

二

00 Java 性能优化，是进阶高级架构师的炼金石

你好，我是李国。作为《Java 性能优化与面试 21 讲》这个课程的作者，我先来简单介绍一下自己。

我曾任京东金融、陌陌科技高级架构师，工作期间，我接触的都是比较底层的中间件和操作系统，涉及大量高并发场景下的调优工作，比如缓存优化、多线程优化、JVM 调优等。因此，我在高并发下的性能优化方面积累了丰富的实践经验，同时积累了一套自己独有的优化思路和优化技巧。我曾经优化过一个运行缓慢的复杂业务，单机 QPS 由原来的 2k/s 提升到了 2w/s，整个集群 QPS 达到近 100w/s。

随着互联网的发展，高可靠、高并发以及降本增效，已成为各大公司面临的现实挑战，性能优化需求愈发迫切，大到分布式系统，小到代码块的算法优化，都已经成为你日常工作中必须要面对的事情。对于开发者而言，性能优化也从加分项变为一个热门技能，缺乏相关知识将很难在面试或工作中脱颖而出。

性能优化有哪些困扰

但是作为过来人，我发现很多学习者和实践者在 Java 性能优化上面临着很多的困扰，比如：

- 工作场景中遇到“性能优化”难题，往往只能靠盲猜和感觉，用**临时性的补救措施**去掩盖，看似解决了问题，但下次同样的问题又会发作，原因则是**缺乏方法论、思路的指引，以及工具支持**；
- 能力修炼中，由于常年接触 CRUD，缺乏高并发这一实践环境，对“性能优化”只能通过理论知识进行想象，**无法认识其在工作实战中的真实面目和实操过程**；
- 职场晋升中，**只管功能开发，不了解组件设计原理，缺少深入地思考与总结**，无法完成高并发、高性能系统设计这类高阶工作，难以在工作中大展拳脚，而有挑战的工作往往留给有准备的人。

总之，一旦遇到“性能优化”问题，很少人能够由点及面逆向分析，最终找到瓶颈点和优化方法，而**性能优化是软件工程的深水区，也是衡量一个程序员能力高低的标准**。

进行 Java 性能优化的关键

俗话说，知己知彼百战百胜，想要克服“性能优化”这一难题，先要了解性能优化的特点，并抓住其关键和本质。

作为面试必考内容，很多应聘者反映说面试官的一些问题会让其陷入模棱两可的境地，不知如何作答，比如很多人就搞不懂缓冲与缓存的区别。这种问题的答案，只能靠体系化的整理，依靠零零散散的知识是行不通的。**你需要具备触类旁通的能力**，才能对面试的散点知识既有深度又有广度地做进一步升华，才会让面试官眼前一亮。

性能优化是个系统性工程，对工程师的技术广度和深度都有要求。它不仅需要你精通编程语言，还需要深刻理解操作系统、JVM 以及框架原理的相互作用关系，需要你多维度、全方位地去分析排查。

此外，很多人能够遇到问题解决问题，但救火式治理只能临时补救表面问题，无法真正找出病灶，这次的解决只是为下次发作埋下了伏笔。事实上，很多性能问题往往隐藏的很深，比如，spring-aop 所引起的性能问题就比较难以排查。

再比如，有人细致到会关注 switch 语句速度快还是 if 语句快，但并不能真正解决性能问题。原因是什么呢？他虽然做了“性能优化”这个动作，但思路方向却错了。这种极细微级别的优化对性能提升的影响面是很小的；而且，细节上极度地追求性能，反而会把代码写得晦涩难懂，难以维护，导致最后舍本逐末。其实，**性能优化更多要求我们关注整体效果，兼顾可靠性、扩展性，以及极端的异常场景，这样才能体现性能优化的价值。**

实践比理论重要。性能优化并不是对固定、单一场景的优化，场景不同，方法也会不同。比如，如果你的业务是串行的，耗时很长，就不能简单地通过增加 CPU 资源进行性能提升；如果你的业务是并行的，也不能钻牛角尖地优化每一行代码，要照顾各个资源的协调，对短板着重进行优化，以便达到最优效果。

在过去你面临以上情况时，可能会仅凭感觉入手，或者先动手才思考，无法发现抓住本质，但在本课程中，我会向你讲解正确的思路，让你进行性能优化时有理可依。

课程设计

在这个课程中，我汇总了 Java 性能优化的经典案例，结合大量代码示例，尽力为你还原真实的业务场景。

课程分为 5 个模块，共 21 篇，我将从理论分析、工具支持、案例与面试点，以及 JVM 优化四大方面展开系统讲解：

- **模块一：理论分析**，针对平常对性能优化的盲猜问题，我们会首先讲解大量的衡量指标，然后以此为依据，盘点一下常用的优化方法，包括业务优化、复用优化、计算优化、结果集优化、资源冲突优化、算法优化、高效实现等方面。学完后，你将会了解如何描述性能，并对性能优化有个整体的印象。
- **模块二：工具支持**，工欲善其事，必先利其器。此部分将介绍一些评估操作系统设备性能的工具，包含大量实用的命令行解析；还会介绍 Java 中最有效的基准测试工具 JMH，以及一些监测 JVM 性能的应用。本模块的目的，是为大家提供一些测量性能的工具，为实践环节做准备。
- **模块三：实战案例与高频面试点**，该模块为课程的主要内容，结合之前模块的理论分析和工具支持，通过海量实战案例，深入专项性能场景，并将每个场景下的高频面试点逐一击破，点拨调优思路，目标是能够做到举一反三，在遇到相似的性能问题时，能够快速想到合适的切入点进行优化。
- **模块四：JVM 优化**，该模块对系统的性能提升是巨大的。本部分主要介绍垃圾回收的一些基本知识，看一下 JIT 在性能提升上所做的文章；最后列举了一些常见的优化参数，以及对编码方面的要求。学完本模块，你将掌握和 JVM 相关的常见优化措施。
- **模块五：特别放送**，最后，针对工作中最常用的服务和框架，我想和你介绍一个 SpringBoot 服务的优化案例，涵盖 Tomcat、Undertow、JVM、网络等场景，同时再进行优化方法和求职面经总结，希望以一个全局的案例，帮助你掌握从系统层到应用层的整个优化技巧。

你将收获

建立完整的性能优化知识体系。你可以系统地学习相关知识，而不是碎片化获取，基础理论实用性强，直入主题，让你在工作实战时有理可依，有据可循。

能够对线上应用输出优化思路。掌握各种实战排查工具，并灵活应用，定位至应用中的症结瓶颈点，并输出优化思路方案。正确的方法比努力更重要，有了正确的思路方法，才能在实际工作中避免跑偏，避免把大力气花在一些细枝末节上。我还会分享大量的操作系统方面的知识，让你对应用性能有更好的评测。

收获海量实战经验分享。作为这门课最硬核内容，我将从流行的中间件介绍到常用的工具类，再到 JDK 中的知识点，用实战分析和经验分享高度还原真实的业务场景，带你了解性能优化的全过程。

获得面试 Offer 收割利器。本课程的大多数案例，都是 Java 面试题的重灾区，我将直接指出高频考点，让你既能在整体上对性能优化提供建议，也能深入细节进行针对性优化。

讲师寄语

最后，**性能优化既是工程师们进阶的“拦路虎”，也可以是你能力的炼金石**。希望这个专栏可以让 这个非常难啃的老大难问题，变得“平易近人”“通俗易懂”“一点就通”，希望可以让你体会到“哦，原来如此简单！”的感觉，体会到久违的学习的快乐，并能学有所用。

另外，我去年就与拉勾教育平台合作了[《深入浅出 Java 虚拟机》（已完结）]课程，用户口碑还不错，Java 虚拟机这门课可作为 Java 性能优化课程的一个补充，我也推荐你去学习了解。

Java 性能优化对知识广度和知识深度都有比较高的要求，让我们掌握性能调优的思路，多多实践，使自己的编码水平更上层楼。

[下一页](#)