

Apache Spark and Hadoop Ecosystem Processing

Batch analytics, streaming, graph processing and data integration

FA-0751 · 5 days

Data & Analytics · DevOps, Cloud & Infrastructure · Software Development

Introduction

Build practical Spark data-processing workflows and connect them to the wider Hadoop ecosystem. The programme combines guided installation, coding exercises and a final applied problem across five days.

Core Spark exercises receive hands-on attention. The broader ingestion, database and ecosystem landscape is explored through selected demonstrations and comparisons, with legacy technologies clearly distinguished from current deployment options.

Learning outcomes

- Set up a compatible local Spark environment and explain how an application runs on a cluster.
- Frame a data-analysis problem as distributed transformations and actions using RDDs, Spark SQL and DataFrames.
- Use broadcast variables and accumulators appropriately, and work through a breadth-first-search example.
- Explore clustering, machine-learning and graph-processing examples using the appropriate Spark APIs.
- Build a guided Structured Streaming example and explain the older Spark Streaming/DStreams model.
- Compare relational, flat-file and non-relational integration patterns, including Hive, MySQL, HBase, Cassandra and MongoDB.
- Explain the roles of ingestion, execution, coordination, notebook and orchestration tools without treating them all as cluster managers.

Prerequisites

Basic programming, SQL and command-line familiarity. A compatible Spark runtime and prepared datasets are required; cluster and external-system demonstrations use a prepared teaching environment.

Training outline

1. Day 1 — Spark foundations and setup

- Spark execution concepts and the RDD interface: transformations, actions and distributed processing.
- Guided local installation, libraries, resources and a first data-processing workflow.
- Practice exercises and a foundation assessment.

2. Day 2 — Structured data and iterative processing

- Spark SQL, DataFrames and the Dataset API in languages that support it.
- Advanced processing patterns and an introduction to partitioning and reuse of intermediate results.
- Broadcast variables and accumulators: appropriate uses and limitations.
- Guided iterative breadth-first-search exercise.

3. Day 3 — Clustering, machine learning, streaming and graphs

- Distributed execution and clustering concepts; a selected Spark machine-learning example.
- Structured Streaming with DataFrames, checkpointing and a prepared streaming input.
- Older DStreams-based Spark Streaming as a legacy comparison; latency and delivery behaviour depend on mode and connectors.
- GraphX graph-processing concepts and a selected Scala example.

4. Day 4 — Hadoop data processing and integration

- Pig scripting as a legacy Hadoop processing approach; revisit equivalent Spark and RDD patterns.
- Hive querying, relational integration with MySQL and flat-file data processing.
- HBase, Cassandra and MongoDB integration concepts with selected prepared examples.
- Kafka ingestion and the historical roles of Flume and retired Sqoop; choose maintained connectors for current exercises.

- Distinguish YARN resource management, Tez execution, ZooKeeper coordination and Zeppelin/Hue user tools. Spark deployment options include standalone, YARN and Kubernetes; Mesos and retired Oozie are historical ecosystem context.

5. Day 5 — Querying and applied processing

- Query integrated datasets and compare batch and streaming problem requirements.
- Apache Storm concepts and a guided example or prepared demonstration; optional comparison with Flink and Spark streaming.
- Apply selected techniques to a real-world-style dataset, review results and discuss trade-offs.