

加餐四-聊一聊Google那些让我快速成长的地方

我一直强调，在项目中实践、由点及面的学习，是提高技术最有效的途径。但是，好的技术氛围、培养机制，也能打破通过项目单一成长的限制，获得更多项目之外的收获。很多公司，特别是一些大公司，在关注员工工作效率、工作产出的同时，也十分关注员工的个人成长、职场发展。今天，我就和你聊一聊Google有哪些让我快速成长的地方。

完善的培训课程

Google内部的课程是非常多的。有些是线下分享，有些是线上录播课程。在内容方面，课程也多种多样，有新技术的DogFood（比如我当年就参加过Go语言的），有入门级的101教程（比如针对后端工程师的前端入门课），还有比较有深度的系列教程（比如AI相关的课程）。所有的课程，只要你感兴趣，都可以自由选择来学习。

而且，最近我还听说，Google有个“G2G”学习计划，全称是Googler to Googler，意思就是Googlers之间互相学习。只要有值得分享的东西，都可以录制成视频分享给其他同事。同理，你也可以从其他同事那里学习他们分享的技术。

总之，Google内部有大把的优质课程供你学习，这点对于好学的人来说，是非常幸福的一件事情。而且，得益于以结果为导向的企业文化，Google非常鼓励员工自我充电学习，不像有些公司反对员工占用上班时间来学习。

公开的文档和代码

除了各种线下和线上的课程之外，Google还有另外一块非常宝贵的学习资料，那就是文档和代码。在Google，除了特别核心的一些代码，比如跟搜索质量相关的代码，几乎所有的文档和代码都是公开的，你可以随意查看某个感兴趣的项目的设计文档和代码，比如Google Adwords的文档和代码等等。

你可能会说，Google就不怕员工泄露代码吗？在这一点上，Google相对是比较信任员工的。毕竟在招聘入口上做了过滤，员工本身的素质都不差，这点我们在下一篇加餐中还会讲到。

我们知道，Google有非常多优秀的开发框架，有些是开源的，有些是还没有公布、只限内部使用的。不管你对哪个开发框架感兴趣，你都可以公司内部获取到一手的文档资料。通过阅读别人的文档，除了学习这些框架如何使用之外，我们还能学到很多关于设计、架构、解决方案等方面的经验和知识。

实际上，我觉得，代码公开最大好处并非是，你没事的时候就可以阅读别人的代码，毕竟泛泛地去读，效率也不会很高，收获也而并不会很大，读完就忘了。相反，我一直强调，带着问题去学习是更有效的一种学习方法。所以，我觉得代码公开最大的好处是，当你想要实现某个功能的时候，或者为实现某个功能绞尽脑汁的时候，你只需要搜索一下代码仓库，就可以找到很多类似的优质代码做参考。对于很多新人或者初级工程师来说，参考别人的设计和实现，这样一方面可以提高开发速度、保证代码质量，另一方面也可以从高手那里学习好的设计思路 and 实现技巧，提高自己的设计和实现能力。

相反，据我所知，很多公司的代码都是限制访问的，员工只能查看一小部分相关项目的代码，极端情况下，只能阅读自己参与的项目的代码。相对而言，Google就“大方”很多。除了Code Review之外，我觉得，从n多Google高手的代码中学习设计和实现，是提高代码水平的另一个非常重要的渠道，也是让我代码能力成长最快的地方之一。

清晰的成长路径

除了有那么多课程、文档、代码来供我们自我学习之外，Google也非常关注员工的个人成长，希望能跟员工共同进步，而不只是将员工当成达成目标的工具。所以，在不同的阶段，公司会帮组员工制定不同的成长计划。

我们知道，像国内的一些大公司，比如BAT，在新员工入职的时候，一般都会有集中几天的入职培训。我当年加入Google的时候并没有这样的培训。我只参加过一次TGIF（Thanks God It's Friday）上NewGoogler的欢迎仪式。

Google虽然没有集中的培训，但对于新入职的员工，公司会有一个新人的学习计划，主要涉及编码规范、单元测试、Code Review、开发工具、行为准则等方面的内容。我觉得，这一点对新人来说还是挺友好的。新人可以根据指引，一步一步完成学习计划里的内容，不会因为刚入职还未参加项目，而觉得无事可做。除此之外，在新员工入职时，公司还会分配一个同事作为mentor（导师）。任何工作、生活中的问题，你都可以向他请教，帮助你快速地融入公司。

除此之外，在工作中，每隔半年，你的Leader还会跟你一块制定OKR，也就是接下来半年你要做的事情，除了工作内容，这里面还会包含你的个人成长部分，比如你可以学习数据挖掘相关的知识，把这部分内容直接写到OKR中，这部分甚至可以跟你的工作没有任何关系，只是你对自己的自我提升。

不仅如此，你的Leader还会帮你制定升职计划。如果你有意申请下一轮晋升，领导一般会提前跟你一块规划，如何做才能达成这个目标，比如做什么项目、做哪些有影响力的事情、管理多大的团队等等，也会有意安排一些帮你晋升的事情给你做。

为了不让工程师一直呆在舒适区，Google还非常鼓励内部转岗，希望员工不要一直做一个项目，鼓励员工跳出舒适区，换岗到其他团队，用不熟悉的语言、不熟悉的技术，去做一些不熟悉的项目。虽然我们知道，内部转岗机制也并非Google所特有的，很多公司都有，但是，真的执行得很顺畅的却不多。大部分公司都是跟风喊喊口号，员工真的要转岗的时候，条条关卡、困难重重。

上面的这些机制，貌似也没有什么特别的，很多公司都有。我要特别说的是，在Google，一对一的沟通非常多，特别是跟自己的Leader或者Manager，一般一两周就有一次，聊一聊自己这周的工作、想法、迷惑。这种沟通也能让你把工作、个人成长中的困难、疑问，及时反馈给领导，领导也可以及时地了解你的想法，对你进行指导，这也能避免很多工程师闷头干活，感觉不爽就立刻离职的情况。

课堂讨论

在你的职业生涯中，你觉得哪段工作经历对你的个人成长最有帮助？为什么呢？

欢迎留言和我分享你的想法。如果有收获，也欢迎你把这篇文章分享给你的朋友。

精选留言：

● Jxin 2020-06-26 01:42:59

1.用近三个月的时间，在迭代中持续重构了近一半的项目代码。并与产品和测试配合，将原有的黑盒业务代码重构，输出白皮书，并最终借着多渠道接入将单流程项目切换成可配置流程的项目。期间没有少发一个需求，也没有额外申请一天用于重构。

2.影响就是，对持续重构，有信心了。

将大的工作拆细，逐步迭代完成，有勇气了。
对边界的界定，项目分层，代码归属更明确了。
码代码考虑的东西更多了。
代码置顶向下的搭建方式更熟练了。
写出来的代码去重，可读和可扩展都有不错的长进。

题外话：

最近在想两个问题。

1.为什么公司的项目烂得出奇。有一个自洽的结论。烂是因为不重视，不重视是因为价值不高，价值不高是因为公司的商业模式只是用到这个软件平台而不是依托这个软件平台。所以站在公司的角度看，公司的价值增长是因为商业模式中其它元素的价值增长，而资金和精力应该往价值增长高的元素流。故而软件平台相对不重要，研发中心不重要，项目不重要，代码质量不重要。反过来想，如果这个软件平台是价值增长高的元素，那么质量自然会上去（要么现在要么将来，在合适的时候）。

2.业务开发将来会怎么样。随着云原生发展，技术复杂性与业务在不断的分离透明。也许不久的将来，业务项目只有业务复杂性这一问题。那么业务开发的门槛会越来越低。并且结合一定的规范约定和类似元编程的手段，是能让初级开发写出中高级开发差不多水平的代码的。那么中高级开发的生存空间是不是越来越窄了？搬砖会不会不再是自嘲了。

[2赞]

- Jackey 2020-06-26 00:27:33
喜欢这样技术氛围好的公司 [2赞]
- 迷羊 2020-06-26 10:29:52
在我的职业生涯中，学习小争哥的专栏对我的个人成长最有帮助。 [1赞]
- GeekAml 2020-06-26 08:53:45
居然发现，美团这边和Google如此相像。 [1赞]