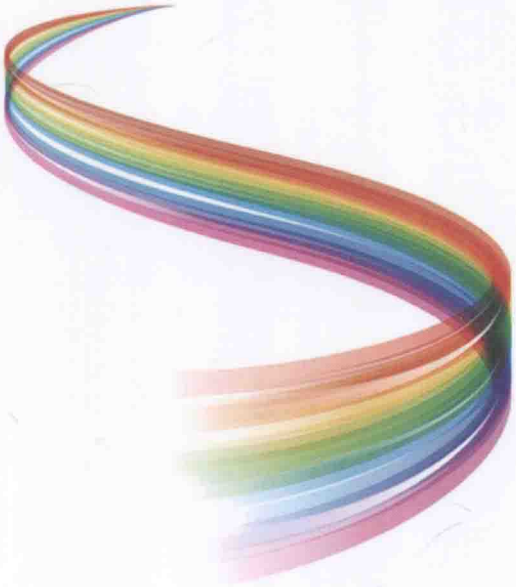
HZ BOOKS
华章IT

阿里巴巴集团专家鼎力推荐，阿里巴巴资深Java开发和大数据专家撰写

结合大量图和示例，对Spark的架构、部署模式和工作模块的设计理念、实现源码与使用技巧进行了深入的剖析与解读

大数据
技术丛书



Spark Internals
Core Design and Source Code Analysis

深入理解Spark

核心思想与源码分析

耿嘉安◎著

机械工业出版社
China Machine Press

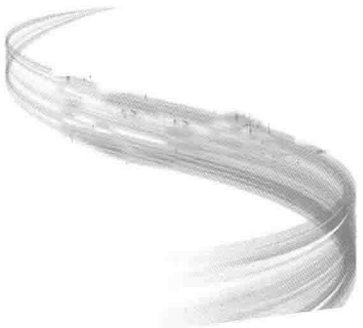
大数据技术丛书

Spark Internals
Core Design and Source Code Analysis

深入理解Spark

核心思想与源码分析

耿嘉安◎著



机械工业出版社
China Machine Press

图书在版编目（CIP）数据

深入理解 Spark：核心思想与源码分析 / 耿嘉安著 . —北京：机械工业出版社，2015.12
（大数据技术丛书）

ISBN 978-7-111-52234-8

I. 深… II. 耿… III. 数据处理软件 IV. TP274

中国版本图书馆 CIP 数据核字（2015）第 280808 号

深入理解 Spark：核心思想与源码分析

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码：100037）
责任编辑：高婧雅　　责任校对：董纪丽
印　　刷：三河市宏图印务有限公司　　版　　次：2016 年 1 月第 1 版第 1 次印刷
开　　本：186mm × 240mm　1/16　　印　　张：30.25
书　　号：ISBN 978-7-111-52234-8　　定　　价：99.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88379426　88361066　　投稿热线：（010）88379604
购书热线：（010）68326294　88379649　68995259　　读者信箱：hzit@hzbook.com

Preface 前　言

为什么写这本书

要回答这个问题，需要从我个人的经历说起。说来惭愧，我第一次接触计算机是在高三。当时跟大家一起去网吧玩 CS，跟身边的同学学怎么“玩”。正是通过这种“玩”的过程，让我了解到计算机并没有那么神秘，它也只是台机器，用起来似乎并不比打开电视机费劲多少。高考填志愿的时候，凭着直觉“糊里糊涂”就选择了计算机专业。等到真正学习计算机课程的时候却又发现，它其实很难！

早在 2004 年，还在学校的我跟很多同学一样，喜欢看 Flash，也喜欢谈论 Flash 甚至做 Flash。感觉 Flash 正如它的名字那样“闪光”。那些年，在学校里，知道 Flash 的人可要比知道 Java 的人多得多，这说明当时的 Flash 十分火热。此外，Oracle 也成为关系型数据库里的领军人物，很多人甚至觉得懂 Oracle 要比懂 Flash、Java 及其他数据库要厉害得多！

2007 年，我刚刚参加工作不久。那时 Struts1、Spring、Hibernate 几乎可以称为那些用 Java 作为开发语言的软件公司的三驾马车。很快，Struts2 替代了 Struts1 的地位，让我第一次意识到 IT 领域的技术更新竟然如此之快！随着很多传统软件公司向互联网公司转型，Hibernate 也难以确保其地位，iBATIS 诞生了！

2010 年，有关 Hadoop 的技术图书涌入中国，当时很多公司用它只是为了数据统计、数据挖掘或者搜索。一开始，人们对于 Hadoop 的认识和使用可能相对有限。大约 2011 年的时候，关于云计算的概念在网上炒得火热，当时依然在做互联网开发的我，对其只是“道听途说”。后来跟同事借了一本有关云计算的书，回家挑着看了一些内容，也没什么收获，怅然若失！ 20 世纪 60 年代，美国的军用网络作为互联网的雏形，很多内容已经与云计算中的某些说法类似。到 20 世纪 80 年代，互联网就已经启用了云计算，如今为什么又要重提这样的概念？这个问题我可能回答不了，还是交给历史吧。

2012 年，国内又呈现出大数据热的态势。从国家到媒体、教育、IT 等几乎所有领域，人人都在谈大数据。我的亲戚朋友中，无论老师、销售人员，还是工程师们都可以针对大数据谈谈自己的看法。我也找来一些 Hadoop 的书籍进行学习，希望能在其中探索到大数据的奥妙。

有幸在工作过程中接触到阿里的开放数据处理服务（open data processing service，ODPS），并且基于ODPS与其他小伙伴一起构建阿里的大数据商业解决方案——御膳房。去杭州出差的过程中，有幸认识和仲，跟他学习了阿里的实时多维分析平台——Garuda和实时计算平台——Galaxy的部分知识。和仲推荐我阅读Spark的源码，这样会对实时计算及流式计算有更深入的了解。2015年春节期间，自己初次上网查阅Spark的相关资料学习，开始研究Spark源码。还记得那时只是出于对大数据的热爱，想使自己在这方面的技术能力有所提升。

从阅读Hibernate源码开始，到后来阅读Tomcat、Spring的源码，我也在从学习源码的过程中成长，我对源码阅读也越来越感兴趣。随着对Spark源码阅读的深入，发现很多内容从网上找不到答案，只能自己“硬啃”了。随着自己的积累越来越多，突然有一天发现，我所总结的这些内容好像可以写成一本书了！从闪光（Flash）到火花（Spark），足足有11个年头了。无论是Flash、Java，还是Spring、iBATIS，我一直扮演着一个追随者，我接受这些书籍的洗礼，从未给予。如今我也是Spark的追随者，不同的是，我不再只想简单攫取，还要给予。

最后还想说一下，2016年是我从事IT工作的第10个年头，此书特别作为送给自己的10周年礼物。

本书特色

- 按照源码分析的习惯设计，从脚本分析到初始化再到核心内容，最后介绍Spark的扩展内容。整个过程遵循由浅入深、由深到广的基本思路。
- 本书涉及的所有内容都有相应的例子，以便于读者对源码的深入研究。
- 本书尽可能用图来展示原理，加速读者对内容的掌握。
- 本书讲解的很多实现及原理都值得借鉴，能帮助读者提升架构设计、程序设计等方面的能力。
- 本书尽可能保留较多的源码，以便于初学者能够在像地铁、公交这样的地方，也能轻松阅读。

读者对象

源码阅读是一项苦差事，人力和时间成本都很高，尤其是对于Spark陌生或者刚刚开始学习的人来说，难度可想而知。本书尽可能保留源码，使得分析过程不至于产生跳跃感，目的是降低大多数人的学习门槛。如果你是从事IT工作1～3年的新人或者是希望学习Spark核心知识的人，本书非常适合你。如果你已经对Spark有所了解或者已经在使用它，还想进一步提高自己，那么本书更适合你。

如果你是一个开发新手，对Java、Linux等基础知识不是很了解，那么本书可能不太适合你。如果你已经对Spark有深入的研究，本书也许可以作为你的参考资料。

总体说来，本书适合以下人群：

- 想要使用 Spark，但对 Spark 实现原理不了解，不知道怎么学习的人；
- 大数据技术爱好者，以及想深入了解 Spark 技术内部实现细节的人；
- 有一定 Spark 使用基础，但是不了解 Spark 技术内部实现细节的人；
- 对性能优化和部署方案感兴趣的大型互联网工程师和架构师；
- 开源代码爱好者。喜欢研究源码的同学可以从本书学到一些阅读源码的方式与方法。

本书不会教你如何开发 Spark 应用程序，只是用一些经典例子演示。本书简单介绍 Hadoop MapReduce、Hadoop YARN、Mesos、Tachyon、ZooKeeper、HDFS、Amazon S3，但不会过多介绍这些框架的使用，因为市场上已经有丰富的这类书籍供读者挑选。本书也不会过多介绍 Scala、Java、Shell 的语法，读者可以在市场上选择适合自己的书籍阅读。

如何阅读本书

本书分为三大部分（不包括附录）：

准备篇（第 1 ～ 2 章），简单介绍了 Spark 的环境搭建和基本原理，帮助读者了解一些背景知识。

核心设计篇（第 3 ～ 7 章），着重讲解 SparkContext 的初始化、存储体系、任务提交与执行、计算引擎及部署模式的原理和源码分析。

扩展篇（第 8 ～ 11 章），主要讲解基于 Spark 核心的各种扩展及应用，包括：SQL 处理引擎、Hive 处理、流式计算框架 Spark Streaming、图计算框架 GraphX、机器学习库 MLlib 等内容。

本书最后还添加了几个附录，包括：附录 A 介绍的 Spark 中最常用的工具类 Utils；附录 B 是 Akka 的简介与工具类 AkkaUtils 的介绍；附录 C 为 Jetty 的简介和工具类 JettyUtils 的介绍；附录 D 为 Metrics 库的简介和测量容器 MetricRegistry 的介绍；附录 E 演示了 Hadoop1.0 版本中的 word count 例子；附录 F 介绍了工具类 CommandUtils 的常用方法；附录 G 是关于 Netty 的简介和工具类 NettyUtils 的介绍；附录 H 列举了笔者编译 Spark 源码时遇到的问题及解决办法。

为了降低读者阅读理解 Spark 源码的门槛，本书尽可能保留源码实现，希望读者能够怀着一颗好奇的心，Spark 当前很火热，其版本更新也很快，本书以 Spark 1.2.3 版本为主，有兴趣的读者也可按照本书的方式，阅读 Spark 的最新源码。

勘误和支持

本书内容很多，限于笔者水平有限，书中内容难免有错误之处。在本书出版后的任何时间，如果你对本书有任何问题或者意见，都可以通过邮箱 beliefer@163.com 或博客 http://www.cnblogs.com/jiaan-geng/ 联系我，说出你的建议或者想法，希望与大家共同进步。

致谢

感谢苍天，让我生活在这样一个时代，能接触互联网和大数据；感谢父母，这么多年来，在学习、工作及生活上的帮助与支持；感谢妻子在生活中的照顾和谦让。

感谢杨福川和高婧雅给予本书出版的大力支持与帮助。

感谢冰夷老大和王贲老大让我有幸加入阿里，接触大数据应用；感谢和仲对 Galaxy 和 Garuda 耐心细致的讲解以及对 Spark 的推荐；感谢张中在百忙之中给本书写评语；感谢周亮、澄苍、民瞻、石申、清无、少侠、征宇、三步、谢衣、晓五、法星、曦轩、九翎、峰阅、丁卯、阿末、紫丞、海炎、涵康、云飏、孟天、零一、六仙、大知、井凡、隆君、太奇、晨炫、既望、宝升、都灵、鬼厉、归钟、梓撤、昊苍、水村、惜冰、惜陌、元乾等同仁在工作上的支持和帮助。

耿嘉安　于北京

Contents 目　录

前言

准　备　篇

第 1 章　环境准备 ····· 2

1.1　运行环境准备 ····· 2

1.1.1　安装 JDK ····· 3

1.1.2　安装 Scala ····· 3

1.1.3　安装 Spark ····· 4

1.2　Spark 初体验 ····· 4

1.2.1　运行 spark-shell ····· 4

1.2.2　执行 word count ····· 5

1.2.3　剖析 spark-shell ····· 7

1.3　阅读环境准备 ····· 11

1.4　Spark 源码编译与调试 ····· 13

1.5　小结 ····· 17

第 2 章　Spark 设计理念与基本架构 ····· 18

2.1　初识 Spark ····· 18

2.1.1　Hadoop MRv1 的局限 ····· 18

2.1.2　Spark 使用场景 ····· 20

2.1.3　Spark 的特点 ····· 20

2.2　Spark 基础知识 ····· 20

2.3　Spark 基本设计思想 ····· 22

2.3.1　Spark 模块设计 ····· 22

2.3.2　Spark 模型设计 ····· 24

2.4　Spark 基本架构 ····· 25

2.5　小结 ····· 26

核心设计篇

第 3 章　SparkContext 的初始化 ····· 28

3.1　SparkContext 概述 ····· 28

3.2　创建执行环境 SparkEnv ····· 30

3.2.1　安全管理器 SecurityManager ····· 31

3.2.2　基于 Akka 的分布式消息系统 ActorSystem ····· 31

3.2.3　map 任务输出跟踪器 mapOutputTracker ····· 32

3.2.4　实例化 ShuffleManager ····· 34

3.2.5　shuffle 线程内存管理器 ShuffleMemoryManager ····· 34

3.2.6　块传输服务 BlockTransferService ····· 35

3.2.7　BlockManagerMaster 介绍 ····· 35

3.2.8 创建块管理器 BlockManager……36
3.2.9 创建广播管理器 BroadcastManager……36
3.2.10 创建缓存管理器 CacheManager…37
3.2.11 HTTP 文件服务器 HttpFileServer……37
3.2.12 创建测量系统 MetricsSystem …39
3.2.13 创建 SparkEnv……40
3.3 创建 metadataCleaner……41
3.4 SparkUI 详解 ……42
3.4.1 listenerBus 详解 ……43
3.4.2 构造 JobProgressListener……46
3.4.3 SparkUI 的创建与初始化……47
3.4.4 Spark UI 的页面布局与展示……49
3.4.5 SparkUI 的启动……54
3.5 Hadoop 相关配置及 Executor 环境变量……54
3.5.1 Hadoop 相关配置信息……54
3.5.2 Executor 环境变量……54
3.6 创建任务调度器 TaskScheduler……55
3.6.1 创建 TaskSchedulerImpl ……55
3.6.2 TaskSchedulerImpl 的初始化 ……57
3.7 创建和启动 DAGScheduler……57
3.8 TaskScheduler 的启动……60
3.8.1 创建 LocalActor……60
3.8.2 ExecutorSource 的创建与注册……62
3.8.3 ExecutorActor 的构建与注册……64
3.8.4 Spark 自身 ClassLoader 的创建…64
3.8.5 启动 Executor 的心跳线程……66
3.9 启动测量系统 MetricsSystem……69
3.9.1 注册 Sources……70
3.9.2 注册 Sinks……70
3.9.3 给 Sinks 增加 Jetty 的 ServletContextHandler……71
3.10 创建和启动 ExecutorAllocationManager……72
3.11 ContextCleaner 的创建与启动……73
3.12 Spark 环境更新……74
3.13 创建 DAGSchedulerSource 和 BlockManagerSource……76
3.14 将 SparkContext 标记为激活……77
3.15 小结……78

第 4 章 存储体系……79

4.1 存储体系概述……79
4.1.1 块管理器 BlockManager 的实现…79
4.1.2 Spark 存储体系架构……81
4.2 shuffle 服务与客户端……83
4.2.1 Block 的 RPC 服务 ……84
4.2.2 构造传输上下文 TransportContext……85
4.2.3 RPC 客户端工厂 TransportClientFactory ……86
4.2.4 Netty 服务器 TransportServer……87
4.2.5 获取远程 shuffle 文件……88
4.2.6 上传 shuffle 文件……89
4.3 BlockManagerMaster 对 BlockManager 的管理……90
4.3.1 BlockManagerMasterActor ……90
4.3.2 询问 Driver 并获取回复方法……92
4.3.3 向 BlockManagerMaster 注册 BlockManagerId……93
4.4 磁盘块管理器 DiskBlockManager……94
4.4.1 DiskBlockManager 的构造过程…94

4.4.2 获取磁盘文件方法 getFile ····· 96
4.4.3 创建临时 Block 方法 createTempShuffleBlock ····· 96
4.5 磁盘存储 DiskStore ····· 97
4.5.1 NIO 读取方法 getBytes ····· 97
4.5.2 NIO 写入方法 putBytes ····· 98
4.5.3 数组写入方法 putArray ····· 98
4.5.4 Iterator 写入方法 putIterator ····· 98
4.6 内存存储 MemoryStore ····· 99
4.6.1 数据存储方法 putBytes ····· 101
4.6.2 Iterator 写入方法 putIterator 详解 ····· 101
4.6.3 安全展开方法 unrollSafely ····· 102
4.6.4 确认空闲内存方法 ensureFreeSpace ····· 105
4.6.5 内存写入方法 putArray ····· 107
4.6.6 尝试写入内存方法 tryToPut ····· 108
4.6.7 获取内存数据方法 getBytes ····· 109
4.6.8 获取数据方法 getValues ····· 110
4.7 Tachyon 存储 TachyonStore ····· 110
4.7.1 Tachyon 简介 ····· 111
4.7.2 TachyonStore 的使用 ····· 112
4.7.3 写入 Tachyon 内存的方法 putIntoTachyonStore ····· 113
4.7.4 获取序列化数据方法 getBytes ····· 113
4.8 块管理器 BlockManager ····· 114
4.8.1 移出内存方法 dropFromMemory ····· 114
4.8.2 状态报告方法 reportBlockStatus ····· 116
4.8.3 单对象块写入方法 putSingle ····· 117
4.8.4 序列化字节块写入方法 putBytes ····· 118
4.8.5 数据写入方法 doPut ····· 118
4.8.6 数据块备份方法 replicate ····· 121
4.8.7 创建 DiskBlockObjectWriter 的方法 getDiskWriter ····· 125
4.8.8 获取本地 Block 数据方法 getBlockData ····· 125
4.8.9 获取本地 shuffle 数据方法 doGetLocal ····· 126
4.8.10 获取远程 Block 数据方法 doGetRemote ····· 127
4.8.11 获取 Block 数据方法 get ····· 128
4.8.12 数据流序列化方法 dataSerializeStream ····· 129
4.9 metadataCleaner 和 broadcastCleaner ····· 129
4.10 缓存管理器 CacheManager ····· 130
4.11 压缩算法 ····· 133
4.12 磁盘写入实现 DiskBlockObjectWriter ····· 133
4.13 块索引 shuffle 管理器 IndexShuffleBlockManager ····· 135
4.14 shuffle 内存管理器 ShuffleMemoryManager ····· 137
4.15 小结 ····· 138

第 5 章 任务提交与执行 ····· 139

5.1 任务概述 ····· 139
5.2 广播 Hadoop 的配置信息 ····· 142
5.3 RDD 转换及 DAG 构建 ····· 144
5.3.1 为什么需要 RDD ····· 144
5.3.2 RDD 实现分析 ····· 146
5.4 任务提交 ····· 152

5.4.1 任务提交的准备 …… 152
5.4.2 finalStage 的创建与 Stage 的划分 …… 157
5.4.3 创建 Job …… 163
5.4.4 提交 Stage …… 164
5.4.5 提交 Task …… 165
5.5 执行任务 …… 176
5.5.1 状态更新 …… 176
5.5.2 任务还原 …… 177
5.5.3 任务运行 …… 178
5.6 任务执行后续处理 …… 179
5.6.1 计量统计与执行结果序列化 …… 179
5.6.2 内存回收 …… 180
5.6.3 执行结果处理 …… 181
5.7 小结 …… 187

第 6 章 计算引擎 …… 188

6.1 迭代计算 …… 188
6.2 什么是 shuffle …… 192
6.3 map 端计算结果缓存处理 …… 194
6.3.1 map 端计算结果缓存聚合 …… 195
6.3.2 map 端计算结果简单缓存 …… 200
6.3.3 容量限制 …… 201
6.4 map 端计算结果持久化 …… 204
6.4.1 溢出分区文件 …… 205
6.4.2 排序与分区分组 …… 207
6.4.3 分区索引文件 …… 209
6.5 reduce 端读取中间计算结果 …… 210
6.5.1 获取 map 任务状态 …… 213
6.5.2 划分本地与远程 Block …… 215
6.5.3 获取远程 Block …… 217
6.5.4 获取本地 Block …… 218
6.6 reduce 端计算 …… 219
6.6.1 如何同时处理多个 map 任务的中间结果 …… 219
6.6.2 reduce 端在缓存中对中间计算结果执行聚合和排序 …… 220
6.7 map 端与 reduce 端组合分析 …… 221
6.7.1 在 map 端溢出分区文件，在 reduce 端合并组合 …… 221
6.7.2 在 map 端简单缓存、排序分组，在 reduce 端合并组合 …… 222
6.7.3 在 map 端缓存中聚合、排序分组，在 reduce 端组合 …… 222
6.8 小结 …… 223

第 7 章 部署模式 …… 224

7.1 local 部署模式 …… 225
7.2 local-cluster 部署模式 …… 225
7.2.1 LocalSparkCluster 的启动 …… 226
7.2.2 CoarseGrainedSchedulerBackend 的启动 …… 236
7.2.3 启动 AppClient …… 237
7.2.4 资源调度 …… 242
7.2.5 local-cluster 模式的任务执行 …… 253
7.3 Standalone 部署模式 …… 255
7.3.1 启动 Standalone 模式 …… 255
7.3.2 启动 Master 分析 …… 257
7.3.3 启动 Worker 分析 …… 259
7.3.4 启动 Driver Application 分析 …… 261
7.3.5 Standalone 模式的任务执行 …… 263
7.3.6 资源回收 …… 263
7.4 容错机制 …… 266
7.4.1 Executor 异常退出 …… 266

7.4.2 Worker 异常退出 …… 268
7.4.3 Master 异常退出 …… 269
7.5 其他部署方案 …… 276
7.5.1 YARN …… 277
7.5.2 Mesos …… 280
7.6 小结 …… 282

扩 展 篇

第 8 章 Spark SQL …… 284
8.1 Spark SQL 总体设计 …… 284
8.1.1 传统关系型数据库 SQL 运行原理 …… 285
8.1.2 Spark SQL 运行架构 …… 286
8.2 字典表 Catalog …… 288
8.3 Tree 和 TreeNode …… 289
8.4 词法解析器 Parser 的设计与实现 …… 293
8.4.1 SQL 语句解析的入口 …… 294
8.4.2 建表语句解析器 DDLParser …… 295
8.4.3 SQL 语句解析器 SqlParser …… 296
8.4.4 Spark 代理解析器 SparkSQL-Parser …… 299
8.5 Rule 和 RuleExecutor …… 300
8.6 Analyzer 与 Optimizer 的设计与实现 …… 302
8.6.1 语法分析器 Analyzer …… 304
8.6.2 优化器 Optimizer …… 305
8.7 生成物理执行计划 …… 306
8.8 执行物理执行计划 …… 308
8.9 Hive …… 311
8.9.1 Hive SQL 语法解析器 …… 311
8.9.2 Hive SQL 元数据分析 …… 313
8.9.3 Hive SQL 物理执行计划 …… 314
8.10 应用举例：JavaSparkSQL …… 314
8.11 小结 …… 320

第 9 章 流式计算 …… 321
9.1 Spark Streaming 总体设计 …… 321
9.2 StreamingContext 初始化 …… 323
9.3 输入流接收器规范 Receiver …… 324
9.4 数据流抽象 DStream …… 325
9.4.1 Dstream 的离散化 …… 326
9.4.2 数据源输入流 InputDStream …… 327
9.4.3 Dstream 转换及构建 DStream Graph …… 329
9.5 流式计算执行过程分析 …… 330
9.5.1 流式计算例子 CustomReceiver …… 331
9.5.2 Spark Streaming 执行环境构建 …… 335
9.5.3 任务生成过程 …… 347
9.6 窗口操作 …… 355
9.7 应用举例 …… 357
9.7.1 安装 mosquitto …… 358
9.7.2 启动 mosquitto …… 358
9.7.3 MQTTWordCount …… 359
9.8 小结 …… 361

第 10 章 图计算 …… 362
10.1 Spark GraphX 总体设计 …… 362
10.1.1 图计算模型 …… 363
10.1.2 属性图 …… 365
10.1.3 GraphX 的类继承体系 …… 367
10.2 图操作 …… 368
10.2.1 属性操作 …… 368

10.2.2 结构操作……368
10.2.3 连接操作……369
10.2.4 聚合操作……370
10.3 Pregel API……371
10.3.1 Dijkstra 算法……373
10.3.2 Dijkstra 的实现……376
10.4 Graph 的构建……377
10.4.1 从边的列表加载 Graph……377
10.4.2 在 Graph 中创建图的方法……377
10.5 顶点集合抽象 VertexRDD……378
10.6 边集合抽象 EdgeRDD……379
10.7 图分割……380
10.8 常用算法……382
10.8.1 网页排名……382
10.8.2 Connected Components 的应用……386
10.8.3 三角关系统计……388
10.9 应用举例……390
10.10 小结……391

第 11 章 机器学习……392

11.1 机器学习概论……392
11.2 Spark MLlib 总体设计……394
11.3 数据类型……394
11.3.1 局部向量……394
11.3.2 标记点……395
11.3.3 局部矩阵……396
11.3.4 分布式矩阵……396
11.4 基础统计……398
11.4.1 摘要统计……398
11.4.2 相关统计……399
11.4.3 分层抽样……401
11.4.4 假设检验……401
11.4.5 随机数生成……402
11.5 分类和回归……405
11.5.1 数学公式……405
11.5.2 线性回归……407
11.5.3 分类……407
11.5.4 回归……410
11.6 决策树……411
11.6.1 基本算法……411
11.6.2 使用例子……412
11.7 随机森林……413
11.7.1 基本算法……414
11.7.2 使用例子……414
11.8 梯度提升决策树……415
11.8.1 基本算法……415
11.8.2 使用例子……416
11.9 朴素贝叶斯……416
11.9.1 算法原理……416
11.9.2 使用例子……418
11.10 保序回归……418
11.10.1 算法原理……418
11.10.2 使用例子……419
11.11 协同过滤……419
11.12 聚类……420
11.12.1 K-means……420
11.12.2 高斯混合……422
11.12.3 快速迭代聚类……422
11.12.4 latent Dirichlet allocation……422
11.12.5 流式 K-means……423
11.13 维数减缩……424
11.13.1 奇异值分解……424
11.13.2 主成分分析……425

11.14 特征提取与转型……425
11.14.1 术语频率反转……425
11.14.2 单词向量转换……426
11.14.3 标准尺度……427
11.14.4 正规化尺度……428
11.14.5 卡方特征选择器……428
11.14.6 Hadamard 积……429
11.15 频繁模式挖掘……429
11.16 预言模型标记语言……430
11.17 管道……431
11.17.1 管道工作原理……432
11.17.2 管道 API 介绍……433
11.17.3 交叉验证……435
11.18 小结……436
附录 A Utils……437
附录 B Akka……446
附录 C Jetty……450
附录 D Metrics……453
附录 E Hadoop word count……456
附录 F CommandUtils……458
附录 G Netty……461
附录 H 源码编译错误……465

准 备 篇

- 第 1 章　环境准备
- 第 2 章　Spark 设计理念与基本架构

Chapter 1 第1章

环境准备

凡事豫则立，不豫则废；言前定，则不跲；事前定，则不困。

——《礼记·中庸》

本章导读

在深入了解一个系统的原理、实现细节之前，应当先准备好它的源码编译环境、运行环境。如果能在实际环境安装和运行 Spark，显然能够提升读者对于 Spark 的一些感受，对系统能有个大体的印象，有经验的技术人员甚至能够猜出一些 Spark 采用的编程模型、部署模式等。当你通过一些途径知道了系统的原理之后，难道不会问问自己："这是怎么做到的？"如果只是游走于系统使用、原理了解的层面，是永远不可能真正理解整个系统的。很多 IDE 本身带有调试的功能，每当你阅读源码，陷入重围时，调试能让我们更加理解运行期的系统。如果没有调试功能，不敢想象阅读源码会怎样困难。

本章的主要目的是帮助读者构建源码学习环境，主要包括以下内容：

- 在 Windows 环境下搭建源码阅读环境；
- 在 Linux 环境下搭建基本的执行环境；
- Spark 的基本使用，如 spark-shell。

1.1 运行环境准备

考虑到大部分公司的开发和生成环境都采用 Linux 操作系统，所以笔者选用了 64 位的 Linux。在正式安装 Spark 之前，先要找台好机器。为什么？因为笔者在安装、编译、调试的过程中发现 Spark 非常耗费内存，如果机器配置太低，恐怕会跑不起来。Spark 的开发语言是

Scala，而 Scala 需要运行在 JVM 之上，因而搭建 Spark 的运行环境应该包括 JDK 和 Scala。

1.1.1 安装 JDK

使用命令 getconf LONG_BIT 查看 Linux 机器是 32 位还是 64 位，然后下载相应版本的 JDK 并安装。

下载地址：

http://www.oracle.com/technetwork/java/javase/downloads/index.html

配置环境：

```
cd ~
vim .bash_profile
```

添加如下配置：

```
export JAVA_HOME=/opt/java
export PATH=$PATH:$JAVA_HOME/bin
export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
```

由于笔者的机器上已经安装过 openjdk，所以未使用以上方式，openjdk 的安装命令如下：

```
$ su -c "yum install java-1.7.0-openjdk"
```

安装完毕后，使用 java –version 命令查看，确认安装正常，如图 1-1 所示。

```
[[redacted]@v218142118 ~]$ java -version
java version "1.7.0_51"
Java(TM) SE Runtime Environment (build 1.7.0_51-b13)
OpenJDK ([redacted]) 64-Bit Server VM (build 24.45-b08-internal, mixed mode)
```

图 1-1 查看安装是否正常

1.1.2 安装 Scala

下载地址：http://www.scala-lang.org/download/

选择最新的 Scala 版本下载，下载方法如下：

```
wget http://downloads.typesafe.com/scala/2.11.5/scala-2.11.5.tgz
```

移动到选好的安装目录，例如：

```
mv scala-2.11.5.tgz ~/install/
```

进入安装目录，执行以下命令：

```
chmod 755 scala-2.11.5.tgz
tar -xzvf scala-2.11.5.tgz
```

配置环境：

```
cd ~
vim .bash_profile
```

添加如下配置：