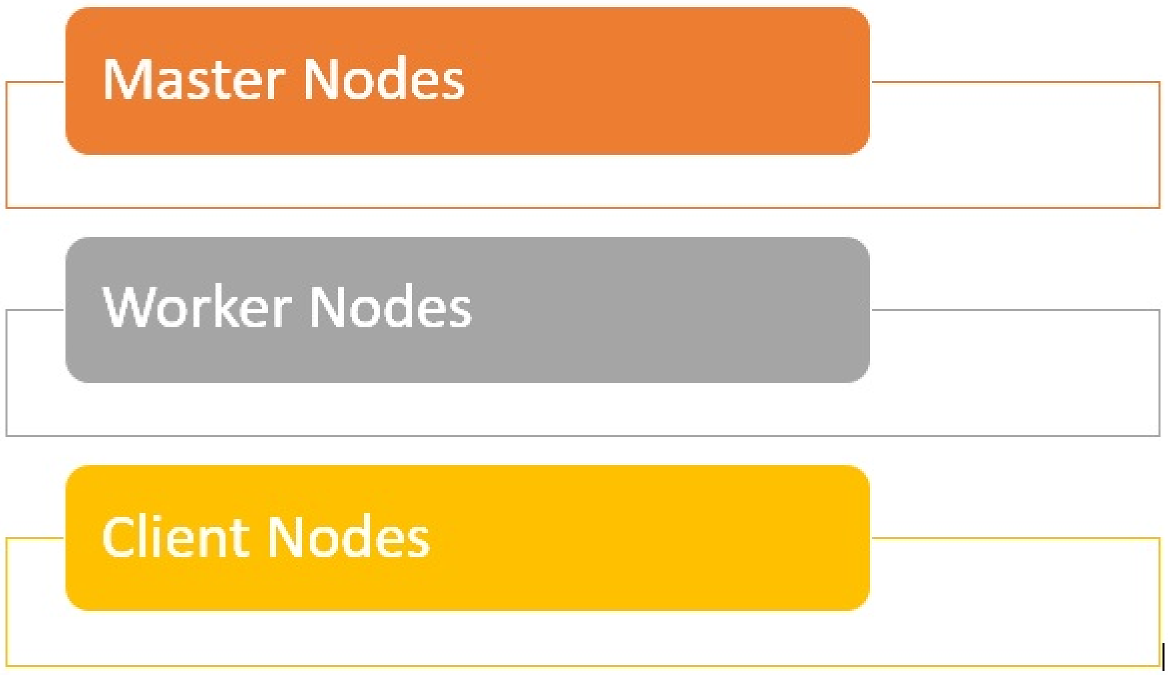
What Is a Hadoop Cluster?

Apache Hadoop is an open source, Java-based, software framework and parallel data processing engine. It enables big data analytics processing tasks to be broken down into smaller tasks that can be performed in parallel by using an algorithm (like the MapReduce algorithm), and distributing them across a Hadoop cluster. A Hadoop cluster is a collection of computers, known as nodes, that are networked together to perform these kinds of parallel computations on big data sets. Unlike other computer clusters, Hadoop clusters are designed specifically to store and analyze mass amounts of structured and unstructured data in a distributed computing environment. Further distinguishing **Hadoop ecosystems** from other computer clusters are their unique structure and architecture. Hadoop clusters consist of a network of connected master and slave nodes that utilize high availability, low-cost commodity hardware. The ability to linearly scale and quickly add or subtract nodes as volume demands makes them well-suited to **big data analytics** jobs with data sets highly variable in size.

Hadoop Cluster Architecture

Hadoop clusters are composed of a network of master and worker nodes that orchestrate and execute the various jobs across the Hadoop distributed file system. The master nodes typically utilize higher quality hardware and include a NameNode, Secondary NameNode, and JobTracker, with each running on a separate machine. The workers consist of virtual machines, running both DataNode and TaskTracker services on commodity hardware, and do the actual work of storing and processing the jobs as directed by the master nodes. The final part of the system are the Client Nodes, which are responsible for loading the data and fetching the results.



- Master nodes are responsible for storing data in HDFS and overseeing key operations, such as running parallel computations on the data using MapReduce.
- The worker nodes comprise most of the virtual machines in a Hadoop cluster, and perform the job of storing the data and running computations. Each worker node runs the DataNode and TaskTracker services, which are used to receive the instructions from the master nodes.
- Client nodes are in charge of loading the data into the cluster. Client nodes first submit MapReduce jobs describing how data needs to be processed and then fetch the results once the processing is finished.

What is cluster size in Hadoop?

A Hadoop cluster size is a set of metrics that defines storage and compute capabilities to run Hadoop workloads, namely :

- Number of nodes : number of Master nodes, number of Edge Nodes, number of Worker Nodes.

- Configuration of each type node: number of cores per node, RAM and Disk Volume.

What are the advantages of a Hadoop Cluster?

- Hadoop clusters can boost the processing speed of many big data analytics jobs, given their ability to break down large computational tasks into smaller tasks that can be run in a parallel, distributed fashion.
- Hadoop clusters are easily scalable and can quickly add nodes to increase throughput, and maintain processing speed, when faced with increasing data blocks.
- The use of low cost, high availability commodity hardware makes Hadoop clusters relatively easy and inexpensive to set up and maintain.
- Hadoop clusters replicate a data set across the distributed file system, making them resilient to data loss and cluster failure.
- Hadoop clusters make it possible to integrate and leverage data from multiple different source systems and data formats.
- It is possible to deploy Hadoop using a single-node installation, for evaluation purposes.

What are the challenges of a Hadoop Cluster?

- Issue with small files - Hadoop struggles with large volumes of small files - smaller than the Hadoop block size of 128MB or 256MB by default. It wasn't designed to support big data in a scalable way. Instead, Hadoop works well when there are a small number of large files. Ultimately when you increase the volume of small files, it overloads the Namenode as it stores namespace for the system.
- High processing overhead - reading and writing operations in Hadoop can get very expensive quickly especially when processing large amounts of data. This all comes down to

Hadoop's inability to do in-memory processing and instead data is read and written from and to the disk.

- Only batch processing is supported - Hadoop is built for small volumes of large files in batches. This goes back to the way data is collected and stored which all has to be done before processing starts. What this ultimately means is that streaming data is not supported and it cannot do real-time processing with low latency.
- Iterative Processing - Hadoop has a data flow structure is set-up in sequential stages which makes it impossible to do iterative processing or use for ML.