

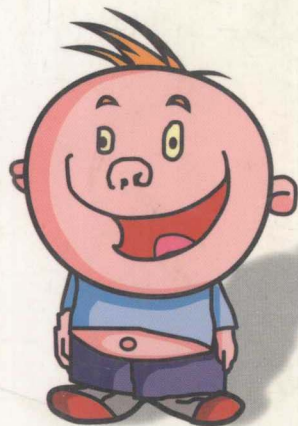
计算机编程起步



系列丛书

JSP 编程起步

清宏计算机工作室 编著



机械工业出版社
China Machine Press



附CD

TP393
2700-9

(附光盘一张)

计算机编程起步 ABC 系列丛书

JSP 编程起步

清宏计算机工作室 编著



200233509



机械工业出版社

200233509

Java 自从诞生的那一天起,就以其优良的设计受到众人的瞩目,并且迅速席卷全球。而近几年来,基于 Java 语言的 JSP 更是因为其优良的性能、良好的可扩展性和强大的支持功能而成为 Web 编程的首选方案。

本书旨在向初学者介绍 JSP 编程的基本知识和技巧。

全书分为预备篇、开始篇和加速篇三大部分,共 9 章。预备篇主要介绍一些学习 JSP 编程必备的基础知识和准备工作;开始篇向读者较为详细地介绍 JSP 编程各方面的基础知识和技巧,如 Java 语言、表单信息的获取和数据库的操作等等;加速篇则向读者介绍了 Java Servlet 和 Java Bean 两方面的内容,再结合两个综合实例进行深入说明。

本书可作为学习 JSP 编程的入门教材和教学参考书,亦可作为 Web 开发人员的参考用书。

图书在版编目(CIP)数据

JSP 编程起步/清宏计算机工作室编著. —北京:机械工业出版社, 2002. 1

(计算机编程起步 ABC 系列丛书)

ISBN 7-111-09472-7

I. J... II. 清... III. 主页制作—技术, JSP
IV. TP393.092

中国版本图书馆 CIP 数据核字 (2001) 第 081824 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:边 萌 许志华

责任印制:路 琳

北京机工印刷厂印刷·新华书店北京发行所发行

2002 年 1 月第 1 版第 1 次印刷

1000mm×1400mm B5·9.125 印张·352 千字

0 001—5000 册

定价:30.00 元 (含 1CD)

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话 (010) 68993821、68326677-2527

前言

随着 21 世纪“软件蓝领”工程的启动，迫切需要对广大非计算机专业大专院校的学生以及高中毕业等就业人员进行培训，从而迅速为社会输送大批能够胜任较简单编程的工作人员（软件蓝领）。

为此，本工作室策划并编著了《计算机编程起步 ABC 系列丛书》，通过浅显生动的讲解，贴近实际的大量实例，以 ABC 逐步引导，使得零起点的读者通过短期培训、自学，即可胜任简单编程，并为更上一层楼打下坚实的基础。

内容编排分三部分：

预备篇（使读者初步了解编程，确立信心、激发兴趣）

简介开发工具、操作系统、应用领域和使用特点；完整实例的演示让读者初步了解开发工具的操作和编程的基本规则。

开始篇（使读者掌握基本概念、实用技术与技巧）

开发工具的安装、环境配置；生动形象地介绍基本概念和知识；以 ABC 逐步引导介绍实例的方式，使读者明白各功能技术用在哪里，如何去用。各章结尾的“问题与提示”，可使读者进一步巩固所学的知识要点。

加速篇（提高读者编程水平）

综合实例给出：功能概述、编程思想、技术要点、开发流程、实现步骤以及详细讲解，让读者初步了解和掌握较大型、全局性开发的概念和方法。

配套光盘提供生动形象的多媒体演示、实例检索、源代码下载，便于读者学习和实际应用。

本套丛书适用于广大编程初学者自学，也可作为软件蓝领人员的培训教程。

我们恳请广大读者对书中的不足之处提出宝贵意见。

清宏计算机工作室

编 者 的 话

JSP(Java Server Pages)由 Sun Microsystem 公司于 1999 年 6 月推出, 是基于 Java Servlet 以及整个 Java 体系的 Web 开发技术。

本书的第一部分为预备篇(第 1~2 章), 主要介绍了 Web 编程和 JSP 编程的基本概念和 JSP 编程的开发环境的安装和设置。这一部分让读者了解什么是 JSP、如何建立 JSP 环境。

第二部分为开始篇(第 3~6 章), 主要向读者介绍了 Java 语言编程基础、JSP 语法、HTML 文件的编写和从 HTML 表单获取信息和 JSP 对数据库的操作等内容。通过这一部分的学习, 读者可以基本上掌握 JSP 编程的常用知识和技巧, 从而能够利用 JSP 进行一般的 Web 编程。

第三部分为加速篇(第 7~9 章), 介绍了 JavaBean 和 JavaServlet 这两部分内容。通过这部分的学习, 读者基本上就掌握了 JSP 编程的所有内容, 从而能够完成大多数 JSP 的编程任务, 可对 JSP 进行灵活的应用。

在某些章节中有一些问题用来复习和巩固读者在该章节所学的知识, 读者可以根据学到的内容来完成这些问题。

本书配套光盘中含有此书中的全部源程序, 这些程序都经过严格的调试和测试。读者可将其下载到硬盘上运行。多媒体演示能帮助读者更好地、快速地了解 JSP。

由于编者的时间关系和水平有限, 书中难免出错, 请读者谅解并批评指正。

编 者

目 录

前言

编者的话

预备篇..... 1

第 1 章 JSP 概述..... 2

1.1 JSP 技术简介..... 3

1.1.1 Web 编程技术..... 3

1.1.2 JSP 是什么..... 3

1.2 JSP 与其他动态网页技术的比较..... 4

1.2.1 JSP 与 ASP..... 4

1.2.2 JSP 与 CGI..... 6

1.2.3 JSP 与 PHP..... 7

1.3 本章小结..... 8

1.4 问题与提示..... 8

第 2 章 JSP 准备..... 9

2.1 Java 开发环境及其安装..... 10

2.1.1 Java 虚拟机..... 10

2.1.2 J2SDK 的介绍..... 10

2.1.3 J2SDK 的安装和设置..... 11

2.1.4 J2SDK 的使用..... 12

2.2 JSP 的安装..... 14

2.2.1 Resin 的安装..... 14

2.2.2 Resin 的参数设置..... 15

2.2.3 Resin 的启动和停止..... 15

2.2.4 安装成功了吗..... 16

2.2.5 其他选择..... 17

2.3 问题与提示..... 17

开始篇..... 18

第 3 章 初识 Java..... 19

3.1 为什么要学习 Java..... 20

3.1.1 Java 的来历..... 20

3.1.2 Java 的优点..... 20

3.1.3 第一个 Java 程序 Hello World..... 23

3.2 Java 的数据类型..... 24

3.2.1 简单类型..... 24

3.2.2 数组..... 25

3.2.3 数据类型转换..... 27

3.3 运算符..... 28

3.3.1 算术运算符..... 28

3.3.2 整数位运算符..... 30

3.3.3 关系运算符..... 33

3.3.4 布尔逻辑运算符..... 34

3.3.5 赋值运算符..... 34

3.3.6 条件运算符..... 35

3.3.7 运算符优先级..... 36

3.4 控制流..... 36

3.4.1 循环..... 37

3.4.2 分支..... 40

3.4.3 例外处理..... 43

3.4.4 程序转移..... 46

3.5 Java 的类..... 49

3.5.1 类的声明和类体..... 50

3.5.2 对象..... 51

3.5.3 new 运算符..... 53

3.5.4 方法声明..... 54

3.5.5 方法调用..... 55

3.5.6 this 的应用..... 55

3.5.7 构造函数..... 56

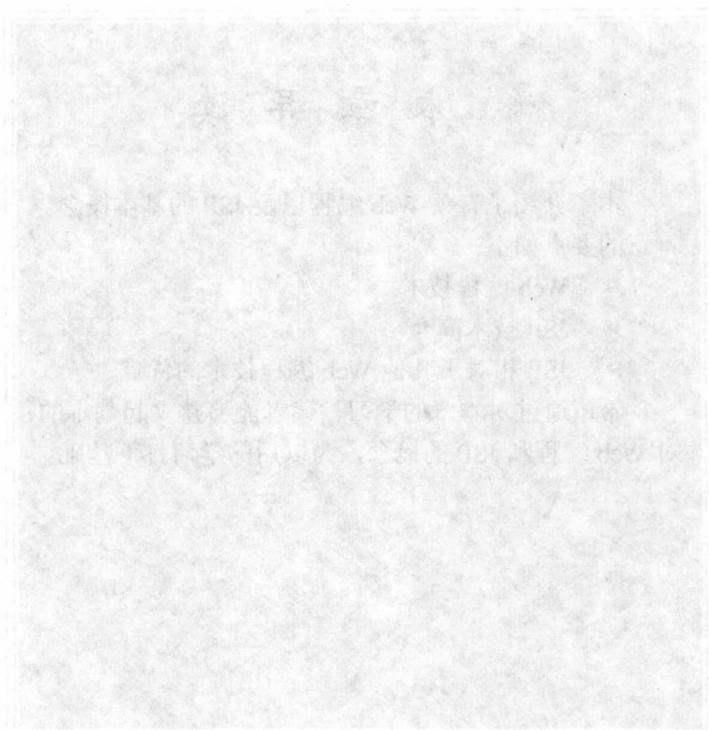
3.5.8 重载..... 58

3.5.9 继承.....	59	5.2 文字的语法与编辑	100
3.5.10 super 的应用	61	5.2.1 标题文字大小	100
3.5.11 final 的应用	63	5.2.2 文字的布局	100
3.5.12 finalize 的应用	64	5.2.3 文字格式的变化	101
3.5.13 static 的应用	65	5.3 网页中表单的设计	102
3.5.14 抽象.....	67	5.3.1 基本表单的制作	102
3.5.15 类的小结.....	68	5.3.2 表单的作用	113
3.6 包和接口	69	5.4 表格的制作	113
3.6.1 包	69	5.5 JSP 与客户的交互	117
3.6.2 接口.....	71	5.5.1 从表单中获取参数	117
3.7 问题与提示	73	5.5.2 会话控制 session/cookie	135
第 4 章 JSP 基本语法	76	5.6 本章小结	143
4.1 指令类标签	77	5.7 问题与提示	143
4.1.1 page 指令	77	第 6 章 操作数据库 JDBC	144
4.1.2 include 指令	79	6.1 JDBC.....	145
4.1.3 tablib 指令	80	6.1.1 从 ODBC 到 JDBC.....	145
4.2 动作类标签	81	6.1.2 JDBC 技术概述.....	146
4.2.1 <jsp:useBean>.....	81	6.1.3 JDBC 的结构.....	149
4.2.2 <jsp:setProperty>	82	6.2 SQL 语言简介.....	152
4.2.3 <jsp:getProperty>.....	83	6.2.1 SQL 介绍.....	152
4.2.4 <jsp:include>.....	84	6.2.2 SQL Server2000 的使用	152
4.2.5 <jsp:forward>.....	85	6.2.3 使用 SELECT 语句从表中 读取数据	155
4.2.6 <jsp:plugin>	86	6.2.4 使用聚类函数	156
4.3 脚本语法	88	6.2.5 创建新表	157
4.3.1 声明.....	88	6.2.6 字段属性	160
4.3.2 表达式.....	90	6.2.7 插入、删除和修改记录	161
4.3.3 实例——猜数游戏.....	91	6.2.8 删除和修改表	162
4.4 本章小结	95	6.3 在 JSP 中使用数据库	163
4.5 问题及提示	95	6.3.1 与数据库建立连接	163
第 5 章 HTML 和 JSP 的结合	96	6.3.2 查询数据库	164
5.1 认识 HTML 语言	97	6.3.3 处理由 SQL 操作返回的 结果集	165
5.1.1 什么是 HTML 语言	97		
5.1.2 HTML 文件结构	98		

6.3.4 更新数据库	168	8.1 JavaServlet 技术简介	203
6.4 重要类的详细说明	169	8.1.1 什么是 JavaServlet	203
6.4.1 DriverManager 和相关方法	169	8.1.2 JavaServlet 的机制	204
6.4.2 Connection 和相关方法	169	8.1.3 JavaServlet 的接口	204
6.4.3 ResultSet 和相关方法	170	8.1.4 JavaServlet 的生命周期	205
6.4.4 SQLException 和相关方法	170	8.1.5 JavaServlet 与 JSP 的关系	206
6.5 实例	171	8.2 JavaServlet 的运行	207
6.5.1 查询数据库	171	8.2.1 JavaServlet 的编译	207
6.5.2 更新数据库	174	8.2.2 JavaServlet 的存放路径	207
6.6 问题与提示	181	8.3 JavaServlet 与用户的交互	208
第 7 章 JavaBean	182	8.3.1 JavaServlet 与表单的交互	208
7.1 JavaBean 模型	183	8.3.2 操作 session	211
7.1.1 组件的概念	183	8.3.3 操作 cookie	218
7.1.2 JavaBean 简介	183	8.3.4 其他参数的获取	222
7.1.3 JavaBean 的特征	185	8.4 实例——留言簿	225
7.1.4 JavaBean 的生命周期	186	8.5 问题与提示	230
7.1.5 COM 与 JavaBean 的比较	187	加速篇	231
7.2 JavaBean 开发工具库简介	187	第 9 章 综合实例	232
7.3 JSP 中使用 JavaBean	188	9.1 聊天室	233
7.4 EJB 简介	192	9.1.1 功能概述	233
7.4.1 EJB 概要	192	9.1.2 编程思路及技术要点	233
7.4.2 EJB 中各角色的分析	193	9.1.3 最终结果	233
7.4.3 EJB 的体系结构	194	9.2 网上书店	245
7.5 JavaBean 实例——发送 邮件	196	9.2.1 功能概述	245
7.6 问题与提示	199	9.2.2 编程思路及技术要点	245
第 8 章 JavaServlet	202	9.2.3 最终结果	247
		9.3 本章小结	282

预备篇

本篇包括第 1 章和第 2 章，简介开发工具、操作系统、应用领域和使用特点；完整实例的演示，让读者初步了解开发工具的操作和编程的基本规则。



第

1

章

JSP 概 述

第 1 章

本章导读

本章介绍了有关 Web 编程以及 JSP 的基本概念。

本章的要点如下：

- Web 编程技术
- JSP 技术简介
- JSP 相对于其他 Web 编程技术的优点

希望通过对本章的学习，读者能够建立起基本的对 Web 编程和 JSP 的概念，为以后的学习打下基础。

1.1 JSP 技术简介

1.1.1 Web 编程技术

随着互联网的迅猛发展,网络带宽的不断增加,接入技术正经历着日新月异的变化。在强大的平台支持下,广大互联网用户对 Web 站点的内容和形式提出了新的要求。如果希望建立一个 Web 站点,仅仅依靠传统的静态页面已经远远不能满足要求,还需要考虑到如何实现对数据库的访问,如何对客户端输入作出及时的响应等等,而且站点必须健壮可靠,动态交互性强,同时还要具备良好的可扩展性。

最早实现这样的想法的是 CGI 技术,即通用网关接口(Common Gateway Interface)技术。CGI 可以用任何语言来编写,并且脚本能够访问数据库、系统服务器以及其他资源,能够动态地产生页面。

但是,CGI 也有它的缺点,例如,开发效率低,无论是用 C 还是 Perl 或者其他编程语言,程序员都必须去写很多繁琐的代码。再如,CGI 的运行速度慢。

而脚本语言是嵌入 HTML 页面中的,因此比较容易掌握,开发效率高,或多或少地弥补了 CGI 的不足。因此,现在流行很多用于 Web 编程的脚本语言。

目前流行的 Web 编程技术有 Perl、ASP、PHP、JSP、JavaServlet 以及 ColdFusion 等,虽然具体形式上并不相同,但是实现的功能基本一样,不管是哪种技术都可以访问数据库,都可以存取系统文件,并且最终都能够生成动态页面。

1.1.2 JSP 是什么

JSP 就是一种 Web 编程脚本语言。它(Java Server Pages)由 Sun Microsystem 公司于 1999 年 6 月推出,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先进、安全和跨平台的动态网站。

让我们先来看一个 HelloJsp.jsp 的小程序:

```
<html>
<head>
<title>JSP</title>
```

```
</head>
```

```
<body>
```

```
<%
```

```
String Str = "I like JSP";  
out.println(Str+"<br>");  
out.println("Hello The Word!");  
  
%>  
</body>  
</html>
```

上面的程序是最基本、最简单的 JSP 编程例子。这个程序会在页面上输出：

I like JSP!

Hello The World!

1.2 JSP 与其他动态网页技术的比较

1.2.1 JSP 与 ASP

总的来讲，Java Sever Pages (JSP) 和 微软的 Active Sever Pages (ASP) 在技术方面有许多相似之处。比如，两者都被基于 Web 应用实现动态交互网页制作提供的技术环境所支持；同时，两者都能够让程序员实现程序编制和网页设计的分离；而且，都能够替代 CGI 使网站建设与发展变得较为简单与快捷。

尽管 Java Sever Pages 技术和微软的 Active Sever Pages 在许多方面都有相似之处，但仍然存在很多不同之处，其中最本质的区别在于：两者来源于不同的技术规范组织，其实现的基础即 Web 服务器平台要求不相同。

1. JSP 技术是开放的技术

JSP 和 ASP 技术明显的不同点：开发人员对各自软件体系设计的深入了解的方式不同。JSP 技术基于平台和服务器的互相独立，支持广泛而专门，被各种工具包，服务器的组件和数据库产品开发商所提供。相比之下，ASP 技术主要依赖微软的技术支持。

(1) 平台和服务器的独立性 JSP 技术依附于一次写入，之后，可以运行在任何符合 Java 语法结构的环境中。与过去那种依附于单一平台或开发商的模式不同，JSP 技术能够运行在任何 Web 服务器上，并且支持来自多家开发商提供的工具包。

由于 ASP 是基于 ActiveX 控件技术提供客户端和服务器的开发组件，因此 ASP 技术基本上局限于微软的操作系统平台。ASP 主要工作环境是微软的 IIS 应用程序结构，又因 ActiveX 对象具有平台特性，所以 ASP 技术不能很容易地实现

第1章 JSP 概述

在跨平台的 Web 服务器上工作。尽管 ASP 技术通过第三方提供的产品能够实现跨平台使用,但是 ActiveX 对象必须事先放置于所选择的平台中。

(2) 开放的开发过程,开放的源代码 SUN 应用互相协作的过程开发 JSP 技术。自从 1995 年以来,SUN 已经用这种开放过程方法同国际 Java 组织合作开发和修改了 Java 技术与规范。针对 JSP 的产品,SUN 授权于工具提供商(如 Macromedia),结盟公司(如 Apache,Netscape),最终用户,协作商等。最近,SUN 将最新版本的 JSP 和 JavaServlet (JSP 1.1, JavaServlet 2.2) 的源代码发放给 Apache,以求 JSP 与 Apache 紧密的相互发展。同时,Apache, SUN 和许多其他的公司及个人公开成立了一个咨询机构以便任何公司和个人都能免费获取信息(详见: <http://jakarta.apache.org>)。

JSP 应用程序界面(API)毫无疑问已经取得成功,并将随 Java 组织的不断开放扩大而继续完善。相反,ASP 技术仅依靠微软本身的推动,其发展是建立在独占、封闭的开发过程基础之上。

(3) 从开发人员的角度来看 ASP 和 JSP 技术都能使开发者实现通过点击网页中的组件制作包括交互式的、动态的内容和应用程序的 Web 站点。ASP 仅支持组件对象模型 COM,而 JSP 技术提供的组件基于 JavaBean 技术或 JSP 标签库。由此可以看出两者虽有相同之处,但其区别也是很明显的。

1) JSP 标签的可扩充性 尽管 ASP 和 JSP 都使用标签与脚本技术来制作动态 Web 网页,JSP 技术能够使开发者应用扩展 JSP 标签库,所以开发者充分利用与 XML 兼容的标签技术的强大功能,大大减少对脚本语言的依赖。由于定制标签技术,降低了开发者制作网页和向多个网页扩充关键功能的复杂程度。

2) JSP 跨平台的可重用性 JSP 的开发人员在开发过程中一直关注可重用性。JSP 组件(EJB, JavaBean, 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 技术的标签可扩充功能为开发人员提供简便的、与 XML 兼容的接口,即共享网页的打包功能,使其完全的工业标准化。

这种基于组件的模式很有效地提高了应用程序的开发效率,因为这种模式能够使开发人员利用快捷的子组件创建模板应用程序,再整合一些附加功能以后便可使用。像这样有效的方法在 JSP 中无处不在,并可将其打包成一个 JavaBean 或一个工业标准化的 JavaBean 组件。

2. Java 的优越性

JSP 技术是用 Java 作为脚本语言的,而 ASP 网页使用微软的 VBScript 或 JavaScript。Java 是成熟的、强大的、易扩充的编程语言,远远优于基于 BASIC 的脚本语言。如:Java 的可执行性优于 VBScript 或 JavaScript 语言。因为它们利

用 Java 技术并且都被编译为 `JavaServlet`，JSP 网页为整个服务器端的 Java 库单元提供了一个接口来为 HTTP 的应用程序服务。

Java 使开发人员的工作在其他方面也变得一样容易、简单。例如，当 ASP 应用程序在 WINDOWS NT 系统被怀疑可能会崩溃时，Java 能有效的防止系统的崩溃。Java 语言通过提供防止内存泄漏的方法，在内存管理方面也大显身手。加之，JSP 还提供了健壮的意外事件处理机制。

(1) 易于维护性 基于 JSP 技术的应用程序比基于 ASP 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，所以它比较容易创建和维护庞大的、组件化的应用程序。

JSP 突出的组件技术使修改内容时不影响逻辑或修改逻辑时不影响内容变得很容易实现。企业级的 `JavaBean` 结构整合了企业逻辑，例如数据库的访问、安全、事务完整性及独立性。因为 JSP 技术是一种开放的、跨平台的结构，因此，Web 服务器、平台及其他的组件能很容易地升级或切换，且不会影响 JSP 基本的应用程序。这一特点使 JSP 能够适用现实世界的各种 WEB 应用程序的不断变化和发展的。

(2) 企业产品的多样性 `Java2` 平台即企业版(J2EE)是适用于多企业应用程序的 Java 结构，作为 J2EE 的一部分，JSP 网页可访问所有 J2EE 的组件，包括 `JavaBean`，企业级 `JavaBean` 及 `JavaServlet`。JSP 网页都能完全编译成为 `JavaServlet`，所以它们都享有灵活性的特点和作为服务器端的 Java 应用程序。J2EE 平台内容不仅包括管理复杂的企业应用程序，而且包括事务管理技术和 `Pooling` 资源管理技术。

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 规范技术：`Java IDL` 和 `RMI`。在企业级 `JavaBean` 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 的开放性产品，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，这样能够使开发者选择最佳的开发方法，选择最适应他们的应用程序开发的工具包，同时，有效地保护开发者在代码和人员培训上的投资。

1.2.2 JSP 与 CGI

CGI (Common Gateway Interface) 即公用网关接口，可以称它为一种机制，因此 CGI 程序可以用不同的语言来编写，如 `Visual Basic`、`Delphi` 或 `C/C++` 等。首

第1章 JSP 概述

先将已经写好的程序放在 Web 服务器的计算机上运行, 然后将其运行结果通过 Web 服务器传输到客户端的浏览器上。我们通过 CGI 建立 Web 页面与脚本程序之间的联系, 再利用脚本程序来处理访问者输入的信息而据此作出响应。事实上, 这样的编制方式比较困难而且效率低下, 因为每一次修改程序都必须重新将 CGI 程序编译成可执行文件。

下面从几个方面比较 JSP 与传统 CGI 的特性。

1. 可移植性

大多数 Web 服务器支持 CGI 接口, 但 CGI 程序自身并不能跨平台运行; Java Servlet 程序则具有 Java 程序的优点, 可运行在任何平台之上, 大多数 WEB 及应用服务器都支持 Java 及 JavaServlet API。

2. 性能

传统的 CGI 环境下, 客户端每一次对 CGI 程序的请求, 都使服务器产生一个新的进程来装载、执行 CGI 程序, 大量的并行请求大大降低了其性能。JSP 则没有这个局限, 每个程序装载一次, 并驻留在服务器内存为以后请求服务。另外 JSP 在多线程的编写与控制方面比 CGI 容易得多。

3. 开发及发布

由于 JavaServlet 具有 Java 的所有优点, 因此开发起来也相对容易, Java 简化了对异常的处理, 它的面向对象特性使开发人员的协作成为一件简单的事。JSP 技术从如下几个方面加快了动态网站的开发。

(1) 将静态页面的开发和动态内容的生成分开 JSP 中通过使用 HTML 或 XML 标记来规划、设计 WEB 页的布局和风格, 而用 JSP 标记来实现动态的内容, 生成动态内容的部分被封装起来运行于服务器端, 这样页面布局和风格可单独编辑与调试而不影响动态内容生成。

(2) 组件的重用 多数 JSP 页面通过重用与平台无关的组件 JavaBean 来完成复杂的要求, 这些组件可在开发人员中重复使用。

(3) 通过标记来简化页面开发 通过 JSP 提供的标记, 开发人员可以很容易地使用 JavaBean 组件, 设置和访问它们的属性。同时 JSP 也允许用户自定义标记, 还可接受第三方开发的标记, 从而可以方便地使用第三方提供的功能组件。

1.2.3 JSP 与 PHP

1. 可移植性

PHP 和 JSP 都具有良好的可移植性。它们都能够运行在各种不同的平台上。但是需要注意的是: PHP 对数据库操作的接口繁多, 这样设计的初衷也许是方便用户, 但是却给移植带来了麻烦。例如, 对 MySQL 数据库系统的操作和对 Oracle

数据库系统的操作接口是不同的,因此如果要更换数据库系统,就得更改源码中所有涉及数据库操作的部分。

2. 可扩展性

PHP 缺少一层逻辑层。JSP 可以使用 JavaBean,ASP 的背后则是 COM/DCOM。PHP 没有相对应的组件,这使得 PHP 的可扩展性大打折扣。读者通过后面的学习会逐渐认识到 JavaBean 的重要性。

3. 性能

很多测试表明,PHP 的计算速度很慢。因此 PHP 不宜使用在运算较多的情况下。而 JSP 在这方面的表现却十分优秀。

4. 支持

JSP 技术有 SUN 公司的支持,而 PHP 却没有相应的公司支持。PHP 秉承 GNU 风格,开放源代码,对它进行维护升级的是一些网络爱好者。因此在这一点上,JSP 具有很大的优势。

1.3 本章小结

本章首先向读者介绍了 Web 编程的概念,然后介绍了 JSP 技术以及它的特点。为了让读者更加清楚 JSP 的优势,本章还把 JSP 和其他 Web 编程技术做了比较。

1.4 问题与提示



问题

1. 简述什么是 Web 编程技术(参考本章 1.1.1 小节)。
2. 简述 JSP 的优点。
3. 通过 JSP 和其他动态网页技术的比较,请问 JSP 是网页技术发展趋势吗?



提示

1. 可从 JSP 的可扩展性、易维护性以及性能这几个方面进行简述。
2. 可以对 JSP 和 CGI、ASP 和 PHP 等其他 Web 编程技术进行比较,然后做出结论。

第

2

章

JSP 准备

本章导读

本章介绍了 JSP 环境的建立。主要介绍了以下三方面：

- 虚拟机的概念
- Java 开发环境 J2SDK 的安装
- JSP 引擎 Resin 的安装

本章结束的时候，读者可以在建立起来的环境上运行我们第一个 JSP 程序：hello.jsp。