

# Open ZFS

July 2013

Matt Ahrens

[mahrens@delphix.com](mailto:mahrens@delphix.com)

twitter: [@mahrens1](https://twitter.com/mahrens1)

---

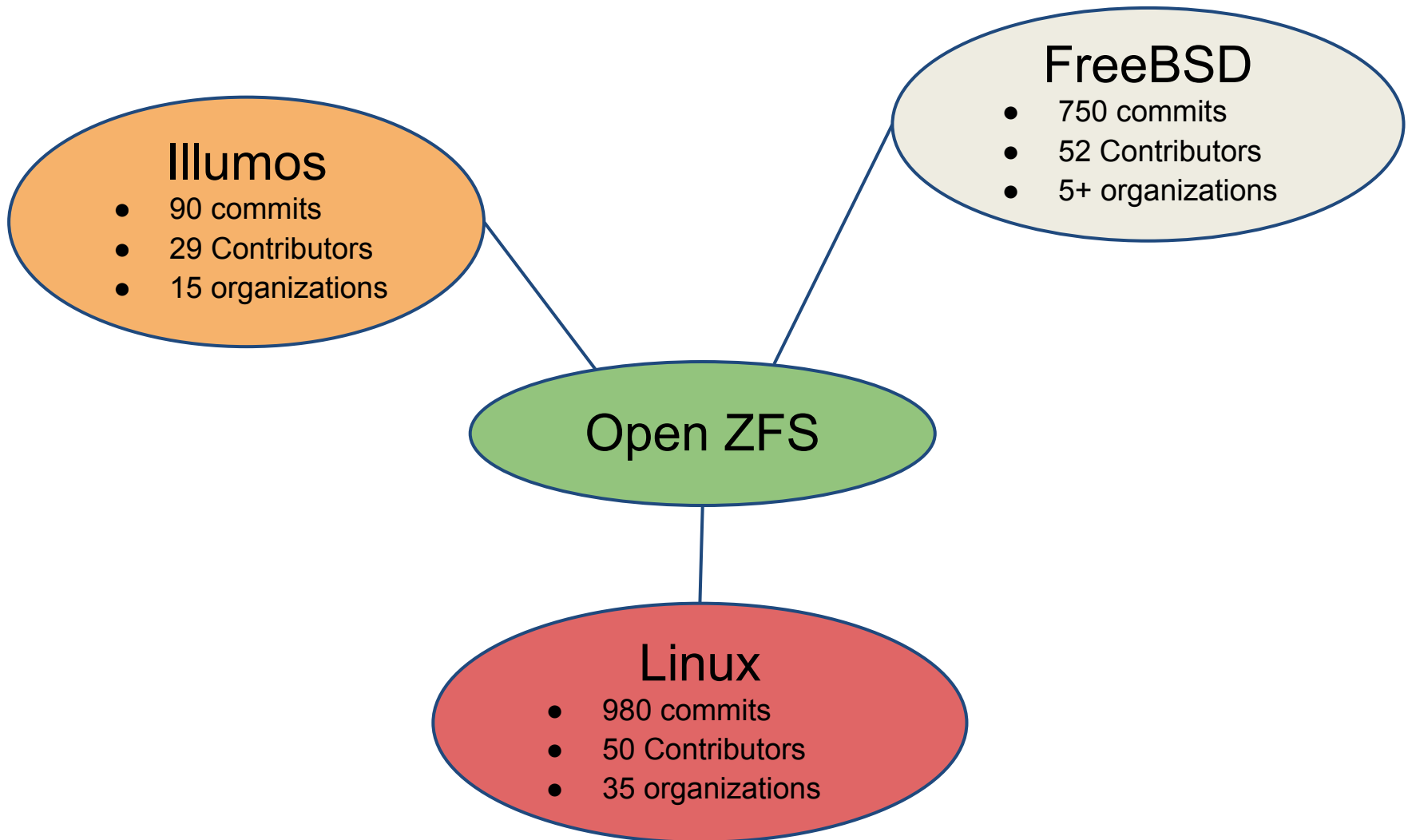
## ZFS History

- 2001: development starts with 2 engineers
- 2005: ZFS source code released
- 2006: ZFS on FUSE for Linux started
- 2008: ZFS released in FreeBSD 7.0
- 2008: Sun's 7000 series ZFS Storage Appliance ships
- 2010: Oracle stops contributing to source code for ZFS
- 2010: illumos is founded as the truly open successor to OpenSolaris
- 2012: ZFS on (native) Linux release candidate

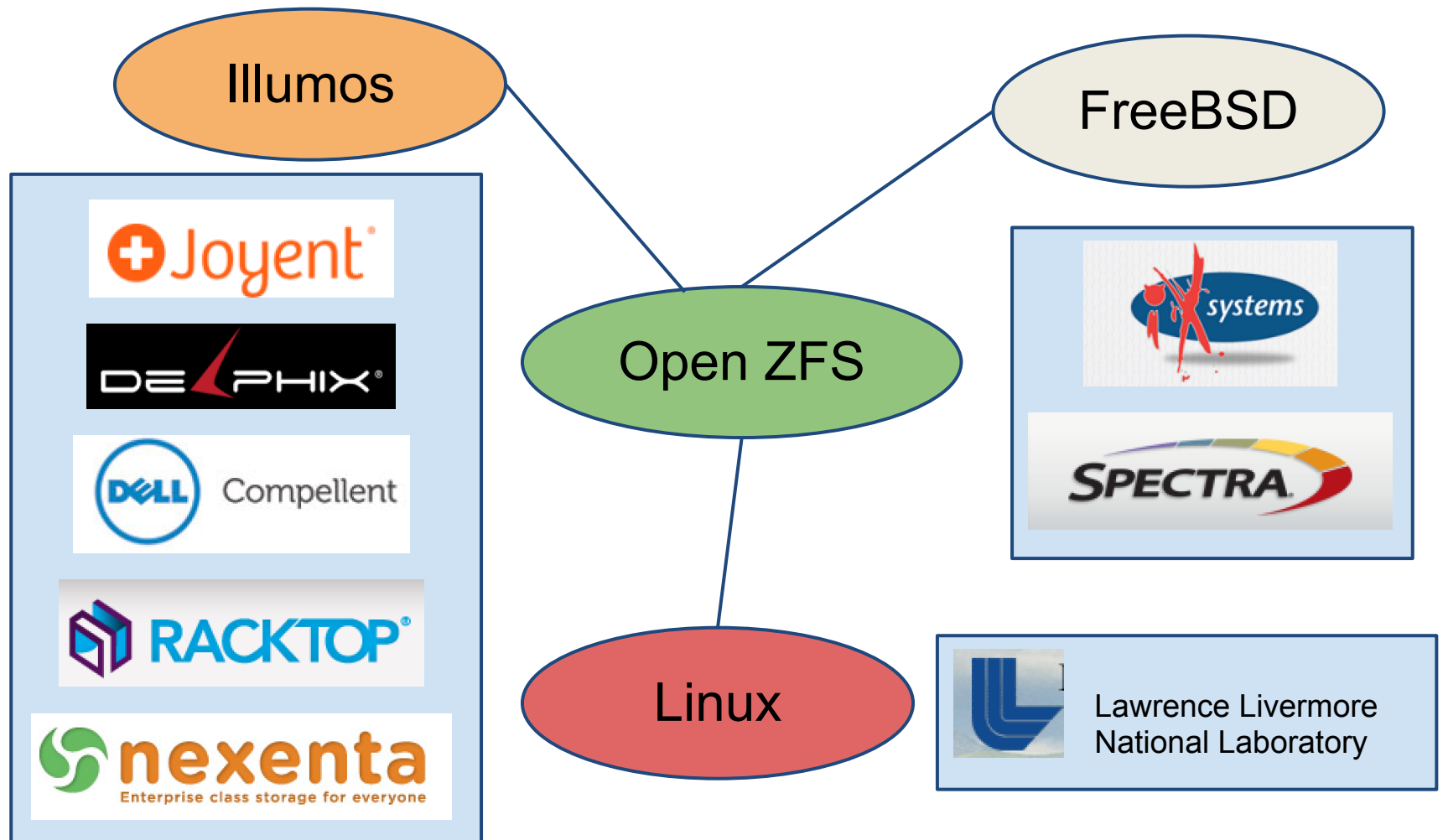
## What is Open ZFS?

- ZFS is not (just) an Oracle technology
- ZFS is not (just) an illumos technology
- ZFS is being developed on several platforms
  - illumos, FreeBSD, Linux, MacOS
- Open ZFS is the continuing work to improve ZFS on every platform
- Open ZFS is ensuring that ZFS is featureful and performant on every platform

# Platform Diversity on Open ZFS



# Platform Diversity on Open ZFS



# Features unique to Open ZFS

- Space accounting
- Debugging and testing improvements
- Performance improvements
- CLI usability
- Programmatic usability

# Space accounting - only in Open ZFS

- refratio property
- written and written@... properties
- zfs **send stream size estimation**
- zfs send **progress reporting** (Bill Pijewski)
- zfs destroy -nv <snapspec> tells how much space is shared by list of snaps

# Debugging & testing - only in Open ZFS

- zfs ioctl args in truss (Gordon Ross)
- ztest backwards compatibility option (Chris Siden)
- **ZFS test suite** returned to working order
  - many tests added
  - new, simpler test framework developed
  - (John Kennedy)
- more complete “zpool history” logging



## Performance - only in Open ZFS

- **single-copy ARC** (George Wilson)
- **compressed L2ARC** (Saso Kiselkov)
- imbalanced LUNs performance improvements (George Wilson)
- zfs destroy <snapspec> destroys many snaps at once
- empty\_bpobj feature
- **background destroy** of filesystems, >100x faster destroy of clones (Chris Siden)
- improved performance for **partial-block writes** (in Spectra Logic)
- per-zone i/o throttling (in SmartOS)
- zfs send/receive of holy objects (e.g. zvols)

## CLI Usability - only in Open ZFS

- “zfs get -t <type>” (Andrew Stormont)
- “zpool iostat” separates out log devices (Mark Harsch)
- “zfs get” takes mountpoint (Sham Pavman)
- zpool “comment” property (Dan McDonald)
- per-vdev space usage & expandsz (George Wilson)
- zdb manpages written (Richard Lowe)

# Programmatic Usability - only in Open ZFS

- clones property
- **libzfs\_core**
  - atomic, thread safe, defined error handling
- zfs snapshot <arbitrary snaps>
- zfs destroy <arbitrary snaps>
- background destroy of filesystems

## Features - only in Open ZFS

- feature flags (Chris Siden & Basil Crow)
- **lz4 compression** (Saso Kiselkov)
- zpool reguid (Garrett D'Amore & George Wilson)
- dump to RAID-Z (Bill Pijewski)
- restore “aclmod” property (Albert Lee)
- SCSI UNMAP support (Dan McDonald)
- TRIM support (in Linux and FreeBSD)
- support for **4k sector size devices** (George Wilson)
- SPA i/o deadman (George Wilson)

# The future of Open ZFS

- persistent I2arc (Saso Kiselkov)
- performance on fragmented pools (George Wilson)
- observability -- zfs dtrace provider
- resumable zfs send/recv
- rainy day performance (e.g. full-ish pools)
- smoother write performance
- filesystem & snapshot count limits (Jerry Jelinek)
- device removal?
- revived MacOS port (Jorgen Lundman)
- wild, application-specific solutions
  - easily extensible architecture
  - modern, object oriented implementation

# Open ZFS

July 2013

Matt Ahrens

[mahrens@delphix.com](mailto:mahrens@delphix.com)

twitter: [@mahrens1](https://twitter.com/mahrens1)

---