



普通高等教育“十一五”国家级规划教材

高职高专计算机系列

ASP.NET 2.0 实用教程 (C#版)

崔良海 主编



普通高等教育“十一五”国家级规划教材

高职高专计算机系列

ASP.NET 2.0 实用教程（C#版）

崔良海 主 编

 人民邮电出版社
POSTS & TELECOM PRESS

北 京

图书在版编目 (CIP) 数据

ASP.NET 2.0 实用教程: C#版/崔良海主编. —北京: 人民邮电出版社, 2008.5

普通高等教育“十一五”国家级规划教材. 高职高专计算机系列

ISBN 978-7-115-17817-6

I. A… II. 崔… III. ①主页制作—程序设计—高等学校: 技术学校—教材②C 语言—程序设计—高等学校: 技术学校—教材 IV. TP393.092 TP312

中国版本图书馆 CIP 数据核字 (2008) 第 032402 号

内 容 提 要

本教材主要介绍以 C#语言为基础、使用 ASP.NET 进行 Web 程序开发与应用的技术。本教材通过大量的实例, 讲解了采用 .NET 所提供的丰富控件, 以可视化与面向对象技术进行 Web 页面的设计与制作。

本教材内容翔实、实例丰富, 介绍了 ASP.NET 的大量基础知识与实用技术。全书共 9 章: 第 1 章为 ASP.NET 基础, 第 2 章为 C#编程基础, 第 3 章为 Web 窗体应用, 第 4 章为 ASP.NET 内置对象模型, 第 5 章为 ASP.NET 控件, 第 6 章为数据绑定技术, 第 7 章为 Web 应用配置, 第 8 章为 Web 服务, 第 9 章介绍一个实例——学生信息管理系统。每章均附有小结、习题与上机操作, 便于读者加深对相关知识与技能的理解。

本教材适合作为高等职业院校和应用型本科院校计算机相关专业的教材, 也可供使用 ASP.NET 技术进行 Web 页面设计与制作的技术人员参考。

普通高等教育“十一五”国家级规划教材

高职高专计算机系列

ASP.NET 2.0 实用教程 (C#版)

-
- ◆ 主 编 崔良海
策划编辑 潘春燕
责任编辑 李 凯
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京铭成印刷有限公司印刷
新华书店总店北京发行所经销
 - ◆ 开本: 787×1092 1/16
印张: 16.25
字数: 395 千字 2008 年 5 月第 1 版
印数: 1—3 000 册 2008 年 5 月北京第 1 次印刷

ISBN 978-7-115-17817-6/TP

定价: 25.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010)67171154

前 言

目前,用户在网上时,已不满足于仅仅被动性地阅读信息,而是迫切希望能够亲身参与其中,进行各种主动性的交流,在网上迅捷方便地发布各种信息。早期的 ASP 技术使这种需求得到了初步的实现。随着 Internet 的发展与普及,Microsoft 公司推出了一种更为先进的 .NET Framework (.NET 框架) 开发平台,基于此框架的 ASP.NET 使 Web 开发变得更加方便与快捷。这项技术已成为当前 Web 开发的最先进技术之一,而本书介绍的 ASP.NET 2.0 更是在原版本基础上有了进一步的完善与改进,使许多原先十分复杂的开发与应用变得轻而易举。

本书以 C#语言为基础来阐述 ASP.NET 的开发过程。C#是从 C 和 C++发展而来的,它继承了 C++语言的众多优点,是面向对象的高级程序设计语言,功能强大,语言简洁、高效,与 Web 技术紧密结合,并具备完整、安全的错误处理机制。C#是 Microsoft .NET 的核心编程语言,能够最大限度地发挥 Visual Studio.NET 平台的威力,使用户能够在该平台上快速开发各种类型的应用程序。

本书的主要特点就是以实例来一步步地阐述如何用 ASP.NET 技术进行 Web 应用程序的开发。Web 窗体的开发可以通过单纯地编制代码进行,而本书注重可视化与面向对象的编程方法,充分利用 ASP.NET 丰富的控件,在设计界面中制作与编程。ASP.NET 最显著的特点就是控件的应用,与以前的 ASP 程序制作一样,虽然在文本文件的编辑状态下也可以进行 ASP.NET 文件的编制,然而可视化与面向对象编程应该是学习的主要方向,也是本教材的显著特点。书中涉及以 C#为基础使用 ASP.NET 技术进行 Web 开发应用的大量基础知识和关键技术。本书注重实际运用,除了阐述必不可少的基础知识之外,着重通过实际操作的案例来为读者提供掌握 ASP.NET 技术的指南。

本书共 9 章,第 1 章讲述了 ASP.NET 编程的基础与开发环境的安装与设置,第 2 章讲述了 Microsoft 公司推出的 C#程序设计语言基础;第 3 章讲解了使用 ASP.NET 技术进行 Web 窗体开发的详细过程;第 4 章阐述了 ASP.NET 的内置对象与实际使用方法;第 5 章通过实例介绍了如何使用控件进行 Web 开发的技术;第 6 章介绍了 ASP.NET 如何访问 Web 数据库;第 7 章、第 8 章和第 9 章通过一个完整的实例来使读者更深入地理解 ASP.NET 的运用。

本书第 1 章~第 4 章由崔良海编写,第 5 章、第 8 章由沙俐敏编写,第 6 章、第 9 章由刘琰编写,第 7 章由陈跃平编写;全书由崔良海统稿并审定。

由于编者水平有限,书中难免存在不足之处,殷切希望广大读者给予批评与指正。

编 者
2008 年 3 月

目 录

第 1 章 ASP.NET 基础	1
1.1 Web 编程基础	1
1.1.1 静态 HTML 的运行与制作	1
1.1.2 动态交互 HTML 的工作原理	2
1.2 .NET 概述	5
1.2.1 .NET 框架	5
1.2.2 ASP.NET	7
1.3 ASP.NET 的运行环境与安装	8
1.3.1 安装 IIS	9
1.3.2 IIS 的设置	11
1.3.3 安装 SQL Server 2000	14
1.3.4 安装 Visual Studio 2005	17
小结	20
习题与上机操作	20
第 2 章 C#编程基础	21
2.1 C#编程语言特点	21
2.2 第一个 C#程序	22
2.3 C#基本知识	25
2.3.1 命名空间	25
2.3.2 类和 Main 函数	26
2.3.3 基本数据类型	27
2.3.4 注释与转义符	30
2.3.5 操作符优先级	31
2.4 条件控制语句	31
2.4.1 if 语句	32
2.4.2 ?条件控制语句	33
2.4.3 switch 语句	33
2.5 循环控制语句	35
2.5.1 for 语句	35

2.5.2	foreach 语句	36
2.5.3	while 语句	37
2.5.4	do 语句	38
2.6	变量类型转换与字符串的处理	39
2.7	面向对象编程	43
2.8	C#编程实例	47
	小结	50
	习题与上机操作	50
第 3 章	Web 窗体	51
3.1	Web 窗体概述	51
3.2	一般 ASP 程序	52
3.3	ASP.NET 可视化 Web 窗体编制	53
3.4	ASP.NET 的优越性	59
3.5	ASP.NET Web 窗体制作基础	61
3.5.1	图片变化	63
3.5.2	网页中表格的生成	64
3.5.3	记事日历	67
3.6	ASP.NET 编程中的调试与对错误的处理	71
	小结	72
	习题与上机操作	72
第 4 章	ASP.NET 内置对象	73
4.1	ASP.NET 内置对象概述	73
4.2	Response 对象	74
4.2.1	输出文本	75
4.2.2	页面跳转	75
4.2.3	停止输出	75
4.3	Request 对象	76
4.4	Server 对象	77
4.5	Application 对象	78
4.6	Session 对象	81
4.7	Cookie 对象	83
	小结	84
	习题与上机操作	85
第 5 章	ASP.NET 控件	86
5.1	ASP.NET 控件的概述	86
5.1.1	ASP.NET 的控件设置	86
5.1.2	ASP.NET 2.0 控件的应用类别	87
5.1.3	ASP.NET 控件的使用方法	88
5.2	注册实例制作	88

5.2.1 功能介绍	88
5.2.2 实例流程	89
5.2.3 实例制作过程	89
5.3 ASP.NET 其他实用控件的运用	100
5.3.1 其他常用控件	100
5.3.2 登录控件	109
5.3.3 登录控件综合实例	110
5.3.4 WebParts 控件	113
小结	119
习题与上机操作	120
第 6 章 数据绑定技术	121
6.1 ASP.NET 2.0 的新功能	121
6.2 数据源控件	122
6.2.1 SqlDataSource 数据源控件	122
6.2.2 AccessDataSource 数据源控件	127
6.2.3 XmlDataSource 数据源控件	128
6.2.4 SiteMapDataSource 数据源控件	129
6.2.5 ObjectDataSource 数据源控件	130
6.3 数据绑定控件	135
6.3.1 GridView 控件	136
6.3.2 DetailsView 数据绑定控件	139
6.3.3 FormView 数据绑定控件	143
6.3.4 DataList 数据绑定控件	145
6.3.5 Repeater 数据绑定控件	154
6.4 数据库的基本操作	156
6.4.1 建立数据库连接	157
6.4.2 数据查询显示	157
6.4.3 数据添加	158
6.4.4 数据修改	158
6.4.5 数据删除	159
小结	159
习题与上机操作	159
第 7 章 Web 应用配置	160
7.1 Web 应用系统概述	160
7.2 Global.asax	161
7.2.1 Global.asax 运行特点	161
7.2.2 Global.asax 的配置	162
7.2.3 Global.asax 的配置实例	164
7.3 Web.Config	166

7.3.1 Web.Config 运行特点	166
7.3.2 Web.Config 的配置	167
7.4 Web.Config 的配置实例	173
7.4.1 实例 1	173
7.4.2 实例 2	175
小结	177
习题与上机操作	177
第 8 章 Web 服务	178
8.1 Web Service 概述	178
8.1.1 XML, SOAP 与 Web Service	178
8.1.2 Web 服务的体系结构	179
8.1.3 构建 ASP.NET Web 服务	180
8.2 一个简单的 Web 服务的实例	183
8.3 访问 Web Service	185
小结	188
习题与上机操作	188
第 9 章 学生信息管理系统	189
9.1 系统功能介绍	189
9.1.1 功能结构分析	189
9.1.2 网站架设	190
9.2 数据库设计	191
9.2.1 关系图	191
9.2.2 表结构	192
9.2.3 存储过程	198
9.3 公共类	199
9.3.1 公共类的编写	199
9.3.2 ASP.NET 配置文件 Web.Config	201
9.3.3 站点地图的制作	202
9.4 学生信息管理系统的制作	203
9.4.1 用户登录	203
9.4.2 网站框架的制作	205
9.4.3 教师管理的制作	210
9.4.4 学生管理的制作	220
9.4.5 课程管理的制作	238
9.4.6 成绩管理的制作	245
9.4.7 班级管理的制作	247

第 1 章 ASP.NET 基础

教学提示：Visual Studio.NET 是基于 .NET 框架的集成开发环境，它集源程序编辑、编译、链接及项目管理和程序发布等于一体。本书所阐述的 Visual Studio 2005 是在 Visual Studio .NET 的基础上，将功能进一步完善，使其更适用于构建高性能、多层次的 Windows、Web 和移动设备应用程序。Visual Studio 2005 使用集成的 Visual 数据库工具来设计数据库、表、存储过程；同时还能设计、调试和部署多层应用程序，并能集成 XSLT 调试等。

要掌握好 .NET 编程技术，首先要构造一个运行与调试程序的环境，所以读者要在计算机中安装 Visual Studio 2005 与 SQL Server 2000，以便进一步边学习边操作。

本章主要介绍 Web 编程基础.NET 概述、ASP.NET 的运行环境与安装等内容。

1.1 Web 编程基础

1.1.1 静态 HTML 的运行与制作

早期的 Web 编程主要就是 HTML 网页的制作，HTML 是 Hyper Text Markup Language 的缩写，中文意思是超文本标记语言。它是在 WWW 上建立超文本文件的语言，通过标记和属性对一段文本的语言进行描述。

HTML 网页支持在文本中嵌入图像、声音、动画等不同格式的文件，使网页丰富多彩；但它并不像 Word 那样将有关内容真正包含在文件中，而是只作出一定的标记。它在运行时能自动指向网页上的任何相关文件，可以是图像、声音、动画等多媒体文件，也可以是另一网页。

HTML 还具有强大的排版功能，利用 HTML 可以制作出各种版式的网页，能够设置网页的背景、色彩、字体、字形与段落格式等各种属性。

HTML 最重要的功能是可以设置超文本链接，即能指向互联网上任何一台计算机中的相关文件，用户在浏览网页时，只要用鼠标单击超文本链接即可直接定位到相关内容，便于阅读和检索定位。

HTML 标准的内容非常复杂，但超文本文件本身却比较简洁，它全部由纯文本字符组成，用来描述文本内容本身和超文本文件之间的各种连接关系。HTML 能适应 Internet 上的不同软硬件系统和各种操作环境，它可以畅通无阻地跨越各种平台。因此 HTML 是 Internet 上的通用语言，它具有如下的特点。

- 简易性: HTML 的版本升级采用超集方式, 从而更加灵活方便。
- 扩展性: HTML 的广泛应用使得用户需要增强功能并增加标识符, HTML 采取子类元素的扩展方式, 从而为系统扩展提供了保证。

- 与平台无关性: HTML 可以应用在广泛的平台上, 无论是 PC 还是 MAC, HTML 都可以在其上运行; 而且 HTML 还兼容 Windows、UNIX 等多种操作系统。

静态网页是一个存储于 Web 服务器的文件, 设计人员在创作时就决定了网页的内容。由于网页的内容不会变化, 所以用户每次对静态网页的访问都返回相同结果。

静态网页的主要优点在于其简单、可靠的性能。由于静态网页的制作过程较为简单, 设计人员不需要掌握编程即可创建。而且静态网页经过测试之后, 便可以无限制地被使用。浏览器可以快速存取静态网页, 同时通过把网页存储在本地内存或磁盘中, 以加快对静态网页的访问速度。

静态网页的主要缺点是不够灵活, 当数据发生变化时必须重新设计网页; 而且, 修改工作是很耗费时间的, 因为它需要人工操作。因此, 静态网页对于频繁变化的需求并不适合。

1.1.2 动态交互 HTML 的工作原理

目前, 网页已从最早的单纯文字形式发展到多媒体集成形式, 如加入动画、音频与视频等信息。先进的 CGI (公共网关接口) 技术能使用户与网页之间具有交互能力, 用户不再单纯地被动接收信息, 还可以主动提出自己的要求, 得到自己所需要的信息。如今, 网站还可以将数据库中的大量数据经过筛选后, 有针对性地提供给上网的用户。

实现动态交互网页的关键技术就是 CGI。概括地说, 采用该技术能够使用户在客户端的浏览器上输入数据, 系统自动将数据提交到 Web 服务器上运行, 再将用户所需要的数据信息返回到用户的浏览器, 这样就可以实现动态交互网页。人们经常使用的各类查询 (如搜索引擎) 就是一个 CGI 程序。

CGI 是一个标准的名字, 它定义了 Web 服务器与其他应用或其他程序交互作用的规范。CGI 并不指定在这种交互作用中必须使用的语言或方法, 所以可以从 Web 服务器的操作系统中动态选用相关的工具。

CGI 提供了一种标准的接口, 使得 Web 服务器能够执行其他程序, 并把文本、图形或声音组织进输出文件之中, 然后发送给客户端 Web 浏览器。这样, 服务器与 CGI 程序结合起来工作, 大大扩充了 WWW 的功能, 并使效率得到提高。

例如, CGI 程序能够处理表单输入, 查找数据库中的记录, 发送电子邮件, 实现网页访问计数器以及许多其他功能。CGI 技术使得网页更为生动活泼, 用途更加广泛。

CGI 技术允许用户在 Web 服务器上发布 CGI 程序, 实际上也就意味着允许 Web 用户在服务器上运行程序, 这就带来了 CGI 的安全性问题, 这需要网页设计人员给予足够的注意, 并采取一定的防范措施。

浏览器与 Web 服务器 (程序) 之间的通信采用超文本传送协议 (HTTP), 一次 HTTP 交易操作由 4 个过程组成, 即建立连接—请求—响应—关闭连接。也可以进一步把上述几个过程更具体细分为如下几个步骤。

- ① 当点击某个链接时, 浏览器解析出统一资源定位符 (URL), 并寻找该服务器, 如寻找成功就与之建立连接。

- ② 浏览器提出服务请求，即把 URL 的其余部分提交给服务器。
- ③ 服务器把 URL 翻译成路径和文件名，并搜索该文件。
- ④ 服务器把搜索到文件发送给浏览器。
- ⑤ 虽然此时客户机与网络仍处于接通状态，但是服务器中断与浏览器的连接。
- ⑥ 浏览器显示返回的文件。

如果再点击另一链接，就再重复一遍上述过程。每次客户端与服务器进行连接时都好像从未访问过这个网页一样，每次请求只产生一个文档，服务器中并不记录用户访问的历史情况和状态，因此这种连接叫无状态连接，它是由 HTTP 1.0 规定的。

无状态连接的协议对于检索和浏览信息比较方便，减去了在服务器中保存各个客户端的浏览器状态信息的负担，而且能够尽快释放服务器来为其他浏览器的请求服务，从而可以大大提高服务器的工作效率。

但是，这种无状态连接协议不适于动态信息和互动式的应用，因为首先服务器程序只能按照 URL 的要求读取指定的文件来传送给浏览器，或者启动另一程序并把它的输出传送给浏览器。也就是说，服务器程序只能读取文件，而无写入或修改文件的功能。另外，这种动态交互网页需要调用的对象往往要涉及各不相同的其他程序，而一个 Web 服务器程序不可能设计成能够与所有其他程序相连接。

因此，需要设计一种通用的接口来解决 Web 服务器（程序）与其他程序或应用之间的信息沟通问题，这种通用接口就是 CGI。CGI 可以解决上述的信息沟通与转换问题，而不能解决网络互连时网络协议之间的转换问题。

CGI 是一种规范或标准。用户采用某种语言工具，例如 C，Perl，ASP，PHP 等，按照 CGI 标准以及具体应用的要求编写出来的程序一般叫做 CGI 程序或 CGI 脚本。CGI 程序不仅可以处理动态 HTML 文档，还可以按要求发送电子邮件，写入文本文件信息；它还可以访问数据库并对数据库进行读写操作或者运行其他程序等。但是，CGI 程序一般都是针对一个具体的应用目的而编写的，任务相对比较单一，例如用户访问网页次数计数器程序、留言板等。通常，无论采用 Perl，还是 ASP 或 PHP 编写的 CGI 程序都可称为 CGI 脚本，它实际上就是一个以适当的扩展名来命名的普通文本。

迄今为止，已涌现出多种可以实现 CGI 技术的方法，如 Perl，ASP，PHP 等，以下做简单介绍。

1. Perl

Perl (Practical Extraction and Report Language) 是由 Larry Wall 推出的，发展至今已有多种版本，较新的版本为 Perl 5.x。

Perl 最初只是 UNIX 系统管理员的工具之一，帮助 UNIX 用户完成一些经常性的任务，以后逐步发展成为一种全功能的程序设计语言，特别是在各种计算平台上，它被用做 Web 编程、数据库处理、XML 处理以及系统管理的编程语言，其应用也越来越广泛。如今 Perl 实际上已经被所有 UNIX 操作系统（包括多种版本的 Linux 操作系统）捆绑在一起作为标准部件发布。Perl 是按 GNU Public License 的形式免费分发的，原先运行于 UNIX 和类 UNIX 操作系统，经过一定的设置，Perl 也可以在 OS/2，Windows 等操作系统下运行。

Perl 可以执行大多数 C 语言所运行的内容，所以其功能十分强大。在 UNIX 环境下，可以执行 UNIX 系统命令，另外 Perl 还有如 C 语言一样的库结构，可以大大减少使用 Perl 编程

的工作量；同时 Perl 也支持复杂的数据结构与长变量名。Perl 是解释运行的，一般 Perl 程序的第 1 行需注明该程序是一个 Perl 程序而不是 Shell 程序，所以一般将“#!/usr/bin/perl”作为文件的第 1 行。

2. PHP

PHP (Hypertext Preprocessor) 是 Rasmus Lerdorf 于 1994 年创建的，最初只是一个简单的用 Perl 语言编写的程序，用来统计网站的访问者，后来又用 C 语言进行重新编写，具有了访问数据库的功能。用户可以从 PHP 官方站点 (<http://www.php.net>) 自由下载。PHP 遵守 GNU 公共许可 (GPL)，在这一许可下诞生了许多流行的软件，诸如 Linux 和 Emacs，用户可以不受限制地获得源码，甚至可以从中加进自己所需要的特色。PHP 程序可以在大多数 UNIX，GUN/Linux 平台上运行。

PHP 提供了类和对象。基于 Web 的编程工作非常需要具有面向对象编程能力的程序设计语言。PHP 支持构造器、提取类等，从而方便于开发人员进行面向对象编程。PHP 4 可以说是一个纯粹的 Script 翻译器，这也是 PHP 3 升级为 PHP 4 的重要改进之一。PHP 4 重写了语法分析器，从而提高了整体性能。PHP 需要支持某种扩充功能时一定要根据相应系统的 Lib 库进行再编译，将 Script 翻译成为需要执行的函数，再去执行它们。PHP 开发语言具有简单易学、跨平台、有良好数据库交互能力等优点，其在 UNIX/Linux 环境中的应用可以与 ASP 相媲美。

3. JSP

JSP (Java Server Pages)是由 Sun Microsystems 公司倡导、许多公司参与一起建立的一种动态网页技术标准，JSP 的官方网址为 <http://www.javasoft.com/products/jsp>。在传统的网页 HTML 文件 (*.htm, *.html) 中加 Java 程序片段 (Scriptlet) 和 JSP 标记 (Tag)，就构成了 JSP 网页 (*.jsp)。Web 服务器在遇到访问 JSP 网页的请求时，首先执行其中的程序片段，然后将执行结果以 HTML 格式返回给客户。程序片段可以操作数据库，重新定向网页以及发送 E-mail 等，这些都是动态网页所需要的功能。动态网页的所有程序操作都在服务器端执行，在网络上传送给客户端的仅仅是得到的结果，对客户浏览器的性能要求很低。

JSP 是一种特别的 Java 语言，加入了一个特殊的引擎，这个引擎将 HTTP Servlet 类的一些对象自动进行初始化以使用户使用，从而使用户不用再去操心前面的工作。同时这个引擎又引入了一系列的特别语法，使用这些语法可以进行一些特定的操作，例如引用文件、URL 转向、JavaBean 引用等，这些都是由引擎自动帮助完成初始化的。

4. ASP

本书所阐述的动态网页制作技术就是基于 ASP(Active Server Pages)技术的，它由 Microsoft 公司推出，其中的 Active 是 Microsoft 软件的重要基础，它采用封装对象、程序调用对象的技术，简化编程，加强程序间的合作。ASP 本身封装了一些基本组件和常用组件，只要在服务器上安装这些组件，通过访问组件，就可以快速、方便地建立各种 Web 应用。

Server 表明 ASP 运行在服务器端。这样就不必担心浏览器是否支持 ASP 所使用的编程语言。ASP 的编程语言可以是 VBScript 和 JavaScript。而在服务器端，则无需考虑用户使用的是何种浏览器。

Pages 表明 ASP 返回标准的 HTML 页面，它可以正常地在常用的浏览器中显示。浏览者查看页面源文件时，看到的是 ASP 生成的 HTML 代码，而不是 ASP 程序代码。

2000 年 6 月，Microsoft 公司推出了“Microsoft.NET 下一代互联网软件和服务战略”，在

全球的 IT 行业中引起了广泛的关注。同年 11 月, Microsoft 公司在 Comdex 计算机展会上发布了 Visual Studio.NET 软件, 并展示了其 .NET 发展战略的框架体系和开发工具的相关特性, 全面加速了以 .NET 技术进军市场的步伐。在激烈的市场竞争中, Microsoft 公司通过推出这样一个崭新的战略和概念, 在软件行业中树立信心、巩固地位。

1.2 .NET 概述

1.2.1 .NET 框架

计算机网络技术与程序语言到底向什么方向发展? 为什么 Microsoft 公司如此不遗余力地推销 .NET, 并为什么将其标志性的开发工具从 Windows 转向 .NET 框架? 要回答这些问题, 首先应该了解 .NET 框架的基础。

.NET 框架是一种分布式的运算框架, 它以 XML 为基础, 以 Web 为核心, 并结合其他多种技术, 最大限度地利用 Internet 上丰富的资源来提高工作效率。与 SUN 公司的 J2EE 相似, Microsoft.NET 不仅仅是一种编程语言, 它更是一种标准平台。图 1-1 所示为 Microsoft.NET 的结构与功能。

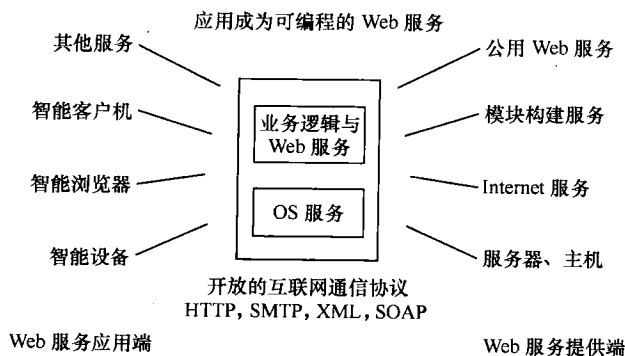


图 1-1 Microsoft .NET 结构功能图

从图中可以看出, Microsoft.NET 体系结构是建立在 XML 和 Internet 标准协议的基础之上。当时, Microsoft 公司的首席执行官鲍尔默认为: “Microsoft .NET 代表了一个集合、一个环境、一个可以作为平台支持下一代 Internet 的可编程结构。” 确实, 这段话简单扼要地表述了 .NET 的特点。

首先 Microsoft.NET 是一个环境, 这是一个理想化的未来 Internet 环境, Microsoft 公司的构想是构建一个“不再关注单个网站、单个设备与 Internet 相连的互联网环境, 而是要让所有的计算机群、相关设备和服务商协同工作”的互联网环境。简而言之, 互联网提供的服务, 要能够完成更高层次的自动化处理。将来的 Internet, 应该以一个整体服务的形式展现在最终用户面前, 用户只需要知道自己想要什么, 而不需要一步步地在网上搜索、操作来达到自己的目的。这是一个多年以来人们梦寐以求的理想, 当然也是互联网技术的发展趋势。

Microsoft.NET 追求的就是上面介绍的这种理想的互联网环境。而要实现这样一种理想的互联网环境, 首先需要解决的问题是针对当前所运行的 Internet 缺陷, 来设计和创造下一代 Internet 的结构。这种结构不仅仅是物理网络层次上的拓扑结构, 更是面向软件和应用层次的

一种有别于浏览器只能静态浏览的可编程 Internet 软件结构。所以 .NET 将自己定位于可以支持下一代 Internet 的可编程结构平台。

Microsoft.NET 的最终目的就是让用户在任何地方、任何时间、利用任何设备都能访问他们所需要的信息、文件和程序。用户无需知道它们存储在什么地方，甚至连如何获得等具体细节都无需知道，用户只需发出所需要的请求，然后接收即可。所有后台的复杂性操作是完全屏蔽的。对于 IT 技术开发人员来说，也不必管理复杂的平台以及各种分布应用之间是如何协调工作的。图 1-2 所示为 Microsoft.NET 整体框架图。

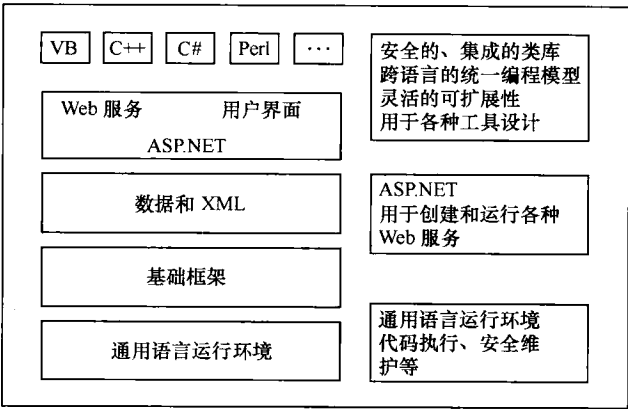


图 1-2 Microsoft.NET 整体框架图

从图中可以看出，Microsoft.NET 试图创建一个通用的多种编程语言的 API 集，可以实现跨语言的统一编程模型，并具有错误处理功能和调试功能，它可用于各种工具设计。

Visual Studio. NET 是 Microsoft 公司开发的用于企业规模的 Web 应用程序，又称为 .NET 框架。从 .NET 框架角度看，所有组件都可以是 Web 服务，而 Web 服务也仅仅是一种组件。实际上，.NET 框架提取出 Microsoft 公司组件对象模型（COM）的精华，生成了强大、高效的 Web 组件系统，极大地改善了应用程序的可靠性和可扩展性。

.NET 框架主要由 3 部分组成：通用语言运行库、一套层次结构的统一类库和 C# 等高级程序设计语言。

1. 通用语言运行库

通用语言运行库除了字面含义外，在开发阶段和运行过程中还扮演着另一个角色，即，在组件运行时，运行库负责管理内存分配、启动或中止线程和进程、强化安全系数，同时还调整任何该组件涉及的其他组件的附件配置。在开发阶段，运行库的作用更大，因为很多方面可以自动实现（例如内存管理等）。运行库可以使开发过程变得非常简单。

实际上每一种编程语言都含有一个运行库。例如 Visual Basic 的运行库为 VBRUN，Visual C++ 也有一个名为 MSVCRT 的运行库，此外，像 Visual FoxPro、JavaScript、SmallTalk、Perl、Python 和 Java 等都是如此。.NET 框架的特点之一就是它提供了一个跨所有编程语言的统一环境。

2. 统一类库

.NET 框架为开发人员提供了一套可以使用的统一的面向对象、异步、层次结构的可扩展类库。现在，C++ 的使用者使用 Microsoft Foundation Classes，Java 程序员使用 Windows Foundation Classes，Visual Basic 的用户使用 Visual Basic APIs。Microsoft 公司用 .NET 框架统

一了这些不同的框架。这样，开发人员不必学多个框架就可以完成自己的工作。而且，通过创建一套跨编程语言的通用 API，.NET 框架可以实现跨语言统一编程模型，并具有纠错处理以及程序调试功能。实际上，从 JavaScript 到 C++ 的所有编程语言，对于 .NET 框架都是相互等同的，开发人员可以自由地选择他们想使用的任何语言，如 Visual Basic，C++，C#，JavaScript，Perl 等，还可以进行各语言之间的深层次交互。同时 Microsoft 公司还支持第三方生产针对 Microsoft.NET 的编译器和开发工具，也就是说几乎所有市场上的编程语言都有可能应用于 Microsoft.NET 开发框架。这种开放性和交互性正是开发者所追求的。

3. C#

Microsoft 公司在 Microsoft.NET 中推出了全新的 C# 语言，这种全新的面向对象的程序设计语言使得开发者可以快速构建从底层系统级到高层商业组件级的不同应用。C# 在保证功能性和灵活性的同时，还给 C 和 C++ 程序员带来了类似于 Visual Basic 的快速开发环境，并且 C# 还针对 Microsoft.NET 做了特别设计，例如 C# 允许 XML 数据直接映射为它的数据类型等，这些特性结合起来使得 C# 成为优秀的下一代网络编程语言，本书所阐述的所有实例都是基于 C# 的。

总之，Microsoft.NET 是一种全新的平台，它将对已有的所有技术产生影响。正如盖茨所说：“微软所有的产品都会被涉及，整个战略是围绕这个平台展开的。”

1.2.2 ASP.NET

ASP.NET 是 Microsoft.NET 中的网络编程模块，可使 Web 编程非常方便、高效。

1. ASP.NET 与 ASP

起初，也有人将 ASP.NET 称为 ASP+，然而它并不是单纯的 ASP 的升级版，也不仅仅是功能上的增强，ASP.NET 是实质性的飞跃。ASP 使用 VBScript 或 JavaScript 脚本语言混合 HTML 来编程，但这些脚本语言属于弱类型、面向结构的编程语言，而非面向对象的编程语言，这就不可避免地会产生许多缺陷。ASP.NET 在理论上可以使用任何编程语言，包括 C++，VB，JavaScript 等，但合适的编程语言还是最新推出的 C#，有人称之为 C++++（读 C Sharp）。尽管 C# 内核比较类似 VC++，但实际上还是与 Java 更相似一些。首先，C# 是面向对象的编程语言，而不是一种脚本，所以它具有面向对象编程语言的一切特性，如封装性、继承性、多态性等，这就彻底解决了前面所说的 ASP 的弱点。封装性使得代码逻辑清晰，易于管理，并且应用到 ASP.NET 上就可以使业务逻辑和 HTML 页面分离，这样无论页面原型如何改变，业务逻辑代码都不必做任何改动；继承性和多态性使得代码的可重用性大大提高，并且 C# 和 C++，Java 一样，提供了完善的调试与纠错体系。

ASP.NET 与现存的 ASP 保持语法上的兼容，只是将现有的 ASP 文件的扩展名由 .asp 改为 .aspx，然后配置在支持 ASP.NET 运行的 IIS 服务器的 Web 目录下。ASP.NET 可以自动编译运行，并可获得优良性能。这是因为 ASP.NET 是编译（Compile）执行的，而 ASP 则是解释（Interpret）执行的，前者比后者有更高的效率。除此之外，ASP.NET 还可以利用 .NET 框架的诸多优越性能，如类型安全以及对 XML，SOAP，WSDL 等 Internet 标准的强大支持。目前，ASP.NET 可以使用 C#，Visual Basic.NET 和 JavaScript 3 种语言来开发。

2. 网络表单与控件

ASP.NET 网络表单的设计目的就是使开发者能够非常容易地创建网络表单，它把 Visual

Basic 中的快速开发模型引入到网络开发中来,从而大大简化了网络应用的开发。具体地说,ASP.NET 可以支持多种语言,它不仅仅支持脚本语言,通用语言运行库支持的所有语言在 ASP.NET 中都可以使用。

同时,ASP.NET 为开发人员提供了一种适用于 Web 开发、功能强大的控件工具,使开发人员能够在短时间内开发出功能强大的 Web 应用程序,编程时只需将所选的控件从工具箱拖放到页面上,通过属性(Properties)的设置便可以方便快速地完成诸多功能。

此外,ASP.NET 还可以通过创建自定义服务器控件来封装大量可重复使用的用户界面特定代码,进一步提高编程效率。

3. 网络服务

Web Service 是下一代网络编程的核心,它实际上就是一个可命名的网络资源,可用在 Internet 范围内方便地表现和使用对象,就像使用现在的 COM 对象一样。与使用 COM 对象不同的是,使用和表现网络服务是通过简单对象访问协议(SOAP)甚至 HTTP 来实现的。在 ASP.NET 中,建立和使用网络服务都非常方便。

Web Service 是为客户端提供功能的一个应用程序,它用托管的模式提供服务或者对数据的访问。这些服务可以来自全球各地,只要是基于 .NET 平台开发的 Web Service,彼此之间便可以共享对方提供的服务。所以 Web Service 并不仅仅提供服务,其本身也是共享服务的用户。加入的网络服务越多,Web Service 的价值就越能得到充分的体现。

总之,ASP.NET 可完全利用 Microsoft.NET 框架的强大、安全、高效的平台特性,运行在服务器端编译后的通用语言运行库代码,即时编译,本地优化,缓存服务,零安装配置,基于运行时代码受管与验证的安全机制等都为 ASP.NET 带来卓越的性能,对 XML, SOAP, WSDL 等 Internet 标准的支持为 ASP.NET 在异构网络的应用提供了强大的扩展性。

ASP.NET 主要包括 Web Form 和 Web Service 两种编程模型:前者为用户提供建立功能强大、外观丰富的基于表单(Form)的可编程 Web 页面;后者通过对 HTTP, XML, SOAP, WSDL 等 Internet 标准的支持,提供在异构网络环境下获取远程服务,连接远程设备,交互远程应用的编程界面。

本书的重点是使读者掌握 ASP.NET 的实际操作,有关 .NET 原理等理论知识在这里仅进行大致的介绍,读者要进一步了解有关知识可以参考微软的技术文档。

1.3 ASP.NET 的运行环境与安装

用户上网时,在浏览器的地址栏中输入网址,信息传到目标网站,网站就能返回用户所需要的信息。用户也可以把网页存入自己的硬盘,在浏览网页时选择“文件”/“另存为”命令即可。存入硬盘的网页在脱机状态下也可以浏览,用户还可以分析其中的代码,作为自己编制网页的借鉴。但是动态交互式网页的源代码用户却看不到,虽然也可以使用上述方法把网页保存下来,但看到的只是转化的 HTML 代码,源代码保存在网站的服务器中。学习用 ASP.NET 进行交互动态式网页的制作,首先要在单机上创建一个 Web 服务器环境,也就是要安装 IIS。

IIS 是 Microsoft 公司主推的 Web 服务器,较新的版本是 Windows Server 2003 操作系统里面包含的 IIS 6.0。IIS 支持 HTTP、文件传输协议(File Transfer Protocol, FTP)以及 SMTP,

通过使用 CGI 和 ISAPI, IIS 可以得到进一步的扩展。

IIS 支持与语言无关的脚本编写和组件, 通过 IIS, 开发人员就可以方便地开发动态交互式网页。IIS 完全支持 VBScript, JavaScript, Java, ASP.NET 等, 也支持 CGI 和 WinCGI, 以及 ISAPI 扩展和过滤器。

IIS 的设计目的是建立一套集成的服务器服务, 用以支持 HTTP, FTP 和 SMTP, 它能够快速提供集成了现有产品、同时可扩展的 Internet 服务器。

IIS 的安装、管理和配置都相当简单, 这是因为 IIS 与 Windows 网络操作系统紧密地集成在一起, 另外, IIS 还使用与 Windows NT Server 操作系统相同的安全性账号管理器 (Security Accounts Manager, SAM)。对于管理员来说, IIS 使用诸如 Performance Monitor 和简单网络管理协议 (Simple Network Management Protocol, SNMP) 之类的 WindowsNT 操作系统已有的管理工具。

IIS 支持 ISAPI, 使用 ISAPI 可以扩展服务器功能, 而使用 ISAPI 过滤器可以预先处理和事后处理存储在 IIS 上的数据。用于 32 位 Windows 应用程序的 Internet 扩展可以把 FTP、SMTP 和 HTTP 置于容易使用且任务集中的界面中, 这些界面将 Internet 应用程序的使用大大简化, IIS 也支持多用于 Internet 邮件扩展 (Multipurpose Internet Mail Extensions, MIME) 协议, 它可以为 Internet 应用程序的访问提供一个简单的注册项。

1.3.1 安装 IIS

目前, PC 主要安装的 Windows 操作系统有 Windows 2000 和 Windows XP 等, 服务器操作系统为 Windows 2000 Server 或 Windows Server 2003 等, 无论是哪种系统都应该安装相应的补丁, 否则运行会报错。下面以在 Windows XP 操作系统中安装与配置 IIS 为例, 介绍其具体操作步骤。

- ① 将 Windows XP 安装光盘放入光驱。
- ② 选择“开始”/“设置”/“控制面板”命令, 打开“控制面板”窗口。
- ③ 双击“添加/删除程序”图标, 并选取“添加/删除 Windows 组件”, 出现图 1-3 所示的“Windows 组件向导”对话框后, 选择 Internet 信息服务 (IIS) 复选框。

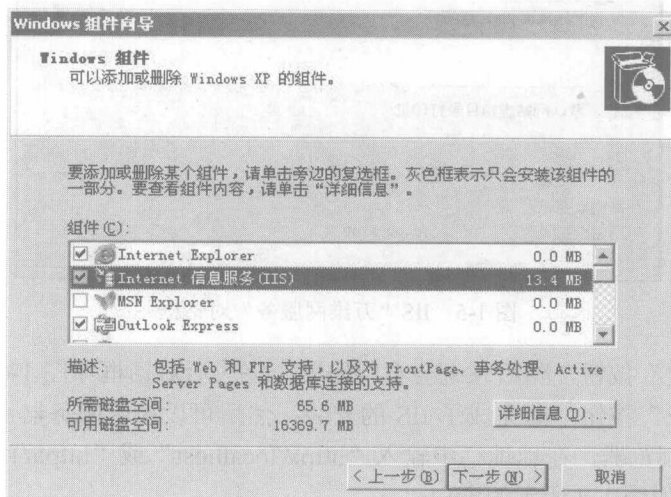


图 1-3 “Windows 组件向导”对话框