Exploiting HPC Technologies to Accelerate Big Data Processing System Software: The HiBD Approach

Presentation at BoF-BigData of ISC '17

by

Xiaoyi Lu

The Ohio State University

E-mail: luxi@cse.ohio-state.edu
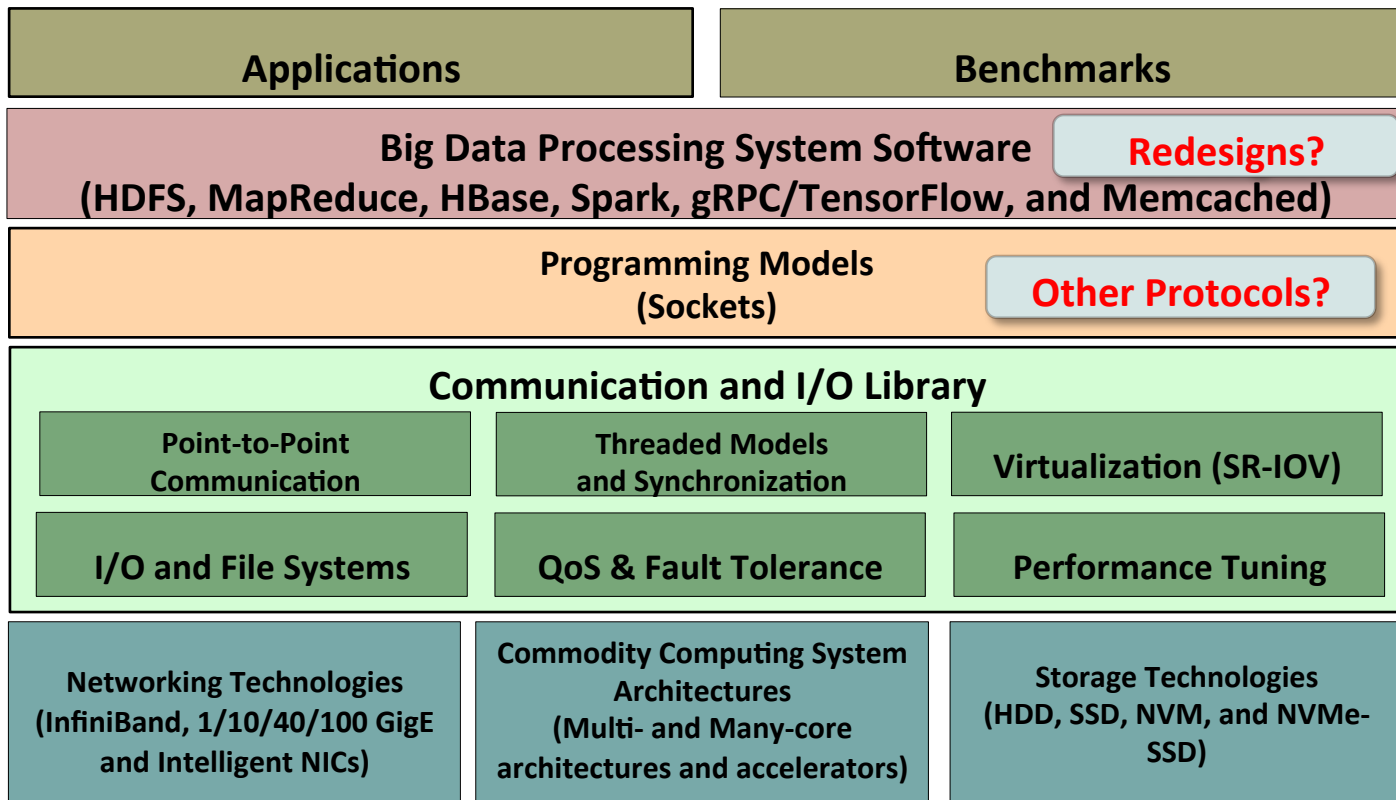
http://www.cse.ohio-state.edu/~luxi

Q1: What are the major technical challenges for accelerating current-generation big data processing system software?

Q2: What opportunities are provided for the HPC community to explore the (re-)design spaces of big data processing system software?
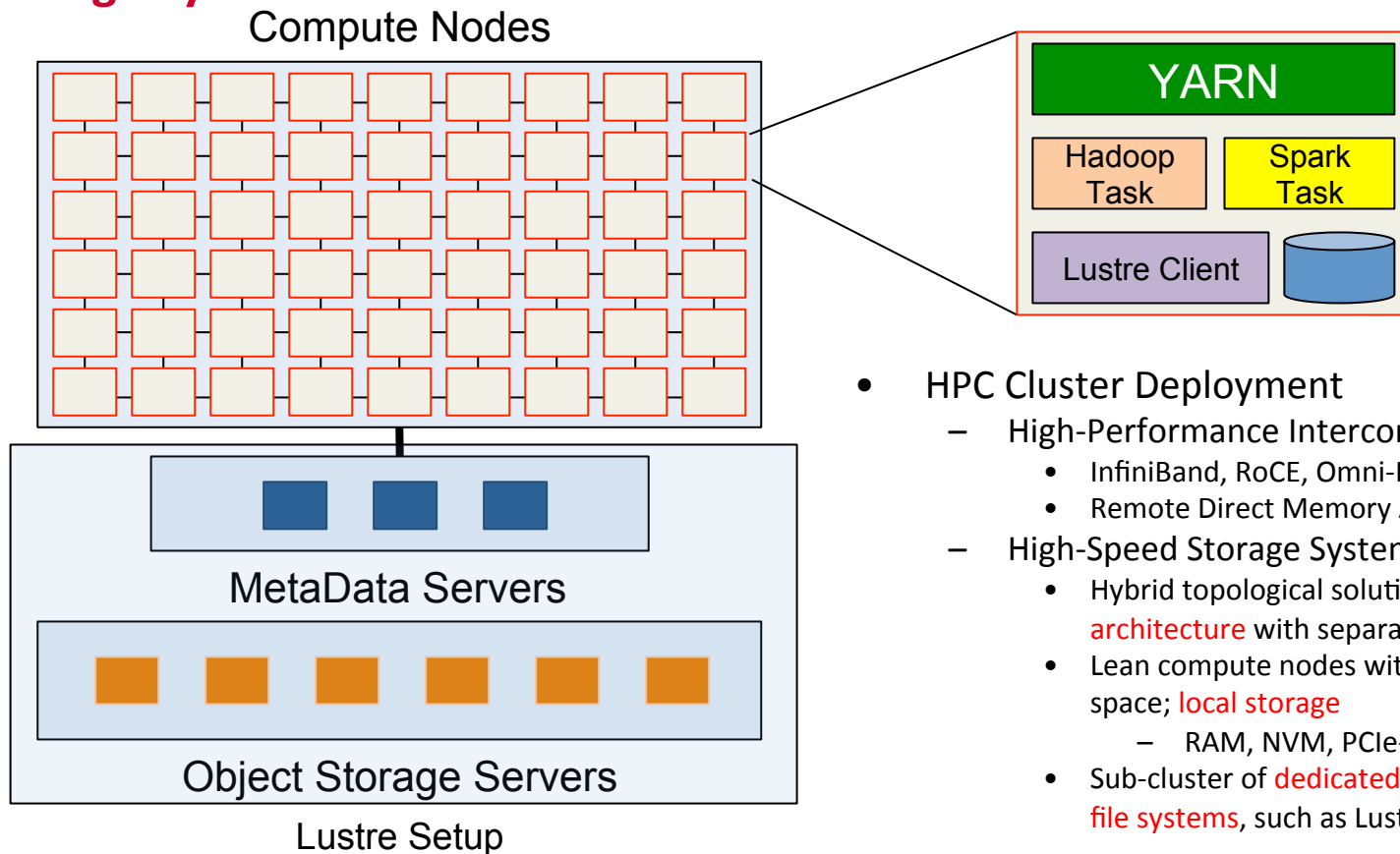
Q3: What highly demanded technologies are missing on modern HPC clusters to accelerate big data processing?

Q4: What are the main driving forces for designing next-generation big data system software?

Q1: Designing Communication and I/O Libraries for Big Data Processing System Software: Challenges

Applications	Benchmarks

Big Data Processing System Software   Redesigns?
(HDFS, MapReduce, HBase, Spark, gRPC/TensorFlow, 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)

Q2: Modern HPC Clusters with High-Performance Interconnects and High-Speed Storage Systems

Compute Nodes

YARN

Hadoop Task

Spark Task

Lustre Client

MetaData Servers

Object Storage Servers

Lustre Setup

- HPC Cluster Deployment
  - High-Performance Interconnects
    - InfiniBand, RoCE, Omni-Path
    - Remote Direct Memory Access (RDMA)
  - High-Speed Storage Systems
    - Hybrid topological solution of Beowulf architecture with separate I/O nodes
    - Lean compute nodes with light OS; more memory space; local storage
      - RAM, NVM, PCIe-/NVMe-SSD, HDD
    - Sub-cluster of dedicated I/O nodes with parallel file systems, such as Lustre

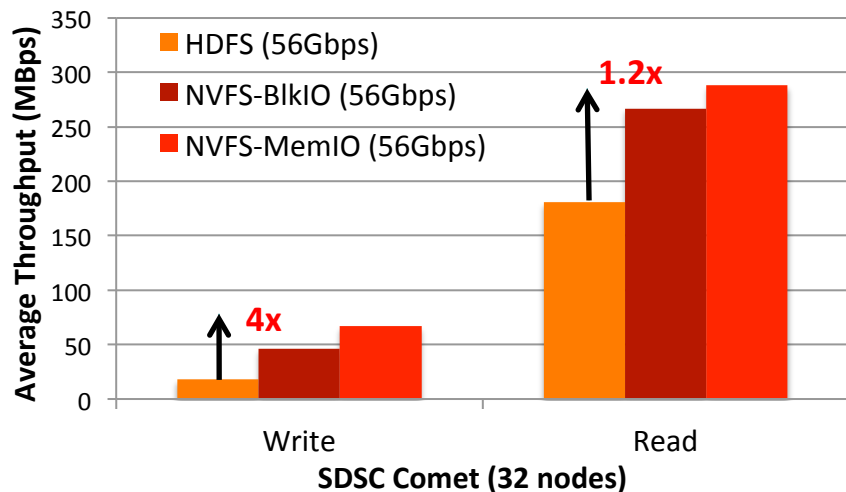# Q2: The High-Performance Big Data (HiBD) Project

- 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: 235 organizations from 30 countries

- More than 22,100 downloads from the project site

Available for InfiniBand and RoCE
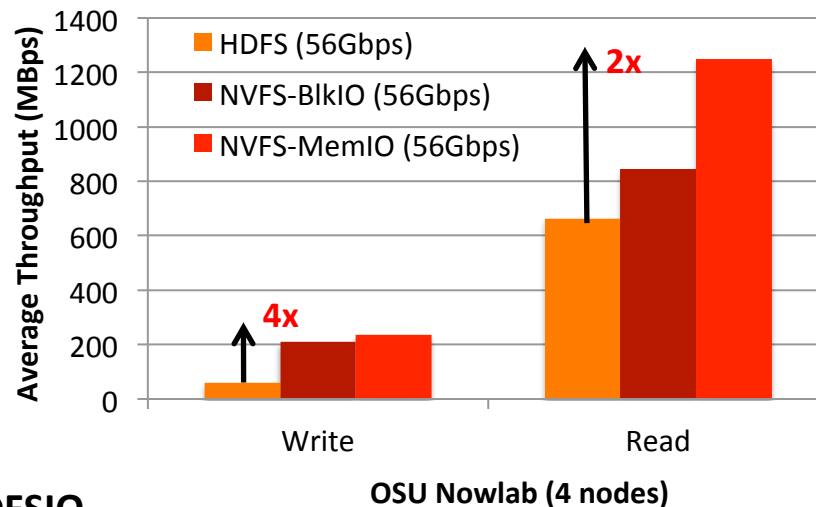
Also run on Ethernet
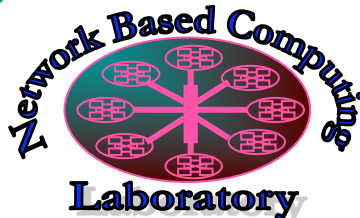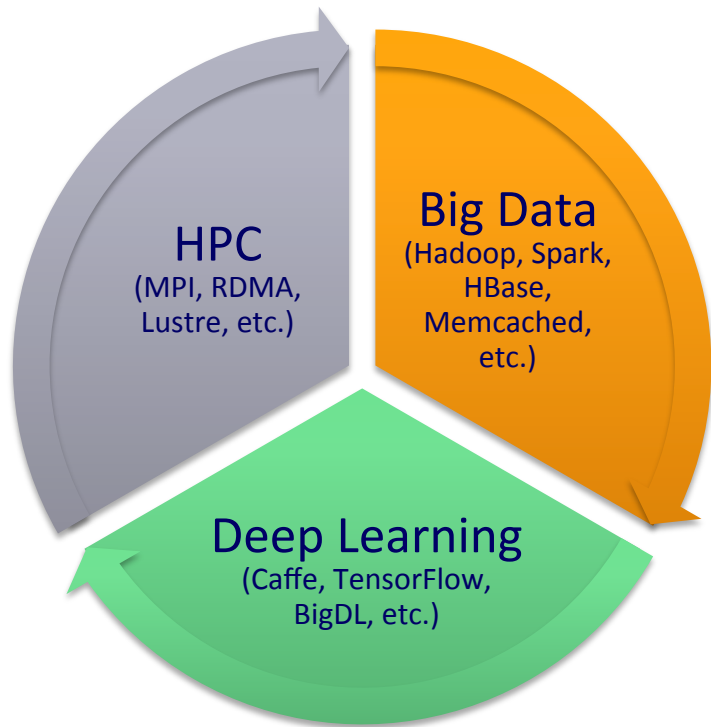
Q3: Missing Technologies or Features

- Large and high-speed local storage

  - Big Data processing systems typically need more high-speed local storage due to locality

- Availability of large-scale NVRAM deployment

  - NVRAM can benefit a lot of Big Data processing system software
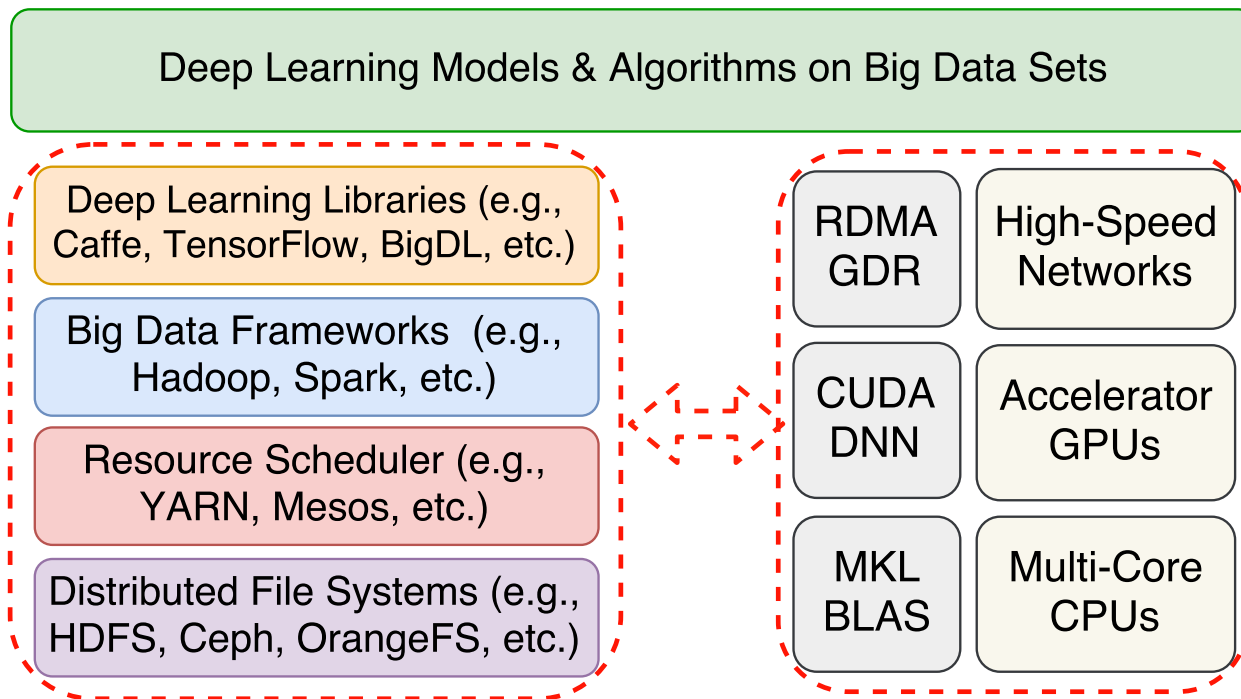

TestDFSIO

Q4: Increasing Usage of HPC, Big Data and Deep Learning

HPC
(MPI, RDMA, Lustre, etc.)

Big Data
(Hadoop, Spark, HBase, Memcached, etc.)

Deep Learning
(Caffe, TensorFlow, BigDL, etc.)

Convergence of HPC, Big Data, and Deep Learning!!!

Q4: High-Performance Deep Learning over Big Data (DLoBD) Stacks

Deep Learning Models & Algorithms on Big Data Sets

Deep Learning Libraries (e.g., Caffe, TensorFlow, BigDL, etc.)

Big Data Frameworks (e.g., Hadoop, Spark, etc.)

Resource Scheduler (e.g., YARN, Mesos, etc.)

Distributed File Systems (e.g., HDFS, Ceph, OrangeFS, etc.)

RDMA GDR

High-Speed Networks

CUDA DNN

Accelerator GPUs

MKL BLAS

Multi-Core CPUs

Xiaoyi Lu, Haiyang Shi, M. Haseeb Javed, Rajarshi Biswas, and Dhabaleswar K. (DK) Panda, Characterizing Deep Learning over Big Data (DLoBD) Stacks on RDMA-capable Networks, HotI 2017.
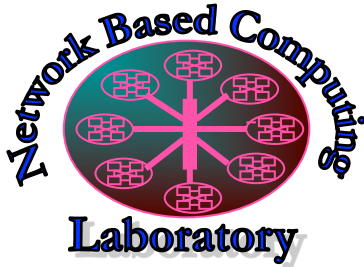
Q4: Open Challenges in Designing Next-Generation High-Performance Big Data Processing System Software

- High-Performance designs for Big Data middleware
  - NVM-aware communication and I/O schemes for Big Data
  - NVMe over Fabric Support
  - High-Bandwidth Memory support
  - Threaded Models and Synchronization
  - Locality-aware designs
- Fault-tolerance/resiliency
  - Live-migration support with virtual machines
  - Online data replication and coding
- Efficient data access and placement policies
- Efficient task scheduling
- Fast deployment and automatic configurations on Clouds
- Optimization for Deep Learning applications

Thank You!

luxi@cse.ohio-state.edu

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

Network-Based Computing Laboratory
http://nowlab.cse.ohio-state.edu/
The High-Performance Big Data Project
http://hibd.cse.ohio-state.edu/