

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 核心开发手册

(第 4 版)

The  
Core iOS 6  
Developer's Cookbook

[美] Erica Sadun 著  
陈宗斌 译

人民邮电出版社  
POSTS & TELECOM PRESS



014006017

PEARSON

TN929. 53  
544

# iOS 6 核心开发手册

(第4版)



The  
Core iOS 6  
Developer's Cookbook

[美] Erica Sadun 著  
陈宗斌 译

TN929.5  
544



北航

C1693077

人民邮电出版社

北京



710800:10

## 图书在版编目 (CIP) 数据

iOS 6核心开发手册：第4版 / (美) 萨顿  
 (Sadun, E.) 著；陈宗斌译. — 北京：人民邮电出版社  
 , 2013. 10  
 ISBN 978-7-115-32876-2

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(2013)第191101号

## 版 权 声 明

Erica Sadun: The Core iOS 6 Developer's Cookbook (4<sup>th</sup> Edition)

Copyright © 2013 Pearson Education, Inc.

ISBN: 0321884213

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 北京市崇文区夕照寺街14号
  - 邮编 100061 电子邮件 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：32
  - 字数：674千字 2013年10月第1版
  - 印数：1-3000册 2013年10月北京第1次印刷
  - 著作权合同登记号 图字：01-2013-5715号
- 

定价：79.00元

读者服务热线：(010)67132692 印装质量热线：(010)67129223

反盗版热线：(010)67171154

# 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，以苹果最新发布的 iOS 6 为基础编写而成。

本书通过源自真实世界的案例和代码解决方案讲解了 iPhone 和 iPad 应用开发过程中用到的所有知识，其内容涵盖了 iPhone、iPad 应用开发的核心技术要点，以及 iOS 开发在真实商业开发中所需要的所有主题。

本书适合对 iOS 开发感兴趣的人员阅读。

## 关于作者

**E**rica 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 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 iOS 自带的数十种应用程序的编写者，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和策划全球控制，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书。他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“真实鱈鱼”<sup>1</sup>吸引作者专注于图书的主题并且不脱离现实。这样的反复鞭挞得以使本书在最终期限前问世，并且使书中的内容非常有吸引力。

这里还要感谢 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 产品团队深表谢意，特别感谢 Kristy Hart、Jovana San Nicolas-Shirley、Keith Cline、Larry Sweazy、Debbie Williams、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。

另外，还要感谢我多年的代理人 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 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、Oliver Drobnik、Rod Strougo、Kevin McAllister、Jay Abbott、Tim Grant Davies、Maurice Sharp、Chris Samuels、Chris Greening、Jonathan Willing、Landon Fuller、Jeremy Tregunna、Wil Macaulay、Stefan Hafenegger、Scott Yelich、chrallalinder、John Varghese、Andrea Fanfani、J. Roman、jtbands、Artissimo、Aaron

---

<sup>1</sup> 在本书的开发和制作过程中并没有伤害真实的或想象的鱈鱼，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。

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](http://irc.saurik.com) 和 [irc.freenode.net](http://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 迎阅读新的 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。本书提供了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,还提供了处理图形、触摸和视图以创建移动应用程序所需的秘诀。

第 2 卷(The Advanced iOS 6 Developer's Cookbook)关注的是公共框架,比如 Store Kit、Game Kit 和 Core Location。它有助于构建利用这些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 开发并且寻求应对专业领域的实用知识的读者。

最后,还有一本图书是 *Learning iOS 6: A Hands-on Guide to the Fundamentals of iOS Programming*, 其中包含了大部分教程材料,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,读者可以在 GitHub 上找到示例代码。在 <https://github.com/eric/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](mailto:erica@ericasadun.com), 我提前表示感谢。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，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。

### 大学生计划

Apple 为学生和教师提供了一个 University Program（大学计划）。如果你是一名计算机科学的学生，并且学习的是大学级别的课程，就要与教师核实一下，看看你所在的学校是否是 University Program 的一部分。有关 iPhone Developer University Program（iPhone 开发者大学计划）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内存。
- 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要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，需要具备 Objective-C 2.0 的知识。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。如果你利用 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。尽管字母表是从字母 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），余下的就应该简单了。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。

### 3 前言

图 0-1 所示的流程图显示了 Pearson Education 提供的关键图书，它们可以提供成为熟练的 iOS 开发者所需要的培训。

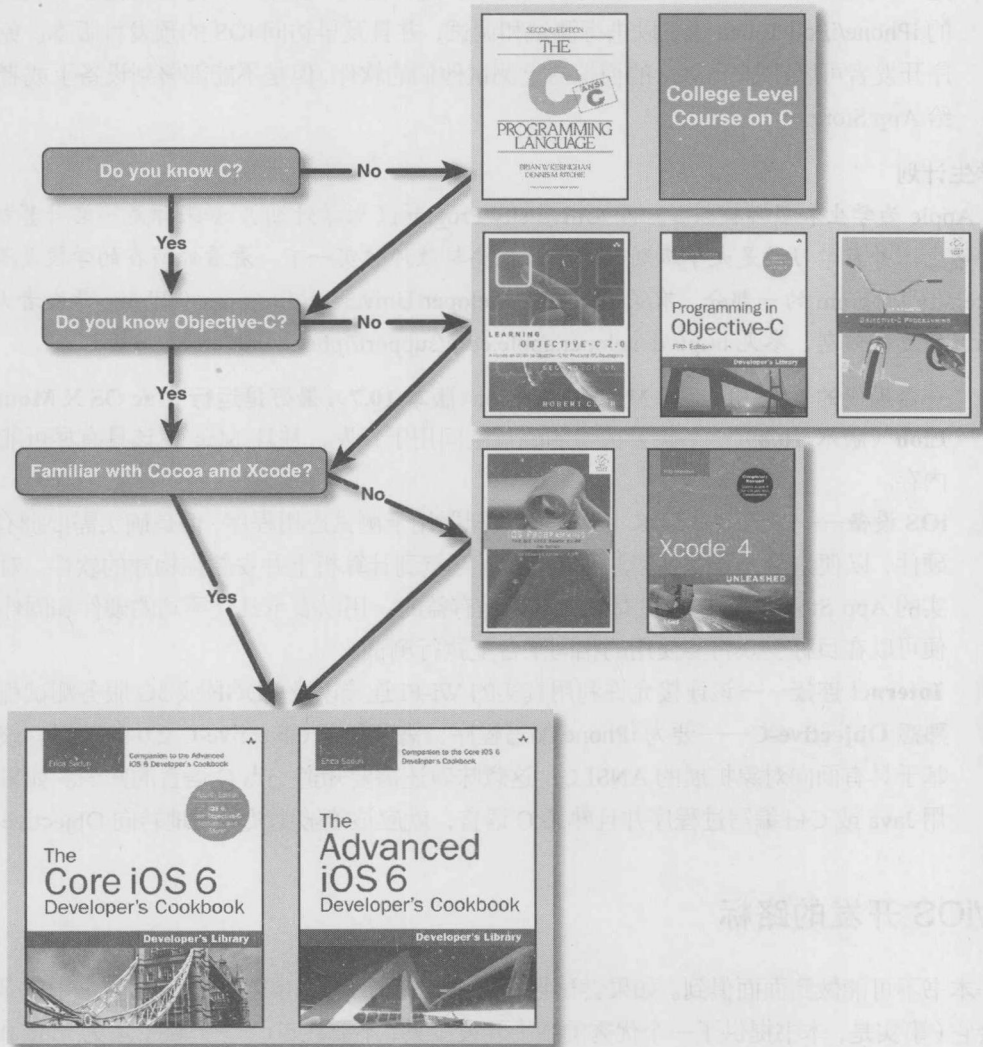


图 0-1 成为 iOS 开发者的路标

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 2.0: A Hands-on Guide to Objective-C for Mac and iOS Developers* 一书 (Addison-Wesley, 2012 年)
- Stephen Kochan 编写的 *Programming in Objective-C, Fifth 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<sup>1</sup>,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<sup>2</sup>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图书,包括 Dave Mark、Jack Nutting 和 Jeff LaMarche 编写的最畅销书 *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,就会知道我说的是什么。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

<sup>1</sup> 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](http://developer.apple.com/mac/library/documentation/DeveloperTools/Conceptual/A_Tour_of_Xcode/A_Tour_of_Xcode.pdf))。

<sup>2</sup> Big Nerd Ranch 公司: [www.bignerdranch.com](http://www.bignerdranch.com)。

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。

- **第 1 章“姿势和触摸”**：在 iOS 上，触摸为用户向应用程序传达其意图提供了最重要的方式。触摸并不仅限于按下按钮和键盘交互。本章介绍了直接操作界面、Multi-Touch（多触摸）等。你将看到如何创建用户可以在屏幕上四处拖曳的视图，并且学习如何区分和解释姿势，以及如何创建自定义的姿势识别器。
- **第 2 章“构建和使用控件”**：把控件带到下一个层次。本章介绍关于控件的工作原理需要知道的一切知识。你将发现如何以多种方式构建和自定义控件。从单调到模糊，本章将介绍一套可以在程序中重用的控件秘诀。
- **第 3 章“提醒用户”**：iOS 利用许多方式给用户提示，从弹出式对话框和进度条到本地通知、弹出信息和音频铃声。第 3 章说明了如何在应用程序中构建这些指示，并且扩展用户提醒方法库。它引入了一些标准方式来使用这些类，并且提供了一些解决方案，允许精心设计线性的程序，而无须进行显式的回调。
- **第 4 章“组装视图和动画”**：UIView 类及其子类可以填充 iOS 设备屏幕。本章将从头开始介绍视图，并将深入讨论视图秘诀，探索获取和操纵视图对象的方式，以及为其制作动画。你将学习如何构建、检查和分解视图层次结构，并将了解视图如何协同工作。你还将发现几何学在创建视图并将其放入界面过程中所起的作用，以及了解如何制作视图的动画，使之在屏幕上移动和变形。
- **第 5 章“视图约束”**：iOS 6 SDK（Software Development Kit，软件开发工具包）彻底改变了视图布局。Apple 的布局特性打算用于使生活更轻松，并且使界面更一致。在具有不同屏幕大小的相同设备家族（比如 iPhone 4S 和 iPhone 5）之间工作时，这一点特别重要。本章将介绍代码级约束开发。你将发现如何创建屏幕上的对象之间的关系，并且指定 iOS 自动排列视图的方式。其结果是一组可以适应屏幕几何学的健壮规则。
- **第 6 章“文本输入”**：本章介绍可以支持广泛解决方案的文本秘诀。你将学习如何控制键盘、使屏幕上的元素“认识文本”、扫描文本、格式化文本等。从文本框和文本视图到 iOS 的内联拼写检查程序，本章介绍了在应用程序中处理 iOS 文本需要知道的一切知识。
- **第 7 章“使用视图控制器”**：在本章中将学习多个不同的视图控制器类，它们允许放大并排序用户交互的虚拟空间，学习一些基本的秘诀，它们涵盖了页面视图控制器、拆分视图控制器、导航控制器等。
- **第 8 章“常用控制器”**：iOS SDK 具有大量系统提供的控制器，可以在日常开发任务中

使用它们。本章将介绍其中一些最流行的控制器。你将学习从照片图库中选择图像、抓取照片，以及录制和编辑视频，还将发现怎样允许用户创建电子邮件和文本消息，以及如何将更新发布给社交服务，比如 Twitter 和 Facebook。

- **第 9 章“可访问性”**：本章简要描述了 VoiceOver 可访问性，用以把你的受众扩展到尽可能最广泛的用户。你将学习如何给应用程序添加可访问性标签和提示，并在模拟器中和 iOS 设备上测试这些特性。
- **第 10 章“创建和管理表格视图”**：表格提供了一个滚动交互类，在小型设备和较大的平板电脑上都工作得特别好。由于表格的简易性，许多 iOS 应用程序都以表格为中心。本章介绍了表格。它解释了表格的工作原理，哪些类型的表格可供开发者使用，以及如何应用程序中利用表格特性。
- **第 11 章“集合视图”**：集合视图使用许多与表格相同的概念，但是提供了更强大的能力和更大的灵活性。本章将介绍开始时需要掌握的所有基础知识。也许将学习如何创建侧面滚动列表、网格、独特的布局（比如圆形）等，以及学习通过布局规范集成可视化效果，并且在滚动后使项目贴紧某个位置，还将发现如何利用内置的动画支持来创建尽可能吸引人的交互。
- **第 12 章“Core Data 体验”**：Core Data（核心数据）提供了可以从应用程序查询和更新的托管数据存储。它提供了基于 Cocoa Touch 的对象界面，并把 SQL 查询中的关系数据管理引入到 iOS 开发的 Objective-C 世界中。本章介绍了 Core Data。它提供了恰好足够的秘诀让你体验该技术，并且为进一步学习 Core Data 提供了一个起点。你将学习如何设计托管数据库存储、添加和删除数据，以及通过代码查询该数据，并把它集成进 UIKit 表格视图和集合视图中。
- **第 13 章“网络基础”**：作为一种基于 Internet 的设备，iOS 特别适合于订阅基于 Web 的服务。Apple 给平台提供了稳固的基础，它带有各种网络计算服务及其支持的技术。本章介绍了用于网络计算的常用技术，并且提供了用于简化日常任务的秘诀。你将学习网络可达性、同步和异步下载、使用操作队列、使用 iPhone 的安全钥匙链来应对身份验证挑战、XML 解析、JSON 串行化等。
- **附录 A“Objective-C 文字常量”**：本附录介绍了一些新的 Objective-C 构造，用于指定数字、数组和字典。

## 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堵塞 iOS 设备。每个示例替换前一个示例，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](https://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。秘诀是按书中所述编号的。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 或者具有后缀（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。例如，第 10 章中的 00 - Cell Types 项目帮助构建图 10-2，显示系统提供的表格视图的单元格样式。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。

## 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，并利用自己的调整和特性扩充它们，以及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](mailto:erica@ericasadun.com)，或者访问 GitHub 代码库，并在那里联系我。



# 目录

第 1 章 姿势和触摸 .....	1	第 2 章 构建和使用控件 .....	45
1.1 触摸 .....	1	2.1 UIControl 类 .....	45
1.1.1 阶段 .....	2	2.1.1 目标—动作 .....	45
1.1.2 触摸和响应者方法 .....	3	2.1.2 控件的种类 .....	46
1.1.3 触摸视图 .....	3	2.1.3 控件事件 .....	47
1.1.4 多触摸 .....	4	2.2 按钮 .....	48
1.1.5 姿势识别器 .....	4	2.3 Interface Builder 中的按钮 .....	50
1.2 秘诀：添加简单的直接操作 界面 .....	5	2.4 秘诀：构建按钮 .....	51
1.3 秘诀：添加平移姿势识别器 .....	7	2.4.1 多行按钮文本 .....	53
1.4 秘诀：同时使用多个姿势识别器 .....	9	2.4.2 给按钮添加动画式元素 .....	54
1.5 秘诀：约束移动 .....	12	2.4.3 给按钮添加额外的状态 .....	54
1.6 秘诀：测试触摸 .....	14	2.5 秘诀：制作按钮响应的动画 .....	54
1.7 秘诀：针对位图进行测试 .....	16	2.6 秘诀：添加具有自定义指针的滑块 .....	56
1.8 秘诀：在屏幕上绘制触摸 .....	18	2.6.1 自定义 UISlider .....	57
1.9 秘诀：平滑绘图 .....	20	2.6.2 提高效率 .....	58
1.10 秘诀：使用多触摸交互 .....	23	2.7 外观代理 .....	61
1.11 秘诀：检测圆形 .....	27	2.8 秘诀：创建可点按两次的分段控件 .....	64
1.12 创建自定义的姿势识别器 .....	31	2.8.1 第二次点按反馈 .....	65
1.13 秘诀：从滚动视图中拖动 .....	33	2.8.2 控件和属性 .....	65
1.14 秘诀：实时触摸反馈 .....	37	2.9 处理选项开关和步进器 .....	67
1.14.1 启用触摸反馈 .....	37	2.10 子类化 UIControl .....	68
1.14.2 截取和转发触摸事件 .....	38	2.10.1 创建控件 .....	69
1.14.3 实现 TOUCHkit 叠加视图 .....	39	2.10.2 跟踪触摸 .....	69
1.15 秘诀：给视图添加菜单 .....	42	2.10.3 调度事件 .....	69
1.16 小结 .....	43	2.11 秘诀：构建星星滑块 .....	72
		2.12 构建触摸式转盘 .....	75