



重点大学 计算机基础课程教材

动态网页开发技术

陈旭东 张宏勋 编著



清华大学出版社 • 北京交通大学出版社

重点大学计算机基础课程教材

动态网页开发技术

陈旭东 张宏勋 编著

清华大学出版社

北京交通大学出版社

·北京·

内 容 简 介

本书全面介绍了 JSP 2.0 规范的相关技术。书中所有知识点全部采用实例来进行说明。

全书共分为 12 章, 前 7 章介绍 JSP 应用开发的基本技术, 包括网页开发技术、JSP 开发环境、HTML、JSP 基本元素、JSP 应用开发调试过程、JSP 页面对象、表达式语言、JSTL 等内容; 后 5 章着重介绍 JSP 2.0 的高级应用技术, 包括以 SQL 标记库、XML 标记库、JavaBeans 组件技术、自定义标记库、Web 应用架构及 Web 应用发布等内容。

本书可以作为高等院校网页开发技术课程和培训用的教学用书, 也可以作为 JSP 页面开发人员的参考用书。

版权所有, 翻印必究。举报电话: 010-62782989 13501256678 13801310933

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。

本书防伪标签采用特殊防伪技术, 用户可通过在图案表面涂抹清水, 图案消失, 水干后图案复现; 或将表面膜揭下, 放在白纸上用彩笔涂抹, 图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

动态网页开发技术 / 陈旭东, 张宏勋编著. —北京: 清华大学出版社; 北京交通大学出版社, 2005.10

(重点大学计算机基础课程教材)

ISBN 7-81082-613-1

I. 动… II. ①陈… ②张… III. JAVA 语言-主页制作-程序设计-高等学校-教材 IV. TP393.092

中国版本图书馆 CIP 数据核字 (2005) 第 100889 号

责任编辑: 谭文芳

出 版 者: 清华大学出版社 邮编: 100084 电话: 010-62776969 <http://www.tup.com.cn>
北京交通大学出版社 邮编: 100044 电话: 010-51686414 <http://press.bjtu.edu.cn>

印 刷 者: 北京东光印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印张: 22.25 字数: 570 千字

版 次: 2005 年 10 月第 1 版 2005 年 10 月第 1 次印刷

书 号: ISBN 7-81082-613-1/TP·229

印 数: 1~5000 册 定价: 29.00 元

本书如有质量问题, 请向北京交通大学出版社质监局反映。对您的意见和批评, 我们表示欢迎和感谢。

投诉电话: 010-51686043, 51686008; 传真: 010-62225406; E-mail: press@center.bjtu.edu.cn。

《重点大学计算机基础课程教材》

编 委 会

(排名不分先后)

吴文虎 (清华大学)

黄刘生 (中国科学技术大学)

叶晓风 (南京大学)

阮秋琦 (北京交通大学)

谢柏青 (北京大学)

郑 骏 (华东师范大学)

施伯乐 (复旦大学)

管会生 (兰州大学)

钱 能 (浙江工业大学)

谢步赢 (同济大学)

朱 敏 (东南大学)

汪 卫 (复旦大学)

杨小平 (中国人民大学)

李丽娟 (湖南大学)

王立福 (北京大学)

何炎祥 (武汉大学)

王行恒 (华东师范大学)

马建峰 (西安电子科技大学)

衷克定 (北京师范大学)

薛永生 (厦门大学)

出版说明

进入 21 世纪,随着国家信息化步伐的加快及各行业信息化进程的不断加速,社会对专业(非计算机专业)人才的信息技术能力要求越来越高。为了适应社会对专业人才的要求,全国各高校在重视专业知识培养的同时也非常注重计算机应用能力的训练,即信息技术能力的培养。计算机应用水平已成为衡量高校毕业生综合素质的突出标志之一。

为此,各高校加大了使用计算机科学等现代科学技术提升、改造传统学科专业的力度,从而实现传统学科专业向现代信息社会学科专业的发展与转变。在发挥传统学科专业师资力量强、办学经验丰富、教学资源充裕等优势的同时,不断更新其教学内容、改革课程体系,使学科专业的教育与社会信息化发展趋势相适应。计算机基础课程教学在改造传统学科向现代信息社会学科转变中起到了至关重要的作用,学科专业中的计算机基础课程设置、内容体系和教学手段及方法等也具有不同于以前传统学科的鲜明特点。

为了配合各高校现代学科专业(非计算机专业)的建设和发展,急需出版一批内容新、体系新、方法新、手段新的高水平计算机基础课程教材。但是计算机基础教育的发展只有短短的二十多年时间,其覆盖的专业门类繁多,涉及的学校类型各异,不同的高校在开展计算机基础教育时还存在各自的认识。目前,非计算机专业的计算机课程教材的建设工作仍滞后于教学改革与实践,如:现有的计算机课程教材中有不少内容陈旧,重理论、轻实践,不能满足教学计划及课程设置的需要;一些课程的教材可供选择的品种太少;一些基础课的教材虽然品种较多,但低水平重复严重;有些教材内容庞杂,书越编越厚;专业课教材、教学辅助教材及教学参考书短缺,等等。这些都不利于学生自学能力的提高和全面素质的培养。可见,高等学校计算机基础教育和教材建设正面临新的形势和任务。

重点大学的教学与科研氛围是培养面向信息社会一流专业人才的基础,其中教材的使用和建设则是这种氛围的重要组成部分,一批具有特色优势的非计算机专业的计算机教材作为各重点大学的重点建设项目成果得到肯定。为了展示和发扬各重点大学在非计算机专业上计算机教育的优势,同时以教材展示各重点大学的优秀教学理念、教学方法、教学手段和教学内容等,在相关教学指导委员会专家的指导和建议下,我们规划并组织出版了本系列教材,以满足非计算机专业计算机课程教学的需要。

本系列教材在规划过程中体现了如下一些基本组织原则和特点。

一、强调应用。本系列教材面向非计算机专业学生,从应用目的出发,强调计算机在各专业中的应用。在教材内容上坚持基本理论适度,反映基本理论和原理的综合应用,强调实践和应用环节。

二、内容新颖。计算机科学和技术的发展日新月异,本系列教材力求介绍这一领域的新技术、新发展,放弃对一些过时的概念和使用价值较小的技术的介绍。教材涉及的计算机软件应具有典型性,在保持通用性的前提下介绍最新版本的特点。

三、体现案例教学。在兼顾基础性和系统性的前提下,重视教材内容的案例编排,力求从内容和结构上突出案例教学的要求,以适应教师指导下学生自主学习的教学模式。

四、实施精品战略,突出重点,保证质量。本系列教材规划的重点在公共基础课和专业基础课的教材建设;特别注意选择并安排了一部分原来基础比较好的优秀教材或讲义修订出版,力求逐步形成精品教材;鼓励教师编写体现专业计算机教学内容和课程体系改革成果的教材。

五、依靠一线教师,择优落实。本系列教材的作者全部来自全国各重点大学的一线授课教师。在落实选题和作者时,引入竞争机制,通过申报和进行严格评审后再进行确定。书稿完成后认真实行审稿程序,确保出书质量。

计算机科学与技术的发展突飞猛进,本系列教材也应动态发展。在教材使用过程中,希望广大的读者积极地向我们提出意见与建议,我们将及时改正和更新。

《重点大学计算机基础课程教材》编委会
2005 年 7 月

前 言

最近几年来,Java 技术已经成为 Web 应用的主要开发平台之一,而 Java Server Pages (JSP)技术在 Web 开发中应用也越来越广泛。2003 年 11 月发布的 JSP 2.0 规范大大地减少了 JSP 页面开发对 Java 程序设计的依赖,使用最新的 JSP 2.0 技术,非 Java 程序员也可以很容易地设计出功能强大的动态网页应用。

本书利用 Tomcat 5.5 作为 Web 应用开发调试平台,全面介绍 JSP 2.0 的相关技术。全书可分为三个部分,共 12 章。

第一部分是 Web 应用开发的基础知识,包括第 1 章到第 3 章的内容。第 1 章是网页开发技术介绍;第 2 章讲述如何搭建 JSP 开发环境;第 3 章以实例介绍超文本标记语言 HTML。

第二部分是 JSP 2.0 开发基础,包括第 4 章到第 7 章的内容。第 4 章介绍 JSP 基本概念、基本元素及 JSP 应用开发调试过程;第 5 章介绍 JSP 页面中使用的对象;第 6 章介绍表达式语言的使用;第 7 章介绍 JSP 标准标记库 JSTL 基本功能的使用。

第三部分是 JSP 2.0 高级开发,包括第 8 章到第 12 章的内容。第 8 章以 MySQL 数据库为例,讲述在 JSP 2.0 中以 SQL 标记库和 JDBC 访问数据库的两种不同方式;第 9 章介绍在 JSP 页面中使用 XML 标记库来访问和操作 XML 数据;第 10 章介绍 Java 组件技术 JavaBeans 在 JSP 中的应用;第 11 章介绍自定义标记库的使用,特别是 JSP 2.0 新引入的标记文件和简单标记的应用;第 12 章则是从 Web 应用架构的角度,介绍 JSP 开发的两种模型,并结合 Servlet 技术介绍 MVC 模式的 Web 应用开发实例,另外对 Web 应用的发布也做了简单的介绍。

全书通过大量实例介绍 JSP 2.0 的基本技术,通过本书的学习可以快速掌握基于 JSP 2.0 的 Web 应用开发新技术。书中的实例程序全部都经过作者的调试运行,读者可以直接使用这些程序代码。书中涉及的所有实例的完整源代码可以从北京交通大学出版社的网站下载:
<http://press.bjtu.edu.cn>。

在本书编写过程中,得到陈雨杨和杨冰的全力支持,在此表示衷心感谢。另外,在选题、文字校对和代码调试等方面还得到了马迪芳、张宏勋、徐保明、戴刚、冯凤娟、刘迪仁、刘春国、陈方红等老师的大力支持,研究生崔贺超和王平进行了细致的文字校对,在此一并表示谢意。

本书第 1 章和第 2 章由张宏勋编写,第 3 章到第 12 章由陈旭东编写,全书由陈旭东统稿。由于作者水平所限,难免会存在很多不足和错误,恳请各位读者不吝指教。作者的电子邮箱为:xudong_chen@tom.com。

陈旭东
2005 年 8 月

目 录

第 1 章 网页开发技术简介	1
1.1 Web 技术简介	1
1.1.1 Web 发展历史	1
1.1.2 Web 工作原理	1
1.1.3 静态网页和动态网页	2
1.2 网页开发技术	3
1.2.1 Web 客户端技术	3
1.2.2 Web 服务端技术	4
1.2.3 Web 服务端开发平台	6
1.2.4 XML 和 Web Service	6
1.2.5 Web 开发技术的未来	7
1.3 JSP 动态网页开发技术	8
1.3.1 JSP 简介	8
1.3.2 JSP 和 ASP 比较	8
小结	9
习题	9
第 2 章 搭建 JSP 开发平台	10
2.1 所需软件	10
2.2 下载和安装 J2SE SDK	10
2.2.1 下载 JDK 5.0	10
2.2.2 安装 JDK 5.0	11
2.3 下载和安装 Tomcat	13
2.3.1 下载 Tomcat 5.5	13
2.3.2 安装 Tomcat 5.5	14
2.4 配置开发环境	17
2.5 测试开发平台是否工作	18
2.6 构造第一个 JSP 程序	19
2.6.1 构造应用	19
2.6.2 基本工作原理	20
小结	21
习题	21
第 3 章 HTML 概述	22
3.1 HTML 基础	22
3.1.1 HTML 标记	22

3.1.2	HTML 页面一般格式	23
3.1.3	HTML 标记的属性	24
3.1.4	文字布局	25
3.1.5	表格(table)	32
3.1.6	样式(style)	35
3.2	HTML 表单	46
3.2.1	表单属性	47
3.2.2	控件	48
3.2.3	完整表单实例	53
3.3	HTML 框架	55
3.3.1	框架集	57
3.3.2	框架	60
3.4	HTML 实例	62
小结		64
习题		64
第 4 章	JSP 基础	65
4.1	JSP 概念	65
4.1.1	JSP 文件命名规则	66
4.1.2	JSP 工作原理	66
4.1.3	JSP 空元素	68
4.1.4	属性值	68
4.1.5	空白	68
4.1.6	注释	68
4.2	JSP 基本元素	71
4.2.1	指令元素	71
4.2.2	行为元素	76
4.2.3	脚本元素	84
4.3	引用转换	88
4.4	JSP 出错处理	90
4.4.1	编译错	90
4.4.2	运行时错	90
4.4.3	使用 JSP 页面处理错误	91
4.5	JSP 简单应用实例	93
小结		97
习题		98
第 5 章	JSP 对象	99
5.1	有效范围	99
5.1.1	页内有效	99
5.1.2	请求有效	102
5.1.3	会话有效	103

5.1.4	应用有效	105
5.1.5	属性操作	107
5.2	隐含对象	107
5.2.1	request	107
5.2.2	response	111
5.2.3	out	113
5.2.4	session	115
5.2.5	application	117
5.2.6	pageContext	119
5.2.7	config	121
5.2.8	page	124
5.2.9	exception	125
5.3	Cookie 对象	127
5.4	JSP 对象应用实例	130
5.4.1	request 对象实例:用户注册	130
5.4.2	session 对象实例:用户登录	133
5.4.3	application 对象实例:简单聊天室	137
小结		139
习题		139
第 6 章	表达式语言	140
6.1	EL 特点	140
6.2	EL 语法	141
6.2.1	EL 变量范围	141
6.2.2	EL 保留关键字	143
6.2.3	访问运算符“.”和“[]”	143
6.2.4	算术运算	145
6.2.5	关系运算	147
6.2.6	逻辑运算	150
6.2.7	empty 运算符	152
6.2.8	自动类型转换	153
6.3	EL 隐含对象	156
6.3.1	pageContext 对象	156
6.3.2	param 和 paramValues	156
6.3.3	header 和 headerValues	156
6.3.4	cookie	157
6.3.5	initParam	157
6.3.6	属性范围	157
6.3.7	EL 隐含对象的使用实例	157
6.4	EL 函数	160
6.4.1	简单 EL 函数	161

6.4.2 复杂 EL 函数	163
6.4.3 标记属性中使用函数	166
6.4.4 函数的嵌套	168
小结	168
习题	169
第 7 章 JSTL 概述	170
7.1 JSTL 基础	170
7.1.1 JSTL 起源	170
7.1.2 JSTL 的安装和使用	170
7.1.3 JSTL 示例应用	173
7.2 核心标记库	174
7.2.1 对象操作	174
7.2.2 条件处理	181
7.2.3 循环和迭代	185
7.2.4 URL 操作	189
7.3 格式标记库	195
7.3.1 I18N 标记	195
7.3.2 数字、日期格式化	207
7.4 函数标记库	215
小结	220
习题	220
第 8 章 访问数据库	221
8.1 数据库和 SQL 语言	221
8.1.1 数据库	221
8.1.2 SQL 简介	221
8.2 安装和配置 MySQL	223
8.2.1 下载和安装 MySQL	223
8.2.2 配置 MySQL 数据库	225
8.2.3 创建数据库	230
8.2.4 创建用户	231
8.2.5 创建表和数据	232
8.2.6 安装和配置 JDBC 驱动	232
8.3 使用 SQL 标记库访问数据库	233
8.3.1 设置数据源	233
8.3.2 查询数据库	234
8.3.3 修改数据库	237
8.3.4 数据库操作实例	241
8.4 使用 JDBC 访问数据库	244
8.4.1 JDBC 简介	244
8.4.2 JSP 中使用 JDBC	244

小结.....	246
习题.....	247
第 9 章 操作 XML 数据	248
9.1 XML 简介.....	248
9.1.1 XML 基本概念	248
9.1.2 XPath	251
9.2 XML 标记库.....	252
9.2.1 XML 标记中使用 XPath	253
9.2.2 XML 核心标记	253
9.2.3 XML 流程控制标记	259
9.2.4 XML 转换	265
小结.....	270
习题.....	270
第 10 章 使用 JavaBeans 组件	272
10.1 JavaBeans	272
10.1.1 JavaBean 简介	272
10.1.2 创建 JavaBean	273
10.2 使用 JavaBean	275
10.2.1 使用 JavaBean 的配置	275
10.2.2 <jsp:useBean>标记.....	275
10.2.3 <jsp:setProperty>标记	277
10.2.4 <jsp:getProperty>标记	279
10.2.5 使用 JavaBean 获取页面提交信息	280
10.3 在 EL 表达式中使用 JavaBean	281
10.4 使用 JavaBean 访问数据库	284
10.4.1 访问数据库的 JavaBean	284
10.4.2 使用 JavaBean 来访问数据库	286
10.5 自动类型转换.....	290
小结.....	290
习题.....	290
第 11 章 自定义标记	291
11.1 自定义标记库.....	291
11.1.1 必要性.....	291
11.1.2 标记和标记库	291
11.1.3 使用自定义标记	292
11.1.4 JavaBeans 与自定义标记	292
11.2 标记文件.....	293
11.2.1 内容重用	293
11.2.2 自定义模板	296
11.2.3 标记文件中使用变量	301

11.3 简单标记	309
11.3.1 SimpleTag 接口	309
11.3.2 简单标记的基本生命周期	309
11.3.3 SimpleTagSupport 类	311
11.3.4 简单标记实例	311
11.4 标记库重用	314
小结	318
习题	318
第 12 章 构建 Web 应用	319
12.1 Web 应用的体系结构	319
12.1.1 模型 1	319
12.1.2 模型 2	320
12.1.3 MVC 模型	321
12.2 Servlet 基础	322
12.2.1 Servlet 简介	322
12.2.2 Servlet 的使用	323
12.2.3 Servlet 的主要方法	326
12.3 MVC 实例:模型 2	327
12.3.1 控制器:Servlet	328
12.3.2 模型:JavaBean	330
12.3.3 视图:JSP	332
12.3.4 配置和运行实例	333
12.4 Web 应用的发布	335
12.4.1 应用打包	335
12.4.2 应用安装	335
12.4.3 WAR 配置选项	336
小结	339
习题	340
参考文献	341

第 1 章 网页开发技术简介

本章主要介绍 Web 技术的发展历史和 Web 的基本工作原理,对客户端和服务端的 Web 开发技术进行简要描述,对两个流行的 Web 应用开发平台 J2EE 和 .NET 进行比较,对 JSP 技术和 ASP 技术也进行简单的对比。

1.1 Web 技术简介

1.1.1 Web 发展历史

随着计算机技术的发展,计算机数量越来越多也越来越分散。20 世纪 60 年代,美国军方建立了一个计算机网 ARPANET,用于连接全国各地的重要计算机。之后,美国的科研机构也连接到该网,ARPANET 发展成为科学研究和协作网络。后来,由于它连接了众多的局域网和广域网,被改名为 Internet。

在 Internet 早期,在科研机构之外很少有人知道它的存在。当时,使用 Internet 非常不方便,第一代 Internet 用户都是使用命令行工具如 Telnet、FTP 和 Gopher 等来获取信息。

1989 年,欧洲原子能研究组织(European Organization for Nuclear Research,CERN)的计算机专家 Tim Berners-Lee 提出了“World Wide Web”的概念,构想在 Internet 之上构建一个交互的超文本(Hypertext)系统以方便世界各地的物理学家的交流。超文本指的是文档中的某个词可以链接到其他文档或文档中的某个位置。1990 年 11 月,第一个 Web 服务器 nxoc01.cern.ch 开始运行,Tim Berners-Lee 在自己编写的图形化 Web 浏览器上看到了最早的 Web 页面。1991 年,CERN 正式发布了 Web 技术标准。之后出现了一些界面友好的 Internet 工具,如 HyperCard、NeXT 等。到 1993 年,Internet 上大约有 50 台 Web 服务器。目前,与 Web 相关的各种技术标准都由著名的 W3C 组织(World Wide Web Consortium)管理和维护。

1993 年,Illinois 大学(University of Illinois)的美国国家超级计算应用中心(National Center for Supercomputing Applications,NCSA)发布了支持 UNIX、PC 和 Macintosh 系统的 Mosaic Web 浏览器,极大地推动了 Internet 的发展。1994 年,随着浏览 Internet 的公众人数的增多,Web 进入大众文化的商机凸现,Mosaic 的部分开发人员在硅谷成立了一个小公司,后来成为著名的 Netscape。Netscape 浏览器为 Internet 和 Web 技术飞速发展奠定了基础。

关于 Internet 历史,请参阅 <http://www.isoc.org/internet/history/>。关于 Web 的历史,请参阅 <http://www.w3.org/History.html>。

1.1.2 Web 工作原理

Internet 是物理上的计算机网络,连接世界各地的计算机。而 Web 是一种基于 Internet 的服务,允许计算机之间的相互通信。除了 Web 服务外,Internet 还提供电子邮件、视屏流、多人联机游戏等服务。

Tim Berners-Lee 说:“Web 是一个抽象的信息空间。”也就是说,作为 Internet 上的一种应用架构,Web 的首要任务就是向人们提供信息服务。从技术层面看,Web 有三个主要优点:用超文本技术 HTML(Hypertext Markup Language,超文本置标语言)实现信息与信息的连接;用统一资源定位技术 URL(Uniform Resource Locator,统一资源定位地址)实现全球信息的精确定位;用新的应用层协议 HTTP(Hypertext Transfer Protocol,超文本传输协议)实现分布式的信息共享。

Web 服务定义了使用 Internet 通信的两个部分:Web 客户端(也称 Web 浏览器)和 Web 服务器。当访问一个 Web 站点时,就会在浏览器和 Web 服务器之间建立了关联,通过消息交换来进行通信。这种关联被称为请求-响应模型,请求和响应使用 HTTP 协议交互。

一般情况下,浏览器可以是非常小的轻量级的应用程序,它为用户提供 Web 页面显示和导航等功能。Web 页面数据可以包含普通文本、HTML、图片等,浏览器将这些数据以易于理解和易于交互的方式展现给用户。Web 页面数据之间如果存在交互链接,由浏览器生成页面文档间的链接。Web 服务器则是一个相对复杂的应用程序,负责定位浏览器请求的 Web 页面数据,将数据打包并发送回浏览器。

HTTP 协议定义了两台远程计算机之间通信时交换的消息内容的格式。浏览器发送的请求被称为 HTTP 请求,服务器返回的响应被称为 HTTP 响应。

请求-响应模型的工作原理为:首先,浏览器发送消息到 Web 服务器,请求特定的 Web 页面;Web 服务器响应浏览器请求,查找该页面;如果该页面存在,Web 服务器就会发送回该页面的 HTML 消息内容。每个被浏览的 Web 页面都是 Web 浏览器发送请求、Web 服务器响应并返回的对应消息。

浏览器的 HTTP 请求可以是直接输入 URL 地址,也可以是单击某个链接,甚至可以是自动执行的请求(如显示页面内的图片)。但是,不管浏览器的请求是如何生成的,服务器都会接收到一个对应的有固定格式的 HTTP 请求。

Web 服务器的 HTTP 响应的实现也有多种情形。最简单的情况是浏览器请求具体的一个 Web 页面文件,这种情况下,在 URL 和文件间一般存在对应关系,如 URL `http://www.helloBeijing2008.org/bj/welcome.html` 可能对应域名为 `www.helloBeijing2008.org` 的服务器上具体的文件 `C:\webroot\bj\welcome.html`。

1.1.3 静态网页和动态网页

Web 页面也就是通常所说的网页。

如果对所有用户请求都返回同样的页面内容,这样的 Web 页面称为静态网页。静态网页由于内容固定不变,一般以 HTML 文件方式存储。静态页面修改时,只需使用文本编辑器或 HTML 编辑器编辑 HTML 文件即可。

除了静态页面,许多 Web 页面是依据时间、用户的输入或参数而动态生成的 HTML 文档,网页的内容是动态变化的,因此被称为动态网页。动态网页主要用于数据经常改变、需要用户输入信息等情形。例如:股票价格查询 Web 站点可能有很多用户在线,假如 Web 服务器只能发送静态页面文件,Web 站点必须为每支股票单独建立一个页面文件,这样会需要大量的文件,而且考虑到股票价格的变动,对这些文件的实时更新也不可能完成。

动态处理和产生 HTML 页面的技术(如 CGI、ASP、JSP 等)称为动态网页技术。这时 Web 服务器不是从文件读取页面而是运行应用程序生成页面。

1.2 网页开发技术

Web 应用中的每次信息交换都要涉及客户端和服务端两个层面。因此,Web 开发技术大体上可以分为客户端和服务端两大类技术。

1.2.1 Web 客户端技术

Web 客户端技术是指浏览器端支持的页面技术。这里介绍构成 Web 页面的 HTML 语言及相关的动态页面技术。

1. HTML 语言

1945 年,Vannevar Bush 在他的文章中阐述了文本和文本之间通过超级链接而相互关联的思想,并给出了一种能实现信息关联的计算机 Memex 的设计方案。在 1960 年前后,Doug Engelbart 等人对信息关联技术进行了最早的实验。与此同时,Ted Nelson 正式将这种信息关联技术命名为超文本技术。

1969 年,IBM 的 Charles Goldfarb 发明了可用于描述超文本信息的 GML (Generalized Markup Language,通用标记语言)。1978 到 1986 年间,在 ANSI 等组织的努力下,GML 语言进一步发展成为著名的 SGML 语言标准(Standard for General Markup Language,通用标记语言标准)。1989 年,当 Tim Berners-Lee 和他的同事们试图创建一个基于超文本的分布式应用系统时,发现 SGML 是描述超文本信息的最好方案,但是 SGML 过于复杂,不利于信息的传递和解析。于是,Tim Berners-Lee 对 SGML 语言进行了简化和完善。1990 年,用于超文本信息展示的 HTML 正式发布。

HTML 目前成为展现 Web 页面信息的最有效载体之一。

2. 浏览器端动态页面技术

1989 年,能存储、展现二维动画的 GIF 图像格式已发展成熟。GIF 第一次为 HTML 页面引入了动感元素。

1995 年,Java 语言问世,由于 Java 与平台无关的特点,使在浏览器中开发动态应用十分方便。1996 年,Netscape 浏览器在其 2.0 版中增加了对 JavaScript 和 Java Applet 的支持。Microsoft 的 IE 3.0 也在这一年开始支持 Java 技术。

JavaScript 语言是一种使用 Java 语言语法并以脚本方式运行的语言,是第一种应用于 Web 开发的脚本技术。为了用纯 Microsoft 技术与 JavaScript 抗衡,Microsoft 为 IE 3.0 设计了另一种脚本语言——VBScript 语言。

Applet 是运行在浏览器端的 Java 图形界面应用程序。

真正让 HTML 页面又炫又酷、动感无限的是 CSS(Cascading Style Sheets,层叠式样式表)和 DHTML(Dynamic HTML,动态 HTML)技术。

CSS 大大提高开发者对信息展现格式的控制能力。1996 年底,W3C 提出了 CSS 的建议标准,同年,IE 3.0 引入了对 CSS 的支持。1997 年的 Netscape 4.0 不但支持 CSS,而且增加了许多 Netscape 公司自定义的动态 HTML 标记,这些标记在 CSS 的基础上,让 HTML 页面中的各种要素“动”了起来。

1997 年,Microsoft 发布了 IE 4.0,并将动态 HTML 标记、CSS 和动态对象模型(DHTML Object Model)发展成了一套完整、实用、高效的客户端开发技术体系,Microsoft 称其为

DHTML。同样是实现 HTML 页面的动态效果,DHTML 技术无需启动 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。现在,CSS 和 DHTML 技术已经成为 HTML 页面开发的必备技术。

为了在 HTML 页面中实现音视频等更为复杂的多媒体应用,1996 年,Netscape 2.0 成功地引入了对 QuickTime 插件(Plug-in)的支持。插件开发方式也迅速在浏览器端页面开发中流行开来。在 Windows 平台上,出现了 COM 和 ActiveX 技术。1996 年,IE 3.0 正式支持在 HTML 页面中插入 ActiveX 控件的功能。1999 年,Realplayer 插件先后在 Netscape 和 IE 浏览器中取得了成功,与此同时,Microsoft 自己的媒体播放插件 Media Player 也被预装到了各种 Windows 版本之中。

20 世纪 90 年代初期,Jonathan Gay 在 FutureWave 公司开发了一种名为 Future Splash Animator 的二维矢量动画展示工具,1996 年,Macromedia 公司收购了 FutureWave,并将 Jonathan Gay 的发明改名为今天大家所熟悉的 Flash。从此,Flash 动画成为 Web 开发者表现自我、展示个性的最佳方式。

除了编写 HTML 页面之外,客户端应用的开发者还可以利用一些成熟的技术将浏览器的功能添加到自己的应用程序中。从 1992 年开始,W3C 就免费向开发者提供 libwww 开发库。借助 libwww,开发人员可以编写自己的 Web 浏览器和 Web 搜索工具,也可以分析、编辑或显示 HTML 页面。1999 年,Microsoft 在 IE 5.0 中引入的 HTAs(HTML Applications,HTML 应用)技术则允许直接将 HTML 页面转换为一个真正的应用程序。从 1997 年的 IE 4.0 开始,Microsoft 为开发者提供了 WebBrowser 控件和其他相关的 COM 接口,允许程序员在自己的程序中直接嵌入浏览器窗口,或调用各种浏览器的功能,如分析或编辑 HTML 页面等。Windows 98 及其后的 Windows 操作系统甚至还利用 WSH(Windows Script Host)技术将原本只在浏览器中运行的 JavaScript、VBScript 变成了可以在 WIN32 环境下使用的通用脚本语言。

1.2.2 Web 服务端技术

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

1. CGI

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。SSI(Server Side Includes,服务器端包含)技术可以让 Web 服务器在返回 HTML 文件前,更新 HTML 文件的某些内容,但其功能非常有限。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,美国国家超级计算应用中心)提出,1995 年,NCSA 开始制定 CGI 1.1 标准,于 1997 年发布 CGI 1.2 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,这样就可以实现客户端和服务端动态交换信息。使用 CGI 技术生成页面内容的过程可以用图 1-1 描述。

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,人们终于可以享受受到信息检索、信息交换、信息处理等更为便捷的信息服务。

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 甚至