



下载APP



开篇词 | 与我一起修炼，成为一名软件性能大师吧！

2021-05-17 尉刚强

性能优化高手课

[进入课程 >](#)**讲述：尉刚强**

时长 12:06 大小 11.09M



你好，我是尉刚强，一名软件设计咨询顾问。从今天开始，我会带你深入软件性能优化的探索之旅。

从业十多年来，我先后在通信领域、大数据领域、人工智能对话领域、SaaS 服务等领域的一线工作，主要解决软件架构设计与性能优化上的问题。同时，我也在为国内一些知名企业提供软件设计咨询服务。

正因为我跨领域的项目性能优化经历，在寻找性能优化思路时，我的视角往往是多维度的，并由此总结出了一套从设计到交付、兼具多个领域优势的全流程性能优化方案。比如，之前我在 SaaS 服务性能优化项目中，所使用的编解码、实时处理任务等高性能实现方案，就是借鉴了嵌入式系统上的优化经验。



你可能会说，老师，我觉得快速编码交付业务的功能，才是最有意思的，为啥要学习一整套优化方案呢？

说实话，我刚开始工作的时候，也是一个只关注软件设计与开发的工程师，和你想的一样。不过后来的一次项目开发经历，让我发现，原来软件的性能也可以从设计和开发的视角去实现和交付，而且**从软件设计与架构维度出发的性能优化，不仅性能收益更大，而且还会让软件架构设计更加合理和简洁。**

支撑软件性能长期保持竞争力的道理其实很简单

事情其实是这样的：那时我还在一家通信企业工作，带领功能交付团队，与另一个专门负责性能攻关和优化的团队一起协作，来支撑子系统版本的交付。不过，在协作的过程中，因为我们两个团队互相不理解对方的工作，产生了不少矛盾和冲突。所以，后来，为了缓解双方的矛盾，领导让我去负责性能优化攻关团队的工作。

可是，由于前期性能优化团队主要是基于编译优化、测量分析手段来进行性能攻关和优化的，已经将代码实现修改的性能优化空间挖掘殆尽了，所以，我在负责性能优化工作时，就只能另辟蹊径。

而我们应该都知道，软件架构设计通常来说对性能的影响很大，况且当时我们面临的困境确实是无法再从代码实现层来进行调优了。

由于我之前一直负责软件设计与开发，非常清楚业务的软件设计架构，所以，我就选择从软件设计与架构的视角出发，通过不断调整设计架构，来尝试优化性能。比如说，根据业务使用内存的特性，实现定制化内存分配器，以提升内存的申请和释放效率；优化业务代码逻辑，实现核心流程中的内存零拷贝机制，等等。

由此，在这个性能优化的过程之中，我真的就找到了破局之道。最终，这个性能优化项目在实现代码规模量减少、设计更加优雅的同时，性能上也有较大幅度的提升。

再后来，我参与的架构设计与实现的软件越来越多，在不同领域下解决的性能优化问题也越来越多，而我的这些经历也在不断印证着一个道理，就是**虽然不同领域的业务知识差异比较大，但其背后支撑性能设计与优化的原理与方法论，却是统一的。**

就比如说，SaaS 服务领域的性能优化中使用延迟计算，与嵌入式领域的延迟计算优化的思想是一致的；智能对话引擎基于消息回复时延的设计方法，也跟实时性系统时延的设计方法是统一的；类似的还有消息队列设计原理、选型设计，等等。

所以到了这里，我才真正理解和摸清了**软件性能优化的关键命脉**，那就是要从软件设计阶段对性能进行建模设计，再在后续的编码实现、测试、维护等阶段进行层层控制，只有这样，才能真正做好性能优化工作。

为什么你学习了很多性能优化知识，却依然解决不了性能问题？

现在，我作为一名软件技术咨询顾问，有机会接触到更多的项目和团队。可遗憾的是，我在跟不同领域的研发团队配合的过程中，发现他们处理、解决性能问题的能力差距比较大。

很多团队还在使用低效的方式去处理解决各种性能问题，甚至还有很多团队没有系统化的性能设计与调优的能力，当碰到比较简单的性能问题时还能处理，可一旦碰到一些复杂、深层次的软件性能问题，就很容易陷入到僵局之中。

这是为什么呢？实际上，**传统的性能优化视角，更多的是从问题与测量数据的角度出发，是被动式地解决处理性能问题**，所以依据这样的视角驱动的性能优化工作，就会存在很多的局限性，比如说：

很容易造成代码可读性差，而且只是做到局部性的优化；

这种工作方式通常是在软件工程的后期才启动，发现问题会比较滞后；

一次性优化合入的代码量很大，导致软件版本发生质量问题的概率会比较大；

由于没有设计与数学理论的支撑，不知道性能优化目标在哪个范围内才是合理的。

而在这门课程中，我要给你介绍的性能优化，是**从整个软件生命周期的视角去关注与审视软件性能，通过主动式地设计与实现来主导软件性能，从而支撑软件性能长期保持竞争力**。

另外，这里我想告诉你的是，这门课程并不是一个工具技术类课程，你可以等有需要或是遇到具体问题再去翻找查看。这门课更像是一个思维训练类课程，它会站在系统端到端

的角度，分析从软件设计到工程发布上线这一整个过程中，如何有效保证软件性能一直处于可控状态。

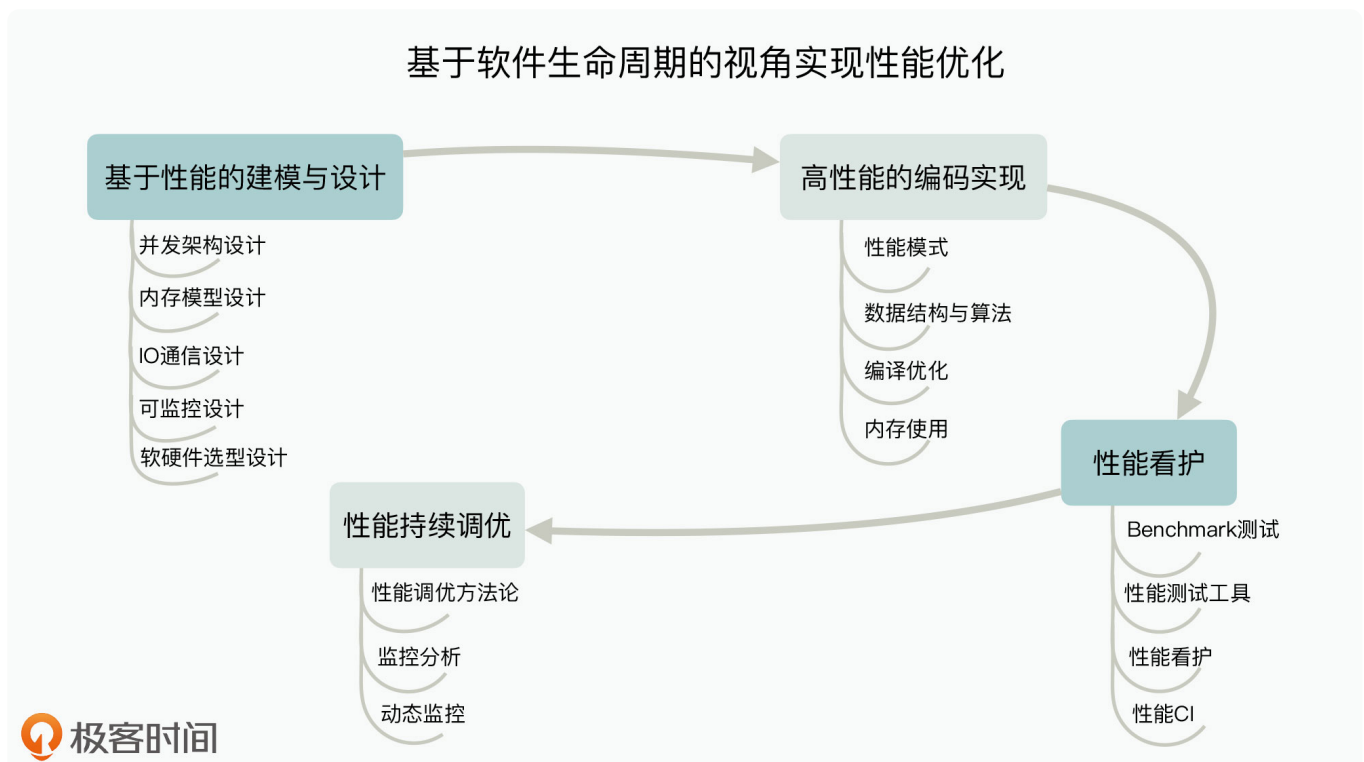
所以在学习的过程中，你需要和我一起去思考分析这样做的出发点，以及背后支撑的理论是什么，从而深入挖掘出全局的性能设计与优化的方法论，并提升软件性能工程的管理能力。

当然，也许有些内容与方法你暂时还用不到，但是当你在实际场景中遇到不同的性能问题时，它也可以启发你寻找到更系统的解决思路和方法。

好了，那么具体我是怎么设计这门课的呢？下面我就来给你介绍下课程的内容框架吧。

这门课是怎么设计的？

我把课程主体划分为了五大模块，分别针对基于性能的建模与设计、高性能的编码实现、性能看护和性能持续调优四个部分进行方法与实践的讲解介绍，让你能够明确地理解和掌握系统级的性能优化的理念和方法，最后还会通过一个案例模块，来帮助你提升具体实施性能优化方法的能力。



性能设计篇

这一模块，我会给你介绍几种在性能优化设计中非常关键、且经常用到的设计方法，包括并发架构设计、内存模型设计、IO 通信设计等。另外，我还会从性能扩展和调优的角度，给你讲解在软件设计阶段需要考虑的可监控设计、可扩展性设计、可移植性设计、软硬件选型设计，以此帮助你在实际的建模和设计过程中，支撑产品更好地实现性能需求。

性能实现篇

我会从编码实现的阶段入手，结合场景案例，带你了解和学习高性能编码的实现模式，让你能够在一些特定场景下选择合适的实现，来大幅度提升性能。另外，我还会带你深入理解各种数据结构与算法，并从使用最为广泛的 Java 语言着手，给你讲解在不同的场景下，使用不同的算法所带来的性能差异，这样你在面临不同的业务问题时，就知道如何选择合适的算法来提升性能了。

性能看护篇

软件是一个需求在不断变化的产品，只有持续地看护性能，才能保证软件性能一直保持在可控的状态。所以，我会从性能测试前置这一核心的性能看护出发点着手，带你深入产品的组件级、系统级的多层级性能测试，让你在理解性能看护核心理念的基础上，清楚地知道该如何实现性能测试用例自动化，并能够集成到流水线中，这样当你在提交代码时，就可以第一时间发现性能腐化问题。

性能调优篇

软件产品在交付运行后，仍需要持续监控系统的运行状态，支撑持续的性能调优。因此，在这一模块，我会给你分享一套性能调优的通用方法论，包括性能问题分析与定位的方法技巧，帮助你在实际的业务场景中，快速找到受限于性能瓶颈的资源，持续地监控分析目标。

案例篇

在实际的业务领域中，你可能会面对形形色色的性能问题，而要想快速解决它们，你需要考虑两方面的问题：如何选择合适的性能优化方法？怎样才能更好地实施性能优化方法？所以，在案例篇中，我会为你讲解多个在真实项目中，实施性能优化的操作过程，帮助你提升落地性能优化方法的能力和技巧。

《性能优化高手课》课程目录

- 开篇词 | 与我一起修炼，成为一名软件性能大师吧！

性能设计篇

- 01 性能建模设计：如何满足软件设计中的性能需求？
- 02 并行设计（上）：如何利用并行设计挖掘性能极限？
- 03 并行设计（下）：如何高效解决同步互斥问题？
- 04 缓存设计：做好缓存设计的关键是什么？
- 05 IO 设计：如何设计 IO 交互来提升系统性能？
- 06 通信设计：请不要让消息通信拖垮了系统的整体性能
- 07 数据库选型：如何基于性能需求选择合适的数据库？
- 08 可监控设计：如何利用 eBPF 来指导可监控设计？

性能实现篇

- 09 性能模式（上）：如何有效提升性能指标？
- 10 性能模式（下）：如何解决核心的性能问题？
- 11 如何针对特定业务场景设计数据结构和高性能算法？
- 12 我们要先实现业务功能，还是先优化代码？

13 编译期优化：只有修改业务代码才能提升系统性能？

14 内存使用：如何高效使用内存来优化软件性能？

15 并发实现：掌握不同并发框架的选择和使用秘诀

16 技术探索：你真的把 CPU 的潜能都挖掘出来了吗？

性能看护篇

17 Benchmark 测试（上）：如何做好微基准测试？

18 Benchmark 测试（下）：如何做好宏基准测试？

19 性能测试工具：如何选择最合适的性能测试工具？

20 性能看护：如何更好地守护产品性能？

21 性能 CI：性能测试也可以集成到 Pipeline 中吗？

性能调优篇

22 性能调优方法论：如何科学高效地定位性能问题？

23 监控分析：你的性能调优工具足够有效吗？

24 动态监控：你的产品系统中有动态监控的能力吗？

25 性能调优什么时候应该停止？

案例篇

26 一个嵌入式实时系统都要从哪些方面做好性能优化？

27 解决一个互斥问题，系统并发用户数提升了 30 倍！

28 Web 服务业务代码一行不动，性能提升 20%，怎么做到的？

结束语

■ 结束语 | 千里之行，始于足下，你已踏上修炼之道！

结语

最后，我希望你在学习的过程中，能够具备批判性的思维，结合自己的知识积累和实践经验，来理解和吸收课程内容，并且你也可以给我多多留言，阐释久悬不决的疑问、分享见解独到的观点，我都会给予解答和回复。

这样，当你学完了课程之后，通过积极地思考学习和输出，你也可以建立起来一套关于软件性能的全局系统认识。今后，当你碰到各种性能问题时，你都可以从软件设计、软件实现、性能看护、性能调优等多个维度，进行系统性的思考，找到合适的解决方案。

好了，接下来我们就开始这段软件性能优化的探索之旅吧！

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。

下一篇 01 | 性能建模设计：如何满足软件设计中的性能需求？

精选留言 (3)

写留言



落叶之棠
2021-05-21

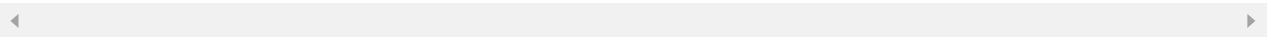
希望学习之后，可以解决现在的一些痛点
展开



danger boy
2021-05-19

非常喜欢这个课程，老师你最棒！
展开

作者回复: 谢谢，我继续努力，还会有更精彩的内容奉上！



Geek_9c3134
2021-05-17

老师 性能调优这个行业怎么样
展开

作者回复: 性能调优这个行业范围比较广，看你聚焦在那个领域上了。一般情况下，在一个大型企业内会比较吃香！

