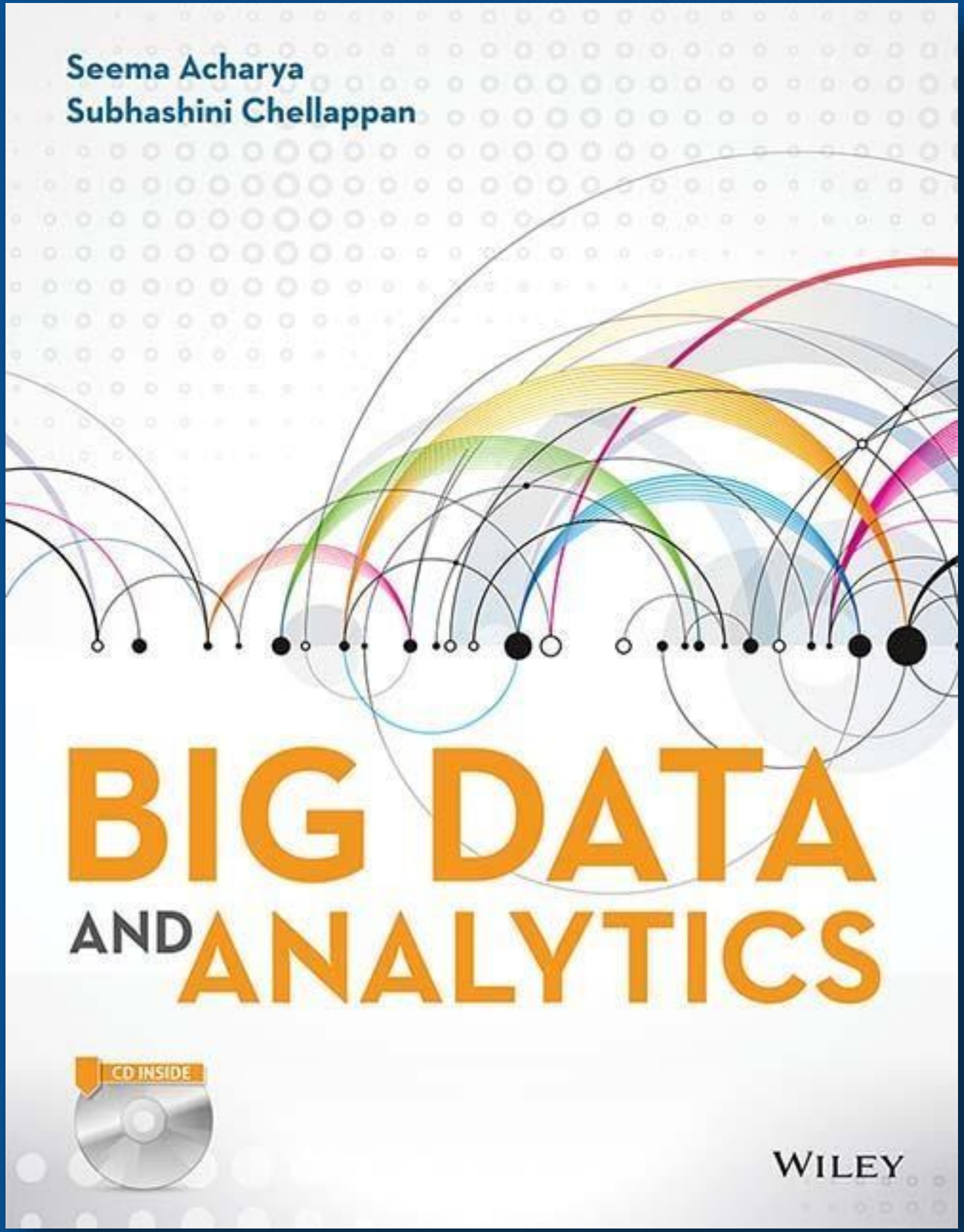


Big Data And Analytics

Seema Acharya
Subhashini Chellappan

Big Data and Analytics by Seema Acharya and Subhashini Chellappan

Chapter 5

Introduction to Hadoop

Learning Objectives and Learning Outcomes

Learning Objectives	Learning Outcomes
Introduction to Hadoop	
1. To study the features of Hadoop.	a) To comprehend the reasons behind the popularity of Hadoop.
2. To learn the basic concepts of HDFS and MapReduce Programming.	b) To be able to perform HDFS operations.
3. To study HDFS Architecture.	c) To comprehend MapReduce framework.
4. To study MapReduce Programming Model	d) To understand the read and write in HDFS.
5. To study Hadoop Ecosystem.	e) To be able to understand Hadoop Ecosystem.

Session Plan

Lecture time	120 to 150 minutes
Q/A	15 minutes

Agenda

- Hadoop - An Introduction
- RDBMS versus Hadoop
- Distributed Computing Challenges
- History of Hadoop
- Hadoop Overview
  - Key Aspects of Hadoop
  - Hadoop Components
  - High Level Architecture of Hadoop
- Use case for Hadoop
  - ClickStream Data
- Hadoop Distributors
- HDFS
  - HDFS Daemons
  - Anatomy of File Read
  - Anatomy of File Write
  - Replica Placement Strategy
  - Working with HDFS commands
  - Special Features of HDFS

Agenda

- Processing Data with Hadoop
  - What is MapReduce Programming?
  - How does MapReduce Works?
  - MapReduce Word Count Example

- Managing Resources and Application with Hadoop YARN
  - Limitations of Hadoop 1.0 Architecture
  - Hadoop 2 YARN: Taking Hadoop Beyond Batch

- Hadoop Ecosystem
  - Pig
  - Hive
  - Sqoop
  - HBase

Hadoop – An Introduction

What is Hadoop

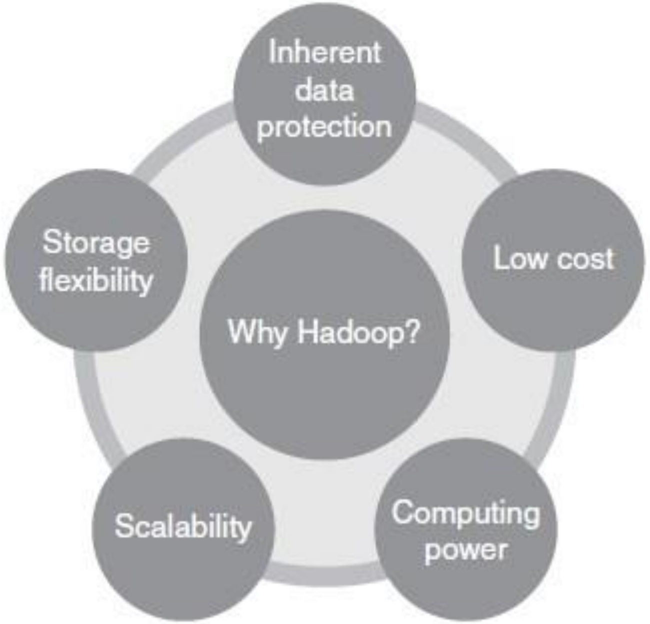
Hadoop is:

Ever wondered why Hadoop has been and is one of the most wanted technologies!!

The key consideration (the rationale behind its huge popularity) is:

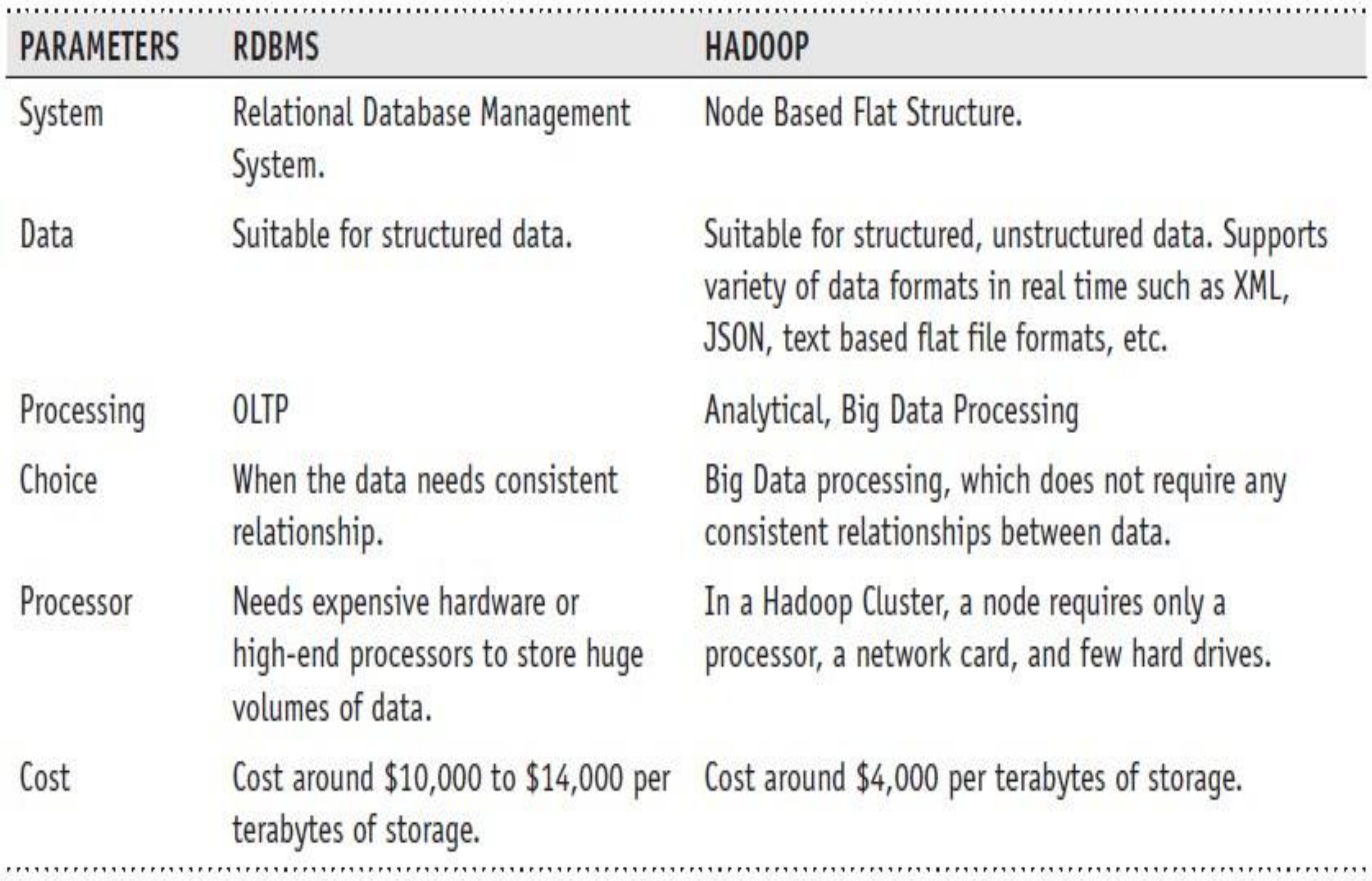
Its capability to handle massive am
categories of data – fairly quickly.

The other considerations are :



RDBMS versus HADOOP

RDBMS versus HADOOP



PARAMETERS	RDBMS	HADOOP
System	Relational Database Management System.	Node Based Flat Structure.
Data	Suitable for structured data.	Suitable for structured, unstructured data. Supports variety of data formats in real time such as XML, JSON, text based flat file formats, etc.
Processing	OLTP	Analytical, Big Data Processing
Choice	When the data needs consistent relationship.	Big Data processing, which does not require any consistent relationships between data.
Processor	Needs expensive hardware or high-end processors to store huge volumes of data.	In a Hadoop Cluster, a node requires only a processor, a network card, and few hard drives.
Cost	Cost around $10,000 to $14,000 per terabytes of storage.	Cost around $4,000 per terabytes of storage.

Distributed Computing Challenges

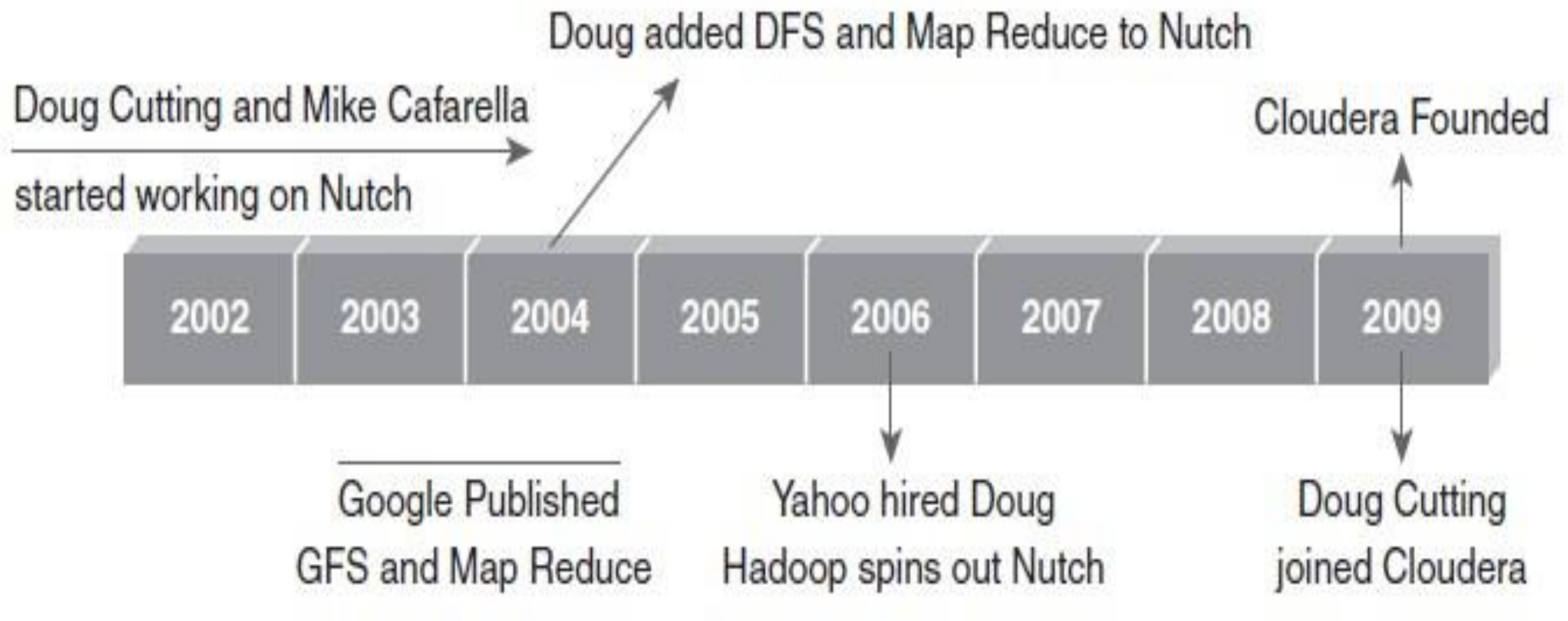
Distributed Computing Challenges

- Hardware Failure

- How to Process This Gigantic Store of Data?

History of Hadoop

History of Hadoop



Doug Cutting and Mike Cafarella started working on Nutch

Doug added DFS and Map Reduce to Nutch

Cloudera Founded

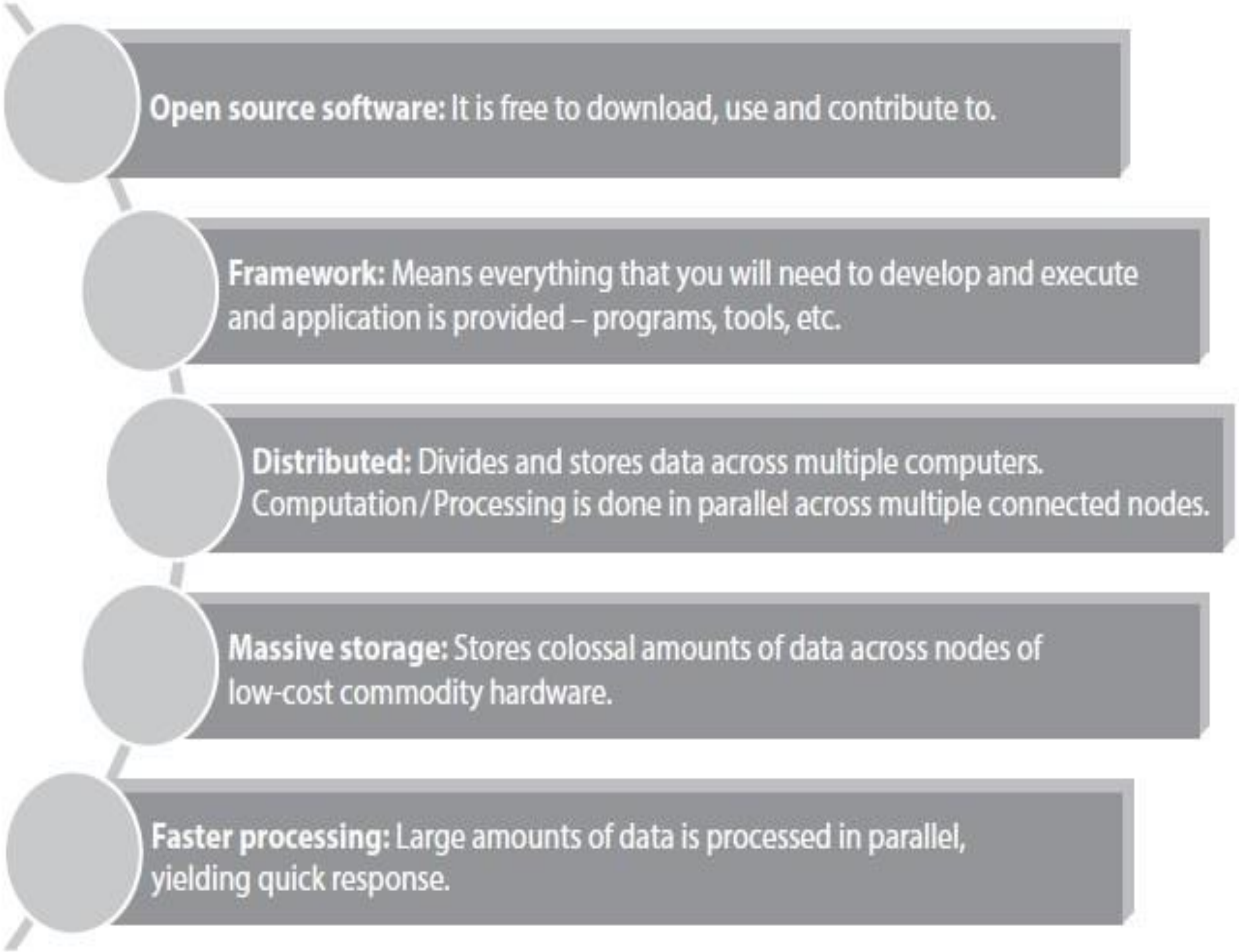
Google Published GFS and Map Reduce

Yahoo hired Doug Hadoop spins out Nutch

Doug Cutting joined Cloudera

Hadoop Overview

Key Aspects of Hadoop



Open source software: It is free to download, use and contribute to.

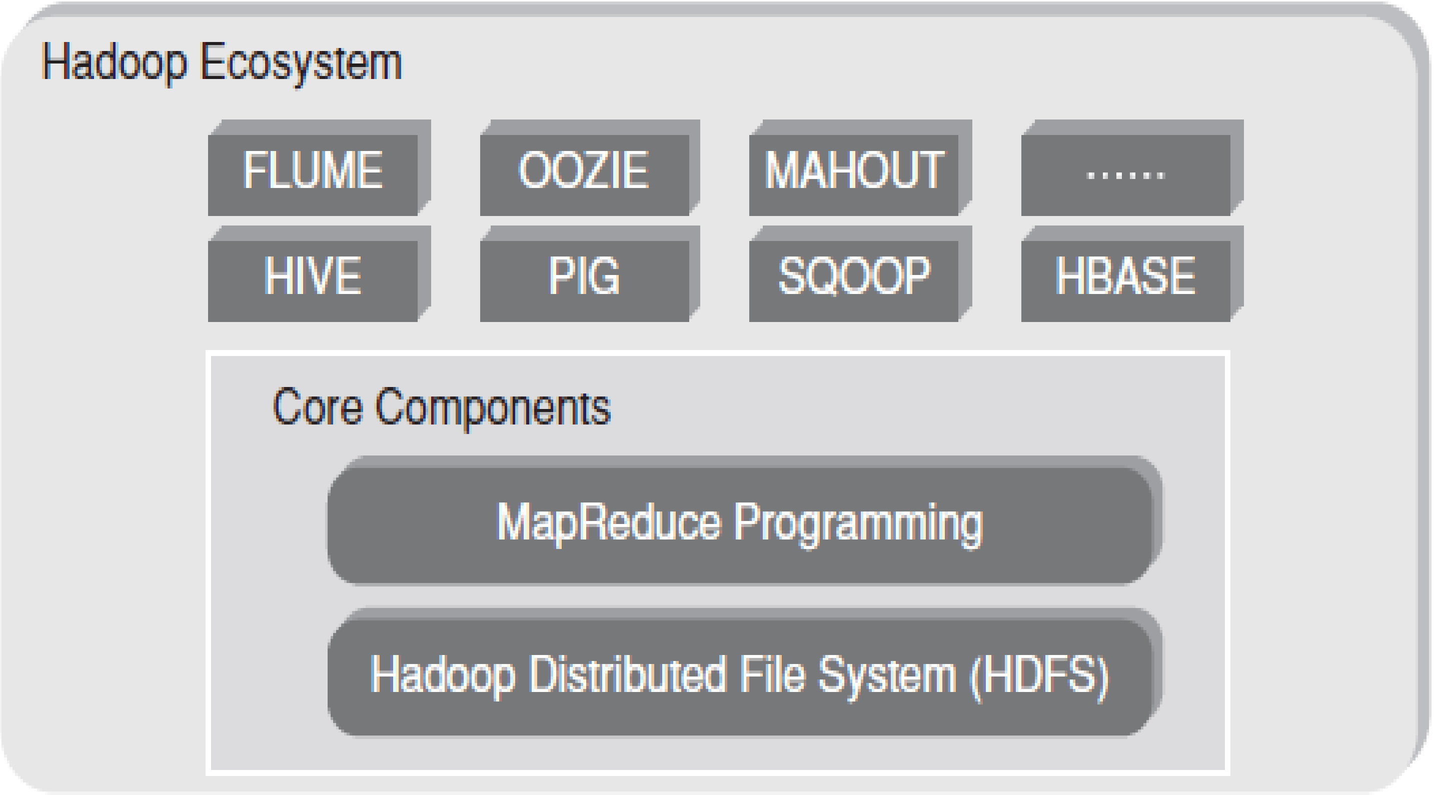
Framework: Means everything that you will need to develop and execute and application is provided – programs, tools, etc.

Distributed: Divides and stores data across multiple computers. Computation/Processing is done in parallel across multiple connected nodes.

Massive storage: Stores colossal amounts of data across nodes of low-cost commodity hardware.

Faster processing: Large amounts of data is processed in parallel, yielding quick response.

Hadoop Components



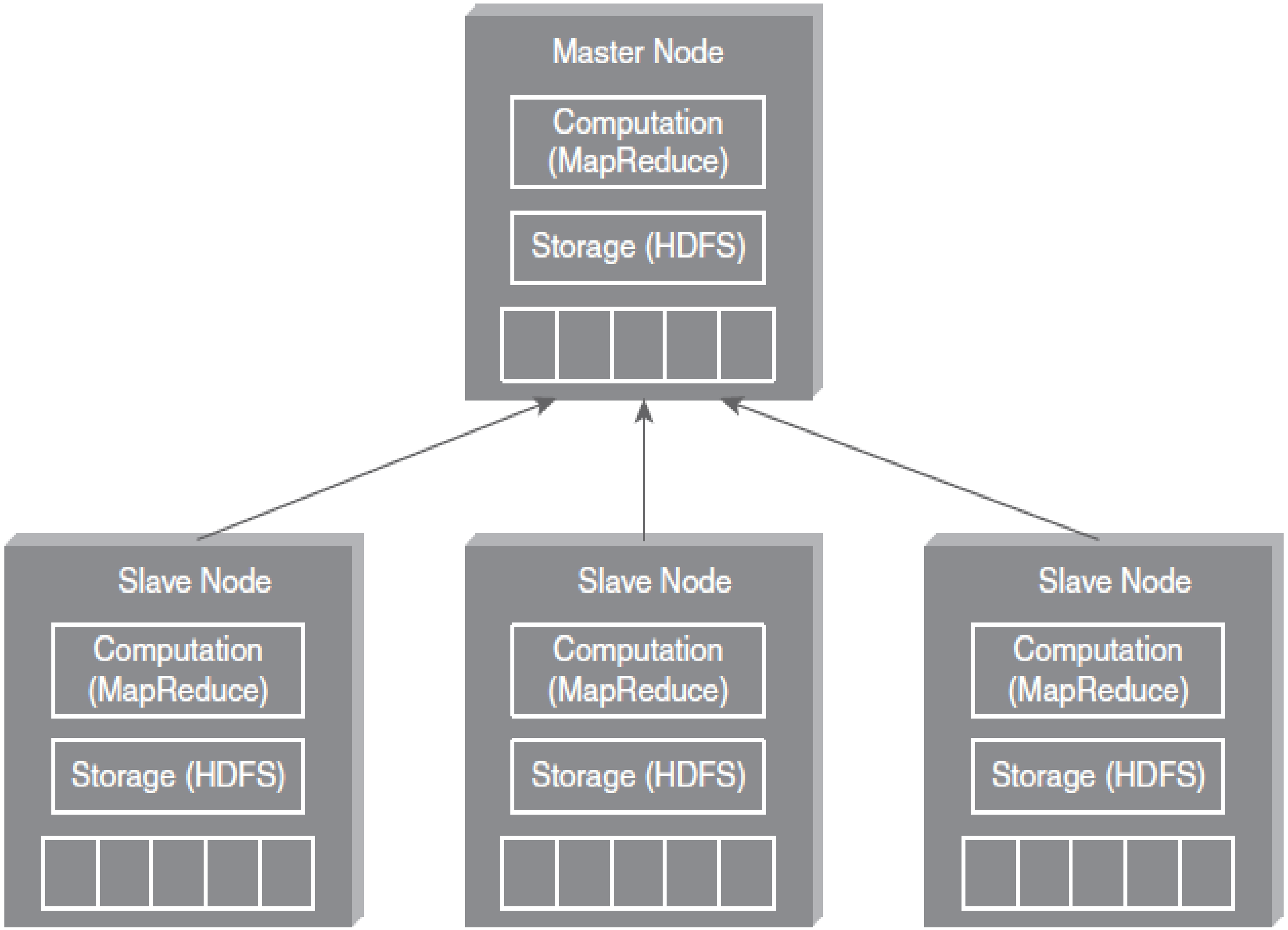
Hadoop Components

Hadoop Core Components:

HDFS:
(a) Storage component.
(b) Distributes data across several nodes.
(c) Natively redundant.

MapReduce:
(a) Computational framework.
(b) Splits a task across multiple nodes.
(c) Processes data in parallel.

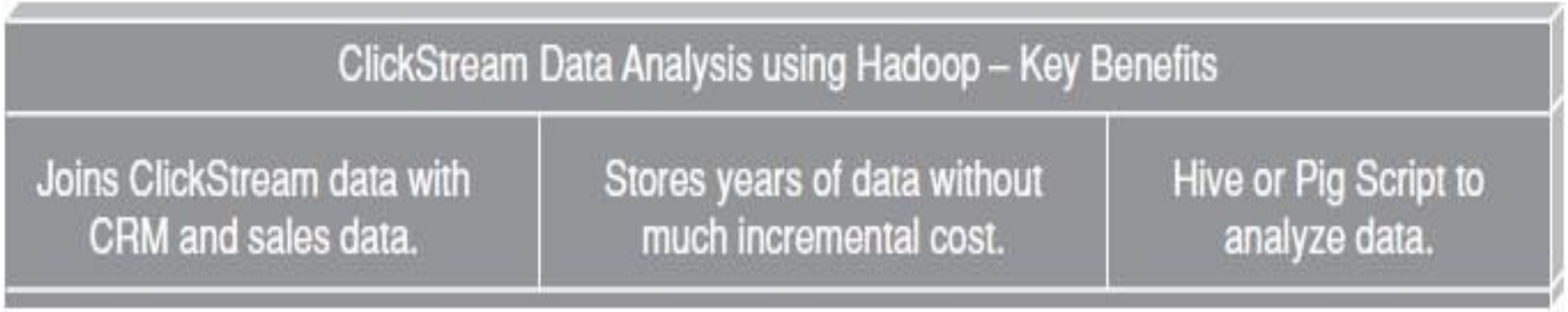
Hadoop High Level Architecture



Use case for Hadoop

ClickStream Data Analysis

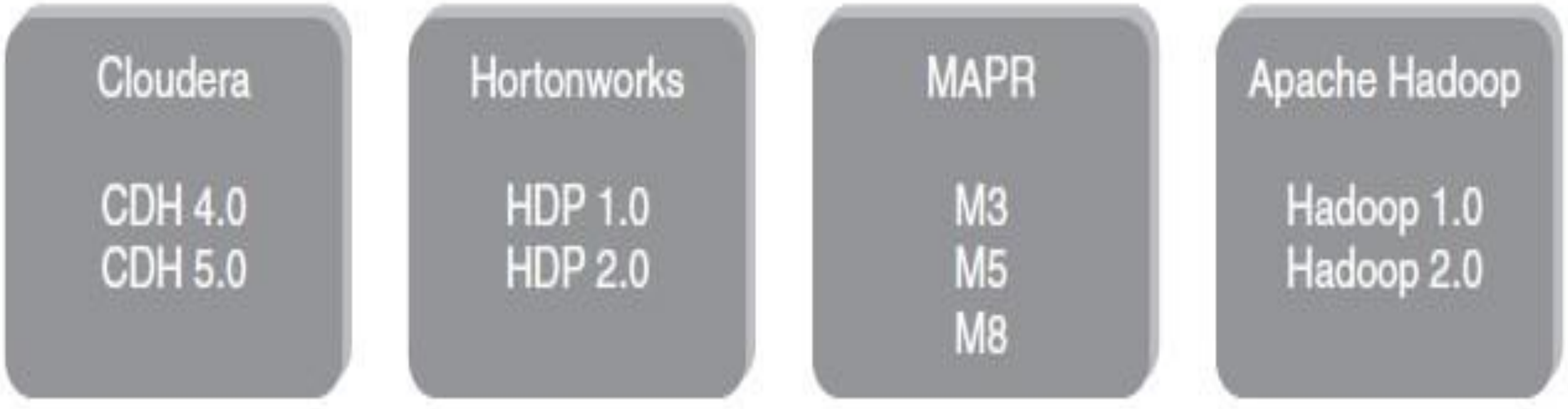
ClickStream data (mouse clicks) helps you to understand the purchasing behavior of customers. ClickStream analysis helps online marketers to optimize their product web pages, promotional content, etc. to improve their business.



ClickStream Data Analysis using Hadoop – Key Benefits		
Joins ClickStream data with CRM and sales data.	Stores years of data without much incremental cost.	Hive or Pig Script to analyze data.

Hadoop Distributors

Hadoop Distributors



Cloudera	Hortonworks	MAPR	Apache Hadoop
CDH 4.0	HDP 1.0	M3	Hadoop 1.0
CDH 5.0	HDP 2.0	M5	Hadoop 2.0
		M8	

HDFS
(HADOOP DISTRIBUTED FILE SYSTEM)

Hadoop Distributed File System

1. Storage component of Hadoop.

2.Distributed File System.

3. Modeled after Google File System.

4. Optimized for high throughput (HDFS leverages large block size and moves computation where data is stored).

5.You can replicate a file for a configured number of times, which is tolerant in terms of both software and hardware.

6.Re-replicates data blocks automatically on nodes that have failed.

7. You can realize the power of HDFS when you perform read or write on large files (gigabytes and larger).

8.Sits on top of native file system such as ext3 and ext4, which is described

HDFS Daemons

NameNode:

- Single NameNode per cluster.
- Keeps the metadata details

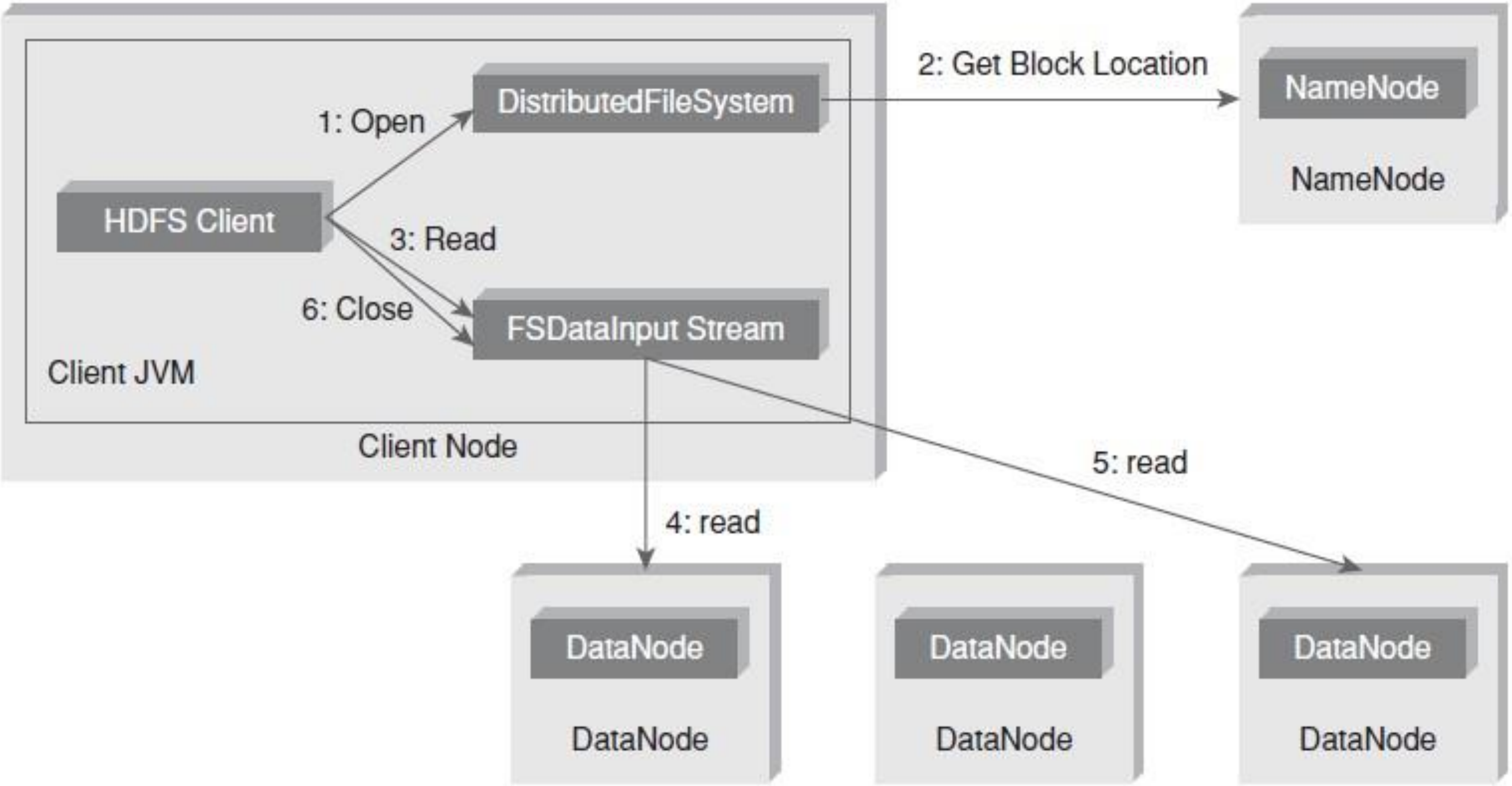
DataNode:

- Multiple DataNode per cluster
- Read/Write operations

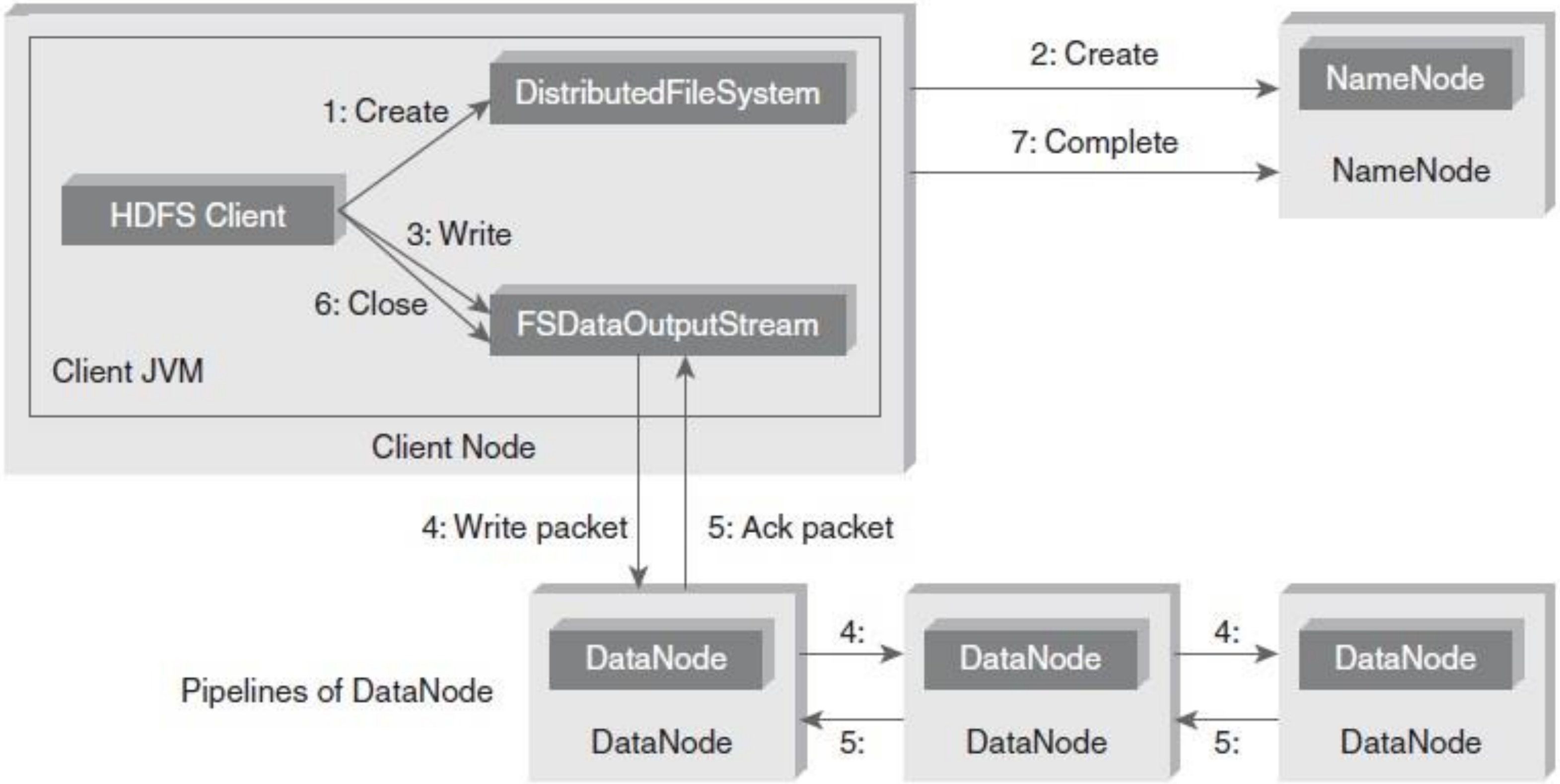
SecondaryNameNode:

- Housekeeping Daemon

Anatomy of File Read

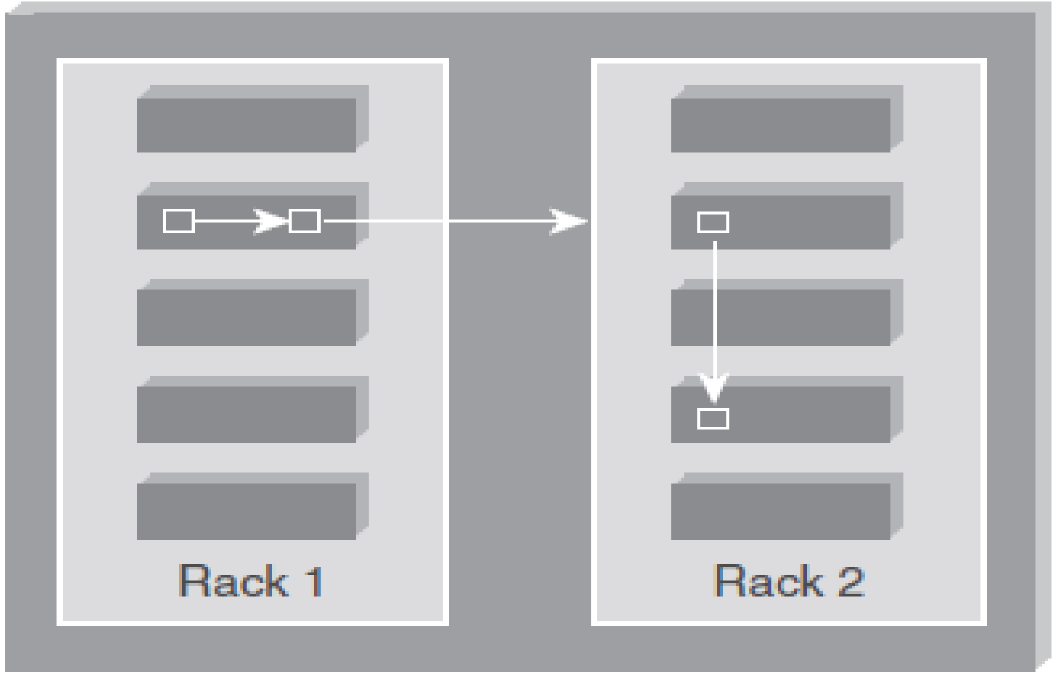


Anatomy of File Write



Replica Placement Strategy

As per the Hadoop Replica Placement Strategy, first replica is placed on the same node as the client. Then it places second replica on a node that is present on different rack. It places the third replica on the same rack as second, but on a different node in the rack. Once replica locations have been set, a pipeline is built. This strategy provides good reliability.



Working with HDFS Commands

Objective: To create a directory (say, sample) in HDFS.

Act:

```
hadoop fs -mkdir /sample
```

Objective: To copy a file from local file system to HDFS.

Act:

```
hadoop fs -put /root/sample/test.txt /sample/test.txt
```

Objective: To copy a file from HDFS to local file system.

Act:

```
hadoop fs -get /sample/test.txt /root/sample/testsample.txt
```

Special Features of HDFS

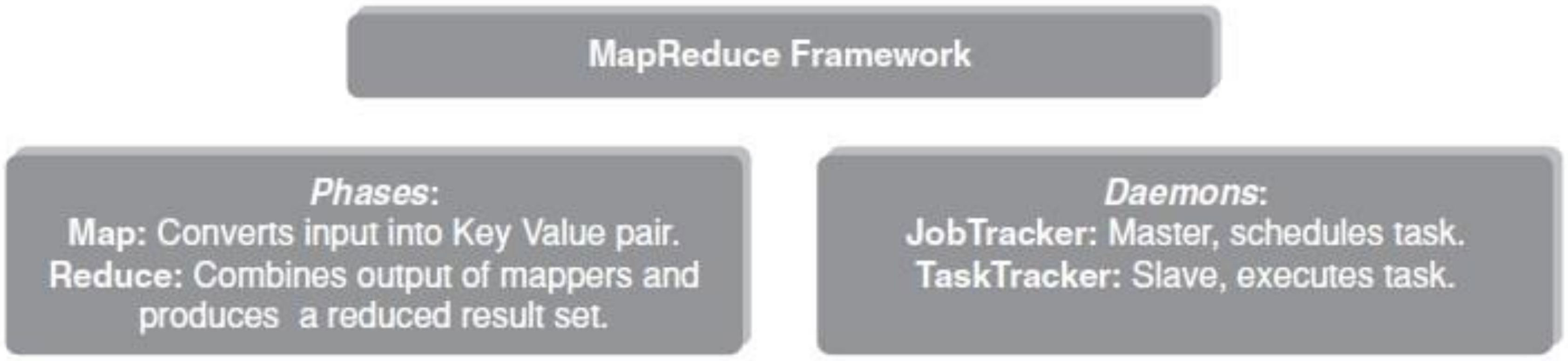
Data Replication: There is absolutely no need for a client application to track all blocks. It directs the client to the nearest replica to ensure high performance.

Data Pipeline: A client application writes a block to the first DataNode in the pipeline. Then this DataNode takes over and forwards the data to the next node in the pipeline. This process continues for all the data blocks, and subsequently all the replicas are written to the disk.

Processing with Hadoop

What is MapReduce Programming?

MapReduce Programming is a software framework. MapReduce Programming helps you to process massive amounts of data in parallel.

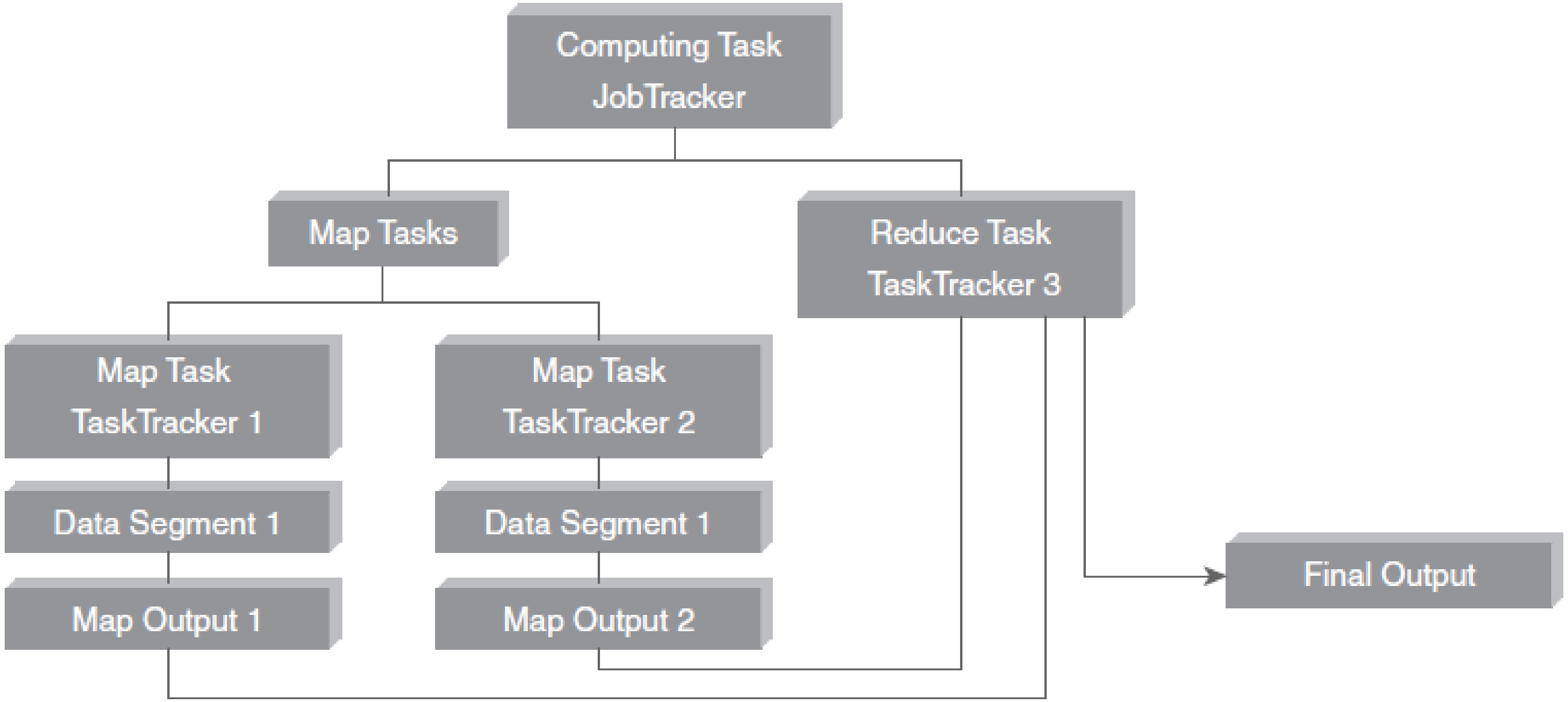


MapReduce Framework

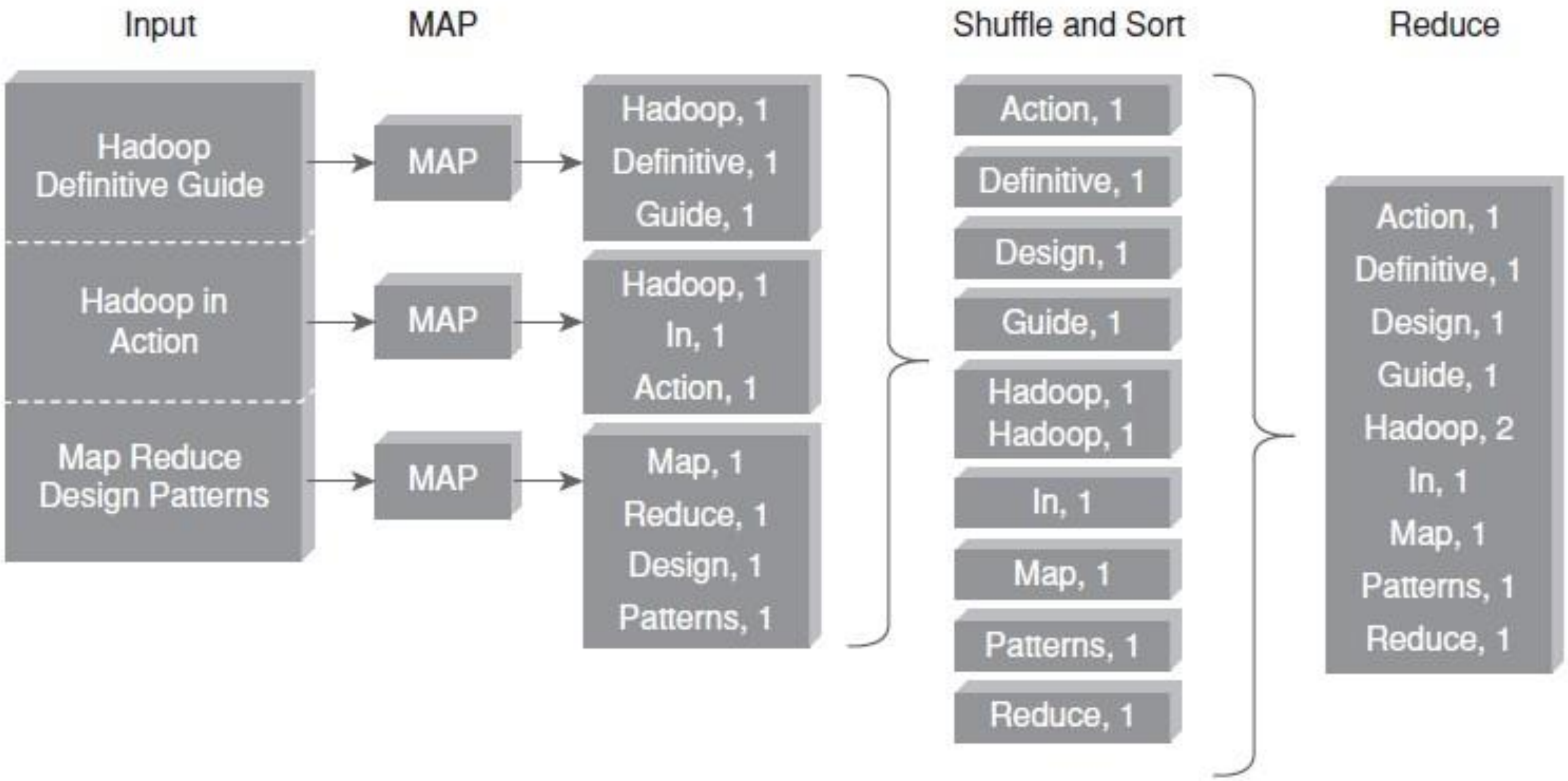
Phases:
Map: Converts input into Key Value pair.
Reduce: Combines output of mappers and produces a reduced result set.

Daemons:
JobTracker: Master, schedules task.
TaskTracker: Slave, executes task.

How MapReduce Programming Works



MapReduce – Word Count Example



Input

Hadoop Definitive Guide

Hadoop in Action

Map Reduce Design Patterns

MAP

MAP

MAP

MAP

Hadoop, 1
Definitive, 1
Guide, 1

Hadoop, 1
In, 1
Action, 1

Map, 1
Reduce, 1
Design, 1
Patterns, 1

Shuffle and Sort

Action, 1

Definitive, 1

Design, 1

Guide, 1

Hadoop, 1
Hadoop, 1

In, 1

Map, 1

Patterns, 1

Reduce, 1

Reduce

Action, 1
Definitive, 1
Design, 1
Guide, 1
Hadoop, 2
In, 1
Map, 1
Patterns, 1
Reduce, 1

MANAGING RESOURCES AND APPLICATIONS WITH HADOOP - YARN

(YET ANOTHER RESOURCE NEGOTIATOR)

Limitations of Hadoop 1.0 Architecture

1. Single NameNode is responsible for managing entire namespace for Hadoop Cluster.

2. It has a restricted processing model which is suitable for batch-oriented MapReduce jobs.

3. Hadoop MapReduce is not suitable for interactive analysis.

4. Hadoop 1.0 is not suitable for machine learning algorithms, graphs, and other memory intensive algorithms.

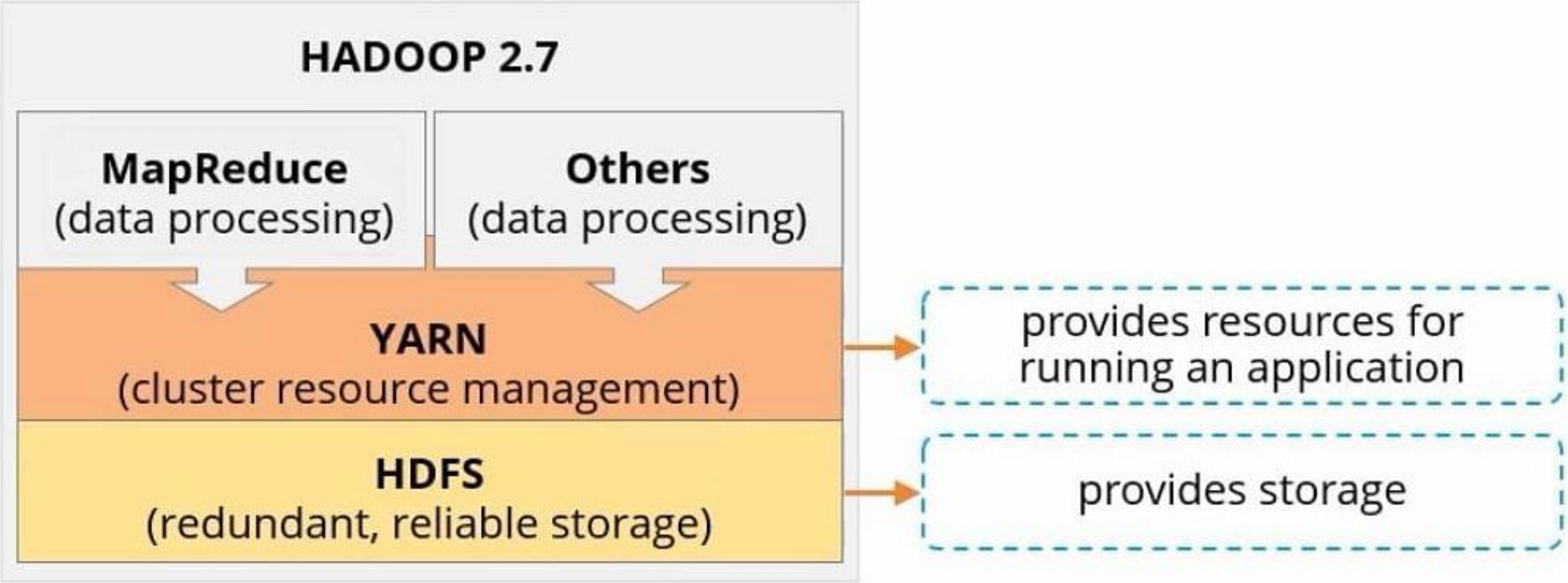
5. **MapReduce** is responsible for **cluster resource management and data processing.**

YARN Infrastructure

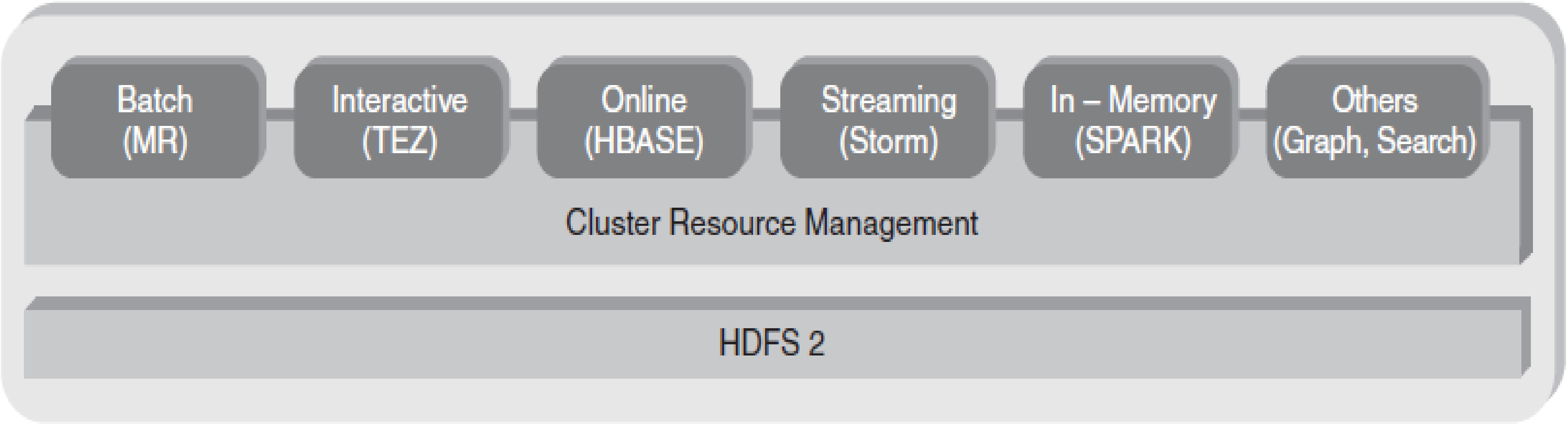
The YARN Infrastructure is responsible for providing computational resources such as CPUs or memory needed for application executions.

YARN infrastructure and HDFS are completely independent.

The MapReduce framework is only one of the many possible frameworks that run on YARN.


HADOOP 2.7
MapReduce
(data processing)
Others
(data processing)
YARN
(cluster resource management)
HDFS
(redundant, reliable storage)
provides resources for running an application
provides storage

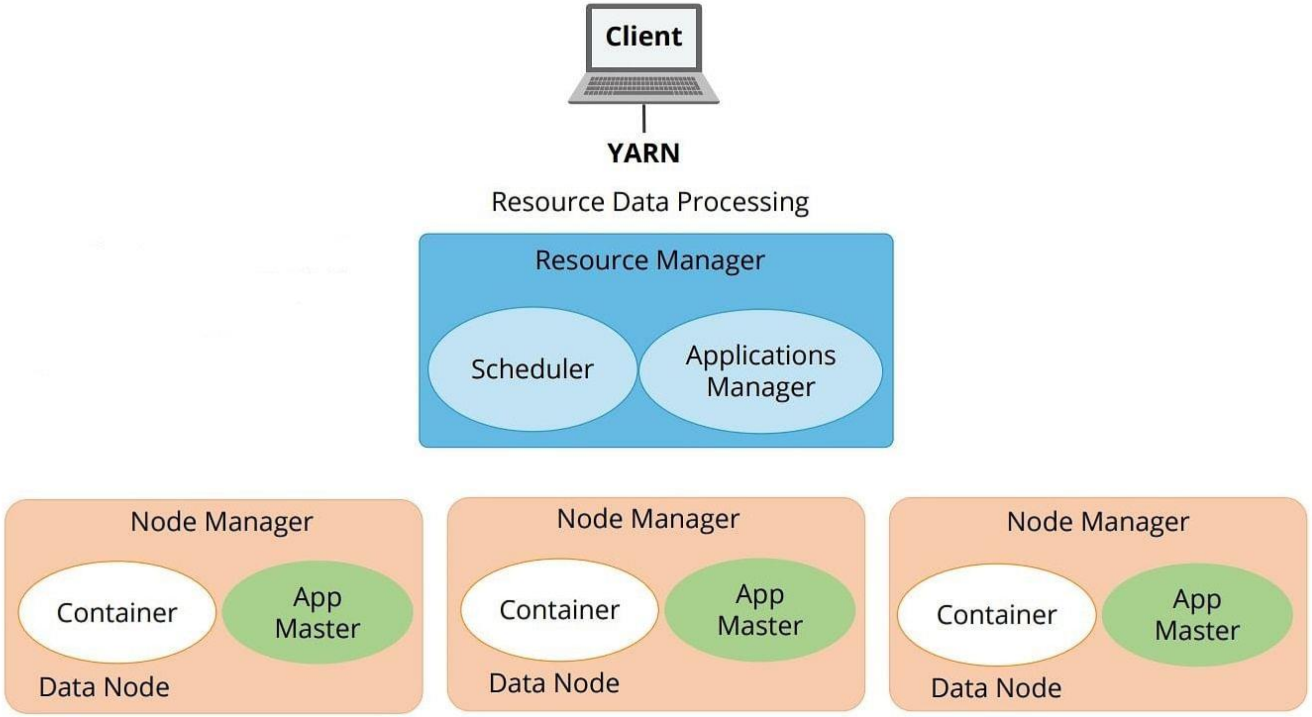
Hadoop 2 YARN: Taking Hadoop beyond Batch



The three important elements of the YARN architecture are:

- Resource Manager
- Application Master
- Node Managers

These three Elements of YARN Architecture are shown in the given below diagram.



Hadoop 2 YARN: Taking Hadoop beyond Batch

The fundamental idea behind this architecture is splitting the JobTracker responsibility of resource management and Job Scheduling/Monitoring into separate daemons. Daemons that are part of YARN Architecture are described below.

A Global ResourceManager: Its main responsibility is to distribute resources among various applications in the system. It has two main components:

NodeManager: This is a per-machine slave daemon. NodeManager responsibility is launching the application containers for application execution. NodeManager monitors the resource usage such as memory, CPU, disk, network, etc. It then reports the usage of resources to the global ResourceManager.

Per-application Application Master: This is an application-specific entity. Its responsibility is to negotiate required resources for execution from the ResourceManager. It works along with the NodeManager for executing and monitoring component tasks.

Interacting with Hadoop Ecosystem

Interacting with Hadoop Ecosytem

Pig : Pig is a data flow system for Hadoop. It uses Pig Latin to specify data flow. Pig is an alternative to MapReduce Programming. It abstracts some details and allows you to focus on data processing.

Hive: Hive is a Data Warehousing Layer on top of Hadoop. Analysis and queries can be done using an SQL-like language. Hive can be used to do ad-hoc queries, summarization, and data analysis. Figure 5.31 depicts Hive in the Hadoop ecosystem.

Sqoop: Sqoop is a tool which helps to transfer data between Hadoop and Relational Databases. With the help of Sqoop, you can import data from RDBMS to HDFS and vice-versa. Figure 5.32 depicts the Sqoop in Hadoop ecosystem.

HBase: HBase is a NoSQL database for Hadoop. HBase is column-oriented NoSQL database. HBase is used to store **billions of rows and millions of columns.** HBase provides random read/write operation. It also supports record level updates which is not possible using HDFS. HBase sits on top of HDFS. Figure 5.33 depicts the HBase in Hadoop ecosystem.

Answer a few quick questions…

Match the columns

Column A	**Column B**
HDFS	DataNode
MapReduce Programming	NameNode
Master node	Processing Data
Slave node	Google File System and MapReduce
Hadoop Implementation	Storage

Match the columns

Column A	**Column B**
JobTracker	Executes Task
MapReduce	Schedules Task
TaskTracker	Programming Model
Job Configuration	Converts input into Key Value pair
Map	Job Parameters

Thank You