

FOURTH EDITION

---

# Hadoop: The Definitive Guide

*Tom White*

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

**O'REILLY**

---

# Table of Contents

<b>Foreword</b> .....	<b>xvii</b>
<b>Preface</b> .....	<b>xix</b>

---

## Part I. Hadoop Fundamentals

<b>1. Meet Hadoop</b> .....	<b>3</b>
Data!	3
Data Storage and Analysis	5
Querying All Your Data	6
Beyond Batch	7
Comparison with Other Systems	8
Relational Database Management Systems	8
Grid Computing	10
Volunteer Computing	11
A Brief History of Apache Hadoop	12
What's in This Book?	15
<b>2. MapReduce</b> .....	<b>19</b>
A Weather Dataset	19
Data Format	19
Analyzing the Data with Unix Tools	21
Analyzing the Data with Hadoop	22
Map and Reduce	22
Java MapReduce	24
Scaling Out	30
Data Flow	30
Combiner Functions	34
Running a Distributed MapReduce Job	37
Hadoop Streaming	37

---

Ruby	37
Python	40
<b>3. The Hadoop Distributed Filesystem.....</b>	<b>43</b>
The Design of HDFS	43
HDFS Concepts	45
Blocks	45
Namenodes and Datanodes	46
Block Caching	47
HDFS Federation	48
HDFS High Availability	48
The Command-Line Interface	50
Basic Filesystem Operations	51
Hadoop Filesystems	53
Interfaces	54
The Java Interface	56
Reading Data from a Hadoop URL	57
Reading Data Using the FileSystem API	58
Writing Data	61
Directories	63
Querying the Filesystem	63
Deleting Data	68
Data Flow	69
Anatomy of a File Read	69
Anatomy of a File Write	72
Coherency Model	74
Parallel Copying with distcp	76
Keeping an HDFS Cluster Balanced	77
<b>4. YARN.....</b>	<b>79</b>
Anatomy of a YARN Application Run	80
Resource Requests	81
Application Lifespan	82
Building YARN Applications	82
YARN Compared to MapReduce 1	83
Scheduling in YARN	85
Scheduler Options	86
Capacity Scheduler Configuration	88
Fair Scheduler Configuration	90
Delay Scheduling	94
Dominant Resource Fairness	95
Further Reading	96

<b>5. Hadoop I/O.....</b>	<b>97</b>
Data Integrity	97
Data Integrity in HDFS	98
LocalFileSystem	99
ChecksumFileSystem	99
Compression	100
Codecs	101
Compression and Input Splits	105
Using Compression in MapReduce	107
Serialization	109
The Writable Interface	110
Writable Classes	113
Implementing a Custom Writable	121
Serialization Frameworks	126
File-Based Data Structures	127
SequenceFile	127
MapFile	135
Other File Formats and Column-Oriented Formats	136

---

## Part II. MapReduce

<b>6. Developing a MapReduce Application.....</b>	<b>141</b>
The Configuration API	141
Combining Resources	143
Variable Expansion	143
Setting Up the Development Environment	144
Managing Configuration	146
GenericOptionsParser, Tool, and ToolRunner	148
Writing a Unit Test with MRUnit	152
Mapper	153
Reducer	156
Running Locally on Test Data	156
Running a Job in a Local Job Runner	157
Testing the Driver	158
Running on a Cluster	160
Packaging a Job	160
Launching a Job	162
The MapReduce Web UI	165
Retrieving the Results	167
Debugging a Job	168
Hadoop Logs	172

---

Remote Debugging	174
Tuning a Job	175
Profiling Tasks	175
MapReduce Workflows	177
Decomposing a Problem into MapReduce Jobs	177
JobControl	178
Apache Oozie	179
<b>7. How MapReduce Works.....</b>	<b>185</b>
Anatomy of a MapReduce Job Run	185
Job Submission	186
Job Initialization	187
Task Assignment	188
Task Execution	189
Progress and Status Updates	190
Job Completion	192
Failures	193
Task Failure	193
Application Master Failure	194
Node Manager Failure	195
Resource Manager Failure	196
Shuffle and Sort	197
The Map Side	197
The Reduce Side	198
Configuration Tuning	201
Task Execution	203
The Task Execution Environment	203
Speculative Execution	204
Output Committers	206
<b>8. MapReduce Types and Formats.....</b>	<b>209</b>
MapReduce Types	209
The Default MapReduce Job	214
Input Formats	220
Input Splits and Records	220
Text Input	232
Binary Input	236
Multiple Inputs	237
Database Input (and Output)	238
Output Formats	238
Text Output	239
Binary Output	239

Multiple Outputs	240
Lazy Output	245
Database Output	245
<b>9. MapReduce Features.....</b>	<b>247</b>
Counters	247
Built-in Counters	247
User-Defined Java Counters	251
User-Defined Streaming Counters	255
Sorting	255
Preparation	256
Partial Sort	257
Total Sort	259
Secondary Sort	262
Joins	268
Map-Side Joins	269
Reduce-Side Joins	270
Side Data Distribution	273
Using the Job Configuration	273
Distributed Cache	274
MapReduce Library Classes	279

---

## Part III. Hadoop Operations

<b>10. Setting Up a Hadoop Cluster.....</b>	<b>283</b>
Cluster Specification	284
Cluster Sizing	285
Network Topology	286
Cluster Setup and Installation	288
Installing Java	288
Creating Unix User Accounts	288
Installing Hadoop	289
Configuring SSH	289
Configuring Hadoop	290
Formatting the HDFS Filesystem	290
Starting and Stopping the Daemons	290
Creating User Directories	292
Hadoop Configuration	292
Configuration Management	293
Environment Settings	294
Important Hadoop Daemon Properties	296

Hadoop Daemon Addresses and Ports	304
Other Hadoop Properties	307
Security	309
Kerberos and Hadoop	309
Delegation Tokens	312
Other Security Enhancements	313
Benchmarking a Hadoop Cluster	314
Hadoop Benchmarks	314
User Jobs	316
<b>11. Administering Hadoop</b> .....	<b>317</b>
HDFS	317
Persistent Data Structures	317
Safe Mode	322
Audit Logging	324
Tools	325
Monitoring	330
Logging	330
Metrics and JMX	331
Maintenance	332
Routine Administration Procedures	332
Commissioning and Decommissioning Nodes	334
Upgrades	337

---

## Part IV. Related Projects

<b>12. Avro</b> .....	<b>345</b>
Avro Data Types and Schemas	346
In-Memory Serialization and Deserialization	349
The Specific API	351
Avro Datafiles	352
Interoperability	354
Python API	354
Avro Tools	355
Schema Resolution	355
Sort Order	358
Avro MapReduce	359
Sorting Using Avro MapReduce	363
Avro in Other Languages	365

<b>13. Parquet.....</b>	<b>367</b>
Data Model	368
Nested Encoding	370
Parquet File Format	370
Parquet Configuration	372
Writing and Reading Parquet Files	373
Avro, Protocol Buffers, and Thrift	375
Parquet MapReduce	377
<b>14. Flume.....</b>	<b>381</b>
Installing Flume	381
An Example	382
Transactions and Reliability	384
Batching	385
The HDFS Sink	385
Partitioning and Interceptors	387
File Formats	387
Fan Out	388
Delivery Guarantees	389
Replicating and Multiplexing Selectors	390
Distribution: Agent Tiers	390
Delivery Guarantees	393
Sink Groups	395
Integrating Flume with Applications	398
Component Catalog	399
Further Reading	400
<b>15. Sqoop.....</b>	<b>401</b>
Getting Sqoop	401
Sqoop Connectors	403
A Sample Import	404
Text and Binary File Formats	406
Generated Code	407
Additional Serialization Systems	408
Imports: A Deeper Look	408
Controlling the Import	410
Imports and Consistency	411
Incremental Imports	411
Direct-Mode Imports	411
Working with Imported Data	412
Imported Data and Hive	413
Importing Large Objects	415



Performing an Export	417
Exports: A Deeper Look	419
Exports and Transactionality	420
Exports and SequenceFiles	421
Further Reading	422
<b>16. Pig.....</b>	<b>423</b>
Installing and Running Pig	424
Execution Types	424
Running Pig Programs	426
Grunt	426
Pig Latin Editors	427
An Example	427
Generating Examples	429
Comparison with Databases	430
Pig Latin	432
Structure	432
Statements	433
Expressions	438
Types	439
Schemas	441
Functions	445
Macros	447
User-Defined Functions	448
A Filter UDF	448
An Eval UDF	452
A Load UDF	453
Data Processing Operators	457
Loading and Storing Data	457
Filtering Data	457
Grouping and Joining Data	459
Sorting Data	465
Combining and Splitting Data	466
Pig in Practice	467
Parallelism	467
Anonymous Relations	467
Parameter Substitution	468
Further Reading	469
<b>17. Hive.....</b>	<b>471</b>
Installing Hive	472
The Hive Shell	473

An Example	474
Running Hive	475
Configuring Hive	475
Hive Services	478
The Metastore	480
Comparison with Traditional Databases	482
Schema on Read Versus Schema on Write	482
Updates, Transactions, and Indexes	483
SQL-on-Hadoop Alternatives	484
HiveQL	485
Data Types	486
Operators and Functions	488
Tables	489
Managed Tables and External Tables	490
Partitions and Buckets	491
Storage Formats	496
Importing Data	500
Altering Tables	502
Dropping Tables	502
Querying Data	503
Sorting and Aggregating	503
MapReduce Scripts	503
Joins	505
Subqueries	508
Views	509
User-Defined Functions	510
Writing a UDF	511
Writing a UDAF	513
Further Reading	518
<b>18. Crunch.....</b>	<b>519</b>
An Example	520
The Core Crunch API	523
Primitive Operations	523
Types	528
Sources and Targets	531
Functions	533
Materialization	535
Pipeline Execution	538
Running a Pipeline	538
Stopping a Pipeline	539
Inspecting a Crunch Plan	540

Iterative Algorithms	543
Checkpointing a Pipeline	545
Crunch Libraries	545
Further Reading	548
<b>19. Spark</b> .....	<b>549</b>
Installing Spark	550
An Example	550
Spark Applications, Jobs, Stages, and Tasks	552
A Scala Standalone Application	552
A Java Example	554
A Python Example	555
Resilient Distributed Datasets	556
Creation	556
Transformations and Actions	557
Persistence	560
Serialization	562
Shared Variables	564
Broadcast Variables	564
Accumulators	564
Anatomy of a Spark Job Run	565
Job Submission	565
DAG Construction	566
Task Scheduling	569
Task Execution	570
Executors and Cluster Managers	570
Spark on YARN	571
Further Reading	574
<b>20. HBase</b> .....	<b>575</b>
HBasics	575
Backdrop	576
Concepts	576
Whirlwind Tour of the Data Model	576
Implementation	578
Installation	581
Test Drive	582
Clients	584
Java	584
MapReduce	587
REST and Thrift	589
Building an Online Query Application	589

Schema Design	590
Loading Data	591
Online Queries	594
HBase Versus RDBMS	597
Successful Service	598
HBase	599
Praxis	600
HDFS	600
UI	601
Metrics	601
Counters	601
Further Reading	601
<b>21. ZooKeeper.....</b>	<b>603</b>
Installing and Running ZooKeeper	604
An Example	606
Group Membership in ZooKeeper	606
Creating the Group	607
Joining a Group	609
Listing Members in a Group	610
Deleting a Group	612
The ZooKeeper Service	613
Data Model	614
Operations	616
Implementation	620
Consistency	622
Sessions	624
States	625
Building Applications with ZooKeeper	627
A Configuration Service	627
The Resilient ZooKeeper Application	630
A Lock Service	634
More Distributed Data Structures and Protocols	636
ZooKeeper in Production	637
Resilience and Performance	637
Configuration	639
Further Reading	640

---

## Part V. Case Studies

<b>22. Composable Data at Cerner.....</b>	<b>643</b>
From CPUs to Semantic Integration	643
Enter Apache Crunch	644
Building a Complete Picture	644
Integrating Healthcare Data	647
Composability over Frameworks	650
Moving Forward	651
<b>23. Biological Data Science: Saving Lives with Software.....</b>	<b>653</b>
The Structure of DNA	655
The Genetic Code: Turning DNA Letters into Proteins	656
Thinking of DNA as Source Code	657
The Human Genome Project and Reference Genomes	659
Sequencing and Aligning DNA	660
ADAM, A Scalable Genome Analysis Platform	661
Literate programming with the Avro interface description language (IDL)	662
Column-oriented access with Parquet	663
A simple example: <i>k</i> -mer counting using Spark and ADAM	665
From Personalized Ads to Personalized Medicine	667
Join In	668
<b>24. Cascading.....</b>	<b>669</b>
Fields, Tuples, and Pipes	670
Operations	673
Taps, Schemes, and Flows	675
Cascading in Practice	676
Flexibility	679
Hadoop and Cascading at ShareThis	680
Summary	684
<b>A. Installing Apache Hadoop.....</b>	<b>685</b>
<b>B. Cloudera's Distribution Including Apache Hadoop.....</b>	<b>691</b>
<b>C. Preparing the NCDC Weather Data.....</b>	<b>693</b>
<b>D. The Old and New Java MapReduce APIs.....</b>	<b>697</b>
<b>Index.....</b>	<b>701</b>