HZ BOOKS
华章IT

资深大数据工程师，立足于企业真实场景，系统梳理和详尽讲解全栈大数据核心技术

为企业大数据技术选型和大数据平台构建提供成熟的解决方案，包含大量实用案例

大数据技术丛书

BigData Processing with Spark, Druid, Flume and Kafka

企业大数据处理

Spark、Druid、Flume与Kafka应用实践

肖冠宇◎著

机械工业出版社
China Machine Press

大数据
技术丛书

BigData Processing with Spark, Druid, Flume and Kafka

企业大数据处理

Spark、Druid、Flume与Kafka应用实践

肖冠宇◎著

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

企业大数据处理：Spark、Druid、Flume 与 Kafka 应用实践 / 肖冠宇著 . —北京：机械工业出版社，2017.9
（大数据技术丛书）

ISBN 978-7-111-57922-9

I. 企… II. 肖… III. 企业管理 – 数据处理 IV. F272.7

中国版本图书馆 CIP 数据核字（2017）第 212182 号

企业大数据处理
Spark、Druid、Flume 与 Kafka 应用实践

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：何欣阳　　责任校对：李秋荣
印　　刷：北京市荣盛彩色印刷有限公司　　版　　次：2017 年 9 月第 1 版第 1 次印刷
开　　本：186mm × 240mm　1/16　　印　　张：13.75
书　　号：ISBN 978-7-111-57922-9　　定　　价：59.00 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

我写本书的初衷是将自己在企业工作中应用的技术归纳总结，系统地将大数据处理相关技术融合在一起，给已经从事大数据相关技术研发工作的朋友，或是准备从其他行业转行进入大数据领域学习相关技术的朋友提供一份参考资料。希望本书能够帮助更多从事大数据相关工作的人，也希望通过本书结识更多热爱大数据的朋友。

目前，大数据已不只停留在概念阶段，而是在各领域成功落地，并取得了丰硕的成果。大数据已经渗透到生活中的各个方面，距离我们最近且与我们生活息息相关的大数据项目有交通大数据、医疗大数据、金融大数据、社交媒体大数据、互联网大数据等。如此多的大数据项目能够成功落地，关键原因在于数据来源的多样化，数据量的爆发式增长，新兴技术的快速发展，以及市场创新需求的不断增多，这为各种大数据项目提供了庞大的数据源，通过多种技术的综合应用，可不断挖掘出大数据背后的社会价值和商业价值。

随着开源社区的不断发展，越来越多的优秀项目被开源，以处理各种大数据场景下的问题和挑战。作为目前大数据生态系统内的早期开源项目，Hadoop 在廉价机器上实现了分布式数据存储和高性能分布式计算，大大降低了数据存储和计算成本。Hadoop 提供的分布式存储系统 HDFS、大数据集并行计算编程模型 MapReduce、资源调度框架 YARN 已经被广泛应用，为大数据生态系统的发展奠定了坚实的基础。如今，Hadoop 大数据生态圈发展已经非常全面，涉及领域众多，在大数据处理系统中常用的技术框架包括数据采集、数据存储、数据分析、数据挖掘、批处理、实时流计算、数据可视化、监控预警、信息安全等。下图展示了大数据生态系统内比较流行并且已经在生产环境验证过的开源技术。

（1）Spark

Spark 是由加州大学伯克利分校 AMP 实验室开源的分布式大规模数据处理通用引擎，具有高吞吐、低延时、通用易扩展、高容错等特点。Spark 内部提供了丰富的开发库，集成了数据分析引擎 Spark SQL、图计算框架 GraphX、机器学习库 MLlib、流计算引擎 Spark Streaming。

Spark 在函数式编程语言 Scala 中实现，提供了丰富的开发 API，支持 Scala、Java、Python、R 等多种开发语言。同时，它提供了多种运行模式，既可以采用独立部署的方式运行，也可以依托 Hadoop YARN、Apache Mesos 等资源管理器调度任务运行。目前，Spark 已经在金融、交通、医疗、气象等多种领域中广泛使用。

大数据生态系统中的开源技术

（2）Druid

Druid 是由美国 MetaMarkets 公司创建并开源的分布式提供海量时序数据存储、支持实时多维数据分析的 OLAP 系统，主要应用于广告数据分析、网络系统监控等场景。Druid 具有高吞吐、易扩展、高容错、低延迟、按时间序列存储等特点。

（3）Flume

Flume 是由 Cloudera 公司开发的分布式、高可用的日志收集系统，是 Hadoop 生态圈内的关键组件之一，目前已开源给 Apache。Flume 的原始版本为 Flume-OG，经过对整体架构的重新设计，现已改名为 Flume-NG。Flume 发展到现在已经不局限于日志收集，还可以通过简单的配置收集不同数据源的海量数据并将数据准确高效地传输到不同的中心存储。目前 Flume 可对接的主流大数据框架有 Hadoop、Kafka、ElasticSearch、Hive、HBase 等。在使用 Flume 的过程中，通过配置文件就可以实现整个数据收集过程的负载均衡和故障转移，而不需要修改 Flume 的任何代码。得益于优秀的框架设计，Flume 通过可扩展、插件化、组合式、高可用、高容错的设计模式，为用户提供了简单、高效、准确的轻量化大数据采集工具。

（4）Kafka

Kafka 是由 LinkedIn 开源的分布式消息队列，能够轻松实现高吞吐、可扩展、高可用，并且部署简单快速、开发接口丰富。目前，各大互联网公司已经在生产环境中广泛使用，而且已经有很多分布式处理系统支持使用 Kafka，比如 Spark、Strom、Druid、Flume 等。

（5）InfluxDB

InfluxDB 是一款开源分布式时序数据库，非常适合存储监控系统收集的指标数据。时序数据库顾名思义就是按照时间顺序存储指标数据，即监控系统的场景大部分是按照时间顺序存储各项指标数据，过期时间太长的指标可能将不会再关注，所以为了提高数据库的存储率，提高查询性能，需要定期删除过期指标。InfluxDB 的诸多特性非常适合监控系统的使用场景。

本书将详细介绍上述技术的原理，通过实践演示每种技术的实际应用场景。希望通过理论与实践相结合的方式使内容更通俗易懂，帮助读者根据实际的业务场景选择合适的技术方案，相信大数据在未来的发展中还会创造更多的价值。

内容概述

本书分三部分展开介绍：

第一部分（第 1 章）主要介绍了企业大数据系统的前期准备工作，包括如何构建企业大数据处理系统的软件环境和集群环境。

第二部分（第 2 ~ 7 章）首先介绍了 Spark 的基本原理，Spark 2.0 版本的 Spark SQL、Structured Streaming 原理和使用方法，以及 Spark 的多种优化方式；然后，介绍了 Druid 的基本原理、集群的搭建过程、数据摄入过程，以及在查询过程中如何实现 Druid 查询 API；接着介绍了日志收集系统 Flume 的基本架构和关键组件，以及分层日志收集架构的设计与实践；最后介绍了分布式消息队列 Kafka 的基本架构和集群搭建过程，以及使用 Java 语言实现客户端 API 的详细过程。

第三部分（第 8 ~ 9 章）主要介绍了企业大数据处理的两个实际应用案例，分别是基于 Druid 构建多维数据分析平台和基于 JMX 指标的监控系统。

目标读者

本书适合从事大数据及相关工作的工程师阅读，也适合准备进入大数据领域的大数据爱好者学习、参考。

读者反馈

本书是在业余时间完成的，由于水平有限，编写时间仓促，书中可能会出现介绍不够详细或者有错误的地方，敬请读者谅解。如果遇到任何问题或者寻求技术交流都可以通过如下联系方式与笔者进行沟通。

大数据爱好者交流 QQ 群：124154694

个人邮箱：xiaoguanyu_java@163.com

致谢

感谢在本书的写作过程中帮助过笔者的朋友、同事、老师，感谢你们一次又一次的帮助和支持！

感谢机械工业出版社杨福川老师，本书从2016年6月份开始筹划，确定了基本的框架，虽然由于笔者个人原因导致写作速度缓慢，但是杨老师一直积极推动本书的出版，并且不断指导笔者写作，感谢杨老师给予的理解、帮助与支持。感谢机械工业出版社编辑李艺老师，李艺老师用严谨的工作态度为本书做了专业的编辑工作，并且耐心指导笔者完成了本书的编写工作。

感谢乐视智能中心大数据部的同事们，感谢他们在工作中帮助笔者分担工作任务；感谢上级领导的耐心指导，使笔者能够顺利地完成工作任务并腾出时间进行写作。在此特别感谢技术总监罗宏宇、技术经理陆松林、刘韦宏、姚会航、张迪等。

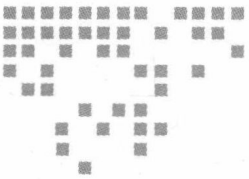
感谢家人在工作和生活中对笔者的帮助和照顾。感谢父母，平时因工作原因很少回家看望，但他们一直在背后支持我、鼓励我。感谢妻子为家庭和工作的付出。家人的陪伴与支持是笔者不断学习、努力奋斗的强大后盾！

Contents 目　录

第一部分 Part 1

准备工作

■ 第 1 章　基础环境准备



Chapter 1 第 1 章

基础环境准备

1.1 软件环境准备

软件版本选择：

操作系统：CentOS 6.6 版本；JDK：1.7 版本；Maven：3.2 版本；Scala：2.10 版本。

所有软件安装目录：/data/soft。

确定了软件版本后，我们将具体介绍软件的安装，本节主要介绍基础的软件安装方式。

1. JDK 安装

JDK 是 Java Development Kit 的简称，为 Java 语言开发的程序提供开发工具包和运行环境。JDK 安装的步骤如下：

（1）下载 JDK 二进制安装包

```
wget http://download.oracle.com/otn-pub/java/jdk/7u15-b03/jdk-7u15-linux-x64.tar.gz
```

（2）解压安装

```
tar -zxvf jdk-7u15-linux-x64.tar.gz
```

（3）创建软连接

软连接相当于快捷方式，便于后续版本更新升级。

```
ls -s /data/soft/jdk-7u15-linux-x64 /usr/local/jdk
```

（4）配置环境变量

```
vim /etc/profile
```

```
export JAVA_HOME=/usr/local/jdk
export JRE_HOME=$JAVA_HOME/jre
exportCLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
                :$JRE_HOME/lib:$CLASSPATH
export PATH=$PATH: $JAVA_HOME/bin
```

刷新环境变量使其生效：source /etc/profile

（5）验证安装是否成功

查看 JDK 版本命令：java -version

2. Maven 安装

Maven 是 Apache 开源的一个目前比较流行的项目管理和整合工具，能够自动完成项目的构建，并根据配置文件自动下载依赖组件，提供代码编译、打包、发布等功能。下面介绍 Maven 的详细安装过程。

Maven 安装的步骤如下：

（1）下载 Maven 二进制安装包

```
wget http://mirror.bit.edu.cn/apache/maven/maven-3/3.3.9/binaries/
apache-maven-3.3.9-bin.tar.gz
```

（2）解压安装

```
tar -zxvf apache-maven-3.3.9-bin.tar.gz
```

（3）创建软连接

软连接相当于快捷方式，便于后续版本更新升级。

```
ls -s/data/soft/apache-maven-3.3.9-bin /usr/local/maven
```

（4）配置环境变量

```
vim /etc/profile
export M2_HOME=/usr/local/maven
export PATH=$PATH: $JAVA_HOME/bin:$M2_HOM/bin
```

刷新环境变量使其生效：source /etc/profile

（5）验证安装是否成功

查看 Maven 版本命令：mvn -version

3. Scala 安装

Scala 编程语言是一种面向对象的函数式编程语言，充分展现了函数式编程语言简约、高效的特点，在程序开发的过程中可以引入 Java 语言，可扩展性强。由于 Scala 具有很多优秀的特性，越来越多的开源项目使用 Scala 语言开发，比如 Spark、Kafka 等。下面详细介绍 Scala 开发环境的安装过程。

Scala 安装的步骤如下：

（1）下载 JDK 二进制安装包

```
wget http://downloads.lightbend.com/scala/2.10.6/scala-2.10.6.tgz
```

（2）解压安装

```
tar -zxvf scala-2.10.6.tgz
```

（3）创建软连接

软连接相当于快捷方式，便于后续版本更新升级。

```
ls -s /data/soft/scala-2.10.6 /usr/local/scala
```

（4）配置环境变量

```
vim /etc/profile
export SCALA_HOME=/usr/local/scala
export PATH=$PATH: $JAVA_HOME/bin:$M2_HOM/bin:$SCALA_HOME/bin
```

刷新环境变量使其生效：`source /etc/profile`

（5）验证安装是否成功

查看 scala 版本命令：`scala-version`

1.2 集群环境准备

1.2.1 Zookeeper 集群部署

Zookeeper 是大数据系统中常用的分布式框架，主要用于公共配置管理、集群资源一致性管理、状态管理、部分分布式系统 Leader 选举等，下面通过完全分布式搭建方式进行介绍。

1. 集群规划

由于 Zookeeper 采用 FastLeaderElection 算法选举 Leader，集群中过半的机器正常运行才能够成功选举 Leader，为保证集群正常运行，集群部署的节点数为奇数个，最少节点个数为 3，生产环境建议部署 5 个以上的奇数个节点，因为 3 个实例其中只要有一个实例不可用，整个 Zookeeper 集群将无法成功选举，仍然不可以提供服务。

2. 部署过程

本例将以三个节点的部署为例，分别在 192.168.1.1、192.168.1.2、192.168.1.3 三台服务器部署一个 Zookeeper 实例。详细部署过程如下：

（1）下载安装包并解压

```
wget http://apache.fayea.com/zookeeper/zookeeper-3.4.6/zookeeper-3.4.6.tar.gz
```

解压到 /data/soft 目录下：

```
tar -zxvf http://apache.fayea.com/zookeeper/zookeeper-3.4.6/zookeeper-3.4.6.tar.gz
```

```
-C /data/soft
```

（2）创建软连接

创建软连接便于以后升级版本，方便统一管理。

```
ls -s /data/soft/zookeeper-3.4.6. /usr/local/zookeeper
```

（3）设置环境变量

```
vim /etc/profile
export ZOOKEEPER_HOME=/usr/local/zookeeper
export PATH=$PATH: $JAVA_HOME/bin:$M2_HOM/bin:$SCALA_HOME/bin
                 : $ZOOKEEPER_HOME/bin
```

刷新环境变量使其生效：`Source/etc/profile`

（4）配置

进入到 Zookeeper 安装目录：`cd /usr/local/zookeeper`

拷贝一份 conf 目录下的配置文件，重命名为 zoo.cfg：`cp ./conf/zoo_sample.cfg ./conf/zoo.cfg`

编辑配置文件设置关键参数：

```
tickTime=2000
initLimit=5
syncLimit=3
dataDir=/data/zookeeper/data
dataLogDir=/usr/local/zookeeper/logs
clientPort=2181
server.1=192.168.1.1:2888:3888
server.2=192.168.1.2:2888:3888
server.3=192.168.1.3:2888:3888
```

关键参数说明：

- tickTime：Zookeeper 中的基础参考时间，所有与时间相关的设置都为 tickTime 时间的整数倍，单位是毫秒。
- initLimit：Zookeeper Leader 与 Follower 初始连接时，Follower 需要从 Leader 同步最新数据，该值表示 Follower 同步数据的最大超时时间，一般为整数，表示是 tickTime 的整数倍时间。
- syncLimit：Leader 和 Follower 之间心跳检测的最大超时时间，超过这个时间则认为 Follower 已经下线。该参数值为整数，表示是 tickTime 的整数倍时间。
- dataDir：Zookeeper 持久化数据目录，建议与安装路径不在同一个路径下。
- dataLogDir：日志文件目录。
- clientPort：监听客户端连接的端口号，默认值为 2181。
- server.X=A:B:C。其中 X 是一个数字，表示这是第几号 server；A 是该 server 所在

的 IP 地址；B 配置该 server 和集群中的 leader 交换消息所使用的端口；C 配置选举 leader 时所使用的端口。

（5）创建 myid 文件

在配置参数 dataDir 对应的路径下新建 myid 文件，写入单独的一个数字，表示集群中该实例的编号，该值在集群中是唯一值，不可以重复，数字必须和 zoo.cfg 配置文件中的 server.X 中的 X 一一对应。

（6）启动 Zookeeper

```
bin/zkServer.sh start
```

（7）验证安装是否成功

`bin/zkServer.sh status`（一个 leader，两个 follower）

或者在 Zookeeper 安装的任何一个节点执行客户端连接命令：

```
bin/zkCli.sh -server 192.168.1.1:2181
```

1.2.2 Hadoop 部署

1. Hadoop 简介

Apache Hadoop 是由著名的 Apache 基金会开源的分布式存储计算系统，能够在廉价的硬件上轻松实现高可靠、高扩展、高性能、高容错等特性。通过增加机器即可直线增加集群的存储和计算能力。Hadoop 在大规模分布式系统中起着重要的作用，目前已经形成一套完整的 Hadoop 生态系统，并且在不断发展扩大。随着 Hadoop 生态系统的不断发展，Hadoop 已应用到互联网、大数据交通、智能医疗、气象监测、金融服务、人工智能等众多领域。

HDFS（Hadoop Distributed File System，Hadoop 分布式文件系统）：通过对文件分块多备份分布式存储的方式保证数据具有高效的容错能力，并且有效提高数据的吞吐量。

MapReduce：应用于规模分布式计算的编程模型，该模型包含 Map 和 Reduce 两种编程原语。Map 阶段常用于接入数据源，数据划分、过滤、整理等操作。Reduce 阶段常用于接收 Map 阶段的数据，聚合计算，持久化结果数据。

YARN：作业调度和集群资源管理框架。目前已经有很多开源项目部署到 YARN 上运行，将 YARN 作为统一的作业调度和资源管理框架，如 Spark、HBase、Tez 等。

2. Hadoop 集群部署

本节主要介绍 Hadoop2.6.4 版本的 Hadoop 集群部署。

1. 集群规划

为保证集群的高可用能力，NameNode 和 ResourceManager 都采用 HA 部署方式，各组件详细分布情况如表 1-1 所示。

表 1-1　Hadoop 集群规划

主机名	IP	运行进程
hadoop01、nn1、rm1	192.168.1.1	NameNode、DataNode、JournalNode、DFSZKFailoverController ResourceManager、NodeManager JobHistory Server QuorumPeerMain
hadoop02、nn2、rm2	192.168.1.2	NameNode、DataNode、JournalNode、DFSZKFailoverController ResourceManager、NodeManager QuorumPeerMain
hadoop03	192.168.1.3	DataNode、JournalNode NodeManager QuorumPeerMain

2. 部署过程

（1）SSH 免密码登录

使用 root 用户登录进入到 .ssh 目录下

```
cd ~/.ssh
```

执行 ssh-keygen -t rsa 生成公钥和私钥。系统会一直提示信息，一直按回车就可以。生成私钥文件 id_rsa，公钥文件 id_rsa.pub，认证文件 authorized_keys。

将公钥文件内容追加到认证文件中

```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

在免密码登录的机器之间互相拷贝公钥然后追加到认证文件中，即可完成 SSH 免密码登录配置。

（2）创建 hadoop 用户和组

```
groupadd hadoop
useradd -m -g hadoophadoop
```

（3）下载安装包并解压

先安装 hadoop01，然后将配置好的安装包拷贝到其他节点。

```
wget http://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-2.6.5/hadoop-2.6.5.tar.gz
```

解压到指定目录 /data/soft/ 下

```
tar -zxvf hadoop-2.6.5.tar.gz-C /data/soft/
```

（4）创建软连接并修改属主为 hadoop

创建软连接便于以后升级版本，方便统一管理。

```
ln -s /data/soft/ hadoop-2.6.5 /usr/local/hadoop
```