Harp: Collective Communication on Hadoop

Judy Qiu, Indiana University

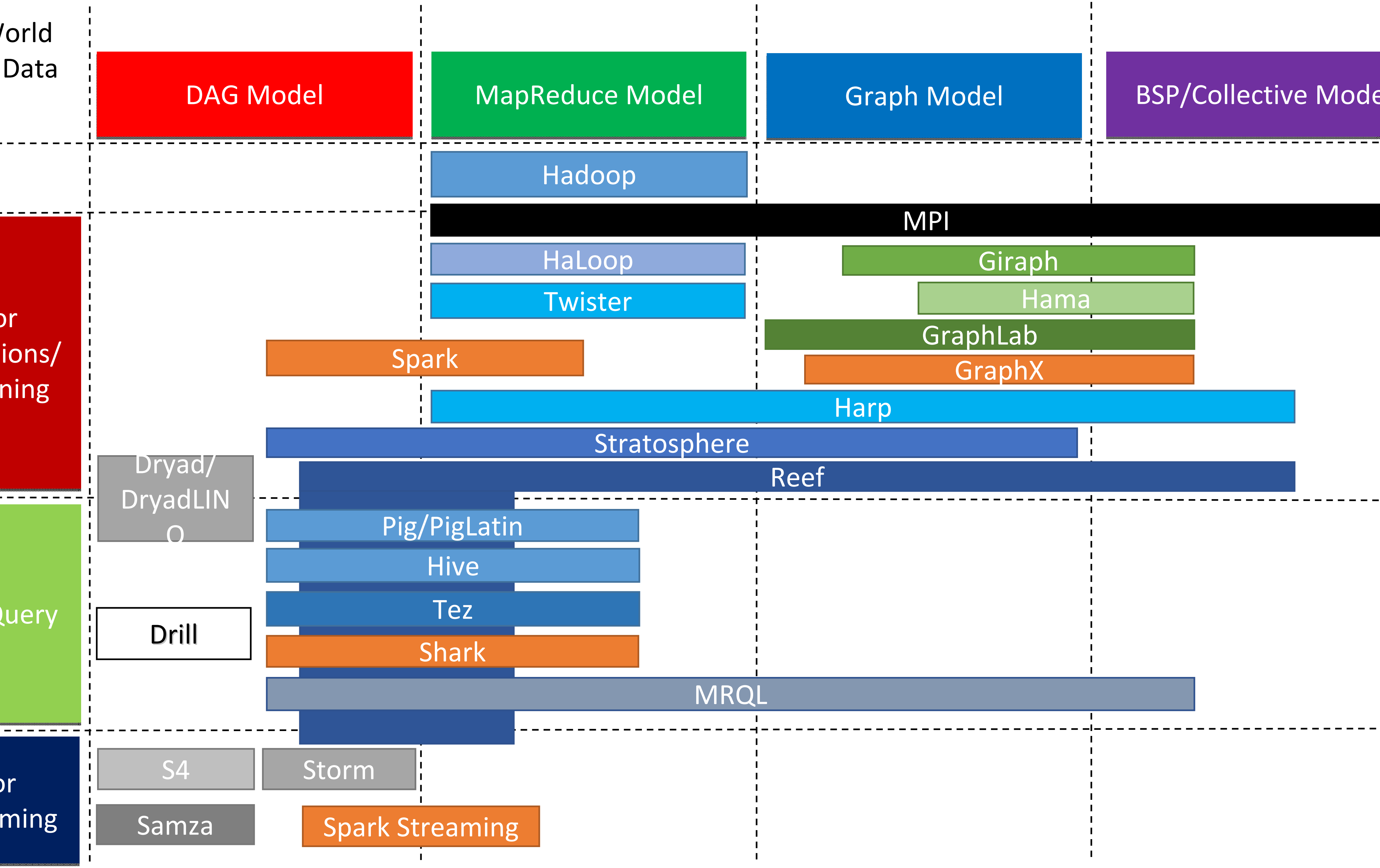
Outline

- Machine Learning on Big Data
- Big Data Tools
- Iterative MapReduce model
    - MDS Demo
- Harp

SA

Machine Learning on Big Data

- Mahout on Hadoop
  - https://mahout.apache.org/
- MLlib on Spark
  - http://spark.apache.org/mllib/
- GraphLab Toolkits
  - http://graphlab.org/projects/toolkits.html
  - GraphLab Computer Vision Toolkit



/orld
Data
DAG Model
MapReduce Model
Graph Model
BSP/Collective Mode
Hadoop
MPI
HaLoop
Giraph
Twister
Hama
or
ions/
ning
GraphLab
Spark
GraphX
Harp
Stratosphere
Dryad/
DryadLIN
Q
Reef
Pig/PigLatin
Hive
Query
Drill
Tez
Shark
MRQL
S4
Storm
or
ming
Samza
Spark Streaming

Big Data Tools for HPC and Supercomputing

- MPI(Message Passing Interface, 1992)
  - Provide standardized function interfaces for communication between parallel processes.

- Collective communication operations
  - Broadcast, Scatter, Gather, Reduce, Allgather, Allreduce, Reduce-scatter.

- Popular implementations
  - MPICH (2001)
  - OpenMPI (2004)
    - http://www.open-mpi.org/

MapReduce Model

Google MapReduce (2004)

- Jeffrey Dean et al. MapReduce: Simplified Data Processing on Large Clusters. OSDI 2004.

Apache Hadoop (2005)

- http://hadoop.apache.org/
- http://developer.yahoo.com/hadoop/tutorial/

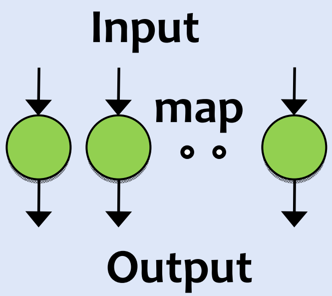
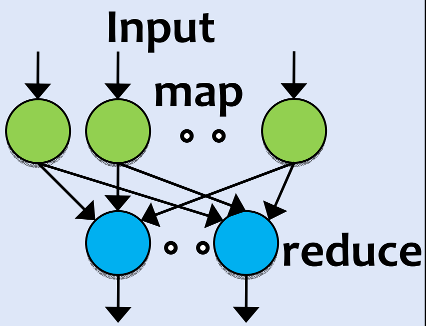
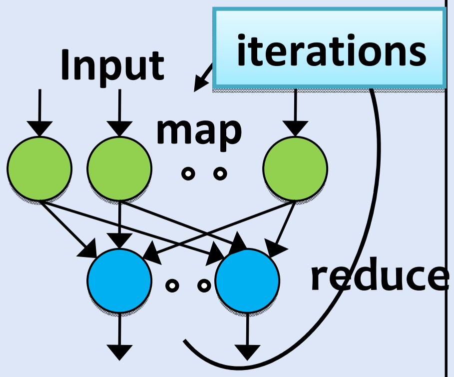
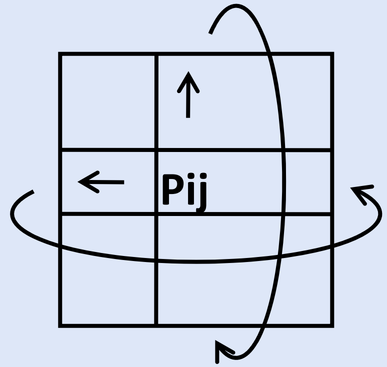
Apache Hadoop 2.0 (2012)

- Vinod Kumar Vavilapalli et al. Apache Hadoop YARN: Yet Another Resource Negotiator, SO 2013.
- Separation between resource management and computation model.

Key Features of MapReduce Model

- Designed for clouds
  - Large clusters of commodity machines
- Designed for big data
  - Support from local disks based distributed file system (GFS / HDFS)
  - Disk based intermediate data transfer in Shuffling
- MapReduce programming model
  - Computation pattern: Map tasks and Reduce tasks
  - Data abstraction: KeyValue pairs

Applications & Different Interconnection Patterns

(a) Map Only (Pleasingly Parallel)	(b) Classic MapReduce	(c) Iterative MapReduce	(d) Loosely Synchronous
 	 	 	
- CAP3 Gene Analysis -Smith-Waterman Distances - Document conversion (PDF -> HTML) - Brute force searches in cryptography - Parametric sweeps - PolarGrid MATLAB data analysis	- High Energy Physics (HEP) Histograms - Distributed search - Distributed sorting - Information retrieval - Calculation of Pairwise Distances for sequences (BLAST)	- Expectation maximization algorithms - Linear Algebra - Data mining, includes K-means clustering - Deterministic Annealing Clustering - Multidimensional Scaling (MDS) - PageRank	Many MPI scientific applications utilizing wide variety of communication constructs, including local interactions - Solving Differential Equations and particle dynamics with short range forces
No Communication	**Collective Communication**		**MPI**

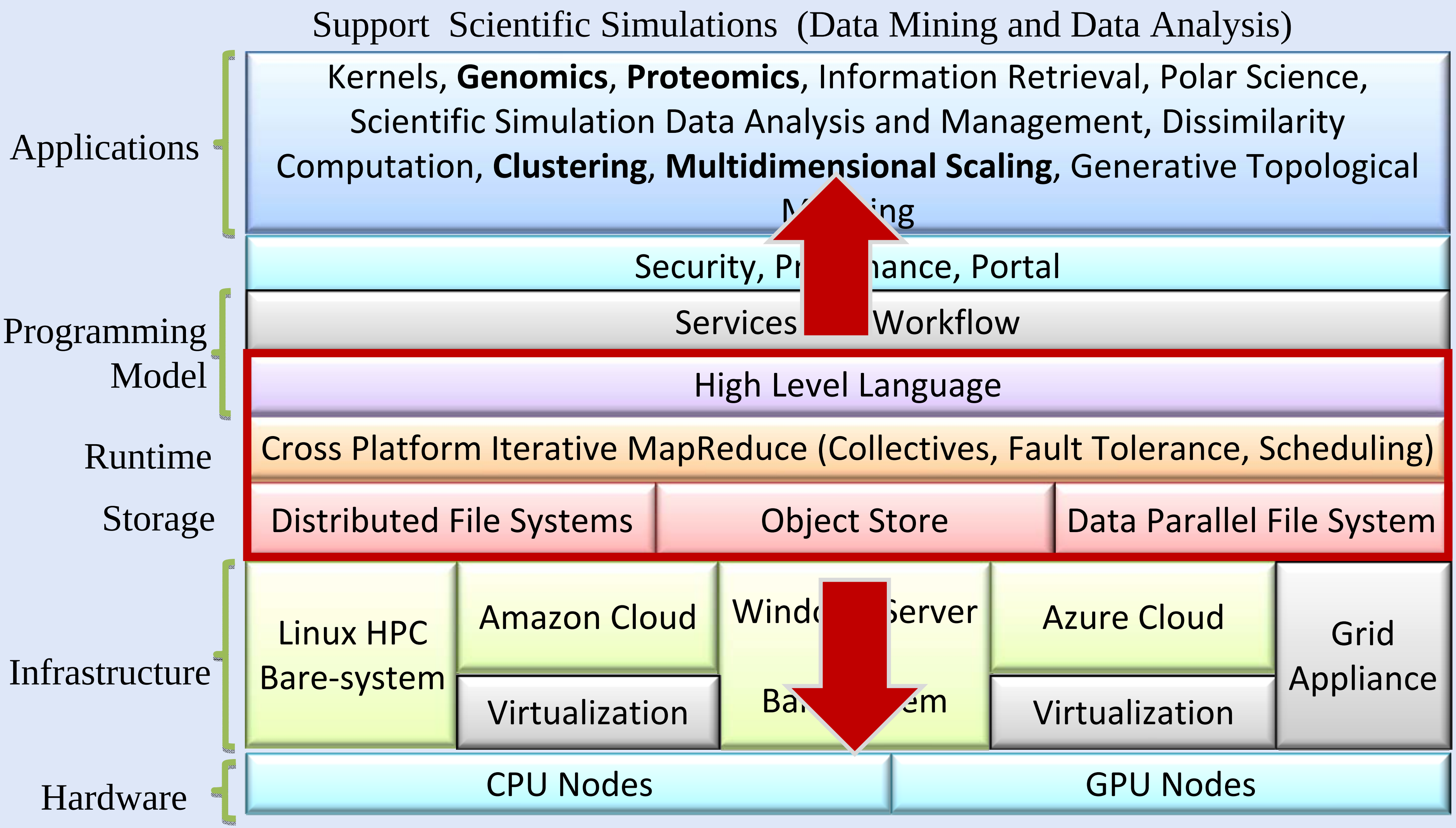
Domain of MapReduce and Iterative Extensions

What is Iterative MapReduce?

Iterative MapReduce

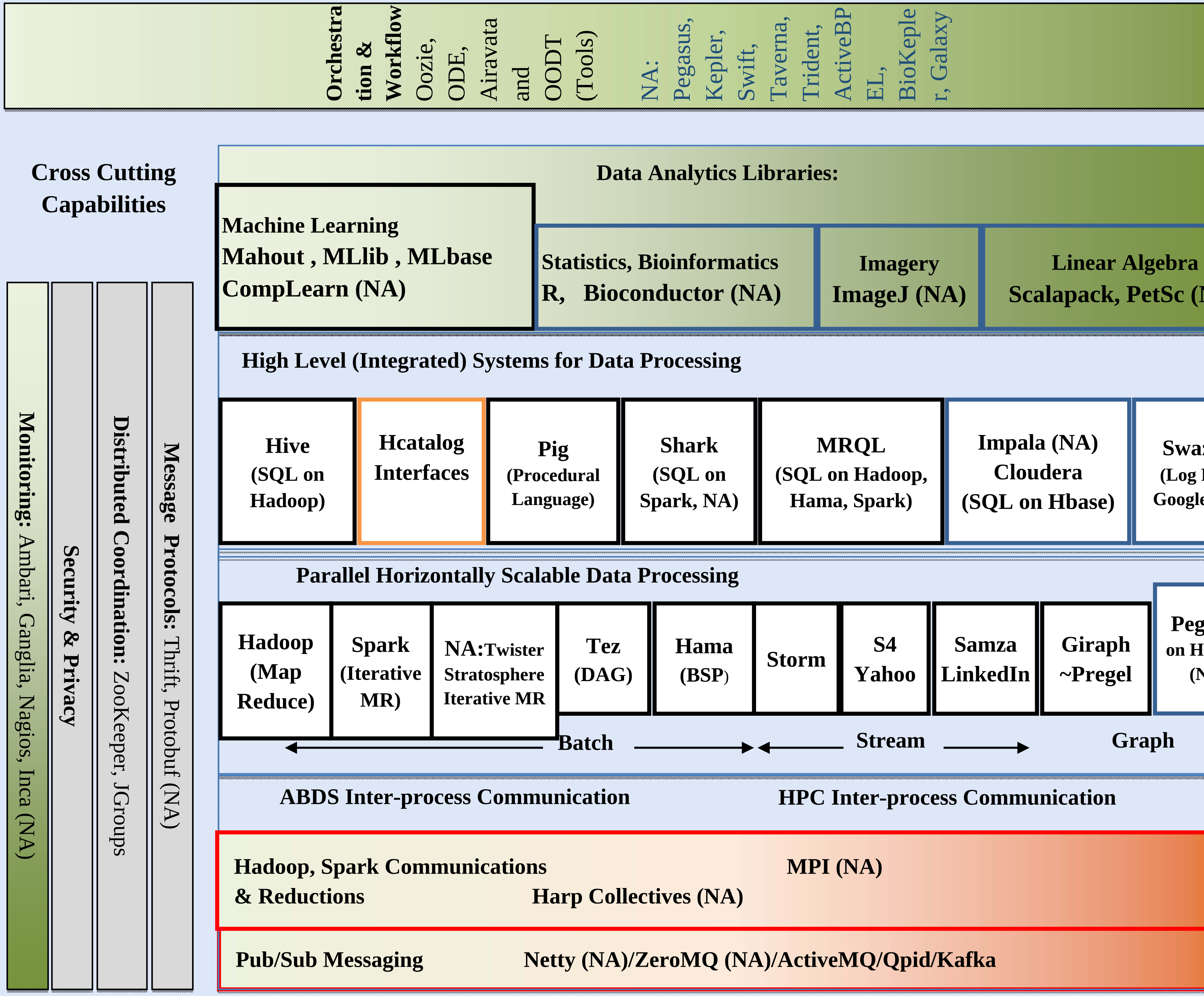
- Mapreduce is a Programming Model instantiating the paradigm of bringing computation to data
- Iterative Mapreduce extends Mapreduce programming model and support iterative algorithms for Data Mining and Data Analysis
- Is it possible to use the same computational tools on HPC and Cloud?
- Enabling scientists to focus on science not programming distributed systems

(Iterative) MapReduce in Context



red
itecture (Upper)

- A – Non Apache projects
- een layers are
pache/Commercial Cloud
ght) to HPC (darker)
- tegration layers



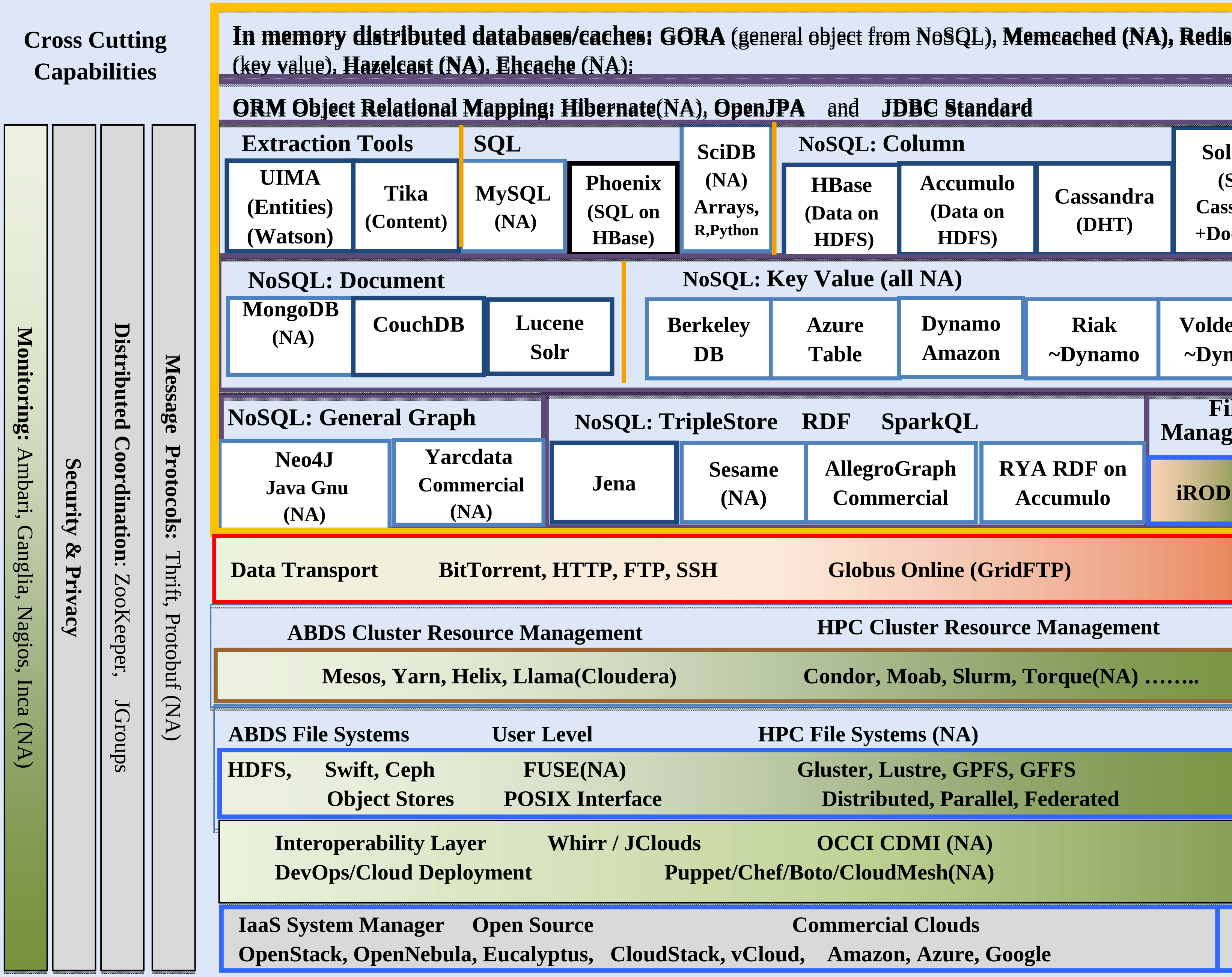
Note: Harp and New Communication Layer

SA

yered
chitecture
wer)

A – Non Apache
rojects

reen layers are
pache/Commercial
loud (light) to HPC
darker) integration
yers



SA

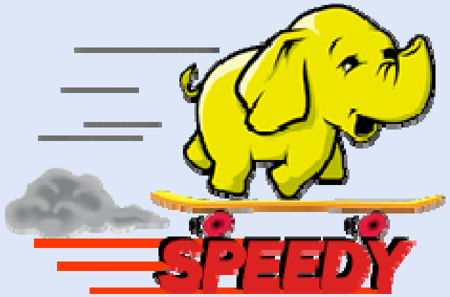
Data Analysis Tools

MapReduce optimized for iterative computations

hadoop

Twister

Cross Platform Iterative MapReduce



Twister: the speedy elephant

Abstractions

In-Memory
- Cacheable map/reduce tasks

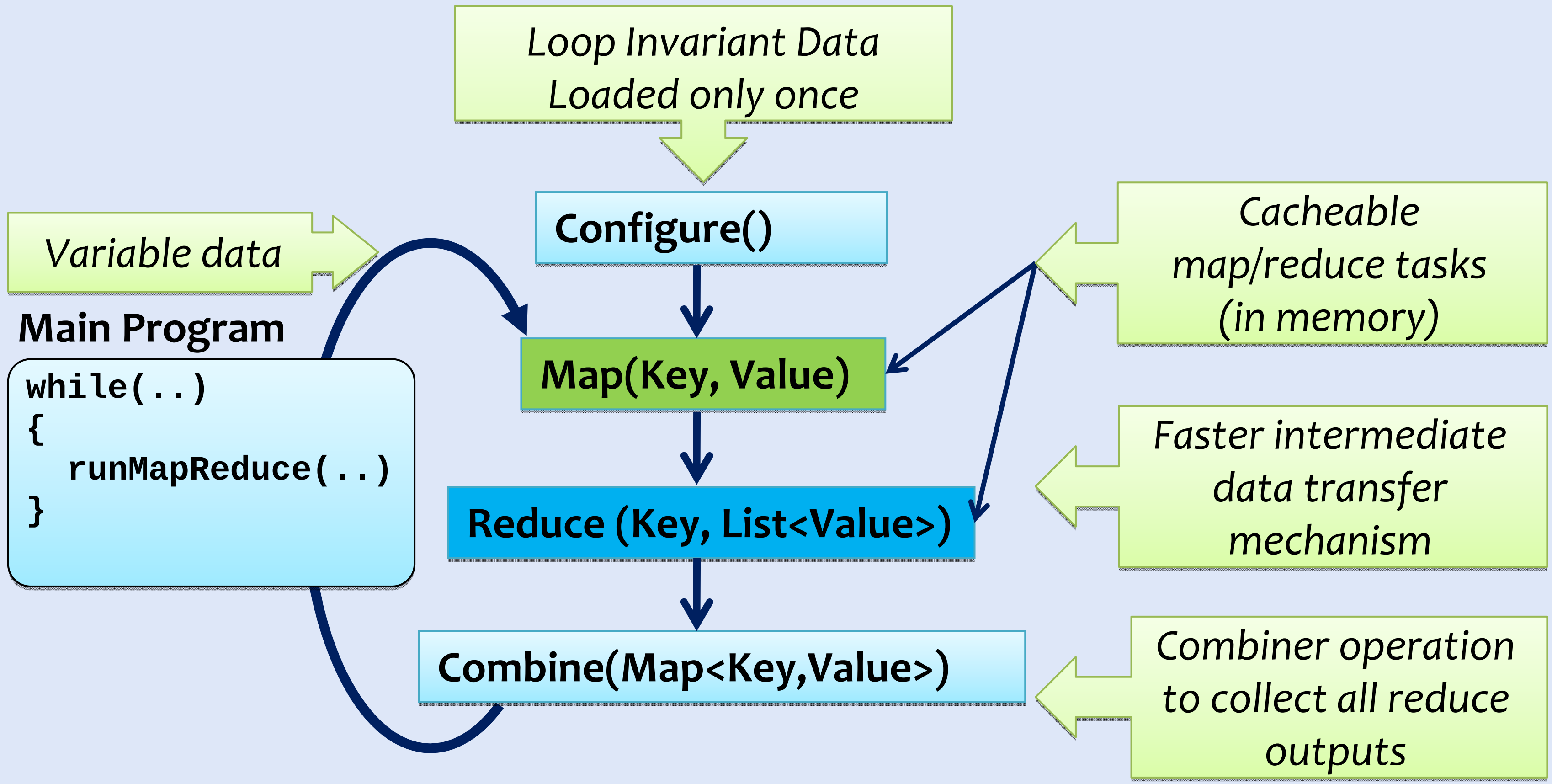
Data Flow
- Iterative
- Loop Invariant
- Variable data

Thread
- Lightweight
- Local aggregation

Map-Collective
- Communication patterns optimized for large intermediate data transfer

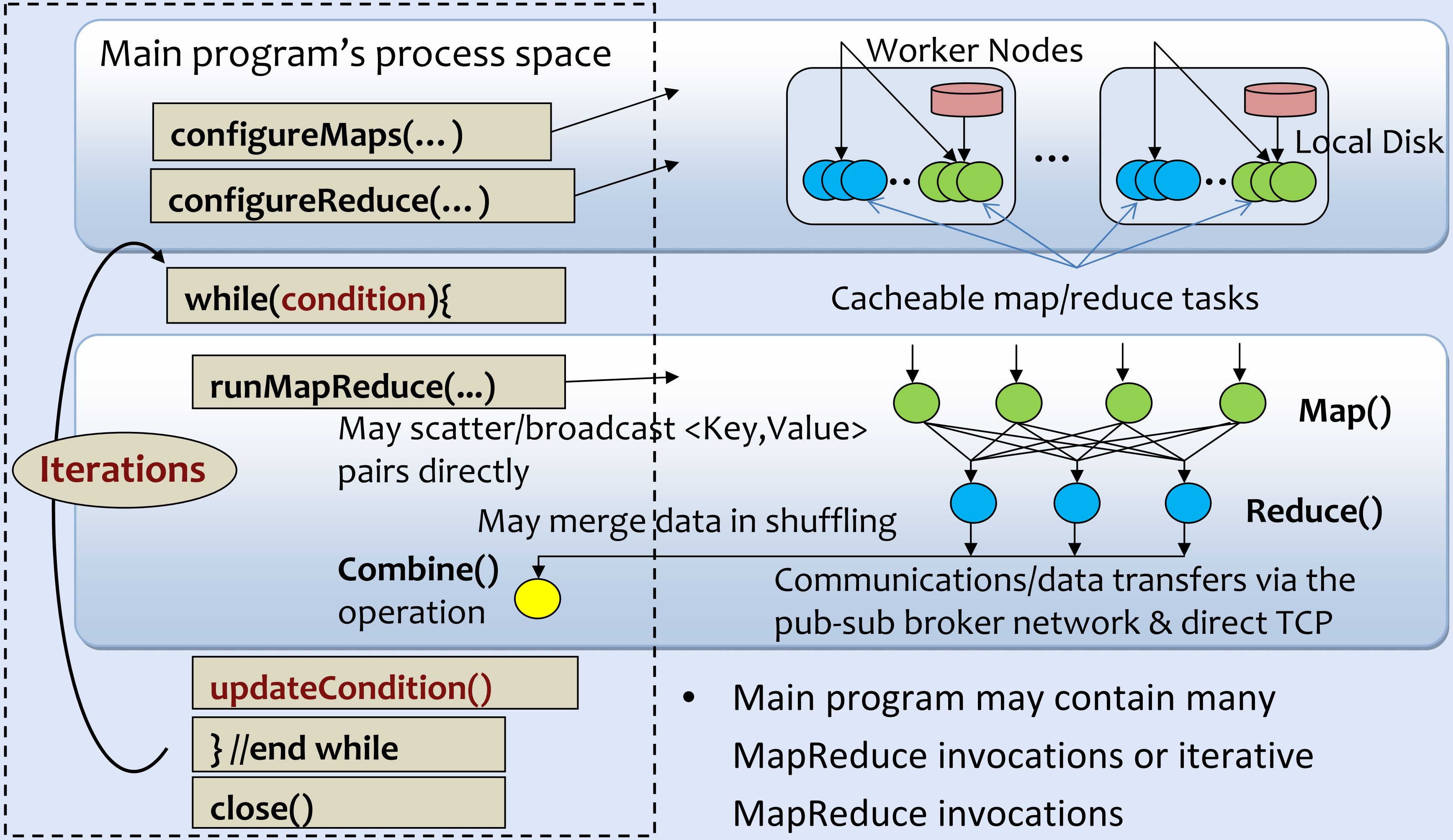
Portability
- HPC (Java)
- Azure Cloud (C#)
- Supercomputer (C++, Java)

Programming Model for Iterative MapReduce



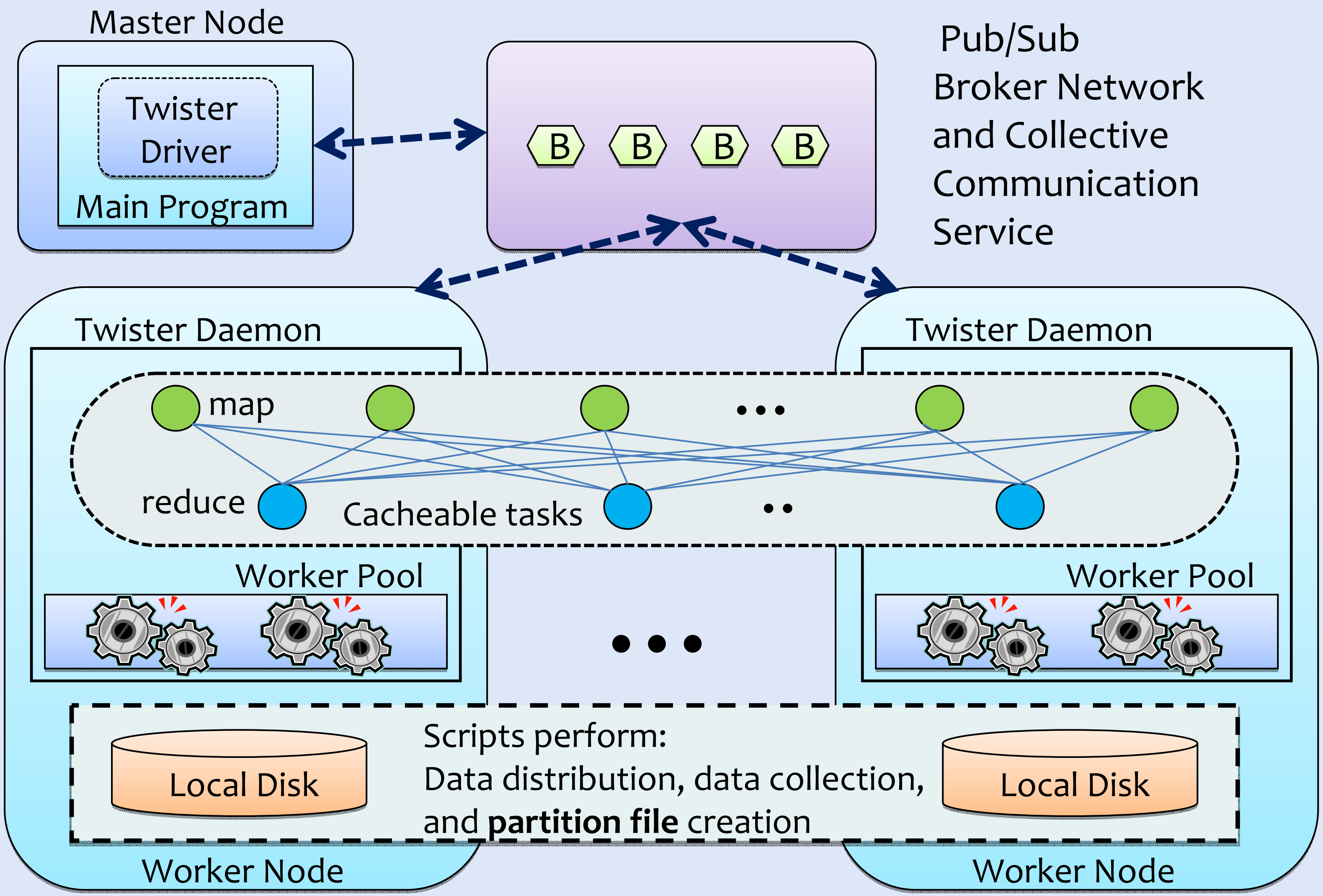
- Distinction on loop invariant data and variable data (**data flow vs. δ flow**)
- Cacheable map/reduce tasks (**in-memory**)
- Combine operation

Twister Programming Model

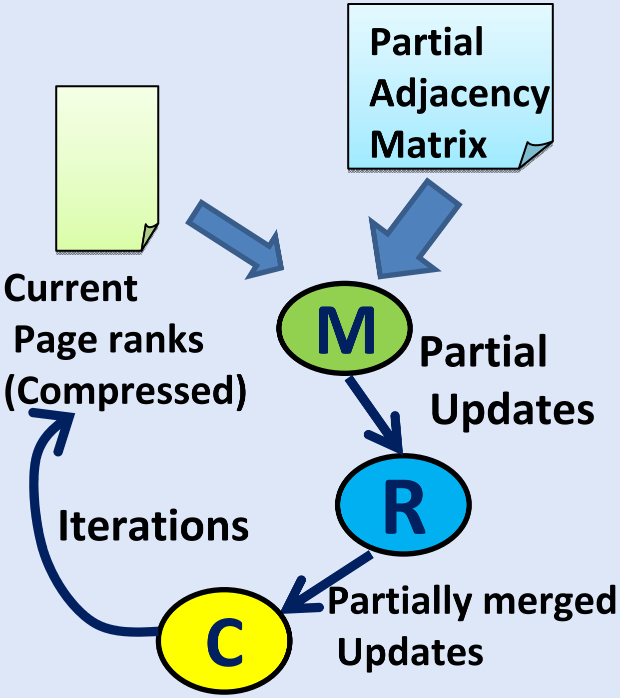


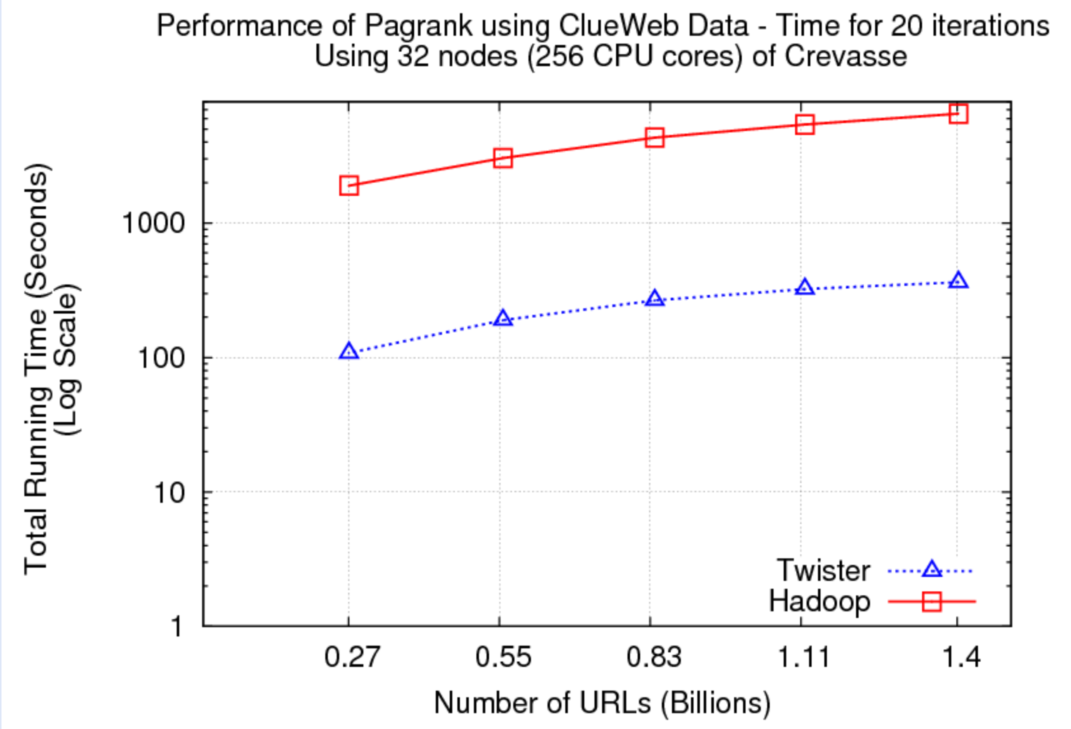
- Main program may contain many MapReduce invocations or iterative MapReduce invocations

Twister Architecture



PageRank



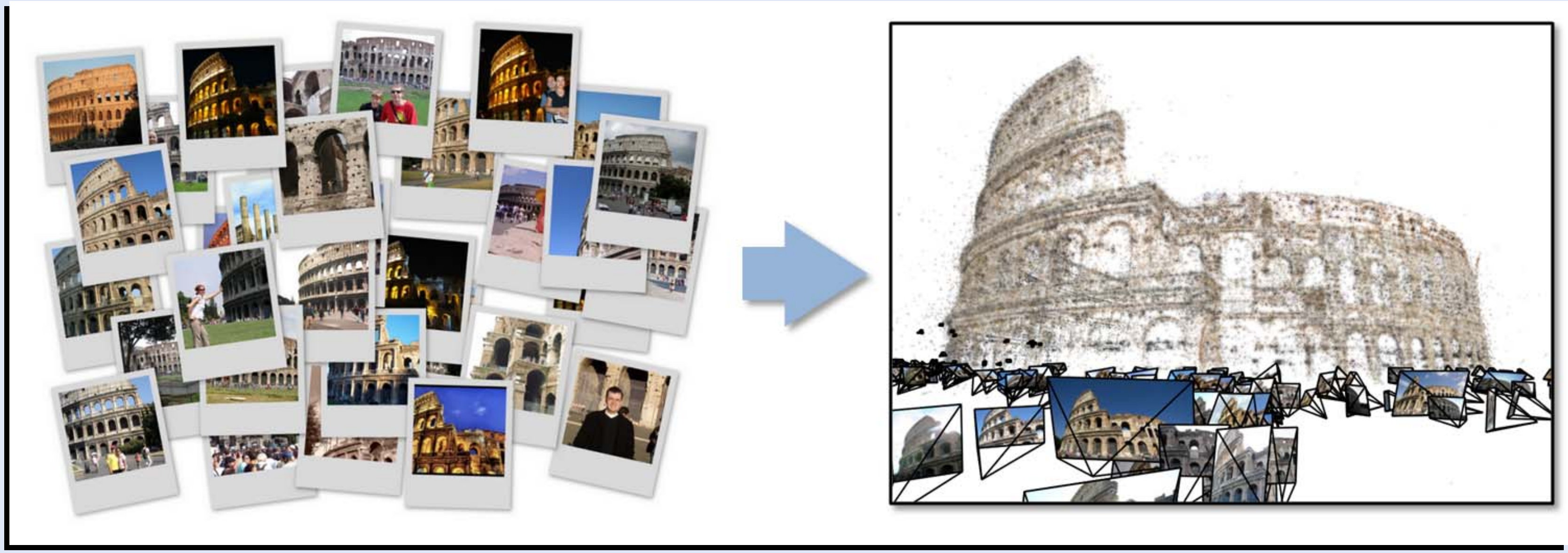


- Well-known page rank algorithm [1]
- Used ClueWeb09 [2] (**1TB in size**) from CMU
- Hadoop loads the web graph in every iteration
- Twister keeps the graph in memory
- Pregel approach seems natural to graph-based problems

[1] Pagerank Algorithm, http://en.wikipedia.org/wiki/PageRank
[2] ClueWeb09 Data Set, http://boston.lti.cs.cmu.edu/Data/clueweb09/

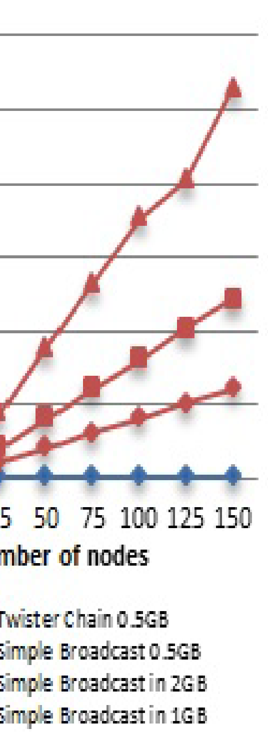
Data Intensive Kmeans Clustering

– *Image Classification:* ***7 million images;*** 512 features per image; 1 million clusters
10K Map tasks; 64G broadcasting data (1GB data transfer per Map task node);
20 TB intermediate data in shuffling.



SA

Broadcast Comparison: Twister vs. MPI vs. Spark



Twister Chain vs.
Broadcasting

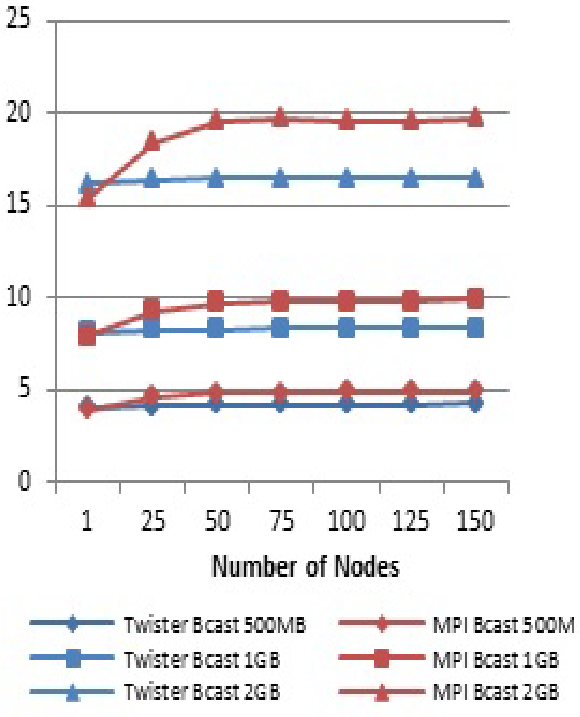


Figure 6. Twister vs. MPI
(Broadcasting 0.5~2GB data)

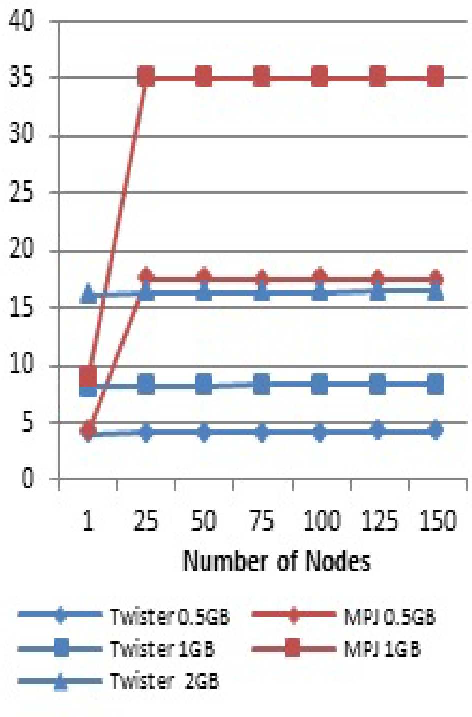


Figure 7. Twister vs. MPJ
(Broadcasting 0.5~2GB data)

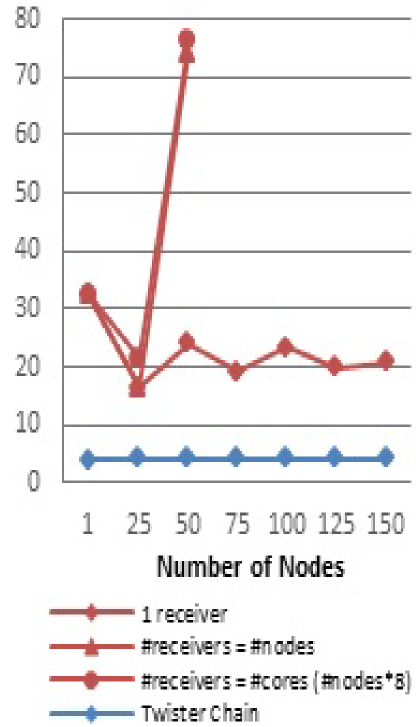


Figure 8. Twister vs. Spark
(Broadcasting 0.5GB data)

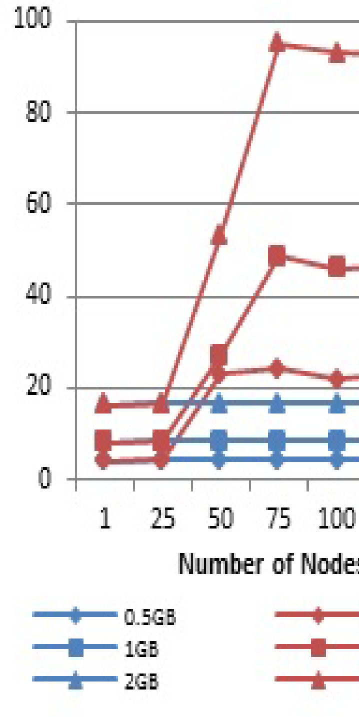


Figure 9. Twister Ch
with/without topology-a

Tested on IU Polar Grid with 1 Gbps Ethernet connection

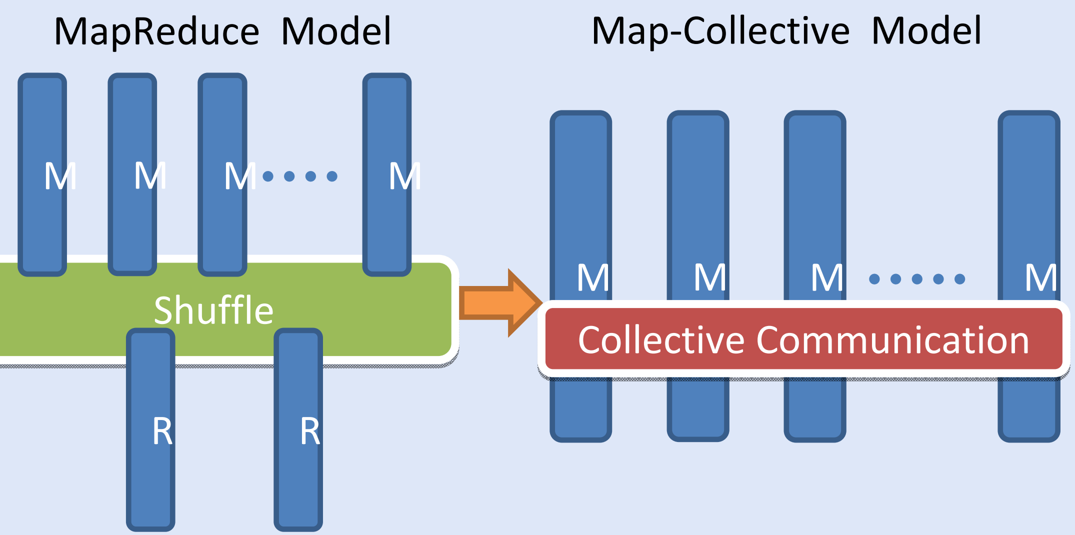
- t least a factor of 120 on 125 nodes, compared with the simple broadcast algorithm
- he new topology-aware chain broadcasting algorithm gives 20% better performance than best C/C++ MPI methods (four time aster than Java MPJ)
- factor of 5 improvement over non-optimized (for topology) pipeline-based method over 150 nodes.

Collective Model

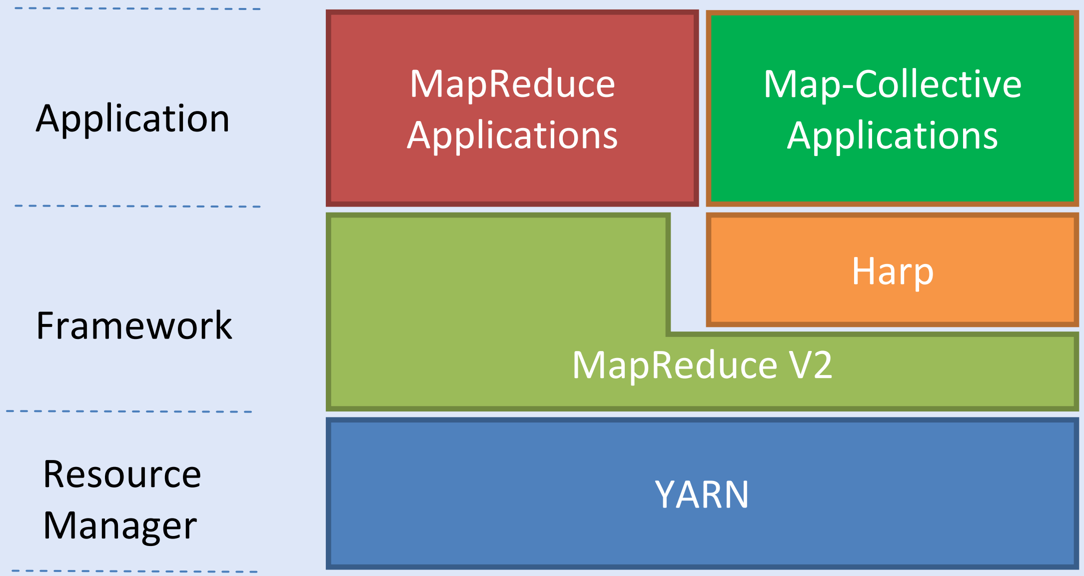
- Harp (2013)
  - https://github.com/jessezbj/harp-project
  - Hadoop Plugin (on Hadoop 1.2.1 and Hadoop 2.2.0)
  - Hierarchical data abstraction on arrays, key-values and graphs for easy programming expressiveness.
  - Collective communication model to support various communication operations on the data abstractions.
  - Caching with buffer management for memory allocation required from computation and communication
  - BSP style parallelism
  - Fault tolerance with check-pointing

Harp Design

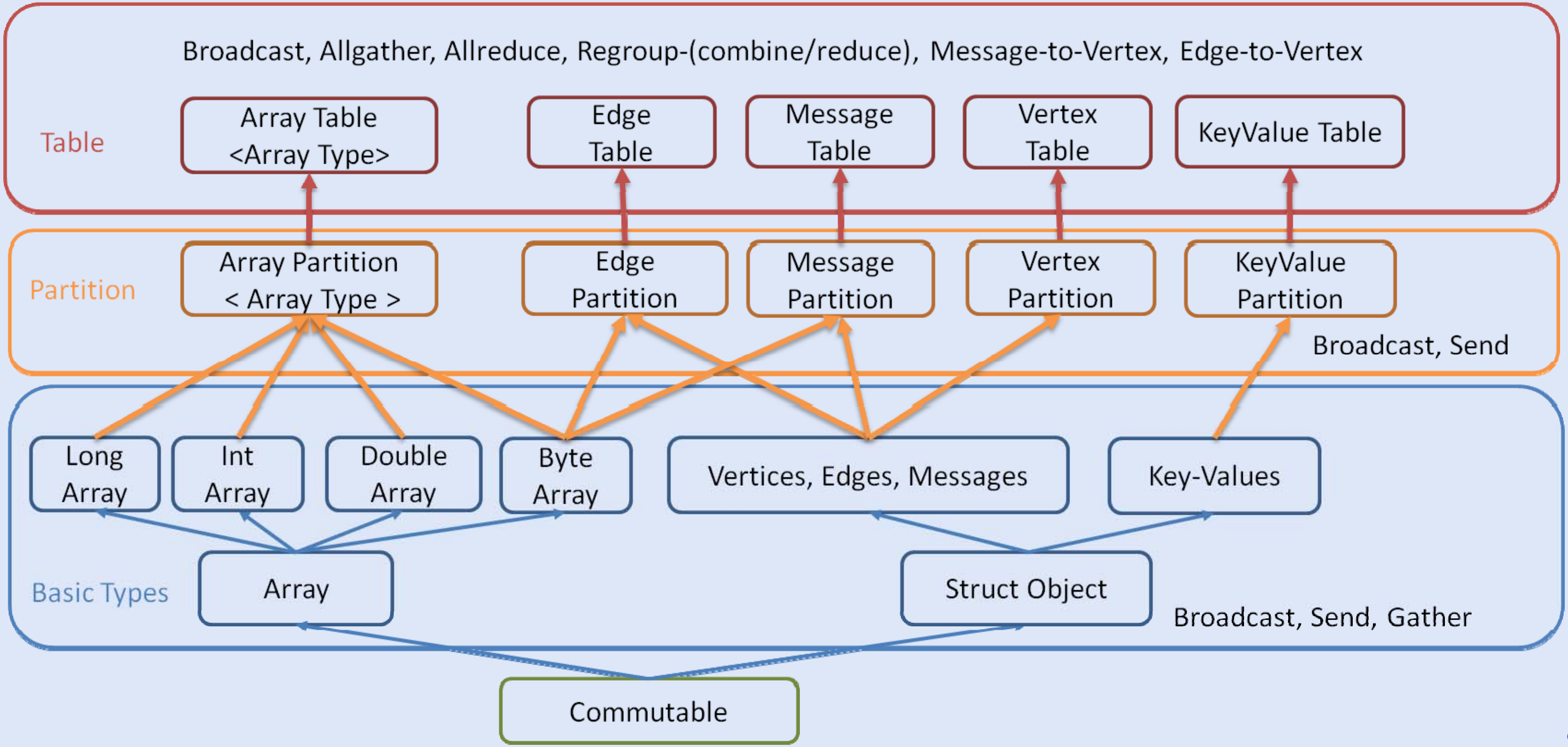
Parallelism Model



- Architecture



Hierarchical Data Abstraction and Collective Communication



Harp Bcast Code Example

```
tected void mapCollective(KeyValReader reader, Context context)
hrows IOException, InterruptedException {

rrTable<DoubleArray, DoubleArrPlus> table =
 new ArrTable<DoubleArray, DoubleArrPlus>(0, DoubleArray.class, DoubleArrPlus.class);

f (this.isMaster()) {
 String cFile = conf.get(KMeansConstants.CFILE);
 Map<Integer, DoubleArray> cenDataMap = createCenDataMap(cParSize, rest, numCenPartition
   vectorSize, this.getResourcePool());
 loadCentroids(cenDataMap, vectorSize, cFile, conf);
 addPartitionMapToTable(cenDataMap, table);


rrTableBcast(table);
```

SA

K-means Clustering Parallel Efficiency

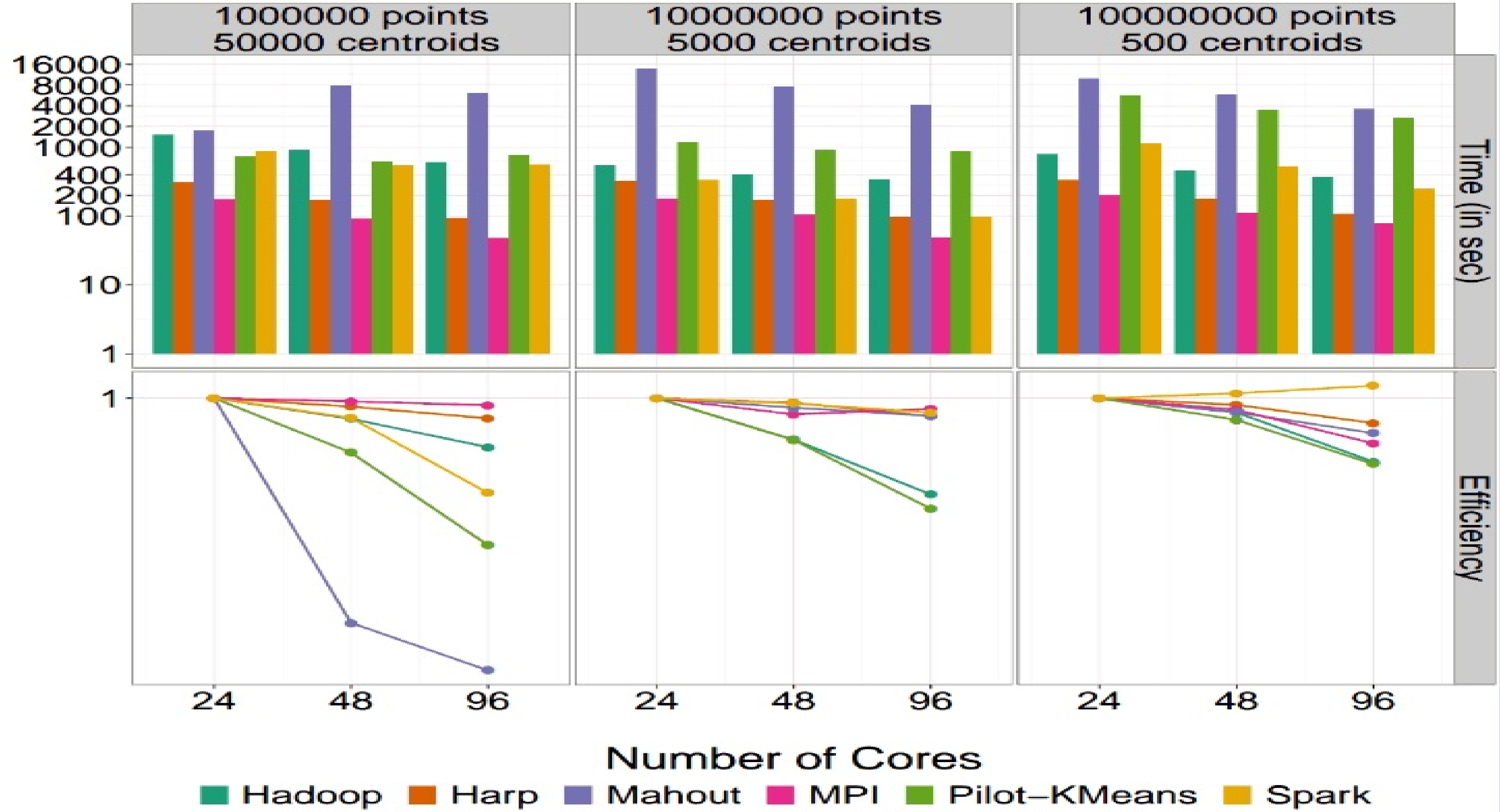


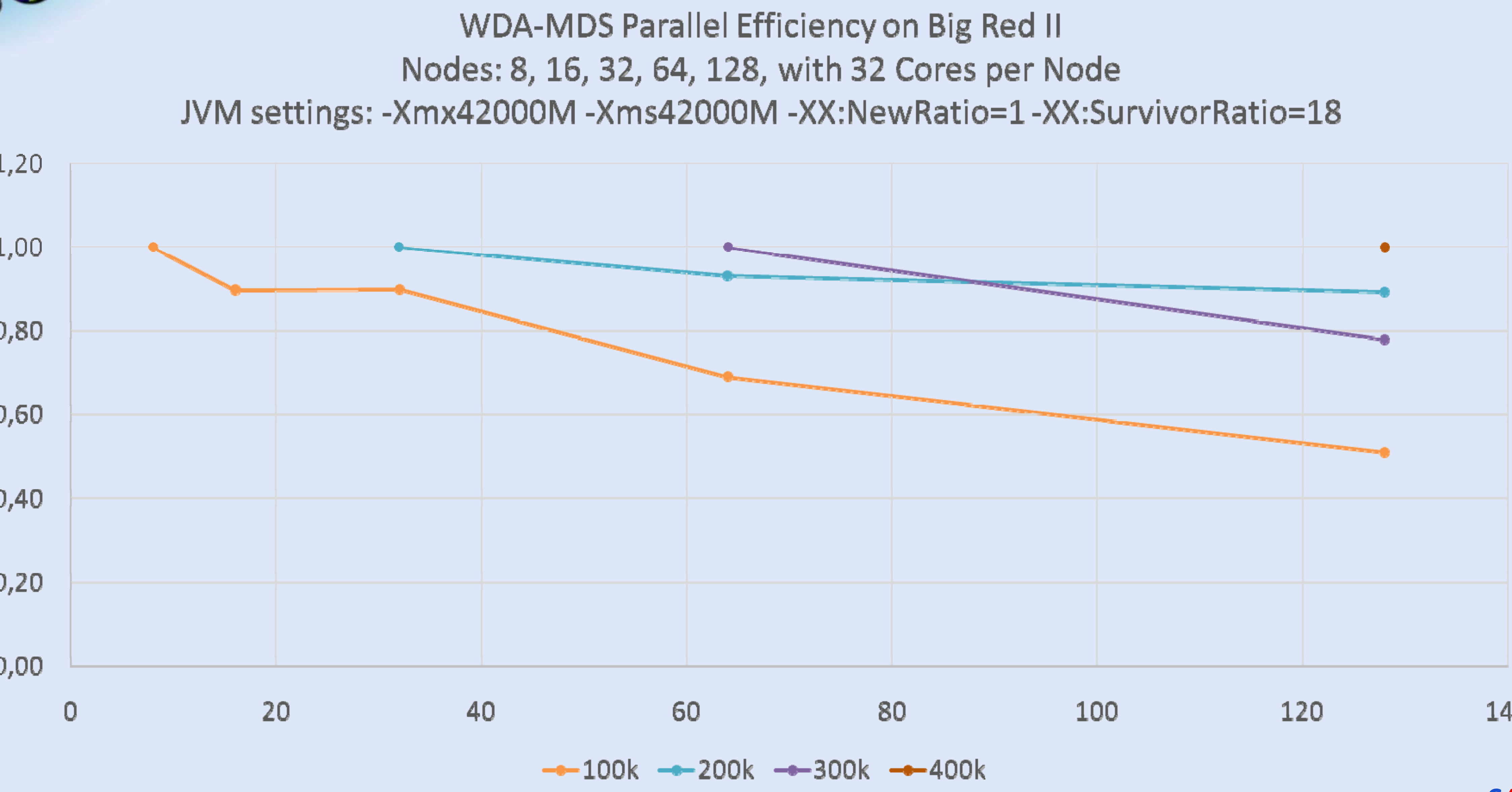
Fig. 5. Time-To-Completion for KMeans on Different Backends

- Shantenu Jha et al. A Tale of Two Data-Intensive Paradigms: Applications, Abstractions, and Architectures. 2014.

WDA-MDS Performance on Big Red II

- WDA-MDS
  - Y. Ruan et. al, A Robust and Scalable Solution for Interpolative Multidimensional Scaling with Weighting. IEEE e-Science 2013.
- Big Red II
  - http://kb.iu.edu/data/bcqt.html
- Allgather
  - Bucket algorithm
- Allreduce
  - Bidirectional exchange algorithm

Parallel Efficiency

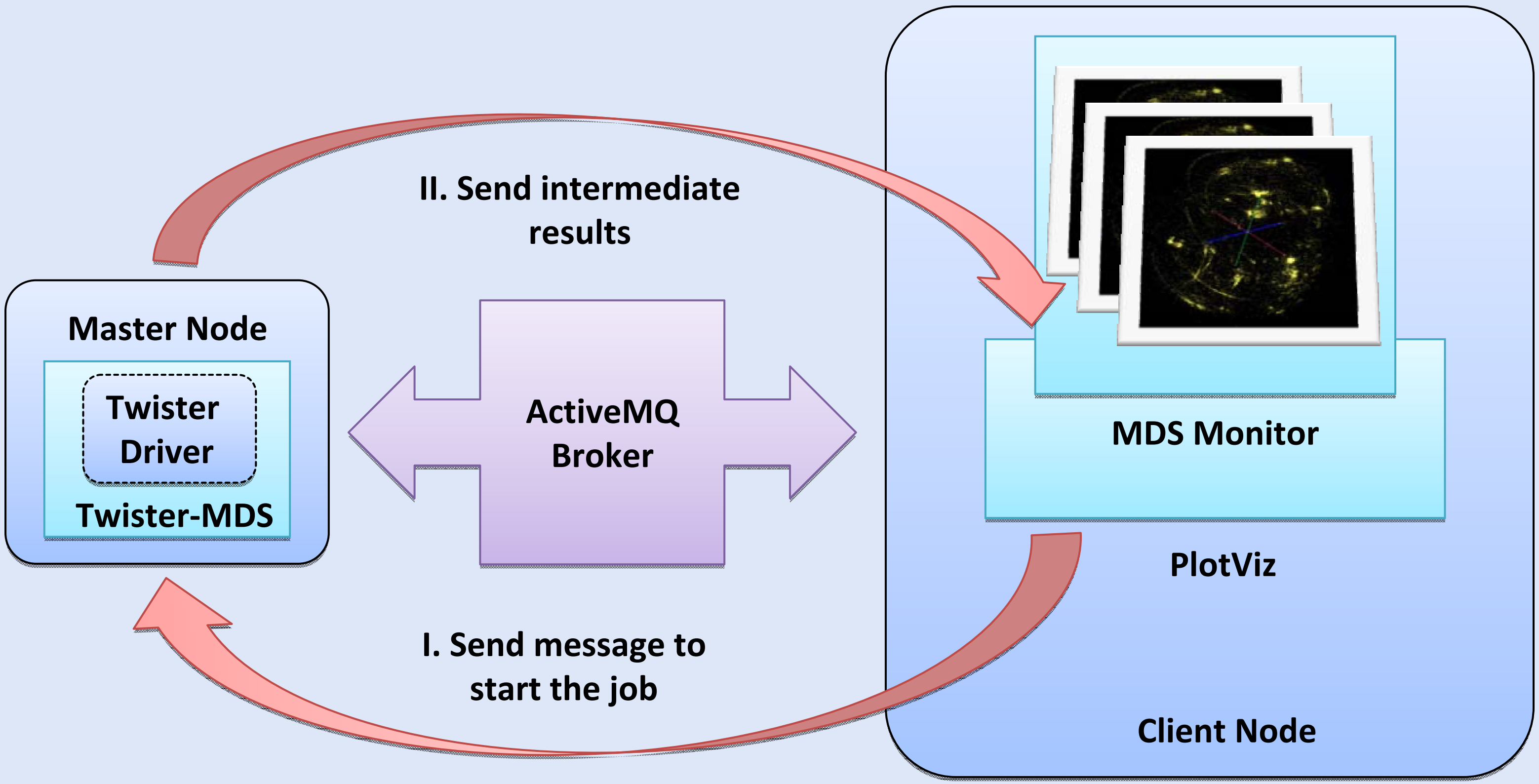


REEF

- Retainable Evaluator Execution Framework
- http://www.reef-project.org/
- Provides system authors with a centralized (pluggable) control flow
  - Embeds a user-defined system controller called the Job Driver
  - Event driven control
- Package a variety of data-processing libraries (e.g., high-bandwidth shuffle, relational operators, low-latency group communication, etc.) in a reusable form.
- To cover different models such as MapReduce, query, graph processing and stream data processing

SA

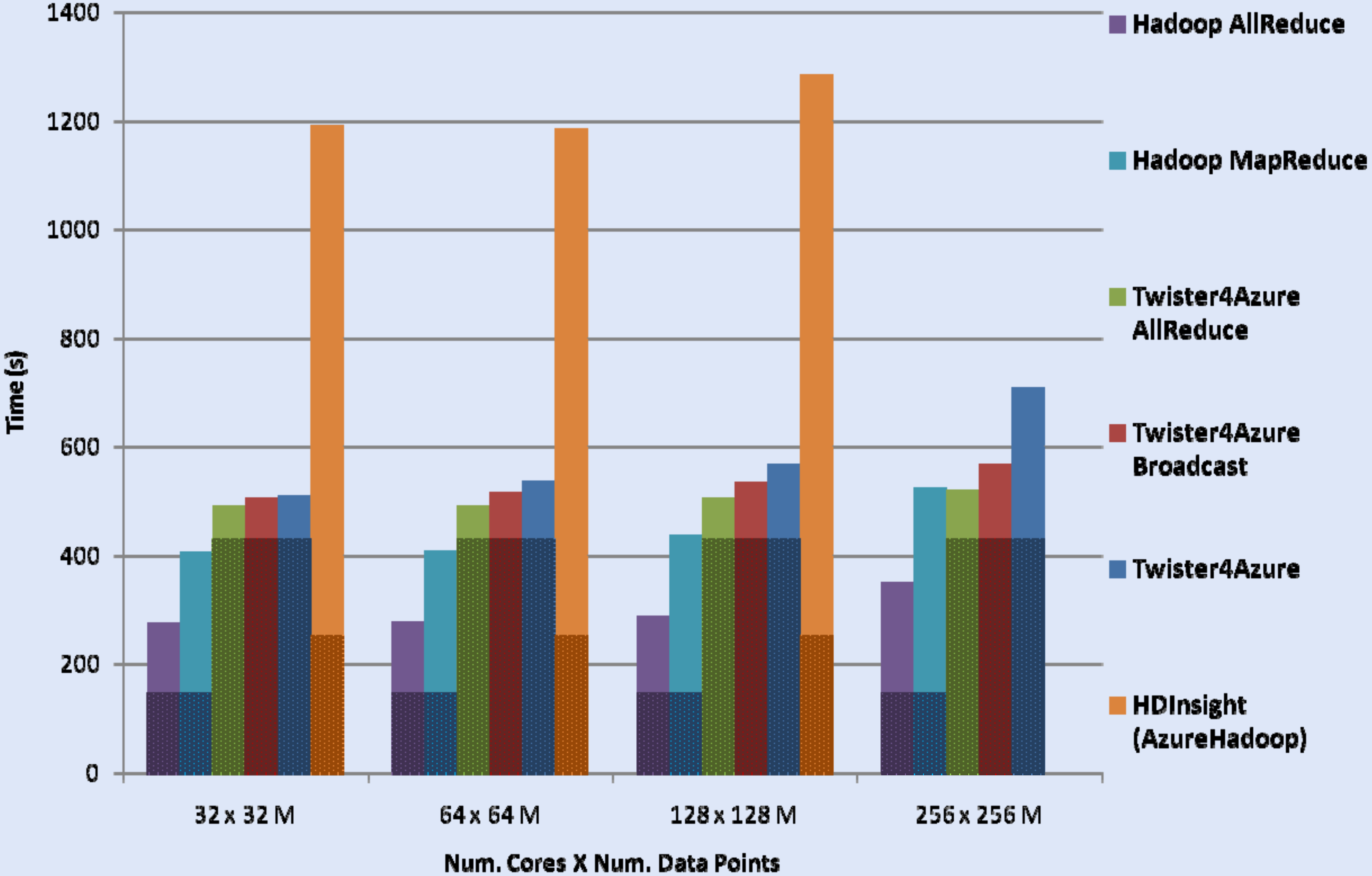
Iterative Mapreduce - MDS Demo



- Input: 30k metagenomics data
- MDS reads pairwise distance matrix of all sequences
- Output: 3D coordinates visualized in PlotViz

Iterative Mapreduce -MDS Demo

KMeans Clustering Comparison
Hadoop vs. HDInsight vs. Twister4Azure



Shaded part is computation

Iterative MapReduce Models

- Twister (2010)
  - Jaliya Ekanayake et al. Twister: A Runtime for Iterative MapReduce. HPDC workshop 2010.
  - http://www.iterativemapreduce.org/
  - Simple collectives: broadcasting and aggregation.

- HaLoop (2010)
  - Yingyi Bu et al. HaLoop: Efficient Iterative Data Processing on Large clusters. VLDB 2010.
  - http://code.google.com/p/haloop/
  - Programming model $R_{i+1} = R_0 \cup (R_i \bowtie L)$
  - Loop-Aware Task Scheduling
  - Caching and indexing for Loop-Invariant Data on local disk

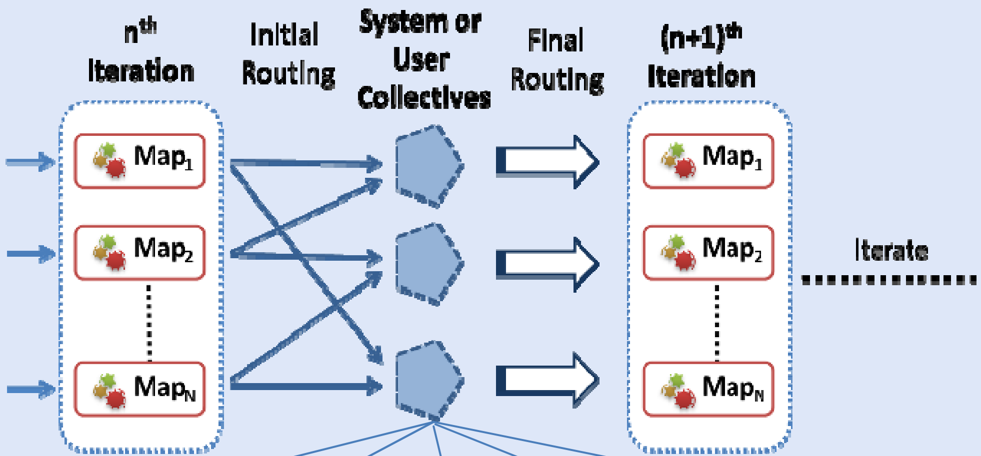
Model Composition

- Apache Spark (2010)
  - Matei Zaharia et al. Spark: Cluster Computing with Working Sets,. HotCloud 2010.
  - Matei Zaharia et al. Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing. NSDI 2012.
  - http://spark.apache.org/
  - Resilient Distributed Dataset (RDD)
  - RDD operations
    - MapReduce-like parallel operations
  - DAG of execution stages and pipelined transformations
  - Simple collectives: broadcasting and aggregation

Graph Processing with BSP model

- Pregel (2010)
  - Grzegorz Malewicz et al. Pregel: A System for Large-Scale Graph Processing. SIGMOD 2010.
- Apache Hama (2010)
  - https://hama.apache.org/
- Apache Giraph (2012)
  - https://giraph.apache.org/
  - Scaling Apache Giraph to a trillion edges
    - https://www.facebook.com/notes/facebook-engineering/scaling-apache-giraph-to-a-trillion-edges/10151617006153920

Map-Collective Communication Model



Patterns

MapReduce
- Wordcount, Grep

MapReduce-MergeBroadcast
- KMeansClustering, PageRank

Map-AllGather
- MDS-BCCalc

Map-AllReduce
- KMeansClustering, MDS-StressCalc

Map-ReduceScatter
- PageRank, Belief Propagation

- We generalize the Map-Reduce concept to Map-Collective, noting that large collectives are a distinguishing feature of data intensive and data mining applications.
- Collectives generalize Reduce to include all large scale linked communication-compute patterns.
- MapReduce already includes a step in the collective direction with sort, shuffle, merge as well as basic reduction.

Future Work

- Run Algorithms on a much larger scale
- Provide Data Service on Clustering and MDS Algorithms

Acknowledgement

Bingjing Zhang Thilina Gunarathne Fei Teng Xiaoming Gao Stephen Wu Yuan Young

Prof. Haixu Tang
Bioinformatics

Prof. David Crandall
Computer Vision

Prof. Filippo Menczer
Complex Networks and Systems

SALSA HPC Group

http://salsahpc.indiana.edu

School of Informatics and Computing

Indiana University

Thank You!

Questions?