

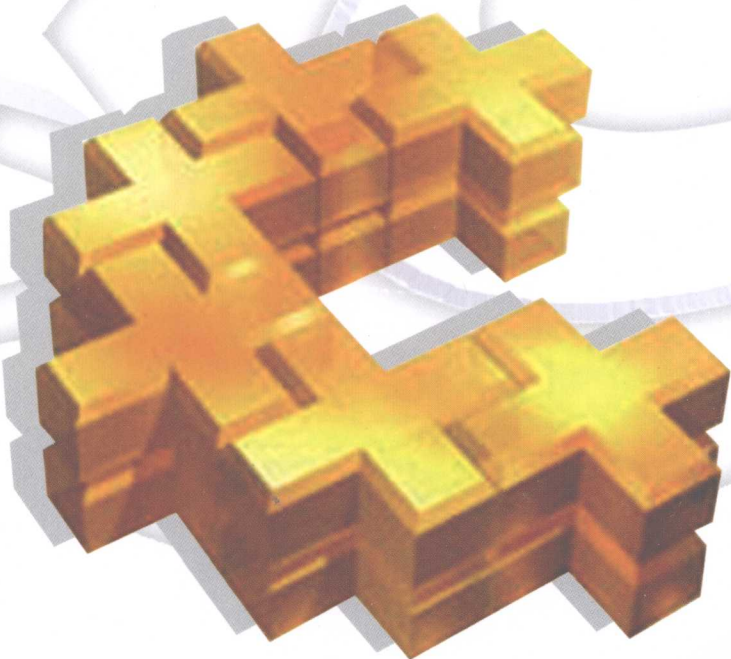
通用程序设计案例丛书

Visual C# .NET

网络编程案例解析

杨富国 主编

王 浩 蒋大伟 唐巧琪 陈 聪 编



附光盘

- 合理的知识体系和学习流程
- 操作步骤更加详尽
- 精美、简洁、清晰的版式设计
- 精挑细选的操作案例
- 合理的教学体例



清华大学出版社
www.tup.tsinghua.edu.cn



北京交通大学出版社
<http://press.bjtu.edu.cn>

通用程序设计案例丛书

Visual C# .NET

网络编程案例解析

杨富国 主编

王 浩 蒋大伟
唐巧琪 陈 聪 编

清华大学出版社
北京交通大学出版社

· 北京 ·

内 容 简 介

本书依照软件工程设计规范,通过具体丰富的网络编程案例,全面系统地介绍了运用 Visual C#开发各种应用系统的思路、方法、技巧和经验,案例范围涉及网络通信、数据库、多媒体应用、基于 Web 的程序和网络服务应用,以及移动智能设备的软件开发等。所有案例均提供了完整的源代码以供引用和借鉴。

本书内容翔实、案例丰富、实践性强,可以帮助读者进一步掌握软件工程方法,增加系统开发经验。本书适用于计算机及其相关专业的本科高年级学生和研究生,以及具有一定程序设计基础的程序设计人员。

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

Visual C#.NET 网络编程案例解析/杨富国主编. —北京:清华大学出版社;北京交通大学出版社,2009.2

(通用程序设计案例丛书)

ISBN 978-7-81123-501-2

I. V… II. 杨… III. C 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 003760 号

责任编辑:刘 洵

特邀编辑:马小强

出版发行:清华大学出版社

邮编:100084

电话:010-62776969

北京交通大学出版社

邮编:100044

电话:010-51686414

印刷者:北京东光印刷厂

经 销:全国新华书店

开 本:185×260 印张:19.5 字数:499 千字

版 次:2009 年 6 月第 1 版 2009 年 6 月第 1 次印刷

书 号:ISBN 978-7-81123-501-2/TP·462

印 数:1~4 000 册 定价:35.00 元(含光盘)

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

本书首先概略地介绍了 Visual Studio .NET 集成开发环境的一些最新工具和技术信息。随后从 Visual C#编程基础切入,通过大量实际案例分析介绍网络程序开发的方法。所设计的案例范围涉及网络通信、数据库、多媒体应用、基于 Web 的程序和网络服务应用,以及移动智能设备的软件开发等诸多方面,难度和规模与计算机专业本科毕业设计的软件系统相当,具有一定的系统性和较宽的覆盖面。

在案例的分析与实现中,运用面向对象原则和软件工程的规范流程,力求系统地展现从系统分析到程序实现的开发全过程,尽量避开编程过程中的旁枝末节问题,使入门级的程序开发人员迅速熟悉和掌握程序开发思路和方法,找出解决实际问题的快捷途径。

本书具有以下主要特色。

① 较强的实用性:本书所列举的案例应用范围较广,涵盖了 Visual C#系统应用的大部分领域,对于正在进行课题设计或项目开发的读者来说,很多案例稍加修改就可以顺利应用。

② 较深的理论性:本书不是仅仅介绍如何应用 Visual C#开发应用系统,而是把软件系统工程系统分析贯穿于全书,可以帮助读者建立起系统分析的观念和规范性的设计思维,帮助读者从入门走向深入。

③ 广泛的读者群:本书适用于计算机及其相关专业的本科高年级学生和研究生,具有一定程序设计基础但缺少系统开发经验的程序设计人员,学习软件工程理论的程序设计人员,也可供 Visual C#开发人员和计算机编程爱好者参考。

全书共分 8 章,第 1 章和第 2 章是关于 Visual Studio .NET 的技术信息和 Visual C#的编程基础。初学者可以通过前两章的学习为后续案例开发奠定必备的基础和掌握程序设计的基本方法。有一定基础的读者可以将其作为对 Visual C#的复习,以利于后面章节的学习。第 3 章应用和服务编程的基本方法,主要介绍创建 Windows 应用程序项目、创建控制台应用程序项目、构建类库、创建组件和创建 Windows 服务及创建用户自定义控件的方法。第 4 章 数据应用编程案例,通过“航班查询与订票数据库管理系统”案例,介绍了数据库编程方法。第 5 章 多媒体应用编程案例,介绍了图形图像处理、动画与音、视频的编程方法,以及运用媒体播放器控件编程的技巧。第 6 章 网络通信编程案例,介绍了多种网络通信软件的编程方法,以及使用 socket 的 TCP 协议实现网络通信的编程技巧。第 7 章 Web 应用编程案例,介绍了创建 ASP .NET 网站的方法,通过 Web 服务编程案例“储蓄利息计算”展示了创建 Web Service 和编写 Web Service 客户端程序的基本方法。第 8 章 智能设备编程案例,介绍了当今最为流行的移动智能设备软件开发技巧,通过“公路票价查询系统”案例,详细介绍了如何在智能设备项目中实现各种任务,从而让读者掌握智能设备应用程序的开发、部署与

打包流程。所有案例代码均在物理设备上调试通过。

本书全部案例均附有完整的源代码和可执行程序以供引用和借鉴。希望通过这些典型案例的开发过程介绍，向读者展示 Visual C#的强大功能和软件开发的基本过程。书中错误之处，敬请批评指正。

编 者

2009 年 6 月

目 录

第 1 章 Visual Studio .NET 概述.....	1
1.1 Microsoft .NET 平台	1
1.2 Microsoft .NET 平台的基础	3
1.2.1 公共语言运行时	4
1.2.2 Framework 类库.....	5
1.3 Visual Studio .NET 集成开发环境介绍.....	5
1.3.1 可视化的开发设计界面	6
1.3.2 交互式的开发环境	9
1.3.3 编译、生成与调试工具	11
1.3.4 可视化数据设计器	12
1.3.5 部署	15
1.4 Visual C#简介	17
1.5 小结.....	18
练习题.....	19
第 2 章 Visual C#编程基础.....	20
2.1 程序结构.....	20
2.1.1 程序	20
2.1.2 类型	23
2.1.3 变量和常量	24
2.1.4 语句	24
2.2 命名空间	27
2.2.1 命名空间声明	27
2.2.2 using 指令	28
2.2.3 声明空间与完全限定名	30
2.2.4 可访问性	31
2.3 类及其成员	32
2.3.1 字段	34
2.3.2 属性	34
2.3.3 方法	35
2.3.4 构造函数	35
2.3.5 索引器	36
2.3.6 事件	36
2.3.7 运算符	36

2.3.8 析构函数	37
2.4 其他类型	37
2.4.1 结构	37
2.4.2 数组	37
2.4.3 接口	38
2.4.4 枚举	38
2.4.5 委托	38
2.4.6 属性	39
2.5 自动内存管理	40
2.6 异常	41
2.7 小结	42
练习题	42
 第 3 章 应用和服务编程的基本方法	43
3.1 创建 Windows 应用程序项目	44
3.2 创建控制台应用程序项目	50
3.3 构建类库	53
3.4 创建组件	57
3.5 创建 Windows 服务	62
3.6 小结	71
练习题	71
 第 4 章 数据应用编程案例	72
4.1 C#创建数据应用程序示例	72
4.2 ADO .NET 简介	75
4.2.1 数据库基础	75
4.2.2 ADO .NET 的结构	77
4.2.3 System.Data 命名空间	79
4.3 使用 ADO .NET 访问数据库	81
4.3.1 服务器资源管理器	81
4.3.2 连接数据源	82
4.3.3 创建数据集	87
4.3.4 DataAdapter/TableAdapter	88
4.3.5 TableAdapter 查询配置向导	93
4.3.6 数据表 DataTable	94
4.3.7 DataRelation	98
4.4 数据库示例	100
4.5 小结	110
练习题	111

第 5 章 多媒体应用编程案例	112
5.1 图形图像编程基础	112
5.1.1 GDI+概述	112
5.1.2 GDI+体系	114
5.1.3 坐标系统、画笔、画刷、字体及颜色	116
5.2 图形绘制	119
5.2.1 Graphics 类	119
5.2.2 图形绘制和操作	120
5.2.3 字体与文本	125
5.2.4 图形绘制编程实例	126
5.3 图像处理	131
5.3.1 图像、位图和图元文件	131
5.3.2 图像处理编程实例	136
5.4 动画与音视频编程	145
5.4.1 动画编程	145
5.4.2 运用媒体播放器控件编程	147
5.5 小结	156
练习题	156
第 6 章 网络通信编程案例	157
6.1 网络通信基础	157
6.1.1 Socket	157
6.1.2 UDP 协议	158
6.1.3 TCP 协议	158
6.1.4 网络编程相关对象	159
6.2 局域网聊天程序简介	161
6.3 局域网聊天程序系统实现	163
6.3.1 服务器端实现	163
6.3.2 客户端实现	181
6.3.3 程序运行	212
6.4 小结	214
练习题	214
第 7 章 Web 应用编程案例	215
7.1 ASP.NET 简介	215
7.1.1 ASP.NET 的优点	215
7.1.2 ASP.NET 基本语法与 Web 页结构	216
7.1.3 ASP.NET 内置对象	219
7.2 ASP.NET 应用程序开发	232

7.2.1	创建 ASP.NET 网站	233
7.2.2	使用控件	235
7.2.3	ASP.NET 网站示例	240
7.3	Web 服务编程示例——储蓄利息计算	256
7.3.1	Web 服务概述	257
7.3.2	创建 Web Service	259
7.3.3	编写 Web Service 客户端程序	265
7.4	小结	268
	练习题	269
第 8 章	智能设备编程案例	270
8.1	智能设备的基本概念	270
8.1.1	什么是智能设备	270
8.1.2	智能设备的操作系统	270
8.1.3	智能设备的编程特点	271
8.1.4	智能设备的开发平台	273
8.2	智能设备程序设计	273
8.2.1	智能设备程序开发环境	273
8.2.2	.NET Compact Framework	274
8.2.3	SQL Server 2005 Compact Edition	275
8.2.4	C#智能设备项目	276
8.3	公路票价查询系统案例	277
8.3.1	公路票价查询系统介绍	277
8.3.2	公路票价查询系统实现	278
8.3.3	部署查询系统应用程序	298
8.3.4	打包与部署智能设备解决方案	300
8.4	小结	303
	练习题	303
	参考文献	304

第 1 章 Visual Studio .NET 概述

Visual Studio .NET 是微软新一代服务与程序开发平台，用于创建 XML Web 服务和基于 Microsoft .NET 的应用程序。

Visual Studio .NET 是 Microsoft .NET 的重要组成部分，C# 是 Visual Studio .NET 中的主要成员之一，是以 .NET 为平台推出的一种新型的面向对象语言。C# 既拥有 C++ 的强大功能，又具有 Visual Basic 简易使用的特性。

探讨 C# 无法脱离 Microsoft .NET 平台，C# 程序的优越性必须依靠 Microsoft .NET 平台才能体现。

1.1 Microsoft .NET 平台

.NET 是一种面向网络支持各种用户终端的开发环境，是以 XML 为基础的新型的 Web 服务平台，建立于开放互联网络协议基础之上，并提供了新的工具和服务类型。

1. Microsoft .NET 的目标

Internet 的迅猛发展对网络服务提出更高要求，只提供浏览 HTML 以获取数据或文件的 Web 页已不能满足用户需要，而要求 Web 应用程序根据用户的需要从数据库中取出相应的数据，动态地生成包含更新数据的网页。在传统的 Internet 模式下，信息储存在服务器中，用户只能浏览它，难以进行更深层次的操作，.NET 的出现意味着用户可以编写、浏览、编辑和分享信息，还能得到功能强大的信息管理工具，任何用户都能使用相同的工具与其客户沟通，分享信息，形成一种全新的协同工作模式，其目标就是要实现在网络海洋中的各个“数字孤岛”之间的有机互联。

Microsoft .NET 的核心是 Web Service，Web Service 主要包括：分布式计算、XML、面向对象的组件技术、即时编译技术等。分布式计算是网络应用的目标；XML 是一种标记语言，作为 .NET 的数据交换标准；面向对象的组件技术用以实现程序的模块化和互操作性；即时编译技术能在程序运行时根据软硬件环境优化代码，并简化代码的发布。

Microsoft .NET 的优点包括以下几方面。

- 支持 Web Service，建立以网络应用为中心的新的技术框架，将当前的计算模式由单机模式、C/S 模式和 Web 网站模式转向分布式计算。Microsoft .NET 支持应用程序通过 Internet 在不同的设备、不同操作系统及不同编程语言开发的应用程序之间进行数据通信和数据操作。
- 支持软件服务从基于单一平台转向基于网络，支持多种终端设备，尤其是手机、笔记本电脑等无线移动设备和家庭智能网络设备等。
- 改进了用户界面，从传统桌面应用程序向基于浏览器的界面转变，增加对自然语言、视觉、手写、语音识别的支持，充分体现智能化与互动。
- 提高软件开发效率，使软件的发布更为便捷，克服不同 DLL 版本带来的障碍；实现资

源自动管理，可以自动分配、释放内存，解决了内存泄漏和无效内存引用问题。

2. Microsoft .NET 的特性

Microsoft .NET 定义了一种公用语言子集（Common Language Subset, CLS），为符合其规范的语言与类库之间提供无缝集成。Microsoft .NET 统一了编程类库，提供了对新一代网络通信标准——可扩展标记语言 XML（Extensible Markup Language）的支持，使开发应用程序变得更容易更简单。最重要的是，.NET 将改变 Internet 的应用方式，使得软件成为服务。Microsoft .NET 增加了对各种用户终端（包括智能移动终端）的支持。与 Microsoft 的其他产品一样，.NET 与 Windows 平台紧密集成，其运行库与操作系统融合在一起。

Microsoft .NET 支持多种开发语言，包括 C++、C#、VB、Perl 和 COBOL 等，其中 C# 是最主要的。

Microsoft .NET 支持即时编译方式（Just-In-Time, JIT），Microsoft 为各种语言提供了不同的 JIT 编译方式，对桌面应用程序提供标准编译器，而为移动计算环境提供压缩的 .NET 框架和经济型编译器（EconoJIT）。

Microsoft .NET 支持 XML/SOAP 协议，此协议由 W3C 组织提出，开放性较好，已有众多厂商支持。

Microsoft .NET 实现运行时与具体的编程语言分离的机制，所有资源管理、内存分配和变量类型等均由运行时处理，Microsoft .NET 的主要内容如图 1-1 所示，包括以下几部分。

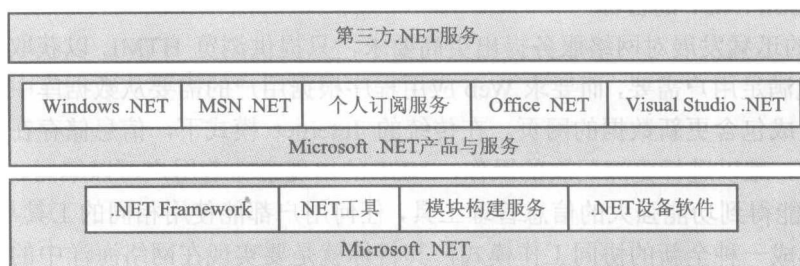


图 1-1 Microsoft .NET 组成

Microsoft .NET 由三部分组成：一个替代 COM 的编程环境或平台即“Microsoft .NET 平台”，一个由一系列 back-office server 支持的平台即“Microsoft .NET 产品和服务”，一个将要取代 Windows DNA 的平台，即“第三方.NET 服务”。知道这些将有助于了解 .NET 及其发展方向。

（1）Microsoft .NET 平台

Microsoft .NET 平台包括用于创建和操作新一代服务的 .NET 基础结构、工具及用于启用新一代智能 Internet 设备的 .NET 模块构建服务和 .NET 设备软件。其中 .NET 模块构建服务是一系列核心的因特网服务（Internet Services）。

（2）Microsoft .NET 产品和服务

Microsoft .NET 产品和服务包括 Windows .NET、MSN .NET、个人订阅服务、Office .NET、Visual Studio .NET 等。

Windows .NET 整合了 .NET 的一系列核心模块构建服务，为数字媒体及其协同工作提供全方位、个性化的支持，并为 .NET 应用程序和服务的开发提供了一个内容丰富的平台。

MSN .NET 是 MSN 与新型 .NET 平台的结合, MSN .NET 提供智能型服务, 确保用户在任何时间和地点都能持续而安全地了解他们所关心的各种内容与服务。

Office .NET 是高级的通信和生产工具, 将通信、浏览和资料撰写集成在统一的环境中, 使用户能够用互动的方式利用信息。通用协调服务使用户进行协作。基于服务的新型架构能够自动地运用到任何设备上, 功能更强大、效率更高。

Visual Studio .NET 是基于 XML 的编程模型和工具, 得到了 MSDN 和 Windows DNA 2000 服务器的充分支持, 支持在独立设备、企业数据中心和因特网之间进行高度分布的应用程序与服务开发。

(3) 第三方 .NET 服务

利用 Microsoft .NET, 第三方开发商可以开发出基于 .NET 平台的企业软件和服务程序。

总之, Microsoft .NET 将把计算和通信带入一个丰富、合作和互动的环境中, 远远胜过今天只提供数据浏览功能的网络。由新型高级软件支持的 Microsoft .NET 将利用一系列的应用程序、服务程序和相关设备来创造一种个性化的数字体验, 它能够不断地按照个性化的需要进行自动调整, 通过由一整套新型软件构成的整体服务程序来管理人们在因特网时代的生活和工作。

1.2 Microsoft .NET 平台的基础

Microsoft .NET Framework 是用于创建、部署和运行 Web 服务及其他应用程序的环境, 是 Microsoft .NET 的语言命令集, 它用统一的命令集支持所有 .NET 编程语言, 消除各种异类框架的束缚, 创建跨越编程语言的公共 API 集, 并提供强大的跨语言继承、错误处理和调试功能。

Framework 包括 2 个主要组件, 它们为 Microsoft .NET 平台提供底层技术支持:

- 公共语言运行时 (Common Language Runtime, CLR);
- Microsoft .NET Framework 类库。

CLR 作为 Microsoft .NET 平台的执行引擎, 建立在操作系统底层。类库用于开发应用程序和服务, 可用于任何符合 CLR 的语言。在类库之上是应用程序模板, 为开发应用程序和服务提供高级的组件和服务。

Framework 的组成如图 1-2 所示。

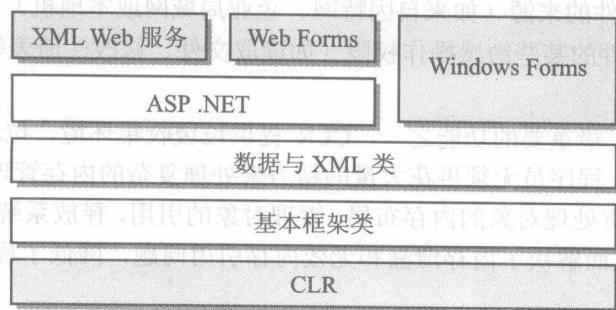


图 1-2 Framework 的组成

1.2.1 公共语言运行时

公共语言运行时（CLR）是一个在执行时管理代码的代理，提供内存管理、线程管理和远程处理等核心服务，并且还强制实施类型安全，以提高安全性和可靠性。

（1）托管

CLR 的基本原则是代码管理，CLR 为 Microsoft .NET 应用程序提供一个托管的代码执行环境。

托管指 CLR 驻留在内存中，负责在应用程序执行期间管理代码，包括内存管理、线程管理、安全管理、远程管理、即时编译、强制安全类型检查等。可见，CLR 代理了一部分操作系统的管理功能，在 CLR 管理下的代码称为托管代码，否则称为非托管代码。

（2）微软中间语言 MSIL

为了使应用程序独立于语言，CLR 并非由某种高级语言实现，而是由一种类似于汇编语言的语言——微软中间语言 MSIL 实现。MSIL 是 Microsoft .NET 平台下编译器输出的 PE 文件语言，是 Microsoft .NET 完整的语言集，即时编译器（JITer）在运行时将 MSIL 编译为机器码。

MSIL 为 Microsoft .NET 平台提供了多语言的底层技术支持机制，Microsoft .NET 为多种语言提供编译器，可以将其编译成 MSIL，从而可以使用 CLR。目前支持 CLR 的编程语言已多达二三十种。

（3）支持基于组件的编程

CLR 下的编程全部是围绕组件进行的。CLR 支持组件（包括属性和事件）、对象、继承性、多态性和接口，使得基于组件的编程变得更简单，而不需要特殊的接口和适配设计模式。在组件运行时，CLR 负责管理内存分配、线程/进程的启动和中止、强化安全系数，同时还调整相关组件的附属配置。序列化（Serialize）支持允许以多种格式操作存储在磁盘上的组件，包括基于 XML 的 SOAP。

（4）元数据

CLR 提供了处理错误条件的有力、协调的方式。每个模块都具有内置的完整的元数据，使得动态创建和方法调用等功能更安全简便。映射允许灵活地创建和执行代码。可以控制应用程序使用的组件的版本，使应用程序更可靠。组件代码采用 MSIL，不是某一种特定的机器语言，与处理器无关，因此组件可以在多种计算机上运行。

CLR 根据托管组件的来源（如来自因特网、企业局域网或本地机）等因素判断信任度，根据信任度来限定组件的某些敏感操作权限（如读取文件、修改注册表等）。

（5）内存管理

内存管理是 CLR 最重要的功能之一，CLR 提供垃圾收集环境，垃圾收集机制自动追踪应用程序操作的对象，程序员无须再花大量的精力来处理复杂的内存管理。在 Microsoft .NET 托管环境下，CLR 负责处理对象的内存布局，管理对象的引用，释放系统不再使用的内存（称为自动垃圾收集），从而解决了内存泄漏和无效内存引用问题，降低了程序开发负担。

（6）类型安全

通用类型系统（Common Type System, CTS）用于对代码类型进行严格的安全检查，避免了不同组件之间可能存在的类型不匹配的问题。

(7) 高效

CLR 已在 Internet 信息服务器 (IIS)、Microsoft SQL Server 等服务器应用程序中运用, 使用这些应用程序的用户可以充分利用 CLR 安全、高效的优点。

1.2.2 Framework 类库

Framework 类库是一组面向对象的可重用类的集合, 与公共语言运行库紧密集成。该类库为各种应用程序 (从命令行程序、Windows 程序到 ASP .NET 或 XML Web Service 程序) 提供以下各种组件和服务。

(1) 系统框架服务

服务框架包括一套基本类库 (如集合、输入/输出, 字符串及数据类)、访问操作系统服务的类 (如图画、网络、线程和加密)、数据访问类库以及开发工具。

(2) ADO .NET 组件

ADO .NET 为网络应用程序提供数据访问服务, 支持传统的基于连接的数据访问和适用于网络的无连接数据访问。

(3) XML 数据组件

XML 作为数据使用转换、传输和确认标准, 系统框架对 XML 数据提供操作支持。系统也支持 ADO .NET 数据与 XML 数据之间的通用转换。

(4) Windows 表单组件

提供 Windows 应用程序模型的 Windows 用户接口, 包括传统的 ActiveX 控件和 Windows XP 的新界面。

(5) ASP .NET 应用服务

用于处理基于低级结构的 HTTP 请求的运行语言, 使用基于构件的 Microsoft .NET 框架配置模板, 支持应用程序的实时更新、高速缓冲。

(6) ASP .NET Web 窗体

ASP .NET Web 窗体继承了基于 Visual Basic 的窗体设计的优点, ASP .NET Web 窗体支持传统的 ASP (HTML 与脚本代码混合), 但主要使用一种新型的、程序代码和用户接口分离的方法。ASP .NET 提供了功能齐全的 ASP .NET Web 窗体控件和 Web 服务器控件。

(7) XML Web 服务

ASP .NET 为建立 XML Web 服务提供了高级的可编程模板, 能够简化开发过程, 使用此模板, 开发人员甚至不需要理解 HTTP、SOAP 或其他任何网络服务规范, ASP .NET XML Web 服务为在 Internet 上绑定应用程序提供了一个简单而灵活的模式。

1.3 Visual Studio .NET 集成开发环境介绍

Visual Studio .NET 是 Microsoft .NET 创建应用程序和服务的工具, 提供集成开发环境 (IDE) 和在线资源的联机使用。IDE 集成了 VB .NET、C#、ASP .NET 的开发环境, 使 VB .NET 和 C# 能共用相同的运行库和组件, 使得编写 C# 程序和编写 VB .NET 程序同样简单高效。

Visual Studio .NET 具有可视化设计界面, 将整个解决方案的管理和开发集成到一起, 支持“所见即所得”的应用程序设计, 具备设计、开发、调试和部署功能。

Visual Studio .NET 的 IDE 由若干元素组成：菜单栏、标准工具栏、各种停靠或隐藏的工具窗。根据所处理的项目或文件类型以及用户的设置，工具窗、菜单和工具栏可以随意调整。IDE 比较复杂，以下分几个方面来介绍。

1.3.1 可视化的开发设计界面

Visual Studio .NET 具有直观的拖放式界面设计器（见图 1-3），内置 Windows XP 主题，支持创建具有 Windows XP 外观的应用程序。



图 1-3 Visual Studio .NET 的 IDE

1. 起始页

起始页为开发人员提供了一个门户，在这里可获取有关最近使用的项目、首选项、更新程序和 MSDN Online 社区的信息。

2. 视图

Visual Studio .NET 的 IDE 提供以下两种视图。

- 解决方案资源管理器：显示解决方案及其项目的组成和结构，可以添加、查看、打开各种文件（包括源代码文件、资源文件等）。

● 类视图：显示项目中所定义和所引用的类，可以展开类并察看类的组成、继承/派生关系以及类成员。

3. 编辑器和设计器

编辑器和设计器包括代码编辑器、文本编辑器、CSS 编辑器、HTML 设计器和网页设计器等，通常有两个视图：图形设计视图和代码设计视图。图形设计视图允许从工具箱中拖动控件到窗体上，“所见即所得”地进行编辑；代码设计视图用于显示文件或文档的源代码，有智能感知（IntelliSense）、可折叠代码、重构和代码段插入等功能。

4. 工具箱

工具箱集成了数十种控件，包括浮动工具条、超链接文本、NotifyIcon 等新的具有 Windows XP 风格和 Visual Studio .NET 风格的控件。这些控件可以直接拖到窗体上，进行可视化编辑，大大简化了程序的开发。

工具箱分为“公共控件”、“菜单和工具栏”、“容器”、“数据”、“组件”等多个分组，也可以在“所有 Windows 窗体”选项卡中按照名称顺序查看到所有控件。

(1) “公共控件”控件组

该控件组除了包含传统的按钮、列表框、文本框等控件和组件外，还包含表 1-1 所示的新控件。

表 1-1 “公共控件”控件组中的新组件

名 称	说 明
NotifyIcon	在任务栏状态通知区中为无界面的后台进程创建图标
LinkLabel	超链接文本
网络浏览控制	Web 浏览器控件
蒙板文本框 (MaskedTextBox)	增强型 TextBox 控件，可指定必需输入的字符、可选输入的字符以及掩码中的给定位置所需的输入类型（数字、字母或字母和数字）

(2) “容器”控件组

该控件组提供对窗体的布局控制，除了 GroupBox、TabControl、SplitContainer、Panel 等传统控件外，还有两种新的布局控件：TableLayoutPanel 和 FlowLayoutPanel，前者提供基于表格的布局控制，后者允许内容在 Form 上自由移动，水平或垂直方向布局的数据显示控件，可以切换到下一行或下一列。

(3) “菜单和工具栏”控件组

该控件组提供 MenuStrip、ContextMenuStrip、StatusStrip、ToolStrip、ToolStripContainer 等控件，其中 ToolStripContainer（浮动工具条容器）用于创建具有 Visual Studio .NET 风格的工具栏，可以拖曳到窗口任意位置停靠、浮动、展开或折叠。

(4) “组件”控件组

该控件组提供一些具有高级管理、监视功能的组件，多数不具有界面而在后台工作，如表 1-2 所示。

在该组中，除了 Visual Studio 附带的 .NET Framework 组件（如 MessageQueue 和 EventLog 组件）之外，还可以添加用户自己的组件或第三方组件。

表 1-2 “组件” 控件组

名 称	说 明
FileSystemWatcher	监视所有可访问的目录或文件的更改
EventLog	与系统和自定义事件日志进行交互，包括向日志中写入事件和读取日志数据
DirectoryEntry	封装 Active Directory 层次结构中的节点或对象，与 Active Directory 服务交互
DirectorySearcher	对 Active Directory 执行查询
MessageQueue	与消息队列进行交互，包括读取消息/写入消息、处理事务和执行队列管理任务
PerformanceCounter	与 Windows 性能计数器交互，创建类别和实例、读计数器及对计数器执行计算
Process	停止、启动和操作与系统上的进程关联的数据
ServiceController	操作现有服务，包括启动和停止服务及向服务发送命令
Timer	向 Web 应用程序或基于服务器的应用程序添加基于时间的功能
ReportDocument	创建和控制 Crystal Report

(5) “数据” 控件组

该控件组用于显示可以添加到窗体和组件的数据对象，包括表 1-3 所示控件。

表 1-3 “数据” 控件组

名 称	说 明
DataSet	数据集
DataGridView	以表格形式显示数据
BindingSource	用于绑定到数据源
BindingNavigator	绑定到数据的导航操作控件
ReportViewer	显示报表及其关联的数据

(6) “打印” 控件组

该控件组包括打印、打印设置、打印预览等控件。利用该控件组，用户无须编写很多代码即可获得高级打印控制功能。

(7) “对话框” 控件组

该控件组提供常见的颜色选择对话框、文件选择对话框、字体选择对话框等。



图 1-4 属性窗口

5. 属性窗口

属性窗口用于查看并设置组件的属性及事件，对于每个属性项或事件项都有下拉菜单供选择。单击可以为控件添加事件，事件窗口下面列出该控件所具有的所有事件，双击事件名即可添加事件处理函数（见图 1-4）。

6. 任务列表

任务列表跟踪当前工程任务完成情况，在尚未完成的代码段输入“//TODO: My comment here”并根据需要添加说明，任务列表功能自动将此代码段位置添