

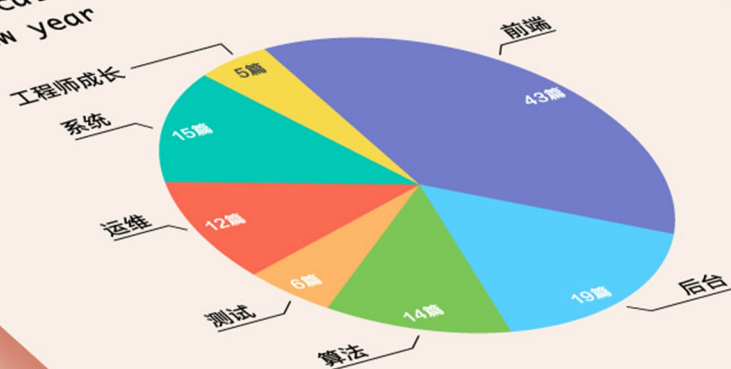
美团点评 2018 技术年货

CODE A BETTER LIFE



2018 美团技术团队答卷

```
System.out.println  
("114 technical articles for you in 2018");  
//Happy new year
```



序

春节已近，年味渐浓。

又到了我们献上技术年货的时候。

不久前，我们已经给大家分享了技术沙龙大套餐，汇集了过去一年我们线上线下技术沙龙 [99位讲师](#)、[85个演讲](#)、[70+小时](#) 分享。

今天出场的，同样重磅——技术博客全年大合集。

2018年，是美团技术团队官方博客第5个年头，[博客网站](#) 全年独立访问用户累计超过300万，微信公众号（meituantech）的关注数也超过了15万。

由衷地感谢大家一直以来对我们的鼓励和陪伴！

在2019年春节到来之际，我们再次精选了114篇技术干货，制作成一本厚达1200多页的电子书呈送给大家。

这本电子书主要包括前端、后台、系统、算法、测试、运维、工程师成长等7个板块。疑义相与析，大家在阅读中如果发现Bug、问题，欢迎扫描文末二维码，通过微信公众号与我们交流。

也欢迎大家转给有相同兴趣的同事、朋友，一起切磋，共同成长。

最后祝大家，新春快乐，阖家幸福。



目录 – 工程师成长篇

人物志 美团首席科学家夏华夏：不断突破边界的程序人生	4
人物志 美团女技术总监任登君：不要给自己的人生设限	16
人物志 技术十年：美团第一位前端工程师潘魏增	22
写给工程师的十条精进原则	29
工作中如何做好技术积累	37

人物志 | 美团首席科学家夏华夏：不断突破边界的程序人生

作者: 技术学院

“成长没有什么秘笈，就是坚持不断地一点点突破自己的边界就好。”

这是美团首席科学家、无人配送部总经理夏华夏在刚刚过去的“1024 程序员节”时送给技术同行的一句话。

这也是夏华夏自己的人生写照：从没摸过计算机的山东高考状元到清华计算机系的学霸，从美国名校深造、Google修炼6年到选择回国，从加入当时还很小的美团到负责公司最大业务的总体架构，从架构师转为无人配送这个前沿业务部门的管理者，夏华夏就是在不断突破自己的边界，做出人生的重要抉择，脚踏实地，一步步成长为业界知名的技术领军人物。

本文由美团技术学院基于夏华夏的访谈记录整理而成。



华夏老师

1998年，夏华夏从清华大学远赴美国留学，先后在谷歌、百度担任架构师。2013年受美团创始人王兴和穆荣均的邀请，选择加入当时还很“弱小”的美团，夏华夏回忆，是王兴的一番话让他备受触动。当时，望着五道口下面熙熙攘攘的人流，王兴说：“其实很多做互联网创业的人，很少考虑怎么去帮助这些人，这些普罗大众，这么多的小商家。其实，他们都在努力改变自己的命运，我们美团就要帮助他们，帮助这些普普通通的老百姓。这也是属于我们的机会。”

夏华夏加入美团已有5年的时间，也是一名地地道道的美团“老人”了。他当时并没有料到，如今这家小小的创业公司，已经成功在香港上市，成为市值仅次于BAT的第四大互联网企业；他当时也没有想到，这个曾经很小的技术团队已经迅速成长为一个拥有万人规模的一流研发组织；他也没有料到，5年后这家公司还会在人工智能、无人驾驶这些前沿科技领域不断开疆拓土，他也成为了无人配送项目的负责人。

- 2013年，美团的主要业务还是团购，夏华夏帮助整个技术团队做了很多技术层面的梳理和重构，并和早期的几位技术团队负责人一起组建了美团技术学院。
- 2015年，经过前期的孵化和运营，外卖业务已经初具规模，但由于基础系统建设不够牢固，导致当时的外卖系统很不稳定，每周要宕机好几次，而且好几次宕机都是发生在用餐高峰时期。临危受命，夏华夏再次被调任到外卖业务部门，帮助外卖技术团队解决了系统稳定性的问题。顶着巨大的业务压力，夏华夏和美团外卖很多技术骨干天天泡在一个被他们称之为“作战室”的大会议室里，经过了近百个不眠不休的夜晚，不断迭代升级、测试、监控整个系统。终于将系统稳定性就从98%提高到了99.9%，从而保障了外卖业务的飞速增长。美团能够成功上市，外卖团队功不可没。
- 2017年年底，夏华夏再次踏上了新的技术征程，他陆续交接了手中其他的工作，将全面精力放在美团无人配送项目上，并带领技术团队研发了专属于美团自己的无人车和无人机。目前夏华夏负责的美团无人配送开放平台，集合了政府、高校、企业三方力量，已经吸引包括清华大学、加州伯克利大学、北京智能车联产业创新中心、华夏幸福、Segway等近20家国内国外合作伙伴加入。

对神秘事物充满好奇，误打误撞报考清华计算机专业

1993年6月，夏华夏面临人生的第一次选择——高考。“当时在小县城，几乎没有人会对专业有概念，大家只在意你考上哪所大学。”对年轻的夏华夏来说，报专业纯属瞎报，当时是看哪个专业的名字有趣就选择一个，而选择计算机是完全是“蒙的”，要知道在上世纪90年代的小县城，大家都没有见过计算机，最多只是听过这个概念。“我觉得，计算机是一个非常神秘的东西，越是神秘的东西就越有意思，对我的吸引力也更大。”夏华夏说，当时他的班主任也不知道哪个系好，在班主任看来，有学生能考上清华大学，“KPI”就完成了。所幸在高考中，夏华夏发挥出色，以山东省第一名的成绩顺利考入清华大学。

进入清华实验班，系院士领入计算机世界的大门

清华有一个实验班，入学摸底考试后，从电子类相关的专业选择了大概五十个同学有资格进入，夏华夏名列前茅。“我们很幸运，因为实验班给我们创造了很好的条件，而且可以直接向系里云集的院士泰斗（张钹、李三立、唐泽圣等老师）请益。”

不过，最开始夏华夏都不知道应该请教什么问题，因为这是他第一次跟计算机世界打交道。现在回忆起第一次上级课的情景，夏华夏记忆犹新，当时电脑配置的是386的处理器，找了老半天都不知道怎么开机，怎么进入系统。

90年代，几乎所有学编程的人都是从C语言开始。这门看起来有些古老的语言，一度耗费了夏华夏几乎全部的精力。“那个时候还不会双打，用两个指头打字，所以编程学习特别慢，现在已经很难想象那种情景了”。人生第一个应用程序，花了夏华夏整整一节课的时间，他记得很清楚，那是一个画图的程序，编译之后也没问题，但是后续检查中，发现目录里有两个没有见过的文件，“.”和“..”（DOS操作系统下的当前目录和父目录），貌似感染了“病毒”，吓得赶紧执行了一个“deltree .”的命令，试图把“病毒”删除，然后就发现整个程序都找不到了。所以夏华夏的第一个程序，其实是一个失败项目。

虽然最开始很窘迫，但一个全新的世界已经向这个懵懂的少年敞开了大门。

小试牛刀，做数据库系统挣到人生的第一桶金

当然，对刚刚接触编程世界的夏华夏来说，每个项目都会倾注很多的精力去完成，而且实验室学习氛围很好，所有同学都会把项目看得非常重要。现在，机器学习、图像识别、人工智能的概念几乎已经众人皆知了，当时还没有这么火爆，其实夏华夏很早就参与了一个人工智能相关的项目，严格来说属于图像识别范畴，项目要求识别一张图上的飞鸟的总数。这是一个难度不小的挑战，夏华夏使用了很多模型进行实践，当时也没有“模式识别”方法可以使用，后来他通过应用算法把“鸟”的边界线数出来，然后逐渐对周围进行“腐蚀”，“腐蚀”到最小的一个点，就认为这是一个“鸟”。夏华夏一直都相信，只要勤思考，肯定能找到解决问题的办法。

在那个年代，人工智能正在经历又一次的低谷期，绝大多数保送的研究生都不愿意去人工智能实验室。“其实很多时候，我们对某些事的认知，确实是受制于环境因素的。可能没有人能够想象到，二十多年后，人工智能领域的人才，已经成为了最为稀缺的资源。”夏华夏非常感叹。

后来，夏华夏又开始接触了Windows编程，做了一个非常炫酷的界面系统，被系里很多同学复制，要知道当时可视化的编程环境比如Visual C++等还没有进入中国，能够做出这种效果，在大家看来已经属于“高手”了。所以，“钱”也开始找上门了。

对夏华夏来说，印象最深的就是大学时做的一个软件外包项目，这是一个仓库的管理查询项目，老板希望通过一个管理软件，可以将货物输入到数据库系统，能够查询货品的信息和库存信息。夏华夏当时刚刚读大二，他就利用几个月的时间，学习了编程和数据库相关的技术，然后通过各种途径学习研究，搞定了这个系统，挣到了人生第一个5000块钱。

这是夏华夏挣的人生第一笔钱，虽然现在看起并不多，但是当时在校普通学生的生活费，基本上也就是每月一两百块钱，所以绝对是一笔“巨款”。当然，清华计算机系人才济济，很多同学开始在外面写书，做各种软件，收入几千也并不特别稀奇。但这是夏华夏第一次通过计算机技术挣到了钱，意义非同一般。

做自己喜欢做的事情，进步会非常快

1993年到2000年，互联网开始在中国风起云涌，后来成为巨头的BAT都诞生在那个时代。夏华夏也触网了。他跟同年级的几个同学一起做了一个“酒井BBS”，名字源自他们住的9#男生楼的谐音。计算机系的女生住7号楼，刚好离的比較近，所以他们搭了一根线连通两个楼，让系里的男生女生可以一起聊天。

9#BBS最初是秦浩瀚、卿芳慧、周霖等人带头做出来的。周霖就睡在夏华夏的上铺，他也是水木清华的BBS的站长（周霖后来曾任搜狐负责技术的高级副总裁，现在是搜狐旗下狐狸金服集团的联合创始人兼CTO）。

大学时期，夏华夏和他的同学们就接触了很多最新鲜的东西，一个全新的世界向他们敞开了怀抱。“我们开始去学习新的技术，学习网络相关的知识，这些都是兴趣驱动的，当时也没有就业压力。其实课堂上的知识带给的成长并是有限的，当你真正动手去做一些事情的时候，进步会非常快，成长也会非常大。”

夏华夏告诉我们，当时有个同学写了一款打升级的扑克游戏，后来很多年之后，发现很多扑克游戏还是基于当时他写的代码。“大学是比较单纯的，我们当时写了很多程序，大多都是因为有趣、好玩，功利性是很少的。”

不过当时，即使是顶尖的学府也没有开职业规划课，大家都是靠自己摸索。夏华夏刚读大学的时候，只想到能够保送研究生，并没有考虑太多。“我们班很多同学选择了出国，因为每个人接触的人不一样，看到的世界也不同。”夏华夏回忆说，即使是清华学习相对较差的同学，出路也都非常好，有个同学因好几门功课不及格而被推迟毕业，不得已去了一家小创业公司，这家小公司的创始人是一个海归，名字叫张朝阳，后来这位同学还成了张朝阳的重要副手。

每个人都有自己的路，关键是选择

“其实，每个人都有自己的路，出国是一种选择，保研是一种选择，工作也是一种选择，关键是看你怎么走。”夏华夏说，当时创业的人几乎没有，因为大家对互联网创业几乎是没有任何概念的，清华的创业协会应该是1997年左右才开始做（王兴是其中的积极分子），后续很多留学生回国创业，那已经是很多年之后了。

在大学时代，对夏华夏影响最大的一个人是研究生期间的导师郑纬民老师。夏华夏大三和大四时期就跟恩师在高性能计算所做项目，而且郑老师对最新出现的计算机技术都跟的很紧，从并行计算，到后来的分布式计算、云计算、大数据、异构计算等。夏华夏一直在学习和实践，这也为后来读博和工作打下特别好的基础。“如果不出国的话，我还准备跟郑老师读博士，后面因为一些原因决定退学出国留学，郑老师也给予很大的支持，还帮忙写了推荐信。”所以夏华夏一直到今天，都特别感谢郑老师的栽培之恩。

6年异国求学路，名师出高徒

本以为读完研究生读完博士，然后找一份稳定的工作，然后就跟大多数人一样平静的度过此生。但命运，变幻无常，夏华夏遇到了生命中最重要的一个人，他又再次踏上了未知的旅程。清华的计算机专业是五年制，夏华夏大五的下学期就认识了自己的女朋友，现在已经成了他太太。

爱情，总是在不经意间，悄然而至。夏太太是同年级的同学，本科毕业后在清华继续读两年制硕士。夏华夏当时正在读博士，因为太太想出国发展，思虑良久后他也决定退学，跟太太一起远赴大洋彼岸继续深造。“其实，从来没有考虑过会遇到自己一生的爱人，也没有想到会出国。因为入学最初的想法就是读个研究生或者博士，出国其实是一个偶然，完全不在人生的规划之中。”

加州大学圣迭戈分校（University of California, San Diego，简称UCSD）位于南加州拉荷亚社区，那里环境优美，气候宜人，且坐拥全美最顶级海滩，被称之为美国“最性感”的理工科学习院校。2000年，夏华夏开启了异国他乡的求学之路，并度过了人生中重要的6年。“我在美国读了很长时间的书，但玩得也很开心，加州可能是全美最适合居住的一个地方，一年四季都不冷不热。”

夏华夏的博士导师是知名的华裔计算机科学家Andrew A.Chien（中文名钱安达，ACM、IEEE和AAAS会士，现为芝加哥大学教授，著名技术刊物《Communications of ACM》的主编）。在1990到1998年期间，钱安达老师在伊利诺伊大学（UIUC）任教授。1998年，钱老师转到UCSD当教授。所以夏华夏就跟随钱老师在UCSD继续做计算机相关的研究工作。钱老师研究的领域也比较广泛，后来研究的方向包括数据中心与超大规模系统架构与编程、弹性，数据密集计算工具，嵌入式与移动计算等。2017年钱老师还受邀来美团TopTalk讲座讲过课。

追寻自己的兴趣，实践出真知

在读博期间，夏华夏依然追寻自己的乐趣，做一些好玩的事情。“实验室有很多机器，我们可以在上面做任何东西，每个人都可以在机器上建网站，建个人主页。”当时，夏华夏和同学们搭建了当地的一个BBS，到目前这个网站还在，后来这个网站被迁移到云端，夏华夏和朋友还会发布一些住房的信息，然后进行一些社交活动。

2000年，MP3开始流行。夏华夏从国内下载了4万多首MP3歌曲。回到美国后，就跟同学一起建立了一个音乐库，对收集歌曲进行打分，开始没有对外开放，但是因为工作量比较大，然后就邀请更多的人参与到这个项目中来共同完成，很快就将想法落地了。“其实，当你有很好的想法时，都是可以去实践的，即使没有收到太多的关注，但你自己会有很大的收获。”

在夏华夏看来，整个读书阶段就是不断的去折腾，参与了很多计算机相关的项目，这才让他积累了很多编程的经验。即使读博阶段做了很多研究型的工作，但是其中参与的一个项目也写了几万行的代码。在夏华夏看来，如果没有去做很多这种小型的、有趣的编程实践项目，很多的知识自己也不会知道，也不会对计算机有那么全面的了解。仅仅靠课程内掌握的技术，可能也很难去面试成功一家公司，而且还是世界上最好的互联网公司。

经过7轮面试，成功加入谷歌

2000年，互联网泡沫破灭。整个IT行业进入低谷，一直到2004年才开始复苏，那一年，谷歌在纳斯达克上市。2006年，夏华夏读完博士，同年10月1日正式加入谷歌，当时最好的互联网公司。谷歌纯技术面试有7轮，前两轮都是通过电话进行技术面试，电话面试通过后，再去谷歌总部面试。总部的面试从早晨一直持续到下午，候选人坐在同一个会议室里；每过45分钟就有一个新的面试官进到会议室来，用各种算法、系统、编程、数学的难题来“刁难”候选人。

“无论是电话面试，还是总部面试，都会强调算法与编程，没有太多网上传言的那种类似脑筋急转弯那类的题目，更多的是算法编程、算法能力、分析能力、编程能力。很多人认为谷歌的面试不太公平，有的工作七八年了，还问这么简单的算法和系统问题，但真实情况就是这样，谷歌的标准是希望大家进入谷歌以后，能够写出更高质量的代码。”因为夏华夏大学和研究生期间的实践经验非常多，所以加入谷歌相对比较顺利。

在谷歌，想要“改变世界”是一件很容易的事情，因为谷歌服务全球市场，拥有的计算机的数量也是非常庞大的，用户数量也非常庞大，只要随便找一个项目去优化一下，那么就会产生特别巨大的效果，不仅可以提升用户体验，甚至能够影响公司的运营成本。

很多书中或者网上也提到，在谷歌工作往往是工程师自己找事情去做，很多团队中甚至没有产品经理。相比之下，国内很多大型的互联网公司对产品经理的要求其实很高，而谷歌属于工程师文化，偏技术主导。包括后面谷歌做PaaS平台，组建了一个两三百人的团队，但是产品人员只有个位数，谷歌的工程师都是凭借很强的自我驱动能力来把事情做完。

夏华夏在谷歌的经历主要分成两个阶段。第一个阶段，2006年谷歌进入中国，面临很多特殊的问题，包括访问受限，数据的存储问题，中国要求有些数据必须放在中国，而谷歌对自己的敏感数据不愿意放在中国，所以需要有一个专门的团队去解决这些问题，然后再做一些新的技术解决方案。夏华夏加入了一个名为China SRE的项目组，去解决这些富有挑战性的工作。

第二个阶段，夏华夏参与了Google+的研发。由于看到了Facebook发展的非常迅速，当时谷歌也希望在社交领域加大投入，所以谷歌在2009年启动了一个代号“Emerald Sea”（“翡翠海”）的项目，目标是研发功能强大的Google+社交平台。当时谷歌把“翡翠海”视为战略级项目，调集了几百人的精英团队去做这个项目，时任CEO的Larry Page也把办公室搬到了项目所在的楼。

注重开放能力，勇于改变

从2006年到2011年，夏华夏在谷歌度过了非常美好的一段时光。谷歌的架构设计非常有特色，后来搭建美团整个架构体系时，夏华夏也借鉴了谷歌架构的很多设计理念，包括容灾系统的设计，到现在也没有过时。“在谷歌6年，让我感触最深的就是谷歌非常、非常注重开放，所以在美团我也非常鼓励工程师开放、多分享技术。”夏华夏说，谷歌几乎所有的代码都是开放的，除了很极少数的核心代码，比如搜索算法、排序算法等等。如果谷歌的同学觉得别人的代码里哪个地方设计的不够好，可以直接上去改。

夏华夏说：“谷歌的理念对我的影响比较大，在后来的工作中，特别是到美团后，我一直都试图往这个方向靠近，包括现在新组建的团队，我们希望让每个工程师都有对代码的控制和访问权，对代码质量的把控权，包括运维和安全的责任，这种理念能够帮助提升整个技术团队的主动性。”还有很重要的一点，谷歌的工程师都具备体系化的思维方式。比如谷歌的工程师，从产品到架构设计，再到最后的上线测试，工程师都是从头跟到尾，所有的代码包括后续优化的代码都统一放到一个代码仓库中，所有的文档也放在Git中，所有的培训资料，工程师可以随时进行修改和优化，这些对夏华夏都产生了影响。

谷歌投入了很多时间和精力去构建公司的课程体系，包括一套名为EngEDU的线上学习系统，这套体系中包括很多编程语言、内部工具、内部研发流程的学习，每个员工都可以从最初级的语言开始学习，还可以学习谷歌公司的一些系统的使用、公司的一些工具和代码库、以及网络系统的配置方法等等，每节课还会有编程的练习。夏华夏说，“未来美团也会向这个方向努力，现在美团技术学院推出了自己的学习平台，还制作了很多技术课程，向公司内部的同学开放，我们也希望能够沉淀成一些体系化的课程，帮助更多的工程师成长。”

其实，从大学毕业到进入谷歌，夏华夏一直没有刻意去选择自己要走那条路，感觉一切都是顺其自然的。对他来说，这些可能都算不上“很重大的决定”。如果说人生很重要的一个抉择，那么放弃谷歌的工作，回国发展，肯定算是夏华夏最重要的决定之一。

因意识形态对立问题，决定回国

回国，是一个很长的话题。2000年到美国后，夏华夏无论是生活还是学习，都处在一个很舒适的环境中。2008年奥运会在北京举办，很多海外的华人都觉得很自豪，夏华夏也觉得为祖国骄傲。后来，华夏夫妇对美国所谓的自由和民主信念开始破裂，其中最重要的一件事就是奥运火炬当时经过旧金山，当时遭到部分反华势力组织的阻挠和捣乱，甚至于美国的很多媒体都在扭曲报道，包括美国最有名的媒体CNN也歧视中国，进行一些不实的报道。他们开始认识到，中美在意识形态层面是对立的，虽然美国有很多先天优势，在这里发展能有更好的全球视野，但是并不是特别理想的国度，所以夏华夏和太太开始把回国发展的提上日程。

还有一个很重要的因素，让夏华夏也深刻感受到国内互联网行业的蓬勃发展，国内很多互联网公司正在高速追赶美国，这点让夏华夏非常激动。夏华夏在谷歌的一位前上司加入百度做技术VP后，在2011年6月份找到夏华夏，希望他能够回国帮助百度做技术架构方面的工作。因为百度当时也在对标谷歌，而且当时发展非常不错，在工作内容方面比较匹配，所以夏华夏开始人生的非常重要的一个决定——回国。

遭遇挫折，用更高的维度去思考问题

2011年底，经过短暂的准备后，夏华夏回到了北京加入百度。夏华夏在百度一年多的时间里，担任运维部的总架构师，他当时的主要工作是把运维部的工作做了梳理，同时也参与了技术架构、基础软件等工作。“我觉得在百度，其实是一个比较好的缓冲或者落地。”夏华夏说。其实谷歌的工程师文化跟百度还是有很多不同的，当时在百度做技术架构工作是有一些“虚”的，也曾经试图跟基础架构部的同学，一起推动国外比较好的技术理念，但是由于种种原因，落地非常困难。

在夏华夏看来，很多公司在做基础架构层面的工作时，往往没有考虑到业务方的需求，只是想做一套很好的系统，很好的架构，让业务方去替换，这种模式存在很大的问题，而且这也是一份非常有挑战，非常非常有风险的工作，成功的可能性很小。百度的工作经历，提升了夏华夏对技术工作整体的认知，也让他能够从一个更高的维度去思考基础架构层面的工作，这也为后续在美团的发展，埋下了很好的伏笔。

太太牵线，结识美团技术团队

很多时候，选择比努力更重要。那么如何才能做出最好的、最适合自己的选择呢？需要勇气，魄力，更重要的是前瞻性的眼光，这些因素都在夏华夏身上得到很自然的体现。

夏华夏能够加入美团，很重要的一个原因也是因为自己的妻子。她此前是在eBay做电商方面的工作，所以回国后也想在这个行业发展，陆续跟国内电商类的公司京东、携程、拉手、窝窝、美团的技术负责人都聊过，其中大部分感觉都算不上很好的技术人员，如果加入他们负责的技术团队，实在是有点担心。唯一留下好印象的就是美团联合创始人穆荣均，在面试中让人感觉很靠谱。最后她决定选择美团。

夏太太还讲到一个细节，其实在2012年回国之前，她就给美团发了简历，很快得到回复，能不能回国面试。在告知了对方回国日期之后，她自己很快就忘记了，可是没想到在回国前两天，美团的HR同学很准时地再次与她联系。这件事让夏太太觉得，美团整个团队做事比较靠谱，令人印象深刻。虽然妻子一直在美团工作，但是夏华夏对美团并没有太多的了解。

2013年3月份，穆荣均通过夏太太向夏华夏抛出了“橄榄枝”。其实，最初就是以朋友的身份见面聊天，吃过几次饭，先建立了友谊，开始穆荣均并没有很快表明“想挖人”的意愿。

因为夏华夏住的小区离美团很近，所以他偶尔去美团接夏太太，当时夏太太在数据组工作，就发现美团的监控工具做的很好，虽然底层也是开源系统，但美团基于它做的功能很简介、很直观、也很好用。后来陆续又接触到很多美团的技术同学，发现整个技术团队很务实，技术氛围很好，整个团队的工作态度非常认真。所以每次面对穆荣均的约饭，夏华夏都欣然前往。

与王兴畅谈云计算，终被美团所打动

后来，穆荣均开始给夏华夏介绍了美团当时的情况，以及未来的发展规划，还有王兴的一个梦想。那还是在2009年年底的时候，王兴说：“其实很多做互联网创业的人，很少考虑怎么去帮助这些人，这些普罗大众，这么多的小商家。其实，他们都在努力改变自己的命运，我们美团就要帮助他们，帮助这些普普通通的老百姓。这也是属于我们的机会。”这句话，深深触动了夏华夏。

不久后，穆荣均将夏华夏介绍给王兴，他以为跟CEO聊天，应该会聊一聊公司的愿景，公司的发展目标之类，但是第一次见面，王兴跟夏华夏聊的主题却是云计算。“2013年，百度还没有正式开始做云计算，国内做云计算的也很少，所以王兴大谈云计算这个话题的时候，让我很诧异。”夏华夏问王兴为什么对云计算感兴趣，王兴说，其实从2012年开始，美团就在做云计算相关的技术储备了。

王兴对云计算技术理解很深刻，他做了很多的阅读，进行过很多的思考。当时给夏华夏留下非常深刻的印象。之后的更多接触，让他觉得美团做的事情，并不是想大多数人想的那样技术很Low，从开始就是一项非常有挑战性的技术工作，而且着眼长远，整个美团的技术团队也是很认真地想通过技术手段来解决吃、喝、玩、乐等一站式生活服务问题。

夏华夏再次选择了一条更难走的路。2013年清明节假期，他终于答应穆荣均，正式宣布加入美团。从国内顶级的互联网巨头，降薪跳到一家前途还不是很明朗的创业公司，充分显示了他的判断力和勇气。

选择需要魄力，更需要信仰

“那时候美团比较小，我太太倒是很担心，因为两个人都在同一家公司，万一公司干砸了怎么办。我自己倒是没有犹豫，因为我是那种不怎么考虑太多后果的人，这件事很有意义，反正想做就做了。而且除了收入减少了，并没有什么特别严重的影响，两个人的收入也够用的。”夏华夏很坦然。

命运，也总是会垂青那些努力的人，坚持的人，也会青睐那些不断有着人生追求的人。有时候，命运会在我们前行的道路上，设下重重障碍，很少有人愿意去打破那些障碍，去看看未知的世界。每一次抉择，夏华夏都是追寻自己的内心，并没有考虑太多财富、名望这些东西，所以他可以比绝大多数人，走的更远。

不择细土，方能成其高

2013年，美团技术团队已经初具规模，整个团队也很务实，但是跟谷歌这些互联网巨头相比，还是有很大的差距，夏华夏也希望能够将谷歌的技术理念带到美团。夏华夏的第一项任务，就是组建技术工程部（包括技术部和移动技术部）。当时美团投入很大的精力在做移动端，夏华夏用两个月左右的时间轮岗，摸清了从前端到后台几乎所有部门的大致情况，然后开始组建系统优化项目组。

美团很多同学对夏华夏的印象都是很务实的一个人，完全没有领导的架子。而且夏华夏也会深度参与很多技术项目，逐个解决，跟大家努力把项目做好，无论是技术层面，还是管理层面，还有对技术梯队的培训，夏华夏都会亲力亲为。

最开始到美团，夏华夏主要跟移动端的同学在一起，讨论如何优化网站性能。当时王兴和穆荣均也经常找夏华夏讨论网站的性能问题，因为他们看到了亚马逊的研究，0.1秒的网页延迟，会直接导致客户活跃度下降1%，当时美团首页加载需要4到5秒，如果提升几秒的话，对公司的价值可以想象有多大。所以夏华夏的当务之急就是提升美团的访问速度。后来在“千团大战”中，美团能够脱颖而出，一方面因为王兴

为首的创始团队从最开始就很注重技术，知道什么是关键因素，另一方面就是美团有很多像夏华夏这样的技术人才，他们对技术的追求非常执着而且很认真，战略和执行的统一，才让美团走的更高、更远。

其实，做基础架构方面的工作，事无巨细，所有跟技术相关的东西，都需要参与。夏华夏发现移动端对推荐算法的要求非常高，因为当时屏幕还很小，在有限的空间展示就需要非常好的算法，而团购产品推荐算法对UPS（用户画像）的要求非常高，所以又开始参与了用户画像的算法项目。

做完UPS后，夏华夏又陆续参与了很多业务项目的开发，包括技术存储、负载均衡、中间件系统等等。夏华夏又接手了运维团队，因为系统的稳定性直接关乎用户体验，这项重任又落在了他的肩上。

除了纯技术项目之外，夏华夏和穆荣均一样，都非常重视工程师文化和技术品牌的建设，他们知道在这方面投入，让更多技术同学有更好的成长，收益很高。夏华夏曾经是美团技术委员会的主席，也是美团技术学院最早的负责人。技术学院的很多项目包括培训、Hackathon、很有特色的图书馆和技术博客等等，最早都是他和从前端工程师转运营的弥新锋一起做起来的。

他本人还是公司很受欢迎的金牌讲师，不仅讲技术、架构，还讲授了非常多软技能方面的课程。

从技术走向管理，从接受C的评价开始

人生，不会是一帆风顺，难免也会遇到很多坎坎坷坷，很多人只会抱怨生活的苦难，而没有看到苦难背后，生活给予的礼物。

当然在美团，夏华夏也承受了一些“委屈”。在2014年年底，他给技术团队同学绩效考评，给某个同学打了C评价，但是这个同学表示不服。后来申诉到CTO那里，穆荣均找夏华夏沟通，问他为什么没有说服这个同学。在夏华夏看来，工作产出相对较差，所以应该给C的评价无可厚非。但是，穆荣均告诉夏华夏，给下面的同学打绩效、说服就是管理工作，如果没有很好地说服同学，就说明管理工作没有到位。

更令华夏没想到的是，“当时穆荣均说，要不这样吧，既然他不接受，要不你接受个C吧？当时我说，那行吧。其实我当时还是觉得比较委屈的。后来想起来，我觉得穆荣均说的非常有道理。”

那一次，夏华夏人生中第一次接受C级的评价。夏华夏说，他跟穆荣均相处的时间里，也学到了很多管理层面的东西。因为自己一直太专注于技术，所以在这方面存在很多问题，后来他也跟穆荣均学了很多管理上的理念和方法。也是因为技术和管理层面的沉淀和积累，夏华夏才能扛起更加重大项目的挑战。

在美团，最难忘的那些经历

2015年，O2O再起硝烟，外卖大战正酣，夏华夏临危受命，第一次开始接手业务直接相关的工作。此前夏华夏一直都是做基础架构方面的工作，包括谷歌和百度，以及美团的前期。当时美团外卖部门业务压力非常大，系统已经快撑不住了，那时候已经做到180万单了。每天中午，整个技术团队都非常紧张，周末大家也加班，还开辟了美团外卖的“作战室”，这种情况一直持续了两个月左右，然后他又带领大家把以前技术的“坑”填补上。通过大家共同努力，夏华夏带领技术团队将外卖系统的高可用从两个9提升到三个9，并解决了很多关键路径上服务的稳定性问题。终于在一个周五的晚上，夏华夏告诉大家，周末不用加班了，大家都表示非常开心。那个时刻，对夏华夏来说特别有感触，也很欣慰。

从谷歌到百度再到美团，夏华夏一直抱有一种理念，他觉得做技术最终是为了让团队越做越轻松。在百度的时候，夏华夏在负责运维就发现，如果公司的质量体系或者运维体系做的不够好，技术团队就会非常辛苦。所以到美团以后，他希望带领技术团队，不断追求卓越，尽量让大家不加班，轻轻松松把事情搞定。后来看到大家都能开开心心回家过周末，他也觉得非常具有成就感。

2015年，从基础架构到业务部门，夏华夏再次突破了技术的挑战，而且技术同学的笑脸，让他记忆非常深刻。这一年，美团技术团队抗过千万级流量并发的考验。这一年，夏华夏在技术层面也实现了自我的突破。

这两年，夏华夏又开始迎接新的挑战，负责公司最前沿的无人配送项目。

无人驾驶技术的落地是世界难题，但美团配送末端物流“小轻慢物”的特点，却提供了一个很好的场景。夏华夏从0开始组建了一支团队，涉及很多自己之前很少接触过的技术：机械结构、电子工程、嵌入式开发等等，并很快实现了产品并开始落地实践。现在，无人配送开放平台已经完成了在朝阳大悦城的B端测试运营，以及深圳联想大厦的C端试运营，并在上海松江大学城实现了从B端到C端的完整闭环运营。在松江大学城内，由无人车配送的美团外卖订单已经超过1000单 / 天，印证了美团无人配送开放平台进行片区规模化运营的可行性。

今年7月，美团还发布了无人配送开放平台，希望集合政府、高校、企业三方力量一起实现这个伟大梦想，已经吸引包括清华大学、加州伯克利大学、北京智能车联产业创新中心、华夏幸福、Segway等近20家国内国外合作伙伴加入。

加入美团后，夏华夏一直在不断打破自己的舒适区，一直对技术有着非常执着的追求。高速成长的美团，也刚好给了夏华夏足够大的舞台去施展。今天，他依然奔跑在技术的最前沿。因为在那里，是距离战场最近的地方，他可以继续探索技术的新边界……

对话夏华夏：程序员的人生抉择

Q：在你自己的成长经历中，你觉得有哪几个重要的里程碑？

夏华夏：我觉得我的成长，很大程度上属于“阴差阳错”，自己的运气也不错。第一个里程碑，应该算是“蒙”到了清华计算机系。不过，后面的学习就是按部就班，每一步都走的比较稳，包括读博也是听系里老师的安排。如果不是我太太出现在我的生命中，我应该会留在国内发展，整个人生的发展路径也会不一样，当然现在说不清哪个好哪个差，因为时间无法倒流。

第二个里程碑就是出国读书，2000年那个时候，在计算机领域国内外差距很大，所以去UCSD学习对我专业能力的提升非常重要。我还记得刚到国外时，由于英文的问题，课程难度非常大，经常赶作业到凌晨12点以后了，最后一班校车也停了，只能走路回家。如果只是靠在清华学习的那点知识，进入谷歌显然没有任何机会，在国外都会要求有很强的编程能力，另外就是对系统的理解，在UCSD读博的过程中，还有自己搭建过一些网站，打下了比较好的基础，所以后面进入谷歌和百度，都非常顺利。

第三个很重要的里程碑就是加入美团，在加入美团之前，我很多时候是独立工程师或者架构师的角色，到美团以后才开始带大的团队，包括跟穆荣均也学习到很多管理层面的东西。后来跟老王（美团联合创始

人、高级副总裁王慧文）工作，他是另外一种风格，实操性比较强、对业务和产品的思考角度非常新颖。所以在美团最大的收获是在从一个纯技术人成长为能力更综合的人。

值得一提的还有，2014年加入了美团管理学院讲师团队，因为在演讲的时候，跟真实情况也是不一样的，需要构建理论体系。其实最好的成长，就是分享，无论是做技术研究还是团队管理，这句话都非常有道理。

美团从创立之初，技术管理就比较规范，所以在这里成长也非常快。开始的时候，几乎每周换一个团队，跟这个团队的同学一起工作、一起学习，深入了解这个团队。我觉得技术管理者应该多接触一线的同学，可以让自己更快的融入团队，后续很多工作的开展，也会起到事半功倍的作用。

Q：你认为优秀的架构师或者技术管理者，都应该具备哪些特质？

夏华夏：首先最主要的就是技术的深度，其实架构师也属于技术梯队，我们在面试的时候，要求技术同学对其所在的领域要非常熟悉，包括使用哪种技术方案解决问题，为什么要使用这种技术方案，在技术选择时要考虑哪些重要的因素等等，对技术细节的把控能力要足够深刻。

其次，就是技术人员要具备足够宽的技术视野，也就是广度，要对自己领域周边的技术发展要有所了解，同时也要了解这种新技术在其他公司的应用情况。我希望能够加入美团点评的技术同学都能够很好的深度和相对的广度。除此之外，也希望他能够具备比较强的学习能力和自我驱动力。因为技术发展变化非常快，我们希望新加入的同学拥有足够的热情去不断的学习。

在技术管理能力层面，一方面是管理项目的能力，如何才能更好的推动一个项目；另一方面就是如何更好的管理好人才，使用好人才，培养好人才。我觉得一个好的技术管理者必须具备一个很重要的特质，心态要足够开放，这样他就愿意去学习新技术，能对事情研究的比较深入。

Q：对于美团的技术同学，您有什么建议？

夏华夏：第一个方面是**保持不断学习的心态**，这也是美团工程师文化特别提倡的，这是我们希望所有工程师具有的特质和建议，包括对技术领域深度和广度的学习，不断延伸个人的能力。

第二个方面是**全栈思维**，从心态方面，希望大家能够站在团队的视角来看问题，我们鼓励开放的工作氛围，美团点评内部的很多代码也在逐渐开放。还有就是从技术能力层面，我建议大家的技术能力能够全面一些，现在美团有6个技术通道，包括前端、后台、算法、运维、测试、系统，每个技术同学应该逐渐从全栈的角度去考虑自己的发展，还有对产品和业务的思考。现在美团点评基础架构、研发服务框架、安全认证体系、容灾体系等等很多项目都还在持续的建设中，我们鼓励更多的工程师主动参与进来。

第三个方面，就是**前瞻思维**，近几年涌现出很多的热点技术，我们应该积极的思考热点技术如何能够更好的跟我们的业务结合起来。比如利用人工智能相关的技术来提升用户体验，像语音识别技术可以帮我们做智能客服，我们也可以在算法层面做很多优化，来提升推荐和搜索引擎的准确度和效率等等，最近技术团队也在探索如何利用人工智能，实现自动化、智能化运维。我们鼓励大家利用新技术，从而来推动研发团队的成长。

最后还有一点，对技术管理者来说，美团有一个特别好的地方。就是没有跟很大大公司一样走两条发展路线，像百度，从T5、T6开始就分成两条线，团队中一个管理职责的人，一个做技术职责的人，管理者往往对技术的接触比较少，其实很容易产生很多问题。在美团点评这个大家庭里面，我们要求管理和技术齐头并进，提倡更均衡的发展，这种模式更适合培养更优秀的技术人才和领导者。

Q：你有什么特别佩服的人吗？

夏华夏：Google的Jeff Dean。在Google，大家都把他奉为”神“一样的人物，我也很佩服他。谷歌的技术架构之所以这么好，其实跟Jeff有很大的关系。Jeff亲手打造的系统包括Google File System、MapReduce、BigTable以及Spanner等等，这些是大规模分布式系统的经典架构，称得上Google和现代互联网存在的“基石”。

Jeff一直没有脱离编程的前线。在2011年的Google+项目中，当时高并发场景下Feed流读写性能跟不上，而读写后台是基于BigTable，所以Jeff就过来解决问题。他当时已经是Fellow级别（相当于副总裁），自己一个人带个ThinkPad过来，找个工位，编了两天程序，就把BigTable架构优化了，顺利支撑了Google+上线后的高并发访问。

Jeff虽然职位已经很高了，但是他还在不断尝试新的领域。他现在是Senior Fellow，Google职级最高的工程师，相当于管理线的SVP。几年前他把注意力从基础架构转向了人工智能，现在是Google Brain的负责人，TensorFlow就是他团队的作品。就在不久前，他还发表了新论文，提出了使用机器学习索引来替代B-Trees，能够提速3倍，再次点燃了整个技术圈，当时朋友圈很多人在转发这个消息。

不断尝试新的领域，不断突破自己的边界。我觉得这是非常值得我们所有技术同学学习的地方。

Q：如果让你给技术同学推荐一些书，你会推荐哪些？

夏华夏：我会推荐尤瓦尔·赫拉里的《人类简史》和《未来简史》，这两本不是计算机技术相关的书籍。但是作者从一个非常宏大的视角阐述了我们整个人类社会的发展，也强调了人与自然的关系，包括现阶段人类如何受到技术的影响，以及人生的意义等很多话题。我觉得这两本书可以帮助我们技术同学更好的理解、认识这个世界，认识“人类”这个种群，或者说这种生物，这两本书对我影响蛮大的，我也推荐给大家，希望大家也能从中有所感悟和收获。

招聘信息

美团无人配送部于2016年组建，自研无人配送产品，开放自身业务场景，致力用先进的技术，对配送侧进行改革，增加运力的供给。团队目前已经自主研发有两款适应不同场景的无人车产品和一款无人机产品，发布美团无人配送开放平台。目前美团无人配送已经完成在雄安、北京、深圳、上海等地的落地试运营，参与制定发布《服务型电动自动行驶轮式车技术要求》，在技术场景和法规等多方面推动产业发展，最终达到用无人配送让服务触达世界每个角落的目标。美团无人配送团队诚招各路英才，简历请投至：walle.hr@meituan.com

人物志 | 美团女技术总监任登君：不要给自己的人生设限

作者：王鹏

在我们美团技术团队超过6000名工程师中，有众多的女同学，她们是支撑中国领先的生活服务电子商务平台不可或缺的力量。3月8日女神节，我们专访了她们的代表——美团广告平台技术负责人任登君。登君也是我们团队里目前职位最高的女性技术Leader，平时喜欢玩游戏，喜欢跑步，外号小A也是当年游戏的ID。

希望任登君的成长故事对大家有所启发。同时，在这个特别的日子里，真诚祝福所有的女工程师们，永远年轻，永远美丽。



任登君

本期人物 | 任登君

2017年加入美团，担任广告平台技术部负责人，是目前美团技术团队中职位最高的女性。她的团队主要负责公司广告变现的技术平台，包括投放、算法、计费、广告主、销售等相关的系统。这支技术团队支撑了美团广告事业部每天收入从百万量级到千万量级的增长。

任登君硕士毕业于东北大学，专业方向是NLP（自然语言处理领域）。曾就职于东芝中国研发中心研究部，从事自然语言处理相关工作和研究。2010年加盟阿里巴巴广告技术部，历任广告主工具、直通车定向广告技术团队、搜索广告产品团队负责人。在淘宝广告部门工作期间，负责搜索广告整体市场和完成无线搜索投放策略的制定，带领团队完成直通车定向广告系统和算法的优化。随后出任悠易互通CTO，全面负责组建并带领高效的技术团队，专注于系统架构、基于大数据的核心算法以及显著提升客户的广告效果。

梦想当医生，误打误撞搞了计算机

Q1：你最初是怎么开始接触技术的？

任登君：其实，我一直到大学才算正式接触计算机，高中时虽然也接触过一些BASIC编程，但都比较简单。因为我爸爸是一个医生，我当时一门心思想读医学院，但是高考不是特别理想，所以落到第二志愿。读东北大学的计算机，并不是精挑细选的，算是误打误撞。在考研之前，也想过转到经管方向，但是本科期间对计算机产生了很浓厚的兴趣，刚好东北大学有一个自然语言处理实验室，研究生方向就定了下来。毕业后就直接去了东芝中国的研发中心，后来就一直做自然语言处理领域（NLP）方向的研究。

Q2：十多年前就搞机器学习、NLP啊，当时这类专业是不是还挺冷门，不好找工作？

任登君：是的，当时工作机会确实不太好，跟现在找工作相比，已经是天壤之别了。不过，NLP方向在我们学校实验室拥有非常长的历史了，虽然人数一直不多。这也是因为我们学校有一个非常厉害的老师姚天顺，我也是慕名而去，很感恩姚老师把我收下了。对当时20多岁的一个小姑娘来说，也不知道NLP是啥，也没想太多，更没考虑哪个方向比较热门、将来好找工作这回事，就是想跟很厉害的老师学习。

当时NLP非常冷门，在我毕业之前的那些届，基本上全部留在学校继续做研究，当个助教或者老师，极少数人转行去了传统软件公司。我还是属于比较幸运的，研究生毕业时，百度、东芝已经在寻找自然语言处理领域的人才了，那时国内这方面的人才储备可谓凤毛麟角，我们实验室每年也就保持10个人的毕业量，加上我们实验室本身就是国内比较顶尖的自然语言处理实验室。即使很冷门，但是我们工作方面还是不存在问题的。我是2005年毕业，如果早毕业三年的话，估计只能去传统软件公司了，因为好的研究性机构，国内并不多。

加入东芝也是巧合，我曾参加了一次学术会议，刚好发现东芝中国的研发中心研究部要招人。像东芝这种研究中心，已经好几年没有招人了，他们也是聊了很多大学毕业生才下的决定。当时东芝中国的研发中心自然语言处理部门负责人，就是现在百度副总裁、AI技术平台体系（AIG）总负责人王海峰。我在东芝待了五年，一直都跟他在一起工作，也学到不少东西。

不想学术偏离实际，离开研究所到企业实践

Q3：在东芝五年期间，有什么比较有意思的事情吗？

任登君：因为一直是做研究相关的方向，所以就是大量地写论文、做实验、研究前沿技术、参加各种学术会议以及写专利等等。其实，跟所有的研发中心一样。但是这期间，我们也有支持东芝本身的这些产品模块，我几乎参与了东芝所有的中文分词、日语分词项目，这些自然语言处理工具都在东芝很多产品线里面进行了应用。

我记得还参加了一次世界级的翻译比赛，当时我们花了很长时间去做数据集和算法优化，大概用了两个月的时间，最后16项比赛，我们团队拿了15项第一，只有一项是第二。这五年期间，其实做了很多事情，但是离真正的应用，还是有些远的。当时很多做自然语言处理领域的人才，基本上都在研究机构里面，说蛰伏也好，说沉淀也行，总之大家都熬了很多年。后来，我们部门的很多同事去了百度和微软，特别是近些年，人工智能爆发，我们这些人多年的积累，终于看到了一些成果，还是很欣慰的。

我记得NLP这个领域，应该是在2008年左右有一次爆发，现在已经是新的一波爆发了。当时主要是在语音理解、分词以及知识图谱层面。最近一两年是深度学习和算力的发展，解决了人工智能方向的一些问

题，它们都是非常典型的、由数据驱动的解决方案。特别是随着GPU的提升，深度学习实现了非常大的飞跃。但是自然语音处理不是这样，因为语音的处理存在太多歧义，不完全是数据问题，还包括很多的知识问题，所以现在大家集中来解决这个问题。我们也看到，市场上存在很多公司在做知识图谱、智能客服、智能问答这些市场。好啃的骨头很多年前都已经啃完了，现在大家都在死磕最难啃的骨头。当然在NLP领域，细分赛道也越来越多，这也代表新机会。

Q4：当时怎么去阿里巴巴的？阿里巴巴带给你哪些成长？

任登君：当时是老大自己加入百度，我们很多同事在两三年内也纷纷离开。当时我有很多选择，可以去百度，也可以去搜狗和雅虎研究院，但是无论去哪里，还都是做NLP方向。我当时想做一些更贴近业务的方向，前面我也讲了，感觉在研究院这些年，业务不接地气，偏离实际使用很远。所以就跟阿里巴巴的搜索和广告都聊了一下，后来决定加入广告部门。

坦白讲，那时我对互联网也不是很了解，对广告也不是很了解，但这个最接近业务，自己写了五年的论文，做了五年的实验，感觉跟社会很脱节，想到一家能够将一身所学落地的公司，所以最终选择了阿里巴巴的广告平台。

在阿里巴巴当时团队叫淘宝广告技术部，团队开放性和包容性非常好，大家也给我很多机会，我个人的成长也非常快。因为我之前也没有带过团队，也不知道自己适不适合带团队，一直到阿里之后，我才发现自己还蛮适合带团队的。我当时主要给广告主做工具，在这个过程中，发现了自己的管理潜质，后面也就带了定向广告团队。人，总是要不断拥抱自己的变化。

从技术到管理，最重要的是接受心理层面的挑战

Q5：从贡献者到管理者，这个阶段你觉得有什么可以分享？

任登君：我觉得从一线技术到承担技术管理职责，都要经过一个心理关，可能表现为个人技能和团队整体技能成长之间的矛盾。因为要带人，需要花时间去管理，可能还需要跟产品沟通，跟很多部门的人沟通。后面可能发现，你没有一线技术看Paper的时间长，没有一线的技术写代码时间久，可能会发慌，担心失去安身立命的本领。我当时也存在这个问题，自己非常的慌。大概经过了半年时间的调整，后来我发现，如果能够把团队带的很完美，也是一件非常有成就感、非常有意思的事情，就解决了这个问题。

但是一定要把团队的产出和自己的产出分开，确定哪些是团队整体的功劳，明确哪些是自己做的事。这样不断提升自己对项目的把控、对技术的把控、对分工的把控。很多技术转管理的同学，在心理上面临很大的挑战，因为管理是软性的东西，看不见、摸不着。如果工程师想转做技术管理，一定要考量哪个让自己更擅长也更喜欢，确定是自己成长更好，还是带领团队一起把事情做好更有成就感。如果是前者，那么未必要走技术管理这条线。我们是一个多元化的大家庭，工程师可以有不同的成长路径。

另一个印象比较深刻的，是跟后进的团队成员谈绩效，阿里采用271，现在应该是361制度，1的这部分同学是没有加薪也没有奖金的。作为经理，我比他们还紧张，在他们的绩效改进过程中，花了非常多的时间辅导，发现其实用处不大。新晋的技术管理同学，很容易花过多时间在低绩效同学辅导和改进上，其实应该把重点放在团队的头部。流失对团队来说，并不见得是一件坏事。

Q6：后来去悠易互通做CTO，从技术经理到CTO，这其中最大的变化和挑战有哪些？

任登君：悠易互通当时技术团队面临很大的挑战，急需一个CTO。我过去后帮助他们打造了一个新的技术团队，重新搭建了整个系统，做完了底层改造，几乎花了小半年的时间。

做CTO跟技术经理也有很大的不同，因为一旦到达这个阶段，不管遇到什么问题，什么挑战，都需要去解决它。而且招聘也是很难的一件事，作为CTO，你要想尽一切办法找到合适的人。我们都知道，一个顶级的技术人才的价值超过10个普通的工程师。这种人才需要花很多精力去招进来。

当然，CTO还要稳住团队，不仅仅要谈工资问题，还要让大家知道，在这里能获得成长。我们要相信，大多数人的自驱力都很好，对未来渴求都非常高。如果聊未来成长、谈人生规划，他们应该成为什么样的人，他们就会相信。我刚接CTO的时候，20多个技术几乎全部离职了，后来我招来60多个技术人才，在我走之前几乎没有离职的。后来，很多人去了360、爱奇艺、今日头条、微博等大公司，这点也让我很开心，挺有成就感的。

加入美团，想对广告业务做一次完整的探索

Q7：当时是如何结缘美团的？加入之前，有没有什么顾虑？

任登君：我做广告六七年了，然后自己想完整的做一次广告业务，刚好美团广告业务刚刚起步，有机会能完整的做一次探索。美团不仅是中国最大的O2O平台，也是全球最大的生活服务类平台，我非常看好这个平台，很喜欢整个公司的文化，我觉得这就是未来的新巨头，所以没有任何顾虑。而且我做了选择，也不会顾虑很多事情。

Q8：目前主要负责哪些方面的工作？平时如何安排自己的时间？

任登君：我们部门主要负责美团广告变现的技术系统，包括投放、算法、计费、广告主以及销售相关的系统。严格来说，所有的广告系统都在我们这里。我们团队现在覆盖了各个类型的广告，包括效果广告、品牌广告、搜索广告等等，基本上所有的广告类型，也都是我们团队做的技术支持，现在我们每天的广告收益，已经达到每日千万的量级。

对我来说，每天到公司第一件事就是看数据、看报表，特别是收入报表。每天也会开很多会议，会跟产品方向和业务方向的同学讨论最新的进展和问题，讨论更多可行的方向。我每天会想给自己留出两到三个小时的时间（可能最后也就能有一小时）看技术的新方向。应该是突然有一天，我发现自己非常熟悉的NLP方向的发展，都跟不上的时候，还是有点慌的。现在每天都会保留一部分时间学习，也会找相关的技术同学，或者我认识的业界的人士讨论一下。

目前，还会看谷歌和很多其他公司出品的一些论文，也会看一些实验室发的论文。比如我对知识图谱比较感兴趣，会找身边的朋友帮忙推荐一下，但是国内的论文看的比较少。我希望自己能够跟上最新的技术前沿方向，而且作为技术管理者，最主要也是判断方向的对与错，如果没有能力判断前进的方向，基本上也就失去了做一个技术管理者的资格。

Q9：目前来讲美团广告这块主要的挑战是什么？

任登君：我们的广告下面有LBS的限制，上面有服务容量的限制，要在两个限制之下找到比较合适的广告主认知方向和技术迭代方向，把广告做到一个合理的水平，这是一个动态平衡问题，是一个很大的挑战。

这方面与传统的纯线上的互联网广告不一样，没有人告诉我们该怎么做，必须自己探索。

对于有一定经验的广告技术人员来说，这种广告场景也是很有吸引力的。因为有挑战，有发挥的空间。甚至对于学术界，也是有意思的研究方向。

Q10：对整个美团现在技术团队整体感觉如何？有什么建议？

任登君：我觉得工程师技术氛围依然需要持续培养，其中最核心的是做到“追求卓越”这四个字。

在美团这样业务导向的公司，我们首先要尊重技术，相信技术的力量，但也要清醒地认识到，不能把业务押宝在技术的突破上。公司对技术团队还是很宽容的，大家有很大的空间。技术团队自身呢也要致力于把自己的事情做得更好，然后发挥更大的影响力，真正能够“追求卓越”。

公司目前制度、薪资都挺好的。技术管理方面，大家需要达成共识。对技术团队的管理者来说，只要你在一件一件小事上都奖赏、鼓励追求卓越的同学，大家都会看到，这种氛围就培养起来了。我们的博客、微信公众号技术影响力方面，我觉得都挺好，这些东西很难一蹴而就。

勇敢选择，并欣然承受所带来的代价

Q11：家人对自己的工作理解吗？如何处理工作和家庭的关系？

任登君：我的家庭很普通，但我感到很幸福，因为家人非常支持我的工作。我老公接管了女儿的学习和陪伴，让我能够比较安心的工作。我还要经常出差，他们也能理解。我女儿读一年级，她其实有非常多的课，除了周末能够陪女儿以外，我平常是没有时间陪她的，我也很歉疚。当然，我老公非常给力，因为他的工作相对可控，不像我一样经常出差，离家又比较近，所以他就花大量的时间陪伴女儿，好让我没有后顾之忧。

其实，对于我们这样的人来说，家庭不拖后腿是非常非常重要的点。女儿的成长肯定需要人陪伴，如果我老公也没有时间陪伴的话，我可能就没有办法这样高强度的工作了。我也花了很多时间，跟家人沟通，特别是我女儿，我会告诉她妈妈每天都在忙什么，不然的话，她会埋怨妈妈为什么不陪我。我会经常跟她讲，甚至在她们学校家长课堂上讲我的工作，也许她还不能完全理解，但是她知道妈妈很忙，知道妈妈不陪她的原因是什么。

Q12：如何平衡生活和工作？如何做选择？

任登君：我现在没法平衡，现阶段主要还是工作。除了工作外，就是陪我女儿上各种辅导班。我认为工作和生活是很难平衡的，因为一天时间就那么多。但是，我觉得做事是有选择的，关键是看你喜欢什么，看你这个阶段追求的是什么。而且我觉得选择什么都是可以的，只是选择带来的结果有好有坏。你能接受那些好的褒奖和收获，同时也要接受那些负面的声音和损失，任何事情都有两面性。

比如有的同学说，在这个阶段想多陪孩子，就把工作稍微时间调整一下，这是可以的，完全没问题。如果这个阶段，工作可以让我更开心，那就稍微牺牲一下生活带来的乐趣。现在是很好的社会，我们公司也能够包容多样性。有所得，也有所失，关键是自己能接受，别纠结。特别是对女孩子来说，女性在这些层面很容易被社会和自己内心的纠结所裹挟，我希望身边的女孩子，要更坚强一点，更坚定一点。

喜欢玩三国杀，希望能为人类的进步做出一点点贡献

Q13：在工作之余还有什么爱好吗？

任登君：我还蛮喜欢打三国杀的，现在很少玩了。很难讲出来理由，就是单纯的喜欢，在打的过程中，会有很大的愉悦感。我很难说，打三国杀对工作和生活会有什么帮助。跑步，倒是一件挺好的事情，但我跑的不多、也不少。说到理想的话，其实现阶段我没有特别伟大的梦想。当然，我也希望自己能够有机会去为人类的进步做出一点点贡献，哪怕是一个很微小很微小的进步。

Q14：最近在看哪些方面的书？给我们推荐一下。

任登君：我最近在看很多管理相关的书籍，因为刚好在管理上有一些心得。一本叫《领导梯队》，我觉得因为它的实操性非常好，非常值得一看。还有一本是《高效能人士的七个习惯》，这本书也比较贴近我们的实际，实操性也很强。最后一本就是《三体》，这种科幻的我也比较喜欢，它的脑洞开的非常大，可以让我们这些平凡人，想象的空间变得大得多。

女孩子要足够努力且专注，不要给自己的人生设限

Q15：针对做技术的女同学，你有什么成长经验可以分享？

任登君：我觉得，最重要的一点就是要突破心理界限。当我们决定要持续从事技术这个领域的那一刻，就应该忘记自己是一个女生，因为在这个过程中，如果你觉得自己是女生，就会给不由自主的给自己设置太多的限制。我觉得做技术，首先要突破性别的界限，自己要突破各种界限。我们现在团队有很多工程师就是女生，她们做的很好，我也为她们感到骄傲和自豪。

今天回过头看所有的专业选择，我也有一些感触。在这个时代，如果你只是一个普通家庭出生的小孩，想要更好的往前冲的话，计算机可能是最好的方向，应该没有之一。因为这个专业延伸足够宽广，可能性足够大，足够包容。互联网在高速发展，新经济时代也不再是完全固化的时代，无论你是男生还是女生，无论你做什么工作，普通人靠自己的努力，可能都不需要万里挑一的天赋，只要足够努力，足够专注，就一定会有非常大的机会，闯出一片属于自己的天地。

人物志 | 技术十年：美团第一位前端工程师潘魏增

作者：技术学院

导读

潘魏增，2006年毕业于南开大学电子系，2008年加入早期饭否团队。美团第一位前端工程师，现在是X项目组终端研发部的负责人。处女座，INTJ，喜欢Linux和Vim，崇尚开源，相信开源可以让世界变得更美好。



潘魏增

从饭否到美团，潘魏增用十年的技术生涯，诠释了“长期有耐心”这句话的含义。在他看来，长期有耐心，其实也是延迟满足感。对从事的行业来讲，我们要把眼光放得更长远一些，十年后才有回报的生意，往往都是大买卖。对个人来讲，不要把职位、职级这些虚的东西看得过重，关键看我们自己在其中承担什么角色，看我们自己的能力是否还有成长的空间。

“

本文系美团技术学院美美对潘魏增的采访内容，希望对大家有所启发。

从电子工程转到计算机

1. 为什么大学读的是电子系，但是毕业后却选择了互联网行业？

潘魏增：高中时，对物理比较感兴趣，学校有一个逸夫图书馆，里面有大量关于物理的课外读物，其中有一本杂志叫《无线电》，特别令我着迷。只需要少量元器件，就可以实现超远距离无线单向通信（收音机），简直太神奇了。于是，我就树立了自己的理想，以后要成为一名电子工程领域的科学家，所以选择了南开大学的电子系。

但是上大学之后突然发现，电子系的课程大部分都是以数学、物理相关的基础理论为主，动手创造的机会很少，特别枯燥。一次偶然的机会，想在电脑上搭建一个HTTP服务器，给各地的高中同学访问，因为不太懂，然后就去学校的BBS请教。当搭建成功的一刹那，我突然感觉到一种“触电”一样的兴奋感，相隔万里的人竟然可以看到我写的网页内容，太不可思议了。于是，我就开始经常泡BBS学习，到后来我就成为了能够回答别人问题的人，再后来，我就成为了南开BBS上WebDevelop版和Linux版的版主。

大学时，互联网逐渐从第一次泡沫中复苏。我在图书馆偶然看到一本讲互联网革命的书，书中那些早期设想的有关互联网的预言，都逐渐在——实现。我深受作者的鼓舞，不过我觉得互联网革命还尚在早期，未来还将获得更加蓬勃的发展，我应该在行业萌芽的时候，加入到这场浪潮当中去。怎么加入？我不可能去「赤手空拳」地创业，毕竟还要吃饭，于是去互联网公司工作就成了我最佳的实际选择。当时在我们系，去互联网公司工作，其实是一个非常另类的、不被人理解的选择，因为绝大部分同学都去做了跟电路或者芯片等本专业相关的工作。但是，我很喜欢。

今天回头看，我呆过的团队做了很多改变世界的事，整个互联网行业也大大地改变了世界的原貌，信息更透明、公开，社会更加平等。科技革命总是短期被高估，长期被忽视。十多年前，我还用笔给同学写信，在图书馆翻查资料，出门带纸质地图……站回当时看现在，几乎是难以想象的：我们可以通过微信实现实时互动，通过知乎、维基百科和搜索引擎查阅无穷无尽的信息，出门有高德地图，而且现在出门还几乎不用带现金。

2. 当时为什么会选择凤凰网？在凤凰网自己收获了哪些？

潘魏增：读大学时，比较喜欢看凤凰卫视的节目，而凤凰网是凤凰卫视集团旗下的子公司。凤凰的节目内容在当时来讲是媒体界的一股清流，这是一群有理想、有抱负的人做的媒体，我希望能和这些人一起工作。而且凤凰网也是最早设立前端岗位的公司之一，这和我的职业方向非常匹配。当我收到Offer的时候，就立即过来报道了，甚至薪水给多少，我都没问。

在凤凰网时，也学到了很多，主要是两点：

第一，完成学生到职业人的转变。加入公司后不久，老板就给机会负责整个网站的重构。迁移新闻系统工作量巨大，连续加班两个多月后，看到自己开发的作品每天有千千万万的网友在使用，非常有成就感。凤凰网是我职业的起点，一直心存感激。

第二，学到做事做人的一些方法。在大学里认为技术人总是特立独行，但在实际工作当中必须依赖团队协作。做事要认真，做人要简单。在凤凰两年，我做过多前端技术和用户体验相关的培训，同事们都乐于跟我请教，包括跟编辑、设计、销售等部门关系也特别好，很欣慰。

在饭否的那些日子

3. 因为什么原因加入的饭否？有什么故事？

潘魏增：从大学开始就喜欢自己在网上写点文章，分享自己的成长心得。2008年初，穆荣均（美团联合创始人）在网上看到我写推荐饭否的博客，然后写邮件邀请我出来喝咖啡。我们喝了几次咖啡，有一次还在五道口的办公室见了兴哥（美团创始人王兴）。没过多久，就收到了饭否的工作邀请，这也是我工作之后，第一次正式收到其他团队的邀请，有点“奇遇”的感觉。

我认真考虑了一段时间，然后也咨询了几个师哥和朋友，大家都认为饭否是非常靠谱的团队，于是，我就下决心过来了。

4. 在饭否几年中主要负责哪些工作？当时你对整个团队的感觉是什么？

潘魏增：在饭否时期，我在团队中的主要职责是负责前端开发，偶尔客串后端、运维、设计师、讲师以及客服。

刚加入这个团队的时候，第一印象是感觉这个团队特别酷。第一次开会，工程师直接用Firebug在投影仪上修改代码，直接浏览器上看效果，然后和大家一起现场讨论。平时的各种文档都是在Google Docs上一起协作完成，在同一份文档上，经常看到几个光标来回游走修改，现在感觉都还很科技、梦幻。

其次，团队每一个成员都很优秀。例如有工程师把PHP手册读到烂熟于心，了解这门语言的方方面面，版本升级也只是看看手册的Diff。

最重要的一点，感觉大家都很拼。我到入职之后才知道，我们一周要上6天班，工作时间是早上10点到晚上10点，经常会忙到凌晨，上班也不用打卡，全靠自觉和相互监督。

5. 加入饭否时，也正值青春年少，有什么梦想吗？当时有没有考虑过职业规划这些问题吗？

潘魏增：青春总是有很多冲动嘛，一心想要去改变世界。在饭否，我们的梦想就是让信息更公开，社会更平等，让信息流动更快。

职业规划方面，想得真的不太多。当时的想法，就是想往前端技术专家的方向努力一下，能对行业产生一些价值，个人能有一些影响力，就很知足了。

6. 从饭否到美团这些年，为什么没有考虑过换一个环境？是什么原因让自己坚持下来？

潘魏增：在饭否被关闭之后的那半年里，突然没那么忙了。坦白讲那段时间，确实有考虑过是否要换个环境，所以说，闲并不是什么好事，容易滋生各种想法（笑）。我有几个师哥和朋友知道我的事情之后，也给我介绍过几个工作，最后都被我婉拒了。在饭否呆过之后，对团队的要求突然变高了很多，开始知道什么是好的，什么是不好的。在美团的日子，工作更是超级忙，就压根没空去想过这个事情。

我很享受在公司工作的时光，喜欢现在的工作。其实，好的工作，不需要坚持。

在创业团队头两年，闷头写代码，技术每天都有进步（开心）。之后开始建团队、Coach团队，从个人贡献者到团队贡献者，公司给我机会从零开始学管理，从一张白纸开始慢慢明白一些道理（开心）。然后

团队规模越来越大，业务越来越复杂，管理能力迭代更新，我培养过很多个技术Leader（更开心）。

其实，开心的工作基本不需要谈“坚持”，它会给你持续的、正向的激励。不好的工作才需要咬牙坚持，而坚持错误，是一件机会成本很高的事情。

工程师眼中的美团

7. 在你眼中，我们美团是一家什么样的公司？有哪些特质让你印象深刻？

潘魏增：印象最深的，也是最喜欢的特质就是我们美团是一家学习型组织。公司创始人都非常善于去学习、思考和总结，并身体力行去分享、去鼓励大家这么做。

比如兴哥，我觉得他的学习面特别广。记得有一次聚会，兴哥聊天时说到各地的方言，还帮忙给大家各自的方言做归类，聊到兴致起来，还拿出一本厚厚的语言书，证明他说的都有理有据。后来还有一次，兴哥给大家送了一本关于地缘结构的书，他说不错，推荐大家看。我看完之后确实对世界格局有了新的认识，对我帮助很大。

兴哥能从学习中受益，我觉得很多人应该也是如此。学城（美团内部知识库）和互联网+大学，有大量内部学习的资料，有大牛的分享，有行业的判断，有方法论，我自己也从上面学到了不少真东西，大家可以利用一些这些资源。

8. 千团大战后，美团又做了电影票（猫眼，已经独立）、外卖业务，现在布局了酒旅、出行和大零售业务，作为其中的一员，最大的感触是什么？

潘魏增：最大的感触就是，每天都像在“打仗”。我们美团进入的领域一部分并不是最新的领域，另外一部分可能相对较新，同时有很多人也都看到了这个机会。表面上，美团像是和其他公司在争夺，其实不然，实际上大家都在竞赛，看谁能给用户提供更多的价值，提供更好的产品。听说有些公司，专门针对美团建立了“抗美办公室”，我觉得他们的思路很奇怪，还在用“零和游戏”的思维在做事情。

前些天，王慧文也分享了自己对这方面的看法，美团就是一种“尝试”的心态在做事，我们最终是希望给用户带来更大的价值。美团的价值观，第一条也是“以客户为中心”。因为竞争是全方位的，有时候看到其他公司写的“黑稿”，里面尽是一些没有逻辑的猜测和诬陷。这个时候，我们反而更像是打了“鸡血”一样，想把我们自己的产品做的更好。

当然，这么多年，也经历了美团的很多从零到一的业务，有时候会感觉会比较煎熬，有时候也会很亢奋，就像坐过山车，有时候嗨到顶峰，有时候也会感觉跌到谷底，但尖叫总是持续不断的。在美团的生活，真的很精彩。

十年技术生涯

9. 对绝大多数青年来说，大学毕业后，应该是人生最好的十年，这十年最大的收获是什么？您怎么理解王兴说的“长期有耐心”这句话？

潘魏增：十年的收获有很多，不过我感觉都是人生体验的东西，到这个时间点，到了这个岁数，每个人总会零零散散收获一些东西，包括物质层面的，还有精神层面的。对我而言，最大的收获可能是更了解自己，找到一些和自己更好相处的方式。

十年前，觉得自己什么都能做，什么都可以做得很好，现在大概明白自己的能力范围在哪里。

十年前，不明白自己想要什么，现在尽管依然不是那么清楚，但已经相对更清晰一些，还需要迭代。

长期有耐心，我的理解就是延迟满足感。

- 对从事的行业来讲，要把眼光放得更长远一些。做个不是那么恰当的比喻，每天计较得失的大部分都难成气候，十年后才有回报的生意，往往都是大买卖。
- 对个人而言，不要太早着急“变现”，拿得多，往往不如拿得稳，也不要吧职位、职级这些虚的东西看得过重。关键看自己在其中承担什么角色，看自己的能力是否还有成长的空间。

10. 这十年，有没有对自己影响特别大的人？

潘魏增：十年间，我有幸认识了很多非常优秀的人。有老板，有同事，也有人生路上的师哥、朋友等，对我影响都很大，感谢得到了他们的教诲和帮助，也从他们身上学到了不少东西。

坦白讲，我自己受穆荣均的影响比较深。我在公司里最早接触的人就是他，也是因为他，我被吸引加入了早期的饭否。他认为应该坚持做正确的事情，即使遇到阻力，也要不折不挠地推进。

比如，技术团队早期要提升并行开发的效率，版本管理工具要从SVN迁移到Git，这个事情虽然是我主导，但事实上是穆荣均在背后鼓励我这么做。

又比如刚开始的美团技术学院，有很多BootCamp培训、技术交流和组织上的杂事要做，同时会面临业务上的一些压力，穆荣均也是很支持我去展开这方面的工作。我相信对早期工程师文化的建设和团队组织的成长，都是有帮助的。

11. 这十年，对前端技术的认知，有哪些改变？

潘魏增：虽然十年来，前端技术层出不穷，但我理解的前端是「万变不离其宗」，它是为最终用户界面服务的，承接用户与远端数据的交互。前端的核心是数据的呈现，不管后端给什么数据，前端只负责忠实的展示。前端也可能会做服务端的开发，也可能会做平台化、工程化的工具，但它最根本的目标还是这个，不要偏离这个目标去做事情。

其次前端技术不能脱离业务而存在。前端工程师当中有部分同学很Geek，有的喜欢研究专深的技术，有的喜欢越界做点事情。这些本身也没错，但把更多精力去推动业务，获得成功是回报更高的事情。如果做技术缺乏业务视角，往往是很危险的。

爱健身，爱读书，爱思考

12. 平时有哪些兴趣爱好？这些兴趣爱好，对自己有什么影响？

潘魏增：我基本没有固定的兴趣爱好，不同阶段有不同的兴趣点。以前喜欢研究跑鞋、键盘、Hack电子书，现在都“退烧”了。喜欢过跑步、游泳、爬山，也都“收山”了。

目前还在坚持的、时间最长的是健身，每周定期活动活动身体。我要争取比同龄人体态更好一些，头发掉得更少一些（笑）。

受家人和朋友的影响，我开始喜欢旅游。我曾经信奉李敖先生所推崇的“读两万卷书，行零里路”，但去过一些地方以后，才发现“知道”和“看见”是两种不同层次的感受。比如在川西，路过四姑娘山时，山就那么横在我面前，让我突然意识到自己的渺小，冲击力特别大，和图片上看到的完全不一样。

还有，自从学会开车上路以后，也喜欢研究汽车知识和汽车文化。互联网是“比特在两点之间的移动”，汽车是“原子在两点间的移动”，这些概念都挺有意思的。

13. 读书多吗？哪几本对自己影响比较大？

潘魏增：比较惭愧，我读书很慢，所以读书并不是很多。豆瓣上标记想读的书永远比读过的书多很多。

我觉得读书有两个乐趣，一个是了解自己，另外一个探索世界。内向者的能量大部分来自读书和独处时的思考，就我来说，读书的比重可能更大一些，所以读书还会给我带来力量。这里有几个读书心得跟大家分享：

一、**读好书**。好书会改变一些东西，例如思维方式、生活理念甚至对世界的认知，而烂书只会浪费时间。找好书有三个渠道，豆瓣评分很高的；大牛文章总结的好书；朋友同事推荐。其中我最认朋友们的推荐，他们一般不推荐书，一旦推荐都极其靠谱，所以可以结交一些爱读书、会读书的朋友。

二、**做笔记**。写笔记可以概括重点、理清思路，对于理解书中的内容有很大帮助。即便不一定会再看一次笔记，但做笔记这个过程可以帮助加深记忆，书中道理也会通过记笔记的形式融入到潜意识当中。这跟抄佛经有点类似，不同的是，做笔记不要摘抄太多原文，那样没有意义，应该去总结、抽象，读书是为了获得思想，而非文字本身。

三、**回到初心**。要回到自己读书的初衷，或乐趣，或好奇，少一些目的性和功利心。读一本书，不要老想它能用来干什么，而是多思考它在表达什么。不要期望每本书读了都有用，马上就能用的书是菜谱。

最后给大家推荐几本我觉得不错的书。其中《禅与摩托车维修艺术》、《邓小平时代》和《素书》是大书，有些部分晦涩难懂，但值得反复咀嚼，像好酒，越喝越香；《请理解我》和《断舍离》是小书，实操简单，效果立竿见影，也许能对工作和生活带来一些改变。

14. 对做技术的同学们，有哪些建议？

潘魏增：有几个建议，也是我个人的一些成长心得，仅供大家参考吧，希望能给大家带来一点帮助。

第一个是**打好基础**。毕业刚走进职场的同学，一定要打好技术基础。「勿在浮沙筑高台」，把地基打扎实，才能在上面建成高楼大厦。怎么打地基，每个人有自己的方式，我个人比较喜欢的是看官方手册、标准文档以及阅读源代码。

第二个是**提升视野**。对于有一定经验和技术基础的工程师，建议多走出去，看看公司内其他团队是怎么做，业界是怎么做的。好的技术往往是因为看得足够多。

第三个是**思考本质**。“老司机”可以多跳出来想想商业的本质、社会的本质，毕竟技术只是这个世界很小的“子集”。我们的社会是一个非常复杂的系统，结构远比技术系统更加复杂。我也是最近几年才开始有这个认识，还在慢慢地摸索、学习。希望大家共同努力吧！

写给工程师的十条精进原则

作者: 云鹏

引言

时间回到8年前，我人生中第一份实习的工作，是在某互联网公司的无线搜索部做一个C++工程师。当时的我可谓意气风发，想要大干一场，结果第一次上线就写了人生中第一个Casestudy。由于对部署环境的不了解，把SVN库里的配置文件错误地发到线上，并且上完线就去吃晚饭了，等吃饭回来发现师傅在焦头烂额地回滚配置。那次故障造成了一个核心服务20分钟不可用，影响了几百万的用户。这仅仅是一个开始，在后来半年的时间里，我几乎把所有职场新人可能犯的错误都犯了个遍。架构师让我调研一个抓取性能提升方案，我闷头搞了两周，也没有得出任何结论；本来安排好的开发计划，由于我临时要回去写论文，搞得经理措手不及；参加项目座谈会，全程“打酱油”……那段时间，自己也很苦恼，几乎每天晚上11点多才走，很累很辛苦，但依然拿不到想要的结果。

8年过去了，自己从一个职场小白逐步成长为一名技术Leader。我发现团队中的很多同学在不停地重复犯着自己当年类似的错误。他们并不是不努力，到底是哪里出了问题？经过一段时间的观察与思考后，我想我找到了答案。那就是：我们大多数同学在工作中缺乏原则的指导。原则，犹如指引行动的“灯塔”，它连接着我们的价值观与行动。不久前，桥水基金创始人雷·达里奥在《原则》一书中所传达的理念，引爆了朋友圈。**每个人都应该有自己的原则，当我们需要作出选择时，一定要坚持以原则为中心。**但是在现实生活中，我们往往缺少对原则的总结，对于很多人来说这是一门“只可意会不可言传”的玄学，是属于老司机的秘密，其实不然。

“追求卓越”是美团的价值观。作为一名技术人员，我们应该如何践行呢？本文总结了十条精进原则，希望能够给大家带来一些启发，更好地指导我们的行动。



原则一：Owner意识

“Owner意识”主要体现在两个层面：一是认真负责的态度，二是积极主动的精神。

认真负责是工作的底线。首先，要对我们交付的结果负责。项目中每一个设计文档、每一行代码都需要认真完成，要对它的质量负责。如果设计文档逻辑混乱，代码没有注释，测试时发现一堆Bug，影响的不仅仅是RD的工程交付质量，还会对协同工作的RD、QA、PM等产生不好的影响。久而久之，团队的整体交付质量、工作效率也会逐步下降，甚至会导致团队成员之间产生不信任感。其次，我们要对开发的系统负责。系统的架构是否需要改进，接口文档是否完善，日志是否完整，数据库是否需要扩容，缓存空间够不够等等，这些都是需要落地的事情。作为系统Owner，请一定要认真履行。

积极主动是“Owner意识”更高一级的要求。RD每天要面对大量的工作，而且很多并不在计划内，这就需要具备一种积极主动的精神。例如我们每天可能会面对大量的技术咨询，如果客户提出的问题很长时间得不到回应的话，就会带来不好的客户体验。很多同学说忙于自己的工作没有时间处理，有同学觉得这件事不是很重要，也有很多同学是看到了，但是不知道怎么回答，更有甚者，看到了干脆装没看见。这些都是缺乏Owner意识的体现。正确的做法是积极主动地推动问题的解决，如果时间无法排开或者不知道如何解决，可以直接将问题反馈给能解决的同学。积极主动还可以表现在更多方面。比如很多同学会自发地梳理负责服务的现状，根据接口在性能方面暴露的问题提出改进意见并持续推动解决；也有同学在跨团队沟通中主动承担起主R的角色，积极发现问题、暴露问题，推动合作团队的进度，保证项目顺利推进。这些同学无一不是团队的中坚力量。所以，我们在做好自己份内工作的同时，也应该积极主动地投入到“份外”的工作中去。一分耕耘一分收获，不要给自己设限，努力成为一个更加优秀的人。

原则二：时间观念

相信大家都有时间观念，但是真正能执行到位的可能并没有那么多。互联网是一个快速发展的行业，RD的研发效率是一个公司硬实力的重要体现。项目的按期交付是一项很重要的执行能力，在很大程度上决定着领导和同事对自己靠谱程度的评价。大家可能会问：难度几乎相同的项目，为什么有的同学经常Delay，而有的同学每次都能按时上线？一个很重要的原因，就是这些按时交付的同学往往具备如下两个特质：**做事有计划，工作分主次。**

工作安排要有计划性。通常，RD在设计评审之后就能预估出精确的开发时间，进而再合理地安排开发、联调、测试计划。如果是项目负责人，那么就会涉及协调FE、QA、PM等多个工种的同学共同完成工作。凡事预则立，不预则废。在计划制定过程中，要尽可能把每一项拆细一点（至少到pd粒度）。事实证明，粒度越细，计划就越精准，实际开发与计划之间的误差就会越小。此外，务必要规定明确的可检查的产出，并在计划中设置一些关键的时间点进行核对。无数血淋淋的事实告诉我们，很多项目延期都是因为在一些关键交付点上双方存在分歧造成的。例如后台RD的接口文档计划在周五提供，FE认为是周五上午，而RD认为是周五下班前提交，无形中会给排期带来了1pd的误差。所以，我们要做到计划粒度足够细，关键时间点要可检查。

工作安排要分清楚主次。我们每天要面对很多的事情，要学会分辨这些工作的主次。可以尝试使用“艾森豪威尔法则”（四象限法则），把工作按照重要、紧急程度分成四象限。优先做重要紧急的事情；重要不紧急的事情可以暂缓做，但是要持续推进；紧急不重要的事情可以酌情委托给最合适的人做；不重要不紧急的事情可以考虑不做。很多项目无法按期交付的原因，都是因为执行人分不清主次。比如在开发中需要使用到ES，一些不熟悉ES的同学可能想系统性地学习一下这方面的知识，就会一头扎进ES的汪洋中。最后才发现，原本一天就能完成的工作被严重拖后。实际工作中，我们应当避免这种“本末倒置”的工作方

式。在本例中，“系统性地学习ES”是一件重要但不紧急的事情。要学会分辨出这些干扰的工作项，保证重要紧急的事情能够按时交付。

原则三：以终为始

“以终为始”（Begin With The End In Mind），是史蒂芬·柯维在《高效能人士的七个习惯》中提到的一个习惯。它是以所有事物都经过两次创造的原则（第一次为心智上的创造，第二次为实际的创造）为基础的。直观的表达就是：**先想清楚目标，然后努力实现。**

在工作中，很多RD往往只是埋头走路，很少抬头看天。每次季度总结的时候，罗列了很多项目，付出很多努力。但是具体这些项目取得了哪些收益，对业务有哪些提升，却很难说出来。这就说明在工作中并没有遵守“以终为始”这一原则。此外，很多同学在做需求的过程中，对于目标与收益关注不够，系统上线之后，也没有持续地跟进使用效果。这一点在技术优化项目中体现得尤为明显。例如在一个接口性能优化的项目中，经过RD的努力优化，系统TP99缩短了60%，支持QPS提升了2倍。但是系统到底需要优化到什么程度呢？是不是缩短60%，提升2倍就能满足需求呢？在优化之前，很多同学常常忘记设置一个预设的目标（TP99小于多少，支持QPS大于多少）。我们必须清楚，优化一定是有原因的，比如预期某节假日流量会暴增或者某接口超时比例过高，如果不进行优化，系统可能会存在宕机风险。解决特定的问题才是技术优化的最终目的，**所以要根据问题设定目标，再进行优化。**

“以终为始”，这一原则还可以作用于我们的学习中。很多同学看过很多技术文章，但是总是感觉自己仍然一无所知。很重要的一个原因，就是没有**带着目标去学习**。在这个信息爆炸的时代，如果只是碎片化地接收各个公众号推送的文章，效果几乎可以忽略不计。在学习之前，我们一定要问自己，这次学习的目标是什么？是想把Redis的持久化原理搞清楚，还是把Redis的主从同步机制弄明白，亦或是想学习整个Redis Cluster的架构体系。如果我们能够带着问题与目标，再进行相关的资料搜集与学习，就会事半功倍。这种学习模式的效果会比碎片化阅读好很多。

原则四：闭环思维

你是否遇到过这样的场景：参加了一个设计（或需求）评审，大家兴致勃勃地提了很多合理的意见，等到再次评审的时候，却发现第一次提的很多问题都没有得到改进，很多讨论过的问题需要从头再开始讨论。这种情况就是一种典型的工作不闭环。

之前看过一句话：一个人是否靠谱，就看他能否做到凡事有交代，件件有着落，事事有回音。这就是闭环思维的重要性。**它强调的是一种即时反馈闭环**，如果别人给我们分配了一个任务，不管完成的结果如何，一定要在规定的时间内给出明确的反馈。例如在跨部门的沟通会议中，虽然各方达成了一致，会议发起者已经将最终的记录周知大家。但是，到这一步其实并没有完成真正的闭环，在落地执行过程中很可能还存在一些潜在的问题。例如，会议纪要是否经各方仔细核对并确认过？会议中明确的To Do进展是什么？完成结果有没有Check的机制？如果这些没有做到的话，就会陷入“沟通-发现问题-再沟通-再发现问题”的恶性循环中。**真正的闭环，要求我们对工作中的事情都能够养成良好的思维习惯，沟通要有结论，通知要有反馈，To Do要有验收。**

“闭环思维”还要求能够定期主动进行阶段性的反馈。刚参加工作时，我接了一个工期为两个月的项目。整个项目需要独自完成，自己每天按照计划，有条不紊地进行开发。大概过了两周之后，Leader询问项目进度，虽然我已经跟他说没问题。然而，Leader告诉我，因为我每天对着电脑也不说话，让他心里很没底。这时，我才意识到一个很重要的问题，我跟Leader之间存在信息不对称。从那以后，我就时不时得跟他汇报一下进度，哪怕就只有简短的一句话，也可以明显感觉，他对我的信心增加了很多。特别是我做Leader之后，对这种闭环反馈的理解，就更加深刻了。从Leader的角度看，其实只是想知道项目是否在正常推进，是否遇到问题需要他协助解决。

原则五：保持敬畏

“君子之心，常怀敬畏”，保持敬畏之心能够让我们少犯错误。在工作中存在各种各样的规范，例如代码规范、设计规范、上线规范等等。我们必须明白，这些规范的制定一定是基于某些客观原因的，它们都是历史上无数Case积累而来的经验。团队里的每一个成员都应该学习并严格遵守，这一点对于新人尤其重要。

当我们进入到一个新的团队，请先暂时忘掉之前的习惯，要尽快学习团队既有的规范，并且让自己与团队保持一致。以编码风格为例，很多同学往往习惯于自己之前的代码写作风格，在做新公司第一个项目时，也按照自己的习惯进行变量、包的命名等等。结果在代码Review过程中，被提了很多修改意见，不得不返工重写，得不偿失。如果能够保持敬畏之心，提前了解编码规范，这种问题完全可以避免。类似的问题，还包括对上线流程不了解，对回滚操作不熟悉，对SRE线上变更过程不了解等等。除了这些显而易见的规范，还有一些约定俗成的规则。个人建议是：如果有事情拿不准，不妨多问问其他同事，不要凭自己的感觉做事情。

保持敬畏之心并不意味着要“因循守旧”。在我们充分了解这些规范和约定之后，如果觉得存在不妥之处，可以跟全组同学讨论，是否采纳新的建议，然后及时去更新迭代。其实，**让规范与约定与时俱进，也是另一种形式的敬畏。**



原则六：事不过二

“事不过二”，是我们团队一贯坚持的原则，它可以解读为两层含义。

一层含义是**“所有的评审与问题讨论，不要超过两次”**。之所以有这样的要求，是因为我们发现，很多RD都把时间花费在一些无休止的评审与问题讨论中，真正投入到实际开发中的时间反而很少。在实际工作场景中，我们经常会遇到一些不是很成熟的需求评审。这些需求文档，要么是背景与目标含糊不清，要么是产品方案描述不够细化，或者存在歧义。RD与PM被迫反复进行讨论，我曾经遇到过一个需求评审，进行了三次还被打回。同样的问题，在设计评审中也屡见不鲜。方案固然需要经过反复的讨论，但是如果迟迟不能达成一致，就会耗费很多RD与PM的宝贵时间，这就与提升研发效率的理念背道而驰。因此，我们团队规定：**所有的评审最多两次**。通过这种方式，倒逼利益相关方尽可能地做好需求与方案设计。评审会议组织前，尝试与所有相关人员达成一致，询问对方的意见，并进行有针对性的讨论，这样能够大大提升评审会议的效率和质量。如果在第一次评审中不通过，那么就只有一次机会进行复审。一旦两次不通过，就需要进行Casestudy。

“事不过二”原则的另一层含义，是**“同样的错误不能犯第二次”**。每次故障之后，Casestudy都必须进行深刻的总结复盘，对故障原因进行5Why分析，给出明确可执行的To Do List。每次季度总结会，大家自我反省问题所在，在下个季度必须有所改善，不能再犯类似的错误。孔子云：“不迁怒，不贰过”，在错误中反思与成长，才能让我们成为更优秀的人。

原则七：设计优先

“设计优先”这条原则，相对来说更加具体一些。之所以单列一项，是因为架构设计太重要了。Uncle Bob曾说过：“软件架构的目标，是为了让构建与维护系统的所需人力资源最小化。”

架构设计，并不仅仅关系到系统的质量，还关乎团队的效能问题。很多团队也有明文规定，开发周期在3pd以上的项目必须有设计文档，开发周期在5pd以上的项目必须有设计评审。在具体的执行过程中，由于各种原因，设计往往并不能达到预期的效果。究其原因，有的是因为项目周期紧，来不及设计得足够详细；有的是因为RD主观上认为项目比较简单，设计草草了事。**无数事实证明，忽略了前期设计，往往会导致后续开发周期被大幅拉长，给项目带来了很大的Delay风险。而且最可怕的是，不当的设计会给项目带来巨大的后期维护成本，我们不得不腾出时间，专门进行项目的优化与重构。**因此，无论什么时候都要记住“设计优先”这一原则。磨刀不误砍柴工，前期良好的设计，会给项目开发以及后期维护带来极大的收益。

“设计优先”这一原则，要求写别人看得懂的设计。我们了解一个系统最直接的途径就是结合设计文档与代码。在实际工作中，很多同学的设计文档让大家看得一头雾水，通篇下来，看不出系统整体的设计思路。其实，设计的过程是一种智力上的创造，我们更希望它能成为个人与集体智慧的结晶。如何才能让我们的设计变得通俗易懂？我个人认为，设计应该尽量使用比较合理的逻辑，进而把设计中的一些点组织起来。比如可以使用从抽象到具体，由总到分的结构来组织材料。在设计过程中，要以需求为出发点，通过合理的抽象把问题简化，讲清楚各个模块之间的关系，再详细分述模块的实现细节。做完设计之后，可以发给比较资深的RD或者PM审阅一下，根据他们的反馈再进行完善。好的设计，一定是逻辑清晰易懂、细节落地可执行的。



原则八：P/PC平衡

“P/PC平衡”原则，即产出与产能平衡原则。伊索寓言中讲述了一个《生金蛋的鹅》的故事。产出好比“金蛋”，产能好比“会下金蛋的鹅”。“重蛋轻鹅”的人，最终可能连产蛋的资产都保不住；“重鹅轻蛋”的人，最终可能会被饿死。**产出与产能必须平衡，才能达到真正的高效能。**为了让大家更清晰的了解这一原则，本文举两个例子。

从系统的角度看，每一个系统都是通过持续不断地叠加功能来实现其产出，而系统的产能是通过系统架构的可扩展性、稳定性等一系列特性来表征。为了达到产出与产能的平衡，需要在不断支持业务需求的过程中，持续进行技术架构层面的优化。如果一味地做业务需求，经过一定的时间，系统会越来越慢，最终影响业务的稳定性；反之，一个没有任何业务产出的系统，最终会消亡。

再从RD的角度来看这个问题，RD通过做需求来给公司创造价值，实现自己的产出。而RD的产能是指技术能力、软素质、身体健康状况，有这些资本后，我们才能进行持续的产出。在日常工作中，我发现很多RD往往只重视产出。他们也在很努力地做项目，但是每一个项目所使用的方法，还是沿用自己先前一贯的思路。最终，不仅项目做得一般，还会抱怨自己得不到任何成长。这就是P/PC不平衡的体现。**如果能在做项目的过程中，通过学习总结持续提升自己的技术能力和软素质，并将其应用于项目实施交付中，相信一定会取得双赢的结果。**

“P/PC平衡”原则还适用于很多其他的领域，例如团队、家庭等，我本人也非常推崇这一原则。希望大家也能将其作为自身的一项基本原则，努力寻找到产出与产能的平衡点。

原则九：善于提问

“善于提问”，首先要勤于提问。求知欲源于好奇心，是人类的一种本能。在工作中要养成勤于提问的好习惯，不懂就问，不要因为自己一时懒惰或者碍于情面，就放弃提问的机会。当遇到不同的观点时，也要礼貌地问出来。波克定理告诉我们，**只有在争辩中，才可能诞生最好的主意和最好的决定。**

在设计评审、代码评审这类体现集体智慧的活动中，遇到有问题的地方一定要提出来。我经常看到，很多同学评审全程一言不发，这就是浪费大家的时间。设计评审的目的，是让大家针对方案提出改进意见并达成一致，如果全程“打酱油”，那就失去了评审的意义。我们鼓励大家多提问，把自己内心的疑惑表达出来，然后通过交流的方式得到答案。

“善于提问”，还要懂得如何提问。为什么同样是参加设计评审，有的同学就能提出很好的问题，而有的同学却提不出任何问题？除了知识储备、专业技能、经验等方面的差异外，还有一点很重要：批判性思维。

批判性思维主张通过批判性思考达到理性思维，即对事物本质的认知和掌握。关于如何进行批判性思维，大家可以参考一些经典的图书如《批判性思维》、《学会提问》等。在工作中面临一项决策时，会有各种各样的意见摆在你面前，所以我们必须要学会使用批判性思维来进行分析，每个人的论据是否可靠，论证是否合理，是否有隐含的立场。同样，在阅读一篇技术博客的时候，也要使用批判性的思维，多问几个为什么，作者得出的结论是否合理？论据是否充分？只有这样，才能不断地获取真正的知识。

原则十：空杯心态

“满招损，谦受益”，“空杯心态”是最后一项原则。我觉得这也是一个人能够持续成长的前提。做技术的人，骨子里通常有股傲气，并且会随着资历、成绩的提升而不断增加。初入职场的小白，可能会非常谦虚，但是工作几年之后，专业技能逐步提升，可能还取得了一些小成就，人就会越来越自信。这时候，如果不能始终保持“空杯心态”，这种自信就会逐步演变为自满。自满的人，往往表现为工作中把别人的建议当成是批评，不接受任何反对意见，学习上也缺乏求知的动力，总是拿自己的长处去跟别人的短处做比较。其实每个人多少都会有一些自满，可怕的是不知道甚至不愿承认自满。

保持“空杯心态”这一原则要求我们时刻进行自我检视与反省。在工作中，多去跟不同级别的同学聊一聊，或者做一个360度评估，这有助于我们更加客观地评价自己。在横向对比中，多向那些优秀的同学看齐，学习他人的优点。很多同学在设计评审或者代码review过程中，针对别人提出的问题与建议，往往都采用一种对立的态度。错误地认为别人是在挑刺，是在针对自己。诚然，在某些方面，我们可能确实比其他同学想得深入，但是这不代表在所有方面都能考虑周全。对于别人的建议，建议使用“善于提问”原则里提到的批判性思维仔细分析一下，虚心地吸取那些好的建议。

工作学习就像“练级打怪”，技能储备的越多，就越容易走到最后。保持空杯心态，可以让我们发现很多以前注意不到的新能力，我们要做的就是努力学习它，将它们转化为自己能力库的一部分。

总结

以上，是我总结的工作与学习的十条基本原则。其中有的侧重于个人做事情的方法，如“Owner意识”、“时间观念”、“以终为始”、“闭环思维”；有的侧重于团队工作标准规范，如“保持敬畏”、“事不过二”、“设计优先”；有的侧重于团队或个人效能提升，如“P/PC平衡”、“善于提问”、“空杯心态”。这些原则是我多年在工作与学习中，不断总结得来的经验。希望大家面临选择时，这些原则能够起到一定的帮助和指导作用。

以原则为中心地工作与生活，让自己与团队变得更加强大。

作者介绍

- 云鹏，2014年加入美团，先后参与了美团酒店供应链体系、分布式调度系统的建设，现在负责美团旅行客户关系管理系统、基础信息服务的建设工作。

工作中如何做好技术积累

作者: 刘丁

引言

古人云：“活到老，学到老。”互联网算是最辛苦的行业之一，“加班”对工程师来说已是“家常便饭”，同时互联网技术又日新月异，很多工程师都疲于应付，叫苦不堪。以至于长期以来流传一个很广的误解：35岁是程序员工作的终点。

如何在繁忙的工作中做好技术积累，构建个人核心竞争力，相信是很多工程师同行都在思考的问题。本文是我自己的一些总结，试图从三个方面来解答：

- 第一部分阐述了一些学习的原则。任何时候，遵循一些经过检验的原则，都是影响效率的重要因素，正确的方法是成功的秘诀。
- 提升工作和学习效率的另一个重要因素是释惑和良好心态。第二部分分析了我在工作中碰到和看到的一些典型困惑。
- 成为优秀的架构师是大部分初中级工程师的阶段目标。第三部分剖析架构师的能力模型，让大家对目标所需能力有一个比较清晰的认知。

如何学习

在繁忙的工作中，持之以恒、不断学习和进步是一件艰巨的任务，需要坚强的毅力和坚定的决心。如果方法不得当，更是事倍功半。幸好我们的古人和现在哲人已经总结了很多优秀的学习方法论，这里汇总了一些重要原则。遵循这些方法必会对大家的工作学习大有裨益。

贵在坚持

有报道指出，过去几十年的知识量超过之前人类几千年的知识量总和。而计算机领域绝对是当代知识更新最快的领域之一，因此，工程师必须要接受这样一个现实，现在所掌握的深厚知识体系很快就会被淘汰。要想在计算机领域持续做优秀架构师，就必须不停的学习，掌握最新技术。总之，学不可以已。

所谓“冰冻三尺，非一日之寒，水滴石穿，非一日之功”，通往架构师的道路漫长而又艰巨，轻易放弃，则所有付出瞬间付之东流。要想成为优秀的架构师，贵在坚持！

虽然知识更新很快，但是基础理论的变化却非常缓慢。这就是“道”和“象”关系，纵是世间万象，道却万变不离其宗。对于那些非常基础的理论知识，我们需要经常复习，也就是“学而时习之”。

重视实践

古人云：“纸上得来终觉浅，绝知此事要躬行。”学习领域有所谓721模型：个人的成长70%来自于岗位实践，20%来自向他人学习，10%来自于培训。虽然这种理论存在争议，但对于工程师们来说，按照实践、学习和培训的方式进行重要性排序，大致是不错的。所以重视实践，在实践中成长是最重要的学习原则。

人类的认知有两种：感性认知和理性认知。这两种认知互相不可替代性。实践很大程度来自于感性学习，看书更像是理性学习。以学开汽车做例子，很难想象什么人能够仅仅通过学习书本知识就会开汽车。

书本知识主要是传道——讲述抽象原型，而对其具体应用场景的讲述往往含糊其辞，对抽象原型之间的关系也是浅尝辄止。采用同样精确的语言去描述应用场景和关联关系将会失去重点，让人摸不着头脑。所以，仅仅通过看书来获得成长就像是用一条腿走路。

重视实践，充分运用感性认知潜能，在项目中磨炼自己，才是正确的学习之道。在实践中，在某些关键动作上刻意练习，也会取得事半功倍的效果。

重视交流

牛顿说：“如果说我看得比别人远一些，那是因为我站在巨人的肩膀上。”我们需要从别人身上学习。从老师、领导、同事、下属甚至对手身上学习，是快速成长的重要手段。

向老师和领导学习已经是人们生活习惯的一部分了。但是从同事甚至对手那里学习也很重要，因为这些人和我们自身更相似。所以要多多观察，取其所长，弃其所短。对于团队的小兄弟和下属，也要“不耻下问”。

此外，在项目中积极参与具体方案讨论也非常重要。参与者先验感知了相关背景，并且讨论的观点和建议也是综合了发言者多种知识和技能。所以，讨论让参与者能够非常全面，立体地理解书本知识。同时，和高手讨论，他们的观点就会像修剪机剪树枝一样，快速的剪掉自己知识领域里面的疑惑点。

重视总结和输出

工程师在实践中会掌握大量细节，但是，即使掌握了所有细节，却没有深刻的总结和思考，也会陷入到“学而不思则罔”的境地。成长的“量变”来自于对细节的逐渐深入地把控，而真正的“质变”来自于对“道”的更深层次的理解。

将经验输出，接受别人的检验是高层次的总结。这种输出不仅帮助了别人，对自身更是大有裨益。总结的方式有很多，包括组织分享，撰写技术文章等等。当然“日三省吾身”也是不错的总结方式。总之，多多总结，多多分享，善莫大焉！

解答别人的问题也是个人成长的重要手段。有时候，某个问题自己本来不太懂，但是在给别人讲解的时候却豁然开朗。所以，“诲人不倦”利人惠己。

重视规划

凡事预则立，不预则废。对于漫长的学习生涯而言，好的计划是成功的一半。

长期规划

长期规划的实施需要毅力和决心，但是做正确的长期规划还需要高瞻远瞩的眼界、超级敏感的神经和中大奖的运气。对于大部分人来说，长期规划定主要是“定方向”。但遵循如下原则能够减少犯方向性错误的概率：

- 远离日暮西山的行业。
- 做自己感兴趣的事情。
- 做有积累的事情。
- 一边走一边看，切勿一条道走到黑。

短期规划

良好的短期规划应该在生活、成长、绩效和晋升之间取得平衡。大部分公司都会制定一个考核周期——少则一个月，多则一年。所以不妨以考核周期作为短期学习规划周期。本质上，规划是一个多目标优化问题，它有一系列的理论方案，这里不一一细说。基于相关理论，我给出一个简单易行的方案：

- 确定目标优先级。比如：成长、生活、绩效。
- 确定每个目标的下限。从优化理论的角度来看，这被称为约束。比如绩效必须在一般以上，之前已经规划好的旅行不能更改，必须读完《Effective Java》等等。
- 优先为下限目标分配足够的资源。比如，事先规划好的旅行需要10天，这10天就必须预算出去。
- 按照各主目标的顺序依次分配资源。比如，最终分配给学习的时间是10天。
- 在给定的学习预算下，制定学习目标，要激进。然后给出执行方案。比如，学习目标是掌握基本的统计学知识，并成为Java专家。具体方案为：完成《Effective Java》、《Java Performance》、《Design Pattern》、《Head First Statistics》四本书的阅读。
- 对规划中的各学习任务按目标优先级进行排序，并最先启动优先级最高的任务。比如，最高优先级是掌握统计理论，那么就要先看《Head First Statistics》。

对于该方案，要注意以下几点：

- 最低目标必须能够轻松达成的目标，否则，从优化理论的角度来讲，该命题无解。比如，类似“半年内完成晋级两次、绩效全部S、从菜鸟成为Java专家”就不太合适作为最低目标。总之，要区分理想和梦想。
- 主要目标规划必须具备一定的挑战性，需要规划出不可能完成的目标。过度规划本质上是一种贪婪算法，目的是目标价值最大化。因为一切皆有变数，如果其他目标能够提前完成，就不妨利用这些时间去完成更多的学习目标。总之，前途必须光明，道路必须坎坷。
- 各目标之间不一定共享资源，规划不一定互有冲突。

此外，短期规划还可以从如下几个方面进行优化：

- 学习计划最好能结合工作计划，理论联系实际结合，快速学以致用。比如，本季度规划去做一些数据分析工作，那么不妨把学习目标设置为学习统计知识。
- 要灵活对待规划的目标和具体执行步骤，需要避免“郑人买履”式的笑话。面临新的挑战 and 变化，规划需要不断地调整。

那些令人纠结的困惑

人生是一场马拉松，在漫长的征途中，难免有很多困惑。困惑就像枷锁，使我们步履蹒跚，困惑就像死锁，让我们停滞不前。

接下来我将总结自己在工作中碰到和看到的一些典型困惑。这些困惑或者长期困扰作者本人，或者困扰我身边的同事和朋友。当这些困惑被释然之后，大家都感觉如重获释，为下一阶段的征程提供满满的正能量。

量。人生就像一场旅途，不必在乎目的地，在乎的，应该是沿途的风景，以及看风景的心情。良好的心态是技术之旅最好的伴侣。期望通过这个解惑之旅，让大家拥有一个愉快的心情去感受漫长的学习旅途。

学无止境吗

必须要承认一个残酷的现实：人的生命是有限的，知识却是无限的。用有限的生命去学习无限的知识是不可能完成的任务。一想到此，有些工程师不免产生一些悲观情绪。如果方法得当并且足够勤奋，悲伤大可不必。

虽然，人类的整体知识体系一直在扩张。但是就很多重要的工程细分领域，基础理论并不高深。计算机的很多重要领域，工程师有能力在有限时间内抓住核心要害。

比如，密码学被认为是门非常高深的学科，但是一大类密码技术的基础是数论中一个非常简单的理论——素因数分解：给出两个素数，很容易算出它们的积，然而反过来给定两个素数的积，分解的计算量却非常惊人。

“一致性”算得上是计算机领域里面最经典的难题，它是所有分布式系统的基础，从多核多CPU到多线程，从跨机器到跨机房，无所不在，几乎所有的计算机从业人员都在解决这个问题，但是Paxos给出了一个很优雅的解决方案。

权限管理是很多工程师的噩梦，但如果你能搞定“Attribute Based Access Control(ABAC)”和“Role-Based Access Control(RBAC)”，也能达到相当高度。

另外，技术学习是一场对抗赛，虽然学无止境，超越大部分对手就是一种胜利。所以，以正确的学习方式，长时间投入就会形成核心竞争力。

没有绝对高明的技术，只有真正的高手

致力于在技术上有所成就的工程师，都梦想有朝一日成为技术高手。但技术高手的标准却存在很大的争议。这是一个有着悠久历史的误解：以某种技术的掌握作为技术高手的评判标准。我经常碰到这样一些情景：因为掌握了某些技术，比如Spring、Kafka、Elasticsearch等，一些工程师就自封为高手。有些工程师非常仰慕别的团队，原因竟是那个团队使用了某种技术。

这种误解的产生有几个原因：首先，技多不压身，技术自然是掌握的越多越好，掌握很多技术的人自然不是菜鸟。其次，在互联网时代来临之前，信息获取是非常昂贵的事情。这就导致一项技能的掌握可以给个人甚至整个公司带来优势地位。互联网时代，各种框架的出现以及开源的普及快速淘汰或者降低了很多技能的价值，同时降低了很多技术的学习门槛。所以，在当前，掌握某项技能知识只能是一个短期目标。怀揣某些技能就沾沾自喜的人需要记住：骄傲使人退步。

所谓“麻雀虽小，五脏俱全”。如果让你来做造物主，设计麻雀和设计大象的复杂度并没有明显区别。一个看起来很小的业务需求，为了达到极致，所需要的技术和能力是非常综合和高深的。真正的高手不是拿着所掌握的技术去卡客户需求，而是倾听客户的需求，给出精益求精的方案。完成客户的需求是一场擂台赛，真正的高手，是会见招拆招的。

不做项目就无法成长吗

在项目中学习是最快的成长方式之一，很多工程师非常享受这个过程。但是一年到头都做项目，你可能是在一家外包公司。对于一个做产品的公司，如果年头到年尾都在做项目，要不然就是在初步创业阶段，要不然就是做了大量失败的项目，总之不算是特别理想的状态。正常情况，在项目之间都会有一些非项目时间。在这段时间，有些同学会产生迷茫，成长很慢。

项目真的是越多越好吗？答案显然是否定的。重复的项目不会给工程师们带来新的成长。不停的做项目，从而缺乏学习新知识的时间，会导致“做而不学则殆”。真正让工程师出类拔萃的是项目的深度，而不是不停地做项目。所以，在项目之间的空档期，工程师们应该珍惜难得的喘息之机，深入思考，把项目做深、做精。

如何提高项目的深度呢？一般而言，任何项目都有一个目标，当项目完成后，目标就算基本达成了。但是，客户真的满意了吗？系统的可用性、可靠性、可扩展性、可维护性已经做到极致了吗？这几个问题的答案永远是否定的。所以，任何一个有价值的项目，都可以一直深挖。深挖项目，深度思考还可以锻炼工程师的创造力。期望不停地做项目的人，就像一个致力于训练更多千里马的人是发明不出汽车的。锻炼创造力也不是一蹴而就的事情，需要长时间地思考。总之，工程师们应该总是觉得时间不够用，毕竟时间是最宝贵的资源。

职责真的很小吗

很多时候，一个工程师所负责系统的数量和团队规模与其“江湖地位”正相关。但是，江湖地位与技术成长没有必然关联。提升技术能力的关键是项目深度以及客户的挑剔程度。项目越多，在单个项目中投入的时间就越少，容易陷入肤浅。特别需要避免的是“在其位不谋其政”的情况。团队越大，在管理方面需要投入的精力就越多。在管理技巧不成熟，技术眼界不够高的前提强行负责大团队，可能会导致个人疲于应付，团队毫无建树。最终“一将无能，累死三军”，效果可能适得其反。

从技术发展的角度来说，技术管理者应该关注自己所能把控的活跃项目的数量，并致力于提高活跃项目的影响力和技术深度。团队人数要与个人管理能力、规划能力和需求把控能力相适应。一份工作让多个人来干，每个人的成长都受限。每个人都做简单重复的工作，对技术成长没有任何好处。团队管理和项目管理需要循序渐进，忌“拔苗助长”。

一定要当老大吗

有一些工程师的人生理想是做团队里的技术老大，这当然是一个值得称赞的理想。可是，如果整个团队技术能力一般，发展潜力一般，而你是技术最强者，这与其说是幸运，不如说是悲哀。这种场景被称之为“武大郎开店”。团队里的技术顶尖高手不是不能做，但为了能够持续成长，需要满足如下几个条件：

- 首先你得是行业里面的顶尖专家了——实在很难找到比你更强的人了！
- 其次，你经常需要承担对你自己的能力有挑战的任务，但同时你拥有一批聪明能干的队友。虽然你的技术能力最高，但是在你不熟悉的领域，你的队友能够进行探索并扩展整个团队的知识。
- 最后，你必须敏而好学，不耻下问。

否则，加入更强的技术团队或许是更好的选择，最少不是什么值得骄傲的事情。

平台化的传说

平台化算得上是“高大上”的代名词了，很多工程师挤破头就为了和“平台化”沾点边。然而和其他业务需求相比，平台化需求并没有本质上的区别。无论是平台化需求还是普通业务需求，它的价值都来自于客户价值。不同点如下：

- 很多平台化需求的客户来自于技术团队，普通需求的客户来自于业务方。
- 产品经理不同。普通业务需求来自于产品经理，平台化需求的产品经理可能就是工程师自己。长期被产品经理“压迫”的工程师们，在平台化上终于找到“翻身农奴把歌唱”的感觉。
- 很多平台化的关注点是接入能力和可扩展性，而普通业务的关注点更多。

归根结底，平台化就是一种普通需求。在实施平台化之前，一定要避免下面两个误区：

- 平台化绝对不是诸如“统一”、“全面”之类形容词的堆砌。是否需要平台化，应该综合考虑：客户数量，为客户解决的问题，以及客户价值是否值得平台化的投入。
- 平台化不是你做平台，让客户来服务你。一些平台化设计者的规划设计里面，把大量的平台接入工作、脏活累活交给了客户，然后自己专注于所谓“最高大上”的功能。恰恰相反，平台化应该是客户什么都不做，所有的脏活累活都由平台方来做。本质上讲，平台化的价值来自于技术深度。真正体现技术深度的恰恰是设计者能够很轻松的把所有的脏活累活搞定。

所以平台化的最佳实践是：投入最少的资源，解决最多的问题。平台解决一切，客户坐享其成。

搞基础技术就一定很牛吗

经常听到同学们表达对基础技术部同学的敬仰之情，而对搞业务技术的同学表现出很轻视，认为存储、消息队列、服务治理框架（比如美团点评内部使用的OCTO）、Hadoop等才能被称为真正的技术。事实并非如此，更基础的并不一定更高深。

比如下面这个流传很久的段子：越高级的语言就越没有技术含量。但真是这样吗，就拿Java和C来说，这是完全不同的两种语言，所需要的技能完全不同。C或许跟操作系统更加接近一点，和CPU、内存打交道的机会更多一点。但是为了用好Java，程序员在面向对象、设计模式、框架技术方面必须要非常精通。Java工程师转到C方向确实不容易，但作者也见过很多转到Java语言的C工程师水土不服。

基础技术和业务应用技术必然会有不同的关注点，没有高低之分。之所以产生这种误解，有两个原因：

- 基础技术相对成熟，有比较完整的体系，这给人一个高大上的感觉。业务应用技术相对来说，由于每个团队使用的不一样，所以成熟度参差不齐，影响力没有那么大。
- 基础技术的门槛相对来说高一点，考虑到影响面，对可靠性、可用性等比较高的最低要求。但是门槛高不代表技术含量高，另外成熟技术相对来说在创新方面会受到很大的约束。但是最先进的技术都来自活跃的创新。

对比下来，业务技术和基础技术各有千秋。但真正的高手关注的是解决问题，所有的技术都是技能而已。

可行性调研的那些坑

工作中开展可行性调研时有发生。做可行性调研要避免如下情况：

- 把可行性调研做成不可行性调研。这真的非常糟糕。不可行性的结论往往是：因为这样或者那样的原因，所以不可行。
- 避免“老鼠给猫挂铃铛”式的高风险可行性方案。“天下大事必作于细”，可行性调研一定要细致入微，避免粗枝大叶。

- 避免调研时间过长。如果发现调研进展进入到指数级复杂度，也就是每前进一步需要之前两倍的时间投入，就应该果断的停止调研。

可行性调研的结论应该是收益与成本的折衷，格式一般如下：

- 首先明确预期的结果，并按照高中低收益进行分级。
- 阐述达成每种预期结果需要采取的措施和方案。
- 给出实施各方案需要付出的成本。

工程师天生不善沟通吗

实际工作中，沟通所导致的问题层出不穷。工程师有不少是比较内向的，总是被贴上“不善沟通”的标签。实际上，沟通能力是工程师最重要的能力之一，良好的沟通是高效工作学习的基础，也是通过学习可以掌握的。下面我按工程师的语言说说沟通方面的经验。

第一类常见的问题是沟通的可靠性。从可靠性的角度来讲，沟通分为TCP模式和UDP模式。TCP模式的形象表述是：我知道你知道。UDP模式的形象表述是：希望你不知道。TCP模式当然比较可靠，不过成本比较高，UDP模式成本低，但是不可靠。在沟通可靠性方面，常见错误有如下两种：

- 经常听到的这样的争论。一方说：“我已经告诉他了”，另一方说：“我不知道这个事情呀”。把UDP模式被当作TCP模式来使用容易产生扯皮。
- 过度沟通。有些同学对沟通的可靠性产生了过度焦虑，不断的重复讨论已有结论问题。把TCP模式当成UDP来使用，效率会比较低。

第二类沟通问题是时效性问题。从时效性讲，沟通分为：同步模式和异步模式。同步沟通形象地说就是：你现在给我听好了。异步沟通的形象表述是：记得给我做好了。在沟通时效性方面，有如下两种常见错误：

- 已经出现线上事故，紧急万分。大家你一言，我一语，感觉事故可能和某几个人有关，但是也不能完全确定，所以没有通知相关人员。最终，一个普通的事故变成了严重事故。对于紧急的事情，必须要同步沟通。
- 半夜三点你正在熟睡，或者周末正在逛街，接到一个电话：“现在有个需求，能否立刻帮忙做完。”这会非常令人郁闷，因为那并不是紧急的事情。不是所有的需求都需要立刻解决。

有效沟通的一个重要原则是提前沟通。沟通本质是信息交流和处理，可以把被沟通对象形象地比喻成串行信息处理的CPU。提前沟通，意味着将处理请求尽早放入处理队列里面。下面的例子让很多工程师深恶痛绝：一个需求策划了1个月，产品设计了2周。当开发工程是第一次听说该需求的时候，发现开发的时间是2天。工程师据理力争，加班加点1周搞定。最后的结论是工程师非常不给力，不配合。就像工程师讨厌类似需求一样。要协调一个大项目，希望获得别人的配合，也需要尽早沟通。

有效沟通的另外一个重点是“不要跑题”。很多看起来很接近的问题，本质上是完全不同的问题。比如：一个会议的主题是“如何实施一个方案”，有人却可能提出“是否应该实施该方案”。“如何实施”和“是否应该实施”是完全不同的两个问题，很多看起来相关的问题实际上跑题很远。“跑题”是导致无效沟通的重要原因。

良好沟通的奥秘在于能掌握TCP模式和UDP模式精髓，正确判断问题的紧急性，尽量提前沟通，避免跑题。

带人之道

有些初为导师的工程师由于担心毕业生的能力太弱，安排任务时候谆谆教诲，最后感觉还是有所顾虑，干脆自己写代码。同样的事情发生在很多刚刚管理小团队的工程师身上。最终的结果他们：写完所有的代码，让下属无代码可写。“事必躬亲”当然非常糟糕，最终的往往是团队的整体绩效不高，团队成员的成长很慢，而自己却很累。

古人说：“用人不疑，疑人不用。”这句话并非“放之四海而皆准”。在古代，受限於通信技术，反馈延迟显著，而且信息在传递过程中有大量噪音，变形严重。在这种情况下，如果根据短期内收集的少量变形的信息做快速决断，容易陷于草率。在公司里，这句话用于选人环节更为恰当，应该改为：录用不疑，疑人不录。

考虑到招聘成本，就算是在录用层面，有时候也无法做到。作为一个小团队的管理者，能够快速准确的获取团队成员的各种反馈信息，完全不需要“用人不疑，疑人不用”。用人的真正理论基础来自于“探索和利用”(Exploration and Exploitation)。不能因为下属能做什么就只让他做什么，更不能因为下属一次失败就不给机会。

根据经典的“探索和利用”(Exploration and Exploitation)理论，良好的用人方式应该如下：

- 首选选择相信，在面临失败后，收缩信任度。
- 查找失败的原因，提供改进意见，提升下属的能力。
- 总是给下属机会，在恰当地时机给下属更高的挑战。总之，苍天大树来自一颗小种子，要相信成长的力量。

效率、效率、效率

经常看到有些同学给自己的绩效评分是100分——满分，原因是在过去一段时间太辛苦了，但最终的绩效却一般般。天道酬勤不错，但是天道更酬巧。工程师们都学过数据结构，不同算法的时间复杂度的差距，仅仅通过更长的工作时间是难以弥补的。为了提升工作学习效率，我们需要注意以下几点：

- 主要关注效率提升。很多时候，与效率提升所带来的收益相比，延长时间所带来的成果往往不值一提。
- 要有清晰的结果导向思维。功劳和苦劳不是一回事。
- 做正确的事情，而不仅仅正确地做事情。这是一个被不断提起的话题，但是错误每天都上演。为了在规定的时间内完成一个大项目，总是要有所取舍。如果没有重点，均匀发力，容易事倍功半。如果“南辕北辙”，更是可悲可叹。

架构师能力模型

前面我们已经讲完了原则和一些困惑，那么工程师到底应该怎么提升自己呢？

成为优秀的架构师是大部分初中级工程师的阶段性的目标。优秀的架构师往往具备八种核心能力：编程能力、调试能力、编译部署能力、性能优化能力、业务架构能力、在线运维能力、项目管理能力和规划能力。

这几种能力之间的关系大概如下图。编程能力、调试能力和编译部署能力属于最基础的能力。不能精通掌握这三种能力，很难在性能优化能力和业务架构能力方面有所成就。具备了一定的性能优化能力和业务架构能力之后，才能在线运维能力和项目管理能力方面表现优越。团队管理能力是最高能力，它对项目管理能力的依赖度更大。

团队管理能力

在线运维能力 项目管理能力

性能优化能力 架构能力

编程能力 调试能力 编译部署能力

编程能力

对工程师而言，编程是最基础的能力，必备技能。其本质是一个翻译能力，将业务需求翻译成机器能懂的语言。

提升编程能力的书籍有很多。精通面向对象和设计模式是高效编程的基础。初级工程师应该多写代码、多看代码。找高手做Code Review，也是提升编程水平的捷径。

调试能力

程序代码是系统的静态形式，调试的目的是通过查看程序的运行时状态来验证和优化系统。本质上讲，工程师们通过不断调试可以持续强化其通过静态代码去预测运行状态的能力。所以调试能力也是工程师编程能力提升的关键手段。很早之前有个传说：“调试能力有多强，编程能力就有多强。”不过现在很多编辑器的功能很强大，调试能力的门槛已经大大降低。

调试能力是项目能否按时、高质量提交的关键。即使一个稍具复杂度的项目，大部分工程师也无法一次性准确无误的完成。大项目都是通过不断地调试进行优化和纠错的。所以调试能力是不可或缺的能力。

多写程序，解决Bug，多请教高手是提升调试能力的重要手段。

编译部署能力

编译并在线上部署运行程序是系统上线的最后一个环节。随着SOA架构的普及以及业务复杂度的增加，大部分系统只是一个完整业务的一个环节，因此，本地编译和运行并不能完全模拟系统在线运行。为了快速验证所编写程序的正确性，编译并在线上部署就成了必要环节。所以编译部署能力是一个必备技能。

让盘根错节的众多子系统运行起来是个不小的挑战。得益于SOA架构的普及以及大量编译、部署工具的发展，编译部署的门槛已经大大降低。基于应用层进行开发的公司，已经很少有“编译工程师”的角色了。但是对于初级工程师而言，编译部署仍然不是一个轻松的事情。

性能优化能力

衡量一个系统成功的一个重要指标是使用量。随着使用量的增加和业务复杂度的增加，大部分系统最终都会碰到性能问题。性能优化能力是一个综合能力。因为：

- 影响系统性能的因素众多，包括：数据结构、操作系统、虚拟机、CPU、存储、网络等。为了对系统性能进行调优，架构师需要掌握所有相关的技术。
- 精通性能优化意味着深刻理解可用性、可靠性、一致性、可维护性、可扩展性等的本质。

- 性能优化与业务强耦合，最终所采取的手段是往往折衷的结果。所以，性能优化要深谙妥协的艺术。

可以说，性能优化能力是工程师们成长过程中各种技能开始融会贯通的一个标志。这方面可以参考之前的博客文章“常见性能优化策略的总结”。市场上还有很多与性能优化相关的书籍，大家可以参考。多多阅读开源框架中关于性能优化方面的文档和代码也不失为好的提升手段。动手解决线上性能问题也是提升性能优化能力的关键。如果有机会，跟着高手学习，分析性能优化解决方案案例（我们技术博客之前也发表了很多这方面的文章），也是快速提升性能优化能力的手段。

在线运维能力

如果说性能优化能力体现的是架构师的静态思考能力，在线运维能力考验的就是动态反应能力。残酷的现实是，无论程序多么完美，Bug永远存在。与此同时，职位越高、责任越大，很多架构师需要负责非常重要的在线系统。对于线上故障，如果不能提前预防以及快速解决，损失可能不堪设想，所以在线运维能力是优秀架构师的必备技能。

为了对线上故障进行快速处理，标准化的监控、上报、升级，以及基本应对机制当然很重要。通过所观察到的现象，快速定位、缓解以及解决相关症状也相当关键。这要求架构师对故障系统的业务、技术具备通盘解读能力。解决线上故障的架构师就好比一个在参加比赛F1的车手。赛车手必须要了解自身、赛车、对手、同伴、天气、场地等所有因素，快速决策，不断调整。架构师必须要了解所有技术细节、业务细节、处理规范、同伴等众多因素，快速决断，迅速调整。

在线运维本质上是一个强化学习的过程。很多能力都可以通过看书、查资料来完成，但在线运维能力往往需要大量的实践来提升。

业务架构能力

工程师抱怨产品经理的故事屡见不鲜，抱怨最多的主要原因来自于需求的频繁变更。需求变更主要有两个来源：第一个原因是市场改变或战略调整，第二个原因是伪需求。对于第一个原因，无论是工程师还是产品经理，都只能无奈的接受。优秀的架构师应该具备减少第二种原因所导致的需求变更的概率。

伪需求的产生有两个原因：

- 第一个原因是需求传递变形。从信息论的角度来讲，任何沟通都是一个编码和解码的过程。典型的需求从需求方到产品经理，最终到开发工程师，最少需要经历三次编码和解码过程。而信息的每一次传递都存在一些损失并带来一些噪音，这导致有些时候开发出来的产品完全对不上需求。此外，需求方和产品经理在需求可行性、系统可靠性，开发成本控制方面的把控比较弱，也会导致需求变形。
- 第二个原因就是需求方完全没有想好自己的需求。

优秀的架构师应该具备辨别真伪需求的能力。应该花时间去了解客户的真实业务场景，具备较强的业务抽象能力，洞悉客户的真实需求。系统的真正实施方是工程师，在明确客户真实需求后，高明的架构师应该具备准确判断项目对可行性、可靠性、可用性等方面的要求，并能具备成本意识。最后，由于需求与在线系统的紧耦合关系，掌握在线系统的各种细节也是成功的业务架构的关键。随着级别的提升，工程师所面对的需求会越来越抽象。承接抽象需求，提供抽象架构是架构师走向卓越的必经之路。

市场上有一些关于如何成为架构师的书，大家可以参考。但是架构能力的提升，实践可能是更重要的方式。业务架构师应该关注客户的痛点而不是PRD文档，应该深入关注真实业务。掌握现存系统的大量技术和业务细节也是业务架构师的必备知识。

项目管理能力

作为工业时代的产物，分工合作融入在互联网项目基因里面。架构师也需要负责几个重大项目才能给自己正名。以架构师角色去管理项目，业务架构能力当然是必备技能。此外，人员管理和成本控制意识也非常重要。

项目管理还意味着要有一个大心脏。重大项目涉及技术攻关、人员变动、需求更改等众多可变因素。面临各种变化，还要在确保目标顺利达成，需要较强的抗压能力。

人员管理需要注意的方面包括：知人善用，优化关系，简化沟通，坚持真理。

- 知人善用意味着架构师需要了解每个参与者的硬技能和软素质。同时，关注团队成员在项目过程中的表现，按能分配
- 优化关系意味着管理团队的情绪，毕竟项目的核心是团队，有士气的团队才能高效达成目标。
- 简化沟通意味着快速决策，该妥协的时候妥协，权责分明。
- 坚持真理意味着顶住压力，在原则性问题上绝不退步。

成本控制意味着对项目进行精细化管理，需要遵循如下几个原则：

- 以终为始、确定里程碑。为了达成目标，所有的计划必须以终为始来制定。将大项目分解成几个小阶段，控制每个阶段的里程碑可以大大降低项目失败的风险。
- 把控关键路径和关键项目。按照关键路径管理理论（CPM）的要求，架构师需要确定每个子项目的关键路径，确定其最早和最晚启动时间。同时，架构师需要关注那些可能会导致项目整体延期的关键节点，并集中力量攻破。
- 掌控团队成员的张弛度。大项目持续时间会比较长，也包含不同工种。项目实施是一个不断变化的动态过程，在这个过程中不是整个周期都很紧张，不是所有的工种都一样忙。优秀的架构师必须要具备精细阅读整体项目以及快速反应和实时调整的能力。这不仅仅可以大大降低项目成本，还可以提高产出质量和团队满意度。总体来说，“前紧后松”是项目管理的一个重要原则。

项目管理方面的书籍很多。但是，提高业务架构能力同样重要。积极参与大项目并观察别人管理项目的方式也是非常重要的提升手段。

团队管理能力

不想做CTO的工程师不是一个好的架构师。走向技术管理应该是工程师的一个主流职业规划。团队管理的一个核心能力就是规划能力，这包括项目规划和人员规划。良好的规划需要遵循如下原则：

- 规划是利益的博弈。良好的规划上面对得起老板，中间对得起自己，下面对得起团队。在三者利益者寻找平衡点，实现多方共赢考验着管理者的智慧和精细拿捏的能力。
- 任何规划都比没有规划好。没有规划的团队就是没头的苍蝇，不符合所有人的利益。
- 规划不是本本主义。市场在变，团队在变，规划也不应该一成不变。
- 客户至上的是项目规划的出发点。
- 就人员规划而言，规划需要考量团队成员的能力、绩效、成长等多方面的因素。

市场上有很多规划管理方面的书籍，值得阅读。最优化理论虽然是技术书籍，但它是规划的理论基础，所以不妨多看看翻阅一下。从自我规划开始，多多学习别人的规划也是规划能力提升的重要手段。

总结

因为受邀去做一个关于“一边工作，一边学习”的分享，作者花了一段时间去思考和汇总学习方法论，接着每天不断地采集谣言并尝试解惑，再根据个人经验绘制出优秀架构师的能力模型，最后汇集成文。

文章系统性地阐述了学习原则、分析了常见困惑，并制定明确学习目标，期望对工程师们的工作学习有所帮助。需要申明的是，文章内容挂一漏万，所谓的架构师能力模型也是作者的个人观点。欢迎大家在评论中分享自己在学习成长方面的心得。



扫码关注技术团队
微信公众号

tech.meituan.com
美团技术博客