

需求“沙漏”的实践：产品线需求Vs具体项目需求 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__E9_9C_80_E6_B1_82_E2_80_9C_E6_c41_65503.htm 沙漏之喻 软件工程其实是人们希望从工程领域中学习经验、借鉴理论来帮助解决在复杂系统和软件开发中遇到的问题。然而，随着软件工程的实践，越来越多的人认识到软件的生产 and 造桥铺路等工程项目的最大不同就是在于开发过程中人的灵活性和创造性。现在软件工程的发展趋势也重视和体现到这一点，既需要包含和鼓励个体的灵活和创造，但同时也希望从工程的角度对活动进行规范，对创造的过程和结果进行很好的保存和展现。产品/项目研发中的需求活动贯穿整个生命周期，前期的市场部门的灵活沟通，广泛收集的活动到后期研发团队紧扣需求、巧妙分析、严格设计的过程不仅反映了工程的力量，也体现了“艺术”的技巧。但如何将客户的凌乱的需求和最后的严谨的解决方案联系起来，如何将人的活动和工程的工作平衡起来，如何在更高的层次分析和分配需求都是系统工程的重点，也是实际工作的难点。上大学时非常喜欢哲学课，它能一个简单的问题通过辩证复杂化，可把一个复杂的问题通过统一简单化。这篇文章无意哗众取宠地追求哲学上的高度，只是想通过以沙漏为模型，以需求为主线，除去细枝末节，更深刻地来认识此间的概念和问题。随着讨论的展开，我们可以看到需求活动的沙漏之喻是如何帮助解答实际工作中的困惑和问题。从“问题”到“答案”沙漏是一种古代的计时工具，以它的式样来刻画需求的过程显得非常恰当。图1中沙漏的上端对应需求捕获的过程，沙漏的下面是需求开发找

到答案的过程。在需求捕获的过程中，我们需要尽可能的扩展我们的思路，争取能收集所有可能对最后系统或产品产生影响的信息。然而，当沙漏的口开得很大的时候，收集的信息是高度分散的、凌乱的和非结构化的，有些需要还可能是互相矛盾和冲突的。因此，我们需要沙漏的筛选：对要求解决的问题进行梳理，对系统的范围做出决定，选择那些合适的，现实的需求，需求就得到了精炼。这时候问题陈述就是对系统要解决问题的陈述，而不是所有问题的陈述。通过这样一个漏斗的过程，漏下来的需求就是我们系统要满足的需求，这个时候的需求是一个正式的结构化的信息以交付给开发团队。在这个基础上，就可以设计解决方案。怎么来做需求精炼和筛选，下文会有更详细地讨论。在这里，我想要谈的是区分问题领域和解决方案领域，这也是Jeremy Dick给我们的忠告之一。经常遇到这样的情形，客户抱怨花了钱没有得到有用的产品，而开发团队也会觉得很郁闷因为拥有这么多而好的功能的系统客户却不懂得“欣赏”，不愿接受。有用和既多且好的功能，这看似统一的表述在这里却成了矛盾。什么对客户来说是有用的？其实很简单，能帮助他解决问题是有用的。而那些拥有很多强大功能的系统和产品如果做不到这一点，矛盾就不可避免。很多团队负责需求收集的人员有着很强的技术背景，习惯将思考的出发点放在系统应该有什么样的功能，怎么实现这些功能。也就是说，他们往往在没有深入理解客户问题的情况下直接进入解决方案，而不是首先定义独立于解决方案的全面而真实需求集合。那么，如何有效地区分问题领域和解决方案领域呢？其实，从需求的原始陈述开始，到最后的系统实现，整个过程是连续的：

体现了从问题到解决方案的持续演化；同时也是离散的，各个阶段的需求信息之间应有明显的差异。最关键的区分在涉众需求和系统需求之间。涉众需求描述相关涉众、用户的想要解决的问题，期望达到的效果；而系统功能需求则是刻画为了解决提出的问题，相关的产品和系统应该具有怎样的功能。通过下表的对比，我们可以清楚地看到两者之间的差别，特别是在最后一项，两者在文字描述的差异上更显示出立足点的不同。由此可知，我们只有在充分理解用户想解决的问题的基础上，才能正确地分析出系统应该提供哪些功能。首先，研发团队一定要注意对涉众真实需求的收集和理解。更进一步看，在涉众需求的收集和定义阶段，风险主要存在于定义了错误的需要解决的问题；相对去后续阶段的设计而言，在定义系统需求阶段，系统工程师应该只是抽象地描述解决方案，这个阶段的风险存在于不必要地带入过多地设计约束，影响详细方案的可能选择。可能有读者会抱怨或怀疑这个思想是不是太理想了：用户常常就告诉开发团队他关于解决方案的设计和想法，这时候怎么办？这就更要求我们的需求捕获人员有清晰的思路，首先，“用户可能自己也不知道自己需要什么”，这是常常发生的事情，我们需要适当地提出“为什么”来引导客户；其次，假定客户知道自己想要的，那么客户提出关于实现的想法也反映了他的潜在目标，也就成为了我们设计时的约束。市场和研发的平衡在很多时候上了一定规模的公司中，有两个团队是一定存在且非常重要的：市场部和研发部。市场团队主要和“人”打交道，进行市场的调研和需求的获取，这处在图2中沙漏的上半部分。在沙漏的下半部分是工程领域，也就是开发队伍工作的领域。对

开发人员来说，他们很少有机会真正地面对客户，他们工作的基础就是需求。对一个项目或产品的成功，这两个团队必须紧密合作，扮演好自己的角色。一旦此间的平衡被打破，团队就会遇到麻烦。很多公司里市场部门“坐大”，为了赢得客户的定单，或不深入理解客户的要求，或是直接“指示”研发部门按照自己对系统实现的理解进行开发。这都违反了上文中提及的需求捕获的关键原则。当另一方面，我也遇到这样一些公司，它们是由研发起家，整个研发部门在公司中占有比较多的“话语权”。会有这样的开发经理或人员，可能是出于追求技术领先或完美的热情，在满足用户需求的同时，也会增加设计/实现其他“多余”的功能点。这些功能点其实是“无源之水”，并没有实际的市场和用户需求与之相对应。需求将市场部和研发部紧密地联系起来，让两者之间的有效连接并维护平衡的角色就是团队中的需求专家。市场部和直接的用户，以及相关系统的投资方、监管方、维护方等等进行交流，我们不能保证他们能够提供准确的清晰的需求描述，因为他们只是在提出他们的问题和对结果的期望，而且要求我们的客户去学习什么是好的需求和如何提需求是不现实的。现实的是团队中有这么一些人，熟悉领域且受到过需求管理方面的指导和培训，他们知道应该怎么去收集、分析和总结需求。例如在西门子，负责捕获客户需求的工作主要由市场部人员去做，但并不是直接将这些需求转给开发部。有一个需求管理小组，隶属于研发中心，主要职责则相当于市场部和开发部的一个接口。需求工程师对从市场部那里获取的零散和庞大的需求文档进行整理和分类，然后提交给开发部。开发人员不需要和市场部争论哪些事情做得到

还是做不到，而是由需求工程师来做。（具体参见《探索需求管理的三步曲》《程序员》2005年9月刊）这是一个很好的做法，需求工程师可以是一个职位，也可以是一个角色。单独的职位可能在小的团队中不太实际，但拥有这个角色的人自己必须很清楚自己的位置：是处在市场和研发中间，需要维护好平衡，做好接口的工作。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