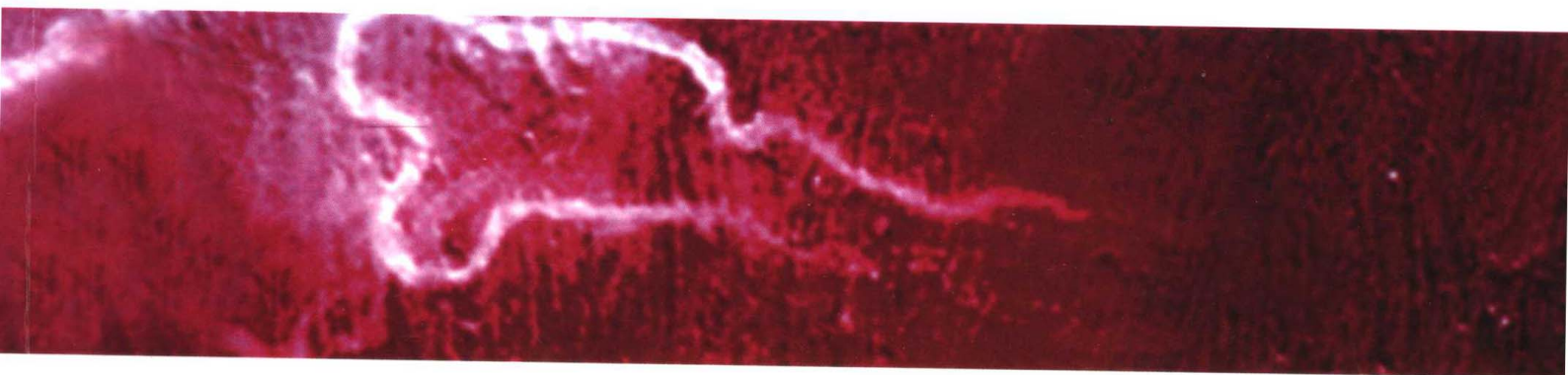




UI设计入门佳作

用户界面设计指南

User Interface Design for Mere Mortals



(美) Eric Butow 著
陈大炜 孙志超 译



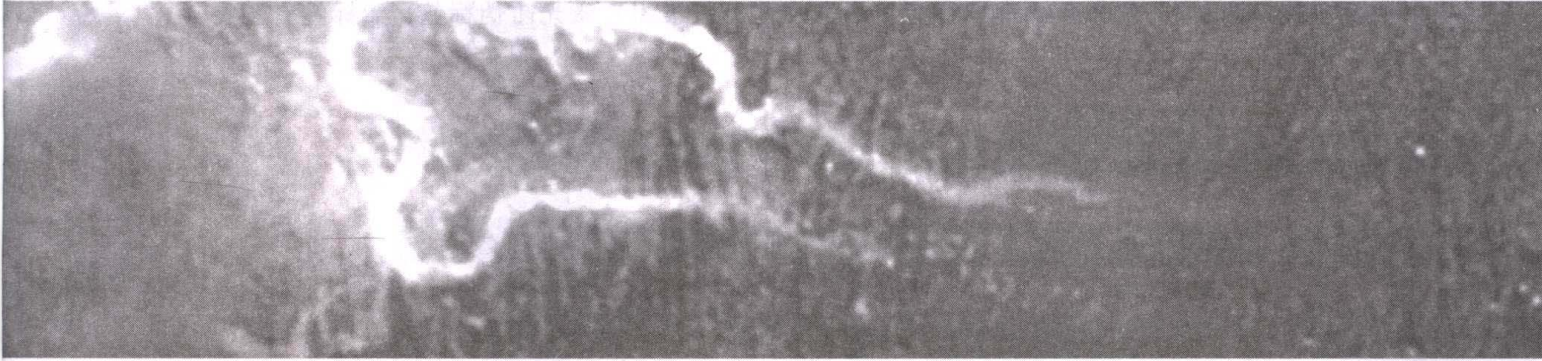
 机械工业出版社
China Machine Press

TP311.1/62

2008

用户界面设计指南

User Interface Design for Mere Mortals



(美) Eric Butow 著
陈大炜 孙志超 译



机械工业出版社
China Machine Press

本书第一次将业界领先的有关用户界面设计、可用性设计和测试的实践与观念融为一炉，全景式地展现了如何设计和实现能够让用户满意的用户界面全过程。全书共9章，包括历史回顾、概念与问题、制作商业案例、优良设计、用户行为、用户分析、设计用户界面、设计网站以及可用性等内容。阅读本书后，您将全面了解用户界面设计和可用性设计及测试领域的相关知识。

Simplified Chinese edition copyright © 2008 by Pearson Education Asia Limited and China Machine Press.

Original English language title: *User Interface Design for Mere Mortals* (0-321-44773-5) by Eric Butow, Copyright © 2007.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Sun Microsystems, Inc..

本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签，无标签者不得销售。

版权所有，侵权必究。

本书法律顾问 北京市展达律师事务所

本书版权登记号：图字：01-2008-0417

图书在版编目 (CIP) 数据

用户界面设计指南/(美)布托(Butow, E.)著;陈大炜,孙志超译. —北京:机械工业出版社, 2008. 1

书名原文: User Interface Design for Mere Mortals

ISBN 978-7-111-22947-6

I. 用… II. ①布… ②陈… ③孙… III. 用户界面 - 程序设计 IV. TP311.1

中国版本图书馆 CIP 数据核字(2007)第 186617 号

机械工业出版社(北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑:李南丰

北京牛山世兴印刷厂印刷·新华书店北京发行所发行

2008 年 1 月第 1 版第 1 次印刷

186mm × 240mm · 16.5 印张

定价:36.00 元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换
本社购书热线:(010)68326294

译者序

交互设计领域最近在互联网行业愈来愈热门，此方面的书籍国内已有许多，侧重各不相同。初次看到本书时，心想这应该是一本关于交互方面的入门级书籍，阅读之后证实了这一猜测，同时得出自己的评价——这是一本很好的入门书。它可以使初学者在很短时间内掌握交互设计方面的诸多概念和设计原则。诚然，它并不具备某些专门论著的深度，但能在较短的篇幅内让读者迅速拥有各领域的全面视角，也实属难得。本书作者长期从事计算机应用领域的教育，并著有多本畅销教材。正是这个原因，全书内容的条理性 and 语言的可读性都令人满意，以较为细腻的方式介绍每一部分的知识点和概念。

除此之外，本书具备其他同类书籍所没有的特色，即更具实用。作者在跨越几章节的范围内构造实用的案例。通过这种介绍方式，初学者可以掌握实际的运用方式以及如何综合运用书中讲解的内容。同时，每章所配备的复习题可以很好的巩固所学知识，也使得整本书的体例可作为教材使用。相信读者在读完本书之后可以成为交互设计方面的准专家，为更深入的学习研究打下了极好的基础。

本书的出版凝聚了多人的努力。全书由孙志超负责组织翻译和审校，由陈大炜和孙志超共同翻译。同时翻译期间得到了以下朋友的帮助，一并致谢：腾讯公司的帮助工程师周伟玉，网易公司的游戏设计师岳亮、邵赞、潘伟，网易游戏理论研究员辛洪红，康佳集团研究院的胡启祯。最后，还要感谢华章公司参与此书出版的所有工作人员所付出的努力。

限于译者水平，书中难免有疏漏不当之处，恳请读者批评指正。译者的 email 为 god-sun163@163.com。

译者

2007年10月

译者简介

陈大伟，华中科技大学计算机博士，主要研究方向为虚拟现实、可视化仿真和交互设计等。现为康佳集团研究院游戏设计部门经理，负责三维网络游戏关键技术和新一代人机交互系统的研发工作。

孙志超，网易互动娱乐游戏主设计师，参与多款游戏项目的设计，对交互领域有丰富的兴趣。

序 言

教养决定一切。桃子从前本是一种带苦味的杏；卷心菜也不过是经过培养的甘蓝罢了。

——马克·吐温

2005年，当时我正筹备着为加州州立大学萨克拉门托分校重新设计一项课程，即在线技术写作认证专业的可用性课程。我从包括图书网站和技术交流社区在内的海量网络资源中寻觅与可用性相关的书籍，为此花费了数小时。接着我走进厨房去取我的信用卡，好订购这一堆书。我的“行程”就此开始。

建立这一课程的过程的确算得上是一次很好的行程。在旅途中我获得了大量的知识，并在每一学期中将其与不同的学生们分享。在汇总整理教程的过程中，我想到为何没有一本书，不仅仅包括当前UI设计的理论与实践，而且包括用于确保UI设计有效的可用性设计与测试。我与我的文稿助理 Carole McClendon 讨论了一下关于将教程扩展成书的事宜，于是她向我介绍了 Addison-Wesley 出版社。

致 谢

任何一本书的写作都不可能仅依赖一人之力，成功的书籍更是许多优秀人员合作的结晶。首先要感谢的是我的文稿助理 Carole McClendon，她持续不断地帮助我写出自己和读者都感兴趣的书籍。每一天我都能从她身上感受到专业和专注。

接下来要感谢的是 Kristin Weinberger，Addison-Wesley 出版社的项目编辑。她临时接手了这本书，并在写作和编辑过程中不断完善它。没有她的持续帮助，你不可能看到手中拿着的这本书。

我还要感谢关注此书的编辑和审稿小组，他们给我提供了极富价值的反馈。小组中的 Mike Hernandez，他名副其实地履行了“面向初学者”这一标准。我需要特别感谢好朋友 Tony Barcellos，他从繁忙的日程安排(包括正在写自己的书)中抽出时间通览本书，并提出了许多有用且幽默的建议。最后还要感谢 Lawrence Smith、Ben Shneiderman、Rebecca Rior-dan、David Whiteman，他们是本书的技术编辑。

关于作者

Eric Butow 是 Butow 通信公司 (BCG) 的 CEO, 这家公司坐落于美国加州的罗斯维尔, 主要从事技术出版和网站设计。从 2000 年以来, Eric 著有或与人合著了 8 本著作, 包括《Master Visually Windows 2000 Server》《Teach Yourself Visually Windows 2000 Server》、《Font-Page 2002 Weekend Crash Course》、《C #: Your Visual Blueprint》、《Creating Web Pages Bible》、《Dreamweaver MX 2004 Savvy》、《The PDF Book for Microsoft Office》, 以及《Using Microsoft Windows Vista》(特别版)。同时, Eric 是很多计算机图书的技术编辑, 并为《SD Times》、《Intercom》等国际性刊物撰写文章。

Eric 还是一项在线教育课程的创立者和讲师。他为 Ed2Go 建立了 2 项 Windows XP 网络课程——一项针对 Windows XP 家庭版用户, 另一项则针对 Windows XP 专业版用户。Eric 还为视觉训练公司 (VTC) 设计了 RoboHelp 与 Windows Vista 的多媒体教程。此外, 他还为加州州立大学萨克拉门托分校创建并教授技术写作认证的继续教育课程。

当 Eric 不忙于写作、教学或自己公司的事务的时候, 通常会选择读书、和朋友聚会 (通常是在最近的星巴克或书店), 或者在丝雅拉山上的父母家中和北加州自家的葡萄庄园中的某处尽享家庭之乐。

前 言

过去 20 多年中个人计算机技术的发展，把用户界面设计放到了一个重要的位置。作为用户界面设计的一部分，许多企业都开始关注产品的可用性。可用性(usability)一词最早出现在 14 世纪，但随着计算机技术在 20 世纪 80 年代成为主流，可用性的定义也发生了变化，可用性不再仅仅意味着功能正常或者可以使用，通常还包含了用户对产品使用体验的最大化。

除了良好的用户界面和可用性设计以外，不管是现在还是将来，完善的可用性测试对于企业产品的成功也是至关重要的。只有了解管理人员、销售人员和目标用户之所想，才能为你所供职的公司，做出可获得公司上下热心支持的实现可用性测试的提案。

因此，无论什么职位，每个牵涉到可用性测试的人员，都应该在着眼于搭建企业长远利润平台的同时，具有为企业创造短期投资回报的商业智慧。

本书的预期读者

阅读本书不需要读者有任何用户界面设计方面的背景，如果你正准备涉足用户界面设计领域，或者准备开发自己的用户界面，那么本书将是你的最佳选择。显然，从设计之初就注重相关设计理论和经验的学习，要比从由于产品设计不良而得到的用户抱怨中学习强得多。

如果你已经开发过某些产品，而且正准备开始新的项目，那么你更应该阅读本书。也许你会下意识地感觉到用户的需求，但又不确定如何为用户将产品的可用性最大化。或者你正负责产品文档或培训，你需要了解用户之所需，他们不仅需要产品本身的信息，而且还需要产品文档或培训模块中所包含的信息。如果这与你的情况类似，那么请不要错过本书。

当然，本书也适合于在用户界面和可用性设计领域有实际经验的读者。尽管你也许会对本书中讨论的某些原理和理论耳熟能详，本书还是可以帮你做一次完整的梳理，而且很

有可能从本书中找到以前被你忽略了的金科玉律，甚至启发一些新的想法，你可以直接应用到实际的产品设计之中。

本书的写作目的

本书第一次将业界领先的有关用户界面设计、可用性设计和测试的实践与观念融为一体，全景式地展现了设计和实现能够让用户满意的用户界面全过程。

本书首先从用户界面的演变开始谈起，你可以清楚地了解用户界面如何一步一步发展到今天这种形态。接着，本书深入地讨论了如何为产品设计用户界面和可用性测试，这里的产品可以是诸如打印机等计算机硬件，也可以是一款应用程序软件，或者是一个网站。

通过阅读本书，你可以全面了解用户界面设计和可用性设计及测试领域的基础知识。需要指出的是，本书是可用性和用户界面设计领域入门级书籍，如果你有兴趣对本书中出现和讨论的某些理论进行更深更专业的研究，笔者建议你阅读附录 B“推荐读物”中列出的相关著作。

本书的阅读方法

笔者强烈建议读者能够按照章节顺序从头至尾地阅读本书，这样做不仅可以让你始终保持在本书的上下文语境中，而且还能够让你对整个用户界面设计过程有一个完整清晰地认识。

当然，如果你阅读本书是为了回顾某些特定的主题，你也完全可以直接翻阅感兴趣的章节。实际上，本书每章的行文也是尽可能地独立于其他各章。

本书的章节组织

下面是你将要阅读的本书中每章的内容概要：

第 1 章“历史回顾”，对图形用户界面和网页设计的发展历程进行了综述，同时也对不同的图形用户界面给用户带来的不同观感进行了讨论。

第 2 章“概念与问题”，涵盖计算机、用户界面模型、可用性和用户分析等方面的概念及趋势，同时也涉及易用性等问题。

X

第3章“制作商业计划”，主要关注可盈利性计划，以及理解合作方需求，从而可以制定出该计划。

第4章“优良设计”，讨论好的用户设计目标、用户和设计师面对的限制条件，以及如何利用合适的原型和故事板进行快速检验以发现潜在的设计问题。

第5章“用户行为”，涵盖的内容包括用户行为心理、用户如何运用自己的知识来完成任务，以及用户如何构建真实世界的概念模型等问题。

第6章“用户分析”，讨论用户心理模型以及用户是怎样与可用性钟形曲线相吻合，从而理解用户的目标和任务。

第7章“设计用户界面”，讨论了以虚拟用户体验为基础的界面设计框架的实现，通过这种方法可以让你更好地确认目标用户及其期望的用户界面。

第8章“设计网站界面”，理清了网站界面设计和一般图形用户界面应用程序设计之间的差异以及网页设计中的一些常见的误区，并给出了若干网页设计准则。

第9章“可用性”，讨论了可用性测试的设计以及测试过程本身，通过可用性测试你可以从目标用户中得到有关界面设计的有益反馈。

附录A“复习题答案”，给出了第1章~第9章后复习题的参考答案。

附录B“推荐读物”，为有意进一步深入研究用户界面和可用性设计的读者列出一些必读的相关著作。

“术语表”逐条对本书中出现的各种专业词汇和短语进行了简明扼要的解释。

目 录

译者序

译者简介

序言

致谢

关于作者

前言

第 1 章 历史回顾	1
1.1 图形用户界面的历史	1
1.1.1 施乐的 Alto	2
1.1.2 苹果的 Macintosh	3
1.1.3 微软的 Windows	4
1.1.4 Linux	6
1.2 网页设计的历史	8
1.2.1 因特网的诞生	8
1.2.2 Mosaic	8
1.2.3 Netscape 的革命	9
1.2.4 Internet Explorer 及其对设计产生的影响	9
1.3 外观风格的差别	10
1.3.1 Windows GUI	10
1.3.2 Mac OS GUI	10
1.3.3 Linux GUI	12
1.3.4 网页	13

1.3.5 Java 和其他网站程序	13
1.4 小结	14
复习题	14
第2章 概念与问题	17
2.1 计算机术语	18
2.1.1 图形用户界面	18
2.1.2 操作系统	18
2.1.3 图形用户界面构成	19
2.1.4 因特网	22
2.1.5 万维网	24
2.2 用户界面模型	26
2.2.1 批处理用户界面	26
2.2.2 命令行用户界面	27
2.2.3 文本用户界面	28
2.2.4 图形用户界面	29
2.2.5 网站界面	30
2.2.6 与用户界面模型集成的界面接口	33
2.2.7 其他用户界面	34
2.3 界面设计改进和不足	35
2.3.1 Windows Vista	35
2.3.2 Mac OS X	36
2.3.3 Linux	37
2.3.4 网站设计改进	38
2.3.5 仍未解决的问题	39
2.4 未来的展望	39
2.4.1 Windows Vienna	40
2.4.2 Mac OS	40
2.4.3 网页浏览器及其对界面设计的影响	40
2.4.4 未来界面	41

2.5 可用性术语	43
2.5.1 可用性工程师	44
2.5.2 可用性科学家	44
2.5.3 用户体验专家	44
2.5.4 可用性设计分类	45
2.6 用户分析术语	45
2.6.1 目标导向设计过程	45
2.6.2 测试方法	46
2.7 用户分析趋势	46
2.8 可及性问题	47
2.8.1 508 条款可及性	48
2.8.2 网站可及性	48
2.8.3 操作系统可及性	50
2.9 小结	51
复习题	52
第3章 制作商业计划	55
3.1 项目关系人之间的分歧	56
3.1.1 用户期望	57
3.1.2 产品工程师与设计师的期望	57
3.1.3 销售与市场人员的期望	57
3.1.4 经理的期望	58
3.2 制订商业计划框架	58
3.3 优良设计的益处	59
3.3.1 降低生产成本	60
3.3.2 降低客户支持费用	61
3.3.3 更有效的保留客户	61
3.4 如何盈利	62
3.5 证实投资回报率	63
3.5.1 投资回报率具体说明	64

3.5.2 计算金额	64
3.6 可用性工程生命周期	65
3.6.1 阶段1: 需求分析	66
3.6.2 阶段2: 设计、测试和开发	67
3.6.3 阶段3: 安装与反馈	68
3.6.4 不会结束的过程	69
3.7 案例研究: Mike 的自行车店	69
3.8 小结	74
复习题	75
第4章 优良设计	77
4.1 优良设计目标	78
4.2 设计师与用户对立吗	79
4.2.1 用户限制	80
4.2.2 设计师限制	80
4.2.3 解决矛盾	81
4.3 纸上原型设计和故事板	81
4.3.1 纸上原型设计的概念	82
4.3.2 克服怀疑态度	83
4.3.3 纸上原型设计优点	84
4.3.4 纸上原型设计缺点	86
4.4 优良的文档设计	87
4.5 为什么要关注良好的设计	93
4.6 案例研究: 创建一个纸上原型设计测试	94
4.7 小结	99
复习题	100
第5章 用户行为	101
5.1 用户行为心理学	102

5.1.1	人格类型	103
5.1.2	四种基本人格	105
5.1.3	人类行为的七个阶段	108
5.2	知识：大脑应付外界	109
5.3	任务结构	111
5.4	有意识行为和潜意识行为	111
5.5	将困难任务变成简单任务	112
5.6	创建概念模型	113
5.7	案例研究：进行访谈以建立概念模型	115
5.8	小结	117
	复习题	117
第6章	用户分析	119
6.1	用户心智模型	120
6.1.1	产生后果	121
6.1.2	实现模型与心智模型	122
6.2	经验钟形曲线	123
6.3	了解用户目标	126
6.4	用户和任务分析	129
6.4.1	构建人物角色	130
6.4.2	观察用户行为	134
6.4.3	人物角色评价	135
6.5	案例研究：创建主要人物角色	137
6.6	小结	140
	复习题	141
第7章	设计用户界面	143
7.1	设计基于人物角色的交互框架	144
7.1.1	现实世界的需求	146

7.1.2	定义框架	146
7.2	交互设计	150
7.2.1	应用设计规则	150
7.2.2	原则	150
7.2.3	模式	151
7.3	软件姿态	151
7.4	界面行为	155
7.4.1	使用鼠标指针	155
7.4.2	窗口行为	157
7.5	帮助用户寻找信息	162
7.5.1	视觉暗示	162
7.5.2	声音提示	162
7.5.3	弹出式信息	163
7.5.4	搜索引擎	164
7.6	与用户交流	165
7.6.1	使功能易于寻找	165
7.6.2	在线帮助	165
7.6.3	助手和向导	166
7.7	精炼形式与行为	166
7.8	案例研究: 精炼纸上原型测试	167
7.9	小结	169
	复习题	170
第8章	设计网站界面	173
8.1	网站与图形用户界面之间的比较	174
8.1.1	图形用户界面规则	174
8.1.2	网站界面规则	175
8.1.3	基于互联网的应用程序	176
8.2	网站界面设计陷阱	176
8.2.1	用法	177

8.2.2	设计	177
8.2.3	可及性	179
8.3	网站姿态	180
8.4	为什么需要网站工程	187
8.4.1	后端编程	187
8.4.2	表单处理	187
8.4.3	数据库	188
8.5	网站标准	189
8.5.1	颜色和文字	189
8.5.2	图片	190
8.5.3	网站导航	190
8.5.4	面包屑链接	191
8.6	四个规则	191
8.6.1	保持简洁	191
8.6.2	保持一致	192
8.6.3	保持最新	192
8.6.4	保持三次点击原则	192
8.7	何时打破规则	192
8.7.1	打破图形用户界面规则	192
8.7.2	打破网站设计规则	193
8.8	案例研究：界面导航特性	193
8.9	小结	195
	复习题	196
第9章	可用性	197
9.1	为可用性测试选择合适的方法	198
9.1.1	观察、倾听和交流	198
9.1.2	其他用户交互方法	199
9.2	定义可用性测试	200
9.2.1	目标和关注点	201

9.2.2	选择测试参与者	201
9.2.3	选择、组织并创建测试实例	202
9.2.4	确定测量可用性的方法	203
9.2.5	准备测试材料	203
9.3	执行可用性测试	204
9.3.1	执行预备性测试	205
9.3.2	锻炼观察技巧	205
9.3.3	锻炼访谈技巧	207
9.3.4	持续关系	209
9.3.5	关心测试参与者	209
9.3.6	进行真实测试	210
9.4	分析并陈述可用性测试结果	211
9.4.1	分析并陈述数据	211
9.4.2	报告	213
9.4.3	陈述	215
9.4.4	准备简报	215
9.4.5	更改产品和过程	216
9.5	案例研究：实现纸上原型测试	217
9.6	小结	221
	复习题	221
附录 A	复习题答案	223
附录 B	推荐读物	237
	术语表	239



第 1 章

历史回顾

过去不会重复，但它的韵律仍存。

——马克·吐温

本章内容提要

- 图形用户界面的历史
- 网页设计的历史
- 外观风格的差别

在讨论用户界面 (GUI) 设计之前，本章将告诉大家 GUI 是怎样开始的，并且它是怎样导致网页出现的。你将了解到，随着网页变得流行，网页的设计与可用性也变得越来越重要。你也将了解到关于网页设计规则的历史。最后，我们将学习 GUI 和网页界面两者在外观风格上的不同。

1.1 图形用户界面的历史

Jeremy Reimer 在 2005 年一篇名为《A History of the GUI》(GUI 之历史) 的文章 (<http://arstechnica.com/articles/paedia/gui.ars>) 中追寻了 GUI 的历史，最早在 20 世纪 30

年代，一个名为 Vannevar Bush 的工程师制造了 memex 存储器，它有两个触摸显示屏、一个键盘和一个扫描仪，这样用户可以查找知识，类似于现在使用网页上的链接。

Bush 介绍 memex 存储器及 20 世纪 30 年代以来其他用户计算机的著作，启发了一位年轻的工程师——Douglas Engelbart (Reimer, 2005)，他供职于斯坦福研究中心，并负责美国空军的基金项目——开发最早的 GUI。1968 年，Engelbart 向 1000 多名计算机专家展示了他们团队的研究成果。

这套在线系统(NLS)的 GUI 展示了几乎包含我们今天所知道的众多硬件和软件特征，如键盘、鼠标、网络计算机和网络用户协作、多媒体视窗、全屏桌面编辑器、电子邮件、超文本链接和可视化会议(Reimer, 2005)。与我们今天使用的系统相比，这套系统相当原始，但这次展示表现出计算机可用性的巨大突破，所以撼动了每个参观者。

展览的成果也受到了出席该展览的施乐(Xerox)公司工程师们所关注，因为他们已经看到了 Engelbart 系统的未来——一个无纸化信息传递时代的到来。对于一个因为在 1959 年发明复印机而在行业中占据主导地位的公司——施乐来说，无纸化未来就意味着施乐的死亡。所以施乐的管理层认为必须掌握这项新的技术，于是公司于 1970 年在加州的 Palo Alto 成立了 PARC，其使命是研制第一代商业化 GUI。

这一部分，我们的讨论从 20 世纪 70 年代施乐 PARC 的研究开始，一直到 20 世纪 90 年代微软的视窗系统出现。Windows 系统进一步强化了 GUI 作为用户界面标准的地位。如果有人想在计算机领域谋份职业，那就必须掌握 GUI 的相关知识。这部分我们还会讨论现在使用的其他几种 GUI 操作系统——苹果的 OS 系统、Linux 系统。

1.1.1 施乐的 Alto

由 PARC 在 1973 年研制的施乐 Alto 是第一次用于个人计算机的 GUI。它体现了 PARC 在计算机界面技术上的重大突破，如鼠标和 GUI 的首次应用。它的显示器仅有黑白两色，主机的大小相当于一台小型的电冰箱。但这是第一台以桌面形式显示的计算机，直到 4 年后，Apple、Tandy 与 Commodore 这几家公司才分别推出个人计算机。(然而，与 1977 年出现的个人计算机不同的是，Alto 计算机单元是一个庞大得你不得不把它放在你的桌子下面，而桌子上面仅仅能够放显示器、键盘和鼠标——这就像今天大多数的桌面计

计算机，尽管今天的计算机比 Alto 看起来要小上许多。)

遗憾的是，施乐不懂得如何推销 Alto 和 1981 年推出的后继产品 Star。这使得 PARC 的这一创举被苹果借鉴，并于 80 年代开发出了全新的计算机——Lisa。

1.1.2 苹果的 Macintosh

20 世纪 70 年代后期，苹果积极向施乐学习，使得苹果 Lisa 计算机于 1983 年诞生。Lisa 计算机虽然失败了，但其姊妹项目 Macintosh 于 1984 年迅速推出，并培养了一小部分但却很忠诚的用户。Macintosh，我们一般简称为 Mac；其销售额最终超过了苹果计算机最初的那几个系列——苹果 II 及后续产品、苹果 II 加强版、苹果 IIe 型、苹果 IIc 型和苹果 II GS 型(这里也该有商用苹果 III 的名字，但它是百分之百失败的产品，而且很快沦为笑柄)。

尽管市场占有率比较低，但 Mac 有一批忠实的追随者。本书写作之时，苹果的增长很大程度上来自其 iPod 音乐播放器的贡献。MacOS 是一个受到长期好评的高级操作系统，其优秀程度甚至超过微软的视窗系统。许多年来苹果计算机被作为教学计算机的首选，并且被许多学生所喜爱。我与 GUI 发生的首次接触就是在 Mac 系统上。

但 Windows 操作系统已经克服了苹果计算机的一些缺点，所以苹果也将 Windows 的很多设置拿来为我所用，如屏幕下端的导航条，可以使用户很快找到程序和文件。

Mac OS 的第十个版本以及最新的版本，称为 Mac OS X(X 表示罗马数字 10)，相比于 Mac OS 之前的版本是一个根本性的转变。实际上，当 Mac OS X 在 2001 年发布的时候，这完全不同于苹果之前的操作系统，不仅仅改变了以 UNIX 操作系统为依据的系统核心部分，并且为全新的界面 Aqua 揭开了面纱。Aqua 具有明快的颜色及半透明的特征，例如标题栏和“果冻”似的按钮，这看起来与果冻杯是那么的相似，请看图 1-1。

Steve Jobs 以 CEO 的身份于 1997 年回归苹果，从那以后他接手的每件事都发生了巨大改变，Aqua 也是这样，它不仅仅影响了网页设计，而且还影响到了微软视窗操作系统的设计。实际上，新版本的视窗操作系统 Vista，许多设计样式(如半透明的标题栏)与 Aqua 如出一辙。苹果已经把 Mac OS X 和 Aqua 捆绑在一起做为未来的主流产品版本来发布，如版本 10.1、10.2、10.3 等。例如，降低了 Aqua 皮肤的亮度，并且添加了窗口样式的导航条。苹果将以“大猫”的名字发布新版本的 Mac OS X，就像文中写到

的一样，下个改进版本的 Mac OS X，将称为 Leopard(美洲豹)。这个版本将会在 2007 年早些时候与 Windows Vista 操作系统同一时间发布。Mac OS X 当前的版本是 Tiger(老虎)，见图 1-2。

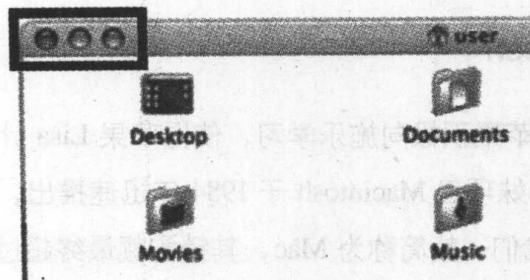


图 1-1 Mac OS X 视窗系统上果冻杯按钮

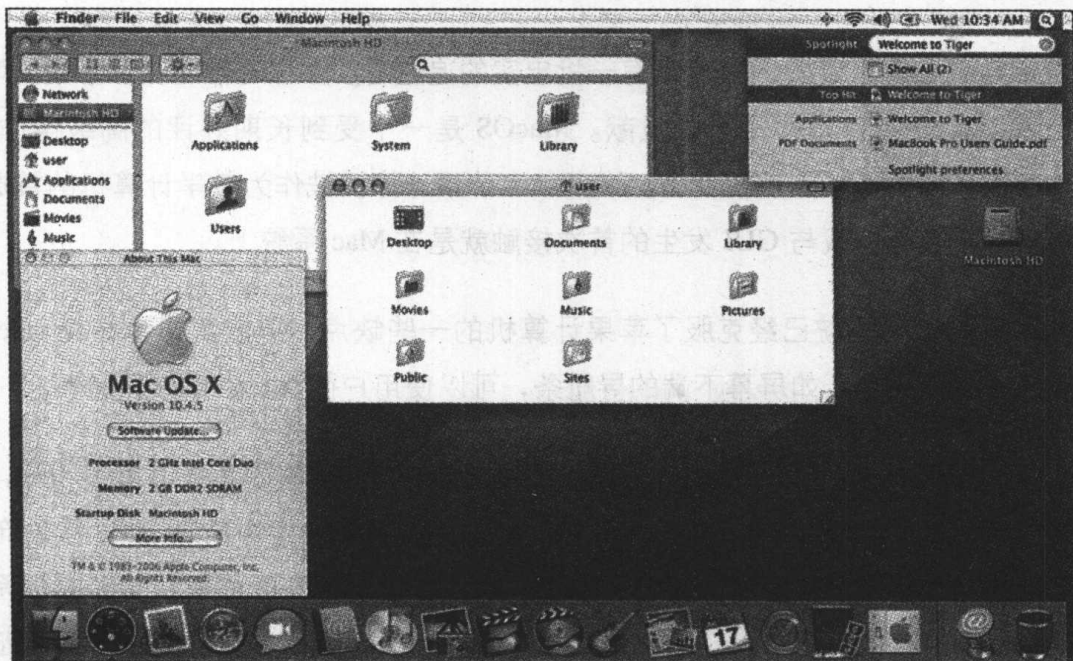


图 1-2 The Mac OS X Tiger 的 Aqua 界面

1.1.3 微软的 Windows

VisiCalc 电子表格软件(后来最受欢迎的商用软件)的厂家曾宣称要为 IBM 的个人计算机研制 VisiOn 的 GUI，对此，微软于 1983 年发布了 Windows。VisiOn 并未出现，但 1985 年微软却又推出了 Windows 1.0 版本。

遗憾的是，硬件性能远远达不到 GUI 的需求，但是当英特尔公司发布 80286 芯片和 80386 芯片时，微软仍然继续改进 Windows 2.0 以充分利用它们。尽管如此，GUI 仍然比较粗糙，Windows 2.0 与之前的版本相比较，在外观风格上没有什么太大的差别。

从 1990 年开始，众多的商业机构都使用配备了 80386 芯片的计算机，80386 (连同它的运算处理器，可选的英特尔 80387) 有足够的处理能力来处理 GUI。除此之外，视频处理技术的进步带来了更大的显示分辨率以及比过去更多的可用色彩。在这种情况下，微软在 1990 年 5 月发布了视窗系统的升级版本 Windows 3.0。

Windows 3.0 有一个修复得非常整洁的界面，并且它支持多个任务同时进行，这就意味着用户们在同一时间能够完成更多任务。很快，计算机类的媒体都给予 Windows 3.0 以好评，并且很多用户，像我一样能够获得免费的拷贝，在新 386 计算机上使用并且发现 Windows 也运行得很好。Windows 集成了新的计算机功能，并获得了良好的口碑。用户很快就放弃了 IBM 1981 年在其个人计算机上引入的、以文本为基础的 DOS 系统 (磁盘操作系统)。Windows 迅速成为 IBM 个人计算机和其他兼容机上 GUI 的标准。微软随后连续发布了 2 个版本——1992 年发布的 Windows 3.1 和 1993 年基于工作组的 Windows 3.11，后者是具有网络功能的 Windows 版本。

微软在 1995 年彻底修复了 Windows 并发布 Windows 95。这一版本引入了 Windows 任务栏、开始按钮和可视化主题 (这是最根本的创意或者说主题)，这些特征一直保留在以后的版本中，包括 Windows Vista。在 Windows 95 发布之后微软随后又发布了 3 个版本的 Windows，两个版本的 Windows 98 和 Windows Me，后一个版本被称为消费者自定义的 Windows 版本。

同一时期，微软从 1993 年开始就紧锣密鼓地开发着“专业”版本的 Windows，这一版本称为 Windows NT。Windows NT 从头开始建立了一个新的内核，不同于 Windows 这一建立在 DOS 内核视窗图形界面基础上的消费者版本。Windows NT 是一个为网络而准备的 Windows 版本，Windows NT 和后来的版本 Windows 2000，取代了 Novell NetWare 成为网络软件的首选。

2001 年，微软发布了建立在以 NT 为内核基础上的视窗操作系统 Windows XP。新的操作系统包含改进过的用户界面并且提供了两个版本，为个人消费者提供的家庭版和为商

业用户提供的专业版。它们主要的不同在于专业版能够使用更多的网络工具，而微软希望家庭版在小型的点对点网络中使用。

在2006年底，微软发布了第一个花费5年时间研制的Windows新版本Windows Vista，如图1-3所示。Vista最大的改变就是拥有了一个能提供类似于Aqua半透明效果(被称为Aero)的新界面。为了实现一些功能(如改变显示器设置)在界面上做了一些改进，增强了地址栏以便于观察与其他窗口相关的位置，增加了即时搜索的功能，并增加了更多安全特性。

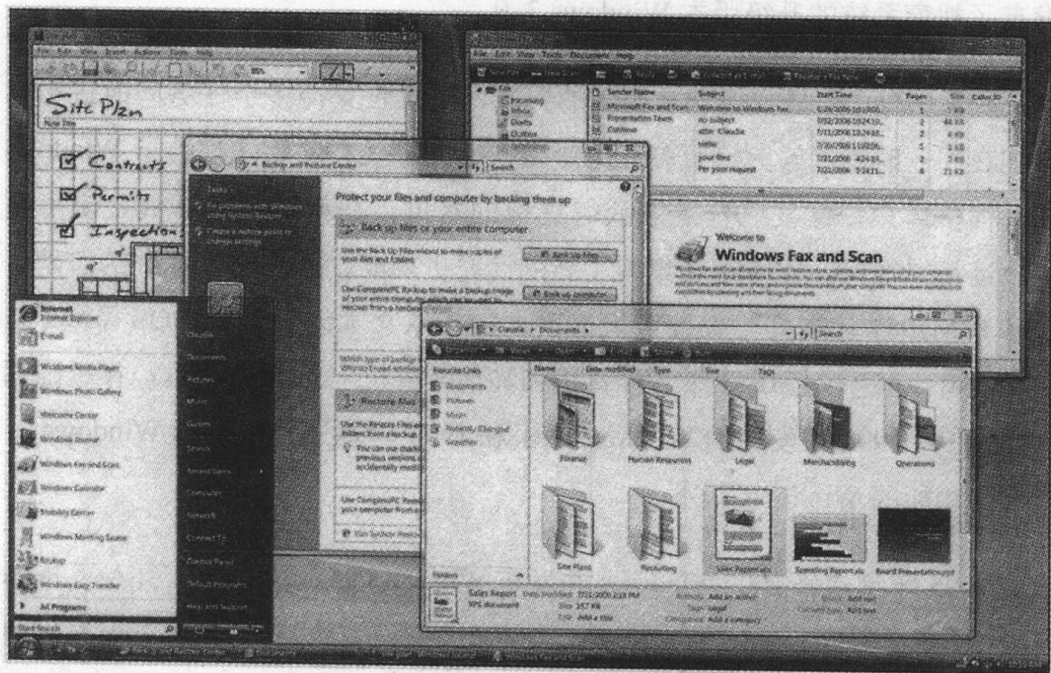


图 1-3 Windows Vista 桌面

1.1.4 Linux

1983年，麻省理工学院的一名程序员Richard Stallman发起了免费软件运动，开发了一款集中UNIX诸多优点的GNU操作系统。GNU操作系统的目标就是开发供人使用的免费软件。Stallman定义的免费，包括能够任意使用、修改甚至重构。1985年，Stallman与人合作建立了“免费软件基金”从而扩展了这项运动，这一基金的目的就是开发供所有人免费使用的软件。

直到1991年，GNU操作系统唯一缺乏的部分就是内核——所有组件中最重要的部分。没有内核，操作系统就不能管理系统资源和计算机软硬件之间的信息互换。芬兰的一

名程序员 Linus Torvalds，于 1991 年用称为 Linux 的系统填补了免费内核 GNU 的空白。Linux 系统在 1991 年免费发布，作为一款开源软件，这意味着其他程序员能使用这个操作系统并完善代码。1993 年，Linux 发布两年之后，第一个发行版本在网上供人们下载使用。这一发行版本是可以在计算机上安装的 Linux 安装包。

由 GNU 和 Linux 结合的操作系统，也就是简化版的 Linux，因其稳定性而变得流行。这个操作系统很快成为在计算机服务器上使用的普遍选择；今天，大约有 25% 的计算机服务器运行 Linux 系统。然而，运行该系统的服务器不能支持一些商业软件，它们很难被安装。个别公司提供了一些满足这一需求的兼容版本。Red Hat 公司提供的产品是最流行的版本。Red Hat 系统并非仅仅提供安装软件的打包文件，还提供了由两名墨西哥程序员 Miguel de Icaza 和 Federico Mena 开发，并持续地由世界各地的程序员维护和升级的一个免费 GUI 环境——GNOME。

到了 20 世纪 90 年代末，Linux 已经有两种 GUI 可以使用：KDE 和 GNOME，后者如图 1-4 所示。这两种 GUI 跟 Windows 类似，且被认为与 Windows XP 实现效果相当。对于

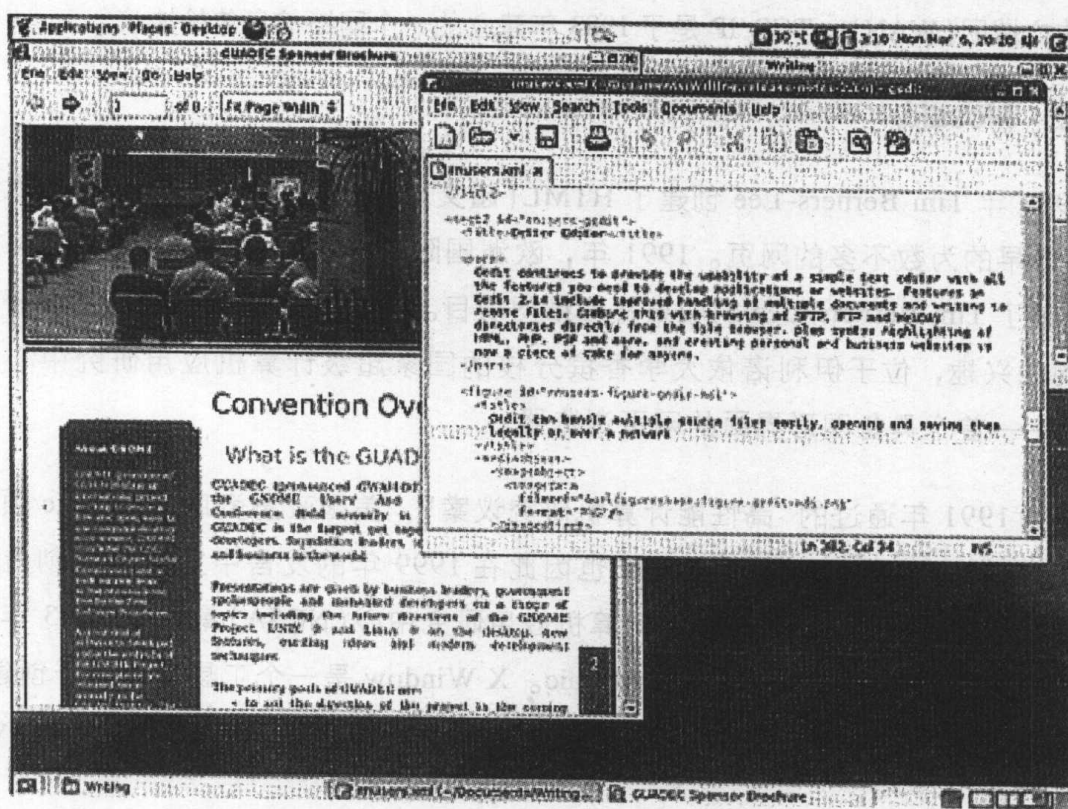


图 1-4 最流行 Linux 版本 Red Hat Linux 中自带的 GNOME 桌面

那些想要一个稳定的操作系统的公司来说，他们不想用微软的操作系统，但苹果系统又过于昂贵，所以具有两种 GUI 的 Linux 系统成了很多公司的首选。IDC 的市场调研预测，到 2008 年 Linux 将会占据操作系统市场 7% 的份额。

1.2 网页设计的历史

网页设计对用户界面设计方面有重要的影响，主要是因为许多计算机用户会在网上冲浪。因此，网上宣传是为一款产品聚集人气的最有效的途径。

1.2.1 因特网的诞生

因特网的产生起源于 Sputnik 这一颗前苏联于 1957 年发射的首枚人造地球卫星。此后，美国国防部立即采取措施使美国重回这一科技前沿。这一努力带来了 ARPA 网的诞生，这是一个由国防部计算机组建的网络，并于 1969 年投入使用 (ARPA 代表的意思是高级研究计划署)。根据 ARPA 网的工作原理，美国国家科学会创立了第一个使用 TCP/IP 协议的广域网 (WAN)。TCP/IP 是于 1983 年建立的一个网络信息传输协议。

1.2.2 Mosaic

1989 年 Tim Berners-Lee 创建了 HTML (超文本标记语言)、HTTP (超文本传输协议) 和最早的为数不多的网页。1991 年，欧洲国际原子能机构 (其缩写 CERN 广为人知) 公布了 Tim Berners-Lee 的成果及万维网项目。为了激发拥有因特网的科研组织对万维网的兴趣，位于伊利诺依大学香槟分校的国家超级计算机应用研究中心发布了 Mosaic——首个具备图形界面的网页浏览器。

根据 1991 年通过的“高性能计算机研究议案”，美国政府资助了 Mosaic 项目。这项法案由 Senator Al Gore 提出 (Gore 也因此 1999 年的发言中宣称自己创造了因特网，这几乎成了笑柄)。国家超级计算机应用研究中心 (NCSA) 最早于 1993 年 4 月发布了能在 X Window 系统上运行的 Mosaic。X Window 是一个工具集，用于创建 UNIX 操作系统下运行的 GUI。到了 1993 年底，Mosaic 实现了在苹果 Mac 和 Windows 系统下运行。

1.2.3 Netscape 的革命

Mosaic 浏览器大大激发了除科研组织之外的人对万维网的兴趣。随着人们对因特网的兴趣日益提升, Mosaic 浏览器的开发者们组建了自己的公司——Netscape, 并开发 Mosaic 浏览器的继承产品 Netscape Navigator。随着因特网变得流行以及美国各地区的因特网服务提供商(ISP)诞生, Netscape Navigator 迅速取得成功, 使得 Netscape 公司的股票价格飙升至 1996 年上市以来的最高点。其产生的另一个影响是 Netscape 所建立的网页浏览器设计标准至今仍在广泛使用。

1.2.4 Internet Explorer 及其对设计产生的影响

微软注意到 Netscape 和万维网的兴起, 并决定顺应潮流。1995 年微软发布了它首个版本的网页浏览器(IE), 但是在 1998 年之前都没有真正超越 Netscape Navigator 的市场地位。之后, 联邦法院裁定微软将 IE 捆绑在 Windows 以获得对于 Netscape Navigator 的不平等优势这一行为不合法, 但法院判决宣布的时间对于 Netscape 而言已经太迟。1999 年, 美国在线(如今的 AOL)收购了 Netscape, 如今 Netscape 仅拥有浏览器市场的一小部分。

Internet Explorer 很不可思议地对用户界面设计产生了影响, 这是因为 2 个原因: 90% 的网络用户在使用它, 同时微软没有对其持续升级。自 2001 年 Internet Explorer6 发布以来, 直到 2006 年都没有发布 7 的版本。在 2001 年到 2006 年这 5 年时间之内, 其他公司和包括 Netscape, Opera 和 Firefox 在内的其他浏览器得到了发展空间。这些“可选”的浏览器中的每一款产品都在界面上做出了改进, 如给浏览的网页标上标签, 以便你能轻而易举地从一个网页转向另外一个网页。IE7 现在融合了许多“可选”浏览器的创新之处。读者可以在本书的第 2 章“概念与问题”中了解到更多有关 IE 在网页设计方面所产生影响的内容。

不同浏览器之间的 HTML 代码存在细微的区别, 同时微软用于增强网站功能的 ASP 技术不被 Firefox 和 Opera 浏览器支持。如果随时间发展网页浏览器的市场激烈变得更加激烈, 网页技术可能会变得更好, 并且轮流推动网页设计向前发展。

1.3 外观风格的差别

至今我们仍然在使用 GUI 操作系统所建立的桌面模拟 (metaphor) 体系, 即在屏幕上摆放物件类似于在现实桌面上摆放物品。尽管所有的主流 GUI 操作系统通过尽量减少用户需要记忆的任务数量以保留桌面模拟的优势, 界面设计者仍然因为要遵循操作系统界面标准而受到限制。此外, 这一部分所讨论的各 GUI 之间有着众多细小的差别。

因此在设计用户界面的时候, 需要理解你所要进行设计的操作系统环境中界面以何种方式运行。如果打算制作一个在不同系统平台 (如 Windows 和 Mac OS) 下工作的软件界面, 你需要认识到每个平台的差异和相似之处。

为网页做设计的时候甚至有着更多引起混淆的东西, 因为网页设计有着一系列不同的约束规则和参数, 例如 Java 及其相关的网页程序, 这是因为 Java 界面也有着不同于其他界面的外观。

1.3.1 Windows GUI

微软引入的一些界面特性在 Windows 95 及后续版本 (包括 Windows Vista) 中是独创的, 尽管其他一些 GUI 操作系统全部或部分地复制了这些 Windows 的特征。它们的特征如下:

- 任务栏, 以按钮的形式显示所有打开的程序并显示“开始”按钮, 可以让你打开程序或者查看服务状态。用户显示程序或文档只需要点击这些按钮。
- “开始”按钮, 打开一个菜单, 在此菜单中可以访问所有 Windows 中的程序, 包括 Windows 自身的功能, 例如查看安装在计算机上的所有打印机的列表。
- 工具栏, 在任务栏中增添一些图标从而可以在任务栏中迅速打开程序。例如, 快速启动工具栏为工具栏增添了图标。你可以最小化所有程序回到桌面以添加程序的快捷方式, 如 Internet Explorer。

1.3.2 Mac OS GUI

Mac OS 的 GUI 是其他一些 GUI 试图追随并模仿的对象, 并且 Mac OS 的 GUI 在第 10

版中经历了革命性的改变，这一版本称为 Mac OS X (X 代表 10)。新版本带来的 Aqua 界面在外观风格以及功能方面有着如下的特征：

- 导航栏 (Dock) 是 Mac OS X 中新的特征。导航栏类似于任务栏，能够在其中添加程序和文档以方便地引用。和 Windows 的任务栏一样，导航栏出现在屏幕的底部，不在 Mac OS 的菜单栏中。
- 和 Mac OS 以前的版本一样，菜单栏会出现在屏幕的顶部。打开新的窗口时菜单栏的选项会发生变化。例如，打开微软 Word 时，菜单栏中有一系列 Word 中应用的菜单选项。时钟出现在菜单栏的右边如同在 Windows 任务栏中的情形。菜单栏如同“开始”菜单一样能够快速访问 Mac OS X 的相关功能，在 Mac OS X 的桌面中有更多功能可以使用。
- 导航栏并非像 Windows 任务栏一样提供特别的工具条，但导航栏将不同的图标聚合分组，并用线条隔开。如图 1-5 所示，导航栏的左边部分有一系列程序图标，最小化的窗口和文档则在右边。类似于 Windows 的任务栏，你可以拖曳程序图标到导航栏中并创建程序的快捷方式。

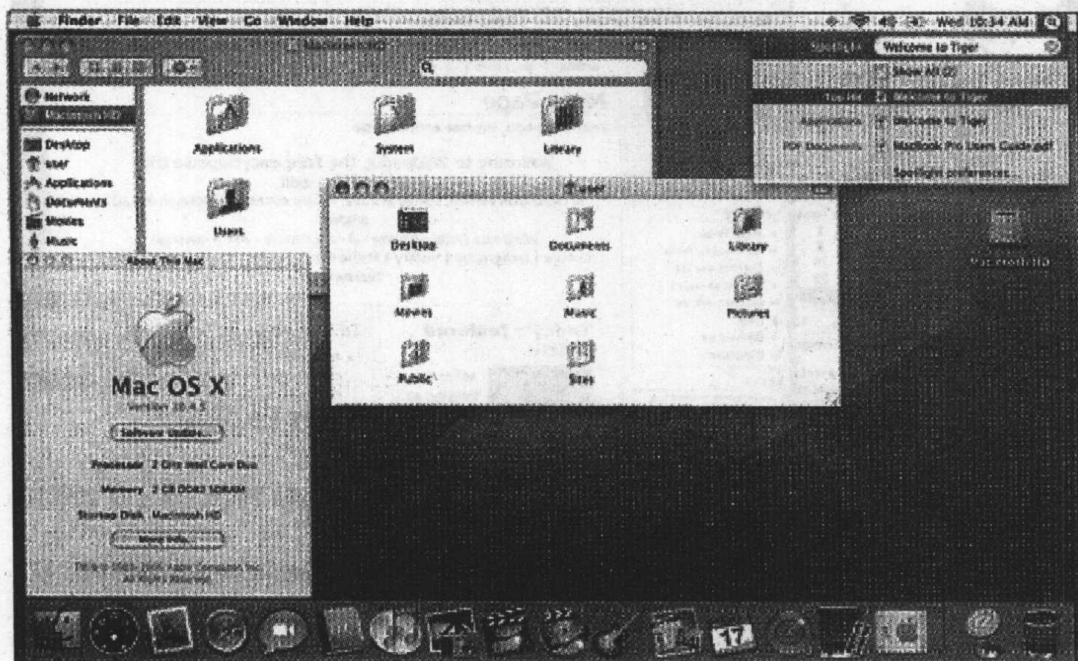


图 1-5 Mac OS X 的桌面，Dock 和打开的窗口

1.3.3 Linux GUI

GNOME 和 KDE 是 Linux 操作系统中的两个高级 GUI。Linux 能够更广泛地被接受为桌面操作系统的关键就在于 GUI，所以人们不会奇怪它既像 Windows 又像其他一些系统。

例如，GNOME 和 KDE 都包含了任务栏、工具栏、标题栏和设置成与 Windows 一样的操作按钮，并且开始按钮与 Windows 的开始按钮如出一辙。尽管如此，GNOME 和 KDE 也有细微的差别。例如 KDE 包含一个多页面的任务栏以方便用户点击任务栏中的页面数，可以看到按钮与页面相关。GNOME 通常被认为是最完整和最“成熟”的 GUI，但是 KDE 被更多 Linux 系统的新用户所接受，因为它看起来更像 Windows，如图 1-6 所示。

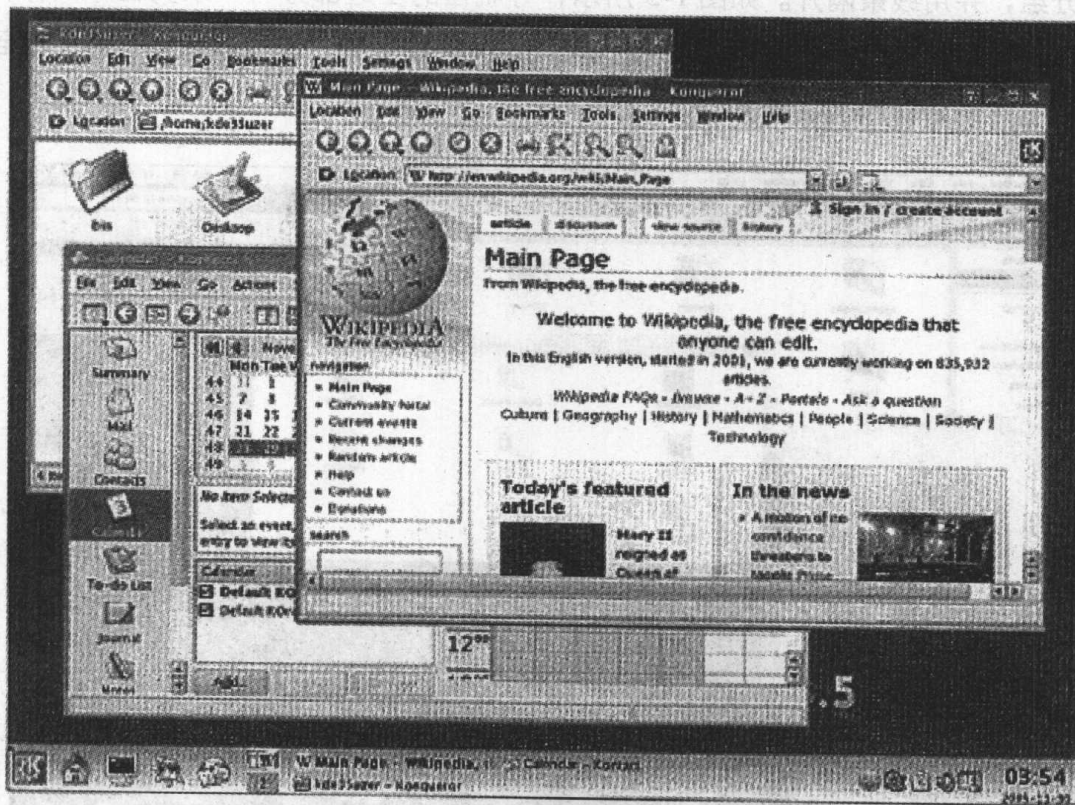


图 1-6 KDE 桌面

1.3.4 网页

就像第2章中将讨论的内容那样,网页可以具备各种不同的界面,但是它们仍然在设计规则和程序限制所规定的范围之内。虽然如此,把网站设计得比较像其他软件的界面也并非难事,就像 MSN.com 网站上那些表现成标签按钮的超文本链接,如图 1-7 所示。



图 1-7 MSN.com 网站置于页顶的标签按钮,可以提供熟悉的用户体验

Flash 这一由 Adobe 公司的开发的产品,已成为网页动画设计的默认标准,并正在开始淡化桌面和网页软件的分。Flash 的应用之一正是创建看上去像桌面的网页软件,并能使界面与富因特网应用程序(RIAS)融合在一起,并由终端用户的计算机完成运行。程序设计师们也用了 AJAX 这一包含了 JavaScript 和 XML 的网页技术,来创建交互式的网站。

1.3.5 Java 和其他网站程序

网站上运行的程序通常通过程序语言来反映外观风格的标准设定。Java 就是一个明显

的例子。Java 程序总是拥有自己的一套按钮、操作窗口和字体的外观风格设定。

1.4 小结

在这章中，我们讨论的内容包括什么是 GUI、当前主流 GUI 操作系统及其发展历史、GUI 的组成部分及其功能。了解网页设计的历史是很重要的，因为通过它能够理解我们今天了解和使用的 GUI 是如何从早先的 GUI 设计中吸取经验的。

除此之外，本章还回顾了网页设计的历史和网页与浏览器之间错综复杂的联系。读者能了解到因特网的发源和第一个浏览器 Mosaic 的诞生，以及在浏览器的开发过程中如何诞生出第一个商业浏览器 Netscape Navigator。同时，读者能够了解到微软开发 Internet Explorer 的过程，它不仅仅成为具有明显优势的浏览器，并且因为微软暂停对其开发而使得其他竞争者不断创新。

在本章的最后讨论了包括微软 Windows、Mac OS X 和 Linux 在内的各主流 GUI 之间外观风格的区别。然后介绍了网页相关的内容，如网站的界面很容易设计得像软件等。最后，我们提到了网页开发所用的基本语言，特别是 Java。

以上对 GUI 和网页历史的简要回顾，为接下来继续在第 2 章中探讨用户界面提供了必要的背景知识。

复习题

在继续第 2 章之前，现在是时候温习一些我们学到的内容了。尝试回答以下问题，并查阅附录 A“复习题答案”来检查你的答案是否正确。

1. 为什么说了解用户界面的历史沿革非常重要？
2. 谁开发出了第一个图形用户界面？
3. 为什么 NLS 如此重要？
4. 第一个带有图形用户界面的个人计算机是什么？

5. 为什么 Linux 这么流行?
6. 第一个带有图形用户界面的网页浏览器是什么?
7. 为什么 IE 浏览器对设计影响如此巨大?
8. 作为一个用户界面设计者, 为什么需要知道用户界面的工作原理?
9. 为什么需要了解不同图形用户界面之间的不同点?
10. 为什么需要知道图形用户界面与网站界面之间的区别?



概念与问题

“如果不敲碎壳，你就得不到核。”

——Persian Proverb

本章内容提要

- 计算机术语
- 用户界面模型
- 设计进步与不足
- 未来的展望
- 可用性术语
- 用户分析术语
- 用户分析趋势
- 可及性问题

用户与计算机进行交互的方式是多种多样的。除了我们常见的图形用户界面(也许您正在使用)，还包括了诸如触摸屏等各类日趋流行的交互界面形式。

在深入讨论用户界面模型之前，需要读者能够很好地理解贯穿本书的每个专业术语。

因为理解这些术语及其背后的含义，对于读者是否能够顺利学习用户界面设计而言是至关重要的。尽管您可能比较熟悉本书中出现的某些术语，但就算是再重新温习一遍对读者来说也是不无裨益的。当然，对于那些不大熟悉的专业术语是需要读者认真学习的，笔者会对本书中出现的所有专业术语进行逐条的阐释。当然，这些术语解释也会以术语表的形式出现在本书的末尾供您参考。

本章会继续讨论用户界面模型的历史，从而使读者能够更好地理解您是如何与计算机进行交互的。尽管当今的用户界面应用了大量界面设计新技术和新方法，但还是存在一些不足之处，本章最后会讨论到。无论如何，用户界面会有一个令人兴奋的前景，本章会讨论目前在这个领域的研究热点，以及使用这些新型用户界面会给用户带来怎样的不同感受。

通读完用户界面模型的部分，读者可以学到有关用户设计和分析方面的专业术语，以及相关的设计趋势。这些趋势之一是将用户界面设计得尽可能便于绝大多数用户的使用，读者不仅可以在本章学到有关用户界面可及性方面的问题，还能够了解一些在着手用户界面设计之前必须掌握的一些基本设计惯例和规则。

2.1 计算机术语

本书中会直接使用大量计算机术语来表达和定义用户界面设计，所以读者务必知道并理解这些术语的含义。

2.1.1 图形用户界面

图形用户界面 (GUI) 是一套通过操作图形元素和文本与计算机进行交互的系统。常见的图形元素包括窗口、按钮、菜单和图标等。因为所有基于 GUI 的应用程序都是用类似的界面元素，非常便于用户上手，所以人们会很容易将 GUI 元素与真实世界相对应。

2.1.2 操作系统

当今世界上的计算机基本上都使用三个主要的 GUI 操作系统，虽然也有其他诸如

基于命令行界面 (CLI) 等类型的操作系统存在, 但以下三种操作系统绝对是目前的主流:

- **Windows**——微软的 Windows 是事实上的标准图形界面操作系统, 因为近 90% 的计算机用户都使用 Windows 操作系统。
- **Mac OS**——Mac OS 是苹果电脑公司专门为苹果机 (Macintosh) 开发的操作系统。与 Windows 和 Linux 不同的是, Mac OS 是苹果机专属的, 也就是说只能在苹果机硬件平台上使用 Mac OS 操作系统。目前, 大约有 4% 的计算机用户使用这种操作系统。
- **Linux**——Linux 是一种可以免费发布并且开放源代码的操作系统, 当然您也可以购买诸如 Red Hat 和 Novell 等由专业公司维护的特殊版本。Linux 操作系统本身并没有内嵌用户图形界面, 但是在这些需要付费的特殊版本中大多都会集成某种流行的 Linux 图形界面, 比如知名的 GNOME 和 KDE 等。在桌面电脑领域, Linux 的份额大概占 3%, 而在服务器领域, 因为人们普遍认为 Linux 操作系统要比 Windows 更稳定, 所以大约 25% 的网络服务器都使用 Linux 操作系统。

2.1.3 图形用户界面构成

无论何种操作系统的用户图形界面总有相同的标准部分。这些部分通常被称为界面构件 (widget), 用户通过这些构件与计算机进行交互从而完成特定的任务。

1. 窗口

窗口通常是指屏幕上的一块矩形区域, 窗口界面中可以用于显示各类输出信息, 也可以允许用户输入信息, 如图 2-1 所示。不同的窗口可以完成不同的功能, 比如, 一个包含了网页浏览器的窗口可以让我们访问网页, 而一个包含了字处理应用程序的窗口则可以让我们编辑、保存并打印一封邮件。典型的窗口通常会包含一个边框, 窗口可以被最小化 (即隐藏)、最大化 (即充满整个桌面)、在屏幕内移动, 或被关闭。

常见的窗口界面可以分为两大类, 即单文档界面 (SDI) 和多文档界面 (MDI)。对于单文档界面而言, 用户可以通过运行相同的应用程序打开多个窗口, 比如可以打开多个网页浏览器窗口浏览不同的网站。对于多文档界面来说, 一个父窗口中可以包含多个子窗口, 比如在文字处理软件中同时编辑多份文档。本书的第 7 章“设计用户界面”, 会更加详细地

讨论这个单文档界面和多文档界面的问题。

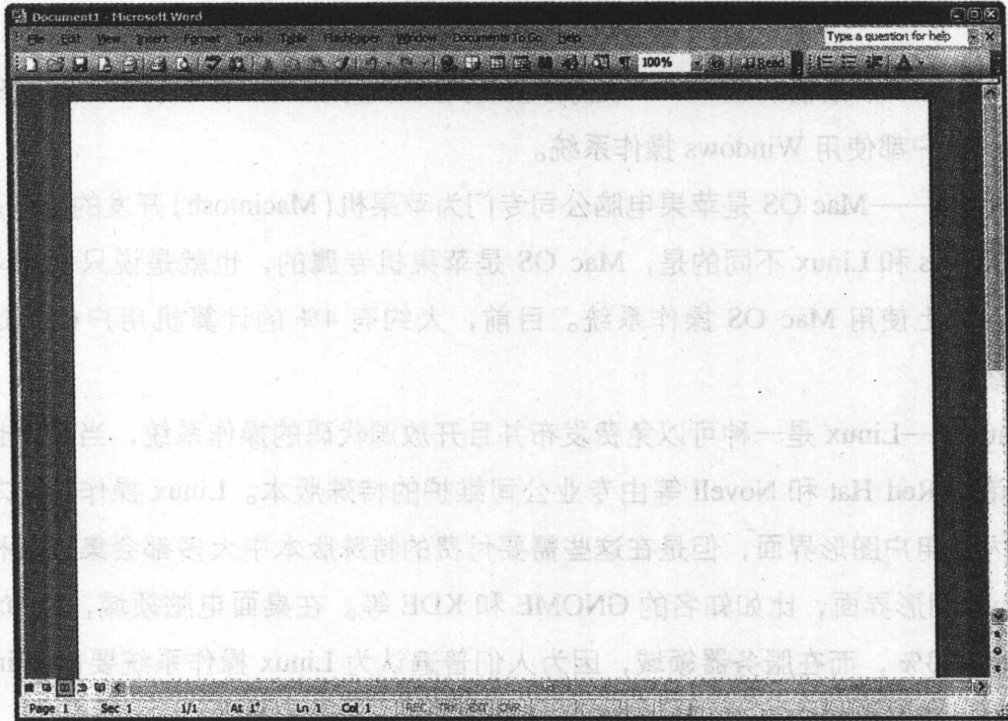


图 2-1 一个窗口的实例

2. 按钮

按钮常用来触发某个命令，如图 2-2 所示。按钮可以有各种不同的形态表现出来，比如点击 **OK** 按钮可以确认某项操作，而点击 **Print** 按钮则可以把文档或者网页输出到打印机打印出来。一般而言，按钮通常都是矩形的，而且当用户按下和松开按钮时，系统会采用类似真实世界中按钮被按下和弹起的效果来表现这两种状态。

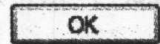


图 2-2 一个按钮的实例

3. 菜单

菜单常常表现为一系列命令的列表，用户可以通过选择菜单项向应用程序发出特定的指令。最常见的菜单类型为下拉菜单，一旦用户点击菜单名称，菜单项便会向下拉出供用户选择。现代操作系统也常使用弹出的菜单，当用户在窗口或屏幕上的某个位置单击时（通常用鼠标右键单击），一个带有选项的菜单就会在单击处弹出。

通过各种菜单，用户可以方便地跟应用程序或者操作系统进行交互。图 2-3 显示了 Microsoft Excel 用户界面中的文件菜单，提供了大量的命令选项，诸如打印当前正在编辑

的表单等。

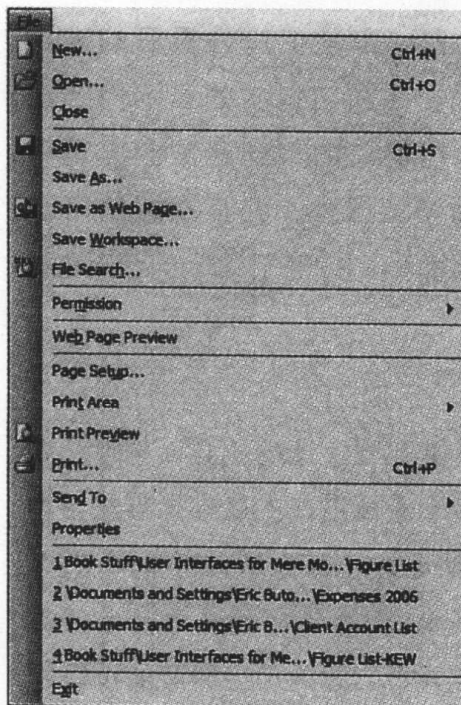


图 2-3 一个菜单的实例

4. 工具条

工具条是一种可以包含按钮和其他各类界面构件的用户界面，可以表现为横状工具条、竖状工具条，或者是以工具箱的形式出现。当用户点击工具条上的图标或者按钮时，应用程序就会启动相应的响应过程。工具条一般紧贴于窗口菜单栏下，或者作为一个单独的浮动窗口出现。例如，打开 Microsoft Excel 软件，用户就会看到菜单栏下面包含了各种按钮的工具条；如图 2-4 所示，通过这些按钮用户可以进行诸如打开一个新的表单等类似的操作。



图 2-4 一个工具条的实例

5. 滚动条

使用滚动条可以方便地在当前窗口中察看超出该窗口面积的连续文本或者图像，如图 2-5 所示。滚动条通常表现为一个位于窗口右边的长条矩形，有时也会出现在窗口底部，

滚动条的滚动块儿可以被上下或者左右拖动。

用户上下拖动滚动块儿时，当前窗口内显示的内容也会做相应的移动。类似的，如果窗口中的显示内容超出了当前窗口的宽度，那么在窗口的下部通常会出现一个水平的滚动条。除了滚动块儿以外，在滚动条的两端常常会包含对应的箭头按钮，可以用于进行小范围的调整。

6. 任务栏

任务栏是用来打开和监控应用程序运行的界面构件，如图 2-6 所示，默认情况下是位于桌面窗口的底部，当然用户也可以将其移动到桌面的其他地方。自从 Windows 95 发布以来，任务栏已经成为 Windows 标准界面构件之一。任务栏包含了标识着当前已打开但并不可见的那些窗口的按钮和图标，但并不是任务栏上的所有界面构件都是可交互的，其中的一些仅仅为了显示当前某些应用程序和服务的运行信息。



图 2-5 一个滚动条的实例

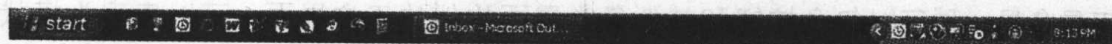


图 2-6 一个任务栏的实例

2.1.4 因特网

因特网是一个面向公众开放的全球计算机网络系统。考虑到因特网对用户界面设计产生了深远的影响，所以本书中也会涉及一些因特网术语。

1. 万维网

万维网通常被简称为 WWW，是因特网上的一种信息共享空间，公众可以通过网页浏览器进行浏览。需要指出的是，万维网本身并不是因特网，它仅是一种人们可用从中发布、获取信息并能与他人共享的信息空间。

2. 网页浏览器

如图 2-7 所示，网页浏览器是一个用来显示网页信息，同时又能与用户进行交互的软

件应用程序，比如用户在网页上填好一个表格然后点击 **Submit** 按钮就能将所输入的信息发送到架设网站的公司。目前市面上有很多可以选择的网页浏览器，根据维基百科网页 (WikipediaWeb) 上的资料显示，在 2006 年中期有 83% ~ 86% 的计算机用户都使用 IE 浏览器在 Windows 和 Mac OS 上浏览网页。

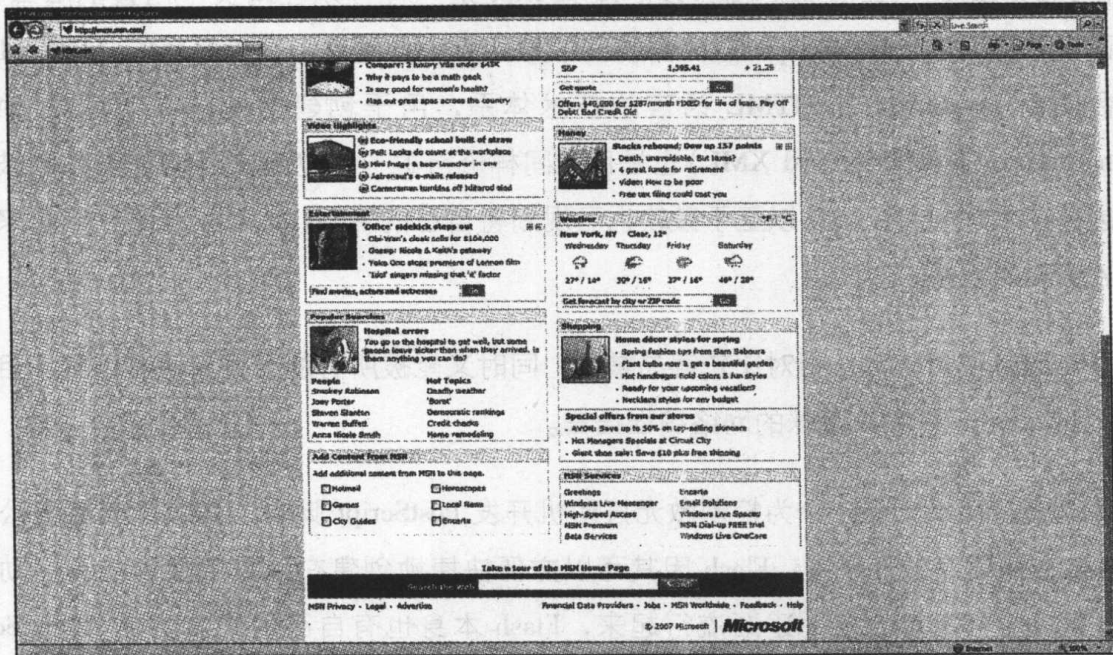


图 2-7 一个网页浏览器的实例——MSN.com

3. URL

统一资源定位符 (Uniform Resource Locator)，也就是我们所熟知的网址是一串标识网页、网站或其他因特网资源地址的字符。一个 URL 是由 4 个不同的部分按照一定的顺序组织起来的，这样才能找到正确的网页。比如，微软的 URL 地址是 `http://www.microsoft.com`。

4. HTML, Java 和其他语言

超文本标识语言 (HTML) 是在因特网上创建网页和网站的标准语言。然而，作为一种简单的标识语言，HTML 在功能上是比较受限制的。随着因特网的发展成熟，人们又开发出了多种能够为网站提供更多特性和功能的语言。

JAVA 是一种独立于操作平台的编程语言。独立于操作平台意味着用 JAVA 编写的应用程序可以不经修改地在任意一台计算机上运行。尽管在 JAVA 诞生之初颇受欢迎，但近

来却越来越有被其他语言超越之势，包括网络动画利器 Adobe Flash，PHP、JavaScript 和微软的 JAP 等网络编程技术。

这些新技术中应用最为广泛的是 PHP、Flash 和 JavaScript。JavaScript 由 Netscape——一个曾经的网络浏览器先锋厂商开发。JavaScript 是一种可以在网页中内置的脚本语言，除了语言名称之外，JavaScript 跟 Java 语言本身并无太多关系。动态 HTML (DHTML) 是指联合使用 JavaScript 和 HTML 语言的技术体系，用于创建交互的动态网站。AJAX (Asynchronous JavaScript and XML) 也是目前同样流行的一种创建交互动态网站的技术体系。这种技术是如此流行，以至于微软也已规划将自己的 Atlas 技术与 ASP 相结合来实现类似 AJAX 的功能。

PHP 因其使用免费、相对比较容易掌握，同时又能被所有的网页浏览器识别，目前已成为比 JavaScript 更令人青睐的网络编程语言。

Flash 是 Adobe——一个为控制激光打印机开发 PostScript 脚本语言起家的著名公司旗下的一款应用程序软件产品。Flash 因其可以方便快捷地创建在网页中使用的漂亮动态图形而在 20 世纪 90 年代末期逐渐流行起来，Flash 本身也有自己的脚本语言 ActionScript，用户可以在 Flash 应用程序中使用这种脚本。

JavaScript、PHP、Flash 和 Java 比使用标准网页语言 HTML 在创建网站时能提供更多的功能。这些语言并不是仅仅用于为网站提供漂亮的动画效果，而且还能够像一般编程语言一样使得浏览者可以与网页上不可见的内容进行通信和交互。举例来说，一个在线购物网站可以通过这些语言来实现的功能有：用户在线查阅公司数据中的商品存货情况，处理用户的信用卡购物流程，联系物流公司将用户在线购买的商品托运到目的地等。

2.1.5 万维网

当读者使用浏览器在因特网上浏览网页时，浏览器和网页都有一些您应该熟知的通用界面。

1. 超链接

超链接是指向因特网上其他内容的链接，如图 2-8 所示。用户可以将超链接绑定到一个或几个文字上，也可以将超链接绑定到网页的图片上。在图 2-8 中，在 MSN.com 网页的顶部

一个超链接被绑定到了“个人教育”标题上。这种基于文字的超链接常被称为内置链接，通常这种内置链接都使用相同的样式来表现，标准的内置链接样式是蓝色文字加下划线。

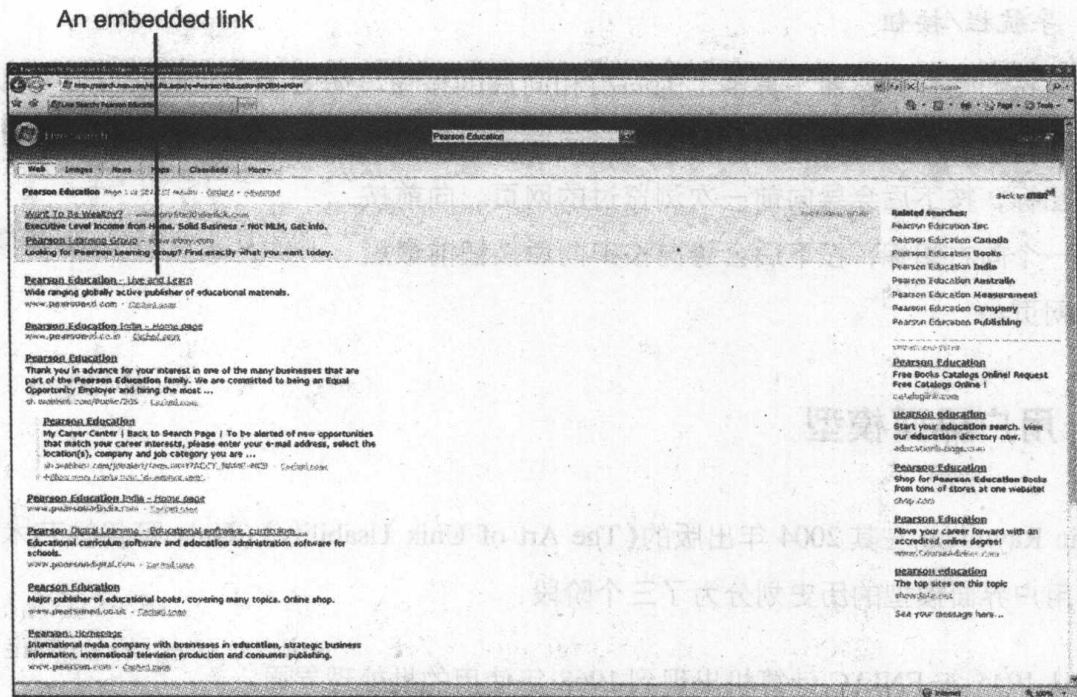


图 2-8 一个超链接的实例

2. 工具条

网页浏览器中的工具条与其他应用程序软件中的工具条一样，如图 2-9 所示。工具条包含了一排代表着浏览器常用命令的图标按钮，当用户点击某个图标，浏览器就会做出相应的反应。例如，用户单击 IE 浏览器中的后退图标按钮，上一次浏览过的网页就会出现在当前浏览器窗口中。



图 2-9 一个浏览器中的工具条实例

3. 地址栏

地址栏一般情况下位于浏览器窗口靠近顶部的位置，如图 2-10 所示。地址栏是用户输入网站地址 URL 的地方，如果用户浏览到网站中的某个子网页，则该网页的地址也会出现在地址栏上，这样用户就可以准确地知道目前的浏览地址。

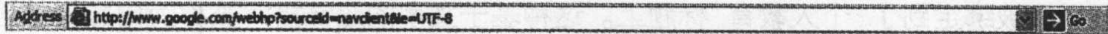


图 2-10 一个地址栏的实例

4. 导航栏/按钮

导航按钮是指浏览器工具条上的向前和向后的按钮，如图 2-11 所示，用户可以通过导航按钮在浏览过的网页之间来回切换。向后按钮上是一个向左图标，按下后会导向前一次浏览过的网页。向前按钮上是一个向右图标，按下后会导向按下向后按钮前最后浏览的网页。



图 2-11 一个导航按钮的实例

2.2 用户界面模型

Eric Raymond 在其 2004 年出版的《The Art of Unix Usability》(Unix 可用性艺术) 电子书中将用户界面模型的历史划分为了三个阶段：

- 从 1945 年 ENIAC 计算机出现到 1968 年使用的批处理界面。
- 从 1969 年到 1983 年使用的文本或者命令行界面。
- 从 1984 年到现在一直在使用的 GUI。

需要指出的是，从一种界面模型阶段到下一种界面模型阶段并不意味着上一种界面模型就会从此消失。20 世纪 80 年代的大多数 IBM PC 兼容计算机用户都还继续使用命令行用户界面，即所谓的磁盘操作系统，直到 20 世纪 90 年代初其中一些用户才逐渐过渡到使用 Windows GUI。实际上，GUI 最早在 20 世纪 70 年代就诞生在 Xerox Alto 实验室。但是，不同用户界面模型还是在前面提到的相应时间里，成为各自时期的主流。

2.2.1 批处理用户界面

批处理用户界面是出现在计算机上的第一种用户界面。这些计算机绝不是目前我们所常见的个人计算机，它们主要在政府机构、大学实验室和大公司中使用，而且这种计算机的占地面积往往大得惊人。这些计算机带有一个或多个界面可以让用户用带有穿孔的特殊规格卡片进行预先编程，比如三孔穿孔纸上穿孔。计算机可以识别这种穿孔卡片上按照一定规格打出的穿孔。一般来说，代表特定字符或者操作的穿孔是在卡片上垂直分布的。每

张卡片或多或少类似于我们后来所熟知的命令行或者程序中的一行代码，当然有些卡片也可以用来控制一批操作任务。这些所谓的任务卡片管理着其他数据卡片的操作，这些卡片会通过计算机的卡片读入装置将卡片中包含的信息送给计算机进行处理。

这种批处理用户界面模型的问题在于，在计算机处理这些数据卡片的时候用户无法与计算机进行交互。如果数据卡片的穿孔是有问题的，那么计算机的操作就会停止，用户就必须提供新的包含正确信息的数据卡片。经常出现的状况是，为了让一个程序运行需要花费数次，甚至数天时间。

2.2.2 命令行用户界面

命令行界面会显示一个命令提示符，让用户知道计算机已经准备好接受用户输入。用户可以使用键盘把命令敲入命令行，按回车键通知计算机根据输入的命令进行处理。然后，计算机就会做出相应的处理并将处理的结果用文本方式显示出来。当然，一个命令也可以调用一个包含了一系列命令或代码段的文件来进行批处理。

命令行用户界面的历史最早可以追述到 20 世纪 50 年代，当时最新的计算机开始出现电传打字设备和显示器。直到 20 世纪 70 年代，大型机和微型机界面都逐步从穿孔卡片过渡到基于阴极射线管 CRT 显示器的命令行用户界面。命令行用户界面为用户提供了与计算机进行交互的可能，当然用户也可以继续使用批处理的方式进行操作，但无论如何命令行用户界面要比批处理穿孔卡片界面进了很大一步。使用命令行用户界面的一个最大的劣势在于，用户必须学会大量的命令，才能指挥计算机进行你想要的操作。

IBM PC 及其兼容机的 DOS 操作系统是当时最流行的基于命令行用户界面操作系统，因为 PC 兼容机是当时最流行的电脑。DOS 内置了一个非常基本的脚本语言用于批处理编程，注意不要将这里的批处理编程和批处理用户界面相混淆。通过执行一个包含一系列命令的批处理文件，DOS 能够按照先后顺序执行批处理文件中的每个命令。

尽管 Windows 已经代替 DOS 成为主流操作系统，但是 Windows 并没有从根本上取代 DOS。相反的，Windows 从本质上讲是在 DOS 之上的一个壳，它可以让用户方便地使用命令而不必将每个命令敲入 DOS 界面中。换句话说，DOS 并没有消失，用户甚至还能够 Windows 下通过打开命令行窗口来激活一个 DOS 模拟环境，如图 2-12 所示。

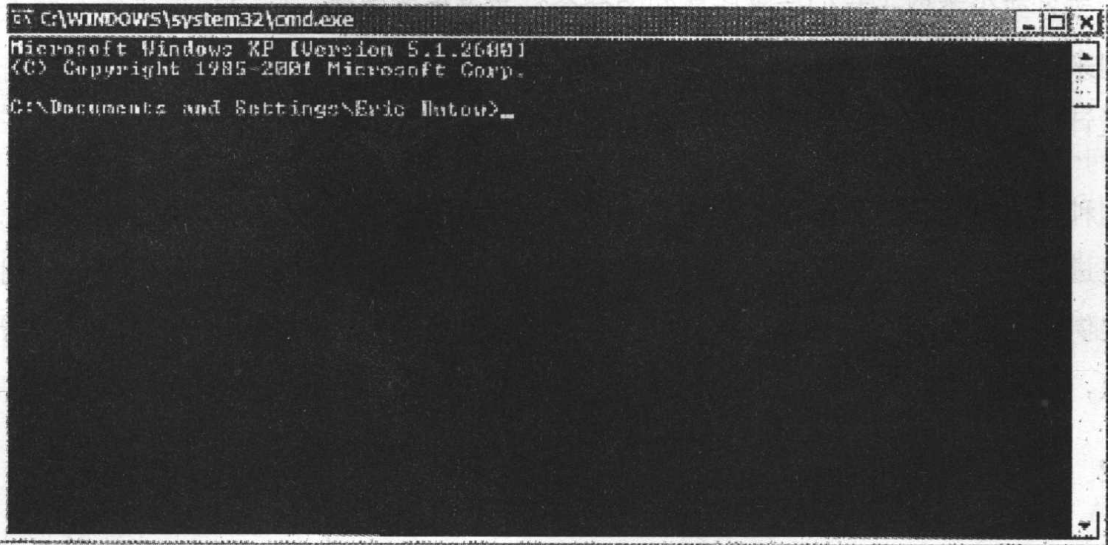


图 2-12 Windows 命令行窗口

尽管 GUI 已经替代命令行用户界面成为计算机用户界面的首选，但是还是有一些网络管理员和程序员仍然使用基于命令行的操作系统，他们认为使用命令行界面可以使他们的工作更有效。Windows XP 仍然提供了命令行窗口，以满足那些过去一直使用 DOS 的用户习惯。诸如 Mac OS X 和 Linux 等基于 UNIX 的操作系统也都允许用户在命令窗口中使用命令行界面来使用操作系统，这些基于 UNIX 的命令行用户界面通常也都拥有脚本编程能力。

命令行用户界面的发展并没有停止，微软正在开发一个名为 PowerShell 的命令行用户界面，包括使用基于微软的 C# 编程语言的脚本语言和减少键盘输入等特性。PowerShell 会在微软的最新通信协作软件系统 Exchange Server 2007 中得到广泛的应用。

基于 UNIX 的命令行用户界面同样也在不断地发展。比如，流行的 Bash (Bourne Again SHell) 壳，最早于 1978 年在 Bourne UNIX 下开发，2004 年升级到 3.0 版本，到笔者写作本书时已经更新到了 3.1 版本。

2.2.3 文本用户界面

自从 GUI 流行起来以后，又出现了一种所谓的文本用户界面以区别于 GUI，如图 2-13 所示。与命令行用户界面不同的是，文本用户界面使用整个屏幕作为用户输入区域。20 世纪 80 年代，IBM PC 及其兼容机大量使用了这种文本用户界面以提高功能性和可用性。

当时的许多知名应用软件都使用这种文本用户界面，比如 IBM PC 及其兼容机上使用的字处理软件 WordPerfect 以及可以让用户通过调制解调器获取在线公告牌系统信息的远程交流软件等。

```

Synchronet Main Menu

  Read/Post Messages      Message Area Selection      Electronic Mail
N New message scan       J Jump to new msg area      E Read/Send E-mail
R Read message prompt    * List sub-boards
Z Continuous new scan    /* List groups
B Browse new scan        { } # Select sub-board
Q QWK packet transfer    [ ] /# Select group

P Post a message         Go to
A Post auto-message      T File transfer section
                          G Text file section
                          C Chat section
                          X External programs

  Message Search
F Find text in messages
S Scan for msgs to you

  Other Commands
D Default user config
& Message scan config
U User lists
I Information
M Minute Bank
/L Node activity
^K Ctrl-key Menu
O Logoff BBS (or /O)

Anytime: Ctrl-U Who's online  Ctrl-P Send private msg  Ctrl-C Abort cmd/text
Main * 0:00:14 [1] Main [1] Notices: █

```

图 2-13 一个文本用户界面的实例

2.2.4 图形用户界面

GUI 最初为人所知是随着 1984 年 Macintosh 苹果电脑的推出而出现的，但真正被广泛接受还是因为 IBM PC 及其兼容机上开始使用 Windows 操作系统，直到 20 世纪 90 年代逐渐成为标准用户界面。在 GUI 下，用户不再需要敲入命令，而只需用鼠标点击图标和菜单来完成相应的操作。用户信息的组织方式很像一个真正的桌面，文件都位于文件夹中，用户可以把各种应用程序和其他资源都放置在屏幕上。实际上，GUI 模型的实现基本上都能够与现实物理世界中找到对应，桌面的概念在 Windows、Mac OS 和 Linux GUI 中都有实现。

在 GUI 中，用户可以通过使用鼠标（一种计算机外部硬件设备）来启动应用程序和操作某个图标对象，比如将鼠标移动到屏幕的特定位置上单击或者拖动特定图标等。上述这些隶属于四种不同类别的对象，即窗口、图标、菜单和指向标识通常被合称为 WIMP。在很多 GUI 中，默认的指向标识形状为一个箭头，但当用户进行不同的操作时指向标识也会做出相应的变化以区分不同的任务，比如在编辑一篇文档的时候和指示计算机处于繁忙状

态时的指向标识是完全不同的。

GUI 的另一个重要特征是用户可以自定义界面外观。不像传统的文本用户界面，用户最多只能做出诸如改变文字颜色等极少数操作，GUI 可以让用户改变屏幕上各种不同元素的外观和位置，包括桌面背景墙纸图片、图标和工具条的位置等。Windows 和其他一些 GUI 操作系统以及网站网页通常还为用户提供界面皮肤和主题。所谓的界面皮肤和主题实际上是一些为 GUI 的图形元素提供自定义外观的一些模板，比如 Windows Media Player 11 的界面效果，如图 2-14 所示。

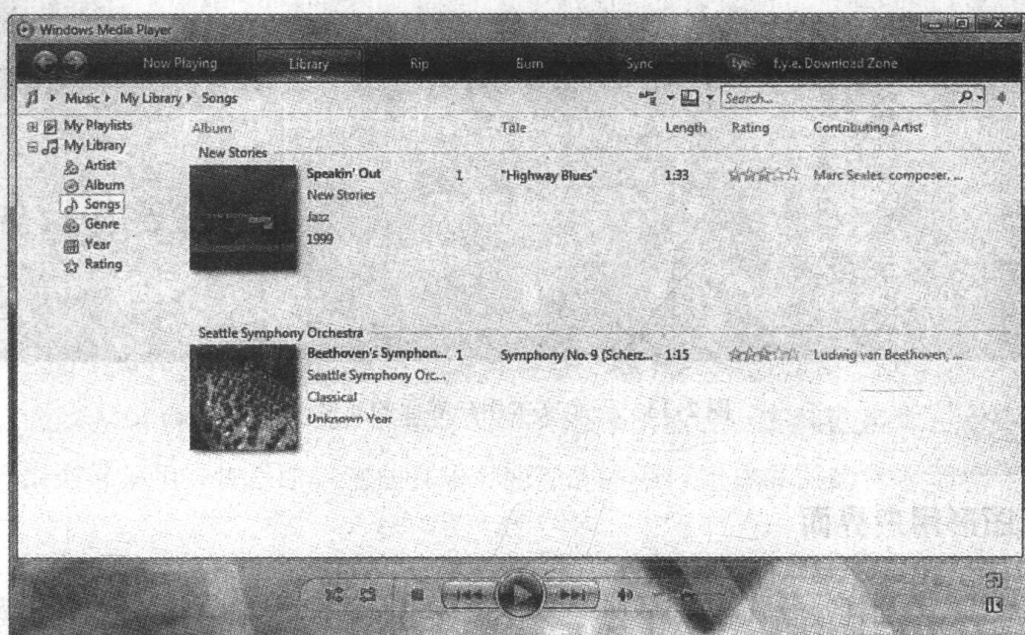


图 2-14 Windows Vista 下的 Windows Media Player11 窗口

在第 1 章中，我们讨论了 3 种主流操作系统和它们的 GUI，即 Windows、Mac OS 以及 Linux 下的 GNOME 和 KDE。实际上还有一些在基于 UNIX 操作系统的计算机上使用的其他 GUI，包括 SUN 工作站上使用的 Sun Solaris 操作系统等。

2.2.5 网站界面

基于 HTML 和其他一些后台技术的网站界面可以让用户与网页进行交互，如图 2-15 所示。比如，用户可以在适当的文本框中输入文字，点击按钮执行操作，或者点击文字和图片上的链接直接跳转到其他网页上。

BCG Customer Service Form - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://www.butow.net/form.htm

Google Search 511 blocked Check AutoLink AutoFill Smart

BCG

Customer Service Form

Thank you for filling out our customer service form. This form will help us understand your needs better so when we call you, we can provide you with more detailed information about how BCG can meet your needs. When you finish filling out the form, click the Send to BCG button at the bottom of the form.

You must fill out the form completely so we can contact you as quickly as possible. If you want to erase all fields in the form and start over, click the Start Over button at the bottom of the form. Thank you for your cooperation.

What is your name?

What is your e-mail address?

What is your telephone number?

Do you prefer being contacted by phone or e-mail?

E-mail Phone

What services are you interested in? Check all that apply.

Done Internet

图 2-15 一个网页表单的实例

网站通常包含了大量相互关联的网页，这些网页都使用相同的用户界面，界面的外观依赖于用户所使用的浏览器，也跟用户浏览网页时所使用的计算机屏幕分辨率相关。举例来说，假设一个网站的界面设计是面向使用 1024×768 像素分辨率的计算机的用户，这种分辨率就是所谓的 XGA (Extended Graphics Array) 分辨率，如果用户所使用的计算机屏幕的分辨率为 800×600 像素，那么该网页就无法一次性地完全展现在用户面前，用户要浏览完整网页则必须通过滚动条来查看当前未显示的部分。图 2-16 和图 2-17 分别展示了通过 IE 浏览器浏览 Person 主页在 800×600 像素分辨率和 1024×768 像素分辨率下的情况。

网站通过网页上的文字和图片链接提供了多种层次的交互，有的也可以通过各种表单进行交互，比如用户可以在线填写订购表单并将其提交到网站所属的公司。

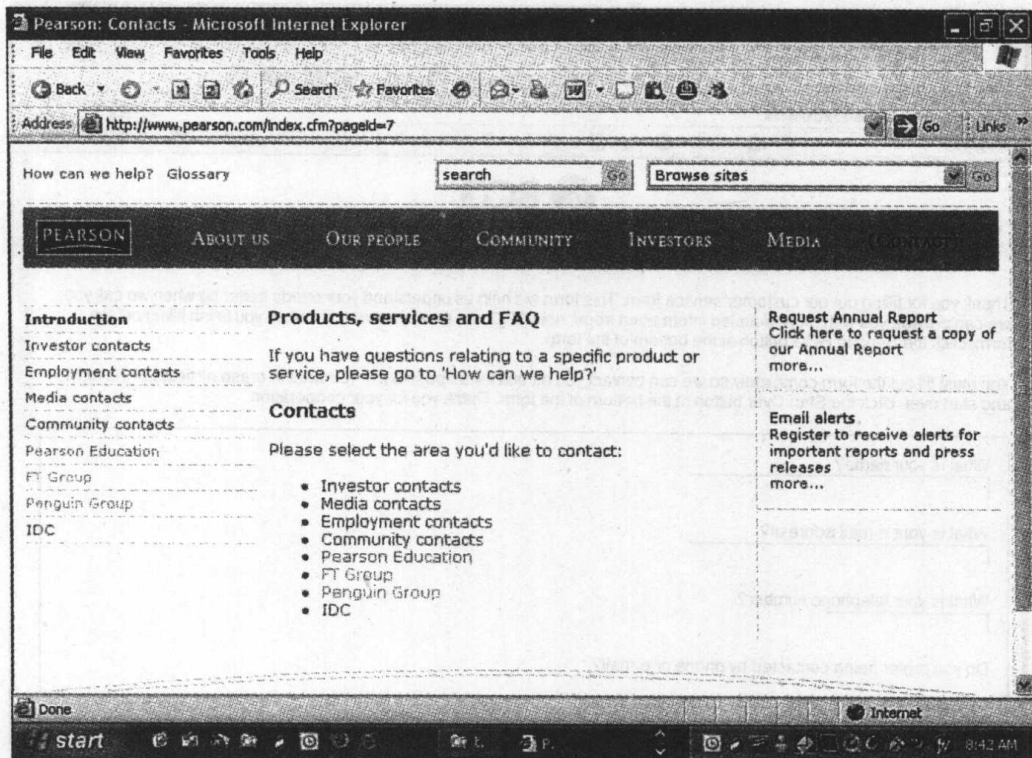


图 2-16 800 × 600 分辨率下的 Pearson 主页

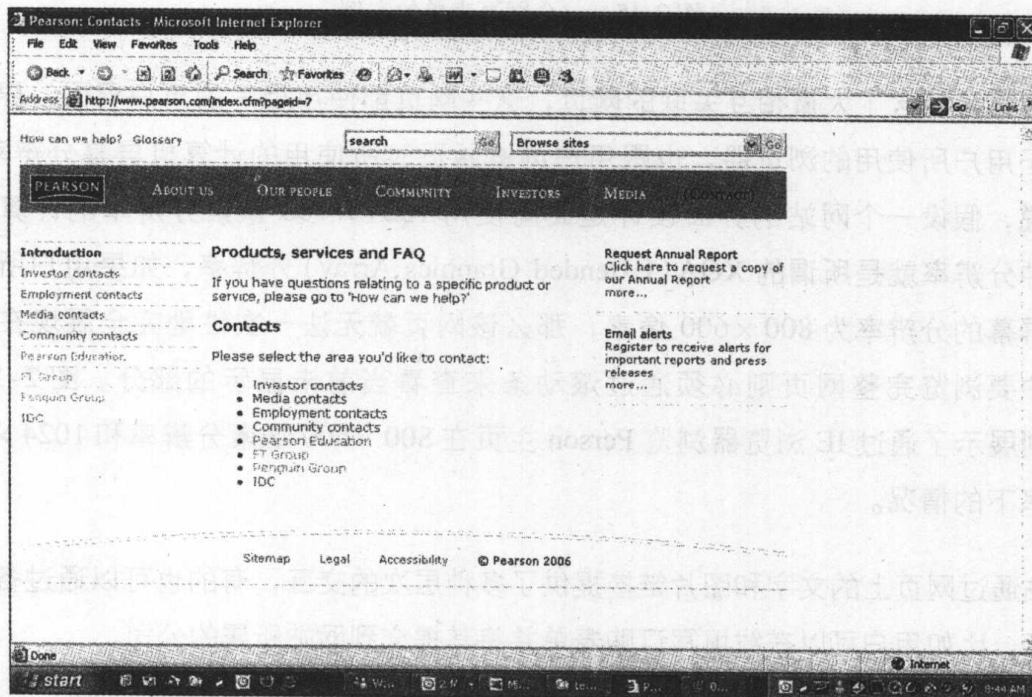


图 2-17 1024 × 768 分辨率下的 Pearson 主页。注意，在 800 × 600 分辨率下页面右边的垂直滚动条在 1024 × 768 分辨率下的已经不再需要

2.2.6 与用户界面模型集成的界面接口

一般来说,安全性对于计算而言是非常重要的,特别是在基于网络的计算更是如此。为了保证应用软件系统获得强大的安全性,通常有各种硬件接口与之相配合使用。这些硬件接口包括:

- **智能卡**,跟信用卡类似,如图 2-18 所示。使用时用户需将其插入一个单独的读卡器中,应用程序则可以从智能卡中读取数据,并将其同本地文件中的数据进行对比。



图 2-18 一个智能卡实物

- **指纹扫描**,用户只需将特定的手指放到一个小巧的扫描设备上,指纹信息会被扫描识别,然后系统会将扫描的结果与本地文件中保存的信息进行对比。
- **视网膜扫描**,听起来就像《星际迷航》(Star Trek)中的情节。跟《星级迷航》中的其他设备类似,这种设备也被应用到真实生活当中。用户需要将面部靠近视网膜扫描仪,扫描设备会扫描用户的眼睛,并将扫描得到的数据与本地文件中的数据进行对比。
- **无线射频识别(RFID)电子标签**,是一种无线移动设备,通过射频识别读取器与计算机系统交互。在本书写作的时候,无线射频识别电子标签已经用于货物跟踪和运输费用支付等领域,比如收费所的电子收费等。无线射频识别芯片也被植入动物体内用于跟踪研究,有时也用于人体。例如,2004年,墨西哥最高检察官办公室将 Verichip 公司提供的无线射频识别芯片植入其十八名员工体内,以便于其能快速进入检察院的一个密室。

2.2.7 其他用户界面

上述这些和其他一些旨在增强计算机安全性的硬件接口，毫无疑问地会对用户界面设计和操作系统功能产生重要影响。最新版本的 Windows 在用户界面上做了大量改进，以彰显其作为一个安全可靠的操作系统的形象。

现有的用户界面绝不仅仅局限于上面讨论的这些。实际上，一些新的用户界面也正在或者即将被广泛使用。

- **感触界面**是依靠触觉与系统进行交互和反馈，这种反馈也称为**触觉反馈**，常常用于各种模拟器中以获得更真实的操作体验。举例来说，飞行模拟器通过感触界面与使用者进行交互训练。当模拟器中的飞行员看到了模拟器屏幕上出现了某种紧急情况的提示时，模拟器中的飞行控制杆也会做出相应的反应，从而让飞行员能产生身临其境之感。
- **触摸界面**是在诸如商店超市、自动取款机、甚至投票站等地方应用非常广泛的一种用户界面。举例来说，去一个食品杂货店使用信用卡付款。那个食品杂货店的付款处有一个读卡机，我不仅能在上面刷卡，还能够用手指直接在上面输入我的折扣卡密码。读者也应该在许多地方发现类似的触摸用户界面，比如汽车上的触摸式地图导航仪、电台控制器和其他触摸设备，再有就是一些公共场合中提供公众服务的设备也通常使用这种触摸用户界面，比如公用因特网接入设备和机场的快速登机通道等。触摸用户界面也广泛应用于手持 PC 设备中，也就是我们熟知的个人数字助理 (PDA) 和袖珍个人电脑 (Pocket PC) 等。这些设备通常都会配备一支触笔，来模拟手指的操作，这样用户就可以通过触笔在面积很小的屏幕上操作图标和按钮了。当然，如果屏幕够大的话用户也可以直接用手指进行操作。
- **电话用户界面**要求用户使用电话按键输入与通讯系统中的程序进行交互。这种用户界面经常出现在向一个公司打电时系统要求用户输入分机号或部门号码，在获取电话答录机语音留言时也会使用到这种用户界面。通常这种电话用户界面会非常繁琐，答录系统会不厌其烦地要求用户键入某个数字来进行下一步的操作，整个过程会花费比较长的时间，以至于用户恨不得直接与接线员通话。不得不说的是，现在的手持 PC 与移动电话的差别越来越模糊，人们也常使用他们

的手机与其他人进行交互，比如传送图片、玩游戏和发送短信等。

● **按钮用户界面**在 iPod 或者视频游戏机上的应用非常广泛。尽管按钮用户界面和触摸用户界面在很大程度上有相似之处，但我还是倾向于将按钮用户界面单独列为一类，因为按钮用户界面是通过按钮等硬件来操作屏幕上的对象，而触摸用户界面则是直接在屏幕上进行操作。按钮用户界面可能以不同的形式出现。举例而言，iPod 的转轮按钮可以让用户快速地在列表中找到想要播放的音乐或者视频。按钮用户界面也可以与其他相关界面联合使用，比如在游戏中使用操纵杆来移动一个物体等。

2.3 界面设计改进和不足

2006 年在用户界面设计领域出现了一系列有趣的发展，先看最流行的操作系统：Microsoft Windows。

2.3.1 Windows Vista

Windows Vista，Windows 的第 6 代 GUI 操作系统在 2006 年末就开始向商业机构发布，2007 年初开始直接面向消费者。此版本的 Windows 对于软件系统有了很大的改善，但这种更新并不是革命性的。最大的改进之一就是 Aero 用户界面，这里首次引入了半透明窗口，使窗口倾斜的能力和全新的组织、搜索功能，如图 2-19 所示。安全性也得到了较大

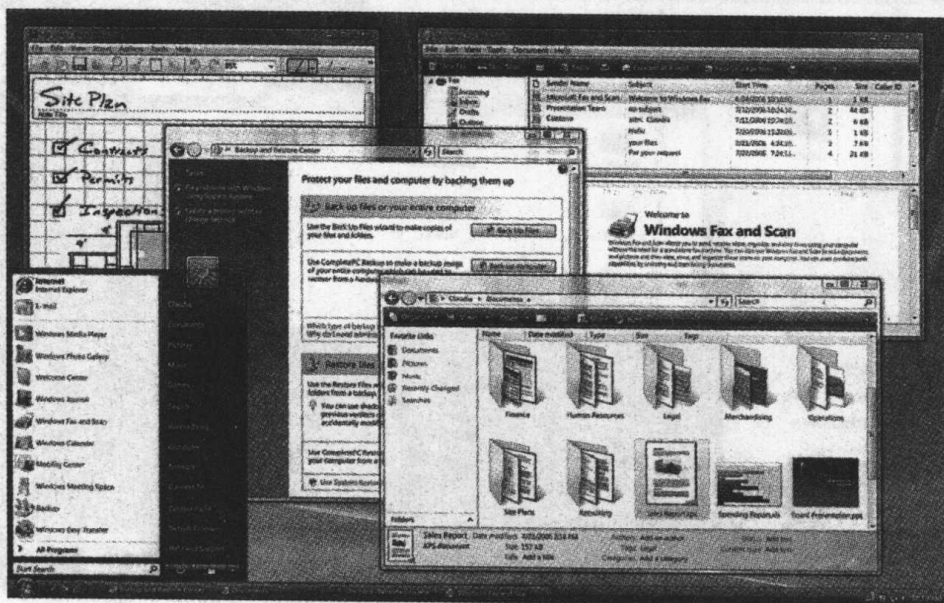


图 2-19 Windows Vista 系统桌面

的改善，特别是在用户权限控制方面的表现更加出色。

一直以来 Windows 所面临的最大的挑战就是系统的安全性。因为 Windows 是目前世界上使用最为广泛的操作系统，所以黑客们对 Windows 的攻击也是由来已久的。因此，Windows Vista 启用了一种被从业媒体说成是“不友好”的安全策略来进行应对。用户在启动应用程序或者启用外部设备时，Windows Vista 在默认模式下会弹出一个对话框确认用户登录账户是否拥有足够的权限进行相应的操作。这种界面设计对于那些，非常反感需要不停地在对话框中点击允许按钮来使用自己的系统的用户来说，将系统升级到 Vista 是要下很大的决心的。这种特性也给 Windows 的竞争对手 Apple 一个机会，可以向 Windows 用户表明他们的操作系统没有这样不友好的安全性控制，并能够提供更好的用户体验。

2.3.2 Mac OS X

Mac OS 一直以来都被认为是操作系统技术和可用性的领先者。在前一个 Mac OS X 版本中就已经在文件搜索、图形界面等方面做出了令人印象深刻的改进，包括 Aqua 图形界面的半透明窗口、魔幻按钮和 Dock 等，如图 2-20 所示。Mac OS 中的 Dock 是屏幕底部的一块区域，上面停靠着一些图标按钮可以让用户方便地进行操作，Windows 的任务栏也就是完成类似的功能。

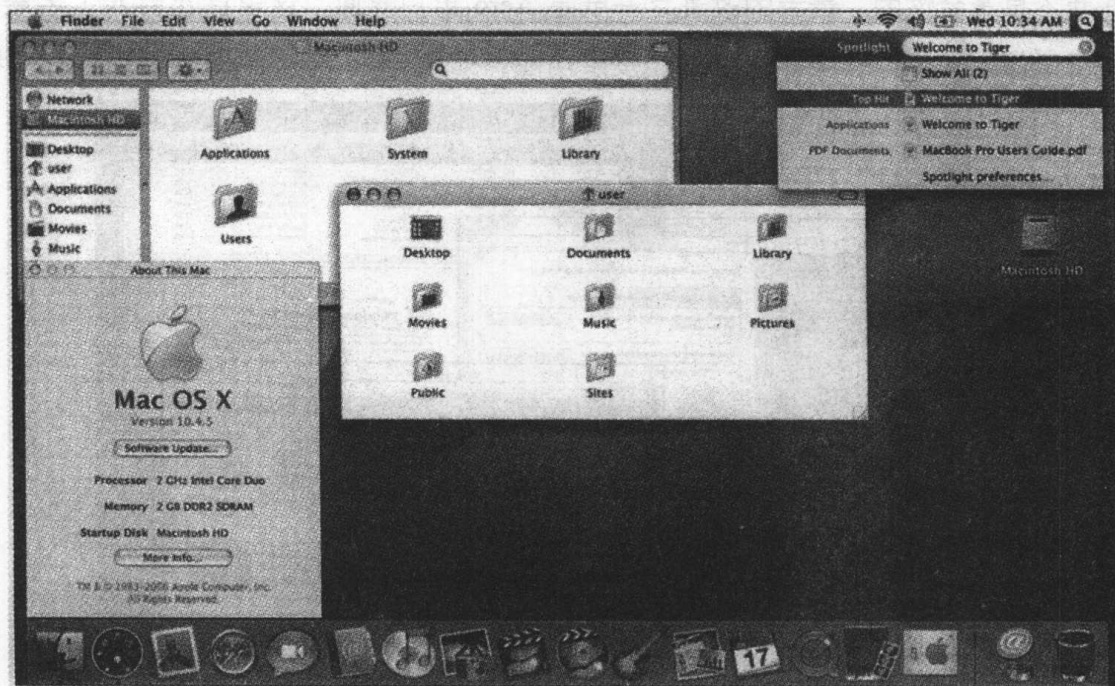


图 2-20 Mac OS X 系统桌面

最新的 Leopard 版本是对 Mac OS 的改进，它带来许多基于用户反馈的搜索和界面新特性。但是，随着微软 Windows Vista 的推出，各种类似于 Mac OS 的用户界面效果层出，Aqua 界面上的优势荡然无存。预计，随着 Apple 电脑的 Intel CPU 芯片战略的完成，Steve Jobs 将在 2008 年推出全新界面的 Mac OS 11 或者 XI、X1。

2.3.3 Linux

Linux 仍然使用着 2 个相互竞争的 GUI，这主要是因为 Linux 是个开源的操作系统。即便是 Linux 用户决定使用其中一个 GUI 作为标准，也可能不会因此为 Linux 带来更多的个人计算机使用者。在本书写作的时候，KDE 的最新版本为 3.5，如图 2-21 所示。KDE 4 将于 2006 年底或者 2007 年初发布，但就 GUI 来说它已经落伍了。GNOME 的新版本也在开发之中。但这些新开发的 GUI 是否能够与 Windows Vista 或者 Mac OS 相抗衡，我们还要拭目以待。

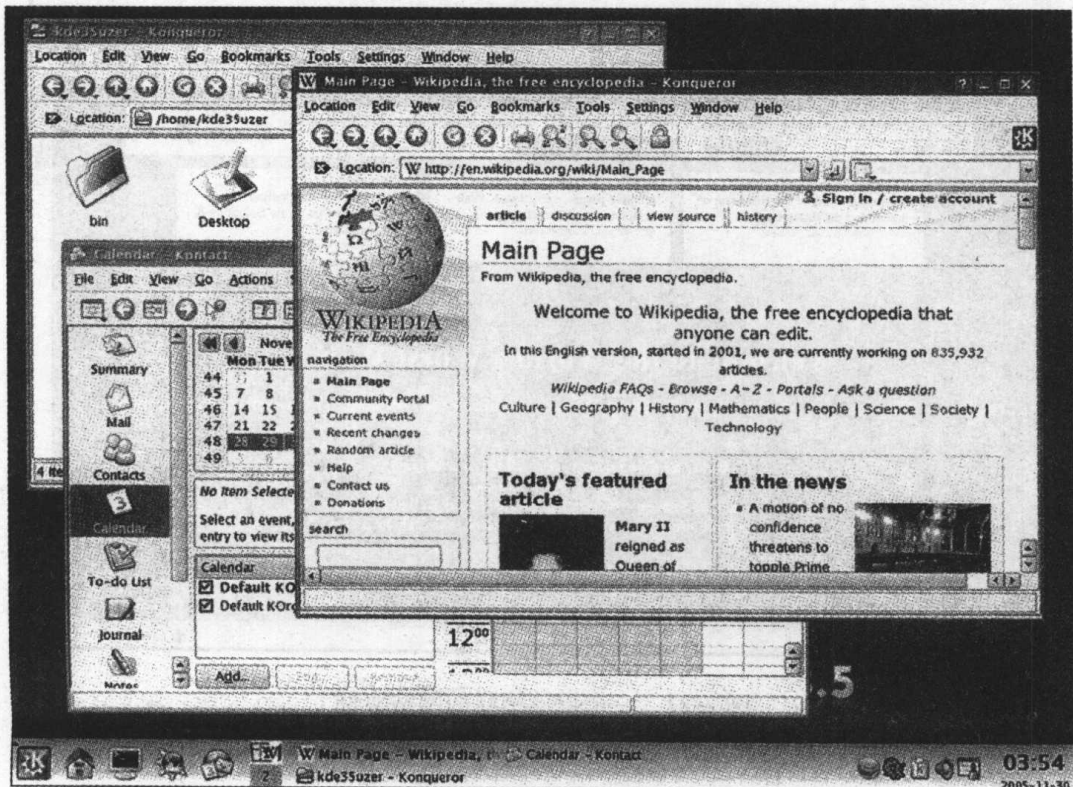


图 2-21 KDE 系统桌面

2.3.4 网站设计改进

在 21 世纪初的网页设计方面,改进基本上都是发生在幕后的,特别是网站编程语言的发展。可扩展标记语言(Extensible Markup Language, XML)可以说是 HTML 的表兄弟,这 2 种语言都源于标准通用标记语言(Standard Generalized Markup Language, SGML)。XML 被设计用于更好地在因特网上的不同系统之间共享信息。因此,HTML 开始有了淡出之势,现在的网站更倾向于使用 HTML 和 XML 的混合编码方式,即所谓的 XHTML。这样做的效果是,由数据驱动的网站在浏览器中的浏览效率得到了提高。

得益于人们坚持不懈地对网站设计好坏的讨论,现在的网站设计早已超越了那种使用黑底蓝字做网页的年代,虽然那样也可以被人们阅读。但是,新进网站设计者可能并不知道设计需求,同时也有一些网站因为设计糟糕而成为新进设计师的反面教材,如图 2-22 所示。

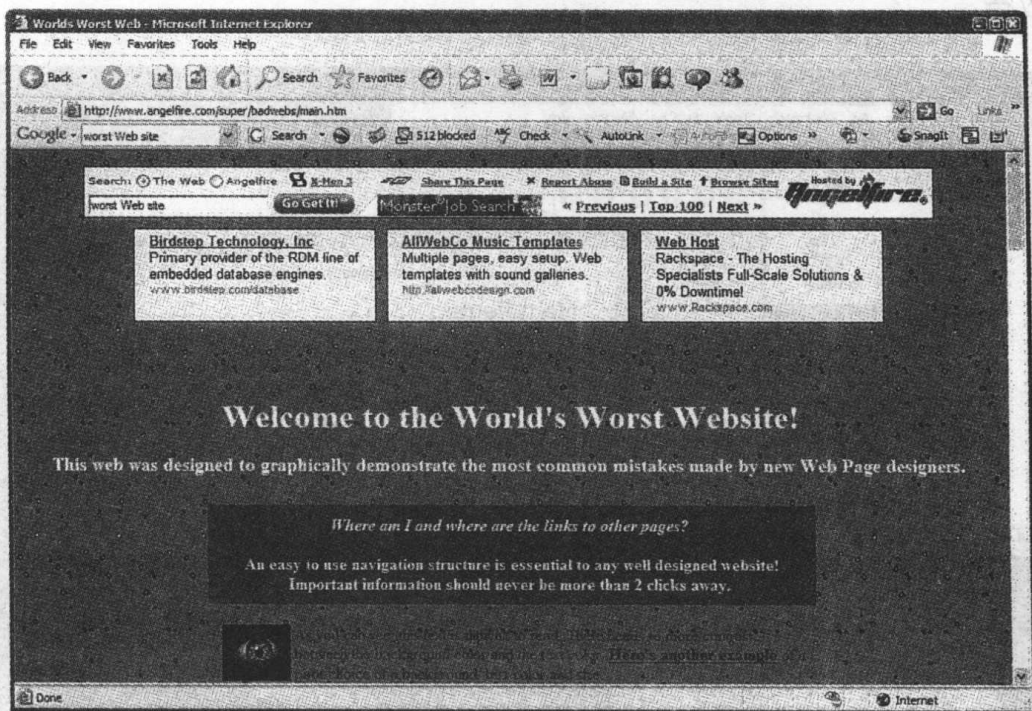


图 2-22 一个设计糟糕的网站实例

图片由 Gold Nugget 网站许可(www.goldnuggetweb.com)

遗憾的是,网站设计还是有一定的限制,如果一个设计师不清楚这些限制,那么设计出来的网站会带来严重的问题,我将在下一节进行解释。

2.3.5 仍未解决的问题

需要指出的是，并不是所有的操作系统，特别是网站浏览器都使用同样的操作方式，它们都设法为自己的应用程序添加其他应用程序没有的特殊操作。这表面上看好像是提高了自己的竞争优势，但这种做法通常会惹恼用户和设计师。

举例来说，每个主流的网页浏览器在显示网页时都会或多或少地有些不同，以区别于其他竞争对手。设计师也许不得不使用一些普通的字体和特定的颜色，即所谓的网页安全颜色来进行设计。这是因为并不是所有用户的系统中都安装了相同的字体，或者支持显示相同数量的颜色，因此设计一个在你看来是最好的网页在其他用户的计算机上看到效果可能完全不同。

而且，运行在同一个操作系统下的应用程序可能不使用同样的操作方式，这样带来的后果是破坏了用户在不同的应用程序中能够按照相似的方法进行操作的观念。比如，我喜欢用组合快捷键而不是用鼠标来选择命令，因为快捷键的方式显然更便捷。但是，相同功能的快捷键在不同的应用软件中可能是不一样的，比如拼写检查在 WordPerfect 中的快捷键为 Ctrl + F1，而 Word 使用的是 F7。

用户界面设计最大的问题可能在于界面很大程度上是静态的，标准 GUI 已经使用了将近 35 年了，尽管多年来已经做了大量的修缮，却没有出现一个能够与计算机进行交互的新方法。这种现象也存在于 GUI 应用程序中，多年来虽做了大量的调整，但是应用程序的界面还是跟 10 年前的版本有几分相似。

但是，微软可能会考虑在下一个重要 Windows 版本发布时加入新的用户界面，另外也有一些其他的用户界面正在开发过程中，人们有望在 21 世纪的第 2 个 10 年中看到全新的用户界面。

2.4 未来的展望

无论这些变化是循序渐进的还是革命性的，用户界面一定会继续发生改变。在节中我们将一起窥探一下，用户界面在不久的将来可能会带来有哪些改变。

2.4.1 Windows Vienna

Vienna 是第七个 Windows 操作系统的内部开发代号，它取代了这个版本的最初代号 Blackcomb。许多原定于在 Windows Vista 中推出的用户界面特性已经被计划到在 Vista 和 Vienna 之间的过渡版本中发布，也许会通过 Vista 升级包的形式出现，而其他的新用户界面特性则会延迟到 Vienna 正式发布时才会被加入。

这些改变之一是基于微软对过去近十年来用户界面的研究，从而对操作系统用户界面的一次彻底的更新。微软非常善于透露出一些可能会也可能不会在下一版本 Windows 操作系统中出现的新特性。而且，微软由于时间限制或者技术问题而导致的新特性推迟发布的这类事情也是有据可查的。所以，对于微软宣称的事情最好持保留态度。

2.4.2 Mac OS

2006 年，苹果公司更关注的是硬件而非软件。苹果公司决定将其一直坚守的使用 PowerPC 芯片的 Macintosh 计算机移植到使用 Intel 芯片平台上。这个转变的初衷是让苹果公司能够提供更多更强大的移动和桌面计算机。

苹果公司在 2007 年初发布了其最新的被称为 Leopard 的 Mac OS X 版本，但更明显的改变是现在可以在商店内很容易购买到基于 Intel 芯片的苹果电脑了。无论如何，Steve Jobs[⊖]已经开始为 Mac OS 下一个版本的开发摩拳擦掌了，他是不会虚张声势的。

2.4.3 网页浏览器及其对界面设计的影响

网页浏览器已经要求操作系统需要应用新的功能来处理诸如拦截弹出广告等一些互联网问题。用户界面设计也会涉及在线帮助系统的使用，这要求帮助信息能够以容易查找和理解的方式展示出来。还有就是对于所有用户的可及性问题，这个我们在第一章就曾经讨论过。

随着互联网和宽带业务的普及，越来越多的人可以长时间在线(如通过有线和 DSL)，电脑桌面和网页浏览器之间的界限也变得模糊起来，网站格式和设计限制也开始影响用户

⊖ 史蒂文·乔布斯，苹果公司 CEO。——译者注

界面设计。比如 Flash 这种用以在网页创建动画对象甚至创建动态网站的事实上的业界标准技术，已经使得现在的网站比以往任何时候都要更具交互性。

举例来说，丰富互联网应用程序系统(RIAS)使用 Flash 来创建看起来更像是一个用户桌面的网站界面。这种 RIAS 界面实际上是 HTML 页面上的一个图形化的壳，就如同 2000 年发布的 Windows Millennium 版中的第一个 Windows 界面是 DOS CLI 的壳一样。另一个例子是 Windows Vista 允许用户直接在桌面搜索互联网信息。

网站开发领域里很有意思的一块是基于网站的应用程序，用户可以在网站上直接使用诸如字处理和电子表格等应用程序，而不必在自己的计算机上安装相关的应用程序。应用程序服务提供商(ASPs)已经行动起来，使得越来越多网站可以提供在线应用程序服务。

应用程序订阅和租赁已开始关注像 Google 这样的大公司，Google 在 2006 年发布了 Google 电子表格。Google 电子表格允许用户通过互联网与其他用户共享表格数据，而不需要使用已有的电子表格应用程序。除了 Google 电子表格的开源软件身份，它对于像微软这样按照授权模式进行软件设计和销售的公司而言也是个巨大的挑战，特别是当这些公司要依靠销售这些软件来获得回报时更是如此。

2.4.4 未来界面

一些新的有趣的界面正在进行设计和研究，其中的一个或多个界面很有可能在不久的将来走进我们的生活。

- **体贴式界面**通过在一个过程中提醒用户潜在的问题来引导用户。从而控制用户的注意力。比如，当用户输入的信息不够而无法让应用程序完成后续的任务时，用户界面会给出相应的提示。这类界面被设计成能够理解用户正在进行的操作，从而能对用户的操作进行响应。举例来说，界面需要关注用户是否已经将注意力转移到其他地方了。
- **手势界面**依靠用户的手部姿态来进行输入。如果你看过电影 *Minority Report*[⊖]，影片中描绘了 2054 年计算机使用的手势界面，用户只需移动手和胳膊就可以操作屏幕上的对象。

⊖ 中影引进译名《少数派报告》，汤姆·克鲁斯主演。——译者注

- **自发界面**允许用户通过用户界面定义和控制整个系统，比如通过改变命令参数来满足用户的特殊需要等。
- **有形界面**可以让信息以某种物理形态的方式表现出来。比如，Durrell Bishop[⊖]发明的弹珠答录机通过弹珠来代表答录机中保存的信息。当你将一颗弹珠放入碟形容器中，答录机就会回放相应的信息。电影《少数派报告》中也用到了类似的系统，那里关于谋杀事件的推测结果不是显示在计算机屏幕上，而是以蚀刻了相应信息的弹珠形式出现。
- **可缩放界面**是图形用户界面演化的结果。我们现在常听到简称 ZUI，就指的是可缩放用户界面。ZUI 代表了在不同细节和规模层次上的对象。当你在一个包含了多种细节规模的无限桌面上平移时，你可以选择其中的某个对象并放大查看或者操作它，然后在完成操作后在将其缩小回先前的状态。ZUI 不使用窗口，而是使用矢量图形来表示对象。一个典型的 ZUI 界面是 MSN.com 地图，当你输入一个地址后，MSN.com 会显示出相应的地图，然后你可以放大到合适的大小来进行查看，如图 2-23 所示。

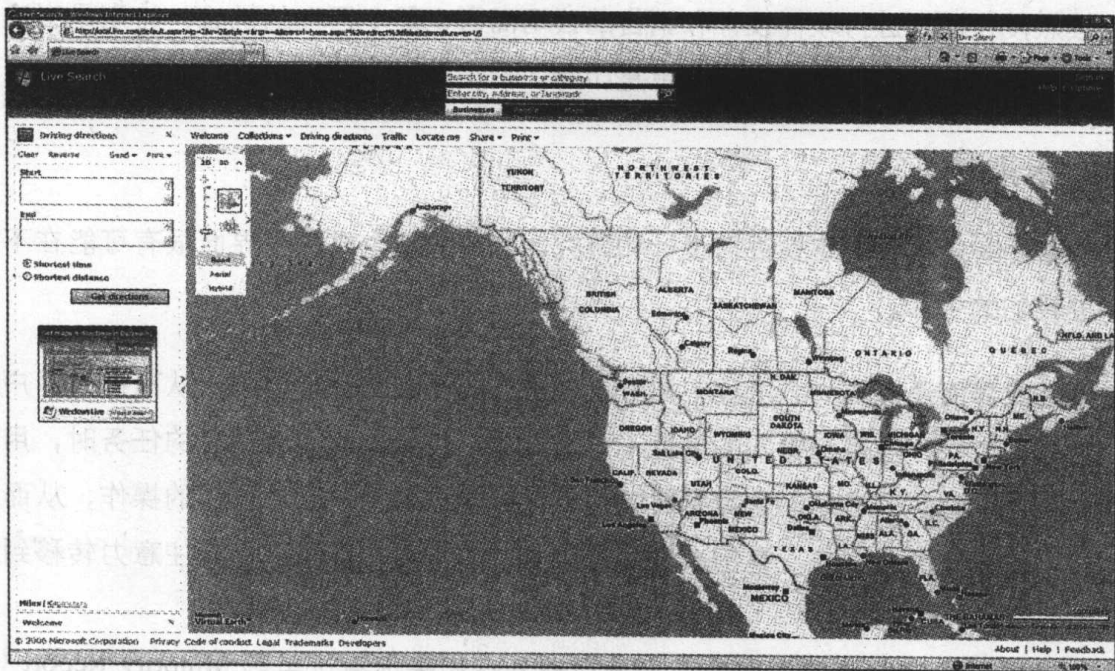


图 2-23 MSN.com 地图，一个 ZUI 实例

- **Archy 界面**是已故的 Jef Raskin 提出的全新的用户界面，Jef Raskin 是一位人机交互

⊖ 一个天才的发明家。——译者注

界面大师，苹果公司的 Macintosh 项目就是由他在 20 世纪 70 年代一手打造的。Raskin 于 1982 年离开了苹果公司并开始了自己的事业，最终在他自己的公司里开发出集成了早在 1987 年提出的最初的 Archy 界面的产品，即与佳能公司联手推出一款小型电脑产品 Canon Cat。这种界面是基于文本的，并没有使用图形用户界面特性。作为替代，Archy 界面使用所谓跳跃的方式进行操作，用户通过增量文字搜索在屏幕上移动。你可以在界面上的任何地方插入并执行命令，而你所要做的只是按住命令键（即大小写锁定键）并键入命令。Archy 界面也会自动帮你完成命令输入操作。Archy 界面也使用了一种被称为缩放世界的 ZUI，你可以通过使用鼠标来进行交互。每个图标都内置了超链接，你可以通过移动缩放区域来接近操作对象，当对象足够大时你就可以获取其更多的信息，并决定是否打开这个对象。

- **脑机界面**可能是最为直接的用户界面。虽然说将微芯片植入人脑并不是一天两天的事，但脑机界面的初步研究和开发已经可以让人和猴子通过脑电波来控制屏幕上的光标。相关的界面已经预示着，通过植入由人脑控制的芯片来帮助听觉和视觉有功能障碍的人士与计算机进行交互成为可能。

2.5 可用性术语

技术通信协会 (STC) 的资深会员 Chauny Wilson 指出，按照牛津英文在线辞典 (www.oed.com) 的解释，可用性 (usability) 一词最早可以追溯到 1382 年，而对可用性第一次引用则最早出现在 1842 年 (Wilson Chauncey, 《Usability and User Experience Design: The Next Decade》(可用性用户体验设计：下一个十年))。当今，Merriam-Webster 在线辞典对可用性给出了 2 个更简明扼要的定义：

- 1) 可以被使用。
- 2) 可以便捷和实际地使用。

这些定义并没有对什么是可用性本身进行解释。Dumas 和 Redish (1999) 为可用性给出一个更直接的定义：“人们能够快捷地使用产品，从而能简单地完成他们的任务”。可以说，Dumas 和 Redish 是基于以下 4 点来定义可用性的：

- 1) 可用性关注的是用户。
- 2) 人们使用产品是为了更好地产出。
- 3) 用户都愿意更快地完成任务。
- 4) 产品何时便于使用是由用户决定的。

随着个人计算机技术的广泛应用，越来越多的普通人而不是计算机专家都开始在家中和办公室里使用计算机硬件和软件产品，因此越来越多的公司都更强调改善用户与计算机的交互体验。

这种计算机受众的转变使得可用性研究领域产生了一系列新的专用术语，包括可用性工程学、以用户为中心的设计和用户体验工程学等。

由于过去的大约 25 年里人们对可用性研究的持续关注，我们甚至可以将这些可用性研究人员分为 3 种类别。

2.5.1 可用性工程师

第一类是可用性工程师，他们的主要工作是提供可用性相关的服务。所谓的可用性相关服务是指所有能够改善人们在使用产品完成任务过程中的用户体验的工作。这些工作包括：设计软件应用程序，编写用户指南，制作培训教程，搭建服务网站等等。

2.5.2 可用性科学家

第二类是可用性科学家，他们往往都在可用性研究和开发相关学科方面受过正规的训练。这些学科包括：可用性科学、可用性工程学、人机工程学、工效学等等。可用性科学家通常都拥有认知科学领域的高级学位。

2.5.3 用户体验专家

第三类是用户体验专家，用户体验专家可以归为上述 2 种类型之中，但不是公司、办公室或某部门中的正式可用性工程师或者供职于可用性工程部门中，却提供可用性相关服务。

举例来说，大部分技术文档编制者都倾向于编写尽可能便于阅读和使用的打印或在线文档。而且，他们还必须对其编写的应用程序代码、网站设计，甚至是软件和硬件设计进行注释。

这样一来客观上造成的结果是，越来越多的技术文档编写者成为了可用性工程师，或者简单地通过学习可用性测试和相关技术来提升自己的个人能力，进而为他们的内部和外部客户提供可用性相关的服务。

2.5.4 可用性设计分类

可用性设计也可以分为 3 个主要类型：

- **文档设计**，即设计纸质的或在线帮助文档供用户作为参考。技术文档编写者通常负责文档设计和可用性技术的应用，包括用户和任务分析等，这些内容我将在本章后面讨论。
- **界面设计**，是一个旨在设计出对用户有吸引力并易于使用的软件或硬件接口的过程。尽管 GUI 设计需求与软件设计不同，但 GUI 并不意味着这就是一个好的软件界面设计。
- **网页设计**，是一个设计网站的过程，将一系列网页连接起来向读者提供信息。

2.6 用户分析术语

随着可用性和用户分析研究的成熟，逐渐出现了用户设计和可用性测试的标准流程。目前，业界领先的用户分析过程是目标导向设计过程。可用性测试可以分为 3 种：快速粗略可用性测试、形成性可用性测试和总结性可用性测试。

2.6.1 目标导向设计过程

Cooper 和 Reimann(2003) 面向软件工程和用户界面设计创造了目标导向设计过程。使用目标导向设计过程的目的是保证每个设计相关人员都始终处于整个设计过程之中，避免主要依靠猜测来主导设计过程，从而保证设计过程中所做的每个决定都是有据可循的。如果读者您是某个产品项目团队中的一员，那么您可能会赞同这种目标导向设计过程。在本

书的第5章“用户行为”中将会更深入讨论目标导向设计过程，以及如何将这种设计过程应用于界面设计。

2.6.2 测试方法

针对用户目标的可用性测试，可以分为以下3种不同类型：

- **快速粗略可用性测试**——这种类型的可用性测试通常在产品已经被制造出来以后进行。这类可用性测试常采用的形式包括：打印的或在线的问卷调查，直接通过客户服务电话或者电子邮件获取用户的反馈。如果用户不喜欢我们的产品，那么我们也可以从客户电话投诉抱怨或者较低的销售业绩等方面得到印证。
- **形成性可用性测试**——这种类型的可用性测试通常在制造产品的过程中进行(Bias等, 2005)。读者可能听说过 *beta 测试员* 这一岗位，他们的职责是对软件的可用性和功能性进行测试。我们可以从这些测试人员那里得到关于产品品质是好还是坏的直接反馈。一般来说，用户在产品正式发布之前是看不到印刷版的产品文档的，所以针对产品文档的这种可用性测试都只局限于内部测试用户。当然，*beta 测试员*还是可以对软件产品的在线帮助系统进行测试，以确保这些帮助是有效和可以使用的。
- **总结性可用性测试**——总结性可用性测试通常在产品开发到由项目团队确定的特定阶段时进行，测试人员可以确认当前产品的开发进度(Bias等, 2005)。这种类型的测试一般都会对被测试产品的有效性、效用和满意度评估进行量化。在这个阶段中，如果有印刷的或在线的用户文档草稿出来，*beta 测试员*也会对这些文档的有用性进行测试和反馈。

理想的情况是，您应该通过*用户和任务分析*来真正理解您的目标用户。用户和任务分析是一个通过观察普通用户的行为进行分析的过程(Hackos和Redish, 1998)。第3章“制作商业案例”中涉及了用户目标的内容，第5章则对用户和任务分析进行了深入的探讨。

2.7 用户分析趋势

Wilson(2005)给出了未来一段时间内人们应该特别关注的6个关于可用性的趋势。

- 产品设计和评估的焦点一定是完全用户体验。本章已经对完全用户体验进行了探讨，第6章“用户分析”会继续深入讨论这个问题。
- 雇主会咨询可用性从业者从而让他们对公司的投资回报(ROI)有更多的信心。读者会在第3章中了解更多关于投资回报以及如何为管理团队计算投资回报的知识。
- 在新型协作和电子商业技术设计中社会心理学的重要性越来越突出。本书的第6章会对日常生活中的心理学及其对用户的影响进行分析和讨论。
- 是否拥有商业技巧和智慧在可用性和用户体验从业者的聘用中会成为重要的判断标准。本书的第3章会对这一问题进行更深入的探讨。
- 简化能力和技巧将变得与设计及评估技巧一样重要。本书的第9章将详细讨论简化技巧，以及如何规划可用性测试。
- 现行可用性测试方法的有效性和可靠性将得到检验。鉴于目前并没有可以遵循的可用性测试过程、测试数据分析和测试报告的标准，所以许多可用性测试和相关研究仍然处于成型阶段。

2.8 可及性问题

另一个没有在上面列出，但读者也应该知道的用户界面和用户分析趋势是，用户界面设计应该考虑到满足那些由于身体功能障碍或者年龄原因等导致的具有特殊可及性问题的用户需求。下列身体功能障碍会影响计算机用户：

- 完全或者部分视觉障碍。
- 颜色识别障碍。
- 听觉障碍或听觉困难。
- 认知障碍，如孤独症和诵读困难。
- 身体活动限制，如麻痹和腕管综合症。

操作系统生产厂商都在自己的操作系统中应用了大量的可及性相关特性。其中一些特性需要配合一些基于协助技术的外部硬件设备一起使用，这些硬件设备是为了满足有身体功能障碍的用户设计的。这些技术包括替代键盘和鼠标设备，摇杆和摇棒，吸吹系统和触摸屏等。

如果读者要开发一个软件应用程序或者一个网站，你就必须要知道美国政府强制执行的 508 法令条款，以及万维网联盟（即我们所熟知的 World Wide Web Consortium, W3C）发布的网站可及性指导规范。其他一些国家，包括澳大利亚和英国等，也有相应的反身体功能障碍歧视的相关法令影响用户界面的可及性。

2.8.1 508 条款可及性

在 1998 年，美国政府修订了 1973 年的康复法案 (Rehabilitation Act)，补充了包含由联邦政府维护的电子和在线内容的可及性规则的第 508 条款。如果你准备为美国联邦政府开发一个用户界面，你就必须遵守 508 条款中的规则，以保证供职于联邦政府的具有身体障碍的公务员也能够方便地访问。需要说明的是，508 条款与 1990 年通过的美国残疾人法案 (Americans with Disabilities Act) 是分开的。

无论是软件还是硬件产品开发者，包括网站设计师在内，都必须注意在产品中包含以下可及性相关特性：

- **软件应用程序**——应用程序中必须包含针对有视觉障碍用户的可及性特性，如替代键盘向导以及与语音识别产品的接口等。
- **通讯产品**——通讯产品必须包含针对听觉障碍用户的可及性特性，如集成 TTY 设备等。
- **视频或多媒体产品**——这些产品必须包含与显示内容同步的字幕，而且字幕的开启和关闭是可控制的。
- **自助式服务产品**——自助式服务系统必须包含可及性特性，比如内置可以将与用户的交互信息清晰读出的扬声器系统。
- **网站和应用程序**——必须使用 ALT 标签描述网页上的图片信息，所谓的 ALT 标签就是与图片绑定在一起的小段文字，用以简要描述图片的内容。当用户将鼠标移动到图片区域上时，ALT 标签就会出现在鼠标箭头旁边。

除了要满足 508 条款的规定外，美国政府还对网站可及性给出了 16 条不同的指导规范。具体的内容可以到 508 条款的官方网站上查看，网址是 www.section508.gov。

2.8.2 网站可及性

在 1999 年，W3C 通过了网站内容可及性指南 (Web Content Accessibility Guidelines) 的

第一个版本，为网站设计师提供了一些能够设计出易于访问的站点的指导规范。这些指导规范包括：

- 对听觉和视觉内容提供等同的替代内容。
- 不要仅依靠颜色。换句话说，就是确保文字和图片在没有颜色的情况下也能被正确理解。
- 适当地使用标记和样式表。
- 使用简明、自然的语言。
- 创建可以正常转换的表格。换言之，就是要确保使用正确的 HTML 命令来创建表格，以保证无论使用盲人显示器还是语音页面都能够正常访问表格。
- 确保以新技术为特性的页面转换正常。也就是说，对于使用了新技术的网页，即使在不支持诸如脚本或者脚本支持被关闭时也能正常显示。
- 确保用户对时间敏感内容更改的控制。
- 确保嵌入式用户界面的直接可及性。
- 进行与设备无关的设计。
- 使用过渡解决方案，以便较老的浏览器也能正常显示你的网站。
- 使用 W3C 技术和指导规范。
- 提供上下文环境和定向信息。
- 提供简洁的导航机制。
- 确保文档清楚、简单。

读者可以在 www.w3.org/TR/WAI-WEBCONTENT 上查阅完整的第一版指导规范。

2006 年 5 月，W3C 发布了网站内容可及性指南第二版的草案。与第一版中使用具体的指导规范不同，第二版的网站可及性指南只给出了 4 个指导原则：

- 1) 网站内容必须是可感知的。举例来说，网站的前景必须与背景区别开来。
- 2) 网站上的用户界面元素必须是可操作的。举例来说，用户界面应该能帮助用户避免误操作，并能方便地克服已经发生的操作错误。

3) 网站内容和控件必须是容易理解的。换言之，网页上的文字必须是有意义的，而且上下文必须前后一致。

4) 网站内容必须支持包括各种辅助技术在内的现在和将来的用户代理。你必须保证网站内容是可访问的，或者提供可访问的选择。

由于第二版的这个草案过于宽泛和含糊已经受到了一些批评，就网站可及性而言，甚至可能是一个倒退。读者可以在 www.w3.org/TR/WCAG20/ 上查阅完整的工作草案。

2.8.3 操作系统可及性

目前，4种主流的操作系统——Windows, Mac OS, Linux/KDE 和 Linux/GNOME——都内置了可及性相关特性。用户可以在控制面板——常见的用于让用户控制操作系统特性的用户界面中，启用可及性特性。这些特性包括：

- 改变字体大小、颜色，以及桌面对象的大小，比如增大桌面图标的尺寸、改用高对比度的配色方案等。
- 通过放大镜工具放大桌面屏幕的一部分，以获得更大的可视范围。
- 增大图标和改变图标闪动频率。
- 通过改变屏幕分辨率增大屏幕。
- 为声音和视频显示字幕。
- 为用户应该知晓的系统提示音显示警告文字。
- 通过设置窗口命令选项让计算机利用扬声器大声读出文字内容。
- 改变鼠标指针的大小和速度。
- 无需按下鼠标按钮进行高亮或者拖动鼠标操作。
- 允许用户通过一次按键来完成组合键的功能。
- 允许用户通过键盘上的数字小键盘移动鼠标。
- 用户按特定键时计算机能发出声音。

以上只是关于可及性特性选项的一小部分列表，是不完整的。而且，上面所列的特性并不是所有的操作系统都会提供。但是，这个列表会提醒读者在设计自己的软件应用程序

时明白操作系统可能拥有哪些可及性特性。别忘了为你的客户列一个这样的单子。读者可以在操作系统生产厂商的网站上查看其操作系统的可及性特性，4个主流操作系统中也可以很方便地查看各自的可及性特性，如 Windows Vista 中的 Ease of Access 窗口，如图 2-24 所示。

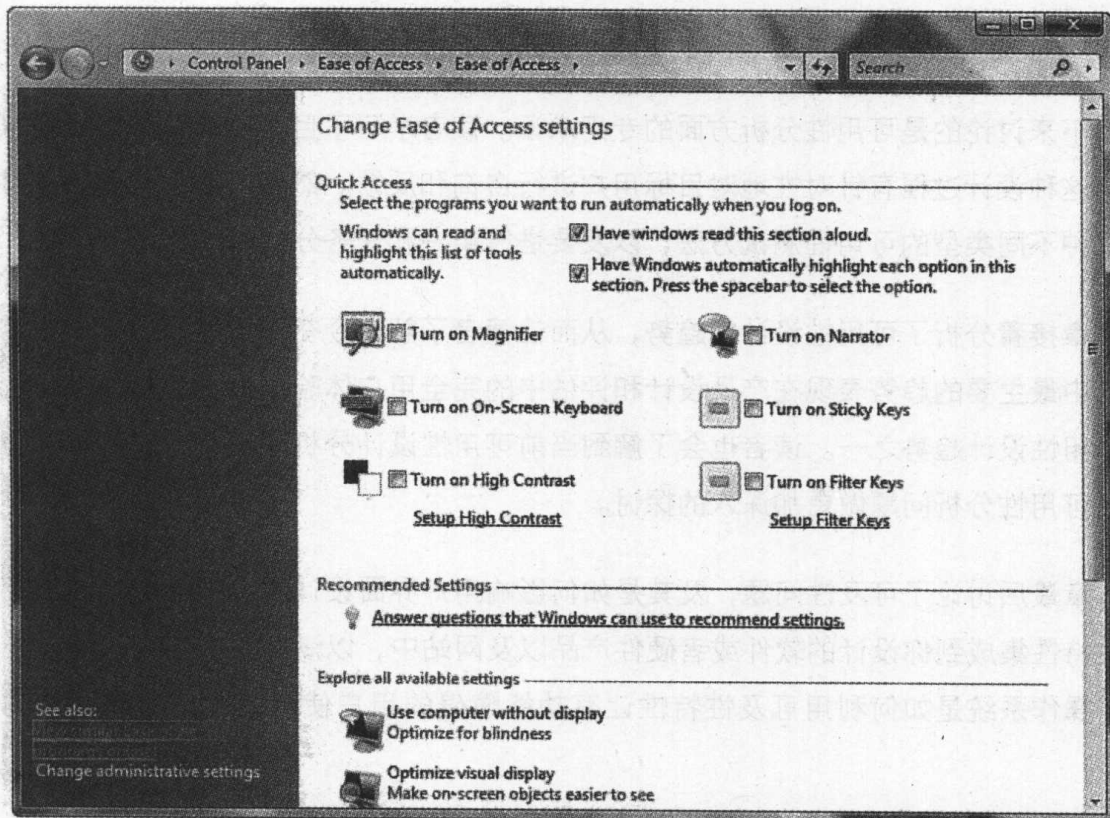


图 2-24 Windows Vista 下的 Ease of Access 窗口

2.9 小结

本章首先对计算机基本术语和概念进行了讨论。接着讨论了 3 类基本用户界面模型：批处理用户界面、命令行用户界面和 GUI。读者学习了这 3 种用户界面模型的区别，以及它们各自不同类型的与软件用户界面相配合对计算机进行控制的硬件用户界面。

然后我们讨论了界面设计的改进和不足。读者了解了在 2006 年发布的最新操作系统的用户界面设计新进展，网页背后的网站设计新技术，以及一些仍然需要强调的问题，如

当前的 GUI 越来越缺乏新意的现实。

接着对 Windows、Mac OS 和 Linux 3 种主流操作系统的用户界面设计的发展规划进行了探讨。在这 3 种主流操作系统中，微软对于下一代 Windows 的规划是最实际的，其中包括了在与计算机的交互方式上的重大改变。读者也学习了在不久的将来会出现的能够改变用户与计算机交互方式的界面，包括体贴的用户界面和人脑计算机界面等。

接下来讨论的是可用性分析方面的专用术语。读者学习了目标导向设计过程，以及如何应用这种设计过程有针对性地为目标用户进行咨询和反馈。然后，读者又接触了目前常用的几种不同类型的可用性测试方法，以及要进行用户和任务分析的原因。

本章接着分析了可用性设计的趋势，从而让读者了解当今在可用性分析方面的需求所在。其中最主要的趋势表现在产品设计和评估中的完全用户体验。社会和商业技巧也是目前的可用性设计趋势之一。读者也会了解到当前可用性设计分析的缺点，以及本书将在何处会对可用性分析问题做更加深入的探讨。

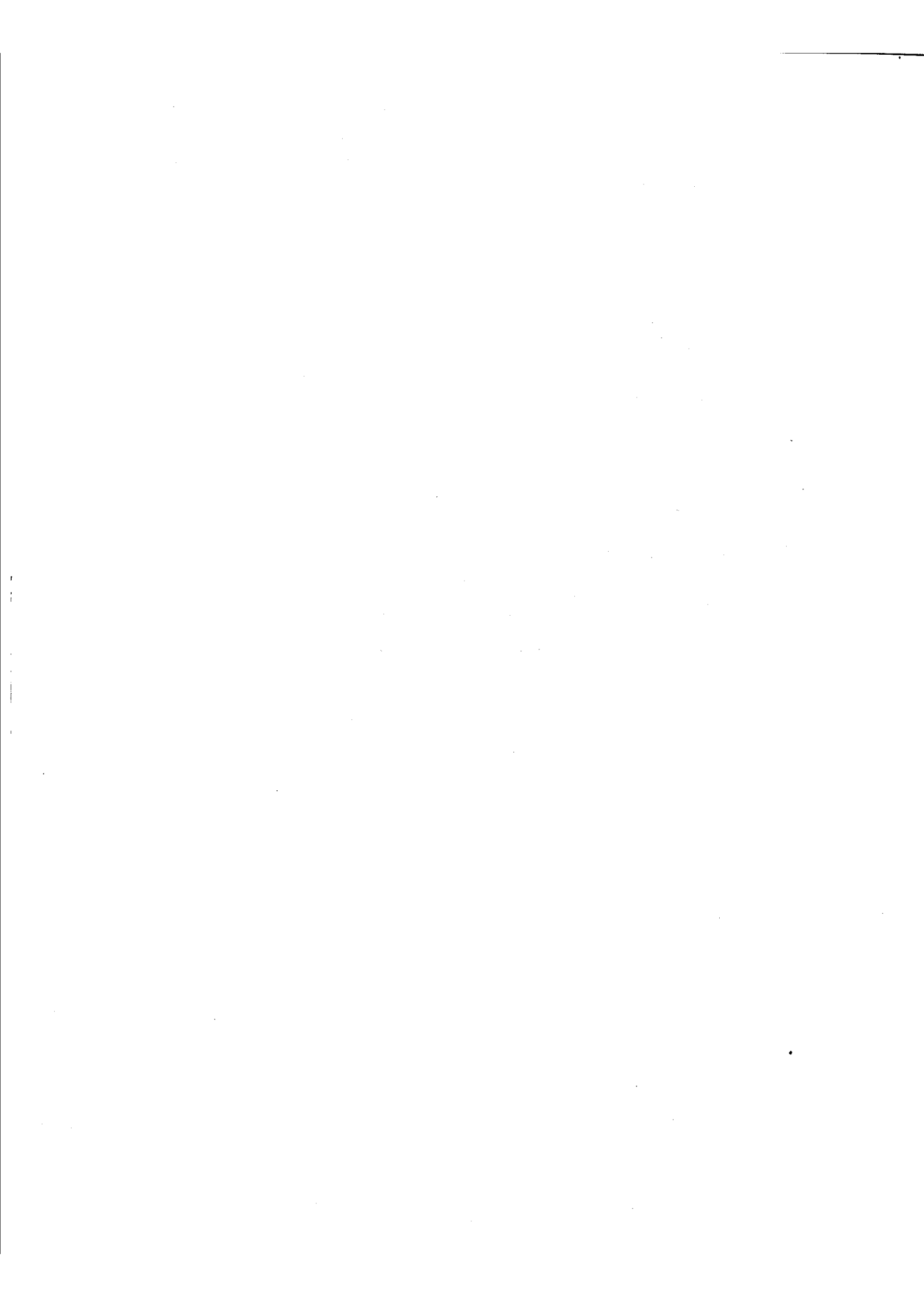
本章最后讨论了可及性问题，及其是如何影响用户界面设计的。读者应该时刻牢记将可及性特性集成到你设计的软件或者硬件产品以及网站中，以满足目标用户的需求。你也知道了操作系统是如何利用可及性特性让有功能障碍的用户使用软件应用程序和浏览网站的。

复习题

在进入第 3 章之前，读者最好能够回顾一下本章中学到的知识。所以，请思考并回答下面的问题，参考答案在附录 A 中。

1. GUI 的定义是什么？
2. 当今 3 种主流的 GUI 是什么？
3. GUI 的主要组成部分是什么？
4. 什么是因特网？

5. 什么是万维网?
6. 3 个主要的用户界面模型是什么?
7. 为什么说网站设计仍然具有很大的挑战性?
8. 为什么网页设计会对用户界面设计产生影响?
9. 什么是 ZUI?
10. Dumas 和 Redish 是如何定义可用性的?
11. 本章中讨论的 3 类可用性相关岗位或职业分别是什么?
12. 什么时候进行快速粗略可用性测试?
13. 什么时候进行总结性可用性测试?
14. 为什么用户界面方法的有效性和可靠性必须经过检验?
15. 可及性为什么重要?
16. 在美国, 什么法令对可及性需求进行了规定?





制作商业计划

“要推动事业，不要让事业来推动你。”

——本杰明·富兰克林

本章内容提要

- 项目合作者之间的分歧
- 制订一个商业计划框架
- 从优良设计中获益
- 可赢利案例
- 证实投资回报率
- 可用性工程生命周期

在因特网革命时代，从20世纪90年代中期到21世纪初期，任何一家和因特网相关或者一般的电脑公司都被视为金矿。实际上，很多没有具备执行性商业计划的公司，因为和因特网扯上了关系就拉到了投资。

这样的时光早已远去。尽管市场活跃躁动的日子不再有，但一个更为成熟的观念留下来，即怎么存活下来。你需要争取一切支持来帮助你在竞争中取胜。并且，一如既往，客

户的满意是生存和发展的法宝。

这就是为什么可用性研究如此重要——它可以帮助理解用户对你的用户界面有怎样的反应，这样就可以知道，在你的用户界面设计中哪些是对的，哪些是错的，同样的还有产品的其他周边材料，如文档。用户研究包括起初的消费者参与，到测量他们面对一个界面时的反应和效率。通过反馈，你有机会在产品面世前做出改进，之后将带来更多的满意用户。

用户界面设计、可用性设计、可用性测试密不可分。不管是软件硬件或是网站，没有好的用户界面设计，用户们就不会喜欢你的产品。没有可用性设计和测试，就永远不会知道你的产品是否可用，直到接受外界的评价。万一用户界面彻底失败了，你和你的项目团队将不得不花费时间、金钱和精力来解决本在开发过程中就应该解决的问题，更不用说产品团队在客户服务时将不得不忍受的头痛问题了。

你需要制作一个出色的商业模型，把好的界面和可用性设计的益处展示给项目关系人。不仅如此，你还需要说明公司如何从这项投资中盈利，这称为*投资回报率(ROI)*。

第一步是要把握和理解用户的目标，还有公司内项目关系人的目标。首先需要拟定一个总的计划来展示给项目关系人，以获得他们的同意。该计划不仅应该摆明优良设计的益处，也应该论述谁是受益人以及如何盈利。最后，应该做一个投资回报率研究，在合理时间和现金预算支出基础上，预计这项投资将赚多少钱。

3.1 项目关系人之间的分歧

产品建立体系包括4种截然不同的项目关系人群体(Donoghue, 2002):

- 用户
- 工程师和设计师
- 销售和市场人员
- 经理

以上每个群体对产品的成功都至关重要，无论是硬件、软件还是网站产品(或者三者

兼而有之)。他们对产品的期望也截然不同。不同的期望导致了项目关系人之间的分歧，你必须在你的可用性商业计划中将分歧解决。

3.1.1 用户期望

用户期望第一次接触产品或文档时就能上手。因为用户对哪些有用有最终发言权，了解你的用户的特性将对考核产品大有裨益。“用户行为”会在后面的第5章中详细讨论。尽管如此，用户在使用产品的时候通常会有以下期望：不论这个产品是网站、软件程序，还是一个手机这样的硬件(Donoghue, 2002)。

- 产品必须容易学习。
- 产品必须解决用户的需求。
- 产品帮助必须简单、容易获取，快速有效地解决用户的问题。

你的用户群对成功可能有着不同的定义标准，这些标准涉及更多产品细节。譬如说，你的网站顾客可能想用最少的点击次数来获取信息。

3.1.2 产品工程师与设计师的期望

产品工程师和设计师期望他们能在很大程度上自主开发产品。他们不希望花很多时间来解决产品发布后才会出现的问题，也不喜欢回答公司内外的顾客关于某些部分如何运作的提问。在从事技术写作的经验当中，我发现开发过程中工程师和设计师喜欢我尽量少提问，并且当回答这些问题只需要三言两语的时候，他们会更高兴。

3.1.3 销售与市场人员的期望

市场和销售人员期望的最终结果是为公司赚到尽可能多的钱。当一个产品面世时，市场部人员总是会考虑以下几个内外因素：

- 公司顾客群的基础和联系，尤其是是否有好的人选主持可用性测试。
- 公共环境，如公众和传媒。公众和传媒反应好坏取决于项目本身。坏的产品评论会对销售产生不好的影响，好的评论则能促进市场销售人员的工作。
- 宏观层面，如经济、政治和技术等。例如，假设你把软件卖给航空公司和货运公司，不管他们多么喜欢你的产品，高额的燃油费用仍然可能让他们对你的产品望而

却步。或者是你的产品可能要求他们安装最新的操作系统或最新发布的服务程序才能正常运行。

- 商业竞争。商业总是在挖掘每一点利益，因此市场销售人员在产品竞争中，总是在研究、找出和修正弱点或可以察觉到的弱点。

3.1.4 经理的期望

经理们通常希望看到一个产品或者创新是如何影响盈亏结算线，这是商业中唯一的价值所在。因此，经理们希望通过可用性测试来获得价值盈利分析从而预估出投资回报率。

经理们希望这一分析包含以下要素，以便他们可以判断可用性设计和测试是否可行：

- 可用性设计和测试的费用。这些花费不应该仅预算在短期内计划测试的内容，还包括长远来看想要测试的东西。举例而言，如果你开发一个未来将升级的软件产品，就需要考虑到后续版本的可用性测试。
- 测试的内容。比如说，要测试的是一个软件产品、还是配套的客户服务支持网站，亦或两者皆有。
- 其他资源的需求，包括人员、设施和器材。例如，如果需要在实验室的电脑或者客户网站上亲自组织可用性测试，需要明确费用和参与人员。同时还应该衡量在可用性工程执行期间相关人员的工作量。

你会发现，随着你在公司的地位越来越高，执行官不仅需要清楚经理制定决策的期望，也要了解其他部门的期望。

3.2 制订商业计划框架

既然已经明确了利益所在，现在需要将可用性计划和整体的开发过程联系起来。如果有可能的话，应该在项目设计伊始就同时启动可用性设计过程。

从一开始就启动可用性设计过程，不仅可以指导可用性测试的设计，而且可以从完全用户体验的角度来审视产品界面和文档。用户体验的5步过程(Donoghue, 2002)包括了整体客户体验：

1) 以商业目标为出发点。这些客户目标包括顾客转换、业绩保持、提升销量等，同时并不仅限于此。

2) 客户目标中的项目团队因素，包括：便于学习、满足客户需求的实现方式、在需要帮助时能便捷获取。帮助应该设计成能够迅速解决用户的需求。通常，帮助设计属于文档专家的工作范畴。在第4章“优良设计”中对于良好文档设计的需求将做更深入的探讨。

3) 由合格的项目开发团队成员设计同时满足商业目标和用户目标的用户界面或文档。

4) 准备进行界面测试之时，团队应该参与交互过程并提供反馈。你可以从许多方面得到反馈，从用户调查到受控测试。在第9章“可用性”中将了解到更多有关创建可用性测试方面的内容。

5) 经过测试之后，测试者的目标得到了满足。这并非表明已经满足了所有客户的目标，因为测试仅受限于选定的用户群。然而在向外界公布你的软硬件或网站前，可用性测试尽可能减少了可能遇到的问题。

在以上5个步骤的最后，客户目标的满足将很自然地带来商业目标的实现。伴随着良好设计带来的好处已经对商业目标产生的影响，你可以将这一过程作为一个开发计划其余部分的框架。

3.3 优良设计的益处

当讨论起好的设计的益处时，你往往会受到他人的质疑。例如，你和你的软件开发团队也许曾一次或几十次进行过下面的谈话：

你：“我们需要可用性测试来确保软件不仅可用而且易用。”

开发人员：“但是这个软件这么简单——我们不需要可用性测试！”

对于以上辩解的反驳基本上很少出现，因为如果开发人员是唯一使用该产品的人，他的观点的确没错。

当然，公司之外的人会付钱使用你的软件。如果这些用户并不认同开发人员的看法，

公司不仅会蒙受经济损失，还将在形象上大打折扣。这是因为公司将背上不能满足用户需求的恶名，这种印象很难消除。

对于你提出的可用性测试的要求，公司中的其他人可能只是耸耸肩，说“那又如何？”在没有具备可展示给他人的实质内容的时候，这些人不免对你想传达的信息充耳不闻。那么，你会告诉他们什么呢？

你应该从与公司讨论优良设计所带来的好处开始，着手你的商业计划 (Mayhew and Tremaine, 2005)。良好的可用性设计和测试对以下 4 个关键业务领域产生积极影响：

- 生产成本
- 客户支持成本
- 客户生产力
- 保留客户

3.3.1 降低生产成本

良好的设计有助于公司降低短期和长期生产成本。

1. 降低短期成本

流行的“桌面”GUI 界面 (如 Windows 和 Mac OS)，在 20 世纪 90 年代产生了流行一时的言论，即，由于用户逐步使用一个标准的界面，人们将完全能使用任何软件。而现实情况要复杂得多。桌面 GUI 界面没能解决以前存在的一个问题：应用软件往往会带来新的任务以及完成它的新的操作方式，同时用户需要知道如何有效地完成这些任务。

为了节约生产成本你应该这样做：像对待新产品一样，让开发团队在开发过程中尽可能早地更新现有的产品。如果市场和销售团队从用户那里了解到产品故障是因为用户设计中的缺陷造成的，整个公司将投入大量的时间、精力或资金来解决这些问题。结果很有可能是公司的信誉受损、产品的声誉受损以及其他开发进程中的产品进度延迟，因为产品团队不得不花时间解决现有的产品问题。

2. 降低长期成本

对于用户已不能接受的产品缺陷，不要花钱去解决这些问题了，而应该花更多时间改进产品并使客户的生活更加如意。你也可以把从当前项目总结到的用户界面经验用在其他

项目中，从而使这些项目更令用户满意。更好的产品能对公司的盈利情况产生正面影响。

3.3.2 降低客户支持费用

在高科技公司中，客户支持的费用在总成本中所占比例高达60%。如果用户发现界面易于使用，他们也许根本不需要接触客服人员。没有设计优良的用户界面作为引导，用户将根据自己的判断，总之通过他们设想的操作方式来得到想要的结果。界面给予的引导必须摆在用户面前，这样他们才不会忽视它，因为无法保证用户一定会阅读软件或硬件自带的说明文档。

举例而言，有很多用户都做过这样的事情：他们把光盘驱动器托盘当作水杯托盘使用，导致托盘破裂。这些用户使用托盘作为杯托，是因为生产商没有提前告诉他们这东西不是杯托，甚至没有在托盘上贴个条子，写上“这不是杯托”之类的话。这样一张贴条也许能给不少人节约点钱并减少些尴尬。

如果你的产品是精心设计的，用户就会因为觉得产品易用而更少依赖于在线帮助和培训材料，同时工作效率也会更高。他们不会遇到或犯下大量错误，但即使他们确实产生错误，良好的文档和在线帮助可以使他们更快地恢复工作。

好的产品可以降低培训费用。某些复杂产品需要更多的培训，但是良好的界面设计可以减少公司所需提供的非必要培训。例如，如果你的软件十分易用则可以采用在线培训的方式，任何人都可以从公司网站进行培训，如此你将可以节省让你的员工在你的地盘上或其他地方培训用户所花费的时间和现金。如果你想亲自培训，一款精心设计的产品将确保你用最少的时间培训用户，让他们更快地变得富有成效。

3.3.3 更有效的保留客户

对大多数客户而言，若要寻找当前问题的答案，帮助文档是首选。帮助文档的劣质或是干脆不存在，将直接导致更多的人尝试致电客服寻求帮助。于是，公司只好试图以声讯收费来降低雇用和培训客服人员的成本。

用户如果不想支付声讯费用而且没到非打电话不可的地步，通常会转而在网上搜索答案。一些用户会访问该公司的网站寻找答案。许多公司不仅列出经常被问到的问题

(FAQ)，也提供网上论坛用于提出和回答问题。此外，经常有一些与公司无关的网上论坛和邮件列表可供用户咨询。尽管如此，用户可能仍然没有找到他们需要的答案，此时他们有两个选择：掏钱并指望得到他们需要的信息；要么就放弃。用户并不乐意打电话给客服，他们觉得换用你竞争对手的产品是更好的办法。

用户为解决问题付出的任何额外时间，都会加深对公司的恶劣印象。在最坏的情况下，糟糕的设计可能造成用户的计算机或数据被破坏，他们会要求你的公司赔偿这些损失。如果你有侵略性的竞争对手，这对你或你的团队而言可不算是好消息。在商业世界里有个流行的警句：争取一个新客户比挽留一个老客户昂贵十倍。

因此，良好的设计不仅可以减少用户的挫折感，还可以为你的公司带来积极的推动力量。客户更加愉快不仅意味着留住了现有客户，而且通过客户推荐和口碑可以增加吸引新客户的机会。口口相传仍是最有效的营销方法。

3.4 如何盈利

在向项目关系人展示良好设计、可用性设计和测试将如何降低成本之后，你需要向他们展示设计和测试将如何帮公司赚钱。

可用性专家 Karen Donoghue (2002) 有 8 条指引公司成功实现可用性盈利的原则。以下即是这些原则：

- **应使用户界面的设计和开发与可用性设计与产品商业计划相辅相成。**如果产品团队已经有一个商业计划，则需要尽量将界面设计、可用性设计、可用性测试商业计划与产品商业计划融合在一起。这样做将有助于你论证可用性设计商业计划将对产品商业计划产生积极影响。
- **确保项目组所有成员(包括业务、技术和设计人员)认清商业目标，并知道可用性如何影响这些目标。**这对于取得项目关系人的认同十分重要，你应该尽快地与他们谈论你的想法和需求。这样的沟通不仅可以帮助有关方面了解你对可用性和良好用户界面设计的重视，而且你也可以知悉团队成员的意图，这么做会令你的商业计划更为完善。

- **将客户满意度的财务指标和可用性指标挂钩，并对它们以一种即时的方式进行衡量。**在这章后面的内容里将介绍如何通过投资回报率的研究以及使用可用性指标和可用性工程生命周期来创建财务指标。
- **成功考评个人的职责。**将考评作为可用性的一个指标(或一套指标)与全队分享。这个人可能是可用性专家、项目团队中的其他人，或者是项目团队之外负责质量监测的某个人。
- **项目组内部施行知识分享，并创造一种学习的氛围，使每个团队成员明白其他队员对用户体验的贡献。**不断的沟通是关键，不仅是达到好的设计目标的关键，而且对于获取准确且有意义的可用性结果也是如此。
- **建立用户体验的延展性。**随着业务模式的变化和用户人群的扩大变迁，延展的用户体验可以帮助推进可用性研究。同时，你应该将主要投资用于架构上而不是外观感受上。例如，如果用户需要你的软件升级到一个新版本的编程软件，则应优先在当前升级软件所必须的架构上投资。
- **了解用户的需求、任务和目标，并确保它们符合用户体验。**这就是可用性设计和测试的来源，但可用性测试并不是客户信息搜集的结束和全部。例如，你可以把用户带入项目团队，这样他们可以提供信息，或者你可以观察用户并举行聚焦小组会议，以确定界面中哪些是用户想要的、哪些是不想要的。
- **仅添加能使客户和公司双赢的特性和功能。**这又回到了第一条原则，你要将良好用户界面的设计开发过程与根据产品开发商业计划所做的可用性测试进行同步结合。在设计过程开始前，好的客户反馈能明晰哪些功能对客户而言是重要的。这样你和产品开发团队能够确定哪些需求能给公司创造最大的价值。

3.5 证实投资回报率

所有本章前面列出的价值主张都是不得已的，但它们通向最强有力的价值主张：投资回报分析。投资回报率提供了可以检测可用性测试对利润影响的财务指标。

投资回报率分析是一种回报计算方法，也就是说，如果你把资金投在某处，投资回报率分析可以让你和其他业务成员知道在一段时间内它是否盈利。如果你有一个或多个共同决策者，有投资回报率研究在手，你将有更大的胜算来实施可用性研究提议。

3.5.1 投资回报率具体说明

如果把投资回报率研究作为商业计划的一部分，你必须记录下获得效益的基础参数，其中包括(Mayhew and Tremaine, 2005)：

- **产生效益的成本**——这些成本基于创造利润所耗费的工作小时数以及实现这些利润所需的有形物品，如为可用性测试租用的空间和设备。
- **效益的价值主张**——这一主张说明公司将如何受惠于投资。你可以利用这一章中介绍的信息来显示效益价值。
- **效益的现金换算**——对公司从中节省多少钱以及赚多少利润进行最准确的预计估算。你应该与组织内的其他项目关系人(如客服经理)讨论这个问题以得到一些实实在在的数据，这样可以建立一个常规的金额预算。
- **效益实现的时间长度**——这一时间长度用若干年或一年的一部分(如0.5年，为期6个月的时间长度)来计算。在与项目团队讨论过之后，给出最准确的估计。项目期限将大大影响估算值。在计算收益的现金换算时应考虑到这个因素。
- **相同时间内特定业务的利润率**——可以从项目团队或公司的财务官员那里得到这方面的资料。

3.5.2 计算金额

为了计算出一个准确的效益金额，必须先计算净现值(NPV)的数额，即折算成当日的美元数量。如果投资回报分析表明，从现在起2年内你将赚到一笔钱，那么由于这期间的通货膨胀钱的数目将会缩水。

你可以用下列公式计算净现值(Mayhew and Tremaine, 2005)：

$$\text{净现值} = \text{未来金额} \times (n) / (1 + k)^n$$

在这一公式中， n 为周期数， k 为利息金额。例如，假设你想从现在开始一年内获得回报，你可以将 n 取为1。利率是1的分数。在这一例子中，我们使用5%的利率，方程中的 k 取为0.05。

如果计划未来的金额为5万元，净现值数额将是：

$$50\,000 \times 1 / (1 + 0.05)^1 = 47\,619.05 \text{ (单位: 美元)}$$

接下来需要计算投资回报率。使用下列公式计算投资回报率：

$$\text{投资回报率} = (\text{净现值金额} - \text{成本}) / \text{成本}$$

如果制定可用性测试的成本是 10 000 元，那么这一例子中，回报率如此计算：

$$(47\,619.05 - 10\,000) / 10\,000 = 3.76$$

故在该例子中，效益将产生 376% 的回报，这意味着在可用性设计和测试每花费 1 美元，公司将获得 3.76 美元。如果把类似这一例子的数字图标图表给项目关系人，极可能吸引住他们的眼球。

3.6 可用性工程生命周期

Bias 和 Mayhew (2005) 创造了可用性工程生命周期 (UEL) 这一手段用以建立一个可用性测试计划。如果一开始就将 UEL 与产品开发周期整合，它将提供严谨的分析和状况测试用于最大程度地帮助可用性设计、分析和测试。

UEL 是一个循环模式，包含 3 个阶段 (Bias and Mayhew, 2005)：

- 1) **需求分析**——在这一步骤中建立用户特征，该产品的操作需要什么任务，这样你可以判断用户的需求，设定可用性研究的目标，并确定可用性研究设计指南。
- 2) **设计、测试和开发**——该步骤中，创建一个结构化的自顶而下的方式来设计产品，产品有可能是用户界面、网页、文件或者三者组合。这一步需要从项目团队获得最多的反馈。
- 3) **安装**——在开发过程中和开发完成之后搜集用户反馈，并与项目团队一起探讨以确认是否需要对产品进行变更。

如果你和团队发现需要对产品做一些变更，很可能需要回到第二阶段再对项目团队做出的变更进行设计、测试和开发。另外用户测试还可以暴露需求分析中的缺点，这些缺点

可能导致重新做需求分析，这一步也得完全返工。

Mayhew 和 Tremaine(2005)认为，实施 UEL 来开发一个可用的网站或网络应用项目需要 8 至 12 个月，才能提供合理的投资回报率，但这种说法只是平均估计。我的经验表明，8 至 12 个月能否设计并推出一个网站取决于该网站的编程量。

因此，对于一个不需要太多编程量的网站，可能得在市场推广和根据需要进行额外的改动上花更多的时间。需要大量编程的网站，如动态驱动的网站，需要调用数据库来管理和输出信息，开发则更费时。这可能会延长实现合理的投资回报率的时间，或维持既定的开发时间并且相应减少实现投资回报率的时间。软件开发同样如此。

正如汽车行业所说，你的优势早晚会变。UEL 只是一个指引，因为每个项目都不相同，你可以对 UEL 进行适合需求的调整。你有可能会受制于紧迫的进度表，进行完整可用性测试的时间也不容许。但是，最好对于工程师如何准备可用性测试准备一份常规的说明，同时 UEL 本身具有足够的灵活性以便于选择需要进行稳固可用性测试的任务。然而，在产品开发进程中执行 UEL 时，要在头脑中切记保持 8 至 12 个月的时间表。

3.6.1 阶段 1：需求分析

可以从多个方面搜集用户的需求。例如，你可以使用纸上原型，让人对你的产品大致有个书面印象，同时知道系统如何对用户输入做出反应。在第 4 章中你可以了解到更多关于纸上原型的内容。你也可以观察用户，看他们如何操作；在第 9 章中可以对用户观察这一部分加深理解。

无论怎样来获得你的需求，都应该确保需求分析里涵盖以下几点。即使你没有创建一个纸上原型或观察用户的操作，也一定要注意下面几点以确保你打好了基础。

- **用户档案**——描述用户的具体特点。不存在特征标准集以供参考，但应该特别注意用户在使用软件时的任何问题，如物理限制。
- **文本环境分析**——研究用户的日常任务、工作流程模式、工作环境和概念框架。这将有助于你理解为什么用户对软件、硬件或正在测试的网站做出如此反应。
- **可用性目标设定**——需要制订质量具体化的目标，它能体现从用户档案中搜集到的需求。例如，你可能想让用户在某个阶段完成一个任务，还想看看他们能否做到。

如果用户受到限制，需要采用不同的方法才能完成任务，那就应该重新设定合适的用户目标。

- **平台功能和约束**——必须确定界面或产品的功能和限制，以界定可用性需求的范畴。这一信息会受用户可用性需求的影响。
- **通用设计指南**——必须使用公认的设计准则来设计你的界面。例如，参照网页设计指南制作出来的网页才能在网页浏览器上表现正常。在第7章“设计用户界面”中可以更多了解用户界面设计指南，第8章“设计网站”则专门论述网站的设计。

3.6.2 阶段2：设计、测试和开发

这一阶段分成3个级别的设计工作。它们涉及从需求分析的概念设计到开发出用户可测试的运行产品。

1. 第一级设计

第一级设计是概念层次的设计，即你在何处设计功能、工作流程以及规则。在决定如何设计概念模型之前，如果你和团队有足够的时间则应该从用户那获得尽可能多的信息。从用户身上推演出的模型，在第三级设计的评估阶段将更有可能让用户认可。该级包括以下4个步骤：

- **工作重整**——在开始设计前，基于用户的任务和流程工作，项目团队筹备功能和工作流设计。这一任务中不存在界面设计的工作。
- **概念模型设计**——团队创造用于表达信息并与软硬件或网站界面交互的高层次设计规则。如果您有产品演示图或网页，则这项任务不属于这一层面。
- **概念模型原型**——就像第4章中所介绍的，你可以创建纸上原型。你也可以创建线框版本，即一些演示部分功能的小程序集而不是全部程序。或者你甚至可以创建一个不能使用的原型，如用小的有色纸方块代表硬件原型的灯。
- **迭代概念模型评估**——项目小组评估原型并通过迭代评估程序修改它们。换句话说，如果团队不喜欢原型的一个或多个部分，迭代程序会反复改进直到该部分通过。

2. 第二级设计

第二级设计将创建项目标准。创建标准尤为重要，因为团队里每个人都需要清楚该项

目是如何组织在一起。让大家在进行用户界面设计的过程中形成了自己的标准，这也是解决混乱的一种方法。这一层次的设计由以下4个步骤组成。

- **设计标准**——现在已经建立好了模型，项目团队必须构建一套界面或特定站点的标准和规格，并运用于产品设计。
- **设计标准原型**——项目组将界面标准应用到产品上。这一功能可以体现在创建特定窗口或网页来测试到达其他窗口或网页的链接的外观风格。
- **迭代设计标准评估**——项目组进行正式的可用性测试或其他类型的评估，以改进界面上的视觉设计标准。这一过程将持续到主要可用性问题已经解决，此时可用性目标触手可及。在第9章中可以深入了解可用性测试。
- **风格指南的建立**——在拥有一套稳定且验证过的视觉设计标准之后，可以在产品风格指南中将它和需求分析的结果放在一起。然后将其以文档的方式分发给所有项目团队成员。其他风格指南，例如公司总体风格指南和文档风格指南，也可能影响到产品的风格指南，反之亦然。

3. 第三级设计

第三级设计指的是在前面的两级设计做好所有的准备工作后真正地设计产品。

- **详细用户界面设计**——项目设计者根据第二级设计中创建的风格指南来设计产品。得到的成果就是供内部或外部测试者使用的“beta”版，这一版本也可以供那些给产品团队提供反馈信息的人使用和测试。
- **迭代详细用户界面设计评估**——项目组进行正式可用性测试或其他类型的评估，以改进界面中的设计标准。这一过程将持续到项目小组确认该产品达到可用性目标。

3.6.3 阶段3：安装与反馈

在产品安装并使用了一段时间之后，公司应收集用户的反馈，了解他们喜欢或不喜欢什么以及他们如何使用产品。

可以通过电子邮件、电话、信件或网站等任意一种或几种方式得到反馈。可以给客户发送调查问卷，可以提供奖品或特别优惠来吸引客户反馈，尤其是在调查问卷比较长的情况下。也可以在公司内进行面对面的焦点小组访谈或利用网上协同工具软件(如 WebEx、微软的 LiveMeeting 或 Raindance)来召开实时视频会议。

3.6.4 不会结束的过程

有一点需要牢记，UEL 不存在真正意义上的结束。开发过程中的反馈可以确保在产品出炉后不会有太多的问题需要解决。好的产品反馈总是起到锦上添花的作用。开发过程结束后，同样需要来自客户对产品的反馈。

除此之外，你可能需要开发产品所需的升级程序，或者可能想更新放在公司网站上的文档。因此一定要把根据需求持续调查反馈所需的额外费用，纳入到投资回报率提案和商业案例当中，尤其是产品发布的中间阶段。

3.7 案例研究：Mike 的自行车店

本书将使用一个例子作为案例研究来阐释可用性设计和测试过程所涉及的步骤。“For Mere Mortals”系列中的其他一些书籍通常使用案例研究方法来说具有一定延续性的流程。在本书中，我将这一方式运用于兼顾内外客户使用的网站设计和相关数据库应用的开发。

你可能还记得由 Mike Hernandez 所写的《Database Design for Mere Mortals》(数据库设计入门)一书中提到的 Mike 的自行车店。在其案例研究中，Mike 的自行车店是新开的一家自行车店，位于西雅图郊区绿湖。我们的案例中将从 3 年后开始讲述，此时 Mike 的自行车店经营得很好，在大西雅图地区另开了 8 家连锁店，雇用了近 120 名员工。鉴于这种增长，Mike 发现客户现在想在网订购自行车和其他用品。员工们则想要一种更强大的工具，可以让他们很快找到所需的信息。

Mike 拥有一个 10 人团队，准备开发新的网站和数据库系统。以下是团队的 10 名成员名单：

- Mike, 店主
- Traci, 财务经理
- Jay, 营销经理
- Laura, 生产经理

- Michelle, 客服经理
- Tony, 系统资源管理员
- Maureen, 数据库和网络管理员
- Bruce 和 Travis, 两个数据库程序员
- Paul, 文书

团队干劲十足, 但不明白为何这一项目要做可用性测试。这就是为什么你和助理 Evan 与团队开动员会议(kickoff meeting)的目的: 建立一个商业计划框架。

建立商业计划框架的第一步, 是与项目团队会谈以了解业务目标。你让 Evan 主持会谈。

Evan: “这一项目的商业目标是什么?”

Mike: “多赚点钱!”(其余成员会意微笑。)

Jay: “让客户和竞争对手认同 Mike 的自行车店拥有自行车及配件方面最好的资源。”

Michelle: “让客户随时随地订购自行车并获得服务, 而且能快速得到产品的信息。”

Laura: “让员工更容易获得更多的信息, 这样工作会更有效率。”

Maureen: “我和程序员可以做一些更重要的事情, 而不只是把客户的电话订单信息录入数据库。”

在有关商业目标和项目完工期限的讨论之后, 会谈继续讨论客户目标, Evan 继续发问以促进讨论。比如, 下面的讨论是关于用户不想在界面中看到的東西, 这样他们可以快速命中和获取他们需要的信息。

Evan: “你怎么看待你当前使用的数据库?”

Mike: “你的意思是……”

Evan: “我想知道现在的界面有没有什么地方让你抓狂得想要改进它。”

Michelle: “我想网站上应该有一个可以从数据库的任一屏访问的网页，告诉我哪个店的哪些资源空置，这样一个分店的顾客可以调用另一个分店的东西。”

Jay: “这会很庞大。如果客户不能从我们这里找到他们想要的，她就不会再来。”

Laura: “我认为数据库需要告诉我们商店缺什么货，而不是仅仅告诉我们一个分店没法支援另一个分店，因为这样商店将没法提供给客户他们想要的东西。”

Mike: “你们建议怎么做？”

Laura: “我们需要产品列表中另外有一栏视觉提示，就像一面旗帜，好让我知道需要订货来保证货源充足。”

Maureen: “这个不难。”

Laura: “我认为那一栏旁边应放个按钮好让我订购零件。点这个按钮就能打开制造商的网站，这样我可以在线订购。”

Traci: “前提是制造商能让你在线订购。”

Evan: “Laura，如果网站出错或者制造商不让商店在线订购，那该怎么提示？”

Laura: “嗯，也许按钮可以打开一个列出联系资料的小窗口，如果我们可以在线订购，还应该包括这个公司的网站链接。”

Evan: “但这让访问网站增加了额外的步骤。创建一个连接到制造商网站的单独按钮，如何？”

Laura: “好主意。如果公司可以在线订购，则放上不同颜色的第二个按钮，直接带到网站的订单页面。”

Maureen: “得在数据库里另外建一个模块管理联络资料，不过这个我们能搞定。”

Jay: “但如何把货发出去而不用麻烦客户开车到我们店里？除非客户立刻要用，否则她不可能穿过城市跑到我们店里来，她可能会去两条街外的 Rob's Cycle World 这家店。”

Mike: “我们需要雇些人, 也许是高中生或大学生, 他们开车或骑自行车到顾客所在地的商店去提货送货。这意味着需要提示商店经理另外一个商店缺货。我得再考虑考虑。”

从反复的讨论中得到的信息可以用来创建一个所有目标的清单。比如下面给出的几个 Mike 的自行车店网站和数据库应用程序需要达到的界面目标:

- 客户需要在网站上尽可能快地找到所需内容。
- 网站必须反映准确的信息, 如可供购买的产品数量。
- 客户可以很轻松地定制定单, 并且快速安全地订购商品。
- 如果用户在网站中迷失位置, 能够返回首页并迅速重新开始。
- 客户需要快捷到达产品和支持文档。
- 我们需要快速获取客户、库存、销售、供应商和员工的信息。

你与 Mike 以及团队的其他成员共同查阅界面目标的基本列表。接下来, 你和 Evan 完善列表并提交给团队。你和 Evan 还要向小组提交一份报告, 用符号形式列出 3 个主要的设计目标:

- 产品供货页面必须可以从数据库程序的任何窗口中打开。这一页面提供下列功能:
 - 显示每家商店还有多少存货。
 - 使雇员更容易快捷地找到零件。
- “零件维修”页面应给出表示每个零件状态的视觉暗示。这一页面将提供给雇员下列功能:
 - 雇员可以看到其他店面需要某个库存的警示信息。
 - 雇员能够确定每个商店中有多少损坏的零件。
 - 雇员可以很容易判断哪些零件需要再次订购。
- 搜索框可以从数据库程序的任何窗口中访问。这一页面可以让雇员寻找产品、客户资料或订单信息。

有了基本列表之后, 你需要询问团队将采取什么方式开发产品和界面。当你谈到开发方式的时候, 要考虑开发项目所需雇员工作时间所产生的费用以及日后的预期成本; 例如, 公司为了在特定日期前完成项目可能打算聘请合同员工。你和 Evan 还需要了解商业利润率, 涉及公司预计获得的利润以及公司何时将实现赢利。

有了这些信息的支援，你和 Evan 将建立一份投资回报的说明，在下次团队会议之前需要与 Mike 和 Traci 共同进行评估。这个投资回报率的计算是这一章前面部分介绍的公式：

$$\text{净现值} = \text{未来金额} \times (n) / (1 + k)^n$$

现在需要把净现值代入投资回报率的计算公式：

$$\text{投资回报率} = (\text{净现值} - \text{成本}) / \text{成本}$$

净现值是公司从可用性设计和测试的投资中获得的百分比回报。你和 Evan 根据 Traci 的估计，未来一年的现金费用为 45 000 美元（每家店 5 000 元）可以带来一年 5.25% 的利润。同时你和 Evan 估计用于生成测试文件和最终报告以及支付与店员面谈的测试人员、创建纸上原型、观察员工如何使用开发中的新系统和现有系统共需要费用 12 500 美元。这些费用记录在图 3-1 所示的表格中。

姓名 _____	日期 _____
费用表	
1. 项目所需的雇员工作时间	<u>20小时</u>
2. 未来现金费用	<u>45 000美元</u>
3. 获利年数	<u>1</u>
4. 利润率	<u>5.25%</u>
5. 进行测试所需费用：	
■ 建立纸上原型及相关文档	<u>2 000美元</u>
■ 记录人员及观察人员酬金 (3位记录人员和2位观察人员，每人工作60小时，35美元/小时)	<u>10 500美元</u>
■ 总计	<u>12 500.00美元</u>

图 3-1 费用表

根据你和 Evan 掌握的数据，计算净现值如下：

$$45\,000 \times (1) / (1 + 0.0525)^1 = 42\,755.34 \text{ (单位：美元)}$$

把净现值代入投资回报率公式：

$$(42\,755.34 - 12\,500.00) / 12\,500.00 = 2.42$$

在后续的团队会议中，将不仅仅是陈述最后的目标清单，也将做出投资回报率的报告以证明这一计划可以盈利。在我们这一计划中，可用性方面投资带来的 242% 的回报率应该可以让 Mike 十分满意。

现在你已经拥有了回报率的计算结果，该怎么处理它呢？你应该将回报率的说明写入书面报告并发送给其他团队成员，并确保报告中包含你上次更新文件的时间。如果需要更新报告，一定要保存旧的报告作为存档并把它放进硬盘或外部存储器（如可擦写光盘）上用于备份的目录中。这样你便拥有一个可追踪的记录，包括了所有对文件所做的改动。如果你使用微软的 Word 程序，另一种记录文件变化方式是采用修订功能。

另外，当你分发这些文档变更给团队时，需确保每位成员都获得最新版本的网上文件，比如说可以通过在网络上创建一个只有团队成员可以访问的文件夹这一方式。对于每一份给团队成员的打印报告，你和 Evan 都需要保留副本以便当你掌握团队的点滴细节时，同时在因为某些原因无法访问网络版本的时候你可以很容易地查看参考资料。

在下一章中你将了解如何在这一项目中应用纸上原型。

3.8 小结

这一章讲述了为良好的用户设计、可用性设计、可用性测试建立商业计划所需的必要步骤。先探讨了开发用户界面时为何要包括可用性研究。

接下来的段落讨论了项目关系人之间的分歧，讨论了在用户界面设计当中哪些人才是项目关系人员。这一节还讨论各项目关系人对于界面的认知以及对良好界面设计、可用性设计和测试所带来利益的不同期望。

然后我们讨论了如何建立一个商业计划框架以及建立商业计划相关的用户体验 5 个步骤。你现在已经知道要先从商业目标开始，然后是客户目标。由合适的项目团队成员设计用户界面或文档，然后整个团队参与测试的过程以获取反馈。如果测试后所有测试者的目标都能达到，客户目标和商业目标才有可能实现。

在讨论盈利的那一部分中，列举了 8 个准则，它们能帮你在可用性研究的争论中击败怀疑者，而且能提高公司的利润。其中首当其冲且最重要的是，促使用户界面及可用性的设计开发与商业计划案例的关系紧密起来。你还需要让团队成员患难与共，并随时与他们沟通。你和你的团队需要了解客户的需求、任务和目标。从这些信息中，你可以创建具备延展性的用户体验，只添加同时满足用户与客户双方价值的功能。达到预计设计目标之后，需要找个人来负责测试该设计的成功与否。

对一个拥有良好用户界面设计、可用性设计、可用性测试的合理商业计划而言，如何计算接下来的可用性研究的投资回报率是至关重要的部分。你已经了解了如何使用净现值公式来计算投资回报率，接着你可以把得到的数据通知项目关系人，让可用性研究摆上台面。

本章结束时，讨论了进行中的可用性测试和可用性工程生命周期，后者将前者置于一个严格并持续的流程当中。那么你就可以把这一进行中流程的成本计入其他产品的开发中，如果日后产品发布的话也同样处理。

复习题

在阅读第 4 章之前，现在到了温习你所学知识的时候了。回答下面的问题，并查看附录 A 来检查是否答对。

1. 为什么说用户目标的满意度非常重要？
2. 项目中有哪 4 种项目关系人？
3. 好的设计能带来什么益处？
4. 为什么在项目设计之初要同时着手可用性设计过程？

5. 启动商业计划之前应该首先讨论什么主题?
6. 如何保证客户目标与用户体验相一致?
7. 应该在什么时候给产品添加特性和功能?
8. 向项目关系人展示良好设计和可用性设计测试能够降低成本之后, 还需要向他们证明什么?
9. 为什么要研究投资回报率?
10. 为什么要使用可用性工程生命周期?
11. 可用性工程生命周期分为哪3个阶段?
12. 为什么必须在开发过程中收集用户反馈信息?



优良设计

“与其花时间解释错在什么地方，不如一开始就把事情弄对。”

——H. W. Longfellow

本章内容提要

- 优良设计目标
- 设计师与用户对立吗？
- 纸上原型设计与故事版
- 优良文档设计

当你着手分析一个商业案例时，投资方可能会问你优良设计是什么。那么，优良设计到底意味着什么，投资方们为什么要关心优良设计呢？

通常，用户对产品的期望和约束与设计师的实际经验是不同的。用户和设计师之间的这种断层需要项目团队在进行用户界面设计时来弥补和融合。优良设计有4个基本目标是您在为用户进行界面设计时应该始终牢记在心的：合乎常理、目的明确、注重实效和设计优雅（本章将会在后面分别对这4个目标进行详细的讨论）。在进行用户界面设计时采用优良设计的这4个目标是解决设计师与用户之间上述矛盾的有效方法。

一种能够设计出满足用户需求的用户界面的有效的设计方法——应用纸上原型设计和故事版，它使你 and 团队在讨论设计方案时能够方便地模仿真实的用户交互情况。纸上原型设计是一种典型的可用性测试形式，它可以在您和团队之间进行，也可以在您和您的特定目标用户之间进行。尽管纸上原型设计可以让您快速地将您的设计思想与团队进行沟通和交流，但它并不是适合任意设计实际的工具。本章将会阐述什么是纸上原型设计，并讨论什么情况下使用纸上原型设计是最合适的。

本章最后将讨论优良文档设计，这也是保证产品成功的另一个重要目标。当今的在线帮助系统中都提供了海量的文档。进行优良设计，使用户能够获得的在线帮助或者其他任意形式的帮助，应该成为一个公司客户服务和支持的第一线。读者将在本章中学习，在将产品正式向公众发布之前如何进行文档开发及对其的严格检查。

本章结束时对读者在第3章中学到的优良设计方法进行了强化，同时也解释了为什么优良设计并不是随着开发第一个版本产品的结束而结束。实际上，如果您计划为产品开发后续版本，要想不被客户抛弃就必须始终坚持优良设计。

4.1 优良设计目标

Robert Reimann, Hugh Dubberly, Kim Goodwin, David Fore 和 Jonathan Korman 在较高的层面对普遍意义上的设计工作给出4个设计目标(Cooper 和 Reimann, 2003)。无论读者是设计软件应用程序还是设计家具，这些设计目标都是非常重要的。也就是说，不管您设计的是什么产品，只要坚持这4个设计目标，您所设计的产品就更容易被他人接受：

- 设计必须是合乎常理的。
- 设计必须是目的明确的。
- 设计必须是注重实效的。
- 设计必须是优雅的。

那么如何在界面设计中实现这些目标？Cooper 和 Reimann(2003)对此给出了如下的一些建议，其中也包括了笔者新添加的：

- **合乎常理的**——用户界面设计应该是无害的，即界面设计不能让用户比现在生活得

更困难。您应该努力开发能实际改善用户生活的用户界面。

举例而言，界面上应该尽量减少无用的、可能会转移用户注意力信息，否则会使用户降低完成任务的效率。

- **目标明确的**——用户界面应该在用户使用软件应用程序过程中帮助其完成他们的目标。目标明确不仅仅意味着能协助用户达成目标，还包括理解用户的局限性并尽可能地增强用户的能力。您可以通过理解用户的行为方式来了解用户的目标，关于这方面的内容我们将在第5章“用户行为”中进行讨论。
- **注重实效的**——开发用户界面应该尽可能早地满足项目关系人们的要求，这点我们在第3章讨论过。记住这一点并能定期与所有项目关系人们——包括用户、工程师、销售商和经理们——就他们的需求进行讨论，那么就很有可能设计出满足所有项目关系人要求的用户界面。当然，要克服用户和设计师之间的隔阂是需要时间的，我们将在后面深入讨论这个问题。
- **优雅的**——用户界面设计必须尽可能地有效。换句话说，如果界面上的某个功能需要用户点击两次才能完成，那么就试着将其简化为一次点击就可生效。特别是对于网站设计而言更是如此，访问者可能会因为必须不停的点击来进行浏览而失去对该网站的兴趣。优雅的界面设计也意味着所有的界面元素看起来应该是一个有机的整体，而不能简单地杂乱无章地堆砌在一起，因为不统一的界面设计可能会让用户感到混淆和挫折。本章后面的小节将会讨论在进行实际产品开发之前，如何利用纸上原型设计和故事版，帮助您在纸上设计出优雅统一的用户界面。

本书的第7章“设计用户界面”将利用实际案例来分析如何将这些优良设计原则和模式应用到实际设计过程中。

4.2 设计师与用户对立吗

对于任何包含用户设计的产品设计来说，设计师和用户之间有着根本上不同的目标。比如，实际设计产品的设计师是与整个开发过程紧密相连的。另一方面，用户直到产品完成之后才会与产品发生联系，他们可能并不同意设计师对产品设计的判断。当然，最终还是用户决定是否花钱来购买设计师设计的产品。

设计师和用户之间这些根本性不同会导致两者之间产生分歧，而只有注意并且克服了这些分歧才能设计出优良的用户界面。在开始设计过程之前，项目团队必须认识到用户和设计师各自的局限性，这样才能在设计过程中有效地避免和克服。反之，如果不这样的话，那么整个项目就有可能失败。

4.2.1 用户限制

用户会面临来自真是世界的一系列限制和约束(Norman, 2002)。这些限制可以被分为5大类：

- **物理的**——例如，用户只能向下坐到椅子上。
- **语义的**——实际情况的含义导致的局限。举例来说，轿车中驾驶员的位置只能被放置到驾驶员能够驾驶车辆的特定区域，这样他才能踩到踏板、把握方向盘、控制传动装置等等。
- **文化的**——举例来说，如果要为国际市场设计一辆汽车，就需要注重不同国家驾驶习惯带来的问题，比如有的国家要求驾驶员沿着道路的左侧行驶，而有点国家则要求驾驶员沿着道路的右侧行驶。
- **逻辑的**——举例而言，我父亲和我最近使用成套材料为我母亲的手工房添置了一张手工桌，我自然地认为桌子是由成套材料按照一定的顺序组装起来的，桌子成套材料的所有元件都会被用到。
- **社会的**——这个局限性可能是因为不愿意在社会公众场合表现的像个哑巴而不承认错误或者不在他人面前承认自己不知道，而带来的压力。

4.2.2 设计师限制

设计师有他们自己需要面对的局限，这些限制导致了设计师与用户关注对象的不同。设计师的局限性包括以下几点：

- **时间**——设计师必须在有限的时间内把产品设计出来，这个时间限制可能是用户期望的，也可能是公司内部时间表所规定的。
- **个性**——我们每个人都希望有自己的个性，设计师进行产品设计时也不例外。毫无疑问的是，为设计师自己考虑要比为其他人考虑产品设计要简单。因此，设计师内

心总想设计出他自己感到美观舒服的产品。

- **因时间和个性带来的压力**——设计师可能会接到公司内部各种各样的指示和压力，比如要求设计师设计诸如“下一个 iPod”等能给公司带来声誉和奖项的产品，而不是设计师本人觉得可能会更实用或功能更强的产品。
- **针对不同用户基础**——设计师设计的产品可能并不是直接面向外部用户的，而是为了满足公司内部如工程部门的需求而只针对内部用户。设计师也有可能只考虑一部分客户，即使这个产品有可能会被客户普遍使用。
- **设计师自认是一个典型用户**——最大的局限在于设计师认为他自己就是产品的典型用户。因此，设计师往往会按照他自己觉得最好的方式来设计产品。设计师没有想到或者不愿认识到，对于产品现在和潜在客户而言，设计师本人并不能决定一个产品可用性到底如何。

4.2.3 解决矛盾

理想的解决方案是在设计师开始进行设计工作之前，就把所有的问题梳理出来并想办法解决掉。显然，这个方法并不总是切实可行的。举例而言，设计师可能会遇到这样的情况：由于产品上市时间的限制，产品必须在某个时间点之前完成，同时也没有什么可以说服人们改变他们这种要求的理由。还有可能会牵涉公司行政上的问题，比如高层领导会因为职位的原因用他们的设想和指示来影响产品设计。

如果目前的开发设计流程不会有什么变化，而您正好处于这样的环境中，最好的办法是尽可能多地收集用户关于产品的反馈意见。这个工作您可以亲自去做，也可以与其他相关部门，如市场部等同事进行交谈，从中获得当前的和预期用户对产品的意见。

收集的信息越多，您就越能判断您设计的产品是否真正符合用户的需要。如果结果是不满足用户需要，那么您就能在产品升级过程中或开发新产品时做好更充分的准备。

4.3 纸上原型设计和故事板

读者可能或多或少听说过电影工作室利用故事板来对准备拍摄的不同场景镜头进行预演，特别是动画电影更是如此。人们把构思画在纸上，然后这些纸片按照一定的顺序摆放

好，用以展现一个分镜头脚本，这就是故事板。使用故事板可以给制片人一个在电影正式拍摄成胶片之前事先检查镜头构思的机会，从而有改进电影拍摄的可能。

在用户界面设计领域，也有一个类似的但更具有交互性的故事板，也就是所谓的纸上原型设计。纸上原型设计的实质是为软件应用程序、硬件产品或者网站制作纸面上的版本，这样设计师就有机会在着手设计产品之前，了解用户是如何与产品进行交互的。在纸上进行设计和修改是比较简单的，甚至可以给设计师更大的设计弹性，比如设计师可以很方便地在纸上根据用户的意见将某个对象从一个位置移动到另一个位置。

本节中的很多内容出自 Carolyn Snyder 在 2003 年出版的《Paper Prototyping》(纸上原型设计)一书。除了书中提到的问题外，我也加入了一些可能在纸上原型设计过程中会遇到的其他问题，比如可及性和不同的用户群需求问题等。这些不同的需求可能会要求你必须进行多次不同的纸上原型测试，甚至是决定完全不需要纸上原型测试。

提示：

读者可以上 Carolyn Snyder 的网站了解更多纸上原型设计的信息，网址是 www.paper-prototyping.com。

纸上原型设计给了您和您的设计团队一个几乎不必花钱就能对设计进行测试的机会。这个过程中您可能会遇到怀疑论者，这一节中我们就来讨论如何来消除这些怀疑。

4.3.1 纸上原型设计的概念

Snyder(2003)用这样的定义来说明纸上原型设计是可用性测试的一种形式，即典型用户通过与纸上的用户界面进行交互来完成真实的任务。这个界面可以是任何形式的，包括用户与计算机之间的交互界面，或者是用户与硬件产品之间的交互界面。

这种测试是可控的和交互的。既可以同时对多个用户进行测试，也可以专门对特定用户进行测试。做这种测试时，不需要向用户解释测试的用户界面的设计意图，尽可能地模拟用户使用产品时的真实情况。

纸上用户界面可以是手绘的，也可以是打印的有增删的屏幕截图，甚至是在一个硬件产品界面上贴上几个纸质按钮来模拟新的功能界面。这时需要测试团队中的一名成员来扮

演计算机或者硬件产品的角色，这样一来当用户对纸上用户界面进行交互时他就可以提供相应的反馈。

与纸上原型设计类似，只是交互程度上有所不同的测试方法还包括：

- 为将要设计的产品制作仿制品(实体模型)，可以是在纸上，也可以是在一个图形文件中。举例来说，设计师可以为网站的不同元素和网页制作演示样本。这些仿制品完全是静态的，也就是说，你只能看，而不能与其进行交互。
- 原型框架，可以是网页或者软件窗口的布局框架，用以展示文本、图片、链接、按钮以及其他元素的摆放位置。原型框架中还可以包含能链接到其他原型框架或窗口的活动链接，这样就提供了一种更具交互性的方式来说明网页或者窗口是如何组织在一起的。
- 故事板，正如在本节的介绍部分讨论的那样，你可以制作不同的纸片并将它们按照一定的规则联系在一起，可以是线性的，也可以是树状结构的。当然，故事板也是一种典型的静态工具。

4.3.2 克服怀疑态度

项目关系人，诸如项目经理和设计师可能会想，为什么必须进行纸上原型设计来获得用户的反馈。Snyder(2003)在她任教的班上讨论了这个关于纸上原型设计的问题，问题主要集中在以下4点：

- **有效性**——通过纸上原型设计是否能够发现我们希望发现的产品设计问题？有没有纸上原型设计无法解决的问题而必须配合其他不同的测试？
- **偏差性**——纸上原型设计会不会因为我们不能信任测试结果而改变用户行为？用户在计算机上进行设计测试后的行为会不会有很大的不同？
- **专业性**——纸上原型设计会不会显得项目团队或者公司不专业？
- **资源问题**——花费时间、投资和各种资源进行纸上原型设计的回报是什么？

持怀疑态度者并不是为了让你难堪。相反，您应该尊重他们，因为他们是诚心诚意地给你提出意见和问题。但是，您也要做好准备应对他们的问题并减轻他们的顾虑。

Snyder (2003) 对此提出了以下 4 种建议来回应对纸上原型设计持怀疑态度的人:

- 用一个实例向人们展示纸上原型设计的过程是如何进行的。当人们了解了纸上原型设计的整个过程之后, 他们就可能更加理解这种测试是如何起作用的。那些持怀疑态度者看了原型设计后, 可能会给你一些甚至能改善原型设计的反馈意见。当然, 你也需要预先知道一些纸上原型设计的缺陷, 并提供一些附加的测试来解决那些纸上原型设计无法解决的问题。
- 让公司中有影响力的人来充当模拟用户进行纸上原型设计。通过与这些有一定影响力的内部用户进行纸上原型设计, 可以让他们获得关于纸上原型设计好处的第一手体验。向这些人询问原型设计的反馈意见, 从而可以在进行实际测试之前改变产品的原型设计。通过预先对纸上原型设计进行测试, 你可以确认有偏差的问题, 同时保证纸上原型设计能够满足设计目标。
- 向公司相关部门寻求支持, 并从这些部门中邀请参加纸上原型设计的人员。例如, 公司的销售部门和市场部门通常对设计测试结果感兴趣, 并愿意作为测试对象或者观察者参与到纸上原型设计过程之中。那些对纸上原型设计感兴趣的人会让你更直接的反馈信息, 并能协助你提供最好的投资回报数据来说服持怀疑态度者。
- 对于那些在纸上原型设计测试完成后仍然对这种测试的好处表示怀疑的反对意见者, 在得到原型测试结论后还应该向他们询问反馈意见, 从而减小他们在这方面的顾虑。举例来说, 为了让纸上原型设计看起来更专业, 可以使用更厚重的纸张或者纸板来操作, 以保证其在测试过程中更加结实、耐用。

4.3.3 纸上原型设计优点

在纸上原型设计过程中, 当你获得反馈意见并处理后, 一定要向公司投资方进行汇报, 并诚恳地告知他们纸上原型设计的优势和劣势。

1. 激发更好的规划

纸上原型设计最大的优势在于设计师能够更确切地知道用户是如何使用产品的。但是, 设计师在设计原型测试时也必须知道产品目标用户分布, 因为可能影响界面使用的因素有很多 (Thatcher 等, 2002, Brick 等, 2002), 主要包括如下一些:

- **有身体功能障碍的用户需求**——在一般人群中 有 15% ~ 30% 的人口，即超过 5000 万人有这样或那样的功能限制，这些身体上的功能障碍会对高科技产品的使用产生影响。
- **不同的市场细分可能有不同的需求**——举例来说，美国那些在所谓“婴儿潮”中出生的人现在大多都逐渐步入退休年龄。尽管他们都可能懂计算机，但也有可能因为视力问题等身体机能限制而影响设计师对产品的设计。
- **文化问题**——如果设计师设计的产品不止会在一个国家上市，那么就有许多文化问题需要在设计中考虑，比如不同的国家可能使用不同的习俗和标记来表示同一个概念，更不用说使用的语言也是不同的。

如果在进行纸上原型设计测试之前知道这些因素，那么你就能设计出一个基于客户的优良的纸上原型测试。

举例而言，如果你准备开发一个面向多种地域文化的用户界面，那么你可以在纸上使用同一个界面，同时准备好几个独立的带有一小段胶带的纸质按钮，按钮上分别绘制着不同的符号。当你针对不同文化背景的用户进行纸上原型设计时，你就可以把当前的按钮换下来，换上特定符号的按钮，比如标识俄国扬声器的斯拉夫语符号。

2. 解决问题和激发创新

纸上原型设计为设计师提供了切实可信的优势 (Snyder, 2003)。使用纸上原型设计能够在产品开发过程早期就获得用户的真实反馈。纸上原型设计也不需要特别的技术技巧，用户可以简单地根据需要使用纸片来完成任务。因此，用户会发现这个过程并不复杂和可怕，并乐于给出他们的反馈。

设计师可以邀请不同部门的同事作为观察员或者测试员来参与测试过程。这种在开发过程中的多部门间的交叉与交流，反过来又能促进部门间的协作和意见交换。

纸上原型设计获得的用户反馈也能够帮助设计团队更好地进行设计，因为这个过程增强了开发团队与目标用户之间的交流，从而在一定程度上消除了二者之间的误解，进而能够激发出开发设计中的创造力。举例来说，假设用户发现了一个问题，测试员就会快速地在界面上做出反应，比如删除特定界面或者在纸上的其他位置重新绘制。设计师也可以在纸上做上记号以便日后进行深入的讨论。

3. 效果

公司投资方可能想从设计师处确认，纸上原型设计到底能在产品设计过程中起到什么作用。下面列出了一些可以从纸上原型设计中发现的产品设计问题(Snyder, 2003)：

- 不清晰的概念和专用术语。
- 导航、工作流程或任务流程的问题。
- 能够改进的内容问题，可以使设计更优良以传达正确的信息。
- 用户希望发现和使用产品文档的途径。
- 功能性问题。
- 硬件产品屏幕上包括文字、窗口和按钮在内的界面对象布局。
- 产品的使用效率。如果产品不便于使用，那么它一定不是高效的产品。

4.3.4 纸上原型设计缺点

纸上原型设计并不是适合于所有情况。某些情况下，可能需要进一步的测试或调用另一种不同类型的可用性测试。

1. 不同的测试

如果出现下列情况，那么你就需要考虑使用其他可用性测试方法而不能只依赖于纸上原型设计了(Snyder, 2003)：

- 项目团队的人数有限，或者公司里没有足够的职员进行纸上原型设计。
- 每个团队成员都熟悉产品所使用的技术。
- 开发团队可以远程对产品进行测试。
- 开发团队可以在产品设计初期方便地创建其他形式的可用性测试环境。比如，设计师可以在公司的训练室里测试一个软件应用程序的升级版本，当然前提是有可以方便安装升级版本的软件环境。
- 开发团队可以控制所有用户看到的内容。
- 用户使用产品不需要持续很长时间(最多1分钟)。
- 可用性测试可以等到产品开发中期或者后期进行。例如，设计团队的任务是负责软件升级，那么升级的测试就不需要在软件产品开发的早期进行。

2. 进一步测试

需要指出的是，纸上原型设计毕竟是属于一种静态测试方法，你不能期望一些诸如滚动条等动态用户界面特性能在纸上原型设计中重现出来。你有义务向项目关系人声明这一点，即纸上原型设计并不能取代真正的软件产品、硬件产品或者网站的测试。换句话说，纸上原型设计只能作为第一阶段开发工具。

当你向项目关系人报告测试结果时，你需要向他们说明还有其他什么问题需要进行进一步的可用性测试才能确定 (Synder, 2003)。这些问题包括：

- 滚动条和滚轮问题。纸上原型设计无法表现动态界面元素的反应，比如用户下拉一个列表，或者当用户将鼠标移动至按钮上时所发生的情况。
- 长的文档和列表。比如，如果你在界面上提供了一个包含很多可选项的长列表，只有等到在软件、硬件或网站最终完成后才能让用户对列表进行测试。
- 键盘和鼠标偏好设定。比如，用户可能有不同的组合快捷键设定偏好，如按下 **Ctrl + S** 保存文档等。
- 需要较长下载时间，屏幕需要较长的加载时间，或者一个功能需要较长时间才能完成。比如说，一个网站需要较长时间才能显示出来，你是无法通过纸上原型设计进行重现的。

4.4 优良的文档设计

我们在第3章中提到，文档对于产品来说不仅是完整用户体验的一部分，而且也是为公司产品客服前线的一部分。用户在使用产品的过程中遇到问题，首先想到的便是求助于产品附带的帮助文档。如果用户找不到帮助文档，那么他们就只能拨打公司服务热线来寻求帮助。对于一个高科技公司而言，由于帮助文档的问题而带来的客户服务支持费用可能占到整个公司服务支持费用的60%。

不幸的是，在关于文档到底对可用性测试起什么作用的争论中，你的开发团队经常会成为牺牲品。这种争论通常是这样的：

你：“我们需要为我们的软件应用程序编写帮助文档，这样用户就知道如何使用了。”

开发人员：“但是我们的软件是这么的简单，根本不需要帮助文档。”

如同可用性测试的情形，这种对文档必要性的反驳是很难轻松绕过的，因为开发人员的说法也是对的——如果开发团队成员是唯一使用软件应用程序的人。当然，这也是你答案的一部分：文档是为其他普通用户编写的。

制订文档计划

另一个有力的证据是拥有良好的产品文档可以带来良好的经济效益，但这并不是足够的。你还需要制订出完整的文档开发计划，并做好将计划给项目关系人展示的准备。当你正式开发文档时，你需要确认开发文档的合作者是谁，以及你需要为用户开发什么内容的文档。正如你准备旅行之前，你需要事先准备好地图和出行计划。对于文档开发过程也是一样，通常包含以下8个步骤：

第一步：组建文档开发团队

在你知道要创造什么之前，你必须先组建团队。团队不仅要有一个或多个技术写手，你还应该知道学科问题专家(SME)是哪些人。许多学科问题专家会在项目团队中，但也有些不在。比如，你可能知道一些并不在项目团队的工程师掌握你所需要的知识，你也有可能去咨询销售和市场部门的同事。

当你了解到一些技术写手或学科问题专家后，你需要通过与他们或者他们的上级交谈等方式，做一些调查以确认他们是否能够帮助你开发产品文档。这样以来，你才能确认他们什么时候才能加入团队，比如，有些人在另外一些项目优先级更高的团队中，了解他们的常用联系方式，以保证他们能尽快地回复各种询问。有的人喜欢电话联系，有的人喜欢电子邮件的形式，而有的人更愿意面对面的交流。

如果你需要的人员由于各种原因无法加入团队，你应该告知其他团队成员或者一些经理，向他们咨询如何克服这个难题。

第二步：为团队列一份文档清单

当你确认了文档团队后，你需要与文档团队成员以及项目团队成员一起列出一份文档清单，用以回答下面2个问题：

- 我们需要创建什么类型的文档？

- 为了创建文档我们需要什么信息？

文档可以有多种形式：

- **纸质文档**——在一篇 2000 年 10 月号《Scientific American》(科学美国人)的文章中，Steve Mirsky 指出相对于在线帮助文档而言，人们阅读纸质文档更容易，因为阅读在线文档需要不同的阅读技巧。但是，对于年轻一代而言，他们对在线文档和纸质文档的态度却与上代不同，因为他们是随着互联网长大的一代，长期的在线阅读经验让他们更能适应在线文档。

纸质文档的优点在于用户可以拿在手中阅读，如图 4-1 所示。当计算机不在身边时，纸质文档的优势就比较明显了，也有些用户习惯拿着诸如高亮记号笔等实际交互工具进行阅读，因此，他们也更喜欢纸质文档。而且，对于硬件产品而言，纸质文档通常是唯一媒介选择。纸质文档的缺点是成本高。纸的价格一直在上升，因此印刷纸质文档的成本也会不断升高。

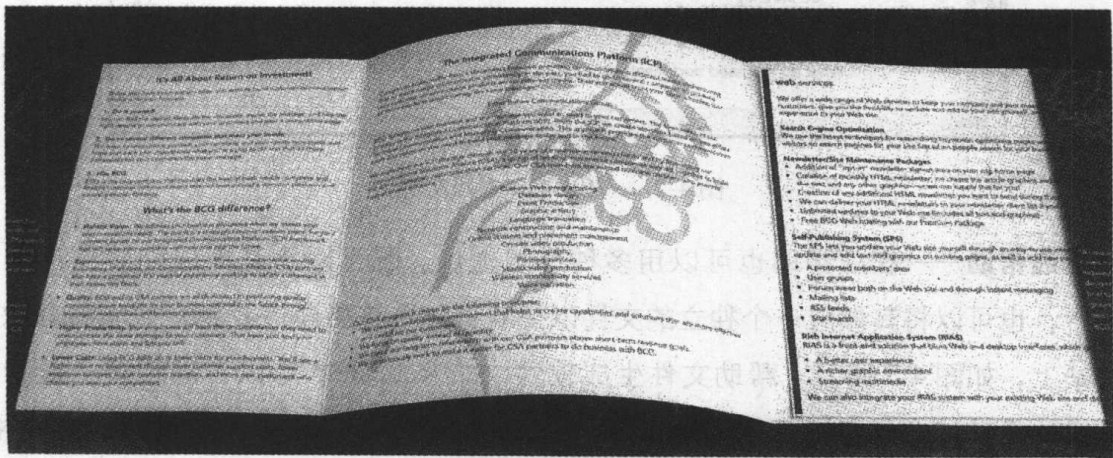


图 4-1 一份打印文档实例

- **便携文档格式 (PDF) 文件**——PDF 是共享、显示和印刷格式文档的实际标准文件格式。Adobe System 开发了 PDF，并维护着这个标准。Adobe 允许用户免费下载 Adobe Reader 阅读器软件，并提供了各种操作平台的版本，包括 Windows、Mac OS、Linux 以及诸如 Palm OS 等主要的手持 PC 操作系统。

PDF 格式的文档看起来就像打印出来的文档一样，不用排版等繁琐操作，用户可以方便地使用打印机直接打印出来，如图 4-2 所示。PDF 文件也是便携的，你可以把

PDF 文件放到软件产品发布光盘中，也可以把它们放到公司的网站上供用户下载。

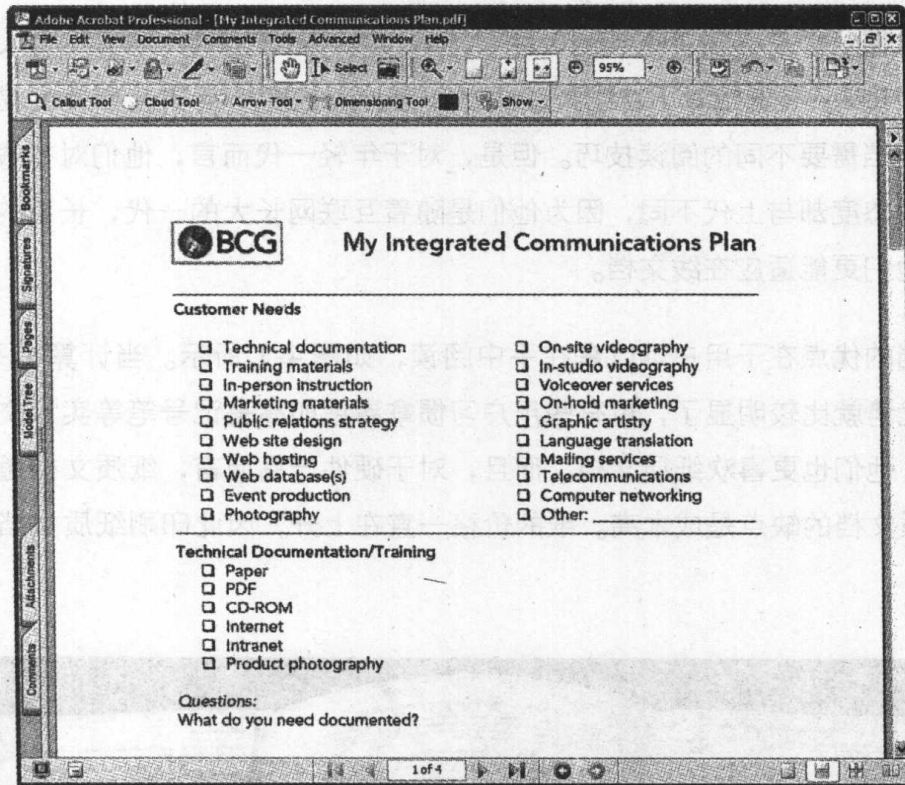


图 4-2 一份 PDF 文档实例

- 在线帮助**——在线帮助也可以用多种形式提供。既可以将其集成到软件应用程序中，也可以将其做成一个独立的文件以便从操作系统桌面上找到，或者将其放到网站上，如图 4-3 所示。帮助文件生成软件，如 RoboHelp 可以自动创建一般帮助系统并将其转换成包括基于 HTML 的 WebHelp 格式在内的多种在线帮助文件格式，在任意操作系统上使用任何一个现代网页浏览器都可以方便查看。你也可以为在线帮助系统创建帮助台支持模块，以供客户服务部门使用。

在线帮助的缺点在于人们不容易找到他们急需的特定帮助信息。许多公司也都没有认识到在线帮助实际上有着不同的设计挑战，他们仅是简单地用在线帮助的形式创建了一个用户手册。而且，与 PDF 文档不同，对于那些喜欢阅读纸质文档的用户而言，在线帮助通常不方便直接打印。

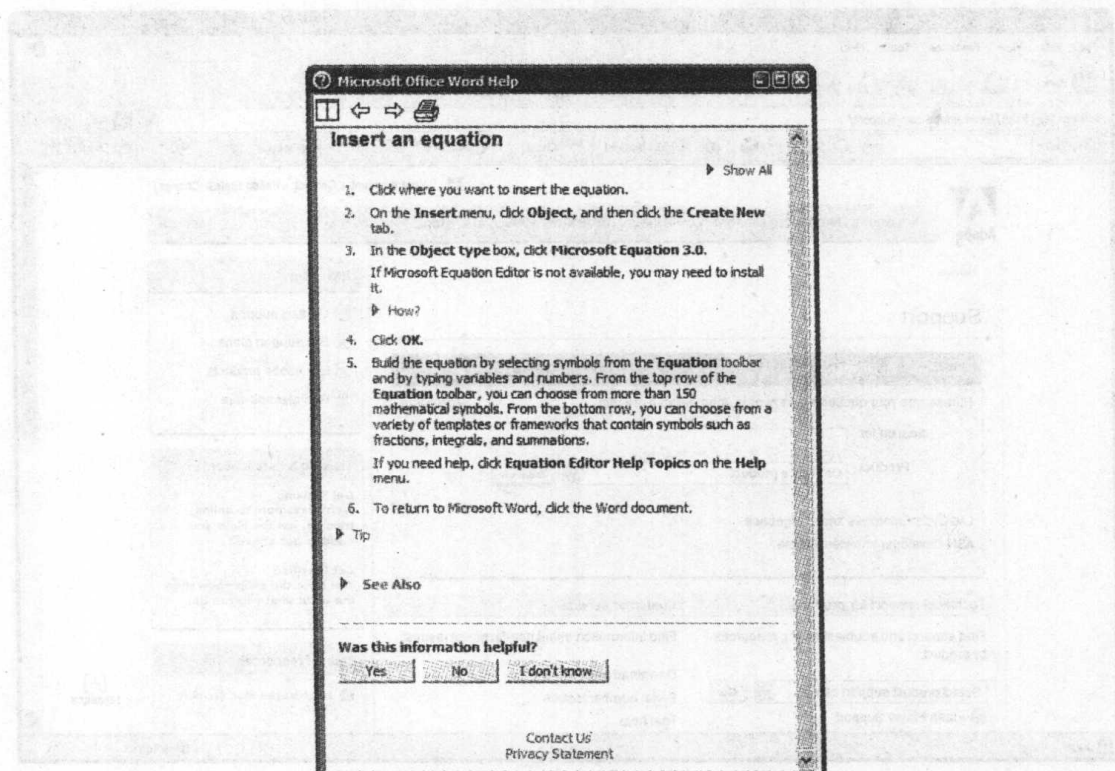


图 4-3 一个在线帮助窗口实例

- **网站**——网站可以是面向公众的，也可以是一个需要输入用户 ID 和密码的私人网站，或者是只为公司内部员工服务的企业内部网站。许多公司网站，比如图 4-4 所示的 Adobe 产品支持网站包含了许多客户服务帮助支持信息，包括文档文件、技术支持问题、经常被问到的问题 (FAQs) 等。公司试图通过网站替代客户服务，但其缺点在于如果用户无法在网站上找到想要知道的问题答案，那么他就可能会觉得购买你的产品是在花冤枉钱。

知道用户需要什么类型的文档是非常重要的。为了做到这一点，你需要尽可能多地访问用户。

第三步：经常访问用户

为了满足用户的需求，你可能需要创建不同类型的文档。举例来说，文档可能包括一本打印的快速入门指南、内置在软件应用程序帮助菜单里的在线帮助、可以打印的 PDF 格式文档文件，以及相关的多媒体教学仿真程序等。

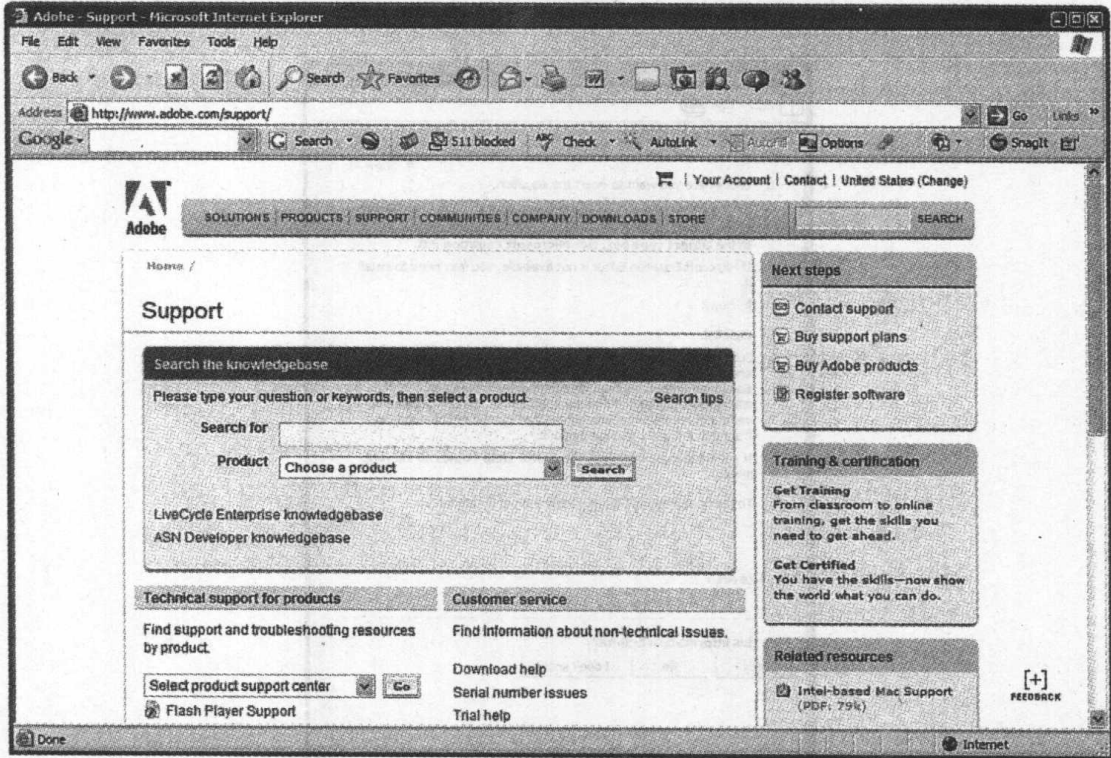


图 4-4 Adobe 产品支持网站

但是，你只有了解了用户真正需要的是什么，你才能知道要为产品制作什么类型的文档。在确定用户需要什么之前，你还需要了解软件、硬件或者网站用户的经验水平如何。本书的第6章“用户分析”将针对用户经验水平问题进行讨论。

在产品设计和开发过程中，你可能希望有用户一直在不同阶段进行测试并提供反馈意见。在软件开发行业中，这些开发阶段的测试用户被称为 *Beta 测试人员*。你可以利用这些测试人员对开发中的文档进行评论并反馈意见。这些测试人员提供的反馈对文档开发来说也是非常重要的，这样才能保证在向公众正式发布之前改善文档的质量。比如，你可以咨询测试人员文档中需要多少图片和屏幕截图，如何在文档中表示信息，以及用户在文档中查询信息的难易程度如何。

第四步：制订文档样式和格式规范

当你确定了需要制作什么内容的文档后，你必须着手制订文档编写样式和格式规范。一份严谨的文档编写样式和格式规范对于文档团队和用户都是有帮助的。一份定制的文档样式和格式规范可以帮助文档团队和学科问题专家明白文档结构和内容的组织方式。用户也会从阅读一个风格、语法、格式、布局统一的文档中获益。公司可能已经有一套完整的

文档样式和格式规范，那么你也可以沿用公司的文档编写规范。

第五步：制订文档大纲

当你确定了文档编写样式和格式规范后，你就需要为文档每个组成部分制订一个大綱，包括在线文档大纲、纸质文档大纲和教学模块大纲等。把准备好的大纲送给学科问题专家、销售和市场部门的同事们，以及其他专业技术写手传阅，尽可能多地收集他们的反馈意见。文档大纲高度概括了文档每个部分的主题，在根据收到的反馈意见修改完大纲后，还需要把更新过的大纲送给提出意见的人士，让他们做最后的确认。

第六步：草拟文档目录

文档大纲一旦确定，就需要严格按照大纲草拟出文档目录。在文档目录中，你需要细化文档大纲的内容至每个小节。最好在文档中加入术语表部分，对于 PDF 文件和在线帮助还要加入索引。你还能在文档中加入附录，编写一些诸如“常见问题问答”等内容，以便读者参考。文档目录草拟完成后，别忘了也送给项目关系人审阅。

第七步：收集必要的信息

在编写文档的过程中，你不可避免地会求助于学科问题专家来咨询一些技术问题。如果你在第一步中获得了专家们的常用联系方式，那么咨询就能顺利进行，否则就很难说了。在收集到关于文档编写的必要信息后，你很可能会对文档目录进行进一步的完善和修改，以达到最好的文档效果。

第八步：彻底审查文档

对于没有经过审查的文档，用户一眼就能发现。因此，一个严格有序的文档审查过程对于文档质量的改善是非常重要的。文档审查团队应该包括项目团队成员、至少一名项目团队之外的成员(比如某个销售经理等)，以及 Beta 测试人员等。

在文档开发的不同阶段，定期对文档进行审查可以尽可能多地发现诸如准确性、格式、语法和信息量等问题。你需要一份打印的或在线的审查文档的拷贝，以便于审查人员在相应的地方提出建议并做上记号。在文档审查的过程中，一定要遵守严格的时间限制，以保证文档能按时、保质、保量的完成。

4.5 为什么要关注良好的设计

在第 3 章中，我们讨论了必须关注良好的设计的一些商业原因。总体来说，这些商业

原因可以归纳为以下3点：

- **节省解决由于设计而导致的问题的不必要的花费**——这些问题不仅包括用户向客户服务部门询问如何使用产品，还可能包括用户不正确地使用产品，从而带来更加严重的问题。
- **让用户相信应该使用产品**——用户决定是否使用软件产品。即使用户需要在他们的工作环境中使用一款软件产品，软件的可用性对于用户决定是否继续使用软件产品、硬件产品或者是网站是十分有帮助的。
- **保留现有用户，吸引新客户**——如果使用产品能够帮助用户解决实际问题，那么用户就会认为公司值得信赖，并对公司的产品更有信心。反之，如果公司产品不能帮助用户，那么用户就可能会让其他人知道产品不够好，进而影响整个公司产品的口碑。

现在，互联网上的各类网站和博客使得无论是好消息还是坏信息传播都要比以前任何时候容易得多，这其中就包括用户对产品和服务的各种评价。当你在诸如 **Blogger**、**WordPress** 和 **Myspace** 等知名博客网站上写博客时，你就在与成千上万的其他用户分享观点。

这些优良设计原则并不仅只适用于你设计的软件、硬件或网站的第一版产品。如果公司生产软件和网站产品，就必须不断升级产品，根据竞争对手的情况为产品添加新的功能，进而避免给用户留下产品和公司不思进取的印象。

但是，对下一版本的好想法可能会让产品走弯路。有没有这样的经验：你本想为了产品有更好的用户体验而升级软件到一个新的版本，却发现某些原有的功能不按照原来的方式起作用了，甚至根本就找不到了？你不仅为第一版产品坚持良好的设计和设计目标，而且还要考虑到其后续版本。良好的设计不仅包括软件、硬件产品和网站本身，还包括你为产品创建的文档。只要设计能完全满足上面提到的3个指导原则，你和公司一定会因此受益的。

4.6 案例研究：创建一个纸上原型设计测试

因为投资回报书已经完成，**Mike** 给了你一个进行可用性测试的任务，从更新现在的数据库应用程序中的信息开始。**Mike** 已经决定先着手更新现在的数据库应用程序，以便在他网站上向客户发布新功能之前事先发现程序内部问题。

因此，现在就是你和 Evan 通过纸上原型设计对数据库应用程序用户界面的改变进行测试的时候了。

Evan 在附近的书店里买了 Susan Snyder 的《纸上原型设计》一书，从书中学到了更多关于进行纸上原型设计测试的信息，包括完成测试所需要的材料和步骤等实用内容。

你和 Evan 决定从附近的办公用品商店里购买下列办公用品：

- 白色海报板，用来当作原型设计参与者放置其他原型设计元素的固定背景。
- 空白纸，用来绘制较大的原型设计图片和做一些笔记。
- 标准索引卡片，用来代表较小的诸如对话框和菜单等原型界面元素。可以买几张 4×6 英寸和 5×7 英寸的索引卡片，以备将其切分成更小的卡片或者需要在卡片上记录时使用。
- 马克笔和钢笔，用来手绘诸如新的按钮等原型界面元素。
- 荧光笔，用于在屏幕上高亮显示某些界面元素。
- 剪刀，用来将较大的纸和卡片剪切成较小的。
- 可重用的胶带，用来将用户界面原型设计卡片贴到背景白板上，并能根据需要方便地移动它们的位置。
- 可移动的纸带，用来代表诸如文字的改变、按钮状态的改变和列表选项等用户界面状态。
- 透明幻灯片，用来通过分层叠加的方式在不改变原来设计样式的前提下往界面上添加新的元素，这对于有大量界面区域需要设计的情况特别有用，这样可以避免使用大量的纸带。
- 幻灯胶片笔，用于在幻灯胶片上写字画图。
- 可以擦拭幻灯胶片的纸巾或布。

现在一个有利的情况是，你和 Evan 可以把现在的用户界面屏幕截图保存下来，并将图片放大打印到一张纸上。这些屏幕截图可以作为当前用户界面的一个参考基础，你和 Evan 还需要在上面手绘原型设计的部分，如图 4-5 所示。

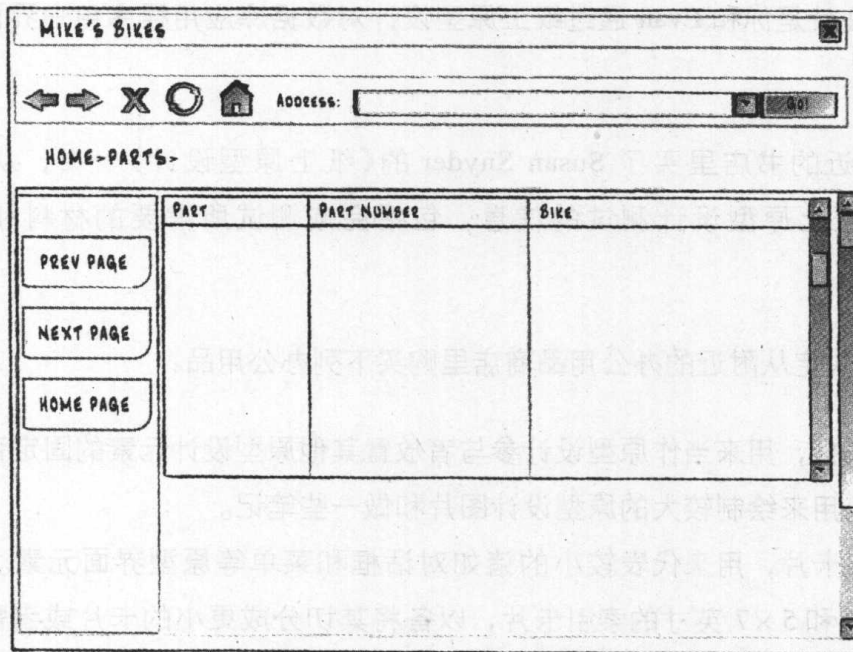


图 4-5 一个屏幕截图替代卡片

正如我们在第 3 章中讨论过的，你和 Evan 通过收集大量的用户反馈信息确定了数据库应用程序升级方案的具体目标。（比如，零件维修页面应该为每个零件的关键状态在界面提供视觉暗示。）每个任务都必须与特定的目标相一致。

每个任务都应该足以让用户完成他们的目标，同时也要保证任务有明确的和可预见的解决方案时间表。对于一个内行来说，任务应该在几分钟内完成。举例来说，在 Mike 的自行车数据库应用程序中，可能有一个只需通过简单的点击按钮就可以完成的在线订购产品的任务。

每个任务都应该按照下面的模板写下来：

- 任务编号和任务名称。
- 任务目标或输出。
- 输入和假设条件。以 Mike 的自行车数据库应用程序为例，你需要列出所有用户完成任务所需的所有实际存在的和可能存在的信息和资源列表，比如登录应用程序所需的用户身份和密码等。
- 完成任务所需的屏幕可见步骤。每个步骤都会让你和 Evan 知道在原型设计时需要为每个屏幕界面准备多少原型界面元素卡片。

- 一个内行完成任务所需的时间。
- 用户完成任务的指令说明。
- 关于任务的其他注意事项，比如进行测试时你和 Evan 需要知道的信息等。

使用这样模板的任务文档如图 4-6 所示。

The image shows a task worksheet template enclosed in a dashed border. At the top, there are two fields labeled 'NAME' and 'DATE'. Below this is a shaded header section titled 'TASK WORKSHEET'. The main body of the form consists of several sections, each with a small square bullet point and a horizontal line for input:

- Task Number/Task Name:
- Task Goals and/or Output:
- Inputs and Assumptions:
- Steps:
- Amount of time:
- Instructions:
- Notes:

图 4-6 一份任务文档实例

任务文档应该尽可能的简练，这样测试人员就能只需关注完成任务所必需的信息，你和 Evan 也能更便捷地查阅任务步骤。

在准备原型测试时，你不仅需要准备空白屏幕，而且还要准备必需的数据。比如，你需要准备一个当用户错误操作时弹出的提示用的对话框。相反，你也要准备写在用户正确操作

后的提示信息框。因为你是正在升级一个已有的应用程序，这样更容易看出使用程序过程中发生的各种错误。**Mike** 给了你和 **Evan** 使用数据库应用程序的权限，以便熟悉其工作流程。

注意纸上原型设计中那些不影响功能性的文字提示，类似这样的界面内容，你可以在卡片上用波浪线代替。

组织一次纸上原型设计可能会导致一些混乱，所以你和 **Evan** 必须首先决定好组织所有纸上原型设计材料的方法。你可以把所有的任务卡片和屏幕卡片使用活页夹整齐地集中在一起以便于检查。活页夹还包括“卡片页”，即贴有一些写有文字的代表界面元素的卡片的纸页，如图 4-7 所示。你和 **Evan** 可能会根据需要将卡片从卡片页上拿下来放到原型设计白板上使用，使用完毕后再贴回活页夹中的“卡片页”。

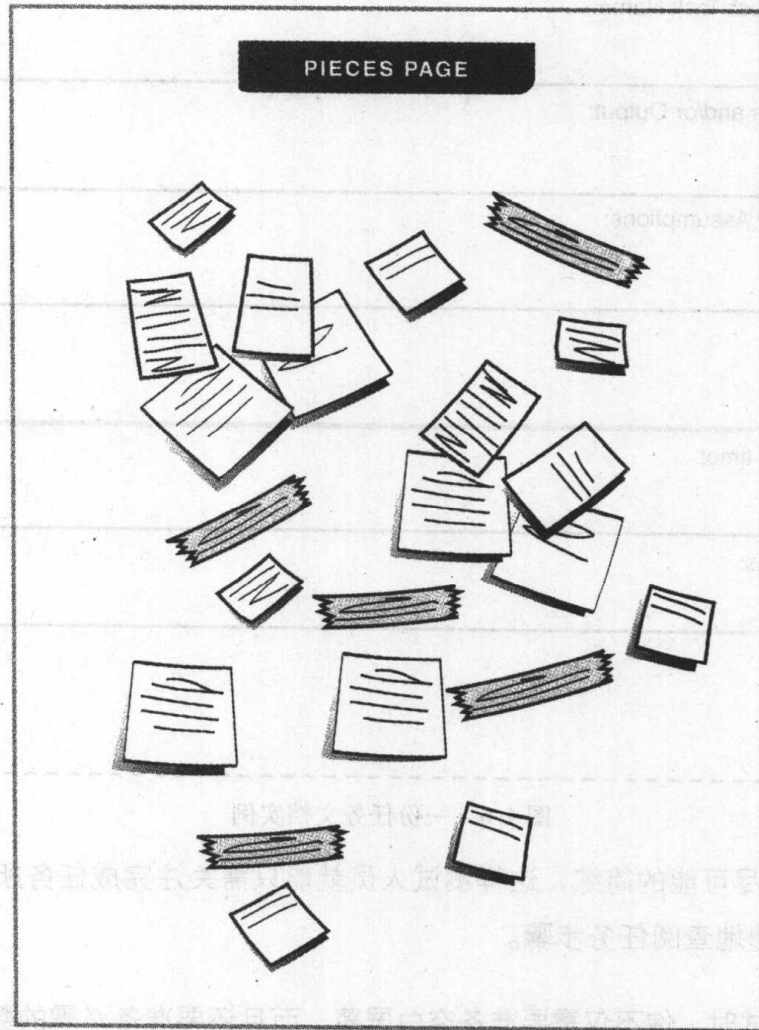


图 4-7 一张卡片页实例

所有 10 名团队成员都会参与到纸上原型设计测试中。在你和 Evan 进行测试排练之前，你应该添加更多关于用户的概念模型，并将其应用到纸上原型测试的任务当中。具体如何进行操作将在下一章中讨论。

4.7 小结

本章首先讨论了优良设计的目标。你必须在用户界面设计中贯彻 4 个优良设计目标：即用户界面必须是合乎常理的、目标明确的、注重实效的和优雅的。进行用户设计的好处包括降低长期产品费用，可以集中精力于产品质量的改善而不是纠缠于因为用户的投诉带来的对产品问题的修补，还可以将这些设计经验应用到公司的其他项目中。

读者了解到在进行用户界面设计过程中，用户和设计师所面临的不同限制及其导致的各自理念上的差异。你应该在设计开始之初尽可能早地弥补这种差异，如果做不到这点，你必须尽可能多地从用户那里收集关于设计师对产品的设计规划是否与用户的期望相吻合的信息。

接下来我们讨论了纸上原型设计和故事板。读者了解了进行纸上原型设计的相关问题，为什么说进行纸上原型设计是在实际开发用户界面之前最有效的测试方法，以及纸上原型设计的局限性。读者也学到了如何应对持怀疑态度者，包括事先告知纸上原型设计的缺点，通过使用更结实的纸质材料让测试过程看起来更专业，这样以来纸上原型设计测试过程中使用的卡片就更不宜磨损。

读者接着学习了良好的文档设计，以及为什么说它不仅事关产品设计大局，而且对良好的用户体验而言也非常重要。创建良好的产品文档需要从创建文档开发团队到对文档的彻底审阅等 8 个步骤，只有严格按照这些步骤开发的文档才会被用户认为是良好的文档，要知道用户一眼就能分辨出没经过审阅的文档。

本章的最后讨论了为什么要关注良好的用户界面设计。主要有 3 个原因：良好的设计可以节省你和团队的时间以及公司的投入成本，可以让潜在的用户下决心购买产品，同时可以让现有的用户感到满意。需要指出的是，产品及其文档的良好设计目标不仅只适用于产品的第一个版本，还必须贯彻到公司后续发布的各个版本产品之中去。

复习题

在进入第5章之前，读者最好能够回顾一下本章中学到的知识。所以，请思考并回答下面的问题，参考答案在附录A中。

1. 为什么在开始用户界面设计过程之前要解决好各种冲突和限制？
2. 为什么用户界面应是优雅的？
3. 如何弥补用户和设计师之间的限制差异？
4. 为什么要使用纸上原型设计？
5. 怎样做才能让纸上原型设计过程看起来更专业？
6. 纸上原型设计的优点是什么？
7. 纸上原型设计的缺点是什么？
8. 为什么一个产品需要良好的文档？
9. 为什么良好的文档设计非常重要？
10. 为什么良好的设计这么重要？



用户行为

无法道出自己内心期望或愿望的人，仍带着模糊的想法与无际的愿望叹息并奋斗着。

——拉尔夫·瓦尔多·爱默生

本章内容提要

- 用户行为心理学
- 知识：大脑应付外界
- 任务结构
- 有意识行为与潜意识行为
- 将困难任务转化为简单任务
- 创建概念模型

你已经了解了优良的用户设计以及如何建立好的用户界面与用户文档，现在需要学习的是用户行为，这样才能创建满足用户需求的软件产品、硬件产品或网络产品。当一个设计师基本完成了产品或文档的设计却忽略了用户心理时，其产品可能会很快变得不再适

用——也只有深窥其径的用户才能看明白这点。

技术试验的故事很多。在 TiVo 与数字录像机问世之前，较差的可用性的“黄金准则”是模拟式磁带录放机——孩子经常能够比大人更快地捉摸出其操作方法。如果你拥有了大量的技术经验，你同时应该清楚你是内部技术维修部门的人，而且周围有一个团队。最近我拜访了我的祖母，她让我修理她的收音机。她无意按了一个按钮，竟然把预设的电台频段(是家里其他人为她设置的)和表上的时间给弄没了。我只花了2分钟来搞定它，因为对于摆弄带很多按钮的电子设备我很熟悉。

为了明白用户需求，你需要稍微研究一下心理学。用户带着经验来面对一个新任务，而且也带着这些经验来面对一个概念模型，考虑外界的规则。

人们必须利用他们脑中的知识来应付外界已存在的知识。我们大多数人大脑中的知识并不确切，但是却经常被外界也同样存在的知识唤醒。有时我们在其他地方(例如互联网)来寻找外界知识。随着不断接触，用户对外界认知也在日渐增加。前后情景影响着认知、态度及方法。用户在利用头脑中的知识应付外界的知识时会做出权衡。

根据 Norman(2002)，当人们赋予一个新任务时，他们会追随3种不同类型的任务结构中的一个。他们还会坚持以前他们遇到过的信息，并在此信息基础上有意识或无意识地做出选择。

你将会看到如何利用数个步骤来开发一个用户界面，将艰难任务转化为简单任务，并把你组建的东西创建成一个概念模型。

5.1 用户行为心理学

每个使用过某个东西或试图执行某项任务而遭到挫折的人都感受过无助的滋味。确实，许多人都会为无法利用一个产品或实物来执行任务(Norman, 2002)找到理由。这些理由归纳如下：

- **自责**——例如，我从健康保险公司那里收到一笔账单，我相信自己必须支付。但是我却忘了自己已经签过保险公司的自动付款方式。所以这是重复付账。我为这样的

失误感到自责，而健康保险公司却从来不在账单上打印任何明显提示，诸如在支付线上说明“请勿付款”。取而代之的是账单上印有小体印刷字，说我不必用支票支付。这一产品(账单)没有透露这样的信息，结果就是我付了两次款(幸运的是保险公司退还了我用支票支付的款项)。

- **坚持错误理念**——你可能会坚持错误理念，因为这样使你感到安稳。所以你决定指责产品或公司，这样做比阅读用户指南或网上查询资料容易多了。例如，买来的路由器不工作，于是我开始抱怨这家公司。直到后来和公司的技术支持维修代表谈过之后我才认识到：我把宽带网线插到错误的路由器端口上了。
- **归罪于错误的原因**——最近，我的干燥器工作的时间越来越长。这个干燥器是一个组合设备的一部分：洗涤器在底部，干燥器在顶部，而干燥器没有显示纤维屑过滤器的位置。于是我坚持了一个错误的概念，心想既然没有任何东西来显示过滤器的位置，干燥器应该自动清除纤维屑。结果是干燥器确实有一个纤维屑过滤器——只是它位于装置的背面，加工得像其中一个排放口，但干燥器装置上并没有任何提示。在搜索这家制造商的网站时，我发现这种组合式洗涤/干燥器已成为过时报废产品；而由于住在公寓，我未从他们管理部门收到任何信息。我不得不对干燥器竟然从没有着火表示惊讶。
- **无助**——用户在某一项事物上不断遭受失败后可以感受到这点，从以往的故障或糟糕的设计中也能感受到这点。例如，若客户想在你的网站上办点事(比如付款)，而你在网站设计中没有采用三次点击规则，那么他们可能在到达目的地之前就放弃了，因为他们在其他地方已经遇到过类似问题，也没耐心去持续点击和搜索。但是在网上搜索的人比试图执行某项具体任务的人更具耐心，所以三次点击规则对于进行搜索的人来说可能就没有那么重要了。

5.1.1 人格类型

你也许熟悉用于识别和研究性格类型的 Myers-Briggs 类型指标 (MBTI)。该指标测试是在第二次世界大战期间由 Katherine Briggs 和他的女儿 Isabelle Myers 共同研制开发的，随后被很多公司用作人际关系学课程，培训职员学习公司中的性格类型以及如何与每种类型的人相处。我记得数年前我曾受过相同类型的 MBTI 强制训练。

MBTI 基于 Carl Jung 及更早的 Hippocrates(希腊的名医，号称医药之父)开创的心理方

式(Eisenberg and Eisenberg, 2006)。心理类型分为4种, Jung 将其称为情感型、思考型、敏感型及直觉型。

由于人们是用左右两半脑共同做出一个决定, 于是 Briggs 和 Myers 将4种性格中的每一种都扩展成两类。这种划分反映了左脑(逻辑和理性)和右脑(创意和情感)所产生的差异。在最新一期测试中, 让人回答93个强制选择问题, 即只有2个可选答案。然后测试者对测试进行打分, 并将信息归入4种性格中的一种(维基百科中的地址为 http://en.wikipedia.org/wiki/Myers-Briggs_Type_Indicator)。

- 内向型和外向型——这种划分反映了人们获得能量的来源不同。内向的人从想法和观感的内在判断上得到能量, 而外向的人从外界获得能量。长时间的社会交往可能会使一个内向的人不堪重负, 然而却可能让一个外向的人精神奕奕。这种差别并不能反映你喜欢别人的程度或想说话的程度。
- 直觉型和敏感型——这种划分说明了人们是如何从外界获得信息的。敏感的人从外界获得的具体信息中收获信息。直觉型的人根据他们过去的经验, 从预感和联想中获得信息。敏感更多的是一种有意识行为, 而直觉更多的是一种潜意识行为。稍后你将在本章进一步学习意识与潜意识。通常来说, 一个敏感的人本质上精于数理, 而直觉型的人能更好地理解文学。
- 思考型和情感型——思考型的人组织事情有逻辑性, 对人和事物的看法很主观。情感型的人对人和事的看法取决于他们内在的感受或本能。然而这并不意味着思考型的人没有感觉或情感型的人不思考——这只是反映了人们在做决定的时候是倾向于先把事情考虑清楚还是依赖自己的直觉。
- 判断型和感知型——这种划分说明了人们怎样组织和构造他们的生活, 如果你喜欢常规的程序生活并且只想生活稍有变化(如偶尔去一次电影院), 你属于判断型的人。然而如果你喜欢不固定的生活并且享受生活带给你的每一天, 你属于感知型的人。判断类型的人喜欢做决定, 感知型喜欢收集信息并采集所有的可能性, 所以他们做决定所花费时间较长。

由于并不是每个人都能符合与左脑和右脑相关的每种性格特点, Myers 和 Briggs 将这4种划分组合成16种不同的性格类型。例如, 一个人可能是内向的, 但他们待人接物却是情感性的。除了这8种不同的性格特征之外, 固然还有许多人是在此范围之外

的——例如那些完全左脑特征者和那些完全右脑特征者，但大多数人还是处在这两个极端之间的。

由于不同背景下人们的行为可能不同，这些性格类型并不总是能预示人们在不同情况下会有怎样的行为。例如：那些参加可用性测试的人可能会与平时的反应不一样，因为他们知道他们是在进行测试。这些人可能更倾向于向你反馈产品的可用性情况，而在现实世界中他们并不这样做（这种情况下他们若发现产品适应性上的不足，可能会要求退钱）。

5.1.2 四种基本人格

Bryan 和 Jeffrey Eisenberg (2006) 也学习了 David Keirsey 和 Marilyn Bates (他们对长期行为模式更感兴趣) 在人格和性格类型上的发展。后来这些研究与 Myers - Briggs 的性格测试相结合，显示了人们做决定的时候是如何将他们的个性带到工作中的。在第 6 章“用户分析”中，你将会了解如何创造用户模式组 (所谓的角色)，群组中将包括一个或更多的这种个性类型。从这些模型中，你可以了解到谁是主要用户，然后确定什么界面最适合那些用户。例如：如果你的主要用户是敏感型的人和判断型的人，那么你所创建的界面可能就不能引进太多陌生的东西，并且应提供完成任务所需要的视觉线索。

当任务快完成或接近某个位置时，Keirsey and Bates 认识到敏感型和直觉型是首先考虑的，因为他们想知道从大脑到外界他们是怎样处理的 (你将在本章后面部分了解大脑中的知识应付外界中知识)。Keirsey 与 Bates 后来采用这个参考标准，与 4 种左脑性格类型相结合，创建了 4 种主要人格 (Eisenberg and Eisenberg, 2006)。

- 敏感型/判断型
- 敏感型/感知型
- 直觉型/情感型
- 直觉型/思考型

Eisenberg 和 Eisenberg (2006) 为 4 种主要人格的每一种都贴上了直观的标签，这种标签可以展现出每种类型人格的用户在面对一个新的任务或一个新的情景时的行为。4 种类型如下：

- 敏感型/判断型：方法型
- 敏感型/感知型：本能型
- 直觉型/情感型：人文型
- 直觉型/思考型：竞争型

提示：

如果你受过性格训练，你可以发现这些不同类型的性格可以和不同类型的动物联系起来。我曾经作为一个高科技公司的合同商接受性格培训，发现这4种性格可以与不同类型的鸟联系起来。方法型对应猫头鹰，本能型对应孔雀，人文型对应鸽子，竞争型对应老鹰。

1. 方法型

方法型的人逻辑性强，他们总是非常缓慢地接近某种新事物，他们表现的是如下一些性格(Eisenberg and Eisenberg, 2006)：

- 他们态度客观，注重细节。
- 他们非常守时、系统地安排任务，以尽可能地在一天中做更多的事。
- 他们想知道一个任务或一种方案是如何为他们解决问题的，以及其实存在的证据。并且，他们还想看到你能够提供优秀的客户服务的证明及其他证据。如果是一个用户界面，方法型的人想看到的是你没有用大量不必要的点击来浪费用户时间。
- 他们相信：完成任务就是任务自身对人的奖励。
- 他们鄙视无序和低效。

方法型的人想让你回答的是指导性问题：

- 完成该任务需要哪些过程？
- 完成该任务需要哪些细节？
- 你有什么证据能证明该任务能完成？

2. 本能型

本能型的人生活在瞬间，他们行动的时候容易冲动。他们希望立即反应，跳过大量细节并且很快地做直接决定。本能型的人表现出以下一些性格特征(Eisenberg and Eisenberg,

2006) :

- 他们喜欢与人打交道，雷厉风行。
- 他们随意地支配时间并且保持快节奏的生活方式。
- 他们喜欢表达即时的需要，并且希望看到相关的、可靠的选择，以便能马上解决问题。他们从不阅读说明。

本能型的人想让你回答“为什么”和“什么时候”类型的问题：

- 你怎样能尽快地弄到我想要的？
- 怎样能减少选项，以便我尽快地做出决定？
- 我可以自定义你们产品的使用方式吗？

3. 人文型

人文型的人凡事先考虑别人。他们不想让任何人为他们做事，这样会让他们的感觉不舒服。人文型的人高度重视人际关系并且喜欢帮助别人，不喜欢个人主义。他们有创意，为人风趣，是个很好的听众。所以他们有很多朋友和熟人也就不足为奇了。人文型更喜欢看大图片，非常看重他们和他人的发展。

人文型的人展示出以下性格特征 (Eisenberg and Eisenberg, 2006) :

- 他们喜欢与人打交道，他们的生活围绕着关系转。
- 他们的时间是不受限制的，喜欢慢节奏，人文主义者一般都不喜欢时间期限。
- 他们想知道还有谁使用了某种方法解决了某个问题，他们想知道在证明过程中其他人的经验以及用户反馈的其他方法。

人文型的人想让你回答的是关于“人”的问题：

- 你的产品让我感觉如何？
- 这个产品对谁有用？
- 我可以自定义你的产品使用方式吗？
- 我可以信任你吗？
- 谁负责领导员工研发的该产品？

4. 竞争型

竞争型的人喜欢与别人和自我竞争，他们的期望不只是理解生活，而是控制生活。他们享受克服挑战，学习新事物并寻求方法达到他们的目标。竞争型的人具有很高的动机性和说服力，一旦感觉手头上掌握了所有信息，他们通常快速地做出抉择。

竞争型的人表现出以下性格特征(Eisenberg and Eisenberg, 2006)：

- 他们是能力型和有条理的。
- 他们严于守时，战略性地管理时间，将每一天都发挥到极至。
- 他们想知道某个建议方案会为他们带来什么，并且想让你提供合理的选择来应付挑战。
- 他们鄙视无序和低效。

竞争型的人想你回答的是关于“什么”的问题：

- 你的产品及其公司可靠吗？
- 你怎样能帮我提高产率？
- 你们产品的竞争优势是什么？

5.1.3 人类行为的七个阶段

许多日常任务并没有经过计划，但他们是机会主义者——多数情况下，人们只是在想到的时候，才会决定使用某个东西。不管你以前是否用过某个产品，你都可能会仅仅因为误解或曲解而导致使用困难。

Norman(2002)中描述了任务执行时7个人类行为步骤，这些误解或曲解可能出现在其中的任何阶段：

1. **形成目标**——例如，如果你有一个网址，上面有用户需要的信息，那么用户就认为其目标就是在网站上查询信息。
2. **产生意向**——如果用户相信该网站有她需要的信息，那么她将决定查询该信息。
3. **规定行为**——用户需要明确点击哪个链接能让她找到网站上的该信息。

4. **执行动作**——用户点击链接，到达网站的下一级页面。

5. **感知外界的状况**——新的网站页面出现在用户的因特浏览器上，用户阅读页面并感知执行完毕后的出现结果。

6. **解释外界的状况**——用户处理网站新页面上的信息，并确定网站上的信息是否需要信息。

7. **估计结果**——用户确定点击的这个链接是否符合她渴望的结果。如果不是，她就会点击另一网页看它是否跟她渴望的结果更接近。

执行一个任务的时候，人们可能会用到一个、一部分或所有的步骤，并且存在一个连续的反馈回路。误解和曲解可能是因为来自某些人由于某些原因没执行所有步骤，例如自信他们以前用过类似产品并且相信其行为对该新产品也同样有效。还有一些误解和曲解来自于糟糕的设计，未说明用户该如何感知外界的状况。

5.2 知识：大脑应付外界

人们依赖于对 2 种偏爱所做出的不同选择，即感知和直觉 (Eisenberg and Eisenberg, 2006)。感知是基于人们从外界收集到的信息中所获得知识；直觉更多的是基于他们脑中的存储信息，例如从相似任务中得到的类型。

多数人依赖着外界传播的不严密的知识，例如，人们能分辨 10 美元钞票是因为他们看见了钞票一角的数字 10。然而，如果你找一群人问钞票上的人是谁或者钞票反面的建筑是什么，恐怕告诉你正确答案的不会很多。很多情况下人们不需要精准的知识。例如，我看到钱包里一张样子和图 5-1 一样的 10 美元，打算拿来买午餐，我之所以不关心钞票正面的人是因为这一信息并非完成交易所需。然而，我需要知道钞票一角的数字 10 才能正确地付账来得到我需求的食物。

我们的大脑只能将如此多的信息接收在记忆当中。因此，我们需要一种强制约束来将信息分解，这样大脑才能进行处理。例如，电话号码是 7 位数字，因为人类的大脑一次只能处理 7 个数字。当我们执行更复杂的任务时，比如组装一个新的电脑桌(需要大量零件

和众多步骤), 我们会把每个任务分解成小任务, 这样才可以统筹全局。



图 5-1 你能说出 10 元美钞上的人物名字吗

人们也可以从外界(即外部信息源)中回忆起知识。如果你像我一样, 就会用一个日历程序(如 Microsoft Outlook)在特定时间提醒自己去执行任务(或者学习我, 将你 Palm 手机或其他掌上 PC 的日历和你的日历调成同步)。同样, 产品说明书就是用来告诉用户如何使用产品, 就像炉子上的调节控制器, 告诉你哪个旋钮可以控制合适的火焰大小。

用多少脑中知识来应付外界, 取决于当前情况对应的背景环境。例如, 当你看到一个新的 Windows 程序, 而你了解 Windows 怎样工作, 所以你可能会调用更多大脑中关于如何使用 Windows 产品的信息, 来引导你使用这一程序。在使用大脑中知识应付外界中存在的信息时有一个交换平衡。例如, 外界中的信息只要是可视的或可听的, 并且在你的视觉和听觉的范围之内, 则都很容易获取。然而如果你依赖于记忆, 那么可能回忆起来就很费时间或被其他相关事分心而抓狂。但是, 当你确实记得某些事情之后会很有效率, 因为你知道如何完成任务。如果你依赖于外界信息的指导(例如你所琢磨的那个产品的组装图), 那效率就低得多了。

当你在第 6 章中构建人物角色时, 会进一步了解用户是如何看待大脑中和外界中的知识的。然后你将利用这些信息来设计一个界面, 帮助用户快速、有效地完成任务。

5.3 任务结构

那么人们又是怎样运用其他人大脑中用于应付外界的知识，来使沟通尽可能高效的呢？他们构造任务，像你在学校中学的那样。执行任务时人们要面对3种类型的任务结构(Norman, 2002)。

- **在执行任务时包含大量选择的深广结构，像是一场比赛。**当执行某个任务，比如在棒球比赛中执行任务时，这种结构在每种情况下都含有多种可能结果。一个击球员击球的时候，可能击出本垒打，也可能击在球场内。若击在场内，可能一垒打、二垒打或三垒打。或他可以尝试将一垒打变成二垒打然后安全跑到二垒或被判出局。或者他可能打出一个界外球让对手抓住或没抓住。或者他打下坠球或腾空球到9个防守队员中的一个。要么他可能三振出局。

这里只有少量的可能结果。

- **在顶级选项下面提供了少量选项的浅层结构。**例如，一个餐馆菜单上包含许多顶级选项让你从中选择，比如开胃小菜、沙拉、主菜、饮料及餐后甜点。在每个顶级选项下面你还有其他选择，比如主菜下面有牛肉、鸡肉、猪肉、鱼肉及素食等。你可以在那些选项选定你需要的副菜和蔬菜。电脑程序窗口上的菜单栏也是一种浅层结构。
- **深窄结构，比如要求采取许多步骤来完成的方法。**当你教用户做某件事的时候，用户文档通常使用的就是这种结构。在这种文档中你可以用一步步的简短说明来指导读者完成任务，我的一个前任老板将这种文档的设计方式称为“食谱结构”。

总体来说，人们发现在执行某项任务时，深广结构挑战性最大、耗时最长，因为可选项太多。此中的原因在于人们是如何有意识和潜意识行动的。

5.4 有意识行为和潜意识行为

我们每天很多的行为都是在潜意识中进行的(Norman, 2002)。我们一次次刷牙、洗澡、穿衣及开车去工作然后再回家。有些时候我们甚至记不起工作路上经过了一个十字路

口，即使我们明白确实经过了那个路口，因为很明显我们在班上！潜意识行为建立在类型和规律性上，而我们的潜意识每天都根据这种类型完成其任务。

有时我们的潜意识是无形中发生的。在我上小学时，我感觉做拼写测试太舒服了，于是我的潜意识决定开上2分钟小差，而我并未觉察到那2分钟。但是当我朝下看试卷时，发现我又写了两个单词，并且都拼写正确了。

如果形成了长期记忆，潜意识行为可能来得很快，看起来似乎方法是瞬间闯进大脑中的。我现在已经是一个成年人了，我发现我的潜意识能带给我几天前考虑好的但已经忘记了的答案，这个答案突然就冒进我的大脑中，然后我赶紧把它写下来。潜意识行为还可以为预测行为或某些事物的原理提供源泉，这与有意识获取方法正好相反。

对比来说，有意识行为是缓慢的、连续的，依赖于短期记忆，有条件限制而且不可靠。当有意识地去执行任务时，你必须仔细考虑，通过人类行为的7个步骤来执行某项任务，这点你之前在本章中已经学过。有意识的念头也依赖于潜意识，看是否有模式能应用于这个任务，这种模式假设可能导致失误。

5.5 将困难任务变成简单任务

在和用户对话之前，设计者怎样使所创建的产品尽可能有用？建议设计者遵循 Norman(2002)原则，将困难任务变成简单任务。

- **要意识到用户使用的知识来自大脑和外界**，人们需要同时用这2种知识来获得对周围环境的精确感觉。当一个用户进入一个用户界面，他带来了从相关任务(脑中)中获得的信息，并且从用户界面(外界)获得反馈信息，然后对手头任务做一个精确评估。作为一个设计者，你需要在可用性试验中观察用户在任务中带来了什么信息，并且要了解他对于界面的这种反馈信息是如何反应的。
- **简化任务结构**。如果在一个界面中需要点击两次才能获得一条信息，而实际上这条信息可以减少到点击一次打开，则为用户简化过程。
- **可视化管理**。如果你可以使用户界面的选项可见，而并没有混乱屏幕，也没有喧宾夺主，则将其可视化。这是一种真正的平衡艺术，当你阅读了第9章“可用性”，了

解用户是如何对可用性试验中的界面反应后，你会知道如何去平衡。

- **确保正确地筹划。**用户感知一个产品是否易于使用，更多的是看查询信息速度而不是界面或网站加载信息的速度(Eisenberg and Eisenberg, 2006)。
- **利用约束开发更好的产品。**用户界面受操作系统限制，比如你的界面要求在一个窗口里面。
- **为故障做设计。**考虑用户使用程序而引起故障的所有可能方法，然后为之做好计划。一种方法以书面形式写下用例，总括用户可能引起故障的所有可能方式，然后针对性地筹划。在你进行可用性测试之前你可以减少很多这种故障。
- **当其他一切都不管用时，标准化。**幸运的是，在用户界面的设计中，操作系统在对你的设计强加了限制条件的同时，也提供了大量的标准。如果你有一个网站，设计规则是不固定的，但书籍和网站可以提供好的或坏的设计实例，这样你可以学习创建实用和高效网站的标准。

5.6 创建概念模型

当第一次接触某个任务时，你带着生活经验、信念以及自己的处事方式来试图完成该任务。对于其他人也是一样的，而且如本书前面所述，对于设计者来说可能很难认识到这一情况。

即使用户的家庭其他成员在帮忙，她多年来积攒起来的这些生活经验、信念及其他方法仍为她构建出自己的概念模型。概念模型是一个人如何执行某项任务的理念。

当某人第一次尝试执行某项任务时，她通过提出以下3个问题(Norman, 2002)一步步地创建出这一概念模型：

1. 这个产品能提供的预设用途是什么？Norman 把事物的感知属性和实际属性作为预设用途。当你使用某个物体来执行基本任务时，预设用途可以给你提供线索。例如，看一个带把的茶杯时，我知道那个把是用来握的。这个把就是预设用途。

2. 有什么限制条件吗？当我把手指握住茶杯把时，发现我只能放进几根手指，这是有限制的。（也可能有人会说你可以优雅一点，伸着小指不放进去，这样可能是一种预设

用途。)

3. 你怎样反应可能的操作? 当我把手指放进茶杯后, 我知道我用的手指并不影响到茶杯的预设用途, 但把手却表明了茶杯是怎样用的。

和使用茶杯相比, 使用手机要复杂得多。我先前所用的 LG 手机, 在键盘面板的顶端有数个按钮, 因为没有看手机资料, 所以我并不知道它们用来干什么。还有, 底部键区的开始键、零键、#号键含有很多不同符号, 我只瞄了两眼, 也不清楚它们的意思。

我将 LG 手机折价卖了换成 Palm Treo650。这款掌上 PC 比起那款旧手机功能强大得多, 但也复杂得多。上部是个空屏幕, 对于如何启动, 则没有任何开机提示——除非你自己低头仔细地看手机的前端, 上面有 6 个按钮和一个中央按钮, 四周还伴有 4 个箭头按钮。在侧边还有 2 个未带标签的按钮, 让人更加迷惑。侧边的按钮比较小, 而且没有任何提示是做什么的, 当然有些按钮还确实具备了暗示作用: 一个小按钮上有一幅家的图片在上面, 还有一个上面有个信封, 似乎表明这些按钮是用来打开主页和电子邮件程序的。

但是这些按钮还是没有告诉你如何开机。有两个按钮上电话手筒的图片。然而。如果你不熟悉“开-关”图标, 你就不知道哪个电话把手是正确的(红色的那个, 它也是打开 Treo 的键)。

我猜想很少的人会认为掌中 Treo 符合用户概念模型。现在你已经看到了 Treo 这种反面教材, 在你想创建一个好的概念模型的时候, 你应该如何设计一个能够创建良好模型的产品呢? Norman(2002) 建议: 设计者在创建某个产品时, 应考虑以下 4 个特点:

- **使一切简化, 让用户形成一个好的心智模式, 能够反映执行任务过程中对象的使用方法。**例如, 在用户界面中, 尽可能不用大量的键迷惑用户。在网页设计中尤为如此, 优秀的设计者始终遵守三次点击原则。这个规则表明: 如果用户不得不点击三次以上才能到达预定位置, 那么他将失去兴趣, 不再访问你的网站。

然而, 埃尔伯特·爱因斯坦曾经说过: 你要使事情尽可能的简单, 而不是仅仅简单一点而已。按照这个理论, 有时也很容易使东西简单到用户无法明白如何使用的地步——甚至找不到使用工具。例如, 有一次我拜访我妹妹和妹夫, 想要用他们的淋浴。而我不知道怎样将浴缸的水从排放切换到喷头。没有把手, 没有调节杆, 什么

都没有。最后，我妹夫发现排口凸边周围有一个环，只要你一拉这个环，水马上就切换过去了。我以前从没有见过那样的东西。

- **尽量可视化。**例如，假设你给某个人打电话，又不想被呼叫等待打搅，那么最后如果你看不到手机上任何呼叫等待的提示时你可能会感觉惊讶与不快。

在执行任务所使用的物体和使用结果之间，需要提供良好的联系纽带——但是，可能你会很容易忽略这一联系过程，因为你喜欢搬用过去的经验，特别是苹果 iPod 的经验。iPod 确实比较流行，但是用户对它的界面设计却是颇有微词。举个例子，如果你在另外一个界面(比如日历)中，想改变音量，是没有办法快速回到主菜单并进入音量控制的。你必须通过一个或数个菜单屏幕，一步步回到主菜单，这样才能够改变音量。

- **对用户执行结果，连续提供合适的反馈信息。**这个简单得就像显示上述掌上 Treo 是开还是关一样。当按了手机前部大多数按钮还无法开机时，用户可以通过不断的尝试和出错来确认到底是如何开机的。如果你点击了一个错误的键或执行了一次错误行为，程序应同时发出声讯或可视警告信息。

5.7 案例研究：进行访谈以建立概念模型

现在你已经把模板和组织模型结合在一起进行纸上原型测试，当团队成员使用数据库应用程序时，你需要对团队成员进行概念模型访谈。概念模型将帮助你 and Evan 确定纸上原型清单中应该加哪些任务。

也可能你的团队的模型需要对现存用户界面进行修改，使其更高效。它应用起来能使事情更加有效。在第 3 章中的初次会谈中给出了几条建议来修改现有情况并整合新特点，使界面的整体可用性更高。

你们俩共同决定会议(所有 10 个团队成员都要在会议中参加 MBTI 测试)时间，你对团队成员进行测试，因为你需要为第 6 章中将要开发的主要用户角色创建性格类型。你还要问团队成员的概念模型，并将他们的答案和他们的心理模型联系起来。

当你和 Evan 给 10 个成员 MBTI 测试结束并分析完结果，你会发现你对性格类型就有了一个良好的剖析。

- Mike 是一个外向、直觉、思考型和感知型的人，取每个词的字母就是 ENTP。
- Traci 是个内向、直觉、思考型和判断型的人(INTJ)。
- Jay 是个外向(你可能会认为是市场部门的人)、敏感、情感型和感知型的人。(ESFP)。
- Laura 是个外向、直觉、思考型和判断型的人(ENTJ)。
- Michelle 是个外向、敏感、情感型和判断型的人(ESFJ)。
- Tony 是个内向、直觉、情感型和判断型的人(INFJ)。
- Maureen 是个内向、直觉、思考型和感知型的人(INTP)。
- Bruce 是个内向、直觉、思考型和判断型的人(INTJ)。
- Travis 是个内向、敏感、情感型和判断型的人(ISFJ)。
- Paul 是个外向、直觉、思考型和感知型的人(ENTP)。

如表 5-1 所示，10 个成员列入 8 个不同的类型(每种类型用一种字母缩写表示，说明每个成员的性格类型)

表 5-1 团队成员的心理类型

ENTP: Mike, Paul	ESFP: Jay	ENTJ: Laura	ESFJ: Michelle
INTJ: Traci, Bruce	INFJ: Tony	INTP: Maureen	ISFP: Travis

既然你和 Evan 知道了用户的心理类型，你可以在项目团队中向每个用户提以下问题：

- 你感觉什么样的应用程序特点能使你的工作容易？(这个问题会告诉你和 Evan，用户会如何考虑这个应用程序的用途。)
- 当你试着做某些事情的时候，应用程序中的哪些特点让你感觉受限制？
- 如果你感到了某种限制，你会如何修改这个操作，使其更好地为你服务？(这个问题会给你提供一个组织各任务的好方法。)

提问完之后，将答案汇总、归类。比方说，从你的提问中，你发现 Laura, Michelle, and Traci 喜欢简单一点的方式，直接联系产品公司。因为他们 3 个人最可能给公司打电话查询信息。然后你可以修改一下任务，在纸上原型中适当多加一些特点。

例如，在联系资料中你可能会想增加一个电话号码的链接，当用户点击这个链接时，

可以通过商店的画外音 IP 电话系统给公司打电话，这样用户不必打电话。然后你可以在纸上模型中增加一个电话实体模型，项目团队成员可以查看一下，如果他们点击了数据库应用程序中的电话号码，VoIP 电话系统是否会激活并拨打号码。

在下一章中，你要问成员更多的问题，你必须明白他们的需要和目标。你将学会如何从这些会谈中收集信息，并且为你的应用程序创建主要角色。

5.8 小结

本章首先讨论了用户行为的心理学和用户的一些误解、曲解以及它们如何、为什么会发生。同时还讨论了当用户被说服做某事时，Myers-Briggs 类型指标所识别的各种性格特点将如何影响他们提出的问题类型。

随后讲述了大脑中知识如何应付外界知识。你了解了大多数人士是如何依赖于不严密的知识来适应一种新情况的。其中很大原因是我们的大脑只能接受这么多的信息，而且我们把复杂的任务分成小任务，然后跟踪每件小任务。同时你还学习了人们是如何被外界知识唤醒记忆的，以及从大脑中和外界中平衡使用知识。

任务分解之后是任务结构的讨论。我们讨论了不同类型的任务结构：提供大量选择的宽广结构；提供顶级选择及子选择的浅层结构；为执行游戏提供步骤式说明的深窄结构。我们有意识和潜意识行为决定我们所使用的任务类型。

接下来，本章还讨论了将困难任务转化成简单任务，以使用户界面得到更好的理解。其中，讨论了 7 步任务简化原则，供你在任何设计背景下使用。第 7 步与最终原则是最重要的一个：如果一切都不管用，标准化。

本章还讨论了计算机用户在接触一个新的用户界面时，大脑中是如何带着概念模型的。概念模型是基于用户以往的经验、信仰及做事方式。本章还叙述了当用户创建一个概念模型时，需要解决的具体步骤。

复习题

在继续第 6 章之前，现在先回顾一下本章所学的知识。问自己以下问题，然后参考附

录 A 来复查你的答案。

1. 什么是预设用途?
2. 什么是约束?
3. 为什么人们在操作失败时会感觉到无助?
4. MBTI 测试是做什么用的?
5. 人类行为的 7 个阶段分别是什么?
6. 大脑和外界中的知识之间的平衡点是什么?
7. 什么样的任务结构对人们而言是最有挑战性的?
8. 为什么在使用有意识的想法时你必须要慎重?
9. 为什么要把较难的任务划分成多个简单的任务?
10. 人们的概念模型是靠什么建立起来的?



用户分析

“最好的教育家应当去观察生活、体验生活，而不是依靠书本上的文字或别人的述说。”

——Amos Bronson Alcott

本章内容提要

- 用户心智模型
- 经验钟形曲线
- 了解用户目标
- 用户与任务目标

在第5章“用户行为”中，你了解了用户的性格类型、经验以及由此带来的用户行为。这种用户心智模型是你设计界面所应当了解的——用户期望的界面看起来应当是什么样的，界面如何运作等等。你的界面设计和用户的视角越接近，用户就会越满意。

如果将用户体验水平等级绘制在一张图上，你会看到它们将形成一个钟形曲线。钟形曲线上的大多数用户都位于中等的知识水平上，因此你可以设计用户界面使其满足大多数用户的需求。

为了给用户创建一个优秀的界面或产品设计，你需要有目标。因此，本章要讨论 Cooper 和 Reimann (2003) 创立的“目标导向设计过程”。它将设计过程分为 5 个步骤来了解用户目标。通过在设计过程中回答一系列的问题，你将明白你所寻找的是什么。你还会学习第 3 章“制作商业案例”中的“可用性工程生命周期”在“目标导向设计过程”中的适用位置。

用户是有目标的，但是怎样才能找出不同情况下的目标呢？你可以通过用户和任务分析来做，这些用户和任务分析同样适用于“目标导向设计过程”。然后你可以创建用户的人物角色，这些人物角色代表了具体类型个体的具体需求。创建完人物角色之后，可以将其分级，确定出你想要为之设计用户界面的那些人物角色。

6.1 用户心智模型

Cooper 和 Reimann (2003) 曾指出：软件开发过程经历了 4 个阶段。电脑问世之初，懂得如何对电脑进行编程的人就算是黑客。一开始，在设计电脑时，程序员什么都要做，无所不及。第一代家用电脑（比如 Commodore PET 和 64、Atari 400 和 800、Radio Shack TRS-80、以及 Apple II）在 20 世纪 70 年代末 80 年代初开始流行之后，情况也还是如此。

Apple II 的 VisiCalc 的问世之后，IBM 个人电脑的普遍接受及其在商务上的适应性，促使编程行业接受商业流程的规范，即由经理来驱动新的软件项目。经理的大部分工作是规定功能清单，然后无奈地看着版本 1 中所规定的软件功能在版本号到了 3 的时候仍没有完成。

随着 20 世纪 90 年代图形用户界面大行其道，以及大多数买回家的电脑都接入了因特网，越来越多的公司开始对外观和体验发生了兴趣。还有部分原因是他们发现许多人给他们打电话抱怨可用性的问题，公司自身也想遏制售后服务成本的增长。*Beta 测试员*（首选是软件正式发布之前测试软件及识别 bug 的客户）的使用及可用性测试在这一时期变得非常普及。

在过去的 5 到 10 年中，编码之前的优秀设计问题变得水涨船高。不幸的是，尽管在软件研发过程中有设计说明，大多数工程师仍然从增加特性和功能的角度来设计软件，因

为功能才是工程师所看重的东西。结果，许多软件程序不是执行起来非常笨拙，就是无法实现对机械时代的结构进行数字时代的改进。

举个日历的例子。早期的 Windows 程序一次只能简单地显示一个月的时间，也不配置其他任何功能，如逐月或逐年滚动显示。我的 Outlook 2003 中的日历要好一些，如图 6-1 所示，但是功能仍然很有限。例如，对于下 3 个月的时间，我可以在 Outlook 窗口的左上角看到日历摘要，但是无法在 Outlook 中看到年度日历。我一次只能察看一个月。

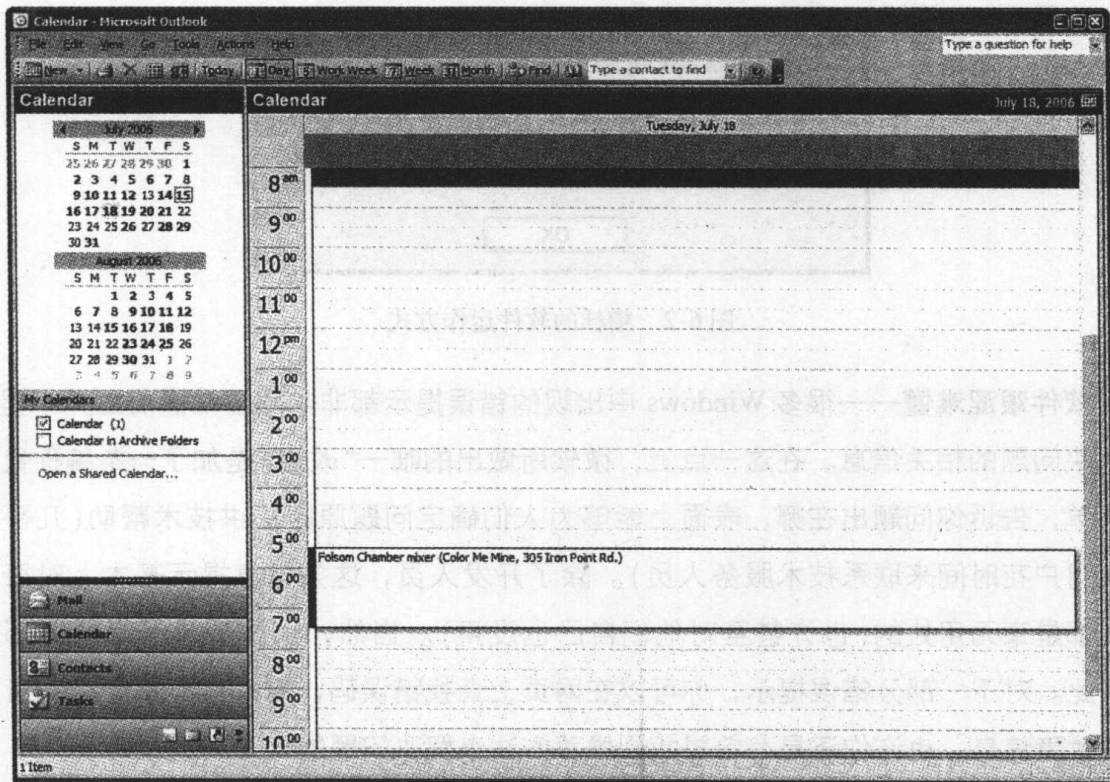


图 6-1 Outlook 2003 中的日历

问题在于软件工程师仍然依赖机械时代的标准进行设计。他们只是逐渐加入一些能扩展常见系统功能(比如日历)的技术。

6.1.1 产生后果

这种机械时代设计的后果是你会有 4 种主要错误行为仍然未得到纠正 (Cooper 和 Reimann, 2003) :

- **软件假设了用户有能力理解为什么需要做一些事情**——如果程序不告诉你为什么要做这些事，你就可能会犯错，使你损失大量的工作。例如，如果你推测软件在你关闭前会自动存档，于是在看到提示后决定不保存文件（因为你认为软件会自动保存），那么在下次打开时你将勃然大怒。
- **软件经常对用户不友好**：出了问题它总是怪罪用户而不自责粗糙设计导致的过失——你曾经多少次收到并不告诉你任何故障发生原因的信息提示（见图 6-2 中的一个实例，虽然已经安装了软件，但系统还请求安装。）

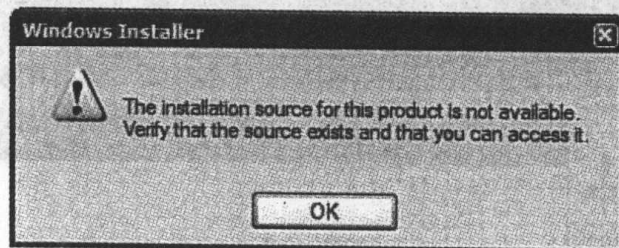


图 6-2 糟糕的软件运作方式

- **软件艰涩难懂**——很多 Windows 中出现的错误提示都非常艰涩难懂而且并不提供解决问题的相关信息。在这一点上，微软所做出的唯一“改进”是加了一条编程代码清单，告诉你问题出在哪，表面上能够为人们确定问题原因提供技术帮助（几乎没有用户花时间来联系技术服务人员）。除了开发人员，这条信息提示基本上对每个人都是毫无用处的（也许甚至对他们自己也没用）。当然，也并不是所有情况都是如此。对于一部分错误提示，你可以在网页上查找错误码，看是否有跟此编码有关的故障信息。也许你还是有希望找到推荐方法来解决问题的。
- **软件行为可能是莫名其妙的**——例如，我在 Microsoft Word 中察看一个文档的“字数统计”（“字数统计”是并不对文档做任何修改的一个任务）后，Word 竟然要求我重新保存文件。这一软件行为在其后的 Office 新版本中并没有修正过。而 WordPerfect 则不向我请求保存文件就直接关闭了程序。

6.1.2 实现模型与心智模型

Cooper 和 Reimann (2003) 对实现模型和用户的心智模型进行了区分：

- **实现模型**——表示一个产品实际是如何工作的。

- **用户心智模型**——规定了用户在使用产品时，并不需要知道其工作原理的细节。例如，你不需要知道你的光盘驱动器是如何将数据写入光盘的——你只需要知道如何把光盘正确地放入光驱中，让你的电脑把数据写入光盘里（而不是把光驱用作茶杯托架）。

不幸的是，由于界面遵从软件内部的逻辑，工程师设计的大多数软件遵照的是实现模型。例如，有一个独立的对话框描述了用户的每一个行为（Cooper 和 Reimann, 2003），向用户提示该信息的时候，是程序需要接收它的时候，而不是此时它本身应该向用户提供该信息。

由于人们形成的心智模型比现实简单，设计师应致力于简单化——即 Norman (2002) 原则的其中一条：将困难任务转化为简单任务。（也许你听说过缩写词 KISS: Keep It Simple, Stupid. 即整简单点，白痴）用户并不关心事物的工作原理，也不关心他们的感觉精确与否——甚至是真实与否。他们知道是在和什么东西互动，他们希望界面能尽可能地反映他们自己的模型。实现模型越接近于心智模型，用户就会感觉界面越容易。

6.2 经验钟形曲线

在第 5 章中我们讨论了人们自身带有的经验、行为及其他个性特点，在前面的章节中还讨论了用户是如何以这些特征创建心智模型的。如果你对多组不同的用户进行测验，并将他们的经验水平绘制成图，那么你会发现他们大多数人都位于 Cooper 和 Reimann (2003) 描述的永久中级用户这个范围内。Cooper 和 Reimann 把中级用户看作是永久型的，是因为他们中的大多数既没有时间、也没有兴趣去学习更多的程序知识，他们只想知道如何能够快速地完成规定任务。

如图 6-3 所示，图表看起来像一个钟形曲线，曲线的大部分是永久的中级用户，两头是初级用户和专家用户。

这个钟形曲线并不能总是精确地表示每一个用户界面下的每一个电脑用户的情况。初级用户不会长时间地原地踏步，一部分原因是没人喜欢当菜鸟的感觉；另一方面，在尝到甜头后用户可能对学习如何操作界面比较感兴趣。如果在一定时间内使用过足够的功能，人们也可以从中级用户转化为专家用户。

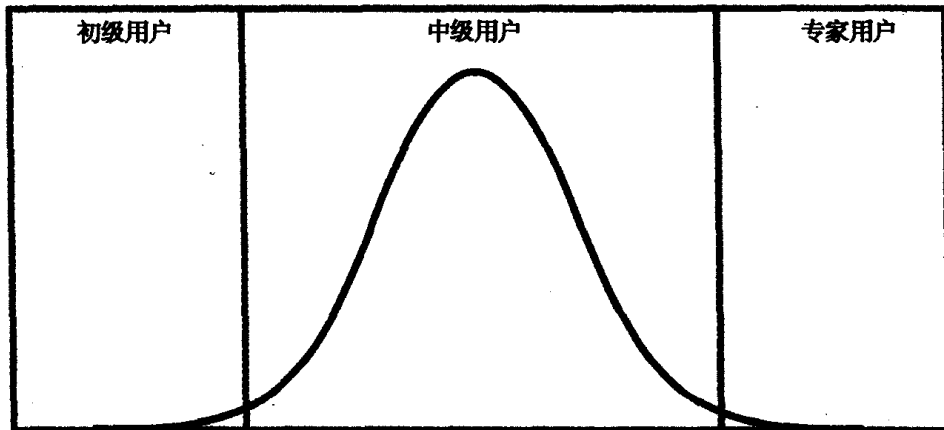


图 6-3 经验钟形曲线

根据对经验定义的不同，曲线形状还可能不一样。举个例子，如果你调查了很多初次使用某个程序的人，曲线的倾斜形状会显示使用这个程序的大多数人都都是初级用户，图中只有很少一部分人（一般是研发者）处于中级至专家阶段。然而，如果测试中的所有用户都熟于 Windows 操作，图表就会倾斜成另外一种形状，显示群体中没有初级用户——只有处于不同阶段的中级用户和专家用户。

不同群体的需求

三组用户——初级用户、中级用户及专家用户——其需求是不一样的（Cooper 和 Reimann, 2003）。如果用户界面（以及其他外围信息，如你的文件）设计能够满足所有用户要求，而不是主要满足一个组或其他组，用户会感到更满意。

1. 初级用户

初级用户在开始使用某个程序的时候，清楚自己是个菜鸟，所以他们不希望程序再让他们强化这种感觉。用户都希望自己被当成一个聪明人来对待，并且希望学得越快越好。因此，你需要快速有效地传递说明信息。这也就很好地诠释了为什么需要把用户界面设计得离用户心智模型越近越好。在本章的后面部分，你将了解如何分析用户心智模型。

初级用户提出的问题较为基础和普遍：

- 该产品是干什么的？
- 我应该从哪开始？

- 我需要做哪些事情来完成该项任务？

有些软件只把在线帮助作为唯一的技术支持方式，但是在线帮助不应设计为初级用户快速入门的工具。如果在线帮助设计得不错，中级用户和专家用户可以用来快速获得某个问题或主题方面的信息。让初级用户快速入门的更好方法是利用演示向他们展示基本任务以及如何使用程序完成这些任务。这种演示应当尽可能可供交互，使用这种入门指导可以强化说明完成某项任务所需要的每个步骤。目前有提供这种交互演示的程序，如果你愿意的话，甚至可以在 PowerPoint 中设计一个交互设计演示(见图 6-4)。

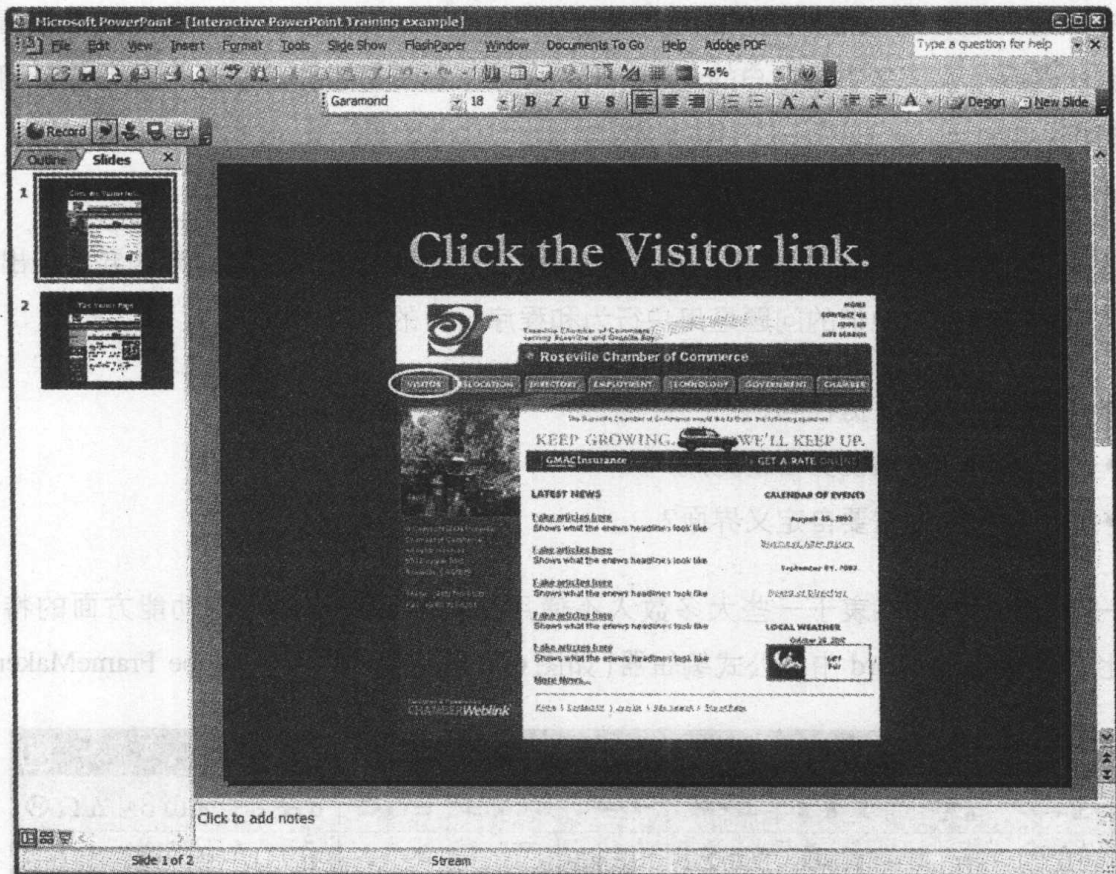


图 6-4 PowerPoint 交互演示实例

2. 中级用户

中级用户需要一些问题的具体答案，包括以下几点：

- 你能够提醒我如何执行这项任务吗？

- 我怎样才能找到该功能?
- 这个升级版有什么新特性?
- 我可以撤销上一步的操作吗?
- 执行该任务的命令是什么?

这部分用户希望能够快速找到他们需要的工具，所以用户界面设计中很重要的一点就是能够让他们快速得到这些工具。在用户界面中你可以在这方面做大量工作。中级用户对在线帮助依赖性大，在线帮助应该在程序中易于调取，并能够快速提供答案。

中级用户不需要了解高级特性，但是他们希望配备这些特性，以备将来在某些情况下用得到。实际上，许多中级用户的 Microsoft Word 中都带有也许将来某一天他们会用到的高级特性。

3. 专家用户

专家用户长期使用用户界面，所以经过一段时期后他们对界面的细微差异发展出了一种本能直觉。因此，他们的问题是用户行为和程序行为的联系问题：

- 完成该项任务有捷径吗?
- 我能不能让这项任务自动完成?
- 如何根据我的需要自定义界面?

一些专家用户还热衷于一些大多数人不使用而他们却经常使用的功能方面的特殊信息，比如 Microsoft Word 中的公式编辑器(如图 6-5 所示)，或者是 Adobe FrameMaker。

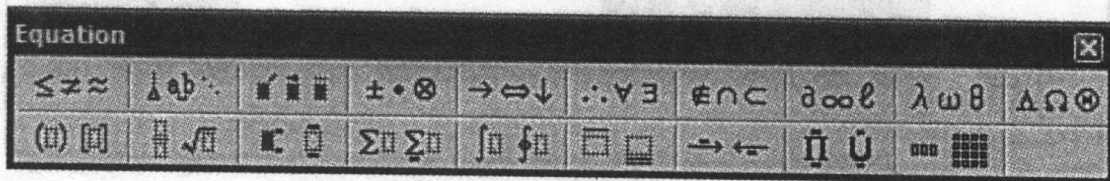


图 6-5 Word 中的公式编辑器

6.3 了解用户目标

因特网改变了人们对界面的看法，也改变了公司对用户的营销方式 (Eisenberg and Eis-

enberg, 2006)。对于用户界面的设计师来说,这既是一个难题也是一个机遇。难题在于现在是用户驱动着产品市场并决定着用户界面设计。机遇在于设计人员可以更好地利用用户目标和用户界面设计之间的断层。

Cooper 和 Reimann(2003)认为,问题在于市场分析师做出的调查研究和设计师做出的界面设计之间有断层。为填补这些断层,Cooper 和 Reimann 为软件工程设计和用户界面设计创建了“目标导向设计过程”。断层表现为研究和提炼阶段之间的3个新的主要措施。

完整的5步“目标导向设计过程”将人种学(一种研究和了解一个人或一个群体的方法)方法的研究、建模及设计组合为5个阶段,顺序如下(Cooper 和 Reimann, 2003):

1. **研究**——该阶段利用观察试验和情景试验以及会谈的形式进一步了解产品的潜在用户和实际用户。研究的主要成果之一是发现了使用模式,这告诉了我们产品使用的目的和动机。例如,如果你想对文字处理器研究调查,其中一个使用动机是相对于手写它能够更快捷地写入和编辑文档。在第9章“可用性”中你将进一步了解观察试验和情景试验。

2. **建模**——研究完成之后,在建模阶段分析研究用户和工作流模式,然后根据这些模式创建用户模型。用户模型是基于用户目标、动机和行为模式的。根据这些用户模型或人物角色,项目组确定每个角色在界面设计中的影响力。在下一章中你将进一步学习关于人物角色的知识。

3. **需求**——在这一阶段,项目组要创建需求,以满足在建模阶段确定的一个或多个人物角色。为了满足这些需求,你需要在用户使用界面的环境下进一步了解用户,这就需要用户和任务分析,这点你将在本章节后面学习到。本阶段的目标是要完成需求定义,以便平衡用户、商务需求与技术需要。

4. **框架**——设计师创建一个交互框架,为程序提供一个结构,这样他们可以在以后加入剩余的代码。该框架综合了交互设计的通用原则和设计模式,为产品创建工作流和行为。部分框架包括输入方法、视角、数据元素、功能元素、群组及群组优先级。你将在第7章“设计用户界面”及第8章“设计网站”中进一步学习界面框架的创建。

5. **提炼**——该阶段完善框架并且纳入设计的详细文档及形式与行为规格说明。本阶

段定义了设计应该做什么，以满足建模阶段确定的每个人物角色的目标及采用这些人角色商务目标。例如，你应识别出在人物角色和商务上都存在的某个问题，如工具不足，无法获取用户反馈；然后确定出你的界面向人物角色和商务提供的解决方案。

“目标导向设计过程”还包含项目组需在设计过程中提出的一些重要问题(Cooper 和 Reimann, 2003):

- 我如何发现用户是什么样的人?
- 我如何了解用户想完成的目标是什么?
- 我的产品应如何运作?
- 我的产品形式应该是怎样的? 例如, 产品应基于 GUI 还是基于网站?
- 用户和我的产品如何交互?
- 怎样将我的产品功能组织得最有效率?
- 我的产品应如何向初级用户进行自我介绍? 换句话说, 初级用户怎样才能知道从哪开始?
- 从技术上来说, 怎样使我的产品外型看起来易懂易控?
- 我的产品如何解决用户问题?
- 我的产品怎样才能使不经常操作的用户变得更在行?
- 对于专家用户来说, 怎样才能让产品感觉有深度? 例如, 如何使其实现自动化, 让专家用户感觉效率更高?

提示

在上述清单中, 文字产品的使用不仅包括用户界面, 还包括整个程序中所有和用户接触的部分(包括错误信息和在线帮助)。只有将界面发给用户的所有信息全盘考虑进去之后, 你才能够有效回答所有这些问题。

“目标导向设计过程”的设计把每一项都考虑进去了; 把猜测全部剔除在设计过程之外, 为每步决策提供清晰的理论依据。注意: 该过程不是一个线性过程。根据需要, 你可能要在不同设计阶段之间反复迂回, 以最大限度地获取信息, 创建一个有效的用户界面。

如果你处于一个产品项目组, 可能使用的也是这个设计过程或者是类似设计过程。这

时你不仅应该注意这个过程本身(在从用户那里发掘信息和提炼文档等方面都相似),你还应尽量利用这个过程连接所有相关工作,如文档与培训。连接你的工作不仅能够确保设计组的所有成员都明白设计过程,而且还可以确保他们能够进行一些交叉研究来采集信息,并给你一个产品交互过程的机会。

6.4 用户和任务分析

用户的心智模型基于多种因素,包括经验、行为、理念及具体情况。在分析用户心智模型时,其实你是在向他们推销。就像前文所述,现在是用户主导着用户界面的市场以及接受度,所以尽最大努力发掘用户的心思就是你当仁不让的义务。

“目标导向设计过程”的“研究”阶段通过定性和定量研究(Cooper 和 Reimann, 2003),调查用户不同情境下的行为。定量研究的信息最客观,但对统计结果的解释有很多种,都可以符合某一观点。另外,定量研究也不能捕捉人与用户界面之间复杂的交互性。

定性研究是一种基于事物的特性而不是数量或测试结果的研究方法。本书仅通过讨论你需要提出的问题以及不同用户类型来定性研究。为了回答这些问题以及了解用户的类型是什么,你需要采用定性研究技术。这些技术包括以下几点(Cooper 和 Reimann, 2003):

- 竞争产品的评估。
- 市场研究的评估,例如计算媒体网站和技术白皮书。
- 研究你正在开发的地区的市场人口统计状况。该研究可包括分析人口统计、地理或行为变量,看是否能够形成任何模式。
- 采访项目关系人(如销售和营销部门)、研发人员、学科问题专家(SME)及其他可能需要的专家。
- 群体访谈,项目组的一个或以上成员采访一组用户。该组人群的选择应基于设计组想通过访谈所证实的研究假设,如了解用户有什么需求等。
- 在一个房间内进行焦点小组采访,让焦点小组回答一系列问题或做出一定选择。
- 进行可用性测试和用户测试。用户界面测试的最有效方法是用户和任务分析,进行这种分析时,应该观察用户在自然环境下工作的情形。这种环境一般是工作场合,但是也不

仅仅指用户的工作间或办公室——它跟用户的工作日和日常执行任务的方式有关。

6.4.1 构建人物角色

用户和任务分析的一部分内容是构建人物角色，这同时也是“目标导向设计过程”中建模阶段的部分内容。在进行用户和任务分析之前，你需要通过构建人物角色来了解用户是什么样的人。就像你在本章开始阶段读到的一样，人物角色是建立在不同用户性格分组上的用户模型。

Eisenberg and Eisenberg (2006) 从销售角度(这是你努力提升用户界面的根本原因)探讨了人物角色的创建：向用户证明该界面值得使用。人物角色将三种不同维度的信息连成一个中心角色：

- **人口统计**——该项将一部分人物角色特性划分出来。例如，人口统计数据显示了用户性别、地理位置及收入等数据。
- **心理**——该项将一些用户需求划分出来，并且确定了每个人物角色可能会提出的问题。例如，一个本能型的人和一個竞争型的人会问不同的问题，也希望得到不同类型的信息。
- **拓扑**——该项可以通过确定说服工作的复杂度来划分人物。复杂度基于用户的感知和经验。

对于拓扑，Eisenberg and Eisenberg 为销售中的说服过程制定了一种四维模型：

- **需求**——这是用户要求某个产品或服务的紧急程度。
- **风险**——这是用户将这种特点作为一种体验或自尊而愿意接受的风险值。
- **知识**——指用户对于产品知识了解多少。这种知识可以影响到需求和风险。例如，如果有些人感觉他对产品或服务信息了解得不多，那么该用户的风险系数就较高。
- **集体共识**——指明白说服过程中需要对多少人及什么时候进行说服。

从三维信息(人口统计、地理及拓扑)中获得的信息可以逐渐反映出你的角色类型的动机，这样你就可以了解用户做某些事情的动机是什么。

1. 使用人物角色的好处

使用人物角色可以克服你在用户设计过程 (Cooper 和 Reimann, 2003) 初期碰到的问

题。当你开发用户角色的时候，你需要联络开发人员、设计人及其他项目关系人，了解他们是怎样理解用户需求的，这样可以促进团队更好地协作。项目组利用这些输入信息来确定一个产品的功能及其运作方式，然后就会为预期用户提供几个设计选择用于测试。

通过人物角色来了解用户，有助于减少或消除在产品研发阶段容易产生的设计问题 (Cooper 和 Reimann, 2003)：

- **无固定的用户概念**——设计师、开发人员或其他项目组成员脑中应该有用户这个定义。根据这个用户定义来设计界面，才能够说他是为用户服务的。确定软件的一般用户之后，项目组才能够按研究结果显示的客户需求(而不是按某个具体团队成员的主观想法)来设计界面。
- **自我参考型设计**——指设计师或开发人员将他们自己的观念注入到项目中，并宣称这是用户的想法。换句话说，设计师或开发人员认为他就是一般用户。
- **为边缘案例进行设计**——指项目组为每一件可能发生但一般不会发生的偶然事件进行设计。有了通过研究得出的人物角色清单，项目组可以把精力集中在任务和功能上，对大多数用户来说，这才是最重要的。
- **瞄准非最终用户**——我们往往会瞄准审查软件的人，或你的 IT 部门及客户技术服务部门中那些要管理和支持这些软件的人。然而，明确谁才是你的真正最终用户之后，你的项目组才能为这些用户进行设计。

然后，针对不同人物角色进行设计选择的测试，项目组可以检查设计的有效性。从这些设计选择中得出的反馈信息有助于项目组对最终设计达成一致并提交。

人物角色同样适用与产品相关的其他开发工作。因为你可以将在一个项目的角色创建中学到的知识应用到另一个项目上。

2. 融合行为、性格及目标

创建一个角色，需要你将用户的行为和性格与他们的目标相融合，以叙事的方式说明人物角色 (Cooper 和 Reimann, 2003)。

本章已经将项目组目标作为“目标导向设计过程”的一部分进行了讨论，但是用户有其自身的目标和动机，这些目标包括以下方面：

- **生活目标**——这些目标通过人物角色动机来实现，如想成为公司的总经理，想什么都学，以了解软件的一个特定领域或地方。
- **经验目标**——它们是通过深层次的理想来实现的，如不犯错误进而感觉能力不错，避免愚蠢感，尽可能地欣赏自己等。
- **使用具体产品的最终目标**——它们可能包括：使特性和服务最佳组合从而找到最划算的方法，快速高效地完成一个任务，完成一个短期及长期目标(如完成一个客户服务表单或完成一件产品的研发)。

用户目标并不是你创建人物角色时考虑的唯一目标。你还必须考虑公司内部或外部其他项目关系人的目标。除此之外，还有3个其他目标是你在设计中必须考虑的(Cooper 和 Reimann, 2003)：

- **技术目标**——这些目标包括确保程序可以在所有现代操作系统中运行，并且抗病毒、蠕虫及其他侵扰。如果是硬件产品，那么你的产品设计组必须确保其安全可靠，功能正常。如果你的产品不安全、不可靠，没人会买它。
- **客户目标**——客户目标和用户目标不一样。客户不准备使用该产品，而是将产品送给其他人去使用。因此，客户关注的是你要确保产品接收人对客户买的礼物感到满意。客户也可能会使用产品，但是不常用，比如一个控制网络服务器，这种情况下，客户更关注程序的稳定性和安全性。
- **商业目标**——举个例子：为开发该公司的产品降低成本并提高效益。商业目标在实现同时也可以向项目关系人表明：在用户界面设计和实用性测试上的投资是有回报的，这证明了该技术是有效的，而且应该应用于其他项目。

这些目标不能取代用户目标，但是就客户目标而言，他们可能会要求你创建新的人物角色，这点你会在本章后面进一步学习。你还可以将这些目标作为提议的一部分，向项目关系人表明，这样你可以向他们表明优秀设计是如何实现这些目标的。

3. 与用户会谈

人物角色代表的是个体，是通过观察现实中的人之后合成而来的。观察法有以下形式(Cooper 和 Reimann, 2003)：

- 在用户的工作情境之外进行采访，比如在会议室中进行采访。

- 请公司中能够提供信息的其他人来采访用户，比如与用户打过交道的销售与市场人员，以及其他项目关系人，如学科问题专家(SME)。
- 市场研究数据，如问卷调查，以及其他形式的直接反馈，如交易量、文献综述、市场分析研究以及诸如白皮书等形式的其他研究方式的反馈信息(见图6-6)。

采访用户时，按照第5章“用户行为”所述，把这些谈话对象和行为变量(以及你或你的项目组想找的任何其他人口数据，如年龄组)形成映射关系。之后将数据模式和人物角色假设进行比较，确定人物角色假设是否需要进行修改。例如，如果你的数据模式显示有意外情况，可能你需要进行新的采访，或将这些行为变量加到一个新的人物角色上(而之前你没想到要加)才能解释这些没预期到的行为。

这种对人物角色信息的深度挖掘可以使你深度理解用户，而高级曲线与之相比则只提供很少数量的用户信息，如用户姓名及一些你在市场分析研究中发现的人口统计数据。

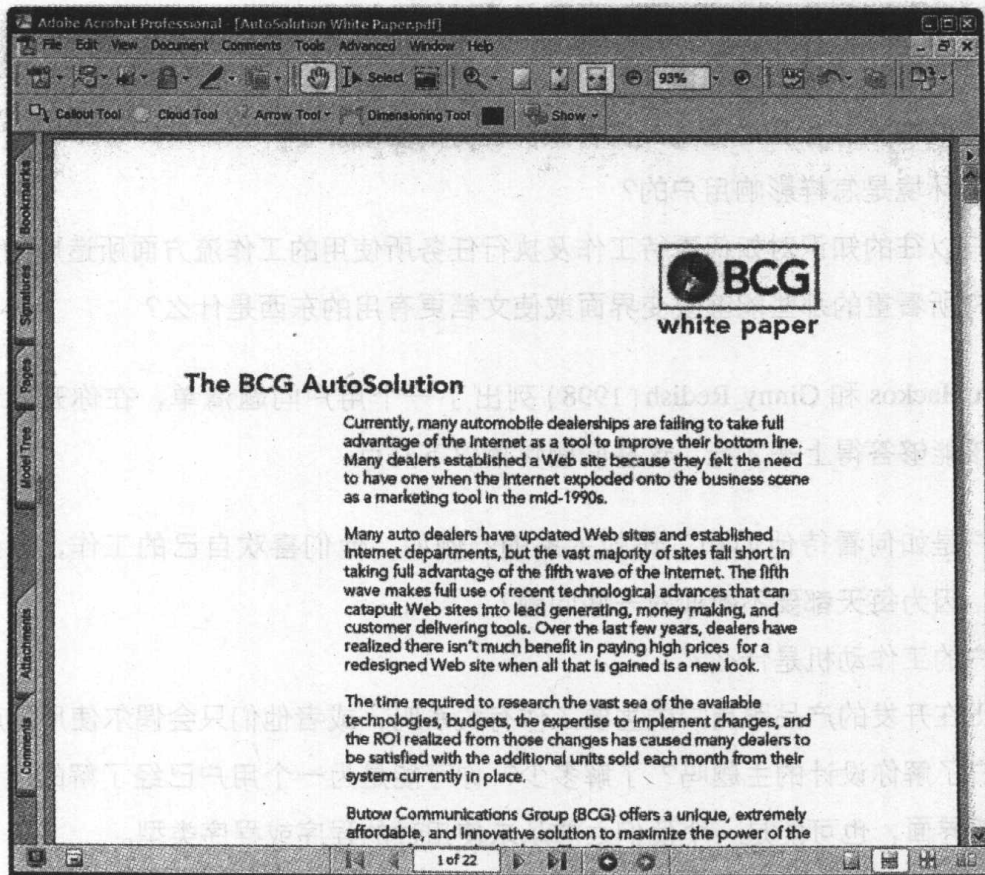


图 6-6 白皮书样例

6.4.2 观察用户行为

在会议室中采访用户，你是没有机会看到用户的感知和观念如何与对工作环境的正确认知相协调，并如何对变化做出反应的。工作环境不仅包括用户在自己办公室中所做的工作，也包括他们和大环境的互动，这包括他们通过会议、电话、电子邮件以及一些物理环境和公司内部其他人进行的沟通。

因此，“现场”进行用户和任务分析，从而获得用户在工作环境中的行为是非常重要的。用户和任务分析是通过观察用户行为来了解普通用户的一个过程(Hackos 和 Redish, 1998)。仅仅通过观察产品用户，就可以使你对用户如何使用产品及如何尽可能地减少他们的问题获得一个清晰的答案——这叫作极端可用性(Donoghue, 2002)。用户和任务分析将使你明白以下问题：

- 用户的目标是什么，他们想要的是什么？
- 用户做哪些事情来达到目标？
- 用户的性格、社会经验及文化背景对任务的影响。
- 物理环境是怎样影响用户的？
- 用户以往的知识对如何看待工作及执行任务所使用的工作流方面所造成的影响。
- 用户所看重的那些能够改变界面或使文档更有用的东西是什么？

JoAnn Hackos 和 Ginny Redish (1998) 列出了一个用户问题清单，在你开始进行观察前，你应该能够答得上来才行。这些问题包括以下几点：

- 用户是如何看待他们和工作的关系的？例如，他们喜欢自己的工作，还是害怕上班，因为每天都要不断面对一些问题？
- 用户的工作动机是什么？
- 你正在开发的产品跟用户的主要工作有关系吗，或者他们只会偶尔使用该产品？
- 用户了解你设计的主题吗？了解多少？你可能是为一个用户已经了解的产品类型设计新界面，也可能是一个他们之前从未使用过的程序或程序类型。
- 用户有过做类似工作或任务的经验吗？
- 用户做这项工作的时候，会应用哪些技术能力和工具知识？

- 用户是喜欢接触技术还是对技术很头疼？有的人使用技术是因为喜欢技术，有的人使用技术是因为没办法，不得不用。
- 用户是喜欢从书面文档中了解信息还是喜欢使用其他方式？人们对不同类型的媒体和培训的反应是不一样的。有的人喜欢自学，而另一些人则喜欢在群组中学习。
- 用户喜欢用什么语言？如果你正在为多种文化的用户创建软件，那么语言是你设计用户界面时需要重点考虑的因素。
- 用户喜欢新的体验，还是倾向于“只要能用就不要修改”的理念？有的用户喜欢使用手头上的老产品，让他们去学习一些新东西会使他们很不高兴，除非新产品能极大提升他们的工作流程和业务水平。

你应该争取采访尽可能多的用户，了解他们的“天生习惯”。这可能需要你与多人合作以获取尽可能多的信息。例如，你可以采访一组用户，然后对另外一组用户进行现场用户和任务分析。另外一个人则现场分析你采访过的用户组，以及采访你进行过现场分析的用户。

在“提炼”阶段的可用性测试过程和建模这一阶段中，你的现场用户和任务分析可能会遇到一些阻力。在第9章中你将学习如何解决这些问题。

6.4.3 人物角色评价

从研究中了解了用户行为和用户目标之后，就应开始审视你的数据并形成人物角色 (Cooper 和 Reimann, 2003)。

在形成你的人物角色时，一定要核实数据的完整性和特征性。若数据有缺陷，如丢失了需要项目关系人关注的的数据，则需填补数据。如果发现有2个人物角色非常相似，你可能需要删除一个，或将其中一个进行重大修改，使其决然各异。每个人物角色必须在至少一个重要行为上有不同之处。

制订好人物角色清单将有助于你的项目组及项目关系人来为每个人物角色制订叙述性故事，显示每种类型的人物角色的特点。这些叙述性故事可以帮助团队了解用户。例如，“Lisa 是一个每天管理8个病人的护士，她非常忙碌，如果在两次点击后界面还不能让她达到预定目的，她就要不耐烦了。她的目的是尽快得到病人的可靠信息。”图6-7中的人物

角色实例就是 Lisa。

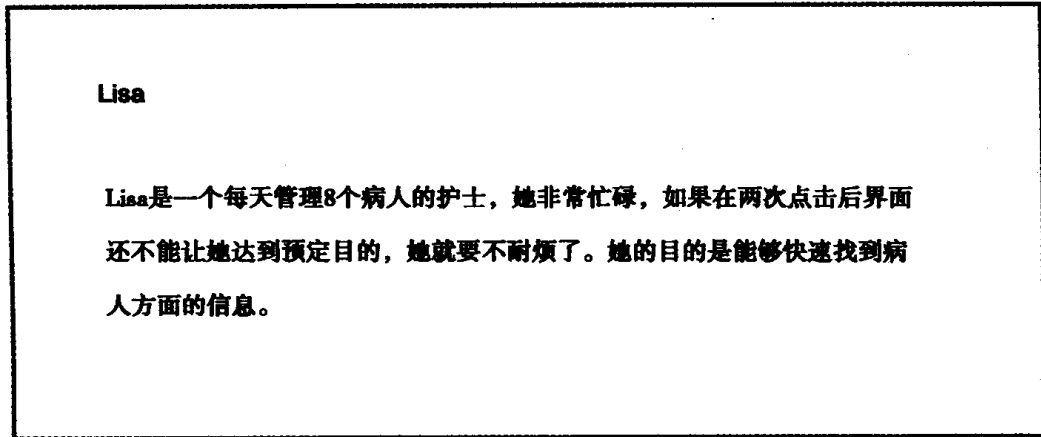


图 6-7 人物角色样例

你也可以为这样的典型用户配备一张库存图片，使人们对该人物角色的定位获得一种视觉暗示(线索)。对于 Lisa 这个人物角色，你可以选一张表情疲惫、身着制服的护士图片。注意一定要为人物角色选择合适的库存照片，否则团队成员和项目关系人就被你搞糊涂了。

人物角色的分级

一个人物角色只是设计的一个候选方案，不要试图开发一个想要吸引每个人的设计。你需要决定产品的目标用户。在你为目标群中的每一个人修改设计之后，就可以合并那些设计以吸引更多可能的用户。经过仔细平衡的将是用户界面的最终设计。

你需要将人物角色归为 6 种类型中的一个，然后排列顺序 (Cooper 和 Reimann, 2003):

- 主要人物角色——这是一个界面的主要目标。如果主要人物角色是用户界面的目标，那么其他所有人物角色至少要对界面满意。如果设计中没有明显的主要人物角色，产品可能就需要为多个人物角色设计多个界面，否则程序或网站恐怕就要做太多工作了。如果你确定了一个以上的主要人物角色，产品的特性可能就太多了。
- 二级人物角色——如果加上一两个具体需求，这些人物角色可能会对界面特性感到满意。如果你的二级人物角色多于 2 个，那么你的程序特性可能就太多了。

- 辅助人物角色——这些人物角色要么对主要人物角色界面十分满意，要么对二级人物角色界面感到十分满意。
- 客户角色——这些人物角色代表了客户需求，这些客户本身不准备使用产品，但是想确认使用软件的人是否喜欢这个用户界面。
- 接受服务的人物角色——这些是些会受产品使用直接影响的人物角色。Lisa(护士)的病人不是界面的直接用户，但是由于 Lisa 改善了对他们的照顾状况，所以他们也是受惠者。
- 消极人物角色——这些人不是产品用户，也不应成为设计目标。例如，护士助手一般不会去使用 Lisa 使用的界面。

你应该创建多少个人物角色？取决于你的实际情况。例如，你的项目关系人可能比较关注特定的用户性格，那么你的人物角色应该表现出这些性格。你至少应该有 2 个角色 (Eisenberg and Eisenberg, 2006)，其中一个应该是主要角色，第二个可以是另外 5 种的任何一个。

现在你已经了解了如何对人物角色分级，你需要将你的目标和用户目标融入设计，这点你将在下一章中继续学习。

提示

尽管可能很诱人，但是不要在产品线上的所有产品中使用相同的人物角色。你可以在一个产品中应用很多你在人物角色创建方面学到的知识，但是不同产品中的人物角色有着重大差异——用户使用产品的背景不同。例如，用户使用网络浏览器和电子邮件系统，你可能会认为这 2 种行为之间有着密切关系，但是你会最终发现，用户使用 2 个程序的背景是截然不同的。

6.5 案例研究：创建主要人物角色

有关性格类型和概念模型的采访问题只是采访过程的一个开始。你和 Evan 现在需要问更进一步的问题来获知用户的个人行为以及目标，这样你才能为用户组创建一个主要角色。

模仿 Hackos 和 Redish 建议的下列问题清单，你自己组合一个问题清单。

- 你喜欢你的工作吗？为什么喜欢，或为什么不喜欢？
- 你工作的动机是什么？
- 数据库产品和你的主要工作有直接关系吗？也就是说，你需要每天用该产品来完成工作吗？
- 对于你所设计的主题，你了解多少？
- 工作上需要什么技术或知识？
- 你是如何对待技术的？你喜欢技术还是不得不忍受技术？
- 你是倾向于从书面文档中获取信息还是喜欢使用在线帮助？你是怎么看待目前系统中的文档的？
- 你愿意汲取新经验，还是认为如非必要不予理会？

由于所有的雇员在软件中使用的都是同一种语言，所以你和 Evan 决定不提这个问题，以后再说。后来，在你完成初次会谈后，Mike 告诉你 Mike 的自行车公司决定在将来跨国发展，这时多种语言支持就成了一个问题。

你和 Evan 决定当面问问题，而不是给用户一个问题清单让他们自己填写答案。因为你想知道用户的非语言行为，了解他们回答问题时的说话语气。用户也可以获得一些额外的信息来更好的回答问题。

例如，在你采访 Jay(市场经理)时，他告诉你他喜欢自己的工作。但是你可以从他的声音语调和肢体姿势中洞察到他在工作中有挫折感。他还告诉你他认为 Mike 可以采取更多措施来拓展 Mike 的自行车店的市场领域，但是当谈到对数据库应用程序和网站进行修改时，他的语调和姿势变得非常放松，因为他相信这些改动会有助于实现他的目标。因此，“你是否喜欢你的工作”这个问题的答案是否定的，但是你明白了新的数据库应用程序会在将来激励他喜欢自己的工作。

利用这些性格类型、概念模型以及个体行为与目标方面的信息，你和 Evan 评估你们的信息，并根据以下特性将项目的 10 个成员划分为不同类型：

- 回答者的性格类型

- 概念模型
- 个体行为和目标


当你将人们划分成这 3 种类型时，会有很多的回答者拥有同一特征，这些人就成了你的主要人物角色。主要角色人物如果看起来像如下情况那就大功告成了：

- 给人物角色起个名字“Bob”（毕竟 Bob 这个名字听起来比较友好）。他是一个熟悉系统内部和外部情况的人。
- Bob 性格外向，并且是一个直觉型、思考型及判断型的人。因此，他享受工作并热衷技术。
- Bob 是一个已经对系统很了解的人，也喜欢新的体验，但前提是新事物能够简化他的工作。Bob 对他的部门也非常清楚，可以向你清晰地说明他所在部门在公司中的作用。
- 因为 Bob 每天都使用系统，所以他渐渐形成了一些改进系统的想法，并且对在该系统下能运行和不能运行的工作作了很多判断。
- Bob 希望看到数据库互相连接，也希望数据库能够让他实时查看进程。
- Bob 不喜欢的情况是无法在网上马上查找到一个产品或一个物品。他认为目前系统的查找功能空洞无效，让他感到非常受限。
- Bob 想在系统的适当地方设置更快速的查找信息的方法。例如，应该在所有页面上都设置查找功能，这样能够在网上立即查找任何需要的信息。同时，查找功能应反馈相应查找结果，并且在查找结果页面中应提供快速返回上一页面或系统中任何其他页面的功能。

将这些信息连同 Bob 照片的实体模型及人物角色测试置于一张纸上（见图 6-8），给团队中所有成员都复印一份，以便在纸上原型测试期间共享。

在下一章中，你将学习把用户界面特性（如视觉与听觉暗示）应用于你的纸上原型测试中。

主要人物角色：BOB



- 给人物角色起个名字“Bob”（毕竟Bob这个名字听起来比较友好）。他是一个熟悉系统内部和外部情况的人。
- Bob性格外向，并且是一个直觉型、思考型及判断型的人。因此，他享受工作并热衷技术。
- Bob是一个已经对系统很了解的人，也喜欢新的体验，但前提是新事物能够简化他的工作。Bob对他的部门也非常清楚，可以向你清晰地说明他在部门在公司中的作用。
- 因为Bob每天都使用系统，所以他渐渐形成了一些改进系统的想法，并且对在该系统下能运行和不能运行的工作作了很多判断。
- Bob希望看到数据库互相连接，也希望数据库能够让他实时查看进程。
- Bob不喜欢的情况是无法在网上马上查找到一个产品或一个物品。他认为目前系统的查找功能空洞无效，让他感到非常受限。
- Bob想在系统的适当地方设置更快速的查找信息的方法。例如，应该在所有页面上都设置查找功能，这样能够在网上立即查找任何需要的信息。同时，查找功能应反馈相应查找结果，并且在查找结果页面中应提供快速返回上一页或系统中任何其他页的功能。

图 6-8 Bob 和人物角色文本

6.6 小结

本章首先讨论了用户心智模型，区分了实现模型(表示产品实际是如何工作的)和用户心智模型(认为用户在使用产品时不需要了解与工作原理有关的方方面面)。用户心智模型

要求设计师应将用户界面设计得简单化。

之后介绍了经验钟形曲线。你了解到用户划分为3种经验类型：初级用户、中级用户及专家用户。大多数用户是处于知识的中间范围。“永久中级用户”最为关注与产品相关问题的具体答案。初级用户想了解一些诸如“从哪里启动”问题的答案，而专家级用户想知道如何才能更高效地利用产品。

接着，本章介绍了“目标导向设计过程”。你了解了“目标导向设计过程”分哪5个阶段，为什么该过程有助于弥补用户界面开发中的缺陷，设计过程是如何为决策提供清晰的理论依据的，以及设计过程回答了哪些问题。“目标导向设计过程”不是一个线性过程，为了创建一个有效的用户界面，你可能要在不同阶段根据实际要求不断往返迂回。

本章的最后讨论了用户和任务分析，以及如何利用分析获取尽可能多的信息来生成人物角色，人物角色代表的是具体个体类型的需要。你学习了构建人物角色及人物角色是如何减少4个主要设计问题的：设计时“无固定的用户概念”、自我参考型设计、设计“边缘”案例、瞄准非最终用户。你还学习到人物角色融合了行为、性格及目标，生成了一个解释人物角色的叙述性故事。最后，你学习了将你的人物角色进行分级，将他们归于6种类型中的一个：主要(最重要)、次要、辅助、客户、接受服务及消极角色。你应创建的人物角色的数量取决于你的实际情况。

复习题

在进入第7章之前，是时候回顾一下本章所学了。问自己以下问题，然后参考附录A来核对答案。

1. 为什么设计师仍然坚持机械时代的标准？
2. 永久中级用户是什么样的人？
3. 初级用户总会问的问题有哪些？
4. 中级用户总会问的问题有哪些？

5. “目标导向设计过程”的5个阶段分别是什么?
6. 为什么要进行用户和任务分析?
7. 人物角色将信息的哪3个维度综合在了一起?
8. 用户目标有哪些类型?
9. 为什么应该“现场”进行用户和任务分析?
10. 为什么需要对人物角色进行分级?



设计用户界面

“养成一种习惯，即观察人们行为的意图，并注意在其可行时他们会如何做。

如果能让这种习惯更具意义，先从自身实践开始。”

——Marcus Aurelius

本章内容提要

- 设计基于人物角色的交互框架
- 交互设计
- 软件姿态
- 界面行为
- 帮助用户寻找信息
- 与用户交流
- 精炼形式与行为

本书引导读者逐步理解设计过程和用户，包括以下主题：

- 了解与用户界面设计有关的内容(第2章“概念与问题”)
- 为项目关系人制作商业计划(第3章“制作商业计划”)

- 学习创建优良设计所必需的工具知识(第4章“优良设计”)。
- 获取用户信息并了解用户(第5章“用户行为”;第6章“用户分析”)。

现在你已经完成了商业计划并且了解了用户,现在需要运用这些用户信息开始定义用户界面的需求了。你应该以第6章讨论的用户人物角色为基础开始设计交互框架。

之后你将学习交互设计以及如何才能使界面合乎常理、目的明确、注重实效及优雅——这是第4章里讨论过的优秀设计所需具备的4个特点。部分交互设计是你想使用的姿态类型。你将会学习到4种不同的GUI姿态及其使用的适宜时机。

最后,你将深入了解界面行为及如何设计界面才能使你有效地跟用户交流并帮助他们找到所需信息。当你在框架中确定了用户的需求及框架中各元素应如何运作,你就可以进行“目标导向设计过程”中的第5步和最后一步了:精炼形式及行为,完成设计。

7.1 设计基于人物角色的交互框架

Cooper 和 Reimann(2003)的“目标导向设计过程”的第3个阶段为“需求”阶段。你需要根据人物角色来定义计划的需求,然后才能设计一个框架。Cooper 和 Reimann 将需求的获取定义为5个步骤(没错,是另外5个步骤)。

- **创建问题报告和愿景说明**——基于你对人物角色和公司目标的双重理解,用问题报告和愿景说明来创建设计目标。
问题报告定义了人物角色目前面临的情况及(若可能)对业务造成怎样的影响。例如,“护士无法快速获得所需的病人信息;因此,情绪低落,结果会造成更多的护理过失,病人平均住院时间延长1天。”
愿景说明应解释新的用户界面将如何帮助用户和公司。例如,“新的用户界面设计使得护士能够执行任务A、B、C、D,结果情绪高涨,护理过失更少,病人出院时间更早,转床更快。”
- **头脑风暴**——和项目团队开一次或更多的头脑风暴会议,看完人物角色数据之后,团队将会明白团队成员之间存在哪些意见偏差。开一次或更多的头脑风暴会议能够发现一些想法,团队可以立刻或在将来某个时间执行这些想法。头脑风暴

会议应集中于一个主题。例如，头脑风暴会议应关注哪些用户界面元素可以满足具体目标。

- **明确人物角色的期望**——每个人物角色对于产品都有自己的心智模型。必须明确每个人物角色的愿望和期望、行为、态度、偏见及其他会影响他们的因素。人物角色提出的或未提出的一些信息，比如他们想要执行的任务，可以作为人物角色需求的参考。
- **构建情景实例**——与人物角色及其行为相关的这些故事，会帮助你明白每个人物角色是如何使用新版系统及改进版系统(其中包括新版用户界面)度过日常一天的。这些实例不讨论形式和功能，只涉及用户行为和界面。
在每个情景实例中，你不仅要确认人物角色日常实例中的环境和组织关系，还要谈到每个主要人物角色及二级人物角色在系统中的重点部分，以及系统中需要和他接触的其他人物角色。例如，主要人物角色是个护士，和某个人物角色共事；但二级人物角色是一个管理人员，所共事的人物角色与护士共事的那个完全不同。这种差异可能会显示出每个人物角色使用系统时会有不同。
- **明确需求**——创建完情景实例后，对之进行分析，确定每个人物角色的需求。人物角色有3种类型需求：数据、功能及情景。

系统中的目标和信息构成了数据需求。例如，打字时文字处理文档中的字母就是你需要注意的目标。

在系统中，目标上需要进行的操作组成了功能需求。例如，如果你想在一个文字处理文档中建立一个文本框，则必须对之进行操作。

一系列对象之间或一系列控制之间的固定关系和可能关系组成了情景需求。例如，当你把喜欢的网页保存在IE的“收藏页”中时，IE必须更新最新收藏页和之前收藏的其他所有网页清单。

注意：上述过程不是一个线性的单次处理过程，而是个重复过程。例如，你可能想只经历一次设计过程，开一次头脑风暴会议，然后根据会议结果精炼人物角色期望、背景实例或需求。

7.1.1 现实世界的需求

当你制定完产品需求的创意部分之后，就该把精力放在了解现实世界对产品的需求上了。这些需求包括以下部分 (Cooper 和 Reimann, 2003)：

- 商业需求，包括：商业开发模式、期限、定价结构。
- 客户和合作伙伴的需求，包括：安装、配置、客户支持及许可协议。
- 技术需求，包括：你所使用的操作系统及产品要求的形式因素。例如，产品可能是个网站，需要特定的分辨率来现实以便于所有的用户都可以在网页上看到该信息。

这些其他需求可能会促使你重新考虑或精炼在需求过程中产生的一些想法，还可能因此需要召开更多的头脑风暴会议。

7.1.2 定义框架

当你做好需求清单之后，就该定义框架了。框架的创建是一个 6 步过程 (Cooper 和 Reimann, 2003)：

1. **定义形式因素及输入方式**——你是在创建台式计算机操作系统中 (比如 Windows) 的一个用户界面？还是在创建一个网站界面？或者是在为手提电脑操作系统创建一个界面？回答决定了软件姿态，关于这点你将在本章进一步学习。

你还需要弄明白人们将如何和界面交互。例如，人们是用键盘和鼠标、语音界面，还是手提电脑上接触显示器所用的识别笔？实际上，有大量的输入选择，为此你可能需要制订计划。

2. **定义视图**——你需要确定用户在其系统中的不同背景下会看到什么，这样他才能合理组织屏幕上的信息。例如，对于不同任务你也许需要不同的视图，比如打开一个对话框来打印一个数据表。但是如果元素之间有相互关系，比如一个数据表及其生成的图标，那么你就可以定义一个综合了数据表和图标的视图。

3. **定义功能和数据元素**——创建完视图清单之后，你需要知道目标 (比如屏幕窗口和控件、按钮及图标) 在屏幕上是如何显示的。用户所使用输入设备的类型将影响到你在屏幕上使用哪种功能与数据元素。图 7-1 给出了 Microsoft Word 中一个输入设备的实例。

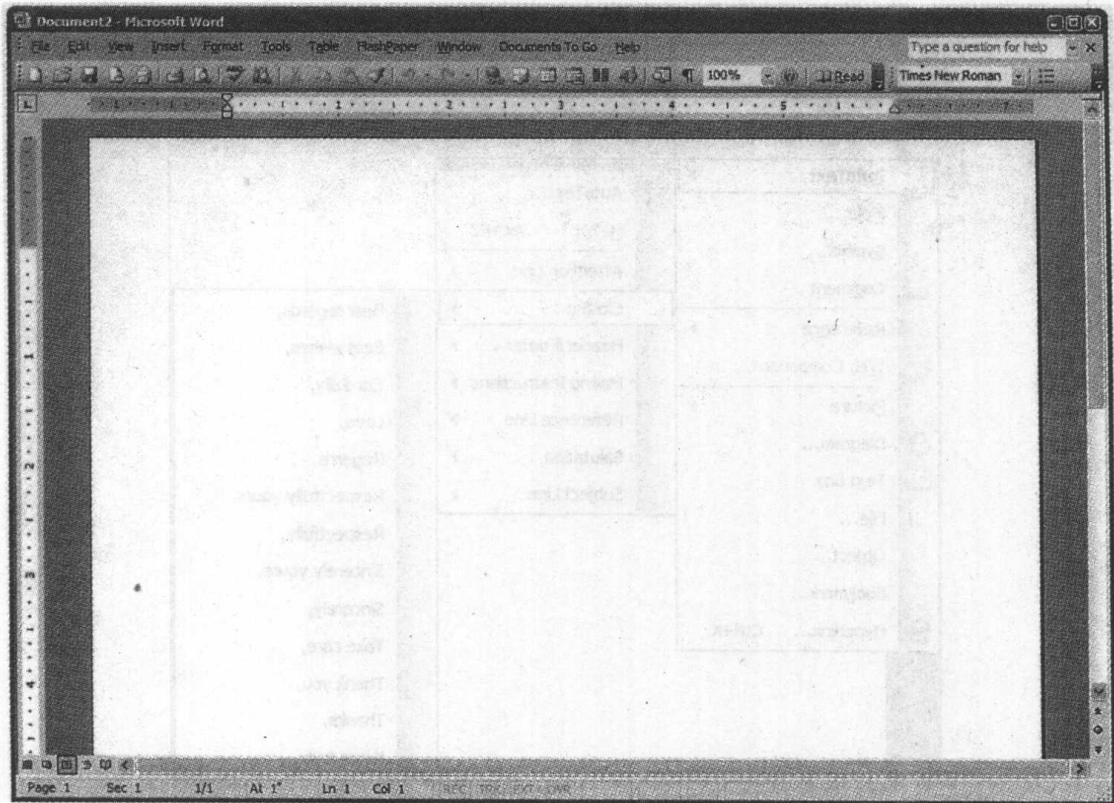


图 7-1 可以用多种输入设备来操作 Microsoft Word

4. **确定功能组及其顺序**——现在你已经知道了什么是功能和数据元素，可以开始将这些元素归类分组，安排好组包，为各组建立顺序关系。以人物角色类型为参考来组织功能和数据元素。比如功能组可以用菜单栏，菜单栏里建立一个组名称，名称下面是单个菜单选项。清单中的一些特性是组本身可以在序位体系中打开一个二级特性清单。比如，当你在“Insert”菜单中的“AutoText”里增加一条结束问候语时，如图 7-2 所示。

5. **勾画交互框架**——你可以绘制用户界面设计的顶层结构草图(“大图”)，并标明界面中不同目标之间的相互关系。此时你不必深入研究界面的某个特定部分，这样做会导致你和团队其他成员分散精力。

6. **构建关键路径情景**——此时你在第 4 章中学到的纸上原型和故事板技术开始发挥作用了。根据人物角色，和团队一起开发用户将在界面中看到的实例，然后创建故事板，以展现当用户完成任务时每一条交互的路径，如图 7-3 所示。每个故事板代表了一幕，当用户开始一个动作时，例如点击一个按钮，故事板上将展现此时屏幕的样子。

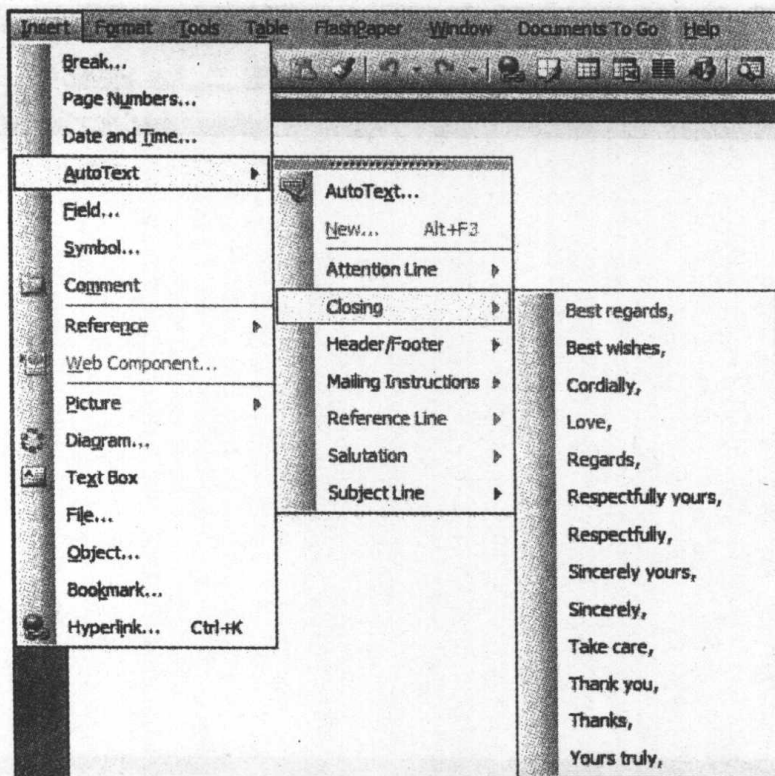


图 7-2 功能组及其顺序的一个菜单实例

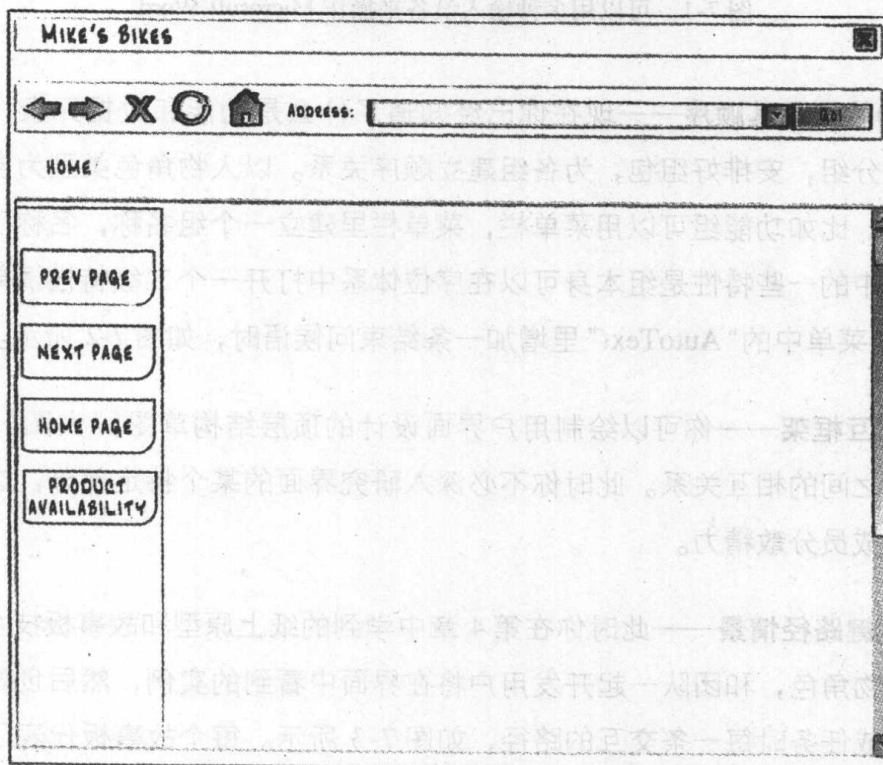


图 7-3 展示情景和方案草图的故事板

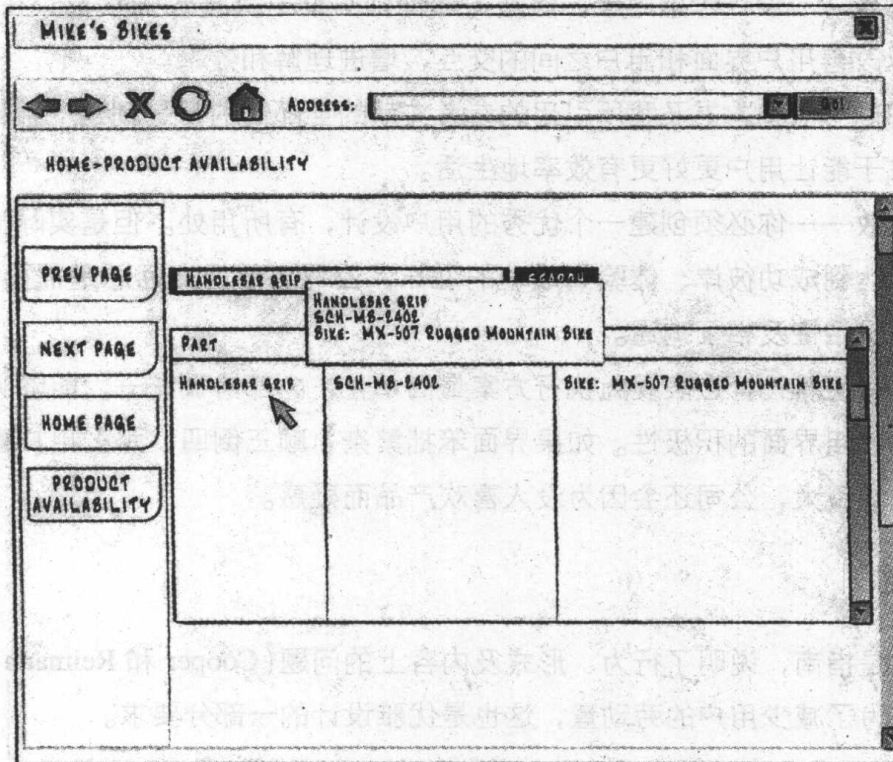
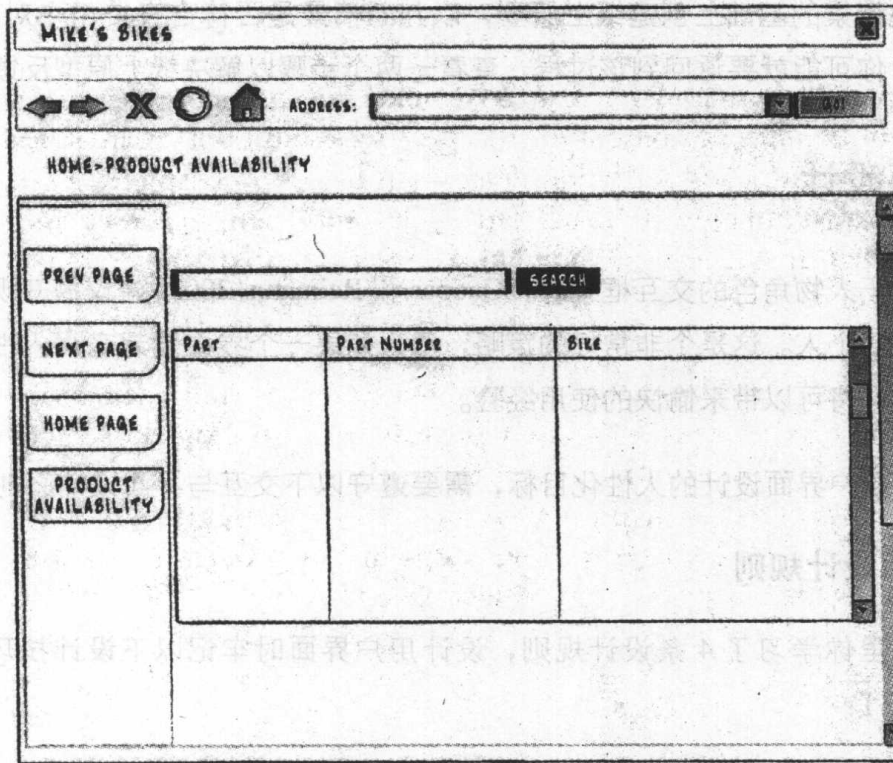


图 7-3 (续)

注意：这个过程也是可以重复的。当你精炼并完成关键路径情景后，就可以在第4章中讨论的那些情景的基础上创建纸上原型，以判断情景是否符合直接用户对原型的操作。如果不符合，你可能就要返回到该过程，查看一两个步骤以解决纸上原型反馈的问题。

7.2 交互设计

在创建基于人物角色的交互框架时，Cooper 和 Reimann(2003)建议抛开现实情况，将操作系统看成一个人。这是个非常好的策略，因为如果一个界面看起来很人性化，对人的需求反应迅速，将可以带来愉快的使用经验。

为了达到用户界面设计的人性化目标，需要遵守以下交互与界面设计原则和模式：

7.2.1 应用设计规则

在第4章里你学习了4条设计规则，设计用户界面时牢记以下设计技巧(Cooper 和 Reimann, 2003)：

- 合乎常理——设计考虑周全，能够实际性地有助于提高人的状况。在界面设计中，你可以改善用户界面和用户之间的交互，增进理解和效率。
- 目标明确——整本书及我所引用的参考文献的全部目的都是创建一种界面设计，其用意在于能让用户更好更有效率地生活。
- 注重实效——你必须创建一个优秀的用户设计，有所用处。但是实际上，为了让一个设计达到成功彼岸、体验到成功的滋味，公司各部门之间总是需要附加的思忖，这需要灵活性及相互沟通。
- 优雅——优雅设计应表现为执行方案最简单化，内部协调统一，能够吸引用户并调动他们使用界面的积极性。如果界面笨拙繁杂，颠三倒四，那么用户喜欢使用它的可能性就不大，公司还会因为没人喜欢产品而疑惑。

7.2.2 原则

原则也就是指南，说明了行为、形式及内容上的问题(Cooper 和 Reimann, 2003)。设计这些原则是为了减少用户的劳动量，这也是优雅设计的一部分要求。

你设计时需要依照3个级别的设计原则：

- **概念级原则**——它定义了产品是什么。在第 5 章、第 6 章及本章中，你了解到如何使用项目组接近用户，定义用户喜欢的产品。
- **交互级原则**——它定义了产品在一般情况和特殊情况下应如何运作。本章将讨论图形用户界面的交互级原则，第 8 章“设计网站”将探讨网站的交互级原则。
- **界面级原则**——它定义了界面的外表和感觉。在本章和第 8 章中探讨这些原则。

如果公司已经制订了样式指南，可能你可以用其中的一些原则来帮助你创建设计原则。但在大多数情况下，公司是不会深入涉及你和项目组开发用户界面所需要的详细等级。（你可以用研发过程中学到的原则来扩充公司的样式指南，事实上，如果你将此视为第二目标并牢记于心，创建一个符合公司首选“像模像样的东西”的可能性就更大。另外一个好处是由于优秀产品适合公司系统，会被更广泛地认可。）

还要注意，这些原则仅仅是指南，并不能为如何创建设计行为提供完全的指南信息。例如，你可以使用用户的反馈信息来指导你和项目组的设计过程。

7.2.3 模式

开发产品时，你将开始创建模式，解决设计中出现的问题。然后你可以用它解决当前项目中的问题。你也可以把这些问题放在开发原则/样式指南中，作为在其他项目中培训和应用这些模式的方法。Cooper 和 Reimann (2003) 列出了 3 种交互型设计模式。

- **姿态型**——它有助于确定产品与用户之间的位置关系。你将在下一节学习更多的软件姿态。
- **结构型**——它解决程序中与数据管理相关的问题，包括信息如何显示及用户如何进入和管理程序中的数据和选项。例如，你可能会注意到 Microsoft 正致力于 Office 软件程序（如 Word）顺序版本的结构问题。
- **行为型**——用单独数据或功能对象或对象组来解决特殊的交互作用问题。

7.3 软件姿态

创建用户界面设计时，人物角色会向你提供他们将如何使用程序的好主意。设计一个界面时，可以使用一个反映主要人物角色工作方式的姿态。注意：用户不大可能仅仅使用其中的一个姿态，而是用好几个，这取决于他们要完成的任务。

有4种基于桌面的GUI姿态(Cooper和Reimann, 2003):

- **独占型**——独立的应用程序是一种全屏幕的保持用户长时间注意的程序。例如,独占型应用程序包括 Microsoft Office(如图7-4所示)以及制图程序,如 Adobe Photoshop 和 Illustrator。

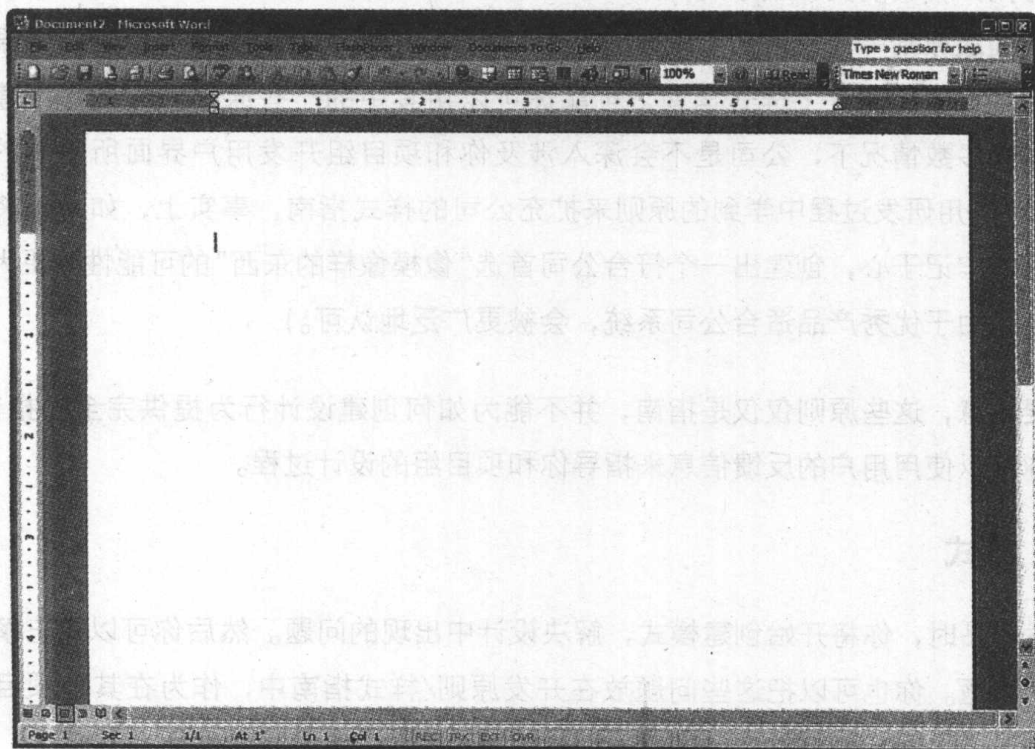


图7-4 独占型姿态实例

在大多数时候这种程序占了整个屏幕,这样独占型应用程序专门为全屏设计。它用那些空间来增加操纵目标的功能。Microsoft Word的窗口是一个很好的例子,有一个菜单栏,工具栏,在屏幕的右边有个任务窗格。这样可以给用户很多反馈和很多输入机制。

- **暂态型**——当用户需要完成一个特别的任务时,暂态型应用程序经常要用到,如用 Windows 字符表去查看一系列字体型的字符。暂态型程序用的不是很多,且没有占据很多的空间。

由于暂态型没有独占型用得更多,它们应该简化并贴近用户。在窗口中,你最好减少使用滚动条和其他复杂的交互功能,除非必要才加上。在窗口字符表中,如图7-5所示,用户看到了可见的字符介绍,所以她可以简单快速地看到想要的字符。程序里有个可以滚动字符列表的滚动栏。用户需要帮助时,她可以点击帮助窗口顶端的帮助按钮。

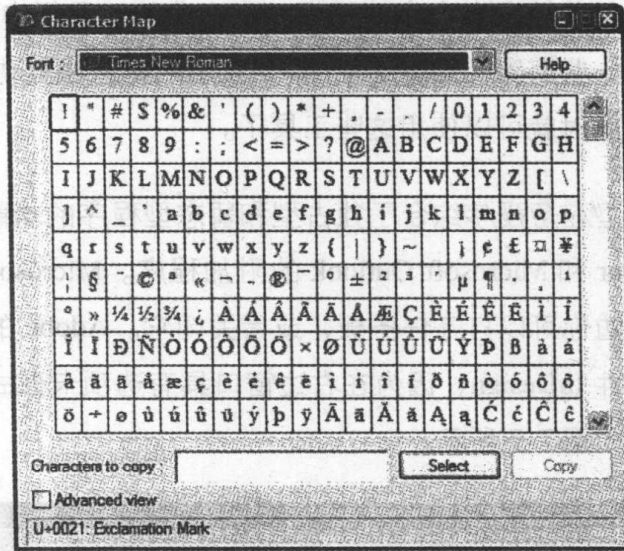


图 7-5 字符表窗口是一个暂态型姿态实例

- **精灵型**——精灵型应用程序通常不与用户相互作用。这些程序包含一些驱动程序，它可以支持打印机、监视器和鼠标。如果你有权使用这些程序，操作系统将提供“控制面板”来修改，就像 Microsoft Windows 中的控制面板一样。控制面板窗口提供一系列你可以改变的精灵程序，如图 7-6 所示。

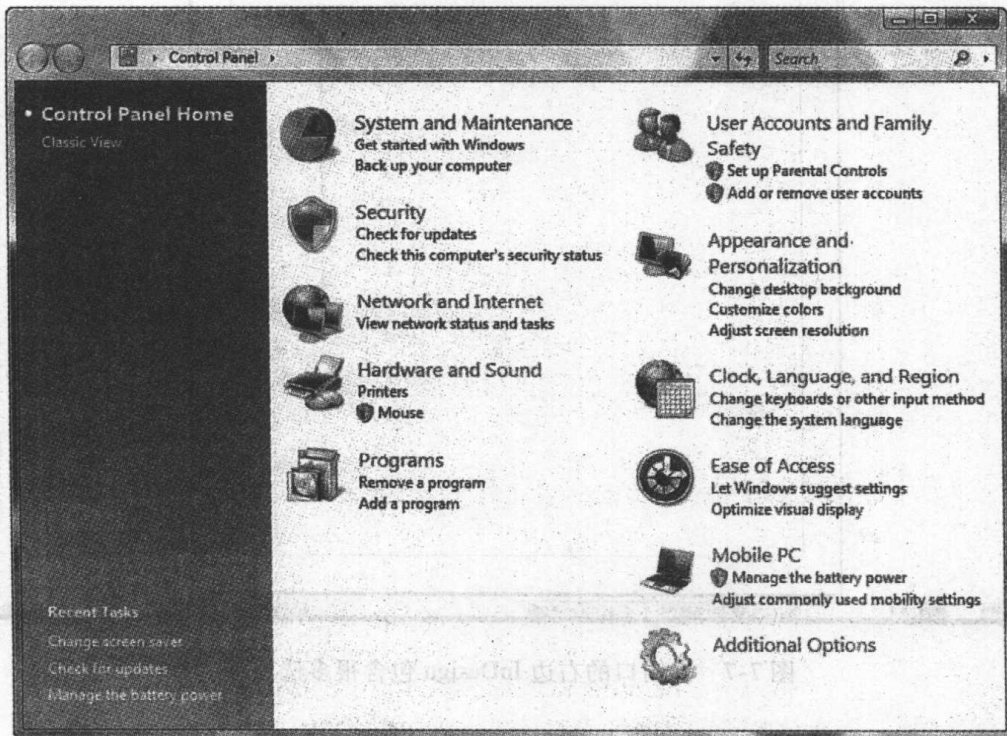


图 7-6 控制面板窗口，让你配置精灵程序

窗口中靠近系统时钟的地方有一小块图标，在右边的工具栏中列出了许多后台程序。当你安装程序的时候，一些有实际用途的程序(如我用的轨迹球驱动)的图标已放置在工具栏上，这样在你安装程序时就可以使用这些工具了。

- 辅助型——辅助型应用程序综合了独占型和暂态型程序的功能。微软发展了它的办公组件，Internet Explorer 和 Microsoft Outlook 的附加程序。Microsoft Word 也展示了它的附加功能。例如，在右边你可以打开任务板，或者关闭它。Adobe 的界面例如比 Microsoft 更好，它的图形设计软件 InDesign，因为在右边窗口具有选项卡形式的暂态程序，这样方便你点击打开或关闭，如图 7-7 所示。

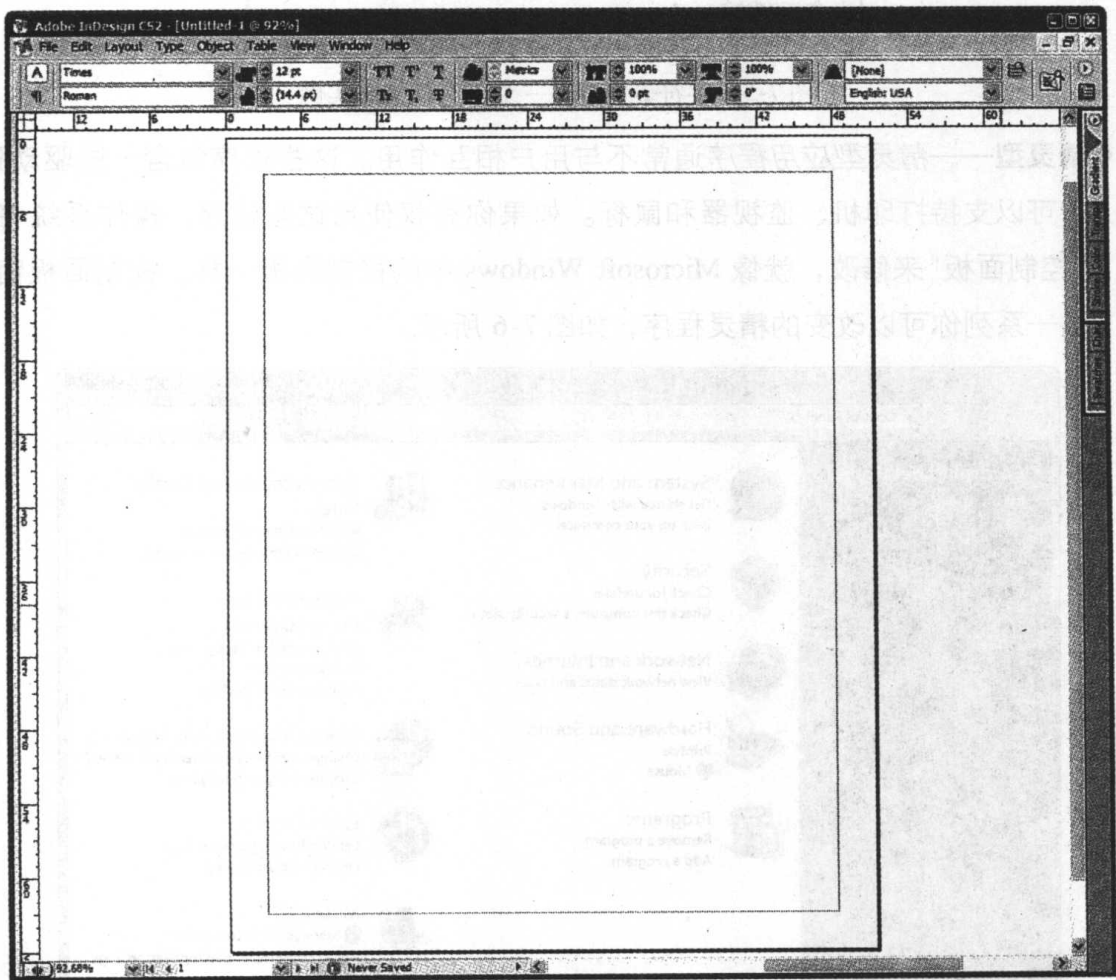


图 7-7 在窗口的右边 InDesign 包含很多选项卡

7.4 界面行为

桌面上的图形用户界面展现了 2 种最常见的访问和查看信息的方式：鼠标指针和窗口。这 2 个功能在图形用户界面中非常常见。你在任何计算机上能找到类似的功能，如膝上型电脑的触摸垫。在这节中，你将学习到一些功能的行为及其相关的问题——一些来自于 Cooper 和 Reimann，一些来自于我自己的经验。

7.4.1 使用鼠标指针

指针是图形用户界面的图形功能，通常是个箭头，你用鼠标直接操作，它的形状与一块肥皂非常相似。鼠标指针是控制设备，能控制不同的目标和机构。如果你发现用鼠标不舒服的话，你也可以用各种各样的操纵设备如轨迹球或触摸垫代替鼠标。例如我在计算机屏幕上用轨迹球来移动鼠标指针。许多笔记本都有鼠标垫。因为用户期望用户界面同其他所有程序所遵循的规则一致，所以你必须注意鼠标操作以及图形用户界面如何使用它。

1. 移动

鼠标在底部有一个球或激光指导系统，所以当你在桌上将鼠标移来移去的时候，鼠标指针也在同一方向移动。当你移动的时候，鼠标指针根据程序的要求会改变它的形状。例如在 Microsoft Word 中，当你将鼠标指针移过文档页的时候，它改变成 I 型条如图 7-8 所示。I 型条指示了你将新的文本插入到文件中的点。当你移动鼠标到了文档页外，它又返回成了箭头。当然一个程序正在运行一个时间比较长的操作时，鼠标指针可能变成忙碌光标，一般是沙漏，在 Windows Vista 系统中，它是一个圆形。

2. 点击和按钮

大多数计算机鼠标有两个按钮，一个在左边一个在右边。默认情况下，左键和右键的功能不同。如果用户或管理人设置了计算机的界面，某些按钮的方式可能与经典的默认设置不同。

3. 鼠标左键

当点击鼠标左键时，操作系统把指针放置在精确的位置（标记在文本中添加、改变或删除的移动指示器，通常是闪烁状态），初始化与按钮或链接相关的某项任务，或将操作

系统的焦点转移到刚点击的地方。例如，如果点击 Microsoft Word 工具栏中的保存按钮，系统将完成保存文档这一操作。

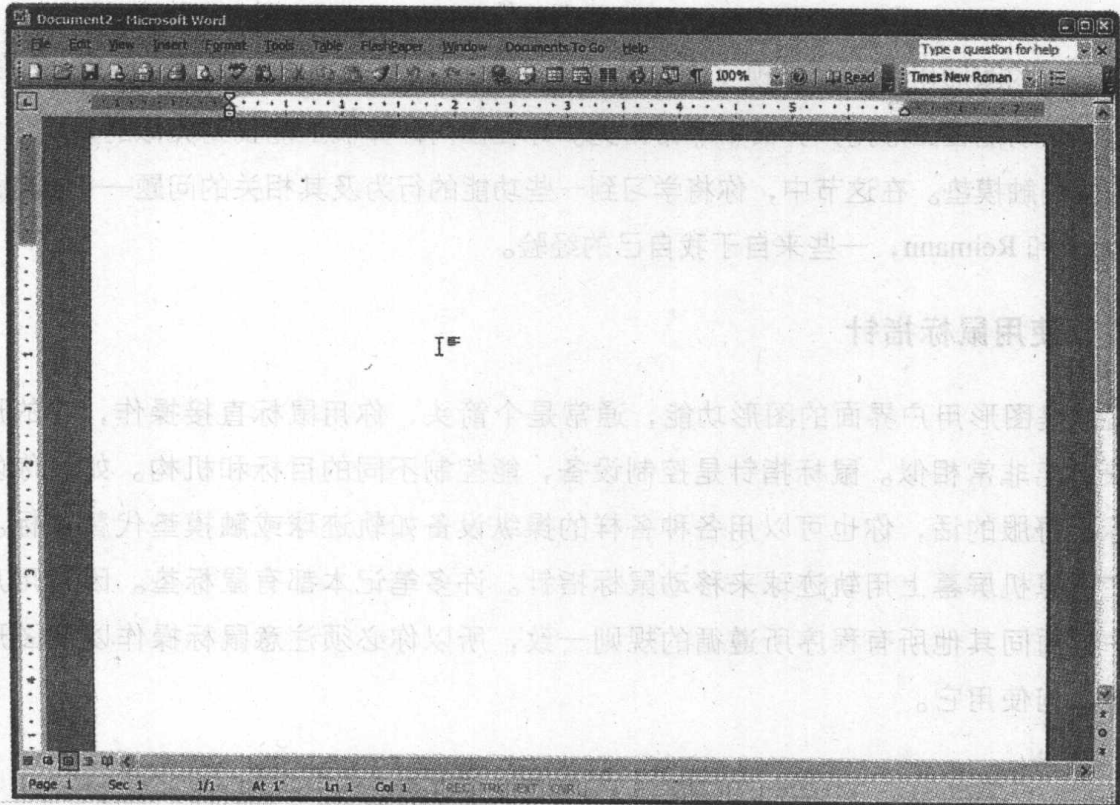


图 7-8 Word 中的 I 型条

某些链接要求你快速地点击鼠标左键 2 次，即双击以打开这个链接。例如，打开控制面板窗口时，必须双击功能列表中的某个图标才能打开附属窗口。这对于防止用户无意点击了某个任务来说是很有用的。

在图标或窗口标题栏上按下鼠标左键，将其移动到桌面上的其他位置，从而使桌面上的图标或打开的窗口移动到其他位置。这项功能叫做拖放操作，它也允许移动图标到另一个上面，例如将一个文件移动另一个文件夹。除此之外，还可以用拖放功能来点击并选择一组目标，如 Word 文档中的文本。

4. 鼠标右键

微软在 Windows 中引入了鼠标右键功能。当用鼠标右键点击图标、窗口或任务栏中的某个项目，称为右击，会出现弹出菜单以便完成特定功能。这些功能和最小最大化、关闭窗口或其

他功能一样简单。例如在 Word 文档中右击一下，将出现特定的菜单选项，如图 7-9 所示。

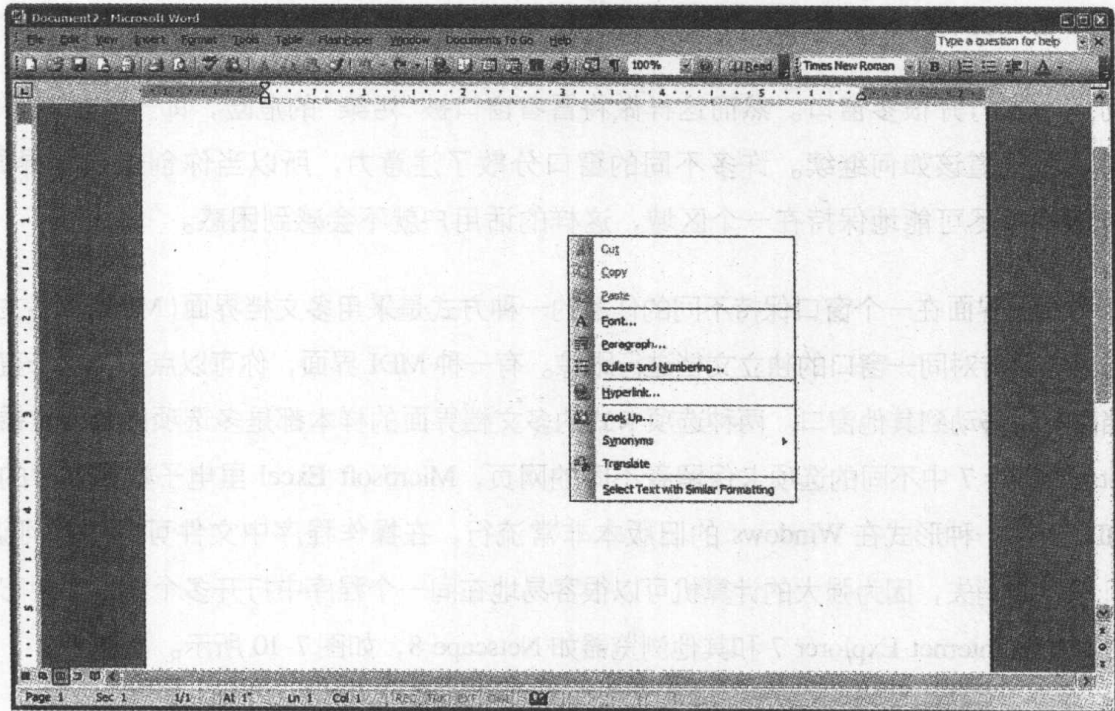


图 7-9 点击鼠标右键访问 Word 中弹出的菜单

鼠标右键在 GNOME 和 KDE 这 2 种 Linux GUI 中很流行，甚至 Mac OS X 也采用了右击功能。

然而，用户可能不知道怎么用右击功能除非他们已经习惯了在目标上用右击功能，或由于失误点了右键，或想知道怎样打开菜单或看看其中的选项。当你确认了用户的需要，你可能要从中精炼这些需要。判断用户是否喜欢用右击功能以及如何展示这些信息才能让用户了解这项功能。

7.4.2 窗口行为

不论使用的是哪个图形用户界面，在创建用户界面时你会发现窗口需要遵循相同的步骤。

1. 打开窗口

可以用 4 种方法打开窗口：

- 点击桌面上链接程序的图标。如果图标是一个文档，程序将自动打开这个文档。
- 双击窗口中的图标或链接。

- 点击程序中的菜单选项。例如点击 Microsoft Word 中文件菜单中的 New 选项。
- 点击任务栏中的按钮或图标。

可以一次打开很多窗口。然而这样做将冒着窗口被“污染”的危险，即一次性打开很多窗口你将不知道该如何继续。许多不同的窗口分散了注意力，所以当你创建用户界面时，把每样事物都尽可能地保持在一个区域，这样的话用户就不会感到困惑。

用户图形界面在一个窗口保持不同的任务的一种方式是采用多文档界面(MDI)。多文档界面用选项卡保持对同一窗口的独立文档进行跟踪。有一种 MDI 界面，你可以点击选项卡直接进入文档而不用移动到其他窗口。两种选项卡式的多文档界面的样本都是多选项网页浏览器，如 Internet Explorer 7 中不同的选项卡保留着不同的网页，Microsoft Excel 里电子数据表中的工作表。MDI 的另一种形式在 Windows 的旧版本非常流行，在操作程序中文件可移动和可调节。MDI 几乎已经消失，因为强大的计算机可以很容易地在同一个程序中打开多个窗口，但它正开始回归，就像 Internet Explorer 7 和其他浏览器如 Netscape 8，如图 7-10 所示。

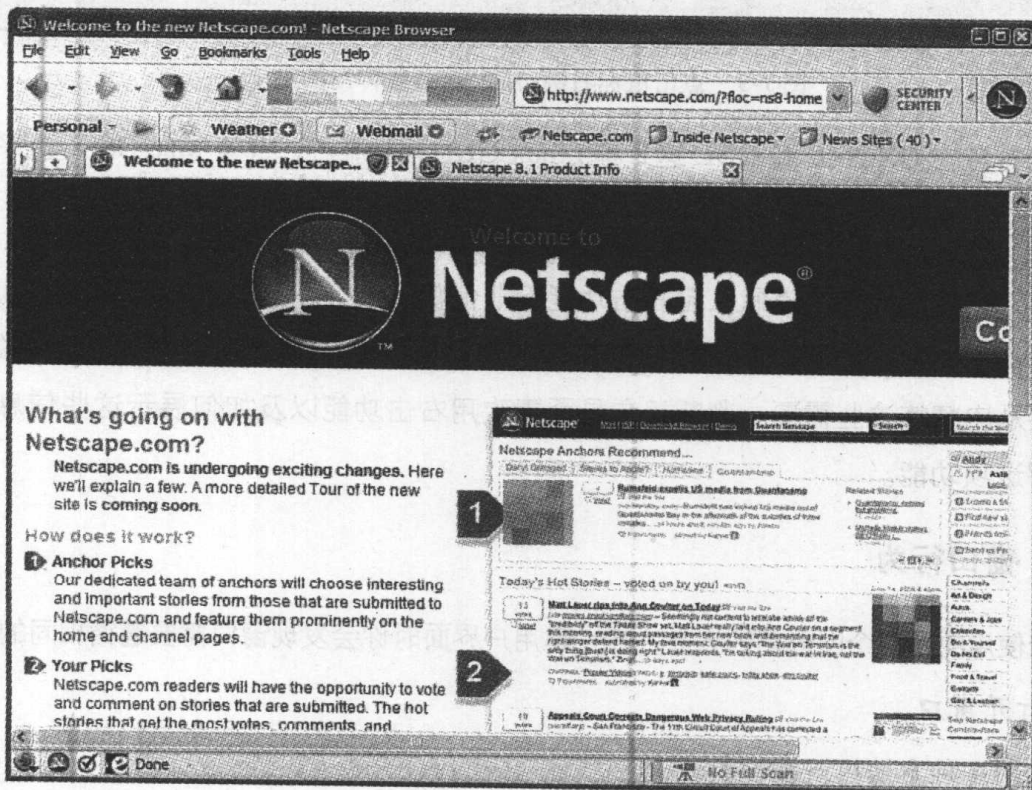


图 7-10 在 Netscape 网页浏览器中的选项面板

Netscape 的标识 N 字母为 Netscape 公司注册商标。经授权使用。

窗口也能打开一个对话框，对话框是个比较小的窗口用来给用户设置和做决定。如果想在 Microsoft Word 中打印文档，将出现对话框，以便确定打印什么、打印多少份等，如图 7-11 所示。

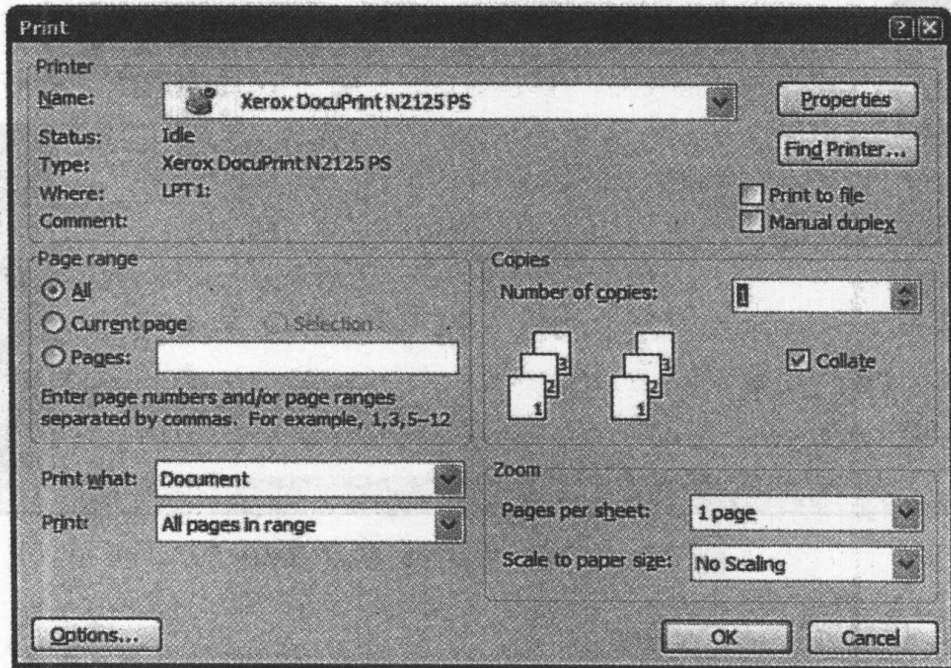


图 7-11 Word 中的打印对话框

当你要求用户打开一个窗口时，要确定用户需要这个窗口的信息去完成任任务。例如在打印文档前你需要指定打印机和完成打印设置。

如果你可以使功能显示于一个主窗口而不用打开其他的窗口，那么最好将功能集中于主窗口中。这将使用户工作起来更容易。例如，用 Microsoft Word 时，我在字体对话框中通过点击【字体】然后设置字体大小以及其他的字体相关的功能(如字体颜色)，这样可以改变格式菜单的字体。然而 Word 在主窗口中包含了工具栏，这样你可以改变文档窗口中的字体，如图 7-12 所示。

一条应用于创建窗口和对话框的重要规则是：确保在每个窗口中仅仅提供给用户需要的信息。太多的选择对于用户是场灾难，功能被淹没在菜单中而不为用户所知，这是因为他们不知道如何接触到这些功能。

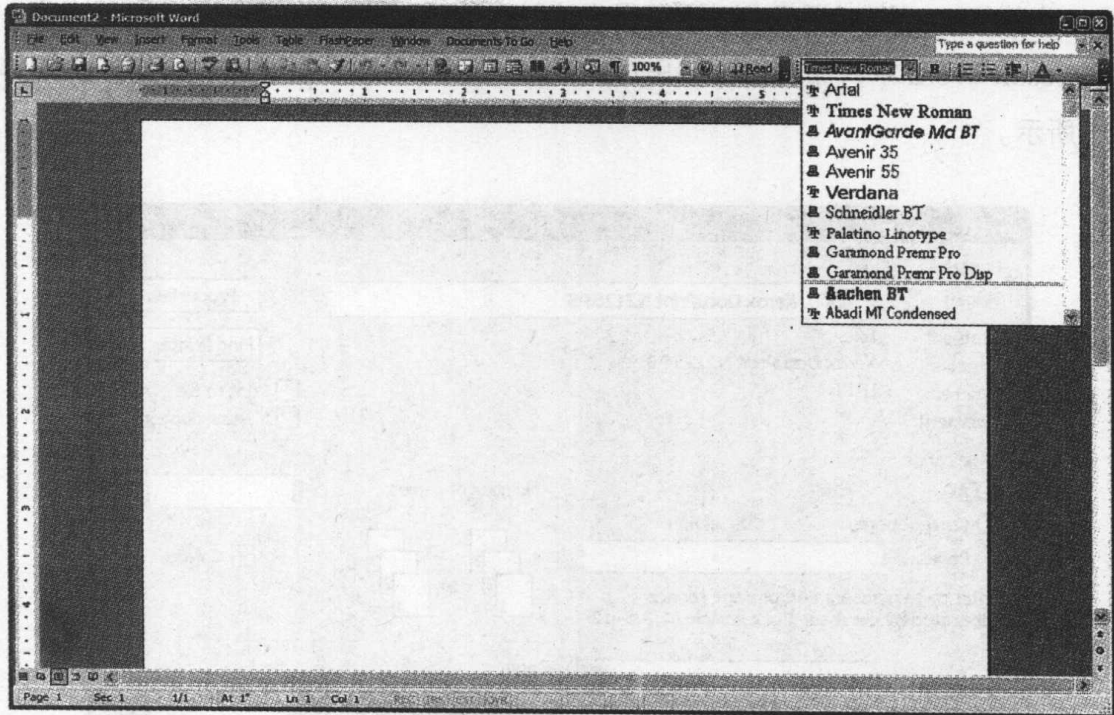


图 7-12 在工具栏中列出了 Word 的字体

2. 最大、最小化，改变窗口大小

可以将窗口最大化从而填满整个屏幕来集中所有注意力。例如当写一个文档时你应该最大化窗口，这样你就可以只关注这个文档而不用理其他的窗口。在标题栏中一般有最大化，最小化，以及关闭按钮。在窗口中这些按钮都在标题栏的右边，如图 7-13 所示。



图 7-13 标题栏中的最小最大化和关闭按钮

如果窗口是最小化的，标题栏中的最小化按钮将改变成恢复按钮，用来改变窗口的大小。你可以通过鼠标指针移动窗口的边缘来改变大小，按住鼠标左键进行拖放操作。如果顺着垂直于边缘的方向拖，将改变窗口的宽度。如果从垂直于水平方向拖，高度将改变。如果从角上拖，宽度和高度将同时改变。

然而对话框窗口的大小可以改变也可以不改变。如果觉得窗口需要特定的大小可以进行设置窗口操作，这样用户将不能改变它的大小。除此之外通过制定窗口模式(modal)，可以阻止对话框出现时用户切换至其他窗口。这项技术应谨慎使用，因为用户界面设计师常常为了给一个或多个程序之间多任务操作加上限制而滥用这一模式。你将在第 9 章中将了解到用户测试中用户对于界面喜好和厌恶的窗口行为。

3. 移动窗口

在屏幕上移动很多窗口可能会让它们彼此重叠。有一个比喻如此说，每个窗口都重复一页的话，图形用户界面不就是复印机吗。很多人同意我的这个想法，我不在意重复很多纸张因为第一张之后的所有纸张我都忽略了。

任务栏是一种解决重复窗口问题的方式。另一种办法是 Microsoft 引入的层叠窗口，它可以并排也可以覆盖。如果有 4 个并排层叠的窗口，则屏幕的四个角落各有一个窗口。层叠窗口的效果是否成功取决于屏幕的大小以及典型用户所使用的分辨率。如果在 800×600 像素分辨率的 15 英寸显示器上并排叠放几个窗口，单个窗口中的内容将很难看清。因此如果程序要求多个窗口正常运行，就决定了使用何种显示器和屏幕分辨率并进行相应的界面设计。你可以用独立窗口的设计或使用在这章前面所讨论的 MDI 技术。

4. 关闭窗口

通过点击标题栏中的关闭按钮来关闭窗口。但是你也可以创建键盘快捷方式以使用户更容易关闭窗口。例如 Windows 用 **Ctrl + W** 和 **Alt + F4** 分别来关闭文件和打开程序。可以参看图 7-14 所示的 Word 标题栏菜单实例。如果应用程序要求很多键盘操作，可以在用户界面中设置键盘快捷方式。这对于用户而言是件高兴事(尤其是残疾人)，他们不必从键盘到任务来回换手。

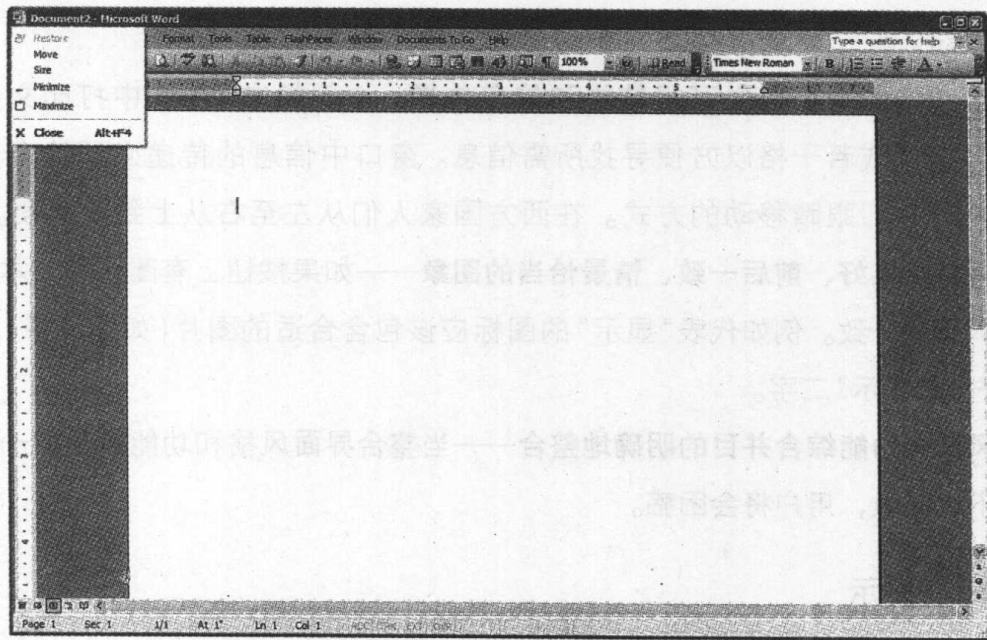


图 7-14 Word 标题栏菜单也展示了可选择的键盘快捷方式

7.5 帮助用户寻找信息

你可以给用户一些线索或暗示来帮助他们寻找信息，然而必须确保提示得恰到好处，不会过分繁杂而使用户迷惑。

7.5.1 视觉暗示

正如在“界面行为”那一部分中所讨论的，鼠标指针会根据其在窗口的位置而改变形状。当然，也有其他的视觉暗示，它们不仅包括按钮和标题栏上的名称也包括菜单选项、按钮栏和图标等。

既然设计界面时可以用到如此多的视觉暗示，那么坚持贯彻视觉界面的5条规则就很重要了(Cooper 和 Reimann, 2003)：

- **避免视觉噪声和混乱**——单个窗口中存在太多功能或太多分散注意力的鲜艳颜色是很容易犯的错误。这会导致用户混淆含义。而且，如果信息太多用户会感到挫折。
- **运用对比、相似性和分层来区分和组织元素**。例如按钮背景颜色加亮以使用户知道更应该注意这些按钮窗口。功能相关的按钮集中在一起，如保存和打开文件按钮。
- **为组织的每一层提供可视结构和信息传递**——例如在 Word 中打印文件，打印对话框分成若干格以方便寻找所需信息。窗口中信息的传递必须符合用户的逻辑路径，即眼睛移动的方式。在西方国家人们从左至右从上到下阅读。
- **运用结合良好、前后一致、情景恰当的图象**——如果按钮上有图标和文本，图标和文本必须一致。例如代表“显示”的图标应该包含合适的图片(如显示器)以及图片下方的“显示”二字。
- **将风格和功能综合并目的明确地整合**——当整合界面风格和功能时保持一致性。如果不这样做，用户将会困惑。

7.5.2 声音提示

现在大多数计算机都有内置声卡和音箱，故可以听到多媒体文件。当今的图形用

户界面也拥有内建音效，用于警示用户某些事件。比方说，如果你的 Windows 中装有 Outlook 或 Outlook Express 程序，可以设置 Outlook 在收到新的邮件时发出默认的 Windows 铃声。

使用听觉提示时需要谨慎，比如说用户每次点击按钮计算机都发出特定的噪声。如果经常这样，用户很快就会烦躁不安，甚至连耳背的用户都受不了。为了减少烦恼用户可能会调小音箱的音量，但如此一来又可能将其他重要的声音提示减弱了。如果确实打算用声音提示，那么一定要给用户提供一个关闭音频的选项。

7.5.3 弹出式信息

尽管视觉上很吸引人，图标也不一定能传达足够的信息。而且有些图标确实非常深奥或让人迷惑。图形用户界面弥补这种缺陷的一个法是使用弹出式信息(也叫作工具提示)。例如将鼠标指针移动到 Microsoft Word 中的工具栏图标上，指针下会出现了一个描述图标用途的小框，如图 7-15 所示。

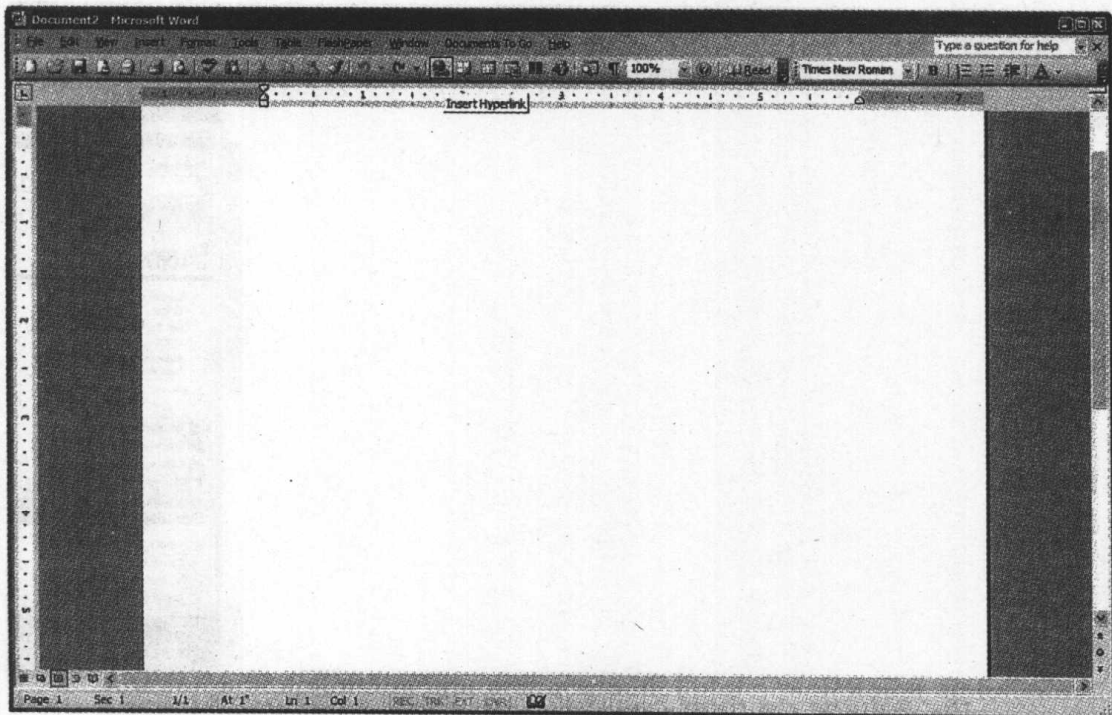


图 7-15 Word 中的弹出信息

创建弹出信息的关键是使信息尽可能的简化——最多几个词，尽可能的简洁。这也是一种挑战，但你仅仅只是向用户提示图标的用途，而不是解释它的全部流程。

7.5.4 搜索引擎

搜索引擎在当前的操作系统中比较常见，因为很多人经常在网页中使用它，对其使用比较熟悉。诸如 Microsoft Office 2003 一类的程序在右上方的窗口程序有一个搜索文本框，期待用户在框内输入问题。如图 7-16 所示。

苹果公司在 Mac OS 8.5 中引入了 Sherlock 搜索引擎，使用户可以在文件、文件夹和网页中查找信息。Google 向 Windows 用户提供了免费下载的程序(称为 Google 桌面)，提供了相似的功能。Windows Vista 系统现在支持“开始”菜单中的搜索功能。取决于用户如何操作软件，你可以依赖这些工具让用户进行信息查找，也可以像 Office 一样在窗口中增加搜索功能。和网站引擎一样，操作系统的搜索引擎并不特别完善，而且它们的功能不应取代在线帮助系统。更详细的说明参见下一节。

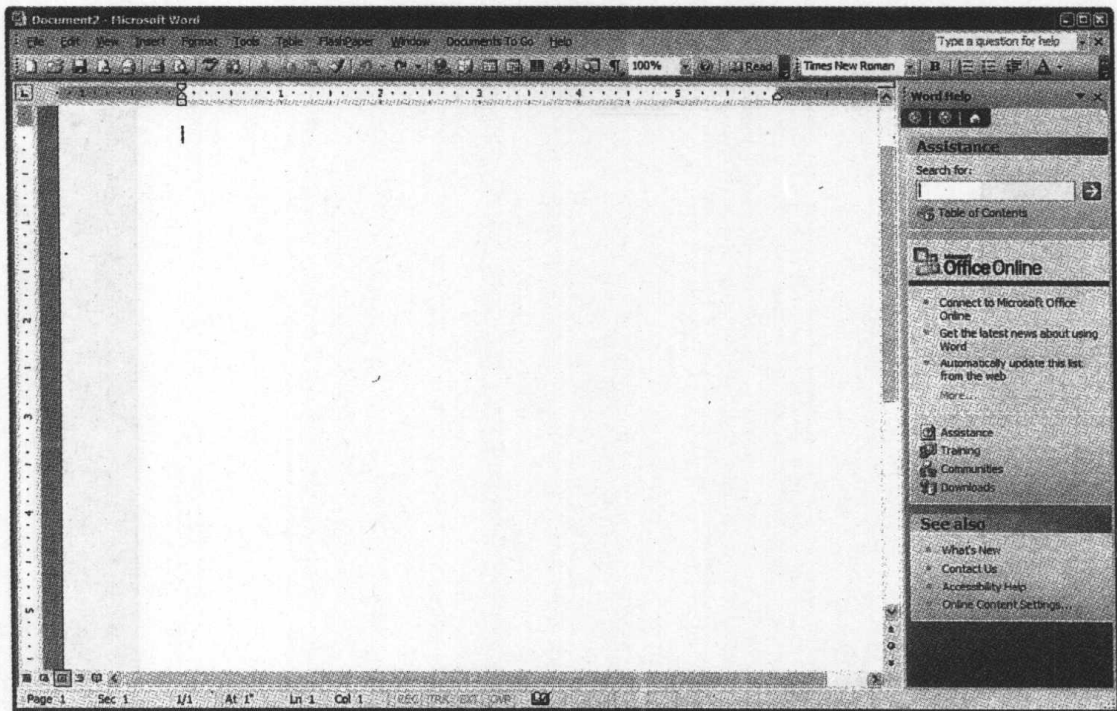


图 7-16 Microsoft Word 中的帮助框

7.6 与用户交流

第6章中探讨了许多用户都是永久的中级用户，也就是说他们对如何尽可能快地找到解决问题的答案很感兴趣，这可以帮助他们继续完成任务。如果你不能通过界面与用户进行有效的交流，你将失去他们。

7.6.1 使功能易于寻找

除了前面所介绍的帮助用户找到信息，在产品设计中用户界面需要遵循一致性和标准性。你或部门其他人将负责设计并实行这些标准。从公司预先设计的软硬件或网站用户界面的标准开始制订标准，并不断修改你觉得需要修改的地方。完成了用户界面设计的最终标准之后需要增强标准的一致性，这样用户可以在同一处地方找到信息。

然而你提前制定了标准这一事实并不意味着在开发过程中标准不可改变。标准会随着用户的想法而改变。这就是可用性测试比较重要的原因，我在之间的章节以及第9章中做了说明。

7.6.2 在线帮助

在线帮助是一种使用户更容易利用软件来获取信息的方式。你也可以把训练模式和相关信息都包括在在线帮助系统中。现在大多数 GUI 程序的系统中都包括了在线帮助，并且在多数程序中你可以通过菜单栏中的帮助菜单选项来获取在线帮助。仅仅在在线帮助系统设计合理的情况下，它才可以让信息更容易获取，并节省公司的打印成本。

用户通常在有问题时使用在线帮助来进行搜寻，例如打印时。搜索功能通常只是在线帮助窗口3个选项卡中的一个，另2个是表格内容和搜索索引。很多公司会简单地将用户指南转化为在线帮助系统，增加一些基本的搜索功能并对大众公开在线帮助系统。这其实是个糟糕的主意，因为用户并不像使用用户手册那样使用在线帮助。用户希望从在线帮助得到的答案能跟得上眼睛移动的速度。

有用的在线帮助系统不仅需要强健的搜索能力，还需要具备强大的索引。如果有使用在线帮助系统的计划，你需要从头开始开发帮助系统，并在各部分程序中测试它的有效性

和实用性。在线帮助系统将成为客服的第一步。如果没有开发好在线帮助，公司将花更多的时间和金钱来训练客户服务人员。

7.6.3 助手和向导

助手和向导能帮助用户更有效地使用程序。它能给出提示或者根据你提供的关键词来寻找答案。你也许知道 Clippy, Microsoft Office 的在线助手(见图 7-17)。很多用户在 Office 2000 和 Windows XP 中对它很厌烦,因此 Microsoft 公司在 Office2003 中把 Office 助手(包括 Clippy)的安装设为可选。

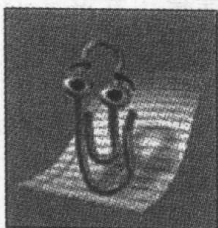


图 7-17 Clippy, Microsoft Office 助手

在图形用户界面中安装程序的时候,你可能会使用到向导。它是引导你逐步增加程序或功能的图形界面,例如创建家庭网络。向导以特定的步骤询问你一些问题,然后根据你提供的信息安装软件或配置功能。也可以在程序中使用助手精灵。例如 Microsoft Office 中提供邮件合并的向导以自动合并邮件。

助手或向导是否合适取决于所在环境。如果你进行过第 6 章中所介绍的用户和任务分析,你将了解用户打算完成的任务以及你是否需要附加助手或向导形式的帮助。

7.7 精炼形式与行为

定义了交互框架之后,就到了“目标导向设计过程”的最后一步:精炼。现在到了精炼界面的形式与行为的时候了,你需要遵循以下 3 步迭代过程(Cooper 和 Reimann, 2003):

1. **草拟外观风格**——设计团队致力于将你创建的纸上原型和故事板转换成全屏幕图像,方便你与团队中的其他成员分享。设计出程序的每个主视图和对话框以使团队可以看到界面是什么样子。你将反复经历这一步骤,确定任何存疑的问题,直到形成可用于设计其他视图、对象或对话框等的最终设计指南。

2. **建立实例验证**——你必须为程序建立实例验证，顺着关键路径的实例进行。因为你所建立的人物角色中每个人使用界面的方式不尽相同。

实例验证有三种喜好的方式(flavor)。一是关键路径，要沿着人物角色的决策树分成几个路径。例如在实例第三个步骤后，用户用按钮 B 代替按钮 A。当按下按钮 B 之后必须反映出将会发生什么。

第二种是必要的用户实例。它详细说明了所有必须完成但却很少完成的行为。例如展示用户如何配置软件。

第三种是边缘案例，说明了用户可选的很少完成的活动。玩计算机游戏时，有时候我打算不按规定的路线来尝试点别的，游戏有时候就会挂掉因为它没有对我角色做出的未预料行为做相应准备。用户使用你开发的程序时也有可能遇到和我玩游戏时的程序崩溃问题，但边缘案例的优先级比较低。产品最高优先级要处理的是日常使用及必要的情况，因为如果你的产品和界面不能处理这些情况，必然失败。

3. **最终设计**——现在你准备设计终稿。你可以通过你所了解的人物角色相关信息来创建形式和设计规格说明，包括屏幕实体模型和故事板，用于共享给公司内的其他项目关系人。你也可以通过编程创建交互原型或在网上创建在线演示来向项目关系人展示界面和程序是如何工作的。

7.8 案例研究：精炼纸上原型测试

现在需要进行精炼纸上原型测试了。在第 6 章中，你询问了 10 个参与者对于系统预设用途与限制的想法以及他们将如何克服那些限制。

如果你和 Evan 是对一项新的应用程序做纸上原型测试，那么对于系统内的视觉暗示和听觉提示及帮助功能需要留出更多的时间。对于现有系统，则会受到更多限制。除非你采访的用户明确提出这是一个可能需要进行修改的功能，否则你必须尽可能地遵守已存在的应用程序标准。

当前程序提供了一个弹出式信息的例子：在产品有效页中，当用户把鼠标放在产品名

上时，弹出消息框。弹出框给出了零件的主要信息资料，包括零件号和零件所属的自行车，如图 7-18 所示。

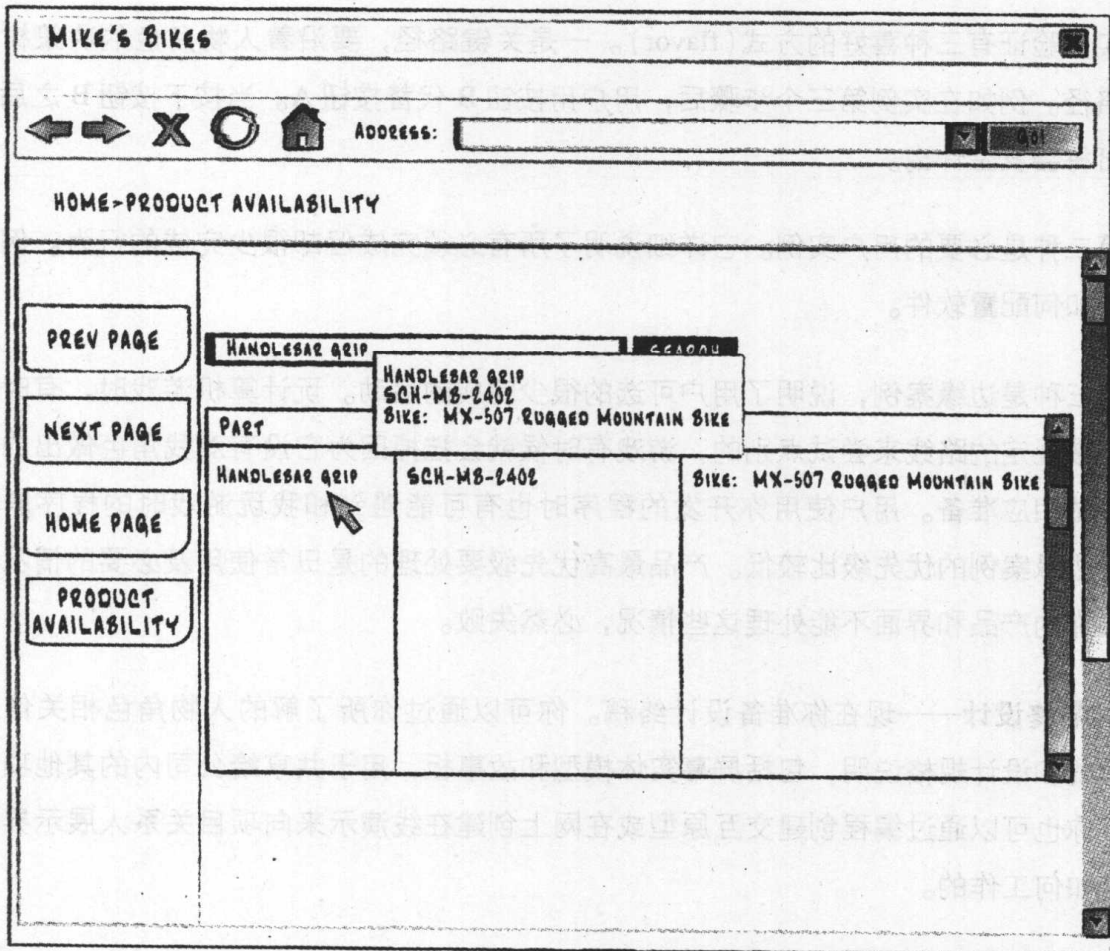


图 7-18 带零件名称、零件号和自行车的弹出信息框

因此，如果你想在系统中的其他地方增加弹出信息，例如用户将鼠标指针移动到产品表上的新标记栏可以查看剩余库存的产品数目，开发时需注意弹出的信息框必须和其他弹出信息的外观风格一样。对于纸上原型测试的弹出信息，其外观风格不一定要雷同，但测试的弹出信息中必须有附加文本。

如果有一个已存在的弹出式菜单，你可能想增加一些其他信息来扩展它的用途。例如，当鼠标移动到产品表上时显示每个产品表中的库存。这种附加功能符合用户“尽快地获取尽可能多的信息”这一目的。因此你可以在测试中出其不意地增加这种附加信息文本，然后观察他们在测试期间看到它们时的反应。

程序中还可以使用许多声音提示。Evan 需要特别注意这些提示，因为他在测试中扮演计算机的角色。在纸上原型测试中如果用户激活其中一个声音提示，Evan 需要在恰当的时候发这些提示声音。例如当用户操作错误时，Evan 需要给出系统错误的声音(或合理的提示)。

程序也包括在线帮助文本。当用户需要在线帮助来了解新功能(如查看产品有效页面)时，你需要让在线帮助窗口打开。不需要增加实际的在线帮助文本(仅仅当用户访问帮助系统时才展示在线帮助文本)。

总的来说，你要对应用程序非常熟悉才能正确地复制测试中的某些功能(如声音提示)。

程序是以网页为基础的，所以你和 Evan 也要注意界面的网络交互。在下一章将学习在纸上原型测试中应用网络交互。

7.9 小结

在第 6 章中你已经创建了人物角色，所以本章开始部分创建了一系列需求，然后设计了基于人物角色的交互式框架。这分别是“目标导向设计过程”的第 3 和第 4 个步骤。创建需求列表是一个包含 5 步骤的过程。你的需求列表需要留意对设计产生影响的现实世界需求，包括商业需求、客户和项目关系人的需求以及技术需求。你认识到建立关键路径情景需要以人物角色为基础的 6 个步骤，于是你可以了解到用户想在界面中看到什么。

接下来讨论的是交互设计。你了解了如何将设计原则及模式应用到优良设计中所必需具备的 4 大原则：合乎常理、目标明确、注重实效以及优雅。你还了解了概念层、交互层、界面层原则，它们界定了产品是什么样子、如何行为以及界面的外观风格。你还学会了分辨姿态、结构和行为模式，当项目出现问题时就可以应用它们。

然后我们学习了程序姿态，还有如何设计以符合用户的工作需求。例如独占型适合那些长期使用程序的用户。在用户界面中独占型包含了大量的工具以方便用户操作对象和控件。

此后继续讨论了界面行为。在主要的输入和输出设备情况下，独立地掌握和观察图形

用户界面上的物体：鼠标和窗口。这节讨论了用鼠标操纵对象的规则，以及快捷键这一按钮和鼠标的替代品。还了解到窗口操纵规则、移动窗口以及用键盘快捷方式来关闭窗口。

其后的内容是帮助用户找到信息。这一部分讨论了帮助用户使用界面的提示，包括视觉的、听觉的、弹出窗口和搜索引擎。你要认识到这些提示存在缺陷，例如声音提示对于听力困难的人来说不是很满意。

接下来讨论了与用户的交流。了解了使信息更容易查找的功能，包含前后一致的应用标准、满足用户更快找到信息这一需求且设计优秀的在线帮助系统、帮助用户更快更容易完成任务的助手和精灵程序。

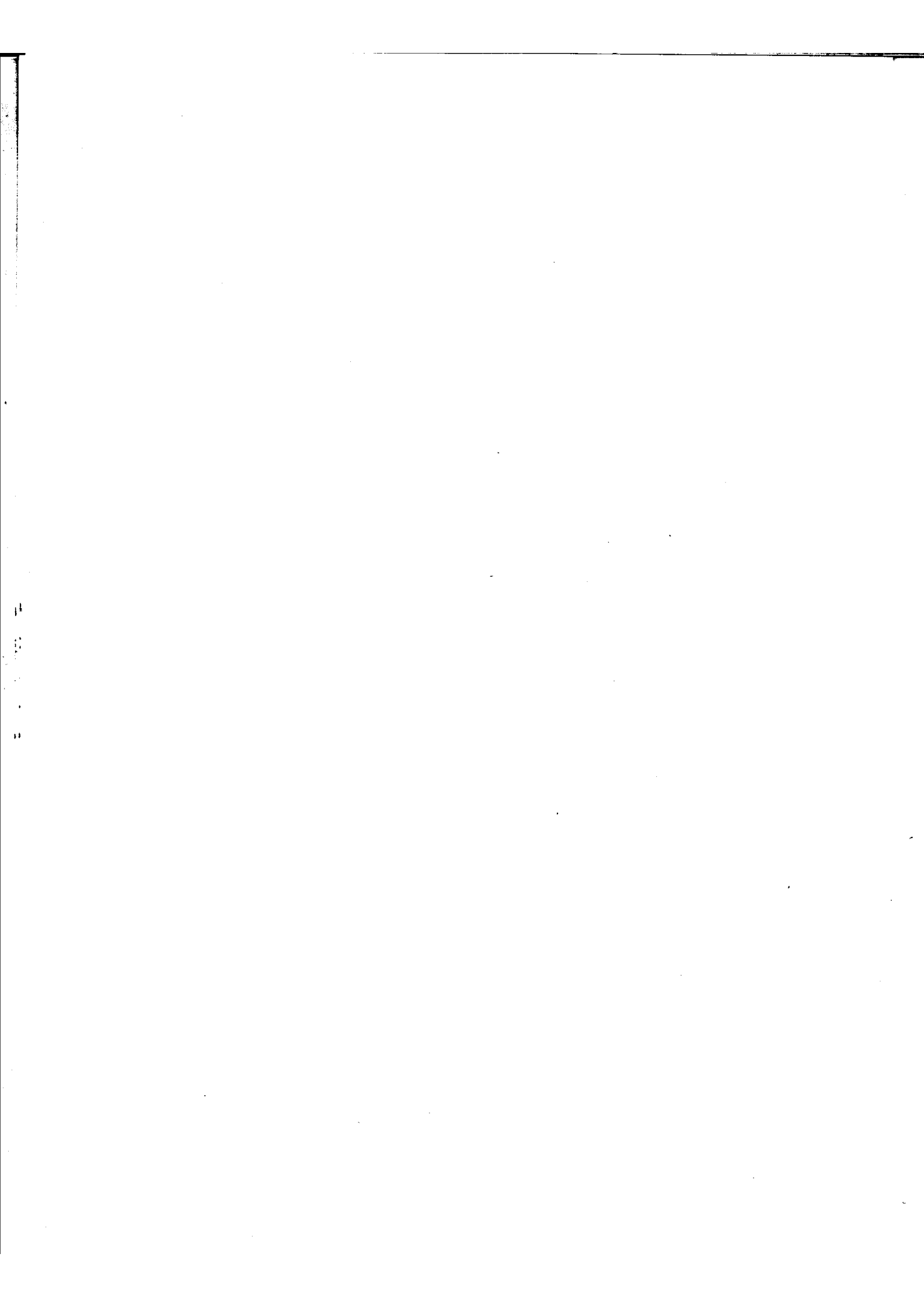
本章最后讨论的是“目标导向设计过程”的第5步和最后一步。精炼程序、界面的形式和行为，然后完成设计以便向项目关系人共享信息。你了解到必须反复多次来修缮界面的外观风格，然后建立实例验证以显示同一人物角色的不同用户是如何使用界面的。最后，你完成设计并共享给公司的项目关系人。

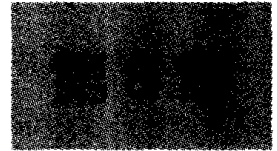
复习题

在继续第7章之前，先复习一下你在本章中所学习到的知识。回答以下几个问题，答案请参阅附录A。

1. 为什么需要对真实世界需求进行规划？
2. 为什么说纸上原型设计和故事板在设定关键路径实例时非常重要？
3. 能够引导你做出最便于用户使用的设计的3级设计原则是什么？
4. 为什么说创建模式非常重要？
5. 4种基于桌面的GUI姿态分别是什么？
6. 辅助应用程序的特点是什么？
7. 在目标上点击鼠标右键会发生什么？

8. 为什么必须避免视觉上的噪声和混乱?
9. 为什么说拥有一个设计优良的在线帮助系统很重要?
10. 图标上的弹出式菜单有什么优点?
11. 界面设计中一致性和标准的使用对于用户有什么好处?
12. 什么时候需要使用助手程序和向导程序?
13. 为什么说必须进行实例验证?
14. 如何向公司其他项目关系人展示最终设计?





设计网站界面

“一些人在几乎到达目标的时候放弃了他们的设计；而另一些人则相反，
在最后时刻通过付出比以往更多的努力来获得胜利。”

——Herodotus

本章内容提要

- 网站与图形用户界面：相似与差异
- 网站迷思
- 网站姿态
- 为什么需要网站工程
- 网站标准
- 四个规则
- 何时打破规则

开发基于网站的应用程序已经成为开发操作系统应用程序之外的更流行的选择，特别是你想为不同的操作系统开发应用程序。

然而，你必须明白网站图形用户界面与桌面图形用户界面应用程序之间既有相似之处又有许多差异。2种界面都遵循一些相同的规则，然而网页必须符合由超文本标记语言(HTML)以及其他相关网络语言确立的标准。

技术的发展开始让桌面图形用户界面与网站界面之间的界限越来越模糊，越拉越多的公司都在开发内嵌互联网功能的应用程序，允许用户使用桌面应用程序与网站进行交互。

如果你准备开发网站，有大量应该被消除的网站迷思存在。你需要知道这些迷思，以便于你了解在进行网站设计时有哪些情景支持和限制。像图形用户界面一样，不同类型的网站的也有不同的姿态。

网站工程是网站设计的一个有机组成部分。没有界面背后的应用程序和数据库的开发，你不能创建出具有电子商务网页表格或者用户数据库的网站。

你也应该理解网站标准和网站设计的四个规则。这些规则不是所有时候都适用的，本章也会讨论到什么情况下你可以突破这些规则。

8.1 网站与图形用户界面之间的比较

当你计划设计网站时，你需要了解桌面图形用户界面和网站界面之间的差异。它们有着各自不同类型的限制。

8.1.1 图形用户界面规则

图形用户界面包含了一系列用户与计算机进行交互的规则，这些规则包括：

- 桌面图标代表对应文件和应用程序。
- 鼠标(或其他诸如轨迹球等替代设备)和键盘用于操作对象。
- 能够让用户查看和操作数据的各种窗口。
- 窗口的排列和存储方式可以让用户同时运行多个程序。
- 严格的强制标准保证所有窗口的感观都基本相同。这样能够让用户不需要学习就可以使用新的应用程序，因为他们的界面是类似的。

8.1.2 网站界面规则

网站界面也有自己的规则，其中一些可能会与图形用户界面的有所重复：

- 网站使用叫做浏览器的带图形用户界面的特定应用程序来访问。因此，当用户访问网站时，用户需要打开一个窗口，使用鼠标来操作网站上的对象。
- 决定网页上的对象放置规则的是 HTML 和其他网站语言，需按其要求来制作网页。尽管新的互联网技术(比如 Adobe 的 Flash 动画软件)的应用越来越模糊了桌面与浏览器之间的界限，但大部分人还是习惯使用浏览器。因此，浏览器仍然限制了网站设计。
- 网页的外观受到网站安全颜色和字体设定的限制，以保证所有的浏览器上都能够正确地显示。使用网站安全颜色和字体，相同的网页在所有计算机平台上的所有可用浏览器中会表现得基本一样，如图 8-1 所示。

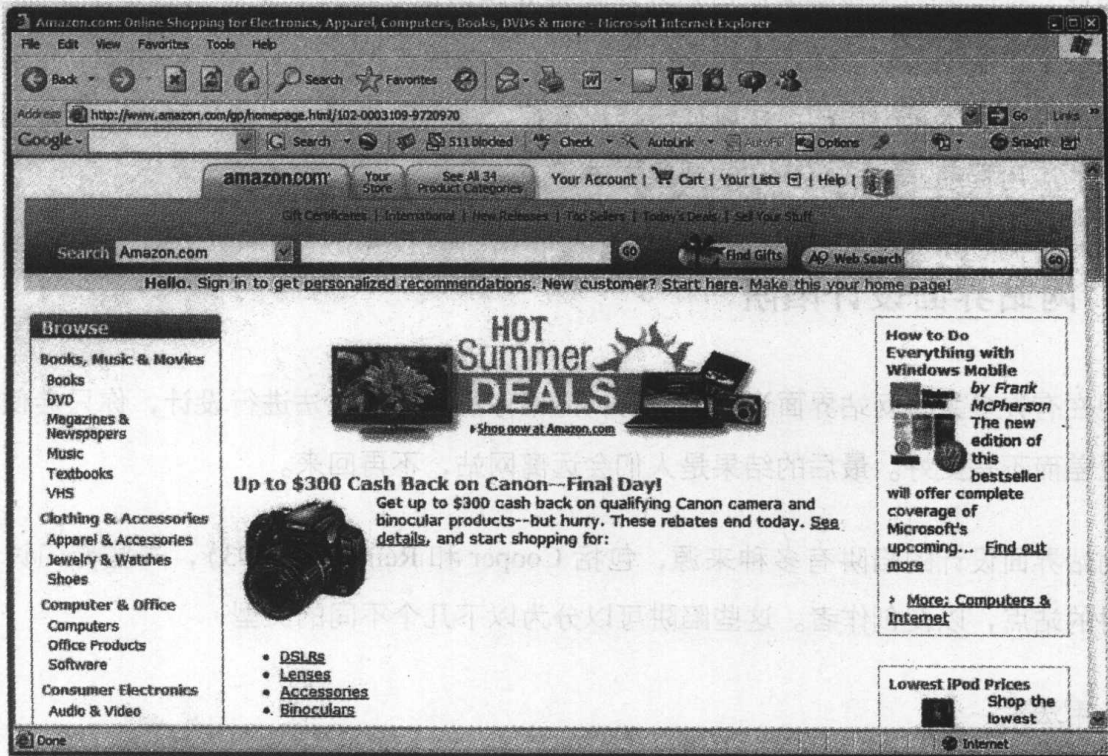


图 8-1 使用网站安全颜色和字体的亚马逊网站
(图片版权归亚马逊公司所有)

尽管有一些普遍接受的可用性惯例需要遵循，但网页的外观风格不必局限于严格的标准。我会在后面的章节中详细讨论这些标准。

8.1.3 基于互联网的应用程序

图形用户界面桌面和浏览器之间的区别越来越模糊，用户甚至将桌面计算经验与网络经验混同起来。人们越来越多地使用一些诸如 Java 编程语言、ActiveX 和 PHP 脚本语言、Adobe 的 Flash 动画软件、AJAX 和动态 HTML、更专有的像微软的 .NET 语言以及其他相关软件来创建更丰富、更具交互性的网络应用程序。

这些技术使得创建一个从桌面启动、对用户透明的、以网络为中心的应用程序要比以往更加容易。也就是说，用户不再需要知道应用程序何时通过网络与一个或多个数据库等在线资源交换信息。用户能够使用桌面图形用户界面来管理在线资源而不必使用浏览器，而且以互联网为中心的应用程序会拥有比网站界面更丰富的界面元素和反馈方法。

因此，你会发现无论是图形用户界面还是网站情境支持和约束在这些情况下都适用。要知道在设计基于互联网的应用程序过程中，一些情境支持和约束与你在创建应用过程中所使用的具体技术是相关的。跟以往一样，咨询人物角色并确认你和项目团队设计的基于互联网的应用程序是否按照用户的愿望在进行。

8.2 网站界面设计陷阱

现在有许多关于网站界面设计的陷阱，如果你按照那些说法进行设计，你只会使用户体验更差而不是更好。最后的结果是人们会远离网站，不再回来。

网站界面设计的陷阱有多种来源，包括 Cooper 和 Reimann (2003)，一些专门讨论这些陷阱的站点，以及创作者。这些陷阱可以分为以下几个不同的类型：

- 用法
- 设计
- 可及性

在你设计网站之前，要了解这些陷阱，这样你才能在规划阶段就能有效避免这些问题。

8.2.1 用法

网站用法陷阱是最近几年才出现的，存在大量的有关如何使用互联网的错误观念而带来的问题。在你开始设计网站之前，一定要理解人们是如何使用网站的陷阱。（陷阱是大胆的，真相隐藏在陷阱背后。）

- **网站界面是方便使用的**——如果网站全部都是只有一些文字和少量链接的静态页面，这个说法也许是对的。但是，近些年来，网站的交互性越来越强，包括表单、框架、脚本以及其他内嵌的应用程序使得网站界面的设计跟桌面应用程序的设计一样复杂了。网站界面还不能像桌面应用程序那样提供所有的丰富的反馈和流程，所以在设计网站界面时你必须知道一些设计约束。
- **网站和互联网是一个东西**——网站仅仅是允许人们通过互联网分享信息的图形化界面。互联网是传输包括多媒体文件、电子邮件消息和新闻组消息等在内的海量信息的互联互通网络。
- **网站设计都是针对浏览器的**——新技术正将网络技术扩展至浏览器以外，延伸到了桌面以及其他设备。举例而言，Adobe 的丰富互联网应用程序使用 Flash 来创建从外观到功能都更像是一个桌面的网站，这点我前面也提到过。Windows Vista 已经内建了能够链接网络的网站搜索功能。我的 Treo 智能手机也能够链接到网络，并通过应用程序在网站和 Treo 之间交换数据。
- **人们是耐心的，他们会留着网站上浏览网页**——有些用户上网进行研究，但许多访问者上网站是因为他们想获得网站上提供的特定信息。他们并不想花时间通读冗长的 HTML 页面或者经过大量点击链接才能找到他们想要的信息。

8.2.2 设计

我的一些商业客户相信下面这些网络设计陷阱。我的公司曾为一些客户设计了几个网站，这些客户觉得他们的网站受众需要下面列出的一些特性，结果证明他们的用户并不真的在意这些特性。（有时我不得不报告客户反馈，有时我会坚持自己的意见，以至于丢掉了不少客户）

- **网站设计与其他媒体设计是有区别的**——对，也不对。说网站设计是不同的的确没

有错。你有不同的信息流。正如在前面章节中提到的，网站设计与图形用户界面设计类似有些共同的约束，但也有很多不同的地方。但是，设计网站仍然意味着你要好好准备。你还是需要与用户进行访谈，了解他们想要的行为来帮助你设计出界面友好的网站。

- **动画和大量的图片是必需的**——网站给你了添加图片和动画的机会，但是你必须尊敬用户以保证他们不会因为网站上的大量图片而感到不安。与报纸或者其他纸质媒介一样，你需要很好地平衡图片与文字之间的比例。特别需要指出的是动画图形，它们可以吸引用户的注意力，但当用户在查看网站上的其他信息时这些动画可能就会让用户分心了。过多的图片或动画会让用户迷惑，阻碍他们在网站上看到想看的東西。结果只会使用户感到苦恼，甚至不再返回网站。而且，过度使用最新的图形技术可能会让使用某些不支持这些新技术的浏览器的用户无法正常使用网站。
- **网站上应该有背景音乐**——你可以在网站中嵌入声音文件，用户在访问网站时背景音乐会自动播放。但是，这很快也会使人烦恼，特别是当用户无法关闭或者无法将音量调到最小或者静音的时候。反复聆听同一段旋律会很快让人感觉枯燥。而且，如果应用程序是要依靠声音运行的，那么你就可能会失去一些有听力障碍的用户。
- **内容在网站设计中是最关键的**——当你设计网站时，你不仅仅在设计那些供用户阅读的静态内容的网页。你在设计一个系统，包括填写信息的表格，系统的行为，网站上的页面之间是如何流转的，以及网页内容本身。如果信息很难获得，那么用户使用应用程序时就会感到挫折和不满。
- **展示是唯一要紧的事，不用关心网站后台的“管道设备”**——当你创建一个网站，你显然想的是网站要好看，所有的链接都正常工作，所有的访问者都能访问网站。然而，当你开始添加表格、脚本或其他网络应用程序代码将网站变成一个网络应用程序后，你需要在网站后台使用诸如 PHP 和 Java 等语言来创建代码。如果网站的基础架构搭建得不够坚固，你就会冒着可用性问题的风险，更糟的是用户的数据或隐私的泄漏风险。
- **网站后台开发是你自己就可以做的事**——为网站制作可见的网页是比较容易的，因为其规则是简单明了的，如果网站仅仅是信息类网站，一般你所需要的就只是设计网页。但是，如果你在设计一个网站应用程序，除非你有丰富的网络编程经验，你最好雇佣有资质的公司来帮助你完成这个工作。虽然市面上有一些诸如电子商务等软件包可以成套购买，但你还是可能在将其集成到网站的过程中遇到各种问题，也

需要其他人或者公司的帮助。网站开发绝不是件简单的事，你需要将其委托给具有丰富网站技术经验的专业人士。

8.2.3 可及性

网站可能有成千上万的潜在访问用户，这些访问者的计算机配置也是千差万别的。因此，你要知道网站可及性陷阱是什么，以及实际情况是怎样的。

- **我要做的就是为最流行的和最新的浏览器设计网站，这样所有人都能够看到我的网站**——并不是所有的人都使用最新版本的微软 IE 浏览器，IE 浏览器在本书写作时是最流行的浏览器，其市场占有率达 90%。你必须考虑到人们在各种各样的平台上使用着各种各样的浏览器。比如，用户可能在 Windows 下使用 Firefox 浏览器，在 Linux 下使用 Netscape 浏览器，在 Mac OS 下使用 Safari 浏览器。
- **可及性好的网页必须面向最老的主流网页浏览器设计**——设计网页的 HTML 语言，就其可及性特性来说具有向后兼容性。当前的 HTML 版本是 4.0，几乎所有的主流浏览器都支持这个版本，而且目前的主流网页设计软件，如 Dreamweaver 和 FrontPages 等也都支持 HTML 4.0 开发。HTML 4.0 版本可以向后兼容到较早的 HTML 2.0 版本，所以你不必为使用了较早版本 HTML 开发的网站或者应用程序担心。需要指出的是，其他的一些网页技术并不都像 HTML 这样具有良好的兼容性，比如，JavaScript 在不同的网页浏览器中的效果就会有所不同。
- **可及性好的网页只能有文字**——你可以在网站上使用不同的颜色、图片和诸如多媒体片段等其他材料。但是，你需要为每幅图片和多媒体对象绑定替代标签或者 ALT 标签，这个我们在第 2 章“概念与问题”中曾经有过详细的讨论。用户应该有可替代的方法来使用网站，而不必依赖于复杂的视觉和听觉效果。
- **可及性好的网页不必为不能看见图片的用户操心图片问题**——网站是一个多媒体环境。有视觉障碍的用户仍然可以通过操作系统提供的可及性特性来感受网站。如果你知道会有视觉障碍用户访问网站，你可能想到提供声音链接，这样一来这些用户就可以通过听觉感受网站。

还需指出的是，大概 25% ~ 30% 的用户由于各种原因在浏览网站时不加载图像文件，常见的原因包括视觉障碍、带宽问题、害怕病毒以及对目前流行的大量图片商业广

告的反感。尽管诸如 DSL 和有线等宽带链接越来越普及，但仍然有不少用户使用电话拨号方式链接互联网，对于这些用户而言，他们也不会浪费时间来等待网页上图片的加载。为图片或多媒体对象添加替代标签，能够让用户知道你想通过图片或多媒体对象传达什么意思。

8.3 网站姿态

在第7章中，你学到了关于桌面图形用户界面应用程序的姿态问题。我们先回顾一下，桌面图形用户界面应用程序有4种类型姿态(Cooper 和 Reimann, 2003)：

- 独占型——长时间保持用户注意力的应用程序。
- 暂态型——用户偶尔用到的面向特定任务和基于需要的应用程序。
- 精灵型——在后台运行的不需要与用户进行交互的应用程序。
- 辅助型——同时具备独占和暂时2种特征的应用程序。

这些姿态对网站也同样适用么？没错，但是因为网站有着不同的功能性，它们有着不同的名称。而且，不同类型的网站需要不同的姿态。

不同的网站类型

你可以创建3种不同类型的网站，按照图形用户界面应用程序姿态的分法，这些网站分别称为(Cooper 和 Reimann, 2003)：

- **信息类网站**——这类网站不需要复杂的处理特性，就像广告中说的那样：只需用户通过点击链接就能在站点内的网页上搜索信息，链接到的网页上包含更多的信息。MSNBC 新闻网站就是这样一个信息类网站，如图 8-2 所示。
- **应用类网站**——这类站点需要通过脚本对网站后台数据进行比较复杂的操作和处理。用户是看不到后台数据的处理过程的。后台处理过程简单的如用户在表单中填写联系信息并将其通过电子邮件发送，复杂的可以是一个完整的电子商务系统，数据被保存在后台数据库中，网站上的数据信息也要动态更新。亚马逊电子商务网站就是这类网站的代表，如图 8-3 所示。

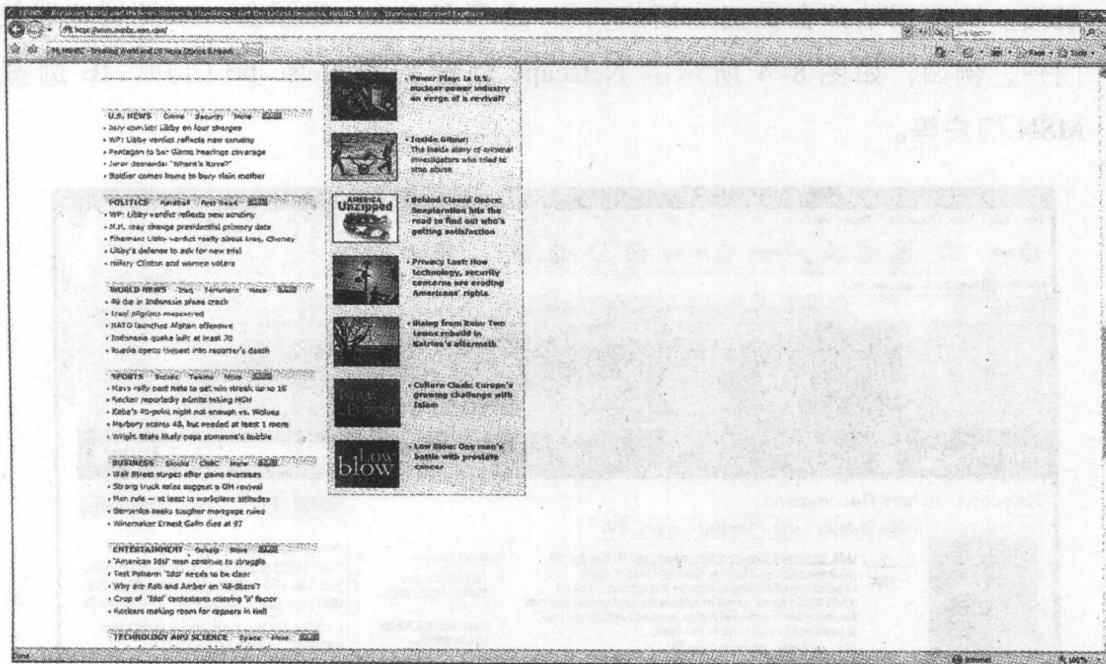


图 8-2 MSNBC 新闻网是一个信息类网站

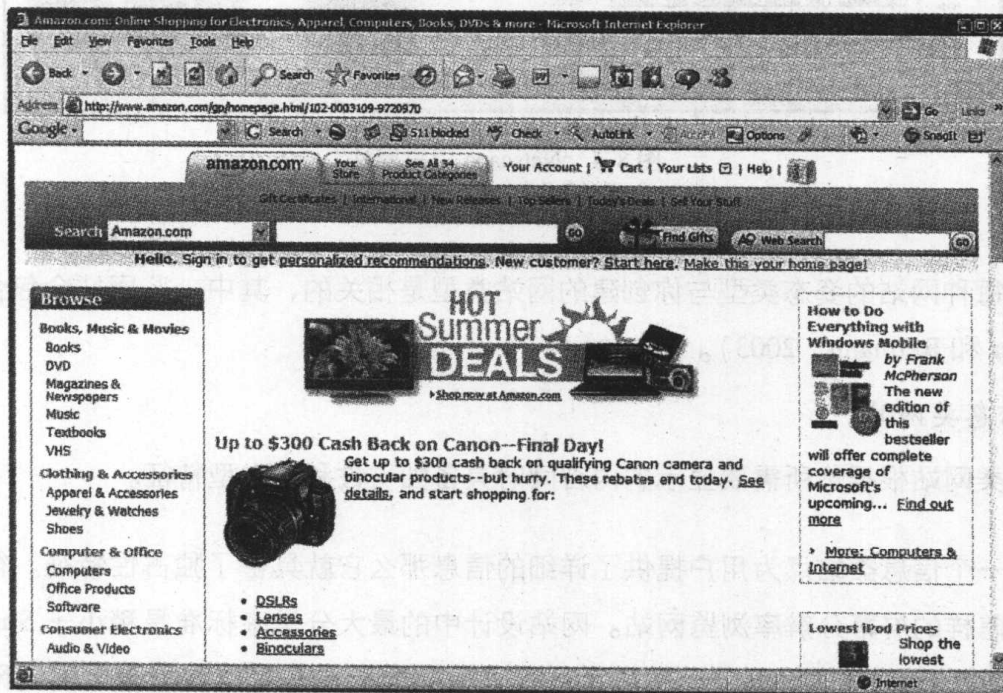


图 8-3 亚马逊电子商务网是一个应用类网站

(图片版权归亚马逊公司所有)

• 门户类网站——这类网站为用户提供公司的相关信息以及能够链接到其他站点的

链接。这些网站都会链接到诸如 AOL 或者 Yahoo! 等服务, 以及浏览器使用的门户, 例如, 如图 8-4 所示的 Netscape 浏览器的 Netscape 门户, IE 浏览器的 MSN 门户等。

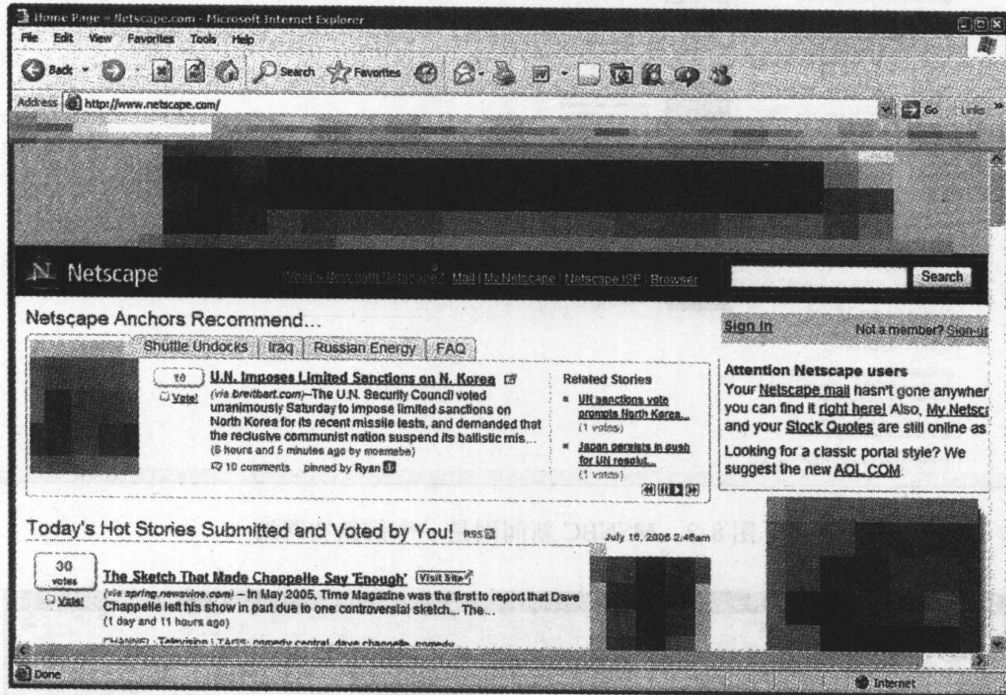


图 8-4 Netscape 门户网站

(图片已经授权使用)

具体每种网站的姿态类型与你创建的网站类型是相关的, 其中一些网站会包含多种姿态 (Cooper 和 Reimann, 2003)。

1. 信息类网站

信息类网站根据你所需要显示的内容同时具备独占性和暂态型特征。

如果一个信息类站点为用户提供了详细的信息那么它就具备了独占性特征。很难知道用户使用怎样的屏幕分辨率浏览网站。网站设计中的最大分辨率标准是稍小于 Super VGA 标准, 即 800×600 像素分辨率。这样可以保证所有的用户都可以在屏幕上看到网页上的文字, 但同时也意味着用户需要经常滚动页面来阅读所有的信息。

如果屏幕分辨率是比较受关注的, 那么你就需要在访谈阶段询问用户关于屏幕分辨率的要求。这样你就可以决定主要人物角色的分辨率需求。相反, 如果你设计的网站需要更

大的分辨率才能正确显示，用户就会抱怨，你也不得不花费时间和金钱来进行修改，甚至会失去部分现在或预期的客户。

如果主要人物角色并不是经常访问网站，那么网站就具备了暂态型姿态。典型的例子是微软的 Windows Update 网站，用户只有在需要对操作系统进行关键更新或者其他安装的情况下才会登录这个网站。

良好的网站导航永远是一个准则，对具有暂态型姿态的网站更是如此，用户是不会忍受需要多次点击才能找到想要的页面。许多网站使用 cookies 技术，cookies 是一些网站在电脑上保存的小的文本文件，里面记录了用户在网站上的偏好，下次用户再次登录同一网站时就会自动加载这些偏好。举例来说，Google 新闻网允许你自己配置自己的网站首页。Google 会将配置信息通过 cookies 保存在电脑内，下次登录时就会显示出你配置好的首页效果。

2. 应用类网站

应用类网站也会有不同的网站姿态，因为处理过程是比较复杂的，所以网站姿态会与网站的主要人物角色相关。

举例来说，如果某个应用只是面向客户的，诸如客户偶尔会上的电子商务网站，那么这个网站就同时具备独占性和暂态型的特征。电子商务网站具有独占性姿态，因为网站上提供了大量可供选择的各种各样的产品，人们可以对比各种产品并了解每种产品的功用。应用类网站也有独占性姿态特征，如果用户经常访问这个网站，就如同某业主老板需要每天来查看一下他网站上存货清单。

但是，应用类网站也会表现出暂态型网站姿态特征，因为用户一般不会每天来访问应用类网站。一个电子商务网站的暂态型姿态特征包括一个一直可见的购物车（有时还包括了选购商品的列表），良好的浏览导航助手，以及 cookies 用以告诉系统用户上次登录时查看的商品或者推荐类似产品给用户。

Cooper 和 Reimann (2003) 指出暂态型姿态网站应用程序应该与暂态型姿态图形用户界面应用程序遵循相同的准则。同时，他们给出了另外一些你应该知道的准则：

- 清晰的暴露应用类网站的功能性。尽量让网站功能明显且可见，比如可以通过点击

网页上的某个链接找到特定类型的服装。你也可以根据需要给出提示，比如跟照片绑定的文字提示说明或替代标签等。举例来说，如果网页上有一张毛线衫的照片，当用户将鼠标移动到照片上时就会弹出一个文字提示标签，提示用户通过该链接可以查看到更多的毛线衫。

- 让应用类网站简单、直接、明了。要记得用户都是带着预定目标来访问网站的，你只需提供能够满足他们需要的信息即可。如果你在网站上给出了过多的冗余信息或者过多的功能，你可能会让用户失去兴趣。
- 让应用类网站符合用户的心智模型同时与网站上下文环境相一致。当你开发一个网站时，特别要注意网站的工作流程以及用户对网站的期望。举例而言，如果网站是一个在线服装商店，一定要保证用户能够在在线商店中买到各种类型的衣服。如果网站同时还提供其他不同类型商品，用户就有可能将那个链接误认为是一个广告。网络用户早已对网络广告不感兴趣了，特别是那些网页顶部常见的横幅广告。
- 对于电子商务网站而言，要尽量降低处理流程产生混乱的可能。对于电子商务网站的交易结算流程而言更是如此，因为结算交易流程是最容易让用户感到混乱和挫折的部分了。但是，如果你能够仔细地设计这个流程，包括清晰的输入、说明和反馈，以及清晰的自始至终的处理过程，你就会使用户感到更满意，电子商务网站也会更成功。
- 仔细规划对用户数据的存取。大部分的暂态型网站都不能在客户端保持用户数据，所以，如果你需要让用户能够登录到网页上的特定区域(或者为了记住网站)，尽量将这个功能做成无缝而且明显的。比如，你可以通过用户账号和密码让用户登录到特定的客户页面上。

幸运的是，现代的浏览器(如微软的 IE 浏览器)都可以帮助用户保持他们的账号和密码信息，这样用户就只用输入账号而不必输入密码就可以登录网站。你可能还需要给用户关于创建安全密码的相关信息。

- 具备独占型姿态的应用类网站与类似的桌面站点几乎一样。丰富互联网应用程序(the Rich Internet Application)就是这样的一个例子，通过它，用户可以使用 Flash 来开发外观和行为与用户桌面类似的网站界面。正如具备独占型姿态的桌面应用程序，一个独占型姿态的网站应该由带有各种控件和对象的全屏应用程序组成，用户

可以方便地获得对互联网尽可能多的控制。

3. 网站门户

你可能已经听说过所谓的“门户”网站，即那些提供海量信息的站点。你可能会在互联网上遇到多种类型的门户网站(Cooper 和 Reimann, 2003):

面向客户的——这种网站提供与特定主题或一组主题相关的内容和功能。例如，MSN.com 门户(如图 8-5 所示)提供了与新闻和天气相关的主题信息，还在页面顶部提供其他一些诸如网络搜索服务等内容的链接。

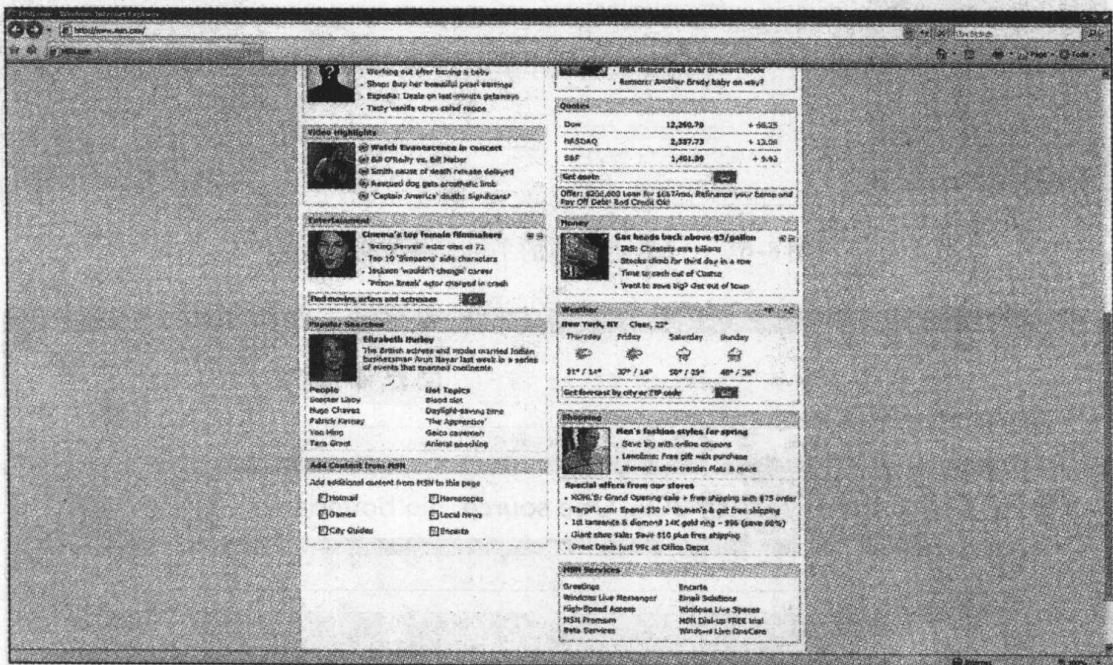


图 8-5 MSN.com 门户

企业门户——这种网站向用户提供了公司信息和商业工具。举例来说，这种网站提供服务允许用户通过企业门户远程登录站点，如图 8-6 所示。当你登录成功后，你可以对站点进行修改，比如添加电子邮件地址或者查看数据。

环境门户——这种网站是实际工作执行的地方，那些提供各种服务的门户网站是最佳范例。如图 8-7 所示，提供服务的环境门户允许主机客户通过点击图标和菜单表示的工具来更改他们的服务信息。

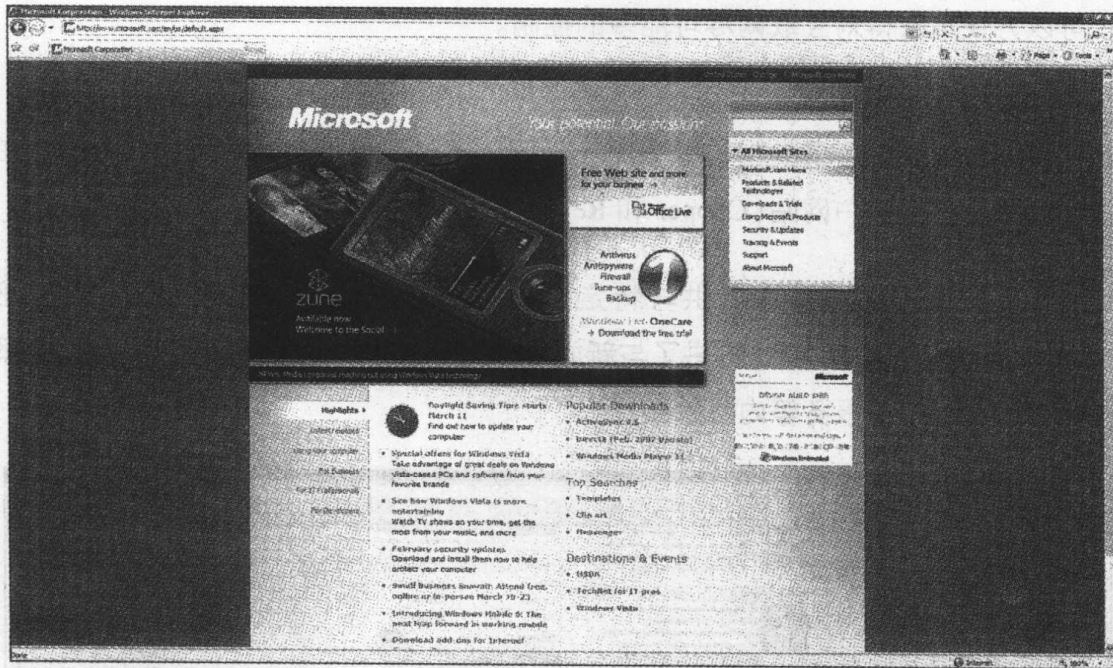


图 8-6 一个企业门户的主机服务控制面板

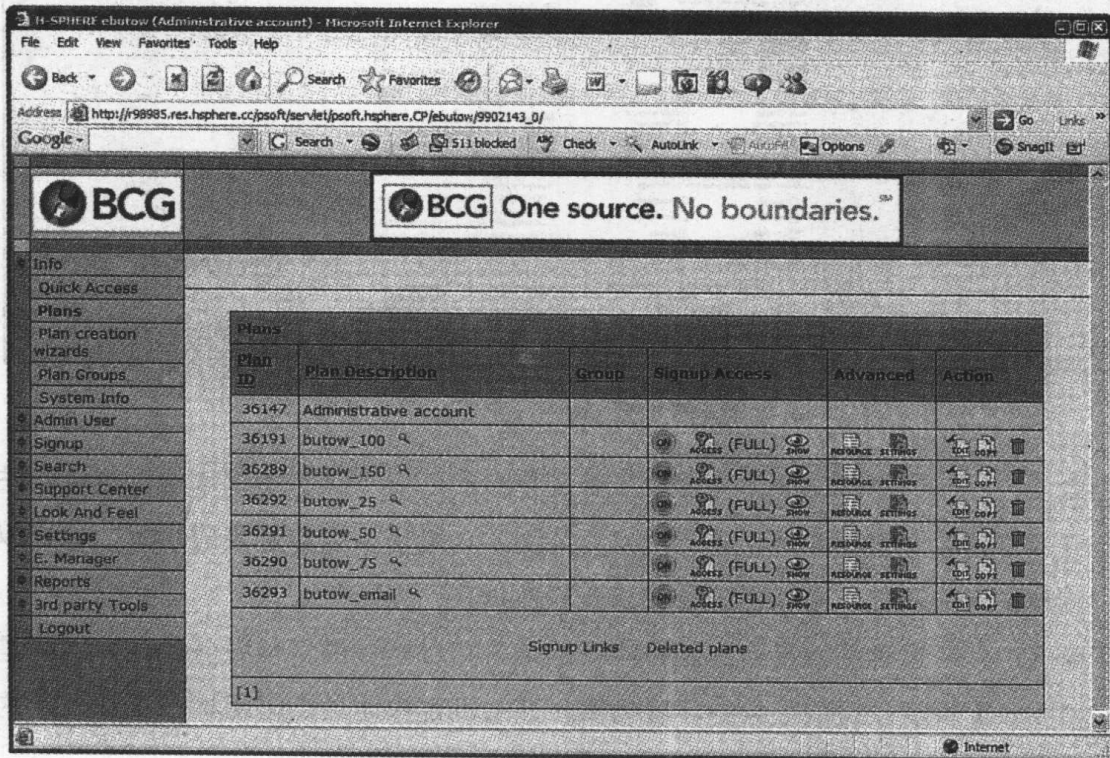


图 8-7 一个面向主机服务的环境门户

客户和企业门户通常具有暂态型姿态，因为它们的用户登录门户来找他们想要的东

西，都是很快地找到后便离开门户。对于环境门户而言，相当于在同时运行多个单独的应用程序元素。这些元素同时具有辅助型和暂态型姿态。

大部分的环境门户元素都有暂态型姿态，因为它们通常代表一系列用户能得以持续访问的信息。举例来说，对于提供服务的门户网站而言，用户可能要访问它的电子邮件地址列表。环境门户也包含了暂态型元素，用户需要时也只是短时间的使用，比如向主机服务的技术支持人员提交一份支持请求。

8.4 为什么需要网站工程

网站设计绝不仅是设计网页，而是设计一个网络系统。这种整体架构表明现代网站设计包括了基于网络技术的网站交互。网络技术包含了许多基于访问者与网站之间交互的处理方法。如果你需要这种工作，你就需要一个网站工程师与设计师一起工作来创建一个综合的整体系统，且它能够按照你和访问者期望的方式运作。

8.4.1 后端编程

网站通过浏览器提供信息，通过链接连接到其他站点。另一方面，网站应用程序可以根据它接收到的用户输入来创建网页。这些数据包括用户想从电子商务网站上购买的T恤数量，或者当用户填写了一份要求获得更多公司信息的在线表单时所要的目录类型。

网页的许多功能性，用户是看不到的。网站应用程序编程也被称作后端编程。用户实际看到的是前端，而大部分工作都是在后端进行的。网页和网站服务是双向工作的一种内容-处理关系。内容能够驱动处理，比如一个电子商务网站，但是处理也可以驱动内容。

8.4.2 表单处理

一个内容驱动处理的例子是网站表单，如图8-8所示。这个表单中包含了一个或多个允许用户输入信息的文本框，用户输入信息后点击页面底部的按钮即可将表单信息发送给服务器。如果后端编程认为用户正确地填写了所有内容，它就会对表单进行处理并通过电子邮件发送至预先设定的接收端。

处理也可以驱动数据，因为当网站完成处理后，应用程序会直接弹出一个感谢信息

网页。

另一个你必须注意的是表单的安全性。现在所有的浏览器都提供了良好的安全措施，从而让黑客们无法破解表单来盗取信用卡账号和其他敏感信息。当你与网络工程师一起工作时，一定要确保他了解最新的安全问题和编程，或者另找一个懂行的工程师。你可以只有一个包含敏感信息的网页，但是如果网站不够安全，那么用户也不会上面输入他们的信息。如果用户输入了他们的信息，而这些信息又被泄露，那么你就有可能会面临严重的法律问题。

BCG Customer Service Form - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://www.butov.net/form.htm

Google Search 511 blocked Check AutoLink AutoFill Options SnagIt

BCG

Customer Service Form

Thank you for filling out our customer service form. This form will help us understand your needs better so when we call you, we can provide you with more detailed information about how BCG can meet your needs. When you finish filling out the form, click the Send to BCG button at the bottom of the form.

You must fill out the form completely so we can contact you as quickly as possible. If you want to erase all fields in the form and start over, click the Start Over button at the bottom of the form. Thank you for your cooperation.

What is your name?

What is your e-mail address?

What is your telephone number?

Do you prefer being contacted by phone or e-mail?

E-mail Phone

Done Internet

图 8-8 一个表单的实例

8.4.3 数据库

后端网站编程一般与数据库接口，数据库里保存着用户可能感兴趣的数据。举例而言，一个电子商务网站常常包含销售产品的信息。数据库驱动内容，因为网站显示可销售的产品数量以及可以购买的产品还有多少（或者产品是否已售罄）。

网站访问者可以订购一个或者多种产品，以及明确每种产品的订购数量。当网站访问

者输入了他的订单和信用卡号码后，数据库就会更新信息，处理订单，并通知网站显示一个感谢页面。当数据库发生改变时，后端应用程序会自动改变现有产品数量，也可能会或不会将这个更新的数量反应到网站上。

8.5 网站标准

就像一个用户界面，网站界面也必须保证站点内所有的页面保持一致，这样给网站访问者带来的印象是：网站是一个有机整体。（没错，这又是整体性观念。）

8.5.1 颜色和文字

颜色和文字仍然要受到浏览器的限制。计算机用户在他们的系统中安装了各种不同的字体。他们的显示器也可能使用不同的颜色设定。举例来说，我的计算机可以显示出 400 万种颜色，而其他人的计算机可能只支持 65 536 种颜色。尽管浏览器和诸如 Adobe 的 Dreamweaver 等网络设计软件允许你在网页上使用任意颜色，但是在设计网站过程中涉及文字和图像时还是应该考虑最一般的设置限定。这个限定指的是网络安全的一组字体和 256 种颜色，所有的浏览器都可以显示。另外，你还要小心，注意避免使用可能有文化禁忌的颜色，以及那些会给色盲用户带来问题的颜色。

设计网站要做到，能够让访问者适当地在网站上阅读网页上的文字。有些对你而言很酷的文字，在某些访问者看来就不是那么回事了。设计时一定要把对比和简洁问题放在心上。举例来说，在白色背景下你应该使用黑色字体，从而获得最佳的对比效果。你还应该考虑什么字体是最具可读性的。很多人发现 Arial 字体在屏幕上要比 Times New Roman 字体更具可读性。

针对网站访问者，按照“最一般限定”进行设计的关键是什么？要记住目标是让网站在所有的浏览器上看起来都是一样的，差异会导致网站的显示效果不是你想要的。举例而言，如果你使用了一个访问者计算机上没有安装的字体，那么在他的计算机上就会使用其他的字体代替，这很可能导致网页布局的混乱，整个网站甚至都可能无法显示。如果访问者的计算机无法显示你使用的某种颜色，那么他的计算机就会使用另一种颜色来代替，那么访问者看到的图像有可能会跟你想要表达的意思大相径庭。例如，你本意是使用红色

来代表某种严重的情况，但是访问者的计算机不能正常显示这种颜色，那么你就无法将你要表达的信息传递给访问者。

另外，要尽量使用有品味的颜色和字体。如果你大量使用较亮的颜色，那么整个页面的颜色就可能不够协调。或者你使用古怪的和不专业字体，那么公司就有可能给用户留下类似的形象。

8.5.2 图片

如果你曾经上网冲浪，那么你一定在某些网站上碰到过滥用图形的例子。滥用图形的表现包括以下一些方面：

- 闪动文字，或闪动一块文字。这种情形在网页上的横幅广告上很常见，用户已经越来越习惯于忽略这些广告了。
- 分散用户阅读文字注意力的动画图片。
- 图片过多地聚集在网页上一块区域上，结果不仅分散了网站访问者的注意力，还会导致访问者的挫折感，因为所有的图像都在试图传递相同的信息。
- 图片分辨率过低导致的像素化和锯齿。

网站应该避免图片的不良使用。无论如何，网站的图像只要能有效地向访问者传递信息就可以了。

某些情况下，只需准备一张公司标识图片。如果网站是一个电子商务网站，你应该将数量有限的图片以合适的空间隔开，这样用户才能看到产品的相关信息而不至于被堆积在网页上的一堆图片弄的头晕脑胀。亚马逊电子商务网站在这方面就做得很好。亚马逊网站上有数量众多的各种各样的图片，但图片都被合理地隔开并配有相关文字，文字和图片之间也有适当的空白位置。

8.5.3 网站导航

网站应该很容易浏览，也就是如果你没有为访问者提供网站内其他相关网页的链接，以便让访问者能够方便地浏览，那么访问者就会感到挫折而且很可能会离开。如果网站是电子商务类网站，那么流失用户就意味着流失收益。如果没有用户访问网站，那么你还要

网站做什么？

8.5.4 面包屑链接

作为良好的导航的一部分，网站应该包括那些能够让用户返回主页或高级别页面（比如购物车页面）的链接。这些链接在网站上被称为**面包屑链接**，如图 8-9 所示。另一种替代方法是，你可以在每个网页上都设置导航条或导航区域供访问者使用。

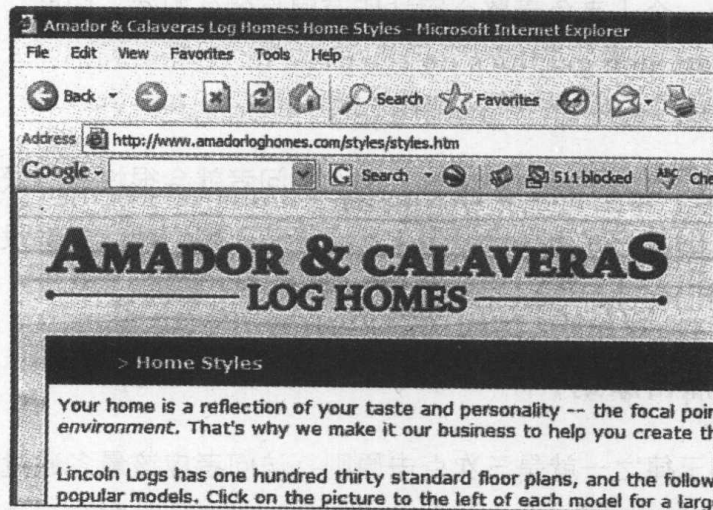


图 8-9 一个网站上的面包屑链接实例

8.6 四个规则

在设计网站时你应该遵循 4 个规则以保证网站尽可能地好用：保持简洁、保持一致、保持最新、保持三次点击原则。

8.6.1 保持简洁

在前面的章节中，你可能已经了解到网站应该尽可能简洁。当然也会有些例外，我将在下一节中详细讨论，但是你必须让网站布局尽可能简单。而且，你应该记得第 7 章中提到的西方国家的阅读习惯，即从左至右，从上至下进行阅读。但并不是所有的访问者都有这样的习惯，所以我在第 7 章中给出的建议是你应该进行用户和任务分析，了解他们来自哪里以及想在网站上获得什么信息等。

8.6.2 保持一致

与用户界面类似，网站界面应该保证在所有的页面之间保持一致。如果网站中某些网页的外观突然改变了，那么访问者就会感到迷惑，甚至误以为他们连接到了另一个完全不同的网站上了。对于软件应用程序的用户界面而言，这种情况会导致其他客户的调用。对于一个网站而言，这样的情况只会让访问者离开而再也不会返回。为了保持网站一致性，一种有效的方式是让一个人来负责整个网站所有网页的外观统一问题。

8.6.3 保持最新

如果你不将最新的信息更新到网站上，那么访问者就会很快感到厌倦而离开网站。如果访问者发现网站上的时间和文字已经很久没有更新，他们就会觉得没理由再返回网站查看更新信息。或者他们会得到这样的印象：公司不够关心客户。

8.6.4 保持三次点击原则

网站设计的金科玉律之一就是三次点击原则：访问者应该最多通过三次点击就能在网站上找到他们想要的信息。如果访问者通过三次点击还是找不到他们想看的内容，那么你就需要重新设计网站结构了。

8.7 何时打破规则

上一节提到的4个规则并不是不可打破的。当这些规则不适应公司或者用户的需求时，你就需要打破这些规则。

8.7.1 打破图形用户界面规则

如果公司现在已经有正在使用的标准，你可能需要打破图形用户界面规则。比如，公司使用面向用户需求的特定软件应用程序用户界面，用户也已经习惯了这个界面，那么你就还应该按照先前的标准进行设计。

尽管用户界面满足图形用户界面规则，但是用户测试结果可能表明某些界面让他们感到不舒服。在这种情况下，你需要判断用户想要看到哪些变化以及是否要进行这些改变。

8.7.2 打破网站设计规则

你可能会设置太多的选项而无法满足三次点击规则。举例来说，如果你有一个电子商务网站，用户可能就需要点击三次以上才能找到他想要的商品。如果这种情况下一定要符合三次点击规则，那么就可能造成用户在一个页面上必须面对大量的可选商品，这会使用户感到困惑，甚至索性什么都不买了。

如果你需要更丰富的界面来进行复杂的处理过程，例如我们在前面章节中提到的丰富互联网应用程序，一些网站设计规则可能就不适用于该项目。

如果你需要在公司内部网上连接一个单独的应用程序，比如微软 Outlook 基于网络的组件，以便通过网络应用程序来组织会议，而且网站也符合这些标准，那么其他的网站规则可能也不能适用于网站。

你也可能根据组织中其他部门的需求而打破网站设计规则。例如，网站需要符合某种特定的外观和行为方式，而你可能需要将已有网站的某些特性移植到新的网站上，这样两个网站才能保持一致。

8.8 案例研究：界面导航特性

你为 Mike 的自行车店更新的数据库应用程序包含了一个基于网络的界面并被集成到了公司的内部网络中，这样每个人都可以通过网络浏览器使用应用程序。在前面的章节中，我们讨论了需要你和 Evan 检查完整的界面来确保纸上原型测试中的计算机(由 Evan 扮演)可以准确地反映出当前地视觉和听觉提示。

那些视觉暗示也包含了用户界面的颜色和导航。幸运的是，你有一台彩色激光打印机，这样当你制作纸上原型测试用到屏幕截图材料时，就可以方便地得到全彩色的屏幕截图代替卡片。当你为测试者添加新的屏幕时，你和 Evan 也决定使用彩色笔来绘制，以便尽可能地反映出页面上的颜色信息，从而让测试者尽可能感到一致。

你还必须了解界面导航特性。现在 Mike 自行车网站上的应用程序在每个页面上都有面包屑导航，以及位于屏幕左侧的导航菜单按钮条，这样用户就可以在各个页面之间方便

地切换, 参见图 8-10。新界面也要实现这些特性, 因为新增页面中在导航按钮条上添加了一个产品有效性按钮用于链接到相应的页面, 所以你需要在纸上原型测试中为每个屏幕截图卡片的导航条上添加这个按钮, 如图 8-10 所示。

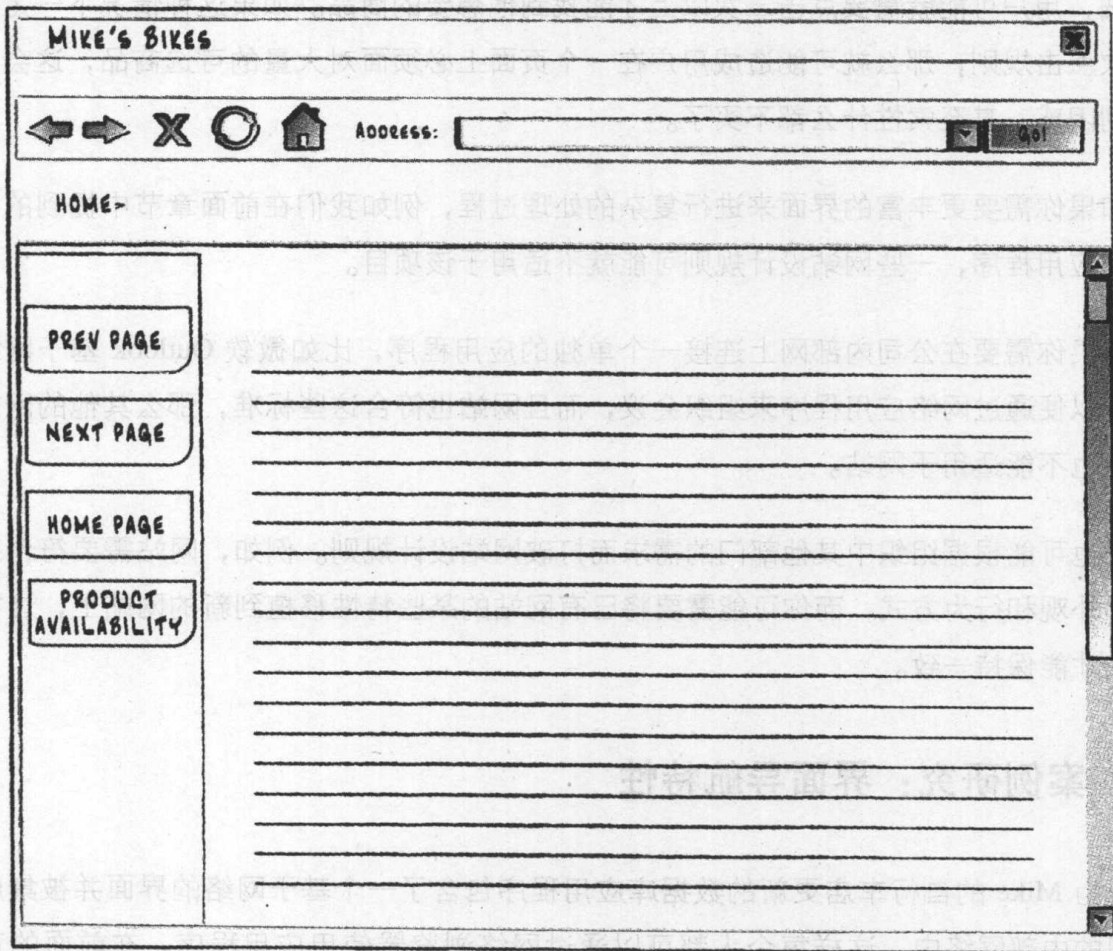


图 8-10 带有产品有效性按钮的新屏幕

尽管你不必在纸上原型中注意诸如让新加的产品有效性按钮与其他按钮使用同样的颜色这样的细节, 但有时确实需要使用不同的颜色来强调某些内容。举例来说, 在 Mike 的自行车应用程序中, 你要在产品标签页下使用不同于其他按钮的颜色添加一个新的网站订购按钮。你得选用一个带颜色(本例中使用黄色)的标识使其与邻近的按钮区别开来, 这样网站上的订购按钮就会显得突出。图 8-11 是灰度图, 但你还是可以看出相邻的高亮色按钮和较暗色按钮之间的区别, 使得订购按钮更能够吸引用户的注意。



图 8-11 订购按钮使用的是不同的颜色

纸上原型测试可以确认你选择的颜色是否会被用户所接受。说到测试，现在该是你和Evan 两人进行测试试验的时候了。然后，你就可以准备实际测试了，下一章会详细讨论实际测试过程。

8.9 小结

本章开头讨论了网站界面与图形用户界面之间的相似之处与差异，以及技术是如何模糊二者之间的界限，使得桌面界面具备网络功能。这些技术包括 Java 编程语言，ActiveX 和 PHP 脚本语言，Adobe 的 Flash 动画软件，AJAX 和动态 HTML，以及诸如微软的 .NET 语言等更专有化的解决方案。

然后，你学到了关于网络的用法、设计和可及性等迷思。在每个迷思后面，我们给出了一些如何避免这些问题的建议，这样在开发过程中如果遇到关于这些迷思的讨论时你就能给出真实的信息了。

接着我们讨论了网站姿态问题，以及所有 3 种类型网站可能具有的姿态，这 3 类网站指的是信息类网站、应用类网站和门户类网站。你知道了要创建的网站类型决定了你设计网站的姿态类型，反过来(按照主要人物角色需求)姿态的类型决定了你要让网站包含哪些特性。

在为什么需要网站工程的部分，我们讨论了网站系统的设计，当今网站如何动态生成网页，以及网站访问者看到的前端和让网站具有功能性和可用性的后端编程之间的交互。你知道了内容可以驱动处理，比如一个电子商务网站的情况，但处理也可以驱动内容。内容驱动处理的一个例子是网站表单。处理也可以驱动内容，因为当网站完成一个处理后，应用程序可以直接弹出一个感谢信息页面。

随后我们讨论了网站标准。你学习了必须注意的网站的 4 个方面标准：颜色和文字，图片，导航，以及留下面包屑链接。你知道如何使用颜色和文字以保证最大的可读性，如何避免使用闪动文字和可能分散访问者注意力的动画图片。作为网站导航的一部分，你学会了应该在网页上设置可以切换到其他网页的链接以及用于返回主页或高级别页面的面包屑链接。

本章最后讨论了网站设计的4个规则：保持简洁，保持一致，保持最新，以及保持三次点击原则。同时也探讨了何时应该打破这些规则，比如当公司已经有自己的规则，你需要界面跟你公司的其他工具界面保持一致，以及用户测试表明存在需要修正的界面可用性问题，即使这些改变与规则格格不入。

复习题

1. 图形用户界面和网站界面之间的相似之处表现在哪些方面？
2. 图形用户界面和网站界面的相似之处是越来越多还是越少？为什么？
3. 为什么你需要了解关于网站的一些迷思？
4. 网站迷思分为哪3类？
5. 为什么说了解网站姿态非常重要？
6. 网站的3种不同类型分别是什么？
7. 你可以创建的3种类型网站是哪些？
8. 一个内容驱动处理的例子是什么？
9. 一个处理驱动内容的例子是什么？
10. 为什么你必须限制颜色和文字选择？
11. 滥用图形的4个表现特征是什么？
12. 为什么必须在任何可能的时候都要坚持网站设计的4个原则？
13. 什么是三次点击原则？
14. 什么情况下可以打破网站设计原则？



可用性

“假设本身没错，但得到结果更好。”

——Samuel Clemens

本章内容提要

- 为可用性测试选择合适的方法
- 定义可用性测试
- 执行可用性测试
- 分析和陈述可用性测试结果

当设计好软件、硬件产品或者网站的界面后，就该将设计交付测试，让用户体验你设计的界面，这样在把产品向公众发布之前，就能有机会根据用户的反馈情况对设计界面进行必要的修改。

你应该首先选择可用性测试中所选用的方法。你也许会有所顾虑，本章会讨论你可能会遇到的一些问题，并给出一些能消除你顾虑的答案。

要知道可用性测试应该使用哪种方法，需要明确你要从用户那里获取什么样的信息，以及你要如何收集这些信息。

当设计好初始测试后，你需要进行一次预备性测试，用以锻炼你测试中所需的观察和访谈技巧。熟悉测试操作流程之后，你就可以准备进行真正的可用性测试了。

测试完成后，就轮到你对测试数据进行分析整理，准备测试报告，并向项目关系人进行相关的陈述，告诉他们你对于如何提升用户界面和整个产品的建议。

9.1 为可用性测试选择合适的方法

你可以使用多种方法进行可用性测试(Hackos 和 Redish, 1998)。根据实际情况，你可以选择其中一个或者所有的技术来获取你需要的信息来改进产品或者文档。

9.1.1 观察、倾听和交流

其中一种方法是直接去用户的所在地进行(现场)面对面的访谈。当你计划进行产品走访时，你可能会有所顾忌，特别是对于走访用户的预算花费较大的情况更是如此。如果你遇到了下面提到的这些非常普遍的问题((Hackos 和 Redish, 1998)，那么你应该据理力争：

- **到底为什么要去？**——我们必须挑战或者检验我们的假设。毕竟，我们并没有亲眼见到用户是如何使用我们的产品进行工作的，也许实际情况会让你惊讶。这些信息会影响我们的产品及其文档。
- **这是一款新产品，为什么还要出去测试？**——我们可以挑战我们的假设，看看设计的产品是否适合客户的工作流程，用户是如何使用新产品的，从而能够顺畅地完成工作。
- **我们仅仅修改了部分功能，为什么还要进行完整测试？**——现场可用性研究显然可以影响功能改变部分。而且，那一部分的改变很可能会影响用户的工作流程，因此还是需要对其进行完整的分析和测试。
- **我们能从一个新用户那里获得什么？**——Nielsen(1993)和Virzi(1992)所做的关于可用性测试的研究表明，测试人员可以从每个用户组的6~8个人中收集到高比例的用户需求。超过8个用户时，我们会看到大致相同的结果。对一个较小的用户组进行测试不但可以获得分析所需的数据信息，而且可以有效地降低测试费用。

- **为什么不用我们已有的用户测试信息？**——我们需要了解客户对这个产品功能的看法，而不是客户现在或者曾经的使用情况。

正如我在第3章中提到的，一定要进行投资回报率分析，其中就要包括进行访谈所需要的成本预算。

当你在现场观察用户时，你需要在规划任何访谈时牢记以下原则(Hackos 和 Redish, 1998)：

- 提前计划。了解访谈话题和目的。
- 在用户群中选择体现差异化的用户。
- 把用户当作合作伙伴。
- 观察，聆听，并与用户交谈。通常，这些需要同时进行，观察用户如何在他们的实际环境中工作。
- 通过在用户使用过程中及时与之交谈确保谈话内容足够具体。
- 从用户那里获得提示。你应该与用户分享你的理解以确保你所看到的和听到的没有被误解。

如果你把这些信息特别是访谈计划告知决策者，这将会对测试更有利。

9.1.2 其他用户交互方法

另外有一些用户交互方法你可以与走访联合或者先后使用(Hackos 和 Redish, 1998)。这些方法跟前面章节中出现的一些相关概念类似，包括远离客户现场的可用性评估，以及一些更传统的市场方法：

- **可用性圆桌**——客户到公司，并将工作一起带过来，这样你就可以针对客户的工作现场进行产品的可用性评估。
- **受控的可用性评估**——你可以预先设计可用性测试，然后在一个特定的环境中进行测试，比如在一个安装了被测试产品的计算机的实验室中。
- **焦点小组**——你可以把用户带到你公司，通过8~12个人的群组来收集信息，包括用户对产品、想法和客户需求的态度的反应和意见。
- **让客户参加需求收集会议**——产品开发团队通过访谈客户来确认客户到底想在产品

中看到什么特性。

- **让一个或多个用户加入设计团队**——让用户参加设计团队有益于在整个产品生命周期中保持长远的用户目标和期望。
- **用户问卷调查**——你可以利用可用性工程生命周期(UEL)第三阶段中讨论过的用户问卷调查来收集信息,并将其应用于产品升级过程中。
- **在交易会上会见用户**——产品交易会和其他专业协会会议是进行用户信息收集的非常好的机会。

上述这些方法都有自己的缺点,大体上可以分为3类:缺乏足够反馈而造成的偏见,用户提供信息量不足和误解。这些误解往往是由混乱、沟通不畅或者不能亲眼看到用户是如何使用产品造成的。

进行可用性测试没有一个最佳方法。但是,通过现场观察用户使用产品已被证明是收集可用性信息最有效的方法(Hackos和Redish,1998)。

9.2 定义可用性测试

整个测试过程的第一步是定义和规划可用性测试。如果你不知道如何进行测试或者测试目的不明确的话,那么你就会浪费一大批人的时间,更别说浪费了公司的资金。你应该对进行的测试、项目,可用性团队成员的职责,以及团队所做的各种决定等做好详尽的记录。

——Dumas和Redish,1999

Dumas和Redish(1999)给出了在你定义可用性测试过程中必须完成的5项任务:

1. 定义目标和关注点。
2. 确认测试参与者。
3. 选择、组织并创建测试情景。
4. 确定测量可用性的方法。

5. 准备测试材料。

下面的章节描述要完成的任务(Dumas 和 Redish, 1999)。

9.2.1 目标和关注点

当你确认了用户,你就得为可用性测试选择目标用户——比如说,是选择专门针对专家用户还是面向大部分的中间用户。确定测试目标可以先从总体目标开始,然后再确定特定目标。我们可以从下面资源中获得目标:

- 任务分析和量化可用性目标,与你在开发纸上原型设计的那些一样。
- 时间性问题,比如需要进行一个可用性研究来解决是否添加某项特性的问题。
- 启发式分析或者专家评审,比如需要关心内部客户(如市场)的意见。
- 该产品或者其他产品以前的测试。一项测试可能会激发其他测试需求。

9.2.2 选择测试参与者

在选择参加可用性测试对象的时候,你一定要慎重。当你按照第6章“用户分析”中的方法创建人物角色时,你实际上是在决定每个测试对象应该具有的特征。你需要考虑2种不同类型的特征:一种是所有用户共有的,另一种是不同用户之间所特有的。下面给出了在决定用户特征时你需要考虑的因素:

- 用户使用计算机或者被测试产品的经验
- 用户的工作经验
- 用户使用你的产品的经验
- 用户使用类似产品的经验

在设定用户角色特征时应尽量考虑全面一些。下面是一些全面考虑的例子:

- 考虑新参加工作的用户。
- 考虑即将使用产品的新客户。
- 如果产品仅在一个内部群组使用,考虑可能会在其他群组使用的情况。
- 考虑一个类别下的差别。比如,较年轻的使用者会比年长的使用者更快地接受新技术。

这里，你可以创建具备相似特征的用户群组甚至子群组，从而可以了解不同子群组中的用户对产品的同一个新特性的差别。

9.2.3 选择、组织并创建测试实例

但不幸的是，你不可能让用户对产品的所有可能的任务进行测试。那么，你怎样才能缩小这个范围呢？关注下列任务：

- 探查潜在可用性问题的任务；
- 根据关注点和经验提出的任务；
- 源于其他评价标准的任务；
- 用户使用产品工作的任务；

在你选择任务的时候，你一定要考虑到该任务会耗用户多长时间来完成，需要什么硬件、软件、过程以及其他必需的信息来完成这个任务等。你应该给每个任务进行编号并给出每个任务的描述（就像你在纸上原型设计中做的那样）。每个任务应该表明所需的时间，所需的软硬件，以及完成任务所需的提要说明和过程。

你可以使用一个情境来告诉参与者你想让他们在测试过程中如何操作。所谓情境就是一种描述任务的方法，有助于用户理解任务的含义。比如，“你有三个新雇员。给他们添加账户。”

一个好的情境应该具有以下特点：

- 短小精悍。
- 使用用户的语言。
- 意思明确。
- 为参与者完成任务提供了足够的信息。
- 直接与任务和关注点相关。

任务和情境不一定非要写下来。你可以找人扮演各种角色，比如客户，客服人员或者管理人员等。你也可以让参与者在每个任务之间停下来，比如在一个较长时间的任务后或者你需要在每个任务完成后，给所有的参与者一份打印的调查表等。但是，你必须给出声

音提示从而让参与者暂停和继续任务，因为参与者可能专注于完成任务而不记得要暂停和继续。

9.2.4 确定测量可用性的方法

可用性测量具有2个维度：

- **性能测量**——对特定操作和行为进行定量测量。
- **主观测量**——人们的感受，看法和判断等。

在性能测量情况下，你可以方便地将测试过程中用户每次做出的特定行为都记录下来，比如用户对使用产品过程的挫折抱怨等。主观测量则难以用数量来表示，除非多个参与者向你说了同样的事，比如页面上的电子邮件按钮不好找到等。市面上有专门用于记录可用性数据的商业应用程序，或者你也可以让产品团队中的程序员为你创建一个满足你特定需求的软件。如果你不能使用软件应用程序，你也可以使用打印的表格来记录数据。

创建记录表格时，你需要为性能测量设置标准。性能测量的一个典型标准是四分制，即强制进行正面的或负面的选择，因为不存在没有倾向性的反应。实际上，这种四分制有3个通过级别和1个未通过级别。你还必须将性能测量设置与一般的和特定的关注点直接相关。比如，假设你关心的是用户读取一条信息是否足够简单，那么你就可以加入诸如用户完成任务所需的时间以及从错误中恢复所花费的时间等测量标准。

你可能想要在可能的情况下对一同测试的绝大多数任务进行相同的性能测量，用以判断用户完成任务的情况。但是，同一测试中的不同任务有时需要不同的性能测量。举例而言，一个网页上的某项功能可能在其下一级页面中不再提供，那么这种情况下你就不能将其作为子页面的错误记录下来。

你还必须把测试环境也考虑进来，比如参与者是否需要在每个任务之前阅读操作说明等。如果要测试用户完成一项任务所用的时间，你还必须把用户时间测试情况时间加入到总的时间内。比如，你应该在测试开始时间前加30秒，以给用户充足的时间来阅读和领会操作说明。

9.2.5 准备测试材料

开始测试之前，你必须为参与者准备好合法表格。作为测试者，你要为以下各项

负责：

- 创建用以明确当事人各方权利的合法表格。
- 确保所有的参与者都阅读并理解这份表格。
- 观察或者证明所有的参与者都签署了表格。

在创建这样的表格时，你应该向你公司的法律部门或者律师（如果可能的话）进行咨询，并尽可能将他们呈现给参与者。如果需要对表格进行解释和阐述，那么一定要用中立而友善的措辞。

你还需要准备一个测试脚本，这样就可以对所有群组的所有用户进行相同的测试。如果你还记得高中时的标准化考试，你一定还记得老师们都是按照同样的脚本以确保每个人都按照相同的方法测试，这样就不会影响测试结果，同时保证所有的测试都在相同的时间内完成。测试脚本应该包括一个任务清单，用以标识所有的任务是否都已完成。如果其他团队成员有他们自己的任务清单，你也必须保证他们也完成了清单上所有的任务。

你可能想要在以下 3 个时间点分发调查表，用以从用户那里获得信息：

- **测试前**——收集参与者的背景资料。
- **任务后**——收集每个任务的判断和评估。
- **测试后**——收集整个测试的判断和评估。

书写的调查问卷是有用而且有效的，因为你问所有的参与者同样的问题，并且你不会忘记回答这些问题。但是，你必须确保调查问卷中都是正确的问题，这样才能得到有效的答案。比如说，假设你想询问关于完成任务的难度的问题，让参与者对难度进行 1 至 5（5 代表非常困难）级别的选择要比只做固定的二元判断要有效。

9.3 执行可用性测试

现在该通过第一次预备性测试来做准备，看看测试过程是否合理。完成预备性测试后，在开始进行正式可用性测试之前你还需要特别学习如何关注测试伙伴。

9.3.1 执行预备性测试

在正式可用性测试开始之前，你因该进行一次预备性测试(Hackos 和 Redish, 1998)。通过预备性测试你可以“调试”可用性测试，从而发现要测试产品或者网站及其文档、测试方法和测试材料等的原始问题。在预备性测试中你可能遇到的问题包括：

- 由于硬件、软件或者网站本身的问题，参与者无法完成某项任务。
- 参与者通过一种你预料之外的过程完成了任务。
- 参与者仅完成了部分任务。
- 参与者持续犯同样的错误。
- 参与者缺少完成任务所需的材料。
- 问题或者答案让参与者感到迷惑。
- 文档中的说明让参与者感到迷惑。
- 由于产品的一项基础可用性问题的导致参与者无法完成任务。

预备性测试一定要跟正式测试的流程完全一样，而且参与者应该能代表你想要测试的用户。通过模拟真实的可用性测试过程，预备性测试能够给你最准确的结果。预备性测试也可以让你测试你接近用户的方法。

为了能够有充足的时间调整可用性测试，预备性测试一般应安排在正式测试的 2 天前进行。这样你可以有一整天时间(如果你把预备性测试安排在早上可能有更多时间)来对正式的可用性测试进行调整，而不会感到时间的压力。如果在预备性测试中暴露出确实需要更改的问题，你可以决定是否升级这个问题。

9.3.2 锻炼观察技巧

通过预备性测试，你可以得到一些能够增进技巧的线索，特别是当你准备到客户所在地观察他们是如何使用产品工作的时候。一次成功的走访会有许多因素(Hackos 和 Redish, 1998)。在你去走访客户之前，请记住下面的原则：

- 切勿不请自来。一定要跟管理人员和走访用户安排好访谈事宜，让他们大体知道你将要做些什么事情。

- 通过各种渠道工作，包括必需的管理人员和协会组织等。
- 安排好各用户的走访日程，确认彼此认可的日期和时间。
- 一旦安排好走访行程就应尽早通过传真进行确认，并在走访前一天进行电话联系。
- 需要的话，打听清楚走访用户线路。你一定不想迟到吧。
- 到用户现场搭建任何设备之前，一定要在公司里进行相应的练习，以避免到时出现一些问题和麻烦。

当你到达用户处，执行下面的步骤：

- 跟管理人员和用户打招呼，如果用户同事在场的话也要跟他们打招呼。
- 需要的话，在征得同意的前提下搭建测试用视听设备。
- 核实这次走访测试预期的时间和相关程序。

在现场的时候，按照下面的提示行事：

- 尽量能让用户配合整个测试过程。确保与用户之间建立良好的关系，必要时帮助他们恢复使用产品的信心。当然，还要让用户明白你关心的是他们是如何使用产品工作的，你不要像专家一样解释应该怎么做。
- 测试安排应该尽量满足用户的日程和需求，比如适当的休息等。
- 与用户进行交流时，你使用的语言要是友善和中立的。
- 尽量多做记录，越多越好。

离开的时候，要记得：

- 感谢用户和管理人员。
- 即使是你自费进行走访也要给用户一个礼物。
- 安静、快速地打包行李。

离开后还要记得给用户和管理人员寄一封感谢信。

做记录

当你作为一个观察者做记录的时候，一定要遵循一个合适的记录格式，以保证方便地记录用户使用产品过程中的一些重要信息以及提问答案。尽管记录格式应该根据特定的可

用性需求确定(可能还需要满足实际测试用户的需求),但至少应包括以下一些内容:

- 项目名称。
- 观察表格编号。因为在走访过程中你可能会采访多个用户。
- 观察人员名字。
- 观察实施日期。
- 观察表格总页数及其每页页码。
- 用户目标。
- 用户任务。
- 有关用户的记录。
- 有关环境的记录。
- 任务开始之初的情况。
- 任务开始时间。
- 任务进行过程中对用户的观察。
- 任务结束时间。
- 任务结束时的情况。
- 可用性目标已经满足的任务结束点。

你还应该在观察过程中记录下关于用户和任务的暗示和问题。在观察的过程中要尽可能地用户多提问题,以获取尽可能多的信息。

9.3.3 锻炼访谈技巧

你可以在用户执行任务的时候进行访谈,当然也可以根据实际用户对象和最适合的预备性可用性测试所使用的方法和技巧来决定询问时机。除了用户执行任务的过程进行所谓的现场访谈之外,你还可以选择以下几种访谈方式(Hackos和Redish,1998):

- **立即回忆访谈**——记录用户的行为,然后在用户完成任务后立即进行访谈。
- **提示回忆访谈**——记录用户的行为,在用户完成任务一段时间后进行访谈,这个过程中可能需要将录制的视频回放。
- **流程访谈**——对单个用户或整个工作组进行访谈,了解整个工作流程。
- **多用户访谈**——首先访谈一个用户得到关键信息,然后再访谈其他用户,并在访谈

过程中观察相关的讨论。

- **对人工制品预排的线索回忆或基于语篇的访谈**——从用户那里收集典型结果，然后围绕这些结果进行访谈。
- **关键事件访谈**——在观察到关键事件发生时对用户进行访谈。
- **群组访谈或焦点小组访谈**——访谈单个用户或用户组关于产品的看法、期待、偏好等。
- **可用性圆桌访谈**——在用户工作地点外对用户进行访谈。
- **客户合伙访谈**——与一组用户一起工作一段时间，在工作过程中进行访谈。

无论你决定采用哪种访谈方式，你必须在访谈过程中牢记以下3件事情：

- 把用户当作伙伴对待，而不是研究对象。
- 假定用户非常了解他们的工作。
- 多听少说。正如俗语所说，人有两个耳朵和一张嘴，应该按比例使用。

基本技巧

在上面提到的3个访谈准则中，蕴含了一系列你应该掌握的基本技巧，只有这样才能从访谈过程中获得更多信息。

- 针对不同的角色和知识设置期望值。
- 提前为访谈准备一些问题。
- 明确你想要学到什么东西。
- 认识到通过不同类型的问题才能获得你想要的答案。比如，在测试的不同阶段，你可能要向用户提出笼统的或具体的问题来获得所需的信息。
- 提中性问题而不要提倾向性问题。
- 允许沉默，必要的时候留出时间让用户思考他们的答案(或者他们的问题)。
- 注意用户的身体语言和其他信号，以及其中的文化差异。
- 捕捉用户话语背后的真正含义。
- 严格执行走访用户计划。
- 让用户有机会回答你并没有提出的问题。
- 尊重用户的提问，并承诺将用户所关心的、要求的、赞扬的以及其他评论反馈给适当的人。然后遵守承诺。

9.3.4 持续关系

有时访谈要求你与用户要建立一种持续关系来跟踪产品或者文档的发展。随着用户从新手发展成专家，他们会经历以下几个阶段(Kuniavsky, 2003)：

1. 他们会犯错误。
2. 他们会对产品使用方法或者文档阅读方式建立心智模型。
3. 他们会随着对产品或者文档的熟悉程度逐渐形成期望。
4. 他们会形成使用产品的习惯，这个习惯可能会，也可能不会是有效的完成任务的方法。
5. 他们会在获得更多产品使用经验后形成自己的选择。他们可能会喜欢或者不喜欢某个特定的功能或者子系统。
6. 最后，他们会根据他们的目标、工作以及与其他工具的关系将产品视为不可或缺的工具，正如分类用的书架或者收集灰尘用的抽屉。

有多种方法可在一段较长的时间里获得可用性信息(Dumas 和 Redish, 1999)：

- 让用户填写日记式的调查问卷表格。测试人员定期给用户发送新的表格，可以通过再次走访、电子邮件、传真，甚至传统邮寄的方式。这些表格可以都使用相同的问题，也可以是不同的特定问题。
- 建立专门的咨询委员会用来定期会见用户，这样用户就可以直接将信息反馈给项目团队。
- 将不同经验级别的用户放到一起进行研究，从而可以看出不同的问题是如何影响用户的。
- 如果你关心的是一个网站的可用性，那么你应该获取用户的登录信息用以统计用户在哪些页面上花费了更多的时间等问题。

9.3.5 关心测试参与者

你需要关心测试参与者。当你开始测试时，你要保证测试人员都是舒适的，而且你也

在平静地关注他们。一份清单能够帮助你和测试团队保持一致，并确保从测试开始就与测试人员建立友善关系 (Dumas and Redish, 1999)。

友善关系的建立可以包括一些谈话，让测试工作者和测试人员做自我介绍，内容可以是关于他们的工作、机构，以及想从测试中得到什么。你还应该与参与者谈论测试环境，如果你准备有录像带，可以放给他们看。如果你有专门的测试房间，无论是在用户处还是你自己那里，你都应该向用户介绍测试房间中一直会在测试工程中观察他们的监视设备。

9.3.6 进行真实测试

在进行测试的过程中，要提醒测试人员尽可能地把他们所想的大声说出来 (Dumas 和 Redish, 1999)。将所想的大声说出来有助于测试人员关注并理解他们之所想。这种做法的成功率可能会有所不同，因为有些测试者要比其他人更愿意分享他们的想法。

有时测试过程会出现偏差。下面列出了一些常见状况以及你可以做出的反应 (Dumas 和 Redish, 1999)：

- 有的测试者表示不愿再参与测试。这种情况下，你应该搞清楚是什么原因导致他不愿继续参加测试。如果你无法缓解这种状况，你就必须让该测试者离开。
- 有的测试者非常紧张，以至于无法继续测试。你应该让测试暂停一下，跟那个感觉紧张的测试者进行一对一的交谈，看看你能否让他放松下来。如果不行的话，你就不得不过让该测试者离开。
- 有的测试者不具备参加测试的能力。你通常会在测试前的调查问卷中发现测试者是否具备参加测试的能力，但你也有可能是在测试进行过程中才发现这个问题。这种情况下，你必须立即让该测试者离开。
- 有的测试者花费了过多的时间来完成任务。在测试开始之前，你应该清楚即使一个或多个测试者没有完成特定任务，测试群组也应该继续下一个任务。另外，需要指出的是时间问题不在于测试者，而在于产品本身。如果一个测试者花费了太长的时间，那么有可能表明用户界面难于使用。
- 如果产品设备或者软件失效，你应该将测试组转移到其他房间并解释为什么会这样，同时其他团队成员应试图找出问题的原因。如果你决定停止测试，你必须决定是否重新安排测试日程。

- 有的测试者因为产品难以使用而感到挫折或者紧张。在这种情况下，你可能需要跟该测试者一对一地谈一谈，决定其是否能够继续参加测试。你应该向他指出他碰到的问题是非常有价值的，解决这些问题需要他的反馈信息。

注意，如果你给参加测试者付报酬，你就要根据具体情况决定如何给中途离开的测试者付报酬。

在测试过程中，你应该始终关注问题并将创建一个问题列表。测试团队中的其他成员也应该创建自己的问题列表，因为不同的人观察问题的角度不同所以看到的问题也不一样。写下观察，并给出你观察到的行为的臆测以及解释。尽量保证观察的中立性，并记录下所有用户的问题。只有这样，你才能得到一个完整的问题列表而没有任何遗漏。写下你所有的问题后，你可能会就其中的一些问题跟测试者进行讨论以获得更详细的信息。测试完成后，你应该跟测试团队和项目团队讨论你发现的问题。

本章中的一些内容可能会与前面某些章节的内容有所重复，但你会发现所有这些内容是如何整合在一起的，从而你可以进行可用性测试。在你完成了可用性测试之后，你必须分析并陈述数据，然后给出你建议的行动计划，这些内容我们将在下一节中讨论。

9.4 分析并陈述可用性测试结果

可用性测试会产生大量的需要你检查的数据(Dumas 和 Redish, 1999)。测试完成后，你可能会得到下面一些数据：

- 测试中的问题列表。
- 测试中关于时间、错误和其他性能测试数据，还包括调查问卷中的主观评分。
- 日志、笔记和调查问卷中测试者的评论。
- 测试团队的笔记记录。
- 测试参与者的背景资料数据。
- 测试过程的录像，可以从测试房间的多个角度获得。

9.4.1 分析并陈述数据

分析数据的第一步是把所有的数值(定量)数据汇总成数据表格。这个过程你可以使用

任意一种电子表格软件来完成。你也可以将所有的评论在字处理软件中进行汇总。一些功能强大的办公处理软件，诸如微软的 Office 套件，可以将 Excel 中表格数据内嵌到 Word 文档中。当你在 Excel 中对表格进行修改后，所做的改变会自动在 Word 中得到更新。

当你将所有的数据都输入完成后，你就可以分析这些数据来发现一些趋势和问题。电子表格处理软件在对数据进行一些计算时也非常有用，比如统计所有调查问卷中某个问题的平均得分等。但是，正如马克·吐温所说，总存在着“谎言，可憎的谎言和数据”的问题，特别是对于一些推论性的数据而言。推论性数据是从大量数据中的采样得来的，通过采样数据的结果来推论全部数据。这种方法跟描述性统计不同，描述性统计是描述一组数据，比如执行某个任务的平均时间等。

一个有用的处理数据的技术是对其进行三角形化(Dumas 和 Redish, 1999)。这种方法可以清楚地看出整个数据中每组数据是如何影响其他各组数据的。三角形的每个顶点对应一种不同类型的数据：

- 问题列表。
- 日志和调查问卷中的数值(定量)数据。
- 测试者的评论和测试团队的观察。

根据可用性目标和测试前确定的数值标准来对数据进行检验，从而确定哪些问题位于三角形内部。

你可能会从数据中发现一些需要进一步研究的现象。举例来说，你可能发现一个用户对几个问题的不同反应。这个用户可能感到完成一个任务要比其他人想的要难得多。因为可用性测试的用户数量一般较小，你应该始终慎重对待这种例外数据。如果这个用户可能代表了一大群的潜在用户，那么这个数据就有可能表明你需要另外计划一次专门针对类似该测试者的用户的可用性测试，从而确定这一结果是否在这群用户中依然存在，还是仅仅是由于某种原因的一个例外。

Dumas 和 Redish(1999) 建议遵循以下原则以保证统计分析尽可能相关：

1. 只有在你理解如何应用和解释推论性数据情况下才使用它们。

2. 当你进行了统计测试后, 仔细解释测试的含义。

3. 当你不计算统计测试时, 要对关键数据进行描述。这个描述会给读者一些关于所谓“眼球”测试准确度的一些引导。

无论是定量数据分析还是来自于反馈和笔记的定性数据都有助于你将信息归纳到以下 2 个方面 (Dumas 和 Redish, 1999):

- **范围**——问题的分布状态如何? 最好将问题分成一般的组, 每组表示测试中得到的特定结果中具有代表性的问题。举例来说, 这个组可以是由于应用程序造成的缺少用户反馈, 那么支持这个组的结果可能包括测试者在执行了某个任务后应用程序没有任何反应的情况。
- **严重性**——问题的严重程度如何? 你可以在分析数据的过程中设定严重性标准, 也可以在测试之前就设定好。无论是你还是报告的阅读者都容易理解制定的标准, 这完全由你决定。举例来说, 你可以使用五分制来描述, 1 分的问题代表该问题会影响某项任务的完成, 需要项目团队密切关注, 而 5 分的问题则表示项目团队可以在产品的下个版本中再考虑解决该问题。

9.4.2 报告

研究不是一门精确的科学, 任何研究都会带有一定的偏差。当你陈述报告时, 一定首先要将下面 2 个问题放在心上 (Kuniavsky, 2003):

- **数据收集问题是什么?** 你应该首先表明数据收集的问题, 这样报告受众中就不会有人指出数据收集中的基本缺陷, 而这些缺陷可能会让你所做努力付之东流。
- **分析的局限性是什么?** 本书讨论了其中一些限制。你应该说明为什么选择部分数据进行分析的原因。

通过首先重视这 2 个问题, 你可以将各种问题减到最少, 不仅可以让数据更清晰, 还能够增加论断的分量。

报告类型很大程度上取决于报告的受众。你可能需要将报告陈述给公司里的多个部门, 所以报告措词可能就要根据不同的受众进行调整 (Dumas 和 Redish, 1999):

- **工程师**——工程师是解决问题的人，所以报告和建议应该集中于解决方案。
- **虚拟设计师**——设计师也是问题解决者之一，所以他们大多数也关注解决方案。
- **市场销售人员**——报告应该集中于人们选择的原因以及用户的心智模型。
- **上层管理者**——管理者的期望是有差异的，所以你必须了解准备评估和批准报告的管理者的议程。你也可以直接跟管理者沟通，简要地讨论什么对他们比较重要。

除此之外，你还要决定不同的受众希望怎样的报告格式。你可能只需要在邮件上附一个 Word 文件发给合适的人，也许你需要在执行董事会议上做报告，那么你就需要将打印的纸质报告分发给董事会成员。如果你想要确认报告格式是否能够被受众接受，你可以给他们发一份草稿看看他们的反馈。

确定了报告格式后，你需要按照对报告受众最有效的方式来组织报告信息。你可能需要根据受众的不同准备多个不同版本的报告 (Dumas 和 Redish, 1999)。举例而言：

- **精简报告**，仅包含必须知道的信息。你可能想通过电子邮件发送这种报告，这样报告信息就会被及时看到。
- **通用报告**，包含了必须知道的信息和应该知道的信息，可以发送给每个人。
- **完整报告**，包括了必须知道的、应该知道的以及最好知道的信息，这样每个人都可以不仅知道哪些需要改进，而且还会知道你收集到的、项目团队可能会完成的一些敏感反馈。

按照报纸样式写报告——也就是说，像报纸故事一样的架构，把最重要的事实放在第一句，而将最不重要的放到最后一句 (Dumas 和 Redish, 1999)。将报告分为多个章节，例如：

- 步骤
- 测试概述
- 观察资料
- 测试引用
- 结论
- 建议

在报告第一页对整个报告的内容进行一个概述，这样不想看完整报告的人也可以得到

报告的基本思想以及建议内容。

9.4.3 陈述

当你准备好陈述报告时，建议你遵循以下准则以保证陈述成功 (Dumas 和 Redish, 1999)：

- 让陈述对象准备好，使他们处于一种有助于理解报告内容的状态中。
- 仔细挑选报告要点。在正式陈述中，时间是有限的，你无法将所有的要点都陈述出来，所以一定要保证陈述必须知道的主题。
- 用实例来证明你的论点，如果你对可用性测试进行了拍摄，则可使用录像带，也可引用测试参与者的姓名和谈话内容。
- 在陈述结果时要强调用户市场的观点，这样才能说明用户的想法与项目团队的想法之间的不同之处。
- 慎用专业术语，只有在前面明确定义后再使用术语，以避免让陈述受众感到混乱。
- 谨慎使用数字。一定要保证陈述中的数字是有实际数据支持的，因为在人们看来数字可以完全代表某种观点。
- 留下陈述的三分之一时间来回答问题。

准备好陈述大纲后要进行必要的练习，最好能在其他人面前进行陈述练习，特别是这个人可以代表报告受众时最佳。

9.4.4 准备简报

你可能由于各种原因不能进行正式陈述。而且，有些人因为太忙从来不会完整阅读报告。多媒体技术的发展不仅让人们可以非常方便地创建视频陈述报告，还可以将其转换成流媒体文件附在电子邮件中，通过公司的内部网络发布出去。例如，我在惠普做承包人时曾经使用一款软件创建了不少流媒体文件，那个软件价值仅 50 美金，这完全在部门经理的预算范围之内。因为经理非常关注产品，所以我们就把这些文件放在了惠普内部网站中我们团队的网页上了。

制作这种简报或流媒体文件也有自己的优缺点 (Dumas 和 Redish, 1999)：

- 这种简报或流媒体可以为观看者提供他们想要了解的关于测试的所有数据。但是，简报的时间只有 20 ~ 30 分钟，那么观看者就有可能看不到报告中所有的例外和警告。而且，观看者也无法实时地提出问题，尽管你可以邀请他们通过电话或者电子邮件跟你联系。
- 报告通过图像和声音传达给人们的印象要比通过阅读来的更生动也更持久。尽管如此，受众还是期望看到精炼的视频片段，这种视频陈述简报可能会很快让他们感到枯燥。
- 视频简报可能是你向重要的项目关系人做报告陈述的唯一方式。当然，你无法保证这些受众确实观看了视频简报，而在正式陈述中的情况则完全不同。

在创建视频的时候，按照下列标准写下计划：

- **范围**——阐明对改善现有设计而言最紧要的问题和最重要的建议，解说要尽量简明扼要。
- **目的**——决定你制作视频的目的。
- **受众**——哪些人将观看视频？
- **他们会根据信息做些什么**——你想让受众根据从视频中获得信息做些什么？
- **约束**——视频会持续多长时间？视频中需要什么添加什么内容？时间期限是什么？

需要指出的是，为了制作视频可能需要购买一些硬件和软件，这可能成为另一个障碍。如果你已经为可用性测试录制了视频，你也有视频录制和音频录制设备，那么问题就在于公司是否有制作视频编辑的硬件和软件。

9.4.5 更改产品和过程

当你把信息传达给项目团队后，你怎样才能把信息转化为行动从而改善产品和过程的可用性呢？记住下面 3 点，可以更有利于你帮助管理人员、开发人员及其他项目关系人，特别是那些对更改持反对态度的项目关系人：

- 在项目过程中保持沟通顺畅。从项目伊始就与所有的项目关系人一起工作。如果你能够用开放和包容的心态对待他们，他们会很乐意参与进来。
- 帮助项目关系人整理必需的变更。将报告中的问题和建议整理成容易发现和使用的

样式，要说服项目关系人接受建议并付诸行动是需要很多努力的。

- 在给出变更建议时尽可能现实一点。时间和资金限制可能会要求你改变解决问题的方法。但是，不要听到不行这个词就不努力了，继续努力，让人们知道如果不做这些改变结果会怎样。

这3个指导原则同样适用于设计过程的改变，因为可用性测试能够很充分地暴露出会在测试中导致可用性问题的过程缺陷。如果你发现必须改变设计过程，那么你的角色已经不仅是测试产品了，你现在成了整个组织的改进代理人。当你为公司改变设计过程时，你不仅在帮助改善一个产品的可用性，而且是在帮助改善公司未来所有产品的可用性。

9.5 案例研究：实现纸上原型测试

你已经与测试人员进行了访谈，也检查了现有的应用程序。现在到了创建纸上原型测试草案的时候了，将测试材料装进活页夹，然后让 Evan 执行预备性测试。

在这个预备性测试中，Evan 将像实际测试一样把测试交给你、一个观察员和一个记录员。你招募的其他观察员和记录员都将关注预备性测试过程，并将在预备性测试一结束立即召开的报告讨论会上给出他们的反馈信息。

该预备性测试原本计划是持续 65 分钟，其中包括测试开始的一个 10 分钟的旨在说明测试过程中的主要人物角色的测试简介，以及接下来的一个 10 分钟的问答。根据你收集到的信息，你将要为项目团队测试以下 9 个任务：

- 使用搜索框搜索一个产品或者客户。
- 在适当的页面中查看产品或者客户信息。
- 从产品/客户信息页面跳转(导航)至其他页面。
- 在产品有效性页面显示零件供应信息。
- 在产品有效性页面使用在线帮助。
- 在零件维护网页查看零件无效警告。
- 在零件维护网页查看有缺陷零件警告。
- 通过应用程序打开零件供应网站。
- 在应用程序中通过用户的 VoIP 电话给零件供应商打电话。

每个任务测试不会超过5分钟，所以总共会占用剩下的45分钟的测试时间。Evan 掌握测试活页夹(包括任务表单)，你和其他2个预备性测试人员围坐在圆桌旁进行测试。Ann 和 Sam 是观察员，Debbie、Jim 和 Robyn 是记录员。Ann 和 Jim 同时也作为预备性测试的测试人员。

记录员和观察员位于测试房间的角落，不影响 Evan 和其他测试人员，参见图 9-1。

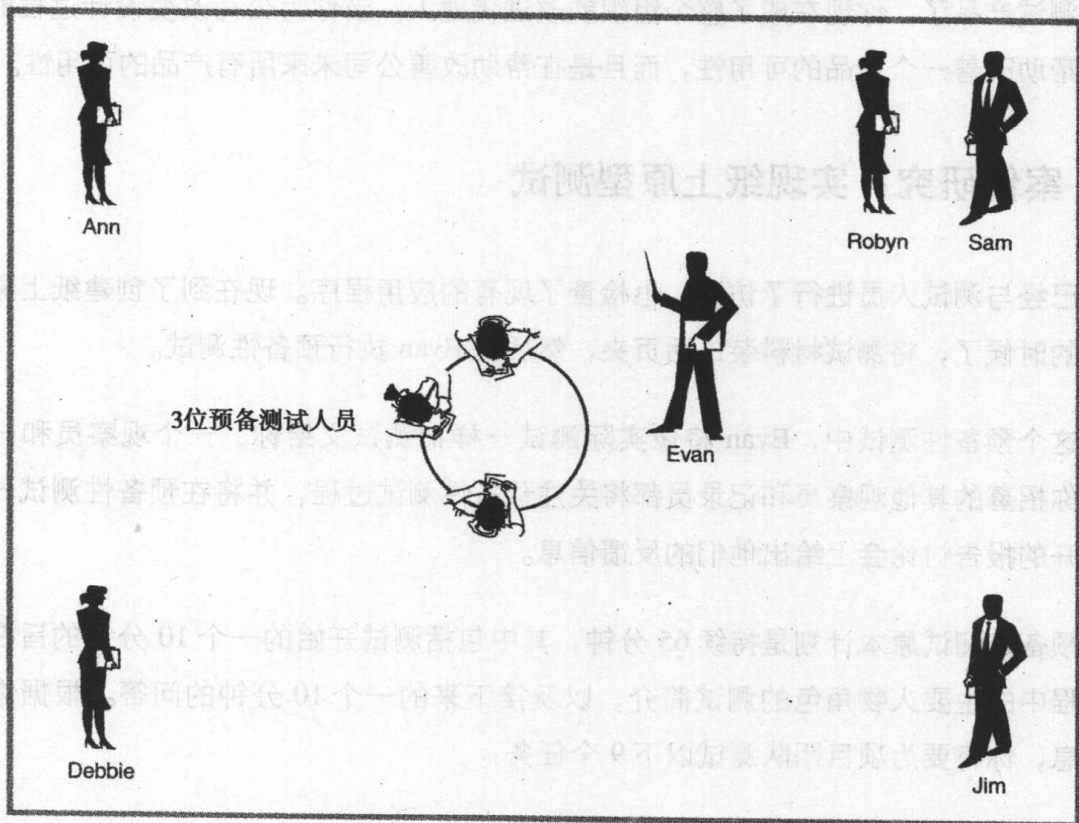


图 9-1 “计算机”和观察员的位置示意图

当测试开始时，Evan 根据用户反馈说明主要测试人物角色，这样测试人员就会明白纸上原型测试选项出自何处。他将给出一些关于测试的简要说明：

- 测试人员必须展示，不能说。
- Evan 明确了可以在原型上写，但不能打字键入，不能点击。
- Evan 代表计算机。为了确保计算机正确地响应，可以找 2~3 个大学实习生作为记录员或观察员。让他们观察并记录用户与计算机(Evan)之间的交互。这些记录会成为任务报告的一部分，在测试完成后你、Evan 和其他记录员、观察员都会完整

查看记录。

- 如果出现了没有预料到的用户行为，Evan 会记录下来，甚至临时仿制一张新的屏幕显示卡片进行响应。
- 如果纸上原型失效，那我们就会休息一下。

预备性测试完成后，你、Evan 以及其他记录员和观察员会在一起分享有关本次测试的信息和看法。

你：“我想最后说，想先听听大家对测试的看法。”

Ann：“我觉得整个测试进行得还算不错，但是 Evan 显得太匆忙了点，我们直到预计测试结束前 2 分钟才完成最后的测试。”

Jim：“我同意 Ann 的说法，我觉得我们应该给测试再增加 30 分钟时间，这样 Evan 才能有足够的时间来完成测试。”

Evan：“你们不认为时间应该延长 15 分钟吗？”

Ann：“我觉得 30 分钟比较好，特别是因为我们关于使用产品有效性页面的讨论持续了太长时间。如果能够改变这个页面，特别是加入更多的信息，我想可以缓解这个问题。”

Robyn：“特别是如果 Evan 还需要回答问题。我觉得他在测试过程中表现得很好。”

Debbie：“Evan，你给出跟我们现在的应用程序一样的声音提示，这点做得很好，Ann 在测试过程记录下来。这有助于测试人员统一行动，并能及时知道在某一时刻用户做对还是做错。”

Evan：“这是一次很棒的测试。我认为我们可以根据 Ann 的建议对界面进行一些调整，然后进行实际测试。”

Sam：“再额外多给 30 分钟时间应该可以发现测试中的所有问题，特别是需要问答时间来做提议的情况下。”

你：“我同意。我想我们需要一些粘贴纸条让测试者写下他们的建议，然后我们再收

集观察员和测试人员的建议，一起为下一步的测试做一个计划清单。”

Evan：“我要跟 Mike 说一下，申请把测试延长至 95 分钟，然后再进行一次纸上原型测试。”

你：“一定要记得告诉他，根据第二次纸上原型测试的情况，我们可能还要进行第三次纸上原型测试。然后在应用程序草案准备好后，我们还需要进行一次可用性测试。请跟 Mike 一起讨论时间安排事宜，如果他有什么问题，让他直接与我联系。”

与预备性测试团队的沟通是 6 个星期之前完成的。你和 Evan 与整个团队一起检查了纸上原型测试和接下来的原型测试。可喜的是，这些测试都在一个工作周内完成了，而且开发人员根据测试团队的建议对应用程序草案进行的升级也在 12 个工作日内完成了。

你和 Evan 对项目团队的所有成员进行了访谈，了解了他们在测试地点使用草案升级后的应用程序的情况。他们给出了一些有用的信息，你把这些信息汇总并报告给了 Mike。报告中包含了经 Mike 批准的更改建议。程序员很快做了修改，Mike 也对应用程序的最后修改进行了及时确认。

2 天前，所作的更改上线了。今天早些时候你走访了 Mike 的公司，告诉了他用户对系统变更的反应。

你打开门，看到 Mike 和 Traci 微笑着坐在 Mike 的桌子旁。“反响好的让人难以置信！”Mike 大声说道。

“我们已经希望超过我们预计的投资回报率了。”Traci 补充道。

你报以微笑，看到 Mike 这么高兴你也放心了：“你喜欢我很高兴。你还有什么其他需要么？”

Mike 将双手放到桌上：“我很高兴你问我这个问题，”他说，“现在是时候让我们的客户测试我们新的与数据库应用程序绑定的网站了。你什么时候可以开始可用性测试呢？”

9.6 小结

本章开始讨论了选择可用性测试方法的问题，以及有效的几种不同类型的测试。并没有一个最好的可用性测试，你必须自己决定最适合你实际情况的可用性测试。本章讨论的最主要的可用性测试是观察、聆听并引导用户。应用这种类型的可用性测试，你会碰到一系列问题。本章对这些问题进行了讨论并给出了答案。

随后本章讨论了定义可用性测试。你学到了有关测试的定义目标和测量方法，选择测试参与者，创建并挑选测试情境，决定如何测量可用性，以及准备测试材料等内容。当你创建测试情境时，它们一定要短小精悍、意思明确，并能给参与者足够的信息来完成任务。当然，测试情境应该以用户的语言描述，而且应该跟任务和关注点直接相关。

接下来是执行可用性测试的部分。你学到了在进行真实测试之前要进行一次预备性测试，锻炼测试技巧，要关心测试参与者。你学到了必须通过一次预备性测试来发现可能会与实际用户参与的真实测试过程中出现的问题。在进行真实测试的时候，你学到了你应该尽可能地鼓励测试人员把所想的大声说出来，以及在测试出现偏差时应该怎么处理。

本章最后讨论了如何分析可用性测试结果的问题，如何陈述测试结果，以及通过何种方式你能将结果陈述给特定受众。正如可用性测试一样，向人们陈述测试结果也没有一种最佳方式，应该根据实际情况选择合适的方式。本章讨论了准备一次正式的介绍你需要做些什么，以及如果你不能做正式介绍那么如何做概要介绍。提供了信息后，你需要把信息转化成行动，这样你才能不仅改善产品的可用性，而且还能够将你从用户那里所获得信息应用到公司的其他产品设计中去。

复习题

现在该回顾一下你在本章学到的知识了。请思考并回答下面的问题，参考答案在附录A中。

1. 促使人们使用某个产品的3个共通目标是什么？

2. UEL 的 3 个阶段分别是什么?
3. 在规划一个可用性测试过程中应该坚持哪些原则?
4. 你应该测试哪些类型情境?
5. 为什么要进行预备性测试?
6. 为什么说进行工作现场走访是有作用的?
7. 进行一次有用且有效的预备性测试需要多少用户?
8. 测试结果中可能的偏差来源是什么?
9. 你应该如何表述测试报告中的偏差问题?
10. 怎样才能确保项目团队实现你建议的更改?



附录 A

复习题答案

如果读者想检查一下自己回答本书中每章后的复习题的情况，可以在这里找到正确答案。

第 1 章

1. 为什么说了解用户界面的历史沿革非常重要？

因为你需要知道图形用户界面是如何逐步成为绝大多数计算机相关操作中所必需的标准用户界面的。

2. 谁开发出了第一个图形用户界面？

Douglas Engelbart。

3. 为什么 NLS 如此重要？

因为它是第一个带有图形用户界面的操作系统。

4. 第一个带有图形用户界面的个人计算机是什么？

Xerox Alto 计算机。

5. 为什么 Linux 这么流行？

因为 Linux 被认为是非常稳定的操作系统。

6. 第一个带有图形用户界面的网页浏览器是什么？

Mosaic 网页浏览器。

7. 为什么 IE 浏览器对设计影响如此巨大？

因为 IE 浏览器的市场份额非常大，而且微软暂停了对 IE 的开发，从而给其他公司应用创新技术开发浏览器的机会。

8. 作为一个用户界面设计者，为什么需要知道用户界面的工作原理？

尽管各种图形用户界面具有很强的相似性，但是它们之间还是有一些微妙的差别，需要在进行用户界面设计时注意。

9. 为什么需要了解不同图形用户界面之间的不同点？

因为不同的图形用户界面在感观上和所提供的界面特性上都有所不同，比如 Windows 中的任务栏。

10. 为什么需要知道图形用户界面和网站界面之间的区别？

因为你必须理解用户界面是如何在你所运行的操作系统中发挥作用的。

第 2 章

1. GUI 的定义是什么？

图形用户界面是一个通过操作图形元素和文本与计算机进行交互的系统。

2. 当今 3 种主流的 GUI 是什么？

3 个图形用户界面操作系统包括 Windows、Mac OS 和 Linux。

3. GUI 的主要的组成部分是什么？

包括窗口、按钮、菜单、工具条、滚动条和任务栏等。

4. 什么是因特网？

因特网是面向公众开发的世界范围的计算机互联网络系统。

5. 什么是万维网?

万维网,即我们所熟知的缩写 WWW,是一个基于因特网的可以通过网页浏览器进行查看的信息分享空间。

6. 3个主要的用户界面模型是什么?

批处理界面、命令行界面和 GUI。

7. 为什么说网站设计仍然具有很大的挑战性?

因为目前的网页浏览器仍然有很多限制。

8. 为什么网页设计会对用户界面设计产生影响?

因为很多用户都浏览网站,操作系统厂商都不得不考虑网站可及性问题。

9. 什么是 ZUI?

即所谓的缩放用户界面(Zooming User Interface)。

10. Dumas 和 Redish 是如何定义可用性的?

使用产品的用户可以快捷地完成他们的任务。

11. 本章中讨论的 3 类可用性相关岗位或职业分别是什么?

本章中讨论的 3 类可用性相关岗位或职业分别是:可用性工程师、可用性科学家和用户体验专家。

12. 什么时候进行快速粗略可用性测试?

通常在产品生产出来后。

13. 什么时候进行总结性可用性测试?

通常在产品开发到一定阶段时,测试人员需要确定产品开发进度如何。

14. 为什么用户界面方法的有效性和可靠性必须经过检验?

因为目前的设计过程、数据分析和报告还没什么标准。

15. 可及性为什么重要？

因为你需要为所有的潜在用户或网民设计用户界面。

16. 在美国，什么法令对可及性需求进行了规定？

1973 年的康复法案的修订，第 508 条款。

第 3 章

1. 为什么说用户目标的满意度非常重要？

因为用户目标的满意度通常意味着商业目标的满意度。

2. 项目中有哪 4 种项目关系人？

分别是用户、设计师、销售和市场人员、产品经理。

3. 好的设计能带来什么益处？

好的设计可以在生产成本、客户服务成本、客户生产力和客户保留方面带来益处。

4. 为什么在项目设计之初要同时着手可用性设计过程？

在项目设计之初就同时着手可用性设计过程可以让你能够按照完全用户体验进行可用性测试、产品界面和产品文档的设计。

5. 启动商业案例之前应该首先讨论什么主题？

优良设计的好处。

6. 如何保证客户目标与用户体验相一致？

了解用户的需求、任务和目标。

7. 应该在什么时候给产品添加特性和功能？

当这些功能对用户的价值和对公司的价值相一致时。

8. 向项目关系人展示良好设计和可用性设计测试能够降低成本之后，还需要向他们证明什么？

在向项目关系人说明了优良设计和可用性设计及测试如何能降低成本之后，你还需要向他们证明这些设计和测试可以为公司带来效益。

9. 为什么要研究投资回报率？

为了计算出一款产品多久可以收回成本。

10. 为什么要使用可用性工程生命周期？

因为它提供了开发用户界面和文档的严格过程说明。

11. 可用性工程生命周期分为哪 3 个阶段？

需求分析；设计、测试和开发；安装。

12. 为什么必须在开发过程中收集用户反馈信息？

用户反馈可以确保产品一旦发布后没有什么需要修改的问题。

第 4 章

1. 为什么在开始用户界面设计过程之前要解决好各种冲突和限制？

因为只有这样才能设计出能够更好服务用户的界面。

2. 为什么用户界面应是优雅的？

因为所有的界面组成部分工作时必须是一个有机整体。

3. 如何弥补用户和设计师之间的限制差异？

在设计过程之初让他们与开发团队进行面对面沟通。如果这样不行的话，就尽可能多地获得用户的反馈，并将这些反馈意见传达给设计师，帮助他们更好地设计其他产品。

4. 为什么要使用纸上原型设计?

这样能够在开发产品之前了解设计意图是否与用户交互时的实际情况相符。

5. 怎样做才能让纸上原型设计过程看起来更专业?

可以使用更加结实的纸版或卡片，这样在进行原型设计测试时更加耐用。

6. 纸上原型设计的优点是什么?

纸上原型设计可以得到真实用户的反馈，不需要特殊技巧，便于沟通，激发创造力，并能减少误解。

7. 纸上原型设计的缺点是什么?

纸上原型设计可能不适合你的实际情况，一些诸如键盘和鼠标输入等技术问题无法通过纸上原型设计测试的方法发现。

8. 为什么一个产品需要良好的文档?

良好的产品文档对公司而言是客户支持服务的第一道防线。

9. 为什么良好的文档设计非常重要?

因为当用户在使用产品中遇到问题首先想到的是求助于产品文档，如果文档能够帮助用户解决问题，那么就能为公司节省客户支持服务经费。

10. 为什么良好的设计这么重要?

良好的设计能够节省资金，说服用户购买产品，有利于维持现有客户，并能带来新的用户。

第5章

1. 什么是预设用途?

预设用途是指事物能够被感知的真实属性。

2. 什么是约束?

约束是由于某些原因给允许的操作带来的限制。

3. 为什么人们在操作失败时会感觉到无助?

因为他们会埋怨自己或归咎于错误的原因, 坚持错误的观念, 或者习惯性无助。

4. MBTI 测试是做什么用的?

MBTI 测试帮助人们识别和理解他们的个性类型偏好。

5. 人类行为的 7 个阶段分别是什么?

形成目标, 形成意图, 设定行为, 执行行为, 感知世界状态, 解释世界状态, 评估结果。

6. 大脑和外界中的知识之间的平衡点是什么?

只要可看见的和可听见的事物在视域或者听觉范围之内, 那么你就能很容易地从现实世界中获取知识。你可能会因为分心的事物而难于记住一些知识, 但是一旦记住了, 你就会很有效掌握了。

7. 什么样的任务结构对人们而言是最有挑战性的?

宽广结构。

8. 为什么在使用有意识的想法时你必须要慎重?

因为在这种情况下利用的是短期记忆。

9. 为什么要把较难的任务划分成多个简单的任务?

这样才能尽可能地方便你进行思考。

10. 人们的概念模型是靠什么建立起来的?

人们的生活经验、信仰, 以及他多年来完成任务的其他方法。

第6章

1. 为什么设计师仍然坚持机械时代的标准?

因为那是设计师们最熟悉的标准。

2. 哪些人是永久中级用户?

那些对软件应用程序或者产品有足够经验并且知道特定问题答案的用户。

3. 初级用户总会问的问题有哪些?

诸如“这个程序是干什么用的?”、“我应该从哪里开始?”以及“我应该怎样做才能完成这个任务?”等问题。

4. 中级用户总会问的问题有哪些?

诸如“我怎么才能找到这个功能?”、“我怎么取消上一步操作?”以及“执行这个任务的操作指令是什么?”等问题。

5. “目标导向设计过程”的5个阶段分别是什么?

研究、建模、需求、框架和精炼。

6. 为什么要进行用户和任务分析?

因为需要得到关于用户及其执行任务操等问题的答案。

7. 信息与角色关系的3个维度分别是什么?

人口统计、心理和拓扑。

8. 用户目标有哪些类型?

生活目标、经验目标以及使用具体产品的最终目标。

9. 为什么应该“现场”进行用户和任务分析?

因为你需要了解用户是如何与大环境进行交互的，包括与其他人以及客观环境的交互等。

10. 为什么需要对人物角色进行分级？

因为你需要了解产品的主要人物角色和二级人物角色，主要角色是用户界面的最基本的目标用户。

第 7 章

1. 为什么需要对真实世界需求进行规划？

因为从其他项目关系人那里获得需求有助于你对在需求分析过程中产生的想法进行提炼。

2. 为什么说纸上原型设计和故事板在设定关键路径情景时非常重要？

因为你可以应用它们表现用户完成任务的每一个交互过程路径。

3. 能够引导你做出最便于用户使用的设计的 3 级设计原则是什么？

分别是概念设计，交互设计和界面设计。

4. 为什么说创建模式非常重要？

因为你可以在项目中使用这些模式，从而更快更好地解决项目中的问题。

5. 4 种基于桌面的 GUI 姿态分别是什么？

四种基于桌面的 GUI 姿态分别是独占型、暂态型、精灵型和辅助型。

6. 辅助应用程序的特点是什么？

独占的和暂态的。

7. 在目标上点击鼠标右键会发生什么？

会弹出一个对象相关的可选择菜单。

8. 为什么必须避免视觉上的噪声和混乱?

因为提供过多的信息会让用户感到不安和混乱。

9. 为什么说拥有一个设计优良的在线帮助系统很重要?

因为永久中级用户想尽快地找到所需要的东西，而且在线帮助系统可以成为客户支持的第一线。

10. 图标上的弹出式菜单有什么优点?

弹出菜单可以包含更多关于该图标的特定信息。

11. 界面设计中一致性和标准的使用对于用户有什么好处?

这样能够确保用户能够找到他们所需要的信息，因为这些信息会保存在同一个地方。

12. 什么时候需要使用助手程序和向导程序?

当你的用户和任务分析表明助手程序和向导程序作为用户界面的一部分能够让你的用户收益。

13. 为什么说必须进行情景验证?

因为并不是所有的人都以同样的方式使用界面。

14. 如何向公司其他项目关系人展示最终设计?

通过创建交互式原型或者在线演示。

第 8 章

1. 图形用户界面和网站界面之间的相似之处表现在哪些方面?

网站界面运行于一个浏览器窗口之中，所以，它会应用到许多 GUI 特性，比如通过鼠标箭头与网站网页进行交互等。

2. GUI 和网站界面的相似之处是越来越多还是越少? 为什么?

随着 GUI 引入了更多网站界面元素, 同时网站技术的发展使得网站界面的行为更像那些桌面应用程序, 两者之间的界限越来越模糊了。

3. 为什么你需要了解关于网站的一些迷思?

因为你必须在设计你的网站时避免这些问题。

4. 网站迷思分为哪 3 类?

这些误区分为用法、设计和可及性 3 类。

5. 为什么说了解网站姿态非常重要?

因为你需要根据用户使用网站的方式来设计你的网站。

6. 网站的 3 种不同类型分别是什么?

3 种不同类型的网站分别是信息类、应用类和门户类。

7. 你可以创建的 3 种类型网站是哪些?

信息网站、应用网站和门户网站。

8. 一个内容驱动处理的例子是什么?

网站表格。

9. 一个处理驱动内容的例子是什么?

用户在网站上提交了表格并且系统处理完毕后弹出一个感谢提示。

10. 为什么你必须限制颜色和文字选择?

因为你要确保所有的计算机上的所有浏览器上看到的颜色和文字都是原样不变的。

11. 滥用图形的 4 个表现特征是什么?

分别是闪动文字或闪动一块文字, 混乱的动画图形, 网页一个区域中包含过多的图形, 以及分辨率过低导致的锯齿状图形。

12. 为什么必须在任何可能的时候都要坚持网站设计的 4 个原则？

因为这些原则会让你的网站尽可能好用。

13. 什么是 3 次点击原则？

用户最多通过三次点击就能在网站中找到想要的内容。

14. 什么情况下可以打破网站设计原则？

当这些原则不适合你的用户和公司需求的时候。

第 9 章

1. 促使人们使用某个产品的 3 个共同目标是什么？

用户能够很快上手，产品能够解决用户的需要，以及很容易获得帮助。

2. UEL 的 3 个阶段分别是什么？

需求分析；设计、测试和开发；安装。

3. 在规划一个可用性测试过程中应该坚持哪些原则？

事先计划；在用户群中选择具有差异性的用户；将用户作为伙伴对待；观察，聆听并与用户交谈；让谈话尽量具体；从你的用户那里获得线索。

4. 你应该测试哪些类型情境？

存在潜在可用性问题的任务，你所关心的和经验上需要测试的任务，源自用户标准的任务，以及那些用户想要进行的任务。

5. 为什么要进行预备性测试？

预备性测试可以“调试”你的可用性测试方案，找出产品、网站、文档、测试方法和测试材料中的问题。

6. 为什么说进行工作现场走访是有作用的？

可以看到用户是如何工作的，以及用户是如何在他们的工作环境中使用你的产品的。

7. 进行一次有用且有效的预备性测试需要多少用户？

找一个能够代表你想测试的用户对象的参与者即可。

8. 测试结果中可能的偏差来源是什么？

数据收集问题和分析本身的限制。

9. 你应该如何表述测试报告中的偏差问题？

要首先承认数据收集和分析限制中存在的问题。

10. 怎样才能确保项目团队实现你建议的更改？

保持项目过程沟通顺畅；帮助你的项目关系人组织必须的更改；对更改的提议一定要确实是可行的。





推荐读物

书籍

以下这些书籍包括了用户界面设计、交互设计和可用性设计领域中最重要且实用的最新研究成果。如果你对本书讨论的某些主题有更深入的兴趣，可以在本地书店中购买若干以下的书籍。

- 《A Practical Guide to Usability Testing》，作者 Joseph S. Dumas and Janice C. Reish, ISBN 1-84150-020-8
- 《About Face 2.0》，作者 Alan Cooper and Robert Reimann, ISBN 0-7645-2641-3
- 《Built for Use》，作者 Karen Donoghue, ISBN 0-07-138304-2
- 《Constructing Accessible Web Sites》，作者 Jim Thatcher et al. , ISBN 1-904151-00-0
- 《Cost-Justifying Usability, Second Edition》，作者 Randolph G. Bias and Deborah J. Mayhew, ISBN 0-12-095811-2
- 《Observing the User Experience》，作者 Mike Kuniavsky, ISBN 1-55860-923-7
- 《Paper Prototyping》，作者 Carolyn Snyder, ISBN 1-55860-870-2

-
- 《The Design of Everyday Things》，作者 Donald A. Norman, ISBN 0-465-06710-7
 - 《Usability for the Web》，作者 Tom Brinck et al. , ISBN 1-55860-658-0
 - 《User and Task Analysis for Interface Design》，作者 JoAnn T. Hackos and Janice C. Redish, ISBN 0-471-17831-4
 - 《Waiting for Your Cat to Bark?》，作者 Bryan Eisenberg and Jeffrey Eisenberg, ISBN 0-7852-1897-1



术语表

Active Server Pages (ASP)——一种由微软开发的允许页面包含动态内容的技术。动态内容通常来自于数据库。

Address bar (地址栏)——一个在浏览器中用来输入想要访问的网站网址的地方。

Aero——一个用于 Microsoft Windows Vista 操作系统的图形用户界面风格。

AJAX——参见 Asynchronous JavaScript and XML。

ALT——一个 HTML 标示，当一个浏览器不能打开显示那些图片等非文本元素时提供用来转换的文本。

Alto——由施乐 (Xerox) 公司制造的是第一台带有 GUI 界面的个人电脑。

Aqua——Mac OS X 操作系统图形用户界面。

ARPA NET——美国国防部高级研究计划署 1969 年开发的计算机网络，因特网的前驱。

ASP——参见 Active Server Pages。

assistant——通常内嵌在另一个程序中的程序。一般用于提供提示或者让用户输入关键字以搜索答案。

Asynchronous JavaScript and XML (AJAX)——一种用于开发交互式页面应用程序的技术。

Atlas——一种 Microsoft 开发的工具箱。用于减轻开发 AJAX 程序的工作量。

auxiliary (辅助型)——一种软件姿态。在这种姿态下应用程序同时展现出完全以及暂时状态的特征。

batch interface (批处理界面)——第一种计算机可用界面。这种计算机有一个或者多个界面用来让用户预编程序制成特殊的带有打孔洞的卡片。

bread crumbs (面包屑链接)——用于让用户从子页面返回首页或者更高一级页面的链接。

CLI——参见命令行界面。

command-line interface (命令行界面)——显示一个命令提示让使用者知道计算机已经准备好接受输入。此时，用户可以用键盘输入命令并用 Enter 键提交到程序。

conceptual model (概念模型)——一种个人基于生活经验、信仰和其他他多年形成的方式来思考如何操作一个特定的任务。

daemonic (精灵型)——一种软件姿态。这一姿态的应用程序与使用者没有交互，在后台运行。

DHTML——参见 Dynamic HTML。

dialog box (对话框)——一种较小的窗口。用于让使用者更改设定和做出决定。

Dynamic HTML (DHTML)——一种 HTML 的扩展。可以使页面包含一些小的动画或者动态菜单。

Flash——一种 Adobe 开发的多媒体制作程序。通常用于为网页应用程序创建内容。

GNOME——一种用于 Linux 操作系统的图形用户界面。

GNU——GNU 网络目标模型环境是一个兼容于 UNIX 操作系统并作为 Linux 操作系统基础的程序。

graphical user interface(图形用户界面, GUI)——通过操作图形元素和文本与计算机进行交互的系统。这里的图形元素包括窗口、按钮、菜单和图标。

GUI——参见 graphical user interface。

HTML——参见 Hypertext Markup Language。

HTTP——参见 Hypertext Transfer Protocol。

hyperlink(超链接)——对网络上另一个文件的引用。

Hypertext Markup Language (HTML)——一种用来在万维网上创建超文本文件的程序语言。

Hypertext Transfer Protocol (HTTP)——在万维网上交换文件(包括文本、图形和其他资源)的一组规则。

Internet Explorer——一种由 Microsoft 开发, 使用于 Microsoft Windows 和 Mac OS X 的免费网页浏览器。

Java——一种由 Sun 微系统公司开发的跨平台的、面向对象的程序语言。

JavaScript——一种作为 JAVA 编程语言的子集的计算机语言。用于使网页具备更强的交互性。

K Desktop Environment (KDE)——一种使用于 Linux 操作系统的图形界面。

KDE——参见 K Desktop Environment。

Linux——一种基于 GNU 操作系统的操作系统。

Mac OS——Apple Macintosh 上的操作系统, 当前为第十个版本, 也叫 Mac OS X。

MBTI——参见 Myers-Briggs Type Indicator。

menu(菜单)——一个命令的列表，提供程序命令的选择。

Mosaic——第一个具有图形界面的万维网浏览器。

mouse pointer(鼠标指针)——一个 GUI 图形界面的功能，通常是一个箭头，可以使你直观地操作鼠标。

Myers-Briggs Type Indicator(MBTI)——一种由 Katherine Briggs 和 Isabelle Myers 开发的用于界定及理解人格类型的方法。

.NET——一个由 Microsoft 为网页应用程序快速开发所打造的开发平台。

Netscape Navigator——个人电脑上第一个流行的网络浏览器，现在由 AOL 命名为 Netscape 浏览器，提供给用户免费使用。

NLS——参见 oN-Line System。

online help(在线帮助)——通过计算机软件提供主题、流程或参考信息的系统。

oN-Line System(NLS)——由 Douglas Engelbart 开发的第一个图形用户界面。

Palo Alto Research Center(PARC)——施乐(Xerox)公司的研究中心。

paper prototyping(纸上原型)——创建软件程序、硬件产品或网站的纸上版本的过程。通过它可以在开发产品前就了解使用者如何与此设计互动。

PARC——参见 Palo Alto Research Center。

PDF——参见 Portable Document Format。

personas(人物角色)——基于不同种类用户特征的用户模式。

PHP——PHP Hypertext Preprocessor(超文本预处理器)用于让 Web 开发者创建与数据库交互的动态内容。

Portable Document Format (PDF)——由 Adobe 公司开发的用于共享、显示和打印格式化文本的文件格式。

portal(门户)——向使用者提供公司相关信息的站点，并告知使用者如何到达别的页面。

posture(姿态)——产品或界面与使用者关联状态的一种描述方式。

principles(原则)——关于确定行为、格式和内容的设计指引。

return on investment(投资回报率, ROI)——一项开销或投资的未来价值，可能是实际情况也可能是预期。

ROI——参见投资回报率。

scrollbar(滚动条)——在区域内文本或图形因太大而无法显示完全时，可以让你在一个窗口内进行连续观看的一种界面工具。

Search Engine Software——用于从数据库或者因特网上寻找信息的软件。

Section 508——一个对 1973 年修复法案的增订。要求界面开发需要满足对联邦残疾员工具备可及性。

SME——参见学科问题专家。

sovereign(独占型)——一种长时间保持使用者注意力的软件姿态。

subject matter expert(学科问题专家, SME)——对于产品、流程或主题方面具备丰富知识的人。

TCP/IP——参见 Transmission Control Protocol/Internet Protocol。

text user interface(文本用户界面, TUI)——一种使用全屏幕区域执行任务的用户界面。

toolbar(工具栏)——包含图标和其他界面工具的行、列或工具箱。

transient(暂态型)——基于特定任务和需求的,用户偶尔使用的一种应用程序姿态。

Transmission Control Protocol/Internet Protocol (TCP/IP)——因特网数据传输协议的标准。

TUI——参见文本用户界面。

Uniform Resource Locator (URL)——用于描述网页、站点或因特网上其他资源地址的字符序列。

UNIX——贝尔实验室于1969年开发的一种操作系统。

URL——参见 Uniform Resource Locafor。

usability defined by Dumas and Redish (Dumas and Redish (1999) 定义的可用性)——“使用产品者可以快捷轻松地完成他们的任务。”

usability engineer(可用性工程师)——提供可用性服务的人。

usability scientist(可用性科学家)——在可用性研究和开发理论方面接受过正式训练的人。

user experience professional(用户体验专家)——可能是可用性工程师、可用性科学家,也可能是公司、办公室、部门中并非正式可用性工程师或在可用性测试部门工作的人。但是他的确能提供可用性服务。

Web——参见 World Wide Web。

Web browser(网页浏览器)——用于显示网页信息并可以使你与网页进行交互的软件程序。

widget(界面构件)——图形用户使用界面中的一个组件。

window(窗口)——通常形状规整的一片区域,用于显示包括用户文件在内的信息。用户可以打开或者关闭窗口,也可以在桌面范围内拖动它,有时候还可以改变它的尺寸,

滚动它，或者编辑它的内容。

Windows——由 Microsoft 开发并维护的操作系统。当前的最新版本是 Windows Vista。

Windows Vista——发布于 2007 年年初的最新主流 Microsoft Windows 版本。

wizard (向导)——它是引导你逐步增加程序或功能的图形用户界面，例如添加一个家庭网络。

World Wide Web——万维网，因特网上的信息共享空间，可通过网页浏览器访问。

World Wide Web Consortium——万维网联盟，通常被称为 W3C。负责管理网页的标准。

用户界面设计指南

User Interface Design for Mere Mortals

本书给出了为桌面程序和网页程序设计界面效果的灵丹妙药。对于任何想提供给用户直观和易用界面软件的人而言，这本书都是值得阅读的。软件成功的秘诀往往依赖于提供给用户一个乐于使用的界面，同时这一界面可以节省用户时间、降低其挫折感并以最小的投入完成工作。本书的读者将会发现优秀界面设计的秘诀在于，了解用户行为表征（user behave）以及用户对各种界面类型的需求。

本书的读者将有如下收获：

- 获得对各系统和平台下界面的“外观与样式”差异的鉴识能力。
- 学习如何着手为开发中的软件或网站设计并创建最合理的界面。
- 逐渐熟悉用于构建界面的各项组件以及这些组件在与用户交流中所扮演的角色。
- 了解优秀界面设计所带来的商业利益，如显著降低支援成本（support cost）等。
- 掌握极具价值的对用户行为表征的洞察力，包括人机交互的7个步骤。
- 通过个案研究，深入分析用户界面设计过程中的每个步骤。
- 了解网站设计与传统桌面软件设计之间的异同。
- 学习如何定义、引导和分析可用性测试。

本书以轻松幽默的语言以及精确简洁的表达方式，揭开了有效进行用户界面设计这一领域的种种“神秘”之处。本书的写作形式使其内容易懂易用。

Eric Butow是Butow通信公司（BCG）的CEO，这家公司坐落于美国加州的罗斯维尔，主要从事技术出版和网站设计。从2000年以来，Eric著有或与人合著了8本技术著作。同时，Eric是很多计算机图书的技术编辑，并为《SD Times》、《Intercom》等国际性刊物撰写文章。

本书研究细致、写作出色、结构顺畅。我都有些嫉妒，早知道自己来写这本书了。

——《Seeing Data》作者
Rebecca M. Flordan

客户不愿使用的产品乏善可陈。Eric Butow与你分享他在用户界面设计方面的知识，并告诉你如何将真实的用户友好性付诸实践。

——Anthony Barcellos教授，
加利福尼亚大学戴维斯分校及美国大河学院博士



投稿热线：(010) 88379604
购书热线：(010) 68995259, 68995264
读者信箱：hzjsj@hzbook.com

华章网站 <http://www.hzbook.com>

网上购书：www.china-pub.com

封面设计 · 王建敏

上架指导：计算机/Web设计

ISBN 978-7-111-22947-6



9 787111 229476

ISBN 978-7-111-22947-6

定价：36.00 元