1.Mention Hadoop distribution? Difference between CDH and CDP

Cloudera Distribution of Hadoop (CDH) and Cloudera Data Platform (CDP) are both Hadoop distributions, but they have some key differences:

Cloudera Distribution of Hadoop (CDH):

CDH is a traditional Hadoop distribution that includes Apache Hadoop, Hive, Spark, Pig, and other tools.

CDH has a fixed-size data hub. Cloudera is moving away from Hadoop.

Cloudera Data Platform (CDP):

CDP, on the other hand, is a more comprehensive data platform that includes separate services for Data Engineering, Machine Learning, Data Warehouse, and more, all using one data governance solution.

CDP is based on having multiple Kubernetes clusters that can automatically scale up and down.

CDP is the future of Cloudera, CDP provides the benefit of dealing with one vendor, but it may not have the best possible solution for all workloads.

In summary, CDH is a traditional Hadoop distribution with a focus on HDFS, MapReduce, and related tools, while CDP is a more comprehensive data platform that includes separate services for different aspects of data processing and management. CDP is designed to be more scalable and flexible, while CDH is more focused on providing a traditional Hadoop stack.

2.Explain Hadoop Architecture

Hadoop is an open-source framework used for storing, processing, and analyzing large amounts of data. The key components of the Hadoop architecture are:

Hadoop Distributed File System (HDFS): HDFS is the distributed file system that provides high-throughput access to application data. It stores data across multiple machines, providing fault tolerance and high availability. HDFS has a master-slave architecture, with a NameNode as the master and DataNodes as the slaves.

MapReduce: MapReduce is the programming model and software framework for writing applications that rapidly process vast amounts of data in parallel on large clusters of commodity hardware. It consists of the Map and Reduce phases.

YARN (Yet Another Resource Negotiator): YARN is the resource management and job scheduling component of Hadoop. It is responsible for allocating resources to the various applications running in a Hadoop cluster and scheduling tasks.

Hadoop Common: Hadoop Common provides the common utilities and libraries that support the other Hadoop modules. It includes the Java libraries and utilities required by other Hadoop modules.

The Hadoop architecture is designed to be fault-tolerant, scalable, and able to handle large datasets. The NameNode manages the file system namespace and access to files, while the DataNodes store and retrieve data. MapReduce processes data in parallel, and YARN manages the resources and scheduling of the cluster. Together, these components enable Hadoop to efficiently store, process, and analyze large amounts of data.

3. Configuration files used during Hadoop installation

The key configuration files used during Hadoop installation are:

Core-site.xml, hdfs-site.xml, mapred-site.xml, yarn-site.xml, workers (or slaves), log4j.properties

4. Difference between Hadoop fs and hdfs dfs

Hadoop fs is a generic file system abstraction layer that can work with various file systems like local file system, HDFS, Amazon S3, Azure Blob Storage, etc. HDFS dfs, on the other hand, is the specific implementation of the distributed file system within the Hadoop ecosystem, designed for storing and processing large datasets in a distributed manner across a cluster of machines.

5.Difference between Hadoop 2 and Hadoop 3

Hadoop 2 introduced significant changes with the addition of YARN, Hadoop 3 focuses on improving scalability, efficiency, and support for new hardware and deployment models, making it better suited for modern data processing workloads and infrastructure. The split YARN architecture, Erasure Coding, GPU support, better container integration, and performance optimizations are major enhancements in Hadoop 3.

6.What is replication factor ? why its important

Replication:

-each file is split into blocks of fixed size (64/128MB etc)(size is not fixed. it could be changed)

-each block is replicated (3replicas by default)( replicas can change. we can have 2 replicas / 4 replicas)

-these replicas are distributed to different machines in the cluster

Advantages of having replicas:

-if a replica on a machine is lost or fails it's not a big problem because the same copy is in another machine, so that we can use that to complete our process.

-having different replicas (copies) means multiple machine failures are easily tolerated

-having different replicas means several versions of data are available for reading, so a data block can be read that is closest to an application on the network, which in turn speeds up processing.

-HDFS is designed to track and manage the number of available replicas of a block, so if the no of copies of a block drops bcoz of failure, the file system automatically makes a new copy from one of the remaining replicas.

7.What if Datanode fails?

HDFS is designed to handle DataNode failures gracefully by maintaining multiple replicas of data blocks. When a DataNode fails, the NameNode detects the failure, initiates replication of under-replicated blocks, and ensures data availability as long as at least one replica exists. The system automatically recovers and rebalances the cluster once the failed DataNode is repaired or replaced

8.What if Namenode fails?

The High available Hadoop cluster also has 2 or more than two Name Node i.e. Active NameNode and Passive NameNode also known as stand by NameNode. In case if Active NameNode fails then the Passive node will take the responsibility of Active Node and provide the same data as that of Active NameNode which can easily be utilized by the user.

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

The default block size of 128 MB is a balance between performance, storage efficiency, and typical hardware configurations. Increasing the block size can improve throughput for large sequential workloads but may lead to storage inefficiency, while decreasing the block size can benefit workloads with many small files or random access patterns but may increase metadata overhead and seek operations. The optimal block size depends on the specific workload characteristics and should be carefully evaluated based on the trade-offs involved.

10.Small file problem

Having lots of small files will result in lots of seeks and lots of movement from one datanode to another datanode to retrieve each small file, this whole process is a very inefficient data access pattern.

11.What is Rack awareness?

Rack awareness in Hadoop ensures that data replicas are strategically placed across racks in a cluster. This optimizes fault tolerance and network efficiency by minimizing data transfer across racks and leveraging local rack communication for faster performance. The NameNode maintains information about rack locations to facilitate data placement and retrieval from the nearest nodes.

12.What is SPOF ? how its resolved ?

The NameNode is crucial in Hadoop, acting as the master node. In older versions, its failure caused system-wide failure due to a single point of failure (SPOF). However, in later versions, this issue was addressed by introducing HDFS High Availability with two NameNodes (active and standby), preventing system downtime if one NameNode fails. Data remains intact on DataNodes even if the NameNode fails, but access to file metadata requires recovery techniques using the standby NameNode.

13.Explain zookeeper?

There was a huge issue of management of coordination and synchronization among the resources or the components of Hadoop which resulted in inconsistency, often. Zookeeper overcame all the problems by performing synchronization, inter-component based communication, grouping, and maintenance.

ZooKeeper is often used for coordinating activities across various Hadoop ecosystem components such as HDFS, YARN, and distributed computing frameworks like Apache Spark or Apache Kafka. It provides a reliable foundation for managing distributed coordination challenges, ensuring consistency and reliability across the distributed system.

14.Difference between -put and -CopyFromLocal?

Both the –put (older versions Hadoop 1) and -copyFromLocal (newer versions Hadoop 2) commands are used to copy files from the local file system to HDFS (Hadoop Distributed File System).

15.What is erasure coding?

Erasure coding is a fault tolerance technique used in Hadoop Distributed File System (HDFS). It's a form of RAID technology that reconstructs data in case of disk failures. Hadoop 3 introduced erasure coding, which requires less storage compared to Hadoop 2's replication mechanism, yet maintains the same fault tolerance level. This efficiency is crucial for handling increasing data volumes, as it significantly reduces storage requirements while improving fault tolerance by up to 50%.

16.What is speculative execution?

Speculative execution is a proactive approach to handling slow tasks in distributed computing systems by running duplicate tasks concurrently. This technique optimizes job completion times and enhances system performance in large-scale data processing frameworks like Hadoop.

17.Explain Yarn Architecture

YARN (Yet Another Resource Negotiator) is the resource management layer of Apache Hadoop. It consists of two main components: the ResourceManager (RM) and the NodeManager (NM).

ResourceManager: Manages resources across the cluster, including scheduling and allocation to applications.

NodeManager: Manages resources on individual cluster nodes and executes application tasks within containers.

Applications running on YARN have their own ApplicationMaster, responsible for negotiating resources, managing task execution, and monitoring progress. Containers provide isolation and resource enforcement for tasks. YARN enables multiple data processing frameworks to run concurrently on the same infrastructure, improving resource utilization and cluster efficiency.

18.How does ApplicationManager and Application Master differ

Both the ApplicationManager and the ApplicationMaster are involved in managing applications within YARN, the ApplicationManager is a component of the ResourceManager responsible for managing application lifecycles at a high level, while the ApplicationMaster is a framework-specific component responsible for managing the execution of individual applications and coordinating tasks within YARN.

19.Explain Mapreduce working?

Map: Input data is divided and processed by multiple mapper nodes, with each mapper applying the map function to its local data and producing intermediate key-value pairs.

Shuffle, Combine, and Partition: Intermediate data is shuffled and sorted based on keys, and optionally, combiners (reducers) may run on mappers to reduce data volume. Partitioning decides how data is presented to reducers and assigns it to specific reducers.

Reduce: Reducers process groups of key-value pairs in parallel, aggregating values with the same key and producing final output. Unlike mapping, reducing is optional.

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

For a 1 GB file in Hadoop with a default block size of 128 MB, there would be approximately 8 mappers created, as each block is processed by a separate mapper.

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

The number of reducers created for a 1 GB file in a MapReduce job depends on various factors such as cluster configuration and job settings. By default, Hadoop may use a smaller number of reducers, such as 1 or 2, but this can vary based on cluster resources and data distribution.

22.What is combiner?

A combiner is a function used in MapReduce to perform local aggregation of intermediate data on mapper nodes, reducing the amount of data transferred between map and reduce tasks and improving overall performance.

23.What is partitioner?

A partitioner in Hadoop's MapReduce framework is responsible for partitioning the output key-value pairs from the mappers and ensuring that related data with the same key ends up at the same reducer for efficient processing. It plays a crucial role in workload distribution and parallel processing in MapReduce jobs.