

如何解决工作中遇到的问题？

前言

本文主要从项目开发的角度来讲，大体分为解决实际遇到的问题、团队协作开发、寻找适合的团队三个方面，分析来剖析什么是小问题，什么是大问题。

组织定义

先说说问题的由来，软件公司现在叫互联网公司，是为了提升人类协作效率而产生的组织，它们的研发成果是一套套协作系统，将人与人连接起来；百度将人们的位置连接起来，淘宝把人们的购物需求连接起来，微信把人们的社交需求连接起来，谷歌把人们的搜索需求连接起来；连接的人越多，组织的价值就越大，小到社群大到公司，均是如此。

上面讲到互联网公司作为一个协作系统，将人与人连接起来，成就它的价值；而我们自身的价值，是建立在公司的价值基础之上；公司作为一个大的团队集合，满足社会群体的一部分需求，研发出一套系统来调度社会资源，比如饿了么、美团等；而实现这个系统，需要公司的一个个团队共同完成，品牌、市场、营销、项目、人力等方方面面，而项目团队又由产品、设计、开发、测试、运维等成员共同完成。

目前我们要讲的就是项目团队，如何解决问题的过程？先明确一件事情，公司需要团队，团队需要项目，项目需要开发人员。而开发过程必然会出现问题，重点是问题的大小、影响程度、当下的紧急程度，然后明确问题复杂程度，预计解决时间，与相关人员沟通，达成共识，解决问题，减少公司损失。这是相对合适的一套解决问题流程。

问题产生

关于问题是如何产生的？可能因为你的认真而产生，也可能因为你的马虎而产生，更可能因为你的无知而产生，还可能是因为团队配合而产生，更可能因为一个挖掘机而产生等等。

也就是问题除了端正态度以外，其他都是不可把控的，而程序员的一生就是加强“问题把控能力”的一生；吾生也有涯，而知也无涯。以有涯随无涯，殆已。意思是如果你追求是境遇是“没有问题”，那是不可能的，就像奇素数不可穷尽一样。所以端正态度，积极解决问题，同时承认自己能力有限，是十分必要的。

问题实例

说到这儿，前段时间，在中午点餐的时候，美团出现支付问题，重复支付但无法成功下单，影响大批用户；据说是其团队CTO在高峰期做容灾测试引起；这个事情，从CTO个人来讲，增加系统的扩展能力和弹性能力是生产力，从公司来讲，网站吞吐量大小直接影响用户规模的增长速度和服务能力（双11购物节也是从2千万做到3000亿的）。

阿里的双11指挥官振飞就很擅长做这件事，通过突然断电、关停服务器来测试系统的容灾能力，不过他是半夜做的，在网站流量最小的时候，影响不是很大，而美团的这个行为显得有些莽撞，正从饿了么手中抢用户的时候（群里发美团红包的人慢慢盖过饿了么，直到几乎全部是美团红包的事实可以说明），出现这种事情，可谓是很不专业。这个问题需要快速解决，时间越短越少，最后花费2个半小时，美团据说因此损失几千万，代价不可谓不惨痛。

第一节 解决实际问题

遇到问题，解决问题，是每个程序员都应当必备的能力，不论你是挑灯夜战，还是求助于人，或者查询论文；解决问题能力代表你能贡献的价值，一般来说公司越大，问题越细致越复杂，除了工作需要，不断解决问题，随着能力的提升，也能让你成为行业专家。

1.1 源码理解程度

不论你是做Java，还是C，又或者是Python等，第一行程序往往是从Hello World开始的。这行代码代表着“初识”，代表着管中窥豹、只见一斑，代表着破冰之旅的开始；接着你会学习变量定义、函数定义，关键字、控制符，运算、数据结构，同步、算法等，比如Android，除了Java基础外，Android发展历史、库用法、生命周期、Framework、第三方库也是需要掌握和学习的。

对源码的理解程度，不关乎解决问题的能力，而关乎解决问题的质量；通过搜索查到雷同方案，一个个拿来试，每个改一改，最终拿到一个效果，但其中原理如何，下次遇到同样问题，是否依然掉进坑里，却不得而知；当然，一直陷于解决效果、功能问题，而不研究背后运行规律，如何能叫提升了技术水平？

为什么毕业两三年，出去面试还被人当作入门选手，因为过多关注表面的东西，而忽视最本质的东西！这也是被人称道“一年经验用十年，还叫一年工作经验”的原因。

HR往往关注你毕业到现在的时间，但你的主管一定会要求，招聘一个可以独立完成工作的同事，这里的独立完成工作指可以独立分析需求、可以独立设计方案、考虑问题全面、代码健壮（可用性高）；此刻你可以对照下自己，日常工作是否漏洞百出，是否经常加班熬夜补漏洞、赶进度（接下来我们会聊一聊）。

当然，只看源码，也是只看到冰山一角，重要的是解决问题，比如设计一个生产者-消费者模式，模拟线程的运用和同步策略的设计。看源码是点；解决一个实际问题，并归纳总结叫片；熟练掌握多种解决方案叫面；一人能掌握多项技术叫多面手，也叫多技术栈或全栈，自然竞争力也要强一些。

1.2 整体设计思维

如果你听说过“如何把梳子卖给和尚”的故事，那么我们要讲的就跟它类似；解决问题的水平，一定程度上取决于你的眼界：如果单纯的劝说和尚，自然成效较小；如果把握双方的共同利益（做成“善心梳”卖给善男信女），那么成效自然较大。

而整体的设计思维，是在你连点成片，连片成面的过程，经过不断思考产生的；不要为目前的处境而担忧，坚持努力，坚持学习，坚持交流，上天不会辜负有心的人。

在初入职场之际，作者也只是知道使用一些点，达成一些片，好在大学学习的软件工程和UML设计能力还在，可以简单的做软件分析和设计，把功能、界面、存储用仅有的知识连接起来，完成一件初见成效的作品。

整体设计清晰，磨刀不误砍柴工，能有效规避相当的问题，比如开发问题、迭代问题、Bug修改问题等。

1.3 代码写作水平

当你熟悉一些点，并把点连成片的时候，就是考验代码写作水平的时候，重写还是重载，接口还是抽象，泛型还是入参，一直到注解生成代码，这些都是较为高级的写作方式；从开始的MVC设计模式，简单的把功能分类，将公共业务写在一起，慢慢的知道如何利用自己的代码影响别人，用代码规范文档约束自己和别人，再逐渐把抽象的业务拆离，形成框架发布到Github，这前进的每一步都是重要节点。

代码写作是一门艺术，跟作家写作一样，命名规范、格式清晰、逻辑明了，团队成员看到就明白，修改知道如何下手，那就达到了相当的水平。这种水平也有个不成文的说法“最好的注释就是代码结构本身”。这样，解决问题时，心情大好，解决过程也会顺利很多。

1.4 搜索目标方案

也许尚未成为大牛，无法根据业已掌握的Api，来解决若干UI或功能，那选择一件趁手的工具是必要的，那就是“谷歌搜索”。我们知道国内的百度搜索属于竞价排名的方式，即谁出钱多谁排前面，往往搜索的也只是国内的网站，而国内的网站繁杂，各种爬虫抓取后生成的网页也多，百度往往难以进行过滤，因而搜索的准确性比较低。而谷歌搜索依据文章的查看次数和有效性，利用算法清除不确定因素，推送给我们的往往是相对精准的答案，因此搜索工具只推荐用谷歌。这是加快问题解决效率的重要一环，另外英文也要够好，Stackoverflow和Github上的解决方案众多，耐心去一个个看，就像写论文检索期刊一样，那么你的解决问题效率已经排在业内前30%。

作者在2010年左右开始工作时，Github还没有流行起来，当时只是在Csdn上搜索较多，逐渐的大家开始转移到Github上，这也难怪，互联网是从美国传来的，前些年最新的技术也往往是从美国传来，大家习惯在上面学习经验，到现在大部分大公司都有自己的开源项目，成为知识分享的一片热土。而通过github搜索项目功能，也是不可或缺的能力，而搜索主要靠关键字，这时英文就派上用场了，将要查询的问题整理出来，知道自己想问什么，已经成功一半，把几个关键字输入，将star最多的几个项目打开，依次查看比较测试，最终根据实际需求进行选择即可。

另外搜索也需要具备全局性思维，当你修改一个功能时，要考虑到不污染其他功能，这时有必要将涉及到的方法或类或变量，进行全局（整个项目）查找，判断其他地方使用是否被影响，小心谨慎的修改，并将涉及到的地方进行认真测试；如果时间比较赶，可以将用到的资源，单独拎出来使用，事后再进行优化。

第二节 团队协作开发

2.1 必要开发准则

前面讲了问题的成因，那么也讲几条准则。

第一、不要跟产品争需求。他/她的需求明白了，而你的开发却刚刚开始，你不加班谁加班。把自己当成产品经理是对的，承认自己能力有限也是对的，你要做的是把能做的先做完，留下有争议的内容，在有限的时间内告知产品。由于职责关系，最终产品是由他决定的，而不是你，所以端正角色很重要。

第二、代码要有规范。无规矩不成方圆，几个月前写的代码，连自己读起来都困难，还怎么去维护。

第三、项目结构清晰。刚开始项目比较少，可以使用MVC，简单定义一下项目结构，方便开发和维护；大一点之后，推荐MVP，将View层和Model层分离，防止调用关系模糊不清；最经典的还是MVVM，使用Xml直接关联Model，与View层协调，相当于Xml变成P层，缺点在于公用模块不适用此种设计，只适用于相对独立的模块。

第四、上一节所讲的4小块的掌握程度。项目开始，先进行分析，需要哪些功能，关联度如何，怎么进行软件设计，使用什么设计模式，引入哪些库，使用哪些工具等等。同时提升“竞品分析”的能力，从中挑选出最适合目前开发项目的，这一块技巧和挑选公司意思相近，后面再做介绍。

2.2 沟通需求能力

沟通无处不在，解决问题的必要一步，就是沟通，让大家配合你的工作。

首先，定义问题。

如果能将你遇到的问题，试图弄个明白，那么你得先知道你遇到的是什么问题。比如，客户说他需要一匹马；你的理解呢，哦，客户需要一匹马，那我就争取为他找到一匹最快的马，交给客户即可；不，不，不！客户实现需要的是一个交通工具，更快更舒适的交通工具，那么你很可能就是下一个亨利·福特。对，弄清问题的本质很重要，试图脱离具体的实现，将需求抽象化，那么解决方案将变成无数种。

实际来个例子，产品说你的视频模块上有个“休止符”，这不是我想要的。那你要讲，休止符是什么？为什么要这么问，因为“休止符”是人设计的，而人的想象空间是无限的，那么你想的跟他想的是否是同一个东西，你必须确认，可能他看到的只是一个暂停键，是吗？人们的误会，往往也是认知不同而导致；为什么人会配合默契，因为经历的很多事件是类似的，认识是一致的，就这样。

其次，提出方案。

如果只会提出问题，那么你相当解决了一半问题；将问题抽象化，考虑若干方案，并分析比较，确定优先级；带着问题和方案去沟通，你会得到对方欣赏的眼光，什么叫专业能力，这就叫专业能力，将事情考虑全面，并拿着你的方案和理由。

接着，共情能力。

说到共情，这其实讲到了沟通技巧。共情指，你了解对方的专业能力、沟通方式、时间忙闲，在一定程度上，能体谅或理解对方的说法，这有助于你们尽快达成统一意见，或者纠正对方的错误理解。

最后，妥协能力。

问题提出了，方案各执一词，如果你们互不相让，那时间成本就会升高，加班危险指数增加。妥协能力，指双方各退一步，为了推进项目前进，而做出必要的牺牲，无论设计能力还是研发能力，只有项目前进才会有更好的结果，对吗？如何互不相让，那么根据职责来定，产品设计由他定，开发方案由你定，这也是最后一招；如果闹僵关系，那么以后的工作，由于沟通方式不同，可能产生更大误会；只注重当下问题的解决，而无视未来会出现更大问题的态度是不可取的。

2.3 内部技术分享

项羽再能干，也只是一个人，罗马也不是一天建成的。意思是，项目是大家一天天完成的，并非一人之力逞一时一能完成的。考虑到“木桶原理”，大家不断学习，总有人会落后，所以团队内部，进行不断的项目总结，复盘问题产生原因，讨论出解决方案，或者分享不同解决问题的技术方案，提供另外一种视角，那对于顺利完成工作是很有必要的。

2.4 求助其他同事

上面说自己再能干，也只是一个人，团队的共同前进，离不开互帮互助。如果前端实现有难度，那么问问移动端和服务端有没有方案，偶尔也是可以提一提的，毕竟大家目标

一致。今天你帮我，明天我帮你，默契是在慢慢配合中产生的。所以，提出求助和接受求助，也是一种工作技巧。

第三节 寻找适合团队

3.1 熟悉自身能力

正所谓知己知彼，百战不殆。正确认识问题的难度，安排时间去解决它；那你可能会问，一个问题的解决通常多久算正常？我说一般难度半天为限度，多则一天，不会再多；为什么？如果花时间太长，多半是原本计划出了问题，要么偏离问题的方向，要么自己并不能很好处理，需要求助他人。

有一些在处理一个异步请求，并将数据赋值到Adapter的时候，一直想做View的缓存，请求结束拿到View去设置数据，同时触发视频播放行为，结果由于View是复用的，所以往往达不到效果。小半天后，意识到这其实是两个问题，设置数据和解决视频滑到中间播放，所以最终分开后，使用两个方案解决这个问题。这样问题产生的主要原因，在于设计的时候，功能的颗粒度分的还是太粗。

3.2 明白自己底限

有些东西可以妥协，比如加班赶进度，比如产品一定要的功能，但有些不能妥协，比如身体透支，没有生活只有工作等极端情况。当然每个人都有自己的忍耐度，都有自己的利益权衡方式；但只有明白自己，守住底线，才不至于让麻烦接踵而来，问题不断出现，不仅工作问题，还有生活问题；同时只有有底限的人，才会被人尊重，才会被人信任。

3.3 明确发展方向

无论你是未来定义的是专业路线，还是走管理路线，你都需要在日常工作中发现自己，逐渐有意识的决定自己的发展方向，争取某些发展机会；如果你不擅言谈，不擅长权衡利益关系，不懂得经营，毋庸置疑走专业路线会让你更舒服一些；如果相反，建议你有机会选择管理路线，整合资源创造更大价值。

当然工作前5年是积累专业知识的阶段，我觉得从第6年起，你有必要明确自己的发展方向，因为这个时候你就是一个“专业黑洞”，由于个人的分析能力、思考方式的提升，解决问题时能力不自觉就会得到提升。

3.4 学会挑选公司

和一群价值观一致的小伙伴工作，是件开心的事，甚至大于报酬，因为工作也是生活的一部分，我们工作是为了生活，而生活却不仅仅是为了工作，它们是包含与被包含的关

系。在面试时，公司在挑选你，随着个人能力的提升，你也会挑选公司，哪些是我最想要，而你目前能提供的，这个是属于第一位的。

那么如何挑选公司，这些年来积累下来一些经验。

第一、看公司网站，如果色调陈旧，样式不规整，没有特色；连门脸都不关注的人，就不期望它的发展会好。

第二、跟人事交流时，是否会注重你的感受，是否使用谦词，是否通过邮件沟通面试要求和约定面试要素，是体现公司专业性的地方。当然如果有，是加分项，以前作者在深圳是都有邮件的，到杭州后只有30%较为专业的公司才有，可能这也是一些人要逃离二线城市的原因吧。

第三、跟面试官交谈时，除了必要的面试流程，是否根据你的经历进行面试，是否关注你的发展潜力，是否为你切实规划未来发展，都是非常重要的因素。

第四、如果不是高级管理岗位，那么招你的人水平如何，以后是否有向他学习的空间，这个也很重要，毕竟跟人学是最快的。

第五、其他各个网站的评价，百度搜索公司关键字，了解高级管理人员的战略规划，查看舆论关注度等等，都是评估一家公司的重要因素。

第四节 结论

通过阅读本文，熟悉了问题产生的本质原因，以及问题的解决过程，和其中可以改变的因素，那么就可以明确自身提升的方面，形成未来解决问题的套路，养成习惯，做一名合格的“老司机”。

最后，祝大家“车到山前必有路”，解决问题“游刃有余”，“胸中有沟壑、腹里有乾坤”，工作愉快，生活顺心！谢谢大家！

阅读小贴士：由于作者写文章比较随心，如果被烦到思路不清晰，可能阅读起来可能会有一定程度的困难，因此把重点部分标红，如果没标红就说明可以全部阅读。谢谢忍耐！