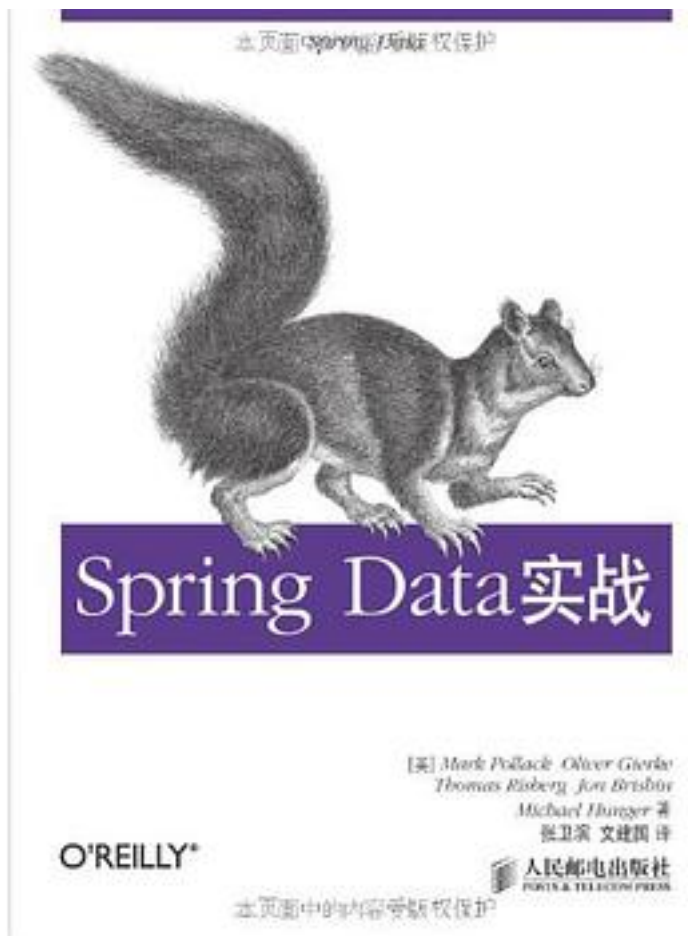


Spring Data实战



[Spring Data实战_下载链接1](#)

著者:[美] Mark Pollack

出版者:人民邮电出版社

出版时间:2014-4

装帧:平装

isbn:9787115343703

数据访问领域正在发生一场变革，关系型数据库无法解决的问题需要新的数据访问技术来解决。Spring Data项目就是一种简化Java应用构建的数据访问技术，它可以帮助开发人员高效地使

用最新的数据处理和管理工具，同时还能够以最新的方式使用传统的数据库。

《Spring Data实战》从Spring Data背景知识、关系型数据库、NoSQL、快速应用开发、大数据、数据网格6个方面深度解析了数据访问技术，介绍的内容都是最流行和前沿的，其中文档数据库、图数据库、键/值存储、Hadoop以及GemFire数据结构等是最重要的内容。《Spring Data实战》介绍了Spring Data、Repository、Querydsl的基础理念，然后阐述了借助Spring Data如何简化NoSQL和大数据的访问，并且涵盖了使用Spring Roo和Spring Data REST导出功能如何实现应用的快速开发，除此之外，书中还涉及与其他Spring子项目的协同工作，如Spring Integration和Spring Batch。

《Spring Data实战》面向实战、结构清晰，示例丰富，适用于各类Java开发人员和数据库开发人员，也可以作为各大、中专院校相关专业师生的参考用书和相关培训机构的培训教材。

作者介绍:

Mark Pollack博士曾在布鲁克黑文国家实验室研究高能物理学方面的大数据解决方案，随后转移到金融服务领域担任前端交易系统的技术领导和架构师。他长期以来关注软件开发流程的最佳实践和改善，Mark从2003年就参与核心Spring（Java）的开发，并在2004年成立了Microsoft对应的项目也就是Spring.NET。Mark现在领导着Spring Data项目，在使用大数据和NoSQL数据库这些新技术时，这个项目能够简化应用的开发。

Oliver Gierke是SpringSource的工程师，这是VMware的一个子部门，目前他担任Spring Data JPA、MongoDB以及核心模块的领导者。他参与企业级应用和开源项目的开发已经超过了6年，其工作的关注点在软件架构、Spring以及持久化技术方面。他经常在德国以及一些国际会议上进行演讲，写过很多的技术文章。

Thomas Risberg目前是Spring Data团队的成员，关注于MongoDB和JDBC扩展项目。他也是Spring框架项目的提交者，主要的贡献在于对JDBC框架的增强方面。Thomas在VMware的Cloud Foundry团队，为Cloud Foundry所支持的各种框架和语言开发集成方案。他是《Professional Java Development with the Spring Framework》一书的合著者，这本书出版于2005年，作者还包括Rod Johnson、Juergen Hoeller、Alef Arendsen以及Colin Sampaleanu。

Jon Brisbin是SpringSource Spring Data团队的成员，致力于为开发人员提供有用的库，从而简化下一代数据形式的管理。他曾帮忙将Grails GORM的对象映射器转移到基于Java的应用程序之中，并为集成Riak数据存储与Rabbit MQ消息代理提供组件。除此之外，他还针对事件应用模型撰写博客并发表演讲，他勤奋工作的领域还包括为前沿的非阻塞模式和传统的基于JVM的应用搭建桥梁。

Michael Hunger长期热衷于软件开发。他尤其关注于开发软件的人、软件技艺、编程语言以及代码提升。最近两年间，他与Neo Technology协作开发Neo4j图数据库。作为Spring Data

Neo4j的领导者，他为对象-图映射开发出了便利且完整的解决方案。他还参与Neo4j云托管。作为一名开发人员，Michael喜欢使用各种编程语言、每天学习新的东西、参与有趣且有前景的开源项目，并且参与编写了多本与编程相关的图书。Michael还是InfoQ的活跃编辑和采访者。

目录: 第一部分 背景知识

第1章 Spring Data项目 3

1.1 为Spring开发人员提供的NoSQL数据访问功能 3

1.2 主题概述 4

1.3 领域 5

1.4 示例代码 6

1.4.1 将源码导入到IDE 6

第2章 Repository：便利的数据访问层 11

2.1 快速入门 11

2.2 定义查询方法 14

2.2.1 查找查询的策略 14

2.2.2 衍生查询 14

2.2.3 分页和排序 15

2.3 定义Repository 16

2.3.1 调整Repository接口 17

2.3.2 手动实现Repository方法 18

2.4 IDE集成 20

2.4.1 IntelliJ IDEA 21

第3章 使用Querydsl实现类型安全的查询 23

3.1 Querydsl简介 23

3.2 生成查询元模型 26

3.2.1 构建系统集成 26

3.2.2 所支持的注解处理器 27

3.2.3 使用Querydsl对存储进行查询 28

3.3 集成Spring Data Repository 28

3.3.1 执行断言 29

3.3.2 手动实现Repository 29

第二部分 关系型数据库

第4章 JPA Repository 33

4.1 示例工程 33

4.2 传统方式 38

4.3 启动示例代码 39

4.4 使用Spring Data Repository 42

4.4.1 事务性 45

4.4.2 Repository与Querydsl集成 46

第5章 借助Querydsl SQL实现类型安全的JDBC编程 48

5.1 示例工程与搭建过程 48

5.1.1 HyperSQL数据库 49

5.1.2 Querydsl的SQL模块 50

5.1.3 构建系统集成 53

5.1.4 数据库模式 54

5.1.5 示例工程的领域实现 54

5.2 QueryDslJdbcTemplate 57

5.3 执行查询 58

5.3.1 Repository实现起步 59

5.3.2 查询单个对象 60

5.3.3 OneToManyResultSetExtractor抽象类 61

5.3.4 CustomerListExtractor实现 63

- 5.3.5 RowMapper的实现类 64
- 5.3.6 查询对象列表 65
- 5.4 插入、更新和删除操作 65
 - 5.4.1 使用SQLInsertClause进行插入操作 65
 - 5.4.2 使用SQLUpdateClause进行更新操作 66
 - 5.4.3 使用SQLDeleteClause进行删除行操作 67
- 第三部分 NoSQL
- 第6章 MongoDB: 文档存储 71
 - 6.1 MongoDB简介 71
 - 6.1.1 设置MongoDB 72
 - 6.1.2 使用MongoDB Shell 73
 - 6.1.3 MongoDB Java驱动 74
 - 6.2 使用Spring命名空间搭建基础设施 75
 - 6.3 映射模块 76
 - 6.3.1 领域模型 76
 - 6.3.2 搭建映射的基础设施 81
 - 6.3.3 索引 83
 - 6.3.4 自定义转换 84
 - 6.4 MongoTemplate 86
 - 6.5 Mongo Repository 88
 - 6.5.1 搭建基础设施 88
 - 6.5.2 Repository详解 88
 - 6.5.3 Mongo Querydsl集成 90
- 第7章 Neo4j: 图数据库 92
 - 7.1 图数据库 92
 - 7.2 Neo4j 93
 - 7.3 Spring Data Neo4j概览 95
 - 7.4 将领域建模为图 96
 - 7.5 使用Spring Data Neo4j持久化领域对象 101
 - 7.5.1 Neo4jTemplate 103
 - 7.6 组合发挥图和Repository的威力 104
 - 7.6.1 基本的图Repository操作 106
 - 7.6.2 衍生和基于注解的查找方法 106
 - 7.7 示例领域模型中的高级图用例 109
 - 7.7.1 单个节点的多重角色 109
 - 7.7.2 以产品分类和标签为例讲解图中的索引 110
 - 7.7.3 利用类似的兴趣(协同过滤) 111
 - 7.7.4 推荐 111
 - 7.8 事务、实体生命周期以及抓取策略 112
 - 7.9 高级映射模型 113
 - 7.10 使用Neo4j服务器 114
 - 7.11 从这里继续学习 115
- 第8章 Redis: 键/值存储 116
 - 8.1 Redis概述 116
 - 8.1.1 搭建Redis 116
 - 8.1.2 使用Redis Shell 117
 - 8.2 连接到Redis 118
 - 8.3 对象转换 119
 - 8.4 对象映射 121
 - 8.5 原子级计数器 123
 - 8.6 发布/订阅功能 123
 - 8.6.1 对信息进行监听和响应 124
 - 8.6.2 在Redis中使用Spring的缓存抽象 125
- 第四部分 快速应用开发

- 第9章 使用Spring Roo实现持久层 129
 - 9.1 Roo简介 129
 - 9.2 Roo的持久层 131
 - 9.3 快速起步 132
 - 9.3.1 借助命令行使用Roo 132
 - 9.3.2 借助Spring Tool Suite使用Roo 133
 - 9.4 Spring Roo JPA Repository示例 135
 - 9.4.1 创建工程 135
 - 9.4.2 搭建JPA持久化 135
 - 9.4.3 创建实体 135
 - 9.4.4 定义Repository 137
 - 9.4.5 创建Web层 138
 - 9.4.6 运行示例 139
 - 9.5 Spring MongoDB JPA Repository的例子 140
 - 9.5.1 创建工程 140
 - 9.5.2 搭建MongoDB持久化 140
 - 9.5.3 创建实体 140
 - 9.5.4 定义Repository 141
 - 9.5.5 创建Web层 141
 - 9.5.6 运行示例 141
- 第10章 REST Repository导出器 143
 - 10.1 示例工程 144
 - 10.1.1 与Rest导出器进行交互 146
 - 10.1.2 访问Product 148
 - 10.1.3 访问Customer 151
 - 10.1.4 访问Order 154
- 第五部分 大数据
- 第11章 Spring for Apache Hadoop 159
 - 11.1 Hadoop开发面临的挑战 159
 - 11.2 Hello World 161
 - 11.3 揭秘Hello World 163
 - 11.4 使用Spring for Apache Hadoop的Hello World 166
 - 11.5 在JVM中编写HDFS脚本 170
 - 11.6 结合HDFS脚本与Job提交 172
 - 11.7 Job调度 173
 - 11.7.1 使用TaskScheduler调度MapReduce Job 173
 - 11.7.2 使用Quartz调度MapReduce Job 175
- 第12章 使用Hadoop分析数据 176
 - 12.1 使用Hive 176
 - 12.1.1 Hello World 177
 - 12.1.2 运行Hive服务器 178
 - 12.1.3 使用Hive Thrift客户端 179
 - 12.1.4 使用Hive JDBC客户端 181
 - 12.1.5 使用Hive分析Apache日志文件 183
 - 12.2 使用Pig 184
 - 12.2.1 Hello World 185
 - 12.2.2 运行PigServer 187
 - 12.2.3 控制运行期脚本的执行 189
 - 12.2.4 在Spring Integration数据管道中调用Pig脚本 191
 - 12.2.5 使用Pig分析Apache日志文件 192
 - 12.3 使用HBase 193
 - 12.3.1 Hello World 193
 - 12.3.2 使用HBase Java客户端 194
- 第13章 使用Spring Batch和Spring Integration创建大数据管道 197

- 13.1 收集并将数据加载到HDFS 197
 - 13.1.1 Spring Integration介绍 198
 - 13.1.2 复制日志文件 200
 - 13.1.3 事件流 203
 - 13.1.4 事件转发 206
 - 13.1.5 管理 207
 - 13.1.6 Spring Batch简介 208
 - 13.1.7 从数据库中加载并处理数据 211
- 13.2 Hadoop工作流 214
 - 13.2.1 Spring Batch对Hadoop的支持 214
 - 13.2.2 将wordcount样例改造为Spring Batch应用 216
 - 13.2.3 Hive和Pig的步骤 218
- 13.3 从HDFS导出数据 219
 - 13.3.1 从HDFS到JDBC 219
 - 13.3.2 从HDFS到MongoDB 224
- 13.4 收集并加载数据到Splunk 225

第六部分 数据网格

第14章 分布式数据网格：GemFire 231

- 14.1 GemFire简介 231
- 14.2 缓存与域 232
- 14.3 如何获取GemFire 233
- 14.4 通过Spring XML命名空间配置GemFire 234
 - 14.4.1 缓存配置 234
 - 14.4.2 域配置 238
 - 14.4.3 缓存客户端配置 240
 - 14.4.4 缓存服务端配置 241
 - 14.4.5 WAN配置 242
 - 14.4.6 磁盘存储配置 243
- 14.5 使用GemfireTemplate进行数据访问 244
- 14.6 使用Repository 245
 - 14.6.1 POJO映射 245
 - 14.6.2 创建Repository 246
 - 14.6.3 PDX序列化 246
- 14.7 支持持续查询 247
- • • • • [\(收起\)](#)

[Spring Data实战_下载链接1](#)

标签

Spring

Data

NOSQL

Java

JPA

框架

技术

已经购买

评论

前面说的引人入胜，但是后面就没有说的那么好，完全是api方式的介绍。没有什么亮点。如果对spring repository了解的话，可以不用看。

太水。

稍微翻了一下, Spring DATA的巨大野心,却用了这么薄的书来说.每个点都说了一下讲不透彻,翻译的也一般,可读性不好.可能核心是书的项目代码吧.书的内容只是个参考..

[Spring Data实战_下载链接1](#)

书评

使用orm框架的时候，查询一直是一个痛点。如何有效率的做查询是一个很大的课题。querydsl是java查询中一个比较热门的框架，可是我看不出他的流行是基于什么，如果所有entity的需要单独生成一个类进行辅助查询，这得有多麻烦，为什么这么多人会接受？ 这么多辅助类在大型的项目...

翻译的太烂了 jlkdsjf lksroieuw rokldnx,cnvmsnldkjfoipewur slkdfnl
cnv,mxlksjoirpuewio jsalkdjfnl kasjdfkajseoiru laskdjfl;kasdf;l nxck jsjroiewur pao
skflkndjfgkjopiweur aslkdj lkajf oaiuer sjdfkaskjfoiewjur iousjflkasjdf oewiujr
asjdfkjdslkfj oiewur adls...

[Spring Data实战_下载链接1](#)