Below are the main components in Hadoop 2.x:

1. Name Node
2. Data Node
3. YARN (Yet another resource negotiator)

**Namenode**

1. NameNode is the Heart of HDFS.
2. NameNode is also known as the Master.
3. NameNode only stores the metadata of HDFS – the directory tree of all files in the file system, and tracks the files across the cluster.
4. NameNode does not store the actual data or the dataset. The data itself is actually stored in the DataNodes.
5. NameNode knows the list of the blocks and its location for any given file in HDFS. With this information NameNode knows how to construct the file from blocks.
6. NameNode is so critical to HDFS and when the NameNode is down, HDFS/Hadoop cluster is inaccessible and considered down.
7. NameNode is usually configured with a lot of memory (RAM).

**DataNode**

1. DataNode is responsible for storing the actual data in HDFS.
2. DataNode is also known as the Slave
3. NameNode and DataNode are in constant communication.
4. When a DataNode starts up it announce itself to the NameNode along with the list of blocks it is having.
5. When a DataNode is down, it does not affect the availability of data or the cluster. NameNode will arrange for replication for the blocks managed by the DataNode that is not available.
6. DataNode is usually configured with a lot of hard disk space. Because the actual data is stored in the DataNode.

**YARN (Yet Another Resource Negotiator)**

YARN is a latest processing and management framework of Hadoop. It was created by dividing the processing engine of Hadoop into smaller, more manageable parts. It also monitors and manages jobs submitted by user in a highly performance efficient manner.

The six important elements of the YARN architecture are the Resource Manager, Node Manager, Application Master, Application Manager, Containers and scheduler

1. The Resource Manager, or RM, usually numbered one per cluster, is the master and knows where the slaves are located, referred to as Rack Awareness, and how many resources they have. The RM runs several services, the most important of which is the Resource Scheduler that decides how to assign the resources.
2. The Node Manager, of which there can be many in one cluster, is the slave of the infrastructure. When it starts, it announces itself to the RM and periodically sends a heartbeat to the RM. Each Node Manager offers resources to the cluster, the resource capacity being the amount of memory and the number of vcores. At run-time, the Resource Scheduler decides how to use this capacity. A container is a fraction of the Node Manager capacity and it is used by the client for running a program.

Each Node Manager takes instructions from the Resource Manager, and reports and handles containers on a single node.

1. The ApplicationMaster is a framework-specific process that negotiates resources for a single application, that is, a single job or a directed acyclic graph of jobs, which runs in the first container allocated for the purpose. Each ApplicationMaster requests resources from the Resource Manager, then works with containers provided by Node Managers.
2. A container is an abstraction and runtime on the commodity machines that helps runs a programming unit.
3. Application Manager is the Master of all the Application Masters that means it has status of all the Application Masters.
4. Scheduler is a daemon which keeps the repository of all the containers on the cluster.

