

Join the discussion @ p2p.wrox.com



Wrox Programmer to Programmer™

移动与嵌入式开发技术



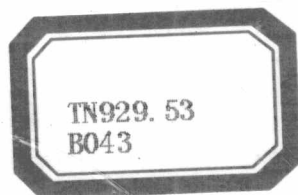
Professional iPhone and iPad Application Development

iPhone & iPad 高级编程

(美) Gene Backlin 著
岳虹 凌冲 译



清华大学出版社



郑州大学 *04010744903W*

移动与嵌入式开发技术

iPhone & iPad 高级编程

(美) Gene Backlin 著

岳虹 凌冲 译



清华大学出版社

北 京

TN929.53

Gene Backlin

Professional iPhone and iPad Application Development

EISBN: 978-0-470-87819-4

Copyright © 2011 by Wiley Publishing, Inc. Indianapolis, Indiana

All Rights Reserved. This translation published under license.

本书中文简体字版由 Wiley Publishing, Inc. 授权清华大学出版社出版。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。

北京市版权局著作权合同登记号 图字: 01-2011-0944

本书封面贴有 Wiley 公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

iPhone & iPad 高级编程/(美) 贝克林(Backlin, G.) 著; 岳虹, 凌冲 译. —北京: 清华大学出版社, 2012.1

书名原文: Professional iPhone and iPad Application Development

(移动与嵌入式开发技术)

ISBN 978-7-302-27445-2

I. i… II. ①贝… ②岳… ③凌… III. ①移动电话机—应用程序—程序设计 ②便携式计算机—应用程序—程序设计 IV. ①TN929.53 ②TP368.32

中国版本图书馆 CIP 数据核字(2011)第 247001 号

责任编辑: 王 军 韩宏志

装帧设计: 牛艳敏

责任校对: 邱晓玉

责任印制: 李红英

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 清华大学印刷厂

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 185×260 印 张: 34.25 字 数: 943 千字

版 次: 2012 年 1 月第 1 版 印 次: 2012 年 1 月第 1 次印刷

印 数: 1~4000

定 价: 59.80 元

产品编号: 041253-01

译者序

苹果公司分别于 2007 年 1 月 9 日和 2010 年 1 月 27 日推出了 iPhone 和 iPad 产品。iPhone 是将移动通信、宽屏触控、网页、地图和电子邮件功能完美融合在一起的智能手机；iPad 则是同样具有网页、地图和电子邮件功能，并能提供音频和视频服务的平板电脑。iPhone 和 iPad 引入了基于大型多点触控显示屏和新软件的全新用户界面，用户只需轻舒手指，就能酣畅地享用各种功能和服务。目前为止，iPhone 更新到 iPhone 4，而 iPad 则更新到 iPad 2。33 年风雨路让苹果公司经历了大起大落，从 1996 年开始，重新入主苹果公司的 Jobs 带领苹果重回个人 PC 舞台中心，再次迎来了黄金发展期。

用户眼中看到的是苹果产品完美的细节和非同凡响的想象力；程序员眼中看到的则是苹果公司从硬件到系统软件、从 Objective-C 语言到 Cocoa 框架的长期核心技术沉淀和积累。目前，苹果公司的各种产品在国内已经拥有相当高的市场占有率，这对程序员而言无疑是难得的良机。目前国内关于此类应用程序开发的中文参考书籍十分匮乏，因此，我们决定翻译本书，希望本书能够进一步促进 iPhone 和 iPad 程序员的开发水平，也希望帮助对此感兴趣的初学者步入 iPhone 和 iPad 应用程序开发殿堂。

本书作者 Gene Backlin 在计算机工业领域有超过 30 年的技术咨询经验，供职于 DePaul 大学计算与数字媒体学院。Gene 拥有多年的实践经验和教学经验，书中在讨论每个主题时都紧贴实际，详细列出开发步骤，并对涉及的源代码进行精讲。本书还讲述了如何查阅苹果开发者计划网站中的资料、文档和代码，以便您了解更多技术信息和学习方法，“授您以渔”，力求使您收到圆满的学习效果。

如果您跟随作者的讲述，动手完成书中的每个示例，那么您将拥有 iPhone 和 iPad 应用程序开发的编程能力和宝贵经验，更重要的是，这些示例中的技术细节很可能在您今后的程序设计中遇到，有些代码甚至可以原封不动地应用到开发当中。可以说，本书绝不是一本乏味的参考书，而是一本能够使读者在收获知识同时感受愉悦的读物。

关于本书内容的全面介绍，请参考前言部分，在此不再赘述。对于本书的翻译，我们力求做到语言平实无华，技术方面准确无误，期望能给读者带来轻松的阅读体验。在四个月的翻译过程中，我们反复对本书的内容进行推敲、修改和查证，真诚希望您奉献一本优秀作品，使本书能成为您的良师益友。

作者简介

Gene Backlin 是 MariZack 咨询公司的所有者兼首席顾问, MariZack 公司成立于 1991 年, 将“帮助客户取得成功”奉为唯一宗旨。Gene Backlin 为客户提供咨询服务的时间已逾 30 年, 服务过的客户包括 IBM、McDonnell Douglas、Waste Management、美国环境保护署(U.S. Environmental Protection Agency)、国家银行(Nations Bank)、美国银行(Bank of America)和第一银行(Bank One)等。Gene Backlin 也是 DePaul 大学计算与数字媒体学院的教师。

Gene Backlin 少年时代就对电子学产生了浓厚的兴趣, 这也帮助他成年后进入了计算机行业。Gene Backlin 至今还保留着他在 1978 年制作的 H-8 数字计算机和 H-9 视频终端。他自学了 Extended Benton Harbor Basic 编程语言。在 IBM-PC 上市后, Gene Backlin 开发出符合 Heathkit H-151 标准的计算机, 这些计算机至今仍在运行。如果您问他什么是信息革命, 他的回答一定是“这场革命令人心醉神迷, 带给人巨大的欢乐”。他在开发中使用过从纸带加载程序的计算机, 乃至革命性的 NeXT 计算机(他至今仍有两台), 再到 iPhone 和 iPad。Gene 为自己亲眼目睹计算机行业的飞速发展而感到荣幸, 更为能在该行业中大显身手而自豪。

Gene Backlin 曾在 1995 年撰写 *Developing NeXTSTEP Applications* 一书, 本书是该书的姊妹篇。

技术编辑简介

Daniel W. Meeks 目前在贝尔实验室为 Unix 操作系统 9.0(及其以后的版本)和 2.8 BSD 操作系统开发 grep、awk 和 sed 工具。Daniel W. Meeks 曾在内珀维尔的贝尔实验室参与开发 C++ Cfront 1.0 编译器。此后, Daniel W. Meeks 一直带领其技术小组为全球多家金融机构开发 C、Objective-C、C++、Java 和 C#项目并提供支持。他现在利用周末时间建设网络基础设施, 并将在 Apple、Windows 和 Android 操作系统上开发移动应用程序作为自己的爱好。

致 谢

首先感谢妻子 Roseann、儿子 Zachary 和女儿 Marissa，在撰写本书的过程中，你们给予我极大的关爱。还有我的儿子 Ethan，你使我青春永驻，永远年轻。

还要感谢我的母亲 Mary Louise 和女儿 Hannah Angel，你们教会了我很多东西，你们的精神伴随我生命中的每一天。

感谢我的父亲，当我需要您的时候您总出现在我的身边，您还给了我一个伟大的姓氏——Gene！

感谢 Helen 和 Jerry，你们对我来说无比重要。

感谢 Dan，很高兴时隔多年后与你重逢并共享美好时光。

本书是多位人士辛勤劳动的结晶；Carol、Ami 和 Luann 为本书的问世付出了极大的心血，特此对三位表达衷心的感谢。

最后再次感谢 Jean-Marc Krikorian，与您在 1995 年所做的一切相比，我所做的真的不算什么。

我不会忘记你们——我的读者，你们抽出了很多时间与我分享本书。本书为你们而写的，希望能对你们有所帮助。如有任何问题或意见，请告诉我，再次感谢你们！

前言

在 1975 年，我购买了一部 TI 公司生产的 SR-56 袖珍计算器，之所以购买它，是因为我看中了它的可编程功能。SR-56 袖珍计算器有 10 个存储器和 100 个编程步骤。在摆弄了大概一个星期后，我开始领会到程序的力量。我至今仍然保存着这个计算器，虽然它已经不能运行了，但它会唤起我对那段美好编程时光的回忆。

在过去几十年中，我在工作中使用过很多种技术和程序语言，久而久之，开发过程趋于平淡和乏味。起初令我心潮澎湃的那部 SR-56 也逐渐光环不在；直到 1989 年，NeXT 计算机的问世重新点燃我的激情。NeXT 计算机为开发人员提供了一套丰富的工具集，使用这套工具集可在很短的时间内得心应手地开发出复杂应用程序。NeXT 计算机使软件开发工作变得富有意义。

在 Mac 平台引入了 NeXT 开发环境后，两者开始了完美的融合，孕育出今天的 iPhone、iPod touch 和 iPad。从我开始在那部 SR-56 上编写第一个程序至今已经 35 年了，我始终对编程充满兴趣，就像一个孩子进了糖果店一样，迷恋其中，流连忘返。本书将帮您深入了解如何在上述设备中使用相应的开发工具得心应手地完成开发工作。

读者对象

本书主要面向熟悉 Xcode 开发环境和 Objective-C 程序语言的开发人员。

如果您熟悉应用程序开发但不了解 Xcode 开发环境和 Objective-C 程序语言，请首先访问苹果公司 iPhone 开发中心(Apple iPhone Dev Center)进行学习，网址为 <https://developer.apple.com/iphone/>。

本书每一章都讨论一个关于 iPhone/iPad 设备的专题或特性，将带您一步步地创建具有这些特性的应用程序(完整的工作示例)。开发过程中的源代码以模块形式组织，这样您就可以提取这些源代码，并在自己的应用程序中实现相同的功能。

本书内容

本书的主题涵盖使用当前的 SDK 4.0 开发 iPhone 和 iPod touch 应用程序，以及使用 SDK 3.2 开发 iPad 应用程序。Xcode 和 Interface Builder 是本书创建所有应用程序的主要开发工具。最后讨论了使用 Instruments 应用程序提高性能的问题。

编排方式

每一章都会浓墨重彩地描述 SDK 中的应用程序框架，然后通过循序渐进的过程指引您设计应用程序，以便帮助您理解这些框架，并学会在何时和何处将框架整合到自己的应用程序中。本书中

的内容需要通过动手实践来学习，在完成了所有“开发步骤”小节中的动手练习后，您将会拥有使用 iOS 4 为 iPhone 和 iPad 创建和添加功能的经验。本书主题涵盖表格视图、图像视图、选择器、数据存储、音频和视频等。最后，本书最后讨论如何确保应用程序高效运行，以便为那些从苹果公司 iTunes App Store 购买您的应用程序的用户提供令他们满意的用户体验。

使用本书前的准备工作

为了开发在 iPhone 或 iPod Touch 上运行的应用程序，需要下载 iPhone SDK 4.0。SDK 4.0 中包括了 SDK 3.2，以便您开发在 iPad 上运行的应用程序。可以从 <http://developer.apple.com/iphone/> 网页中获取 SDK。

虽然 SDK 是免费的，但您仍然需要注册为苹果公司的会员才能进行下载。如果只是想在 SDK 中的 iPhone/iPad 模拟器上运行的应用程序，那么注册为开发会员是免费的。如果想开发能在真实设备中安装的应用程序，或想在苹果公司的 iTunes Store 销售应用程序，就必须注册为付费会员。目前，价格最低的注册年费 99 美元。

安装 iPhone SDK 4.0 需要一台运行 Mac OS X 10.6.2(Snow Leopard)或更高版本的 Macintosh 计算机。

源代码

读者在学习本书中的示例时，既可以手动输入所有的代码，也可以使用本书附带的源代码文件。本书使用的所有源代码都可以从本书合作站点 <http://www.wrox.com/> 或 www.tupwk.com.cn/downpage 上下载。只要登录站点 <http://www.wrox.com/>，使用 Search 工具或使用书名列表就可以找到本书。接着单击本书细目页面上的 Download Code 链接，就可以获得所有源代码。



提示：由于许多图书的书名都很类似，所以按 ISBN 进行搜索是最简单的，本书英文版的 ISBN 是 978-0-470-87819-4。

勘误表

尽管我们已经尽了最大的努力来保证文章或代码中不出现错误，但是错误总是难免的，如果您在本书中找到了错误，例如拼写错误或代码错误，请告诉我们，我们将非常感激。通过勘误表，可以让其他读者避免走入误区，当然，这还有助于提供更高质量的信息。

要在网站上找到本书英文版的勘误表，可以登录 <http://www.wrox.com>，通过 Search 工具或书名列表查找本书，然后在本书的细目页面上，单击 Book Errata 链接。在这个页面上可以查看到 Wrox 编辑已提交和粘贴的所有勘误项。完整的图书列表还包括每本书的勘误表，网址是 www.wrox.com/misc-pages/booklist.shtml。

如果你在勘误表上没有找到错误,那么可以到 www.wrox.com/contact/techsupport.shtml 上,完成上面的表格,并把找到的错误发送给我们。我们将会核查这些信息,如果无误的话,会把它放置到本书的勘误表中,并在本书的后续版本中更正这些问题。

p2p.wrox.com

要与作者和同行讨论,请加入 p2p.wrox.com 上的 P2P 论坛。这个论坛是一个基于 Web 的系统,便于您张贴与 Wrox 图书相关的消息和相关技术,与其他读者和技术用户交流心得。该论坛提供了订阅功能,当论坛上有新的消息时,它可以给您传送感兴趣的论题。Wrox 作者、编辑和其他业界专家和读者都会到这个论坛上来探讨问题。

在 <http://p2p.wrox.com> 上,有许多不同的论坛,它们不仅有助于阅读本书,还有助于开发自己的应用程序。要加入论坛,可以遵循下面的步骤:

- (1) 进入 p2p.wrox.com, 单击 Register 链接。
- (2) 阅读使用协议, 并单击 Agree 按钮。
- (3) 填写加入该论坛所需要的信息和自己希望提供的其他信息, 并单击 Submit 按钮。
- (4) 你会收到一封电子邮件, 其中的信息描述了如何验证账户和完成加入过程。



提示: 不加入 P2P 也可以阅读论坛上的消息,但要张贴自己的消息,就必须加入该论坛。

加入论坛后,就可以张贴新消息,回复其他用户张贴的消息。可以随时在 Web 上阅读消息。如果要让该网站给自己发送特定论坛中的消息,可以单击论坛列表中该论坛名旁边的 Subscribe to this Forum 图标。

关于使用 Wrox P2P 的更多信息,可阅读 P2P FAQ,了解论坛软件的工作情况以及 P2P 和 Wrox 图书的许多常见问题。要阅读 FAQ,可以在任意 P2P 页面上单击 FAQ 链接。

目 录

第 1 章 导航..... 1	2.2 一个简单的 iPhone-iPod touch 和 iPad 警告视图..... 52
1.1 导航栈..... 2	2.2.1 开发步骤：向 UITextView 中加载一个文件 53
1.2 导航栏..... 2	2.2.2 测试应用程序 63
1.2.1 UINavigationController 协议 2	2.3 操作列表..... 63
1.2.2 配置导航栏 2	2.3.1 设计操作列表 63
1.2.3 压入和弹出导航项 3	2.3.2 UIActionSheetDelegate 协议..... 64
1.3 一个简单的导航栏..... 3	2.4 一个 iPhone-iPod touch 操作列表..... 64
1.3.1 开发步骤：一个简单的导航栏 4	2.4.1 开发步骤：创建一个 iPhone 或 iPod touch 操作列表 65
1.3.2 测试应用程序 16	2.4.2 测试应用程序 76
1.4 工具栏..... 17	2.5 一个 iPad 操作列表..... 76
1.5 一个简单的工具栏..... 17	2.5.1 开发步骤：创建一个 iPad 操作列表 76
1.5.1 开发步骤：一个简单的工具栏 18	2.5.2 最终步骤：建立关联..... 89
1.5.2 测试应用程序 25	2.5.3 测试应用程序 91
1.6 选项卡栏..... 25	2.6 模态视图..... 91
1.6.1 UITabBarDelegate 协议..... 25	2.6.1 显示和取消模态视图..... 91
1.6.2 定制选项卡栏 25	2.6.2 转换方式 91
1.7 一个简单的选项卡栏..... 25	2.6.3 模态显示方式(仅适用于 iPad)..... 91
1.7.1 开发步骤：一个简单的选项卡栏 26	2.7 一个 iPhone 和 iPod touch 模态视图..... 91
1.7.2 设计视图控制器 28	2.7.1 开发步骤：创建一个 iPhone 和 iPod touch 模态视图 92
1.7.3 测试应用程序 49	2.7.2 测试应用程序 101
1.8 小结..... 49	2.8 一个 iPad 模态视图..... 101
第 2 章 警告、操作列表和模态视图..... 51	
2.1 警告..... 51	
2.1.1 设计警告视图 51	
2.1.2 UIAlertViewDelegate 协议 52	

2.8.1 开发步骤: 创建一个 iPad 模态视图	103	5.1.2 多点触控	154
2.8.2 测试应用程序	113	5.1.3 点击	155
2.9 小结	113	5.1.4 滑动和手势	155
第 3 章 定制表格视图	115	5.2 一个简单的触控处理程序	156
3.1 表格视图	115	5.2.1 开发步骤: 一个简单的触控 处理程序	157
3.1.1 表格视图单元格	116	5.2.2 测试应用程序	167
3.1.2 UITableViewDataSource 协议	116	5.3 一个简单的手势识别器	167
3.1.3 UITableViewDelegate 协议	116	5.3.1 开发步骤: 一个简单的手势 识别器	169
3.2 一个定制表格视图应用程序	116	5.3.2 测试应用程序	190
3.2.1 开发步骤: 一个定制表格视图 应用程序	117	5.4 小结	190
3.2.2 创建 Contacts.plist 属性列表 文件	128	第 6 章 处理通知	191
3.2.3 测试应用程序	129	6.1 NSNotification 概念	192
3.3 小结	129	6.1.1 通知中心	192
第 4 章 拆分视图	131	6.1.2 注册本地通知	192
4.1 UISplitViewController 类	131	6.1.3 发布本地通知	192
4.1.1 UIPopoverControllerDelegate 协议	132	6.1.4 注销观察者	193
4.1.2 UISplitViewControllerDelegate 协议	132	6.2 一个本地命名通知	193
4.2 一个简单的拆分视图 应用程序	132	6.2.1 开发步骤: 一个本地命名 通知	193
4.2.1 开发步骤: 一个简单的 iPad 拆分视图应用程序	135	6.2.2 测试应用程序	202
4.2.2 创建 DataSource.plist 属性 列表文件	138	6.3 一个本地键盘通知	202
4.2.3 测试应用程序	152	6.3.1 开发步骤: 一个本地键盘 通知	203
4.3 小结	152	6.3.2 测试应用程序	216
第 5 章 触控事件	153	6.4 小结	216
5.1 处理触控事件	153	第 7 章 联网概念	217
5.1.1 单点触控	154	7.1 网络通信	218
		7.2 一个简单的网络浏览器	220
		7.2.1 开发步骤: 一个简单的 网络浏览器	220
		7.2.2 测试应用程序	234
		7.3 对等设备通信	234
		7.3.1 开发步骤: 对等设备通信	235
		7.3.2 测试应用程序	252

7.4 小结	252	9.4 小结	325
第 8 章 多媒体	253	第 10 章 数据存储	327
8.1 音频框架	253	10.1 属性列表	327
8.1.1 媒体播放器	253	10.1.1 属性列表的使用	327
8.1.2 AV 基础框架	253	10.1.2 推荐的数据元素类型	328
8.1.3 音频工具箱	254	10.1.3 保存和还原属性列表	328
8.1.4 音频单元框架	254	10.2 核心数据	329
8.1.5 OpenAL 框架	254	10.2.1 核心数据栈	329
8.2 视频框架	254	10.2.2 托管对象	329
8.2.1 MPMoviePlayerController	255	10.2.3 托管对象上下文	329
8.2.2 支持的格式	255	10.2.4 托管对象模型	329
8.3 一个播放 iPod 库中音频的 应用程序	255	10.2.5 持久存储协调器	329
8.3.1 开发步骤: 播放 iPod 库中的 音频	256	10.2.6 Xcode 建模工具	330
8.3.2 测试应用程序	280	10.2.7 获取托管对象	330
8.4 一个播放 iPod 库中视频的 应用程序	281	10.2.8 删除托管对象	330
8.4.1 开发步骤: 一个播放 iPod 库中视频的应用程序	281	10.3 数据存储的共同基础	331
8.4.2 测试应用程序	302	10.3.1 开发步骤: 一个使用属性 列表的简单应用程序	332
8.5 小结	302	10.3.2 测试应用程序	354
第 9 章 应用程序首选项	303	10.3.3 开发步骤: 一个使用核心 数据的简单应用程序	354
9.1 应用程序配置	303	10.3.4 测试应用程序	381
9.1.1 应用程序首选项指南	304	10.4 小结	381
9.1.2 首选项元素类型	304	第 11 章 剪贴板	383
9.1.3 实现首选项层次结构	304	11.1 剪贴板的概念	384
9.1.4 访问应用程序首选项	304	11.1.1 命名的剪贴板	384
9.2 设置简单的首选项	305	11.1.2 持久性	384
9.2.1 开发步骤: 设置简单的 首选项	305	11.1.3 编辑菜单	384
9.2.2 测试应用程序	314	11.1.4 剪切所选内容	385
9.3 创建一个子窗格首选项层次 结构	314	11.1.5 粘贴项	386
9.3.1 开发步骤: 创建一个子窗格 首选项层次结构	315	11.1.6 取消编辑菜单	386
9.3.2 测试应用程序	325	11.2 剪切和粘贴文本	386
		11.2.1 开发步骤: 剪切和 粘贴文本	387
		11.2.2 测试应用程序	395
		11.3 剪切和粘贴图像	395

11.3.1	开发步骤: 剪切和粘贴 图像.....	396	第 14 章	集成 iAd 网络.....	455
11.3.2	测试应用程序.....	403	14.1	加入 iAd 网络.....	455
11.4	为 iPad 创建定制菜单.....	404	14.1.1	设置银行账户信息.....	456
11.4.1	开发步骤: 为 iPad 创建 定制菜单.....	405	14.1.2	为应用程序启用 iAd 网络选项.....	456
11.4.2	测试应用程序.....	413	14.1.3	配置 iAd 首选项.....	456
11.5	小结.....	413	14.2	使应用程序做好使用 iAd 网络的准备.....	456
第 12 章	单元测试.....	415	14.2.1	实现 ADBannerView.....	456
12.1	设置环境.....	415	14.2.2	集成 iAd 框架.....	456
12.2	一个简单的单元测试.....	416	14.3	一个使用 iAd 网络的简单 应用程序.....	457
12.2.1	开发步骤: 一个简单的 单元测试.....	417	14.3.1	开发步骤: 一个使用 iAd 网络的简单应用程序.....	457
12.2.2	创建 Data.plist 数据源.....	427	14.3.2	测试应用程序.....	464
12.2.3	测试应用程序.....	430	14.4	小结.....	464
12.3	创建单元测试的步骤: 一个 简单的单元测试.....	431	第 15 章	多任务处理.....	465
12.3.1	对应用程序进行单元 测试.....	435	15.1	多任务处理服务.....	465
12.3.2	注册设备.....	436	15.1.1	UIApplication 委托消息.....	466
12.3.3	对应用程序进行单元 测试.....	436	15.1.2	多任务处理的作用.....	466
12.4	小结.....	437	15.1.3	Xcode 模拟器中多任务 处理的限制.....	467
第 13 章	调整和优化性能.....	439	15.1.4	支持多任务处理的 设备.....	467
13.1	性能分析.....	439	15.2	一个多任务音频应用程序.....	467
13.1.1	使用模拟器.....	440	15.2.1	开发步骤: 一个多任务音频 应用程序.....	468
13.1.2	使用设备.....	440	15.2.2	测试应用程序.....	486
13.1.3	性能分析的好处.....	440	15.3	小结.....	487
13.2	一个简单的内存泄漏测试.....	440	附录 A	开发应用程序的初始步骤.....	489
13.2.1	开发步骤: 内存泄漏 测试.....	444	附录 B	iPhone 开发中心.....	499
13.2.2	开发步骤(续): 使用检测 应用程序.....	452	附录 C	Cocoa touch 静态库.....	505
13.2.3	开发步骤(续): 在主菜单中 使用 Build and Analyze 选项.....	452	附录 D	苹果开发资源.....	527
13.3	小结.....	453			

第 1 章

导 航

本章内容:

- 如何在 iPad 拆分视图中进行导航
- 使用工具栏旋转图像
- 使用选项卡栏实现一个简单的银行账户交易跟踪器

导航是一个通过层次结构找到所需信息的过程。在 iPhone 和 iPad 中，可以使用以下组件来导航数据；其中每个组件的导航观感各不相同：

- **导航栏(Navigation bar)**——按照层次结构安排数据，可以使用下查功能进行导航，同时提供返回上层或顶层的路径。
- **工具栏(Toolbar)**——提供一些用于操作当前视图上下文的选项。
- **选项卡栏(Tab bar)**——为同一组数据提供不同的视图。

本章介绍了为上述每个导航组件创建应用程序的步骤，向您演示每种导航风格的一个简单用例。运行应用程序的设备决定了所实现的导航风格。由于受限于可视区域，iPhone 比 iPad 具有更多的下查数据。如果准备开发同时兼容这两种设备的应用程序，就必须注意这一点。从视觉表现方面讲，iPad 不应该只是复制一个应用程序。



在设计应用程序的导航方式时，请查看苹果公司的 iPad 和 iPhone 用户界面指南。请参阅附录 D，查看苹果公司提供的相关文档资源及其他开发指南。

1.1 导航栈

导航过程基于栈。视图以后进先出(LIFO)的方式存入由视图控制器(View Controller)管理的栈中。初始视图是根视图控制器, 可被其他视图覆盖。但与其上的其他视图不同, 根视图控制器不能删除。导航过程对用户交互作出响应, 将视图控制器压入和弹出导航栈。每个当前视图负责将下一个视图压入导航栈中。

为导航项提供具体内容的是导航栏。用户单击导航栏上的按钮将产生委托消息, 这些消息发送到视图控制器, 从而执行视图的压入或弹出操作。

1.2 导航栏

应用程序中的导航栏主要包括用于导航视图的控件对象。导航栏为用户提供了按照应用程序的层次结构将视图压栈或弹栈所需的全部控件。由关联的视图控制器通过委托方法来处理委托消息。

1.2.1 UINavigationController 协议

要将一个导航项压入或弹出导航栈, 视图控制器需要实现 UINavigationController 协议的多个方法。

需要实现的方法如下:

- 压入导航项
 - (1) navigationController:shouldPushItem:
 - (2) navigationController:didPushItem:
- 弹出导航项
 - (1) navigationController:shouldPopItem:
 - (2) navigationController:didPopItem:

1.2.2 配置导航栏

导航栏位于视图顶部, 会显示当前视图的标题。此外, 导航栏还可能包含一些按钮控件, 以使用户在当前视图上下文中完成一些操作。为了实现此功能, 可以使用以下几种方法:

- 位于导航栏左边的 backButtonItem 和 leftBarButtonItem。
- 位于导航栏中间的 titleView。
- 位于导航栏右边的 rightBarButtonItem。

导航栏本身具有一些可以被修改的属性:

- barStyle
- translucent
- tintColor

1.2.3 压入和弹出导航项

从一个视图导航到另一个视图，要么是继续下查(压栈)，要么是返回上层(弹栈)，这些操作都由应用程序中的视图控制器来处理。可将导航过程简单地视为由导航控制器在导航栈中管理一系列视图控制器的行为。

视图控制器负责在导航栈中压入(在层次结构中下压)和弹出(在层次结构中返回)其他视图控制器。导航过程如下：

- (1) 创建 UINavigationController。
- (2) 导航控制器将视图控制器压入导航栈。
- (3) 视图控制器显示下一个视图。
- (4) 视图控制器取消上一个视图。

1.3 一个简单的导航栏

在此应用程序中，导航功能是显示 1~20 之间的数字。这 20 个数字按照奇偶数分组。需要注意，使用 iPad 的拆分视图(split view)时，在纵向方向上，导航栏呈现为弹出菜单(popover)，而在横向方向上，导航栏则显示在拆分视图的左侧窗格中，如图 1-1 所示。

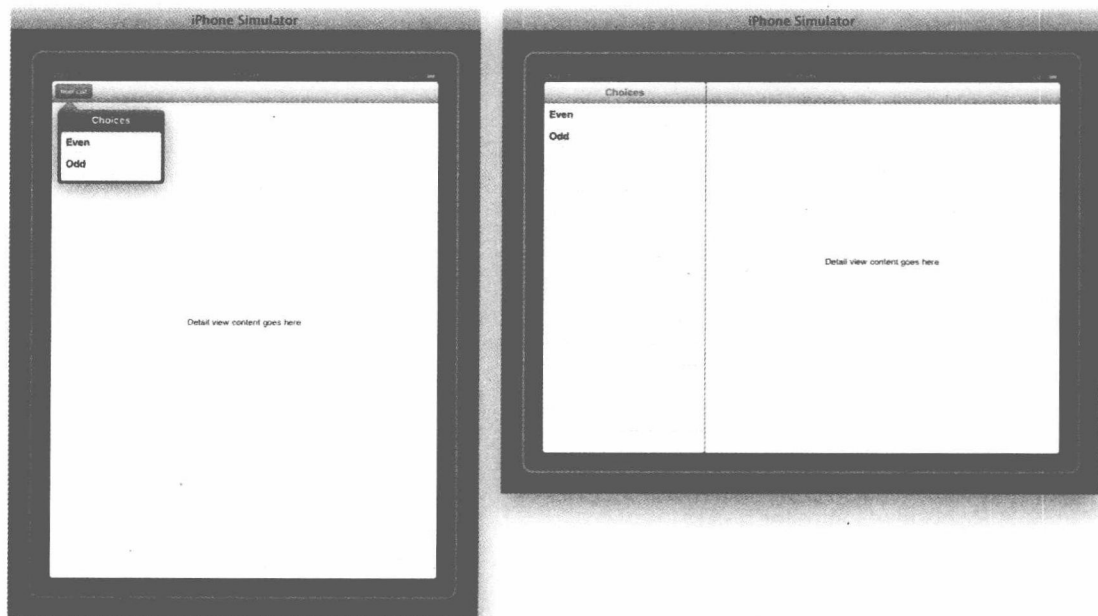


图 1-1

点击 Odd 选项将看到另一个导航栏，其中列出了 1~20 之间的奇数，如图 1-2 所示。

点击该列表中的任意一个数字，将在详细信息视图中看到所选择的内容，而弹出菜单则会消失(如图 1-3 所示)。