

Introduction to the World of Big Data

1. What Is Big Data?

Big Data means:

- Very **large amounts of data**
- Data that is **complex**
- Data that comes **very fast**
- Data that can be **structured, semi-structured, or unstructured**

Example:

Every day the world generates **2.5 quintillion bytes** of data from WhatsApp, Facebook, YouTube, sensors, and more.

2. Why Traditional Databases Failed

Traditional databases (RDBMS) failed because they:

- Cannot handle **huge data volumes** (terabytes → petabytes)
- Become **very expensive** when scaled
- Cannot store **unstructured** or **semi-structured** data
- Cannot handle **high-speed streaming data**

Big Data technologies were created to solve these problems.

3. The 3 Vs of Big Data

Big Data has 3 special features:

1) Volume

- Refers to the **size of the data** generated and stored.

- Big Data reaches **terabytes, petabytes, and even zettabytes**.
- Growth is driven by social media, online transactions, sensors, GPS, etc.
- **Example:**
Facebook generates **about 500 terabytes of data per day**.

2) Velocity

- Refers to how **fast data is produced and needs to be processed**.
- Every 60 seconds:
 - 3.3 million Facebook posts
 - 450,000 tweets
 - 400 hours of YouTube videos

3) Variety

- Different types of data:
 - **Structured:** tables (E.g. employee database, bank records)
 - **Semi-structured:** JSON, XML, HTML (Has some structure but does not fit traditional tables.)
 - **Unstructured:** photos, videos, emails, social media posts, sensor logs



Structured Data



Unstructured Data



Semi-Structured Data

4. Sources of Big Data

Big Data comes from many places:

1) Sensors

- Accelerometers in phones
- Proximity sensors
- Vehicle sensors

- Medical devices

2) Healthcare

- Electronic Health Records (EHRs)
- Patient portals
- Clinical repositories

3) Machines

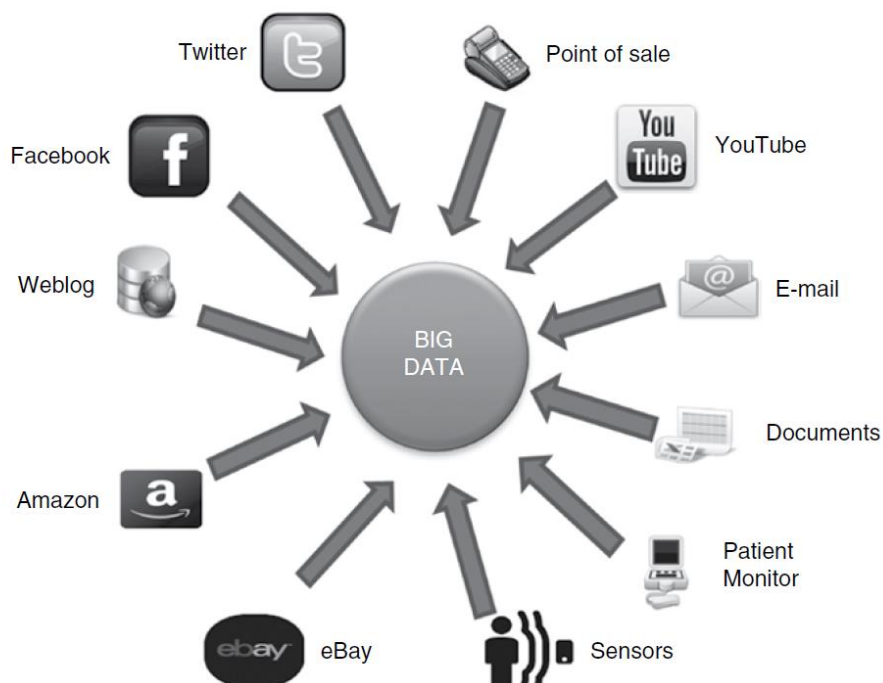
- Airplane black boxes
- Satellites

4) Websites

- Click-stream data (what users click)

5) Organizations

- Emails
- Documents
- Internal transactions



5. Types of Data

There are two broad categories :

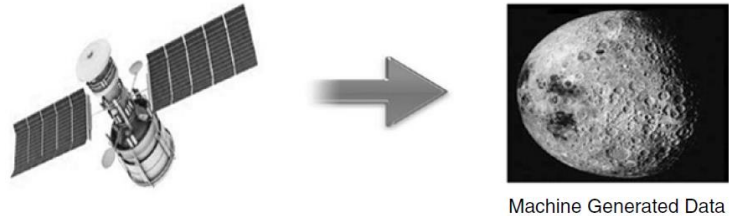
A) Human-Generated Data

- Social media posts
- Emails
- Documents



B) Machine-Generated Data

- Sensors
- Weather systems
- Satellites



Data formats:

1. **Structured** → tables, rows, columns
2. **Unstructured** → images, videos, PDFs
3. **Semi-structured** → XML, JSON

6. Big Data Infrastructure

Big Data uses special technologies:

Hadoop

- An open-source framework written in Java.
- Stores and processes huge amounts of structured, semi-structured, and unstructured data.
- Uses a cluster of many inexpensive machines (commodity hardware).
- Runs on **cheap hardware** (commodity servers)
- Highly scalable: performance increases by adding more nodes.

- Files are written once and read many times.
- Designed to be cost-effective and fault-tolerant.

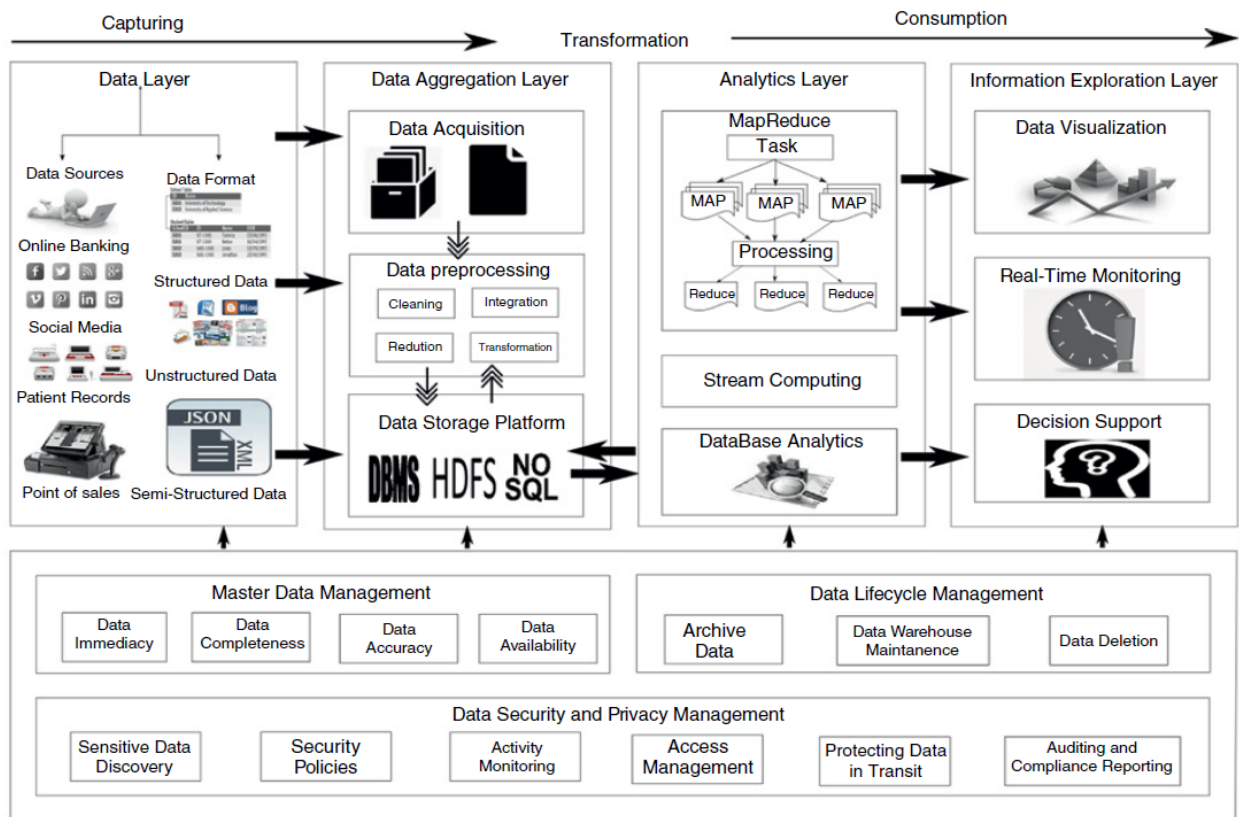
HDFS (Hadoop Distributed File System)

- A distributed file system for storing massive datasets across many machines.
- Works well with low-cost hardware.
- Stores data in a **write-once, read-many** formats.
- Handles data from multiple sources.
- Very fault-tolerant: data is replicated across several machines.

MapReduce

- A batch-processing programming model inside Hadoop.
- Uses a **divide-and-conquer** approach:
 - **Map** phase → splits the task into smaller parts
 - **Reduce** phase → combines the results
- Processes data **in parallel** across many machines.
- Highly scalable and fault-tolerant.
- Faster than traditional systems because processing happens **where the data is stored**, not moved elsewhere.

7. Big Data Life Cycle



The Big Data journey has several steps:

1. Data Generation

- Data is created continuously from many sources:
 - Social media
 - Online banking
 - Sensors / IoT
 - Smartphones
 - Transactions
 - Emails, images, videos
- This is where the life cycle starts.

2. Data Aggregation (Collection + Transfer)

- Raw data from different sources is:
 - **Collected**
 - **Transferred** to storage systems (HDFS, NoSQL, DBMS)
- Goals:
 - Bring all data into one place
 - Handle data arriving at high velocity
 - Compress or organize data if needed

3. Data Preprocessing

Raw data is usually **dirty, incomplete, inconsistent**, so it must be cleaned and prepared.

The four main stages:

a) Data Cleaning

- Fix missing values
- Correct errors and inconsistencies
- Remove duplicates
- Reduce noise

b) Data Integration

- Combine data from many sources into one unified dataset
- Example sources:
 - Smartphones
 - Online transactions
 - Organization files
 - Social media posts
 - Sensors, satellite images

c) Data Reduction

- Reduce data size while keeping important information
- Techniques:
 - Compression
 - Dimensionality reduction
 - Attribute selection
 - Sampling

d) Data Transformation

- Convert data into a useful, consistent format
- Techniques:
 - Smoothing (remove noise)
 - Aggregation (combine values)
 - Generalization (higher-level categories)
 - Discretization (convert numeric → categories)

4. Analytics

Data is analyzed to extract insights and reveal patterns.

Three main types:

a) Descriptive analytics

- What happened?
- Summaries, statistics, trends

b) Predictive analytics

- What will happen?
- Machine learning, forecasting

c) Prescriptive analytics

- What should we do?

- Recommendations, optimization

Analytics uses tools such as MapReduce, YARN, NoSQL engines, and streaming systems.

5. Visualization

- Results are presented visually so humans can understand them easily.
- Tools: dashboards, Tableau, Power BI, charts.
- Common charts:
 - Line, bar, scatter, bubble, pie charts

6. Decision Making

- The final stage.
- Organizations use the insights to make:
 - Business decisions
 - Customer analysis
 - Operational improvements
 - Strategic planning

8. Big Data Applications

Big Data is used in almost every industry to improve efficiency, reduce costs, and make smarter decisions.

1. Healthcare

Big Data helps improve patient care and medical research.

Uses:

- **Early disease detection**
(Analyzing patient history, lab results, genomics to identify diseases early.)
- **Remote patient monitoring**
(Wearable devices send real-time health data to doctors.)

- **Medical image analysis**
(MRI, CT scans analyzed using AI.)
- **Personalized treatment**
(Treatment plans based on patient data.)
- **Public health monitoring**
(Tracking epidemics using large data sets.)

2. Banking & Financial Services

Big Data increases security and improves customer service.

Uses:

- **Fraud detection**
(Analyzing transaction patterns to catch fraudulent credit card activity.)
- **Credit risk scoring**
(Predicting if a customer is likely to repay loans.)
- **Wealth management**
(Personalized investment recommendations.)
- **Customer retention analysis**
(Identifying customers who may switch banks.)
- **Real-time analytics**
(Detecting fraud within milliseconds.)

3. Telecom

Big Data improves network quality and customer experience.

Uses:

- **Network optimization**
(Analyzing traffic patterns to prevent slowdowns and outages.)
- **Personalized data plans**
(Offering plans based on customer usage.)

- **Reducing churn**
(Identifying customers likely to leave the company.)
- **Call center analytics**
(Detecting common issues, improving resolution time.)
- **Predicting service demand**
(Using historical usage data.)

4. Marketing & Retail

Big Data helps companies understand and predict customer behavior.

Uses:

- **Customer behavior analysis**
(Purchase history, website clicks, social media activity.)
- **Recommendation systems**
(Products shown on Amazon, Netflix, YouTube.)
- **Targeted advertising**
(Ads tailored to user interests.)
- **Sales forecasting**
(Predicting demand for products.)

5. Finance (General Business Analytics)

Beyond traditional banking, Big Data supports financial operations.

Uses:

- **Real-time fraud detection**
(Streaming data analysis to detect suspicious activity instantly.)
- **Customer retention analysis**
(Monitoring customer actions across platforms.)
- **Risk management**
(Predicting market risks, interest rate changes.)

- **Investment insights**
(Analyzing market trends and portfolios.)

6. Additional Application Areas

Web Analysis

- Uses search engine data and social media activity to show relevant ads.

Call Centers

- Identifies common problems and improves staff performance.

Agriculture

- Uses sensors to monitor crops and optimize growth.

Smartphones

- Facial recognition
- Security features
- Personalized assistant tools

Lecture Summary

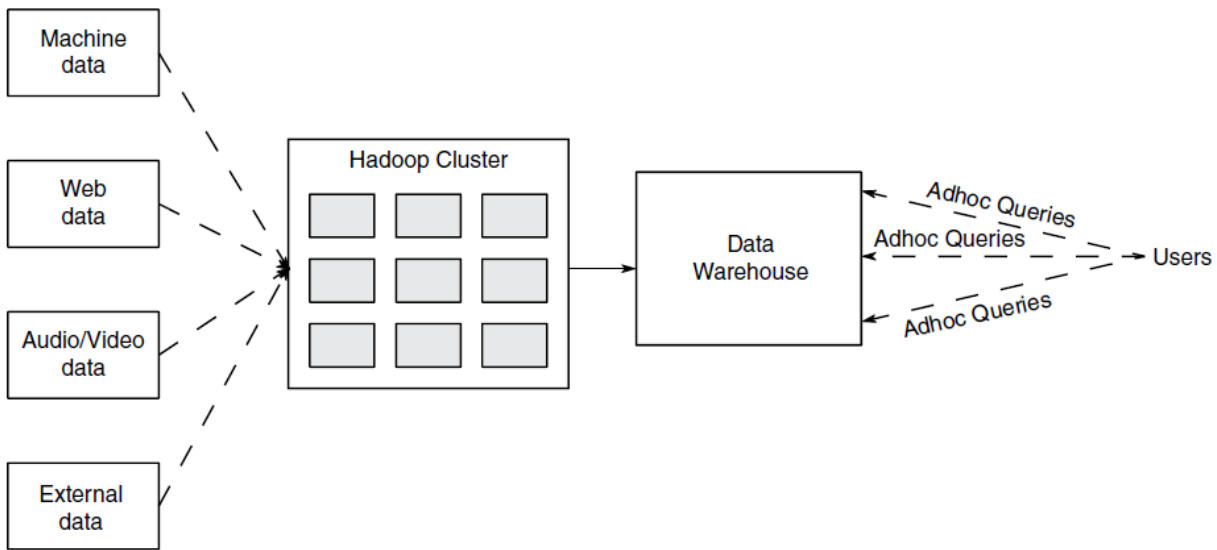
In this lecture you learned:

- What Big Data is
- Why traditional systems failed
- The 3 Vs (Volume, Velocity, Variety)
- Where Big Data comes from
- Types of data
- What Hadoop, HDFS, and MapReduce do
- The Big Data life cycle
- Real applications

Lecture 2

Big Data Storage & Architecture

Big Data needs **special storage methods** because normal computers cannot store or process huge amounts of data efficiently:



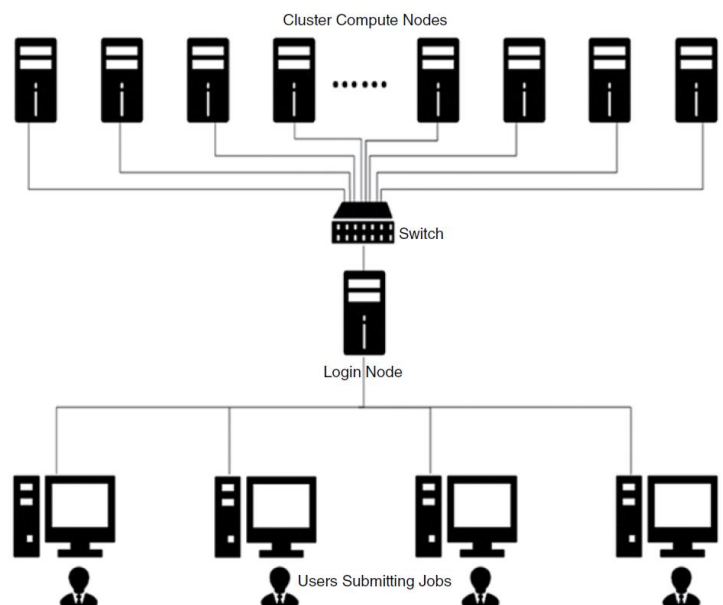
In this lecture, you will learn how Big Data storage works—**clusters, distributed systems, sharding, replication, and HDFS**.

1. What Is Cluster Computing?

Cluster computing means:

- Many **separate computers** (nodes)
- Connected together through a **local area network (LAN)**.
- Working as **one powerful system**

Clusters provide:



- **High availability** (system keeps working even if one node fails)
- **Fault tolerance:**
Redundant nodes ensure continuous service.
- **Scalability:**
You can add more nodes whenever needed.
- **High performance at low cost:**
Uses **commodity hardware** (cheap machines).

2. Types of Clusters

Clusters are designed differently depending on what the organization needs.

A. High-Availability Cluster

- Designed to avoid downtime
- If one node fails → another takes over automatically (failover)
- Suitable for systems that must run 24/7:
 - **Banking**
 - **Billing systems**
 - **E-commerce**
- Has redundant nodes for safety.
- Ensures data integrity even during failures.

B. Load-Balancing Cluster

- Distributes workload across many nodes
- Prevents overload on a single computer
- Improves performance (Improves response time.)
- Great for web servers, cloud services, online applications.

Common load balancing algorithms:

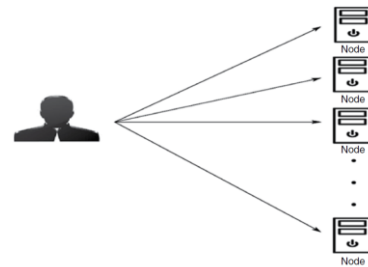
- **Round Robin:** Each request goes to the next server in a circle.
- **Weight-Based:** Nodes with higher capacity get more requests.
- **Random:** Requests sent to random nodes.
- **Server Affinity:** Same client is routed to the same server.

3. Cluster Structures

Clusters can be connected internally in **two main architectural ways**:

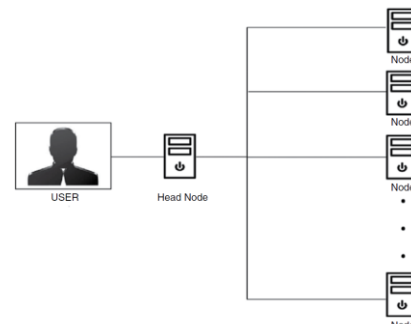
A. Symmetric Cluster

- All nodes work equally
- Every node can run applications.
- No special “master” node.
- Easy to scale: add or remove nodes anytime.



B. Asymmetric Cluster

- One **head node** controls everything.
- Other nodes mainly perform processing tasks.
- Users connect only to the head node.



4. Data Distribution Models

When Big Data grows too large, we must distribute data across many machines.

There are **three main techniques**:

1. **Sharding (Data Partitioning)**
2. **Replication (Copying data)**
3. **Sharding + Replication together (Combined approach)**

Let's understand each one.

5. Sharding (Data Partitioning)

Sharding means:

- Splitting large data into **small parts** called *shards*
- Storing each shard on a **different node**

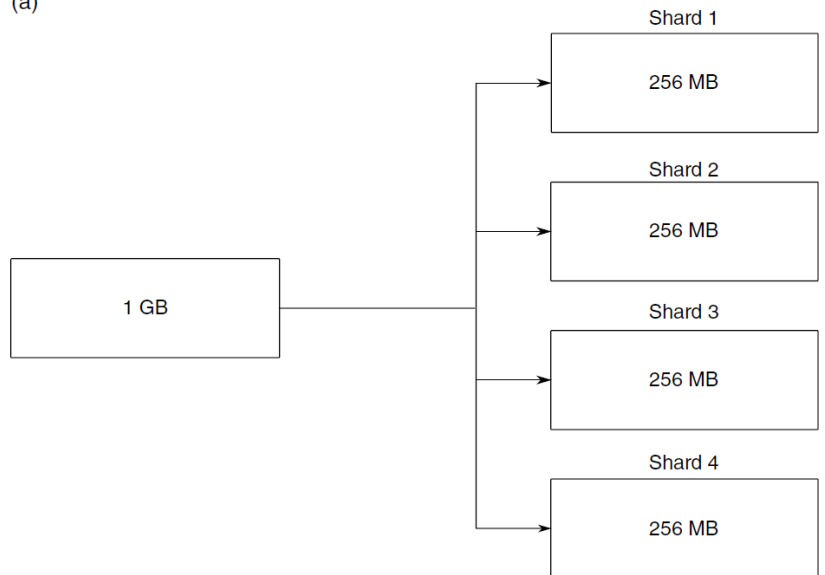
Example:

A 1 GB file → split into 4×256 MB shards → each stored on a different server.

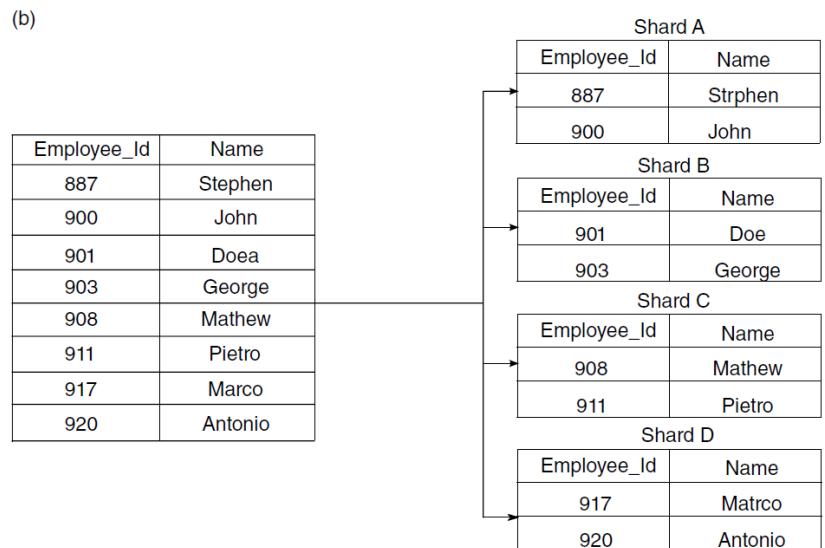
Benefits of Sharding:

- Reduces load on each node
- Improves query performance
- Greatly increases scalability
- Limits failure impact (If one node fails, **only its shard is lost**)

(a)



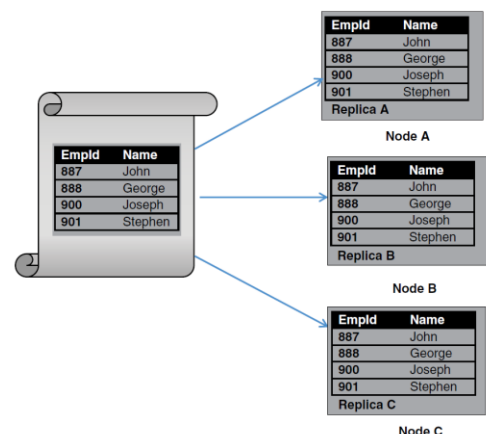
(b)



6. Replication (Copying Data)

Replication means:

- Making **multiple copies** of the same data
- Storing copies on different nodes
- Prevents data loss

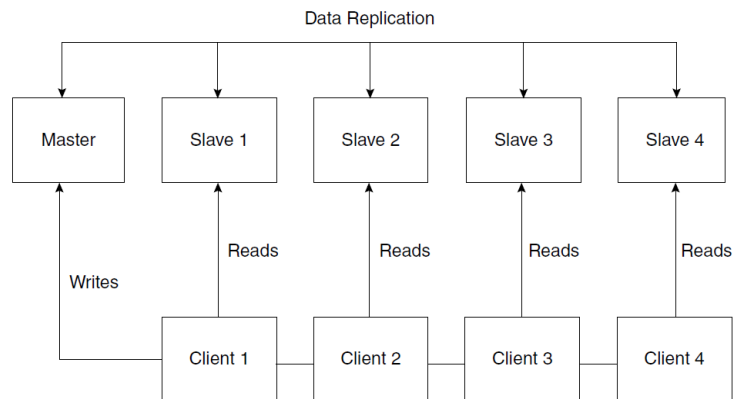


If Node A fails → Node B still has the same data.

Two replication models exist:

A. Master-Slave Model

- One node = Master
- Other nodes = Slaves
- Master node handles all **writes**
- Slaves handle **reads**

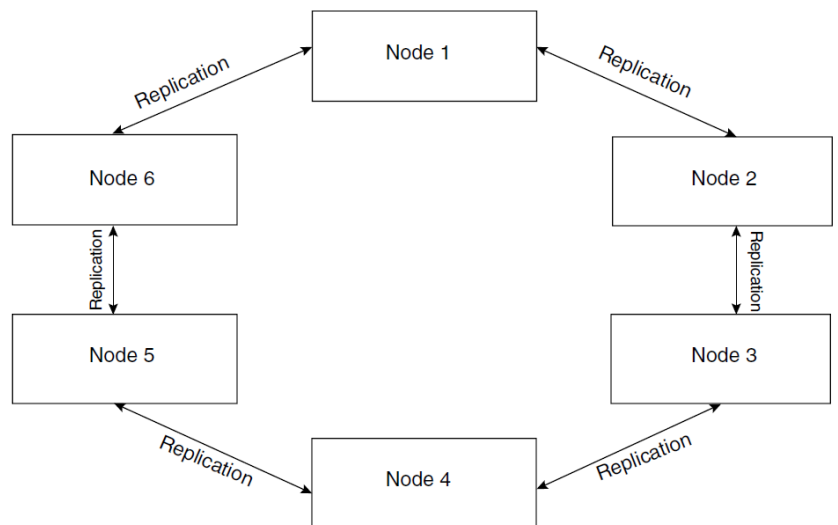


Disadvantages

- If master fails → no writes possible → **single point of failure**
- Scalability limited for writes

B. Peer-to-Peer Model

- **No master node**
- All nodes are equal
- Each node can handle read + write (Better for reliability.)
- No single point of failure
- Good for distributed global systems



Advantages

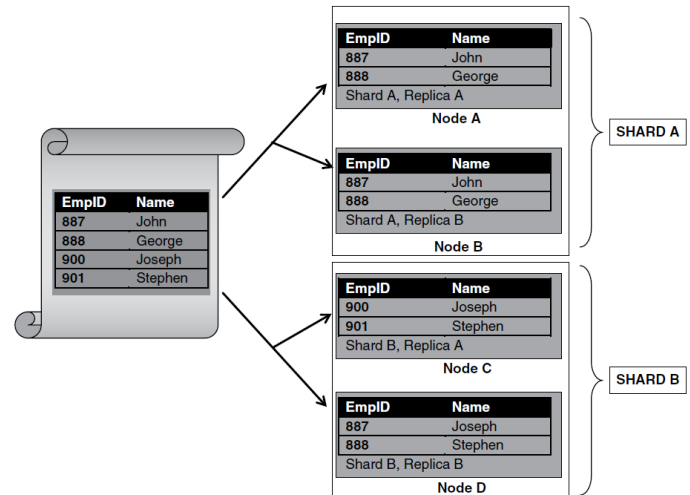
- More reliable
- Better scalability
- More balanced resource usage

7. Combining Sharding & Replication

This combined model is used in **real-world Big Data systems** like Cassandra, MongoDB, HDFS.

How it works

1. **First shard** the dataset;
2. **Then replicate each shard** to multiple nodes.



Example

- **Shard A** → stored on **Node 1** and **Node 2**
- **Shard B** → stored on **Node 3** and **Node 4**

Benefits

- **High scalability** (due to sharding)
- **High reliability** (due to replication)
- Ensures both efficiency and fault tolerance
- Reduces risk of data loss

8. Distributed File Systems (DFS)

A Distributed File System:

- Stores files across many machines (different cluster nodes)
- Appears to users like **one single system** → Users don't need to know where the file is physically stored.
- Supports multiple users reading/writing at the same time
- Uses replication for reliability → Copies of data (replicas) are stored on several nodes.

- Improves fault tolerance → If one node fails, data is still available from other nodes.

Most famous DFS: **Hadoop Distributed File System (HDFS)**.

9. Why HDFS Is Needed

Traditional storage cannot handle:

- Multi-terabyte data
- High-speed data
- Low-cost scaling

HDFS solves this by:

- Splitting files into large blocks (64 MB default)
- Storing blocks across DataNodes
- Replicating blocks (default: 3 copies)
- Running on cheap commodity hardware

10. Scaling Big Data Storage

Two ways to scale:

A. Scaling Up (Vertical)

Increase the power of one single server by adding:

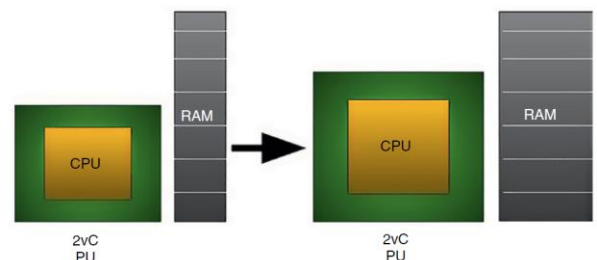
- more RAM, CPU cores, Faster hardware

Advantages:

- Simple to upgrade
- No need to change system architecture

Disadvantages:

- **Very expensive**



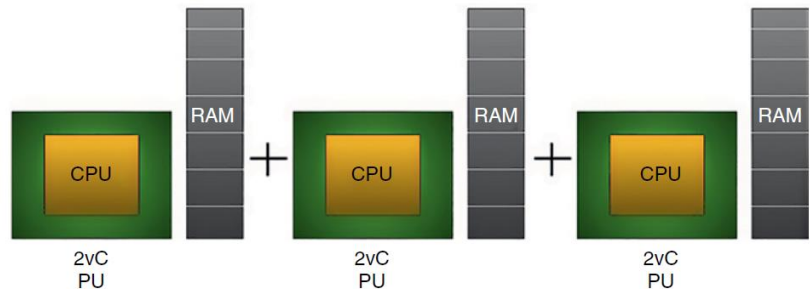
- **Has a hard limit** (you cannot upgrade forever)
- If the single server fails → everything stops
- Not suitable for Big Data

B. Scaling Out (Horizontal)

Add **more servers (nodes)** instead of upgrading a single one.

Advantages:

- Easy to expand anytime
- Much lower cost
- No hardware limits
- Highly fault-tolerant
- Perfect for Big Data systems



Used by:

- **Hadoop / HDFS**
- **NoSQL databases** like MongoDB, Cassandra, HBase

11. Key Big Data Storage Concepts Summary

Concept	Meaning	Why Important
Cluster	Many machines work as one	More power, more reliability
Sharding	Split data into pieces	Scalability
Replication	Duplicate data	Fault tolerance
DFS	Files stored across many nodes	Large, reliable storage
HDFS	Hadoop's storage system	Handles big data easily
Scaling Out	Add machines	Cheapest + most scalable

Lecture 2 Summary

You learned:

- How cluster computing works
- Types of clusters
- Sharding and replication
- Why distributed file systems are needed
- How HDFS stores and protects data
- Difference between scaling up and scaling out