

“东北林业大学优秀教材及学术专著
出版与奖励专项资金”资助出版



ASP.NET 程序设计

ASP.NET CHENGXU SHEJI

主 编 李瑞改
副主编 王阿川 李林辉
主 审 王霓虹



东北林业大学出版社
Northeast Forestry University Press

“东北林业大学优秀教材及学术专著
出版与奖励专项资金”资助出版

ASP.NET 程序设计

主 编 李瑞改

副主编 王阿川 李林辉

主 审 王霓虹

常州大学图书馆
藏书章

东北林业大学出版社
Northeast Forestry University Press

• 哈尔滨 •

版权专有 侵权必究
举报电话: 0451-82113295

图书在版编目 (CIP) 数据

ASP.NET程序设计 / 李瑞改主编. — 哈尔滨: 东北林业大学出版社, 2016.12

ISBN 978-7-5674-0971-2

I. ①A… II. ①李… III. ①网页制作工具-程序设计
IV. ①TP393.092.2

中国版本图书馆CIP数据核字 (2017) 第003724号

责任编辑: 任兴华

责任校对: 潘琦

封面设计: 乔鑫鑫

出版发行: 东北林业大学出版社

(哈尔滨市香坊区哈平六道街6号 邮编: 150040)

印 装: 哈尔滨市石桥印务有限公司

开 本: 787mm×1092mm 1/16

印 张: 12.5

字 数: 280千字

版 次: 2017年1月第1版

印 次: 2017年1月第1次印刷

定 价: 25.00元

如发现印装质量问题, 请与出版社联系调换。(电话: 0451-82113296 82191620)

前言

ASP.NET 是微软公司推出的网络应用程序的开发平台,是当前 Web 应用程序开发的主流技术之一。使用 ASP.NET 技术进行 Web 应用程序开发,具有开发效率高、上手容易、开发流程简单等优点,可以大大缩短程序开发周期,这些优势使得 ASP.NET 在 Web 应用程序开发市场上占有较大的份额。因此,学好 ASP.NET 编程对计算机科学与技术及相关专业学生就业具有非常重要的积极作用。

本书以 Microsoft Visual Studio 2010 为开发平台,以 C# 为开发语言,面向计算机科学与技术及相关专业本科生,从 0 开始,全面、细致地介绍了 ASP.NET 的基础知识、特点和具体应用,具体内容包括 ASP.NET Web 应用程序概述、使用 C# 程序 ASP.NET 窗体代码、ASP.NET 内部对象的使用、C# 编程基础、C# 面向对象编程、命名空间及异常处理、ASP.NET 服务器控件、数据验证控件、页面设计技术、ASP.NET 操作数据库、Web 窗体的数据控件、项目与实践,由浅入深,层次清晰,易于理解。为初学者考虑,程序设计中的操作过程以详尽的讲述结合截图来说明,以求读者对操作清清楚楚。在代码设计中尽可能多地给出程序注释,以便于读者学习。

本书由李瑞改任主编,王阿川、李林辉任副主编,本书章节结构框架由李瑞改拟定,各章编写分工如下:第 1 章、第 3~4 章、第 9 章、第 11 章由李瑞改编写;第 2 章、第 5 章由李林辉编写;第 6 章由张锡英编写;第 7~8 章、第 10 章、第 12 章由王阿川编写。

全书由李瑞改统稿,王霓虹审稿,东北林业大学出版社的各位同仁为本书的策划和编写提供了很好的建议及出版方面的支持,在此一并致谢。

由于作者水平有限及时间仓促,书中难免会出现一些错误或不当之处,恳请批评指正。

编者

2016 年 10 月

目 录

1	ASP.NET Web 应用程序概述.....	(1)
1.1	软件编程体系简介.....	(1)
1.2	Web 应用程序概述.....	(3)
1.3	使用 ASP.NET 创建 Web 应用程序.....	(9)
1.4	ASP.NET 页面请求、执行、响应过程.....	(12)
1.5	ASP.NET 体系概述.....	(13)
2	使用 C# 编写 ASP.NET 窗体代码.....	(18)
2.1	C# 简介.....	(18)
2.2	在 ASP.NET 页面中编写 C# 代码.....	(23)
3	ASP.NET 内置对象的使用.....	(26)
3.1	Web 窗体页面管理与 Page 对象.....	(26)
3.2	ASP.NET 中的内置对象.....	(28)
3.3	ASP.NET 中的 Web 应用程序状态管理.....	(34)
3.4	Web 应用程序对象和 Global.asax 文件.....	(42)
3.5	ASP.NET 跟踪与程序调试.....	(45)
4	C# 编程基础.....	(49)
4.1	C# 语言基础.....	(49)
4.2	各种语句结构.....	(61)
4.3	数组.....	(65)
4.4	类.....	(71)
4.5	对象的构造和析构.....	(78)
5	C# 面向对象编程.....	(81)
5.1	什么是面向对象.....	(81)
5.2	封装.....	(82)
5.3	继承.....	(86)
5.4	多态.....	(92)

6	命名空间与异常处理	(96)
6.1	命名空间	(96)
6.2	System 命名空间	(98)
6.3	System.Collections 命名空间	(99)
6.4	异常处理	(104)
7	ASP.NET 服务器控件	(106)
7.1	ASP.NET Web 服务器控件的共有属性	(106)
7.2	简单控件	(107)
7.3	文本框控件	(109)
7.4	按钮控件	(111)
7.5	单选控件和单选组控件	(114)
7.6	复选控件和复选组控件	(116)
7.7	列表控件	(117)
7.8	文件上传控件	(119)
8	数据验证控件	(122)
8.1	数据验证模式	(122)
8.2	数据验证控件概述	(122)
8.3	数据验证控件介绍	(124)
9	页面设计技术	(131)
9.1	母板页与内容页	(131)
9.2	用户控件	(134)
10	ASP.NET 操作数据库	(139)
10.1	ADO.NET 体系结构	(139)
10.2	ADO.NET 的常用对象	(141)
10.3	DataSet	(146)
11	Web 窗体的数据控件	(149)
11.1	数据源控件	(149)
11.2	数据显示控件	(160)
12	项目与实践	(173)
12.1	系统分析与设计	(173)
12.2	系统数据需求	(179)
12.3	系统的详细设计	(181)
	参考文献	(193)

1 ASP.NET Web 应用程序概述

ASP.NET 技术是 Microsoft Web 开发史上一个重要的里程碑, 使用 ASP.NET 开发 Web 应用程序并维持其运行, 比其他开发语言更加简单。与 Java, PHP 和 Perl 等高级技术相比, ASP.NET 具有方便性、灵活性、生产效率高、安全性高及面向对象等优点, 是目前主流的网络编程技术之一。

1.1 软件编程体系简介

C/S 和 B/S 是当今世界开发模式技术架构的两大主流技术。C/S 是基于操作系统平台的客户机 / 服务器 (Client/Server) 架构, 美国 Borland 公司最早研发; B/S 是基于 WWW 服务的 Browser/Server 架构, 由美国微软公司研发。目前, 这两项技术已被世界各国所掌握, 国内公司以 C/S 和 B/S 技术开发出的产品也很多。这两种技术都有自己一定的市场份额和客户群, 也都有各自的优点和局限性。

1.1.1 C/S 架构

C/S (Client/Server) 结构 (图 1-1), 即客户机和服务器结构, 它是软件系统体系结构, 通过它可以充分利用两端硬件环境的优势, 将任务合理分配到 Client 端和 Server 端来实现, 降低了系统的通信开销。目前大多数应用软件系统都是 Client/Server 形式的两层结构, 由于现在的软件应用系统正在向分布式的 Web 应用发展, Web 和 Client/Server 应用都可以进行同样的业务处理, 应用不同的模块共享逻辑组件; 因此, 内部的和外部的用户都可以访问新的和现有的应用系统, 通过现有应用系统中的逻辑可以扩展出新的应用系统。这也就是目前应用系统的发展方向。

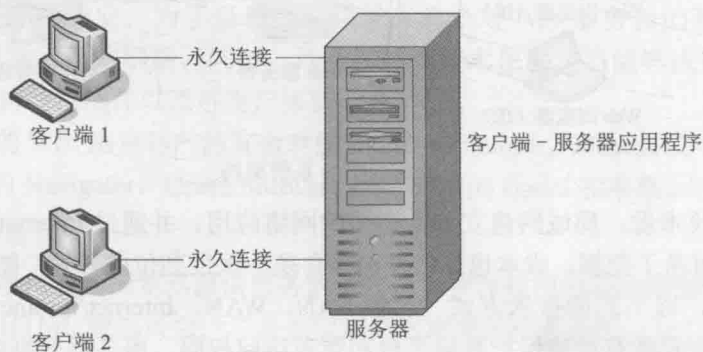


图 1-1 C/S 系统架构

传统的 C/S 体系结构虽然采用的是开放模式，但这只是系统开发一级的开放性，在特定的应用中无论是 Client 端还是 Server 端都还需要特定的软件支持。由于没能提供用户真正期望的开放环境，C/S 结构的软件需要针对不同的操作系统开发不同版本的软件，加之产品的更新换代十分快，已经很难适应百台电脑以上局域网用户同时使用，而且代价高，效率低。

客户端是自行开发的应用程序，服务器端一般是数据库服务器，在客户端程序运行的时间内，连接是永久性的，功能由客户端完成。

C/S 结构程序主要采用的开发语言有 VB, VC, VC#, Delphi, Java 等。

C/S 结构的优点是速度快，安全性高，部署方便，开发过程简单，在局域网和广域网内部都可以使用。

C/S 结构的弊端如下。

(1) 每个客户端都必须安装客户端软件，客户端出问题后需要专人维护，当客户端要升级时，所有用户都必须升级后才能够正常使用，工作量很大，维护成本也很高。

(2) 客户端到服务器端的连接是永久性的，将占用大量资源，服务用户数量少。

(3) 这种程序受地域性限制较强，在一些局域网中、公司内部或商业管理等都可以使用这种 C/S 模式。

1.1.2 B/S 架构

B/S (Browser/Server) 结构，即浏览器和服务器结构 (图 1-2)。它是随着 Internet 技术的兴起，对 C/S 结构的一种变化或者改进的结构。在这种结构下，采用操作系统自带的浏览器作为客户端应用程序。用户工作界面是通过 WWW 浏览器来实现，极少部分事务逻辑在前端 (Browser) 实现，主要事务逻辑在服务器端 (Server) 实现，形成所谓三层 (3-tier) 结构。这样就大大简化了客户端电脑载荷，减轻了系统维护与升级的成本和工作量，降低了用户的总体成本 (TCO)。

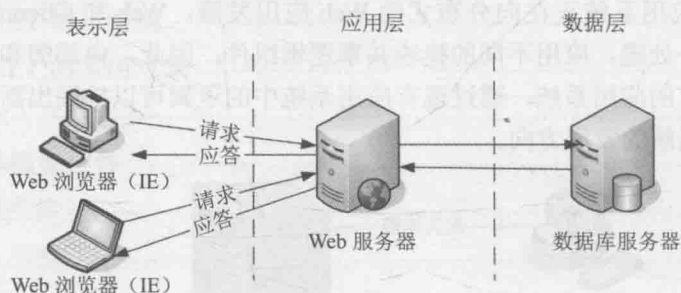


图 1-2 B/S 系统架构

以目前的技术看，局域网建立 B/S 结构的网络应用，并通过 Internet/Intranet 模式下数据库应用，相对易于把握，成本也是较低的。它是一次性到位的开发，能实现不同的人员，在不同的地点，以不同的接入方式 (比如 LAN, WAN, Internet/Intranet 等) 访问和操作共同的数据库；它能有效地保护数据平台和管理访问权限，服务器数据库也很安全。特别是在 Java 这样的跨平台语言出现之后，B/S 架构管理软件更是方便、快捷、高效。

B/S 结构的优点如下。

(1) 维护简单, 当服务器程序更新后再也不需要更新客户端, 只需要维护 Web 应用程序即可。

(2) B/S 结构程序基于 HTTP 协议, 是断开式连接, 程序运行占用服务器端资源较少, 执行效率高, 可以服务于大量的用户。

B/S 结构的弊端: 速度慢、开发过程复杂、部署复杂。

1.2 Web 应用程序概述

Web 应用程序是一种可以通过 Web 访问的应用程序, 指 Web 服务器上所包含的许多静态和动态资源的集合, 对于 ASP.NET 应用程序来说, 这些资源包括静态 HTML 页面、动态的 ASP.NET 页面、ASP.NET 用户控件和 Web 服务等。

1.2.1 Web 应用程序体系结构

Web 应用程序位于 Web 服务器上, 并通过客户端的 Web 浏览器来进行访问。Web 服务器与 Web 浏览器之间的通信需要通过 HTTP 来进行实现。因此, 在开始学习编写 Web 应用程序之前, 有必要对它们进行介绍。

1.2.1.1 Web 浏览器

Web 浏览器是用于通过 URL (统一资源定位符) 来获取并显示 Web 网页的一种软件工具, 它可以以与平台无关的方式显示用 HTML 所编码的 Web 页面, 而 HTML 可将信息呈现在任何操作系统中, 同时对窗口尺寸没有任何约束, 这就是网页与平台无关的原因所在。网页的位置以 URL 指示, 即网页的地址, 以 “http:” 开始的便是通过 HTTP 协议登录的。很多浏览器同时支持其他类型的 URL 及协议, 如以 “ftp:” 开始的是 FTP 协议 (档案传送协议) 等。

Web 浏览器担负着两个方面的职责, 一是收集用户在浏览器中输入的信息, 并以 HTTP 请求的形式向服务器发送用户的页面请求; 二是接收服务器返回的页面响应, 并解释执行在页面上进行显示。Web 浏览器能够显示文本、图像, 并对链接到其他页面的超链接做出相应的响应。为了呈现新的信息, 每个对 Web 服务器的页面请求都会导致 Web 浏览器对显示器的刷新。同时, 许多客户端技术使浏览器能够执行脚本代码, 如 JavaScript, 也支持使用插件以改进用户体验。

当前市面上的 Web 浏览器产品有微软的 Internet Explorer, Mozilla 的 Firefox, 网景公司 (NetScape) 的 Navigator, Opera Software ASA 公司的 Opera 和苹果公司的 Safari 等。

1.2.1.2 Web 服务器

Web 服务器承担着为浏览器提供服务的责任, 主要包括两个方面的职责: 一是向浏览器提供其请求的 HTML 静态页面; 二是处理浏览器请求的动态页面。

Web 服务器产品: 基于 .NET 平台的是 IIS Web 服务器, 基于 Java 平台的主要有

Apache, Tomcat, Jboss, Weblogic 等服务器产品。

1.2.1.3 超文本传输协议

HTTP(Hypertext Transfer Protocol, 超文本传输协议)是 TCP/IP 协议的一个应用层协议, 用于定义 Web 浏览器与 Web 服务器之间交换数据的过程和客户端与服务器通信的格式。网页通常使用超文本标记语言(HTML)文件格式, 并在 HTTP 协议内以 MIME 内容形式来定义。HTTP 协议具有如下特点。

第一, HTTP 是无状态的, 第一次请求完成后, 服务器不会记住客户的状态, 因此第二次请求时, 服务器需要重新读取客户的信息。

第二, HTTP 协议基于请求和应答的模式, 浏览器向服务器发送请求后, 服务器根据浏览器的请求做出不同的应答, 返还给客户端进行显示和处理。

第三, HTTP 协议是断开式的连接, 当浏览器发生 Web 页请求时, HTTP 协议将建立一个与 Web 服务器的连接, 当连接畅通后, 服务器将找到请求的页面, 并将它发送给客户端, 信息发送到客户端后, HTTP 将释放此连接, 这样服务器端将可以接受并服务大量的客户端。

1.2.1.4 IIS 和虚拟目录

Internet Information Services(IIS, 互联网信息服务), 是由微软公司提供的基于运行 Microsoft Windows 平台的互联网服务器产品。IIS 包括 Web 服务器、FTP 服务器、NNTP 服务器和 SMTP 服务器, 分别用于网页浏览、文件传输、新闻服务和邮件发送等方面, 它使得在网络(包括互联网和局域网)上发布信息成了一件很容易的事。

虚拟目录是在 URL 中使用的逻辑目录名, 与服务器上的物理目录相对应。因此, 虚拟目录不是实际文件目录, 虚拟目录为目录的别名。每个 Internet 服务可以从多个目录中发布。通过以通用命名约定(UNC)名、用户名及用于访问权限的密码指定目录, 可将每个目录定位在本地驱动器或网络上。虚拟服务器可拥有一个宿主目录和任意数量的其他发布目录, 其他发布目录称为虚拟目录。

1.2.2 网页开发技术

Web 应用程序中可以包括静态网页和动态网页, 静态网页指利用 HTML 和 CSS 语法构造的无法与使用者进行互动的网页。静态网页只能单纯地显示网页的内容, 无法针对不同的网页浏览状况做出实时响应。而动态网页则是采用动态网页开发技术构造的能够与使用者进行互动的网页。动态网页能够根据不同的浏览状况做出实时响应。

1.2.2.1 静态网页开发技术

早期的 Web 网页基本上都是以 .html 或者 .htm 文件保存的静态网页。在用户浏览前, 网页制作者用 HTML 语言编写网页的内容, 主要包括文本、图像、超链接等内容。当网页被发布后, 页面的内容和外观总是保持不变, 不能针对用户的不同操作进行响应。

静态网页使用 HTML+CSS 来实现, 当客户端浏览器请求浏览服务器端的一个静态网页时, IIS 服务器根据请求网页的资源定位获取本地文件, 并将其作为 HTTP 响应发送到客户端浏览器进行显示。具体工作过程如下(图 1-3)。

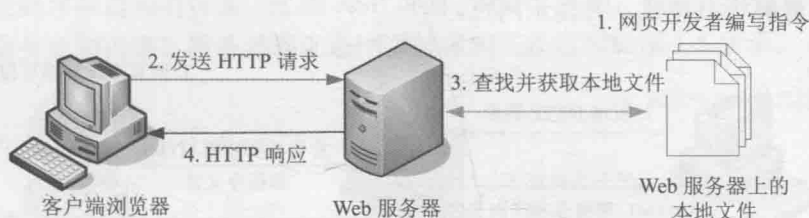


图 1-3 静态网页技术的工作过程

(1) 网页开发者编写由纯 HTML 代码组成的网页，将其以 .htm 或 .html 文件形式保存到 Web 服务器上。

(2) 用户在客户端浏览器的地址栏中输入一个 HTTP 网页请求，该请求以 URL 形式输入，通过网络从浏览器发送到 Web 服务器。

(3) Web 服务器在本地文件系统中定位访问的 .htm 文件，将其转换为 HTML 流。

(4) Web 服务器将 HTML 流通过网络传到客户端的浏览器。

(5) 浏览器解析 HTML 流，并将网页内容进行显示。

静态的网页可以包含文字、图像、声音、视频等多媒体信息，但是不包含任何与客户端用户交互的动态内容，这种简单的静态技术目前仍然在很多 Internet 站点使用，其优点在于信息访问的效率很高，网站的架设与开发相当容易。

1.2.2.2 动态网页技术

随着 Internet 的快速普及，静态的网页内容已经不能满足人们日益复杂的应用需求，动态网页应运而生。开发人员利用动态网页开发技术给静态网页添加动态内容，以满足客户端的不同应用需求，满足不同应用领域的需要。动态网页开发技术主要有两个发展方向：客户端动态网页技术和服务器端动态网页技术。

(1) 客户端网页技术。

客户端动态网页技术是指 Web 服务器把原始的 HTML 页面，以及一组包含了页面逻辑的脚本、组件等一起发送到客户端，这些脚本和组件包含了如何与用户交互并产生动态内容的指令，由客户端浏览器及其附带的插件解析 HTML 页面并执行这些指令。

客户端动态网页一般使用脚本来实现，主要使用 JavaScript, VBScript, ActiveX 控件等结合 HTML 及 CSS 进行开发，实现动态网页开发。除此之外，还有 Microsoft 的 Jscript, Sun 公司的 Java Applet 技术及 Micromedia 公司的 Flash 技术等。客户端动态网页技术的工作过程如图 1-4 所示。

① Web 开发者用 HTML 语言编写网页，并将其保存为 .html 或者 .htm 文件，同时，用脚本语言编写指令，并将指令嵌入 HTML 语言中，也可以以单独文件的形式进行保存后包含到 HTML 文件中；

② 用户在客户端浏览器中输入一个动态网页请求，通过网络传到 Web 服务器上；

③ Web 服务器在接收到客户端发送的请求后，在本地文件系统中定位 .htm 或者 .html 文件，以及 HTML 文件指令中包含的其他文件，并生成 HTML 流；

④ Web 服务器将创建的 HTML 流和其他指令通过网络传递到客户端浏览器；

⑤ 浏览器插件解析指令，并将其返回到 .htm 文件中；

⑥ 浏览器解析 HTML，并显示网页。

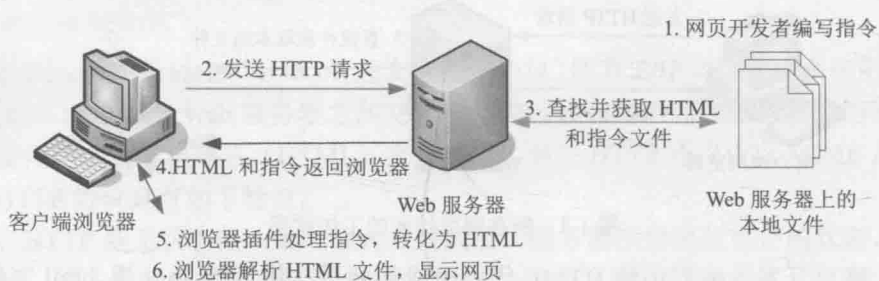


图 1-4 客户端动态网页的获取过程

客户端动态网页技术的主要优点是充分利用了客户端的计算机资源，减轻了服务器和网络上的压力，同时可以很方便地实现基于图形的用户交互界面。首先，因为客户端动态网页技术需要把脚本或者组件下载到客户端计算机中，如果脚本或者组件比较大，那么下载速度就会很慢。其次，现存的每种客户端浏览器可以按照不同的方法解析客户端脚本代码，用户几乎没有办法保证所有的浏览器按照同样的方法解析代码。最后，组件或者脚本下载带来的一对很难调和的矛盾就是对于开发者来讲，如何保证源代码的保密，而对于客户端用户来说，如何确保下载到自己计算机上的脚本或者组件中的代码不包含恶意攻击的代码。因此，客户端动态网页技术在 Web 应用程序上的应用一般局限在显示动画、验证用户输入等方面。图 1-5 显示了一个常规情况下，客户端脚本技术的主要应用情景。

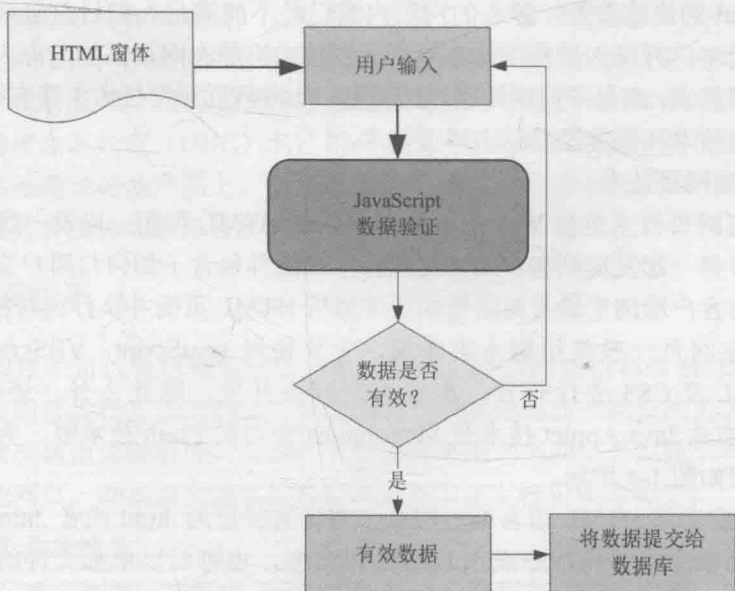


图 1-5 客户端脚本的用法

(2) 服务器端网页技术。

服务器端网页技术是指在 Web 服务器端，根据客户端浏览器的不同请求，动态地生成相应的内容，然后发送给客户端浏览器。服务器端动态页面允许双向的交流，需要服务

器执行相关代码才能返回的页面,比如 ASP 页面、ASPX 页面,动态内容能够为单个用户的每次访问量身定制内容。服务器端动态网页技术的工作过程如图 1-6 所示。



图 1-6 服务器端动态网页技术的工作过程

- ① Web 开发者编写、创建 HTML 的指令,将这些指令保存在一个文件中;
- ② 用户在客户端浏览器中输入一个 HTTP 网页请求,通过网络传送到 Web 服务器上;
- ③ Web 服务器查找并定位指令文件;
- ④ Web 服务器解析并处理动态代码,根据指令创建 HTML 流;
- ⑤ Web 服务器将新创建的 HTML 流通过网络传到客户端浏览器中;
- ⑥ 浏览器解析 HTML,显示网页。

使用服务器端动态网页技术,所有指令在传回浏览器之前都在服务器上进行处理,用户不再需要考虑指令在哪里处理。这种技术的一个最关键的点就是只有 HTML 被传回浏览器,这意味着原始页面代码被隐藏在服务器中,开发者不需要去适应各种复杂的客户端计算机环境,只需要集中在服务器端处理请求就可以了。因此,它便于采用各种复杂的、结合特定服务器平台的技术,并形成成熟可靠的标准,为应用程序开发提供了良好的环境。服务器端动态网页技术的缺点在于对 Web 服务器端的硬件要求比较高,容易产生性能上的瓶颈。

服务器端技术安装在 Web 服务器上,所有网页运行在服务器上。而使用客户端技术,网页运行在浏览器中。因此,在服务器脚本传回浏览器前,Web 服务器必须首先将其转换为 HTML;如果没有转换,浏览器将不能理解服务器端代码,那么浏览器中也就不会显示信息。

1.2.2.3 ASP.NET 动态编程技术

ASP.NET 是一项功能强大、非常灵活的服务器端技术,用于创建动态 Web 页面。它是一个已编译的、基于 .NET 的环境,可以用任何与 .NET 兼容的语言开发应用程序。自从 .NET Framework 1.0 在 2002 年初首次发布以来,Microsoft 花了大量精力和时间来开发 ASP.NET,它是 .NET Framework 的一部分,可以用来构建 Web 应用程序。

(1) ASP.NET 的历史。

早期的 Web 程序开发是一个十分烦琐的事情,一个简单的动态页面就需要编写大量的代码才能完成。

1996 年,Microsoft 推出了 ASP (Active Server Page, 活动服务器页面,现在人们常称之为传统 ASP) 1.0 版。它允许采用 VBScript/JavaScript 等脚本语言编写动态代码,允许将动态代码嵌入 HTML 中,从而使得设计动态 Web 页面的工作变得简单。ASP 能够

通过内置的组件实现强大的功能（如 Cookie）。ASP 最显著的贡献就是推出了 Active X Data Objects (ADO)，它使得程序对数据库的操作变得十分简单。

1998 年，微软发布了 ASP 2.0 和 IIS 4.0。与 ASP 1.0 相比，2.0 版最大的改进是外部的组件需要初始化。用户能够利用 ASP 2.0 和 IIS 4.0 构建各种 ASP 应用，而且每个组件有了自己单独的内存空间，可以进行事务处理。

随后，微软在 Windows 2000 Server 系统中提供了 IIS 5.0 和 ASP 3.0。此次升级，最主要的改变就是把很多事情交给 COM+ 来做，效率比以前的版本更高，而且更稳定。

2002 年，ASP.NET 1.0 的发布意味着从过去的 Microsoft 技术向构建 ASP Web 站点的飞跃。ASP.NET 1.0 在结构上与传统的 ASP 版本截然不同，几乎完全是基于组件和模块化的。ASP.NET 1.0 及相关的 Visual Studio.NET 2002 的引入给开发人员带来了很大的方便，提供了代码隐藏类的概念，把逻辑代码和表现页面分离开来，使用户很容易使用后台代码来控制页面的逻辑功能。

2003 年，Microsoft 公司发布了 Visual Studio.NET 2003，提供了在 Windows 操作系统下开发各类基于 .NET 框架的全新的应用程序开发平台（称为 .NET 1.1）。

2005 年 11 月，Microsoft 发布了 Visual Studio 2005 和 ASP.NET 2.0。它修正了以前版本中的一些 Bug 并在移动应用程序开发、代码安全以及对 Oracle 数据库和 ODBC 的支持等方面都做了很多改进。

2007 年 11 月发布的 Visual Studio 2008 和 ASP.NET 3.5 中添加了一系列很酷的新功能，主要包括 LINQ 以及 AJAX 框架整合。2008 年 8 月，Microsoft 发布了用于 Visual Studio 和 .NET Framework 的 Service Pack 1，其中引入了一些重要的新功能，如 ADO.NET Entity Framework 和动态数据。

2010 年推出 Visual Studio 2010 和 ASP.NET 4。目前最新版本是 2015 年 7 月 20 日发布的 Visual Studio 2015 和 ASP.NET 4.6。

（2）ASP.NET 的开发环境。

ASP.NET 的开发环境包括编程语言和集成开发环境两个方面，ASP.NET 开发时需要使用的语言是 Visual Basic.NET 或 C#，这两种语言都是 .NET 环境下的程序设计语言。

目前最流行的 Windows 平台应用程序开发环境为 Visual Studio 2010，但并不是必须使用 .NET 集成开发环境才能进行 ASP.NET 程序设计。因为 ASP.NET 文件实际上是一个纯文本文件，编译工作是在用户向服务器第一次发出对该文件的 HTTP 请求时由 Web 服务器进行的，并不是由 Visual Studio 完成的，所以从理论上讲，用记事本或其他文本编辑器就可以编写 ASP.NET Web 应用程序。但是，集成开发环境为开发人员提供了大量有助于快速创建复杂应用程序的工具，因此，绝大多数开发人员还是倾向于使用集成开发环境进行程序开发的。本教材主要采用 Visual Studio 2010 作为集成开发系统。

Visual Studio 2010 为程序员提供的功能主要有网页设计功能，代码编辑功能，调试、内置 Web 服务器功能，支持 AJAX，jQuery，MVC 等主流开发技术。

ASP.NET 的编程特点：ASP.NET 代码是编译执行、支持事件编程、支持页面与代码的分离。

1.3 使用 ASP.NET 创建 Web 应用程序

使用 Visual Studio 2010（以下简称 VS 2010）可以轻松地创建各类应用程序，下面简要介绍 VS 2010 集成开发环境，并介绍如何创建 Web 应用程序。

1.3.1 VS 2010 集成开发环境

Visual Studio 是微软公司推出的集成开发环境，是目前最流行的 Windows 平台应用程序开发环境。VS 2010 版本于 2010 年 4 月 12 日上市，其集成开发环境（IDE）的界面被重新设计和组织，变得更加简单明了。VS 2010 同时带来了 NET Framework 4.0，Microsoft Visual Studio 2010 CTP（Community Technology Preview，CTP），并且支持开发面向 Windows 7 的应用程序。除了 Microsoft SQL Server，它还支持 IBM DB2 和 Oracle 数据库。

首次启动 VS 2010 时，选择好编程语言，在本教材中，选择 C# 作为编程语言。图 1-7 即为 VS 集成开发环境的截图。各主要子窗体主要包括设计器视图或代码视图、工具箱、解决方案资源管理器、属性子窗体、程序错误提示窗体等。

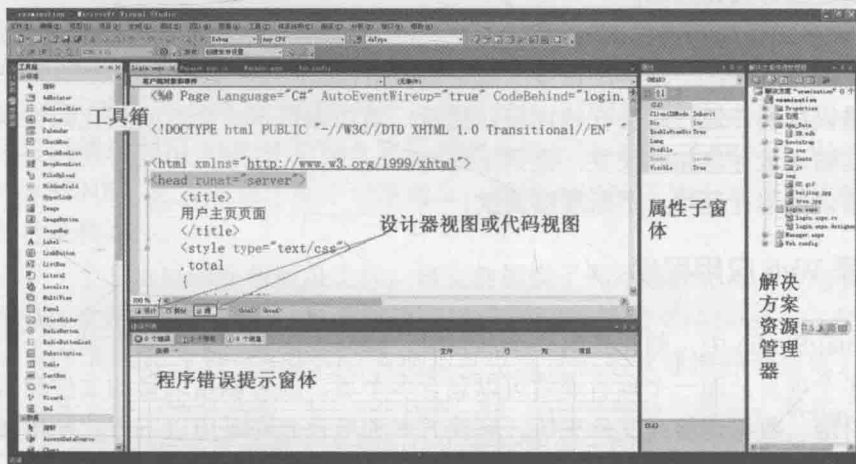


图 1-7 VS 集成开发环境界面

（1）设计器视图与代码视图。

设计器视图与代码视图是程序设计最常用的两个子窗口。设计器视图用来设计 Windows 窗体或 Web 窗体，代码视图用于设计程序源代码。

在设计器视图中可以为 Windows 或 Web 界面添加控件，代码视图实质上是一个纯文本编辑器，两者之间可以通过资源管理器上的按钮进行切换。

（2）解决方案资源管理器。

使用 Visual Studio.NET 开发的每一个应用程序叫解决方案，每一个解决方案可以包含一个或多个项目。

一个项目通常是一个完整的程序模块，一个项目可以有多个项。“解决方案资源管理

器”子窗口显示 Visual Studio.NET 解决方案的树型结构。在“解决方案资源管理器”中可以浏览组成解决方案的所有项目和每个项目中的文件，可以对解决方案的各元素进行组织和编辑。

(3) 属性子窗口。

“属性”子窗口用于设置解决方案中各个子项的属性，当选择设计器视图、解决方案、类视图中的某一子项时，“属性”子窗口将以两列表格的形式显示该子项的所有属性。

注意“属性”子窗口上面的事件按钮。单击“事件”按钮，窗口将显示被选择窗体或控件的事件列表。

(4) 工具箱。

“工具箱”用于向 Windows 应用程序或 Web 应用程序添加控件。

“工具箱”使用选项卡分类管理其中的控件，打开“工具箱”将显示 Visual Studio 项目中使用的各个不同的控件列表。根据当前正在使用的设计器或编辑器，“工具箱”中可用的选项卡和控件会有所变化。

(5) 子窗口的操作。

在主区域显示的窗口，以选项卡的形式叠放在一起。用右上角关闭按钮关闭当前窗口。

主区域周围的窗口，有停靠与浮动两种情况，停靠又有隐藏与固定显示两种状态，这两种状态通过图钉按钮控制。

(6) 程序错误提示窗体。

程序错误有两种：一是语法错误；二是逻辑错误。

语法错误在编译运行时会在错误窗口提示，双击该提示，会返回到代码错误的行，改正后，继续编译运行，如此反复，直到正确为止。

逻辑错误可逐行运行，不断观察调试。

1.3.2 创建 Web 应用程序

在 Visual Studio 中，创建任何一个项目时，都得先创建一个解决方案。一个解决方案可以包含多个项目，而一个项目通常可以包含多个项，项可以是项目的文件和项目的其他部分，如引用、数据连接或文件夹等。解决方案和项目允许采用以下方式使用集成开发环境 (IDE)。

第一，作为一个整体管理解决方案的设置或管理各个项目的设置；

第二，在集中精力处理组成开发工作的项的同时，用“解决方案资源管理器”处理文件管理细节；

第三，添加对解决方案中的多个项目有用或者对该解决方案有用的项，而不必在每个项目中引用该项；

第四，处理与解决方案或项目独立的杂项文件。

下面将详细介绍 C# 创建 Web 应用程序项目和 Web 窗体的过程。

(1) C# 创建 Web 应用程序项目。

启动 VS 2010 后，在程序界面内点击“文件”菜单下的“新建”子菜单下的“项目”，将弹出如图 1-8 所示的“新建项目”对话框，在“已安装的母版”下选择“Visual C#”下

的“Web”，在右侧应用程序母版列表中选择“ASP.NET 空 Web 应用程序”，然后点击“浏览”按钮选择项目的存放位置如“D:\aspnet”，输入项目名称如“test01”，最后点击“确定”按钮创建 Web 应用程序。

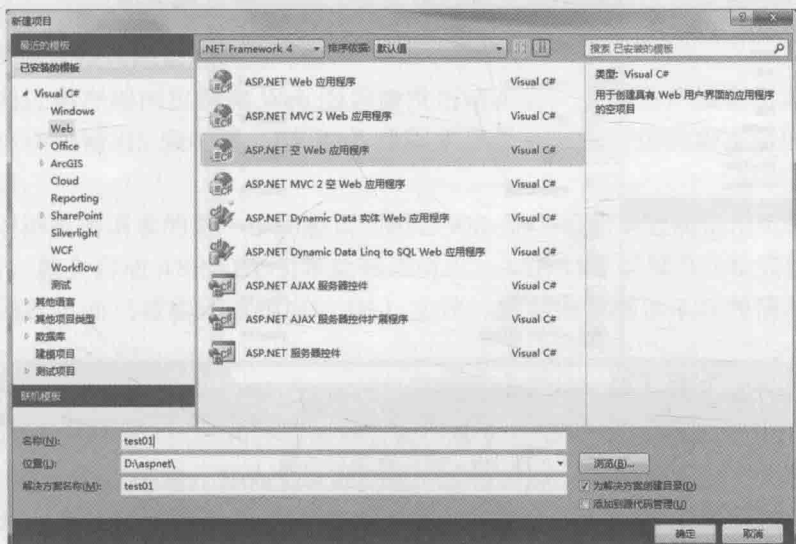


图 1-8 新建项目对话框

使用 ASP.NET 空 Web 应用程序母版创建的项目是一个非常干净的项目，只拥有一个 Web.config 文件。它基本上就是一个空架子，没有任何可执行的文件。

用户还可以通过利用 ASP.NET Web 应用程序母版创建项目。ASP.NET Web 应用程序母版正好与 ASP.NET 空 Web 应用程序母版相反，创建 ASP.NET Web 应用程序时其中预先生成了一些目录和文件。

它包含了一个 Site.Master 母版页文件，该文件提供了网站总的布局（含有页眉、页脚等），在 Styles 文件夹里使用了一个含有所有样式的 CSS 样式文件 Site.css。在 Scripts 文件夹里面内含了 jQuery 文件（ASP.NET AJAX 可以通过脚本管理控件来提供）。在根目录中，它还包含了基于母版页的“Default.aspx”和“About.aspx”网页。除此之外，还在 Account 文件夹内包含并实现了基于表单的认证系统的若干网页，可用来登录、注册和改变用户的密码。在这里，程序员不用编写任何代码或配置文件就可以运行这个项目，得到一个运行正常的网站。

（2）创建窗体。

在项目 test01 的解决方案资源管理器中右键单击项目名称，在弹出的快捷菜单中选择“添加”->“新建项”，将弹出如图 1-9 所示的“添加新项”对话框，选择“Visual C#”下的“Web”，在中间的项目列表中选择“Web 窗体”，在下方名称后对应的文本框中输入新建窗体的名称，如“HelloWorld.aspx”后点击“添加”按钮，完成窗体的创建任务。

（3）编写代码。

在项目的解决方案资源管理器界面中双击 HelloWorld.aspx 文件，在“设计”视图中的空白位置双击，将打开代码隐藏类文件，即 HelloWorld.aspx.cs。在 protected void Page_Load(object sender, EventArgs e) 函数体中输入如下代码：