

21世纪高等学校计算机教育实用规划教材

JSP Web技术及应用教程

王春明 史胜辉 主编

清华大学出版社



21世纪高等学校计算机教育实用规划教材

JSP Web技术及应用教程

王春明 史胜辉 主编

清华大学出版社
北京

内 容 简 介

本书由浅入深、循序渐进地介绍了 JSP Web 的技术原理,每个知识点都配有实例说明,并以网上书店为典型应用案例贯穿项目设计学习过程。

全书共 11 章,内容包括 Web 原理及 HTML 基础、JSP 开发及运行环境、JSP 技术基础、JSP 基本语法、JSP 内置对象、JSP 数据库应用、JavaBean 与 Servlet 技术及应用、过滤器技术、EL 表达式、自定义 JSP 标记等。最后一章还安排了两个 JSP 实际案例,用于提高和拓宽读者对 JSP 的掌握与应用,也可作为课程设计的选题参考。

本书内容理论联系实际,注重项目实践,内容安排科学合理,体系结构清晰自然,语言描述简练顺畅,满足既要有扎实的理论基础,又要达到应用型人才培养目标的教学要求。本书不仅可以作为高等院校计算机及相关专业的教材,也适合 JSP 技术开发人员参考使用。

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 13701121933

图书在版编目(CIP)数据

JSP Web 技术及应用教程/王春明,史胜辉主编.--北京:清华大学出版社,2014

21 世纪高等学校计算机教育实用规划教材

ISBN 978-7-302-36867-0

I. ①J… II. ①王… ②史… III. ①JAVA 语言—网页制作工具—高等学校—教材 IV. ①TP312
②TP393.092

中国版本图书馆 CIP 数据核字(2014)第 127072 号

责任编辑:魏江江 赵晓宁

封面设计:傅瑞学

责任校对:时翠兰

责任印制:沈 露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:23.5 字 数:574 千字

版 次:2014 年 9 月第 1 版 印 次:2014 年 9 月第 1 次印刷

印 数:1~2000

定 价:49.00 元

前 言

JSP(Java Server Pages)是由 Sun 公司倡导、许多公司一起参与建立的一种动态网页技术标准,是一种强大的服务器端动态网页开发技术。目前是在中国乃至全球最流行、应用最广泛的软件开发技术之一。JSP 与 Microsoft 公司的 ASP 技术非常相似,两者都提供在 HTML 代码中混合某种程序代码、由语言引擎解释执行程序代码的能力。

JSP 技术是 J2EE 技术的核心之一,是基于 Java Servlet 以及整个 Java 体系的 Web 开发技术,利用这一技术可以建立安全、跨平台的先进动态网站。JSP 使用的是 Java 语言,以 Java 技术为基础,又在许多方面做了改进,具有动态页面与静态页面分离、能够脱离硬件平台的束缚以及编译后运行等优点。需要强调的是,要想真正地掌握 JSP 技术,必须有较好的 Java 语言基础,以及 HTML 语言方面的知识。

本书从 JSP 基本的语法和规范入手,结合 Servlet 的最新规范,由浅入深、循序渐进地介绍了 JSP Web 的技术原理,深入浅出地讲解了 JSP 开发中的种种问题。

Web 的基本工作原理和 HTML 语言作为 JSP 技术的基础内容之一,本书在前两章对它们进行了介绍,作为读者进行普通网站设计的重要参考。对于 JSP 开发中常遇到的工程实际问题,在相关章节也做了详细介绍。

全书共 11 章。第 1 章为 Web 的基本原理,主要介绍了常用的 Web 服务器, IIS Web 服务器的配置等相关技术。第 2 章为 HTML 语言基础,主要介绍了 HTML 文件结构、常用标记和 HTML 事件,对 DIV+CSS 布局和 JavaScript 语言也做了简要介绍,为 JSP 页面设计做了基础性准备。如果学时较少,这两章可作为学生的自学内容。第 3 章是 Java Web 开发环境搭建,介绍了 JSP 的工作原理、JSP 项目的创建与发布。对于 Tomcat 服务器的安全性进行了详细讨论。第 4 章是 JSP 技术基础,详细介绍了 JSP 标准语法、JSP 指令,重点介绍了 JSP 的九大内置对象及其使用方法。第 5 章是使用 JSP 访问数据库,介绍了 JDBC 的工作原理和使用方法,对数据库连接池技术也做了详细介绍。从这一章开始引入网上书店实际工程项目的设计,随着课程的推进,逐步完成项目的设计任务。第 6 章是 JavaBean 技术,对 JavaBean 的定义与应用方法做了详细介绍。第 7 章是关于 Servlet 技术的介绍,这也是 JSP 技术的核心内容。这一章结合 Servlet 在验证码与文件上传两个典型案例中的实际应用,加深了对 Servlet 的理解。同时,在网上书店项目中也大量采用了 Servlet 进行业务逻辑处理,使读者体会到 Servlet 在项目开发中至关重要的作用。第 8 章对过滤器做了专门讨论,给出了多个过滤器实用案例。第 9 章对 EL 表达式和 JSTL 标签库做了介绍。第 10 章

介绍了 JSP 自定义标签的设计方法。第 11 章给出了聊天室和网上投票系统两个 JSP 应用项目案例,以便巩固对 JSP 技术原理的掌握。

全书由讲授 JSP 课程的教师,在总结多年教学经验和项目开发经验的基础上精心编写而成。在内容结构、知识衔接、关键知识点的讲解、典型案例的分析等方面进行了精心安排。本书采用的开发环境为 JDK 1.6+MyEclipse 7. x+Tomcat 6. x+MySQL 5. 5。

本书由王春明负责统筹编写,史胜辉、陆培军、王进、沈学华、曹利、陈亮、王则林、朱浩、张晓峰、严燕、王丹丹、魏晓宁、蒋峥峥、朱浩、朱晓辉、陈森博、丁浩、袁鸿燕、朱鹏程等在本书的编写、代码测试等方面给予了许多帮助,本书的编辑与出版还得到清华大学出版社的大力支持,在此向他们表示由衷的感谢。

本书配有完整课件和实例源代码,如有需要,请在清华大学出版社网站下载。

为了便于课程的学习交流,我们还专门开辟了“JSP Web 技术及应用教程”课程学习网站,网址为 <http://210.29.65.153:8080/jsp>。网站中提供了本书所需的全部开发工具软件、项目实例源代码和教学课件,课程的最新资料也将在该网站发布。教材中所列的程序清单均包含在相关项目文件中,读者可从本资源网站下载。

限于作者水平,加之时间仓促,书中难免有不足之处,敬请读者批评指正。

编 者

2014 年 7 月

目 录

第 1 章 Web 的基本原理	1
1.1 常用的 Web 服务器	2
1.2 IIS Web 服务器配置	3
1.3 客户端技术	5
1.4 服务器端技术	6
习题 1	9
第 2 章 HTML 语言基础	10
2.1 HTML 文件的基本结构	10
2.2 HTML 常用标记	11
2.3 HTML 事件	37
2.4 DIV+CSS 布局	40
2.4.1 CSS 样式表引入方法	41
2.4.2 CSS 语法	44
2.4.3 DIV+CSS 布局	49
2.4.4 DIV+CSS 布局定位	50
2.4.5 DIV+CSS 布局实例	55
2.5 JavaScript	60
2.5.1 JavaScript 函数	62
2.5.2 JavaScript 数据类型	63
2.5.3 JavaScript 运算符	63
2.5.4 JavaScript 中的控制语句	65
2.5.5 JavaScript 内部对象	66
2.5.6 浏览器对象	73
习题 2	82
第 3 章 Java Web 开发环境搭建	84
3.1 Java Web 工作原理	84
3.2 Tomcat 的安装配置	85
3.3 在 MyEclipse 中配置 Tomcat	87

3.4	使用 MyEclipse 创建 Web 工程	88
3.5	使用 MyEclipse 发布 Web 工程	89
3.6	Tomcat 其他常用设置	90
3.7	Servlet 容器介绍	93
3.8	HTTP 协议分析	96
	习题 3	103
第 4 章	JSP 技术基础	104
4.1	JSP 简介	104
4.2	JSP 标准语法	106
4.2.1	JSP 注释	108
4.2.2	JSP 声明	108
4.2.3	JSP 表达式	110
4.2.4	JSP 程序段	110
4.2.5	JSP 与 HTML 的混合使用	112
4.3	JSP 编译指令	113
4.3.1	page 编译指令	113
4.3.2	include 编译指令	118
4.3.3	taglib 编译指令	119
4.4	JSP 动作指令	120
4.4.1	forward 动作指令	120
4.4.2	include 动作指令	121
4.4.3	plugin 动作指令	122
4.4.4	param 动作指令	124
4.4.5	相对基准地址	125
4.5	JSP 的隐含对象	127
4.5.1	JSP 内置对象作用域	128
4.5.2	out 对象	130
4.5.3	page 对象	131
4.5.4	request 对象	132
4.5.5	response 对象	136
4.5.6	session 对象	140
4.5.7	application 对象	144
4.5.8	config 对象	146
4.5.9	exception 对象	146
4.5.10	pageContext 对象	147
4.5.11	Cookie 对象	149
	习题 4	152

第 5 章 JSP 访问数据库	155
5.1 MySQL 数据库	155
5.2 项目案例 1 网上书店数据库创建	160
5.3 使用 JDBC 访问数据库	162
5.3.1 JDBC 简介	162
5.3.2 JDBC 工作原理	162
5.3.3 常用 SQL 语句	163
5.4 JDBC 驱动类型	166
5.5 JDBC 常用接口、类介绍	167
5.6 数据库连接池原理	178
习题 5	181
第 6 章 JavaBean 技术	183
6.1 什么是 JavaBean	183
6.2 在 JSP 中使用 JavaBean	192
6.2.1 <jsp:useBean>	192
6.2.2 <jsp:setProperty>	193
6.2.3 <jsp:getProperty>	194
6.3 项目案例 2 网上书店用户登录设计	197
习题 6	201
第 7 章 Servlet 基础知识	203
7.1 Servlet 概念及设计步骤	203
7.1.1 Servlet 基本概念	203
7.1.2 Servlet 设计步骤	204
7.2 Servlet 的生命周期	209
7.3 Servlet API 层次结构	211
7.4 主要 Servlet API 介绍	212
7.4.1 HttpServlet 类	212
7.4.2 HttpServletRequest 接口	214
7.4.3 HttpServletResponse 接口	215
7.4.4 ServletContext 接口	216
7.4.5 HttpSession 接口	219
7.4.6 ServletConfig 类	220
7.5 Servlet 应用举例	220
7.5.1 利用 Servlet 实现验证码功能	220
7.5.2 利用 Servlet 实现文件上传功能	224
7.6 项目案例 3 网上书店后台设计	232

7.6.1	图书实体类设计	232
7.6.2	数据库底层操作业务类设计	233
7.6.3	逻辑处理业务类(Servlet)设计	236
7.6.4	后台功能模块设计	243
7.7	项目案例 4 网上书店前台设计	250
7.7.1	顾客浏览图书	253
7.7.2	显示图书详细信息	253
7.7.3	图书添加到购物车并显示购物车信息	256
7.7.4	添加订单信息并结账	260
7.8	JSP 设计模式	264
7.8.1	Model I 体系结构	264
7.8.2	Model II 体系结构	265
	习题 7	266
第 8 章	过滤器	268
8.1	Servlet 过滤器简介	268
8.2	Servlet 过滤器体系结构	269
8.3	Servlet 过滤器实例	269
8.4	JSP 中文乱码问题	275
	习题 8	278
第 9 章	EL 与 JSTL	279
9.1	EL 表达式基础知识	279
9.2	EL 表达式的应用示例	286
9.3	JSTL 简介	292
9.4	JSTL 核心标签库	293
	习题 9	302
第 10 章	JSP 自定义标签	303
10.1	JSP 自定义标签简介	303
10.2	开发自定义 JSP 标签	304
10.2.1	创建标签处理类	304
10.2.2	创建标签库描述文件 TLD	306
10.2.3	JSP 中使用自定义标签	307
10.3	自定义分页标签示例	307
	习题 10	314
第 11 章	JSP Web 项目实例	315
11.1	聊天室程序设计实例	315

11.1.1 聊天室基础	316
11.1.2 聊天室窗口框架	324
11.1.3 聊天信息处理与退出机制	328
11.1.4 聊天室程序小结	333
11.2 投票系统设计实例	334
习题 11	343
附录 A HTML 常用标记	344
附录 B CSS 属性一览表	349
附录 C JSP 样卷	352
附录 D 课程设计选题参考	358
参考文献	364

第 1 章

Web 的基本原理

本章学习目标

- 掌握 Web 的工作原理；
- 了解常用的 Web 服务器；
- 熟练掌握 IIS Web 服务器的配置与使用。

Web 的应用架构是由英国人 Tim Berners-Lee 在 1989 年提出的,1990 年 11 月第一个 Web 服务器开始运行,由 Tim Berners-Lee 编写的图形化 Web 浏览器第一次出现在人们面前。目前,与 Web 相关的各种技术标准都由著名的 W3C 组织管理和维护。

Web 的原文含义是编织网,自从互联网出现以后,Web 变成了 WWW(World Wide Web)的简称,也称为万维网。Web 的基本工作原理是请求与响应原理,它由遍布在互联网中的 Web 服务器和安装了 Web 浏览器的计算机组成,用于发布、浏览、查询网络信息。它是一种基于超文本方式工作的信息系统,作为一个能够处理文字、图像、声音、视频等多媒体信息的综合系统,它提供了丰富的信息资源,这些信息资源以 Web 页面的形式分别存放在各个 Web 服务器上,用户可以通过浏览器向服务器发出资源请求,服务器对请求作出处理和响应,将响应结果发给客户端,由浏览器解析显示所请求的结果信息。它的发展同时也推动了网络的不断发展,从而影响到社会及个人的生活。

Web 是一种典型的分布式应用架构,从技术层面上看,Web 架构的精华包括三方面:用超文本标记语言(Hypertext Markup Language,HTML)技术实现信息文档的表示;用统一资源定位(Uniform Resource Locator,URL)技术实现全球信息的精确定位;用超文本传输协议(HyperText Transfer Protocol,HTTP)实现分布式的信息传输。

Web 应用中的每一次信息交换都要涉及客户端和服务端两个层面,如图 1-1 所示。因此,Web 开发涉及的技术大体上也可以被分为客户端技术和服务器端技术两大类。客户端主要是安装了浏览器的计算机;服务器端通常是保存网页等资源并提供 Web 服务的远程服务器。



图 1-1 Web 技术架构

1.1 常用的 Web 服务器

Windows NT/2000/2003 平台下最常用的服务器是微软公司的 IIS。UNIX 和 Linux 平台下常用的服务器有 Apache、Tomcat、IBM WebSphere、Nginx、Lighttpd 等,其中应用最广泛的是 Apache。

1. Microsoft IIS

微软公司的 Web 服务器产品为 IIS (Internet Information Server)。IIS 是由微软公司提供的基于 Microsoft Windows 的互联网基本服务,最初是 Windows NT 版本的可选包,随后内置在 Windows 2000、Windows XP Professional 和 Windows Server 2003 中一起发行。它是目前最流行的 Web 服务器产品,很多著名的网站都是建立在 IIS 平台上的。IIS Web 服务组件包括 Web 服务器、FTP 服务器、NNTP 服务器和 SMTP 服务器,分别用于网页浏览、文件传输、新闻服务和邮件发送等方面,它使得在网络上发布信息成了一件很容易的事。

2. Tomcat 服务器

Tomcat 服务器是一个免费的开放源代码的 Web 应用服务器。Tomcat 是 Apache 软件基金会的 Jakarta 项目中的一个核心项目,由 Apache、Sun 和其他一些公司及个人共同开发而成。由于有了 Sun 的参与和支持,最新的 Servlet 和 JSP 规范总是能在 Tomcat 中得到体现。因为 Tomcat 技术先进、性能稳定,而且免费,因而深受 Java 爱好者的喜爱并得到了部分软件开发商的认可,成为目前最为流行的 Web 应用服务器之一。

3. Apache 服务器

Apache 是世界使用排名第一的 Web 服务器软件,可以运行在几乎所有广泛使用的计算机平台上。Apache 的模块支持非常丰富,虽然在速度、性能上不及其他轻量级 Web 服务器,属于重量级产品,所消耗的内存也比其他 Web 服务器要高,但由于其跨平台和安全性而被广泛使用,是最流行的 Web 服务器端软件之一。Apache 取自“a patchy server”的读音,意思是充满补丁的服务器,因为它是自由软件,所以不断有人来为它开发新的功能、新的特性,修改原来的缺陷。Apache 的特点是简单、速度快、性能稳定,并可做代理服务器来使用。

4. IBM WebSphere 服务器

WebSphere Application Server 是一个基于 Java 的 Web 应用程序服务器,提供了丰富的电子商务应用程序部署环境,带有一整套应用程序服务,包括事务管理、安全性、群集、性能、可用性、连接性和可伸缩性功能。它构建在开放标准的基础之上,能帮助用户部署与管理从简单的 Web 站点到强大的电子商务解决方案的诸多应用程序。它遵循 J2EE 并为 Java 组件、XML 和 Web 服务提供了一个可移植的 Web 部署平台,这个平台能够与数据库交互并提供动态 Web 内容。

5. Lighttpd 服务器

Lighttpd 是一个由德国人写的开源软件,其目标是提供一个专门针对高性能网站,安全、快速、兼容性好并且灵活的轻量级 Web Server 环境。它具有内存开销低、CPU 占用率低、效能好,以及模块丰富等特点,支持 FastCGI、CGI、Auth、输出压缩(Output Compress)、URL 重写及 Alias 等重要功能。

1.2 IIS Web 服务器配置

为了便于 Web 技术的入门,首先介绍 IIS Web 服务器的配置与使用方法,第 2 章中介绍的 HTML 语法及示例都可以在 IIS Web 服务器中发布运行。这些也为后面的 JSP Web 技术的学习打下基础。

IIS Web 服务器配置步骤如下。

1. 确认计算机上是否已安装 IIS。

可先试试步骤 3,如果步骤 3 中的文件夹属性窗口中没有“Web 共享”选项卡,表示该机器未装 IIS。

如果机器未安装 IIS 服务组件,可通过下列步骤安装 IIS。

“控制面板”→“添加或删除程序”→“添加/删除 Windows 组件”,在“Internet 信息服务(IIS)”复选框前打钩,如图 1-2 所示。

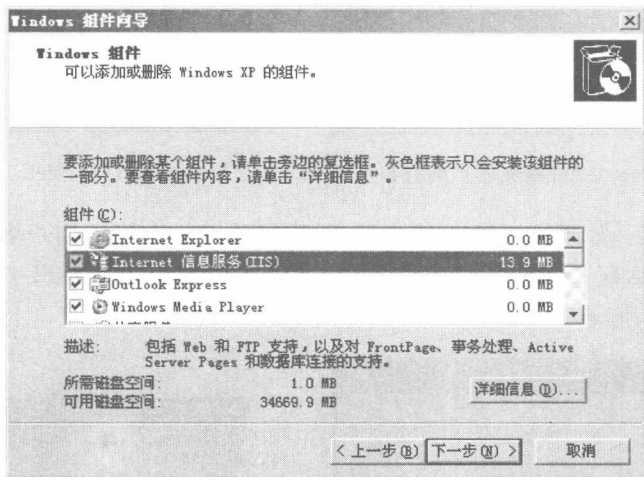


图 1-2 Internet 信息服务(IIS)打钩

单击“详细信息”按钮,在“FrontPage 2000 服务器扩展”复选框前打钩,单击“确定”按钮,如图 1-3 所示。安装过程中需要多次浏览之前已下载的 IIS 组件目录,以便安装程序能够找到所需要的安装文件。

2. 编写 Web 网页程序

这是第一个 IIS Web 例子,该网页作为 IIS Web 服务器资源,浏览器可以向 IIS Web 服务器请求打开这个 Web 页面。

在计算机 D:\myweb 文件夹下,创建 index.html 网页文件。

```
<html>
<head>
  <title>我的第一个网页</title>
  <bgsound src = 111.mp3 loop = 10> <!-- 背景音乐 -->
</head>
<body bgcolor = yellow>
```

```

        < font size = 10 color = "# FF0000">欢迎进入精彩的 Web 世界</font>
    </body>
</html>

```

3. Web 发布

右击 D:\myweb 示例文件夹,在弹出的菜单中选择“属性”命令。在打开的对话框中,切换至“Web 共享”选项卡,选择“共享文件夹”单选按钮,即可将该文件夹进行 IIS Web 发布,如图 1-4 和图 1-5 所示。

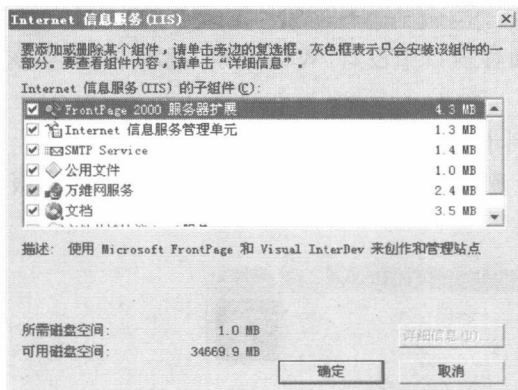


图 1-3 FrontPage 2000 服务器扩展打钩



图 1-4 Web 共享设置

IIS Web 发布后,如果未设置匿名访问,则访问站点时需输入用户名和密码,即会出现如图 1-6 所示的对话框。实际上对于一个站点访问应该不需要输入用户名和密码,站点访问都应该是匿名的,应该开放匿名账户的访问权限。

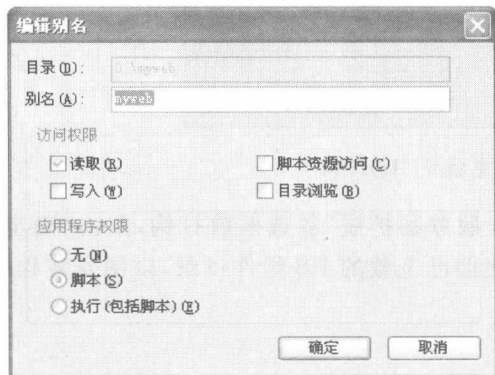


图 1-5 Web 共享目录设置

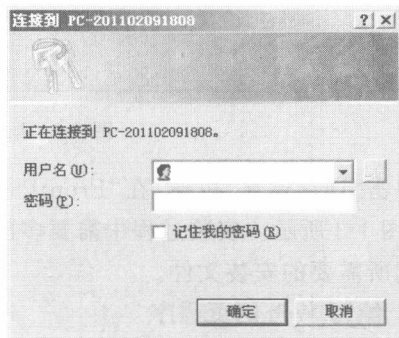


图 1-6 未设置匿名访问时需输入用户名和密码

开放匿名账户的访问权限的方法是：“控制面板”→“管理工具”→“Internet 信息服务”→“网站/默认站点”，在站点名称 myweb 上右击,在弹出的菜单中选择“属性”项,弹出“myweb 属性”对话框,打开“目录安全性”选项卡,编辑“匿名访问和身份验证控制”项。在“身份验证方法”对话框的“匿名访问”复选框中选中即可,如图 1-7 所示。

4. 通过浏览器访问 Web 站点

打开浏览器,在地址栏输入 <http://localhost/myweb/index.html>。

运行效果如图 1-8 所示。

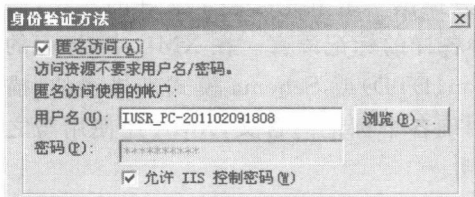


图 1-7 开放匿名账户的访问权限

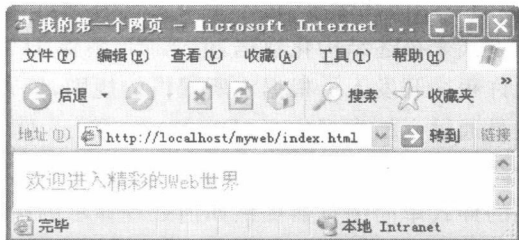


图 1-8 第一个 Web 程序运行效果

浏览器地址栏中的 `http://localhost/myweb/index.html` 表示向服务器资源发出请求,请求信息含义为:采用 http 协议,向 localhost(本地)服务器发出请求,请求的资源位于服务器中的 myweb 站点中,请求的资源名称为 index.html。可见,Web 服务器上存放的都是各种资源文件,供客户端请求访问。这些资源文件类型有静态 HTML 页面、动态页面、图片、音视频等。对于不同类别的 Web 服务器,其支持的资源类型区别较大。服务器对浏览器发来的各种请求作出处理和响应,将响应结果以 HTML 形式回送给客户端浏览器,浏览器将收到的 HTML 文件解析后显示给用户。

1.3 客户端技术

无论采用何种 Web 技术,都涉及客户端技术和服务器端技术。

Web 客户端的主要任务是展现信息内容,HTML 语言是信息展现的最有效载体之一。最初的 HTML 语言只能在浏览器中展现静态的文本或图像信息,随后由静态技术向动态技术逐步转变。客户端技术用于为最终用户构造一个友好的人机界面。

从 Internet 诞生开始,客户端技术就在不断地发展,从最早的 HTML 到 DHTML,再到目前最有发展前途的 XML 技术。

1. HTML

HTML 是浏览器识别的语言代码,可以使浏览器显示任何信息(文本、表格和图像等)。HTML 代码生成的是一种静态的页面,其优点是不用经过其他处理,而且可以被浏览器或代理服务器存储在缓存中,所以 HTML 页面请求的速度比较快。HTML 代码可以通过一些网页编辑软件(如 FrontPage、Dreamweaver 等),以所见即所得的方式生成和编辑,便于维护和修改。

2. DHTML

DHTML 是对 HTML 扩充的一种动态超文本标记语言。在 DHTML 中,HTML 页面上的所有元素都被当作对象来处理,它们有自己的属性和事件。对它们的控制是通过改变它们的属性和触发它们的某些事件来实现的,所有这些对象共同构成了 DOM(Document Object Model,文档对象模型)。DHTML 为 Web 应用提供了一种动态机制和一些简单的操作,可以在一定程度上减轻服务器的负荷,大大缩短响应的时间。

3. XML

XML(Extensible Markup Language)是可扩展标记语言,是由 W3C 组织给出的一种可

扩展的源标记语言。它是 SGML(Standard General Markup Language, 标准化通用标记语言)的一个简化子集,这个子集是专为 Web 环境设计的。XML 通过在数据中加入附加信息的方式来描述结构化数据,但 XML 不像 HTML 只提供一组事先已经定义好的标记,而是允许程序开发人员根据它所提供的规则,编写各种各样的标记语言。在 XML 中,标记的语法是通过文档类型定义(Document Type Definition, DTD)或 Schema 模式来描述的,描述什么是有有效的标记,定义标记语言的结构。为了明确各个标记的含义,XML 还使用与之相连的样式单向浏览器提供如何处理显示的指示说明。

1995 年,Java 语言的问世也给 Web 技术的发展带来了更大的变革。Java 语言天生就具备的平台无关的特点,让人们一下子找到了在浏览器中开发动态应用的捷径。CSS 和 DHTML 技术真正让 HTML 页面又酷又炫、动感无限起来。1997 年,微软公司将动态 HTML 标记、CSS 和动态对象模型发展成了一套完整、实用、高效的客户端开发技术体系,微软公司称其为 DHTML。同样是实现 HTML 页面的动态效果,DHTML 技术无须启动 Java 虚拟机或其他脚本环境,可以在浏览器的支持下,获得更好的展现效果和更高的执行效率。

1.4 服务器端技术

与客户端技术从静态向动态的演进过程类似,Web 服务端的开发技术也是由静态向动态逐渐发展、完善起来的。

最早的 Web 服务器简单地响应浏览器发来的 HTTP 请求,并将存储在服务器上的 HTML 文件返回给浏览器。

第一种真正使服务器能根据运行时的具体情况,动态生成 HTML 页面的技术是大名鼎鼎的 CGI(Common Gateway Interface)技术。CGI 技术允许服务端的应用程序根据客户端的请求,动态生成 HTML 页面,这使客户端和服务端的动态信息交换成为了可能。

早期的 CGI 程序大多是编译后的可执行程序,其编程语言可以是 C、C++、Pascal 等任何通用的程序设计语言。为了简化 CGI 程序的修改、编译和发布过程,人们开始探寻用脚本语言实现 CGI 应用的可行方式。

1994 年,发明了专用于 Web 服务端编程的 PHP 语言。PHP 语言是一种在服务器端执行的嵌入 HTML 文档的超文本预处理脚本语言,语言的风格类似于 C 语言,直到目前仍被广泛地运用。PHP 语言将 HTML 代码和 PHP 指令合成为完整的服务端动态页面,可以用一种更加简便、快捷的方式实现动态 Web 功能。1996 年,微软公司在其 Web 服务器 IIS 3.0 中引入了 ASP 技术。ASP 使用的脚本语言是 VBScript 和 JavaScript。1998 年,JSP 技术诞生。

随后,XML 语言及相关技术又成为主流。XML 语言对信息的格式和表达方法做了最大程度的规范,应用软件可以按照统一的方式处理所有 XML 信息。这样一来,信息在整个 Web 世界里的共享和交换就有了技术上的保障。

HTML 提供了控制超文本格式的信息,关心的是信息的表现形式,利用这些信息可以在用户的屏幕上显示出特定设计风格的 Web 页面。XML 语言关心的是信息本身的格式和数据内容。

Web 服务器使用 HTTP 超文本传输协议,将 HTML 文档从 Web 服务器传输到用户的 Web 浏览器上。

Web 技术的发展主要分为三个阶段,大体总结为静态技术阶段、动态技术阶段和 Web 2.0 新时期。

1. 静态 Web 页面

Web 技术发展的第一阶段——静态文档,主要是用于静态 Web 页面的浏览。用户使用客户端的 Web 浏览器,可以访问 Internet 上各个 Web 站点,在每一个站点上都有一个主页作为进入 Web 站点的入口。每一 Web 页中都可以含有信息及超文本链接,超文本链接可以带用户到另一 Web 站点或是其他的 Web 页。从服务器端来看,每一个 Web 站点由一台主机、Web 服务器及许多 Web 页所组成,以一个主页为首,其他的 Web 页为支点,形成一个树状的结构。每一个 Web 页都是以 HTML 的格式编写的。

这一阶段,Web 服务器基本上只是一个 HTTP 的服务器,负责客户端浏览器的访问请求、建立连接、响应用户的请求、查找所需的静态的 Web 页面,最后再将响应结果返回到客户端。

随着互联网技术的不断发展以及网上信息呈几何级数的增加,人们逐渐发现手工编写包含所有信息和内容的页面,对人力和物力都是一种极大的浪费,而且几乎变得难以实现。此外,采用静态页面方式建立起来的站点只能简单地根据用户的请求传送现有页面,而无法实现各种动态的交互功能。具体来说,静态页面在以下几个方面都存在明显的不足:

(1) 无法支持后台数据库。随着网上信息量的增加,以及企业和个人希望通过网络发布产品和信息的需求的增强,人们越来越需要一种能够通过简单的 Web 页面访问服务端后台数据库的方式。这是静态页面不能实现的。

(2) 无法有效地对站点信息进行及时的更新。用户如果需要对传统静态页面的内容和信息进行更新或修改的话,只能采用逐一更改每个页面的方式。在互联网发展初期网上信息较少的时代,这种做法还是可以接受的。但是现在即使是网友们的个人站点也包含着各种各样的丰富内容,因此及时、有效地更新页面信息也成为亟待解决的问题。

(3) 无法实现动态显示效果。所有的静态页面都是事先编写的,是一成不变的,因此访问同一页面的用户看到的都是相同的内容,静态页面无法根据不同的用户做不同的页面显示。

而这些不足之处,促使 Web 技术进入了发展的第二阶段。

2. 动态 Web 技术

为了克服静态页面的不足,人们将传统单机环境下的编程技术引入互联网络,与 Web 技术相结合,从而形成新的网络编程技术。网络编程技术通过在传统的静态页面中加入各种程序和逻辑控制,在网络的客户端和服务端实现了动态和个性化的交流与互动。人们将这种使用网络编程技术创建的页面称为动态页面。

动态网页与静态网页是相对应的,动态网页文件名不仅可以是 htm、html、shtml、xml 等静态网页的常见形式,还可以 jsp、asp、php、perl、cgi 等为后缀。这里说的动态网页,与网页上的各种动画、滚动字幕等视觉上的“动态效果”没有直接关系。无论网页是否具有视觉上的动态效果,采用动态网站技术生成的网页都称为动态网页。动态网页既可以是纯文字的内容,也可以是包含各种动画的内容。