We use Hadoop because normal Database Servers cannot handle Big data which will be in tera and Peta bytes. ETL will also cannot handle to extract big data from different resources of different data formats.

**DataWare House:-** All the data from different sources which will be in different formats will be extracted and loaded in a database is called as DataWare House. In Data Ware House Transformations and Loading will happen to get insight of the data.

To integrate any type(format) of data, cheaper, real time, scalable will be fulfilled by Hadoop. **Hadoop** is a batch Processing System.

**MapReduce** is a default programming platform for Hadoop.

**Cloudera** is a commercial distributor for hadoop and related tools.

**Distributed Systems:-** When you install Hadoop on four servers we have to select master and slaves. Here consider if we want to save data in slaves. If there are three slaves and one master. If each slave is 2TB, Then Hadoop will consider three slaves as one Server and shows as 6TB Disk. If we need more slaves, Horizontal scaling is available.

Three Components we get when we install Hadoop. 1.Yarn – Resource Manager,

2.Hadoop Distributed File System(HDFS) – Storage: You can store any format of data. Modification is not possible. Storing and Deleting is only possible. Only Sequential access is available, No Random Access.

3.MapReduce - Processing.

**Master Node** = Name Node, **Slave Node** = Data Node

We can connect to hadoop cluster by two ways. 1. Hadoop client(Rare cond) 2. Gateway node.

When we upload a file in Hadoop Cluster Hdfs will tell us about the block size and Gateway node where Hadoop client already in it, it will divide the file according to block size and send those to Master node. No Master node will save those in different slave.

On Hadoop 2, we can mention how many replicas our data can have while saving in Hadoop Cluster. In Hadoop 3, No need of replication.

**Oozie(Job Scheduler):-** To schedule our batch jobs to run Oozie will come in to the place.

**Queue :-**  To give % of resources to Different employees.Developers nedd to run batch job we will give 60% resource to them. Tester want to test we can allot 25% resources.

If you have more than 5000 slave nodes we should add one more Name Nodes(Master Node). Adding a Master Node is called as Federation.

**Sqoop:-** To extract SQL stores from different sources to Hadoop can be done by Sqoop.

**FLUME:-** To extract structured or unstructured data from different sources to Hadoop in realtime. Flume reads the data from different sources and dumps into HDFS.

Flume uses pull method where you don’t need to install any thing in source to get the data, With Kafka we have to install kafka publisher in source. So best thing is Connect all dbs to Flume and Flume to Kafka and kafka to HDFS.

**KAFKA:-** Its message Queue in Big Data World. Kafka itself is a cluster which will safe data up to 7 days predefined, We can change that if we want. And for kafka there can be n number of subscribers. But flume can have only one subscriber.

From Kafka we can send to streaming spark where we can do real time analysis and update the customer **Ex:-** if credit card is swiped we can send a message.. There are other real time streaming tools like (FLUNK/STORM). **Spark streaming** will do batch processing for 2 seconds data it will not do processing on single transaction. **Storm or Flink** can do single transaction processing.

**MapReduce:** To do batch processing on HDFS we use MapReduce.

**Hive:** To write sql on top of Hadoop.

**IMPALA:** Multi Parallel Processing tool where we can search one record randomly using Impala on HDFS storage. But it is **not fault tolerant**. **HIVE,SparkSql** is **fault tolerant.**

**HBASE:** It is a Nosql database of hadoop. Hbase install on Hadoop which can do random read and write records. It has his own language.

If you want to run program on data, data will be in n number of nodes in the cluster. So we have MapReduce framework or program in Hadoop where mapper program will be executed in every node and the result set in each node will be collected by reducer in one node and consolidate all the node results and will give o/p as single.