

[美] Rob Napier Mugunth Kumar 著
陈晓亮 武海峰 邓 强 周庆成 译

iOS 6 编程实战



iOS 6 Programming Pushing the Limits

- 深度揭秘iOS 6高级特性与开发技巧
- 发掘最佳实践，借鉴开发经验 真正掌握iOS开发
- 移动开发进阶必备，在最新平

```
/* Begin PBXFrameworksBuildPhase section */
7213FCFE15FF5E6E004A58FF /* Frameworks */ = {
isa = PBXFrameworksBuildPhase;
buildActionMask = 2147483647;
files = (
7213FD0615FF5E6E004A58FF /* UIKit.framework in Frameworks */,
7213FD0815FF5E6E004A58FF /* Foundation.framework in Frameworks */,
7213FD0A15FF5E6E004A58FF /* CoreGraphics.framework in Frameworks */
);
runOnlyForDeploymentPostprocessing = 0;
/* End PBXFrameworksBuildPhase section */
```



人民邮电出版社
POSTS & TELECOM PRESS

iOS 6 编程实战



iOS 6 Programming
Pushing the Limits

[美] Rob Napier Mugunth Kumar 著

陈晓亮 武海峰 邓强 周庆成 译

```
/* Begin PBXFrameworksBuildPhase section */
7213FCFE15FF5E6E004A58FF /* Frameworks */ = {
isa = PBXFrameworksBuildPhase;
buildActionMask = 2147483647;
files = (
7213ED0615FF5E6E004A58FF /* UIKit.framework in Fram
7213FD0815FF5E6E004A58FF /* Foundation.framework
7213FD0A15FF5E6E004A58FF /* CoreGraphics.framework
);
runOnlyForDesktopPostprocessing = 0;
/* End PBXFrameworksBuildPhase section */
```



人民邮电出版社

北京

图书在版编目 (CIP) 数据

iOS 6编程实战 / (美) 内皮尔 (Napier, R.), (美) 库马尔 (Kumar, M.) 著; 陈晓亮等译. — 北京: 人民邮电出版社, 2013. 4

(图灵程序设计丛书)

书名原文: iOS 6 Programming Pushing the Limits
ISBN 978-7-115-31218-1

I. ①i… II. ①内… ②库… ③陈… III. ①移动电话机—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆CIP数据核字(2013)第044937号

内 容 提 要

iOS 平台不断发展变化, 本书基于此进行了全面更新, 分 4 部分深入介绍 iOS 6 开发。第一部分主要介绍 iOS 6 新功能。第二部分带你熟练掌握常用工具(含表视图通知和动画图层), 内容涉及 Cocoa 设计模式与苹果利用其解决问题的思路、通过 Objective-C 的 ARC 管理内存、正确使用表视图(含解决无限滚动等问题)、集合视图与自动布局、自定义绘图、视图动画与 Core Animation 框架、错误处理、位置服务。第三部分介绍特定情况下要使用的工具与相关技术, 涉及表视图、多任务、REST 式服务、安全(用钥匙串、证书和密码保护应用与用户数据)、基于硬件和平台调整应用、轻松打造国际化与本地化、应用内购买、用 Instruments 进行性能调优。第四部分介绍了更高级的开发工具, 涵盖故事板与自定义切换效果、KVO、块与函数式编程、离线支持、云、文本布局、Core Foundation, 并最后深度解析了 Objective-C。

本书适合具有一定 iOS 开发经验, 致力于深入了解 iOS 开发、最佳实践以及他人实际开发经验的人学习参考。

图灵程序设计丛书 iOS 6编程实战

-
- ◆ 著 [美] Rob Napier Mugunth Kumar
 - 译 陈晓亮 武海峰 邓 强 周庆成
 - 责任编辑 毛倩倩
 - 执行编辑 李 静 刘美英
 - ◆ 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>
 - 北京天宇星印刷厂印刷
 - ◆ 开本: 800×1000 1/16
 - 印张: 28.25
 - 字数: 703千字 2013年4月第1版
 - 印数: 1-5 000册 2013年4月北京第1次印刷
 - 著作权合同登记号 图字: 01-2013-1018号

ISBN 978-7-115-31218-1

定价: 99.00元

读者服务热线: (010)51095186转604 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

All Rights Reserved. This translation published under license. Authorized translation from the English language edition, entitled *iOS 6 Programming Pushing the Limits*, ISBN 978-1-118-44995-0, by Rob Napier, Mugunth Kumar, Published by John Wiley & Sons. No part of this book may be reproduced in any form without the written permission of the original copyrights holder.

Simplified Chinese translation edition published by POSTS & TELECOM PRESS Copyright © 2013.

本书简体中文版由John Wiley & Sons, Inc.授权人民邮电出版社独家出版。

本书封底贴有John Wiley & Sons, Inc.激光防伪标签，无标签者不得销售。

版权所有，侵权必究。

献给Neverwood。感谢你的耐心。

——Rob Napier

献给我的母亲，是她塑造了第一个20年的我。

——Mugunth Kumar

译者序

能生活在这个时代，既不幸，又幸运。不幸的是，百年之后人类的生活将是现在的我们所无法想象的。幸运的是，世界在飞速变化，改变世界的产品在不断出现。今天的移动互联网就是改变人类生活的强大力量。

哪怕是 10 年前，正当互联网如火如荼时，我们也不会想到在今天智能移动设备会如此普及，人们能无时无刻地访问网络。而在所有的移动操作系统中，iOS 可能不是市场占有率最高的，却无疑是最优秀的。无论是在 UI 设计、交互设计还是应用质量上，iOS 应用总体上来讲都要优于其他移动平台的应用，这固然是因为 iOS 开发者整体上比较优秀，同样也和 iOS 本身的高质量 SDK 分不开。然而，要成为优秀的 iOS 开发者，却并不是容易的事。我一直认为苹果的官方文档是最好的学习资料，但是本书确实质量很高，两位作者凭借多年的实践经验，向我们展示了从初级到高级的各种 iOS 编程技巧和理念，相信会对读者有所助益。

全书的翻译由陈晓亮、武海峰、邓强、周庆成完成，由于时间仓促（真的是时间仓促，不是套话），尽管我们努力地统一术语，检查错误，但是肯定会有所疏漏，如果读者发现有错误，我们欢迎拍砖，同时也欢迎来信交流。

前言

苹果操作系统历来都是一个面向用户的版本，一个面向开发者的版本，这样交替着发布。iOS 6 更多关注用户，具有全新的地图界面和导航功能，社交网络特性更是随处可见。Pass Kit 更以一种全新的方式将 iPhone 与日常生活联系到了一起。增强的 Siri 功能使 iPhone 4S 更具吸引力。但愿用户能快一点升级。

这都是好消息，因为 iOS 6 同样也令开发人员兴奋不已。很多事物变得更加简单，集合视图与自动布局可以简化 UI 设计，UIKit 添加了对富文本的支持。当你的应用进入后台，状态保留可以更加容易记录它们的进度。很多过去复杂麻烦的东西开始变得简单了。

如果你准备好了去探索最新的苹果系统，从而把你的应用提升到新高度，那这本书就是你应该看的。

读者对象

这并不是一本入门书。其他一些书会教你 Objective-C 并一步步指导你学习 Interface Builder，不过本书假定你已经拥有一些 iOS 开发经验；可能是自学的，或者上过培训班。总之，你有能力写程序，甚至已经有应用上架了。如果你打算学习更深入的内容、最佳实践，以及作者源自真实工程的开发经验，那你就找对书了。

这本书并不是示例的简单堆砌，它包含大量代码，不过主线还是围绕如何设计、编写和维护优秀的 iOS 应用展开。本书会让你了解为什么这么做而不只是局限于怎么做。你会学到很多关于设计模式、编写可重用代码，以及语法与新框架的知识。

本书内容

iOS 平台总是向前发展，本书也一样。书中大部分示例需要至少 iOS 5 才能运行，有些需要 iOS 6。所有的示例都启用了自动引用计数、自动属性合成和对象字面量。除了很少几处外，本书不会讨论向后兼容。如果你的代码过于庞大必须要向后兼容，你可能知道如何处理。本书主旨是通过最好的特性来创造最佳应用。

虽然本书主要针对 iPhone 4、iPad 2 之后的设备，但大部分内容也适用于第一代 iPad、iPod touch、iPhone 3GS 以及 Apple TV。第 16 章还会专门讨论如何处理各个平台之间的差异。

新版内容

这一版涵盖了 iOS 6 的大部分新功能，包括集合视图和自动布局（第 7 章），还有一整章介绍自动

引用计数。本书新增一章讲解如何将数据上传到云端，介绍了一些第三方 BaaS（Backend as a Service，后端即服务）。本书还扩展了 iOS 相关的内容，主要涵盖改进后的 LLVM 编译器与调试器。为此，我们增加了第 2 章 LLVM 一节的篇幅，针对如何使用它写出下一代应用讲解了 LLVM 的重要特性。它还添加了一章（第 19 章）专门介绍 LLDB 调试，包括 Python 脚本、寄存器布局和读取寄存器等一些有助于解决调试难题的高级技巧。这一章还提到了 `otool`、`atos` 等可以辅助调试的命令行工具。最后，第 20 章展示了如何围绕 CPU 性能、内存占用、网络带宽以及电池使用进行优化。

即便你有本书的上一版，关于调试（第 19 章）、性能调优（第 20 章）以及集合视图和自动布局（第 7 章）的新内容也足够成为你买下这一版的理由了。

本书结构

iOS 提供了非常丰富的工具，既有 UIKit 这样的高层框架，也有 Core Text 这样的低层框架。有时候，同一个目标可以通过多种方式来达成。作为开发人员，如何找到最正确的工具呢？

本书既考虑了日常开发需求，也考虑了特定的用途，能够帮你作出正确的选择。学完本书，你会明白每个框架存在的价值、框架之间的相互关系，以及什么时候选择哪一个。最终，你会知道哪个框架最适合解决哪一类问题。

本书分 4 部分，从最常用的工具一直讲到最强大的工具。这一版新增的章会在前面用“更新”字样标识出来。

第一部分：新功能

如果你对 iOS 5 很熟悉，这一部分可以带你快速了解 iOS 6 的新功能。

- ❑（更新）第 1 章“全新的系统”。iOS 6 增加了大量新功能，在这里你将快速了解它们。
- ❑（更新）第 2 章“熟悉 Xcode 4 与 LLVM 编译器”。Xcode 是一个庞大的 IDE，很多功能并不显而易见。无论你是新手还是上一版的读者，这一章都会帮助你了解它强大（也有令人沮丧）的工具。

第二部分：熟能生巧

作为一名 iOS 开发人员，你应该掌握很多常用工具，比如表视图通知和动画图层。不过要想发挥它们的全部潜力，就要熟悉它们。在这一部分，我们将学到 Cocoa 开发的最佳实践。

- ❑ 第 3 章“Objective-C 基础知识”。如果你想在 Objective-C 开发方面更上一层楼，这一章将引导你学习一些改善应用设计、稳定性和可重用性的常用工具。
- ❑ 第 4 章“Cocoa 设计模式”。Cocoa 建立于大量常用的设计模式之上。这一章会介绍相关的设计模式和苹果用它们来解决问题的思路。
- ❑（新增）第 5 章“通过 Objective-C 的 ARC 管理内存”。自动引用计数根本改变了 iOS 开发者的工作方式。大部分转换也很容易，移植代码时可能会出现一些微妙的问题。本章讨论如何避开问题和充分利用这个强大的技术。

- ❑ 第 6 章“熟练使用表视图”。表视图可能是 iOS 中最复杂但也最常用的 UI 元素了。它们的设计非常简单优雅，但不了解原理却很难用好。这一章将为你讲清楚如何正确使用它们，以及如何解决无限滚动等问题。
- ❑ (新增) 第 7 章“集合视图与自动布局”。WWDC 2012 中展示的 UICollectionView 令世人瞩目。到目前为止，UITableView 仍然垄断着 iOS 应用的界面。不过当开发人员慢慢习惯新控件后，这个局面很可能在 iOS 6 时代被改变。加上 OS X 中基于约束（constraint-base）的布局，实现复杂而美观的界面突然变得简单起来。这一章的内容就是这些。
- ❑ 第 8 章“绘图功能”。很多新开发者都对自定义绘图退避三舍，但它却是快速创建美观用户界面的关键。这一章将探究 UIKit 和 Core Graphics 中有关绘图的功能，告诉大家怎么才能做到既快且美。
- ❑ 第 9 章“Core Animation”。iOS 设备对动画的支持是无与伦比的。借助强大的 GPU 和高度优化的 Core Animation，你可以创建直观又吸引人的界面。在这一章中，我们会介绍一些基础知识以及动画的原理。
- ❑ 第 10 章“错误处理”。你总想写出完美的代码，不过每每又事与愿违。如何应对意外是优秀应用与一般应用的分水岭。这一章会告诉你常用的错误处理技巧、如何记录错误，以及如何让代码在出错时表现得更加灵活。
- ❑ (新增) 第 11 章“位置服务”。许多应用都需要知道用户在什么地方。iOS 使一切变得简单，但也要警惕不要把电量耗尽。这一章我们讲一讲如何平衡功能与效率，同时给予用户最佳体验。

第三部分：选择工具

有些工具几乎时刻都要用，而有些工具则只有特定的时候才有用。在这一部分中，你会学到这些较为特殊的工具以及相关的技术。

- ❑ (新增) 第 12 章“表视图常用 UI 范例”。与集合视图一样令人欲罢不能的是表视图，它一直是 iOS 开发的中坚力量。这一章介绍如何使用高级表视图的无限滚动视图和其他高级主题。
- ❑ (更新) 第 13 章“多任务”。多任务是许多应用程序的重要部分，这一章将介绍如何在应用程序运行时或在后台时执行多任务。iOS 6 新的状态保留系统使这一切更加容易，这一章还会介绍一些常见用例。
- ❑ (更新) 第 14 章“REST 解惑”。基于 REST 的服务是现代应用程序的核心，这一章将教会你在 iOS 中最好地实现它们。
- ❑ (更新) 第 15 章“通过安全服务巩固系统安全”。用户安全和保护隐私永远是第一位的。这一章会介绍如何通过钥匙串、证书和密码保护应用和用户数据不会被盗用。
- ❑ 第 16 章“在多个苹果平台和苹果设备上运行应用”。iOS 家族人丁兴旺，不仅有了 iPod touch、iPhone、iPad、Apple TV，而且新机型仍会不断涌现。目前还无法一次编写随处运行。为了保证应用在任何平台上都表现卓越，本章将讨论如何基于硬件和平台调整应用。
- ❑ 第 17 章“国际化和本地化”。虽然你现在可能只想关注某个国家的市场，但让应用明天能够顺利走向世界也只需做一点点工作。本章会告诉你如何不影响当前开发，又能减少未来的麻烦和成本。

- ❑ 第 18 章“通过应用内购买提高销售业绩”。应用内购买对于很多程序员来说仍然是陌生领域。用户喜欢额外内容，而开发人员喜欢额外收入。这一章告诉你在应用中把这一功能变为现实的最佳方法。
- ❑ (新增) 第 19 章“调试”。要是每个应用第一次就能完美运行该有多好。幸运的是，Xcode 和 LLDB 提供了很多能帮助你抓住“狡猾”bug 的工具。你会学到很多高级的内容，了解实际开发中如何处理错误。
- ❑ (全新) 第 20 章“性能调优”。高性能可以让应用脱颖而出。优化 CPU 和内存性能非常重要，不过你也需要优化电池以及网络使用。苹果公司提供了 Instruments 这个强大的工具来解决这些问题。你会学到如何使用 Instruments 查找瓶颈，以及如何在找到问题后改善性能。

第四部分：超越极限

这一部分是全书最精彩的内容。你已经学到了基础知识，掌握了基本技能，现在该使用高级工具来超越极限了。这一部分将带你深入了解 iOS。

- ❑ (新增) 第 21 章“故事板及自定义切换效果”。故事板仍然会使一些熟悉 Nib 文件的开发人员感到费解。你在这里将会学到如何使用故事板来提升应用。
- ❑ 第 22 章“Cocoa 的大招：键值编码和观察”。苹果的许多强大框架都是依靠 KVO (Key-Value Observing) 来维护性能和灵活性的。你会学到如何利用灵活性和 KVO 的速度。
- ❑ 第 23 章“不同凡响：块和函数式编程”。许多开发人员依然对 Objective-C 的块感到好奇。块可以用来与苹果框架交互，不过也开启了编程新思路。接受一种新风格，并在你的下一个工程中将它的优势表现出来吧。
- ❑ 第 24 章“离线支持”。网络编程很困难，不过更难的是支持无缝离线功能。学习如何最佳地缓存数据并将其集成到你的网络引擎上是这一章的重点。
- ❑ (新增) 第 25 章“云端数据”。开发人员依然在适应新的 iCloud 服务。支持的话可以快速让你的应用程序引人注目。本章中你将学习这个强大的系统。
- ❑ (更新) 第 26 章“精妙的文本布局”。从 UIKit 到 Core Text，iOS 为显示文本提供了充足的功能。iOS 又在 UIKit 中添加了富文本支持。在这一章，你会学到如何使用属性化字符串、Web 视图，以及用 Core Text 来处理最棘手的布局问题。
- ❑ 第 27 章“创建 Core Foundation 应用”。说到 iOS 中最强大的框架，你能想到的 Core 框架可能会有 Core Graphics、Core Animation、Core Text，但它们都是基于 Core Foundation 框架的。在这一章中，我们学习如何使用 Core Foundation 数据类型，以便充分利用 iOS 提供的功能。
- ❑ 第 28 章“深度解析 Objective-C”。这一章致力于揭开 Objective-C 背后的秘密，包括如何使用 Objective-C 运行时直接动态地修改类和方法、如何通过 Objective-C 函数调用 C 方法，以及如何通过系统来扩展程序。

以上各章可以跳读，除了需要 Core Foundation 数据对象（特别是 Core Graphics、Core Animation 和 Core Text）的几章，其他章都是相互独立的。关于 Core Foundation 的内容，最终会归总到第 27 章“创建 Core Foundation 应用”。

阅读条件

本书所有示例都是用 Mac OS X 10.8 上的 Xcode 4.5 以及 iOS 6 开发的。你需要一个苹果开发人员账户来访问大部分工具和文档，并且需要一个开发人员许可证来运行 iOS 设备上的应用程序。对此，请参考 <http://developer.apple.com/programs/ios> 并注册账号。

本书中大部分示例可以在 Xcode 4.5 的 iOS 模拟器中运行。使用 iOS 模拟器就不需要苹果开发人员许可证了。

苹果文档

苹果公司在自己的网站上和 Xcode 中提供了大量文档。这些文档的 URL 地址变动很频繁而且非常长。本书会使用标题而不是 URL 来引用这些文档。如果想在 Xcode 中寻找文档，请按下 `Cmd-Option-?` 快捷键或点击 `Help->Documentation and API Reference`。在 Documentation Organizer 窗口中点击搜索图标，输入文档的标题，并从搜索结果中选择文档。可以参考图 1 中搜索 *Coding Guidelines for Cocoa* 的示例。

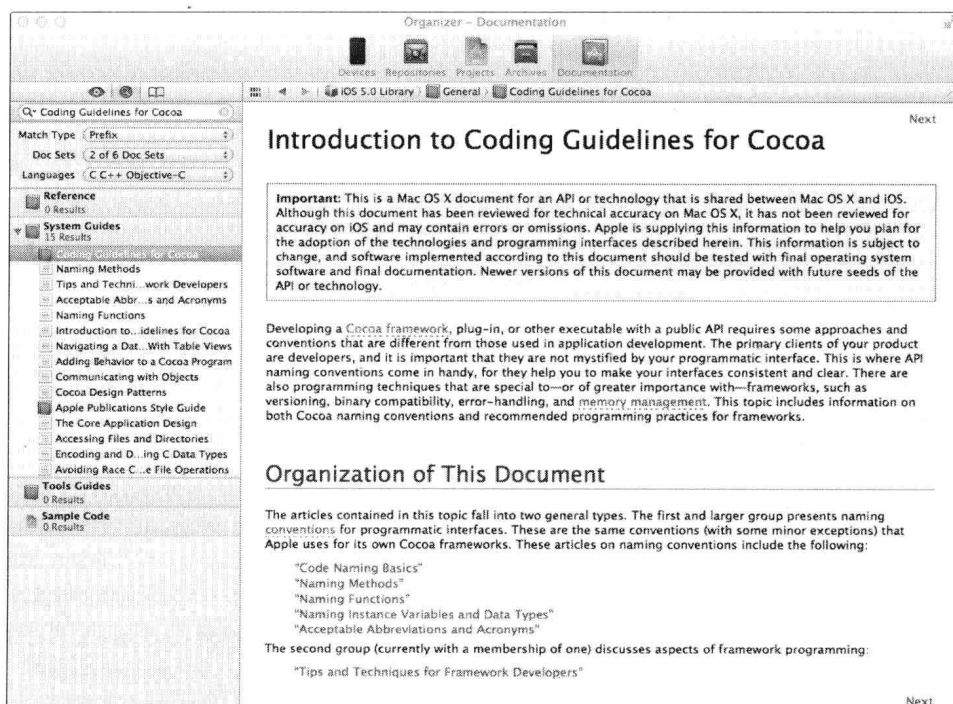


图 1 搜索 Coding Guidelines for Cocoa

如果想在苹果官方网站查找文档，可以访问 <http://developer.apple.com>，点击 Member Center，并登录。选择 iOS Dev Center，并在搜索框中输入文档的标题。

在线文档与 Xcode 文档是相同的。你可能会接收到 iOS 和 Mac 两个平台的结果，请阅读 iOS 版。很多 iOS 文档是 Mac 版的副本并且偶尔会包含 iOS 不支持的函数调用或常量。本书会告诉你哪些功能在 iOS 上可用。

源代码

在学习本书示例的时候，可以手工输入代码，也可以使用本书附带的源代码文件。本书所有的源代码可以在 <http://iosptl.com/code> 或 <http://www.wiley.com/go/ptl/ios6programming> 上下载得到。^①下载之后，在第 26 章文件夹 SimpleLayout 工程的 CoreTextLabel.m 文件中可以看到如下代码：

CoreTextLabel.m (SimpleLayout)

```
- (id)initWithFrame:(CGRect)frame {
    if ((self = [super initWithFrame:frame])) {
        CGAffineTransform
            transform = CGAffineTransformMakeScale(1, -1);
        CGAffineTransformTranslate(transform,
                                   0, -self.bounds.size.height);
        self.transform = transform;
        self.backgroundColor = [UIColor whiteColor];
    }
    return self;
}
```

本书中有些代码片段并不完整，其目的只是为了辅助上下文说明问题。要想查看完整代码，就需要参考下载到的源代码。

勘误

虽然我们已尽了最大努力，但错误在所难免。有些是输入错误，有些可能是我们理解上有偏差，有些可能是因为内容有变化。为了确保代码与时俱进，请参考 <http://iosptl.com/code> 上的最新版本，以及博客中的相关文章。任何问题都可以发送给 robnapier@gmail.com 或 contact@mk.sg。

致谢

在去年投入大量时间撰写针对 iOS 5 的本书上一版之后，今年又着手撰写针对 iOS 6 的这个新版本，对此 Rob 非常感谢自己家人的耐心。Mugunth 感谢父母和朋友的支持。感谢 Wiley 出版社，尤其是 Craig Smith 和 Ellie Scott，在他们不懈的支持、鼓励下本书才得以问世。感谢 Jay Thrash 热心的技术审校，还有 Tom Dinse 对文字的润色。感谢 WWDC 的苹果工程师们容忍我们无穷无尽的疑问、追寻，还有偶尔的抱怨。特别感谢史蒂夫·乔布斯，他毕生致力于改变世界。人们不会忘记他。

^① 也可以访问图灵社区本书页面下载，地址为：<http://www.ituring.com.cn/book/1053?q=ios6>。——编者注

目 录

第一部分 全新功能

第 1 章 全新的系统	2	2.6 Xcode 之核心: LLVM 编译器	14
1.1 新功能	2	2.6.1 Clang 编译器前端	15
1.1.1 集合视图	3	2.6.2 我是一个 Bug! 修复我	15
1.1.2 自动布局	3	2.7 LLVM 4 的新功能	15
1.1.3 自动引用计数	3	2.7.1 字面量	15
1.1.4 应用内购买的托管内容	4	2.7.2 字面量和可用性检查	18
1.1.5 社交框架	4	2.7.3 实例变量的自动合成	18
1.1.6 UI 状态保留	4	2.8 使用 Git 进行版本控制	19
1.1.7 其他新功能	4	2.8.1 内置的 Git 版本控制系统	19
1.2 小结	6	2.8.2 版本编辑器	19
1.3 扩展阅读	6	2.8.3 Git 最佳实践	19
第 2 章 熟悉 Xcode 4 和 LLVM 编译器	7	2.9 Scheme	20
2.1 了解用户界面	8	2.9.1 为什么使用 Scheme	20
2.1.1 标签式编辑器	9	2.9.2 建议用 Scheme 实现你的意图	21
2.1.2 按键绑定的变化	9	2.9.3 创建一个 Scheme	21
2.1.3 工程设置编辑器	10	2.9.4 共享你的 Scheme	22
2.1.4 内置的版本控制系统	10	2.10 为编译配置添加注释	22
2.1.5 工作区	10	2.10.1 创建 xcconfig 文件	22
2.2 单窗口工作方式	10	2.10.2 重构编译配置文件	23
2.3 导航面板	11	2.11 将代码转换为支持 ARC 的 Objective-C 代码	23
2.3.1 工程导航面板	12	2.12 Xcode 4 Organizer	25
2.3.2 符号导航面板	12	2.12.1 Automatic Device Provisioning	25
2.3.3 搜索导航面板	12	2.12.2 查看崩溃日志和控制台输出	25
2.3.4 问题导航面板	12	2.12.3 查看应用沙盒数据	26
2.3.5 调试导航面板	12	2.12.4 管理版本库	26
2.3.6 断点导航面板	12	2.12.5 管理应用程序归档	26
2.3.7 日志导航面板	13	2.12.6 查看 Objective-C 和 SDK 文档	26
2.4 辅助功能	13	2.13 小结	26
2.5 内置的 Interface Builder	13	2.14 扩展阅读	26
		2.14.1 苹果文档	26

2.14.2 WWDC 讲座	27
2.14.3 博客	27
2.14.4 网上资源	27
2.14.5 图书	27

第二部分 熟能生巧

第 3 章 Objective-C 基础知识

3.1 命名约定	30
3.2 自动引用计数	32
3.3 属性	35
3.3.1 属性修饰关键字	36
3.3.2 属性最佳实践	37
3.3.3 私有实例变量	37
3.4 存取器	37
3.5 分类和扩展	38
3.5.1 +load 方法	40
3.5.2 使用关联引用为分类添加数据	40
3.5.3 类扩展	42
3.6 正式协议和非正式协议	42
3.7 小结	43
3.8 扩展阅读	44
3.8.1 苹果文档	44
3.8.2 其他资源	44

第 4 章 Cocoa 设计模式

4.1 理解模型-视图-控制器	45
4.1.1 使用模型类	46
4.1.2 使用视图类	46
4.1.3 使用控制器类	47
4.2 理解委托和数据源	47
4.3 使用命令模式	48
4.3.1 使用目标-动作	48
4.3.2 使用方法签名和调用	49
4.3.3 使用蹦床	52
4.3.4 撤销	55
4.4 使用观察者模式	55
4.5 使用单例模式	58
4.6 小结	60
4.7 扩展阅读	60
4.7.1 苹果文档	60

4.7.2 其他资源	61
------------------	----

第 5 章 通过 Objective-C 的 ARC 管理内存

5.1 Objective-C 的 ARC 基础入门	62
5.1.1 历史简介	62
5.1.2 手动引用计数与自动引用计数	63
5.2 什么是 ARC	63
5.2.1 在 ARC 工程中集成非 ARC 的第三方代码	64
5.2.2 在非 ARC 工程中集成 ARC 代码	64
5.2.3 框架中的 ARC 代码	64
5.2.4 ARC 空声明变量	67
5.2.5 Objective-C 命名约定	67
5.2.6 覆盖默认行为	67
5.2.7 自由桥接	68
5.2.8 ARC 内部原理	68
5.3 常见的 ARC 移植错误	69
5.4 小结	72
5.5 扩展阅读	72
5.5.1 苹果文档	72
5.5.2 WWDC 讲座	72
5.5.3 博客	73

第 6 章 熟练使用表视图

6.1 UITableView 类的继承关系	74
6.2 熟悉表视图	75
6.2.1 UITableViewController	75
6.2.2 UITableViewCell	75
6.2.3 提高表视图的滚动速度	75
6.2.4 自定义非重复表单元	82
6.2.5 表视图最佳方案: 编写简洁代码, 简化控制器内容	83
6.3 小结	87
6.4 扩展阅读	87
6.4.1 苹果文档	87
6.4.2 其他资源	87

第 7 章 集合视图与自动布局

7.1 集合视图	88
----------------	----

7.1.1 类与协议	88	9.3 图层绘制	132
7.1.2 示例	89	9.3.1 直接设置内容	134
7.2 自动布局	97	9.3.2 实现 display 方法	134
7.2.1 使用自动布局	98	9.3.3 自定义绘图	135
7.2.2 了解自动布局	98	9.3.4 在自己的上下文中绘图	136
7.2.3 相对布局约束	98	9.4 移动对象	136
7.2.4 视觉化格式语言	101	9.4.1 隐式动画	137
7.2.5 调试布局错误	102	9.4.2 显式动画	137
7.3 小结	103	9.4.3 模型与表示	138
7.4 扩展阅读	104	9.4.4 关于定时	140
7.14.1 苹果文档	104	9.5 三维动画	142
7.14.2 WWDC 讲座	104	9.6 美化图层	145
第 8 章 绘图功能	105	9.7 用动作实现自动动画	146
8.1 iOS 的不同绘图系统	105	9.8 为自定义属性添加动画	147
8.2 UIKit 和视图绘图周期	106	9.9 Core Animation 与线程	148
8.3 视图绘制与视图布局	108	9.10 小结	149
8.4 自定义视图绘制	108	9.11 扩展阅读	149
8.4.1 通过 UIKit 绘图	109	9.11.1 苹果文档	149
8.4.2 路径	109	9.11.2 其他资源	149
8.4.3 理解坐标系	111	第 10 章 错误处理	150
8.4.4 重新调整大小以及内容模式	114	10.1 错误处理模式	150
8.4.5 变形	115	10.2 断言	151
8.4.6 通过 Core Graphics 进行绘制	117	10.3 异常	153
8.4.7 混用 UIKit 与 Core Graphics	120	10.4 捕获与报告程序崩溃	153
8.4.8 管理图形上下文	121	10.5 错误和 NSError	154
8.5 优化 UIView 绘制	123	10.6 错误处理块	156
8.5.1 避免绘图	123	10.7 日志	157
8.5.2 缓存与后台绘制	123	10.7.1 记录敏感信息	158
8.5.3 自定义绘图与预渲染	123	10.7.2 获取日志文件	159
8.5.4 像素对齐与模糊文本	124	10.8 小结	159
8.5.5 透明、不透明与隐藏	125	10.9 扩展阅读	159
8.6 CGLayer	125	10.9.1 苹果文档	159
8.7 小结	127	10.9.2 其他资源	159
8.8 扩展阅读	127	第 11 章 位置服务	161
8.8.1 苹果文档	127	11.1 Core Location 框架	161
8.8.2 其他资源	128	11.1.1 获取用户位置	162
第 9 章 Core Animation	129	11.1.2 使用内置的罗盘获取用户 前进方向	162
9.1 视图动画	129	11.1.3 位置服务和用户隐私	163
9.2 管理用户交互	131		

11.2	后台位置服务	165
11.2.1	后台获取位置更新	165
11.2.2	显著位置变更通知	165
11.2.3	区域监控(地理围栏)	167
11.3	位置模拟	167
11.4	时刻关注电量消耗	168
11.5	小结	168
11.6	扩展阅读	168

第三部分 选择工具

第 12 章 表视图常用 UI 范例 170

12.1	下拉刷新	170
12.2	无限滚动	173
12.3	行内编辑与键盘	175
12.4	UITableView 的动画效果	177
12.5	表视图动画的实际应用	178
12.5.1	实现折叠列表	178
12.5.2	实现选项抽屉	180
12.5.3	在表视图单元格中使用手势识别器	181
12.6	小结	181
12.7	扩展阅读	182

第 13 章 多任务 183

13.1	后台运行最佳实践:能力越大责任越大	183
13.2	状态恢复系统	185
13.2.1	测试状态恢复系统	186
13.2.2	选择性加入	186
13.2.3	应用启动过程的变化	187
13.2.4	状态恢复标识符	187
13.2.5	状态编码器与状态解码器	188
13.2.6	表视图和集合视图	191
13.2.7	状态恢复系统的高级应用	192
13.3	多任务和运行循环简介	192
13.4	以操作为中心的多任务开发	193
13.5	用 GCD 实现多任务	197
13.5.1	GCD 和 ARC	198
13.5.2	源和定时器	198
13.5.3	用分派屏障创建同步点	199

13.5.4	队列目标和优先级	200
13.5.5	分派组	201
13.5.6	队列关联数据	202
13.5.7	分派数据	202
13.6	小结	203
13.7	扩展阅读	203
13.7.1	苹果文档	203
13.7.2	WWDC 讲座	203
13.7.3	其他资源	203

第 14 章 REST 解惑 204

14.1	REST 简介	205
14.2	选择数据交换格式	205
14.2.1	在 iOS 中解析 XML	205
14.2.2	在 iOS 中解析 JSON	206
14.2.3	XML 与 JSON	207
14.2.4	模型版本化	208
14.3	假想的 Web 服务	208
14.4	重要提醒	209
14.5	RESTfulEngine 架构(iHotelApp 示例代码)	209
14.5.1	NSURLConnection 与第三方框架	209
14.5.2	创建 RESTfulEngine	210
14.5.3	使用访问令牌对 API 调用进行认证	213
14.5.4	在 RESTfulEngine.m 中覆盖相关方法以添加自定义认证头部	213
14.5.5	取消请求	214
14.5.6	请求响应	214
14.5.7	对 JSON 数据进行键值编码	215
14.5.8	列表页面的 JSON 对象与详细页面的 JSON 对象	217
14.5.9	嵌套 JSON 对象	217
14.5.10	少即是多	219
14.5.11	错误处理	219
14.5.12	本地化	221
14.5.13	使用分类处理其他格式	221

14.5.14 在 iOS 中提升性能的小技巧	222	16.1.3 检查框架、类和方法的可用性	252
14.6 小结	222	16.2 检测设备的功能	253
14.7 扩展阅读	222	16.2.1 检测设备及判断功能	253
14.7.1 苹果文档	222	16.2.2 检测硬件和传感器	254
14.7.2 其他资源	223	16.3 应用内发送 Email 和短信	258
第 15 章 通过安全服务巩固系统安全	224	16.4 检查多任务支持	258
15.1 理解 iOS 沙盒	224	16.5 支持 iPhone 5	259
15.2 保证网络通信的安全	225	16.6 UIRequiredDeviceCapabilities	260
15.2.1 证书工作原理	226	16.7 小结	261
15.2.2 检验证书的有效性	228	16.8 扩展阅读	262
15.2.3 判断证书的可信度	231	16.8.1 苹果文档	262
15.3 使用文件保护	233	16.8.2 其他资源	262
15.4 使用钥匙串	234	第 17 章 国际化和本地化	263
15.5 使用加密	236	17.1 什么是本地化	263
15.5.1 AES 概要	237	17.2 本地化字符串	264
15.5.2 使用 PBKDF2 将密码转换成密钥	237	17.3 对未本地化的字符串进行审查	265
15.5.3 AES 模式和填充	239	17.4 格式化数字和日期	266
15.5.4 初始化向量	240	17.5 nib 文件和 Base Internationalization	269
15.5.5 使用 HMAC 进行认证	241	17.6 本地化复杂字符串	270
15.5.6 错误的密码	241	17.7 小结	272
15.5.7 进行单槽加密	242	17.8 扩展阅读	272
15.5.8 提高 CommonCrypto 的性能	244	17.8.1 苹果文档	272
15.5.9 组合使用加密和压缩	247	17.8.2 WWDC 讲座	272
15.6 小结	247	第 18 章 通过应用内购买提高销售业绩	273
15.7 扩展阅读	248	18.1 开始之前	273
15.7.1 苹果文档	248	18.2 应用内购买什么	273
15.7.2 WWDC 讲座	248	18.2.1 禁止通过应用内购买出售的产品	274
15.7.3 其他资源	248	18.2.2 重新思考你的商业模式	275
第 16 章 在多个苹果平台和苹果设备上运行应用	249	18.3 在 iTunes Connect 上设置产品配置	275
16.1 开发多平台应用	249	18.3.1 第一步: 为你的应用创建一个新的 App ID	276
16.1.1 可配置的目标设置: Base SDK 和 Deployment Target	250	18.3.2 第二步: 生成配置文件	277
16.1.2 支持多个 SDK 时的注意事项: 框架、类和方法	250	18.3.3 第三步: 创建应用的产品项	277
		18.3.4 第四步: 创建应用内购买产品项	278