The Ultimate Hands-On Hadoop – Tame your Big Data!

### Section 1: Learn all the buzzwords! And install Hadoop.

In this section, we will take a quick tour of all technologies and manage "big data" using Hadoop and related technologies. We will understand the Hadoop overview and history and install it on desktop.

### Video 1 (Free Preview)

[Activity] Introduction, and install Hadoop on your desktop! 17:26

We dive right in and install Hortonworks Sandbox in a virtual machine on PC. We will the download some real movie ratings data, and use Hive to analyse it.

### Video 2 (Free Preview)

Hadoop Overview and History 07:44

We will learn more about Hadoop history, what it is used for and the type of problem it solves.

### Video 3

Overview of Hadoop Ecosystem 16:46

Here, we take a quick tour of all technologies and how all they fit in together. Learn more about the buzzwords used in this course.

### Video 4

Tips for Using This Course 01:09

Learn shortcut tips for successful completion of the course.

### Section 2: Using Hadoop’s Core: HDFs and MapReduce

Use HDFS and MapReduce for storing and analyzing data at scale.

### Video 1

HDFS: What it is, and how it works 13:53

Learn how Hadoop’s Distributed Filesystem allows you to store massive data sets across a cluster of commodity computers, in a reliable and scalable manner.

### Video 2 (Free Preview)

[Activity] Install the MovieLens dataset into HDFS using the Ambari UI 06:20

Import some real movie ratings data into HDFS by just using web-based UI provided by Ambari.

### Video 3

[Activity] Install the MovieLens dataset into HDFS using the command line 07:50

Import data from command prompt and interact with HDFS via and command line interface.

### Video 4

MapReduce: What it is, and how it works 10:40

Learn how mappers and reducers provide a clever way to analyse massive distributed datasets quickly and reliable.

### Video 5

How MapReduce distributes processing 12:57

Learn what makes MapReduce so powerful, by horizontally scaling across a cluster of computers.

### Video 6

MapReduce example: Break down movie ratings by rating score 11:35

Simple example on counting how many of each rating type exists in our movie ratings data.

### Video 7

[Activity] Installing Python, MRJob, and nano 07:19

Set up a quick and easy way to get started with MapReduce is by using Python's MRJob package, which lets you use MapReduce's streaming feature to write MapReduce code in Python instead of Java.

### Video 8 (Free Preview)

[Activity] Code up the ratings histogram MapReduce job and run it 07:36

We learn to code for building a breakdown of movie ratings, and actually run it on our system.

### Video 9

[Exercise] Rank Movies by their popularity 07:06

 See if you can write your own MapReduce script that sorts movies by how many ratings they received.

### Video 10

[Activity] Check your results against mine! 08:23

 We'll change our script to count movies instead of ratings, and then review and run my solution for sorting by rating count.

### Section 3: Programming Hadoop with Pig

In this section, we introduce Ambari and Pig and Use Pig and Spark to create scripts to process data on a Hadoop cluster in more complex ways.

### Video 1

Introducing Ambari 09:49

We can use Ambari- Hortonworks’ web-based UI as an easy way to experiment with Pig.

### Video 2

Introducing Pig 06:25

An overview of what Pig is used for, who it's for, and how it works.

### Video 3

Example: Find the oldest movie with 5-star rating using Pig 15:07

Learn to use Pig to script a chain of queries on MovieLens to solve a more complex problem.

### Video 4(Free Preview)

[Activity] Find old 5-star movies with Pig 09:40

Run program on Hadoop sandbox, find some good, old movies.

### Video 5

More Pig Latin 07:34

Let us look at what else Pig Latin can do.

### Video 6

[Exercise] Find the most-rated one-star movie 01:56

Write your own Pig script that finds the most popular really bad movie.

### Video 7

Pig Challenge: Compare Your Results to Mine! 05:37

Compare code results of yours with authors.

### Section 4: Programming Hadoop with Spark

In this section, we work with spark and datasets. RDD is core building block of spark. Datasets can make spark scripts much faster to write.

### Video 1

Why Spark? 10:06

Learn how Sparks efficiency and versatility make Apache Spark one of the hottest Hadoop-related technologies right now, and how it achieves this under the hood.

### Video 2

The Resilient Distributed Datasets(RDD) 10:13

Learn how they are used and the functions available in RDD.

### Video 3

[Activity] Find the movie with the lowest average rating - with RDD's 15:33

 Let's write a Spark script to find the movie with the lowest average rating using RDD.

### Video 4(Free Preview)

Datasets and Spark 2.0 06:28

Learn how Datasets can make your Spark scripts even faster and easier to write.

### Video 5

[Activity] Find the movie with the lowest average rating - with DataFrames 10:00

Revisit previous problems using DataFrames.

### Video 6(Free Preview)

[Activity] Movie recommendations with MLLib 12:16

We'll use Spark's machine learning library to produce movie recommendations using the ALS algorithm.

### Video 7

[Exercise] Filter the lowest-rated movies by number of ratings 02:51

We filter the results of movies with a given number of ratings.

### Video 8

[Activity] Check your results against mine! 06:40

Review the solution and run the resulting scripts.

### Section 5: Using relational data stores with Hadoop

In this section we introduce you to Apache Hive, and learn how it works. We also integrate MySQL with Hadoop.

### Video 1

What is Hive? 06:31

We introduce you to Apache Hive and how it enables relational queries on HDFS-hosted data.

### Video 2

[Activity] Use Hive to find the most popular movie 10:45

We'll import the MovieLens data set into Hive using the Ambari UI, and run a simple query to find the most popular movies.

### Video 3 (Free Preview)

How Hive Works? 09:10

Learn how hive works under the hood while using SQL commands.

### Video 4

[Exercise] Use Hive to find the movie with the highest average rating 01:55

We use the same HIVE database to find the best-rated movie.

### Video 5

Compare your solution to mine. 04:10

Here, we compare solution to authors.

### Video 6

Integrating MySQL with Hadoop 08:00

A quick overview of MySQL and how it might fit into our Hadoop-based work.

### Video 7

[Activity] Install MySQL and import our movie data 07:35

Let import the MovieLens data set into MySQL, and run a query.

### Video 8

[Activity] Use Sqoop to import data from MySQL to HFDS/Hive 07:31

Learn how Sqoop works as a way to transfer data from an existing RDBMS like MySQL into Hadoop.

### Video 9

[Activity] Use Sqoop to export data from Hadoop to MySQL 07:16

Let's build a new table with Hive and export it back into MySQL.

### Section 6: Using non-relational data stores with Hadoop

Here, we analyze non-relational data using HBase, Cassandra, and MongoDB.

### Video 1

Why NoSQL? 13:54

Learn why NoSQL databases are important for efficiently and scalably vending your data.

### Video 2

What is HBase 12:55

Learn what is HBase used for and how it works. It is a NoSQL columnar data store.

### Video 3

[Activity] Import movie ratings into HBase 13:28

We use RESTful service interface to import our movie ratings into HBase.

### Video 4

[Activity] Use HBase with Pig to import data at scale. 11:19

We’ll see how HBase can integrate with Pig to store big data into HBase.

### Video 5

Cassandra Overview 14:50

Cassandra is a popular NoSQL database that is appropriate for vending data at massive scale outside of Hadoop.

### Video 6

[Activity] Installing Cassandra 11:43

We need to install Cassandra since it is not part of Hortonworks.

### Video 7

[Activity] Write Spark output into Cassandra 11:00

We'll modify our HBase example to write results into a Cassandra database.

### Video 8

MongoDB overview 16:54

A popular alternative to Cassandra has different features.

### Video 9

[Activity] Install MongoDB, and integrate Spark with MongoDB 12:44

We'll install MongoDB on our virtual machine using Ambari. Then, we'll study and run a script to load up a Spark DataFrame of user data, store it into MongoDB.

### Video 10

[Activity] Using the MongoDB shell 07:48

We will setup an index by using MongoDB’s command line interface.

### Video 11(Free Preview)

Choosing a database technology 15:59

We take an inform decision by analysing various attributes of databases.

### Video 12

[Exercise] Choose a database for a given problem 05:00

Compare your decision of choosing a database with that of author.

### Section 7: Querying Your Data Interactively

Query data interactively with Drill, Phoenix, and Presto

### Video 1

Overview of Drill 07:55

We familiarize with Drill and what problems does it solve.

### Video 2

[Activity] Setting up Drill 11:19

We install Drill with help from this section.

### Video 3(Free Preview)

[Activity] Querying across multiple databases with Drill 07:07

We'll use Drill to execute a query that spans data on MongoDB and Hive at the same time.

### Video 4

Overview of Phoenix 08:55

We familiarize with Phoenix.

### Video 5

[Activity] Install Phoenix and query HBase with it 07:08

We will work with Phoenix and use it to query our HBase Database.

### Video 6

[Activity] Integrate Phoenix with Pig 11:45

We'll use Phoenix with Pig to store and load MovieLens user’s data, and accelerate queries on it.

### Video 7

Overview of Presto 06:39

We familiarize with Presto and compare with Drill and Phoenix.

### Video 8

[Activity] Install Presto, and query Hive with it. 12:26

We'll install Presto, and issue some queries on Hive through it.

### Video 9(Free Preview)

[Activity] Query both Cassandra and Hive using Presto 09:01

Do a JOIN query that spans both data in Cassandra and Hive.

### Section 8: Managing your Cluster

Here we understand how Hadoop clusters are managed by YARN, Tez, Mesos, Zookeeper, Zeppelin, Hue, and Oozie.

### Video 1(Free Preview)

YARN Explained 10:01

Learn how YARN works in more depth as it controls and allocates the resources of your Hadoop cluster.

### Video 2

Tez explained 04:56

Learn how Tez also uses Directed Acyclic Graphs to optimize tasks on your cluster.

### Video 3

[Activity] Use Hive on Tez and measure the performance benefit 08:35

Here, we will execute Hive query with and without it.

### Video 4

Mesos explained 07:13

Learn how Mesos differs from alternative cluster manager to Hadoop YARN.

### Video 5

ZooKeeper explained 13:10

Learn how it works, and what systems depend on Zookeeper for reliable operation.

### Video 6

[Activity] Simulating a failing master with ZooKeeper 06:47

We will use ZooKeeper's command line interface to explore how it works.

### Video 7

Oozie explained 11:56

Oozie allows you to set up complex workflows on your cluster using multiple technologies, and schedule them. Let's look at some examples of how it works.

### Video 8

[Activity] Set up a simple Oozie workflow 16:39

We'll use Oozie to import movie data into HDFS from MySQL using Sqoop, then analyze that data using Hive.

### Video 9

Zeppelin overview 05:01

Apache Zeppelin provides a notebook-based environment for importing, transforming, and analyzing your data.

### Video 10

[Activity] Use Zeppelin to analyze movie ratings, part 1 12:28

We'll set up a Zeppelin notebook to load movie ratings and titles into Spark dataframes, and interactively query and visualize them.

### Video 11

[Activity] Use Zeppelin to analyze movie ratings, part 2 09:46

We perform the above process again, in this section.

### Video 12

Hue Overview 08:07

Let’s see what Hue offers and how is it different than Ambari.

### Video 13

Other technologies worth mentioning 04:35

We familiarize ourselves with Chukwa and Ganglia.

### Section 9: Feeding Data to your Cluster

In this section, we publish data to your Hadoop Cluster using Kafka, Sqoop and Flume.

### Video 1

Kafka explained 09:48

Learn how Kafka provides a scalable, reliable means for collecting data across a cluster of computers and broadcasting it for further processing.

### Video 2

[Activity] Setting up Kafka, and publishing some data. 07:24

We set kafka up to publish and consume some data from a new topic.

### Video 3

[Activity] Publishing web logs with Kafka 10:21

We use Kafka connector to monitor Apache log files.

### Video 4

Flume explained 10:16

Learn about sinks and Flume's architecture, and how it differs from Kafka.

### Video 5

[Activity] Set up Flume and publish logs with it 07:46

We'll connect a source listening to a telnet connection to a sink that just logs information received.

### Video 6(Free Preview)

[Activity] Set up Flume to monitor a directory and store its data in HDFS 09:12

Here, we'll configure Flume to monitor a directory on our local file system for new files, and publish their data into HDFS, organized by the time the data was received.

### Section 10: Analysing Streams of Data

In this section, introduce yourself to spark streaming, apache storm and flink.

### Video 1

Spark Streaming: Introduction 14:27

Learn how Spark streaming allows you to write "continuous applications" that process micro-batches of information in real time.

### Video 2

[Activity] Analyze web logs published with Flume using Spark streaming 14:20

We'll write and run a Spark Streaming application that analyzes web logs as they are streamed in from Flume.

### Video 3

[Exercise] Monitor Flume-published logs for errors in real time 02:02

Let’s look for status codes in the web log and aggregate how often different status codes appear.

### Video 4

Exercise solution: Aggregating HTTP access codes with Spark Streaming 04:24

Let’s review author’s solution to the previous exercise.

### Video 5

Apache Storm: Introduction 09:27

Learn more about storm streaming, a better solution to Spark streaming.

### Video 6

[Activity] Count words with Storm 14:35

We'll walk through, and run, the word count topology sample included with Storm.

### Video 7(Free Preview)

Flink: An Overview 06:53

Let’s learn the difference between Apache Flink and Storm.

### Video 8

[Activity] Counting words with Flink 10:20

Let's install Flink.

### Section 11: Designing Real-World Systems

For designing real world system, we need briefly cover systems we may encounter. We also go through some sample examples.

### Video 1

The Best of the Rest 09:24

We familiarize with other systems like Impala, NiFi, Falcon, Accumulo, AWS, Kinesis, Redis, Ignite, Elasticsearch, and Slider.

### Video 2

Review: How the pieces fit together 06:29

Learn how they are categorized and fit together.

### Video 3

Understanding your requirements 08:02

We need to ask the right questions about performance, reliability, and complexity.

### Video 4(Free Preview)

Sample Application: consume web server logs and keep tracks of top-sellers 10:06

Learn the technologies needed and how they will fit together for this sample application.

### Video 5

Sample application: serving movie recommendations to a website 11:18

Get familiarize with the tools needed for this sample application.

### Video 6

[Exercise] Design a system to report web sessions per day 02:52

Let’s learn to handle web analytics data for really large websites.

### Video 7

Exercise solution: Design a system to count daily sessions 04:24

Compare your system design with that of author.

### Section 12: Learning More

Learn more through books and online resources as promoted by the author.

### Video 1

Books and online resources 05:32

Some of the author’s favourite tools and websites for reference.

### Video 2

Bonus lecture: Discounts on my other big data / data science courses! 01:06

Let to stay in touch, and how you can get discounts on author’s other courses.