HiBD
High-Performance
Big Data

HPC Meets BigData: Accelerating Apache Hadoop, Spark, and Memcached with HPC Technologies

Keynote Talk at HPCAC, Stanford (February 2017)

by

Dhabaleswar K. (DK) Panda

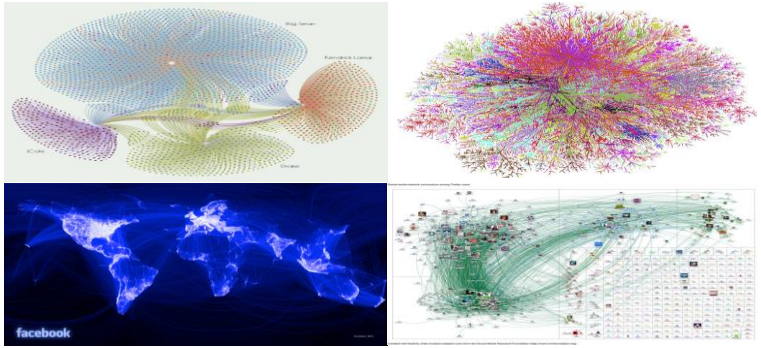
The Ohio State University

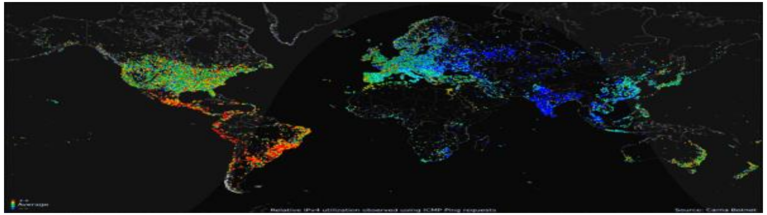
E-mail: panda@cse.ohio-state.edu

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

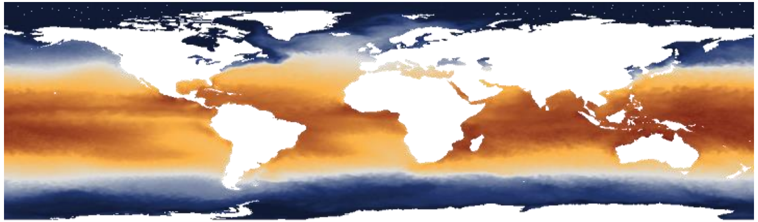
Introduction to Big Data Analytics and Trends

- Big Data has changed the way people understand and harness the power of data, both in the business and research domains
- Big Data has become one of the most important elements in business analytics
- Big Data and High Performance Computing (HPC) are converging to meet large scale data processing challenges
- Running High Performance Data Analysis (HPDA) workloads in the cloud is gaining popularity
  - According to the latest OpenStack survey, 27% of cloud deployments are running HPDA workloads





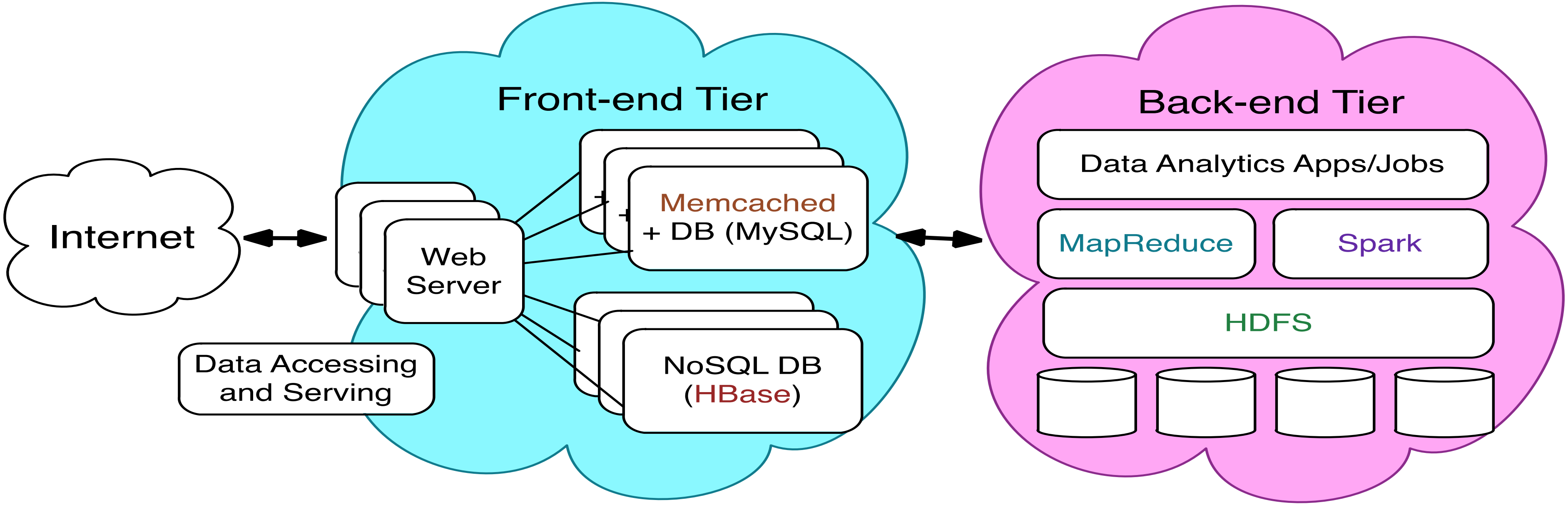
http://www.coolinfographics.com/blog/tag/data?currentPage=3



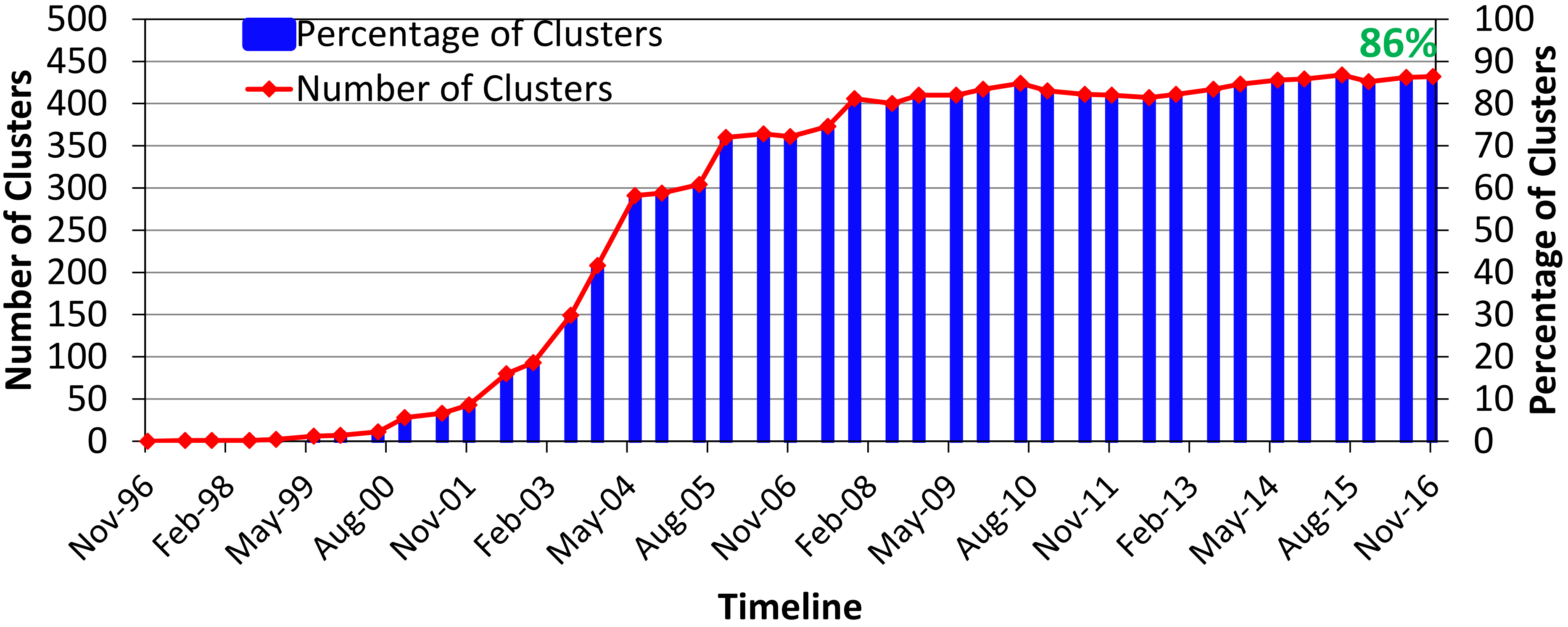
http://www.climatecentral.org/news/white-house-brings-together-big-data-and-climate-change-17194

Data Management and Processing on Modern Clusters

- Substantial impact on designing and utilizing data management and processing systems in multiple tiers
  - Front-end data accessing and serving (Online)
    - Memcached + DB (e.g. MySQL), HBase
  - Back-end data analytics (Offline)
    - HDFS, MapReduce, Spark



Trends for Commodity Computing Clusters in the Top 500 List (http://www.top500.org)



Drivers of Modern HPC Cluster and Data Center Architecture

Multi-/Many-core Processors

High Performance Interconnects – InfiniBand (with SR-IOV)
<1usec latency, 200Gbps Bandwidth>

Accelerators / Coprocessors
high compute density, high performance/watt
>1 TFlop DP on a chip

SSD, NVMe-SSD, NVRAM

- Multi-core/many-core technologies
- Remote Direct Memory Access (RDMA)-enabled networking (InfiniBand and RoCE)
  - Single Root I/O Virtualization (SR-IOV)
- Solid State Drives (SSDs), NVM, Parallel Filesystems, Object Storage Clusters
- Accelerators (NVIDIA GPGPUs and Intel Xeon Phi)

SDSC Comet

TACC Stampede

Microsoft Azure

amazon web services EC2

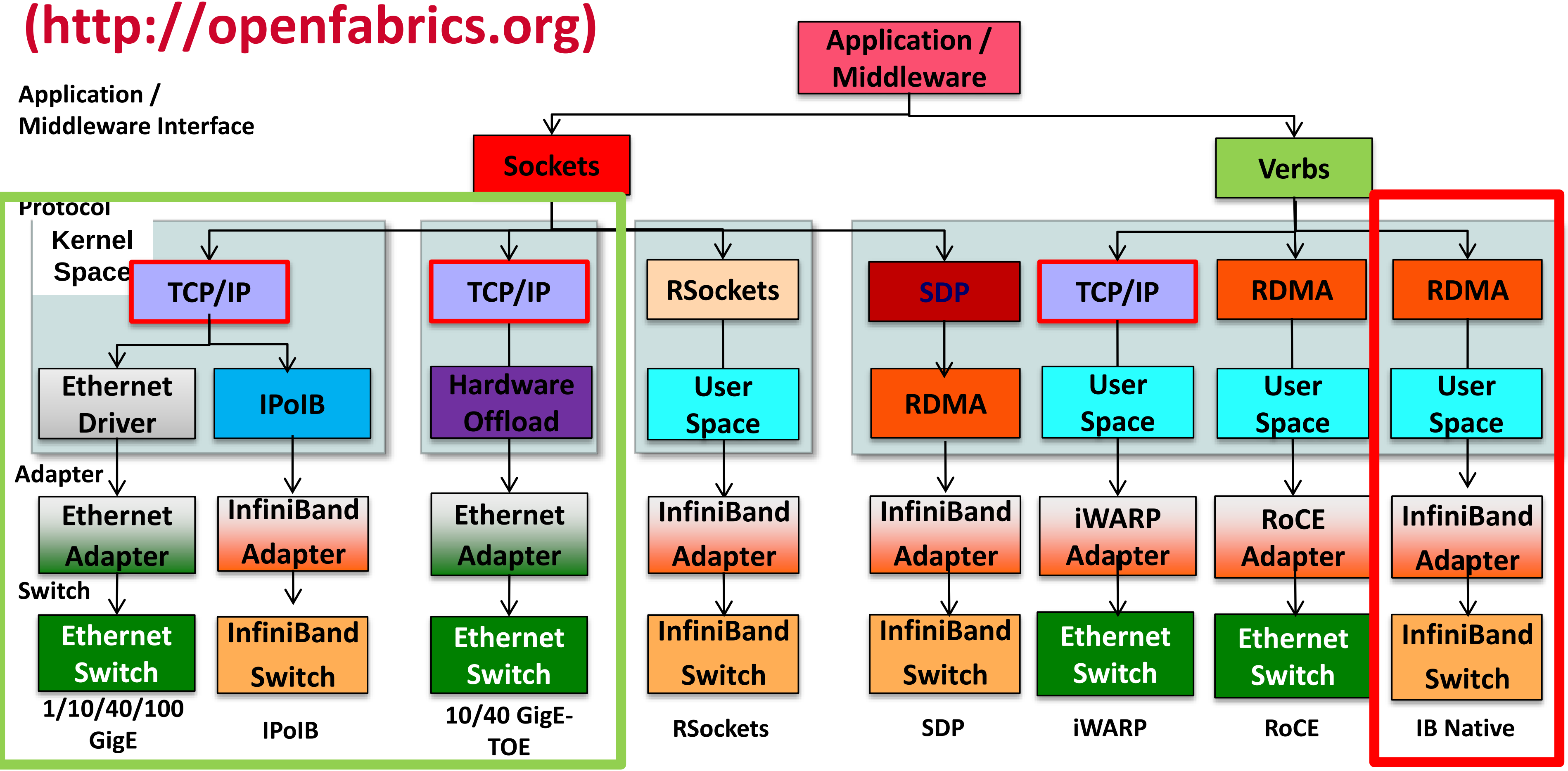
ORACLE Cloud

Chameleon Cloud

Trends in HPC Technologies

- Advanced Interconnects and RDMA protocols
  - InfiniBand
  - Ethernet/iWARP
  - RDMA over Converged Enhanced Ethernet (RoCE)
- Delivering excellent performance (Latency, Bandwidth and CPU Utilization)
- Has influenced re-designs of enhanced HPC middleware
  - Message Passing Interface (MPI) and PGAS
  - Parallel File Systems (Lustre, GPFS, ..)
- SSDs (SATA and NVMe)
- NVRAM and Burst Buffer

Interconnects and Protocols in OpenFabrics Stack for HPC (http://openfabrics.org)



Large-scale InfiniBand Installations

- 187 IB Clusters (37%) in the Nov’16 Top500 list
  - (http://www.top500.org)
- Installations in the Top 50 (15 systems):

241,108 cores (Pleiades) at NASA/Ames (13th)	147,456 cores (SuperMUC) in Germany (36th)
220,800 cores (Pangea) in France (16th)	86,016 cores (SuperMUC Phase 2) in Germany (37th)
462,462 cores (Stampede) at TACC (17th)	74,520 cores (Tsubame 2.5) at Japan/GSIC (40th)
144,900 cores (Cheyenne) at NCAR/USA (20th)	194,616 cores (Cascade) at PNNL (44th)
72,800 cores Cray CS-Storm in US (25th)	76,032 cores (Makman-2) at Saudi Aramco (49th)
72,800 cores Cray CS-Storm in US (26th)	72,000 cores (Prolix) at Meteo France, France (50th)
124,200 cores (Topaz) SGI ICE at ERDC DSRC in US (27th)	73,440 cores (Beaufix2) at Meteo France, France (51st)
60,512 cores (DGX SATURNV) at NVIDIA/USA (28th)	42,688 cores (Lomonosov-2) at Russia/MSU (52nd)
72,000 cores (HPC2) in Italy (29th)	60,240 cores SGI ICE X at JAEA Japan (54th)
152,692 cores (Thunder) at AFRL/USA (32nd)	**and many more!**

Open Standard InfiniBand Networking Technology

- Introduced in Oct 2000
- High Performance Data Transfer
  - Interprocessor communication and I/O
  - Low latency (<1.0 microsec), High bandwidth (up to 12.5 GigaBytes/sec -> 100Gbps), and low CPU utilization (5-10%)
- Multiple Operations
  - Send/Recv
  - RDMA Read/Write
  - Atomic Operations (very unique)
    - high performance and scalable implementations of distributed locks, semaphores, collective communication operations
- Leading to big changes in designing
  - HPC clusters
  - File systems
  - Cloud computing systems
  - Grid computing systems

Overview of the MVAPICH2 Project

- High Performance open-source MPI Library for InfiniBand, Omni-Path, Ethernet/iWARP, and RDMA over Converged Ethernet (RoCE)
  - MVAPICH (MPI-1), MVAPICH2 (MPI-2.2 and MPI-3.0), Started in 2001, First version available in 2002
  - MVAPICH2-X (MPI + PGAS), Available since 2011
  - Support for GPGPUs (MVAPICH2-GDR) and MIC (MVAPICH2-MIC), Available since 2014
  - Support for Virtualization (MVAPICH2-Virt), Available since 2015
  - Support for Energy-Awareness (MVAPICH2-EA), Available since 2015
  - Support for InfiniBand Network Analysis and Monitoring (OSU INAM) since 2015
  - **Used by more than 2,725 organizations in 83 countries**
  - **More than 408,000 (> 0.4 million) downloads from the OSU site directly**
  - Empowering many TOP500 clusters (Nov '16 ranking)
    - 1st, 10,649,600-core (Sunway TaihuLight) at National Supercomputing Center in Wuxi, China
    - 13th, 241,108-core (Pleiades) at NASA
    - 17th, 462,462-core (Stampede) at TACC
    - 40th, 74,520-core (Tsubame 2.5) at Tokyo Institute of Technology
  - Available with software stacks of many vendors and Linux Distros (RedHat and SuSE)
  - http://mvapich.cse.ohio-state.edu
- Empowering Top500 systems for over a decade
  - System-X from Virginia Tech (3rd in Nov 2003, 2,200 processors, 12.25 TFlops) ->
  - Stampede at TACC (12th in Jun'16, 462,462 cores, 5.168 Plops)

How Can HPC Clusters with High-Performance Interconnect and Storage Architectures Benefit Big Data Applications?

What are the major bottlenecks in current Big Data processing middleware (e.g. Hadoop, Spark, and Memcached)?

Can the bottlenecks be alleviated with new designs by taking advantage of HPC technologies?

Can RDMA-enabled high-performance interconnects benefit Big Data processing?

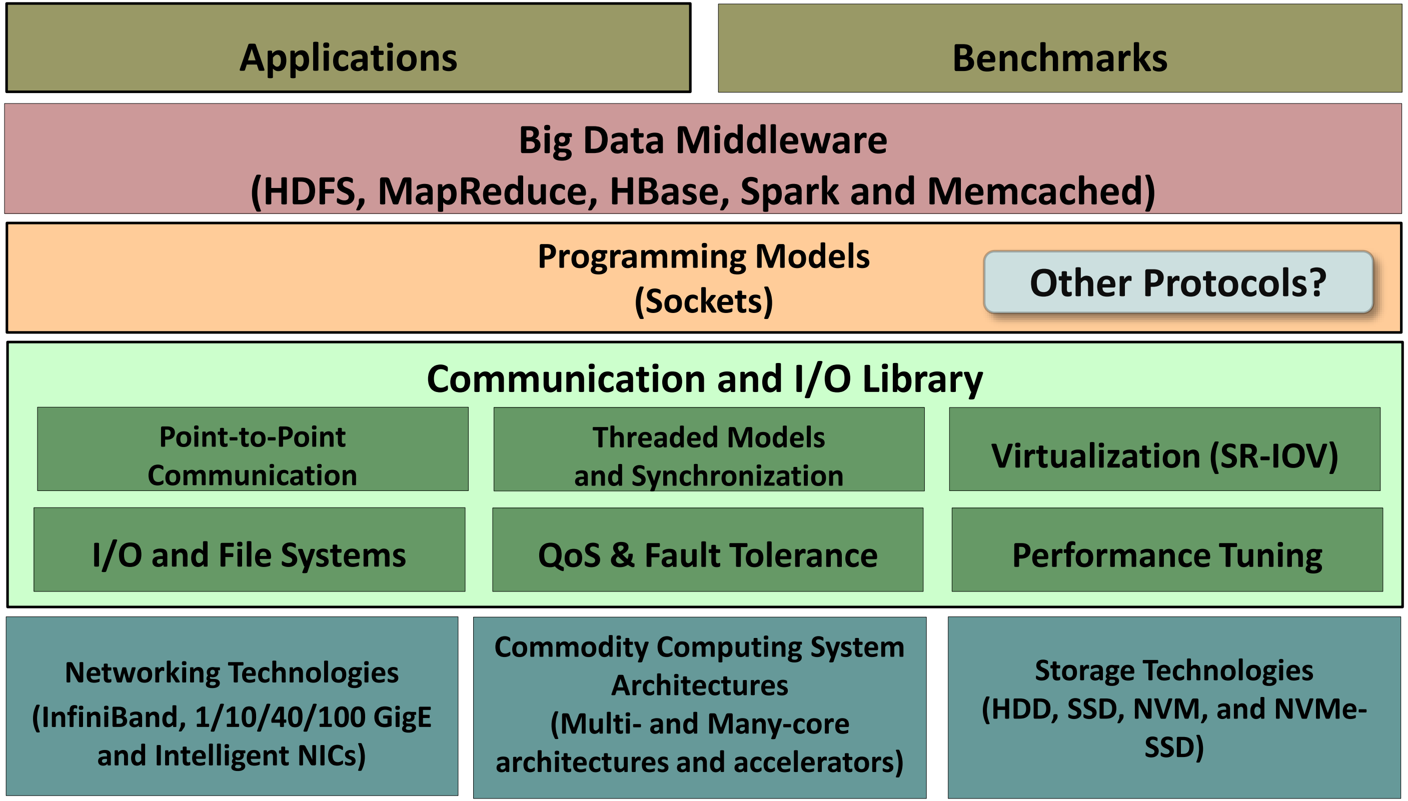
Can HPC Clusters with high-performance storage systems (e.g. SSD, parallel file systems) benefit Big Data applications?

How much performance benefits can be achieved through enhanced designs?

How to design benchmarks for evaluating the performance of Big Data middleware on HPC clusters?

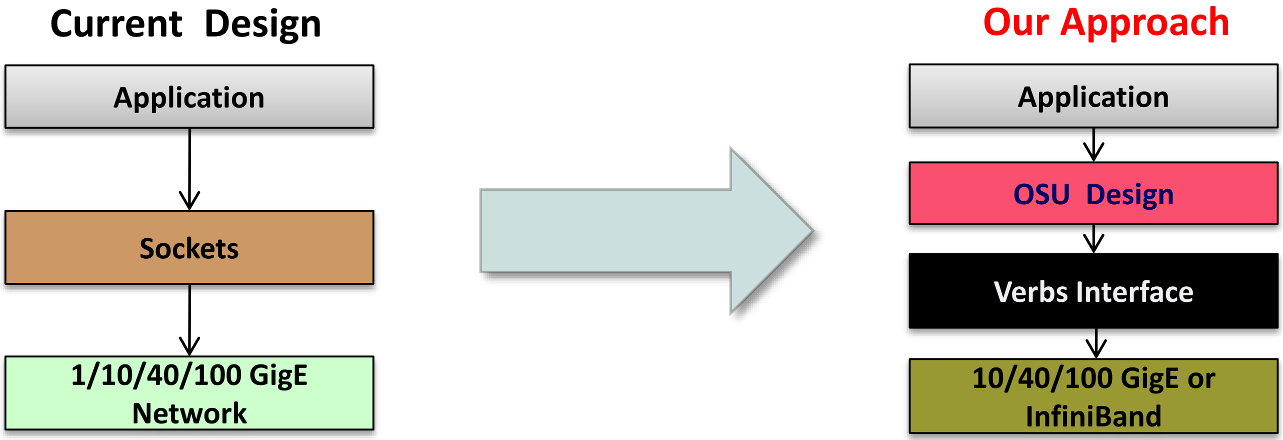
Bring HPC and Big Data processing into a "convergent trajectory"!

Designing Communication and I/O Libraries for Big Data Systems: Challenges



Applications
Benchmarks
Big Data Middleware
(HDFS, MapReduce, HBase, Spark and Memcached)
Programming Models
(Sockets)
Other Protocols?
Communication and I/O Library
Point-to-Point Communication
Threaded Models and Synchronization
Virtualization (SR-IOV)
I/O and File Systems
QoS & Fault Tolerance
Performance Tuning
Networking Technologies
(InfiniBand, 1/10/40/100 GigE and Intelligent NICs)
Commodity Computing System Architectures
(Multi- and Many-core architectures and accelerators)
Storage Technologies
(HDD, SSD, NVM, and NVMe-SSD)

Can Big Data Processing Systems be Designed with High-Performance Networks and Protocols?



- Sockets not designed for high-performance
  - Stream semantics often mismatch for upper layers
  - Zero-copy not available for non-blocking sockets

The High-Performance Big Data (HiBD) Project

- RDMA for Apache Spark
- RDMA for Apache Hadoop 2.x (RDMA-Hadoop-2.x)
  - Plugins for Apache, Hortonworks (HDP) and Cloudera (CDH) Hadoop distributions
- RDMA for Apache HBase
- RDMA for Memcached (RDMA-Memcached)
- RDMA for Apache Hadoop 1.x (RDMA-Hadoop)
- OSU HiBD-Benchmarks (OHB)
  - HDFS, Memcached, HBase, and Spark Micro-benchmarks
- http://hibd.cse.ohio-state.edu
- Users Base: 205 organizations from 29 countries
- More than 19,700 downloads from the project site

Available for InfiniBand and RoCE

HiBD
High-Performance
Big Data

Network Based Computing
Laboratory

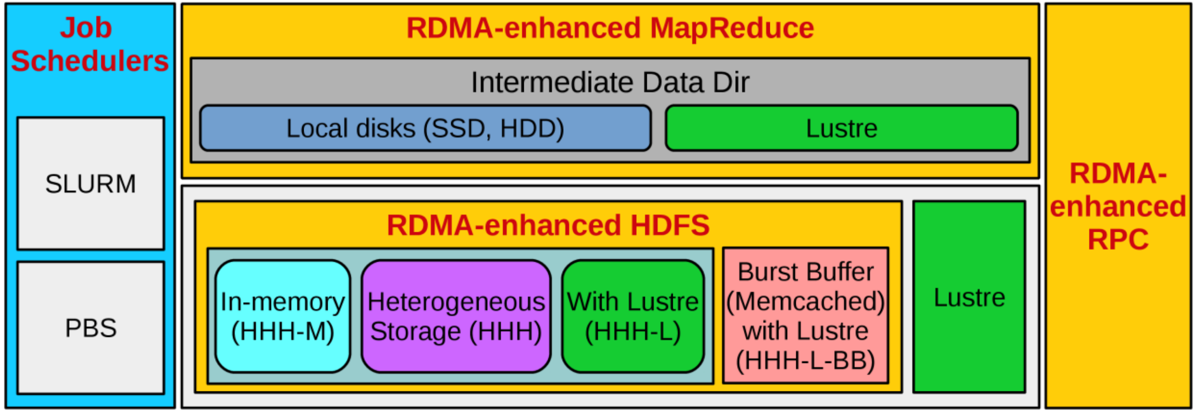
The Ohio State University

RDMA for Apache Hadoop 2.x Distribution

- High-Performance Design of Hadoop over RDMA-enabled Interconnects
  - High performance RDMA-enhanced design with native InfiniBand and RoCE support at the verbs-level for HDFS, MapReduce, and RPC components
  - Enhanced HDFS with in-memory and heterogeneous storage
  - High performance design of MapReduce over Lustre
  - Memcached-based burst buffer for MapReduce over Lustre-integrated HDFS (HHH-L-BB mode)
  - Plugin-based architecture supporting RDMA-based designs for Apache Hadoop, CDH and HDP
  - Easily configurable for different running modes (HHH, HHH-M, HHH-L, HHH-L-BB, and MapReduce over Lustre) and different protocols (native InfiniBand, RoCE, and IPoIB)
- Current release: 1.1.0
  - Based on Apache Hadoop 2.7.3
  - Compliant with Apache Hadoop 2.7.1, HDP 2.5.0.3 and CDH 5.8.2 APIs and applications
  - Tested with
    - Mellanox InfiniBand adapters (DDR, QDR, FDR, and EDR)
    - RoCE support with Mellanox adapters
    - Various multi-core platforms
    - Different file systems with disks and SSDs and Lustre

http://hibd.cse.ohio-state.edu

Different Modes of RDMA for Apache Hadoop 2.x



- **HHH**: Heterogeneous storage devices with hybrid replication schemes are supported in this mode of operation to have better fault-tolerance as well as performance. This mode is enabled by **default** in the package.
- **HHH-M**: A high-performance in-memory based setup has been introduced in this package that can be utilized to perform all I/O operations in-memory and obtain as much performance benefit as possible.
- **HHH-L**: With parallel file systems integrated, HHH-L mode can take advantage of the Lustre available in the cluster.
- **HHH-L-BB**: This mode deploys a Memcached-based burst buffer system to reduce the bandwidth bottleneck of shared file system access. The burst buffer design is hosted by Memcached servers, each of which has a local SSD.
- **MapReduce over Lustre, with/without local disks**: Besides, HDFS based solutions, this package also provides support to run MapReduce jobs on top of Lustre alone. Here, two different modes are introduced: with local disks and without local disks.
- **Running with Slurm and PBS**: Supports deploying RDMA for Apache Hadoop 2.x with Slurm and PBS in different running modes (HHH, HHH-M, HHH-L, and MapReduce over Lustre).

RDMA for Apache Spark Distribution

- High-Performance Design of Spark over RDMA-enabled Interconnects
  - RDMA-enhanced design with native InfiniBand and RoCE support at the verbs-level for Spark
  - RDMA-based data shuffle and SEDA-based shuffle architecture
  - Non-blocking and chunk-based data transfer
  - RDMA support for Spark SQL
  - Integration with HHH in RDMA for Apache Hadoop
  - Easily configurable for different protocols (native InfiniBand, RoCE, and IPoIB)
- Current release: 0.9.3
  - Based on Apache Spark 1.5.1
  - Tested with
    - Mellanox InfiniBand adapters (DDR, QDR and FDR)
    - RoCE support with Mellanox adapters
    - Various multi-core platforms
    - RAM disks, SSDs, and HDD
  - http://hibd.cse.ohio-state.edu

HiBD Packages on SDSC Comet and Chameleon Cloud

- RDMA for Apache Hadoop 2.x and RDMA for Apache Spark are installed and available on SDSC Comet.
  - Examples for various modes of usage are available in:
    - RDMA for Apache Hadoop 2.x: /share/apps/examples/HADOOP
    - RDMA for Apache Spark: /share/apps/examples/SPARK/
  - Please email help@xsede.org (reference Comet as the machine, and SDSC as the site) if you have any further questions about usage and configuration.
- RDMA for Apache Hadoop is also available on Chameleon Cloud as an appliance
  - https://www.chameleoncloud.org/appliances/17/

M. Tatineni, X. Lu, D. J. Choi, A. Majumdar, and D. K. Panda, Experiences and Benefits of Running RDMA Hadoop and Spark on SDSC Comet, XSEDE'16, July 2016

RDMA for Apache HBase Distribution

- High-Performance Design of HBase over RDMA-enabled Interconnects
  - High performance RDMA-enhanced design with native InfiniBand and RoCE support at the verbs-level for HBase
  - Compliant with Apache HBase 1.1.2 APIs and applications
  - On-demand connection setup
  - Easily configurable for different protocols (native InfiniBand, RoCE, and IPoIB)
- Current release: 0.9.1
  - Based on Apache HBase 1.1.2
  - Tested with
    - Mellanox InfiniBand adapters (DDR, QDR, FDR, and EDR)
    - RoCE support with Mellanox adapters
    - Various multi-core platforms
  - http://hibd.cse.ohio-state.edu

RDMA for Memcached Distribution

- High-Performance Design of Memcached over RDMA-enabled Interconnects
  - High performance RDMA-enhanced design with native InfiniBand and RoCE support at the verbs-level for Memcached and libMemcached components
  - High performance design of SSD-Assisted Hybrid Memory
  - Non-Blocking Libmemcached Set/Get API extensions
  - Support for burst-buffer mode in Lustre-integrated design of HDFS in RDMA for Apache Hadoop-2.x
  - Easily configurable for native InfiniBand, RoCE and the traditional sockets-based support (Ethernet and InfiniBand with IPoIB)
- Current release: 0.9.5
  - Based on Memcached 1.4.24 and libMemcached 1.0.18
  - Compliant with libMemcached APIs and applications
  - Tested with
    - Mellanox InfiniBand adapters (DDR, QDR, FDR, and EDR)
    - RoCE support with Mellanox adapters
    - Various multi-core platforms
    - SSD
  - http://hibd.cse.ohio-state.edu

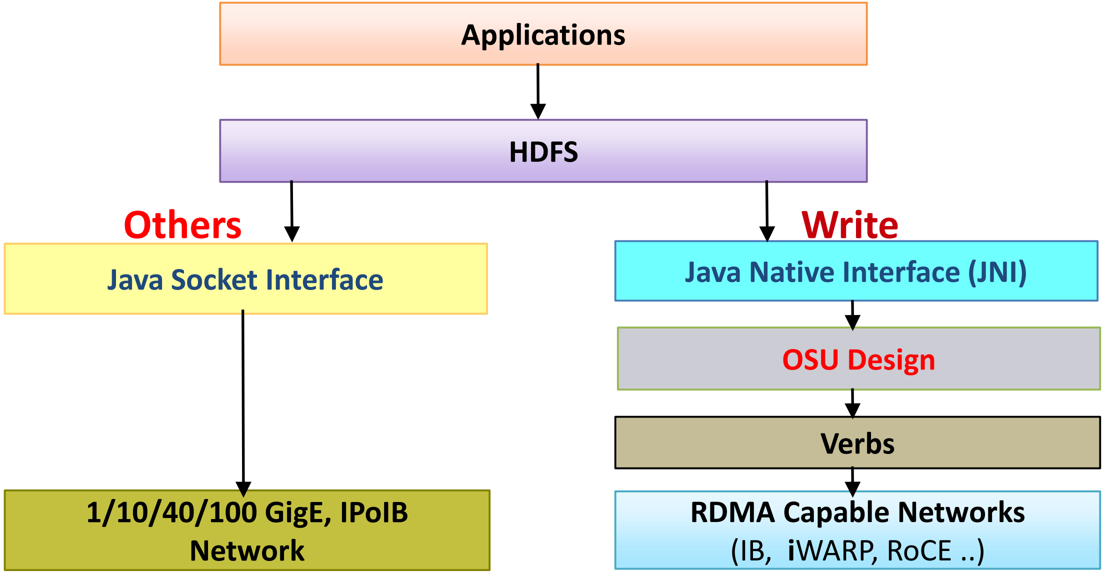
OSU HiBD Micro-Benchmark (OHB) Suite – HDFS, Memcached, HBase, and Spark

- Micro-benchmarks for Hadoop Distributed File System (HDFS)
  - Sequential Write Latency (**SWL**) Benchmark, Sequential Read Latency (**SRL**) Benchmark, Random Read Latency (**RRL**) Benchmark, Sequential Write Throughput (**SWT**) Benchmark, Sequential Read Throughput (**SRT**) Benchmark
  - Support benchmarking of
    - Apache Hadoop 1.x and 2.x HDFS, Hortonworks Data Platform (HDP) HDFS, Cloudera Distribution of Hadoop (CDH) HDFS
- Micro-benchmarks for Memcached
  - **Get** Benchmark, **Set** Benchmark, and **Mixed** Get/Set Benchmark, **Non-Blocking API** Latency Benchmark**, Hybrid Memory** Latency Benchmark
- Micro-benchmarks for HBase
  - **Get** Latency Benchmark, **Put** Latency Benchmark
- Micro-benchmarks for Spark
  - GroupBy, SortBy
- Current release: 0.9.2
- **http://hibd.cse.ohio-state.edu**

Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

Design Overview of HDFS with RDMA



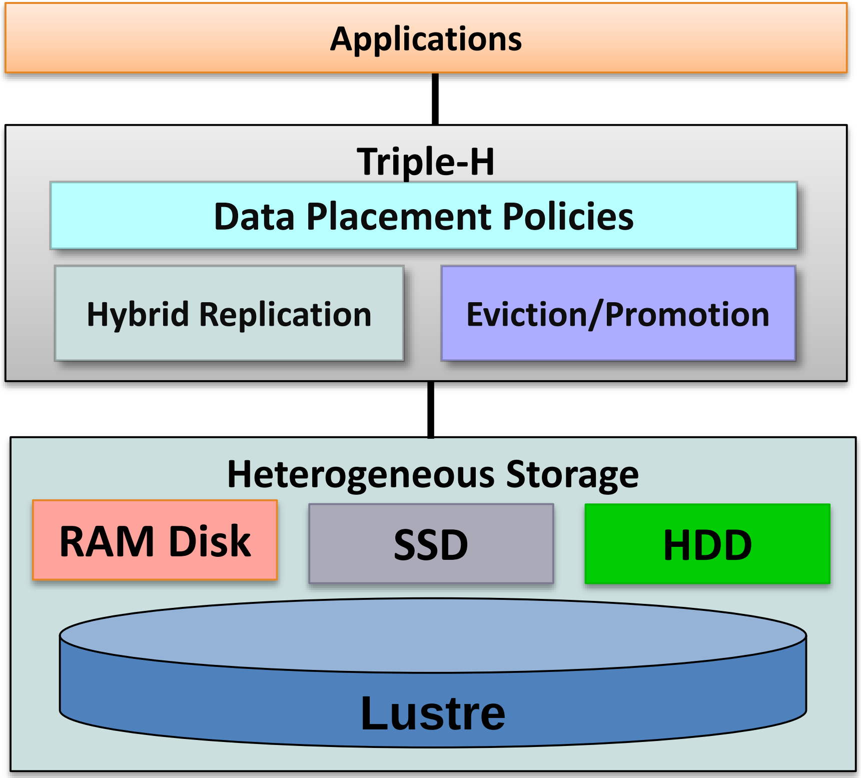
- Design Features
  - RDMA-based HDFS write
  - RDMA-based HDFS replication
  - Parallel replication support
  - On-demand connection setup
  - InfiniBand/RoCE support

- Enables high performance RDMA communication, while supporting traditional socket interface
- JNI Layer bridges Java based HDFS with communication library written in native code

N. S. Islam, M. W. Rahman, J. Jose, R. Rajachandrasekar, H. Wang, H. Subramoni, C. Murthy and D. K. Panda , High Performance RDMA-Based Design of HDFS over InfiniBand , Supercomputing (SC), Nov 2012

N. Islam, X. Lu, W. Rahman, and D. K. Panda, SOR-HDFS: A SEDA-based Approach to Maximize Overlapping in RDMA-Enhanced HDFS, HPDC '14, June 2014

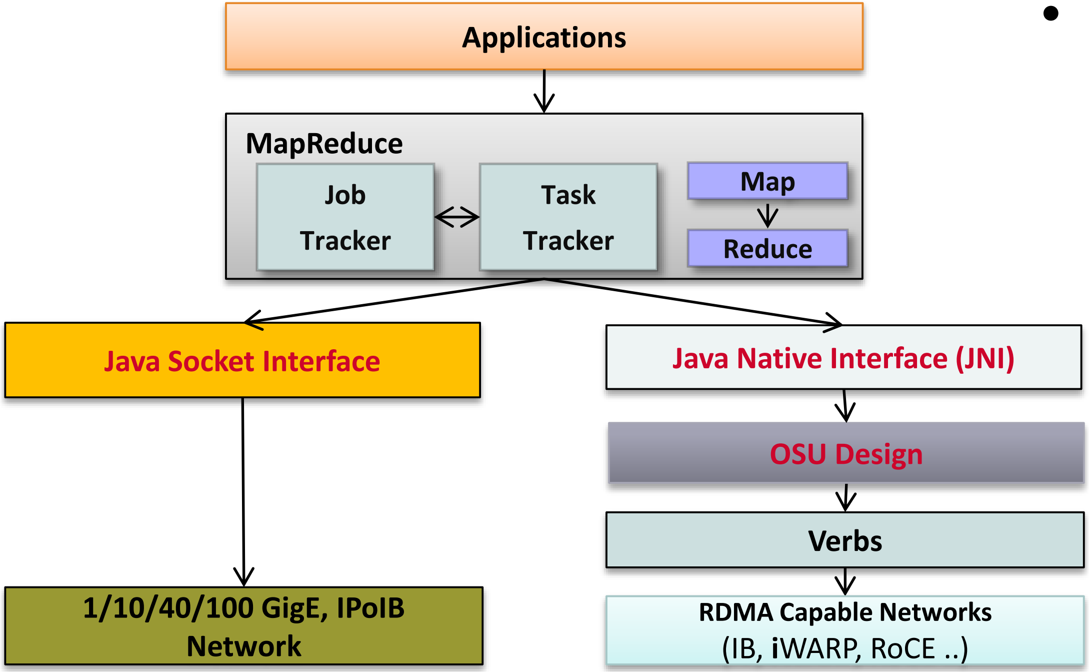
Enhanced HDFS with In-Memory and Heterogeneous Storage



- Design Features
  - Three modes
    - Default (HHH)
    - In-Memory (HHH-M)
    - Lustre-Integrated (HHH-L)
  - Policies to efficiently utilize the heterogeneous storage devices
    - RAM, SSD, HDD, Lustre
  - Eviction/Promotion based on data usage pattern
  - Hybrid Replication
  - Lustre-Integrated mode:
    - Lustre-based fault-tolerance

N. Islam, X. Lu, M. W. Rahman, D. Shankar, and D. K. Panda, Triple-H: A Hybrid Approach to Accelerate HDFS on HPC Clusters with Heterogeneous Storage Architecture, CCGrid ’15, May 2015

Design Overview of MapReduce with RDMA

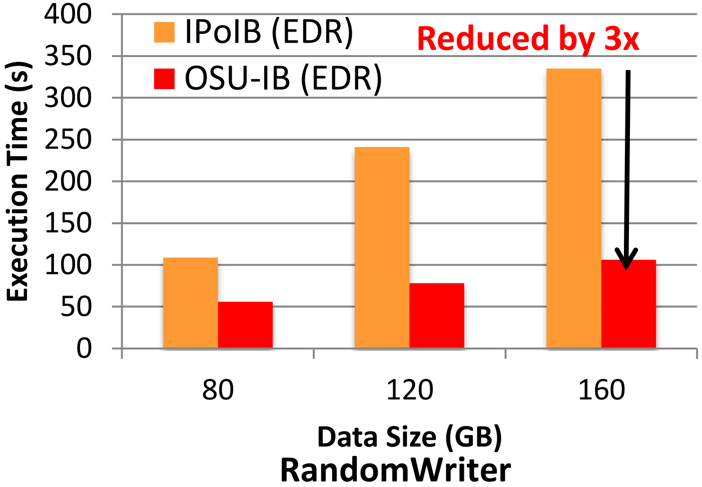

Applications
MapReduce
Job Tracker
Task Tracker
Map
Reduce
Java Socket Interface
Java Native Interface (JNI)
OSU Design
Verbs
1/10/40/100 GigE, IPoIB Network
RDMA Capable Networks (IB, iWARP, RoCE ..)

- Design Features
  - RDMA-based shuffle
  - Prefetching and caching map output
  - Efficient Shuffle Algorithms
  - In-memory merge
  - On-demand Shuffle Adjustment
  - Advanced overlapping
    - map, shuffle, and merge
    - shuffle, merge, and reduce
  - On-demand connection setup
  - InfiniBand/RoCE support

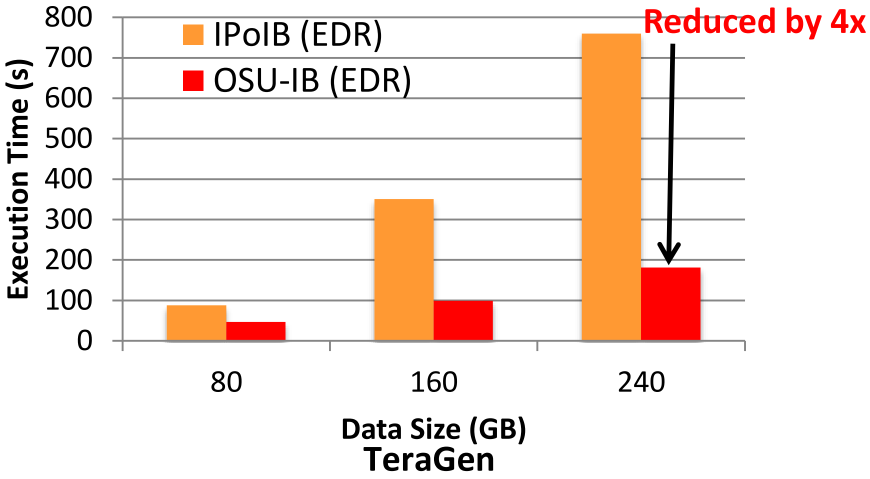
- Enables high performance RDMA communication, while supporting traditional socket interface
- JNI Layer bridges Java based MapReduce with communication library written in native code

M. W. Rahman, X. Lu, N. S. Islam, and D. K. Panda, HOMR: A Hybrid Approach to Exploit Maximum Overlapping in MapReduce over High Performance Interconnects, ICS, June 2014

Performance Numbers of RDMA for Apache Hadoop 2.x – RandomWriter & TeraGen in OSU-RI2 (EDR)



RandomWriter



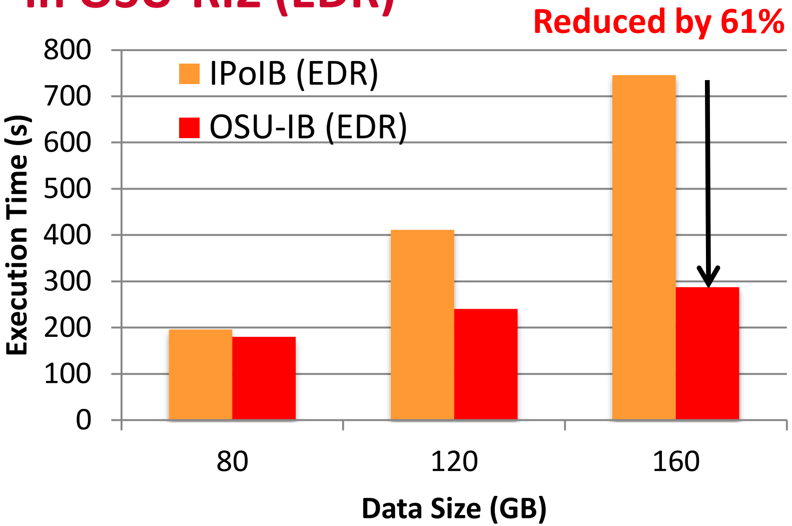
TeraGen

Cluster with 8 Nodes with a total of 64 maps

- RandomWriter
  - 3x improvement over IPoIB for 80-160 GB file size

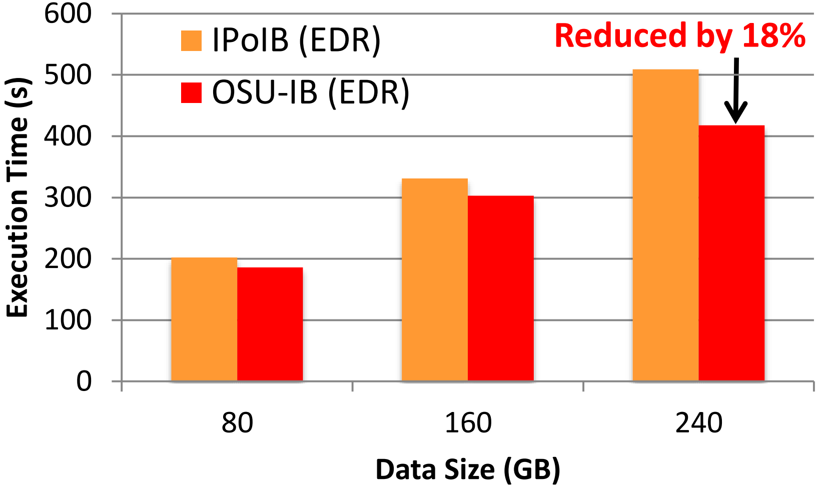
- TeraGen
  - 4x improvement over IPoIB for 80-240 GB file size

Performance Numbers of RDMA for Apache Hadoop 2.x – Sort & TeraSort in OSU-RI2 (EDR)



Sort

Cluster with 8 Nodes with a total of 64 maps and 14 reduces



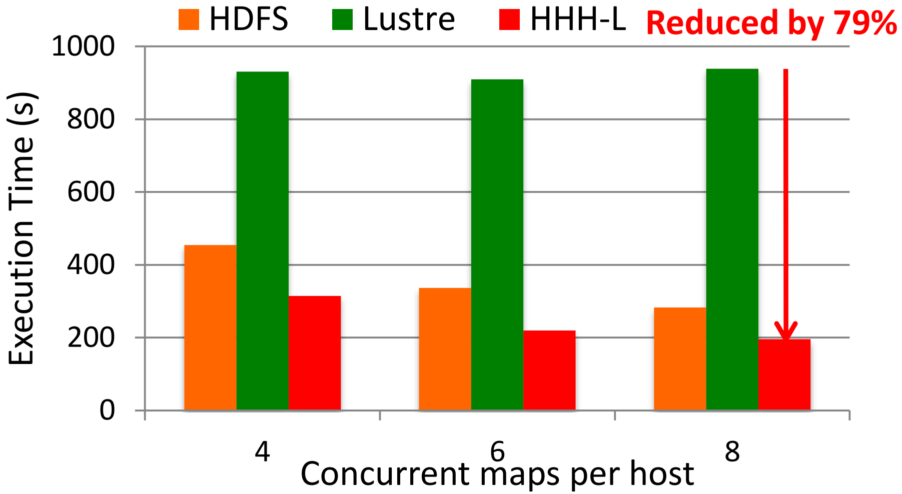
TeraSort

Cluster with 8 Nodes with a total of 64 maps and 32 reduces

- Sort
  - **61%** improvement over IPoIB for 80-160 GB data

- TeraSort
  - **18%** improvement over IPoIB for 80-240 GB data

Evaluation of HHH and HHH-L with Applications



Reduced by 79%

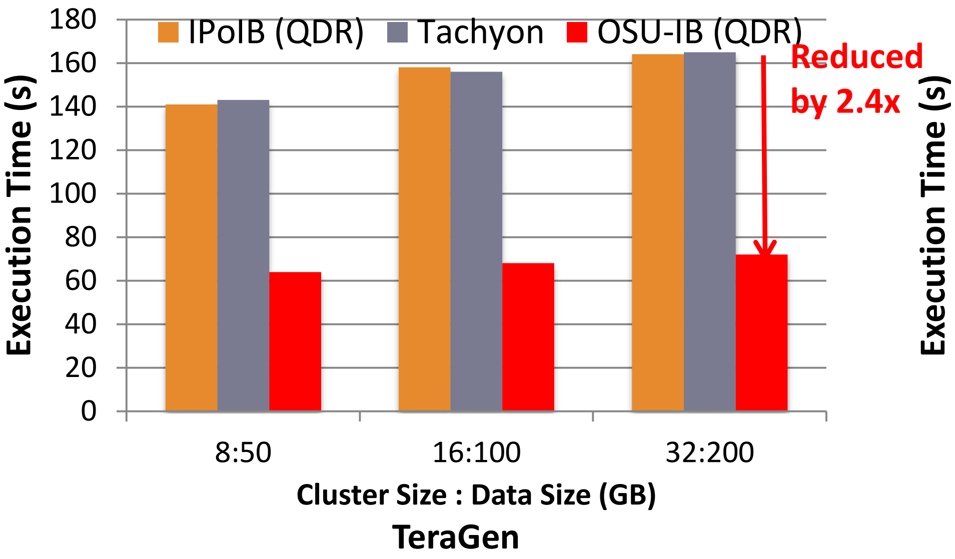
MR-MSPolyGraph

HDFS (FDR)	HHH (FDR)
60.24 s	48.3 s

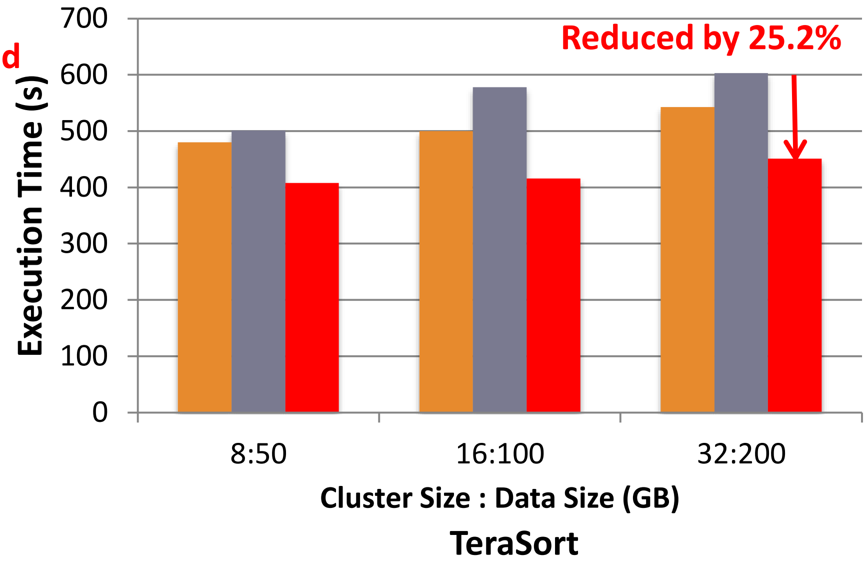
CloudBurst

- MR-MSPolygraph on OSU RI with 1,000 maps
  - HHH-L reduces the execution time by **79%** over Lustre, **30%** over HDFS
- CloudBurst on TACC Stampede
  - With HHH: **19%** improvement over HDFS

Evaluation with Spark on SDSC Gordon (HHH vs. Tachyon/Alluxio)



TeraGen



TeraSort

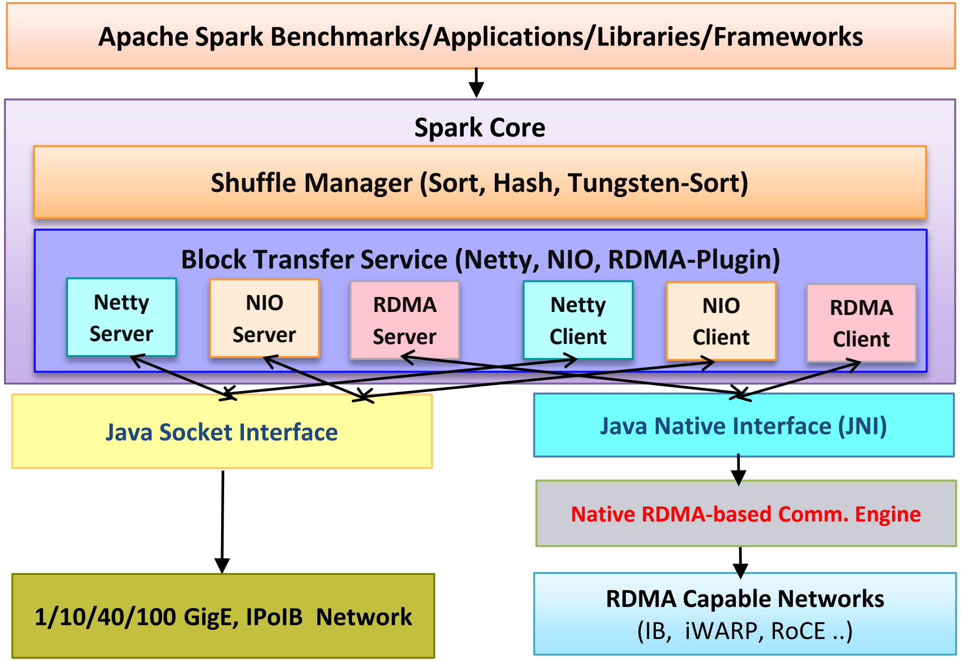
- For 200GB TeraGen on 32 nodes
  - Spark-TeraGen: HHH has 2.4x improvement over Tachyon; 2.3x over HDFS-IPoIB (QDR)
  - Spark-TeraSort: HHH has 25.2% improvement over Tachyon; 17% over HDFS-IPoIB (QDR)

N. Islam, M. W. Rahman, X. Lu, D. Shankar, and D. K. Panda, Performance Characterization and Acceleration of In-Memory File Systems for Hadoop and Spark Applications on HPC Clusters, IEEE BigData ’15, October 2015

Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

Design Overview of Spark with RDMA



Apache Spark Benchmarks/Applications/Libraries/Frameworks
Spark Core
Shuffle Manager (Sort, Hash, Tungsten-Sort)
Block Transfer Service (Netty, NIO, RDMA-Plugin)
Netty Server
NIO Server
RDMA Server
Netty Client
NIO Client
RDMA Client
Java Socket Interface
Java Native Interface (JNI)
Native RDMA-based Comm. Engine
1/10/40/100 GigE, IPoIB Network
RDMA Capable Networks
(IB, iWARP, RoCE ..)

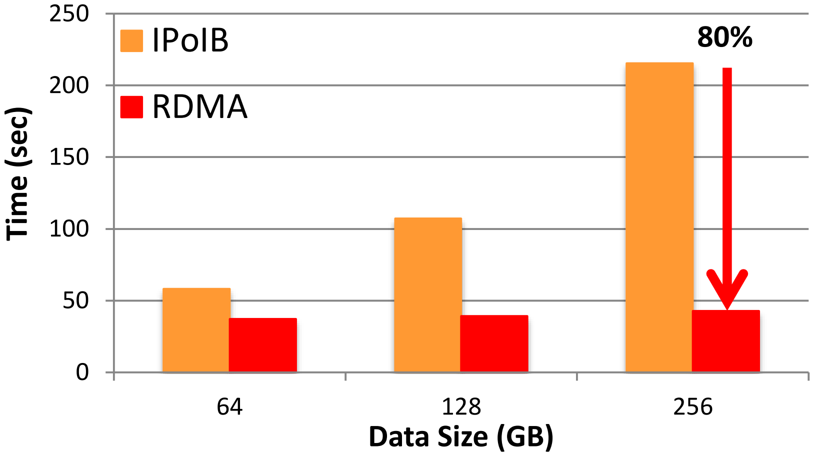
- Design Features
  - RDMA based shuffle plugin
  - SEDA-based architecture
  - Dynamic connection management and sharing
  - Non-blocking data transfer
  - Off-JVM-heap buffer management
  - InfiniBand/RoCE support

- Enables high performance RDMA communication, while supporting traditional socket interface
- JNI Layer bridges Scala based Spark with communication library written in native code

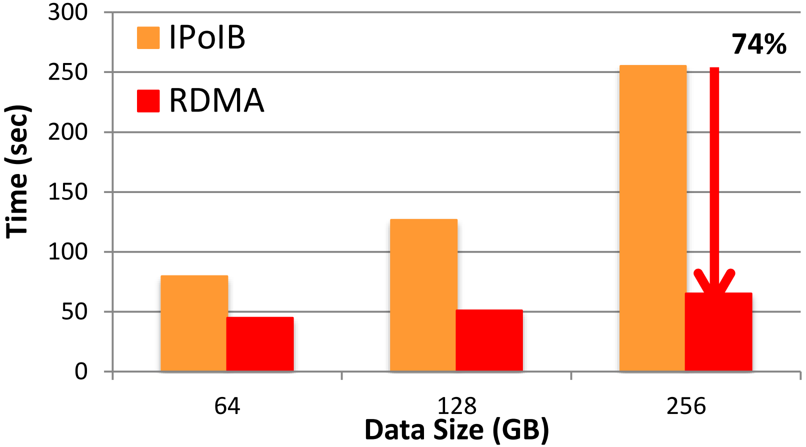
X. Lu, M. W. Rahman, N. Islam, D. Shankar, and D. K. Panda, Accelerating Spark with RDMA for Big Data Processing: Early Experiences, Int'l Symposium on High Performance Interconnects (HotI'14), August 2014

X. Lu, D. Shankar, S. Gugnani, and D. K. Panda, High-Performance Design of Apache Spark with RDMA and Its Benefits on Various Workloads, IEEE BigData '16, Dec. 2016.

Performance Evaluation on SDSC Comet – SortBy/GroupBy



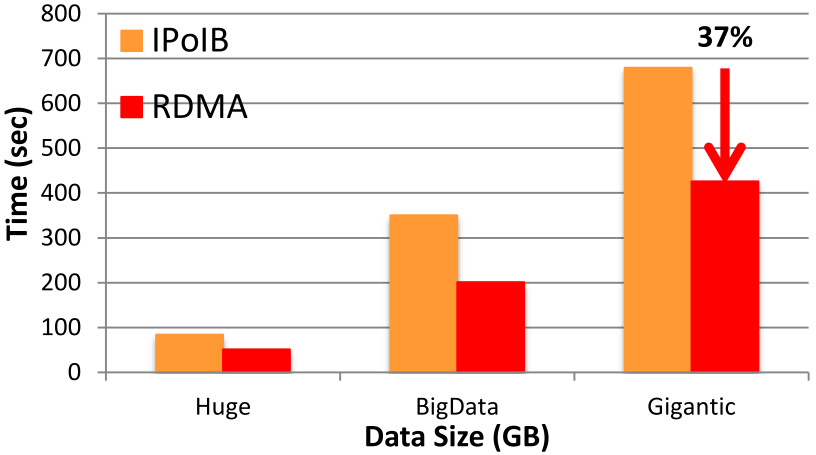
64 Worker Nodes, 1536 cores, SortByTest Total Time



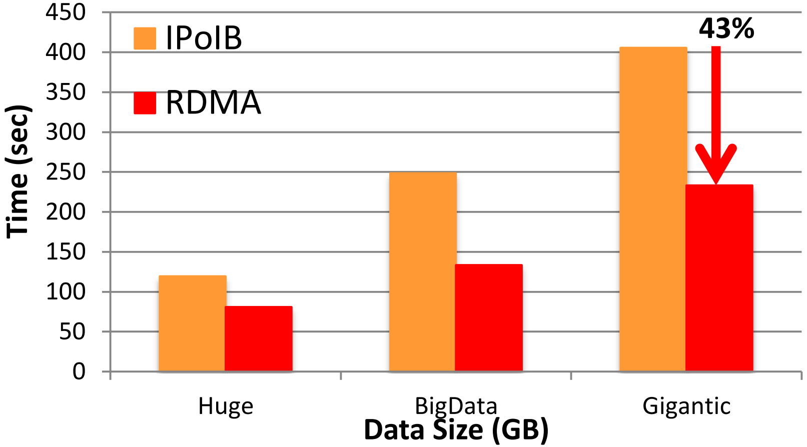
64 Worker Nodes, 1536 cores, GroupByTest Total Time

- InfiniBand FDR, SSD, 64 Worker Nodes, 1536 Cores, (1536M 1536R)
- RDMA-based design for Spark 1.5.1
- RDMA vs. IPoIB with 1536 concurrent tasks, single SSD per node.
  - SortBy: Total time reduced by up to 80% over IPoIB (56Gbps)
  - GroupBy: Total time reduced by up to 74% over IPoIB (56Gbps)

Performance Evaluation on SDSC Comet – HiBench PageRank



32 Worker Nodes, 768 cores, PageRank Total Time



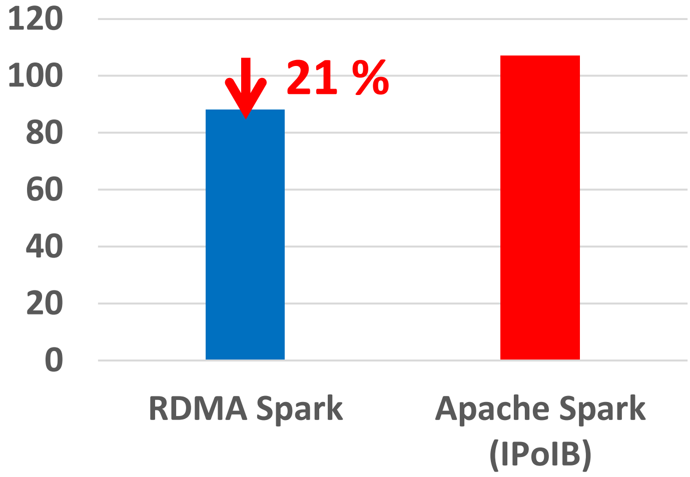
64 Worker Nodes, 1536 cores, PageRank Total Time

- InfiniBand FDR, SSD, 32/64 Worker Nodes, 768/1536 Cores, (768/1536M 768/1536R)
- RDMA-based design for Spark 1.5.1
- RDMA vs. IPoIB with 768/1536 concurrent tasks, single SSD per node.
  - 32 nodes/768 cores: Total time reduced by 37% over IPoIB (56Gbps)
  - 64 nodes/1536 cores: Total time reduced by 43% over IPoIB (56Gbps)

Performance Evaluation on SDSC Comet: Astronomy Application

- **Kira Toolkit[1]:** Distributed astronomy image processing toolkit implemented using Apache Spark.
- Source extractor application, using a 65GB dataset from the SDSS DR2 survey that comprises 11,150 image files.
- Compare RDMA Spark performance with the standard apache implementation using IPoIB.

[1] **Z. Zhang, K. Barbary, F. A. Nothaft, E.R. Sparks, M.J. Franklin, D.A. Patterson, S. Perlmutter. Scientific Computing meets Big Data Technology: An Astronomy Use Case. *CoRR, vol: abs/1507.03325*, Aug 2015.**



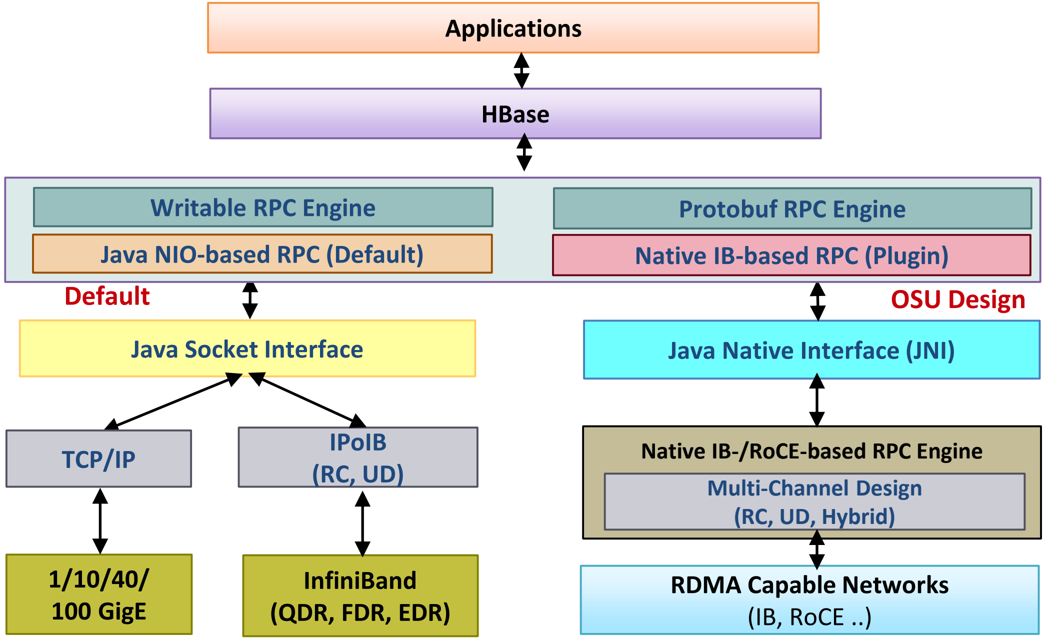
Execution times (sec) for Kira SE benchmark using 65 GB dataset, 48 cores.

M. Tatineni, X. Lu, D. J. Choi, A. Majumdar, and D. K. Panda, Experiences and Benefits of Running RDMA Hadoop and Spark on SDSC Comet, XSEDE'16, July 2016

Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

Overview of IB-based Hadoop-RPC and HBase Architecture



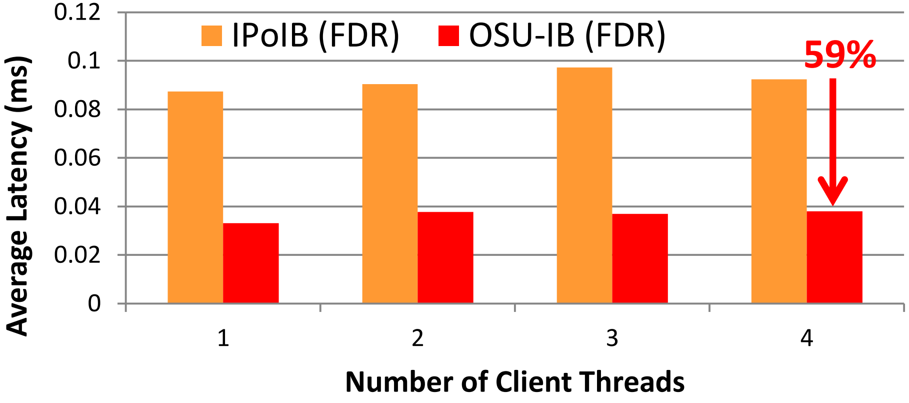
- Design Features
  - RDMA or send/recv based adaptive communication
  - SEDA-based Thread Management
  - Support RC, UD, and Hybrid transport protocols
  - Architecture-aware designs for Eager, packetized, and zero-copy transfers
  - JVM-bypassed buffer management
  - Intelligent buffer allocation and adjustment for serialization
  - InfiniBand/RoCE support for bare-metal and SR-IOV
  - On-demand Connection Management
  - Integrated design with HBase

J. Huang, X. Ouyang, J. Jose, M. W. Rahman, H. Wang, M. Luo, H. Subramoni, Chet Murthy, and D. K. Panda, High-Performance Design of HBase with RDMA over InfiniBand, IPDPS'12

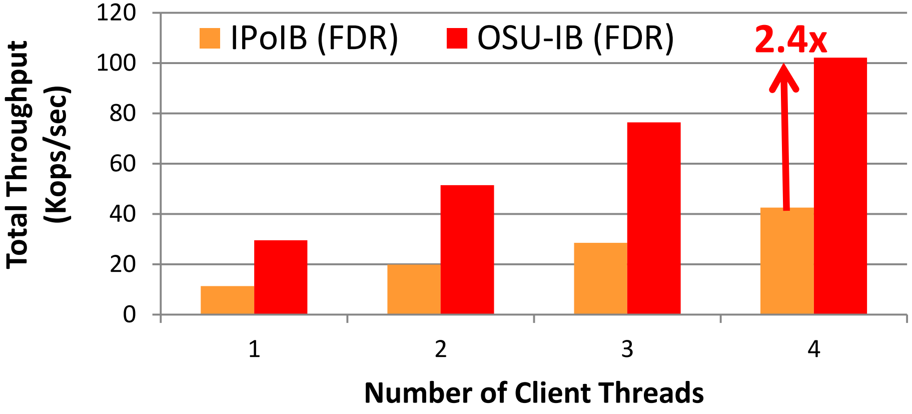
X. Lu, N. Islam, M. W. Rahman, J. Jose, H. Subramoni, H. Wang, and D. K. Panda, High-Performance Design of Hadoop RPC with RDMA over InfiniBand, ICPP '13, October 2013.

X. Lu, D. Shankar, S. Gugnani, H. Subramoni, and D. K. Panda, Impact of HPC Cloud Networking Technologies on Accelerating Hadoop RPC and HBase, CloudCom, 2016.

HBase – YCSB Get Latency and Throughput on SDSC-Comet



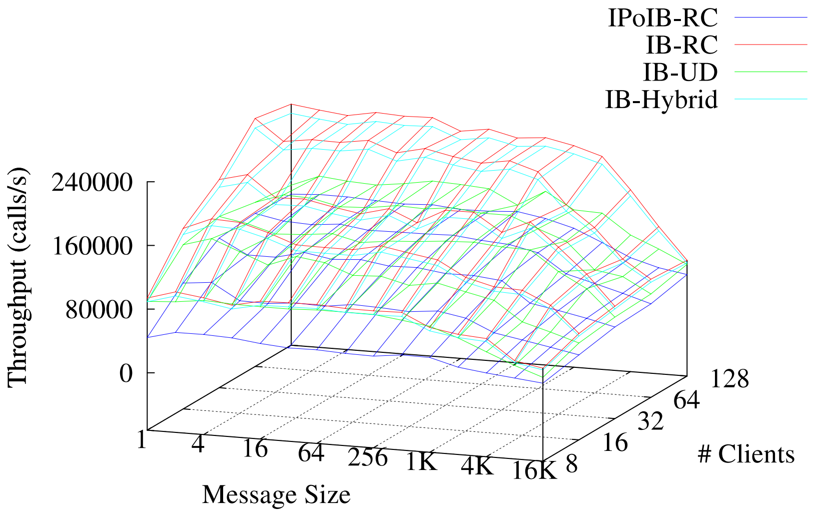
Get Latency



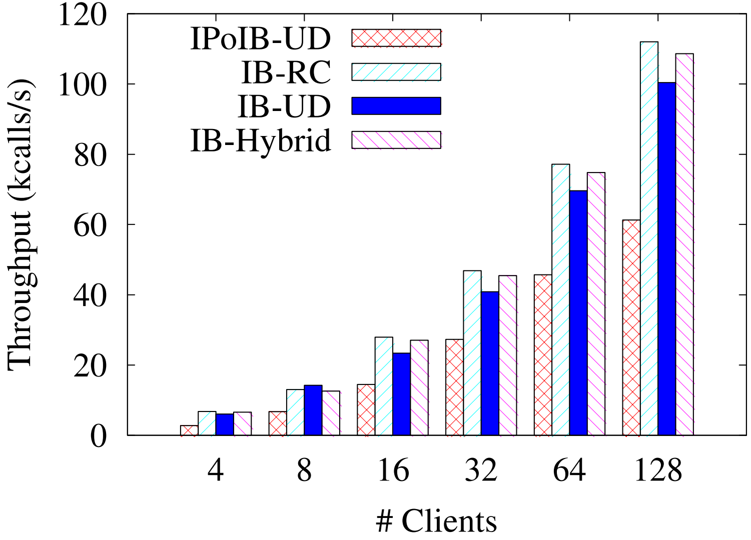
Get Throughput

- HBase Get average latency (FDR)
  - 4 client threads: 38 us
  - 59% improvement over IPoIB for 4 client threads
- HBase Get total throughput
  - 4 client threads: 102 Kops/sec
  - 2.4x improvement over IPoIB for 4 client threads

Performance Benefits for Hadoop RPC and HBase



Hadoop RPC Throughput on Chameleon-Cloud



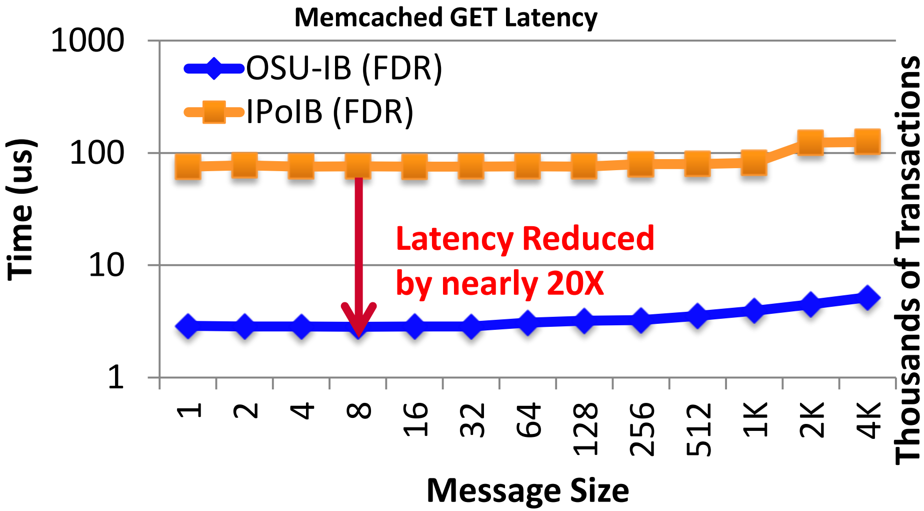
HBase YCSB Workload A on SDSC-Comet

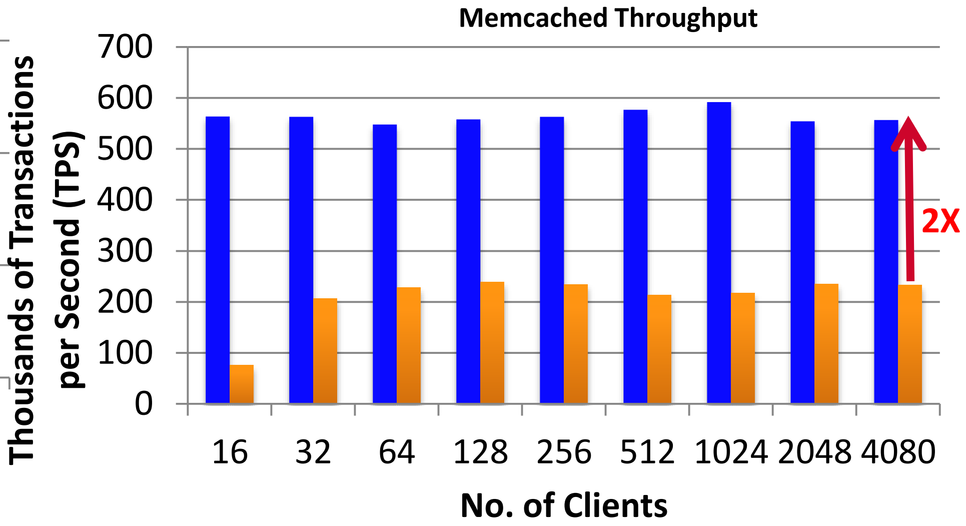
- Hadoop RPC Throughput on Chameleon-Cloud-FDR
  - up to 2.6x performance speedup over IPoIB for throughput
- HBase YCSB Workload A (read: write=50:50) on SDSC-Comet-FDR
  - Native designs (RC/UD/Hybrid) always perform better than the IPoIB-UD transport
  - up to 2.4x performance speedup over IPoIB for throughput

Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

Memcached Performance (FDR Interconnect)

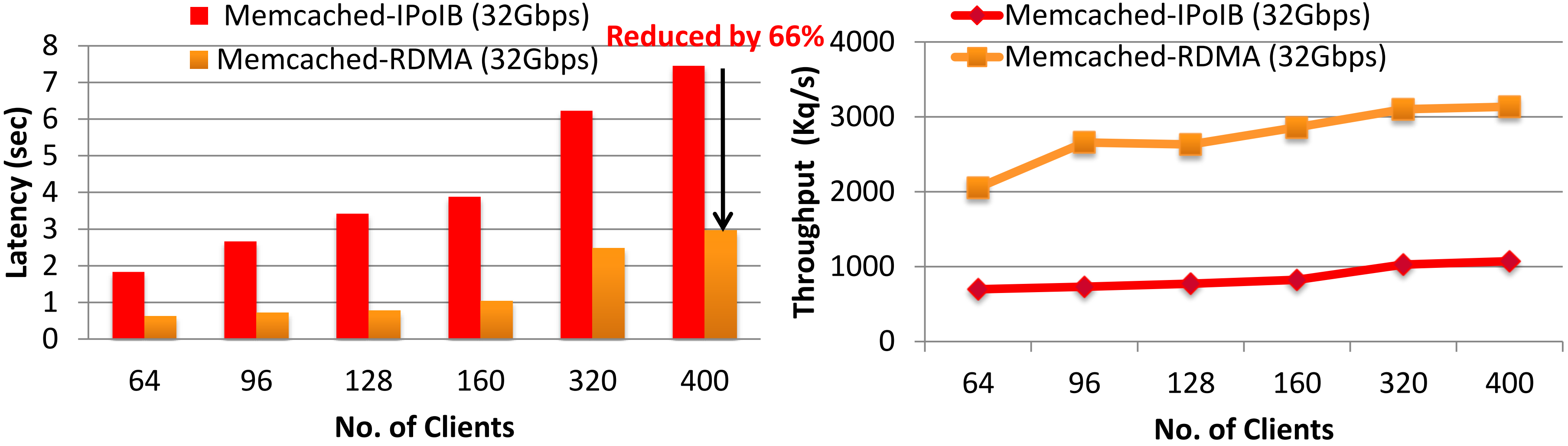




Experiments on TACC Stampede (Intel SandyBridge Cluster, IB: FDR)

- Memcached Get latency
  - 4 bytes OSU-IB: 2.84 us; IPoIB: 75.53 us
  - 2K bytes OSU-IB: 4.49 us; IPoIB: 123.42 us
- Memcached Throughput (4bytes)
  - 4080 clients OSU-IB: 556 Kops/sec, IPoIB: 233 Kops/s
  - Nearly 2X improvement in throughput

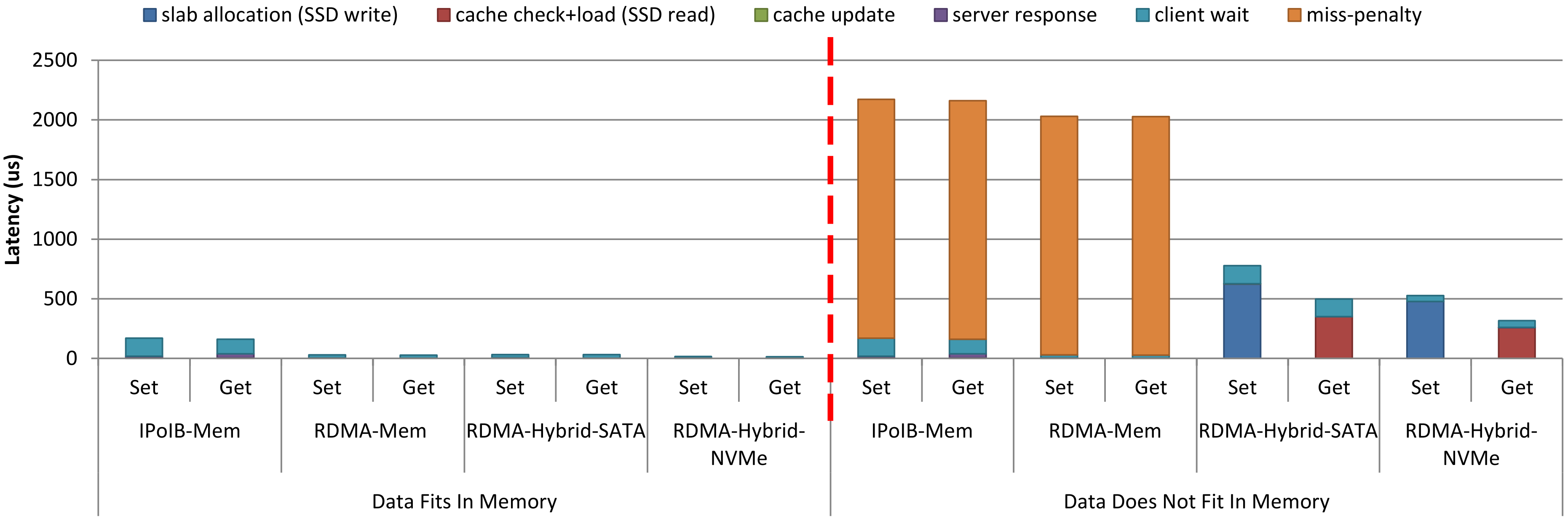
Micro-benchmark Evaluation for OLDP workloads



- Illustration with Read-Cache-Read access pattern using modified mysqlslap load testing tool
- Memcached-RDMA can
  - improve query latency by up to 66% over IPoIB (32Gbps)
  - throughput by up to 69% over IPoIB (32Gbps)

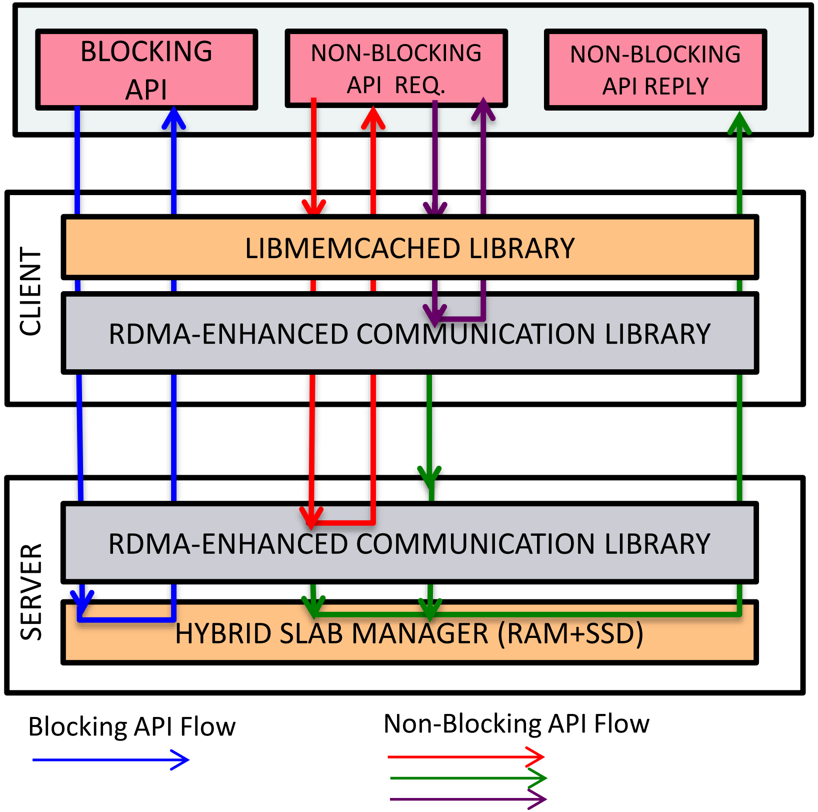
D. Shankar, X. Lu, J. Jose, M. W. Rahman, N. Islam, and D. K. Panda, Can RDMA Benefit On-Line Data Processing Workloads with Memcached and MySQL, ISPASS'15

Performance Evaluation on IB FDR + SATA/NVMe SSDs



- Memcached latency test with Zipf distribution, server with 1 GB memory, 32 KB key-value pair size, total size of data accessed is 1 GB (when data fits in memory) and 1.5 GB (when data does not fit in memory)
- When data fits in memory: RDMA-Mem/Hybrid gives 5x improvement over IPoIB-Mem
- When data does not fit in memory: RDMA-Hybrid gives 2x-2.5x over IPoIB/RDMA-Mem

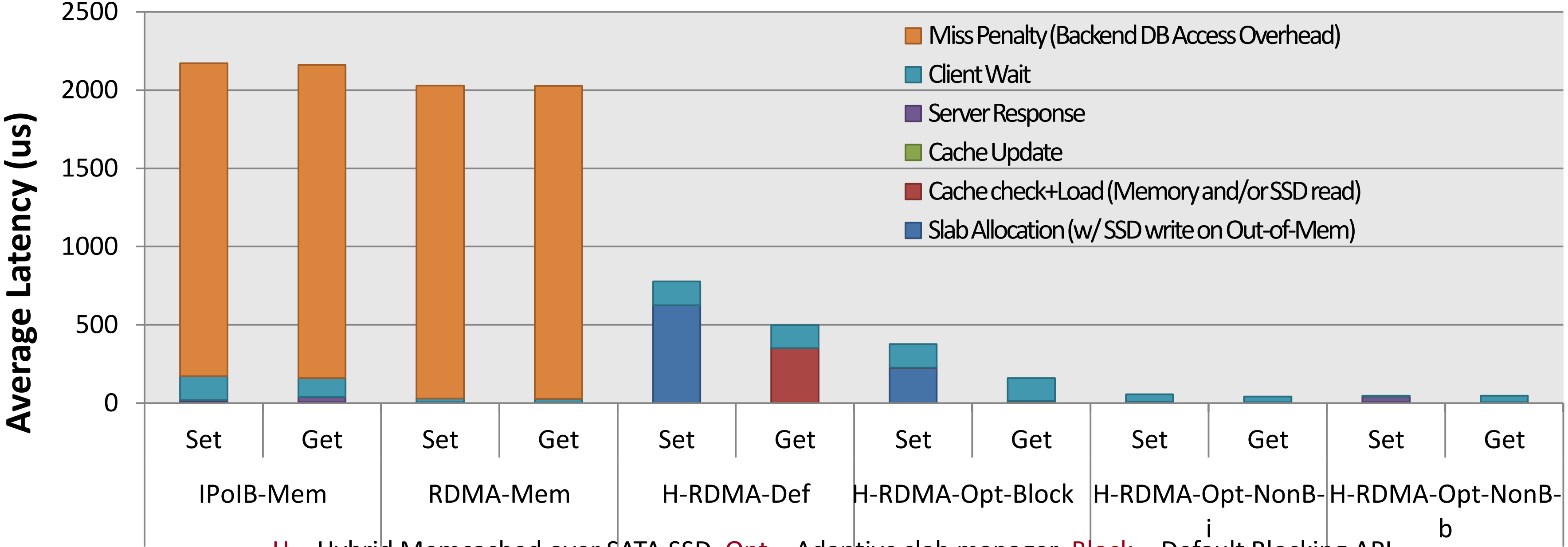
Accelerating Hybrid Memcached with RDMA, Non-blocking Extensions and SSDs



- RDMA-Accelerated Communication for Memcached Get/Set
- Hybrid 'RAM+SSD' slab management for higher data retention
- **Non-blocking API extensions**
  - **memcached_(iset/iget/bset/bget/test/wait)**
  - Achieve near in-memory speeds while hiding bottlenecks of network and SSD I/O
  - Ability to exploit communication/computation overlap
  - Optional buffer re-use guarantees
- Adaptive slab manager with different I/O schemes for higher throughput.

D. Shankar, X. Lu, N. S. 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, May 2016

Performance Evaluation with Non-Blocking Memcached API



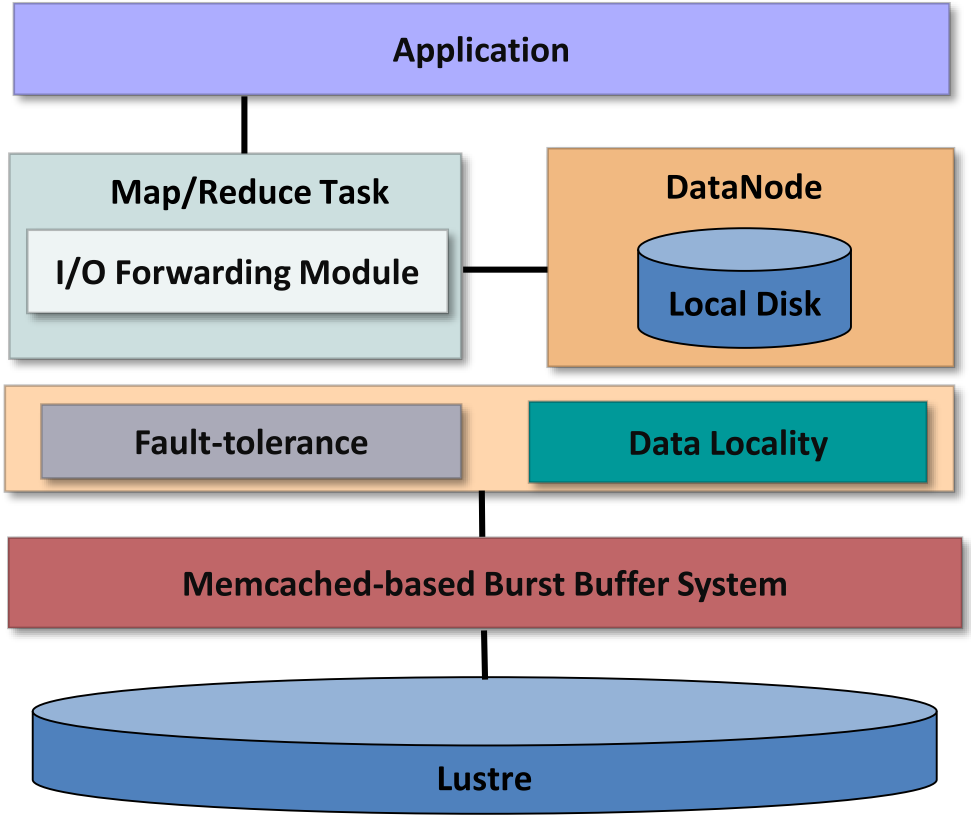
H = Hybrid Memcached over SATA SSD Opt = Adaptive slab manager Block = Default Blocking API
NonB-i = Non-blocking iset/iget API NonB-b = Non-blocking bset/bget API w/ buffer re-use guarantee

- Data does not fit in memory: Non-blocking Memcached Set/Get API Extensions can achieve
  - >16x latency improvement vs. blocking API over RDMA-Hybrid/RDMA-Mem w/ penalty
  - >2.5x throughput improvement vs. blocking API over default/optimized RDMA-Hybrid
- Data fits in memory: Non-blocking Extensions perform similar to RDMA-Mem/RDMA-Hybrid and >3.6x improvement over IPoIB-Mem

Acceleration Case Studies and Performance Evaluation

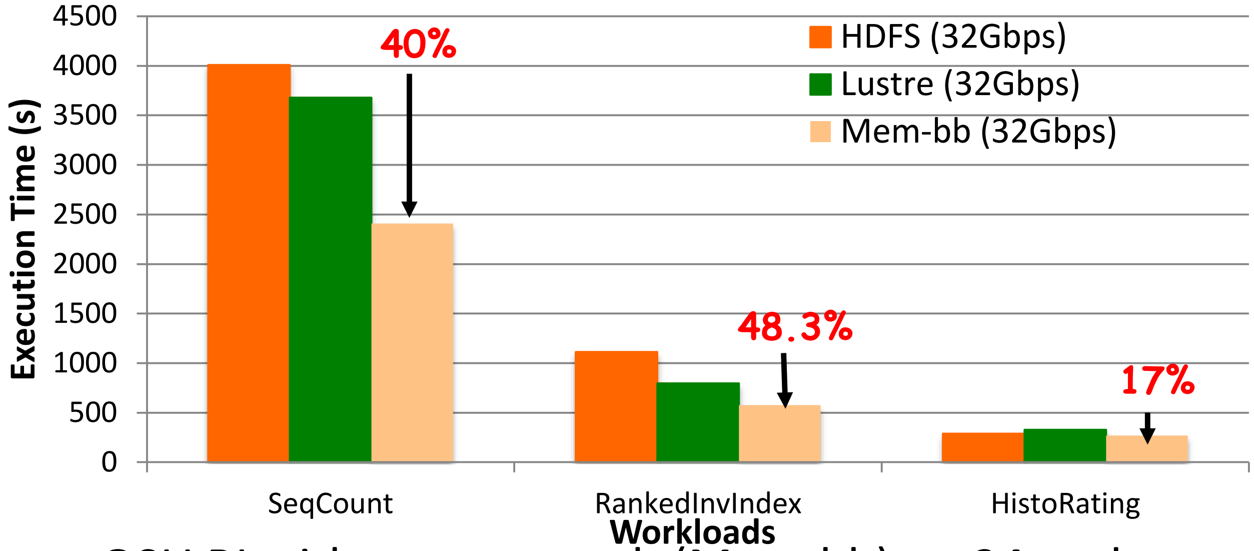
- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

Accelerating I/O Performance of Big Data Analytics through HDFS with RDMA-Memcached based Burst Buffer



- Design Features
  - Memcached-based burst-buffer system
    - Hides latency of parallel file system access
    - Read from local storage and Memcached
  - Data locality achieved by writing data to local storage
  - Different approaches of integration with parallel file system to guarantee fault-tolerance

Evaluation with PUMA Workloads



Gains on OSU RI with our approach (Mem-bb) on 24 nodes

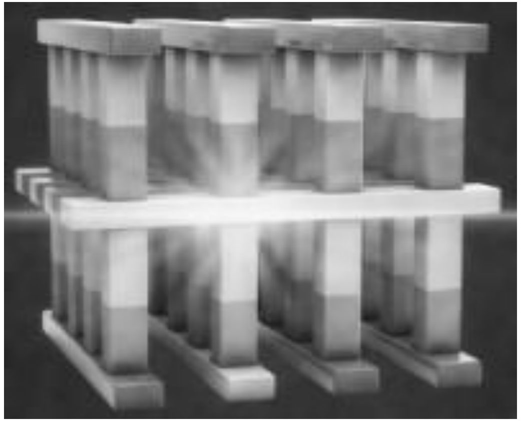
- SequenceCount: 34.5% over Lustre, 40% over HDFS
- RankedInvertedIndex: 27.3% over Lustre, 48.3% over HDFS
- HistogramRating: 17% over Lustre, 7% over HDFS

N. S. Islam, D. Shankar, X. Lu, M. W. Rahman, and D. K. Panda, Accelerating I/O Performance of Big Data Analytics with RDMA-based Key-Value Store, ICPP '15, September 2015

Acceleration Case Studies and Performance Evaluation

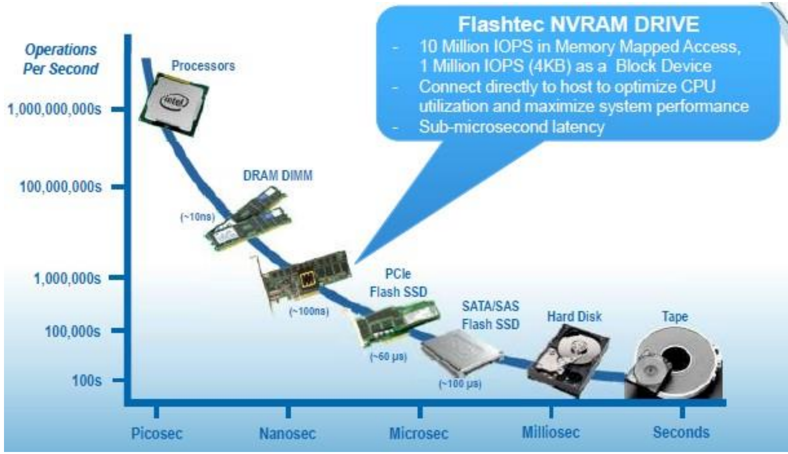
- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

Non-Volatile Memory (NVM) and NVMe-SSD



3D XPoint from Intel & Micron

Samsung NVMe SSD

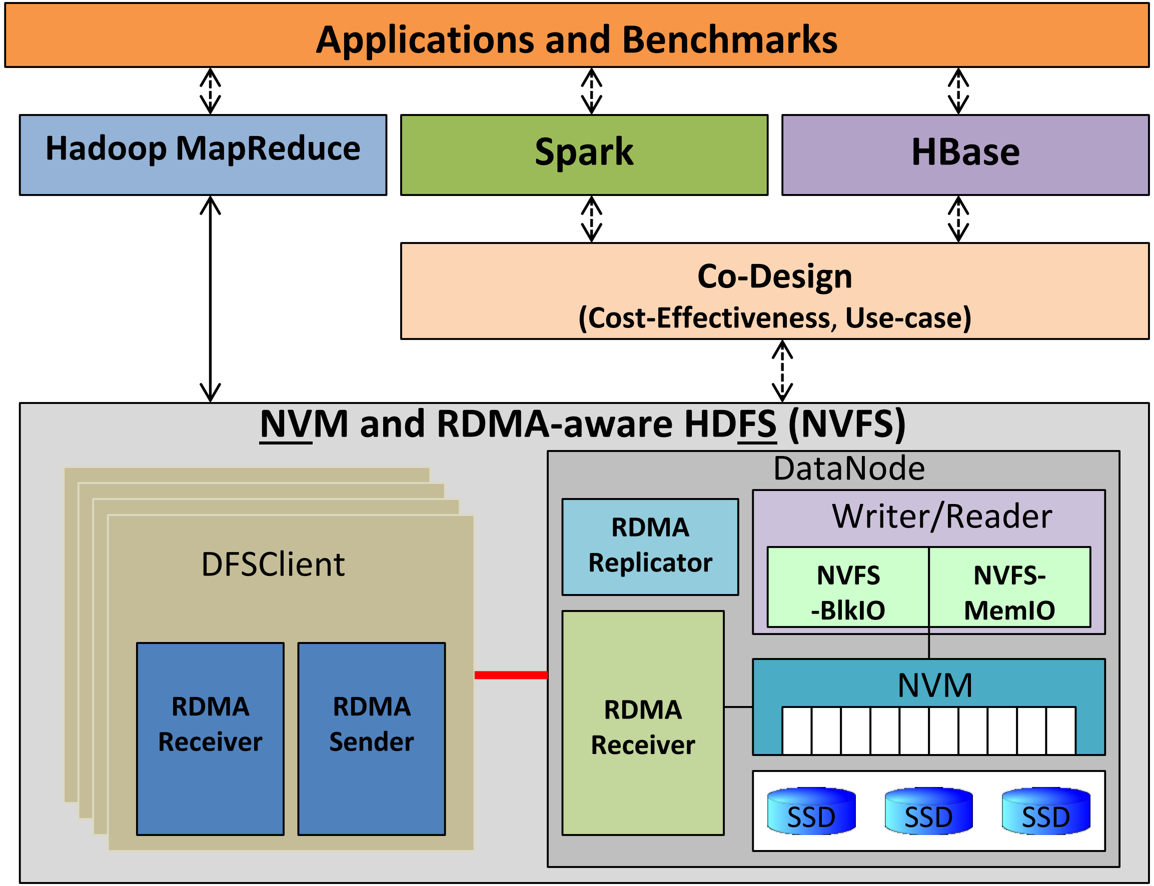


Performance of PMC Flashtec NVRAM [*]

- Non-Volatile Memory (NVM) provides byte-addressability with persistence
- The huge explosion of data in diverse fields require fast analysis and storage
- NVMs provide the opportunity to build high-throughput storage systems for data-intensive applications
- Storage technology is moving rapidly towards NVM

[*] http://www.enterprisetech.com/2014/08/06/ flashtec-nvram-15-million-iops-sub-microsecond- latency/

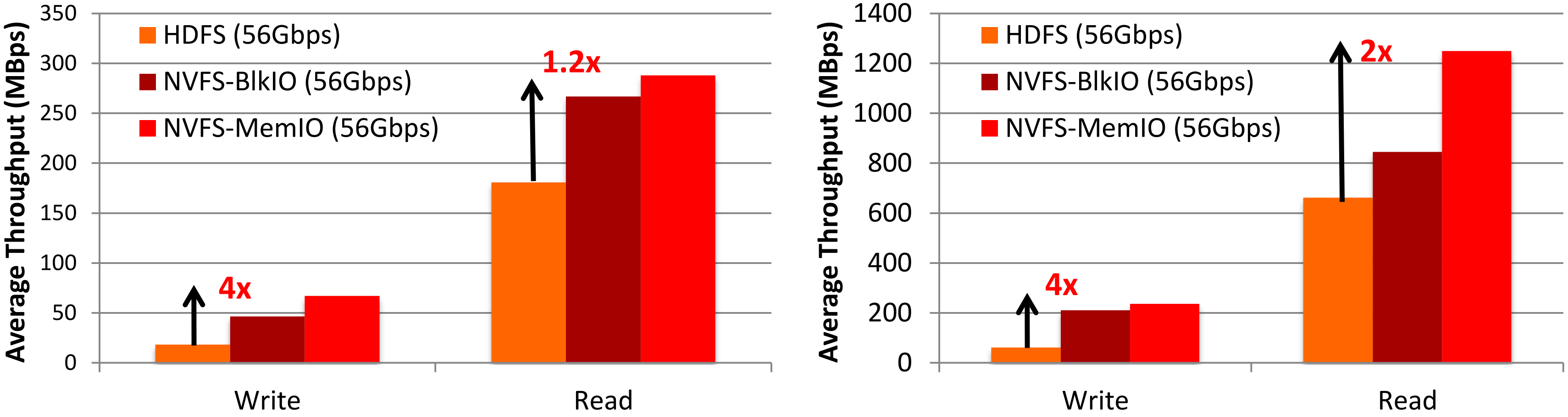
Design Overview of NVM and RDMA-aware HDFS (NVFS)



- **Design Features**
  - RDMA over NVM
  - HDFS I/O with NVM
    - Block Access
    - Memory Access
  - Hybrid design
    - NVM with SSD as a hybrid storage for HDFS I/O
  - Co-Design with Spark and HBase
    - Cost-effectiveness
    - Use-case

N. S. Islam, M. W. Rahman , X. Lu, and D. K. Panda, High Performance Design for HDFS with Byte-Addressability of NVM and RDMA, 24th International Conference on Supercomputing (ICS), June 2016

Evaluation with Hadoop MapReduce



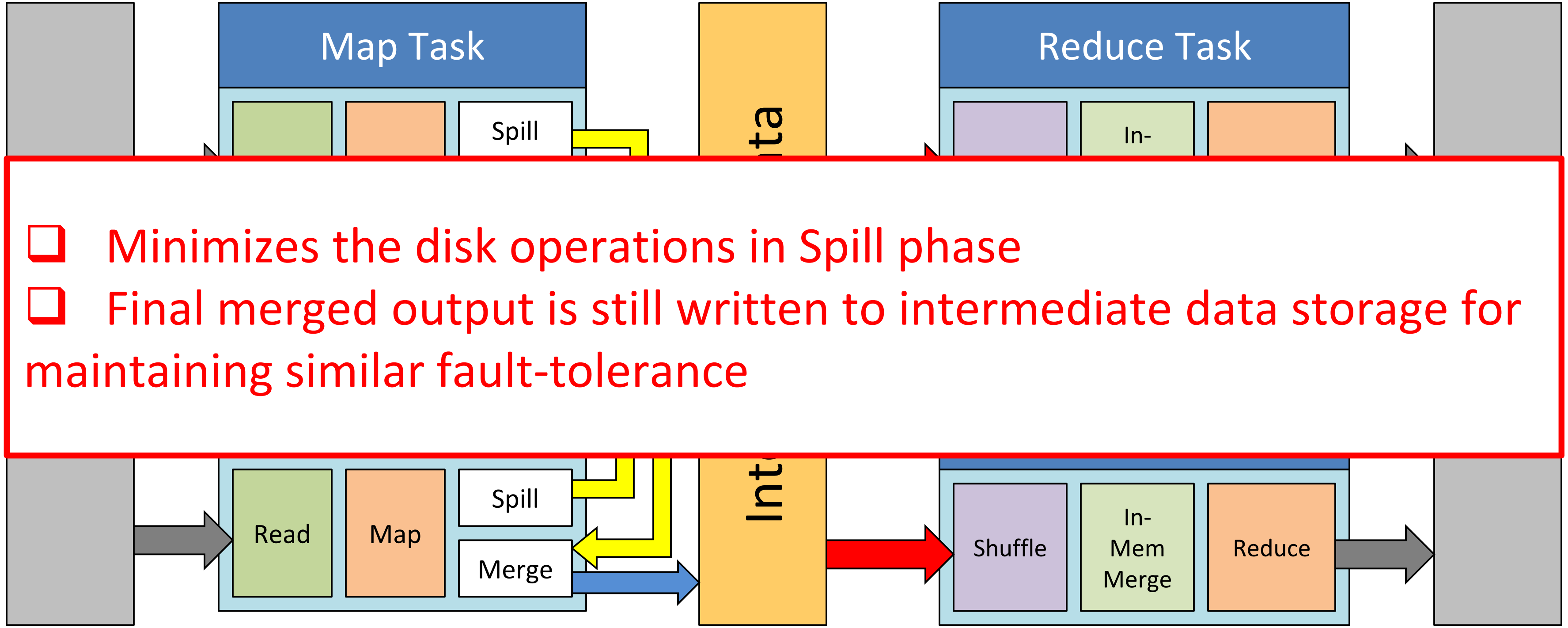
SDSC Comet (32 nodes)

OSU Nowlab (4 nodes)

TestDFSIO

- TestDFSIO on SDSC Comet (32 nodes)
  – Write: NVFS-MemIO gains by **4x** over HDFS
  – Read: NVFS-MemIO gains by **1.2x** over HDFS
- TestDFSIO on OSU Nowlab (4 nodes)
  – Write: NVFS-MemIO gains by **4x** over HDFS
  – Read: NVFS-MemIO gains by **2x** over HDFS

NVRAM-Assisted Map Spilling in HOMR

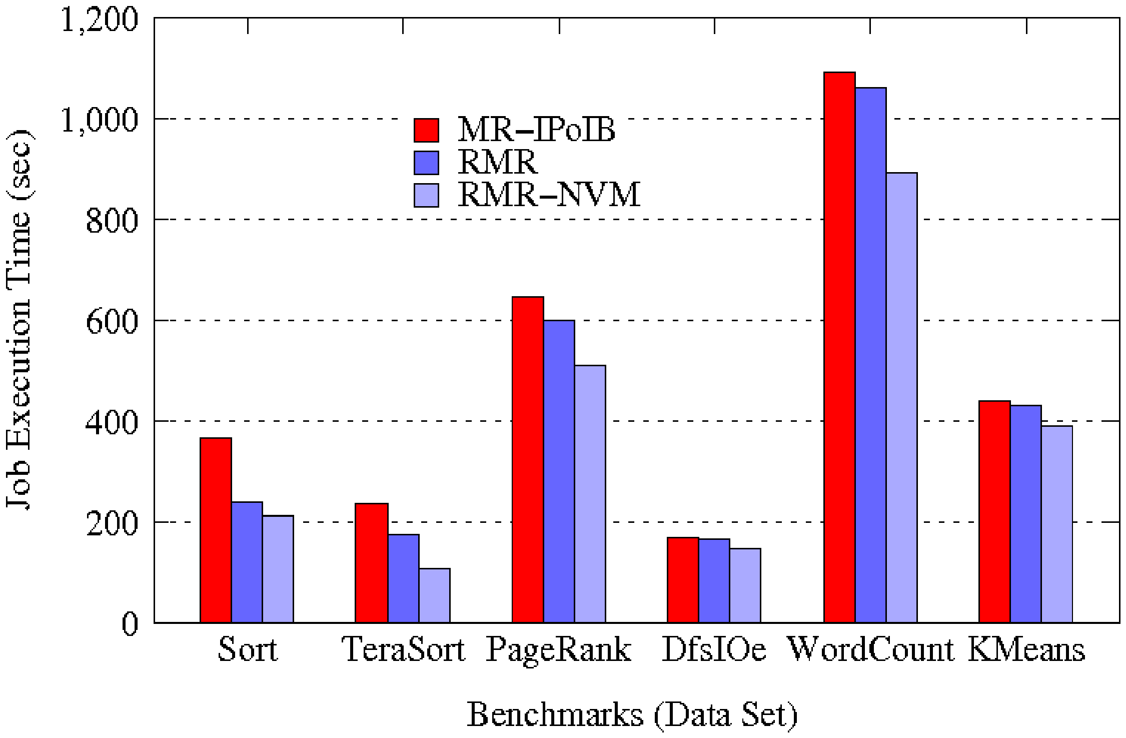


- ❑ Minimizes the disk operations in Spill phase
- ❑ Final merged output is still written to intermediate data storage for maintaining similar fault-tolerance

M. W. Rahman, N. S. Islam, X. Lu, and D. K. Panda, Can Non-Volatile Memory Benefit MapReduce Applications on HPC Clusters? PDSW-DISCS, with SC 2016.

Evaluation of Intel HiBench Workloads

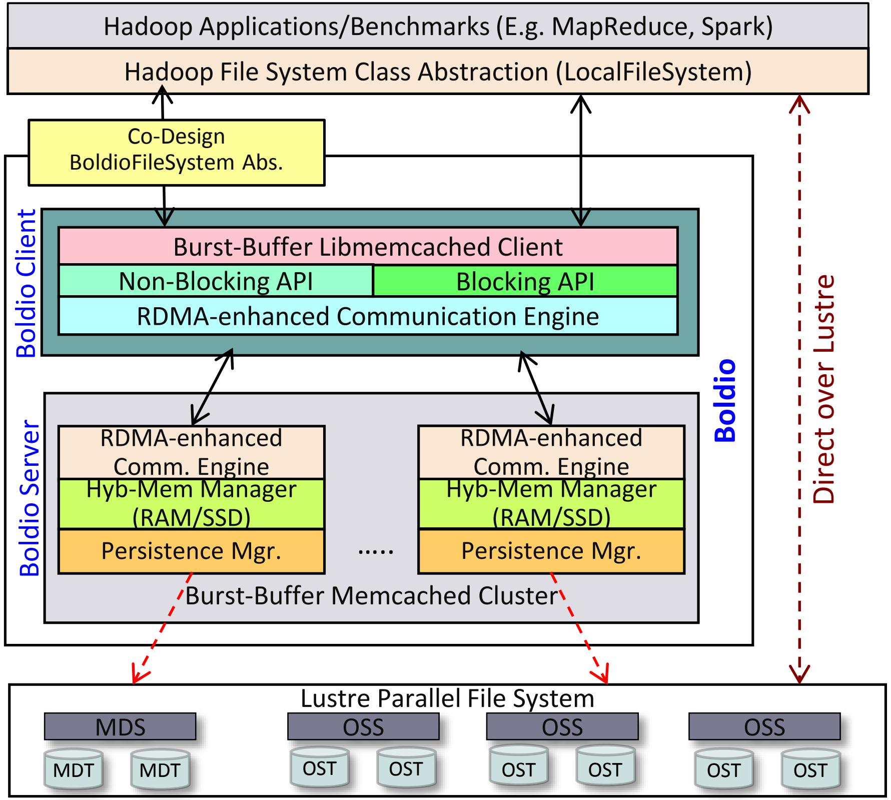
- We evaluate different HiBench workloads with Huge data sets on 8 nodes
- Performance benefits for Shuffle-intensive workloads compared to MR-IPoIB:
  - Sort: **42%** (25 GB)
  - TeraSort: **39%** (32 GB)
  - PageRank: **21%** (5 million pages)
- Other workloads:
  - WordCount: **18%** (25 GB)
  - KMeans: **11%** (100 million samples)



Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

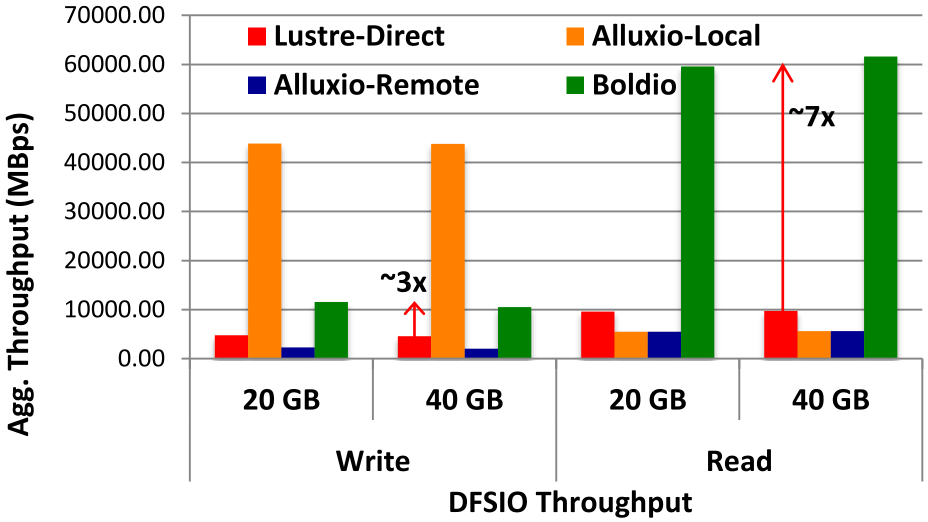
Burst-Buffer Over Lustre for Accelerating Big Data I/O (Boldio)

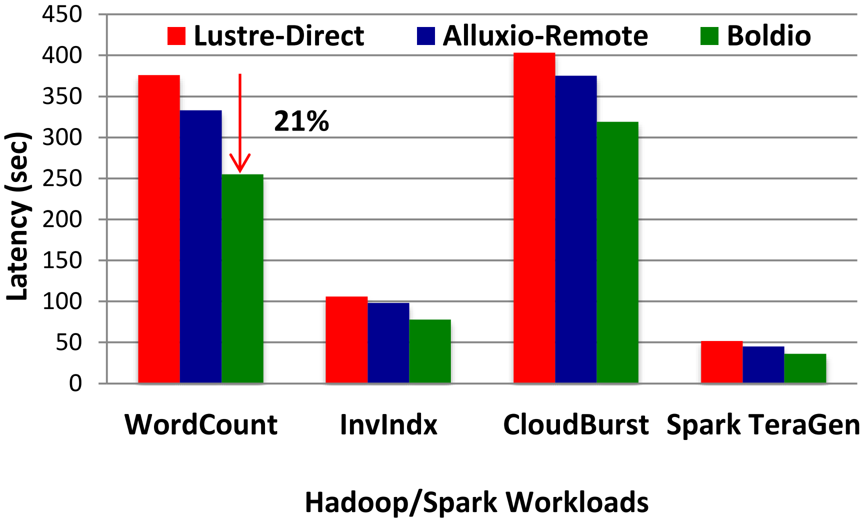

Hadoop Applications/Benchmarks (E.g. MapReduce, Spark)
Hadoop File System Class Abstraction (LocalFileSystem)
Co-Design
BoldioFileSystem Abs.
Boldio Client
Burst-Buffer Libmemcached Client
Non-Blocking API
Blocking API
RDMA-enhanced Communication Engine
Boldio
Direct over Lustre
Boldio Server
RDMA-enhanced Comm. Engine
Hyb-Mem Manager (RAM/SSD)
Persistence Mgr.
.....
RDMA-enhanced Comm. Engine
Hyb-Mem Manager (RAM/SSD)
Persistence Mgr.
Burst-Buffer Memcached Cluster
Lustre Parallel File System
MDS
MDT
MDT
OSS
OST
OST
OSS
OST
OST
OSS
OST
OST

- Hybrid and resilient key-value store-based Burst-Buffer system Over Lustre
- Overcome limitations of local storage on HPC cluster nodes
- Light-weight transparent interface to Hadoop/Spark applications
- Accelerating I/O-intensive Big Data workloads
  - Non-blocking Memcached APIs to maximize overlap
  - Client-based replication for resilience
  - Asynchronous persistence to Lustre parallel file system

D. Shankar, X. Lu, D. K. Panda, Boldio: A Hybrid and Resilient Burst-Buffer over Lustre for Accelerating Big Data I/O, IEEE Big Data 2016.

Performance Evaluation with Boldio





- Based on RDMA-based Libmemcached/Memcached 0.9.3, Hadoop-2.6.0
- InfiniBand QDR, 24GB RAM + PCIe-SSDs, 12 nodes, 32/48 Map/Reduce Tasks, 4-node Memcached cluster
- Boldio can improve
  - throughput over Lustre by about **3x** for write throughput and **7x** for read throughput
  - execution time of Hadoop benchmarks over Lustre, e.g. Wordcount, Cloudburst by **>21%**
- Contrasting with Alluxio (formerly Tachyon)
  - Performance degrades about 15x when Alluxio cannot leverage local storage (Alluxio-Local vs. Alluxio-Remote)
  - Boldio can improve throughput over Alluxio with all remote workers by about 3.5x - 8 .8x (Alluxio-Remote vs. Boldio)

Acceleration Case Studies and Performance Evaluation

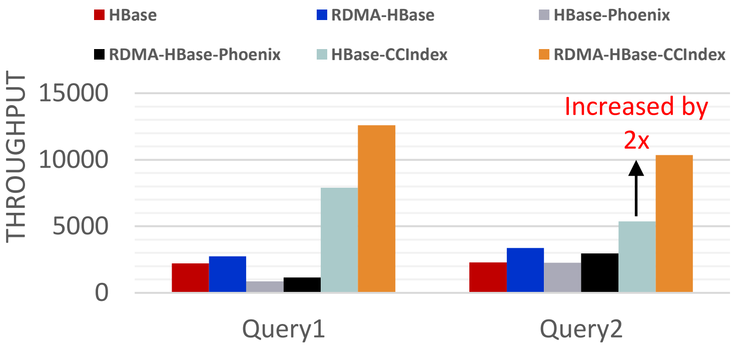
- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

Accelerating Indexing Techniques on HBase with RDMA

- Challenges
  - Operations on Distributed Ordered Table (DOT) with indexing techniques are network intensive
  - Additional overhead of creating and maintaining secondary indices
  - Can RDMA benefit indexing techniques (Apache Phoenix and CCIndex) on HBase?
- Results
  - Evaluation with Apache Phoenix and CCIndex
  - Up to 2x improvement in query throughput
  - Up to 35% reduction in application workload execution time

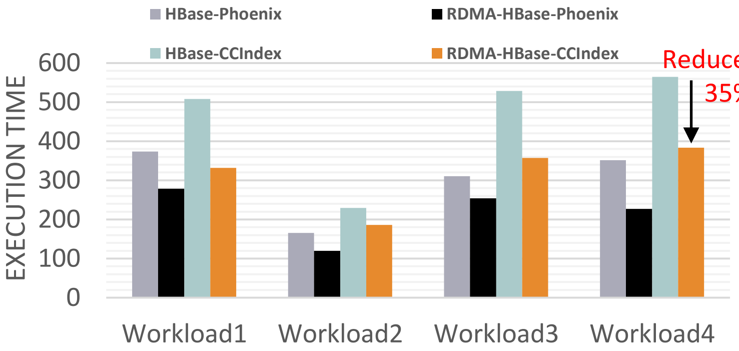
Collaboration with Institute of Computing Technology, Chinese Academy of Sciences

TPC-H Query Benchmarks



Increased by 2x

Ad Master Application Workloads



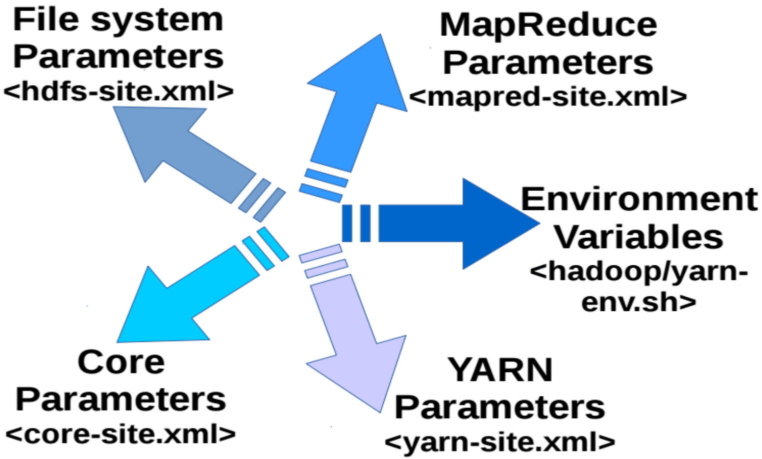
Reduced by 35%

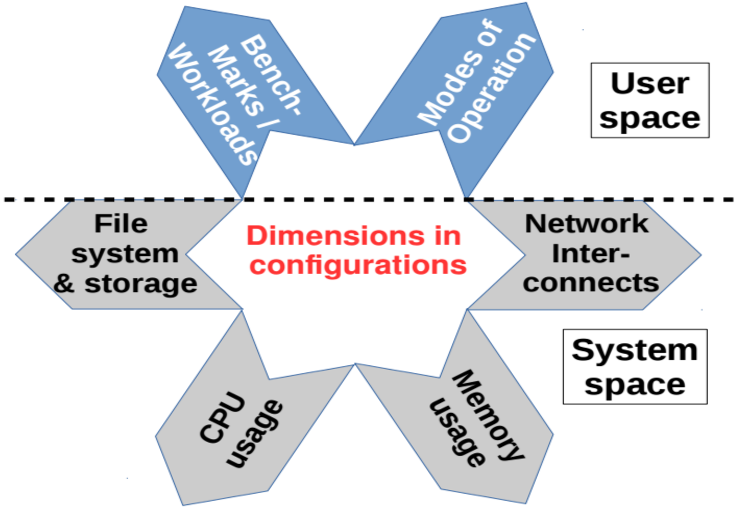
S. Gugnani, X. Lu, L. Zha, and D. K. Panda, Characterizing and Accelerating Indexing Techniques on HBase for Distributed Ordered Table-based Queries and Applications (Under review)

Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

Challenges of Tuning and Profiling

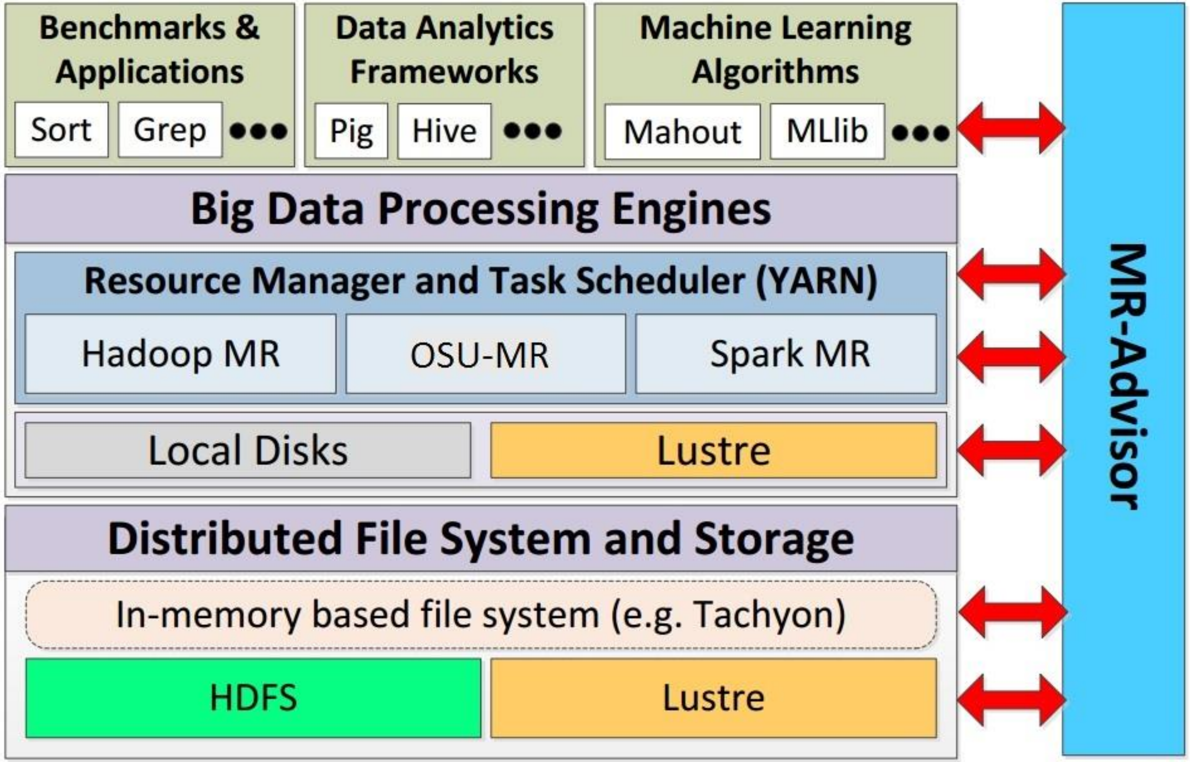




- MapReduce systems have different configuration parameters based on the underlying component that uses these
- The parameter files vary across different MapReduce stacks

- Proposed a generalized parameter space for HPC clusters
- Two broad dimensions: user space and system space; existing parameters can be categorized in the proposed spaces

MR-Advisor Overview

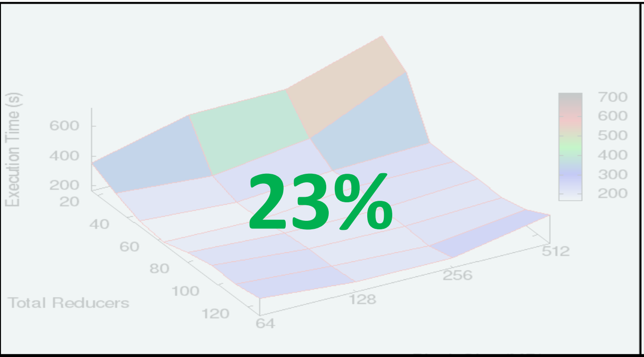


Benchmarks & Applications
Sort
Grep
●●●
Data Analytics Frameworks
Pig
Hive
●●●
Machine Learning Algorithms
Mahout
MLlib
●●●
Big Data Processing Engines
Resource Manager and Task Scheduler (YARN)
Hadoop MR
OSU-MR
Spark MR
Local Disks
Lustre
Distributed File System and Storage
In-memory based file system (e.g. Tachyon)
HDFS
Lustre
MR-Advisor

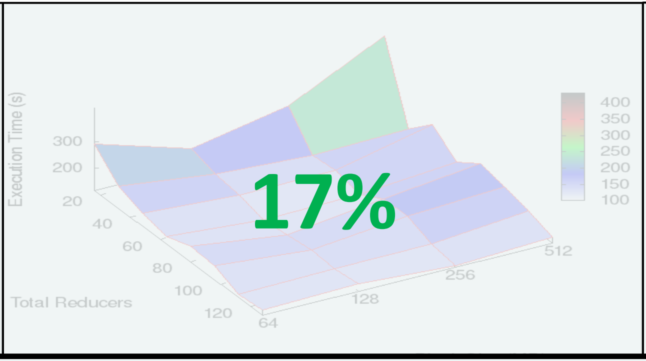
M. W. Rahman , N. S. Islam, X. Lu, D. Shankar, and D. K. Panda, *MR-Advisor: A Comprehensive Tuning Tool for Advising HPC Users to Accelerate MapReduce Applications on Supercomputers,* SBAC-PAD, 2016.

- A generalized framework for Big Data processing engines to perform tuning, profiling, and prediction
- Current framework can work with Hadoop, Spark, and RDMA MapReduce (OSU-MR)
- Can also provide tuning for different file systems (e.g. HDFS, Lustre, Tachyon), resource managers (e.g. YARN), and applications

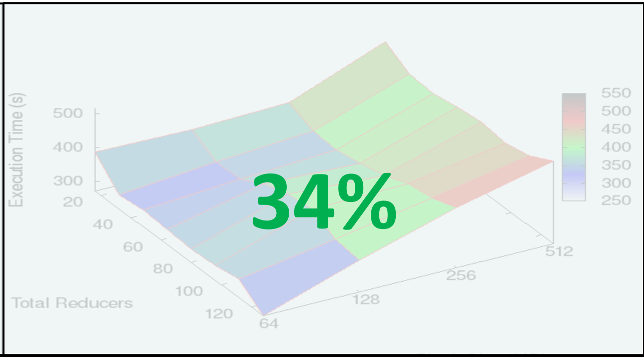
Tuning Experiments with MR-Advisor (TACC Stampede)



23%

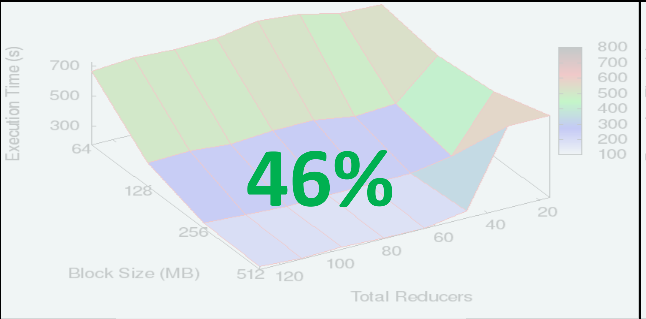


17%

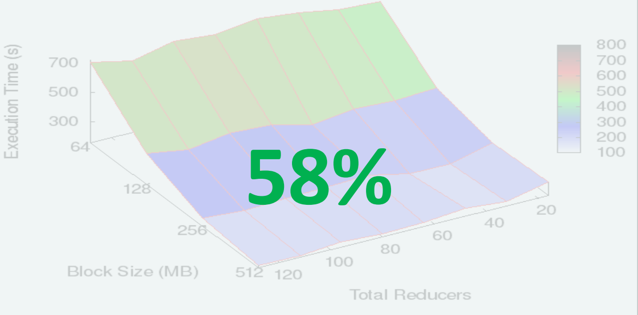


34%

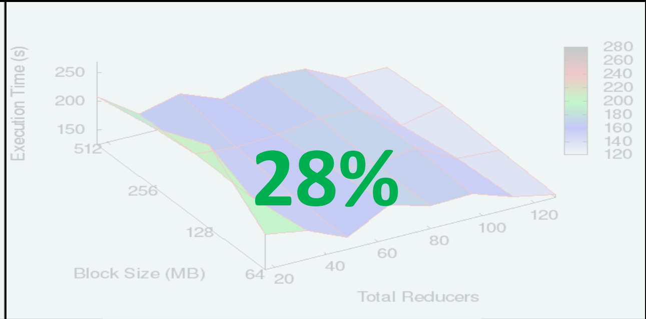
Performance improvements compared to current best practice values



46%



58%



28%

Apache MR over Lustre

OSU-MR over Lustre

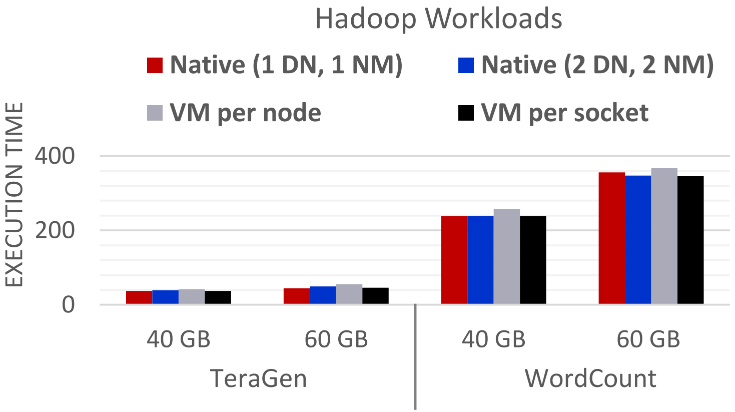
Apache Spark over Lustre

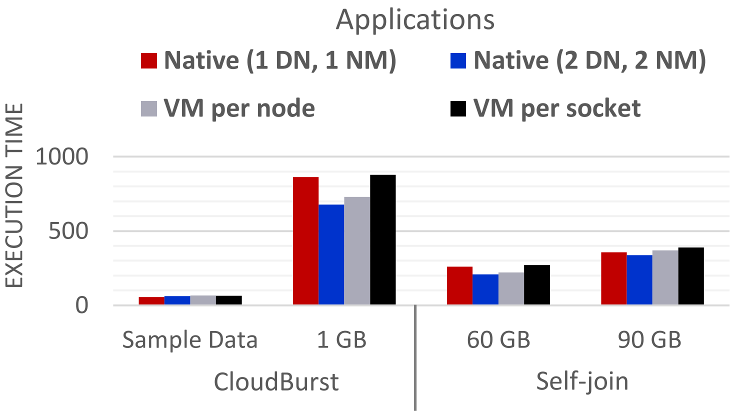
Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS and MapReduce
  - Spark
  - Hadoop RPC and HBase
  - Memcached
  - HDFS with Memcached-based Burst Buffer
- Advanced Designs
  - HDFS and MapReduce with NVRAM
  - Accelerating Big Data I/O (Lustre + Burst-Buffer)
  - Efficient Indexing with RDMA-HBase
  - MR-Advisor
- BigData + HPC Cloud

Performance Characterization of Hadoop Workloads on SR-IOV-enabled Clouds

- Motivation
  - Performance attributes of Big Data workloads when using SR-IOV are not known
  - Impact of VM subscription policies, data size, and type of workload on performance of workloads with SR-IOV not evaluated in systematic manner
- Results
  - Evaluation on Chameleon Cloud with RDMA-Hadoop
  - Only 0.3 – 13% overhead with SR-IOV compared to native execution
  - Best VM subscription policy depends on type of workload

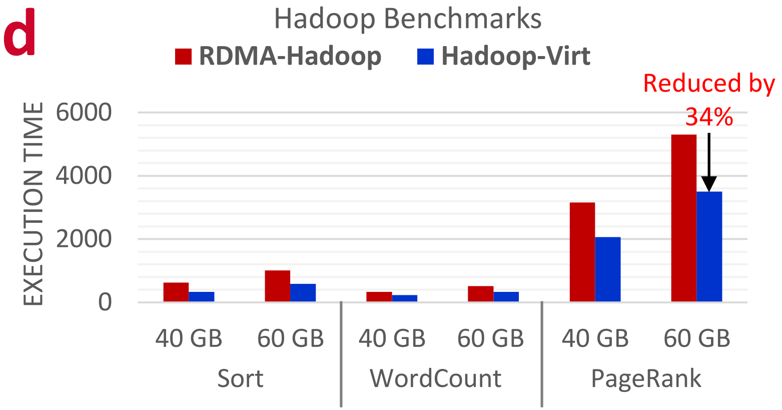


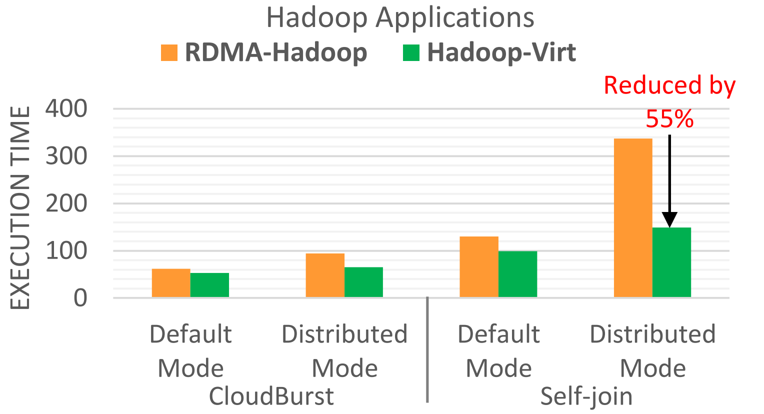


S. Gugnani, X. Lu, and D. K. Panda, Performance Characterization of Hadoop Workloads on SR-IOV-enabled Virtualized InfiniBand Clusters, accepted at BDCAT'16, December 2016

Virtualization-aware and Automatic Topology Detection Schemes in Hadoop on InfiniBand

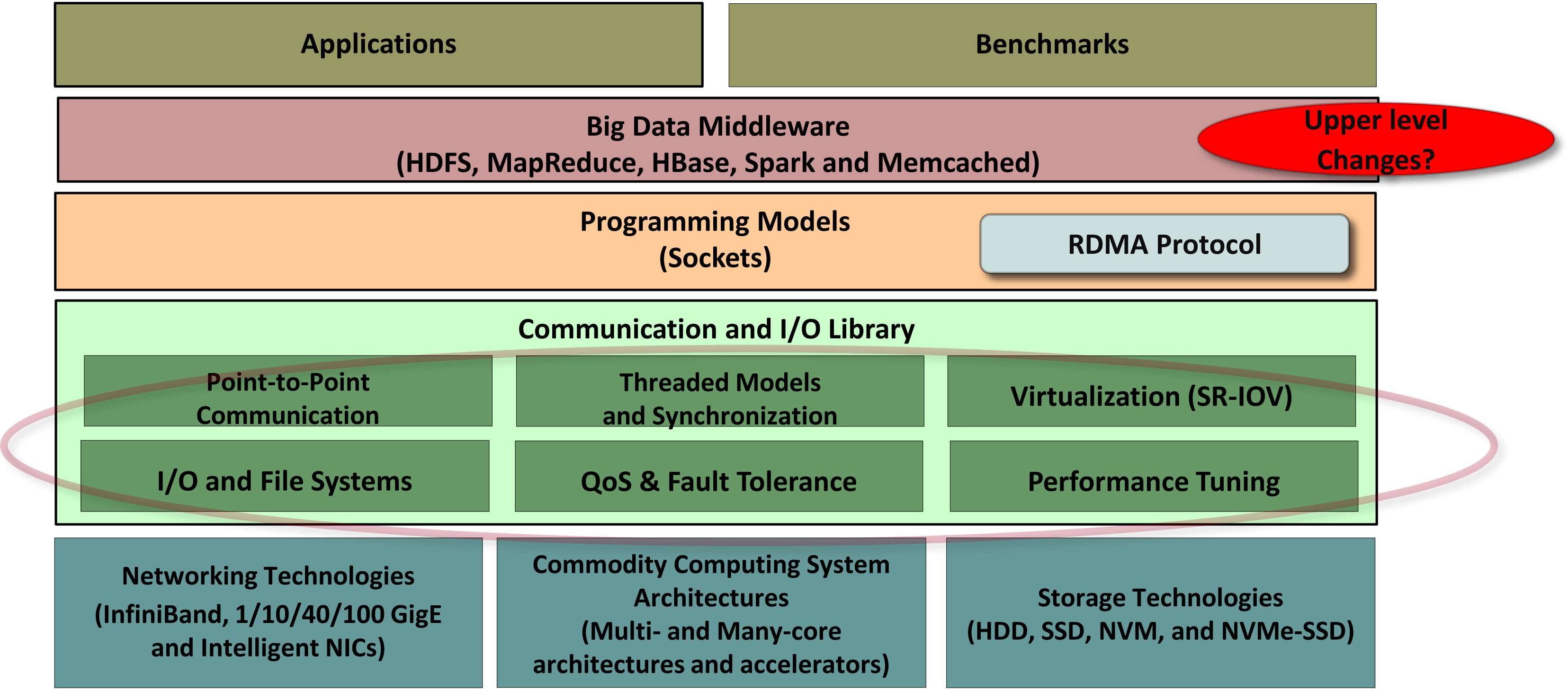
- Challenges
  - Existing designs in Hadoop not virtualization-aware
  - No support for automatic topology detection
- Design
  - Automatic Topology Detection using MapReduce-based utility
    - Requires no user input
    - Can detect topology changes during runtime without affecting running jobs
  - Virtualization and topology-aware communication through map task scheduling and YARN container allocation policy extensions





S. Gugnani, X. Lu, and D. K. Panda, Designing Virtualization-aware and Automatic Topology Detection Schemes for Accelerating Hadoop on SR-IOV-enabled Clouds, CloudCom'16, December 2016

Designing Communication and I/O Libraries for Big Data Systems: Solved a Few Initial Challenges


Applications
Benchmarks
Big Data Middleware
(HDFS, MapReduce, HBase, Spark and Memcached)
Upper level Changes?
Programming Models
(Sockets)
RDMA Protocol
Communication and I/O Library
Point-to-Point Communication
Threaded Models and Synchronization
Virtualization (SR-IOV)
I/O and File Systems
QoS & Fault Tolerance
Performance Tuning
Networking Technologies
(InfiniBand, 1/10/40/100 GigE and Intelligent NICs)
Commodity Computing System Architectures
(Multi- and Many-core architectures and accelerators)
Storage Technologies
(HDD, SSD, NVM, and NVMe-SSD)

On-going and Future Plans of OSU High Performance Big Data (HiBD) Project

- Upcoming Releases of RDMA-enhanced Packages will support
  - Upgrades to the latest versions of Hadoop and Spark
  - Streaming
  - MR-Advisor
  - Impala
- Upcoming Releases of OSU HiBD Micro-Benchmarks (OHB) will support
  - MapReduce, RPC
- Advanced designs with upper-level changes and optimizations
  - Boldio (Burst Buffer over Lustre for Big Data I/O Acceleration)
  - Efficient Indexing

Concluding Remarks

- Presented an overview of Big Data, Hadoop (MapReduce, HDFS, HBase, Spark, RPC) and Memcached
- Provided an overview of Networking Technologies
- Discussed challenges in accelerating Hadoop and Memcached
- Presented basic and advanced designs to take advantage of InfiniBand/RDMA for HDFS, MapReduce, HBase, Spark, RPC and Memcached on HPC clusters and clouds
- Results are promising
- Many other open issues need to be solved
- Will enable Big Data community to take advantage of modern HPC technologies to carry out their analytics in a fast and scalable manner

Funding Acknowledgments

Funding Support by

Office of Science U.S. DEPARTMENT OF ENERGY

NSF

Mellanox TECHNOLOGIES

CISCO SYSTEMS

QLOGIC

CRAY THE SUPERCOMPUTER COMPANY

intel

LINUX NETWORX

NVIDIA

NetApp

Sun microsystems

Equipment Support by

Mellanox TECHNOLOGIES

intel

AMD

Sun microsystems

APPRO

advanced clustering technologies, inc.

Microway Technology you can count on

NVIDIA

QLOGIC

Personnel Acknowledgments

Current Students

- A. Awan (Ph.D.)
- M. Bayatpour (Ph.D.)
- S. Chakraborthy (Ph.D.)
- C.-H. Chu (Ph.D.)
- S. Guganani (Ph.D.)
- J. Hashmi (Ph.D.)
- N. Islam (Ph.D.)
- M. Li (Ph.D.)
- M. Rahman (Ph.D.)
- D. Shankar (Ph.D.)
- A. Venkatesh (Ph.D.)
- J. Zhang (Ph.D.)

Current Research Scientists

- X. Lu
- H. Subramoni

Current Research Specialist

- J. Smith

Past Students

- A. Augustine (M.S.)
- P. Balaji (Ph.D.)
- S. Bhagvat (M.S.)
- A. Bhat (M.S.)
- D. Buntinas (Ph.D.)
- L. Chai (Ph.D.)
- B. Chandrasekharan (M.S.)
- N. Dandapanthula (M.S.)
- V. Dhanraj (M.S.)
- T. Gangadharappa (M.S.)
- K. Gopalakrishnan (M.S.)
- W. Huang (Ph.D.)
- W. Jiang (M.S.)
- J. Jose (Ph.D.)
- S. Kini (M.S.)
- M. Koop (Ph.D.)
- K. Kulkarni (M.S.)
- R. Kumar (M.S.)
- S. Krishnamoorthy (M.S.)
- K. Kandalla (Ph.D.)
- P. Lai (M.S.)
- J. Liu (Ph.D.)
- M. Luo (Ph.D.)
- A. Mamidala (Ph.D.)
- G. Marsh (M.S.)
- V. Meshram (M.S.)
- A. Moody (M.S.)
- S. Naravula (Ph.D.)
- R. Noronha (Ph.D.)
- X. Ouyang (Ph.D.)
- S. Pai (M.S.)
- S. Potluri (Ph.D.)
- R. Rajachandrasekar (Ph.D.)
- G. Santhanaraman (Ph.D.)
- A. Singh (Ph.D.)
- J. Sridhar (M.S.)
- S. Sur (Ph.D.)
- H. Subramoni (Ph.D.)
- K. Vaidyanathan (Ph.D.)
- A. Vishnu (Ph.D.)
- J. Wu (Ph.D.)
- W. Yu (Ph.D.)

Past Research Scientist

- K. Hamidouche
- S. Sur

Past Programmers

- D. Bureddy
- M. Arnold
- J. Perkins

Past Post-Docs

- D. Banerjee
- X. Besseron
- H.-W. Jin
- J. Lin
- M. Luo
- E. Mancini
- S. Marcarelli
- J. Vienne
- H. Wang

The 3rd International Workshop on High-Performance Big Data Computing (HPBDC)

HPBDC 2017 will be held with IEEE International Parallel and Distributed Processing Symposium (IPDPS 2017), Orlando, Florida USA, May, 2017

Keynote Speaker: Prof. Satoshi Matsuoka, Tokyo Institute of Technology, Japan
Panel Moderator: Jianfeng Zhan (ICT/CAS)
Panel Topic: Sunrise or Sunset: Exploring the Design Space of Big Data Software Stack

http://web.cse.ohio-state.edu/~luxi/hpbdc2017

HPBDC 2016 was held in conjunction with IPDPS'16
Keynote Talk: Dr. Chaitanya Baru,
Senior Advisor for Data Science, National Science Foundation (NSF);
Distinguished Scientist, San Diego Supercomputer Center (SDSC)

Panel Moderator: Jianfeng Zhan (ICT/CAS)
Panel Topic: Merge or Split: Mutual Influence between Big Data and HPC Techniques
Six Regular Research Papers and Two Short Research Papers

http://web.cse.ohio-state.edu/~luxi/hpbdc2016

Thank You!

{panda}@cse.ohio-state.edu

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

Network-Based Computing Laboratory

http://nowlab.cse.ohio-state.edu/

The High-Performance Big Data Project

http://hibd.cse.ohio-state.edu/