Designing High-Performance, Resilient and Heterogeneity-Aware Key-Value Storage for Modern HPC Clusters

Dipti Shankar

Dr. Dhabaleswar K. Panda (Advisor)

Dr. Xiaoyi Lu (Co-Advisor)

Department of Computer Science & Engineering
The Ohio State University

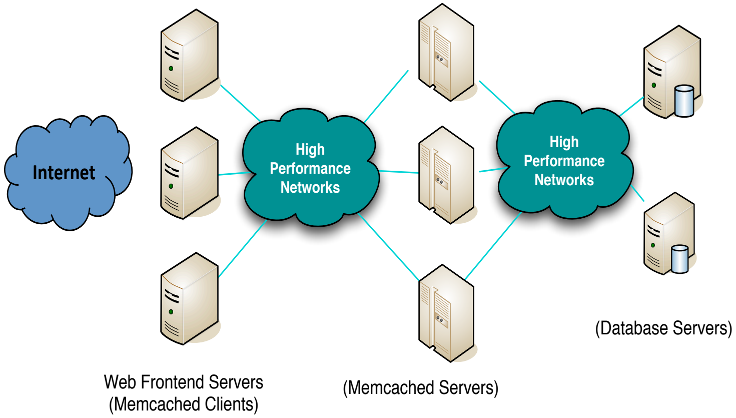
Outline

- Introduction and Problem Statement
- Research Highlights and Results
- Broader Impact
- Conclusion & Future Avenues

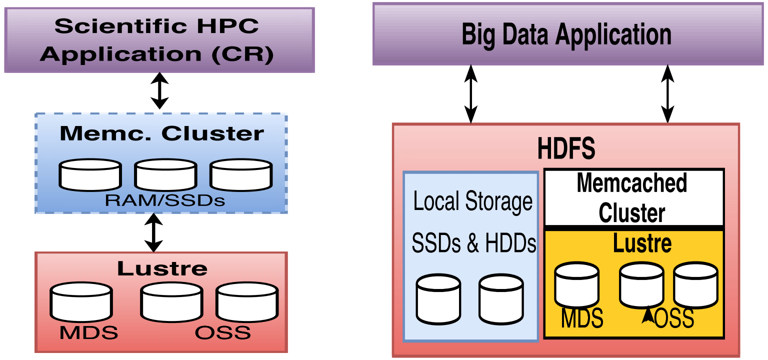
Key-Value Storage in HPC and Data Centers

- General purpose distributed memory-centric storage
  - Allows to aggregate spare memory from multiple nodes (e.g, Memcached)
- Accelerating Online and Offline Analytics in High-Performance Compute (HPC) environments
- Our Basis: Current High-performance and hybrid key-value stores for modern HPC clusters
  - ❖ High-Performance Network Interconnects (e.g., InfiniBand)
    - ❖ Low end-to-end latencies with IP-over-InfiniBand (IPoIB) and Remote Direct Memory Access (RDMA)
  - ❖ 'DRAM+SSD' hybrid memory designs
    - ❖ Extend storage capabilities beyond DRAM capabilities using high-speed SSDs

(Online Analytical Workloads: OLTP/NoSQL Query Cache)

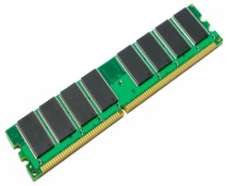


(Offline Analytical Workloads: Software-Assisted Burst-Buffer)

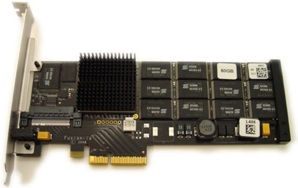


Drivers of Modern HPC Cluster Architectures

High Performance Interconnects



Large Memory Nodes (DRAM)



SSD, NVMe-SSD, NVRAM

Multi-core Processors with vectorization support + Accelerators (GPUs)

- Multi-core/many-core technologies
- Remote Direct Memory Access (RDMA)-enabled networking (InfiniBand and RoCE)
- Solid State Drives (e.g., PCIe/NVMe-SSDs), NVRAM (e.g., PCM, 3DXpoint), Parallel Filesystems (e.g., Lustre)
- Accelerators (e.g., NVIDIA GPGPUs)
- Production-scale HPC Clusters: SDSC Comet, TACC Stampede, OSC Owens, etc.

Holistic Approach for HPC-Centric KVS

- Our focus: Holistic approach to designing a high-performance and resilient key-value storage for current and emerging HPC systems
  - RDMA-enabled networking
  - Hybrid NVM storage-awareness: High-speed NVMe-/PCIe-SSDs and upcoming NVRAM technologies
  - Heterogeneous compute devices (e.g., SIMD capabilities of CPUs and GPUs)
- Goals: Maximize end-to-end performance to
  - Optimally exploit available compute/storage/network capabilities
  - Enable data-intensive Big Data applications to leverage memory-centric key-value storage to improve their overall performance

Research Framework

Application Workloads

Online
(E.g., SQL/NoSQL query cache/
LinkBench and TAO Graph KV workloads)

Offline
(E.g., Burst-Buffer over PFS for Hadoop for
MapReduce/Spark)

High-Performance, Hybrid and Resilient Key Value Storage

Non-Blocking RDMA-aware API Extensions

Fast Online Erasure Coding
(Reliability)

NVRAM-aware Communication
Protocols (Persistence)

Accelerations on SIMD-based
(e.g., GPU) Architecture
(Scalability)

Heterogeneity-Aware (DRAM/NVRAM/NVMe/PFS) Key-Value Storage Engine

Modern HPC System Architecture

Multi-Core
Nodes
(w/ SIMD units)

GPUs

Volatile/Non-Volatile
Memory Technologies
(DRAM, NVRAM)

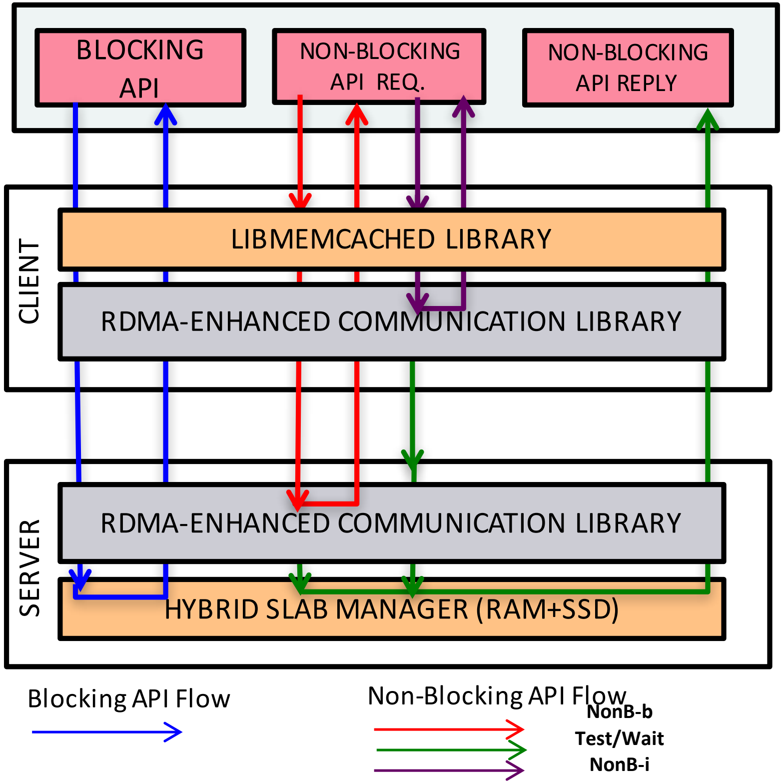
Local Storage
(PCIe SSD/
NVMe-SSD)

RDMA-Capable
Networks (InfiniBand,
40Gbe RoCE)

Parallel File
System (Lustre)

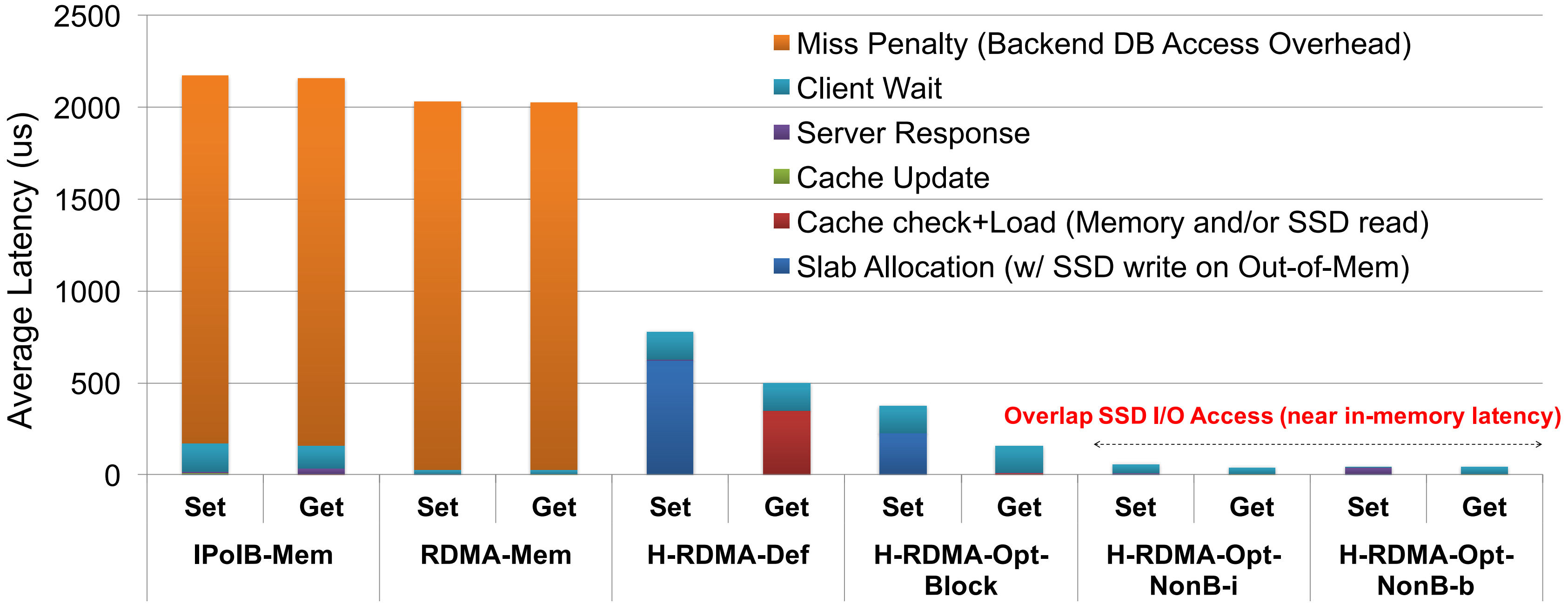
High-Performance Non-Blocking API Semantics

- Heterogeneous Storage-Aware Key-Value Stores (e.g., 'DRAM + PCIe/NVMe-SSD')
  - Higher data retention at the cost of SSD I/O; suitable for out-of-memory scenarios
  - Performance limited by Blocking API semantics
- Goals: Achieve near in-memory speeds while being able to exploit hybrid memory
- Approach: Novel Non-blocking API Semantics to extend RDMA-Libmemcached library
  - memcached_(iset/iget/bset/bget) APIs for SET/GET
  - memcached_(test/wait) APIs for progressing communication



D. Shankar, X. Lu , N. Islam , M. W. Rahman , and D. K. Panda, "High-Performance Hybrid Key-Value Store on Modern Clusters with RDMA Interconnects and SSDs: Non-blocking Extensions, Designs, and Benefits", IPDPS 2016

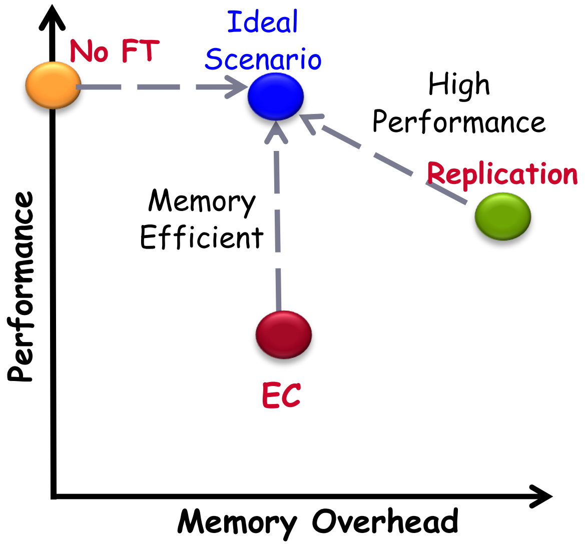
High-Performance Non-Blocking API Semantics



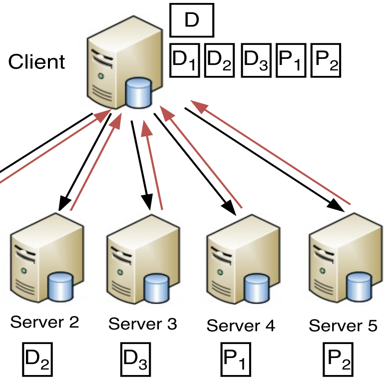
- Set/Get Latency with Non-Blocking API: Up to 8x gain in overall latency vs. blocking API semantics over RDMA+SSD hybrid design
- Up to 2.5x gain in throughput observed at client; Ability to overlap request and response phases to hide SSD I/O overheads

Fast Online Erasure Coding with RDMA

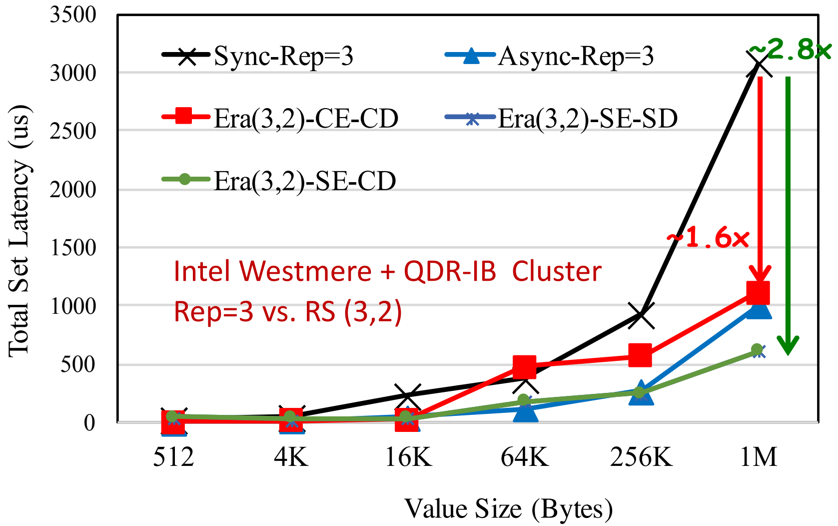
- Erasure Coding (EC): A low storage-overhead alternative to Replication
- Bottlenecks: (1) Encode/decode compute overheads (2) Communication overhead of scattering/ gathering distributed data/parity chunks
- Goal: Making Online EC viable for key-value stores
- Approach: Non-blocking RDMA-aware semantics to enable compute/communication overlap
- Encode/Decode offload capabilities integrated into Memcached client (CE/CD) and server (SE/SD)

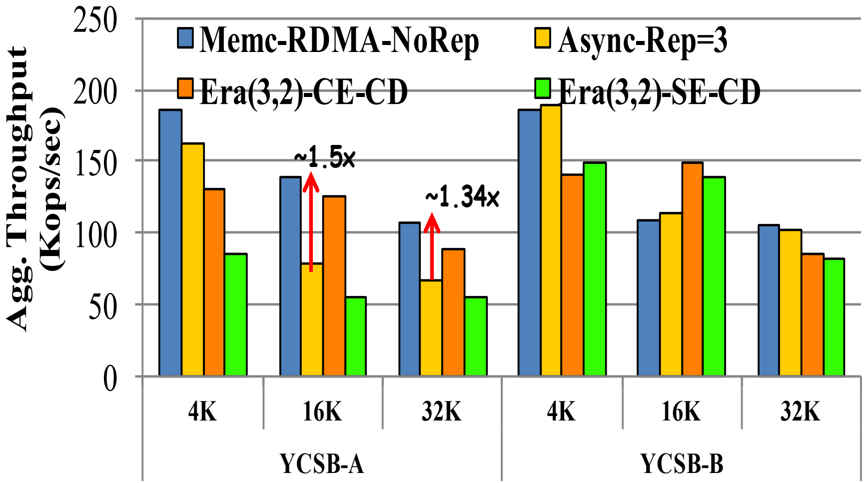


RS (3,2) => 66% Storage Overhead vs 200% of Replication-Factor=3



Fast Online Erasure Coding with RDMA

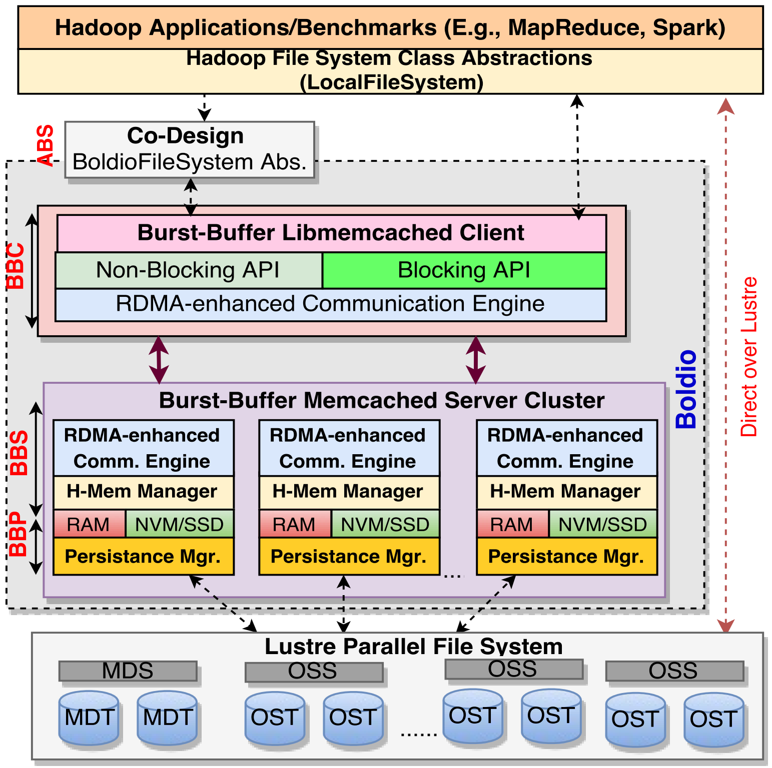




- Experiments with YCSB for Online EC vs. Async. Rep:
  - 150 Clients on 10 nodes on SDSC Comet Cluster (IB FDR + 24-core Intel Haswell) over 5-node RDMA-Memcached Cluster
  - (1) CE-CD gains ~1.34x for Update-Heavy workloads; SE-CD on-par (2) CE-CD/SE-CD on-par for Read-Heavy workloads

D. Shankar , X. Lu , and D. K. Panda, “High-Performance and Resilient Key-Value Store with Online Erasure Coding for Big Data Workloads”, 37th International Conference on Distributed Computing Systems (ICDCS 2017)

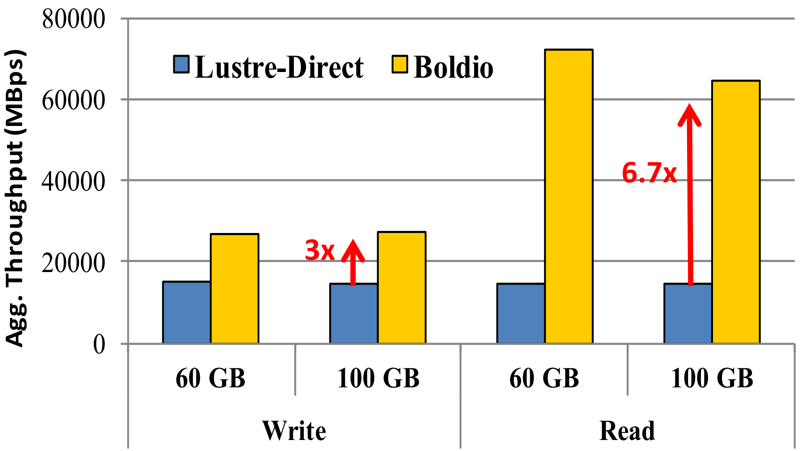
Co-Designing Key-Value Store-based Burst Buffer over PFS

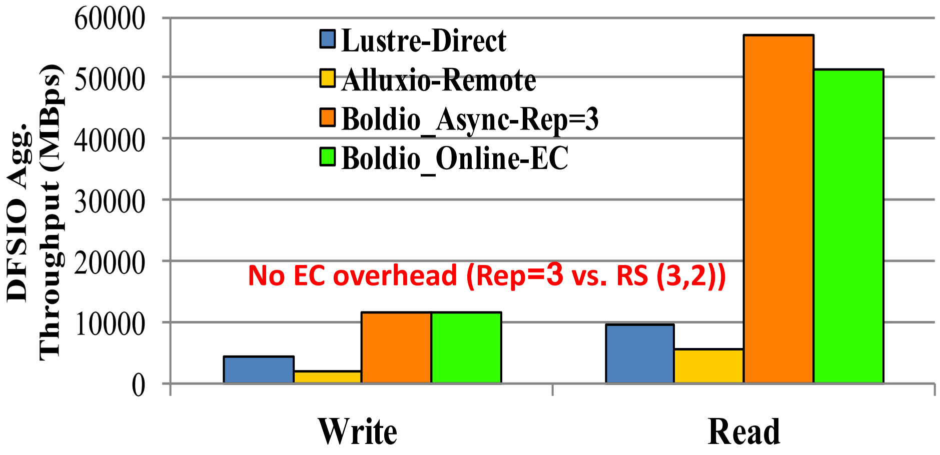


- Offline Data Analytics Use-Case: Hybrid and resilient key-value store-based Burst-Buffer system Over Lustre (Boldio)
- Overcome local storage limitations on HPC nodes; performance of `data locality'
- Light-weight transparent interface to Hadoop/ Spark applications
- Accelerating I/O-intensive Big Data workloads
  - Non-blocking RDMA-Libmemcached APIs to maximize overlap
  - Client-based replication or Online Erasure Coding with RDMA for resilience
  - Asynchronous persistence to Lustre parallel file system at RDMA-Memcached Servers

D. Shankar, X. Lu, D. Panda, Boldio: A Hybrid and Resilient Burst-Buffer over Lustre for Accelerating Big Data I/O, IEEE International Conference on Big Data 2016 (Short Paper)

Co-Designing Key-Value Store-based Burst Buffer over PFS



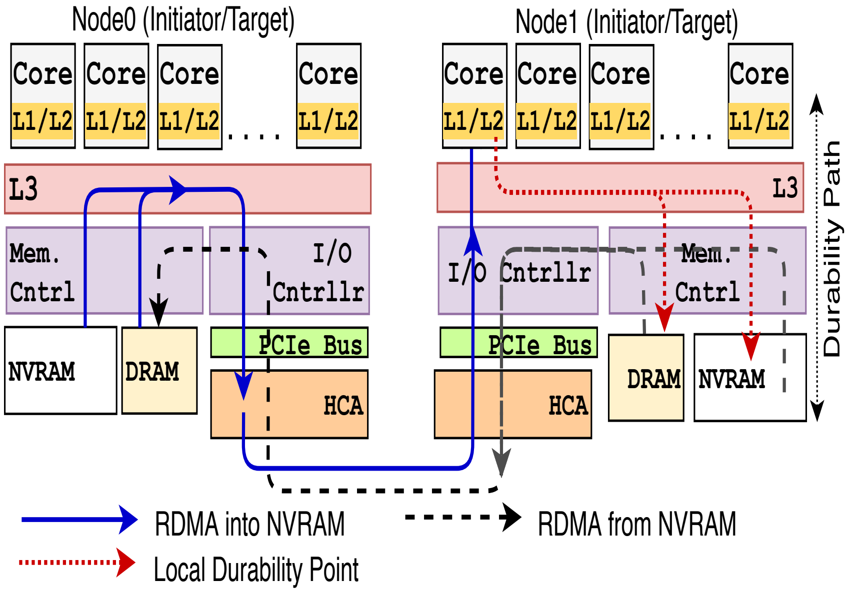


- TestDFSIO on SDSC Gordon Cluster (16-core Intel Sandy Bridge and IB QDR) with 16-node MapReduce Cluster + 4-node Boldio Cluster
- Boldio can sustain 3x and 6.7x gains in read and write throughputs over stand-alone Lustre

- TestDFSIO on Intel Westmere Cluster (8-core Intel Sandy Bridge and IB QDR); 8-node MapReduce Cluster + 5-node Boldio Cluster over Lustre
- Performance gains over designs like Alluxio (formerly Tachyon) in HPC environments with no local storage

Exploring Opportunities with NVRAM and RDMA

- Emerging Non-volatile Memory technologies (NVRAM)
- Potential: Byte-addressable and persistent; capable of RDMA
- Observations: RDMA writes into NVRAM needs to guarantee remote durability
  - Appliance Method: Hardware-Assisted remote direct persistence
  - General Purpose Server Method: Server-Assisted software-based persistence
- Opportunities: Designing high-performance RDMA-based Persistence Protocols (e.g., Persistent In-Memory KVS over NVRAM)



D. Shankar, X. Lu, D. Panda, RDMP-KVS: Remote Direct Memory Persistence Aware Key-Value Store for NVRAM Systems (Under Submission)

Broader Impact: The High-Performance Big Data (HiBD) Project

- RDMA for Apache Spark (RDMA-Spark), Apache Hadoop 2.x (RDMA-Hadoop-2.x), RDMA for Apache HBase
- RDMA for Memcached (RDMA-Memcached)
  - RDMA-aware `DRAM+SSD’ hybrid Memcached server design
  - Non-Blocking RDMA-based Client API designs (RDMA-Libmemcached)
  - Based on Memcached 1.5.3 and Libmemcached client 1.0.18
  - Available for InfiniBand and RoCE
- OSU HiBD-Benchmarks (OHB)
  - Memcached Set/Get Micro-benchmarks for Blocking and Non-Blocking APIs, and Hybrid Memcached designs
  - YCSB plugin for RDMA-Memcached
  - Also includes HDFS, HBase, Spark Micro-benchmarks
- http://hibd.cse.ohio-state.edu
- Users Base: 290 organizations, 34 countries, 27,950 downloads

HiBD
High-Performance Big Data

Network Based Computing Laboratory

The Ohio State University

Conclusion & Future Avenues

- Holistic approach to designing key-value storage by exploiting the capabilities of HPC clusters for (1) performance, (2) scalability, and, (3) data resilience/availability
- RDMA-capable Networks: (1) Proposed Non-blocking RDMA-based Libmemcached APIs (2) Fast Online EC-based RDMA-Memcached designs
- Heterogeneous Storage-Awareness: (1) Leverage `RDMA+SSD' hybrid designs, (2) `RDMA+NVRAM' Persistent Key-Value Storage
- Application Co-Design: Memory-centric data-intensive applications on HPC Clusters
  - Online (e.g., SQL query cache, YCSB) and Offline Data Analytics (e.g., Boldio Burst-Buffer for Hadoop I/O)
- Future Work: Ongoing work in this thesis direction
  - Heterogeneous compute capabilities of CPU/GPU: End-to-end SIMD-aware KVS designs
  - Exploring co-design of (1) Read-intensive Graph-based workloads (E.g., LinkBench, RedisGraph) (2) Key-value storage engine for ML Parameter Server frameworks

Thank You!

shankar.50@osu.edu

http://www.cse.ohio-state.edu/~shankard

Network-Based Computing Laboratory
http://nowlab.cse.ohio-state.edu/

The High-Performance Big Data Project
http://hibd.cse.ohio-state.edu/