

一片枫叶

追求卓越，成功就会在不经意间追上你

跟随自己的节奏学习，思考，总结，找到自己，别人才会找到你

下一篇:基于NSBundle的工程结构

IOS：屏幕旋转学习笔记

屏幕旋转学习笔记

加速计是整个IOS屏幕旋转的基础，依赖加速计，设备才可以判断出当前的设备方向，IOS系统共定义了以下七种设备方向：

```
typedef NS_ENUM(NSInteger,
    UIDeviceOrientation) {
    UIDeviceOrientationUnknown,
    UIDeviceOrientationPortrait, //
    Device oriented vertically, home button on the
    bottom
    UIDeviceOrientationPortraitUpsideDown,
    // Device oriented vertically, home button on
    the top
    UIDeviceOrientationLandscapeLeft, //
    Device oriented horizontally, home button on
    the right
    UIDeviceOrientationLandscapeRight, //
    Device oriented horizontally, home button on
    the left
    UIDeviceOrientationFaceUp, //
    Device oriented flat, face up
    UIDeviceOrientationFaceDown //
    Device oriented flat, face down
};
```

以及如下四种界面方向：

```
typedef NS_ENUM(NSInteger,
    UIInterfaceOrientation) {
    UIInterfaceOrientationPortrait =
    UIDeviceOrientationPortrait,
    UIInterfaceOrientationPortraitUpsideDown =
    UIDeviceOrientationPortraitUpsideDown,
    UIInterfaceOrientationLandscapeLeft =
    UIDeviceOrientationLandscapeLeft,
    UIInterfaceOrientationLandscapeRight =
    UIDeviceOrientationLandscapeRight
};
```

导航

联系 管理

≤	2013年4月						≥
日	一	二	三	四	五	六	
31	1	2	3	4	5	6	
7	8	9	10	11	12	13	
14	15	16	17	18	19	20	
21	22	23	24	25	26	27	
28	29	30	1	2	3	4	
5	6	7	8	9	10	11	

统计

随笔 - 46 文章 - 3 评论 - 452

搜索

找找看

谷歌搜索

随笔分类(44)

- IOS(36)
- 数据结构与算法
- 杂谈(8)

随笔档案(46)

- 2015年1月 (1)
- 2014年11月 (1)
- 2014年10月 (2)
- 2014年8月 (1)
- 2014年6月 (1)
- 2014年5月 (2)
- 2013年11月 (1)
- 2013年8月 (1)
- 2013年7月 (1)
- 2013年6月 (2)
- 2013年5月 (3)
- 2013年4月 (4)
- 2013年3月 (1)
- 2013年2月 (1)
- 2012年12月 (3)
- 2012年11月 (2)
- 2012年10月 (1)
- 2012年8月 (1)
- 2012年7月 (2)
- 2012年6月 (3)
- 2012年5月 (4)

一、UIKit处理屏幕旋转的流程

当加速计检测到方向变化的时候，会发出 [UIDeviceOrientationDidChangeNotification](#) 通知，这样任何关心方向变化的view都可以通过注册该通知，在设备方向变化的时候做出相应的响应。上一篇博客中，我们已经提到了在屏幕旋转的时候，UIKit帮助我们做了很多事情，方便我们完成屏幕旋转。

UIKit的相应屏幕旋转的流程如下：

- 1、设备旋转的时候，UIKit接收到旋转事件。
- 2、UIKit通过AppDelegate通知当前程序的window。
- 3、Window会知会它的rootViewController，判断该view controller所支持的旋转方向，完成旋转。
- 4、如果存在弹出的view controller的话，系统则会根据弹出的view controller，来判断是否要进行旋转。

二、UIViewController实现屏幕旋转

在响应设备旋转时，我们可以通过UIViewController的方法实现更细粒度的控制，当view controller接收到window传来的方向变化的时候，流程如下：

- 1、首先判断当前viewController是否支持旋转到目标方向，如果支持的话进入流程2，否则此次旋转流程直接结束。
- 2、调用 [willRotateToInterfaceOrientation:duration:](#) 方法，通知view controller将要旋转到目标方向。如果该viewController是一个container view controller的话，它会继续调用其content view controller的该方法。这个时候我们也可以暂时将一些view隐藏掉，等旋转结束以后在现实出来。
- 3、window调整显示的view controller的bounds，由于view controller的bounds发生变化，将会触发 [viewWillLayoutSubviews](#) 方法。这个时候self.interfaceOrientation和statusBarOrientation方向还是原来的方向。
- 4、接着当前view controller的 [willAnimateRotationToInterfaceOrientation:duration:](#) 方法将会被调用。系统将会把该方法中执行的所有属性变化放到动画animation block中。
- 5、执行方向旋转的动画。
- 6、最后调用 [didRotateFromInterfaceOrientation:](#)

[2012年3月\(6\)](#)

[2011年12月\(2\)](#)

积分与排名

积分 - 122147

排名 - 1286

最新评论

[1. Re:iOS7: 如何获取不变的UDID](#)

大神，求加qq指导 啊！1772025238！谢谢

--Just_Lee

[2. Re:iOS7: 如何获取不变的UDID](#)

大哥，我iOS新手啊。第一个地方是plist文件中的accessGroup中的AppIdentifier。此处的AppIdentifier事指什么？第二个地方是SvUDIDTools.m中的kKeyC.....

--Just_Lee

[3. Re:UITableView学习笔记](#)

@QYMa_HAZA恩，我是西安人，不过还从来没有在西安工作过呢...

--一片-枫叶

[4. Re:UITableView学习笔记](#)

今天刚看到博主的博客，看到里面的示例图片，猜测楼主是西安的吧？我也是西安的，今年刚开始北漂生活。向博主学习！PS：转载了您的这篇文章。感谢！

--QYMa_HAZA

[5. Re:关于UIWindow的一点儿思考](#)

在视图A里点击按钮（presentViewController）到视图B，视图B里的代码是：- (void)viewDidLoad { [super viewDidLoad]; sel.....

--张雨涛

阅读排行榜

[1. UITableView学习笔记\(79882\)](#)

[2. UITabBarController详解\(54040\)](#)

[3. UINavigationController详解\(44093\)](#)

[4. iOS7: 如何获取不变的UDID\(33330\)](#)

[5. 正确使用PresentModalViewController\(25725\)](#)

评论排行榜

[1. iOS7: 如何获取不变的UDID\(61\)](#)

[2. iOS: FFMpeg编译和使用问题总结\(48\)](#)

[3. 工作三年的思考\(31\)](#)

[4. UITableView学习笔记\(31\)](#)

[5. 做IOS开发这一年\(30\)](#)

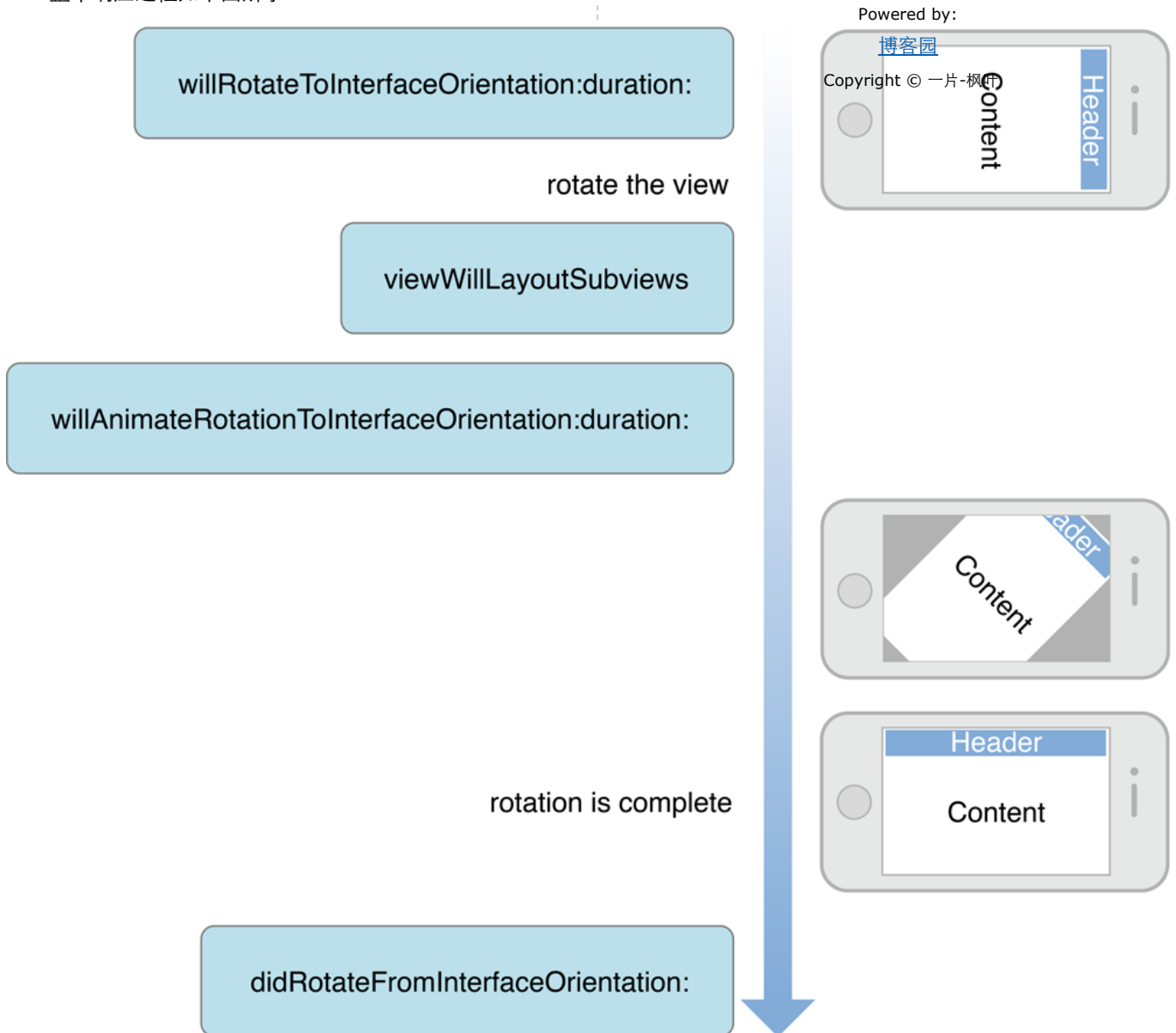
推荐排行榜

[1. UITableView学习笔记\(29\)](#)

[2. iOS7: 如何获取不变的UDID\(21\)](#)

方法, 通知view controller旋转动画执行完毕。
这个时候我们可以将第二部隐藏的view再显示出来。

整个响应过程如下图所示:



以上就是UIKit下一个完整的屏幕旋转流程, 我们只需要按照提示做出相应的处理就可以完美的支持屏幕旋转。

三、注意事项和建议

1) 注意事项

当我们的view controller隐藏的时候, 设备方向也可能发生变化。例如view Controller A弹出一个全屏的view controller B的时候, 由于A完全不可见, 所以就接收不到屏幕旋转消息。这个时候如果屏幕方向发生变化, 再dismiss B的时候, A的方向就会不正确。我们可以通过在view controller A的viewWillAppear中更新方向来修正这个问题。

2) 屏幕旋转时的一些建议

- 在旋转过程中，暂时界面操作的响应。
- 旋转前后，尽量当前显示的位置不变。
- 对于view层级比较复杂的时候，为了提高效率在旋转开始前使用截图替换当前的view层级，旋转结束后再将原view层级替换回来。
- 在旋转后最好强制reload tableview，保证在方向变化以后，新的row能够充满全屏。例如对于有些照片展示界面，竖屏只显示一列，但是横屏的时候显示列表界面，这个时候一个界面就会显示更多的元素，此时reload内容就是很有必要的。

注：以上内容和插图全部来自苹果官方文档：[View Controller Programming Guide for iOS](#)

分类: [IOS](#)

标签: [IOS](#), [Rotate](#)

绿色通道: [好文要顶](#) [关注我](#) [收藏该文](#) [与我联系](#)



[一片-枫叶](#)

[关注 - 50](#)

[粉丝 - 305](#)

[+加关注](#)

3

0

(请您对文章做出评价)

« 上一篇: [IOS: 屏幕旋转与Transform](#)

» 下一篇: [IOS: 聊一聊UIImage几点知识](#)

posted on 2013-04-24 23:28 [一片-枫叶](#) 阅读(11717)

评论(2) [编辑](#) [收藏](#)

评论

#1楼 2013-11-12 17:19 武松

楼主写的很好，赞！

支持(0) 反对(0)

#2楼 2015-03-31 21:55 best_man?

博主，向你学习。

支持(0) 反对(0)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或