Big Data, Hadoop, Map-Reduce

Atılım Üniversity

Department of Computer Engineering

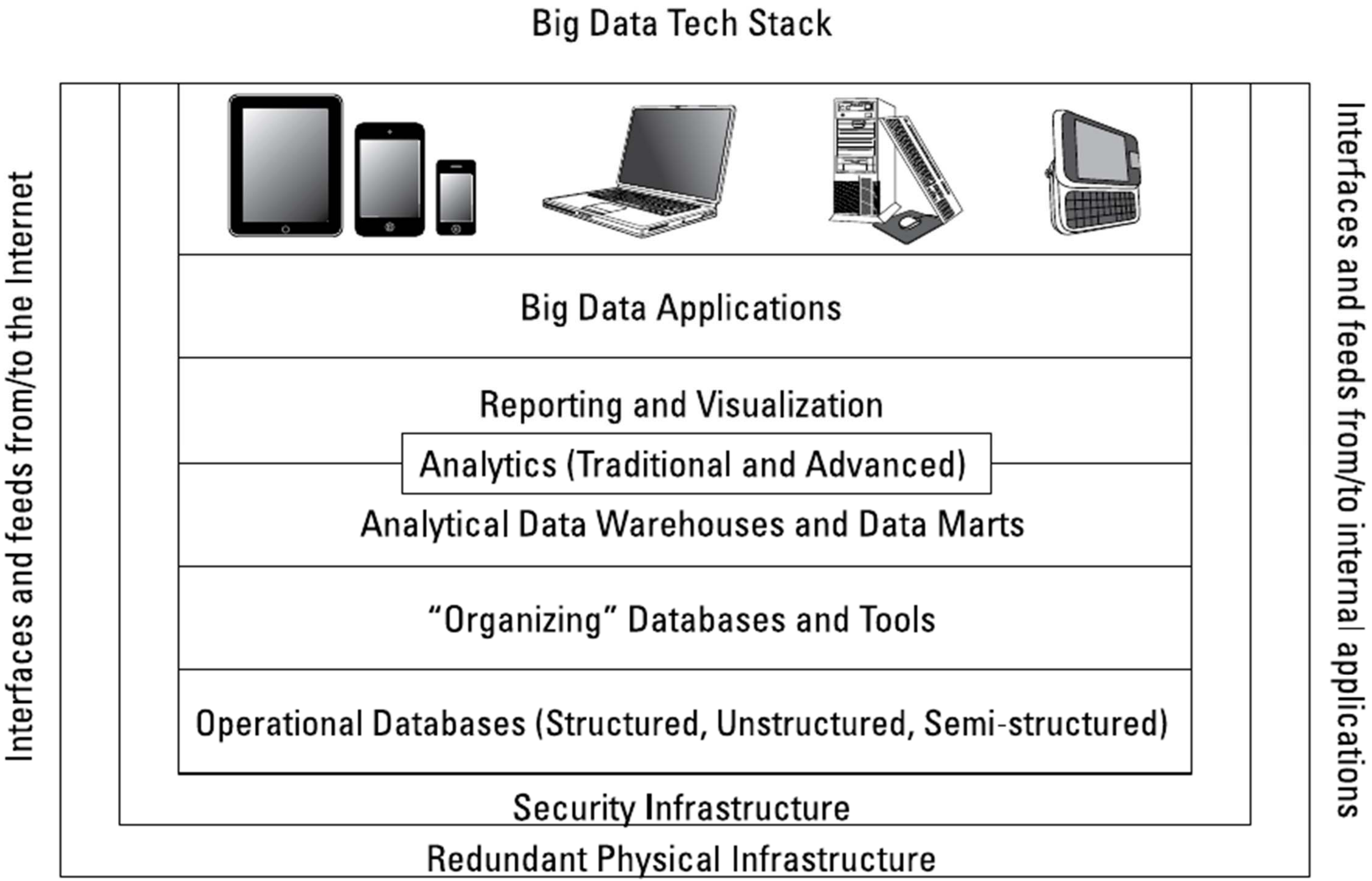
Asst. Prof. Dr. Ziya Karakaya

December 14, 2016

Contents

- Big Data Technology Stack
- Hadoop definition and ecosystem
- HDFS (Hadoop Distributed File System)
- YARN (Yet Another Resource Manager)
- Map-Reduce Programming Model
- Streaming Data Processing
- Complementary Technologies
- Three Trend Topics

Big Data Technology Stack



Big Data Tech Stack

Big Data Applications

Reporting and Visualization

Analytics (Traditional and Advanced)

Analytical Data Warehouses and Data Marts

“Organizing” Databases and Tools

Operational Databases (Structured, Unstructured, Semi-structured)

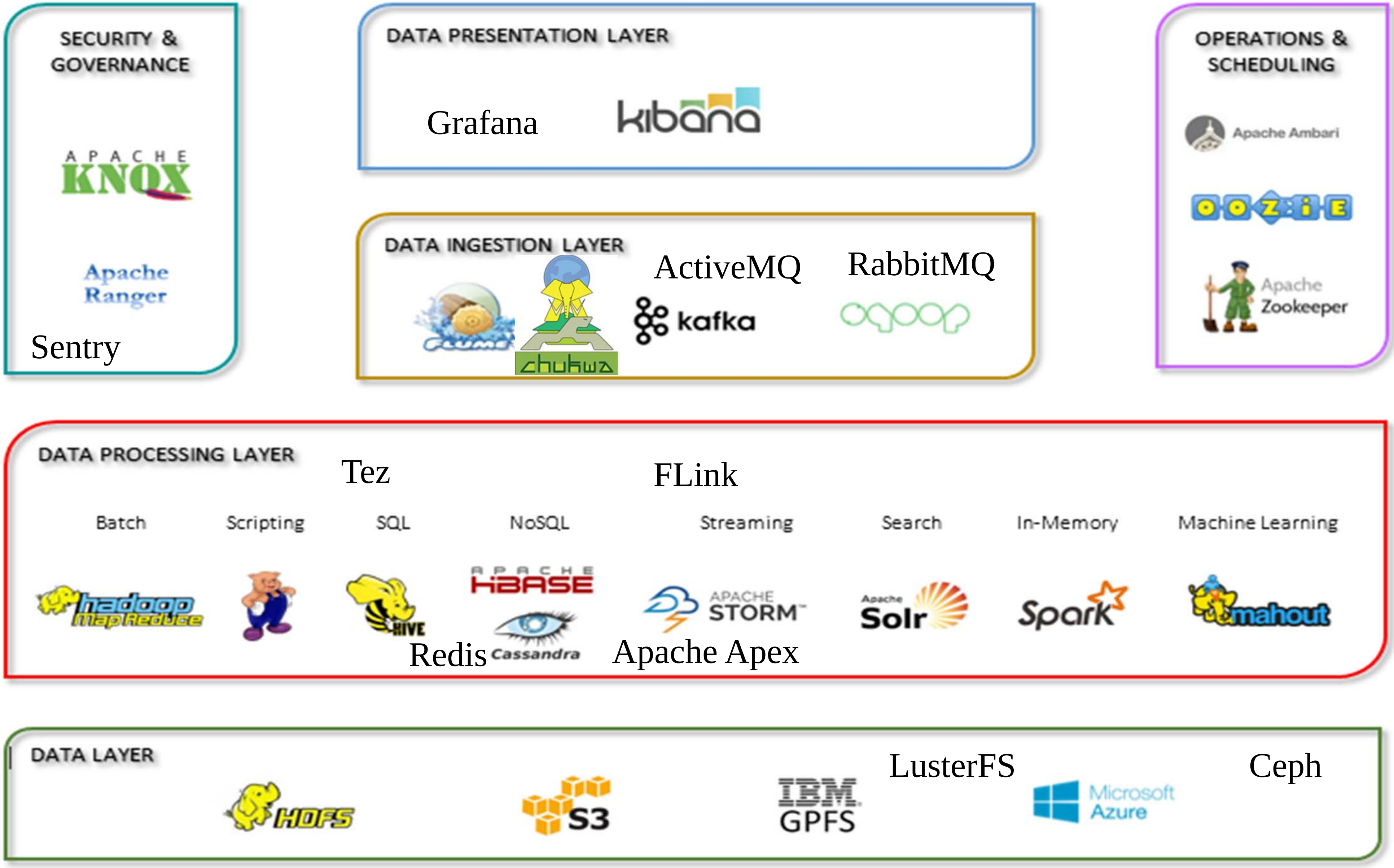
Security Infrastructure

Redundant Physical Infrastructure

Interfaces and feeds from/to the Internet

Interfaces and feeds from/to internal applications

Big Data Technology Stack



SECURITY & GOVERNANCE

Sentry

DATA PRESENTATION LAYER

Grafana

DATA INGESTION LAYER

ActiveMQ

RabbitMQ

OPERATIONS & SCHEDULING

DATA PROCESSING LAYER

Tez

FLink

Batch

Scripting

SQL

NoSQL

Streaming

Search

In-Memory

Machine Learning

Redis

Apache Apex

DATA LAYER

LusterFS

Ceph

Big Data Technology Stack

http://edgeoftesting.com

Definition of Hadoop

Open sourced, flexible and **high-avail**

What Is Apache Hadoop?

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 laye**r, 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:

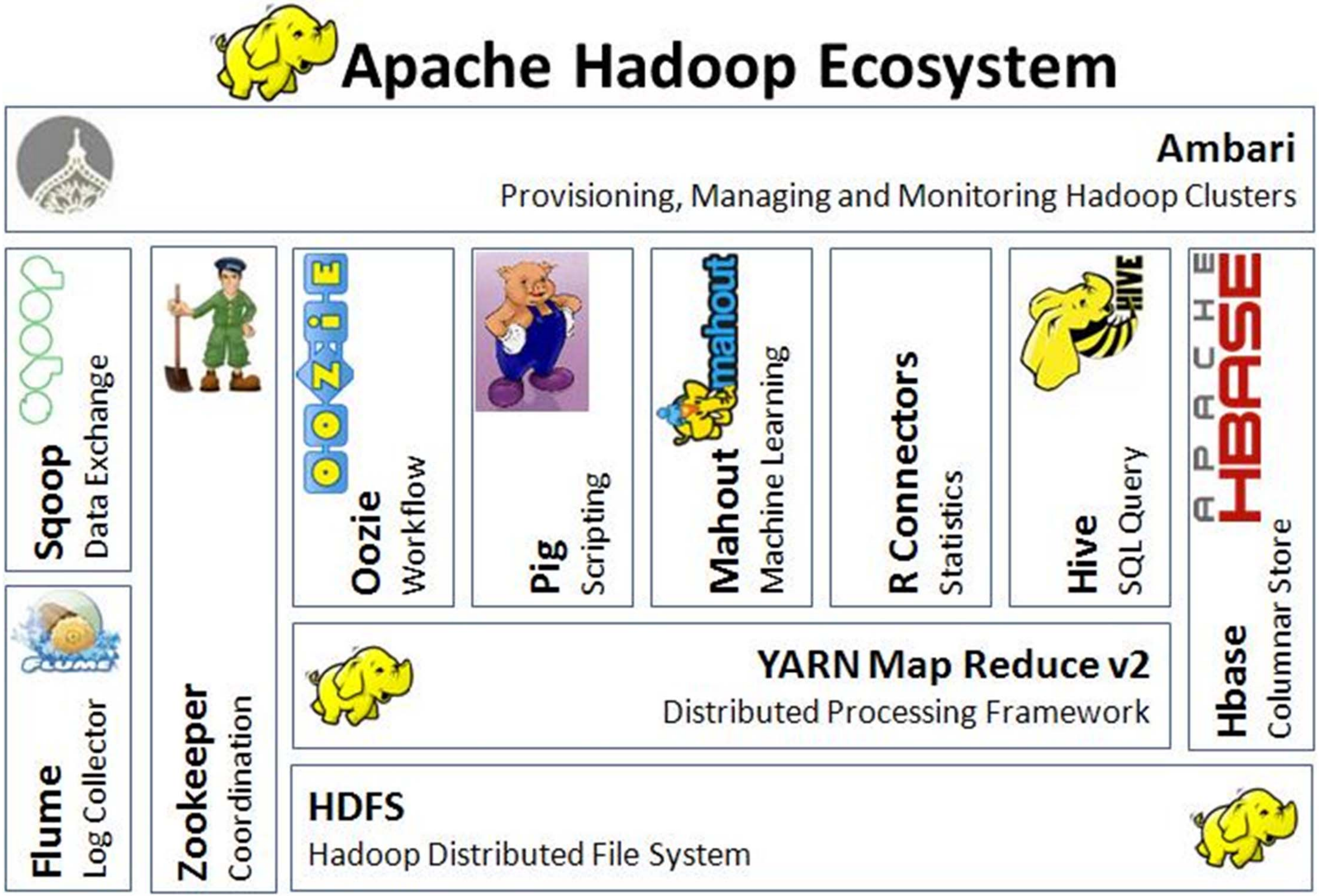
Hadoop Common: The common utilities that support the other Hadoop modules.

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.

Ecosystem...



Apache Hadoop Ecosystem

Ambari
Provisioning, Managing and Monitoring Hadoop Clusters

Sqoop
Data Exchange

Flume
Log Collector

Zookeeper
Coordination

Oozie
Workflow

Pig
Scripting

Mahout
Machine Learning

R Connectors
Statistics

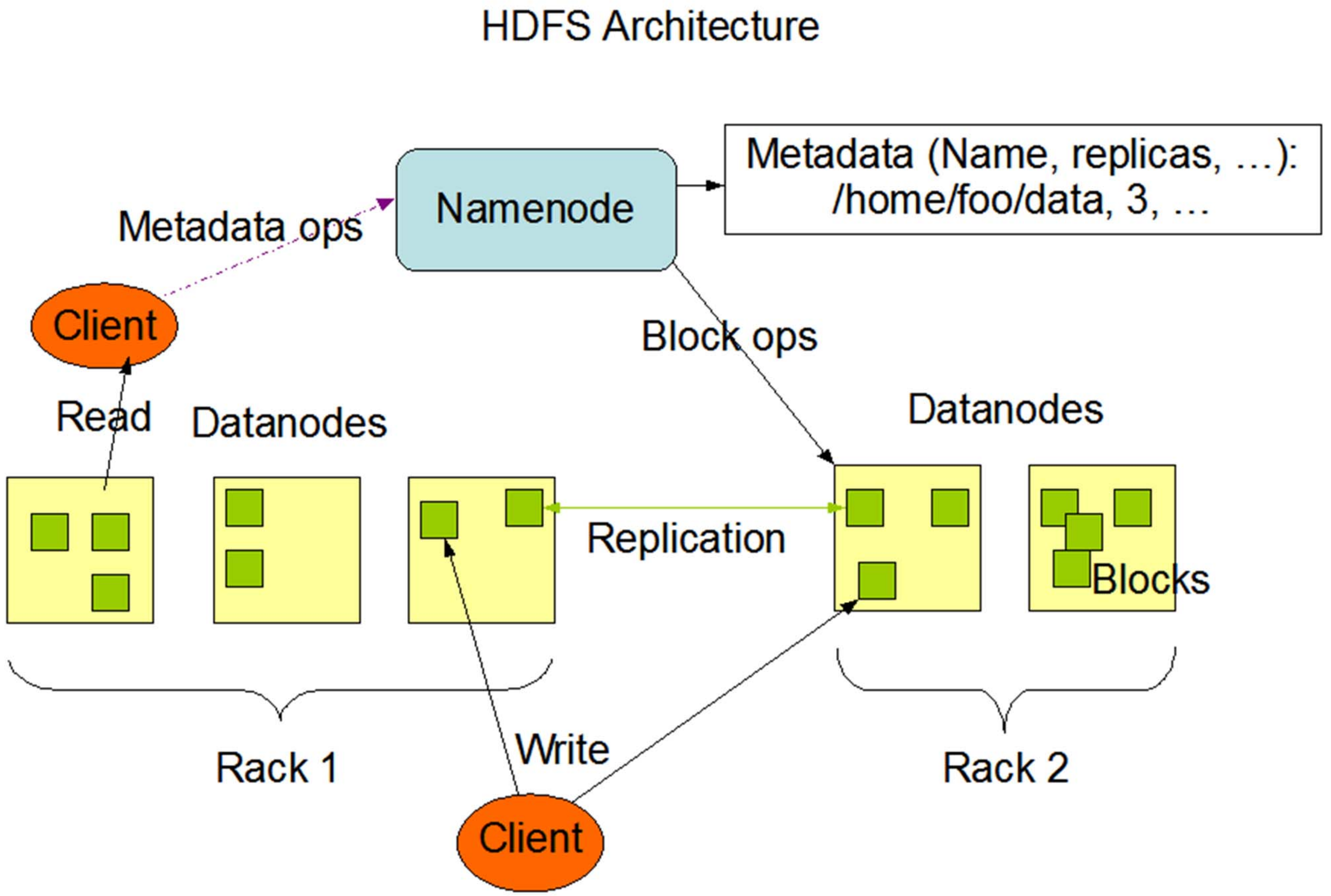
Hive
SQL Query

Hbase
Columnar Store

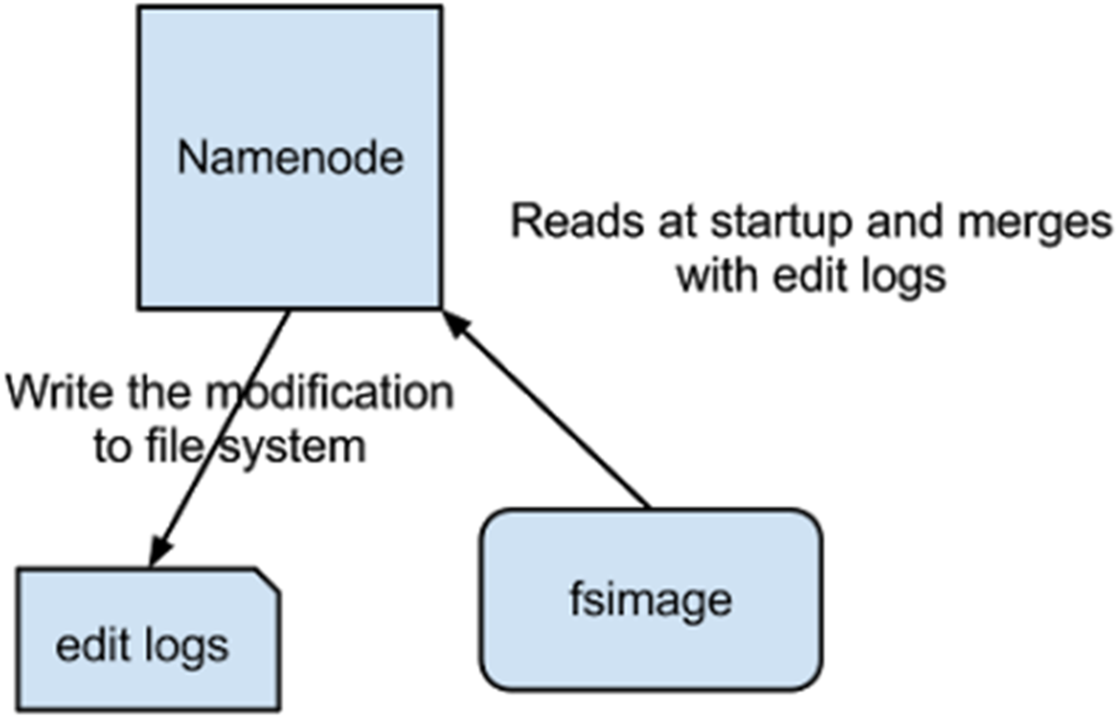
YARN Map Reduce v2
Distributed Processing Framework

HDFS
Hadoop Distributed File System

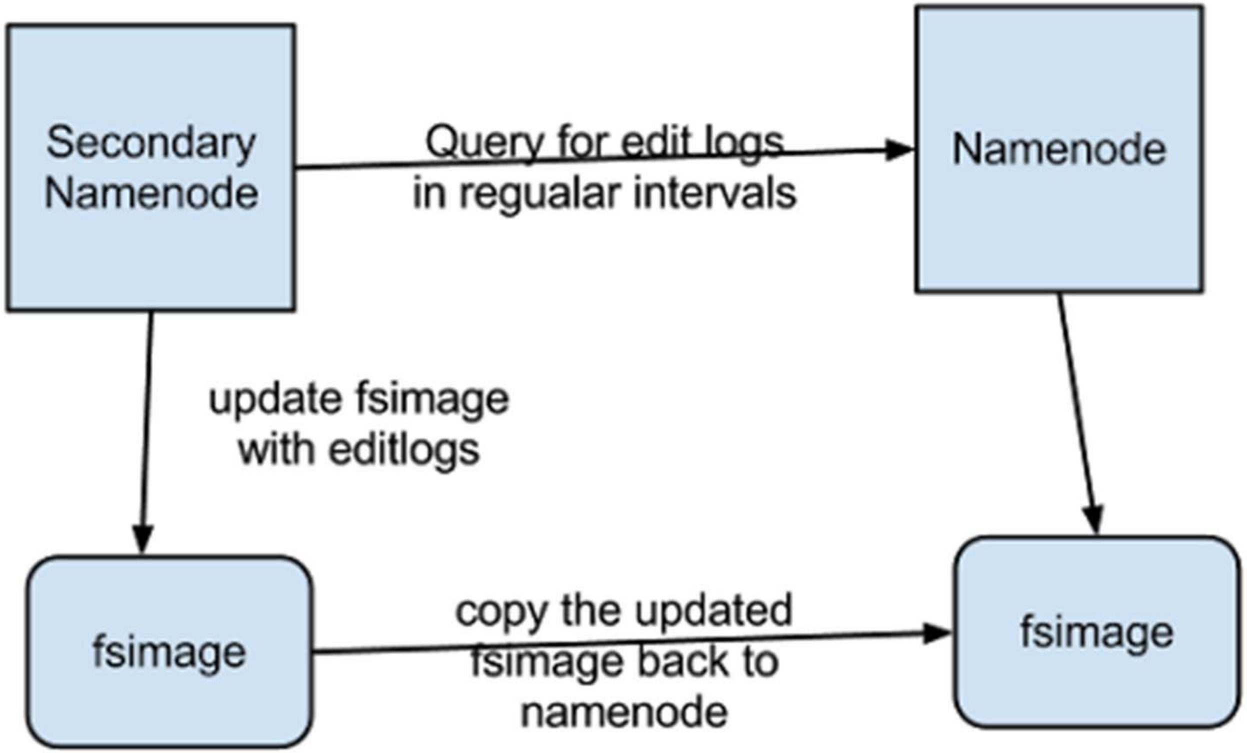
HDFS Architecture (SPOF ???)



NameNode & Secondary NameNode


Namenode
Reads at startup and merges with edit logs
Write the modification to file system
edit logs
fsimage

Secondary NameNode


Secondary Namenode
Query for edit logs in regualar intervals
Namenode
update fsimage with editlogs
fsimage
copy the updated fsimage back to namenode
fsimage

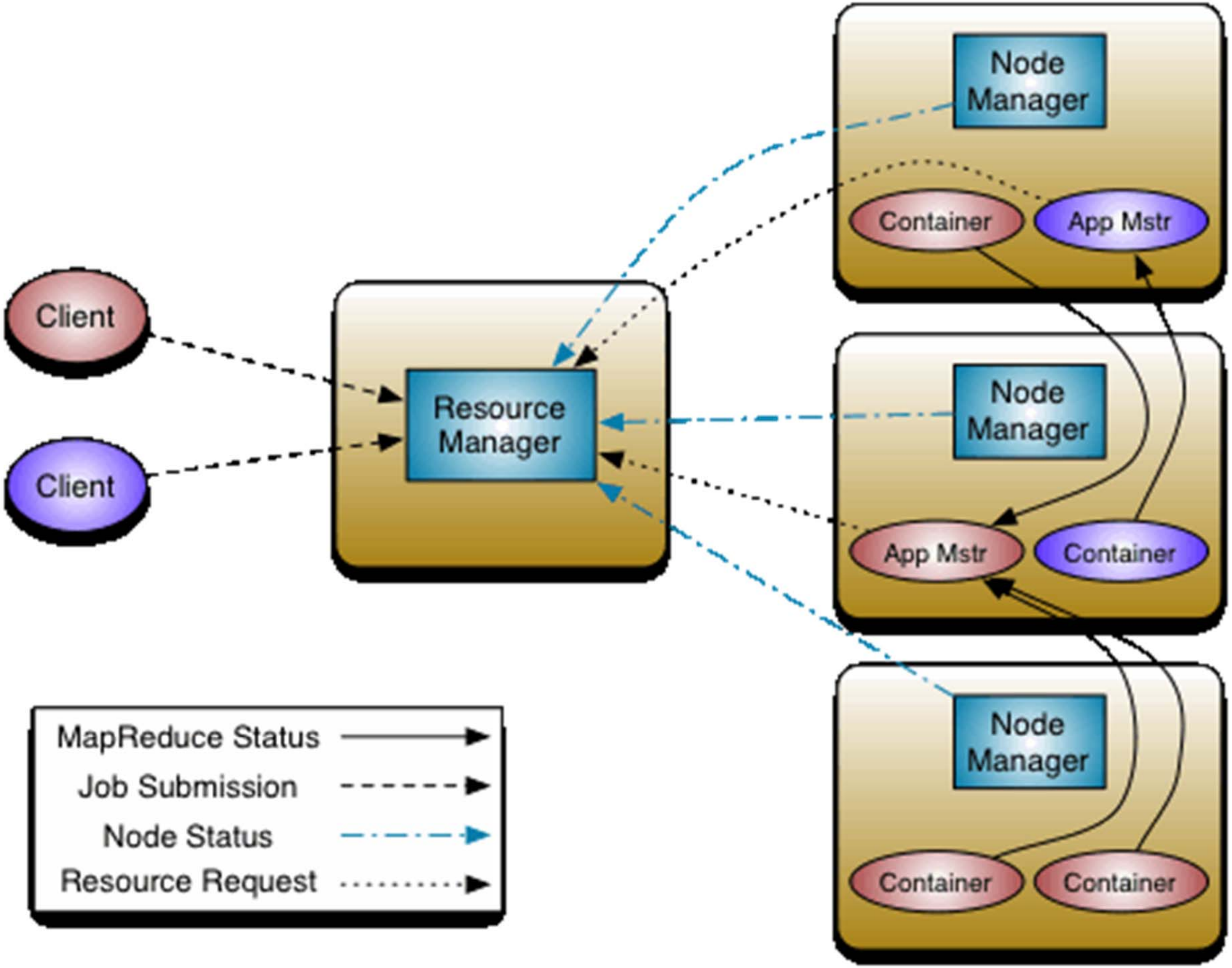
YARN (Yet Another Resource Manager)

The fundamental idea of YARN is to split up the functionalities of resource management and job scheduling/monitoring into separate daemons. The idea is to have a global ResourceManager (RM) and per-application ApplicationMaster (AM). An application is either a single job or a DAG of jobs.

The ResourceManager and the NodeManager form the data-computation framework. The ResourceManager is the ultimate authority that arbitrates resources among all the applications in the system. The NodeManager is the per-machine framework agent who is responsible for containers, monitoring their resource usage (cpu, memory, disk, network) and reporting the same to the ResourceManager/Scheduler.

The per-application ApplicationMaster is, in effect, a

How YARN Works



3 mapper, 1 reducer

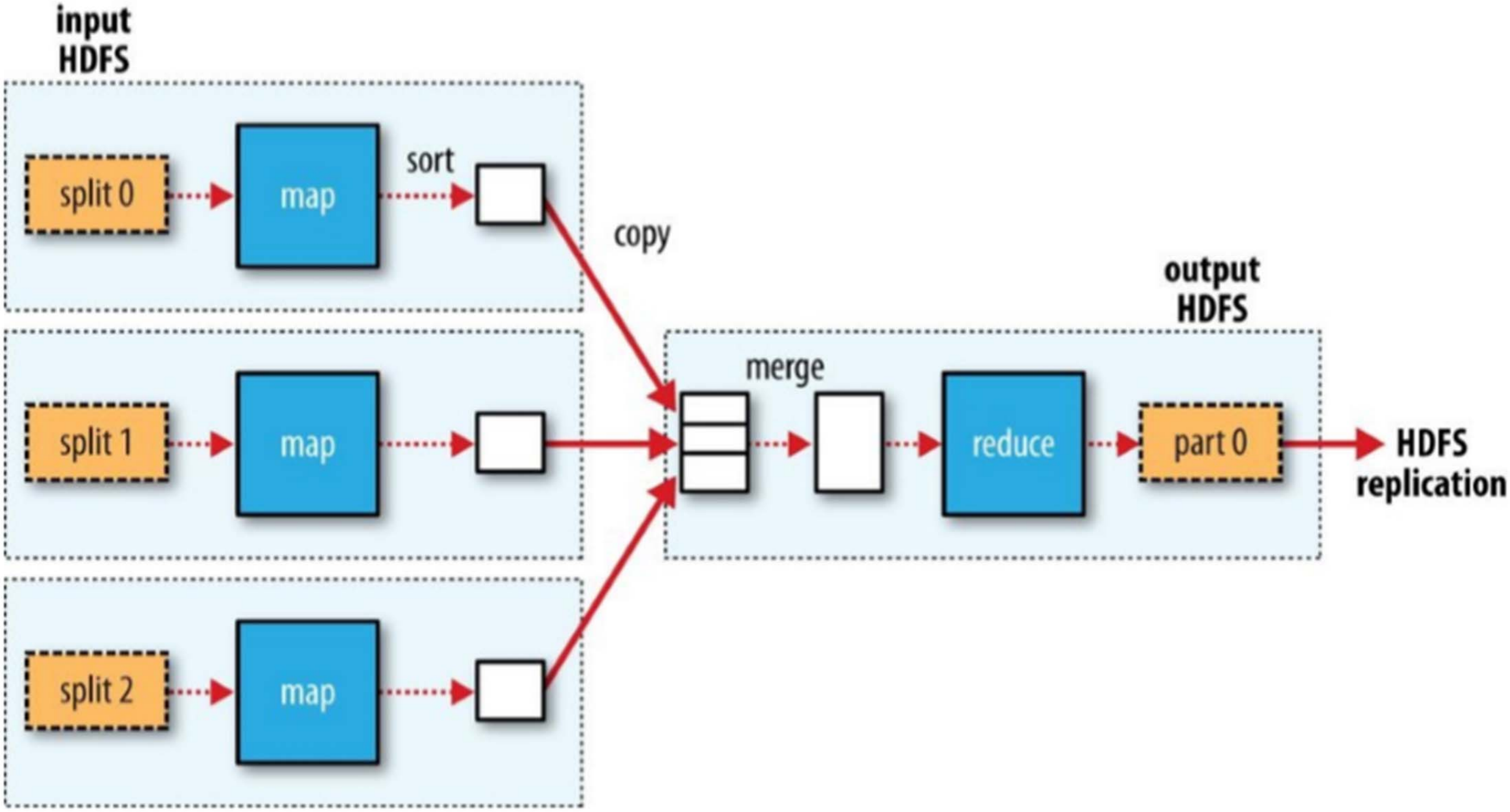


Figure 2-3. MapReduce data flow with a single reduce task

3 mapper, 2 reducer

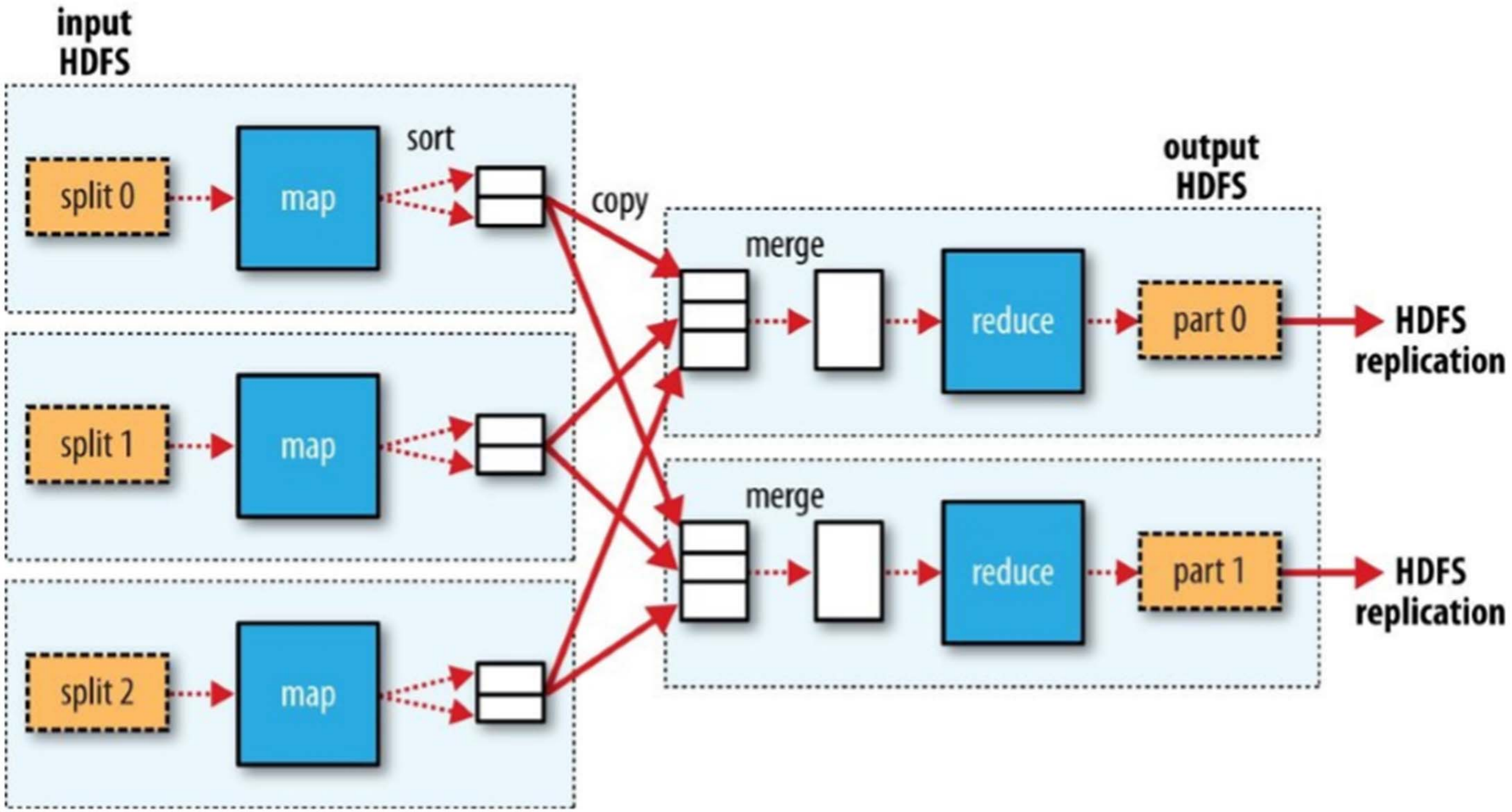
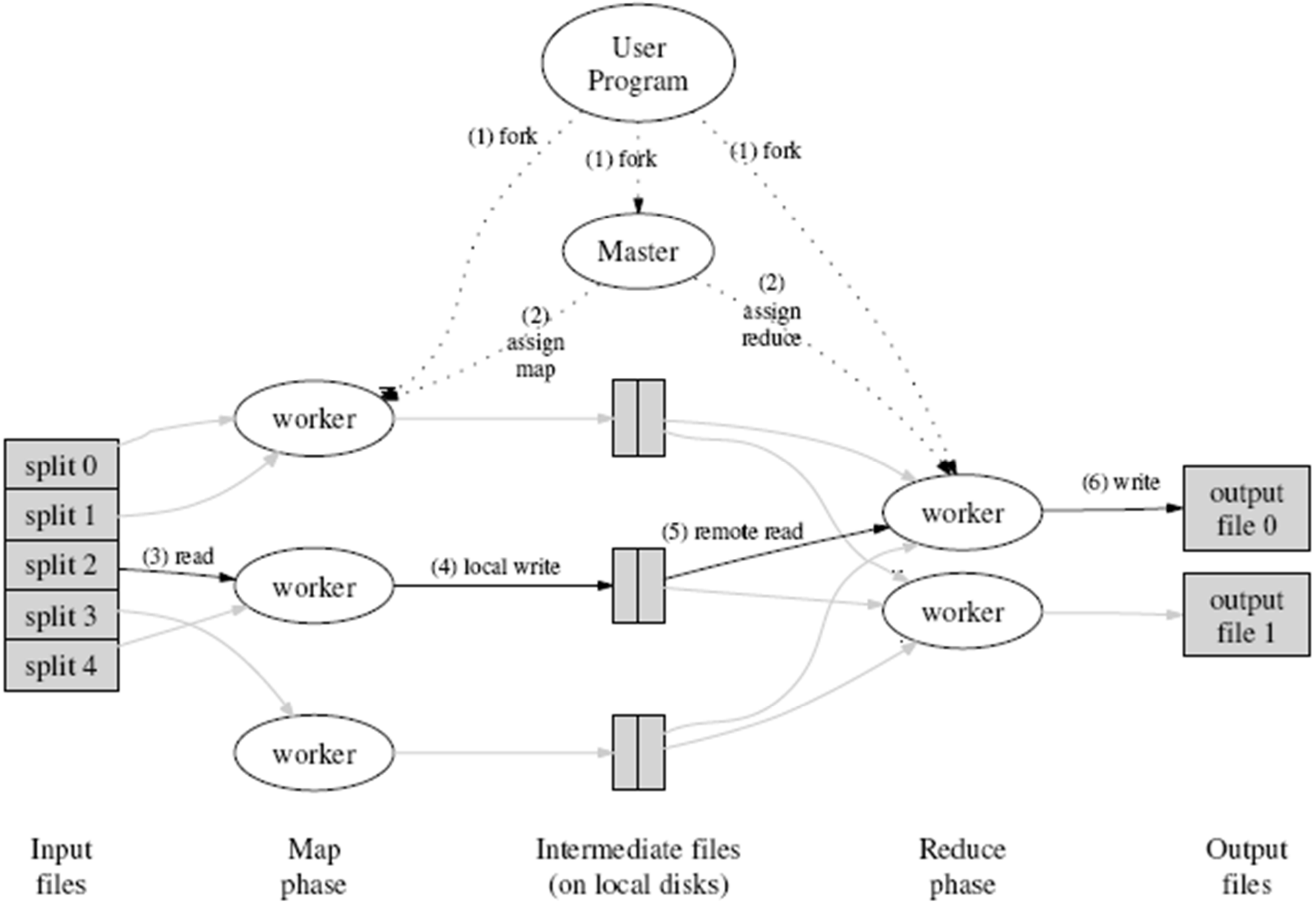
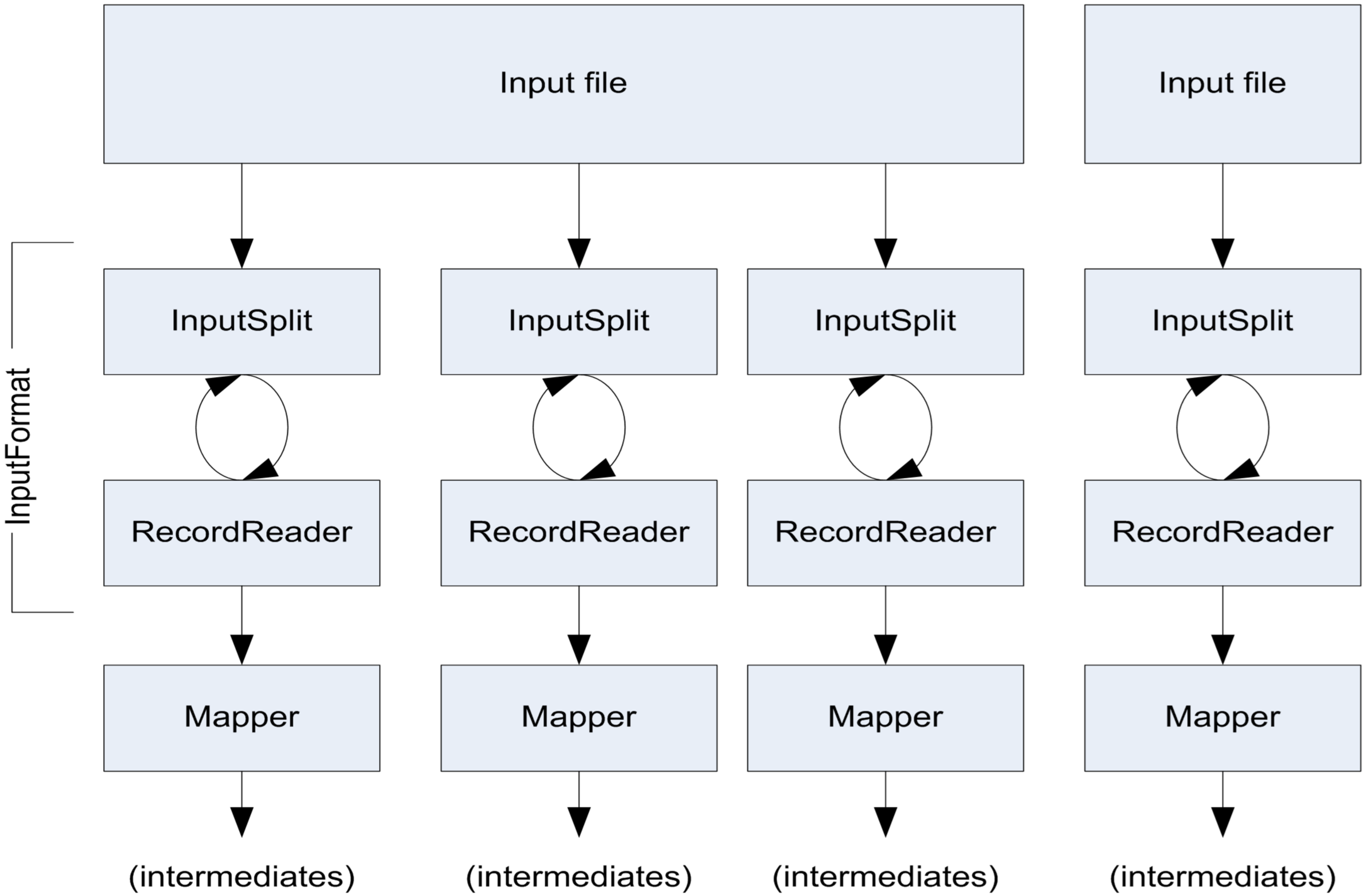


Figure 2-4. MapReduce data flow with multiple reduce tasks

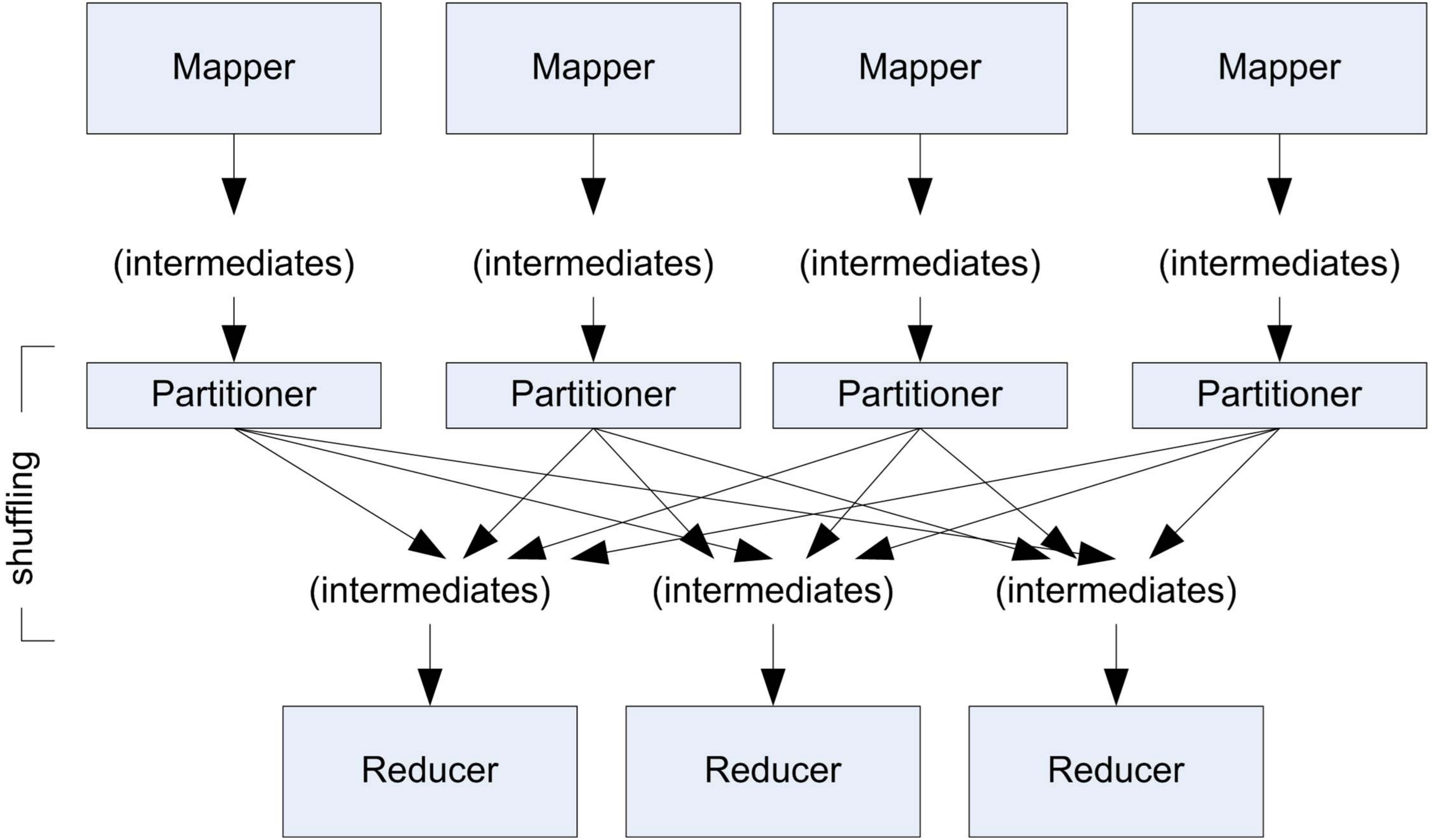
A bit details



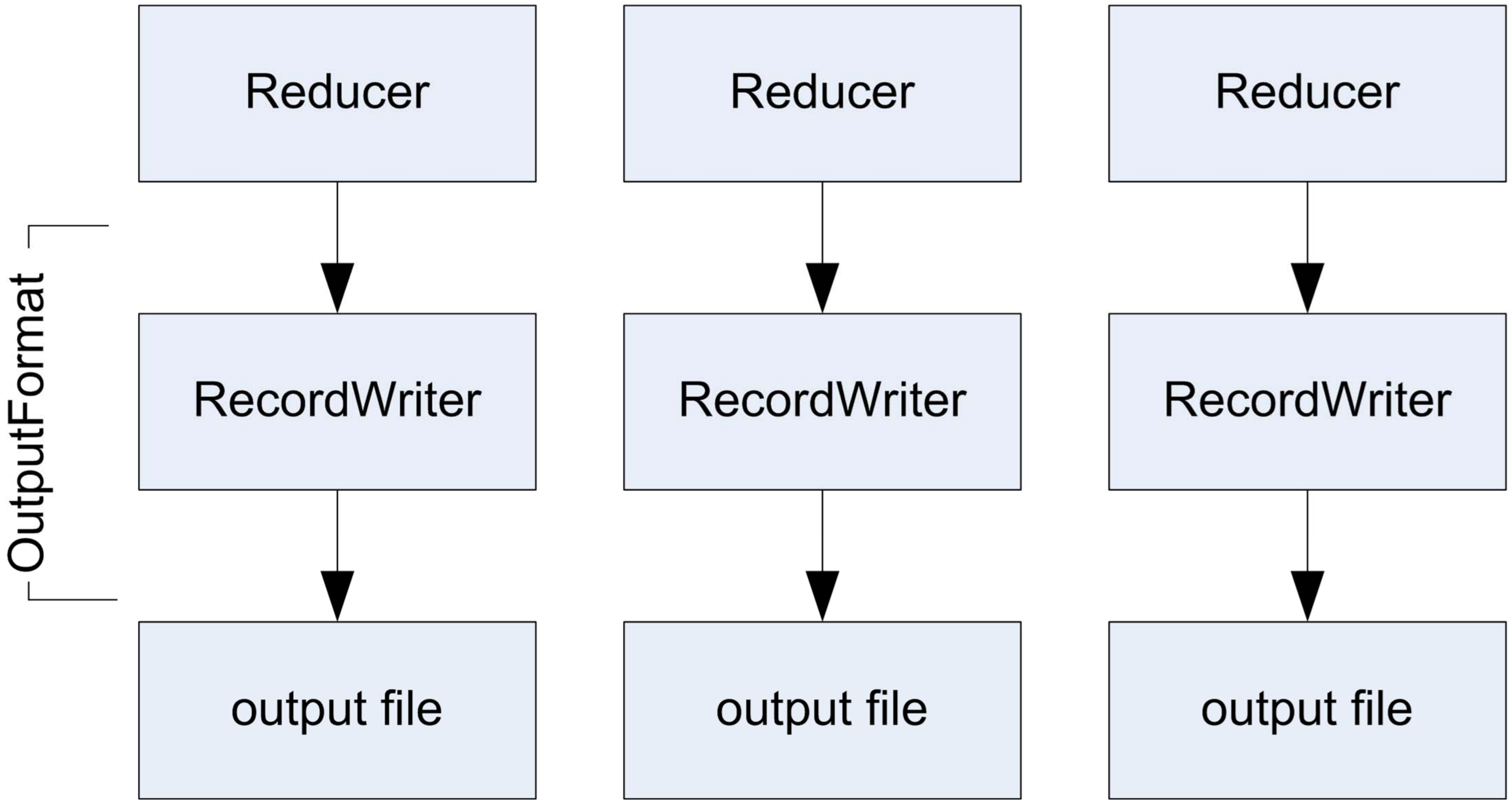
More detailed



....


Mapper
Mapper
Mapper
Mapper
(intermediates)
(intermediates)
(intermediates)
(intermediates)
Partitioner
Partitioner
Partitioner
Partitioner
shuffling
(intermediates)
(intermediates)
(intermediates)
Reducer
Reducer
Reducer

....


OutputFormat
Reducer
Reducer
Reducer
RecordWriter
RecordWriter
RecordWriter
output file
output file
output file

References

- http://hadoop.apache.org/
- http://edgeoftesting.com/
- http://cassandra.apache.org/
- http://chukwa.apache.org/docs/r0.5.0/
- http://hbase.apache.org/
- http://hive.apache.org/
- http://pig.apache.org/
- http://spark.apache.org/
- https://apex.apache.org/
- http://www.alluxio.org/
- http://zookeeper.apache.org/
- http://insidebigdata.com/2015/12/08/big-data-industry-predictions-2016/
- https://en.wikipedia.org/wiki/Big_data
- http://tracker.ceph.com/

Questions ?

Thank you !