# Apache Hadoop Lab – 1

## What is Apache Hadoop?

Hadoop is an open-source framework that enables distributed storage and processing of large datasets across clusters of computers using simple programming models. It is designed to scale up from a single server to thousands of machines, each offering local computation and storage. The core idea behind Hadoop is distributed computing, where large tasks are divided into smaller chunks and processed in parallel across multiple machines, making big data processing feasible and efficient.

Imagine you have 10,000 pages of a book and one person takes weeks to read it. But if you divide the pages among 100 people, they can finish it much faster. Hadoop does the same—it distributes data and processes it simultaneously across multiple computers.

## Core Components:

Hadoop consists of four main components:

1. Hadoop Distributed File System (HDFS) – Storage Layer
2. Yet Another Resource Negotiator (YARN) – Resource Management Layer
3. MapReduce – Processing Framework
4. Hadoop Common – Shared Libraries & Utilities

Hadoop Distributed File System (HDFS):

HDFS is the primary storage system used by Hadoop. It enables reliable and scalable storage of large datasets across multiple machines, ensuring fault tolerance and high availability.

How Does HDFS Work?

1. Data is divided into blocks – When a file is stored in HDFS, it is split into fixed-sized **blocks** (default: 128MB or 256MB).
2. Blocks are stored across multiple machines – HDFS distributes these blocks among different DataNodes in the cluster.
3. Data replication ensures fault tolerance – Each block is replicated (default: 3 copies) across different machines to prevent data loss in case of hardware failure.

Two key components in HDFS:

* NameNode (Master Node): Manages metadata (file locations, replication info, etc.).
* DataNodes (Worker Nodes): Store actual data blocks and periodically send reports to the NameNode.

Fault Tolerance in HDFS:

* HDFS is fault tolerant, meaning it can handle node failures without data loss. The NameNode keeps track of all data locations, and DataNodes periodically send health reports. If a DataNode fails, Hadoop replicates the lost data to another node

Scalability in HDFS:

* HDFS is highly scalable, meaning it can accommodate growing data by adding more machines to the cluster.

Yet Another Resource Negotiator (YARN):

YARN is Hadoop’s resource management framework, introduced in Hadoop 2.0 to improve cluster efficiency. It ensures effective resource allocation for running distributed applications.

How Does YARN Work?

1. Receives job requests – When a user submits a job, YARN receives the request.
2. Allocates resources dynamically – YARN decides which machines will run which tasks, based on available CPU, memory, and disk space.
3. Manages multiple workloads – It ensures that multiple applications can run simultaneously without conflict.

Key Components in YARN:

* ResourceManager (Master Node): Monitors available resources (CPU, memory) and assigns them to jobs.
* NodeManagers (Worker Nodes): Manage resources on individual nodes and report status to the ResourceManager.
* ApplicationMaster: Coordinates the execution of a specific application.

YARN enables parallel processing, allowing multiple applications to run simultaneously without slowing down other tasks

MapReduce:

MapReduce is Hadoop’s data processing framework. It processes massive datasets in parallel by breaking jobs into smaller tasks that run across multiple machines.

How Does MapReduce Work?

1. Splits the job into smaller tasks (Map Phase) – The input data is divided into chunks and processed in parallel.
2. Processes data in parallel – Each chunk is processed independently by Mapper nodes.
3. Aggregates results (Reduce Phase) – The Reducer nodes combine intermediate results to generate the final output.

Key Components in MapReduce:

* Mapper: Processes input data in parallel and generates intermediate results.
* Reducer: Aggregates and summarizes the intermediate results into the final output.

MapReduce follows a master-slave architecture:

* JobTracker (Master Node): Assigns jobs to TaskTrackers.
* TaskTrackers (Worker Nodes): Execute the Map and Reduce tasks

MapReduce leverages parallel processing to analyze large-scale datasets efficiently

Hadoop Common:

Hadoop Common consists of libraries and utilities that enable other Hadoop components to function properly.

Key Features of Hadoop Common:

* Provides APIs and Libraries – Essential tools for Hadoop programs.
* Ensures compatibility – Makes sure all Hadoop components can work together seamlessly.
* Manages configuration settings – Stores and distributes configuration files across the cluster.

## Let’s Understand Big Words of Big Data in Hadoop:

Distributed Computing: Hadoop distributes tasks across multiple computers to process large datasets efficiently.

Fault Tolerance: Hadoop maintains multiple copies of data across different machines, ensuring no data loss even if a node fails.

Scalability: Hadoop can handle exponentially growing data by adding more machines to the cluster.

Parallel Processing: Multiple tasks run simultaneously across nodes, reducing processing time.

Commodity Hardware: Hadoop runs on low-cost, standard machines instead of expensive high-end servers.

Cluster: A group of connected computers working together as a single unit.

Nodes: Individual computers in a Hadoop cluster.

Blocks: Hadoop divides files into smaller fixed-size pieces called blocks, which are stored across multiple nodes.