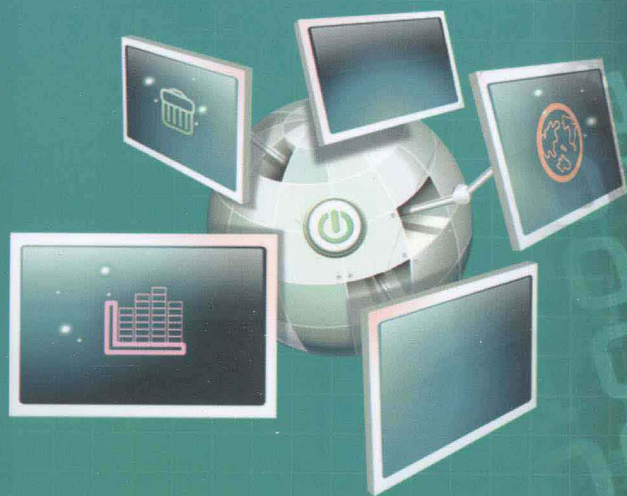




北京市职业院校专业创新团队建设计划资助项目
北京劳动保障职业学院国家骨干校建设资助项目

JSP开发

案例教程



何福贵 张梅 张力展 ◎ 编著



机械工业出版社
CHINA MACHINE PRESS



北京市职业院校专业创新团队建设计划资助项目
北京劳动保障职业学院国家骨干校建设资助项目

JSP 开发案例教程

何福贵 张 梅 张力展 编著



机械工业出版社

本书详细讲解了JSP的语法和Web程序设计方法,全面介绍了JSP动态网页制作技术和相关理论。全书共分11章,内容包括JSP开发概述、JSP开发架构、JSP基本语法、JSP内置对象、Servlet基础、JavaBean技术、应用JDBC进行数据库开发、JSP和XML、使用JSP+Servlet+JavaBean实现MVC、JSP实用组件、JSP高级开发。本书在动态网页开发方面紧跟主流技术,各章之间紧密联系,前后呼应,循序渐进,并且融入了大量实例,供读者参考和实践。

本书适合作为高职高专院校的教材,既可作为JSP初学者的入门教材,也可作为社会培训班和广大JSP开发爱好者的参考用书。

图书在版编目(CIP)数据

JSP开发案例教程/何福贵,张梅,张力展编著. —北京:机械工业出版社,2013.10

北京市职业院校专业创新团队建设计划资助项目 北京劳动保障职业学院国家骨干校建设资助项目

ISBN 978-7-111-44234-9

I. ①J… II. ①何…②张…③张… III. ①JAVA语言—网页制作工具—高等职业教育—教材 IV. ①TP312②TP393.092

中国版本图书馆CIP数据核字(2013)第233929号

机械工业出版社(北京市百万庄大街22号 邮政编码100037)

策划编辑:罗莉 责任编辑:罗莉

版式设计:常天培 责任校对:刘秀芝

封面设计:赵颖喆 责任印制:张楠

唐山丰电印务有限公司印刷

2014年1月第1版第1次印刷

184mm×260mm·17.25印张·426千字

0001—3000册

标准书号:ISBN 978-7-111-44234-9

定价:49.00元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心:(010) 88361066 教材网:<http://www.cmpedu.com>

销售一部:(010) 68326294 机工官网:<http://www.cmpbook.com>

销售二部:(010) 88379649 机工官博:<http://weibo.com/cmp1952>

读者购书热线:(010) 88379203 封面无防伪标均为盗版

前 言

感谢您选择本书，为了帮助您更好地学习本书的知识，请仔细阅读下面的内容。

本书是根据教育部对高职高专教育人才培养工作的指导思想编写的，全书贯穿案例项目教学的思想，从初学者的角度出发，采用循序渐进、逐步扩展的模式进行编写，深入浅出地介绍 JSP 开发技术。

本书主要针对高职高专院校的教学特点设计，对内容的选择和编排尽量满足他们的要求，将教学内容主要定位于 JSP 动态网站开发的初步能力，让学生通过这门课程的学习，能够具备综合运用专业软件对中大型网站进行设计和开发的能力，为今后的职业发展打下良好基础。

全书共分 11 章，内容概括如下：

第 1、2 章介绍了 JSP 开发的前期工作。第 1 章介绍了目前正在使用的 Web2.0，及未来 Web 的发展方向，介绍了 JSP 运行的环境的搭建，介绍了 JSP 开发环境的搭建；第 2 章介绍了常用的网站开发架构，并进行了比较，还介绍 JSP 的开发框架。

第 3~7 章主要介绍的是 JSP 开发的基本内容，其中每一章涉及一个知识模块。第 3 章介绍了 JSP 基本语法，包括 JSP 的脚本元素、指令元素和动作元素；第 4 章介绍了 JSP 内置对象，主要介绍了 9 个内置对象的使用方法；第 5 章介绍了 Servlet 的功能、技术特点、工作原理和使用；第 6 章介绍了 JavaBean 技术，包括 JavaBean 的特点和使用；第 7 章介绍了应用 JDBC 进行数据库开发，包括连接各种数据库的方法和连接池等。每个模块都结合了相关的实用案例，对于掌握知识起到了很好的作用。

第 8 章介绍了 JSP 和 XML。XML 是一种可以将数据结构化整理并管理的下一代语言标准，更具有扩展性，与其他语言相比有更加简单易用的特点，是继 HTML 和 Java 之后在 Internet 上的热点。JSP 通过 XML 实现软件的可伸缩性和可升级性。本章叙述了 XML 的语法规则，介绍了 Java 和 XML 之间的转化。

第 9 章通过实例介绍了如何使用 JSP、Servlet、JavaBean 实现 MVC。本章介绍了 MVC 的模式和特点，介绍了 JSP、Servlet、JavaBean 实现 MVC 三层架构购书网的设计。

第 10 章介绍了 JSP 实用组件。利用 JSP 实用组件可以完成一些实用的功能，加快开发的速度。本章介绍了常用的 JSP 实用组件——文件操作、发送接收邮件、动态图表组件、报表组件和 JExcel 组件，以及它们的使用方法。

第 11 章介绍了 JSP 高级开发技术。本章主要对 JSP 高级开发技术部分内容如 Struts2、Spring、Hibernate 的基本内容进行了介绍，介绍了这三部分的原理和作用、环境的配置和简单地使用，并介绍了 SSH 框架的整合。

本书有下列特点：

(1) 面向高职。本书按照高职的教学特点进行编写，以案例为主线进行内容的讲解。

(2) 有序分类。按照循序渐进的学习方式,本书对学习内容进行重新进行了整理排列,使得每一章既具有独立性,整体上又具有完整性。

(3) 内容全面实用。本书包括 JSP 初级和高级开发技术。

(4) 体现新技术的使用。

(5) 主要章最后有综合实例,是对本章的综合应用。

对在写作过程中给予我们帮助的朋友们,在此表示深深的谢意,并感谢机械工业出版社给予的帮助。由于编写时间仓促,加之作者水平有限,书中疏漏和错误之处在所难免,望广大专家、读者提出宝贵意见,以便修订时加以改正。

作 者

目 录

前言

第 1 章 JSP 开发概述	1
1.1 Web 技术概述	1
1.1.1 静态网页与动态网页	1
1.1.2 从 Web1.0 到 Web 2.0	2
1.1.3 Web 2.0 的应用发展方向	3
1.1.4 Web 2.0 的相关应用	5
1.2 JSP 运行环境	7
1.2.1 JDK 的下载与安装	7
1.2.2 安装与配置 Tomcat	8
1.2.3 启动与停止 Tomcat	8
1.2.4 案例 1: 一个简单的 JSP 程序	8
1.3 JSP 开发工具	10
1.3.1 JSP 的开发和应用平台的介绍	10
1.3.2 配置 Eclipse 的 JSP 开发环境	11
1.3.3 MyEclipse 开发 JSP	12
1.3.4 案例 2: 在 MyEclipse 下创建一个 JSP 程序	14
第 2 章 JSP 开发架构	15
2.1 软件编程体系简介	15
2.1.1 C/S 结构编程体系	16
2.1.2 B/S 结构编程体系	16
2.2 企业应用开发架构	17
2.2.1 两层架构	17
2.2.2 三层架构	17
2.2.3 N 层架构	17
2.2.4 开发架构比较	18
2.3 JSP 概述	18
2.3.1 什么是 JSP	18
2.3.2 JSP 技术原理	19
2.3.3 JSP 和其他动态网站开发技术	20
2.4 JSP 知识体系及学习之路	21
2.4.1 JSP 知识体系	22
2.4.2 JSP 程序员学习路径	22
2.5 小结	23
第 3 章 JSP 基本语法	24
3.1 JSP 文件基本结构	24

3.2 JSP 的脚本元素	25
3.2.1 注释	25
3.2.2 声明语句	26
3.2.3 脚本段	26
3.2.4 表达式	27
3.3 JSP 指令元素	27
3.3.1 page 指令	27
3.3.2 include 指令	29
3.3.3 taglib 指令	29
3.4 JSP 动作元素	30
3.4.1 <jsp: include >	30
3.4.2 <jsp: forward >	31
3.4.3 <jsp: param >	32
3.4.4 <jsp: useBean >	33
3.4.5 <jsp: plugin >	35
3.5 案例: 计算三角形的面积	38
第 4 章 JSP 内置对象	40
4.1 JSP 内置对象概述	40
4.2 request 对象常用方法和应用实例	42
4.2.1 request 对象常用方法	42
4.2.2 request 对象应用实例	43
4.3 response 对象常用方法和应用实例	46
4.3.1 response 对象的常用方法	46
4.3.2 response 对象应用实例	47
4.4 out 对象常用方法和应用实例	50
4.4.1 out 对象常用方法	50
4.4.2 out 对象应用实例	50
4.5 session 对象	51
4.5.1 session 的概念	52
4.5.2 session 对象的 Id	52
4.5.3 session 的有效期限	53
4.5.4 访问 session 中的数据	53
4.5.5 其他 session 对象的常用方法	54
4.5.6 session 对象应用实例	54
4.6 application 对象常用方法和应用实例	57
4.6.1 存取 application 中的数据	57
4.6.2 使用 application 对象取得信息	58

4.6.3 application 对象应用实例	58
4.7 其他 JSP 内置对象	60
4.7.1 pageContext 对象	60
4.7.2 config 对象	63
4.7.3 page 对象	64
4.7.4 exception 对象	65
4.8 小结	65
第 5 章 Servlet 基础	66
5.1 Servlet 介绍	66
5.1.1 Servlet 技术功能	67
5.1.2 Servlet 技术特点	67
5.1.3 JSP 与 Servlet 的关系	68
5.1.4 Servlet 的工作原理	69
5.1.5 Servlet 常用接口和类	70
5.2 开发部署一个简单的 Servlet	70
5.2.1 创建 Servlet 文件	73
5.2.2 Servlet 的配置文件	74
5.3 Servlet 实现相关的接口和类	76
5.3.1 GenericServlet	77
5.3.2 HttpServlet	77
5.3.3 Servlet 实现相关实例	78
5.4 Servlet 请求和响应相关	82
5.4.1 HttpServletRequest 接口	82
5.4.2 HttpServletResponse 接口	84
5.4.3 Servlet 请求和响应相关实例	84
5.5 Servlet 配置相关	86
5.5.1 ServletConfig 接口	87
5.5.2 获取 Servlet 配置信息的例子	87
5.6 Servlet 中的会话追踪	90
5.6.1 HttpSession 接口	90
5.6.2 HttpSession 应用实例	91
5.7 Servlet Context	94
5.7.1 ServletContext 接口	94
5.7.2 ServletContext 接口的应用实例	94
5.8 Servlet 协作	96
5.8.1 RequestDispatcher	96
5.8.2 forward () 控制页面跳转	97
5.8.3 include () 控制页面包含	98
5.9 Servlet 异常相关	98
5.9.1 声明式异常处理	98
5.9.2 程序式异常处理	101
5.10 Servlet 应用实例	104
5.11 小结	108

第 6 章 JavaBean 技术	109
6.1 剖析 JavaBean	109
6.1.1 什么是 JavaBean	110
6.1.2 JavaBean 的特征	111
6.1.3 创建一个 JavaBean	112
6.2 在 JSP 中使用 JavaBean	113
6.2.1 调用 JavaBean	114
6.2.2 访问 JavaBean 属性	114
6.2.3 设置 JavaBean 属性	114
6.2.4 JavaBean 的生命周期	114
6.2.5 类型自动转换规则	119
6.3 案例: 使用 JavaBean 处理表单数据	119
6.3.1 JSP 处理与 form 相关的常用标签 简单实例	120
6.3.2 设置中文编码	126
6.3.3 POST 与 GET 的差异	127
6.4 小结	127
第 7 章 应用 JDBC 进行数据库 开发	128
7.1 JDBC 概述	128
7.1.1 JDBC 的用途	129
7.1.2 JDBC 的典型用法	129
7.1.3 JDBC 体系结构	130
7.1.4 驱动器类型	130
7.1.5 安装驱动器	132
7.2 JDBC 连接数据库的方法	132
7.3 使用 JDBC 操作数据库	133
7.3.1 使用 JDBC 访问数据库的过程	134
7.3.2 使用 Statement 执行 SQL 语句	139
7.3.3 PreparedStatement 接口	149
7.3.4 CallableStatement 对象	154
7.3.5 使用 ResultSet 处理结果集	156
7.4 Java 与 SQL 的数据类型转换	160
7.5 连接池	161
7.5.1 连接池的实现原理	162
7.5.2 在 Tomcat 上配置数据源与 连接池	163
7.5.3 配置连接池时需要注意的问题	165
7.6 存取二进制文件	166
7.6.1 图像文件存取到数据库的过程	166
7.6.2 声音文件存取到数据库的过程	170
7.6.3 视频文件存取到数据库的过程	174

7.7 JSP 使用 JavaBean 访问数据库的 分页显示的实现	175	组件将文件上传到服务器	221
7.8 小结	183	10.2 发送邮件	223
第8章 JSP 和 XML	184	10.2.1 JavaMail 组件简介	223
8.1 XML 简介	184	10.2.2 JavaMail 组件简介	224
8.1.1 XML 的特点	184	10.2.3 搭建 Java Mail 的开发环境	228
8.1.2 XML 的内容	186	10.2.4 案例: 利用 JavaMail 组件 发送 Email	228
8.1.3 XML 的语法规则	188	10.3 JSP 动态图表组件	231
8.2 JDK 中的 XML API	191	10.3.1 JFreeChart 的下载与使用	231
8.3 XML 解析模型	191	10.3.2 JFreeChart 的核心类	232
8.3.1 DOM 解析	191	10.3.3 案例: 利用 JFreeChart 生成论坛 版块人气指数排行的柱形图	232
8.3.2 SAX 解析	192	10.4 JSP 报表组件	235
8.4 XML 与 Java 类映射 JAXB	193	10.4.1 iText 组件简介	235
8.4.1 什么是 XML 与 Java 类映射	193	10.4.2 iText 组件的下载与配置	235
8.4.2 JAXB 的工作原理	194	10.4.3 案例: 应用 iText 组件 生成 JSP 报表	235
8.4.3 Java 对象转化成 XML (Marshal)	194	10.5 jExcel 组件	239
8.4.4 XML 转化为 Java 对象 (Unmarshal)	196	10.5.1 jExcel 组件一下载与配置	239
8.4.5 更为复杂的映射	197	10.5.2 jExcel 组件—基本操作	239
8.5 案例: JSP + XML 实现电子 广告系统	200	10.5.3 jExcel 组件—高级操作	240
8.6 小结	203	10.6 小结	241
第9章 使用 JSP、Servlet、JavaBean 实现 MVC	204	第11章 JSP 高级开发	242
9.1 MVC 基础	204	11.1 Struts	242
9.1.1 MVC 的需求	204	11.1.1 配置 Struts 开发环境	243
9.1.2 MVC 的基本模式	205	11.1.2 Struts 工作原理	244
9.1.3 使用 MVC 的优点	206	11.1.3 一个简单的 Struts2 实例	246
9.2 案例: JSP、Servlet、JavaBean 实现 MVC 三层架构购书网	206	11.1.4 深入使用 Struts2	248
9.2.1 数据库设计	206	11.2 Spring	250
9.2.2 视图—JSP 页面开发	208	11.2.1 Spring 的起源和背景	250
9.2.3 模型—定义 Bean 来处理数据	211	11.2.2 Spring 的下载和安装	251
9.2.4 控制—编写 Servlet 处理请求	213	11.2.3 Spring 的核心机制: 依赖注入	254
9.2.5 其他 Bean 类—Util	218	11.2.4 Spring 容器的管理	255
9.2.6 部署	218	11.3 Hibernate	255
9.3 小结	218	11.3.1 Hibernate 和 ORM	256
第10章 JSP 实用组件	219	11.3.2 Hibernate 的体系结构	257
10.1 JSP 文件操作组件	219	11.3.3 Hibernate 的下载和安装	259
10.1.1 添加表单及表单元素	220	11.3.4 例子: Hibernate 访问 MySQL 数据库	260
10.1.2 创建上传对象	220	11.4 MyEclipse + Struts + Spring + Hibernate 整合	264
10.1.3 解析上传请求	220	11.5 小结	267
10.1.4 案例: 应用 commons - fileUpload 将文件上传到服务器	221	参考文献	268

第 1 章 JSP 开发概述



知识目标

- 熟悉动态网页和静态网页的区别
- 掌握 Web 2.0 的运行模式



能力目标

- 掌握 JSP 环境的配置
- 掌握 Myeclipse 的使用

本章主要对 JSP 技术进行概要介绍，首先介绍了 Web 技术、Web 2.0 的应用发展方向和 JSP 运行环境的搭建，重点介绍了 JSP 开发工具 Myeclipse 的使用方法。本章对 JSP 知识体系的剖析有助于读者学习和掌握 JSP 知识体系中的各个模块，使读者对 JSP 技术有一个总体性的了解。

1.1 Web 技术概述

1.1.1 静态网页与动态网页

静态网页是网站建设初期经常采用的一种形式。网站建设者把内容设计成静态网页，访问者只能被动地浏览网站建设者提供的网页内容。其特点如下：

- 1) 网页内容不会发生变化，除非网页设计者修改了网页的内容。
- 2) 不能实现与浏览网页的用户之间的交互。信息流向是单向的，即是从服务器到浏览器的。服务器不能根据用户的选择调整返回给用户的内容。

网络技术日新月异，许多网页文件扩展名不再只是 .htm，还有 .php、.asp 等，这些都是采用动态网页技术制作出来的。动态网页其实就是建立在 B/S 架构上的服务器端脚本程序。在浏览器端显示的网页是服务器端程序运行的结果。

静态网页与动态网页的区别在于 Web 服务器对它们的处理方式不同。当 Web 服务器接收到对静态网页的请求时，服务器直接将该页发送给客户浏览器，不进行任何处理。如果接收到对动态网页的请求，则从 Web 服务器中找到该文件，并将它传递给一个称为应用程序服务器的特殊软件扩展，由它负责解释和执行网页，将执行后的结果传递给客户浏览器。图 1-1 所示为动态网页的工作原理图。

动态网页的一般特点如下：

- 1) 动态网页以数据库技术为基础，可以大大降低网站维护的工作量。

2) 采用动态网页技术的网站可以实现更多的功能,如用户注册、用户登录、搜索查询、用户管理、订单管理等。

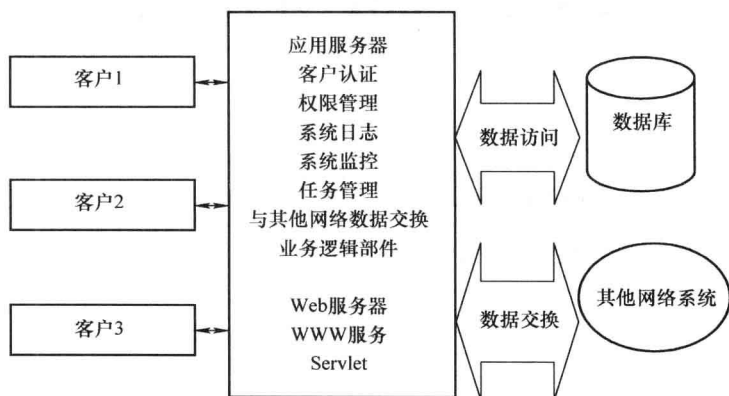


图 1-1 动态网页的工作原理图

3) 动态网页并不是独立存在于服务器上的网页文件,只有当用户请求时服务器才返回一个完整的网页。

4) 搜索引擎一般不可能从一个网站的数据库中访问全部网页,因此采用动态网页的网站在进行搜索引擎推广时需要做一定的技术处理才能适应搜索引擎的要求。

1.1.2 从 Web1.0 到 Web 2.0

Web 的发展就不得不提到 Web1.0 和 Web2.0,这不是技术性的概念,它们并没有技术上的严格划分,仅仅是 Web 发展历史断代的成果。

回忆 Web1.0 的时代,是使用 HTML 语言将信息编写成静态的页面,然后发布在 Web 上供用户浏览。而后技术略有发展,将要呈现的数据存储在数据库中,通过 Web 服务端的程序、应用户的请求取出数据,加上事先设计的模板,动态地生成 HTML 代码,发送到用户的浏览器那里,这就是动态 HTML,人们称之为 Web1.5。但是可以看到,在效果和影响上 Web1.5 只是对 Web1.0 的扩展和加深,与 Web1.0 并没有实质性的区别。

Web 的发展永远不会停滞,而后在 Web1.5 的基础上又加上了 0.5,成为了 Web2.0, Web2.0 的变化又在哪里呢?它和 Web1.0 相比有什么不同,笔者将从以下几个方面对 Web1.0 与 Web2.0 进行讨论,以使读者了解从 Web1.0 到 Web2.0 的转变。

(1) 运行模式的转变

Web1.0 时代,网站处于主导地位,用户只能被动地接收信息。用户通过访问门户网站来浏览网页、查询信息。Web1.0 的运行模式如图 1-2 所示。

到了 Web2.0 时代,网站开始慢慢地向用户靠拢,越来越多的网站意识到用户参与的重要性。于是 Digg、Wiki、BLOG 等新概念和新应用诞生了,用户上网不再仅仅是单纯地接收信息,而是可以发布信息。网站逐渐成为信息发布的主要平台,用户逐渐开始主导整个网络信息的流向。Web2.0 的运行模式如图 1-3 所示。

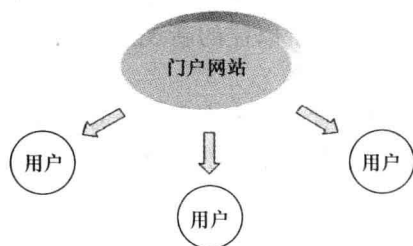


图 1-2 Web1.0 运行模式

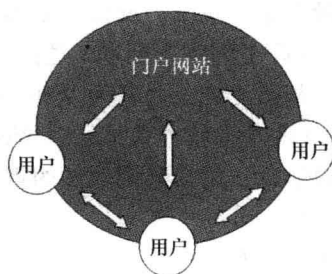


图 1-3 Web2.0 运行模式

(2) 关注点的转变

在 Web1.0 中网站关注的重点是内容，由于在 Web1.0 时代用户只能从 Web 上读取信息，因此如何方便、正确地获取信息将是用户选择网站的唯一理由，提供大量丰富、正确的信息也是网站获取点击率的唯一保证。而当时的网站内容又全部需要网站管理人员进行维护和更新，所以在 Web1.0 时代的各个门户网站中发布网站内容的优秀编辑将是网站赖以生存的保障。

而进入 Web2.0 时代后，网站逐渐由信息发布者演变为平台提供者。各个 Web2.0 网站中的内容都将由用户进行完善，用户与网站、用户与用户之间都可以进行交流。这时如何提供方便舒适的应用来帮助用户之间进行交流和在网站上发布信息就成为了网站需要考虑的问题。因此 Web2.0 时代丰富和方便的应用就成为了网站赖以生存的保障。

(3) 网络营销的转变

Web1.0 是“草根赚钱”，以量取胜。Web1.0 时代评价网络营销成果的普遍方法是比拼用户数和点击率。这种观念一直影响着当今的大部分网站经营者，在这种模式下开创了基于网页浏览量的广播式网络广告、无线两种收入模式，构成第一代互联网商业模式的核心，甚至这两种模式的增长潜力和创新能力直到今天都远未达到高潮，还存在巨大的发展空间。

Web2.0 是“大师赚钱”，以质取胜。Web2.0 不像 Web1.0 那样靠流量赚钱，而更比拼用户的粘着度，少量的用户可以通过使用大量的应用带来极大的利润。草根和大师的差别在于，草根工作着眼于广度，大师的工作着眼于深度。大师除了体力劳动外，更有创造性思维，增加了客户对他的依赖程度，也增加了客户对他的投资。虽然大师的客户数量远比草根少，但单个客户带来的收益远大于草根，这也是有名的“二八”原则。

从上面的比较中读者不难发现，Web2.0 在很大程度上并不是从技术创新角度出发的，或者说与技术没有很大的相关性。而应该从运营思路的角度来看待它的转变。

Web1.0 到 Web2.0 的转变可以总结为，从模式上，是从读向写、信息共同创造的一个转变；从基本结构上，则是由网页向工具的转变；从内容上，是由专业人士向普通用户的转变；从运行机制上，则是自 Client Server 向 Web Services 的转变。Web2.0 的精髓就是以人为本，提升用户使用互联网的体验。如此远大理想的发展趋势如何，将在本章 1.1.3 节进行简单介绍。

1.1.3 Web 2.0 的应用发展方向

Web 崛起于 20 世纪 90 年代，而后尽管经历了 2000 年初的股市崩盘与泡沫化洗礼，但

其发展的脚步毫不减缓,反而发展得更为稳健和成熟。2004 年,以 O'Reilly 总裁 Tim O'Reilly 为首的许多网络先驱者;通过观察和归纳,创造了不同于以往的新观念与新做法,并给这些观念和做法统一命名为 Web2.0。

Web2.0 的基本精神包括:以 Web 为平台,强调从使用者的角度来考虑,让使用者自己掌握自己的信息,尤其强调软件的重要性以及群体智慧的运用。它的发展方向可以从以下 4 个方面来归结。

(1) 内容 (Content)

在 Web2.0 时代强调提供双向交流,由使用者参与并贡献内容。即在 Web2.0 时代用户不只是消费者,更是内容的提供者。分析由使用者贡献内容的众多案例,可看出应用范围非常广泛,从具有高度知识性的百科全书 Wikipedia 到仅只是由使用者投票决定新闻重要性的 Digg.com 都是有名的成功案例。从中可以看到今后用户主体地位的不断提高和用户参与度的深入将成为 Web 发展的主要方向。

(2) 社群 (Community)

社群经营的理念早在 Web1.0 时代就有,通常以加入会员然后共享部分特定资源为主要经营方式,如公告栏、讨论版、聊天室、下载空间等。但 Web2.0 的经营方式则更为多元且活泼,在各种服务上都对社会网络的功能进行强化。例如,各类交友空间中大量建立人际连接的节点,不论是现实中的人脉、书签、相簿、地理位置甚至共同爱好等都可以成为建立人际网络的起点。通过大量的社会网络功能,促进分享也强化服务对使用者的粘性。Web2.0 的发展将会使网络上人们之间的联系聚合得更为紧密。

(3) 消费者使用体验 (Customers Experience)

良好的使用者体验有助于降低学习门槛,提升使用者对服务的接受度,并在强调以使用者为中心的 Web2.0 概念实践过程当中,加深用户的参与度。如在 Web1.0 中使用者想要描述地点位置时,往往只能写地址,并用文字极力描述附近的地标,这样的定位方式不仅沟通困难而且容易出错。Web2.0 出现后,伴随着 Google Maps 等地图的流行,使用者能够直接在地图上点选地点,大大降低了使用上的负担,也造就了如 Platial.com 等由使用者共同创造生活地图或景点指引的服务。

除了前述有关地图界面造成的使用体验差异之外,地图服务自身界面的进步其实也是极具代表性的。过去的地图服务在界面设计上往往局限于所谓的“8 个小箭头”的设计,使用者必须通过上、下、左、右、左上、右上、左下、右下 8 个箭头来操作地图界面。而现在的地图服务,如 Google Maps 等,几乎都提供了直接以鼠标拖拉的操作界面。表面上差异似乎不大,但在操作体验上却截然不同。

因此,如何得到用户的认可,如何改善用户体验,使用户在使用网站时感觉到舒适和方便,将是 Web2.0 发展过程中需要不断研究和不断改进的一个方向。

(4) 设计跨服务的互动与整合 (Cross-Service Integration)

伴随着技术的进步和成熟,以及经营理念的改变,Web2.0 时代的服务借助开放界面来带动社群与第三方开发者的投入,发挥 Web2.0 服务的网络效应。同时将这些开发者所创造的各种 Web 应用相结合,使用混搭这样的新开发风格以提供更为丰富的服务。典型的案例如将 Craigslist 上的租屋/售屋信息与 Google Maps 提供的地图服务整合而成的 Housing Map 地图房地产服务。

1.1.4 Web 2.0 的相关应用

从2003年开始, Web2.0代表着互联网发展的新理念,逐渐成为该领域发展的热点和主流。Web2.0倡导用户主导、用户参与、用户分享、用户创造,最大限度地帮助用户实现个性化生产和满足用户个性化需求是Web2.0服务的中心内容。围绕上述中心, Web2.0提供了非常丰富的应用,这些应用正在改变在互联网上生成、共享和分发信息的传统概念。下面列举几种较为典型的应用。

(1) 博客 (BLOG)

BLOG也就是Web LOG的缩写,简单来说就是网络日记。一种极其简易便捷的网络个人出版形式,使得任何一位网民都可以在几分钟之内拥有自己的个人网站,自由挥写自己的观点。

随着博客的普及,人们对于博客的理解也是千姿百态,出现了如下列举的一系列声音:

- 1) 博客代表着“新闻媒体3.0”,旧媒体→新媒体→互媒体。
- 2) 博客是继E-mail、BBS、ICQ(IM)之后的第4种网络交流方式。
- 3) 博客是互联网上独立的思想。
- 4) 博客是媒体的开放源代码运动。
- 5) 博客是“个人出版2.0”。
- 6) 博客是用文字进行对话的网上咖啡屋。

综上所述,通过BLOG,互联网成为个人信息的集散地。BLOG也是网络发展竞相追逐的目标,其兼有新媒体和新技术的优势,可以有效地构筑网络社团,形成“圈子化”的社区集群。博客让每个人都可以成为互联网中自主的主体表达者,并且与网络世界建立全面的交流。读者可以到当前用户访问量较多的新浪博客社区,表达自己的观点、展示自己的文学才华,体验一回当博主的感觉,如新浪博客网址为<http://blog.sina.com.cn/>。

(2) 播客 (Podcast)

播客这个词来源于苹果电脑的iPod与广播(broadcast)的合成词,其指的是一种在互联网上发布文件并允许用户订阅feed以自动接收新文件的方法,或用此方法来制作的电台节目。这种新方法在2004年下半年开始在互联网上流行,以用于发布音频文件。

播客与其他音频内容传送的区别在于其订阅模式,它使用RSS 2.0文件格式传送信息。该技术允许个人进行创建与发布,这种新的传播方式使得人人可以说出他们想说的话。读者可以访问网址<http://you.video.sina.com.cn/>,过一把播客的瘾。

(3) 标签 (Tag)

标签是一种新的组织和管理在线信息的方式,它不同于传统的、针对文件本身的关键字检索,而是一种模糊化、智能化的分类。

标签给用户带来了灵活性,在每篇日志、每个帖子或者每张图片等对象中添加一个或多个标签,并且通过标签可以使用户之间产生更多的联系。标签体现了群体的力量,使内容之间的相关性和用户之间的交互性大大增强。

如当前比较流行的搜索引擎中就使用了标签技术。

(4) 社会性网络软件 (SNS)

SNS (Social Network Software) 来自于美国哈佛大学心理学教授 Stanley Milgram 的“六度分离”理论, 简单地说就是你和任何一个陌生人之间所间隔的人不会超过 6 个。按照“六度分离”理论, 个体的社交圈都呈不断放大的趋势, 最后成为一个大型网络。根据这种理论, 后期便有人创立了面向社会性网络的互联网服务, 通过“熟人的熟人”来进行网络社交拓展。

在经济高速发展的今天, 人际关系的作用越发明显, 人脉资源对于一个人的成功非常重要。目前很多职业人士在人际资源方面其实都比较匮乏, 因此 SNS 这种网络社交方式一出现, 便迅速流行于欧美国家, 已经成为精英阶层拓展人际关系的主要方式之一。

Facebook 是起源于美国的一个社交网络服务网站, 于 2004 年 2 月 4 日上线。Facebook 是美国排名第一的照片分享站点, 每天上载 850 万张照片。随着用户数量增加, Facebook 的目标已经指向另外一个领域: 互联网搜索。2012 年 2 月 1 日, Facebook 正式向美国证券交易委员会 (SEC) 提出首次公开发行 (IPO) 申请, 目标融资规模达 50 亿美元, 并聘请摩根士丹利、高盛和摩根大通为主要承销商。这将是硅谷有史以来规模最大的 IPO。2012 年 5 月 18 日, Facebook 正式在美国纳斯达克证券交易所上市。2012 年 6 月, Facebook 称将涉足在线支付领域

(5) 聚合技术 (RSS)

RSS 就是一种简单的信息发布和传递方式, 使得一个网站可以方便地调用其他提供 RSS 订阅服务的网站的内容, 从而形成“新闻聚合”, 让网站发布的内容在更大的范围内传播。

RSS 是基于 XML 标准, 用以实现站点之间共享内容的数据交换规范, 这一技术起源于美国网景 (Netscape) 通信公司。目前对这一技术有 3 种解释:

- 1) Really Simple Syndication——真正简单聚合。
- 2) RDF——资源描述架构站点摘要。
- 3) Rich Site Summary——丰富站点摘要。

其实 3 种解释本质是一致的, 都是指聚合技术。如果从 RSS 阅读者的角度来看, RSS 获取信息的模式与加入邮件列表获取信息有一定的相似之处, 也就是可以不必登录各个提供信息的网站而通过客户端浏览方式 (称为“RSS 阅读器”) 或者在线 RSS 阅读方式阅读这些内容。

(6) 维客 (Wiki)

Wiki 是 Ward Cunningham 于 1995 年所创的一种可在网络上开放多人协同创作的超文本系统, Wiki 一词来源于夏威夷语的 wee kee wee kee, 原本是“快点”的意思。Wiki 包含一套能简易创造、改变 HTML 网页的系统, 容许任何造访网站的人能快速轻易地加入、删除、编辑所有的内容, 因此特别适合团队合作的写作方式。作为一种知识网络系统, Wiki 简便开放的特点使互联网使用者在一个社群内共享某个领域的知识, 这一特性淋漓尽致地体现在著名的开放式百科网站维基百科 (如中文维基百科网站 <http://zh.wikipedia.org/>) 上, 读者可以自己体会一下维客的魅力。Web2.0 丰富的应用也需要先进的技术支持, 如 Web2.0 所用到的核心技术——Ajax 就是实现众多应用的保证。那么 Ajax 究竟是一项什么样的技术? 笔者将在 1.2 节进行介绍。

1.2 JSP 运行环境

JSP 运行环境有 2 个基本条件：①需要在计算机上安装 Java 2，并进行相关的环境变量的设置；②需要在计算机上安装 JSP 引擎，比如 JavaEE 服务器、Resin 和 Tomcat 服务器等。

Tomcat 作为 Web 服务器，有三种方案：①J2SDK + Tomcat；②J2SDK + Apache + Tomcat；③J2SDK + IIS + Tomcat。

1.2.1 JDK 的下载与安装

JDK 可以通过 oracle 公司网站提供的下载地址进行免费下载，如下载 Java SE 的版本的网址：<http://www.oracle.com/technetwork/java/javase/downloads/index.html>。目前最新的版本是 Java SE 7u17，下载之前要选择运行的操作系统平台，如图 1-4 所示。

Java SE Development Kit 7u17		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
☐ Accept License Agreement ☒ Decline License Agreement		
Product / File Description	File Size	Download
Linux x86	106.65 MB	jdk-7u17-linux-i586.rpm
Linux x86	92.97 MB	jdk-7u17-linux-i586.tar.gz
Linux x64	104.78 MB	jdk-7u17-linux-x64.rpm
Linux x64	91.71 MB	jdk-7u17-linux-x64.tar.gz
Mac OS X x64	143.78 MB	jdk-7u17-macosx-x64.dmg
Solaris x86 (SVR4 package)	135.39 MB	jdk-7u17-solaris-i586.tar.Z
Solaris x86	91.67 MB	jdk-7u17-solaris-i586.tar.gz
Solaris SPARC (SVR4 package)	135.92 MB	jdk-7u17-solaris-sparc.tar.Z
Solaris SPARC	95.32 MB	jdk-7u17-solaris-sparc.tar.gz
Solaris SPARC 64-bit (SVR4 package)	22.97 MB	jdk-7u17-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	17.69 MB	jdk-7u17-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	22.61 MB	jdk-7u17-solaris-x64.tar.Z
Solaris x64	15.02 MB	jdk-7u17-solaris-x64.tar.gz
Windows x86	88.75 MB	jdk-7u17-windows-i586.exe
Windows x64	90.42 MB	jdk-7u17-windows-x64.exe

图 1-4 选择下载的操作系统平台

下载完毕后，如直接运行所下载的 jdk-7u17-windows-i586.exe，提示进行安装，如图 1-5 所示的 JDK 安装后目录。



图 1-5 JDK 安装后目录

1.2.2 安装与配置 Tomcat

Tomcat 服务器是当今使用最广泛的 Servlet/Jsp 服务器，它运行稳定、性能可靠，是学习 JSP 技术和中小型企业应用的最佳选择。下载的网址：<http://tomcat.apache.org/>，目前最新的版本是 Apache Tomcat 7。图 1-6 所示为 Tomcat 下载界面。

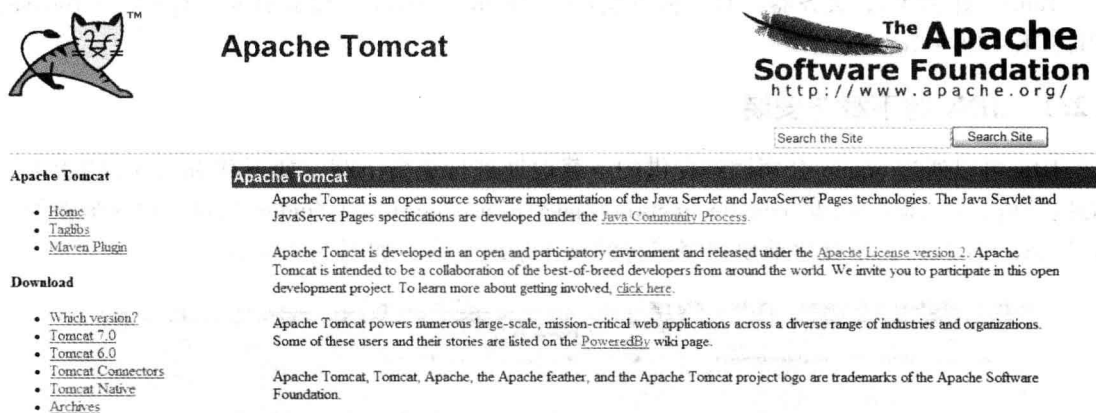


图 1-6 Tomcat 下载界面

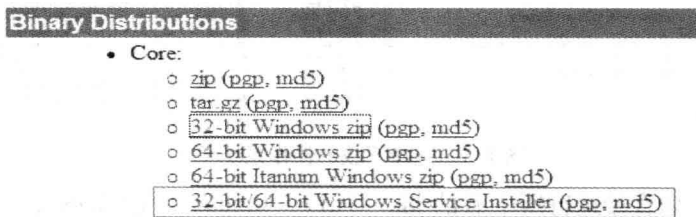


图 1-7 选择下载的文件

选择下载版本 Tomcat 6，选择下载的文件，进入图 1-7 所示的下载页面。选择下载 Windows 环境下的程序包 `apache - tomcat - 6.0.36.exe`。运行 `apache - tomcat - 6.0.36.exe`，按照提示安装。

1.2.3 启动与停止 Tomcat

Tomcat 安装以后，应启动 JSP 才能运行，在 JSP 动态网页修改以后，需重新启动 Tomcat，Tomcat 服务器的启动和停止如图 1-8 所示。

1.2.4 案例 1：一个简单的 JSP 程序

进入 Tomcat 的安装目录的 `webapps` 目录，在 `webapps` 目录下新建一个目录，起名叫 `myapp`（见图 1-9），在 `myapp` 下新建一个目录 `WEB-INF`（注意，目录名称是区分大小写的），`WEB-INF` 下新建一个文件 `web.xml`，在 `myapp` 文件夹建立文件 `index.jsp`，文件内容如下：
`index.jsp` 文件内容

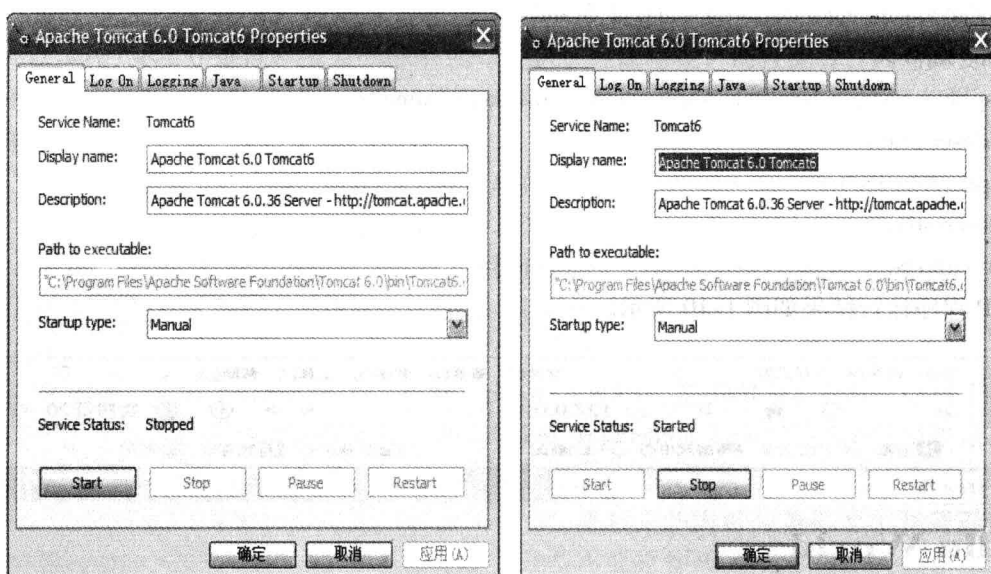


图 1-8 Tomcat 服务器的启动与停止

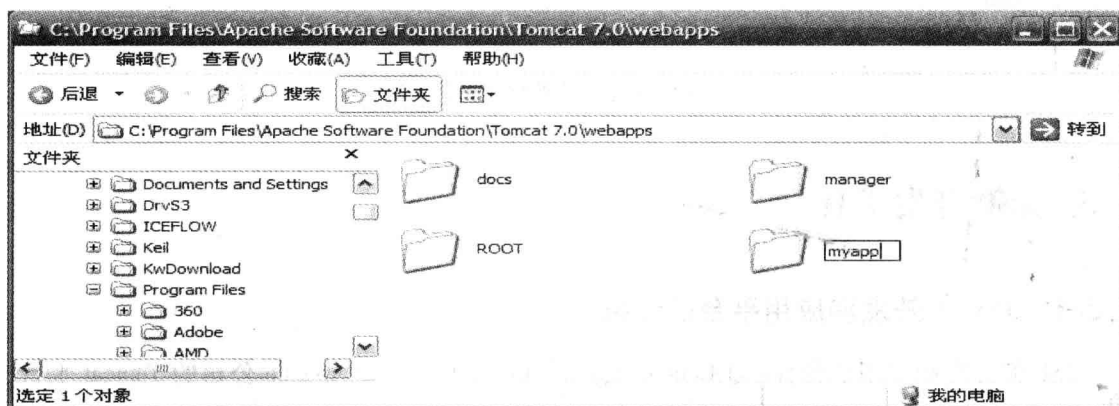


图 1-9 Tomcat 的安装目录 webapps 目录下新建一个目录 myapp

```
<html>
  <head> <title>FirstJsp</title> </head>
  <body>
    <h1> <% out.println("Hello World");% > </h1>
    <h2> <% out.println("This is the first JSP program!");% > </h2>
  </body>
</html>
```

文件 web.xml 内容:

```
<? xml version = "1.0" encoding = "ISO - 8859 - 1" ? >
<! DOCTYPE web - app PUBLIC " - //Sun Microsystems, Inc. //DTD Web Application 2.3//EN"
```