

杭建 / 著

Java工程师修炼之道

为入门者条分缕析地梳理Java技术体系 串联后端关键技能供有经验者实战进阶



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn



Java工程师修炼之道

杭建 / 著

電子工業出版社

Publishing House of Electronics Industry

北京•BEIJING

内容简介

Java 开发一直是当前互联网领域最火热的开发技能之一,Java 工程师也一直是需求量非常大的开发职位。本书主要针对一名合格的 Java 工程师的必备技能做了大纲性的总结和阐述。本书内容包括了工程化、常用开发框架、数据存储、数据传输、Java 编程高级知识、性能优化、安全技术等内容,基本涵盖了 Java 工程师需要掌握的绝大部分技能点。

本书可以看作一本 Java 工程师的入职指南,也可以看作一本串联 Java 后端技能点的参考手册。通过精心编排的内容,刚入门的 Java 工程师能够体系化地学习相关开发技能,有经验的 Java 工程师能够查漏补缺,巩固自己的相关开发技能,进一步完善自身的 Java 技术体系。

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

Java 工程师修炼之道 / 杭建著. —北京: 电子工业出版社, 2018.3

ISBN 978-7-121-33501-3

I . ① J… II . ① 杭… III . ① JAVA 语言—程序设计 IV . ① TP312.8

中国版本图书馆 CIP 数据核字 (2018) 第 010963 号

策划编辑: 张春雨

责任编辑: 付 睿

印 刷: 三河市双峰印刷装订有限公司

装 订: 三河市双峰印刷装订有限公司

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: 25.25 字数: 541 千字

版 次: 2018 年 3 月第 1 版

印 次: 2018 年 7 月第 3 次印刷

定 价: 89.00 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 zltz@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。

推荐序一

Hey！新来的读者，为了吸引你的注意力我真是煞费苦心，但最终还是没能写出一句特别吸引眼球的话来，毕竟写序的我不是标题党出身。此刻我真的非常能理解你拿到新书之后那渴望知识的心情，所以你恨不得一个字的“序”也不要看到，直接到达“最有价值”的知识点。但作为一名资深转业码农（对！你没看错，是“转业”，不是“专业”）还是想说一句，你先看完序，5分钟后到达知识的战场，会更稳！

相信你已经在看“序”了，那么我们来说点正经事。

你的知识体系的养成有3个关键阶段：看山是山，看山不是山，看山还是山。本书的适用人群是“看山不是山”的那些人，如果你恰好处于这个阶段，恭喜你！书钱没白花。

Java 是一门非常容易入门的语言，初学者经过初期的学习之后基本能掌握 DEMO 级别的编程应用。相信读者你已经度过了这个阶段，但是 Java 庞大的体系可能会把你绕晕，又或者你还没看到 Java 的生态系统有多么复杂。此时，你需要本书。从事程序员这个工作，到比较高阶的时候，其实是不挑语言的，语言只是工具，而你可以在纷繁复杂中游刃有余。但几乎每一位高手都是先深入一个领域，再横向发展的。你可以不用着急后续的横向发展，先坚定自己学习 Java 的信心！因为，从广泛的应用场景、顶级的开源生态、未来可期的薪水和职位来说，Java 都是非常不错的选择。

敲黑板，画重点！下面来解释一下，为什么本书面向的是“看山不是山”的人群。在度过 Java 的入门期之后，会有一个烦恼，那就是面对 Java 这么庞大的体系，我们究竟应该学习什么？选择方向，往往比努力更重要！是使用 J2SE 编写桌面程序？是使用 J2ME 编写嵌入式应用？还是使用 J2EE 编写企业级应用？这些是我们那个泛黄的年代特有的烦恼。而现在的烦恼可能是学 Android？还是学 Java 后端？即便大方向你已经十分坚定，而且选择

了 Java 后端编程，但因为复杂的知识体系和 Google 发布的各种教程文档，眼前看到的已经不再是清晰的山脉，而是一片迷雾。此时，你需要本书，因为它给你指明了努力的方向。

本书的结构、阐述的方式和大部分的“指南”书籍有较大的区别，本书是以笔记和要点的形式进行呈现的，用现在的话说就是捞干货。本书涵盖的知识，是以现代工程实践中的实际案例出发来组织的，所以知识点范围非常广泛，每一个点都对最关键的“Best Practice”简明扼要地进行了说明。你在阅读本书的时候需要一些相关经验，不然无法跟上作者的节奏，建议在有一定的知识准备后再阅读本书，这样你会受益匪浅。从另外一个角度看，在你有了一定的基础积累之后，本书可以帮助你全面地了解一个现代化的最先进的工程实践是怎样的。本书讲述了目前行业中最常用的，经过了实践的工程方案，这将是快速进阶的最佳指引。

——孙建，随身云（中华万年历）联合创始人 & CEO

推荐序二

2013 年，我和本书作者的接触是从基于网易的一个大型互联网应用合作开始的，我见证了从第一行代码到整个系统服务于亿级用户的过程，并且相信这种经历对开发者来说是一笔巨大的财富，其中大量的开发和实战经验都会在本书中得到充分的体现，相信读者能从书中直接领略到丰富的实战知识。在与本书作者的合作过程中，其对 Java 技术的热爱与追求，对问题刨根问底，直到理解透彻、灵活应用，这些都令我印象深刻。这些年，我与本书作者一直保持沟通交流、相互学习，他将近十年的实战经验沉淀于本书以实现对后端技术的探索、布道，非常值得开发者与近高窗卧听秋。

后端技术涉及内容非常广泛，Java 语言也是互联网开发行业使用的主流语言，相信后续也将继续流行很长一段时间，而本书作者也一直从事 Java 后端开发工作。在本书中作者比较系统地从总体上描述了后端技术相关的理论知识，包括基础设施、网关服务及框架选型等基本原则，然后以实际经验进行示例说明，接着详细梳理了 Java 的后端技术，相信读者读完本书后会更全面地理解后端技术。互联网的业务建设需要不同角色的开发者共同协作完成，因此，系统工程化是开发者首先要共同遵守的规范或约定，包括代码规范、版本管理和代码质量检查等。

开发框架的选型进一步地为工程化提供了基础，也能加速推进互联网开发，尽管是否重复造轮子是一个恒久的话题，但是没有永远的银弹，只要在合适的时间里根据团队的能力选择合适的技术框架就好。一般来讲，目前常用的框架包括基本的依赖注入、AOP、事务管理、连接池管理、数据操作、日志服务等，在众多的框架中，本书作者选用目前在 Java 领域使用最广泛的 Spring 做深入的分析，详细地说明各组件的基础知识、基本原理和实际使用案例，最难得的是把较多开发者遇到的坑都用真实的示例进行了说明，可以帮助开发者快速地跳过这些伤心地带，同时也把最佳实践画龙点睛地带给开发者。

数据存储无疑是所有系统应用中非常重要的一环，应用的场景用例也和数据库的选型有着极其重要的关系，开发者选择关系型数据库还是非关系型数据库是需要根据软件成本与人力成本来进行权衡的，比如选择 MySQL、Oracle 等开源或商业的数据库。本书重点从数据库的基础知识、索引和表优化等方面以详尽的示例为更好地选择数据库的存储类型提供了更多的知识。

早期的关系型数据库一般能满足数据达到一定规模的企业的需求，而在互联网业务领域，特别是移动互联网领域内的元数据或者日志数据等，达到亿数量级别是很常见的，这时通常使用非关系型数据库，在非关系型数据库里使用非常多的有 MongoDB、HBase 等分布式数据库系统。作者在自身的企业开发实践中，得到了大量的使用经验和最佳实践。为了加速后端应用，缓存热数据是加速业务、提高业务性能、提升用户体验的重要手段，通过使用本地缓存、远程缓存进行数据加速、数据预热或提高数据的命中率，是开发者在应用开发的过程中常会遇到的场景。

“路漫漫其修远兮，吾将上下而求索”，后端技术每年都在不断发展，所用技术也有变化，近些年 Java 语言的发展速度不那么快了，但是总体是在不断前进发展的，本书作者带领的团队一直深耕此领域并希望通过本书为技术开发人员带来更多帮助。

——尧飘海，网易云基础服务（蜂巢）首席架构师

前言

目前互联网行业如火如荼，进入这个行业的技术人员也越来越多。对于研发来说，从工程角度其主要分为前端工程师、客户端工程师（又分为 iOS 和 Android 工程师）、后端工程师、算法工程师等职位。本书所说的 Java 工程师指的是以 Java 作为主要开发语言的后端工程师。

笔者从 2008 年还未毕业时做一些小的项目至今，做后端开发已经有差不多 10 年时间。经历过刚学 Java 时的迷茫，第一次写出 Java 程序时的激动，第一次写出一个 Web 系统的醍醐灌顶，一直到接触 Java 更底层的东西，总的来说对 Java 有了系统性的认识，对后端技术体系有了宏观的感受。这期间，笔者用过各种各样的编程语言，尝试过各种开源软件，挖过各种坑，也填过各种坑。针对后端技术来说，笔者认为自己的这些知识体系，还是有一定价值的。

此外，还记得当笔者毕业后进入第一家公司时，入职培训的课程虽然不难，但确实有种恍然大悟的感觉。业界的最佳实践和自己在学校里学到的、使用到的知识，差别还是非常大的。直到后来加入现在的这家公司，给新老员工做过一系列后端技术的培训课程，在校招的笔试和面试过程中深刻体会到学校与业界脱节之严重，在平时的社招中遇到很多对后端技术缺乏系统性认识、技能点不足的工程师，并且也经常被人问起如何学习 Java 后端技术，于是就打算将目前后端工程师一些比较主流、前沿的技术以及实际工作中会用到的技能串联起来，给刚上大学以后打算以 Java 后端为职业的学生、刚毕业入职的应届生以及初学者们一些入门的指引，使其少走弯路。另外也希望给一些有经验的工程师提供一个参考手册，将零散的知识点串起来，减少在解决某些实际问题时无头绪搜索带来的时间成本，同时也是对自己的一个阶段性总结和查漏补缺。需要注意的一点是，像数据结构、计算机网络等计算机科学基础知识以及 Java SE 的基本用法，笔者认为是从从事程序开发工作的 Java 工程师应该必备的知识，因此并不包括在内。

本书会针对 Java 后端开发工作中经常用到的关键技能点做阐述，会尽量覆盖实际工作中需要的所有技能。但由于很多技能并非一两个章节就能完整讲述，因此本书仅做一些实践性的经验总结和阐述，更加详细和深入的学习则需要参考专门的书籍或者官方文档。

本书的大部分内容都来自笔者的博客以及平时工作、学习中的一些自我总结和笔记，记录了笔者进入这个行业以来的一些经验教训和思考。

面向读者

- 未入门或者刚入门的 Java 工程师

包括未来以 Java 后端开发为职业方向的在校学生、刚毕业入职的 Java 工程师以及未形成知识体系的 Java 工程师。这类读者通过阅读本书能够对 Java 工程师的必备技能有一个全局认识，逐步形成自己的 Java 技术体系。

- 有经验的 Java 工程师

有经验的 Java 工程师可以通过本书查漏补缺，巩固自己的开发技能，进一步完善自身的 Java 技术体系。

- 对 Java 后端开发感兴趣的非 Java 工程师

非 Java 工程师可以通过本书了解 Java 工程师的技能体系，尤其对于其他语言的后端工程师来说，本书的很多内容也是通用的，并不局限于 Java 开发。

内容概览

- 第 1 章 后端技术导言

本章主要从总体上描述后端技术的概念、组成、作用、需要的知识点，并给出了学习后端技术的建议。

- 第 2 章 Java 项目与工程化

本章主要讲述 Java 项目与工程化需要掌握的软件、技能等。

- 第 3 章 开发框架

本章主要讲述 Java 后端开发中的一些主流框架的使用方法。

- 第 4 章 Spring

本章主要讲述 Spring 核心、数据操作以及一些常用组件的使用。

- 第 5 章 数据存储

本章主要讲述 Java 应用中数据存储上使用的一些软件、服务等。

- 第 6 章 数据通信

本章主要讲述 Java 应用中数据传输、通信上使用的一些软件、服务等。

- 第 7 章 Java 编程进阶

本章主要介绍一些 Java 开发中的高级特性以及在 Java 开发中非常流行的类库。

- 第 8 章 性能调优

本章主要讲述如何对 Java 应用的性能进行分析和调优，并给出了开发建议。

- 第 9 章 安全技术

本章主要对 Java 开发中常用的加密技术、HTTPS 以及防范各种攻击的方案做了阐述。

参考资料

在写作本书的过程以及平时的工作中，笔者阅读、参考过很多书籍，以下是其中具有代表性的一些书籍。对于本书讲述不够深入的地方，可以参考这些书籍进一步学习。

- 《Effective Java（第 2 版）》：此书讲解了 Java 的一些高级特性和技巧。
- 《Java 并发编程实战》：此书是并发编程经典书籍，涵盖了并发编程的各种知识点以及相关理论知识。
- 《七周七并发模型》：此书讲述了主流的 7 种并发编程模式。
- 《深入理解 Java 虚拟机：JVM 高级特性与最佳实践（第 2 版）》：此书讲解了 JVM 的内存、GC、字节码、编译器等高级特性和优化实践。
- 《高性能 MySQL（第 3 版）》：此书讲述了 MySQL 各种优化技巧，并结合原理给予讲解。
- 《Redis 开发与运维》：此书在原理层面对于 Redis 的使用、优化做了详尽的描述。
- 《深入理解 Elasticsearch（原书第 2 版）》：此书讲述了对 Elasticsearch 的使用、原理和优化技巧。
- 《Java 性能权威指南》：此书是 Java 性能调优的权威书籍，几乎涵盖了 Java 调优的方方面面。

- 《构建高性能 Web 站点（修订版）》：此书从各种案例出发，讲解了高性能 Web 站点需要的各种优化技巧、实践经验等。
- 《白帽子讲 Web 安全》此书基本涵盖了 Web 安全技术的方方面面，包括客户端安全、服务器端安全等。

虽然以上书籍都是非常实用的参考资料，但就笔者自己来看，更为推崇的则是直接通过相关技术的官方文档来学习，这样既能够锻炼自己的英文阅读能力，又能够直面相关技术的第一手文档，避免了在看相关书籍时被一些偶然的纰漏所误导。

此外，上面的《Effective Java（第 2 版）》和《Java 并发编程实战》这两本书都是基于 Java 的旧版本来写作的，但是里面介绍的很多内容并不过时，尤其是 JDK 底层源码、设计理论、优化思想等仍然适用于现在的 Java 开发。

勘误和支持

在本书的写作过程中，笔者一直是战战兢兢的，一直害怕写成那种侃侃而谈却没有实质内容的东西或者传递给读者一些误导信息，因此对于每一个知识点，都是在查阅官方文档以及其他权威资料并经过自己深入思考之后才敢落笔的。但由于笔者知识能力有限，难免有错误和疏漏，希望得到各位读者的理解和指正。

如果在阅读本书的过程中发现错误，请提交到网址 <https://github.com/superhj1987/pragmatic-java-engineer/issues> 或博文视点官网本书页面。

同时，请随时注意勘误信息的发布：<https://github.com/superhj1987/pragmatic-java-engineer/wiki/Mistakes>。

致谢

由于工作以及个人身体等方面的原因，中间数次延期，历时一年多才完成本书。因此首先要特别感谢笔者的父母和妻子，在笔者写作本书的过程中给予了非常大的后勤支持和鼓励，让笔者能够专心地完成写作。

同时要感谢中华万年历的同事们在平时的工作中给了笔者很多启发和思路，感谢公司的设计总监张喜亮抽出时间帮助修饰了一些图片，尤其要感谢 CEO 孙建在本书写作过程中给予笔者了充分的信任和支持。

还要感谢笔者的前同事，也是笔者刚毕业时的工作导师尧飘海，他在百忙之中审阅了本书并给本书写序；也要感谢前同事张小川、阙杭宁和好友秦绪震、饶洵，他们抽出了宝贵的业余时间校对了本书并给出了很有价值的建议。

最后，感谢电子工业出版社永恒的侠少，他找到我出版本书，并允许我一次次延期。也感谢付睿编辑的辛苦校对和修改，让本书得以顺利出版。

也把本书献给我刚出生的女儿——依依。

联系方式

邮箱: superhj1987@126.com

博客: <http://rowkey.me>

微博: <http://weibo.com/superhj1987>

读者服务

轻松注册成为博文视点社区用户（www.broadview.com.cn），您即可享受以下服务。

- **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/33501>



目录

第 1 章 后端技术导言	1
1.1 后端基础设施	2
1.1.1 请求统一入口——API 网关	3
1.1.2 业务应用和后端基础框架	4
1.1.3 缓存、数据库、搜索引擎、消息队列	5
1.1.4 文件存储	6
1.1.5 统一认证中心	6
1.1.6 单点登录系统	7
1.1.7 统一配置中心	7
1.1.8 服务治理框架	8
1.1.9 统一调度中心	9
1.1.10 统一日志服务	10
1.1.11 数据基础设施	10
1.1.12 故障监控	13
1.2 Java 后端技术概览	14
1.2.1 软件开发的核心原则	15
1.2.2 软件开发的过程管理	15
1.2.3 日常开发常用工具	16
1.2.4 应用的运行环境	16
1.2.5 常用第三方服务	17
1.2.6 计算机基础科学知识	18

1.2.7	数据处理相关技能	19
1.2.8	Java 编程知识	21
1.2.9	系统架构演化	22
1.2.10	典型的部署架构	23
1.3	如何学习后端技术	24
1.3.1	扎实的计算机基础知识	25
1.3.2	知其然更要知其所以然	26
1.3.3	动手实践	26
1.3.4	频繁练习	26
1.3.5	持续学习	27
1.3.6	自我总结	27
1.3.7	如何学习一门新技术	28
1.3.8	小结	29
第 2 章	Java 项目与工程化	30
2.1	项目构建	31
2.1.1	传统构建工具——Ant	31
2.1.2	主流构建工具——Maven	34
2.1.3	新兴构建工具——Gradle	43
2.2	代码版本控制	46
2.2.1	集中式代码版本管理——SVN	46
2.2.2	分布式代码版本管理——Git	49
2.2.3	提交日志的规范	55
2.3	代码质量保证	58
2.3.1	使用单元测试保证代码质量	59
2.3.2	衡量单元测试的标准	66
2.3.3	开发规范与建议	67
第 3 章	开发框架	70
3.1	依赖注入	71
3.1.1	JSR-330 依赖注入规范	73
3.1.2	Guice	75

3.1.3	PicoContainer	76
3.1.4	Dagger	76
3.1.5	Spring Framework	77
3.1.6	循环依赖问题	79
3.2	对象关系映射	79
3.2.1	表元数据的映射	80
3.2.2	CRUD 以及属性的查询	82
3.2.3	查询缓存的使用	83
3.2.4	结果的映射	84
3.2.5	规范 SQL 书写的语句构建器	84
3.2.6	使用提示	85
3.3	日志	86
3.3.1	JDK Logging	87
3.3.2	Log4j	88
3.3.3	Log4j2	90
3.3.4	Logback	93
3.3.5	统一日志 API 的门面框架	95
3.3.6	统一日志框架的使用	98
3.4	Web MVC	99
3.4.1	为什么是 Spring MVC	99
3.4.2	Spring MVC 的请求处理流程	100
3.4.3	典型的配置方式	102
3.4.4	无 XML 的配置方式	105
3.4.5	对 MVC 应用做单元测试	106
3.4.6	验证 Web 请求的参数	107
3.4.7	使用异步 Servlet	110
3.4.8	使用提示	112
第 4 章 Spring		115
4.1	Spring 核心组件	117
4.1.1	Spring 的双亲上下文机制	118
4.1.2	Spring 中的事件机制	119
4.1.3	Bean 的初始化和销毁	120

4.1.4	Bean 的动态构造	122
4.1.5	注入集合、枚举、类的静态字段	124
4.1.6	面向方面编程——AOP	125
4.1.7	进阶 XML 的配置	130
4.1.8	无 XML 的配置方式	133
4.2	Spring 数据操作框架	135
4.2.1	Spring JDBC	135
4.2.2	Spring Data Redis	136
4.2.3	Spring Data MongoDB	138
4.3	Spring Boot	140
4.3.1	Spring Boot 使用示例	140
4.3.2	Spring Boot 的运行原理	141
4.3.3	Spring Boot 的组成模块	143
4.3.4	小结	144
4.4	Spring 常用组件	144
4.4.1	表达式引擎——Spring Expression Language	144
4.4.2	远程过程访问的支持——Spring Remoting	145
4.4.3	Spring 与 JMX 的集成	146
4.4.4	定时任务的支持——Spring Quartz	147
4.4.5	跨域请求的支持——Spring CORS	148
4.5	总结	149
第 5 章	数据存储	151
5.1	关系型数据库——MySQL	152
5.1.1	存储引擎	152
5.1.2	字符集和校对规则	153
5.1.3	索引的使用	154
5.1.4	查询缓存的使用	158
5.1.5	数据同步中的 Binlog	159
5.1.6	事务机制	159
5.1.7	大表优化	163
5.1.8	高可用支持	164
5.1.9	使用提示	166

5.2 非关系型数据库	169
5.2.1 KV 数据库	170
5.2.2 文档数据库——MongoDB	171
5.2.3 列数据库——HBase	181
5.3 缓存	185
5.3.1 本地缓存	186
5.3.2 分布式缓存——Redis	188
5.3.3 缓存设计的典型方案	195
5.4 搜索引擎——Elasticsearch	196
5.4.1 开源全文检索库——Apache Lucene	197
5.4.2 关键概念	198
5.4.3 查询的优化	199
5.4.4 内存的使用优化	201
5.4.5 开源日志管理方案——ELK	202
第 6 章 数据通信	204
6.1 RESTful 架构风格	204
6.1.1 支持的操作	205
6.1.2 返回码	206
6.1.3 资源概念	207
6.1.4 数据的安全保障	208
6.1.5 请求的限流	210
6.1.6 超文本 API	211
6.1.7 编写文档	211
6.1.8 RESTful API 实现	211
6.2 远程过程调用——RPC	212
6.2.1 JDK 自带的 RPC——RMI	213
6.2.2 Hessian	213
6.2.3 Thrift	214
6.2.4 Dubbo	216
6.2.5 数据的序列化机制	217
6.2.6 使用提示	222