

Java Web

开发技术与项目实战

Java Web Kaifa Jishu Yu Xiangmu Shizhan

► 黎才茂 邱 钊 等 / 编著

中国科学技术大学出版社

Java Web 开发技术与项目实战

黎才茂 邱 钊 符 发 陈少凡
黄 萍 纪洲鹏 郭 桢 / 编著

中国科学技术大学出版社

内 容 简 介

Java Web 技术是目前较流行、发展较快的一种编程技术,在 Web 开发领域占有重要地位。由于其开放和跨平台的特点,吸引了众多的开发人员和软件公司。同时,在众多开发人员的努力下,出现了许多优秀的开源框架,为 Java Web 技术在企业级开发领域的应用注入了新的活力。

本书共 12 章,从 HTML 语言开始,到 Struts 2 框架的开发,讲述了如何使用 Java Web 技术开发应用系统。书中主要内容包括 Web 基础、HTML 基础、CSS 基础、JavaScript 基础、JSP 技术、JavaBean 技术、JDBC 技术、Servlet 技术、EL 表达式、JSTL 标签库、Struts 2 框架等。每章内容都涵盖了理论和实践教学的全过程,有助于读者更好地掌握知识和提高动手能力。

本书可作为大学本科和专科相关课程的教材、课程设计用书或教学参考书,也可作为从事 Java Web 应用系统开发的技术人员的学习、培训教材或参考书。

图书在版编目(CIP)数据

Java Web 开发技术与项目实战/黎才茂,邱钊等编著. —合肥:中国科学技术大学出版社, 2016. 8

ISBN 978-7-312-03976-8

I. J… II. ①黎… ②邱… III. Java 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2016)第 179249 号

出版 中国科学技术大学出版社
安徽省合肥市金寨路 96 号,230026
网址: <http://press.ustc.edu.cn>
印刷 安徽省瑞隆印务有限公司
发行 中国科学技术大学出版社
经销 全国新华书店
开本 787 mm×1092 mm 1/16
印张 19
字数 500 千
版次 2016 年 8 月第 1 版
印次 2016 年 8 月第 1 次印刷
定价 39.00 元

前 言

随着 Java 语言的广泛流行,Java 技术在企业级应用开发中使用得越来越普遍,Java Web 技术已经成为基于 Web 的 Java 企业级应用解决方案中不可缺少的重要组成部分。学习 Java Web 开发技术不仅是一种技术时尚,更是一种技术人才的需求。现在国内许多高校的计算机专业普遍开设了 Java Web 开发课程,对此,各出版社出版了不少讲解 Java Web 开发技术的书籍,但是许多书存在知识点讲解不完整,缺少必需的基础技术知识的讲解,甚至是直接面向企业级开发的中高级应用而编写的,所以,要么是技术基础知识涵盖不全,要么是技术知识点比较高级,不能适应初级读者的学习需要。

本书是假设读者完全不具有基本网页技术知识的前提编写的。本书从网页设计基础知识入手,系统完整地讲解了 Java Web 开发中的各种技术,从知识的讲解到知识的运用,一步一步地引导读者掌握 Java Web 开发的知识体系结构。本书讲解内容尽可能配以简单示例,让读者理解每一个知识点在程序中的应用。通过每一章给出的综合实例,使得读者加深对这章内容的理解和掌握,并希望通过综合实例的设计实现,更好地提高学习者的动手能力。通过对本书的系统学习,读者可以快速掌握 JSP 技术,提高网页编程能力和实际应用开发水平。

本书内容主要包含以下几个方面:

- Web 基础。介绍 Web 开发基础,包括 Web 基本概念、B/S 体系结构、Web 常用技术、开发环境的安装与配置。
- HTML 基础。讲解 HTML 语言知识,包括 HTML 基本元素、表格元素、表单元素和框架应用。
 - CSS 基础。讲解 CSS 知识,包括 CSS 规则、CSS 调用、CSS 选择器、CSS 样式。
 - JavaScript 基础。讲解 JavaScript 脚本语言,包括 JavaScript 引用方式、语法、控制语句与函数、事件驱动与浏览器对象。
- JSP 技术。讲解 JSP 技术,包括 JSP 语法、指令元素、动作元素、内置对象和应用实例。
- JavaBean 技术。讲解 JavaBean 技术,包括 JSP+JavaBean 设计模式、JavaBean 属性和方法及其作用范围、JavaBean 组件应用实例。
- JDBC 技术。讲解 JDBC 技术,包括 JDBC 驱动程序、常用接口与类、JDBC 配置和与不同数据库的连接、JDBC 连接 MySQL 数据库实例。
- Servlet 技术。讲解 Servlet 技术,包括 JSP+Servlet 设计模式、Servlet 基本结

构、Servlet 生命周期及应用实例。

- EL 表达式。讲解 EL 表达式,包括 EL 格式、语法、隐含对象和自定义函数以及应用实例。

- JSTL 标签库。讲解 JSTL 标签库,包括 JSTL 简介、核心标签库、格式化标签库、函数标签库及应用实例。

- Struts 2 框架。讲解 Struts 2 框架,包括 Struts 2 简介、基础知识、开发步骤及基本应用开发实例。

- JSP 开发模式应用实例。讲解一个 JSP 开发模式的综合应用实例,包括系统功能分析、数据库设计与创建、模块设计与功能实现。

本书的配套资源包括教学课件、各章节的示例代码和“网上书店”的系统源代码文件,以方便学校课程教学、读者学习和参考。读者可以登录出版社提供的资源网站下载,或者直接向作者索取(电子邮箱:lcaim@126.com)。

本书适用于大学本科计算机类专业、高职高专计算机及相关专业的教学,也可作为从事 Java Web 应用系统开发的技术人员的学习、培训教材和参考书。

本书共分 12 章,分别由海南大学的黎才茂、邱钊、符发、陈少凡、黄萍、郭祯和海口经济学院的纪洲鹏编写。其中,第 1、7 章由纪洲鹏编写,第 2、8 章由符发编写,第 3、4 章由郭祯编写,第 5 章由陈少凡编写,第 6、9 章由邱钊编写,第 10 章由黄萍编写,第 11、12 章由黎才茂编写。全书由黎才茂、邱钊负责统稿和定稿。

这里要特别感谢参考文献中所列的各位作者,包括众多未能在参考文献中一一列出的作者,正是他们提供的宝贵参考资料,使得编者能形成本书完整的编写思路以及能够及时完成编著。

由于 Java Web 开发技术及应用涉及的内容非常广泛,再加上时间仓促和编者的水平有限,书中难免会存在错误、疏漏和不妥之处,诚望读者不吝赐教,对错误、疏漏、不妥之处给以批评指正。

编者

2016 年 7 月于海南大学

目 录

前言	(i)
第 1 章 Web 基础	(1)
1.1 Web 简介	(1)
1.1.1 Web 概念	(1)
1.1.2 Web 技术的发展	(1)
1.1.3 URI 和 URL	(1)
1.1.4 HTTP 协议	(2)
1.2 B/S 体系结构	(3)
1.2.1 Web 应用	(3)
1.2.2 Web 工作原理	(3)
1.3 Web 常用技术	(4)
1.3.1 HTML	(4)
1.3.2 CSS	(5)
1.3.3 JavaScript	(5)
1.3.4 动态网页技术	(5)
1.3.5 Servlet	(6)
1.3.6 Struts	(6)
1.4 开发环境安装与配置	(6)
第 2 章 HTML 基础	(12)
2.1 HTML 基本元素	(12)
2.1.1 HTML 简介	(12)
2.1.2 HTML 基本语法	(12)
2.1.3 HTML 标记	(13)
2.2 表格元素	(19)
2.2.1 表格功能	(19)
2.2.2 表格制作	(20)
2.2.3 表格美化	(22)
2.2.4 特殊表格	(22)
2.3 表单元素	(24)
2.3.1 表单基本格式	(24)
2.3.2 表单制作	(25)
2.4 框架应用	(27)

2.4.1 建立框架	(27)
2.4.2 混合框架	(28)
第3章 CSS 基础	(30)
3.1 CSS 规则	(30)
3.1.1 基本语法	(30)
3.1.2 CSS 规则	(31)
3.2 CSS 调用	(32)
3.2.1 内联样式	(32)
3.2.2 内嵌样式	(33)
3.2.3 导入样式	(33)
3.2.4 链接样式	(34)
3.3 CSS 选择器	(35)
3.3.1 元素选择器	(35)
3.3.2 类选择器	(35)
3.3.3 ID 选择器	(36)
3.3.4 派生选择器	(37)
3.4 CSS 样式	(40)
3.4.1 CSS 背景	(40)
3.4.2 CSS 文本	(42)
3.4.3 CSS 字体	(45)
3.4.4 CSS 链接	(47)
3.4.5 CSS 列表	(48)
3.4.6 CSS 表格	(50)
3.4.7 CSS 轮廓	(51)
第4章 JavaScript 基础	(53)
4.1 JavaScript 引用方式	(53)
4.1.1 嵌入方式	(53)
4.1.2 引入方式	(54)
4.2 JavaScript 语法	(55)
4.2.1 基本数据类型	(55)
4.2.2 常量	(55)
4.2.3 变量	(55)
4.2.4 表达式和运算符	(56)
4.3 JavaScript 控制语句与函数	(57)
4.3.1 JavaScript 控制语句	(57)
4.3.2 JavaScript 函数	(59)
4.4 JavaScript 事件驱动与浏览器对象	(63)
4.4.1 事件处理程序	(63)
4.4.2 事件驱动	(63)

4.4.3 JavaScript 浏览器对象	(65)
第 5 章 JSP 技术	(70)
5.1 JSP 语法	(70)
5.1.1 JSP 脚本	(70)
5.1.2 JSP 声明	(71)
5.1.3 JSP 表达式	(72)
5.1.4 JSP 注释	(73)
5.1.5 JSP 运算符与常量	(74)
5.2 JSP 指令元素	(74)
5.2.1 page 指令	(74)
5.2.2 include 指令	(76)
5.2.3 taglib 指令	(77)
5.3 JSP 动作元素	(77)
5.3.1 <jsp:include>动作	(78)
5.3.2 <jsp:forward>动作	(78)
5.3.3 <jsp:params>和<jsp:param>动作	(79)
5.3.4 <jsp:plugin>动作	(79)
5.3.5 <jsp:useBean>动作	(80)
5.3.6 <jsp:setProperty>和<jsp:getProperty>动作	(81)
5.4 JSP 内置对象	(82)
5.4.1 out 对象	(83)
5.4.2 request 对象	(85)
5.4.3 response 对象	(90)
5.4.4 session 对象	(96)
5.4.5 application 对象	(99)
5.4.6 pageContext 对象	(101)
5.4.7 exception 对象	(102)
5.5 JSP 应用实例	(103)
第 6 章 JavaBean 技术	(109)
6.1 JSP+JavaBean 设计模式	(109)
6.1.1 JSP 基本设计模式	(109)
6.1.2 JSP+JavaBean 设计模式	(109)
6.2 JavaBean 属性与方法	(111)
6.2.1 简单属性	(111)
6.2.2 索引属性	(112)
6.3 JavaBean 作用范围与属性访问	(113)
6.3.1 JavaBean 的作用范围	(114)
6.3.2 访问 JavaBean 属性	(114)
6.3.3 JSP 调用 JavaBean	(115)

6.4	JavaBean 应用实例	(116)
6.4.1	添加新书	(116)
6.4.2	购物车的实现	(121)
第7章	JDBC 技术	(127)
7.1	JDBC 技术与驱动程序	(127)
7.1.1	JDBC 概述	(127)
7.1.2	JDBC 驱动程序	(128)
7.2	JDBC 常用接口与类	(128)
7.2.1	JDBC API	(128)
7.2.2	Statement 接口的主要方法	(129)
7.2.3	PreparedStatement 接口的主要方法	(129)
7.3	JDBC 与不同数据库的连接	(130)
7.3.1	JDBC 连接数据库一般步骤	(130)
7.3.2	数据库连接池简介	(134)
7.3.3	其他常见数据库的连接	(136)
7.4	JDBC 连接 MySQL 数据库实例	(137)
第8章	Servlet 技术	(144)
8.1	JSP+Servlet 设计模式	(144)
8.1.1	Servlet 概述	(144)
8.1.2	JSP+Servlet 设计模式	(145)
8.2	Servlet 生命周期	(146)
8.2.1	Servlet 生命周期	(146)
8.2.2	简单 Servlet 举例	(147)
8.3	Servlet 常用接口	(152)
8.3.1	Servlet 的实现接口	(152)
8.3.2	Servlet 的配置接口	(155)
8.3.3	Servlet 的上下文接口	(155)
8.3.4	Servlet 的请求与响应接口	(156)
8.3.5	Servlet 的会话跟踪接口	(158)
8.3.6	Servlet 的请求调度接口	(158)
8.3.7	Servlet 的过滤功能	(159)
8.4	Servlet 表单处理	(160)
8.4.1	获取 HTTP 请求信息	(160)
8.4.2	生成 HTTP 请求响应并返回给客户	(161)
8.4.3	中文乱码问题	(162)
8.4.4	表单处理示例	(163)
8.5	Servlet 应用实例	(168)
第9章	EL 表达式	(179)
9.1	EL 格式	(179)

9.2 EL 语法	(180)
9.2.1 作用范围及变量	(180)
9.2.2 算术运算	(181)
9.2.3 关系运算	(183)
9.2.4 逻辑运算	(184)
9.2.5 “.”和“[]”运算	(184)
9.2.6 empty 运算	(185)
9.2.7 EL 保留字	(185)
9.2.8 自动类型转换	(185)
9.2.9 运算符的优先级	(186)
9.3 EL 隐含对象	(186)
9.3.1 pageContext 对象	(187)
9.3.2 param 和 paramValues 对象	(189)
9.3.3 header 和 headerValues 对象	(190)
9.3.4 cookie 对象	(191)
9.3.5 initParam 对象	(191)
9.3.6 属性范围	(191)
9.4 EL 函数	(192)
9.4.1 标签库的 EL 函数	(192)
9.4.2 自定义 EL 函数	(194)
第 10 章 JSTL 标签库	(197)
10.1 JSTL 简介	(197)
10.1.1 JSTL 标签库	(197)
10.1.2 安装 JSTL	(198)
10.2 核心标签库	(199)
10.2.1 表达操作标签	(199)
10.2.2 流程控制	(204)
10.2.3 循环控制	(208)
10.2.4 URL 操作	(214)
10.3 I18N 国际化标签库	(216)
10.3.1 <fmt:setLocale>设置本地化环境标签	(217)
10.3.2 <fmt:bundle>执行信息资源标签	(217)
10.3.3 <fmt:setBundle>设置资源文件标签	(218)
10.3.4 <fmt:message>获取资源属性值标签	(219)
10.3.5 <fmt:param>获取参数值标签	(220)
10.3.6 <fmt:requestEncoding>设置字符编码标签	(220)
10.4 函数标签库	(220)
第 11 章 Struts 2 框架	(223)
11.1 Struts 2 简介	(223)

11.1.1	Struts 2 框架结构	(223)
11.1.2	Struts 2 配置文件	(224)
11.1.3	Struts 2 控制器	(225)
11.1.4	Struts 2 标签库	(225)
11.2	Struts 2 开发准备	(226)
11.2.1	配置 MyEclipse 开发工具	(226)
11.2.2	下载 Struts 2 框架开发包	(227)
11.3	Struts 2 基本开发实例	(228)
11.3.1	创建一个 Web Project	(228)
11.3.2	加载 Struts 2 框架支持	(229)
11.3.3	修改 web.xml 配置	(231)
11.3.4	创建 JSP 用户页面	(232)
11.3.5	实现 Action 控制器	(233)
11.3.6	配置 struts.xml	(234)
11.3.7	创建结果页面	(235)
11.3.8	工程部署和运行	(235)
第 12 章	JSP 开发模式应用实例	(237)
12.1	系统分析	(238)
12.1.1	系统概述	(238)
12.1.2	需求分析	(238)
12.2	系统总体设计	(239)
12.2.1	系统总体设计	(239)
12.2.2	用户工作流程	(240)
12.3	数据库设计与创建	(241)
12.3.1	数据表定义与创建	(241)
12.3.2	数据库代码的设计	(244)
12.4	客户端模块设计与实现	(246)
12.4.1	用户注册/登录模块	(247)
12.4.2	图书分类模块	(252)
12.4.3	图书浏览与搜索模块	(254)
12.4.4	实现分页功能	(265)
12.4.5	购物车功能	(271)
12.5	管理端模块设计与实现	(281)
12.5.1	浏览图书列表	(281)
12.5.2	添加图书信息	(286)
12.5.3	订单查询	(290)
12.5.4	用户信息	(291)
参考文献		(292)

第 1 章 Web 基础

本章对 Java Web 的基本概念、开发环境、工具和技术进行基本介绍。通过本章的学习，读者能够对 Java Web 的开发环境、工具进行安装和配置，以及了解基本的 Web 技术。

1.1 Web 简介

1.1.1 Web 概念

Web 是 World Wide Web 的简称，也就是我们常说的万维网 WWW。Web 起源于 1980 年欧洲量子物理实验室 Tim Berners Lee 构建的 ENQUIRE 项目。Web 使用超文本技术 HTML 来表示信息资源，以及建立资源与资源之间的链接；使用统一资源定位器(Uniform Resource Locator, URL)定位 Web 服务器中信息资源的位置；使用 HTTP 协议定义客户端与 Web 服务器之间的通信。

1.1.2 Web 技术的发展

Web 技术的发展可以这样简要描述：过去由网站主导的 Web1.0 阶段——现在由用户主导的 Web2.0 阶段——未来向更加个性智能化发展的 Web3.0 时代。

Web1.0 阶段，早期的 Web 网页是静态的，一次展现所有内容，通过网络编程技术(ASP、JSP、PHP 等)在传统静态页面中加入各种程序和逻辑控制，实现客户端和服务器的动态和个性化交流互动，Web 技术逐步过渡到以 Blog、Wiki 和 SNS 等社交软件应用为核心的 Web2.0 阶段。Web3.0 阶段是一个个性智能化的时代，用户通过任何终端，都能看到自己所关心的内容，具体体现包括无处不在的互联网络、云计算、开放软件平台和数据、语义网技术、智能网络和智能应用程序等。Web3.0 将带给使用者全新的用户体验。

1.1.3 URI 和 URL

1. URI

URI(Uniform Resource Identifier)，统一资源标识符，是以特定语法标识某一网络资源的字符串。该种标识允许用户对任何资源通过特定的协议进行交互操作。URI 由模式和模式特有的部分组成，它们之间用冒号隔开，一般格式如下：

scheme:scheme-specific-part

URI 以 scheme 和冒号开头。scheme 用大写/小写字母开头,后面为空或者跟着更多的大写/小写字母、数字、加号、减号和点号。冒号把 scheme 与 scheme-specific-part 分开了,并且 scheme-specific-part 的语法和语义由 URI 的名字空间决定。

URI 的常见模式包括 file(表示本地磁盘文件)、ftp(FTP 服务器)、http(使用 HTTP 协议的 Web 服务器)、mailto(电子邮件地址)等。例如:

```
http://www.hainu.edu.cn
```

其中,“http”是 scheme,“//www.hainu.edu.cn”是 scheme-specific-part,并且它的 scheme 与 scheme-specific-part 被冒号分开了。

2. URL

URL(Uniform Resource Locator),统一资源定位器,指向 Internet 上位于某个位置的某个资源。资源包括 HTML 文件、图像文件和 Servlet 等。我们可以通过浏览器的地址栏输入 URL 地址而获得 Internet 资源。URL 的格式为:

```
scheme://host:port/path
```

其格式说明如下:

- scheme 表示 Internet 资源类型,指明是什么样的 Internet 服务。如“http://”表示 WWW 服务,“ftp://”表示 FTP 服务。
- host 表示服务器地址,指出 Internet 服务所在的服务器域名或者 IP 地址。
- port 表示服务端口,对 Internet 服务资源的访问,需给出相应的服务器端口号,默认端口号可以省略。
- path 表示路径,指明资源在服务器上的相对路径和名称。其格式通常为“目录/子目录/文件名”的结构。与端口一样,路径并非是必需的。

例如:

```
http://www.hainu.edu.cn/stm/vnew/shtml_liebiao.asp@bbsid=95.shtml
```

上面的例子就是一个典型的 URL 地址,客户程序通过“http”(超文本传送协议)识别处理 HTML 链接,“www.hainu.edu.cn”是 host 域名地址,path 资源路径是“/stm/vnew/shtml_liebiao.asp@bbsid=95.shtml”。

1.1.4 HTTP 协议

从本质上讲,Web 的基本结构是开放式的客户机/服务器(Client/Server)体系结构,分成服务器、客户机及通信协议三个部分。Web 服务器通过 Web 浏览器与用户交互操作,相互间采用 HTTP 协议通信。

Web 浏览器与服务器之间遵循 HTTP 协议进行通信传输。HTTP(HyperText Transfer Protocol,超文本传输协议)是 Web 应用的核心技术协议。该协议详细规定了 Web 客户机与服务器之间如何通信。它是一个基于请求—响应(Request-Response)的无状态的协议,这种请求—响应的过程如图 1-1 所示。

HTTP 协议定义了 Web 浏览器向 Web 服务器发送索取 Web 页面请求的格式,以及 Web 页面在 Internet 上的传输方式。用户首先通过浏览器程序建立到 Web 服务器的连接

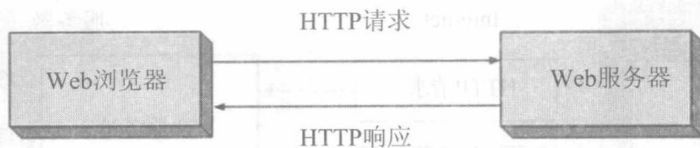


图 1-1 HTTP 请求—响应过程

并向服务器发送 HTTP 请求消息。Web 服务器接收到客户的请求后,对请求进行处理,然后向客户发送回 HTTP 响应。客户接收服务器发送的响应消息,对消息进行处理并关闭连接。

1.2 B/S 体系结构

1.2.1 Web 应用

Web 应用的典型模式为 B/S 模式,即浏览器/服务器(Browser/Server)模式。用户在计算机上使用浏览器向 Web 服务器发出请求,服务器响应客户请求,向客户送回所请求的网页,客户在浏览器窗口上显示网页的内容。

1. Web 服务器

Web 服务器是向浏览器提供服务的程序,主要功能是提供网上信息浏览服务。Web 服务器应用层使用 HTTP 协议,信息内容采用 HTML 文档格式,信息定位使用 URL。常见的 Java Web 服务器有 Tomcat、WebLogic、JBoss、WebSphere 等。本书使用的 Web 服务器是 Apache 服务器,它是 Apache 软件基金会(Apache Software Foundation)提供的开放源代码软件,是一个非常优秀的专业的 Web 服务器。

2. Web 浏览器

浏览器是 Web 服务的客户端程序,可向 Web 服务器发送各种请求,并对从服务器发来的网页和各种多媒体数据进行解释、显示和播放。

浏览器的主要功能是解析网页文件内容并正确显示,网页一般为 HTML 格式。常见的浏览器有 Internet Explorer、Firefox、Opera 和 Chrome。浏览器是最常使用的客户端程序。

3. 通信机制

在 B/S 体系结构中,客户端和服务器之间采用 HTTP 协议进行通信。HTTP 协议是浏览器和 Web 服务器通信的基础,是应用层协议。

1.2.2 Web 工作原理

在 B/S 结构中,Web 服务器接收到 Web 浏览器的请求后,将请求的数据发送到 Web 浏览器,浏览器对接收到的数据进行解释并在屏幕上显示出来,其工作原理如图 1-2 所示。

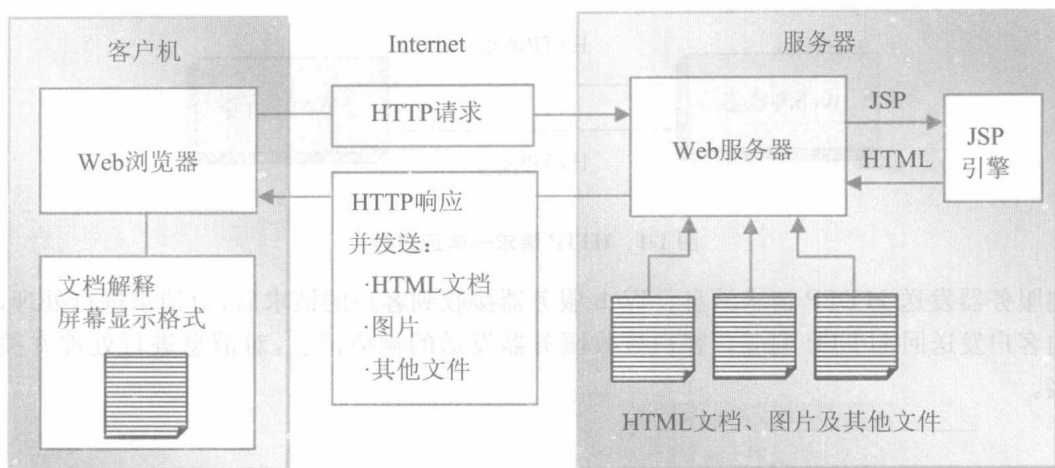


图 1-2 Web 工作原理

图 1-2 中表示的 Web 工作过程实际上是一个 Web 的请求—响应过程,这个过程遵循以下步骤:

(1) 在计算机上运行一个客户机程序(Web 浏览器),如 Microsoft Internet Explorer。

(2) 通过网络与 Internet Web 服务器建立连接。

(3) 向 Internet 上的 Web 服务器请求一个页面,发送一个包含以下内容的消息:

- 传输协议(HTTP)。
- URL 地址,例如 `http://www.hainu.edu.cn/`。

(4) 服务器收到请求后,查找客户机所请求的 HTML 页面文件或其他文件。若客户机所请求的是 JSP 页面,则 Web 服务器将调用 JSP 引擎解释执行 JSP 页面程序,在需要时生成并返回标准的 HTML 页面。

(5) 服务器将所请求的页面文件传到客户机上。

(6) 浏览器接收到服务器传来的页面文件后,对它进行解释并在屏幕上显示出来。

根据以上步骤,我们要想很好地理解 Web 工作原理,除了知道它是客户机/服务器结构外,还需要了解在 WWW 中,Web 服务器做什么工作,客户机上的浏览器做什么工作,客户机和服务器通信时采用的协议,以及浏览器和 Web 服务器之间传输的是什么东西。所以,前一小节所讲述的 Web 系统结构是非常重要的。

1.3 Web 常用技术

1.3.1 HTML

HTML(HyperText Mark-up Language,超文本标记语言)是万维网上应用最为广泛的一种信息表示语言。使用 HTML 语言编写的文件称为 HTML 文件,扩展名为“.html”或者

“.htm”。HTML 语言包括一系列的元素和标签,可以将文本、表格、图片、声音以及动画等组合在一起,进行各种资源的排列及显示。

HTML 语言简单易学、容易掌握。HTML 文件独立于操作系统,只需要使用客户端的浏览器就可以运行。

HTML 文件的结构包括头部(Head)和主体(Body)两部分。其中,头部用来描述 HTML 文件的属性信息,例如页面的类别、字符编码、刷新间隔、缓存控制、Cookie 设置等。头部的内容不会在页面中显示。HTML 的主体部分是正文,也是最主要的部分,是浏览器要显示的内容。

1.3.2 CSS

CSS(Cascading Style Sheets,级联式菜单或层叠样式表)是一种用来表现 HTML 或 XML 等文件样式的技术。

使用 CSS 可以实现页面内容和样式的分离,例如用 HTML 语言定义页面的内容,用 CSS 定义页面的样式或风格。

CSS 更高效,更灵活,维护简单方便,也更容易使整个网站的页面风格统一。

CSS 布局也可以在一个独立的样式表文件中完成,从而实现网页的表现和内容相分离。

采用 CSS 布局的页面容量比使用表格布局的页面小,页面的浏览速度更快;另外,采用 CSS 布局的页面修改和维护起来更方便。

1.3.3 JavaScript

JavaScript 是一种广泛用于客户端 Web 开发的基于对象(Object)和事件驱动(Event Driver)的脚本语言。

通过在 HTML 超文本标记语言中嵌入或调入 JavaScript 脚本,可实现在 HTML 页面中链接多个对象、与客户的交互以及客户端动态效果的应用等。

1.3.4 动态网页技术

常用的动态网页技术有 CGI、ASP、PHP 和 JSP 等。

CGI 的全称是 Common Gateway Interface,即通用网关接口。CGI 是用于 Web 服务器和外部应用之间信息交换的标准接口。CGI 的组成一般分成两个部分:一部分是 HTML 页面,即客户端浏览器上显示的页面;另一部分是运行在服务器上的 CGI 程序。当多个 CGI 程序同时执行时,服务器将启动多个进程,导致负载过重,从而影响服务器的性能。

ASP 的全称是 Active Server Pages,它是微软公司开发的一种动态网页技术。ASP 采用 JavaScript 和 VBScript 脚本语言编程,在 HTML 代码中嵌入相关的脚本代码,就可以实现相关功能。

PHP 的全称是 Personal Home Page,是一种创建动态交互性站点的强有力的服务器端脚本语言。混合了 C、Java、Perl 语法,并加入了自己的特性。由于免费,使用广泛。可搭配 Apache 作为 Web 服务器一起使用,支持 ISAPI(Internet Server Application Programming

Interface, Internet 服务器应用程序接口), 并且可以运行于 Windows 的 IIS 平台。

JSP(Java Server Pages)页面是在 HTML 页面中嵌入 JSP 元素的页面, 这些元素称为 JSP 标签。JSP 元素具有严格定义的语法并包含完成各种任务的语法元素, 比如声明变量和方法、JSP 表达式、指令和动作等。JSP 的特点是一次编写, 处处运行。字节码文件可以在具有 JVM(Java Virtual Machine, Java 虚拟机)的任何平台上运行。具有强大的可伸缩性和强大的开发工具支持。JSP 在运行时, 先转译成 Servlet, 然后再编译成 class 文件, 需要内存开销来存储 class, 需要硬盘空间存储类文件以及 class 文件。如果出现错误, 浏览器中显示的错误是 Servlet 的错误信息, 会给调试带来一定的困难。

1.3.5 Servlet

Servlet 是用 Java 编写的服务器端程序。Servlet 运行于支持 Java 的应用服务器中, 使用 Servlet API 以及相关的类编写的 Java 程序, 可以响应任何类型的请求。

Servlet 的主要功能在于交互式地浏览和修改数据, 生成动态 Web 内容。在大多数情况下 Servlet 只用来扩展基于 HTTP 协议的 Web 服务器的功能。

1.3.6 Struts

Struts 是 Apache 软件基金会(ASF)赞助的一个开源框架项目。使用 Struts 的目的是帮助我们减少运用 MVC 设计模型开发 Web 应用的时间。如果我们想混合使用 Servlet 和 JSP 的优点来建立可扩展的应用, Struts 2 是一个不错的选择。

1.4 开发环境安装与配置

这一节主要介绍 Web 开发所需的开发环境 JDK 7.0、Tomcat 7.0、MyEclipse 10.0、MySQL 5.5 等的安装与配置。虽然 Java Web 开发是和平台无关的, 但为了便于读者学习, 本书采用 Windows 平台环境, 所以, 以下采用的都是 Windows 软件包, 讲解 Windows 环境下的安装与配置过程。

1. 安装包名称及下载地址

(1) JDK 7.0

下载地址: <http://www.oracle.com/technetwork/java/javase/downloads/>。

(2) Tomcat 7.0

下载地址: <http://tomcat.apache.org/download-70.cgi>。

(3) MyEclipse 10.0

下载地址: <http://www.genuitec.com/products/myeclipse/>。

(4) MySQL 5.5

下载地址: <http://www.mysql.com/downloads/>。