Big Data Timeline

What is Hadoop?

Hadoop is a framework designed to solve Big Data problems.

Industry Use Case: Used by companies like Facebook and LinkedIn to process massive data sets.

Why is it called a Framework?

Because it's not a single tool but a combination of multiple tools and technologies — an ecosystem.

Hadoop Versions

2007 – Hadoop 1.0

2012 – Hadoop 2.0

Current – Hadoop 3.0 (latest major release)

---

3 Core Components

1. HDFS (Hadoop Distributed File System)

Used for storing large data sets in a distributed manner across multiple nodes.

Use Case: Distributed storage system used in telecom industries for call data records.

2. MapReduce

A programming model for distributed processing of large datasets using Java.
Use Case: Used by search engines to index and rank websites.

3. YARN (Yet Another Resource Negotiator)

Manages and allocates resources among different applications in a Hadoop cluster.
Example: In a 20-node cluster, YARN assigns CPU/RAM to different user jobs dynamically.

MapReduce Challenges

Writing MapReduce code is hard and verbose (written in Java).

Logic is important, but Spark is now preferred due to easier coding and faster execution.

Hadoop Ecosystem Components

1. Sqoop

For importing/exporting data between Hadoop and relational databases like MySQL.
Use Case: Data migration from on-prem databases to HDFS.

2. Oozie

Workflow scheduler to manage Hadoop jobs in a sequence.
Cloud Equivalent: Azure Data Factory for orchestration.

3. Pig

A scripting platform to process and clean data on Hadoop.
Use Case: Used in data pipelines for cleaning social media data.

4. Hive

Provides an SQL-like interface to query data stored in Hadoop (Warehouse).
Use Case: Analytics for marketing campaign data using SQL queries.

5. HBase

A NoSQL database used for fast, random access to large datasets.
Cloud Equivalent: Cosmos DB in Azure.

Why NoSQL over Hive in Some Cases?

Hive queries are internally converted to MapReduce — works like sequential search.

For faster lookup (e.g., querying employee ID = 800000), NoSQL like HBase is better as it allows random access.

Challenges with Hadoop

MapReduce is slow and complex to code.

Steep learning curve — every component has its own configuration and use case.

Primarily On-premise — needs infrastructure setup (unlike cloud-native solutions).

Summary

1. What is Hadoop:
A Big Data framework to process and store large-scale data using distributed computing.

2. Core Components:

HDFS: Distributed storage

MapReduce: Distributed computation

YARN: Resource management

3. Ecosystem Tools:
Includes Sqoop, Hive, Pig, Oozie, HBase, each solving specific data engineering needs.

4. Challenges:

Complex code (MapReduce)

Learning multiple tools

On-prem infrastructure needss