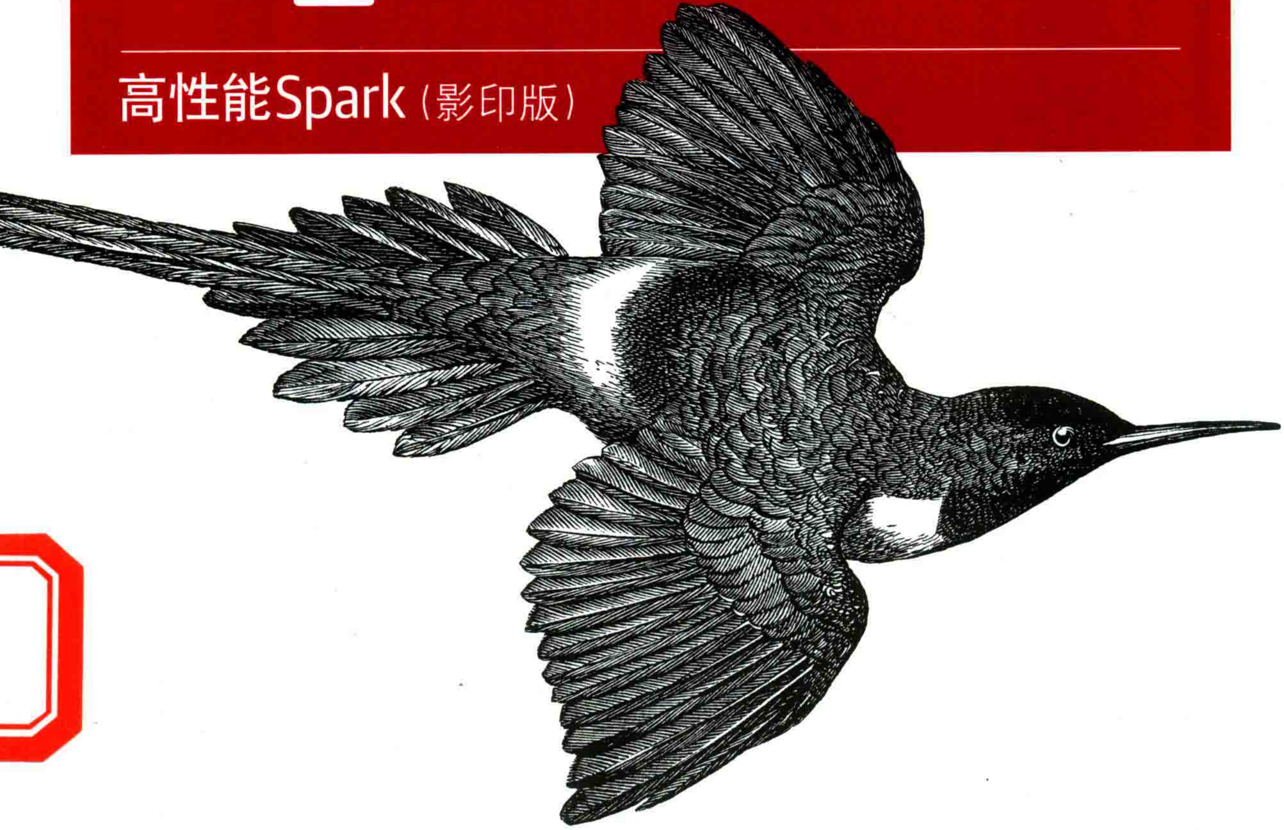
O'REILLY®

High Performance Spark

高性能Spark（影印版）



東南大學出版社

Holden Karau, Rachel Warren 著

高性能Spark（影印版）

High Performance Spark

Holden Karau, Rachel Warren 著

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

O'Reilly Media, Inc.授权东南大学出版社出版

南京 东南大学出版社

图书在版编目(CIP)数据

高性能 Spark:英文/(加)霍尔顿·卡劳(Holden Karau),(加)雷切尔·沃伦(Rachel Warren)著. —影印本. —南京:东南大学出版社,2018.2

书名原文:High Performance Spark

ISBN 978-7-5641-7518-4

Ⅰ.①高… Ⅱ.①霍… ②雷… Ⅲ.①数据处理软件–英文 Ⅳ.①TP274

中国版本图书馆 CIP 数据核字(2017)第 296215 号

图字:10-2017-431 号

高性能 Spark(影印版)

出版发行:东南大学出版社
地　　址:南京四牌楼 2 号　　邮编:210096
出 版 人:江建中
网　　址:http://www.seupress.com
电子邮件:press@seupress.com
印　　刷:常州市武进第三印刷有限公司
开　　本:787 毫米×980 毫米　　16 开本
印　　张:22.5
字　　数:441 千字
版　　次:2018 年 2 月第 1 版
印　　次:2018 年 2 月第 1 次印刷
书　　号:ISBN 978-7-5641-7518-4
定　　价:88.00 元

本社图书若有印装质量问题,请直接与营销部联系。电话(传真):025-83791830

Preface

We wrote this book for data engineers and data scientists who are looking to get the most out of Spark. If you've been working with Spark and invested in Spark but your experience so far has been mired by memory errors and mysterious, intermittent failures, this book is for you. If you have been using Spark for some exploratory work or experimenting with it on the side but have not felt confident enough to put it into production, this book may help. If you are enthusiastic about Spark but have not seen the performance improvements from it that you expected, we hope this book can help. This book is intended for those who have some working knowledge of Spark, and may be difficult to understand for those with little or no experience with Spark or distributed computing. For recommendations of more introductory literature see "Supporting Books and Materials" on page x.

We expect this text will be most useful to those who care about optimizing repeated queries in production, rather than to those who are doing primarily exploratory work. While writing highly performant queries is perhaps more important to the data engineer, writing those queries with Spark, in contrast to other frameworks, requires a good knowledge of the data, usually more intuitive to the data scientist. Thus it may be more useful to a data engineer who may be less experienced with thinking critically about the statistical nature, distribution, and layout of data when considering performance. We hope that this book will help data engineers think more critically about their data as they put pipelines into production. We want to help our readers ask questions such as "How is my data distributed?", "Is it skewed?", "What is the range of values in a column?", and "How do we expect a given value to group?" and then apply the answers to those questions to the logic of their Spark queries.

However, even for data scientists using Spark mostly for exploratory purposes, this book should cultivate some important intuition about writing performant Spark queries, so that as the scale of the exploratory analysis inevitably grows, you may have a better shot of getting something to run the first time. We hope to guide data scientists, even those who are already comfortable thinking about data in a distributed way, to think critically about how their programs are evaluated, empowering them to explore their data more fully, more quickly, and to communicate effectively with anyone helping them put their algorithms into production.

Regardless of your job title, it is likely that the amount of data with which you are working is growing quickly. Your original solutions may need to be scaled, and your old techniques for solving new problems may need to be updated. We hope this book will help you leverage Apache Spark to tackle new problems more easily and old problems more efficiently.

First Edition Notes

You are reading the first edition of *High Performance Spark*, and for that, we thank you! If you find errors, mistakes, or have ideas for ways to improve this book, please reach out to us at *high-performance-spark@googlegroups.com*. If you wish to be included in a "thanks" section in future editions of the book, please include your preferred display name.

Supporting Books and Materials

For data scientists and developers new to Spark, *Learning Spark* by Karau, Konwinski, Wendell, and Zaharia is an excellent introduction,[1] and *Advanced Analytics with Spark* by Sandy Ryza, Uri Laserson, Sean Owen, and Josh Wills is a great book for interested data scientists. For individuals more interested in streaming, the upcoming *Learning Spark Streaming* by François Garillot may also be of use once it is available.

Beyond books, there is also a collection of intro-level Spark training material available. For individuals who prefer video, Paco Nathan has an excellent introduction video series on O'Reilly (*http://shop.oreilly.com/product/0636920036807.do*). Commercially, Databricks (*https://databricks.com/spark/training*) as well as Cloudera (*https://www.cloudera.com/more/training/courses/developer-training-for-spark-and-hadoop.html*) and other Hadoop/Spark vendors offer Spark training. Previous recordings of Spark camps, as well as many other great resources, have been posted on the Apache Spark documentation page (*http://spark.apache.org/documentation.html*).

If you don't have experience with Scala, we do our best to convince you to pick up Scala in Chapter 1, and if you are interested in learning, *Programming Scala*, 2nd Edition, by Dean Wampler and Alex Payne is a good introduction.[2]

1 Though we may be biased.

2 Although it's important to note that some of the practices suggested in this book are not common practice in Spark code.

Conventions Used in This Book

The following typographical conventions are used in this book:

Italic
: Indicates new terms, URLs, email addresses, filenames, and file extensions.

`Constant width`
: Used for program listings, as well as within paragraphs to refer to program elements such as variable or function names, databases, data types, environment variables, statements, and keywords.

`Constant width bold`
: Shows commands or other text that should be typed literally by the user.

`Constant width italic`
: Shows text that should be replaced with user-supplied values or by values determined by context.

This element signifies a tip or suggestion.

This element signifies a general note.

This element indicates a warning or caution.

Examples prefixed with "Evil" depend heavily on Apache Spark internals, and will likely break in future minor releases of Apache Spark. You've been warned—but we totally understand you aren't going to pay much attention to that because neither would we.

Using Code Examples

Supplemental material (code examples, exercises, etc.) is available for download from the High Performance Spark GitHub repository (*https://github.com/high-performance-spark/high-performance-spark-examples*) and some of the testing code is available at the "Spark Testing Base" GitHub repository (*https://github.com/holdenk/spark-testing-base*) and the Spark Validator repo (*https://github.com/holdenk/spark-validator*). Structured Streaming machine learning examples, which are generally in the "evil" category discussed under "Conventions Used in This Book" on page xi, are available at *https://github.com/holdenk/spark-structured-streaming-ml.*

This book is here to help you get your job done. In general, if example code is offered with this book, you may use it in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing a CD-ROM of examples from O'Reilly books does require permission. Answering a question by citing this book and quoting example code does not require permission. The code is also available under an Apache 2 License. Incorporating a significant amount of example code from this book into your product's documentation may require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: "*High Performance Spark* by Holden Karau and Rachel Warren (O'Reilly). Copyright 2017 Holden Karau, Rachel Warren, 978-1-491-94320-5."

If you feel your use of code examples falls outside fair use or the permission given above, feel free to contact us at *permissions@oreilly.com*.

O'Reilly Safari

Safari (formerly Safari Books Online) is a membership-based training and reference platform for enterprise, government, educators, and individuals.

Members have access to thousands of books, training videos, Learning Paths, interactive tutorials, and curated playlists from over 250 publishers, including O'Reilly Media, Harvard Business Review, Prentice Hall Professional, Addison-Wesley Professional, Microsoft Press, Sams, Que, Peachpit Press, Adobe, Focal Press, Cisco Press, John Wiley & Sons, Syngress, Morgan Kaufmann, IBM Redbooks, Packt, Adobe Press, FT Press, Apress, Manning, New Riders, McGraw-Hill, Jones & Bartlett, and Course Technology, among others.

For more information, please visit *http://oreilly.com/safari*.

How to Contact the Authors

For feedback, email us at *high-performance-spark@googlegroups.com*. For random ramblings, occasionally about Spark, follow us on twitter:

Holden: *http://twitter.com/holdenkarau*

Rachel: *https://twitter.com/warre_n_peace*

How to Contact Us

Please address comments and questions concerning this book to the publisher:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (in the United States or Canada)
707-829-0515 (international or local)
707-829-0104 (fax)

To comment or ask technical questions about this book, send email to *bookquestions@oreilly.com*.

For more information about our books, courses, conferences, and news, see our website at *http://www.oreilly.com*.

Find us on Facebook: *http://facebook.com/oreilly*

Follow us on Twitter: *http://twitter.com/oreillymedia*

Watch us on YouTube: *http://www.youtube.com/oreillymedia*

Acknowledgments

The authors would like to acknowledge everyone who has helped with comments and suggestions on early drafts of our work. Special thanks to Anya Bida, Jakob Odersky, and Katharine Kearnan for reviewing early drafts and diagrams. We'd like to thank Mahmoud Hanafy for reviewing and improving the sample code as well as early drafts. We'd also like to thank Michael Armbrust for reviewing and providing feedback on early drafts of the SQL chapter. Justin Pihony has been one of the most active early readers, suggesting fixes in every respect (language, formatting, etc.).

Thanks to all of the readers of our O'Reilly early release who have provided feedback on various errata, including Kanak Kshetri and Rubén Berenguel.

We'd also like to thank our dedicated (official) technical reviewers, Neelesh Srinivas Salian and Denny Lee, who read through every page providing detailed feedback and helped us decide what content belonged where.

Finally, thank you to our respective employers for being understanding as we've worked on this book. Especially Lawrence Spracklen who insisted we mention him here :p.

Table of Contents

CHAPTER 1

Introduction to High Performance Spark

This chapter provides an overview of what we hope you will be able to learn from this book and does its best to convince you to learn Scala. Feel free to skip ahead to Chapter 2 if you already know what you're looking for and use Scala (or have your heart set on another language).

What Is Spark and Why Performance Matters

Apache Spark is a high-performance, general-purpose distributed computing system that has become the most active Apache open source project, with more than 1,000 active contributors.[1] Spark enables us to process large quantities of data, beyond what can fit on a single machine, with a high-level, relatively easy-to-use API. Spark's design and interface are unique, and it is one of the fastest systems of its kind. Uniquely, Spark allows us to write the logic of data transformations and machine learning algorithms in a way that is parallelizable, but relatively system agnostic. So it is often possible to write computations that are fast for distributed storage systems of varying kind and size.

However, despite its many advantages and the excitement around Spark, the simplest implementation of many common data science routines in Spark can be much slower and much less robust than the best version. Since the computations we are concerned with may involve data at a very large scale, the time and resources that gains from tuning code for performance are enormous. Performance does not just mean run faster; often at this scale it means getting something to run at all. It is possible to construct a Spark query that fails on gigabytes of data but, when refactored and adjusted with an eye toward the structure of the data and the requirements of the cluster,

1 From *http://spark.apache.org/*.

succeeds on the same system with terabytes of data. In the authors' experience writing production Spark code, we have seen the same tasks, run on the same clusters, run 100× faster using some of the optimizations discussed in this book. In terms of data processing, time is money, and we hope this book pays for itself through a reduction in data infrastructure costs and developer hours.

Not all of these techniques are applicable to every use case. Especially because Spark is highly configurable and is exposed at a higher level than other computational frameworks of comparable power, we can reap tremendous benefits just by becoming more attuned to the shape and structure of our data. Some techniques can work well on certain data sizes or even certain key distributions, but not all. The simplest example of this can be how for many problems, using `groupByKey` in Spark can very easily cause the dreaded out-of-memory exceptions, but for data with few duplicates this operation can be just as quick as the alternatives that we will present. Learning to understand your particular use case and system and how Spark will interact with it is a must to solve the most complex data science problems with Spark.

What You Can Expect to Get from This Book

Our hope is that this book will help you take your Spark queries and make them faster, able to handle larger data sizes, and use fewer resources. This book covers a broad range of tools and scenarios. You will likely pick up some techniques that might not apply to the problems you are working with, but that might apply to a problem in the future and may help shape your understanding of Spark more generally. The chapters in this book are written with enough context to allow the book to be used as a reference; however, the structure of this book is intentional and reading the sections in order should give you not only a few scattered tips, but a comprehensive understanding of Apache Spark and how to make it sing.

It's equally important to point out what you will likely not get from this book. This book is not intended to be an introduction to Spark or Scala; several other books and video series are available to get you started. The authors may be a little biased in this regard, but we think *Learning Spark* by Karau, Konwinski, Wendell, and Zaharia as well as Paco Nathan's introduction video series (*http://shop.oreilly.com/product/0636920036807.do*) are excellent options for Spark beginners. While this book is focused on performance, it is not an operations book, so topics like setting up a cluster and multitenancy are not covered. We are assuming that you already have a way to use Spark in your system, so we won't provide much assistance in making higher-level architecture decisions. There are future books in the works, by other authors, on the topic of Spark operations that may be done by the time you are reading this one. If operations are your show, or if there isn't anyone responsible for operations in your organization, we hope those books can help you.