

Swift语言指南

这份指南汇集了Swift语言主流学习资源，并以开发者的视角整理编排。

GitHub: [ipader/SwiftGuide](https://github.com/ipader/SwiftGuide) | 网站: <http://dev.swiftguide.cn> | 欢迎开发者一起[维护](#), 或[反馈/投稿](#)

@SwiftLanguage 更新于 2014-11-30, 更新内容详见 [《2014-11-30收录周报》](#)



目录

- [Swift文档](#)
 - [Welcome to Swift](#)
 - [Swift Programming Language \(中译\)](#)
 - [Using Swift with Cocoa and Objective-C \(中译\)](#)
 - [App Extension Programming Guide \(中译\)](#)
 - [Swift Blog - Apple Developer \(中译\)](#)
 - [iOS Human Interface Guidelines \(中译\)](#)
- [Swift教程](#)
 - [官方教程](#)
 - [快速入门](#)
 - [教程精选](#)
 - [开发技巧](#)
 - [相关文档](#)
- [Swift项目](#)
 - [类库](#)
 - [示例项目](#)
 - [完整项目](#)
- [Swift网站](#)
- [相关工具](#)
- [开放平台](#)
 - [文档](#)
 - [示例项目](#)
 - [实用资源](#)
- [Apple Watch指南](#)
 - [文档](#)
 - [WatchKit Programming Guide \(中译\)](#)
 - [Apple Watch Human Interface Guidelines \(中译\)](#)
 - [Watch教程](#)
 - [快速入门](#)

- [教程精选](#)
 - [Watch项目](#)
 - [媒体文章](#)
 - [其它相关](#)
-

Swift文档

1. [Welcome to Swift](#)

苹果针对Swift开发者官方文档入口。其中包括: [Swift概括](#), [Swift Programming Language](#), [Using Swift with Cocoa and Objective-C](#)

2. [Swift Programming Language](#)

- 苹果官方文档: [在线版\(英文\)](#) [iBooks版\(英文\)](#)
- 爱好者翻译版: [在线版\(中文\)](#) (By @Swift中文翻译组) | [PDF版](#) (By @CocoaChina) | [百度阅读版](#) (By 小岂子)
- [Basic Operators - Nil Coalescing Operator](#)解读
 1. [Swift中Nil Coalescing运算符的使用技巧](#) By @DevTalking
 2. [Swift's Nil Coalescing Operator In Xcode 6 Beta 5](#) By Jameson Quave
- [Swift 编程风格指南](#)(译文来自 [swiftist.org](#)): 本风格指南的目标是让Swift代码更简洁、可读更强。原文: [《The Official raywenderlich.com Swift Style Guide》](#)
- [《The Swift Programming Language》](#) 读簿: "读书就是要取其精华, 去其糟粕、无用、简单和已知的内容。By @Hawstein"。站在作者自己的理解能力上, 应该是一份很好的去繁求简读薄。不过, 每个初学者基础不同, 理解能力也千差万别。因此, 对于浓缩版, 当有不理解时, 应该对照原版章节再进一步细读, 千万不能蒙混。

3. Using Swift with Cocoa and Objective-C

- 苹果官方文档: [在线版\(英文\)](#) [iBooks版\(英文\)](#)
- 爱好者翻译版: [在线版\(中文\)](#) (By @CocoaChina) | [PDF版](#) (By @CocoaChina)

4. [App Extension Programming Guide](#)

- 应用扩展要点 (App Extension Essentials)

译文	译者	原文
应用扩展如何工作	@DevTalking	Understand How an Extension Works
开发应用扩展	@DevTalking	Creating an App Extension
APP扩展提高你的应用影响力	@DevTalking	App Extensions Increase Your Impact
常见问题的处理方案	@DevTalking	Handling Common Scenarios

- 应用扩展类型 (App Extension Types)

译文	译者 / 校对	原文
Today	@CocoaChina / 唧唧歪歪	Today
Share	@CocoaChina / 张国鹏	Share
Action	@CocoaChina	Action
照片编辑	@CocoaChina	Photo Editing
Finder同步	@星夜暮晨	Finder Sync
文档提供	@星夜暮晨	Document Provider
第三方输入法	@星夜暮晨	Custom Keyboard

5. [Swift Blog - Apple Developer](#)

"值得一提的是，Swift博客是苹果官方网站的第一个blog，这也代表了苹果对开发者和消费者的态度正变得越来越开放。"

译文	译者	原文
Swift REPL入门介绍	@pockry	Introduction to the Swift REPL
可失败构造器	@DevTalking	Failable Initializers
—	-	Building Your First Swift App Video
通过Playground展示一些编码模式	@DevTalking	Patterns Playground
Swift中Optional类型的使用案例分析: valuesForKeys	@DevTalking	Optionals Case Study: valuesForKeys
Swift中的访问控制与protected	@DevTalking	Access Control and protected
Swift中的值类型和参照类型	@DevTalking	Value and Reference Types
WWDC2014大会中的Playground大炮气球示例	@DevTalking	Balloons
在Swift中构建布尔类型	@DevTalking	Boolean
Swift中的文件和初始化	@DevTalking	Files and Initialization
在Swift中使用C语言的指针	@DevTalking	Interacting with C Pointers
* Swift新特性 -- 访问控制 (文档版)	@DevTalking	Access Control

6. iOS Human Interface Guidelines

- 苹果官方文档: [在线版\(英文\)](#), [iBooks版\(英文\)](#)
- 非官方中译版

译文	译者	原文
UI设计基础	@腾讯ISUX	Designing for iOS
设计策略	@腾讯ISUX	Design Principles

- iOS 7人机交互指南（旧版）: [中译汇总](#)

Swift教程

1. 官方教程

- Swift入门(视频翻译 By @赵哲A): [A001.01](#), [A001.02](#), [A001.03](#), [A001.04](#): WWDC 2014官方Session 402视频翻译。讲得很细致，口齿清晰，英语发音准确，声音非常好听。
- Swift进阶(视频翻译 By @赵哲A): [A002.01](#), [A002.02](#), [A002.03](#), [A002.04](#), [A002.05](#) WWDC 2014官方Session 403视频翻译。
- [Testing with Xcode](#): 本文的目的在于让测试成为你软件开发的重要组成部分，并使测试更方便并易于使用。

2. 快速入门

- [Swift Cheat Sheet\(PDF\)](#): 形式以代码先行，极简、有效地Swift语言快速学习指南。另一个更新版本: [iOS8 Swift Cheat Sheet and Quick Reference Guide](#)
- [An Absolute Beginner's Guide to Swift](#): 相对于Swift Cheat Sheet带了不少说明，但整个文档不显啰嗦，可读性强
- [Swift Language FAQ](#): Raywenderlich的Swift语言FAQ说清楚了很多问题。这份FAQ确实浅显易读，初学者不可或缺好文章！
- [Strings in Swift](#): 了解使用String的更高级技巧（尤其在Unicode的使用上）。文章中作者附上了示例代码在Playground项目中，似乎Swift开发者们已经慢慢地习惯了结合playground讲授Swift使用小技巧及语言特性。
- 与其它语言对比表: [C vs. Go vs. Swift](#), [C# vs. Swift](#), [Scala vs. Swift](#), [Go vs. Swift](#)
- [iOS 8应用程序扩展开发技巧](#): 比较全面的概括了iOS扩展开发小技巧。By @星夜暮晨 原文[iOS 8 App Extension Development Tips](#)
- [Objective-C开发者对Swift亮点的点评](#): 这篇译文确实不错，包括了常见的亮点。尽管对于Swift相较于Objective C的亮点描述还不够全面，对初学者很受用。 [原文在此](#)
- [开启iOS/Mac开发之旅, 过来人告诉你16件事 \(译文来自@CocoaChina\)](#): "我曾向iOS开发者推荐了 [《Twitter上最值得关注的30个人》](#)，收到了不少开发者的反馈，受此鼓舞，我向知名iOS开发者和设计师询问了这样一个问题--回到你开始iOS/Mac app开发的时候，你以现在的角度会给“最初的你”哪些建议。" 译文来自，英文原文 [《13 Things You Must Know When Starting Out in iOS/Mac Development》](#)
- [74个Swift标准库 \(译文\)](#): Swift包含了74个内置函数，但在The Swift Programming Language一书中只介绍了其中的7个，其它的都没有在文档中体现。"文中作者没有提及他是怎么发现这么多未在文档中体现的内置函数的，估计是反编译的结果。我测试了好多个都能用，而且Xcode还会给出语法提示：)" by @李洁信
- [ochococo/Design-Patterns-In-Swift](#): 这个项目分享了Swift编程中如何使用常用设计模式。作者提供的Playground示例及常用设计模式的简单介绍。

- [马上着手开发 iOS 应用程序 \(Start Developing iOS Apps Today\)](#): 来自苹果官方文档
- [WKWebView](#): "WKWebView 是现代 WebKit API 在 iOS 8 和 OS X Yosemite 应用中的核心部分。它代替了 UIKit 中的 UIWebView 和 AppKit 中的 WebView，提供了统一的跨双平台 API。" 作者: [Mattt Thompson](#)
- [Core Location in iOS 8](#): 有关iOS 8核心位置服务介绍。文章从权限、室内位置跟踪、CLVisit等角度做了较全面的介绍。

3. 教程精选

- 中译精选

译文	译者	原文	来源
Swift自适应布局 (Adaptive Layout) 教程 (一)	@DevTalking	Beginning Adaptive Layout Tutorial	raywenderlich
如何使用iOS 8的虚化效果	@CocoaChina	iOS 8 Visual Effects Tutorial	raywenderlich
Web工程师和设计师必须要知道的iOS 8的十个变化	@罗罗磊磊	iOS 8 and iPhone 6 for web developers and designers: next evolution for Safari and native webapps	mobilexweb
如何在Swift中创建Action扩展	@DevTalking	Action Extension in Swift	nsnerd.co
iOS8 Day-by-Day :: Day2 :: 分享应用扩展	@DevTalking	iOS8 Day-by-Day :: Day 2 :: Sharing Extension	shinobicontrols
Swift 的强大之处	李子轩	The Power of Swift	objc.io
结构体和值类型	@Onetaway	A Warm Welcome to Structs and Value Types	objc.io
Swift方法的多面性	@唯木念	The Many Faces of Swift Functions	objc.io
Swift 的函数式 API	@请叫我汪二	Functional APIs with	objc.io

		Swift	
Playground快速原型制	Programmer.Du	Rapid Prototyping in Swift Playgrounds	objc.io
在Swift中自定义下标	@DevTalking	Custom Subscripts in Swift	codingexplorer
如何用 Swift 语言构建一个自定义控件	@BeyondVincent	How To Make a Custom Control in Swift	raywenderlich
UIKit重力学（二，二）	@DevTalking	UIKit Dynamics Tutorial in Swift	raywenderlich
Swift中的延迟加载	@Swiftist	Lazy Initialization with Swift	mikebuss
The Swift Programming Language: 实验项目相关解决方案	@CocoaChina	The Swift Programming Language: Solutions to Experiments	swiftcast.tv

- [S008 - iOS 应用基础](#): 教程对于最基础的UI编程讲得特别细，时有反复强调Swift基础特性。个人感觉这个适合无iOS UI编程经验的同学，甚至也适合无Swift基础的同学。By [@赵哲A](#)
- [Getting Your App Ready for iOS 8](#): 让你的应用程序适配iOS 8，这篇文章总结的比较完整。By [@Janselz](#)
- [VectorPDFSupportTest](#): "Xcode 6 支持 vector PDF 来支持多尺寸的图片，虽然是伪矢量图，但我还是写了点东西来把这东西说清楚。" By [@你全家都快到碗里来](#)
- [nettlep/learn-swift](#): 学习《Swift Programming Language》的同时，配合运行这些playgrounds，对于快速掌握Swift最合适不过了。
- [Swift Reference Playground/Swift Operators Playground](#): "Swift Reference is a handy playground file that can be used when you are starting to learn swift.It covers the basic syntax definitions and different ways to use" 初学Swift语言的同学一定先玩玩这个，可以让您更快感性的了解Swift语言语法、操作符及语言新特点（代码涵盖地比较完整）。
- [ShinobiControls/iOS8-day-by-day](#): 追剧iOS 8开发，文章几乎篇篇带示例项目。iOS 8开发者进阶必备！
- [VirtualGS教程 \(By @林泰前\)](#): 几十年的老程序员，资深的iOS开发工程师，把自己儿子培养成苹果应用商店最年少的应用开发者，哪里能找到这样优质的编程老师？教程

包括《How to》系列连载和《图形编程》示例项目

- [Swift Weather APP](#):“林永坚老师将带领大家使用Swift语言开发一个完整的天气 iOS APP。同时大家能够学习到Interface Builder、CocoaPods、Core Location、AFNetworking的使用, 以及如何通过Swift调用Objective-C组件”
- [使用Swift开发iOS8 App实战](#): 实战学习是最有效的编程学习方法, 推荐[@林永坚](#)老师的实战视频。教程代码: 1.[ChineseZodiac](#), 2. [LoveFinder](#), 3. [BeautyGallery](#)
- [Developing iOS 8 Apps Using Swift \(I\) – Animations, Audio, and Custom Table View Cells](#): "这个教程大概7篇blog, 一步步讲解了一个『itunes music search app』的制作对于async http, api delegate, 异步图片, 动画, 代码分层, interface builder都做了清晰的介绍。" By [gaohailang](#)
- [Replicating Twitter's bird zoom startup animation \(in Swift!\)](#) (源代码): 这是一个有关用Swift开发简单动画效果的编程示例, 示例开发灵感来源于近期的Twitter iOS版小鸟启动动画。
- [WWDC2014 详解OSX/iOS8/Swift语言](#): "极客学院全国首发课程, 详解OSX/iOS8新特性/崭新编程语言Swift! 极客学院团队通宵录制、上传, 只为在这一刻把最新最实战的课程呈现给开发者"
- [从零开始学Swift计时器App开发](#) (源代码): "通过完成此教程, 我对Swift语言的理解也更进了一步。Swift是一门全新的语言, 作为开发者, 我们需要不断加深对这门语言的理解, 并灵活使用语言提供的特性来编程。..." by [@李洁信](#)
- [如何用Swift写UIDynamic](#): 如何用Swift写UIDynamic。代码虽短, 功能性完整。可读性也很赞! By [@戴铭](#)
- [Swift如何检查系统版本](#): 介绍使用Swift语言检测操作系统版本的方法。译者: [@pockry](#) 原文: [Swift System Version Checking](#) 作者: [Mattt Thompson](#)
- [How to build a nice Hamburger Button](#): 小小的按钮, 无论在设计上, 还是代码上, 都进行了精雕细琢。期待作者能尽早发布更完整的Hamburger Buttons. 源代码: [robb/hamburger-button](#)
- [Hamburger Button Animation](#): 又一个实用的Hamburger Button。另外, 看了作者的[博客](#), 感觉他对iOS Animation编程非常有经验, 强烈推荐关注。源代码: [fastred/HamburgerButton](#)
- [在Playground中添加说明文档](#): 教你如何在Playground中添加说明文档。By [@DevTalking](#)
- [使用 Swift 和 AVOSCloud 构建 iOS 应用](#): "使用 AVOSCloud SDK 和 Swift 构建 iOS 应用 | AVOS Cloud Blog" By [@AVOS江宏](#) | 这就是业界的速度, 先机才是致胜法宝。
- [为iPhone6设计自适应布局 \(一、二\)](#): "当你们学习完这篇文章后, 你们应该会比较自如的使用 storyboard、constraints、size classes 这三个Apple在Xcode里提供的工具, 去探索和构建巧妙的自适应布局。" 译者: [@DevTalking](#) 原文: [ADAPTIVE LAYOUTS FOR IPHONE 6](#)
- [iOS App集成Apple Pay教程](#): "Apple Pay还给开发者带来了处理支付的新渠道, 用户

将期望在应用中使用它，因为它将验证和交易极端简化，仅需手指轻轻一触即可完成。如果应用里面有涉及到交易，开发者很有必要集成Apple Pay。那么如何将Apple Pay功能集成到你的应用里呢？" 可惜Demo是Objective-C版本的。译者：@pockry 原文：[Integrating Your iOS App with Apple Pay](#)

- [WWDC 2014 Session笔记 - 可视化开发, IB 的新时代](#): "通过一个简单例子介绍了Xcode 6 的 IB 中自定义 view 的基本使用。" 作者：@onevc
- [CMDeviceMotion](#): 一篇关于CMDeviceMotion类解读及使用方法的文。对于iPhone三轴陀螺仪和加速感应器，以及M7/M8协处理器，核心运动框架（The Core Motion framework）提供了完整的API支持。然而文章更重要的意义是启发开发者重视感应器，并结合场景开发新颖好玩的应用。
- [如何在Swift中运用Text Kit框架](#): "现在iOS8发布了，在文本渲染方面在延续了之前强大功能的基础上，又提升了其易用性。" 译者：@DevTalking 原文：[Text Kit Tutorial in Swift](#)
- [Swift中的函数柯里化\(Function Currying\)](#): 柯里化提供了强大的动态函数创建方法，也是动态语言重要组成部分。译文有关柯里化使用方法及应用场景讲解的很清楚。"Swift的各种特性已经被很多人研究过，但有一个特性只有较少人提及，那就是函数柯里化。本文将试图覆盖柯里化函数的基本知识点和一些可能的使用场景，希望对你了解Swift中的函数柯里化有所帮助。" 译者：@pockry 原文：[《Swift Function Currying》](#)
- [使用 Swift 构建自定义（且“可设计”的）控件](#): "使用 Swift 构建自定义（且“可设计”的）控件。喜欢别人发布的自定义控件？那设计并实现自定义控件需要几步呢？只要三步。读了本教程，你就可以实现任何你想要的自定义控件。" 译者：@nixzhu，原文[How to building a custom and designabl control in swift](#)
- [UIPrintInteractionController](#): 文章介绍了如何格式化打印内容及相关打印定制技巧或控制技术。或许，在如今移动及数字化时代，打印已经被许多人所忽视。无论如何，对于传统行业，需求仍然是切实存在的，因此，这是一篇难得好文章。
- [在iOS 8 SDK中使用Touch ID API — Part I, Part II](#): 译文详述了如何在一款笔记编辑应用程序中集成iOS 8 SDK中Touch ID API，包括上下两个部分，在开发过程中使用了Swift语言。
- [Core Data 版本迁移经验总结](#): 版本升级是必备经验。文章或许还不够全面，但都是作者点滴经验，非常难得。对于进阶开发的同学文档《[Core Data Model Versioning and Data Migration Programming Guide](#)》是必读的，参考文章《[自定义 Core Data 迁移](#)》也值得收录。作者：@Chun_iOS

4. 开发技巧

- [Problems With Building a Today Extension in iOS 8](#): 构建Today扩展时可能碰到的一些问题。
- [开发Yosemite几个小技巧](#): "Apple 终于发布了 Xcode 6.1，带来了 Swift for OSX 等多

个更新，这几天我简单研究了下在 Yosemite 下实现一些新的小需求的方法，这里使用 Swift 语言描述总结一下。" 作者: @diumoo

- [详解UICoordinateSpace和UIScreen在iOS 8上的坐标问题](#): 今天在整理AutoLayout API时，发现了一个关于UIScreen 坐标的坑。作者: @pockry
- [我在开发第一个Swift App过程中学到的四件事](#): "本文翻译自raywenderlich.com，作者Greg Heo，是Razeware（Ray创办的公司）的员工，这是他为讲授iOS 8 App Extensions视频教程而实际使用Swift开发了一款App的经验，来看看他的心得体会。" 译者: @pockry
- [LTBouncyPlaceholder代码解读](#): "我希望你已经下载了 LTBouncyPlaceholder 的 Demo，用 Xcode 6 打开并编译、运行，然后在界面中显示的几个 UITextField 里输入一些文字来体验这个扩展。看到 Placeholder 的动画了吗？" By nixzhu
- [Swift中下划线的妙用](#): "在Swift中，下划线有很多妙用，这里将已经看到的妙用进行总结，希望可以帮助更多学习Swift的朋友。..." By @twlkyao
- [WWDC14 Session 402 学习笔记](#) By @糖炒小虾_txx
- [Swift与Objective-C混编高级教程之混编框架的创建和调用](#)
- [SwiftSingleton](#): 三种方法介绍了Swift中Singleton的写法
- [Don't Miss These Navigation Bar Interactions in iOS8](#): iOS8新方法，像Safari那样滑动时自动隐藏navigation。
- [Beginning iOS 8 Programming with Swift 读书笔记](#): 点滴都是经验，也是日日夜夜与汗水的交织。好笔记！作者: @荧星诉语
- [为大屏手机而设计](#): 用了一个简单例子讲解了大屏手机基本的可操作性。总之，无论屏幕有多大，轻易放弃单手操作是不明智的。
- [ochococo/Design-Patterns-In-Swift](#): 列举了很多用Swift实现的程序设计模式。

5. 相关文档

- WWDC 2014
 - [WWDC 2014发布会\(带中文字幕\)](#): 有关Swift语言演讲部分从103:54开始，首先苹果iOS/OS X及公共平台负责人克雷格·费德里吉(Craig Federighi)针对Xcode及Objective C的简要回顾，克里斯·拉特纳(Chris Lattner)上台做Swift语言演示从107:15开始。
 - [WWDC 2014 Videos](#): 需要苹果开发者帐号才能观看。
 - [WWDC 2014 PDFs](#): 107个PDF压缩包 By @清醒疯子
- [Chris Lattner \(译文: Swift 编程语言首席架构师\)](#): "Chris Lattner（1978年出生）是 LLVM 项目的主要发起人与作者之一，Clang 编译器的作者。 he 现在是苹果公司『开发者工具』部门的主管，领导 Xcode、Instruments 和 编译器团队，从 2010 年 7 月开始主导" By 伯乐在线
- [关于Swift, 开发者最需要了解的7个方面 \(英文原文\)](#): 简明扼要的阐述了Swift语言的几个重要特点

- [程序员眼中的苹果Swift语言：简单 易学 高效](#)：“以下是记者准备的七个问题，涉及Swift的优点、缺点。以及Objective-C的结局。索菲斯的答案中有些会有点骇人、令人震惊，或许还有几丝伤感。” 相关参考：[知乎《如何评价 Swift 语言？》](#)
- [行走于 Swift 的世界中](#)：总结了一下近一周以来的一些觉得这个语言里有意思的地方。By [@onevcat](#)
- [和Swift亲密接触的这半个月](#)：虽然学习能力及基础不同，但这样的学习体会及认识，至少能带给初学爱好者以启发！[转发] "未来 Swift 会发展的怎么样我无法预言，我是肯定会怀念这段时间和世界人民一起为 Swift 疯狂的日子。" By [@图拉鼎](#)
- [我不懂 Swift 语言](#)：能听到不同的声音是非常有益的，何况作者有些观点很有建设性。比如："Swift 仍旧在改变，它是 beta 版本，所以它肯定是能够改变的。要知道如果你在它是 beta 版的时候还不提出问题，那么如果你可能会需要很久时间才能让它进行改进。"
- [从Objective-C到Swift](#)："Swift背后的概念大多与Objective-C类似，但更为简洁、自然，也吸收了很多其他语言的语法。本文将对Swift的语法、特点及改进进行全面介绍。" By [黄兢成](#)
- [Swift の 函数式编程](#)："Swift 相比原先的 Objective-C 最重要的优点之一，就是对函数式编程提供了更好的支持。Swift 提供了更多的语法糖和一些新特性来增强函数式编程的能力，本文就在这方面进行一些讨论。" By [@diumoo](#)
- [iOS界面开发的大一统](#)："简单介绍了下 Size Classes 和 UIPresentationController 的内容。" By [@onevcat](#)
- [MVVM 介绍](#)（译者：[@nixzhu](#)，原文：[Introduction to MVVM](#)）：MVVM相较于MVC未必更轻量化，不过它达成了View Controller的轻量化。界面层逻辑(View Model)的抽象，不仅有利于理清View/Controller逻辑的粘连不清，对于单元测试也更容易了。
- [Top 5 iOS 7 Animations](#)：iOS 7 排名前5的动画效果预览。谁能解读一下到这些动画效果对应的程序库、或找到对应或相近开源代码分享吗？（Swift版本更佳，OBJC也可以）
 - [IFTTT/JazzHands](#)：Flickr开始界面动画类似效果，OBJC版本实现程序库。
- [An analysis of sorts between Objective-C and Swift](#)："Swift 到底比 Objective-C 快多少？结论是：6倍以上（仅排序测试）。[这里](#)有为什么快的原因" By [@图拉鼎](#)
- [iPhone 6 屏幕揭秘](#)(译者：[@小雪-Joanna](#))：关于iPhone6屏幕渲染的归纳。建议新手脑补一下这部分知识。原文：[iPhone 6 Screens Demystified](#)
- [10 Swift iOS open-source projects you cannot ignore](#)：10大不能错过的#Swift#开源项目。P.S. 本指南都有收录。
- iPhone6 & 6 Plus 尺寸适配说明
 - [iPhone6 & 6 Plus 视觉设计适配说明](#) 作者：[@英小俊](#)
 - [iPhone6和iPhone6 plus的iOS8设计尺寸参考指南](#) 作者：[@芒果道长](#)
- [iOS 5+ 设备一览](#)：作者的意图是通过这样一张直观的图，指导选择相对合理的设备组合进行开发和测试。[PDF版本](#)
- [超越Objective-C, 超越Swift](#)：很有趣的作者，除了对Swift有比较专业地概括式理

解，还从应用的纬度对新语言提出了自己的要求。比如：自动持久化，自动RESTful API，有表现力的触发器和响应，所见即所得的UI编辑等。译者：[@pockry](#) 原文：[Beyond Objective-C, Beyond Swift](#)

- [Some ideas for projects in Swift](#): “正在造轮子的不妨看看~” By [Chris Eidhof](#)(creator of objc.io)
- [iOS设计指南](#)(译者: shengbanx): 非官方文档，不过原文《[The iOS Design Guidelines](#)》遵循并基于《iOS Human Interface Guidelines》进行解读介绍。非常值得收藏。原作者Ivo Mynttinen，是一位来自德国的年轻UI设计师。其个人博客网站也值得同学们收录：[ivomynttinen.com](#)

Swift项目

1. 实用类库

“

以下实用类库做为快速参考，更多实用类库需求，也可以去这里([Swift toolbox is a community-supported catalog of iOS and OSX libraries](#)) (By [Adam Leonard](#) from [swiftcast.tv](#)), 这个网站收藏了GitHub开源社区中优质Swift资源库，并提供了分类及查询服务。

- [Mattt Thompson](#): 著名开源作者，开发了知名的[AFNetworking](#)网络库。也是知名开发博客[NSHipster](#)网站主。

类库	相关文章	备注
Alamofire/Alamofire	Alamofire	网络请求和相关的封装
mattt/Euler	Swift Operators	这样写是否很数学、很逻辑？ $\sqrt[3]{27} \div 3 \pm 5$, $\sum[3,1,2]$, $\sim 0 \vee 1$
mattt/Surge		基于苹果Accelerate高性能计算框架库，计算效率提升惊人
mattt/Literally	Swift Literal Convertibles	常用数据类型的使用及转换工具库

- 工具类
 - [Dollar](#): 尤其在Array和Dictionary的支持上（实现类库仅一个Dollar.swift文件）。用过Javascript版的Lo-Dash或underscore就知道其实用程度。它是一套函数化编程的工具库。另一个类似的更完整的版本是[ExSwift](#)，它的实现基于对

象化扩展的。

- [ExSwift](#): 实用类扩展库。另一个Lo-Dash或underscore的Swift版本实现，相对于Dollar.swift，此版本有更完整的API实现。包括了对Array, Int, String, Float, Range, Dictionary, NSArray的扩展及其它实用方法。完全遵循面向对象的扩展原则。
- [lingoer/SwiftyJSON](#): 这是解析JSON字符串封装类。实现功能与Javascript中的JSON.parse相近，使用方便。By [@lingoer](#)
- [SwiftyJSON/Alamofire-SwiftyJSON](#)：简单地整合Alamofire和SwiftyJSON后，远程JSON使用方便了。
- [typelift/Basis](#)：这个实用类库支持范围很宽泛，封装主要以简化及方便调用系统API为主，使程序代码看上去更优雅。遗憾地是，目前作者并没有提供API文档或示例代码，对于初学者使用会额外带来一点点学习成本。
- [owensd/json-swift](#): 功能与Swifty几乎一致，使用上要更方便。
- [gfx/Swift-JsonSerializer](#)：JSON解析又多了一种选择。
- [Ahmed-Ali/JSONExport](#)：一款实用工具，可以将JSON转换并输出为Swift语言类模型（也支持Android上的Java），支持简单的定制。
- [hubertr/Swell](#)：同时支持Swift和ObjC的Log实用类。喜欢Log4j风格的日志管理类的同学可以收入。
- [RSBarcodes_Swift](#): "RSBarcodes allows you to read 1D and 2D barcodes using metadata scanning capabilities introduced with iOS 7 and generate the same set of barcode images for displaying and sharing." By [@yeahdongcn](#)
Swift语言版二维码识别及生成类库。
- [SwiftP2P/SwiftSSL](#)：常用 Digest 和 HMAC 的封装。如此封装后，使用加密算法确实很方便。很不错的一个国产"轮子"! By [@你全家都快到碗里来](#)
- [Hearst-DD/ObjectMapper](#)：对象与JSON互转实用类库。需要的同学可以收一下。
- [DaveWoodCom/XCGLogger](#)：这是一款功能比较健全日志输出框架（Log4Swift）。之前推荐过一款类似的 hubertr/Swell，但它的功能明显没有这款强大。
- [isair/JSONHelper](#)：又一款JSON转对象类库，算上，lingoer/SwiftyJSON，owensd/json-swift，gfx/Swift-JsonSerializer，已经有四款了。同学们根据喜好及需求，择优使用吧。
- [kylef/Stencil](#)：通过在模板中支持if, for, ifnot 等语法标签，实现更强大、更灵活的数据及格式输出。
- [kylef/PathKit](#)：简单、实用的文件路径操作类，常用功能一个也不少。
- [thoughtbot/Argo](#)：一个带有实战说明的JSON解析方案及类库。
- [typelift/swiftz](#)：一个帮助实现函数式编程的类库，定义了一系列相应的数据结构。
- [dongri/OAuthSwift](#)：Swift版OAuth(1/2)授权类库。对于作者提供的API，看着

舒服，用着方便。

- 图片

- [kaishin/ImageScout](#): 最小网络代价获得图片大小（宽和高）及类型，它对于很多场合，比如：布局和排版都有积极作用。
- [kaishin/gifu](#): 高性能GIF显示类库Swift版本。

- 存储

- [SQLiteDB in Swift](#): "This is a basic SQLite wrapper for Swift. It is very simple at the moment and does not provide any advanced functionality. Additionally, it's not pure Swift at the moment due to some difficulties in making all of the necessary sqlite C API calls from Swift."
- [Realm - a mobile database](#): Realm主打移动数据库。除了更轻量，甚至还可以应用到可穿戴。低耦、面向对象的设计风格也是非常亮丽的。
- [nerdyc/Squeal](#): 一套非常完善的SQLite数据库访问API，无论建库，建表，CRUD等常用数据库操作命令都进行了很好的封装，另外，版本管理、事务管理、并发管理、命令执行及Prepare Statement等都提供相应支持。它是一套不可多得的SQLite管理访问库。
- [stephencelis/SQLite.swift](#): 简单、轻量，或是使用上最SQL的SQLite封装库。
- [BjornRuud/Swiftache](#): Mustache的Swift语言实现版本。
- [Haneke/HanekeSwift](#): 貌似这个轻量地带缓存图片组件还不错。主要功能包括图片二级缓存、异步加载、后台执行、自动缩放等。感兴趣的同学可以试用一下。
- [SugarRecord/SugarRecord](#): 相对于SQL，CoreData可以更方便、高效存储数据，而SugarRecord类库可以让你更方便的使用CoreData，同时作者已经封装好iCloud，使你的应用开发更如虎添翼。
- [michaelarmstrong/SuperRecord](#): Swift版CoreData框架扩展类库。相对于同类型CoreData框架库SugarRecord/SugarRecord <http://t.cn/RhYLS4n>，SuperRecord要简单、轻量得多。很棒的快速开发类库。
- [aschuch/AwesomeCache](#): Swift语言写的高效能缓存对象存储及管理，定义及使用简单、易于理解。支持为每个Cache对象设定缓存有效期。
- [daltoniam/Skeets](#): 网络图片的获取、缓存及显示类库，支持缓存的清理及时效性管理。

- 远程访问

- [Minimalistic Swift HTTP request agent for iOS and OS X](#): 一个简单、小巧、实用的HTTP请求Swift语言实现类（仅一个Agent.swift类）
- [lingoer/GRequest](#): "@李洁信：个人认为AFNetworking这种较重的第三方网络库将逐渐淡出舞台，而像楼主这种基于NSURLSession并充分利用Swift特性写出来的小而美的库会受到更多人的青睐！" 作者解读 [《GRequest for HTTP Request》](#)。

- [daltoniam/SwiftHTTP](#): Thin wrapper around NSURLSession in swift. Simplifies HTTP requests.感兴趣的可以看看代码，比较与lingoer/GRequest差别。
- [AshFurrow/Moya](#): 如果你需要[如此功能](#)的Swift网络API，可以考虑使用它。它基于Alamofire, swfitz等优秀开源类实现。
- [aleclarson/emitter-kit](#): 用EmitterKit代替NSNotificationCenter。这个库貌似短小、精悍、实用的样子。更重要是语法精练。
- [daltoniam/starscream](#): Swift版本WebSocket客户端类库，支持iOS/OS X。使用方便，跨平台开发不可或缺的"轮子"。
- [shuoli84/SocketIO Cocoa](#): 又一款WebSocket实用客户端类库，作者提供了详细的测试用例可以做为[示例代码](#)使用。
- 社交网络
 - [lingoer/SwiftWeiboKit](#): "封装了整个OAuth2.0的授权流程,并提供了几个简便易用的请求方法"
 - [A Twitter framework for iOS & OS X written in Swift](#): 一套很完整的Twitter访问框架类库，支持iOS/OS X
- 框架
 - [robb/Cartography](#): Set up your Auto Layout constraints declaratively. 这是有关自动布局约束一个实用的Swift项目，代码看似简单清晰，不过由于设计巧妙。[代码解读](#) By [@kaiyuz](#)
 - [railsware/Sleipnir](#): 一个基于Swift的行为驱动开发框架(BDD-style framework)。API安装，示例及说明相对比较齐全。
 - [Quick/Quick](#): 另一款基于Swift的行为驱动开发框架。
 - [inamiy/SwiftTask](#): 一个很标准的任务及其生命周期管理类库。作者还附上了一个基于Alamofire库完成的网络文件下载的任务管理示例。
 - [inamiy/SwiftState](#): Swift版本State Machine，这是SwiftTask的姐妹篇。开发过Workflow类应用的同学有没有很亲切。
 - [colemancda/NetworkObjects](#): 基于Swift的轻量版HttpServer框架，可以做为iOS/OS X分布式对象的替代。可惜缺少演示或示例代码。
 - [Masonry/Snappy](#): 一个基于AutoLayout封装，语法简洁的轻量级布局框架。当然，用代码还是storyboard，全凭开发者喜好。若偏好可视化自动布局，@DevTalking 的两篇译文值得参考 [《Swift自适应布局教程》](#)和 [《为iPhone6设计自适应布局》](#)。
 - [jpsim/PeerKit](#): 基于事件驱动的多端对等连接开源框架。作者提供了该组件使用案例[jpsim/CardsAgainst](#)，具体引用代码详见[这里](#)。
 - [Quick/Nimble](#): 功能更强于Xcode自带测试框架（XCTest）的测试类库。
 - [nghialv/MaterialKit](#): Swift版Google Material Design组件库。虽然个人并不推荐在iOS平台上使用非iOS风格界面，不过，对于Android强迫症“患者”，多一

种选择也是友好的。顺带附上Google提供的开源[Material Design图标集](#)，可以一起收入。

- UI组件

- [jcavar/refresher](#): 一个常用的下拉即刷新列表工具类，提供开放接口定制刷新动态变换效果。
- [ariok/BWWalkthrough](#): 让你的页面切换动起来，示例效果杠杠的。作者还提供了比较完整的开发文档及示例解说。
- [vikmeup/SCLAlertView-Swift](#): 动画效果弹出框封装库（管理于CocoaPods），使用也足够方便。试着运行了一下，效果还不错。
- [varshylmobile/MapManager](#): 地图管理封装库（默认支持Google和Apple地图服务）。
- [varshylmobile/LocationManage](#): 位置管理封装库（默认支持Google和Apple地图服务）。
- [ortuman/SwiftForms](#): 这个表单递交库简单实用，支持主要数据类型及定制。快速开发利器。
- [kentya6/KYCircularProgress](#): 这个进程条类库实现及使用都很简单，支持进路路径的定制，灵活方便。
- [coolbeet/CBStoreHouseRefreshControl](#): 这个下拉刷新库，提供完全可定制进度变换效果，且极具艺术感。建议有能力的同学把Objective-C版本改写为Swift版本。
- [kevinzhov/PNChart-Swift](#): PNChart的Swift版实现。PNChart <http://t.cn/zRWXJCZ> 是饼图、折线图、柱状图等常用可视化图表工具类库。开发者: [@周楷雯Kevin](#)
- [codestergit/SweetAlert-iOS](#): 该弹窗控件设计简单直接，功能完整，动画效果恰如其分。实在不可多得，必须收录。
- [Ramotion/animated-tab-bar](#): 一款灵动的动画标签栏类库。可在storyboard中直接使用，非常方便。Demo效果非常漂亮。连续三天GitHub分类排名第一，且迅速过600星。它的代码并非复杂或特别，相信大家中意它的还是其设计感。

2. 示例项目

- [官方示例](#): "比起GitHub上的开源项目来说，官方的代码我觉得是更有参考价值的，比如Session 406的代码，Lister，就用一个Swift实现的包含OSX和iOS的" By [@晨钰Lancy](#)
- [VirtualGS教程示例](#): 以下示例程序来源于林泰前老师[微博](#)或[博客](#)发布，为方便大家学习Swift编程，有幸获得林老师的准许在这里发布。
- [Easy-Cal-Swift](#): "实在忍不了Swift的数字计算时候的好麻烦的强制转换了，重载了一

下加减乘除之类的操作符，这样就不用显式地转换类型了...（对于像我这样现在连补全都没有的孩子来说，能省好多时间啊- -）" By [@onevcats](#)

- [An app to list the available fonts on iOS](#): 用Swift语言调用UIKit，列出设备内所有字体名称的小程序。
- [Swift版的知乎日报](#): 学习一门新语言，光看是没有用的，想要快速的掌握它，就得投入到真实项目的开发中，仿照@YANGReal 的糗事百科，做了一个Swift版的知乎日报
- [Swift RSS Sample](#): 用Swift语言开发的RSS阅读器
- [Swift版PM2.5的例子](#): 一个很好的Swift与Objective C协同工作GitHub开源项目。用到的开源类库有[TFHpple](#): 以XPath方式解析HTML, [SCLAlertView](#): 使用Swift写的AlertView SVProgressHUD 进度条 By [@荧星诉语](#)
- [lexrus/LTMorphingLabel](#): 实现文字飘入飘出的效果。效果非常赞！
- [android1989/CharacterText](#): 相比[lexrus/LTMorphingLabel](#) 的各种酷炫效果，这个版本比较简单实用。
- [practicalswift/Pythonic.swift](#): 用Swift语言实现Python标准库的一部分。然后，用Swift写一段Python风格的程序，这是Python程序员想要的吗？相信这不过是一个实验，以此说明Swift语言的多变性、动态性的能力。
- [AshRobinson/GoogleWearAlert](#): Swift语言实现模拟Google Wear风格Alert窗口。
- [rafaelconde/ios8-ui-kit](#): IOS 8 UI KIT + FOR SKETCH — 最新IOS8免费设计资源
- [gemtot/iBeacon](#): Swift版iBeacon简单项目（支持最新Beta 6编译）。感兴趣的同学可以学习一下。有关于Passbook应用及Passkit框架编程知识这里有一篇网友较早发布的[《iOS 6 - PassKit编程指南》](#)
- [evnaz/ENSwiftSideMenu](#): 一个简单的Slide侧拉菜单实现。使用很方便：sideMenu = SideMenu(sourceView: self.view, menuData: ["UIDynamics", "UIGestures", "UIBlurEffect"])
- [iluuu1994/Pathfinder](#): 一个有趣的算法类项目。虽然目前只是个演示项目，不过，作者有计划加入更多算法优化程序，同时，也有计划支持3D地图。
- [KhaosT/HomeKit-Demo](#): HomeKit演示项目，可以与HomeKit模拟器协同工作。由此进一步思考：HomeKit配合iBeacon、蓝牙，甚至更具DIY潜力的树莓派(Raspberry Pi)组成未来家居智控中心，这个方向上的开发潜力巨大。感兴趣的同学可以关注一下。
- [mathewsanders/Animated-Transitions-Swift-Tutorial](#): 结合Xcode开发步骤介绍如何开发动画过渡Prototyping Animatted Transition in Swift(Part I)] (<http://mathewsanders.com/custom-menu-transitions-in-swift/>)
- [mathewsanders/Custom-Menu-Transition-Swift-Tutorial](#): 结合Xcode开发步骤介绍如何开发动画过渡Transition in Swift(Part II)](<http://mathewsanders.com/custom-menu-transitions-in-swift/>)
- [vandadnp/iOS-8-Swift-Programming-Cookbook](#): 来自《O'Reilly's iOS 8 Swift Programming Cookbook》的配套示例。书买不买另说，如此即时、完整、丰富的新书示例项目真不多见。悟性好的同学，开发时参考一下示例是不是就不用买书了？

- [ide/UIVisualEffects](#): Swift实现模糊化等视觉效果示例代码。代码详见[这里](#)
- [sxyx2008/FoodPin](#): 如此完整的Swift示例项目，可以帮助学习到不少Swift编程技巧。"用Swift写的一个功能比较完善的App，参照Beginning iOS 8 Programming with Swift一书。功能比较完善。AutoLayout、Core Animation、Core Data、国际化等。" By [@荧星诉语](#)

3. 完整项目

- [FlappySwift](#): 用Swift语言实现的 FlappyBird
- [天气预报iOS项目](#): 新界面还不错，简单，还带点卡通的味道
- [tnantoka/edhita](#): edhita是一款用Swift重写并完全开源的文本编辑器。AppStore上已经有更新版下载。试用后感觉还不错。它甚至支持Markdown, HTML等文件编辑后的预览显示。
- [jurre/TravisToday](#): 一个功能相对完整的服务于Travis CI系统的OS X应用扩展Today项目案例。
- [stakes/Frameless](#): 一款基于WKWebView的最小化全屏浏览器，主要功能是方便开发者采用HTML5做原型设计或App发布。对于测试工程师或喜欢极简、全屏浏览用户，可以直接在[AppStore下载](#)使用。

Swift网站

- [ksm/SwiftInFlux](#): 作者(Karol Mazur)将Apple Developer Forums上有关Swift特性、缺陷及变更讨论分类汇总并更新到Github，具有很好的可读性。从中可以一窥Swift缺陷及未来潜在地变化。最关键地是有Chris Lattner及核心团队答疑解惑。
- [raywenderlich.com](#)([中文版](#)): 由Ray Wenderlich创建，专注于开发高质量编程指南（近期优质Swift文章及视频教程不断），著名的iOS/OS X博客及开发教程网站，非常适合新手学习。近期第一时间出了[三本Swift新书](#)。
- [Natasha The Robot](#): 时髦码农不容错过的Swift开发实战教程类网站。为了简化学习复杂性，每一篇文章涵盖技术点单一又独立，配以开发步骤、运行结果以及代码等标准方式教授。难能可贵的是，它往往出品"追剧式"（最新开发特性或API跟进式）教程，实在是开发者必藏精品网站。
- [NShipster](#) ([中译版](#)): 著名开源作者[Matt Thompson](#)创建的开发技术博客网站，他开发了[AFNetworking](#)网络库，也是非常多产的开源作者。更多了解参考：[《COCOA 潮人 MATT THOMPSON》](#) By [@程序员付恒](#)
- [jamesonquave.com](#): 移动开发者，优秀个人博客（近期文章同样关注于Swift语言，写得很优质）。同时他将于8/30发布一本新书[《Developing iOS 8 Apps in Swift》](#) (Learn To Make Real World iOS 8 Apps)及视频教程。
- [objc.io](#)([中译版](#) By [@onevcats](#) [及其朋友们](#)): "关于 Objective-C 最佳实践和先进技术的

期刊。由 Chris Eidhof, Daniel Eggert 和 Florian Kugler 成立于柏林。我们成立 objc.io 的目的是针对深入的、跟所有 iOS 和 OS X 开发者相关的技术话题创建一个正式的平台。”

- [iOS Dev Weekly](#): 收录一周以来iOS开发资讯链接，并于周五发布。由[Dave Verwer](#)创办，他是一位iPhone和iPad开发者以及培训师。
- [Appcoda.com](#): 质量很高的一个iOS开发教程站，其中[iOS Programming Course](#)这个专题很适合刚接触iOS开发的新手学习。
- [中文 iOS/Mac 开发博客列表](#): By [@唐巧 body](#)
- [devtalking.com](#): 高产的中译博客。翻译了官方博客Swift Blog - Apple Developer, 《App Extension Programming Guide》。参与翻译了《Swift Programming Language》等。
- 资源合集：以下是其它开发者社区或Swift爱好者整理的有关Swift语言学习的资源列表。
 - [码农周刊《Swift 特刊》](#)
 - [CocoaChina《Swift新手入门汇集帖》](#)
 - [CSDN CODE《Swift编程语言资料大合集》](#)
 - [InfoQ《学习苹果Swift语言的一些在线资源\(英文\)》](#)
 - [刘兰涛《Swift学习资源》](#) By [@懒桃儿吃桃儿](#)
 - [learnsswift.tips](#): 国外主流Swift学习资源集合。
 - [Awesome iOS](#): 一个iOS的各类优秀的开源项目集合。真不错！可惜Swift开源项目资源不足。
 - [iOS Developer Tips](#): 还是有关iOS的开发资源及文章合集。
 - [Wolg/awesome-swift](#): 一位俄罗斯朋友分类整理的Swift资源列表（有持续更新）。

相关工具

1. 开发工具

- [Xcode 6 beta 下载](#): 苹果应用集成开发环境。支持C/C++, Objective C, Swift等。不用购买开发者计划，直接下载。
- [Textmate](#): Mac OS X上一个可高度自定义的编辑器，尤其在我想做出一个快速改变但又不想等待Xcode加载的时候。该工具目前已经[开源](#)
- [Mou](#): OS X上一款Markdown的编辑器。非常适用于编写自述文件、变更日志以及其他方面的内容。作者：[罗晨](#)
- [Sublime Text \(\\$\)](#): Mac OS X上另一款非常受欢迎的轻量级，可高度自定义的编辑器。
- [RunSwift](#): 正在犹豫是否入手苹果电脑开始一段Swift编程旅程的同学们，或仅仅为了

试验一段简单Swift代码又懒得打开Xcode，可以试试这款Web版Swift编译环境RunSwift。

2. 代码管理

- [GitHub](#)：声望日盛的资源分享之地。
- [GitHub for Mac](#)：一个设计的非常美观的git客户端，不能取代你从命令行获得的所有功能，但使用起来非常简单。
- [Git](#)：分布式版本控制系统和源码管理系统，其优点是：快和简单易用。对于新手来说，可在此查看免费电子书籍。

3. Xcode插件

- [CocoaPods](#)：第三方库的管理利器，允许你简单地把第三方库整合进自己的应用中。对我个人来说，我基本上每个项目都使用CocoaPods。
- [CocoaPods Xcode Plugin](#)：一款Xcode插件，允许你直接从Xcode管理CocoaPod依赖。
- [onevc/VVDocumenter-Xcode](#)：快捷注释Xcode插件。By [@onevc](#)

4. 管理工具

- [HomeBrew](#)：OS X上非常出色的包管理工具。
- [Transmit \(\\$\)](#)：一个Mac OS X 上FTP客户端，有着非常漂亮的用户界面和有用的功能。

5. 调试工具

- [matt/fuckingclangwarnings.com](#)：警告与语义对照表。以后再也不用为Xcode各种警告纠结啦！By [@foogry](#)

6. 参考文章

- [iOS开发工具](#)："这是我们多篇iOS开发工具系列篇中的一篇，此前的文章比如：那些不能错过的Xcode插件，iOS开发者有价值的工具集，iOS/OS X开发：各种工具快到碗里来！，App原型设计工具使用心得（上）& App原型设计工具使用心得（下），你用哪种工具进行iOS app自动化功能测试？，iOS 开发者必知的 75 个工具" By @CocoaChina
-

开放平台

开放平台相对于Swift语言更具战略意义，这是开发者不得不面对的挑战。也是苹果新一代创新应用的催化剂。通过Extension达成应用之间的协同及通讯，这是对生态内应用开放的基础。让我更期待的是，Extension在Safari Action上实现及支持，这是实现平台开放及跨平台应用最简单直接的方案。

1. 文档

- [谈谈 iOS 8 和 OS X 10.10 的 Extension](#): "我个人认为这是 iOS 和 OS X 发展至今非常具有里程碑意义的一处改进，甚至比 UI 上的改变重要的多。我想简单地谈一下为何 Extensions 这么重要。" By [@图拉鼎](#)
- [App Extensions学习笔记](#): "系统中支持extension的区域，extension的类别也是据此区分的，iOS上共有Today,Share,Action,Photo,Editing,Storage Provider,Custom keyboard几种，其中Today中的extension又被称为widget" By [@foogry](#)
- [详解 iOS 8 的动作扩展](#): "动作扩展的出现，意味着用户能在应用程序间的切换上花更少的时间和精力，这是相当大的进步。" 译文作者: [@米斯特苹果](#), 原文: [《Action extensions in iOS 8: Explained》](#)
- [苹果iBeacon让智能家居走进现实](#): "iBeacon最初发布的时候是一个协议，苹果希望利用这一协议取代NFC技术。iBeacon技术则可以利用支持该技术的设备创建一个信号区域，相当于实现了地理围栏的功能，当其他支持iBeacon技术的设备如手机进入这一区域时，对应的应用程序就会自动连接这一区域的信号网络，或者对用户进行提示"
- [Android L/iOS8/WP8.1到底谁抄了谁?](#): "我们发现在三场发布会上都听到了观众这样的声音：“无耻抄袭！抄了谁谁谁的！！”，这种事情似乎说也说不清，所以我们决定把这三个新系统放到一起来看看，然后再下结论。"
- [Google开源字体Noto Sans CJK简介](#)(By [@洋气书生](#)): 这篇Noto Sans CJK(Chinese, Japan, Korean)字体普及文章简单、直接、专业，且易于理解。作者友善地提供了一份[本地下载](#)，赞一个！P.S. 毫无疑问，新版Android上会很快支持，iOS/Mac/Windows上也会尽快缺省支持吗？
- [iOS 8 与 OS X 10.10 间的自由联动：Handoff 使用详解](#): 有关如何开启和使用Handoff，文章做了比较详细的介绍，并以Safari为例进行了使用说明。作者([@iTumbledSea](#))撰写此文的英文参考 [《How to use Handoff with iOS 8 and Yosemite》](#)
- [Web Apps](#): 这篇新鲜文章较全面论述了Native应用和Web应用。有关Native应用和Web应用，最近争论渐少，原因应该有两点：1. 无论Apple还是Google，Native应用才是其生态的基础，过于激进推动，将损害其固有商业利益。而微软的话语权严重受阻于移动系统的发展；2. 基于Web View的混合应用基本满足应用跨平台性。
- [Back to Mac - XPC by objc.io](#): XPC 是 OS X 下的一种 IPC (进程间通信) 技术，它实

现了权限隔离, 使得 App Sandbox 更加完备。

2. 示例项目

- [ViewSource\(Swift+Objective-C\)](#): 通过iOS 8 Extension实现让Web工程师喜欢的"显示网页源代码"。
- [indragiek/Unzip](#): 浏览ZIP文件的iOS 8 Action扩展。

3. 实用资源

- [google/material-design-icons](#) : 对于喜欢 Material Design风格的同学, 这是难得好资源。Google提供了极为完整的各种图标 (包含iOS各种精度及SVG) 设计资源。

媒体文章

- [WWDC 2014给开发者带来了什么?](#): "苹果向第三方开发者开放了大量的可调用特性和4000个新API——指纹识别、云存储、智能家居平台、相机控制、健康数据平台、3D图形API、对iOS 8可扩展程序的调用, 以及新的编程语言Swift。PingWest制作了一种信息图带你一览这些新特性"
- [编程语言进化链的顶端: 为什么说Swift正在颠覆整个互联网生态? \(36Kr\)](#): "Swift 代表的程序猿先进生产力的发展要求 (提高编程效率), 代表了计算机先进文化的发展方向 (语法简洁, 现代), 代表了广大人民的根本利益 (写起来爽, 学起来快)。"
- [苹果编程语言Swift解析: 将推动应用开发巨变\(CNET\)](#): "如果编程语言更加易学易用, 那么应用开发的门槛将会降低, 导致更多新手开发者参与这一行业。"
- [外媒评论: 苹果公司Swift语言将改变一切](#): 美国财经网站Motley Fool针对Swift比较中性的一篇评论文章, 因此目标读者是非技术人员, 可读性还不错。
- [苹果新贵 Swift 之前世今生\(池建强\)](#): 这篇文章故事性很强, 不过, 的确把前世今生、来龙去脉交待了一遍, 适合刚开始了解swift语言的程序员。
- [WWDC 2014: 给第三方开发者的情书\(Qdaily 李如一\)](#): "iOS 和 Mac 上的御用语言 Objective-C 可以追溯至 NeXT 时代, 换言之, 它已经有超过二十年的历史。Swift 作为苹果发明的编程语言, 也继承了苹果产品的传统"
- [苹果发布Xcode 6 Beta 3, Swift迎来重大更新!](#): "苹果在面向开发者推送iOS 8 Beta 3以及OS X Yosemite的第三个预览版的同时, 也发布了全新的Xcode 6 Beta 3, 并对Swift语言进行了大幅改进。新版Swift修正了许多开发者提出的请求, 尤其是对数组进行了重新设计。" 作者: [@唐门教主](#) 扩展阅读 [《Swift 在 Beta3 中的变化》](#)
- [《连线: 为什么苹果Swift语言将会迅速普及》](#): 为苹果硬件开发了15年软件产品的麦克·艾什 (Mike Ash) 相信, 苹果最终会将Swift开源, 而且他也相信该语言会在苹果的控制之外发展壮大——因为该语言的开发者拉特纳有着很深的开源情节。"有他在

掌舵，我觉得他会做出正确的选择”。原文：[Why Apple's Swift Language Will Instantly Remake Computer Programming](#)

- [Swift中文翻译组](#): 近30人9天协作完成翻译近670页的英文文档
 1. [协同写作的力量——中国开发者9天完成《Swift语言》中文版](#): "详细介绍了GitHub上开源翻译《Swift语言》这个开完项目，发起者是一个90后的大学生，整个翻译团队在9天内完成了近670页的Swift语言文档翻译工作。" By 36Kr
 2. [翻译暂时告一段落 写点感想吧](#): "现在翻译已经告一段落，感觉这段时间的效率真是高的可怕，也许是因为一种成就感，或许带了那么一点功利心（我想每个人或多或少的都会有一点吧）。" By @CoverXiT
 3. [Apple的Swift语言](#): "哥也无意中在Github上看到这个翻译项目，并有幸参与了翻译，算是亲身体验了一把多人协作的开源项目，感觉很赞也很有成就感....." By @TimothyYe
 4. [雨燕Swift](#): "...我做翻译这件事的目的其实挺自私的，没想改变世界，没想着跟世界同步，没想干什么轰轰烈烈的大事。... 我是Aminby，和大多数程序员一样默默无闻地用国内外先进的技术或解决方案为工作忙活着的普通人。" By @老白经aminby
- [蒂姆·库克的苹果](#): "当库克走向舞台左侧的暗处时，气氛一时变得神秘起来。这时苹果软件工程负责人克莱格·费德里西(Craig Federighi)快步走上台。他和库克擦肩而过，走到聚光灯下，向大家介绍这款新品。它不是一款新的消费产品，而是一套名为“开发工具包”的软件工具，可以帮助开发人员开发出更好的应用。这个世界上的其他人可能会对此打哈欠，但开发者们站起身，兴奋地叫嚷起来。"
- [苹果 A8X 芯片难逢敌手！英特尔、高通、三星纷纷落败](#): 回想当年乔布斯团队从iPad开始，毅然选择自己设计基于ARM架构的A4芯片，无论在iPhone/iPad系列产品快速演变还是商业上，都获得了罕见的收获。从CPU革命延伸到如今开发语言Swift，奠定了其庞大生态环境内基础软、硬件坚实的基础。未来的苹果显然更值得期待。
- [新视频透露了苹果都不知道的iPhone隐藏技能](#): "iPhone的性能和扩展是否已经超乎苹果的想象？近日国外网友 Ivo Leko 利用iPhone上多项传感器用于完成实时的磁铁位置追踪演示。"

Apple Watch指南

“

WatchKit确定了开发者在第一代Apple Watch能做什么，显然，这只是一个保守的开始。

1. 文档

- [WatchKit Programming Guide](#)

译文	译者	原文
概览 - 开始为Apple Watch进行开发	@星夜暮晨	Overview - Developing for Apple Watch
概览 - 配置Xcode项目	-	Overview - Configuring Your Xcode Project
概览 - Watch应用的体系结构	-	Overview - Watch App Architecture
概览 - 使用iOS技术	-	Leveraging iOS Technologies
Watch Apps - 文本和标签、图片	@sherlockdan	Watch Apps - Text and Labels, Images
Glances - Glance开发基础	@CocoaChina	Glances - Glance Essentials
Glances - 管理你的Glance界面	-	Glances - Managing Your Glance Interface
通知 - 有关通知的一些要点	@星夜暮晨	Notifications - Notification Essentials
通知 - 自定义通知界面	-	Notifications - Custom Notification Interfaces

- [Apple Watch Human Interface Guidelines](#)
 - [中译PDF版](#) By [@微博UDC-M](#)
 - [中译在线版](#) By [@CocoaChina](#)

译文	译者	原文
UI设计基础 - 为Apple Watch而设计	@CocoaChina	UI Design Basics - Designing for Apple Watch
UI设计基础 - App 剖析	-	UI Design Basics - App Anatomy
UI设计基础 - Glances	-	UI Design Basics - Glances
UI设计基础 - 通知	-	UI Design Basics - Notifications
UI设计基础 - 模态页面、布局	-	UI Design Basics - Modal Sheets, Layout
UI设计基础 - 动画、品牌化	-	UI Design Basics - Animations, Branding
UI设计基础 - 颜色和字体	-	UI Design Basics - Color and Typography
UI元素 - 标签、图像、Groups	-	UI Elements - Labels, Images, Groups
UI元素 - 列表、按钮、开关、滑杆	-	UI Elements - Tables, Buttons, Switches, Sliders
UI元素 - 地图、日期和计时器、菜单	-	UI Elements--Maps, Dates and Timers, Menus
图标与图片尺寸	-	Icon and Image Sizes

- 中译在线版 By [@优秀网页设计](#)

译文	译者	原文
UI设计基础 - Part 1, Part 2	@陈子木	UI Design Basics
UI元素设计	@阿布	UI Elements
图标与图片设计	@阿布	Icon and Image Design

3. Watch教程

- 快速入门
 - [Apple WatchKit 初探](#): "随着今天凌晨 Apple 发布了第一版的 Watch Kit 的 API, 对于开发者来说, 这款新设备的一些更详细的信息也算是逐渐浮出水

面。可以说第一版的 WatchKit 开放的功能总体还是令人满意的。Apple 在承诺逐渐开放的方向上继续前进。By [@onevcat](#)"

- [RayWenderlich – WatchKit初探](#) (译者: [@DevTalking](#), 原文: [WatchKit Initial Impressions](#)) : "Raywenderlich对WatchKit第一版的评价: 1.不简单的只是App extension;2.Watch负责视图, iPhone运行控制器和模型;3.新的layout方式;4.强大的快览和通知;5.动画不原生支持。By [@WatchKit开发](#)"。
- [Apple Watch 开发者套件 WatchKit 发布后, 你需要知道这些](#): "Apple 上线了供 Apple Watch 开发者使用的开发工具 WatchKit。从这款工具和相关文档中, The Verge 发现了一些重要细节。By [@36氪](#)"
- WatchKit快速入门两篇: 《[使用WatchKit前需要先了解的几件事](#)》和《[如何新建一个“Hello World” WatchKit应用](#)》。两篇文章一起组成WatchKit应用快速开发入门参考。
- 教程精选
 - [Apple Watch Tutorial - Learn How to Make a Simple Apple Watch App](#): 50分钟入门级开发视频。可惜Youku上的版本不够清晰。By [@36氪](#) 《[如何正确开发一款Apple Watch应用?](#)》
 - [用帧动画做AppleWatch小游戏](#): "WatchKit已经发布了2天, 先行者们已经进行了探索并发表了总览性的文章去帮助引导开发者。..." By [@WatchKit开发](#)

4. Watch项目

- [saigyoji205/Map_For_AppleWatch](#): 一个简单的WatchKit地图示例应用扩展。
- [frosty/Flipbook](#): "Flipbook是一个能录制iPhone上UIView动画生成帧动画序列并传输给AppleWatch以供直接播放的工具。By [@WatchKit开发](#)"

5. 媒体文章

- [Apple Watch应用开发正在紧锣密鼓地进行](#): "因此大家可以想象, 终于可以在最新 Apple Watch身上小试身手时, iOS开发者们该有多兴奋。周四, 苹果公司首席执行官蒂姆·库克 (Tim Cook) 宣布将推出WatchKit开发工具, 让开发者和移动行业创业者能够开始为这款最新设备开发新的应用程序。" By [福布斯](#)
- [Apple Watch需求被低估](#): "大摩分析师凯蒂·霍伯特(Katy Huberty)表示, 投资者还低估了苹果即将发布的智能手表AppleWatch的市场需求量。大摩分析师凯蒂·霍伯特 (Katy Huberty)表示, 投资者还低估了苹果即将发布的智能手表AppleWatch的市场需求量。"
- [苹果 Apple Watch 的杀手级特性](#): "方便简单: 目前我们已经了解了很多关于Apple Watch的功能以及特性, 但是了解了这些你是否会产生这么一种感觉: 我的生活需要 Apple Watch? 其实这种感觉并不强烈吧。" 文章短, 且论述平实, 适合大众阅读。

其它相关

- [VM10装Mac OS X 10.9.3及更新到Mac OS X 10.10](#): "嗯，我写的。屌丝学生党买不起Mac，只能在黑苹果和虚拟机上先折腾会了。" By [@Vong_HUST](#)
- [Origami](#): 快速原型动画开发工具。Origami由Facebook开发Quartz Composer工具库，它使原型开发更容易。参考文章：
 1. [次时代交互原型神器Origami档案](#): "随着iOS 7的推出，扁平化和极简主义设计风格在移动互联网领域流行起来，App动效越来越成为了决定App气质的重要因素，原型的动态保真度似乎成为了阻碍设计师发挥想象力的一道门槛。传统的以点按为主的App设计，逐渐演变成为大量手势交互，这使得Axure类工具表现手势交互显得心有余而力不足。基于QC的Origami应运而生。"
 2. [用 Quartz Composer 和 Origami 制作一个简单的按钮动画\(译文\)](#): "结识了 QC 和 Origami 之后，我就能用很短的时间制作出这个动画的原型。我爱上了 QC 和 Origami —— 我希望你在使用它们之后，也会爱上它们。同时，我十二分地感谢 Facebook 创造了 Origami，以及 Apple 创造了 Quartz Composer。" By [@nixzhu](#)
- [10 个 Safari for iOS 8 的实用小技巧](#): "继上回 7 个 Safari for Mac 实用技巧之后，这次来介绍 10 个 Safari for iOS 8 的小技巧，相信一定有你可能还不知道的使用新姿势。by [@子不语Rex](#) " 本身的细节，外加应用扩展，组合成更强大的Safari浏览器。
- [WWDC 2014 PDF 及session 视频下载脚本](#):一段下载WWDC2014 全部PDF 和 Session 的终端脚本。
- [cesards/AndroidElementals](#): "分类整理了GitHub上常用的Android Library和Open Project，整理的结构目录很清晰，找起来很方便。By [@Vanilla Alan](#)"