Hadoop is an open-source framework that allows to store and process big data in a distributed

environment across clusters of computers using simple programming models.

3-tier architecture

client, server and database

Hadoop

Using the solution provided by Google, Dough Cutting and his team developed an Open Source

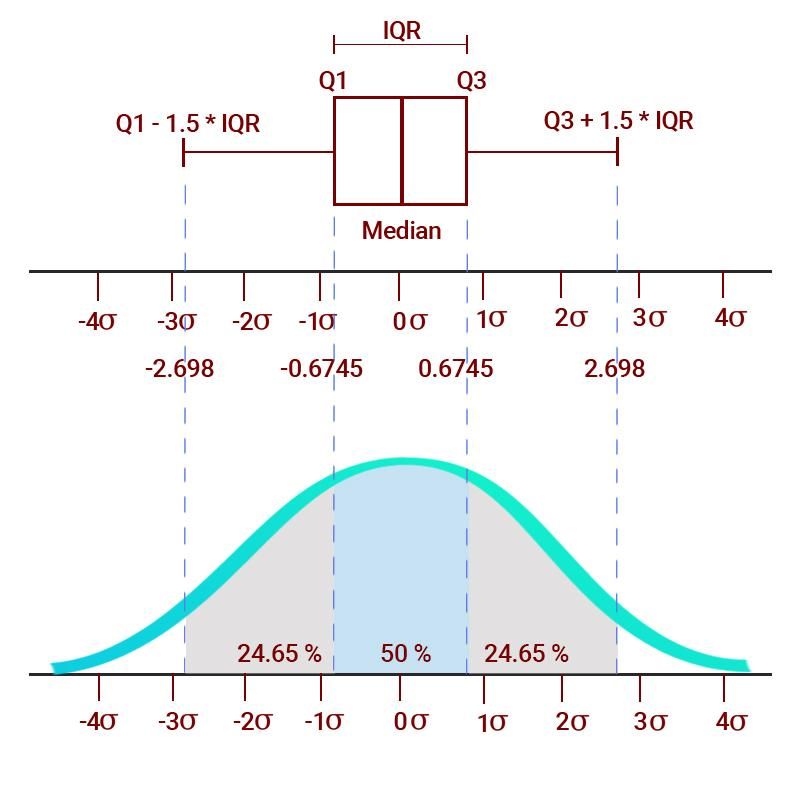
Project called Hadoop.

MapReduce Algorithm

mean --- sum and n

median

Box and whisker plot

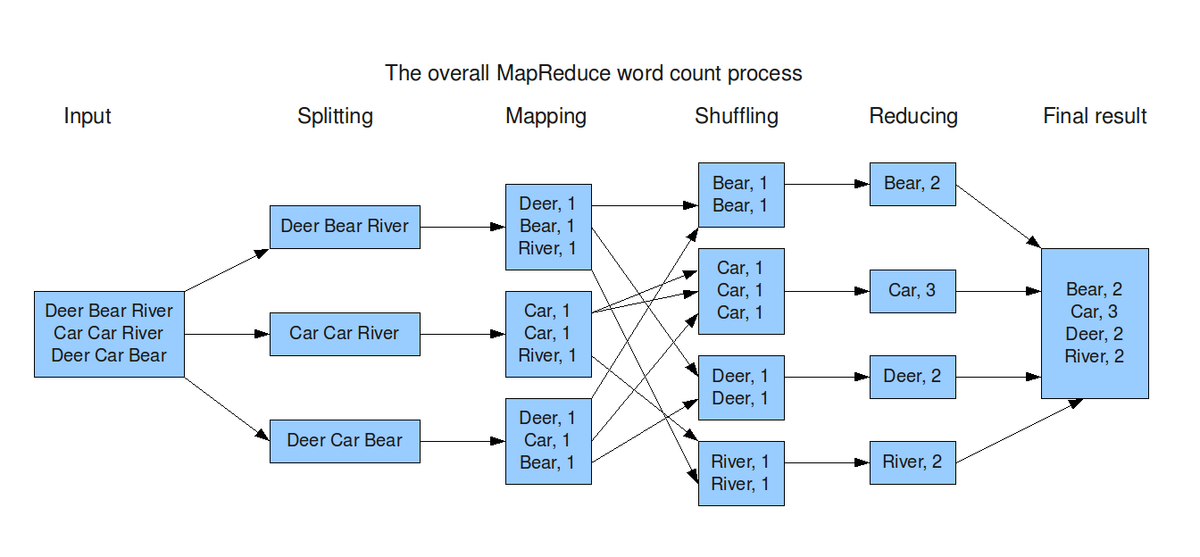


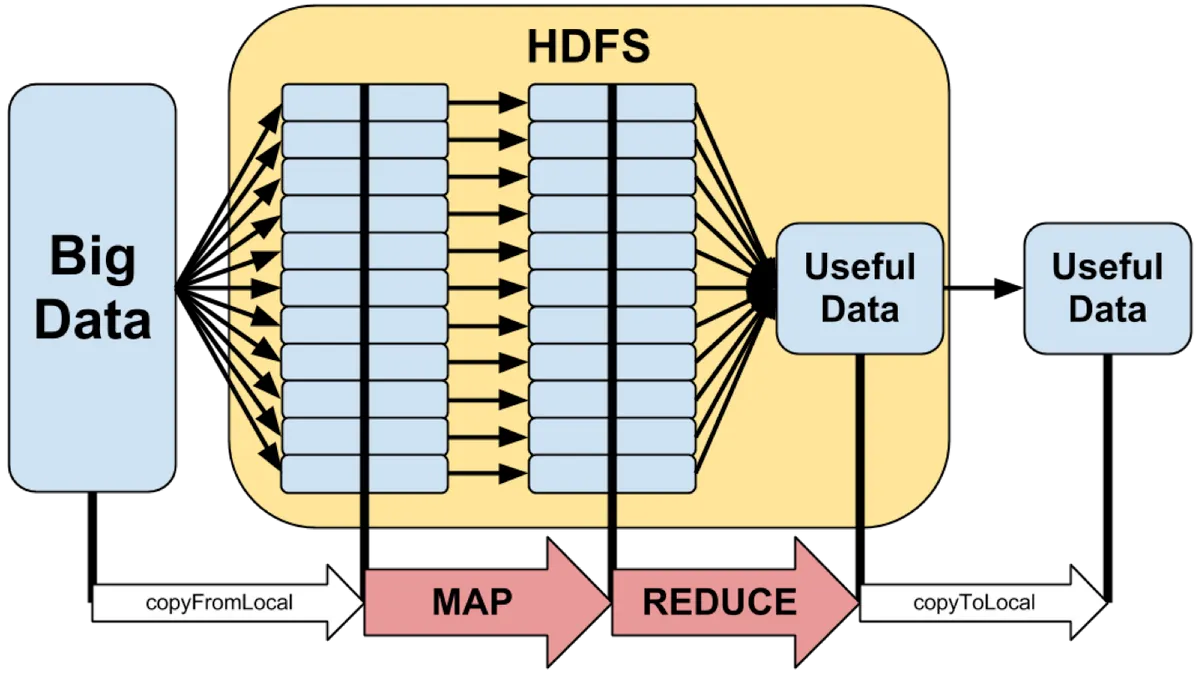
map

and

reduce

different value calculation must be done





HDFS

Hadoop Distributed File System

Hadoop Architecture

At its core, Hadoop has two major layers namely −

* Processing/Computation layer (MapReduce), and
* Storage layer (Hadoop Distributed File System).



Distributed Storage

YARN-Yet Another Resource Navigator

HDFS

DFS

<https://www.geeksforgeeks.org/what-is-dfsdistributed-file-system/>

DFS is a file system that is distributed on multiple file servers or multiple locations.



#### Features of DFS :

* **Transparency :**
  + **Structure transparency –**  
    There is no need for the client to know about the number or locations of file servers and the storage devices. Multiple file servers should be provided for performance, adaptability, and dependability.
  + **Access transparency –**  
    Both local and remote files should be accessible in the same manner. The file system should be automatically located on the accessed file and send it to the client’s side.
  + **Naming transparency –**  
    There should not be any hint in the name of the file to the location of the file. Once a name is given to the file, it should not be changed during transferring from one node to another.
  + **Replication transparency –**  
    If a file is copied on multiple nodes, both the copies of the file and their locations should be hidden from one node to another.
* **User mobility :**   
  It will automatically bring the user’s home directory to the node where the user logs in.
* **Performance :**   
  Performance is based on the average amount of time needed to convince the client requests. This time covers the CPU time + time taken to access secondary storage + network access time. It is advisable that the performance of the Distributed File System be similar to that of a centralized file system.
* **Simplicity and ease of use :**   
  The user interface of a file system should be simple and the number of commands in the file should be small.
* **High availability :**   
  A Distributed File System should be able to continue in case of any partial failures like a link failure, a node failure, or a storage drive crash.   
  A high authentic and adaptable distributed file system should have different and independent file servers for controlling different and independent storage devices.
* **Scalability :**   
  Since growing the network by adding new machines or joining two networks together is routine, the distributed system will inevitably grow over time. As a result, a good distributed file system should be built to scale quickly as the number of nodes and users in the system grows. Service should not be substantially disrupted as the number of nodes and users grows.
* **High reliability :**  
  The likelihood of data loss should be minimized as much as feasible in a suitable distributed file system. That is, because of the system’s unreliability, users should not feel forced to make backup copies of their files. Rather, a file system should create backup copies of key files that can be used if the originals are lost. Many file systems employ stable storage as a high-reliability strategy.
* **Data integrity :**  
  Multiple users frequently share a file system. The integrity of data saved in a shared file must be guaranteed by the file system. That is, concurrent access requests from many users who are competing for access to the same file must be correctly synchronized using a concurrency control method. Atomic transactions are a high-level concurrency management mechanism for data integrity that is frequently offered to users by a file system.
* **Security :**   
  A distributed file system should be secure so that its users may trust that their data will be kept private. To safeguard the information contained in the file system from unwanted & unauthorized access, security mechanisms must be implemented.
* **Heterogeneity :**  
  Heterogeneity in distributed systems is unavoidable as a result of huge scale. Users of heterogeneous distributed systems have the option of using multiple computer platforms for different purposes.

#### History :

The server component of the Distributed File System was initially introduced as an add-on feature. It was added to Windows NT 4.0 Server and was known as “DFS 4.1”. Then later on it was included as a standard component for all editions of Windows 2000 Server. Client-side support has been included in Windows NT 4.0 and also in later on version of Windows.

Linux kernels 2.6.14 and versions after it come with an SMB client VFS known as “cifs” which supports DFS. Mac OS X 10.7 (lion) and onwards supports Mac OS X DFS.

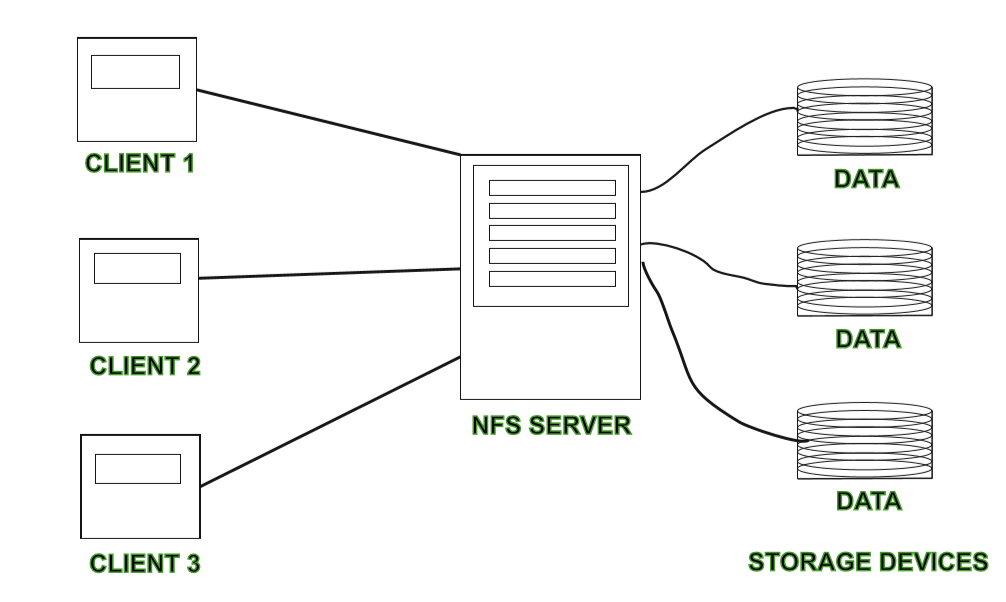
#### Applications :

* **NFS –**  
  NFS stands for Network File System. It is a client-server architecture that allows a computer user to view, store, and update files remotely. The protocol of NFS is one of the several distributed file system standards for Network-Attached Storage (NAS).
* **CIFS –**  
  CIFS stands for Common Internet File System. CIFS is an accent of SMB. That is, CIFS is an application of SIMB protocol, designed by Microsoft.
* **SMB –**  
  SMB stands for Server Message Block. It is a protocol for sharing a file and was invented by IMB. The SMB protocol was created to allow computers to perform read and write operations on files to a remote host over a Local Area Network (LAN). The directories present in the remote host can be accessed via SMB and are called as “shares”.
* **Hadoop –**  
  Hadoop is a group of open-source software services. It gives a software framework for distributed storage and operating of big data using the MapReduce programming model. The core of Hadoop contains a storage part, known as Hadoop Distributed File System (HDFS), and an operating part which is a MapReduce programming model.
* **NetWare –**  
  NetWare is an abandon computer network operating system developed by Novell, Inc. It primarily used combined multitasking to run different services on a personal computer, using the IPX network protocol.

#### Working of DFS :

There are two ways in which DFS can be implemented:

* **Standalone DFS namespace –**  
  It allows only for those DFS roots that exist on the local computer and are not using Active Directory. A Standalone DFS can only be acquired on those computers on which it is created. It does not provide any fault liberation and cannot be linked to any other DFS. Standalone DFS roots are rarely come across because of their limited advantage.
* **Domain-based DFS namespace –**  
  It stores the configuration of DFS in Active Directory, creating the DFS namespace root accessible at **\\<domainname>\<dfsroot>** or **\\<FQDN>\<dfsroot>**



Collaborative editing tool

shareLatex

Overleaf

<https://github.com/overleaf/overleaf>

Overleaf, Online LaTeX Editor

Google docs

Animation collaborative video editing

Google meet record now banned

Subscribe to use

Next Feature is

Replication

At least 3 replication factor

We want to

Read large data set

Scalability

Fault tolerance: replication

transparency

DFS

Simply a classical model of a file system that can be distributed across multiple algorithms.

Performance is concerned with throughput and response time.

FAT and NTFS

<https://www.geeksforgeeks.org/difference-between-fat32-and-ntfs/>

PU

yearly student passed failed

average marks

34151 students

Pdf excel

Web scrapping

Marks

Distributed system

Local data

Summary statistics

5 years marks data comparison

Subject wise

Batchwise

NoSQL databases

OS

Virtual Machine

File System

NTFS

HDFS ma ubuntu distribute garna milxa

GFS(Google, 2003)

Lustre(CLuster File Systems, 2003)

BeeGFS(Fraunhofer, 2005)

 Hadoop release date

April 1, 2006

Google client

Gmail

Google drive

Google photos

Microsoft

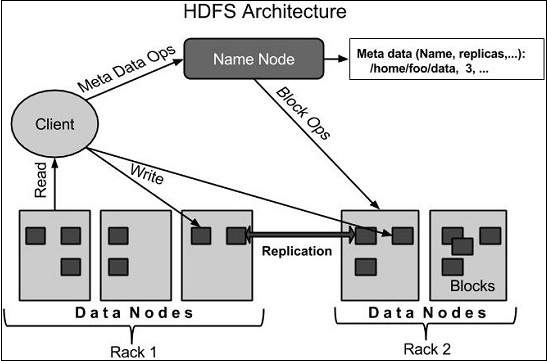
Onedrive

Components of Distributed File System

Clients-

Chunk nodes-

Master node-



<https://www.irjet.net/archives/V4/i9/IRJET-V4I957.pdf>

17 V’s of Big Data

# 5 V's of big data

Volume

Velocity: CCTV footage

Value

Variety

Veracity: authenticity of data eg. election data

Sensor

Young mountain