

二

36 技术落地之道：你真的知道自己要解决的问题是什么吗？

做软件开发，其实就是用软件的手段完成业务需求，而业务需求一定是用来解决某些问题的，用户的问题、老板的问题、运营的问题等等。软件工程师常常疲于奔命，开发各种需求，但是这些需求到底想要解决什么问题，开发完成以后是否真的解决了问题，实现了功能的自身价值。对于这些问题，很多开发者常常既不了解，也不关心。

我们讲一个小故事吧。北欧有一个度假胜地，是欧洲人民夏天避暑度假的好去处，去度假胜地需要经过一个长长的隧道，隧道的工程师为了保证隧道的安全使用，在隧道入口处立了一块牌子，写着：请打开车灯。

游客们开着汽车，打开车灯，穿过隧道，到达度假胜地，愉快地去玩耍了。而等他们要回去的时候，有些人却发现车子无法启动——他们忘记关闭车灯，汽车电池耗尽了。小镇的警察们只好开着自己的警车四处为游客们充电，疲惫不堪。而沮丧的游客们则在回去以后四处抱怨，分享他们糟糕的旅游经验，导致小镇旅游业大受影响，镇长压力山大。

于是人们找到隧道的工程师，要求他在隧道的尽头再立一块牌子，写上：请关闭车灯。工程师照做了以后，却发现麻烦来了：夜晚穿过隧道的游客看到牌子，虽然非常疑惑，但还是按照指示关闭了车灯，结果却发生了车祸，麻烦更大了。于是工程师不得不写上：如果是白天，请关闭车灯。结果有的游客没看到隧道入口的牌子，却看到了隧道出口的牌子，同样疑惑。为了解决新问题，工程师不得不在牌子上继续写下去……

这个场景和软件工程师们日常的工作场景是不是很相似？总有客户、老板、产品经理过来跟你说，这里需要这样一个按钮，那里需要这样一个功能。你照做了以后，发现带来了更多的麻烦，为此，你不得不在代码里不断地写if/else。你不是在解决问题，而是在制造问题。

回到这个故事，我们重新思考一下：这是谁的问题？谁能够解决这个问题？如果这是镇长的问题，那么能不能让镇长在停车场修建充电桩让游客们充电？如果这是警察的问题，那么能不能多招一个警察，专门帮游客充电。如果这是游客的问题，能不能在隧道出口立一块牌子，写上：你的灯亮着吗？提醒他们问题的存在，让他们自己去解决问题。

所以，你在每次解决问题的时候，是否想清楚了问题的本质究竟是什么？这是谁的问题？谁能解决这个问题？你在为谁解决问题？这些问题决定了你是否能真正解决问题，为公司创造

价值，也决定了你是否能选择最合适的技术去解决问题，进而提升自己的技术能力以及自己的技术影响力。

作为一个软件工程师，如果只是听从别人的指令开发代码，却不了解这些代码究竟想要解决什么问题，那么很多时候你是在制造问题，而不是解决问题，你加班加点辛苦工作只是在为公司制造麻烦。而对于你自己而言，日复一日重复执行解决方案，距离你成为一个技术专家也越来越远。

关于如何发现真正的问题，这里有几个小的建议，供你参考。

不要把解决方案当作问题的定义，而忽略了真正要解决的问题是什么

我工作这么多年来，经历过很多公司，参加过很多次技术会议，就我所见，几乎所有的技术会议都没有有意识地讨论过一个主题：这个会要解决的问题是什么？

很多时候，会议一开始就讨论解决方案。有的会议上，产品经理上来就说我们需要一个什么样的功能，请技术部门给一个技术方案和工作量评估，至于这个功能用来解决什么问题，给用户或者公司带来什么价值，几乎很少说明。有的会议上，架构师上来就说我们打算推广一个什么样的技术，请相关技术团队配合，至于这个技术用来解决什么问题，给用户或者公司带来什么价值，也几乎很少说明。

所以，这样的会议，讨论的重点就是解决方案本身：这个功能怎么做，这个技术怎么应用落地。而不是讨论真正的问题是什么：为了解决真正的问题，这个功能是不是必须要做，有没有更好的解决办法；这个技术是不是必须要上，能不能带来足够的价值。

这样的会议，即使有争论，争论的也是解决方案本身，而不是问题。关于解决方案的争论又往往陷入各种细节之中，经过一番讨论，更加不知道要解决的问题是什么了。

所以，以后参加技术会议的时候，也许不需要急于参与到讨论之中，而是要多思考：这次会议把要解决的问题说清楚了吗？需求背后真正要解决的问题是什么？当前讨论的内容真的能解决问题吗？

想清楚了这些，你会对当前的局面有更加清晰的认识，你会发现其他与会者的激烈争论，都是在盲人摸象，自说自话，彼此的关注点根本不在同一个问题上。

这个时候，你出手把大家拉回到问题本身，主导会议的讨论方向，你就会成为最有技术影响力的那个人。

你不需要去解决别人的问题，你只需要提醒他问题的存在

在有关育儿教育的经典书籍中，对于如何面对婴幼儿的哭闹，比如小孩子摔倒了，开始哭闹的时候，给出的解决方案是，不要立即鼓励小孩子，要让他们勇敢一点，自己爬起来。更不要斥责他没出息，走路不小心什么的，而是把他抱在怀里，轻轻在他耳边说，（爸爸）妈妈知道你摔疼了。重复这句话，直到小孩子不哭了，然后再跟他说，你是个勇敢的孩子，你可以自己面对的，下次你可以自己爬起来。

在这个例子中，小孩子摔倒了哭，是谁的问题？当然是小孩子自己的问题，但是他太小，又处在巨大的挫折之中，无法独自解决问题。所以，父母这时候要做的是，安抚好孩子的情绪，告诉孩子，爸爸妈妈和你在一起，理解你的痛苦。等他从挫折中恢复过来，不哭了，然后鼓励他，让他自己解决问题。

我们开篇那个隧道车灯的故事也是如此，忘了关闭车灯导致汽车无法启动是谁的问题？是游客自己的问题。谁最适合解决问题？是游客自己，他只需要关闭车灯就可以了。所以镇长设立充电桩，多招一个警察帮游客充电，都使问题更加复杂。但是游客又没有意识到问题的存在，所以不去解决问题。那么要做的事情就不是去帮游客解决问题，而是提醒他问题的存在：****你的灯亮着吗？****游客意识到问题的存在，他就会自己解决问题。

软件需求开发中，也有很多帮用户解决问题的场景。日常开发中，产品、运营、开发、测试、运维，也有很多交互合作，需要互相帮助；哪些问题对方可以轻易解决，哪些问题应该通过修改软件功能来解决，应该思考清楚。

鱼是最后一个看到水的，身处问题之中的人往往并不觉得有问题

身处问题之中的人常常并不能感知到问题的存在，正如身在水中的鱼儿看不到水一样。**太多的问题被人们的适应能力忽略掉了，直到有人解决了这些问题**，身处其中的人才恍然，原来过去的方式都是有问题的。

所以，如果你到一个新环境中，发现存在着一些问题，而身处其中的人却熟视无睹，往往不是他们有问题，也不是你有问题，可能只是他们已经适应了问题的存在，而你还没有适应。

关于问题的定义有个公式：**问题 = 期望-体验**。

到一个新环境中，大家体验差不多，但是你的期望和其他人不同，你就会感受到问题。而这种感受则可能是你出人头地的机会：如果你解决了这些问题，其他人也会明白过去的方式是有问题的，而你就是那个解决问题的人。

小结

一个技术，是不是真的能解决问题，是衡量一个技术是否有效的标准。而业务究竟遇到了什么问题，用什么样的技术才能真正有效地解决问题，是工程师在进行技术落地之前必须要考虑清楚的事情。

不去思考，真正地面对问题，总是试图用自己擅长的技术，或者业界热门的技术解决工作中看似一样，其实大不相同的业务问题，既不能够真正解决问题，为公司创造价值，也不能够提升自己的技术水平，获得真正的进步。

如果自己用技术总是能有效解决问题，在这个过程中，也会不断增强自己的技术自信，知道自己用技术可以创造真正的价值，自己可通过技术参与到改造世界的过程中，也会树立起技术的信仰。不会总是犹豫，自己是不是要转管理，是不是要转行。

思考题

隧道车灯的小故事对你有什么启发？

如果你是一个管理者，你团队某个员工工作不认真，工作效率低，是谁的问题？是公司的吗？是你的问题吗？是员工自己的问题吗？如果是员工自己的问题，你该如何提醒他问题的存在，并进而帮助他提高工作效率？

欢迎你在评论区写下你的思考，也欢迎把这篇文章分享给你的朋友或者同事，一起交流一下。

[上一页](#)

[下一页](#)