

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

Java Web 程序设计教程

Java Web Development

范立锋 林果园 编著

- 内容讲解深入浅出，通俗易懂，实用性强
- 知识体系搭配合理，内容翔实而全面
- 案例驱动开发，注重实践能力与工程思想的培养



精品系列



人民邮电出版社
POSTS & TELECOM PRESS

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

Java Web 程序设计教程

Java Web Development

范立锋 林果园 编著



精品系列

人民邮电出版社

北京

图书在版编目 (CIP) 数据

Java Web程序设计教程 / 范立锋, 林果园编著. —
北京: 人民邮电出版社, 2010.4
21世纪高等学校计算机规划教材
ISBN 978-7-115-21974-9

I. ①J… II. ①范… ②林… III. ①
JAVA语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2009)第244140号

内 容 提 要

本书介绍使用 Java 语言开发 Web 应用的主流技术。首先, 从基本开发技术入手, 讲解了 JSP、Servlet、SQL 以及 JDBC 的基础概念及应用方法。然后, 逐步过渡到框架技术的讲解, 包括 Struts 2 框架技术应用、Hibernate 框架技术应用、Spring 框架技术应用以及 3 个框架的整合应用方式。每章为读者配备了简明而又实用的示例, 在书的最后通过一个完整的项目开发案例对所学技术进行总结和应用。

本书可作为普通高等院校计算机及相关专业课程教材, 同时也可作为 Java 编程爱好者及开发人员的参考用书。

21 世纪高等学校计算机规划教材

Java Web 程序设计教程

-
- ◆ 编 著 范立锋 林果园
责任编辑 刘 博
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市海波印务有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 22
字数: 578 千字
印数: 1-3 000 册
- 2010 年 4 月第 1 版
2010 年 4 月河北第 1 次印刷

ISBN 978-7-115-21974-9

定价: 36.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223
反盗版热线: (010)67171154

出版者的话

计算机应用能力已经成为社会各行业最重要的工作要求之一，而计算机教材质量的好坏会直接影响人才素质的培养。目前，计算机教材出版市场百花争艳，品种急剧增多，要从林林总总的教材中挑选一本适合课程设置要求、满足教学实际需要的教材，难度越来越大。

人民邮电出版社作为一家以计算机、通信、电子信息类图书与教材出版为主的科技教育类出版社，在计算机教材领域已经出版了多套计算机系列教材。在各套系列教材中涌现出了一批被广大一线授课教师选用、深受广大师生好评的优秀教材。老师们希望我社能有更多的优秀教材集中地呈现在老师和读者面前，为此我社组织了这套“21世纪高等学校计算机规划教材-精品系列”。

“21世纪高等学校计算机规划教材-精品系列”具有下列特点。

(1) 前期调研充分，适合实际教学需要。本套教材主要面向普通本科院校的学生编写，在内容深度、系统结构、案例选择、编写方法等方面进行了深入细致的调研，目的是在教材编写之前充分了解实际教学的需要。

(2) 编写目标明确，读者对象针对性强。每一本教材在编写之前都明确了该教材的读者对象和适用范围，即明确面向的读者是计算机专业、非计算机理工类专业还是文科专业的学生，尽量符合目前普通高等教学计算机课程的教学计划、教学大纲以及发展趋势。

(3) 精选作者，保证质量。本套教材的作者，既有来自院校的一线授课老师，也有来自IT企业、科研机构等单位的资深技术人员。通过他们的合作使老师丰富的实际教学经验与技术人员丰富的实践工程经验相融合，为广大师生编写出适合目前教学实际需求、满足学校新时期人才培养模式的高质量教材。

(4) 一纲多本，适应面宽。在本套教材中，我们根据目前教学的实际情况，做到“一纲多本”，即根据院校已学课程和后续课程的不同开设情况，为同一科目提供不同类型的教材。

(5) 突出能力培养，适应人才市场要求。本套教材贴近市场对于计算机人才的能力要求，注重理论技术与实际应用的结合，注重实际操作和实践动手能力的培养，为学生快速适应企业实际需求做好准备。

(6) 配套服务完善，共促提高。对于每一本教材，我们在教材出版的同时，都将提供完备的PPT课件，并根据需要提供书中的源程序代码、习题答案、教学大纲等内容，部分教材还将在作者的配合下，提供疑难解答、教学交流等服务。

在本套教材的策划组织过程中，我们获得了来自清华大学、北京大学、人民大学、浙江大学、吉林大学、武汉大学、哈尔滨工业大学、东南大学、四川大学、上海交通大学、西安交通大学、电子科技大学、西安电子科技大学、北京邮电大学、北京林业大学等院校老师的大力支持和帮助，同时获得了来自信息产业部电信研究院、联想、华为、中兴、同方、爱立信、摩托罗拉等企业和科研单位的领导和技术人员的积极配合。在此，人民邮电出版社向他们表示衷心的感谢。

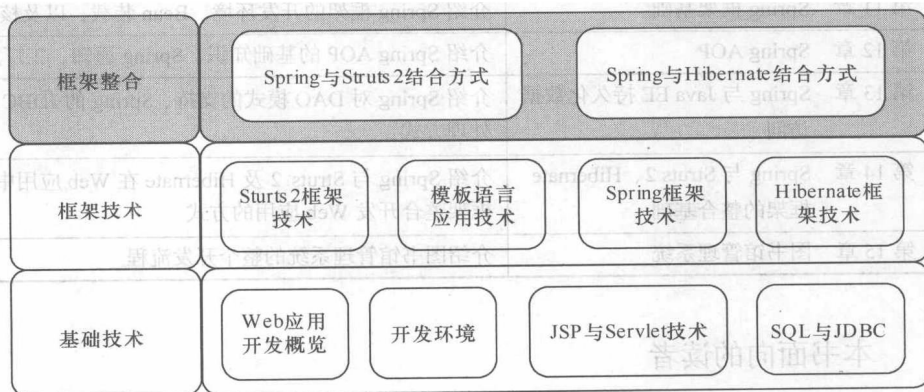
我们相信，“21世纪高等学校计算机规划教材-精品系列”一定能够为我国高等院校计算机课程教学做出应有的贡献。同时，对于工作欠缺和不妥之处，欢迎老师和读者提出宝贵的意见和建议。

Java 语言以其简单易学、适用范围广泛等优点，成为了近年来最为流行的编程语言之一。尤其在 Web 应用开发方面，Java 更具有得天独厚的优势。随着 Java 语言的推广和应用，各种针对 Web 开发的框架技术也应运而生。本书从 Java Web 应用开发的基础技术入手，重点讲解了 Struts 2，Spring 以及 Hibernate 框架在实际开发中的应用技巧。

本书是作者在总结了多年开发经验与成果的基础上编写的。书中全面、翔实地介绍了 Java Web 应用开发所需的各种知识和技巧。通过本书的学习，读者可以快速、全面地掌握使用框架开发 Web 应用程序的方法，并可达到融会贯通、灵活运用之目的。

Java Web 应用知识体系

Java Web 应用的知识体系如下图所示。



本书特点

(1) 教材知识体系结构合理。知识安排强调整体性和系统性，知识表达强调层次性和有序性，便于读者学习和理解。

(2) 书中采用程序结构、页面交互图、表格等多种方式给出了问题描述及解决的全部流程，使读者从多个角度来更好地理解问题。

(3) 重点难点醒目。书中采取“注意”、“说明”、“技巧”的特殊体例，将各知识点中的重点内容或相关应用技巧突现出来，使读者能快速抓住问题关键。

(4) 注重理论性与实践性的结合。本书在每章知识点讲解之后，都给出一个综合应用本章技术的案例，使读者在实践基础上加深对理论的认识及掌握。

(5) 综合案例实用性强。本书最后给出的综合性案例，详细介绍了从需求分析、数据库设计到系统设计、编码设计的整个流程。读者通过此案例不仅能够掌握书中知识点的综合应用，更能学会软件开发的一般流程。

本书结构

本书围绕 Java Web 应用开发循序渐进地介绍相关知识, 全书共分 15 章, 各章主要内容如下表所示。

章 名	主 要 内 容
第 1 章 Web 应用开发简介	介绍 Web 应用的概念, Java 在 Web 应用开发方面的相关技术及集成开发环境
第 2 章 Java EE 运行及开发环境	介绍构建 Java EE 开发环境所需各种开发工具、软件
第 3 章 JSP 和 Servlet	介绍 JSP 及 Servlet 相关的基础技术点及其应用
第 4 章 SQL 与 JDBC	介绍 SQL 与 JDBC 相关的基础技术点及其应用
第 5 章 Struts 2 框架基础	介绍 Struts 2 框架的工作原理、基础知识及简单应用
第 6 章 Struts 2 高级应用	介绍 Struts 2 框架的高级应用技巧
第 7 章 Struts 2 中应用模板语言	介绍 FreeMarker 和 Velocity 这两种模板语言的使用方法
第 8 章 Hibernate 框架基础	介绍 Hibernate 框架工作原理、工作流程及简单应用
第 9 章 Hibernate 查询	介绍 Hibernate 框架的数据检索策略、查询方式及各种关联查询操作
第 10 章 Hibernate 性能优化	介绍 Hibernate 事务与并发操作及缓存原理和实现
第 11 章 Spring 框架基础	介绍 Spring 框架的开发环境、Bean 装载, 以及核心控制器的应用
第 12 章 Spring AOP	介绍 Spring AOP 的基础知识、Spring 通知、工厂及代理工厂技术
第 13 章 Spring 与 Java EE 持久化数据访问	介绍 Spring 对 DAO 模式的支持、Spring 的 JDBC 及 Spring 中的事务处理方式
第 14 章 Spring 与 Struts 2、Hibernate 框架的整合基础	介绍 Spring 与 Struts 2 及 Hibernate 在 Web 应用中的作用, 以及 3 个框架整合开发 Web 应用的方式
第 15 章 图书馆管理系统	介绍图书馆管理系统的整个开发流程

本书面向的读者

本书可作为普通高等院校计算机及相关专业课程教材, 也可作为具有一定 Java 语言编程基础, 并想快速掌握 SSH 开发技术的初学者及开发人员的参考用书。

技术支持

本书实例开发中用到的程序源代码, 可以在“人民邮电出版社教学服务与资源网 (www.ptpedu.com.cn)”中免费下载, 以供读者学习和使用。

编 者

2009 年 9 月

目 录

第 1 章 Web 应用开发简介	1	第 3 章 JSP 和 Servlet	33
1.1 何为 Web 应用	1	3.1 开发第一个 JSP+Servlet 应用	33
1.1.1 Web 的概念及发展	1	3.1.1 创建工程	33
1.1.2 Web 应用程序	2	3.1.2 编写程序	34
1.2 使用 Java 开发 Web 应用	3	3.1.3 部署应用	35
1.2.1 面向对象的编程语言	3	3.2 认识 JSP	36
1.2.2 丰富的框架技术	4	3.2.1 JSP 的工作原理	37
1.2.3 XML、CSS 的应用	4	3.2.2 JSP 注释方式	37
1.2.4 使用 JavaScript 与 Ajax 提升用户 体验	7	3.2.3 JSP 声明方式	38
1.3 多种集成开发环境	9	3.2.4 JSP 表达式的应用	39
1.3.1 集成开发环境简介	9	3.2.5 JSP 的脚本段	39
1.3.2 Web 应用服务器说明	11	3.2.6 JSP 的编译指令	40
本章小结	14	3.2.7 JSP 的动作指令	41
课后练习	14	3.2.8 JSP 的内置对象	43
第 2 章 Java EE 运行及开发 环境	15	3.3 认识 Servlet	46
2.1 JDK 的下载与安装	15	3.3.1 Servlet 的开发	46
2.1.1 JDK 的下载与安装	15	3.3.2 使用 HttpServlet	47
2.1.2 配置环境变量	17	3.3.3 Servlet 的生命周期	49
2.2 Tomcat 的下载与安装	19	3.3.4 load-on-startup Servlet	50
2.2.1 下载并安装 Tomcat 服务器	19	3.4 自定义标签库	51
2.2.2 基本配置	22	3.5 预处理用户请求: Filter	53
2.2.3 服务器页面介绍	23	3.6 使用 Listener	56
2.3 Eclipse 的下载与安装	24	3.7 项目实战——用户登录	59
2.3.1 Eclipse 的下载与安装	24	本章小结	62
2.3.2 熟悉 Eclipse 开发环境	24	课后练习	63
2.4 项目实战——第一个 Java EE 应用: HelloWorld	28	第 4 章 SQL 与 JDBC	64
2.4.1 开始创建第一个 Java EE 应用	28	4.1 SQL	64
2.4.2 熟悉 HelloWorld 框架结构	29	4.1.1 SQL 概述	64
本章小结	32	4.1.2 DDL 与 DML 简介	64
课后练习	32	4.1.3 SQL 使用方法	65
		4.2 JDBC	67
		4.2.1 JDBC 概述	67
		4.2.2 JDBC 驱动程序	68

4.2.3 使用 JDBC 读取数据	69	6.3.4 自定义校验器	130
4.3 项目实战——存储图书信息	73	6.4 国际化	131
本章小结	78	6.4.1 国际化实现原理	131
课后练习	79	6.4.2 准备国际化资源文件	131
第 5 章 Struts 2 框架基础	80	6.4.3 调用国际化资源文件	134
5.1 MVC 框架	80	6.5 上传和下载	135
5.1.1 Model 1 与 Model 2	80	6.5.1 文件上传的实现原理	135
5.1.2 MVC 设计模式	81	6.5.2 Struts 2 文件上传实现方式	136
5.1.3 Struts 2 框架的 MVC 架构	82	6.5.3 Struts 2 文件下载实现方式	141
5.2 Struts 2 概览	84	本章小结	143
5.2.1 Struts 2 的工作流程	84	课后练习	144
5.2.2 Struts 2 的简单应用	85	第 7 章 Struts 2 中应用模板语言	145
5.3 Struts 2 基础	87	7.1 模板语言简介	145
5.3.1 Action 详解	88	7.2 应用 Velocity	146
5.3.2 结果与视图	91	7.2.1 Velocity 基础	146
5.3.3 struts.xml 的配置	94	7.2.2 Struts 2 对 Velocity 的支持	153
5.4 值栈与 OGNL 表达式	100	7.3 应用 FreeMarker	155
5.5 Struts 2 的标签库	103	7.3.1 FreeMarker 基础	155
5.5.1 控制标签	103	7.3.2 Struts 2 整合 FreeMarker	166
5.5.2 数据标签	104	7.3.3 使用 Struts 2 标签设计模板	170
5.5.3 表单标签	105	7.4 FreeMarker 与 Velocity 的比较	171
5.5.4 非表单 UI 标签	107	本章小结	171
本章小结	108	课后练习	172
课后练习	109	第 8 章 Hibernate 框架基础	173
第 6 章 Struts 2 高级应用	110	8.1 ORM 简介	173
6.1 拦截器	110	8.1.1 应用 ORM 的意义	173
6.1.1 拦截器工作机制	110	8.1.2 流行的 ORM 框架	174
6.1.2 拦截器及拦截器栈的应用	111	8.2 准备 Hibernate 运行环境	174
6.1.3 自定义拦截器	115	8.2.1 下载与安装 Hibernate	175
6.2 类型转换器	117	8.2.2 Hibernate 发布包介绍	175
6.2.1 Struts 2 内置类型转换器	117	8.3 认识 Hibernate	176
6.2.2 引用类型的转换方式	117	8.3.1 Hibernate 框架结构	176
6.2.3 特殊对象的类型转换	118	8.3.2 Hibernate 配置文件	177
6.2.4 类型转换的错误处理	121	8.3.3 Configuration 与 SessionFactory	178
6.3 输入校验	123	8.3.4 Session 类	179
6.3.1 输入校验的必要性	123	8.3.5 Hibernate 中的关联关系	179
6.3.2 编程方式实现输入校验	125	8.3.6 Hibernate 映射文件	180
6.3.3 应用 Struts 2 输入校验框架	128		

8.3.7 Hibernate 工作原理	182	课后练习	224
8.4 项目实战——新闻内容显示	183	第 11 章 Spring 框架基础	226
本章小结	186	11.1 Spring 框架概述	226
课后练习	187	11.1.1 认识 Spring 框架	226
第 9 章 Hibernate 查询	188	11.1.2 Spring 框架特点	226
9.1 Hibernate 的数据检索策略	188	11.1.3 Spring 框架核心架构	227
9.1.1 立即检索	188	11.2 建立 Spring 开发环境	228
9.1.2 延迟检索	189	11.2.1 下载 Spring 框架	228
9.1.3 预先检索	190	11.2.2 Spring 发布包与软件包	229
9.1.4 批量检索	191	11.2.3 创建 Spring 应用环境	230
9.2 Hibernate 的数据查询方式	193	11.3 Bean 的装配	231
9.2.1 HQL 方式	193	11.3.1 Bean 基本配置	232
9.2.2 QBC 方式	194	11.3.2 为 Bean 添加属性	232
9.2.3 原生 SQL 方式	195	11.3.3 简化配置	233
9.3 Hibernate 的关联查询	196	11.4 理解 Spring 的核心——IoC	234
9.3.1 一对一关联关系的使用	196	11.4.1 控制反转	234
9.3.2 一对多、多对一关联关系的 使用	197	11.4.2 依赖注入的 3 种方式	236
9.3.3 多对多关联关系的使用	199	11.5 BeanFactory 与 ApplicationContext	238
9.4 Hibernate 过滤	201	11.5.1 认识 BeanFactory	238
9.4.1 Session 过滤	201	11.5.2 使用 ApplicationContext	238
9.4.2 Filter 过滤	201	11.6 项目实战——Spring 问候程序	239
9.5 项目实战——客户订单管理	202	本章小结	241
本章小结	207	课后练习	241
课后练习	207	第 12 章 Spring AOP	242
第 10 章 Hibernate 性能优化	208	12.1 AOP 基础	242
10.1 Hibernate 事务与并发	208	12.1.1 AOP 与 OOP 的比较	242
10.1.1 什么是事务	208	12.1.2 AOP 的核心概念	243
10.1.2 Hibernate 中的事务处理	209	12.1.3 Java 动态代理与 AOP	244
10.1.3 在 Hibernate 中使用 JTA 事务	210	12.1.4 Spring AOP 简介	245
10.1.4 并发控制	211	12.2 使用 Spring 的通知	246
10.2 Hibernate 缓存	213	12.2.1 BeforeAdvice	246
10.2.1 缓存的工作原理	213	12.2.2 AfterReturningAdvice	248
10.2.2 应用一级缓存	214	12.2.3 MethodInterceptor	249
10.2.3 应用二级缓存	214	12.2.4 ThrowAdvice	250
10.2.4 应用第三方缓存	216	12.3 使用 Spring 的切入点	251
10.3 项目实战——借还图书	217	12.3.1 静态切入点	251
本章小结	224	12.3.2 动态切入点	253
		12.4 Spring AOP 的代理工厂	253

12.4.1 选择合适的代理	253	14.1.1 Struts 2 应用的扩展方式	277
12.4.2 ProxyFactory	254	14.1.2 Spring 插件的应用	278
12.4.3 ProxyFactoryBean	254	14.2 Spring 和 Hibernate 的整合	279
12.5 项目实战——输出日志	256	14.2.1 Spring 对 Hibernate 的支持	279
本章小结	258	14.2.2 管理 SessionFactory	279
课后练习	259	14.2.3 Hibernate 的 DAO 实现	281
第 13 章 Spring 与 Java EE 持久化数据访问	260	14.2.4 使用 HibernateTemplate	281
13.1 Spring 对 DAO 模式的支持	260	14.2.5 管理 Hibernate 事务	282
13.1.1 统一的数据访问异常	260	14.3 项目实战——学生成绩查询系统	283
13.1.2 通用的数据访问模板及抽象支持类	261	本章小结	292
13.2 Spring 的 JDBC	262	课后练习	293
13.2.1 为什么需要 JdbcTemplate	262	第 15 章 图书馆管理系统	294
13.2.2 通过 JdbcDaoSupport 使用 JdbcTemplate	263	15.1 系统功能解析	294
13.2.3 JdbcTemplate 提供的常用数据操作方法	264	15.2 系统数据库设计	295
13.3 Spring 中的事务处理	265	15.2.1 数据库分析	295
13.3.1 Spring 事务处理概述	266	15.2.2 数据表关联关系分析	298
13.3.2 编程式事务处理	266	15.3 系统框架搭建	300
13.3.3 声明式事务处理	267	15.3.1 创建工程	300
13.3.4 标注式事务处理	268	15.3.2 工程目录结构	300
13.4 项目实战——公司人事管理	269	15.4 系统代码实现	302
本章小结	276	15.4.1 数据库连接的实现	303
课后练习	276	15.4.2 工具类的实现	304
第 14 章 Spring 与 Struts 2、Hibernate 框架的整合基础	277	15.4.3 管理员登录与退出实现	306
14.1 Spring 与 Struts 2 的整合方式	277	15.4.4 系统管理实现	313
		15.4.5 图书管理实现	319
		15.4.6 读者管理实现	329
		15.4.7 借还管理实现	333
		15.4.8 前台功能实现	336
		本章小结	342

第 1 章

Web 应用开发简介

Web 是 Internet 上的一种服务,它使用超文本技术将 Internet 上的资源以页面的形式表示出来;Web 应用程序是一种使用 HTTP 作为核心通信协议,通过 Internet 让 Web 浏览器和服务端通信的计算机程序。开发 Java Web 应用中可使用的框架有多种,如 Struts 2、WebWork 等。开发应用程序的 IDE 种类也是多种多样,如 Eclipse、IntelliJ IDEA 等。Web 应用程序的运行离不开 Web 应用服务器,Java 应用中常用的 Web 应用服务器有 Tomcat、WebLogic 等。

通过本章的学习,读者将会了解 Web、Web 应用程序、开发 Java Web 应用中常用的框架、常用的集成开发环境、常用的 Web 服务器等知识。

1.1 何为 Web 应用

Internet 是一个全球计算机互连网络,同时它也是全球信息资源的总汇。Web 则是 Internet 上集文本、声音、图像、视频等多媒体信息于一身的全球信息资源网络,是 Internet 上的重要组成部分。

1.1.1 Web 的概念及发展

World Wide Web 简称为 Web,中文译为万维网,是 Internet 上的一种服务。Web 的历史最早可追溯到 1980 年 Tim Berners-Lee 负责的 Enquire (Enquire Within Upon Everything 的简称)项目(用于科学家之间方便交流信息等方面),该项目虽然和目前的 Web 不太一样,但是它已经具有了和 Web 类似的核心思想。之后,一直到 1990 年,第一台 Web 服务器“nxoc01.cern.ch”开始运行, Tim Berners-Lee 在自己编写的图形化 Web 浏览器“World Wide Web”上看到了最早的 Web 页面。1991 年, CERN (European Particle Physics Laboratory) 正式发布了 Web 技术标准,这也意味着 Web 正式登上了历史的舞台。

Web 使用超文本技术将 Internet 上的资源以页面的形式表示出来,以供用户使用,并且资源之间可以通过超链接链接起来,以达到多种资源共享的目的。同时, Web 上的资源是十分丰富的,包括图片、文本、多媒体等,因此用户可以通过 Web 来获取知识、进行娱乐、在线交易等。

Web 在组成上包括以下两部分。

- 服务器:物理设备方面指的是存放供用户访问的信息资源的远程计算机,如某个公司的网站服务器架设在操作系统为 Linux 的计算机上。软件方面指的是能根据用户的请求将信

息资源传递给用户的应用程序，如 Apache 服务器。

- 客户端: 物理设备方面指的是客户所使用的本地计算机, 如上网时使用的个人计算机。软件方面指的是能接收并显示服务器上传过来的信息资源的应用程序, 如 Internet Explorer。

发展到今天, Web 共经历了两个阶段: Web 1.0 和 Web 2.0, 其中 Web 1.0 被称为 Internet 第一代, 指的是 2003 年以前的 Internet 模式。在 Web 1.0 时代, Internet 采用的是技术创新主导模式, 比较著名的网站有早期的新浪、搜狐等。而 Web 2.0 则是以 Internet 作为平台, 利用集体智慧, 通过数据库的支持完成超越单一设备的软件及网络应用, 它将软件发布周期作为一个循环, 提升了用户的体验, 比较著名的例子有博客、播客、维基、社区、分享服务等。Web 2.0 以用户为灵魂, 它允许许多人参与, 以可读可写的模式成为了 Internet 新的发展趋势。

针对 Web 2.0 有人又提出了 Web 3.0, 不过对 Web 3.0 的争议非常大, 如将 Web 3.0 描述为一条最终通向人工智能的网络进化的道路或者构思成将整个网络转化为一系列的 3D 空间。不管 Web 3.0 最终将向何处发展, 但不可否认的是, 从 Web 诞生至今, 它不仅改变着人们联系、交流、获取知识的方式, 而且也在改变着商业的运行模式, 它已经成为人们生活、工作中非常重要的一部分。

1.1.2 Web 应用程序

最初, Web 上的内容是由静态页面组成的, 页面上包含了一些文本、图片等信息资源, 用户可以通过超链接来浏览信息。采用静态页面的缺陷非常多, 如不能与用户进行交互, 不能实时更新 Web 上的内容, 因此像搜索引擎、股市行情等许多功能无法实现。之后出现了动态页面, 即根据不同的用户或不同的时间, 呈现给用户不同的信息资源。其中, 动态内容是由 Web 应用程序来实现的。

Web 应用程序是一种使用 HTTP 作为核心通信协议, 通过 Internet 让 Web 浏览器和服务端通信的计算机程序。不同于静态网站, Web 应用程序能够动态创建页面, 实现网站和用户的实时交互。

说明: HTTP 称为超文本传输协议, 它主要用来定义客户端和服务端通信的规范。

Web 应用中的每一次数据交换都要涉及客户端和服务端两个层面。因此, Web 应用程序的开发技术分为客户端开发技术和服务器端开发技术两种。客户端开发技术如下。

- HTML: 超文本标记语言, 是 Web 的描述语言。
- CSS: 用于(增强)控制网页样式并允许将样式信息与网页内容分离的一种标记性语言。
- DOM: 文档对象模型的缩写, 使用 DOM 可以访问页面其他的标准组件。
- ActiveX: 一个集成平台, 使用 ActiveX 可轻松方便地在 Web 页中插入多媒体效果、交互式对象、复杂程序等。

- JavaScript: 客户端脚本语言, 可以为客户提供更流畅的浏览效果。

- 其他: VBScript、Applet 等。

服务器端开发技术如下。

- JSP/Servlet: 服务器端的 Java 应用程序, 可以生成动态的 Web 页面。
- PHP: 在服务器端执行的嵌入 HTML 文档的脚本语言。
- ASP: 用于构建 Windows 服务器平台上的 Web 应用程序。
- 其他: CGI、Perl、ISAPI 等。

应用程序的模式分为两种: C/S 模式和 B/S 模式。其中, C/S 模式(客户端/服务器端模

式)的程序一般能够独立运行;B/S 模式(浏览器端/服务器端模式)的应用程序一般需要通过浏览器来运行。Web 应用程序一般采用的是 B/S 模式。采用 B/S 模式的 Web 应用程序分为 3 层结构。

- 表示层:采用 Web 浏览器实现。
- 业务逻辑层:由位于 Web 服务器上的各种服务器端程序实现。
- 数据服务层:由数据库服务器提供,数据库服务器如 MySQL、SQL Server 等。

随着 Web 的普及,Web 应用程序已经成为目前最流行的应用程序。

1.2 使用 Java 开发 Web 应用

Java 提供的 JSP 和 Servlet 是开发 Web 应用中引人注目的技术,同时它的开源项目也是层出不穷,如 Web 框架 Struts、Struts 2 等,持久层框架 Hibernate、Ibatis 等,J2EE 框架 Spring,模板引擎 Velocity、FreeMarker 等。

1.2.1 面向对象的编程语言

自从第一台计算机诞生以来,程序设计方法与程序设计语言不断发展。早期,由于计算机硬件条件的限制,使程序员片面地追求高效率,而忽略了程序的可理解性、可扩充性等因素。随着计算机硬件与通信技术的发展,计算机应用领域越来越广泛,应用规模也越来越大,程序设计不再是几个程序员可以完成的任务。在这种情况下,程序员开始综合考虑程序的稳定性、扩充性、重用性、理解性等因素,正是这种需求刺激了程序设计方法与程序设计语言的快速发展。

从最初的机器语言到汇编语言,一直发展到今天的高级语言。高级语言的出现使得程序编写的效率和程序的可读性都有了一个质的飞跃。早期的高级语言(如 C、Cobol、Pascal 等)采用的是一种称为“面向过程”的编程思想,面向过程是以事件为中心,它将重点围绕在数据的使用上,在程序设计过程中通过流程图的方式辅助程序设计,然后用结构化的编程语句来编写程序。面向过程曾经在编程语言中占据着主导地位,但是随着时间的流逝,面向过程的缺点也日渐显露,如可重用性差、可维护性差、稳定性差等。

针对面向过程方面的不足,程序开发人员开始寻求解决的措施,并于 20 世纪 70 年代开发出了第一个面向对象的编程语言——Smalltalk 语言,在此之后涌现出了大批的面向对象的编程语言,如 Java、C++、Self 等。

面向对象将要解决的问题分解成各个对象,建立对象的目的不是为了完成一个步骤,而是为了描述解决问题的各个步骤中的行为。面向对象编程方式是建立在面向过程编程方式基础上,其最重要的改变在于面向对象编程中,程序将围绕被操作的对象来设计,而不是操作本身。面向对象编程方式以类作为构造程序的基本单位,具有封装、抽象、继承、多态性等特点。

面向对象对程序的灵活性和可维护性都有了很大程度的提高,并且在大型项目设计中被广泛应用。在面向对象编程语言中,非常流行的语言之一就是 Java,Java 是由 Sun Microsystems 公司于 1995 年 5 月推出的一个跨平台语言,它的一个非常重要的贡献就是用于创建动态的 Web 应用。

1.2.2 丰富的框架技术

面向对象的一个突出优点就是复用,面向对象系统获得的最大的复用方式就是使用框架。框架其实就是可重用的设计架构,应用框架强调的是软件的设计重用性和系统的可扩充性,以缩短大型应用软件系统的开发周期,提高开发质量。

下面是 Java 中常用的框架。

1. Struts

Struts 是 Apache 基金会 Jakarta 项目组的一个开源项目,是一个基于 Sun Java EE 平台的 MVC 框架,它将 Servlet 和 JSP 标签作为实现自身功能的一部分。

2. WebWork

WebWork 是由 OpenSymphony 组织开发的,是一个基于 Web 的 MVC 框架。它在运行时通过 Interceptor (拦截器) 自动应用,因此脱离了 Action 类。

3. Struts 2

Struts 2 是 Apache 基金会有一个开源项目,它建立在 Struts 框架与 WebWork 框架基础上,继承了二者的优点,是目前非常流行的一个 Web 框架。

4. Spring

Spring 是一个以 IoC 和 AOP 为核心的轻量级容器框架。它提供了一系列的 Java EE 开发解决方案,包括表示层的 Spring MVC、持久层的 Spring JDBC、业务层事务管理等众多的企业级应用技术。

5. Hibernate

Hibernate 是一个 ORM (对象关系映射) 框架,它对 JDBC 进行了轻量级的封装。通过使用 Hibernate 框架,开发人员能够以面向对象的思维方式来操作数据库。

6. Ibatis

相对于 Hibernate 而言,Ibatis 是一个“半自动化”的 ORM 实现框架,它主要致力于 POJO 与 SQL 之间的映射关系,是对“全自动化”ORM 框架的一种有益补充。

7. EasyJWeb

EasyJWeb 是一个核心基于模板技术实现的 MVC 框架,主要致力于 Java Web 应用程序的快速开发。

除了上面介绍的这些框架,Java 中还有很多框架,在应用开发中,可根据实际的需求来选择使用。

1.2.3 XML、CSS 的应用

XML 中文称为可扩展标记语言,它是由 W3C (万维网协会) 推出的新一代数据交互的标准,主要用于定义 Web 网页上的文档元素和商业文档。

XML 的前身是标准通用标记语言 (Standard Generalized Markup Language, SGML)。XML 最早于 1996 年出现,并由相关人士向 W3C 提案。直到 1998 年 2 月,W3C 正式推出了 XML (XML 1.0)。在 Internet 背景下,XML 以其应用简单、使用灵活等优势在 Web 应用中占据着重要地位,并得到了迅速的发展。如今,XML 不仅已经广泛用于计算机及计算机网络的各个方面,还在机械、物理、化学、数学等领域发挥着越来越重要的作用。

XML 是一种数据存储语言,它使用一系列简单的标记 (或者称为元素) 来描述数据,包

含 XML 的文件称为 XML 文件，它通常以“.xml”结尾。下面是一个名为“student.xml”文件中的代码：

```
<?xml version="1.0" encoding="GB2312"?>
<student>
  <!-- 姓名 -->
  <name>张三</name>
  <!-- 班级 -->
  <class>09213 班</class>
</student>
```

在上述代码中，“<?xml version="1.0" ?>”是 XML 文件中的声明，属性 version 指定 XML 文件的版本号，属性 encoding 指定 XML 文件的字符编码格式；<student>是 XML 文件中的根元素，一个 XML 文件中有且只有一个根元素，其他元素都包含在根元素中（注释可以例外）；“<!-- 姓名 -->”是注释部分。XML 文件中每个元素都有开始和结束部分。

XML 文件通常都是配合其他应用程序而使用的，它也可以单独运行，如使用浏览器打开。图 1-1 所示为一个 XML 文件在 IE 浏览器中的运行结果。

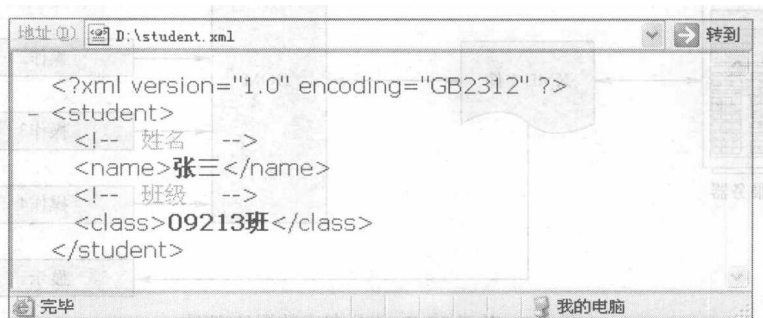


图 1-1 XML 文件在 IE 浏览器中的运行结果

XML 的用途是非常广泛的。包括以下几个方面。

1. 实现不同应用程序之间的数据交互

XML 是跨平台的，它提供了一种不同的应用程序之间进行数据库交换的公共标准，是一种公共的交互平台，如图 1-2 所示。

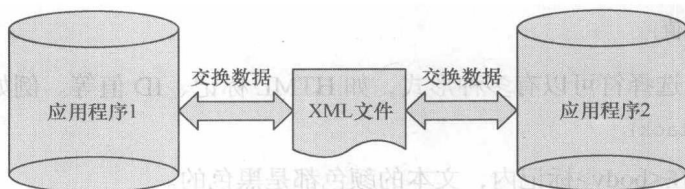


图 1-2 不同的应用程序之间的数据交换

2. 对一种数据实现多种样式

XML 将信息的数据部分和信息的显示样式部分分离开来，这样就可以给同一份数据添加多种样式（样式部分必须由其他语言来定义，如使用 CSS），从而得到多种显示效果，如图 1-3 所示。

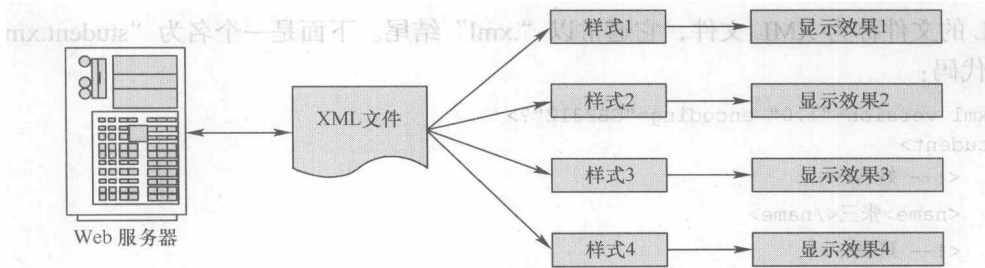


图 1-3 XML 数据样式示意图

3. 实现数据的分布式处理

XML 是一种针对 Internet 设计的标记语言，它可以在 Internet 上自由传送。当 XML 数据被发送给客户后，客户可以通过应用软件从 XML 文档中提取数据，进而对数据进行编辑和处理。XML 文档对象模型允许用脚本和其他编程语言处理 XML 格式的数据。这种情况下的数据处理可以在客户端完成，而没有必要返回到 Web 服务器上，因此节省了 Internet 上的数据带宽。基于 XML 的分布式数据处理图如图 1-4 所示。

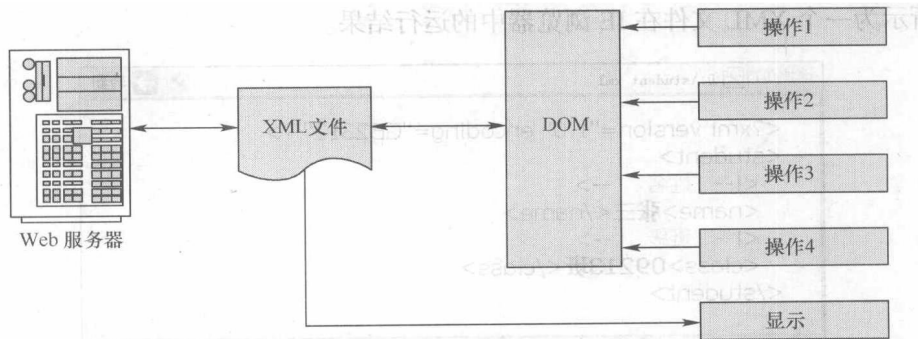


图 1-4 基于 XML 的分布式数据处理图

CSS 英文全称为 Cascading Style Sheets，中文称为层叠样式表单，由 W3C 的 CSS 工作组设置和维护，它是一种用来表现 HTML 或 XML 等文件式样的计算机语言，属于浏览器解释型语言，不需要编译，就可以直接由浏览器执行。

CSS 语法非常简单，组成 CSS 语法的元素只有 CSS 选择符与 CSS 属性。每个 CSS 选择符可以包含一个或多个 CSS 属性。基本格式如下：

选择符 {属性:属性值}

在 HTML 中，选择符可以有多种形式，如 HTML 标记、ID 值等。例如：

```
body {color:black}
```

上述代码说明在<body>标记内，文本的颜色都是黑色的。

在 XML 中，选择符可以是标记名称或标记名称的 id/class 属性值。例如，将下面的 XML 代码通过 id 和 class 属性来设置样式：

```
<?xml version="1.0" encoding="GB2312"?>
<student>
  <name id="className">张三</name>
  <class class="studentClass">09213 班</class>
</student>
```



```
</student>
```

CSS 文件内容:

```
#studentName {color:blue}
.studentClass {color:red}
```

id 和 class 的区别在于, id 属性的值在文件中是唯一的, 可以唯一地标识一个标记, 而 class 属性的值不唯一, 可以标识一组标记。在 CSS 文件中, 定义 id 的样式用井号 (“#”) 加 id 值来定义, 定义 class 属性的样式用点号 (“.”) 加 class 值来定义。如果对父标记做了样式的定义, 而没有对其子标记做定义, 则子标记具有和父标记相同的样式。

如果一个选择符含有多个属性, 则属性之间用分号 (“;”) 隔开。如果有多个选择符具有相同的属性和属性值, 则可以把这些选择符组合起来书写, 用逗号将选择符分开。如:

```
a,b,c{color:red}
```

这样书写的好处是, 减少了相同属性列表出现的次数, 使 CSS 结构变得更加简洁。在 CSS 语法中是不区分大小写的, 但推荐使用小写。

CSS 样式文件通常以 “.css” 结尾, 如 “student.css”。在 XML 文件中引用 CSS 文件的方式为:

```
<?xml-stylesheet type="text/css" href="CSS 文件的 URI" ?>
```

在上述代码中, href 属性的值是一个 CSS 文件的 URI, URI 必须是一个有效的资源。例如, 在下面的 XML 文件中引用 CSS 样式文件, 示例代码如下:

```
<?xml version="1.0" encoding="GB2312"?>
<?xml-stylesheet type="text/css" href="school.css" ?>
<student>
  <!-- 省略代码 -->
</student>
```

说明: CSS 文件不仅可以被 XML 文件引用, 还可以直接将 CSS 代码写入 XML 文件中, 例如:

```
<?xml version="1.0" encoding="GB2312"?>
<?xml-stylesheet type="text/css" ?>
<student xmlns:HTML="http://www.w3.org/Profiles/XHTML-transitional">
  <HTML:style>
    name {color:blue}
    class {color:red}
  </HTML:style>
  <!-- 姓名 -->
  <name>张三</name>
  <!-- 班级 -->
  <class>09213 班</class>
</student>
```

将 CSS 代码直接写入 XML 文件中会降低 XML 文件的可读性以及可维护性, 因此更建议使用 CSS 样式文件的方式。

1.2.4 使用 JavaScript 与 Ajax 提升用户体验

JavaScript 是由 Netscape 公司开发的一种客户端脚本语言, 它允许在 HTML 基础上进行交互式 Web 网页的开发。JavaScript 的出现使网页与用户之间实现了一种实时性的、动态的、