

结束语 | Java 程序员的成长之路和从业方向

首先，恭喜你，坚持学到了最后。

今天我想借最后这个机会，结合自己多年工作的经历和感悟，和你聊一下有关 Java 程序员的成长之路，以及几个从业方向的问题。

成长之路

我是 2014 年参加工作的，现在 2021 年，我已正式成为一名自由职业者。

回想自己这 7 年的工作经历，真的是发生了太多太多的事情。从后台到前端、从开发到运维、从数据到系统、从服务器到嵌入式、从技术到业务，编写程序、负责项目、担任架构师，一路走来，有太多的收获，也有太多的感触。

所以我希望结合自己这些年的工作经历，来跟你谈下 Java 开发人员的几个职业发展阶段。这既是对自己人生阶段的一个总结，也希望这些经验能够给后来的开发者带来稍微帮助。

到目前为止，我的职业生涯经历了四个阶段。

第一个阶段，是毕业后参加工作的第 1 年左右。在这个时期，我负责开发云数据库服务，这些服务全都部署在 Linux 云服务器上。这就要求我对数据库和 Linux 系统非常熟悉。这里我必须感谢自己在读书的时候就认真研究过 Linux 相关知识。对于数据库，我则重新详细学习了数据库工作原理方面的内容。

这些知识都对我的工作起到了巨大的帮助。我这里总结下我认为在这个阶段需要掌握的技能。

1. 软件迭代开发的流程。这对你以后进行高效的软件开发以及进行项目管理，都是非常重要的知识点。另外，学会 Git 版本管理，会大大提高你写代码和写文档的效率。
2. Java 编程基础。我推荐《Java 编程思想》这本书，这是我阅读过好几遍并且书脊都被翻裂开的书。
3. Java 软件开发的各种规范，比如编程规范、安全规范（密码、随机数、SQL 注入等）、软件版权等。
4. Spring 相关知识，包括 IoC、AOP、HTTP、Servlet、JSP、ORM、MVC 等。
5. Linux 相关知识，包括操作系统原理，和 Shell 工具及脚本。
6. Java 虚拟机。配合操作系统、编译原理等知识，理解 Java 虚拟机工作原理，这是当 Java 程序上线后出问题，你最有力的解决问题的武器。
7. 数据库工作原理。我推荐《数据库系统概念》一书，讲得很全面，让我全面地认识了 SQL 语法以及数据库工作原理。

第二个阶段，是工作 1 到 3 年的阶段。这段时期是我技术水平提升最迅速的时期。这门《实时流计算》课程，就是对我从这段时期开始主要工作内容的总结。当时，我的任务是开发一款风控产品。为了尽可能提高程序性能，充分榨干机器的 CPU 和 IO 资源，尽可能降低硬件成本，我进行了艰苦卓绝的调研和思考。

最终，我意识到 NIO 和异步才是问题的关键所在。并且，在 JVM 不支持纤程的情况下，“流”是最佳的异步编程模式。当意识到这两点后，我的思路从此豁然开朗，编程水平有了实质性的提升。我庆幸自己能够有这段经历。对于这段经历，我认为最重要的是以下几点。

1. 踏实做事，理解公司完整的业务流程、完整的系统架构，养成从全局考虑问题的习惯。这是架构师思维的起点。
2. 对于技术方案选型，一定要考虑全面。比如，很多同事写程序前几乎不会考虑性能、并发安全、OOM、数据过期淘汰等问题。这样的结果必然是一次又一次地线上崩溃，一遍又一遍地重写代码。

3. 遇到问题，不要轻易就放弃了，一定要想方设法解决它。在这个过程中，你会遇到非常非常多的问题，比如性能、网络、安全、运维等各种问题。但请你一定要去努力解决，因为每解决一个问题，都将会使你的技术得到进一步提升。
4. 并发编程的知识。我推荐《Java 并发编程实战》这本书（Java Concurrency in Practice），这又是一本我认真读过两遍的书，逻辑性强，可以好好品读。读完这本书后，并发对你而言会是件信手拈来的事情。
5. 网络的知识。网络拓扑、TCP/IP、Nginx 反向代理以及负载均衡等网络相关知识，绝非只是运维的事情，而是一名合格架构师必备的知识。比如我们公司的 CTO，他经常只用一个 tcpdump 工具，几分钟就解决了我们几个开发人员半天都解决不了的问题（比如 Kafka 集群连不上）。这点让我获益匪浅。
6. 大数据相关知识。包括 Hadoop（HDFS/MapReduce/YARN）、Spark、Flink、Hive、Flume 等。这些分布式技术，会极大地扩展你的技术视野，并能够给你提供更多的架构设计思路。千万不要觉得 Spring/Spring Boot/Spring MVC/Spring Cloud 就是 Java 的全部了。
7. 运维的知识。包括 DevOps 思想和 Docker/Kubernetes/Rancher。这是你以后成为技术管理者时，必须具备的知识。

第三个阶段，是工作 3 到 5 年的阶段。这个阶段，我已经成为公司技术部门的绝对主力，并最终成功晋级系统架构师。作为一名架构师，你必须具备从业务系统、整体架构和团队配合等诸多方面，来考虑问题的能力。

这个时候，团队成员解决不了的疑难杂症，也会需要你来参与分析和决策。这将是你再一次提升自己的重要机会。因为这个时候遇到的业务或技术问题，都是属于比较高级或比较困难的问题。你一定要去想方设法解决它们，并给出一个权衡了多方面考量因素后的方案。

总的来说，我认为在第三个阶段，最重要的有这几点。

1. 作为业务系统的架构设计者和技术负责人，你应该对这个领域的业务理解透彻，并知晓为实现业务系统所采用技术方案的方方面面，包括但不限于应用框架（如 Spring、Netty 等）、集群（水平扩展和高可用）、数据库（不同数据库适合不同的查询计算）、网络、DevOps 等。
2. 系统架构设计能力和表述能力。要能够让其他开发同事按照你的架构方案，切切实实开发出一个可以落地的系统。出了问题你自己要知道怎么解决，并且当同事遇到解决不了的问题时，你能亲自上场。架构绝不是纸上谈兵的事情，也绝不是单纯画几页 PPT 的事情。如果你自己都不清楚怎么去实现你画在 PPT 上的东西，那其他同事就更不清楚了。
3. 最好，你能够总结出一套解决你所从事领域问题的模式。如果能够做到这点，那么我必须祝贺你，你的思想会变得通透，并且会在你从事的领域无往不利。这是一种非常美好的感觉和体验。你可以将你的这些经验总结出来，这既是构建方法论的过程，也是提升思想水平的过程。并且你的总结，既可以帮助后来者，也有利于提升你个人的品牌价值。

第四个阶段，是工作 5 到 7 年的阶段。说实话，这段时期的一开始，我是有些迷茫的。既有个人原因，也有公司原因。个人方面，主要在于技术水平很难再进一步提高。公司方面，则是因为公司的核心产品被收购后，一下子没有找到其他的发展方向。

但好在，经过将近两年时间的调整，以及对更多行业的尝试，比如物联网和电商，我接触了更多诸如供应链方面的社科知识。这些社科知识让我的思想变得更加成熟，包括对技术的理解，以及对世界的认识。我想这是我最终决定成为一名自由职业者的重要原因。

所以，对于第四个阶段，以及之后的路，就只能由你自己去探索了。毕竟，每个人都终将有自己的路，每个人的路都各有精彩！

从业方向

接下来，我想谈下有关 Java 开发人员的几个从业方向的问题。

其实说心里话，虽然这门课程的主题是《实时流计算》，但除了让你理解“流”计算外，我还有另外的私心，是希望你能够理解 Java 高性能编程的关键技术点（也就是 NIO 和异步），以及开拓 Java 开发的视野。对于这两点，相信你在课程学习中应该已经有所体会。

我认为，作为一个 Java 程序员，你的视野一定不应该是狭隘的。千万不要将自己的视野局限在了 Spring，也千万不要离开了 Spring 就不会写程序了。

作为一个 Java 程序员，你至少有以下这些从业方向可以选择：

1. 微服务架构体系，如 Spring Boot + Spring Cloud + Docker + Kubernetes + Rancher；
2. 大数据开发，如 Hadoop、Spark、Flink 等；

3. 全栈开发，如 Spring Boot + Ant Design Vue 等；

4. 嵌入式或移动开发，如 Android。

由于我自己以及这门课程的目标学员，主要是后端开发和大数据开发，所以我就谈下我对 1 和 2 这两个方向的想法。

时至今日，对于后端开发而言，微服务架构体系已经不单纯是 Spring Boot + Spring Cloud 的事情了。以 Golang + Docker + Kubernetes + Rancher 为代表的微服务技术，正在如星星之火，大有准备燎原之势。Docker 提供了方便稳定的二进制发布方式，Kubernetes 提供了统一、灵活、可水平扩展的微服务治理方案，Rancher 更是极大地降低了 Kubernetes 的使用难度，让 DevOps 变得方便简单。而 Golang 目前确实对 Java 造成了很大的冲击，特别是在云原生这个方面。对于这点，我替 Java 感到一丝担忧，因为到目前为止我还没有看到 Java 应用能够不做大改，就能方便解决云原生问题的方法。所以，希望你能够提前做好准备，至少先了解下 Docker + Kubernetes + Rancher 这类微服务技术。

除了微服务外，大数据开发也是 Java 开发人员一个非常好的主攻方向。特别是对中间件开发比较感兴趣，以及对分布式计算技术理解比较深刻的开发人员，他们甚至可以参与到 Hadoop、Flink 等大数据工具本身的开发。并且在可预见的将来，像 Hadoop 这样的大数据系统，都不会主要部署在 Kubernetes 上，而是作为单独集群进行部署。所以，他们可以专注于 Java 和大数据技术的开发。另外，正如我在彩蛋 1 课时中说过的，如果将 Flink 视为分布式 JVM 的话，其实它也是可以构建微服务系统的。从这个角度来看，以 Flink 为代表的实时分布式计算技术，也已经开始准备蚕食微服务的势力范围了。这也是我为什么极力推荐你学习大数据和 Flink 这类技术的重要原因。

对于以上这两个方向，在工作多年后，用人单位都会对更深层次的技术点，有更高的要求，比如高并发、高性能、高可用、分布式一致性、各种开源软件内核（或核心引擎）等。所以，作为 Java 开发人员，我们千万不能局限于当一个“CRUD 工程师”，而应该是未雨绸缪，从现在开始就要深入钻研至少一个从业方向上的核心技术。这样，我们才能够在未来稳定立足，避免大浪淘沙，不进则退。

最后，不管你从事哪一个行业，也不管你将来会处于哪个职位，我都提前祝你取得成功！

同时我邀请你为本专栏课程进行结课评价，因为你的每一个观点都是我和拉勾教育最关注的点。

点击链接，即可参与课程评价，编辑同学会随机抽 5 位同学送精美礼品。