

浅析软件项目管理中10个误区 PDF转换可能丢失图片或格式
，建议阅读原文

https://www.100test.com/kao_ti2020/65/2021_2022__E6_B5_85_E6_9E_90_E8_BD_AF_E4_c41_65180.htm

随着计算机硬件水平的不断提高，计算机软件的规模和复杂度也随之增加。计算机软件开发从“个人英雄”时代向团队时代迈进，计算机软件项目的管理也从“作坊式”管理向“软件工厂式”管理迈进。这就要求软件开发人员特别是软件项目管理人员更深一步地理解和掌握现代软件工程的理论方法，完成思想观念上的转变。笔者在此分析了10个在现代项目管理中思想观念上容易陷入的误区，希望能够抛砖引玉，引发大家更多的思索和讨论。

误区1：在项目的需求分析阶段，开发方与客户方在各种的问题的基本轮廓上达成一致即可，具体细节可以在以后填充。因为无论开始时有多么细致，以后对需求的修改几乎是必然的。分析：这是一种非常危险的思想。实际上许多软件项目失败的最主要的原因就是需求阶段对问题的描述不够细致，导致后来预算超出或者时间进度达不到要求。正确的做法是：在项目需求分析阶段，双方必须全面地尽可能细致地讨论项目的应用背景、功能要求、性能要求、操作界面要求、与其他软件的接口要求，以及对项目进行评估的各种评价标准。并且，在需求分析结束以后，双方还要建立可以直接联系的渠道，以尽早地对需求变动问题进行沟通。

误区2：软件项目的需求可以持续不断的改变，而且这些改变可很容易地被实现。分析：的确，在具体实际中由于种种原因客户方很难在需求分析阶段全面而准确地描述所有问题。随着开发进度的推进，往往会有一些需求的改变。而现代软件

工程理论也利用软件的灵活性特点通过各种方式来适应这种情况。不过，这并不表明“软件项目的需求可以持续不断的改变，而且这些改变可很容易地被实现”。实践表明：随着开发进度的推进，实现软件需求更改所需要的代价呈指数形式增长。假定在需求分析阶段实现需求更改需要花费1倍的代价；那么，在系统设计和编码阶段，需要花费1.5-6倍的代价；在系统测试阶段需要花费10-20倍的代价；在软件版本发布以后，甚至可能要花费60-100倍的代价。由此可见，在项目开展过程中，软件需求的改变应当尽量早地提出。这样才可能花费少，容易被实现。

误区3：软件程序主要由代码组成，因此编码阶段是整个软件项目的最重要的阶段，应该给与大量的时间，并且集中主要的资源。

分析：与以前相比，由于软件的规模和复杂度的增加，以及半自动化软件代码开发平台的出现，现代软件项目管理的中心发生了转移不是着重编码阶段，而是着重系统总体/详细设计阶段。一般说来，在现代软件项目管理中各种资源的合理分配比例是：项目论证、风险评估阶段3%，项目需求分析阶段8%，系统总体/详细设计阶段45%，编码阶段10%，系统测试阶段34%。

误区4：为了便于代码的维护修改，在系统的详细设计阶段文档工作应该做到写出所有程序的伪码。

分析：通常伪码的最大作用是对程序的算法流程进行描述，便于人们深入了解程序的功能和实现过程。可见，在一定程度上伪码的确有利于对程序代码的维护和修改。但是，我们知道为了保证项目文档和程序代码的一一对应关系，维护程序代码的时候同时需要对项目文档进行维护。伪码和程序代码是非常接近的，对伪码进行维护的话，相当于进行了2倍的程序代码维护。工作量是

很大的。所以切合实际的方式应该是对一般的程序文档做到程序流程图即可，对于涉及了较复杂算法的才需要伪码。

误区5：既然在项目人员配置中设置了专门的测试人员，那么软件所有的内部测试工作全部应该由测试人员完成。分析：软件程序测试可以分为“白盒法”和“黑盒法”两种方式。由于使用“白盒法”对测试人员各方面素质的种种要求，在进行程序测试时测试人员总是最优先使用“黑盒法”。他们的工作方式往往是先对程序进行“黑盒法”测试；如果测试没有通过，不得已这才考虑对程序代码进行“白盒法”测试。显然，这种对“白盒法”有意无意的“逃避”，对软件的可靠性和稳定性构成了威胁。如何解决这个问题？一方面需要提高对测试人员的要求，另一方面也需要程序员完成部分的“白盒法”测试（实际上，程序员往往也是进行“白盒法”测试的最佳人选）。

误区6：软件项目管理只是相关技术部门的事情，与公司其他部门无关。分析：在竞争日益激烈的今天，软件项目规模大、复杂度高而且时间要求紧迫。要想提高公司的软件项目管理水平，这就需要通过提高公司的整体参与意识，需要公司各个部门协同作战。例如需要会计部门协助进行项目预算，财务管理和费用控制；需要研究部门（技术委员会）指派专家协助进行各种风险评估，提供技术指导；需要后勤部门提供各种保障。

误区7：在开发进度滞后的情况下，可以聘请更多的程序员加入到开发团队中，通过增加人力资源来赶上进度。分析：在注重团队开发的时代，开发方应该根据目前的软件项目管理水平慎重考虑这个做法。如果新加入的程序员对目前软件项目的应用行业有一定了解，并且可以很快适应了开发方的项目管理方式、软件开发风

格、团队协作氛围；那么“新人”的加入是有益的。否则，可能会“好心好意做坏事”。因为尽管其个人能力很高，但是为了使其与大家一起协同工作，开发团队不得不分出人手对其进行与项目有关的技术/业务培训，更重要的（也是难度最大的）是还要引导其融入团队。这可能需要花费开发团队许多时间和精力，很有可能使项目进度更慢。

误区8：技术骨干应该成为项目的项目经理，项目经理一定是所有项目成员中薪水最高的。分析：在“软件作坊”时代，这是一种普遍使用而且效果不错的方法；而在“软件工厂”时代，这种方法却带来各种问题，有时甚至直接导致项目失败。究其原因这主要是因为随着现代软件开发分工的细化，对项目经理的要求也发生了根本的改变最注重的不是其对某项专业技术的掌握程度，而是其组织、领导、协调开发团队的能力（当然，可以两者均突出最好）。至于项目经理的薪水问题，这和定薪制度有很大关系。通常，项目经理执行的是管理人员的薪酬体系，而其他人员执行的是技术人员的薪酬体系。项目经理的薪水在项目成员中是比较高的，但不一定是最高的。有时候，为了激励技术人员，项目中的技术骨干得到的酬劳比项目经理要高。

误区9：只有项目经理以及部门主管才会关心项目整体进度，程序员只关心自己的开发进度。分析：这是一种“官僚”的想法。实际上程序员作为团队中的一员，他不仅仅是在打一份工，更重要的是在参与一件“作品”的创作。在体味工作的辛苦的同时，程序员更重要的是要享受创作的快感。项目经理不应该漠视程序员对“成就感”的追求，应该向每一个人详细描述最终“作品”将会如何美妙和令人兴奋，并且在到达最终目标的路上设立一系列的里程

碑。每当项目整体推进到一个里程碑的时候，项目经理应该把这个消息告诉每一位项目成员。实际上，这不仅仅可以让所有的项目成员享受到阶段胜利的喜悦，还可以激发大家更大的工作热情，提高工作效率。误区10：为了保证项目继续，为了留住核心程序员，加薪吧。分析：加薪可以说是很多企业在挽留程序员时所使用的常用方法。这一招可能暂时奏效，不过往往是人留下来，但副作用也来了加薪的人未必见得多干活，没有加薪的人却开始消极怠工了。其实，项目的进行过多地依赖程序员的个人技术是“作坊”时代沿袭下来的“陋习”。既然IT行业人员的流动是无法控制的，现在项目的执行应该更加注重团体的力量，应该更多的考虑公司整体技术水平和核心技术能力。例如形成公司自己的专家知识库，类/函数库，第三方控件库，拥有自主知识产权的开发平台等。另外，实际上程序员萌生去意的原因很大程度上不是薪水，而是缺少激励和尊重。这需要项目经理使用“老土”一点的办法，找适当的时机对程序员做一做思想工作，向其描述项目的美好未来，让其感受关心和尊重。总之，要从多方面着手保证项目的顺利开展，而不是简单地加薪。100Test 下载频道开通，各类考试题目直接下载。详细请访问

www.100test.com