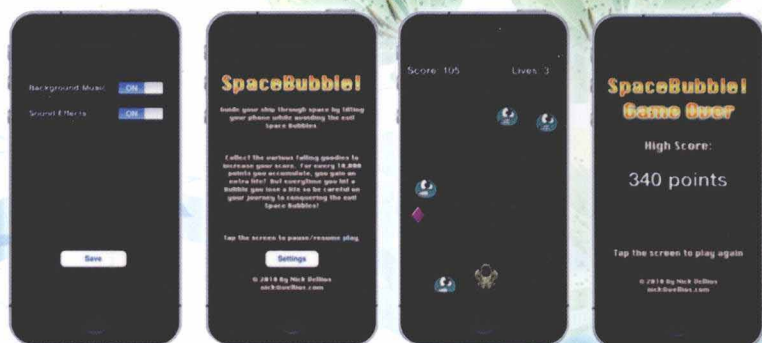


移动平台
开发书库



iPhone 与 iPad 开发实战

—— iOS 经典应用剖析

关东升 赵志荣 等编著

iPhone and iPad Develop Actual Combat

- 真实案例——课程中采用的案例全部是在App Store发布的开源应用程序
- 全程指导——从分析、设计到实战，全面掌握开发流程及技巧
- 通俗易学——剖析细致，讲解深入，实现iOS开发人员进阶
- 516团队www.51work6.com亲情打造



机械工业出版社
CHINA MACHINE PRESS

移动平台开发书库

iPhone 与 iPad 开发实战 ——iOS 经典应用剖析

关东升 赵志荣 贾云龙 编著

机械工业出版社

本书是一本关于 iOS 项目分析与实战的教程,旨在帮助读者能够实际动手开发一些 iOS 项目,同时能够学习到一些理论知识。

本书收录了 6 个 iOS 项目,这些项目是在 iOS 开源网站下载的。涵盖了工具类应用、体育类应用、效率类应用、娱乐类应用和游戏类应用等几个方面。从第 3 章开始,每章都是一个独立的 iOS 项目,除第 6 章的项目外,其他都是在 App Store 上发布的产品。

在介绍项目的章节中,首先是应用介绍,其次是应用剖析,然后是应用实战与讲解,最后是总结。

本书适合作为有一定基础的 iOS 程序员的自学辅导用书,也可作为相关培训机构的培训教材。

图书在版编目(CIP)数据

iPhone 与 iPad 开发实战: iOS 经典应用剖析 / 关东升等编著. —北京: 机械工业出版社, 2012.4

(移动平台开发书库)

ISBN 978-7-111-37674-3

I. ①i… II. ①关… 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 (2012) 第 040498 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 郝建伟 王 凯

责任印制: 杨 曦

北京中兴印刷有限公司印刷

2012 年 5 月第 1 版第 1 次印刷

184mm × 260mm · 23.25 印张 · 573 千字

0 001—3 500 册

标准书号: ISBN 978-7-111-37674-3

定价: 55.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心: (010)88361066

门户网: <http://www.cmpbook.com>

销售一部: (010)68326294

教材网: <http://www.cmpedu.com>

销售二部: (010)88379649

读者购书热线: (010)88379203 封面无防伪标均为盗版

前言

2008 年苹果公司开放 iPhone SDK API，允许第三方人员为其开发 iPhone 应用。作为回报，开发者可以获得一定比例的分成。这样吸引了众多开发人员开始从事 iPhone 和 iPad 应用开发。作者有幸在 2009 年开始接触 iPhone，并深深为之吸引，从 2010 年初开始从事 iOS 教学与培训工作，并主持开发了一些 iOS 项目。

本书的编写目的就是能够使 iOS 初级开发人员进阶成为高级开发人员，对扩展知识层面有一定帮助。

本书收录的 6 个项目，是从 iOS 开源网站下载的。在这里我们要感谢他们的无私，才有了这本书的诞生。本书总共 8 章，第 1 章做开篇综述，第 2 章介绍了 Objective-C 语言，第 3~8 章，每一章介绍一个项目，按照介绍→剖析→实战讲解→总结的顺序介绍。这样安排章节结构是希望能够既介绍相关知识点，又能够让广大读者一步一步做出整个项目。

本书项目都是使用 Xcode3.2.5 开发的，iOS SDK 平台主要采用 4.1 版本，而原作者的很多项目都是较低版本的 Xcode 开发的，它们的转换在书中进行了详细讲解。6 个开源项目的源代码和修改过的程序代码可供读者在 www.51work6.com/ios_book.html 下载。

本书主要由关东升和赵志荣编著，参加本书编写、调试工作的还有贾云龙。本书的顺利出版，还要感谢 516INC (www.51work6.com) 团队给予的大力支持和帮助。

由于作者水平有限，书中难免存在疏漏和不足之处，请读者原谅，并提出宝贵意见。

关东升

目 录

前言

第 1 章 开篇综述	1
1.1 iOS 概述	1
1.2 App Store	2
1.3 iOS 应用程序分类	2
1.3.1 效率型应用程序	2
1.3.2 实用型应用程序	3
1.3.3 沉浸式应用程序	4
1.4 iOS SDK 和开发工具	6
1.4.1 iOS SDK 介绍	6
1.4.2 开发工具 Xcode 介绍	6
1.4.3 Xcode 安装和卸载	7
第 2 章 Objective-C 语言基础	11
2.1 Objective-C 概述	11
2.1.1 Objective-C 特点	11
2.1.2 Objective-C 代码的文件扩展名	12
2.2 编写第一个 Objective-C 程序	12
2.2.1 编写过程	12
2.2.2 解释说明	14
2.2.3 使用 Xcode 运行并调试	15
2.3 数据类型、运算符和表达式	17
2.3.1 数据类型	18
2.3.2 数据类型转换	21
2.3.3 常量与变量	22
2.3.4 运算符和表达式	23
2.4 控制语句	24
2.4.1 分支语句	24
2.4.2 循环语句	26
2.4.3 跳转语句	28
2.5 Objective-C 语言的面向对象特性	29
2.5.1 Objective-C 中的类	30
2.5.2 方法和消息	31

2.5.3	访问成员变量	31
2.5.4	属性	33
2.5.5	构造方法	34
2.5.6	实例成员变量作用域限定符	36
2.5.7	类变量和类方法	38
2.6	继承性	40
2.6.1	继承概念	40
2.6.2	方法重写	41
2.7	多态性	44
2.7.1	多态概念	44
2.7.2	动态类型和动态绑定	46
2.8	分类与协议	51
2.8.1	分类	51
2.8.2	协议	52
2.9	内存管理	54
2.9.1	内存管理概述	54
2.9.2	重写 dealloc 方法	54
2.9.3	自动释放池	57
2.9.4	属性中的内存管理参数	59
2.10	Foundation 框架	60
2.10.1	数字类	61
2.10.2	字符串类	61
2.10.3	数组类	66
2.10.4	字典类	68
	本章小结	70
第 3 章	工具类应用开发实战——密码生成应用	71
3.1	应用介绍	71
3.2	应用剖析	74
3.2.1	核心类图	74
3.2.2	RootViewController 类	75
3.2.3	MainViewController 类	75
3.2.4	FlipsideViewController 类	76
3.2.5	MainView 类	76
3.2.6	FlipsideView 类	77
3.2.7	Nib 文件	77
3.2.8	Cocoa 和 Cocoa Touch 中 MVC 设计模式	78
3.3	应用实战与讲解	79
3.3.1	构建应用骨架	79
3.3.2	视图翻转动画	83

3.3.3	主视图 UI	96
3.3.4	主视图代码	99
3.3.5	主视图控制器代码	104
3.3.6	背后视图 UI	107
3.3.7	背后视图和视图控制器代码	108
本章小结		109
第 4 章 体育类应用开发实战——2012 奥林匹克应用 110		
4.1	应用介绍	110
4.2	应用剖析	114
4.2.1	核心类图	114
4.2.2	主页模块视图控制器类	116
4.2.3	比赛项目模块控制器类	117
4.2.4	倒计时器模块控制器类	117
4.2.5	比赛日程表模块控制器类	118
4.2.6	奥运会歌模块视图控制器类	120
4.2.7	比赛项目模块视图类	120
4.2.8	Nib 文件	122
4.3	应用实战与讲解	122
4.3.1	构建应用骨架	123
4.3.2	同一个工程中实现 iPhone 版和 iPad 版	126
4.3.3	主页模块实现	137
4.3.4	比赛项目模块 UI 实现	141
4.3.5	比赛项目模块代码实现	147
4.3.6	倒计时器模块实现	154
4.3.7	比赛日程表模块 UI 实现	159
4.3.8	比赛日程表模块代码实现	161
本章小结		171
第 5 章 效率类应用开发实战——记事本应用 172		
5.1	应用介绍	172
5.2	应用剖析	175
5.2.1	核心类图	176
5.2.2	RootViewController 类	177
5.2.3	工具类	178
5.2.4	DetailNoteViewControler 类	178
5.2.5	WebViewController 类	179
5.2.6	JSON 类	179
5.2.7	Nib 文件	180
5.3	应用实战与讲解	180

5.3.1	构建应用骨架	180
5.3.2	列表视图实现	182
5.3.3	帮助功能实现	198
5.3.4	同步功能实现	205
5.3.5	添加功能实现	213
5.3.6	修改功能实现	222
5.3.7	发送功能实现	227
5.3.8	删除功能实现	239
5.3.9	保存功能实现	239
本章小结		240
第 6 章	娱乐类应用开发实战 —— 我的位置应用	241
6.1	应用介绍	241
6.2	应用剖析	242
6.2.1	核心类图	242
6.2.2	RootViewController 类	243
6.2.3	MainViewController 类	243
6.2.4	FlipsideViewController 类	244
6.2.5	MainView 类	244
6.2.6	FlipsideView 类	244
6.2.7	Nib 文件	244
6.3	应用实战与讲解	245
6.3.1	构建应用骨架	245
6.3.2	主视图 UI	250
6.3.3	主视图控制器代码	255
6.3.4	背后视图模块实现	260
6.4	增加地图功能	264
6.4.1	基于 Web 的 Google 地图	264
6.4.2	基于本地 Google 地图	266
本章小结		278
第 7 章	工具类应用开发实战 —— 铅锤应用	279
7.1	应用介绍	279
7.2	应用剖析	281
7.2.1	核心类图	281
7.2.2	RootViewController 类	282
7.2.3	MainViewController 类	282
7.2.4	FlipsideViewController 类	283
7.2.5	MainView 类	283
7.2.6	FlipsideView 类	283
7.2.7	PlumbBobAppDelegate 类	284

7.2.8 Nib 文件	284
7.3 应用实战与讲解	284
7.3.1 构建应用骨架	284
7.3.2 主视图业务模块	289
7.3.3 背后视图业务模块	295
7.3.4 根控制器实现	298
本章小结	304
第 8 章 游戏类应用开发实战——太空泡泡应用	305
8.1 应用介绍	305
8.2 应用剖析	306
8.2.1 核心类图	306
8.2.2 SplashViewController 类	308
8.2.3 SpaceBubbleViewController 类	308
8.2.4 SettingsWindow 类	309
8.2.5 GameOverViewController 类	310
8.2.6 Sprite 类	310
8.2.7 SpaceBubbleAppDelegate 类	311
8.2.8 SoundEffects 类	311
8.2.9 SpriteHelpers 类	311
8.2.10 Nib 文件	312
8.3 应用实战与讲解	312
8.3.1 构建应用骨架	312
8.3.2 设置视图模块实现	313
8.3.3 启动视图模块实现	320
8.3.4 游戏结束视图模块实现	324
8.3.5 编写精灵和精灵辅助类	327
8.3.6 编写音效类	332
8.3.7 游戏视图模块实现	336
8.3.8 修改应用程序委托对象	355
本章小结	359
参考文献	361

第1章 开篇综述

2007 年一个叫史蒂夫·乔布斯的人，拿出一个小盒子，并打开了它，从此世界改变了。它的魔力吸引着无数的人，有人排了几天的长队，就是想第一个购买到它，它的销售传奇至今无人能及。它为软件开发者提供了梦想的舞台，并成就了诸多英雄。这个充满魔力的盒子就是 iPhone 它所使用的操作系统是 iOS，它的兄弟姐妹 iPod Touch 和 iPad 也同样采用 iOS 操作系统。

1.1 iOS 概述

iOS 是由苹果公司为 iPhone、iPod Touch 以及 iPad 使用而开发的操作系统。与苹果公司 Mac OS X 操作系统一样，都是基于 Unix 操作系统的。原本这个系统名为 iPhone OS，直到 2010 年 6 月 7 日苹果电脑全球研发者大会（Apple Worldwide Developers Conference，简称 WWDC）大会上宣布改名为 iOS。系统操作占用大概 240MB 的存储器空间。iOS 的系统架构分为 4 个层次：核心操作系统层（the Core OS layer），核心服务层（the Core Services layer），媒体层（the Media layer），可轻触层（the Cocoa Touch layer）。与 Android 系统不同，iOS 是封闭的，无法在苹果以外的设备上使用 iOS。

iOS 通过下面的几个方面诠释了什么是现代的智能手机和平板电脑。

1. 多点触摸和手势

触摸屏幕在 iOS 之前就有了，但是基本上都是单点触摸的，即只能用一个手指，而 iOS 的设备能够感应到多个手指触摸。为了配合这种多点触摸，iOS 上的触摸会有很多种手势，它们有：触击、双击、滑动、长时间触击、轻拂、刷屏和手指合拢、张开等。

2. 统一的屏幕尺寸

iOS 目前屏幕尺寸只有两种，一个是 iPhone 和 iPod Touch 的 3.5 英寸（1 英寸=0.0254 米），另一个是 iPad 的 9.7 英寸。统一的屏幕尺寸给应用软件开发带来很多好处，开发人员可以不用关心屏幕尺寸适配的问题，把精力集中在其他方面。

3. 高分辨率

iOS 拥有超高分辨率屏幕，采用了 Retina 技术，原来只能显示 480×320 像素分辨率的 3.5 英寸屏幕，现在可以显示 960×640 像素分辨率，只有 9.7 英寸的 iPad 屏幕可以显示 1024×768 像素分辨率，这使其像素密度增加了一倍，使得 iOS 画质非常清晰。

4. 重力加速计

iOS 内置重力加速计，用户可以玩很多有意思的游戏，例如：极品飞车，可以使用 iOS



作为方向盘，通过重力加速计感应方向的变化。还有很多与重力加速计相关的应用软件，如水平尺应用，也是借助重力加速计进行感应。

5. 指南针

iOS 内置指南针设备，很多应用是基于指南针的，例如：导航软件及地图应用软件等。

6. 蓝牙和 WiFi 连接

iOS 内置蓝牙和 WiFi 通信模块，这样 iOS 设备之间既可以采用 WiFi 互相连接，也可以采用蓝牙进行连接。还可以通过 WiFi 上网，这样可以节约用户的上网费用。不仅是 iOS 设备之间的连接，还可以与计算机进行连接，这样用户就可以体验很多基于 iOS 的立体感游戏。

1.2 App Store

无论手机有多么的先进，如果没有可玩的游戏，可以使用的软件，手机就会回归它的原始用途——打电话。iOS 之所以取得了如此大的成功，除了其本身优秀的品质外，还有就是它拥有高质量的应用软件商店——App Store。

到目前为止，App Store 超过 40 万款软件，栏目分为：图书、商务、教育、娱乐、财经、游戏、保健、生活、医药、音乐、导航、新闻、摄影、工作、参考书、社交网络、体育、旅游、工具和天气等。应用软件的数量还在不断的增长。

1.3 iOS 应用程序分类

根据视觉和行为特征、数据模型以及用户体验，苹果公司为 iOS 定义了 3 种应用程序风格。这只是为了帮助开发者确认设计方案，并不表示所有的 iOS 应用程序都必须严格遵循这些分类。这 3 类应用程序是：

- 效率型应用
- 实用型应用
- 沉浸式应用

1.3.1 效率型应用程序

效率型应用程序具有组织和操作具体信息的功能，通常用于完成比较重要的任务。邮件是效率型应用程序的典型例子。

效率型应用程序需要简单的用户体验，才不会阻碍用户的工作。因此，成功的效率型应用程序将用户体验的重点放在任务上，用户可以快速找到需要的信息，轻松完成必要的操作，完成该任务继而转向其他事务。

效率型应用程序通常使用列表视图组织数据，因为用户可以逐步将选择具体化，直到达到期望的程度，来找到所需的信息。在本书收录的 6 个项目中，2012 奥林匹克（2012 Olympics）如图 1-1 和图 1-2 所示，记事本（PlainNote）如图 1-3 和图 1-4 所示，两个应用属于效率型应用程序。

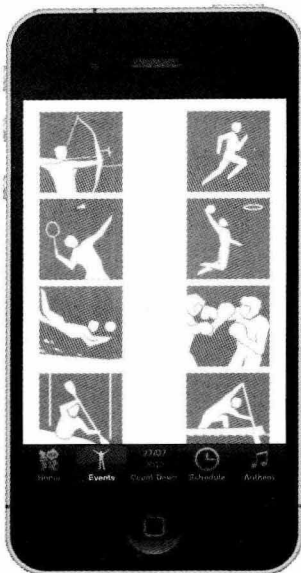


图 1-1 2012 奥林匹克比赛项目

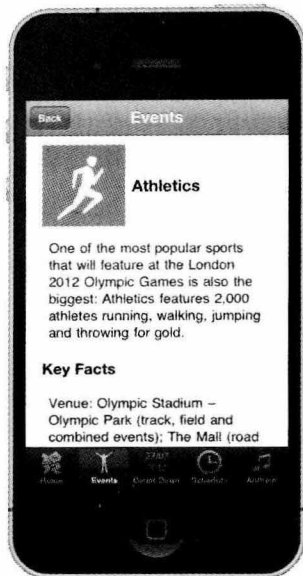


图 1-2 2012 奥林匹克比赛项目详细信息



图 1-3 记事本列表



图 1-4 修改记事本

1.3.2 实用型应用程序

实用型应用程序对用户输入要求很低。用户打开实用型应用程序，是为了快速查看信息摘要或执行简单任务。实用型应用程序一般会有一个主画面和背后画面，在主画面浏览信息



摘要，在背后画面选择改变信息的设置或来源。主画面右下角有信息按钮，可以进入背后画面，在调整完成之后，用户单击完成按钮即可返回主画面。背后画面的选项是应用程序功能的一部分，而不是用户设置一次以后就极少再用的一组偏好设置。因此，实用型应用程序不应该把特定设置放在设置应用程序中。

在本书收录的 6 个项目中，密码生成（Amuck Password Generator）如图 1-5 和图 1-6 所示，我的位置（WhereAmI）如图 1-7 和图 1-8 所示，两个应用属于实用型应用程序。

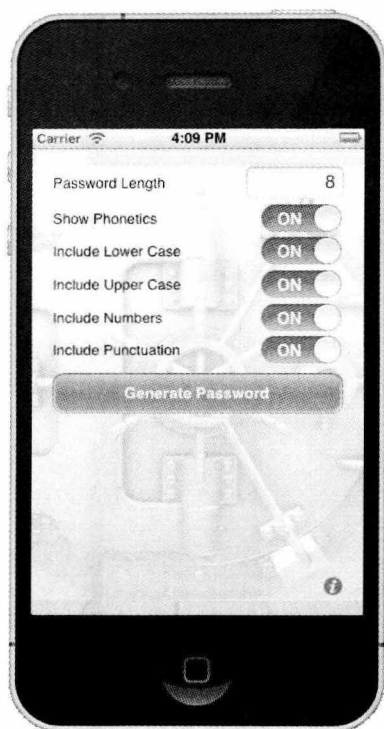


图 1-5 密码生成主画面

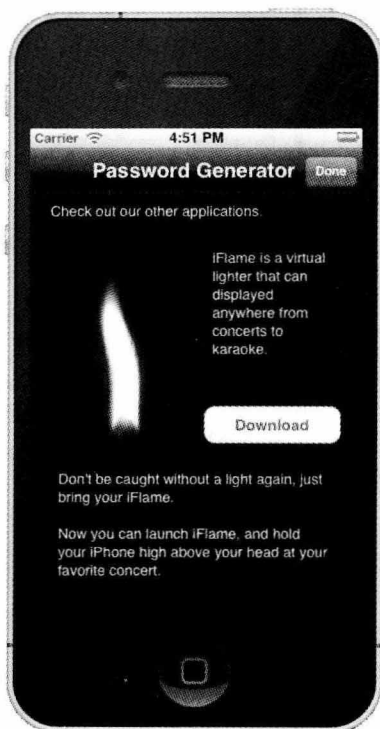


图 1-6 密码生成背后画面

1.3.3 沉浸式应用程序

沉浸式应用程序提供具有丰富视觉效果的全屏环境，专注于内容以及用户对内容的体验。用户通常使用沉浸式应用程序玩游戏或观看影片等。

这种 iOS 应用程序风格不仅适合游戏，而且其特点也可以增加其他类型任务的吸引力。呈现一个独立环境而不显示大量的文字信息，需要用户集中注意力的任务就十分适合采用沉浸式应用程序。

在本书收录的 6 个项目中，太空泡泡（SpaceBubble）如图 1-9 所示和铅锤（PlumbBob）如图 1-10 所示。

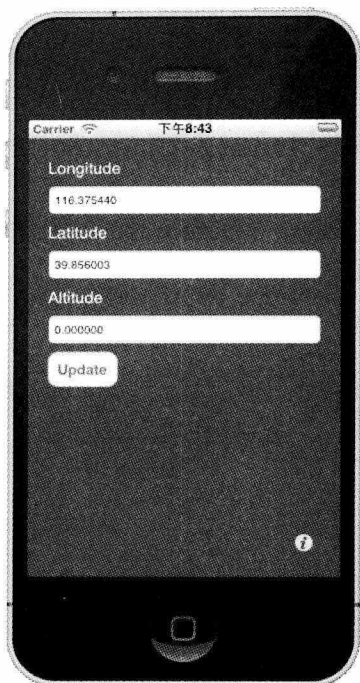


图 1-7 我的位置主画面

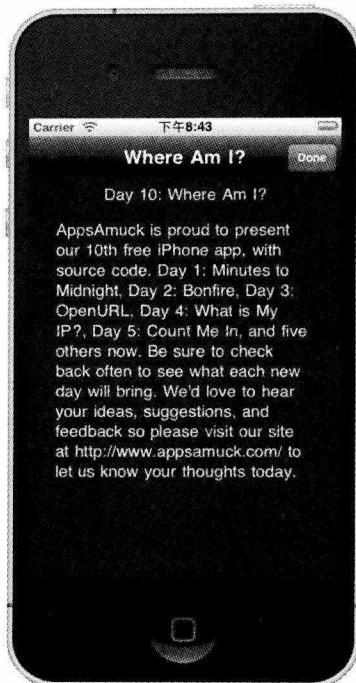


图 1-8 我的位置背后画面

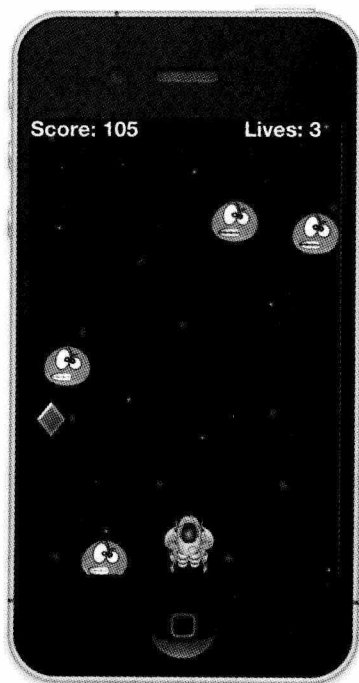


图 1-9 太空泡泡

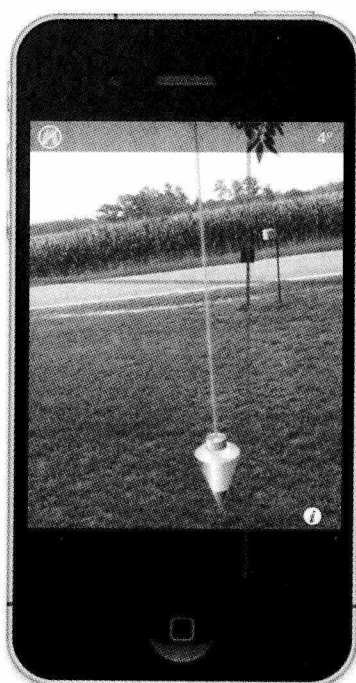


图 1-10 铅锤



1.4 iOS SDK 和开发工具

苹果公司 2008 年 2 月开始为第三方开发商提供 iOS 软件开发工具包，并于 2008 年 3 月 6 日发布此软件开发工具包。允许开发者开发 iPhone 和 iPod Touch 的应用程序，并对其进行测试。这里包括了 Xcode 开发工具、iPhone SDK 和 iPhone 手机模拟器。但是开发者必须在付费后加入 iPhone 手机开发计划，才能发布应用程序。

1.4.1 iOS SDK 介绍

第一个 Beta 版本是 iPhone SDK 1.2b1 (build 5A147p)，它在发布后就立即能够使用。但是同时推出的 App Store 所需要的固件更新则到 2008 年 7 月 11 日才发布。目前，iOS SDK 5 Beta 7 版本已经发布。iOS SDK 主要包括：

- UIKit
- 触控，多点触控事件和控制
- 重力加速计
- 本地化
- 相机支持
- 音频播放和录制
- 视频播放和录制
- OpenAL 混音及录音
- QuartzCore
- OpenGL ES
- SQLite 嵌入式数据库
- 位置服务
- Google 地图 API
- 线程
- TCP/IP 协议套接字
- 电源管理
- 文件系统安全

1.4.2 开发工具 Xcode 介绍

自从 Xcode 3.1 发布以后，Xcode 就成为了 iPhone 软件开发工具包的开发环境。Xcode 可以开发 Mac OS X 和 iOS 应用程序，并且 Xcode 的版本是与 SDK 相对应的。例如：Xcode 3.2.5 与 iOS SDK 4.2 对应，Xcode 4.1 与 iOS SDK 4.3 对应，Xcode 4.2 与 iOS SDK 5 对应。

Xcode 4.1 之前还有一个配套使用的工具 Interface Builder，它是 Xcode 套件的一部分，用来设计窗体和视图，通过它可以“所见即所得”地拖曳控件、定义事件等，其数据以 XML 的形式存储在.xib 文件中。在 Xcode 4.1 之后，Interface Builder 成为了 Xcode 的一部分，与 Xcode 集成在一起。



1.4.3 Xcode 安装和卸载

由于本书采用的是 Xcode 3.2.5 和 iOS SDK4.2, 因此这里介绍一下 Xcode 3.2.5 的安装和卸载。事实上就安装和卸载过程而言, Xcode 3.2.5 与其他的版本没有什么区别。

首先要获得 Xcode3.2.5 安装文件, 可以在苹果官方网站 <http://developer.apple.com/devcenter/ios/index.action#downloads> 下载, 但是必须要有开发者账号才可以下载, 下载的文件是 `xcode_3.2.5_and_ios_sdk_4.2_final.dmg`。对于环境的要求: Mac OS X 10.6 以上, 安装占用空间 6.44G 左右。

安装过程是: 双击 `xcode_3.2.5_and_ios_sdk_4.2_final.dmg` 文件, 打开如图 1-11 所示的安装帮助界面。在这个画面可以先看看安装文档, 这对于安装是有帮助的。

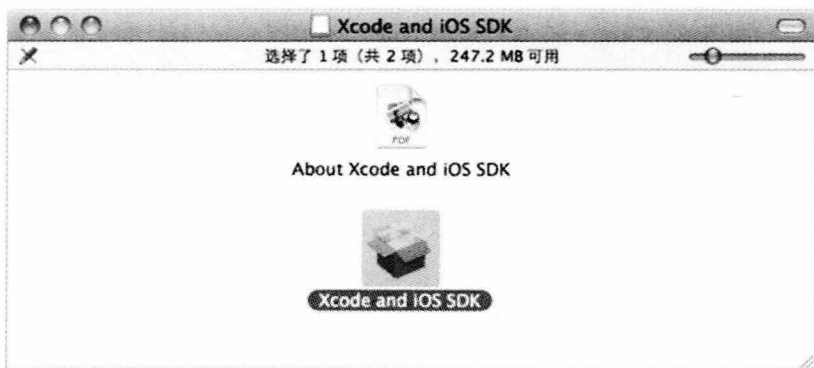


图 1-11 安装帮助画面

单击 Xcode and iOS SDK 安装包, 则进入安装许可等画面, 通过这些许可之后, 进入如图 1-12 所示的选择安装磁盘画面。



图 1-12 选择安装磁盘



单击“继续”按钮则进入安装内容选择画面，如图 1-13 所示，选中除 Mac OS X 10.4 SDK 以外的全部复选框。

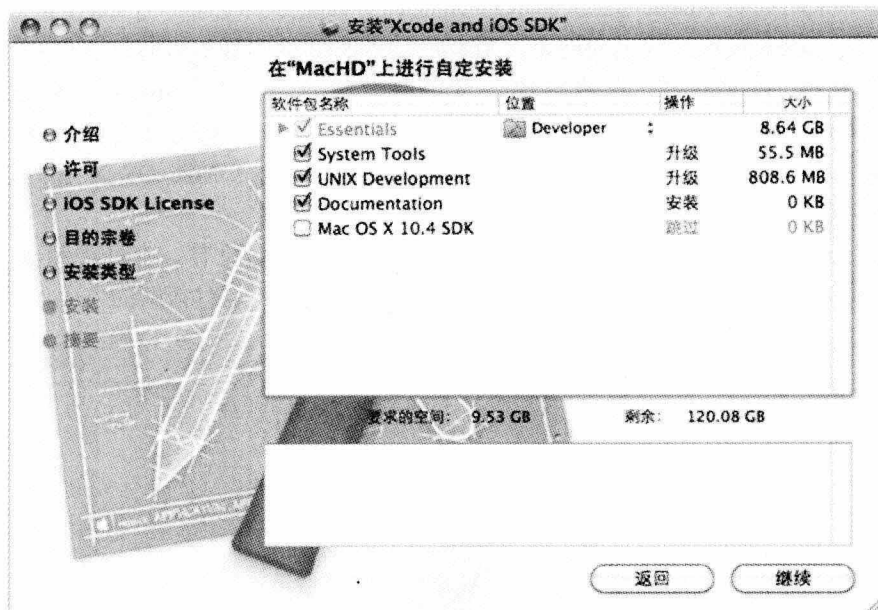


图 1-13 安装内容选择

单击“继续”按钮则进入安装目录更改选择画面，如图 1-14 所示，可以单击“更改安装位置”按钮，变更安装目录。

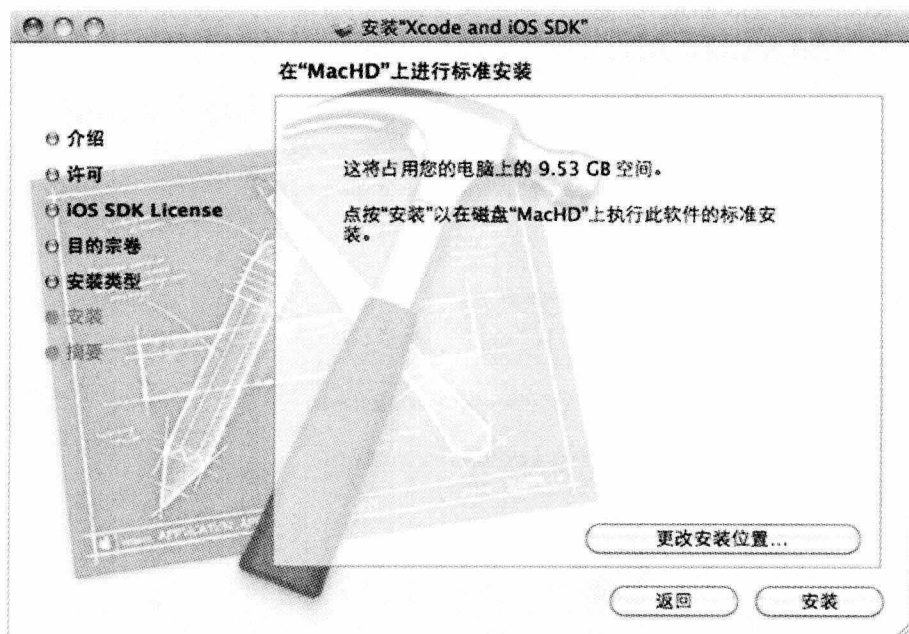


图 1-14 更改安装目录