

结束语 | 愿你能将Dubbo所学迁移至其他微服务框架

2023-02-22 何辉 来自北京

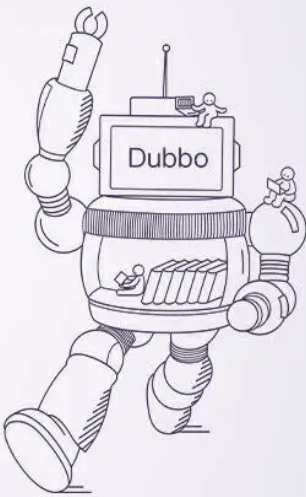
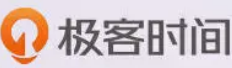
《Dubbo源码剖析与实战》

课程介绍 >



何辉
平安壹钱包架构师

你好，我是何辉。
有术无道止于术，有道无术术尚可求。纯有技术但不会思考，随着时间的推移，自然会固步自封，有思维的加持，你才能在技术之路上走得更高更远。



讲述：何辉
时长 07:00 大小 6.40M



你好，我是何辉。

首先，恭喜你完成了这门课程！

从课程一开始，我就非常强调要抓主干、重思考，在课程的学习过程中，也带你从常见问题一步步深挖，完成了很多次源码探索。不知道你有没有好奇过，为什么会这么设计学习思路？

其实这和我的 Dubbo 学习经历有关。我第一次接触 Dubbo 的时候，完全是懵圈的状态，当时整个项目在往 Dubbo 转型，但是之前我没接触过这个框架，对 Dubbo 的各种参数设置、各种高级特性，几乎是一窍不通，在陌生的 Dubbo 微服务框架体系中，我每天以赶鸭子上架的状态，迭代需求。

领资料

时间紧，项目急，这是我们 IT 行业的常态。面对 Dubbo 框架中需要迭代的第一个需求，很多人求快，选择火急火燎地投入到编码开发中，但是这种方式往往是最慢的，也是最后患无穷

的。

虽然没有 Dubbo 开发经验，但是有着丰富项目经验（踩坑经验）的我，在理解清楚需求后，选择花很多时间，查阅其他系统分支代码，希望找到成功的 Dubbo 相关配置案例代码，找到相对正确的编码开发姿势。

花了大半天，我看了公司十几个系统的代码，主要关注三个方面。你想迅速了解一个系统，也可以参考这个思路。

1. 暴露的接口（即各种 XxxFacade）以及接口在系统中被哪些地方使用了。
2. 调度的入口以及触发调度的配置。
3. MQ 的入口以及监听 MQ 的配置。

针对第一个方面，我主要想看，定义一个对外暴露的接口，别的系统一般是怎么编码的，结果发现都是“接口 + 接口实现类 + XML 配置”这种组合的方式，然后在实现类中通过“@DubboReference”或“XML 配置”方法进行远程调用，而且这些都是跑在线上的代码，所以我至少已经找到了一种暴露接口提供服务的正确姿势。

针对第二、三方面，主要是想看系统集成 Dubbo 之后，这些常用的功能是否有什么特别之处，如果有，细究到底有哪些不同，然后记下来，避免自己开发需求时遗漏了。

经过这一番成功的案例代码寻找后，我从原本惧怕 Dubbo 框架，突然变得非常有底气，虽然一行代码都还没写，但是一种已经征服了 Dubbo 框架的自信感油然而生。

在之后常态需求迭代开发过程中，凡是编码过程遇到和 Dubbo 有关的技术知识点，我都会记录下来，需求完成后，自己会去查阅资料仔细研究，久而久之，迭代几个版本后，就摸清了 Dubbo 框架的整个体系。

在研究 Dubbo 知识点的时候，就像我们在课程中反复强调的，我并没有一股脑钻进源码中深究原理，而是会分成三步。

- 先思考，这个技术知识点有什么用，到底带来了什么便利性，又或者帮我们解决了什么问题。



- 然后，我会站在框架设计者的角度去思考，怎么设计功能并解决这样的问题，自己尝试一步步推出解决方案。
- 最后，才深入源码，边探索，边思考和自己设计方案之间的偏差，逐渐理解框架设计者的意图。

这也是，我在开篇词中提到的“重思考、重推导、重理解”。

啃源码的过程也是有技巧的，千万别一头扎进各种细节中，先大致翻阅一下，按照你自己的理解，梳理出代码的主干流程，在阅读主干流程代码的同时，体会流程所要表达的思路，这也是我在开篇词中说的“抓体系、抓主干、抓思路”。

在阅读源码的过程中，你肯定会冒出十万个为什么，别着急，这正是你发现问题的时候，也是你所看到的**问题表象**。

针对这些表象问题，你可以尝试站在全局的角度，分析这个问题产生的原因，然后用 Debug 方式来调试源码，验证你的猜想，逐步找到答案，在分析过程中你会逐渐找到问题的**本质特征**。

最后，从这些本质特征中，你要尝试提炼出方便自己理解的、稳定的**不变体系**，之后在另外的场景中，如果你发现了意外的变化，也就发展成了新的**问题表象**。

整个过程，你会发现你一直处在**表象 -> 本质 -> 不变 -> 表象**这个循环中，从表象中深挖问题的本质，从本质中提炼出不变的体系，依据不变的体系，在不同场景，你又会发现意外的变化，对比相似的知识点，分析变化的因素，你会发现新的问题表象。最终你会分类、提炼、总结出自己的知识体系，形成自己的知识结晶。

不止 Dubbo，这么反复捣鼓几遍后，你再去看其他框架，会觉得特别容易上手。

- 当你理解了“参数验证”“事件通知”“缓存操作”“流量控制”的核心思想，再在 Spring、Mybatis、ElasticSearch 框架进行统一拦截扩展时，你会非常得心应手，无非就是想办法找到适合的拦截必经之路。
- 当你理解了“实例注入”的核心流程，再去研究 Spring Bean 从初始化到实例化的过程，你会发现套路都是相似的。

领资料

- 当你理解了“集成框架”中的扫描器机制原理，再去研究 Mybatis 扫描 Mapper、Seata 扫描 TCC 模式，会觉得原来扫描自定义注解如此简单方便。
- 当你理解了“调用流程”的核心思想，再去深入研究 Tomcat 接收数据流程、SpringMvc 接收请求流程等，会发现都是相似的味道。

总的来说，**术固然重要**，这是我们日常开发必不可少的基本技能，**但是随着你的工龄增长，道会变得越来越重要**，因为道能让我们走得更远，正所谓“**有术无道止于术，有道无术术尚可求**”。纯有技术但不会思考，随着时间的推移，自然会固步自封，有思维的加持，你才能在技术道路上走得更高更远。

我想再送你一句话：“将来的你，一定会感谢现在努力拼搏的你”。估计很多同学都在默默学习，努力加油，但在专栏要结束的今天，我非常希望能在留言区听到你的声音，听听你学习这个专栏的感受和收获（戳 [🔗 这里](#) 填写你对课程的反馈与建议）。

感谢你选择我的《Dubbo 源码剖析与实战》。感谢你陪我们一路走到这里。

接下来，就看你的了。

分享给需要的人，Ta 购买本课程，你将得 **18** 元

 生成海报并分享

 赞 5  提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

[上一篇](#) 加餐 | 大厂高频面试：底层的源码逻辑知多少？

领资料

更多课程推荐

李三红·搞定 Java 开发基础

极客时间 × 阿里云开发者社区联合出品

李三红
阿里云程序语言
与编译器技术总监
Java Champion



免费订阅



精选留言 (4)

写留言



杨老师

2023-03-09 来自北京

感谢老师，第一遍过了一下，等第二遍开刷了。

作者回复: 你好，杨老师：三克油，初看留个印象底子，望你复看探究出一片新天地~



Lum

2023-03-03 来自北京

感谢老师的陪伴，然后继续第二遍

作者回复: 你好，Lum：专栏一路有你陪伴不孤独，望你再探从前绽芬芳~



领资料



熊悟空的凶

2023-03-03 来自北京

需要沉下心来，再撸第二遍；书读百遍、其义自见

作者回复: 你好，熊悟空的凶：沉的下心，吃的了苦，耐得住寂寞，何愁大业不成~



1



豪

2023-03-01 来自上海

多谢老师的陪伴，第一遍学的磕磕碰碰，准备二刷了

作者回复: 你好，豪：初看有感磕磕碰碰，再看望你豪情万丈。春风十里我在极客时间等你，加油@_@



1

领资料

