CAP - Developing with Spark and Hadoop

cloudera®

ACADEMIC PARTNER

Course Description

The Big Data landscape is continuously evolving as new technologies emerge and existing technologies mature. This is a comprehensive course covering Spark and key elements of the Hadoop Ecosystem used in developing end to end applications for processing Big Data efficiently.

StudentswhocompletethiscoursewillunderstandkeySparkandHadoopconcepts, and they will learn to apply Spark and Hadoop tools in developing applications for solving the types of problems faced by enterprises and research institutions today.

Prerequisites

This course is designed for developers and engineers who have programming experience. Apache Spark examples and homework labs are presented in Scala and Python, therefore, the ability to program in one of those languages is required. Basic familiarity with the Linux command line is assumed. Basic knowledge of SQL is helpful; prior knowledge of Hadoop is not required.

Texts

Recommended

- Learning Spark, by Karau, Konwinski, Wendell, and Zaharia

Optional

- Hadoop: The Definitive Guide (third edition), by Tom White
- Using Flume, by Hari Shreedharan
- Hadoop Operations, by Eric Sammer
- Programming Hive, by Capriolo, Wampler, and Rutherglen
- Advanced Analytics with Spark, by Ryza, Laserson, Owen, and Wills

Course Objectives

During this course, you will learn:

- How the Hadoop Ecosystem fits in with the data processing lifecycle
- How data is distributed, stored and processed in a Hadoop cluster
- How to use Sqoop and Flume to ingest data
- How to process distributed data with Spark

- How to model structured data as tables in Impala and Hive
- How to choose a data storage format for your data usage patterns
- Best practices for data storage

Course Outline

Module 1: Course Introduction

1. **Introduction**
   a) About This Course
   b) About Cloudera

Module 2: Introduction to Hadoop

2. **Introduction to Hadoop and the Hadoop Ecosystem**
   a) Problems with Traditional Large-scale Systems
   b) Hadoop!
   c) Data Storage and Ingest
   d) Data Processing
   e) Data Analysis and Exploration
   f) Other Ecosystem Tools
   g) Introduction to the Homework Labs
   h) **Homework Labs:** Setup and General Notes

3. **Hadoop Architecture and HDFS**
   a) Distributed Processing on a Cluster
   b) Storage: HDFS Architecture
   c) Storage: Using HDFS
   d) **Homework Lab:** Access HDFS with Command Line and Hue
   e) Resource Management: YARN Architecture
   f) Resource Management: Working with YARN
   g) **Homework Lab:** Run a YARN Job

Module 3: Importing and Modeling Structured Data

4. **Importing Relational Data with Apache Sqoop**
   a) Sqoop Overview
   b) Basic Imports and Exports
   c) Limiting Results
   d) Improving Sqoop's Performance
   e) Sqoop 2
   f) **Homework Lab:** Import Data from MySQL Using Sqoop

5. Introduction to Impala and Hive

a) Introduction to Impala and Hive
b) Why Use Impala and Hive
c) Querying Data With Impala and Hive
d) Comparing Hive and Impala to Traditional Databases

6. Modeling and Managing Data with Impala and Hive

a) Data Storage Overview
b) Creating Databases and Tables
c) Loading Data into Tables
d) HCatalog
e) Impala Metadata Caching
f) **Homework Lab:** Create and Populate Tables in Impala or Hive

7. Data Formats

a) File Formats
b) Avro Schemas
c) Avro Schema Evolution
d) Using Avro with Impala, Hive and Sqoop
e) Using Parquet with Impala, Hive and Sqoop
f) Compression
g) **Homework Lab**: Select a Format for a Data File

8. Data File Partitioning

a) Partitioning Overview
b) Partitioning in Impala and Hive
c) Conclusion
d) **Homework Lab**: Partition Data in Impala or Hive

Module 4: Ingesting Streaming Data

9. Capturing Data with Apache Flume

a) What is Apache Flume
b) Basic Flume Architecture
c) Flume Sources
d) Flume Sinks
e) Flume Channels
f) Flume Configuration
g) **Homework Lab:** Collect Web Server Logs with Flume

Module 5: Distributed Data Processing with Spark

10.Spark Basics

a) What is Apache Spark

b) Using the Spark Shell

c) RDDs Resilient Distributed Datasets)

d) Functional Programming in Spark

e) **Homework Labs:**

- View the Spark Documentation
- Explore RDDs Using the Spark Shell
- Use RDDs to Transform a Dataset

11.Working with RDDs in Spark

a) Creating RDDs

b) Other General RDD Operations

c) **Homework Lab:** Process Data Files with Spark

12.Aggregating Data with Pair RDDs

a) Key Value Pair RDDs

b) Map Reduce

c) Other Pair RDD Operations

d) **Homework Lab:** Use Pair RDDs to Join Two Datasets

13.Writing and Deploying Spark Applications

a) Spark Applications vs. Spark Shell

b) Creating the SparkContext

c) Building a Spark Application Scala and Java)

d) Running a Spark Application

e) The Spark Application Web UI

f) **Homework Lab:** Write and Run a Spark Application

g) Configuring Spark Properties

h) Logging

i) **Homework Lab:** Configure a Spark Application

14.Parallel Processing in Spark

a) Review: Spark on a Cluster

b) RDD Partitions

c) Partitioning of File based RDDs

d) HDFS and Data Locality

e) Executing Parallel Operations

f) Stages and Tasks

g) **Homework Lab:** View Jobs and Stages in the Spark Application UI

15.Spark RDD Persistence

a) RDD Lineage
b) RDD Persistence Overview
c) Distributed Persistence
d) **Homework Lab**: Persist an RDD

16. Common Patterns in Spark Data Processing

a) Common Spark Use Cases
b) Iterative Algorithms in Spark
c) Graph Processing and Analysis
d) Machine Learning
e) Example: k means
f) **Homework Lab**: Iterative Processing in Spark
g) **Optional Homework Lab**: Partition Data Files Using Spark

17. Spark SQL and DataFrames

a) Spark SQL and the SQL Context
b) Creating DataFrames
c) Transforming and Querying DataFrames
d) Saving DataFrames
e) DataFrames and RDDs
f) Comparing Spark SQL, Impala and Hive on Spark
g) **Homework Lab:** Use Spark SQL for ETL

Module: Course Conclusion

18. Conclusion