

普通高等教育软件工程专业“十二五”规划教材

JSP程序设计

主编 于 彬

 科学出版社

普通高等教育软件工程专业“十二五”规划教材

JSP 程序设计

主编 于 彬

副主编 陈 玮 张志峰 张 丽

科学出版社

北 京

内 容 简 介

本书遵循以语言为基础、以应用为主导的编写原则，理论联系实际，并通过大量的实例，循序渐进地介绍了有关 JSP 开发所涉及的各类知识。

全书共 11 章，首先从最基础的 JSP 程序设计基础知识和 JSP 开发环境的搭建开始，然后介绍 JSP 基础、内置对象、和 Servlet 结合、和 JavaBean 结合、JSTL 和 AJAX 技术等，最后通过案例开发讲解 JSP 知识的综合应用。本书对基础知识介绍详细，理论联系实际，具有很强的操作性。书中还提供了大量通过测试的、可运行的完整实例代码，这些实例都有相应的设计步骤、代码详解、程序运行结果等，使读者能够快速掌握和运用 JSP 编程技巧。

本书可作为大学计算机软件及相关专业的 Web 编程课程教材，也可作为技术支持人员和程序开发人员的参考书。

图书在版编目(CIP)数据

JSP 程序设计 / 于彬主编. —北京: 科学出版社, 2013
普通高等教育软件工程专业“十二五”规划教材
ISBN 978-7-03-037564-3

I. ①J… II. ①于… III. ①Java 语言—网页制作工具 IV. ①TP312
②TP393.092

中国版本图书馆 CIP 数据核字(2013)第 110103 号

责任编辑: 于海云 / 责任校对: 李 影
责任印制: 闫 磊 / 封面设计: 迷底书装

科 学 出 版 社 出 版

北京东黄城根北街 16 号
邮政编码: 100717
<http://www.sciencep.com>

保定市中华美凯印刷有限公司 印刷

科学出版社发行 各地新华书店经销

*

2013 年 6 月第 一 版 开本: 787×1092 1/16

2013 年 6 月第一次印刷 印张: 20

字数: 506 000

定价: 45.00 元

(如有印装质量问题, 我社负责调换)

普通高等教育软件工程专业“十二五”规划教材

编 委 会

主任委员

李占波 郑州大学软件技术学院副院长

副主任委员

车战斌 中原工学院软件学院院长
刘黎明 南阳理工学院软件学院院长
刘建华 华北水利水电大学软件学院院长
乔保军 河南大学软件学院副院长

委 员

高 岩 河南理工大学计算机科学与技术学院副院长
邓璐娟 郑州轻工业学院软件学院副院长
史玉珍 平顶山学院软件学院副院长
周文刚 周口师范学院计算机科学与技术学院副院长
席 磊 河南农业大学信息与管理科学学院系主任
陈建辉 郑州航空工业管理学院计算机科学与技术系副主任
张永强 河南财经政法大学计算机与信息工程学院副院长
郑延斌 河南师范大学计算机与信息工程学院副院长
谭营军 河南职业技术学院信息工程系副主任
赵素萍 洛阳师范学院信息技术学院软件工程系主任
潘 红 新乡学院计算机与信息工程学院院长

《JSP 程序设计》编委会

主 编 于 彬

副主编 陈 玮 张志峰 张 丽

参 编 王 倩 熊蜀峰 宋振方 屠晓云

前 言

本书是为了区别于传统的教材而编写的适应软件工程专业高等教育、培养应用型本科专业人才需要的教材。在编写方案制定上力求体现教育部的教育教学改革精神，树立以素质教育为基础、能力为本位、就业为导向的“以项目为驱动的教学模式”改革新观念。教材编写紧紧围绕高等院校的软件工程专业培养目标，突出应用型本科人才培养模式；在内容体系上打破科学性，突出实用性和技能性；理论部分坚持以够用为度，以应用为主旨，强调实践、实训教学环节，培养学生的综合实践能力。

本书的读者应具有一定的 Java 语言基础。通过本书的学习，学生能够掌握 Java Web 开发的基本知识和主要技术。本书各章节的例子都是从简单到复杂，逐步深入。叙述上追求简单直观，通俗易懂，便于读者学习掌握 JSP 技术。

全书共 11 章。第 1 章介绍了 JSP 程序设计基础知识；第 2 章介绍了 Java Web 环境配置等内容；第 3 章主要介绍 JSP 基础知识；第 4 章详细地介绍了 JSP 的内置对象；第 5 章介绍了 Servlet 技术，该技术是非常重要的—种组件技术，能极大地发挥 Java 语言的优点，Servlet 被称为 Java 组件技术的核心；第 6 章介绍了 JavaBean 组件技术以及 JavaBean 在 Java Web 开发中的应用；第 7 章介绍了在 JSP 页面中如何实现对文件的操作；第 8 章介绍了 JDBC 技术；第 9 章介绍了 JSTL 技术；第 10 章介绍了 AJAX 异步交互技术；第 11 章通过两个比较典型的实例，综合运用相关 Java Web 技术，介绍了如何使用 JSP+JavaBean 模式和 JSP+JavaBean+Servlet 模式来开发具体的 Web 应用系统。

本书配备有教学课件、教学大纲、教学日历、代码、模拟考试试题，以及课后习题参考答案。

本书由于彬担任主编。其中第 1、10 章由河南农业大学张丽编写，第 2、7 章由南阳理工学院软件学院于彬编写，第 3、9 章由河南理工大学计算机学院陈玮编写，第 4 章由河南大学软件学院王倩编写，第 5 章由平顶山学院软件学院熊蜀峰编写，第 6 章由洛阳师范学院信息技术学院宋振方编写，第 8 章由郑州轻工业学院软件学院张志锋编写，第 11 章由南阳理工学院软件学院屠晓云编写。

最后感谢科学出版社、郑州轻工业学院、河南理工大学、河南农业大学、河南大学、平顶山学院、洛阳师范学院、南阳理工学院给予的大力支持。

由于时间仓促，加之编者水平有限，书中难免存在不足之处，恳请读者批评指正。

编 者

2013 年 5 月

目 录

前言

第 1 章 JSP 程序设计基础	1
1.1 Web 应用程序	1
1.1.1 Web 技术的由来与发展	1
1.1.2 Web 动态网页技术	3
1.1.3 Web 应用程序的工作原理	4
1.2 JSP 技术概述	5
1.3 JSP 的工作原理	6
1.4 JSP 程序开发模式	7
1.4.1 JSP 两种体系结构	7
1.4.2 JSP 开发 Java Web 站点的主要方式	8
1.5 小结	8
1.6 习题	9
第 2 章 JSP 开发环境搭建	10
2.1 JSP 的运行环境	10
2.1.1 JSP 的运行环境	10
2.1.2 JDK 的安装与配置	11
2.1.3 Tomcat 的安装与启动	15
2.1.4 JSP 开发工具——Eclipse	19
2.2 Java Web 项目的建立与部署	22
2.2.1 Java Web 项目的建立	22
2.2.2 创建一个 JSP	25
2.2.3 创建 Servlet	27
2.2.4 Java Web 项目的部署	31
2.3 BBS 案例的分析与设计	32
2.3.1 系统分析	32
2.3.2 系统设计	33
2.4 小结	36
第 3 章 JSP 基础	37
3.1 JSP 简介	37
3.1.1 JSP 页面的基本结构	37
3.1.2 JSP 的运行原理	38
3.2 JSP 语法	42

3.2.1	变量和方法的声明	42
3.2.2	JSP 表达式	43
3.2.3	Java 程序片 (scriptlet)	44
3.2.4	JSP 注释	45
3.3	JSP 指令标签	47
3.3.1	page 指令	47
3.3.2	include 指令	49
3.4	JSP 动作标签	51
3.4.1	jsp:include 动作标签	51
3.4.2	jsp:forward 动作标签	51
3.4.3	param 动作标签	52
3.4.4	useBean 动作标签	54
3.5	小结	56
3.6	习题	56
第 4 章	JSP 内建对象	57
4.1	out 对象	57
4.2	request 对象	60
4.2.1	获取 HTML 表单提交的数据	61
4.2.2	汉字信息处理	66
4.2.3	常用方法举例	67
4.3	response 对象	70
4.3.1	动态响应 contentType 属性	70
4.3.2	response 实现网页的自动刷新	71
4.3.3	response 重定向	72
4.3.4	response 的状态行	72
4.4	session 对象	73
4.4.1	session 对象的 ID	74
4.4.2	session 对象与 URL 重写	75
4.4.3	session 对象的常用方法	77
4.5	application 对象	80
4.5.1	application 对象的常用方法	80
4.5.2	用 application 对象制作留言板	81
4.6	小结	84
4.7	习题	84
第 5 章	Servlet	85
5.1	Servlet 概述	85
5.1.1	Servlet 简介和优点	85
5.1.2	Servlet 与 JSP 的关系	86
5.1.3	JSP 文件编译过程	86

5.1.4	HTTP 基础知识	90
5.2	Servlet 的编译和运行	92
5.2.1	一个简单的 Servlet 例子	92
5.2.2	存放 Servlet 的目录	93
5.2.3	运行 Servlet	94
5.3	Servlet 的体系结构	95
5.3.1	一个基本 Servlet 程序的组成	95
5.3.2	Servlet 应用程序体系结构	96
5.3.3	Servlet 层次结构	96
5.4	Servlet 的生命周期	97
5.4.1	Servlet 的生命周期	97
5.4.2	Servlet 的基本方法	99
5.5	JSP 和 Servlet 的交互	101
5.5.1	通过表单向 Servlet 提交数据	101
5.5.2	从 Servlet 到 JSP 的信息传递	106
5.6	Servlet 的高级应用	109
5.6.1	Servlet 的初始化参数	109
5.6.2	过滤器	112
5.6.3	监听器	117
5.7	小结	122
5.8	习题	122
第 6 章	JavaBean 组件技术	124
6.1	JavaBean 概述	124
6.2	JavaBean 的编写和使用	124
6.2.1	编写 JavaBean	124
6.2.2	使用 JavaBean	126
6.2.3	设置 JavaBean 属性	127
6.2.4	获得 JavaBean 属性	134
6.2.5	设置 JavaBean 的范围	136
6.3	Java Web 开发模型	147
6.3.1	JSP 和 JavaBean 开发模型	147
6.3.2	JSP+JavaBean+Servlet 开发模型	151
6.3.3	两种开发模型比较	156
6.4	分页 JavaBean 开发	156
6.5	小结	166
6.6	习题	166
第 7 章	JSP 中的文件操作	167
7.1	File 类和数据流	167
7.1.1	File 类	167

7.1.2	字节输入流类和字节输出流类	171
7.2	文件上传	173
7.2.1	JSP 页面处理文件上传	173
7.2.2	Servlet 处理文件上传	175
7.3	文件下载	178
7.4	小结	180
7.5	习题	180
第 8 章	JDBC 与数据库访问	181
8.1	JDBC 基础知识	181
8.1.1	JDBC 简介	181
8.1.2	DriverManager	184
8.1.3	Connection	185
8.1.4	Statement	186
8.1.5	ResultSet	187
8.2	使用 JDBC 访问数据库	187
8.2.1	使用 JDBC 访问数据库的一般步骤	187
8.2.2	使用 JDBC 驱动访问 MySQL 数据库	189
8.2.3	访问 Microsoft SQL Server 2000 数据库及其应用实例	193
8.2.4	访问 Microsoft SQL Server 2008 数据库及其应用实例	197
8.3	数据库的增、删、改、查操作	203
8.3.1	数据库的增、删、改、查操作	203
8.3.2	基于 MVC 模式的学生信息管理系统	205
8.4	JSP 在数据库应用中的其他相关问题	227
8.4.1	分页技术	227
8.4.2	常见中文乱码处理方式	229
8.5	小结	231
8.6	习题	231
第 9 章	JSP 标准标签库	232
9.1	JSP 标准标签库简介	232
9.1.1	概述	232
9.1.2	JSTL 的使用	232
9.2	核心标签库	233
9.3	国际化标签库	237
9.4	数据库标签库	241
9.5	XML 标签库	244
9.6	函数标签库	248
9.7	小结	251
9.8	习题	251

第 10 章	AJAX	252
10.1	AJAX 简介	252
10.1.1	AJAX 包含的技术	252
10.1.2	AJAX 的运行原理	253
10.2	AJAX 开发框架	256
10.3	AJAX 应用	259
10.3.1	基于 JSP 的 AJAX	259
10.3.2	基于 Servlet 的 AJAX	265
10.3.3	AJAX 的应用场景	273
10.4	小结	274
10.5	习题	274
第 11 章	综合案例	275
11.1	博客网站	275
11.1.1	系统功能	275
11.1.2	数据库设计	275
11.1.3	系统主要功能的实现	276
11.2	网上书店	278
11.2.1	系统功能	278
11.2.2	数据库设计	278
11.2.3	系统的关键技术	279
11.2.4	各个页面的设计	281
11.3	小结	307
11.4	习题	307
参考文献		308

第 1 章 JSP 程序设计基础

当今对 Web 技术的强大需求推进着各种 Web 技术的应运而生。本章主要讲解 Web 技术的相关概念与原理，主要内容包括：Web 应用程序、JSP 技术概述、JSP 工作原理、JSP 程序开发模式。

1.1 Web 应用程序

随着信息化时代的到来，我们对网络的依赖越来越多，我们从网络上获取许多的信息资源。作为信息传送的主题，Web 受到越来越多人的青睐。

1.1.1 Web 技术的由来与发展

Web (World Wide Web, 简称 WWW 或者 Web) 是由蒂姆·伯纳斯-李 (Tim Berners-Lee, 万维网之父, 1955 年出生于英国, 不列颠帝国勋章获得者, 英国皇家学会会员, 英国皇家工程师学会会员, 美国国家科学院院士) 于 1989 年 3 月提出的万维网设想而发展起来的。1990 年 12 月 25 日, 他在日内瓦的欧洲粒子物理实验室里开发出了世界上第一个网页浏览器。他是关注万维网发展的万维网联盟的创始人, 并获得世界多国授予的各种荣誉。他最杰出的成就是免费把万维网的构想推广到全世界, 让万维网科技获得迅速的发展, 并极大地改变了人类的生活方式。

国际互联网 Internet 在 20 世纪 60 年代就诞生了, 为什么没有迅速流传开来呢? 其实, 很重要的原因是因为连接到 Internet 需要经过一系列复杂的操作, 网络的权限也很分明, 而且网上内容的表现形式极其单调枯燥。Web 通过一种超文本方式把网络上不同计算机内的信息有机地结合在一起, 并且可以通过超文本传输协议 (HTTP) 从一台 Web 服务器转到另一台 Web 服务器上检索信息。Web 服务器能发布图文并茂的信息, 在软件支持下还可以发布音频和视频信息。此外, Internet 的许多其他功能, 如 E-mail、Telnet、FTP 等都可通过 Web 实现。美国著名的信息专家尼葛洛庞帝教授认为: 1989 年是 Internet 历史上划时代的分水岭。Web 技术确实给 Internet 赋予了强大的生命力, Web 浏览的方式给互联网打开了新局面。

Web 的前身是 1980 年由蒂姆·伯纳斯-李负责的一个项目演变而来的。1990 年第一个 Web 服务器开始运行。1991 年, CERN (Conseil Européen pour la Recherche Nucléaire, 欧洲核子研究组织) 正式发布了 Web 技术标准。W3C (World Wide Web Consortium, 万维网联盟或者 W3C 理事会) 于 1994 年 10 月由蒂姆·伯纳斯-李在麻省理工学院计算机科学实验室成立, 负责组织、管理和维护 Web 相关的各种技术标准, 目前的 Web 版本是 Web 3.0。

早期的 Web 应用主要是使用 HTML 语言编写、运行在服务器端的静态页面。用户通过浏览器向服务器端的 Web 页面发出请求, 服务器端的 Web 应用程序接收到用户发送的请求后, 读取地址所标识的资源, 加上消息报头把用户访问的 HTML 页面发送给客户端的浏览器。

HTML (Hypertext Markup Language, 超文本标记语言) 是一种描述文档结构的语言, 不能描述实际的表现形式。HTML 的历史最早可以追溯到 1945 年。1945 年, 范内瓦·布什

(Vannevar Bush) 提出了文本和文本之间通过超级链接相互关联的思想, 并给出设计方案。范内瓦·布什是具有 6 个不同学位的科学家、教育家和政府官员, 与 20 世纪许多著名的事件都有关系, 如组织和领导了制造第一颗原子弹的著名的“曼哈顿计划”、氢弹的发明、登月飞行、“星球大战计划”等。正如历史学家迈克尔·雪利所言: “要理解比尔·盖茨和比尔·克林顿的世界, 你必须首先认识范内瓦·布什。”正是因其在信息技术领域多方面的贡献和超人远见, 范内瓦·布什获得了“信息时代的教父”的美誉。1960 年这种信息关联技术被正式命名为超文本技术。从 1991 年 HTML 语言正式诞生以来推出了多个不同的版本, 其中对 Web 技术发展具有重大影响的主要有两个版本: 1996 年推出的 HTML 3.2 和 1998 年推出的 HTML 4.0。1999 年 W3C 颁布了 HTML 4.0.1。目前大多数 Web 服务器和浏览器等相关软件均支持 HTML 4.0.1 标准。HTML V5 版本将拥有更大的应用空间。

但是让 HTML 页面丰富多彩、动感无限的是 CSS (Cascading Style Sheets, 级联样式表) 和 DHTML (Dynamic HTML, 动态 HTML) 技术。1996 年底, W3C 提出了 CSS 标准, CSS 大大提高了开发者对信息展现格式的控制能力。DHTML 技术无需启动 Java 虚拟机或其他脚本环境, 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。

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, 这远不能满足人们对信息丰富性和多样性的强烈需求。这就促使 Web 技术由静态技术向动态技术转化。

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 CGI (Common Gateway Interface, 公共网关接口) 技术。1993 年, CGI 1.0 的标准草案由 NCSA (National Center for Supercomputing Applications, 国家计算机安全中心) 提出。1995 年, NCSA 开始制定 CGI 1.1 标准。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。随着 CGI 技术的普及, 聊天室、论坛、电子商务、信息查询、全文检索等各式各样的 Web 应用蓬勃兴起, 人们终于可以享受到信息检索、信息交换、信息处理等更为便捷的信息服务了。

CGI 是 Web 服务器扩展机制, 它允许用户调用 Web 服务器上的 CGI 程序。用户通过单击某个链接或者直接在浏览器的地址栏中输入 URL 访问 CGI 程序, Web 服务器接收到请求后, 发现该请求是给某个 CGI 程序, 就启动并运行该 CGI 程序, 对用户请求进行处理。CGI 程序解析请求中的 CGI 数据, 处理数据, 并产生一个响应 (HTML 页面)。该响应被返回 Web 服务器, Web 服务器包装该响应, 如添加报头消息, 以 HTTP 响应的形式发送给客户端浏览器。

但是, 用户在使用 CGI 程序时候发现编写程序比较困难, 而且对用户请求和响应时间较长。由于 CGI 程序的这些缺点, 开发人员需要其他的 CGI 方案。

1994 年, Rasmus Lerdorf 发明了专用于 Web 服务端编程的 PHP (Personal Home Page, 个人网页) 语言。它开始是一个用 Perl 语言编写的简单的程序, Rasmus Lerdorf 用它来与访问其主页的人保持联系。后来它逐步流行, Rasmus 又重新写了整个解析器, 并命名为 PHP V1.0, 当然功能还不是十分完善。此后其他程序员开始参与 PHP 源码的编写, 1997 年, Zeev Suraski 和 Andi Gutamns 又重新编写了解析器, 使功能基本完善, 随后发布了 PHP 3。

PHP 程序可以运行在 UNIX、Linux 或者 Windows 操作系统下, 对客户端浏览器也没有特殊的要求。PHP 也是将脚本描述语言嵌入 HTML 文档中, 它大量采用了 C、Java 和 Perl 语言的语法, 并加入了各种 PHP 自己的特征。PHP 的优点如下: 首先它是免费的, 这对于许多要考虑运行成本的商业网站来说尤其重要; 其次它开放源代码, 因为这一点, 所以才会有很

多爱好者不断发展它，使之更具有生命力；再次它有多平台支持，可以运行在所有的操作系统之下；最后它效率高，同 ASP 相比，PHP 占用资源少，执行速度较快。

1996 年，微软公司（微软公司是世界个人计算机软件开发的先导，由比尔·盖茨与保罗·艾伦创始于 1975 年，总部设在华盛顿州的雷德蒙市。目前是全球最大的计算机软件提供商。其主要产品为 Windows 操作系统、Internet Explorer 浏览器、Microsoft Office 办公软件套件、SQL Server 数据库软件和开发工具等。1999 年推出了 MSN 网络即时信息客户程序，2001 年推出 Xbox 游戏机，参与游戏终端机市场竞争）借鉴 PHP 的思想，推出了 ASP 技术。ASP 脚本所使用的 VBScript 脚本语言直接来源于 VB 语言，它秉承了 VB 简单易用的特点，可以把脚本语言直接嵌入 HTML 文档中，无需编译连接就可运行，不存在浏览器兼容的问题。2002 年微软发布 .NET 的正式版本 .Net Framework 1.0，其中的 ASP 版本就是 ASP.NET 1.0，在此之前发布了两个 .NET 测试版本 Beta1 和 Beta2。2003 年微软发布了 .Net Framework 1.1 正式版，其中 ASP 版本就是 ASP.NET 1.1。2005 年微软发布了 .Net Framework 2.0 正式版本，也就是 ASP.NET 2.0。微软在 2006 年发布了 .Net Framework 3.0。

ASP.NET 目前能支持 3 种语言：C#、Visual Basic.NET 和 Jscript.NET。与 ASP 相比，ASP.NET 的优点如下：使用 .NET 提供的所有类库，可以执行以往 ASP 所不能实现的许多功能；引入了服务器端控件的概念，这使开发交互式网站更加方便；引入了 ADO.NET 数据访问接口，大大提高了数据访问效率；提供 ASP.NET 的可视化开发环境 Visual studio.NET，进一步提高编程效率；保持对 ASP 的全面兼容；全面支持面向对象程序设计。

1997 年，Sun 公司推出了 Servlet 技术（Servlet 是运行在服务器端的程序，可以被认为是服务器端的 Applet。Servlet 被 Web 服务器（例如 Tomcat）加载和执行，就如同 Applet 被浏览器加载和执行一样。Servlet 从客户端（通过 Web 服务器）接收请求，执行某种操作，然后返回。它的主要优点包括持久性、平台无关性、可扩展性和安全性），成为 Java 阵营的 CGI 解决方案。1998 年，Sun 公司又推出了 JSP 技术，JSP 允许在 HTML 页面中嵌入 Java 脚本代码，从而实现动态网页功能。JSP 最大的优点是其开放的、跨平台的结构，它可以运行在所有的服务器系统上。2009 年 4 月 20 日，甲骨文（Oracle）公司以 74 亿美元收购 Sun 公司。

2000 年以后，随着 Web 应用程序复杂性不断提高，人们逐渐意识到，单纯依靠某种技术，很难实现快速开发、快速验证和快速部署的效果，必须整合 Web 开发技术形成完整的开发框架或应用模型，来满足各种复杂的应用程序的需求。Web 技术出现了几种主要的技术整合方式：MVC 设计模式、门户服务和 Web 内容管理。Struts、Spring、Hibernate 框架技术等都是在开源世界里与 MVC 设计模式、门户服务和 Web 内容管理相关的优秀解决方案。

1.1.2 Web 动态网页技术

动态网页技术是运行在服务器端的 Web 应用程序，根据用户的请求，在服务器端进行动态处理后，把处理的结果以 HTML 文件格式返回给客户端。当前主流的三大动态 Web 开发技术是：PHP、ASP/ASP.NET 和 JSP。

1. PHP

1994 年 Rasmus Lerdorf 创建了 PHP。1995 年初 Personal Home Page Tools（PHP Tools）发布了 PHP 1.0；1995 年又发布 PHP 2.0；1997 年发布 PHP 3.0；2000 年，发布 PHP 4.0；2009 年发布 PHP 5.3 版本；2011 年发布 PHP 5.4。

PHP 是一个基于服务端来创建动态网站的脚本语言，可以用 PHP 和 HTML 生成网站主页。当一个访问者打开主页时，服务端便执行 PHP 的命令并将执行结果发送至访问者的浏览器中，这类似于 ASP 和 JSP。然而 PHP 和它们的不同之处在于 PHP 开放源码和跨越平台，PHP 可以运行在 Windows NT 和多种版本的 UNIX 上。PHP 消耗的资源较少，当 PHP 作为 Apache Web 服务器一部分时，运行代码不需要调用外部二进制程序，服务器不需要承担任何额外的负担。

2. ASP/ASP.NET

ASP (Active Server Pages, 活动服务器页面) 是一种允许用户将 HTML 或 XML 标记与 VBScript 代码或者 JavaScript 代码相结合生成动态页面的技术，用来创建服务器端功能强大的 Web 应用程序。当一个页面被访问时，VBScript/JavaScript 代码首先被服务器处理，然后将处理后得到的 HTML 代码发送给浏览器。ASP 只能建立在 Windows 的 IIS Web 服务器上。

ASP 是 Microsoft 公司开发、用于代替 CGI 脚本程序的一种 Web 应用技术，可以与数据库和其他程序进行交互，是一种简单、方便的编程工具。ASP 是基于 Web 的一种编程技术，是 CGI 的一种。ASP 可以轻松地实现对页面内容的动态控制，根据不同的浏览者，显示不同的页面内容。1996 年，Microsoft 公司推出 ASP 1.0；1998 年，Microsoft 公司推出 ASP 2.0；1999 年，Microsoft 公司推出 ASP 3.0；2001 年，推出 ASP.NET。

ASP.NET 技术又称为 ASP+，是在 ASP 基础上发展起来的，是 ASP 3.0 升级版本，保留 ASP 的最大优点并全力使其扩大化，是 Microsoft 公司推出的新一代 Web 开发技术，是 .NET 战略中的重要一员，它全新的技术架构使编程变得更加简单，是创建动态网站和 Web 应用程序的最好技术之一。

3. JSP

JSP (Java Server Pages, Java 服务器页面) 是由 Sun 公司倡导、许多公司参与共同建立的一种动态网页技术标准。JSP 技术类似 ASP/ASP.NET 技术，它是在传统的网页 (HTML 文件) 中插入 Java 代码段和 JSP 标记，从而形成 JSP 文件。Web 服务器接收到访问 JSP 网页的请求时，首先将 JSP 转化为 Servlet 文件，Servlet 文件经过编译后处理用户请求，然后将执行结果以 HTML 格式返回给客户。

1998 年，Sun 公司推出 JSP 0.9 版本；1999 年推出 JSP 1.1 版本；2000 年推出 JSP 1.2 版本。现在主要使用的是 JSP 2.0 版本。

自 JSP 推出后，许多大公司都支持 JSP 技术的服务器，如 IBM、甲骨文、微软公司等，所以 JSP 迅速成为主流商业应用的服务器端动态 Web 技术。

1.1.3 Web 应用程序的工作原理

JSP 页面是运行在服务器端的一种 Web 应用程序。在学习 JSP 技术前，先了解一下 Web 应用程序的工作原理。

目前，在 Internet 上的信息大多以网页形式存储在服务器上，通过浏览器获取网页内容，这是一种典型的 B/S (浏览器/服务器) 模式。

B/S 模式的工作过程是：客户端请求-服务器处理-对客户端响应。

B/S 模式工作时，浏览器提交请求，Web 服务器接收到请求后把请求提交给相应的应用

服务器，由应用服务器调用相应的 Web 应用程序对客户端请求进行处理，将处理结果返回给 Web 服务器，Web 服务器将处理结果（网页）响应给客户端（浏览器）。Web 应用程序由动态网页技术开发，如 JSP、ASP、PHP 等，其实现的原理如图 1-1 所示。

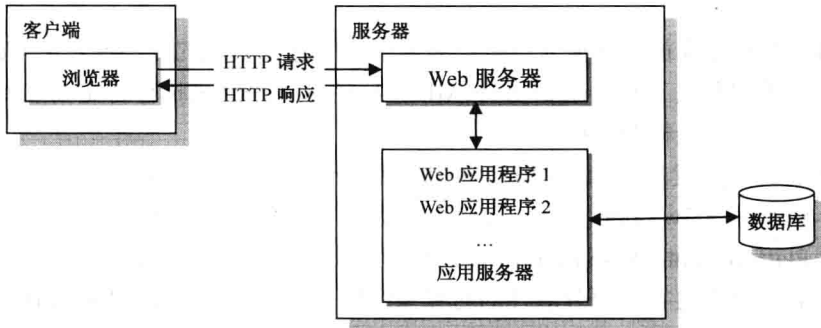


图 1-1 Web 应用程序的工作原理

1.2 JSP 技术概述

JSP 技术是由原 Sun 公司倡导、许多公司共同参与建立的一种基于 Java 语言的动态 Web 应用开发技术，利用这一技术可以建立安全、跨平台的先进动态网页技术。JSP 是 JavaEE 系统中的 Web 层技术，负责动态生成用户界面。JSP 页面在执行时采用编译方式，编译生成 Servlet 文件。

JSP 是从 Servlet 上分离出来的一小部分，简化了开发，加强了界面设计。JSP 定位在交互网页的开发。运用 Java 语法，但功能较 Servlet 弱了很多，并且高级开发中只充当用户界面部分。JSP 容器收到客户端发出的请求时，首先执行其中的程序片段，然后将执行结果以 HTML 格式响应给客户端。其中程序片段可以是：操作数据库、重新定向网页以及发送 E-Mail 等，这些都是建立动态网站所需要的功能。所有程序操作都在服务器端执行，网络上传送给客户端的仅是得到的结果，与客户端的浏览器无关，因此，JSP 被称为 Server-Side Language。JSP 的主要优点如下：

- 一次编写各处执行（Write once, Run Anywhere）

作为 Java 平台的一部分，JSP 技术拥有 Java 语言“一次编写，各处执行”的特点。随着越来越多的供货商将 JSP 技术添加到他们的产品中，用户可以针对自己公司的需求，做出谨慎评估后，选择符合公司成本及规模的服务器，假若未来的需求有所变更时，更换服务器平台并不影响之前所投下的成本、人力、所开发的应用程序。

- 搭配可重复使用的组件

JSP 技术可依赖于重复使用跨平台的组件（如：JavaBean 或 Enterprise JavaBean 组件）来执行更复杂的运算、数据处理。开发人员能够共享开发完成的组件，或者能够加强这些组件的功能，让更多用户或客户团体使用。基于善加利用组件的方法，可以加快整体开发过程，也大大降低公司的开发成本和人力。

- 采用标签化页面开发

Web 网页开发人员不一定是熟悉 Java 语言的程序员。因此，JSP 技术能够将许多功

能封装起来，成为一个自定义的标签，这些功能是完全根据 XML 的标准来制订的，即 JSP 技术中的标签库 (Tag Library)。因此，Web 页面开发人员可以运用自定义好的标签来达到工作需求，而无须再写复杂的 Java 语句，让 Web 页面开发人员亦能快速开发出动态内容网页。

今后，第三方开发人员和其他人员可以为常用功能建立自己的标签库，让 Web 网页开发人员能够使用熟悉的开发工具，用如同 HTML 一样的标签语法来执行特定功能的工作。

- N-tier 企业应用架构的支持

鉴于网络的国际化发展，未来服务会越来越繁杂，且不再受地域的限制，因此，必须放弃以往 Client-Server 的 Two-tier 架构，进而转向更具威力、弹性的分散性对象系统。由于 JSP 技术是 Java 2 Platform Enterprise Edition (J2EE) 集成中的一部分，它主要是负责前端显示经过复杂运算之后的结果内容，而分散性的对象系统则是主要依赖 EJB (Enterprise JavaBean) 和 JNDI (Java Naming and Directory Interface) 构建而成。

1.3 JSP 的工作原理

JSP 应用程序是运行在服务器端。服务器端收到用户通过浏览器提交的请求后进行处理，再以 HTML 的形式返回给客户端，客户端得到的只是在浏览器中看到的静态网页。JSP 工作原理如图 1-2 所示。

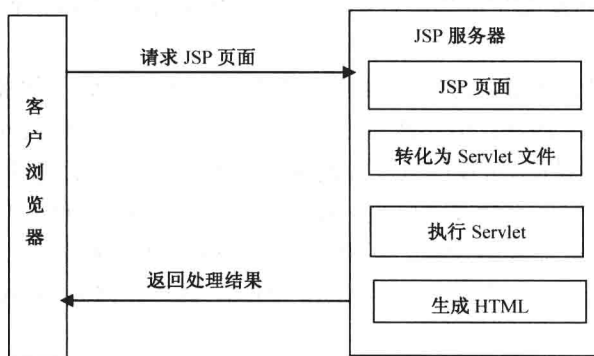


图 1-2 JSP 工作原理

所有的 JSP 应用程序在首次载入时都被编译成 Servlet 文件，然后再运行，这个工作主要是由 JSP 引擎来完成。当第一次运行一个 JSP 页面时，JSP 引擎要完成以下操作：

- (1) 当用户访问一个 JSP 页面时，JSP 页面将被编译成 Servlet 文件 (Java 文件)。
- (2) JSP 引擎调用 Java 编译器，编译 Servlet 文件为可执行的代码文件 (.class 文件)。
- (3) 用 Java 虚拟机 (JVM) 解释执行 .class 文件，并将执行结果返回给服务器。
- (4) 服务器将执行结果以 HTML 格式发送给客户端的浏览器。

由于一个 JSP 页面在第一次被访问时要经过编译成 Servlet 文件、Servlet 编译和 .class 文件执行这几个步骤，所以客户端得到响应所需要的时间比较长。当该页面再次被访问时，它对应的 .class 文件已经生成，不需要再次翻译和编译，JSP 引擎可以直接执行 .class 文件，因此 JSP 页面的访问速度会大大提高。