

学习攻略 | 怎样学好软件工程？

2019-02-21 宝玉 来自北京

《软件工程之美》



你好，我是宝玉。

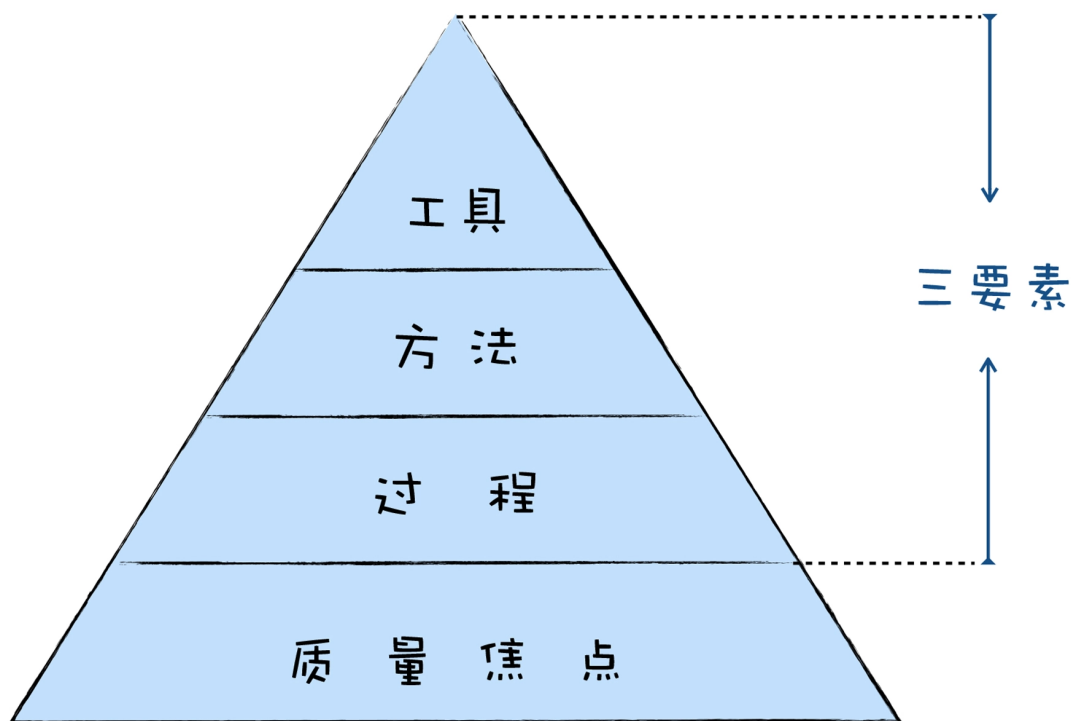
关于“什么是软件工程”和“为什么要学软件工程”，我在开篇词中已经简单介绍过了。总结来说：软件工程是软件行业知识体系的内核。无论你想走技术路线，还是转向做管理，想要走的更快更稳，那就绕不开软件工程。

在正式开始学习前，我们先来聊聊应该如何学习软件工程。你要先知道，软件工程学科的“知识树”结构是什么样的，才能更好地理解每个知识点和彼此间的联系。

软件工程知识架构全景图

首先你要明确，当我们谈软件工程学时，究竟在讲些什么呢？

在《软件工程——实践者的研究方法》这本经典软件工程教材中，作者 Roger S.Pressman 画了一张图，高度概括了整个软件工程的核心知识。



由图可见，“质量焦点”在最底层，这不难理解软件工程是为了应对软件危机诞生的学科，其目标就是为了要**聚焦于质量，构建和维护高质量的软件**。可以说，聚焦于质量就是软件工程的基石。

那“过程”指的是什么呢？

要构建高质量软件，则要解决软件过程中的混乱，将软件开发过程中的沟通、计划、建模、构建和部署等活动有效地组织起来。而软件过程，就是在软件项目的生命周期内，也就是软件从诞生到结束这期间，在开发与构建系统时要遵循的步骤。

有两种过程框架你一定经常听到，那就是瀑布模型和敏捷开发。这是在软件工程多年的发展中，逐步形成的两种主流的软件过程指导框架。

那么，何为“方法”？

方法是指在整个过程中，如何构建系统的方法学。比如说，如何分析用户需求；如何对产品进行测试验收；如何进行系统架构设计等。

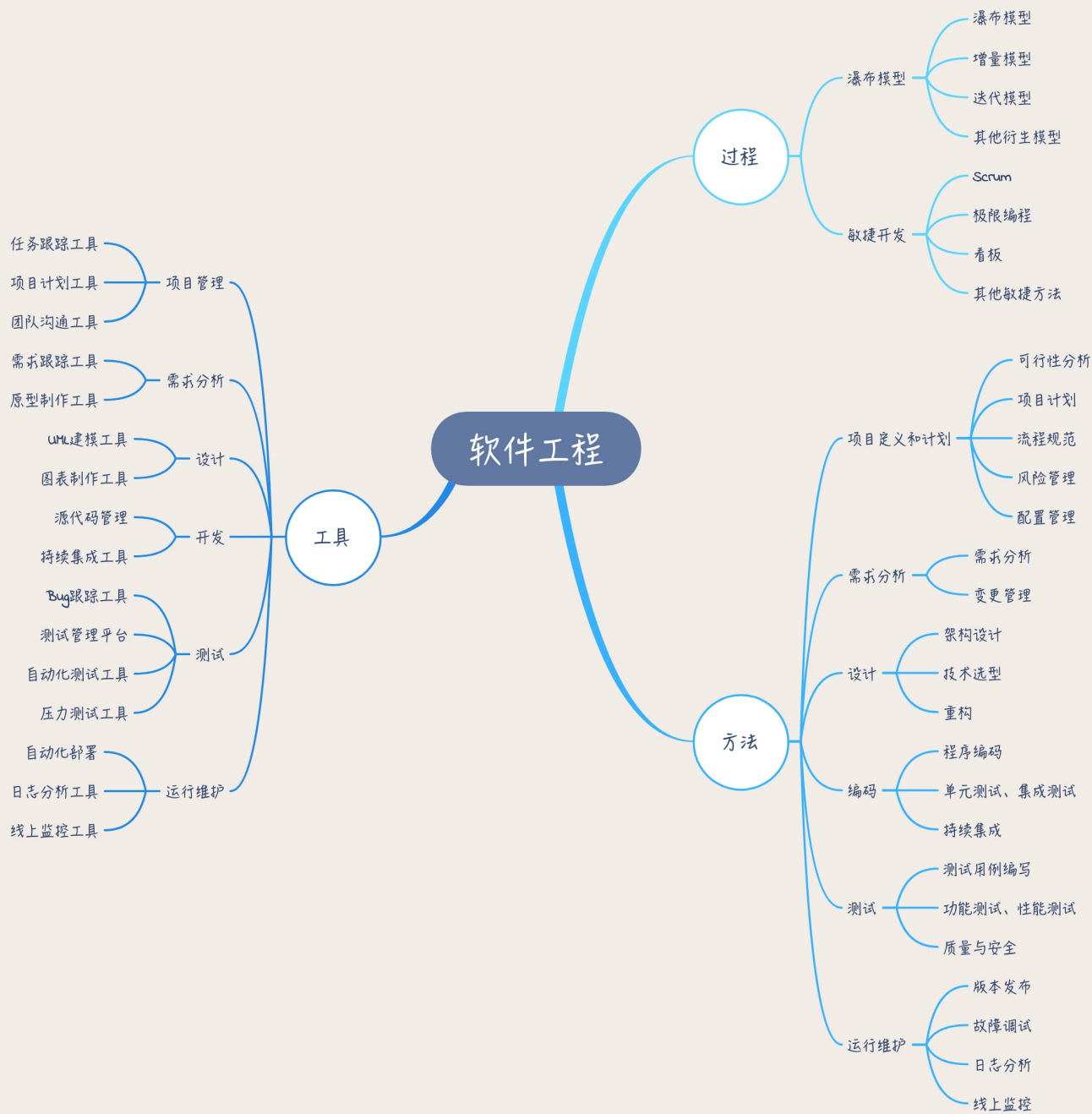
知道了过程，掌握了方法，那么具体落到操作层面，就会涉及到工具的使用。

我们需要工具来辅助方法的执行，提高效率。通过工具，可以把一些手动的工作自动化，比如自动化测试工具，自动构建部署工具；通过工具，可以帮助把一些流程规范起来，比如 Bug 跟踪、源代码管理；还可以通过工具，帮助提高编码效率，比如各种编辑器 IDE、各种高级语言。

如果现在再回头总结一下，软件工程的核心知识点，**就是围绕软件开发过程，产生的方法学和工具。**

你可以用一个简单的公式来理解软件工程，那就是：**软件工程 = 工具 + 方法 + 过程。**

根据这个公式，我将软件工程的知识结构做成了思维导图，方便你对知识点有更好地理解，高效学习。



如何学习软件工程？

我给了你软件工程学的公式，也对软件工程有了更为全面的了解，看起来软件工程学很简单，但这些内容一下子要吃透也不容易。在开篇词中，我介绍了会从“道、术和器”三个维度去讲这个专栏，这其实对应了学习软件工程的四重境界。

学习软件工程的四重境界

第一重：用器

“器”就是工具，工具规则简单，上手就可以用，也很快就能看到效果。比如，原型设计工具可以帮助你确定需求，持续集成工具可以帮助你简化测试和部署的流程。对工具的学习是最为简单的，也是最基础的。

第二重：学术

“术”就是方法，学会方法，你就能应用方法去完成一个任务，例如用需求分析的方法，你去搞清楚用户想要什么，用 Scrum 去组织项目开发过程。

掌握了术，甚至是可以脱离器的，例如你没用原型设计工具，你用纸和笔，用白板，一样可以去沟通确认需求。

第三重：悟道

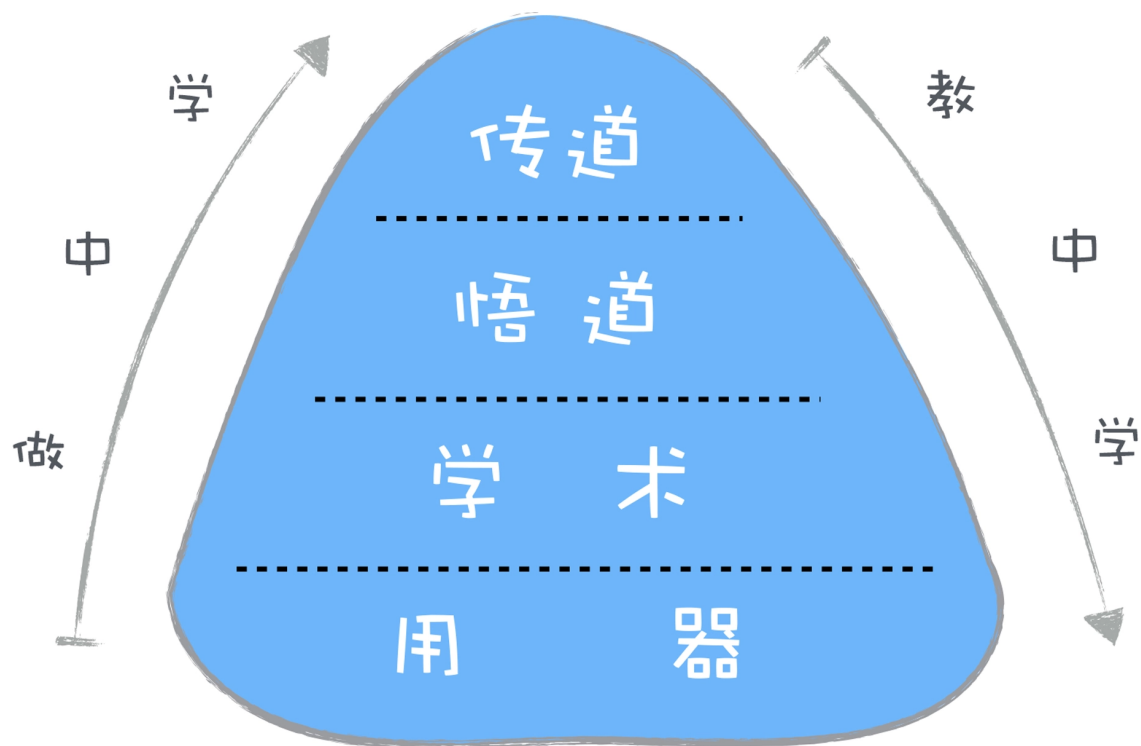
“道”就是本源，软件工程知识的核心思想和本质规律。就像敏捷开发，本身并不是一种方法，而是一套价值观和原则，领悟了这个道，就可以成为你在处理项目过程中各种问题决策的依据。道是可以产生术的，你掌握了敏捷开发的道，你就可以领悟出 Scrum、极限编程这样的术。

第四重：传道

当你能把复杂的知识通过浅显易懂的方式传授给别人，那就说明你对知识的领悟已经到了更高的境界。同时，教学也是最好的学习方式，通过传授别人知识，可以让你对知识本身有更深入的理解。

做中学和教中学

你可能会问，怎样学，才能到达以上这四重境界？我在做技术管理的工作中，经常要做一些培训的工作，在这过程中我总结了两套行之有效的方法：“做中学”和“教中学”。



“做中学”，是一种自下而上的学习方法，通过实践，从使用工具到学习方法，再从方法中提炼出道。

在学习本专栏的时候，你可以采用“做中学”的方式，把专栏中的知识应用起来，在实践的过程中去巩固你学到的知识，去思考背后的道。把已经积累的项目经验和软件工程的知识点关联起来，这样才能加深你的理解，学以致用，把经验和知识转化为能力。

“教中学”，是一种自上而下的学习方法，通过教学，去进一步深入领会别人总结出来的道，去模仿推导方法，去学习如何使用工具。

比如，你学习完一篇专栏文章后，把学到的知识进行输出，写成微博或博客分享出去；在公司内部讲给你的同事们听等。在教学分享的过程中，去进一步深化吸收知识内容，构建你的知识体系。

“做中学”和“教中学”，这两种方法你可以配合起来使用。

参考书目

另外，在学习软件工程的过程中，我看过一些不错的相关书籍，在这里列个书单，供大家参考。

《构建之法》

作者邹欣是微软的研发总监，同时在多所高校进行了软件工程的教学实践，在此基础上对软件工程的各个知识点和技能要求进行了系统性整理，形成教材。也是对本专栏知识很好的补充。

《人月神话》

这是软件工程历史上的经典著作，内容发人深省，40 年来一直畅销不衰，里面的观点即使到现在也不过时。这本书即使你以前看过，隔一段时间再翻看一遍，可能都会有新的感悟。

《人件》

如果说《人月神话》关注“软件开发”本身，《人件》则关注软件开发中的“人”。作者指出知识型企业的核心是人，而不是技术。

《知行合一：实现价值驱动的敏捷和精益开发》

作者丛斌有二十多年从事软件工程教学、咨询和研究的经验，所以书写的特别接地气，文章有很多真实案例，对敏捷开发和 CMMI 都有很深入描述。

《软件工程——实践者的研究方法》

这是大部分高校采用的软件工程标准教材，可以作为一个参考。

《持续交付》

讲述如何实现更快、更可靠、低成本的自动化软件交付，描述了如何通过增加反馈，并改进开发人员、测试人员、运维人员和项目经理之间的协作来达到这个目标。

《走出软件作坊》

这本书生动的描述了国内小型 IT 企业在发展过程中遇到的一系列项目管理问题，以及作者是如何去解决这些问题的。

总结

今天，我带你浏览了软件工程的全景图，也为你讲解了学习软件工程的四重境界。同时，我也介绍了“做中学”和“教中学”这两套行之有效，并且特别适合软件工程学科的学习方法，所以希望你在后面的学习中，可以付诸行动。

分享你学到的知识。 将你从专栏学习到的知识写成微博或博客等，分享给大家。写作是一种特别好的总结和学习方式，在你写的过程中，很多不清楚的问题就想明白了。

做几次内部分享或培训。 如果你从来没做过公司内部的分享或培训，不妨迈出第一步，把你学到的知识，写成 PPT，小范围地讲给你的同事或朋友。如果你已经做过类似的分享，那么就再做几次软件工程相关的。准备 PPT 的过程，就是你最好的学习过程。

把你学习的知识应用起来。 学到的知识只有用起来才能变成你自己的经验，尝试着把在专栏中学到的知识应用到你的项目中去。多问多思考。有疑问就提出来；看到其他人问的问题，也可以去思考为什么，一起探讨问题的答案。

感谢阅读，如果你觉得这篇文章对你有一些启发，也欢迎把它分享给你的朋友。

© 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。

精选留言 (27)



David Mao

2019-02-21

老师总结的器、术、道、传道四重境界很棒，结合自己的工作实践深有感悟。我是做软件测试的，最近在自学python开发，希望老师给一些建议，感谢。

作者回复: 看到你在学Python，不知道你所在团队自动化测试方面做的如何，我觉得这方面其实大有可为，可以极大的提高测试效率。

另外多学习一些软件工程知识，将来如果要兼顾或转型项目管理，也是不错的发展方向。



👍 17



alva_xu

2019-02-21

还有一本机械工业出版社出的《软件工程》，作者是Ian Sommerville，也特别好。已经出了十多版了吧。

作者回复：软件工程的书确实还有很多不错的，很遗憾没能一一例举，感谢推荐👍



👍 9



彭燕林

2019-02-21

原版是英文参考书籍能不能介绍一下英文信息，方便查找。

作者回复：你是说《软件工程——实践者的研究方法》吗？

Software Engineering – A Practitioner's Approach

作者是 Roger S. Pressman

如果专栏内容涉及英文专业术语的，一般会注明英文。如果对具体某个术语的英语有疑问的，可以留言，我会补充说明。



👍 5



一路向北

2019-02-21

老师说的学中教，教中学的4重境界，非常形象，容易让人领会。

实践的过程中，最难的应该是悟道了，期望在后面的学习中能够多悟道。这方面也是需要自己独立的深入思考才能产生的结果。

作者回复：是的，这是个漫长的过程。

“悟道”最重要的不是结果，而是这个实践和思考的过程。多实践多思考，长期看自然就会有很多收获的。

尤其是当你尝试去“传道”的时候，你可能会发现你突然“悟道”了。



Sam_Deep_Thinking

2019-10-06

关于软件工程的书籍，我推荐一本 周爱民先生的《大道至简》，讲的也很不错，作者思考的比较深，可以作为学习软件工程的参考书籍之一。

作者回复: 是的，写得很好，🙏感谢推荐



廖师虎

2019-04-14

CRUD是择指增删改查，基于此模式很难做到可审计，可溯源，只有临时的状态，不可重放恢复

作者回复: 如果要做到可信，其实和CRUD不矛盾，如果你想可审计、可溯源、可重访，可以在架构层面解决。

比如说基于分层架构，你添加一条记录，并不是直接写一条SQL，而是调用数据访问层的方法，这个方法会帮助你增加相应的日志、审计记录。这样从编程角度看，还是CRUD，但是从结果上来说，已经具备你的要求。包括以后如果要增加更多审计逻辑，只要修改你响应的架构，对于你业务层的代码几乎没什么影响。



rocedu

2019-03-27

过程分为瀑布模型和敏捷开发两类，合适吗？

作者回复: 从大类上来看，我觉得没什么问题。当然也想听听老师的意见。





2019-02-22

瓜大软微学弟前来报道~

作者回复: 学弟好! 我们那会还叫软件学院呢



👍 3



javaadu

2019-02-21

听老师的话, 已经整理一篇学习笔记。也分享下自己关于学习的心得:

1. 知行合一: 理论与实践相结合, 在软件开发中使用工具的时候, 要及时总结使用工具的技巧; 在经历过一个项目后, 要总结这个过程中对的得失经验。然后将这些经验和技巧应用到下一个项目中。
2. 持之以恒: 想要做到卓越, 不可能一蹴而就, 因此要持之以恒得践行知行合一。
3. 乐于分享: 将自己的心得体会讲给别人, 可以用博客、短视频、内部培训的方式, 在讲给别人的过程中, 也是对知识的重新梳理和查缺补漏。

作者回复: 赞👍

改变, 从改变习惯开始



👍 3



E

2019-03-23

请问老师, 软件开发的过程和方法之间的关系是什么?

作者回复: 软件开发过程就是指你开发软件时的整个过程的开发模式, 比如说瀑布模型还是敏捷开发。

选择了开发过程, 你就需要有具体方法来执行。

比如你选择了瀑布模型, 整个软件开发过程就是按照瀑布模型的分阶段来进行, 对应的方法就是瀑布模型中的方法, 例如需求分析、架构设计; 如果你选择了敏捷开发, 则整个开发过程就是一种敏捷迭代方式, 后面的方法对应的就是敏捷开发的一套方法体系, 例如Scrum、用户故事、持续集成等。



👍 2

gfkdcadet



2019-03-02

买了课程相关的全部6本参考书籍，也算是全身心投入了！

作者回复：赞，有疑惑的地方也欢迎留言提问：)



👍 2



cxy

2019-02-21

老师，那思维导图中的工具分类，不知道是否可以把常用的工作具软件名加进去，那样可以我们让有目的地去细化学习。

作者回复：我在后面的文章中会有各个阶段常用工具的介绍，因为还在整理中，等到结束后会考虑重新编辑加上名字。



👍 2



技术修行者

2019-12-01

1. 做中学和学中做，用器->学术->悟道->传道，提炼得挺好的。
2. 主动分享，教学相长。
3. 推荐的书目，看上去大部分都是比较老的，算是经典了。例如《走出软件作坊》，我记得是2010年左右买的，《构建之法》大概是4、5年前买的。



👍 1



打工皇帝

2019-09-17

聚焦质量 清楚过程 知道方法 会用工具



👍 1



梁中华

2019-03-17

不知道怎么搞得，现在都不提RUP了，当年可是很火的，以用例和架构驱动的软件开发过程，大型软件开发中还是比较适合的。

作者回复：RUP是很好的模式，但大型软件开发项目毕竟相对少，而且RUP的实施也需要专业的支持。不像瀑布模型那样简单易行，也不像敏捷那样适用面广，遍地开花。



ifelse

2022-06-16

软件工程=过程+方法+工具



edward

2022-05-02

老师总结的真好！



Geek_41c9f0

2020-05-02

希望自己能够坚持学习下去，真正做出改变，以期看到自己能力上的变化。

作者回复：祝学习顺利，有问题也欢迎通过留言提出！



bidinggong

2020-03-27

软件工程 = 工具 + 方法 + 过程。“做中学”和“教中学”结合。老师言传身教！谢谢



贵州小欧哥

2020-03-26

从道、术、器来讲，深入浅出。易理解

