

ft

# Data Just Right

## Introduction to Large-Scale Data & Analytics

**Michael Manoochehri**

**WAddison-Wesley**

Upper Saddle River, NJ • Boston • Indianapolis • San Francisco  
New York • Toronto • Montreal • London • Munich • Paris • Madrid  
Capetown • Sydney • Tokyo • Singapore • Mexico City

# Contents

<b>Foreword</b>	<b>xv</b>
<b>Preface</b>	<b>xvii</b>
<b>Acknowledgments</b>	<b>xxv</b>
<b>About the Author</b>	<b>xxvii</b>

## **I Directives in the Big Data Era** ■

<b>1 Four Rules for Data Success</b>	<b>3</b>
When Data Became a BIG Deal	3
Data and the Single Server	4
The Big Data Trade-Off	5
Build Solutions That Scale (Toward Infinity)	6
Build Systems That Can Share Data (On the Internet)	7
Build Solutions, Not Infrastructure	8
Focus on Unlocking Value from Your Data	8
Anatomy of a Big Data Pipeline	9
The Ultimate Database	10
Summary	10

## **II Collecting and Sharing a Lot of Data** 11

<b>2 Hosting and Sharing Terabytes of Raw Data</b>	<b>13</b>
Suffering from Files	14
The Challenges of Sharing Lots of Files	14
Storage: Infrastructure as a Service	15
The Network Is Slow	16
Choosing the Right Data Format	16
XML: Data, Describe Thyself	18
JSON: The Programmer's Choice	18
Character Encoding	19
File Transformations	21
Data in Motion: Data Serialization Formats	21
Apache Thrift and Protocol Buffers	22
Summary	23

<b>3 Building a NoSQL-Based Web App to Collect Crowd-Sourced Data</b>	<b>25</b>
Relational Databases: Command and Control	<b>25</b>
The Relational Database ACID Test	<b>28</b>
Relational Databases versus the Internet	<b>28</b>
CAP Theorem and BASE	<b>30</b>
Nonrelational Database Models	<b>31</b>
Key-Value Database	<b>32</b>
Document Store	<b>33</b>
Leaning toward Write Performance: Redis	<b>35</b>
Sharding across Many Redis Instances	<b>38</b>
Automatic Partitioning with Twemproxy	<b>39</b>
Alternatives to Using Redis	<b>40</b>
NewSQL: The Return of Codd	<b>41</b>
Summary	<b>42</b>
<b>4 Strategies for Dealing with Data Silos</b>	<b>43</b>
A Warehouse Full of Jargon	<b>43</b>
The Problem in Practice	<b>45</b>
Planning for Data Compliance and Security	<b>46</b>
Enter the Data Warehouse	<b>46</b>
Data Warehousing's Magic Words: Extract, Transform, and Load	<b>48</b>
Hadoop: The Elephant in the Warehouse	<b>48</b>
Data Silos Can Be Good	<b>49</b>
Concentrate on the Data Challenge, Not the Technology	<b>50</b>
Empower Employees to Ask Their Own Questions	<b>50</b>
Invest in Technology That Bridges Data Silos	<b>51</b>
Convergence: The End of the Data Silo	<b>51</b>
Will Luhn's Business Intelligence System Become Reality?	<b>52</b>
Summary	<b>53</b>

<b>III Asking Questions about Your Data</b>	<b>55</b>
<b>5 Using Hadoop, Hive, and Shark to Ask Questions about Large Datasets</b>	<b>57</b>
What Is a Data Warehouse?	<b>57</b>
Apache Hive: Interactive Querying for Hadoop	<b>60</b>
Use Cases for Hive	<b>60</b>
Hive in Practice	<b>61</b>
Using Additional Data Sources with Hive	<b>65</b>
Shark: Queries at the Speed of RAM	<b>65</b>
Data Warehousing in the Cloud	66
Summary	<b>67</b>
<b>6 Building a Data Dashboard with Google BigQuery</b>	<b>69</b>
Analytical Databases	<b>69</b>
Dremel: Spreading the Wealth	<b>71</b>
How Dremel and MapReduce Differ	<b>72</b>
BigQuery: Data Analytics as a Service	<b>73</b>
BigQuery's Query Language	<b>74</b>
Building a Custom Big Data Dashboard	<b>75</b>
Authorizing Access to the BigQuery API	<b>76</b>
Running a Query and Retrieving the Result	<b>78</b>
Caching Query Results	<b>79</b>
Adding Visualization	<b>81</b>
The Future of Analytical Query Engines	<b>82</b>
Summary	<b>83</b>
<b>7 Visualization Strategies for Exploring Large Datasets</b>	<b>85</b>
Cautionary Tales: Translating Data into Narrative	86
Human Scale versus Machine Scale	<b>89</b>
Interactivity	<b>89</b>
Building Applications for Data Interactivity	<b>90</b>
Interactive Visualizations with R and ggplot2	<b>90</b>
matplotlib: 2-D Charts with Python	<b>92</b>
D3.js: Interactive Visualizations for the Web	<b>92</b>
Summary	<b>96</b>

## IV Building Data Pipelines 97

### 8 Putting It Together: MapReduce Data Pipelines 99

What Is a Data Pipeline?	99
The Right Tool for the Job	100
Data Pipelines with Hadoop Streaming	101
MapReduce and Data Transformation	101
The Simplest Pipeline: stdin to stdout	102
A One-Step MapReduce Transformation	105
Extracting Relevant Information from Raw NVSS Data: Map Phase	106
Counting Births per Month: The Reducer Phase	107
Testing the MapReduce Pipeline Locally	108
Running Our MapReduce Job on a Hadoop Cluster	109
Managing Complexity: Python MapReduce Frameworks for Hadoop	110
Rewriting Our Hadoop Streaming Example Using mrjob	110
Building a Multistep Pipeline	112
Running mrjob Scripts on Elastic MapReduce	113
Alternative Python-Based MapReduce Frameworks	114
Summary	114

### 9 Building Data Transformation Workflows with Pig and Cascading 117

Large-Scale Data Workflows in Practice	118
It's Complicated: Multistep MapReduce Transformations	118
Apache Pig: "Ixnay on the Omplexitycay"	119
Running Pig Using the Interactive Grunt Shell	120
Filtering and Optimizing Data Workflows	121
Running a Pig Script in Batch Mode	122
Cascading: Building Robust Data-Workflow Applications	122
Thinking in Terms of Sources and Sinks	123

	»	
Building a Cascading Application	<b>124</b>	
Creating a Cascade: A Simple JOIN Example		<b>125</b>
Deploying a Cascading Application on a Hadoop Cluster	<b>127</b>	
When to Choose Pig versus Cascading		<b>128</b>
Summary	<b>128</b>	

## **V Machine Learning for Large Datasets 129**

<b>10 Building a Data Classification System with Mahout</b>	<b>131</b>	
Can Machines Predict the Future?		<b>132</b>
Challenges of Machine Learning		<b>132</b>
Bayesian Classification		<b>133</b>
Clustering		<b>134</b>
Recommendation Engines		<b>135</b>
Apache Mahout: Scalable Machine Learning		<b>136</b>
Using Mahout to Classify Text		<b>137</b>
MLBase: Distributed Machine Learning Framework		<b>139</b>
Summary		<b>140</b>

## **VI Statistical Analysis for Massive Datasets 143**

<b>11 Using R with Large Datasets</b>	<b>145</b>	
Why Statistics Are Sexy		<b>146</b>
Limitations of R for Large Datasets		<b>147</b>
R Data Frames and Matrices		<b>148</b>
Strategies for Dealing with Large Datasets		<b>149</b>
Large Matrix Manipulation: bigmemory and biganalytics		<b>150</b>
ff: Working with Data Frames Larger than Memory		<b>151</b>
biglm: Linear Regression for Large Datasets		<b>152</b>
RHadoop: Accessing Apache Hadoop from R		<b>154</b>
Summary		<b>155</b>

## Contents

### **12 Building Analytics Workflows Using Python and Pandas 157**

The Snakes Are Loose in the Data Zoo	<b>157</b>
Choosing a Language for Statistical Computation	<b>158</b>
Extending Existing Code	<b>159</b>
Tools and Testing	<b>160</b>
Python Libraries for Data Processing	<b>160</b>
NumPy	<b>160</b>
SciPy: Scientific Computing for Python	<b>162</b>
The Pandas Data Analysis Library	<b>163</b>
Building More Complex Workflows	<b>167</b>
Working with Bad or Missing Records	<b>169</b>
iPython: Completing the Scientific Computing Tool Chain	<b>170</b>
Parallelizing iPython Using a Cluster	<b>171</b>
Summary	<b>174</b>

### **VII Looking Ahead 177**

#### **13 When to Build, When to Buy, When to Outsource 179**

Overlapping Solutions	<b>179</b>
Understanding Your Data Problem	<b>181</b>
A Playbook for the Build versus Buy Problem	<b>182</b>
What Have You Already Invested In?	<b>183</b>
Starting Small	<b>183</b>
Planning for Scale	<b>184</b>
My Own Private Data Center	<b>184</b>
Understand the Costs of Open-Source	<b>186</b>
Everything as a Service	<b>187</b>
Summary	<b>187</b>

#### **14 The Future: Trends in Data Technology 189**

Hadoop: The Disruptor and the Disrupted	<b>190</b>
Everything in the Cloud	<b>191</b>
The Rise and Fall of the Data Scientist	<b>193</b>

Convergence: The Ultimate Database	»	195
Convergence of Cultures	196	
Summary	197	
<b>Index</b>	<b>199</b>	