

TURING

图灵程序设计丛书 移动开发系列

Apress®

- 《iPhone 3开发基础教程》作者又一力作
- 轻松打造令iPad熠熠生辉的应用程序
- 从iPhone进军iPad，一路凯歌



Beginning iPad Development for iPhone Developers
Mastering the iPad SDK

iPad开发基础教程

[美] Dave Wooldridge 著
David Mark

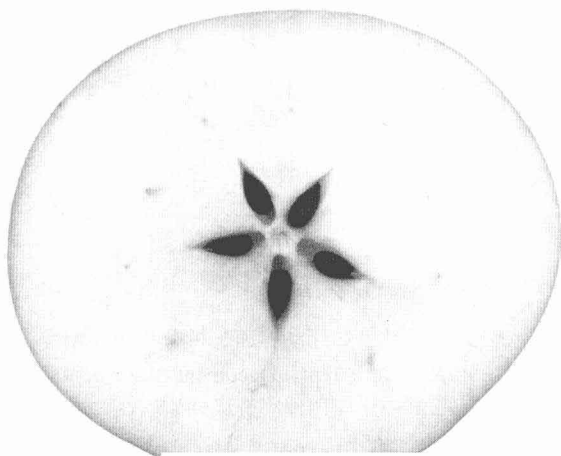
盛海艳 曾少宁 李光杰 等译



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书 移动开发系列



Beginning iPad Development for iPhone Developers

iPad开发基础教程

Jack Nutting

[美] Dave Wooldridge 著

David Mark

盛海艳 曾少宁 李光杰 等译

人民邮电出版社

北京

图书在版编目 (CIP) 数据

iPad开发基础教程 / (美) 纳丁 (Nutting, J.),
(美) 伍尔德里奇 (Wooldridge, D.), (美) 马克
(Mark, D.) 著; 盛海艳等译. — 北京: 人民邮电出版
社, 2011.1

(图灵程序设计丛书)

书名原文: Beginning iPad Development for
iPhone Developers: Mastering the iPad SDK
ISBN 978-7-115-24322-5

I. ①i… II. ①纳… ②伍… ③马… ④盛… III. ①
便携式计算机—程序设计—教材 IV. ①TP368.32

中国版本图书馆CIP数据核字(2010)第221518号

内 容 提 要

本书主要介绍了如何使用 UIKit 新特性(如浮动界面、拆分视图控制器和新的模态视图等)设计 iPad 应用程序界面, 如何使用新的图形功能、Core Text、Document Sharing 和高级输入方法来构建强大的解决方案, 如何显示视频以及如何将已有的 iPhone 应用程序升级到能够在 iPad 上运行。

本书适合所有 iPad 开发人员阅读。

图灵程序设计丛书

iPad开发基础教程

◆ 著 [美] Jack Nutting Dave Wooldridge David Mark
译 盛海艳 曾少宁 李光杰 等
责任编辑 王军花

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京艺辉印刷有限公司印刷

◆ 开本: 800×1000 1/16
印张: 14.25
字数: 331千字 2011年1月第1版
印数: 1-3 500册 2011年1月北京第1次印刷

著作权合同登记号 图字: 01-2010-6602号

ISBN 978-7-115-24322-5

定价: 45.00元

读者服务热线: (010)51095186 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Original English language edition, entitled *Beginning iPad Development for iPhone Developers: Mastering the iPad SDK* by Jack Nutting, Dave Woodridge, David Mark, published by Apress, 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705 USA.

Copyright © 2010 by Jack Nutting, Dave Wooldridge, David Mark. Simplified Chinese-language edition copyright © 2011 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由Apress L.P.授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

献给 Weronica, 感谢她对我的信任。

—— Jack Nutting

献给我优秀的妻子、灵魂之伴侣, Madeline, 感谢她对我无私的爱与支持。

—— Dave Wooldridge

献给我生命中最好的朋友和伙伴, Deneen, LFU4FREIH。

—— David Mark

前言

世界已经改变了。在本书开始编写时，iPad 发布了（上市后 80 天销售了 300 万台），iPhone OS 改名为 iOS，并作为 iOS 4 在新的 iPhone 4 中首次亮相（发布的第一天就有 60 万人预订）。与此同时，有 11 000 多个针对 iPad 的应用程序发布，包括仅用于 iPad 的应用程序和通用应用程序（既能在 iPhone 上运行，也能在 iPad 上运行）。不管从哪方面来看，iPad 都是非常成功的。它激发了众多创新用途，远远超出苹果公司在 1 月份所声称的终极媒体消费设备的范畴。

在本书开始印刷并最终到你手中时，Xcode 4 可能已经推出了，到那时苹果公司可能已经售出了一两百万台 iPad，甚至 iOS 4 可能也已经可以安装到 iPad 上使用了。苹果公司总是让 iPhone 和 iPad 开发人员疲于奔命，我们这些作者也不例外！我们在写作本书时也一直牢记这些，希望我们写的书能经得住时间的考验，无论操作系统版本和发布日期如何变化。

总之，iPhone 和 iPad 实际上是两个不同的东西，尽管其底层操作系统具有相似性，但它们有不同的规格和功能，鼓励不同的使用模式。而本书的目的是要突出 iPhone 和 iPad 的不同之处，帮助你利用新工具和新技巧在 iPhone 开发知识的基础上创建优秀的 iPad 应用程序。

致 谢

在此感谢为本书的出版付出辛勤劳动的 Apress 编辑：Clay Andres，他负责组建本书的作者团队，使本书得以顺利启动；Kelly Moritz，他总是在我们要松懈的时候让我们的弦绷得更紧，使得本书能够按计划顺利完成；Douglas Pundick，他帮助我们每一章串联起来，让全书看起来更连贯；Marilyn Smith，经过她专业的润色，本书的每一页都更有文采。我们还要特别感谢 Mark Dalrymple，他独到敏锐的眼光帮助我们避免了很多技术错误。

目 录

第 1 章 改变思维模式	1
1.1 重塑平板电脑	1
1.1.1 软件就是一切	2
1.1.2 iPad 不是大号的 iPod touch	2
1.2 面向大众的个人计算	4
1.3 为 iPad 开发应用程序	5
1.3.1 当你的 iPhone 应用程序 不再完美	6
1.3.2 探索无限可能	6
1.4 机不可失	13
第 2 章 iPad 开发起步	14
2.1 获取交易工具	14
2.1.1 加入 iPhone Developer Program	14
2.1.2 安装 iPhone SDK 3.2	15
2.1.3 使用 SDK 的 beta 版本	16
2.2 不熟悉 Objective-C 和 Cocoa Touch	17
2.3 接受 MVC 概念	18
2.3.1 Xcode 中的可重用性	19
2.3.2 在 Interface Builder 中设计	21
2.4 委托的重要性	26
2.5 借助 UIKit 改进应用程序可用性	28
2.6 准备编程	30
第 3 章 iPhone SDK 中的新 iPad 功能	31
3.1 为 iPad 优化应用程序	31

3.1.1 了解平板电脑	31
3.1.2 管理内存	32
3.1.3 在 iPad 上测试应用程序	32
3.2 iPhone SDK 3.2 中面向 iPad 的新功能	33
3.2.1 绘制形状	33
3.2.2 PDF 文件	35
3.2.3 Core Text	36
3.2.4 浮动界面	37
3.2.5 视频回放和显示选项	39
3.2.6 拆分视图控制器	41
3.2.7 模态呈现样式	43
3.2.8 高级输入法	44
3.2.9 Document Support	47
3.2.10 通用应用程序	49
3.3 iPad 应用程序需要的项目图像	51
3.3.1 应用程序图标图像	51
3.3.2 文档类型图标图像	52
3.3.3 默认启动图像	52
3.4 小结	53
第 4 章 新图形功能	54
4.1 贝塞尔路径	54
4.2 Duple 简介	55
4.2.1 创建 Duple 项目	55
4.2.2 添加一个简单的 GUI	58
4.3 基本绘图架构	59

4.4 我们需要的是工具	61	6.5.2 连接栅格	125
4.4.1 铅笔工具	64	6.5.3 处理两个操作	127
4.4.2 直线工具	70	6.6 强大的浮动界面	130
4.4.3 椭圆和矩形工具	73	第7章 视频及显示输出	131
4.4.4 手绘工具	77	7.1 显示多个视频	131
4.5 生成 PDF	81	7.1.1 创建视频应用程序项目	132
4.6 小结	83	7.1.2 指定视频文件	132
第5章 使用 Core Text	84	7.1.3 使用 videoCell 输出口 加载 GUI	133
5.1 为什么使用 Core Text	84	7.1.4 创建 VideoCell 用户界面	135
5.2 Core Text 的结构	85	7.1.5 运行视频应用程序	136
5.3 准备 Dudel 的新工具	87	7.2 输出到外部屏幕	137
5.3.1 准备控制器接口	87	7.2.1 扩展视频应用程序以支 持外部屏幕	137
5.3.2 设置 GUI	88	7.2.2 实现 VideoCell 委托方法	140
5.3.3 修改控制器类	89	7.2.3 测试外部屏幕功能	141
5.3.4 创建文本工具	90	7.3 显示解决方案	142
5.3.5 创建新的 Drawable 类	96	第8章 拆分视图和模态模式	143
5.3.6 呈现多种样式	99	8.1 拆分视图的概念	143
5.4 小结	101	8.2 保存和加载的基础知识	145
第6章 浮动界面	102	8.3 文档管理	147
6.1 准备工作	102	8.3.1 列出文件	148
6.1.1 基本的 GUI	103	8.3.2 创建和删除文件	157
6.1.2 注意事项	105	8.3.3 重命名文件	161
6.2 设置字体	107	8.4 以模态方式实现 About 面板	164
6.2.1 最简单的浮动界面	107	8.4.1 创建模态网页视图控制器	164
6.2.2 后台代码	111	8.4.2 显示网页	166
6.3 设置字号	112	8.4.3 与 Dudel 视图控制器集成	167
6.3.1 创建 GUI	114	8.5 小结	168
6.3.2 运行新界面	115	第9章 新的输入方法	169
6.4 笔画宽度	117	9.1 添加菜单项	169
6.4.1 准备工作	117	9.2 手势识别	171
6.4.2 创建 GUI	117	9.2.1 向 Dudel 添加撤销功能	172
6.4.3 使用自定义视图预览笔画宽度	118	9.2.2 重新设置选中工具的状态	174
6.4.4 实现控制器	119	9.3 键盘扩展和替换	174
6.4.5 运行新界面	120		
6.5 双重作用的颜色设置界面	122		
6.5.1 创建简单的颜色栅格	122		

9.3.1 在 Dudel 中添加键盘按钮	174	11.2.3 配置主体 iPad GUI	203
9.3.2 替换键盘	175	11.2.4 为 iPad 创建选项视图控制器	205
9.4 小结	183	11.2.5 实现拆分视图委托方法	205
第 10 章 处理文档	184	11.2.6 调整导航逻辑	206
10.1 向另一个应用程序传递文件	184	11.2.7 改善未选中状态的主视图	207
10.1.1 添加 PDF 发送菜单操作	185	11.3 在多种目标上运行	208
10.1.2 准备一个用于发送的文件	186	11.4 玩转电子设备	209
10.1.3 调用文档交互控制器	187	第 12 章 用于 iPad 开发的其他资源	210
10.2 接收文件	188	12.1 登录到主站点	210
10.2.1 注册为接收端	188	12.1.1 iPad 开发	210
10.2.2 测试文件接收功能	190	12.1.2 Objective-C 和 Cocoa Touch	211
10.2.3 从启动选项中检索文件信息	191	12.1.3 iPad 应用程序部署	211
10.2.4 发送 Dudeldoc 文件	192	12.2 向专家学习	211
10.3 桌面同步	193	12.2.1 图书	212
10.4 共享以及类似共享的功能	194	12.2.2 教程和代码示例	212
第 11 章 从 iPhone 到 iPad	195	12.3 设计 iPad 应用程序的用户界面	213
11.1 为 iPhone 引入 NavApp	195	12.3.1 纸面原型	213
11.1.1 创建 NavApp 项目	196	12.3.2 数字线框图	213
11.1.2 选择 Active SDK	200	12.3.3 用户界面图标	214
11.2 向项目中添加 iPad 支持	200	12.3.4 设计思考和灵感	214
11.2.1 运行升级后的 NavApp	201	12.4 从在线论坛中寻找答案	214
11.2.2 条件行为：了解术语	202	12.5 把握未来	215



除非你生活在原始部落，否则一定会对苹果公司产品线中的新宠 iPad 有所耳闻，这款薄薄的触摸屏平板电脑旨在为移动计算和便携式媒体消费体验带来一场革命。iPad 无疑是最近传闻最多、炒作最厉害、人们预期也最高的产品，至少是自 2007 年苹果公司推出最初的 iPhone 以来最受人们关注的产品。

二者间的一大区别就是首个 iPhone 型号不包括 App Store。直到一年多以后，苹果公司才推出了 iTunes App Store，其中只有 500 个本机应用程序来自第三方开发人员。而到了 2010 年，App Store 中的应用程序超过了 200 000 个。而大部分现有应用程序无需进行任何修改就可在 iPad 上运行，因此，用户一启动他们崭新的 iPad，立刻就能从庞大的软件库中挑选心仪的软件。但是与本机 iPad 应用程序的唯美与灵活性相比，iPhone 应用程序就显得暗淡无光了。

有了全新的专属于 iPad 的特性以及 iPhone SDK 3.2 中提供的用户界面元素，再加上后台提供的强大的图形和处理引擎，iPad 带给开发人员的商机大大超过了早期的 iPhone。与 iPhone 有限的内存和小小的屏幕不同，iPad 为开发人员提供了独一无二的移动平台，可供其开发真正完美的、具有桌面质量的应用程序！

但是要充分利用这个令人兴奋的新机会，开发消费者想要的应用程序，就要了解 iPad 是面向谁设计的。

1.1 重塑平板电脑

为什么是平板电脑呢？要在笔记本电脑和智能手机之间打造一个新的产品类别，该设备必须满足这两种产品目前所无法满足的需求。在过去几年，上网本试图弥补这一空白。但是正如 Steve Jobs 的著名评论所述，上网本无非就是体积较小的廉价笔记本电脑。在运行 Microsoft Windows 或 Linux 操作系统时，与笔记本电脑相比，上网本并未使计算变得更轻松。它们唯一的优势就是价格相对较低。

很薄的平板设备比笔记本电脑或上网本更具亲和力，几乎可以从任何角度手持使用——比如在沙发上或在床上。而且我敢打赌有些人甚至会将 iPad 拿到浴室，享受不多的休闲时光，在网上冲浪或者看看电子书。

iPad 显然不是第一个冲击市场的平板电脑，而且也不会是最后一个。自 20 世纪 90 年代初以

来,无数公司试图用功能丰富的平板电脑吸引消费者,但是无一获得成功。原因何在?因为软件功能有限或者软件很难使用。

1993年,苹果公司推出了 Newton MessagePad,这是苹果公司在平板电脑方面的首次尝试。由于其单色的屏幕和有限的软件选择,它更多地被看做一个大号的 PDA,而不是真正的移动计算机。因为该产品的吸引力似乎从未超越小范围但是忠诚如信徒般的客户群,所以苹果公司在 1998 年中断了 Newton 的开发。

其他很多硬件公司随后纷纷尝试用不同的方法推出了各自的平板电脑。在各种 Windows 或 Linux 版本上运行时,这些平板电脑是强大的计算机,但是对于大多数消费者而言,它们最终仍旧无法成为合适的移动解决方案。像苹果公司的 Newton 一样,很多平板电脑都需要使用手写笔才能准确操作微小的屏幕按钮和菜单选项。这些平板电脑的根本问题在于,这些桌面操作系统不是为手指驱动的触摸屏而设计的。从可用性的角度来看,相对于通常用一只手就能操作的移动平板设备而言,基于光标的桌面操作系统界面显得太过笨重了。

1.1.1 软件就是一切

在 2009 年,当关于苹果公司开发神秘平板电脑的谣言四起时,人们最大的疑问就是它会运行哪个操作系统。有了更大的屏幕,它当然可以使用 Mac OS X,并且我必须承认,我曾暗暗希望苹果公司会发布一款基于 Mac 的平板电脑,这样我就可以在其上运行我最喜欢的 Mac 应用程序了。但是作为开发人员,我不是普通的消费者。如果苹果公司将 iPad 发布为采用 Mac OS X 的设备,它肯定会与之前无数的平板电脑一样遭遇不温不火的市场。

当然,苹果公司要聪明得多。为了获得成功,苹果公司知道这款新型移动设备必须比笔记本电脑更易用,比智能手机更强大。为了实现这一目标,这个平板电脑需要一个从头开发的操作系统,适合多点触摸的手指动作和高效的触摸屏导航。苹果作为技术公司的一个最大优势在于它可以同时控制硬件和软件的设计,并且一直在寻求两者的完美结合点。iPad 就是这一理念的完美体现。

采用专为平板电脑改进的 iOS 版本,iPad 避免了让鼠标驱动的桌面软件适应触摸屏环境的常见困境。因为几百万 iPhone 和 iPod touch 用户已经熟悉了 iOS 界面,所以使用 iPad 时会感觉很舒服。它看起来很易于使用,因为事实的确如此。苹果公司首次发布 iPad 时,这款平板电脑所强调的简单性似乎并未给批评家们留下深刻印象,然而正是这一点将使其成为移动计算世界的转折点。

1.1.2 iPad 不是大号的 iPod touch

对 iPad 的未来表示怀疑的否定论者,仅仅将其定义为大号的 iPod touch,他们并没有了解其本质。他们没有看到 iPad 的主要部分。立即可用的 iOS 经证实是小型智能手机设备的理想之选,而且还将证明对于 iPad 较大的屏幕而言,它甚至更为高效。但是不要被 iPad 所具有的欺骗性外表所愚弄。它虽然看起来像个超大号的 iPod touch,但是实际上,iPad 却拥有强大的图形引擎和苹果公司推出的高速、省电的 A4 处理器。

iPad 不仅仅只适合游戏。尽管游戏使得 iPod touch 成为便携式游戏市场中的主力，而且很多游戏将针对苹果公司的平板电脑进行优化，但是我相信 iPad 将成为生产力应用程序的流行平台——甚至会超过 iPhone。

除了较大的显示屏和全尺寸的触摸屏键盘带来的便利外，iPhone SDK 3.2 还包含了 iPad 支持的 Core Text 和其他几个出色的新框架以及精美的用户界面，让你可以更为轻松地开发功能丰富的生产力应用程序。苹果公司已经将 iPad 定位为不仅是普通大众的首选便携式计算机，而且是医生、教师、学生、销售人员、科学家、飞行员、工程师和无数其他行业从业人员首选的便携式计算机。

苹果公司通过为 iPad 推出全新的 iWork 套件（参见图 1-1），表明了这一企图。通过提供相应 Mac 程序（Keynote、Pages 和 Numbers）复杂的 iPad 版本，iWork 是苹果公司对批评家们的一个有力反击，证明了其新款平板电脑不仅仅是大号的 iPod。

当人们觉得笔记本电脑不易于使用或者太沉而不便于携带时，他们会发现 iPad 的大小很适合，而且能够运行最新的桌面级应用程序。通过简单的 iOS 界面，这种新型移动应用程序将会大大提高人们的生产力，并提供更为亲切、更吸引人的体验，即使对技术一窍不通的新手也可以轻松使用。

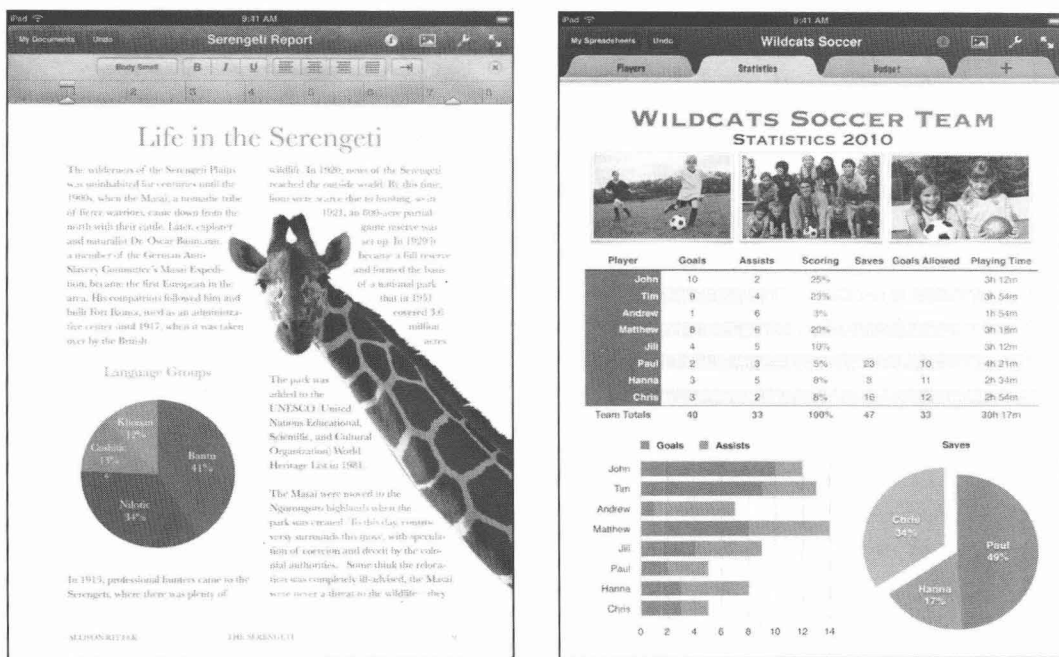


图 1-1 页面（左）和数字（右）是苹果公司面向 iPad 的 iWork 套件的一部分，本图完美展现了如何为易用的移动触摸屏设计桌面质量的复杂应用程序

受到 iPad 巨大潜力的鼓舞，一家出色的 iPhone 开发公司 Agile Web Solutions 在苹果公司刚

发布第一个试用版 SDK 的时候, 就开始开发 iPad 应用。Agile Web Solutions 的首席媒体制作人对 iPad 的潜力作出了如下评价:

有些人说 iPad 就像是个“大号的 iPod touch”, 这种看法太过短浅。我认为较大屏幕上的 iOS 会支持更丰富的多点触摸体验。只要看看苹果公司在 iWork [for iPad] 中包含的所有功能, 就会理解这一点。iPad 提供了更为强大的硬件, 并为开发人员提供了更多功能。它确实可以作为一台备用电脑, 甚至对于大多数用户而言可以作为首选电脑, iPad 确实让人眼前一亮。

1.2 面向大众的个人计算

在后面的章节探索新的以 iPad 为中心的 SDK 特性时, 会立即发现苹果公司为创建非常健壮的应用程序提供了丰富的工具。出于多年为 Mac OS X、Microsoft Windows 或 Linux 编程的经验, 尽管这些强大的工具唾手可得, 大多数用户还是喜欢构建类似传统桌面界面模型的应用程序。这种方法是完全错误的, 因为 iPad 平台已经去除了很多在 iPhone 上牵绊开发人员的用户界面设计限制和内存限制。

苹果公司使用 iOS 而不是 Mac OS X 作为 iPad 的操作系统是有原因的。首先它是为基于手指操作的触摸屏而设计的, 另外 iOS 还有另一个更重要的作用。与传统桌面操作系统不同, iOS 对用户隐藏了文件系统, 将用户的焦点放在了内容上。

很多人将 iPad 仅仅看做是笔记本的替代品, 而我认为苹果公司的新平板电脑设计时一定有更远大的目标: 重塑主流个人计算, 就像 1984 年原始 Mac 所做的那样。苹果公司认为, 对于普通消费者而言电脑太过复杂了, 它要简化这种体验。尽管像 Mac OS X 这样的桌面操作系统还要盛行几年, 但 iPad 代表了计算领域的一次重大的思维模式变革。

iPad 是为那些不喜欢使用电脑的人设计的。但是它也具备了足够的工程要素, 可以轻松满足高级用户的需求。总之, iPad 是一款适用于任何人的便携式个人电脑。

iPad 的精髓是其无与伦比的简单性。我知道这点我已经提过很多次了, 但是这一点确实是最重要的因素, 在为这款新设备开发应用程序时要牢记。苹果公司提供更有组织、更直观的用户体验的一个很重要的方法就是, 在界面设计过程中采用了对真实世界的模拟。苹果公司预先安装的 Notes 应用程序就是个很好的例子 (参见图 1-2)。

一个电脑新手第一次启动 Notes 应用程序时, 就可以立即弄清楚如何使用它, 无需任何说明或先前的电脑知识。你的爷爷奶奶都可以使用这个应用程序! 为了使整个体验更有趣, 苹果公司甚至添加了逼真的图形装饰, 比如手缝皮革的笔记本包, 里面有黄色横格的笔记本和白色卡片似的笔记清单。并且当前选中的笔记用一个“手画的”红色圆圈标记, 这样的细节处理得很贴近实际。

苹果公司鼓励开发人员在其自己的应用程序界面设计中包含这些新颖的变化。显然, 融合真实世界的对象、图像和纹理来表达功能不是在每个场景都实用, 但是这样做的主要目标很明确: 保持简单。在将潜在的应用程序想法付诸于开发时, 要记住这一基本准则。我们将在第 3 章中介

绍其他用户界面设计注意事项。

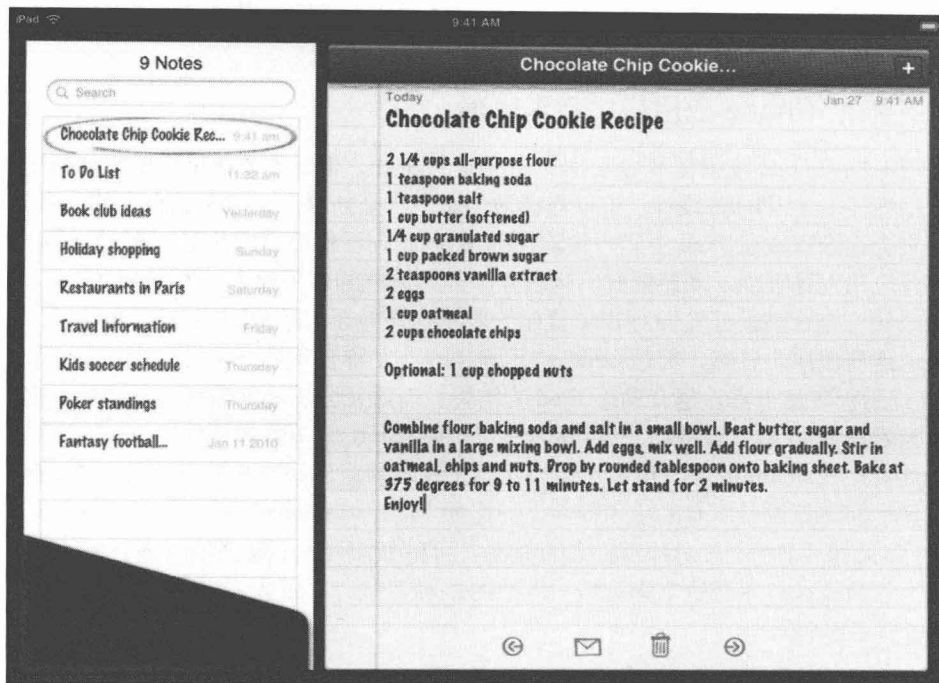


图 1-2 苹果公司设计的 Notes 应用程序界面，该界面模拟真实的笔记本，第一次使用的用户也一看就会

iPad 的不断成功完全依赖于为支持它的软件。苹果公司革命性的平板电脑为开发人员提供了一个全新的机会，他们可以为想要简化计算机体验而又不愿牺牲功能的人们打造应用程序。成功的 iPad 应用程序应该具有丰富的功能，同时又能保持非常直观易用的界面。

跟大多数开发人员一样，我经常被亲朋好友当做技术支持。绝大多数时候，他们遇到的都是些琐碎的小问题，比如找到放错位置的文件和电子邮件附件，解释数不完的配置选项，纠缠于晦涩难解的软件安装程序等。如果他们都有 iPad——只需指尖轻敲几下就可以轻松下载和使用应用程序——我就可以有更多的休息时间陪陪家人，不用花那么多时间解决台式机的问题了。在不久的将来，大家就会逐渐认识到 iPad 究竟是什么：它是面向大众的个人计算领域的下一次革命性变革。

1.3 为 iPad 开发应用程序

尽管 App Store 中 20 多万种应用程序的大部分都可以在 iPad 上“原样”运行，但是 iPhone 应用程序针对的是 320×480 像素的小尺寸，比 iPad 768×1024 像素的大屏幕的一半还小。尽管 iPad 支持对 iPhone 应用程序的向后兼容性，但其最终结果还是差强人意。在 iPad 上运行 iPhone 应用程序，苹果公司只提供了两个选项：在屏幕中央以原始大小显示（其余未使用区域保留为黑

色)和放大两倍填满屏幕。iPad 缩放算法似乎可以很好地工作,但是全屏 iPhone 应用程序仍然是像素化的。在被漂亮、高清晰度的 iPad 应用程序宠坏了之后,用户会发现大屏幕上放大后的 iPhone 应用程序非常粗糙、没有吸引力。

1.3.1 当你的 iPhone 应用程序不再完美

尽管现有的 iPhone 应用程序可以在 iPad 上运行良好,但是不要满足于低水准的用户体验。iPhone 应用程序专为 iPhone 设计。iPad 应该被当做一个全新平台,它拥有自己的一套设计要求。消费者一定会逐渐厌倦在 iPad 上运行像素化的 iPhone 应用程序,尤其是 App Store 中提供 iPad 增强版替代应用程序之后。既然如此,就应立即开始为你的应用程序开发 iPad 版。

苹果公司一直在鼓励开发针对 iPad 优化的应用程序并在 App Store 上专门开设了一个 iPad 专区用来展示这些程序。显然,促进 iPad 应用程序的开发符合苹果公司的最大利益,因为可以选择的 iPad 应用程序越丰富,越有助于 iPad 的销售。而这又会反过来最终帮你销售更多的应用程序。因为随着新 iPad 用户的增加,他们都会来到 App Store 来下载新软件。

作为这种开发动力的证明,现在苹果公司预先安装的应用程序几乎没有保持不变的。苹果公司花了很多时间重新设计所有的应用程序——比如 Mail、Calendar、Contacts、Photos、Safari 甚至 Notes (参见图 1-2)——来利用扩大的屏幕空间和 iPad 平台的新界面功能。而且绝大部分应用程序开发人员也认可这是适当满足用户期望的正确方向。下面是 Agile Web Solutions 的 David Chartier 对此问题的看法:

当然,在 iPad 上可以选择 2x 按钮在全屏模式下运行现有 iPhone 应用程序。但是一旦 iPad 客户看到了 iPad 更多的屏幕空间以及相对于 iPhone 和 iPod touch 的硬件和软件优势,你会发现“2x”模式很快就会变成 iPad 上的 Mac OS Classic。要在 iPad 上取得成功,我认为开发人员无疑要采用新特性和界面工具来提供最佳用户体验。如果你做不到这一点,用户会毫不犹豫地把你淘汰出局。

了解到 iPad 用户不会满足于运行放大的像素化的 iPhone 应用程序,开发人员正在加紧将其现有 iPhone 应用程序迁移到新的增强型 iPad 版本。他们的工作不仅仅是缩放界面以适应较大的屏幕空间。他们还对应用程序导航和用户界面架构实施了重大更改以便利用 iPad 独一无二的软硬件特性。

在为 iPad 开发应用程序时,有几个重要的设计方法和推荐的界面指导原则值得注意,这些内容会在第 3 章中详细介绍。现在,我们需要一点灵感激发我们的创意。我们先看一下几个 iPhone 开发人员如何为 iPad 重新设计应用程序。

1.3.2 探索无限可能

有一些知名开发人员乐于分享他们为 iPad 平台开发应用程序的心得。下面来详细了解一下这些应用程序,我们将介绍几个特定于 iPad 的用户界面元素和概念。如果你还不熟悉这些内容,不要着急。iPhone SDK 3.2 中提供的所有新 iPad 框架和用户界面控件都将在本书的其他部分详细

介绍, 这些介绍从第 3 章开始。

1. 画笔

Steve Sprang 赢得广泛称赞的 iPhone 应用程序——画笔 (<http://brushesapp.com/>) 是专为移动屏幕设计的绘图程序。它的界面看似简单, 但却提供了丰富的功能, 比如高级的颜色拾取器、几个真实的画笔、多个层、出色的缩放功能, 甚至还有撤销/重做选项。这款强大的工具用于在 iPhone 上绘图, 它深受广大移动数码艺术家们的欢迎。

选择新颜色或不同笔刷需要移动到新的界面视图。由于 iPhone 屏幕较小, 为了保持用户界面整洁易用, 这种设计策略十分必要。做出选择后, 艺术家可以返回到主画布界面 (参见图 1-3)。

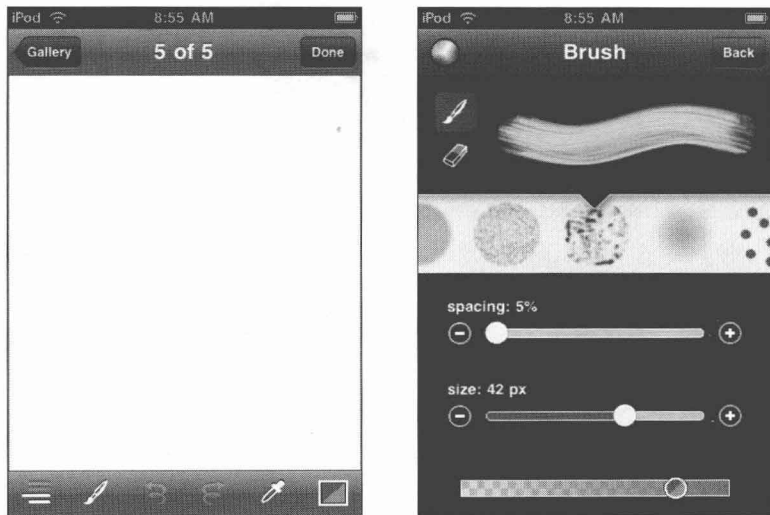


图 1-3 在 iPhone 上, 画笔将颜色拾取器和画笔调板分配到不同的界面视图。这需要用户在各种界面间切换, 但是在一个如此之小的设备上, 为了保持界面整洁易用, 这种设计策略必不可少

很多人第一次听说画笔程序是在 2009 年 6 月 1 日, 这期《纽约客》杂志漂亮的封面是艺术家 Jorge Colombo 完全用画笔程序在 iPhone 中创作的。然后, 在 2010 年 1 月, 画笔程序又重回公众视线, 因为在苹果公司的 iPad 主题发布会上, Steve Sprang 应邀推介其即将推出的 iPad 版画笔程序。除了展示 iPad 大屏幕上更大的绘图空间, 他还展示了如何使用新的浮动控制器在一个主画布界面内轻松访问不同的颜色拾取器和画笔调板 (参见图 1-4)。

浮动调板使 iPad 版画笔程序的操作方法更像传统桌面应用程序, 减少了在各种界面视图之间来回切换的需要, 例如在较小的 iPhone 或 iPod touch 上就需要切换。这只是众多新用户界面特性中的一个, 这些特性使 Sprang 能够在 iPad 上提供更强大、更简单的画笔体验。

得知了 Steve Sprang 是第一批使用 iPad SDK 框架和用户界面插件的非苹果开发人员之一, 我非常想了解他为 iPad 编程的体验。十分荣幸, Steve Sprang 从百忙中抽出了一些时间接受了我的简短访问。