

# Hadoop Interview Questions and Answers Part-1

## 1. What is Big Data ?

Big data is vast amount of data (generally in GBs or TBs of size) that exceeds the regular processing capacity of the traditional computing servers and requires special parallel processing mechanism. This data is too big and its rate of increase gets accelerated. This data can be either structural or unstructured data which may not be able to process by legacy databases.

## 2. What is Hadoop ?

Hadoop is an open source frame work from Apache Software Foundation for storing & processing large-scale data usually called Big Data using clusters of commodity hardware.

## 3. Who uses Hadoop ?

Big organizations in which data grows exponentially day by day and must require Hadoop platform to process such huge data. For example , Facebook, Google, Amazon, Twitter, IBM, LinkedIn etc... companies uses hadoop technology to solve their big data processing problems.

## 4. What is commodity hardware?

Commodity hardware is a non-expensive system which is not of high quality or high-availability.

Hadoop can be installed on any commodity hardware. We don't need super computers or high-end hardware to work on Hadoop. Commodity hardware includes RAM because there will be some services which will be running on RAM.

## 5. 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 we want to seek one record from Big data, whereas, Hadoop will be useful when we want Big data in one shot and perform analysis on that later.

## 6. What are the modes in which Hadoop can run ?

Hadoop can run in three modes.

Stand alone or Local mode – No daemons will be running in this mode and everything runs in a single JVM.

Pseudo distributed mode – All the Hadoop daemons run on a local machine, simulating cluster on a small scale.

Fully distributed mode – A cluster of machines will be setup in master/slaves architecture to distribute and process the data across various nodes of commodity hardware.

## 7. What are main components/projects in Hadoop architecture ?

Hadoop Common: The common utilities that support the other Hadoop modules.

HDFS: Hadoop distributed file system that provides high-throughput access to application data.

Hadoop YARN: A framework for job scheduling and cluster resource management.

Hadoop MapReduce: A YARN-based system for parallel processing of large data sets.

## 8. List important default configuration files in Hadoop cluster ?

The default configuration files in hadoop cluster are:

core-default.xml

hdfs-default.xml

yarn-default.xml

mapred-default.xml

## 9. List important site-specific configuration files in Hadoop cluster ?

In order to override any hadoop configuration property's default values, we need to provide configuration values in site-specific configuration files. Below are the four site-specific .xml configuration files and environment variable setup file.

core-site.xml : Common properties are configured in this file.

hdfs-site.xml : Site specific hdfs properties are configured in this file

yarn-site.xml : Yarn specific properties can be provided in this file.

mapred-site.xml : Mapreduce framework specific properties will defined here.

hadoop-env.sh : Hadoop environment variables are setup in this file.

All these configuration files should be placed in hadoop's configuration directory etc/hadoop from hadoop's home directory.

## 10. How many hadoop daemon processes run on a Hadoop System ?

As of hadoop-2.5.0 release, three hadoop daemon processes run on a hadoop cluster.

NameNode daemon – Only one daemon runs for entire hadoop cluster.

Secondary NameNode daemon – Only one daemon runs for entire hadoop cluster.

DataNode daemon – One datanode daemon per each datanode in hadoop cluster

11. How to start all hadoop daemons at a time ?

\$ start-dfs.sh command can be used to start all hadoop daemons from terminal at a time.

12. If some hadoop daemons are already running and if we need to start any one remaining daemon process then what are the commands to use ?

Instead of start-dfs.sh which will trigger all the hadoop three daemons at a time, we can also start running each daemon separately by the below commands.

\$ hadoop-daemon.sh start namenode

\$ hadoop-daemon.sh start secondarynamenode

\$ hadoop-daemon.sh start datanode

13. How to stop all the three hadoop daemons at a time ?

By using stop-dfs.sh command, we can stop the above three daemon processes with a single command.

14. What are commands that need to be used to bring down a single hadoop daemon?

Below hadoop-daemon.sh commands can be used to bring down each hadoop daemon separately.

\$ hadoop-daemon.sh stop namenode

\$ hadoop-daemon.sh stop secondarynamenode

\$ hadoop-daemon.sh stop datanode

15. How many YARN daemon processes run on a cluster ?

Two types of Yarn daemons will be running on hadoop cluster in master/slave fashion.

ResourceManager – Master daemon process

NodeManager – One Slave daemon process per node in a cluster.

16. How to start Yarn daemon processes on a hadoop cluster ?

By running \$ start-yarn.sh command from terminal on each machine on hadoop cluster, Yarn daemons can be started.

17. How to verify whether the daemon processes are running or not ?

By using java's processes command \$ jps to check what are all the java processes running on a machine. This command lists down all the daemon processes running on a machine along with their process ids.

18. How to bring down the Yarn daemon processes ?

Using \$ stop-yarn.sh command, we can bring down both the Yarn daemon processes running on a machine.

19. Can we start both Hadoop daemon processes and Yarn daemon processes with a single command?

Yes, we can start all the above mentioned five daemon processes (3 hadoop + 2 Yarn) with a single command

\$ start-all.sh

20. Can we stop all the above five daemon processes with a single command ?

Yes, by using \$ stop-all.sh command all the above five daemon processes can be bring down in a single shot.

21. Which operating systems are supported for Hadoop deployment ?

The only supported operating system for hadoop's production deployment is Linux. However, with some additional software Hadoop can be deployed on Windows for test environments.

22. How could be the various components of Hadoop cluster deployed in production?

Both Name Node and Resource Manager can be deployed on a Master Node, and Data nodes and node managers can be deployed on multiple slave nodes.

There is a need for only one master node for namenode and Resource Manager on the system. The number of slave nodes for datanodes & node managers depends on the size of the cluster.

One more node with hardware specifications same as master node will be needed for secondary namenode.

23. 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.

24. 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.

25. What is the difference between jps and jps -lm commands ?

jps command returns the process id and short names for running processes. But jps -lm returns long messages along with process id and short names as shown below.

```
hadoop1@ubuntu-1:~$ jps
```

```
5314 SecondaryNameNode
```

```
5121 DataNode
```

```
5458 Jps
```

```
4995 NameNode
```

```
hadoop1@ubuntu-1:~$ jps -lm
```

```
5314 org.apache.hadoop.hdfs.server.namenode.SecondaryNameNode
```

```
5121 org.apache.hadoop.hdfs.server.datanode.DataNode
```

```
5473 sun.tools.jps.Jps -lm
```

```
4995 org.apache.hadoop.hdfs.server.namenode.NameNode
```

## Hadoop Interview Questions and Answers Part-2

1. What is HDFS ?

HDFS is a distributed file system implemented on Hadoop's framework. It is a block-structured distributed file system designed to store vast amount of data on low cost commodity hardware and ensuring high speed process on data. HDFS stores files across multiple machines and maintains reliability and fault tolerance. HDFS support parallel processing of data by Mapreduce framework.

2. What are the objectives of HDFS file system?

- Easily Store large amount of data across multiple machines

- Data reliability and fault tolerance by maintaining multiple copies of each block of a file.

- Capacity to move computation to data instead of moving data to computation server. I.e. processing data locally.

- Able to provide parallel processing of data by Mapreduce framework.

3. What are the limitations of HDFS file systems?

- HDFS supports file operations reads, writes, appends and deletes efficiently but it doesn't support file updates.

- HDFS is not suitable for large number of small sized files but best suits for large sized files. Because file system namespace maintained by Namenode is limited by it's main memory capacity as namespace is stored in namenode's main memory and large number of files will result in big fsimage file.

4. What is a block in HDFS and what is its size?

It is a fixed size chunk of data usually of size 128 MB. It is the minimum of size of data that HDFS can read/write.

HDFS files are broken into these fixed size chunks of data across multiple machines on a cluster. Thus, blocks are building bricks of a HDFS file. Each block is maintained in at least 3 copies as mentioned by replication factor in Hadoop configuration to provide data redundancy and maintain fault-tolerance.

5. What are the core components in HDFS Cluster ?

- Name Node

- Secondary Name Node

- Data Nodes

- Checkpoint Nodes

- Backup Node

6. What is a NameNode ?

Namenode is a dedicated machine in HDFS cluster which acts as a master serve that maintains file system namespace in its main memory and serves the file access requests by users. File system namespace mainly contains fsimage and edits files. Fsimage is a file which contains file names, file permissions and block locations of each file. Usually only one active namenode is allowed in HDFS default architecture.

7. What is a DataNode ?

DataNodes are slave nodes of HDFS architecture which store the blocks of HDFS files and sends blocks information to namenode periodically through heart beat messages.

Data Nodes serve read and write requests of clients on HDFS files and also perform block creation, replication and deletions.

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

It depends upon the cluster we 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.

9. What is a Secondary Namenode?

The Secondary NameNode is a helper to the primary NameNode. Secondary NameNode is a specially dedicated node in HDFS cluster whose main function is to take checkpoints of the file system metadata present on namenode. It is not a backup namenode and doesn't act as a namenode in case of primary namenode's failures. It just checkpoints namenode's file system namespace. For in depth details please refer the post [here](#).

10. What is a Checkpoint Node?

It is an enhanced secondary namenode whose main functionality is to take checkpoints of namenode's file system metadata periodically. It replaces the role of secondary namenode. Advantage of Checkpoint node over the secondary namenode is that it can upload the result of merge operation of fsimage and edits log files while checkpointing. For indepth details please refer the post [here](#).

11. What is a checkpoint ?

During Checkpointing process, fsimage file is merged with edits log file and a new fsimage file will be created which is usually called as a checkpoint.

12. What is a daemon?

Daemon is a process or service that runs in background. In general, we use this word in UNIX environment. Hadoop or Yarn daemons are Java processes which can be verified with jps command.

13. 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.

14. Are Namenode and Resource Manager run on the same host?

No, in practical environment, Namenode runs on a separate host and Resource Manager runs on a separate host.

15. What is the communication mode between namenode and datanode?

The mode of communication is SSH.

16. If we want to copy 20 blocks from one machine to another, but another machine can copy only 18.5 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.

17. 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.

18. If a data Node is full, then how is it 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.

19. 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.

20. 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 because it might result in data written by one node can be overwritten by other.

For example, we 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.

21. What is fsimage and edit log in hadoop?

Fsimage is a file which contains file names, file permissions and block locations of each file, and this file is maintained by Namenode for indexing of files in HDFS. We can call it as metadata about HDFS files. The fsimage file contains a serialized form of all the directory and file inodes in the filesystem. EditLog is a transaction log which contains records for every change that occurs to file system metadata. Note: Whenever a NameNode is restarted, the latest status of FsImage is built by applying edits records on last saved copy of FsImage.

# Hadoop Interview Questions and Answers Part-3

## 1. What is a Backup Node?

It is an extended checkpoint node that performs checkpointing and also supports online streaming of file system edits. It maintains an in memory, up-to-date copy of file system namespace and accepts a real time online stream of file system edits and applies these edits on its own copy of namespace in its main memory. Thus, it maintains always a latest backup of current file system namespace.

## 2. What are the differences between Backup Node and Checkpoint node or Secondary Namenode ?

Multiple checkpoint nodes can be registered with namenode but only a single backup node is allowed to register with namenode at any point of time. To create a checkpoint, checkpoint node or secondary namenode needs to download fsimage and edits files from active namenode and apply edits to fsimage and saves a copy of new fsimage as a checkpoint. But in backup node, no need to download fsimage and edits files from active namenode because, it already has an up-to-date copy of fsimage in its main memory and accesses online streaming of edits which are provided by namenode. So, applying these edits into fsimage in its own main memory and saving a copy in local FS. So, checkpoint creation in backup node is faster than that of checkpoint node or secondary namenode. The diff between checkpoint node and secondary namenode is that checkpoint node can upload the new copy of fsimage file back to namenode after checkpoint creation where as a secondary namenode can't upload but can only store in its local FS. Backup node provides the option of running namenode with no persistent storage but a checkpoint node or secondary namenode doesn't provide such option. In case of namenode failures, data loss in checkpoint node or secondary namenode is certain at least to a minimum amount of data due to time gap between two checkpoints. But in backup node, data loss is not certain and it maintains namespace which is in sync with namenode at any point of time.

## 3. What is Safe Mode in HDFS ?

Safe Mode is a maintenance state of NameNode during which Name Node doesn't allow any changes to the file system. During Safe Mode, HDFS cluster is ready-only and doesn't replicate or delete blocks. Name Node automatically enters safe mode during its start up and maintain blocks replication value within minimum and maximum allowable replication limit.

## 4. What is Data Locality in HDFS ?

One of the HDFS design idea is that "Moving Computation is cheaper than Moving data". If data sets are huge, running applications on nodes where the actual data resides will give efficient results than moving data to nodes where applications are running. This concept of moving applications to data, is called Data Locality. This reduces network traffic and increases speed of data processing and accuracy of data since there is no chance of data loss during data transfer through network channels because there is no need to move data.

## 5. 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.

## 6. What is Rack Awareness ?

The concept of maintaining Rack Id information by NameNode and using these rack ids for choosing closest data nodes for HDFS file read or writes requests is called Rack Awareness. By choosing closest data nodes for read/writes request through rack awareness policy, minimizes the write cost and maximizing read speed.

## 7. How does HDFS File Deletes or Undeletes work?

When a file is deleted from HDFS, it will not be removed immediately from HDFS, but HDFS moves the file into /trash directory. After certain period of time interval, the NameNode deletes the file from the HDFS /trash directory. The deletion of a file releases the blocks associated with the file. Time interval for which a file remains in /trash directory can be configured with fs.trash.interval property stored in core-site.xml. As long as a file remains in /trash directory, the file can be undeleted by moving the file from /trash directory into required location in HDFS. Default trash interval is set to 0. So, HDFS Deletes file without storing in trash.

## 8. What is a Rebalancer in HDFS ?

Rebalancer is a administration tool in HDFS, to balance the distribution of blocks uniformly across all the data nodes in the cluster. Rebalancing will be done on demand only. It will not get triggered automatically. HDFS administrator issues this command on request to balance the cluster (\$ hdfs balancer). If a Rebalancer is triggered, NameNode will scan entire data node list and when Under-utilized data node is found, it moves blocks from over-utilized data nodes or not-under-utilized data nodes to this current data node If Over-utilized data node is found, it moves blocks from this data node to other under-utilized or not-over-utilized data nodes.

9. What is the need for Rebalancer in HDFS ?

Whenever a new data node is added to the existing HDFS cluster or a data node is removed from the cluster then some of the data nodes in the cluster will have more/less blocks compared to other data nodes. In this unbalanced cluster, data read/write requests become very busy on some data nodes and some data nodes are under utilized. In such cases, to make all the data nodes space is uniformly utilized for blocks distribution, rebalancing will be done by Hadoop Administrator.

10. When will be a cluster in balanced status ?

A cluster is in a balanced status when, % of space used in each data node is within limits of Average % of space used on data nodes +/- Threshold size . Percentage space used on a data node should not be less than Average % of space used on data nodes – Threshold size. Percentage space used on a data node should not be greater than Average % of space used on data nodes + Threshold size. Here Threshold size is configurable value which is 20 % of used spaced by default.

11. What is Delegation Token in Hadoop ?

Delegation token is an authentication token used to access to secure Namenode from a non-secure client node. HDFS fetchdt command can be used to get the delegation token and store it in a file on the local system. its common syntax is as follows.

```
$ hadoop fetchdt fileName
```

This fetchdt uses either RPC or HTTPS (over Kerberos) to get the token, and thus it requires kerberos tickets to be present before the fetchdt run. Once the token is obtained then user can run an HDFS command without having Kerberos tickets, by pointing HADOOP\_TOKEN\_FILE\_LOCATION environmental variable to the delegation token file.

12. What is Hadoop Streaming ?

Streaming is a generic API that allows programs written in virtually any language to be used as Hadoop Mapper and Reducer implementations

13. What is the characteristic of streaming API that makes it flexible run mapreduce jobs in languages like perl, ruby, awk etc. ?

Hadoop Streaming allows to use arbitrary programs for the Mapper and Reducer phases of a Map Reduce job by having both Mappers and Reducers receive their input on stdin and emit output (key, value) pairs on stdout.

14. What is a metadata?

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

15. What is the lowest granularity at which you can apply replication factor in HDFS

- We can choose replication factor per directory (True)
- We can choose replication factor per file in a directory (True)
- We can choose replication factor per block of a file (False)

16. What is HBase?

A distributed, column-oriented database. HBase uses HDFS for its underlying storage, and supports both batch-style computations using MapReduce and random reads.

17. What is ZooKeeper?

A distributed, highly available coordination service. ZooKeeper provides primitives such as distributed locks that can be used for building distributed applications.

18. What is Chukwa?

A distributed data collection and analysis system. Chukwa runs collectors that store data in HDFS, and it uses MapReduce to produce reports.

19. What is Avro?

A data serialization system for efficient, cross-language RPC, and persistent data storage.

20. What is Sqoop in Hadoop?

It is a tool to transfer the data between Relational database management system(RDBMS) and Hadoop. Thus, we can sqoop the data from RDBMS like mySql or Oracle into HDFS , Hive or HBase as well as exporting data from HDFS, Hive or HBase to RDBMS. Sqoop will read the table row-by-row and the import process is performed in Parallel. Thus, the output may be in multiple files. We will cover few more Hadoop interview Questions and answers in upcoming posts.

# Hadoop Interview Questions and Answers Part-3

1. As 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.

2. What happens if you get a 'connection refused java exception' when you type `hadoop fs -ls /`?

It could mean that the Namenode is not working on our hadoop cluster.

3. Can we have multiple entries in the master files?

Yes, we can have multiple entries in the Master files.

4. Why do we need a password-less SSH in Fully Distributed environment?

We need a password-less SSH in a Fully-Distributed environment because when the cluster is LIVE and running in Fully Distributed environment, the communication is too frequent. The Resource Manager/Namenode should be able to send a task to Node manager /Datanodes quickly.

5. Does this lead to security issues?

No, not at all. Hadoop cluster is an isolated cluster. And generally it has nothing to do with an internet. It has a different kind of a configuration. We needn't worry about that kind of a security breach, for instance, someone hacking through the internet, and so on. Hadoop has implemented Kerberos for strong security way to connect to other machines to fetch and to process data.

6. which port does SSH work on?

SSH works on Port 22, though it can be configured. 22 is the default Port number.

7. If we only had 32 MB of memory how can we sort one terabyte of data?

Take a smaller chunk of data and sort it in memory, so we partition it in lots of little data sets. Then merge those sorted lists into one big list before writing the results back to disk.

8. How can we create an empty file in HDFS ?

We can create empty file with `hadoop fs -touchz` command. It creates a zero byte file.

```
$ hadoop fs -touchz /user/hadoop/filename
```

9. How can we see the contents of a Snappy compressed file or SequenceFile via command line?

With the help of `$ hadoop fs -text /user/hadoop/filename` command we can see the contents of sequencefile or any compressed format file in text format.

10. How can we check the existence of a file in HDFS ?

The `hadoop test` is used for file test operations. The syntax is shown below:

```
hadoop fs -test -[ezd] URI
```

Here "e" for checking the existence of a file, "z" for checking the file is zero length or not, "d" for checking the path is a directory or not. On success, the test command returns 1 else 0.

11. How can we set the replication factor of directory via command line?

Hadoop `setrep` is used to change the replication factor of a file. Use the `-R` option for recursively changing the replication factor.

```
$ hadoop fs -setrep -w 4 -R /user/hadoop/dir
```

12. How can we apply the permission to all the files under a HDFS directory recursively?

Using `$ hadoop fs -chmod -R 755 /user/hadoop/dir` command we can set the permissions to a directory recursively.

13. What is `hadoop fs -stat` command used for?

Hadoop `stat` returns the stats information of a file. It returns the last updated date and time. The syntax of `stat` is shown below:

```
$ hadoop fs -stat /filename
```

```
2014-09-28 18:34:06
```

14. What is `expunge` command in HDFS and why is it used for ?

Hadoop `fs expunge` command is used to empty the trash directory in HDFS. Syntax is:

```
$ hadoop fs -expunge
```

15. How can we see the output of a MR job as a single file if the reducer might have created multiple `part-r-0000*` files ?

We can use `hadoop fs -getmerge` command to combine all the `part-r-0000*` files into single file and this file can be browsed to view the entire results of the MR job at a time. Syntax is:

```
$ hadoop fs -getmerge <hdfssrc> <localdestination> [addnl]
```

The `addnl` option is for adding new line character at the end of each file.

16. Which of the following is most important when selecting hardware for our new Hadoop cluster?

1. The number of CPU cores and their speed.
2. The amount of physical memory.
3. The amount of storage.
4. The speed of the storage.
5. It depends on the most likely workload.

Answer 5 – Though some general guidelines are possible and we may need to generalize whether our cluster will be running a variety of jobs, the best fit depends on the anticipated workload.

17. Why would you likely not want to use network storage in your cluster?

1. Because it may introduce a new single point of failure.
2. Because it most likely has approaches to redundancy and fault-tolerance that may be unnecessary given Hadoop's fault tolerance.
3. Because such a single device may have inferior performance to Hadoop's use of multiple local disks simultaneously.
4. All of the above.

Answer 4: Network storage comes in many flavors but in many cases we may find a large Hadoop cluster of hundreds of hosts reliant on a single (or usually a pair of) storage devices. This adds a new failure scenario to the cluster and one with a less uncommon likelihood than many others. Where storage technology does look to address failure mitigation it is usually through disk-level redundancy.

18. We will be processing 10 TB of data on our cluster. Our main MapReduce job processes financial transactions, using them to produce statistical models of behavior and future forecasts. Which of the following hardware choices would be our first choice for the cluster?

1. 20 hosts each with fast dual-core processors, 4 GB memory, and one 500 GB disk drive.
2. 30 hosts each with fast dual-core processors, 8 GB memory, and two 500 GB disk drives.
3. 30 hosts each with fast quad-core processors, 8 GB memory, and one 1 TB disk drive.
4. 40 hosts each with 16 GB memory, fast quad-core processors, and four 1 TB disk drives.

Answer 3. Probably! We would suggest avoiding the first configuration as, though it has just enough raw storage and is far from under powered, there is a good chance the setup will provide little room for growth. An increase in data volumes would immediately require new hosts and additional complexity in the MapReduce job could require additional processor power or memory. Configurations B and C both look good as they have surplus storage for growth and provide similar head-room for both processor and memory. B will have the higher disk I/O and C the better CPU performance. Since the primary job is involved in financial modelling and forecasting, we expect each task to be reasonably heavyweight in terms of CPU and memory needs. Configuration B may have higher I/O but if the processors are running at 100 percent utilization it is likely the extra disk throughput will not be used. So the hosts with greater processor power are likely the better fit. Configuration D is more than adequate for the task and we don't choose it for that very reason; why buy more capacity than we know we need.