

iOS 7

应用开发实战详解

朱元波 管蕾 编著



新版本

全面讲解了 iOS 7 开发的各种技术。

热门技术

基本控件、数据存储、多场景处理、界面旋转、图形图像、动画处理、声音服务、多媒体技术、定位处理、互联网应用、触摸手势、硬件操作、地址簿等。

014041232

TP316
703

iOS 7

应用开发实战详解

朱元波 管蕾 编著



北航

C1729370

TP316
703

人民邮电出版社
北京

图书在版编目 (C I P) 数据

iOS 7应用开发实战详解 / 朱元波, 管蕾编著. --
北京: 人民邮电出版社, 2014.5
ISBN 978-7-115-34369-7

I. ①i... II. ①朱... ②管... III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆CIP数据核字(2014)第005151号

内 容 提 要

iOS 系统从诞生到现在经历了短短的几年时间,凭借其硬件产品 iPhone 和 iPad 强大的用户体验,受到了广大用户和开发者的青睐,目前一直在智能手机操作系统中位居前列。

全书分为 4 篇共 20 章,循序渐进地讲解了 iOS 系统开发的基本知识。本书从搭建开发环境讲起,依次讲解了下载并安装 Xcode 开发工具、基本控件、数据存储、多场景处理、界面旋转、图形图像、动画处理、声音服务、多媒体技术、定位处理、互联网应用、触摸手势、硬件操作、邮箱、地址簿等高级知识。本书几乎涵盖了 iOS 开发所需要的全部内容,读者可以从本书中学到很多有用的知识。全书内容言简意赅,并且讲解方法通俗易懂、详细,特别适合于初学者学习。

本书适合作为 iOS 初学者、iOS 爱好者、iPhone 开发人员、iPad 开发人员的学习用书，也可以作为相关培训学校的培训教材和大专院校相关专业的教学用书。

◆ 编 著 朱元波 管 蕾

责任编辑 张 涛

责任印制 彭志环 焦志炜

◆ 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>

三河市潮河印业有限公司印刷

◆ 开本: 787×1092 1/16

印张: 24.5

字数：677 千字

2014年5月第1版

印数：1—3 000 册

2014年5月河北第1次印刷

定价：59.00 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

前言

2013年6月11日，在WWDC大会上，苹果公司发布了iOS 7系统。这套系统采用了一套全新的配色方案，整个界面为很明显的半透明果冻色，对拨号、天气、日历、短信等几乎所有应用的交互界面都进行了重新设计，整体效果更为动感、时尚。为了帮助读者迅速掌握iOS 7的开发知识，笔者在第一时间写作了本书。

iOS 的发展历程

iOS最早于2007年1月9日的苹果Macworld展览会上公布，随后苹果公司于同年6月发布第一版iOS操作系统，当初的名称为“iPhone运行Mac OS X”。当时的苹果公司CEO史蒂夫·乔布斯先生说服了各大软件公司以及开发者先搭建低成本的网络应用程序（Web APP），这样可以使得它们能像iPhone的本地化程序一样来测试“iPhone运行Mac OS X”平台。当前的市场调查显示，搭载iOS系统的iPhone手机仍然是最受欢迎的一款智能手机，搭载iOS系统的iPad电脑仍然是最受欢迎的一款平板电脑。下面回忆过去几年中iOS的辉煌时刻吧！

2007年10月17日，苹果公司发布了第一个本地化iPhone应用程序开发包（SDK）。

2008年3月6日，苹果公司发布了第一个测试版开发包，并且将“iPhone runs OS X”改名为“iPhone OS”。

2008年9月，苹果公司将iPod touch的系统也换成了“iPhone OS”。

2010年2月27日，苹果公司发布iPad，iPad同样搭载了“iPhone OS”。

2010年6月，苹果公司将“iPhone OS”改名为“iOS”，同时还获得了思科iOS的名称授权。

2010年第四季度，苹果公司的iOS占据了全球智能手机操作系统26%的市场份额。

2011年10月4日，苹果公司宣布iOS平台的应用程序已经突破50万款。

2012年2月，iOS平台的应用总量达到552247个，其中游戏应用最多，达到95324个，比重为17.26%；书籍类以60604个位居第二，比重为10.97%；娱乐应用排在第三，总量为56998个，比重为10.32%。

2012年6月，苹果公司在WWDC 2012上推出了全新的iOS 6，提供了超过200项新功能。

本书特色

本书内容十分丰富，实例内容覆盖全面。我们的目标是通过一本图书提供多本图书的价值，读者可以根据自己的需要有选择地阅读。在内容的编写上，本书具有以下一些特色。

（1）结构合理

从用户的实际需要出发，科学安排知识结构。全书详细地讲解了与iOS开发有关的所有知识点，内容循序渐进，由浅入深。

（2）易学易懂

本书条理清晰、语言简洁，可帮助读者快速掌握每个知识点，使读者既可以按照本书编排的章节顺序进行学习，也可以根据自己的需求对某一章节进行针对性的学习。

（3）实用性强

本书彻底摒弃枯燥的理论和简单的操作，注重实用性和可操作性，通过实例的实现过程，详细讲解了各个知识点的基本知识。

（4）内容全面

本书可以号称市面上“内容最全面的一本iOS书”，无论是搭建开发环境，还是控件接口，

或是网络、多媒体和动画，在本书中读者都能找到解决问题的答案。

本书的内容安排

第一部分 必备技术篇

第1章 iOS 开发基础

第2章 搭建开发环境

第3章 Cocoa Touch

第4章 Xcode Interface Builder 界面开发

第5章 编写 MVC 程序

第二部分 核心技术篇

第6章 基本控件

第7章 UIView 详解

第8章 表视图 (UITableView)

第9章 视图控制器

第10章 实现多场景和弹出框

第11章 界面旋转、大小和全屏处理

第三部分 进阶技术篇

第12章 图形、图像、图层和动画

第13章 声音服务

第14章 多媒体应用

第15章 定位处理

第16章 多点触摸和手势识别

第17章 和硬件之间的操作

第18章 地址簿、邮件和 Twitter

第四部分 综合实战篇

第19章 体验 iOS 7 的全新功能

第20章 开发一个通讯录系统

读者对象

· 初学 iOS 编程的自学者

大中专院校的老师和学生

做毕业设计的学生

iOS 编程爱好者

相关培训机构的老师和学员

从事 iOS 开发的程序员

本书在编写过程中，得到了人民邮电出版社工作人员的大力支持，正是由于各位编辑的求实、耐心和高效率，本书才能在这么短的时间内出版。另外也十分感谢我的家人在我写作的时候给予的大力支持。另外，由于编者水平有限，书中纰漏和不尽如人意之处在所难免，诚请读者提出宝贵意见或建议，以便修订并使之更臻完善。

编 者

目 录

第一部分 必备技术篇

第 1 章 iOS 开发基础..... 2

- 1.1 全新的 iOS 7 系统..... 2
 - 1.1.1 iOS 发展史..... 2
 - 1.1.2 iOS 7 的全新功能..... 3
- 1.2 从 iPhone 到 iPad..... 4
 - 1.2.1 让世界疯狂的 iPhone..... 4
 - 1.2.2 改变世界的 iPad..... 4
- 1.3 iOS 的常用开发框架..... 5
 - 1.3.1 Foundation 框架简介..... 5
 - 1.3.2 Cocoa 框架简介..... 6

第 2 章 搭建开发环境..... 8

- 2.1 开发前的准备——加入 iOS 开发团队..... 8
- 2.2 安装 Xcode..... 10
 - 2.2.1 Xcode 介绍..... 10
 - 2.2.2 iOS SDK 介绍..... 11
 - 2.2.3 下载并安装 Xcode..... 12
- 2.3 熟悉 Xcode 集成环境..... 15
 - 2.3.1 创建一个 Xcode 项目并启动模拟器..... 15
 - 2.3.2 Xcode 集成开发环境概述..... 18
- 2.4 常用的第三方工具..... 28
 - 2.4.1 iPhone Simulator..... 28
 - 2.4.2 Interface Builder..... 29

第 3 章 Cocoa Touch..... 30

- 3.1 Cocoa Touch 基础..... 30
 - 3.1.1 Cocoa Touch 概述..... 30
 - 3.1.2 Cocoa Touch 中的框架..... 31
 - 3.1.3 Cocoa Touch 的优势..... 31
- 3.2 iPhone 的技术层..... 32

- 3.2.1 Cocoa Touch 层..... 32
- 3.2.2 多媒体层..... 36
- 3.2.3 核心服务层..... 37
- 3.2.4 核心 OS 层..... 38

- 3.3 Cocoa Touch 中的框架..... 39
 - 3.3.1 Core Animation（图形处理）框架..... 39
 - 3.3.2 Core Audio（音频处理）框架..... 40
 - 3.3.3 Core Data（数据处理）框架..... 40

- 3.4 iOS 程序的生命周期..... 41
 - 3.4.1 从一段代码看 iOS 程序的生命周期..... 41
 - 3.4.2 iOS 程序生命周期的原理..... 42
 - 3.4.3 UIViewController 的生命周期..... 43

- 3.5 Cocoa 中的类..... 45
 - 3.5.1 核心类..... 45
 - 3.5.2 数据类型类..... 46
 - 3.5.3 UI 界面类..... 48

- 3.6 国际化..... 50
- 3.7 使用 Xcode 学习 iOS 框架..... 51
 - 3.7.1 使用 Xcode 文档..... 51
 - 3.7.2 快速帮助..... 52

第 4 章 Xcode Interface Builder 界面开发..... 54

- 4.1 Interface Builder 基础..... 54
 - 4.1.1 Interface Builder 的作用..... 54
 - 4.1.2 Interface Builder 的新特色..... 54
- 4.2 Interface Builder 采用的方法..... 56
- 4.3 Interface Builder 的故事板..... 57

4.3.1	推出的背景	57	6.2	文本视图 (UITextView)	94
4.3.2	故事板的文档大纲	58	6.2.1	文本视图基础	94
4.3.3	文档大纲的区域对象	59	6.2.2	实战演练——在屏幕中 换行显示文本	95
4.4	创建一个界面	59	6.3	标签 (UILabel)	96
4.4.1	对象库	60	6.3.1	标签 (UILabel) 的属性	96
4.4.2	将对象加入到视图中	61	6.3.2	实战演练——使用 标签 (UILabel) 显示 一段文本	96
4.4.3	使用 IB 布局工具	61	6.4	按钮 (UIButton)	99
4.5	定制界面外观	64	6.4.1	按钮基础	99
4.5.1	使用属性检查器	64	6.4.2	实战演练——按下 按钮后触发一个事件	100
4.5.2	设置辅助功能属性	65	6.5	滑块 (UISlider)	101
4.5.3	测试界面	66	6.5.1	滑块 (UISlider) 的 属性	101
4.6	将界面连接到代码	67	6.5.2	实战演练——实现 各种各样的滑块	101
4.6.1	打开项目	67	6.6	步进控件 (UIStepper)	108
4.6.2	输出口和操作	68	6.7	图像视图控件 (UIImageView)	108
4.6.3	创建到输出口的连接	69	6.7.1	UIImageView 的 常用操作	109
4.6.4	创建到操作的连接	71	6.7.2	实战演练—— 在屏幕中显示图像	112
第 5 章	编写 MVC 程序	73	6.8	开关控件 (UISwitch)	113
5.1	MVC 模式基础	73	6.8.1	开关控件基础	113
5.1.1	MVC 的结构	73	6.8.2	联合使用 UISlider 与 UISwitch 控件	113
5.1.2	MVC 的特点	74	6.9	分段控件 (UISegmentedControl)	115
5.2	Xcode 中的 MVC	74	6.9.1	分段控件的属性和 方法	115
5.3	在 Xcode 中实现 MVC	75	6.9.2	实战演练——使用 UISegmentedControl 控件	117
5.3.1	Xcode 中的视图	75	6.10	Web 视图 (UIWebView)	119
5.3.2	Xcode 中的视图控 制器	75	6.10.1	Web 视图基础	119
5.4	数据模型	77	6.10.2	实战演练——在屏幕中 显示指定的网页	120
5.5	使用模板 Single View Application	78	6.11	可滚动的视图 (UIScrollView)	122
5.5.1	创建项目	78	6.11.1	UIScrollView 的 基本用法	122
5.5.2	规划变量和连接	83			
5.5.3	设计界面	85			
5.5.4	创建并连接输出口和 操作	86			
5.5.5	实现应用程序逻辑	89			
5.5.6	生成应用程序	90			
第二部分 核心技术篇					
第 6 章	基本控件	92			
6.1	文本框 (UITextField)	92			
6.1.1	文本框基础	92			
6.1.2	实战演练——设置文本 输入框的边框线样式	92			

6.11.2 实战演练——使用 可滚动视图控件·····	122	第 9 章 视图控制器·····	164
6.12 提醒视图 (UIAlertView) ·····	126	9.1 导航控制器 (UIViewController) 简介·····	164
6.12.1 UIAlertView 基础·····	126	9.1.1 UIViewController 基础·····	164
6.12.2 实战演练——实现一个 自定义提醒对话框·····	128	9.1.2 实战演练——实现 不同界面之间的跳转 处理·····	165
6.13 操作表 (UIActionSheet) ·····	131	9.2 使用 UINavigationController ·····	166
6.14 工具栏 (UIToolbar) ·····	131	9.2.1 导航栏、导航项和 栏按钮项·····	167
6.14.1 工具栏基础·····	131	9.2.2 UINavigationController 详解·····	168
6.14.2 实战演练——实现 一个播放、暂停按钮·····	133	9.2.3 在故事板中使用导航 控制器·····	170
6.15 选择器视图 (UIPickerView) ·····	135	9.2.4 实战演练——使用 导航控制器展现 3 个 场景·····	172
6.15.1 选择器视图基础·····	135	9.3 选项卡栏控制器·····	176
6.15.2 实战演练——实现 两个 UIPickerView 控件间的数据依赖·····	137	9.3.1 选项卡栏和选项卡 栏项·····	177
6.16 日期选择 (UIDatePicker) ·····	140	9.3.2 实战演练——使用 选项卡栏控制器 构建 3 个场景·····	179
第 7 章 UIView 详解·····	144	第 10 章 实现多场景和弹出框·····	185
7.1 UIView 基础·····	144	10.1 多场景故事板·····	185
7.1.1 UIView 的结构·····	144	10.1.1 多场景故事板基础·····	185
7.1.2 视图架构·····	146	10.1.2 创建多场景项目·····	186
7.1.3 视图层次和子视图 管理·····	146	10.1.3 实战演练——实现多个 视图之间的切换·····	190
7.1.4 视图绘制周期·····	147	第 11 章 界面旋转、大小和全屏处理·····	196
7.2 实战演练——设置 UIView 的 位置和尺寸·····	147	11.1 启用界面旋转·····	196
7.3 实战演练——隐藏指定的 UIView 区域·····	148	11.2 设计可旋转和调整大小的 界面·····	197
7.4 实战演练——改变背景颜色·····	150	11.2.1 自动旋转和自动 调整大小·····	197
7.5 实战演练——实现背景透明·····	152	11.2.2 调整框架·····	197
第 8 章 表视图 (UITableView) ·····	154	11.2.3 切换视图·····	197
8.1 表视图基础·····	154	11.2.4 实战演练——使用 Interface Builder 创建可 旋转和调整大小的 界面·····	197
8.1.1 表视图的外观·····	154		
8.1.2 表单元格·····	154		
8.1.3 添加表视图·····	155		
8.1.4 UITableView 详解·····	157		
8.2 实战演练·····	157		
8.2.1 实战演练——列表 显示 18 条数据·····	157		
8.2.2 实战演练——自定义 UITableViewCell·····	158		

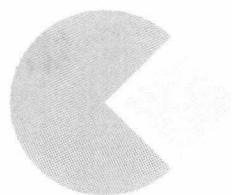
11.2.5 实战演练——在旋转时 调整控件	200	14.1.4 使用多媒体选择器	248
第三部分 进阶技术篇		14.1.5 使用音乐播放器	249
第 12 章 图形、图像、图层和动画	208	14.1.6 实战演练——使用 Media Player 播放 视频	249
12.1 图形处理	208	14.2 AV Foundation 框架	251
12.1.1 iOS 的绘图机制	208	14.2.1 准备工作	252
12.1.2 实战演练——在屏幕中 绘制一个三角形	209	14.2.2 使用 AV 音频播放器	252
12.2 图像处理	209	14.2.3 使用 AV 录音机	253
12.2.1 实战演练——实现颜色 选择器/调色板功能	212	14.3 图像选择器 (UIImagePickerController)	253
12.2.2 实战演练——实现滑动 颜色选择器/调色板 功能	213	14.3.1 使用图像选择器	254
12.3 图层	216	14.3.2 图像选择器控制器 委托	254
12.3.1 视图和图层	217	14.3.3 用 UIImagePickerController 调用系统照相机	255
12.3.2 实战演练——在屏幕中 实现 3 个重叠的矩形	217	14.4 一个多媒体的应用程序	256
12.4 实现动画	218	14.4.1 实现概述	256
12.4.1 UIImageView 动画	218	14.4.2 创建项目	256
12.4.2 视图动画 UIView	219	14.4.3 设计界面	257
12.4.3 Core Animation 详解	223	14.4.4 创建并连接输出和 操作	257
12.4.4 实战演练——实现 “烟花烟花满天飞” 效果	226	14.4.5 实现电影播放器	259
第 13 章 声音服务	229	14.4.6 实现音频录制和播放	261
13.1 访问声音服务	229	14.4.7 使用照片库和相机	264
13.1.1 声音服务基础	229	14.4.8 实现 Core Image 滤镜	266
13.1.2 实战演练——播放 声音文件	230	14.4.9 访问并播放音乐库	267
13.2 提醒和震动	235	第 15 章 定位处理	271
13.2.1 播放提醒音	235	15.1 Core Location 框架	271
13.2.2 实战演练——实现 iOS 的提醒功能	235	15.1.1 Core Location 基础	271
第 14 章 多媒体应用	246	15.1.2 使用流程	271
14.1 Media Player 框架	246	15.2 获取位置	274
14.1.1 Media Player 框架中的类	246	15.2.1 位置管理器委托	274
14.1.2 使用电影播放器	247	15.2.2 处理定位错误	275
14.1.3 处理播放结束	248	15.2.3 位置精度和更新 过滤器	275
		15.2.4 获取航向	275
		15.3 地图功能	276
		15.3.1 Map Kit 基础	276
		15.3.2 为地图添加标注	277
		15.4 实战演练——创建一个 支持定位的应用程序	278
		15.4.1 创建项目	278

15.4.2	设计视图	280
15.4.3	创建并连接输出口	281
15.4.4	实现应用程序逻辑	281
15.4.5	生成应用程序	283
15.5	实战演练——在屏幕中实现 一个定位系统	284
15.5.1	设计界面	284
15.5.2	具体编码	285
第 16 章	多点触摸和手势识别	289
16.1	多点触摸和手势识别基础	289
16.2	触摸处理	290
16.2.1	触摸事件和视图	290
16.2.2	实战演练——触摸 屏幕中的按钮	294
16.2.3	实战演练——同时滑动 屏幕中的两个滑块	295
16.3	手势处理	296
16.3.1	手势处理基础	296
16.3.2	实战演练——实现一个 手势识别器	300
第 17 章	和硬件之间的操作	309
17.1	加速计和陀螺仪	309
17.1.1	加速计基础	309
17.1.2	陀螺仪	314
17.1.3	实战演练——检测 倾斜和旋转	314
17.2	访问朝向和运动数据	319
17.2.1	两种方法	319
17.2.2	实战演练——检测 朝向演练	321
第 18 章	地址簿、邮件和 Twitter	324
18.1	地址簿	324
18.1.1	框架 Address Book UI	324
18.1.2	框架 Address Book	326
18.2	电子邮件	326
18.3	使用 Twitter 发送推特信息	327
18.4	实战演练——联合使用地址簿、 电子邮件、Twitter 和地图	328

18.4.1	创建项目	329
18.4.2	设计界面	329
18.4.3	创建并连接输出口 和操作	330
18.4.4	实现地址簿逻辑	331
18.4.5	实现地图逻辑	333
18.4.6	实现电子邮件逻辑	335
18.4.7	实现 Twitter 逻辑	336
18.4.8	生成应用程序	337

第四部分 综合实战篇

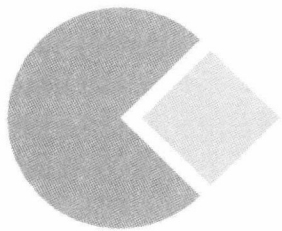
第 19 章	体验 iOS 7 的全新功能	340
19.1	UI 方面的变化	340
19.1.1	新的 UI 变化改进	340
19.1.2	实战演练——体验 扁平化设计风格	340
19.1.3	实战演练——体验 iOS 7 的动画效果	347
19.1.4	实战演练——体验 iOS 7 的模糊效果	349
19.2	使用 SpriteKit	351
19.2.1	Sprite Kit 介绍	352
19.2.2	使用 Sprite Kit 框架	352
19.2.3	实战演练——使用 Sprite Kit 框架开发 一个小游戏	354
19.3	全新的 Game Center	359
19.3.1	GameCenter 设置	359
19.3.2	实战演练——使用 GameCenter 开发 一个简单的多人 游戏	361
第 20 章	开发一个通讯录系统	369
20.1	设计 UI 视图	369
20.2	实现根视图	370
20.3	添加联系人	373
20.4	查看联系人视图	377
20.5	实现编辑视图	379
20.6	视图配置	381



第一部分

必备技术篇

- 第 1 章 iOS 开发基础
- 第 2 章 搭建开发环境
- 第 3 章 Cocoa Touch
- 第 4 章 Xcode Interface Builder 界面开发
- 第 5 章 编写 MVC 程序



第1章 iOS 开发基础

iOS 是一款强大的智能手机操作系统，被广泛地应用于 iPhone、iPad 和 iTouch 等苹果公司的系列产品设备中。iOS 通过这些移动设备，向用户展示了一个多点触摸、可始终在线、视频以及内置众多传感器的界面。本章将带领读者一起来认识 iOS 这款神奇的系统，为读者进入本书后面知识的学习打下基础。

1.1 全新的 iOS 7 系统

iOS 系统是由苹果公司开发的手持设备操作系统。苹果公司最早于 2007 年 1 月 9 日的 Macworld 大会上公布这个系统，最初是设计给 iPhone 使用的，后来陆续套用到 iPod touch、iPad 以及 Apple TV 等苹果公司的产品上。iOS 与苹果的 Mac OS X 操作系统一样，最初的系统名为“iPhone OS”，在 2010 年 6 月 7 日 WWDC 大会上宣布改名为“iOS”。截止至 2012 年 7 月，根据 Canalys 的数据统计资料，iOS 已经占据了全球智能手机系统市场份额的 37%，在美国市场的占有率为 45%。

1.1.1 iOS 发展史

iOS 最早于 2007 年 1 月 9 日的苹果 Macworld 展览会上公布，随后苹果公司于同年的 6 月发布第一版 iOS 操作系统，当初的名称为“iPhone 运行 OS X”。当时的苹果公司 CEO 史蒂夫·乔布斯先生说服了各大软件公司以及开发者先搭建低成本的网络应用程序（Web APP），这样可以使得它们能像 iPhone 的本地化程序一样来测试“iPhone 运行 OS X”平台。

2007 年 10 月 17 日，苹果公司发布了第一个本地化 iPhone 应用程序开发包（SDK）。

2008 年 3 月 6 日，苹果发布了第一个测试版开发包，并且将“iPhone runs OS X”改名为“iPhone OS”。

2008 年 9 月，苹果公司将 iPod touch 的系统也换成了“iPhone OS”。

2010 年 2 月 27 日，苹果公司发布 iPad，iPad 同样搭载了“iPhone OS”。

2010 年 6 月，苹果公司将“iPhone OS”改名为“iOS”，同时还获得了思科 iOS 的名称授权。

2010 年第四季度，苹果公司的 iOS 占据了全球智能手机操作系统 26% 的市场份额。

2011 年 10 月 4 日，苹果公司宣布 iOS 平台的应用程序已经突破 50 万款。

2012 年 2 月，iOS 平台的应用总量达到 552247 个，其中游戏应用最多，达到 95324 个，比重为 17.26%；书籍类以 60,604 个排在第二，比重为 10.97%；娱乐应用排在第三，总量为 56,998 个，比重为 10.32%。

2012 年 6 月，苹果公司在 WWDC 2012 上推出了全新的 iOS 6，提供了超过 200 项新功能。

2013 年 6 月 11 日，在 WWDC 大会上苹果公司发布了 iOS 7 系统。这个系统采用了一套全新的配色方案，整个界面有很明显的半透明果冻色，拨号、天气、日历、短信等几乎所有应用的交

互界面都进行了重新设计，整体看来更为动感、时尚。全新的 iOS 7 系统可应用在 iPhone 4 及以上机型中。

1.1.2 iOS 7 的全新功能

iOS 7 发布后让人有眼前一亮的感觉，由于新系统加入了大量的 3D 效果，加之部分功能全部采用了悬浮式半透明结构设计，这让 iOS 7 看起来既有科技感又清新。

除了全新的扁平化界面外，苹果还重新设计了 iOS 7 的控制中心，并且新系统支持真正的多任务（卡片式），同时还加入了不少手势操作功能。下面咱们就来看看 iOS 7 的新功能吧。

① iOS 7 增加了 AirDrop 功能，iOS 用户可以在多台设备之间分享文件，操作也非常简单，即选中相关文件，发送给网内指定的人即可。不过，该功能只支持 iPhone 5、iPad 4、iPad mini 以及 iPod touch 5。

② iOS 7 中的 Siri 除了换上了新界面外，还支持车载导航设备（可以在汽车显示屏当中查看信息、拨打电话），并且加入男声和一个全新的接口（整合更多的第三方功能与服务），此外 Siri 还整合了维基百科和 Twitter 的内容。

③ iOS 7 中的 Safari 支持全屏显示、智能搜索功能以及酷炫的窗口切换 3D 效果，同时还改进了收藏夹和标签体验，增加了家长控制和 iCloud 钥匙串功能。

④ iOS 7 还具有全新拍照功能，首先是拍照应用中加入了 Square 特性和各式各样的滤镜效果，同时相册中照片可以按照时间进行自动分类，而用户还可以把照片分享到别人的“相片流”里。

⑤ iOS 7 的原生应用中加入了全新的手势操作，通过手势返回到主界面（多任务处理过程中，用户可以左右滑动来选择切换应用），同时 App Store 具有自动更新的特性，系统可以在后台自动更新软件。

⑥ iOS 7 的控制中心中加入了“手电筒”功能，而天气应用也经过了大幅改动，缩放查看天气综述，采用了动态天气背景。

⑦ iOS 7 整合了苹果新的 iTunes Radio 流媒体音乐服务。

除此之外，iOS 7 还有大量的改进，只是苹果公司没有详细说明，包括邮件搜索的改进、App Store 的购买改进、Safari 的防数据追踪、与腾讯微博的合作、Wi-Fi 的升级、Map 的黑夜模式、智能邮箱系统、PDF 阅读、企业版的登入、单个 APP VPN、长 MMS 短信等。

全新的 iOS 7 界面效果如图 1-1 所示。



▲图 1-1 全新的 iOS 7 界面的效果

1.2 从 iPhone 到 iPad

强大的 iOS 系统被广泛地应用于苹果公司的移动系列产品中。广大开发人员无需纠结于开发的程序是否能在不同的硬件设备中运行，因为只要是 iOS 程序就可以在支持 iOS 系统的设备中运行。不同设备之间的差异只是表现在屏幕大小而已。在当前的 iOS 开发项目中主要存在开发两类程序：iPhone 程序和 iPad 程序，这两者的屏幕大小不一样。本节将简要讲解运行 iOS 系统最火的两个产品——iPhone 和 iPad。

1.2.1 让世界疯狂的 iPhone

iPhone 是一个集合了照相、个人数码助理、媒体播放器以及无线通信设备的掌上智能手机。iPhone 最早由史蒂夫·乔布斯在 2007 年 1 月 9 日举行的 Macworld 上宣布推出，并于 2007 年 6 月 29 日在美国上市。2007 年 6 月 29 日，iPhone 2G 在美国上市；2008 年 7 月 11 日，苹果公司推出 3G iPhone。2010 年 6 月 8 日凌晨 1 点，乔布斯发布了 iPhone 4。2011 年 10 月 5 日凌晨，iPhone 4S 发布。2012 年 9 月 13 日凌晨（美国时间 9 月 12 日上午），iPhone 5 发布。全新的 iPhone 5 如图 1-2 所示。



▲图 1-2 全新的 iPhone 5

与上一代产品 iPhone 4S 相比，iPhone 5 更轻薄，屏幕尺寸更大，它的厚度大概是 7.6mm，比前一代薄了 18%；重量在 112 克左右，比 4S 轻了 20%；采用速度更快的 A6 处理器；整体外观也拉长。iPhone 5 的屏幕尺寸扩大到 4 英寸，屏幕的比例是 16:9，应用程序的图标比前一代增加了一行。iPhone 5 的运算速度两倍于 iPhone 4S，当时因为 iPhone 4S 采用的是 A5 处理器，而新的处理器的尺寸缩小了 22%。iPhone 5 支持 4G 技术的 LTE 网络。

1.2.2 改变世界的 iPad

iPad 是苹果公司于 2010 年发布的一款平板电脑的名称，定位介于苹果的智能手机 iPhone 和笔记本电脑产品之间，通体只有 4 个按键，与 iPhone 布局一样，提供了浏览互联网、收发电子邮件、观看电子书、播放音频和播放视频等功能。

2010 年 1 月 27 日，在美国旧金山欧巴布也那艺术中心（芳草地艺术中心）所举行的苹果公司发布会上，平板电脑 iPad 正式发布。

2012 年 3 月 8 日，苹果公司在美国芳草地艺术中心发布第三代 iPad。受到市场普遍期待的苹

果新一代平板电脑全新 iPad 的外形与 iPad 2 相似，但电池容量增大，有 3 块 4000mAh 锂电池；芯片速度更快，使用 A5X 双核处理器，图形处理器功能增强、配四核 GPU，并且在美国的售价与 iPad 2 一样。第三代 iPad 如图 1-3 所示。



▲图 1-3 第三代 iPad

1.3 iOS 的常用开发框架

为了提高开发 iOS 程序的效率，除了可以使用 Xcode 集成开发工具之外，还可以使用第三方提供的框架。这些框架是由许多类、方法、函数、文档按照一定的逻辑组织起来的集合，为我们提供了完整的项目解决方案，使研发程序变得更容易。OS X 下的 Mac 操作系统中大约存在 80 个框架，这些框架可以用来开发应用程序，处理 Mac 的 Address Book 结构、刻制 CD、播放 DVD，使用 QuickTime 播放电影、播放歌曲等。

在 iOS 的众多框架中有两个最为常用的框架，即 Foundation 框架和 Cocoa 框架。本节将简要讲解这两个框架的基本知识。

1.3.1 Foundation 框架简介

在 Mac 的操作系统中，为所有程序开发奠定基础的框架是 Foundation 框架。该框架允许使用一些基本对象，例如数字和字符串，以及一些对象集合，如数组、字典和集合。其他功能包括处理日期和时间、自动化的内存管理、处理基础文件系统、存储(或归档)对象、处理几何数据结构(如点和长方形)。

Foundation 头文件的存储目录是：

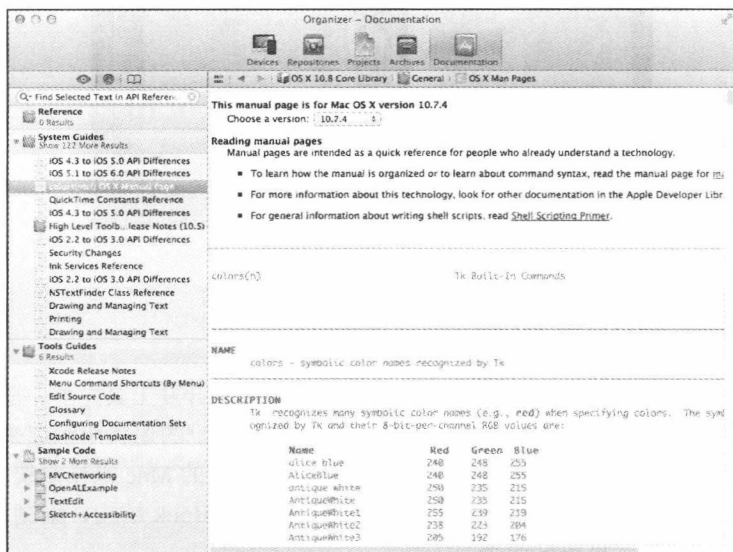
```
/System/Library/Frameworks/Foundation.framework/Headers
```

上述头文件实际上与其存储位置的其他目录相链接。请读者查看这个目录中存储在系统上的 Foundation 框架文档，熟悉它的内容和用法简介。Foundation 框架文档存储在计算机系统中（具体位于/Develop/Documentation 目录中），另外 Apple 网站上也提供了此说明文档。大多数文档为 HTML 格式的文件，读者可以通过浏览器浏览学习。同时 Apple 网站上也提供了 Acrobat pdf 文件，我们可以学习到包含 Foundation 的所有类及其实现的所有方法和函数的描述。

如果正在使用 Xcode 开发程序，可以通过 Xcode 的 Help 菜单中的 Documentation 窗口访问学习文档。通过这个窗口，可以搜索和访问存储在计算机本机中或者在线的文档。如果正在 Xcode 中编辑文件并且想要快速访问某个特定头文件、方法或类的文档，可以通过高亮显示编辑器窗口中的文本并鼠标右键单击的方法来实现。在弹出的快捷菜单中，可以适当选择 Find Selected Text in

Documentation 或者 Find Selected Text in API Reference。Xcode 将搜索文档库，并显示与查询相匹配的结果。

接下来的内容将带领读者一起简单了解 Foundation 框架是如何工作的。NSString 类是 Foundation 框架中的一个类，可以使用它来处理字符串。假设正在编辑某个使用该类的程序，并且想要获得更多关于这个类及其方法的信息，无论何时，当单词 NSString 出现在编辑窗口时，都可以将其高亮显示并右键单击。如果从出现的菜单中选择“Find Selected Text in API Reference”命令，会得到一个外观与图 1-4 所示类似的文档窗口。



▲图 1-4 NSString 类的文档

如果向下滚动标有 NSString Class Reference 的面板，将发现（在其他内容中间）一个该类支持的所有方法的列表。这是一个能够获得有关实现哪些方法等信息的便捷途径，包括它们如何工作以及它们的预期参数。

读者们可以在线访问 developer.apple.com/referencelibrary，打开 Foundation 参考文档（通过 Cocoa、Frameworks、Foundation Framework Reference 链接），在这个站点中还能够发现一些介绍某些特定编程问题的文档，例如内存管理、字符串和文件管理。除非订阅的是某个特定文档集，否则在线文档要比存储在计算机硬盘中的文档从时间上有更新。

在 Foundation 框架中包括了大量可供使用的类、方法和函数。在 Mac OS X 上大约有 125 个可用的头文件。作为一种简便的形式，我们可以使用如下代码头文件。

```
#import <Foundation/Foundation.h>
```

因为 Foundation.h 文件实际上导入了其他所有 Foundation 头文件，所以不必担心是否导入了正确的头文件，Xcode 会自动将这个头文件插入到程序中。虽然使用上述代码会显著地增加程序的编译时间，但是通过使用预编译的头文件，可以避免这些额外的时间开销。预编译的头文件是经过编译器预先处理过的文件。在默认情况下，所有 Xcode 项目都会受益于预编译的头文件。在本章使用每个对象时都会用到这些特定的头文件，这会有助于我们熟悉每个头文件所包含的内容。

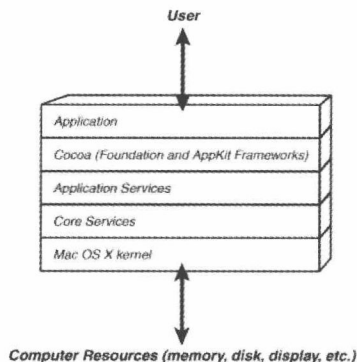
1.3.2 Cocoa 框架简介

在 iOS 应用中，通过使用 Application Kit 框架包提供了与窗口、按钮、列表等相关的类。

Application Kit 框架包含广泛的类和方法，它们能够开发交互式图形应用程序，使得开发文本、菜单、工具栏、表、文档、剪贴板和窗口等应用变得十分简便。在 Mac OS X 操作系统中，术语 Cocoa 是指 Foundation 框架和 Application Kit 框架。术语 Cocoa Touch 是指 Foundation 框架和 UIKit 框架。由此可见，Cocoa 是一种支持应用程序提供丰富用户体验的框架，它实际上由如下两个框架组成。

- Foundation 框架。
- Application Kit（或 AppKit）框架。

其中后者用于提供与窗口、按钮、列表等相关的类。在编程语言中，通常使用示意图来说明框架最顶层应用程序与底层硬件之间的层次。例如图 1-5 就是一个这样的图。



▲图 1-5 应用程序层次结构

图 1-5 中各个层次的具体说明如下所示。

- User: 用户。
- Application: 应用程序。
- Cocoa (Foundation and AppKit Frameworks): Cocoa (Foundation 和 AppKit 框架)。
- Application Services: 应用程序服务。
- Core Services: 核心服务。
- Mac OS X kernel: Mac OS X 内核。
- Computer Resources(memory, disk,display, etc.): 计算机资源（内存、磁盘、显示器等）。

内核及设备驱动程序能够提供与硬件的底层通信，它负责管理系统资源，包括调度要执行的程序、管理内存和电源，以及执行基本的 I/O 操作。

核心服务提供的支持比它上面的层次更加底层或更加“核心”，例如，在 Mac OS X 中主要实现了对集合、网络、调试、文件管理、文件夹、内存管理、线程、时间和电源的管理。

应用程序服务层包含对打印和图形呈现的支持，包括 Quartz、OpenGL 和 QuickTime。由此可见，Cocoa 层直接位于应用程序层之下。正如图中指出的那样，Cocoa 包括 Foundation 和 AppKit 框架。Foundation 框架提供的类用于处理集合、字符串、内存管理、文件系统、存档等。通过 AppKit 框架中提供的类，可以管理视图、窗口、文档等用户界面。在很多情况下，Foundation 框架为底层核心服务层（主要用过程化的 C 语言编写）中定义的数据结构定义了一种面向对象的映射。

Cocoa 框架用于 Mac OS X 桌面与笔记本电脑的应用程序开发，而 Cocoa Touch 框架用于 iPhone 与 iTouch 的应用程序开发。Cocoa 和 Cocoa Touch 都有 Foundation 框架。然而在 Cocoa Touch 下，UIKit 代替了 AppKit 框架，以便为很多相同类型的对象提供支持，比如窗口、视图、按钮、文本域等。另外，Cocoa Touch 还提供使用加速器（它与 GPS 和 Wi-Fi 信号一样都能跟踪位置）的类和触摸式界面，并且去掉了不需要的类，比如支持打印的类。