

# 程序员<sup>的</sup> 成长课

有效增值、加薪、升职、转型

安晓辉 周鹏 著



程序员成长课

# 程序员成长课

程序员进阶、成长、升职指南

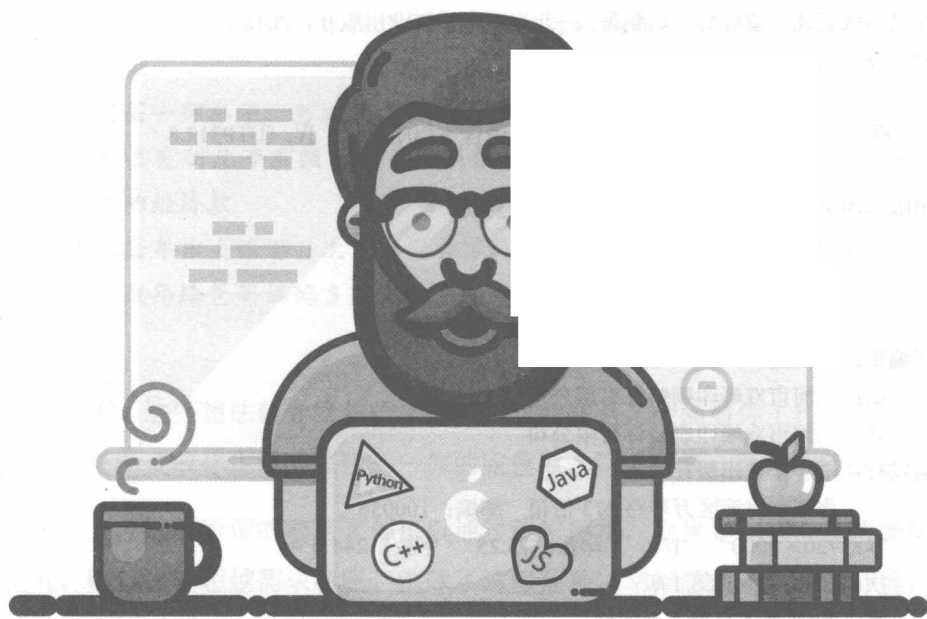
程序员进阶、成长、升职指南



中国IT教育网 [www.it-ebooks.com](http://www.it-ebooks.com)

# 程序员的 成长课

安晓辉 周鹏 著



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

知道和做到之间有一道鸿沟，要想跨越它，你不但要努力，还要讲究方法。本书不仅传授知识，更教给读者如何做到极致。书中的理念、方法、工具可以指导读者更好地看清定位及发展方向，设计未来。如果读者不知道选择什么技术栈来学习，困惑于怎样在技术上持续精进，想转技术管理却没途径，想有章法地为跳槽加薪做准备，或者想转型却不知道除了技术还能干什么，建议认真读本书，它提供的方法和工具可以帮助读者很快找到答案。

本书适用的读者：所有程序员，IT 相关专业的学生、教师、培训机构学员等。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有，侵权必究。

### 图书在版编目 (CIP) 数据

程序员的成长课 / 安晓辉, 周鹏著. —北京: 电子工业出版社, 2018.1  
ISBN 978-7-121-32909-8

I. ①程… II. ①安… ②周… III. ①C 语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2017) 第 259466 号

责任编辑：高洪霞

印 刷：三河市双峰印刷装订有限公司

装 订：三河市双峰印刷装订有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：720×1000 1/16 印张：15.25 字数：244 千字

版 次：2018 年 1 月第 1 版

印 次：2018 年 3 月第 3 次印刷

定 价：49.80 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888, 88258888。

质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn)，盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

本书咨询联系方式：(010) 51260888-819, [faq@phei.com.cn](mailto:faq@phei.com.cn)。

# 前 言

作为一名普通的程序员：

- 你想买一套房子，不想再租住在远离公司的偏僻地带、每天通勤3个小时上下班
- 你想买一部车子，可以周末开着去山里转转，看看红叶，听听鸟鸣
- 你想去窗明几净有微笑服务的商场买衣服而不是每次都网购一个不知名品牌的经济款
- 你想每年出去旅游  $n$  次，今天在苏梅岛潜水，明天在魁北克吃枫糖
- 你想每年给爸爸妈妈5万元的生活费，让他们露出欣慰的笑脸

实现不了这些想法就会给你带来痛苦。这种痛苦，会随着你工作时间的增长而加深，渐渐变成你生活的底色——你的底色原本简单明快，现在幽暗阴郁。

实际上，这些所谓的痛苦大都可以通过钱来解决。前提是：你的价值能够不断提升，赚钱速度超越需求膨胀。消除了这些痛苦，幸福感就有生长的环境，就不那么容易被淹没。

可是作为普通的程序员，你却发现瓶颈一个接一个地扑过来。做技术，不知道怎么做持续精进、怎么坚持；转管理，又不知如何开始。结果还没等想明白呢，半载一年就过去了，蓦然回首，好像自己的能力没怎么提高，薪水增速却越

来越跑不过通货膨胀了。

有时候你觉得开发工作越来越吃力，转型的呼声越来越高，却不知道如果离开开发岗位自己还能干什么。看着别人可以选择当自由职业者，或者能实现财务自由，内心羡慕，然而转过身却只能叹息：自己的路，究竟在哪里？

仔细想想，你就会发现，要搞定这些事情和问题，只要能赚到更多的钱就可以了！

这个结论很俗吗？

不，现实正是如此！

对于大部分开发者来讲，工作和生活的诸多烦恼，其实都源自于：怎么赚到更多的钱。

要想赚到更多的钱，就要回到问题的原点，想想个人赚钱的本质是什么。

个人赚钱的本质是——出售时间！对吗？

从出售时间的角度来看：

个人收入=每天可售时间数量×单位时间价格×单位时间出售次数

在这个公式里，有三个要素，简单描述就是：

- 每天可出售的时间数量
- 单位时间价格
- 同一份时间的出售次数

结合开发者的具体情况，可以找到多种提升收入的方式。参考下表：

时间单价	业余时间工作（时间数量）	一份时间卖多次
提升专项技能	接外包项目	录制技能类课程
提升架构设计能力	技术咨询	个人站点广告位
培养项目管理能力	众包	出版技术图书
提升领导力	技术自媒体	在线技术分享
转管理岗位	撰写技术图书书评	股票、期权
跳槽	翻译外文图书	投资理财
……	……	……

或许你知道所有这些方式甚至知道更多，但是，怎么做呢？

这是个大问题！

知道和做到之间有一道鸿沟，要想跨越它，你不但要努力，还要讲究方法。

这本书不会承诺“看完本书就能快速赚钱”，不提供任何安慰和幻觉，它只是一本工具书，仅仅是发出一份邀请，让你看到有一些方法可以用来分析自己、帮助自己去有效选择和提升，但最终你是否可以用书里的理念、方法、工具指导自己的工作与生活，则取决于你的认真程度和执行力！

所以，如果你愿意付出努力让改变发生，那么我们现在开始一起行动！

本书共有 8 章。

第 1 章，介绍如何选择技术方向，我们结合技术成长三阶段模型，讨论在入门、构建技能树、技术转型、团队技术方案选型等常见场景中如何选择适合自己的技术，提高增值的效率。

第 2 章，讲述如何在技术上持续精进，着重讨论如何在工作中将持续提升自我能力和价值落在实处。为此我们引入了个人对标管理法，让你从优秀的同行者、一般性规律、技术自身的深浅层次和软件项目指标四个方面出发，找到随时随地可以引领自己前进的小目标、小台阶，再配合职业目标的指引，小步快跑，日有寸进。

第3章，探讨了开发者无法回避的发展方向——技术管理。从开发者到管理者，不是职位序列自然发展的结果，而是一种转型，且并不适合所有人，因此我们提供了一些工具和方法，让你自测一下，看看自己是否适合。如果你发现自己很想试试管理路线，可以接着看走向技术管理的4种常见方式，然后了解怎样成为管理者做准备。从概率上讲，成为管理者后，可以通过团队实现更大的产出，也会因此赚到更多的钱，你的未来会有更多可能性。

第4章，讨论了技术管理新人面临的18种常见挑战，比如角色适应、委派任务、激励他人、冲突管理、一对一谈话等。如果你刚刚升任管理者，一定会碰上这些挑战，翻翻本章提到的应对策略，会对你有所帮助。

第5章，介绍了开发者在跳槽时常见的8个问题。比如什么时候跳槽好、依据什么跳槽、选大公司还是小公司、去北上广深等一线城市还是找个小地方享受生活等。这些问题会给你触动，引发你思考自己的选择。

第6章，讨论了如何针对匹配度来优化你的简历。最关键的要点有两个，一是一个岗位一份简历，二是分析目标职位的要求，针对要求呈现你的价值点。你能注意到这两点，就可以大大提升简历的通过率，为自己赢得机会。

第7章，如何在跳槽时获得想要的薪水，这恐怕是每个人都关心的话题。其实从你决定应聘这家公司职位的那一刻就开始涉及这一点了，核心要点是匹配度。所以，从简历优化到了解公司、产品、部门、职位，再到了解目标岗位薪酬区间，你都要围绕匹配度做准备。如何准备？这一章提供了流程、框架和方法。

第8章，转型，也许有一天，我们会离开软件，想要去做别的事情。那么你还适合做什么？想做什么？如果你“拔剑四顾心茫然”，可以看看这里的人、事、物模型和发现职业方向的5步法。只要你做好准备、花些精力，就可以找到适合自己的转型方向。这一章还讨论了转型时如何准备目标职业所需的知识、技能，并提供我和我的朋友们实践过的最佳方法：双职业策略。

在正文之后，有一个附录是我个人分类总结的书单，对开发者的综合素养提升大有帮助。



轻松注册成为博文视点社区用户 ([www.broadview.com.cn](http://www.broadview.com.cn)), 扫码直达本书页面。

- **提交勘误:** 您对书中内容的修改意见可在 [提交勘误](#) 处提交, 若被采纳, 将获赠博文视点社区积分 (在您购买电子书时, 积分可用来抵扣相应金额)。
- **交流互动:** 在页面下方 [读者评论](#) 处留下您的疑问或观点, 与我们和其他读者一同学习交流。

页面入口: <http://www.broadview.com.cn/32909>



# 目 录

<b>第 1 章 如何选择技术方向</b> .....	1
1.1 技术成长三阶段模型.....	2
1.2 选择技术方向都要考虑哪些因素.....	4
1.3 入行时怎么选技术方向.....	9
1.4 构建技能树时选择技术方向.....	12
1.5 技术转型时的方向选择.....	13
1.6 方案选型.....	16
1.7 工具推荐.....	20
<b>第 2 章 如何在技术上持续精进</b> .....	28
2.1 做开发还是转管理.....	29
2.2 找到激励你前进的动力.....	32
2.3 学习型心态.....	34
2.4 技术精进之道.....	34
2.5 目标的设定与执行.....	44
2.6 精进的 4 个习惯.....	47
2.7 习惯养成指南.....	50
2.8 超越技术层面的核心竞争力.....	54
2.9 公司内的职业规划.....	58
<b>第 3 章 成为技术管理者</b> .....	60
3.1 真的要做管理吗.....	61
3.2 5 大职能定位.....	62

3.3	理解管理角色	66
3.4	走向技术管理的4种方式	82
3.5	怎样为成为管理者做准备	90
<b>第4章 技术管理新人面临的挑战</b>		<b>97</b>
4.1	挑战1: 角色转变	98
4.2	挑战2: 被动管理	100
4.3	挑战3: 弄不清职责	101
4.4	挑战4: 委派任务	103
4.5	挑战5: 目标管理	104
4.6	挑战6: 资源管理	107
4.7	挑战7: 压力管理	108
4.8	挑战8: 冲突管理	112
4.9	挑战9: 绩效变差	118
4.10	挑战10: 担心失去技术竞争力	119
4.11	挑战11: 有效的反馈机制	120
4.12	挑战12: 别人的议论	121
4.13	挑战13: 和下属进行一对一沟通	122
4.14	挑战14: 怕犯错	124
4.15	挑战15: 时间管理	125
4.16	挑战16: 激励他人	128
4.17	挑战17: 向上管理	134
4.18	挑战18: 提升领导力和管理能力	137
<b>第5章 跳槽8问</b>		<b>139</b>
5.1	为什么要跳槽	139
5.2	什么时候跳槽好	142
5.3	什么是好的跳槽和坏的跳槽	148
5.4	跳槽还是卧槽	149
5.5	要不要追薪式跳槽	152
5.6	选大公司还是小公司	152
5.7	去大城市还是小城市	156

5.8 自己的选择是明智的吗 .....	158
<b>第6章 简历优化指南 .....</b>	<b>160</b>
6.1 简历优化模型 .....	160
6.2 明确你想要的工作特征 .....	161
6.3 盘点你的价值和亮点 .....	163
6.4 寻找机会 .....	167
6.5 招聘信息分析 .....	175
6.6 简历优化 .....	177
6.7 检验简历优化效果 .....	182
6.8 如何提高简历投递成功率 .....	183
<b>第7章 如何在跳槽时获得想要的薪水 .....</b>	<b>184</b>
7.1 决定薪水的7大因素 .....	185
7.2 我们能直接控制哪些因素 .....	188
7.3 如何为谈薪水做准备 .....	189
7.4 薪资谈判的6个秘密 .....	195
7.5 什么时候可以降薪求职 .....	197
7.6 薪水与幸福成正比吗 .....	198
<b>第8章 转型 .....</b>	<b>200</b>
8.1 你真的不再喜欢开发工作了吗 .....	201
8.2 程序员转型的难处 .....	203
8.3 转型的分类 .....	208
8.4 与开发者相关的转型方向 .....	211
8.5 如何确认哪个职业适合你 .....	212
8.6 转型的最佳实践 .....	222
<b>附录A 实用图表、方法、工具索引 .....</b>	<b>225</b>
<b>附录B 私房书单 .....</b>	<b>227</b>
<b>附录C 值得订阅的技术类公众号 .....</b>	<b>233</b>

# 第1章 如何选择技术方向

努力只有在方向正确时才有价值。

开发者选择技术也是如此。假如你阴阳差错选择了一门过时的技术或者在当地没什么单位使用的技术，你在找工作时就会不断受挫，即便你找到了工作，也会忧心自己的发展前景。

- 你刚进入软件开发领域时，需要选择适合你和市场的技术才能更好地就业。
- 你做了一两年技术工作，开始构建自己的技能树（知识图谱），需要选择与你现有技术相互补充、1+1>2 的技术来学习和运用。
- 你的技术能力慢慢在团队里凸显出来，成了先锋官，有新的项目、新的产品时，你总是被赋予探索技术方案的角色，可是一着不慎，你选择的技术就可能导致整个项目的失败。
- 你使用 C++ 语言做了 5 年开发工作，越来越觉得这门语言不适合自己，想要学习一种新技术作为自己以后的主要开发技术，很担心再选错——因为你做开发的黄金时间，加起来也不过三四个 5 年。
- 你成为技术负责人或者研发经理，需要决定团队的技术图谱，需要为团队承接的新项目选择技术方案，要考虑哪些因素呢？

所有这些问题，我们都会在本章讨论，我们会找到选择技术方向时要考虑的各种因素，然后结合特定的场景，看看你在选择时需要考虑哪些因素。

通过本章的学习，作为个人，你会掌握怎么选择一个适合自己的技术方向，好的方向会让你越来越有竞争力，越来越值钱；作为团队的一员，你会知道在为某个项目选择技术时该如何综合考虑，在风险、成长、代价之间进行权衡。

## 1.1 技术成长三阶段模型

为了更好地选择技术方向，我们先来介绍一个模型，我给它取了个名字，叫“技术成长三阶段”，如图 1-1 所示。

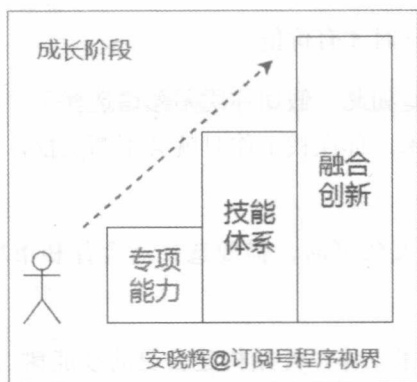


图 1-1

我们在一个专业领域内的成长，基本上都会经历三个阶段：

- 专项能力的提升
- 技能体系的构建
- 融合创新

### 1. 初级阶段：专项能力的提升

专项能力提升阶段是初级阶段，你为了做事情，必须先具备某些基础能力，比如从事软件开发工作，你要学会某种语言、某个 IDE、某种技术框架。

如果你刚刚从学校毕业开始做开发，或者刚刚从别的跑道转换到开发领域，那么你就处在这个阶段。这个阶段最重要的就是提升专项能力，让自己能够迅速

搞定一些别人安排给你的事情，体现出你的价值。

这个阶段持续的时间长短因人而异，可能会是 1~3 年。如果超过 3 年还没有进入下一个阶段，那么你的成长速度就需要提升了。

### 2. 中级阶段：技能体系的构建

这是中级阶段，你拥有了一组技能，围绕某个方向构建了自己的知识图谱，能够用自己的方式来解决实际问题。比如你使用 Java 语言做后台方面的开发，你的技能体系可能由 Java、MyBatis、Spring、SpringMVC、Netty、MySQL、Python、Linux 等组成。此时你在团队中，应该已经可以独立负责某个模块，能够完成模块的设计和开发工作，也能够指导初级阶段的同事进行开发。

你可能需要 2~3 年，经历 2~3 个项目才能慢慢构建起自己的技能树，然后还会在这个阶段待上 1~2 年，不断练习你的技能体系中的各项技能。所以当你能拥有一棵强大的技能树时，距离你刚从事开发工作可能有 5~8 年了。

假如你超过这个年限，用过的技术还是散乱的，东一榔头西一棒槌，不能有机组合在一起，那么你的开发经验和能力，肯定大大落后于你的工作年限。换句话说，你可能把一年工作经验学到的知识、技能机械地重复了几年，没有获得应有的成长。

### 3. 融合创新

这是高手阶段，你具有了丰富的实践经验，具备了 T 型知识结构，形成了自己的思维框架和解决问题的框架，能够融合不同领域的知识，组合各种资源，创造性地解决各种问题。

进入这个阶段非常重要的一种标志就是，你遇到问题，不再从下而上去思考（即从技术实现细节来考虑问题好不好实现、拿技术去裁剪问题或重定义问题），而是从实现细节跳脱出来，站到更高的层面，自顶向下去思考、去分析，先运用框架、逻辑去分析真正的问题是什么、问题的目的、问题的现状、如何去解决。搞明白这些之后，你才会沉降到技术层面去考虑实现的选择，而且实现时，你也不会拘泥于某种技术，而是什么技术合适就用什么技术（你的目的不是“用 Java 或 Redis 解决问题”，而是“解决问题”）。简单说，就是你走出了被技术束缚和塑

造的过程（前两个阶段），可以反过来回到问题本源来思考了。

在这三个阶段中，我们都会遇到选择技术方向的问题。比如在初级阶段，入门时要选择，发现某种技术不适合自己的时候要重新选择；在中级阶段，我们要选择某些技术来构造我们的技能树，或者要做技术转型，需要重新选择技术方向；再比如在高级阶段，我们往往会负责产品的技术方案探索与选型，免不了要选择技术方向，甚至在这个时候，我们个人也可能会在技术方向上转型，必须有所选择。

所以，我们根据成长阶段的划分，挑选了下面几个可能需要选择技术方向的时机，分开来讲在这些时机怎么选择技术方向。

- 入行
- 构建技能树
- 技术转型
- 方案选型

但在展开之前，我们需要先通盘梳理一下选择技术方向都需要考虑哪些因素。

## 1.2 选择技术方向都要考虑哪些因素

我们先来看看选择一种技术可能会考虑的因素，然后看看在每个选择时刻运用哪些因素。

### 1. 就业机会

这个因素，考虑的是哪种技术更容易就业，需要结合特定地区、特定行业来看，因为你找工作是面向地区和行业的。

这中间又有两种典型的策略，热门的和冷门的。比如 Java 和 JavaScript 是典型的热门语言，你可以选择学习它们；再比如汇编语言、R 语言、Scala 语言、Qt 框架就没那么热门。

热门和冷门是相对的，根据开发者多少、市场需求而定。



### 2. 自我感觉的难易程度

比如我当年就在 Java 和 C 语言当中选择了 C 语言，因为《The C Programming Language》比《Java 2 核心技术》薄得多，面向过程的语言也比面向对象更容易入门（对我来讲）。

难易程度与人相关，所以最好的方法，就是把你感兴趣的语言，都学学试试，可能每种语言花上一到两个星期，你就可以做出判断。

### 3. 兴趣

很多人做开发可能是出于某种兴趣，比如对游戏感兴趣就做游戏开发，对智能手机感兴趣就做 APP 开发，对电子商务感兴趣就做前端开发。

当你喜欢某一类产品时，这类产品往往可以关联到技术。比如你喜欢做手机游戏，那么可能会选择 Unity 3D 或者 Cocos 2d-x；如果你喜欢樊登读书会这种学习类 APP，可能会选择 Java、Android、Objective-C、Swift 等。

### 4. 薪水高低

大多数人选择技术的出发点是工作和未来发展前景，而选择工作时一定会把薪水作为参考因素，而且是非常重要的因素。所以我们在选择技术时，也可以考虑某种技术对应职位的薪水高低。

要了解这一点，可以去看各种薪酬报告，如 100offer 每个季度都会有类似的报告出来，拉勾网也会发布互联网职场生态白皮书，你用“互联网人才流动报告”“开发者薪酬报告”或者“互联网薪资调查”为关键字搜索，能搜到很多这类报告。

通过报告，你可以看到使用不同语言的开发者的薪水差异。图 1-2 是我从 100offer 2016 年春季互联网高端人才流动报告中截取的。

从图 1-2 中可以看出不同编程语言所关联职位的平均年薪差异。

### 5. 技术在将来的发展前景

你肯定不愿意看到这种情况发生：选择了一门技术，结果 1 年后它就没人用、没市场了。所以我们在选择时就要考虑这一点，做一些判断。