

首页 > iOS开发

如何使用masonry设计复合型cell

2015-06-12 08:57 编辑: suiling 分类: iOS开发 来源: 里脊串的开发随笔 0 3995

设计Masonry复合型cell

招聘信息: iOS工程师&技术经理&总监



前言

其实早在@sunnyxx同学发布UIView-FDCollapsibleConstraints的时候 我就说要写一下怎么用代码来稍微麻烦的实现复用的问题 但是一直各种没时间(主要是我的办法太复杂 - -) 正好看到@叶孤城同学也说了一下他的解决办法 所以我来说一下我是如何解决这个问题的

分析

我们以叶孤城同学的例子来简单分析一下 假设view是这样的(为了方便 将所有的间隙设定为20)

正常的布局是这样的




布局代码如下:


```
@interface ComplexCell()

@property (nonatomic, strong) UIView *vB; //view blue height:30
```


热门资讯




8行代码教你搞定导航控制器全屏滑动返回
点击量 10075




不说鸡汤, 谈谈现实: 半路学编程能成大牛
点击量 7795




UIWebView与JS的深度交互
点击量 7767




超详细! iOS 并发编程之 Operation Queues
点击量 6721




打造安全的App! iOS 安全系列之 HTTPS
点击量 6665




为什么程序员的业余项目大多都死了?
点击量 6148




为什么技术总是被轻视? 不服来辩!
点击量 6110



一文让你彻底了解iOS 字体相关知识
点击量 5676



深入理解Objective-C: 方法缓存
点击量 4507



我的iOS高效编程秘诀 - 坚持编程习惯
点击量 4160

综合评论

我也要给我
047691641 评论了 七夕我们在一起, 送你12只萌萌哒的COCO熊! 速抢...

Objective-C runtime真心很强大
yiwanji 评论了 深入理解Objective-C: 方法缓存

什么都没有啊
xxfftyywc90 评论了 三年一个人使用虚幻引擎(UDK)开发的一个游戏心路

这个看起来很好用 不过现在 用的xcode6.3 需要升级才能用
hall_of_fame 评论了 iOS 9学习系列: UINavigationController如何让你...

遇到这种情况, 不要着急赞成, 也不要着急否认。即使你是一个有过几个成功
jarod姜赫 评论了 大神支招: 如何应对老板干涉游戏设计问题

牛, 都说到点上了。

```
@property (nonatomic, strong) UIView *vY; //view yellow  height:30

@property (nonatomic, strong) UIView *vR; //view red    height:30

@property (nonatomic, strong) UIView *vG; //view green  height:100

@end

- (instancetype)initWithStyle:(UITableViewCellStyle)style reuseIdentifier:(NSString *)reuseIdentifi


    self = [super initWithStyle:style reuseIdentifier:reuseIdentifier];

if ( self ) {

    CGFloat spacing = 20.0f;

    self.vB = [UIView new];
    [self.contentView addSubview:self.vB];

    [self.vB mas_makeConstraints:^(MASConstraintMaker *make) {

        make.left.top.equalTo(self.contentView).insets(UIEdgeInsetsMake(spacing,spacing,0,0));

        make.width.equalTo(@60);

        make.height.equalTo(@30).priorityLow();

        self.cB = make.height.equalTo(@0).priority(UILayoutPriorityRequired);

    }];

    self.vB.backgroundColor = [UIColor blueColor];


    self.vY = [UIView new];
    [self.contentView addSubview:self.vY];

    [self.vY mas_makeConstraints:^(MASConstraintMaker *make) {

        make.left.equalTo(self.vB.mas_right).offset(spacing);

        make.right.top.equalTo(self.contentView).insets(UIEdgeInsetsMake(spacing,0,0,spacing));

        make.height.equalTo(@30).priorityLow();

        self.cY = make.height.equalTo(@0).priority(UILayoutPriorityRequired);

    }];

    self.vY.backgroundColor = [UIColor yellowColor];


    self.vR = [UIView new];
    [self.contentView addSubview:self.vR];

    [self.vR mas_makeConstraints:^(MASConstraintMaker *make) {

        make.top.equalTo(self.vB.mas_bottom).offset(spacing);

        make.left.right.equalTo(self.contentView).insets(UIEdgeInsetsMake(0,spacing,0,spacing))

        make.height.equalTo(@30).priorityLow();

        self.cR = make.height.equalTo(@0).priority(UILayoutPriorityRequired);

    }];

    self.vR.backgroundColor = [UIColor redColor];


    self.vG = [UIView new];
    [self.contentView addSubview:self.vG];

    [self.vG mas_makeConstraints:^(MASConstraintMaker *make) {

        make.top.equalTo(self.vR.mas_bottom).offset(spacing);

        make.left.right.equalTo(self.contentView).insets(UIEdgeInsetsMake(0,spacing,0,spacing))

        make.height.equalTo(@100).priorityLow();

        self.cG = make.height.equalTo(@0).priority(UILayoutPriorityRequired);
```

jarod姜赫 评论了 国外游戏开发者：创
业路上9大禁区不可进

楼主一生平安!!!!

向宏simida 评论了 源码推荐(8.18)：类
似支付宝余额数字动画——贝...

66666666

fu779115154 评论了 阿峥教你实现
UITableView循环利用

作者的编程习惯是值得学习的，而且这
样做也能提高编码效率。这样的习惯也

icebergcwp1990 评论了 我的iOS高效
编程秘诀—坚持编程习惯

。。。。意义不明

leonajkl 评论了 阿峥教你实现
UITableView循环利用

相关帖子

请问怎么实现安卓上运行游戏，自动隐
藏3个虚拟按键？

哪里做假 护照【+9 2 7 9 6 7 6 9
1】银行流水账

哪里做假 机动车登记证【+9 2 7
9 6 7 6 9 1】保险单

哪里做假 大学文凭毕业证【+9
2 7 9 6 7 6 9 1】银行回单

apk送手机后帧数很低

App Store 怎么没有类别销售排名了。

高德地图逆地理编码结果没有数据，是
什么问题？

使用UIImagePickerController出现内存
泄漏的问题

有审核期间苹果要求留电话等来电的开
发者吗？

微博

CocoaChina
加关注

【最新 El Capitan 测试版默认关闭 D
ashboard 功能】在早些时候发布的
OS X El Capitan 第七个开发者测试
中，苹果默认关闭了 Dashboard 功
能。Dashboard 是 OS X 的桌面组件
功能，已经有10年左右的历史，在过
去4年之中一直没有更新。详见：htt
p://t.cn/RLDD7Rq



19分钟前 转发 | 评论(3)

【Swift-er SDK：简述OC的Nullabilit
y Annotations特性】苹果在Xcode
6.3中为Objective-C添加了一项新特

```
    }  
    self.vG.backgroundColor = [UIColor greenColor];  
  
    }  
  
    return self;  
}
```

实际效果如图



看上去还不错

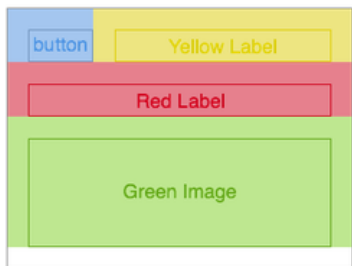
在Masonry中 针对单条的**MASLayoutConstraint**可以进行**active**和**deactive**操作 那么意味着可以动态的启用或者禁用某条预置的约束 所以我们只要预先设置一条高优先级的**高度为0(或者宽度为0)**的约束 然后在适当的时候激活它不就行了? 先尝试隐藏红色的view 隐藏后如下



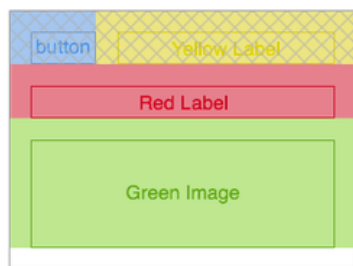
啊~哦~ 结果不正确 隐藏是隐藏了 但是间隙没有隐藏 导致缝变大了 这是因为我们仅仅隐藏了view 而没有隐藏view之间的间隔 那么应该如何处理这种情况呢?

主流的做法是将这个view的所有约束值全设置成0 然后恢复的时候再还原 这种方法需要记录原值 但是在前言我说了要用稍微麻烦的方法来解决这个问题 所以肯定不是这样做啦

我采用的方法是group法 具体如下图



其实在第一行还有一个groupview如图



但是因为图显示出来不太好看(不会画图 T_T) 所以我隐藏了 具体可以看代码细节

每个(或者每组)可以隐藏的view 都对应有一个group view(group其实就是包含了view和spacing) 需要隐藏的时候 直接隐藏这个group 就可以达到既隐藏view又缩短间隙的目的

代码较长 大家可以选择跳过 - -!

```
@interface ComplexCell()
@property (nonatomic, strong) MASConstraint *cF; //constraint first row
@property (nonatomic, strong) MASConstraint *cB; //constraint blue
@property (nonatomic, strong) MASConstraint *cY; //constraint yellow
@property (nonatomic, strong) MASConstraint *cR; //constraint red
@property (nonatomic, strong) MASConstraint *cG; //constraint green
@property (nonatomic, strong) UIView *gF; //group first row
@property (nonatomic, strong) UIView *gB; //group blue
@property (nonatomic, strong) UIView *gY; //group yellow
@property (nonatomic, strong) UIView *gR; //group red
@property (nonatomic, strong) UIView *gG; //group green
@property (nonatomic, strong) UIView *vB; //view blue    height:30
@property (nonatomic, strong) UIView *vY; //view yellow height:30
@property (nonatomic, strong) UIView *vR; //view red    height:30
@property (nonatomic, strong) UIView *vG; //view green  height:100
@end

@implementation ComplexCell
- (instancetype)initWithStyle:(UITableViewCellStyle)style reuseIdentifier:(NSString *)reuseIdentifi

    self = [super initWithStyle:style reuseIdentifier:reuseIdentifier];

    if ( self ) {

        CGFloat spacing = 20.0f;

        self.gF = [UIView new];
        [self.contentView addSubview:self.gF];
        [self.gF mas_makeConstraints:^(MASConstraintMaker *make) {
            make.left.top.right.equalTo(self.contentView);

            self.cF = make.height.equalTo(@0).priority(UILayoutPriorityRequired);
            [self.cF deactivate];
        }];
        self.gF.clipsToBounds = YES;
```

```
self.gB = [UIView new];
[self.gF addSubview:self.gB];
[self.gB mas_makeConstraints:^(MASConstraintMaker *make) {
    make.left.top.bottom.equalTo(self.gF);

    self.cB = make.width.equalTo(@0).priority(UILayoutPriorityRequired);
    [self.cB deactivate];
}];
self.gB.clipsToBounds = YES;

self.gY = [UIView new];
[self.gF addSubview:self.gY];
[self.gY mas_makeConstraints:^(MASConstraintMaker *make) {
    make.right.top.bottom.equalTo(self.gF);
    make.left.equalTo(self.gB.mas_right);

    self.cY = make.width.equalTo(@0).priority(UILayoutPriorityRequired);
    [self.cY deactivate];
}];
self.gY.clipsToBounds = YES;

self.gR = [UIView new];
[self.contentView addSubview:self.gR];
[self.gR mas_makeConstraints:^(MASConstraintMaker *make) {
    make.left.right.equalTo(self.contentView);
    make.top.equalTo(self.gF.mas_bottom);

    self.cR = make.height.equalTo(@0).priority(UILayoutPriorityRequired);
    [self.cR deactivate];
}];
self.gR.clipsToBounds = YES;

self.gG = [UIView new];
[self.contentView addSubview:self.gG];
[self.gG mas_makeConstraints:^(MASConstraintMaker *make) {
    make.left.right.equalTo(self.contentView);
    make.top.equalTo(self.gR.mas_bottom);

    self.cG = make.height.equalTo(@0).priority(UILayoutPriorityRequired);
    [self.cG deactivate];
}];
self.gG.clipsToBounds = YES;

self.vB = [UIView new];
[self.gB addSubview:self.vB];
[self.vB mas_makeConstraints:^(MASConstraintMaker *make) {
    make.edges.equalTo(self.gB).insets(UIEdgeInsetsMake(spacing, spacing, 0, 0)).priorityLo
    make.height.equalTo(@30);
    make.width.equalTo(@60);
}];
self.vB.backgroundColor = [UIColor blueColor];
```

```

        self.vY = [UIView new];
        [self.gY addSubview:self.vY];
        [self.vY mas_makeConstraints:^(MASConstraintMaker *make) {
            make.edges.equalTo(self.gY).insets(UIEdgeInsetsMake(spacing, spacing, 0, spacing)).prio
            make.height.equalTo(@30);
        }];
        self.vY.backgroundColor = [UIColor yellowColor];

        self.vR = [UIView new];
        [self.gR addSubview:self.vR];
        [self.vR mas_makeConstraints:^(MASConstraintMaker *make) {
            make.edges.equalTo(self.gR).insets(UIEdgeInsetsMake(spacing, spacing, 0, spacing)).prio
            make.height.equalTo(@30);
        }];
        self.vR.backgroundColor = [UIColor redColor];

        self.vG = [UIView new];
        [self.gG addSubview:self.vG];
        [self.vG mas_makeConstraints:^(MASConstraintMaker *make) {
            make.edges.equalTo(self.gG).insets(UIEdgeInsetsMake(spacing, spacing, 0, spacing)).prio
            make.height.equalTo(@100);
        }];
        self.vG.backgroundColor = [UIColor greenColor];
    }

    return self;
}

```

然后 为每种不同的布局定义一个枚举(为了举例我随便定义的 0和1代表这个view是否被显示)

```

typedef NS_ENUM(NSUInteger, ComplexType) {
    ComplexType1111,
    ComplexType1110,
    ComplexType0111,
    ComplexType0011,
    ComplexType0010,
    ComplexType1101
};

@interface ComplexCell : UITableViewCell
@property (nonatomic, assign) ComplexType type;
@end

- (void)setType:(ComplexType)type {

    [self.cF deactivate];
    [self.cB deactivate];
    [self.cY deactivate];
    [self.cR deactivate];
    [self.cG deactivate];
}

```

```

switch (type) {
    case ComplexType1111:
    {
        break;
    }
    case ComplexType0111:
    {
        [self.cB activate];
        break;
    }
    case ComplexType0011:
    {
        [self.cF activate];
        break;
    }
    case ComplexType1110:
    {
        [self.cG activate];
        break;
    }
    case ComplexType1101:
    {
        [self.cR activate];
        break;
    }
    case ComplexType0010:
    {
        [self.cF activate];
        [self.cG activate];
        break;
    }

    default:
        break;
}
}

```

这样 在tableview的datasource中我们只要这样做就可以了

```

- (NSInteger)numberOfSectionsInTableView:(UITableView *)tableView {
    return 1;
}

- (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSectionSection:(NSInteger)section {
    return 6;
}

- (CGFloat)tableView:(UITableView *)tableView heightForRowAtIndexPath:(NSIndexPath *)indexPath {
    return [ComplexCell getHeightByType:indexPath.row%6];
}

- (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath

    ComplexCell* cell = [tableView dequeueReusableCellWithIdentifier:@"cell"];
    cell.type = indexPath.row%6;

```

看看效果 是不是很不错



文中的demo可以[在这里找到](#) 要注意的地方是约束的priority的设置 另外 这种方式也支持不定长内容的Autolayout

使用group的方式 面对同时在横向和纵向都有隐藏要求的时候 会方便很多

比如文中举的例子 第一行有时会隐藏蓝色的按钮 有时整个一行都会不显示 这样的话 当我想隐藏按钮时 只要激活按钮的约束 想隐藏整行时 只要激活整行的那条约束就行了



微信扫一扫

订阅每日移动开发及APP推广热点资讯

公众号：CocoaChina

我要投稿

收藏文章

分享到：

12

上一篇： iOS与Android的音频互通

下一篇： 应用内购（In-App Purchase）常见问题解答

相关资讯	
6个iOS图片文本设计小技巧	IOS自适应前段库-Masonry的使用
那些在学习iOS开发前就应该知道的事（part 1）	iOS应用架构谈 网络层设计方案
有趣的Autolayout示例-Masonry实现	如何自己动手实现 KVO
AutoLayout 布局技巧 – 等宽子视图	浅谈iOS中MVVM的架构设计与团队协作
UIMotionEffect和Home页背景视差效果	iOS开发使用半透明模糊效果方法整理

清理我的 Mac

MacKeeper - 即刻清理 Mac。 确保 Mac 安全。

文章评论 (0)

对这篇文章有什么感想，写一下吧.....

发表评论

