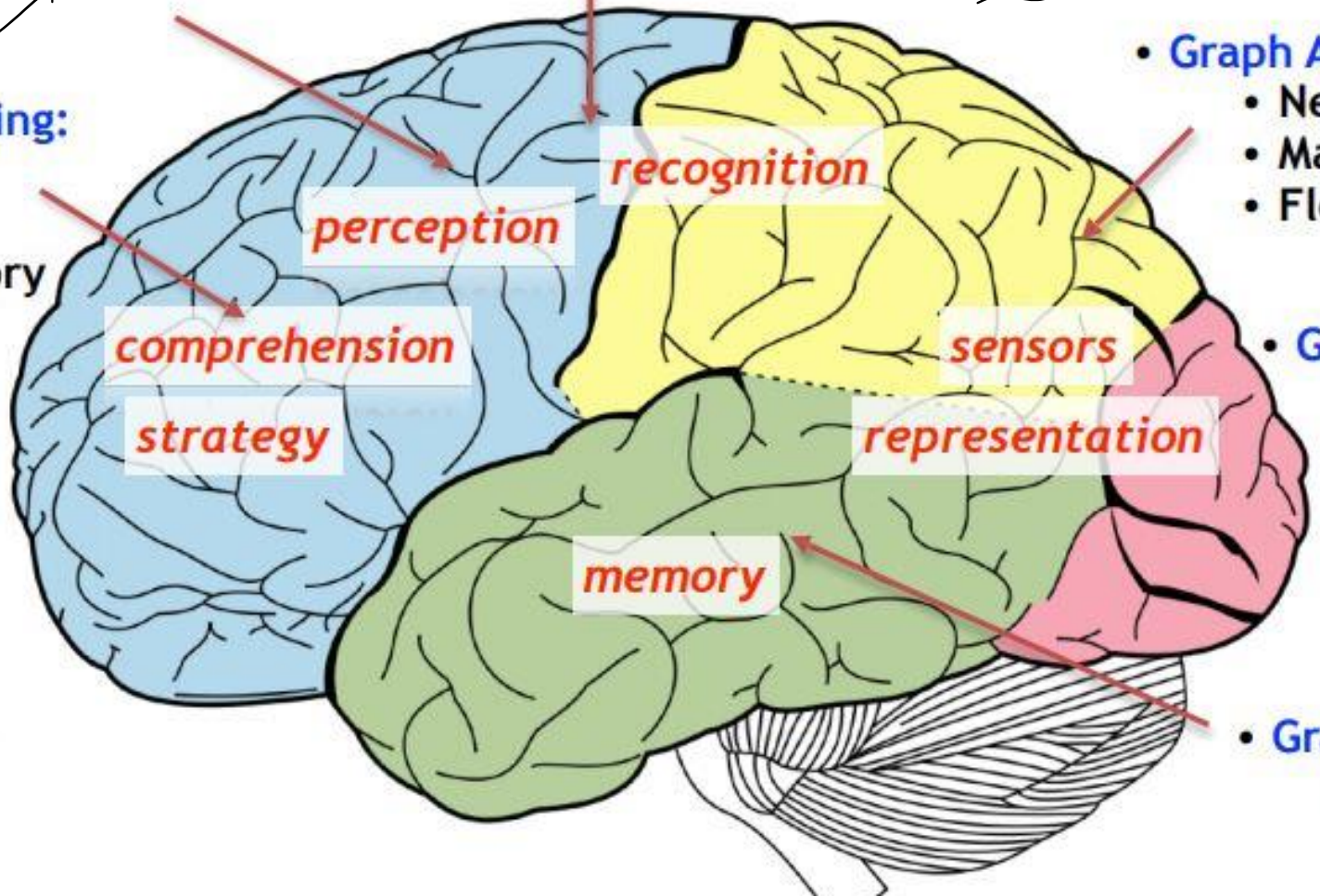
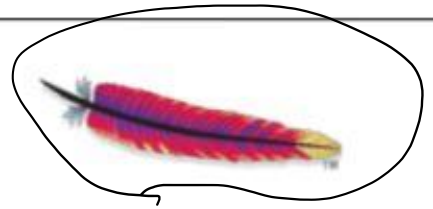
Human brain is a graph/network of 100B nodes and 700T edges.

- **Machine Cognition:**
  - Robot Cognition Tools
  - Feeling
- **Machine Reasoning:**
  - Bayesian Networks
  - Game Theory Tools
- **Machine Learning:**
  - Machine Learning Tools
  - Deep Learning Tools
- **Graph Analytics:**
  - Network Analysis
  - Matching and Search
  - Flow Prediction
- **Graph Visualization:**
  - Dynamic Graph
  - Big Graph
- **Graph Database:**
  - Large-Scale Native Store



Course Main Thrust 1: Apache Hadoop and Big Data



The Apache™ Hadoop® project develops open-source software for reliable, scalable, distributed computing.

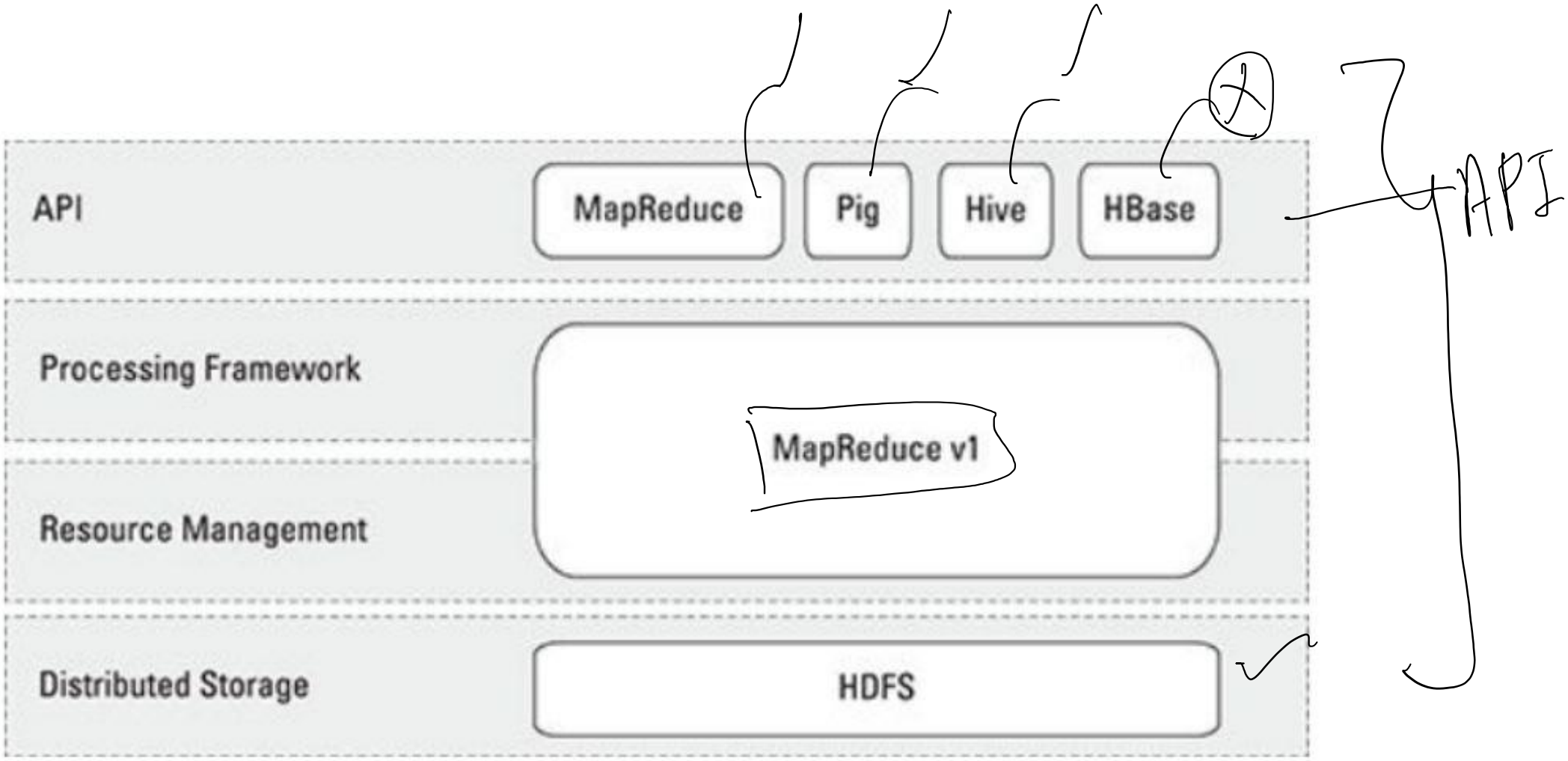
The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

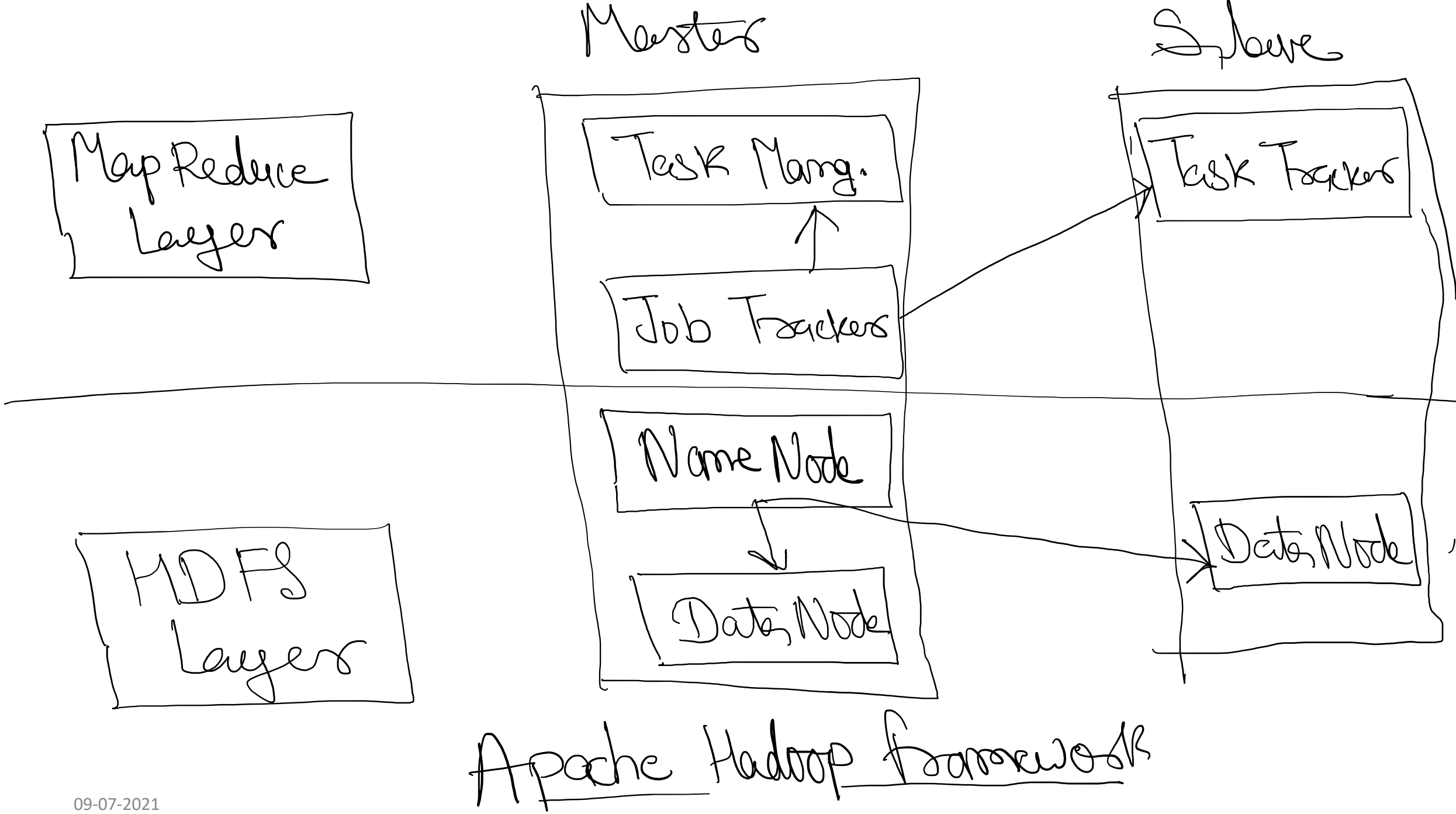
The project includes these modules:

- **Hadoop Common**: The common utilities that support the other Hadoop modules.
- **Hadoop Distributed File System (HDFS™)**: A distributed file system that provides high-throughput access to application data.
- **Hadoop YARN**: A framework for job scheduling and cluster resource management.
- **Hadoop MapReduce**: A YARN-based system for parallel processing of large data sets.

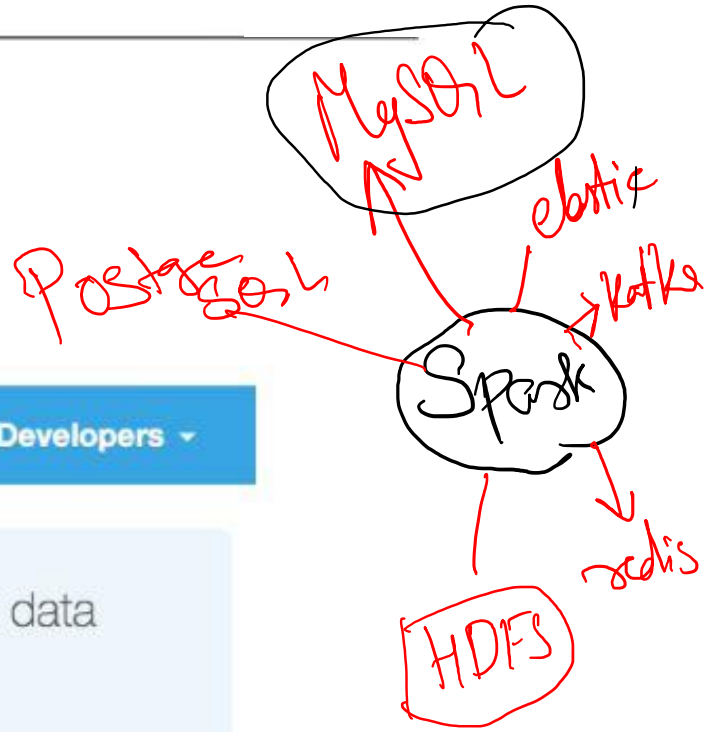
http://hadoop.apache.org

Four distinctive layers of Hadoop




Master
Slave
Map Reduce Layer
Task Mang.
Task Tracker
Job Tracker
Name Node
Data Node
Data Node
HDFS Layer
Apache Hadoop framework

Course Main Thrust 2: Apache Spark and ML



Lightning-fast unified analytics engine

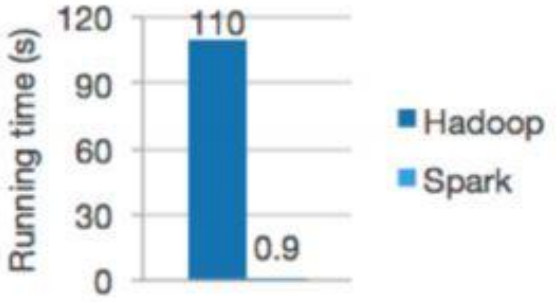
Download | Libraries | Documentation | Examples | Community | Developers

Apache Spark™ is a unified analytics engine for large-scale data processing.

Speed

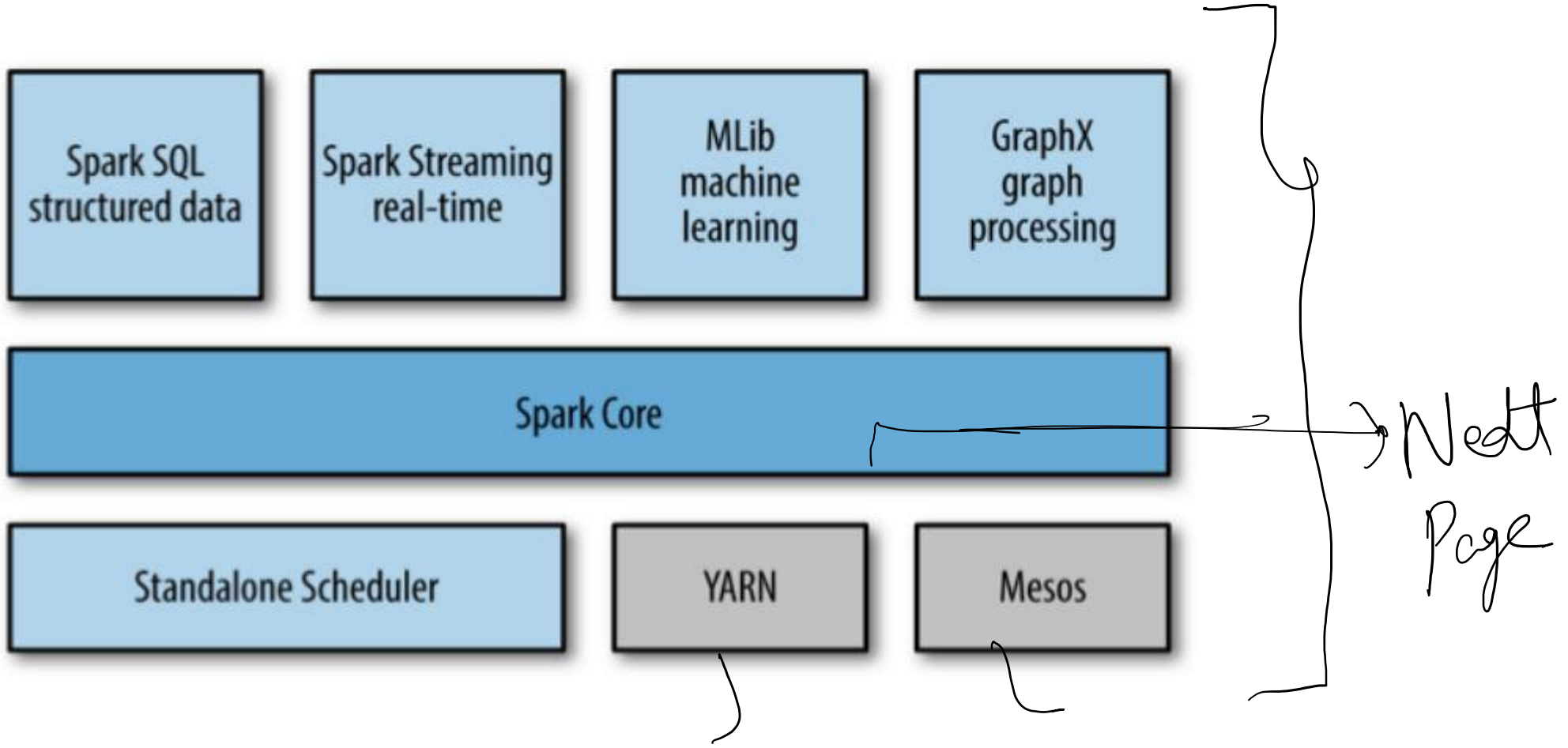
Run workloads 100x faster.

Apache Spark achieves high performance for both batch and streaming data, using a state-of-the-art DAG scheduler, a query optimizer, and a physical execution engine.



Logistic regression in Hadoop and Spark

Main Spark Stack



Spark Core API (Speed, Ease of Use, Engine)

R | SQL | Python | Scala | Java — Add-ON

Spark SQL + Data Frames

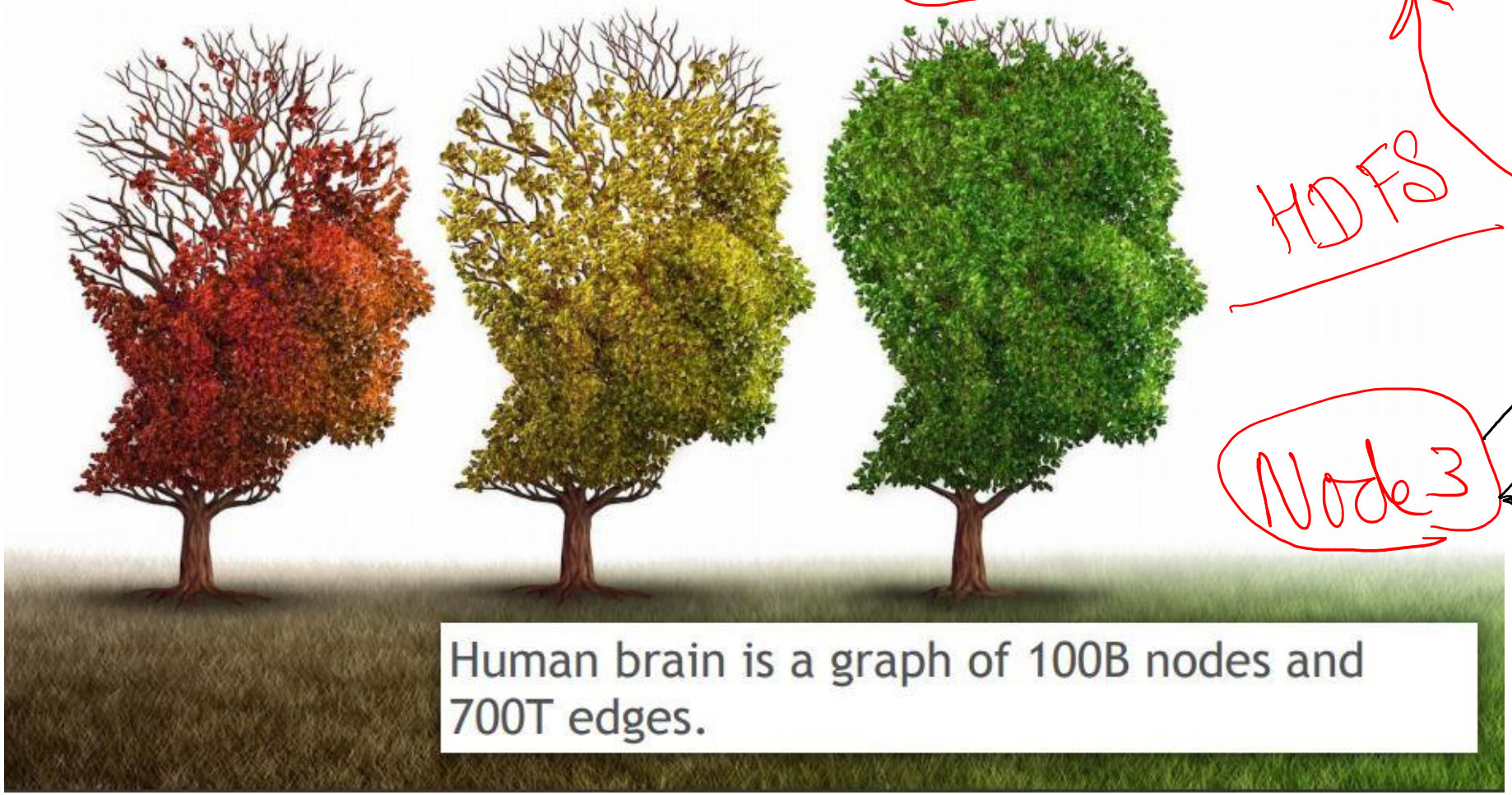
Streaming

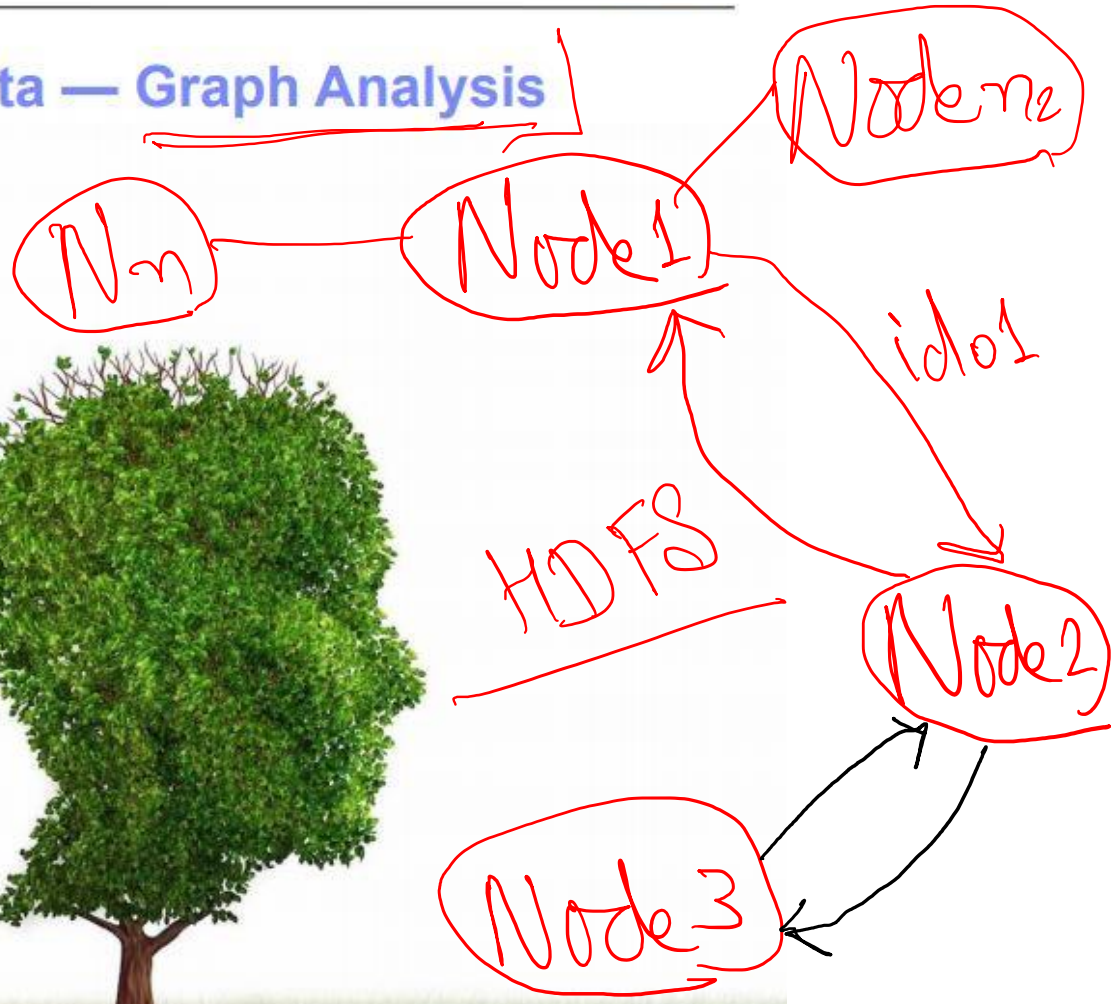
ML Lib.

graph API

Ecosystem - Apache Spark

Course Main Thrust 3: Linked Big Data — Graph Analysis



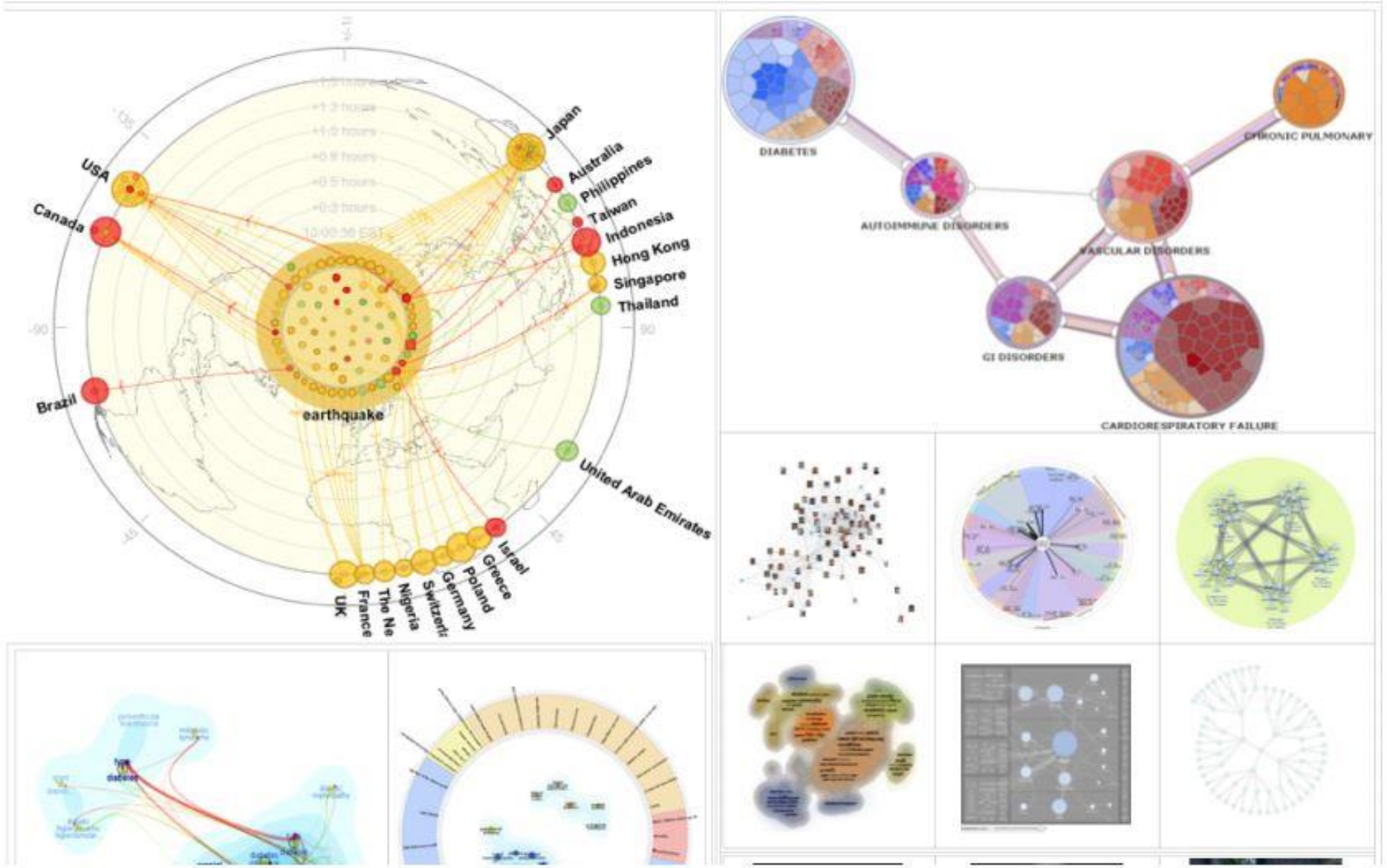


Course Main Thrust 4: Streaming Big Data Analytics

Data → DBMS → Hadoop

Process Data → Insights → DBMS → Reports

Course Main Thrust 5: Big Data Visualization



Course Main Thrust 6: Big Data System and AI Solutions

- Big Data Pipeline
- Big Data and AI for Finance
- Big Data and AI for Healthcare

Why you want to take this class — MooC / YouTube / edx / MiT

- **Key Differentiator of this class:** Focusing on building a full-spectrum understanding of the latest Big Data Analytics technologies and using them to build real industry real-world solutions.

6

- **Sapphire Big Data Analytics Open Source Applications:** Create a Big Data open source toolsets for various industries (and disciplines)

- **Dataset and Use Cases:** Welcome!! — DataSets

Big Data Analytics Platforms

Remind -- Apache Hadoop

— HDFS

The Apache™ Hadoop® project develops open-source software for reliable, scalable, distributed computing.

The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

The project includes these modules:

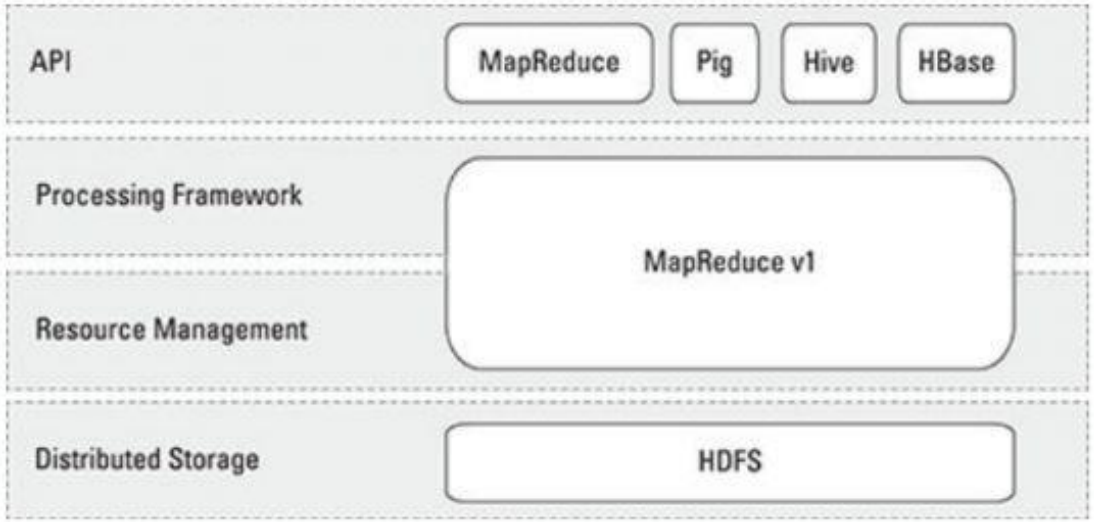
- **Hadoop Common**: The common utilities that support the other Hadoop modules.
- **Hadoop Distributed File System (HDFS™)**: A distributed file system that provides high-throughput access to application data.
- **Hadoop YARN**: A framework for job scheduling and cluster resource management.
- **Hadoop MapReduce**: A YARN-based system for parallel processing of large data sets.

http://hadoop.apache.org

Remind -- Hadoop-related Apache Projects

- Ambari™: A web-based tool for provisioning, managing, and monitoring Hadoop clusters.It also provides a dashboard for viewing cluster health and ability to view MapReduce, Pig and Hive applications visually.
- Avro™: A data serialization system.
- Cassandra™: A scalable multi-master database with no single points of failure.
- Chukwa™: A data collection system for managing large distributed systems.
- HBase™: A scalable, distributed database that supports structured data storage for large tables.
- Hive™: A data warehouse infrastructure that provides data summarization and ad hoc querying.
- Mahout™: A Scalable machine learning and data mining library.
- Pig™: A high-level data-flow language and execution framework for parallel computation.
- Spark™: A fast and general compute engine for Hadoop data. Spark provides a simple and expressive programming model that supports a wide range of applications, including ETL, machine learning, stream processing, and graph computation.
- Tez™: A generalized data-flow programming framework, built on Hadoop YARN, which provides a powerful and flexible engine to execute an arbitrary DAG of tasks to process data for both batch and interactive use-cases.
- ZooKeeper™: A high-performance coordination service for distributed applications.

Four distinctive layers of Hadoop



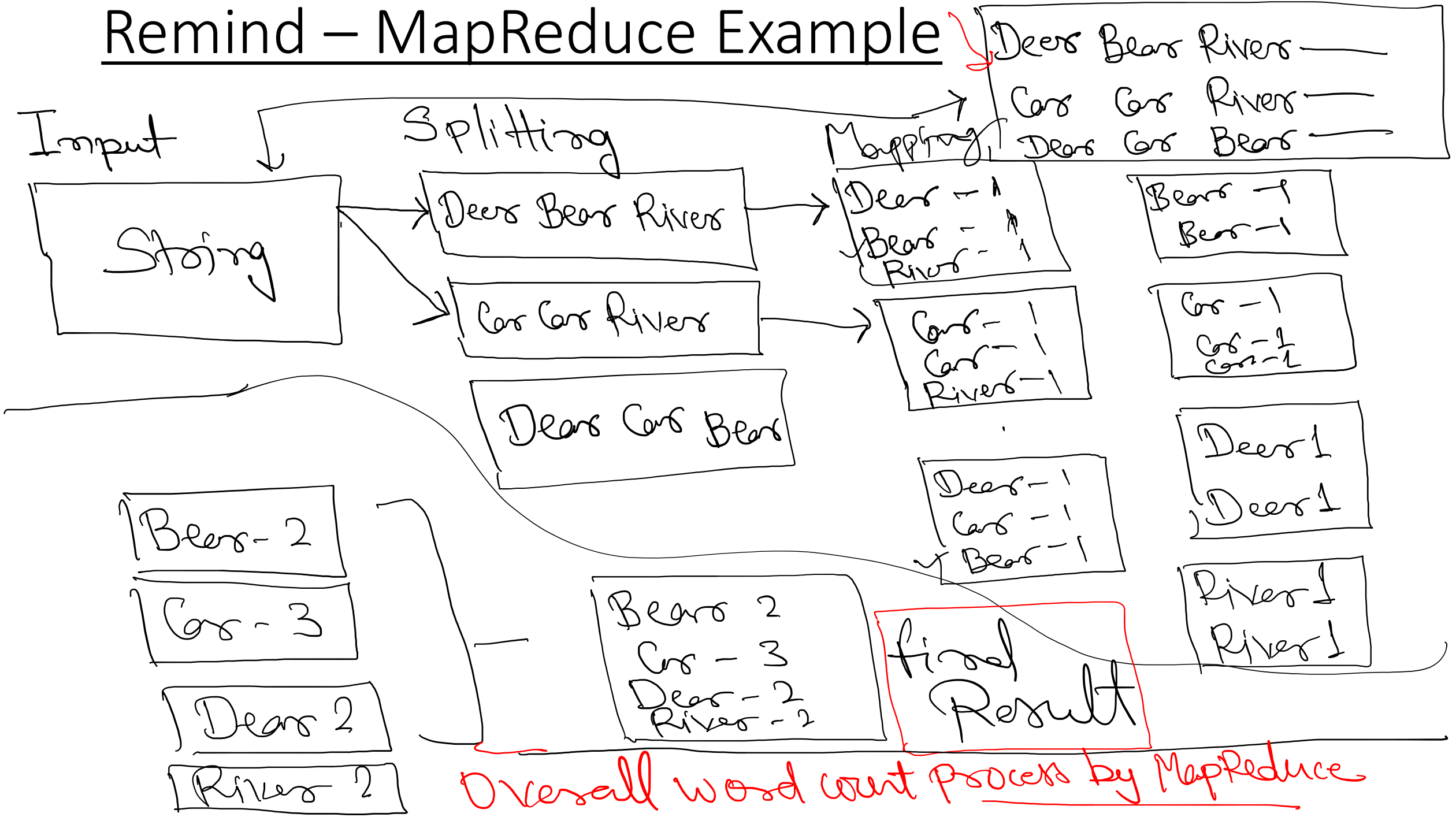
Distributed storage: The Hadoop Distributed File System (HDFS) is the storage layer where the data, interim results, and final result sets are stored.

Resource management: In addition to disk space, all slave nodes in the Hadoop cluster have CPU cycles, RAM, and network bandwidth. A system such as Hadoop needs to be able to parcel out these resources so that multiple applications and users can share the cluster in predictable and tunable ways. This job is done by the JobTracker daemon.

Processing framework: The MapReduce process flow defines the execution of all applications in Hadoop 1. As we saw in Chapter 6, this begins with the map phase; continues with aggregation with shuffle, sort, or merge; and ends with the reduce phase. In Hadoop 1, this is also managed by the JobTracker daemon, with local execution being managed by TaskTracker daemons running on the slave nodes.

Application Programming Interface (API): Applications developed for Hadoop 1 needed to be coded using the MapReduce API. In Hadoop 1, the Hive and Pig projects provide programmers with easier interfaces for writing Hadoop applications, and underneath the hood, their code compiles down to MapReduce.

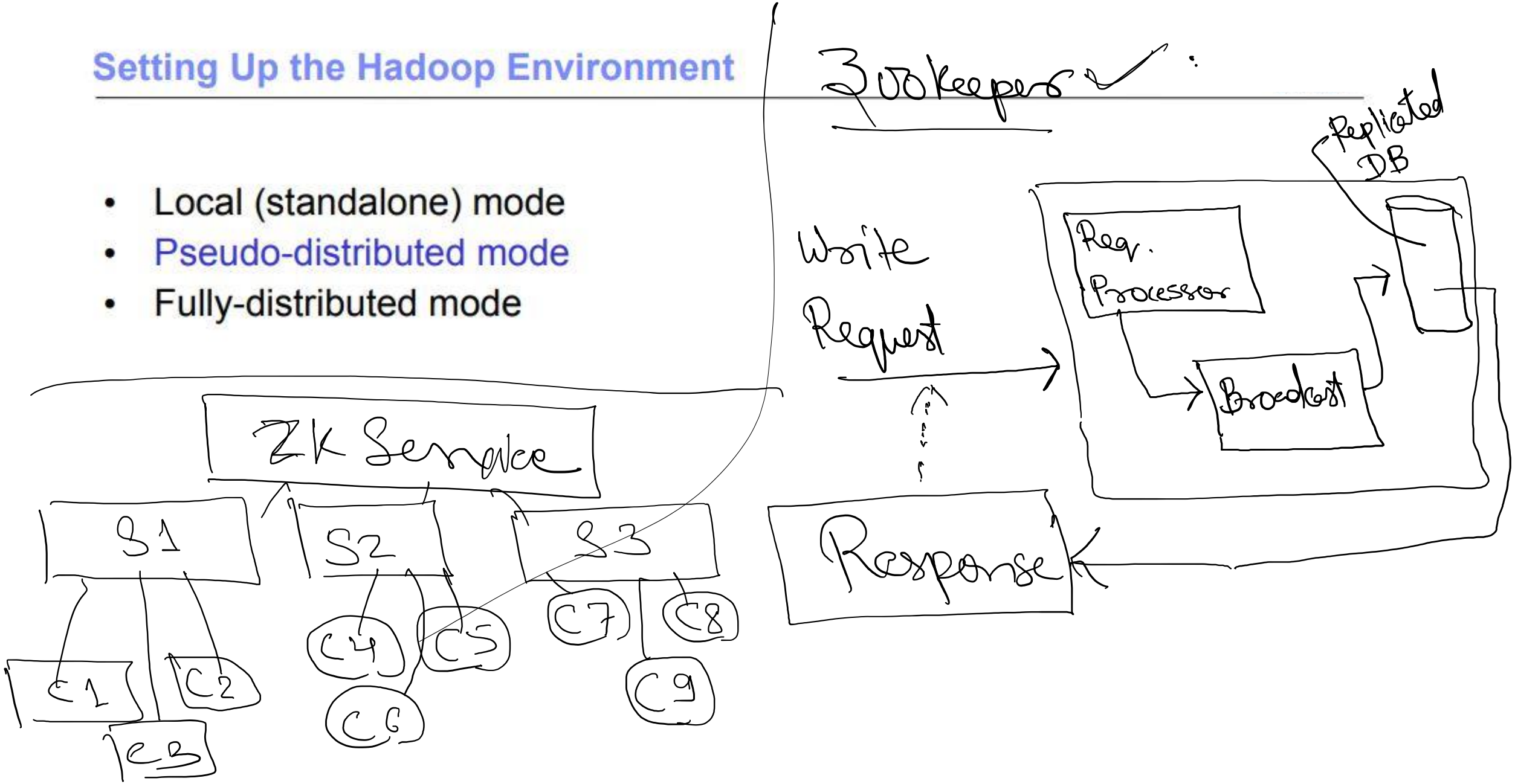
Remind – MapReduce Example



Overall word count process by MapReduce

Setting Up the Hadoop Environment

- Local (standalone) mode
- Pseudo-distributed mode
- Fully-distributed mode



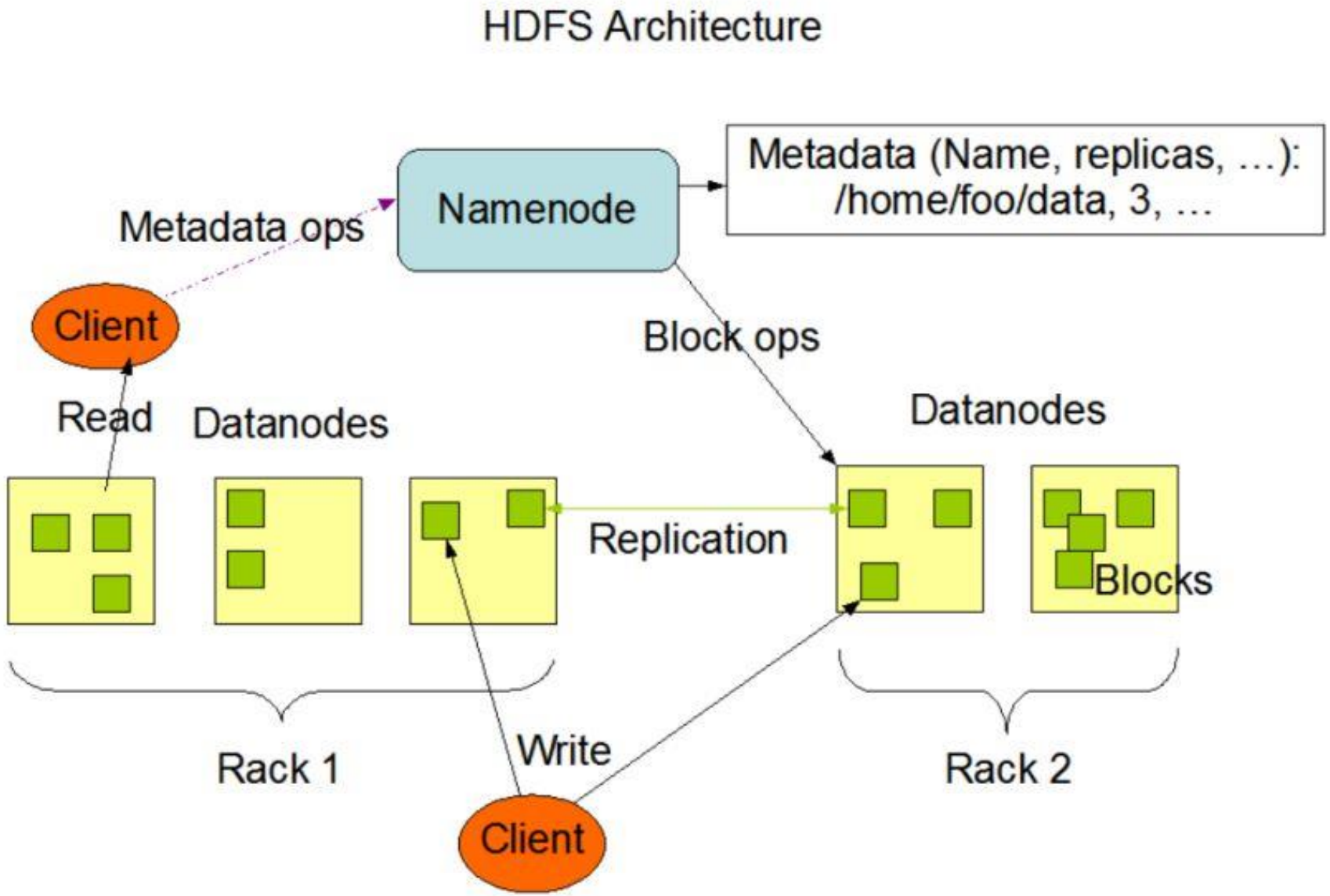
Data Storage Operations on HDFS

- Hadoop is designed to work best with a modest number of extremely large files.
- Average file sizes ➔ larger than 500MB.

- Write Once, Read Often model.
- Content of individual files cannot be modified, other than appending new data at the end of the file.

- What we can do:
  - Create a new file
  - Append content to the end of a file
  - Delete a file
  - Rename a file
  - Modify file attributes like owner

Remind -- Hadoop Distributed File System (HDFS)



http://hortonworks.com/hadoop/hdfs/