

# Xcode实战:Apple平台开发实用技术、技巧及最佳流程



[Xcode实战:Apple平台开发实用技术、技巧及最佳流程\\_下载链接1](#)

著者:弗里茨·安德森 (Fritz Anderson)

出版者:机械工业出版社

出版时间:2016-1-20

装帧:平装

isbn:9787111525509

本书将会帮助你使用Xcode6工具提高工作效率，编写优秀的代码并能使用全新的iOS8和OSXYosemite特性，包括Apple全新的Swift编程语言。这是一本专注于介绍需要深入

掌握且每天都要使用的Xcode6书籍。作者发现了更佳的方法介绍storyboard、instrument、构建和编译代码，帮助你应用新特性：从InterfaceBuilderLiveRendering到ViewDebugging和XCTestPerformanceTesting。

本书深入浅出地介绍了Xcode 6的使用，Apple平台开发经验丰富的作者通过三个示例程序，分别介绍了命令行开发，iPhone app开发和Mac app开发的\*佳流程，在这些开发流程中，详细介绍了Xcode 6的各项功能。示例程序全部用新的Swift语言编写。通过阅读本书，你可以掌握Xcode 6的使用方法和常见功能。熟练使用Xcode 6的各项功能可以极大的提高工作效率。

全书分为四部分，共28章，第一部分（第1~7章）主要介绍Xcode的安装、命令行应用程序基本开发技术及调试技巧，涉及简单的工作流程和被动调试、主动调试、编译、添加目标库、版本控制等；第二部分（第8~18章）介绍一个小型iPhone应用程序，以及如何使用Apple开发者工具构建这个应用，涉及设计Core Data计划、创建控制器、构建新视图、新视图中的自动布局、添加表元素、添加编辑器视图、单元测试、度量、分析和配置等；第三部分（第19~23章）详细讲解OS X开发，介绍相应链、Interface Builder、绑定、本地化、程序包和属性列表等相关技术的使用方法和技巧；第四部分（第24~28章）深入探讨Xcode的文档系统、构建系统、Instruments、调试器，同时总结了Xcode开发过程中的一些常见陷阱及实用技巧。此外，书号的两个附录还提供了帮助你掌握构建系统的参考信息，以及寻找帮助和支持的途径。

作者介绍:

Fritz Anderson融合自己30余年Apple平台开发和顾问经验，全方位展示如何使用Apple的集成开发环境在进行高质量产品开发的过程中达到事半功倍的效果。

目录: 译者序

前言

第一部分 起步

第1章 获取Xcode 2

1.1 开始之前 2

1.2 安装Xcode 3

1.3 移除Xcode 4

1.4 Apple开发者计划 5

1.5 下载Xcode 6

1.6 额外需要下载的内容 7

1.7 小结 8

第2章 初识Xcode 9

2.1 启动Xcode 9

2.2 Hello World 11

2.2.1 新建项目 11

2.2.2 让Xcode安静下来 13

2.2.3 构建和运行 14

2.2.4 货真价实的程序 15

2.3 删除项目 15

2.4 小结 15

第3章 简单的工作流程和被动调试 16

3.1 创建项目 16

3.2 构建 19

3.3 运行 21

|                     |    |
|---------------------|----|
| 3.4 简单调试            | 22 |
| 3.5 小结              | 23 |
| 第4章 主动调试            | 24 |
| 4.1 一个简单的测试用例       | 24 |
| 4.2 开始主动调试          | 24 |
| 4.2.1 设置断点          | 25 |
| 4.2.2 变量窗格          | 26 |
| 4.2.3 单步调试          | 27 |
| 4.3 解决这个问题          | 28 |
| 4.3.1 行为            | 28 |
| 4.3.2 解决方法          | 31 |
| 4.4 小结              | 31 |
| 第5章 编译              | 32 |
| 5.1 编译              | 32 |
| 5.2 动态载入            | 38 |
| 5.3 Xcode和Clang     | 38 |
| 5.3.1 本地分析          | 39 |
| 5.3.2 跨函数分析         | 41 |
| 5.3.3 索引            | 42 |
| 5.4 Swift           | 43 |
| 5.5 编译产品            | 47 |
| 5.5.1 中间产品          | 47 |
| 5.5.2 预编译           | 49 |
| 5.6 小结              | 51 |
| 第6章 添加库target       | 52 |
| 6.1 添加target        | 52 |
| 6.2 target关系        | 54 |
| 6.2.1 添加文件到target中  | 54 |
| 6.2.2 target中的头文件   | 56 |
| 6.3 从属target        | 56 |
| 6.3.1 添加库           | 57 |
| 6.3.2 调试依赖target    | 59 |
| 6.4 小结              | 59 |
| 第7章 版本控制            | 60 |
| 7.1 采取控制            | 61 |
| 7.2 文件状态            | 63 |
| 7.3 首次提交            | 65 |
| 7.4 使用远程版本库         | 66 |
| 7.4.1 本地建立一个“远程”版本库 | 68 |
| 7.4.2 将文件推送到远程服务器   | 69 |
| 7.5 合并与冲突           | 70 |
| 7.5.1 User A        | 70 |
| 7.5.2 User B        | 73 |
| 7.5.3 返回User A      | 75 |
| 7.6 版本编辑器           | 78 |
| 7.6.1 Comparison    | 78 |
| 7.6.2 Blame         | 80 |
| 7.6.3 Log           | 81 |
| 7.7 分支              | 81 |
| 7.8 小结              | 83 |
| 第二部分 iOS应用程序的生命周期   |    |
| 第8章 开始制作iOS应用程序     | 86 |
| 8.1 规划应用程序          | 86 |
| 8.1.1 模型-视图-控制器     | 86 |

- 8.1.2 模型 87
- 8.1.3?视图 87
- 8.1.4 控制器 88
- 8.2 开始制作一个全新的iOS项目 89
- 8.3 项目中都包含什么 91
- 8.4 小结 93
- 第9章 iOS应用程序：模型 94
- 9.1 实现模型 94
- 9.1.1 实体 95
- 9.1.2 属性 95
- 9.1.3 关系 97
- 9.2 托管对象类 100
- 9.2.1 创建类——错误的方法 100
- 9.2.2 为什么按照Xcode的方式做是个错误 102
- 9.2.3 正确的方法——使用mogenerator 102
- 9.3 准备 105
- 9.3.1 Utilities 105
- 9.3.2 扩展类 105
- 9.3.3 passer\_rating 107
- 9.4 定制Core Data类 108
- 9.4.1 在项目中添加Game数据 108
- 9.4.2 在项目中添加Passer数据 109
- 9.4.3 一些测试数据 110
- 9.4.4 源代码控制和产品文件 112
- 9.5 让模型更容易调试 114
- 9.6 小结 115
- 第10章 iOS应用程序：控制器 116
- 10.1 Objective-C中的符号重命名 116
- 10.1.1 重构Objective-C方法名 117
- 10.1.2 重构类名 117
- 10.2 在Swift中重命名类 118
- 10.3 编辑View Controller 119
- 10.3.1 表视图 120
- 10.3.2 构建Passer List 121
- 10.3.3 新建Passer 121
- 10.4 生命周期问题及处理方法 122
- 10.5 真正的Passer Rating 123
- 10.5.1 另外一个bug 123
- 10.5.2 运行Passer Rating 127
- 10.6 小结 128
- 第11章 构建新视图 129
- 11.1 下一个视图控制器 129
- 11.1.1 如果想要添加新视图控制器 129
- 11.1.2 storyboard、scene和segue 130
- 11.2 创建视图 133
- 11.2.1 Passing中的Outlet和Assistant 134
- 11.2.2 Billboard视图 136
- 11.2.3 链接视图与视图控制器 137
- 11.2.4?特定场合下的自动布局 138
- 11.2.5?各种标签 140
- 11.2.6 清理 142
- 11.3 表视图 144
- 11.4 outlet 145
- 11.4.1 连接outlet 146

- 11.4.2 检查连接 147
- 11.4.3 连接GameListController 148
- 11.4.4 代码补全和片段 149
- 11.4.5 代码片段 150
- 11.5 测试Billboard View 152
- 11.6 小结 153
- 第12章 新视图的自动布局 154
- 12.1 为什么选择自动布局 154
- 12.1.1 Autoresizing的局限 155
- 12.1.2 Auto Layout 155
- 12.1.3 需要牢记的问题 155
- 12.2 重新制作Player Billboard 156
- 12.3 重构子视图布局 157
- 12.3.1 Playground 158
- 12.3.2 StatView 160
- 12.3.3 安装StatView 164
- 12.4 设计约束 165
- 12.5 两行、两个标签 167
- 12.6 实际的约束 169
- 12.6.1 默认 (Any/Any) 170
- 12.6.2 Any Height (非Compact) 170
- 12.6.3 Landscape (wAny/hCompact) 171
- 12.6.4 解决问题 171
- 12.6.5 细节修改 171
- 12.7 小结 172
- 第13章 添加表格单元格 173
- 13.1 Game表 173
- 13.1.1 Table View中的outlet 173
- 13.1.2 添加所需的Protocol方法 174
- 13.1.3 添加Model-to-View的支持 175
- 13.1.4 原型元素 176
- 13.2 Game Table: 第一次运行 177
- 13.3 自定义表格单元格 179
- 13.4 添加一些图形 182
- 13.4.1 带有图片的单元格 182
- 13.4.2 将Image视图与图片联系在一起 182
- 13.4.3 Assets目录 183
- 13.4.4 向Assets Catalog中添加图片 184
- 13.4.5 图标和启动图片 186
- 13.5 小结 188
- 第14章 添加编辑器 189
- 14.1 计划 189
- 14.2 添加模型场景 189
- 14.2.1 嵌入的View Controller 191
- 14.2.2 将Passer List与Editor相连 193
- 14.2.3 Static Table Cells 194
- 14.3 编辑器视图控制器 195
- 14.3.1 编辑器表 195
- 14.3.2 将数据传递给编辑器 197
- 14.3.3 从编辑器中获得数据 198
- 14.4 segue 200
- 14.5 小结 201
- 第15章 单元测试 202
- 15.1 Test Navigator 203

|                               |     |
|-------------------------------|-----|
| 15.2 测试CSV Reader             | 204 |
| 15.2.1 CSV 测试代码               | 205 |
| 15.2.2 测试数据                   | 209 |
| 15.2.3 运行测试程序                 | 210 |
| 15.3 测试和调试器                   | 211 |
| 15.4 添加测试类                    | 213 |
| 15.5 异步测试                     | 216 |
| 15.5.1 测试异步代码                 | 217 |
| 15.5.2 文档                     | 217 |
| 15.6 XCTest断言                 | 218 |
| 15.6.1 简单测试                   | 218 |
| 15.6.2 等价                     | 219 |
| 15.6.3 异常                     | 219 |
| 15.7 小结                       | 220 |
| 第16章 度量和分析                    | 221 |
| 16.1 速度                       | 221 |
| 16.1.1 Debug 调试器              | 222 |
| 16.1.2 Instruments            | 224 |
| 16.2 XCTest和性能                | 231 |
| 16.3 内存                       | 232 |
| 16.4 小结                       | 233 |
| 第17章 iOS扩展                    | 234 |
| 17.1 添加Today Target           | 235 |
| 17.2 设计挂件                     | 235 |
| 17.3 框架内的共享库                  | 239 |
| 17.4 Today扩展                  | 244 |
| 17.5 构建依赖                     | 247 |
| 17.6 结果                       | 248 |
| 17.7 小结                       | 248 |
| 第18章 配置                       | 249 |
| 18.1 Apple开发者计划               | 249 |
| 18.1.1 通用 (App Store) 计划      | 250 |
| 18.1.2 企业计划                   | 250 |
| 18.2 iOS的配置                   | 251 |
| 18.2.1 你会看到什么                 | 251 |
| 18.2.2 注册你的App                | 253 |
| 18.2.3 保护资源                   | 254 |
| 18.2.4 提交iOS应用程序              | 255 |
| 18.3?功能编辑器                    | 257 |
| 18.3.1 OS X独有的功能              | 257 |
| 18.3.2 iOS和OS X中都有的功能         | 257 |
| 18.3.3 iOS功能                  | 258 |
| 18.4 OS X沙箱                   | 259 |
| 18.4.1 为什么要选择沙箱               | 261 |
| 18.4.2?沙箱的缺点                  | 261 |
| 18.5 Gatekeeper 和Developer ID | 262 |
| 18.5.1 获取Developer ID         | 262 |
| 18.5.2 使用 Developer ID        | 263 |
| 18.5.3 限制                     | 264 |
| 18.6 发布构建                     | 264 |
| 18.6.1 基本的构建设置                | 265 |
| 18.6.2 调整构建设置                 | 265 |
| 18.6.3 构建                     | 267 |
| 18.7 小结                       | 268 |

### 第三部分 Mac OSX开发中的Xcode

#### 第19章 开始制作OS X应用程序 270

##### 19.1 目标 270

##### 19.2 开始 271

##### 19.3 模型 274

##### 19.3.1 从iOS移植 274

##### 19.3.2 添加实体 275

##### 19.4 制作菜单 278

##### 19.4.1 Target/Action 279

##### 19.4.2 First Responder 280

##### 19.4.3 将数据载入LeagueDocument 281

##### 19.4.4 适应托管文档 281

##### 19.4.5 测试命令 283

##### 19.4.6 为League 数据确定一个类型 284

##### 19.4.7 确定App如何处理League文件 285

##### 19.4.8 应用程序和文档图标 286

##### 19.5 小结 287

#### 第20章 绑定：连接OS X 应用程序 288

##### 20.1 OS X中的Storyboard Segue 288

##### 20.2 构建文档窗口 290

##### 20.2.1 载入窗口 290

##### 20.2.2 表视图 291

##### 20.3 填充表——绑定 294

##### 20.3.1 对象控制器 294

##### 20.3.2 将表绑定到Team 296

##### 20.3.3 将列绑定到Team属性 297

##### 20.4 League文档数据的结构 299

##### 20.4.1 从League表到Source List 299

##### 20.4.2 捕获Team的选择 301

##### 20.4.3 从Team到表 303

##### 20.4.4 Passer部分 304

##### 20.5 小结 311

#### 第21章 本地化 312

##### 21.1 本地化的工作原理 312

##### 21.2 添加本地化 313

##### 21.2.1 Base Localization 313

##### 21.2.2 为什么需要 Base Localization 314

##### 21.3 哪些需要本地化 315

##### 21.3.1 游戏细节视图：布局 315

##### 21.3.2 游戏细节视图：代码 317

##### 21.3.3 模块和命名空间 319

##### 21.4 法语本地化 320

##### 21.4.1 添加语言环境 320

##### 21.4.2 由简入繁：Credits.rtf 323

##### 21.4.3 本地化Main.storyboard 325

##### 21.4.4 本地化资源 328

##### 21.4.5 本地化程序字符串 329

##### 21.4.6 genstrings 330

##### 21.4.7 xlf文件 331

##### 21.4.8 其他 332

##### 21.5 本地化系统字符串 332

##### 21.6 小结 334

#### 第22章 程序包 335

##### 22.1 简单的包：RTFD 335

- 22.2 bundle 336
- 22.3 应用程序的bundle 337
- 22.4 Info.plist File 338
- 22.5 Info.plist中用于应用程序的键 340
  - 22.5.1 用于iOS和OS X的键 340
  - 22.5.2 OS X中使用的键 342
  - 22.5.3 iOS中使用的键 345
- 22.6 小结 347
- 第23章 属性列表 348
  - 23.1 属性列表数据类型 348
  - 23.2 编辑属性列表 350
    - 23.2.1 属性列表编辑器 352
    - 23.2.2 为什么不使用属性列表编辑器 356
  - 23.3 其他格式 358
    - 23.3.1 文本属性列表 358
    - 23.3.2 二进制属性列表 359
    - 23.3.3 JSON 360
  - 23.4 特殊的属性列表 361
  - 23.5 小结 362
- 第四部分 Xcode任务集
- 第24章 Xcode中的文档 364
  - 24.1 Quick Help 364
    - 24.1.1 Inspector 364
    - 24.1.2 弹出窗口 365
  - 24.2 快速打开 366
  - 24.3 帮助 367
  - 24.4 文档窗口 368
    - 24.4.1 侧边导航栏 368
    - 24.4.2 侧边栏的内容表 368
    - 24.4.3 类信息 368
    - 24.4.4 查找和导航 369
  - 24.5 保持同步 371
  - 24.6 自定义的Quick Help 373
    - 24.6.1 C语言系列的文档 373
    - 24.6.2 Doxygen 375
    - 24.6.3 运行Doxygen 380
    - 24.6.4 安装文档集合 380
  - 24.7 Swift和reStructuredText 381
  - 24.8 小结 383
- 第25章 Xcode构建系统 384
  - 25.1 Xcode如何组织构建过程 384
  - 25.2 构建变量 387
  - 25.3 设置的层级 388
  - 25.4 编辑构建变量 389
  - 25.5 配置 390
  - 25.6 配置文件 392
    - 25.6.1 创建配置文件 392
    - 25.6.2 SDK和指定的架构设置 393
    - 25.6.3 预处理xcconfig文件 393
  - 25.7 命令行工具 394
    - 25.7.1 xcodebuild 394
    - 25.7.2 xcode-select 395
    - 25.7.3 xcrun 396
  - 25.8 自定义构建规则 397

- 25.9 构建日志 398
- 25.10 简单的构建记录 400
- 25.11 小结 408
- 第26章 Instruments 409
  - 26.1 Instruments是什么 409
  - 26.2 运行Instruments 410
  - 26.3 记录文档窗口 411
    - 26.3.1 工具栏 412
    - 26.3.2 记录区域 414
    - 26.3.3 Detail区域 415
    - 26.3.4 Extended Detail区域 415
    - 26.3.5 库 418
  - 26.4 追踪 419
    - 26.4.1 记录 419
    - 26.4.2 保存和重新打开 422
    - 26.4.3 不启动instrument记录 422
  - 26.5 Instruments 422
    - 26.5.1 Behavior 423
    - 26.5.2 Core Data 423
    - 26.5.3 Dispatch 423
    - 26.5.4 Filesystem 423
    - 26.5.5 Graphics 424
    - 26.5.6 Input/Output 424
    - 26.5.7 Master Tracks 424
    - 26.5.8 Memory 425
    - 26.5.9 系统 425
    - 26.5.10 System-iOS Energy Instruments 427
    - 26.5.11 线程/锁 427
    - 26.5.12 记录 428
    - 26.5.13 UI Automation 428
    - 26.5.14 用户界面 428
  - 26.6 自定义Instruments 429
  - 26.7 模板 430
    - 26.7.1 全部平台 430
    - 26.7.2 仅限iOS 430
    - 26.7.3 仅限Mac 431
  - 26.8 小结 431
- 第27章 调试 432
  - 27.1 Scheme选项 432
    - 27.1.1 Info 432
    - 27.1.2 参数 433
    - 27.1.3 选项 433
    - 27.1.4 诊断 435
  - 27.2 使用断点做更多的工作 435
  - 27.3 视图层级 437
  - 27.4 lldb命令行 439
  - 27.5 小技巧 441
  - 27.6 小结 444
- 第28章 零散的知识 445
  - 28.1 技巧 445
    - 28.1.1 一般技巧 445
    - 28.1.2 代码折叠 449
    - 28.1.3 Assistant编辑器 450
    - 28.1.4 Instruments和调试 452

28.1.5 构建 452  
28.2 陷阱 454  
第五部分 附录  
附录A 一些构建变量 460  
附录B 资源 474  
• • • • • ([收起](#))

[Xcode实战:Apple平台开发实用技术、技巧及最佳流程\\_下载链接1\\_](#)

## 标签

iOS

编程

翻译

新手

前端开发

入门

OSX

Mac

## 评论

新手入门不错，高手略过

-----  
[Xcode实战:Apple平台开发实用技术、技巧及最佳流程\\_下载链接1\\_](#)

# 书评

-----  
[Xcode实战:Apple平台开发实用技术、技巧及最佳流程 下载链接1](#)