

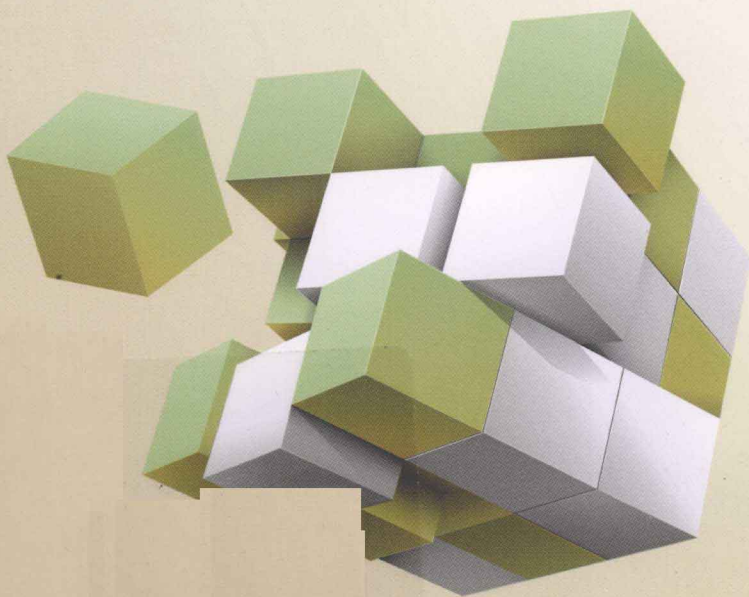


中软国际卓越人才培养系列丛书

JavaEE Mainstream Open-source Framework

JavaEE主流开源框架

- 唐振明 主 编
- 王晓华 于瑞云 蔡国民 编 著



- ★ 详细示例源代码
- ★ 完整PPT讲义
- ★ 教材同步视频教程



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

中软国际卓越人才培养系列丛书

JavaEE 主流开源框架

唐振明 主 编

王晓华 于瑞云 蔡国民 编 著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

目前,很多企业应用中都会使用各种框架技术,其中 Struts2、Hibernate、Spring 是三个常用的 JavaEE 开源框架,掌握这些主流框架技术是很多企业对开发人员的基本要求。本书将这三大框架“一网打尽”,是一把开启 JavaEE 开源框架世界的钥匙。本教材主要分为三个部分,每部分学习一个框架,而每部分却不是孤立的,教材使用一个案例贯穿始终。在每个部分,都将结合学习到的新技能不断修改完善案例,直到最终将三大框架都应用到了案例当中,能够帮助读者深入理解三大框架如何应用在真实应用中。

本书适合各层次的 JavaEE 开发人员阅读。

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

JavaEE 主流开源框架 / 唐振明主编; 王晓华, 于瑞云, 蔡国民编著. —北京: 电子工业出版社, 2011.10
(中软国际卓越人才培养系列丛书)

ISBN 978-7-121-14775-3

I. ①J… II. ①唐… ②王… ③于… ④蔡… III. ①JAVA 语言—程序设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2011) 第 204893 号

策划编辑: 程超群

责任编辑: 程超群 特约编辑: 袁 玺

印 刷: 北京天宇星印刷厂

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: 787×1 092 1/16 印张: 19.75 字数: 496 千字

印 次: 2011 年 10 月第 1 次印刷

印 数: 4 000 册 定价: 35.00 元

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

“框架”一词是很长一段时间来，在 JavaEE 开发领域炙手可热的一个名词。目前存在很多种框架技术，能够有效解决 JavaEE 应用开发中很多通用的问题，其中 Struts2、Hibernate 以及 Spring 是目前使用较多的三个框架，每个框架专注于解决不同的问题，非常有代表性。本书将这三个框架“一网打尽”，提取每个框架中常用的内容，旨在帮助读者在有限时间内，能够掌握这三个框架在企业应用中最常用的功能，从而胜任 Web 应用开发。

Struts2 框架是一个 MVC 框架；Hibernate 框架是一个 ORM 框架；而 Spring 框架是一个综合性的框架，是一个轻量级的解决方案。这三个框架往往各司其职，在应用中的不同层面发挥作用。本书分为三个部分，每个部分学习一个框架，而这三个部分并不是互相独立的，而是设计了一个“教材案例”将这三部分联系在一起。在教材的第一部分，设计了一个“教材案例”，这个案例没有复杂的业务逻辑，主要作用是辅助学习相关知识点。第一部分结束，使用了 Struts2 框架以及 JDBC 技术实现了这个案例。第二部分学习 Hibernate，使用 Hibernate 框架替代了 JDBC 编程，完善了“教材案例”。第三部分学习 Spring，使用 Spring 框架整合了 Struts2、Hibernate，使得最终的“教材案例”中使用了三个框架进行实现。这样的设计和编写思路，能够帮助读者真正理解每个框架的作用，能够正确选择合适的框架解决问题。

编者认为，“快速入门是提高兴趣的捷径”。框架是相对比较复杂的技术，如果能帮助读者快速搭建其开发运行环境，并顺利部署运行成功第一个应用，那无疑会有效地提高读者兴趣。有了学习兴趣，掌握相关技术将不再是一件难事。教材中处处围绕这个思路展开，任何一个新的知识点，都尽量避免过于冗长的理论铺垫，而是先从简单的实例开始。相关实践步骤都配有详细截图和代码说明，能够帮助读者从“使用”开始，保证较好的学习效果。

本书配套的 PPT 也显示在对应的章节，这样的设计能够帮助读者快速了解每个章节的主要内容，起到提纲挈领的作用，也能够帮助读者建立一个完整的知识结构，而不仅仅是掌握了一些凌乱的知识。另外，这样的设计也使得本书非常适合作为高校中 JavaEE 主流框架相关课程的教材，能够方便教师授课，保证授课思路及内容与教材完全匹配、完全同步，从而达到较好的教学效果。另外，附录部分提供了企业关注的技能点，并从企业的角度给予了解析，能够帮助读者进一步整理书中内容，掌握企业需要的技能。

本书所有配套讲义、源代码及视频均可到华信教育资源网 (www.hxedu.com.cn) 免费下载。

在编写本书的过程中，得到了很多领导、同事以及朋友的帮助。感谢中软国际的所有领导以及 CTO 办公室的所有同事，是他们的帮助、鼓励以及支持才有这本书的问世。感谢电子工业出版社的编辑们，如果没有他们的辛苦工作以及真诚建议，这本书的出版将不会这么顺利。

由于编者水平有限，也由于时间仓促，书中一定存在一些不尽如人意的地方，甚至会有一些错误。如果您发现了任何内容方面的问题，烦请一定通知我们 (wangxh@chinasofti.com)，我们将争取尽快勘误。

序

当我翻阅了“中软国际卓越人才培养系列丛书”后，不禁为这套丛书的立意与创新之处感到欣喜。教育部“卓越工程师教育培养计划”有三个主要特征：一是行业企业深度参与培养过程；二是学校按通用标准和行业标准培养工程人才；三是强化培养学生的工程能力和创新能力。这套丛书紧紧围绕“卓越计划”的要求展开，以企业人才需求为前提，同时又充分考虑了高校教育的特点，能让企业有效参与高校培养过程，是一套为“卓越计划”量身打造的丛书。

丛书的设计理念紧扣中软国际 ETC 的“5R”理念，即真实的企业环境、真实的项目经理、真实的项目案例、真实的工作压力、真实的就业机会，切实地将企业真实需求展现给读者。丛书中的知识点力求精简、准确、实用，显然是编著者经过反复推敲并精心设计的成果。丛书中对企业用之甚少的知识点，都进行了弱化，用较少篇幅讲解，而对于企业关注的知识点，都使用非常详尽的内容进行学习。这样的设计对初学者尤其是在校学生非常必要，能够节省很多学习时间，在有限的时间内学习到企业关注的技能，而不是花费很多精力去钻研并不实用的内容。

丛书非常强调“快速入门”这一法宝，能够对某门技术“快速入门”永远是激发学习兴趣的关键。丛书设计了很多“快速入门”章节，使用详尽丰富的图示以及代码示例，保证读者只要根据丛书的指导进行操作，就能够尽快构建出相关技术的实例。

丛书非常注重实际操作，很多知识点都是从提出问题引出，从而在解决问题的过程中讲解相关的技能。丛书中没有大篇幅的理论描述，尽力用最通俗、最简练的语言讲解每一个问题，而不是“故作高深”地使用很多新名词。

非常值得一提的是，丛书配备了对应的 PPT 讲义，并将 PPT 讲义显示到了相应章节，这种形式令人耳目一新。首先能起到提纲挈领的作用，帮助读者快速了解每个章节的主要内容，掌握完整的知识体系。另外，这种方式非常适合在高校教学中使用，能够完全与教材同步，方便学生课后复习及课前预习，可以有效提高教学效果。

这套丛书是中软国际多年行业经验的积累和沉淀，也是众多编者智慧与汗水的结晶，一定能在校企合作的道路上发挥积极长远的作用。

国家示范性软件学院建设工作办公室副主任
北京交通大学软件学院院长



目 录

第一部分 Struts2 框架	(1)
第 1 章 Struts2 快速入门	(2)
1.1 Struts2 概述	(2)
1.2 Struts2 简单实例	(4)
1.3 实例的运行过程	(10)
1.4 Struts2 的特点	(12)
1.5 教材案例准备	(13)
1.6 本章小结	(19)
第 2 章 Struts2 的控制器	(20)
2.1 过滤器	(20)
2.2 拦截器	(21)
2.3 Action	(26)
2.4 本章小结	(28)
第 3 章 自定义拦截器	(29)
3.1 编写拦截器类	(29)
3.2 配置使用拦截器	(30)
3.3 本章小结	(31)
第 4 章 Struts2 框架的 Action	(32)
4.1 Action 接口	(32)
4.2 Action 类中的方法	(33)
4.3 将多个 Action 类“合并”	(36)
4.4 Action 类的不同调用方式	(38)
4.5 本章小结	(41)
第 5 章 Action 类与 Servlet API	(42)
5.1 使用 ActionContext	(42)
5.2 使用 ServletActionContext	(45)
5.3 IoC 方式	(46)
5.4 ActionContext 使用实例	(47)
5.5 本章小结	(49)
第 6 章 Action 类封装请求参数	(50)
6.1 Field-Driven 方式	(50)
6.2 Model-Driven 方式	(51)
6.3 本章小结	(53)
第 7 章 Action 类的属性	(54)
7.1 Action 是多实例的	(54)

7.2	Action 属性封装请求参数	(56)
7.3	Action 属性传递对象	(57)
7.4	Action 属性封装 Action 配置参数	(58)
7.5	JSP 文件中如何获得 Action 属性	(59)
7.6	本章小结	(60)
第 8 章	值栈与 OGNL	(61)
8.1	值栈	(61)
8.2	OGNL	(62)
8.3	本章小结	(65)
第 9 章	国际化	(66)
9.1	哪些内容需要国际化	(66)
9.2	Struts2 国际化资源文件	(67)
9.3	struts.properties 文件	(68)
9.4	使用国际化资源文件	(69)
9.5	使用多个国际化资源文件	(71)
9.6	本章小结	(74)
第 10 章	输入校验	(75)
10.1	ActionSupport 类	(75)
10.2	JSP 中显示校验信息	(76)
10.3	input 视图	(77)
10.4	手工校验方式	(77)
10.5	Action 中使用国际化资源文件	(82)
10.6	校验器校验	(86)
10.7	类型转换	(89)
10.8	本章小结	(91)
第 11 章	Struts2 标签	(92)
11.1	Struts2 标签库概述	(92)
11.2	表单 UI 标签	(93)
11.3	非表单 UI 标签	(95)
11.4	控制标签	(96)
11.5	数据标签	(97)
11.6	本章小结	(98)
第 12 章	Struts2 异常处理	(99)
12.1	Model 层抛出异常	(99)
12.2	Action 中直接捕获异常	(100)
12.3	在 struts.xml 中声明异常映射	(101)
12.4	本章小结	(102)
第 13 章	Struts2 的 Ajax 支持	(103)
13.1	Ajax 简介	(103)

13.2	Ajax 简单案例	(104)
13.3	struts2 中对 Ajax 的支持	(109)
13.4	本章小结	(113)
第 14 章	配置文件总结	(114)
14.1	web.xml	(114)
14.2	struts.xml	(114)
14.3	struts.properties	(118)
14.4	本章小结	(118)
第二部分	Hibernate 框架	(119)
第 1 章	Hibernate 快速入门	(120)
1.1	Hibernate 概述	(120)
1.2	常用 API	(124)
1.3	Eclipse 中开发 Hibernate 应用	(126)
1.4	本章小结	(136)
第 2 章	Hibernate 核心知识点	(137)
2.1	持久化类	(137)
2.2	对象状态	(138)
2.3	Hibernate 属性配置	(139)
2.4	ORM 映射基础	(141)
2.5	HQL 语言	(143)
2.6	本章小结	(146)
第 3 章	HQL 语言详解	(147)
3.1	from 子句	(147)
3.2	select 子句	(148)
3.3	聚集函数	(149)
3.4	where 子句	(150)
3.5	order by 子句	(152)
3.6	group by 子句	(152)
3.7	子查询	(153)
3.8	本章小结	(153)
第 4 章	粒度设计	(154)
4.1	基于设计的粒度设计	(154)
4.2	基于性能的粒度设计	(157)
4.3	本章小结	(160)
第 5 章	关联关系映射	(161)
5.1	关联的方向与数量	(161)
5.2	一对多/多对一	(162)
5.2.1	基于主外键的一对多/多对一关联	(163)
5.2.2	基于连接表的一对多/多对一关联	(167)

5.3	一对一关联	(171)
5.3.1	基于主键的一对一关联	(171)
5.3.2	基于唯一外键的一对一关联	(175)
5.4	多对多关联	(177)
5.5	关联映射配置文件	(182)
5.6	连接查询	(183)
5.7	本章小结	(186)
第 6 章	继承关系映射	(187)
6.1	本章实例准备	(187)
6.2	TPS (Table Per SubClass)	(189)
6.3	TPH (Table Per Class Hierarchy)	(192)
6.4	TPC (Table Per Concrete Class)	(194)
6.5	多态查询	(197)
6.6	本章小结	(198)
第 7 章	Hibernate 性能提升	(199)
7.1	批量操作	(199)
7.2	延迟加载	(200)
7.3	batch-size 属性	(202)
7.4	本章小结	(204)
第 8 章	整合 Struts/Hibernate	(205)
第三部分	Spring 框架	(208)
第 1 章	Spring 概述	(209)
1.1	Spring 框架的模块	(209)
1.2	使用 Eclipse 开发 Spring 应用	(210)
1.3	本章小结	(212)
第 2 章	IoC (控制反转)	(213)
2.1	什么是 IoC	(213)
2.2	IoC 的使用	(216)
2.3	需要使用 IoC 的对象	(218)
2.4	如何实例化 bean	(220)
2.5	setter 注入和构造器注入	(222)
2.6	属性值的配置方式	(223)
2.7	集合类型属性配置	(225)
2.8	bean 的作用域	(228)
2.9	bean 的初始化和析构	(230)
2.10	IoC 的技术基础	(231)
2.10.1	反射技术	(231)
2.10.2	JavaBean 自省技术	(233)
2.11	IoC 使用实例 (教材案例)	(235)

2.12 本章小结	(238)
第3章 AOP (面向切面编程)	(239)
3.1 AOP 中的术语	(239)
3.2 Spring AOP 快速入门	(240)
3.3 不同类型的 Advice	(243)
3.4 使用 Advisor	(248)
3.5 Spring AOP 的技术基础	(251)
3.5.1 代理模式	(251)
3.5.2 动态代理	(253)
3.6 本章小结	(254)
第4章 Spring 整合 Struts2	(255)
4.1 导入必要的类库	(255)
4.2 配置 web.xml 文件	(256)
4.3 修改 Struts2 框架的 Action 类	(257)
4.4 修改 struts.properties 文件	(258)
4.5 修改 struts.xml 文件	(259)
4.6 修改 applicationContext.xml	(260)
4.7 本章小结	(261)
第5章 Spring 整合 JDBC	(262)
5.1 为什么要整合 JDBC	(262)
5.2 Spring JDBC 包结构	(263)
5.3 JdbcTemplate 类	(264)
5.4 获得 JdbcTemplate 实例	(266)
5.5 JdbcTemplate 使用实例	(268)
5.6 本章小结	(270)
第6章 Spring 整合 Hibernate	(271)
6.1 创建 SessionFactory	(271)
6.2 HibernateTemplate 类	(273)
6.3 Spring 整合 Hibernate 的实例	(274)
6.4 本章小结	(279)
第7章 Spring 中的事务管理	(280)
7.1 平台事务管理器接口	(280)
7.2 编程式事务管理	(281)
7.3 声明式事务管理	(283)
7.4 本章小结	(286)
第8章 SSH 整合实例	(287)
附录 A 企业关注的技能	(289)
第一部分 Struts2 框架	(289)
第二部分 Hibernate 框架	(295)
第三部分 Spring 框架	(297)

第一部分 Struts2 框架

如果精通 Servlet/JSP 等组件技术，就可以使用 JavaEE 技术开发企业级 Web 应用。然而，Web 应用中有很多通用的功能，如页面跳转、输入页面的信息回显、用户输入信息校验等，如果使用 Servlet/JSP 技术实现这些通用功能，需要很多重复的代码，从而导致代码冗余、维护困难等后果。随着技术的发展，很多社区和开源组织开发了不同的 MVC 框架。

目前有很多 MVC 框架，其中 Struts 框架是一个被广泛使用的开源框架。本教材的第一部分将学习 Struts 框架的第二个版本，即 Struts2。Struts2 的前身并不是 Struts1，而是另一个优秀的 MVC 框架 WebWork，Struts2 综合了 Struts1 和 WebWork 两大框架的优点。

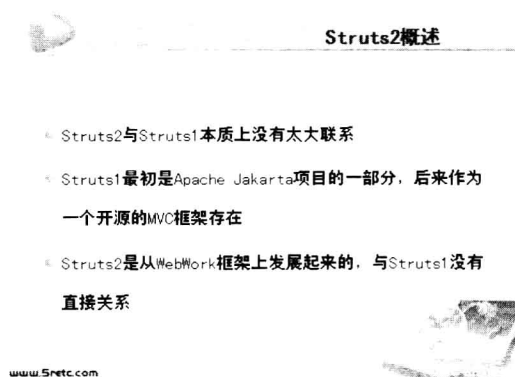
本部分从 Struts2 框架的工作原理开始学习，首先通过简单例子，帮助读者快速上手。Struts2 框架的最大改变是 MVC 中的控制器部分，Struts2 的控制器包括过滤器、拦截器、Action 三种组件，教材中将深入学习控制器的配置使用以及如何进行自定义开发。其中，Action 是 Struts2 应用中使用最为广泛的控制器，被称为业务控制器。本部分将详细学习 Action 有关的知识，包括如何创建 Action 类、如何配置 Action、Action 类如何封装请求参数，以及 Action 类与 Servlet API 交互等。Struts2 框架对国际化、输入校验、类型转换、Ajax 技术都进行了支持，本部分将结合实例学习相关知识点。Struts2 可以支持多种视图技术，包括 JSP、FreeMarker、Velocity，框架对这些模板技术提供了功能强大的标签库，使得视图开发更为便捷。

本部分并不想“面面俱到”地罗列 Struts2 的所有功能和特征，而是尽量简洁地讲解在实际开发中常用的技术点，以帮助读者在短时间内掌握 Struts2 框架中的核心功能，能够快速胜任 Struts2 应用开发。

第 1 章 Struts2 快速入门

Struts2 是在 WebWork 基础上发展起来的一个新的 MVC 框架，与 Struts1 没有太多关系。本章将通过简单案例帮助读者快速理解 Struts2 框架的工作原理、开发部署步骤以及主要优点。

1.1 Struts2 概述



从名称上看，Struts2 应该是 Struts1 版本的扩展和升级。实际上，Struts2 和 Struts1 没有太多的关系。

Struts1 最初是 Apache Jakarta 项目的一部分，后来作为一个开源的 MVC 框架存在。Struts1 曾经被很多 Web 应用采用，作为构建 MVC 的基础框架使用。Struts1 最大的特点是提供了 JSP 标记库以及页面导航。

Struts2 是从 WebWork 框架上发展起来的。WebWork 是一个很优秀的 MVC 框架，然而，由于是一个新的框架，在一段时间内并没有被广泛应用。后来，Struts 和 WebWork 社区决定将二者合并，推出 Struts2 框架。Struts2 兼具 Struts1 和 WebWork 的优点，从而得到了广泛的使用。

为了尽快对 Struts2 框架有较为全面的了解，本节首先介绍 Struts2 的工作原理。Struts2 工作原理比较复杂，图 1-1-1 是官方提供的 Struts2 工作原理图。

下面结合图 1-1-1 展示的 Struts2 工作原理，对 Struts2 的基本工作过程进行总结：

- (1) 客户端向服务器端提交请求，容器初始化 `HttpServletRequest` 请求对象。
- (2) 请求对象被一系列的 Servlet 过滤器过滤。Struts2 中的过滤器有三种，如下所述：
 - ① `ActionContextCleanUp` 过滤器：是一个可选的过滤器，主要用来集成其他框架。
 - ② 其他插件的核心过滤器：如 `SiteMesh` 插件的过滤器。
 - ③ `FilterDispatcher` 过滤器：是 Struts2 API 中提供的过滤器，必须使用。
- (3) `FilterDispatcher` 过滤器调用 `ActionMapper`，决定该请求是否需要调用某个 Action。

(4) 如果请求需要调用某个 Action，`ActionMapper` 将通知 `FilterDispatcher` 过滤器把请求的处理交给 `ActionProxy` 来处理。

(5) `ActionProxy` 通过 `Configuration Manager` 解析框架的配置文件 `struts.xml`，找到需要调用的 Action 类。

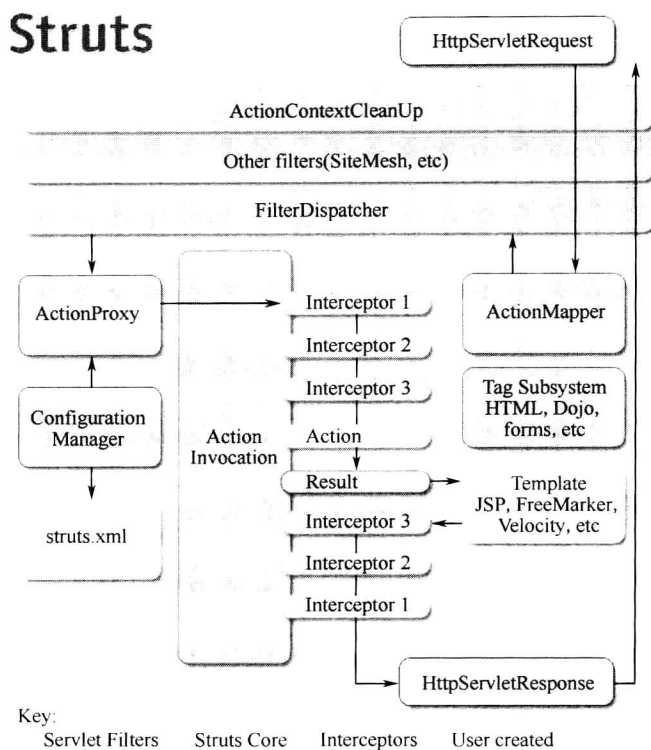


图 1-1-1 Struts2 工作原理

(6) **ActionProxy** 将创建一个 **ActionInvocation** 实例。

(7) **ActionInvocation** 实例使用命令模式回调 **Action** 中的 **execute** 方法，**Action** 调用业务逻辑类完成业务逻辑。在调用 **Action** 的前后，将调用该 **Action** 涉及的相关拦截器 (**Interceptor**)。

(8) **Action** 执行完毕后，**ActionInvocation** 根据 **struts.xml** 中的配置找到对应的返回结果 (称为 **Result**)。返回结果通常是 **JSP**、**FreeMarker** 等模板文件。

在阐述 **Struts2** 工作原理的过程中，涌现了很多新名词，如拦截器、返回结果等，读者暂时不需要深入考虑，在后续章节将继续详细学习。

为了帮助读者快速入门 **Struts2** 开发，下面总结 **Struts2** 应用开发过程中开发人员需要做的主要工作。

(1) **Model** 层开发。

Struts2 框架对于 **Model** 层没有特别要求，也没有特殊支持，**Model** 层可以使用任何开发技术实现，如 **JavaSE** 的普通 **Java** 类 (**POJO**)、**EJB** 组件、**WebService** 等。**Model** 层的业务逻辑往往使用 **Action** 类调用。

(2) 在 **web.xml** 中配置 **FilterDispatcher**。

FilterDispatcher 是 **Struts2** 中的核心控制器，在 **Struts2 API** 中已经定义。在 **Struts2** 应用中，必须通过 **web.xml** 配置 **FilterDispatcher** 过滤器，代码如下所示：

```
<filter>
  <filter-name>FilterDispatcher</filter-name>
  <filter-class>org.apache.struts2.dispatcher.FilterDispatcher</filter-class>
```

```

</filter>
<filter-mapping>
    <filter-name>FilterDispatcher</filter-name>
    <url-pattern>/*</url-pattern>
</filter-mapping>

```

上述配置中将 FilterDispatcher 映射到/*路径，那么客户端对服务器端的任何路径的请求，都将被 FilterDispatcher 过滤。

(3) 开发 Action 类。

Action 类是一个符合一定命名规范的 JavaSE 类，不是 Servlet。Action 实现的功能与 Servlet 类似，承担调用 Model、根据返回结果决定页面导航等职责。在 Struts2 应用中，需要编写大量的 Action 类，作为业务控制器使用。

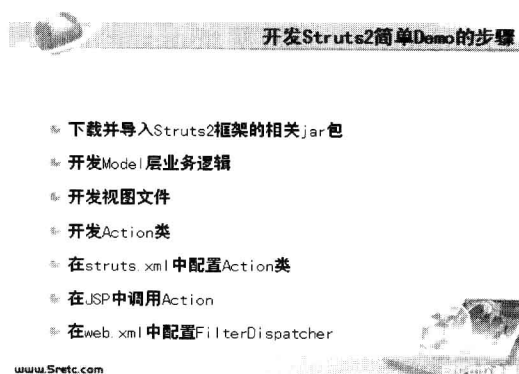
(4) 拦截器 (Interceptor) 的配置或自定义开发。

拦截器用来在 Action 类的前后执行一些通用的操作，Struts2 API 中已提供了常用的拦截器，只需要进行配置即可使用。如果应用中需要自定义一些通用操作，需要自定义拦截器，并通过配置使用。

(5) 开发视图。

Struts2 可以支持多种视图技术，包括 JSP、FreeMarker、Velocity。目前，使用较多的仍然是 JSP 技术，本教材中也采用 JSP 技术实现视图。Struts2 框架提供了强大的 JSP 标记库，可以便捷地开发 JSP 页面。

1.2 Struts2 简单实例



通过上一节的学习，读者已经了解 Struts2 基本概念以及工作原理，并且对 Struts2 应用开发中的主要工作也有所了解。本节将构建简单的 Struts2 实例，该实例实现如下逻辑：用户在 index.jsp 中输入用户名和密码，如果用户名和密码是 ETC 和 123，则登录成功，显示欢迎页面 welcome.jsp；否则，登录失败，跳转到 index.jsp 页面。该实例不注重业务逻辑，主要目的是帮助读者快速掌握开发 Struts2 应用的步骤。

(1) 到 Struts 官方网站下载 Struts 的 jar 包 (<http://struts.apache.org/>)。

(2) 将下载到的 jar 包导入到 Web 工程中。

Struts2 框架有很多 jar 包，某些包是和其他插件有关的。如果将全部 jar 包都导入到工程中，需要同时导入其他插件的 jar 包，否则将出现错误。如果不需要使用其他插件，仅导入下面 5 个 jar 包即可实现 Struts2 的基本功能，如图 1-1-2 所示。

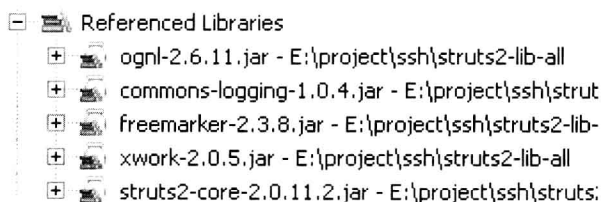


图 1-1-2 导入 Struts2 的 5 个核心包

(3) 开发 Model 层业务逻辑。

导入所需要的 jar 包后，下面使用 Java 类实现 Model 层的登录逻辑，代码如下所示：

```
public class LoginService {
    public boolean login(String custname,String pwd){
        if(custname.equals("ETC")&&pwd.equals("123")){
            return true;
        }else{
            return false;
        }
    }
}
```

(4) 开发视图文件。

完成业务逻辑后，进一步可以开发视图文件，视图使用 JSP 实现。JSP 文件中可以使用 Struts2 框架提供的 JSP 标记库，使用 Struts2 的标记与使用 JSTL 的标记步骤相同。Struts2 标记库只有一个 tld 文件，存在于 struts2-core.jar 包的 META-INF 目录下，如图 1-1-3 所示。



图 1-1-3 struts-tags.tld 文件所在目录

首先编写 index.jsp 文件，用来输入用户名和密码进行登录，代码如下所示：

```
<body>
  <%@taglib uri="/struts-tags" prefix="s" %>
  <s:form action="">
    Input your custname:<s:textfield name="custname"></s:textfield><br>
    Input your password:<<s:password name="pwd"></s:password><br>
    <s:submit value="Login"></s:submit>
  </s:form>
```



```
</body>
```

上述代码中使用了 Struts2 框架的标记库来构建表单，如<s:textfield>表示文本框，<s:password>表示密码框。

接下来编写 welcome.jsp 文件，当登录成功后，显示欢迎信息，代码如下所示：

```
<body>
    Welcome,${param.custname}
</body>
```

欢迎页面将显示登录的用户名，使用 EL 显示请求参数 custname 的值。

(5) 定义 Action 类，调用业务逻辑，返回结果视图。

业务逻辑和视图都完成后，需要创建控制器，将二者联系起来，Action 是 Struts2 使用的业务控制器。Action 类不需要继承或实现任何父类或接口，只要遵守某些命名规范即可：如果该 Action 类是通过表单提交调用，且 Action 类中需要使用表单提交的请求参数，那么必须在 Action 类中声明与表单域的名字对应的变量，并为变量提供 getters/setters 方法；Action 类中必须有一个 public String execute(){} 形式的方法，该方法将在访问 Action 时被 Struts2 框架自动调用，实现控制逻辑。

下面创建 LoginAction 类，调用 Model 中的登录逻辑，并根据登录结果不同而返回不同的结果，代码如下所示：

```
public class LoginAction {
    private String custname;
    private String pwd;
    public String getCustname() {
        return custname;
    }
    public void setCustname(String custname) {
        this.custname = custname;
    }
    public String getPwd() {
        return pwd;
    }
    public void setPwd(String pwd) {
        this.pwd = pwd;
    }
    public String execute(){
        LoginService ls=new LoginService();
        boolean flag=ls.login(custname, pwd);
        if(flag){
            return "success";
        }else{
            return "fail";
        }
    }
}
```

```
}}}
```

可见，由于 LoginAction 类通过 index.jsp 中的表单提交请求，且 LoginAction 类中需要使用登录表单的请求参数值，所以 LoginAction 类中声明了与 index.jsp 中表单元素对应的属性 custname 和 pwd，并提供了 getters 和 setters 方法。在 LoginAction 类的 execute 方法中，可以直接使用请求参数 custname 和 pwd，不需要像 Servlet 中那样使用 request.getParameter 方法获取。请求参数的赋值过程将在后面章节学习。execute 方法通过调用 LoginService 类中的 login 方法进行登录验证，登录成功返回 success 字符串，否则返回 fail 字符串。

(6) 在 struts.xml 中配置 Action 类。

前面的步骤已经创建了 Action 类 LoginAction，但必须在 struts.xml 中进行配置才能使用。在 Web 工程的 src 文件夹下，创建 struts.xml 文件，如图 1-1-4 所示。

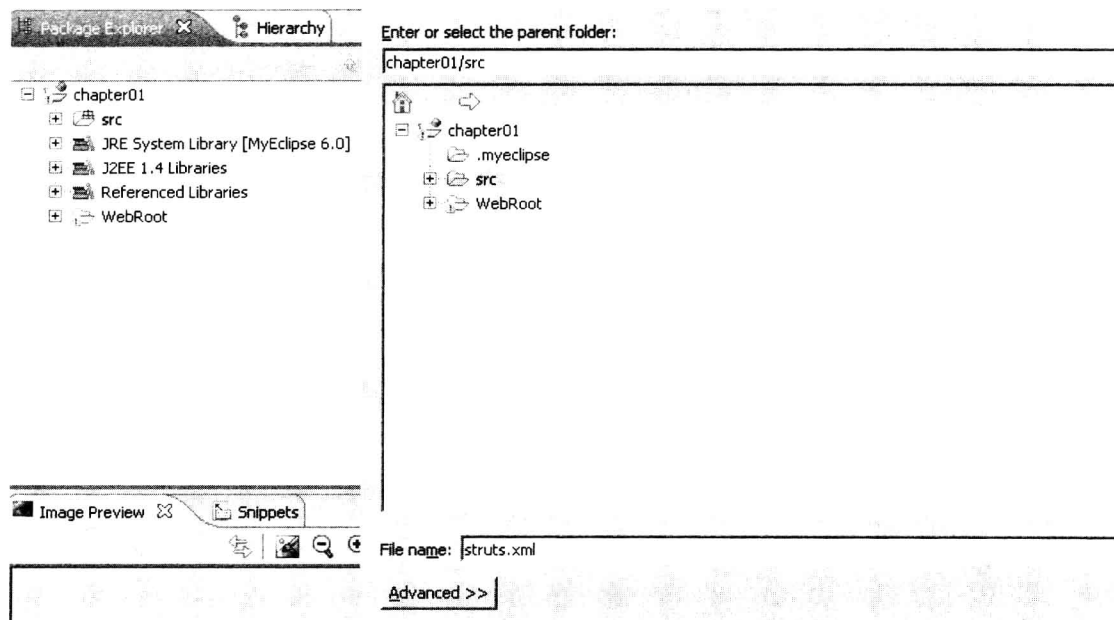


图 1-1-4 在 src 目录下创建 struts.xml 文件

框架在加载自定义的 struts.xml 文件前，会先加载框架自带的配置文件 struts-default.xml，所以首先了解一下 struts-default.xml 文件。struts-default.xml 文件存在于 struts2-core.jar 包中，定义了 Struts2 的基本配置信息，定义了名字为 struts-default 的包。struts-default.xml 的部分配置信息如下：

```
<package name="struts-default" abstract="true">
<result-types>
<result-type name="chain" class="com.opensymphony.xwork2.ActionChainResult"/>
<result-type name="dispatcher" class="org.apache.struts2.dispatcher.ServletDispatcherResult" default="true"/>
<result-type name="freemarker" class="org.apache.struts2.views.freemarker.FreemarkerResult"/>
<result-type name="httpheader" class="org.apache.struts2.dispatcher.HttpHeaderResult"/>
<result-type name="redirect" class="org.apache.struts2.dispatcher.ServletRedirectResult"/>
```