**初识J2EE**

**一.简介**

J2EE（Java 2 Platform, Enterprise Edition）是一个为大企业主机级的计算类型而设计的Java平台。Sun微系统（与其工业伙伴一起，例如IBM）设计了J2EE，以此来简化在受客户级环境下的应用开发。由于创造了标准的可重用模块组件以及由于构建出能自动处理编程中多方面问题的等级结构，J2EE简化了应用程序的开发，也降低了对编程和对受训的程序员的要求。

J2EE是一套全然不同于传统应用开发的技术架构，包含许多组件，主要可简化且规范应用系统的开发与部署，进而提高可移植性、安全与再用价值。

J2EE核心是一组技术规范与指南，其中所包含的各类组件、服务架构及技术层次，均有共同的标准及规格，让各种依循J2EE架构的不同平台之间，存在良好的兼容性，解决过去企业后端使用的信息产品彼此之间无法兼容，企业内部或外部难以互通的窘境。

J2EE组件和“标准的” [Java](https://baike.baidu.com/item/Java/85979)类的不同点在于：它被装配在一个J2EE应用中，具有固定的格式并遵守J2EE规范，由J2EE服务器对其进行管理。J2EE规范是这样定义J2EE组件的：客户端应用程序和applet是运行在客户端的组件；Java Servlet和Java Server Pages (JSP) 是运行在服务器端的Web组件；Enterprise Java Bean (EJB )组件是运行在服务器端的业务组件。

**二.优势**

1. 保留现存的IT资产

2. 高效的开发

3. 支持异构环境

4. 可伸缩性

5. 稳定的可用性

**三.四层模型**

J2EE使用多层的[分布式应用](https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E5%BA%94%E7%94%A8)模型，[应用逻辑](https://baike.baidu.com/item/%E5%BA%94%E7%94%A8%E9%80%BB%E8%BE%91)按功能划分为组件，各个应用组件根据他们所在的层分布在不同的机器上。现在J2EE 的多层企业级应用模型将两层化模型中的不同层面切分成许多层。一个多层化应用能够为不同的每种服务提供一个独立的层，以下是 J2EE 典型的四层结构：

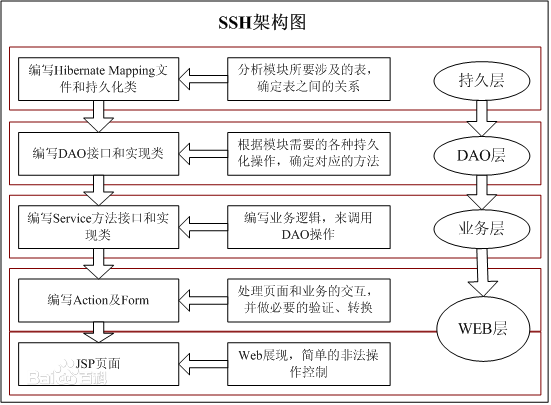
* 运行在客户端机器上的客户层组件
* 运行在J2EE[服务器](https://baike.baidu.com/item/%E6%9C%8D%E5%8A%A1%E5%99%A8)上的Web层组件
* 运行在J2EE服务器上的[业务逻辑层](https://baike.baidu.com/item/%E4%B8%9A%E5%8A%A1%E9%80%BB%E8%BE%91%E5%B1%82" \t "_blank)组件
* 运行在EIS服务器上的[企业信息系统](https://baike.baidu.com/item/%E4%BC%81%E4%B8%9A%E4%BF%A1%E6%81%AF%E7%B3%BB%E7%BB%9F" \t "_blank)层软件

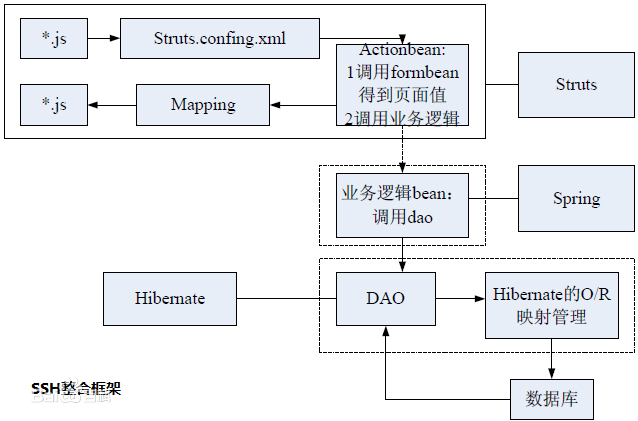
**四.三大框架（SSH框架）**

SHH不是一个框架，而是多个框架（**struts+spring+hibernate**）的集成，是目前较流行的一种Web应用程序开源集成框架，用于构建灵活、易于扩展的多层Web应用程序。

集成SSH框架的系统从职责上分为四层：**表示层、业务逻辑层、数据持久层和域模块层（实体层）**。

Struts作为系统的整体基础架构，负责MVC的分离，在Struts框架的模型部分，控制业务跳转，利用Hibernate框架对持久层提供支持。Spring一方面作为一个轻量级的IoC容器，负责查找、定位、创建和管理对象及对象之间的依赖关系，另一方面能使Struts和Hibernate更好地工作。





由SSH构建系统的基本业务流程是：

1、在表示层中，首先通过JSP页面实现交互界面，负责传送请求(Request)和接收响应(Response)，然后Struts根据配置文件(struts-config.xml)将ActionServlet接收到的Request委派给相应的Action处理。

2、在业务层中，管理服务组件的Spring IoC容器负责向Action提供业务模型(Model)组件和该组件的协作对象数据处理(DAO)组件完成业务逻辑，并提供事务处理、缓冲池等容器组件以提升系统性能和保证数据的完整性。

3、在持久层中，则依赖于Hibernate的对象化映射和数据库交互，处理DAO组件请求的数据，并返回处理结果。

采用上述开发模型，不仅实现了视图、控制器与模型的彻底分离，而且还实现了业务逻辑层与持久层的分离。这样无论前端如何变化，模型层只需很少的改动，并且数据库的变化也不会对前端有所影响，大大提高了系统的可复用性。而且由于不同层之间耦合度小，有利于团队成员并行工作，大大提高了开发效率。

**五.MVC架构**

[MVC](https://baike.baidu.com/item/MVC)全名是Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，是[Xerox PARC](https://baike.baidu.com/item/Xerox%20PARC)在二十世纪八十年代为编程语言Smalltalk－80发明的一种[软件设计模式](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E8%AE%BE%E8%AE%A1%E6%A8%A1%E5%BC%8F)，已被广泛使用。后来被推荐为[Oracle](https://baike.baidu.com/item/Oracle)旗下[Sun](https://baike.baidu.com/item/Sun)公司[Java EE](https://baike.baidu.com/item/Java%20EE)平台的设计模式，并且受到越来越多的使用[ColdFusion](https://baike.baidu.com/item/ColdFusion)和[PHP](https://baike.baidu.com/item/PHP)的开发者的欢迎。模型－视图－控制器模式是一个有用的[工具箱](https://baike.baidu.com/item/%E5%B7%A5%E5%85%B7%E7%AE%B1)，一种软件设计典范，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被独特的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。

**六.MVC架构与SSH框架的关系**

MVC三层架构：模型层，控制层和视图层。

模型层，用Hibernate框架让来JavaBean在数据库生成表及关联，通过对JavaBean的操作来 对数据库进行操作；

控制层，用Struts框架来连接数据层和视图层的，接收、处理、发送数据并控制流程；

视图层，用JSP模板把页面展现给用户以及提供 与用户的交互。而Spring框架粘合了Hibernate和Struts，透明的管理了整个架构，提供IOC容器使代码松耦合以及AOP框架的切面功能 等等。

**学号：201502056**

**信管11501**

**霍雅馨**