

<<交互设计之路>>

图书基本信息

书名：<<交互设计之路>>

13位ISBN编号：9787121021626

10位ISBN编号：7121021625

出版时间：2006-3

出版时间：电子工业出版社

作者：库帕

页数：230

译者：Chris Ding

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<交互设计之路>>

### 内容概要

本书是基于众多商务案例，讲述如何创建更好的、高客户忠诚度的软件产品和基于软件的高科技产品的书。

本书列举了很多真实可信的实际例子，说明目前在软件产品和基于软件的高科技产品中，普遍存在着“难用”的问题。

作者认为，“难用”问题是由这些产品中存在着的高度“认知摩擦”引起的，而产生这个问题的根源在于现今软件开发过程中欠缺了一个为用户利益着想的前期“交互设计”阶段。

“难用”的产品不仅损害了用户的利益，最终也将导致企业的失败。

本书通过一些生动的实例，让人信服地讲述了由作者倡导的“目标导向”交互设计方法在解决“难用”问题方面的有效性，证实了只有改变现有观念，才能有效地在开发过程中引入交互设计，将产品的设计引向成功。

本书虽然是一本面向商务人员而编写的书，但也适合于所有参与软件产品和基于软件的高科技产品开发的专业人士，以及关心软件行业和高科技行业现状与发展的人士阅读。

## 作者简介

Alan Cooper, 作为20世纪70年代中叶的一名软件发明家, Alan Cooper坚信必然存在一种更好的方式创建软件。

新的方法, 应该通过应用“使用者第一、硅片第二”原则的设计和工程过程, 使软件使用者从让人厌烦、困惑、不恰当的软件行为中解放出来。

采用这种方法, 技术团队能够在第一时间做对事情, 因而, 也就能更快地创建更好的产品。

他的信念结出了硕果。

1990年, 他创建了Cooper公司, 一家技术产品设计公司。

今天, Cooper在软件设计方面的创新手法已经被认可为行业标准。

在Cooper打开面向商业的大门后的十年时间里, 这家旧金山公司向诸如Abbott Laboratories、Align Technologies、Discover Financial Services、Dolby、爱立信、富士通、富士通Softtek、惠普、Informatica、IBM、罗技、Merck-Medco、微软、Overture、SAP、SHS Healthcare、Sony、Sun Microsystems、Toro公司、Varian, 以及VISA公司提供了创新的、以使用者为焦点的解决方案。

Cooper团队提供他们

## &lt;&lt;交互设计之路&gt;&gt;

## 书籍目录

第1篇 电脑的逆向文化1 信息时代的谜语将电脑置于机舱，你会得到什么将电脑和照相机结合在一起，你会得到什么将电脑和闹钟结合在一起，你会得到什么将电脑和汽车结合在一起，你会得到什么将电脑和银行结合在一起，你会得到什么电脑更容易导致麻烦商业软件也同样遭殃将电脑和军舰结合在一起，你会得到什么技术的愤怒整个行业都在拒绝承认本书的起源2 认知摩擦与物理力量无关的行为设计是一个重要的词程序员和交互设计师之间的关系大多数软件是偶然设计的“交互”设计VS.“界面”设计为何基于软件的产品与众不同跳舞的熊添加功能的代价辩护者和幸存者我们如何应对认知摩擦消费力量日渐平民化对使用者进行谴责软件的种族隔离第2篇 将使你付出巨大的代价3 浪费金钱期限管理“完成”的软件是什么样的帕金森定律永远交付不了的产品推迟交付并不会带来伤害对功能列表的讨价还价在程序员的控制之下功能多未必就好迭代与不可预测的市场坏软件的隐藏成本惟一比编写软件更昂贵的事情是编写坏软件失去机会的代价建造原型的代价4 跳舞的熊如果有问题，为什么不立刻解决掉消费电子类产品的受害者电子邮件软件如何失败日程计划软件如何失败日历软件如何失败3W的神秘面纱软件出什么问题了软件健忘软件懒惰软件吝于提供信息软件不灵活软件责备使用者软件不负责任5 客户叛离期望性对比面市时机第3篇 用叉子喝汤第4篇 交互设计第5篇 夺回控制权

## &lt;&lt;交互设计之路&gt;&gt;

## 章节摘录

第一版的前言 研究交互设计商务案例的书 我原本想写一本与本书截然不同的书：一本讲述如何进行交互设计的技术类书籍。

但是，在1997年5月我去托斯卡纳的探亲旅途中，我的好友Don McKinney和Dave Carlick曾谈及此事，他们说服我应该首先为商务人员编写一本书。

他们知道我在计划编写一本关于如何进行交互设计的书。

他们尽管表示了鼓励，但同时也怀疑交互设计有哪些需求。

他们要我先写一本能够让他们确信交互设计有价值的书。

他们的想法很有吸引力，但是当时我没有把握写出他们所期望的书。

一个深夜，在可以眺望到佛罗伦萨的土黄色别墅的阳台上，我和Dare、Don进行着诚挚的交谈。

桌子上是几个喝空的葡萄酒瓶，还有一些面包、奶酪和橄榄。

天空中星光闪烁，萤火虫轻盈地飞过草坪，远处托斯卡纳区政府的圆顶古代建筑的灯光闪烁不定。

Dave再一次向我建议，我应该推迟编写如何进行交互设计的书，而应先“列举出一些使用了交互设计的商业案例”，让大家对交互设计信服。

我强烈地抗议道：“但是Dave，我不知道如何写你说的那本书。”

我掰着手指头一点一点地抛出了我的理由：“那意味着我不得不解释目前流行的开发过程是如何的混乱；我得解释软件公司是如何因低效率的开发而浪费金钱；我得解释客户们为什么这么容易‘移情别恋’；我得解释一个好的交互设计应如何解决这类问题。”

Dave打断我的话，轻松地说：“这些都可以是书中的某些章节呀，Alan。”

他的话让我无话可说。

我意识到我在老调重弹，Dave是对的。

一本“面向商务人员”的书比解释“如何做交互设计”的书有更迫切的市场需求，更合时宜。

Dave和Don最终让我相信，我确实可以写这样的书。

懂商务的技术专家/懂技术的商人 21世纪，能够成功的人是：理解商务的技术专家或懂技术的商人。

这本书是为他们而写的。

懂技术的商人知道，他自己的成功依赖于获取高质量信息的能力和有效地处理这些信息的能力。

另一方面，懂商务的技术专家是具备商业眼光的工程师或科学家。

他们具有敏锐的商业眼光，懂得信息的威力。

这两类新人将主宰当代商业。

我们可以将所有的商人分为两类：一类是精通高科技的人；另一类是正在走向破产的人。

作为商人的企业主管再也不能将处理信息的工作托付给专业人员。

商务就是处理信息！

企业主管需要在处理信息方面，而不是在制造系统方面，有别于人。

如果一个企业制造某种产品，那么这种产品中是否含有微型芯片，就是成功与否的关键所在。

如果提供某种服务，则胜算可能来自于这种服务是通过计算机化的工具提供的。

因此试图识别商务是否依赖于高技术与识别商务是否依赖于电话的商务一样，没有任何意义。

因为高科技已经渗透到了每一种商务活动，数字信息已成为我们每天工作中的脉搏。

有一句这样的话，“人总会犯错，要振兴只有靠电脑。”

低效率的机械系统可能会在加工的每个部件上浪费几个美分，而错误的信息处理则有可能毁掉一个企业。

基于软件的产品——制作这些产品的工程师们——对一个企业的影响力是不可估量的。

可悲的是，数字化工具非常难于学习、使用和理解，经常使我们达不到的目标。

我们因此而浪费金钱、时间和机会。

一位理解商务的技术专家或懂得技术的商人可能制作或使用软件产品，或者是两者兼而有之。

因此，拥有更好、更容易掌握、更容易使用的高技术产品，会为你个人及企业带来更大的利益。

## &lt;&lt;交互设计之路&gt;&gt;

其实，开发更好的产品并不需要更长的时间，也不需要更高的成本。

具有讽刺意味的是，它们本来不必如此困难，事实上却如此困难。

那是因为我们采用的软件产品开发过程是过时的，需要修正。

长期根植于我们头脑中的传统的错误观念使我们得不到好的产品。

本书将向我们展示如何要求并得到我们渴望以久的更好的产品。

本书的观点不难理解：在开始编制软件之前做好交互设计，我们就可以创建强大而令人愉悦的软件产品。

目前广泛流行的观点正好相反，我们已经不按流行的观点做了。

设计有交互的基于软件的产品是一种专业，像构造它们一样需要很多技能，并要付出相当的努力。

由于我已经选择了写一本基于商务案例的书，而不是写一本关于如何设计的技术类书籍，估计阅读此书的交互设计师们或许会失望，因此我请求你们能够予以谅解。

本书中仅用了少量篇幅提及交互设计方法的基础知识及其核心（主要在本书的第4篇）。

但这些篇幅足以证明：这样的设计方法确实存在；它对任何项目都适用；任何人都能看到它的好处，不管这个人的技术背景如何。

前言 最近，我在一家世界的技术公司见到一位高级主管，他的头衔是“易用性”副总裁。

他负责相当数量的大小软件产品。

他很精干，并在正统的“人机交互”领域有所建树。

他和他的公司一样，崇尚“可用性”（usability）方法，即通过单向隔离窗观察并测试人们对产品的反应。

但是他开始谈论设计而不是测试，谈论人物角色而不是用户。

他说，他的公司已经完全停止了开发后进行可用性测试的方法，现在采用的是事先设计的方法。

他进一步肯定，他们公司所有受过观察用户技能培训的人员正在接受人种学研究方面的培训。

这位主管和他所在公司态度的转变，象征着本书首次出版后短短的5年时间内，软件业界发生众多巨大的变化。

本书既是一本革命的宣言，也是一本提供规范的指南。

无数中层产品经理在阅读本书首版后给我发电子邮件，说明他们为自己部门的资深高级主管们购买本书的原因。

另一方面，软件制作者和大学将本书第4篇的三个章节：“交互设计”作为实施基于人物角色的目标导向设计的主要参考指南。

我深深感谢那些运用本书中描述的方法，帮助将可用性测试方法从实验室带到实际工作中，并将关注重心从测试转变到设计的所有主管、程序员、高级主管和可用性方面的实践者。

正是因为他们的努力，整个可用性专业领域发生了变化。

今天，我接触到的多数企业拥有一位或多位专业的交互设计人员。

他们对软件产品、服务的质量和行为有着越来越大的影响力。

得知这本书对他们的成功做出贡献，着实让我欣慰。

记得本书刚刚出版后的1999年，我在一个程序员论坛上做过一次主题演讲。

演讲的题目与本书的副题相同。

我开门见山地提出我的见解，“精神病人在运营管理着精神病院，而你们就是精神病人。”

当时，会场静得连一根针落地的声音也可以听得到。

我可以感觉到由2500名工程师组成的听众群体有强烈的抵触情绪。

在鸦雀无声的会场，我开始讲解本书的基本逻辑。

演讲结束时，这群“逻辑人”已经被我彻底说服，并给予我热烈的掌声。

令人惊讶的是，现在大多数程序员已经成为设计和设计者的忠实支持者。

他们认识到他们需要在软件的人性化方面得到帮助。

他们非常高兴终于得到有价值的指点。

他们已经认识到，任何有助于改进质量和让他们的程序易于被人们接受的实践，都不会威胁到他们的职业地位。

## &lt;&lt;交互设计之路&gt;&gt;

在过去，高层主管认为交互设计是编程问题，而将其职责授权给了程序员。程序员们也努力去解决交互设计方面的问题，虽然他们的技能、培训、想法、工作计划无法使他们成功。

本着分析问题的精神，本书真实地描述了一些由于程序员的原因而导致项目失败的故事。

有些程序员认为我在中伤，有些程序员则认为我谴责他们制造了劣质软件，因而他们攻击我的主张。

劣质软件的确是由他们制造的，但是我认为应该受到谴责的不是他们。

如果我给了他们那样的印象，那么我深感歉意。

除了个别人，我知道多数程序员在为了让最终用户感觉良好而勤奋工作，在为改进他们的程序质量而不断努力。

就像用户一样，程序员们只不过是混乱开发过程的另外一群受害者，时间太短，大量互相矛盾的命令，指导也严重不足。

如果任何程序员有了我在挑剔他们的印象，那么我也同样深感歉意。

软件制作过程的烦琐性——特别是编程工作的高昂成本和软件产品很低的交互性——根本不是技术问题。

那是过时的商业实践强加于软件编程行业的结果。

带着真情、好的愿望、上级管理层的祝福，程序员们更加努力地试图通过工程实践解决这个问题。

但是无论如何，工程方法是不能解决这些问题的。

程序员们已经感觉到他们的努力徒劳无为，工作中的挫折感也与日俱增。

在我最近的一次旅行中，我注意到在程序员群体中，消沉的气氛在扩散。

令人叹息的是，那些做得最好、最有经验的程序员们的遭遇却最差。

他们对自己的努力感到疲倦和无奈，因为他们知道自己的技能被无情地浪费了。

他们也许不能确切地知道事情是如何发生的，但是他们无法忽视事实。

事实上，很多优秀的程序员已经停止了编程，因为他们发现编程工作令人沮丧。

他们改行去做培训、促销、写作和咨询，因为这些工作让他们感觉到时间没有被浪费，工作效率也不低。

这是一个悲剧，它本来是可以完全避免的。

（虽然有些争议，不过开放源码运动成为这些沮丧的程序员们的天堂。

他们根据自己的标准编程，仅由他们的伙伴评价其工作的好坏，而不需要市场或管理人员的建议或干预。

）没有人给程序员们充裕的时间、明确的方向和恰当的设计，因而他们无法获得成功。

企业主管们不愚笨也不是恶魔，然而这种情况的确是现实状况，因此，他们应对此负责。

企业主管们仅仅是因为没有用恰当的工具武装自己，而无法解决信息时代所面临的复杂而独特的问题。

现在可能又让人觉得我在树敌。

是的，也只有在这样的時候，我的视野中出现的是商务人员的形象而不是程序员的形象。

我想再次强调的是，解决问题的途径是有破才有立。

我追寻的是解决问题之道，而不是替罪羊。

管理大师Peter Drucker在他92年生涯中的多数时间里，观察并指导着企业主管们。

他具有用独特的眼光看待问题的能力。

在最近一期CIO（首席信息官）杂志的采访中，他对20世纪五六十年代出现的对数字计算机盲目乐观的企业主管们进行了评价。

那些企业主管们曾想像计算机“会对如何运营企业产生巨大的影响”。

但是Drucker解释道，“事实并非如此。

没有几个主管提出‘我的工作需要什么信息？’

这样的问题。

”虽然数字计算机给予企业主管前所未有的大量数据，但是没有多少人问及这些数据是否能适当地引导企业走向成功。

## &lt;&lt;交互设计之路&gt;&gt;

企业的运营模式发生了巨大变化，但是企业管理却没能跟着变化。

他谴责生于商业社会的萌芽期，成熟于蒸汽机和钢铁时代，步履蹒跚地进入21世纪信息时代的会计系统。

Drucker进一步肯定，“我们最需要的是与外部有关的信息，但是计算机并没有向我们提供这些信息。”  
在20世纪的最后几年里，随着.com泡沫的膨胀，因特网上大量的文章兜售了关于新经济的想法。

学术界的人士宣称，在因特网上构建的商店里销售东西，与在那些用砖瓦水泥建筑的商店里销售东西是截然不同的，传统的“旧经济”已经名存实亡。

当然，现在几乎所有的新经济公司都已经死亡并消失，这让在背后支持这些公司的风险投资家感到震惊。

而那些学术界人士也已经适时地修正了他们的观点，声称所谓的新经济是空中楼阁。

最最新锐的主张告诉我们，必须继续维持原来那种古老的经济形式。

实际上，我相信我们真的是处于新经济时代了。

但是我不认为.com参与了新经济。

相反，.com是旧经济（制造业经济）的苟延残喘。

在计算机软件出现以前的工业时代，产品是通过加工有形物资制造出来的。

采矿、冶炼、采购、运输、加热、成型、焊接、喷涂等过程消耗的费用远大于其他开支。

财务人员将这些费用叫做“变动成本”，因为那些费用随着每一产品的生产数量而变化。

而“固定成本”（你可能猜得到）则基本上不随产品生产数量的变化而变化，如企业行政管理和建厂初期成本。

传统的商业管理源于工业时代的制造传统。

不幸的是，我们必须正视现在是信息时代这样一个现实。

在信息时代，产品不是由构成物质的原子制成的，而是由构成软件的比特（bit）的特定排列制成的。

而且，“比特经济”不遵循“原子经济”的规则。

有些基本经济原理同时适用于旧经济和新经济。

所有商业的目的都是创造可持续的利润。

达到这一点的合法途径只有一种：以高于制造成本或进货成本的价格销售一些商品或服务。

进一步细分的话，有两种增加利润的方法：或者降低成本，或者增加收入。

在旧经济时代，降低成本是最有成效的。

而在新经济时代，增加收入则更有成效。

当今世界最为关键和昂贵的产品基本上或完全由软件组成。

软件不需要原材料。

软件没有制造成本，也没有运输成本。

不需焊接，不需锤打，不需喷涂。

工业时代经济与信息时代经济的本质区别在于：在信息时代几乎没有变动成本，而在工业时代，变动成本是决定因素。

正是由于不存在变动成本，造就了现今的新经济。

企业支付给程序员的工资是固定成本还是变动成本？

程序员的一个工时肯定没有与一套软件产品的销售量产生直接的联系。

企业可以反复销售相同的代码。

投资于编程可以影响不同品种的上百万个产品，就像投资于工厂可以影响在工厂中生产的所有产品一样。

编写软件的成本不是变动成本，但也不是固定成本。

编写软件不同于建造工厂，它是企业实现销售收入的持续过程。

建造工厂的工匠虽然收费昂贵，但是一旦建筑完成，他们会去另外一个地方做同样的工作。

程序员比工匠和铁匠更加昂贵，但是他们不会去别处，因为他们的工作似乎永无休止。

有人认为编程是研究与开发，两者的确存在着相似之处。

## &lt;&lt;交互设计之路&gt;&gt;

可是，研究与开发是建立一项产品的理论上的可行性的思考和实验过程，其环境与实际生产环境有很大的不同。

用一个恰当的比喻来说，传统的会计原则将研究与开发的经费从产生销售收入的日常营运费用中分离出来。

而编写软件的经费，却无法准确合理地在传统的会计方法中列支。

现在，有人可能将此看做是文字游戏，但是事实上它对软件项目如何获得融资，如何管理，如何被资深主管看待（这是最重要的）具有巨大影响。

程序员创建软件，商务主管创造收入和利润中心。

程序员用产品质量衡量自己的成功，而商务主管用投资的收益性衡量自己的成功。

商务主管通过可以识别的，如固定成本、变动成本、企业的日常开支、研发费用等商业数学语言衡量收益性。

但是，不幸的是，不存在适合软件或编程业务的会计模式。

会计是基本的商业语言，上述商业术语对所有商业核算和商业沟通是极其重要的，以至于当代商务主管将其完全教条化。

他们将编程业务简单地看做可以归于现存某一会计类别的另一项企业费用。

在实践中，大多数商务主管简单地将编程当做制造工作 - 变动成本（只有在计算纳税额时，大多数企业才将编程归类于研发）。

这是最坏的选择，因为这个选择无情地导致了他们商业决策的失误。

工业时代的最大益处是产品可以大量生产，使得产品变得让大多数消费者买得起。

消费者的收益是可以买到从前买不起的东西。

手工产品只面向富裕阶层。

企业围绕着与变动成本（制造成本和流通成本）直接关联的价格进行竞争。

在信息时代，软件产品可以以非常低廉的价格获得。

任何人可以通过下载方式或其他媒体方便地获得软件，而几乎不花费任何代价，也无需多少人为参与。

请记住，企业可以通过增加收入或减少成本的方式增加利润。

也就是说，企业可以增加固定成本投资以提高产品质量和销售价格，或者降低变动成本，亦即降低制造成本。

在传统的制造经济中，降低成本是简单而有效的手段，因而被广泛采用。

当今天的商务主管们将编程与制造等同起来时，他们以为降低编程成本同样简单而有效。

很不幸，这些规则不再适用了。

相对而言，软件受变动成本的左右较小，努力减少其成本并不能带来很大的商业收益。

从财务角度看，程序员的薪资看似变动成本，但他们更像长期投资——固定成本。

降低编程成本并不像降低制造成本那样有效。

这样的效果更接近于给工人便宜的工具而不是压低工资。

那些为降低薪资而将编程工作委托给国外去做的企业根本不得要领。

进一步讲，在信息时代提高经济效益的惟一途径是通过提高质量，制造出更有吸引力的产品。

这是不可能通过削减花费在设计和编程上的资金投入来实现的。

事实上，必须在研究、思考、策划和设计阶段投入更多的实践和资金，才能让产品更好地满足客户的需求。

当然，这种方式是一种不为21世纪的商务人员所熟悉的思考方式。

他们需要增加在制造所有产品上的整体费用，而不是降低单个产品上的个体费用。

这是新经济的本质，也正是Peter Drucker所强调的。

发明新药特药的现代药业公司，具有与新软件经济类似的特征。

一粒药丸的实际生产成本微不足道，而一种新药特药的研制可能需要10年甚至更长的时间，投入上亿美元的资金。

如果能提供一种奇效的新药，那么收益将接近无限，但是如果将不成熟的药品推向市场，则风险无比

## &lt;&lt;交互设计之路&gt;&gt;

巨大。

药业公司的主管们都清楚地了解削减研究开发成本是不可行的商业战略。

创建软件更像发明药物而不像建造工厂。

工厂是企业可见的固定资产，工人可以相对自由地相互替换。

软件体现着一种无形的，极其复杂的思想模式，只有与编写它的程序员在一起的时候才有价值。

企业不可以像对待工厂工人那样对待程序员。

程序员们需要远超过于任何工厂工人的持续的关注和支持。

在创建软件的过程中，软件架构——程序的人性化设计部分——用户研究、用例定义、交互设计、窗体布置、行为描述——常常是在压缩成本时最先被考虑的部分。

设计过度的情况可能会发生，但是简化设计不会带来任何益处。

花费在软件架构上的每一美元和每个小时都会在编程阶段节省十余倍的金钱和时间。

更进一步地讲，如果在有竞争力的设计上投资，产品就会满足用户的期望，那样产品会创造更多的销售收入。

期望的满足会建立品牌，有利于提高价格，提高客户忠诚度，使产品具有更长、更强的生命周期。

这样做不仅会降低成本，而且非常有助于质量改进。

具有讽刺意味的是，在信息时代，增加利润的最好方法是增加支出。

不幸的是，大多数主管具有无法克制的削减对编程时间和资金投资的愿望。

他们运用的削减成本的方式已经过时。

他们没有看到，削减在编程上的投资对产品的远期质量、期望性、利润具有负面作用。

当然，只是简单地花更多的钱也不能保证产品质量有改观，而且如果没有智慧、分析、引导相伴，增加花费往往使事情变得更糟。

我的启蒙老师Dan Joaquin常说，“有多大的投入，就有多大的收获”这句古老的谚语也许应该说成“没有投入就没有收获”。

没有适当的计划就贸然前行的风险太大。

这里的秘笈是花费适量的金钱，但是做到这一点要求管理人员具有很强的软件工程管理方面的专业知识，还要求有一种向管理人员提供做出正确决策所需的深度信息和处理工具。

提供这样的工具是本书的目标。

.com热是由那些以削减变动成本为主要商业模式的企业传播开的。

虽然.com企业宣扬各种在线商务的好处，但它们的网站过于笨拙和无助，并不比驱车到购物中心让人舒适和方便。

.com的创立者们（还有媒体）被兴奋冲昏了头脑，以为可以用非常低的变动成本重建流通企业。

他们彻底而惊人壮观的失败无可争议地显示，信息时代的经济规则与工业时代的经济规则是不同的。

在旧经济时代，更低的变动成本意味着更广泛的流通和更低的流通成本。

这两项优点直接让消费者获益，它们是工业革命经济成功的基石。

在新经济时代，商业上的成功依赖于向消费者提供更新、更好的附加价值高的东西。

交易过程中每一部分的质量，从浏览页面到比较购物到多样性，必须让终端使用者明显感觉更好。

如果在细心地阅读了11个画面的内容后，最终还得打电话的话，那就远不如传统的购物方式方便。

反复三四次输入自己的名字、地址、信用卡信息，却发现不能从这个网站购买任何东西，还得去传统商店，只能使得整个在线销售变成不必要，令人失望。

今天，简单地降低供应商的成本是不能保证成功的。

当Pets.com公司在因特网上出售狗食物时，它并没有提供更好的东西，它没有提供比传统的宠物商店更好的用户购物体验；它没有提供更丰富的信息，没有显得更聪明，也没有显示更强的信心。

Pets.com公司提供的仅仅是更便宜的配送、存储和销售——所有这些对Pets.com公司而言都是变动成本。

这是忽视新经济基本原理的典型的工业时代的经营策略。

Pets.com公司的商业模式不是新经济的开端，而是旧经济的残喘。

我相信在因特网上可以销售任何东西并获利和成功。

## &lt;&lt;交互设计之路&gt;&gt;

实现的秘笈就是在线商店必须比传统商店提供更令购物者满意的可衡量的服务，价格只是满意内容的很小一部分。

实现的方法只有一种：必须整体地设计系统，才能尽可能地向终端用户提供最高的满意度。

像对待制造工程那样对待软件设计和制作的任何方面都会导致失败。

软件的设计和编程不应该成为传统的削减成本手段的目标。

在创建软件的过程中过度花费时间和金钱的可能性不是没有，但是因过少花费而导致的风险要大得多。

你可能熟知这样的风险，但是却很难让那些运营大企业的资深商业主管认同。

这些主管依旧使用在蒸汽机时代就流行的会计系统，虽然其企业的运营、决策、沟通、财务等各个方面都已完全依赖于软件。

他们使用的术语、概念根本不能反映现代商业独特的本质。

在现代商业中，商业工具和产品是由看不见的比特排列而成的，而不是成吨的钢铁修筑的铁路。

尽管企业聘用交互设计师并应用目标导向方法，但软件产品的质量并无太大改进。

更为甚者，编程成本仍然高居不下，软件的创建过程仍难以驾驭。

这是为什么？

在资深商务主管认识到软件问题不是技术问题，而是值得注意的商务问题之前，改变是不可避免的。

只有改变我们的创建过程和组织模式，问题才会得到解决。

很多企业不仅遵循着过时的财务模式，而且还遵循着不恰当的组织模式。

这样的模式是直接从学术界抄袭过来的。

在学术界，创建软件的行为与软件的计划 and 工程联系紧密。

这是研究的本质。

可悲的是，这样的典范被悄悄地完整无损地带到了并不属于它的商业世界。

除了软件，所有现代制造规范源自前工业时代，软件这种独特的媒介出现在后工业化时代，只有编程直接来自学术界。

在学术界，研究没有时间限制，学生劳力极其低廉，追求利润会遭到鄙视，失败的程序有可能被看做是非常成功的试验。

像微软、IBM、Oracle和其他领先的软件企业驻扎在“校园”并非巧合。

大学永远不需要挣钱，恪守交付时间，或创建满足期望，有用的产品。

所有非软件企业以研究开始，结束于大量生产和产品或服务的流通。

因为知道过早地生产不完美的产品会危害到收入和名声，他们会在研究和生产流通之间精心计划。

他们知道在计划上投资时间、思考、金钱产生的结果是速度和顺畅，在最终产品的受欢迎成度和获利方面将取得巨大收获。

在其他建造行业里，工程师做出建造规划，由工人来实施。

工程师不修筑桥梁，那是建筑工人的工作。

只有在软件行业，工程师的工作与创建产品的过程紧密相连；只有在软件行业，“建筑工人”参与如何生产产品的决策活动；只有在软件行业，这两项工作是同时而不是顺序地实施。

但是，创建软件的企业似乎没有察觉到这个特殊性。

在软件行业，工程和建筑如此密不可分地混杂在一起，以至于实践者和主管们也无法区分开来。

各种计划或被省略，或被推迟，而此时已为时已晚。

即使到了即将上市的代码进入了正常的开发周期，相当复杂的工程技术问题也经常性地被遗留下来。

这时在经费上已经陷入窘境，通常需要额外的资金支持才能继续前行。

系统架构的规划必须集成到工程计划的早期阶段。

事实上，它应该对早期阶段的工程有所促进，但是因为早期工程经常被推迟到建造过程开始之后，并且会受到生产编码（production code）的破坏，架构设计缺乏进入建造工程的切入点。

尽管事实上有些企业已经聘用了交互设计师，并且也重新培训他们的可用性测试人员来创建“人物角色”，但他们的工作对制作成本和最终产品质量的影响微乎其微。

## &lt;&lt;交互设计之路&gt;&gt;

解决办法在企业总裁和首席执行官手中。

如果这些企业主管们将解决办法下放到首席技术官或工程副总手中，就说明他们还不得要领。

那些有价值的主管是技术人员，但是问题却不是技术问题。

就像Drucker指出的那样，CEO们依赖的会计工具根本不代表其企业的真实状态。

就像是说，因为速度表的显示是准确的，所以汽车行走的方向是正确的。

在被数字技术主宰的商务世界里，这不再是真理。

在创建软件的过程中，错误地应用会计和组织模式而引发的最大问题之一是主管们不能认识到有多少编程资金被浪费。

一个精确的系统能够展示出每个美元都有一半是被花在了错误的地方，而且还需要额外的两三美元去解决由于不好的原始投资引起的问题。

在任何其他行业中，这样的统计数据都会成为改革的动力，但是在软件业，我们却处在极度无知的状态。

在过去的13年中，我的Cooper公司为几百家企业提供了咨询服务。

我们富有才华的设计师们为多数企业提供了可以极大地帮助他们产品蓝图，但是只有屈指可数的几家企业从中获益。

多数企业仅将交互设计和软件架构当做建议，而最后决策仍是由他们的程序员和工程师们做出的。

没有一家企业的CEO知晓，在工程师的工作间里发生着什么事情，因此他们毫无理由地压缩时间表。

程序员们总是在资源不足的环境中工作，主要是缺乏足够的时间编写好的程序，同时还缺乏决定编写什么的时间。

他们被迫拒绝建议，搪塞他们的经理以保护自己。

我相信有两种主管：一种主管本人就是工程师，另一种主管是惧怕工程师的人。

前者将交互设计视为建议，是因为他们的眼睛被利益所蒙蔽。

后者轻率决断，是因为他们不会说程序员的语言。

我指的不是Java或C#语言。

我的意思是商务人员和程序员缺乏共同的工具和目标。

“普通人”将非技术问题托付给“逻辑人”，并不知道如果由他们自己在主管层面应用适当的财务和组织模式，解决方法会更好。

企业面临着巨大的机会去突破障碍，去建立围绕着客户的满意度而不是软件组织企业的，围绕着角色而不是技术，围绕着利润而不是程序员的业务模式。

我殷切等待着受启发的主管出现，他们能够抓住这个机会，并通过提供软件业大胆而成功的例子而永远地改变创建软件的方式。

Alan 于加州-门洛帕克 2003年10月 inmates@cooper.com Cooper在软件设计方面的创新手法已经被认可为行业标准。

在Cooper打开面向商业的大门后的十年时间里，这家旧金山公司向诸如Abbott Laboratories、Align Technologies、Discover Financial Services、Dolby、爱立信、富士通、富士通Softtek、惠普、Informatica、IBM、罗技、Merck-Medco、微软、Overture、SAP、SHS Healthcare、Sony、Sun Microsystems、Toro公司、Varian，以及VISA公司提供了创新的、以使用者为焦点的解决方案。

Cooper团队提供他们自己发明和优化多年的目标导向交互设计工具。

这些工具包括在本书第一版首次披露的，叫做“角色”的用户建模和模拟用户的革命性技巧。

1994年，比尔·盖茨向因Alan Cooper发明了Visual Basic语言背后的可视化编程的概念而向他授予了视窗先锋奖（Windows Pioneer Award），1998年，Alan Cooper在软件开发者论坛上获得了声望很高的软件梦幻奖（Software Visionary Award）。

在1995年，Alan Cooper在他的首部畅销书“About Face：The Essentials of User Interface Design”一书中，将分类学引入了软件设计。

2003年，Alan Cooper和合著者Robert Reimann出版了此书的大幅修订版“About Face 2.0：The Essentials of Interactive Design”。

20年以来，Alan Cooper设计和开发的消费类软件产品包括，SuperProject，MicroPhone II for

## <<交互设计之路>>

Windows和微软公司的Visual Basic语言的视觉编程用户界面。

早在1976年，Alan Cooper创办了Structured Systems Group公司，一家被“Fire In the Valley”报道说制作出了“也许是为微型电脑的第一套商务软件”。

Alan Cooper还是一位为在电子产品开发过程中被遗忘的人——客户——的利益大声疾呼的人。

Alan Cooper是企业设计基金和美国设计中心的会员。

他曾任软件设计协会硅谷分会会长，是总部理事会成员。

Alan Cooper是软件设计和软件论坛的理事，还是SEF的Windows SIG——世界上最大的Windows开发者群体——的创始人。

他是关于用户操作界面和概念软件设计领域中的一位经常性的、有独到见解的、专注的行业演讲者和作者。

Alan Cooper的妻子，Susan Cooper，是Cooper公司的总裁和CEO。

他们有两个十多岁的男孩，Scott和Marty。

除了软件设计，Alan Cooper非常热爱通用航空、城市规划、架构、机动两轮车、烹饪、火车模型、飞盘高尔夫等。

请向inmates@cooper.com发邮件给他，或访问Cooper公司的网站www.cooper.com。

媒体关注与评论

Cooper的设计方法已经赢得了很多大牌公司的青睐，如Sun公司、可口可乐公司、康柏公司和Dow Jones。

——Fast Company杂志 Alan Cooper再一次展示他的魅力！  
他的书值得所有认为自己正在服务客户的技术公司反思。  
我们需要更多这样书，更多像Alan Cooper样的人。

——Don Norman《情感化设计》一书的作者 本书主题鲜明地叙述着，要创建能赢得市场的系统，领导者们应该知道些什么。

.....你将会发现，在你读过的书当中，这是一本最具有思想、最实用的书。

——Larry Keeley Doblin Group的总裁 Cooper的书是一纸宣言，它告诉人们，在我们需要花如此之多的时间与技术进行交互的时代里，如何提升生活质量。

——Peter Hirshberg Elemental Software的CEO

编辑推荐

本书是基于众多商务案例，讲述如何创建更好的、高客户忠诚度的软件产品和基于软件的高科技产品的书。

本书通过一些生动的实例，让人信服地讲述了由作者倡导的“目标导向”交互设计方法在解决“难用”问题方面的有效性，证实了只有改变现有观念，才能有效地在开发过程中引入交互设计，将产品的设计引向成功。

本书面向商务人员，也适合于所有参与软件产品和基于软件的高科技产品开发的专业人士，以及关心软件行业和高科技行业现状与发展的人士阅读。

#### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>