HZ BOOKS
华章科技

多位Spark的贡献者和专家联袂推荐，详细剖析Spark内核各个模块并辅以相应源码解析的著作。以源码为基础，全面分析Spark内核各个模块的设计思想和实现原理，深入理解其内部运作机制乃至实现细节。帮助Spark领域的从业人员全面掌握Spark核心技术，进而在应用开发中做到游刃有余和性能调优时做到有的放矢。

大数据技术丛书

Spark Internals
Design and Implement Principle of Spark Core

Spark技术内幕

深入解析Spark内核架构设计与实现原理

张安站◎著

机械工业出版社
China Machine Press

技术丛书

Spark Internals

Design and Implement Principle of Spark Core

Spark技术内幕

深入解析Spark内核架构设计与实现原理

张安站◎著

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

Spark 技术内幕：深入解析 Spark 内核架构设计与实现原理 / 张安站著 . —北京：机械工业出版社，2015.8
（大数据技术丛书）

ISBN 978-7-111-50964-6

I. S… II. 张… III. 数据处理软件 IV. TP274

中国版本图书馆 CIP 数据核字（2015）第 170549 号

Spark 是不断壮大的大数据分析解决方案家族中备受关注的新成员。它不仅为分布式数据集的处理提供了一个有效框架，而且以高效的方式处理分布式数据集。它支持实时处理、流处理和批处理，提供了统一的解决方案，因此极具竞争力。本书以源码为基础，深入分析 Spark 内核的设计理念和架构实现，系统讲解各个核心模块的实现，为性能调优、二次开发和系统运维提供理论支持，为更好地使用 Spark Streaming、MLlib、Spark SQL 和 GraphX 等奠定基础。

Spark 技术内幕

深入解析 Spark 内核架构设计与实现原理

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：李 艺
责任校对：殷 虹
印 刷：北京瑞德印刷有限公司
版 次：2015 年 9 月第 1 版第 1 次印刷
开 本：186mm × 240mm 1/16
印 张：13.5
书 号：ISBN 978-7-111-50964-6
定 价：49.00 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
购书热线：（010）68326294 88379649 68995259
投稿热线：（010）88379604
读者信箱：hzit@hzbook.com

Forword 序

Apache Spark项目的高速发展超出了很多人的预期。在2009年到2013年，Spark还是UC Berkeley大学AMPLab的一个研究项目，因为其架构设计得简洁和高效，逐渐吸引了工业界和学术界的广泛关注。我还记得2013年2月份在Santa Clara召开的Strata Conference上，虽然是长达一整天的Spark技术培训，大厅里还是人满为患，大家都在认真学习这种新的计算框架，并被其高速的性能所折服。尽管在随后的一年半时间内，主流的Hadoop厂商并没有接受这个新框架：Cloudera在忙着开发自己的Impala引擎，Hortonworks经过评估后认为可以改造Map/Reduce来实现类似Spark的DAG机制，也就是后来的Tez，而MapR还在纠结是否要全力投入Drill项目。但在2014年的夏天，第二次Spark Summit召开时，已经在Spark上积累大量开发者和用户，从互联网到传统行业，甚至是生物神经学家都用Spark来分析脑活动的数据。在这次会议上，大部分的Hadoop厂商以及应用开发商开始接受Spark，并宣布支持Spark作为Hadoop上的另一个计算引擎。自此以后，Spark的被接受程度飞速提高。到2014年10月份，几乎所有的大数据厂商都宣布支持Spark，Spark作者们创办的DataBricks公司也宣布认证了50多个以Spark为基础的应用系统。而到了2015年，大家在谈论的是Spark即将全面替代Hadoop中的Map/Reduce。

星环科技从2013年创业的第一天，就开始改造Spark引擎来开发批处理和交互式分析引擎。今天在星环的全系列产品中，已经几乎看不到Map/Reduce计算框架。星环科技已经证明了在所有Map/Reduce擅长的领域，Spark计算引擎都可以更高效地执行，性能可以提升数倍到数十倍，并且可以7x24稳定运行。这也从侧面证明了Spark引擎的潜力。

本书详细剖析了 Spark 核心引擎的源代码及其工作原理，内容翔实准确，也是我目前看到的一本比较全面解析 Spark Core 的不可多得的好书。特别是有志于 Spark 内核开发的研发人员，仔细阅读本书并研读代码，将起到事半功倍的效果。

孙元浩

星环科技创始人兼 CTO

2015 年 8 月上海

Preface 前　言

诞生于2005年的Hadoop解决了大数据的存储和计算问题，已经成为大数据处理的事实标准。但是，随着数据规模的爆炸式增长和计算场景的丰富细化，使得Hadoop越来越难以满足用户的需求。针对不同的计算场景，开源社区和各大互联网公司也推出了各种大数据分析的平台，旨在满足特定应用场景下的计算需求。但是，众多的平台使得用户不得不为平台开发类似的策略，这增加了运维开发成本。

2009年诞生于AMPLab的Spark，它的设计目标就是为大数据的多种计算场景提供一个通用的计算引擎，同时解决大数据处理的4V难题，即Volume（海量）、Velocity（快速）、Variety（多样）、Value（价值）。正如Spark的核心作者之一的Ion Stoica所说，“The goal is to build a new generation of data analytics software，to be used across academia and industry。”Hadoop之父Doug Cutting也说过，MapReduce引擎将被Spark替代（Use of MapReduce engine for Big Data projects will decline，replaced by Apache Spark）。可以说，Spark自诞生之日起就得到了广泛的关注，也是近年来开源社区最活跃的项目之一。Spark的1.X版本的每次发布，都包含了数百位贡献者的上千次提交。最新的版本是发布于2015年6月11日的1.4.0，是迄今为止Spark最大的一次版本发布，涵盖了210位开发者的贡献。

Spark得到了众多大数据公司的支持，这些公司包括Hortonworks、IBM、Intel、Cloudera、MapR、Pivotal和星环科技；Spark也被百度、阿里、腾讯、京东、携程、优酷土豆等互联网公司应用到多种不同的计算场景中，并且在实际的生产环境中获得了很多收益。当前百度的Spark已应用于凤巢、大搜索、直达号、百度大数据等业务；

阿里利用 GraphX 构建了大规模的图计算和图挖掘系统，实现了很多生产系统的推荐算法；腾讯 Spark 集群达到 8000 台的规模，是当前已知的世界上最大的 Spark 集群。

但是，当前并没有一本系统介绍 Spark 内核实现原理的书，而 Spark 内核是 Spark SQL、Spark Streaming、MLlib、GraphX 等多个模块的基础，这些模块最终的计算执行都是由内核模块完成的。为了在应用开发中做到游刃有余，在性能调优时做到有的放矢，需要了解内核模块的实现原理。笔者从 Spark 发布版本 0.8.1 时开始关注 Spark，并深入学习内核模块的架构实现原理。Spark 在 1.0 发布后，内核模块趋于稳定，虽然内核模块依旧会有不断地改进和完善，但是整体的设计思想和实现方法是不会变的，因此笔者决定为 Spark 社区的用户和关注者写一本书，详细介绍 Spark 内核模块的实现原理。最终，笔者基于 Spark 1.2.0 版本完成了本书。

写作是一件严肃的事情，同样是一份苦差事，尤其是在工作比较忙的时候。本书在半年前就完成了基本的框架，但是随后又对本书进行了多次修改和完善。笔者认为，对一本架构分析的书，一个最基本的要求就是基于源码如实描述系统的实现，能做到这点就是一本及格的书；如果能做到分析这个架构的好坏，指出架构改进的方案，那么就是一本质量比较好的书；如果能高屋建瓴地进行再次抽象，指出类似架构不同实现的优劣，抽象出一些理论，那么这就是一本质量上乘，可以当作教科书的书。我深知自己的能力水平，希望这本书最起码是一本及格的书，即能基于源码如实描述系统的实现，对那些希望深入学习 Spark 架构实现的同仁有所帮助。

目标读者

本书适合大数据领域的架构师、运维人员，尤其是 Spark 领域的从业人员阅读，也适合作为研究生和高年级的本科生大数据领域分布式架构具体实现原理的参考资料。

内容概述

第 1 章介绍了 Spark 的技术背景和特点，给出了架构的整体概述，并简单介绍了 Spark 的生态圈。

第 2 章介绍了 Spark 源码如何获取和学习环境如何搭建。

第 3 章是 RDD 的详细介绍，介绍了 RDD 的定义和 Spark 对于 DAG 的实现，最后通过 RDD 计算的详细介绍，讲解了 Spark 对于计算的实现原理。

第 4 章详细介绍任务调度的实现，包括如何通过 DAG 来生成计算任务，最后通过“Word Count”来加深对这个实现过程的理解。

第 5 章介绍了 Spark 的运行模式，尤其是 Standalone 模式。Standalone 是 Spark 自身实现的资源管理和调度的模块，这里会详细介绍它的实现原理。

第 6 章是 Executor 模块的详细讲解。Executor 是最终执行计算任务的单元，这章将详细介绍 Executor 的实现原理，包括 Executor 的分配、Task 在 Executor 的详细执行过程。

第 7 章详细介绍了 Spark 对于 Shuffle 的实现原理，包括基于 Hash 和基于排序的实现。除了详细阐述基于 Hash 和排序的 Shuffle 写和 Shuffle 读之外，还介绍了 Shuffle Pluggable 框架，为需要实现特定 Shuffle 逻辑的读者介绍其实现原理。

第 8 章详细介绍了 Spark 的 Storage 模块，在详细介绍了模块的架构后详细解析了不同存储级别的实现细节。

第 9 章介绍了 Spark 在百度、腾讯和阿里等国内互联网领域的应用现状。

致谢

本书在写作的过程中，得到了很多朋友、同仁的帮助和支持，在此表示衷心感谢！

感谢七牛云的技术总监陈超先生、蘑菇街的资深架构师刘旭晖先生、百度公司的高级架构师柴华先生、Databricks 软件工程师连城先生在百忙之中为本书审稿，并提出了很多宝贵的修改意见。尤其感谢星环科技的创始人兼 CTO 孙元浩先生对本书的完成给予了很多的支持，还在百忙之中为本书作序。感谢华为诺亚方舟实验室的董振华博士，在 Spark 上做机器学习方面给了我很多指导。

感谢百度上海研发中心网页搜索部的同事们。在一年多的工作中，笔者得到了很多同事的指导、支持和帮助，尤其感谢曲晶莹、吴永巍、汪韬、葛俊、刘桐仁、段雪涛、周波涛、马鑫云、李战平、杨大毛、朱晓阳、赵鹏程等。

感谢机械工业出版社的姚蕾编辑，她不但积极策划和推动本书的出版，还容忍我蜗牛般的写作速度；感谢她在这一年多的时间中给予的理解与支持。感谢机械工业出版社的李艺编辑为本书做了非常辛苦和专业的编辑工作。

还要感谢我的家人一直以来对我的支持和宽容。感谢父亲、母亲和三个姐姐，你们是我漫漫求学路的最强大支柱和后盾；感谢我的妻子王珊珊，不但在家庭方面付出很多，也为本书的顺利出版做出了很重要的贡献；感谢我的女儿，你的微笑是爸爸消除疲惫的良药。

联系方式

由于本书的写作都是在业余时间完成，加上笔者自身水平有限，错误在所难免，敬请读者谅解，如果有任何问题，可以通过下列方式与 Spark 的关注者和使用者进行交流沟通：

Spark 架构实现原理交流 QQ 群：473842835

Spark 用户使用交流 QQ 群：473853269

也可以直接与笔者联系：

邮箱：anzhsoft@gmail.com

新浪微博：@anzhsoft

个人博客：http://blog.csdn.net/anzhsoft

Contents 目　录

第 1 章 Chapter 1

Spark 简介

Apache Spark 是一种快速、通用、可扩展的大数据分析引擎。它是不断壮大的大数据分析解决方案家族中备受关注的明星成员，为分布式数据集的处理提供了一个有效框架，并以高效的方式处理分布式数据集。Spark 集批处理、实时流处理、交互式查询与图计算于一体，避免了多种运算场景下需要部署不同集群带来的资源浪费。Spark 在过去的一年中获得了极大关注，并得到广泛应用，Spark 社区也成为大数据领域和 Apache 软件基金会最活跃的项目之一，其活跃度甚至远超曾经只能望其项背的 Hadoop。

1.1 Spark 的技术背景

无论是工业界还是学术界，都已经广泛使用高级集群编程模型来处理日益增长的数据，如 MapReduce 和 Dryad。这些系统将分布式编程简化为自动提供位置感知性调度、容错以及负载均衡，使得大量用户能够在商用集群上分析超大数据集。大多数现有的集群计算系统都是基于非循环的数据流模型。即从稳定的物理存储（如分布式文件系统）中加载记录，记录被传入由一组确定性操作构成的 DAG（Directed Acyclic Graph，有向无环图），然后写回稳定存储。DAG 数据流图能够在运行时自动实现任务

调度和故障恢复。

尽管非循环数据流是一种很强大的抽象方法，但仍然有些应用无法使用这种方式描述。这类应用包括：①机器学习和图应用中常用的迭代算法（每一步对数据执行相似的函数）；②交互式数据挖掘工具（用户反复查询一个数据子集）。基于数据流的框架并不明确支持工作集，所以需要将数据输出到磁盘，然后在每次查询时重新加载，这会带来较大的开销。针对上述问题，Spark 实现了一种分布式的内存抽象，称为**弹性分布式数据集**（Resilient Distributed Dataset，RDD）。它支持基于工作集的应用，同时具有数据流模型的特点：自动容错、位置感知性调度和可伸缩性。RDD 允许用户在执行多个查询时显式地将工作集缓存在内存中，后续的查询能够重用工作集，这极大地提升了查询速度。

RDD 提供了一种高度受限的共享内存模型，即 RDD 是只读记录分区的集合，只能通过在其他 RDD 执行确定的转换操作（如 map、join 和 groupBy）而创建，然而这些限制使得实现容错的开销很低。与分布式共享内存系统需要付出高昂代价的检查点和回滚机制不同，RDD 通过 Lineage 来重建丢失的分区：一个 RDD 中包含了如何从其他 RDD 衍生所必需的相关信息，从而不需要检查点操作就可以重建丢失的数据分区。尽管 RDD 不是一个通用的共享内存抽象，但它具备了良好的描述能力、可伸缩性和可靠性，能够广泛适用于数据并行类应用。

1.2 Spark 的优点

Spark 凭借什么在众多的大数据分析处理平台中脱颖而出呢？这得益于 Spark 的几个主要优点，具体如下所示。

第一个优点是速度。与 Hadoop 的 MapReduce 相比，Spark 基于内存的运算要快 100 倍以上（如图 1-1 所示）；而基于硬盘的运算也要快 10 倍以上。Spark 实现了高效的 DAG 执行引擎，可以通过基于内存来高效处理数据流。

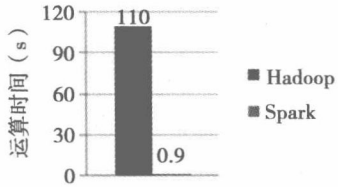

图 1-1 逻辑回归在 Hadoop、Spark 的运算速度对比