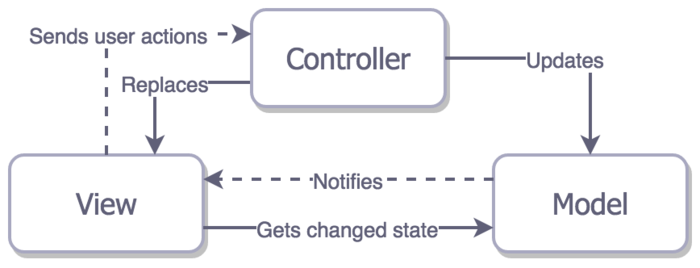
mvc:

## 传统mvc

v聚合m



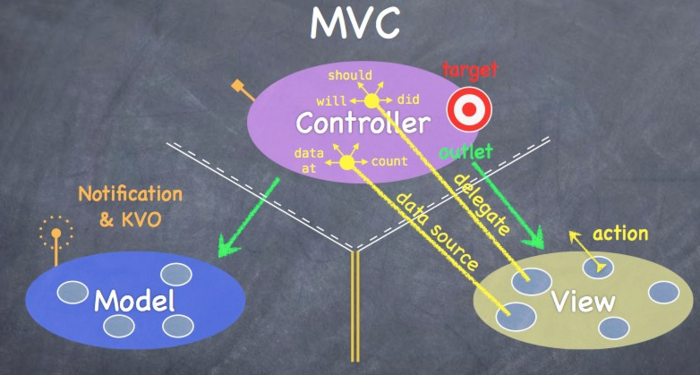
缺点：view无法复用

## apple mvc，

c组合m，c组合v

[官方说明](https://developer.apple.com/library/content/documentation/General/Conceptual/DevPedia-CocoaCore/MVC.html)：





[斯坦福大学MVC架构](https://open.163.com/movie/2015/2/M/7/MAIKHN60A_MAIPGF0M7.html)

都没有明确定义业务逻辑放哪里。所以会出现：

1，全写在contorller里

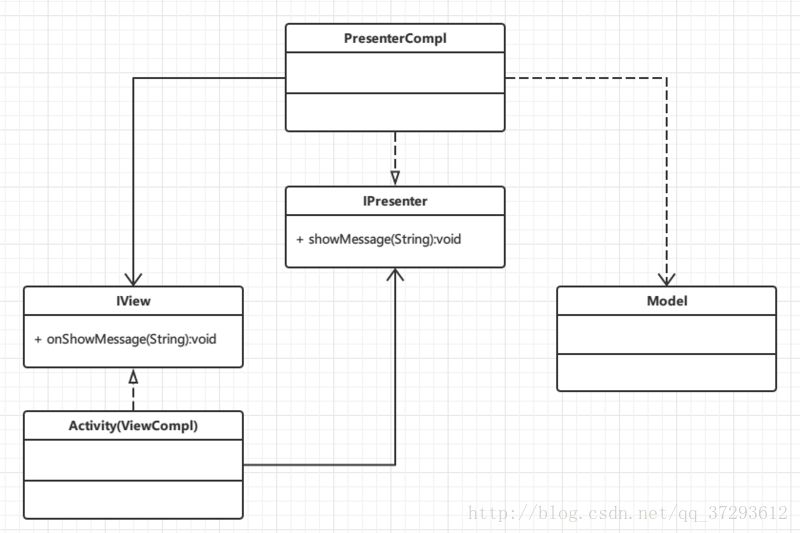
2，全写在model里

3，部分写model里，部分写controller里

## mvp：

[MVP设计模式的实现](http://www.cnblogs.com/qq260250932/p/4238541.html)

[开发模式——MVP框架开发](http://blog.csdn.net/qq_37293612/article/details/54879716)



v = view+uiviewcontroller，在uiviewcontroller里布局

p负责业务逻辑，协调view和model

uiviewcontroller不负责跳转，需要一个路由层来控制页面跳转

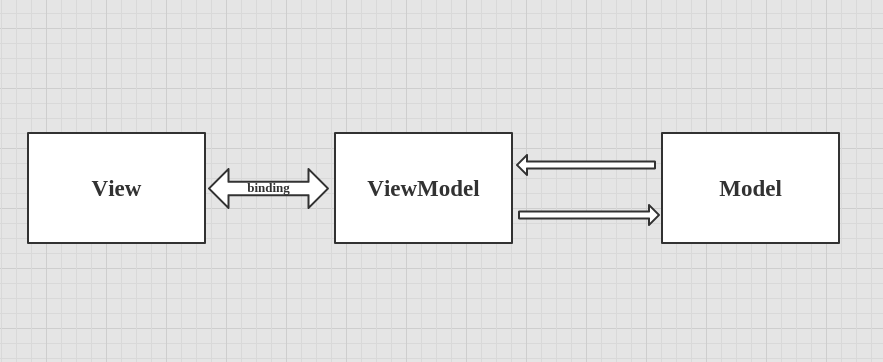
用接口的方式来对接

优点：p可以单元测试，view可以被模拟。

缺点：类多，需要写很多方法做类之间的通讯

## mvvm:

MVVM 是一种软件架构模式，它是 Martin Fowler 的 Presentation Model 的一种变体，最先由微软的架构师 John Gossman 在 2005 年提出，并应用在微软的 WPF 和 Silverlight 软件开发中。MVVM 衍生于 MVC ，是对 MVC 的一种演进，它促进了 UI 代码与业务逻辑的分离。



v = view+uiviewcontroller

双向绑定

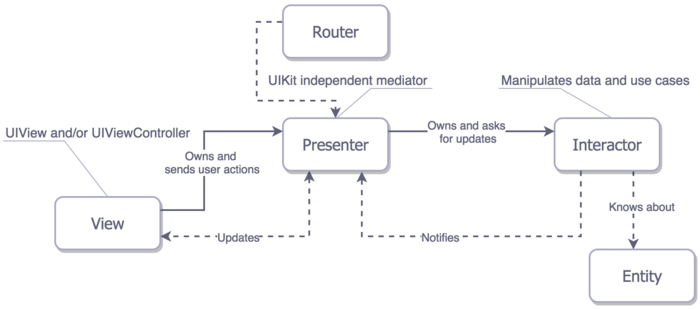
优点：更新model后自动更新view，无需手动写代码来更新view，减少代码量

缺点：1，堆栈长

2，不兼容其他模式

[被误解的MVC和被神化的MVVM](http://www.infoq.com/cn/articles/rethinking-mvc-mvvm)

## viper



[使用VIPER构建iOS应用程序](https://www.objc.io/issues/13-architecture/viper/)

v = view+uiviewcontroller

i = 业务逻辑

p = 显示逻辑

e = 纯数据

r = 路由控制页面跳转

presenter接口interactor

presenter接口view

interactor持有entity