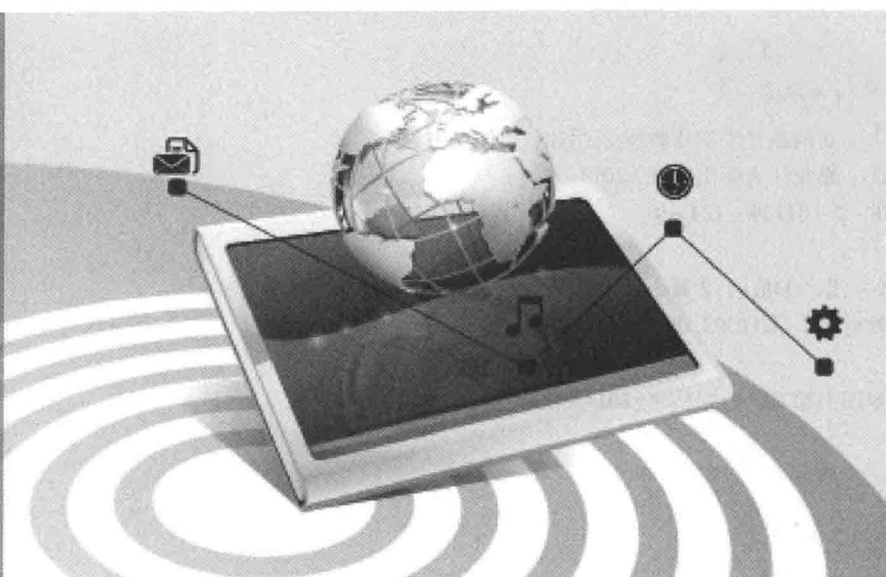




ASP.NET 4.0

网站开发实用教程

主编◇岳 冰 聂佳志



ASP.NET 4.0

网站开发实用教程

主编◇岳 冰 聂佳志

图书在版编目(CIP)数据

ASP.NET 4.0 网站开发实用教程 / 岳冰, 聂佳志主编
-- 哈尔滨: 黑龙江大学出版社, 2014.5
ISBN 978-7-81129-724-9

I. ①A… II. ①岳… ②聂… III. ①网页制作工具 - 程序设计 - 教材 IV. ①TP393.092

中国版本图书馆 CIP 数据核字(2014)第 087229 号

ASP.NET 4.0 网站开发实用教程
ASP.NET 4.0 WANGZHAN KAIFA SHIYONG JIAOCHENG
岳 冰 聂佳志 主编

责任编辑 张永生 于 丹
出版发行 黑龙江大学出版社
地 址 哈尔滨市南岗区学府路 74 号
印 刷 哈尔滨市石桥印务有限公司
开 本 787 × 1092 1/16
印 张 21.25
字 数 478 千
版 次 2014 年 5 月第 1 版
印 次 2014 年 5 月第 1 次印刷
书 号 ISBN 978-7-81129-724-9
定 价 40.00 元

本书如有印装错误请与本社联系更换。

版权所有 侵权必究

前 言

ASP.NET 是微软公司推出的基于 .NET Framework 的 Web 应用开发平台,是 Web 应用开发的主流技术之一,它带给人们的是全新的技术和由此产生的开发效率的提高、网站性能的提升。使用 ASP.NET 进行 Web 应用开发,程序结构更加清晰,开发流程更加简单,从而可以提高开发效率,缩短开发周期。ASP.NET 4.0 是在 ASP.NET 3.5 的基础上构建的,保留了其中很多令人喜爱的功能,并增加了一些其他领域的新功能和工具。

进行 ASP.NET 开发可以使用 Visual Basic.NET 或者 C#,这两种语言都是 .NET 环境下的程序设计语言,但并不是必须使用 .NET 集成开发环境才能进行 ASP.NET Web 程序设计。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,但大多数开发人员还是希望安装 Microsoft Visual Web Developer(VWD)2010。VWD 是专门为构建 ASP.NET Web 站点而开发的,其中包含了大量有助于快速创建复杂 ASP.NET Web 应用程序的工具。为了使广大学生和网站开发技术人员尽快掌握 ASP.NET 4.0 以及 VWD 2010 的使用技巧与新增功能,作者在多年教学经验与科研成果的基础上编写了本书。本书全面翔实地介绍了 ASP.NET 动态网站开发的基本流程和操作方法,可以使读者快速、全面地掌握使用 VWD 2010 快速开发 ASP.NET Web 应用程序的方法,并达到融会贯通、灵活应用之目的。

概括起来,本书具有以下主要特点。

(1) 结构清晰、内容翔实。在每一章的开始都概要说明了本章将介绍的内容,使读者做到心中有数;介绍每一个内置对象或服务器控件时,首先介绍该对象或控件的常用属性、方法和事件,然后介绍具体的应用场景和注意事项,且在介绍过程中都配有插图和实例说明。

(2) 按照动态网站的开发流程,从网站基础知识讲起,循序渐进地介绍了 ASP.NET 动态网站开发涉及的各种知识和技巧,并在各章配有精心选择的应用实例。这些实例既有较强的代表性和实用性,又能够综合应用对应章节介绍的知识,使读者能够全方位地理解所学内容,融会贯通章节之间的联系,达到举一反三之目的。

(3) 每一章最后都配有上机实训。这些实训有助于读者更好地掌握本章所学知识,提高读者的实际技能。

全书共分 11 章。第 1 章介绍了网站建设的基础知识、ASP.NET 的发展过程、VWD 2010 的开发环境以及 ASP.NET 的工作原理;第 2 章介绍了 C#语言基础知识;第 3 章介绍了 ASP.NET 的页面框架和页面类、ASP.NET 的内置对象以及 ASP.NET 的配置管理;

第4章介绍了ASP.NET 服务器控件的基本用法以及不同类别控件的功能;第5章介绍了CSS 样式、主题和母版页,这些技术对于页面布局、创建具有一致外观的网站非常有用,也有利于使站点看起来更专业和有吸引力;第6~8章介绍了数据库的基本知识、SQL 语言、使用ADO.NET 访问数据库的方法、数据绑定技术和数据控件的使用以及使用ADO.NET 读写XML 文件的方法,其中第6章介绍了SQL 语言及其语法,第7章介绍了Web 站点中的安全性和ASP.NET 提供的登录控件的用法,第8章介绍了ASP.NET 与XML 技术;第9章介绍了ASP.NET AJAX 的基本知识以及ASP.NET AJAX 服务器控件的使用方法;第10章介绍了Web 服务的基本概念以及如何创建和调用Web 服务,包括如何在AJAX 站点中使用Web 服务;第11章综合运用全书所学知识,开发一个简易的新闻发布系统。

本书第1、2、6、8、9章由聂佳志老师编写,共23.8万字;其余部分由岳冰老师编写,共24万字。全书由岳冰老师主编并统稿。本书参考了一些有关文献,在此向这些文献的作者深表感谢。由于编者水平有限,本书不足之处在所难免,欢迎广大读者批评指正。

编 者

2014年1月

目 录

第 1 章 ASP.NET 概述	1	第 2 章 C#语言基础	55
1.1 ASP.NET 简介	2	2.1 C#程序的基本结构与元素...	56
1.1.1 ASP.NET 的历史和特性...	3	2.1.1 C#程序的基本结构	56
1.1.2 .NET Framework	4	2.1.2 C#程序的基本元素	57
1.1.3 ASP.NET 的优势	8	2.2 C#面向对象技术	59
1.2 ASP.NET 的开发环境	10	2.2.1 面向对象的概念	59
1.2.1 安装和配置 IIS	10	2.2.2 封装	64
1.2.2 安装 Microsoft Visual Studio		2.2.3 继承	66
2010	22	2.2.4 多态	70
1.2.3 安装 Microsoft SQL Server		2.3 C#高级技术	74
2008	28	2.3.1 静态变量和方法	74
1.3 ASP.NET 网页语法	36	2.3.2 密封类和方法	76
1.3.1 ASP.NET 网页的文件		2.3.3 访问控制	77
扩展名	36	2.3.4 抽象类与抽象方法	78
1.3.2 ASP.NET 页面指令	37	2.3.5 接口	80
1.3.3 ASP.NET 注释	42	2.3.6 集合	85
1.3.4 ASP.NET 应用程序的		2.4 C# 4.0 的新特性	88
子目录	43	2.4.1 大整数类型	88
1.4 ASP.NET 的配置文件	45	2.4.2 动态数据类型	88
1.4.1 Web.config 文件	45	2.4.3 可选参数	89
1.4.2 Global.asax 文件	46	2.4.4 命名参数	89
1.4.3 ASP.NET 配置文件的		2.5 上机实训	89
层次结构和继承	47	第 3 章 ASP.NET 常用内置对象	91
1.4.4 ASP.NET 配置文件的格式		3.1 Request 对象	91
.....	47	3.1.1 Request 对象的常用	
1.5 制作一个 ASP.NET 网站 ...	52	属性和方法	91
1.6 上机实训	54	3.1.2 网页之间传递数据	92

3.1.3 获取客户端浏览器信息 ...	3.6.2 Application 对象的方法.....
..... 95	 124
3.2 Response 对象 97	3.6.3 Application 对象的使用.....
3.2.1 Response 对象的常用	 125
属性和方法 97	3.7 上机实训 128
3.2.2 Write 方法的使用..... 98	第4章 Web 服务器控件 130
3.2.3 Redirect 方法的使用 ... 99	4.1 Web 服务器控件 130
3.2.4 End 方法的使用 100	4.1.1 概述 130
3.3 Server 对象 101	4.1.2 Web 服务器控件的功能 ...
3.3.1 MapPath 方法的使用 ... 103	 131
3.3.2 HtmlEncode 方法的使用...	4.1.3 常用的 Web 服务器控件 ...
..... 104	 131
3.3.3 UrlEncode 方法的使用...105	4.2 验证控件 150
3.3.4 Execute 方法和 Transfer	4.2.1 必须字段验证控件 ... 150
方法的使用 106	4.2.2 比较验证控件 151
3.4 Cookie 对象..... 108	4.2.3 范围验证控件 152
3.4.1 概述 109	4.2.4 正则表达式验证控件... 154
3.4.2 Cookie 对象的属性..... 109	4.2.5 自定义验证控件 155
3.4.3 Cookie 对象的方法..... 109	4.3 导航控件 157
3.4.4 Cookie 对象的使用..... 110	4.3.1 Web. sitemap 文件 157
3.4.5 测试浏览器是否支持	4.3.2 SiteMapDataSource 控件 ...
Cookie 对象..... 111	 158
3.4.6 Cookie 对象的应用举例 ...	4.3.3 TreeView 控件..... 158
..... 114	4.3.4 Menu 控件 161
3.5 Session 对象 115	4.3.5 SiteMapPath 控件 161
3.5.1 概述 115	4.4 登录控件 162
3.5.2 Session 对象的属性 ... 115	4.4.1 登录控件概述 163
3.5.3 Session 对象的方法 ... 116	4.4.2 常用的登录控件 163
3.5.4 Session 对象的使用 ... 116	4.5 上机实训 164
3.5.5 Session 对象的应用举例 ...	第5章 主题与母版页 165
..... 117	5.1 主题 165
3.5.6 Session 的存储 121	5.1.1 什么是主题 165
3.6 Application 对象 123	5.1.2 主题的应用范围 166
3.6.1 Application 对象的属性.....	5.1.3 创建主题并应用网页 ... 167
..... 123	5.2 应用主题 170

5.2.1 设置应用主题的方法···	170	7.1 SqlDataSource 控件·····	224
5.2.2 以编程方式应用 ASP.NET 主题·····	171	7.1.1 SqlDataSource 控件的属性 ·····	224
5.3 母版页·····	171	7.1.2 SqlDataSource 控件的事件 ·····	225
5.3.1 母版页的工作原理···	172	7.1.3 配置数据连接·····	225
5.3.2 创建母版页·····	172	7.2 GridView 控件·····	232
5.3.3 设计母版页的布局···	173	7.2.1 常用属性、方法和事件·····	233
5.3.4 使用母版页创建内容页··· ·····	176	7.2.2 绑定数据·····	234
5.4 母版页的嵌套·····	181	7.2.3 显示数据·····	235
5.5 上机实训·····	181	7.2.4 排序设计·····	238
第6章 数据库与 ADO.NET 基础···	182	7.2.5 分页设计·····	241
6.1 数据库基础·····	182	7.3 DataList 控件·····	242
6.1.1 结构查询语言·····	182	7.3.1 DataList 模板·····	242
6.1.2 表和视图·····	183	7.3.2 DataList 控件的事件···	244
6.1.3 存储过程和触发器···	184	7.4 FormView 控件·····	244
6.2 使用 Microsoft SQL Server 2008 管理数据库·····	187	7.4.1 FormView 控件的常用 属性和事件·····	245
6.2.1 启动 Microsoft SQL Server 2008 服务管理器·····	187	7.4.2 利用模板显示数据···	245
6.2.2 Microsoft SQL Server 2008 使用的网络协议·····	188	7.4.3 编辑数据·····	247
6.2.3 Microsoft SQL Server 2008 相关操作·····	190	7.5 DetailsView 控件·····	249
6.3 ADO.NET 与数据库的访问·····	204	7.5.1 DetailsView 控件的常用 属性和事件·····	249
6.3.1 ADO.NET 基础·····	204	7.5.2 显示数据·····	250
6.3.2 连接 SQL Server·····	205	7.5.3 DetailsView 与 GridView 的 联合使用·····	251
6.4 ADO.NET 常用对象·····	207	7.6 上机实训·····	255
6.4.1 Connection 对象·····	208	第8章 ASP.NET 与 XML 技术·····	256
6.4.2 Command 对象·····	213	8.1 XML 概述·····	256
6.4.3 DataSet 数据集对象···	216	8.1.1 XML 定义·····	257
6.4.4 DataReader 对象·····	220	8.1.2 XML 基本语法规则···	259
6.5 上机实训·····	223	8.1.3 DTD 与 XML Schema···	268
第7章 数据访问服务器控件·····	224	8.2 XML 转换·····	273
		8.2.1 XML 转换概述·····	273

8.2.2 XSL 使用	274	9.4 上机实训	306
8.3 XML 的操作	276	第 10 章 ASP.NET Web 服务	308
8.3.1 使用 XML 控件	276	10.1 面向服务的软件架构概述	308
8.3.2 使用 XmlTextReader 和 XmlTextWriter	279	10.1.1 面向服务的软件架构 (SOA)	309
8.3.3 使用 XmlDocument(W3C DOM) 技术	285	10.1.2 SOA 与 Web 2.0	309
8.3.4 使用 DataSet 对象	287	10.2 Web Services 的概念	310
第 9 章 ASP.NET AJAX 技术	289	10.2.1 Web Services 的定义	310
9.1 ASP.NET AJAX 基础	289	10.2.2 Web Services 的核心技术	310
9.1.1 AJAX 概述	289	10.2.3 Web Services 原理	311
9.1.2 开发模式	290	10.3 ASP.NET 与 Web 服务	313
9.1.3 ASP.NET AJAX 优点	291	10.4 简单 Web Services 示例	314
9.1.4 ASP.NET AJAX 架构	291	第 11 章 新闻发布系统	316
9.2 ASP.NET AJAX 服务器控件	292	11.1 系统概述	316
9.2.1 ScriptManager 控件	292	11.2 需求分析	316
9.2.2 ScriptManagerProxy 控件	295	11.3 开发过程	316
9.2.3 UpdatePanel 控件	298	11.3.1 数据库设计	316
9.2.4 Timer 控件	299	11.3.2 配置 Web.config	317
9.3 引入 ASP.NET AJAX Control Toolkit 中的控件	301	11.3.3 公共类编写	317
9.3.1 TextBoxWatermark 控件	301	11.3.4 后台登录模块设计	321
9.3.2 PasswordStrength 控件	303	11.3.5 后台新闻管理模块设计	322
		11.3.6 前台主要功能模块设计	326

第1章 ASP.NET 概述

学习目的与要求

本章讲解了 ASP.NET 的基本概念以及 .NET Framework 的基本概念。这些概念在初学 ASP.NET 时会显得非常困难,但是在今后的开发中会逐渐清晰。同时,本章着重讲解了 Microsoft Visual Studio 2010 开发环境以及如何安装 Microsoft SQL Server 2008,以便于 ASP.NET 应用程序的数据存储。Microsoft Visual Studio 2010 和 Microsoft SQL Server 2008 的紧密集成能够提高 ASP.NET 应用程序的开发效率与运行效率。通过本章的学习将能够:

掌握 ASP.NET 开发工具的基本知识。

掌握 .NET Framework 的基本知识。

了解如何安装 Microsoft Visual Studio 2010。

了解如何安装 Microsoft SQL Server 2008。

掌握简单的 ASP.NET 应用程序的安装、编译和运行方法。

本章主要内容

.NET 的历史与展望:包括 .NET 应用程序的过去以及发展前景。

ASP.NET 开发工具:讲解了 ASP.NET 开发工具的基本知识。

.NET Framework:讲解了 .NET Framework 的基本知识。

.NET Framework 类库:讲解了 .NET Framework 类库的基本知识。

安装 Microsoft Visual Studio 2010:讲解了如何安装 Microsoft Visual Studio 2010。

安装 Microsoft SQL Server 2008:讲解了如何安装 Microsoft SQL Server 2008。

ASP.NET 应用程序基础:讲解了 ASP.NET 应用程序的安装、编译和运行方法。

互联网的出现已经完全改变了人们的生活方式。从静态页面到能够与用户交互的动态页面,互联网已经能够实现很强大的功能。Web 应用系统能够根据用户的要求动态处理数据,向用户提供个性化的服务。

现在的浏览器页面各自独立,互不相干。在互联网模式中,信息被存储在 Web 服务器内,用户的所有操作都依靠它,但不同的网页无法互相合作,传递有意义的信息,提供更深层的服务。

于是,微软公司梦想把整个互联网变成一个操作系统,使用户在互联网上开发应用

程序、使用互联网上的所有应用时,就好像在自己办公室里的计算机上一样,感觉不到互联网的存在。

.NET 是微软公司推出的下一代面向互联网的软件和服务战略,它的出现标志着新的软件设计理念和服务理念的产生。.NET 使得用户、企业和服务商三者联系更加紧密。

1.1 ASP.NET 简介

ASP.NET 应用程序是文件、页面、处理器、模块和可执行代码的组合,并且它们能够从服务器上的一个虚拟目录中被引用。ASP.NET 应用程序是一系列资源和配置的组合,这些资源和配置只在同一个应用程序内共享,而其他应用程序则不能享用这些资源和配置。每个 ASP.NET 应用程序都在一个单独的应用程序域中运行。应用程序域是内存中的独立区域,这样可以确保在同一台服务器上的应用程序不会相互干扰。

ASP.NET 页面作为代码在服务器上运行。因此,要得到处理,页面必须在用户单击按钮时或者当用户选中复选框与页面中的其他控件交互时提交到服务器。传递 Web 页面的过程如下:

(1) 用户请求页面。

(2) 页面将标记动态呈现到浏览器。

(3) 用户键入信息或从可用选项中进行选择,然后单击按钮。

(4) 页面发送到 Web 服务器。浏览器执行 HTTP POST 方法,该方法在 ASP.NET 中称为“回发”。在 Web 服务器上,该页面再次运行,并且可在页面上使用用户键入或选择的信息。

(5) 页面执行通过编程所要执行的操作。

(6) 页面将其自身呈现回浏览器。

软件开发是一项系统工程,本身需要项目策划、项目开发和后期维护三个主要的步骤。最主要的项目开发步骤包括代码的编写、编译和调试等过程。在这些过程中需要使用多种软件,例如编程语言编辑器、编译器/解释器、调试器等。

早期的软件开发者使用一些非常简陋的软件开发工具。随着软件开发技术的逐渐发展,越来越多的开发者趋向于使用一些集语言编辑、代码编译和调试于一体的综合性软件包,这一趋势促使 IDE (Integrated Development Environment, 集成开发环境) 软件诞生。

IDE 是一种综合性的软件开发辅助工具,其通常包括编程语言编辑器、编译器/解释器、调试器、安装包建立工具,有时还会包含版本控制系统和一些可以设计图形用户界面的工具。在开发基于 .NET Framework 的应用程序时,最常用的开发工具就是微软公司的 Visual Studio 系列。

Microsoft Visual Studio 2010 是微软公司开发的 Visual Studio 系列 IDE 的最新版本,也

是目前唯一支持 Microsoft .NET Framework 4.0 开发工具的 IDE。Microsoft Visual Studio 2010 是一套完整的开发工具,本身包含编程语言编辑器、编译器/解释器、调试器、安装包建立工具等多种工具,适合开发各种 Windows 程序。

Visual Studio 是一款强大的 .NET Framework 平台开发工具,也是开发 Windows 应用程序最流行的开发工具。其主要包含以下几种功能。

(1) 支持多种语言的代码编辑器

Visual Studio 集成开发环境作为之前微软公司提供的多种开发工具的集大成者,提供了功能强大的代码编辑器和文本编辑器,允许开发者编写 XHTML、HTML、CSS、Java Script、VBScript、C#、C++、J#、VB.NET、JScript.NET 等多种编程语言的代码,并可以通过组件的方式安装更多第三方的编程语言支持模块,支持编写更多的第三方编程语言。

在编写以上各种编程语言时,Visual Studio 提供了强大的代码提示功能和语法纠正功能,降低了开发者学习编程语言的成本,提高了程序开发的效率。

(2) 编译部署

Visual Studio 提供了强大的编程语言与中间语言编译功能,将其自身支持的多种编程语言和用户扩展的更多编程语言编译为统一的中间语言,并将其打包为程序集,发布和部署到各种服务器与终端上。

(3) 设计用户界面

Visual Studio 提供了功能强大的 Windows 窗体设计工具,允许开发者为 Windows 应用程序设计统一风格的窗口、对话框等人机交互界面,使用窗体控件实现软件与用户的交互。

(4) 团队协作

Visual Studio 提供了代码版本管理工具以及 SVN 平台等多种团队协作工具,帮助开发团队做好协同开发工作,管理开发进度,提高团队开发的效率。另外,用户可以使用最先进的 Team Foundation Server 服务器套件,更高效地进行版本控制、工作项跟踪、生产报表与工作簿规划等。

(5) 多平台程序发布

Visual Studio 具有强大的代码编译器和解析器,可以发布基于桌面、服务器、移动终端和云计算终端的多种应用程序。在非 Windows 平台应用方面,Visual Studio 也可以开发支持最新 Web 标准的前端网页,并针对多种网页浏览器进行调试。

1.1.1 ASP.NET 的历史和特性

自从 Microsoft .NET Framework 1.0 在 2002 年初首次发布以来,微软公司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。早期的 Web 程序开发是十分烦琐的事情,一个简单的动态页面需要编写大量的代码(一般用 C 语言)才能完成。

1996 年,微软公司推出了 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 这些简单的脚本语言编写代码,允许将代码直接嵌入 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 ActiveX Data Objects(ADO),它使得程序对数据库的操作变得十分简单。

1998 年,微软公司发布了 ASP 2.0 和 IIS 4.0。与前一版相比,ASP 2.0 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,微软公司在 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。

ASP.NET 是微软公司于 2002 年推出的新一代体系结构.NET 的一部分,用来在服务器端构建功能强大的 Web 应用。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。

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

2005 年 11 月,微软公司发布了 Microsoft Visual Studio 2005 和 ASP.NET 2.0,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。尽管 Microsoft Visual Studio 2005 和 ASP.NET 2.0 的功能已经很丰富了,但微软公司仍旧在努力。

2007 年 11 月,微软公司发布的 Microsoft Visual Studio 2008 和 ASP.NET 3.5 中添加了一系列很酷的新功能。主要的新功能包括 LINQ 以及 AJAX 框架整合。2008 年 8 月,微软公司又发布了用于 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 4 月微软公司发布了 Microsoft Visual Studio 2010(通常读作 twenty-ten)和 ASP.NET 4.0,它是在已成功发行的 Microsoft Visual Studio 2008 和 ASP.NET 3.5 的基础之上构建的,保留了其中很多令人喜爱的功能,并增加了一些其他领域的新功能和工具。

1.1.2 .NET Framework

.NET Framework 技术是微软公司提供的一种致力于快速应用开发的通用编程框架,为开发者提供一种类似虚拟机技术的平台,允许开发者以通用的代码实现多种硬件架构和操作系统的应用程序,降低软件开发的成本,提高工作效率。

.NET Framework 是微软公司面向下一代移动互联网、服务器应用和桌面应用的基础开发平台,是微软公司为开发者提供的基本开发工具,其中包含许多有助于互联网应用迅捷开发的新技术,如图 1-1 所示。

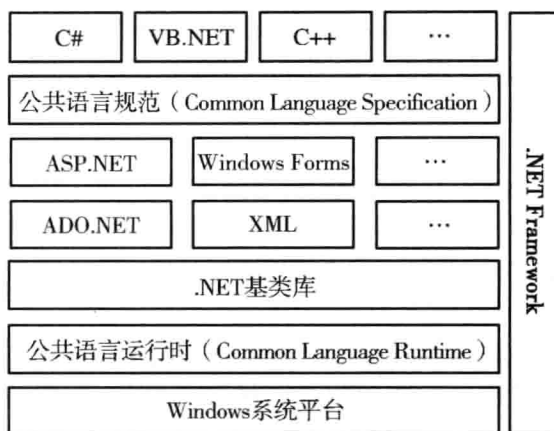


图 1-1 .NET Framework 开发平台

1. .NET Framework 的产生

在传统的软件开发工作中,开发者需要面对的是多种服务器和终端系统,包括用于个人计算机的 Windows 操作系统,用于服务器的 Windows 服务器系统和非 Windows 系统(如 FreeBSD、Linux 和 BSD),用于平面设计的 Mac OS X 操作系统以及各种移动终端系统(如 Windows Mobile、iOS、Android)等。

在开发基于以上这些系统的软件时,开发者需要针对不同的硬件和操作系统编写大量实现兼容性的代码,并使用不同的方式对代码进行编译。这一问题,都给软件设计和开发带来很多困难。

以 Windows 操作系统为例,目前主要使用的 Windows 操作系统内核包括 Win 9X、NT 4.0、NT 5.0/5.1、NT 6.0/6.1、Windows CE、Windows Mobile 6.X 和 Windows Phone OS 等。在这些操作系统中进行软件开发,可使用的技术包括以下几种:

- (1) 用于图形图像开发的 GDI、DirectX、OpenGL 等技术。
- (2) 用于数据库操作的 ADO、DAO、RDO、ODBC 等技术。
- (3) 用于 Web 应用开发的 ASP、JSP、PHP 等技术。
- (4) 用于移动终端的 XNA、HTML 5 等技术。

以上这些技术都有各自的标准和接口,相互不兼容。有些软件,开发者必须学习和使用相同的技术才能实现协作;而企业在实施项目开发时,也需要聘用指定技术的开发人员,才能实现最终的产品。

基于以上问题,微软公司在 21 世纪初开发出一种敏捷而快速的软件开发框架,其更加注重平台无关化和网络透明化,以 CLR(Common Language Runtime,公共语言运行时)为基础,支持多种编程语言,这就是 .NET Framework。

2. .NET Framework 的特点

.NET Framework 是一个灵活、稳定的运行服务器端程序、富互联网应用、移动终端程序和 Windows 桌面程序的软件解析工具(类似于虚拟机程序),也是软件开发的基础资源包,具有以下特点。

(1) 统一应用层接口

.NET Framework 将 Windows 操作系统底层的 API(Application Programming Interface, 应用程序接口)进行封装,为各种 Windows 操作系统提供统一的应用层接口,消除了不同 Windows 操作系统带来的不一致性。用户只需直接调用 API 进行开发,无须考虑平台。

(2) 面向对象的开发

.NET Framework 使用面向对象的设计思想,更加强调代码和组件的重用性,提供了大量的类库。每个类库都是一个独立的模块,供用户调用。同时,开发者也可以自行开发类库给其他开发者使用。

(3) 支持多种语言

.NET Framework 支持多种开发语言,允许用户使用符合 CLR 规范的多种编程语言开发程序,包括 C#、VB.NET、J#、C++ 等,然后再将代码转换为中间语言存储到可执行程序。在执行程序时,通过 .NET 组件对中间语言进行编译执行。

3. .NET Framework 的版本

.NET Framework 与 Windows 操作系统和 Visual Studio 集成开发环境保持着紧密的联系,发布的版本也与这两者紧密相关,如表 1-1 所示。

表 1-1 .NET Framework 的版本

发布日期	版本	对应 Windows 版本	对应 Visual Studio 版本
2002 年 2 月 13 日	1.0	Windows XP	Microsoft Visual Studio .NET
2003 年 4 月 24 日	1.1	Windows Server 2003	Microsoft Visual Studio .NET 2003
2005 年 11 月 7 日	2.0	—	Microsoft Visual Studio 2005
2006 年 11 月 6 日	3.0	Windows Vista/Windows Server 2008	—
2007 年 11 月 19 日	3.5	Windows 7/Windows Server 2008 R2	Microsoft Visual Studio 2008
2010 年 4 月 12 日	4.0	—	Microsoft Visual Studio 2010

目前最新版本的 Microsoft .NET Framework 4.0 具有以下特性。

(1) 图表控件

在开发 Microsoft .NET Framework 4.0 的应用程序时,开发者可以直接从 Microsoft Visual Studio 2010 中调用之前必须从 TechNet 下载的图表控件,创建更具可视化效果的数据图表。

(2) 托管可扩展性框架

托管可扩展性框架(MEF)是 Microsoft .NET Framework 4.0 中的一个新库,可以帮助

开发者创建可扩展和组合的应用程序,允许开发者指定应用程序中的扩展点,为其他应用程序服务。

(3) 并行计算

针对越来越多支持多线程技术的处理器,Microsoft .NET Framework 4.0 引入了一种新的编程模式,简化了应用程序和库开发者的编程。此模式可以帮助开发者在不使用线程或线程池时编写高效、具有可扩展性的并行计算程序。

(4) 垃圾收集

Microsoft .NET Framework 4.0 改进了之前版本的并行垃圾收集机制,支持从后台进行垃圾收集,从而提供了更好的系统性能。

4. .NET Framework 的应用

在微软公司发布 .NET Framework 之初,该技术仅仅是一种面向 Windows XP 和 Windows Server 2003 桌面应用的实现方式。随着富互联网应用和移动计算技术的发展,.NET Framework 不断得到增强,目前已经可以作为一种综合的开发平台应用到多种领域中。

(1) 桌面应用

桌面应用是 .NET Framework 最基本的应用。使用 .NET Framework,开发者可以开发基于 Windows 2000/NT5 以上版本桌面操作系统和服务器的桌面应用程序,并通过用户计算机的 .NET 组件实现本地文档和数据的操作。

使用 .NET Framework 开发桌面程序,开发者只需要将精力专注于程序算法和架构本身,不需再考虑这些桌面操作系统之间的差异,因此可以从繁杂的程序调试和兼容性测试工作中解放出来,极大地提高工作效率。

(2) 服务器应用

服务器应用也是 .NET Framework 的重要应用之一。使用 .NET Framework 开发出的服务器应用程序名为 ASP.NET 程序。相比于传统的 ASP 程序,.NET Framework 将网页分成前台页面和后台系统两个模块,将页面开发层和应用逻辑层完全隔离,提高网页开发的效率和代码的重用性,增强服务器应用程序的稳定性和安全性。

(3) 增强 Office 功能

作为微软公司提供的开发工具,.NET Framework 可以与微软公司开发的 Office 系列办公软件紧密地结合,开发应用于该软件的宏、加载项等,增强 Office 系列办公软件的功能,提高办公效率。

(4) 富互联网应用

为抗衡 Adobe 公司开发的 AIR(Adobe Integrated Runtime,Adobe 集成运行时)等富互联网应用技术,微软公司提出了 Silverlight 计划,通过 .NET Framework 编写基于 Web 的多媒体应用程序,通过丰富的可视化元素实现用户体验。

(5) 移动设备应用

.NET Framework 不仅可以应用到个人计算机、工作站等平台上,还可以为一些移动设备提供支持,例如使用 Windows CE 操作系统的 PDA、使用 Windows Mobile 和 Windows

Phone 7 等操作系统的智能手机等。开发者开发的 .NET 程序同样可以在这些设备上执行。

5. 其他平台中的 .NET Framework

除了微软公司开发的桌面、服务器和移动设备操作系统外,.NET Framework 还可以应用在其他几种操作系统中,通过以下几种技术实现跨平台应用。

(1) SSCLI 技术

SSCLI(Microsoft Shared Source Common Language Infrastructure,微软共享源公共语言平台)是由微软公司提供的代码共享实现,允许在 Windows XP、FreeBSD、Mac OS X 等操作系统上执行 .NET Framework。

(2) Mono

Mono 是一个开源的 .NET Framework 运行时与开发库的实现,由 Novell Ximian 和开源软件社区负责开发维护。它目前已经实现了对 ASP.NET 和 ADO.NET 的支持,同时支持部分 Windows Forms 库,允许在 Linux 等类 Unix 系统中开发和执行 .NET 程序。

1.1.3 ASP.NET 的优势

ASP.NET 技术是 .NET 技术的一个子集,提供了大量用于开发 Web 服务器端程序的类库,将这些类库封装在 System.Web.dll 文件中。

在编程实现上,这些类库存在于 System.Web 命名空间内,可以实现 Web 内容处理、扩展以及 HTTP 通道的应用程序与通信处理。另外,ASP.NET 还能够实现快速的异步数据交互,以及增强的数据库连接功能。

1. ASP.NET 的特点

ASP.NET 是 ASP 技术的后继者,可提供比传统 ASP 更加强大、高效而稳定的实现。ASP.NET 具有以下特点。

(1) 高效执行

ASP.NET 技术可将开发者编写的代码编译为中间语言代码,然后再通过专用的编译器转译为服务器计算机可以用的代码。相比于传统的 ASP、CGI、PHP 等即时解析的语言,其执行效率更高,更加安全和稳定。

(2) 语言无关性

ASP.NET 并不是一门编程语言,其本身只是一个类库和程序集。因此,编写 ASP.NET 代码与编程语言无关,开发者可以使用多种编程语言调用 ASP.NET 的类库,例如 Visual C#、Visual Basic.NET、JScript.NET、Perl、Python 等。

(3) 强大的适应性

ASP.NET 与 C#程序类似,都是通过 .NET 编译工具编译为中间语言,然后再交给解释器执行的,因此具有强大的适应性,只要是安装 .NET Framework 的操作系统都可以执行 ASP.NET 程序。开发者可以将通用语言的基本库、消息机制、数据接口的处理无缝地整合到 ASP.NET 的 Web 应用中。