



Struts 2.x

权威指南

第3版

李刚 编著

疯狂源自梦想

技术成就辉煌

疯狂源自梦想
技术成就辉煌



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

光盘包含20小时教学视频+代码





Struts 2.x

权威指南

第3版

李刚 编著

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书是《Struts 2 权威指南》的第 3 版，本书介绍了 Struts 2 最新版本：Struts 2.3.1.2。

本书全面介绍了 Struts 2 框架的各知识点，从 Struts 2 框架的 MVC 原理、Struts 2 的处理机制开始讲起，全面覆盖了 Struts 2 的 Action 开发、Action 配置管理、Result 配置管理、Struts 2 异常机制、Struts 2 类型转换、输入校验、文件上传和下载、Struts 2 国际化支持、Struts 2 的 Ajax 支持、拦截器机制、Convention 插件与 REST 等方面，并详尽讲解了 Struts 2 与 Spring、Hibernate、JSF、SiteMesh、JasperReports、JFreeChart 等框架的整合开发，力求让读者不仅可以掌握 Struts 2 理论知识，而且能真正步入实际开发殿堂。

本书保留了前两版通俗易懂的写作风格：按 Struts 2.3.1.2 的架构体系，细致地介绍了 Struts 2 各个知识点。在介绍过程中，笔者依照读者的学习规律，首先介绍基本概念和基本操作，然后对内容进行深入讲解。在介绍 Struts 2 框架时，既照顾到 Struts 2 初学者的能力，对每个知识点都给出简单的快速上手的示例；同时也考虑到开发者在日后开发中可能遇到的问题，详细讲解了每个知识点的各种用法，本书覆盖了 Struts 2.1 近 80% 的 API。

本书适合于有较好的 Java 编程基础，有一定 JSP、Servlet 等 Web 编程基础或具备初步 Java EE 编程基础的读者阅读。本书给出了 Struts 2 入门级的示范实例，可作为学习 Struts 2 框架的学习指南；也详细讲解了 Struts 2 各个知识点及各选项的详细用法，可作为项目开发的参考手册。

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.x 权威指南 / 李刚编著. —3 版. —北京：电子工业出版社，2012.10
ISBN 978-7-121-18047-7

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

中国版本图书馆 CIP 数据核字 (2012) 第 200297 号

策划编辑：张月萍

责任编辑：葛 娜

印 刷：北京东光印刷厂

装 订：三河市鹏成印业有限公司

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：42

字数：1073千字

彩插：1

印 次：2012年10月第1次印刷

印 数：4000册

定价：89.00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 zltz@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 已经成为 Java EE 企业开发最实用、使用最广泛的 MVC 框架，Struts 2 框架不仅成熟、稳定，而且具有很好的扩展性，能与大部分 Java EE 框架整合。所以掌握 Struts 2 技术是从事 Java EE 开发的必备技能。

从 2007 年开始，本书历经《Struts 2 权威指南》（国内第一本全面、深入介绍 Struts 2 的图书）和《Struts 2.1 权威指南》前两版的积累、5 年多的时间沉淀，知识体系十分完备，示例程序非常典型，而且真正能让读者“读得懂、学得会、做得出”；本书最后还配备了两个 Struts 2 + Spring 3+Hibernate 4 整合开发案例，力求再现企业开发的实用技术组合。

本书的知识点覆盖了 Struts 2 框架的绝大部分内容，本书从 Web 应用的请求—响应架构讲起，深入剖析了 Struts 2 框架的 MVC 原理和 Struts 2 框架的运行机制，详细介绍了 Struts 2 的 Action 开发、Action 配置管理、Result 配置管理这些基本知识——介绍 Result 配置管理时，详细分析了 Struts 2 的 Result 机制的原理与优势，并通过示例讲解了动态结果、请求参数决定结果等内容，这些内容表明本书的定位并非简单的 Struts 2 入门图书，而是全面、深入的 Struts 2 图书。除此之外，Struts 2 的异常机制、类型转换、输入校验、文件上传和下载、Struts 2 国际化支持、Struts 2 的 Ajax 支持、拦截器机制、Convention 插件与 REST 等方面，都可以在本书中找到全面、透彻的讲解；本书还详尽讲解了 Struts 2 与 Spring、Hibernate、JSF、SiteMesh、JasperReports、JFreeChart 等框架的整合开发，力求让读者能真正步入实际开发殿堂。

读者在学习本书时，应遵循本书的介绍体系，首先掌握相关知识的基本概念和基本操作，然后结合各知识点配套的示例程序，重点掌握它们在实际开发中的用法，最后通过本书后面的两个综合实例，掌握 Struts 2 在实际开发中的应用。最后希望：打算阅读此书的朋友，一定是准备用手敲代码的人。如果你只是想“读”本书，不想动“手”实践，那么建议不要阅读本书。

如果读者在阅读本书时遇到难以解决的知识问题，可以登录疯狂 Java 联盟（<http://www.crazyit.org>）与本书庞大的读者群交流，笔者也会通过该平台与大家一起交流、学习。

本书有什么特点



本书为 Struts 2 的所有知识点都配套了小案例让读者学以致用，然后详细介绍该知识点的各种选项和参数，再指出该知识点在实际开发中的应用场景，最后简要说明该知识点的底层运行机制。

除此之外，本书还有如下几个特色。

1. 经验丰富，针对性强

作为一个多年的程序员，笔者积累了大量的实际开发经验和感悟。同时笔者也是疯狂软件教育中心的教学总监，并为多个公司做过企业培训，因此能感受到各种层次学习者在学习过程中的苦痛，因此针对性很强。

2. 讲解具体，示范性强

书中通过大量的示范性实例（全书范例近百个），逐一、详细地讲解了 Struts 2 各个知识点的用法，并细致地讲解每个用法的各种参数、选项，可以帮助读者从入门到精通。

3. 配合案例，快速提高

本书最后配备了两个典型案例，可以让读者在掌握前面的基础知识点的情况下，将所有知识融会贯通，把基础知识点应用到项目开发过程中。

本书写给谁看



本书适合于有较好的 Java 编程基础，有一定 JSP、Servlet 等 Web 编程基础或具备初步 Java EE 编程基础的读者阅读。本书给出了 Struts 2 入门级的示范实例，可作为学习 Struts 2 框架的学习指南；也详细讲解了 Struts 2 各个知识点及各选项的详细用法，可作为项目开发的参考手册。

2012-7-2



光盘说明

一、光盘内容



本光盘是《Struts 2.x 权威指南》一书的配书光盘，书中的代码按章节存放，即第 3 章第 4 节所使用的代码放在 codes 文件夹的 03\3.4 文件夹下，依此类推。

书中每份源代码也给出了与光盘源文件的对应关系，以方便读者查找。



本光盘 codes 目录下有 19 个文件夹，其内容和含义说明如下。

(1) 02~20 文件夹名对应于《Struts 2.x 权威指南》中的章名，即第 3 章所使用的代码放在 codes 文件夹的 03 文件夹下，依此类推。

(2) 19 文件夹下有 reg_login 和 reg_login_Eclipse 两个文件夹，它们是同一个项目的源文件，其中 reg_login 是 IDE 平台无关的项目，使用 Ant 来编译即可；而 reg_login_Eclipse 是该项目在 Eclipse IDE 工具中的项目文件。

(3) 20 文件夹下有 auction 和 auction_Eclipse 两个文件夹，它们是同一个项目的源文件，其中 auction 是 IDE 平台无关的项目，使用 Ant 来编译即可；而 auction_Eclipse 是该项目在 Eclipse IDE 工具中的项目文件。

(4) 光盘中很多文件夹下都包含了 .classpath、.mymetadadata、.project 等文件，它们是 Eclipse 项目文件，请不要删除。



本光盘“Struts2 视频”目录下有 39 个视频文件，本视频是作者李刚老师在疯狂软件教育中心讲解 Struts 2 框架时的授课实况，也是荣获中国书刊发行协会“2011 年度全行业优秀畅销品种”的《轻量级 Java EE 企业应用实战》一书的配套视频。

本视频对 Struts 2 的讲解并不是仅仅停留在 Struts 2 框架使用的层次上，很多都已“深挖”到了 Struts 2 框架的设计、底层运行机制。

二、运行环境



本书中的程序在以下环境调试通过。

(1) 安装 jdk-7u3-windows-i586.exe，安装完成后，添加 CLASSPATH 环境变量，该环境变量的值为.;%JAVA_HOME%/lib/tools.jar;%JAVA_HOME%/lib/dt.jar。如果为了可以编译和运行 Java 程序，还应该在 PATH 环境变量中增加%JAVA_HOME%/bin。其中 JAVA_HOME 代表 JDK（不是 JRE）的安装路径。

(2) 安装 Apache 的 Tomcat 7.0.20，不要使用安装文件安装，而要采用解压缩的方式安装。安装 Tomcat 需要增加 JAVA_HOME 环境变量，环境变量值为 JDK 安装路径。安装完成后，将 Tomcat 安装路径的 lib 下的 jsp-api.jar 和 servlet-api.jar 两个 JAR 文件添加到 CLASSPATH 环境变量之后。

(3) 安装 apache-ant-1.8.2。将下载的 Ant 压缩文件解压缩到任意路径，然后增加 ANT_HOME

的环境变量，让变量的值为 Ant 的解压缩路径，并在 PATH 环境变量中增加%ANT_HOME%/bin 环境变量。

(4) 安装 MySQL 5.5 或更高版本，安装 MySQL 时选择 GBK 编码方式。

(5) 安装 Eclipse-jee-indigo-SR2 版。

三、注意事项

(1) 独立应用程序的代码中都包括 build.xml 文件，在 DOS 或 Shell 下进入 build.xml 文件所在路径，执行如下命令：

```
ant compile -- 编译程序
```

```
ant run --运行程序
```

(2) 对于 Web 应用，将该应用复制到%TOMCAT_HOME%/webapps 路径下，然后进入 build.xml 所在路径，执行如下命令：

```
ant compile -- 编译应用
```

启动 Tomcat 服务器，使用浏览器即可访问该应用。

(3) 对于 Eclipse 项目文件，导入 Eclipse 开发工具即可。

(4) 第 19、20 章的案例，请参看项目下的 readme.txt 文件。

(5) 程序中有大量代码需要连接数据库，读者应修改数据库 URL，以及用户名、密码，让这些代码与读者运行环境一致。如果项目下有 SQL 脚本，导入 SQL 脚本即可；如果没有 SQL 脚本，系统将在运行时自动建表，读者只需创建对应的数据库即可。

(6) 在使用本光盘的程序时，请将程序拷贝到硬盘上，并去除文件的只读属性。

四、技术支持

如果在使用本光盘中遇到不懂的技术问题，可以登录如下网站与作者联系：

<http://www.crazyit.org>

目 录 CONTENTS

第 1 章 Struts 2 概述..... 1	2.7 完成程序国际化..... 32
1.1 MVC 思想概述..... 2	2.7.1 定义国际化资源文件..... 33
1.1.1 Web 技术的发展..... 2	2.7.2 加载资源文件..... 34
1.1.2 Model 1 和 Model 2..... 4	2.7.3 输出国际化信息..... 35
1.1.3 MVC 思想及其优势..... 5	2.8 增加数据校验..... 36
1.1.4 常用的 MVC 框架..... 6	2.8.1 继承 ActionSupport..... 37
1.2 Struts 1 的缺陷与现状..... 8	2.8.2 使用 Struts 2 的校验框架..... 39
1.2.1 支持的表现层技术单一..... 8	2.9 本章小结..... 40
1.2.2 与 Servlet API 严重耦合..... 8	
1.2.3 代码依赖于 Struts 1 API..... 9	第 3 章 Struts 2 核心技术..... 41
1.3 Struts 2 的优势与体系..... 9	3.1 在 Eclipse 中开发 Struts 2..... 42
1.3.1 Struts 2 的优势..... 9	3.1.1 创建 Web 应用..... 42
1.3.2 Struts 2 架构..... 11	3.1.2 增加 Struts 2 支持..... 44
1.3.3 Struts 2 的配置文件..... 11	3.1.3 部署 Struts 2 应用..... 45
1.3.4 Struts 2 的标签库..... 12	3.1.4 增加应用的 Action..... 46
1.3.5 Struts 2 的控制器组件..... 13	3.2 Struts 2 的基本流程..... 47
1.4 本章小结..... 14	3.2.1 核心控制器
	StrutsPrepareAndExecuteFilter 47
第 2 章 Struts 2 的 HelloWorld..... 15	3.2.2 业务控制器..... 48
2.1 下载和安装 Struts 2 框架..... 16	3.2.3 Struts 2 的模型组件..... 48
2.2 从用户请求开始..... 17	3.2.4 Struts 2 的视图组件..... 49
2.3 创建 Struts 2 的 Web 应用..... 19	3.2.5 Struts 2 的运行流程..... 49
2.3.1 创建 Web 应用..... 19	3.3 Struts 2 的基本配置..... 50
2.3.2 增加 Struts 2 功能..... 20	3.3.1 配置 web.xml 文件..... 50
2.4 实现控制器..... 21	3.3.2 struts.xml 配置文件..... 51
2.4.1 实现控制器类..... 21	3.3.3 struts.properties 文件与常量配置..... 54
2.4.2 配置 Action..... 22	3.3.4 struts.xml 文件结构..... 59
2.4.3 增加视图资源完成应用..... 23	3.4 深入 Struts 2 配置文件..... 61
2.5 改进控制器..... 24	3.4.1 Bean 配置..... 61
2.5.1 实现 Action 接口..... 24	3.4.2 常量配置..... 62
2.5.2 跟踪用户状态..... 25	3.4.3 包配置..... 62
2.5.3 添加处理信息..... 26	3.4.4 命名空间配置..... 64
2.5.4 输出处理信息..... 28	3.4.5 包含配置..... 66
2.6 改进视图组件..... 29	3.4.6 拦截器配置..... 66
2.6.1 保护 JSP 页面..... 30	3.5 Struts 2 的 Action..... 67
2.6.2 改善输出页面..... 30	3.5.1 实现 Action 处理类..... 67
2.6.3 使用 UI 标签简化表单页面..... 31	3.5.2 Action 访问 Servlet API..... 71
	3.5.3 Action 直接访问 Servlet API..... 74

3.5.4	配置 Action.....	76	4.4.6	基于 Struts 2 的类型转换器.....	130
3.5.5	配置 Action 的默认处理类.....	78	4.4.7	数组属性的类型转换器.....	132
3.5.6	动态方法调用.....	78	4.4.8	集合属性的类型转换器.....	135
3.5.7	为 action 元素指定 method 属性.....	81	4.5	集合类型转换的高级特性.....	137
3.5.8	使用通配符.....	82	4.5.1	指定集合元素的类型.....	137
3.5.9	配置默认的 Action.....	87	4.5.2	为 Set 集合的元素指定索引属性.....	138
3.6	管理处理结果.....	87	4.6	类型转换中的错误处理.....	140
3.6.1	处理结果的改进.....	88	4.6.1	类型转换的错误处理流程.....	140
3.6.2	配置结果.....	88	4.6.2	处理类型转换错误.....	142
3.6.3	Struts 2 支持的处理结果类型.....	89	4.6.3	输出类型转换错误.....	142
3.6.4	chain 结果类型.....	91	4.6.4	处理集合属性的转换错误.....	145
3.6.5	plainText 结果类型.....	93	4.7	本章小结.....	147
3.6.6	redirect 结果类型.....	95	第 5 章	Struts 2 的输入校验.....	148
3.6.7	redirectAction 结果类型.....	96	5.1	输入校验概述.....	149
3.6.8	动态结果.....	97	5.1.1	为什么需要输入校验.....	149
3.6.9	请求参数决定结果.....	98	5.1.2	客户端校验.....	150
3.6.10	全局结果.....	100	5.1.3	服务器端校验.....	152
3.7	属性驱动和模型驱动.....	101	5.1.4	类型转换和输入校验.....	155
3.7.1	模型的作用.....	101	5.2	基本输入校验.....	156
3.7.2	使用模型驱动.....	103	5.2.1	编写校验规则文件.....	156
3.8	Struts 2 的异常机制.....	105	5.2.2	国际化提示信息.....	159
3.8.1	Struts 2 的异常处理哲学.....	105	5.2.3	使用客户端校验.....	161
3.8.2	声明式异常捕捉.....	107	5.3	校验器的配置风格.....	164
3.8.3	输出异常信息.....	110	5.3.1	字段校验器配置风格.....	164
3.9	未知处理器.....	111	5.3.2	非字段校验器配置风格.....	165
3.10	本章小结.....	113	5.3.3	短路校验器.....	166
第 4 章	Struts 2 的类型转换.....	114	5.3.4	校验顺序和短路.....	167
4.1	类型转换的意义.....	115	5.4	校验文件的搜索规则.....	168
4.1.1	表现层数据处理.....	115	5.5	内建校验器.....	170
4.1.2	传统的类型转换.....	116	5.5.1	注册校验器.....	170
4.2	Struts 2 内建的类型转换器.....	118	5.5.2	必填校验器.....	172
4.3	基于 OGNL 的类型转换.....	120	5.5.3	必填字符串校验器.....	172
4.3.1	使用 OGNL 转换成 Map 集合.....	121	5.5.4	整数校验器.....	173
4.3.2	使用 OGNL 转换成 List 集合.....	122	5.5.5	日期校验器.....	174
4.4	自定义类型转换器.....	123	5.5.6	表达式校验器.....	175
4.4.1	系统需求.....	123	5.5.7	字段表达式校验器.....	175
4.4.2	实现类型转换器.....	125	5.5.8	邮件地址校验器.....	176
4.4.3	局部类型转换器.....	127	5.5.9	网址校验器.....	176
4.4.4	全局类型转换器.....	128	5.5.10	Visitor 校验器.....	177
4.4.5	局部类型转换器和全局类型 转换器的说明.....	130	5.5.11	转换校验器.....	180

5.5.12 字符串长度校验器.....	181	7.2.1 Struts 2 的国际化.....	224
5.5.13 正则表达式校验器.....	181	7.2.2 视图页面的国际化.....	224
5.6 基于 Annotation 的输入校验.....	182	7.2.3 Action 的国际化.....	225
5.7 手动完成输入校验.....	184	7.2.4 使用包范围的国际化资源.....	227
5.7.1 重写 validate 方法.....	184	7.2.5 使用全局国际化资源.....	228
5.7.2 重写 validateXxx 方法.....	185	7.2.6 输出带占位符的国际化消息.....	231
5.7.3 Struts 2 的输入校验流程.....	186	7.2.7 加载资源文件的顺序.....	232
5.8 本章小结.....	188	7.3 允许用户自行选择程序语言.....	233
第 6 章 上传和下载文件.....	189	7.3.1 Struts 2 国际化机制的原理.....	234
6.1 文件上传的原理.....	190	7.3.2 创建下拉列表框.....	235
6.1.1 表单元素的 enctype 属性.....	190	7.3.3 选择程序语言.....	237
6.1.2 手动上传.....	193	7.4 本章小结.....	238
6.1.3 使用 Servlet 3.0 API 完成上传.....	194	第 8 章 Struts 2 的标签库.....	239
6.2 Struts 2 的文件上传.....	197	8.1 Struts 2 标签库概述.....	240
6.2.1 Struts 2 的文件上传支持.....	197	8.1.1 标签的优势.....	240
6.2.2 实现文件上传的 Action.....	197	8.1.2 Struts 2 的标签分类.....	241
6.2.3 配置文件上传的 Action.....	200	8.2 Struts 2 标签入门.....	242
6.2.4 手动实现文件过滤.....	201	8.2.1 使用 Struts 2 标签的准备.....	242
6.2.5 拦截器实现文件过滤.....	203	8.2.2 Struts 2 的 OGNL 表达式语言.....	243
6.2.6 输出错误提示.....	205	8.2.3 OGNL 中的集合操作.....	245
6.2.7 文件上传的常量配置.....	206	8.2.4 访问静态成员.....	246
6.3 使用 Pell 上传.....	206	8.2.5 Lambda (λ) 表达式.....	246
6.4 同时上传多个文件.....	207	8.3 控制标签.....	247
6.4.1 使用数组同时上传多个文件.....	208	8.3.1 if/elseif/else 标签.....	247
6.4.2 使用 List 同时上传多个文件.....	210	8.3.2 iterator 标签.....	248
6.5 使用 Struts 2 控制文件下载.....	212	8.3.3 append 标签.....	250
6.5.1 实现文件下载的 Action.....	212	8.3.4 generator 标签.....	252
6.5.2 配置 Action.....	214	8.3.5 merge 标签.....	253
6.5.3 下载前的授权控制.....	214	8.3.6 subset 标签.....	254
6.6 本章小结.....	216	8.3.7 sort 标签.....	256
第 7 章 Struts 2 的国际化.....	217	8.4 数据标签.....	258
7.1 程序国际化简介.....	218	8.4.1 action 标签.....	258
7.1.1 程序国际化的意义.....	218	8.4.2 bean 标签.....	260
7.1.2 Java 国际化的思路.....	219	8.4.3 date 标签.....	262
7.1.3 Java 支持的国家 and 语言.....	219	8.4.4 debug 标签.....	263
7.1.4 完成程序国际化.....	220	8.4.5 i18n 标签.....	263
7.1.5 使用 MessageFormat 处理带 占位符的消息.....	222	8.4.6 include 标签.....	263
7.2 Struts 2 的国际化支持.....	224	8.4.7 param 标签.....	264
		8.4.8 property 标签.....	264
		8.4.9 push 标签.....	265

8.4.10 set 标签.....	265	9.4.1 实现拦截器类.....	313
8.4.11 text 标签.....	267	9.4.2 使用拦截器.....	314
8.4.12 url 标签.....	267	9.4.3 拦截器和 Struts 2 插件的关系	316
8.5 主题和模板.....	269	9.5 深入拦截器编程.....	317
8.5.1 选择主题.....	269	9.5.1 拦截方法的拦截器.....	317
8.5.2 自定义主题.....	271	9.5.2 拦截器的执行顺序.....	319
8.6 表单标签.....	273	9.5.3 拦截结果的监听器.....	321
8.6.1 表单标签的通用属性.....	273	9.5.4 覆盖拦截器栈里特定拦截器的 参数.....	322
8.6.2 表单标签的 name 和 value 属性.....	275	9.6 拦截器应用示例: 实现权限控制	323
8.6.3 checkboxlist 标签.....	275	9.6.1 实现拦截器类.....	323
8.6.4 combobox 标签.....	277	9.6.2 配置权限控制拦截器.....	324
8.6.5 doubleselect 标签.....	278	9.7 本章小结.....	326
8.6.6 head 标签.....	280		
8.6.7 optiontransfersselect 标签	280	第 10 章 Struts 2 的 Ajax 支持.....	327
8.6.8 select 标签.....	282	10.1 Struts 2 基本的 Ajax 支持	328
8.6.9 radio 标签	284	10.1.1 Ajax 技术概述	328
8.6.10 optgroup 标签	284	10.1.2 使用 stream 结果类型实现 Ajax	331
8.6.11 token 标签.....	285	10.2 基于 Dojo 的异步表单	334
8.6.12 updownselect 标签.....	287	10.2.1 安装 Dojo 插件.....	335
8.7 非表单标签.....	288	10.2.2 定义异步表单.....	335
8.7.1 actionerror 和 actionmessage 标签.....	289	10.2.3 实现 Action.....	337
8.7.2 component 标签.....	289	10.2.4 实现服务器响应页面	338
8.8 本章小结.....	291	10.3 基于 Dojo 的 Pub-Sub 事件 模型.....	340
第 9 章 Struts 2 的拦截器.....	292	10.3.1 Pub-Sub 原理	340
9.1 理解拦截器	293	10.3.2 Pub-Sub 示例	341
9.1.1 理解 DRY 规则	293	10.3.3 阻止请求.....	343
9.1.2 拦截器的意义.....	295	10.4 基于 Dojo 的 Ajax 标签	343
9.1.3 拦截器的实现原理.....	297	10.4.1 div 标签.....	343
9.1.4 拦截器和 AOP 的关系.....	300	10.4.2 a 和 submit 标签	349
9.1.5 拦截器在 Struts 2 中的作用	301	10.4.3 autocompleter 标签.....	354
9.2 Struts 2 内建拦截器介绍.....	303	10.4.4 tabbedpanel 标签.....	360
9.2.1 Struts 2 内建拦截器.....	303	10.4.5 datetimepicker 标签	363
9.2.2 struts-default.xml 里配置的 拦截器.....	304	10.4.6 tree 和 treenode 标签	365
9.3 拦截器配置初步	308	10.4.7 textarea 标签	369
9.3.1 配置拦截器.....	308	10.4.8 bind 标签.....	369
9.3.2 使用拦截器.....	310	10.5 基于 JSON 插件的 Ajax 实现.....	371
9.3.3 配置默认的拦截器.....	311	10.5.1 JSON 的基本知识	371
9.4 开发自己的拦截器.....	313	10.5.2 安装 Struts 2 的 JSON 插件	374

10.5.3 实现 Action 逻辑.....	374	12.3.4 集合连接运算符.....	419
10.5.4 实现 JSP 页面.....	376	12.3.5 Map 连接运算符.....	419
10.6 本章小结.....	377	12.3.6 算术运算符.....	420
第 11 章 Convention 插件与 REST 支持	378	12.3.7 比较运算符.....	420
11.1 Convention 插件和零配置.....	379	12.3.8 逻辑运算符.....	421
11.1.1 搜索 Action.....	379	12.3.9 内建函数.....	421
11.1.2 按约定映射 Action.....	380	12.3.10 空值处理运算符.....	422
11.1.3 按约定映射 Result.....	382	12.3.11 运算符优先级.....	422
11.1.4 Action 链的约定.....	385	12.4 FreeMarker 的常用指令.....	422
11.1.5 自动重加载映射.....	386	12.4.1 if 指令.....	423
11.2 Convention 插件的相关常量.....	386	12.4.2 switch、case、default、break 指令.....	423
11.3 Convention 的 Annotation.....	387	12.4.3 list、break 指令.....	424
11.3.1 Action 配置相关的 Annotation.....	388	12.4.4 include 指令.....	424
11.3.2 Result 配置相关的 Annotation.....	389	12.4.5 import 指令.....	425
11.3.3 包和命名空间相关的 Annotation.....	391	12.4.6 noparse 指令.....	425
11.3.4 异常处理相关的 Annotation.....	391	12.4.7 escape、noescape 指令.....	425
11.3.5 拦截器配置相关的 Annotation.....	392	12.4.8 assign 指令.....	426
11.4 REST 简介.....	393	12.4.9 setting 指令.....	427
11.4.1 资源和标识符.....	394	12.4.10 macro、nested、return 指令.....	427
11.4.2 操作资源的方式.....	394	12.5 使用 FreeMarker 作为 Struts 2 的 视图技术.....	430
11.5 Struts 2 的 REST 支持.....	395	12.5.1 在 FreeMarker 模板中使用 Struts 2 标签.....	430
11.5.1 RestActionMapper 简介.....	395	12.5.2 使用 Freemarker Result.....	433
11.5.2 安装 REST 插件.....	397	12.5.3 解析模板中的变量.....	434
11.5.3 实现支持 REST 的 Action 类.....	397	12.5.4 访问 Servlet/JSP 范围的值.....	436
11.6 本章小结.....	404	12.6 本章小结.....	438
第 12 章 FreeMarker 详解	405	第 13 章 整合 Spring	439
12.1 体验 FreeMarker.....	406	13.1 Spring 简介.....	440
12.1.1 下载和安装 FreeMarker.....	406	13.1.1 Spring 的起源和核心.....	440
12.1.2 在 Java 程序中使用 FreeMarker.....	407	13.1.2 Spring 在 Java EE 应用中的 作用.....	442
12.1.3 在 Web 应用中使用 FreeMarker.....	408	13.2 Struts 2 插件.....	444
12.2 FreeMarker 的模板文件.....	411	13.2.1 插件概述.....	444
12.2.1 模板结构.....	412	13.2.2 插件开发概述.....	445
12.2.2 FTL 指令规则.....	412	13.3 使用 Spring 插件完成整合.....	447
12.2.3 插值规则.....	413	13.3.1 整合 Spring 前的准备工作.....	447
12.3 表达式.....	414	13.3.2 整合 Spring 的思考.....	449
12.3.1 直接指定值.....	414	13.4 让 Spring 管理控制器.....	450
12.3.2 输出变量值.....	417		
12.3.3 字符串操作.....	418		

13.4.1 整合流程.....	450	16.1 JasperReports 简介.....	496
13.4.2 从页面请求开始.....	451	16.1.1 下载和安装 JasperReports.....	496
13.4.3 实现控制器逻辑.....	451	16.1.2 开发 JasperReports 静态报表.....	497
13.4.4 实现业务逻辑.....	453	16.1.3 JasperReports 的开发流程.....	503
13.5 使用自动装配.....	454	16.1.4 解决 PDF 报表的中文问题.....	504
13.5.1 指定自动装配.....	454	16.2 动态报表.....	504
13.5.2 修改配置文件.....	455	16.2.1 带参数的报表.....	504
13.5.3 改变自动装配策略.....	456	16.2.2 带查询的报表.....	508
13.6 本章小结.....	458	16.2.3 以 VO 集合作为数据源的 动态报表.....	512
第 14 章 整合 JSF.....	459	16.3 使用 iReport 设计报表.....	516
14.1 JSF 简介.....	460	16.3.1 下载和安装 iReport.....	516
14.1.1 JSF 概述.....	460	16.3.2 使用 iReport.....	518
14.1.2 JSF 和 Struts 2 的对比.....	461	16.3.3 建立数据源.....	520
14.2 使用 MyFaces.....	461	16.3.4 设计报表布局.....	522
14.2.1 下载和安装 MyFaces.....	462	16.4 安装 JasperReports 插件.....	525
14.2.2 从输入页面开始.....	463	16.5 使用 JasperReports 报表.....	525
14.2.3 在 Web 应用中配置 MyFaces.....	465	16.5.1 创建生成报表的 Action.....	525
14.2.4 实现后台 Bean.....	466	16.5.2 配置生成 JasperReports 报表的 Action.....	526
14.2.5 定义导航规则.....	467	16.5.3 生成 JasperReports 报表.....	527
14.3 安装 JSF 插件.....	469	16.6 本章小结.....	529
14.4 使用 JSF 插件.....	470	第 17 章 整合 JFreeChart.....	530
14.4.1 实现业务逻辑组件.....	470	17.1 JFreeChart 简介.....	531
14.4.2 列出全部图书.....	473	17.1.1 JFreeChart 的下载和安装.....	531
14.4.3 添加/修改图书.....	475	17.1.2 使用 JFreeChart.....	532
14.5 整合 JSF 的思考.....	480	17.1.3 JFreeChart 开发步骤.....	534
14.6 本章小结.....	480	17.2 JFreeChart 的常用统计图表.....	535
第 15 章 整合 SiteMesh.....	481	17.2.1 柱状图.....	536
15.1 SiteMesh 简介.....	482	17.2.2 折线图.....	540
15.1.1 SiteMesh 概述.....	482	17.2.3 时间顺序图.....	540
15.1.2 下载和安装 SiteMesh.....	483	17.3 在网页中生成带交互功能的 统计图.....	542
15.1.3 定义装饰器页面.....	485	17.4 安装 JFreeChart 插件.....	547
15.1.4 定义原始页面.....	487	17.5 在 Struts 2 应用中使用 JFreeChart.....	548
15.1.5 更复杂的装饰器页面.....	488	17.5.1 创建 Action 类.....	548
15.2 使用 SiteMesh 插件.....	490	17.5.2 配置 Action.....	550
15.2.1 安装 SiteMesh 插件.....	491	17.5.3 改进 JFreeChart 插件.....	551
15.2.2 在 Struts 2 中使用 SiteMesh.....	492	17.5.4 使用改进后的 JFreeChart 插件.....	554
15.3 本章小结.....	494		
第 16 章 整合 JasperReports.....	495		

17.6 本章小结.....	555	19.6 使用 SiteMesh 装饰页面.....	599
第 18 章 整合 Hibernate.....	556	19.7 本章小结.....	600
18.1 Hibernate 概述.....	557	第 20 章 电子拍卖系统.....	601
18.1.1 ORM 的功能与 Hibernate.....	557	20.1 系统功能简介和架构设计.....	602
18.1.2 流行的 ORM 框架简介.....	560	20.1.1 系统功能简介.....	602
18.2 Struts 2 整合 Hibernate 的思考.....	560	20.1.2 系统架构设计.....	603
18.2.1 自定义的整合方案.....	560	20.1.3 相关技术介绍.....	604
18.2.2 整合方案的实现策略.....	561	20.2 持久层设计.....	606
18.3 整合示例.....	562	20.2.1 系统实体.....	606
18.3.1 持久层设计.....	562	20.2.2 系统 E-R 图和数据表.....	607
18.3.2 DAO 层设计.....	565	20.2.3 实现 Hibernate PO.....	609
18.3.3 业务逻辑层设计.....	568	20.2.4 管理 SessionFactory.....	616
18.3.4 初始化组件工厂.....	570	20.3 实现系统 DAO 层.....	617
18.3.5 控制器实现.....	571	20.3.1 实现 DAO 基类.....	618
18.4 整合后的思考和 Spring 框架.....	572	20.3.2 实现系统 DAO 组件.....	619
18.5 本章小结.....	573	20.3.3 配置系统 DAO 组件.....	622
第 19 章 “零配置” 案例: 注册登录系统.....	574	20.4 实现业务逻辑层.....	624
19.1 系统概述.....	575	20.4.1 定义业务逻辑组件接口.....	624
19.1.1 系统功能简介.....	575	20.4.2 依赖注入 DAO 组件.....	626
19.1.2 系统总体设计.....	576	20.4.3 业务逻辑组件中的异常处理.....	627
19.2 实现 Hibernate 持久层.....	576	20.4.4 处理用户竞价.....	628
19.3 实现 DAO 层.....	577	20.4.5 判断拍卖物品状态.....	630
19.3.1 管理 SessionFactory.....	578	20.4.6 事务管理.....	632
19.3.2 实现 DAO 基类.....	579	20.4.7 配置业务层组件.....	633
19.3.3 实现系统 DAO 组件.....	580	20.5 实现系统 Web 层.....	635
19.3.4 配置 DAO 组件.....	582	20.5.1 配置 Struts 2 的核心 Filter.....	636
19.4 实现业务逻辑层.....	582	20.5.2 处理用户登录.....	637
19.4.1 定义业务逻辑组件接口.....	582	20.5.3 图形验证码.....	640
19.4.2 提供业务逻辑组件实现类.....	583	20.5.4 Web 层的异常处理.....	641
19.4.3 配置业务逻辑组件.....	585	20.5.5 登录控制.....	642
19.5 使用 Struts 2.....	586	20.5.6 添加物品.....	644
19.5.1 整合 Struts 2 和 Spring.....	586	20.5.7 处理用户竞价.....	650
19.5.2 处理登录.....	587	20.6 使用 SiteMesh 页面装饰.....	655
19.5.3 图形验证码.....	591	20.6.1 在 Struts 2 中整合 SiteMesh 页面装饰.....	655
19.5.4 使用 Ajax 验证用户名.....	594	20.6.2 定义页面装饰.....	655
19.5.5 注册新用户.....	597	20.7 本章小结.....	658

第 1 章

Struts 2 概述

本章要点

- ✎ Web 应用的发展
- ✎ Model 1 和 Model 2
- ✎ MVC 思想
- ✎ MVC 模式的优势
- ✎ 常用 MVC 框架及其特征
- ✎ Struts 1 的缺陷与没落
- ✎ Struts 2 的起源
- ✎ Struts 2 的框架结构
- ✎ Struts 2 的标签库
- ✎ Struts 2 的控制器组件

Struts 1 是全世界第一个发布的 MVC 框架,它由 Craig McClanahan 在 2001 年发布,该框架一经推出,就得到了世界上 Java Web 开发者的拥护。但随着时间的推移,Struts 1 的缺点不断暴露,大量开发者不得不开始寻觅新的 MVC 框架。

在这样的需求背景下,Struts 框架衍生了两个版本:①基于传统 Struts 的 Struts 1,这个框架除了被一些技术守旧的项目、团队所使用之外,基本已趋于“死亡”;②基于 WebWork 的 Struts 2,它其实就是 WebWork 的升级版。实际上,WebWork 框架已不存在,完全变成了现在的 Struts 2,连 WebWork 所属的 OpenSymphony 也关闭了。(叹息!)

目前主流的 MVC 框架还有 JSF、Spring MVC 和 Tapestry 等。除了这些著名的 MVC 框架外,还有一些边缘团队的 MVC 框架也很有借鉴意义。但对于企业实际使用 MVC 框架而言,框架的稳定性则应该是最值得考虑的问题。一般不建议选择刚刚起步的新框架,因为框架本身可能就存在一些隐藏的问题,会将其自身的 Bug 引入企业应用。

1.1 MVC 思想概述

今天所见到的绝大部分应用,都是基于 B/S(浏览器/服务器)架构的,其中的服务器就是 Web 服务器。可见,Web 应用是目前广泛使用的应用模式,而 Struts 2 是一个具有很好的实用价值的 Web MVC 框架。在介绍 Struts MVC 框架之前,我们首先介绍 Web 应用的发展历史和 MVC 思想。

➤➤ 1.1.1 Web 技术的发展

随着 Internet 技术的广泛使用,Web 技术已经广泛应用于 Internet 上,早期的 Web 应用全部是静态的 HTML 页面,用于将一些文本信息呈现给浏览者,但这些信息是固定写在 HTML 页面里的,该页面不具备与用户交互的能力,没有动态显示的功能。

很自然地,人们希望 Web 应用里应该包含一些能动态执行的页面,最早的 CGI(通用网关接口)技术满足了该要求,CGI 技术使得 Web 应用可以与客户端浏览器交互,不再需要使用静态的 HTML 页面。CGI 技术可以从数据库读取信息,将这些信息呈现给用户;还可以获取用户的请求参数,并将这些参数保存到数据库里。

CGI 技术开启了动态 Web 应用的时代,给了这种技术无限的可能性。但 CGI 技术存在很多缺点,其中最大的缺点就是开发动态 Web 应用难度非常大,而且在性能等各方面也存在限制。到 1997 年时,随着 Java 语言的广泛使用,Servlet 技术迅速成为动态 Web 应用的主要开发技术。相比传统的 CGI 应用而言,Servlet 具有如下优势。

- Servlet 是基于 Java 语言创建的,而 Java 语言则内建了多线程支持,这一点大大提高了动态 Web 应用的性能。
- Servlet 应用可以充分利用 Java 语言的优势,例如 JDBC(Java DataBase Connection)等。同时,Java 语言提供了丰富的类库,可以大大简化 Servlet 的开发。
- 除此之外,Servlet 运行在 Web 服务器中,由 Web 服务器去负责管理 Servlet 的实例化,并对客户端提供多线程、网络通信等功能,这都保证了 Servlet 有更好的稳定性和性能。

Servlet 在 Web 应用中被映射成一个 URL(统一资源定位符),该 URL 可以被客户端浏览器请求,当用户向指定 URL 对应的 Servlet 发送请求时,该请求被 Web 服务器接收到,该 Web 服务器负责处理多线程、网络通信等功能,而 Servlet 的内容则决定了服务器对客户端的响应内容。

图 1.1 显示了 Servlet 的响应流程。

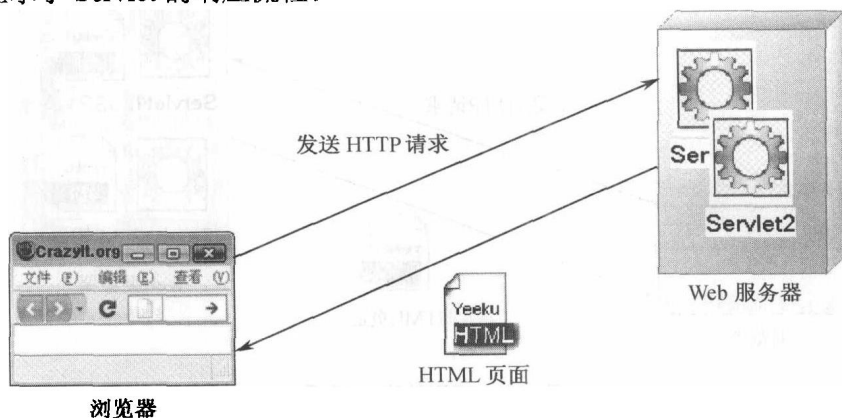


图 1.1 Servlet 的响应流程

正如图 1.1 所显示的，浏览器向 Web 服务器内指定的 Servlet 发送请求，Web 服务器根据 Servlet 生成对客户端的响应。

实际上，这是后来所有的动态 Web 编程技术所使用的模型，这种模型都需要一个动态的程序，或者一个动态页面，当客户端向该动态程序或动态页面发送请求时，Web 服务器根据该动态程序来生成对客户端的响应。

1998 年，微软发布了 ASP 2.0。它是 Windows NT 4 Option Pack 的一部分，作为 IIS 4.0 的外接式附件。它与 ASP 1.0 的主要区别在于它的外部组件是可以初始化的，这样，在 ASP 程序内部的所有组件都有了独立的内存空间，并可以进行事务处理。这标志着 ASP 技术开始真正作为动态 Web 编程技术。

当 ASP 技术在世界上广泛流行时，人们很快感受到这种简单的技术的魅力：ASP 使用 VBScript 作为脚本语言，它的语法简单，开发效率非常高。而且，世界上已经有了非常多的 VB 程序员，这些 VB 程序员可以很轻易地过渡成 ASP 程序员——因此，ASP 技术马上成为应用最广泛的动态 Web 开发技术。

随后，由 Sun 带领的 Java 阵营，立即发布了 JSP 标准，从某种程度上来看，JSP 是 Java 阵营为了对抗 ASP 推出的一种动态 Web 编程技术。

ASP 和 JSP 名称上如此相似，但它们的运行机制存在一些差别，这主要是因为 VBScript 是一种脚本语言，无须编译，而 JSP 使用 Java 作为脚本语句——但 Java 从来就不是解释型的脚本语言，因此 JSP 页面并不能立即执行。因此，JSP 必须编译成 Servlet，这就是说，JSP 的实质还是 Servlet。不过，书写 JSP 比书写 Servlet 要简单得多。

JSP 的运行机理如图 1.2 所示。

对比图 1.1 和图 1.2，不难发现不论是 Servlet 动态 Web 技术，还是 JSP 动态 Web 技术，它们的实质完全一样。可以这样理解：JSP 是一种更简单的 Servlet 技术，这也是 JSP 技术出现的意义——作为一种和 ASP 对抗的技术，简单就是 JSP 的最大优势。

**提示：**

实际上我们编写的 JSP 页面并不能运行，甚至不能被其他程序调用，读者可以把 JSP 页面文件理解成“草稿文件”，这份 JSP “草稿文件”体现了开发者想完成的工作，而 Web 服务器会根据该 JSP “草稿文件”来生成对应的 Servlet。