

# HADOOP

## Descripción General

Curso orientado a usuarios y desarrolladores de aplicaciones que requieren acceder a grandes volúmenes de datos para llevar a cabo el procesamiento y análisis de información

## Objetivo

Que el estudiante aprenda y aplique las técnicas y metodologías de procesamiento y análisis de grandes volúmenes de datos en un sistema Hadoop

## Duración

60 Horas 10 días

## Requisitos

Conocimientos básicos de Linux

1. Meet Hadoop

- 1.1. Data!
- 1.2. Data Storage and Analysis
- 1.3. Comparison with Other Systems
- 1.4. Rational Database Management System
- 1.5. Grid Computing
- 1.6. Volunteer Computing
- 1.7. A Brief History of Hadoop
- 1.8. Apache Hadoop and the Hadoop Ecosystem
- 1.9. Hadoop Releases
- 1.10. What's Covered in This Book
- 1.11. Compatibility

2. MapReduce

- 2.1. A Weather Dataset
- 2.2. Data Format
- 2.3. Analyzing the Data with Unix Tools
- 2.4. Analyzing the Data with Hadoop
- 2.5. Map and Reduce

2.6. Java MapReduce

2.7. Scaling Out

2.8. Data Flow

2.9. Combiner Functions

2.10. Running a Distributed MapReduce Job

2.11. Hadoop Streaming

2.12. Ruby

2.13. Python

2.14. Hadoop Pipes

2.15. Compiling and Running

3. The Hadoop Distributed Filesystem

3.1. The Design of HDFS

3.2. HDFS Concepts

3.3. Blocks

3.4. Namenodes and Datanodes

3.5. HDFS Federation

3.6. HDFS High-Availability

3.7. The Command-Line Interface

3.8. Basic Filesystem Operations

3.9. Hadoop Filesystems

- 3.10. Interfaces
- 3.11. The Java Interface
- 3.12. Reading Data from a Hadoop URL
- 3.13. Reading Data Using the FileSystem API
- 3.14. Writing Data
- 3.15. Directories
- 3.16. Querying the Filesystem
- 3.17. Deleting Data
- 3.18. Data Flow
- 3.19. Anatomy of a File Read
- 3.20. Anatomy of a File Write
- 3.21. Coherency Model
- 3.22. Data Ingest with Flume and Sqoop
- 3.23. Parallel Copying with distcp
- 3.24. Keeping an HDFS Cluster Balanced
- 3.25. Hadoop Archives
- 3.26. Using Hadoop Archives
- 3.27. Limitations
- 4. Hadoop I/O
  - 1.1. Data Integrity
  - 1.2. Data Integrity in HDFS
  - 1.3. LocalFileSystem
  - 1.4. ChecksumFileSystem
  - 1.5. Compression
  - 1.6. Codecs
  - 1.7. Compression and Input Splits
  - 1.8. Using Compression in MapReduce Serialization
  - 1.9. The Writable Interface
    - 1.10. Writable Classes
    - 1.11. Implementing a Custom Writable
    - 1.12. Serialization Frameworks
    - 1.13. Avro
    - 1.14. Avro Data Types and Schemas
    - 1.15. In-Memory Serialization and Deserialization
    - 1.16. Avro Datafiles
    - 1.17. Interoperability
    - 1.18. Schema Resolution
    - 1.19. Sort Order
    - 1.20. Avro MapReduce
    - 1.21. Sorting Using Avro MapReduce

- 1.22. Avro MapReduce in Other Languages
- 1.23. File-Based Data Structures
- 1.24. SequenceFile
- 1.25. MapFile
- 5. Developing a MapReduce Application
  - 1.1. The Configuration API
  - 1.2. Combining Resources
  - 1.3. Variable Expansion
  - 1.4. Setting Up the Development Environment
  - 1.5. Managing Configuration
  - 1.6. GenericOptionsParser, Tool, and ToolRunner
  - 1.7. Writing a Unit Test with MRUnit
  - 1.8. Mapper
  - 1.9. Reducer
  - 1.10. Running Locally on Test Data
  - 1.11. Running a Job in a Local Job Runner
  - 1.12. Testing the Driver
  - 1.13. Running on a Cluster
  - 1.14. Packaging a Job
  - 1.15. Launching a Job
- 1.16. The MapReduce Web UI
- 1.17. Retrieving the Results
- 1.18. Debugging a Job
- 1.19. Hadoop Logs
- 1.20. Remote Debugging
- 1.21. Tuning a Job
- 1.22. Profiling Tasks
- 1.23. MapReduce Workflows
- 1.24. Decomposing a Problem into MapReduce Jobs
- 1.25. JobControl
- 1.26. Apache Oozie
- 6. How MapReduce Works
  - 6.1. Anatomy of a MapReduce Job Run
  - 6.2. Classic MapReduce (MapReduce 1)
  - 6.3. YARN (MapReduce 2) Failures
  - 6.4. Failures in Classic MapReduce
  - 6.5. Failures in YARN
  - 6.6. Job Scheduling
  - 6.7. The Fair Scheduler
  - 6.8. The Capacity Scheduler



- 6.9. Shuffle and Sort
- 6.10. The Map Side
- 6.11. The Reduce Side
- 6.12. Configuration Tuning
- 6.13. Task Execution
- 6.14. The Task Execution Environment
- 6.15. Speculative Execution
- 6.16. Output Committers
- 6.17. Task JVM Reuse
- 6.18. Skipping Bad Records
- 7. MapReduce Types and Formats
  - 7.1. MapReduce Types
  - 7.2. The Default MapReduce Job
  - 7.3. Input Formats
  - 7.4. Input Splits and Records
  - 7.5. Text Input
  - 7.6. Binary Input
  - 7.7. Multiple Inputs
  - 7.8. Database Input (and Output)
  - 7.9. Output Formats
- 7.10. Text Output
- 7.11. Binary Output
- 7.12. Multiple Outputs
- 7.13. Lazy Output
- 7.14. Database Output
- 8. MapReduce Features
  - 8.1. Counters
  - 8.2. Built-in Counters
  - 8.3. User-Defined Java Counters
  - 8.4. User-Defined Streaming Counters
  - 8.5. Sorting
  - 8.6. Preparation
  - 8.7. Partial Sort
  - 8.8. Total Sort
  - 8.9. Secondary Sort
  - 8.10. Joins
  - 8.11. Map-Side Joins
  - 8.12. Reduce-Side Joins
  - 8.13. Side Data Distribution
  - 8.14. Using the Job Configuration

- 8.15. Distributed Cache
- 8.16. MapReduce Library Classes
- 9. Setting Up a Hadoop Cluster
  - 9.1. Cluster Specification
  - 9.2. Network Topology
  - 9.3. Cluster Setup and Installation
  - 9.4. Installing Java
  - 9.5. Creating a Hadoop User
  - 9.6. Installing Hadoop
  - 9.7. Testing the Installation
  - 9.8. SSH Configuration
  - 9.9. Hadoop Configuration
  - 9.10. Configuration Management
  - 9.11. Environment Settings
  - 9.12. Important Hadoop Daemon Properties
  - 9.13. Hadoop Daemon Addresses and Ports
  - 9.14. Other Hadoop Properties
  - 9.15. User Account Creation
  - 9.16. YARN Configuration
  - 9.17. Important YARN Daemon Properties
- 9.18. YARN Daemon Addresses and Ports
- 9.19. Security
- 9.20. Kerberos and Hadoop
- 9.21. Delegation Tokens
- 9.22. Other Security Enhancements
- 9.23. Benchmarking a Hadoop Cluster
- 9.24. Hadoop Benchmarks
- 9.25. User Jobs
- 9.26. Hadoop in the Cloud
- 9.27. Apache Whirr
- 10. Administering Hadoop
  - 10.1. HDFS
  - 10.2. Persistent Data Structures
  - 10.3. Safe Mode
  - 10.4. Audit Logging
  - 10.5. Tools
  - 10.6. Monitoring
  - 10.7. Logging
  - 10.8. Metrics
  - 10.9. Java Management Extensions

[www.cursoslinux.com.mx](http://www.cursoslinux.com.mx)

[ventas@plct.com.mx](mailto:ventas@plct.com.mx)

PLCT S.A. de C.V.

Tel.: 55 4522 7839/55 1800 7696/7224447684

- 10.10. Maintenance
- 10.11. Routine Administration Procedures
- 10.12. Commissioning and Decommissioning Nodes
- 10.13. Upgrades
- 11. Pig
  - 11.1. Installing and Running Pig
  - 11.2. Execution Types
  - 11.3. Running Pig Programs
  - 11.4. Grunt
  - 11.5. Pig Latin Editors
  - 11.6. An Example
  - 11.7. Generating Examples
  - 11.8. Comparison with Databases
  - 11.9. Pig Latin
    - 11.10. Structure
    - 11.11. Statements
    - 11.12. Expressions
    - 11.13. Types
    - 11.14. Schemas
    - 11.15. Functions
- 11.16. Macros
- 11.17. User-Defined Functions
- 11.18. A Filter UDF
- 11.19. An Eval UDF
- 11.20. A Load UDF
- 11.21. Data Processing Operators
- 11.22. Loading and Storing Data
- 11.23. Filtering Data
- 11.24. Grouping and Joining Data
- 11.25. Sorting Data Combining and Splitting Data
- 11.26. Pig in Practice
- 11.27. Parallelism
- 11.28. Parameter Substitution
- 12. Hive
  - 12.1. Installing Hive
  - 12.2. The Hive Shell
  - 12.3. An Example
  - 12.4. Running Hive
  - 12.5. Configuring Hive
  - 12.6. Hive Services

- 12.7. The Metastore
- 12.8. Comparison with Traditional Databases
- 12.9. Schema on Read Versus Schema on Write
- 12.10. Updates, Transactions, and Indexes
- 12.11. HiveQL
- 12.12. Data Types
- 12.13. Operators and Functions
- 12.14. Tables
- 12.15. Managed Tables and External Tables
- 12.16. Partitions and Buckets
- 12.17. Storage Formats
- 12.18. Importing Data
- 12.19. Altering Tables
- 12.20. Dropping Tables
- 12.21. Querying Data
- 12.22. Sorting and Aggregating
- 12.23. MapReduce Scripts
- 12.24. Joins
- 12.25. Subqueries
- 12.26. Views
- 12.27. User-Defined Functions
- 12.28. Writing a UDF
- 12.29. Writing a UDAF
- 13. HBase
  - 13.1. HBasics
  - 13.2. Backdrop
  - 13.3. Concepts
  - 13.4. Whirlwind Tour of the Data Model
  - 13.5. Implementation
  - 13.6. Installation
  - 13.7. Test Drive
  - 13.8. Clients
  - 13.9. Java
  - 13.10. Avro, REST, and Thrift
  - 13.11. Example
  - 13.12. Schemas
  - 13.13. Loading Data
  - 13.14. Web Queries
  - 13.15. HBase Versus RDBMS
  - 13.16. Successful Service



- 13.17. HBase
- 13.18. Use Case: HBase at Streamy.com
- 13.19. Praxis
- 13.20. Versions
- 13.21. HDFS
- 13.22. UI
- 13.23. Metrics
- 13.24. Schema Design
- 13.25. Counters
- 13.26. Bulk Load
- 14. ZooKeeper
  - 14.1. Installing and Running ZooKeeper
  - 14.2. An Example
  - 14.3. Group Membership in ZooKeeper
  - 14.4. Creating the Group
  - 14.5. Joining a Group
  - 14.6. Listing Members in a Group
  - 14.7. Deleting a Group
  - 14.8. The ZooKeeper Service
  - 14.9. Data Model
  - 14.10. Operations
  - 14.11. Implementation
  - 14.12. Consistency
  - 14.13. Sessions
  - 14.14. States
  - 14.15. Building Applications with ZooKeeper
  - 14.16. A Configuration Service
  - 14.17. The Resilient ZooKeeper Application
  - 14.18. A Lock Service
  - 14.19. More Distributed Data Structures and Protocols
  - 14.20. ZooKeeper in Production
  - 14.21. Resilience and Performance
  - 14.22. Configuration
- 15. Sqoop
  - 15.1. Getting Sqoop
  - 15.2. Sqoop Connectors
  - 15.3. A Sample Import
  - 15.4. Text and Binary File Formats
  - 15.5. Generated Code
  - 15.6. Additional Serialization Systems

[www.cursoslinux.com.mx](http://www.cursoslinux.com.mx)

[ventas@plct.com.mx](mailto:ventas@plct.com.mx)

PLCT S.A. de C.V.

Tel.: 55 4522 7839/55 1800 7696/7224447684

- 15.7. Imports: A Deeper Look
- 15.8. Controlling the Import
- 15.9. Imports and Consistency
- 15.10. Direct-mode Imports
- 15.11. Working with Imported Data
- 15.12. Imported Data and Hive
- 15.13. Importing Large Objects
- 15.14. Performing an Export
- 15.15. Exports: A Deeper Look
- 15.16. Exports and Transactionality
- 15.17. Exports and SequenceFiles
- 16. Case Studies
  - 16.1. Hadoop Usage at Last.fm
  - 16.2. Last.fm: The Social Music Revolution
  - 16.3. Hadoop at Last.fm
  - 16.4. Generating Charts with Hadoop
  - 16.5. The Track Statistics Program
  - 16.6. Summary
  - 16.7. Hadoop and Hive at Facebook
  - 16.8. Hadoop at Facebook
  - 16.9. Hypothetical Use Case Studies
  - 16.10. Hive
  - 16.11. Problems and Future Work
  - 16.12. Nutch Search Engine
  - 16.13. Data Structures
  - 16.14. Selected Examples of Hadoop Data Processing in Nutch
  - 16.15. Summary
  - 16.16. Log Processing at Rackspace
  - 16.17. Requirements/The Problem
  - 16.18. Brief History
  - 16.19. Choosing Hadoop
  - 16.20. Collection and Storage
  - 16.21. MapReduce for Logs
  - 16.22. Cascading
  - 16.23. Fields, Tuples, and Pipes
  - 16.24. Operations
  - 16.25. Taps, Schemes, and Flows
  - 16.26. Cascading in Practice
  - 16.27. Flexibility

- 16.28. Hadoop and Cascading at ShareThis
- 16.29. Summary
- 16.30. TeraByte Sort on Apache Hadoop
- 16.31. Using Pig and Wukong to Explore Billion-edge
- 16.32. Network Graphs
- 16.33. Measuring Community
- 16.34. Everybody's Talkin' at Me: The Twitter Reply
- 16.35. Graph
- 16.36. Symmetric Links
- 16.37. Community Extraction