

Struts 2

技术详解

基于WebWork核心的MVC开发与实践

知程序 行天下 行天下 知程序

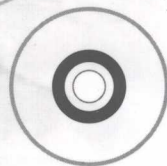
闫术卓 杨强 等编著

程序天下



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

基础+主流框架整合开发+案例



程序天下

知程序 行天下

知程序 行天下

闫木卓 杨强 等编著

Struts 2

技术详解

基于WebWork核心的MVC开发与实践

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

Struts 2 框架是 Struts 1.X 的替代版本, Struts 2 框架整合了 Struts 1.X 框架和 WebWork 框架的优点。相对 Struts 1.X, Struts 2 已经有了非常大的改变, 去掉了 ActionForm, 降低了框架组件之间的耦合性, Struts 2 的 Action 只是普通的 Java 类 (POJO), 给模块测试工作带来了极大的方便。Struts 2 提供了强大的整合能力, 支持多种返回结果类型, 改进了 Struts 1.X 的标签库, 引入 OGNL 表达式和值栈的概念, 给开发者带来了更好的体验。

本书结合 Struts 2 框架的技术特点, 从最基础的框架处理机制讲起。介绍了 Struts 2 框架的核心组件和核心处理机制, 并介绍了拦截器、国际化、输入校验、类型转换等 Struts 2 关键技术, 同时介绍了如何在 Struts 2 框架中整合其他开源技术。

本书非常适合有过 Struts 1.X 和 WebWork 开发经验的读者, 以及 Java Web 开发的初学者, 对于 Web 高级开发者, 也是一本非常好的参考书。

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

Struts 2 技术详解: 基于 WebWork 核心的 MVC 开发与实践 / 闫术卓等编著. —北京: 电子工业出版社, 2008.6

(程序天下)

ISBN 978-7-121-06221-6

I. S… II. 闫… III. 软件工具—程序设计 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2008) 第 036265 号

责任编辑: 葛 娜

印 刷: 北京东光印刷厂

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: 40.75 字数: 965 千字

印 次: 2008 年 6 月第 1 次印刷

印 数: 5000 册 定价: 69.80 元 (含光盘 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

Struts 2 框架作为 Struts 1.X 框架的替代技术, 相对 Struts 1.X 来说, 有着本质上的改变, Struts 2 框架是从 WebWork 框架发展而来的。Struts 1.X 已经成为 Java Web 开发中应用最广泛的 MVC 框架, 被广大的程序员所接受。随着 Web 技术的发展, Struts 1.X 越来越难以适应, 主要体现在支持视图技术单一、难于测试性、庞大的配置文件和大量的 ActionForm 类, 给项目开发的管理维护带来了困难。

Struts 2 框架技术整合了 Struts 1.X 和 WebWork 的优点, Struts 2 框架最大的特点就是其简单性, 引入了 OGNL 表达式和值栈的概念, 可以使开发者使用简单的代码实现复杂的数据访问; Action 实现类就是一个标准的 Java 类 (POJO), 这使得测试工作变得简单。同时, Struts 2 框架取消了 ActionForm, 支持多种返回类型和 Ajax 技术, 简化了同其他技术的整合。从 Apache 官方的宣传来看, Struts 2 不久将会替代 Struts 1.X。

目前, 国内关于 Struts 2 框架介绍方面的书籍比较少, 本书从最基本的处理机制和核心组件讲起, 全面介绍 Struts 2 框架技术, 由浅入深地引导读者掌握 Struts 2 技术, 书中的示例简单易懂, 重点突出, 给读者提供一个良好的学习参考。

由于作者水平有限, 如果读者发现问题或者错误之处, 欢迎相互交流和探讨。

本书的内容安排

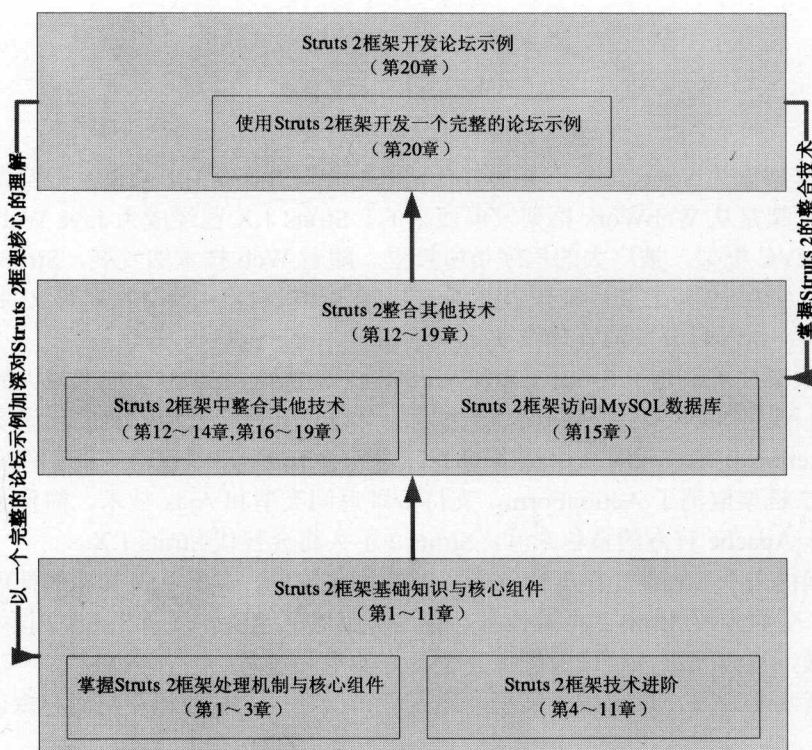
本书共分为 3 个部分, 从最基础的 Struts 2 框架处理机制讲起, 接着深入讲述整合 JSF、Spring 等技术, 最后给出一个完整的论坛代码示例。读者通过这三个部分的学习, 能够全面掌握 Struts 2 框架的精髓。

第一部分 (第 1~11 章) 讲述了 Struts 2 框架的核心处理机制, 介绍了 Struts 2 的核心组件、国际化、输入校验、类型转换、拦截器和 Ajax 支持。每个章节都提供了详细的示例代码, 供读者学习参考。

第二部分 (第 12~19 章) 重点讲述了如何在 Struts 2 框架中整合其他流行技术, 如 JSF、Spring 和 Hibernate 等。在第 15 章介绍了在 Struts 2 框架中如何访问 MySQL 数据库服务器, 每个知识点笔者都提供了详细的示例代码供读者参考。

第三部分 (第 20 章) 是对前面所有章节的一个总结和提高, 笔者引导读者通过一个示例论坛的开发来巩固前面所学到的知识。更加重要的是该部分介绍了一个完整的应用项目开发组织和开发管理过程, 在读者学习体会 Struts 2 精彩之处的同时了解项目开发模式, 对初学者尤其重要。

读者可以遵循本书的写作顺序, 参考下面流程图阅读本书。



本书的特点

笔者试图通过本书的讲解,使读者能够理解 Struts 2 框架技术的精髓,为了避免教科书式的枯燥,在书中加入了许多重点标注和小结,每个代码示例均紧扣知识点,简单易懂。

本书具有如下特点:

- ❑ 本书的编排采用循序渐进的方式,适合初级、中级学者逐步掌握 Struts 2 框架技术。
- ❑ 本书重点讲解 Struts 2 框架核心组件,帮助读者真正理解 Struts 2 的精髓。
- ❑ 本书采用了大量的实例,讲述 Struts 2 框架的知识点。
- ❑ 所有实例都具有代表性和实际意义,着重解决工作中的实际问题。
- ❑ 对于有特点的实例进行详细的解释和分析,帮助读者理解和模拟实践。
- ❑ 引导读者理解规范的代码开发,对软件开发初学者尤其有用。
- ❑ 按递进关系进行案例组织,使用新旧的知识点时相互关联,对比分析易于理解。
- ❑ 本书采用技术要点、详细介绍、运行效果等多种方式进行讲解,系统性强、可用性 强。

本书配有学习论坛:论坛网址是 <http://www.rzchina.net>,读者可以在上面讨论技术,

笔者会及时回答读者提问，并提供各种技术文章，帮助你提高开发水平。

适合阅读本书的读者

- ☐ Java 初学者；
- ☐ Struts 1.X 开发者；
- ☐ J2EE 开发者；
- ☐ Java Web 开发的项目管理者；
- ☐ 将要使用 MVC 设计的程序员；
- ☐ 想了解 Struts 2 框架技术的其他人员。

作 者

2008 年 4 月

目 录

Contents

第一篇 Struts 2 零基础详解

第 1 章 Struts 2 概述

2

Struts 1 是一个开放源码的框架,用来开发 Java Web 应用。Struts 1 最初由程序员 Craig R. McClanahan 发布,Apache 软件基金会于 2002 年接手该项目。Struts 1 提供了一个极好的框架,可以容易地整合 JSP、Servlet 和 Java 代码。随着 Java 的发展,Struts 1 已经成为 MVC 设计模式的常用架构,并被广大的程序员所接受,在 Web 开发应用中获得了巨大成功,成为使用最早、应用最广的 MVC 架构。

1.1 MVC 介绍	3
1.1.1 Web 技术发展	4
1.1.2 MVC 设计模式	6
1.1.3 MVC 的处理过程	7
1.1.4 MVC 的优点	8
1.1.5 MVC 的适用范围	8
1.1.6 Model 1 和 Model 2 体系介绍	9
1.2 Struts 1 简介	10
1.2.1 Struts 1 框架介绍	10
1.2.2 web.xml 配置文件	11
1.2.3 struts-config.xml 配置文件	12
1.2.4 Action 和 ActionForm	12
1.2.5 Struts 1 处理过程	13
1.2.6 Struts 1 的优点	14
1.2.7 Struts 1 的缺点	14
1.3 WebWork 简介	15
1.3.1 WebWork 框架	15
1.3.2 WebWork 的特性	18
1.4 Struts 2 概述	18
1.4.1 Struts 2 框架	19
1.4.2 Struts 2 配置文件	20
1.4.3 Struts 2 控制器	24
1.4.4 Struts 2 标签库	26
1.4.5 Struts 2 与 Struts 1 比较	28
1.4.6 Struts 2 与 WebWork 比较	29
1.5 为什么使用 Struts 2	30
1.6 本章小结	30

HelloWorld 例程已经成为学习软件开发技术的默认入门标准步骤,笔者也不能免俗,在本章内容中,将会引导读者逐步进入 Struts 2 的开发世界。本章将详细介绍 Struts 2 的下载安装、开发环境的搭建和使用,并且以一个 HelloWorld 例程,体验 Struts 2 带来的设计风格。为了避免读者对枯燥乏味的技术描述产生厌倦情绪,笔者将一步一步引导读者学习 Struts 2,希望能给读者一个轻松愉快的学习氛围。

2.1 搭建 Struts 开发环境	33
2.1.1 安装 JDK	33
2.1.2 安装 Eclipse	35
2.1.3 安装 Tomcat	36
2.1.4 安装 Struts 2	37
2.1.5 Eclipse 安装 Struts 2	40
2.2 一个简单的 HelloWorld	43
2.2.1 配置 web.xml 文件	43
2.2.2 配置 struts.xml 文件	44
2.2.3 Action 业务控制器	45
2.2.4 视图资源	46
2.2.5 运行 HelloWorld	48
2.2.6 HelloWorld 小结	49
2.3 Struts 2 特性演示	49
2.3.1 基本实现	49
2.3.2 标签库	54
2.3.3 改进 Action	56
2.3.4 国际化	58
2.3.5 数据校验:使用 validate()方法校验	62
2.3.6 数据校验:使用框架校验	64
2.4 本章小结	66

第3章 Struts 2 核心剖析

上一章介绍了一个 HelloWorld 应用和一个用户注册应用,读者通过自己编写代码,对 Struts 2 有了一个基本的了解。在介绍用户注册应用时,突出展示了 Struts 2 的特点,读者可能对有些方面是比较迷惑的。那么 Struts 2 是如何工作的?配置文件、Action 又有哪些具体的要求呢?笔者在本章里面将试图说明这些问题。通过本章的学习,读者会对 Struts 2 有更深层次的理解。

3.1 Struts 2 工作流程	68
3.1.1 核心控制器 FilterDispatcher	68
3.1.2 业务控制器 Action	73
3.1.3 业务模型组件	73
3.1.4 视图组件	76
3.2 Struts 2 配置文件	76
3.2.1 配置 web.xml 文件	77

3.2.2	配置 struts.xml 文件	81
3.2.3	常量配置	89
3.2.4	包配置	92
3.2.5	命名空间配置	94
3.2.6	包含配置	96
3.2.7	Bean 配置	99
3.2.8	拦截器配置	101
3.2.9	配置 struts.properties 文件	103
3.2.10	配置通配符	106
3.2.11	Struts 2 的零配置	107
3.3	Struts 2 的 Action	109
3.3.1	Action 实现类	109
3.3.2	实例验证: Action 属性和用户参数之间的关系	114
3.3.3	Action 访问 ActionContext	118
3.3.4	值栈 (ValueStack)	122
3.3.5	Action 直接访问 Servlet API	127
3.3.6	Action 的配置	131
3.3.7	Action 的动态方法调用	134
3.3.8	通配符配置	139
3.4	处理结果	145
3.4.1	处理结果流程	145
3.4.2	配置 result	146
3.4.3	result 的类型	147
3.4.4	action-chain 类型示例	150
3.4.5	Freemarker 类型示例	155
3.4.6	redirect 类型示例	155
3.4.7	redirect-action 类型示例	156
3.4.8	Stream 类型示例	157
3.4.9	使用通配符动态配置 result	158
3.4.10	使用 OGNL 动态配置 result	161
3.5	模型驱动	164
3.5.1	模型驱动的意义	164
3.5.2	模型驱动示例	167
3.6	异常处理	172
3.6.1	Java 的异常处理	172
3.6.2	Struts 2 框架的异常处理	175
3.6.3	异常的配置	177
3.7	如何提高 Struts 2 性能	181
3.8	支持 Struts 2 框架的应用服务器	181
3.9	本章小结	182

第二篇 Struts 2 框架技术

第 4 章 国际化

184

互联网的出现,给整个世界范围内的交流和沟通提供了方便快捷的方式,人们之间开始用电子邮件来代替传统的纸质信件,各种各样的网站和工具使得人们在不同的国度或者区域成为交谈的对象。

而这些技术都离不开软件技术的发展,一个软件或者系统要让全世界用户都能接受和使用成了广大软件公司梦寐以求的事情。由于世界各国和地区使用的语言和其他表达形式的不同,制约了一个单一语种界面的软件的应用,所以,软件的国际化已经成为了软件开发技术中重要的一个部分。

4.1 软件的国际化	185
4.2 Java 的国际化支持	186
4.2.1 使用资源文件	189
4.2.2 使用资源类文件	191
4.2.3 MessageFormat 类	193
4.3 Struts 2 的国际化支持	197
4.3.1 配置资源文件	197
4.3.2 Struts 2 国际化应用	199
4.3.3 使用占位符	204
4.3.4 包范围资源文件	208
4.3.5 Action 范围资源文件	210
4.3.6 临时资源文件	211
4.3.7 加载资源文件的顺序	213
4.4 动态访问国际化资源文件	214
4.4.1 动态访问国际化资源文件原理	214
4.4.2 建立资源文件和配置文件	215
4.4.3 建立 Action 和 JSP	216
4.4.4 动态访问资源文件示例	217
4.5 Eclipse 编写资源文件的插件	218
4.6 本章小结	218

第 5 章 Struts 2 进阶——拦截器

219

拦截器 (Interceptor) 是 Struts 2 的核心组成部分。很多功能 (Feature) 都是构建在拦截器基础之上的,例如文件的上传和下载、国际化、转换器和数据校验等, Struts 2 利用内建的拦截器,完成了框架内的大部分操作。

在 Struts 2 文档中对拦截器的解释为——拦截器是动态拦截 Action 调用的对象。它提供了一种机制,使开发者可以定义一个特定的功能模块,这个模块可以在 Action 执行之前或者之后运行,也可以在一个 Action 执行之前阻止 Action 执行。同时也提供了一种可以提取 Action 中可重用的部分的方式。

5.1 拦截器介绍	220
5.1.1 AOP 介绍	220

5.1.2	拦截的实现原理	223
5.1.3	拦截的意义	225
5.2	Struts 2 拦截器	226
5.2.1	Struts 2 拦截器原理	226
5.2.2	HelloWorld 拦截器	228
5.2.3	定义拦截器	231
5.2.4	使用拦截器	234
5.2.5	默认拦截器	235
5.3	自定义拦截器	238
5.3.1	自定义拦截器实现类	238
5.3.2	使用自定义拦截器	240
5.4	拦截器深度剖析	243
5.4.1	拦截器的方法过滤	243
5.4.2	拦截器的执行顺序	248
5.4.3	拦截结果监听器	251
5.4.4	设置拦截器栈中拦截器参数	254
5.4.5	Struts 2 框架的系统拦截器	256
5.5	拦截器应用示例	261
5.5.1	权限拦截器	261
5.5.2	配置拦截器	263
5.5.3	业务控制器 Action	265
5.5.4	JSP 视图	266
5.5.5	运行示例	267
5.6	本章小结	268

第 6 章 Struts 2 的类型转换

270

在 MVC 模型中，视图层一般用于用户输入参数或者输出数据信息，而控制层将视图层和模型层联系起来，将视图层输入的数据传递给模型层，或者是将模型层处理结果返回给视图层。数据在三个层间传递的时候，会涉及数据类型的转换，因为 Java 是强制类型的语言，所以使用的变量和属性必须强制指定其类型。

一般 Web 应用中，视图层用户输入的都是字符串，例如生日、金额等数据都是以字符串类型输入的，业务逻辑模块要处理用户输入的数据，就必须对输入的字符串进行转换，转换为相应的 Java 类型。

6.1	类型转换	271
6.2	编写一个类型转换器	274
6.2.1	类型转换需求	274
6.2.2	编写自定义类型转换器	276
6.2.3	视图资源文件	277
6.2.4	运行示例	278
6.3	自定义类型转换器	278

6.3.1	基于 OGNL 的类型转换器	278
6.3.2	基于 Struts 2 的类型转换器	279
6.3.3	注册自定义类型转换器	281
6.3.4	数组属性类型转换器	284
6.3.5	集合属性类型转换器	287
6.4	使用 Struts 2 的类型转换	289
6.4.1	Struts 2 系统内建的类型转换器	290
6.4.2	项目应用中常见的类型转换	290
6.4.3	使用 OGNL 表达式	292
6.4.4	使用集合类型属性	294
6.4.5	使用 Set 类型属性	297
6.5	类型转换中的异常处理	300
6.5.1	类型转换异常拦截器	300
6.5.2	一个简单的类型转换异常处理	301
6.5.3	改进类型转换异常显示信息	303
6.5.4	处理复合类型转换异常	304
6.6	本章小结	305

第 7 章 输入校验

307

几乎每一个 Web 开发者都会遇到输入校验的问题，输入校验直接影响了系统的抗破坏性能和稳定性。由于采用 B/S 结构，用户使用浏览器同系统进行交互，由于浏览器的限制，不能保证用户按照开发者的意图来输入正确的数据。而往往有的用户喜欢恶意地输入非法数据进行测试和破坏，所以必须对用户输入的信息进行输入校验。良好的输入校验功能是一个成熟项目应用的必备条件。

实际上，输入校验往往同类型转换联系在一起，都是对用户输入数据的规范化检查和处理，Struts 2 框架为开发者的输入校验提供了良好的支持。

7.1	输入校验介绍	308
7.1.1	输入校验的原因	308
7.1.2	使用 JavaScript 完成客户端校验	309
7.2	使用 validate 方法进行输入校验	311
7.2.1	validate()方法输入校验	311
7.2.2	validateXxx()方法输入校验	316
7.2.3	输入校验流程	318
7.3	基于框架的输入校验	318
7.3.1	使用字段校验的输入校验	319
7.3.2	使用客户端的输入校验	321
7.3.3	使用非字段校验的输入校验	324
7.3.4	输入校验的国际化信息	326
7.3.5	校验的搜索顺序	329
7.4	AJAX 输入校验	333
7.4.1	配置 AJAX 环境	334

7.4.2	建立业务控制器	336
7.4.3	建立校验规则文件	336
7.4.4	建立 JSP 视图	337
7.4.5	运行 AJAX 校验示例	338
7.5	复合类型属性的输入校验	338
7.5.1	复合属性的校验	339
7.5.2	集合属性的校验	342
7.6	Struts 2 框架的校验器	343
7.6.1	内建校验器	343
7.6.2	类型转换校验器	345
7.6.3	日期校验器	345
7.6.4	浮点数值校验器	346
7.6.5	邮件地址校验器	346
7.6.6	表达式校验器	347
7.6.7	字段表达式校验器	347
7.6.8	整数校验器	348
7.6.9	正则表达式校验器	348
7.6.10	必填校验器	349
7.6.11	必填字符串校验器	349
7.6.12	字符串长度校验器	350
7.6.13	网址校验器	350
7.6.14	visitor 校验器	351
7.7	本章小结	351

第 8 章 OGNL

352

OGNL 是 Object GraphicNavigation Language (对象图导航语言) 的缩写, OGNL 是一个开源项目, 读者可以访问其官方网站 www ognl.org 以获得源代码和相关资料。OGNL 是一种功能强大的 EL (Expression Language, 表达式语言), 可以通过简单的表达式来访问 Java 对象中的属性。

OGNL 先在 WebWork 项目中得到应用, 也是 Struts 2 框架视图默认的表达式语言, 可以说, OGNL 表达式是 Struts 2 框架的特点之一。

8.1	OGNL 基础知识	353
8.1.1	OGNL 的优势	353
8.1.2	OGNL 语法	353
8.1.3	一个使用 OGNL 的示例	354
8.2	Struts 2 的 OGNL	356
8.2.1	Struts 2 的 OGNL 表达式	357
8.2.2	OGNL 的集合操作	358
8.2.3	Lambda 表达式	359
8.3	Struts 2 中使用 OGNL	360

8.3.1 业务控制器	360
8.3.2 JSP 视图	361
8.3.3 运行示例	362
8.3.4 OGNL 中的#、%和\$符号	363
8.4 本章小结	364

第 9 章 Struts 2 标签库

365

在早期的 Web 开发中, JSP 的视图控制和显示技术主要依靠 Java 脚本来实现, 这样一来, JSP 页面重新嵌入了大量的 Java 脚本代码, 给开发带来了极大的不便。

从 JSP 1.1 规范以后, JSP 增加了自定义标签库的支持。标签库是一种组件技术, 通过标签库, 可以将复杂的 Java 脚本代码封装在组件中, 开发者只需要使用简单的代码就可以实现复杂的 Java 脚本功能。提供了 Java 脚本的复用性, 提高了开发者的开发效率。

Struts 2 标签库相对 Struts 1.x 进行了巨大的改进, 支持 OGNL 表达式, 不再依赖于任何表现层技术。

9.1 Struts 2 标签库概述	366
9.1.1 标签库简介	366
9.1.2 Struts 2 标签库组成	367
9.1.3 Struts 2 标签库的使用	368
9.1.4 Struts 2 同 Struts 1 标签库的比较	370
9.2 控制标签	370
9.2.1 if/elseif/else 标签	371
9.2.2 iterator 标签	372
9.2.3 append 标签	375
9.2.4 generator 标签	377
9.2.5 merge 标签	379
9.2.6 subset 标签	381
9.2.7 sort 标签	383
9.3 数据标签	385
9.3.1 action 标签	386
9.3.2 bean 标签	388
9.3.3 date 标签	390
9.3.4 debug 标签	393
9.3.5 include 标签	393
9.3.6 param 标签	395
9.3.7 push 标签	395
9.3.8 set 标签	396
9.3.9 url 标签	397
9.3.10 property 标签	399
9.4 主题与模板	399
9.4.1 主题	399

9.4.2 模板	400
9.5 表单标签	401
9.5.1 表单标签通用属性	401
9.5.2 checkbox 标签	403
9.5.3 checkboxlist 标签	404
9.5.4 combobox 标签	406
9.5.5 doubleselect 标签	407
9.5.6 datetimepicker 标签	409
9.5.7 head 标签	411
9.5.8 file 标签	411
9.5.9 hidden 标签	412
9.5.10 select 标签	412
9.5.11 optiontransferselct 标签	414
9.5.12 radio 标签	415
9.5.13 optgroup 标签	416
9.5.14 token 标签	417
9.5.15 textarea 标签	418
9.5.16 updownselct 标签	419
9.5.17 password 标签	420
9.5.18 textfield 标签	420
9.6 非表单标签	420
9.6.1 actionerror 和 actionmessage 标签	421
9.6.2 component 标签	422
9.6.3 tree 和 treenode 标签	423
9.7 本章小结	424

第 10 章 AJAX 技术支持

425

AJAX 技术是目前流行的技术，备受开发者和相关人员的关注。传统 B/S 结构的 Web 应用，都是采用了请求-等待-响应的交互流程，同 C/S（客户端/服务器端）的应用相比较，存在着许多不足。AJAX 技术的出现，改变了 B/S 结构应用的用户体验，该技术使用异步方式同服务器端进行交互，用户不用等待服务器的响应，极大地改善了用户体验。

Struts 1.x 并不支持 AJAX 技术，而 Struts 2 框架提供了完善而简洁的 AJAX 支持，Struts 2 框架提供了 ajax 主题的标签库，本章笔者将会介绍相关内容。

10.1 AJAX 介绍	426
10.1.1 为什么使用 AJAX	426
10.1.2 AJAX 技术	427
10.1.3 常见的浏览器端 AJAX 框架	429
10.1.4 常见的服务器端 AJAX 框架	431
10.2 Struts 2 的 AJAX 支持	432
10.2.1 ajax 主题	433

10.2.2	AJAX 输入校验	435
10.2.3	div 标签	439
10.2.4	a 标签	444
10.2.5	submit 标签	446
10.2.6	autocompleter 标签	448
10.2.7	tabbedPanel 标签	452
10.2.8	AJAX 表单	455
10.2.9	widgets	456
10.3	本章小结	458

第 11 章 文件的上传与下载

459

目前大部分 Web 应用都需要用到文件的上传与下载功能。最简单的文件上传例如修改用户自定义显示的图标, 以及复杂的办公文档处理, 都需要系统提供文件的上传服务。Struts 2 框架中内置了文件上传组件, 开发者可以很容易地开发出文件上传功能。笔者在本章将介绍 Struts 2 框架提供的文件上传与下载控制功能。

11.1	Struts 2 框架的文件上传	460
11.1.1	Common-fileUpload 组件	460
11.1.2	文件上传的 JSP	461
11.1.3	文件上传的 Action	462
11.1.4	配置文件和 success 视图	464
11.1.5	运行文件上传示例	465
11.1.6	上传文件的过滤	465
11.1.7	文件上传的常量	467
11.2	多个文件上传	468
11.2.1	使用数组上传多个文件	468
11.2.2	使用 List 上传多个文件	472
11.3	Struts 2 控制文件下载	475
11.3.1	在配置文件中指定下载资源	475
11.3.2	在 Action 中指定下载资源	476
11.3.3	文件下载的权限控制	478
11.4	本章小结	480

第三篇 Struts 2 框架中整合其他技术

第 12 章 使用 FreeMarker 技术

482

FreeMarker 是一个模板引擎, 一个基于文本的模板输出工具(生成任意的 HTML 表单代码)。它是一个 Java package, 面向 Java 程序员的 class library。它本身并不是针对最终用户的应用, 而是允许程序员将其嵌入到其产品中。

12.1	FreeMarker 介绍	483
12.1.1	FreeMarker 基础	483

12.1.2 FreeMarker 简单示例	483
12.2 Struts 2 中使用 FreeMarker	485
12.2.1 FreeMarker 使用 Struts 2 标签库	485
12.2.2 FreeMarker 访问 Servlet 和 JSP 对象	487
12.2.3 使用 FreeMarker 示例	488
12.3 本章小结	490

第 13 章 整合 Spring

491

Spring 和 Struts 一样都是一种轻量级的 J2EE 应用程序框架，Struts 注重的是降低表现和逻辑耦合程度，主要是把业务逻辑和表现层分开，但是不涉及业务层与持久层的关联。Spring 主要是对业务层的细化，也就是更深层次地降低了耦合程度，核心是一个轻量级的容器 container，实现了 IoC 模式的容器。Spring 的目标是实现一个全方位的整合框架。

13.1 Spring 介绍	492
13.1.1 IoC 和 DI	492
13.1.2 Spring 优点	495
13.2 Struts 2 整合 Spring	495
13.2.1 整合步骤	495
13.2.2 整合原理	497
13.3 整合开发示例	498
13.4 本章小结	501

第 14 章 整合 JSF

502

JavaServer Faces (JSF) 是一种刚刚发展起来的技术，是用来构建 Web 应用程序的新标准 Java 框架。它提供了一种以组件为中心来开发 Java Web 用户界面的方法，从而简化了开发过程，提高了开发效率。JavaServer Faces 还引起了广大 Java/Web 开发人员的兴趣。“企业开发人员”和 Web 设计人员将发现 JSF 开发可以简单到只需将用户界面 (UI) 组件拖放到页面上，而“系统开发人员”将发现丰富而强健的 JSF API 为他们提供了无与伦比的功能和编程灵活性。JSF 还通过将良好构建的模型-视图-控制器 (MVC) 设计模式集成到它的体系结构中，确保了应用程序具有更高的可维护性。最后，由于 JSF 是通过 Java Community Process (JCP) 开发的一种 Java 标准，因此开发工具供应者完全能够为 JavaServer Faces 提供易于使用的、高效的可视化开发环境。

14.1 JSF 介绍	503
14.1.1 JSF 体系结构	503
14.1.2 JSF 同 Struts 2 的比较	504
14.1.3 MyFaces	504
14.2 Struts 2 整合 MyFaces	505
14.2.1 整合步骤	505
14.2.2 整合原理	506
14.3 整合 MyFaces 示例	508
14.3.1 JSP 视图	508