struts 控制用的   
  
hibernate 操作数据库的   
  
spring 用解耦的   
  
详细的说：   
  
STRUTS 在 SSH 框架中起控制的作用 , 其核心是 Controller, 即 ActionServlet, 而 ActionServlet 的核心就是 Struts-confi g.xml. 主要控制逻辑关系的处理 .   
  
hibernate 是数据持久化层 , 是一种新的对象、关系的映射工具 , 提供了从 Java 类到数据表的映射，也提供了数据查询和恢复等机制 , 大大减少数据访问的复杂度。把对数据库的直接操作 , 转换为对持久对象的操作 .   
  
SPRING 是一个轻量级的控制反转 (IoC) 和面向切面 (AOP) 的容器框架 , 面向接口的编程 , 由容器控制程序之间的（依赖）关系，而非传统实现中，由程序代码直接操控。这也就是所谓 “ 控制反转 ” 的概念所在：（依赖）控制权由应用代码中转到了外部容器，控制权的转移，是所谓反转。依赖注入，即组件之间的依赖关系由容器在运行期决定，形象的来说，即由容器动态的将某种依赖关系注入到组件之中   
起到的主要作用是解耦   
  
  
Struts 、 spring 、 Hibernate 在各层的作用   
  
1 ） struts 负责 web 层 .   
  
ActionFormBean 接收网页中表单提交的数据，然后通过 Action 进行处理，再 Forward 到对应的网页。   
  
在 struts-config.xml 中定义 <action-mapping>, ActionServlet 会加载。   
  
2 ） spring 负责业务层管理，即 Service （或 Manager).   
  
1 ． service 为 action 提供统计的调用接口，封装持久层的 DAO.   
  
2 ．可以写一些自己的业务方法。   
  
3 ．统一的 javabean 管理方法   
  
4 ．声明式事务管理   
  
5. 集成 Hiberante   
  
3 ） Hiberante ，负责持久化层，完成数据库的 crud 操作   
  
hibernate 为持久层，提供 OR/Mapping 。   
  
它有一组 .hbm.xml 文件和 POJO, 是跟数据库中的表相对应的。然后定义 DAO ，这些是跟数据库打交道的类，它们会使用 PO 。   
  
在 struts+spring+hibernate 的系统中，   
  
对象的调用流程是： jsp-> Action － > Service ->DAO ->Hibernate 。   
  
数据的流向是 ActionFormBean 接受用户的数据， Action 将数据从 ActionFromBean 中取出，封装成 VO 或 PO,   
  
再调用业务层的 Bean 类，完成各种业务处理后再 forward 。而业务层 Bean 收到这个 PO 对象之后，会调用 DAO 接口方法，进行持久化操作。