

TURING

图灵程序设计丛书

[PACKT]
PUBLISHING



[印] Sandeep Karanth 著 刘森 唐凯隼 陈智威 译

精通Hadoop

Mastering Hadoop



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书



[印] Sandeep Karam 著 刘森 唐凯隽 陈智威 译

精通Hadoop

Mastering Hadoop

人民邮电出版社
北京

图书在版编目 (C I P) 数据

精通Hadoop / (印) 卡伦斯 (Karanth, S.) 著 ; 刘
淼, 唐凯隽, 陈智威译. — 北京 : 人民邮电出版社,
2016. 1

(图灵程序设计丛书)
ISBN 978-7-115-41105-1

I. ①精… II. ①卡… ②刘… ③唐… ④陈… III.
①数据处理软件 IV. ①TP274

中国版本图书馆CIP数据核字(2015)第284747号

内 容 提 要

这本高阶教程将通过大量示例帮助你精通 Hadoop, 掌握 Hadoop 最佳实践和技巧。主要内容包括: Hadoop MapReduce、Pig 和 Hive 优化策略, YARN 审读剖析, 如何利用 Storm, 等等。如果你熟悉 Hadoop, 并想将自己的技能再提高一个层次, 本书是你的不二之选。

本书适合数据处理相关工作的专业人士阅读。

-
- ◆ 著 [印] Sandeep Karanth
译 刘 淼 唐凯隽 陈智威
责任编辑 岳新欣
执行编辑 李莉萍
责任印制 杨林杰
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京鑫正大印刷有限公司印刷
 - ◆ 开本: 800×1000 1/16
印张: 16.75
字数: 408千字 2016年1月第1版
印数: 1-4 000册 2016年1月北京第1次印刷
- 著作权合同登记号 图字: 01-2015-6628号

定价: 49.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京崇工商广字第 0021 号

推荐序一

Hadoop作为MapReduce的一个具体实现，已经引起了广泛关注。

今天在商业、科研等方面海量数据处理的需求已经越发普遍，开源的Hadoop成为自然而又合理的选择。

“给我一个参数，我能让大象的尾巴转起来。”Hadoop可以帮助我们应对海量数据存储与分析所带来的挑战。Hadoop具有很好的可扩展性、可靠性，且易于理解和使用。本书作者是一位从事数据工作多年的架构师，在大数据的架构及应用上具有非常丰富的经验，并且发表了不少相关的主题演讲。本书从Hadoop的发展历程开始，介绍了Hadoop 2.0版本带给大家的惊喜。其中不仅包括了Yarn、Pig、Hive等各个组件的高阶应用，而且还对如何在Yarn上引入其他计算框架（如Storm）进行了描述，这也为读者今后在实践中自行引入Tez、Spark等现在流行的计算框架打下了基础。除此以外，本书还包含了很多企业级应用中不可缺少的功能，如云、联合、安全等。而对于上述这些内容，书中既有准确的说明，又有翔实的示例。更为宝贵的是，作者将自己在10多年的项目中所积累的经验归纳总结并书写下来，对于广大读者来说，这是十分珍贵的财富。

本书原著是用英文写作的，它的内容组织得当，思路清晰，紧密结合实际。但是要把它翻译成中文介绍给中国的读者，并非易事。它不单单要求译者能够熟练地掌握英文，还要求他对书中的技术性内容有深入、准确的了解和掌握。本书的三位译者，在知名的互联网公司中从事着大数据架构师的工作，负责数百台规模的Hadoop集群的架构设计以及运营，对于Hadoop生态圈的各个组件有着非常丰富的实践经验，同时在开源社区中也很活跃。在他们的努力下，本书中文版终于要和中国的Hadoop开发者见面了。初学者可以通过阅读这本书入门，有一定经验的开发者、研究者也可以通过阅读这本书更上一层楼。希望大家可以通过阅读这本书获得收获。

春秋航空信息技术部副总经理 张振远

2015.9.8

推荐序二

近年来，随着“互联网+”行动计划的提出，互联网业态进一步繁荣，信息技术正以迅雷不及掩耳之势把信息转化为数据，但是人们如何利用好这些庞大的数据，挖掘其价值呢？古人云：“工欲善其事，必先利其器。”Hadoop正是解决数据处理问题的一把利器。Hadoop发展至今已相当成熟，可以说用好Hadoop，就能在“互联网+”时代中立于不败之地。

但是要驾驭好Hadoop这头大象并非易事。虽然Hadoop有强大的社区作为依托，用户可以从社区获得一些帮助，但是缺少文档描述，信息零碎，各种最佳实践散落在互联网的各个角落。而本书可以作为Hadoop的权威资料，其内容详实，涵盖了Hadoop生态圈的各个重要组件，如Hive、Pig、Storm，对于这些组件的介绍由浅入深，如Hive方面不但介绍了如何使用，还讲解了SQL优化器等内容。本书作者还紧跟技术发展潮流，对Hadoop 2的一些新特性也有深入讲解，如YARN、HDFS块放置策略等，读者可通过这些内容迅速掌握新特性，有助于在今后的工作中更好地使用Tez、Spark等新型计算引擎。此外，本书还详细介绍了一些企业级特性，比如Hadoop Security，这些内容对于安全地保护企业数据资产很有帮助。本书最后还介绍了亚马逊AWS和微软Azure提供的Hadoop云服务，读者可以学到如何在云服务上迅速搭建Hadoop集群。除以上内容之外，作者还在书中融入其多年的工作经验，这些经验可以说是无价之宝，如小文件问题的处理、任务的优化等，可以帮助Hadoop用户少走弯路，降低使用风险。总之，此书与众不同，在理论和应用之间找到了一个绝佳的平衡点。掌握Hadoop，有此书足矣。

另外，本书的译者，在他们的公司中运维着大型Hadoop集群，对于Hadoop、Hive、Tez、Spark等都有着丰富的应用经验，并且在Tez、Hive等开源社区中也很活跃。通过他们的翻译，可以说本书中文版的内容做到了信、达。我相信本书对于Hadoop使用者和开发者来说都是一份珍贵的资料。

Apache Tez PMC 章剑锋

2015.10.23

译者序

“数据爆炸的时代已经来临了吗？”毫无疑问，答案是肯定的。随着移动互联的普及，数据已经呈指数倍增长。但是，我们注意到此轮爆炸的数据大多来自社交及电商类。随着工业互联网的发展，越来越多的数据以及新鲜概念会出现在我们的生活中，并改变我们的生活。试想一下，如冰箱、空调、电视之类的家用电器触网以后，每时每刻向制造商或是内容提供商传输数据的场面；你所穿戴的鞋服，你所乘坐的交通工具，你所居住的智能楼宇，将来你周围的一切一切，包括衣食住行，每时每刻都会产生并传输着各种各样的数据。那时的数据规模将会是如何之庞大，不敢想象，而现在，爆炸才刚刚开始。

“你，做好准备了吗？”每当想到刚才所言的场面时，都会有点儿小怕，但更多的是激动和兴奋，因为我们身处这波浪潮之中。我们迎着风，踏着浪，朝着浪潮的巅峰前进。为此，我们必须有所准备，我们学习如何收集这些海量的数据，如何清洗、转换、分析它们，并最终形成有用的信息，希望能反馈给这个世界。古人云，工欲善其事，必先利其器。而我们的利器就是Hadoop。作为开源的分布式通用计算平台，该系统已被广泛采用。围绕着Hadoop的生态圈，我们可以轻松地完成之前所说的各种工作，并愿意将它推荐给更多的人来使用。

年中，当我们接触到*Mastering Hadoop*一书时，就决定要将它尽快翻译成中文并出版。因为这本书中不仅包含了很多Hadoop重要组件的最新特性，更关键的是它还包含了很多作者在职业生涯中通过各种项目所积累的最佳实践和建议。这些最佳实践和建议，即使是我们这些长期与Hadoop打交道的人也觉得获益匪浅。所以我们也希望这本书能为广大的Hadoop使用者和对大数据感兴趣的人提供帮助。

本书主要分为12章和1个附录。参加本书翻译工作的还有唐凯隽和陈智威两位同事，其中第1、2、6、12章由陈智威翻译，第4、5、8、9、10章以及附录由唐凯隽翻译。他们以自己的专长和领域知识，为本书提供了相应章节的内容翻译。

最后要感谢我们的家人、同事给予的支持，使得我们能够按时完成翻译。感谢图灵公司对我们的信任，将本书的翻译工作交付给我们。感谢所有参与本书编辑、审校和出版的每个人。

刘森@1号店

2015年10月于上海浦东

前 言

我们处在一个由数据主导决策的时代。存储成本在降低，网络速度在提升，周围的一切都在变得可以数字化，因此我们会毫不犹豫地下载、存储或与周围的其他人分享各类数据。大约20年前，相机还是一个使用胶片来捕捉图片的设备，每张照片所捕捉的都要是一个近乎完美的镜头，且底片的存储也要小心翼翼，以防损坏。要冲洗这些照片则需要更高的成本。从你按动快门到看到拍摄的图片几乎需要一天的时间。这意味着捕捉下来的信息要少得多，因为上述因素阻碍了人们记录生活的各个瞬间，只有那些被认为重要的时刻才被记录下来。

然而，随着相机的数字化，这种情况得到了改变。我们几乎随时随地都会毫不犹豫地拍照；我们从来不担心存储的问题，因为TB级别（ 2^{40} ）的外部磁盘可以提供可靠的备份；我们也很少到哪儿都带着相机，因为可以使用移动设备拍摄照片；我们还有如Instagram这样的应用给照片添加特效并分享这些美图；我们收集关于图片的意见和信息，还会基于这些内容做出决策；我们几乎不放过任何时刻，无论它们重要与否，都会将其存入纪念册中。大数据的时代来临啦！

在商业上，大数据时代也带来了类似的变化。每项商业活动的方方面面都被记录了下来：为提高服务质量，记录下用户在电子商务页面上的所有操作；为进行交叉销售或追加销售，记录下用户买下的所有商品。商家连客户的DNA恨不得都想掌握，因此只要是能得到的客户数据，他们都会想办法得到，并一个一个掐指研究。商家也不会受到数据格式的困扰，无论是语音、图像、自然语言文本，还是结构化数据，他们都会欣然接受。利用这些数据点，他们可以驱使用户做出购买决定，并且为用户提供个性化的体验。数据越多，越能为用户提供更好、更深入的个性化体验。

从某些方面来讲，我们已经准备好接受大数据的挑战了。然而，分析这些数据的工具呢？它们能处理如此庞大、快速、多样化的新数据吗？理论上说，所有数据都可以放到一台机器上，但这样一台机器的成本要多少？它能满足不断变化的负载需求吗？我们知道超级计算机可以做到这一点，但是全世界的超级计算机也就那么几台，而且都不具有伸缩性。替代方案就是构建一组机器、一个集群或者串联的计算单元来完成一项任务。一组使用高速网络互相连接的机器可以提供更好的伸缩性和灵活性，但那还不够。这些集群还要可编程。大量的机器，就像一群人，需要更多的协调和同步。机器的数量越多，集群中出现故障的可能性就越大。如何使用一种简单的方法处理同步和容错，从而减轻程序员的负担呢？答案是使用类似于Hadoop的系统。

Hadoop可以认为是大数据处理的同义词。简单的编程模型，“一次编码，任意部署”，和日益增长的生态圈，使得Hadoop成为一个可供不同技能水平的程序员共同使用的平台。今天，它是数据科学领域首屈一指的求职技能。要去处理和分析大数据，Hadoop成为了理所当然的工具。Hadoop 2.0扩张了它的羽翼，使其能覆盖各种类型的应用模式，并解决更大范围的问题。它很快成为所有数据处理需求的一个通用平台，并将在不久的将来成为各个领域每个工程师的必备技能。

本书涵盖了对MapReduce、Pig和Hive的优化及其高级特性，同时也展示了如何使用Hadoop 2.0版本扩展Hadoop的能力。

Hadoop 2.0版本的发布使其成为一个通用群机计算平台。本书阐明了为实现这一点而在平台层面所做出的改变，也介绍了对MapReduce作业以及像Pig、Hive这种高级抽象功能进行优化的行业准则，并对一些高级作业模式以及它们的应用进行了讨论。这些论述将帮助Hadoop用户优化已有的应用作业，并将它们迁移到Hadoop 2.0版本。随后，本书深入探讨了Hadoop 2.0的专属特性，如YARN（Yet Another Resource Negotiator）、HDFS联合，并辅以实例。本书后半部分还探讨了使用其他文件系统替换HDFS的问题。只要理解了上述这些问题，Hadoop用户就可以将Hadoop应用扩展到其他的应用模式和存储格式，使集群的资源得到更高效的利用。

这是一本聚焦于Hadoop高级概念和特性的参考书，每一个基本概念都使用代码段或者示意图来解释，而这些概念在章节中出现的顺序则是由数据处理流程的先后决定的。

本书内容

- 第1章，Hadoop 2.X，讨论Hadoop 2.X版本相比先前的版本做了哪些改进。
- 第2章，MapReduce进阶，通过举例帮助你理解Hadoop MapReduce的最佳实践和最佳模式。
- 第3章，Pig进阶，讨论Pig的高级特性，它是一种在Hadoop中编写MapReduce作业的框架。
- 第4章，Hive进阶，讨论Hadoop MapReduce中一种更高层面的SQL抽象（即Hive）的高级特性。
- 第5章，序列化和Hadoop I/O，讨论Hadoop的IO能力。这一章着重讨论了Hadoop中序列化和反序列化支持的概念，以及它们存在的必要性；Avro，一种第三方序列化框架；Hadoop中可用的数据压缩编码器；上述组件间的权衡；最后，介绍了Hadoop中的文件格式。
- 第6章，YARN——其他应用模式进入Hadoop的引路人。讨论Hadoop 2.X中一种新的资源管理器——YARN（另一种资源协调者），以及它如何将其他计算模式引入到Hadoop平台中。
- 第7章，基于YARN的Storm——Hadoop中的低延时处理，讨论的是一种与MapReduce这样的批量处理系统恰恰相反的计算模型，即让数据流向计算，并对两种不同的模式进行了对比。本章也讨论了Apache Storm的架构以及如何在Storm中开发应用。最后，你将学习如何在Hadoop 2.X中使用YARN来配置Storm。

- 第8章，云上的Hadoop，讨论云计算的特点，以及云计算服务供应商如何使用Hadoop平台提供服务。此外，还深入探讨了亚马逊的Hadoop服务管理，也就是所谓的Elastic MapReduce（EMR），并研究如何在Hadoop EMR集群中准备并执行作业。
- 第9章，HDFS替代品，讨论HDFS相对于其他文件系统的优点和缺点。本章还特别关注Hadoop对亚马逊的S3云存储服务的支持。最后，通过实现Hadoop对S3原生文件系统的支持来扩展Hadoop，从而证明Hadoop HDFS的可扩展性。
- 第10章，HDFS联合，讨论HDFS联合的优点及其架构。同时还讨论在MapReduce环境中HDFS获得成功的核心因素：块布局策略。
- 第11章，Hadoop安全，聚焦于Hadoop集群的安全方面。安全的主要保障是认证、授权、审计和数据保护。我们会在这些方面一一阐述Hadoop的特点。
- 第12章，使用Hadoop进行数据分析，讨论更高级的分析流程、机器学习等相关技术，以及它们对Hadoop的支持。本章列举了一个在Hadoop中使用Pig分析文档的例子，以演示数据分析功能。
- 附录，微软Windows中的Hadoop，探索在Hadoop 2.0版本中微软Windows操作系统对Hadoop的原生支持。本章中，我们将了解Windows是如何原生支持Hadoop的构建和部署的。

阅读准备

在尝试本书中的例子之前，你需要准备以下软件。

- Java开发工具包（JDK 1.7及以上版本）：这是Oracle公司的一个免费软件，它为开发者提供了JRE（Java Runtime Environment）和其他工具。下载地址：<http://www.oracle.com/technetwork/java/javase/downloads/index.html>。
- 编写Java代码用的集成开发环境：这里使用IntelliJ IDEA来开发例子，但其他的集成开发环境也是可以使用的。IntelliJ IDEA的社区版本可以从这里下载：<https://www.jetbrains.com/idea/download/>。
- Maven：本书中使用Maven构建例子。Maven可以自动解决依赖，并使用XML文件进行配置。各章中的示例代码可以使用两个简单的Maven命令来构建成一个JAR文件：

```
mvn compile
mvn assembly:single
```

这些命令将代码编译成一个JAR文件，并与依赖的文件一起生成一个联合JAR文件。在构建联合JAR文件时，很关键的一点是要修改pom.xml文件中引用的mainClass为需要执行的类名。

Hadoop相关的JAR文件可以使用如下命令执行：

```
hadoop jar <jar file> args
```

这个命令直接从pom.xml文件中指定的mainClass获取驱动程序。你可以从<http://maven.apache.org/download.cgi>下载得到Maven。用于构建本书例子的Maven的XML文件模板如下所示：

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://
maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>MasteringHadoop</groupId>
  <artifactId>MasteringHadoop</artifactId>
  <version>1.0-SNAPSHOT</version>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.0</version>
        <configuration>
          <source>1.7</source>
          <target>1.7</target>
        </configuration>
      </plugin>
      <plugin>
        <version>3.1</version>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-jar-plugin</artifactId>
        <configuration>
          <archive>
            <manifest>
              <mainClass>MasteringHadoop.MasteringHadoopTest</mainClass>
            </manifest>
          </archive>
        </configuration>
      </plugin>
      <plugin>
        <artifactId>maven-assembly-plugin</artifactId>
        <configuration>
          <archive>
            <manifest>
              <mainClass>MasteringHadoop.MasteringHadoopTest</mainClass>
            </manifest>
          </archive>
          <descriptorRefs>
            <descriptorRef>jar-with-dependencies</descriptorRef>
          </descriptorRefs>
        </configuration>
      </plugin>
    </plugins>
  </build>
</project>
```

```
</plugin>
</plugins>
<pluginManagement>
  <plugins>
    <!-- 这个插件的配置只用于保存Eclipse m2e的设置。
        它对Maven构建没有影响。 -->
    <plugin>
      <groupId>org.eclipse.m2e</groupId>
      <artifactId>lifecycle-mapping</artifactId>
      <version>1.0.0</version>
      <configuration>
        <lifecycleMappingMetadata>
          <pluginExecutions>
            <pluginExecution>
              <pluginExecutionFilter>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-dependency-plugin</artifactId>
                <versionRange>[2.1,)</versionRange>
                <goals>
                  <goal>copy-dependencies</goal>
                </goals>
              </pluginExecutionFilter>
              <action>
                <ignore />
              </action>
            </pluginExecution>
          </pluginExecutions>
        </lifecycleMappingMetadata>
      </configuration>
    </plugin>
  </plugins>
</pluginManagement>
</build>
<dependencies>
  <!-- 依赖可以设置在这里 -->
</dependencies>
</project>
```

- **Hadoop 2.2.0:** 试验书中的例子一般需要使用Apache Hadoop。附录详细介绍了在Windows平台下安装单机版Hadoop的方式。对于其他的操作系统，如Linux或Mac，安装步骤也类似甚至更简单，它们可以在如下地址找到：<http://hadoop.apache.org/docs/r2.2.0/hadoop-project-dist/hadoop-common/SingleNodeSetup.html>。

本书读者

这是一本为处于各个阶段的读者所写的书。Hadoop的新手可以用它来升级自己在此技术领域

的技能；有经验的人可以增强他们对Hadoop的了解，以解决工作中遇到的具有挑战性的数据处理难题；在工作中使用Hadoop、Pig或者Hive的人可以通过书中的建议提高他们作业运行的速度和效率；喜欢猎奇的大数据专业人士可以通过本书了解Hadoop的扩展领域，掌握它是如何通过融合其他应用模式来扩张自己的领地的，而不是仅限于MapReduce；最后，Hadoop 1.X的用户可以深入了解升级2.X后有哪些不同凡响之处。阅读本书之前，希望读者对Hadoop有一定的了解，但不需要你是这方面的专家。建议你在公司、云平台或者自己的台式机/笔记本上尝试安装Hadoop，以了解一些基本的概念。

排版约定

阅读本书时你会发现不同类别的信息使用了不同的文本样式，下面举例说明其中一些样式，及其表示的含义。

文本中的代码是这样表示的：“FileInputFormat的子类与相关联的类一般作为Hadoop作业的输入。”

代码块的表示方法如下：

```
return new CombineFileRecordReader<LongWritable,  
Text>((CombineFileSplit) inputSplit, taskAttemptContext,  
MasteringHadoopCombineFileRecordReader.class);  
}
```

命令行输入和输出的表示方法如下：

```
14/04/10 07:50:03 INFO input.FileInputFormat: Total input paths to process : 441
```

新术语和重要的关键字用楷体表示。



这个图标表示警告或需要特别注意的内容。



这个图标表示提示或者诀窍。

读者反馈

我们非常欢迎读者的反馈。告诉我们你觉得这本书怎么样，以及你喜欢哪部分或不喜欢哪部分。有了读者的反馈，我们才能继续写出真正能让大家充分受益的作品。

如果你想反馈信息，很简单，发送电子邮件至feedback@packtpub.com，请在邮件的主题中注明书名。

如果你也是某个领域的专家，并且有兴趣编写或者合作出版一本书，请参见我们的作者指南：www.packtpub.com/authors。

客户支持

很高兴你能成为本书的读者，而我们会提供很多东西让你倍感物有所值。

下载示例代码

凡是通过<http://www.packtpub.com>网站账户购买的Packt书籍，书中的示例代码都可以从网站上下载。如果你是从其他地方购买了本书，也可以访问<http://www.packtpub.com/support>并注册，我们会将示例代码通过邮件发送给你。

你也可以从我们的GitHub上获取到最新的示例代码：<https://github.com/karanth/MasteringHadoop>。

勘误表

虽然我们会全力确保书中内容的准确性，但错误仍在所难免。如果你在某本书中发现了错误（文字错误或代码错误），而且愿意向我们提交这些错误，我们感激不尽。这样不仅可以消除其他读者的疑虑，也有助于改进后续版本。若想提交你发现的错误，请访问<http://www.packtpub.com/submit-errata>，在“Errata Submission Form”（提交勘误表单）中选择相应图书，输入勘误详情。勘误通过验证之后将上传到Packt网站，或添加到现有的勘误列表中。若想查看某本书的现有勘误信息，请访问<http://www.packtpub.com/support>，选择相应的书名。

版权声明

对所有媒体来说，互联网盗版都是一个棘手的问题。Packt很重视版权保护。如果你在互联网上发现我们公司出版物的任何非法复制品，请及时告知我们网址或网站名称，以便我们采取补救措施。

如果发现可疑盗版材料，请通过copyright@packtpub.com联系我们。

你的举报可以帮助我们保护作者权益，也有利于我们不断出版高品质的图书。我们对你深表感激。

疑难解答

如果你对本书的任何内容存有疑问，请发送电子邮件到questions@packtpub.com，我们会尽力解决。

致 谢

我想把本书献给我至爱的女儿Avani，她令我在时间管理效率上一次次尝尽苦头。我要感谢我的妻子和父母给予我的不断支持，帮助我按时完成本书。Packt出版社极其热情地给予了我这个机会，因此我想感谢所有参与本书编辑、审校和出版的每个人。本书的许多内容源于我演讲中好奇听众的提问和反馈。书中探讨的一些小话题则源于我在职业生涯中各种各样的项目上所积累的丰富经验。因此，我要感谢我的听众和雇主，正是他们间接帮助我完成了本书。

目 录

第 1 章 Hadoop 2.X	1
1.1 Hadoop 的起源	1
1.2 Hadoop 的演进	2
1.3 Hadoop 2.X	6
1.3.1 Yet Another Resource Negotiator (YARN)	7
1.3.2 存储层的增强	8
1.3.3 支持增强	11
1.4 Hadoop 的发行版	11
1.4.1 选哪个 Hadoop 发行版	12
1.4.2 可用的发行版	14
1.5 小结	16
第 2 章 MapReduce 进阶	17
2.1 MapReduce 输入	18
2.1.1 InputFormat 类	18
2.1.2 InputSplit 类	18
2.1.3 RecordReader 类	19
2.1.4 Hadoop 的“小文件”问题	20
2.1.5 输入过滤	24
2.2 Map 任务	27
2.2.1 dfs.blocksize 属性	28
2.2.2 中间输出结果的排序与溢出	28
2.2.3 本地 reducer 和 Combiner	31
2.2.4 获取中间输出结果——Map 侧	31
2.3 Reduce 任务	32
2.3.1 获取中间输出结果—— Reduce 侧	32
2.3.2 中间输出结果的合并与溢出	33
2.4 MapReduce 的输出	34
2.5 MapReduce 作业的计数器	34
2.6 数据连接的处理	36
2.6.1 Reduce 侧的连接	36
2.6.2 Map 侧的连接	42
2.7 小结	45
第 3 章 Pig 进阶	47
3.1 Pig 对比 SQL	48
3.2 不同的执行模式	48
3.3 Pig 的复合数据类型	49
3.4 编译 Pig 脚本	50
3.4.1 逻辑计划	50
3.4.2 物理计划	51
3.4.3 MapReduce 计划	52
3.5 开发和调试助手	52
3.5.1 DESCRIBE 命令	52
3.5.2 EXPLAIN 命令	53
3.5.3 ILLUSTRATE 命令	53
3.6 Pig 操作符的高级特性	54
3.6.1 FOREACH 操作符进阶	54
3.6.2 Pig 的特殊连接	58
3.7 用户定义函数	61
3.7.1 运算函数	61
3.7.2 加载函数	66
3.7.3 存储函数	68
3.8 Pig 的性能优化	69
3.8.1 优化规则	69
3.8.2 Pig 脚本性能的测量	71
3.8.3 Pig 的 Combiner	72
3.8.4 Bag 数据类型的内存	72
3.8.5 Pig 的 reducer 数量	72
3.8.6 Pig 的 multiquery 模式	73

3.9 最佳实践	73	第 5 章 序列化和 Hadoop I/O	95
3.9.1 明确地使用类型	74	5.1 Hadoop 数据序列化	95
3.9.2 更早更频繁地使用投影	74	5.1.1 Writable 与 WritableComparable	96
3.9.3 更早更频繁地使用过滤	74	5.1.2 Hadoop 与 Java 序列化的区别	98
3.9.4 使用 LIMIT 操作符	74	5.2 Avro 序列化	100
3.9.5 使用 DISTINCT 操作符	74	5.2.1 Avro 与 MapReduce	102
3.9.6 减少操作	74	5.2.2 Avro 与 Pig	105
3.9.7 使用 Algebraic UDF	75	5.2.3 Avro 与 Hive	106
3.9.8 使用 Accumulator UDF	75	5.2.4 比较 Avro 与 Protocol Buffers/ Thrift	107
3.9.9 剔除数据中的空记录	75	5.3 文件格式	108
3.9.10 使用特殊连接	75	5.3.1 Sequence 文件格式	108
3.9.11 压缩中间结果	75	5.3.2 MapFile 格式	111
3.9.12 合并小文件	76	5.3.3 其他数据结构	113
3.10 小结	76	5.4 压缩	113
4.1 Hive 架构	77	5.4.1 分片与压缩	114
4.1.1 Hive 元存储	78	5.4.2 压缩范围	115
4.1.2 Hive 编译器	78	5.5 小结	115
4.1.3 Hive 执行引擎	78	第 6 章 YARN——其他应用模式进入 Hadoop 的引路人	116
4.1.4 Hive 的支持组件	79	6.1 YARN 的架构	117
4.2 数据类型	79	6.1.1 资源管理器	117
4.3 文件格式	80	6.1.2 Application Master	118
4.3.1 压缩文件	80	6.1.3 节点管理器	119
4.3.2 ORC 文件	81	6.1.4 YARN 客户端	120
4.3.3 Parquet 文件	81	6.2 开发 YARN 的应用程序	120
4.4 数据模型	82	6.2.1 实现 YARN 客户端	120
4.4.1 动态分区	84	6.2.2 实现 AM 实例	125
4.4.2 Hive 表索引	85	6.3 YARN 的监控	129
4.5 Hive 查询优化器	87	6.4 YARN 中的作业调度	134
4.6 DML 进阶	88	6.4.1 容量调度器	134
4.6.1 GROUP BY 操作	88	6.4.2 公平调度器	137
4.6.2 ORDER BY 与 SORT BY	88	6.5 YARN 命令行	139
4.6.3 JOIN 类型	88	6.5.1 用户命令	140
4.6.4 高级聚合	89	6.5.2 管理员命令	140
4.6.5 其他高级语句	90	6.6 小结	141
4.7 UDF、UDAF 和 UDTF	90		
4.8 小结	93		