Performance Comparison of Intel® Enterprise Edition for Lustre* software and HDFS for MapReduce Applications

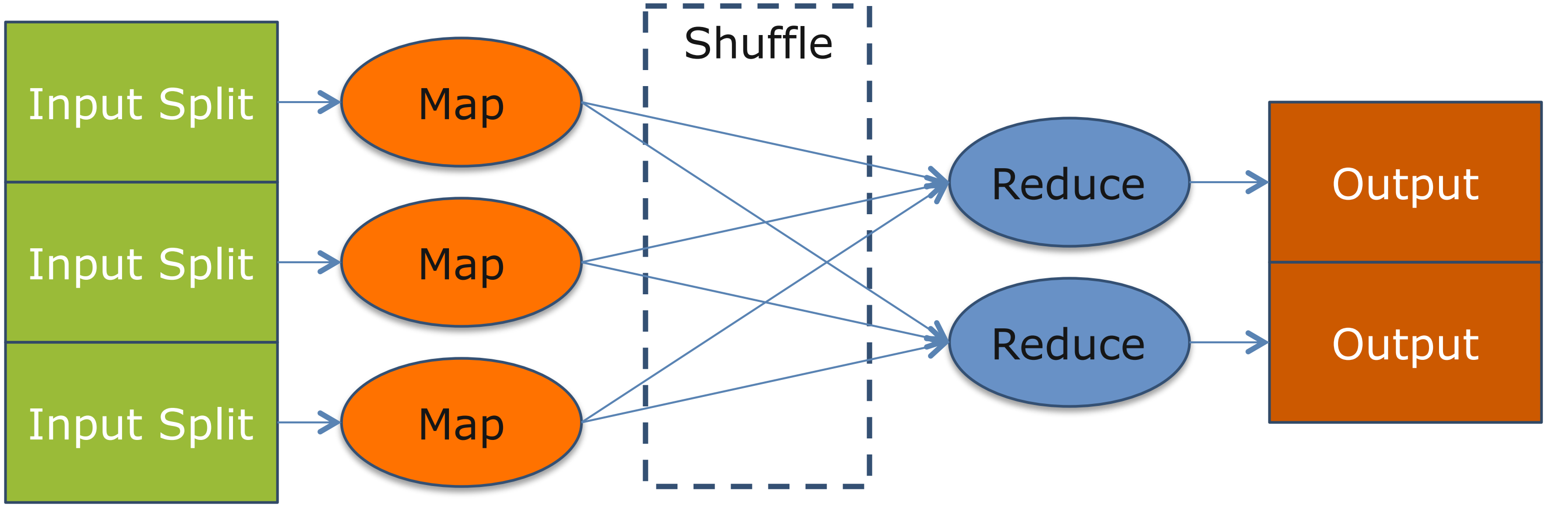
Rekha Singhal, Gabriele Pacciucci and Mukesh Gangadhar

* Other names and brands may be claimed as the property of others.

Hadoop Introduction

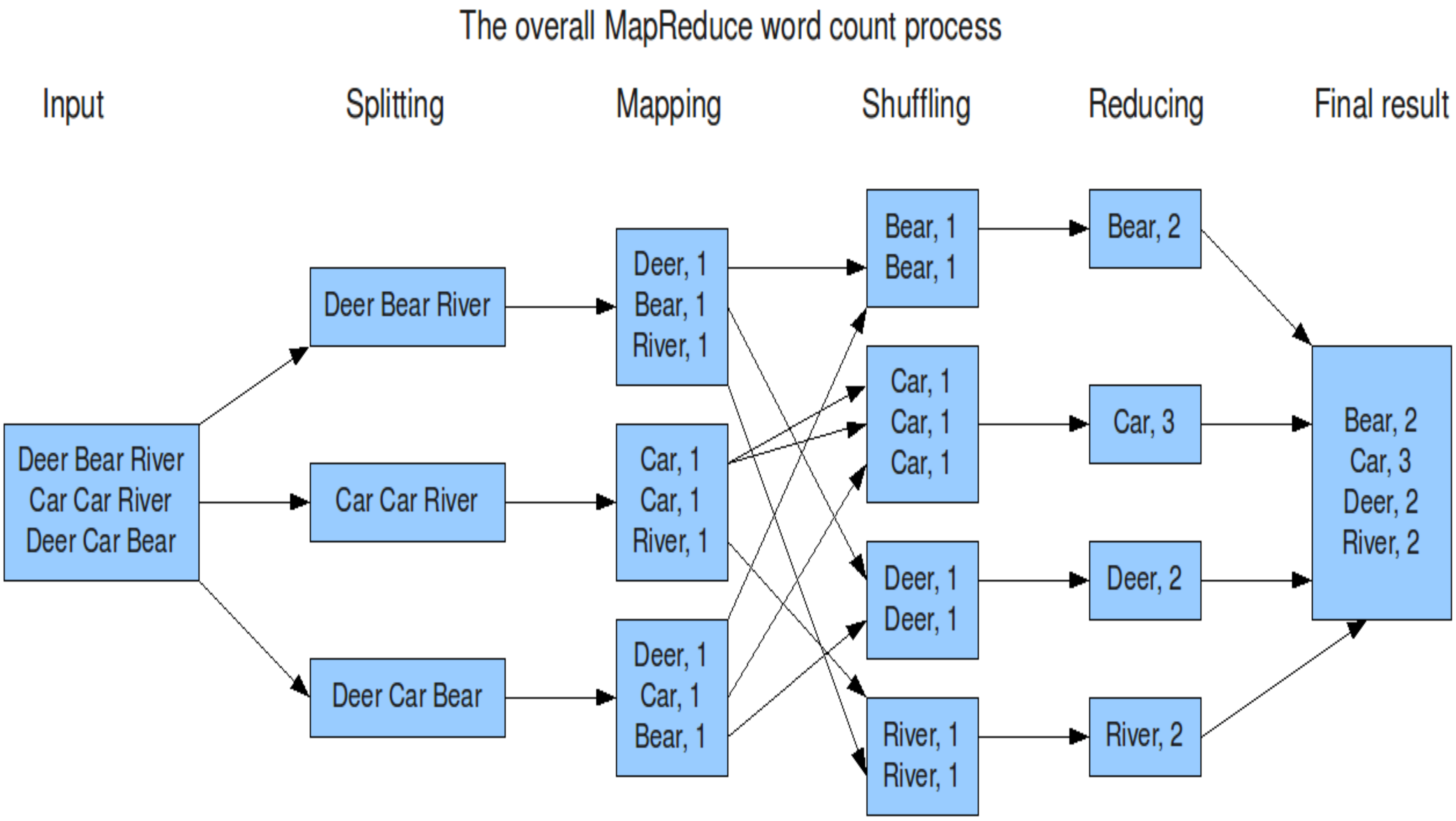
- Open source MapReduce framework for data-intensive computing
- Simple programming model – two functions: Map and Reduce
- Map: Transforms input into a list of key value pairs
  - Map(D) → List[Ki , Vi]
- Reduce: Given a key and all associated values, produces result in the form of a list of values
  - Reduce(Ki , List[Vi]) → List[Vo]
- Parallelism hidden by framework
  - Highly scalable: can be applied to large datasets (Big Data) and run on commodity clusters
- Comes with its own user-space distributed file system (HDFS) based on the local storage of cluster nodes

Hadoop Introduction (cont.)

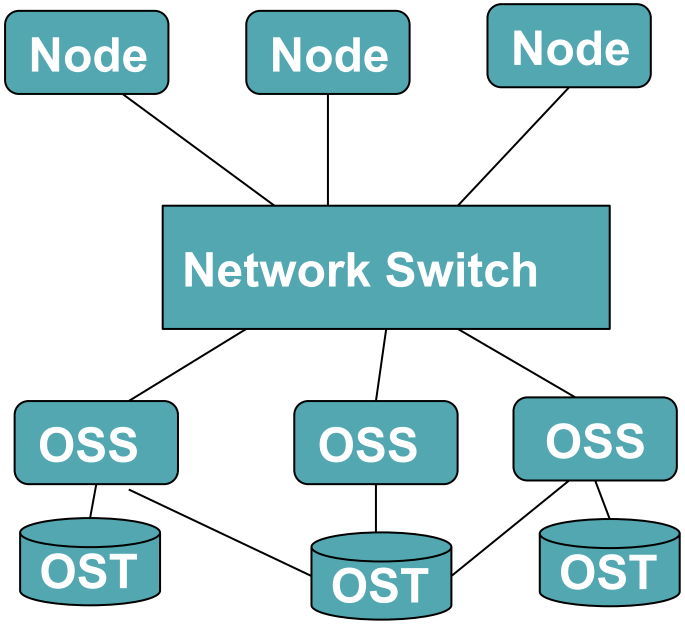


- Framework handles most of the execution
- Splits input logically and feeds mappers
- Partitions and sorts map outputs (Collect)
- Transports map outputs to reducers (Shuffle)
- Merges output obtained from each mapper (Merge)

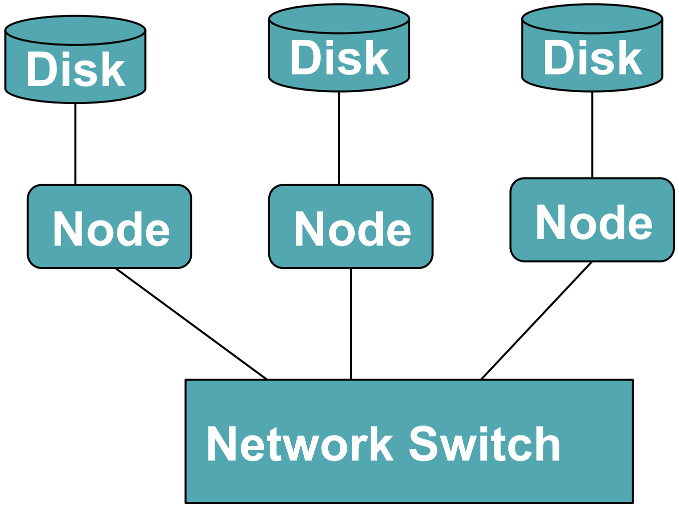
MapReduce Application Processing



Intel® Enterprise Edition for Lustre* software



Hadoop Dist. File System



* Other names and brands may be claimed as the property of others.

Intel® Enterprise Edition for Lustre* software

- Clustered, distributed computing and storage
- No data replication
- No local storage
- Widely used for HPC applications

Hadoop Dist. File System

- Data moves to the computation
- Data replication
- Local storage
- Widely used for MR applications

* Other names and brands may be claimed as the property of others.

Motivation

□ Could HPC and MR co-exist?

□ Need to evaluate use of Lustre software for MR application processing

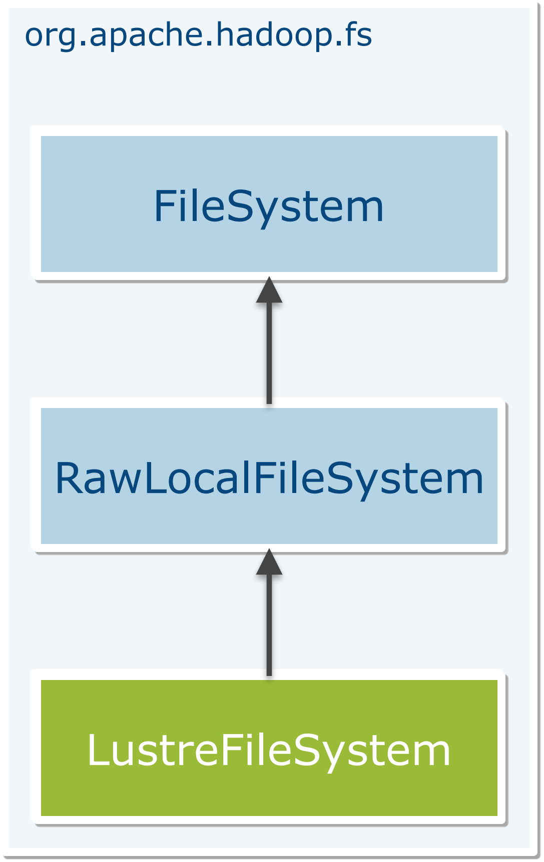
Using Intel® Enterprise Edition for Lustre software with Hadoop*

HADOOP ‘ADAPTER’ FOR LUSTRE

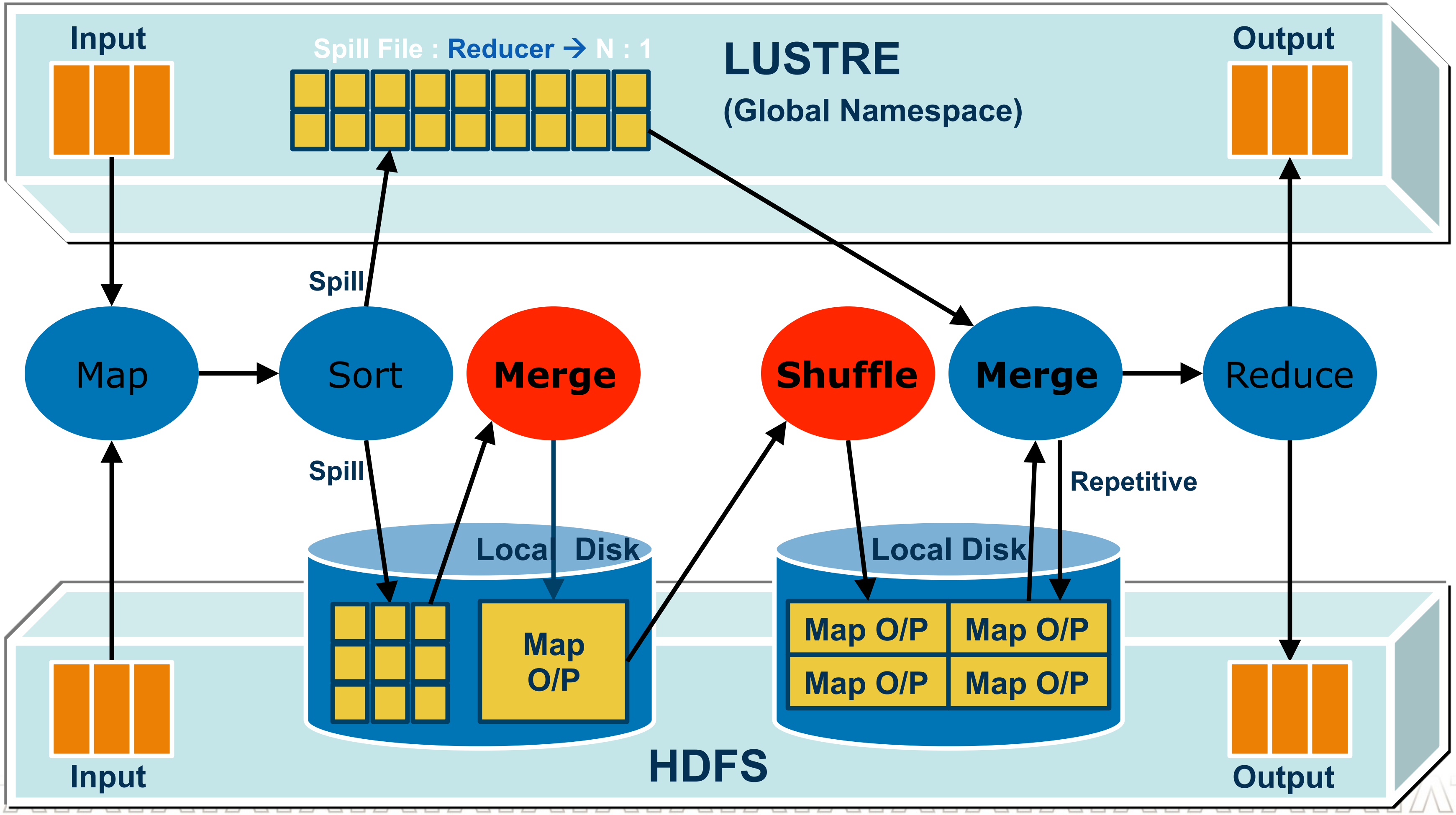
* Other names and brands may be claimed as the property of others.

Hadoop over Intel EE for Lustre* Implementation

- Hadoop uses pluggable extensions to work with different file system types
- Lustre is POSIX compliant:
  - Use Hadoop's built-in LocalFileSystem class
  - Uses native file system support in Java
- Extend and override default behavior: LustreFileSystem
  - Defines new URL scheme for Lustre – lustre:///
  - Controls Lustre striping info
  - Resolves absolute paths to user-defined directory
  - Leaves room for future enhancements
- Allow Hadoop to find it in config files



MR Processing in Intel® EE for Lustre* and HDFS



TATA CONSULTANCY SERVICES

* Other names and brands may be claimed as the property of others.

Conclusions from Existing Evaluations

- ❑ TestDFSIO: 100% better throughput

- ❑ TeraSort: 10-15% better performance

- ❑ High Speed connecting Network Needed

- ❑ Same BOM, HDFS is better for WordCount and BigMapOutput applications

- ❑ Large number of compute nodes may challenge Enterprise Edition for Lustre* for software performance

* Other names and brands may be claimed as the property of others.

Problem Definition

Performance comparison of Lustre and HDFS file systems for MR implementation of FSI workload using HPDD cluster hosted in the Intel BigData Lab in Swindon (UK) using Intel® Enterprise Edition for Lustre* software

Audit Trail System part of FINRA security specifications (publicly available) is used as a representative application.

* Other names and brands may be claimed as the property of others.

EXPERIMENTAL SETUP

Hadoop + HDFS Setup

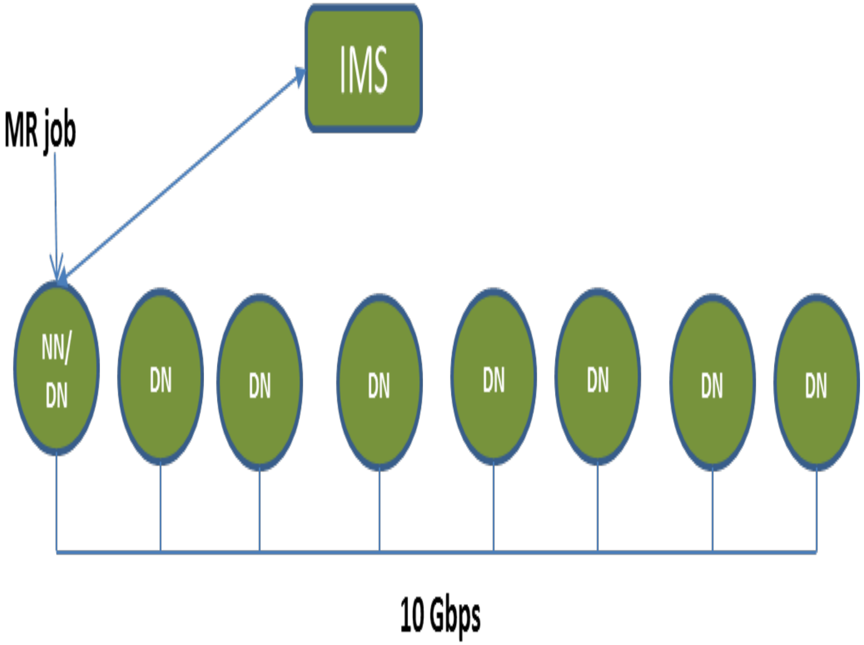
- 1 cluster manager, 1 Name node (NN), 8 Data nodes (DN) including NN.

- 8 nodes, each of Intel(R) Xeon(R) CPU E5-2695 v2 @ 2.40GHz, 320GB cluster RAM

- 27 TB of cluster storage

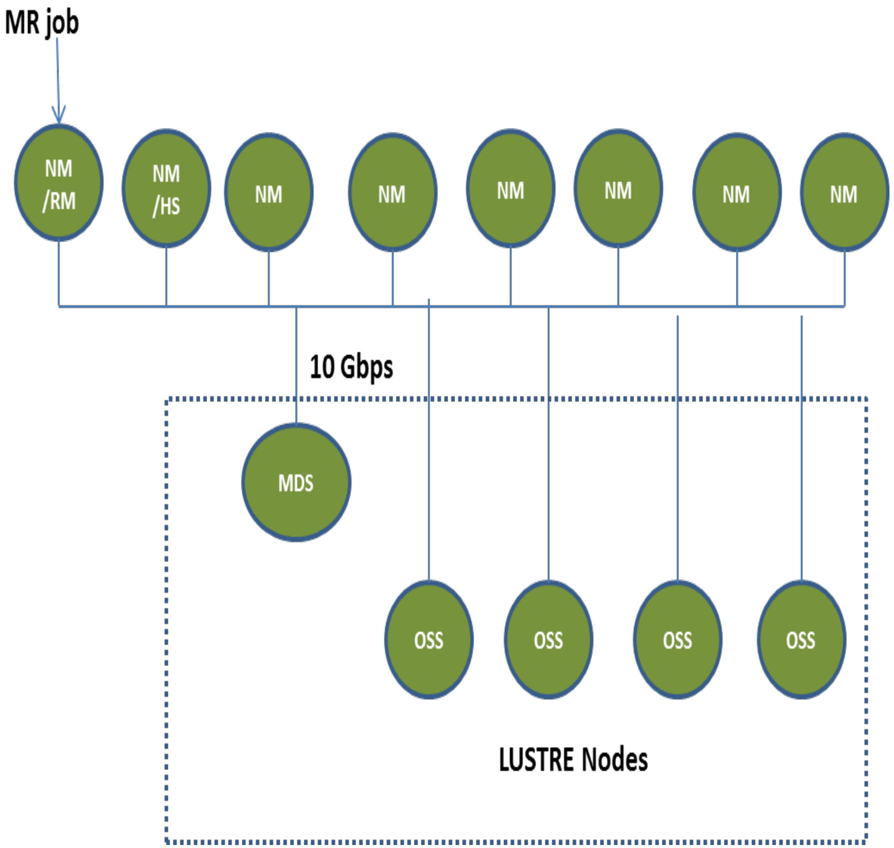
- 10 GB network among compute nodes

- Red Hat 6.5, CDH 5.0.2 and HDFS



Hadoop + Intel EE for Lustre* software - Setup

- 1 Resource manager (RM), 1 History server (HS), 8 Node managers (NM) including RM and HS.
- 8 nodes, each of Intel(R) Xeon(R) CPU E5-2695 v2 @ 2.40GHz, 320GB cluster RAM
- 165TB of usable Lustre storage
- 10 GB network among compute nodes
- Red Hat 6.5, CDH 5.0.2, Intel® Enterprise Edition for Lustre* software 2.0



* Other names and brands may be claimed as the property of others.

Intel® Enterprise Edition for Lustre* 2.0 Setup

- ❑ Four OSS, One MDS, 16 OSTs, 1 MDT.

- ❑ OSS Node
  - ○ CPU- Intel(R) Xeon(R) CPU E5-2637 v2 @ 3.50GHz , Memory - 128GB DDr3 1600mhz
  - ○ Disk subsystem
    - 4 only LSI Logic / Symbios Logic MegaRAID SAS 2108 [Liberator] (rev 05)
    - 4 only 4TB SATA drives per controller raid 5 configuration per raid set
  - ○ 4 OST per OSS node.

Cluster Parameters

- ❑ Number of Compute nodes = 8
- ❑ Map slots = 24
- ❑ Reduce slots = 7
- ❑ Rest of parameters such as Shuffle percent, Merge Percent, Sort Buffer are all kept as default

- ❑ HDFS
  - Replication Factor = 3
- ❑ Intel® EE for Lustre* software
  - stripe count = 1,4,16.
  - stripe size = 4MB

* Other names and brands may be claimed as the property of others.

Job Configuration Parameters

❑ Map Split size= 1GB

❑ Block size = 128MB

❑ Input Data is NOT compressed

❑ Output Data is NOT compressed

Workload

- Consolidated Audit Trail System (part of FINRA application) DB Schema
  - Single table with 12 columns related to share order.
- Data consolidation query
  - Print share order details for share orders during a date range.
  - SELECT issue_symbol,orf_order_id, orf_order_received_ts FROM default.rt_query_extract WHERE issue_symbol like 'XLP' AND from_unixtime(cast((orf_order_received_ts/1000) as BIGINT),'yyyy-MM-ddhh:ii:ss') >= "2014-06-26 23:00:00" AND from_unixtime(cast((orf_order_received_ts/1000) as BIGINT),'yyyy-MM-ddhh:ii:ss') <= "2014-06-27 11:00:00";

Workload Implementation

- ❑ DB is a flat file with columns separated using a token
- ❑ Data generator to generate data for the DB
- ❑ Tool to run queries concurrently
- ❑ Query is implemented as Map and Reduce functions

Workload Size

- Concurrency Tests:
  - Query in isolation, concurrency =1
  - Query in concurrent workload, concurrency =5
  - Thinktime = 10% of query execution time in isolation.

- Data Size:
  - 100GB , 500GB, 1TB and 7TB

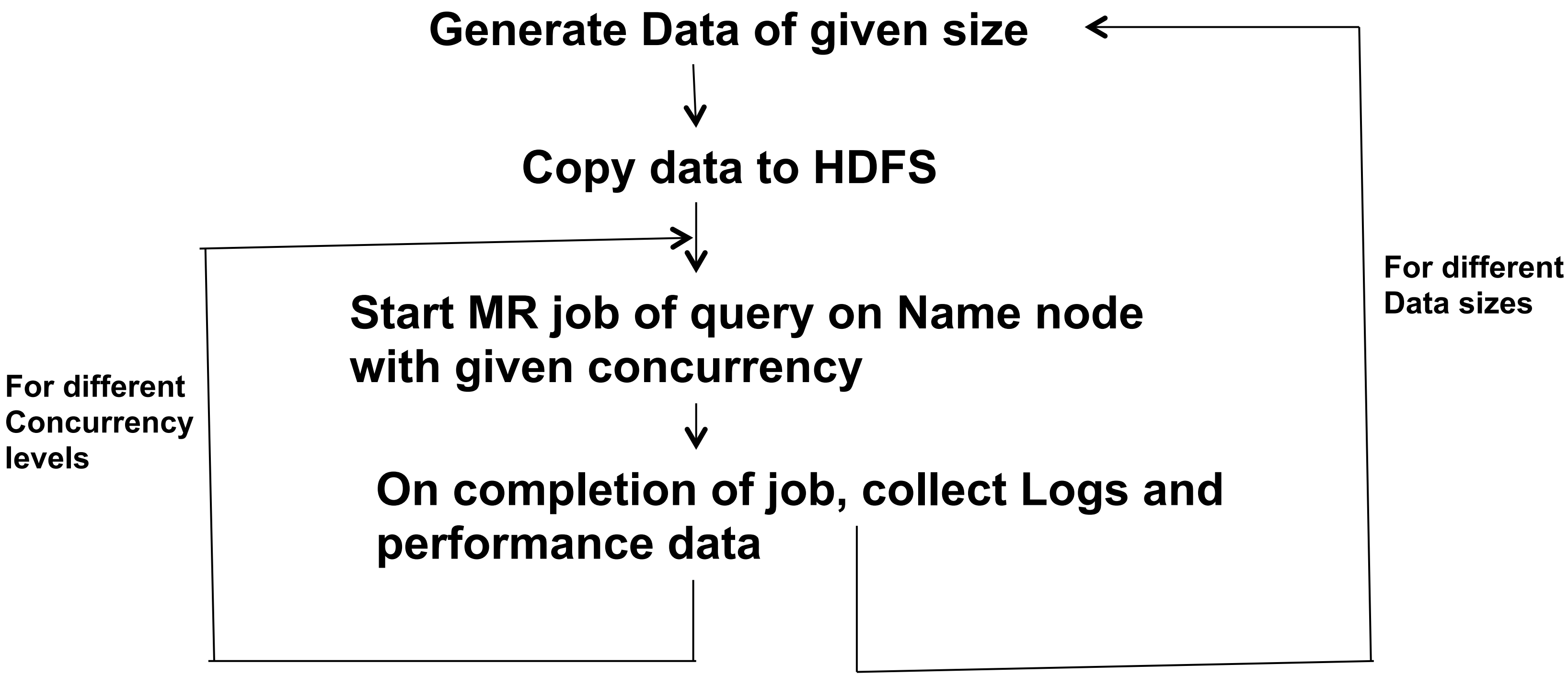
Performance Metric

- MR job execution time in isolation
- MR job average execution time in concurrent workload
- CPU, Disk and Memory Utilization of the cluster

Performance Measurement

- ❑ SAR data is collected from all nodes in the cluster.
- ❑ MapReduce job log files are used for performance analysis
- ❑ Intel® EE for Lustre* software nodes performance data is collected using Intel Manager
- ❑ Hadoop performance data is collected using Intel Manager

Benchmarking Steps



Generate Data of given size

Copy data to HDFS

Start MR job of query on Name node with given concurrency

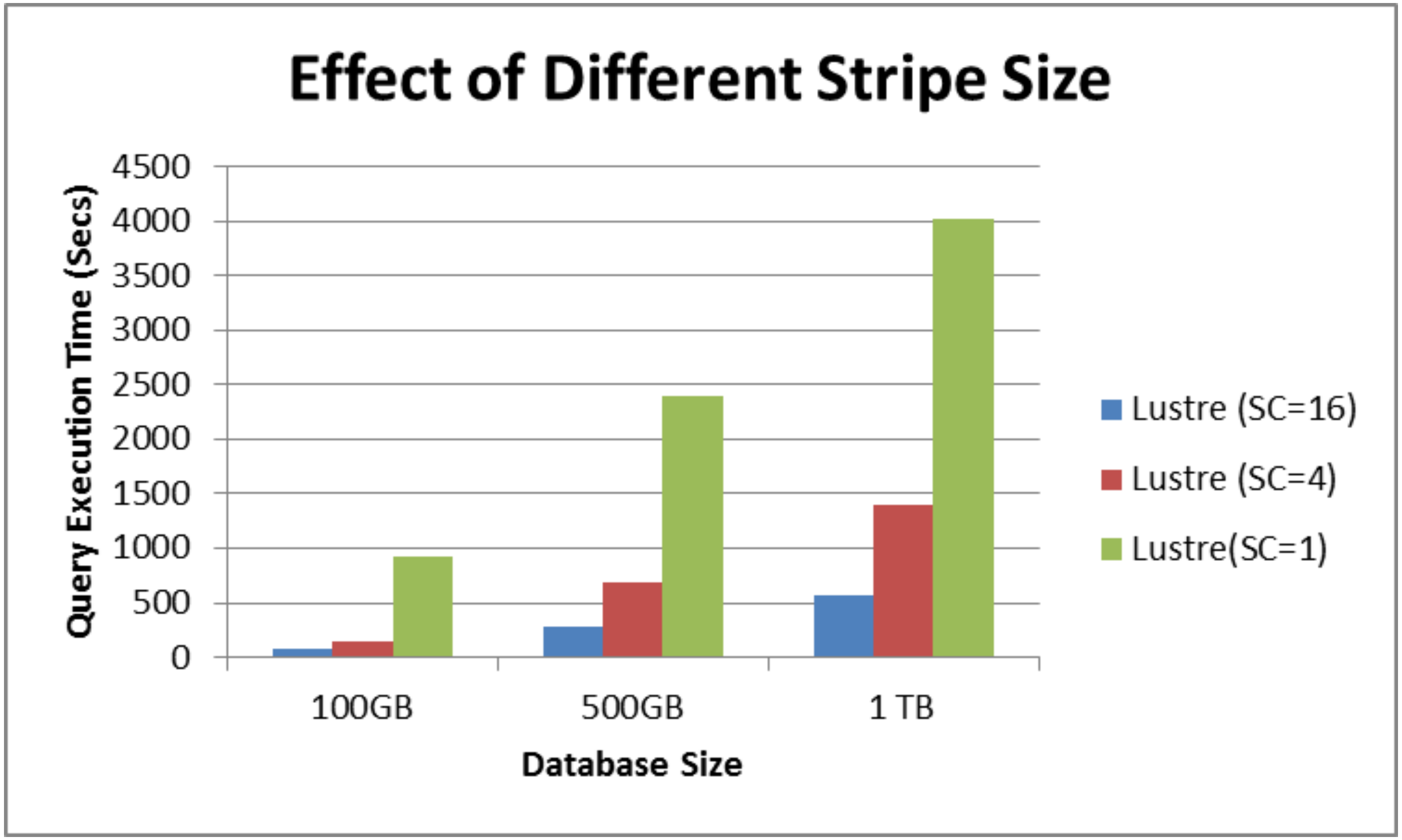
On completion of job, collect Logs and performance data

For different Concurrency levels

For different Data sizes

RESULT ANALYSIS

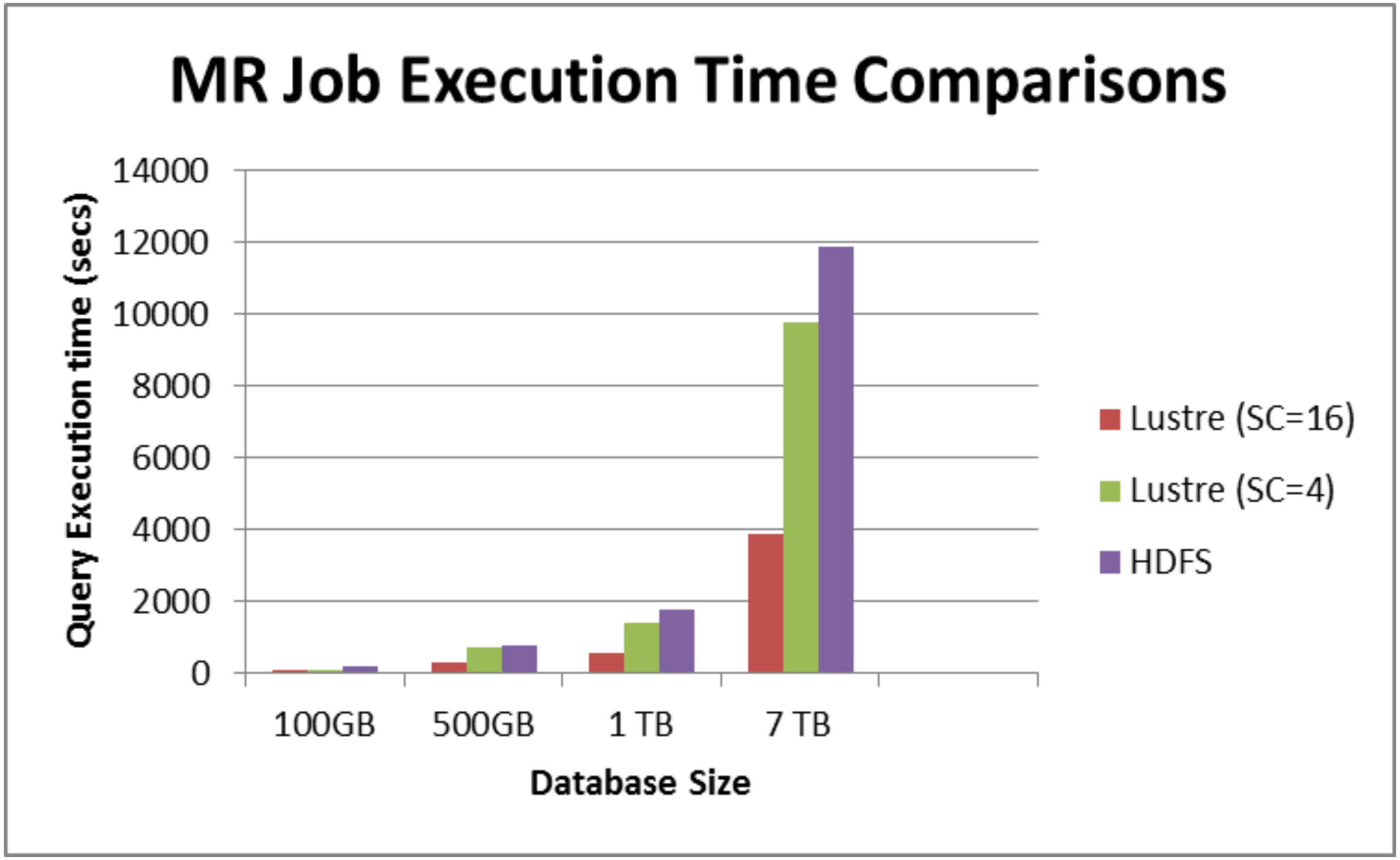
Degree of Concurrency = 1



Intel® EE for Lustre* performs better on large stripe count

* Other names and brands may be claimed as the property of others.

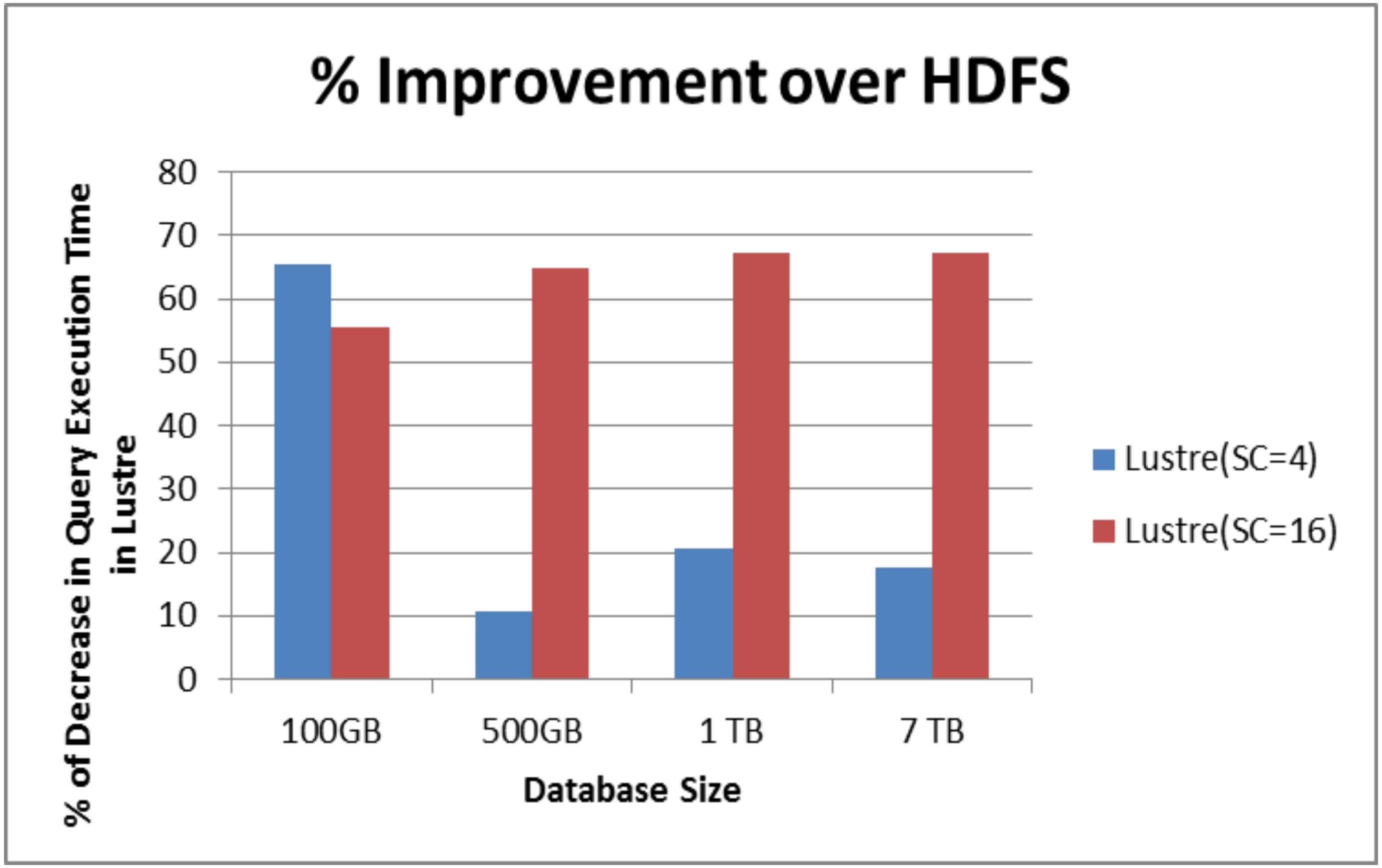
Degree of Concurrency = 1



Intel® EE for Lustre* delivered 3X HDFS for optimal SC settings

* Other names and brands may be claimed as the property of others.

Degree of Concurrency = 1

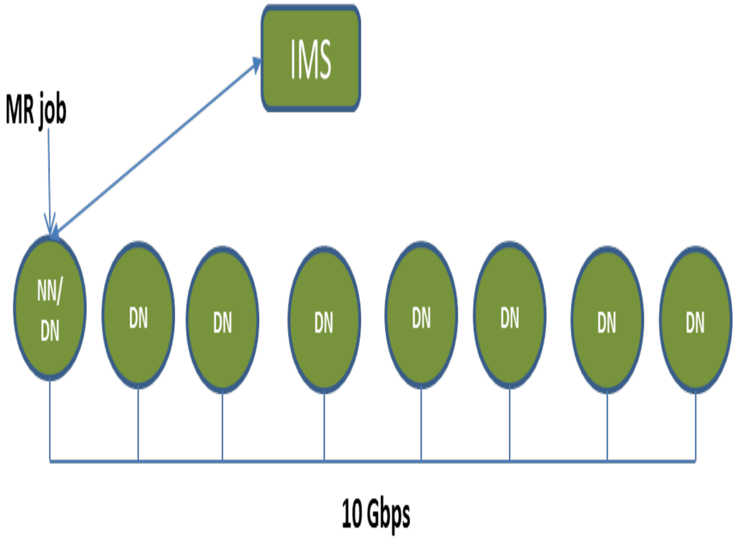


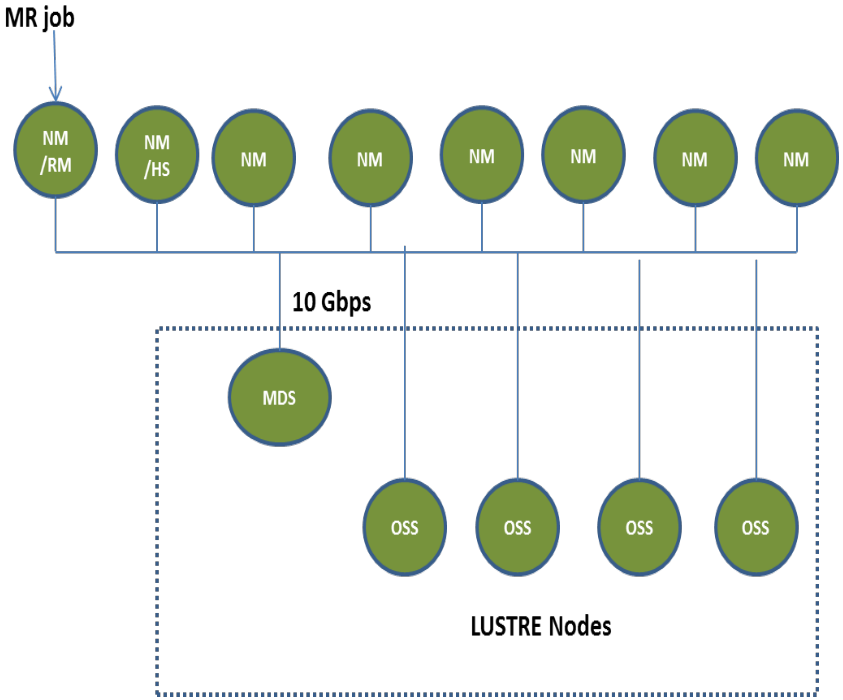
Intel® EE for Lustre* optimal SC gives 70% improvement over HDFS

* Other names and brands may be claimed as the property of others.

Hadoop + HDFS Setup

Hadoop + Intel® EE for Lustre* software - Setup





Nodes = 8

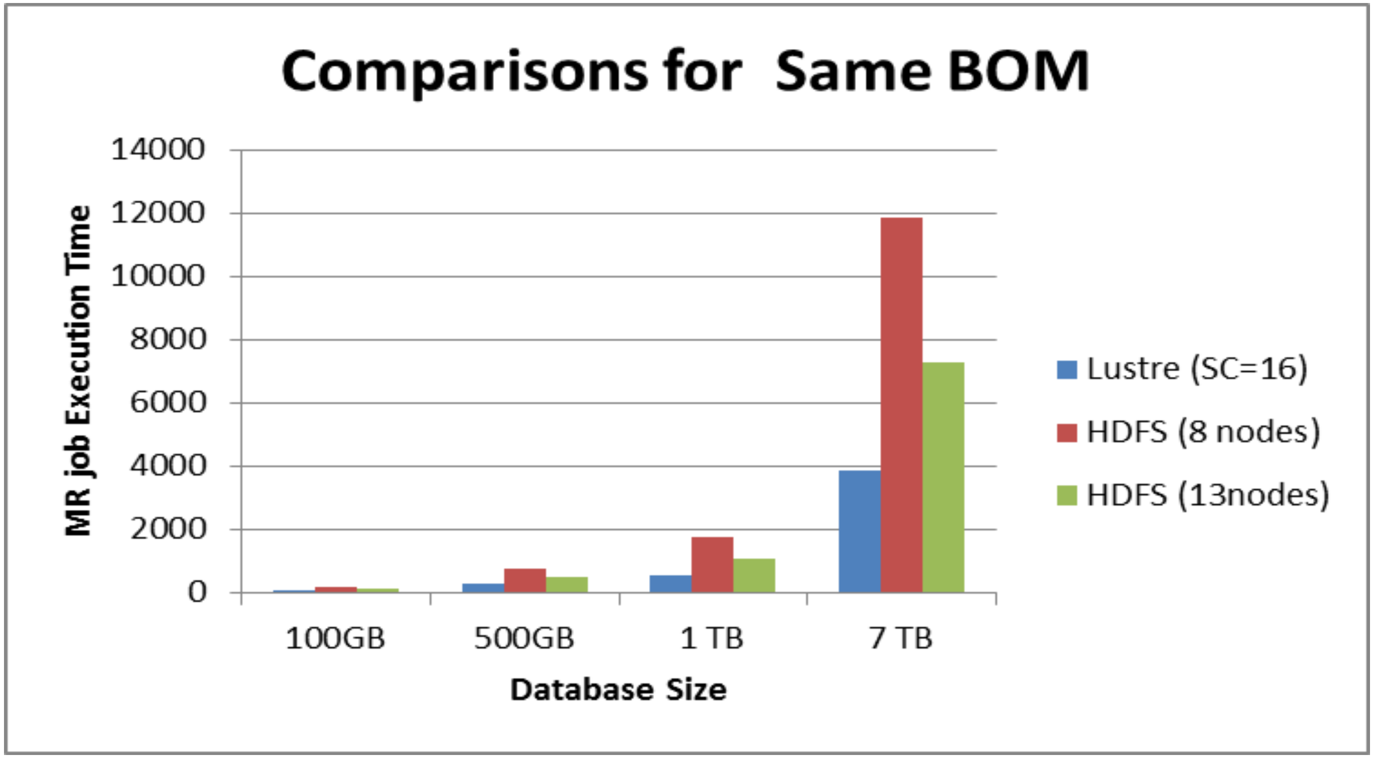
Nodes = 8+5 = 13

Performance Linear extrapolation for Nodes =13

TATA CONSULTANCY SERVICES

* Other names and brands may be claimed as the property of others.

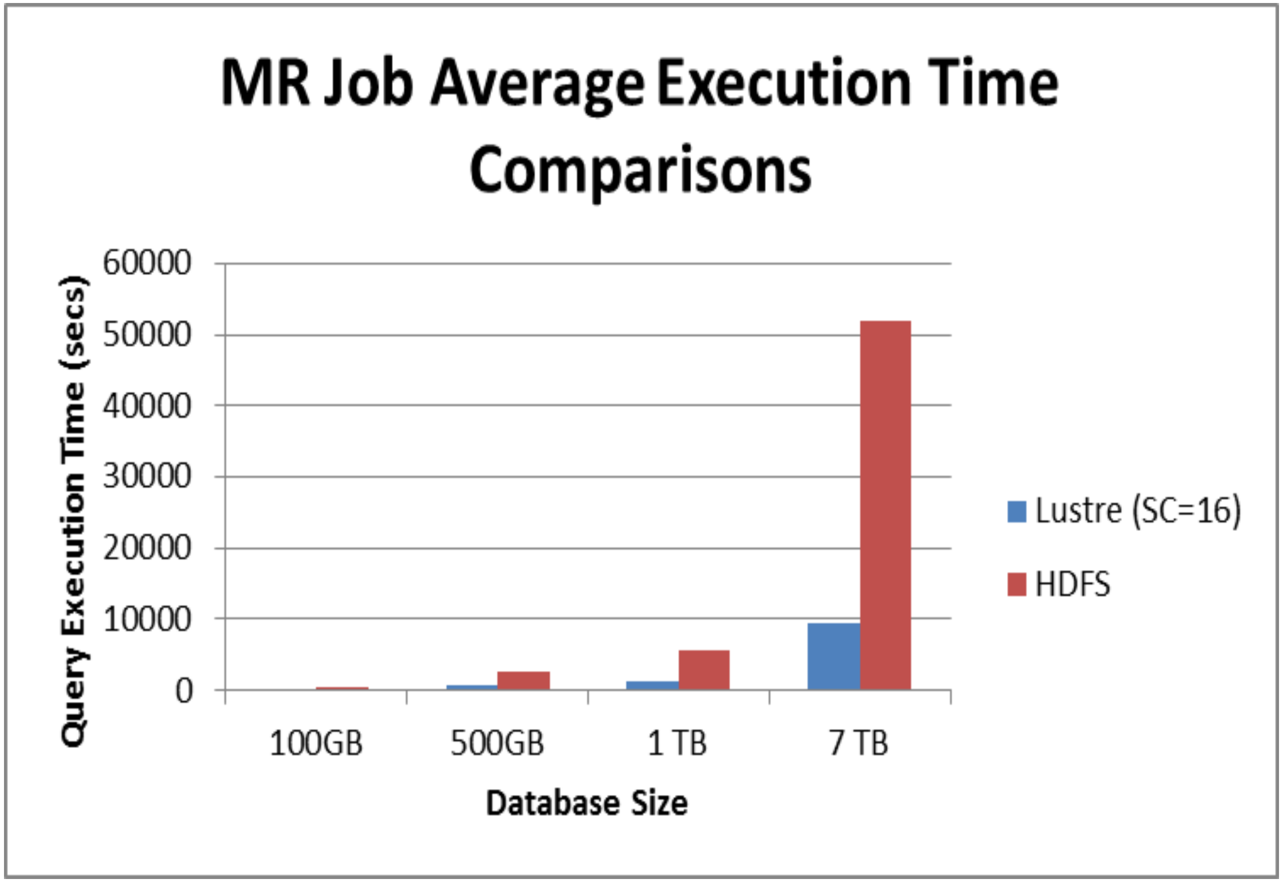
Number of Compute Servers = 13



Intel® EE for Lustre* 2X better than HDFS for same BOM

* Other names and brands may be claimed as the property of others.

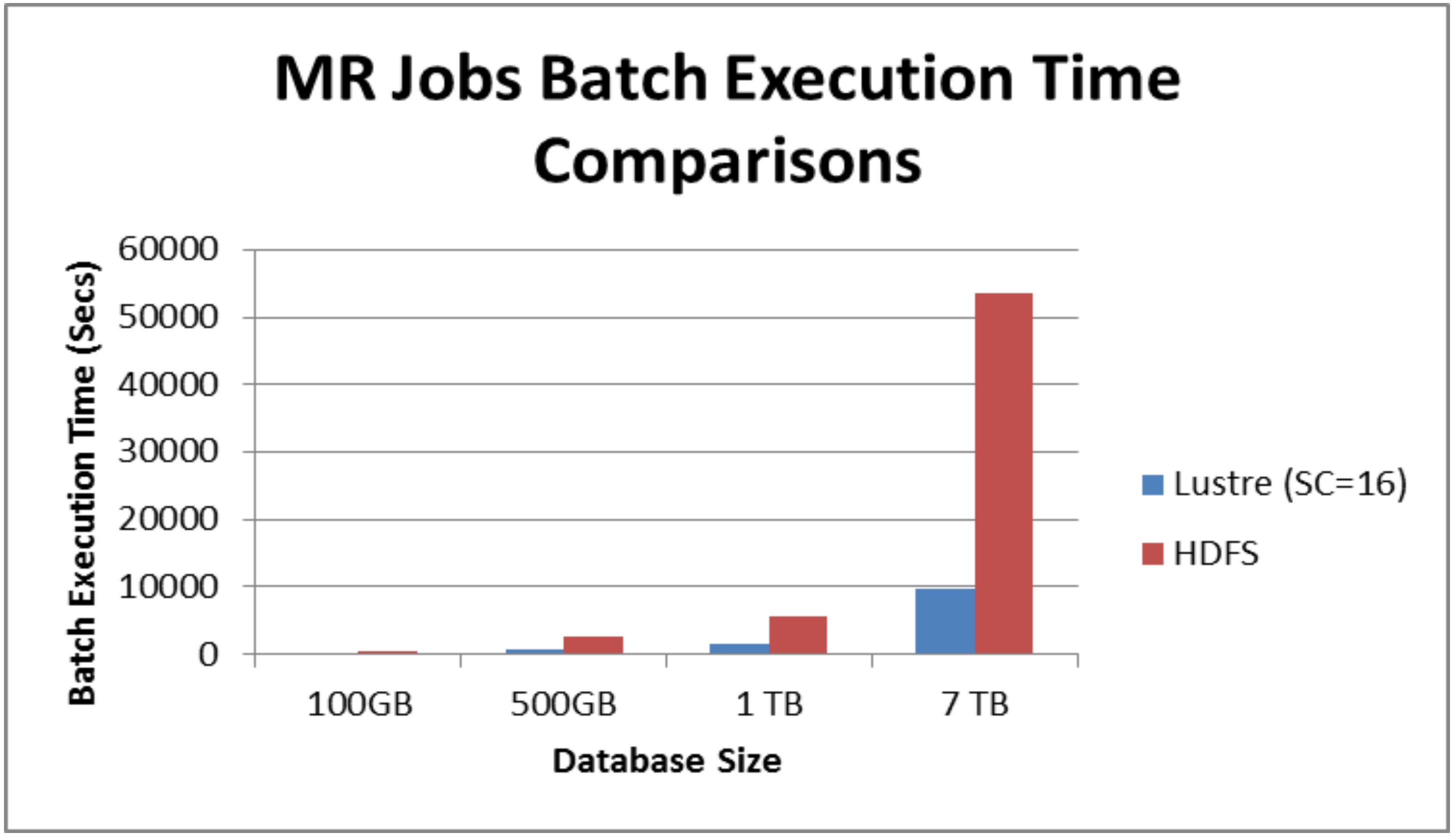
Degree of Concurrency = 5



Intel® EE for Lustre* was 5.5 times better than HDFS on 7 TB data size

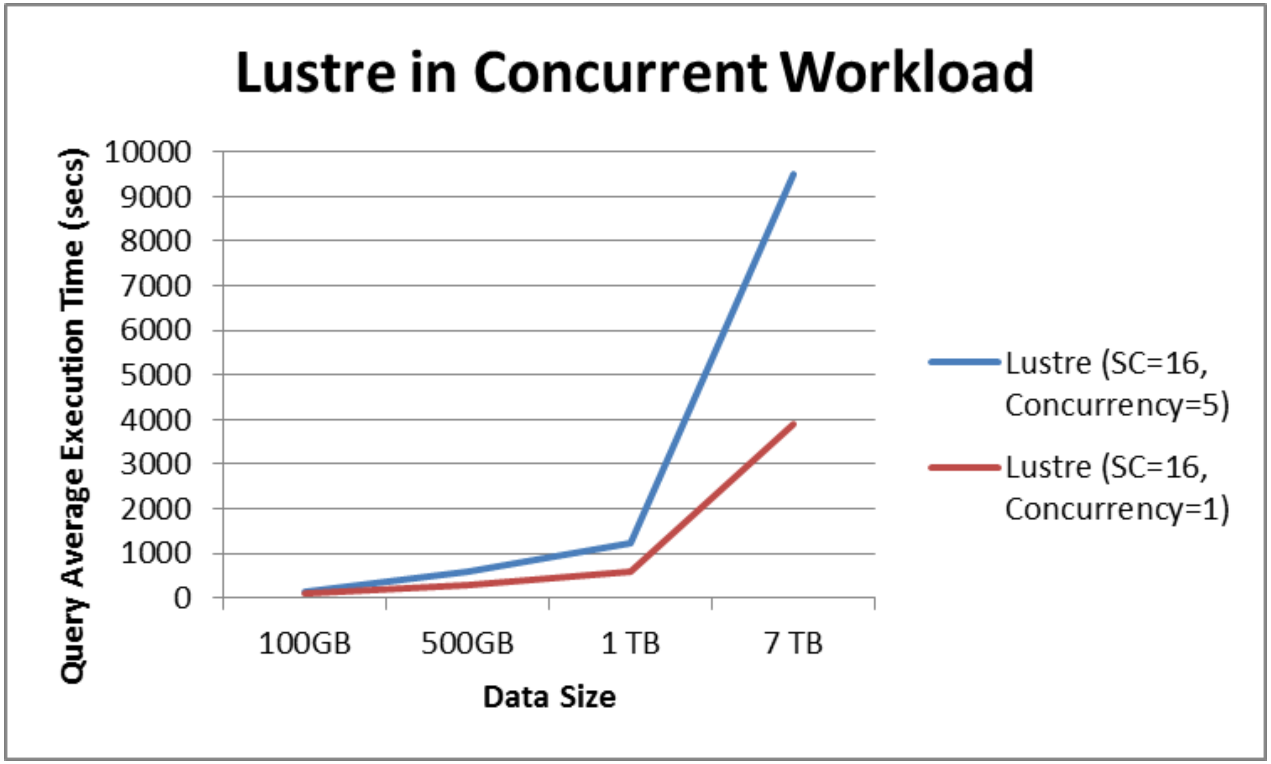
* Other names and brands may be claimed as the property of others.

Degree of Concurrency = 5

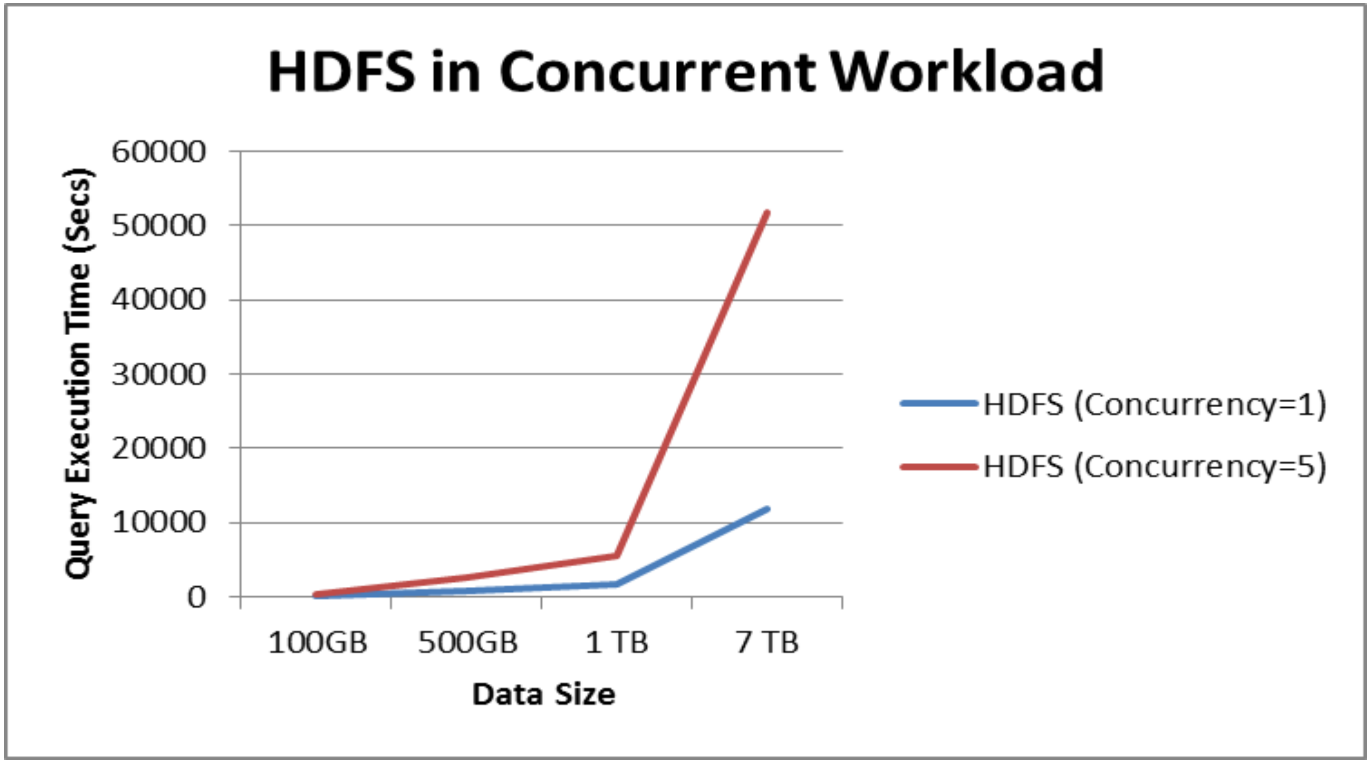


Intel® EE for Lustre* was 5.5 times better than HDFS on 7 TB data size

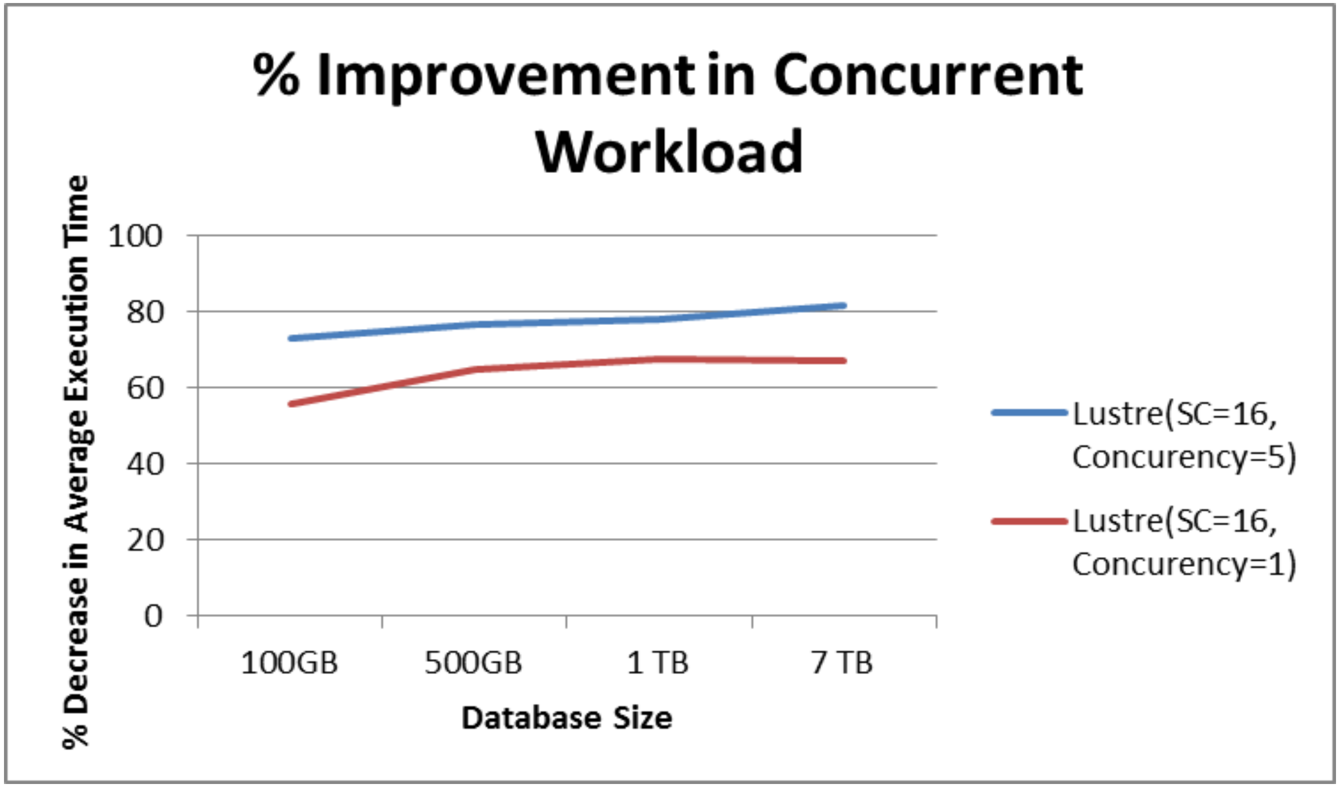
* Other names and brands may be claimed as the property of others.



Concurrent Job Average Execution Time/Single Job Execution Time **= 2.5**


HDFS in Concurrent Workload
Query Execution Time (Secs)
60000
50000
40000
30000
20000
10000
0
100GB
500GB
1 TB
7 TB
Data Size
HDFS (Concurrency=1)
HDFS (Concurrency=5)

= 4.5



Intel® EE for Lustre* software > HDFS for concurrency

* Other names and brands may be claimed as the property of others.

Conclusion

- ❑ Increase in Stripe count improves Enterprise Edition for Lustre* software performance
- ❑ Intel® EE for Lustre shows better performance for concurrent workload
- ❑ Intel® EE for Lustre software = 3 X HDFS for single job
- ❑ Intel® EE for Lustre software = 5.5 X HDFS for concurrent workload
- ❑ Future work
  - Impact of large number of compute nodes (i.e. OSSs <<<< Nodes)

* Other names and brands may be claimed as the property of others.

TATA CONSULTANCY SERVICES

TATA

Thank You

rekha.singhal@tcs.com

Experience certainty.

IT Services
Business Solutions
Outsourcing