**MVC设计模型和Java三大构架**

为了降低成本及开发系统的复杂性，加快应用程序的开发和设计，面向企业应用的体系结构J2EE 平台，提供了一种基于组件的方法，对于一个多层的应用模型根据功能将其应用逻辑划分成组件。MVC 设计模式，最早是在Smalltalk －80中被提出的。MVC 即：模型(Model)、视图(View)、控制器(Controller)。模型表示业务逻辑；视图是应用程序中用户界面相关的部分；控制器的作用是从客户端接收请求，并且选择执行相应的业务逻辑，然后把响应结果送回到客户端。这种设计模式把应用逻辑、处理过程和显示逻辑分成不同的组件实现，通过三者之间的分离达到层间的松散耦合以及各层内部的紧密内聚，以实现代码重用和易于修改，以此来提高系统的灵活性、可重用性和可维护性，达到提高软件质量的目的。

**1 Struts2、Spring 和Hibernate 三大框架简介**

框架是JavaEE 开发领域炙手可热的一个名词。目前技术领域存在很多种框架技术，能够有效的解决JavaEE 应用开发中很多通用的问题，其中Struts2、Hibernate 以及Spring 是目前最主流的、使用最广泛的开源框架，每个框架专注于解决不同的问题，具有很强的代表性。其中，Struts2框架是一个MVC 框架；Hibernate 框架是一个ORM 框架；而Spring 框架是一个综合性的框架，是一个轻量级的解决方案。这三个框架往往各司其职，在应用中的不同层面发挥其作用。

**2 Struts2框架工作原理**

Struts2框架已经盛行多年，然而Struts2与Struts1之间并不是扩展和升级的关系。Struts2是著名框架WebWork 的扩展，WebWork 是一个很优秀的MVC 框架，但是由于是一个新兴的框架，在一段时间内没有被广泛使用。后来，Struts 和WebWork 社区决定将二者合并，于是推出了Struts2框架。Struts2框架兼具Struts1和WebWork 的优点，从而得到了广泛的使用。

Struts2的工作原理比较复杂：

1. 客户端向服务器端提交请求，容器初始化HttpServletRequest 请求对象。

2）请求对象被一系列的Servlet 过滤器过滤。Struts2中的过滤器有3种：

①ActionContextCleanUp 过滤器，是一个可选的过滤器，主要用来集成其他框架；②其他插件的核心过滤器，譬如：SiteMesh 插件的过滤器；③FilterDispatcher 过滤器，是Struts2 API中提供的过滤器，是必选的过滤器。

1. FilterDispatcher 过滤器调用ActionMapper ，决定该请求是否需要调用某个Action 。cute 方法，Action 调用业务逻 辑类完成业务逻辑。在调用 Action 的前后，将调用该 Action 涉及的相关拦截器。 8）Action 执行完毕后，ActionInvocation 根据 struts.xml 中的配置找到对应的返回结果，返 回给 JSP、FreeMarker 等模版文件。

**3 Hibernate 框架的作用及其对象状态**

Hibernate 是一个 ORM（Object-Relational Mapping）框架，主要作用是简化应用的数据持 久层编程，不需要编程人员花大量的时间编写 SQL 和 JDBC 代码。可见，Hibernate 框架位于 应用层和数据库之间，解决数据持久层编程。Hibernate 框架主要包括持久化对象（persistent Object）：是 Hibernate 框架中非常重要的组成部分，简称为 PO。PO 用来映射数据库中的记 录，可以通过修改 PO 来修改数据库记录；包括 Hibernate 属性文件（hibernate.properties）：使 用 Hibernate 进行数据持久层编程，相关的数据库访问信息需要在 Hibernate 属性文件中配置， 譬如，数据库驱动类、连接串、用户名、密码等，也可以使用名字为 hibernate.cfg.xml 的 xml 文件配置属性；还包括 Hibernate 映射文件（XML Mapping）：持久化对象映射数据库中的记 录，其映射关系依靠 Hibernate 框架的映射文件配置，映射文件是 XML 文件，往往使用 \*.hbm.xml 形式命名，其中\*是持久化对象的类名。 Hibernate 框架是一个完整的 ORM 框架，以对象为基础，通过操作对象，进一步操作对象 关联的数据库记录。持久化类与数据表对应，持久化的对象则映射数据库记录。持久化的对象 有三种状态：瞬时状态，当通过 new 操作符实例化了一个对象，而这个对象并没有被 Session 对象操作，也就是该对象没有与一个 Session 对象关联时，那么这个对象就称为瞬时状态对 象，瞬时状态的对象与普通对象没有区别，没有与数据库的记录有映射关系；持久状态，如果 一个对象与某一个 Session 对象关联，例如被 Session 对象刚加载的、刚保存的、刚更新的，那 么该对象就称为持久状态对象。持久状态的对象与数据库中一条数据相对应，并拥有持久化标 识。当持久状态的对象有改变时，当事务提交后，Hibernate 会自动检测到对象的变化，并持久 化到数据库中；脱管状态，当与持久状态对象关联的 Session 关闭后，该对象就变成脱管状 态。脱管状态的对象引用依然有效，可以继续使用，当某个 Session 关联后，脱管状态对象将转变为持久状态，脱管期间进行的修改将被持久化到数据库中。

**4 Spring 框架的模块**

企业级应用开发总是涉及方方面面，Spring 框架是一个轻量级的解决方案，致力于创建 “快装式企业应用”。Spring 框架包括一系列的特性，

被组织在七个模块中，可以把 Spring 框架 看成一个标准的开发组件。Spring 框架被设计成无侵入式的方式，即企业应用可以根据需要选 择 Spring 框架中必要的组件，而忽略其他部分，以做到最小范围依赖的 Spring 框架。 Spring 框架包含了 7 个模块，每个模块对于解决不同的问题。Spring Core 模块：是最核心 的模块，封装了 Spring 框架核心包，主要提供了 Spring IoC（控制反转）容器。IoC 是 Spring 框架的基础，所有其他特性都是基于 IoC 之上；Spring Context 模块：提供了对 Spring 中对象 的框架式访问方式，并包括国际化、事件传播等

特性；Spring DAO 模块：提供了 JDBC 的抽 象层，是集成 JDBC 的封装包，能够避免 JDBC 繁琐冗长代码，同时还提供了声明性事务管理 特性；Spring ORM 模块：提供了集成常用 ORM 框架的封装包，包括 JDO、JPA、Hibernate、 iBatis 等。使用该模块可以更为便捷的使用 ORM 框架，而且还可以同时使用 Spring 的其他特 性，如声明性事务等；Spring Web 模块：提供了 Web 开发的一些基础特性，如上传文件等， 同时提供了与 Web 框架集成的封装包，如集成 Struts2 框架；Spring AOP 模块：AOP（面向切 面编程）是 Spring 中除了 IoC 外的另外一个核心概念，该模块提供了符合 AOP 联盟规范的 AOP 实现，可以降低应用的耦合性，提高扩展性；Spring MVC 模块：该模块提供了一个 MVC 框架。Spring 框架包含了一系列的特性，可以根据应用的具体需要来选择使用部分组件。 Spring 框架可以在任何类型的应用中使用，譬如：桌面应用、Web 应用、C/S 应用等。

**2.1 Web开发框架Struts**

Struts2是一个基于MVC设计模式的Web应用框架，它本质上相当于一个servlet，在MVC设计模式中，Struts2作为控制器(Controller)来建立模型与视图的数据交互。Struts 2是Struts的下一代产品，是在 struts 1和WebWork的技术基础上进行了合并的全新的Struts 2框架。其全新的Struts 2的体系结构与Struts 1的体系结构差别巨大。Struts 2以WebWork为核心，采用拦截器的机制来处理用户的请求，这样的设计也使得业务逻辑控制器能够与ServletAPI完全脱离开，所以Struts 2可以理解为WebWork的更新产品。虽然从Struts 1到Struts 2有着太大的变化，但是相对于WebWork，Struts 2的变化很小。

它提供Controller 组件并整合了其它技术实现Model 和View 。对于Model,Struts 可以使用标准数据访问技术，如JDBC 和EJB ， 也可以使用第三方工具， 如Hibernate ，iBETIS ，O/R 桥。至于View ，Struts 支持JSP 技术，支持JSTL(JSP Standard Tag Liberary)和JSF(Java Server Faces)，也可以使用Velocity 模板、XSLT(the Extensible Stylesheet Langusge Transformations)及其他表示系统。此外，Struts 提供的自定义的标签库可以辅助JSP 进行表示层的设计。Struts 的核心组件有ActionServlet ，Action ，ActionForm ，ActionMapping ，ActionForward 及Taglib 配置文件等。

**2.2 对象关系映射工具Hibernate**

Hibernate是一个开放源代码的对象关系映射框架，它对JDBC进行了非常轻量级的对象封装，它将POJO与数据库表建立映射关系，是一个全自动的orm框架，hibernate可以自动生成SQL语句，自动执行，使得Java程序员可以随心所欲的使用对象编程思维来操纵数据库。 Hibernate可以应用在任何使用JDBC的场合，既可以在Java的客户端程序使用，也可以在Servlet/JSP的Web应用中使用，最具革命意义的是，Hibernate可以在应用EJB的J2EE架构中取代CMP，完成数据持久化的重任。

Hibernate是目前流行的ORM(Object Relation Mapping)框架。它对JDBC 进行了轻量级的对象封装，使得Java 程序员可以使用面向对象编程思维来操作后台关系数据库，而不用再去关心繁琐的SQL 语句。

Hibernate采用运行时反射(Runtime Reflection) 来决定一个类的持久化属性 。在企业级应用中，由于需要频繁地与数据库交互，开发者往往在应用和数据库之间建立一个“持久层”(Persistence Layer)，它负责应用到数据库的数据存储，也负责数据的检索和更新。在J2EE 的应用中，这个持久层使用JDBC 、实体Beans 、JDO 等技术来建立。但是JDBC 中数据访问对象(Databese Access Object,DAO)和SQL 语句直接绑定在一起降低了可维护性，且不支持继承和多态。Hibernate 的出现让使用者又多了一个选择，可以选用Hibernate 这种框架来实现对象、关系之间的映射和数据的持久化。Hibernate 拥有强大的查询语言(Hibernate Query Language,HQL) ，HQL 完全是面向对象的。

Hibernate的核心接口一共有5个，分别是：Session ，SessionFactory ，Configuration ，Transaction 和Query(或Criteria) 。通过这些接口，不仅可以对持久化对象进行存取，还能够进行事务控制。

**2.3 Web 开发框架Spring**

Spring是一个开放源代码的设计层面框架，他解决的是业务逻辑层和其他各层的松耦合问题，因此它将面向接口的编程思想贯穿整个系统应用。Spring是于2003 年兴起的一个轻量级的Java 开发框架，由Rod Johnson创建。简单来说，Spring是一个分层的JavaSE/EEfull-stack(一站式) 轻量级开源框架。

Spring也是一个优秀的Web开发框架。它解决了许多在J2EE开发中常见的问题的强大框架。它提供了管理业务对象的一致方法,并且鼓励对接口的编程而不是对类编程的习惯。Spring的架构基础是基于使用JavaBean属性的InversionofControl容器,数据访问架构集成了Hibernate和其他O/Rmapping解决方案,同时提供了对Struts的支持。它是一个分层的框架,由7个模块组成。Spring模块构建在核心容器之上,核心容器定义了创建、配置和管理Bean的方式。组成Spring框架的每个模块(或组件)都可以单独存在,或者与其他一个或多个模块联合实现。

**2.4　Struts,Hibernate和Spring对MVC模式的实现**

Struts和Spring都是当前流行的MVC实现框架,其中Struts在View上实现较为突出,Spring在业务层实现较好,Hibernate在数据持久层实现较好。因此将三者结合起来,很好地实现了MVC的设计模式。

在这种结构中,用户从浏览器发出请求,该请求首先将被Struts的ActionServlet接收,然后ActionServ2let会根据用户的Struts配置确定应触发哪一个具体的Action。Action的主要任务是调用某一个业务逻辑对象BO(BusinessObject)已完成相应的业务操作。由于在该框架中考虑引入了Spring,因此Action将不必直接创建具体的BO,而是通过Spring框架利用反向注入的原理(Ioc)来得到BO的实例。

**31 MVC架构**

Struts是一个不错的MVC架构，我一直以来都用它，通过简单的配置即可将view,controler,和Model结合起来。View主要以JSP来实现，因为它是面向标签的，所以对于网页设计人员提供了很好的接口。FormBean是介于JSP和Action之间的中间数据载体，它肩负着数据从JSP到ACTION的传递过程。Action是流程的中转站，不同的业务在不同的Action中以不同的Model调用来实现。Model就是实现具体业务的操作过程，不过这种过程是一种在较高水平上的实现。 总之，MVC架构实现了三层结构中的两层，即表现层和业务层，另外还有一层被称之为持久化层。

**3.2、三层架构**

三层架构即“表现层”，“业务层”，“持久化层”。表现层实现的代表作品是Struts框架，业务层实现的代表作品是Spring，持久层实现的代表作品是Hibernate。