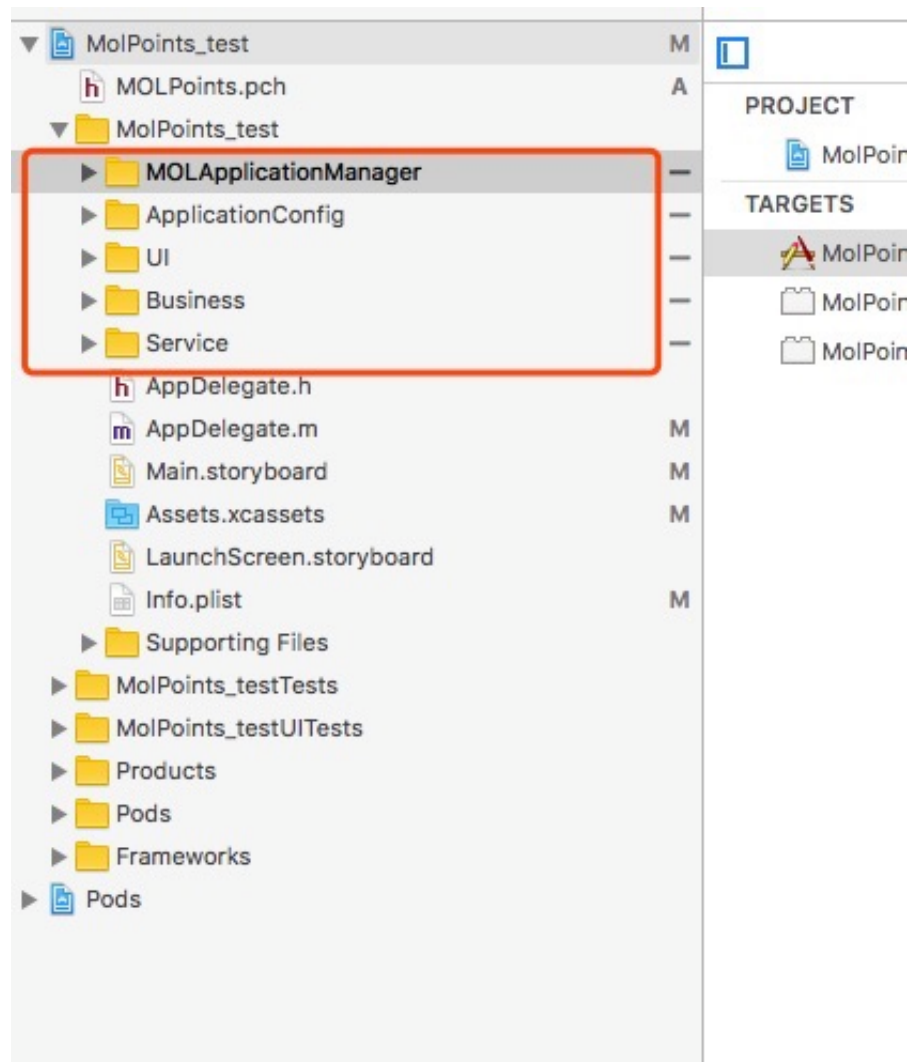


iOS版本MOLPoints Application 架构设计及分析

总框架一览



整个MOLPoints Application的整个大体框架如上图，分为以下几个大的部分：

- 1 MOLApplicationManager ———— 处理一些app全局的事项，如管理用户的登录信息，全局网络状态的监控等。
- 2 ApplicationConfig ———— app全局需要用到的常量定义，以及app网络环境的全局切换等。
- 3 UI ———— app所有的界面视图。
- 4 Business ———— app所有的业务处理
- 5 Service ———— 服务于整个app的工具。

然后，app所有需要用到的三方库，使用cocoapods做统一管理，方便更新三方的库，以及处理三方库的依赖关系。

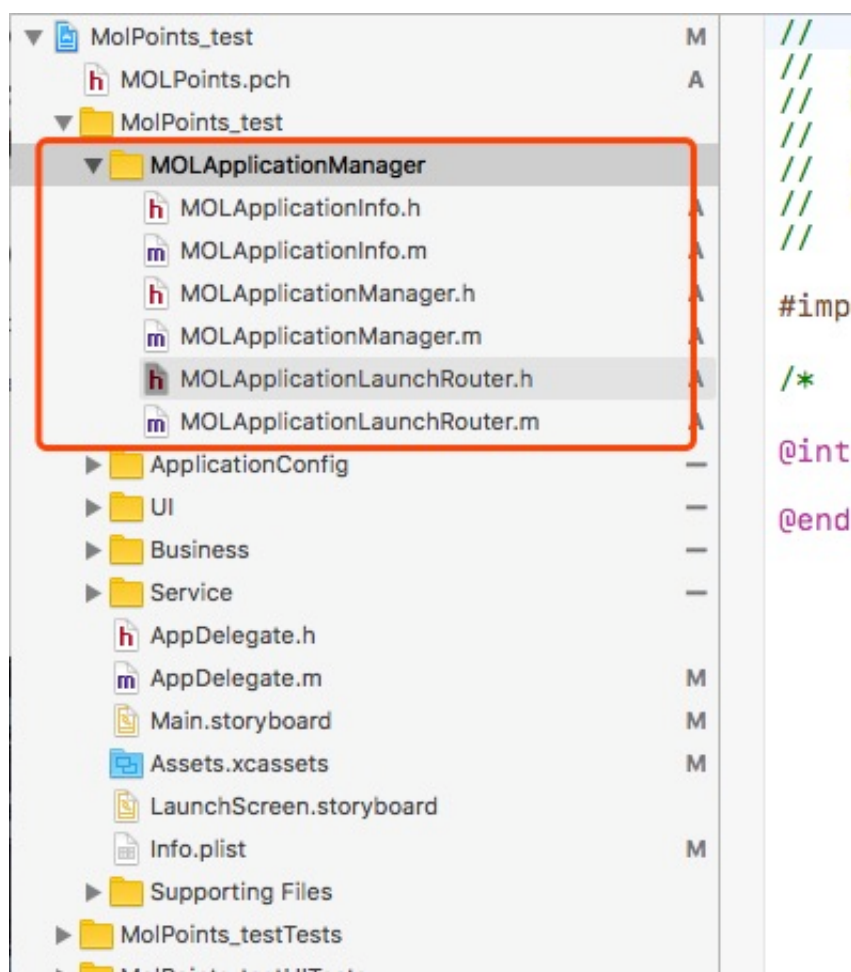
此外，整个app内部的跳转逻辑，包含原生与原生页面，原生与web页面之间的跳转，以及外部链接打开app

的逻辑，包括浏览器链接打开app，外部其他app打开app，统一采用router的管理模式，这样做的好处有以下几点：

- 整个app的跳转逻辑统一管理，而不根据业务分散，方便app源码的阅读，以及跳转逻辑的理解。
- 解除因为业务强耦合的关系，使得代码更加优雅。
- 处理复杂跳转关系的能力增加，例如根据外部链接跳转至app中层级较深的页面。
- 方便了原生页面与web页面的交互。

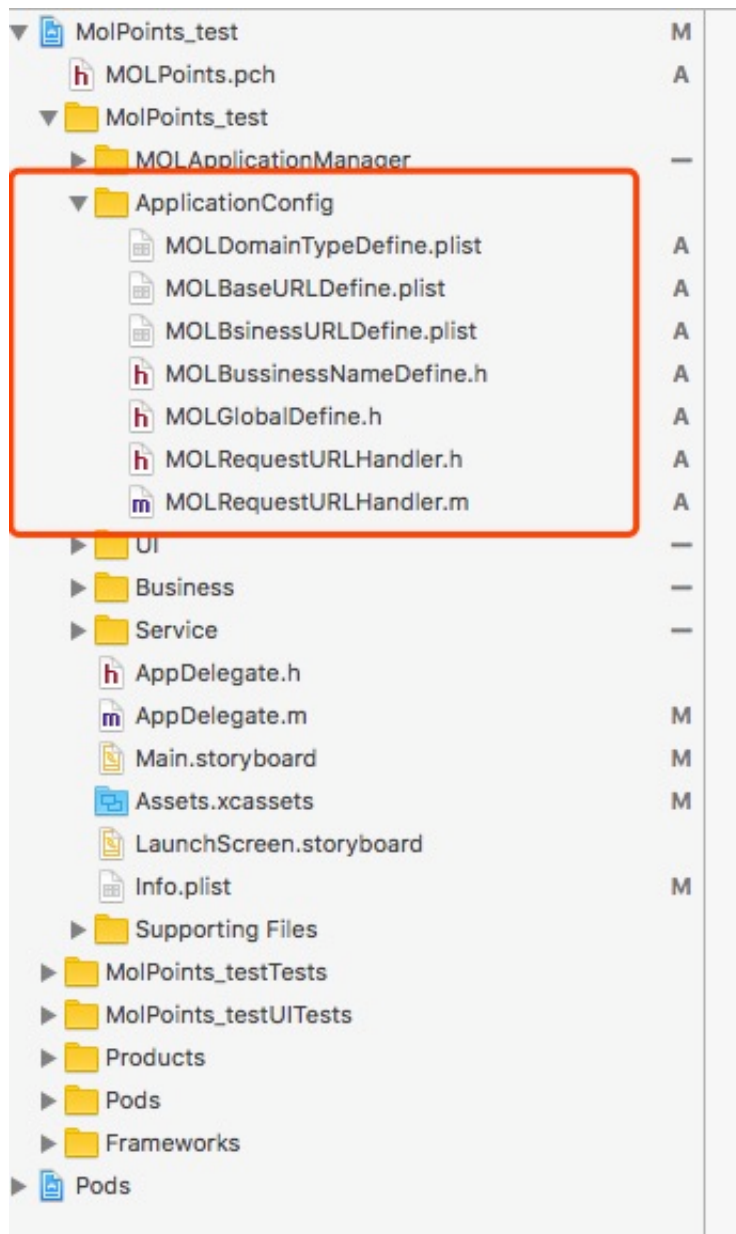
下面我将分条详述每个folder的作用，以及为什么要这样设计。

MOLApplicationManager



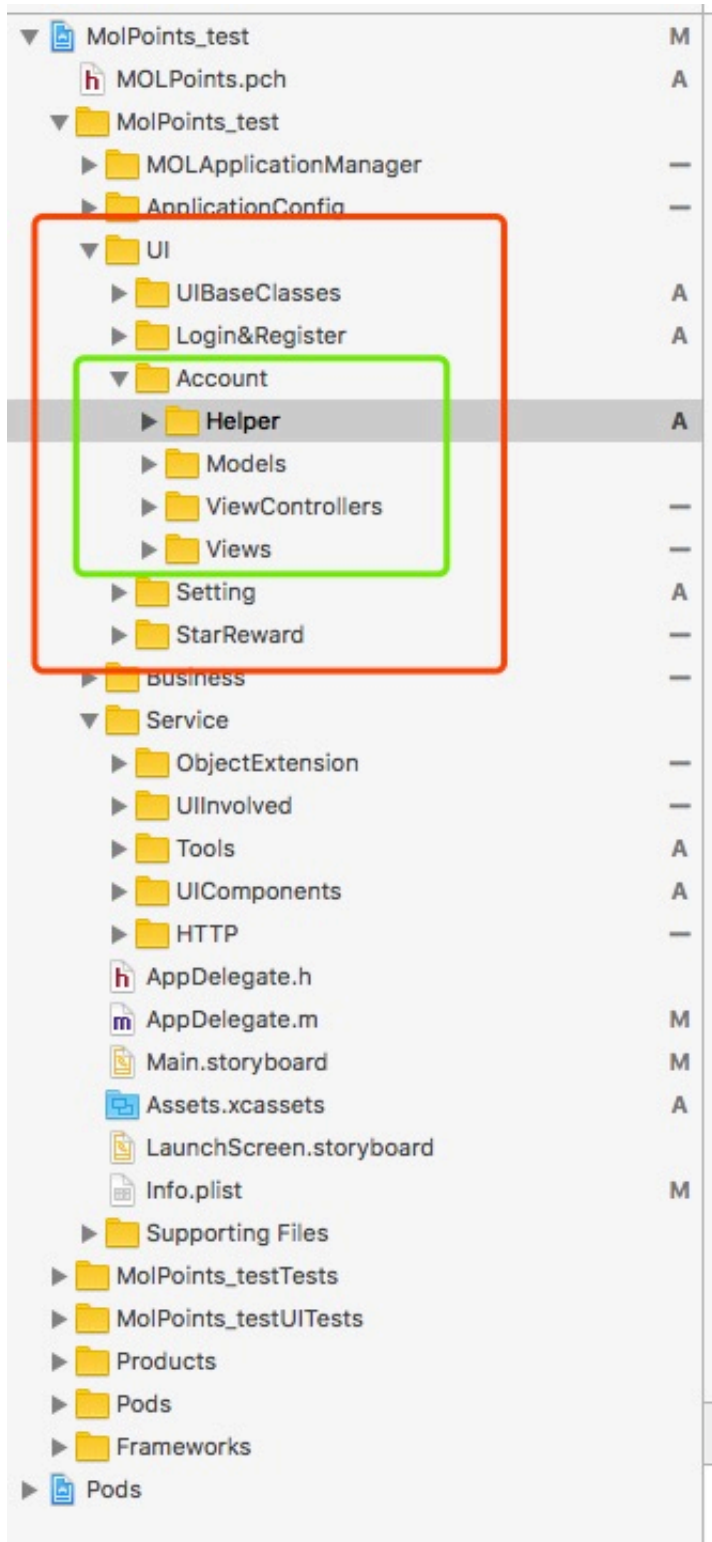
- **MOLApplicationInfo** ——— 目前在这个类里面处理了监控app整个网络状态的逻辑，需要app在全局的类中用到的信息都将放于此处。
- **MOLApplicationManager** ——— 将app的各种组件的注册，以及app状态的管理放在这个类中。以上两个类都可以减少AppDelegate中代码量，使app启动时的部分逻辑剥离，显得更清晰。
- **MOLApplicationLaunchRouter** ——— 处理外部链接，Universal Link等方式打开App的跳转，存储启动项等信息。

ApplicationConfig



- MOLDomainTypeDefine ——— 定义MOLPoints client URL的domain。
- MOLBaseURLDefine ——— 定义MOLPoints client URL的base URL。
- MOLBsinessURLDefine ——— 根据业务定义MOLPoints client URL的path。
- MOLBussinessNameDefine ——— 将业务名称定义为常量，方便使用。
- MOLGlobalDefine ———
- MOLRequestURLHandler ——— 整合以上对URL的定义，并读取定义后按需拼接为一个完整的URL，动态的化的拼接，易于app api环境的切换，增加等。

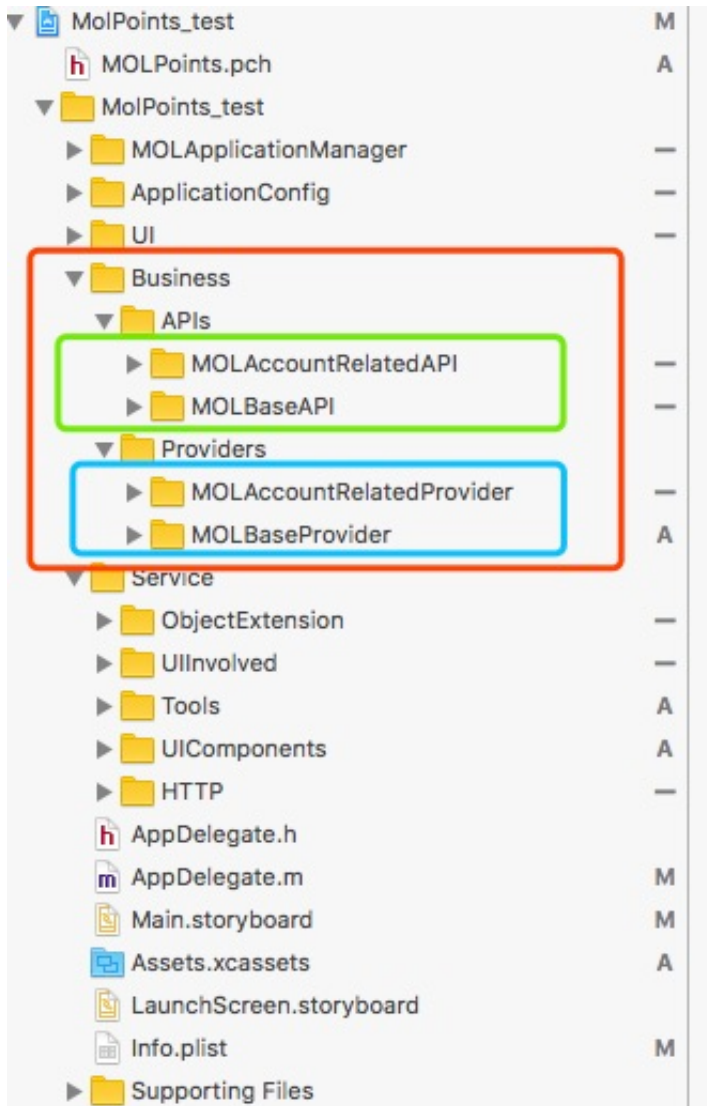
UI



UI层包含app所有的视图，MOLPoints可以划分为如图的四大部分：Login&Register，Account，StarReward，Setting。以Account为例，说明具体的机构。此外UIBaseClasses，指UI中可以进行抽象的基础类，方便更多子类快速的获得功能。

- Helper ——— 一般用于分离ViewController中的业务逻辑，使ViewController的更清晰易读。
- Models ——— 拆分好的，对应视图的显示需要的数据。
- ViewController ——— 控制器，协调数据与视图之前的通信。
- View ——— 具体的加载数据的视图。

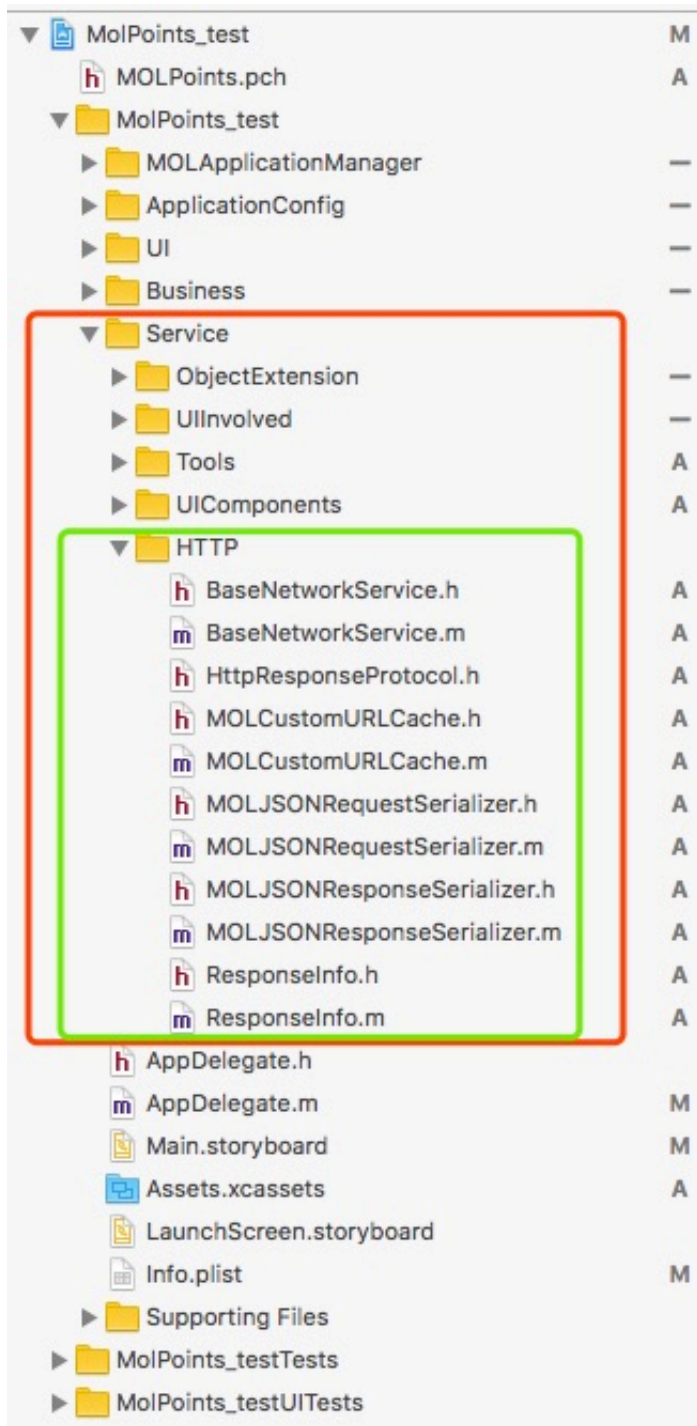
Business



Business层本身是负责app的业务处理，业务其实似根据数据衍生，所以Business也可以认为是处理并包装向服务器请求回来的数据。Business可以细分为Provider，API。另外，API,Provider都有自己的基类，用于实现一些公用的接口。

- API ——— 这一层主要负责向服务器请求数据，请求回来的json数据转化为数据的entity，纯粹的得到数据，并不根据视图显示的需要拆分。得到的数据，callback至Provider层。
- provider ——— 在得到API层的数据之后，进行进一步的细化，处理这些数据，然后再进行细化的包装，最后UI层进行调用。

Service



Service层是服务于app的工具，组件等，以对http请求的封装为例说明，http folder下面有对网络请求缓存的封装，数据解析的封装，服务器返回相应协议的制定等等。

总结

整个MOLPoints的框架设计就是这样的，但是或许会根据后续具体的业务，做进一步的完善，毕竟，跟具体业务最适合的才是最好的。