**Hadoop**

1.Mention Hadoop distribution? Difference between CDH and CDP

* Cloudera distribution of Apache Hadoop
* Hortonworks data platform
* MapR

Hadoop, as an open-source framework for distributed storage and processing of large data sets, has various distributions provided by different vendors.

Two popular distributions are Cloudera's Distribution including Apache Hadoop (CDH) and Cloudera Data Platform (CDP).

Cloudera Data Platform (CDP)- it has cloudera manager which manages of the platform across various environments(on prem and cloud)

2.Explain Hadoop Architecture

Hadoop was the first framework designed to solve Big Data problems. It is a framework because It is not just a single tool but a combination / ecosystem of several tools and technologies to solve Big Data problems.

It has 3 core components

HDFS-storage

1.Name Node -

Name node is the Master node that acts like an index to where the actual data is retained. It holds the Metadata table that keeps track of what is stored where.

Request from a Client (to read a file) -> Namenode -> Name node provides the actual location details of the File -> Client uses this data to fetch the file from the Data Node for reading.

How does Namenode know if Datanode is alive?

• Every Datanode sends a heartbeat to Namenode every 3 seconds indicating that it's alive.

• If the Namenode does not receive a heartbeat for 30 seconds, then Namenode assumes Datanode is dead.

What if a Namenode fails?

• Namenode is a good quality hardware and chances of going down is very less.

• But if it goes down, we have a secondary Name node to tackle this situation.

2.Mapreduce-distributed processing

3.YARN-resource manager

The ecosystem tools are

Sqoop(ingestion from RDBMS to HDFS and vice versa), hive(Datawarehouse), HBase(NoSQL database), pig(data cleaning), oozie(workflow scheduler)

**3.Configuration files used during hadoop installation**

During the installation and setup of Hadoop, several key configuration files are utilized to define parameters and settings that govern the behavior of Hadoop components. These configuration files include:

core-site.xml: Defines core configuration settings such as the default file system (fs.defaultFS) and Hadoop temporary directories (hadoop.tmp.dir).

hdfs-site.xml: Configures Hadoop Distributed File System (HDFS) properties like block replication (dfs.replication), block size (dfs.blocksize), and data node directories (dfs.datanode.data.dir).

mapred-site.xml (Hadoop 1.x) or mapred-site.xml (Hadoop 2.x+): Configures MapReduce framework settings including job tracker address (mapred.job.tracker) and task slots.

yarn-site.xml: Configures Apache YARN settings such as ResourceManager hostname (yarn.resourcemanager.hostname) and NodeManager resource allocation.

**4.Difference between Hadoop fs and hdfs dfs**

hadoop fs is associated with Hadoop 1.x, while hdfs dfs is associated with Hadoop 2.x and later versions.

hadoop fs is more generic and can be used to interact with different file systems supported by Hadoop.

hdfs dfs is specific to HDFS and provides commands tailored for HDFS operations.

**5.Difference between Hadoop 2 and Hadoop 3**

Hadoop 2 introduced YARN and NameNode HA, significantly improving the capabilities and scalability of the Hadoop framework.

Hadoop 3 builds upon the foundation laid by Hadoop 2 and introduces additional features like erasure coding, enhanced scalability, improved resource utilization, and security enhancements

**6.What is replication factor ? why its important**

It is for fault tolerance. What if the data node fails. Replication factor helps in having a backup stored on other Data Nodes and when the current data node goes down, the data can be accessed from the backup nodes

**7.What if Datanode fails?**

Replication factor helps in having a backup stored on other Data Nodes and when the current data node goes down, the data can be accessed from the backup nodes

**8.What if Namenode fails?**

Secondary Name node will come into picture to keep the system up and running.

Master node connects with zookeeper

Zookeeper follows leader-follower rule

Followers job is to keep themsleves synced with leader

Whe4n NN fails leader will become the name node automatically

One name node will be leader and all the other name nodes will be the followers

Write will happen only through the leader

**9.Why is block size 128 MB ? what if I increase or decrease the block size**

A balanced approach of 128 mb is considered to get the best performance.

**Decreasing block size**

Advantage : leads to an increase in the number of blocks and thereby increases parallelism

Drawback : The Name Node gets overburdened with having to store many entries in its table

**Increasing block size**

Advantage : Name node is less burdened as it needn't have to store too many entries in the metadata table.

Drawback : Compromising on Parallelism. Less number of nodes implies less parallelism.

**10.Small file problem**

The small file problem in Hadoop arises when a large number of small files are stored, leading to increased metadata overhead and inefficiencies in data processing. To mitigate this issue, consolidate small files into larger ones, use optimized file formats like SequenceFile or Avro, employ Hadoop Archives (HAR) for bundling files, and batch small writes to reduce the number of individual files stored in HDFS. These strategies help improve NameNode scalability, reduce metadata operations, and enhance overall performance of Hadoop clusters.

**11.What is Rack awareness?**

Rack is a group of servers in different geographical locations

It is the group of servers placed in different geographical locations. It is spread across different geographic locations, so that if there is a natural calamity , you don’t lose your data. The balanced approach is to place replicas(copies) in two different racks, that is, one rack will have 1 copy and the other rack will have 2 copies of the data

**12.What is SPOF ? how its resolved ?**

In the context of Hadoop, addressing Single Points of Failure (SPOF) is essential to ensure the reliability and availability of the Hadoop cluster. Hadoop, being a distributed system, consists of multiple components that work together to store and process large volumes of data. Single points of failure within a Hadoop cluster can lead to downtime, data loss, or degraded performance. Here's how SPOF is typically addressed in a Hadoop environment:

1. NameNode High Availability (HDFS):

2. ResourceManager High Availability (YARN):

3. DataNode and TaskTracker Redundancy:

4. Network and Hardware Redundancy:

5. Monitoring and Alerting:

6. Disaster Recovery Planning:

**13.Explain zookeeper?**

If Name node failure occurs, The following name nodes will be available

Secondary NN

Journal node 1

Journal node 2

Journal node 3

Journal node 4

Journal node 5

Master node connects with zookeeper

Zookeeper follows leader-follower rule

Followers job is to keep themsleves synced with leader

Whe4n NN fails leader will become the name node automatically

One name node will be leader and all the other name nodes will be the followers

Write will happen only through the leader

**14.Difference between -put and -CopyFromLocal?**

Both commands copies the file from local to hdfs

However, copyfromlocal explicitly indicates that we are copying the file from local file system to HDFS.

**15.What is erasure coding?**

erasure coding is a technique used to enhance data reliability and efficiency in distributed storage systems by generating redundant encoded pieces of data that enable recovery from data loss or corruption without storing multiple complete copies of the data. It provides a balance between storage efficiency, fault tolerance, and computational overhead in large-scale storage environments.

**16.What is speculative execution?**

speculative execution in Hadoop is a strategy to mitigate the impact of slow tasks by running duplicate tasks in parallel and using the result of the fastest task to determine the final outcome. This technique helps improve job completion times and overall system efficiency in distributed computing environments.

**17.Explain Yarn Architecture**

YARN consists of two major components

1. Resource Manager (Master)

2. Node Manage (Worker / Slave)

On executing the job on the client machine, following things happen :

1. Request goes to Resource Manager

2. Resource Manager creates a Container in one of the Node Managers (by coordinating with the Node Manager)

3. An Application Master Service will be started inside this Container.

**18.How does Application Manager and Application Master differ**

There is only one Application Manager per YARN cluster, responsible for managing all applications running on the cluster.

Its primary responsibility is to manage the lifecycle of an application within the YARN framework.

Each application running on YARN has its own isolated instance of an Application Master.

(Application Master is a local Manager that manages the application) Note : Application Master is responsible to negotiate for the required resources from the Resource Manager. It will interact with the Name Node to know where the Data is located and accordingly request for resources on specific nodes as it works on the principle of Data Locality. Every Application has an Application Master i.e., If there are 20 Applications, then there would be 20 Application Masters running.

Note : Every Spark Job has one Driver, Application Master acts like a driver which gets registered with the Resource Manager. If the Driver goes down, then the application crashes.

**19.Explain Mapreduce working?**

It has 2 phase-map and reduce

Principal of data locality Is the data going to code or code is going to the data?? Code is generally small but data is big. So, the principle of data locality means the data is processed on the same machine where it is kept.

- The output of mapper is not the final output , but it is an intermediate output.

- The output of all the mappers go to one other machine (it can be one of the DN), where the reduce activity takes place.

- Mapper will give you parallelism.

-Reduce is for aggregation

Map------combiner-----shuffle----sort-----reducter

**20.How many mappers are created for 1 GB file?**

Number of Mappers = Number of blocks. So 8 mappers will be created

but

Number of Mappers Running in Parallel = Number of Data Nodes in the Cluster

**21.How many reducers are created for 1 GB file?**

No.of Reducers Launched = Can be Configured to a desired number (Default = 1 Reducer)

Case 1 : When to increase the number of Reducers - To avoid the bottleneck due to a single reducer. If the reducer has to do a lot of aggregation, then a single reducer might become a bottleneck and we would increase the number of reducers.

Case 2 : When to decrease the number of Reducers to Zero - For the jobs that don't require any Aggregation and Mapper output is the final output. Ex - Filter

**22.What is combiner?**

Advantage of introducing Local Aggregation at Mapper - Combiner

1. Improve Parallelism

2. Reduce Data Transfer

Use it with caution

**23. What is partitioner?**

PARTITION - Is for distributing the Data that is in the form of (Key,Value) pairs based on some logic across different Reducers.

No.of Partitions = No.of Reducers

The output [Key,Value] from the Mapper with a specific key always goes to a specific Reducer based on a logic as follow - 1. Default System Defined Hash Function - Ex : Mod (%) function 2. Custom Function