

开发专家

之 Sun ONE

Tomcat与 Java Web

开发技术 详解

书附赠光
为书中范
源文件及
天软件



Sun Weiqin 孙卫琴
Li Hongcheng 李洪成
飞思科技产品研发中心

编著
监制



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>



开发专家之 Sun ONE

Tomcat 与 Java Web 开发技术详解

孙卫琴 李洪成 编著

飞思科技产品研发中心 监制

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书详细介绍了在最新 Tomcat 5 版本上开发 Java Web 应用的各种技术。主要内容包括: Tomcat 和 Java Web 开发的基础知识, Java Web 开发的高级技术, Tomcat 与当前其他通用软件的集成, 以及 Tomcat 的各种高级功能。

书中内容注重理论与实践相结合, 列举了大量具有典型性和实用价值的 Web 应用实例, 并提供了详细的开发和部署步骤。由于 Java Web 技术是 SUN 公司在 Java Servlet 规范中提出的通用技术, 因此本书讲解的 Java Web 应用例子可以运行在任何一个实现 SUN 的 Servlet 规范的 Java Web 服务器上。随书配套光盘内容为书中范例源程序, 以及本书涉及到的软件的安装程序。

本书语言深入浅出, 通俗易懂。无论对于 Java Web 开发的新手还是行家来说, 本书都是精通 Tomcat 技术和开发 Java Web 应用的必备的实用手册。

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

Tomcat 与 Java Web 开发技术详解/孙卫琴, 李洪成编著. —北京: 电子工业出版社, 2004.4

(开发专家之 Sun ONE)

ISBN 7-5053-9392-8

I. T... II. ①孙...②李... III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2003) 第 107424 号

责任编辑: 陆舒敏

印 刷: 北京东光印刷厂

出版发行: 电子工业出版社

北京海淀区万寿路 173 信箱 邮编: 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 28.25 字数: 723.2 千字

印 次: 2004 年 4 月第 1 次印刷

印 数: 5 000 册 定价: 45.00 元 (含光盘 1 张)

凡购买电子工业出版社的图书, 如有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系电话: 010-68279077。质量投诉请发邮件至 zhts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

出版说明

“开发专家”是电子工业出版社计算机图书研发部长期以来精心培育的计算机科学技术类本版品牌。这个品牌是由多个专题系列组成的横向大系列，涵盖了计算机技术的各个方面，特别是一直受到极大关注的程序开发类系列，例如《Borland 开发专家》、《开发专家之数据库》、《开发专家之网络编程》、《开发专家之 Delphi》以及《开发专家之 Sun ONE》等。这些专题系列基于各自的角度，从纵向上包含了该专题的所有内容。因此，整个“开发专家”的品牌架构纵横交错，囊括了所有的计算机技术和所有的技术层面，海纳百川而又极具可扩展性。

“开发专家”的作者队伍主要依托于“飞思科技产品研发中心”。“飞思科技产品研发中心”是由专业的策划人员、权威的技术专家和资深的作者队伍共同构成的。在图书的出版上，形成了以研发为基础、以出版为中心、以服务为支持的专业化出版框架和流程。通过深入的市场调查和技术跟踪，在综合了技术需求和读者焦点等因素的基础上，形成各系列丛书的写作重点和大纲，然后聘请业界的最前沿学者进行写作。同时，策划工作全程介入写作进程，严格控制写作质量，用最专业的技术背景、最深刻的理论基础、最具代表性的案例、最能为专业读者接受的形式，为读者提供品质最佳的图书产品，体现了出版者和著作者的完美结合。

多年来，计算机图书研发部始终把创造社会效益摆在首位，秉承一切为国内计算机技术专业读者服务的精神，为推动国内 IT 技术发展、为体现国内技术的原创水平，穷尽所有的创意与努力，将出版者的命运与读者的支持紧紧地连在了一起。

在此，我们临出版之残酷竞争而不惧，旌旗猎猎而异军突起，这与广大读者的支持是分不开的。为使我们的脚步更坚实、使我们的队伍永葆活力和创造力，我们期待着您能为我们的前进贡献出您的意见和建议。同时，我们也在等待着您的加入。

我们的联系方式如下：

咨询电话：(010) 68134545 68131648

答疑邮件：support@fecit.com.cn

网 址：<http://www.fecit.com.cn> <http://www.fecit.net>

答疑网址：<http://www.fecit.com.cn/> “问题解答”专区

通用网址：计算机图书、FECIT、飞思教育、飞思科技、飞思

电子工业出版社计算机研发部

关于飞思

新世纪之初的北京，一群满怀共同理想的年轻人聚集在飞思教育产品研发中心的旗帜下，他们将新的希望和活力注入了中国 IT 教育产品开发领域。飞思人在为把自己打造成为中国 IT 教育产品研发的精英团队而更加不懈努力。

21 世纪的今天，飞思人在多元化教育产品的开发和出版等方面已经迈出了坚实的第一步，开拓出属于自己的一片天空，初步赢得了涓涓细流。

如今，本着教育为科技服务的宗旨，飞思科技产品研发中心以崭新的面貌等待您的支持与关注。

飞思人理念

我们经常感谢生活的慷慨，让我们这些原本并不同源的人得以同本，为了同一个梦想走到一起。

因为身处科技教育前沿，我们深感任重道远；因为伴随知识更新节奏，我们一刻不敢停歇。虽然我们年轻，但我们拥有：

“严谨、高效、协作”的团队精神

全方位、立体化的服务意识

实力雄厚的作者群和开发队伍

当然，最重要的是我们拥有：

恒久不变的理想和永不枯竭的激情和灵感

正因如此，我们敢于宣称：

飞思科技 = 丰富的内容 + 完美的形式

这也是我们共同精心培育的品牌  的承诺。

“问渠哪得清如许，为有源头活水来”。路再远，终需用脚去量；风景再美，终需自然抚育。

年轻的飞思人愿为清风细雨、阳光晨露，滋润您发芽、成长；更甘当坚实的铺路石，为您铺就成功之路。

前 言

Jakarta Tomcat 服务器是在 SUN 公司的 JSWDK (JavaServer Web DevelopmentKit, SUN 公司推出的小型 Servlet/JSP 调试工具)的基础上发展起来的一个优秀的 Java Web 应用容器,它是 Apache-Jakarta 的一个子项目。Tomcat 被 JavaWorld 杂志的编辑选为 2001 年度最具创新的 Java 产品(Most Innovative Java Product),同时它又是 SUN 公司官方推荐的 Servlet/JSP 容器(参见 <http://java.sun.com/products/jsp/tomcat/>),因此它受到越来越多软件公司和开发人员的喜爱。Servlet 和 JSP 的最新规范都在 Tomcat 的新版本中得到了实现。

作为一个开放源码的软件, Tomcat 得到了开放源码志愿者的广泛支持,它可以和目前大部分的主流 HTTP 服务器(如 IIS 和 Apache 服务器)一起工作,而且运行稳定、可靠、效率高。

作者根据多年的 Java Web 开发经验,详细阐明了在最新的 Tomcat 5.x 版本上开发 Java Web 应用涉及的各种技术,并且介绍了如何将 Tomcat 和其他主流 HTTP 服务器集成并创建具有实用价值的企业 Java Web 应用的方案。

本书的组织结构和主要内容

本书的内容总体上分为两部分,一部分介绍如何配置 Tomcat 服务器,从而为 Java Web 应用创建高效的开发和运行环境;还有一部分依据 SUN 的 Java Servlet 规范,介绍了开发 Java Web 的各种技术。在组织结构上,这两部分内容穿插在一起,贯穿全书。本书按照由浅到深、前后呼应的顺序来安排内容。本书涉及以下内容。

1. Tomcat 的基础知识

包括如下内容:

- (1) Tomcat 服务器结构和安装步骤
- (2) Java Web 应用的结构和发布
- (3) 配置虚拟主机

2. Java Servlet、JavaServer Page 以及 Java Web 应用的基础知识

包括如下内容:

- (1) Servlet 的原理
- (2) 创建 Servlet 的基本步骤
- (3) JSP 语法
- (4) 在 Java Web 应用中访问数据库,配置数据源
- (5) 在 Java Web 应用中访问 Java Bean, Java Bean 在不同范围内的生命周期
- (6) 在 Java Web 应用中使用 Session
- (7) 使用 ant 工具来管理 Web 应用

3. Java Web 开发高级技术

包括如下内容:

- (1) Servlet 过滤器

- (2) 创建自定义 JSP 标签
- (3) 网站的模板设计
- (4) 开发 Java Mail Web 应用, 配置 Mail Session

4. Tomcat 与当前其他通用软件的集成

包括如下内容:

- (1) Struts 和 MVC 设计模式
- (2) 使用 Log4J 进行日志操作
- (3) Tomcat 与 Jboss 服务器集成, 创建 J2EE 应用
- (4) Tomcat 与 Apache SOAP 集成
- (5) Tomcat 与 Apache AXIS 集成
- (6) Tomcat 与其他 HTTP 服务器 (如 Apache Web 服务器和 IIS Web 服务器) 集成
- (7) Velocity 模板语言

5. Tomcat 的高级功能

包括如下内容:

- (1) 持久性会话管理
- (2) Tomcat 的控制和管理平台
- (3) 安全域
- (4) Tomcat 阀
- (5) 创建嵌入式 Tomcat
- (6) 在 Tomcat 中配置 SSL

本书的范例程序

本书将 3 个 Web 应用的例子贯穿全书:

- helloapp 应用
- bookstore (网上书店) 应用
- Java Mail Web (javamail) 应用

1. helloapp 应用

本书通过 helloapp 应用的例子来讲解 Java Web 开发的基础知识, 比如, 在第 2 章以 helloapp 应用为例, 讲述了发布 Web 应用的步骤。在其他章节中, 所有针对单个知识点的 Servlet 和 JSP 例子, 都被发布到 helloapp 应用中。

2. bookstore 应用

bookstore 应用是一个充分运用了所有 Java Web 开发技术的综合例子, 它实现了一个网上书店, 更加贴近于实际应用。为了便于读者循序渐进地掌握 Java Web 技术, 在书中提供了 bookstore 应用的 4 个版本, 它们分别侧重于某些技术。

(1) bookstore version0: 通过这个例子读者可以进一步掌握 JSP 编程的技巧, 能够灵活运用 Java Bean 和 Session, 并掌握通过 JDBC 访问数据库的技术。

(2) bookstore version1: 该例介绍如何在 Tomcat 中配置 JNDI DataSource, 以及如何在 Web 应用中访问 JNDI DataSource。

(3) bookstore version2: 使读者掌握创建 JSP 自定义标签的高级技术, 并掌握对网页

进行模板设计的方法。

(4) bookstore version3: 实现了基于 J2EE 架构的 bookstore 应用, 并介绍了在 Jboss-Tomcat 的集成服务器上发布 J2EE 应用的方法。

3. javamail 应用

Java Mail Web 应用是一个基于 Web 的邮件客户程序, 它向 Web 客户提供了访问 Mail 服务器上的邮件账号、进行收发信件和管理邮件夹等功能。通过这个例子, 读者可以了解电子邮件的发送和接收协议, 掌握 Java Mail API 的使用方法, 以及通过 Java Mail API 创建 Java Mail Web 应用的过程。通过这个例子, 读者还可以掌握在 Tomcat 中配置 Mail Session 的步骤, 以及在 Web 应用中访问 Mail Session 的方法。

这本书是否适合你

本书面向所有打算或已经开发 Java Web 应用的读者。尽管本书在讲解 Java Web 技术时以 Tomcat 作为开发和运行平台, 但由于 Java Web 技术是 SUN 公司在 Java Servlet 规范中提出的通用技术, 因此本书讲解的应用例程可以运行在任何一个实现 SUN 的 Servlet 规范的 Java Web 服务器上。另一方面, 由于 Tomcat 是 SUN 公司官方推荐的 Servlet/JSP 容器, 因此在学习 Java Web 开发技术或进行实际的开发工作时, Tomcat 是首选的 Java Web 应用容器。

如果你是开发 Java Web 应用的新手, 建议按照本书的先后顺序来学习。如果你已经在 Tomcat 上开发 Java Web 应用方面有丰富经验, 则可以把本书作为实用的 Tomcat 技术参考资料。本书详细介绍了如何把 Tomcat 与当前其他通用的 Web 技术集成, 以及 Tomcat 的各种高级功能。灵活运用本书介绍的 Tomcat 最新技术, 将使 Java Web 应用开发更加得心应手。

实践是掌握 Java Web 技术最迅速、有效的办法。本书提供了大量典型的例子, 在本书配套光盘上提供了完整的源代码, 以及软件安装程序。本书所有程序都在 Tomcat 5.0.12 版本中测试通过, 读者可以按照书上介绍的详细步骤亲自动手, 在本地机器上配置 Tomcat 开发和运行环境, 然后创建和发布 Java Web 应用。

光盘使用说明

本书配套光盘包含以下目录。

1. software 目录

在该目录下包含了本书涉及到的所有软件的最新版安装程序, 包括:

- (1) JDK 的安装软件 (JDK1.4.2)
- (2) Tomcat 的安装软件 (Tomcat5.0.12)
- (3) MySQL 服务器的安装软件 (MySQL4.0.14)
- (4) Apache 服务器的安装软件 (Apache2.0.47)
- (5) Ant 的安装软件 (Ant1.5.4)
- (6) Apache SOAP 软件 (Apache SOAP2.3.1)
- (7) Apache AXIS 软件 (Apache AXIS1.1)
- (8) Log4J 软件 (Log4J1.2.8)

- (9) Struts 软件 (2003/09/03 发布)
- (10) Jboss 与 Tomcat 的集成软件 (Jboss-3.2.1_tomcat-4.1.24)
- (11) Mail Server 的安装软件 (Merak Mail Server 试用版)
- (12) Velocity 软件 (Velocity 1.3.1)

2. lib 目录

在该目录下提供了运行与本书有关的 Java Web 应用, 以及配置 Tomcat 服务器时所需的 JAR 文件和 DLL 文件:

- (1) activation.jar (JavaBeans Activation Framework API)
- (2) mail.jar (Java Mail API)
- (3) log4j-1.2.8.jar (Log4J API)
- (4) soap.jar (SOAP API)
- (5) xerces.jar (Xerces API)
- (6) struts.jar (Struts API)
- (7) j2ee.jar (J2EE API)
- (8) isapi_redirect.dll (Tomcat 与 IIS 集成的 JK 插件)
- (9) mod_jk_2.0.46.dll (Windows/NT 2000 下 Tomcat 与 Apache 集成的 JK 插件)
- (10) mod_jk.so-ap2.0.46-rh72..46-rh72 (Linux 下 Tomcat 与 Apache 集成的 JK 插件)
- (11) jsp-api.jar (JSP API)
- (12) servlet-api.jar (Server API)
- (13) mysqldriver.jar (MySQL 驱动程序)

3. sourcecode 目录

在该目录下提供了本书所有的源程序, 每一章的源程序位于相应的 sourcecode/chapterX 目录下 (X 代表章节号)。bookstore 应用和 javamail 应用分别位于 sourcecode/bookstores 和 sourcecode/javamails 目录下。

本书在编写过程中得到了 Apache 软件组织和 SUN 公司在技术上的大力支持, 飞思科技产品研发中心负责监制工作, 此外曹文伟、潘玉峰和李红军为本书的编写提供了有益的帮助, 在此表示衷心的感谢! 尽管我们尽了最大努力, 但本书难免会有不妥之处, 欢迎各界专家和读者朋友批评指正, 我们的联系方式是:

电 话: (010) 68134545 68131648

电子邮件: support@fecit.com.cn tomcat_author@yahoo.com.cn

飞思在线: <http://www.fecit.com.cn> <http://www.fecit.net>

答疑网址: <http://www.fecit.com.cn/question.htm>

通用网址: 计算机图书、飞思、飞思教育、飞思科技、FECIT

编 者

飞思科技产品研发中心

目 录

第 1 章 Tomcat 简介	1
1.1 Tomcat 与 Servlet 容器	1
1.2 Tomcat 的结构	2
1.3 Java Web 应用简介	4
1.4 Tomcat 的工作模式	5
1.5 Tomcat 的版本	6
1.6 安装和配置 Tomcat 所需的资源	7
1.7 安装 Tomcat	7
1.8 测试 Tomcat 的安装	9
1.9 Tomcat 的运行脚本	11
1.10 小结	12
第 2 章 创建和发布 Web 应用	13
2.1 Tomcat 的目录结构	13
2.2 创建和发布 Web 应用	14
2.3 配置虚拟主机	25
2.4 小结	27
第 3 章 Servlet 技术	29
3.1 Servlet 简介	29
3.2 Servlet API	29
3.3 Servlet 的生命周期	32
3.4 HTTP 与 HttpServlet	33
3.5 创建 HttpServlet 的步骤	37
3.6 ServletContext 和 Web 应用的关系	39
3.7 小结	42
第 4 章 JSP 技术	43
4.1 JSP 简介	43
4.2 JSP 语法	44
4.3 JSP 与 Cookie	50
4.4 转发 JSP 请求	53
4.5 JSP 异常处理	54
4.6 再谈部署 JSP	57
4.7 小结	57
第 5 章 bookstore 应用简介	59
5.1 bookstore 应用的软件结构	59
5.2 浏览 bookstore 应用的 JSP 网页	60

5.3	JavaBean 和实用类	66
5.4	发布 bookstore 应用	71
5.5	小结	72
第 6 章	访问数据库	73
6.1	安装和配置 MySQL 数据库	73
6.2	通过 JDBC 访问数据库	75
6.3	数据源 (DataSource) 简介	89
6.4	配置数据源	90
6.5	程序中访问数据源	93
6.6	处理中文编码	101
6.7	小结	102
第 7 章	Session 的使用与管理	105
7.1	Session 简介	105
7.2	Session 范例程序	107
7.3	Session 的跟踪	111
7.4	Session 的持久化	115
7.5	小结	121
第 8 章	访问 JavaBean	123
8.1	JavaBean 简介	123
8.2	JSP 访问 JavaBean 的语法	124
8.3	JavaBean 的范围	125
8.4	在 bookstore 应用中访问 JavaBean	128
8.5	小结	134
第 9 章	用 ant 工具管理 Web 应用	135
9.1	安装配置 ant	135
9.2	创建 build.xml 文件	135
9.3	运行 ant	140
9.4	小结	141
第 10 章	Tomcat 的控制平台和管理平台	143
10.1	访问 Tomcat 的控制平台和管理平台	143
10.2	Tomcat 的控制平台	144
10.3	Tomcat 的管理平台	148
10.4	小结	150
第 11 章	安全域	151
11.1	安全域概述	151
11.2	为 Web 资源设置安全约束	152
11.3	内存域	158
11.4	JDBC 域	160
11.5	DataSource 域	162

11.6	在 Web 应用中访问用户信息	165
11.7	小结	166
第 12 章	Tomcat 阀	167
12.1	Tomcat 阀简介	167
12.2	客户访问日志阀	167
12.3	远程地址过滤器	169
12.4	远程主机过滤器	170
12.5	客户请求记录器	171
12.6	小结	172
第 13 章	Servlet 过滤器	173
13.1	Servlet 过滤器简介	173
13.2	创建 Servlet 过滤器	174
13.3	发布 Servlet 过滤器	177
13.4	串联 Servlet 过滤器	181
13.5	小结	194
第 14 章	自定义 JSP 标签	195
14.1	自定义 JSP 标签简介	195
14.2	创建标签处理类	196
14.3	创建标签库描述文件	201
14.4	在 Web 应用中使用标签	203
14.5	发布支持中、英文版本的 helloapp 应用	206
14.6	小结	208
第 15 章	采用模板设计网上书店应用	209
15.1	如何设计网站的模板	209
15.2	创建负责流程控制的 Servlet	210
15.3	创建模板标签和模板 JSP 文件	212
15.4	修改 JSP 文件	226
15.5	发布采用模板设计的 bookstore 应用	227
15.6	小结	231
第 16 章	Struts 和 MVC 设计模式	233
16.1	MVC 设计模式简介	233
16.2	Struts 实现的 MVC 设计模式	234
16.3	创建采用 Struts 的 Web 应用	238
16.4	运行 helloapp-struts 应用	246
16.5	小结	251
第 17 章	使用 Log4J 进行日志操作	253
17.1	Log4J 简介	253
17.2	Log4J 的基本使用方法	257
17.3	在 helloapp 应用中使用 Log4J	262

17.4	小结.....	265
第 18 章	Tomcat 与 Jboss 集成.....	267
18.1	安装 Jboss 和 Tomcat 整合服务器.....	267
18.2	J2EE 体系结构简介.....	268
18.3	创建 EJB 组件.....	270
18.4	在 Web 应用中访问 EJB 组件.....	275
18.5	发布 J2EE 应用.....	277
18.6	小结.....	283
第 19 章	开发 Java Mail Web 应用.....	285
19.1	E-mail 协议简介.....	285
19.2	Java Mail API 简介.....	287
19.3	Java Mail 应用程序开发环境.....	288
19.4	创建 Java Mail 应用程序.....	291
19.5	Java Mail Web 应用简介.....	294
19.6	Java Mail Web 应用的程序结构.....	295
19.7	在 Tomcat 中配置 Mail Session.....	318
19.8	发布和运行 javamail 应用.....	321
19.9	小结.....	322
第 20 章	Tomcat 与 Apache SOAP 集成.....	323
20.1	SOAP 简介.....	323
20.2	建立 Apache SOAP 环境.....	325
20.3	在 Tomcat 上发布 Apache-SOAP Web 应用.....	326
20.4	创建 SOAP 服务.....	327
20.5	管理 SOAP 服务.....	329
20.6	创建和运行 SOAP 客户程序.....	332
20.7	小结.....	335
第 21 章	Tomcat 与 Apache AXIS 集成.....	337
21.1	建立 Apache AXIS 环境.....	337
21.2	在 Tomcat 上发布 Apache-AXIS Web 应用.....	337
21.3	创建 SOAP 服务.....	338
21.4	管理 SOAP 服务.....	340
21.5	创建和运行 SOAP 客户程序.....	341
21.6	发布 JWS 服务.....	344
21.7	小结.....	345
第 22 章	Tomcat 与其他 HTTP 服务器集成.....	347
22.1	Tomcat 与 HTTP 服务器集成的原理.....	347
22.2	在 Windows 下 Tomcat 与 Apache 服务器集成.....	349
22.3	在 Linux 下 Tomcat 与 Apache 服务器集成.....	353
22.4	Tomcat 与 IIS 服务器集成.....	355

22.5	小结.....	362
第 23 章	创建嵌入式 Tomcat 服务器.....	363
23.1	将 Tomcat 嵌入 Java 应用.....	363
23.2	创建嵌入了 Tomcat 的 Java 示范程序.....	365
23.3	运行嵌入式 Tomcat 服务器.....	370
23.4	小结.....	372
第 24 章	在 Tomcat 中配置 SSL.....	373
24.1	SSL 简介.....	373
24.2	在 Tomcat 中使用 SSL.....	376
24.3	小结.....	380
第 25 章	JSP 2.0 的新特征.....	381
25.1	JSP 表达式语言.....	381
25.2	简单标签扩展.....	389
25.3	小结.....	393
第 26 章	Velocity 模板语言.....	395
26.1	安装 Velocity.....	395
26.2	Velocity 的简单例子.....	395
26.3	注释.....	399
26.4	引用.....	400
26.5	指令.....	405
26.6	其他特征.....	414
26.7	小结.....	415
附录 A	server.xml 文件.....	417
A.1	配置 Server 元素.....	420
A.2	配置 Service 元素.....	420
A.3	配置 Engine 元素.....	420
A.4	配置 Host 元素.....	421
A.5	配置 Context 元素.....	422
A.6	配置 Connector 元素.....	422
附录 B	web.xml 文件.....	425
B.1	配置 Servlet 过滤器.....	427
B.2	配置 Servlet.....	428
B.3	配置 Servlet 映射.....	429
B.4	配置 Session.....	429
B.5	配置 Welcome 文件清单.....	430
B.6	配置 Tag Library.....	430
B.7	配置资源引用.....	430
B.8	配置安全约束.....	431
B.9	配置安全验证登录界面.....	432

B.10 配置对安全验证角色的引用	432
附录 C XML 简介	433
C.1 SGML、HTML 与 XML 的比较.....	433
C.2 DTD 文档类型定义.....	434
C.3 有效 XML 文档以及简化格式的 XML 文档.....	435
C.4 XML 中的常用术语	436

第 1 章 Tomcat 简介

Jakarta Tomcat 服务器是在 SUN 公司的 JSWDK (JavaServer Web DevelopmentKit, 是 SUN 公司推出的小型 Servlet/JSP 调试工具) 的基础上发展起来的一个优秀的 Servlet/JSP 容器, 它是 Apache-Jakarta 软件组织的一个子项目。它不但支持运行 Servlet 和 JSP, 而且还具备了作为商业 Java Web 应用容器的特征。

作为一个开放源码的软件, Tomcat 得到了开放源码志愿者的广泛支持, 它可以和目前大部分的主流 HTTP 服务器 (如 IIS 和 Apache 服务器) 一起工作, 而且运行稳定、可靠、效率高。

Tomcat 服务器除了能够运行 Servlet 和 JSP, 还提供了作为 Web 服务器的一些特有功能, 如 Tomcat 管理和控制平台、安全域管理和 Tomcat 阀等。Tomcat 已成为目前开发企业 Java Web 应用的最佳选择之一。

本章简单介绍 Tomcat 作为 Servlet 容器的工作过程, 以及 Tomcat 的结构和工作模式, 最后介绍安装 Tomcat 服务器的步骤。

1.1 Tomcat 与 Servlet 容器

Jakarta Tomcat 服务器是一种 Servlet/JSP 容器。Servlet 是一种运行在支持 Java 语言的服务器上的组件。Servlet 最常见的用途是扩展 Java Web 服务器功能, 提供非常安全的、可移植的、易于使用的 CGI 替代品。它是一种动态加载的模块, 为来自 Web 客户的请求提供服务。它完全运行在 Java 虚拟机上。由于它在服务器端运行, 因此它的运行不依赖于浏览器。

Tomcat 作为 Servlet 容器, 负责处理客户请求, 把请求传送给 Servlet 并把结果返回给客户。Servlet 容器与 Servlet 之间的接口是由 Java Servlet API 定义的, 在 Java Servlet API 中定义了 Servlet 的各种方法, 这些方法在 Servlet 生命周期的不同阶段被 Servlet 容器调用; Servlet API 还定义了 Servlet 容器传递给 Servlet 的对象类, 如请求对象 ServletRequest 和响应对象 ServletResponse。

当客户请求访问某个 Servlet 时, Servlet 容器将创建一个 ServletRequest 对象和 ServletResponse 对象。在 ServletRequest 对象中封装了客户请求信息, 然后 Servlet 容器把 ServletRequest 对象和 ServletResponse 对象传给客户所请求的 Servlet。Servlet 把响应结果写到 ServletResponse 中, 然后由 Servlet 容器把响应结果传给客户。Servlet 容器响应客户请求的过程如图 1-1 所示。



图 1-1 Servlet 容器响应客户请求的过程

本书将讨论 Tomcat 和 Java Web 开发的各种技术,如果要获取关于 Tomcat 的更多信息,可以访问 Tomcat 的主页 (<http://jakarta.apache.org/tomcat/index.html>), 如图 1-2 所示。

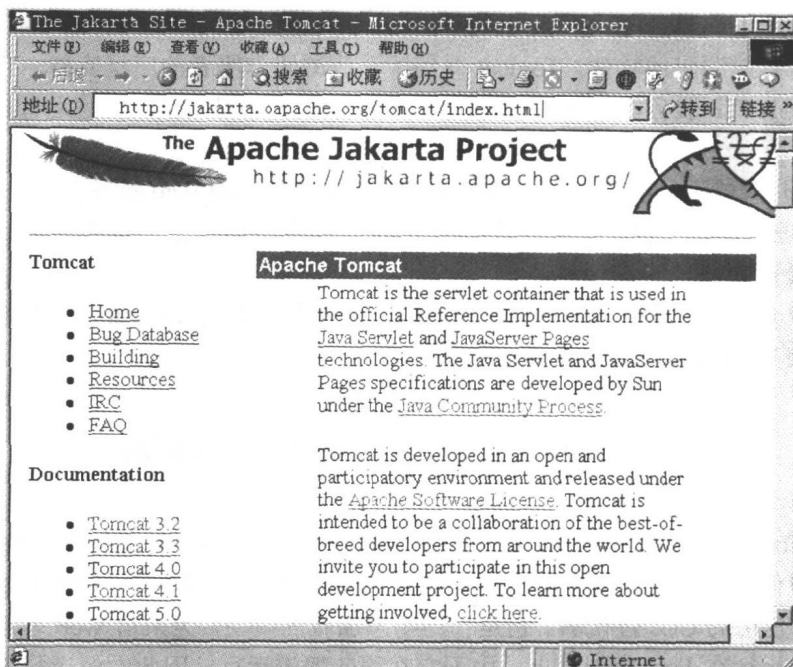


图 1-2 Tomcat 主页

1.2 Tomcat 的结构

Tomcat 服务器是由一系列可配置的组件构成,其中核心组件是 Catalina Servlet 容器,它是所有其他 Tomcat 组件的顶层容器。Tomcat 的组件可以在 `<CATALINA_HOME>/conf/server.xml` 文件中进行配置,每个 Tomcat 组件在 `server.xml` 文件中对应一种配置元素。以下代码以 XML 的形式展示了各种 Tomcat 组件之间的关系:

```
<Server>
  <Service>
    <Connector />
    <Engine>
      <Host>
        <Context>
        </Context>
      </Host>
    </Engine>
  </Service>
</Server>
```