MODULE - 02.

HADOOP CORE COMPONENTS.

Fig : Core components of Hadoop

1. Hadoop Common: The common module contains libraries and utilities that are required by other module.

2. HDFS : A java based distributed file system which can store all kinds of data on the disk at the cluster.

3. Map Reduce V1 - Software programming model in Hadoop 1 using Mapper and Reduce. The V1 process large sets of data in parallel or in batches.

4. YARN - Software for managing resources for computing. The user application tasks, or sub-tasks run in parallel at the Hadoop, uses scheduling and handles the requests for resources in distributed running of tasks.

5. Map Reduce V2 - Hadoop 2 YARN - based for parallel processing of large datasets and distributed processing of the application tasks.

SPARK

- Spark is an open source cluster computing framework of Apache Software foundation.
- Hadoop deploys data at the disk.
- Spark provision for in memory analytics enables OLAP, and real time processing.
- Spark does faster procession of Big Data.
- Spark has been adapted by large organisations, such as Amazon, E-Bay and Yahoo.
- Spark run on cluster with thounds of nods.

FEATURES OF HADOOP

1. Fault efficient, scalable, flexible and modular design
2. Robust design of HDFS
3. Store and process Big Data.
4. Distributed cluster computing model with data locality.
5. Hardware fault tolerant
6. Open source framework
7. Java and Linux based.

HADOOP ECOSYSTEM COMPONENTS

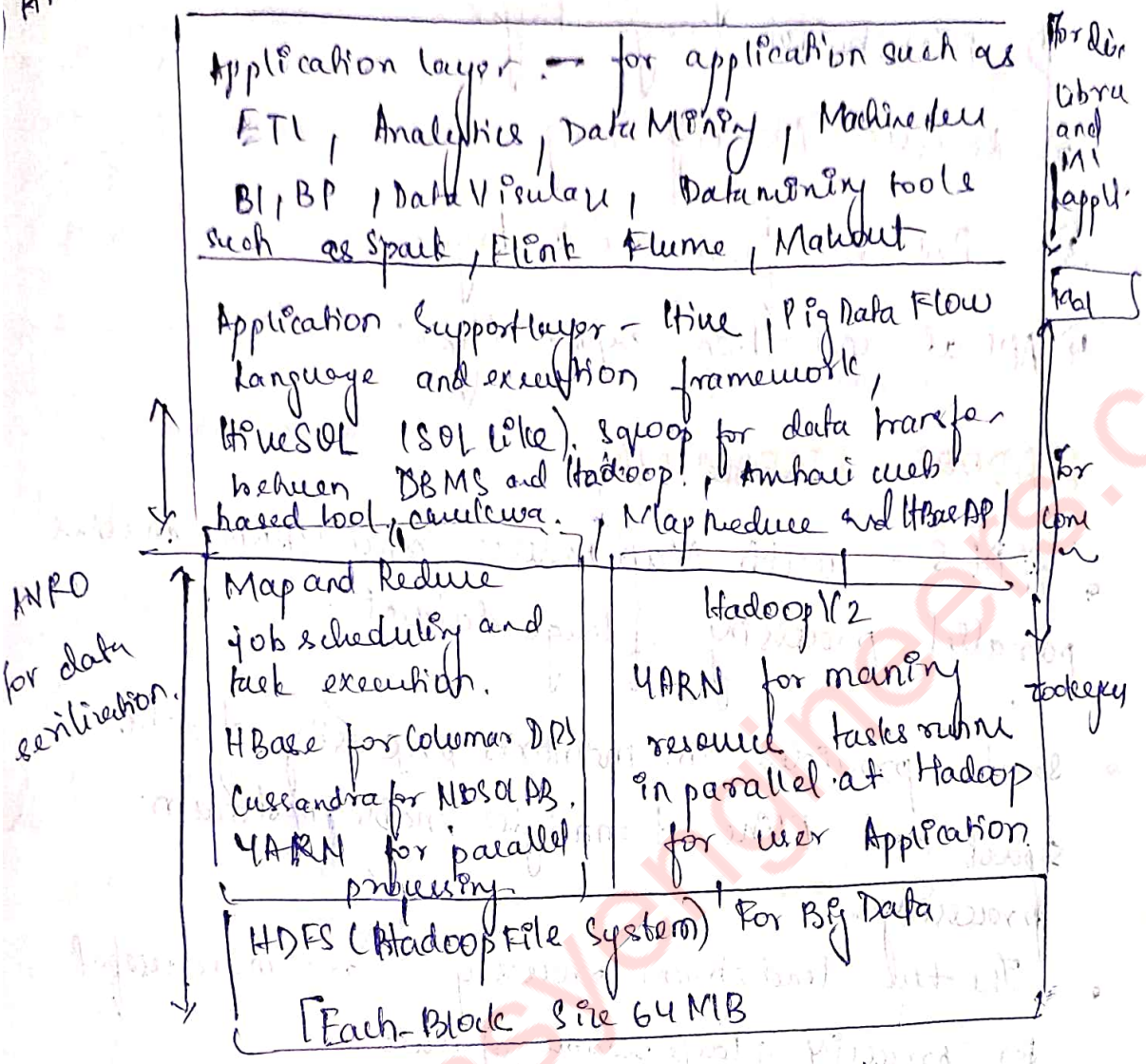


Fig: Hadoop main components and ecosystem compu

- Hadoop ecosystem is combination of different technology
- Hadoop consists of own applications which tie up together with hadoop.
- System components support the storage, processing, access and analysis of security for Big Data.
- The system includes application and support layer such as AVRO, Zookeeper, Spark, Flink, Flume, Makout, Cassandra, Ambari, Pig, Hive, Mahout.

YT has 4 layers

1) Distributed storage layer.
2) Resource Manager layer.
3) Process framework layer.
4) API at application support layer

HADOOP STREAMING

- HDFS with MapReduce and YARN enables parallel processing of large dataset.
- Spark provides in-memory processing of data. Spark and Flink technologies enable in-stream processing.
- The two lead stream processing are more useful for processing a large volume of data.

HADOOP PIPE

- Hadoop Pipe is name of C++ interface to Hadoop MapReduce.
- Pipes uses standard input and output to communicate with map and reduce tool.
- Pipe is socket or channel over which task tracker communicate with the process.
- Applications which require faster numerical computation can achieve higher throughput using C++ when we through pipes, compare

HDFS Data storage

Hadoop stores data in cluster.

Each cluster has data store called racks.

Each racks stores no of DataNodes

Each DataNode has large no of data Blocks.

The nodes are distributed across the cluster

The nodes have processing and storage capabilities.

The Data Block by default replicate on at least three DataNodes in same or remote node

A data block default size is 64MB.

HDFS feature

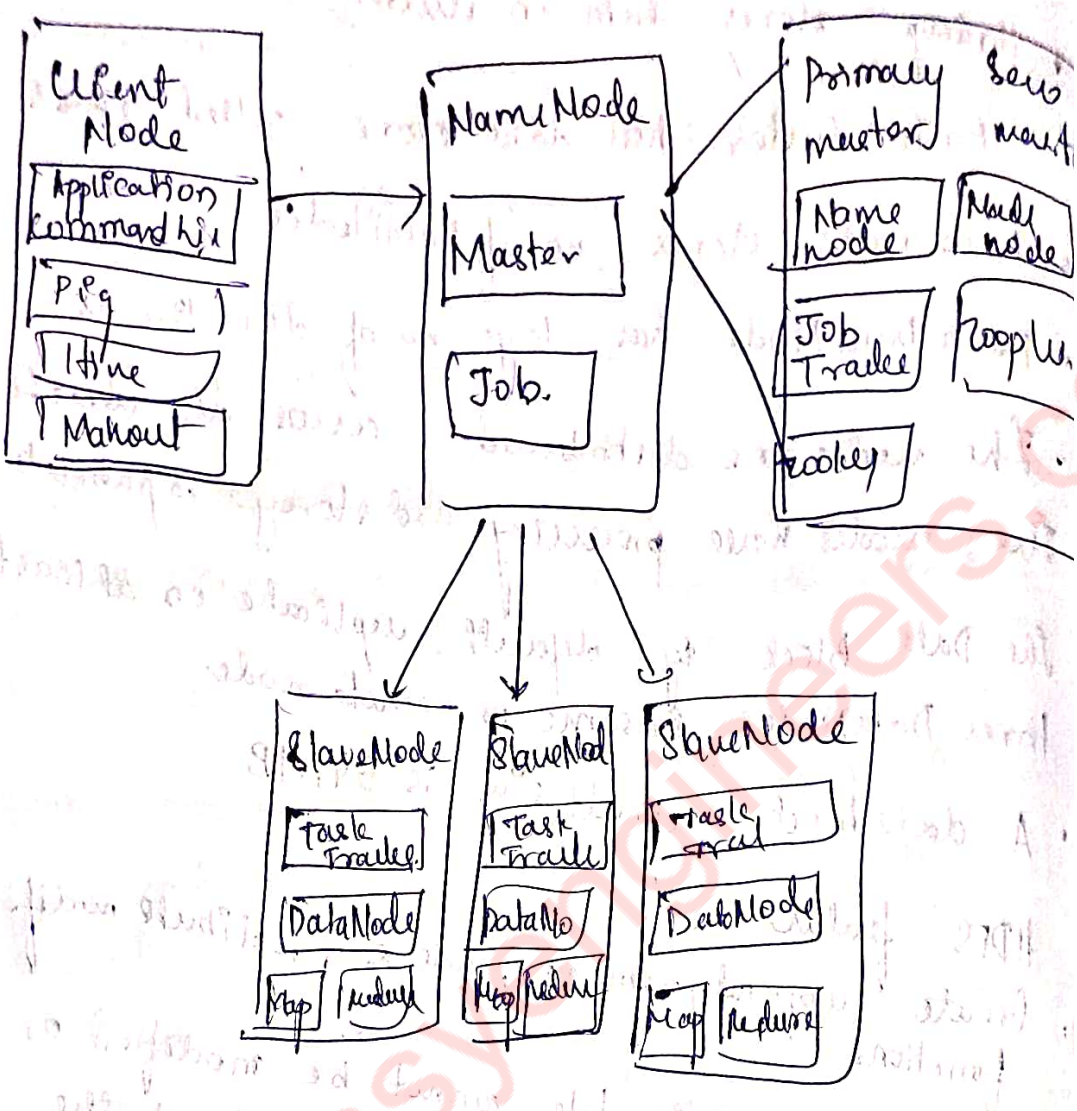
Create, append, rename, delete, attribute modify functions.

Content of individual file cannot be modified or replaced but appended with new data at the end.

Write once and read many times usage

Average file size can be more than 500MB.

(5) HADOOP PHYSICAL ORGANISATION.



Hadoop 2

⇒ Hadoop 2 has multiple name node each node with high resource availability.

- Each Master Node has following

1. An Associated NameNode
2. Zookeeper coordination client functions as a centralized repository for distribution application.

Zookeeper uses synchronization, serialization and coordination activities

5. An associated Journal Node (JN). The JN keeps the record of state, resources assigned, and intermediate result of execution of application tasks.

- It has two master nodes
- MN1 is in active state and other is standby state.
- If there is network fault in active NameNode MN1 the secondary NameNode is activated.
- Data copies from JN1 in MN1 into JN2 which is at newly active MasterNode MN2.
- Therefore application runs uninterrupted and resources are available uninterrupted.

YARN

- YARN is resource management platform, which manages compute resources.
- YARN provides the computational resources such as CPU's, memory, network I/O which are needed when an application executes.
- YARN manages the scheduling of running of subtasks.
- YARN separates resource management and processing components.
- An application consists of number of tasks. Each task can run in parallel at the nodes in the cluster.

(9) HADOOP 2 EXECUTION MODEL.

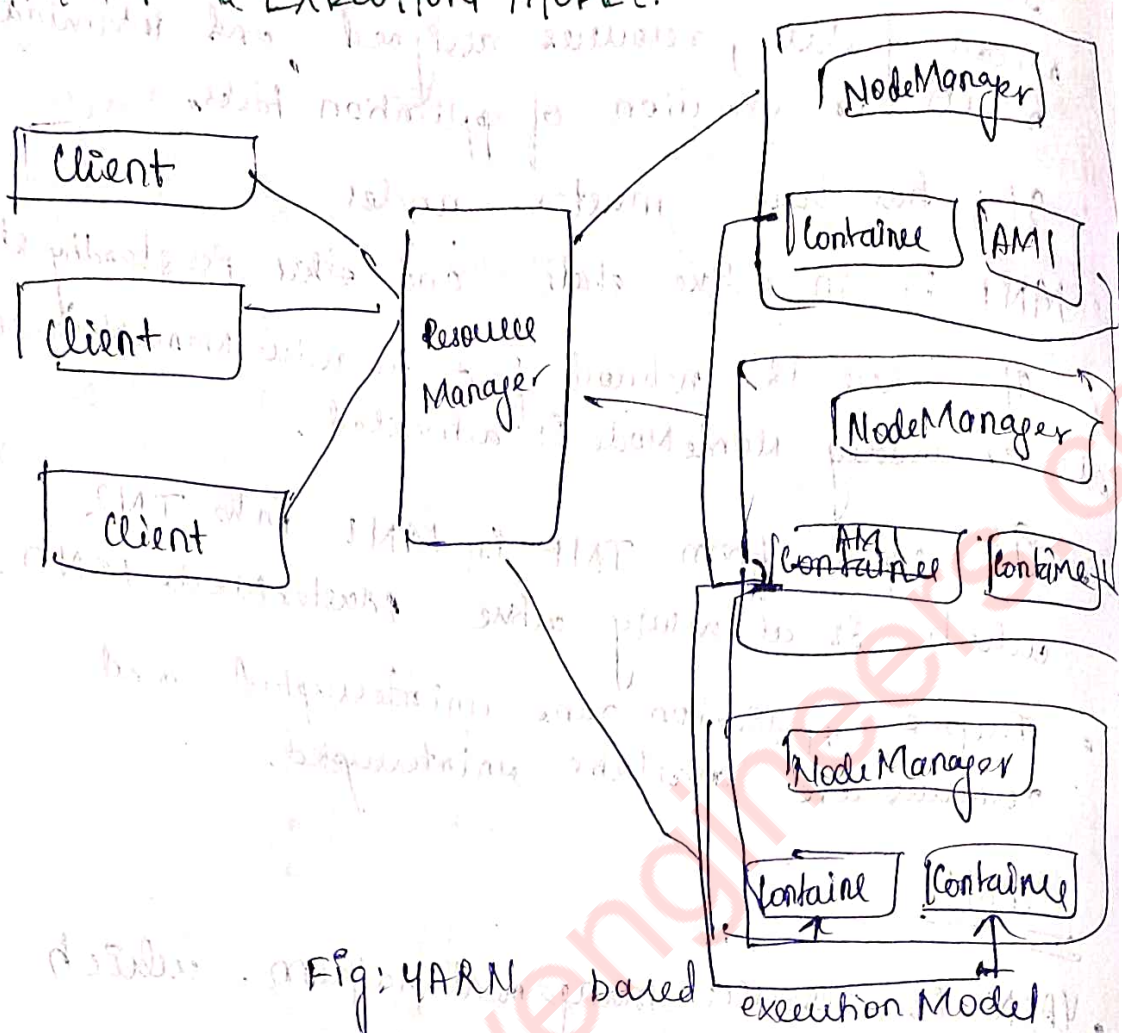


Fig: YARN based execution Model

YARN components

1) Client
2) Resource Manager
3) NodeManager.
4) Container
5) Application Manager

A Master node has 2 components 1) JobHistory server 2) Resource Manager.

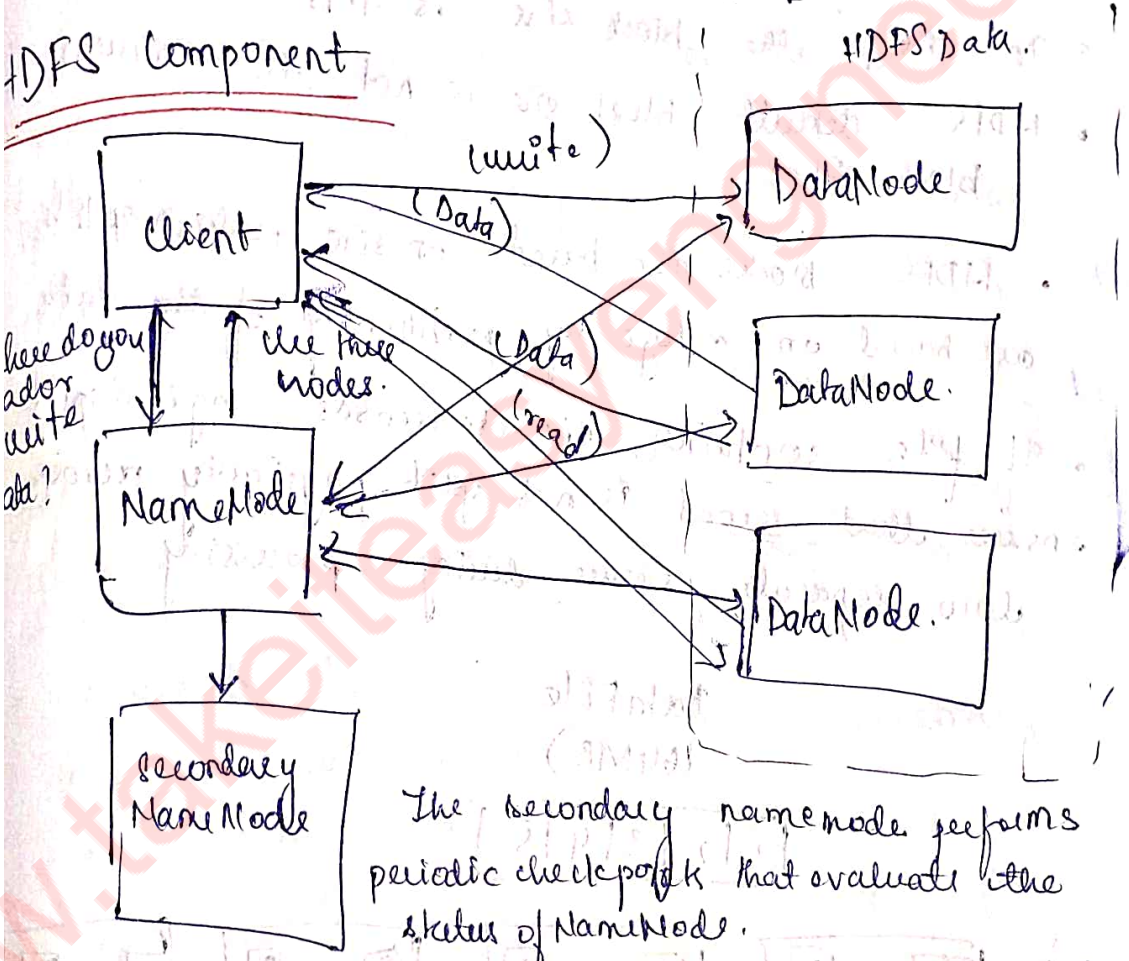
A Client Node submits the request of an application to the RM. RM is master. One RM exists per cluster. The RM keeps information slave node, location of resource. RM also runs several services that decides how to assign resources.

multiple NMs are in the cluster. AN AM creates AM instance. AM starts itself and registers with NM with AM can created in an AM.

The AM performs role of Application Manager that requests resource requirements for running an subtask. The APPMS send this request for necessary resource to the Resource Manager.

AM is slave of the infrastructure. It signals when initializes.

HDFS Component

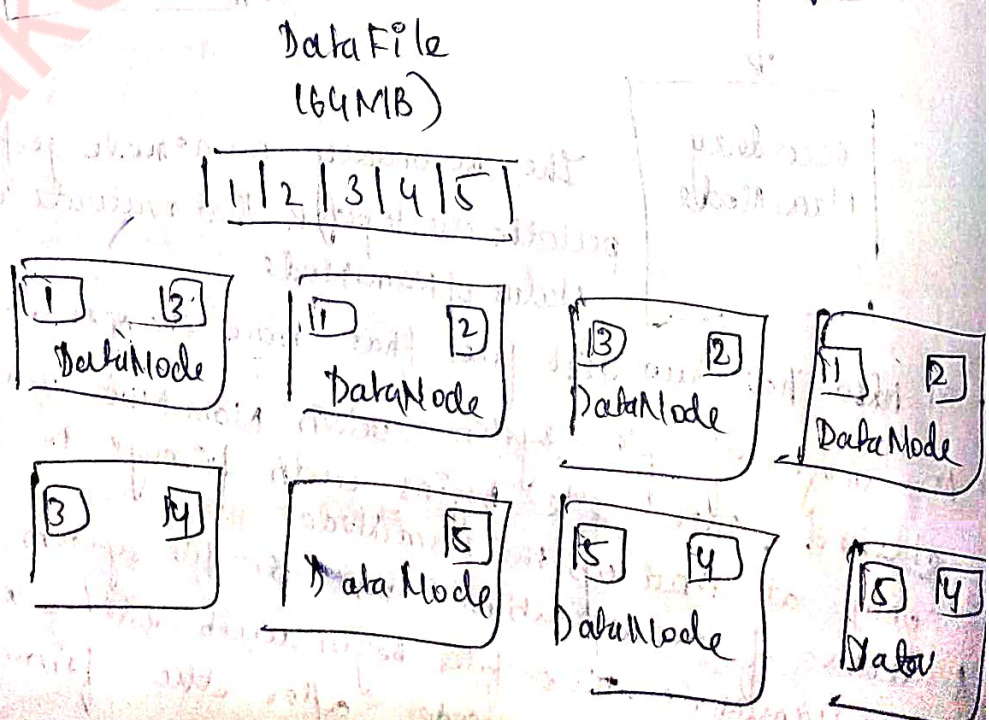


The secondary namenode performs periodic checkpoint that evaluate the status of NameNode.

- It also has two disk files that track changes to metadata
- An image of file system when NameNode was started. This file begins with fsimage-* is used only at start up the NameNode.
- A series of modification done to file system after starting NameNode this files begin with edit-* and reflect the changes made after the fsimage file was read.

HDFS BLOCK REPLICATION.

- The amount of replication is based on value dfs.replication in hdfs-site.xml file.
- The default value be changed with ~~hdfs-dfs-setrep~~ hdfs dfs -setrep command for Hadoop clusters containing more the 8 DataNodes, the replication value is usually set to 3.
  - The default block size is 64MB.
  - On OS, the block size is 4KB or 8KB.
  - HDFS default block size is not the minimum block size.
    - HDFS blocks are based on size, which splits are based on a logically portioning of the data.
    - If file contains discrete records, logical split ensures that record is not split physically across two separate server during processing.



APACHE PIG

- Apache Pig is high level language that enables to write complex Map Reduce transformations using scripting language
- pig latin is scripting language it defines set of transformation on dataset such as aggregate, join and sort.
- Pig is used to extract, transform, load data pipelines, quick research on raw data, and iterative data processing.

Pig has several modes. local mode which does processing in local machine.

The non local mode are MapReduce and Tez. These modes execute the job on cluster using MapReduce engine or optimized Tez engine.

- Interactive Mode

We can run Apache pig in interactive mode using grunt shell.

- Batch Mode

We also run Apache Pig in Batch mode, with the pig latin script file with .pig extension.

	LocalMode	MapReduce Mode	Tez Mode
Interactive Mode	Yes	Yes	Yes
Batch Mode	Yes	Yes	Yes.

Eg pig operation in HDFS mode

file -> passwd.

Step① → copy logfile passwd in HDFS file

$ hdfs dfs -put passwd input

passwd -> input

Eg 2 : $ pig -x local. (Interactive mode)

grunt > A = load 'passwd' using pigstorage(':');

grunt > B = foreach A generate $0 as id;

grunt > dump B;

$pig -x local.

$pig -x tez

$pig -x mapreduce.

USING APACHE HIVE

- Apache Hive is data warehouse infrastructure builds on top of hadoop that provides
  - Data summarization.
  - hoc queries.
  - analytics of large data set using HQL language.
- Hive offers following features
  - Tools to enable easy data extraction, transformation and loading
  - mechanism to impose structure on variety of data formats

- Access to file stored directly in HDFS or other storage system such as HBase
- Query execution via MapReduce and Tez

Hive makes it possible for programmers who are familiar with the MapReduce framework to add their custom mapper and reducers to Hive queries.

- To start Hive, $ hive
- If Hive starts correctly, we get Hive> prompt
- To exit Hive, we give the command.

hive> exit

The following Hive queries create, display and drop table.

hive > CREATE TABLE pokes (foo INT, bar STRING);

hive> SHOW TABLE

hive> DROP TABLE pokes;

APACHE SQOOP

- Sqoop is a tool designed to transfer data between Hadoop and relational database.
- Sqoop is used to
  - import data from RDBMS into HDFS
  - Transform the data in Hadoop
  - Export data back into RDMS
- Sqoop can be used with any databases that are JDBC compliant
- Eg JDBC compliant database include Microsoft SQL Server, MySQL, Oracle etc.

APACHE SQOOP IMPORT

- Sqoop examines the database to gather the necessary metadata for data to be imported.
- Sqoop examines it
- Step 2. is map only. Hadoop job that Sqoop submits to cluster. The job does actual data transfer using the metadata captured in previous step.
- Each node doing import must have access to the database. ~~The job does the actual.~~
- imported data are saved in HDFS directory
- By default file contains comma-delimited fields, new line separate different records.

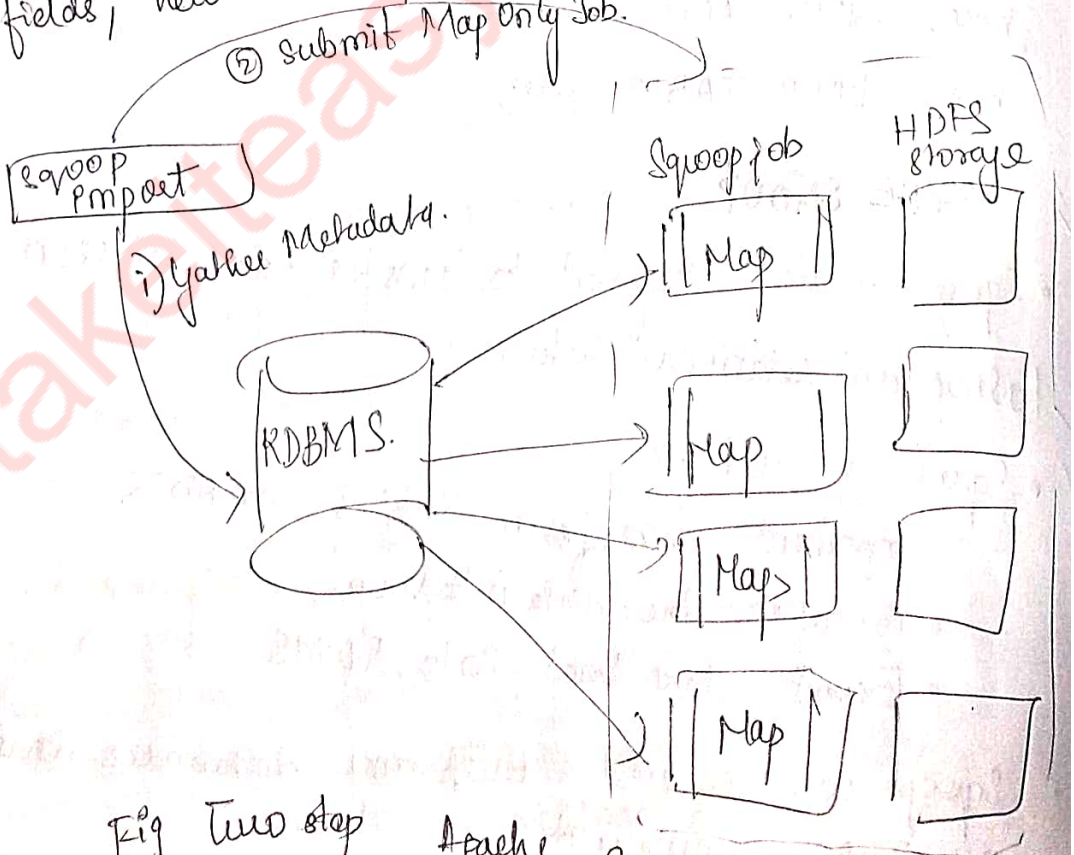


Fig Two step Apache Sqoop data import method.

The export is done in 2 steps:

Step 1: examine the database for the metadata.

The export step again uses a map-only Hadoop job to write the data to the database.

Sqoop divides the input data set into splits then uses individual map tasks to push the splits to the database. This process assumes the map tasks have access to the database

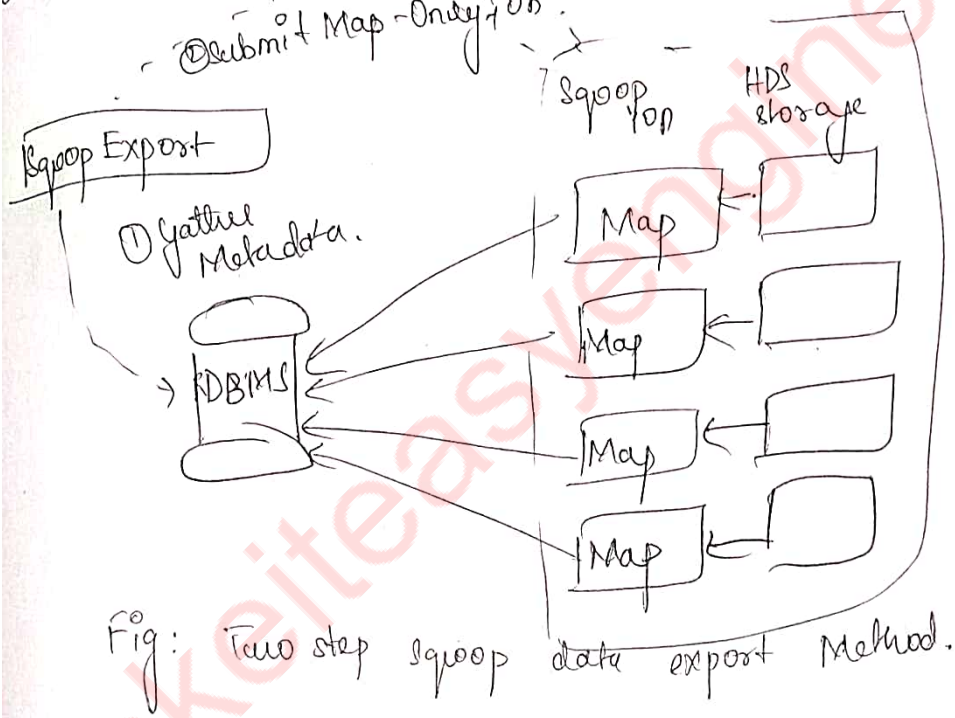


Fig: Two step Sqoop data export Method.

USING APACHE FLUME TO ACQUIRE DATA STREAMS

- Apache Flume is an independent agent designed to collect, transport, store data into HDFS.

Data transport involve no of Flume agents that may traverse a series of machine and locations.