

柯博文老师系统论述iOS APP开发方法、技巧及项目实战（13个APP完整案例）的著作！
简体中文版、繁体中文版、英文版三种版本全球同步发售！
赠送配书源代码及全程教学视频！

清华

开发者书库



The Practical Development Guide for iOS

Step

iOS实战指南

手把手教你掌握APP项目开发

柯博文◎著

Poweren Ko

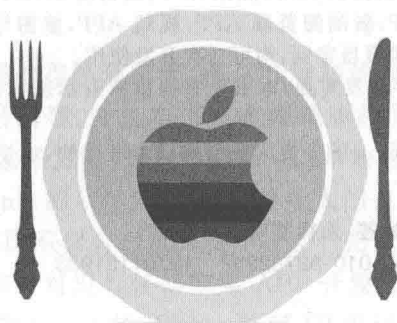
教学视频
工程文件

清华大学出版社



清华

开发者书库



The Practical Developing Guide for iOS

Programming iOS APP Step by Step

iOS实战指南

手把手教你掌握APP项目开发

柯博文◎著

Rowen Ko

清华大学出版社

北京

内 容 简 介

本书系统介绍了 iOS APP 开发的一般原理、方法与项目实践。全书主要采用项目实例的方式来介绍 APP 开发的全过程。全书内容包括 iOS 概述,开发环境和安装 XCode,第一个 Objective-C 程序,Objective-C 语法,iOS 基本组件,触发事件、换页与 Storyboard,信息视窗,Tab Bar Controllers,列表组件,引导画面和 APP 图标的设置与动画效果,多线程,屏幕尺寸调整和旋转,网络,文件、SQLite 数据库和 Core Data,JSON、XML 和 SOAP,手机功能,多媒体,广告,手机地图 GPS,SpriteKit,蓝牙 4.0 BLE iBeacon,测试和版本控制,上架、实际测试和函数库。书中详细介绍了 13 个 APP 项目案例的开发过程,用以帮助读者巩固理论知识,包括度量衡换算 APP,切换画面 APP,点菜 APP,购物清单 APP,按钮选单 APP,iPad/iPhone 版旅行 APP,电子书 APP,记事本 APP,新闻阅读器 APP,视频 APP,旅游导航地图 APP,Flappy Bird 游戏 APP,手机控制周边设备 APP。这些项目案例,均可二次开发使用。

为提高读者学习效果,便于动手开发实践,本书配套提供所有示例源代码,并精心录制了所有章节的教学视频。

本书适合作为广大移动开发人员、iOS 开发人员的入门参考读物,也适合作为高等学校计算机相关专业移动开发类课程的参考书。

本书封面贴有清华大学出版社防伪标签,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

iOS 实战指南:手把手教你掌握 APP 项目开发/柯博文著. —北京:清华大学出版社,2016

(清华开发者书库)

ISBN 978-7-302-43790-1

I. ①i… II. ①柯… 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(2016)第 100192 号

责任编辑:盛东亮

封面设计:李召霞

责任校对:梁毅

责任印制:宋林

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

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

课件下载: <http://www.tup.com.cn>, 010-62795954

印装者:三河市春园印刷有限公司

经 销:全国新华书店

开 本:186mm×240mm 印 张:35.25

字 数:814 千字

版 次:2016 年 7 月第 1 版

印 次:2016 年 7 月第 1 次印刷

印 数:1~2500

定 价:89.00 元

产品编号:068509-01

目录

CONTENTS

第 1 章 iOS 概述	1
1.1 iOS 的起源	1
1.2 使用界面	2
1.3 iOS 系统结构	2
1.4 iOS 版本功能介绍	4
1.4.1 iOS 7.0 介绍	4
1.4.2 iOS 8.0 介绍	6
1.4.3 iOS 9.0 介绍	7
第 2 章 开发环境和安装 XCode	10
2.1 安装 XCode 方法 1: 网站下载	10
2.2 安装 XCode 方法 2: APP Store	15
2.3 引导开发环境 XCode	18
本章习题	19
第 3 章 第一个 Objective-C 程序	20
3.1 创建第一个 iOS APP	20
3.2 修改程序	23
3.3 调试	25
本章习题	26
第 4 章 Objective-C 语法	27
4.1 字符串 NSString、NSLog 的使用方法	27
4.2 数据类型	30
4.2.1 int	30
4.2.2 long	31
4.2.3 float	31

4.2.4	double	32
4.2.5	bool / Boolean	32
4.2.6	综合练习	33
4.3	数学计算	34
4.4	判断式	36
4.4.1	if 条件判断	36
4.4.2	switch 条件判断	37
4.4.3	综合练习	38
4.4.4	enum 枚举	39
4.5	创建循环	41
4.5.1	for 语句	41
4.5.2	goto 语句	42
4.5.3	while 语句	43
4.5.4	do-while 语句	44
4.6	数组	45
4.6.1	Array	45
4.6.2	NSArray	47
4.6.3	NSMutableArray	48
4.6.4	NSDictionary	50
4.6.5	NSMutableDictionary	51
4.7	Class 类型	52
4.7.1	创建自己的 Class 类型	52
4.7.2	类型函数和参数	55
4.7.3	id 类型	57
4.7.4	Class 类型的继承	58
4.7.5	super 调用父类函数	60
4.7.6	self 本身	62
4.8	权限声明	63
4.8.1	public、protected 和 private 的使用	64
4.8.2	@property 和@synthesize	66
4.9	static	68
4.9.1	static 变量	68
4.9.2	static 静态函数	69
4.9.3	extern	71

第 5 章 iOS 基本组件

(项目: 度量衡换算 APP)	73
5.1 Storyboard 工具设计画面	73
5.1.1 添加组件	74
5.1.2 调整组件	74
5.1.3 新增组件变量	75
5.2 UILabel 文字组件——更改文字颜色和尺寸	77
5.3 Storyboard 设定组件指派事件	80
5.3.1 新增指派事件	81
5.3.2 组件的指派事件和触发事件	83
5.3.3 取消指派事件	83
5.3.4 取消组件变量	83
5.4 UIButton——按钮反应	84
5.5 UITextField——文字输入	86
5.6 UITextField——设置键盘样式	89
5.7 综合练习——公尺英里计算器 APP	90
5.8 UIImageView——图片切换	92
5.9 UIButton——美食蛋糕 APP	95
5.10 UITextView——多样化的文字显示	97
5.11 UISegmentedControl——设置 APP 后台颜色	100
5.12 UISwitch——显示口令	103
5.13 UISlider——调整图片的圆角	105
5.14 UIProgressView——显示完成进度	108
5.15 UIStepper——购买数量	110
5.16 UIDataPicker——动态输入时间	112
5.17 UIScrollView——超过画面的处理	115
5.18 UIView——组件动态产生和重叠	118

第 6 章 触发事件、换页与 Storyboard

(项目: 切换画面 APP)	122
6.1 触发事件 1——Event	123
6.2 触发事件 2——delegate	127
6.3 触发事件 3——selector	130
6.4 用 Storyboard 来换页	132
6.5 新增类型来控制新页面	137
6.6 UINavigationController	141

6.7	NavigationController 的 push 和 modal 的差异以及标题设置	145
6.8	NavigationController——通过程序切换页和传递数据	148
6.9	NavigationController 的显示和隐藏	151
6.10	UIBarButtonItem	153
第7章 信息视窗		
	(项目：点菜 APP)	156
7.1	UIAlertView——单一按钮信息确认	157
7.2	UIAlertView——询问信息	159
7.3	UIAlertController——没有按钮	161
7.4	UIAlertController——OK 按钮	164
7.5	UIAlertController——Yes 和 No 按钮	166
7.6	UIAlertController——使用名称和密码	168
7.7	UIActivityIndicatorView——显示读取中	171
第8章 Tab Bar Controllers		
174		
8.1	Tab Bar Controllers 控制画面切换方法一	174
8.2	Tab Bar Controllers 控制画面切换方法二	178
第9章 列表组件		
	(项目：购物清单 APP)	184
9.1	UITableView——使用列表功能	185
9.2	UITableView——APP 直接引导列表	191
9.3	UITableView——内容设置	193
9.4	UIViewController——详细内容	195
9.5	UITableView——自定义列表样式	201
9.6	UITableView——多段内容和表头设计	205
9.7	UISearchDisplayController——搜索内容	208
9.8	UITableView——删除数据	211
9.9	UITableView——添加数据	213
9.10	UITableView——移动数据	215
9.11	UITableView——Grouped 设置页	216
第10章 引导画面和 APP 图标的设置与动画效果		
	(项目：按钮选单 APP)	221
221		
10.1	修改 APP 的 icon	221
10.2	设定启动画面——Launch Storyboard	225

10.3	设定启动画面——xcassets	226
10.4	动画移动——animateWithDuration 设置动画时间	231
10.5	旋转动画——CGAffineTransformMakeRotation 设置旋转角度	234
10.6	缩放动画——缩小窗口	236
10.7	缩放并旋转动画	238
10.8	混合动画效果	240
第 11 章 多线程		243
11.1	dispatch_async——后台下载网络图片 APP	244
11.2	NSTimer	247
11.3	delegate	251
第 12 章 屏幕尺寸调整和旋转 (项目: iPad/iPhone 版旅行 APP)		254
12.1	iPhone/iPad 的不同和调整	254
12.2	设计 iPhone 和 iPad 应用程序和旋转	255
12.3	Autosizing	258
12.4	Auto Layout	263
12.5	判断 iPhone 和 iPad 设备	269
第 13 章 网络 (项目: 电子书 APP)		271
13.1	UIWebView——显示网页和部落格 APP	272
13.2	UIWebView——显示 PDF 文件和电子书 APP	274
13.3	HTTPGet 的样例程序——会员报到	277
13.4	HTTPPost 的样例程序——会员登录 APP	281
13.5	NSURLConnection——异步下载 HTTP Get	283
13.6	NSURLConnection——异步下载 HTTP Post	286
13.7	Hybrid 网页型 APP	288
13.7.1	Hybrid APP 概述	288
13.7.2	Web APP、Hybrid APP 和 Native APP 纵向对比	289
13.7.3	Hybrid APP 的第三方工具	290
13.8	在 iOS 使用 PhoneGap	293
13.9	网络连接状态	299

第 14 章 文件、SQLite 数据库和 Core Data

(项目：记事本 APP)	304
14.1 文件存储——文件总管 APP	305
14.1.1 iOS 的 APP 文件结构	305
14.1.2 创建目录	308
14.1.3 目录下创建文件	309
14.1.4 取得目录中的所有文件名称	310
14.1.5 改变路径	310
14.1.6 移除文件	311
14.1.7 Doc 目录下文件的存储和读入	312
14.1.8 存储多种数据类型	313
14.1.9 读取多种数据类型	313
14.1.10 读取专案中的文件	314
14.1.11 读取专案中的文件	314
14.2 UserDefaults 简易数据访问	315
14.3 KeyChain 机密数据	317
14.4 SQLite 数据库	323
14.4.1 SQLite 数据准备	323
14.4.2 SQLite 初始化	325
14.4.3 读取数据库,显示数据	327
14.4.4 新增数据	330
14.4.5 删除数据	332
14.5 CoreData	334
14.5.1 准备	334
14.5.2 设置 Core Data 数据库 Data Model	337
14.5.3 读取和显示 Core Data 数据库 Data model	341
14.5.4 Core Data 增加数据	344
14.5.5 删除数据	346
14.5.6 数据的选择 NSPredicate	348

第 15 章 JSON、XML 和 SOAP

(项目：新闻阅读器 APP)	352
15.1 JSON	352
15.2 XML 和 RSS——新闻 APP	357
15.3 SOAP 数据处理	362

第 16 章 手机功能	367
16.1 调整屏幕尺寸	368
16.2 iOS 设备唯一认证码 UUID	373
16.3 发简讯方法 1	375
16.4 发简讯方法 2	376
16.5 网页	380
16.6 邮件	381
16.7 拨号	383
16.8 简易版地图	384
16.9 iTunes 音乐和打开其他 APP——软件市场 APP	386
16.10 Youtube 影片——电视影集 APP	388
16.11 拍照	389
16.12 获取相簿	391
16.13 震动	393
16.14 摇一摇	394
16.15 手机水平	397
第 17 章 多媒体	
(项目：视频 APP)	400
17.1 AVAudioRecorder——声音录制	400
17.2 AVAudioPlayer——音乐盒	405
17.3 MediaPlayer 播放视频	408
17.4 MediaPlayer 录影	411
第 18 章 广告	414
18.1 广告 iAd 后台设置	414
18.2 ADBannerView——广告 iAd	417
18.3 ADInterstitialAdDelegate——全画面广告	421
第 19 章 手机地图 GPS	
(项目：旅游导航地图 APP)	424
19.1 GPS 位置和高度	425
19.2 罗盘和指南针	429
19.3 显示地图	433
19.4 指定地图位置、尺寸和 GPS 显示用户的位置	437
19.5 地图上加标记点	439

19.6	自制地图标记点	441
19.7	导航	444
第 20 章 SpriteKit		
(项目：Flappy Bird 游戏 APP)		448
20.1	SpriteKit 游戏	448
20.2	画面设计	452
20.3	显示主角——动画切换图片	457
20.4	后台移动	458
20.5	背景持续地移动	460
20.6	远近效果	461
20.7	水管移动	463
20.8	游戏性——显示多个水管	465
20.9	物理	467
20.10	物理对象碰撞	469
20.11	物理碰撞处理	472
20.12	跳线——物理施加外力	473
20.13	计分	475
第 21 章 蓝牙 4.0 BLE iBeacon		
(项目：手机控制周边设备 APP)		478
21.1	iOS 显示所有的 iBeacon 设备	479
21.2	显示 iBeacon 的远近	486
21.3	iOS 传送和接收数据给 Arduino 设备	491
第 22 章 测试和版本控制		500
22.1	Analyze 单元测试工具	501
22.2	Profile 测写工具介绍	501
22.3	Activity Monitor 执行分析	503
22.4	Allocations 内存泄露	505
22.5	Automation 自动测试程序	507
22.6	版本控制	510
22.6.1	将专案设置和使用 Git 版本控制	510
22.6.2	把程序记录到版本控制	512
22.6.3	check-in 程序登记到版本控制	513
22.6.4	查阅版本的差异	514
22.6.5	退回之前的版本	515

第 23 章 上架、实际测试和函数库	517
23.1 Apple 后台设置——产生凭证密钥	517
23.2 下载安装开发证、测试认证和登记测试机器	524
23.3 在 iPhone 和 iPad 机器上做测试	533
23.4 设置 APP 售价和语句	535
23.5 上传 APP	542
23.6 送审和上架販售	547

第 24 章 iOS 的测试

本书第 23 章介绍了如何上架 APP，本章将介绍如何测试 APP。在上架之前，必须先进行测试，以确保 APP 在各种设备上都能正常运行。本章将介绍如何使用 Xcode 进行单元测试和 UI 测试，以及如何使用 TestFlight 进行 beta 测试。

本章将介绍如何使用 Xcode 进行单元测试和 UI 测试。单元测试是验证代码逻辑正确性的方法，而 UI 测试则是验证用户界面交互是否符合预期。本章还将介绍如何使用 TestFlight 进行 beta 测试，以便在正式发布前收集用户反馈。

本章将介绍如何使用 Xcode 进行单元测试和 UI 测试。单元测试是验证代码逻辑正确性的方法，而 UI 测试则是验证用户界面交互是否符合预期。本章还将介绍如何使用 TestFlight 进行 beta 测试，以便在正式发布前收集用户反馈。

第 1 章



iOS 概述

本章介绍 iOS 常见的组件功能和相关函数、属性。

1.1 iOS 的起源

苹果于 2007 年 6 月 29 日发布了 iOS 的第一个版本,并且发布了第一代 iPhone——闪亮的玻璃外壳,人性化的作业界面,创新的 Multi-Touch 技术——它带起了手机应用软件的风潮,创造了许多惊人的纪录。自从 iPhone 于 2007 年出现后,苹果应用商店已有超过 100 亿次下载量的惊人数据,提供了超过 100 万种软件的下载,汇集了各种创新的应用设计,并造就了无数个百万富翁传奇。

当前苹果公司已发布 iOS 9.0,并在 2015 年 9 月 16 日提供公开下载。iOS 9.0 是 iPhone、iPad 和 iPod touch 的基础。其设计不但外观出色,操作性能出众,甚至连最简单的作业都魅力无比。因为精心打造的 iOS 9.0,就是为了将内建于 Apple 硬件的先进软件技术发挥到淋漓尽致。自始至终,Apple 的设备性能总是远远超前。Apple 同时打造 iPad、iPhone 与 iPod touch 的硬件和软件系统,是为了更好地协同运作,所以 APP 可充分发挥硬件功能,例如双核心处理器、加速的绘图处理和无线天线,等等,路复用处理就是最好的写照。iOS 9.0 会自适应地学习用户何时想使用 APP,并在节能时段更新相应 APP 的内容,例如当连接 WiFi 时。因此用户所喜爱 APP 的内容会保持在最新状态,同时不会大量耗电。

iOS 可以通过 iTunes 对设备进行升级,也可以通过 OTA 的方式进行软件更新。iOS 设备必须通过苹果服务器进行引导,引导方式包括 iTunes 和 iCloud,并且自动同步。最新版本的 iOS 使用 1GB 左右的储存空间,彻底更改用户接口界面,并且加入了近 200 项新功能,具有超过 1500 个 API,界面和使用方法也与以往大不相同,并透过 iPhone 6s 的压力深度的面板,分成一般点选和用力点选,让使用方法更加多变。iOS 9.0 也带来了精彩的新功能,如控制中心、iOS 版 AirDrop、付款机制、健康 API 和更聪明的路复用处理等,从而让用户日常所做的一切更加轻松、快速且充满享受,如图 1-1 所示。许多 APP 虽看似不同以往,但当用户在进行作业时却会感到无比契合的熟悉。所以从一开始,用户便知道如何使用这全世界最先进的行动作业系统。



图 1-1 iOS 的样式

1.2 使用界面

1. 设计特点

iOS 用户接口能使用按键和多点触控对设备进行控制,此外通过其内建的加速器,可以启动旋转装置以使屏幕改变方向,这样的设计使设备更便于使用。

实体按键包括屏幕下方的 Home 按键(用于结束应用程序,回到主接口,长按打开语音控制或 Siri 和 iPhone 5s 集成 Touch ID 感应器)、顶部的 Power 按键(用于锁定屏幕和关机)和侧面的音量控制按键。

2. 多点触控

多点触控包括滑动(swiping)、轻按(tapping)、挤压(pinching)、反向挤压(reverse pinching or unpinching),可以与系统互动。

3. 屏幕界面

屏幕界面以应用程序方格的形式呈现,最底部的一栏成为 Dock,可以有最多四个(iPad 系列产品则为 6 个)程序,其他的应用程序可以通过滑动的方式变换显示。状态栏处于屏幕上方,能显示时间、电池电量和信号强度等信息。从屏幕顶端向下轻扫可以显示通知中心,将图像向下轻扫可以引导 Spotlight,从屏幕底端向上轻扫可以显示控制中心,如图 1-2 所示。

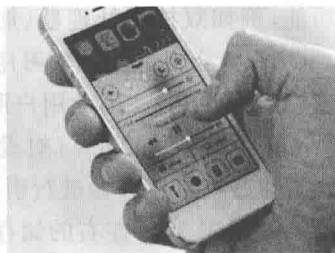


图 1-2 使用接口

1.3 iOS 系统结构

1. 系统结构

底层硬件和创建的应用程序之间的作业系统,可以使应用程序开发者在不理解底层硬

件的结构下,通过 iOS 所提供的 API 很容易地与硬件进行沟通。

iOS 的结构是分层结构,如图 1-3 所示。下层包含基本的核心服务 Core Service 和核心作业系统 Core OS,上层的 Cocoa Touch 是处理用的动作,Media 媒体层用于处理画面显示和多媒体。

iOS 使用苹果麦金塔计算机的 ARM 架构的 64 位或 32 位中央处理器,并且使用由 PowerVR 视频卡渲染的 OpenGL ES 3.0,以 Darwin 作为基础。虽然 iOS 跟 Mac OS 很类似,但是彼此的 iOS 应用程序还是不兼容,该问题利用新的程序语言 Swift 重新编译程序,就可以解决。

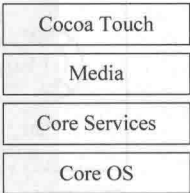


图 1-3 iOS 结构层

2. 系统架构层次

iOS 系统架构层次如图 1-4 所示,包括

- (1) 应用程序层 Applications;
- (2) 核心作业系统层 the Core OS layer;
- (3) 核心服务层 the Core Services layer;
- (4) 触控应用层 the Cocoa Touch layer。

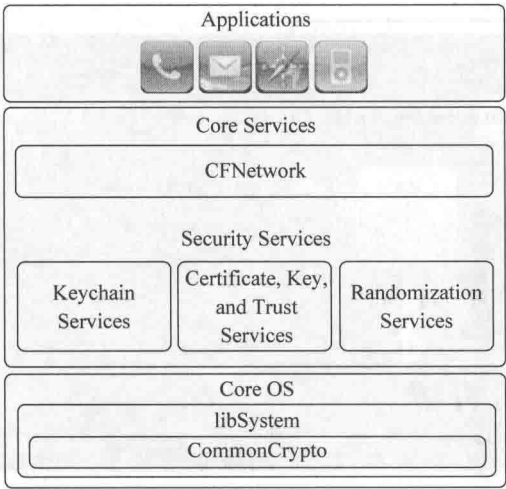


图 1-4 iOS 系统架构层次

下面举例说明程序开发者在编写程序时,APP 是如何处理反应的。首先如图 1-5 所示,整个程序会有一个叫做 Controller 的控制者,用于监听用户所有的输入动作,所以当用户触摸手机的时候,作业系统便会产生反应,然后 Event Loop 返回询问的同时,就会查到该反应,然后这一个动作会被应用程序中的 UIApplication 察觉到并产生一个触发事件给 Application Delegate,然后该事件再交给不同的下属如专门管组件的 View Controller 或窗口的 UIWindow 处理。

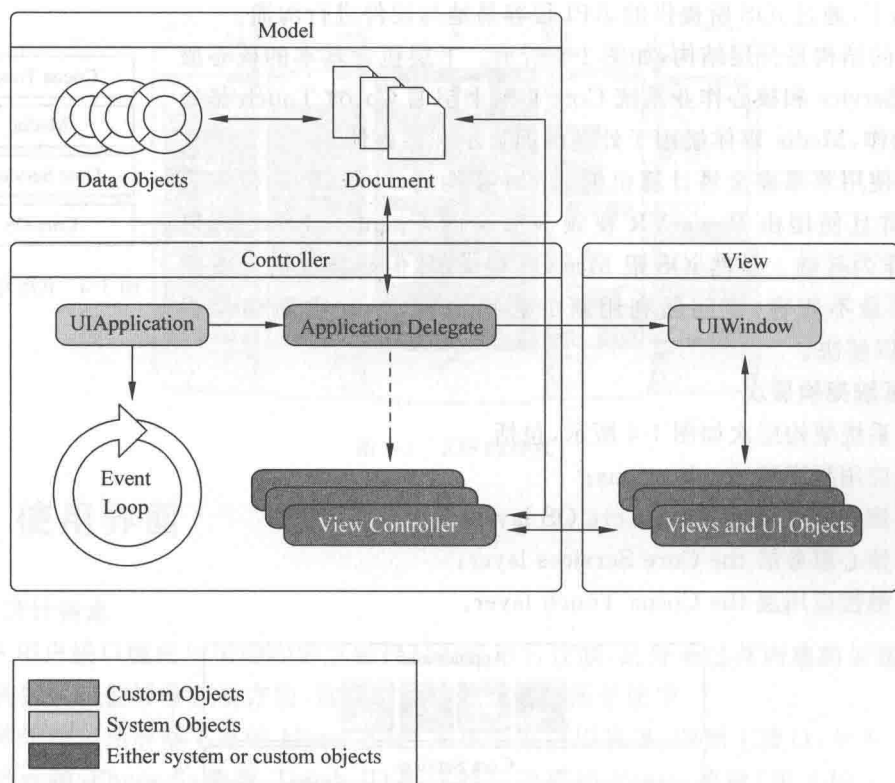


图 1-5 使用接口

1.4 iOS 版本功能介绍

1.4.1 iOS 7.0 介绍

iOS 7.0 于 2013 年推出,该版本较之前的版本修改程度非常大,算是历年来修改最大的版本之一,尤其是画面的风格和设计,由原本的黑色为主的设计风格换成白色的精简风格,如图 1-6 所示。

感兴趣的读者可以到苹果官方网站上查阅更详细的说明,下面介绍几个重要的新功能。

(1) iBeacons 如图 1-7 所示,使用蓝牙 4.0 低功耗 Bluetooth low energy 的设计方法,让 iOS 的设备使用 Bluetooth 4.0 和其他的 Bluetooth 4.0 设备进行数据的传递。需要强调的是这一组 iBeacons 对应的新的功能 API 并不只用于蓝牙配对,也可以用于传递数据消息。

(2) AirDrop 如图 1-8 所示,iOS 上的应用程序可以使用 AirDrop 功能与其他的 APP 可以分享照片文件。

(3) Apple Maps——苹果一直致力于地图功能,所以在该版本中,苹果把地图功能推升

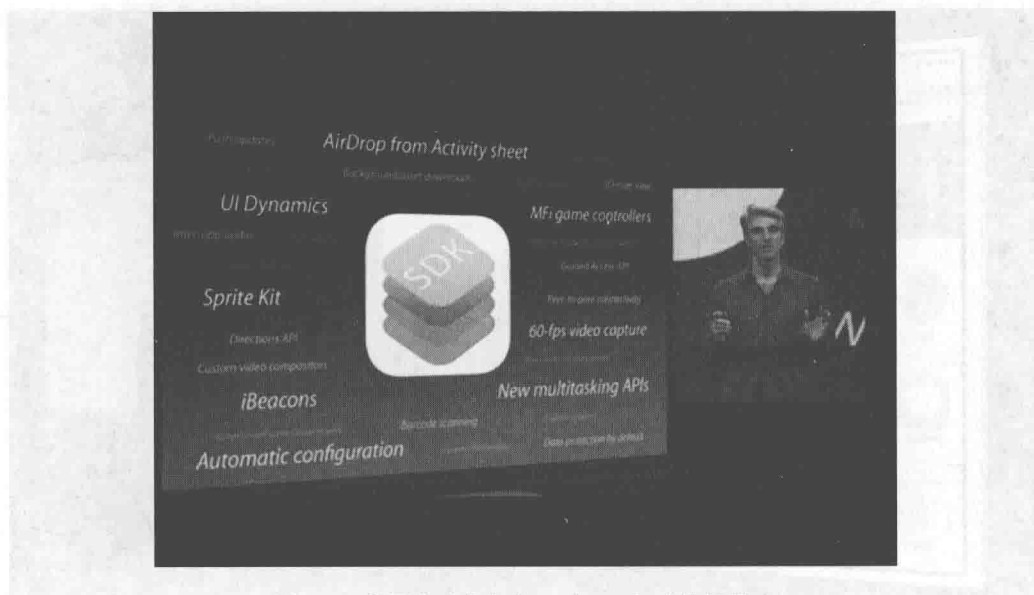


图 1-6 苹果公司发布会上对 iOS 新功能的描述



图 1-7 iBeacons

到 3D 的效果。

(4) MFi Game Controllers 如图 1-9 所示。在该版本中,硬件公司可依照苹果公司的规范,使用 MFi 数据传递技术设计浏览器的摇杆,将 iPhone 和 iPad 直接整合到用户的遥控器之中。

(5) Sprite Kit——The Sprite Kit framework 提供了图形显示和动画的 API,用户可以使用任何图像。Sprite Kit 可以使游戏开发者在处理画面时,每一秒的张数均与游戏场景的内容切换的流程保持一致。