



全国信息技术人才培养工程指定培训教材
Web应用开发工程师职业教育系列规划教材

轻量级Java EE体系架构程序开发

工业和信息化部教育与考试中心 组编
清华大学计算机与信息管理中心 主编

严谨

勤奋

求实

创新



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

全国信息技术人才培养工程指定培训教材
Web 应用开发工程师职业教育系列规划教材

轻量级 Java EE 体系架构 程序开发

工业和信息化部教育与考试中心 组编
清华大学计算机与信息管理中心 主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

Struts+Spring+Hibernate 即 SSH 框架,是目前主流的轻型 Java EE 开发模式。而 JSF 框架则是新近崛起的一个新的界面开发框架。这几种框架在实际的项目应用中都非常流行。因为本书是一个 Java EE 开发的完整教程,所以书中选择了这四种最流行的框架进行分析讲解,确保书中的每一部分内容都能为读者带来有益的提高。

本教材与其他同类教材最大的区别在于,本教材将一个电子商务系统所要用的知识点分散到每一章,并在每一章中都有一个部分来做实例讲解。将每一章的实例讲解内容融合到一起,读者自然就能组成一个完整的电子商务程序。采用这种组织方式,每一章的知识点也都立刻能用一个实例进行验证。因此,读者读完本书之后,就能对开发一个完整的 Java EE 应用程序的工作流程有一个清晰的认识。本教材的主要目标就是让读者在阅读本书的过程中,感觉好像在实际参与一个项目一样。

因此,本教材分为四个部分,共 18 章。每一个部分介绍一种框架,每一章主要包括四个部分:学习目标、知识点讲解、实例讲解以及习题。学习目标让读者清楚了解学习本章的目的,知识点讲解主要提供各种理论知识的讲解,实例讲解则将本章的知识点融入电子商务系统的某一个部分中作为一个实例进行分析,最后的习题将针对本章的重点内容让读者练习。

最后,希望本教材能够作为读者在学习 Java EE 开发过程中的一个有益帮手。

图书在版编目(CIP)数据

轻量级 Java EE 体系架构程序开发/清华大学计算机与
信息管理中心主编. —北京:中国铁道出版社,2010.2
全国信息技术人才培养工程指定培训教材
Web 应用开发工程师职业教育系列规划教材
ISBN 978-7-113-11059-8

I. ①轻… II. ①清… III. ①JAVA 语言—程序设计—
教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2010)第 025959 号

书 名:轻量级 Java EE 体系架构程序开发

作 者:工业和信息化部教育与考试中心 组编 清华大学计算机与信息管理中心 主编

策划编辑:秦绪好 王春霞

责任编辑:翟玉峰

封面设计:付 巍

编辑助理:邱雪娇

编辑部电话:(010) 63560056

封面制作:白 雪

责任印制:李 佳

出版发行:中国铁道出版社(北京市宣武区右安门西街 8 号 邮政编码:100054)

印 刷:河北省遵化市胶印厂

版 次:2010 年 3 月第 1 版 2010 年 3 月第 1 次印刷

开 本:787mm×1092mm 1/16 印张:25.25 字数:626 千

印 数:3 000 册

书 号:ISBN 978-7-113-11059-8

定 价:38.00 元

版权所有 侵权必究

本书封面贴有中国铁道出版社激光防伪标签,无标签者不得销售

凡购买铁道版图书,如有印制质量问题,请与本社计算机图书批销部联系调换。

当今世界，随着信息技术在经济社会各领域不断深化的应用，信息技术对生产力以至于人类文明发展的巨大作用越来越明显。党的“十七大”提出要“全面认识工业化、信息化、城镇化、市场化、国际化深入发展的新形势新任务”，“发展现代产业体系，大力推进信息化与工业化融合”，明确了信息化的发展趋势，首次鲜明地提出了信息化与工业化融合发展的崭新命题，赋予了我国信息化全新的历史使命。近年来，日新月异的信息技术呈现出新的发展趋势，信息技术与其他技术的结合更加紧密，信息技术应用的深度、广度和专业化程度不断提高。

我国的信息产业作为国民经济的支柱产业正面临着有利的国际、国内形势，电子信息产业的规模总量已进入世界大国行列。但是我们也清楚地认识到，与国际先进水平相比，我们在产业结构、核心技术、管理水平、综合效益、普及程度等方面，还存在较大差距，缺乏创新能力与核心竞争力，“大”而不强。国际国内形势的发展，要求信息产业不仅要做大，而且要做强，要从制造大国向制造强国转变，这是信息产业今后的重点工作。要实现这一转变，人才是基础。机遇难得，人才更难得，要抓住本世纪头二十年的重要战略机遇期，加快信息行业发展，关键在于培养和使用好人才资源。《中共中央、国务院关于进一步加强对人才工作的决定》指出，人才问题是关系党和国家事业发展的关键问题，人才资源已成为最重要的战略资源，人才在综合国力竞争中越来越具有决定性意义。

为抓住机遇，迎接挑战，实施人才强业战略，原信息产业部于2004年启动了“全国信息技术人才培养工程”。根据工业和信息化部人才工作要点关于“继续组织实施全国信息技术人才培养工程”的要求，工业和信息化部教育与考试中心将继续推进全国信息技术人才培养工程二期工作的开展。该项工程旨在通过政府政策引导，充分发挥全行业和社会教育培训资源的作用，建立规范的信息技术教育培训体系、科学的培训课程体系、严谨的信息技术人才评测服务体系，培养造就大批行业急需的、结构合理的高素质信息技术应用型人才，以促进信息产业持续快速协调发展。

根据信息产业对技术人才素质与能力的需求，在充分吸取国内外先进信息技术培训课程优点的基础上，工业和信息化部教育与考试中心组织各方专家精心编写了信息技术系列培训教材。这些教材注重提升信息技术人才分析问题和解决问题的能力，对各层次信息技术人才的培养工作具有现实的指导意义。我们谨向参与本系列教材规划、组织、编写的同志们致以诚挚的感谢，并希望该系列教材在全国信息技术人才培养工作中发挥有益的作用。

工业和信息化部教育与考试中心
2009年10月

本书根据作者多年的 Java EE 开发经验,对 Java EE 领域中最流行的四个框架(Struts、Spring、JSF 以及 Hibernate)进行了详细的阐述和分析。本书是一本引领读者进入 Java EE 世界的教程,书中将通过一个实际的电子商务网站作为实例进行讲述,每一章都将电子商务网站中与本章理论内容相关的知识进行统一论述,从而使读者能容易地把每一章的理论和实际项目相结合。

本书的目的是向读者介绍所有的企业级 Java API,最大的特色就是将一个实际的电子商务实例中所需要用到的各种开发技术分散到每一章去讲解,使读者不仅仅学习理论知识,也不仅仅学习一个个零散的实例,而是让整个学习过程连贯起来,让读者能够清楚地看到一个完整的项目从开始实施到最后完成的整个过程。在 Java EE 的世界里,有着众多实现各种功能和目的的框架,对于一个刚刚进入 Java 开发领域的读者来说,往往会觉得无所适从。本书将从专业的角度为读者分析目前 Java EE 领域中最流行的这四个框架各自的价值,以及它们各自希望解决的问题。而作为一个程序员,在实际的开发过程中,应该根据各个框架所提供的功能及其优缺点,针对实际的项目进行合理的选择。本书会针对同一个问题,对相关框架针对这个问题的解决方案进行讲解,并进行一定程度上的对比,方便读者了解各自的优缺点。

在这里要特别说明的是,大约在 2005 年, Sun 公司(在撰写本书时 Sun 公司已经并入了 Oracle 公司)的伙伴们将原来的名称 J2EE 1.5 正式改名为 Java EE 5,对于以前的版本,则还依然称为 J2EE。这一次的改名,使大家更清楚 J2EE 只是 Java 企业应用,更加突出 Java 在其中的地位,原来的称呼容易让初学者误解是独立于 Java 的一套技术方案。

本书的阅读对象是有一定 Java EE 基础,已经了解了 Java、JSP 开发等基本的 Java EE 开发方法的读者。因此,本书不再介绍编程语言本身,而是认为读者已经知道如何在其所使用的开发环境编译和调试 Java EE 程序。

本书主要由四个部分组成,分别介绍 Struts、Spring、JSF 以及 Hibernate 这四种开发框架。另外,在本书中也简单介绍了 Maven 工具的使用。因此,读者可以根据自己的需要,选择不同的部分进行阅读。

本书由清华大学计算机与信息管理中心主编。参加编写的人员有:孙睿、邹文艳、孙燕、罗琳、祝军。由于编者水平有限,书中疏漏和不足之处在所难免,敬请有关专家和读者批评指正。

编 者

2010 年 3 月

第 1 章 Struts 简介	1	2.3.2 配置 Struts 的配置文件	24
1.1 Struts 概述	1	2.4 编译、部署和测试 Struts 应用	
1.2 MVC 设计模式	2	程序	25
1.2.1 Model 2 的出现	3	2.5 Struts 使用小结	29
1.2.2 应用层—表示层之间的		2.6 实例讲解——电子商务后台登录	
去耦合	3	程序	30
1.2.3 Struts 与 MVC	4	小结	35
1.3 Struts 的获取	5	习题	35
1.3.1 Struts 包简介	5	第 3 章 Struts 框架结构	36
1.3.2 Struts 的运行环境	6	3.1 JSP 开发模式	36
1.4 运行 Struts 样例程序	7	3.1.1 Model 1	36
1.4.1 部署 struts-blank 样例	7	3.1.2 Model 2	37
1.4.2 部署 struts-cookbook 包	7	3.1.3 Model 2x	38
1.4.3 部署 struts-el-example 包	8	3.2 Struts 结构	39
1.4.4 部署 struts-examples 包	9	3.2.1 控制器 (Controller)	40
1.4.5 部署 struts-faces-example1		3.2.2 模型 (Model)	40
包	9	3.2.3 视图 (View)	40
1.4.6 部署 struts-faces-example2		3.3 Struts 概述	40
包	10	3.3.1 ActionServlet 类	41
1.4.7 部署 struts-mailreader 包	10	3.3.2 Action 类	41
1.4.8 部署 struts-scripting-		3.3.3 ActionForm 类	42
mailreader 包	10	3.3.4 ActionMappings 类	42
1.5 实例讲解——搭建使用 Struts 的		3.3.5 ActionMapping 类	43
开发环境	11	3.3.6 Error 类	43
小结	14	3.4 基于 Struts 进行开发	43
习题	15	3.5 实例讲解——电子商务产品展示	
第 2 章 Struts 入门	16	程序	44
2.1 编写 Action	16	小结	52
2.2 编写第一个页面	19	习题	52
2.2.1 设置文件的编码格式	19	第 4 章 使用 Eclipse 和 MyEclipse 开发	
2.2.2 对国际化的支持	20	Struts 应用程序	53
2.2.3 编写 JSP 页面	22	4.1 Eclipse 和 MyEclipse 简介	53
2.3 配置 Action 和 Forward	23	4.1.1 Eclipse 简介	53
2.3.1 配置 web.xml 文件	23	4.1.2 MyEclipse 简介	55



4.2	安装与配置	56	6.3.1	文件上传功能的实现	112
4.3	开发步骤	57	6.3.2	文件下载功能的实现	117
4.3.1	创建 Web 项目	57	6.4	实例讲解——电子商务后台产品 发布程序	118
4.3.2	添加新的类库	59	小结	124	
4.3.3	添加 Struts 组件	59	习题	124	
4.3.4	编辑 Struts 配置文件	60	第 7 章	Validator 的使用	126
4.3.5	进行 Struts 开发	62	7.1	Validator 框架概述	126
4.4	实例讲解——构建电子商务程序的 开发环境	70	7.1.1	简介	127
小结	73		7.1.2	安装和配置	127
习题	73		7.2	使用 Validator 验证用户输入	129
第 5 章	用户界面开发基础	75	7.2.1	修改 ActionForm 的父类 ..	129
5.1	用户界面开发模式	75	7.2.2	配置 ActionForm	129
5.1.1	服务器页面	77	7.2.3	配置 ActionForm 中属性的 验证规则	130
5.1.2	JSP	77	7.2.4	初始化资源信息	134
5.2	使用 Struts 开发 JSP	78	7.3	自定义验证规则	135
5.2.1	HTML 标记库	78	7.4	JavaScript 和 Validator	137
5.2.2	Bean 标记库	83	7.5	使用正则表达式进行验证	138
5.3	数据分页显示	85	7.6	用 Struts 解决重复提交问题	139
5.3.1	原理讲解	86	7.6.1	Struts 解决重复提交的 原理	139
5.3.2	具体实现	86	7.6.2	避免重复响应用户的 实现	140
5.4	表达式语言	92	7.6.3	使用 JavaScript 解决用户 重复提交的问题	144
5.4.1	表达式语言简介	92	7.7	实例讲解——电子商务前台用户 注册流程	146
5.4.2	变量的作用域	93	小结	151	
5.4.3	隐式对象	93	习题	152	
5.5	XML 界面的输出	94	第 8 章	JSF 的开发	153
5.5.1	实现原理	95	8.1	JSF 概述	153
5.5.2	开发方法	95	8.1.1	JSF 简介	153
5.6	实例讲解——电子商务系统设计 ..	97	8.1.2	JSF 的获取和配置	154
小结	103		8.2	JSF 的导航开发	155
习题	103		8.2.1	简单的例程	155
第 6 章	使用多样性的 Action	104	8.2.2	简单的导航	157
6.1	Action 概述	104	8.2.3	导航规则设置	159
6.2	分类 Action	105			
6.2.1	IncludeAction	105			
6.2.2	ForwardAction	106			
6.2.3	DispatchAction	107			
6.3	文件上传和下载的实现	112			

8.3 JSF Expression Language	161	10.1.6 加载并存储对象	207
8.4 国际化信息	163	10.2 关联映射	211
8.5 可管理的 Bean	165	10.2.1 映射 Person 类	211
8.5.1 Backing Beans	165	10.2.2 单向 Set-based 的关联 ...	213
8.5.2 Bean 的配置与设定	166	10.2.3 使关联工作	214
8.5.3 Bean 中的 List 和 Map 类型 属性	168	10.2.4 值类型的集合	215
8.6 事件驱动的界面开发	170	10.2.5 双向关联	216
8.6.1 动作事件	170	10.2.6 使双向连起来	217
8.6.2 即时事件	173	10.3 实例讲解——使用 Hibernate 重构电子商务持久层	218
8.6.3 值变事件	174	小结	226
8.6.4 Phase 事件	176	习题	227
8.7 实例讲解——利用 JSF 框架开发 电子商务前台程序	178	第 11 章 Hibernate 的体系结构	228
小结	181	11.1 Hibernate 的体系结构简介	228
习题	181	11.1.1 体系架构概述	228
第 9 章 Hibernate 简介	182	11.1.2 Hibernate 核心组件	231
9.1 持久化技术简介	182	11.2 实例生命周期	231
9.1.1 软件体系架构简介	182	11.2.1 Transient (瞬态)	231
9.1.2 持久层技术方案	184	11.2.2 Persistent (持久态) ...	232
9.1.3 Java 对象持久化	186	11.2.3 Detached (脱管态) ...	233
9.2 ORM 概述	186	11.3 API 简介	234
9.3 Hibernate 概述	187	11.3.1 Session 接口	234
9.4 Hibernate 的组成结构	188	11.3.2 SessionFactory 接口	237
9.4.1 Hibernate Core	188	11.3.3 Configuration 类	237
9.4.2 Hibernate Tools	189	11.3.4 Query 接口	238
9.4.3 Hibernate Annotations	189	11.3.5 Criteria 接口	238
9.5 实例讲解——运行 Hibernate 样例 程序	190	11.3.6 Transaction 接口	239
小结	193	11.4 实例讲解——使用 Hibernate 开发产品搜索功能	240
习题	193	小结	243
第 10 章 Hibernate 入门	194	习题	243
10.1 开发概述	194	第 12 章 Hibernate 的映射技巧	244
10.1.1 第一个 class	195	12.1 手动配置映射文件	244
10.1.2 映射文件	198	12.1.1 Doctype 元素	244
10.1.3 Hibernate 配置	200	12.1.2 <hibernate-mapping> 元素	245
10.1.4 用 Maven 构建	201	12.1.3 <class>元素	247
10.1.5 启动和辅助类	205	12.1.4 <id>元素	250



12.1.5	<discriminator>元素	252	13.2.3	二级缓存	284
12.1.6	<version>元素	253	13.2.4	Hibernate 查询方法与缓存 的关系	290
12.1.7	<timestamp>元素	254	13.3	性能优化	291
12.1.8	<property>元素	254	13.3.1	设计阶段的考虑问题 ...	292
12.2	映射集合类	256	13.3.2	批量插入、更新和 删除	292
12.2.1	集合类的映射	256	13.4	实例讲解——电子商务客户 管理的开发	294
12.2.2	集合外键	257	小结		297
12.2.3	集合元素	258	习题		297
12.2.4	索引集合类	258	第 14 章	Spring 简介	298
12.2.5	值集合与多对多关联	259	14.1	简介	298
12.2.6	一对多关联	260	14.1.1	Spring 特点	299
12.3	实体对象关联关系映射	260	14.1.2	应用场景	300
12.3.1	单向关联	261	14.2	什么是 IoC	302
12.3.2	使用连接表的单向 关联	262	14.3	面向切面的编程方法	303
12.3.3	双向关联	264	14.3.1	AOP 概念	303
12.3.4	使用连接表的双向 关联	266	14.3.2	Spring AOP 的功能和 目标	305
12.3.5	更复杂的关联映射	268	14.3.3	AOP 代理	305
12.4	映射继承	269	14.4	Spring 框架概述	306
12.4.1	每个具体类一张表	269	14.5	实例讲解——在电子商务程序中 加入 Spring	307
12.4.2	每个类分层结构 一张表	271	小结		309
12.4.3	每个子类一张表	272	习题		309
12.5	实例讲解——电子商务中进行订单 管理	273	第 15 章	Spring 的核心技术	311
小结		277	15.1	IoC 容器和 Bean	311
习题		277	15.2	使用资源	314
第 13 章	Hibernate 的事务处理、缓存 及性能优化	278	15.2.1	Resource 接口	315
13.1	事务处理	278	15.2.2	Resource 的实现	316
13.1.1	Hibernate 中的事务 处理	279	15.2.3	ResourceLoader 接口	317
13.1.2	长对话	280	15.2.4	资源小结	318
13.1.3	数据库的隔离级别	281	15.3	数据绑定	318
13.2	两级缓存概述	282	15.3.1	BeanWrapper 类	318
13.2.1	什么是缓存	282	15.3.2	设置和获取属性值以及 嵌套属性	319
13.2.2	Hibernate 一级缓存	283			

15.3.3 内建的 PropertyEditor 实现.....	320	17.2 脱离 Struts 的 MVC 框架	358
15.4 实例讲解——用 Spring 来管理 电子商务系统中的订单	322	17.3 使用 Spring MVC 进行工作	359
小结.....	325	17.3.1 Spring MVC 简介	359
习题.....	325	17.3.2 配置 DispatcherServlet..	360
第 16 章 Spring 的数据访问.....	326	17.3.3 创建自己的控制器.....	360
16.1 与 JDBC 的整合	326	17.3.4 页面视图的实现	365
16.1.1 Spring JDBC 简介	327	17.3.5 配置控制器的映射.....	368
16.1.2 Spring JDBC 核心包的 使用	328	17.4 实例讲解——构造新的电子商务 程序表示层程序.....	373
16.1.3 Spring JDBC 数据源包 ..	337	小结	375
16.1.4 Spring JDBC 对象包	337	习题	375
16.2 与 Hibernate 的整合	338	第 18 章 Spring 的 Web 集成	376
16.2.1 利用 Spring 的 IoC 容器 创建 SessionFactory	339	18.1 多视图输出	376
16.2.2 DAO 的开发	342	18.1.1 视图和视图解析器.....	376
16.2.3 Hibernate 的事务管理...344		18.1.2 准备集成	377
16.2.4 OpenSessionInView 模式.....	346	18.1.3 JSP 和 JSTL 的使用	378
16.3 事务控制	347	18.1.4 Excel 文件的输出	379
16.3.1 事务管理的框架	347	18.1.5 PDF 的输出	381
16.3.2 声明式事务管理	349	18.1.6 XSLT 的使用	382
16.3.3 编程式事务管理	351	18.1.7 视图集成技术小结.....	384
16.4 实例讲解——电子商务系统中业务 逻辑层与数据层的整合	352	18.2 Web 框架集成.....	384
小结.....	356	18.2.1 通用配置	384
习题.....	356	18.2.2 Struts 集成	385
第 17 章 Spring MVC 入门	357	18.3 实例讲解——导出电子商务应用 系统的订单.....	388
17.1 MVC 简介	357	小结	391
		习题	391
		参考文献	392

第 1 章

Struts 简介

Struts 是一组相互协作的类、Servlet 和 JSP 标记，它们组成一个可重用的 MVC 2 设计。这个定义表示 Struts 是一个框架，而不是一个库，但 Struts 也包含了丰富的标记库和独立于该框架工作的实用程序类。Struts 最早是作为 Apache Jakarta 项目的组成部分来开发的。项目的创立者希望通过对该项目的研究，改进和提高 Java Server Pages (JSPs)、Servlet、标记库以及面向对象的技术水准。现在的 Struts 已经成长为一个独立的 Web 应用程序的开发框架，很好地解决了早期使用 JSP 和 Servlet 进行开发时遇到的各种问题。

学习目标

本章的主要目标是了解 Struts 的概况和设计方面的理念，并通过运行 Struts 本身提供的样例来说明 Struts 的各项功能。

- 了解 Struts 的功能构成；
- 学习 Struts 的设计理念；
- 了解 Struts 的获得方式；
- 运行 Struts 的相关样例。

1.1 Struts 概述

Struts 包含两个不同的框架，它们是 Struts Action Framework 和 Struts Shale Framework。其中，Struts Action Framework 是在原来 Struts 框架的基础上发展而来的，而 Struts Shale Framework 则是为了 Java Server Faces (简称 JSF) 而开发的一个基于组件的框架。

Struts Action Framework 除了实现了 MVC 模式之外，还提高了开发的速度和效率。Struts 还提供了很多扩展功能，这些扩展功能有些包含在 Struts 包中，有些已经成为了单独的组件。这些扩展功能如下所述。

说明：在本书中，主要介绍 Struts Action Framework 的使用和开发，如果了解 Struts 的更多详细资料以及使用 Struts 来开发 Web 应用程序的方法，请访问 Struts 的官方网站，其网址为 <http://struts.apache.org/>。



1. Struts Taglibs

这是 Struts 的标记库，用来简化基于 JSP 的表示层开发。它主要包括 HTML、Bean 和 Logic 这几个基本的标记。这些基本标记的功能描述如下：

- HTML 标记用来创建表单中用于用户输入的各种控件以及生成各种基于 HTML 的用户界面元素。它的输出兼容 HTML 4.01 标准。Struts 的 HTML 标记可以大致分为以下几个功能：显示表单元素和输入控件、显示错误信息和显示其他 HTML 元素。
- Bean 标记库中包含用于定义新 Bean、访问 Bean 及其属性的标记。Struts 框架提供了多种自定义标记用来在 JSP 页面中处理 JavaBean。Bean 标记库中包含 4 种类型的标记，分别是：创建和复制 Bean 的标记、脚本变量定义标记、Bean 翻译标记和消息国际化标记。
- Logic（逻辑）标记库中的标记是用来处理外观逻辑的，不需要使用 Scriptlet 脚本。Struts 逻辑标记库中包含的标记能够有条件地产生输出文本，在对象集合中循环从而重复地产生输出文本，以及应用程序流程控制。它也提供了一组在 JSP 页面中处理流程控制的标记。逻辑标记库中定义的标记能够执行下列 3 种功能：条件逻辑、循环和转发、重定向响应。

2. Struts EL

这是 Struts 标记库的扩展项目，这个标记库中的每个类都是 Struts 标准标记库中类的子类。Struts 的标记都可以使用 rtexprvalue（runtime scriptlet expression）来动态计算属性的值。例如，为了根据资源关键值打印来自属性文件中的信息，可以以下列形式编写：

```
<bean:message key="<%=stringvar%>" />
```

这样写是假定 stringvar 是 JSP 中的脚本变量，如果使用 Struts 的 EL 标记库，将采用如下形式：

```
<bean-el:message key="${stringvar}" />
```

3. Validator

Validator 现在是 Apache 软件基金组织的一个独立项目，主要用于用户输入合法性的检查。它提供了大部分需要的用户输入合法性的校验功能，主要是用户输入数据的格式以及数字范围的检查。

4. Tiles

Tiles 是一个模板框架，它是 Struts 的子项目。Tiles 用于创建风格一致的 Web 网站或应用程序以及创建可重用的视图组件。

5. Struts Scripting

这个项目使得 Struts 允许使用脚本语言来编写 Struts 的 Action，而不是通常情况下的 Java 类。它的实现依赖于 BSF（Bean Scripting Framework）组件，所有 BSF 组件支持的脚本语言在这里都可以使用。例如：Perl、Python、Ruby、Java Script、Groovy 以及 VBScript 等。

说明：关于 BSF 的详细信息请访问其官方网站，网址为 <http://jakarta.apache.org/bsf/index.html>。

1.2 MVC 设计模式

模型—视图—控制器（MVC）是 20 世纪 80 年代为编辑语言 Smalltalk-80 发明的一种软件设计模式，现在已经被广泛使用。MVC 三个部分的职责是不相同的，其各部分职责描述如下：

- 模型（Model）：模型是应用程序的主体部分。模型表示业务数据或者业务逻辑。

- 视图 (View)：视图是应用程序中用户界面相关的部分，是用户看到并与之交互的界面。
- 控制器 (Controller)：控制器就是根据用户的输入控制用户界面的数据显示和更新模型对象的状态。

第3章我们会详细描述 MVC 三个部分要完成的具体功能，这里就不再赘述。

1.2.1 Model 2 的出现

JSP (Java Server Page) 首先是作为 Servlet 的替代引入的，与其功能类似的就是微软的 ASP (Active Server Page)。Servlet 功能强大，但强大的功能伴随着巨大的责任。很多团队发现，如果他们一不小心，项目就会因为纠缠如麻的页面变得容易崩溃。进一步的特性需要使用复杂的脚本程序。但脚本程序是非常难于重用的——除非在页面之间进行代码的复制粘贴。

开发人员很快意识到，JSP 和 Servlet 可以一起使用来部署 Web 应用。Servlet 可以应付控制流，而 JSP 则可专注于编写 HTML 的任务。在这种情况下，结合使用 JSP 和 Servlet 开始被称为 Model 2 (单独使用 JSP 称为 Model 1)。

很多人很快指出 JSP Model 2 类似于经典的 Model—View—Controller 架构。现在交互使用 Model 2 和 MVC 这两个词已经很平常了。没有观察者通知的 Model—View—Controller 有时被称为 MVC 2 或 Web MVC。

1.2.2 应用层—表示层之间的去耦合

Model 2 被认为区别于 MVC 的一个原因是，观察者/通知模式不能在 Web 环境内很好地工作。HTTP 是一个“拉”的协议：客户请求然后服务器响应。没有请求就没有响应。观察者模式需要一种“推”的协议来进行通知，以便服务器能在模型改变时将信息推送到客户端。虽然也有一些方法能模拟将数据推送到客户端，但这和基本情况相悖，并且被视为权宜之计的修补。

图 1-1 所示是典型的 Model—View—Controller 范式，经常被表示为一个互相连接的三角形。在 Web 应用中维护范式中的“通知改变”部分是非常困难的。当所有资源都在一台服务器上，而且客户端保持一个开放连接的情况下，这种模式工作得非常好。如果资源分布在不同的服务器上，并且客户端不能维护一个开放的连接情况下，该模式工作得并不理想。

绝大多数情况下，远程应用是按分层的模式设计的。基本上，分层模式下，层内的对象可以和同一层或者相邻层的对象进行通信。在一个复杂应用中，这可以在添加组件时防止依赖关系呈指数增长。在设计远程应用时，分层是一个核心模式。从 MVC 上下文中，引入分层模式将状态改变和状态查询的职责加于控制器之上，并伴随着改变通知。

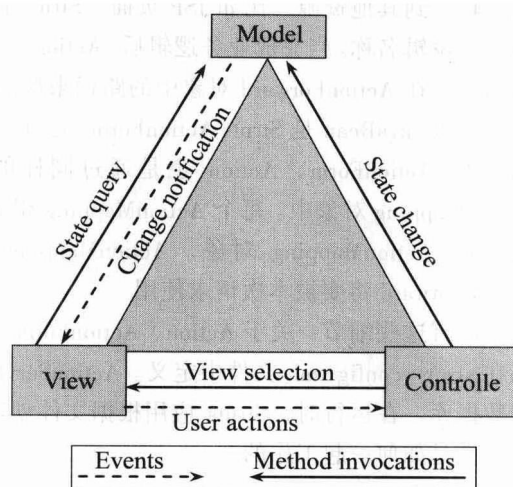


图 1-1 Model—View—Controller 范式

如图 1-2 所示,分层的 Web 应用使用一种比传统 MVC 模式更加“扁平”的模式。控制器被夹在表示层 (View) 和应用逻辑层 (Model) 之间。每个组件的主要职责并没有改变,只是流程有轻微改变,即查询状态和改变通知都必须通过控制器。另一个改变是,当视图或者表示层需要渲染动态页面时,它使用从控制器传递的数据而不是直接来自于模型层。这种改变去除了 View 和 Model 的耦合,允许控制器选择数据和显示这些数据的视图。

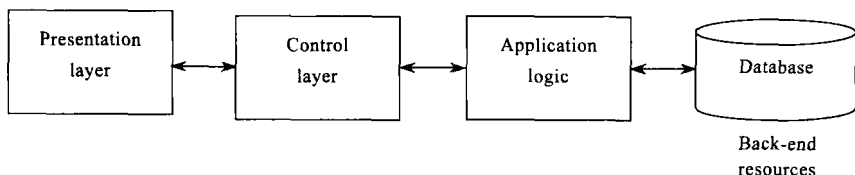


图 1-2 Web 应用的层模式

Struts 的出现就是为了将 MVC 的实现封装到框架里,这样,程序员不需要再去关注框架的实现,就可以开发 MVC 框架程序,从而降低了开发的门槛。下一节详细阐述 Struts 与 MVC 框架之间的关系。

1.2.3 Struts 与 MVC

Struts 实现 MVC 是通过 ActionServlet 控制导航流,其他 Struts 类,比如 Action,用来访问业务逻辑类。当 ActionServlet 从容器接收到一个请求,它使用 URI (或者路径“path”)来决定哪个 Action 将用来处理请求。一个 Action 可以校验输入,并且访问业务层以从数据库或其他数据服务中检索信息。为校验输入或者使用输入来更新数据库,Action 需要知道什么值被提交上来。它并不是强制每个 Action 都要从请求中抓取这些值,而是由 ActionServlet 将输入绑定到 JavaBean 中。

每个请求都必须以 HTTP 响应进行应答。通常,Struts Action 并不自行渲染响应信息,而是将请求转发到其他资源,比如 JSP 页面。Struts 提供一个 ActionForward 类,用来将一个页面的路径保存为逻辑名称。当完成业务逻辑后,Action 选择并向 Servlet 返回一个 ActionForward。然后 Servlet 使用保存在 ActionForward 对象中的路径来调用页面完成响应。

输入 JavaBean 是 Struts ActionForm 类的子类。ActionServlet 通过查找请求的路径可以决定使用哪个 ActionForm,Action 也是通过同样的方法选取的。Struts 将这些细节都绑定在一个 ActionMapping 对象中。每个 ActionMapping 相对于一个特定的路径。当某个路径被请求时,Servlet 就查询 ActionMapping 对象。ActionMapping 对象告诉 Servlet,哪个 Action、ActionForm 和 ActionForward 将要被本次请求使用。

所有这些细节,关于 Action、ActionForm、ActionForward、ActionMapping 以及其他一些东西,都在 struts-config.xml 文件中定义。ActionServlet 在启动时读取这个配置文件,并创建一个配置对象数据库。在运行时,Struts 应用根据文件创建的配置对象,而不是文件本身。图 1-3 显示了这些组件是如何一起工作的。

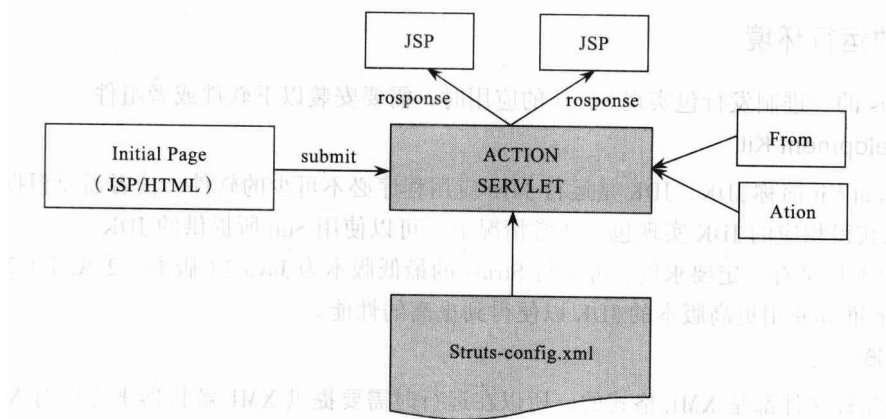


图 1-3 Struts 组件

1.3 Struts 的获取

经过上述介绍，大家已经对 Struts 有了一个大致的了解，是否有些迫不及待地想要亲自试试 Struts 的开发了呢？那么，这一节就开始介绍如何开始使用 Struts。当然，第一步就是要获取 Struts 的发行包。这个发行包可以直接在 <http://struts.apache.org/download.html> 上获得。

1.3.1 Struts 包简介

在 Struts 的官方网站上提供了两个正式的发行包，分别是 Struts 包和 Struts Scripting 包。这两个包的功能描述如下：

- Struts 包就是前面所介绍的 Struts 主要框架的实现包，它提供了对前面所介绍的各种功能的支持。虽然 Validator 已经作为一个独立的组件来进行单独的管理，但在 Struts 的发行包中还是包含了这方面的内容。
- Struts Scripting 包所提供的功能使 Struts 可以支持通过脚本语言来编写 Action 程序。

这两种功能的发行包都提供了相应的二进制形式和源代码形式的两个包。对于只是使用 Struts 的用户来说可以直接下载二进制形式的发行包，这样可免去自己进行编译所带来的麻烦。而对于想了解 Struts 实现方法的读者，则可以通过源代码包去了解。Struts 的二进制发行包主要包括以下几个部分：

- contrib 文件夹：这个文件夹中主要包含了 Struts EL 标记库的库文件以及包含标记定义的 DTD 文件。另外在这个文件夹中还提供了一个 Struts EL 标记库的样例包，将其部署到 Web 容器中可以直接了解到 Struts EL 的主要功能。
- lib 文件夹：lib 文件夹中主要包括 Struts 的库文件以及 Struts 所依赖的一些库文件，另外还包括了 Struts 标记库定义的 DTD 文件。
- apps 文件夹：这个文件夹中包含了 Struts 所提供的样例程序包，这些样例程序都是 war 格式的，部署到 Web 容器中后可以直接通过浏览器访问。

Struts 的源代码包则要复杂一些了，它包括 Struts 的源代码以及所有样例程序的源代码，另外还包含了大量的相关文档。

1.3.2 Struts 的运行环境

直接使用 Struts 的二进制发行包实现 Struts 的应用时，需要安装以下软件或者组件。

1. Java Development Kit

Java Development Kit 简称 JDK，JDK 是运行 Java 应用程序必不可少的软件。读者需要根据自己所使用的平台来获得相应的 JDK 实现包。通常情况下，可以使用 Sun 所提供的 JDK。

对于 JDK 的版本也是有一定要求的，可运行 Struts 的最低版本为 Java 2（版本 1.2 或者 1.2 以上），但通常情况下推荐使用更高版本的 JDK 以便得到更高的性能。

2. XML 解析器

由于 Struts 的配置文件都是 XML 格式的，所以在运行时需要提供 XML 解析器来完成对 XML 配置文件的解析工作。所要求的 XML 解析器必须要兼容 Java API for XML Parsing（JAXP）1.1 或者 1.1 版本以上的规范。

说明：由于版本 1.4 或者 1.4 以上的 JDK 已经自带了 XML 解析器，所以使用 1.4 或者 1.4 版本以上的 JDK 就不需要再自己提供 XML 解析器了。

3. Servlet Container

必须要为 Struts 提供一个 Servlet Container 运行的环境，要求这个 Servlet Container 必须严格遵守 Servlet API 以及 JSP 的说明。其中对于 Servlet API 的支持要求在版本 2.2 或者 2.2 以上，而对于 JSP 的支持则要求在版本 1.1 或者 1.1 以上。

通常情况下会选择 Apache 软件基金组织的另一个开源项目 Tomcat 来作为运行 Struts 的 Servlet Container。在本书的介绍过程中，也主要选择 Tomcat 作为运行各个框架的 Servlet Container。

说明：Tomcat 官方网站的网址为 <http://tomcat.apache.org>。

4. 其他组件

Struts 在运行过程中还需要一些其他组件的支持，如表 1-1 所示。

表 1-1 Struts 的依赖组件

组 件 名 称	网 址
antlr	http://www.antlr.org/
Jakarta Commons Beanutils	http://jakarta.apache.org/commons/Beanutils/
Jakarta Commons Digester	http://jakarta.apache.org/commons/digester/
Jakarta Commons FileUpload	http://jakarta.apache.org/commons/fileupload/
Jakarta Commons Logging	http://jakarta.apache.org/commons/logging/
Jakarta Commons Validator	http://jakarta.apache.org/commons/validator/
Jakarta ORO	http://jakarta.apache.org/oro/

说明：以上软件包在 Struts 的二进制发行包中都有，欲了解其详细信息请参考其官方网站上的资料。

这里所列举的只是使用 Struts 开发 Web 应用所必须使用的软件或者组件,如果要通过源代码来编译 Struts 则需要更多的组件提供支持。关于这方面的详细信息请参考 <http://struts.apache.org/struts-doc-1.2.x/userGuide/installation.html> 上的内容。

1.4 运行 Struts 样例程序

在这里首先要运行一下 Struts 的样例程序,这样便于对 Struts 有一个初步的了解。在运行具体的 Struts 应用之前,还需要进行以下准备工作来完成 Struts 样例程序运行环境的构建。

说明:本书不再对 JDK 的安装以及 Tomcat 的安装进行详述,具体这两部分的安装过程请参阅相关书籍。这里主要介绍在 Tomcat 环境中部署 Struts 发行包中包含的各个样例程序。

1.4.1 部署 struts-blank 样例

struts-blank 包是一个空的 Struts 项目,它只提供了一个最基本的显示页面。Struts 提供这个包的主要目的是为 Struts 的开发工作提供一个基础的框架,使开发人员可以方便地在此基础上进行 Struts 应用的各种开发工作。

由于 Struts 所提供的是 war 格式的压缩包,所以部署起来非常方便,只需要直接将这个包放置在 Tomcat 的 webapps 目录中就可以了。

说明:后面的样例程序包都可以采用此方法进行部署。并且,所有的 war 包都可以使用这种部署方法。

运行后的 struts-blank 包如图 1-4 所示。

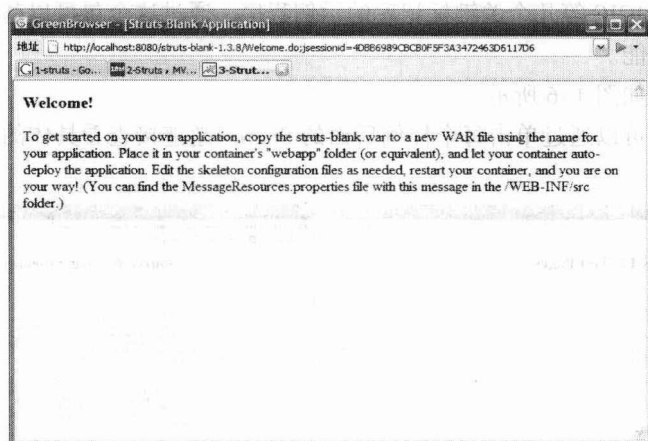


图 1-4 struts-blank 项目首页

说明:以后的开发项目都能够以此项目为基础进行开发,从而免去很多初始配置工作。

1.4.2 部署 struts-cookbook 包

struts-cookbook 包所包含的内容是 Struts 的一些带有源码的例子,这些例子都有比较详细的注释说明,因此作为一个刚开始学习 Struts 的手册是非常有作用的。