由于上学期的课程学习，所以对MVC架构模式有一些了解，MVC架构相比其他的一些架构模式比较简单。主要应用于web端。这个也跟我们小组的项目有关联。然后我又了解了一下C/S架构**，**腾讯就是以C/S架构起家的，其旗下的QQ这一软件就是典型的C/S结构应用。

**MVC**

模型-视图-控制器模式，也称为MVC模式（Model View Controller）。用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC被发展起来用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。它把软件系统分为三个基本部分：

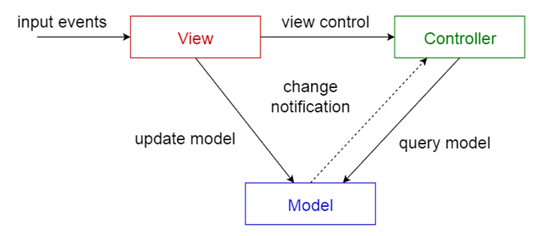
模型（Model）：负责存储系统的中心数据。

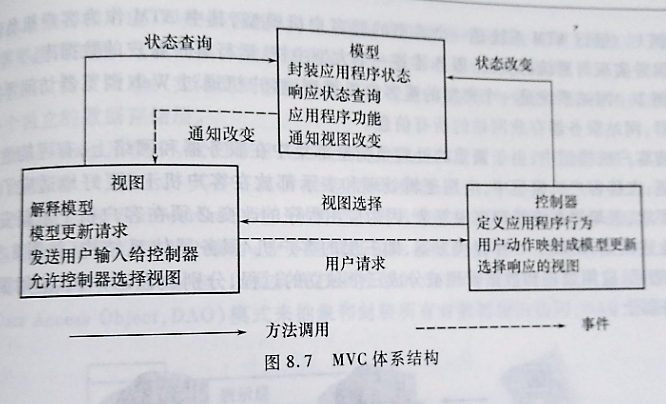
视图（View）：将信息显示给用户（可以定义多个视图）。

控制器（Controller）：处理用户输入的信息。负责从视图读取数据，控制用户输入，并向模型发送数据，是应用程序中处理用户交互的部分。负责管理与用户交互控制。

　　视图和控制器共同构成了用户接口。

　　且每个视图都有一个相关的控制器组件。控制器接受输入，通常作为将鼠标移动、鼠标按钮的活动或键盘输入编码的时间。时间被翻译成模型或试图的服务器请求。用户仅仅通过控制器与系统交互。





**结构**

　　模型组件包含应用程序的功能内核，他封装了相应的数据并输出执行特定应用程序处理的过程；模型也提供访问数据的函数，这些数据有获得待显示的数据的视图组件使用。

　　控制器代表用户调用所有这些过程。

**目的**

　　实现一种动态的程序设计，是后序对程序的修改和扩展简化，并且使程序某一部分的重复利用称为可能。

　　通过对复杂度的简化，使程序结构更加直观。

　　将信息的内部表示与信息的呈现方式分离开来，并接受用户的请求。它分离了组件，并允许有效的代码重用。即，将模型和视图的实现代码分离，从而使同一个程序可以使用不同的表现形式。比如一批统计数据你可以分别用柱状图、饼图来表示。C存在的目的则是确保模型和视图的同步，一旦模型改变，视图应该同步更新。

**实例**

应用于基于MVC架构模式的框架，常见的服务器端MVC框架有：Struts、Spring MVC、ASP.NET MVC、Zend Framework、JSF；常见前端MVC框架：angularjs、reactjs、backbone；由MVC演化出了另外一些模式如：MVP、MVVM

**MVC的优点**

耦合性低：视图层和业务层分离，这样就允许更改视图层代码而不用重新编译模型和控制器代码，同样，一个应用的业务流程或者业务规则的改变只需要改动 MVC 的模型层即可。因为模型与控制器和视图相分离，所以很容易改变应用程序的数据层和业务规则。

重用性高：一个模型能被多个视图共享，这意味着从模型传回来的数据能以多种方式显示，而不需要为每一种显示方式编写特定的代码，大大减少了代码量。

部署快：使用 MVC 模式使开发时间得到相当大的缩减，后台程序员集中精力于业务逻辑，前端程序员集中精力于表现形式上。

可维护性高：分离视图层和业务逻辑层也使得 WEB 应用更易于维护和修改。

有利软件工程化管理：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、工具化管理程序代码。

**MVC的缺点**

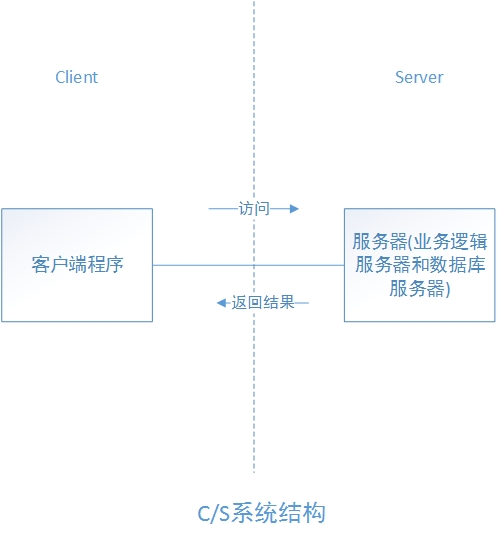
增加系统结构和实现的复杂性：对于简单的界面，严格遵循 MVC，使模型、视图与控制器分离，会增加结构的复杂性，并可能产生过多的更新操作，降低运行效率。所以说MVC不适合用于小型、中型应用程序。

视图与控制器间的过于紧密的连接：视图与控制器是相互分离，但却是联系紧密的部件，视图没有控制器的存在，其应用是很有限的，反之亦然，这样就妨碍了他们的独立重用。

视图对模型数据的低效率访问：依据模型操作接口的不同，视图可能需要多次调用才能获得足够的显示数据。对未变化数据的不必要的频繁访问，也将损害操作性能。

**C/S**

C/S架构即客户机和服务器架构。它是软件系统体系结构，通过它可以充分利用两端硬件环境的优势，将任务合理分配到Client端和Server端来实现，降低了系统的通讯开销。



C/S结构在技术上很成熟，它的主要特点是交互性强、具有安全的存取模式、网络通信量低、响应速度快、利于处理大量数据。因为客户端要负责绝大多数的业务逻辑和UI展示，又称为胖客户端。它充分利用两端硬件，将任务分配到Client 和Server两端，降低了系统的通讯开销。C/S结构的软件需要针对不同的操作系统系统开发不同版本的软件，加之产品的更新换代十分快，已经很难适应百台电脑以上局域网用户同时使用。。

C/S 架构是一种典型的两层架构，其客户端包含一个或多个在用户的电脑上运行的程序，而服务器端有两种，一种是数据库服务器端，客户端通过数据库连接访问服务器端的数据；另一种是Socket服务器端，服务器端的程序通过Socket与客户端的程序通信。

**（1）优点：**

能充分发挥客户端PC的处理能力，很多工作可以在客户端处理后再提交给服务器，所以CS客户端响应速度快。

操作界面漂亮、形式多样，可以充分满足客户自身的个性化要求。

C/S结构的管理信息系统具有较强的事务处理能力，能实现复杂的业务流程。

安全性能可以很容易保证，C/S一般面向相对固定的用户群，程序更加注重流程，它可以对权限进行多层次校验，提供了更安全的存取模式，对信息安全的控制能力很强。一般高度机密的信息系统采用C/S结构适宜。

**（2）缺点：**

需要专门的客户端安装程序，分布功能弱，针对点多面广且不具备网络条件的用户群体，不能够实现快速部署安装和配置。

兼容性差，对于不同的开发工具，具有较大的局限性。若采用不同工具，需要重新改写程序。

开发、维护成本较高，需要具有一定专业水准的技术人员才能完成，发生一次升级，则所有客户端的程序都需要改变。。

用户群固定。由于程序需要安装才可使用，因此不适合面向一些不可知的用户，所以适用面窄，通常用于局域网中。