 16, 2014

Brian Behlendorf
behlendorf1@llnl.gov

Lawrence Livermore National Laboratory

LLNL-PRES-XXXXXX
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

High Performance Computing

- Advanced Simulation
  - Massive Scale
  - Data Intensive
- Top 500 (June 2014)
  - #3 Sequoia
    - 20.1 Peak PFLOP/s
    - 1,572,864 cores
    - 55 PB of storage at 850 GB/s
  - #9 Vulcan
    - 5.0 Peak PFLOPS/s
    - 393,216 cores
    - 6.7 PB of storage at 106 GB/s

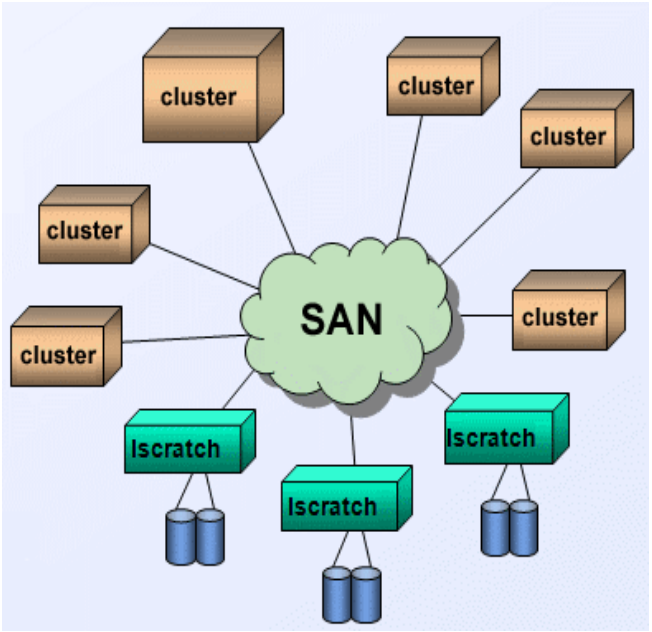
World class computing resources

Linux Clusters

- 2001 – First Linux cluster
- Today – ~10 large Linux clusters (100–3,000 nodes) and 10-20 smaller ones
- Near-commodity hardware
- Open source software stack (TOSS)
- Tri-Labratory Operating System Stack
  - Modified RHEL targeted for HPC
  - RHEL kernel optimized for clusters
  - Moab and SLURM for scheduling
  - Lustre parallel filesystem
  - Additional packages for monitoring, power/console, compilers, etc

Lustre

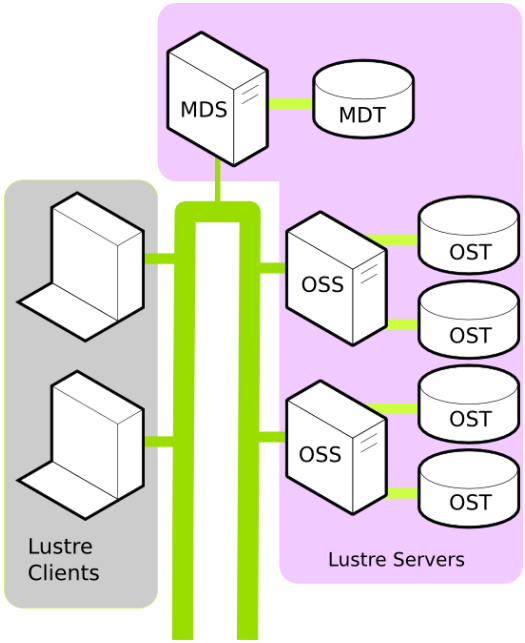
- Lustre is our parallel filesystem of choice
- Livermore Computing Filesystems:
  - Open – 20PB in 5 filesystems
  - Secure – 22PB in 3 filesystems
  - Plus the 55PB Sequoia filesystem
  - 1000+ storage servers
  - Billions of files
- Users want:
  - Access to their data from every cluster
  - Good performance
  - High availability
- How do we design a system to handle this?



Lustre is used extensively and pushed to its limits

OpenZFS and Lustre

- Traditionally Lustre uses ext4/ldiskfs
  - Good for small filesystems
  - Problematic for large filesystems
- Drawbacks to ext4/ldiskfs
  - No server scalability means lots of servers
  - No data integrity means expensive hardware
  - No online filesystem check means downtime
- Something better is needed for large systems
  - All of our Lustre filesystems have been migrated to use OpenZFS on Linux



ext4/ldiskfs did not meet our requirements, OpenZFS does

What is ZFS

- Pooled storage
  - Functionality of a filesystem and volume manager
  - Free space is shared by all filesystems
  - Designed to be scalable

- End-to-end data integrity
  - Detects and corrects silent data corruption
  - Resilient by design
  - Enables commodity storage solutions

What is ZFS

- Transactional object model
  - Always consistent on disk (no offline checking)
  - Universal building block (file, block, object, etc)

- Simple administration
  - Concise commands which do what you want
  - Online everything (scrubbing, resilvering, snapshots, clones, rollback, send / receive, etc...)

ZFS History

- 2001: development starts at Sun
- 2005: ZFS source code released
- 2008: ZFS released in FreeBSD 7.0
- 2010: Oracle stops contributing to released ZFS code
- 2010: illumos is founded as the successor to OpenSolaris
- 2011: ZFS released for Linux
- 2013: ZFS developers band together and form OpenZFS
- 2014: OpenZFS for Mac OS-X

What is OpenZFS?

OpenZFS is a community project founded by open source ZFS developers from multiple operating systems:

- The goals of the OpenZFS project are:
  - Raise awareness of the quality, utility, and availability
  - Encourage open communication about ongoing efforts
  - Ensure consistent reliability, functionality, and performance

Organizations / Companies

OpenZFS Features

- Feature Flags
  - Solves the on-disk format version problem.
  - Allows for independent development of on-disk features.
- Smoother more predictable IO performance
  - Since applications can dirty cache faster than it's written to storage, ZFS must delay some writes.
  - To prevent large delays small delays are injected in to the writes.
  - Helps reduces starvation and improves I/O fairness.
- LZ4 Compression
  - Improved performance and compression ratio.

More OpenZFS Features

- Performance
  - Asynchronous destroy
  - Single-copy ARC
  - L2ARC compression
  - Intelligent N-way mirrors
  - Smarter free space allocation
  - Faster block freeing
  - Faster xattrs
- Usability
  - zfs send progress reporting
  - zfs send/destroy size estimates
  - ZFS Bookmarks
  - Fragmentation metrics
  - POSIX ACLs
  - SELinux support
  - ZFS Event Daemon

http://open-zfs.org/wiki/Features

OpenZFS on Linux

- Current version is 0.6.3
- Near feature parity with other OpenZFS platforms
- Packages available for many distributions:

- Easy to install with DKMS or binary packages
- Large enthusiastic user community
- Mailing list: zfs-discuss@zfsonlinux.org
- Website: http://zfsonlinux.org

Development Model

- Project is hosted at Github
  - https://github.com/zfsonlinux/
  - 113 contributors, 289 forks, 1178 watchers
- Independent of the Linux kernel
  - Allows for supporting for a wide variety of Linux distributions and kernels
  - Allows updates independent of the kernel version
  - Simplifies code sharing with illumos and FreeBSD
  - Allows utilities and kernel modules to share code
- Issue Tracker
  - Features requests, bugs reports, and milestones
  - Everything is as open and as public as possible
  - Discussion in the issue tracker is encouraged

Development Model

- Pull requests
  - Used to submit proposed code changes
  - All patches are reviewed
  - All patches are automatically tested by the buildbot
    - Test results are published in the pull request
    - Only patches which pass the testing are merged
    - All merged patches are tested a second time
  - The master branch is ***always*** in a stable state
  - LLNL is the gatekeeper for the main repository
- Active community of developers
  - 74 different developers contributed to 0.6.3 release
  - Contributed Systemd, SELinux, POSIX ACL, ARM64

Next Steps

- Large Linux based JBOD storage systems
  - 10s-100s of disk per node, GB/s of bandwidth
  - Flexible management and monitoring functionality

- Lots of options
  - Scalable NFS / SMB file server
  - Distributed AFS, BeeFS/FhGFS, Ceph, GlusterFS, Lustre, PVFS filesystem
  - Desktop / Virtual machines
  - iSCSI block storage
  - Archival storage system

OpenZFS Features in Development

- Linux Page Cache Integration
  - Improved performance
  - Reduced memory fragmentation

- ZFS Event Daemon (ZED)
  - Flexible infrastructure for monitoring and management
  - Can provide automatic hot sparing, autoreplace, autoexpand, SMART integration, etc

- Persistent L2ARC
  - L2ARC is preserved over a pool export/import
  - Eliminates the L2ARC to warm up time

OpenZFS Features in Development

- Large block support
  - Today the maximum block size is 128K.
  - This can limit performance due to small disk I/Os.
  - Maximum block size increased to 1MB+.

- Large dnode support
  - Today dnodes are only 512 bytes
  - Prevents extended attributes from being stored on disk with the dnode which means extra IO.
  - Maximum dnode size increased to 16K

Questions


Lawrence Livermore
National Laboratory