**Big Data Testing:**

**Big Data:**

Big data is data which is beyond storage and processing capacity of conventional database systems is called big data.

Or

Big Data is the term for collection of data sets of large and complex that it becomes difficult to process using conventional data bases system.

**Sources of Bigdata**

* **Social Media**

Facebook, Twitter, Linkedin etc

* **Mobile Devices**

Call, text, location, and app activity etc.

* **Internet transactions**

Banking transactions, ecommerce transactions etc.

* **Network devices / sensors**

Sensors (Proximity, Temperature), Internet connection to different devices etc.

**Big data Characteristics**

We can categorize any data is big data based on 5 V’s

1.Volume (Size of data)

2.Velocity (Processing speed of data)

3.Variety (Types of data)

4.Value (Worth of data being extracted)

5.Veracity (How much accurate is the data being extracted)

If any data having above 5 characteristics called as BIG DATA.

**Big data Classification:**

It has been classified in three different types of data

1.Structured Data

Data which is stored in tabular format and which is having proper structure and has relationship between the tables.

Ex: Table format data like employee table.

2.Semi-Structured Data

Data Which has some structure but can not save in tabular format is known as semi-structred data.

Ex: .xlsx, email, .txt, .doc etc.

3.Unstructured Data

Data which is not having any structure and cannot save in tabular format is known as unstructured data.

Ex: video files, call recordings, audio files etc.

Big data concept has been implemented by using **Hadoop** eco-system

**Hadoop:**

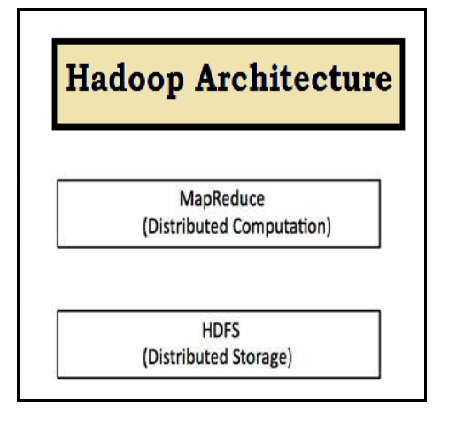
Hadoop is a free, Java-Based programming framework that supports processing of large data sets in a distributed computing environment.

It has maintained by Apache software foundation.

Hadoop used commodity (cheap) hardware and clusters concepts.

Hadoop is recommended for large datasets.

**Hadoop Architecture 1.0**



**Hadoop Components:**

**MapReduce:**

Processing of data which is stored on **HDFS (Hadoop Distributed File System).**

**HDFS:**

It is used to store huge amount of data sets.

**HDFS (Hadoop Distributed File System):**

HDFS is specially designed file system for storing data set with cluster of commodity hardware with streaming access pattern.

**Cluster:** Group of systems connected to LAN.

**Commodity H/W:** Cheap hardware.

**Streaming access pattern:** Streaming access pattern means you can write once, read number of times but cannot change the content of file once it is kept in HDFS.

WORM (write once read many times)

Normal file system block size = 4KB

In HDFS 1.0 block size 64MB and HDFS 2.0 size is 128MB

**HDFS Services (Daemons/ nodes)**

HDFS provides 5 services

**Master Services**

1.Name Node

2.Secondary Name Node

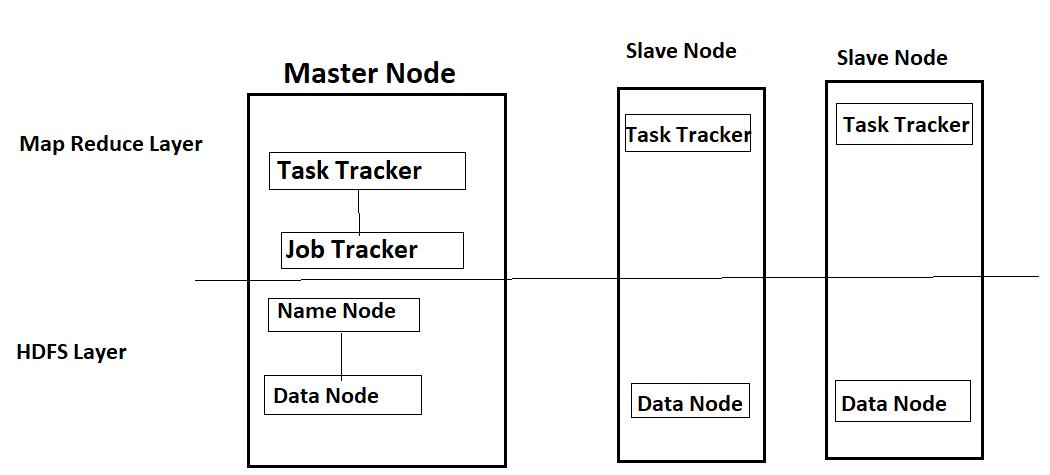
3.Job Tracker (Resource Manager)

**Slave Services**

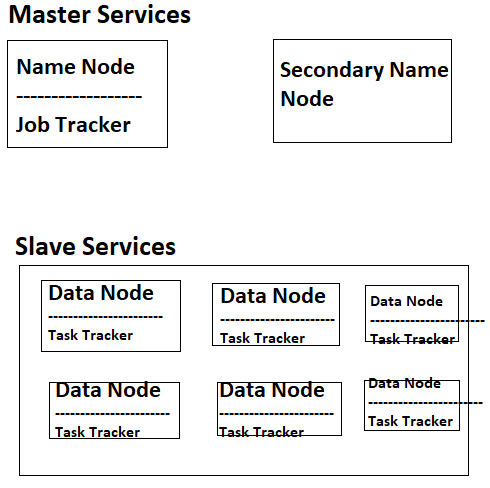
4.Data Node

5.Task Tracker (Node Manager)

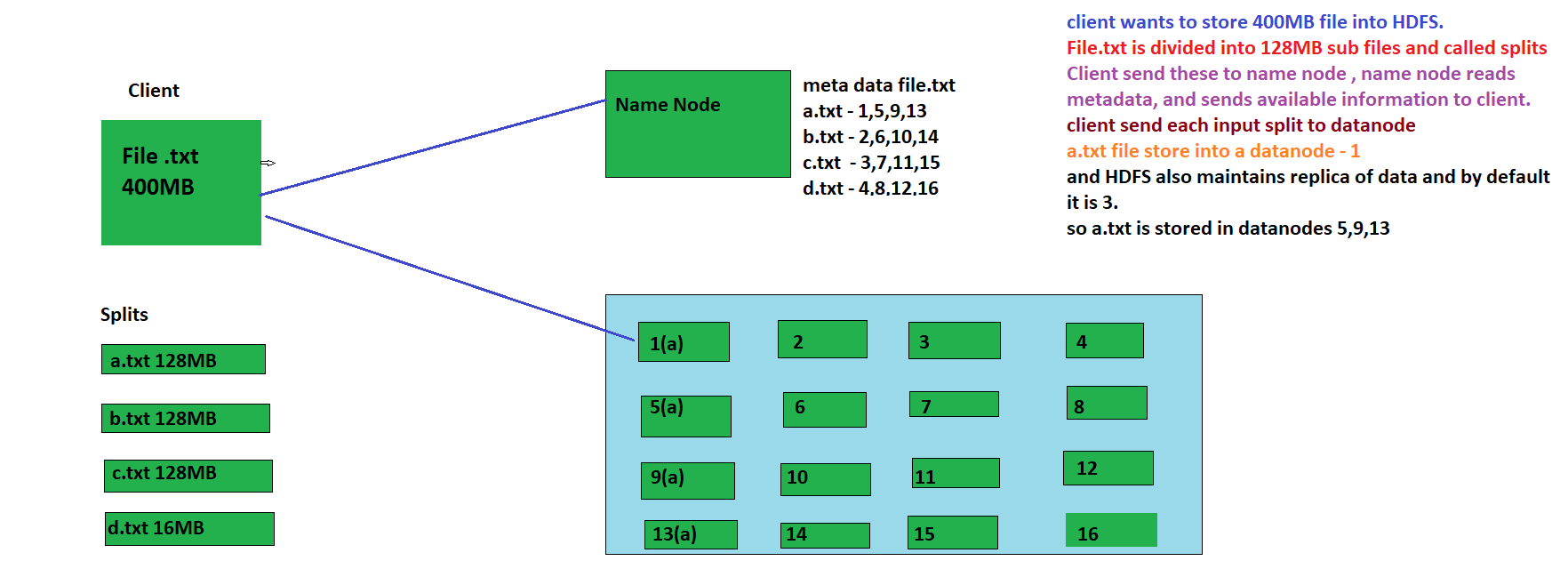
**High Level Architecture of Hadoop**



**Master and Salve Services**



**Storing data into HDFS**



**Map Reduce (Retrieving of data)**

* If we want to process data then we write simple program for processing, it is called as **MAP.**
* Whenever we want to process data then we send command/ program to Job Tracker for processing job/file/data.
* Job tracker asks to Name Node and name node will reply metadata to job tracker.
* Job tracker knows on which blocks data is stored on which data nodes.
* Using meta data, Job tracker communicates to Task tracker and assign the task and task will perform or complete by task tracker and same will be informed to Job tracker once it completes job.

***Map reduce task is always performed by bigdata developer and the scripts are generally written in JAVA or Python.***

**Example: Sum of 1000 numbers**

Suppose we have 10 persons to add 1000 numbers

Map Reduce Process

**Step -I:** Input splitting: 1-> 1 to 100, 2->101 to 200, 3->201 to 300, ………………………. 10->901 to 1000

**Step-II:** Mapping Method: 1 -🡪 1+2 +… 100 =5050

Map 1🡪 5050 ,2🡪 15150…….10🡪 95950

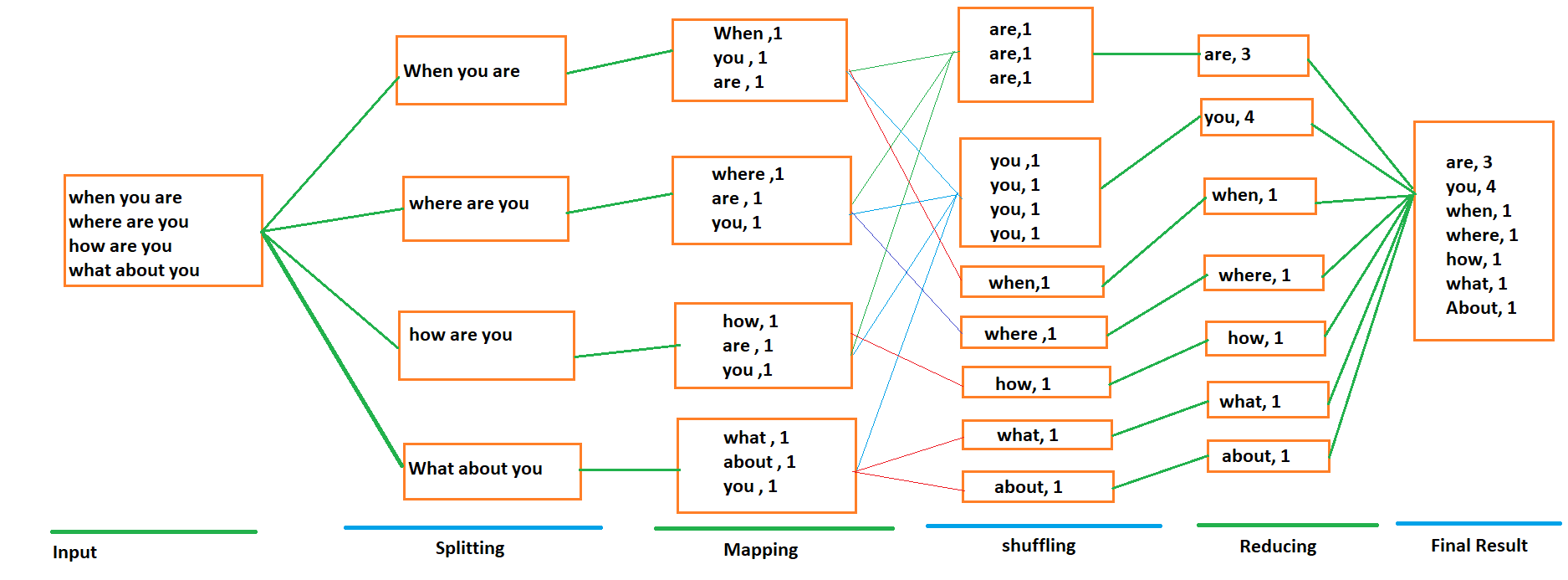
**Step-III:** Reduce Method 🡪 after addition performed by all 10 persons, reducer will collect and again add all the results.

***Big data Developer will develop two scripts one for mapper and second for reducer and implement the processing concept.***

Two ways to implement map reduce

1. Splitting 🡪 Mapping 🡪 Reducer
2. Splitting 🡪 Mapping 🡪 Shuffling 🡪 Reducing

Map Reduce ex: word Count



**How to Test Map Reduce Job created in Python**

Prerequisite: Python Scripts or Java Scripts

1.mapper.py

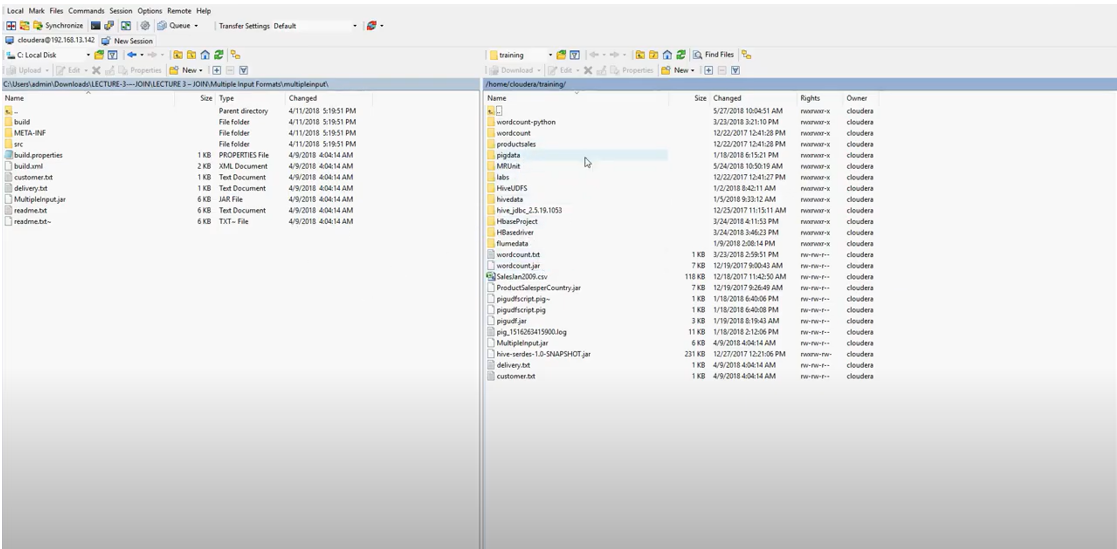
2.reducer.py

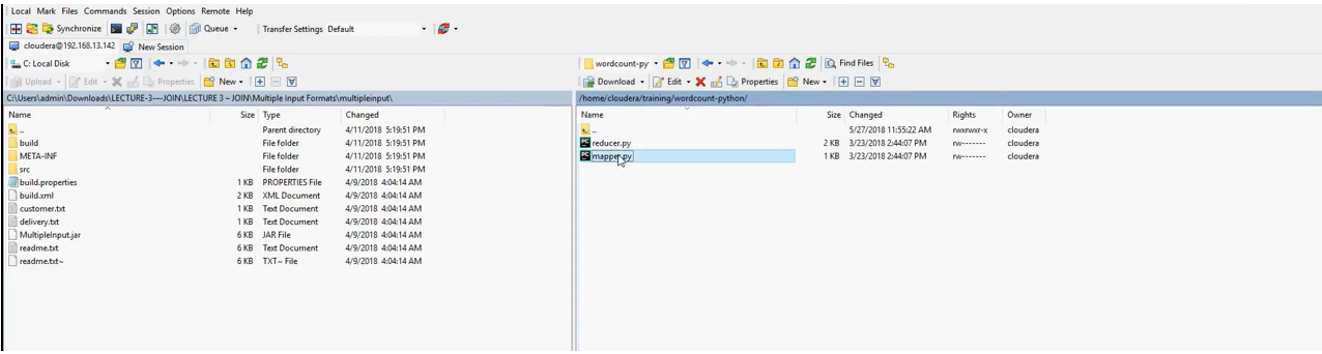
**Step1:** Test python scripts locally before using in MapReduce.

**Step2:** Copy data files into HDFS

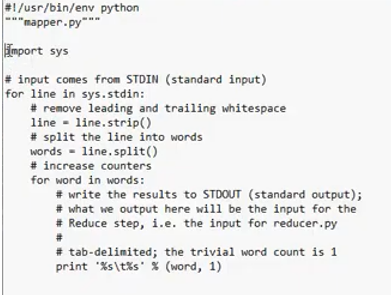
**Step3:** Execute MapReduce Scripts in HDFS path or on top of HDFS storage.

**Win SCP** – it is used to copy files from widows to Unix and vice-versa

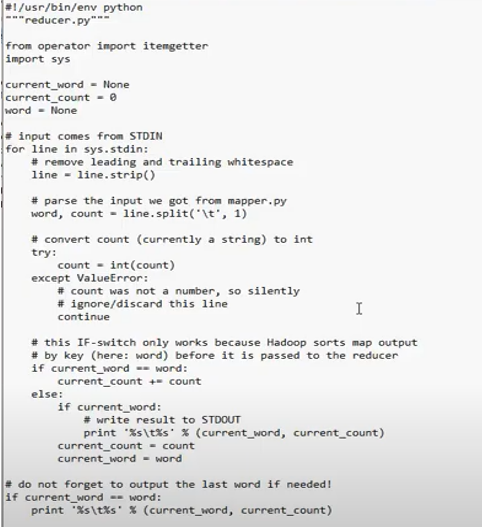




Mapper Script



Reducer Script



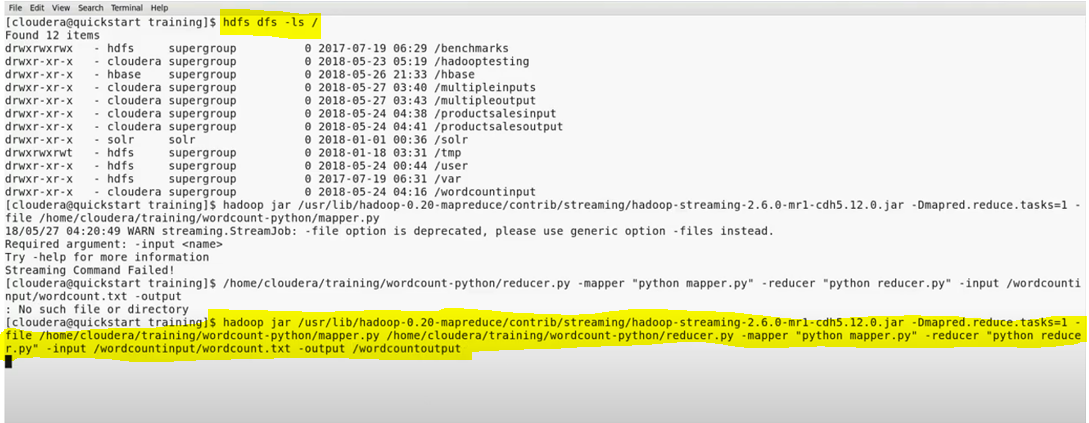


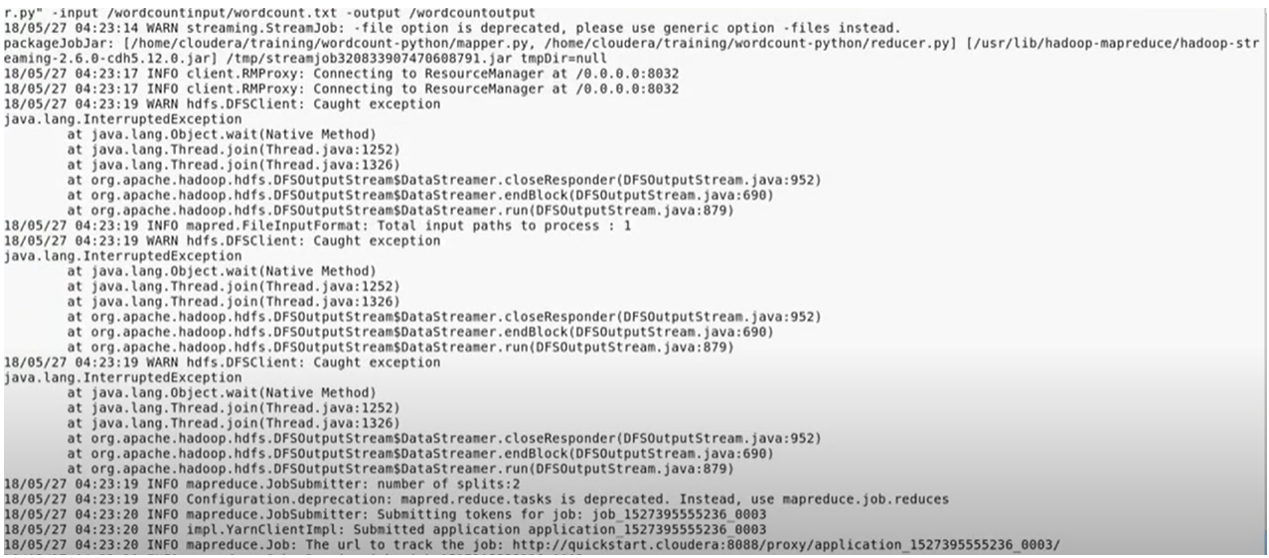
Using Hadoop terminal by using Hadoop commands

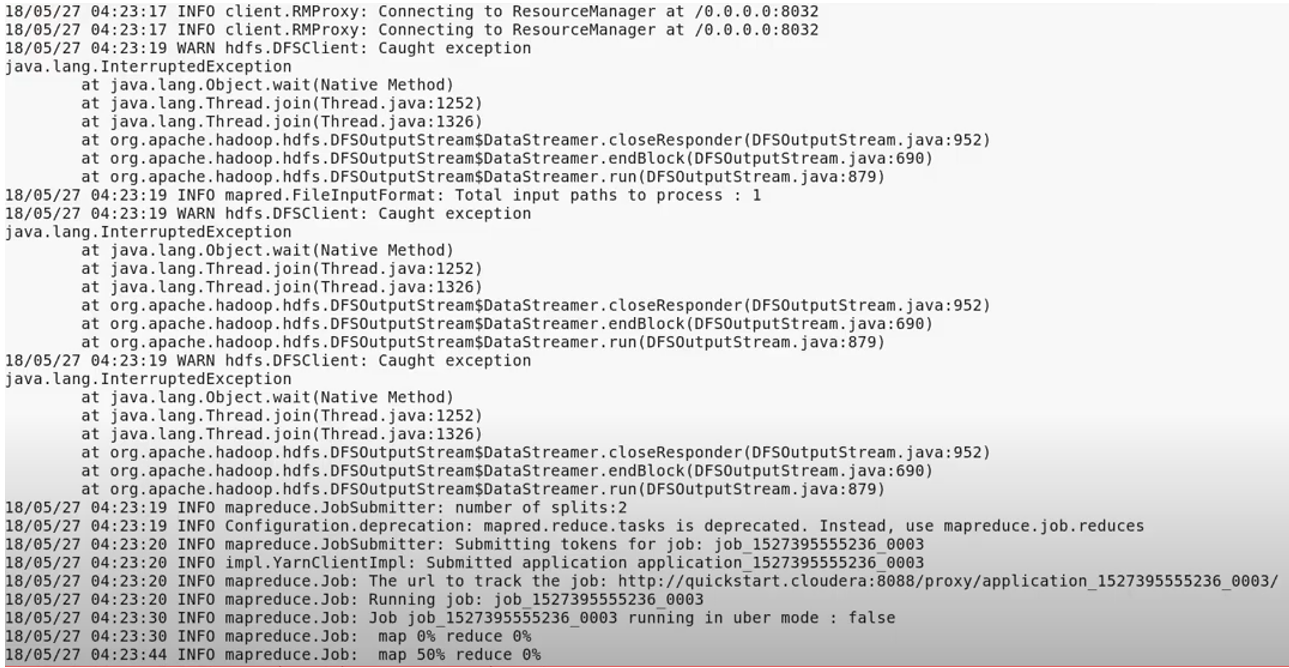
If We want to use Unix commands on Hadoop Server then we must and should include either

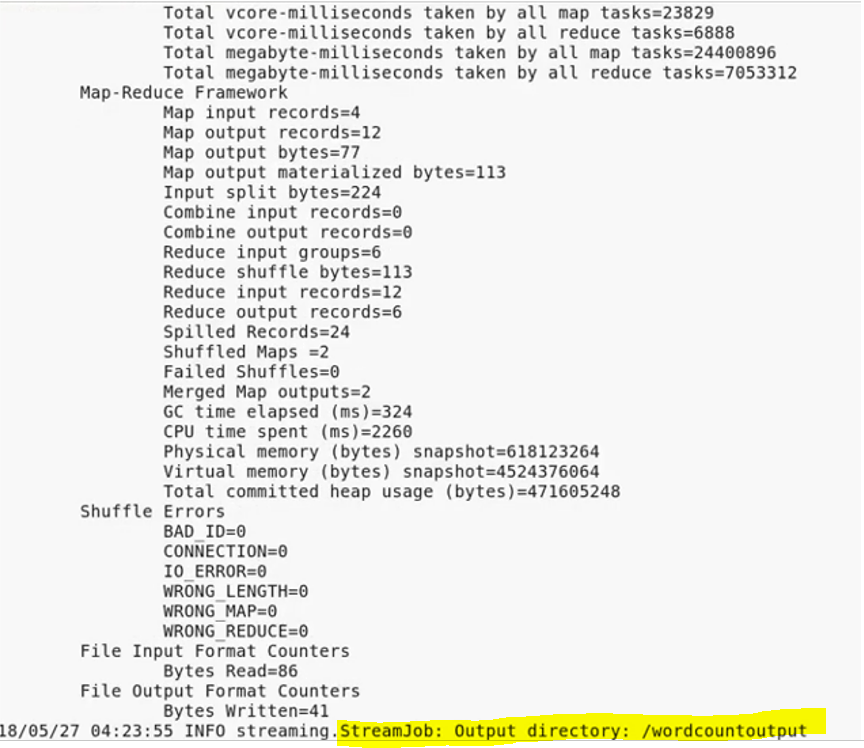
**Hdfs dfs -unix commands or Hadoop fs -unix commands**

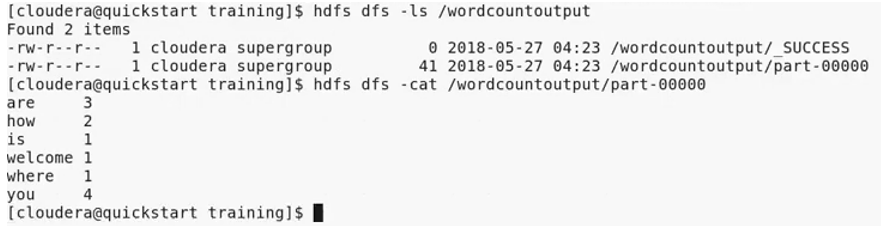
**For ex:** hdfs dfs -ls -l **or** hadoop fs -ls -l











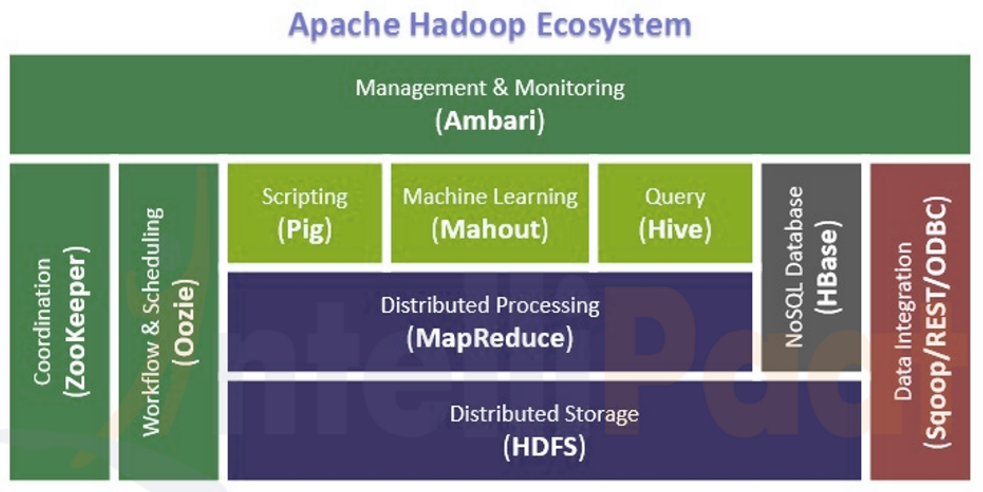
After successful extraction of data from HDFS by using MapReduce algorithm, we will capture some of source records from source and compare against extracted HDFS data.

Hadoop Tool – Language – It will convert language into map-Reduce script and extract data from HDFS.

**Hadoop Ecosystem**

In Hadoop 2.x from onwards supports YARN (Yet another Resource Navigator), Which helps to connect different application like Pig, Hive etc.

Because of YARN it is possible to built an Apache Hadoop Ecosystem.



**Hadoop:** Hadoop provides scalable solution to store and process huge data sets in distributed fashion.

**Apache Hive:** Hive is data warehousing tool that allows us to perform big data analytics using Hive query language and which is very similar to SQL.

**Apache Pig:** Apache pig is used to analyze large data sets.

**Apache Spark:** Spark is used for data processing engine that allows us to efficiently execute streaming or SQL workloads.

**Apache HBase:** HBase is NoSQL database that allows us store unstructured and semi-structured data with real time read/write access.

**Apache Sqoop:** Sqoopis used for transferring data in between Hadoop and relational data base servers.

**Apache Oozie:** Apache Oozie is used to schedule hadoop jobs. It will integrate multiple jobs sequence wise and it is integrated with hadoop by using YARN

**Apache Zookeeper:** Zookeeper is used for maintaining configuration like synchronization among all the tools present on hadoop ecosystem. Apache Kafka is also used to manage configuration.

**Ambari:** Ambari is used for managing and monitoring Apache hadoop ecosystem.

**HDFS Commands:**

**1.JPS:** Prints hadoop processes.

**2.fsck:** it will check the health of HDFS.

**3.hdfs dfs -ls:** Listing of files files / directories present in HDFS.

All the Unix command we can use on Hadoop server but before using any Unix command we must add **hdfs dfs -command or hadoop fs -command.**

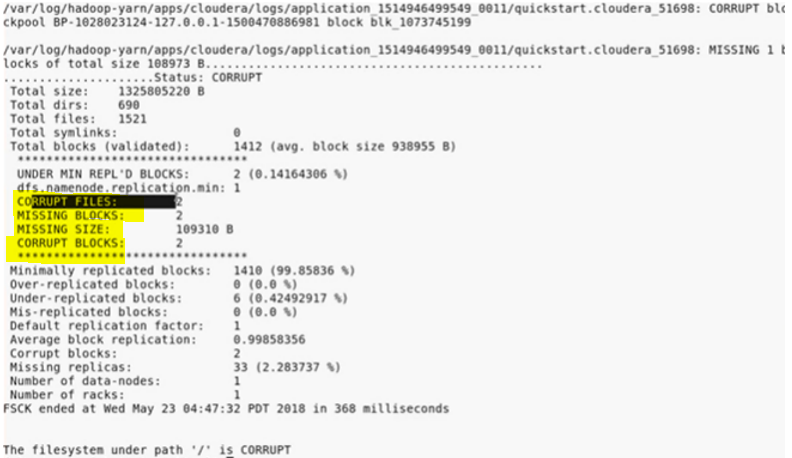
**1.JPS (Java Process Status):**

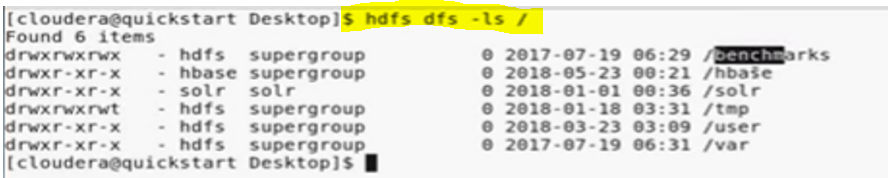
JPS will show whether all the hadoop services are up and running or not and print all the hadoop process.



**2.fsck (File system Check):**

Check the health of HDFS.



**3.hdfs dfs -ls:**

**Apache Hive:**

Hive is data warehouse tool and it will process structured data in hadoop.

It resides on top of HDFS and it is used for easier data analysis and processing.

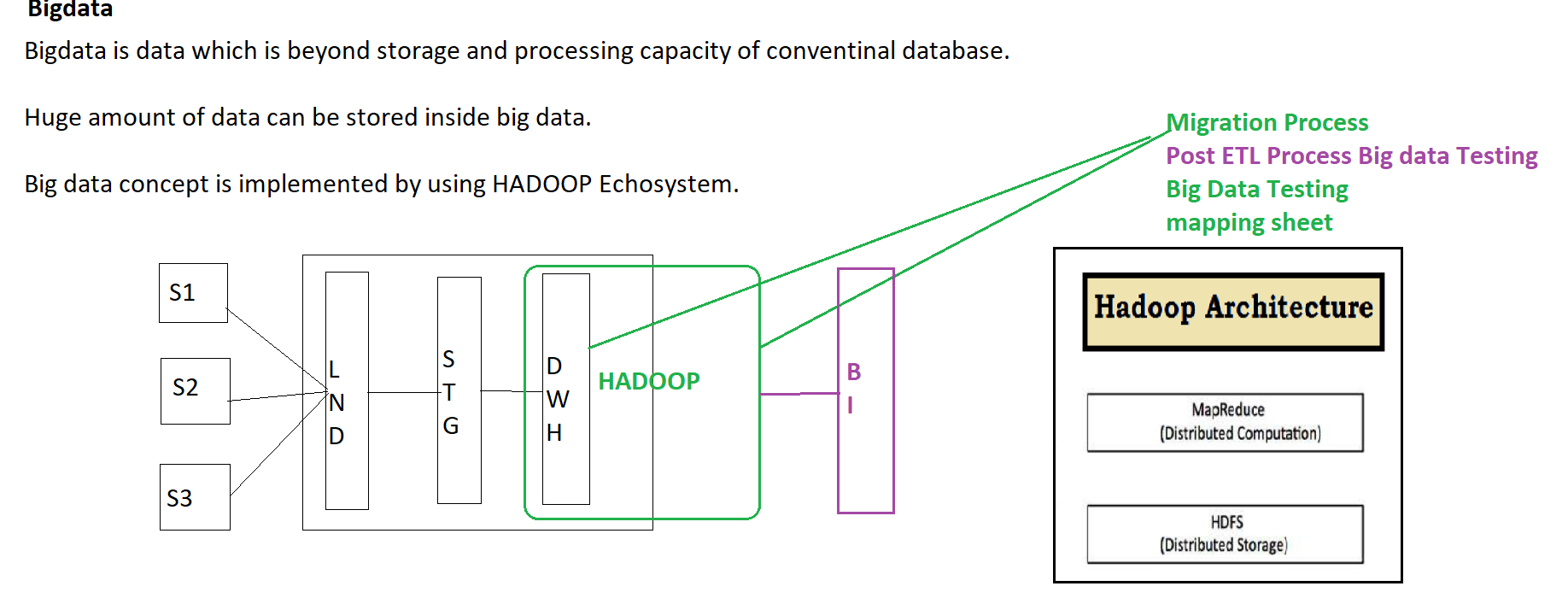
**Features:**

* It stores schemas in database and processed data into HDFS.
* It is designed for OLAP.
* It provides SQL type of language for querying and it is called Hive query language or HQL.

**Hive Architecture:**

Big data Process

**Post ETL**



**Pre ETL**