BÀI TẬP LỚN CUỐI KỲ

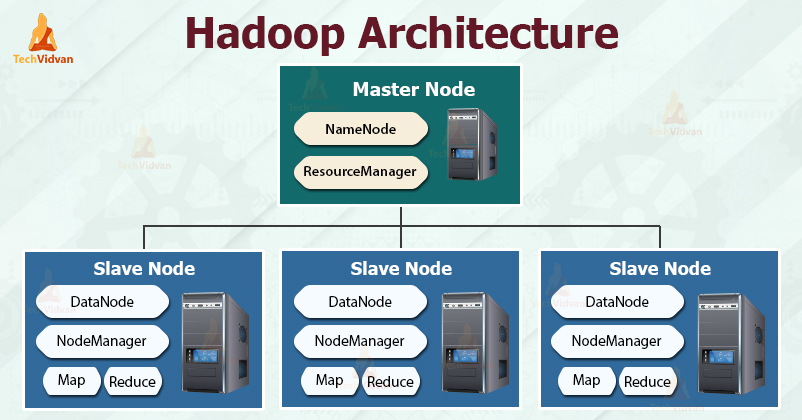
# MÔN KỸ THUẬT DỮ LIỆU – 055240

CHỦ ĐỀ: big data VÀ hadoop ecosystem

Giới thiệu:

Hadoop là một Apache framework mã nguồn mở cho phép phát triển các ứng dụng phân tán (distributed processing) để lưu trữ và quản lý các tập dữ liệu lớn. Hadoop hiện thực mô hình MapReduce, mô hình mà ứng dụng sẽ được chia nhỏ ra thành nhiều phân đoạn khác nhau được chạy song song trên nhiều node khác nhau. Hadoop được viết bằng Java tuy nhiên vẫn hỗ trợ C++, Python, Perl bằng cơ chế streaming.

# Apache Hadoop Architecture – HDFS, YARN & MapReduce



The Hadoop architecture comprises three layers. They are:

1. Storage layer (HDFS)
2. Resource Management layer (YARN)
3. Processing layer (MapReduce)

#### 1. HDFS

HDFS is the **Hadoop Distributed File System**, which runs on inexpensive commodity hardware. It is the storage layer for Hadoop. The files in HDFS are broken into block-size chunks called data blocks.

These blocks are then stored on the slave nodes in the cluster. The block size is 128 MB by default, which we can configure as per our requirements.

### 6 Important Features of HDFS

#### 1. Fault Tolerance

#### 2. High Availability

#### 3. High Reliability

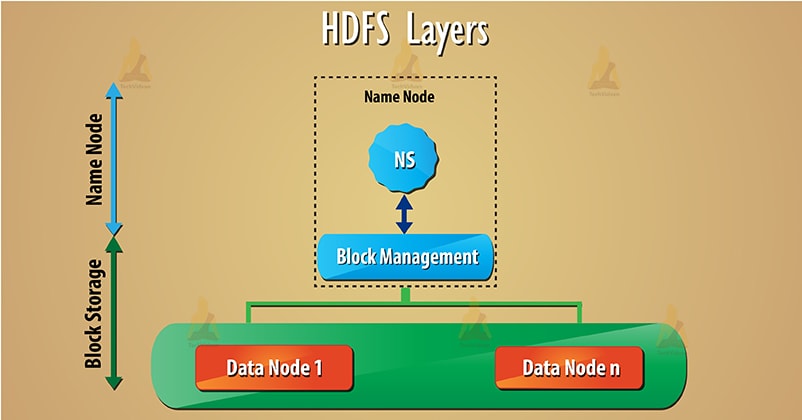
#### 4. Replication

#### 5. Scalability

#### 6. Distributed Storage

### HDFS Architecture

HDFS has two main layers given below:



a) Namespace– This layer manages files, directories, and [blocks](https://techvidvan.com/tutorials/hadoop-hdfs-data-block/). This layer supports basic file system operation such as creation, deletion of files.

b) Block Storage– It has two parts-

* Block management – It supports block related operation such as creation, deletion of the blocks. It manages data nodes in the cluster and takes care of replication management.
* Physical storage – This stores the blocks on the local file system and provides access to read or write operation. Follow this link to learn HDFS data read and write operation.

### Limitation of current HDFS Architecture

#### 1. Tightly coupled block storage and Namespace

Namespace layer and storage layer are tightly coupled. It makes alternate implementation of namenode difficult. And it restricts other services to use block storage.

#### 2. Namespace Scalability

The namespace is not scalable like datanode. Scaling in HDFS cluster is horizontally by adding datanodes. But we can’t add more namespace to an existing cluster. We can vertically scale namespace on a single namenode.

#### 3. Performance

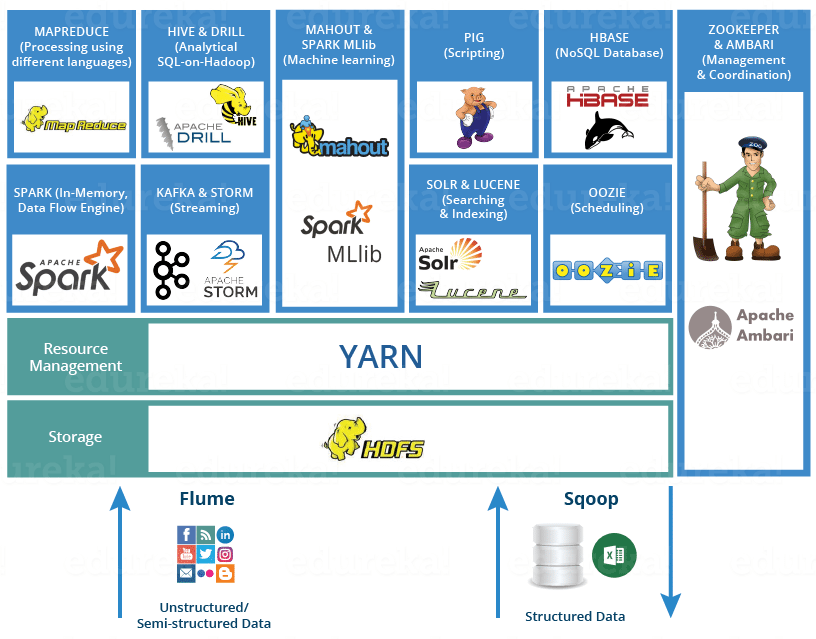
Hadoop entire performance depends on the throughput of the namenode. An operation of current file system depends on the throughput of a single namenode. NameNode at present supports 60,000 concurrent tasks.

Upcoming **MapReduce** will have support for more than 1,00,000 concurrent tasks. And this will need more namenode.

#### 4. Isolation

There is no separation of the namespace. So there is no isolation among tenant organization that is using the cluster.

## **HADOOP ECOSYSTEM**



### Advantages of HDFS

After learning what is HDFS Data Block, let’s now discuss the advantages of Hadoop HDFS.

#### 1. Ability to store very large files

Hadoop HDFS store very large files which are even larger than the size of a single disk as Hadoop framework break file into blocks and distribute across various nodes.

#### 2. Fault tolerance and High Availability of HDFS

Hadoop framework can easily replicate Blocks between the datanodes. Thus provide fault tolerance and **high availability** HDFS.

#### 3. Simplicity of storage management

As HDFS has fixed block size (128MB), so it is very easy to calculate the number of blocks that can be stored on the disk.

#### 4. Simple Storage mechanism for datanodes

Block in HDFS simplifies the storage of the**Datanodes**. **Namenode** maintains metadata of all the blocks. HDFS Datanode does not need to concern about the block metadata like file permissions etc.