edureka!

- [Blog Home](#)
- [Webinars](#)
- [Courses »](#)
- [Interview Questions](#)

Hadoop Interview Questions – HDFS

April 23, 2013 | [Big Data and Hadoop](#), [Interview Questions](#)

1

Tweet

28

Like

3

Share

5

g+1

0

Email

Print

Looking out for Hadoop Interview Questions that are frequently asked by employers? Here is the first list of Hadoop Interview Questions which covers HDFS…

What is BIG DATA?

Big Data is nothing but an assortment of such a huge and complex data that it becomes very tedious to capture, store, process, retrieve and analyze it with the help of on-hand database management tools or traditional data processing techniques. To know more about BIG DATA, browse through **The Hype Behind Big Data!**

Can you give some examples of Big Data?

There are many real life examples of Big Data! Facebook is generating 500+ terabytes of data per day, NYSE (New York Stock Exchange) generates about 1 terabyte of new trade data per day, a jet airline collects 10 terabytes of censor data for every 30 minutes of flying time. All these are day to day examples of

Big Data!

Can you give a detailed overview about the Big Data being generated by Facebook?

As of December 31, 2012, there are 1.06 billion monthly active users on facebook and 680 million mobile users. On an average, 3.2 billion likes and comments are posted every day on Facebook. 72% of web audience is on Facebook. And why not! There are so many activities going on facebook from wall posts, sharing images, videos, writing comments and liking posts, etc. In fact, Facebook started using Hadoop in mid-2009 and was one of the initial users of Hadoop.

According to IBM, what are the three characteristics of Big Data?

According to IBM, the three characteristics of Big Data are: **Volume:** Facebook generating 500+ terabytes of data per day**. Velocity:** Analyzing 2 million records each day to identify the reason for losses. **Variety:** images, audio, video, sensor data, log files, etc.

How Big is 'Big Data'?

With time, data volume is growing exponentially. Earlier we used to talk about Megabytes or Gigabytes. But time has arrived when we talk about data volume in terms of terabytes, petabytes and also zettabytes! Global data volume was around 1.8ZB in 2011 and is expected to be 7.9ZB in 2015. It is also known that the global information doubles in every two years!

How analysis of Big Data is useful for organizations?

Effective analysis of Big Data provides a lot of business advantage as organizations will learn which areas to focus on and which areas are less important. Big data analysis provides some early key indicators that can prevent the company from a huge loss or help in grasping a great opportunity with open hands! A precise analysis of Big Data helps in decision making! For instance, nowadays people rely so much on Facebook and Twitter before buying any product or service. All thanks to the Big Data explosion.

Who are 'Data Scientists'?

Data scientists are soon replacing business analysts or data analysts. Data scientists are experts who find solutions to analyze data. Just as web analysis, we have data scientists who have good business insight as to how to handle a business challenge. Sharp data scientists are not only involved in dealing business problems, but also choosing the relevant issues that can bring value-addition to the organization.

What is Hadoop?

Hadoop is a framework that allows for distributed processing of large data sets across clusters of commodity computers using a simple programming model. Click on **What Is Hadoop all about** to know more!

Why the name 'Hadoop'?

Hadoop doesn't have any expanding version like 'oops'. The charming yellow elephant you see is basically named after Doug's son's toy elephant!

Why do we need Hadoop?

Everyday a large amount of unstructured data is getting dumped into our machines. The major challenge is not to store large data sets in our systems but to retrieve and analyze the big data in the organizations, that too data present in different machines at different locations. In this situation a necessity for Hadoop arises. Hadoop has the ability to analyze the data present in different machines at different locations very quickly and in a very cost effective way. It uses the concept of MapReduce which enables it to divide the query into small parts and process them in parallel. This is also known as parallel computing. The link **Why Hadoop** gives you a detailed explanation about why Hadoop is gaining so much popularity!

What are some of the characteristics of Hadoop framework?

Hadoop framework is written in Java. It is designed to solve problems that involve analyzing large data (e.g. petabytes). The programming model is based on Google's MapReduce. The infrastructure is based on Google's Big Data and Distributed File System. Hadoop handles large files/data throughput and supports data intensive distributed applications. Hadoop is scalable as more nodes can be easily added to it.

Give a brief overview of Hadoop history.

In 2002, Doug Cutting created an open source, web crawler project. In 2004, Google published MapReduce, GFS papers. In 2006, Doug Cutting developed the open source, Mapreduce and HDFS project. In 2008, Yahoo ran 4,000 node Hadoop cluster and Hadoop won terabyte sort benchmark. In 2009, Facebook launched SQL support for Hadoop.

Give examples of some companies that are using Hadoop structure?

A lot of companies are using the Hadoop structure such as Cloudera, EMC, MapR, Hortonworks, Amazon, Facebook, eBay, Twitter, Google and so on.

What is the basic difference between traditional RDBMS and Hadoop?

Traditional RDBMS is used for transactional systems to report and archive the data, whereas *Hadoop* is an approach to store huge amount of data in the distributed file system and process it. RDBMS will be useful when you want to seek one record from Big data, whereas, Hadoop will be useful when you want Big data in one shot and perform analysis on that later.

What is structured and unstructured data?

Structured data is the data that is easily identifiable as it is organized in a structure. The most common form of structured data is a database where specific information is stored in tables, *that is,* rows and columns. Unstructured data refers to any data that cannot be identified easily. It could be in the form of images, videos, documents, email, logs and random text. It is not in the form of rows and columns.

What are the core components of Hadoop?

Core components of Hadoop are HDFS and MapReduce. HDFS is basically used to store large data sets and MapReduce is used to process such large data sets.

What is HDFS?

HDFS is a file system designed for storing very large files with streaming data access patterns, running clusters on commodity hardware.

What are the key features of HDFS?

HDFS is highly fault-tolerant, with high throughput, suitable for applications with large data sets, streaming access to file system data and can be built out of commodity hardware.

What is Fault Tolerance?

Suppose you have a file stored in a system, and due to some technical problem that file gets destroyed. Then there is no chance of getting the data back present in that file. To avoid such situations, Hadoop has introduced the feature of fault tolerance in HDFS. In Hadoop, when we store a file, it automatically gets replicated at two other locations also. So even if one or two of the systems collapse, the file is still available on the third system.

Replication causes data redundancy then why is is pursued in HDFS?

HDFS works with commodity hardware (systems with average configurations) that has high chances of getting crashed any time. Thus, to make the entire system highly fault-tolerant, HDFS replicates and stores data in different places. Any data on HDFS gets stored at atleast 3 different locations. So, even if one of them is corrupted and the other is unavailable for some time for any reason, then data can be accessed from the third one. Hence, there is no chance of losing the data. This replication factor helps us to attain the feature of Hadoop called Fault Tolerant.

Since the data is replicated thrice in HDFS, does it mean that any calculation done on one node will also be replicated on the other two?

Since there are 3 nodes, when we send the MapReduce programs, calculations will be done only on the original data. The master node will know which node exactly has that particular data. In case, if one of the nodes is not responding, it is assumed to be failed. Only then, the required calculation will be done on the second replica.

What is throughput? How does HDFS get a good throughput?

Throughput is the amount of work done in a unit time. It describes how fast the data is getting accessed from the system and it is usually used to measure performance of the system. In HDFS, when we want to perform a task or an action, then the work is divided and shared among different systems. So all the systems will be executing the tasks assigned to them independently and in parallel. So the work will be completed in a very short period of time. In this way, the HDFS gives good throughput. By reading data in parallel, we decrease the actual time to read data tremendously.

What is streaming access?

As HDFS works on the principle of '*Write Once, Read Many*', the feature of streaming access is extremely important in HDFS. HDFS focuses not so much on storing the data but how to retrieve it at the

fastest possible speed, especially while analyzing logs. In HDFS, reading the complete data is more important than the time taken to fetch a single record from the data.

What is a commodity hardware? Does commodity hardware include RAM?

Commodity hardware is a non-expensive system which is not of high quality or high-availability. Hadoop can be installed in any average commodity hardware. We don't need super computers or high-end hardware to work on Hadoop. Yes, Commodity hardware includes RAM because there will be some services which will be running on RAM.

What is a Namenode?

Namenode is the master node on which job tracker runs and consists of the metadata. It maintains and manages the blocks which are present on the datanodes. It is a high-availability machine and single point of failure in HDFS.

Is Namenode also a commodity?

No. Namenode can never be a commodity hardware because the entire HDFS rely on it. It is the single point of failure in HDFS. Namenode has to be a high-availability machine.

What is a metadata?

Metadata is the information about the data stored in datanodes such as location of the file, size of the file and so on.

What is a Datanode?

Datanodes are the slaves which are deployed on each machine and provide the actual storage. These are responsible for serving read and write requests for the clients.

Why do we use HDFS for applications having large data sets and not when there are lot of small files?

HDFS is more suitable for large amount of data sets in a single file as compared to small amount of data spread across multiple files. This is because Namenode is a very expensive high performance system, so it is not prudent to occupy the space in the Namenode by unnecessary amount of metadata that is generated for multiple small files. So, when there is a large amount of data in a single file, name node will occupy less space. Hence for getting optimized performance, HDFS supports large data sets instead of multiple small files.

What is a daemon?

Daemon is a process or service that runs in background. In general, we use this word in UNIX environment. The equivalent of Daemon in Windows is "services" and in Dos is " TSR".

What is a job tracker?

Job tracker is a daemon that runs on a namenode for submitting and tracking MapReduce jobs in Hadoop. It

assigns the tasks to the different task tracker. In a Hadoop cluster, there will be only one job tracker but many task trackers. It is the single point of failure for Hadoop and MapReduce Service. If the job tracker goes down all the running jobs are halted. It receives heartbeat from task tracker based on which Job tracker decides whether the assigned task is completed or not.

What is a task tracker?

Task tracker is also a daemon that runs on datanodes. Task Trackers manage the execution of individual tasks on slave node. When a client submits a job, the job tracker will initialize the job and divide the work and assign them to different task trackers to perform MapReduce tasks. While performing this action, the task tracker will be simultaneously communicating with job tracker by sending heartbeat. If the job tracker does not receive heartbeat from task tracker within specified time, then it will assume that task tracker has crashed and assign that task to another task tracker in the cluster.

Is Namenode machine same as datanode machine as in terms of hardware?

It depends upon the cluster you are trying to create. The Hadoop VM can be there on the same machine or on another machine. For instance, in a single node cluster, there is only one machine, whereas in the development or in a testing environment, Namenode and datanodes are on different machines.

What is a heartbeat in HDFS?

A heartbeat is a signal indicating that it is alive. A datanode sends heartbeat to Namenode and task tracker will send its heart beat to job tracker. If the Namenode or job tracker does not receive heart beat then they will decide that there is some problem in datanode or task tracker is unable to perform the assigned task.

Are Namenode and job tracker on the same host?

No, in practical environment, Namenode is on a separate host and job tracker is on a separate host.

What is a 'block' in HDFS?

A 'block' is the minimum amount of data that can be read or written. In HDFS, the default block size is 64 MB as contrast to the block size of 8192 bytes in Unix/Linux. Files in HDFS are broken down into block-sized chunks, which are stored as independent units. HDFS blocks are large as compared to disk blocks, particularly to minimize the cost of seeks. **If a particular file is 50 mb, will the HDFS block still consume 64 mb as the default size?** No, not at all! 64 mb is just a unit where the data will be stored. In this particular situation, only 50 mb will be consumed by an HDFS block and 14 mb will be free to store something else. It is the MasterNode that does data allocation in an efficient manner.

What are the benefits of block transfer?

A file can be larger than any single disk in the network. There's nothing that requires the blocks from a file to be stored on the same disk, so they can take advantage of any of the disks in the cluster. Making the unit of abstraction a block rather than a file simplifies the storage subsystem. Blocks provide fault tolerance and availability. To insure against corrupted blocks and disk and machine failure, each block is replicated to a small number of physically separate machines (typically three). If a block becomes unavailable, a copy can be read from another location in a way that is transparent to the client.

If we want to copy 10 blocks from one machine to another, but another machine can copy only 8.5 blocks, can the blocks be broken at the time of replication?

In HDFS, blocks cannot be broken down. Before copying the blocks from one machine to another, the Master node will figure out what is the actual amount of space required, how many block are being used, how much space is available, and it will allocate the blocks accordingly.

How indexing is done in HDFS?

Hadoop has its own way of indexing. Depending upon the block size, once the data is stored, HDFS will keep on storing the last part of the data which will say where the next part of the data will be. In fact, this is the base of HDFS.

If a data Node is full how it's identified?

When data is stored in datanode, then the metadata of that data will be stored in the Namenode. So Namenode will identify if the data node is full.

If datanodes increase, then do we need to upgrade Namenode?

While installing the Hadoop system, Namenode is determined based on the size of the clusters. Most of the time, we do not need to upgrade the Namenode because it does not store the actual data, but just the metadata, so such a requirement rarely arise.

Are job tracker and task trackers present in separate machines?

Yes, job tracker and task tracker are present in different machines. The reason is job tracker is a single point of failure for the Hadoop MapReduce service. If it goes down, all running jobs are halted.

When we send a data to a node, do we allow settling in time, before sending another data to that node?

Yes, we do.

Does hadoop always require digital data to process?

Yes. Hadoop always require digital data to be processed.

On what basis Namenode will decide which datanode to write on?

As the Namenode has the metadata (information) related to all the data nodes, it knows which datanode is free.

Doesn't Google have its very own version of DFS?

Yes, Google owns a DFS known as "*Google File System (GFS)*" developed by Google Inc. for its own use.

Who is a 'user' in HDFS?

A user is like you or me, who has some query or who needs some kind of data.

Is client the end user in HDFS?

No, Client is an application which runs on your machine, which is used to interact with the Namenode (job tracker) or datanode (task tracker).

What is the communication channel between client and namenode/datanode?

The mode of communication is SSH.

What is a rack?

Rack is a storage area with all the datanodes put together. These datanodes can be physically located at different places. Rack is a physical collection of datanodes which are stored at a single location. There can be multiple racks in a single location.

On what basis data will be stored on a rack?

When the client is ready to load a file into the cluster, the content of the file will be divided into blocks. Now the client consults the Namenode and gets 3 datanodes for every block of the file which indicates where the block should be stored. While placing the datanodes, the key rule followed is "*for every block of data, two copies will exist in one rack, third copy in a different rack*". This rule is known as "*Replica Placement Policy*".

Do we need to place 2nd and 3rd data in rack 2 only?

Yes, this is to avoid datanode failure.

What if rack 2 and datanode fails?

If both rack2 and datanode present in rack 1 fails then there is no chance of getting data from it. In order to avoid such situations, we need to replicate that data more number of times instead of replicating only thrice. This can be done by changing the value in replication factor which is set to 3 by default.

What is a Secondary Namenode? Is it a substitute to the Namenode?

The secondary Namenode constantly reads the data from the RAM of the Namenode and writes it into the hard disk or the file system. It is not a substitute to the Namenode, so if the Namenode fails, the entire Hadoop system goes down.

What is the difference between Gen1 and Gen2 Hadoop with regards to the Namenode?

In Gen 1 Hadoop, Namenode is the single point of failure. In Gen 2 Hadoop, we have what is known as Active and Passive Namenodes kind of a structure. If the active Namenode fails, passive Namenode takes over the charge.

What is MapReduce?

Map Reduce is the '*heart*' of Hadoop that consists of two parts – 'map' and 'reduce'. Maps and reduces are programs for processing data. 'Map' processes the data first to give some intermediate output which is further processed by 'Reduce' to generate the final output. Thus, MapReduce allows for distributed processing of the map and reduction operations.

Can you explain how do 'map' and 'reduce' work?

Namenode takes the input and divide it into parts and assign them to data nodes. These datanodes process the tasks assigned to them and make a key-value pair and returns the intermediate output to the Reducer. The reducer collects this key value pairs of all the datanodes and combines them and generates the final output.

What is 'Key value pair' in HDFS?

Key value pair is the intermediate data generated by maps and sent to reduces for generating the final output.

What is the difference between MapReduce engine and HDFS cluster?

HDFS cluster is the name given to the whole configuration of master and slaves where data is stored. Map Reduce Engine is the programming module which is used to retrieve and analyze data.

Is map like a pointer?

No, Map is not like a pointer.

Do we require two servers for the Namenode and the datanodes?

Yes, we need two different servers for the Namenode and the datanodes. This is because Namenode requires highly configurable system as it stores information about the location details of all the files stored in different datanodes and on the other hand, datanodes require low configuration system.

Why are the number of splits equal to the number of maps?

The number of maps is equal to the number of input splits because we want the key and value pairs of all the input splits.

Is a job split into maps?

No, a job is not split into maps. Spilt is created for the file. The file is placed on datanodes in blocks. For each split, a map is needed.

Which are the two types of 'writes' in HDFS?

There are two types of writes in HDFS: *posted and non-posted write*. Posted Write is when we write it and forget about it, without worrying about the acknowledgement. It is similar to our traditional Indian post. In a Non-posted Write, we wait for the acknowledgement. It is similar to the today's courier services. Naturally, non-posted write is more expensive than the posted write. It is much more expensive, though both writes

are asynchronous.

Why '*Reading*' is done in parallel and '*Writing*' is not in HDFS?

Reading is done in parallel because by doing so we can access the data fast. But we do not perform the *write* operation in parallel. The reason is that if we perform the *write* operation in parallel, then it might result in data inconsistency. For example, you have a file and two nodes are trying to write data into the file in parallel, then the first node does not know what the second node has written and vice-versa. So, this makes it confusing which data to be stored and accessed.

Can Hadoop be compared to NOSQL database like Cassandra?

Though NOSQL is the closet technology that can be compared to Hadoop, it has its own pros and cons. There is no DFS in NOSQL. Hadoop is not a database. It's a filesystem (HDFS) and distributed programming framework (MapReduce).

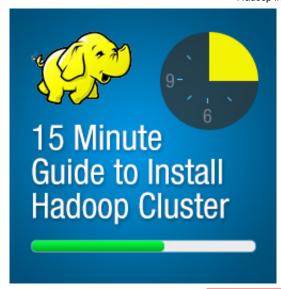## How can I install Cloudera VM in my system?

When you enrol for the hadoop course at Edureka, you can download the *Hadoop Installation steps.pdf* file from our dropbox. This will be shared with you by an e-mail. **To refer to the second list, click *[Hadoop Interview Questions – setting Up Hadoop Cluster!](#)* To refer to the third list, click *[Hadoop Interview Questions – MapReduce!](#)* To refer to the fourth list, click *[Hadoop Interview Questions - PIG!](#)***

Disqus seems to be taking longer than usual. [Reload](#)?

- # Search

Recent Posts

- [Free Webinar on 'Introduction to MongoDB'](#)
- [Importance of Data Science With Cassandra](#)
- [How to become a Hadoop Administrator?](#)
- [Why Big Data Professionals Need to Learn MongoDB?](#)
- [Why Should you go for Hadoop Administration Course?](#)

Categories

- [Android](#)
- [Apache Cassandra](#)
- [Apache Storm](#)
- [Big Data and Hadoop](#)
- [Business Analytics With R](#)
- [Cloud Computing](#)
- [Data Science](#)
- [Hadoop Administration](#)
- [Interview Questions](#)
- [Java](#)
- [MongoDB](#)
- [PMI-ACP](#)
- [PMP Exam Preparation](#)
- [Python for Big Data Analytics](#)
- [Resources & Misc.](#)
- [Uncategorized](#)
- [Webinars](#)

Enter your Email Address...

Subscribe

[RSS Feed](#)

[RSS Feed](#)