

Android 应用程序框架

知识解析

MVC 简单介绍

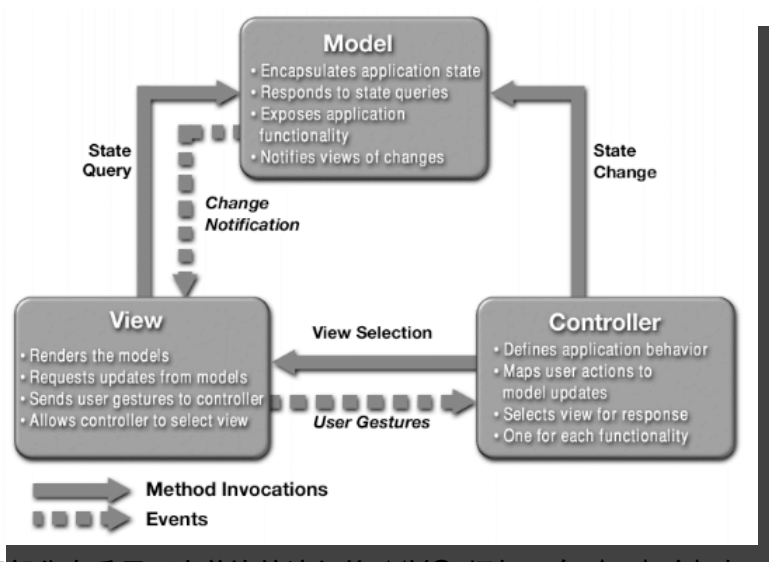
Android 应用程序框架采用 MVC 模式。MVC(Model-View-Controller, 模型-视图-控制器模式)用于表示一种软件架构模式。MVC 把软件分为三个基本部分:

模型(Model)

视图(View)

控制器(Controller)

MVC 开始是存在于桌面程序中的, M 是指业务模型, V 是指用户界面, C 则是控制器, 使用 MVC 的目的是将 M 和 V 的实现代码分离, 从而使同一个程序可以使用不同的表现形式。比如一批统计数据可以分别用柱状图、饼图来表示。C 存在的目的则是确保 M 和 V 的同步, 一旦 M 改变, V 应该同步更新。



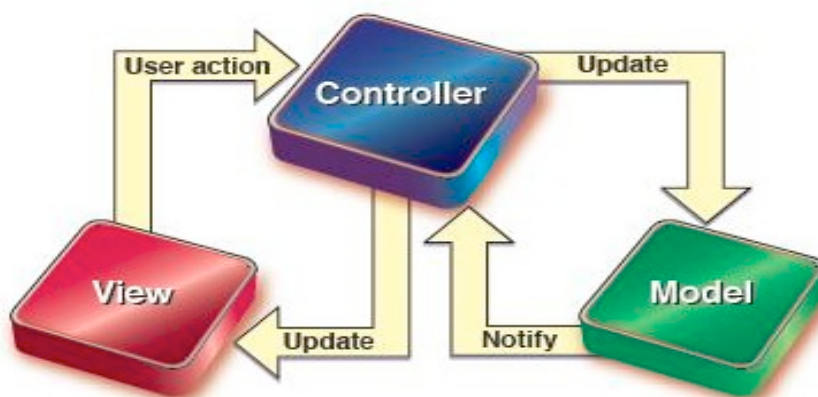
Android 中界面部分也采用了当前比较流行的 MVC 框架, 在 Android 中:

1) 视图层 (View): 一般采用 XML 文件进行界面的描述, 使用的时候可以非常方便的引入。当然, 如何你对 Android 了解的比较的多了话, 就一定可以想到在 Android 中也

可以使用 JavaScript+HTML 等方式作为 View 层，当然这里需要进行 Java 和 JavaScript 之间的通信，幸运的是，Android 提供了它们之间非常方便的通信实现。

2) 控制层 (Controller)：Android 的控制层的重任通常落在了众多的 Activity 的肩上，这句话也就暗含了不要在 Activity 中写代码，要通过 Activity 交割 Model 业务逻辑层处理，这样做的另外一个原因是 Android 中的 Activity 的响应时间是 5s，如果耗时的操作放在这里，程序就很容易被回收掉。

3) 模型层 (Model)：对数据库的操作、对网络等的操作都应该在 Model 里面处理，当然对业务计算等操作也是必须放在的该层的。就是应用程序中二进制的数据库。



在 Android SDK 中的数据绑定，也都是采用了与 MVC 框架类似的方法来显示数据。在控制层上将数据按照视图模型的要求（也就是 Android SDK 中的 Adapter）封装就可以直接在视图模型上显示了，从而实现了数据绑定。比如显示 Cursor 中所有数据的 ListActivity，其视图层就是一个 ListView，将数据封装为 ListAdapter，并传递给 ListView，数据就在 ListView 中现实。

职业素质

在工作中，我们可以用更多的设计模式来完善精简我们的项目。而MVC就是其中最为经典最为实用的模式。MVC要求对应用分层，虽然要花费额外的工作，但产品的结构清晰，产品的应用通过模型可以得到更好地体现。

首先，最重要的是应该有多个视图对应一个模型的能力。在目前用户需求的快速变化下，可能有多种方式访问应用的要求。

由于一个应用被分离为三层，因此有时改变其中的一层就能满足应用的改变。一个应用的业务流程或者业务规则的改变只需改动MVC的模型层。

控制层的概念也很有效，由于它把不同的模型和不同的视图组合在一起完成不同的请求，因此，控制层可以说是包含了用户请求权限的概念。

最后，它还有利于软件工程化管理。由于不同的层各司其职，每一层不同的应用具有某些相同的特征，有利于通过工程化、工具化产生管理程序代码。