Rearchitecting Storage Stacks For Modern Hardware: Is it worth it?

Luke Logan, Jay Lofstead*, Xian-He Sun, Anthony Kougkas
llogan@hawk.iit.edu, gflofst@sandia.gov, sun@iit.edu, akougkas@iit.edu

DUG'22

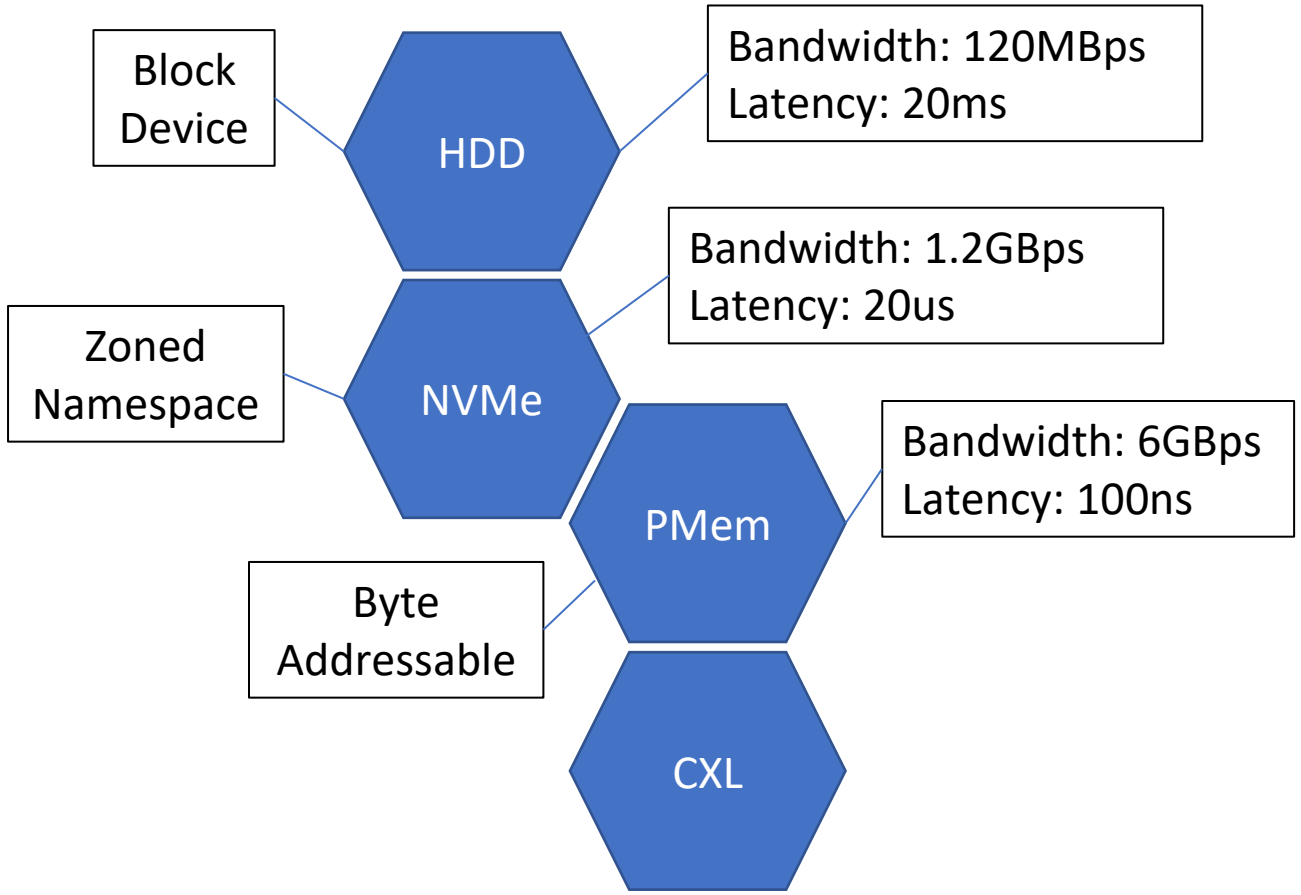
ILLINOIS INSTITUTE OF TECHNOLOGY

Scalable Computing Software Laboratory

Sandia National Laboratories

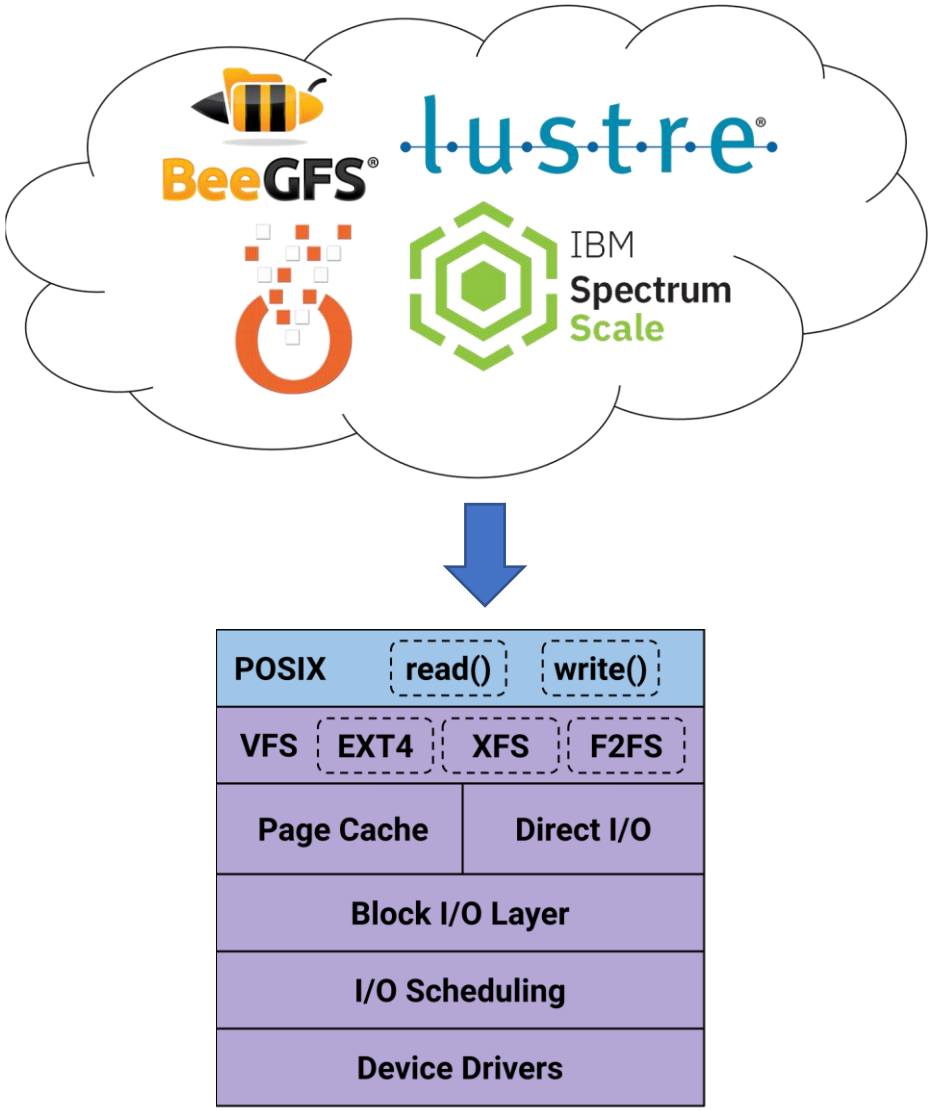
Rapid Evolution of Storage Hardware

- Order of magnitude performance improvements per generation
- New interfaces being exposed
- Hardware-specific optimization!



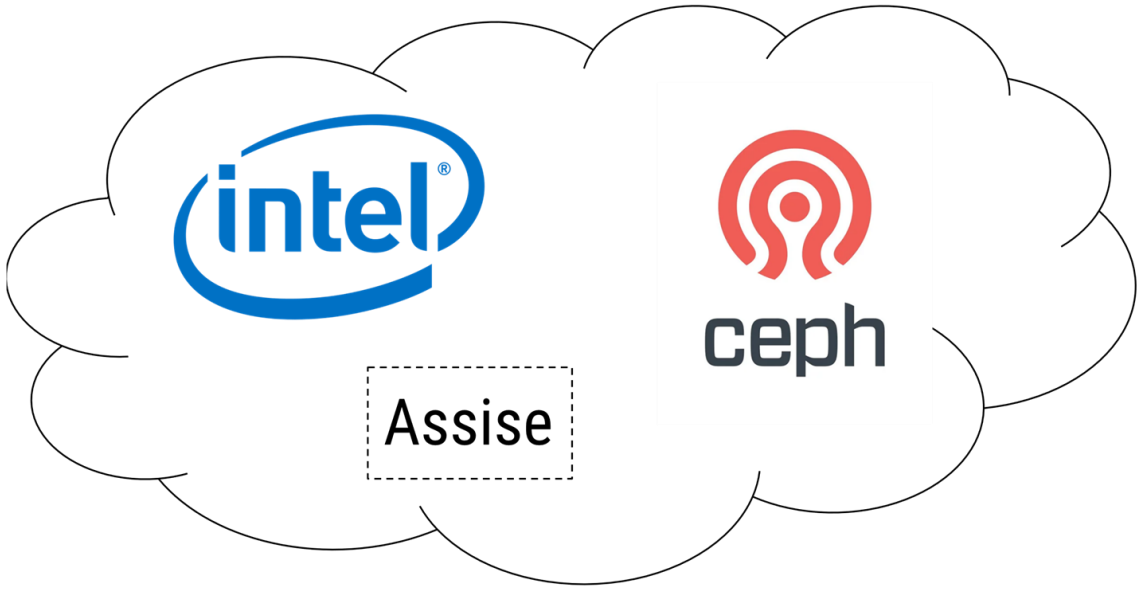
Traditional Parallel Filesystems

- Used in majority of HPC sites
- Designed for hard drives
- Rely on kernel I/O stack
- **Do not take full advantage of new hardware interfaces!**



Rebuilding The Storage Stack

- Reduce software overhead
- Provide hardware-specific optimization
- Large implementation effort



But, what's the impact?

The value of optimizing storage stacks is not well-understood in a distributed setting

- Many single-node evaluations (not distributed)
- Emulate PMEM using DRAM
- Old software versions (e.g., DAOS 1.0)

Our Goal

We quantify the performance benefit of using DAOS compared to traditional storage stacks over real hardware

Evaluation

Testbed

- 4 PMEM nodes
- **CPU:** 96 cores / 192 threads
  - 2x Intel(R) Xeon(R) Gold 6342 CPU@2.80GHz
- **Network:** 100GBe, IPoIB
- **PMEM:** 2TB
  - 8x Intel Optane DC Persistent Memory (256GB per module)
- **NVMe:** 64TB
  - 16x 4TB NVMe

Software

- **OS:** Centos8
  - Kernel 4.18
- **DAOS:** 2.1.104-tb
- **OrangeFS:** 2.9.8
- **BeegFS:** 3.7.1
- **Io500:** isc'22
- **MPI:** mpich 3.3.2

Comparative Study

How well does DAOS perform compared to BeeGFS and OrangeFS under various workloads on real hardware?

Experimental Setup

OrangeFS + BeeGFS

- Default config
- Stripe Size:
  - OrangeFS: 64KB
  - BeeGFS: 512KB
- Filesystem: EXT4
- Co-locate metadata + data servers

DAOS (NVMe)

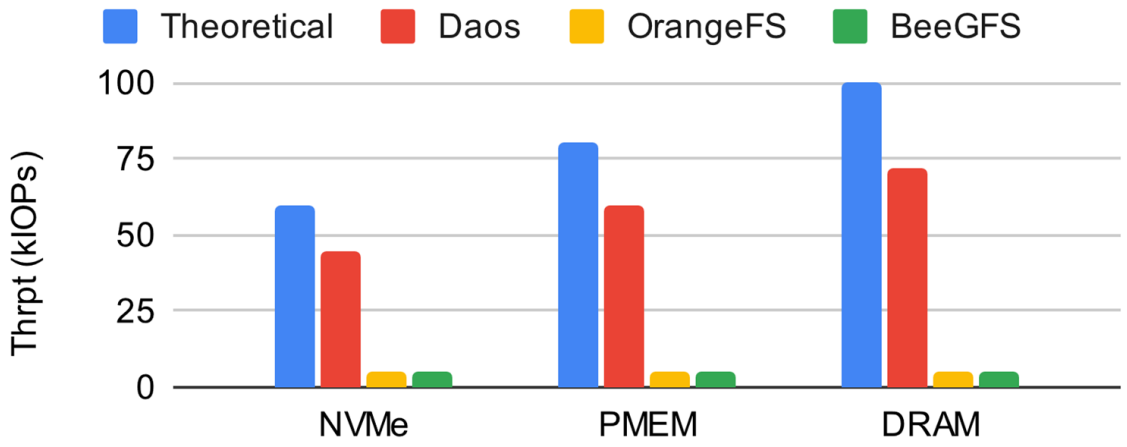
- Cache: 50GB PMEM
  - As low as it would go
- 5TB NVMe
- 1 dedicated core

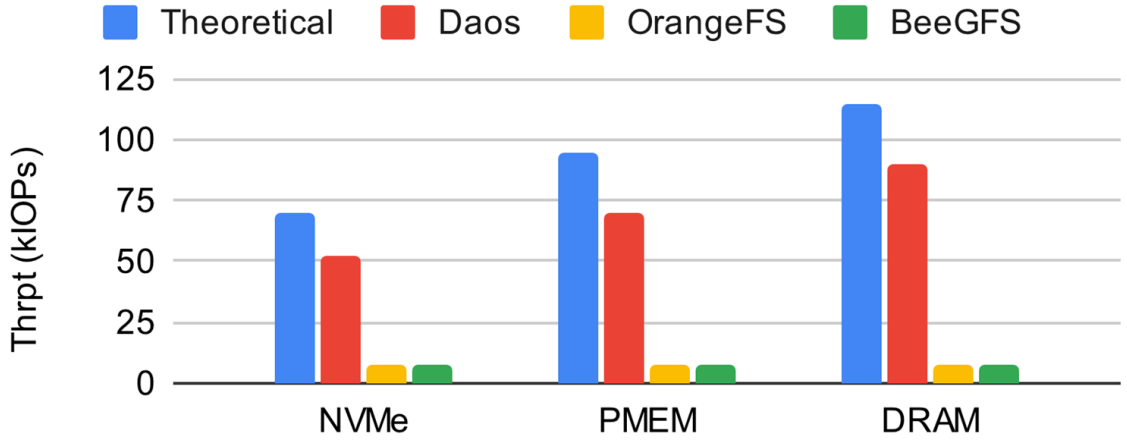
DAOS (PMEM)

- 6TB PMEM

IO500 (Mdtest)

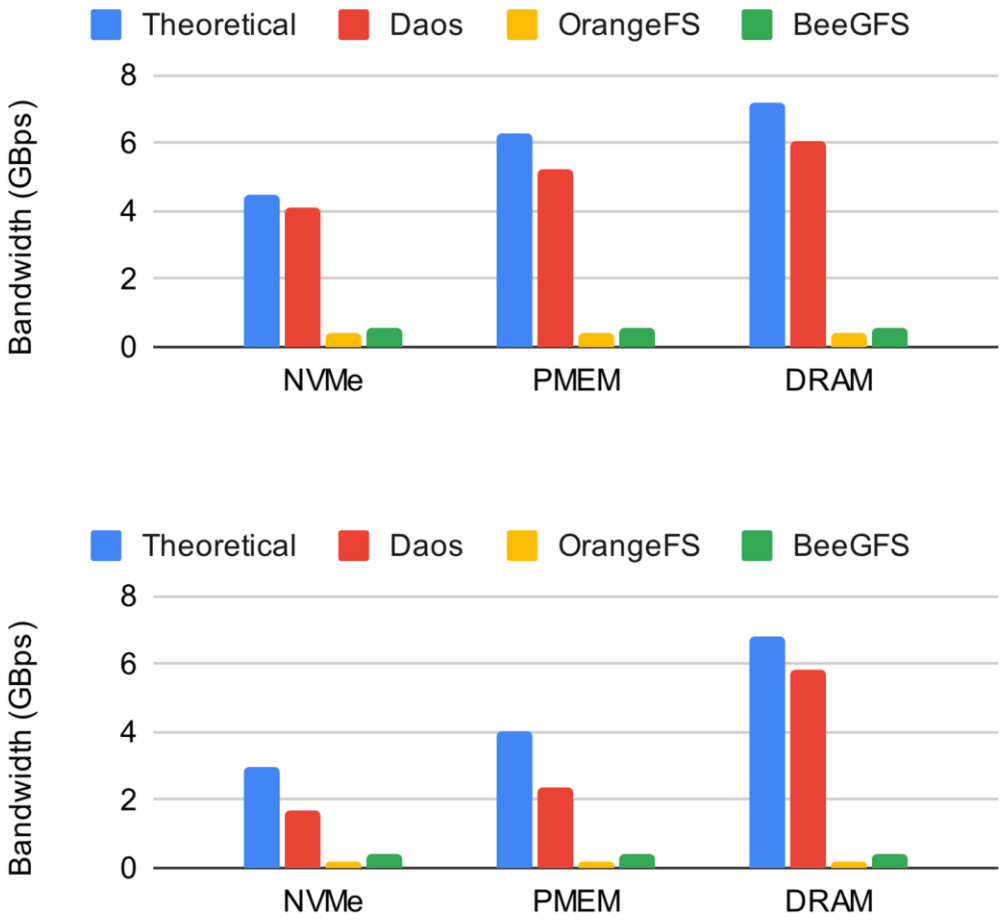
- Stress metadata ops
  - E.g., open / close
- DAOS is 15x faster than others
- Less software overhead due to leaner stack
  - Kernel uses interrupts + context switching for I/O
  - DAOS uses API interception





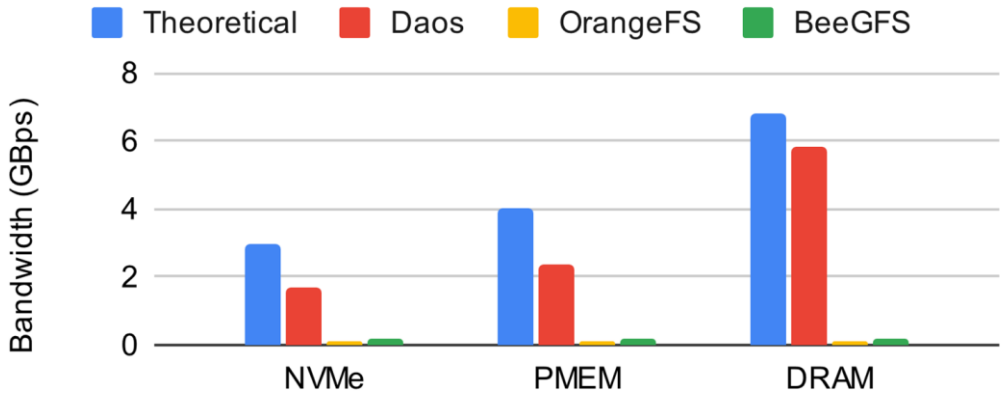
IO500 (IOR-Hard)

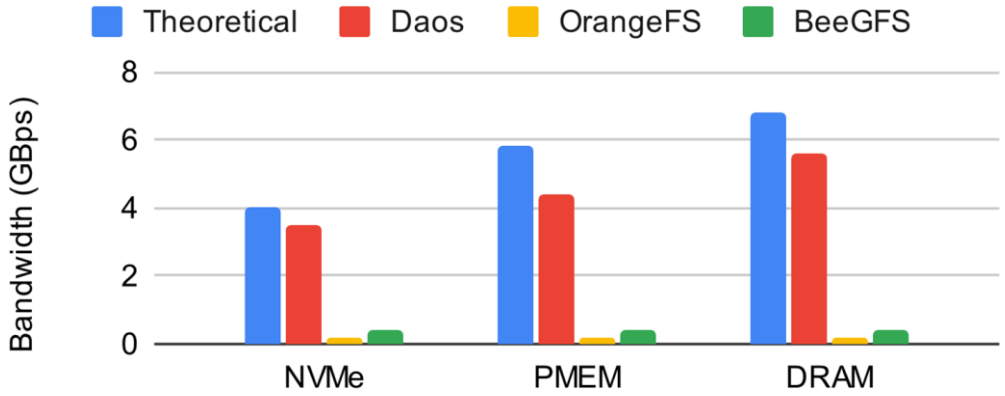
- Small, unaligned I/O
- Worst case for PFS
- DAOs was 8x faster than others



IO500 (IOR-Easy)

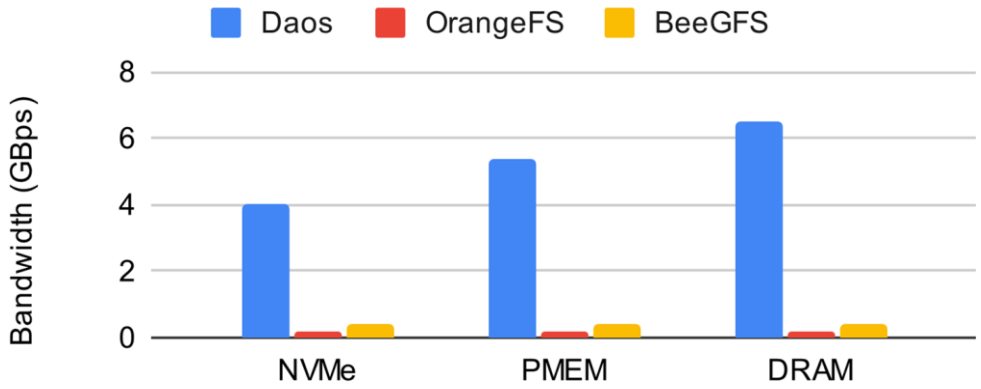
- Large-sequential I/O for 5 minutes
- Best case for a PFS
- Still outperforms traditional stack by 10x

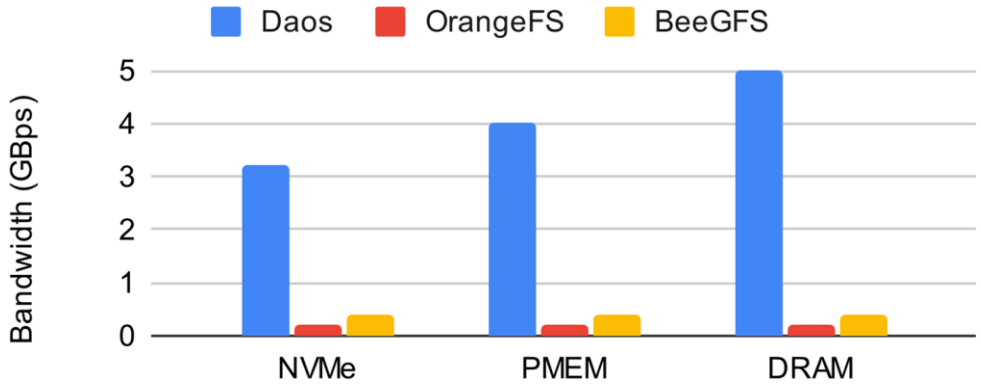




VPIC + BD-CATS

- VPIC: write-only particle simulator
  - Checkpoint-restart
  - 30GB / checkpoint
  - 16 checkpoints (480GB in total)
- BD-Cats: read-only clustering
- Similar to IOR-Easy
- Overall, 6x faster than others





Storage Backend Study

How well does DAOS perform when using different storage backends?

(i.e., kernel vs SPDK)

Experimental Setup

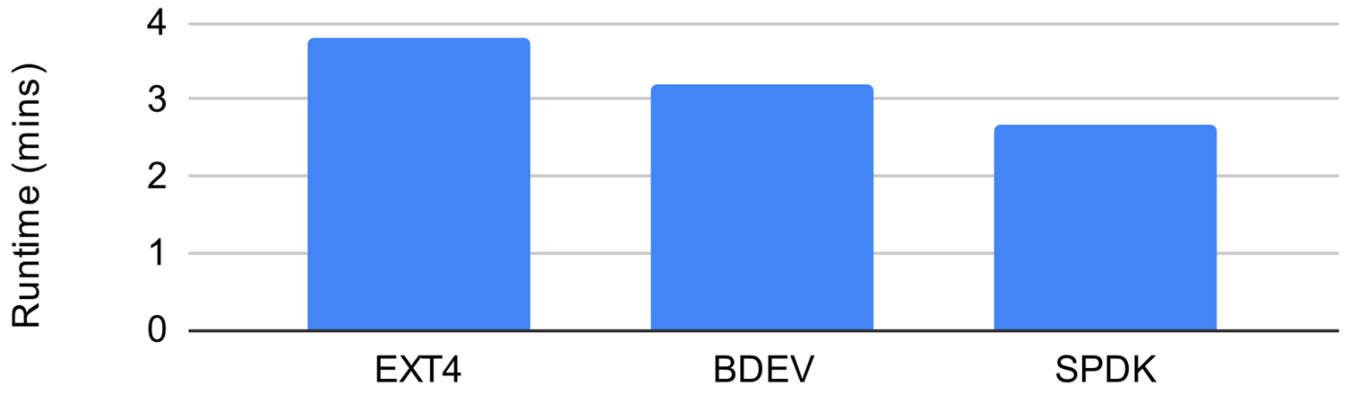
Cosmic Tagger

- Convlutional Neural Net for separating neutrino pixels
- 430,000 samples in the dataset
- 430GB size
- 20 – 40KB I/O sizes

DAOS

- 50GB of PMEM
- 5TB of NVMe
- 1 server core

Cosmic Tagger



- 4 minutes on EXT4 -> 2.5 minutes on SPDK
- 40% performance improvement due to using SPDK!

Conclusion

Conclusion

- We conducted numerous benchmarks of DAOS over modern hardware in a distributed setting
- DAOS outperforms traditional storage stacks by as much as 15x on modern hardware
- The performance improvement is due to hardware optimization
- It is worthwhile to utilize hardware-optimized storage stacks


thank
you