

JSP

数据库通用模块开发与系统移植

张 蓓 赵国锋 编著

本书主要内容：

JSP数据库开发相关技术

注册与登录模块

文件上传和下载模块

邮件收发系统模块

网上聊天室模块

留言簿模块

数据分页显示模块

BBS论坛模块

构建网上社区和网上书店系统

数据库系统的移植与升级



清华大学出版社

数据库通用模块开发与系统移植丛书



TP393.092

653D

2007

JSP

数据库通用模块开发 与系统移植

张 蓓 赵国锋 编 著

清华大学出版社

北 京

内 容 简 介

JSP(Java Server Pages)是基于 Java 的技术,用于创建可支持跨平台及 Web 服务器的动态网页。JSP 继承了 Java 语言的优势,是一种与平台无关的开发技术,而 Java 技术的强大功能也赋予了 JSP 能够为用户提供强大功能的技术支持。

本书从实例入手,介绍了数据库通用模块、综合实例和数据库的移植等方面的内容。全书共 12 章,可以分成 4 个部分:第 1 章和第 2 章是第一部分,讲述了 JSP 和数据库编程的基本理论知识;第 3~9 章是第二部分,介绍了数据库通用模块的开发方法;第 10 章和第 11 章是第三部分,通过两个综合实例讲述了数据库系统开发的方法和技巧;第 12 章是第四部分,讲述了数据库系统的移植和升级方法。

本书既可以作为 JSP 初学者的入门教材,也可以作为 JSP 开发人员的参考资料,而且对高等院校计算机专业的学生进行数据库设计和毕业设计有较高的参考价值。

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 13501256678 13801310933

图书在版编目(CIP)数据

JSP 数据库通用模块开发与系统移植/张蓓,赵国锋 编著. —北京:清华大学出版社,2007.5

(数据库通用模块开发与系统移植丛书)

ISBN 978-7-302-15143-2

I. J… II. ①张… ②赵… III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2007)第 060786 号

责任编辑:王 定 鲍 芳

封面设计:久久度文化

版式设计:康 博

责任校对:胡雁翎

责任印制:王秀菊

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

c-service@tup.tsinghua.edu.cn

社 总 机:010-62770175 邮购热线:010-62786544

投稿咨询:010-62772015 客户服务:010-62776969

印 刷 者:北京鑫丰华彩印有限公司

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:195×260 印 张:30.75 字 数:768 千字

附光盘 1 张

版 次:2007 年 5 月第 1 版 印 次:2007 年 5 月第 1 次印刷

印 数:1~5000

定 价:48.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:022189-01

前 言

当今社会是一个信息化的社会,PC 技术和 Internet 技术的高速发展给人们的生活带来了极大的方便。而各种动态编程语言的出现,也为电子商务系统的开发不断地带来新的技术选择。从最早的 CGI、SSI 等技术到后来的 ASP、PHP 等,都为各类电子商务系统的开发注入了活力。但是这些技术都没能同时兼顾到开发效率和运行效率,不能同时满足分布式事件处理的功能,也不具有强大的扩展能力。

JSP 技术的出现给编程人员提供了一种全新的技术选择。JSP 继承了 Java 语言的优势,是一种与平台无关的开发技术,而 Java 技术也赋予了 JSP 为用户提供强大功能的技术支持。目前,JSP 技术作为后起之秀,已经越来越受到编程者的关注和喜爱。

在 JSP 的应用中,以数据库技术的应用最为广泛,用 JSP 开发的各类电子商务系统都离不开数据库技术的支持。目前市场上与 JSP 技术相关的书籍资料非常多,但是却很少有专门介绍各类数据库通用模块、数据库移植和升级的书籍。

本书从最基本的数据库知识出发,介绍了目前比较流行的数据库通用模块,这些模块是所有编程人员进行数据库开发时都会经常用到的。同时以综合实例讲述了将这些功能模块整合到数据库系统中的方法,使读者能够很好地掌握本书中通用模块的使用。通过数据库系统移植和升级的学习,了解在数据库系统中调用各模块的方法。

本书共 12 章,可以分成 4 个部分。

第一部分包括第 1 章和第 2 章,介绍了 JSP 和数据库相关的基础知识,包括 JSP 技术概述、JSP 的开发和应用平台、JSP 语法知识、JSP 的内置对象、JavaBean 在 JSP 中的应用、数据库的后台产品、SQL 语言基础、JDBC 数据库开发、JDBC 接口等知识,这些都是进行 JSP 数据库开发的基础,同时还介绍了本书程序的开发和运行环境。

第二部分包括第 3~9 章,介绍了常用的数据库通用模块,它们分别是注册和登录模块、文件上传和下载模块、邮件收发系统模块、网上聊天室模块、留言簿模块、数据分页显示模块、BBS 论坛模块等。

第三部分包括第 10 章和第 11 章,主要介绍了网上社区和网上书店两个数据库综合系统,

网上社区完全是由第二部分中的通用模块所构成,而网上书店则是一个具有完整功能的数据库系统,它也调用到了第二部分中的一些通用模块。

第四部分是第 12 章,主要介绍数据库系统的移植和升级,包括各种 JSP 服务器平台之间的移植、各种数据库之间的移植和升级、系统功能升级和系统技术升级等内容。

本书主要由张蓓、赵国锋执笔,此外,参与本书编写的还有赵伟仁、石葵香、张洪伟、白文磊、徐路迎、邓伟杰、霍玉嵩、邹维涛、易诚斌、孙琼、张圣亮等人。

由于时间仓促,加之水平有限,书中不足之处在所难免,敬请读者批评指正。

编 者

2007 年 3 月

CONTENTS

目 录

第 1 章 JSP 编程基础	1
1.1 JSP 概述	1
1.2 JSP 的开发和应用平台	3
1.2.1 Caucho 公司的 Resin 平台	3
1.2.2 Apache 公司的 Tomcat 平台	4
1.2.3 BEA 公司的 WebLogic 平台	4
1.2.4 IBM WebSphere Application Server 平台	4
1.3 本书程序的开发和运行环境	5
1.3.1 安装和配置 J2SDK	5
1.3.2 安装和配置 Tomcat	7
1.3.3 本书使用的开发工具	8
1.3.4 本书的项目发布	10
1.4 JSP 语法知识	11
1.4.1 JSP 脚本元素	11
1.4.2 JSP 指令	14
1.4.3 JSP 动作	18
1.5 JSP 内置对象	24
1.6 JavaBean 在 JSP 中的应用	27
1.6.1 JavaBean 简介	27
1.6.2 JavaBean 的生命周期	28
1.6.3 JavaBean 开发实例	29
1.7 本章小结	34

第 2 章 JSP 数据库编程基础	35
2.1 JDBC 数据库开发	35
2.1.1 JDBC 概述	35
2.1.2 JDBC 应用模型	36
2.1.3 JDBC driver	37
2.1.4 JDBC 连接数据库的方法	39
2.1.5 JDBC 访问数据库	41
2.2 JDBC 接口类和对象	43
2.2.1 DriverManager 类	43
2.2.2 Connection 对象	44
2.2.3 Statement 对象	47
2.2.4 ResultSet 对象	49
2.2.5 PreparedStatement 接口	51
2.2.6 CallableStatement 对象	53
2.3 ODBC 数据源	56
2.4 JDBC 连接各种数据库	58
2.4.1 连接 Oracle 数据库	58
2.4.2 连接 DB2 数据库	58
2.4.3 连接 SQL Server 数据库	59
2.4.4 连接 Sybase 数据库	59
2.4.5 连接 MySQL 数据库	59
2.4.6 连接 Access 数据库	60
2.5 本章小结	60

第 3 章 注册与登录模块	61	第 5 章 邮件收发系统模块	119
3.1 数据库设计	61	5.1 JavaMail API 概述	119
3.2 最基本的注册模块	62	5.1.1 相关协议介绍	119
3.2.1 页面效果	62	5.1.2 Java Mail API 核心类	120
3.2.2 代码实现	63	5.1.3 安装 JavaMail API	124
3.3 带验证码的注册模块	70	5.2 邮件发送模块	125
3.3.1 页面效果	70	5.2.1 发送文本格式的邮件	125
3.3.2 代码实现	71	5.2.2 发送 HTML 格式的邮件	129
3.4 最基本的登录模块	75	5.2.3 发送带附件的邮件	131
3.4.1 页面效果	75	5.3 邮件接收模块	136
3.4.2 代码实现	77	5.3.1 查看邮件列表	136
3.5 修改用户资料模块	81	5.3.2 查看邮件详细信息	139
3.5.1 页面效果	81	5.4 带邮件发送的注册模块	142
3.5.2 代码实现	82	5.4.1 用户注册页面	142
3.6 本章小结	92	5.4.2 注册信息处理页面	143
第 4 章 文件上传和下载模块	93	5.5 本章小结	146
4.1 用 jspSmartUpload 组件进行上传 和下载	93	第 6 章 网上聊天室模块	147
4.1.1 相关类介绍	93	6.1 数据库介绍	147
4.1.2 用 jspSmartUpload 组件实现 文件上传	99	6.1.1 users 数据表	147
4.1.3 用 jspSmartUpload 组件实现 文件下载	103	6.1.2 chat 数据表	148
4.1.4 用 jspSmartUpload 组件进行较 复杂的表单处理	104	6.1.3 userlist 数据表	149
4.2 用 JavaBean 实现文件上传	108	6.2 文件结构	149
4.2.1 FileUpload.java	108	6.3 系统公用类	151
4.2.2 file_upload_javabean_form.jsp	113	6.3.1 数据库连接类	151
4.2.3 file_upload_javabean_do.jsp	114	6.3.2 数据库操作类	152
4.3 采用文件流输出的方式实现文件 下载	116	6.4 普通用户模块	155
4.3.1 file_download_form.jsp	116	6.4.1 用户登录页面	155
4.3.2 file_download_do.jsp	116	6.4.2 登录页面后台处理页面	155
4.4 本章小结	118	6.4.3 聊天室主页面	161
		6.4.4 更改房间页面	163
		6.4.5 发送聊天信息页面	165
		6.4.6 聊天信息显示页面	171
		6.4.7 在线用户显示列表页面	174
		6.4.8 用户退出页面	177
		6.5 管理员模块	179
		6.5.1 管理员登录页面	179

6.5.2 聊天室管理页面	180	第 9 章 BBS 论坛模块	241
6.5.3 “踢人”功能页面	183	9.1 数据库介绍	241
6.6 本章小结	185	9.1.1 articles 数据表	241
第 7 章 留言簿模块	187	9.1.2 boards 数据表	242
7.1 数据库介绍	187	9.1.3 denylist 数据表	243
7.1.1 users 数据表	187	9.1.4 email 数据表	244
7.1.2 words 数据表	188	9.1.5 users 数据表	244
7.2 留言簿模块结构	188	9.2 系统概述	245
7.2.1 功能介绍	189	9.2.1 论坛系统的模块构成	245
7.2.2 文件结构	189	9.2.2 功能模块的调用	246
7.3 系统公用类	190	9.2.3 系统文件结构	247
7.3.1 数据库访问类	190	9.3 系统公用类	249
7.3.2 users 数据表操作类	193	9.3.1 数据库连接类	250
7.3.3 words 数据表操作类	198	9.3.2 数据库操作类	250
7.4 留言簿模块页面实现	203	9.3.3 articles 数据表操作类	250
7.4.1 非登录用户主页面	203	9.3.4 boards 数据表操作类	257
7.4.2 非登录用户功能选择页面	204	9.3.5 denylist 数据表操作类	261
7.4.3 用户注册模块	205	9.3.6 email 数据表操作类	264
7.4.4 用户登录模块	207	9.3.7 users 数据表操作类	269
7.4.5 查看留言信息列表页面	209	9.3.8 分页显示类	275
7.4.6 查看留言信息内容页面	212	9.3.9 在线人数计算类	276
7.4.7 登录用户主页面	214	9.4 非登录用户模块	277
7.4.8 登录用户功能选择页面	216	9.4.1 非登录用户主页面	277
7.4.9 发表留言页面	216	9.4.2 非登录用户功能选择页面	279
7.4.10 留言信息保存页面	218	9.4.3 信息显示页面	279
7.4.11 用户退出页面	220	9.4.4 用户注册模块	283
7.5 本章小结	220	9.4.5 用户登录模块	283
第 8 章 数据分页显示模块	221	9.4.6 讨论区列表页面	284
8.1 数据库介绍	221	9.4.7 讨论区普通文章列表页面	286
8.2 数据分页显示技术	222	9.4.8 查看普通文章详细内容页面	290
8.2.1 最基本的分页显示技术	222	9.4.9 精华文章列表页面	295
8.2.2 分页显示类	227	9.4.10 文章查询页面	296
8.2.3 改进的分页显示类	235	9.4.11 文章查询处理页面	297
8.3 本章小结	240	9.4.12 讨论区查询处理页面	300
		9.5 登录用户模块	303
		9.5.1 登录用户主页面	303

9.5.2	登录用户功能选择页面	304
9.5.3	发表文章页面	304
9.5.4	文章信息处理页面	309
9.5.5	个人邮箱管理页面	312
9.5.6	收件箱邮件列表页面	313
9.5.7	查看邮件信息页面	315
9.5.8	填写邮件信息页面	318
9.5.9	邮件发送处理页面	322
9.5.10	发件箱邮件列表页面	323
9.5.11	删除邮件信息页面	323
9.5.12	修改用户资料模块	325
9.5.13	删除文章页面	327
9.5.14	添加精华文章页面	330
9.5.15	禁止用户发言页面	332
9.5.16	禁止用户发言信息处理页面	332
9.6	管理员模块	335
9.6.1	管理员主页面	335
9.6.2	管理员功能选择页面	335
9.6.3	用户管理主页面	336
9.6.4	删除用户信息页面	337
9.6.5	讨论区管理主页面	337
9.6.6	添加讨论区信息页面	338
9.6.7	添加讨论区信息处理页面	339
9.6.8	修改讨论区信息页面	341
9.6.9	修改讨论区信息处理页面	342
9.6.10	删除讨论区信息页面	345
9.7	本章小结	345
第 10 章	构建网上社区	347
10.1	网上社区概述	347
10.1.1	网上社区的功能介绍	347
10.1.2	文件结构	348
10.1.3	数据库介绍	349
10.1.4	系统公用类	350
10.2	系统公共模块	351
10.2.1	用户注册模块	351
10.2.2	用户登录模块	352

10.2.3	用户退出页面	357
10.3	嵌入 BBS 论坛模块	358
10.3.1	修改类调用位置	358
10.3.2	修改对应用户退出 页面的链接	359
10.3.3	修改网上社区的配置文件	359
10.4	嵌入网上聊天室模块	359
10.4.1	修改类调用位置	360
10.4.2	修改对应用户退出页面 的链接	360
10.5	嵌入留言簿模块	360
10.5.1	修改类调用位置	360
10.5.2	修改对应用户退出页面 的链接	360
10.5.3	修改调用的数据库访问类	360
10.6	系统演示说明	361
10.7	本章小结	361
第 11 章	网上书店	363
11.1	数据库介绍	363
11.1.1	users 数据表	363
11.1.2	books 数据表	364
11.1.3	sorts 数据表	365
11.1.4	words 数据表	366
11.1.5	goods 数据表	367
11.2	系统概述	367
11.2.1	网上书店的模块构成	368
11.2.2	功能模块的调用	368
11.2.3	系统文件结构	369
11.3	系统公用类	371
11.3.1	数据库连接类	371
11.3.2	数据库操作类	373
11.3.3	books 数据表操作类	373
11.3.4	goods 数据表操作类	378
11.3.5	sorts 数据表操作类	381
11.3.6	users 数据表操作类	384
11.3.7	words 数据表操作类	390

11.3.8 分页显示类.....	392	11.6.16 删除订单信息页面.....	436
11.4 系统公共模块.....	392	11.7 本章小结	437
11.4.1 用户注册模块.....	392	第 12 章 系统移植与升级	439
11.4.2 用户登录模块.....	393	12.1 系统移植	439
11.4.3 获取用户密码模块	394	12.1.1 移植到 Resin 服务器平台	439
11.4.4 用户退出页面.....	398	12.1.2 移植到 JBoss 服务器平台	441
11.5 普通用户模块.....	398	12.1.3 移植到 WebLogic 服务器平台	443
11.5.1 普通用户主页面	398	12.1.4 移植到 Tomcat 低版本平台	448
11.5.2 普通用户功能选择页面.....	400	12.2 数据库移植	448
11.5.3 浏览所有图书列表页面.....	401	12.2.1 MySQL 数据库移植	448
11.5.4 查看图书详细信息页面.....	403	12.2.2 SQL Server 数据库移植	450
11.5.5 推荐图书处理页面	408	12.2.3 从 SQL Server 移植到 MySQL	453
11.5.6 留言信息处理页面	409	12.2.4 从 SQL Server 移植到 Oracle	455
11.5.7 添加购物车页面	410	12.2.5 从 MySQL 移植到 Oracle 的注意事项	457
11.5.8 购物车信息处理页面	411	12.3 系统功能升级	458
11.5.9 搜索图书信息页面	412	12.3.1 增加站内邮件管理模块	458
11.5.10 搜索图书信息处理页面.....	412	12.3.2 删除热门图书模块.....	460
11.5.11 热门图书列表页面	414	12.3.3 转换成网上商城.....	461
11.5.12 购物车信息列表页面	415	12.4 系统技术升级	467
11.5.13 删除购物车信息页面	417	12.4.1 Servlet 技术.....	467
11.6 管理员模块.....	418	12.4.2 数据库连接池技术.....	472
11.6.1 管理员主页面	418	12.4.3 数据库存储过程.....	477
11.6.2 管理员功能选择页面	419	12.5 本章小结	481
11.6.3 图书管理主页面	420		
11.6.4 添加图书信息页面	421		
11.6.5 添加图书处理页面	421		
11.6.6 修改图书信息页面	424		
11.6.7 修改图书处理页面	425		
11.6.8 删除图书信息页面	427		
11.6.9 图书类别管理主页面	429		
11.6.10 添加图书类别页面	431		
11.6.11 添加图书类别处理页面.....	432		
11.6.12 修改图书类别页面	434		
11.6.13 修改图书类别处理页面.....	434		
11.6.14 删除图书类别信息页面.....	435		
11.6.15 订单管理主页面	436		

JSP编程基础

Java Server Page(JSP)是基于 Java 的技术,用于创建可支持跨平台及 Web 服务器的动态网页。JSP 页面代码一般由普通的 HTML 语句和特殊的嵌入标记组成,可以使用任何编辑工具书写代码,然后将动态部分用特殊标记嵌入即可。

本章首先介绍了 JSP 基本情况,然后介绍了当前 JSP 的开发和应用平台,并对本书程序的开发和运行环境做了介绍,最后介绍了 JSP 的基础知识,这些都是进行 JSP 编程所必须掌握的内容。



1.1 JSP概述

JSP 是由 Sun Microsystems 公司于 1999 年 6 月推出的新技术,由 Sun Microsystems 公司倡导、许多公司参与建立的一种动态网页技术标准,是基于 Java Servlet 以及整个 Java 体系的 Web 开发技术,是 Servlet 2.1 API 的扩展。

JSP 的设计目标主要是提供一种更为简便、有效的动态网页编写手段,并且在增强网页程序的独立性、兼容性和可重用性方面也具有许多优点。

1. Write Once, Run Anywhere

JSP 技术继承了 Java 的先进概念——Write Once, Run Anywhere,这种概念正影响着互联网行业的交互式 Web 页面的设计理念。JSP 页面可以非常容易地跨平台、跨 Web 服务器软件来设计和开发源代码。

2. 支持多种动态网页格式

对于使用 JSP 技术设计出来的网页应该采取什么样的格式,目前还没有一个明确的标准。一般来说,JSP 技术目前既可支持 HTML/DHTML 的传统浏览器文件格式,又可以支持应用于无线通信(例如移动电话、PDA 等设备)进行网页浏览的 WML 文件格式,还可以支持其他一些

B2B 应用的 XML 格式。

3. JSP 标签可扩充性

JSP 是使用标签和脚本技术来制作动态 Web 网页的。JSP 技术能够使开发者扩展 JSP 标签、定制标签库,所以制作网页时充分利用与 XML 兼容的标签技术可以大大减少对脚本语言的依赖。定制标签技术使用户降低了制作网页和向多个网页扩充关键功能的复杂程度。

4. JSP 跨平台的可重用性

JSP 开发人员在开发过程中一直关注可重用性。JSP 组件(如 EJB、Java Bean 或定制的 JSP 标签)都是跨平台可重用的。EJB 组件可以访问传统的数据库,并能以分布式系统模式工作于 UNIX 和 Windows 平台。JSP 标签的可扩充功能为开发人员提供简便的、与 XML 兼容的接口(即共享网页打包功能),使其完全地实现工业标准化。

JSP 这种基于组件的模式能有效提高应用程序的开发效率,能够使开发人员快速创建模板应用程序,然后再整合一些附加功能后便可使用。

5. 易维护性

基于 JSP 技术的应用程序比基于 ASP 技术的应用程序易于维护和管理。脚本语言都能很好地服务于小的应用程序,但不能适应大型的、复杂的应用程序。因为 Java 是结构化的,因此比较容易创建和维护庞大的、组件化的应用程序。

JSP 突出的组件技术使得修改内容和修改逻辑互不影响,而且也非常容易实现。EJB 是为企业级应用而设计的 Java 组件,它整合了企业逻辑,例如数据库的访问、安全、事务完整性及独立性(即独立于应用程序)。

因为 JSP 技术是一种开放的、跨平台的结构,因此,Web 服务器、平台及其他组件能很容易地升级或切换,并且不会影响 JSP 基本的应用程序。这一特点使得 JSP 能够适用现实世界的各种 Web 应用程序不断地变化和发展。

6. 企业产品的多样性

Java 2 平台即企业版 J2EE 是适用于多企业应用程序的 Java 结构。JSP 作为 J2EE 的部分,JSP 网页可访问所有 J2EE 的组件,包括 Java Bean, EJB 及 Java Servlets。由于 JSP 网页都能被编译成 Servlet,因此,JSP 网页能够较完整地继承有关 Servlet 的许多优点,为客户提供基于服务器端的 Java 应用服务。J2EE 平台内容不仅包含管理复杂的企业应用程序,而且还包括事务管理技术和 Pooling 资源管理技术。

JSP 网页可以访问标准的 J2EE 服务,包括:

- Java 名称和目录页面 API。
- JDBC API(与关联数据库通信)。
- Java Mail(支持基于 Java 邮件和消息应用程序的类)。
- Java 消息服务。

通过 J2EE, JSP 网页能够用许多方式同企业系统交互访问。J2EE 支持两种 CORBA 规范

的技术, Java IDL 和 RMI-IIOP。在 EJB 技术支持下, JSP 网页通过运用高级的、对象映射的方式访问数据库。

因为 JSP 技术是基于 Java 开放性过程的产品, 因此它能够广泛支持不同提供商提供的工具、Web 服务器和应用程序的服务, 这样能够使用户选择最佳的开发方法, 选择最适合他们的应用程序开发的工具包。同时, 可以有效地保护用户在代码和人员培训上的投资。



1.2 JSP的开发和应用平台

JSP 基于 Java Servlet 技术, 是 Servlet 2.1 API 的扩展, 因此, 支持 Servlet 的新版本平台都支持 JSP。这样的平台现在越来越多, 要学习 JSP 和 Servlet 开发, 首先必须准备一个符合 Servlet 2.1/2.2 和 JSP 1.1 或更高规范的开发环境。

除了开发工具之外, 还要安装一个支持 Java Servlet 的 Web 服务器, 或者在现有的 Web 服务器上安装 Servlet 软件包。目前许多 Web 服务器都自带了一些必要的软件。

到现在为止, Apache Geronimo, BEA, CAS, IBM, JBoss, NEC 等厂家的产品都支持 JSP 技术和 Java Servlet。

1.2.1 Caucho 公司的 Resin 平台

Resin 平台是由 Caucho 公司发布的 JSP 平台, 根据 <http://www.caucho.com/> 提供的测试结果, Resin 平台是迄今为止最快的商业 JSP 平台。

Resin 提供了最快的 Jsp/Servlets 运行平台。在 Java 和 JavaScript 的支持下, Resin 可以为任务灵活选用合适的开发语言。Resin 的一种先进的语言 XSL(XML stylesheet language)可以使得形式和内容相分离。

如果选用 JSP 平台作为 Internet 商业站点的支持, 那么速度、价格和稳定性都是要考虑到, Resin 十分出色, 表现更成熟, 很具备商业软件的要求, 从站点下载的就是完整版本。

Resin 的特性包括以下方面:

- 支持 JSP 和在服务器端编译的 JavaScript。
- 比 mod_perl, mod_php 更快, 比 Tomcat 快 3 倍。
- 自动的 Servlet/Bean 编译。
- 支持 Servlet, XSL Filtering。
- 支持 IIS, Apache, Netscape 和其他内置了 HTTP/1.1 的 Web 服务器。
- XSLT 和 XPath1.0 引擎。
- 企业级的共享软件(基于一个开放源码的协议)。

通过 <http://www.caucho.com/> 可以访问 Resin 平台的首页。



1.2.2 Apache 公司的 Tomcat 平台

Tomcat 是 Apache Jakarta 软件组织的一个子项目, Tomcat 是一个 JSP/Servlet 容器, 它是在 SUN 公司的 JSWDK(Java Server Web Development Kit)基础上发展起来的, 也是一个 JSP 和 Servlet 规范的标准实现, 使用 Tomcat 可以体验 JSP 和 Servlet 的最新规范。经过多年的发展, Tomcat 具备了很多商业 Java Servlet 容器的特性, 并被一些企业用于商业用途。

Tomcat 是 Servlet 2.2 和 JSP 1.1 规范的官方参考实现。Tomcat 既可以单独作为小型 Servlet、JSP 测试服务器, 也可以继承 Apache Web 服务器。直到 2000 年, Tomcat 还是唯一支持 Servlet 2.2 和 JSP 1.1 规范的服务器, 但现在已经有许多其他服务器宣布对这方面的支持。

Tomcat 和 Apache 一样是免费的。但是, Tomcat 服务器的安装和配置有点麻烦, 和其他商业级的 Servlet 引擎相比, 配置 Tomcat 的工作量显然要多一些。目前最新的 Tomcat 版本是 5.5.16, 它支持 Servlet 2.4 和 JSP 2.0。在 <http://tomcat.apache.org/>上列出了所有 Tomcat 版本的下载和其他信息。

1.2.3 BEA 公司的 WebLogic 平台

BEA 公司的 WebLogic 平台是一套基于 Java 的功能强大的电子商务套件, 它提供了许多功能强大的中间件, 能方便编程人员编写的 JSP、Servlet 等电子商务应用, 可以为企业提供完整的商务应用解决方案, 是为超大型电子商务应用系统而设计的。它采用 CORBA(公共对象)的系统结构, 提供基于分布式的 JSP 应用系统。CORBA 的核心是 ORB(对象请求中介), ORB 的作用就像一个中间人, 使各个对象能够互递请求。尽管 ORB 是在 Client/Server 环境中工作, 但是与 ORB 一起工作的对象, 既可以是客户, 又可以是服务器, 取决于具体情况。将 ORB, IDL 和接口存储库连接起来, 就是一个基本的 CORBA 模型。由于 BEA 公司的 WebLogic 平台是针对超大型电子商务应用系统而设计的, 有兴趣的读者可以访问 <http://www.bea.com/>。

对于开发人员, 可以从 <http://www.bea.com/>免费下载一套完整的 WebLogic, 并得到一个限制了 IP 的 license, 用于学习和开发这个套件的代码。如果需要正式投入使用的话, 则必须支付一定的费用获得无限制的 license。由于这个套件基于这种发布方式, 一般网站开发人员可以轻易地得到 WebLogic 用于学习开发。

1.2.4 IBM WebSphere Application Server 平台

WebSphere Application Server 是基于 Java 的应用环境, 用于建立、部署和管理 Internet 和 Intranet Web 应用程序。这一整套产品进行了扩展, 以适应 Web 应用服务器的需要, 范围从简单到高级直到企业级。

IBM WebSphere Application Server 是一种功能完善、开放的 Web 应用程序服务器, 是 IBM 电子商务计划的核心部分。

IBM WebSphere Application Server 具有以下特性:

- 基于 Java 和 Servlet 的 Web 应用程序运行环境, 包含了为 Web 站点提供服务所需要的一切, 如项目管理、连接数据库、Java Servlet 代码生成器、Bean 和 Servlet 开发工具、HTML 编译器、网站发布等, 为开发 Servlet 和 Java Bean 提供了多种向导。WebSphere Performance Pack 作为网络优化管理工具, 可以减少网络服务器的拥挤现象, 扩大容量, 提高 Web 服务器性能。
- 运行时可以协同并扩展 Apache, Netscape, IIS 和 IBM 的 HTTP Web 服务器, 因此可以成为强大的 Web 应用服务器。
- 包含了 eNetworkDispatcher, WebTrafficExpress 代理服务器和 AFS 分布式文件系统, 可以提供伸缩的 Web 服务器环境。其基本工作过程是: 客户发出请求后, 由 Http Server 将 Servlet 调用请求交给 Application Server, 由 Application Server 和 Java Servlet Engine 执行用户调用的 Servlet 进行数据库连接, 将 SQL 请求发送给数据库进行处理; 数据库将结果返回 Application Server; Servlet 生成动态页面后, 将处理结果交给 Http Server, Http Server 将页面返回给用户。由于 Websphere 面向专业人员, 所以要完全掌握的话有一定的难度。

在 <http://www-306.ibm.com/software/webservers/appserv/was/> 上有关于 WebSphere Application Server 的相关信息。



1.3 本书程序的开发和运行环境

前面介绍了 JSP 的多个开发和应用平台, 它们有各自的优点, 也适合于不同的开发环境。本书程序都是在 Tomcat 平台下运行的, 使用的软件版本分别是 JDK 1.5 和 Tomcat 5.5.16。下面介绍它们的安装和配置, 计算机所使用的操作系统是 Windows XP。

1.3.1 安装和配置 J2SDK

在 SUN 公司的网站上面可以免费下载 J2SDK 安装软件, 下载网址是:

<http://java.sun.com/javaee/downloads/index.html>

注意要下载 J2SDK 的 Windows 版本, 这里下载的是 jdk-1_5_0_04-windows-i586-p.exe。

下载完成后, 直接双击下载软件即可执行安装。按照安装指示进行即可, 将 J2SDK 安装在 C:\jdk1.5.0_04 目录下。

安装完成后, 需要做一些配置工作, 以便 J2SDK 能正常工作。可以按照以下步骤来配置 J2SDK。

(1) 在桌面上右击“我的电脑”图标, 选择“属性”命令, 这时候会出现“系统属性”对话框, 选中对话框中的“高级”选项卡, 单击“环境变量”按钮, 便会出现如图 1-1 所示的对

话框。

(2) 在“系统变量”列表中查看是否有 Path 变量, 如果没有, 就单击“系统变量”选项区的“新建”按钮。这时会弹出如图 1-2 所示的“新建系统变量”对话框。在“变量名”文本框输入 Path, 在“变量值”文本框中添加路径 C:\jdk1.5.0_04\bin;, 完成后单击“确定”按钮。

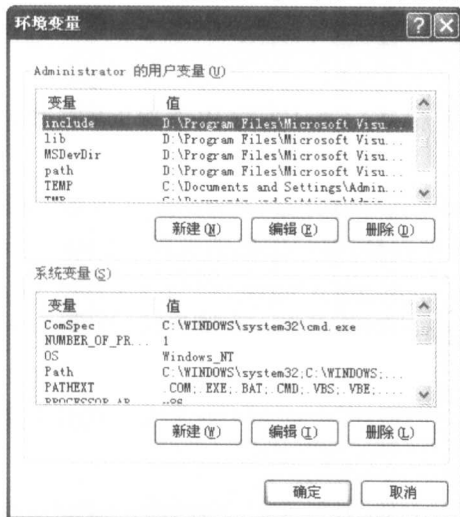


图 1-1 环境变量设置

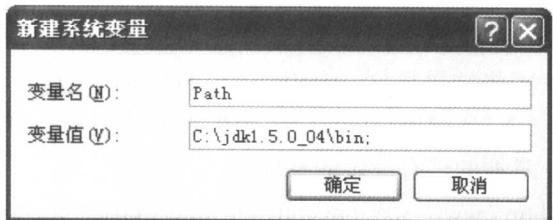


图 1-2 新建 Path 变量

(3) 如果“系统变量”中的 Path 变量已经存在, 只需要选中 Path 变量, 然后单击“编辑”按钮, 便会弹出如图 1-3 所示的“编辑环境变量”对话框。将路径 C:\jdk1.5.0_04\bin; 添加到原有变量值的后面, 然后单击“确定”按钮。

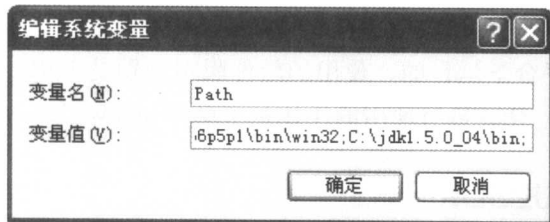


图 1-3 编辑 Path 变量

(4) 单击“环境变量”对话框上的“确定”按钮完成 Path 变量的添加。

注意:

如果是在“环境变量”对话框上的用户变量选项区中添加 Path 变量, 那么该配置只对当前用户有效, 而对其他用户无效。

(5) 按照添加 Path 变量的方法, 在“系统变量”列表中新建一个名为 JAVA_HOME 的变量, 变量值为 C:\jdk1.5.0_04;。

(6) 按照添加 Path 变量的方法, 在“系统变量”列表中新建一个名为 CLASSPATH 的变量, 变量值为.;C:\jdk1.5.0_04\lib\tools.jar;C:\jdk1.5.0_04\lib\dt.jar。

注意:

CLASSPATH 的变量值必须以 “.” 开头。

下面编写一个简单的 Java 程序来测试 J2SDK 是否安装和配置成功。用记事本编写下面代码。

```
public class Test
{
    public static void main(String args[])
    {
        System.out.println("Success!");
    }
}
```

然后按照以下步骤来编译和运行这个程序。

➤ 将程序保存到 E:\ch1 目录下，并命名为 Test.java。

注意:

在这里，文件名应和类名相同，因为 Java 解释器要求公共类必须放在与其同名的文件中。

➤ 打开“命令提示符”，将当前目录转到 E:\ch1 中，然后对程序进行编译。

```
E:\ch1>javac Test.java
```

编译的结果是生成字节码文件 Test.class。

➤ 用 Java 解释器来运行该字节码文件。

```
E:\ch1>java Test
```

如果安装成功，将会打印出 Success!。

1.3.2 安装和配置 Tomcat

在 Apache 公司的官方网站上提供了 Tomcat 软件包的免费下载，网址是 <http://jakarta.apache.org/tomcat/index.html>。

这里下载的是 Windows 版本的 Tomcat 软件包，文件名是 apache-tomcat-5.5.16.exe。

双击 apache-tomcat-5.5.16.exe 文件，便会执行安装程序。将 Tomcat 安装到 C:\Tomcat 5.5 目录中，其他的都按默认选项，最后会提示已经安装成功。

安装完 Tomcat 后，它还不能直接工作，我们还需要对它进行一些配置。按照前面所讲述的添加环境变量的方法，添加下面 3 个变量及它们相应的变量值。

```
CATALINA_HOME, C:\Tomcat 5.5
CATALINA_BASE, C:\Tomcat 5.5
TOMCAT_HOME, C:\Tomcat 5.5
```

