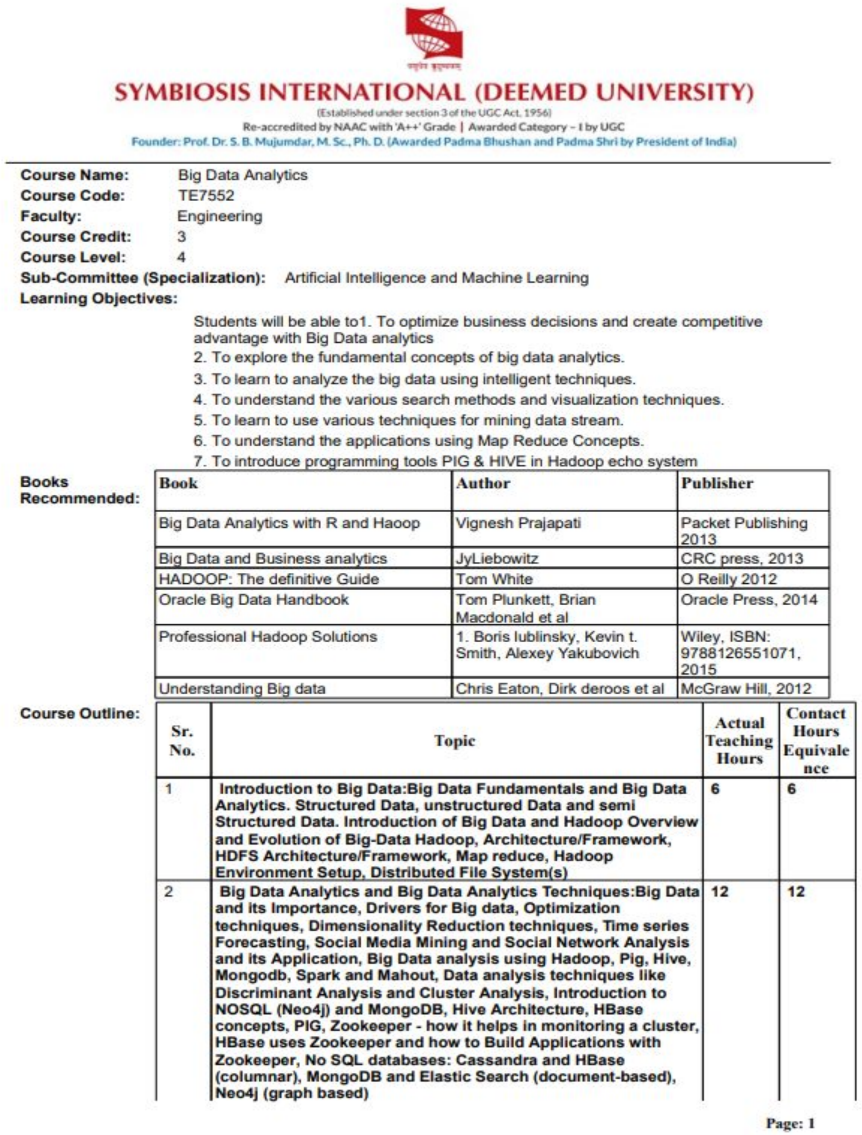
Big Data Analytics

DR. SHILPA BADE-GITE

Syllabus



SYMBIOSIS INTERNATIONAL (DEEMED UNIVERSITY)

(Established under section 3 of the UGC Act, 1956)
Re-accredited by NAAC with 'A++' Grade | Awarded Category – I by UGC
Founder: Prof. Dr. S. B. Mujumdar, M. Sc., Ph. D. (Awarded Padma Bhushan and Padma Shri by President of India)

Course Name:	Big Data Analytics
Course Code:	TE7552
Faculty:	Engineering
Course Credit:	3
Course Level:	4
Sub-Committee (Specialization):	Artificial Intelligence and Machine Learning

Learning Objectives:

Students will be able to1. To optimize business decisions and create competitive advantage with Big Data analytics

2. To explore the fundamental concepts of big data analytics.
3. To learn to analyze the big data using intelligent techniques.
4. To understand the various search methods and visualization techniques.
5. To learn to use various techniques for mining data stream.
6. To understand the applications using Map Reduce Concepts.
7. To introduce programming tools PIG & HIVE in Hadoop echo system

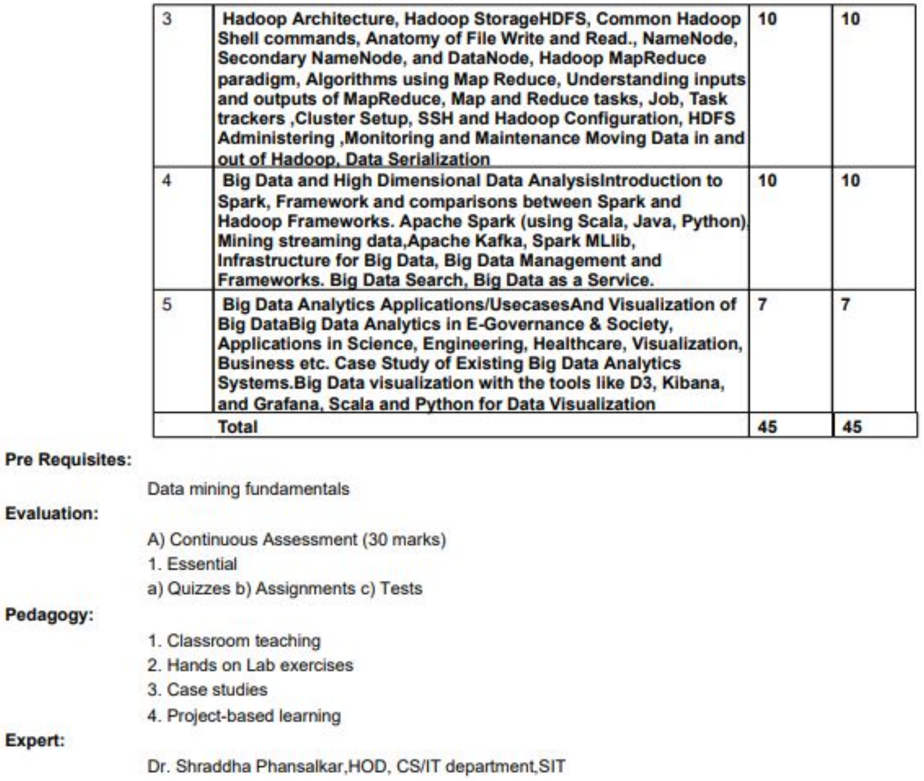
Books Recommended:

Book	Author	Publisher
Big Data Analytics with R and Haoop	Vignesh Prajapati	Packet Publishing 2013
Big Data and Business analytics	JyLiebowitz	CRC press, 2013
HADOOP: The definitive Guide	Tom White	O Reilly 2012
Oracle Big Data Handbook	Tom Plunkett, Brian Macdonald et al	Oracle Press, 2014
Professional Hadoop Solutions	1. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich	Wiley, ISBN: 9788126551071, 2015
Understanding Big data	Chris Eaton, Dirk deroos et al	McGraw Hill, 2012

Course Outline:

Sr. No.	Topic	Actual Teaching Hours	Contact Hours Equivalence
1	**Introduction to Big Data:Big Data Fundamentals and Big Data Analytics. Structured Data, unstructured Data and semi Structured Data. Introduction of Big Data and Hadoop Overview and Evolution of Big-Data Hadoop, Architecture/Framework, HDFS Architecture/Framework, Map reduce, Hadoop Environment Setup, Distributed File System(s)**	**6**	**6**
2	**Big Data Analytics and Big Data Analytics Techniques:Big Data and its Importance, Drivers for Big data, Optimization techniques, Dimensionality Reduction techniques, Time series Forecasting, Social Media Mining and Social Network Analysis and its Application, Big Data analysis using Hadoop, Pig, Hive, Mongodb, Spark and Mahout, Data analysis techniques like Discriminant Analysis and Cluster Analysis, Introduction to NOSQL (Neo4j) and MongoDB, Hive Architecture, HBase concepts, PIG, Zookeeper - how it helps in monitoring a cluster, HBase uses Zookeeper and how to Build Applications with Zookeeper, No SQL databases: Cassandra and HBase (columnar), MongoDB and Elastic Search (document-based), Neo4j (graph based)**	**12**	**12**

Page: 1



3	Hadoop Architecture, Hadoop StorageHDFS, Common Hadoop Shell commands, Anatomy of File Write and Read., NameNode, Secondary NameNode, and DataNode, Hadoop MapReduce paradigm, Algorithms using Map Reduce, Understanding inputs and outputs of MapReduce, Map and Reduce tasks, Job, Task trackers ,Cluster Setup, SSH and Hadoop Configuration, HDFS Administering ,Monitoring and Maintenance Moving Data in and out of Hadoop, Data Serialization	10	10
4	Big Data and High Dimensional Data AnalysisIntroduction to Spark, Framework and comparisons between Spark and Hadoop Frameworks. Apache Spark (using Scala, Java, Python), Mining streaming data,Apache Kafka, Spark MLlib, Infrastructure for Big Data, Big Data Management and Frameworks. Big Data Search, Big Data as a Service.	10	10
5	Big Data Analytics Applications/UsecasesAnd Visualization of Big DataBig Data Analytics in E-Governance & Society, Applications in Science, Engineering, Healthcare, Visualization, Business etc. Case Study of Existing Big Data Analytics Systems.Big Data visualization with the tools like D3, Kibana, and Grafana, Scala and Python for Data Visualization	7	7
	Total	**45**	**45**

Pre Requisites:

Data mining fundamentals

Evaluation:

A) Continuous Assessment (30 marks)

1. Essential

a) Quizzes b) Assignments c) Tests

Pedagogy:

1. Classroom teaching
2. Hands on Lab exercises
3. Case studies
4. Project-based learning

Expert:

Dr. Shraddha Phansalkar,HOD, CS/IT department,SIT

Mode of Conduction

- Unit 1,2 and 3- 2 credits-Dr Shilpa Bade-Gite-July-Sept 24
- Unit 4 and 5-1 credit-Mr. Amit Khedkar-Oct 7-11, 2024-3 hrs daily

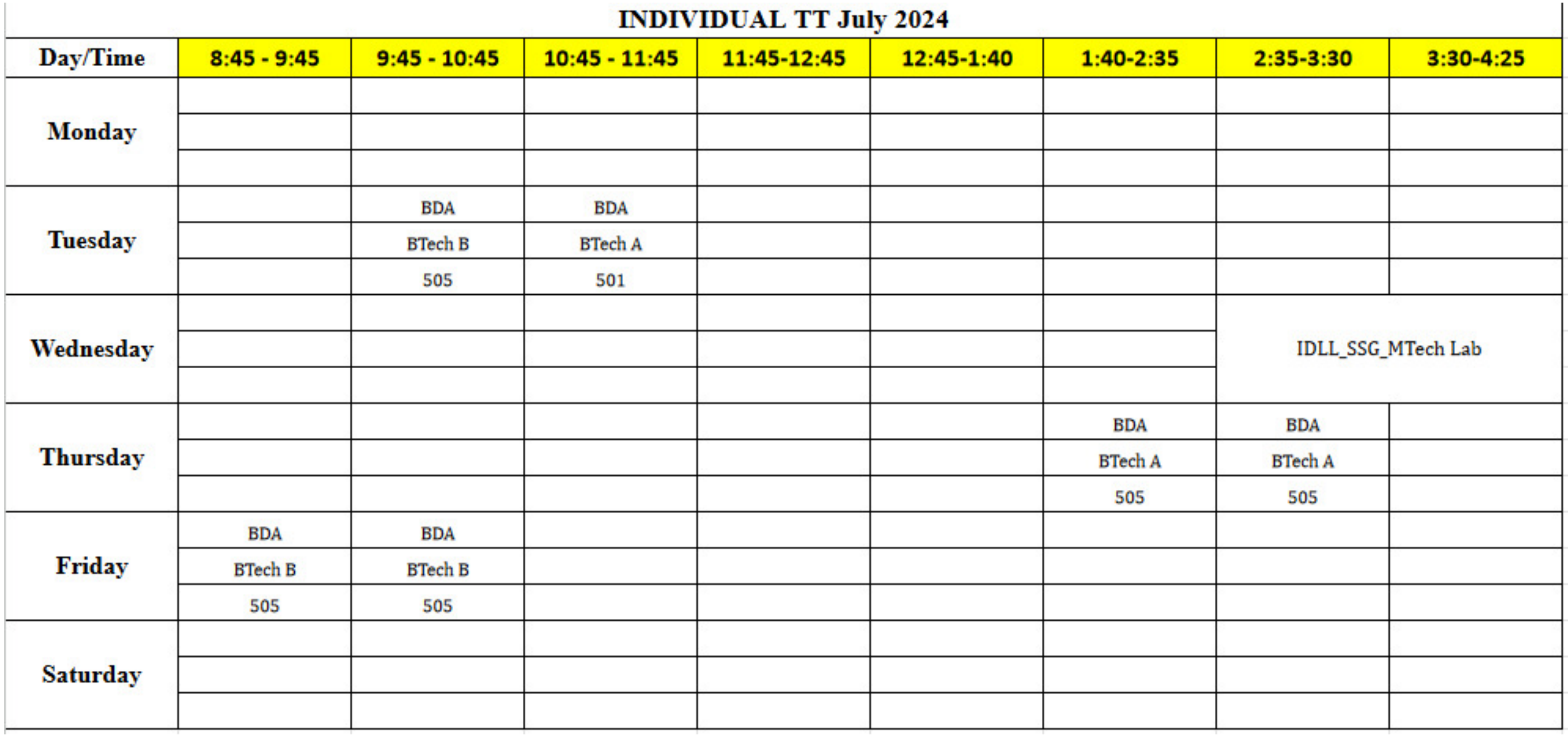
Amit Khedkar’s profile

https://www.linkedin.com/in/amit-khedkar-023758166/?originalSubdomain=in

Director and Lead Instructor at Talentum Global Technologies

akhedkar@talentumglobal.com

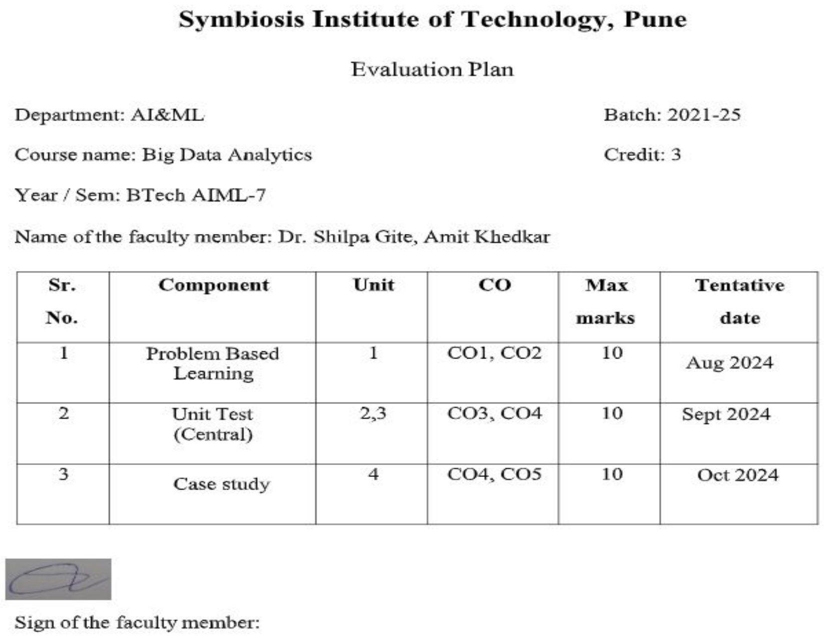
My Timetable



INDIVIDUAL TT July 2024

Day/Time	8:45 - 9:45	9:45 - 10:45	10:45 - 11:45	11:45-12:45	12:45-1:40	1:40-2:35	2:35-3:30	3:30-4:25
Monday								
Tuesday		BDA BTech B 505	BDA BTech A 501					
Wednesday							IDLL_SSG_MTech Lab	
Thursday						BDA BTech A 505	BDA BTech A 505	
Friday	BDA BTech B 505	BDA BTech B 505						
Saturday								

Evaluation Plan



Symbiosis Institute of Technology, Pune

Evaluation Plan

Department: AI&ML Batch: 2021-25

Course name: Big Data Analytics Credit: 3

Year / Sem: BTech AIML-7

Name of the faculty member: Dr. Shilpa Gite, Amit Khedkar

Sr. No.	Component	Unit	CO	Max marks	Tentative date
1	Problem Based Learning	1	CO1, CO2	10	Aug 2024
2	Unit Test (Central)	2,3	CO3, CO4	10	Sept 2024
3	Case study	4	CO4, CO5	10	Oct 2024

Sign of the faculty member:

Unit test is cancelled….

BDA Final Evaluation-30 Marks

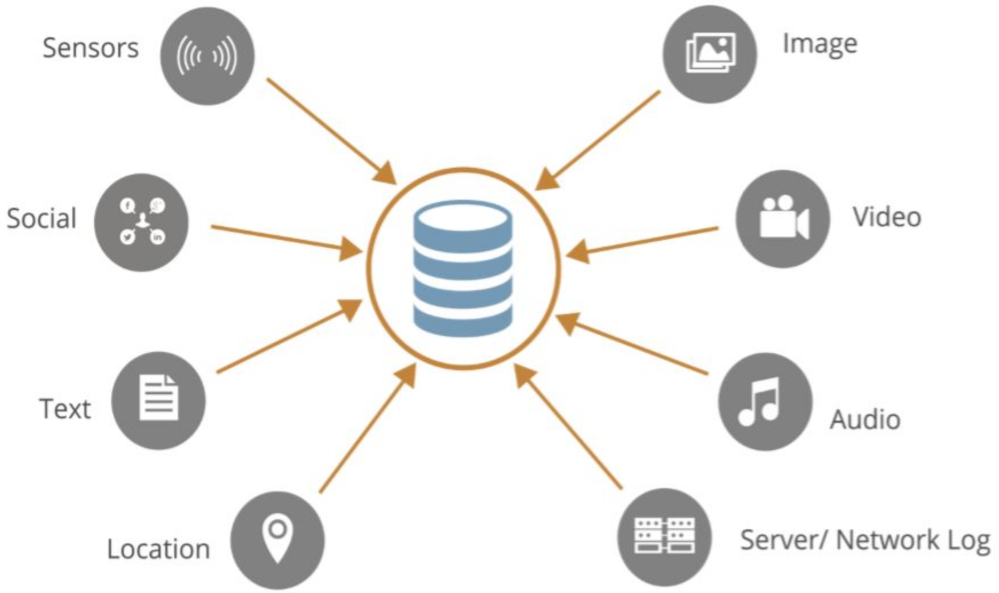
1. Quiz-CO1, CO2-Unit 1 ,Unit 2-12 Marks- Individual submission-31 Aug 24.
2. Poster-CO3-Unit 3-6 Marks- Group submission-22 Sept 24.
3. Case study-CO4,CO5-Unit4, Unit 5-5 Marks-12 Marks- Individual submission-20 Oct 24.

Unit 1-Introduction to Big Data-6 Hrs

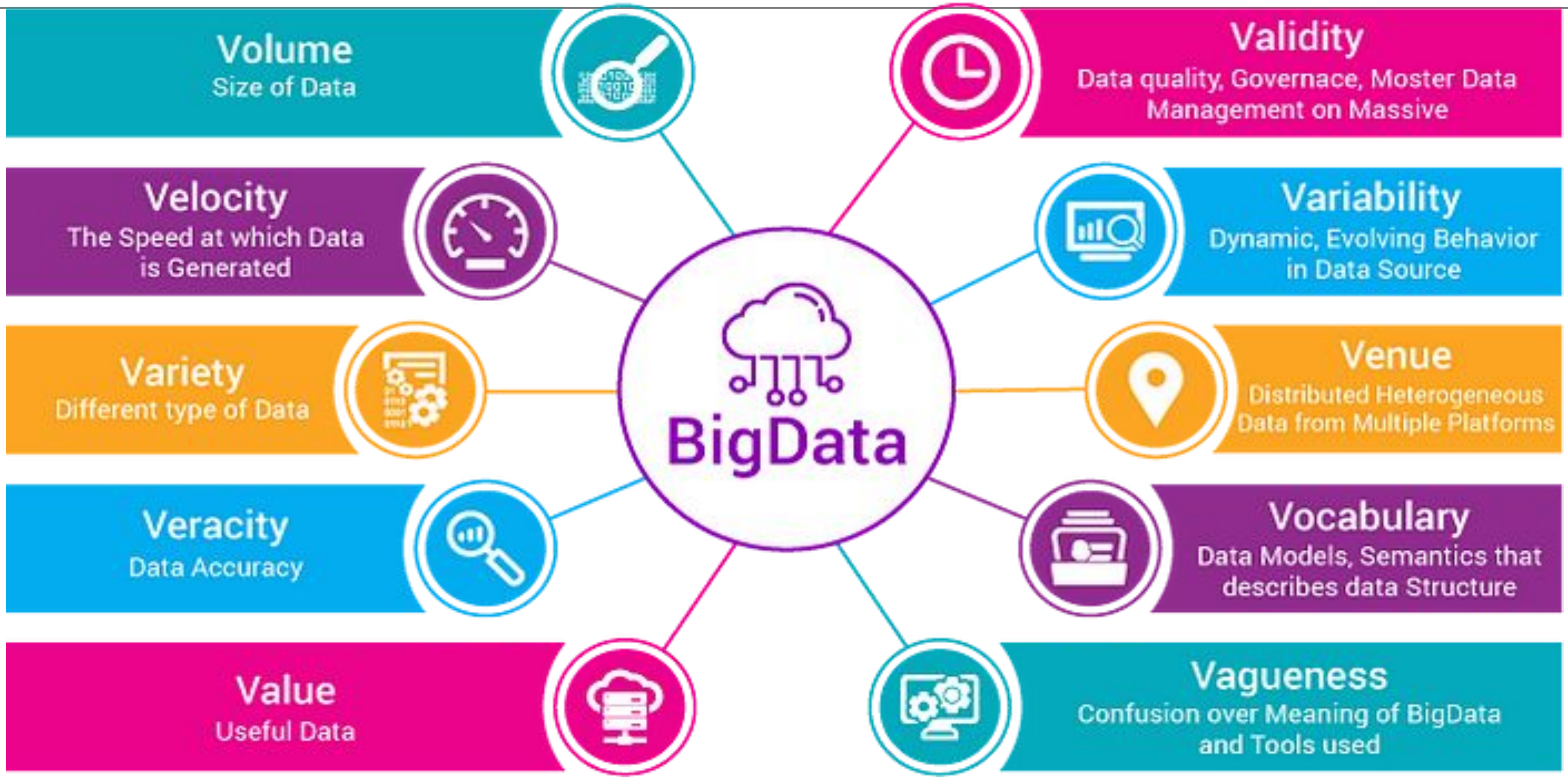
- Big Data Fundamentals and Big Data Analytics.
- Structured Data, unstructured Data and semi Structured Data.
- Hadoop Overview and Evolution of Big-Data Hadoop
- Hadoop Architecture/Framework
- HDFS
- Map reduce
- Hadoop Environment Setup
- Distributed File System(s)

“Big data is high-volume, high-velocity, and/or high-variety information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making, and process automation.”

- Gartner, Research and Advisory Company



What is Big Data?



BigData

Volume
Size of Data

Velocity
The Speed at which Data is Generated

Variety
Different type of Data

Veracity
Data Accuracy

Value
Useful Data

Validity
Data quality, Governace, Moster Data Management on Massive

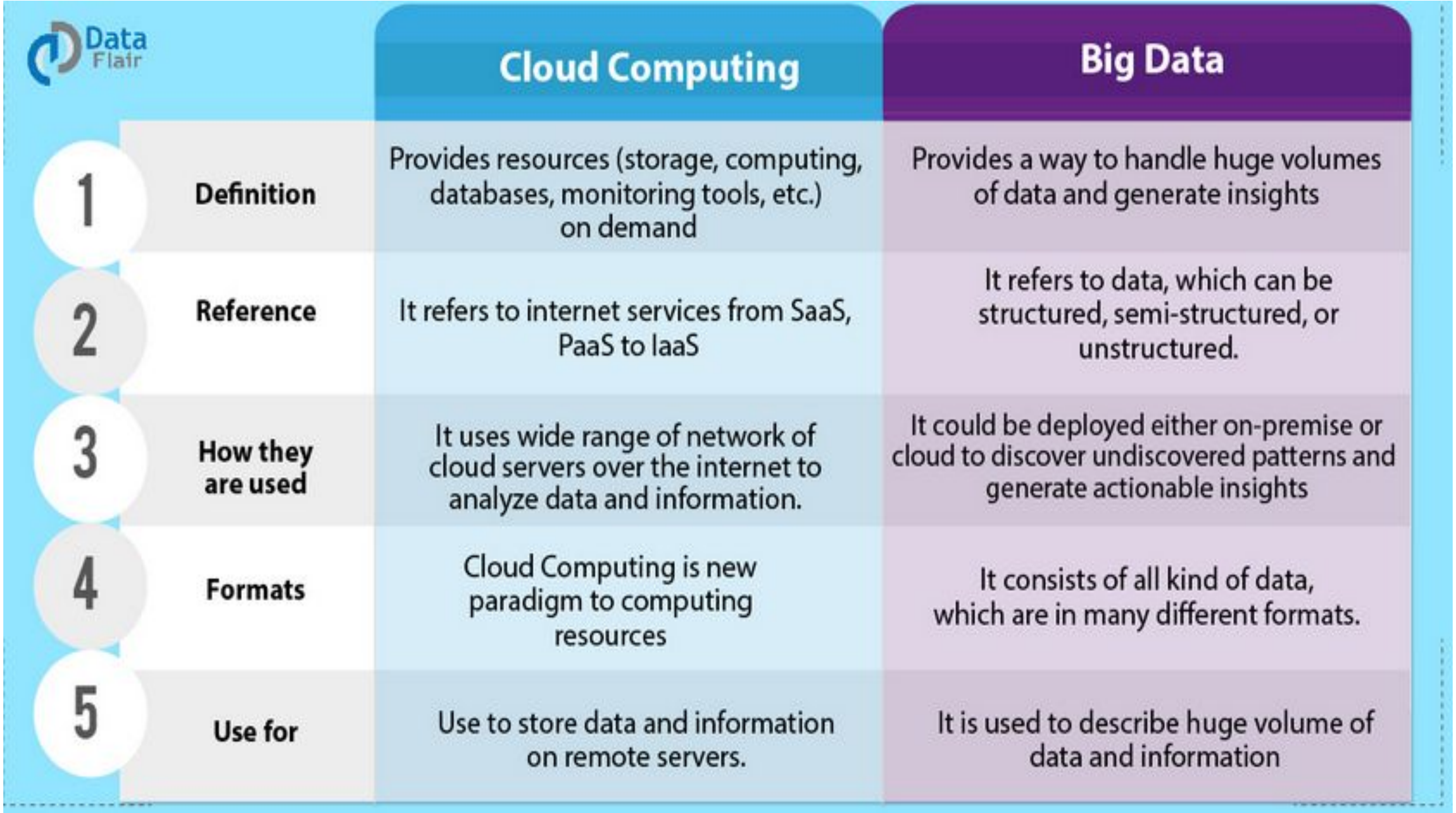
Variability
Dynamic, Evolving Behavior in Data Source

Venue
Distributed Heterogeneous Data from Multiple Platforms

Vocabulary
Data Models, Semantics that describes data Structure

Vagueness
Confusion over Meaning of BigData and Tools used

Source-https://medium.com/analysts-corner/what-is-big-data-and-why-is-it-important-to-business-41d3d0bd9d87



		Cloud Computing	Big Data
1	Definition	Provides resources (storage, computing, databases, monitoring tools, etc.) on demand	Provides a way to handle huge volumes of data and generate insights
2	Reference	It refers to internet services from SaaS, PaaS to IaaS	It refers to data, which can be structured, semi-structured, or unstructured.
3	How they are used	It uses wide range of network of cloud servers over the internet to analyze data and information.	It could be deployed either on-premise or cloud to discover undiscovered patterns and generate actionable insights
4	Formats	Cloud Computing is new paradigm to computing resources	It consists of all kind of data, which are in many different formats.
5	Use for	Use to store data and information on remote servers.	It is used to describe huge volume of data and information

Why Big Data Analytics?

- Risk Management
- Product Development and Innovations
- Quicker and Better Decision making
- Improve Customer Experience
- Complex Supplier Networks
- Focused And Targeted Campaigns

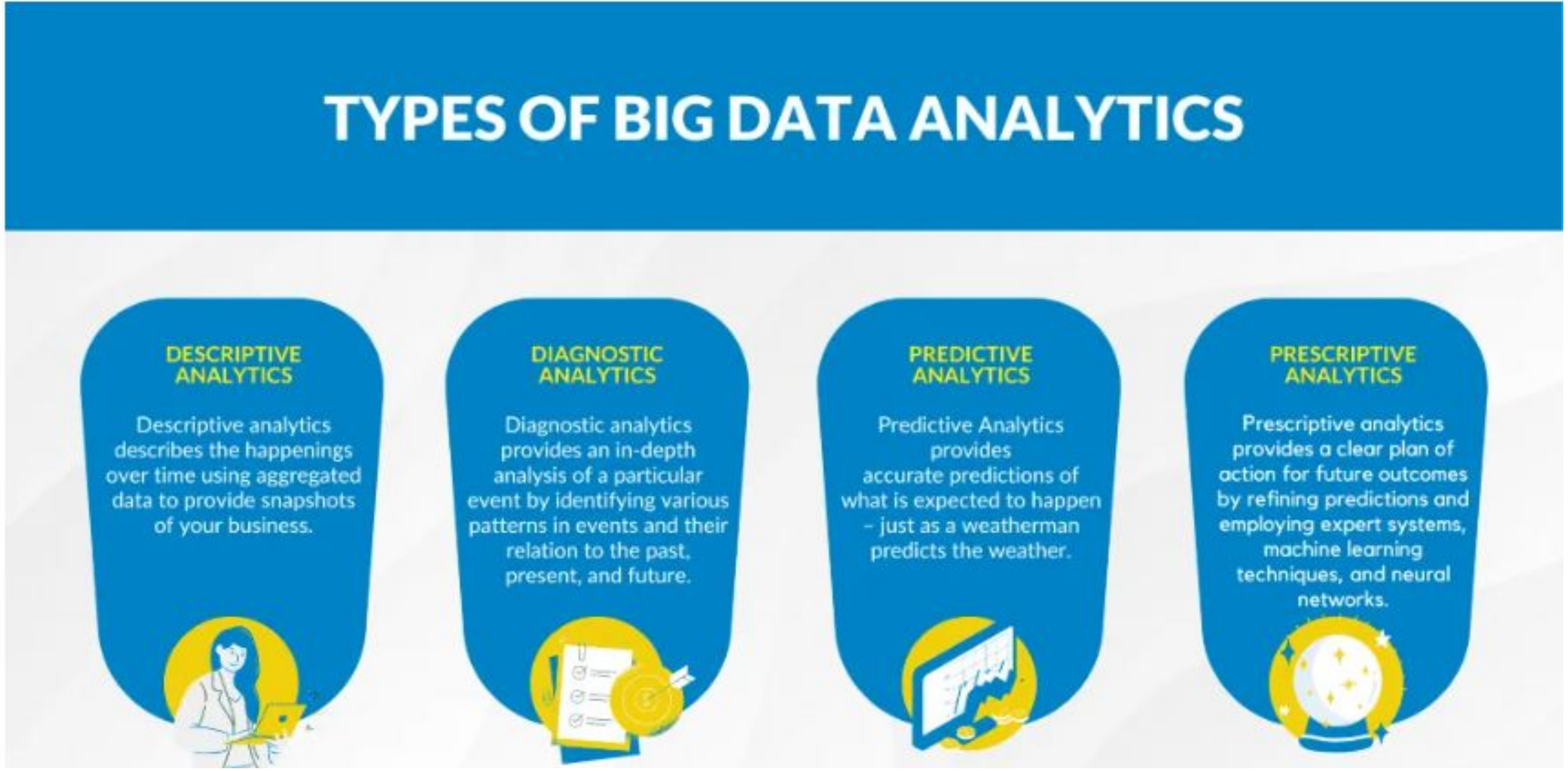
https://www.analyticssteps.com/blogs/what-big-data-analytics-definition-advantages-and-types

Types of BDA

Big data analytics is categorized into four subcategories that are:

- Descriptive Analytics
- Diagnostic Analytics
- Predictive Analytics
- Prescriptive Analytics

https://www.analyticssteps.com/blogs/what-big-data-analytics-definition-advantages-and-types



TYPES OF BIG DATA ANALYTICS

DESCRIPTIVE ANALYTICS

Descriptive analytics describes the happenings over time using aggregated data to provide snapshots of your business.

DIAGNOSTIC ANALYTICS

Diagnostic analytics provides an in-depth analysis of a particular event by identifying various patterns in events and their relation to the past, present, and future.

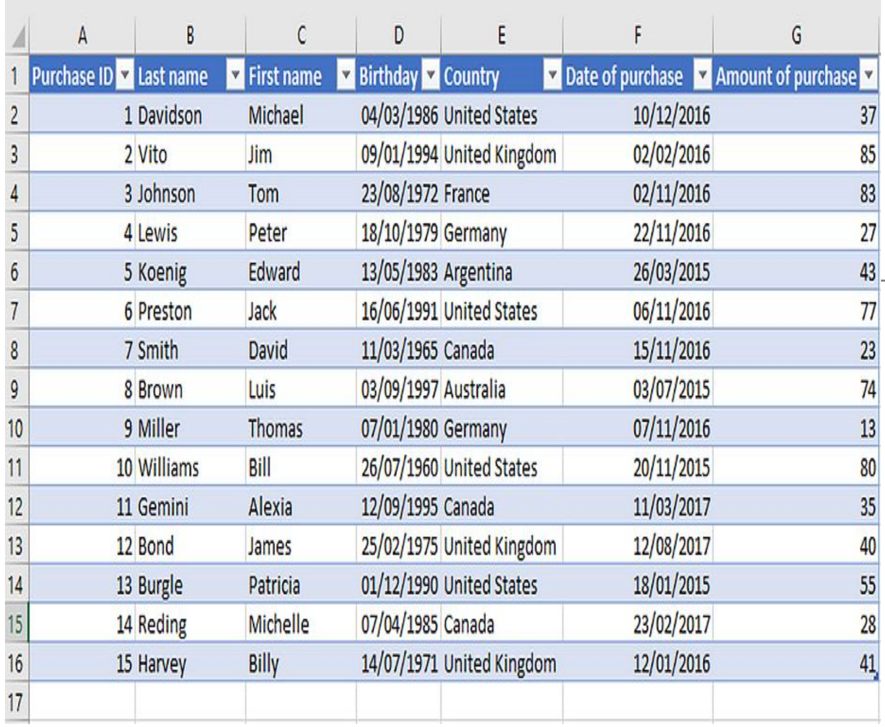
PREDICTIVE ANALYTICS

Predictive Analytics provides accurate predictions of what is expected to happen – just as a weatherman predicts the weather.

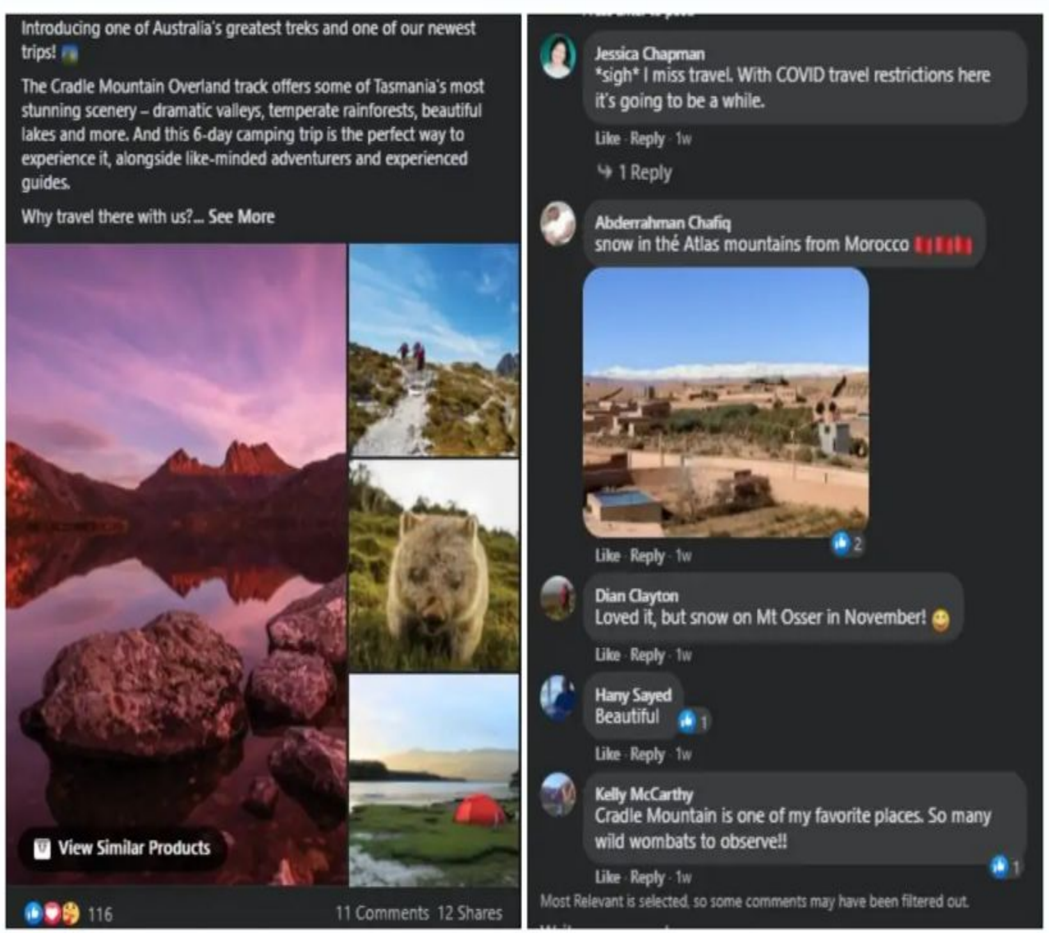
PRESCRIPTIVE ANALYTICS

Prescriptive analytics provides a clear plan of action for future outcomes by refining predictions and employing expert systems, machine learning techniques, and neural networks.

https://www.zucisystems.com/blog/big-data-analytics/



	A	B	C	D	E	F	G
1	Purchase ID	Last name	First name	Birthday	Country	Date of purchase	Amount of purchase
2	1	Davidson	Michael	04/03/1986	United States	10/12/2016	37
3	2	Vito	Jim	09/01/1994	United Kingdom	02/02/2016	85
4	3	Johnson	Tom	23/08/1972	France	02/11/2016	83
5	4	Lewis	Peter	18/10/1979	Germany	22/11/2016	27
6	5	Koenig	Edward	13/05/1983	Argentina	26/03/2015	43
7	6	Preston	Jack	16/06/1991	United States	06/11/2016	77
8	7	Smith	David	11/03/1965	Canada	15/11/2016	23
9	8	Brown	Luis	03/09/1997	Australia	03/07/2015	74
10	9	Miller	Thomas	07/01/1980	Germany	07/11/2016	13
11	10	Williams	Bill	26/07/1960	United States	20/11/2015	80
12	11	Gemini	Alexia	12/09/1995	Canada	11/03/2017	35
13	12	Bond	James	25/02/1975	United Kingdom	12/08/2017	40
14	13	Burgle	Patricia	01/12/1990	United States	18/01/2015	55
15	14	Reding	Michelle	07/04/1985	Canada	23/02/2017	28
16	15	Harvey	Billy	14/07/1971	United Kingdom	12/01/2016	41
17							



Introducing one of Australia's greatest treks and one of our newest trips!

The Cradle Mountain Overland track offers some of Tasmania's most stunning scenery – dramatic valleys, temperate rainforests, beautiful lakes and more. And this 6-day camping trip is the perfect way to experience it, alongside like-minded adventurers and experienced guides.

Why travel there with us?... See More

Jessica Chapman
sigh I miss travel. With COVID travel restrictions here it's going to be a while.
Like · Reply · 1w
1 Reply

Abderrahman Chafiq
snow in thé Atlas mountains from Morocco
Like · Reply · 1w 2

Dian Clayton
Loved it, but snow on Mt Osser in November! 😊
Like · Reply · 1w

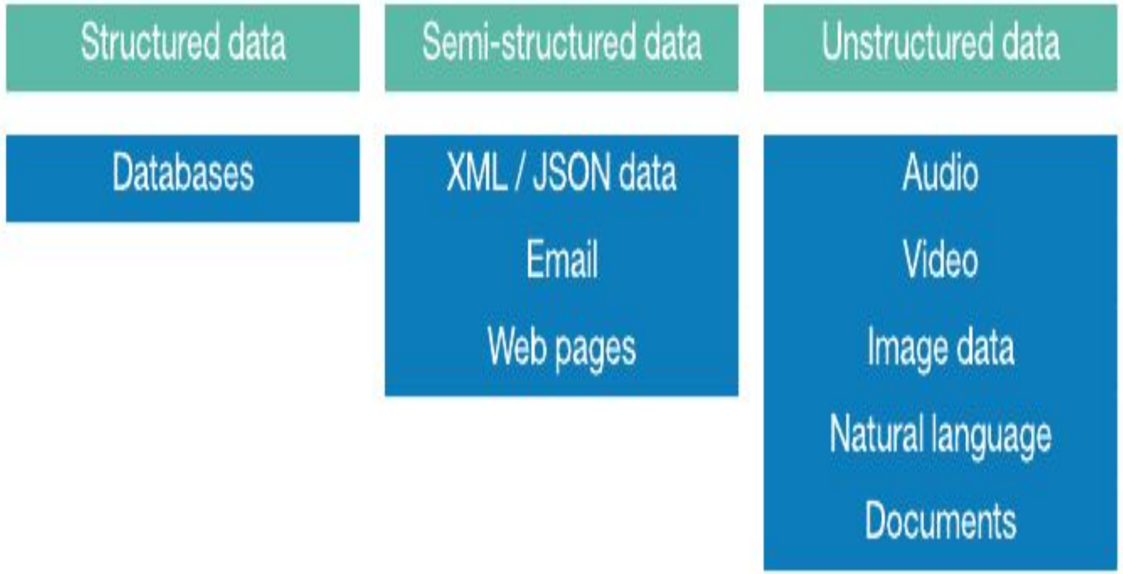
Hany Sayed
Beautiful 1
Like · Reply · 1w

Kelly McCarthy
Cradle Mountain is one of my favorite places. So many wild wombats to observe!! 1
Like · Reply · 1w

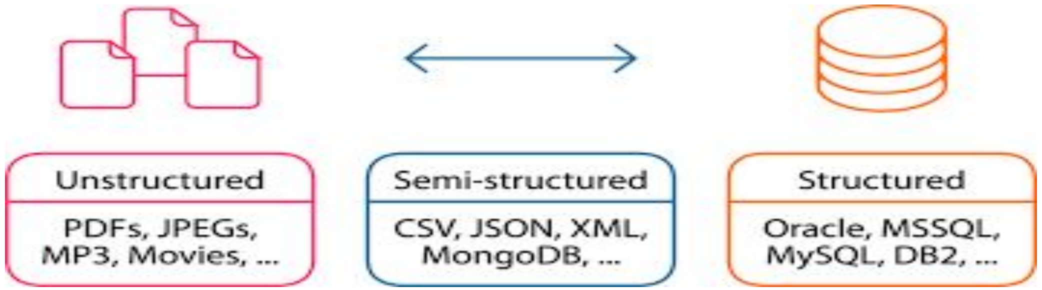
Most Relevant is selected, so some comments may have been filtered out.

The travel agency Facebook post: an example of unstructured data.

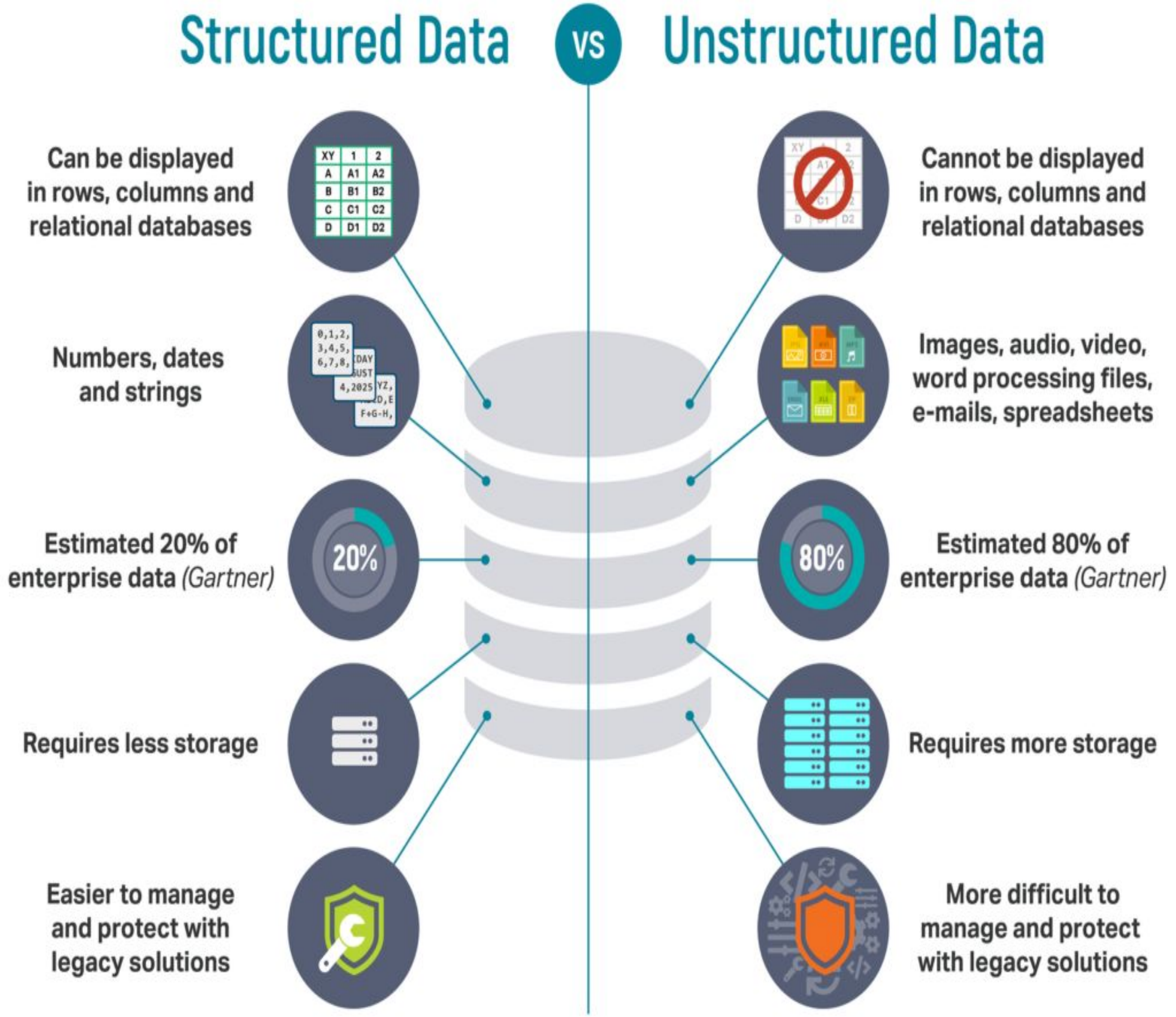
Types of Data



Structured data	Semi-structured data	Unstructured data
Databases	XML / JSON data Email Web pages	Audio Video Image data Natural language Documents



Unstructured	Semi-structured	Structured
PDFs, JPEGs, MP3, Movies, ...	CSV, JSON, XML, MongoDB, ...	Oracle, MSSQL, MySQL, DB2, ...



Structured Data	Unstructured Data
Can be displayed in rows, columns and relational databases	Cannot be displayed in rows, columns and relational databases
Numbers, dates and strings	Images, audio, video, word processing files, e-mails, spreadsheets
Estimated 20% of enterprise data *(Gartner)*	Estimated 80% of enterprise data *(Gartner)*
Requires less storage	Requires more storage
Easier to manage and protect with legacy solutions	More difficult to manage and protect with legacy solutions

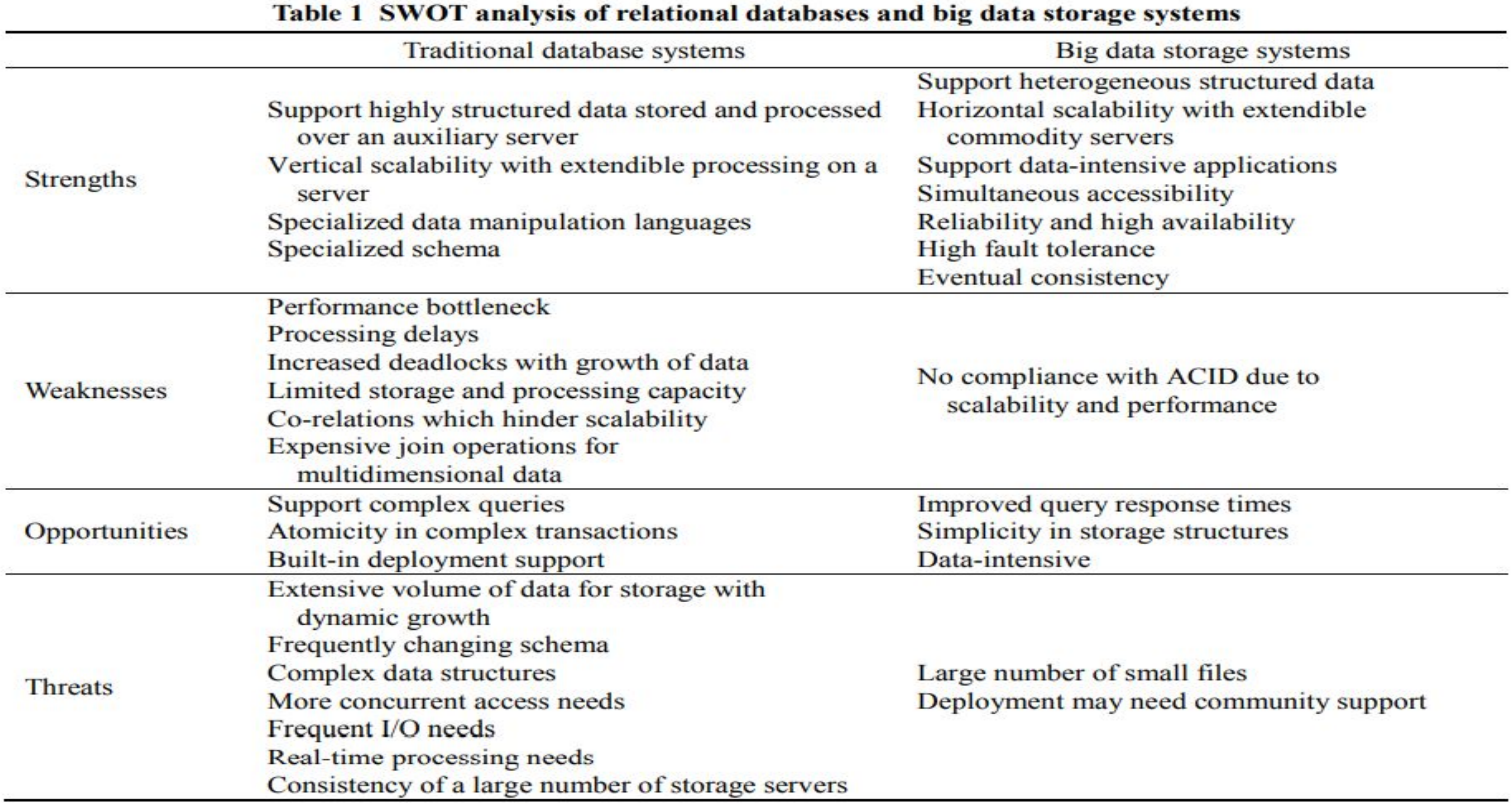
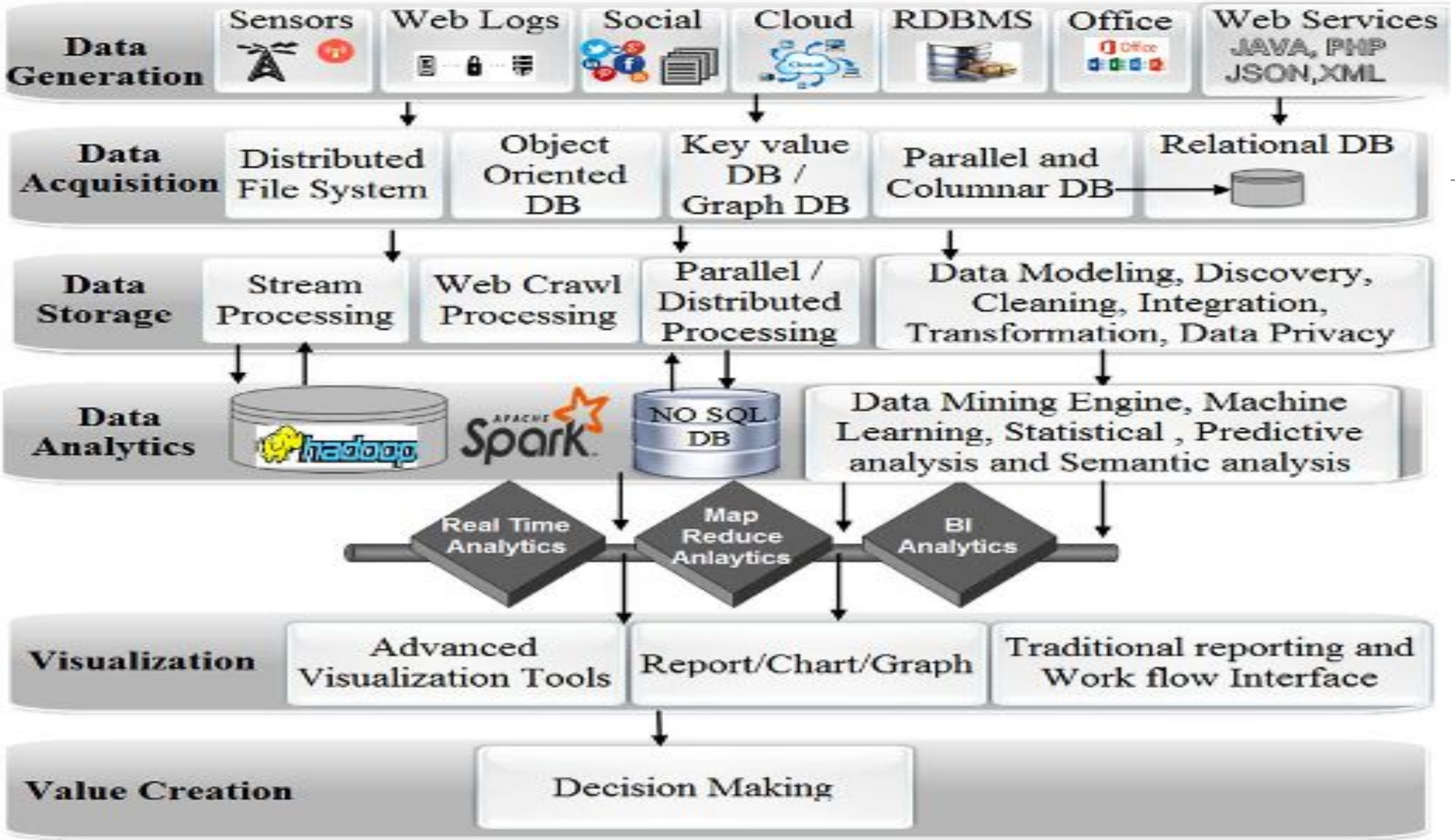
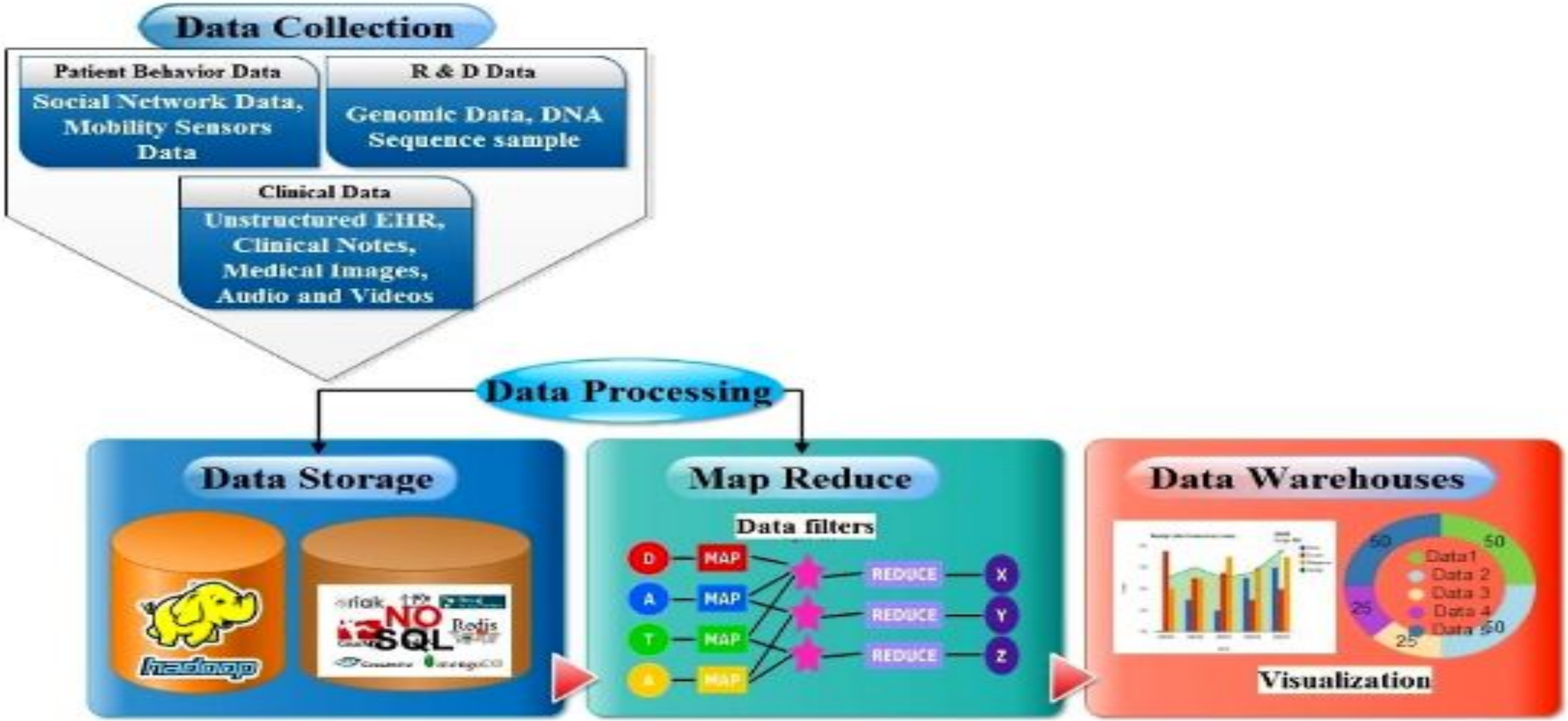


Table 1 SWOT analysis of relational databases and big data storage systems

	Traditional database systems	Big data storage systems
Strengths	Support highly structured data stored and processed over an auxiliary server Vertical scalability with extendible processing on a server Specialized data manipulation languages Specialized schema	Support heterogeneous structured data Horizontal scalability with extendible commodity servers Support data-intensive applications Simultaneous accessibility Reliability and high availability High fault tolerance Eventual consistency
Weaknesses	Performance bottleneck Processing delays Increased deadlocks with growth of data Limited storage and processing capacity Co-relations which hinder scalability Expensive join operations for multidimensional data	No compliance with ACID due to scalability and performance
Opportunities	Support complex queries Atomicity in complex transactions Built-in deployment support	Improved query response times Simplicity in storage structures Data-intensive
Threats	Extensive volume of data for storage with dynamic growth Frequently changing schema Complex data structures More concurrent access needs Frequent I/O needs Real-time processing needs Consistency of a large number of storage servers	Large number of small files Deployment may need community support


Data Generation
Sensors
Web Logs
Social
Cloud
RDBMS
Office
Web Services
JAVA, PHP
JSON,XML
Data Acquisition
Distributed File System
Object Oriented DB
Key value DB / Graph DB
Parallel and Columnar DB
Relational DB
Data Storage
Stream Processing
Web Crawl Processing
Parallel / Distributed Processing
Data Modeling, Discovery, Cleaning, Integration, Transformation, Data Privacy
Data Analytics
hadoop
APACHE Spark
NO SQL DB
Data Mining Engine, Machine Learning, Statistical , Predictive analysis and Semantic analysis
Real Time Analytics
Map Reduce Anlaytics
BI Analytics
Visualization
Advanced Visualization Tools
Report/Chart/Graph
Traditional reporting and Work flow Interface
Value Creation
Decision Making

Healthcare Example

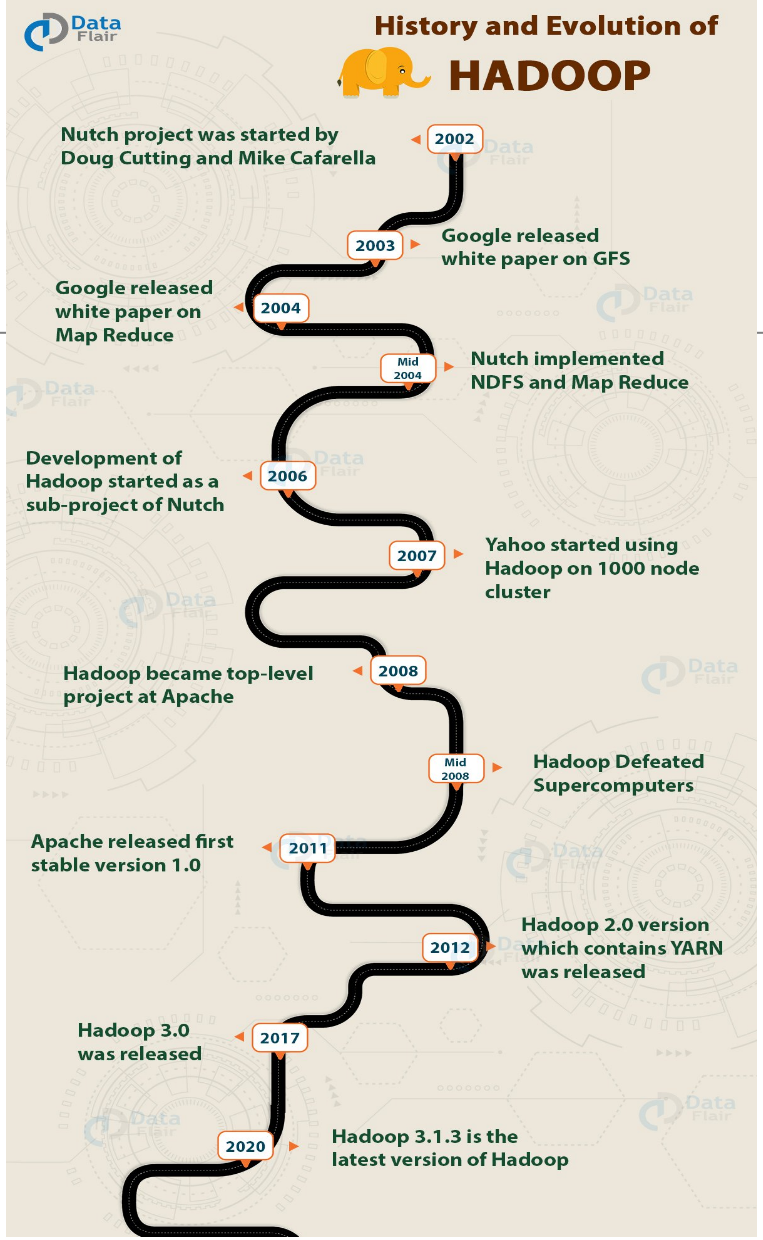


Hadoop

Hadoop is an open source framework overseen by Apache Software Foundation which is written in **Java** for storing and processing of huge datasets with the cluster of commodity hardware.

There are mainly two problems with the big data. First one is to **store** such a huge amount of data and the second one is to **process** that stored data.

There are mainly two components of Hadoop which are **Hadoop Distributed File System (HDFS)** and **Yet Another Resource Negotiator(YARN)**.



History and Evolution of HADOOP

2002 Nutch project was started by Doug Cutting and Mike Cafarella

2003 Google released white paper on GFS

2004 Google released white paper on Map Reduce

Mid 2004 Nutch implemented NDFS and Map Reduce

2006 Development of Hadoop started as a sub-project of Nutch

2007 Yahoo started using Hadoop on 1000 node cluster

2008 Hadoop became top-level project at Apache

Mid 2008 Hadoop Defeated Supercomputers

2011 Apache released first stable version 1.0

2012 Hadoop 2.0 version which contains YARN was released

2017 Hadoop 3.0 was released

2020 Hadoop 3.1.3 is the latest version of Hadoop

What is Hadoop

Hadoop is an open source framework from Apache and is used to store process and analyze data which are very huge in volume.

Hadoop is written in Java and is **not OLAP** (online analytical processing).

It is used for batch/offline processing.

It is being used by Facebook, Yahoo, Google, Twitter, LinkedIn and many more.

Moreover it can be scaled up just by adding nodes in the cluster.

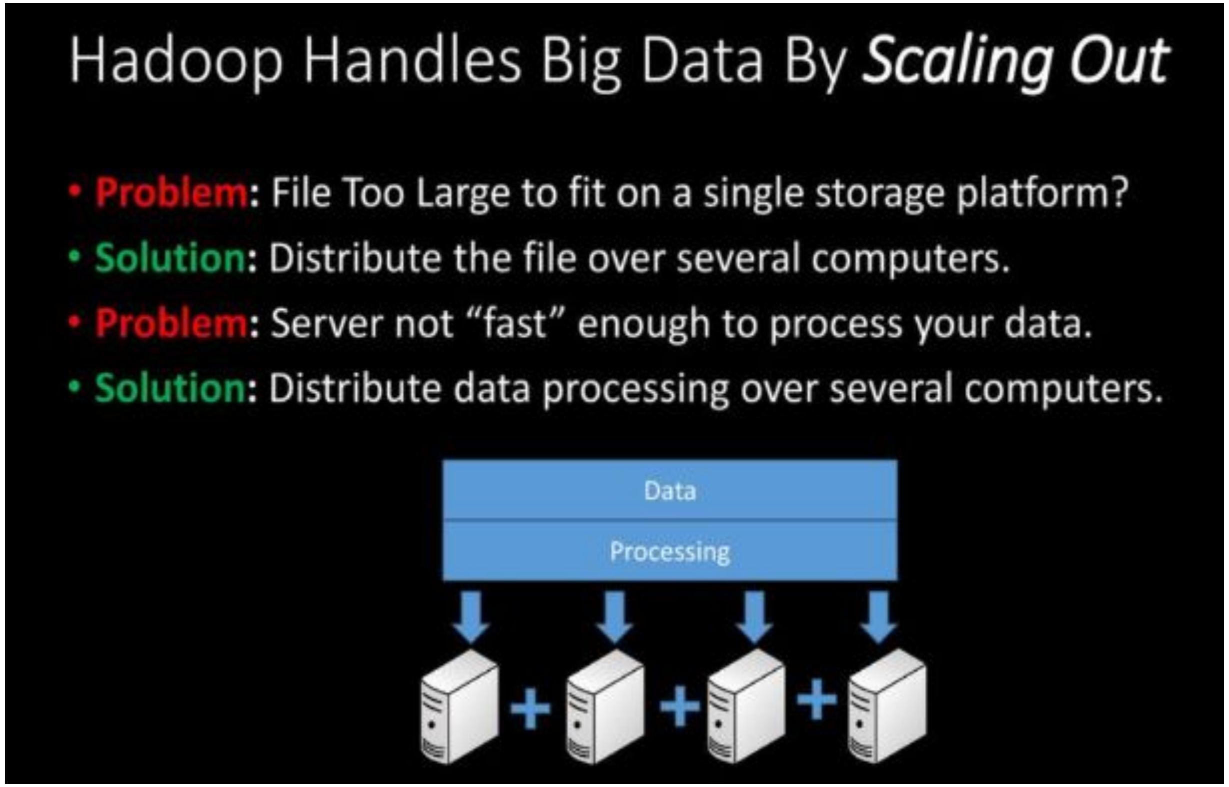
Modules of Hadoop

HDFS: Hadoop Distributed File System. Google published its paper GFS and on the basis of that HDFS was developed. It states that the files will be broken into blocks and stored in nodes over the distributed architecture.

Yarn: Yet another Resource Negotiator is used for job scheduling and manage the cluster.

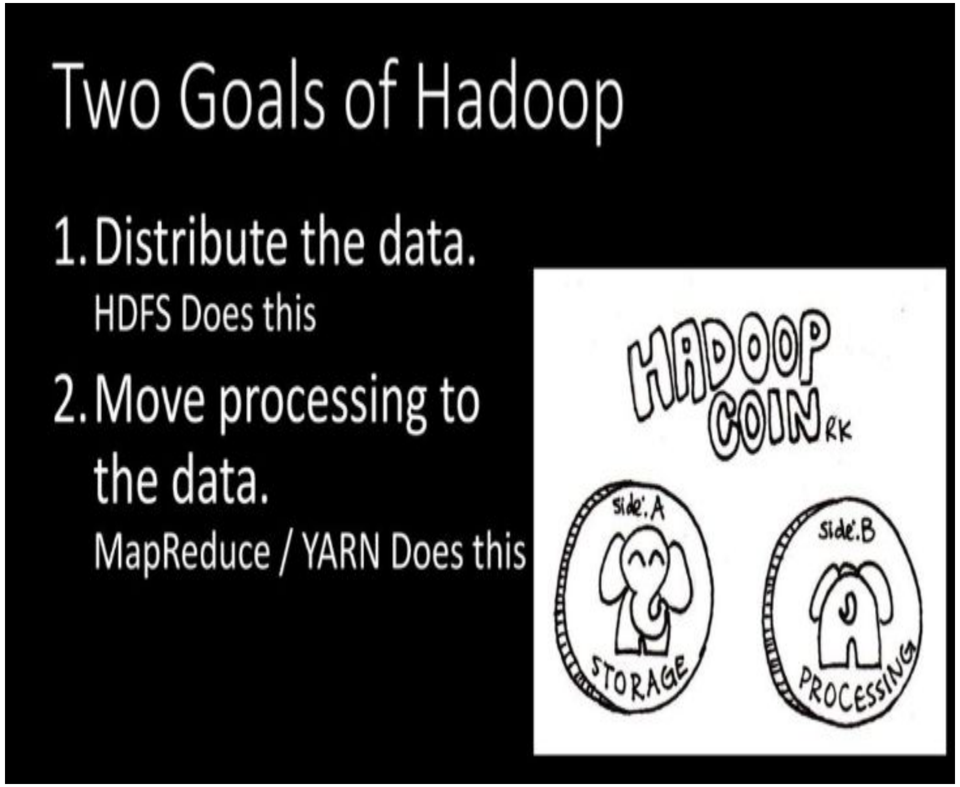
Map Reduce: This is a framework which helps Java programs to do the parallel computation on data using key value pair. The Map task takes input data and converts it into a data set which can be computed in Key value pair. The output of Map task is consumed by reduce task and then the out of reducer gives the desired result.

Hadoop Common: These Java libraries are used to start Hadoop and are used by other Hadoop modules.



Hadoop Handles Big Data By ***Scaling Out***

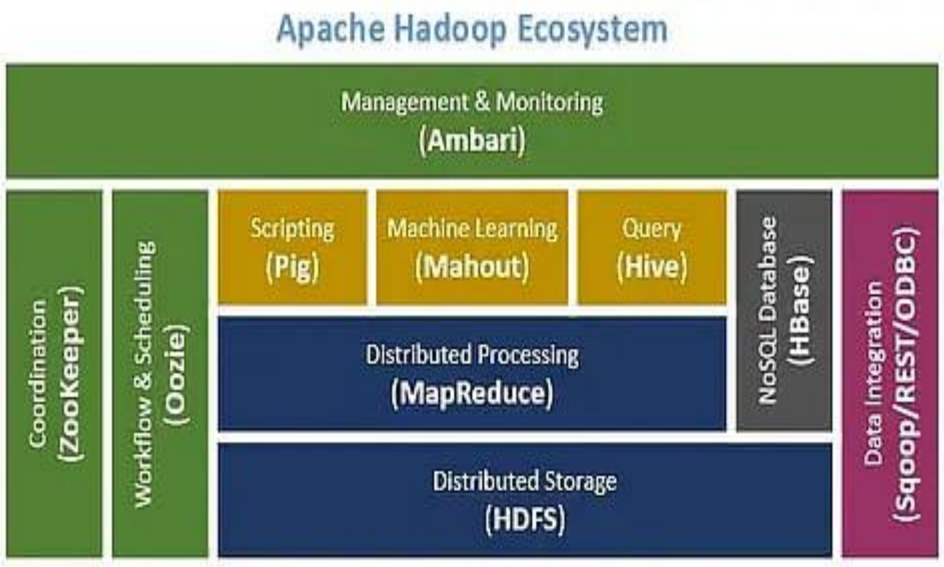
- **Problem**: File Too Large to fit on a single storage platform?
- **Solution**: Distribute the file over several computers.
- **Problem**: Server not “fast” enough to process your data.
- **Solution**: Distribute data processing over several computers.



Two Goals of Hadoop

1. Distribute the data.
   HDFS Does this
2. Move processing to the data.
   MapReduce / YARN Does this

Hadoop Architecture



Apache Hadoop Ecosystem

Management & Monitoring
(Ambari)

Coordination
(ZooKeeper)

Workflow & Scheduling
(Oozie)

Scripting
(Pig)

Machine Learning
(Mahout)

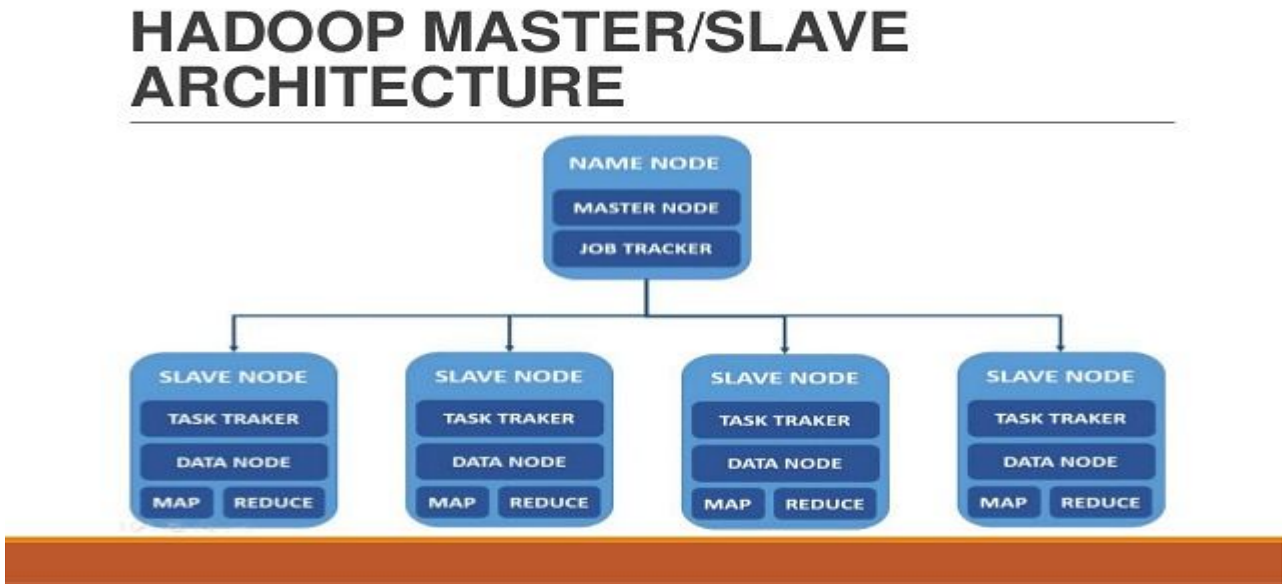
Query
(Hive)

Distributed Processing
(MapReduce)

Distributed Storage
(HDFS)

NoSQL Database
(HBase)

Data Integration
(Sqoop/REST/ODBC)



HADOOP MASTER/SLAVE ARCHITECTURE

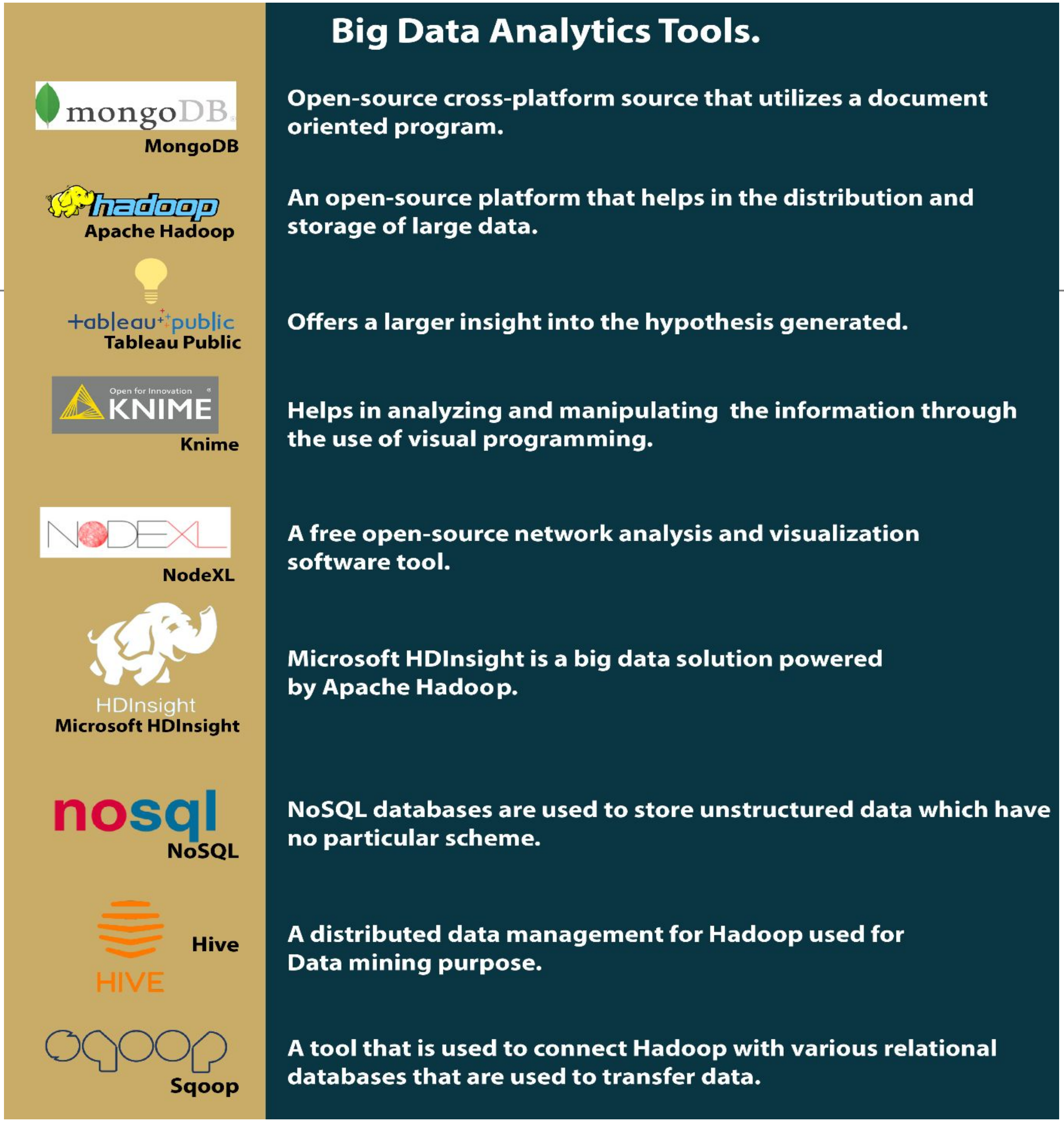
NAME NODE
MASTER NODE
JOB TRACKER

SLAVE NODE
TASK TRAKER
DATA NODE
MAP
REDUCE

SLAVE NODE
TASK TRAKER
DATA NODE
MAP
REDUCE

SLAVE NODE
TASK TRAKER
DATA NODE
MAP
REDUCE

SLAVE NODE
TASK TRAKER
DATA NODE
MAP
REDUCE


mongoDB
hadoop
+ableau+public
Open for Innovation KNIME
NODEXL
HDInsight
nosql
HIVE
sqoop

Big Data Analytics Tools.

MongoDB

Open-source cross-platform source that utilizes a document oriented program.

Apache Hadoop

An open-source platform that helps in the distribution and storage of large data.

Tableau Public

Offers a larger insight into the hypothesis generated.

Knime

Helps in analyzing and manipulating the information through the use of visual programming.

NodeXL

A free open-source network analysis and visualization software tool.

Microsoft HDInsight

Microsoft HDInsight is a big data solution powered by Apache Hadoop.

NoSQL

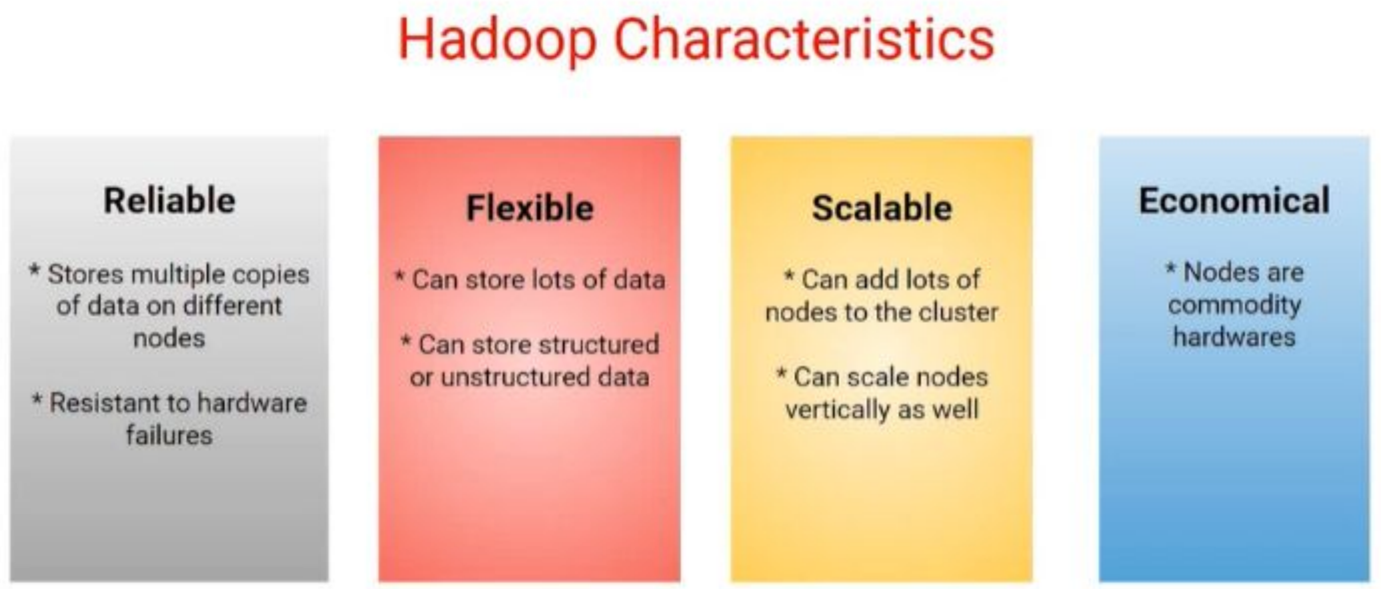
NoSQL databases are used to store unstructured data which have no particular scheme.

Hive

A distributed data management for Hadoop used for Data mining purpose.

Sqoop

A tool that is used to connect Hadoop with various relational databases that are used to transfer data.



Hadoop Characteristics

Reliable

* Stores multiple copies of data on different nodes

* Resistant to hardware failures

Flexible

* Can store lots of data

* Can store structured or unstructured data

Scalable

* Can add lots of nodes to the cluster

* Can scale nodes vertically as well

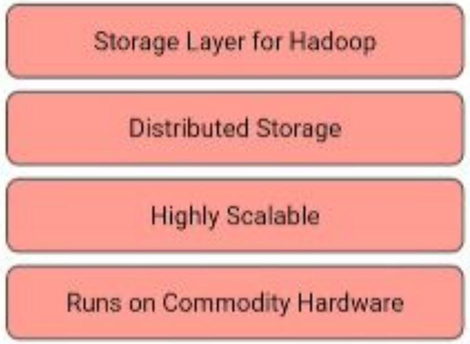
Economical

* Nodes are commodity hardwares

Hadoop Ecosystem Tools

Hadoop Distributed File System (HDFS)

- Storage Layer for Hadoop
- It stores the data in distributed manner on different machines on a cluster.
- It is highly scalable as cluster can be scaled as an when required.
- It Runs of commodity hardware which means that we do not need expensive machines and this can reduce the cost considerably



Hadoop Ecosystem Tools

Hadoop MapReduce

Main processing engine of Hadoop

Consists of two parts: Map and Reduce tasks

Fault tolerant

It can recover from any failures that happens during the execution of the job and the task

Parallel Computation

Task are computed on different machine in parallel fashion all machines do processing on certain amount of data in isolation

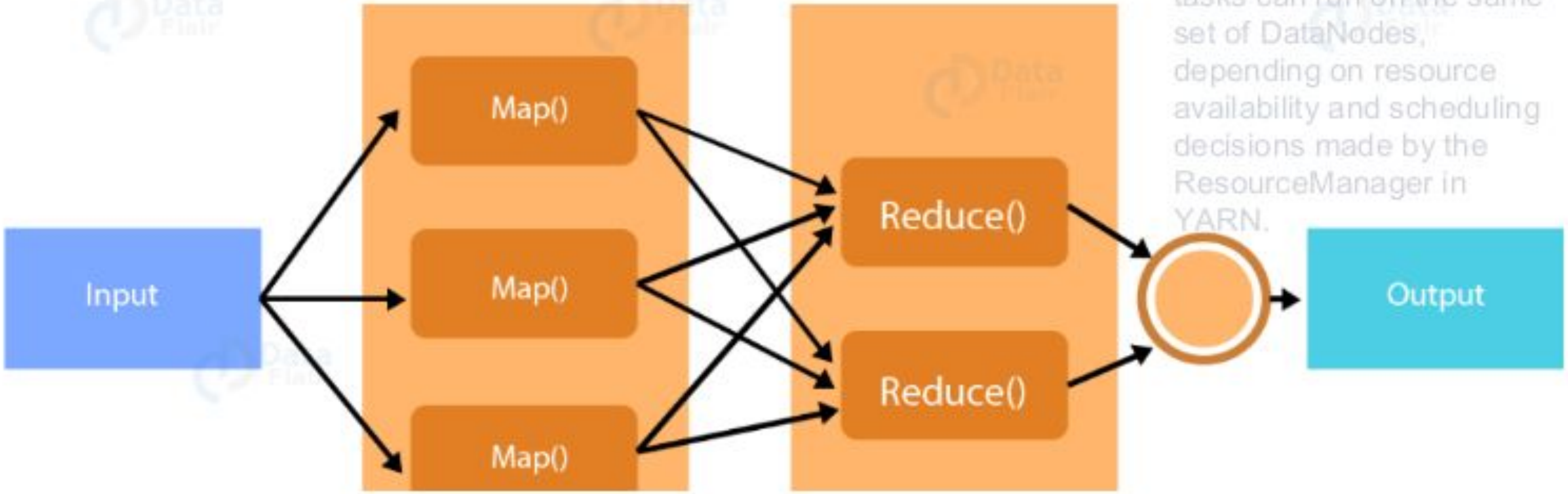
- **MapReduce** is the processing layer of **Hadoop**.
- MapReduce programming model is designed for processing large volumes of data in parallel by dividing the work into a set of independent tasks.

MapReduce works in two phases –

- **Map Phase** – This phase takes input as key-value pairs and produces output as key-value pairs. It can write custom business logic in this phase. Map phase processes the data and gives it to the next phase.
- **Reduce Phase** – The MapReduce framework sorts the key-value pair before giving the data to this phase. This phase applies the summary type of calculations to the key-value pairs.

Working of MapReduce

- Mapper reads the block of data and converts it into key-value pairs.
- Now, these key-value pairs are input to the reducer.
- The reducer receives data tuples from multiple mappers.
- Reducer applies aggregation to these tuples based on the key.
- The final output from reducer gets written to HDFS.



Both mapper and reducer tasks can run on the same set of DataNodes, depending on resource availability and scheduling decisions made by the ResourceManager in YARN.

Hadoop Ecosystem Tools

NoSQL Database

Stores data in HDFS

Random read/write

Real-time read/write

- NoSQL database build on top of HDFS
- Unlike SQL it does not store the data in tabular format and data can have any structure
- One drawback of HDFS is it access the data only in sequential manner.
- So, it consumes lot of time for large datasets
- HDFS allows us to randomly read and write data on HDFS which is much faster and less time consuming than HDFS itself
- Also, HBase provides real time read and write access of data whereas Hadoop support batch process

Hadoop Ecosystem Tools

Abstraction over Map-Reduce

Analyze large dataset

Uses Pig Latin

- Pig is a SQL like language used for querying and analyzing data stored in HDFS.
- Abstraction over MapReduce
- Its very helpful- as coding in MapReduce is difficult task and lengthy
- Developed by yahoo
- Can analyze large datasets
- It uses Pig Latin which is high level language
- Code written in Pig Latin is internally converted to MapReduce task by Pig
- Any one can execute map reduce task who do not have prior knowledge of programming language

Hadoop Ecosystem Tools

HIVE

Distributed data warehouse system

Supports Hive Query Language (HQL)

Executes queries using map-reduce.

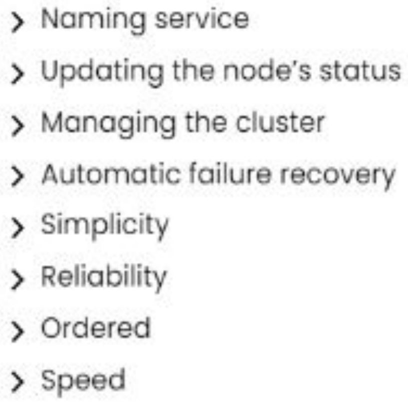
Used for Analytical Jobs

Data Exploration
data analytics

Any one can use it who do not have prior knowledge of programming language

Hadoop Ecosystem Tools

- Manages overall cluster
- Maintains the Hadoop as single unit
- Responsible for synchronizing Hadoop task
- It also serves as naming service it identifies the node in the cluster by name
- It is distributed coordination service
- Provides a centralize service for various kinds of information and distributed systems like configuration information naming synchronization etc



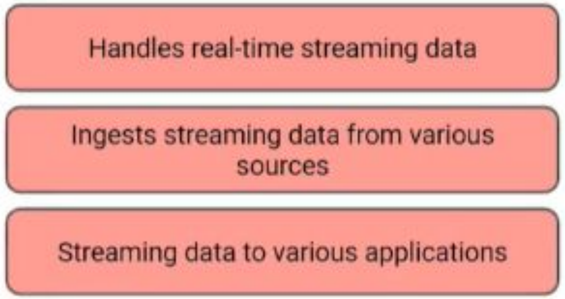
- Naming service
- Updating the node's status
- Managing the cluster
- Automatic failure recovery
- Simplicity
- Reliability
- Ordered
- Speed

Hadoop Ecosystem Tools

kafka

- Apache Kafka is a distributed data store that is highly optimize for ingesting and processing streaming data in real time.
- It ingest the streaming data from various sources and also provides streaming data to various applications as well.
- It is mostly used for monitoring operational data

A Distributed Data Store is a storage system where data is spread across multiple servers, nodes, or locations, rather than being confined to a single machine.

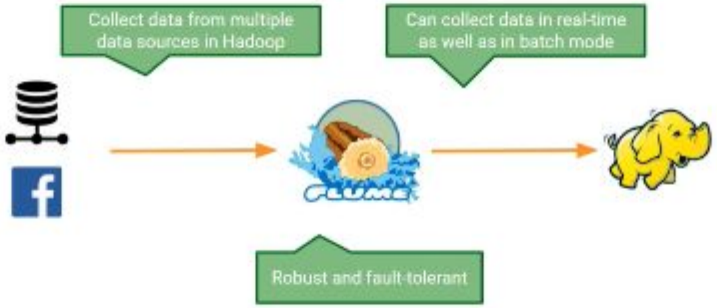


Handles real-time streaming data

Ingests streaming data from various sources

Streaming data to various applications

Hadoop Ecosystem Tools



Collect data from multiple data sources in Hadoop

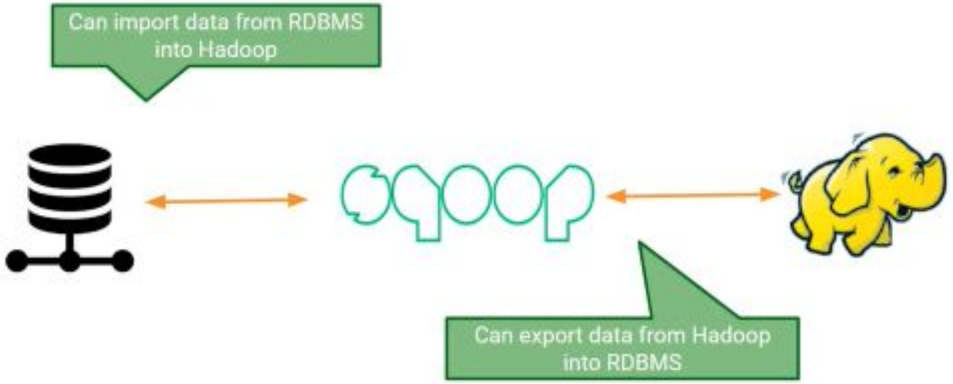
Can collect data in real-time as well as in batch mode

Robust and fault-tolerant

- Apache Flume is distributed and reliable system for efficiently collecting aggregating and moving large amounts of log data from many different sources to a centralize data store.
- It can collect data from multiple data sources in Hadoop.
- It is robust and fault tolerant and can collect data in real time as well as in batch mode.
- We can use Apache Flume to move Hugh amount of data generated by Applications servers into the Hadoop Distributed File system (HDFS) at a higher speed

Hadoop Ecosystem Tools

- Apache Sqoop tool designed to transfer data between Hadoop and relational databases
- It can be used to import data from RDBMS to such as MySQL , Oracle into HDFS.
- It transforms the data in Hadoop and MapReduce and then export the data back in RDBMS



Can import data from RDBMS into Hadoop

Can export data from Hadoop into RDBMS

Traditional Systems-Challenges

Limited Scaling

Limit to storage and processing increasing any one of them is very expensive

Higher risk of Downtime

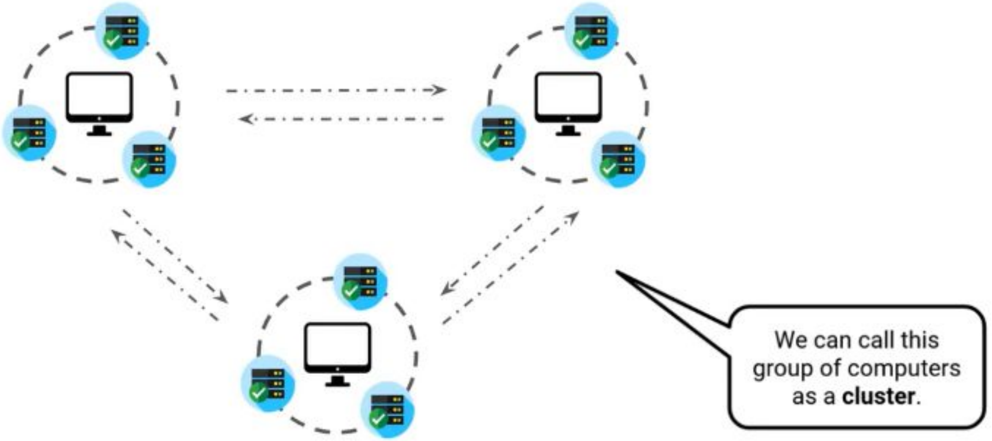
Any hardware failure data will be inaccessible as everything is stored in one place lead to potentially massive loss of business

Expensive maintenance

As everything resides on a single system makes any updation upgradation expensive as downtime will be higher

This is where distributed systems comes in

Distributed Systems

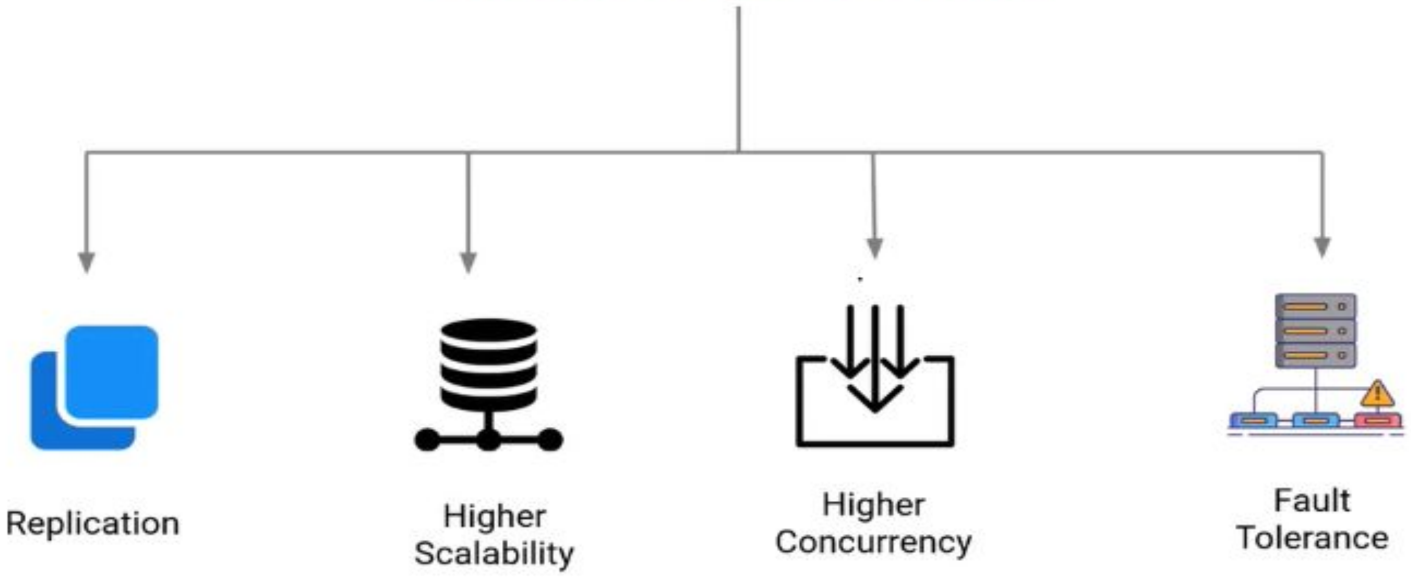


We can call this group of computers as a **cluster**.

They are group of computers or nodes working together so as to appear as a single computer to end user.

We can call this group of computers as a cluster. Lets look at few features of distributed systems

Distributed Systems



Replication

Higher Scalability

Higher Concurrency

Fault Tolerance

Data Can be replicated on more than one system

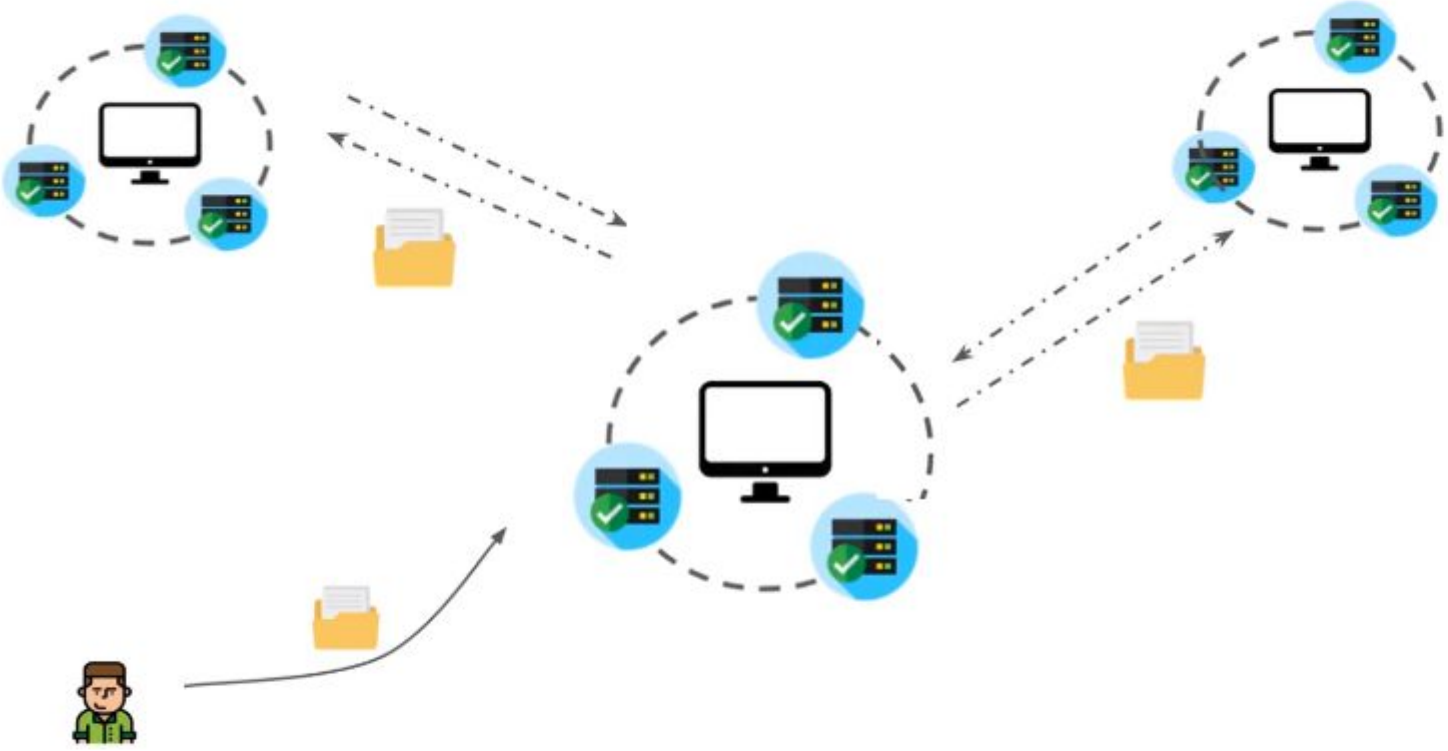
Scaling a distributed system is much easier and cheaper unlike traditional system

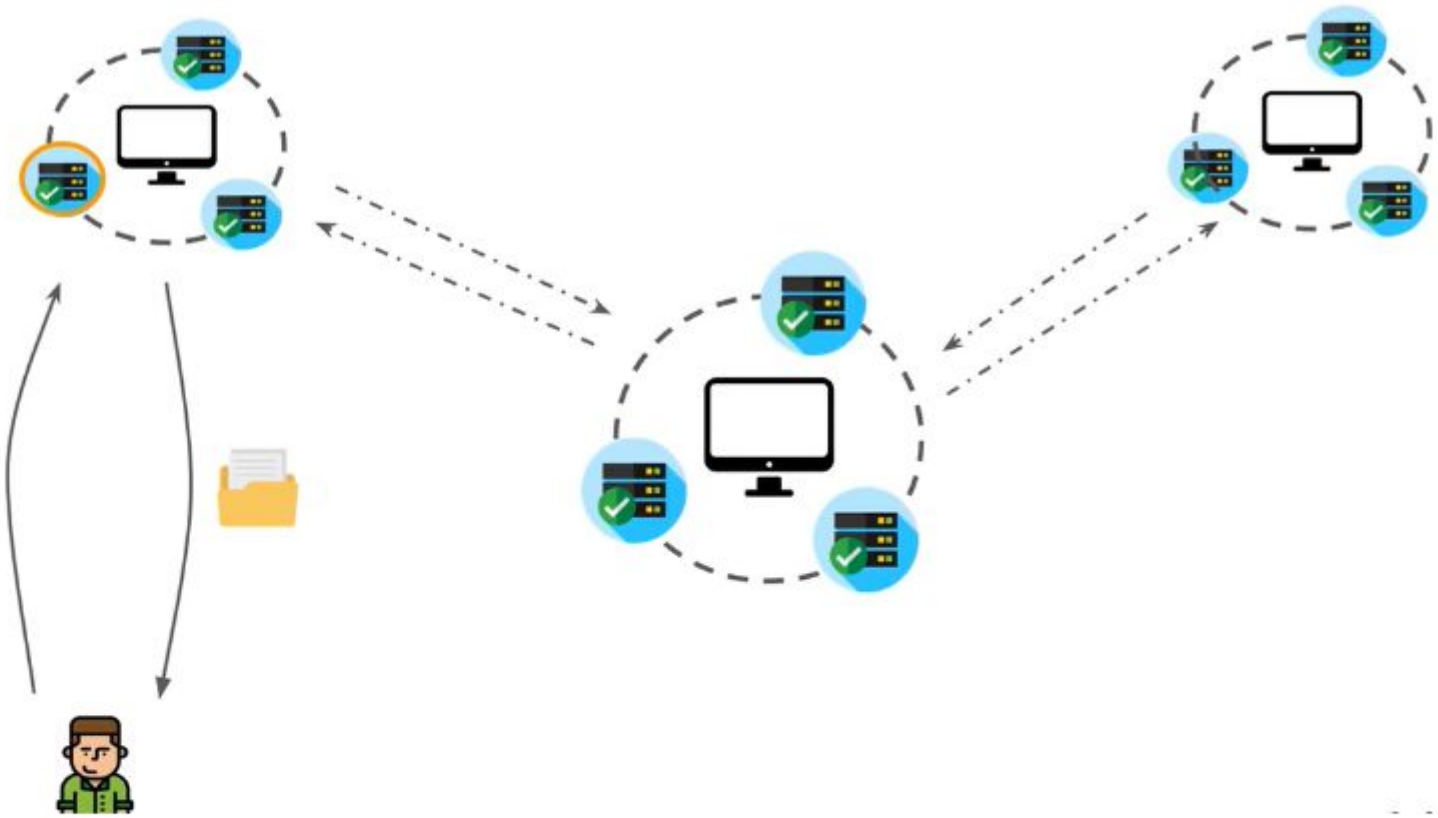
Support higher concurrency

Ability of a system to continue functioning smoothly even when one or more components fail.

Replication

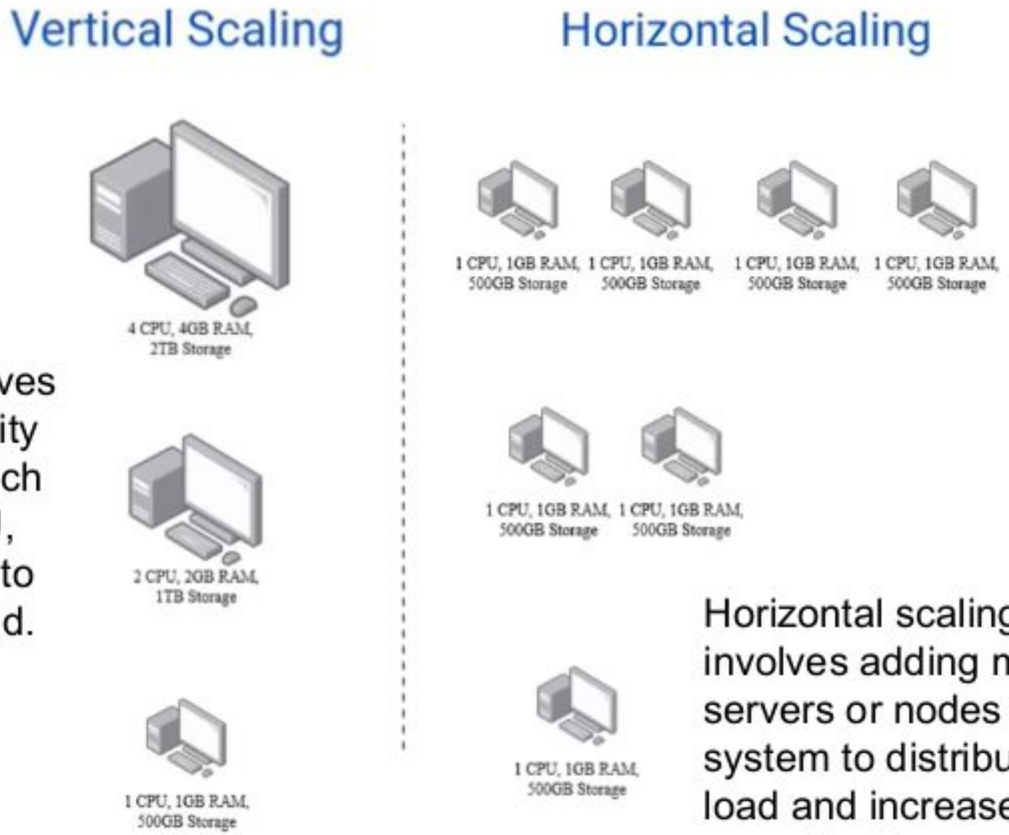
Replication in a distributed system involves creating multiple copies of data across different nodes to enhance availability, fault tolerance, and performance.





Higher Scalability

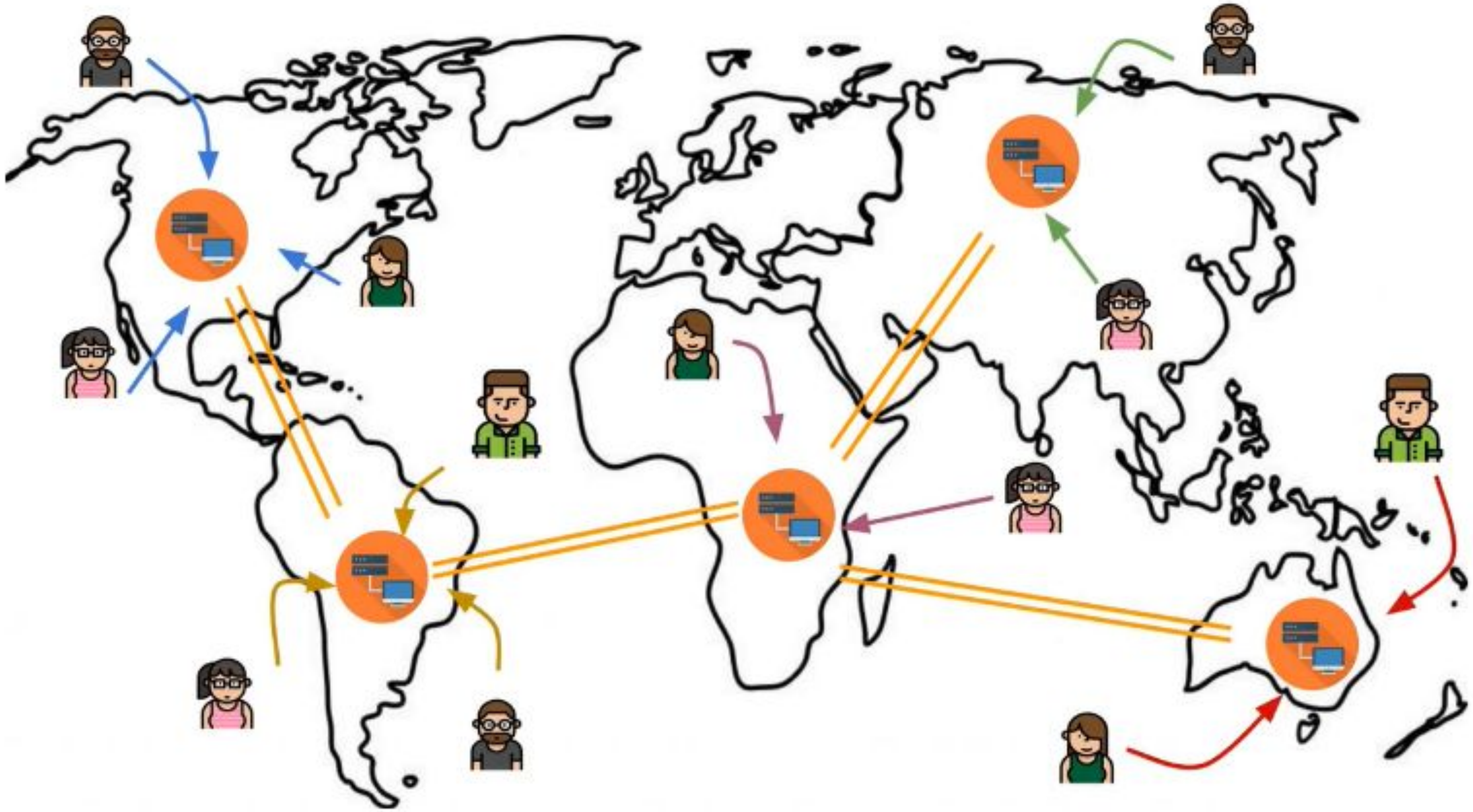
Vertical scaling involves increasing the capacity of a single server, such as adding more CPU, memory, or storage, to handle increased load.


Vertical Scaling
Horizontal Scaling
4 CPU, 4GB RAM, 2TB Storage
2 CPU, 2GB RAM, 1TB Storage
1 CPU, 1GB RAM, 500GB Storage
1 CPU, 1GB RAM, 500GB Storage
1 CPU, 1GB RAM, 500GB Storage
1 CPU, 1GB RAM, 500GB Storage
1 CPU, 1GB RAM, 500GB Storage
1 CPU, 1GB RAM, 500GB Storage
1 CPU, 1GB RAM, 500GB Storage
1 CPU, 1GB RAM, 500GB Storage

Horizontal scaling involves adding more servers or nodes to a system to distribute the load and increase capacity. Can easily isolate single system for repair and maintenance

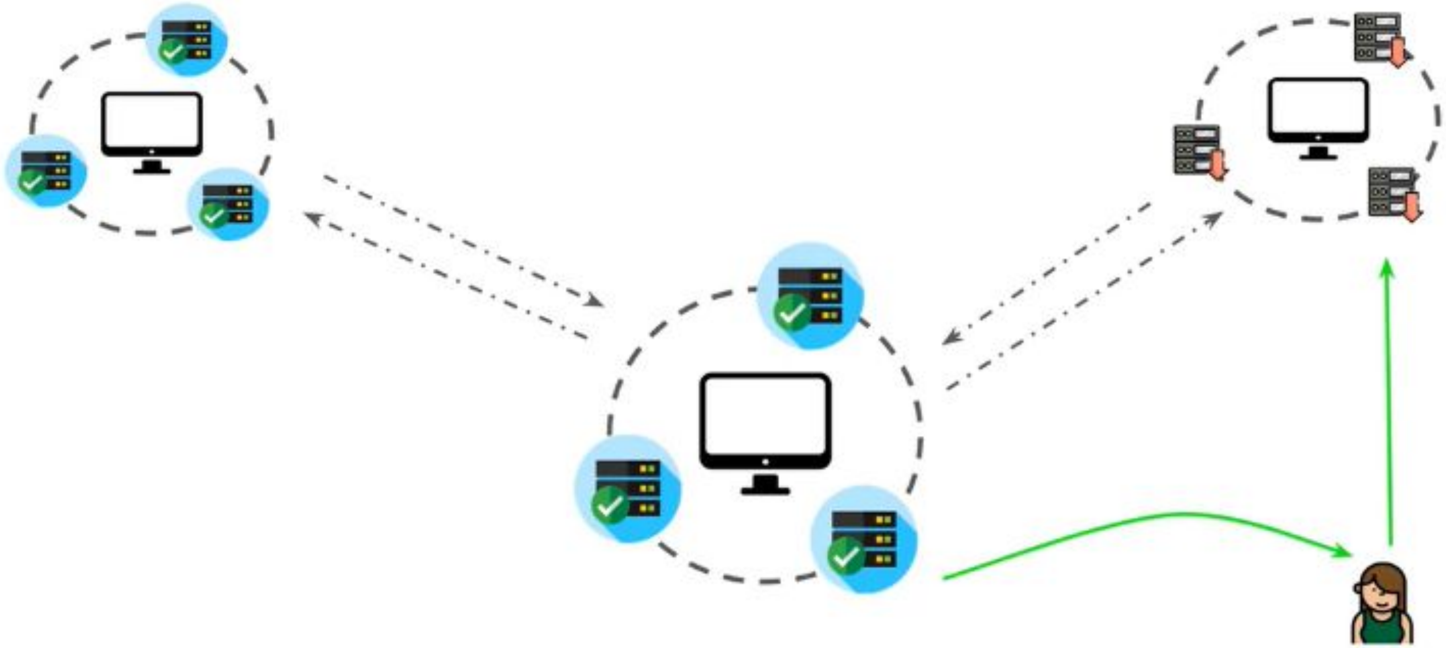
Distributed systems include multiple systems these can be placed in different physical locations but each of them off course interconnected with the others this allows them to handle lot more users than traditional systems and manages their queries concurrently without any system failure

Higher Concurrency



Since Distributed systems are clusters of multiple computers and each computer has a data replicated on other computer therefore even if a computer goes down due to any reason it will not make any difference and it will continue to do its work

Fault Tolerance



DFS (Distributed File System)

A **Distributed File System (DFS)** is a file system that is distributed on multiple file servers or multiple locations. It allows programs to access or store isolated files as they do with the local ones, allowing programmers to access files from any network or computer.

DFS (Distributed File System) is a technology that allows you to group shared folders located on different servers into one or more logically structured namespaces.

The main purpose of the Distributed File System (DFS) is to allows users of physically distributed systems to share their data and resources by using a Common File System.

A collection of workstations and mainframes connected by a Local Area Network (LAN) is a configuration on Distributed File System.

A DFS is executed as a part of the operating system. In DFS, a namespace is created and this process is transparent for the clients.

Components of DFS

Location Transparency

Redundancy

A **distributed file system (DFS)** is a file system that is distributed on various file servers and locations.

It permits programs to access and store isolated data in the same method as in the local files.

It also permits the user to access files from any system. It allows network users to share information and files in a regulated and permitted manner. Although, the servers have complete control over the data and provide users access control.

DFS's primary goal is to enable users of physically distributed systems to share resources and information through the **Common File System (CFS)**.

It is a file system that runs as a part of the operating systems. Its configuration is a set of workstations and mainframes that a LAN connects. The process of creating a namespace in DFS is transparent to the clients.

Hadoop Distributed File System

It has distributed file system known as HDFS and this HDFS splits files into blocks and sends them across various nodes in form of large clusters. Also in case of a node failure, the system operates and data transfer takes place between the nodes which are facilitated by HDFS.

Advantages of HDFS: It is inexpensive, immutable in nature, stores data reliably, ability to tolerate faults, scalable, block structured, can process a large amount of data simultaneously and many more. **Disadvantages of HDFS:** It’s the biggest disadvantage is that it is not fit for small quantities of data. Also, it has issues related to potential stability, restrictive and rough in nature. Hadoop also supports a wide range of software packages such as Apache Flumes, Apache Oozie, Apache HBase, Apache Sqoop, Apache Spark, Apache Storm, Apache Pig, Apache Hive, Apache Phoenix, Cloudera Impala.

Some common frameworks of Hadoop

Hive- It uses HiveQl for data structuring and for writing complicated MapReduce in HDFS.

Drill- It consists of user-defined functions and is used for data exploration.

Storm- It allows real-time processing and streaming of data.

Spark- It contains a Machine Learning Library(MLlib) for providing enhanced machine learning and is widely used for data processing. It also supports Java, Python, and Scala.

Pig- It has Pig Latin, a SQL-Like language and performs data transformation of unstructured data.

Tez- It reduces the complexities of Hive and Pig and helps in the running of their codes faster.

Hadoop framework is made up of the following modules:

Hadoop MapReduce- a MapReduce programming model for handling and processing large data.

Hadoop Distributed File System- distributed files in clusters among nodes.

Hadoop YARN- a platform which manages computing resources.

Hadoop Common- it contains packages and libraries which are used for other modules.

Advantages

It allows the users to access and store the data.

It helps to improve the access time, network efficiency, and availability of files.

It provides the transparency of data even if the server of disk files.

It permits the data to be shared remotely.

It helps to enhance the ability to change the amount of data and exchange data.

Disadvantages

In a DFS, the database connection is complicated.

In a DFS, database handling is also more complex than in a single-user system.

If all nodes try to transfer data simultaneously, there is a chance that overloading will happen.

There is a possibility that messages and data would be missed in the network while moving from one node to another.

Working of Distributed File System

There are two methods of DFS in which they might be implemented, and these are as follows:

Standalone DFS namespace

Domain-based DFS namespace

Standalone DFS namespace

It does not use Active Directory and only permits DFS roots that exist on the local system. A Standalone DFS may only be acquired on the systems that created it. It offers no-fault liberation and may not be linked to other DFS.

Domain-based DFS namespace

It stores the DFS configuration in Active Directory and creating namespace root at **domainname>dfsroot>** or **FQDN>dfsroot>**.

Hadoop has several key features that make it well-suited for big data processing:

Distributed Storage: Hadoop stores large data sets across multiple machines, allowing for the storage and processing of extremely large amounts of data.

Scalability: Hadoop can scale from a single server to thousands of machines, making it easy to add more capacity as needed.

Fault-Tolerance: Hadoop is designed to be highly fault-tolerant, meaning it can continue to operate even in the presence of hardware failures.

Data locality: Hadoop provides data locality feature, where the data is stored on the same node where it will be processed, this feature helps to reduce the network traffic and improve the performance

High Availability: Hadoop provides High Availability feature, which helps to make sure that the data is always available and is not lost.

Flexible Data Processing: Hadoop’s MapReduce programming model allows for the processing of data in a distributed fashion, making it easy to implement a wide variety of data processing tasks.

Data Integrity: Hadoop provides built-in checksum feature, which helps to ensure that the data stored is consistent and correct.

Data Replication: Hadoop provides data replication feature, which helps to replicate the data across the cluster for fault tolerance.

Data Compression: Hadoop provides built-in data compression feature, which helps to reduce the storage space and improve the performance.

YARN: A resource management platform that allows multiple data processing engines like real-time streaming, batch processing, and interactive SQL, to run and process data stored in HDFS.

Disadvantages

Not very effective for small data.

Hard cluster management.

Has stability issues.

Security concerns.

Complexity: Hadoop can be complex to set up and maintain, especially for organizations without a dedicated team of experts.

Latency: Hadoop is not well-suited for low-latency workloads and may not be the best choice for real-time data processing.

Limited Support for Real-time Processing: Hadoop’s batch-oriented nature makes it less suited for real-time streaming or interactive data processing use cases.

Limited Support for Structured Data: Hadoop is designed to work with unstructured and semi-structured data, it is not well-suited for structured data processing

Data Security: Hadoop does not provide built-in security features such as data encryption or user authentication, which can make it difficult to secure sensitive data.

Limited Support for Ad-hoc Queries: Hadoop’s MapReduce programming model is not well-suited for ad-hoc queries, making it difficult to perform exploratory data analysis.

Limited Support for Graph and Machine Learning: Hadoop’s core component HDFS and MapReduce are not well-suited for graph and machine learning workloads, specialized components like Apache Graph and Mahout are available but have some limitations.

Cost: Hadoop can be expensive to set up and maintain, especially for organizations with large amounts of data.

Data Loss: In the event of a hardware failure, the data stored in a single node may be lost permanently.

Data Governance: Data Governance is a critical aspect of data management, Hadoop does not provide a built-in feature to manage data lineage, data quality, data cataloging, data lineage, and data audit.

References

https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html

https://www.databricks.com/glossary/hadoop-distributed-file-system-hdfs

https://static.googleusercontent.com/media/research.google.com/en//archive/gfs-sosp2003.pdf

https://www.geeksforgeeks.org/hadoop-history-or-evolution/

https://data-flair.training/blogs/hadoop-history/