

Spring 5 核心原理

与30个类手写实战

谭勇德 (Tom) ©著

Java架构师实战宝典
10年Spring研究精华总结



咕泡学院 Java架构师成长丛书
gupaoedu.com



Spring 5 核心原理

与30个类手写实战

谭勇德 (Tom) ◎著



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书基于编程开发实践，不仅深度解析 Spring 5 的原理与新特性，更从环境准备、顶层结构设计、数据访问等方面一步步地推导出 Spring 的设计原理。在每个知识点上，均以大量的经典代码案例辅助讲解，使理论紧密联系实际。最后手写 30 个类，以体会 Spring 作者的创作过程，让每一位读者学以致用。

对于立志成为 Java 架构师的技术人员，以及对以 Spring 为核心的 Java Web 开发感兴趣的计算机专业高校生、在职 Java 开发人员来说，本书是一本具备超强实战意义的技术升级指南。读者通过本书可以看源码不再“晕车”，轻松找到入口；系统学习设计思想，提高解决问题的效率；培养架构思维能力，以及自驱学习能力。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目（CIP）数据

Spring 5 核心原理与 30 个类手写实战 / 谭勇德著. —北京：电子工业出版社，2019.7
（咕泡学院 Java 架构师成长丛书）
ISBN 978-7-121-36741-0

I. ①S… II. ①谭… III. ①JAVA 语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字（2019）第 111737 号

责任编辑：董 英

印 刷：三河市良远印务有限公司

装 订：三河市良远印务有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱

邮编：100036

开 本：787×980 1/16 印张：35

字数：778 千字

版 次：2019 年 7 月第 1 版

印 次：2019 年 7 月第 1 次印刷

印 数：6000 册 定价：118.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlt@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：010-51260888-819，faq@phei.com.cn。

不只是一个技术者，
更要做一个思考者！

谭勇德/Tom

序 言

在 1996 年，Java 还只是一门新兴的、初出茅庐的编程语言。2004 年 3 月 24 日，Spring 正式对外发布 1.0 版本。我在 2009 年开始接触 Spring 2.0，从此爱不释手。Spring 的出现，恰如其名，开启了全世界 Java 程序员的春天。如今，Spring 已然成为一个生态，使用 Spring 简直就是一种享受。

本书结合我多年的 Spring 使用经验，整理了珍藏多年的 Spring 学习笔记，采用类图和代码片段的形式，加以中文注释，通俗、生动、全面、深入地剖析了 Spring 源码的重要细节。要想练就“降龙十八掌”，先得修炼内功。本书从设计模式开始讲解，以帮助大家更好地理解 Spring，让大家知其然，且知其所以然。

如果你现在的工作经验与应有的能力无法匹配，
如果你在自学路上找不到方向，
如果你现在是初级程序员，想往高级程序员晋级，
如果你希望成为架构师，
……

本书都能帮到你。

编写本书旨在使大家进入不一样的思维境界，手写源码不是为了重复造轮子，也不是为了装高大上，其实只是我推荐给大家的一种学习方式。

书中个人观点若有不妥之处，恳望纠正！

关于本书

适用对象	<ul style="list-style-type: none">• 具有Spring开发经验，想深入了解Spring实现原理的人• 看源码不知如何下手的人• 希望理解设计模式在源码中的应用的人• 立志成为Java架构师的人
源码版本	Spring 5.0.2.RELEASE
IDE版本	IntelliJ IDEA 2017.1.4
JDK版本	JDK 1.8及以上
Gradle	Gradle 4.0及以上
Maven版本	3.5.0及以上

读者服务

轻松注册成为博文视点社区用户（www.broadview.com.cn），扫码直达本书页面。

- **下载资源**：本书如提供示例代码及资源文件，均可在 [下载资源](#) 处下载。
- **提交勘误**：您对书中内容的修改意见可在 [提交勘误](#) 处提交，若被采纳，将获赠博文视点社区积分（在您购买电子书时，积分可用来抵扣相应金额）。
- **交流互动**：在页面下方 [读者评论](#) 处留下您的疑问或观点，与我们和其他读者一同学习交流。

页面入口：<http://www.broadview.com.cn/36741>



关于我

为什么说来自文艺界？

我自幼爱好书法和美术，长了一双能书会画的手，而且手指又长又白，因此以前的艺名叫“玉手艺人”。中学期间，曾获市级书法竞赛一等奖，校园美术竞赛一等奖，校园征文比赛二等奖。担任过学生会宣传部长，负责校园黑板报、校园刊物的编辑、排版、设计。

2008 年参加工作后，做过家具建模、平面设计等工作，亲自设计了咕泡学院的 Logo。做讲师之后，给自己起了一个跟姓氏谐音的英文名字“Tom”，江湖人称“编程界写字写得最好的、书法界编程最牛的文艺汤”。

我的技术生涯

我的 IT 技术生涯应该算是从 2009 年开始的，在此之前做过 UI 设计，做过前端网页，到 2009 年才真正开始参与 Java 后台开发。在这里要感谢所有帮助过我入门编程的同事和老师。从 2010 年至 2014 年担任过项目组长、项目经理、架构师、技术总监，对很多的开源框架建立了自己的独特见解。我会习惯性地用形象思维来理解抽象世界。譬如：看到二进制 0 和 1，我会想到《周易》中的两仪——阴和阳；看到颜色值用 RGB 表示，我会想到美术理论中的太阳光折射三原色；下班回家看到炒菜流程，我会想到模板方法模式；坐公交车看到学生卡、老人卡、爱心卡，我会想到策略模式；等等。大家看到的这本书，很多地方都融入了这种形象思维。

为什么写书？

其实一开始我没想过要写书，写书的初衷主要是满足学员的诉求。大家认为我个人的学习方法、思维模式、教学方式通俗易懂，很容易让人接受，但是通过视频形式传播受众有限，学员建议我把这些宝贵的经验以纸质书的形式奉献给大家，这样定会给社会带来更大的价值。

借此机会，特别感谢责任编辑董英及电子社的团队成员为本书审稿纠错；感谢我老婆在无数个加班的夜晚给我默默的支持；感谢咕泡学院的学员给本书内容提出宝贵的修改意见。

谭勇德 (Tom)

2019 年 5 月 于 长沙

反侵权盗版声明

电子工业出版社依法对本作品享有专有出版权。任何未经权利人书面许可，复制、销售或通过信息网络传播本作品的行为；歪曲、篡改、剽窃本作品的行为，均违反《中华人民共和国著作权法》，其行为人应承担相应的民事责任和行政责任，构成犯罪的，将被依法追究刑事责任。

为了维护市场秩序，保护权利人的合法权益，我社将依法查处和打击侵权盗版的单位和个人。欢迎社会各界人士积极举报侵权盗版行为，本社将奖励举报有功人员，并保证举报人的信息不被泄露。

举报电话：(010)88254396；(010)88258888

传 真：(010)88254397

E-mail：dbqq@phei.com.cn

通信地址：北京市万寿路 173 信箱

电子工业出版社总编办公室

邮 编：100036

目 录

第 1 篇 Spring 内功心法

第 1 章 软件架构设计原则.....	2
1.1 开闭原则.....	2
1.2 依赖倒置原则.....	4
1.3 单一职责原则.....	7
1.4 接口隔离原则.....	10
1.5 迪米特原则.....	12
1.6 里氏替换原则.....	14
1.7 合成复用原则.....	19
1.8 设计原则总结.....	20
第 2 章 Spring 中常用的设计模式.....	21
2.1 为什么要从设计模式开始.....	21
2.1.1 写出优雅的代码.....	22
2.1.2 更好地重构项目.....	24
2.1.3 经典框架都在用设计模式解决问题.....	36
2.2 工厂模式详解.....	36
2.2.1 工厂模式的由来.....	36
2.2.2 简单工厂模式.....	37
2.2.3 工厂方法模式.....	41
2.2.4 抽象工厂模式.....	43
2.2.5 利用工厂模式重构的实践案例.....	47
2.3 单例模式详解.....	53
2.3.1 单例模式的应用场景.....	53
2.3.2 饿汉式单例模式.....	53

2.3.3	懒汉式单例模式.....	54
2.3.4	反射破坏单例.....	60
2.3.5	序列化破坏单例.....	61
2.3.6	注册式单例模式.....	68
2.3.7	线程单例实现 ThreadLocal.....	74
2.3.8	单例模式小结.....	75
2.4	原型模式详解.....	75
2.4.1	原型模式的应用场景.....	75
2.4.2	浅克隆.....	77
2.4.3	深克隆.....	79
2.4.4	克隆破坏单例模式.....	81
2.4.5	clone()方法的源码.....	82
2.5	代理模式详解.....	82
2.5.1	代理模式的应用场景.....	82
2.5.2	静态代理.....	83
2.5.3	动态代理.....	88
2.5.4	代理模式与 Spring.....	107
2.5.5	静态代理和动态代理的本质区别.....	108
2.5.6	代理模式的优缺点.....	109
2.6	委派模式详解.....	109
2.6.1	委派模式的定义及应用场景.....	109
2.6.2	委派模式在源码中的体现.....	111
2.7	策略模式详解.....	114
2.7.1	策略模式的应用场景.....	114
2.7.2	用策略模式实现选择支付方式的业务场景.....	114
2.7.3	策略模式在 JDK 源码中的体现.....	122
2.7.4	策略模式的优缺点.....	125
2.7.5	委派模式与策略模式综合应用.....	125
2.8	模板模式详解.....	129
2.8.1	模板模式的应用场景.....	129
2.8.2	利用模板模式重构 JDBC 操作业务场景.....	132
2.8.3	模板模式在源码中的体现.....	136
2.8.4	模板模式的优缺点.....	138
2.9	适配器模式详解.....	139
2.9.1	适配器模式的应用场景.....	139
2.9.2	重构第三方登录自由适配的业务场景.....	141
2.9.3	适配器模式在源码中的体现.....	149

2.9.4	适配器模式的优缺点	153
2.10	装饰者模式详解	153
2.10.1	装饰者模式的应用场景	153
2.10.2	装饰者模式和适配器模式对比	163
2.10.3	装饰者模式在源码中的应用	163
2.10.4	装饰者模式的优缺点	165
2.11	观察者模式详解	165
2.11.1	观察者模式的应用场景	165
2.11.2	观察者模式在源码中的应用	175
2.11.3	基于 Guava API 轻松落地观察者模式	176
2.11.4	观察者模式的优缺点	177
2.12	各设计模式的总结与对比	177
2.12.1	GoF 23 种设计模式简介	177
2.12.2	设计模式之间的关联关系	178
2.12.3	Spring 中常用的设计模式	182
2.13	Spring 中的编程思想总结	183

第 2 篇 Spring 环境预热

第 3 章	Spring 的前世今生	186
3.1	一切从 Bean 开始	187
3.2	Spring 的设计初衷	188
3.3	BOP 编程伊始	188
3.4	理解 BeanFactory	189
3.5	AOP 编程理念	189
第 4 章	Spring 5 系统架构	191
4.1	核心容器	192
4.2	AOP 和设备支持	192
4.3	数据访问与集成	193
4.4	Web 组件	194
4.5	通信报文	194
4.6	集成测试	194
4.7	集成兼容	194
4.8	各模块之间的依赖关系	194

第 5 章	Spring 版本命名规则.....	196
5.1	常见软件版本号命名.....	196
5.2	语义化版本号命名通行规则.....	197
5.3	商业软件中常见的修饰词.....	197
5.4	软件版本号使用限定.....	198
5.5	Spring 版本命名规则.....	199
第 6 章	Spring 源码下载及构建技巧.....	200
6.1	Spring 5 源码下载.....	200
6.2	基于 Gradle 的源码构建技巧.....	201
6.3	Gradle 构建过程中的坑.....	207
第 3 篇 Spring 核心原理		
第 7 章	用 300 行代码手写提炼 Spring 核心原理.....	210
7.1	自定义配置.....	210
7.1.1	配置 application.properties 文件.....	210
7.1.2	配置 web.xml 文件.....	210
7.1.3	自定义注解.....	211
7.1.4	配置注解.....	212
7.2	容器初始化.....	213
7.2.1	实现 1.0 版本.....	213
7.2.2	实现 2.0 版本.....	216
7.2.3	实现 3.0 版本.....	223
7.3	运行效果演示.....	227
第 8 章	一步一步手绘 Spring IoC 运行时序图.....	228
8.1	Spring 核心之 IoC 容器初体验.....	228
8.1.1	再谈 IoC 与 DI.....	228
8.1.2	Spring 核心容器类图.....	229
8.1.3	Web IoC 容器初体验.....	232
8.2	基于 XML 的 IoC 容器的初始化.....	237
8.2.1	寻找入口.....	238
8.2.2	获得配置路径.....	238
8.2.3	开始启动.....	240

8.2.4	创建容器.....	242
8.2.5	载入配置路径.....	243
8.2.6	分配路径处理策略.....	244
8.2.7	解析配置文件路径.....	247
8.2.8	开始读取配置内容.....	249
8.2.9	准备文档对象.....	250
8.2.10	分配解析策略.....	251
8.2.11	将配置载入内存.....	252
8.2.12	载入<bean>元素.....	257
8.2.13	载入<property>元素.....	261
8.2.14	载入<property>子元素.....	264
8.2.15	载入<list>子元素.....	266
8.2.16	分配注册策略.....	267
8.2.17	向容器注册.....	267
8.3	基于注解的 IoC 初始化.....	270
8.3.1	注解的前世今生.....	270
8.3.2	定位 Bean 扫描路径.....	271
8.3.3	读取注解的元数据.....	273
8.3.4	扫描指定包并解析为 BeanDefinition.....	277
8.3.5	注册注解 BeanDefinition.....	283
8.4	IoC 容器初始化小结.....	285
第 9 章	一步一步手绘 Spring DI 运行时序图.....	287
9.1	Spring 自动装配之依赖注入.....	287
9.1.1	依赖注入发生的时间.....	287
9.1.2	寻找获取 Bean 的入口.....	288
9.1.3	开始实例化.....	293
9.1.4	选择 Bean 实例化策略.....	297
9.1.5	执行 Bean 实例化.....	299
9.1.6	准备依赖注入.....	301
9.1.7	解析属性依赖注入规则.....	306
9.1.8	注入赋值.....	310
9.2	Spring IoC 容器中那些鲜为人知的细节.....	314
9.2.1	关于延时加载.....	314
9.2.2	关于 FactoryBean 和 BeanFactory.....	317
9.2.3	再述 autowiring.....	322

第 10 章 一步一步手绘 Spring AOP 运行时序图	326
10.1 Spring AOP 初体验	326
10.1.1 再述 Spring AOP 应用场景	326
10.1.2 AOP 中必须明白的几个概念	327
10.1.3 使用 Spring AOP 的两种方式	329
10.1.4 切入点表达式的配置规则	333
10.2 Spring AOP 源码分析	334
10.2.1 寻找入口	334
10.2.2 选择代理策略	338
10.2.3 调用代理方法	341
10.2.4 触发通知	347
第 11 章 一步一步手绘 Spring MVC 运行时序图	352
11.1 初探 Spring MVC 请求处理流程	352
11.2 Spring MVC 九大组件	353
11.2.1 HandlerMapping	353
11.2.2 HandlerAdapter	353
11.2.3 HandlerExceptionResolver	354
11.2.4 ViewResolver	354
11.2.5 RequestToViewNameTranslator	354
11.2.6 LocaleResolver	354
11.2.7 ThemeResolver	355
11.2.8 MultipartResolver	355
11.2.9 FlashMapManager	355
11.3 Spring MVC 源码分析	355
11.3.1 初始化阶段	356
11.3.2 运行调用阶段	359
11.4 Spring MVC 优化建议	367

第 4 篇 Spring 手写实战

第 12 章 环境准备	370
12.1 IDEA 集成 Lombok 插件	370
12.1.1 安装插件	370
12.1.2 配置注解处理器	373

12.1.3	使用插件	374
12.2	从 Servlet 到 ApplicationContext	375
12.3	准备基础配置	376
12.3.1	application.properties 配置	377
12.3.2	pom.xml 配置	377
12.3.3	web.xml 配置	378
12.3.4	GPDispatcherServlet	378
第 13 章	IoC 顶层结构设计	380
13.1	Annotation（自定义配置）模块	380
13.1.1	@GPService	380
13.1.2	@GPAutowired	381
13.1.3	@GPController	381
13.1.4	@GPRequestMapping	382
13.1.5	@GPRequestParam	382
13.2	core（顶层接口）模块	382
13.2.1	GPFactoryBean	382
13.2.2	GPBeanFactory	383
13.3	beans（配置封装）模块	383
13.3.1	GPBeanDefinition	383
13.3.2	GPBeanWrapper	384
13.4	context（IoC 容器）模块	385
13.4.1	GPAbstractApplicationContext	385
13.4.2	GPDefaultListableBeanFactory	385
13.4.3	GPApplicationContext	385
13.4.4	GPBeanDefinitionReader	388
13.4.5	GPApplicationContextAware	391
第 14 章	完成 DI 模块的功能	392
14.1	从 getBean()方法开始	393
14.2	GPBeanPostProcessor	395
第 15 章	完成 MVC 模块的功能	396
15.1	MVC 顶层设计	396
15.1.1	GPDispatcherServlet	396

15.1.2	GPHandlerMapping	402
15.1.3	GPHandlerAdapter	403
15.1.4	GPModelAndView	406
15.1.5	GPViewResolver	406
15.1.6	GPView	407
15.2	业务代码实现	409
15.2.1	IQueryService	409
15.2.2	QueryService	410
15.2.3	IModifyService	410
15.2.4	ModifyService	411
15.2.5	MyAction	412
15.2.6	PageAction	413
15.3	定制模板页面	414
15.3.1	first.html	414
15.3.2	404.html	414
15.3.3	500.html	415
15.4	运行效果演示	415
第 16 章 完成 AOP 代码织入		417
16.1	基础配置	417
16.2	完成 AOP 顶层设计	418
16.2.1	GPJoinPoint	418
16.2.2	GPMethodInterceptor	419
16.2.3	GPAopConfig	419
16.2.4	GPAdvisedSupport	420
16.2.5	GPAopProxy	422
16.2.6	GPCglibAopProxy	423
16.2.7	GPJdkDynamicAopProxy	423
16.2.8	GPMethodInvocation	425
16.3	设计 AOP 基础实现	427
16.3.1	GPAdvice	427
16.3.2	GPAbstractAspectJAdvice	427
16.3.3	GPMethodBeforeAdvice	428
16.3.4	GPAfterReturningAdvice	429
16.3.5	GPAfterThrowingAdvice	430
16.3.6	接入 getBean()方法	430

16.4	织入业务代码	432
16.4.1	LogAspect	432
16.4.2	IModifyService	433
16.4.3	ModifyService	434
16.5	运行效果演示	435

第 5 篇 Spring 数据访问

第 17 章	数据库事务原理详解	438
17.1	从 Spring 事务配置说起	438
17.2	事务的基本概念	439
17.3	事务的基本原理	439
17.4	Spring 事务的传播属性	440
17.5	数据库事务隔离级别	441
17.6	Spring 中的事务隔离级别	441
17.7	事务的嵌套	442
17.8	Spring 事务 API 架构图	444
17.9	浅谈分布式事务	444
第 18 章	Spring JDBC 源码初探	446
18.1	异常处理	447
18.2	config 模块	448
18.3	core 模块	450
18.4	DataSource	456
18.5	object 模块	457
18.6	JdbcTemplate	458
18.7	NamedParameterJdbcTemplate	458
第 19 章	基于 Spring JDBC 手写 ORM 框架	459
19.1	实现思路概述	459
19.1.1	从 ResultSet 说起	459
19.1.2	为什么需要 ORM 框架	464
19.2	搭建基础架构	467
19.2.1	Page	467
19.2.2	ResultMsg	470