

PEARSON

- 世界知名 iOS 开发专家 Erica Sadun 全新打造
- iOS 开发领域扛鼎之作，全面涵盖 iOS 开发细节

iOS 6

高级开发手册

(第4版)

The
Advanced iOS 6
Developer's Cookbook

[美] Erica Sadun 著
陈宗斌 译



人民邮电出版社
POSTS & TELECOM PRESS

PEARSON

iOS 6 高级开发手册

(第4版)

The
Advanced iOS 6
Developer's Cookbook

[美] Erica Sadun 著
陈宗斌 译

人民邮电出版社
北京

图书在版编目(CIP)数据

iOS 6高级开发手册：第4版 / (美) 萨顿
(Sadun, E.) 著；陈宗斌译. — 北京：人民邮电出版社，
2014. 4

ISBN 978-7-115-34425-0

I. ①i… II. ①萨… ②陈… III. ①移动终端—应用
程序—程序设计—技术手册 IV. ①TN929.53-62

中国版本图书馆CIP数据核字(2014)第015888号

版 权 声 明

Erica Sadun: The Advanced iOS 6 Developer's Cookbook (4th Edition)

Copyright © 2013 Pearson Education, Inc.

ISBN: 978-0321884220

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise without the prior consent of Addison Wesley.

版权所有。未经出版者书面许可，对本书任何部分不得以任何方式或任何手段复制和传播。
本书中文简体字版由人民邮电出版社经 **Pearson Education, Inc.** 授权出版。版权所有，侵权必究。
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。

-
- ◆ 著 [美] Erica Sadun
 - 译 陈宗斌
 - 责任编辑 傅道坤
 - 责任印制 程彦红 杨林杰
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
 - 邮编 100164 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 大厂聚鑫印刷有限责任公司印刷
 - ◆ 开本：800 × 1000 1/16
 - 印张：30.5
 - 字数：607 千字 2014 年 4 月第 1 版
 - 印数：1—2 500 册 2014 年 4 月河北第 1 次印刷
 - 著作权合同登记号 图字：01-2013-5716 号
-

定价：89.00 元

读者服务热线：(010)81055410 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京崇工商广字第 0021 号

内容提要

本书是市面上最畅销的 iOS 开发图书的全新升级版本，以苹果发布的 iOS 6 为基础编写而成。在本书中，资深 iOS 开发专家 Erica Sadun 与大家分享了一些用于 iOS 6 高端开发的成熟、可靠的方法，并借助大量的代码示例对这些方法进行演示讲解，从而降低了 iOS 开发的学习难度。

本书是《iOS 6 核心开发手册（第 4 版）》的姊妹篇，总共分为 13 章，先后讲解了特定设备的 iOS 开发、文档和数据的共享、Core Text 的使用方式、iOS 开发中使用到的几何学知识、应用在接入网络时需要解决的问题、图像的处理、图像捕获、音频处理、Address Book 框架在应用中的使用、地理定位、GameKit 和 StoreKit 的使用，以及如何使用推送通知等内容。

本书语言简练、内容丰富，并在配套网站上提供了完整的示例代码，适合具有一定 iOS 开发经验或其他移动开发经验的人员阅读。对 iOS 开发感兴趣的入门者，也可以从本书姊妹篇《iOS 6 核心开发手册（第 4 版）》开始起步，逐步学会、掌握 iOS 的开发。

关于作者

Erica Sadun 是数十本畅销书的著者、合著者和供稿者，这些书涉及程序设计、数字视频、数字摄影和网页设计，包括广受欢迎的 *The iOS 5 Developer's Cookbook*。她目前在 tuaw.com 上开通了博客，过去也曾在 O'Reilly 的 Mac DevCenter、Lifehacker 和 Ars Technica 上发表过博文。Erica 是数十种 iOS 原生应用程序的编写者，并拥有佐治亚理工学院图形、可视化和可用性计算中心的计算机科学博士学位。作为极客、程序员和写作者，她还没遇到过不喜欢的小工具。笔耕之余，她和她同为极客的丈夫还养育了三个小极客。这三个小极客如果不是忙着在屋子里重新布线 and 策划控制全球，就会按耐着困惑，盯着他们的爸爸妈妈。

献辞

谨 将本书献给我的丈夫 **Alberto**，多年以来，他一直在默默忍受我不时提及与 **iOS** 开发相关的各种 **SDK** 和小工具，而且在本书完成之日，他依然能够保持良好的心态。

致谢

如果没有 Chuck Toporek 的努力,本书将不会问世,Chuck Toporek 是我的编辑,多年来担任过多家出版社的项目负责人。他现在在 Apple 工作,我非常想念他。如果没有他,读者将不会看到这本 Cookbook。他兼具两种优秀的技能:激励作者做他们认为自己做不到的事情,以及挥舞着巨大的“真实鳟鱼”¹吸引作者专注于图书的主题并且不脱离现实。这样的反复鞭挞使图书得以在最后期限前问世,并且使书中的内容非常有吸引力。

还要感谢 Trina MacDonald (我的非常优秀的新编辑)、Chris Zahn (才华横溢的开发编辑) 和 Olivia Basegio (忠实且富有激情的助理编辑,总能在幕后把事情办好)。此外,还要对整个 Addison-Wesley/Pearson 制作团队深表谢意,特别感谢 Kristy Hart、Jovana San Nicolas-Shirley、San Dee Phillips、Nonie Ratcliff 和 Chuti Prasertsith。还要感谢 Safari 的工作人员使我的图书初步成形,并且在出现技术性问题时进行修正。

还要感谢我多年的经纪人 Neil Salkind,以及技术评审 Oliver Drobnik、Rich Wardwell 和 Duncan Champney,他们帮助我使本书合乎情理,而不是一厢情愿地去组织内容,还要感谢我在 TUAW、Ars Technica 和 Digital Media/Inside iPhone 博客的所有现在和以前的同事。

我非常感谢人才济济的 iOS 开发人员社区,包括 Jon Bauer、Tim Burks、Matt Martel、Tim Isted、Joachim Bean、Aaron Basil、Roberto Gamboni、John Muchow、Scott Mikolaitis、Alex Schaefer、Nick Penree、James Cuff、Jay Freeman、Mark Montecalvo、August Joki、Max Weisel、Optimo、Kevin Brosius、Planetbeing、Pytey、Michael Brennan、Daniel Gard、Michael Jones、Roxfan、MuscleNerd、np101137、UnterPerro、Jonathan Watmough、Youssef Francis、Bryan Henry、William DeMuro、Jeremy Sinclair、Arshad Tayyeb、Jonathan Thompson、Dustin Voss、Daniel Peebles、ChronicProductions、Greg Hartstein、Emanuele Vulcano、Sean Heber、Josh Bleacher Snyder、Eric Chamberlain、Steven Troughton-Smith、Dustin Howett、Dick Applebaum、Kevin Ballard、Hamish Allan、Lutz Bendlin、Oliver Drobnik、Rod Strougo、Kevin McAllister、Jay Abbott、Tim Grant Davies、Maurice Sharp、Chris Samuels、Chris Greening、Jonathan Willing、Landon Fuller、Jeremy Tregunna、Christine Reindl、Wil Macaulay、Stefan Hafenegger、Scott Yelich、Mike Kale、chralllelinder、John

¹ 在本书的开发和制作过程中没有伤害真实的或想象的鳟鱼,但是无数罐健怡可乐就没有这么好的命运了,在编写本书原稿的过程中,我们喝了大量的可乐。

2 致谢

Varghese、Robert Jen、Andrea Fanfani、J. Roman、jtbandes、Artissimo、Aaron Alexander、Christopher Campbell Jensen、Nico Ameghino、Jon Moody、Julián Romero、Scott Lawrence、Evan K. Stone、Kenny Chan Ching-King、Matthias Ringwald、Jeff Tentschert、Marco Fanciulli、Neil Taylor、Sjoerd van Geffen、Absentia、Nownot、Emerson Malca、Matt Brown、Chris Foresman、Aron Trimble、Paul Griffin、Paul Robichaux、Nicolas Haunold、Anatol Ulrich (hypnocode GmbH)、Kristian Glass、Remy “psy” Demarest、Yanik Magnan、ashikase、Shane Zatezalo、Tito Ciuro、Mahipal Raythatha、Jonah Williams of Carbon Five、Joshua Weinberg、biappi、Eric Mock，以及 irc.saurik.com 和 irc.freenode.net 上的 iPhone 开发人员频道中的每一个人，还有许多其他的人，在此不能一一指出。他们的技术、建议和反馈帮助使本书成为可能。如果我忽视了任何帮助过我的人，请接受我的道歉。

在此要特别感谢我的家人和朋友，他们支持我月复一月地进行新测试版发布，并且对我莫名其妙的缺席和绝望的嗥叫表现出了极大的耐心。我感谢你们一直坚持陪伴在我身边。我还要感谢我的孩子们，当他们知道他们的母亲天天驼着背，忙着敲击键盘而无暇顾及他们时，依然表现出了坚定的支持。我的孩子们在过去几个月给我提供了非常宝贵的帮助，他们测试应用程序、提供建议，并且成长为了了不起的人。我尽量每天都提醒自己，这些孩子成为我生命的一部分我该有多幸运。

前言

欢迎阅读另一本 iOS Cookbook 图书!

有了 iOS 6, Apple 的移动设备家族在使人兴奋和可能做到的事情方面达到了一个新的层次。本书可以帮助你开始开发程序。这个修订版介绍了最新的 WWDC 中公布的所有新特性,说明了如何把它们纳入应用程序中。

对于这个版本,我的出版团队明智地把 Cookbook 材料划分成册,以使之容易管理。本书关注的是公共框架(比如 StoreKit、GameKit 和 Core Location)以及一些便于使用的技术(比如图像处理排版)。它有助于构建利用专用库的应用程序并超越基本技术。本册针对的是那些精通 iOS 开发并且谋求掌握专业领域的实用知识的读者。

与之配套的一册图书是《iOS 6 核心开发手册(第4版)》,它提供了针对日常开发的核心解决方案。它涵盖了使用标准 API 和接口元素创建 iOS 应用程序所需的所有类,还包含处理图形、触摸和视图以创建移动应用程序所需的秘诀。

最后,还有一本图书是 *Learning iOS 6: A Hands-on Guide to the Fundamentals of iOS Programming*, 其中包含了大量教程材料,它由 Cookbook 的前几章组成。在该书中,可以找到从头开始学习 iOS 6 开发所需的所有基本知识。从 Objective-C 到 Xcode,从调试到部署,该书讲述了如何开始使用 Apple 的开发工具套件。

像过去一样,可以在 GitHub 上找到示例代码。在 <https://github.com/erica/iOS-6-Cookbook> 上可以找到这本 Cookbook 的知识库,在 WWDC 2012 发布之后,它全都针对 iOS 6 进行了更新。

如果你有什么建议、错误修正方法和校正意见,或者可以为将来的版本做贡献的其他任何想法,可以给我发送电子邮件: erica@ericasadun.com。让我提前感谢你。我会感谢所有的反馈,它们有助于使本书变得更好、更优秀。

——Erica Sadun, 2012 年 9 月

用户需要什么

如果计划构建 iOS 应用程序,将至少需要一个 iOS 设备,用于测试应用程序,首选新款 iPhone 或平板电脑。下面的列表涵盖了开始时必须准备的一些物品。

- **Apple 的 iOS SDK**——可以从 Apple 的 iOS Dev Center (<http://developer.apple.com/ios>) 下载 iOS SDK 的最新版本。如果计划通过 App Store 销售应用程序,可以变成一名付费的 iOS 开发者。对于个人开发者,其费用是 99 美元/年,对于企业(即公司)开发者,费用则是 299 美元/年。注册的开发者将收到证书,允许他们“登录”并把应用程序下载到他们的 iPhone/iPod Touch 上,以进行测试和调试,并且及早访问 iOS 的预发行版本。免费程序开发者可以在基于 Mac 的模拟器上测试他们的软件,但是不能部署到设备上或者提交给 App Store。

大学计划

Apple 还为学生和教师提供了一个大学计划 (University Program)。如果你是一名计算机学科的学生,并且学习的是大学级别的课程,就要与教员核实一下,看看你所在的学校是否是 University Program 的一部分。有关 iPhone 开发者大学计划 (iPhone Developer University Program) 的更多信息,参见 <http://developer.apple.com/support/iphone/university>。

- **一台现代的 Mac 机**, 运行 Mac OS X Lion (版本 10.7), 最好是运行 Mac OS X Mountain Lion (版本 10.8)——需要大量的磁盘空间用于开发,并且 Mac 应该具有尽可能多的 RAM。
- **iOS 设备**——尽管 iOS SDK 包括一个模拟器用于测试应用程序,但是确实需要拥有 iOS 硬件,以便为平台开发程序。可以把单元连接到计算机上并安装你构建的软件。对于真实的 App Store 部署,手边有多个单元是有益的,用以表示几代不同的硬件和固件,以便可以在目标受众将使用的相同平台上执行测试。
- **Internet 连接**——该连接允许利用真实的 Wi-Fi 连接以及 EDGE 或 3G 服务测试程序。
- **熟悉 Objective-C**——要为 iPhone 编写程序,需要知道 Objective-C 2.0。该语言基于具有面向对象扩展的 ANSI C,这意味着还需要知道一点 C 语言的知识。如果利用 Java 或 C++编写过程序并且熟悉 C 语言,就应该能够比较容易地转向 Objective-C。

Mac/iOS 开发的路标

一本书不可能做到面面俱到。如果尝试把你需要知道的所有知识都打包进这本书中,那么你将学不会它(事实是,本书提供了一个优秀的智力开发工具,不要让自己过于紧张)。为 Mac 和 iOS 平台开发程序的确需要知道许多知识。如果你只是刚刚起步并且没有任何编程经验,那么第一步应该是学习 C 程序设计语言的大学级别的课程。尽管字母表可能从字母 A 开始,但是大多数程序设计语言都起源于 C,当然你通往开发者的道路也是如此。

一旦知晓了 C 语言以及如何使用编译器(在基本的 C 语言课程中将会学到它),余下的应该就简单了。从此,可以直接跳到 Objective-C,并且学习如何利用该语言以及 Cocoa 框架编程。

图 0-1 所示的流程图显示了 Pearson Education 提供的关键图书，它们可以提供必要的培训，而这是你为了变成熟练的 iOS 开发者所需要接受的。

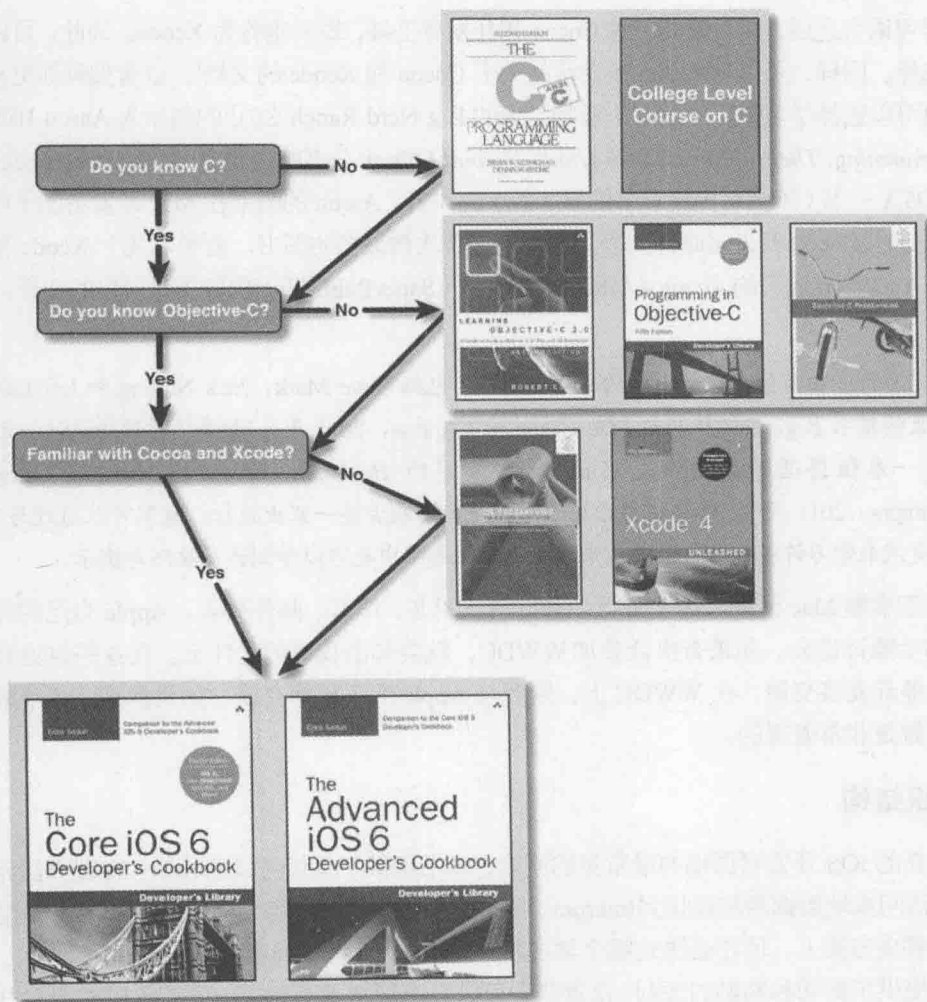


图 0-1 成为 iOS 开发者的路标

一旦知道了 C 语言，就可以选择学习如何利用 Objective-C 编程。如果你想深入了解该语言，可以阅读 Apple 自己的文档，或者从下面这些关于 Objective-C 的图书中选择一本来阅读。

- Aaron Hillegass 编写的 *Objective-C Programming: The Big Nerd Ranch Guide* 一书（Big Nerd Ranch，2012 年）。
- Robert Clair 编写的 *Learning Objective-C: A Hands-on Guide to Objective-C for Mac and*

iOS Developers 一书 (Addison-Wesley, 2011 年)。

- Stephen Kochan 编写的 *Programming in Objective-C 2.0, Fourth Edition* 一书 (Addison-Wesley, 2012 年)。

在学习语言之后,接下来要应对 Cocoa 和开发者工具,后者也称为 Xcode。为此,可以有几个不同的选择。同样,可以参考 Apple 自己的关于 Cocoa 和 Xcode 的文档¹,或者如果你更喜欢阅读图书,也可以选择学习最好的图书。亚特兰大的 Big Nerd Ranch 公司²的创始人 Aaron Hillegass 是 *iOS Programming: The Big Nerd Ranch Guide, Second Edition* 一书的合著者,也是 *Cocoa Programming for Mac OS X* 一书(不久将出现该书的第 4 版)的作者。Aaron 的图书在 Mac 开发者圈子中享有盛誉,并且是在 Cocoa 开发者邮件列表上看到的推荐次数最多的图书。要学习关于 Xcode 的更多知识, Fritz Anderson 编写的 *Xcode 4 Unleashed* 一书 (Sams Publishing 出版) 是最佳的选择。

注意:

市场上还有许多由其他出版社出版的其他类图书,包括 Dave Mark、Jack Nutting 和 Jeff LaMarche 编写的畅销书 *Beginning iPhone 4 Development* (Apress, 2011 年)。如果你对程序设计一无所知,那么另一本值得选择的图书是 Tim Isted 编写的 *Beginning Mac Programming* (Pragmatic Programmers, 2011 年)。不要把自己局限于某一本书或者某一家出版社。就像可以通过与不同的开发者交流来学习许多知识一样,从市场上的其他图书中也可以学到许多技巧与提示。

要真正掌握 Mac 开发,需要查看多种材料:图书、博客、邮件列表、Apple 自己的文档,最好是参加专题讨论会。如果有机会参加 WWDC,就会知道我说的是。在这些讨论会上要花时间与其他开发者交谈,在 WWDC 上,则要与 Apple 的工程师交谈,如果你是一名严肃的开发者,这样做是非常值得的。

本书组织结构

对于新的 iOS 开发者面临的最常见的问题,本书提供了单任务式的秘诀:布置界面元素、响应用户、访问本地数据源和连接到 Internet。每一章都把相关的任务组织在一起,允许直接跳转到所寻找的解决方案上,而不必决定哪个类或框架最匹配所讨论的问题。

本书提供了剪切和粘贴的便利,这意味着可以自由地把本书秘诀中的源代码重用在自己的应用程序中,然后调整代码以适应应用程序的需要。

下面简要描述了本书各章的内容。

¹ 要从头开始学习 Cocoa, 可以参阅 *Cocoa Fundamentals Guide* (<http://developer.apple.com/mac/library/documentation/Cocoa/Conceptual/CocoaFundamentals/CocoaFundamentals.pdf>); 要学习 Xcode, 可以参阅 *A Tour of Xcode* (http://developer.apple.com/mac/library/documentation/DeveloperTools/Conceptual/A_Tour_of_Xcode/A_Tour_of_Xcode.pdf)。

² Big Nerd Ranch 公司: www.bignerdranch.com。

- **第 1 章，“特定于设备的开发”**，每个 iOS 设备都代表独特、共享、短暂和持久属性的融合。这些属性包括设备的当前物理方位、它的型号名称、它的电池状态以及它对机载硬件的访问。本章从设备的构建配置到它的活动机载传感器进行了广泛探讨，并且提供了一些秘诀，用于返回关于正在使用的单元的各类信息项。
- **第 2 章，“文档和数据共享”**，在 iOS 下，应用程序可以使用系统提供的多个特性共享信息和数据，以及把控制从一个应用程序转移给另一个应用程序。本章介绍可以在应用程序之间集成文档和进行数据共享的方式。你将看到如何把这些特性添加到自己的应用程序中，并且聪明地使用它们，使应用程序成为 iOS 生态系统中具有协作精神的一员。
- **第 3 章，“Core Text”**，本章介绍了属性化文本处理，并且探索了如何将文本特性构建到应用程序中。你将学到把属性化字符串添加到公共 UIKit 元素中，如何创建 Core Text 强化的视图，以及如何为自由的文本排版突破更大的界限。在学完本章后，你将发现 Core Text 带给 iOS 的强大能力。
- **第 4 章，“几何学”**，尽管与 Core Animation 或 Open GL 相比，UIKit 不太需要应用数学，但是在处理贝塞尔路径和视图变换时，几何学仍然扮演着重要的作用。为什么需要几何学呢？它有助于以非标准的方式操纵视图，包括沿着自定义的路径布置文本，以及沿着路径运行动画。如果你在一提到贝塞尔曲线、凸包和样条时就目光呆滞，本章可以帮助你澄清这些术语，使你能够向工具箱中添加一些功能强大的自定义选项。
- **第 5 章，“联网”**，Apple 以各种网络计算及其支持技术给 iOS 打下了坚实的基础。本书姊妹篇《iOS 核心开发手册（第 4 版）》中有关联网的章节介绍了网络状态检查、同步和异步下载、JSON 和 XML 解析。本书中的这一章继续讨论这个主题，并且介绍了更高级的技术。其中包括身份验证质询、使用系统密钥链（keychain）、处理 OAuth 等。这里介绍了一些方便的方法，应该可以帮助你从事开发工作。
- **第 6 章，“图像”**，图像是抽象的表示，存储构成图片的数据。本章介绍了 Cocoa Touch 图像，特别是 UIImage 类，并且讲述了在 iOS 上处理图像数据所需的全部专业知识。在本章中，你将学习如何在应用程序中加载、存储和修改图像数据。你将发现如何处理图像数据以创建特殊效果，如何逐个字节地访问图像，等等。
- **第 7 章，“照相机”**，照相机把图像抬高到了下一个层次。它们使你能够把实时馈送和用户指示的快照集成到应用程序中，并且提供源于现实世界的原始数据。在本章中将会学习图像捕获。你将发现如何使用 Apple 提供的类来获取图片，以及如何从零开始操纵图像。你将学习控制图像元数据，以及如何把实时馈送与高级过滤集成起来。本章从硬件的角度重点介绍了图像捕获。无论你是在打开照相机闪光灯，还是在检测面孔，本章都会介绍 iOS 图像捕获技术的方方面面的知识。
- **第 8 章，“音频”**，iOS 设备是一个媒体大师。其内置的 iPod 特性可以专业地处理音频

和视频。iOS SDK 向开发者展示了这种功能。一套丰富的类通过回放、搜索和录制简化了媒体处理。本章介绍了一些秘诀，使用那些类操纵音频、把媒体展示给用户，以及允许用户与媒体交互。你将看到如何构建音频播放器和录音机，还将发现如何浏览 iPod 库以及如何选择要播放的项目。

- **第 9 章，“连接到 Address Book”**，本章将介绍 Address Book 并将演示如何在应用程序中使用它的框架。你将学到如何访问各个联系人的信息，如何修改和更新联系人信息，以及如何使用谓词只查找感兴趣的联系人。本章还将介绍 GUI 类，它们提供了交互式解决方案，用于挑选、查看和修改联系人。
- **第 10 章，“位置”**，快速计算正变得与计算的方式和计算的内容同样重要。iOS 终日忙个不停，整天都与它的用户一起旅行。Core Location 给 iOS 注入了按需进行地理定位的功能。MapKit 添加了交互式的应用程序内的地图绘制功能，使用户能够查看和操纵加注释的地图。利用 Core Location 和 MapKit，可以开发应用程序，帮助用户与朋友会面、搜索本地资源，或者提供基于位置的个人信息流。本章介绍了这些可以感知位置的框架，并且说明了如何把它们集成进 iOS 应用程序中。
- **第 11 章，“GameKit”**，本章介绍了可以通过 GameKit 创建相联系的游戏玩法的多种方式。GameKit 提供了一些特性，使应用程序能够超越单玩家/单设备场景，并过渡到使用 Game Center（游戏中心）和设备与设备之间的联网。Apple 的 Game Center 添加了一种集中式服务，使游戏能够提供共享的排行榜和基于 Internet 的配对。GameKit 还为对等连接提供了即席的联网解决方案。
- **第 12 章，“StoreKit”**，StoreKit 提供了应用程序中的采购功能，可以把它集成进你的软件中。利用 StoreKit，最终用户可以使用他们的 iTunes 凭证从应用程序内购买可以解锁的特性、媒体订阅或可消费的资产，比如鱼食或阳光。本章介绍了 StoreKit，并且说明了如何使用 StoreKit API 为用户创建采购选项。
- **第 13 章，“推送通知”**，当脱离设备的服务需要直接与用户通信时，推送通知提供了一种解决方案。就像本地通知允许应用程序在预定的时间联系用户一样，推送通知可以从基于 Web 的系统递送消息。推送通知可以让设备显示一条提醒、播放一段自定义的声音，或者更新应用程序图标。脱离设备的服务可以用这种方式联系基于 iOS 的客户，使他们能够知道新的数据或更新。本章将介绍你需要知道的关于推送通知的所有基本的知识。

关于示例代码

出于教学的目的，本书的示例代码使用单个 main.m 文件。人们通常不这样开发 iPhone 或 Cocoa 应用程序，或者诚实地讲，不应该这样开发它们，但它提供了一种极佳的方式来表达一个好主意。当读者必须同时查看 5 个、7 个或 9 个单独的文件时，将很难讲述一个故事。提供单个

文件可以把注意力集中在那个故事上，允许在单个代码块中访问那个主意。

这些示例并不打算作为独立的应用程序。它们只是为了演示单个秘诀和单个主意。一个具有中心表达方式的 `main.m` 文件可以在一个位置呈现实现故事。读者可以研究这些集中的主意，并且使用标准的文件结构和布局把它们转换成正常的应用程序结构。本书中的表达方式不会以标准的日常最佳实践方法产生代码。作为替代，它会反映一种教学方法，提供可以根据需要纳入到工作中的简明的解决方案。

相比之下，在 Apple 的标准示例代码中，必须梳理许多文件来构建将要演示的概念的心智模型（`mental model`）。这些示例被构建为完整的应用程序，它们所执行的任务通常与你需要解决的问题相关，但并非是必要的。只找出那些相关的部分就要做许多的工作，可能会导致得不偿失。

在本书中，将会发现这种用一个文件讲故事的规则的例外情况：当类实现就是秘诀时，`Cookbook` 将提供标准的类和头文件。一些秘诀不是为了强调某种技巧，而是提供这些类和类别（即对预先存在的类（而不是新类）的扩展）。对于这些秘诀，除了封闭故事余下部分的骨架式 `main.m` 文件之外，还要寻找单独的 `.m` 和 `.h` 文件。

一般来讲，本书中的示例都使用单个应用程序标识符：`com.sadun.helloworld`。本书使用一个标识符来避免数十个示例同时塞满 iOS 设备。每个示例将替换前一个示例，确保主屏幕保持相对整洁。如果想同时安装多个示例，只需简单地编辑标识符，添加独特的后缀，比如 `com.sadun.helloworld.table-edits`。也可以编辑自定义的显示名称，使应用程序看上去有所区别。团队配置文件（`Team Provisioning Profile`）将匹配每个应用程序标识符，包括 `com.sadun.helloworld`。这允许把编译过的代码安装到设备上，而不必更改标识符，只需确保在每个项目的构建设置中更新签名身份即可。

获取示例代码

在开源 GitHub 托管站点上的 github.com/erica/iOS-6-Cookbook 中可以找到本书的源代码。其中，可以找到按章划分的源代码集合，它们提供了本书中所介绍材料的工作示例。秘诀是按书中那样编号的。例如，第 5 章中的秘诀 6 出现在 06 子文件夹中的 C05 文件夹中。

任何编号为 00 或者带有后缀（比如 05b 或 02c）的项目都指示用于创建大量文本和图形的材料。通常，我会删除这些额外的项目。本书原稿的早期读者要求我在这个版本中包括进它们。你将发现其中有 6 个左右的额外示例散布在代码库周围。

如果你感觉直接使用 git 不那么得心应手，GitHub 提供了一个下载按钮。在编写本书时，该按钮位于站点主页的右边，大约在首页下面一半的位置。它使你能够将整个代码库作一个 ZIP 存档文件或压缩包下载。

投稿

示例代码永远不是一个固定的目标。随着 Apple 更新其 SDK 和 Cocoa Touch 库，它也会继续演化。参与进来吧！可以通过建议错误修正方式和校正措施以及扩展代码介入这个过程。GitHub 允许分解代码库，并利用自己的调整 and 特性扩充它们，并把它们共享回主代码库中。如果提出新的主意或方法，请让我知道。我的团队和我都很高兴在代码库和本书的下一个版本中吸纳非常好的建议。

获取 git

可以使用 git 版本控制系统下载本书的源代码。在 <http://code.google.com/p/git-osx-installer> 上提供了 git 的 OS X 实现。OS X git 实现包括命令行和 GUI 解决方案，因此要搜索最适合开发需求的版本。

获取 GitHub

GitHub (<http://github.com>) 是最大的 git 托管站点，具有超过 15 万个公共代码库。它既为公共项目提供了免费托管，也为私有项目提供了付费选项。利用一个自定义的 Web 界面，其中包括 wiki (维基) 托管、问题跟踪以及对项目开发者的社交网络的强调，很容易发现新代码或者在现有库上开展合作。可以在它们的网站上注册一个免费账户，这样就能够复制和修改 Cookbook 代码库，或者创建自己的开源 iOS 项目以与其他人共享。

联系作者

如果你对本书有任何建议或问题，请给我发送电子邮件，地址是：erica@ericasadun.com，或者访问 GitHub 代码库，并在那里联系我。

目录

第1章 特定于设备的开发	1	1.13.4 添加显示器链接	31
1.1 访问基本的设备信息	1	1.13.5 过扫描补偿	31
1.2 添加设备能力限制	2	1.13.6 VIDEOkit	31
1.2.1 用户权限描述	3	1.14 跟踪用户	35
1.2.2 其他常用的 Info.plist 键	4	1.15 还有一件事: 检查可用的磁盘空间	36
1.3 秘诀: 检查设备接近度和电池状态	4	1.16 小结	37
1.3.1 启用和禁用接近度传感器	5		
1.3.2 监测电池状态	5	第2章 文档和数据共享	39
1.3.3 检测 Retina 支持	8	2.1 秘诀: 处理统一类型标识符	39
1.4 秘诀: 取回额外的设备信息	8	2.1.1 通过文件扩展名确定 UTI	40
1.5 秘诀: 使用加速能力“向上”定位	10	2.1.2 从 UTI 转向扩展名或 MIME 类型	41
1.6 处理基本的方向	12	2.1.3 测试顺应性	42
1.7 同步获取当前的加速计角度	13	2.1.4 获取顺应性列表	43
1.7.1 通过加速计计算方向	14	2.2 秘诀: 访问系统粘贴板	45
1.7.2 计算相对角度	14	2.2.1 存储数据	46
1.8 使用加速度移动屏幕上的对象	15	2.2.2 存储公共类型	46
1.9 秘诀: 基于加速计的滚动视图	18	2.2.3 获取数据	47
1.10 秘诀: Core Motion 基础	21	2.2.4 被动更新粘贴板	47
1.10.1 测试传感器	22	2.3 秘诀: 监测 Documents 文件夹	48
1.10.2 处理程序块	22	2.3.1 支持文档文件共享	48
1.11 秘诀: 获取和使用设备姿势	26	2.3.2 用户控制	49
1.12 使用运动事件检测晃动	27	2.3.3 Xcode 访问	49
1.13 使用外部屏幕	29	2.3.4 扫描新文档	50
1.13.1 检测屏幕	29	2.4 秘诀: 展示活动视图控制器	54
1.13.2 获取屏幕分辨率	30	2.4.1 展示活动视图控制器	55
1.13.3 设置 Video Out	30		