## What is Hadoop?

* Allows distributed storage (HDFS) & processing (MapReduce) of data across cluster of commodity servers
* It is fault tolerant & reliable (Replicates data in its node)
* It has a processing framework called MapReduce
* HDFS converts a file into separate blocks of data and stores across nodes
* HDFS has two types of process (Name Node, Data Node / Salve nodes)
* Name Node : Stores the meta data (Data about Data such as name of file, permission of the files ,timestamp, no. of blocks file created, file to block mapping, block to data node mapping)
* Name node acts a point of entry into Hadoop
* Data Node : Stores actual data
* Typically a Hadoop system has 2 or 3 name nodes and 100, 1000 of data nodes
* Hive, Pig, Mahout and MapReduce are batch oriented, so we need a JVM for each MapReduce process
* Hadoop supports shell likes commands
* Hive runs on top of HDFS
* Hive uses HQL, syntax like SQL
* Hibernate Query Language (HQL) is an object-oriented query language, similar to SQL, but instead of operating on tables and columns, HQL works with persistent objects and their properties. HQL queries are translated by Hibernate into conventional SQL queries which in turns perform action on database.
* MapReduce is replaced by Spark

## What is HDFS

* Data in a Hadoop cluster is broken down into smaller pieces called blocks and are distributed throughout the cluster. This enables the Map and Reduce programs to execute in parallel with smaller subset of data. This provides the scalability
* The goal of Hadoop is to use commonly available servers
* For higher performance, MapReduce assign the task to the servers where the data to be processed is stored. This is called Data locality
* It has built in fault tolerance
* To maintain availability when disk fails, HDFS replicates the data blocks onto additional servers. The default is 2 additional copes and can be increased or decreased on a profile basis or for the whole environment

## Benefits of Hadoop

* Linear scale without extreme price tag
* Lots of flexibility – You can always change your data ingest pipeline or data models with low impact
* Ability to combine and analyze previously soiled data sets
* Open door to expand business

## Hadoop Modules

* HDFS – A distributed file system
* Hadoop Common – Common utilities that support other Hadoop modules
* Hadoop YARN (Yet Another Resource Negotiator) – A framework for job scheduling and cluster resource management
* Hadoop MapReduce – A YARN based system used for parallel processing of large data sets

## Various Hadoop Distributions

* Cloudera
* Hortonworks
* MapR

## Cloudera Hadoop Distribution (CDH)

* Most popular distribution of Hadoop
* Founded by folks from Yahoo, Google, Facebook
* Contain automation, Reliable Vendor support exist
* CDH Includes
  + Core elements of Hadoop (HDFS & MapReduce)
  + Cloudera Search (For implementing search engine)
  + Storage for any type of data
  + Data Integration support (Sqoop, Flume, NFS)
  + NoSQL Database (HBase)
  + Batch Processing (MapReduce, Hive, Pig)
  + Machine Learning (Spark, MapReduce & Mahout)
  + Stream Processing of Data (Spark)
  + Analytic SQL (Impala)
  + YARN
  + Cloudera Manager Console (UI for installing, configuring, integrating, managing etc ), Also used to configure clusters

## Hartonworks

* Complete open source distribution
* It gives all features given by Cloudera and also extends
* Integrated with Microsoft data insights
* Integrated with IBM big insight solution & Google cloud

## MapR Distrubution

* It uses its own properatory file system
* Used for Batch and Real-time applications
* It comes with MapR Database (Multi-model database)
* Can use JASON based APIs
* Provides direct access to file system with interoperability
* Written in C, so there is no Java Garbage collection
* Can bring files directly to the distribution

## Hartonworks and MapR?

Dd

## How to choose Hadoop Distribution?