

Ministry of Higher Education and Scientific Research
University of Mila
Faculty of Mathematics and Computer Science
Department of Computer Science
Master 2 I2A – Big Data
2026/2027

Chapter 1

Introduction to Big Data

Presented by: Dr. Brahim Benabderrahmane

Chapter Learning Objectives

- 01 Define** Big Data and explain why traditional relational database management systems fail when scaling to handle modern data requirements.
- 02 Categorize** and analyze diverse datasets into structured, semi-structured, and unstructured formats across modern digital data sources.
- 03 Evaluate** real-world data processing challenges and system characteristics by applying the 5 Vs framework (Volume, Velocity, Variety, Veracity, and Value).
- 04 Assess** storage memory hierarchies and apply the core principle of data locality within distributed computing platforms like Hadoop and Spark.



Table of Contents

- 01** **Motivations**
- 02** **Defining Big Data**
- 03** **Data Sources**
- 04** **Challenges of Big Data**
- 05** **The 5 Vs of Big Data**
- 06** **Memory Hierarchy & Distributed Storage**
- 07** **Disciplines of Big Data**



01

Motivations

Explosion of Digital Data



Data Generation

Massive growth in data generation across all sectors



IoT & Sensors

Smart devices, GPS signals, health trackers



Web

Billions of daily Google searches



Transactions

E-commerce, banking, digital payments



Social media

500M+ tweets per day, billions of Facebook and Instagram posts



Every second

Petabytes of new data created

Real-World Example: Data Growth at Scale



Google Search

- ~99,000 queries *every second*
- 8.5 billion searches per day



YouTube

- 500 hours of video uploaded every minute
- Billions of daily views



Instagram

- ~95 million photos & videos shared daily
- Over 2 billion active users



TikTok

- 34 million videos uploaded every day
- 1 billion monthly active users

Why Traditional Tools Are Not Enough



Scalability limits

RDBMS designed for single-server or small clusters

Rigid schema

Difficult to handle semi-structured or unstructured data (e.g., videos, social media text)

Performance bottlenecks

Can't efficiently manage petabytes of data

Real-time processing

RDBMS are optimized for transactions, not high-velocity streams

Cost & complexity

Scaling vertically (bigger servers) is expensive and limited



02

Definition of Big Data

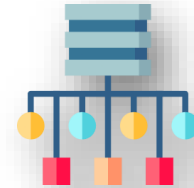
Definition of Big Data



Big Data refers to **extremely large** and **complex** datasets that cannot be efficiently **stored**, **processed**, or **analyzed** using traditional database systems and tools.

It also encompasses the **hardware** and **software** solutions developed to manage such data at scale, as well as the scientific and industrial field dedicated to studying methods for handling **massive**, **diverse**, and **fast-growing** information.

Big Data is not only about the **size** of the data, but also about its **variety**, **velocity**, and the ability to extract valuable insights from it.



Large, Complex Datasets Beyond Traditional DBMS

Size

Too large for single-server databases

Diversity

Structured, semi-structured, and unstructured data

Limits

RDBMS cannot scale horizontally

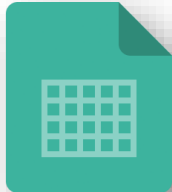
Solutions

Hadoop, Spark, and NoSQL for massive data

Types of Data in Big Data

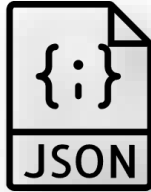
Structured:

Organized in rows/columns
(e.g., relational databases,
financial records)



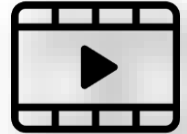
Semi-Structured:

Flexible format with
tags/markers
(e.g., XML, JSON, log files)



Unstructured :

No predefined model
(e.g., videos, images, social
media posts, emails)



Hardware & Software Solutions for Big Data

Solutions

```
graph LR; Solutions --> Hardware; Solutions --> Software; Hardware --> Clusters; Hardware --> Storage; Hardware --> Parallelism; Hardware --> Scaling; Software --> Frameworks; Software --> Databases; Software --> Tools; Software --> Analytics;
```

Hardware

Clusters → Many interconnected machines

Storage → Distributed file systems (HDFS)

Parallelism → Multi-core CPUs & GPUs

Scaling → Horizontal expansion

Software

Frameworks → Hadoop, Spark for distributed processing

Databases → NoSQL (MongoDB, Cassandra, HBase)

Tools → Hive, Kafka for data management & streaming

Analytics → ML libraries and visualization platforms

Big Data as a Field of Study

Practical solutions for
storage, processing, and
analytics

Industry

Combines computer
science, statistics, AI,
and domain knowledge

Interdisciplinary

New algorithms &
architectures for large-
scale data

Research

Drives progress in
healthcare, finance, IoT,
e-commerce, and more

Innovation



03

Data Sources

Social Media as a Source of Big Data (2025)

Global data size

In 2025, **181 zettabytes** generated worldwide:

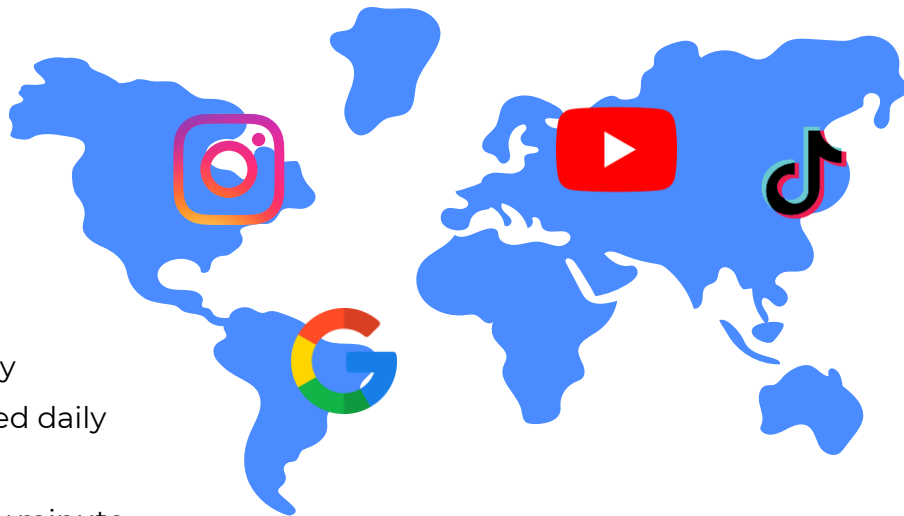
181,000 EB (exabytes)

181,000,000 PB (petabytes)

181,000,000,000 TB (terabytes)

181,000,000,000,000 GB (gigabytes)

- **Users** → ~5.24B people on social media
- **Facebook** → ~3.07B monthly active users
- **Instagram** → ~2B users, ~95M posts shared daily
- **TikTok** → ~1.8B users, ~34M new videos uploaded daily
- **X (Twitter)** → ~500M tweets per day
- **YouTube** → >500 hours of video uploaded every minute
- **Streaming platforms** → Netflix, Spotify = petabytes of viewing/listening data daily



E-Commerce & Transactions as Big Data Sources (2025)

Online shopping	Global e-commerce sales expected to exceed \$6.3 trillion in 2025
Amazon	Millions of transactions daily, generating petabytes of purchase & clickstream data
Digital payments	Billions of credit card and mobile wallet transactions (PayPal, Apple Pay..) every day
Banking	Real-time financial records and fraud detection systems rely on massive datasets
Impact	Transactional data is structured, but huge volume and velocity requires Big Data frameworks

GPS & IoT Sensors: Real-Time Big Data



Scientific Data as a Source of Big Data



Climate



Genomics



Astronomy

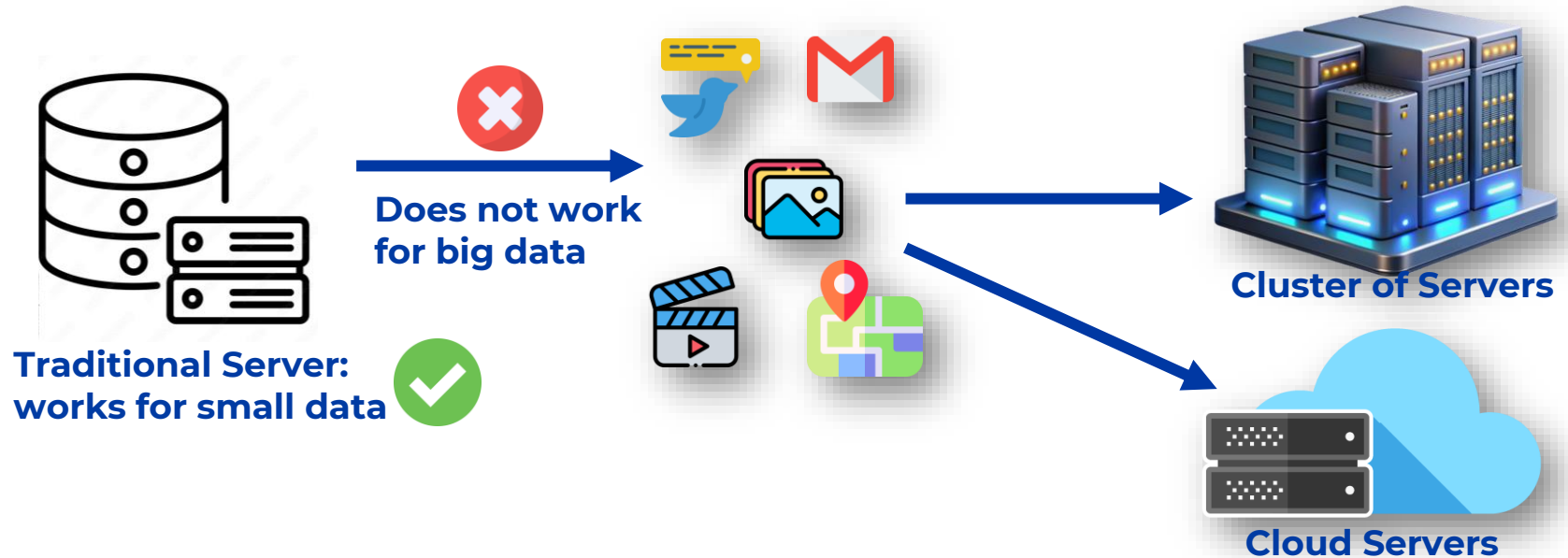
- **Climate science** → Petabytes from satellites, weather stations, and climate models
- **Genomics** → Human genome sequencing generates ~200 GB per genome; projects involve millions of samples
- **Astronomy** → Telescopes like the Square Kilometre Array expected to produce hundreds of petabytes annually
- **Other fields** → Particle physics (CERN LHC), oceanography, space missions
- **Impact** → Scientific research creates some of the largest datasets on Earth



04

Challenges of Big Data

From Traditional Servers to Big Data Storage

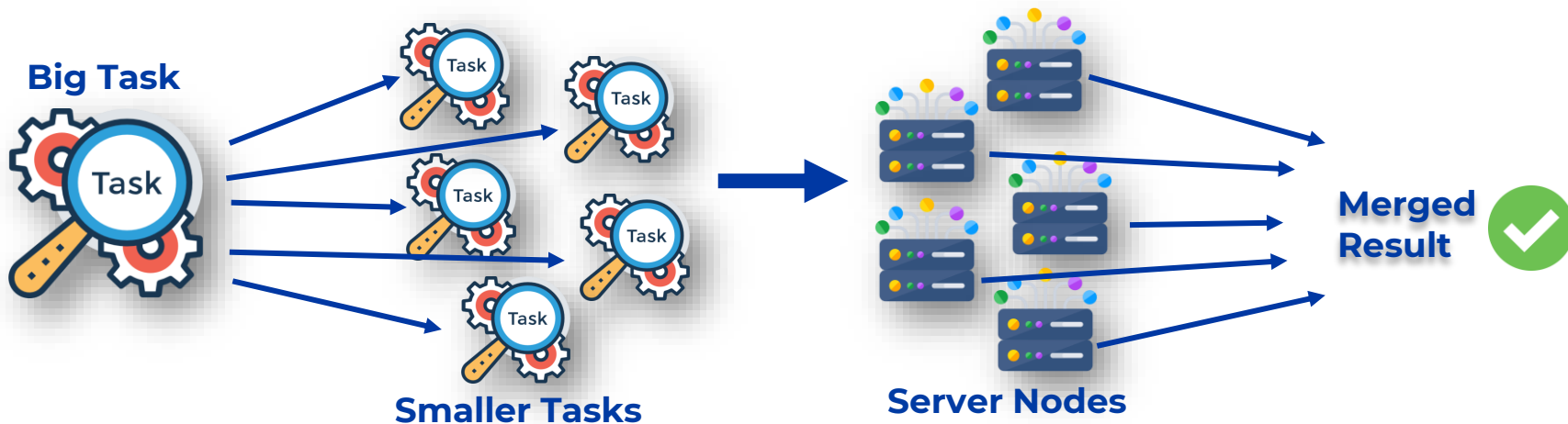


- **Challenge 1: Capacity** → Hard drives can't keep up with petabytes/zettabytes
- **Challenge 2: Reliability** → Hardware failures risk massive data loss
- **Challenge 3: Scalability** → Expanding vertically (bigger servers) is costly and limited

Distributed Systems & Parallel Computation

Before (Traditional): One big server trying to process everything → ⚠ bottleneck

After (Distributed): Many small servers working together → ✅ faster & scalable

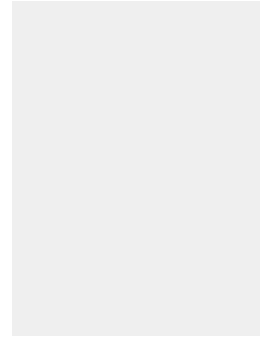


- **Scalable** → Add more machines easily
- **Reliable** → System survives node failures
- **Fast** → Parallel execution reduces time

Why We Need New Database Paradigms

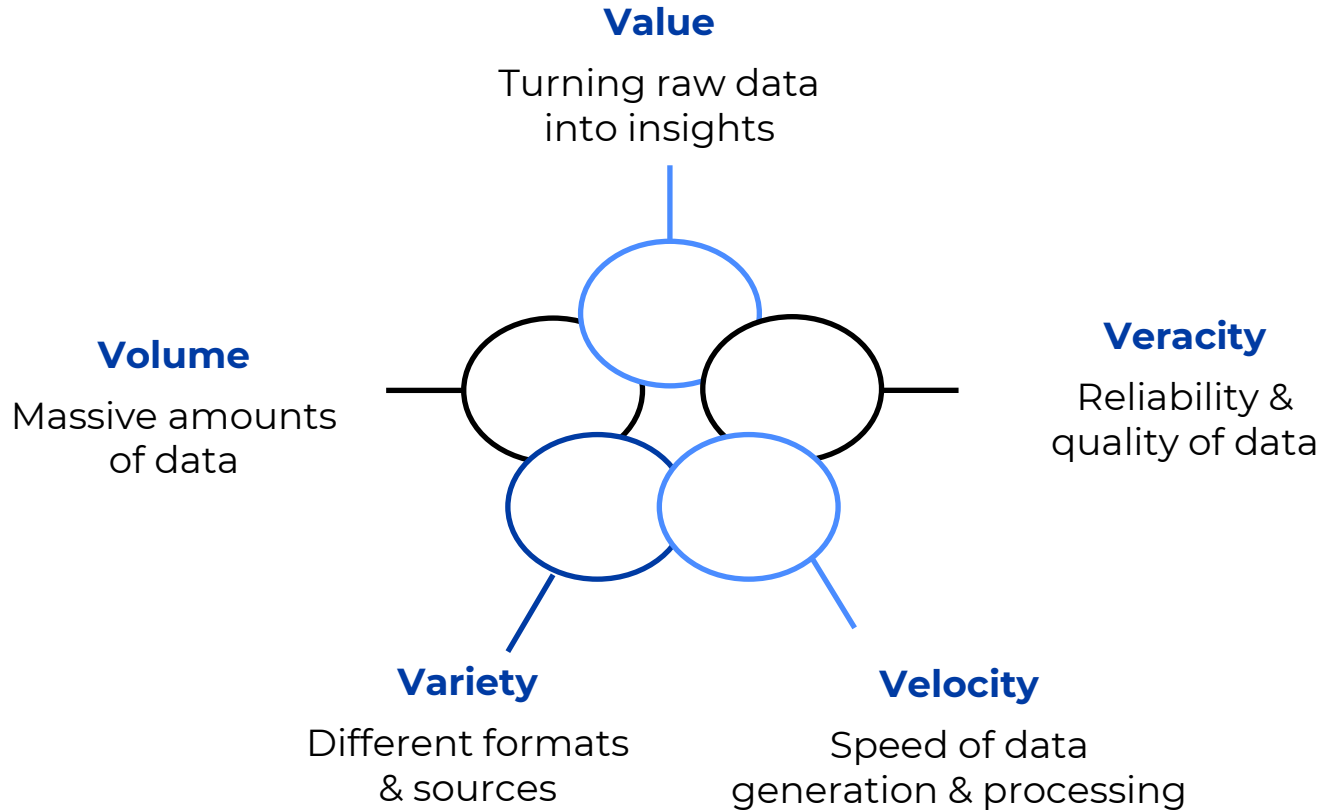
Traditional RDBMS	New Paradigms
Works well for structured data (tables, rows, columns)	NoSQL databases (document, key-value, column, graph) → flexible schemas
Strong consistency (ACID transactions)	Distributed databases → scale horizontally across many servers
Struggles with scale, flexibility, and unstructured data	Designed for massive, diverse, real-time data
Example: relational DB trying to handle billions of JSON logs or videos	Examples: MongoDB, Cassandra, HBase, Neo4j

05



The 5 Vs of Big Data

The 5 Vs of Big Data



Big Data Characteristic – Volume

Scale:

Data measured in petabytes to zettabytes

Examples :

- Netflix stores petabytes of viewing history.
- YouTube gets 500h of video/min

Challenge:

Traditional systems can't handle this magnitude

Solution:

Distributed storage & scalable architectures (HDFS, data lakes)



Big Data Characteristic – Variety

Structured:

Rows & columns (financial records, inventory)

Semi-Structured:

Flexible format (XML, JSON, logs)

Unstructured:

No fixed schema (images, videos, social media posts, emails)

Challenge:

Integrating multiple formats consistently

Solution:

NoSQL databases & data lakes for heterogeneous data



Big Data Characteristic – Velocity

Speed:

Data generated and updated in real time

Examples :

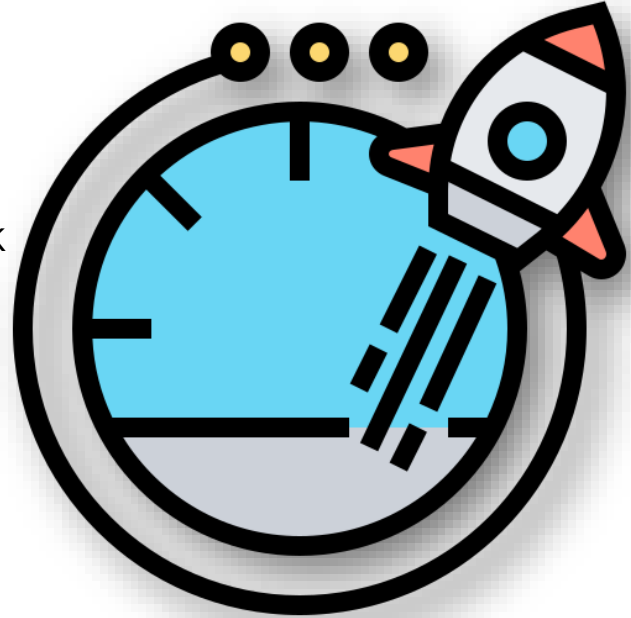
- 500M tweets/day.
- IoT sensors streaming every second, stock market transactions

Challenge:

Processing continuous data flows with low latency

Solution:

Real-time frameworks (Kafka, Spark Streaming, Flink)



Big Data Characteristic – Veracity

Uncertainty:

Data may be incomplete, inconsistent, or noisy

Examples :

- Fake news on social media.
- sensor errors, duplicated records.

Challenge:

Low-quality data leads to unreliable insights.

Solution:

Data cleaning, validation, and governance practices.



Big Data Characteristic – Value

Meaning:

The ultimate goal is extracting insights, not just storing data.

Examples :

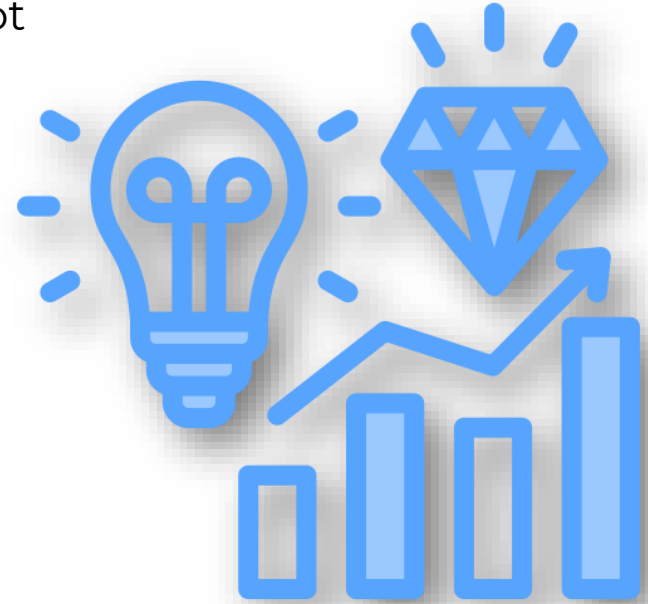
- Netflix recommendations from viewing history.
- Fraud detection in banking.
- Predictive healthcare analytics.

Challenge:

Turning raw, messy data into actionable knowledge.

Solution:

Advanced analytics, machine learning, and visualization.

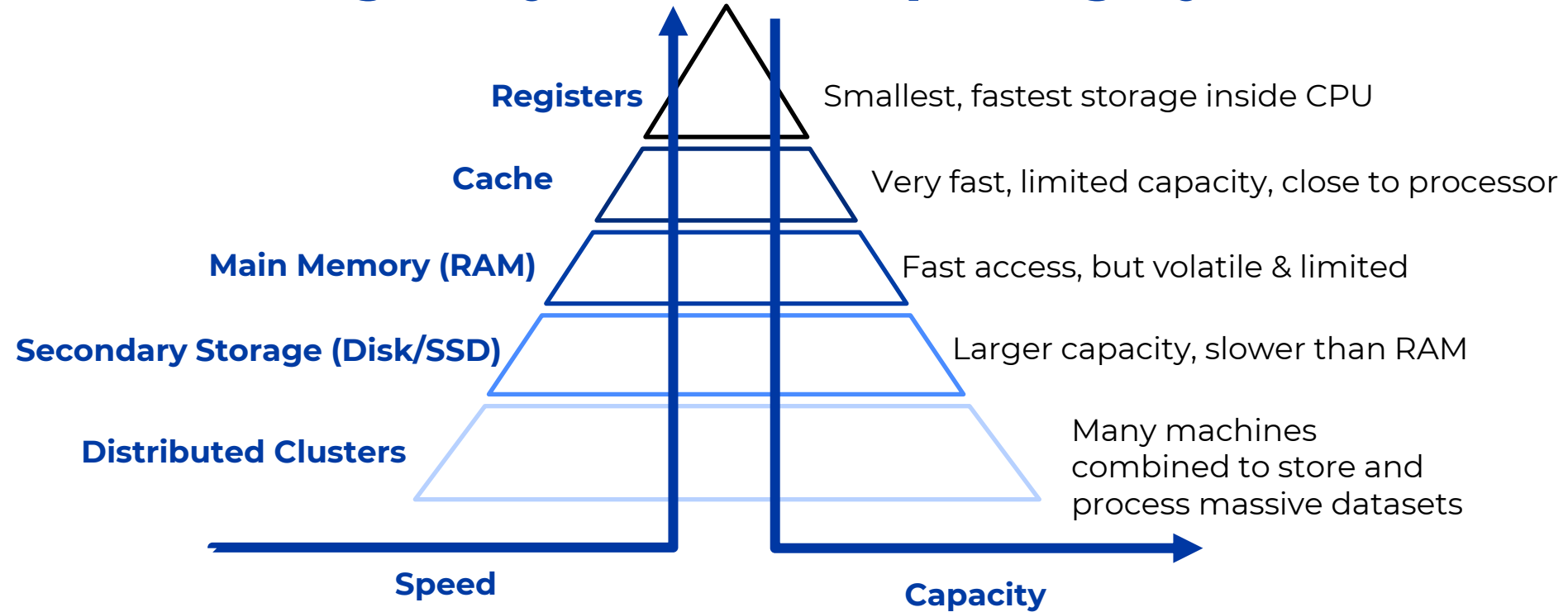




06

Memory Hierarchy & Distributed Storage

Storage Layers in Computing Systems



Performance depends on locality of access

Clusters for Big Data Storage

Definition:

A cluster is a group of interconnected machines working as one system

Shared storage:

Data is distributed across nodes, often replicated for reliability

Locality:

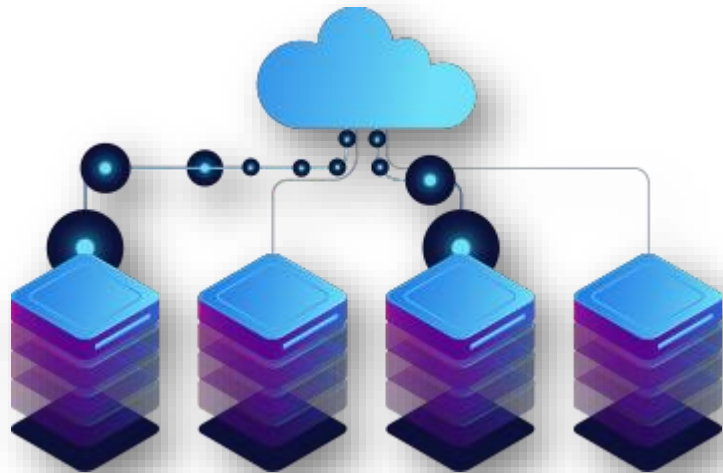
Each machine stores and processes part of the data

Scalability:

Add more nodes to handle larger datasets

Fault tolerance:

Cluster keeps working even if some machines fail



Locality Principle in Big Data

Problem: Moving huge datasets across the network is **slow** & **expensive**.

Idea:

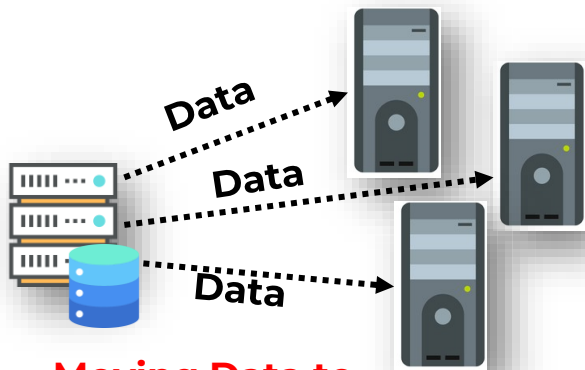
Instead of moving data to computation, **we move computation** close to where data is stored

Implementation: Each node processes its local data chunk

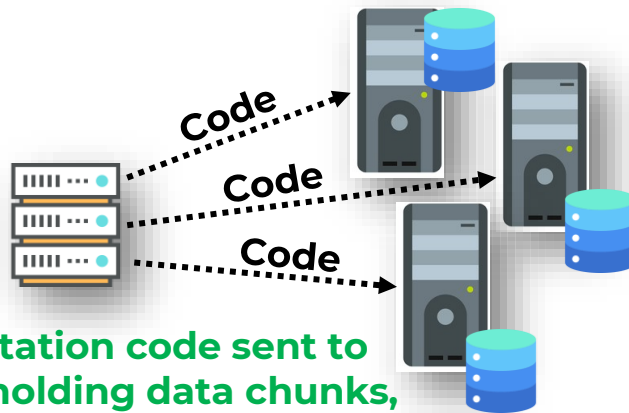
MapReduce: Embodies this principle:

“map tasks run on nodes with data, results are combined later.”

Benefit: Reduces network traffic, increases efficiency & scalability.



**Moving Data to
Computational Units**



**Computation code sent to
nodes holding data chunks,
then results combined**

07

Disciplines of Big Data

Distributed Computing for Big Data



- Batch-oriented framework based on MapReduce
- Uses HDFS for distributed storage
- Designed for fault tolerance and scalability



- In-memory distributed processing engine
- Faster than Hadoop for iterative and interactive tasks
- Rich ecosystem: Spark SQL, MLlib, Streaming, GraphX

Common Goal:

Process massive datasets across clusters of machines efficiently.



Dataset split → nodes process chunks → results combined

Parallel computing (HPC, GPU, supercomputers)

HPC (High Performance Computing):

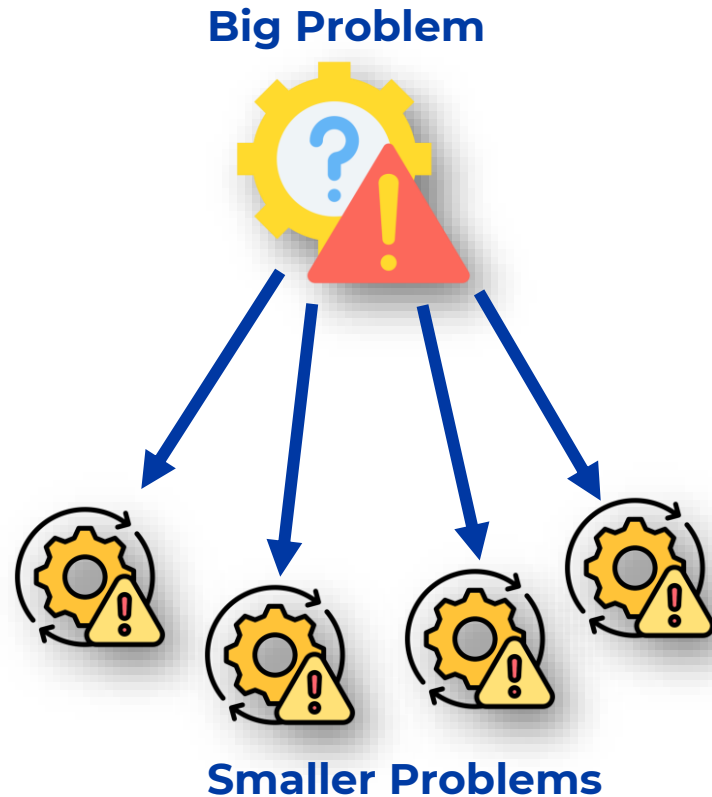
- Large-scale clusters & supercomputers
- Solve scientific & engineering problems (e.g., simulations, climate models)

GPU Computing:

- Thousands of small cores optimized for parallel tasks
- Widely used in AI, deep learning, and graphics processing

Supercomputers:

- Exascale systems (10^{18} operations/sec)
- Used for weather prediction, physics, genomics, space exploration



Types of NoSQL Databases

Document Stores 📁

- Store semi-structured data (JSON, BSON, XML)
- Flexible schema, great for hierarchical data
- **Examples:** MongoDB, CouchDB

Key-Value Stores 🔑➡📦

- Simplest NoSQL model: key maps directly to a value
- Ultra-fast lookups, caching, session storage
- **Examples:** Redis, DynamoDB

NoSQL = Flexible, Scalable, Big Data Ready

Column Stores 📊

- Data stored by columns instead of rows
- Optimized for analytics and large-scale queries
- **Examples:** Cassandra, HBase

Graph Databases 🌐

- Data stored as nodes and edges (relationships first-class citizens)
- Perfect for social networks, recommendations, fraud detection
- **Examples:** Neo4j, ArangoDB

**End of
Chapter 1**

