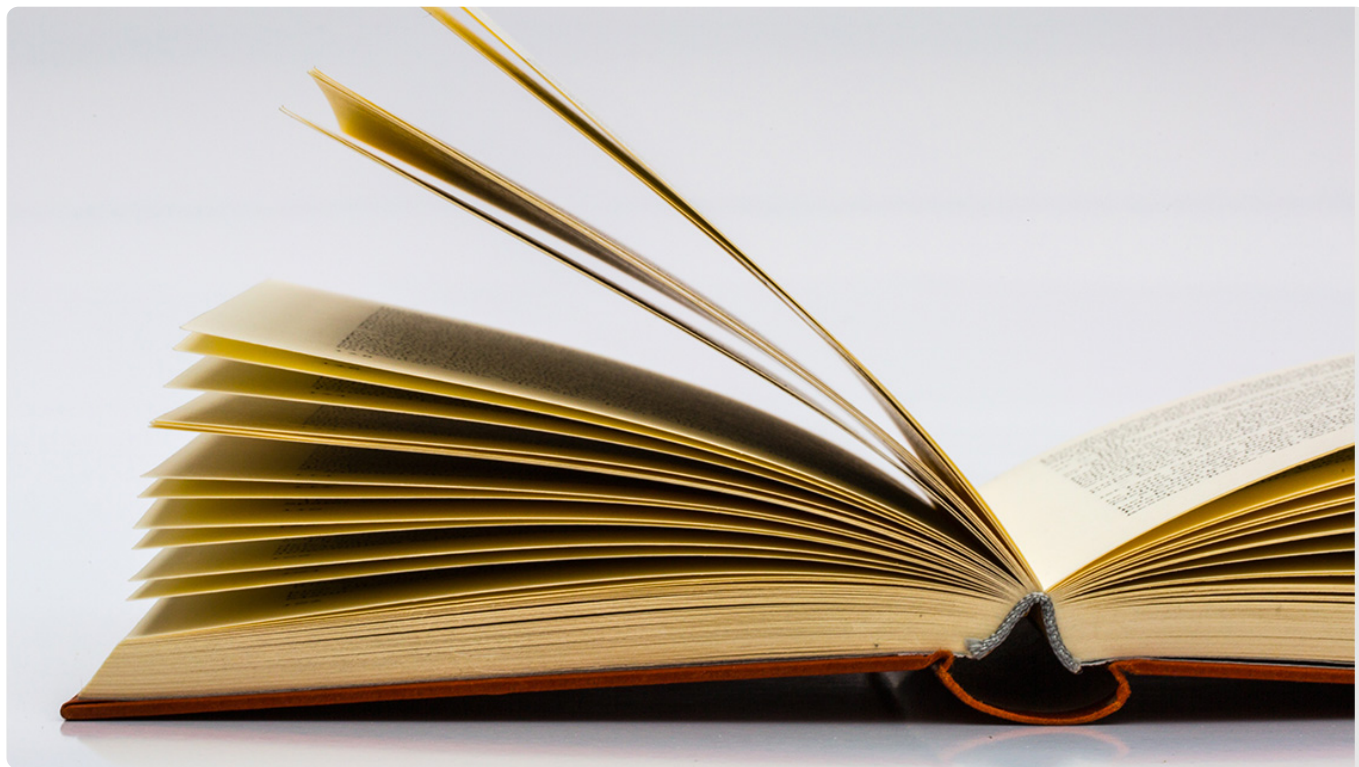


## 用户故事 | 站在前人的肩膀上，领取属于你的高效工作秘籍

2019-03-04 丁雪丰/毅/one day

10x程序员工作法

[进入课程 >](#)



丁雪丰

极客时间《玩转 Spring 全家桶》专栏作者，平安壹钱包高级架构师

### 1. 为什么订阅《10x 程序员工作法》？

我与郑晔老师是多年好友，不过大家平时联系并不多，去年在一次聚会上得知他要在极客时间上开专栏，当时并没有去了解专栏的内容，直到专栏上线后我才知道，这门课就是《10x 程序员工作法》。

从专栏名称上就能大概得知，这门课程的目的主要是为了提升大家的工作效率，开篇词中有一句大实话：

大部分程序员忙碌解决的问题，都不是程序问题，而是由偶然复杂度导致的问题。

咨询师，他帮这么多公司做过各种项目，一定是有秘籍的。因此我毫不犹豫地第一时间订阅了这个专栏。

## 2. 对于高效工作有哪些心得？

郑晔老师在专栏每一课的最后，都会用一句话来总结这一课的内容，我很喜欢这个风格，那我也先用一句话来说一下，我对高效工作的理解：能用尽可能小的时间颗粒度来安排自己的工作。

高效工作，如果按字面理解，就是在指定时间里能完成更多的工作（当然还得是合格的工作）。《五分钟商学院》的作者刘润老师曾提到过时间颗粒度的概念，你会发现，很多成功人士划分时间的颗粒度都比较小。王健林的时间颗粒度是十五分钟，比尔盖茨的时间颗粒度是五分钟。

你可以思考一下，你是按什么颗粒度来安排自己工作的呢？1天，半天，1小时还是15分钟？

另外，在把时间拆细的同时，也不能忽略对工作内容的分解。其实，对工作内容进行分解的颗粒度大小，同样对工作效率有很大影响。

讲到这里，又不得不提到《10x 程序员工作法》专栏，我看到郑晔老师在“任务分解”这个模块里，详细地讲解了怎样将工作任务进行拆解，如果对这部分内容感兴趣的同学，也可以去订阅专栏。

## 3. 学习《10x 程序员工作法》专栏的感受

我认为这个专栏最大的价值，就是为我们程序员的工作，划出了清晰的、可以遵循的高效工作原则。

我举个具体的例子吧，我团队中不止有一个同学是技术能力非常突出的，他们在拿到一个任务后，立马就能想到大概的解决方案，紧接着就开始写代码了……没错，撸袖子就开始写代码了。

做为过来人，我在看到他们这么快就开始写代码后，就会把他们拦下来，跟他们聊聊各种需要考虑的事项，他们之前忽略掉的问题，或者是需要细化的点出来。

在我看了专栏中的“以终为始”这个模块内容之后，我发现这不正是我一直在寻找的东西吗？我平时零散地问各种问题，让他们考虑各种点，现在郑晔老师都已经把这些常见的实践，提炼总结成为高效工作的原则。

我们可以不用在针对工作中的各种问题孤立地作出反应，而是有了一套行之有效的原则可以遵循。

如果说我们平时实践的是“术”，那郑晔老师讲的就是“道”。我觉得只“以终为始”这一部分的内容就已经完全值回票价了。

Ivan (毅)

10+ 年研发经验，坐标苏州，目前在医疗信息化软件公司从事研发兼管理工作。

## 1. 为什么订阅《10x 程序员工作法》？

记得刚开始工作时，一门心思想着多做项目，多学技能，多攒经验，当然，战斗数值也增长不少，但就像武侠小说中的习武之人都会碰到的困惑，总觉得是围绕着招式在打转，而修为不足。

之后我也逐渐也有意识培养自己的思维方式和从多角度看问题的习惯，这些年也算形成了一些自己的价值观和方法论。

我现在服务的公司是一家初创公司，产品化和定制项目开发之路走的并不轻松，我发现不是团队效率低下，也不是不知道如何解决各种问题，而是我们投入很多精力在处理“偶然复杂度”的问题，我想这可能也是工作不轻松的原因之一吧。

所以在听郑晔老师的开篇词时，我就有一种被吸引的感觉，因为我觉得问题既然抛出了，后续肯定会有解决方案。

趣也越来越浓了。

我是个偏向实用主义的人，除了“醍醐灌顶”的那一时快感，更看重方法在实践中的运用和反思，我相信《10x 程序员工作法》中介绍的经验和最佳实践一定能给我和团队带来工作效率和工作绩效的提升！

为了保持学习思考的连贯，我都是紧跟一周三次的更新节奏同步收听，即使春节期间也未间断，而且部分内容会反复听，每次重听都有新的感受。

而从第二部分开始，我觉得不光是收听，还要用文字表达出来，毕竟有互动有交流的学习才是有效的学习，而且我是强制自己一定要留言，有思考的留言。

## 2. 学习《10x 程序员工作法》的心得

基于之前的学习过程，我再总结下我的学习心得。

### a. 思维方式的改变

课程虽然针对程序员，但受众却超过程序员范围。即使对于程序员来说，在工作中也要拓展自己的上下文，将自己放在更大的范围、平台上去思考问题，主动发掘问题关键点，在面对不同职能人员间沟通时，多运用“以终为始”模块中的知识尝试解决分歧，达成共识。

此时我或许明白了之前公司老板常说的一句话“你们都是工程师思维”，或许并不是将自己想象成其他角色，而是在更大的上下文背景中寻找突破点。

### b. 工作习惯的培养

程序员的工作习惯不仅仅是每天上班打开邮箱，提交代码前编译通过等等事务性习惯，还包括：**你的思维方式是否合适，由此产生的执行方式是否到位，检验方式是否科学，理论修正是否能产生更好的思维方式。**

就像专栏中讲精益创业时，提到的“想法 -> 产品 -> 数据”对应的“开发 -> 测量 -> 认知”这样的反馈模型。

我们可能习惯于之前的工作方式，有时即便感觉别扭，但仍然只是希望在原有方式上修修补补，而不愿去仔细区分现实和期望，常常会以忙为理由，为自己辩解。

**但如果不忙，你知道该怎么做吗？**这恐怕是绝大多数程序员都要自省的一个问题，但这也恰恰说明我们在工作方式上是有很大的提升空间。

#### d. 主动思考也很重要

孔子说“学而不思则罔，思而不学则殆”，《10x 程序员工作法》并不是技术技能课，可以直接准备环境，代码走起，而是一种方法论课、答疑解惑课、实践验证课，听起来很过瘾，很能戳中痛点，但如何解决自己的实际问题则需要主动思考，主动交流。

这也是我后来逼着自己每学完一课都要思考后留言的初衷，并且我还会向周围人推荐这种学习方式。

其实，对于任何方面的知识，我都奉行“一学二懂三要用”，四要知道局限性的准则，不断更新自己的认知体系，力求做到融汇贯通，逐步形成适合自身的知识架构和行动准则。

#### One day

4 年研发工作经验，坐标上海，目前就职于一家汽车行业的公司做后端开发

#### 1. 为什么订阅《10x 程序员工作法》？

想起刚毕业的时候，我还是能够前后端通吃的，因为那个时候，很多公司使用的还是 jsp+servlet 等简单的单体应用，所有功能全部集成在一个工程项目中。

等所有人把各自模块开发完，就丢一个 war 包到服务器，然后就进行测试就完事了。等测试小哥哥或小姐姐发现 Bug 的时候，自己就从前端到后端一步步的 debug，虽然有很多血与泪的历史，但是累并快乐着。

但工作久了之后，我发现很多工作是重复且无意义的，然后就是加班与熬夜并存。现在，我已经在软件开发行业已经混迹 4 年多，从一个混沌不知的菜鸟也渐渐有了自己学习方法。



最开始看到专栏标题的时候，还心有疑惑，极客时间会有标题党？我学了这个专栏工作学习能提高 10 倍吗？

但当看到 10X 程序员工作法的内容，我就已经被郑晔老师的能力所折服，因为专栏目录中提到的各种思考原则，真的是很贴合实际。

工作能力的提升，本身是经验的积累，但是如果长时间堆砌同类型的业务知识，对于大脑来说是一种重复，也不利个人经验的沉淀。实践和理论总是相辅相成的，我是一个实践派，但是对于理论的方法，我也非常赞赏。

这相当于是站在前人的肩膀上，你在实践的过程中，得到与之类似或者不同的结果。以结果为导向，再结合理论就有不同的火花，我认为这是一件很神奇的事情。

## 2. 学习《10x 程序员工作法》专栏的心得

### a. 多思、多问

我们的实际工作常常会被打乱，需求频繁变更，做出的东西与产品有出入等等。这其中很多情况我都经历过，比如，拿到新需求任务时，头脑没有思绪，大脑出现空白，直接上手就做。然后，到测试和交付的时候就经常出现问题。

结合老师说四项基本原则，简化思考框架，遇到问题多思考、多提问、多练习，去梳理真正要做的事情。清楚事物背后的本质，有效剔除不必要的思虑，让自己专注于任务本身。

现在我在接收一个新需求的时候，常常会思考这个需求的意义是什么？到底需不需要做？要怎么做才能更好？有没有好的替代方案？需求应用的场景是什么？这样做以后方便维护吗？

之类的问题，思考的越多，在以后的写代码的时候，就会考虑更多。需要并发吗？并发量多少？我们服务器的承受压力多大？我应该用多少个线程跑这个才合适呢？这样写出来的代码也会很健壮。修修改改，缝缝补补的代码即不利于个人发展，对后面的迭代也是一种煎熬。

### b. 以结果为导向

我们日常都是按部就班的完成需求任务，确定好任务安排，于是就开始写代码。这个时候常

这时，我们就要以结果为导向，考虑到即将要做的事情，再将要做的事情一步步细分，然后做项目推演，带着明确的目的去工作，这样可以少走弯路。

### c. 站在不同角色角度看问题

我们工作出现的很多问题，都是因为我们只站在程序员的角度，只能看到局部，缺少大局观导致的。不同职位在协作的过程中，认知和理解的差异直接导致最终不能达成一致意见，到最后还是苦逼了程序员，在做无限制的修改。

项目本身是有不同人员角色组合的共同体，大家为了达到共同的目的在做事，所以有必要跳出程序员角色思维的限制，站在项目本身去看待每一个问题。必要的沟通是需要的，对待不合理的需求，我们程序员也要敢于说不。



## 10x 程序员工作法

掌握主动权，忙到点子上

郑晔

火币网首席架构师  
前 ThoughtWorks 首席咨询师  
TGO 鲲鹏会会员



新版升级：点击「👤 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 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 26 | 作为程序员，你也应该聆听用户声音

下一篇 27 | 尽早暴露问题：为什么被指责的总是你？



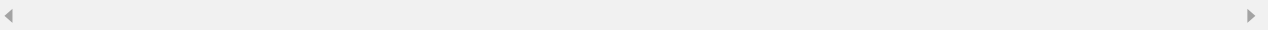
**One day**

2019-03-04

 10

分享知识，发散思维，授业解惑，不同的想法碰撞，这种进步是我人生的收获，谢谢郑老师

作者回复: 我的初衷有人看到，谢谢你们！



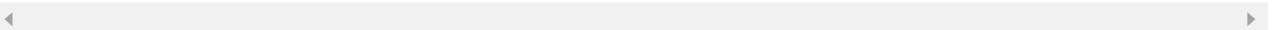
**莫小鹏**

2019-03-06

 1

以终为始，去掉不必要的工作，效率自然提高了。很多性能优化的思路也是这样

作者回复: 其实就是把熟悉的原则用在不同的地方。



**四方街个**

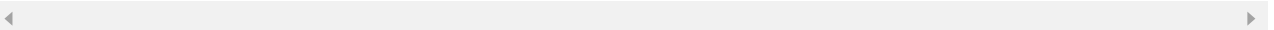
2019-03-04

 1

思考-总结-输出，这样学习的效率最高

展开 

作者回复: 输出，是一种好的学习方式。



**246小言**

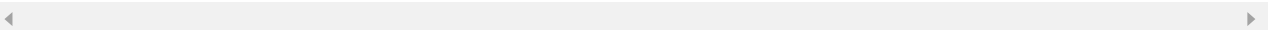
2019-03-31



很棒的课程

展开 

作者回复: 欢迎分享你的所得！





结底是要对自己负责。team听完就挑大拇指，说收获很多。另外，这一周几个骨干陆续提出辞职，找到更好的自动化测试的职位😁

展开 ∨

作者回复: 这个反转让人猝不及防。



**hua168**

2019-03-05



花了几天时间一口气看完了20多篇，对我我这个没有项目经验的人来说帮助感觉很大，算是长见识了~~以后按照老师专栏中的建议，我相信一定会少走很多弯路。

同时我有一个疑问：像我三十五六才学编程会不会迟了？

我现在是在小公司做运维，之前一直是用shell，打算做运维方面的开发，去年开始按网...

展开 ∨

作者回复: 迟不迟这种问题，你要先问问自己为啥要学编程。

站在以终为始的角度，你先了解一下你未来打算从事的工作究竟需要什么样的技能，然后是有针对性的努力。自己闭着眼睛瞎努力就只能靠运气了。



**Sudouble**

2019-03-04



知行合一，知道了改进的方向，努力大胆的干起来！

展开 ∨

作者回复: 欢迎留言分享你的进展！



**WL**

2019-03-04



思维模式×热情×能力=成功，好的思维方式能让我们的能力发挥更大的价值

作者回复: 学习应该成为习惯，成就更好的自己。

◀ ▶



**it-spurs**

2019-03-04



看来很多读者的心声都很多一致的地方。学好做事的方法，比学会如何完成一件事，学会一个技能，意义更重大。

作者回复: 建构起自己的知识体系，是非常重要的。

◀ ▶