

C# 网络与通信

程序设计案例精讲

- 10个实用、高效的经典案例
- 案例讲解：必备知识－系统功能－系统设计－系统实现
- 案例来自实际项目，包括游戏开发、Web Service
- 完全针对C#开发，同时与ASP.NET结合

吴晨 陈建孝 编著

Programming



清华大学出版社

C#网络与通信程序设计案例精讲

吴 晨 陈建孝 编著

清 华 大 学 出 版 社

北 京

内 容 简 介

本书通过经典案例详细讲解了用 C# 语言开发网络与通信程序的方法和技巧。

全书分为两个部分。第 1 部分是 C# 网络编程基础,介绍了 C# 集成开发环境、网络编程基础知识、网络应用开发流程。第 2 部分是 C# 案例应用部分,共开发了 10 个完整的网络与通信技术系统;包括 FTP 服务器和客户端、电子邮箱客户端、聊天室、多线程下载、电子书店、网络游戏、浏览器、通信系统、Telnet 客户端等。案例讲解详细,涵盖面广,并都有一定的工程背景,具有很强的实用性。读者通过本书的学习,即可洞悉 C# 网络与通信编程的特色和优点,又可把这些技术应用到实际开发中。

本书主要面向 C# 开发人员、IT 各领域从业人员,也可以作为各高等院校师生的教学、自学丛书和社会相关领域培训班的首选教材。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

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。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

C#网络与通信程序设计案例精讲/吴晨,陈建孝编著. —北京:

清华大学出版社, 2006

ISBN 7-302-13673-4

I. C... II. ①吴... ②陈... III. C 语言—程序设计

IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 099487 号

出 版 者: 清华大学出版社

<http://www.tup.com.cn>

社 总 机: 010-62770175

地 址: 北京清华大学学研大厦

邮 编: 100084

客户服务: 010-82896445

组稿编辑: 夏非彼

文稿编辑: 刘秀青

封面设计: 林 陶

版式设计: 科 海

印 刷 者: 北京市耀华印刷有限公司印刷

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 25.25 字数: 614 千字

版 次: 2006 年 10 月第 1 版 2006 年 10 月第 1 次印刷

书 号: ISBN 7-302-13673-4/TP·8249

印 数: 0 001~4 000

定 价: 39.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话: (010) 82896445

前 言

随着计算机技术和网络技术的发展,基于网络的开发也越来越多。C#是微软开发的一种简单易懂、面向对象、类型安全的编程语言。随着.NET技术的不断发展,C#在各个方面的编程优势都已经显现出来。本书将给读者讲解用C#进行网络与通信编程的技术和方法,向读者阐述C#在网络开发技术方面的独特优势。

本书共11章,各章内容如下。

第1章主要对C#的开发环境、网络开发功能、网络应用开发流程等方面做了比较系统的介绍,使读者可以对C#的网络开发应用有一个比较全面的了解。

第2章是FTP服务器的开发,重点介绍了一个FTP服务器接收命令、发送响应和数据的方法。

第3章是FTP客户端的开发,重点介绍FTP客户端访问服务器接收发送文件的应用。

第4章是电子邮箱客户端的实现,重点介绍了邮件的封装、邮件的接收和发送的应用。

第5章是基于Socket聊天室的实现,介绍了聊天室服务器端和客户端的实现方法。

第6章是多线程下载的实现,介绍了多线程开发的知识。

第7章是基于Web的电子书店的实现,对数据库中使用的存储过程和数据库后台应用进行了具体的讲解。

第8章是网络游戏的实现,介绍了游戏服务器和客户端的实现方法,以及游戏的实现方法。

第9章是基于HTTP协议的浏览器实现,介绍了浏览器的设计和实现的过程,以及浏览器的一些简单操作的实现方法。

第10章是基于Web Service技术通信的实现,介绍了Web Service技术服务器端和客户端的实现方法。

第11章是Telnet客户端实现,重点介绍了Telnet选项的定义和命令模式,实现了客户端与服务器端的连接。

最后的附录部分对C#的Socket类进行了说明,对于该部分内容不太了解的用户可以翻阅参考。

本书所列举的应用基本涵盖了网络与通信编程中常见的各类网络应用程序。由于C#是基于.NET框架的,因此本书中并不单纯讲解C#的网络应用,还讲解了C#结合ASP.NET开发网络应用的方法,使得本书的内容既有基于窗体的网络开发,也有基于Web的网络开发。

本书既兼顾了传统技术,如FTP、电子邮箱、Telnet等,又涵盖了目前比较流行网络技术,如Web Service、网络游戏开发技术等。每个章节内容在安排上,注重了案例与技术的结合,对于关键技术,笔者做了比较详细的阐述,使得读者在阅读案例时能够比较容易理解。

本书由吴晨和陈建孝编著整理，其中第7章由陈建孝执笔，其余章节由吴晨执笔；此外，张爱华、缪珩珩、黄瑜、张一琳、冒小飞、张蓓、张英、朱勇、冯志刚、潘华、金伟、缪辉、戴旭东、滑朝辉、刘斌、程志艳、张亮、许宝建、蔡东军、梁小军和刘小松等同志在整理材料方面给予了编者很大的帮助，在此，编者对他们表示衷心的感谢。

由于本书涉及的范围比较广泛，作者的经验有限，书中难免有不足或错误之处，敬请各位专家和广大读者批评指正。E-mail: LXQ@KHP.COM.CN

编 者

2006年8月

目 录

第 1 章 C#网络编程基础 1	2.5 系统实现 27
1.1 Visual C#集成开发环境 2	2.5.1 服务器主界面..... 27
1.1.1 起始页 2	2.5.2 封装客户信息..... 32
1.1.2 解决方案资源管理器 3	2.5.3 处理客户端的请求..... 39
1.1.3 类视图 4	2.6 程序的运行和测试 50
1.1.4 服务器资源管理器 4	
1.1.5 属性窗口 5	第 3 章 FTP 客户端 52
1.1.6 工具箱窗口 5	3.1 FTP 命令 53
1.2 Visual C#网络通信编程基础 6	3.1.1 连接操作 53
1.2.1 网络体系结构与网络编程..... 6	3.1.2 设置模式 55
1.2.2 .NET 中的网络编程组件..... 7	3.1.3 目录操作 56
1.2.3 .NET 中的 Socket 编程..... 7	3.1.4 发送操作 58
1.3 Visual C#网络应用开发流程 7	3.1.5 获取操作 58
1.3.1 系统功能分析 8	3.1.6 其他命令 59
1.3.2 系统设计 8	3.2 必备知识 59
1.3.3 系统实现 8	3.2.1 TreeView 和 ListView 的使用..... 60
1.3.4 程序的调试和运行 12	3.2.2 文件访问方法..... 63
第 2 章 FTP 服务器 16	3.3 系统功能 64
2.1 FTP 协议 17	3.3.1 主界面 64
2.1.1 FTP 命令 17	3.3.2 登录管理 65
2.1.2 FTP 响应 18	3.3.3 目录管理 65
2.2 必备知识 19	3.3.4 文件传输 66
2.2.1 接收命令 19	3.4 系统设计 66
2.2.2 发送响应和数据 20	3.5 系统实现 68
2.2.3 如何获取本机 IP 地址 21	3.5.1 主界面 68
2.3 系统功能 23	3.5.2 登录和退出服务器..... 73
2.3.1 主界面 23	3.5.3 进入目录 76
2.3.2 登录管理 24	3.5.4 显示当前目录..... 77
2.3.3 目录管理 24	3.5.5 删除文件或者目录..... 78
2.3.4 文件传输 25	3.5.6 创建子目录 78
2.4 系统设计 25	3.5.7 目录或者文件改名..... 78
	3.5.8 列目录 79
	3.5.9 发送文件 83

3.5.10 接收文件	86	5.3.1 聊天室服务器	155
3.5.11 其他功能	87	5.3.2 聊天室客户端	155
3.6 程序的运行和测试	88	5.4 系统设计	156
第4章 电子邮箱客户端	90	5.4.1 聊天室服务器的协议解析流程	156
4.1 POP 和 SMTP 协议	91	5.4.2 聊天室客户端的协议解析流程	158
4.1.1 POP3 协议	91	5.5 系统实现	159
4.1.2 SMTP 协议	94	5.5.1 服务器端设计	159
4.2 必备知识	96	5.5.2 客户端设计	170
4.2.1 发送命令	96	5.6 程序的运行和测试	177
4.2.2 解析邮件	97	第6章 多线程下载	181
4.3 系统功能	101	6.1 多线程概述	182
4.3.1 主界面	101	6.2 必备知识	182
4.3.2 账户管理	102	6.2.1 Thread (线程) 类	182
4.3.3 新建邮件	103	6.2.2 WebClient 类	183
4.3.4 接收/发送邮件	104	6.2.3 WebRequest 类	184
4.4 系统设计	104	6.3 系统功能	184
4.4.1 数据库设计	104	6.4 系统设计	184
4.4.2 主要功能设计	105	6.5 系统实现	185
4.5 系统实现	108	6.5.1 主程序架构	185
4.5.1 主程序框架	108	6.5.2 线程的创建	190
4.5.2 封装邮件信息	117	6.5.3 停止线程	190
4.5.3 封装账号信息	120	6.5.4 文件下载	191
4.5.4 封装邮件操作功能	124	6.6 程序的运行和测试	194
4.5.5 基于 POP3/SMTP 的邮件接收和 发送操作	129	第7章 基于 Web 的电子书店	195
4.5.6 设置邮箱账号	140	7.1 Web 应用概述	196
4.5.7 邮箱属性	143	7.2 必备知识	196
4.5.8 编辑邮件	147	7.2.1 存储过程	196
4.5.9 接收和发送进度显示	149	7.2.2 ASP.NET 调用存储过程	198
4.6 程序的运行和测试	149	7.3 系统功能	199
第5章 基于 Socket 的聊天室	150	7.3.1 主界面	199
5.1 服务器/客户端聊天室模型	151	7.3.2 用户登录管理	199
5.2 必备知识	152	7.3.3 图书查询	200
5.2.1 聊天协议的应答	152	7.3.4 浏览图书信息	202
5.2.2 聊天状态	153	7.3.5 购物车	202
5.2.3 聊天命令	153	7.4 系统设计	203
5.3 系统功能	154		

7.4.1 数据库设计	204	9.3.4 浏览器设置	292
7.4.2 模块设计	206	9.3.5 历史记录	292
7.5 系统实现	208	9.4 系统设计	293
7.5.1 主程序架构	209	9.5 系统实现	293
7.5.2 用户注册	225	9.5.1 主程序架构	294
7.5.3 用户登录	228	9.5.2 浏览功能实现	298
7.5.4 用户信息修改	230	9.5.3 浏览设置	299
7.5.5 按类型查询图书	233	9.5.4 浏览器设置	300
7.5.6 详细查询	236	9.5.5 历史记录	301
7.5.7 浏览图书信息	239	9.6 程序的运行和测试	302
7.5.8 购物车模块	241		
7.6 程序的运行和测试	251		
第 8 章 网络游戏	253	第 10 章 基于 Web Service 技术的通信	304
8.1 网络游戏概述	254	10.1 Web Service 概述	305
8.2 必备知识	254	10.2 必备知识	306
8.3 系统功能	255	10.3 系统功能	307
8.3.1 登录界面	255	10.4 系统设计	309
8.3.2 游戏主界面	255	10.4.1 数据库的设计	309
8.3.3 连接服务器	256	10.4.2 Web 服务的设计	311
8.3.4 游戏	256	10.4.3 客户端的设计	312
8.3.5 判断胜负	256	10.5 系统实现	314
8.4 系统设计	257	10.5.1 数据库的实现	314
8.5 系统实现	258	10.5.2 Web 服务的实现	326
8.5.1 登录窗口的实现	258	10.5.3 客户端的实现	341
8.5.2 游戏功能的实现	264	10.6 程序运行和测试	352
8.5.3 游戏服务器的实现	270		
8.5.4 游戏客户端实现	276	第 11 章 Telnet 客户端实现	353
8.6 程序的运行和测试	282	11.1 Telnet 协议	354
第 9 章 基于 HTTP 协议的浏览器	283	11.1.1 Telnet 选项协商	354
9.1 HTTP 协议概述	284	11.1.2 协商子选项	356
9.2 必备知识	284	11.1.3 网络虚拟终端字符	356
9.2.1 Registry 类	285	11.2 必备知识	357
9.2.2 RegistryKey 类	286	11.2.1 Telnet 选项的定义	357
9.3 系统功能	288	11.2.2 Telnet 命令模式	358
9.3.1 主界面	288	11.3 系统功能	358
9.3.2 浏览网页	291	11.3.1 主界面	358
9.3.3 浏览设置	291	11.3.2 连接设置	359
		11.4 系统设计	360
		11.5 系统实现	361

11.5.1 主界面	362
11.5.2 封装账户信息.....	381
11.6 程序的运行和测试	383

附录 .NET 中的 Socket 编程	386
-----------------------------------	------------

A Socket 简介	387
B Socket 构造函数.....	387
C Socket 属性.....	387
D Socket 方法	389

第1章

C#网络编程基础

- ▶▶ Visual C#集成开发环境
- ▶▶ Visual C#网络通信编辑基础
- ▶▶ Visual C#网络应用开发流程

微软公司推出的“Microsoft.NET 下一代互联网软件和服务战略”，通过先进的软件构架、网络协议和格式，来协调众多的智能设备与网站，从而提供更为简单、更为个性化和更为有效的互联网服务。在本章，读者需简单了解C#.NET的基础内容，重点学习Visual C#开发环境、Visual C#网络编程基本知识和Visual C#网络应用开发流程。

1.1 Visual C#集成开发环境

Visual C#集成开发环境提供了优秀的多种编程语言的编辑和编译统一开发平台，它是快速开发企业级 Web 应用程序以及高性能桌面应用的工具，如图 1-1 所示。下面对 Visual C#集成开发环境特有的内容进行介绍。

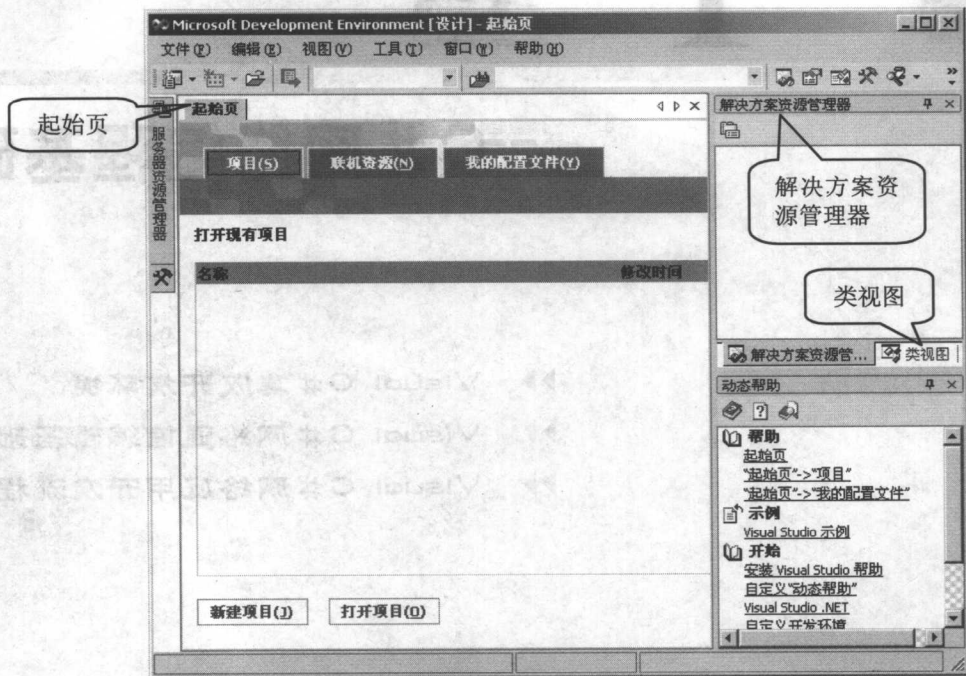


图 1-1 Visual C#集成开发环境

1.1.1 起始页

起始页是打开 Visual.NET 2003 时显示的第一个窗口（见图 1-2），每次启动时，该页都会作为默认页显示。在起始页中包含了 3 个选项卡，分别为：项目、联机资源和我的配置文件。

1. “项目”选项卡

“项目”选项卡用于打开已有项目或新建项目，其中包括 3 个选项：

- 打开现有项目：通过单击下面列表中的项目名称，可打开此项目。列表中的项目为最近编辑过的项目。
- 新建项目：可打开“新建项目”对话框。用户在对话框中可选择新建项目的名称、存储位置等。
- 打开项目：通过选择路径，打开已存在的某个项目。

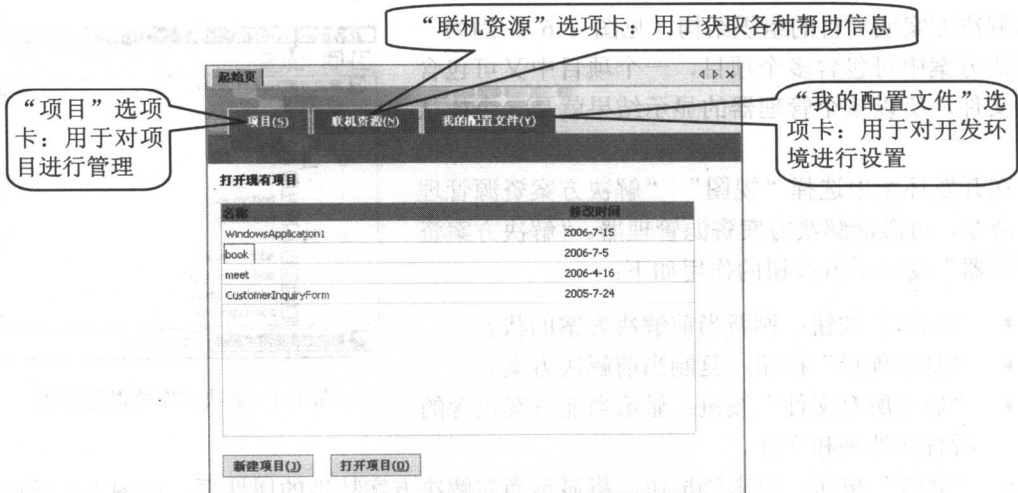


图 1-2 起始页

2. “联机资源”选项卡

“联机资源”选项卡（见图 1-3）用于获取各种帮助信息，包括以下内容：开始、新增功能、网上社区、标题新闻、联机搜索、下载、XML Web、services、Web 宿主。单击窗口左侧的导航条，可显示对应的信息，实现导航条中的功能。例如单击“下载”链接，窗口如图 1-4 所示。在此窗口中可实现下载功能。

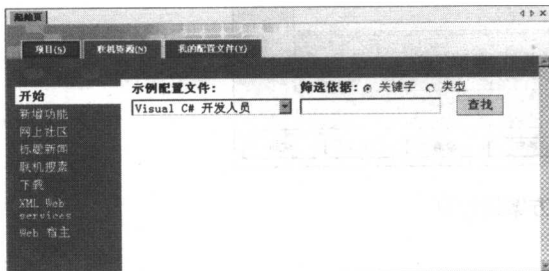


图 1-3 “联机资源”选项卡



图 1-4 下载功能

3. “我的配置文件”选项卡

“我的配置文件”选项卡是用户根据个人爱好对开发环境进行设置的场所。在“配置文件”下拉列表框中包含了多种语言的开发设计方案，此处我们应该选择“Visual C#开发人员”选项；“键盘方案”、“窗口布局”和“帮助筛选器”保持默认选项即可；在“启动时”下拉列表框中选择“加载最近加载的解决方案”选项，如图 1-5 所示。

1.1.2 解决方案资源管理器

“解决方案资源管理器”以可视化的方式显示已经创

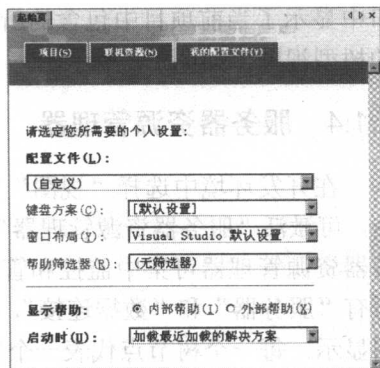


图 1-5 个人设置

建的解决方案或项目的组织结构(见图 1-6)。因为一个解决方案中可包含多个项目,一个项目中又可包含多个文件,所以这个管理器的显示结果就是一个树状图。

在开发环境中选择“视图”|“解决方案资源管理器”命令,可激活解决方案资源管理器。“解决方案资源管理器”窗口上方按钮的作用如下:

- “刷新”按钮:刷新当前解决方案的状态。
- “复制项目”按钮:复制当前解决方案。
- “显示所有文件”按钮:显示当前方案包含的所有文件夹和文件。
- “属性”按钮:单击该按钮,将显示当前解决方案特性的属性页,如图 1-7 所示。

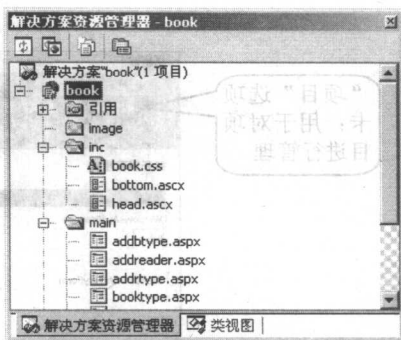


图 1-6 解决方案资源管理器

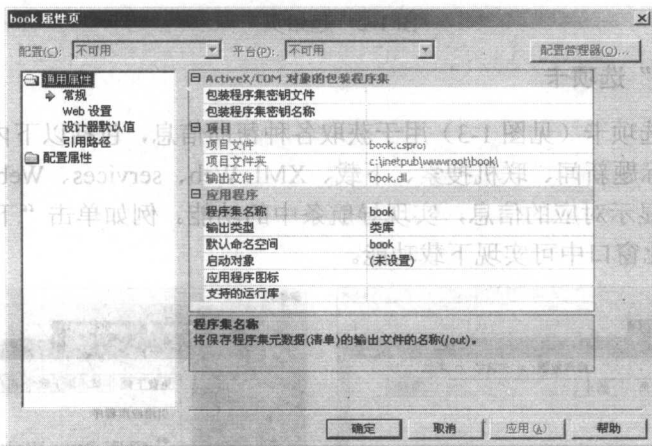


图 1-7 解决方案属性页

1.1.3 类视图

类视图窗口与解决方案资源管理器窗口层叠。在此窗口中显示了当前项目中包含的所有类、方法和特性等信息的树型视图,如图 1-8 所示。

1.1.4 服务器资源管理器

在开发环境中选择“视图”|“服务器资源管理器”命令,可激活“服务器资源管理器”窗口,如图 1-9 所示。服务器资源管理器可集中监控和管理一个应用方案所涉及的所有“服务器”和“数据连接”,在窗口中服务器以树型视图显示,每一个树节点代表一个服务器。

在监控服务器之前,必须先注册该服务器,方法如下:

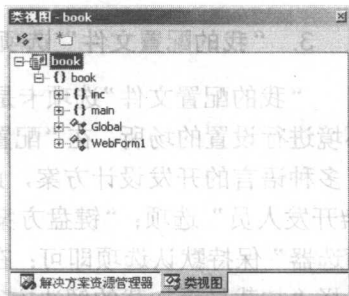


图 1-8 类视图窗口

(1) 在“服务器资源管理器”窗口中的“服务器”节点上单击鼠标右键，在弹出的快捷菜单中选择“添加服务器”命令，打开“添加服务器”对话框，如图 1-10 所示。

(2) 在“计算机”文本框中输入计算机名称或 IP 地址等信息，单击“确定”按钮，则被注册的服务器将作为一个新节点添加到树中。

用户在数据连接中可实现修改表、视图和存储过程等操作。创建新数据库的方法如下：

(1) 在“服务器资源管理器”窗口中的“数据连接”节点上单击鼠标右键，在弹出的快捷菜单中选择“创建新 SQL Server 数据库”命令，打开“创建数据库”对话框，如图 1-11 所示。

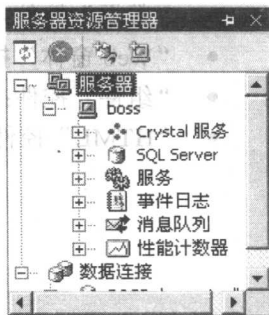


图 1-9 服务器资源管理器

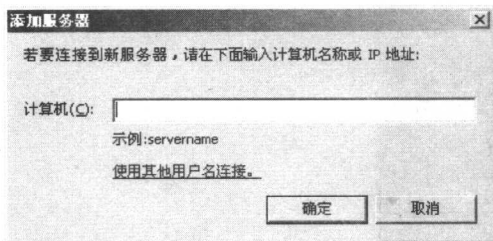


图 1-10 “添加服务器”对话框

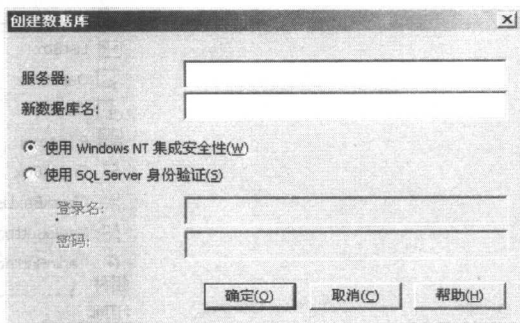


图 1-11 “创建数据库”对话框

(2) 在“服务器”文本框中输入 SQL Server 服务器的名称，在“新数据库名”文本框中输入数据库的名称。若数据库服务器使用 Windows NT 集成安全性登录，则选择“使用 Windows NT 集成安全性”单选按钮，否则选择“使用 SQL Server 身份验证”单选按钮，此时“登录名”和“密码”文本框变化为可编辑状态，在“登录名”和“密码”文本框中输入登录数据库的信息，单击“确定”按钮，即可实现对数据库的访问。

1.1.5 属性窗口

“属性窗口”用来查看或设置控件、类或项目的特殊属性。只需选中某个控件或类，该窗口就会显示相应项的特性，并可对特性进行编辑，如图 1-12 所示。

1.1.6 工具箱窗口

将鼠标移至图 1-1 窗体左侧的工具箱标签上，将弹出工具箱窗口，如图 1-13 所示。在工具箱窗口中包含了許多用于界面设计的控件。开发 C# 的 Web 应用时，控件可划分为以下几类：



图 1-12 属性窗口

- “数据”控件：用于应用程序与数据库的接口。
- “Web 窗体”控件：用于 Web 窗体中的控件。
- “组件”控件：用于窗体开发的组件。
- “HTML”控件：用于 Web 开发中的 HTML 应用。

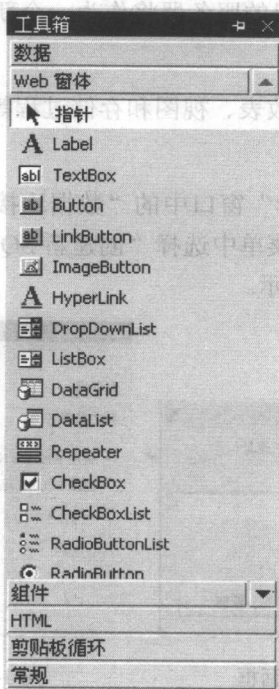


图 1-13 工具箱

1.2 Visual C#网络通信编程基础

本节主要向读者介绍 Visual C#中与网络通信编程相关的必备知识，是后面各章具体应用开发的基础。

1.2.1 网络体系结构与网络编程

OSI/RM 是 ISO 在网络通信方面所定义的开放系统互连模型，它定义了一个开放的协议标准。此协议共分 7 层，其模型如图 1-14 所示。

在网络体系结构中，各个层在网络的通信中起着不同的作用：

- 物理层 这是整个 OSI 参考模型的最底层，它的任务就是提供网络的物理连接。物理层提供的服务包括：物理连接、物理服务数据单元顺序化和数据电路标识。
- 数据链路层 此层的主要任务就是进行数据封装和数据链接的建立。此层的功能包括：数据链路连接的建立与释放，构成数据链路数据单元，数据链路连接的分裂、定界与同步，顺序和流量控制，差错的检测和恢复等方面。

- 网络层 此层用于解决的是网络与网络之间连接。此层的功能包括：建立和拆除网络连接、路径选择和中继、网络连接多路复用、分段和组块、服务选择和传输、流量控制等。
- 传输层 此层解决的是数据在网络之间的传输质量问题。此层的功能包括：映像传输地址到网络地址、多路复用与分割、传输连接的建立与释放、分段与重新组装、组块与分块。
- 会话层 此层利用传输层来提供会话服务。此层的功能主要有：会话连接到传输连接的映射、数据传送、会话连接的恢复和释放、会话管理、令牌管理和活动管理。
- 表示层 此层用于数据管理的表示方式。此层的功能主要有：数据语法转换、语法表示、表示连接管理、数据加密和数据压缩。
- 应用层 此层直接面对用户的具体应用，也就是用户通常应用的使用界面。

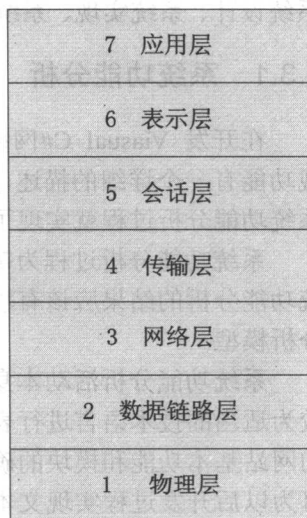


图 1-14 网络体系模型

在 Visual C#网络通信开发中，应用到了 TCP/IP 协议、HTTP 协议、FTP 协议、Telnet 协议和 SMTP 协议。其中 IP 协议和 SMTP 协议应用于网络层，为网络层协议；TCP、HTTP、FTP 和 Telnet 协议应用于传输层，为传输层协议。在应用层和网络层之间是通过套件字（Socket）连接起来的。Socket 位于协议之上，屏蔽了不同网络协议之间的差异。应用这些协议实现网络通信的开发在后面章节中都有具体的应用实例。

1.2.2 .NET 中的网络编程组件

.NET 网络编程组件是 Visual C#网络编程的基础，网络数据的传输在很大程度上依赖这些组件来完成。这些组件可以用来开发邮件服务、分布式数据库等各种功能强大的程序。

在 .NET 网络编程组件中，比较重要的组件就是 Socket 组件，此组件存在于 System.Net.Sockets 命名空间中，在网络编程中此组件中的属性、方法将会经常被使用。

1.2.3 .NET 中的 Socket 编程

.NET 中的 Socket 类主要用于在 TCP/IP 网络中传送和接收数据。使用 Socket，能够在网络上处理复杂数据。在各种网络应用程序中，如果涉及到数据传送和接收，一般都会使用 Socket。

有关 Socket 编程的详细技术细节，可以参考附录。

1.3 Visual C#网络应用开发流程

应用 Visual C#开发网络实例的流程可分为 4 个部分，流程顺序为：系统功能分析、

系统设计、系统实现、系统的运行和调试。下面对流程中各部分需要完成的功能进行介绍。

1.3.1 系统功能分析

在开发 Visual C#网络应用中,系统开发人员需了解系统要实现的功能,并对每个实现功能有一个详细的描述,使客户和程序开发人员对系统实现的功能有一个明确的了解。系统功能分析过程就实现了上面所阐述的功能。

系统功能分析过程为需求的收集、分析、细化及核实的步骤,并将它编写成文档。系统功能分析的结果应该有项目视图和范围文档、使用实例文档、软件需求规格说明及相关分析模型。

系统功能分析活动本身就是一个和客户交流、正确引导客户能够将自己的实际需求用较为适当的技术语言进行表达以明确项目目的的过程,这个过程中也同时包含了对要建立的网站基本功能和模块的确立和策划活动。开发人员建立一个明确的系统功能分析结果,可为以后开发过程实现文本形式的备忘,并可为日后的开发项目提供有益的借鉴和范例,成为项目开发中积累的符合自身特点的经验财富。

1.3.2 系统设计

在实现系统功能分析后,系统开发人员的工作就是对系统模块流程和数据库进行设计。

1. 模块流程设计

模块流程设计过程为需求分析的明确化、具体化。在模块流程设计中,开发人员对需求分析中的要求建立功能模块,将模块实现的功能用文档和流程图的方式表示。在模块设计时,需要将功能详细的实现流程用适当的技术语言描述清楚,将模块操作中要注意的约束条件讲解清楚。

2. 数据库设计

数据库的设计包括:数据库、数据表、主键、外键、约束、视图和存储过程等应用。在网络实例数据库的应用中,数据传输比较慢,而且数据量比较大,因此多用存储过程实现数据的操作。

在数据库设计中,要对实例中应用的实体建立 E-R 图,再将实例具体化,并根据 E-R 图建立实例的数据表,随后建立主键、约束等应用。数据库的设计可说是实例应用最为关键的部分,一个好的数据库设计不但能减轻程序编写的压力,而且也可轻松实现系统的升级、更新以及维护。

1.3.3 系统实现

系统实现由程序人员根据模块设计和数据库设计进行代码的编写,最终实现模块功能的应用。在第 7 章中应用了一个 Web 开发的实例,下面以该系统为例介绍 Visual C# + ASP.NET 类 Web 应用程序的开发过程。