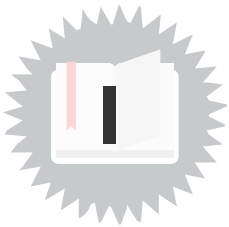
Big Data and Cloud Computing

Dr. Walid Miloud Dahmane khemis miliana University SMI Department Email : w.miloud-dahmane@univ-dbkm.dz

1.0 March 2025

Table of contents

Chapter objectives



By the end of this chapter, the student will be able to:

- Learn the main Big Data tools and how they help manage and analyze large data efficiently.

Chapter 4: Big Data Tools

1. Overview

Introduction

Big Data tools are software and technologies that help **collect, store, process, and analyze** huge amounts of data **quickly and efficiently**. These tools make working with large data easier and allow extracting valuable information to support **decision-making**.

Extra

Apache Flink is an open-source framework and distributed processing engine designed for working with large-scale data. It is especially powerful because it can handle both real-time (stream) processing and batch processing in the same system.

Flink processes data across many connected computers (a cluster), allowing it to work with massive amounts of information at very high speed. In stream processing mode, it can process events the moment they arrive—making it useful for applications like fraud detection, live analytics, and monitoring systems. In batch mode, it can handle large stored datasets for analysis or transformation.

It also offers features like fault tolerance (recovery if something fails), scalability (adding more machines to handle more data), and support for complex event processing. Flink is widely used in industries that need fast, reliable, and scalable data handling.

Apache Spark is an open-source big data processing engine designed to handle very large datasets quickly and efficiently. It can work in both batch processing (handling stored data in bulk) and stream processing (handling data in real time).

Spark runs on a cluster of computers, dividing the work among multiple nodes so tasks finish much faster than on a single machine. One of its main strengths is in-memory processing, meaning it keeps data in RAM instead of reading it from disk each time, which makes it much faster than older systems like Hadoop MapReduce.

It supports many tasks such as data analysis, machine learning, graph processing, and streaming. Spark can work with data from many sources like Hadoop Distributed File System (HDFS), NoSQL databases, or cloud storage. Because it's flexible, scalable, and fast, Spark is used in industries for real-time analytics, recommendation systems, large-scale data transformation, and scientific research.

Apache Hive is an open-source data warehouse system built on top of Hadoop. It is designed to make working with large datasets easier by allowing users to write queries in a language similar to SQL, called HiveQL, instead of writing complex MapReduce programs.

Hive translates these SQL-like queries into MapReduce, Spark, or Tez jobs that run on a Hadoop cluster, so it can process massive amounts of data stored in the Hadoop Distributed File System (HDFS) or compatible storage systems.

It is mainly used for batch processing—analyzing and summarizing historical data rather than real-time data. Hive is popular for tasks like data summarization, reporting, and business intelligence because it's easier for people familiar with SQL to use.

In short, Hive acts as a **bridge** between SQL users and the Hadoop ecosystem, making big data querying more **accessible**.

Apache HBase is an open-source, non-relational (NoSQL) database that runs on top of the Hadoop Distributed File System (HDFS). It is designed to store and manage very large amounts of data—billions of rows and millions of columns—across many machines in a cluster.

Unlike traditional relational databases, HBase stores data in a column-oriented way, which makes it faster for certain types of big data operations. It is especially good for handling sparse data (where many fields might be empty) and for scenarios that require quick read and write access to large datasets.

HBase is often used for real-time data access on top of Hadoop, such as storing user profiles, time-series data, or logs that need fast lookup and updates. It also integrates with tools like Hive, Spark, and Pig for analytics.

In short, HBase provides **real-time, scalable, and fault-tolerant** storage for massive datasets in the Hadoop ecosystem.

Apache Storm is an open-source, real-time stream processing system. It is designed to process data as soon as it arrives, making it useful for applications that need instant analysis and action.

Storm works on a cluster of computers, where incoming data streams are split into smaller tasks and processed in parallel across different nodes. Unlike batch systems (which handle stored data in bulk), Storm focuses on continuous, never-ending data flows—like tweets, sensor readings, stock market updates, or server logs.

It is fast (processing millions of messages per second), scalable (can handle more data by adding more machines), and fault-tolerant (can recover from failures without losing data).

In short, Apache Storm is ideal for **real-time analytics, monitoring systems, live dashboards**, and **event-driven** applications.

Elasticsearch is an open-source search and analytics engine designed to store, search, and analyze large volumes of data very quickly. It is often used when you need to find information fast—even in datasets containing millions or billions of records.

It stores data in a JSON document format and indexes it so that searches are extremely fast. This makes it great for full-text search (like searching words in documents), as well as filtering and aggregating structured data.

Elasticsearch works in a cluster of servers, meaning it can handle huge datasets by splitting the data across multiple nodes. It is also fault-tolerant and scalable, so you can add more servers to handle more data or more users.

It's commonly used for:

- Log analysis (often with Logstash and Kibana in the ELK stack)
- Real-time search on websites or apps
- Data monitoring and visualization

In short, Elasticsearch is like a **super-fast**, **scalable** search engine for both text and structured data.

Talend is an open-source data integration and management tool that helps collect, clean, transform, and move data between different systems. It is used to make sure data from different sources can work together in one place.

With Talend, you can connect to many types of data sources—databases, cloud storage, applications, APIs, big data systems—and create data pipelines that automatically process and transfer the data. It has a visual interface, so instead of writing all the code yourself, you can build workflows by dragging and dropping components.

Talend is useful for:

- Data integration – combining data from different systems.
- ETL (Extract, Transform, Load) – taking data from a source, changing it to fit your needs, and loading it into a target system.
- Big data processing – working with tools like Hadoop, Spark, or cloud data warehouses.
- Data quality – detecting and fixing errors in the data.

In short, Talend is a **data workflow** tool that makes it easier to prepare and move data where it's needed, in the right format and quality.

MongoDB is an open-source, NoSQL database designed to store and manage large amounts of data in a flexible way. Unlike traditional databases that use tables and rows, MongoDB stores data as documents in a format similar to JSON (key–value pairs).

This flexibility means each document can have its own structure—perfect for handling unstructured or semi-structured data like logs, user profiles, product catalogs, or sensor data. You can also change the data structure easily without redesigning the whole database.

MongoDB is:

- Scalable – can store data across many servers in a cluster.
- Fast – optimized for quick reads and writes.
- Flexible – no fixed schema, so data can evolve over time.
- Geared for modern apps – integrates well with web, mobile, and IoT applications.

It's often used for real-time analytics, content management, e-commerce platforms, and applications that deal with rapidly changing data.

In short, MongoDB is a **flexible**, **document-based database** that makes storing and retrieving big, varied datasets easy and fast.


mongoDB.