

- **技术全面**：涵盖Java Web开发的主流技术和主流框架；
- **工具详尽**：包括JDK、Tomcat、Eclipse、MySQL、Log4j、Ant、JUnit、CVS、解析XML文件等；
- **由浅入深**：每章从简单示例入手，最后给出一个详细示例加深理解；
- **实战性强**：书中提供了大量典型的应用案例，实战性很强；
- **示例丰富**：提供了大量高质量和从实际项目总结的示例，有较高的复用性；
- **项目经典**：通过项目的经典开发流程来讲解项目，并对代码进行了详细注释。

搞定 J2EE

Struts+Spring+Hibernate 整合详解与典型案例

杜聚宾 编著

从现在开始, 让我们踏上J2EE的成功之路吧!

搞定J2EE

Struts+Spring+Hibernate

整合详解与典型案例

杜聚宾 编著

电子工业出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书是笔者在多年项目开发过程中的经验总结,它通过丰富的实例由浅入深、循序渐进地介绍了 J2EE 的体系结构和如何利用开源框架 Struts、Spring 和 Hibernate 进行 J2EE 程序的开发,从而帮助软件开发人员快速掌握 J2EE 的设计思想,并能将其应用于实战中。

通过本书,读者可以了解到 J2EE 的基本思想,熟练掌握如何进行 J2EE 程序的开发。全书共分 16 章。第 1 章简要介绍了 J2EE 的来源、整体框架及其核心技术。第 2 章对 MVC 模式的设计思想、规范及其优缺点进行了讲解。第 3 章至第 9 章对 JDK, Tomcat, Eclipse, Log4j, Ant, JUnit 和 CVS 进行了介绍,目的是使读者能快速地建立 J2EE 的开发环境,迅速掌握在开发过程中经常要使用到的一些开源工具的使用方法。第 10 章至第 12 章主要介绍如何使用 Spring、Struts 和 Hibernate 来进行 J2EE 程序的开发,并对它们之间的整合方式进行讲解。第 13 章至第 16 章主要通过网上调查系统、内容管理系统、在线考试系统和信息发布系统这 4 个实例,从可行性分析开始,按照软件开发的过程,深入地讲解软件开发的步骤,使读者达到实战的要求。

本书既有理论基础又讲解了实践案例,而且基本概括了进行 J2EE 开发所要用到的所有开发工具和软件,目的就是使读者跟着本书的讲解学习完之后,能熟练地将其应用到实战中。

本书适用于初、中、高级软件开发人员,同时也可用做高校相关专业师生和社会培训班的教材。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

搞定 J2EE: Struts+Spring+Hibernate 整合详解与典型案例 / 杜聚宾编著. —北京: 电子工业出版社, 2012.6
ISBN 978-7-121-16964-9

I. ①搞… II. ①杜… III. ①JAVA 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 088210 号

策划编辑: 胡辛征

责任编辑: 徐津平

文字编辑: 王 静

印 刷: 北京中新伟业印刷有限公司

装 订: 三河市皇庄路通装订厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 41.25 字数: 1056 千字

印 次: 2012 年 6 月第 1 次印刷

印 数: 4000 册 定价: 79.00 元(含 DVD 光盘 1 张)

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

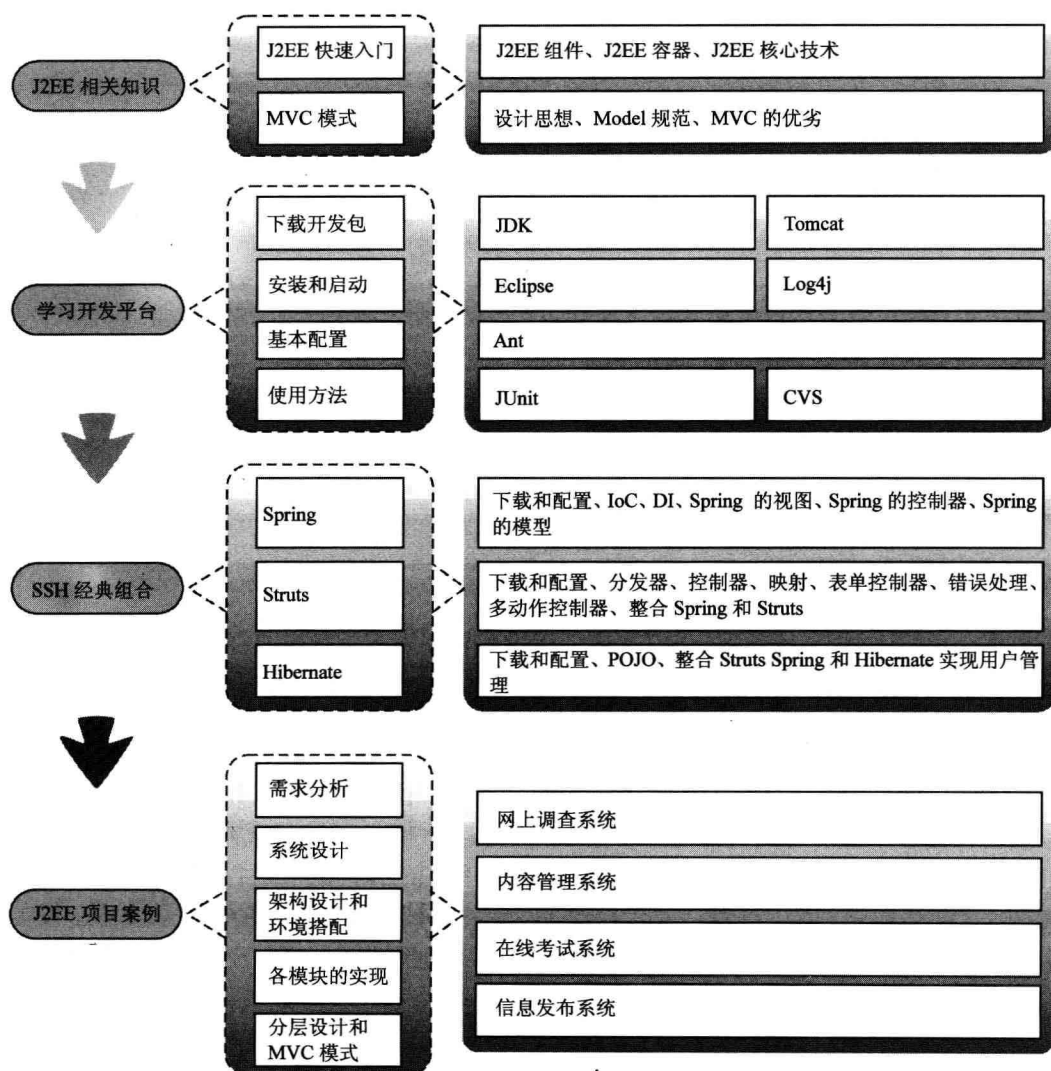
质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前言：作者的提示

作为一本全面介绍 J2EE 的书籍，本书可以帮助读者搞定 Spring,Struts,Hibernate，还可以搞定 Tomcat,Eclipse,Log4j,Ant,JUnit,CVS 等众多开发工具。

本书内容



本书的优势

- **配有学习论坛** <http://www.rzchina.net>: 读者可以在上面讨论技术问题, 笔者会及时回答读者提问, 并提供各种技术文章, 帮助读者提高开发水平。
- **由浅入深**: 让读者能很快地进行 J2EE 开发。每一章都从一个简单的示例入手, 让读者快速了解本章内容, 然后再详细讲解本章涉及的基本原理和知识点。最后, 再通过一个详细的示例来巩固学习成果, 这样由浅入深符合读者的接受过程。
- **内容广泛**: 本书不但对 J2EE 和 MVC 的相关理论进行了介绍, 还对目前市场上众多的开源项目进行了介绍, 涵盖了进行 J2EE 开发所需要的所有工具和软件, 比如, JDK, Tomcat, Eclipse, Log4j, Ant, Junit, CVS, Struts, Spring 和 Hibernate, 目的是使读者通过阅读本书, 即可掌握开发 J2EE 程序所需要的相关工具和使用方法。
- **实战性强**: 本书在对 J2EE 的理论思想和相关工具进行了讲解之后, 又通过几个目前实际应用中使用比较多的实例对 J2EE 的开发方法进行了详细的讲解, 从而使读者可以通过本书的学习快速进入实战项目的开发。
- **提纲挈领**: 本书每章内容前面都有本章所要讲解内容的简介, 读者可以宏观掌握本章的内容, 进而理清思路, 快速地掌握本章的具体内容。

读者需要具备的基础

阅读本书时, 读者应该具有 Java 语言的基础知识, 并且对 J2EE 有一个初步的了解。除此之外, 本书假定读者不再需要掌握其他知识。当然, 如果你开发过 JSP, ASP 或 PHP 程序, 你将能够更好地理解本书所讲述的内容。

另外, 本书中的实例都是在 Windows 平台上进行讲解的, 但所讲述的内容也可以在 UNIX 平台上使用。

统一思想——让我们在这些方面达成共识

• 有效而愉快地学习

目前, 市场上有关 J2EE 开发的书籍种类繁多, 但很多书籍要么只讲理论, 要么只讲如何使用软件, 或者大篇幅地讲一些在实际项目中用不到的知识, 使读者无所适从, 掌握不到学习 J2EE 的真正要领, 也导致初学者对 J2EE 产生畏惧心理。其实 J2EE 并不难掌握。

为了帮助众多初学者快速掌握 J2EE 的开发方法, 笔者精心编著了本书。它是笔者在多年项目实践中的经验总结。本书根据读者的学习规律, 首先通过实例介绍基本概念和基本操作, 然后在读者掌握了这些基本概念和操作的基础上, 再对内容进行深入的讲解, 严格遵循由浅入深、循序渐进的原则。本书按照掌握 J2EE 知识的先后顺序进行编排。本书对于每一个实例, 从环境配置开始, 到最后的运行都有详尽的介绍, 从而使读者很容易就能运行实例, 掌握开发方法, 并体会到学习的快乐, 不断增强学习的动力。

• 用到什么介绍什么

本书从头到尾都是按照读者的学习兴趣来安排知识点的。虽然讲解了开发 J2EE 项目所

要具备的几乎所有的工具，但并没有针对某个工具讲解它详细的命令和操作，而只是针对开发人员在日常开发 J2EE 项目时所要用到的操作进行讲解，目的只有一个，就是使读者能够快速掌握开发 J2EE 程序的技能。

光盘说明

本书中所有的源代码都附在随书光盘中，源代码以章节分类。只要按照本书的介绍将 J2EE 的开发环境配置好，直接将每章的源代码放在 Eclipse 相应的目录下，启动 Tomcat，即可运行。书中的章节和代码对照表参见光盘中的 `readme.txt` 文档。

本书适合的读者

本书是一本关于 J2EE 开发实践的技术书籍。还不了解如何进行 J2EE 开发的初学者，以及 ERP 开发、电子商务、网站开发的爱好者，都可以从本书中了解到开发实践所需的知识和思路。

本书以“实战性强”为指导思想，书中介绍的所有知识，无论是开发 J2EE 所需要的理论知识，所需要的环境配置、开源软件，还是所介绍的 4 个实例，无一例外都是以突出“实战性强”为出发点的。所以，如果你想拥有使用 J2EE 进行项目开发的实践经验，那么，本书可以使你的学习事半功倍。在读完本书后，你不但能熟练掌握开发 J2EE 所涉及的相关工具，而且能够马上进行 J2EE 项目的实战开发。

如果你原来读过有关 J2EE 纯理论性的书籍，或者读过虽然是介绍 J2EE 实例，但那些实例都不适用于实际应用的书籍，那么，当你读完本书后，你的感受将会截然不同。在本书中，你将不会看到纯理论性的介绍，而是会看到涉及 J2EE 的一些工具和软件是如何来实现这些理论规范的。

特别要说明的是，本书尤其适合作为引路指南。因为它详细地介绍了 J2EE 的初学者入门时所必须掌握的基本知识和工具的应用，并且对每个应用都采用实例的方式，帮助读者一步一步实现实例，让初学者在不知不觉中既理解了 J2EE 的理论知识，又掌握了工具的使用方法。在学习过程中，如果只是为了满足日常的实践应用，那么读者可以不用再参考其他书籍，只通过本书就可以掌握 J2EE 相关工具和软件的使用，以及实战项目的开发方法。

编 者

目 录

第 1 章 J2EE 快速入门 15

1.1 J2EE 概述	16
1.1.1 J2EE 的来源	16
1.1.2 J2EE 整体框架	16
1.1.3 从 J2EE 到 JavaEE	17
1.2 J2EE 组件	18
1.2.1 客户端组件	19
1.2.2 Web 组件	19
1.2.3 业务逻辑组件	19
1.3 J2EE 容器	19
1.3.1 容器服务	19
1.3.2 容器类型	20
1.4 J2EE 核心技术	20
1.4.1 Servlet	21
1.4.2 JSP (Java 服务页面)	22
1.4.3 EJB (企业 JavaBean)	23
1.4.4 JDBC (Java 数据库连接)	23
1.4.5 JTA/JTS (Java 事务)	24
1.4.6 JNDI (Java 命名和目录服务)	24
1.4.7 JavaMail (Java 邮件服务)	25
1.4.8 RMI (远程方法调用)	25
1.4.9 JMS (Java 消息服务)	26
1.4.10 JMX (Java 分布式管理)	26
1.4.11 JACC (Java 容器授权合同)	26
1.4.12 JCA (Java 连接器体系)	26
1.5 小结	27

第 2 章 使用 MVC 更好地实现 J2EE 28

2.1 MVC 模式概述	29
--------------	----

2.1.1 MVC 模式的设计思想	29
2.1.2 MVC 模式的处理过程	30
2.2 Model 规范概述	30
2.2.1 Model1 简介	31
2.2.2 Model2 简介	31
2.3 使用 MVC 的优劣	32
2.3.1 使用 MVC 模式的优势	32
2.3.2 使用 MVC 模式的劣势	32
2.4 小结	32

第 3 章 建立 Java 的开发平台 33

3.1 建立 Java 的开发环境	34
3.1.1 下载 JDK	34
3.1.2 安装 JDK	34
3.1.3 设定 Path, CLASSPATH 和 JAVA_HOME	36
3.2 验证 JDK 是否安装成功	37
3.3 建立 J2EE 的开发环境	38
3.3.1 下载 SDK	38
3.3.2 安装 SDK	39
3.3.3 设定 Path, CLASSPATH 和 J2EE_HOME	42
3.4 小结	42

第 4 章 Tomcat 使用指南 43

4.1 Tomcat 简介	44
4.1.1 Tomcat 的目录结构	44
4.1.2 Tomcat 的配置参数	44
4.2 建立 Tomcat 的开发环境	47

4.2.1 下载 Tomcat	48
4.2.2 设定 TOMCAT_HOME	49
4.3 验证 Tomcat 是否安装成功	49
4.4 创建和发布 Web 应用程序	52
4.4.1 创建和发布 JSP 应用程序	52
4.4.2 创建和发布 Servlet 应用程序	55
4.5 小结	57

第 5 章 Eclipse 使用指南 58

5.1 Eclipse 简介	59
5.1.1 Eclipse 的历史	59
5.1.2 Eclipse 的运行机制	59
5.2 建立 Eclipse 的开发环境	59
5.2.1 下载 Eclipse	59
5.2.2 配置 Eclipse	60
5.3 整合 Eclipse 和 Tomcat	60
5.3.1 下载 Eclipse 的 Tomcat 插件	61
5.3.2 为 Eclipse 配置 Tomcat 插件	61
5.4 使用 Eclipse 建立 Web 开发项目	62
5.5 Eclipse 的常用快捷键	66
5.5.1 有关格式化的快捷键	66
5.5.2 有关调试的快捷键	67
5.5.3 有关重构的快捷键	67
5.6 小结	67

第 6 章 Log4j 使用指南 68

6.1 Log4j 介绍	69
6.1.1 Log4j 历史	69
6.1.2 Log4j 组成	69
6.2 建立 Log4j 的开发环境	70
6.2.1 下载 Log4j	70
6.2.2 配置 Log4j	71
6.3 Log4j 的使用方法	71
6.3.1 配置 Log4j	71
6.3.2 配置根 Logger	72
6.3.3 指定日志输出位置	73
6.3.4 指定日志输出格式	74
6.3.5 指定日志输出优先级	75

6.3.6 在代码中使用 Log4j	75
6.4 改进 Log4j	75
6.5 小结	77

第 7 章 Ant 使用指南 78

7.1 Ant 介绍	79
7.1.1 Ant 简介	79
7.1.2 为什么要使用 Ant	79
7.2 建立 Ant 的开发环境	80
7.2.1 下载 Ant	80
7.2.2 配置 Ant	80
7.3 Ant 的使用方法	82
7.3.1 Ant 能完成的工作	82
7.3.2 配置文件 build.xml	82
7.3.3 编译源代码	84
7.3.4 创建 JAR 文件	85
7.4 小结	85

第 8 章 JUnit 使用指南 86

8.1 JUnit 介绍	87
8.1.1 JUnit 简介	87
8.1.2 为什么要使用 JUnit	87
8.2 建立 JUnit 的开发环境	88
8.2.1 下载 JUnit	88
8.2.2 配置 JUnit	88
8.3 JUnit 的使用方法	90
8.3.1 继承 TestCase	90
8.3.2 编写测试方法	93
8.3.3 编写断言	93
8.4 JUnit 的新特性	98
8.4.1 改变测试方法的命名方式	98
8.4.2 不再继承 TestCase	99
8.4.3 改变初始化和销毁方式	99
8.4.4 改变异常处理的方式	101
8.5 小结	103

第 9 章 CVS 使用指南 104

9.1 CVS 介绍	105
------------------	-----

9.1.1	CVS 简介	105
9.1.2	为什么要使用 CVS	105
9.2	建立 CVS 的开发环境	105
9.2.1	下载 CVS	105
9.2.2	配置 CVS	105
9.3	CVS 的使用方法	109
9.3.1	添加 CVS 资源库	109
9.3.2	提交和更新代码	112
9.4	小结	114

第 10 章 使用 Spring 快速实现 Web 开发 115

10.1	Spring 介绍	116
10.1.1	Spring 简介	116
10.1.2	下载 Spring	117
10.1.3	配置 Spring	118
10.2	Spring 核心思想	119
10.2.1	反向控制 (IoC)	119
10.2.2	依赖注入 (DI)	123
10.3	利用 Spring 在 JSP 页面输出 “HelloWorld” 的示例	125
10.3.1	建立 myHelloWorld 工程	125
10.3.2	编写 JSP 页面 helloWorld.jsp	129
10.3.3	编写控制器 HelloWorldController.java	129
10.3.4	建立 Spring 的配置文档 dispatcherServlet-servlet.xml	130
10.3.5	配置 web.xml	131
10.3.6	启动 Tomcat 运行程序	132
10.4	Spring 的视图 (View)	132
10.4.1	视图简介	132
10.4.2	视图解析	136
10.5	Spring 的控制器 (Controller)	141
10.5.1	Controller 接口	141
10.5.2	命令控制器 (BaseCommandController)	144
10.5.3	表单控制器 (SimpleFormController)	151

10.5.4	多动作控制器 (MultiActionController)	155
10.6	Spring 的视图映射机制	168
10.6.1	分发器 (DispatcherServlet)	168
10.6.2	视图映射	184
10.7	Spring 的模型 (Model)	186
10.7.1	模型简介	186
10.7.2	模型封装	188
10.8	小结	189

第 11 章 使用 Struts 2 快速实现 Web 开发 190

11.1	Struts 2 介绍	191
11.2	Struts 2 的环境配置	191
11.2.1	下载 Struts 2	191
11.2.2	搭建 Struts 2 的开发运行环境	192
11.3	Struts 2 入门示例	195
11.3.1	编写实现消息输出的 HelloWorld.jsp 页面	195
11.3.2	编写控制器 HelloWorld.java	196
11.3.3	配置 struts.xml 和 struts.properties 文件	197
11.3.4	配置 web.xml	198
11.3.5	测试运行该示例	199
11.4	Struts 2 的核心机制	200
11.4.1	FilterDispatcher 处理流程	200
11.4.2	Struts 2 的输入校验	203
11.4.3	Struts 2 的拦截机制	205
11.4.4	Struts 2 的国际化	209
11.4.5	Struts 2 的标签库	211
11.5	Struts 2 的配置文件	212
11.5.1	struts.properties	212
11.5.2	struts.xml	214
11.6	小结	215

第 12 章 使用 Hibernate 快速实现持久层处理 216

12.1	Hibernate 介绍	217
------	--------------------	-----

第 13 章 网上调查系统 288

13.5.6 填写调查的页面 survey.jsp.....	338	13.8.9 设定调查题目实现类	
13.5.7 查看调查结果的页面		SetSurveyImpl.java	373
surveyOK.jsp.....	339	13.8.10 设定调查项目实现类	
13.6 根据数据库表生成 XML 和 POJO.....	340	SetSurveyOptionImpl.java	374
13.6.1 生成对应的 XML.....	340	13.8.11 设定调查子项目实现类	
13.6.2 生成 POJO.....	354	SetSurveyChildImpl.java	375
13.7 编写网上调查系统的 DAO.....	362	13.8.12 统计调查结果实现类	
13.7.1 用户 DAO 接口 UserDao.java.....	362	SetSurveyChildCountImpl.java	376
13.7.2 调查题目 DAO 接口		13.9 编写网上调查系统的控制器类.....	376
SurveyDAO.java.....	363	13.9.1 登录控制器类 LoginAction.java.....	377
13.7.3 调查项目 DAO 接口		13.9.2 注册控制器类 RegeditAction.java.....	378
SurveyOptionDAO.java.....	363	13.9.3 设定调查题目控制器类	
13.7.4 调查子项目 DAO 接口		SurveyAction.java.....	379
SurveyChildDAO.java.....	363	13.9.4 设定调查项目控制器类	
13.7.5 调查结果 DAO 接口		SetSurveyAction.java	380
SurveyChildCountDAO.java.....	364	13.10 编写配置文件	
13.7.6 用户 DAO 实现类		dispatcherServlet-servlet.xml	389
UserDAOImpl.java	364	13.11 运行验证程序	395
13.7.7 调查题目 DAO 实现类		13.12 小结.....	399
SurveyDAOImpl.java.....	365		
13.7.8 调查项目 DAO 实现类		第 14 章 内容管理系统	400
SurveyOptionDAOImpl.java.....	366		
13.7.9 调查子项目 DAO 实现类		14.1 系统概述	401
SurveyChildDAOImpl.java.....	367	14.2 需求分析	402
13.7.10 调查结果 DAO 实现类		14.2.1 系统用例图	402
SurveyChildCountDAOImpl.java.....	368	14.2.2 需求规格说明书	402
13.8 编写网上调查系统的业务逻辑类.....	369	14.3 系统设计	406
13.8.1 注册接口 Regedit.java3	369	14.3.1 设计视图层.....	406
13.8.2 登录接口 Login.java.....	369	14.3.2 设计持久层.....	412
13.8.3 设定调查题目接口 SetSurvey.java.....	369	14.3.3 设计业务逻辑层	419
13.8.4 设定调查项目接口		14.3.4 设计控制层.....	419
SetSurveyOption.java.....	370	14.4 架构设计和环境搭配	420
13.8.5 设定调查子项目接口		14.4.1 内容管理系统采用的架构.....	420
SetSurveyChild.java.....	370	14.4.2 在 Eclipse 下建立项目 myEdition	420
13.8.6 统计调查结果接口		14.4.3 编写本项目的 Ant build 文件	424
SetSurveyChildCount.java	371	14.4.4 配置本项目的 web.xml 文件	425
13.8.7 注册实现类 RegeditImpl.java.....	371	14.5 编写内容管理系统的 JSP 页面.....	426
13.8.8 登录实现类 LoginImpl.java	372	14.5.1 用户注册页面 regedit.jsp	426
		14.5.2 用户登录页面 login.jsp.....	427

14.5.3	设定内容类别的页面 type.jsp	428
14.5.4	发布内容页面 edit.jsp	430
14.5.5	浏览内容页面 index.jsp	432
14.6	根据数据库表生成 XML 和 POJO	433
14.6.1	生成对应的 XML	433
14.6.2	生成 POJO	444
14.7	编写内容管理系统的 DAO	450
14.7.1	用户 DAO 接口 UserDao.java	450
14.7.2	内容类别 DAO 接口 ContentTypeDAO.java	450
14.7.3	内容 DAO 接口 ContentDAO.java	450
14.7.4	用户 DAO 实现类 UserDAOImpl.java	451
14.7.5	内容类别 DAO 实现类 ContentTypeDAOImpl.java	452
14.7.6	内容 DAO 实现类 ContentDAOImpl.java	453
14.8	编写内容管理系统的业务逻辑类	454
14.8.1	注册接口 Regedit.java	454
14.8.2	登录接口 Login.java	455
14.8.3	设定内容类别的接口 SetContentType.java	455
14.8.4	发布接口 Edit.java	455
14.8.5	注册实现类 RegeditImpl.java	456
14.8.6	登录实现类 LoginImpl.java	457
14.8.7	设定内容类别的实现类 SetContentTypeImpl.java	457
14.8.8	发布实现类 EditImpl.java	458
14.9	编写内容管理系统的控制器类	460
14.9.1	登录控制器类 LoginAction.java	460
14.9.2	注册控制器类 RegeditAction.java	461
14.9.3	设定内容类别的控制器类 ContentTypeAction.java	462
14.9.4	发布控制器类 EditAction.java	463
14.9.5	内容控制器类 ContentAction.java	465
14.9.6	浏览内容的控制器类 IndexAction.java	466
14.10	编写 Struts 的配置文件 struts.xml	467

14.11	编写 Spring 和 Hibernate 的配置文件 applicationContext.xml	468
14.12	解决 Struts 的中文乱码问题	469
14.13	运行验证程序	469
14.14	小结	472

第 15 章 在线考试系统 473

15.1	系统概述	474
15.2	需求分析	475
15.2.1	系统用例图	475
15.2.2	需求规格说明书	475
15.3	系统设计	480
15.3.1	设计视图层	480
15.3.2	设计持久层	490
15.3.3	设计业务逻辑层	500
15.3.4	设计控制层	501
15.4	架构设计和环境搭配	501
15.4.1	在线考试系统采用的架构	502
15.4.2	在 Eclipse 下建立项目 myExam	502
15.4.3	编写本项目的 Ant build 文件	505
15.4.4	配置本项目的 web.xml 文件	506
15.5	编写在线考试系统的 JSP 页面	507
15.5.1	用户注册页面 regedit.jsp	507
15.5.2	用户登录页面 login.jsp	508
15.5.3	设定试题种类的页面 shitiType.jsp	509
15.5.4	设定试题基本信息的页面 shiti.jsp	510
15.5.5	设定试题项目的页面 shitiOption.jsp	512
15.5.6	组织试卷的页面 examInfo.jsp	513
15.5.7	参加考试的页面 exam.jsp	515
15.5.8	考试成绩的页面 examChengji.jsp	516
15.6	根据数据库表生成 XML 和 POJO	517
15.6.1	生成对应的 XML	517
15.6.2	生成 POJO	530
15.7	编写在线考试系统的 DAO	539
15.7.1	用户 DAO 接口 UserDao.java	539
15.7.2	试题种类 DAO 接口 ShitiTypeDAO.java	539

15.7.3	试题题目 DAO 接口		15.9.4	设定试题题目控制器类	
	ShitiDAO.java	540		ShitiAction.java	557
15.7.4	试题项目 DAO 接口		15.9.5	设定试题项目控制器类	
	ShitiOptionDAO.java	540		ShitiOptionAction.java	559
15.7.5	试卷 DAO 接口 ExamDAO.java	540	15.9.6	试卷控制器类 ExamAction.java	561
15.7.6	用户 DAO 实现类		15.10	编写配置文件	
	UserDAOImpl.java	541		dispatcherServlet-servlet.xml	565
15.7.7	试题种类 DAO 实现类		15.11	解决 Spring 的中文乱码问题	571
	ShitiTypeDAOImpl.java	542	15.12	运行验证程序	573
15.7.8	试题题目 DAO 实现类		15.13	小结	577
	ShitiDAOImpl.java	543	第 16 章	信息发布系统	578
15.7.9	试题项目 DAO 实现类		16.1	系统概述	579
	ShitiOptionDAOImpl.java	544	16.2	需求分析	579
15.7.10	试卷 DAO 实现类		16.2.1	系统功能模块划分	579
	ExamDAOImpl.java	545	16.2.2	系统流程分析	579
15.8	编写在线考试系统的业务逻辑类	546	16.2.3	系统设计	580
15.8.1	注册接口 Regedit.java	546	16.3	建立项目	581
15.8.2	登录接口 Login.java	546	16.3.1	在 MyEclipse 下建立项目	581
15.8.3	设定试题种类接口		16.3.2	编写 Ant 文件	584
	SetShitiType.java	546	16.3.3	配置 Web.xml 文件	585
15.8.4	设定试题题目接口 SetShiti.java	547	16.4	建立数据库表	585
15.8.5	设定试题项目接口		16.4.1	存放用户信息的数据库表	586
	SetShitiOption.java	547	16.4.2	存放信息的数据库表	587
15.8.6	设定试卷接口 SetExam.java	548	16.4.3	存放信息类别的数据库表	589
15.8.7	注册实现类 RegeditImpl.java	548	16.4.4	存放用户权限的数据库表	590
15.8.8	登录实现类 LoginImpl.java	549	16.4.5	建立表之间的关系	592
15.8.9	设定试题种类实现类		16.5	设计信息发布系统	594
	SetShitiTypeImpl.java	550	16.5.1	生成对应的 Hibernate 映射文件	594
15.8.10	设定试题题目实现类		16.5.2	生成持久化 POJO 文件	604
	SetShitiImpl.java	551	16.5.3	设计值对象类 VO 对象	612
15.8.11	设定试题项目实现类		16.5.4	信息发布系统在持久层的整体	
	SetShitiOptionImpl.java	552		UML 图	616
15.8.12	设定试卷实现类		16.5.5	初步设计原型静态页面	616
	SetExamImpl.java	553	16.6	显示层设计	623
15.9	编写在线考试系统的控制器类	554	16.6.1	显示信息的页面 show.jsp	623
15.9.1	登录控制器类 LoginAction.java	554	16.6.2	发布信息页面 release.jsp	624
15.9.2	注册控制器类 RegeditAction.java	555	16.6.3	用户注册页面 regedit.jsp	625
15.9.3	设定试题种类控制器类		16.6.4	登录页面 login.jsp	626
	ShitiTypeAction.java	556			

16.6.5 错误处理的页面 error.jsp	627	16.8.2 开发注册用户接口	636
16.7 编写值对象 VO 和 DAO 层	628	16.8.3 开发发布接口	636
16.7.1 用户类	628	16.8.4 开发登录实现类	637
16.7.2 用户权限类	629	16.8.5 开发注册实现类	637
16.7.3 信息类	629	16.8.6 开发发布实现类	638
16.7.4 信息类别类	631	16.9 开发控制器层	640
16.7.5 开发用户 DAO 接口	631	16.9.1 开发用户登录控制器类	640
16.7.6 开发信息 DAO 接口	632	16.9.2 开发用户注册控制器类	641
16.7.7 信息类别 DAO 接口	632	16.9.3 开发信息发布控制器类	642
16.7.8 实现用户 DAO 接口	633	16.9.4 开发显示信息控制器类	644
16.7.9 实现信息 DAO 接口	633	16.10 编写辅助工具类	646
16.7.10 实现信息类别 DAO 接口	635	16.11 编写 Spring 配置文件	652
16.8 开发业务逻辑层	635	16.12 测试本项目程序	656
16.8.1 开发登录接口	636	16.13 小结	660

第 1 章

J2EE 快速入门

本章导读 ●●●●●

很多初学 Java 的读者经常会被这样一个词汇所迷惑，那就是到底什么是 J2EE 呢？这里给出一个定义：J2EE 是一种用于建立服务器方应用程序的系统平台，它被定义为开发、部署、管理多层以网络和组件技术为基础、以服务端为中心的企业级应用提供的开放的标准平台。J2EE 本身是一个标准，一个为企业分布式应用开发提供的标准平台。J2EE 也是一个框架，包括 JDBC, JNDI, RMI, JMS, EJB, JTA 等技术。

本章将从 J2EE 的来源讲起，讲解 J2EE 的整体框架，并介绍目前最新的 JavaEE，然后着重讲解 J2EE 的组件、容器及其核心技术，让读者真正明白什么是 J2EE，从而建立一种大局观。



1.1 J2EE 概述

J2EE 是使用 Java 技术开发企业级应用的一种事实上的工业标准（但 Java 及其相关技术并没有纳入标准化组织的体系），它是 Java 技术不断适应和促进企业级应用过程中的产物。Java 体系共有三个版本：适用于小型设备和智能卡的 J2ME、适用于桌面系统的 J2SE 和适用于企业级应用的 J2EE。本书主要讲解 J2EE。

1.1.1 J2EE 的来源

面向对象技术的出现和应用大大提高了软件的重用性和质量。但如今许多企业都需要扩展他们的业务范围，降低自身经营成本，缩短他们和客户之间的响应时间，这就需要在企业、合作伙伴和雇员之间提供一种简捷、快速的服务，传统的 Client/Server (C/S) 结构已经逐渐不能适应新的要求。为了克服传统 C/S 结构的弊病，快速设计和开发企业级的应用程序，迎合 B/S 架构的潮流，Sun 公司推出了一种全新概念的模型——J2EE，其英文全称是 Java2 Platform Enterprise Edition，它有着传统的互联网应用程序模型不可比拟的优势。

J2EE 为应用 Java 技术开发服务器端应用提供一个独立的、可移植的、多用户的、安全的和基于标准的企业级平台，从而简化企业应用的开发、管理和部署。J2EE 是一个标准，而不是一个现成的产品。各个平台开发商按照 J2EE 规范分别开发了不同的 J2EE 应用服务器，J2EE 应用服务器是 J2EE 企业级应用的部署平台。由于它们都遵循了 J2EE 规范，因此，使用 J2EE 技术开发的企业级应用可以部署在各种 J2EE 应用服务器上。

传统的 C/S 结构如图 1.1 所示，该结构属于两层结构，虽然很容易配置，但很难扩展企业的应用和服务。

为了更好地为企业应用提供服务，使企业应用具有高可用性、安全性、可依赖性和可扩展性，从图 1.1 可看出，需要将客户端中的逻辑处理功能分离出来，从而使客户端只负责显示功能。

在 1999 年底，Sun 公司推出了以 Java2 技术为基础的 J2EE 技术规范，该规范提供了与平台无关的、可移植的、安全的并且完全基于 Java 技术的服务器端中间件标准。从整体架构来看，应用服务器涵盖了 Web 服务器及其他的服务。J2EE 降低了开发中间层服务的成本和复杂程度，因而使得服务可以被快速地展开，并能够更轻松地面面对竞争中的压力。

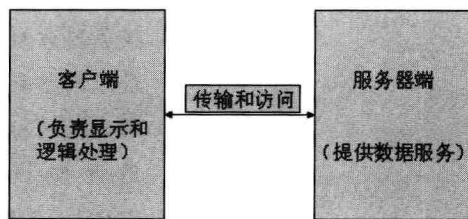


图 1.1 传统的 C/S 结构

1.1.2 J2EE 整体框架

J2EE 平台使用了一个多层的分布式应用程序模型。应用程序的逻辑根据其实实现的不同功能被封装到组件中，组成 J2EE 应用程序的大量应用程序组件根据其所属的层被安装到不同的机器中，该模型具有重用组件的能力、基于扩展标记语言的数据交换、统一的安全模式和