1. **HDFS:**

* HDFS is a Java-based file system that provides scalable and reliable data storage, and it was designed to span large clusters of commodity servers. HDFS has demonstrated production scalability of up to 200 PB of storage and a single cluster of 4500 servers, supporting close to a billion files and blocks. When that quantity and quality of enterprise data is available in HDFS, and YARN enables multiple data access applications to process it, Hadoop users can confidently answer questions that eluded previous data platforms.
* HDFS is a scalable, fault-tolerant, distributed storage system that works closely with a wide variety of concurrent data access applications, coordinated by YARN. HDFS will “just work” under a variety of physical and systemic circumstances. By distributing storage and computation across many servers, the combined storage resource can grow linearly with demand while remaining economical at every amount of storage.

**Features of HDFS:**

* It is suitable for the distributed storage and processing.
* Hadoop provides a command interface to interact with HDFS.
* The built-in servers of namenode and datanode help users to easily check the status of cluster.
* Streaming access to file system data.
* HDFS provides file permissions and authentication.

Goals of HDFS:

## HDFS is used for fault detection and recovery.

## It can manage the application having huge datasets.

## It reduces the network traffic.

## HADOOP CLUSTER:

## Hadoop clusters are comprised of three different node types:

## (i)master nodes

## (ii)worker nodes and

## (iii)client nodes.

## MASTER NODE:

## Master nodes oversee the key operations that comprise (i) storing data in the Hadoop Distributed File System (HDFS)

## (ii) running parallel computations on that data using MapReduce.

## (iii)The NameNode coordinates the data storage function of the HDFS.

## (iv) the JobTracker oversees and coordinates the parallel processing of data using MapReduce.

## WORKER NODE:

## Worker nodes make up the majority of virtual machines and perform the job of storing the data and running computations.

## Each worker node runs both a DataNode and TaskTracker service that communicates with, and receives instructions from their master nodes.

## The TaskTracker service is subordinate to the JobTracker, and the DataNode service is subordinate to the NameNode.

## CLIENT NODE:

* Client nodes have Hadoop installed with all the cluster settings, but are neither master or worker nodes.
* Instead, the client node loads data into the cluster, submits MapReduce jobs describing how that data should be processed, and then retrieves or views the results of the job when processing is finished.

Managing Resources for Hadoop Clusters

* The Data Director for Hadoop Management Server allows you to manage the resource pools, datastores, and networks that you use to create your Hadoop clusters.

## FEATURES OF HADOOP CLUSTER:

## Hadoop clusters are known for boosting the speed of data analysis applications.

## They also are highly scalable.

## If a cluster's processing power is overwhelmed by growing volumes of [data](http://searchdatamanagement.techtarget.com/definition/data), additional cluster nodes can be added to increase throughput.

## Hadoop clusters also are highly resistant to failure because each piece of data is copied onto other cluster nodes, which ensures that the data is not lost if one node fails.

1. HDFS BLOCK:

Blocks:

A block is the smallest unit of data that can be stored or retrieved from the disk. Filesystems deal with the data stored in blocks. Filesystem blocks are normally in few kilobytes of size. Blocks are transparent to the user who is performing filesystem operations like read and write.

Hdfs blocks:

* Hadoop distributed file system also stores the data in terms of blocks.
* However the block size in HDFS is very large. The default size of HDFS block is 64MB.
* The files are split into 64MB blocks and then stored into the hadoop filesystem.
* The hadoop application is responsible for distributing the data blocks across multiple nodes.

Reason why hdfs blocks are large in size:

The main reason for having the HDFS blocks in large size is to reduce the cost of seek time. In general, the seek time is 10ms and disk transfer rate is 100MB/s. To make the seek time 1% of the disk transfer rate, the block size should be 100MB. The default size HDFS block is 64MB.

ADVANTAGES OF HDFS BLOCKS:

* The blocks are of fixed size, so it is very easy to calculate the number of blocks that can be stored on a disk.
* HDFS block concept simplifies the storage of the datanodes. The namenode maintains the metadata of all the blocks.
* If the size of the file is less than the HDFS block size, then the file does not occupy the complete block storage.
* As the file is chunked into blocks, it is easy to store a file that is larger than the disk size as the data blocks are distributed and stored on multiple nodes in a hadoop cluster.
* Blocks are easy to replicate between the datanodes and thus provide fault tolerance and high availability