HiBD
High-Performance Big Data

High Performance Big Data (HiBD): Accelerating Hadoop, Spark and Memcached on Modern Clusters

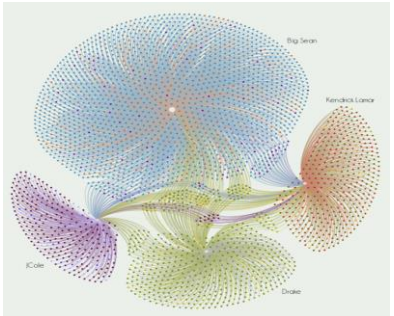
Presentation at Mellanox Theatre (SC '16)

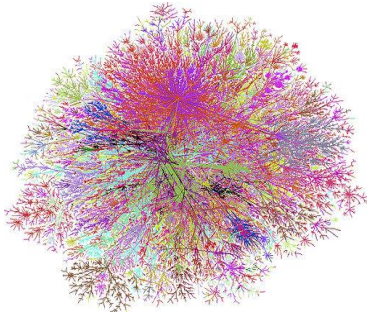
by

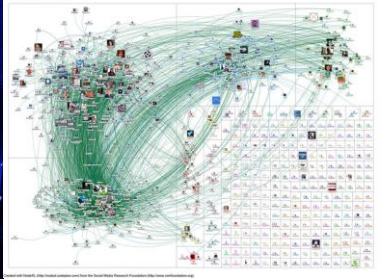
Dhabaleswar K. (DK) Panda
The Ohio State University
E-mail: panda@cse.ohio-state.edu
http://www.cse.ohio-state.edu/~panda

Introduction to Big Data Applications and Analytics

- Big Data has become the one of the most important elements of business analytics
- Provides groundbreaking opportunities for enterprise information management and decision making
- The amount of data is exploding; companies are capturing and digitizing more information than ever
- The rate of information growth appears to be exceeding Moore's Law

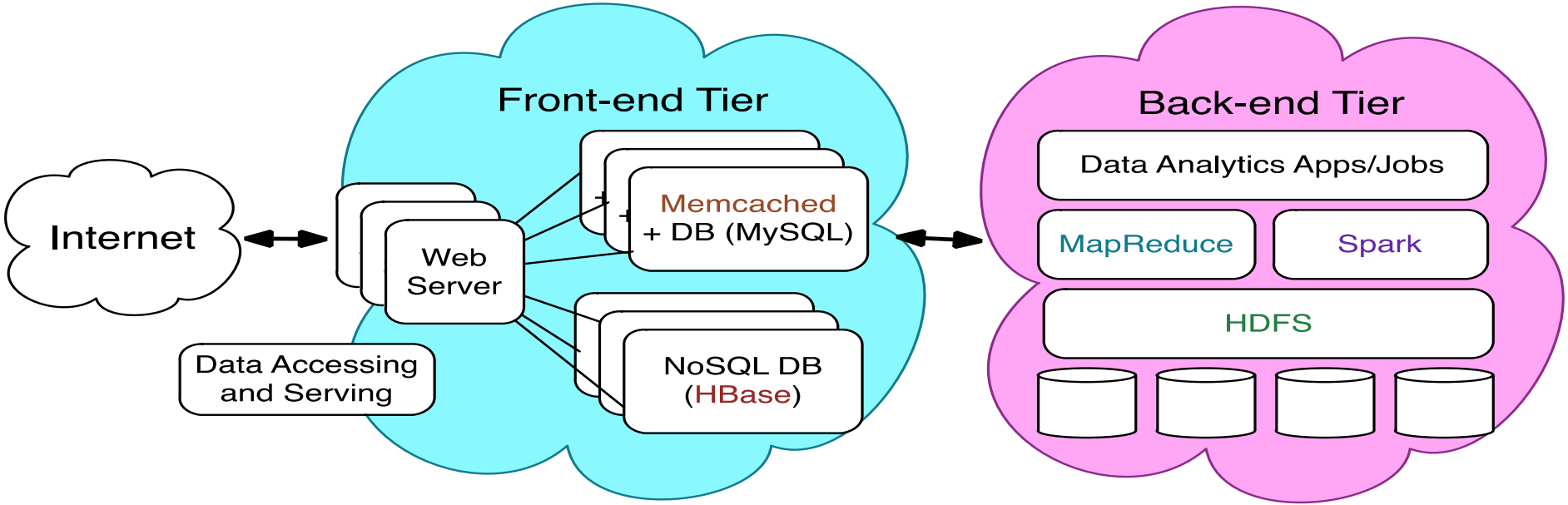






Data Management and Processing on Modern Clusters

- 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



Drivers of Modern HPC Cluster Architectures

Multi-core Processors

High Performance Interconnects - InfiniBand
<1usec latency, 100Gbps 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)
- Solid State Drives (SSDs), Non-Volatile Random-Access Memory (NVRAM), NVMe-SSD
- Accelerators (NVIDIA GPGPUs and Intel Xeon Phi)

Tianhe – 2

Titan

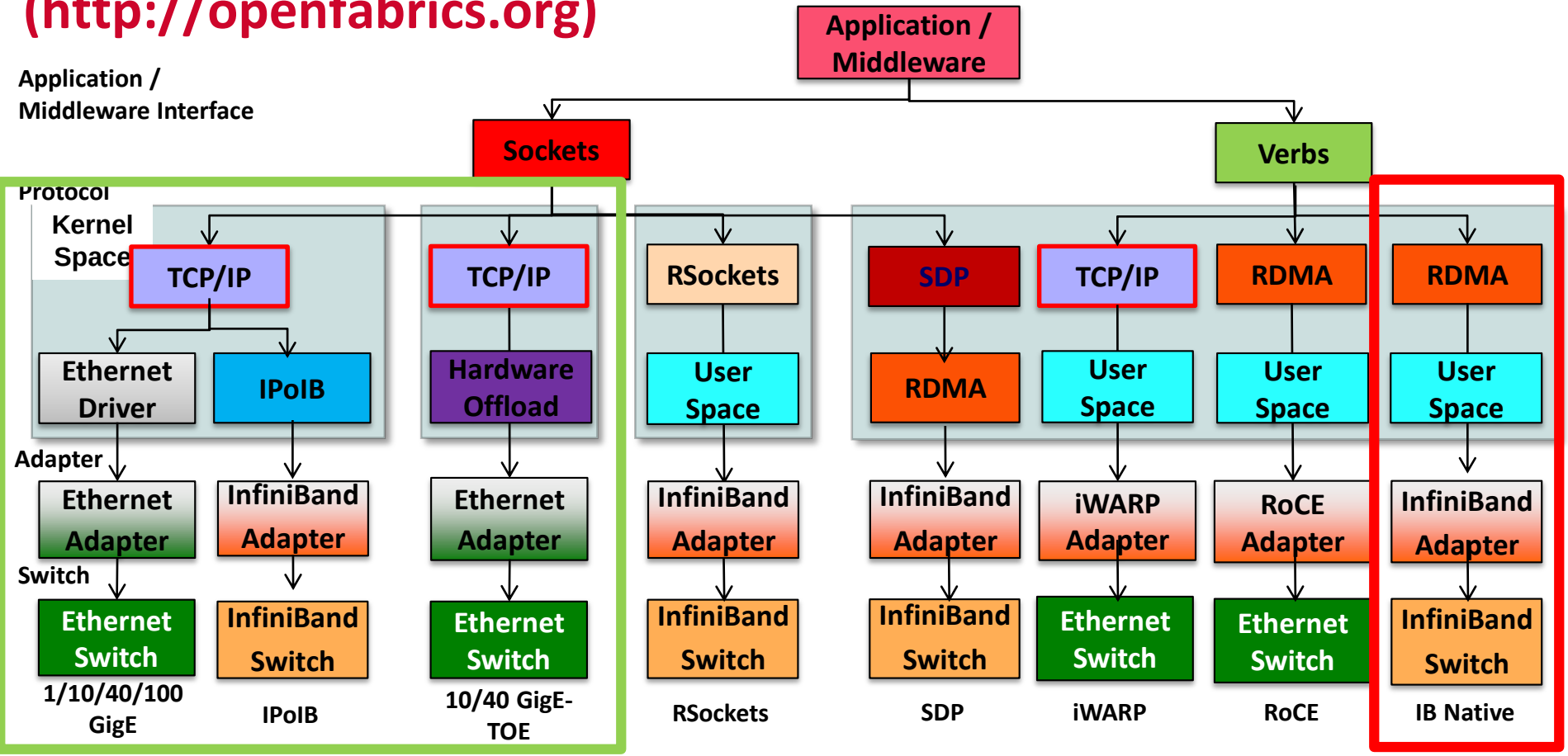
Stampede

Tianhe – 1A

Trends in HPC Technologies

- Advanced Interconnects and RDMA protocols
  - InfiniBand
  - 10-40 Gigabit 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)



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 2,690 organizations in 83 countries**
  - **More than 402,000 (> 0.4 million) downloads from the OSU site directly**
  - Empowering many TOP500 clusters (Nov '16 ranking)
    - **1st ranked 10,649,640-core cluster (Sunway TaihuLight) at NSC, Wuxi, China**
    - 13th ranked 241,108-core cluster (Pleiades) at NASA
    - 17th ranked 519,640-core cluster (Stampede) at TACC
    - 40th ranked 76,032-core cluster (Tsubame 2.5) at Tokyo Institute of Technology and many others
  - 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) ->
    Sunway TaihuLight at NSC, Wuxi, China (1st in Nov'16, 10,649,640 cores, 93 PFlops)

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

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?

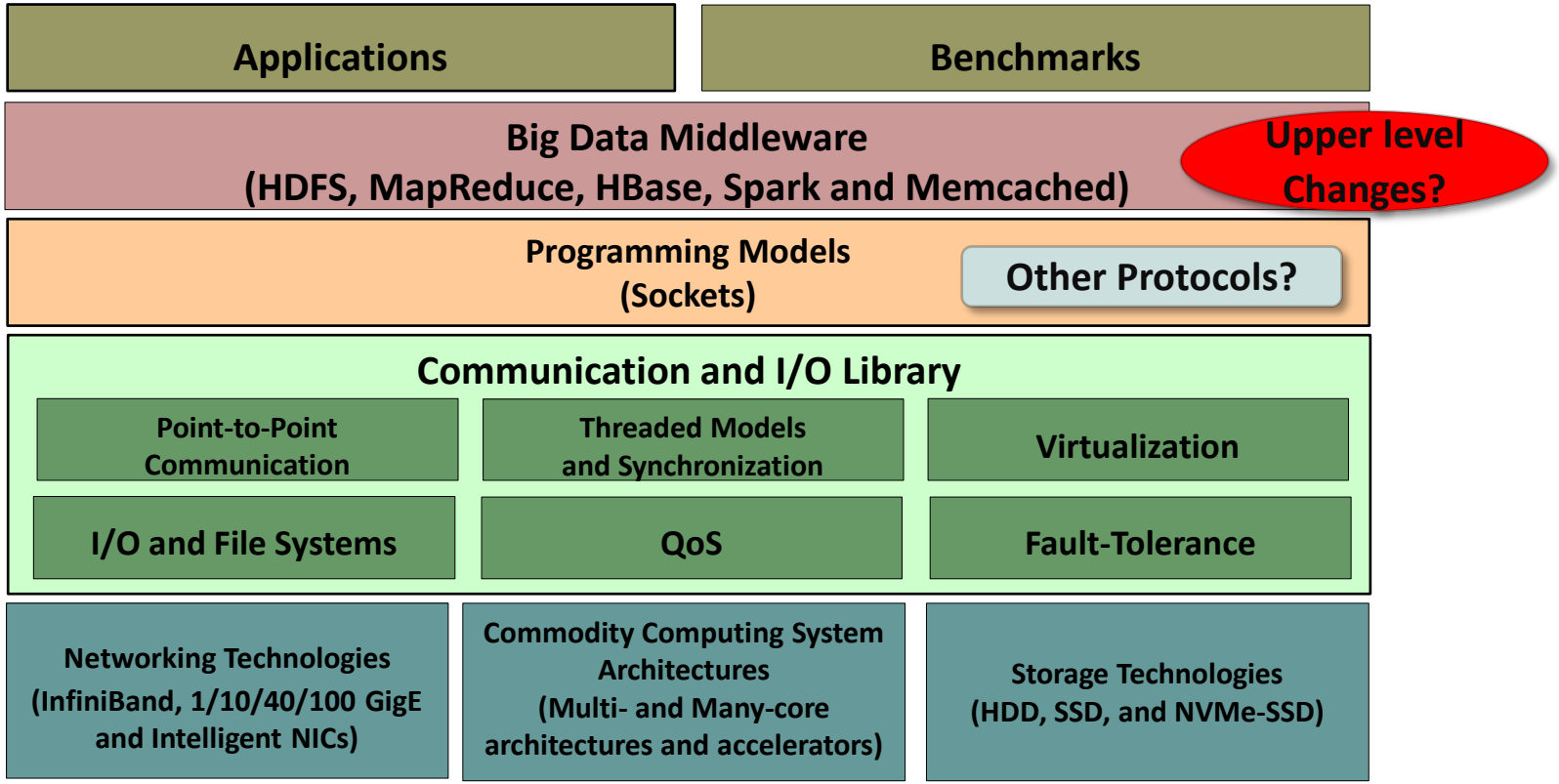
How much performance benefits can be achieved through enhanced designs?

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

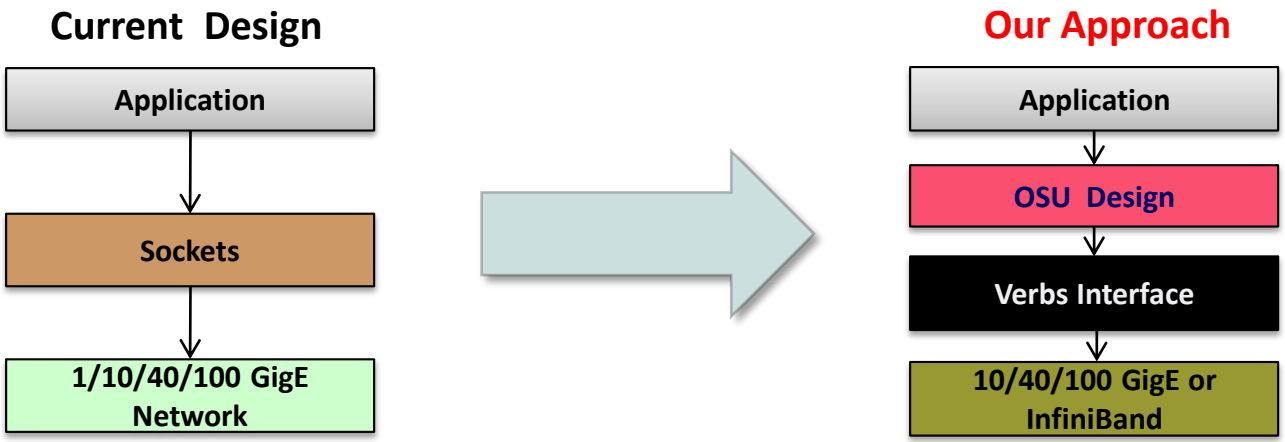
How to design benchmarks for evaluating the performance of Big Data middleware on HPC clusters?

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

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



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

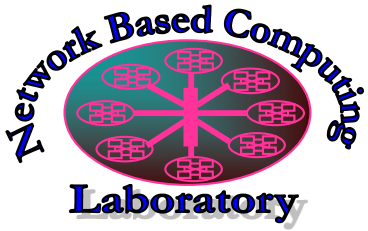


- Sockets not designed for high-performance
  - Stream semantics often mismatch for upper layers
  - Zero-copy not available for non-blocking sockets

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, and HBase Micro-benchmarks
- http://hibd.cse.ohio-state.edu
- Users Base: 195 organizations from 27 countries
- More than 18,600 downloads from the project site
- RDMA for Impala (upcoming)

Available for InfiniBand and RoCE

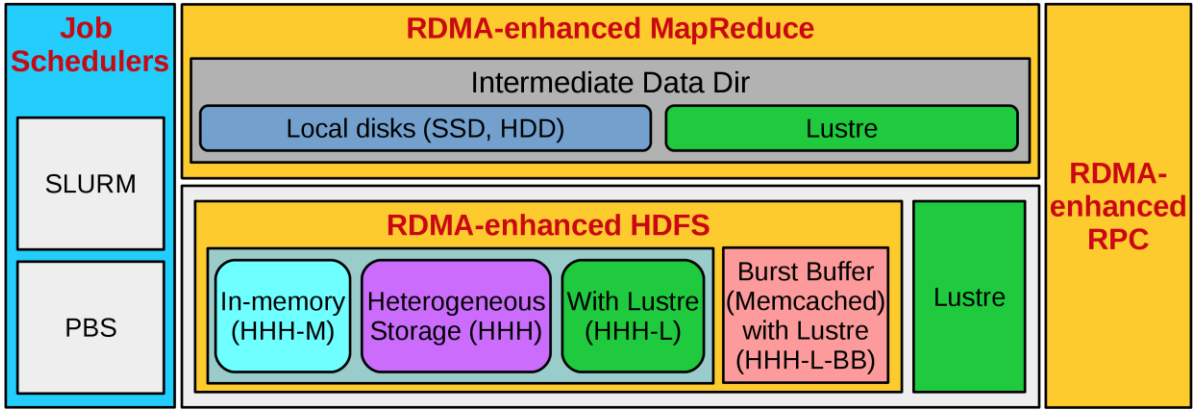


RDMA for Apache Hadoop 2.x Distribution

- 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
  - High performance 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
  - Easily configurable for different protocols (native InfiniBand, RoCE, and IPoIB)
- Current release: 0.9.1
  - 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

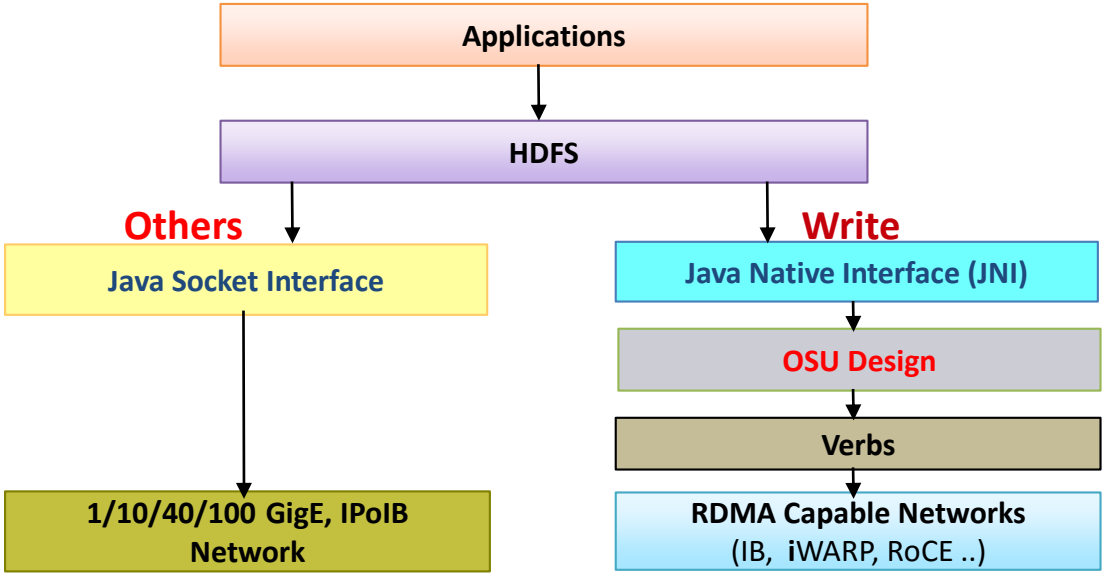
OSU HiBD Micro-Benchmark (OHB) Suite – HDFS, Memcached, and HBase

- 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
- Current release: 0.9.1
- http://hibd.cse.ohio-state.edu

Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS, MapReduce, and RPC
  - HBase
  - Spark
  - Memcached
- Advanced Designs
  - HDFS+Memcached-based Burst-Buffer
  - Memcached with Hybrid Memory and Non-blocking APIs
  - MR-Advisor
- BigData + HPC Cloud

Design Overview of HDFS with RDMA



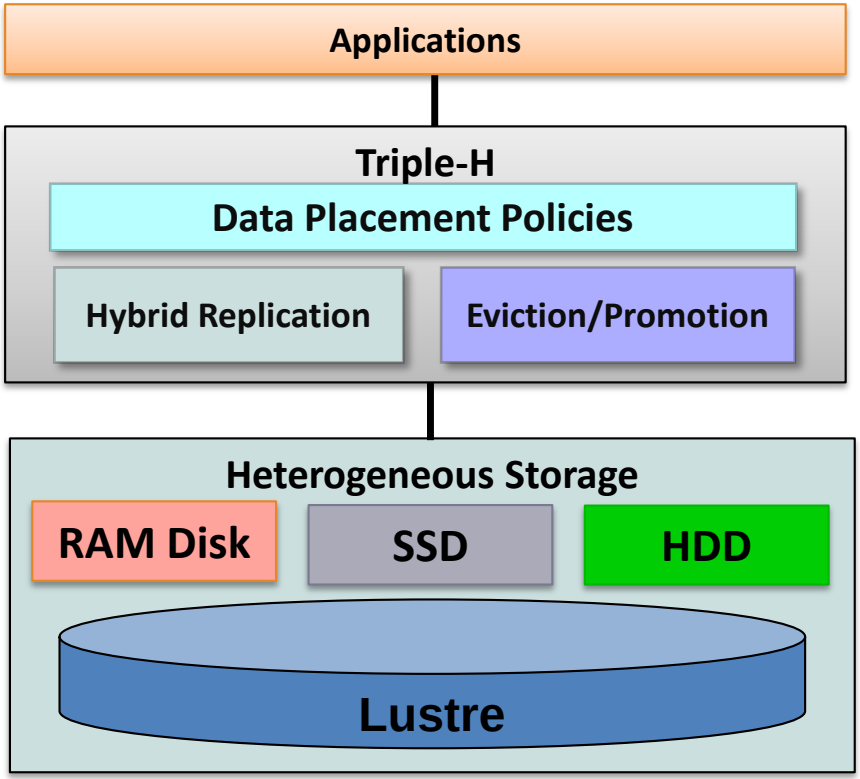
- Design Features
  - RDMA-based HDFS write
  - RDMA-based HDFS replication
  - Parallel replication support
  - On-demand connection setup
  - InfiniBand/RoCE support

- 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

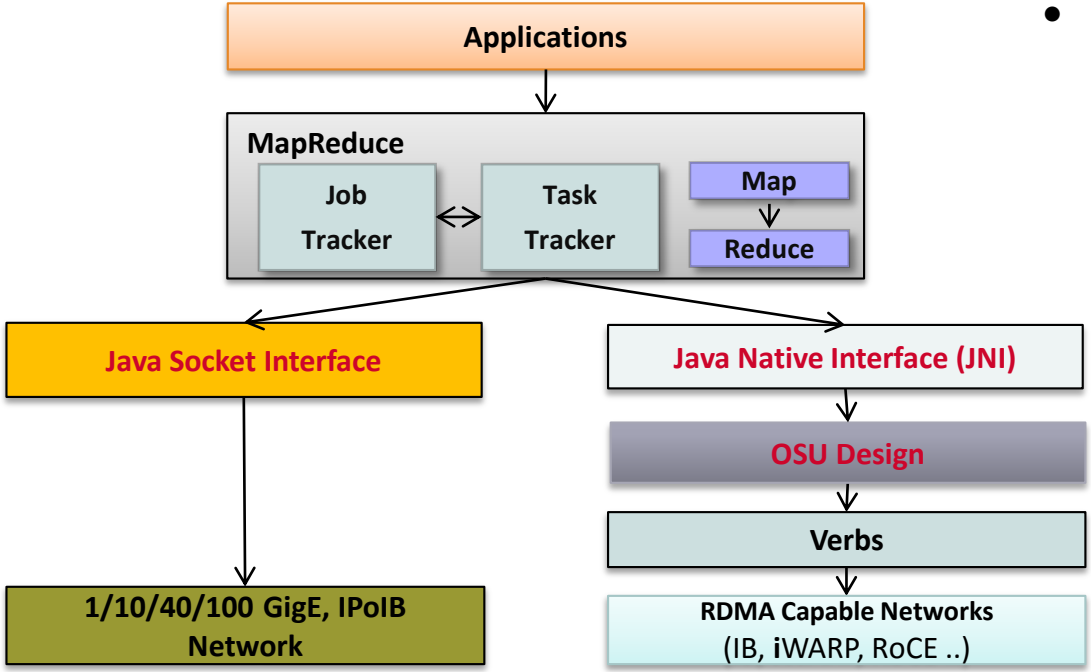
Enhanced HDFS with In-Memory and Heterogeneous Storage



- 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

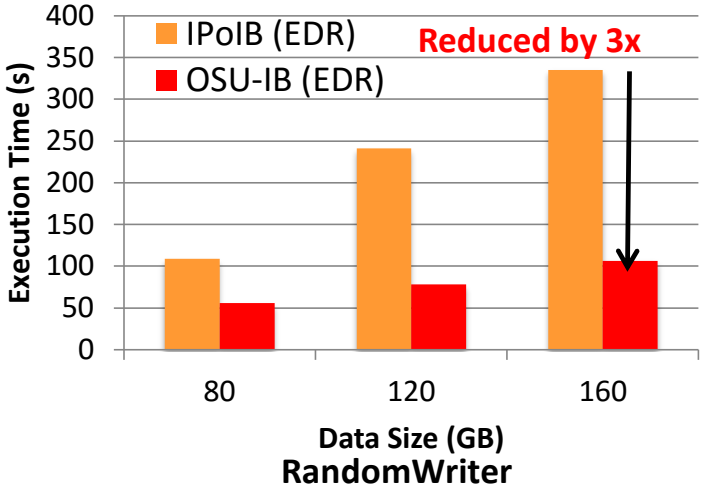


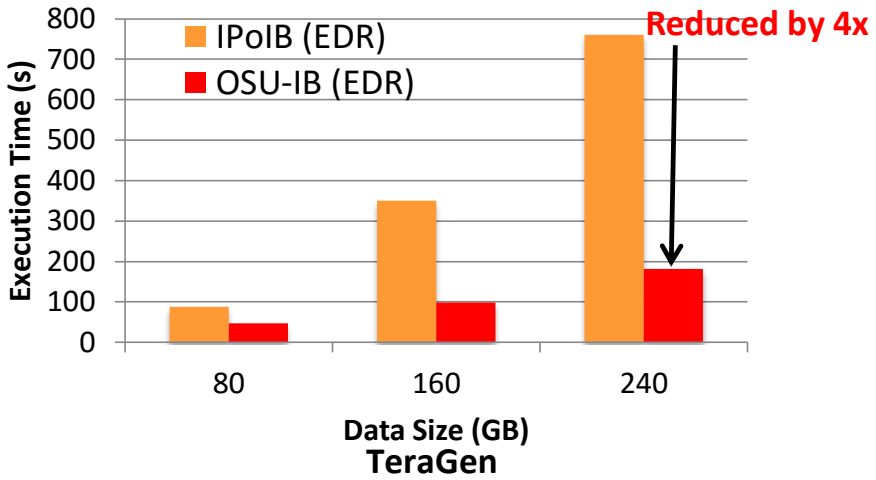
- Design Features
  - RDMA-based shuffle
  - Prefetching and caching map output
  - Efficient Shuffle Algorithms
  - In-memory merge
  - On-demand Shuffle Adjustment
  - Advanced overlapping
    - map, shuffle, and merge
    - shuffle, merge, and reduce
  - On-demand connection setup
  - InfiniBand/RoCE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

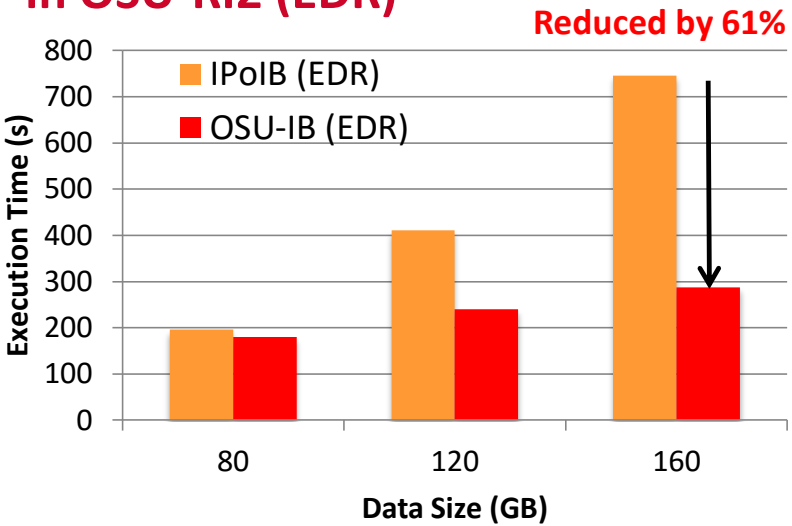




Cluster with 8 Nodes with a total of 64 maps

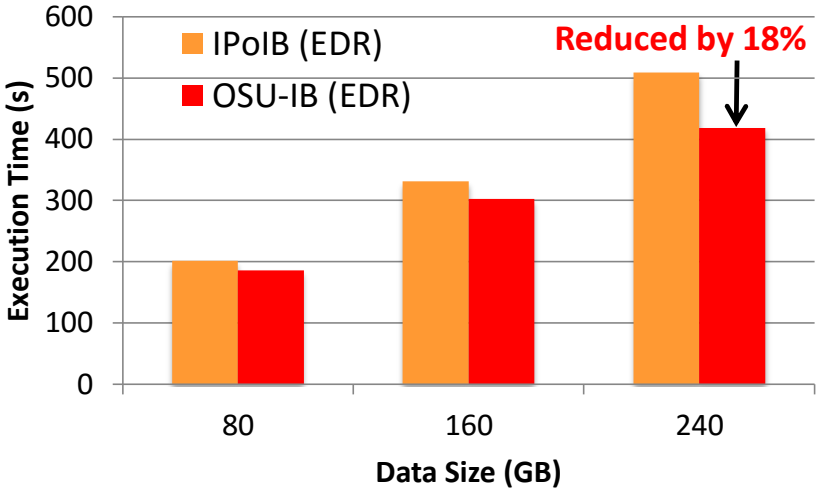
- RandomWriter
  - **3x** improvement over IPoIB for 80-160 GB file size
- TeraGen
  - **4x** improvement over IPoIB for 80-240 GB file size

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

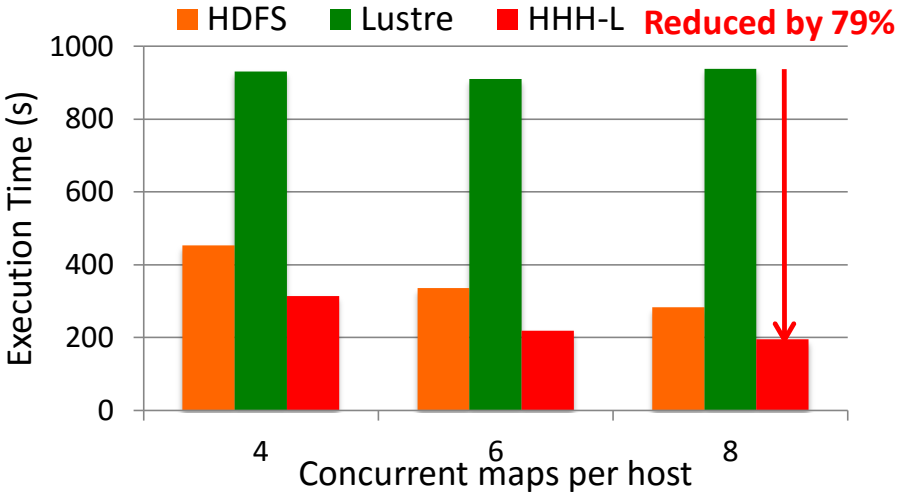


TeraSort

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



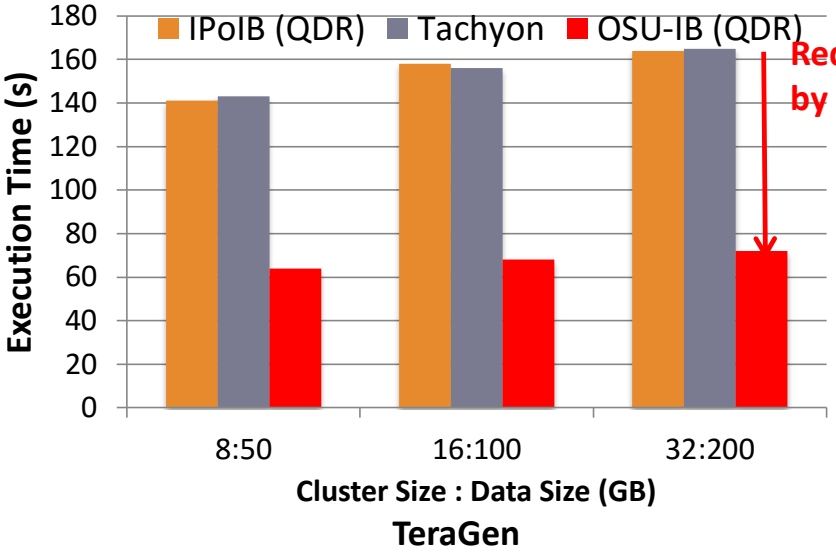
MR-MSPolyGraph

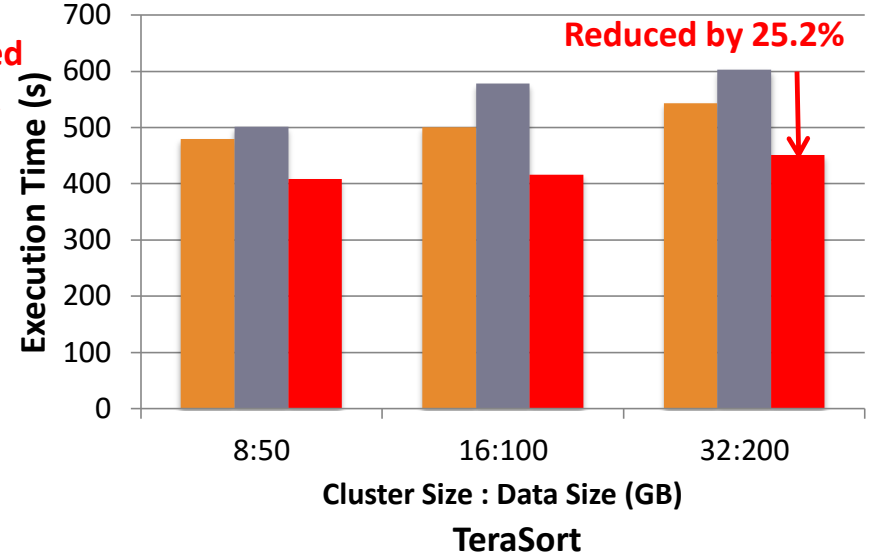
HDFS (FDR)	HHH (FDR)
60.24 s	48.3 s

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)





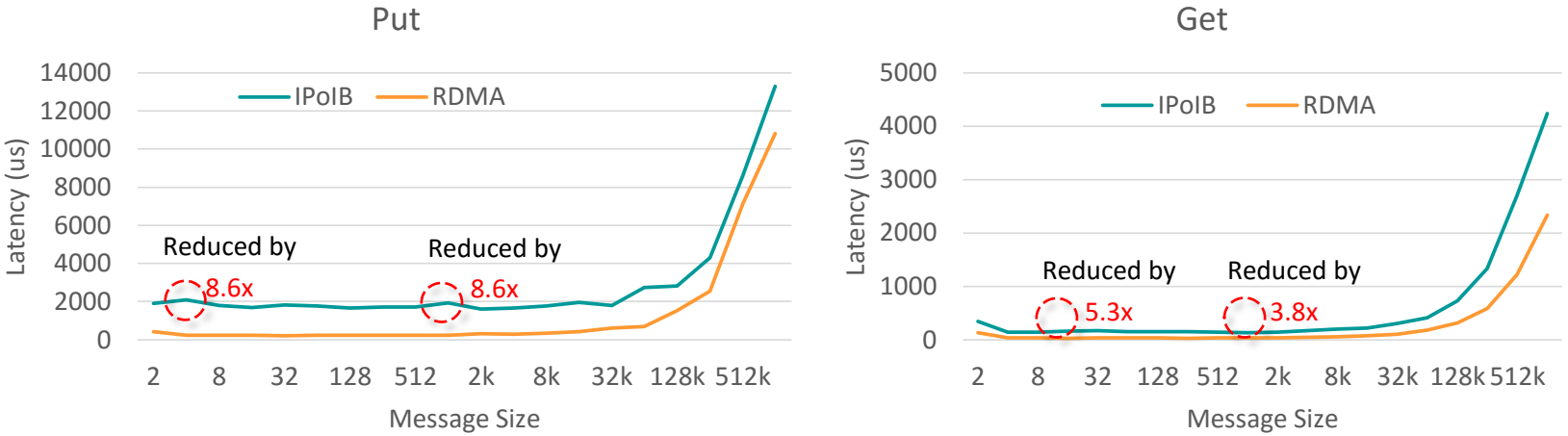
- For 200GB TeraGen on 32 nodes
  - Spark-TeraGen: HHH has 2.4x improvement over Tachyon; 2.3x over HDFS-IPoIB (QDR)
  - Spark-TeraSort: HHH has 25.2% improvement over Tachyon; 17% over HDFS-IPoIB (QDR)

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, MapReduce, and RPC
  - HBase
  - Spark
  - Memcached
- Advanced Designs
  - HDFS+Memcached-based Burst-Buffer
  - Memcached with Hybrid Memory and Non-blocking APIs
  - MR-Advisor
- BigData + HPC Cloud

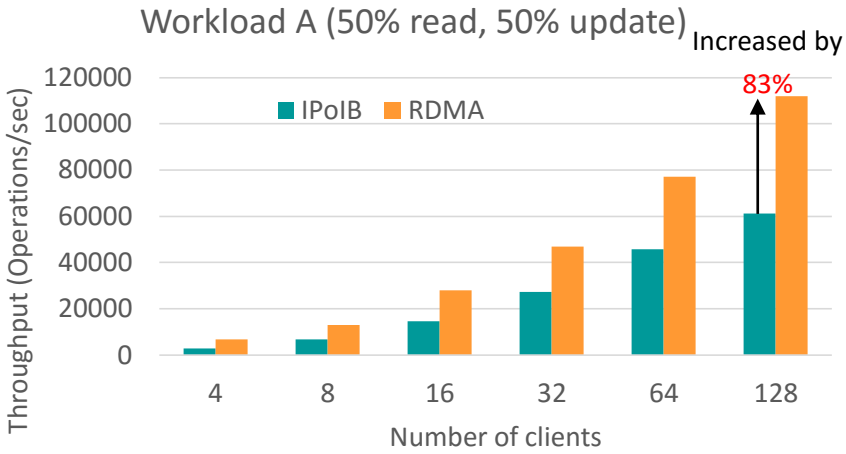
Performance Numbers of RDMA for Apache HBase – OHB in SDSC-Comet

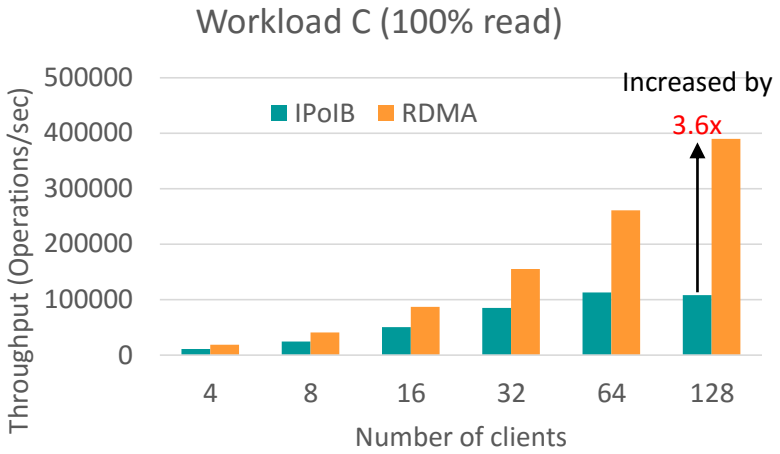


Evaluation with OHB Put and Get Micro-Benchmarks (1 Server, 1 Client)

- Up to **8.6x** improvement over IPoIB
- Up to **5.3x** improvement over IPoIB

Performance Numbers of RDMA for Apache HBase – YCSB in SDSC-Comet





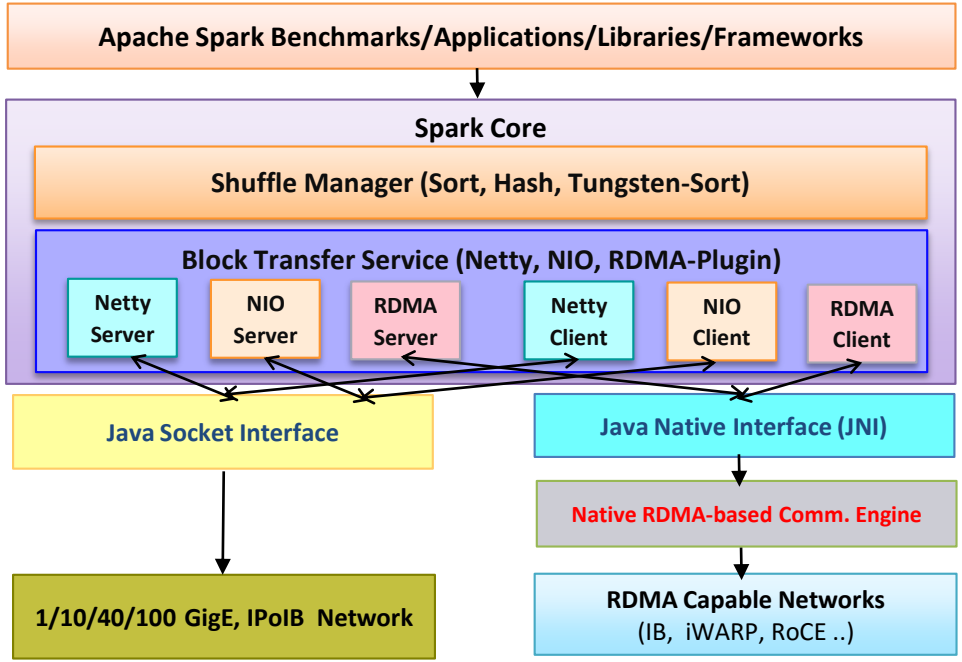
Evaluation with YCSB Workloads A and C (4 Servers)

- Up to **2.4x** improvement over IPoIB
- Up to **3.6x** improvement over IPoIB

Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS, MapReduce, and RPC
  - HBase
  - Spark
  - Memcached
- Advanced Designs
  - HDFS+Memcached-based Burst-Buffer
  - Memcached with Hybrid Memory and Non-blocking APIs
  - MR-Advisor
- BigData + HPC Cloud

Design Overview of Spark with RDMA



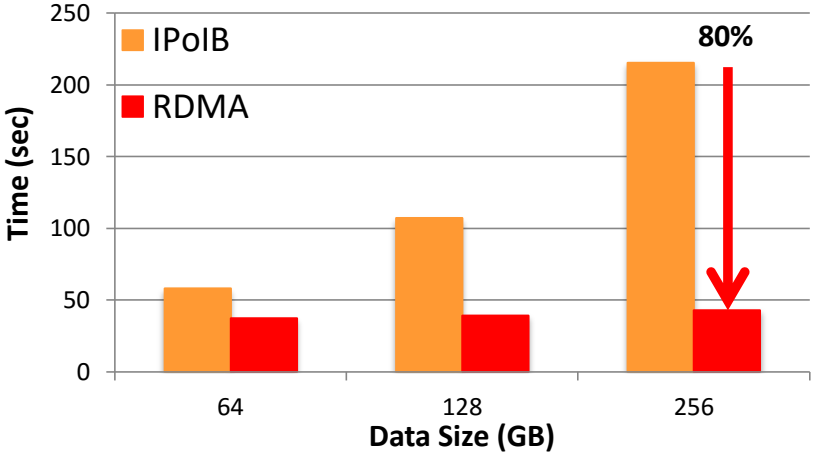
- Design Features
  - RDMA based shuffle plugin
  - SEDA-based architecture
  - Dynamic connection management and sharing
  - Non-blocking data transfer
  - Off-JVM-heap buffer management
  - InfiniBand/RoCE support

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

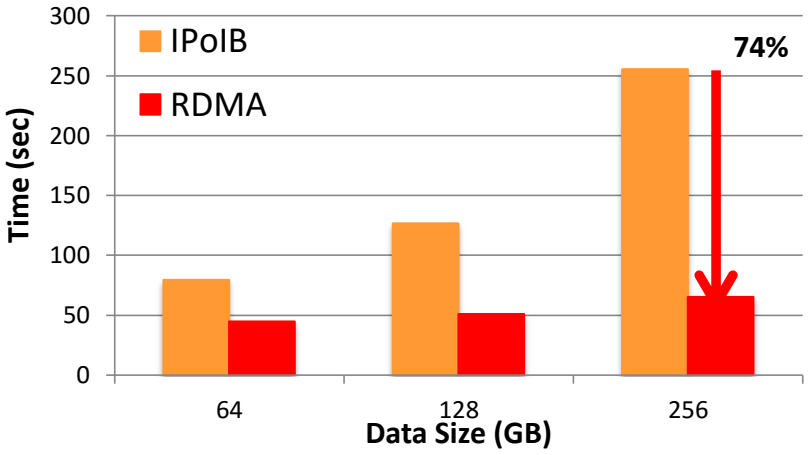
X. Lu, M. W. Rahman, N. Islam, D. Shankar, and D. K. Panda, Accelerating Spark with RDMA for Big Data Processing: Early Experiences, Int'l Symposium on High Performance Interconnects (HotI'14), August 2014

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



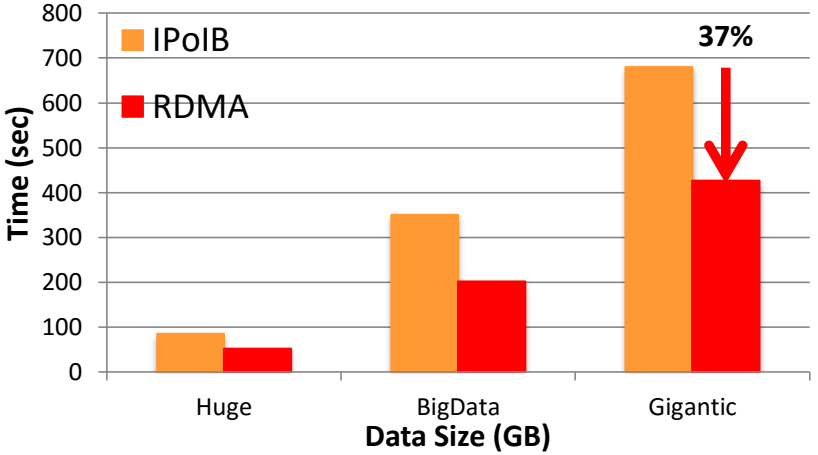
64 Worker Nodes, 1536 cores, SortByTest Total Time



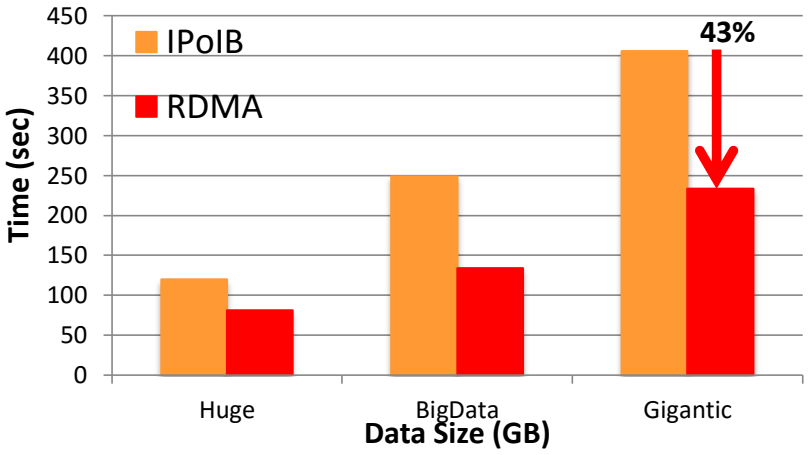
64 Worker Nodes, 1536 cores, GroupByTest Total Time

- 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

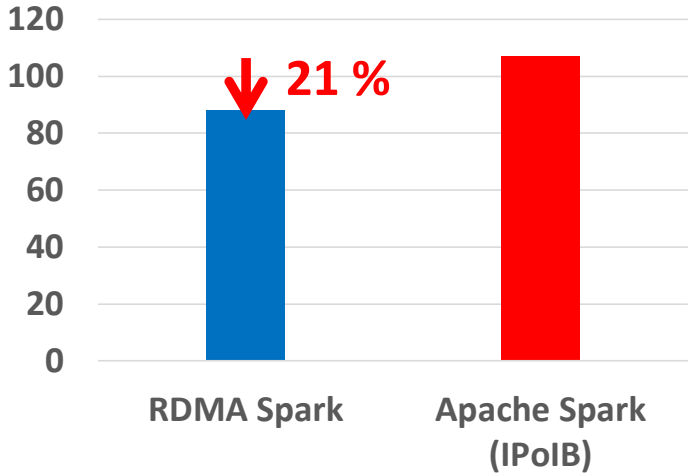


64 Worker Nodes, 1536 cores, PageRank Total Time

- 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.



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

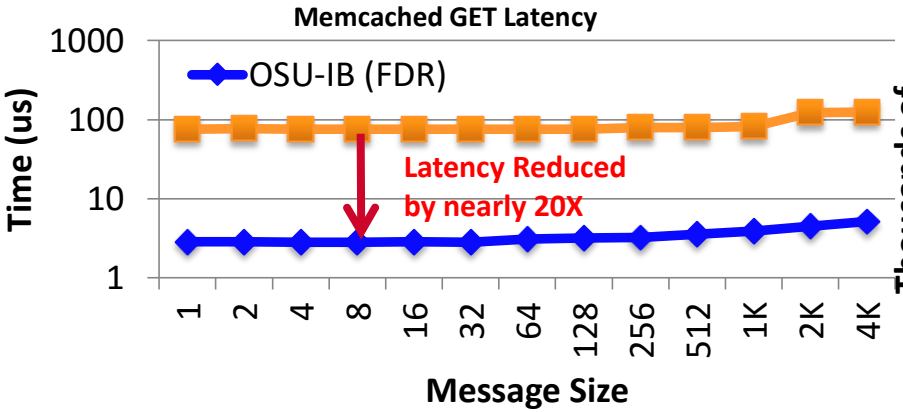
1. Z. Zhang, K. Barbary, F. A. Nothaft, E.R. Sparks, M.J. Franklin, D.A. Patterson, S. Perlmutter. Scientific Computing meets Big Data Technology: An Astronomy Use Case. *CoRR, vol: abs/1507.03325*, Aug 2015.

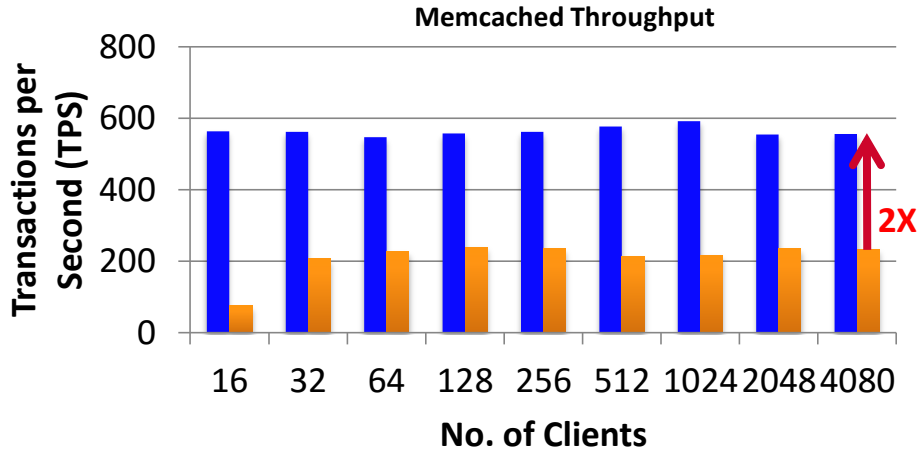
M. Tatineni, X. Lu, D. J. Choi, A. Majumdar, and D. K. Panda, Experiences and Benefits of Running RDMA Hadoop and Spark on SDSC Comet, XSEDE'16, July 2016

Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS, MapReduce, and RPC
  - HBase
  - Spark
  - Memcached
- Advanced Designs and Studies
  - HDFS+Memcached-based Burst-Buffer
  - Memcached with Hybrid Memory and Non-blocking APIs
  - MR-Advisor
- BigData + HPC Cloud

Memcached Performance (FDR Interconnect)





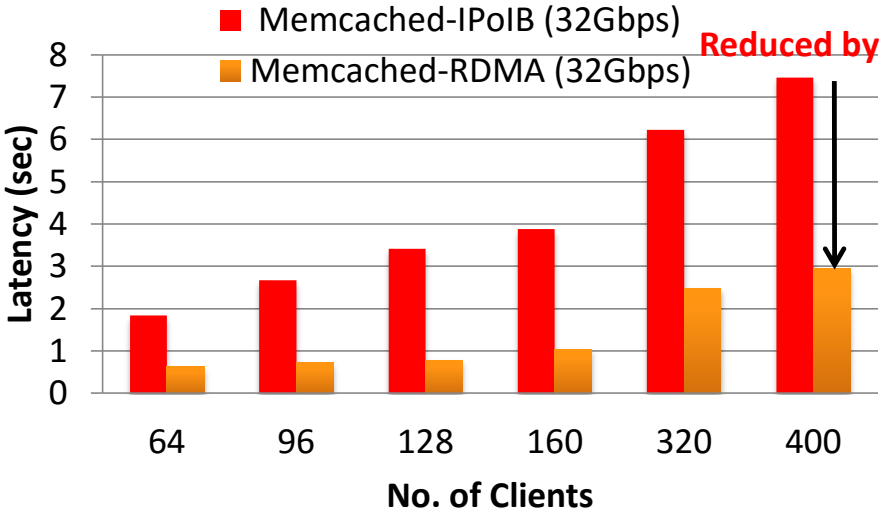
Experiments on TACC Stampede (Intel SandyBridge Cluster, IB: FDR)

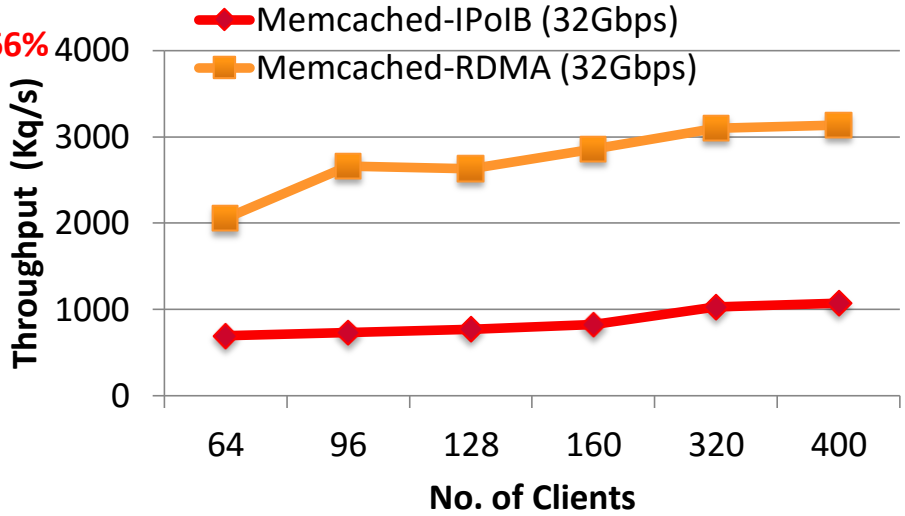
- Memcached Get latency
  - 4 bytes OSU-IB: 2.84 us; IPoIB: 75.53 us, 2K bytes OSU-IB: 4.49 us; IPoIB: 123.42 us
- Memcached Throughput (4bytes)
  - 4080 clients OSU-IB: 556 Kops/sec, IPoIB: 233 Kops/s, Nearly 2X improvement in throughput

J. Jose, H. Subramoni, M. Luo, M. Zhang, J. Huang, M. W. Rahman, N. Islam, X. Ouyang, H. Wang, S. Sur and D. K. Panda, Memcached Design on High Performance RDMA Capable Interconnects, ICPP'11

J. Jose, H. Subramoni, K. Kandalla, M. W. Rahman, H. Wang, S. Narravula, and D. K. Panda, Scalable Memcached design for InfiniBand Clusters using Hybrid Transport, CCGrid'12

Micro-benchmark Evaluation for OLDP workloads





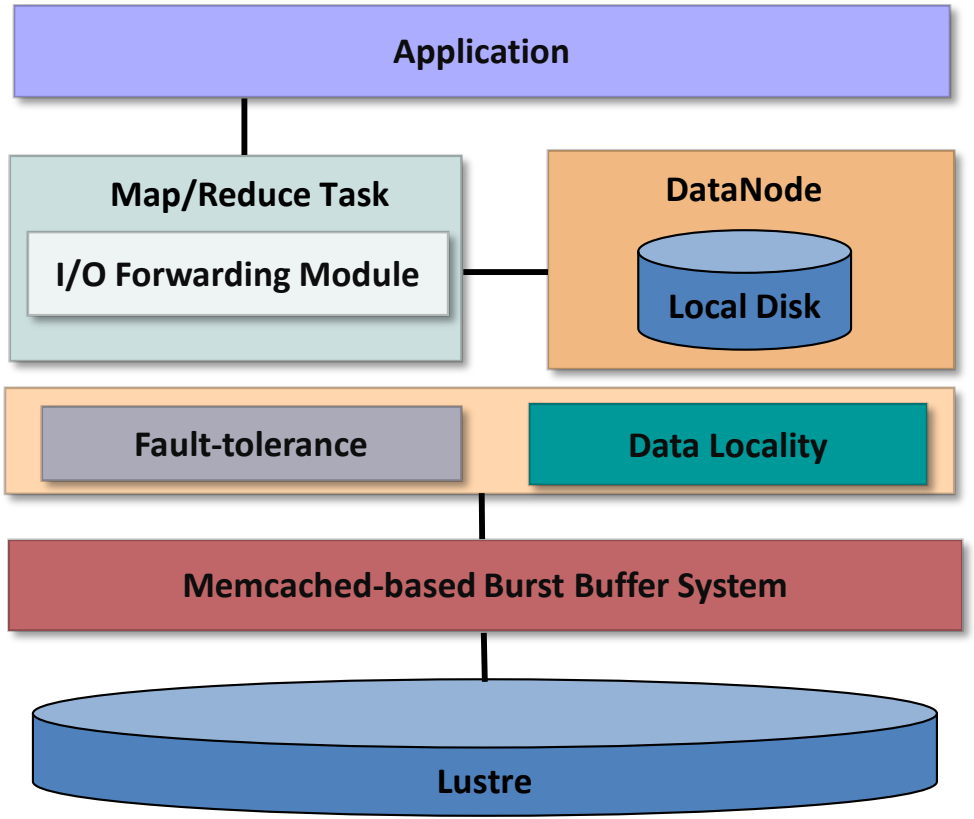
- Illustration with Read-Cache-Read access pattern using modified mysqlslap load testing tool
- Memcached-RDMA can
  - improve query latency by up to 66% over IPoIB (32Gbps)
  - throughput by up to 69% over IPoIB (32Gbps)

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

Acceleration Case Studies and Performance Evaluation

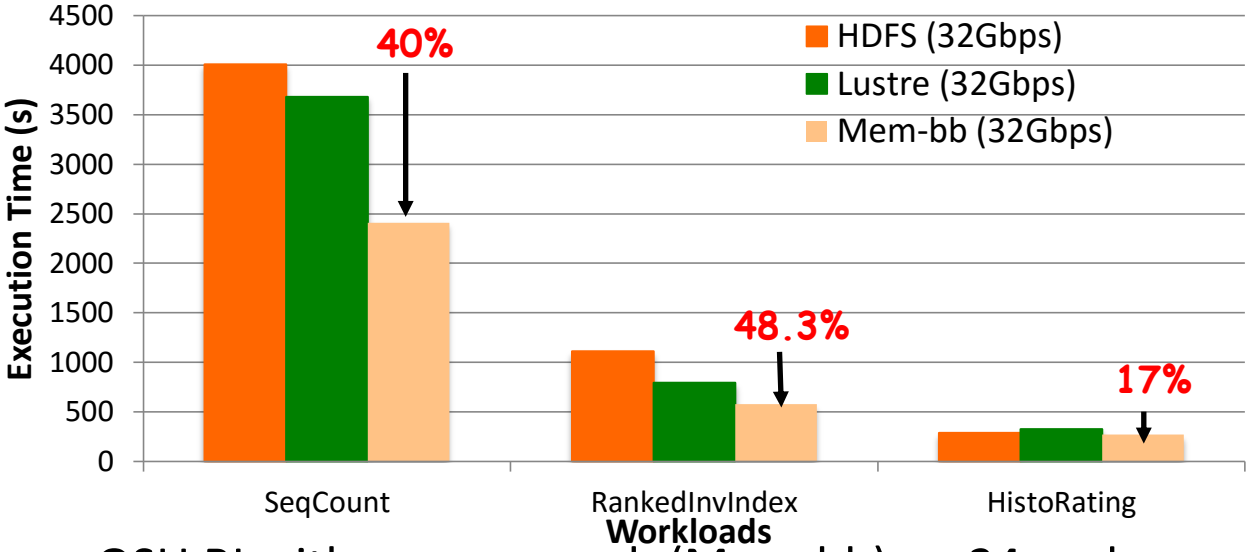
- Basic Designs
  - HDFS, MapReduce, and RPC
  - HBase
  - Spark
  - Memcached
- Advanced Designs
  - HDFS+Memcached-based Burst-Buffer
  - Memcached with Hybrid Memory and Non-blocking APIs
  - MR-Advisor
- BigData + HPC Cloud

Accelerating I/O Performance of Big Data Analytics through RDMA-based Key-Value Store



- 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

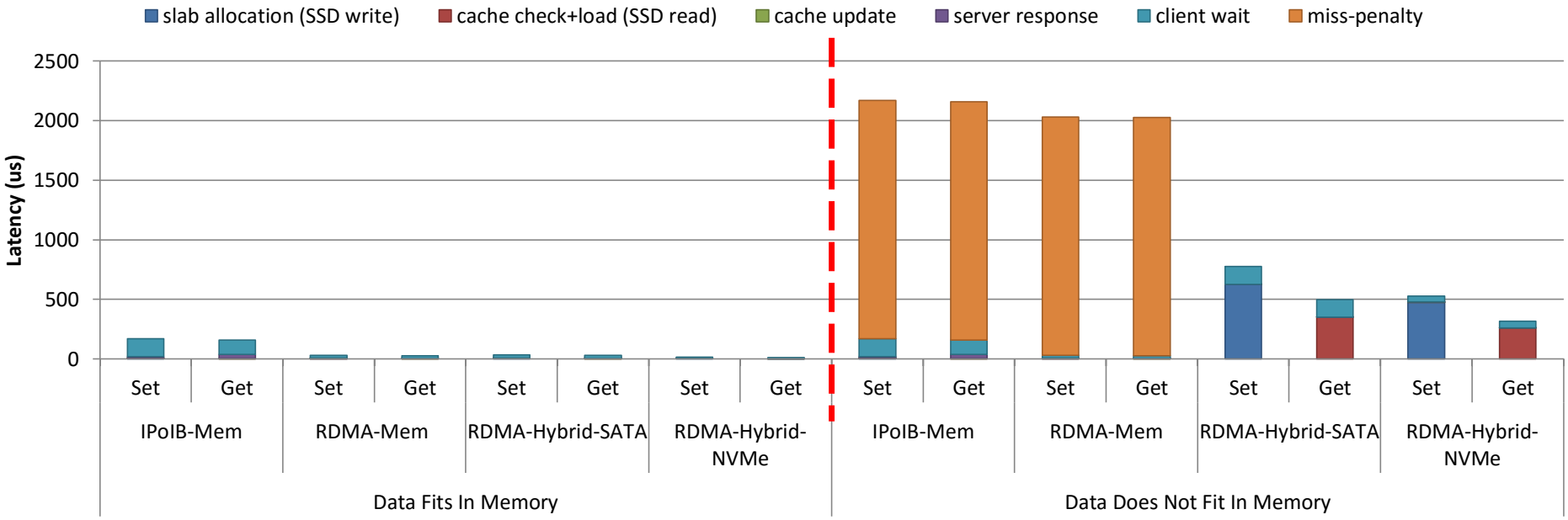
- SequenceCount: 34.5% over Lustre, 40% over HDFS
- RankedInvertedIndex: 27.3% over Lustre, 48.3% over HDFS
- HistogramRating: 17% over Lustre, 7% over HDFS

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

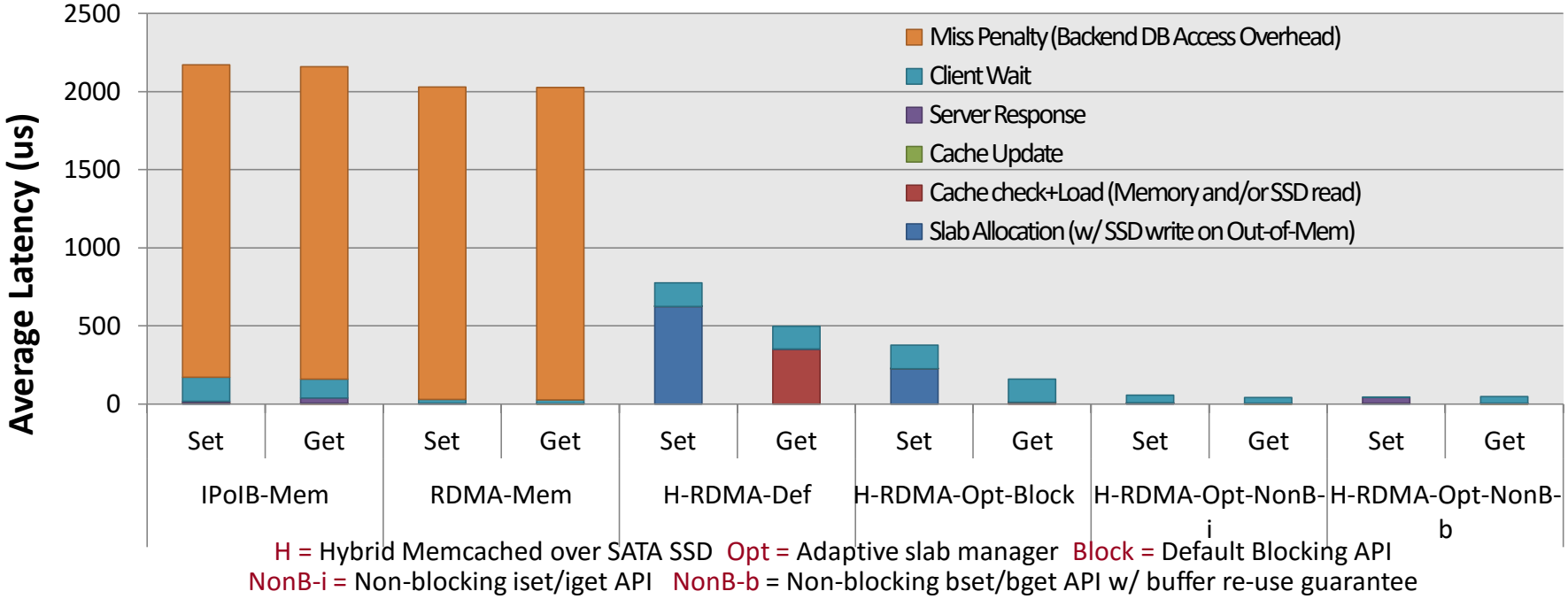
- Basic Designs
  - HDFS, MapReduce, and RPC
  - HBase
  - Spark
  - Memcached
- Advanced Designs
  - HDFS+Memcached-based Burst-Buffer
  - Memcached with Hybrid Memory and Non-blocking APIs
  - MR-Advisor
- BigData + HPC Cloud

Performance Evaluation on IB FDR + SATA/NVMe SSDs (Hybrid Memory)



- 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

Performance Evaluation with Non-Blocking Memcached API

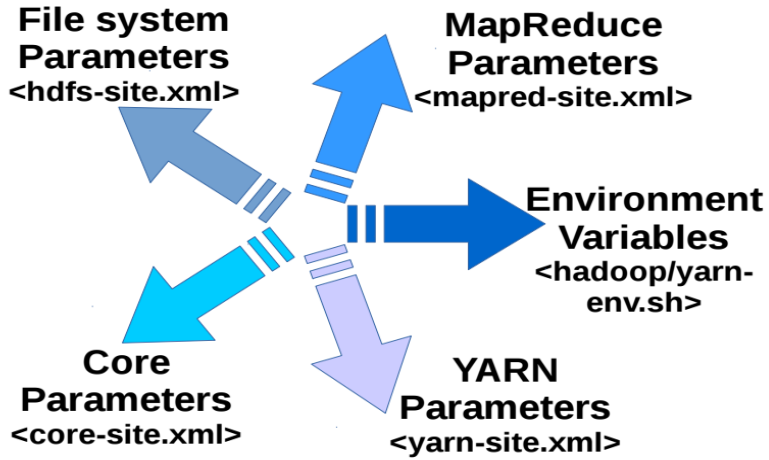


- Data does not fit in memory: Non-blocking Memcached Set/Get API Extensions can achieve
  - >16x latency improvement vs. blocking API over RDMA-Hybrid/RDMA-Mem w/ penalty
  - >2.5x throughput improvement vs. blocking API over default/optimized RDMA-Hybrid
- Data fits in memory: Non-blocking Extensions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

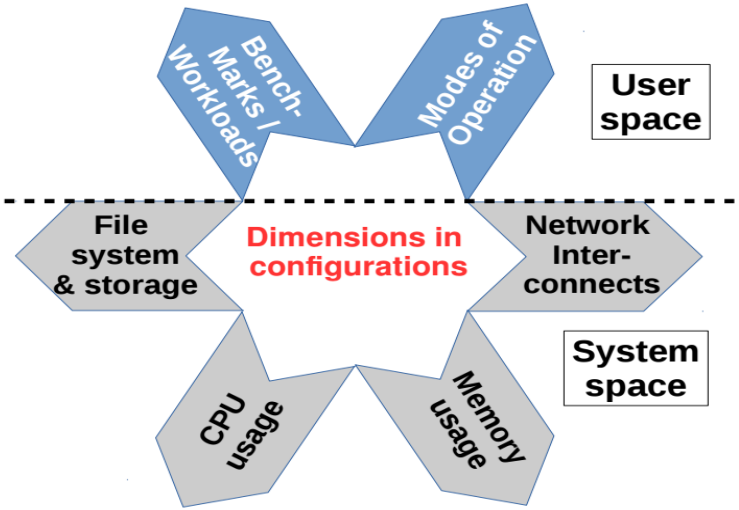
Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS, MapReduce, and RPC
  - HBase
  - Spark
  - Memcached
- Advanced Designs
  - HDFS+Memcached-based Burst-Buffer
  - Memcached with Hybrid Memory and Non-blocking APIs
  - MR-Advisor
- BigData + HPC Cloud

Challenges of Tuning and Profiling

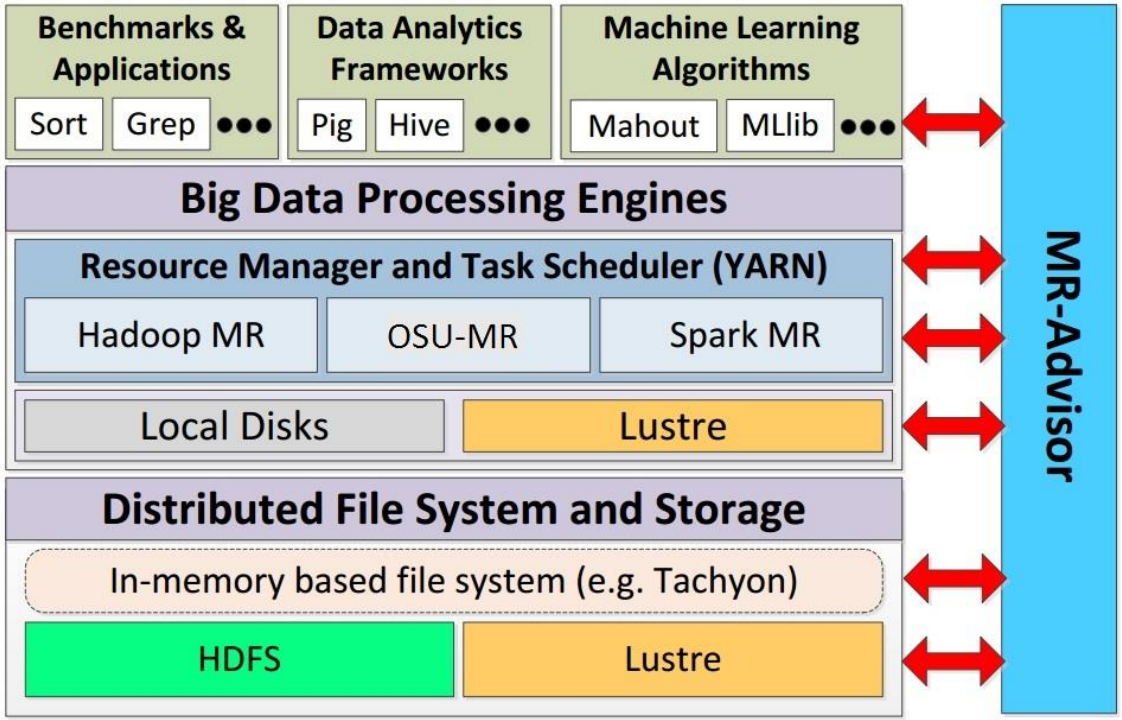


- MapReduce systems have different configuration parameters based on the underlying component that uses these
- The parameter files vary across different MapReduce stacks



- 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

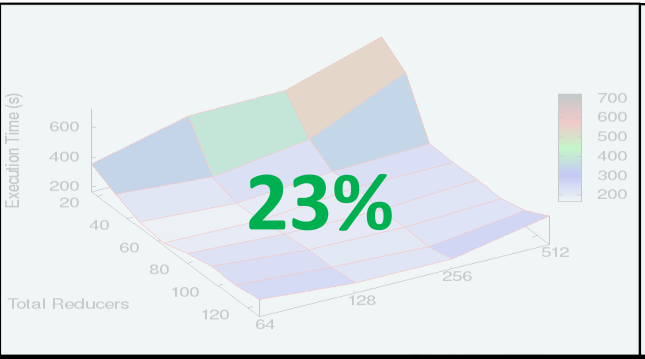
MR-Advisor Overview

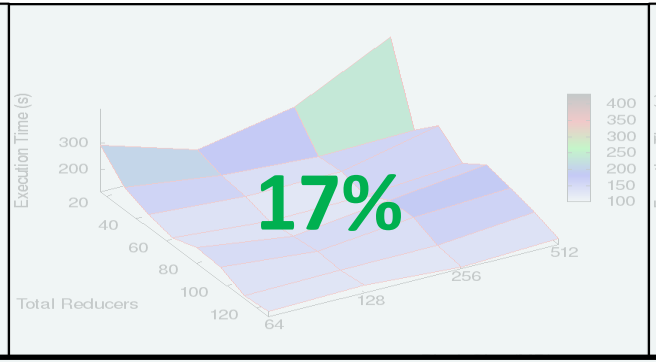


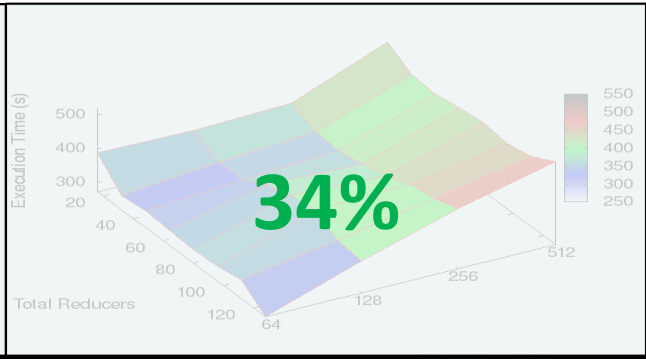
M. W. Rahman , N. S. Islam, X. Lu, D. Shankar, and D. K. Panda, *MR-Advisor: A Comprehensive Tuning Tool for Advising HPC Users to Accelerate MapReduce Applications on Supercomputers*, SBAC-PAD, 2016.

- 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

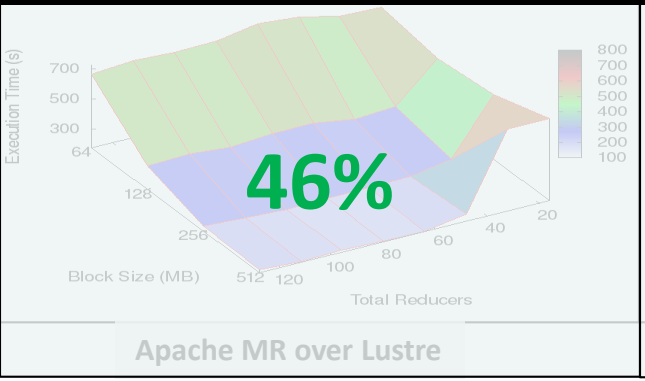
Tuning Experiments with MR-Advisor (TACC Stampede)

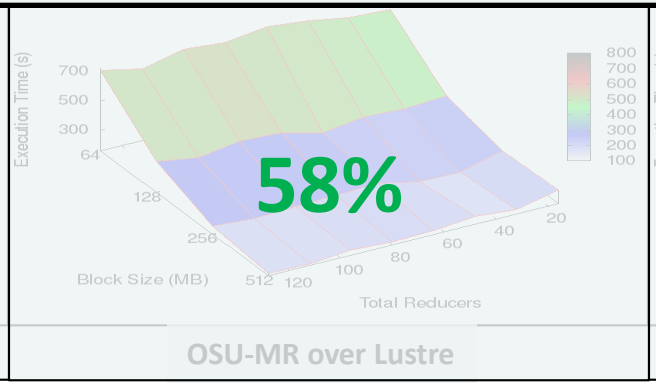


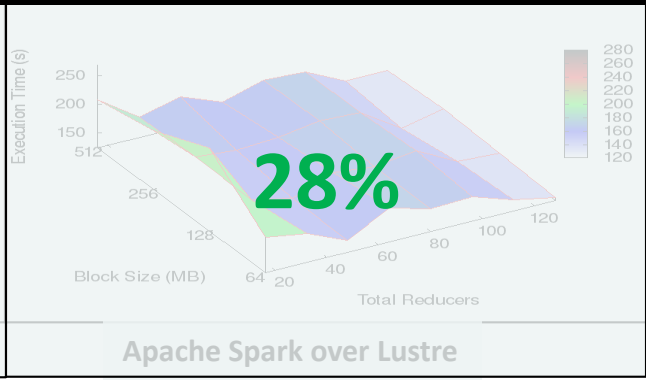




Performance improvements compared to current best practice values





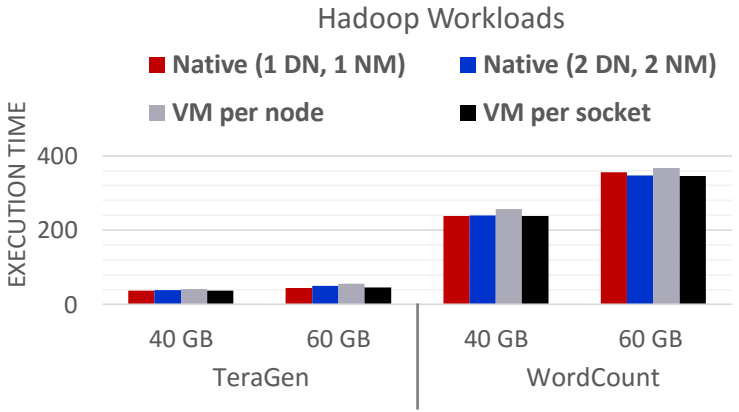


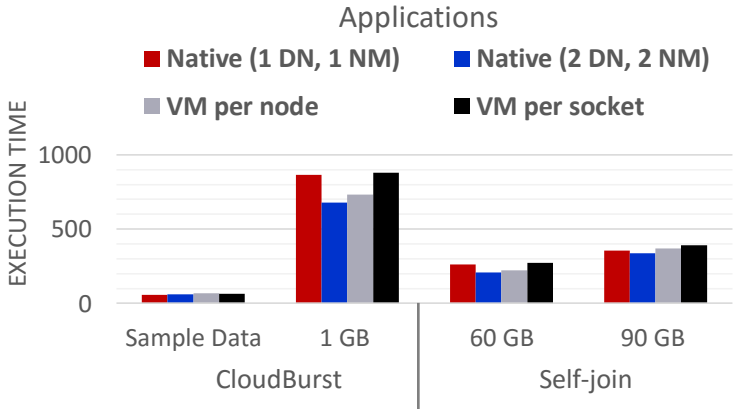
Acceleration Case Studies and Performance Evaluation

- Basic Designs
  - HDFS, MapReduce, and RPC
  - HBase
  - Spark
  - Memcached
- Advanced Designs
  - HDFS+Memcached-based Burst-Buffer
  - Memcached with Hybrid Memory and Non-blocking APIs
  - 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

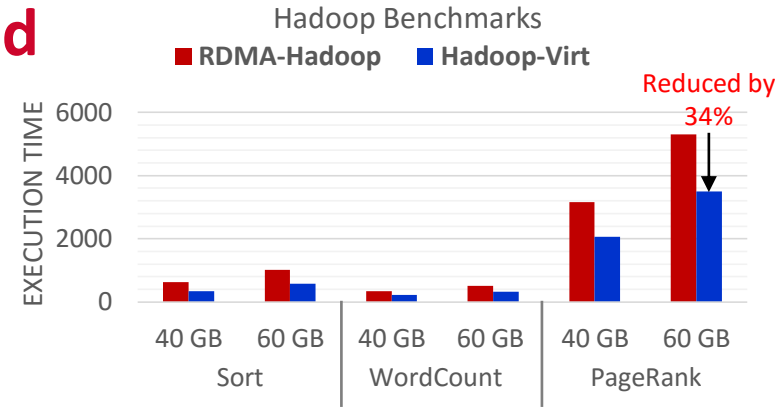


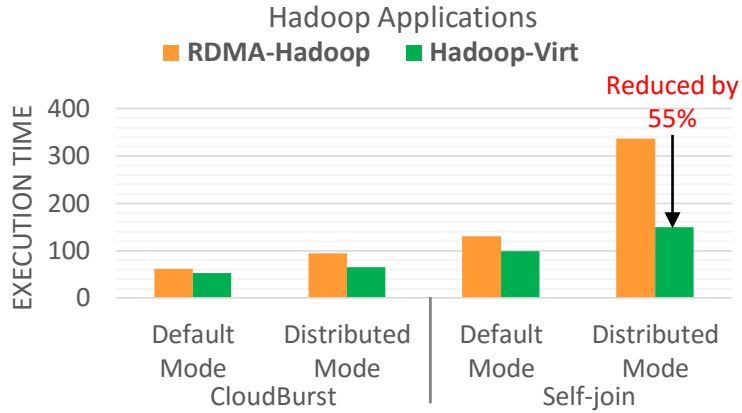


S. Gugnani, X. Lu, and D. K. Panda, Performance Characterization of Hadoop Workloads on SR-IOV-enabled Virtualized InfiniBand Clusters, accepted at BDCAT'16, December 2016

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

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 and Spark
- Advanced designs with upper-level changes and optimizations
  - Boldio (Burst Buffer with Memcached)
  - Efficient Indexing for HBase

Concluding Remarks

- Discussed challenges in accelerating Big Data middleware with HPC technologies
- Presented basic and advanced designs to take advantage of InfiniBand/RDMA for HDFS, MapReduce, RPC, HBase, Memcached, and Spark
- 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
- Looking forward to collaboration with the community

One More Presentation

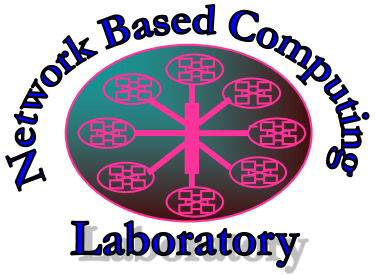
- Thursday (11/17/16) at 11:00am

MVAPICH2-GDR: Pushing the Frontier of HPC and Deep Learning

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/