

1. What is the main purpose of Name Node in HDFS? How does a Name Node try to avoid a single point of failure issue?

- Main purpose of Name Node in HDFS is a master node that keeps track of which blocks make up a file, and where these tracks are located
- High availability mode: this helps namenode avoid a single point of failure issue (if namenode fails, Hadoop cluster is inaccessible or out of service), by have two name nodes available. One is an active one and one is a standby one. In classic mode, there is one main name node and a secondary name node (as a helper and bookkeeping, not a backup node) and can be called by the master if the master namenode fails.

2. What is speculative execution (also called backup tasks)? What problem does it solve?

- It is when Hadoop creates backup tasks on a different machine when a task (usually a Mapper) is detected as being slow or a task attempt fails. Depending on which task is faster, then Hadoop will delete the one that is slower.
- Resolves the problem of a bottleneck created by Hadoop only transferring the data only once the Mappers have finished. Reducers won't start until it gets all the intermediate data from Mappers. This happens when:
  - Unequal sized hardware – Its recommended that all the nodes in the network are equal sized. however, if the hardware is not sized equally causes delays in execution of some of the tasks and the framework kick starts the parallel running of the task.
  - Network congestion
  - Faulty hardware
- The goal of Speculative execution is reducing the execution time, but not at the cost of cluster efficiency. In a Busy Cluster, this will result in reducing the overall output.

3. You have a web log file (140MB) and want to store it in HDFS. Your Hadoop system has 10 slave nodes. Illustrate how this file is stored in this system. And explain the steps how this file can be accessed later.

- The 140 mb log file will be divided into multiple blocks of either 128 or 64 MB depending on version the last block will not be of full size. For example if we use version 1 than 3 blocks will be created, 2x64 MB and 1 with 12 MB. If we use version 2 then it'll be 128MB and 12MB (2 blocks)
- Then each block is replicate 3 times in the 10 nodes and NameNode will be storing the meta data
- When a client submits tasks, JobTracker will be the one coordinating the tasks (MR) and assign them to tasktrackers at the slave node level, determining the location of the data
- TaskTracker will then monitor individual M and R tasks and report back to client
- ResourceManager, on the other hand, will launch application masters to assign to slave nodes

4. What is the role of Jobtracker in MapReduce?

- Manages MapReduce jobs and distributes tasks to tasktrackers (one per slave note)
- JobTracker is an essential Daemon for MapReduce. It can also be Resource Manager depending on the installation

- It receives the requests for MapReduce execution from the client, then talks to the NameNode to determine the location of the data.
- JobTracker finds the best TaskTracker nodes to execute tasks based on the data locality (proximity of the data) and the available slots to execute a task on a given node.
- Its monitors the individual TaskTrackers and the submits back the overall status of the job back to the client.
- JobTracker process is critical to the Hadoop cluster in terms of MapReduce execution.
- When the JobTracker is down, HDFS will still be functional but the MapReduce execution cannot be started and the existing MapReduce jobs will be halted.

5. Explain how HBase distributes its table over multiple machines.

- By using column families, this allows data separation.
- It segregates information by access patterns
- HBase tables are divided horizontally by row key range into “Regions.” A region contains all rows in the table between the region’s start key and end key. Regions are assigned to the nodes in the cluster, called “Region Servers,”
- Zookeeper is the coordinator that does the maintaining server state in the cluster.

6. What is sparse data and how HBase is handling sparse data?

- Sparse data is when a table has many columns but there are also a lot of missing values in each column (gaps in data being recorded)
- Sparse data can be filled up overtime (i.e. patient records)
- Since HBase is a scheme-less system, it allows us to add fields dynamically or columns in HBase vernacular overtime without having to disrupt operations and redesign schema
- It allows random reads and writes