

加餐6 | 这15年来，我是如何在工作中学习技术和英语的？

2020-05-16 朱晔

Java业务开发常见错误100例

[进入课程 >](#)



讲述：王少泽

时长 16:33 大小 15.17M



你好，我是朱晔。今天，我来和你聊聊如何在工作中，让自己成长得更快。

工作这些年来，经常会有同学来找我沟通学习和成长，他们的问题可以归结为两个。

一是，长期参与 CRUD 业务开发项目，技术提升出现瓶颈，学不到新知识，完全没有办法实践各种新技术，以后会不会被淘汰、找不到工作？

二是，英语学得比较晚，大学的时候也只是为了应试，英语水平很低，看不了英文的资料，更别说到外企找工作了。



不知道你是不是也面临这两个问题呢？今天，我就通过自己的经历和你分享一下，如何利用有限的环境、有限的时间来学习技术和英语？

学好技术

在我看来，知识网络的搭建就是在造楼房：基础也就是地基的承载力，决定了你能把楼造多高；广度就像是把房子造大、造宽；深度就是楼房的高度。因此，如果你想要提升自己的水平，那这三个方面的发展缺一不可。

第一，学习必须靠自觉。

虽说工作经历和项目经验是实践技术、提升技术的一个重要手段，但不可能所有的工作经历和项目都能持续地提升我们的技术。所以，我们要想提升自己的技术水平，就必须打消仅仅通过工作经历来提升的念头，要靠业余时间主动地持续学习和积累来提升。

比如，你可以针对项目中用到的技术，全面阅读官方文档，做各种 Demo 来论证其技术特性。在这个过程中，你一定还会产生许多技术疑问，那就继续展开学习。

第二，不要吝啬分享。

刚毕业那会，我花了很多时间和精力在 CSDN 回答问题，积极写博客、写书和翻译书。这些经历对我的技术成长，帮助非常大。

很多知识点我们自认为完全掌握了，但其实并不是那么回事儿。当我们要说出来教别人的时候，就必须 100% 了解每一个细节。因此，分享不仅是帮助自己进一步理清每一个知识点、锻炼自己的表达能力，更是一种强迫自己学习的手段，因为你要保证按时交付。

当然了，分享的过程也需要些正向激励，让自己保持分享的激情。就比如说，我获得的几次微软 MVP、CSDN TOP3 专家等荣誉，就对我激励很大，可以让我保持热情去不断地学习并帮助别人。

第三，不要停留在舒适区。

分享一段我的真实经历吧。我加入一家公司组建新团队后，在做技术选型的时候，考虑到成本等因素，放弃了从事了七年的 .NET 技术，转型 Java。有了 .NET 的积累，我自己转型

Java 只用了两周。其实，一开始做这个决定非常痛苦，但是突破自己的舒适区并没有想象得那么困难。随后，我又自学了 iOS、深度学习、Python 等技术或语言。

随着掌握的技术越来越多，这些技术不但让我触类旁通，更让我理解了技术只是工具，解决问题需要使用合适的技术。因此，我也建议你，利用业余时间多学习几门不同类型的编程语言，比如 Java、Python 和 Go。

有些时候，我们因为恐惧跳出舒适区而不愿意学习和引入合适的新技术来解决问题，虽然省去了前期的学习转型成本，但是后期却会投入更多的时间来弥补技术上的短板。

第四，打好基础很重要。

这里的“基础”是指，和编程语言无关的那部分知识，包括硬件基础、操作系统原理、TCP/IP、HTTP、数据结构和算法、安全基础、设计模式、数据库原理等。学习基础知识是比较枯燥的过程，需要大块的时间来系统阅读相关书籍，并要尝试把学到的知识付诸实践。只有实践过的技术才能映入脑子里，否则只是书本上的知识。

比如，学习 TCP/IP 的时候，我们可以使用 Wireshark 来观察网络数据。又比如，学习设计模式的时候，我们可以结合身边的业务案例来思考下，是否有对应的适用场景，如果没有能否模拟一个场景，然后使用所有设计模式和自己熟悉的语言开发一个实际的 Demo。

这些看似和我们日常业务开发关系不大的基础知识，是我们是否能够深入理解技术的最重要的基石。

第五，想办法积累技术深度。

对开发者而言，技术深度体现在从一个框架、组件或 SDK 的使用者转变为开发者。

虽然不建议大家重复去造轮子、造框架，但我们完全可以阅读各种框架的源码去了解其实现，并亲手实现一些框架的原型。比如，你可以尝试把 MVC、RPC、ORM、IoC、AOP 等框架，都实现一个最基本功能点原型。在实现的过程中，你一定会遇到很多问题和难点，然后再继续研究一下 Spring、Hibernate、Dubbo 等框架是如何实现的。

当把自己的小框架实现出来的那一刻，你收获的不仅是满满的成就感，更是在技术深度积累上的更进一步。在这个过程中，你肯定会遇到不少问题、解决不少问题。有了这些积累，之

后再去学习甚至二次开发那些流行的开源框架，就会更容易了。

除了实现一些框架外，我还建议你选择一个中间件（比如 Redis、RocketMQ）来练手学习网络知识。

我们可以先实现它的客户端，用 Netty 实现 TCP 通信层的功能，之后参照官方文档实现协议封装、客户端连接池等功能。在实现的过程中，你可以对自己实现的客户端进行压测，分析和官方实现的性能差距。这样一来，你不仅可以对 TCP/IP 网络有更深入的了解，还可以获得很多网络方面的优化经验。

然后，再尝试实现服务端，进一步加深对网络的认识。最后，尝试把服务端扩展为支持高可用的集群，来加深对分布式通信技术的理解。

在实现这样一个分布式 C/S 中间件的过程中，你对技术的理解肯定会深入了许多。在这个过程中，你会发现，技术深度的“下探”和基础知识的积累息息相关。基础知识不扎实，往深了走往往会步履维艰。这时，你可以再回过头来，重新系统学习一些基础理论。

第六，扩大技术广度也重要。

除了之前提到的多学几门编程语言之外，在技术广度的拓展上，我们还可以在两个方面下功夫。

第一，阅读大量的技术书籍。新出来的各种技术图书（不只是编程相关的），一般我都会买。十几年来，我买了 500 多本技术图书，大概有三分之一是完整看过的，还有三分之一只翻了一个大概，还有三分之一只看了目录。

广泛的阅读，让我能够了解目前各种技术的主流框架和平台。这样的好处是，在整体看技术方案的时候，我可以知道大家都在做什么，不至于只能理解方案中的一部分。对于看不完的、又比较有价值的书，我会做好标签，等空闲的时候再看。

第二，在开发程序的时候，我们会直接使用运维搭建的数据库（比如 Elasticsearch、MySQL）、中间件（比如 RabbitMQ、ZooKeeper）、容器云（比如 Kubernetes）。但，如果我们只会使用这些组件而不会搭建的话，对它们的理解很可能只是停留在 API 或客户端层面。

因此，我建议你去尝试下从头搭建和配置这些组件，在遇到性能问题的时候自己着手分析一下。把实现技术的前后打通，遇到问题时我们就不至于手足无措了。我通常会购买公有云按小时收费的服务器，来构建一些服务器集群，尝试搭建和测试这些系统，加深对运维的理解。

学好英语

为啥要单独说英语的学习方法呢，这是因为学好英语对做技术的同学非常重要：

国外的社区环境比较好，许多技术问题只有通过英文关键字才能在 Google 或 Stackoverflow 上搜到答案；

可以第一时间学习各种新技术、阅读第一手资料，中文翻译资料往往至少有半年左右的延迟；

参与或研究各种开源项目，和老外沟通需要使用英语来提问，以及阅读别人的答复。

所以说，学好英语可以整体拓宽个人视野。不过，对于上班族来说，我们可能没有太多的大块时间投入英语学习，那如何利用碎片时间、相对休闲地学习英语呢？还有一个问题是，学好英语需要大量的练习和训练，但不在外企工作就连个英语环境都没有，那如何解决这样的矛盾呢？

接下来，我将从读、听、写和说四个方面，和你分享一下我学习英语的方法。

读方面

读对于我们这些搞技术的人来说是最重要的，并且也是最容易掌握的。我建议你这么学：

先从阅读身边的技术文档开始，有英语文档的一定要选择阅读英语文档。一来，贴近实际工作，是我们真正用得上的知识，比较容易有兴趣去读；二来，这些文档中大部分词汇，我们日常基本都接触过，难度不会太大。

技术书籍的常用词汇量不大，有了一些基础后，你可以正式或非正式地参与翻译一些英语书籍或文档。从我的经验来看，翻译过一本书之后，你在日常阅读任何技术资料时基本都不需要查字典了。

订阅一些英语报纸，比如 ChinaDaily。第一，贴近日常生活，都是我们身边发生的事儿，不会很枯燥；第二，可以进一步积累词汇量。在这个过程中，你肯定需要大量查字

典打断阅读，让你感觉很痛苦。但一般来说，一个单词最多查三次也就记住了，所以随着时间推移，你慢慢可以摆脱字典，词汇量也可以上一个台阶了。

技术方面阅读能力的培养，通常只需要三个月左右的时间，但生活方面资料的阅读可能需要一年甚至更长的时间。

听方面

读需要积累词汇量，听力的训练需要通过时间来磨耳朵。每个人都可以选择适合自己的材料来磨耳朵，比如我是通过看美剧来训练听力的。

我就以看美剧为例，说说练听力的几个关键点。

量变到质变的过程，需要 1000 小时的量。如果一部美剧是 100 小时，那么看前 9 部的时候可能都很痛苦，直到某一天你突然觉得一下子都可以听懂了。

需要确保看美剧时没有中文字幕，否则很难忍住不看，看了字幕就无法起到训练听力的效果。

在美剧的选择上，可以先选择对话比较少，也可以选择自己感兴趣的题材，这样不容易放弃。如果第一次听下来，听懂率低于 30%，连理解剧情都困难，那么可以先带着中文字幕看一遍，然后再脱离字幕看。

看美剧不在乎看的多少，而是要找适合的素材反复训练。有人说，反复看 100 遍《老友记》，英语的听说能力可以接近母语是英语的人的水平。

如果看美剧不适合你的话，你可以选择其他方式，比如开车或坐地铁的时候听一些感兴趣的 PodCast 等。

总而言之，选择自己喜欢的材料和内容，从简单开始，不断听。如果你有一定词汇量的话，查字典其实不是必须的，很多时候不借助字典，同一个单词出现 10 遍后我们也就知道它的意思了。

一定要记住，在积累 1000 小时之前，别轻易放弃。

写方面

如果有外企经历，那么平时写英语邮件和文档基本就可以让你的工作英语过关；如果没有外企经历也没关系，你可以尝试通过下面的方式锻炼写作：

每天写英语日记。日记是自己看的，没人会嘲笑你，可以从简单的开始。

在保持写作的同时，需要确保自己能有持续的一定量的阅读。因为，写作要实现从正确到准确到优雅，离不开阅读的积累。

写程序的时候使用英语注释，或者尝试写英语博客，总之利用好一切写的机会，来提升自己的英语表达。

再和你分享一个小技巧。当你要通过查词典知道中文的英语翻译时，尽量不要直接用找到的英文单词，最好先在英语例句中确认这个翻译的准确性再使用，以免闹笑话。

说方面

训练说英语的机会是最少的，毕竟身边说英语的人少，很难自己主动练习。

这里我和你分享两个方法吧。

第一是，买外教的 1-1 对话课程来训练。这些课程一般按小时计费，由母语是英语的人在线和你聊一些话题，帮助你训练对话能力。

买不买课程不重要，只要能有母语是英语的人来帮你提升就可以。同时，大量的听力训练也可以帮助你提升说的能力，很多英语短句经过反复强化会成为脱口而出的下意识反应。所以，你会发现在听力达到质变的时候，说的能力也会上一个台阶。

第二，大胆说，不要担心有语法错误、单词发音问题、表达不流畅问题而被嘲笑。其实，你可以反过来想想，老外说中文时出现这些问题，你会嘲笑他吗。

这里有一个技巧是，尽量选用简单的表达和词汇去说，先尝试把内容说出来，甚至是只说几个关键字，而不是憋着在脑子里尝试整理一个完整的句子。灵活运用有限的单词，尽可能地流畅、准确表达，才是聪明的做法。

总结

最后我想说，如果你感觉学得很累、进步很慢，也不要放弃，坚持下来就会越来越好。我刚毕业那会儿，有一阵子也对 OOP 很迷茫，感觉根本无法理解 OOP 的理念，写出的代码完全是过程化的代码。但我没有放弃，参与写了几年的复杂业务程序，再加上系统自学设计模式，到某一个时刻我突然就能写出 OOP 的业务代码了。

学习一定是一个日积月累、量变到质变的过程，希望我分享的学习方法能对你有启发。不过，每个人的情况都不同，一定要找到适合自己的学习方式，才更容易坚持下去。

持续学习很重要，不一定要短时间突击学习，而最好是慢慢学、持续积累，积累越多学习就会越轻松。如果学习遇到瓶颈感觉怎么都学不会，也不要沮丧，这其实还是因为积累不够。你一定也有过这样的经验：一本去年觉得很难啃的书，到今年再看会觉得恰到好处，明年就会觉得比较简单，就是这个道理。

我是朱晔，欢迎在评论区与我留言分享你学习技术和英语的心得，也欢迎你把今天的内容分享给你的朋友或同事，一起交流。

课程预告

6月-7月课表抢先看

充 ¥500 得 ¥580

赠「¥ 118 月球主题 AR 笔记本」



【点击】图片，立即查看>>>

© 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 (15)

写留言



2020-05-18

个人看法：

1.学习的自觉。这个我觉得专栏的读者倒是都没问题，大家基本上都是多少有自我学习习惯的。

2.分享。难易度从低到高我认为 回答他人的问题->博客->公众号文章->个人主页->真正的创作。我个人从一开始是，单纯帮别人解答一些小儿科的问题。然后开始尝试写博客...
展开

作者回复: 赞



3



云韵

2020-05-16

朱老师，podcast 有没有推荐的软件

展开

作者回复: 我之前听的有如《Opening Language（英语版本）》《English as a Second Language》



3



刘大明

2020-05-16

按照老师的方法学习，希望我也能成为一个像老师这么强的人。

作者回复: 😊



3



Darren

2020-05-18

逃离舒适区这点我比较有感触：

14年毕业去了深圳，一家国企，技术很老，没有缓存，没有消息中间件，没有微服务.....，因为确定要从深圳离开，虽然那时候超级轻松，每天的活半个小时就能搞定，但还是毅然选择了离职，怕温水煮青蛙，拖下去怕找不到工作；很快就从深圳离职，回到一个二线城市，新公司用到redis、springboot、rabbitmq等，因为不会从头学起，呆了2年多，个...
展开 ▾

作者回复: 📬



👍 2



Demon.Lee

2020-05-19

学习没有捷径，但可以有方法，一起努力吧。

我现在就是把java相关的课程第一遍都理论学习，因为很多概念不知道，走一遍之后，大致有一个印象；

第二遍就是把案例都跑一遍，这一遍会花比较长的时间，因为要去把一些经典书上相关的内容也看一遍； ...

展开 ▾



👍 1



许童童

2020-05-17

老师说得真好

展开 ▾



👍 1



终结者999号

2020-05-17

老师真的厉害📬，很多都说到心里的想法，成为一个大神，要有技术深度和广度



👍 1



无名之辈

2020-05-17

给大佬这样的人点赞

展开 ▾



👍 1



风轻扬

2020-05-17

想成为大神，就得承受孤独，持续努力！所有大神都是这么过来的。我也要成为大神，我会重走老师的路！

作者回复: 加油

◀ ▶



👍 1



Wiggle Wiggle

2020-05-16

学习更多编程语言前，建议先对自己的主力编程语言达到运用自如的程度。本科时先学的c++，当时也就一知半解，然后又让学Python，结果就是觉得Python也很难，勉强做完作业就放弃了.....后来自己选择了Java，自学了网上商城项目，对编程语言有了自己的理解，然后研究生阶段要用Python做AI研究，很快就达到了会用的水平，基本没怎么花时间在语言的学习上，还在对比之下深切体会到了脚本语言的便捷

展开 ▾



👍 1



青柠

2020-05-23

谢谢老师的加餐，最近也准备学英语正在被不知道从哪里下手，就看到了老师的文章，如春雨般滋润心田，谢谢无私分享

展开 ▾

作者回复: 不客气 如果觉得有帮助可以点赞转发

◀ ▶



认知更正

2020-05-23

我想问的是底层的诸如操作系统，网络这些知识，您能推荐一些书吗？

作者回复: TCP/IP详解、鸟哥的Linux、现代操作系统、深入理解计算机系统、UNIX环境高级编程、MySQL技术内幕

◀ ▶



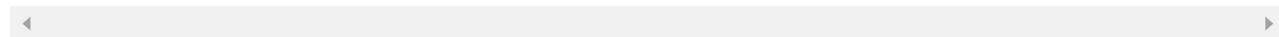
认知更正

2020-05-23

关于打基础的部分，有什么比较好的书籍推荐吗？

展开 ▾

作者回复: 如果是Java的话，可以搜索Java四大名著。



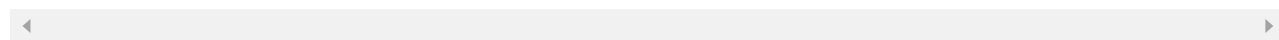
汝林外史

2020-05-23

老师推荐的pod那是app还是啥？

展开 ▾

作者回复: 中文 播客



Geek_3b1096

2020-05-19

尝试写博客写英语博客坚持1000小时

展开 ▾

