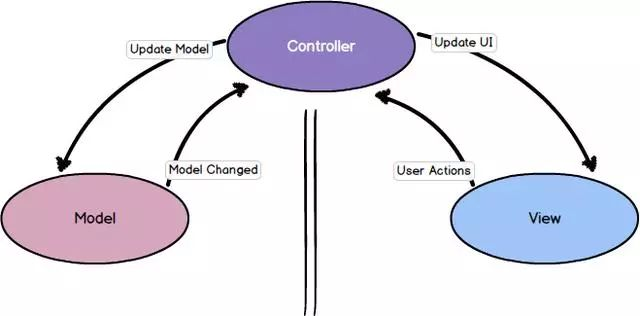
“饿了么”软件架构：

1. MVC架构

最初采用MVC架构，优点层次结构简单清晰、代码易于开发。

在这种架构中，Controller层负责整个APP中主要逻辑功能的实现；Model层则负责数据结构的描述以及数据持久化的功能；而View层作为展现层负责渲染整个APP的UI。分工清晰，简洁明了；并且这种系统架构在语言框架层就得到了Apple的支持，所以非常适用于APP的startup开发。

然而，这种架构在开发的后期会由于其超高耦和性，从而造就庞大Controller层，这就成为了MVC架构的一大缺点。

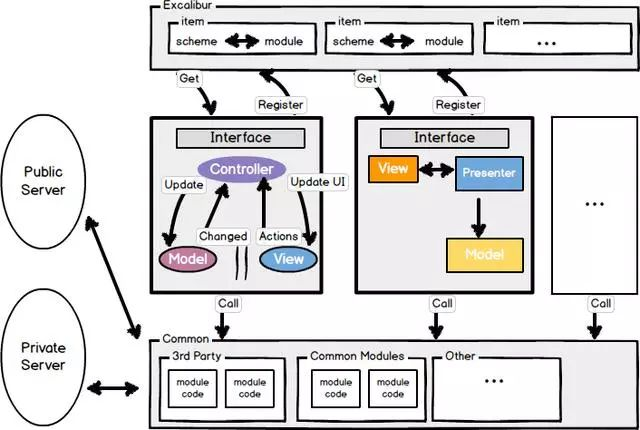


2. EMC架构

随后为了降低耦合，复用已有模块，采用了EMC架构，这种架构以高内聚、低耦合为主要的特点，以面向接口编程为出发点，降低了模块与模块之间的联系。

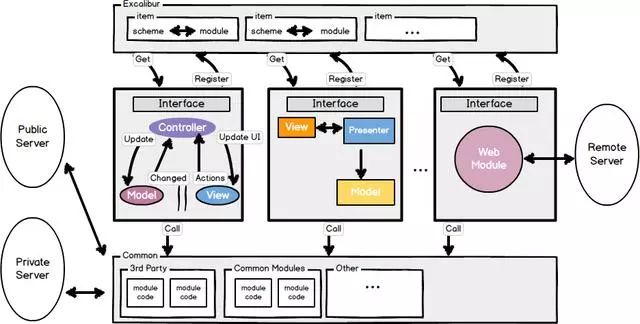
该架构的另外一大好处则在于解决了不同系统版本的兼容性问题。这里举iOS平台下的WebView作为例子来进行说明。Apple从iOS8系统开始提供了一套更好的Web支持框架——WebKit，但在iOS7系统下却无法兼容，从而导致Crash。

使用此类架构，可以在iOS7系统下仍然注册使用传统的WebView来渲染网页，而在iOS8及其以上系统注册WebKit来作为渲染网页的内核。即避免了Apple严格的审核机制，又达到了动态加载的目的。



3. Hybrid-EMC架构

随后这种架构综合了Native App和Web App二者的优缺点，纯粹展示性的模块使用Web页面来达到渲染的目的，而更多的数据操作性、动画渲染性的模块则采用Native的方式。



4. HotPatch-EMC架构

随后这种架构的主要目标在于解决移动APP的稳定性问题。通过RN与Native的主备，可以减少系统APP出错带来的失误成本。

