Big-Data and Hadoop Developer
Developer Training for Apache Hadoop

COURSE BROCHURE & SYLLABUS

Course Overview

Apache Hadoop is an open-source software framework written in Java for distributed storage and distributed processing of very large data sets on computer clusters built from commodity hardware. All the modules in Hadoop are designed with a fundamental assumption that hardware failures are common and should be automatically handled by the framework. Apache Hadoop's MapReduce and HDFS components were inspired by Google papers on their MapReduce and Google File System.

The Hadoop framework itself is mostly written in the Java programming language, with some native code in C and command line utilities written as shell scripts. Though MapReduce Java code is common, any programming language can be used with "Hadoop Streaming" to implement the "map" and "reduce" parts of the user's program. Other projects in the Hadoop ecosystem expose richer user interfaces.

This Developer training course for Hadoop Trainings delivers the key concepts and expertise necessary to create robust data processing applications using Apache Hadoop.

Audience Profile

This course is intended and appropriate for developers who will be writing, maintaining, or optimizing Hadoop jobs

Participants should have programming experience, preferably with Java. Understanding of common computer science concepts is a plus.

At Course Completion

Through instructor-led discussion and interactive, hands-on exercises, participants will navigate the Hadoop ecosystem, learning topics such as:

- MapReduce and the Hadoop Distributed File System (HDFS) and how to write MapReduce code
- Best practices and considerations for Hadoop development
- Debugging techniques and implementation of workflows and common algorithms
- How to leverage Hive, Pig, Sqoop, Flume, Oozie and other projects from the Apache Hadoop ecosystem
- Optimal hardware configurations and network considerations for building out maintaining and monitoring your Hadoop cluster
- Advanced Hadoop API topics required for real-world data analysis

Developer Training for Apache Hadoop

Course Duration: 30 Hours

Course Outline

- **The Motivation for Hadoop**
  - Problems with traditional large-scale systems
  - Requirements for a new approach
- **Hadoop: Basic Concepts**
  - An Overview of Hadoop
  - The Hadoop Distributed File System
  - How MapReduce Works
  - Anatomy of a Hadoop Cluster
  - Other Hadoop Ecosystem Components
- **Writing a MapReduce Program**
  - The MapReduce Flow
  - Examining a Sample MapReduce Program
  - Basic MapReduce API Concepts
  - The Driver Code
  - The Mapper
  - The Reducer
  - Hadoop's Streaming API
  - Using Eclipse for Rapid Development
- **Integrating Hadoop into the Workflow**
  - Relational Database Management System
  - Storage Systems
  - Importing Data from RDBMS With Sqoop
  - Importing Real-Time Data with Flume
  - Accessing HDFS Using FuseDFS and Hoop
- **Graph Manipulation in Hadoop**
  - Introduction to graph techniques
  - Representing graphs in Hadoop
  - Implementing a sample algorithm: Single Source Shortest Path
- **Using Hive and Pig**
  - Hive Basics
  - Pig Basics

- **Delving Deeper Into the Hadoop API**
    - Using LocalJobRunner Mode for Faster Development
    - Reducing Intermediate Data With Combiners
    - The configure and close methods for Map/Reduce Setup and Teardown
    - Writing Partitioners for Better Load Balancing
    - Directly Accessing HDFS
    - Using the Distributed Cache
- **Practical Development Tips and Techniques**
    - Testing with MRUnit
    - Debugging MapReduce Code
    - Using LocalJobRunner Mode for Easier Debugging
    - Eclipse development techniques
    - Retrieving Job Information with Counters
    - Logging
    - Splittable File Formats
    - Determining the Optimal Number of Reducers
    - MapReduce Jobs
    - Implementing Multiple Mappers using ChainMapper
- **Common MapReduce Algorithms**
    - Sorting and Searching
    - Indexing
    - Machine Learning with Mahout
    - Term Frequency - Inverse Document Frequency
    - Word Co-Occurrence
- **Advanced MapReduce Programming**
    - Custom Writables and WritableComparables
    - Saving Binary Data using SequenceFiles and Avro Files
    - Creating InputFormats and OutputFormats
- **Joining Data Sets in MapReduce Jobs**
    - Map-Side Joins
    - The Secondary Sorts
    - Sort Reduce-Side Joins
- **Creating Workflows with Oozie**
    - The Motivation for Oozie's Workflow
    - Oozie’s Workflow Definition Format