

21世纪

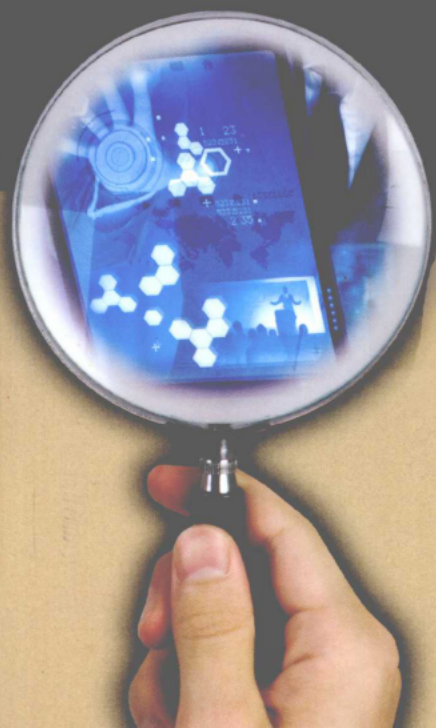
面向对象程序设计开发经典系列

Object-oriented Programming

面向对象 程序设计

——C#&ASP.NET实现

• 刘勇军 编著 •



 武汉理工大学出版社
WUTP Wuhan University of Technology Press

- 内容简介 -

本书以C#为语言基础，以ASP.NET为Web编程的核心组件，系统地介绍.NET体系结构、C#语言基础、面向对象编程基础，详细讲述C#中数组、集合、泛型、继承、接口、委托、事件以及程序的调试和异常处理；深入探讨页面对象、导航技术、服务器控件、内置对象、ADO.NET、数据控件、XML数据处理和Web程序的发布。依据知识体系内在的逻辑关系而设计的教程内容一定可以让读者轻松地掌握C#和ASP.NET的精髓，并最终能够开发出一个完整的Web应用程序。

本书包含教程、课件、实例程序、实践项目，全书试图以Web程序开发为导向，以C#为语言基础，帮助初学者以最快的速度系统地学习和掌握Web编程技术。书中大量采用对比、归纳、延伸的写作手法，以促使读者更好地理解和掌握各章知识点。书中内容简洁精炼、通俗易懂、逻辑严密；提供的实例程序既能帮助读者理解C#的语法和ASP.NET的控件的应用，又能使读者形成良好的编程思维和习惯。每一章中都安排了综合实验，并提供了Web开发的样例，以帮助读者灵活运用各种编程技巧及方法。

本书几乎涵盖C#语言和ASP.NET的所有功能，对于C#初学者和业内程序开发人员是值得大力推荐的参考书。

本书主要内容包括：

1. 回顾.NET体系结构，介绍C#的基础知识
2. 介绍数组、集合、泛型，并进行三者之间的比较
3. 介绍面向对象的基本理论，并重点介绍继承、接口、委托、事件等面向对象的特征及其应用
4. 介绍ASP.NET中的Web页面，包括Page对象、母版、用户控件
5. 介绍ASP.NET中的常用对象和控件，包括Web页面设计中用到的导航控件、服务器控件、内置对象
6. 重点介绍数据库编程技术，详细介绍Connection、Command、DataReader、DataAdapter、DataSet五大对象的功能，以及是用来显示和处理数据的Repeater、DataList、DetailsView、FormView和GridView五大数据控件
7. 介绍应用程序调试和发布，包括应用程序测试、调试、异常处理、应用程序的配置以及部署

本书附带一张光盘，此光盘中包含全书的教程、书中应用到的所有实例的源代码以及实践项目的源代码。

Object-oriented Programming

面向对象程序设计

——C#&ASP.NET实现

上架建议：程序开发



配套光盘含
本书所有示例程序源代码及
配套教学课件

ISBN 978-7-5629-3480-6



9 787562 934806 >

定价：52.00元

面向对象程序设计

——C# & ASP.NET 实现

刘勇军 编著

武汉理工大学出版社

· 武汉 ·

内 容 提 要

本书以 C# 为语言基础,以 ASP.NET 为 Web 编程的核心组件,系统地介绍 .NET 体系结构、C# 语言基础、面向对象编程基础,详细讲述 C# 中数组、集合、泛型、继承、接口、委托、事件以及程序的调试和异常处理;深入探讨页面对象、导航技术、服务器控件、内置对象、ADO.NET、数据控件、XML 数据处理和 Web 程序的发布。依据知识体系内在的逻辑关系而设计的教程内容一定可以让读者轻松地掌握 C# 和 ASP.NET 的精髓,并最终能够开发出一个完整的 Web 应用程序。

本书包含教程、课件、实例程序、学生实验部分,可作为高等院校信息管理类、电子商务类、计算机类等相关专业教材或实践参考书。

图书在版编目(CIP)数据

面向对象程序设计——C# & ASP.NET 实现/刘勇军编著. —武汉:武汉理工大学出版社,2011.8
ISBN 978-7-5629-3480-6

I. ① 面… II. ① 刘… III. ① C 语言-程序设计 ② 网页制作工具-程序设计 IV. ① TP312
② TP393.092

中国版本图书馆 CIP 数据核字(2011)第 148223 号

项目负责人:陈军东

责任编辑:陈军东

责任校对:陈 硕

装帧设计:吴 极

出版发行:武汉理工大学出版社

社 址:武汉市洪山区珞狮路 122 号

邮 编:430070

网 址:<http://www.techbook.com.cn>

经 销:各地新华书店

印 刷:湖北恒泰印务有限公司

开 本:787×1092 1/16

印 张:25.25

字 数:705 千字

版 次:2011 年 8 月第 1 版

印 次:2011 年 8 月第 1 次印刷

印 数:1—2000 册

定 价:52.00 元(含光盘)

凡购本书,如有缺页、倒页、脱页等印装质量问题,请向出版社发行部调换。

本社购书热线电话:027-87394412 87383695 87384729 87397097(传真)

· 版权所有 盗版必究 ·

前 言

本书是作者根据自己的教学和实验指导以及实际工作经验来编写的。全书旨在以 Web 程序开发为导向,以 C# 为语言基础,帮助初学者以最快的速度系统地学习和掌握 Web 编程技术。书中大量采用对比、归纳、延伸的写作手法,以促使读者更好地理解 and 掌握各章知识点。书中内容简洁精炼、通俗易懂、逻辑严密;提供的实例程序既能帮助读者理解 C# 的语法和 ASP. NET 的控件的应用,又能使读者形成良好的编程思维和习惯。每一章中都安排了综合实验,并提供了 Web 开发的样例,以帮助读者灵活运用各章知识点。

本书几乎涵盖 C# 语言和 ASP. NET 的所有功能,主要包括:

- ① 回顾 .NET 体系结构,介绍 C# 的基础知识;
- ② 介绍数组、集合、泛型,并进行三者之间的比较;
- ③ 介绍面向对象的基本理论,并重点介绍继承、接口、委托、事件等面向对象的特征及其应用;
- ④ 介绍 ASP. NET 中的 Web 页面,包括 Page 对象、母版、用户控件;
- ⑤ 介绍 ASP. NET 中的常用对象和控件,包括 Web 页面设计中用到的导航控件、服务器控件和内置对象;
- ⑥ 重点介绍数据库编程技术。详细介绍 Connection、Command、DataReader、DataAdapter 和 DataSet 五大对象的功能,以及用来显示和处理数据的 Repeater、DataList、DetailsView、FormView 和 GridView 五大数据控件;
- ⑦ 介绍应用程序调试和发布,包括应用程序测试、调试、异常处理、应用程序的配置以及部署。

本书附带一张光盘,光盘中包含全书的课件、书中应用到的所有实例的源代码以及实验的源代码。为运行这些代码,读者需配置 .NET 运行环境并安装 SQL Server,详细配置请参考第一章中的介绍。

本书主要是由刘勇军编写和审校,参与本书编写工作的还有:刘伦、熊峰、覃涛、黄伟、李伯奇等。由于时间仓促及水平有限,书中难免有些错误和不足,欢迎广大读者提出意见,邮箱为:liuyongjunnew@sina.com。

编者

2011-05-17

目 录

第 1 章 .NET 简介	(1)
1.1 .NET 基本思想	(1)
1.1.1 Visual Studio .NET 的历史和发展	(1)
1.1.2 对 .NET 的理解	(2)
1.1.3 .NET 项目的成功案例	(3)
1.2 .NET Framework	(6)
1.2.1 .NET Framework 的定义	(6)
1.2.2 .NET Framework 的主要内容	(6)
1.2.3 .NET Framework 的体系结构	(7)
1.2.4 .NET 应用程序执行步骤	(7)
1.3 Visual Studio .NET 集成开发环境	(8)
1.3.1 搭建 ASP.NET 开发环境	(8)
1.3.2 Web 窗体的 IDE	(18)
1.3.3 新建网站	(19)
1.3.4 新建解决方案和项目	(21)
1.4 应用程序类型	(25)
1.4.1 第一个控制台程序	(25)
1.4.2 第一个 Windows Forms 应用程序	(27)
1.4.3 第一个 Web 程序	(30)
第 2 章 C# 基础知识	(35)
2.1 变量与常量	(35)
2.1.1 变量	(35)
2.1.2 常量	(37)
2.2 C# 数据类型	(38)
2.2.1 C# 数据类型概述	(38)
2.2.2 值类型	(41)
2.2.3 引用数据类型	(43)
2.2.4 System.Object 类	(48)
2.2.5 变量类型的转换	(49)
2.3 装箱和拆箱	(54)
2.3.1 装箱	(55)
2.3.2 拆箱	(55)
2.4 C# 运算符和表达式	(57)
2.4.1 运算符和表达式的基本形式	(57)

2.4.2	运算符类型	(57)
2.4.3	Equals()与==的区别	(60)
2.5	基本语句	(61)
2.5.1	选择结构	(61)
2.5.2	循环结构	(64)
2.6	结构与枚举	(70)
2.6.1	结构	(70)
2.6.2	枚举	(71)
2.7	String、StringBuilder 和 DateTime	(74)
2.7.1	String 类型及其方法	(74)
2.7.2	StringBuilder 属性和方法	(76)
2.7.3	DateTime 类型	(78)
第3章	数组、集合对象和泛型	(79)
3.1	简单数组	(79)
3.1.1	数组概述	(79)
3.1.2	多维数组和交错数组	(80)
3.2	Array 数组类	(80)
3.2.1	System.Array 概述	(80)
3.2.2	System.Array 的属性和方法	(81)
3.3	集合对象	(82)
3.3.1	System.Collections 简介	(82)
3.3.2	ArrayList	(83)
3.3.3	Hashtable	(87)
3.4	泛型	(88)
3.4.1	泛型简介	(88)
3.4.2	List<T>	(88)
3.4.3	Dictionary<K,V>	(89)
3.4.4	数组、集合对象与泛型的比较	(90)
第4章	面向对象基础	(92)
4.1	继承面向对象的基本思想和概念	(92)
4.1.1	面向对象的由来和发展	(92)
4.1.2	面向对象的基本思想	(93)
4.1.3	面向对象的基本概念	(94)
4.2	类结构	(97)
4.2.1	类的基础知识	(97)
4.2.2	修饰符	(98)
4.3	构造函数	(99)
4.3.1	构造函数基础	(99)
4.3.2	参数化构造函数	(100)
4.4	析构函数	(101)
4.5	数据成员	(102)
4.5.1	成员常量	(102)

4.5.2 成员变量	(102)
4.5.3 只读成员变量	(103)
4.6 类的方法	(104)
4.6.1 方法的声明	(104)
4.6.2 方法的调用	(104)
4.6.3 方法的参数传递	(106)
4.6.4 方法的重载	(107)
4.6.5 方法的 XML 代码注释功能	(108)
4.7 关键字 static	(109)
4.7.1 静态成员变量	(109)
4.7.2 静态构造函数	(110)
4.7.3 静态方法	(111)
4.7.4 静态类	(112)
4.8 关键字 this	(113)
4.9 属性	(114)
4.10 索引器	(116)
4.11 命名空间	(119)
第 5 章 继承和接口	(122)
5.1 继承	(122)
5.1.1 继承的基础知识	(122)
5.1.2 关键字 base	(124)
5.1.3 关键字 virtual 和 override	(125)
5.1.4 关键字 new	(126)
5.1.5 关键字 abstract	(128)
5.1.6 关键字 sealed	(130)
5.2 接口	(131)
5.2.1 接口的基础知识	(131)
5.2.2 多重接口实现	(132)
5.2.3 接口作为参数传递和返回	(134)
第 6 章 委托和事件	(137)
6.1 委托	(137)
6.1.1 委托的引入	(137)
6.1.2 委托的含义	(139)
6.1.3 委托的语法	(139)
6.2 事件	(141)
6.2.1 事件的引入	(141)
6.2.2 事件的含义	(143)
6.2.3 事件的编码规范	(143)
6.2.4 事件的语法	(144)
6.2.5 事件的实验	(145)
第 7 章 测试、调试和异常处理	(149)
7.1 错误类型	(149)

7.1.1	应用程序的要求	(149)
7.1.2	错误的分类	(149)
7.2	调试过程	(149)
7.2.1	设置断点	(150)
7.2.2	调试窗口	(151)
7.3	异常处理	(153)
7.3.1	异常的含义	(153)
7.3.2	System.Exception 类	(153)
7.3.3	try...catch...finally 语句	(155)
7.3.4	throw 语句	(155)
第 8 章	Web 页面、母版和用户控件	(157)
8.1	Page 对象	(157)
8.1.1	Web 页面组成	(157)
8.1.2	页面指令	(158)
8.1.3	Page 对象的属性	(163)
8.1.4	Page 对象的事件	(163)
8.1.5	页面中弹出窗体	(164)
8.2	母版页	(165)
8.2.1	页面布局	(165)
8.2.2	母版页的定义	(166)
8.2.3	母版页的使用	(167)
8.2.4	母版页事件顺序	(168)
8.3	用户控件	(168)
8.3.1	用户控件生成	(168)
8.3.2	Web 窗体向用户控件转化	(169)
8.3.3	用户控件注册	(169)
第 9 章	导航技术	(170)
9.1	页面导航系统	(170)
9.1.1	超链接	(170)
9.1.2	页面重定向	(171)
9.2	站点导航系统	(172)
9.2.1	站点地图	(173)
9.2.2	Web.config 文件设置	(174)
9.2.3	SiteMapPath 控件	(174)
9.2.4	SiteMapDataSource 控件	(175)
9.3	TreeView 控件	(175)
9.3.1	TreeView 控件概述	(175)
9.3.2	TreeView 控件的属性	(176)
9.3.3	TreeView 控件的事件	(178)
9.4	Menu 控件	(179)
9.4.1	Menu 控件概述	(179)
9.4.2	Menu 控件的属性	(179)

9.4.3 使用 Menu 控件导航	(181)
9.4.4 以编程的方式添加菜单项	(182)
第 10 章 ASP.NET 常用对象	(184)
10.1 Response 对象	(184)
10.1.1 Response 对象的含义	(184)
10.1.2 Response 对象的属性	(184)
10.1.3 Response 对象的方法	(185)
10.1.4 Response 对象的应用	(185)
10.2 Request 对象	(188)
10.2.1 Request 对象的含义	(188)
10.2.2 Request 对象的属性	(188)
10.2.3 Request 对象的方法	(188)
10.2.4 页面提交的 get 和 post 方法	(189)
10.2.5 Request 对象的应用	(190)
10.3 Session 对象	(191)
10.3.1 Session 对象的作用	(191)
10.3.2 Session 对象的属性	(191)
10.3.3 Session 对象的方法	(192)
10.4 Application 对象	(192)
10.4.1 Global.asax 文件	(192)
10.4.2 Application 对象的作用	(193)
10.4.3 Application 对象的属性	(193)
10.4.4 Application 对象的方法	(194)
10.5 Server 对象	(196)
10.5.1 Server 对象的作用	(196)
10.5.2 Server 对象的属性	(196)
10.5.3 Server 对象的方法	(197)
10.6 Cookie 对象	(198)
10.6.1 Cookie 对象的作用	(198)
10.6.2 Cookie 对象的属性	(198)
10.6.3 Cookie 对象的应用	(198)
10.7 ViewState 对象	(201)
10.7.1 ViewState 对象的作用	(201)
10.7.2 ViewState 对象的使用	(201)
10.7.3 ViewState 对象的禁用	(203)
10.7.4 ViewState 对象的安全性	(203)
第 11 章 服务器控件	(205)
11.1 HTML 服务器控件	(206)
11.1.1 HTML 服务器控件简介	(206)
11.1.2 HtmlForm 控件	(206)
11.1.3 HtmlAnchor 控件	(209)
11.1.4 HtmlInputText 控件	(210)

11.1.5	HtmlInputFile 控件	(212)
11.1.6	HtmlInputHidden 控件	(214)
11.1.7	HtmlButton 控件	(216)
11.1.8	HtmlInputButton 控件	(217)
11.1.9	HtmlImage 控件	(218)
11.1.10	HtmlInputImage 控件	(220)
11.1.11	HtmlInputRadioButton 控件	(222)
11.1.12	HtmlInputCheckBox 控件	(223)
11.2	Web 服务器控件	(225)
11.2.1	Web 服务器控件简介	(225)
11.2.2	文本输入控件	(227)
11.2.3	Image 控件	(233)
11.2.4	控制权转移控件	(234)
11.2.5	选择控件	(239)
11.2.6	容器控件	(245)
11.2.7	其他 Web 服务器控件	(248)
11.3	验证控件	(254)
11.3.1	验证控件简介	(254)
11.3.2	RequiredFieldValidator 控件	(255)
11.3.3	CompareValidator 控件	(256)
11.3.4	RangeValidator 控件	(259)
11.3.5	RegularExpressionValidator 控件	(262)
11.3.6	CustomValidator 控件	(266)
11.3.7	ValidationSummary 控件	(268)
第 12 章	ADO.NET	(270)
12.1	ADO.NET 架构	(270)
12.1.1	ADO.NET 简介	(270)
12.1.2	ADO.NET 的结构	(270)
12.1.3	ADO.NET 对数据库的访问	(271)
12.1.4	数据源控件	(272)
12.2	Connection 对象	(275)
12.2.1	Connection 对象的属性和方法	(275)
12.2.2	建立数据库连接	(277)
12.3	Command 和 SqlTransaction 对象	(278)
12.3.1	SqlCommand 类	(278)
12.3.2	构造 SqlCommand 对象	(279)
12.3.3	ExecuteReader 和 ExecuteScalar 方法	(281)
12.3.4	执行带参数的文本命令	(282)
12.3.5	执行存储过程	(283)
12.3.6	方法的抽象	(284)
12.3.7	SqlTransaction 事务处理对象	(284)
12.4	DataReader 和 DataAdapter 对象	(286)

12.4.1	DataReader 对象	(286)
12.4.2	DataAdapter 对象	(287)
12.5	DataSet 对象	(291)
12.5.1	DataSet 简介	(291)
12.5.2	DataTable	(293)
12.5.3	DataRow 对象	(298)
12.6	SqlHelper 类	(300)
12.6.1	SqlHelper 类简介	(300)
12.6.2	SqlHelper 类的方法	(301)
12.6.3	SqlHelperParameterCache 类	(302)
第 13 章	数据控件	(305)
13.1	数据绑定	(306)
13.1.1	数据绑定简介	(306)
13.1.2	Eval 方法与 Bind 方法比较	(307)
13.1.3	数据控件模板	(307)
13.1.4	数据控件 Field 字段类型	(310)
13.2	GridView 控件	(311)
13.2.1	GridView 控件简介	(311)
13.2.2	基于 SqlDataSource 的 GridView 控件实现	(312)
13.2.3	GridView 控件的编程实现	(316)
13.3	DetailsView 控件	(320)
13.3.1	基于 SqlDataSource 的 DetailsView 控件实现	(320)
13.3.2	DetailsView 控件的编程实现	(321)
13.4	Repeater 控件	(325)
13.4.1	基于 SqlDataSource 的 Repeater 控件实现	(325)
13.4.2	Repeater 控件的编程实现	(326)
13.5	DataList 控件	(328)
13.5.1	DataList 控件简介	(328)
13.5.2	基于 SqlDataSource 的 DataList 控件实现	(329)
13.5.3	DataList 控件的编程实现	(330)
13.6	FormView 控件	(332)
13.6.1	FormView 控件简介	(332)
13.6.2	基于 SqlDataSource 的 FormView 控件实现	(333)
13.6.3	FormView 控件的编程实现	(334)
13.7	ASP.NET 数据控件综述	(337)
第 14 章	XML 文件处理	(338)
14.1	XML 简介	(338)
14.1.1	XML 的产生	(338)
14.1.2	XML 文档节点	(338)
14.1.3	XML 文件解决方案	(339)
14.1.4	XML 相关文件的创建	(339)
14.2	XML 文件操作	(342)

14.2.1 XML 命名空间和相关类	(342)
14.2.2 XML 文档读操作	(343)
14.2.3 XML 文档写操作	(344)
14.2.4 XmlDocument 类	(345)
第 15 章 应用程序配置和部署	(348)
15.1 配置文件	(348)
15.1.1 配置文件简介	(348)
15.1.2 配置文件的结构	(349)
15.1.3 ASP.NET 的异常处理	(350)
15.1.4 ASP.NET 身份验证(Authentication)与授权(Authorization)	(352)
15.1.5 其他配置	(353)
15.1.6 配置文件的程序访问	(355)
15.2 应用程序的安装和部署	(356)
15.2.1 XCOPY 命令部署	(356)
15.2.2 利用 VS.NET 的“发布”功能部署	(356)
15.2.3 利用 VS.NET 的“Web 安装项目”部署	(357)
第 16 章 计件工资管理系统开发实例	(360)
16.1 项目概述	(360)
16.1.1 项目背景简介	(360)
16.1.2 项目需求分析	(360)
16.1.3 业务流程分析	(361)
16.2 计件工资管理系统的设计	(362)
16.2.1 系统功能设计	(362)
16.2.2 E-R 图设计	(363)
16.2.3 表结构设计	(364)
16.2.4 编码设计	(366)
16.3 计件工资管理系统的实现	(366)
16.3.1 用户登录模块的实现	(366)
16.3.2 产品管理模块的实现	(368)
16.3.3 人力管理模块的实现	(370)
16.3.4 生产管理模块的实现	(371)
16.3.5 财务管理模块的实现	(373)
16.3.6 系统管理模块的实现	(377)
附录	(378)
SQL 简介:基于 SQL Server 2008	(378)
HTML 简介	(382)
CSS 简介	(385)

第 1 章

.NET 简介

1.1 .NET 基本思想

1.1.1 Visual Studio .NET 的历史和发展

Visual Studio .NET 是一个集成开发环境(Integrated Development Environment, IDE),是把代码的编写,程序的调试、编译、运行以及其他的相关操作都集成在一起的编程工具。目前 Visual Studio .NET 已成为最流行的 Windows 平台应用程序开发环境,版本已经升级到 Visual Studio 2010。图1.1展示了 Visual Studio .NET IDE 的发展历程。

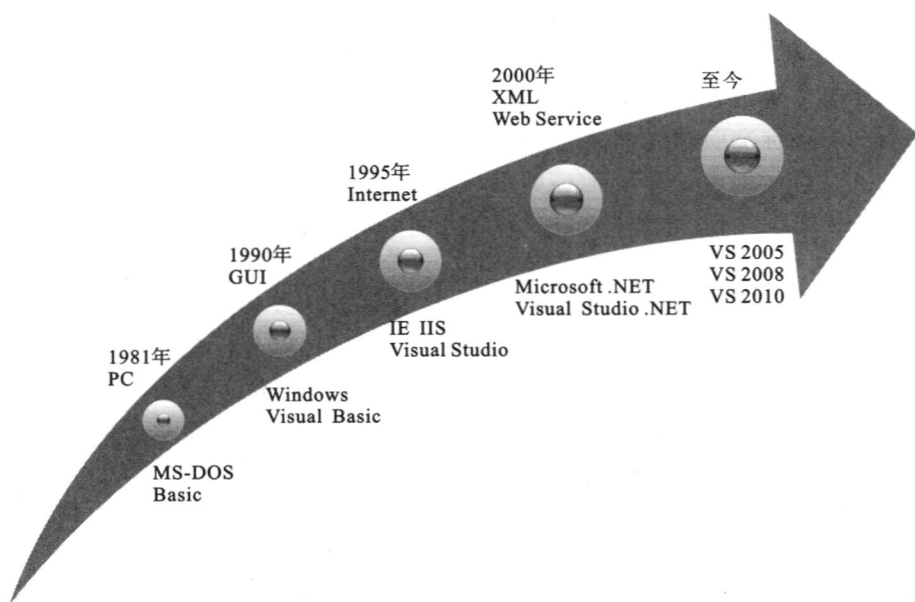


图 1.1 Visual Studio .NET 的发展历程

早在 20 世纪 80 年代,微软公司的比尔·盖茨亲手创造了 MS-DOS 系统,这是 Windows 操作系统的始祖。该系统是用 Basic 语言编写的,它与如今的 IDE 相距甚远。到了 1990 年,VB 的产生使得用语言制作一个漂亮友好的界面(GUI)成为一件简单的事情,从此人们也越来越关注应用程序界面等非功能方面的技术。到了 1995 年,Visual Studio IDE 日趋成熟,这套 IDE 不仅包含 VB 6.0 和 VC++ 6.0,也包括数据库开发的 Visual Foxpro,还包括网页开发的 Visual Inte-eDev 等。有了这个 IDE,人们不仅可以创建基于 Windows 的桌面应用程序,也可以创建基于 IE 浏览器的 Web 程序,可以说这套 IDE 是非常强大的。但问题是,在这个开发环境中似乎每做一件事都有一个工具与其对应,各个开发工具之间也彼此独立,比如要开发网页程序,就只能使用

Visual InteDev,用 VB 开发的程序,在 VC 中很难调用,而且针对不同的应用比如移动应用还需要另外一套与桌面应用相似的开发 IDE。

Visual Studio 这些不足之处在 Visual Studio .NET 中得以弥补。Visual Studio .NET IDE 以 .NET Framework 为基础,整合了 VB.NET、VC++ .NET、C# .NET、VJ# .NET 等开发环境,提供了极其强大的开发能力。这种整合不仅仅意味着你打开一个 Visual Studio .NET IDE 工具即可进行以上四种语言的开发,更重要的是不同语言的开发环境和开发技巧基本相同。

1.1.2 对.NET 的理解

Microsoft .NET 框架是生成、部署和运行 Web 服务及应用程序的平台。它提供了一个生产率高且基于标准的多语言环境,将现有系统与下一代应用程序和服务集成,同时解决了 Internet 上应用程序的部署和操作难题。对于 .NET,不同的人理解程度不一样:很多人认为 .NET 是一个编程开发工具,因为它有新的开发语言(C#),有人认为它是一个 Web 服务平台、一种软件环境(.NET Framework),也有人觉得这是微软的一个新战略,还有人觉得 .NET 是一个概念,是一种构想。

(1) 微软定义

Microsoft .NET 是微软以 Web Service 为核心,为信息、人、系统、各种设备提供无缝连接的一组软件产品(Smart Client、服务器、开发工具)、技术(Web Service)或服务。根据微软的定义,.NET 的精髓在于 Web Service。Web Service 允许应用程序通过 Internet 进行通信和共享数据,而不管所采用的是哪种操作系统、设备或编程语言。Microsoft .NET 平台提供了创建 Web Service 并将这些服务集成在一起的功能,这个平台包含的产品系列范围很广,它们都是基于 XML 和 Internet 行业标准而构建,提供从开发、管理、使用到体验 Web 服务的每一方面。现在,Microsoft 正从五个方面创建 .NET 平台,即工具、服务器、Web 服务、客户端和 .NET 体验。

在微软的产品发布会上,主持人曾经说过一句话:在未来,我们可以在任何时间、任何地点,使用任何设备获取信息。但是这还不是 .NET 的全部,仅仅能够获取信息是不够的,通过 .NET,用户还将获得由程序封装过的数据——也就是服务。微软想通过这一点,将全世界的互连网络组成一个庞大的服务中心,用户不需要亲自去获得信息、分析信息,只需要将用户的要求输入或者发送至用户所拥有的终端设备(台式机、手机、PDA、平板 PC 等),这些终端设备就会自动地去互联网上查找相关的服务,经过其智能处理与整合,以最有效率的方式完成任务。

其实这仅仅是 .NET 体验的一小部分而已。.NET 为开发人员提供了新的开发平台(.NET 框架)、新的开发语言(C#)、新的开发工具(VS .NET)以及新的开发方式:为普通用户提供了 Windows CE、Windows XP、Stinger、Xbox、Tablet PC、.NET My Services、MSN 等产品;为企业提供了 bCentral。在未来,最终大多数流行的 Microsoft 软件应用程序(包括 Office 和 VS .NET)将与 XML Web 服务实现交互,并把它们的主要功能作为 Web 服务公开,以便其他开发人员使用。通过 Web Service 将企业开发的模式从 Client/Server 或 Browser/Server 转换到 Web Server/Smart Client。

(2) 微软战略

Microsoft .NET 是微软的一个网络战略,更是微软意图全面占领互联网领域的最强有力的武器。在全球互联网络市场的抢夺战上,微软似乎慢了半拍,在浏览器方面差点就败在 Netscape 手上,还为此吃了官司;在流媒体上又被 Real 斩于马下。随着互联网以前所未有的速度席卷全球,很多人都希望借此机会重新洗牌,但是惯于制定游戏规则的微软又怎会轻易将主动权交到别人的手中?同时,由于开发源代码组织的不断扩大,微软紧抱源代码不放的做法招致越来越

多人的不满,很多人出于不同的考虑(安全、开放、免费)投靠了 Linux 阵营。对于微软来说,其产品必须要变,而这个变,就是 .NET 带来的由一个软件公司向一个服务公司的转变。事实上,微软将来可能会变成一个全球最大的网络服务商(ASP)。Windows 这个给微软带来令人炫目的财富和辉煌的十年视窗时代结束了,微软打算全面规划其未来,它将把所有的产品全部重新改写为与 .NET 构造相一致的形态,以应用服务商的方式提供,这之后微软不再主要依靠授权和销售软件光盘赚取利润,而是要通过互联网上运行的大量软件服务赚取利润,从软件供应商走向 ASP——这就是微软的新战略。

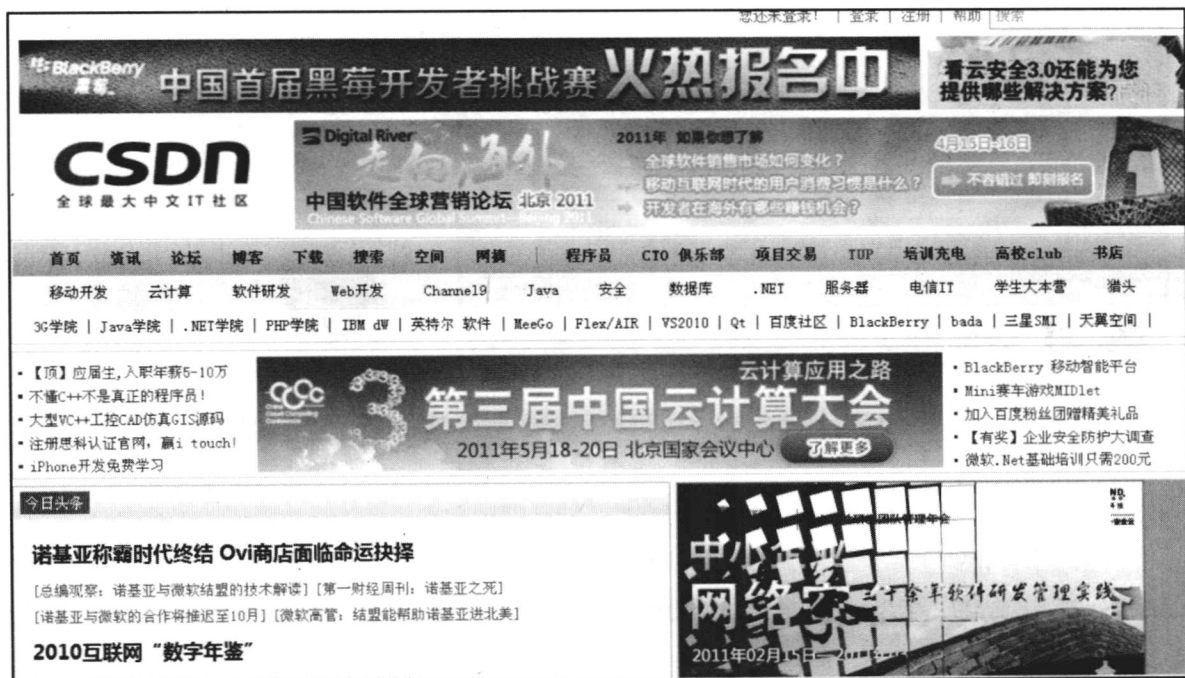
事实上,很早就有人提出,对计算机发展和普及作出巨大贡献的软件行业已经到了一个转折点:留在终端的软件会越来越少,目前通过软件包发行的方式即将消失,而改为网上出租的形式获得利润。用户只要在本地上发出请求,就可以在网上直接使用它,而这个软件的供应商会根据用户使用的次数来收费。一旦这种设想成为现实,集成诸多功能的浏览器就将取代现在的操作系统的地位,成为终端上唯一需要预安装的软件。所以,微软也在竭尽全力的压制 Netscape。

1.1.3 .NET 项目的成功案例

.NET 作为微软全力推出的一个崭新的平台,经过最近几年的发展,在各领域中已经有了很多成功的项目案例,比如金融业的纳斯达克、目前我国技术最先进的银行——招商银行、全球最大的中文 IT 社区——CSDN、著名的网络电视——PPTV、著名的汽车生产商——东风汽车公司等,其官方网站都使用了 .NET 平台,并使用 C# 语言来实现,具体如图 1.2~图 1.6 所示。



图 1.2 纳斯达克 <http://www.nasdaq.com/>

图 1.3 招商银行 <http://www.cmbchina.com/>图 1.4 CSDN 社区 <http://www.csdn.net/>