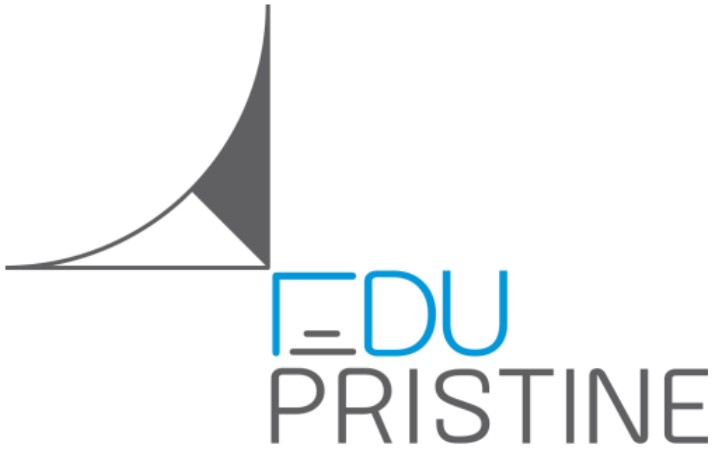


Apache Spark

Agenda

- Recap of Hadoop
- Limitations of Hadoop
- Opportunity for In memory computing
- Spark Ecosystem
- Time comparisons with Map Reduce
- History Review
- Conclusion

Traditional tools for data analysis

- SAS
- SPSS
- R
- Python

All have one common problem, they all run on single machine..!!!

It all started in 2000's

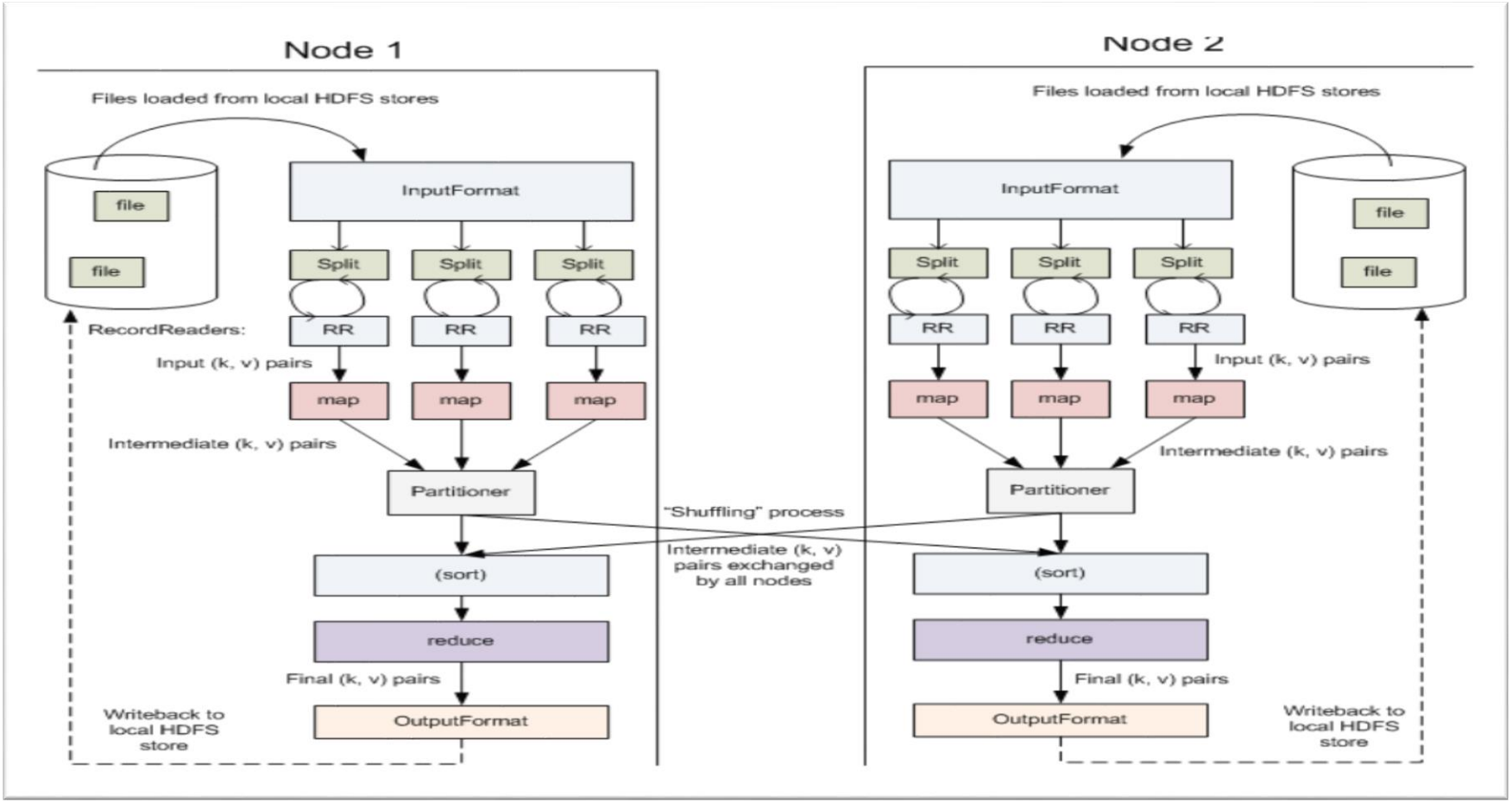
- The word Internet – Connecting machines over the network has led to explosion of data.
- Around in the same time, Google had a problem in storing this massive amount of data and process the same.
- For storage, Google file system is designed
- For processing, Map reduce is designed
- The image which is in the back ground, is Google data center, with 50000+ commodity machines.

Hadoop - 2004

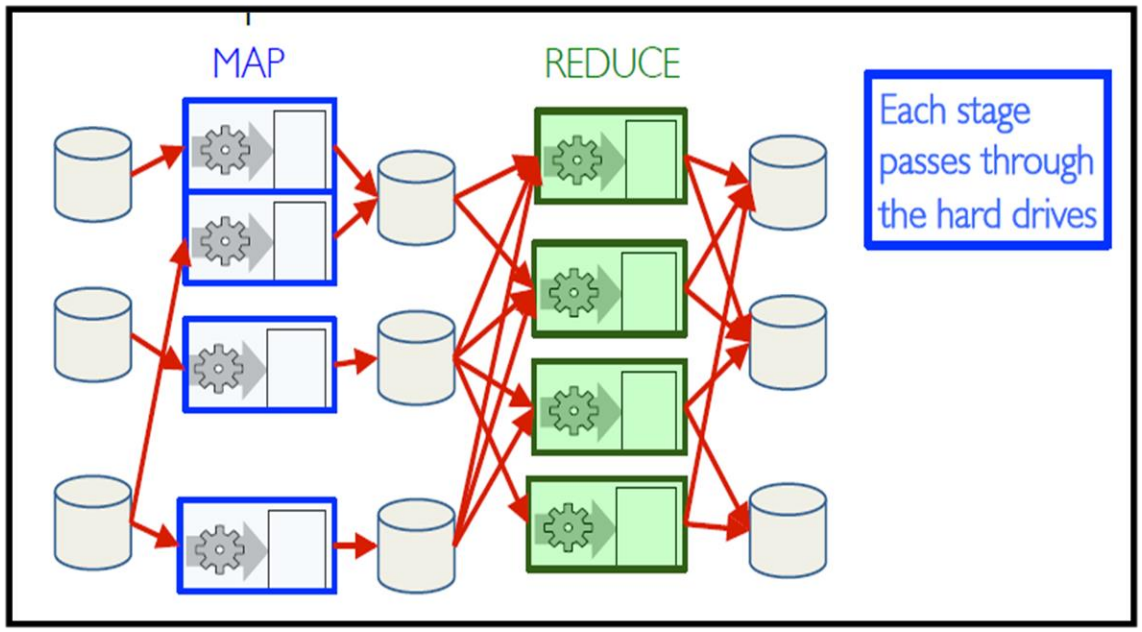
- Doug cutting, designed a web crawler, to crawl the data from web. But imagine, world wide web has zillion documents, how would one store that?
- Doug had the same question and he read google white paper, and re implemented it.
- Now what can be the name of tool? Ask his Kid, he knows it better!! His kid's toy elephant name is hadoop!!

Recap of Map Reduce

- Stages in map phase; Inputformat, split, record reader, mapper, partitioner stages .
- Stages in reduce phase; Shuffle, sort, reduce, outputformat, record writer.



Map Reduce Distributed Execution



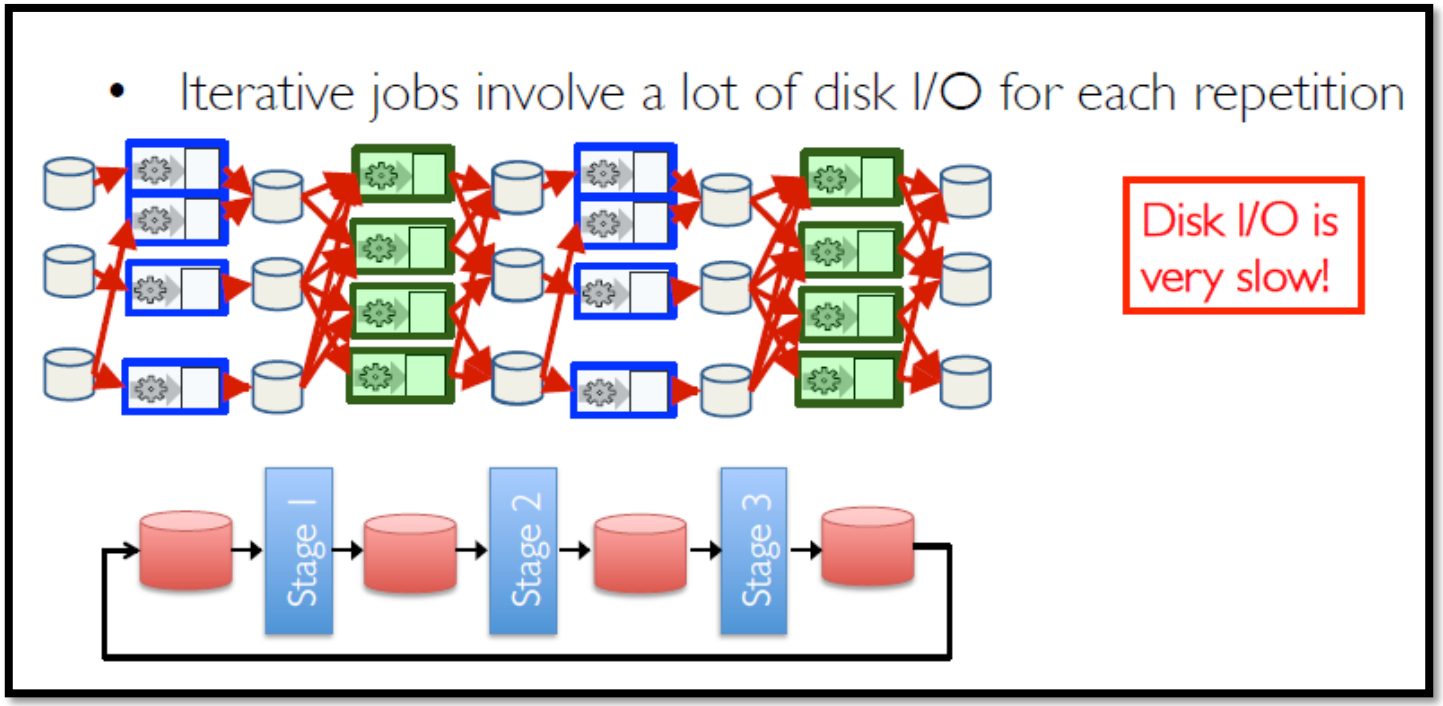
In map reduce, each stage passes through disk.

- Input Data is on HDFS
- Output of mapper is on local disk
- Output of Shuffle and sort is on local disk
- Output of reducer will be written back to HDFS

If you carefully observe, map reduce is disk IO intensive.

Map Reduce: Iterative jobs

- Now think about iterative jobs. Say there are multiple map reduce stages, and each Stage involves disk I/O. In the above diagram, there will be 6 local disk I/O's and 6 HDFS I/O Operations.
- Now you understand, why map reduce is slow!!



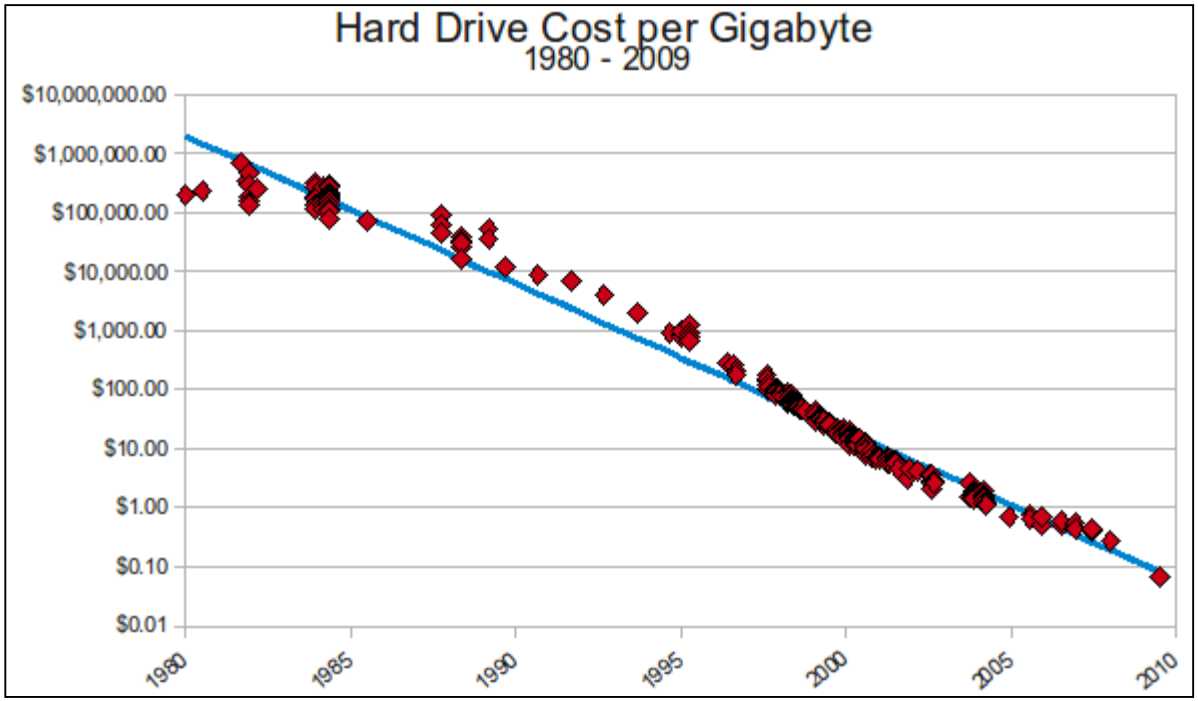
Hardware for Big Data

By this time, you should have understood The hardware ware for bigdata is:

- Many Hard disks
- Many CPU's

Why so many hard disks?

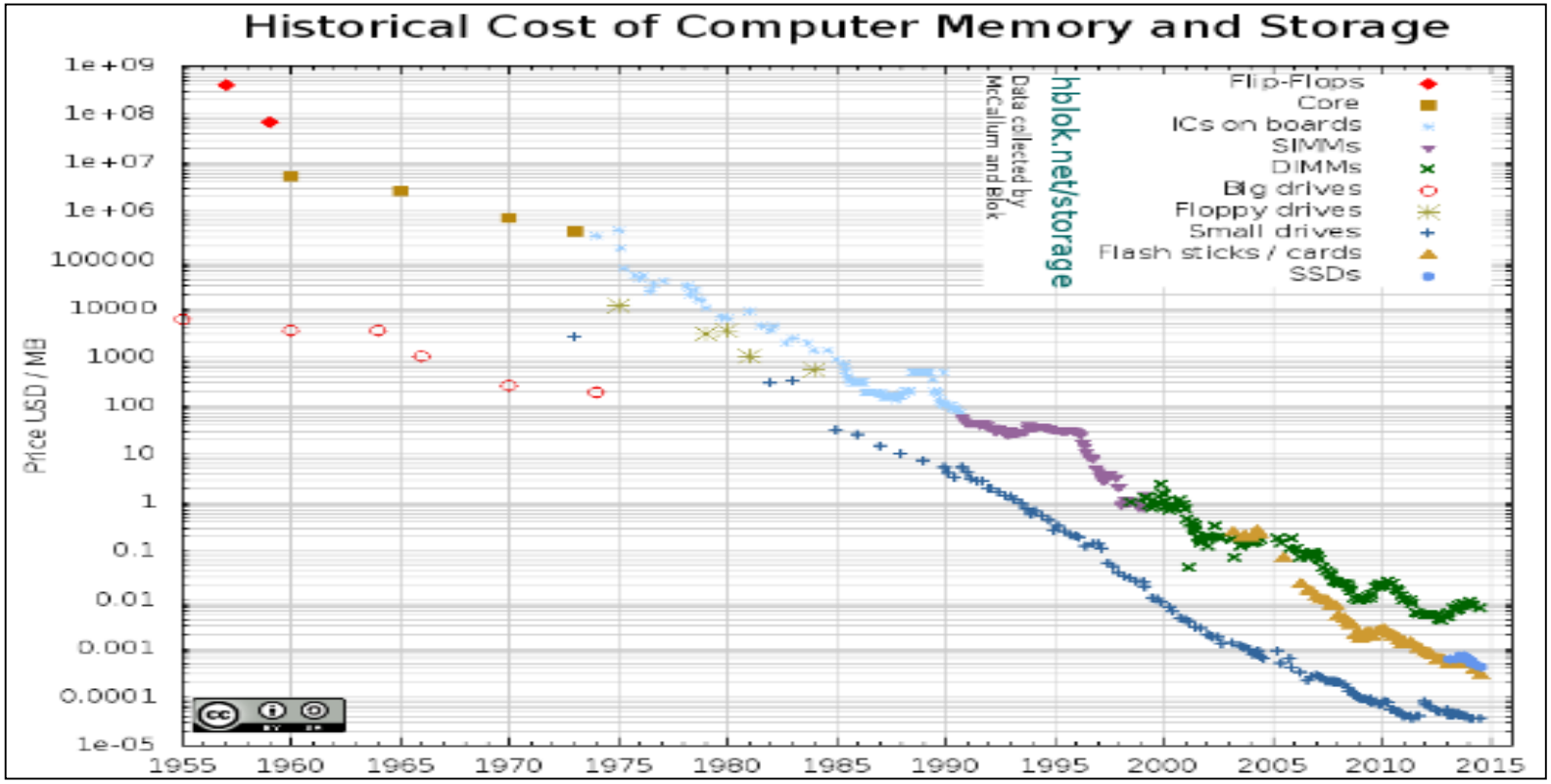
- As you see, cost per gb over the years is linearly coming down, and now it has become so cheap!!



University of california had a question

- In a big data cluster, as the number of machines increase, size of cluster scales up.
- In that case, cant we use RAM for faster processing, as even the RAM available on cluster scales up with size of cluster.

Tech Trend: Cost of memory



Lower cost means can put more memory in each server

Current hardware for Big Data

- Now that RAM cost is coming down,
  - Hard ware for bigdata:
  - Storage; many harddisks
  - Processing; CPU
  - Faster processing: RAM

Lots of hard drives

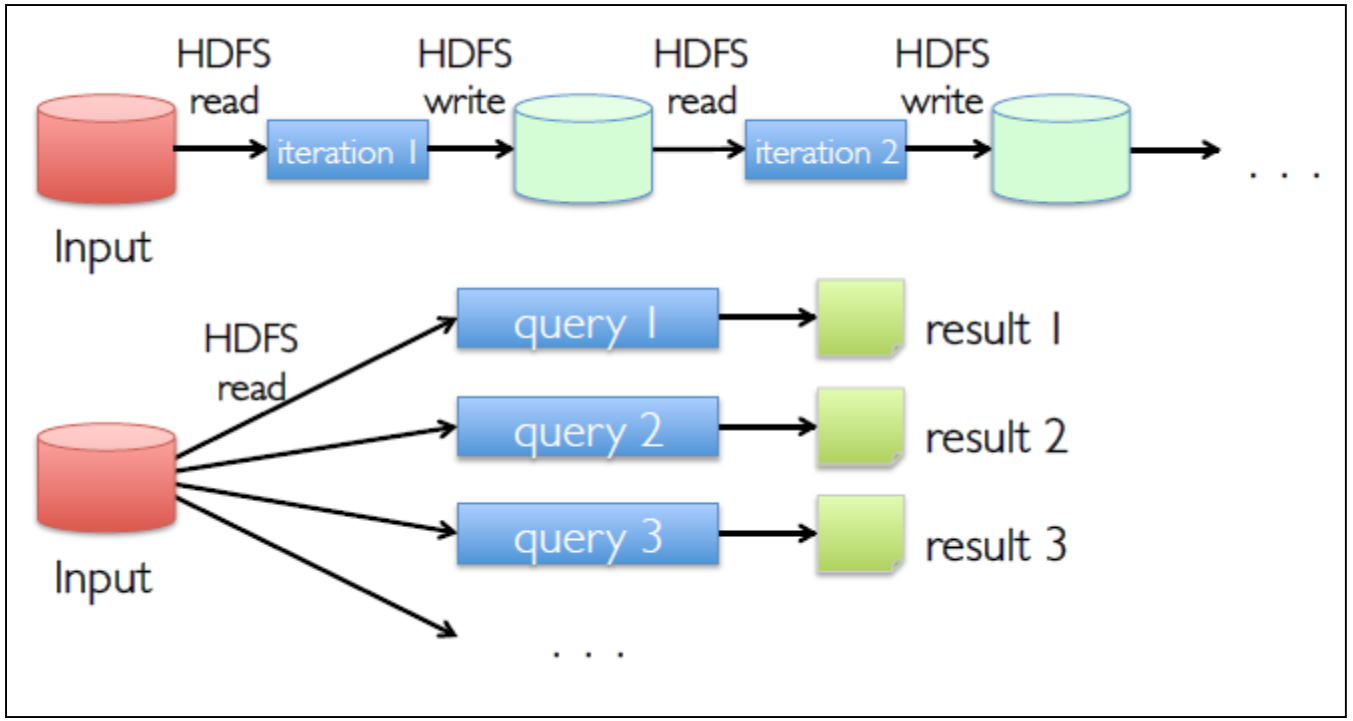
…. CPU's

Opportunity

- Keep more data in-memory
- Chance for lighting fast cluster computing.
- Design a new distributed execution engine:

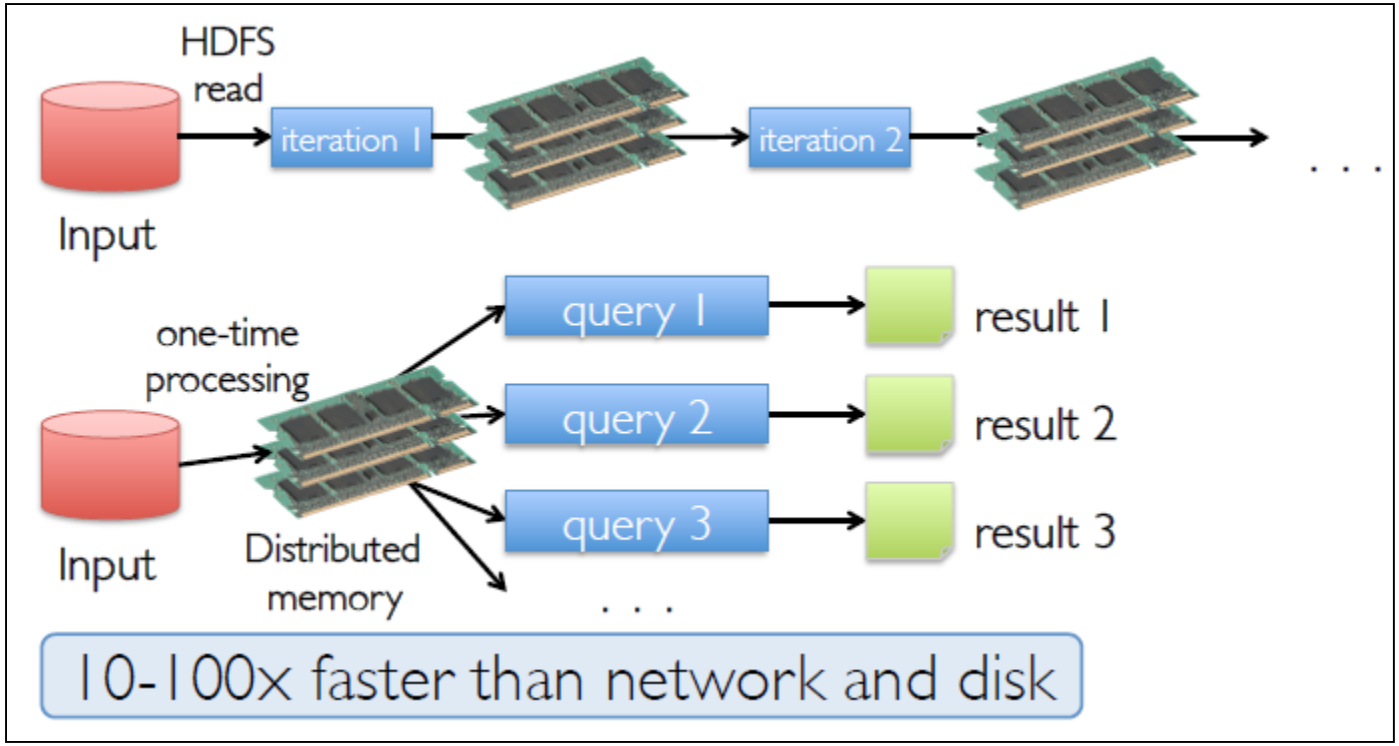
Use Memory instead of Disk

- In map reduce, each iteration leads to HDFS I/O operations.
- Adding to this, consider you have a source data, on which you want to perform different queries. Would you like to read the same data, for each query???



In-Memory Data Sharing

- Now for faster processing and quick data sharing, use Memory.
- In the previous example, you read the same data multiple times, now with distributed in memory, you have an option to cache data on distributed memory, perform all the required queries at much faster pace!!

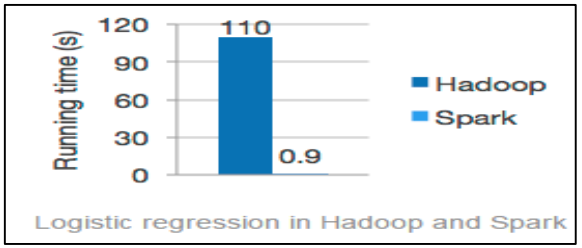


Apache Spark Motivation

- Using Map Reduce for complex jobs, interactive queries and online processing involves lots of disk I/O!
- Keep more data in memory, when ever possible.

Key points of Apache Spark

- Fast and general engine for large-scale data processing.
- **Speed**; Run programs up to 100X faster than hadoop.



Logistic regression in Hadoop and Spark

- **Ease of use**: Write applications in language you are comfortable in; Java, Scala, Python and R.

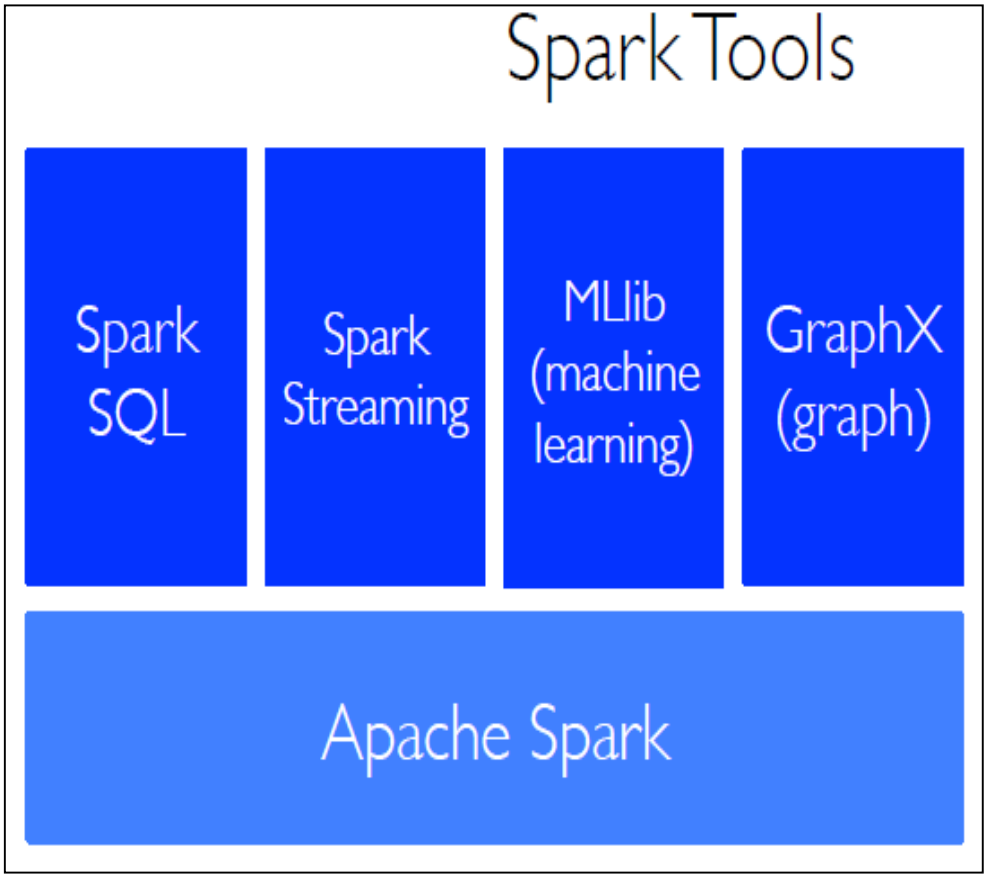
```
text_file = spark.textFile("hdfs://...")

text_file.flatMap(lambda line: line.split())
    .map(lambda word: (word, 1))
    .reduceByKey(lambda a, b: a+b)
```

Word count in Spark's Python API

- **Generality**: Combine SparkSQL, streaming and complex analytics
- **Runs Everywhere**: Spark runs on standalone, on YARN, on MESOS. It can connect to NoSQL, Amazon S3!!

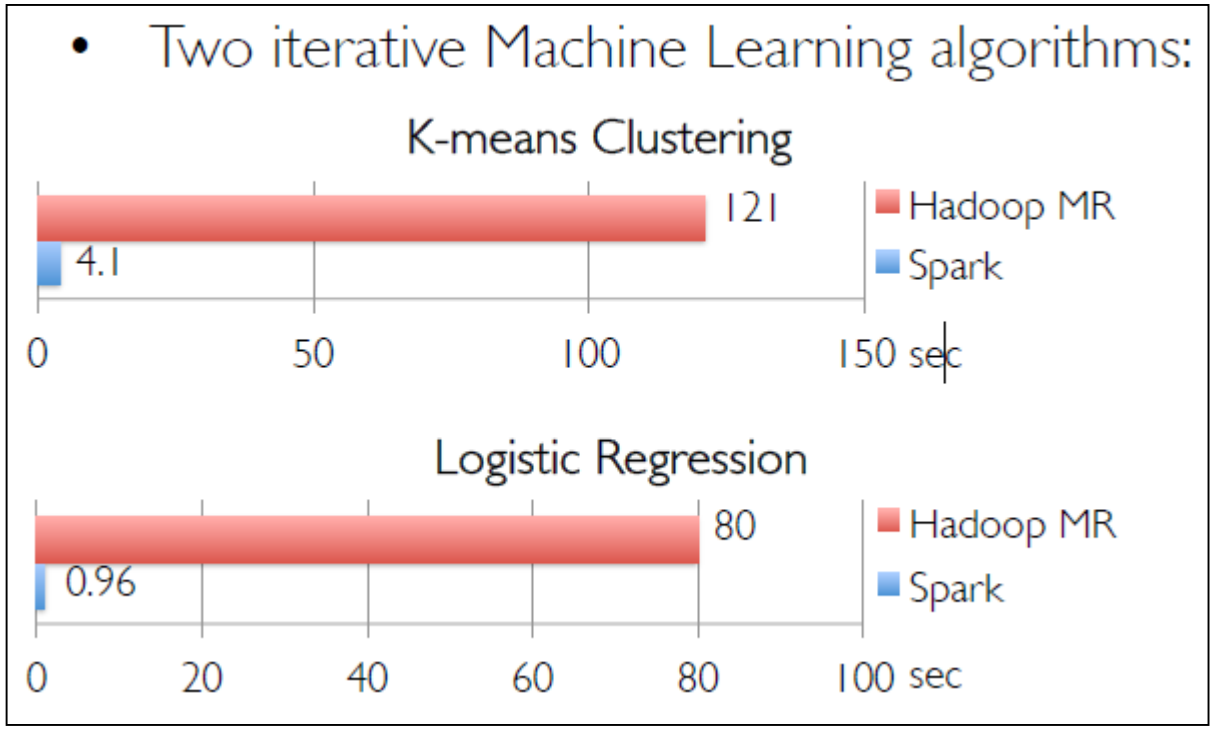
Rise of in memory computing; Ecosystem



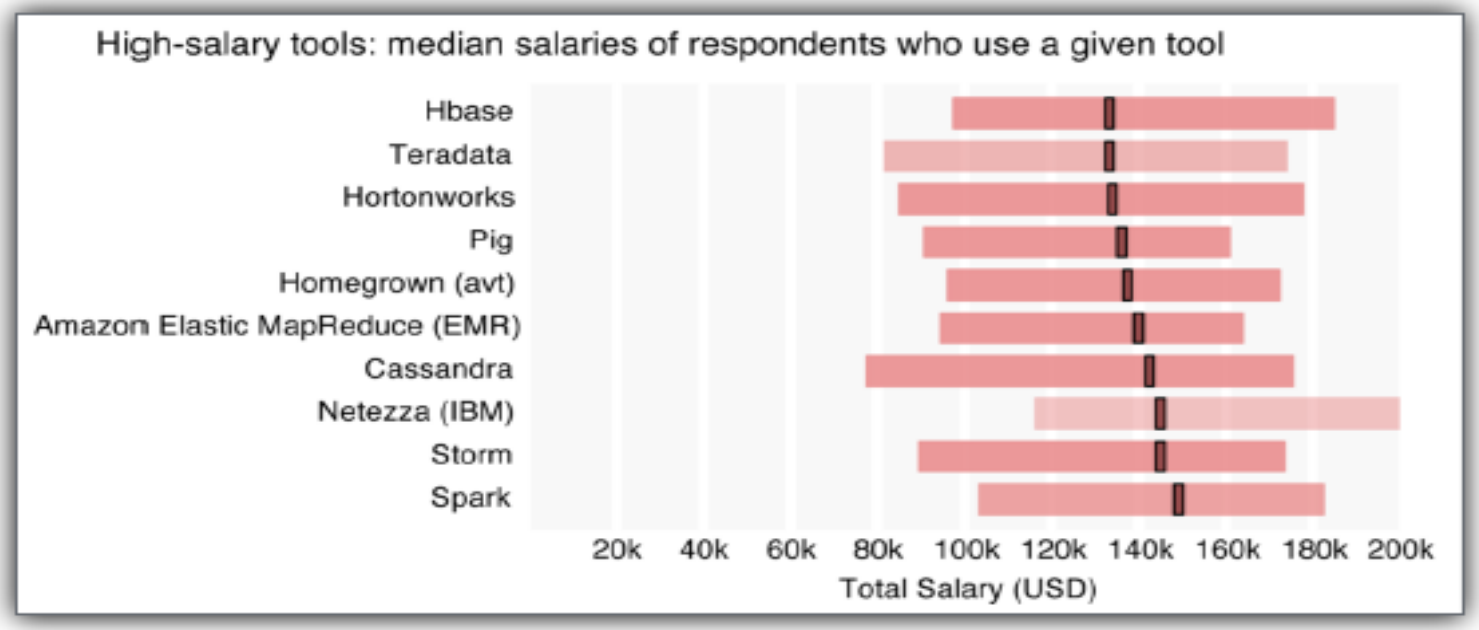
- **Apache spark:**
  - Core library for developers.
- **Spark SQL:**
  - Enabling data analysts to write queries on spark.
- **Spark Streaming:**
  - Real time analytics.
- **Mllib**:
  - Machine learning libraries for predictive analytics
- **GraphX**:
  - Graph Processing

In-Memory can make a big difference

- Iterative machine learning algorithms, will require multiple stages of jobs to predict the outcome.
- In this case, Spark compared to Hadoop MR, runs near 100 times faster!!



Spark is worth to learn!!



Over 800 respondents across 53 countries and 41 U.S. states

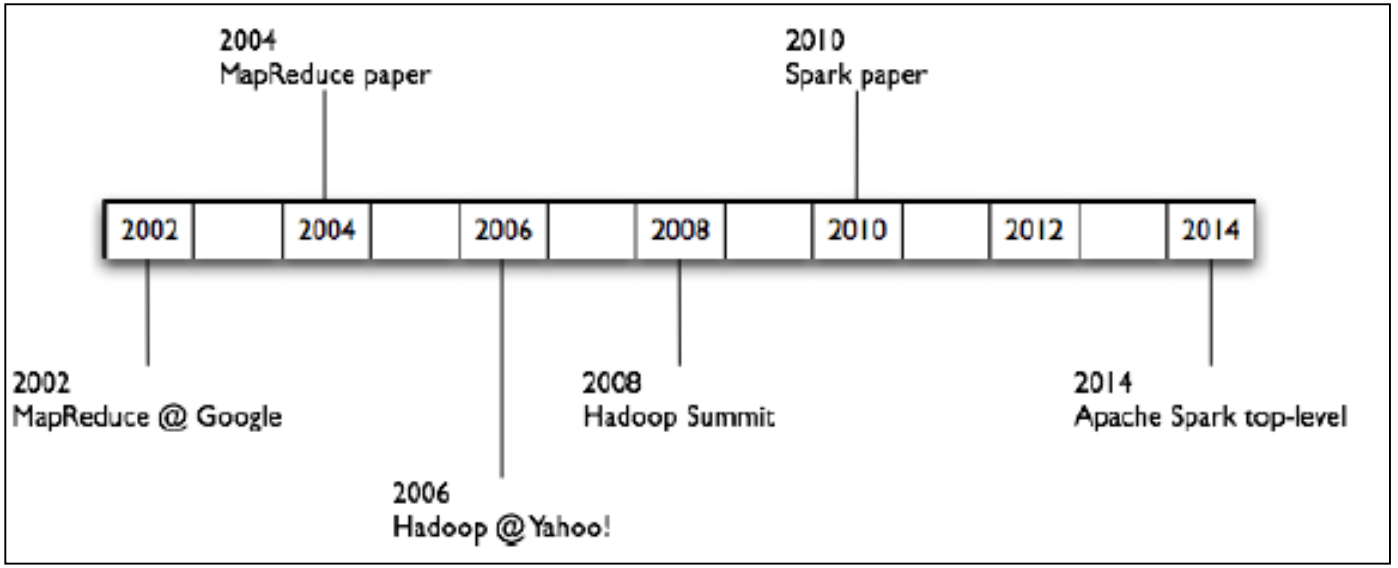
http://www.oreilly.com/data/free/2014-data-science-salary-survey.csp

Stress Testing: First cloud peta byte sort!

	Hadoop MR Record	Spark Record	Spark 1 PB
Data Size	102.5 TB	100 TB	1000 TB
Elapsed Time	72 mins	23 mins	234 mins
# Nodes	2100	206	190
# Cores	50400 physical	6592 virtualized	6080 virtualized
Cluster disk throughput	3150 GB/s (est.)	618 GB/s	570 GB/s
Sort Benchmark Daytona Rules	Yes	Yes	No
Network	dedicated data center, 10Gbps	virtualized (EC2) 10Gbps network	virtualized (EC2) 10Gbps network
Sort rate	**1.42 TB/min**	**4.27 TB/min**	**4.27 TB/min**
Sort rate/node	**0.67 GB/min**	**20.7 GB/min**	**22.5 GB/min**

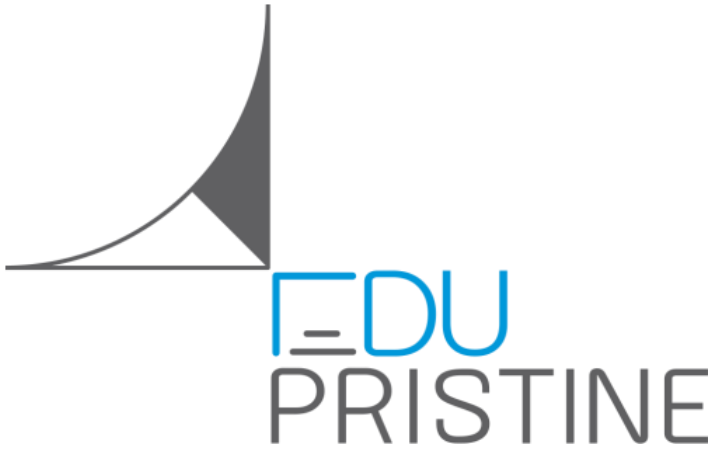
http://databricks.com/blog/2014/11/05/spark-officially-sets-a-new-record-in-large-scale-sorting.html

History Review



Conclusion

- Spark is more like an advanced version of Map Reduce.
- In Map Reduce, programmer is expected to know the entire frame work, and write them.
- In Map Reduce, no matter how small the operation is, it will execute the full cycle, starting from mapper, shuffle and sort and reducer, if you even don't want them to be executed.
- Each Map Reduce job involves, lots of Disk IO.
- Looking at the limitations of Map Reduce, spark designed almost 80+ functions, which will support your data analysis needs.
- For example, distinct is a function in map-reduce, which will help you to find distinct values. If you were to do the same in Map Reduce, you should write another program!!
- Spark tries to store data in distributed memory, whenever possible.
- Spark is born out of map reduce, and used the lazy evaluation, data flow concepts of apache pig.



Thank You!

help@edupristine.com

www.edupristine.com