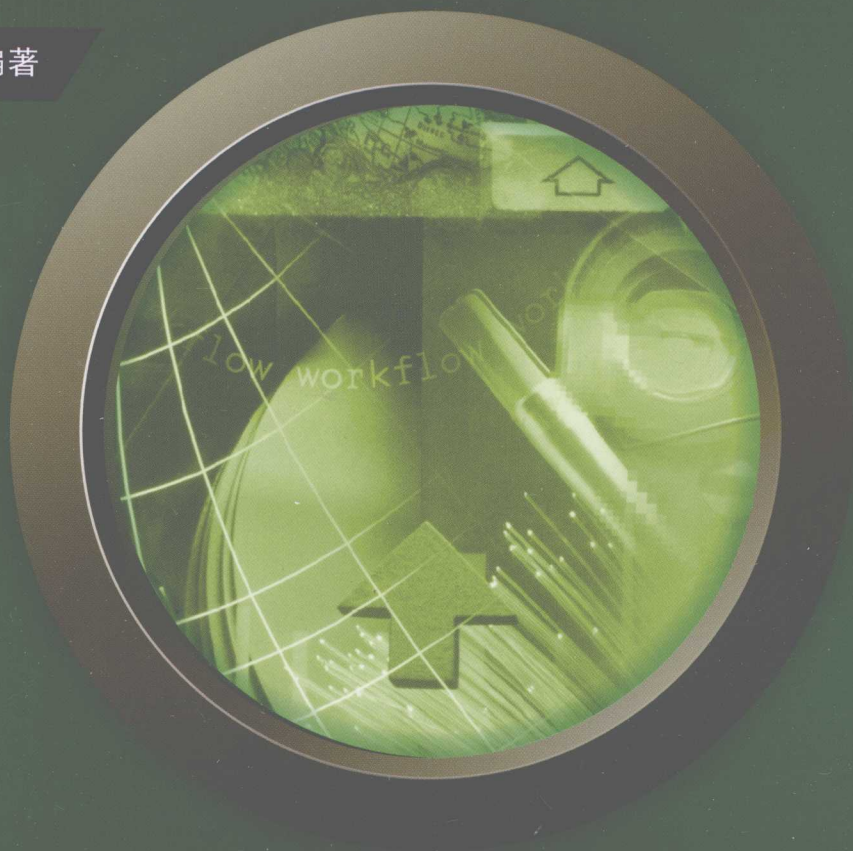




高等院校计算机应用技术系列教材

ASP 网络编程基础

蒲 鹏 陈 慧 编著



清华大学出版社 • 北京交通大学出版社

高等院校计算机应用技术系列教材

ASP 网络编程基础

蒲 鹏 陈 慧 编著

清华大学出版社

北京交通大学出版社

• 北京 •

内 容 简 介

本书主要内容包括网络编程及 ASP 的简介, 服务器端 IIS 的安装、配置及常见故障排除方法, HTML 语言的标记, 服务器端脚本语言 VBScript 的语法, ASP 内置对象 Request、Response、Session、Application、Server, 在 ASP 网页中对 Access 数据库的存取操作, ASP 典型案例(考生管理系统), 等等。除此之外, 为了拓展读者关于网络编程的知识面, 本书在最后两章介绍了客户端脚本语言 JavaScript 和 ASP 常用的内置组件(如广告滚动组件、文件访问组件)。

本书按“引导、实践、创新”三步走的指导思想, 设置了实例、上机练习题和网站设计综合练习题, 力求让每一个读者迅速进入实践过程, 突出强调技能训练, 最终达到让读者学会独立“做网站”的目的。

本书内容翔实, 实例丰富, 既适合作为高校针对特定的研究生选修课(文科)及公选课(面向全校学生)的授课教材, 又适合作为大专院校的计算机专业课程授课教材。此外, 对有志于投身网络编程的广大读者, 起到了很好的入门指路的作用。

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 13501256678 13801310933

图书在版编目(CIP)数据

ASP 网络编程基础 / 蒲鹏, 陈慧编著. —北京: 清华大学出版社; 北京交通大学出版社, 2009.12

(高等院校计算机应用技术系列教材)

ISBN 978-7-81123-904-1

I. ①A… II. ①蒲… ②陈… III. ①主页制作-程序设计 IV. ①TP393.092

中国版本图书馆 CIP 数据核字(2009)第 202976 号

责任编辑: 谭文芳 特邀编辑: 宋林静

出版发行: 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: 北京瑞达方舟印务有限公司

经 销: 全国新华书店

开 本: 185×260 印张: 10 字数: 253 千字

版 次: 2010 年 1 月第 1 版 2010 年 1 月第 1 次印刷

书 号: ISBN 978-7-81123-904-1/TP·542

印 数: 1~4 000 册 定价: 17.00 元

本书如有质量问题, 请向北京交通大学出版社质监组反映。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@bjtu.edu.cn。

前 言

当今的 Web 编程已经成为软件开发的主流,原来 C/S 架构的程序也慢慢地开始往 B/S 架构转移,小到公司企业网站,大到办公系统 OA,客户信息管理系统 CRM,企业资源规划 ERP 系统,都无不是 Web 应用大显身手的地方,于是网络编程迅速成为本专科院校的专业或非专业学生感兴趣的技术。

本教材所有方案都是编者在实际项目开发过程中总结出来的经验,是多年来的心血。本教材核心思想就是让广大读者以 ASP 作为编程基础,逐步掌握网络编程的思想,本书在编写过程中强化核心理论,切忌泛泛而谈,精选新颖、全面的内容,注重通过应用来体现理论价值,培养读者的实践动手能力,使读者轻松、快捷地掌握本书的知识。本书共分 8 章,主要内容包括:第 1 章为动态网页的发展背景。本章主要是为全书整个 ASP 内容作一个铺垫,让读者了解 ASP 是什么,在当前的动态网页制作中处于一个什么样的位置。第 2 章为 ASP 的开发环境。本章主要详细地介绍了 ASP 的编写环境、开发环境、运行环境的配置,这是本书的实践基础。第 3 章为 HTML 基础。本章主要介绍静态网页的主要内容,ASP 制作的动态网页是在静态网页的基础上发展而来的。且 ASP 的精髓就在于它可以控制 HTML 的输出。第 4 章为 ASP 的内置对象。本章主要介绍了 ASP 作为一种技术主要包含的内容。第 5 章为服务器端脚本 VBScript。本章主要介绍了用来实现 ASP 技术的服务器端脚本 VBScript 的语法,这是本书的重点,如果不懂 VBScript 语法,就不会使用 ASP 技术。第 6 章为 ASP 进行数据库的操作。本章主要介绍了 ASP 操作数据库的过程,其中包括了检索、更新、插入、删除等基本操作,这是本书的核心章节,也是制作动态网站的核心内容。第 7 章为 JavaScript 的应用。本章主要介绍了另外一种在客户端应用广泛的脚本语言 JavaScript,它可以辅助 ASP 技术,使得开发的网站有更好的动态交互以及良好的用户体验。第 8 章为 ASP 的内置组件。本章介绍了 ASP 的常用内置组件,主要目的是为了拓展读者的知识面及保证 ASP 内容的完整。随着 Web 技术的发展,很多内置组件都被其他技术所替代。

本书的第 1、2、5、6 章由蒲鹏编写,第 3、4、7、8 章由陈慧编写。本书的作者都在网络编程领域从事 ASP 教学以及 Web 开发,有着丰富的工作经验和教学经验,在该领域曾编写过多本教材,发表过多篇文章。

由于计算机技术的发展日新月异,并且本书作者水平有限书中难免有不妥之处,敬请同行与读者指正。

编 者

2009 年 10 月

目 录

第 1 章 动态网页的发展背景	1
1.1 服务器端动态技术概述	1
1.1.1 服务器端动态技术的发展过程	2
1.1.2 服务器端动态技术的工作原理	3
1.1.3 主要的服务器端动态技术	4
1.2 客户端技术的概述	5
1.3 动态技术的最新发展	6
1.3.1 Web 编程的三层架构	6
1.3.2 客户端技术 Ajax 的迅猛发展	7
1.4 关于 ASP 技术	8
1.4.1 什么是 ASP	8
1.4.2 ASP 所包含的内容	9
1.4.3 ASP 的执行原理	10
1.5 小结	10
1.6 习题	11
第 2 章 ASP 的开发环境	12
2.1 ASP 编写环境的配置	12
2.1.1 基本编写工具的概述	12
2.1.2 编写 helloworld.asp	13
2.2 ASP 运行环境的安装	14
2.2.1 Windows 2003 Server 操作系统下 IIS 的安装配置	14
2.2.2 Windows XP 下环境 IIS 的安装	21
2.2.3 建立主目录和虚拟目录	22
2.3 ASP 调试环境	25
2.4 小结	28
2.5 习题	29
第 3 章 HTML 基础	30
3.1 HTML 概述	30
3.1.1 什么是 HTML 语言	30
3.1.2 HTML 文件的编写规则	31
3.1.3 HTML 文件的常用编写方法	32
3.2 HTML 语言的常用标识符	33
3.2.1 文档头标记	34

3.2.2	文字标记	35
3.2.3	图像标记	35
3.2.4	超链接标记	37
3.2.5	表格标记	40
3.3	HTML 的表单	44
3.4	HTML 的框架	50
3.5	小结	54
3.6	习题	54
第 4 章	ASP 的内置对象	56
4.1	ASP 内置对象的概述	56
4.2	Response 对象	57
4.2.1	Response 对象的方法	58
4.2.2	Response 对象的属性	63
4.3	Request 对象	64
4.4	Cookies 数据集合	66
4.5	小结	66
4.6	习题	67
第 5 章	服务器端脚本 VBScript	69
5.1	常量和变量、数组、逻辑运算符号	70
5.1.1	常量和变量	70
5.1.2	数组	72
5.1.3	操作符	73
5.2	条件结构、循环结构	76
5.2.1	条件结构	77
5.2.2	循环结构	79
5.3	常用函数	81
5.4	实践篇	85
5.5	小结	88
5.6	习题	88
第 6 章	ASP 操作数据库	90
6.1	数据库概述	90
6.1.1	Access 数据库简介	90
6.1.2	ASP 对数据库的访问	91
6.1.3	记录集对象	93
6.2	ASP 操作数据库的添加操作	95
6.2.1	使用 SQL 语句操作命令添加	95
6.2.2	使用记录集对象的 Addnew()方法添加	96
6.2.3	网站注册功能的实现	98
6.3	ASP 操作数据库的检索操作	101

6.3.1	检索的 SQL 语句语法	101
6.3.2	使用记录集对象进行检索	103
6.3.3	对网站注册的完善	105
6.3.4	网站登录功能的实现	106
6.4	ASP 操作数据库的更新操作	108
6.4.1	使用 SQL 语句操作命令更新	108
6.4.2	使用记录集对象更新	109
6.5	ASP 操作数据库的删除操作	110
6.6	ASP 操作数据库的综合实例——考生管理系统	111
6.6.1	考生管理系统的设计	111
6.6.2	考生管理系统的实现	113
6.7	小结	121
6.8	习题	122
第 7 章	JavaScript 的应用	124
7.1	JavaScript 概述	124
7.1.1	什么是 JavaScript?	124
7.1.2	JavaScript 在网页中的位置	125
7.2	JavaScript 的基本语法	125
7.2.1	JavaScript 的常量	125
7.2.2	JavaScript 的变量	126
7.2.3	JavaScript 常用的语法结构	127
7.2.4	JavaScript 的对象	129
7.2.5	JavaScript 的事件	136
7.3	JavaScript 对表单的操作	137
7.3.1	正则表达式的概述	137
7.3.2	正则表达式在表单中的使用	140
7.4	小结	142
7.5	习题	142
第 8 章	ASP 的内置组件	143
8.1	获取浏览器信息组件	143
8.2	广告轮显组件	144
8.3	Web 导航链接组件	146
8.4	文件访问组件	146
8.5	小结	149
参考文献	参考文献	150

第 1 章 动态网页的发展背景

本章学习指引:

- 了解服务器端动态技术的发展过程;
- 掌握动态网页的工作原理;
- 了解当今动态技术的发展形势;
- 了解 ASP 技术的发展概况。

在计算机网络以及相关网络技术日趋普及成熟以前, 社会生产、生活应用领域中的桌面应用程序研发一直是 IT 应用技术的重要组成部分。随后在桌面应用程序研发基础上发展起来的基于客户-服务器 (Client/Server, C/S) 工作模式的网络编程, 则真正掀开了网络编程的序幕。这段时间诞生了很多经典编程语言, 例如, Visual Basic、Delphi 和 PowerBuild, 等等。后来, 随着网络编程的工作模式发生改变, 诞生了网页这个概念。最初的网页只是由一系列 HTML (HyperText Mark-up Language, 超文本标记语言) 标识来描述, 通常把这种网页称为静态网页。与原来的 C/S 工作模式的网络程序相比, 静态网页的最大缺点是缺乏交互性, 而且信息也不能适时更新。为了解决这些问题, 各种服务器端动态技术应运而生。在它们的支持下, 动态网页诞生了, 这也使得网络编程发展到了今天以浏览器-服务器 (Browser/Server, B/S) 工作模式为基础的 Web 阶段。当然, 在这个阶段, 最主要的两种 IT 技术就是服务器端的动态技术和基于浏览器的各种客户端脚本技术。而本书所讨论的 ASP (Active Server Page, 动态服务器页面) 技术则是最经典的服务器端动态技术。因此, 也把基于服务器端 ASP 技术的动态网页称为 ASP 网页。

1.1 服务器端动态技术概述

服务器端动态技术指的是运行在 Web 服务器上的、负责网页的动态显示及逻辑的动态处理的技术。图 1-1 生动地说明了在 Web 应用领域, Web 服务器端动态技术的主要作用。

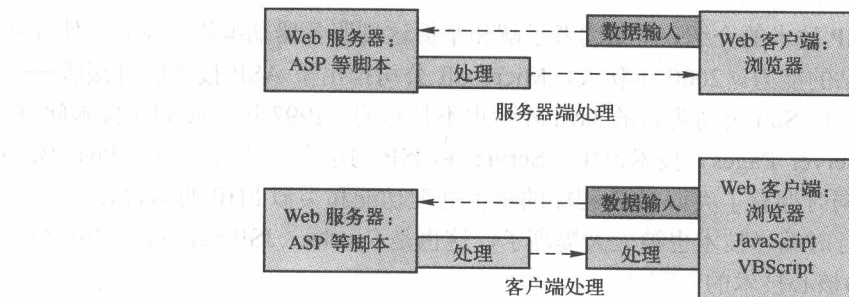


图 1-1 Web 服务器端动态技术的主要作用

首先,客户端(即 Web 浏览器)根据某资源的 URL (Uniform Resource Locator, 统一资源定位器)向 Web 服务器提出请求;然后,Web 服务器将此请求交给服务器端动态技术引擎来处理,如果要存取数据库服务器上的数据,则向数据库服务器发出处理请求,由数据库服务器将执行结果返回给服务器端动态技术引擎;最后,服务器端动态技术引擎把处理结果以标准 HTML 输出给客户端,通过浏览器解析执行,使客户最终看到这个结果。这就是一个简单的动态网页从被客户请求到显示给客户的过程。在这个过程中,读者也可以简单、直观地认识服务器端动态技术的作用。下面,就从 Web 服务器端动态技术的发展过程开始,更深入地认识和掌握这门技术。

1.1.1 服务器端动态技术的发展过程

服务器端动态技术发展到今天经历了一系列的过程,其中,最早投入使用并迅速成为主流的是通用网关接口(Common Gateway Interface, CGI)技术。CGI 技术的出现使得服务器端动态技术拥有了强大的交互能力,并且扩展了 Web 服务器的功能。例如,Web 服务器和数据库管理系统交互的能力、Web 服务器运行服务器进程的能力,等等。这些扩展加速了 Web 技术在商业上的应用,并且促进了各种大型网站的诞生。CGI 技术在其诞生之初以其强大的功能成为 90%大型网站首选的构建技术。但是大量使用 CGI 技术后不久,人们发现 CGI 技术存在两个致命的问题:首先,由于 CGI 技术提供了 Web 服务器和主机程序的交互能力,因此,这也使 CGI 技术存在比较严重的安全问题,恶意黑客可以通过 CGI 技术的漏洞控制服务器并造成服务瘫痪和数据损失;其次,CGI 技术需要为每一次访问建立一个进程来完成对 CGI 脚本的处理,如果网站的访问量比较大,那么这种处理方式将严重影响网站的性能。由于这些问题的存在,各大软件厂商开始探寻用脚本语言实现 CGI 应用的可行方式。从这一点也可以看出 CGI 技术最大的意义在于它提出的全新的网络工作模式。

1994 年,Rasmus Lerdorf 发明了专用于 Web 服务器端编程的 PHP(Hypertext Preprocessor, 超级文本预处理语言)。与以往的 CGI 程序不同,PHP 语言将 HTML 语言和 PHP 指令合成为完整的服务器端动态页面,Web 应用的开发者可以用一种更加简便、快捷的方式实现动态 Web 功能。1996 年,Microsoft 借鉴 PHP 的思想,在其 Web 服务器 IIS(Internet Information Server) 3.0 中引入了 ASP 技术,该技术也迅速成为了 Windows 操作系统下 Web 服务器端的主流开发技术。

本书将通过 ASP 技术的介绍,帮助读者了解和掌握这种服务器端编程技术,并使读者初步拥有建立动态网站的能力。2000 年秋天,Microsoft 公司推出了 ASP 技术的升级版——ASP.NET 技术。当然,以 Sun 公司为首的 Java 阵营也不甘示弱。1997 年,Servlet 技术问世,1998 年,JSP(Java Server Pages)技术诞生。Servlet 和 JSP 的组合(还可以加上 Java Bean 技术)让 Java 开发者同时拥有了类似 CGI 程序的集中处理功能和类似 PHP 的 HTML 嵌入功能,此外,Java 的运行时编译技术也被大大提高了,这也是 Servlet、JSP Servlet 和 JSP 被后来的 J2EE 平台吸纳为核心技术的原因之一。

时至今日,服务器端动态技术仍然在不断地向前发展,新的技术层出不穷。

1.1.2 服务器端动态技术的工作原理

在介绍服务器端动态技术的时候，提到了 B/S 结构。它是随着 Internet 技术的兴起，网络编程步入 Web 阶段的主要工作模式，同时，它也是对 C/S 结构的一种改变或者改进的结构。本小节就从这两种结构入手，简单介绍服务器端动态技术的工作原理。

1. C/S 结构

C/S 结构中，“C”更强调的是在客户机上单独安装一个客户端软件，所以，桌面应用程序的很多优点都可以在客户端上安装的软件上得以体现，具体来说它具有下列优点。

- ☞ 由于客户端实现与服务器的直接相连，没有中间环节，因此响应速度快；
- ☞ 操作界面漂亮、形式多样，可以充分满足客户自身的个性化要求；
- ☞ C/S 结构的管理信息系统具有较强的事务处理能力，能实现复杂的业务流程。

当然，它也有很多缺点。根据前面的介绍可以知道，在客户机上需要专门的客户端安装程序，在开发的过程中当然也要兼顾客户端的开发，这或多或少会影响整体的开发进度，并且增加开发成本。此外，C/S 结构的兼容性也比较差，对于不同的开发工具，具有较大的局限性，例如，采用不同的工具，就需要重新改写程序。

2. B/S 结构

随着 Internet 和 WWW 的流行，以往的 C/S 结构的系统很难满足当前的全球网络开放、互连、信息随处可见和信息共享的新要求，于是就出现了 B/S 结构的网络应用。在这种结构下，用户工作界面上只有极少部分事务逻辑通过客户端的浏览器实现，而“B”实际上代表的就是客户端的浏览器，例如，一般计算机上的 IE (Internet Explorer)、FireFox、Opera，等等。在 B/S 结构中，主要事务逻辑在服务器端实现，“S”指的就是服务器 (Server)，这个服务器可以包含 Web 服务、Ftp 服务、数据库服务，等等，形成所谓的三层结构。这样就大大简化了客户端计算机的载荷，减轻了系统维护与升级的成本和工作量，降低了开发的总体成本。如图 1-2 所示为典型的 B/S 结构的工作流程示意图。

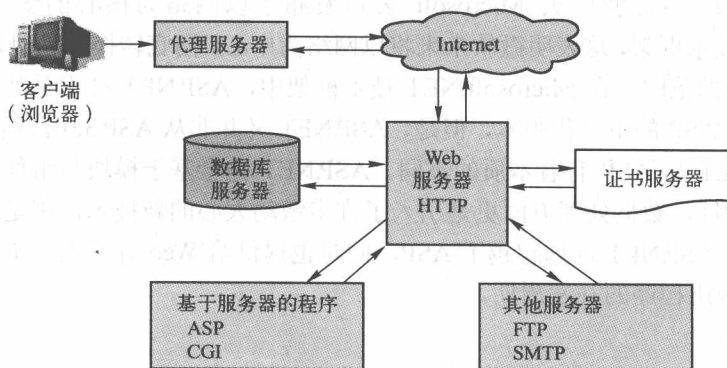


图 1-2 B/S 结构的工作流程示意图

用户可以通过客户端的浏览器去访问 Internet 上的文本、数据、图像、动画、视频点播

和声音信息, 这些信息都是由许许多多的 Web 服务器产生的, 而每一个 Web 服务器又可以通过各种方式与数据库服务器连接, 大量的数据实际存放在数据库服务器中。客户端除了浏览器, 一般不需要任何用户程序, 只需从 Web 服务器上下载程序到本地来执行, 在下载过程中若遇到与数据库有关的指令, 由 Web 服务器交给数据库服务器来解释执行, 并返回给 Web 服务器, Web 服务器又返回给用户。在这种结构中, 无数的网连接形成一个巨大的网, 即全球网。各个企业可以在此结构的基础上建立自己的 Intranet。下面介绍 B/S 模式的优点和缺点。

优点:

- ✧ 具有分布性特点, 可以随时随地进行查询、浏览等业务处理;
- ✧ 业务扩展简单方便, 通过增加网页即可增加服务器功能;
- ✧ 维护简单方便, 只需要改变网页, 即可实现所有用户的同步更新;
- ✧ 开发简单, 共享性强。

缺点:

- ✧ 个性化特点明显降低, 无法实现具有个性化的功能要求;
- ✧ 操作是以鼠标为最基本的操作方式, 无法满足快速操作的要求;
- ✧ 页面动态刷新、响应速度明显降低;
- ✧ 功能弱化, 难以实现传统模式下的特殊功能要求。

1.1.3 主要的服务器端动态技术

正像 1.1.1 节所讲述的那样, 服务器端动态技术发展到今天, 已经形成了百花齐放的局面, 但就广泛性和技术性而言, 基本上是三分天下: 微软公司的 ASP 和 ASP.NET, Zend 公司的 PHP 和 SUN 公司倡导的 JSP (就在本书正在编写的过程中, SUN 公司已经正式被甲骨文公司兼并了, 这将是 IT 领域又一次重大的改变, 这次兼并给 IT 领域带来的影响, 在不久的将来将会初现端倪)。下面分别加以介绍。

1. Microsoft ASP.NET

2000 年秋天, IT 业巨头 Microsoft 公司发布了以网络为核心的新的技术框架——Microsoft.NET 技术框架, 这表明微软未来将以网络为中心, 彻底转换产品研发、发布的方式, 改变产品和服务的范围。在 Microsoft.NET 技术框架中, ASP.NET 技术是其中最主要的组成部分之一, 它是 ASP 的下一代版本, 但是, ASP.NET 又并非从 ASP 3.0 自然演化而来, 在许多方面, ASP.NET 与 ASP 有着本质的不同。ASP.NET 完全基于模块与组件, 具有更好的可扩展性与可定制性, 数据处理方面更是引入了许多激动人心的新技术, 正是这些具有革新意义的新特性, 使 ASP.NET 远远超越了 ASP, 同时也提供给 Web 开发人员更好的灵活性, 有效缩短了 Web 应用程序的开发周期。

2. PHP

PHP 是一种服务器端的脚本语言, 嵌入在 HTML 中, 它可以在多种平台上运行。PHP 的语法与 C 语言、Java 语言的语法极为相似, 但也有自己独特的语法。它具有庞大的函数库, 这主要是因为它是开源式的, 源代码完全公开, 每个使用 PHP 的程序员都可以加入自己的函

数据库,从而实现更多的功能,PHP支持几乎所有的数据库。但是,PHP对组件的支持不够,扩展性较差,常与免费的数据库系统MySQL一起来构建中小型B/S应用的网站或信息系统。PHP的第四代核心引擎大幅改动,让程序的执行速度更快。最佳化之后的效率已经比传统CGI或者ASP等程序有更好的表现。

3. JSP

JSP使用的编程语言是Java。它是由SUN公司提出、多家公司合作建立的一种动态网页技术。该技术最初的目的只是为了整合已经存在的Java编程环境(例如,Java ServerLet等),结果产生了一种全新的足以和ASP抗衡的网络程序语言。JSP的最大优点是开放的、跨平台的结构。它可以运行在几乎所有的服务器系统上,包括Windows NT、Windows 2000 Server、Windows 2003 Server、UNIX、Linux等操作系统,当然,这些服务器需要安装JSP服务器引擎软件。

1.2 客户端技术的概述

Web是一种典型的分布式应用架构。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。因此,Web开发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。鉴于这一点,下面将简要介绍客户端技术的发展过程。

Web客户端的主要任务是以网页的形式展现信息内容,而HTML语言则是信息展现的最有效载体之一。作为一种实用的超文本语言,到1990年,第一个图形化的Web浏览器“World Wide Web”终于可以使用一种为Web量身定制的语言——HTML语言来展现超文本信息。

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,满足不了人们对信息丰富性和多样性的强烈需求,这使得由静态技术向动态技术的转变成为Web客户端技术演进的必然趋势。

Web出现后,GIF(Graphics Interchange Format,图像互换格式)第一次为HTML页面引入了动感元素。但更大的变革来源于1995年Java语言的问世。Java语言天生所具备的与平台无关的特点,使它很快成为在浏览器中开发动态应用的捷径。1996年,著名的Netscape浏览器在其2.0版中增加了对Java Applets和JavaScript的支持。作为Netscape的竞争对手,Microsoft的浏览器IE 3.0也在这一年开始支持Java技术。这就使得喜欢动画、喜欢交互操作、喜欢客户端应用的开发人员,可以用Java或JavaScript语言随心所欲地丰富HTML页面的功能,而不用再考虑浏览器是否支持这些功能。

真正让HTML页面又酷又炫、动感无限的是层叠样式表(Cascading Style Sheets,CSS)和DHTML(Dynamic HTML)技术。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。CSS大大提高了开发者对信息展现格式的控制能力。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“活动”起来。1997年,Microsoft发布了IE 4.0,并将动态HTML标记、CSS和动态对象模型发展成为一套完整、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。

1.3 动态技术的最新发展

近年来, 动态技术的发展没有局限于 1.1 节所提到的对各种开发语言的完善和发展, 而是开发出许多以服务于开发进度、抽象业务逻辑、增加用户体验为目标的发展模式和应用框架, 读者可以通过本书下面介绍内容对它们有个基本的了解。使这些内容能给有志于在动态技术编程领域更上一层楼的读者提供指导性的意见。

1.3.1 Web 编程的三层架构

随着 Web 编程的发展, 在这个领域也慢慢开始有了一定的开发模式或开发架构。这些模式或架构基本上都建立在把整个 Web 的开发过程分为业务逻辑模块、用户视图模块和流程控制模块的基础上, 这样做的目的可以使整个开发更加标准, 逻辑更加清楚。在这个基础上形成的就是 Web 编程中一个非常出色的三层架构——MVC (Model View Controller) 模型。

所谓 MVC 模型, 是指业务逻辑模块、用户视图模块、流程控制模块三元组, 它们分别对应上面提到的把 Web 开发过程划分的三个部分, 下面分别介绍这三部分。

1. 业务逻辑模块

业务逻辑模块 (Model) 表示应用领域有关的类别、物体、数据结构及企业业务逻辑等。通常, 业务逻辑模块是企业业务流程的软件近似模型, 所以, 在定义业务逻辑模块时只需对企业业务流程中的对象进行合理抽象、封装对象的属性和对象隐含的业务逻辑。

2. 用户视图模块

一方面, 用户视图模块 (View) 为用户提供了输入手段, 并触发流程控制模块运行; 另一方面, 它通过业务逻辑模块访问企业数据, 并用某些方式显示这些数据。同时, 当业务逻辑模块变化时, 它做出相应变化。用两种方法之一: Push (推) 模式, 让用户视图模块在业务逻辑模块处注册, 以便在业务逻辑模块变化时获得提醒; Pull (拉) 模式, 让用户视图模块在需要获得最新数据时调用业务逻辑模块。

3. 流程控制模块

流程控制模块 (Controller) 将用户与用户视图模块的交互转换为基于应用程序行为的标准业务事件, 再将标准业务事件解析为业务逻辑模块应执行的动作。业务逻辑模块执行的动作包括激活业务逻辑和改变业务逻辑模块的状态。根据用户交互信息和业务逻辑模块执行动作的结果, 流程控制模块选择一个合适的用户视图模块展现给客户。

传统的 MVC 模型只是从概念上将视图从流程控制、业务逻辑中独立出来, 并定义了相互间作用的机制, 使各个模块的开发相对独立, 但是, 它没有针对不同视图类型, 解决流程控制等对象的统一问题。

1.3.2 客户端技术 Ajax 的迅猛发展

当今动态技术发展的另一领域就是以用户体验型的网络服务为特征的 Web 应用, 这个应用正在影响着目前互联网的发展趋势。而主导这次应用的就是日趋流行的 Ajax (Asynchronous JavaScript and XML) 技术, 它可以使客户端与服务器之间传输的数据量大大减少, 可感知的系统响应时间大幅度缩短。由于反馈及时, 用户感觉对应用程序的控制更加得心应手。下面将简单介绍 Ajax 技术。

Ajax 实际上是几项技术按一定的方式组合, 在共同的协作中发挥各自的作用。在 Ajax 之前, Web 站点强制用户进入提交、等待、重新显示范例, 用户的动作总是与服务器的“思考时间”同步。而 Ajax 诞生之后, 它提供了与服务器异步通信的能力, 这个“异步”也体现了 Ajax 的核心思想, 这也使得用户从一般的请求、响应的循环中解脱出来。例如, 借助于 Ajax, 可以在用户单击按钮时, 使用 JavaScript 和 DHTML 立即更新用户界面, 并向服务器发出异步请求, 以执行更新或查询数据库。当请求返回时, 就可以使用 JavaScript 和 CSS 技术来更新用户界面, 而不是刷新整个页面。最重要的是, 用户甚至不知道浏览器正在与服务器通信, Web 站点看起来是即时响应的。

1. Ajax 中包含的内容以及相关技术

(1) XML 技术

通过 XML (Extensible Markup Language, 可扩展标记语言), 可以规范地定义结构化数据, 使网上传输的数据和文档符合统一的标准。用 XML 表述的数据和文档, 可以很容易地让所有程序共享。同时, XML 还可以与 XSLT (eXtensible Stylesheet Language Transformation, 可扩展样式表转换) 进行数据交换与处理, 改变数据的显示外观。

(2) DOM 技术

DOM (Document Object Model, 文档对象模型) 是提供给 HTML 和 XML 使用的一组 API, 它提供了文件的表述结构, 并可以利用它改变其中的内容和可见物。脚本语言通过 DOM 才可以跟页面进行交互。Web 开发人员操作及建立文件的属性、方法以及事件都以对象来展现, 实现动态显示和交互; 它以树形结构表示 HTML 和 XML 文档, 定义了遍历这个树和检查、修改树的节点的方法和属性。

(3) XMLHttpRequest 对象

XMLHttpRequest 对象是 XMLHttpRequest 组件的对象, 又是 Ajax 的核心组成内容。通过这个对象, Ajax 可以像桌面应用程序一样只同服务器进行数据层面的交换, 而不用每次都刷新界面, 也不用每次将数据处理的工作都交给服务器来做; 这样既减轻了服务器负担, 又加快了响应速度、缩短了用户等待的时间。

(4) JavaScript 语言

JavaScript 是使用最多的客户端语言, 本书将在第 7 章中进行介绍, 它主要被用于操作 XMLHttpRequest 对象和数据库进行通信, 最后用它绑定和处理所有数据。

2. 几种常用的 Ajax 框架

(1) Google Web Toolkit

这是一个开放版本的 Ajax 框架，用户通过它可以构建像 Google Maps 和 Gmail 之类的 Web 应用程序。GWT (Google Web Toolkit) 已经加入到 Google 的代码库当中，这个代码库包含了很多 API 和开发工具。Google 不仅以它出色的搜索引擎闻名天下，还是各种热门技术的倡导者，更多有关 Google 的 Ajax 信息，可以访问 goog-ajaxslt.sourceforge.net。如图 1-3 所示是经常使用的 Google Suggest 功能，它是使用 GWT 技术开发的。

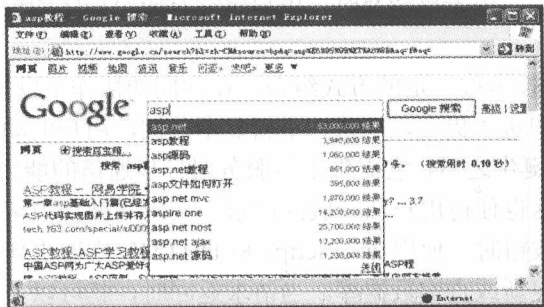


图 1-3 Google Suggest 功能

(2) xajax 框架

这是一个工作在 PHP 环境下的 Ajax 框架，它通过 JavaScript 直接调用 PHP 函数、对象方法和类方法。服务器端先创建 xajax 响应对象，该对象提供非常多的命令，例如，给一个 HTML 元素赋值，显示一个警告对话框，等等。它还支持自定义脚本输出。所有特定浏览器代码（如使用 XMLHttpRequest）都是抽象的，并且发送各种数据类型到服务器端都非常容易实现。xajax 提供多种配置选项以便易于与现存 Web 应用或框架相集成。它用 JavaScript 编写的核心程序库简洁明了，而且可以通过覆盖来支持更多的高级 JavaScript 功能。

(3) Prototype 框架

这是最受欢迎的 Ajax 框架之一。使用之后会明显感觉到它几乎就是 JavaScript 语言的自然扩展，甚至坚信 Prototype 框架应该被纳入 JavaScript 语言。对于许多 Web 开发案例，紧凑、强大并且设计合理的 Prototype 框架非常实用。

类似的框架还有很多，其中也不乏国内开发人员的成熟之作。总之，Ajax 交互模型的出现提供了一种新的 Web 设计方法，通过对 Ajax 模式的应用，不仅改变了传统 Web 应用模式，同时也开拓了一片 Web 开发的新领域。使用 Ajax 可以构建动态性更强和响应更灵敏的 Web 应用程序，它直接通过浏览器与后端服务进行通信，既能提高应用程序数据通信效率和响应灵敏度，又能提升用户的浏览体验。

1.4 关于 ASP 技术

上面简单介绍了动态网页发展的概况，本节将详细介绍本书的核心——动态网页技术 ASP。

1.4.1 什么是 ASP

ASP (Active Server Pages, 动态服务器页面) 是基于微软 Windows 平台的服务器端动态

网页开发技术，可以用 VBScript 或 Jscript（微软开发的一种类似于 JavaScript 的服务器脚本语言）语言来编写，支持 COM/DCOM 构件模型，使用它可以创建和运行动态、交互的 Web 服务器应用程序。使用 ASP 可以组合 HTML 语言、脚本命令和 ActiveX 组件以创建交互的 Web 页和基于 Web 的功能强大的应用程序。ASP 只是一种技术，不是一种开发语言。

ASP 的第一个版本是 0.9 测试版，它给 Web 开发带来一次不小的革命，它能够将代码直接嵌入到 HTML 网页中，使得设计 Web 页面变得更加简单，并且通过内置的组件能够使 Web 页实现强大功能。接着又研发出 Active Server Page 1.0，并且不久就在 Windows 平台上广泛使用。

1998 年，微软公司又发布了 ASP 2.0。ASP 1.0 与 ASP 2.0 的主要区别是外部的组件需要实例化，而且每个组件有了自己单独的内存空间，内置的 MTS（Microsoft Transaction Server，微软事务处理服务器）也使得制作组件变得简单。有了 ASP 2.0 和 IIS 4.0，就可以建立 ASP 应用。

微软公司接着发布了 Windows 2000 操作系统。这个 Windows 版本支持 IIS 5.0 及 ASP 3.0。此次并不是简单对 ASP 进行了补充，而是针对核心的工作方式进行了改革。微软结合了 MTS 与 COM 核心环境做出了 COM+，使 Web 服务器有了一种新的方法来使用组件，同样给 Web 服务器带来了更多的稳定性，使得 Web 服务器成为一个可以升级的高效率的工作平台。

1.4.2 ASP 所包含的内容

ASP 提供了最常用的五大内置对象、一个集合和一个文件。五大对象分别是：Response、Request、Session、Application 和 Server，每个对象都提供了一系列的集合、属性和方法，这些内容将在后续章节中加以详细介绍。一个集合是 Cookies，一个文件是 Global.asa。除此之外，ASP 还包含内置组件。ASP 最常用的内置组件是操作文件的组件和操作广告条的组件。ASP 最强大的功能是使用外置组件，比如使用外置组件实现文件上传，发送 E-mail，等等。

ASPError 对象是 ASP 3.0 中的一个新的对象，通过 Server 对象使用 GetLastError() 方法。它提供了发生在 ASP 中的上一次错误的详细信息。如图 1-4 所示表现了 ASP 各个内置对象在整个 ASP 的 Web 工作流程中的作用。

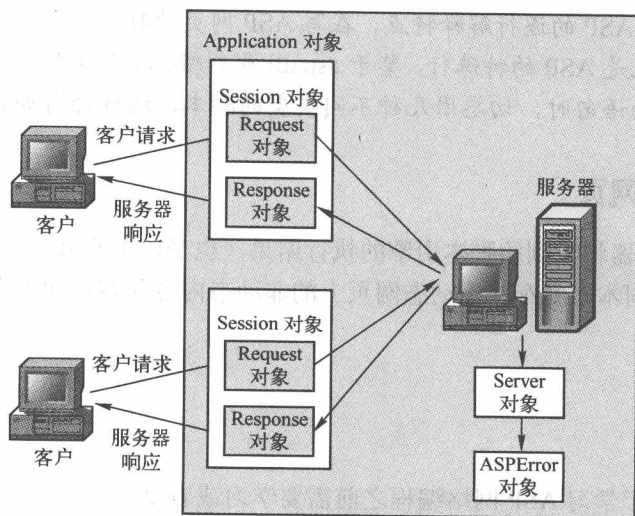


图 1-4 ASP 内置对象在 ASP 工作流程中的作用

1.4.3 ASP 的执行原理

ASP 本身包含了一个动态链接库文件, 文件名是 `asp.dll`, 默认安装在 `Winnt\System32\inet_srv` 目录下。当浏览器从 Web 服务器上请求动态网页 (`.asp` 文件) 时, 这个动态链接库文件负责找到动态网页, 然后对它进行语法分析, 通过标识判别服务器端脚本的类别, 可能是 `VbScript`, 也可能是 `Jscript`。本书重点介绍用 `VbScript` 编写 ASP 网页。判断完成后, 将这个脚本传送给相应的脚本引擎, 脚本的执行结果与 ASP 页中的 HTML 和模板文本结合在一起, 送往原先提出请求的客户端。对这个过程, 可以分为以下 3 个阶段。

1. 分析动态网页代码

`asp.dll` 首先判断动态网页中是否有 ASP 服务器端的代码需要执行。假如没有, 它简单地通知 Web 服务器这种情况, 并让 Web 服务器向客户端直接发送该页面。

 提示: 后缀是 `asp` 的网页中, 可能包含、也可能不包含 `asp` 语句。如果一个网页中包含有 `asp` 语句, 它的后缀名必须是 `asp`, 而当网页中不包含 `asp` 语句的时候, 网页的后缀可以是, 也可以不是 `asp`。作为初学者一定要谨记。

2. 执行动态网页代码

当 `asp.dll` 接收到包含有服务器端脚本代码的页面时, 它会交给相应的脚本引擎逐行进行解释, 脚本引擎处理后的结果被发送回 Web 服务器时, 都会插入到页面上相应的位置。那些非服务器端的脚本, 或不需要 ASP 进行服务器处理的, 将直接返回给 Web 服务器。为提高操作的效率, ASP 也常把脚本引擎创建的编译代码, 放在高速缓存中以备再次调用。这个代码与发往客户端的输出结果是不相同的。客户看到的是脚本代码在经过解释、语法检查和编译后的执行结果。在服务器上高速缓存的只是编译后的代码, 在原来的源文件变化后, 这些代码会被放弃。

 提示: 鉴于 ASP 的逐行解释特点, 在写 ASP 网页的时候, 要尽可能将注释写得简明扼要, 因为注释行也是 ASP 的特殊行, 鉴于 `asp.dll` 要判断脚本的类型, 所以初学者也一定要记住, 在编写 `asp` 的语句时, 切忌用几种不同类型的脚本, 这样会增加 `asp.dll` 的负担。

3. 返回静态网页

最后 Web 服务器将收到的脚本引擎的执行结果 (也是以 HTML 语言形式存在的, 暂且称为新 HTML) 连同本来就存在于动态网页上的非动态内容和客户端脚本程序一起发送给客户端浏览器。

1.5 小结

本章主要介绍了学习 ASP 网络编程之前需要学习或者复习的一些预备知识, 如静态网页和动态网页的概念, 服务器端动态技术的概念, 关于 ASP 等网络的概念, Web 应用程序的开