OpenZFS Development

Lustre User Group Developer Day

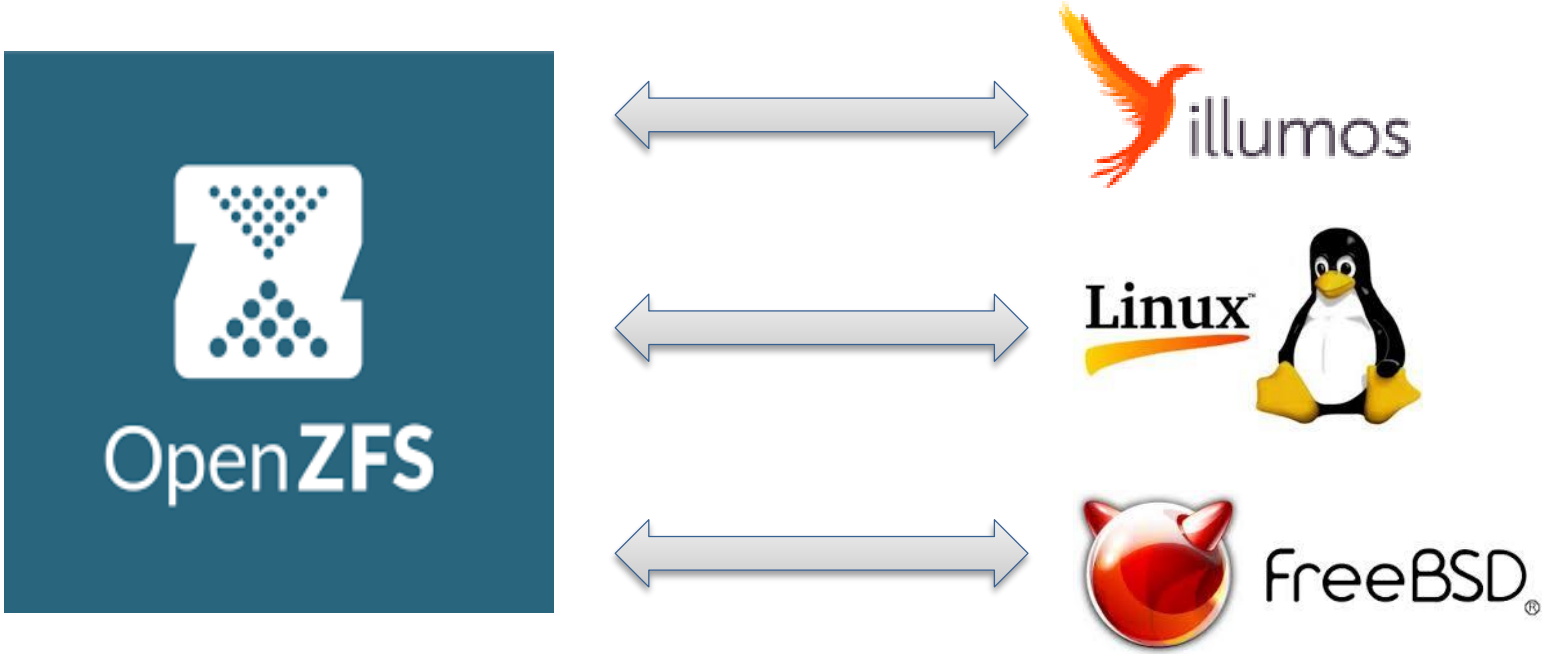
Brian Behlendorf
Lawrence Livermore National Laboratory

May 30th, 2017

LLNL-PRES-725805

Lawrence Livermore National Laboratory

OpenZFS is Available on Multiple Platforms



Developers from all platforms contribute to OpenZFS

ZFS on Linux Releases

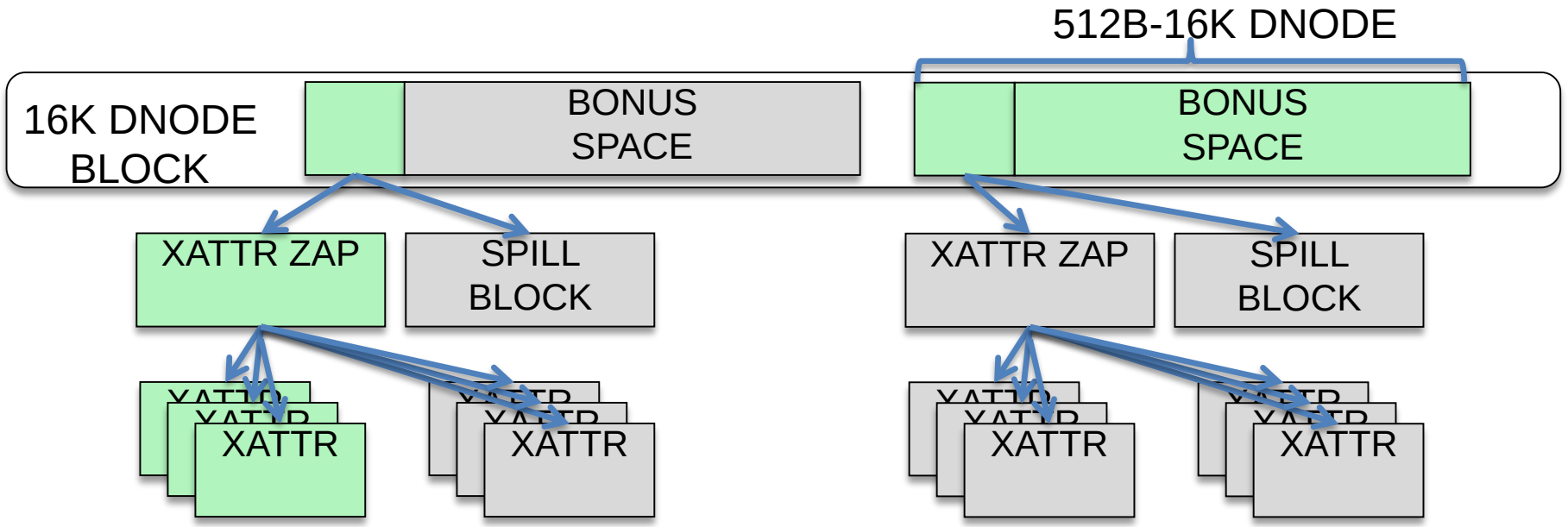
- Current Release (v0.6.x)
  - Critical bug fixes
  - Linux kernel compatibility
  - Low-risk update for distributions
- Upcoming Release (v0.7.x)
  - New features
  - Performance improvements
  - v0.7.0-rc4 released May, 2017

ZFS on Linux provides tagged releases for Linux distributions

Feature User/Group Object Accounting and Quota

- Works the same as space accounting and quota
- Extended ‘zfs userspace’ command
- Existing datasets can be upgraded online

Feature Large Dnodes

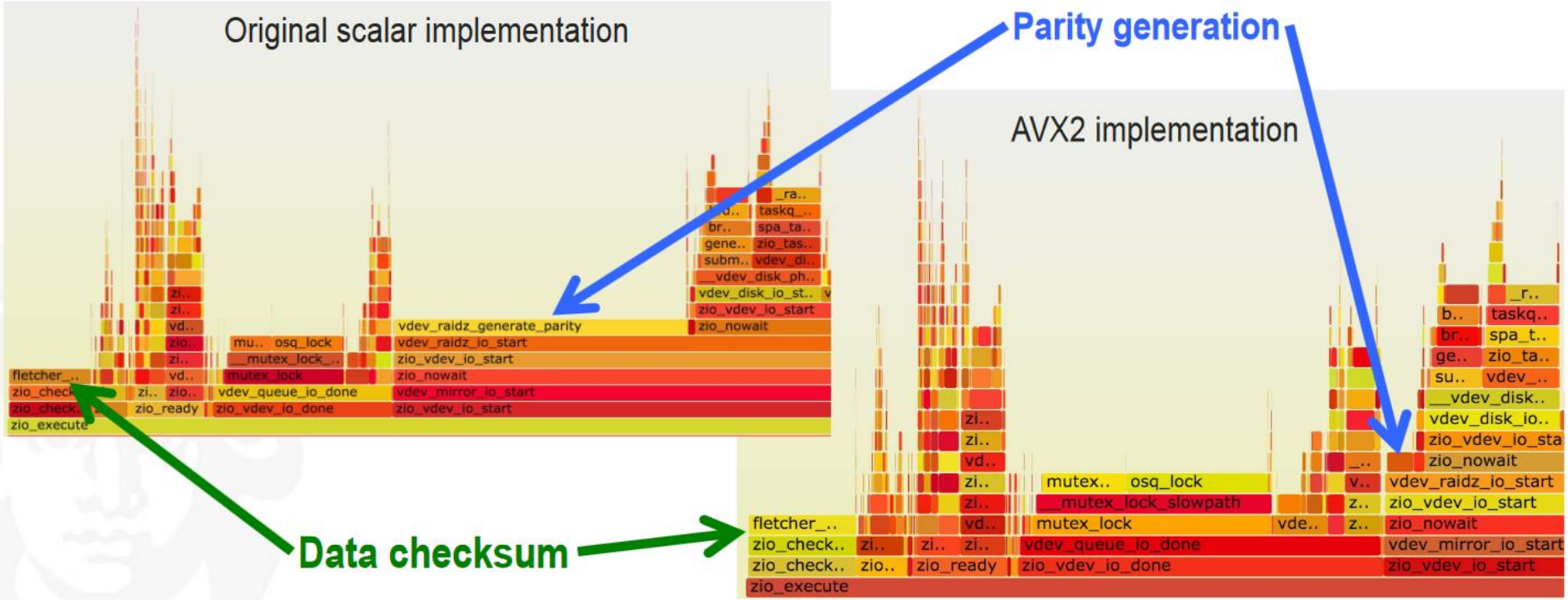


Xattr's stored in dnode, single IO for all small xattrs

Meta Data Performance Improvements

- Multi-threaded TXG syncing
- Multi-threaded object allocation
- Batched quota updates
- Reduced dnode lookups (added by_dnode functions)
- Additional optimizations

Profiling with **perf** and FlameGraph[1)]



1) "Flame Graphs", Brendan Gregg, http://www.brendangregg.com/flamegraphs.html

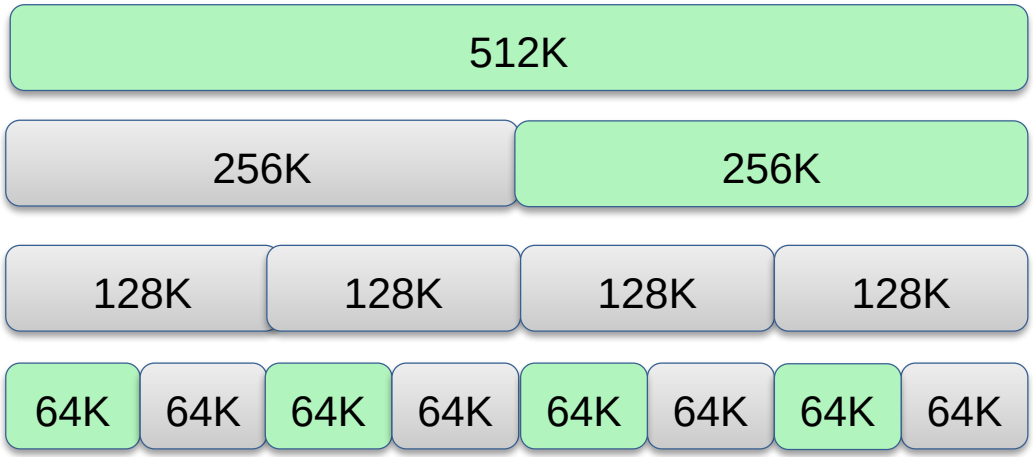
	Scalar	SSE	AVX2	AVX512	NEON
P generate	2.1	2.9	4.3	5.9	1.9
P reconstruct	1.0	1.4	2.1	2.9	1.4
PQ generate	2.0	8.2	12.8	16.7	2.7
Q reconstruct	3.9	6.2	11.7	18.1	4.3
PQ reconstruct	3.0	8.9	16.2	23.8	11.6
PQR generate	2.6	9.1	14.6	19.6	3.5
R reconstruct	14.7	27.8	54.7	75.3	24.5
PR reconstruct	11.4	33.6	61.3	87.0	36.1
QR reconstruct	7.3	19.5	38.8	56.0	26.0
PRQ reconstruct	7.4	19.8	38.2	57.0	33.2

RAIDZ Vectorization - Speed-up relative to original methods

Vectorization – Checksums

- Adapted RAIDZ infrastructure for Fletcher 4
- Micro-benchmarks, ZFS Test Suite, etc
- RAIDZ SIMD implementations:
  - avx2, sse2, ssse3, avx512f, avx512bw, neon, neonx2
- Fletcher4 checksum SIMD implementations:
  - avx2, sse2, ssse3, superscaler, avx512f, neon

Slab Caches

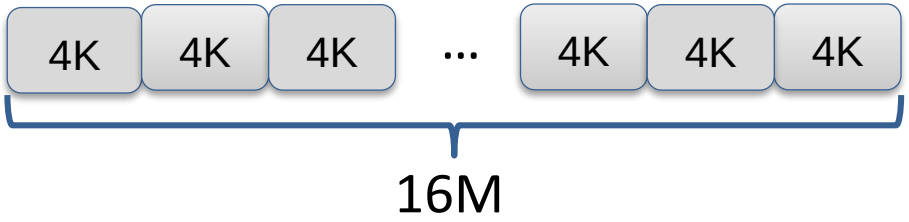


ARC - 1M allocated
System - 2M allocated

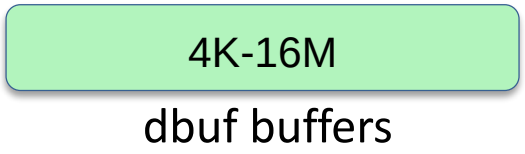
- Large buffers for data blocks
- Slab allocator
  - Reduces allocation cost
  - Fragmentation is complicated
- Slabs can distort the ARCs internal memory accounting
- ARC collapse occurs because slabs cannot be free until all buffers are returned

ARC Buffer Data (ABD)

- All buffers are vectors of pages
  - Minimal waste
  - Fast allocations



- ARC pages compressed in memory
- Uncompressed cache of buffers maintained as for working set

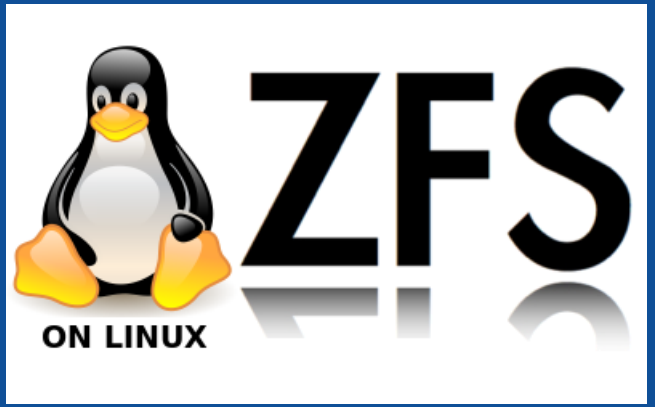


JBOD / Drive Management Features

- Drive monitoring
- Flexible event notification infrastructure
- Auto-online / auto-replace / hot spares
- Fault LED management
- Extended ‘zpool iostat’ and ‘zpool status’ commands

Additional Features In-Progress

- Device Removal/Evacuation - Delphix
- TRIM/Discard - Nexenta
- Native Encryption - Datto
- Declustered Parity (DRAID) – Intel
- Pool Allocation Classes – Intel
- Channel Programs – Delphix
- Scrub/Resilver Performance – Nexenta



http://open-zfs.org

http://zfsonlinux.org

behlendorf1@llnl.gov

Lawrence Livermore National Laboratory