

# 关于软件开发，你老板不知道的7件事

2015-08-14 分类：程序员人生、首页精华  2人评论

本文由[码农网](#) – 小峰原创翻译，转载请看清文末的转载要求，欢迎加入技术翻译小组！

你的老板是否不理解你的工作？本文将有助于你更好地理解为什么你的老板不理解软件开发。

你的老板可能真的很棒。我在我自己的编程生涯中就遇到过几个真心棒的老板，但即使是最棒的老板似乎也常常总是不能理解软件开发。

事实上，我想说的是当涉及到不止编程的几个元素时，大多数软件开发经理都有点目光短浅。

所以，我编译了一个简短的清单，用来说明关于编程一些最让你老板、开发经理、技术大咖等等误解的方面。

## 1.技术债务最会拖累项目

工作在一个满是技术债务的代码库上，就像是在烂泥堆中奔跑。起初，在泥浆还不是很厚的时候，勉强强走过去还没问题，但当有个1米深的时候，你就寸步难行了。



我最喜欢的作家之一，《7 Habits of Highly Effective People》的作者，Steven Covey，称之为“P/PC平衡”或“产量vs产能。”

通常，管理人员和其他非技术性人员在推动生产力的时候，宁愿牺牲质量——就像杀掉了下金蛋的鹅一样——从而招致技术债务。

当然，通过绞拧这只鹅的脖子，威胁它，你或许暂时可以得到更多的蛋，但用不了多少时间，死去的鹅就永远不会再产蛋了。

如果你的老板正遭受着不懂技术债务的痛苦，不知道技术债务是如何正在拖累你的，那么建议你可以给他讲讲《7 Habits of Highly Effective People》中关于P/PC平衡中技术债务的条款。

大多数管理人员可能都会看过这本经典的书，所以比起你说写新功能很难是因为代码库太糟糕，还不如说说书中的观点，更容易被他们所理解。

## 2. 预估大多是废话

---

软件开发中的预估大多是废话。

这一点，你知我知，甚至团队可怜的项目经理也知——当然，也有可能他不知道，但是他应该知道。

预估软件开发中的任何事情都是非常非常困难的，因为各种意外会让你防不胜防。

每一个软件项目，每一项任务都是新的。每次你坐下来写代码，总有一些意想不到的狗屎事情发生。

但事情就是这样。没有人应该为此负责。不是你的错，也不是我的错，不是任何人的错。它就是要发生。

然而，我们依然情不自禁地会去玩这个“预估”游戏。

“Joe，你建立客户登录页面需要多久？”

“哦.....呃.....” 随便想了个时间，“2天.....哦等等——” 忘了CYA加倍。 “——要4天。”

“好吧，那我给你5天”。

“好，那就5天。”

还有一个很好的解决办法是把任务分解到足够小的程度，将所有的预估控制在4小时以内。

经验告诉我，半天时间内的预估，通常能让你体面地完成工作。超过这个点，那你就是废话了。

## 3. 可以立刻或快速完成

---

催促专业人士通常是一个糟糕的主意。

我用我从写代码到现在超过15年的时间里，学到了这个道理，所以我知道当我雇用别人做一些事情的时候，如果我催促他们，没准他们会按时完成，但结果将是无用功。



不幸的是，我发现许多软件开发经理似乎不知道这个普遍真理。不知怎的，他们认为代码可以得既快又好。

可不要误解我的意思，我也承认是有例外的。有时，你的确可以快速地写出好的代码，但通常而言总是慢工出细活。

同样不幸的是，大多数[程序员](#)，当被告知要快点完成任务的时候，往往会选择走捷径，通过牺牲质量来加快进程。

更不幸的是，这样的“代码快手”经常被当作英雄称赞，因为他们能够更快地完成任务，因为他们从不推迟或要求更多的时间。

然而，这些“代码快手”往往会将代码写得乱七八糟，给其他人留下一连串的技术债务。

如果你正在打交道的开发经理不明白，快与好之间的关系，那么你最好拿出一些统计数据，以说明后期修复bug比前期预防要耗费更高的成本。

组织越大，以及涉及的公事程序越繁琐，那么快速完成任务比正确完成任务的最终成本就会越高。

（对于这个问题的经典书籍——《人月神话》。）

## 4.有的开发人员实际上是在帮倒忙

有一个事实是，我们都碰到过那种对团队弊大于利的程序员。

不同的程序员，他们的软件开发领域能力和技能水平有着巨大的差异。

事实上，有些软件开发人员糟糕到他们在工作中写的每行代码都是在浪费公司的时间和资源。

这种类型的开发人员也许应该付钱给公司，而不是公司付给他钱。

这对你而言可能是显而易见的，但对老板却不尽然。比如说，在你看来，可能Joe是一个彻底的失败者，是需要被解雇的，因为他只会干些“点金成石”的蠢事。所有他接触的东西都变成了没用的“石头”。

但是如果你的老板不明白团队中有这些人比没有更糟，那你能做些什么？

好吧，大多数软件开发人员都怕自己被认为是一个爱搬弄是非打小报告的人——我完全理解。

但是.....你必须这么做。这是对的。如果某人确实是团队中的害虫，那么让管理人员知道这一点也是你的份内事。

我知道这将会令你陷入不舒服的处境，但是如果你不报告，那你就不是称职的程序员。你会成为杀死项目的帮凶。

至于所谓的报告，只要谨慎措辞和给一些提示就可以。

比如你可以这么说：

“嘿，我不喜欢做这种事情，但我觉得，如果我是你的话，我会想知道是否有人直接妨碍了团队。所以，我觉得这是我的责任来告诉你一些我一直在观察的东西。

.....

当然，这些都只是我的观察，所以请和其他团队成员商量一下，根据你自己的经验来判断。”

或者，如果你也可以使用这种不那么婉转的方式：

“嘿！Joe很菜。他写代码实在是太慢了。事实上，他唯一可取之处就是他的慢，因为自从他来了之后，项目就基本上只能以蜗牛的速度前进。你再不正确了解他就晚了。”

## 5.更好的设备是你可以投资的最便宜的生产力之一

---

我非常讨厌程序员告诉我——他们的小气鬼老板不给他们配备第二个显示器，或者让他们用的是已经5岁高龄的电脑。

老实说，我真心不理解为什么有的软件开发经理不同意为那些8k+美元的程序员花费2k美元改善硬件——这是很划算的投资。



老旧的硬件装备让程序员真心很懊恼，很烦躁！

每当有程序员抱怨这种情况时，我通常会建议他们另谋高就，因为这种老板的愚蠢恐怕无药可治。

什么都别说，我告诉你：

好的设备能让软件开发人员一天多干一小时的活，让开发人员更有生产力。即使只有我说的数值一半的量，加起来的总时间也不会少。

如果算一年250个工作日，那么就是 $250 \times \$35 \text{ 美元} = \$8750$ 。即使你砍去一半或四分之一，这依然是个不错的投资。

如果你的老板不懂这个道理，那么老实说，我不认为你能有什么办法。如果你真心喜欢你现在的工作，那么就只能自己给自己投资了。

## 6.新技术的风险其实没你认为得那么高

---

没什么比提及.NET最新最棒的JavaScript框架版本更让软件开发经理感觉恐惧的了。

这已然成为了大家心照不宣的雷区。

曾几何时，每隔一年左右软件供应商才发布框架或补丁的新版本，因此，错误的代价会相当高。

曾几何时，大部分源代码是封印在“墓穴”中的，不允许其他任何人访问的，所以一旦该公司不再支持它，你就会进退两难。

但是现在，情况有所不同。

如今的框架，有时甚至是在每天的基础上发布补丁，并且大多数流行框架都是开源的，所以风险并不高。

当然，你也可以破罐子破摔，但这种情况不多，并且只需要稍作修改即可，而不是大刀一挥，好的坏的都割掉。

所以，如果编程经理依然还活在1980年，那么你可能需要为他指出事情发生了什么变化，以及为什么停留在框架或库的旧版本比升级它更危险。

恐惧策略中的安全漏洞就是一个很好的突破口。

## 7.画蛇添足的业务分析师和项目经理

---

我知道接下来我可能会得罪不少人，但我不在乎。我在这里说的都是真话，我是那种看到什么就说什么的人。

当然，首先要声明的是，还是有一些好的业务分析师和项目经理的——但说实话，大部分的业务分析师和项目经理毫无价值。





曾经一度这些角色是开发项目所必要的。

但是现在，在大多数情况下，我们不再需要他们了。

软件开发人员应该直接与客户对话，以便于让他们自己搞清楚客户的需求。所以我们不需要业务分析师。

这是一个残疾岗位，因为他们做的是软件开发人员应该做的工作的一半，却对另一半工作于事无补。

而项目经理更神奇，他的目标似乎就是奔着妨碍我们开发，将一切搞得一团乱而来的。

我知道他们是好意，但在当前的敏捷世界，项目经理已经发挥不了作用了，所以还要他们干什么呢？他们四处走动试图让自己显得重要，试图找出他们可以做的事情，最终却只会妨碍大家。

很多软件开发经理的想法仍然停留在过去，停留在那个职位更有意义的年代，他们听信了世界500强咨询公司的所谓的“潮流”——据这些公司所言，很多软件公司需要更多的高薪顾问人员来满足这些工作岗位。

如果你的经理还是不懂，那么唯一的解决方法就是接受敏捷教育。

我不建议你直接告诉你的老板说“业务分析师和项目经理纯粹是在浪费氧气”——这可能也不那么容易被接受——所以，相反的，我会专注于说明一支敏捷团队应该如何工作以及需要哪些角色。

然后很明显在一个真正的敏捷环境中，是没有业务分析师和项目经理的，他们可能更适合作为Scrum管理员或产品所有者。

## 积极主动

---

当然，上面我说的这些东西，有的我可能会用一种开玩笑的口吻说。但在现实中，软件开发经理和软件开发人员之间对于软件开发的理解常常是脱节的。

我敢肯定，软件开发经理会抱怨说，软件开发人员不懂业务方面，不知道安排会议安排的难点——但那是另一回事了。

无论如何，关键点是：这不是一个对立的局面，而是一个可以解决的关于沟通不畅的问题——至少在一定程度上——可以通过合理的交流沟通解决。

采取积极而不是对抗的方式，往往才是解决这些问题的最佳途径。

你有什么高见吗？欢迎分享。

译文链接：<http://www.codeceo.com/article/7-things-your-boss-not-know.html>

翻译作者：[码农网](#) – 小峰

[ 转载必须在正文中标注并保留原文链接、译文链接和译者等信息。 ]