

在线服务：视频库、源代码库、专业论坛、专家实时支持

Struts 2

完全学习手册

王伟平 等编著



50段全程配音语音教学视频

全书实例源代码，使学习、分析、调试程序更方便

在线服务方式

在线服务网站：www.itzcn.com

QQ群在线服务：45368980、33925615、107423140

清华大学出版社

在线服务：视频库、源代码库、专业论坛、专家实时支持

Struts 2

完全学习手册

王伟平 等编著



50段全程配音语音教学视频

全书实例源代码，使学习、分析、调试程序更方便

在线服务方式

在线服务网站：www.itzcn.com

QQ群在线服务：45368980、33925615、107423140

清华大学出版社
北 京

内 容 简 介

Struts 是第一款真正意义上按照 MVC 架构模式搭建的 Web 开发框架。本书全面讲述 Struts 2 开发框架的知识, 包括 Struts 2 中配置文件的详细讲解、国际化与异常处理、数据类型的转换、数据的输入校验、Struts 2 强大的标签库、Struts 2 中文件的上传与下载、Struts 2 对 Ajax 的支持、Struts 2 与流行框架的整合, 最后以 3 个实例由简单到复杂地介绍 Struts 2 在实际应用中的使用。本书光盘提供了全程配音视频文件和实例源代码。

本书适合大专院校在校学生、网站开发人员、Struts 学习和开发人员以及编程爱好者学习和参考。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目 (CIP) 数据

Struts 2 完全学习手册 / 王伟平等编著. —北京: 清华大学出版社, 2011. 1
ISBN 978-7-302-22312-2

I. ①S… II. ①王… III. ①软件工具—程序设计—手册 IV. ①TP311.56-62

中国版本图书馆 CIP 数据核字 (2010) 第 055720 号

责任编辑: 夏兆彦

责任校对: 徐俊伟

责任印制: 李红英

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62795954, jsjic@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 清华大学印刷厂

装 订 者: 北京市密云县京文制本装订厂

经 销: 全国新华书店

开 本: 190×260 印 张: 38 字 数: 938 千字

附光盘 1 张

版 次: 2011 年 1 月第 1 版

印 次: 2011 年 1 月第 1 次印刷

印 数: 1~5000

定 价: 69.50 元

产品编号: 034174-01

前 言

Struts 1 自推出以来,经过多年的发展,已经成为了一个高度成熟的框架,不管是稳定性还是可靠性都得到了广泛的证明,拥有丰富的开发人群。但是随着时间的流逝以及技术的进步,Struts 1 的局限性也越来越多地暴露出来,并且制约了 Struts 1 的继续发展。

对于 Struts 1 框架而言,由于与 JSP/Servlet 耦合非常紧密,因此导致了一些严重的问题。首先,Struts 1 支持的表现层技术单一。由于 Struts 1 出现的年代比较早,那个时候没有 FreeMarker、Velocity 等技术,因此它不可能与这些视图层的模板技术进行整合。其次,Struts 1 与 Servlet API 的严重耦合,使应用难于测试。最后,Struts 1 代码严重依赖于 Struts 1 API,属于侵入性框架。

为了适应开发需求,以及应对新型视图层框架(例如 JSF、Tapestry 和 Spring MVC)的冲击,Struts 社区与大家熟知的 WebWork 社区联合推出了最新的 Struts 框架——Struts 2。Struts 2 吸取了 WebWork 框架与 Struts 1 框架的优点,在稳定性与性能方面都有了更好的保证。

1. 本书内容

本书全面地介绍 Struts 2 框架的各个知识层面,最主要的内容包括 Struts 2 框架的配置文件、Struts 2 强大的标签库以及 Struts 2 与其他框架的整合,向读者全面展示 Struts 2 框架。本书共分为 4 篇,分别是 Struts 2 基础篇、Struts 2 框架技术篇、Struts 2 整合其他技术篇和 Struts 2 综合实例篇。各篇主要内容如下。

第 1 篇: Struts 2 基础篇(第 1~8 章)。本篇主要介绍 Struts 2 的 MVC 设计理念、工作流程、文件配置、国际化和异常处理、拦截器、类型转换、输入校验以及 Struts 2 的默认表达式语言 OGNL。其中,拦截器是 Struts 2 框架的核心。

第 2 篇: Struts 2 框架技术篇(第 9~14 章)。本篇着重介绍 Struts 2 框架的标签库、文件上传与下载以及 Struts 2 对 Ajax 的支持,同时也非常具体地介绍 Struts 2 的数据库应用、Struts 2 的应用程序安全以及如何避免表单重复提交与设置等待页面。

第 3 篇: Struts 2 整合其他技术篇(第 15~18 章)。与各种流行框架的整合是学习 Struts 2 的一个重点。本篇主要介绍 Struts 2 与 Hibernate、Spring、JSF、SiteMesh、Tiles 等框架的整合。

第 4 篇: Struts 2 综合实例篇(第 19~21 章)。本篇介绍 3 个综合实例,分别是用户注册系统、企业 BBS 系统和电子相册系统。其中,用户注册系统单独使用 Struts 2 框架实现;企业 BBS 系统结合 Struts 2 与 Hibernate 技术实现;电子相册系统结合 Struts 2+Hibernate+Spring 这 3 种主流框架技术整合开发实现。

2. 本书特色

本书采用大量的实例进行讲解,力求通过实际操作使读者轻松地掌握 Struts 2 框架技术的使用。本书难度适中,内容由浅入深,实用性强,覆盖面广,条理清晰。

- ❑ **知识点全** 本书全面地介绍 Struts 2 框架中的各个知识点，具有很强的系统性。
- ❑ **基于理论，注重实践** 在讲述过程中，不仅仅只介绍理论知识，而且在合适位置安排具有代表性、操作性强的综合应用，将理论应用到实践当中来加强读者的实际应用能力，巩固对 Struts 2 框架理论知识的学习。
- ❑ **快速掌握** 注重技术原理和实际应用的高度融合，通过循序渐进的内容组织，帮助读者快速掌握和应用 Struts 2 框架技术。
- ❑ **案例实用** 本书花了很大篇幅去讲解 Struts 2 框架与其他流行框架的整合，让读者切切实实地掌握 Struts 2 框架的实际应用，并在最后以 3 个综合实例，更具体地介绍 Struts 2 与 Hibernate、Spring 这两个目前最为流行的框架的整合。
- ❑ **随书光盘** 本书为实例配备了视频教学文件，读者可以通过视频文件更加直观地学习 Struts 2 框架知识。
- ❑ **网站技术支持** 读者在学习或者工作的过程中，如果遇到实际问题，可以直接登录 www.itzcn.com 与我们取得联系，作者会在第一时间内给予帮助。

3. 读者对象

本书具有知识全面、实例精彩、指导性强的特点，力求以全面的知识及丰富的实例来引导读者透彻地学习 Struts 2 框架各方面的知识。本书可以作为 Struts 2 框架的入门书籍，也可以帮助中级读者提高技能，对高级读者也有一定的启发意义。

本书适合以下人员阅读学习。

- ❑ 系统开发人员。
- ❑ 系统维护人员。
- ❑ Struts 2 初学者。
- ❑ MVC 框架编程人员。
- ❑ 各大中专院校的在校学生和相关授课老师。

除了封面署名人员之外，参与本书编写的还有杨梦、于永军、张秋香、李乃文、张仕禹、夏小军、赵振江、李振山、李文才、吴越胜、李海庆、何永国、李海峰、陶丽、吴俊海、安征、张巍屹、崔群法、王咏梅、康显丽、辛爱军、牛小平、贾栓稳、王立新、苏静、赵元庆、郭磊、徐铭、李大庆、王蕾、张勇、郝安林、郭新志、牛丽平、唐守国等。在编写过程中难免会有疏漏，欢迎读者与我们联系，帮助我们改正提高。

目 录

第一篇 Struts 2 基础篇

第 1 章 Struts 2 概述 1

- 1.1 MVC 概述 1
 - 1.1.1 MVC 设计模式 1
 - 1.1.2 MVC 的优点 3
 - 1.1.3 Model 1 和 Model 2 体系介绍 3
- 1.2 Struts 1 简介 4
 - 1.2.1 Struts 1 框架介绍 5
 - 1.2.2 Struts 1 配置文件 6
 - 1.2.3 Struts 1 处理过程 8
 - 1.2.4 Struts 1 简单示例 9
- 1.3 WebWork 概述 1 14
- 1.4 Struts 2 简介 15
 - 1.4.1 Struts 2 框架体系 15
 - 1.4.2 Struts 2 配置文件 16
 - 1.4.3 Struts 2 控制器 17
 - 1.4.4 Struts 2 标签库 18
 - 1.4.5 与 Struts 1、WebWork 对比 19

第 2 章 Struts 2 起步 22

- 2.1 搭建 Struts 2 开发环境 22
 - 2.1.1 安装配置 JDK 22
 - 2.1.2 下载安装 Tomcat 23
 - 2.1.3 下载安装 Struts 2 25
 - 2.1.4 使用开发工具 MyEclipse 26
- 2.2 Struts 2 应用示例 28
 - 2.2.1 web.xml 配置文件 29
 - 2.2.2 Action 业务控制器 29
 - 2.2.3 struts.xml 配置文件 30
 - 2.2.4 视图文件 31

- 2.2.5 运行示例 31

2.3 Struts 2 的工作流程 32

- 2.3.1 核心控制器 FilterDispatcher 32
- 2.3.2 业务控制器 Action 34
- 2.3.3 业务逻辑组件 34

2.4 Struts 2 中使用 POJO 34

第 3 章 Struts 2 配置详解 38

3.1 Struts 2 配置文件 38

- 3.1.1 web.xml 38
- 3.1.2 struts.xml 40
- 3.1.3 struts-default.xml 44
- 3.1.4 struts.properties 46
- 3.1.5 struts-plugin.xml 48
- 3.1.6 静态内容 49

3.2 struts.xml 配置详解 50

- 3.2.1 Bean 配置 50
- 3.2.2 常量配置 51
- 3.2.3 包配置 53
- 3.2.4 命名空间配置 54
- 3.2.5 包含配置 56
- 3.2.6 拦截器配置 56

3.3 配置 Action 57

- 3.3.1 Action 实现类 58
- 3.3.2 间接访问 Servlet API 59
- 3.3.3 直接访问 Servlet API 62
- 3.3.4 Action 配置 65
- 3.3.5 动态方法调用 66
- 3.3.6 使用 method 属性 68
- 3.3.7 通配符映射 70

3.3.8 默认 Action	71	4.5.3 配置异常处理	121
3.4 配置 Result	71	4.5.4 异常处理实例	122
3.4.1 结果映射	72	第 5 章 拦截器	125
3.4.2 结果类型	73	5.1 拦截器介绍	125
3.4.3 dispatcher 结果类型	75	5.1.1 AOP 介绍	125
3.4.4 redirect 结果类型	76	5.1.2 拦截的实现原理	126
3.4.5 redirectAction 结果类型	77	5.1.3 拦截的意义	127
3.4.6 其他结果类型	78	5.2 Struts 2 拦截器	128
3.4.7 使用通配符动态配置 result	82	5.2.1 Struts 2 拦截器原理	128
3.4.8 使用 OGNL 动态配置 result	83	5.2.2 配置拦截器	129
3.5 使用注解配置 Action	84	5.2.3 使用拦截器	131
3.5.1 与 Action 配置相关的注解	84	5.2.4 自定义默认拦截器	133
3.5.2 使用注解配置 Action 的注意事项	86	5.2.5 使用拦截器注解	134
3.5.3 使用注解配置 Action 示例	87	5.3 自定义拦截器	137
第 4 章 国际化与异常处理	90	5.3.1 自定义拦截器类	137
4.1 软件国际化	90	5.3.2 使用自定义拦截器	139
4.2 Java 国际化	91	5.4 拦截器深度剖析	142
4.2.1 Java 对国际化的支持	91	5.4.1 拦截器的方法过滤	142
4.2.2 Java 支持的语言和区域	94	5.4.2 拦截器的执行顺序	146
4.2.3 使用资源文件	95	5.4.3 拦截结果监听器	147
4.2.4 使用资源类文件	97	5.4.4 设置拦截器栈中拦截器参数	149
4.2.5 MessageFormat 类	100	5.5 Struts 2 框架的系统拦截器	152
4.3 Struts 2 国际化	101	5.5.1 系统拦截器	152
4.3.1 配置资源文件	101	5.5.2 timer 拦截器实例	157
4.3.2 Struts 2 国际化应用	102	5.6 权限拦截器实例	158
4.3.3 使用占位符	105	5.6.1 权限拦截器	158
4.3.4 包范围资源文件	108	5.6.2 配置拦截器	159
4.3.5 Action 范围资源文件	111	5.6.3 业务控制器 Action	159
4.3.6 临时资源文件	112	5.6.4 JSP 视图	160
4.3.7 加载资源文件的顺序	114	5.6.5 运行程序	160
4.4 动态访问国际化资源文件	115	第 6 章 类型转换	161
4.4.1 动态访问国际化资源文件原理	115	6.1 传统的类型转换	161
4.4.2 建立资源文件	116	6.2 编写一个类型转换器	162
4.4.3 建立 Action 和 JSP	116	6.2.1 类型转换需求	162
4.4.4 动态访问资源运行结果	117	6.2.2 编写自定义类型转换器	164
4.5 Struts 2 异常处理	118	6.2.3 视图资源文件	165
4.5.1 传统异常处理方式	118	6.2.4 运行示例	166
4.5.2 Struts 2 异常处理机制	120		



6.3 自定义类型转换器	166	7.3.1 定义校验规则	203
6.3.1 基于 OGNL 的类型转换器	166	7.3.2 字段校验器配置风格	205
6.3.2 基于 Struts 2 的类型转换器	167	7.3.3 非字段校验器配置风格	206
6.3.3 注册自定义类型转换器	168	7.3.4 输入校验的国际化信息	207
6.3.4 数组属性类型转换器	171	7.3.5 校验的搜索顺序	208
6.3.5 集合属性类型转换器	173	7.4 复合类型属性的输入校验	212
6.4 使用 Struts 2 的类型转换	174	7.4.1 复合属性的校验	212
6.4.1 Struts 2 系统内建的类型转换器	174	7.4.2 集合属性的校验	215
6.4.2 null 属性的处理	174	7.5 Struts 2 框架的校验器	217
6.4.3 项目应用中常见的类型转换	175	7.5.1 内建校验器	217
6.4.4 使用 OGNL 表达式	177	7.5.2 必填校验器	218
6.4.5 使用集合类型属性	179	7.5.3 必填字符串校验器	220
6.4.6 使用 Set 类型属性	182	7.5.4 字符串长度校验器	222
6.5 类型转换中的异常处理	185	7.5.5 整数校验器	223
6.5.1 类型转换异常拦截器	185	7.5.6 类型转换校验器	224
6.5.2 一个简单的类型转换异常处理	186	7.5.7 日期校验器	226
6.5.3 改进类型转换异常显示信息	188	7.5.8 浮点数值校验器	227
6.5.4 处理复合类型转换异常	188	7.5.9 邮件地址校验器	228
6.6 使用类型转换注解	189	7.5.10 网址校验器	229
6.6.1 TypeConversion 注解	190	7.5.11 表达式校验器	231
6.6.2 Conversion 注解	191	7.5.12 字段表达式校验器	232
6.6.3 Element 注解	192	7.5.13 正则表达式校验器	234
6.6.4 Key 注解	192	7.5.14 visitor 校验器	235
6.6.5 KeyProperty 注解	192	7.6 自定义校验器	236
6.6.6 CreateIfNull 注解	193	第 8 章 OGNL	239
第 7 章 输入校验	194	8.1 OGNL 简介	239
7.1 输入校验介绍	194	8.2 OGNL 表达式	239
7.1.1 输入校验的原因	194	8.2.1 常量	240
7.1.2 客户端和服务端校验	195	8.2.2 操作符	240
7.2 使用 validate 进行输入校验	197	8.2.3 调用方法	241
7.2.1 validate()方法输入校验	197	8.2.4 索引访问	242
7.2.2 validateXxx()方法输入校验	199	8.2.5 对集合进行操作	244
7.2.3 输入校验流程	202	8.3 OGNL 的简单应用	246
7.3 基本输入校验	203	8.4 Struts 2 的 OGNL	247
7.3.1 定义校验规则	203	9.1 Struts 2 标签库概述	253
7.3.2 字段校验器配置风格	205	9.1.1 标签库简介	253
7.3.3 非字段校验器配置风格	206	9.1.2 Struts 2 标签库组成	254
7.3.4 输入校验的国际化信息	207		
7.3.5 校验的搜索顺序	208		
7.4 复合类型属性的输入校验	212		
7.4.1 复合属性的校验	212		
7.4.2 集合属性的校验	215		
7.5 Struts 2 框架的校验器	217		
7.5.1 内建校验器	217		
7.5.2 必填校验器	218		
7.5.3 必填字符串校验器	220		
7.5.4 字符串长度校验器	222		
7.5.5 整数校验器	223		
7.5.6 类型转换校验器	224		
7.5.7 日期校验器	226		
7.5.8 浮点数值校验器	227		
7.5.9 邮件地址校验器	228		
7.5.10 网址校验器	229		
7.5.11 表达式校验器	231		
7.5.12 字段表达式校验器	232		
7.5.13 正则表达式校验器	234		
7.5.14 visitor 校验器	235		
7.6 自定义校验器	236		
第 8 章 OGNL	239		
8.1 OGNL 简介	239		
8.2 OGNL 表达式	239		
8.2.1 常量	240		
8.2.2 操作符	240		
8.2.3 调用方法	241		
8.2.4 索引访问	242		
8.2.5 对集合进行操作	244		
8.3 OGNL 的简单应用	246		
8.4 Struts 2 的 OGNL	247		
9.1 Struts 2 标签库概述	253		
9.1.1 标签库简介	253		
9.1.2 Struts 2 标签库组成	254		

第二篇 Struts 2 框架技术篇

第 9 章 Struts 2 标签库

9.1 Struts 2 标签库概述

9.1.1 标签库简介

9.1.2 Struts 2 标签库组成



9.2	控制标签	255
9.2.1	if/elseif/else 标签	256
9.2.2	iterator 标签	257
9.2.3	append 标签	258
9.2.4	generator 标签	259
9.2.5	merge 标签	259
9.2.6	subset 标签	261
9.2.7	sort 标签	262
9.3	数据标签	264
9.3.1	action 标签	264
9.3.2	bean 标签	266
9.3.3	date 标签	268
9.3.4	debug 标签	269
9.3.5	include 标签	269
9.3.6	param 标签	270
9.3.7	push 标签	271
9.3.8	set 标签	272
9.3.9	url 标签	273
9.3.10	property 标签	274
9.4	主题与模板	275
9.4.1	主题	275
9.4.2	模板	276
9.4.3	使用主题模板	276
9.5	表单标签	277
9.5.1	表单标签通用属性	277
9.5.2	checkboxlist 标签	278
9.5.3	combobox 标签	280
9.5.4	doubleselect 标签	281
9.5.5	datetimepicker 标签	282
9.5.6	head 标签	283
9.5.7	file 标签	284
9.5.8	hidden 标签	284
9.5.9	select 标签	284
9.5.10	optiontranserselect 标签	285
9.5.11	radio 标签	287
9.5.12	optgroup 标签	288
9.5.13	token 标签	289
9.5.14	textfield、password 和 textarea 标签	290

9.5.15	updownselect 标签	291
9.6	非表单标签	292
9.6.1	actionerror 和 actionmessage 标签	292
9.6.2	component 标签	293
9.6.3	tree 和 treenode 标签	294

第 10 章 Struts 2 数据库应用 296

10.1	MySQL 下载和安装	296
10.1.1	下载 MySQL 数据库	296
10.1.2	安装 MySQL 数据库服务器	297
10.1.3	下载 MySQL JDBC 驱动	298
10.2	访问 MySQL 数据库	298
10.2.1	创建一个数据库	298
10.2.2	建立 JDBC 连接	299
10.3	使用 Tomcat 数据源	300
10.4	Struts 2 数据分页	304
10.5	Struts 2 数据库操作	309
10.5.1	添加用户	310
10.5.2	修改用户	312
10.5.3	删除用户	316

第 11 章 文件上传与下载 319

11.1	文件上传	319
11.1.1	基于表单的文件上传设置	319
11.1.2	手动上传文件	322
11.1.3	Struts 2 对文件上传的支持	325
11.1.4	在 Struts 2 中实现文件上传	325
11.1.5	实现上传文件过滤	328
11.2	多个文件同时上传	330
11.2.1	使用数组同时上传多个文件	330
11.2.2	使用 List 同时上传多个文件	333
11.3	文件下载	335
11.3.1	Struts 2 实现文件下载	335
11.3.2	下载权限控制	337

第 12 章 避免表单重复提交与等待 页面 340

12.1	避免表单重复提交	340
12.2	Struts 2 的实现方式	341

12.3 使用 token 拦截器	342	13.7.4 getRemoteUser()方法	362
12.4 使用 tokenSession 拦截器	345	第 14 章 Struts 2 对 Ajax 的支持	363
12.5 自动显示等待页面	346	14.1 Ajax 快速入门	363
12.6 使用 execAndWait 拦截器	347	14.1.1 Ajax 简介	363
第 13 章 Struts 2 应用程序安全性	350	14.1.2 XMLHttpRequest	364
13.1 系统攻击的实现因素	350	14.2 Ajax 应用开发	370
13.2 Web 应用程序的攻击类型	350	14.2.1 使用 XML 作为数据的载体	370
13.3 用户和角色	351	14.2.2 使用 JSON 串作为数据的载体	373
13.4 编写安全策略	352	14.2.3 使用 Dojo 简化 Ajax 应用的开发	382
13.4.1 保护资源	352	14.3 在 Struts 2 中使用 JSON	385
13.4.2 指定登录方法	353	14.4 Struts 2 的 Ajax 标签	390
13.5 身份验证方法	354	14.4.1 Struts 2 的 Ajax 支持	390
13.5.1 基本身份验证方法	354	14.4.2 div 标签	394
13.5.2 基于表单的身份验证方法	356	14.4.3 a 标签	402
13.6 Struts 2 的安全配置	358	14.4.4 submit 标签	405
13.7 以编程方式提高安全性	359	14.4.5 tabbedPanel 标签	406
13.7.1 getAuthType()方法	359	14.4.6 autocompleter 标签	408
13.7.2 isUserInRole()方法	360		
13.7.3 getUserPrincipal()方法	361		

第三篇 Struts 2 整合其他技术篇

第 15 章 整合 Spring	413	16.1.3 Hibernate 核心接口	427
15.1 Spring 概述	413	16.2 Struts 2 整合 Hibernate	428
15.1.1 Spring 框架组件	413	16.2.1 下载安装 Hibernate	428
15.1.2 Spring 的优点	415	16.2.2 Struts 2 与 Hibernate 的整合方案	429
15.1.3 下载安装 Spring	415	16.2.3 功能设计与数据库设计	429
15.1.4 IoC 和 DI	415	16.2.4 Hibernate 映射对象到数据库	430
15.2 Struts 2 整合 Spring	418	16.2.5 hibernate.cfg.xml 配置文件	432
15.2.1 原理介绍	418	16.2.6 HibernateSessionFactory 类文件	433
15.2.2 整合步骤	418	16.2.7 数据库操作 DAO	434
15.2.3 整合示例	419	16.2.8 业务控制 Action	436
第 16 章 整合 Hibernate	426	16.2.9 struts.xml 配置文件	440
16.1 Hibernate 简介	426	16.2.10 视图文件	442
16.1.1 Hibernate 的优点	426	16.2.11 校验文件	446
16.1.2 理解 Hibernate 架构	427	16.2.12 运行程序	446

第 17 章 整合 JSF.....449

- 17.1 JSF 介绍449
 - 17.1.1 JSF 的优势449
 - 17.1.2 JSF 体系451
 - 17.1.3 与 Struts 2 比较451
 - 17.1.4 JSF 的简单应用452
 - 17.1.5 下载 MyFaces456
- 17.2 Struts 2 整合 MyFaces457
 - 17.2.1 整合原理457
 - 17.2.2 整合步骤457
 - 17.2.3 整合示例458

第 18 章 整合其他框架技术.....467

- 18.1 整合 SiteMesh467
 - 18.1.1 SiteMesh 简介467
 - 18.1.2 下载安装 SiteMesh468
 - 18.1.3 SiteMesh 的简单应用469

18.1.4 更复杂的装饰器页面472

18.1.5 Struts 2 中使用 SiteMesh475

18.2 整合 FreeMarker480

- 18.2.1 FreeMarker 简介480
- 18.2.2 FreeMarker 的简单应用481
- 18.2.3 FreeMarker 常用指令483
- 18.2.4 Struts 2 中使用 FreeMarker486

18.3 整合 Tiles491

- 18.3.1 Tiles 简介492
- 18.3.2 下载安装 Tiles492
- 18.3.3 Tiles 的简单应用493
- 18.3.4 Struts 2 中使用 Tiles497

18.4 整合 Struts 1502

18.5 整合 JFreeChart506

- 18.5.1 JFreeChart 简介507
- 18.5.2 下载安装 JFreeChart507
- 18.5.3 JFreeChart 的简单应用508
- 18.5.4 Struts 2 中使用 JFreeChart509

第四篇 Struts 2 综合实例篇**第 19 章 用户注册系统**.....515

- 19.1 系统概述515
 - 19.1.1 系统分析与设计515
 - 19.1.2 系统结构516
- 19.2 数据库设计517
- 19.3 DAO 组件518
 - 19.3.1 创建实体类518
 - 19.3.2 实现 DAO 组件519
- 19.4 用户注册模块522
 - 19.4.1 用户注册页面522
 - 19.4.2 Action 类524
 - 19.4.3 校验文件525
 - 19.4.4 struts.xml 文件527
 - 19.4.5 效果演示527
- 19.5 浏览用户信息模块528
 - 19.5.1 Action 类528

19.5.2 权限控制529

19.5.3 struts.xml 文件530

19.5.4 用户列表页面531

19.5.5 效果演示532

第 20 章 企业 BBS 系统.....533

- 20.1 系统分析和设计533
 - 20.1.1 需求分析533
 - 20.1.2 系统功能模块533
- 20.2 数据库设计535
- 20.3 系统实体及其关联关系536
- 20.4 实现 Hibernate 持久层536
 - 20.4.1 创建持久化类537
 - 20.4.2 映射持久化类538
 - 20.4.3 连接数据库541
- 20.5 实现 DAO 组件层542
 - 20.5.1 创建 SessionFactory542



20.5.2	实现 DAO	544
20.6	Web 层实现	550
20.6.1	回复列表的实现	550
20.6.2	对帖子的管理	553
20.6.3	JSP 嵌入文本编辑器	558
20.6.4	图形验证码的实现	559
第 21 章	电子相册系统	562
21.1	系统分析和设计	562
21.1.1	需求分析	562
21.1.2	系统功能模块	562
21.2	数据库设计	564
21.2.1	数据库建表	564
21.2.2	系统实体及其关联关系	566
21.3	实现 Hibernate 持久层	566
21.3.1	创建持久化类	566
21.3.2	编写持久化类的映射配置文件	568
21.3.3	配置 Hibernate 的 SessionFactory	571
21.4	实现 DAO 组件层	572
21.4.1	实现 DAO 组件	572
21.4.2	DAO 组件配置	577
21.5	实现业务逻辑层	578
21.5.1	实现业务逻辑组件	578
21.5.2	配置业务逻辑组件	582
21.6	实现 Web 层	583
21.6.1	列出相册列表	583
21.6.2	上传相片	586
21.6.3	查看相片列表	589
21.6.4	查看相片和发表相片评论	591

第一篇 Struts 2 基础篇

第 1 章 Struts 2 概述

全世界发布的第一个 MVC 框架是 Struts 1, 它是一个开源框架, 用于开发 Java Web 应用。经过多年的使用和完善, Struts 1 变得越来越成熟稳定, 成为世界上使用最广泛的 MVC 框架。与 Struts 1 相比, Struts 2 有着许多革命性的改进, 但它并不是一个全新的框架, 而是在 WebWork 框架的基础上发展起来的。本章将先对 Struts 1 与 WebWork 进行简单介绍, 最后介绍 Struts 2, 并对它们做简单的比较。

- 理解 MVC 模式
- 了解 Model 1 与 Model 2 体系
- 了解 Struts 1 框架
- 了解 WebWork 框架
- 熟悉 Struts 2 框架体系
- 掌握 Struts 2 配置文件
- 熟悉 Struts 2 控制器
- 熟悉 Struts 2 标签库

1.1 MVC 概述

模式-视图-控制器 (Model-View-Controller, MVC) 是一种目前广泛流行的软件设计架构模式。随着网络应用的快速增加, MVC 模式对于 Web 应用的开发已经成为一种非常先进的设计思想。

1.1.1 MVC 设计模式

MVC 设计模式的宗旨就是把一个 Web 应用的输入、处理和输出流程按照 Model、View 和 Controller 的方式进行分离, 这样一个应用就被分成 3 个层: 模型层、视图层和控制层。

□ 视图 (View)

视图代表用户交互界面。一个 Web 应用可能有很多不同的视图, MVC 设计模式对于视图的处理, 仅限于视图中数据的采集与处理以及用户请求的处理, 而不包括对视图中业务流程的处理。

□ 模型 (Model)

在 Web 应用中, 模型表示业务数据与业务逻辑, 它是 Web 应用的主体部分, 视图中的

业务数据由模型提供。



使用 MVC 设计模式开发 Web 应用，关键的一点就是让一个模型为多个视图提供业务数据，这样可以提高代码的可重用性与可读性，也给 Web 应用后期的维护带来方便。

□ 控制器 (Controller)

控制器是视图与模型之间的纽带。控制器将视图接收的数据交给相应的模型去处理，将模型的返回数据交给相应的视图去显示。

1. MVC 模块层关系

MVC 设计模式的 3 个模块层之间的关系如图 1-1 所示。

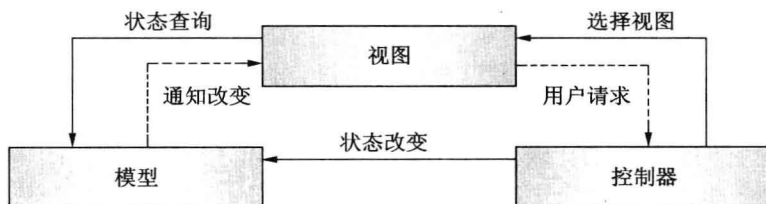


图 1-1 MVC 模块层的关系

2. MVC 处理过程

从前面介绍的内容中可以知道 MVC 模式处理请求的大致过程。MVC 具体的处理过程时序图如图 1-2 所示。

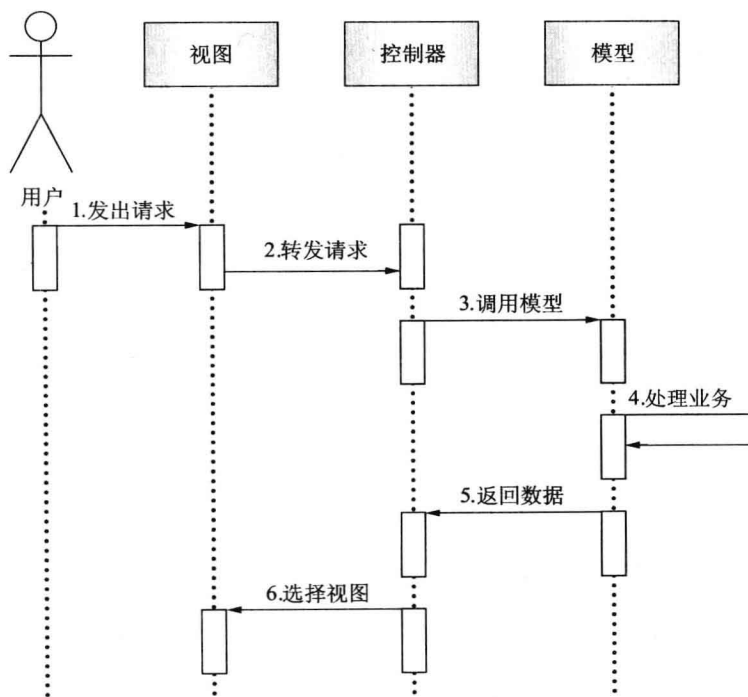


图 1-2 MVC 处理过程时序图

MVC 的处理过程从用户发出请求开始，用户发出请求后，视图将请求转发到控制器，控制器调用模型，由模型对业务进行处理，模型完成业务处理后向控制器返回处理结果，控制器根据返回数据选择相应的视图反馈给用户。

1.1.2 MVC 的优点

MVC 模式的优点主要有以下几个方面。

□ 低耦合性

视图层、模型层和控制层分离，这样就允许在不重新编译模型和控制器代码的情况下更改视图层代码。同样，改变一个 Web 应用的业务流程或者业务规则时，只需要改变 MVC 的模型层即可。



因为模型、控制器和视图相分离，所以可以很容易改变应用程序的数据层和业务规则。

□ 高重用性

多个视图可以对应一个模型，这样可以减少代码的重复性及代码的维护量，一旦模型发生改变，也易于维护。

□ 可适用性

随着技术的不断进步，现在需要用越来越多的方式来访问应用程序。MVC 模式允许使用各种不同的表现层技术（如 JSP、Velocity 等）来访问同一个服务器端的代码。

□ 可维护性

MVC 的 3 个模块层相分离，使得 Web 应用更易于维护和修改。

1.1.3 Model 1 和 Model 2 体系介绍

在开发 Web 应用时，经常提到 Model 1 和 Model 2，它们是对采用 JSP 技术构成 Web 应用的不同模型的描述。下面简单介绍 Model 1 和 Model 2。

□ Model 1

在一个 JSP 文件中，不但可以包含 HTML 代码，还可以包含业务逻辑、服务端处理的实现代码。JSP 页面中可以非常容易地结合业务逻辑、服务端处理过程和 HTML。在 JSP 页面中同时实现数据显示、业务逻辑和流程控制，可以快速地完成应用开发。现在很多 Web 应用就是由一组 JSP 页面构成的。这种以 JSP 为中心的开发模型可以称之为 Model 1。



Model 1 开发模型的不足之处在于，应用的实现一般是基于过程的，一组 JSP 页面实现一个业务流程，如果要对业务处理进行改动，必须在多个地方进行修改，这样非常不利于应用扩展和更新。

Model 1 体系结构如图 1-3 所示。

□ Model 2

Model 2 表示的是基于 MVC 模式的框架。通过 MVC 这种设计模型把应用逻辑、处理过

程和显示逻辑分成不同的组件实现，这些组件可以进行交互和重用，从而弥补 Model 1 的不足。

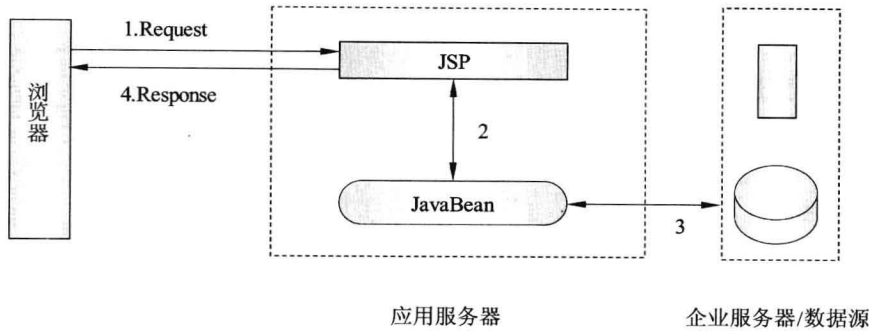


图 1-3 Model 1 体系结构



Model 2 具有组件化的优点，从而使得它更易于实现对大规模系统的开发和管理。

Model 2 体系结构如图 1-4 所示。

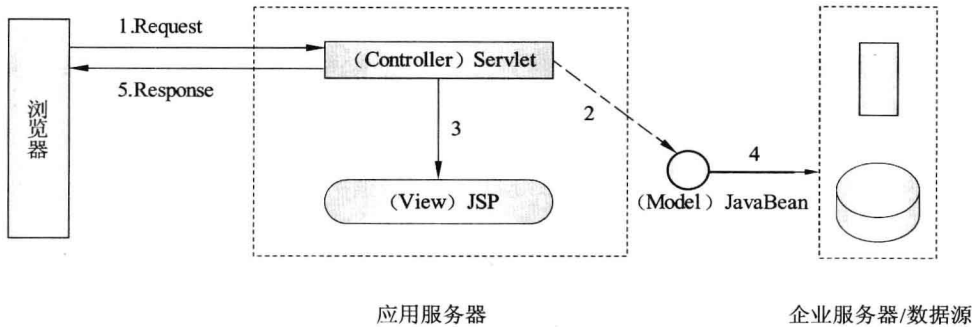


图 1-4 Model 2 体系结构

在 Model 2 架构中，Servlet 作为前端控制器，负责接收客户端发送的请求，在 Servlet 中只包含控制逻辑和简单的前端处理。然后，调用后端 JavaBean 来完成实际的逻辑处理。最后，转发到相应的 JSP 页面处理显示逻辑。



在 Model 2 模式下，模型（Model）由 JavaBean 充当；视图（View）由 JSP 页面充当；控制器（Controller）由 Servlet 充当。

1.2 Struts 1 简介

Struts 1 最早是作为 Apache Jakarta 项目的组成部分，项目的创立者希望通过对该项目的

研究,从而改进和提高 JSP、Servlet、标签库以及面向对象的技术水准。

1.2.1 Struts 1 框架介绍

Struts 1 是一个为开发基于 MVC 模式的应用架构的开源框架,是利用 Java Servlet 和 JSP 构建 Web 应用的一项非常有用的技术。Struts 1 能充分满足应用开发的需求,而且简单易用。



Struts 1 把 Servlet、JSP、自定义标签和信息资源整合到一个统一的框架中,开发人员利用其进行开发时,不用再自己编码实现全套 MVC 模式,极大地节省了开发时间。

1. Struts 1 的主要组成部分

Struts 1 框架主要由以下几个主要部分组成。

□ 模型 (Model)

Struts 1 的 Model 部分主要由底层的业务逻辑组件充当,这些业务逻辑组件封装了底层数据库访问和业务逻辑方法实现。Struts 1 框架分为如下组成部分:

➤ 模型 (Model)

Struts 1 的 Model 部分主要由底层的业务逻辑组件充当,这些业务逻辑组件封装了底层数据库的访问和业务逻辑方法的实现。Struts 1 的 Model 主要是 FormBean, FormBean 可以封装客户的表单数据, JSP 文件使用 Struts 1 标签将客户数据传递至 FormBean。所有的 FormBean 都必须继承 ActionForm 类,以便于对表单数据进行初始化以及验证。

➤ 视图 (View)

Struts 1 中的 View 部分采用 JSP 实现。不过, Struts 1 提供了丰富的标签库,借助这些标签库,可以最大限度地减少 Java 脚本的使用。

➤ 控制器 (Controller)

Controller 是 Struts 1 框架中的核心组件,由 ActionServlet 类实现。Struts 1 的 Controller 由如下两部分组成:

- (1) 系统核心控制器,指系统中的 ActionServlet,由 Struts 1 框架提供。
- (2) 业务逻辑控制器,指用户自己实现的 Action 实例,由 Struts 1 框架提供。



ActionServlet 继承了 Javax.Servlet.http.HttpServlet 类,主要负责接收 HTTP 请求信息,根据配置文件 struts-config.xml 的配置信息,将请求转发给适当的 Action 对象。

2. Struts 1 体系结构

Struts 1 框架体现了典型的 MVC 设计模式的特点,其体系结构如图 1-5 所示。

Struts 1 的 Model 部分主要由底层的业务逻辑组件充当,这些业务逻辑组件封装了底层数据库的访问、业务逻辑方法的实现。

Struts 1 的 View 部分采用 JSP 实现,通过自定义标签库可以输出控制器的处理结果。

Struts 1 的控制器部分由系统核心控制器 (ActionServlet) 和业务逻辑控制器 (Action) 组成。系统核心控制器负责拦截所有 HTTP 请求,然后根据用户请求决定是否调用业务逻辑控制器。业务逻辑控制器负责处理用户请求,其本身不具有处理能力,而是调用 Model