

C# .NET

网络程序设计

黄嘉辉 编著



C# .NET 网络程序设计

黄嘉辉 编著

科 学 出 版 社

北 京

图字: 01-2003-7124

内 容 简 介

本书是通过 C#讲解 TCP/IP 与网络程序设计的书籍。

全书共分为 11 章和 13 个附录,主要介绍了 Microsoft.NET Framework、网络概述、Socket、基本应用、服务器程序结构、客户端程序结构、SMTP 协议、POP3 协议、System.Web.Mail、FTP 协议、HTTP 协议与 Web 服务器等方面的内容。

本书内容丰富,讲解详尽,可作为计算机及相关专业的研究生、本科生、大专生的学习参考书,同时也可作为相关专业从业人员的实用参考书。

本书繁体字版名为《C#.NET 网际网路程式设计》,由文魁信息股份有限公司出版,版权属黄嘉辉所有。本书简体字中文版由文魁信息股份有限公司授权科学出版社独家出版。未经本书原版出版者和本书出版者书面许可,任何单位和个人均不得以任何形式或任何手段复制或传播本书的部分或全部。

图书在版编目(CIP)数据

C#.NET 网络程序设计/黄嘉辉编著. —北京:科学出版社,2004

ISBN 7-03-014500-3

I. C… II. 黄… III. ① C 语言-程序设计② 因特网-程序设计
IV. ① TP312 ② TP393.4

中国版本图书馆 CIP 数据核字(2004)第 108075 号

责任编辑:吕建忠 丁 波/责任校对:柏连海

责任印制:吕春珉/封面设计:北新华文

科 学 出 版 社 出 版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

新 蕾 印 刷 厂 印 刷

科学出版社发行 各地新华书店经销

*

2004 年 11 月 第 一 版 开本: 787×1092 1/16

2004 年 11 月 第一次印刷 印张: 23

印数: 1--4 000 字数: 531 000

定价: 36.00 元

(如有印装质量问题,我社负责调换<环伟>)

前 言

2002 年, Microsoft 公司推出了新一代的平台与开发工具——Microsoft.NET Framework 与 Visual Studio.NET, 2003 年, Microsoft 公司进一步改进 Microsoft.NET Framework 与 Visual Studio.NET, 推出了 Microsoft Visual Studio.NET 2003。

Microsoft.NET Framework, 简单地说, 就是 Microsoft 为 XML Web Services 所提供的平台。Microsoft.NET 构建开放式的机构, .NET 平台的目标是简化 Web 开发过程, Microsoft.NET Framework 设计的目的在于能够与网络相关标准完整结合, 如通过 XML Web Services, 数据可以轻松从网上取得, 并且通过 XML 的应用, 数据可以在不同的系统平台之间轻松交换。

本书将以 C#来介绍如下的应用:

- 服务器端 Socket 程序构建。
- 客户端 Socket 程序构建。
- 相关通信协定。

书中还介绍了如下所示的 Microsoft.NET Framework 类库的类别:

- System.Uri。
- System.Net。
- System.Net.Cookie。
- System.Net.HttpWebRequest。
- System.Net.HttpWebResponse。
- System.Net.Sockets.Socket。
- System.Net.Sockets.TcpListener。
- System.Net.Sockets.TcpClient。
- System.Web.Mail。

本书得以完成, 要特别感谢阎慧玲、宋倩芬、廖雯霞、侯克明、林硕彦、蒋贞豪、邱伟光的建议与鼓励。

黄嘉辉

目 录

第1章 Microsoft .NET Framework.....	1
1.1 Microsoft .NET Framework概述.....	1
1.2 CLR.....	2
1.3 Microsoft.NET Framework 类库概述.....	2
1.4 安装Visual Studio.NET 2003	5
第2章 网络概论.....	11
2.1 历史与沿革.....	11
2.2 OSI七层模型	13
2.3 网络结构与局域网.....	15
2.4 TCP/IP.....	17
2.5 TCP与UDP	18
2.6 IP地址	19
2.7 IPv6协议	21
2.8 DNS.....	23
2.9 端口.....	25
2.10 RFC.....	26
第3章 Socket.....	29
3.1 Berkeley Socket	29
3.2 Java Socket.....	32
3.3 Microsoft Windows Socket API	37
3.4 Microsoft WinSock Control.....	42
3.5 Microsoft .NET Framework类库.....	46
3.5.1 System.Net	46
3.5.2 System.Net.Sockets.Socket	47
3.5.3 System.Net.Sockets.TcpListener	49
3.5.4 System.Net.Sockets.TcpClient	51
3.5.5 System.Net.Sockets.UdpClient.....	52
第4章 基本应用.....	54
4.1 IP与DNS	54
4.1.1 System.Net.IPAddress类	55
4.1.2 System.Net.IPEndPoint类	58
4.1.3 System.Net.DNS类	60
4.2 URL与System.Uri类.....	66
4.3 Cookie与System.Net.Cookie类	70

4.4	请求与响应.....	72
4.4.1	HTTP请求.....	74
4.4.2	System.Net.HttpWebRequest类.....	75
4.4.3	HTTP响应.....	78
4.4.4	System.Net.HttpWebResponse类.....	79
第5章	服务器程序结构.....	85
5.1	服务器应用程序流程.....	85
5.2	建立服务器Socket.....	86
5.2.1	Socket.....	86
5.2.2	TcpListener.....	92
5.3	接受客户端连接.....	94
5.3.1	Socket.Accept.....	95
5.3.2	TcpListener.AcceptSocket / AcceptTcpClient.....	101
5.4	接收与发送客户端信息.....	103
5.4.1	Socket.Send / Receive.....	103
5.4.2	服务器输入输出数据流——NetworkStream.....	108
5.5	关闭连接.....	112
5.5.1	Socket.Close.....	112
5.5.2	TcpListener.Stop.....	113
5.6	DateTime服务器应用.....	114
第6章	客户端程序结构.....	119
6.1	客户端应用程序流程.....	119
6.2	建立客户端Socket.....	120
6.2.1	Socket.....	120
6.2.2	TcpClient.....	122
6.3	连接服务器.....	124
6.3.1	Socket.Connect.....	124
6.3.2	TcpClient.Connect.....	125
6.4	接收与传输服务器信息.....	127
6.4.1	Socket.Send / Receive.....	127
6.4.2	客户端输入输出数据流——NetworkStream.....	128
6.5	关闭连接.....	130
6.5.1	Socket.Close.....	130
6.5.2	TcpClient.Close.....	131
6.6	DateTime客户端应用.....	131
第7章	SMTP协议.....	135
7.1	SMTP介绍.....	135
7.1.1	邮件格式.....	136
7.1.2	SMTP信息.....	137

7.1.3 SMTP指令	139
7.1.4 SMTP流程	140
7.2 SMTP Client——Socket	142
7.3 SMTP Client——TcpClient	153
第8章 POP3协议	156
8.1 POP3介绍	156
8.1.1 认证	158
8.1.2 邮件发送	159
8.1.3 更新	162
8.2 POP3 Client-Socket	162
8.3 POP3 Client-Tcp客户端	166
第9章 System.Web.Mail	169
9.1 System.Web.Mail介绍	169
9.2 指定SMTP服务器	171
9.3 设置E-mail地址	171
9.4 处理E-mail内容	172
9.5 处理附件	173
9.6 发送邮件	175
9.7 System.Web.Mail应用	176
第10章 FTP协议	186
10.1 FTP协议介绍	186
10.1.1 FTP结构	186
10.1.2 FTP指令	187
10.1.3 FTP 服务器响应信息	193
10.1.4 FTP流程	195
10.2 FTP服务器范例	197
第11章 HTTP协议与Web服务器	218
11.1 HTTP协议	218
11.1.1 HTTP标题	219
11.1.2 HTTP方法	221
11.1.3 HTTP响应信息	224
11.1.4 URL	226
11.1.5 HTTP流程	227
11.2 Web 服务器——Socket	228
11.3 Web 服务器——TcpListener	237
11.4 取得网页内容	239
附录A 面向对象与系统开发	241
附录B UML	245
附录C 控制流程	248

附录D 窗口用户界面类.....	253
附录E 线程.....	290
附录F Java与.NET的比较.....	297
附录G SSL.....	299
附录H 使用Windows API.....	303
附录I Microsoft Windows Networking API.....	305
附录J Microsoft RAS API.....	317
附录K Microsoft Win32 Internet API.....	325
附录L RFC.....	354
附录M 端口.....	357
参考文献.....	358

第 1 章 Microsoft.NET Framework

2002 年, Microsoft 推出了新一代的平台与开发工具——Microsoft.NET Framework 与 Visual Studio.NET。Microsoft.NET Framework, 简单地说, 就是 Microsoft 为 XML 网络服务所提供的平台。Microsoft.NET 构建开放式的结构, NET 平台的目标在于简化 Web 开发过程。

Microsoft.NET Framework 设计的目的在于能够与互联网相关标准完整结合, 如通过 XML 网络服务, 可以轻易从网站上取得数据, 并且通过运用 XML, 使数据可以在不同的系统之间交换。

Microsoft.NET 包含以下重要技术 (图 1.1):

- Microsoft.NET Framework。
- Microsoft Visual Studio .NET 开发环境。
- Microsoft.NET Enterprise Servers。
- Building Block Services。

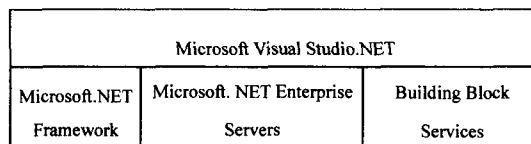


图 1.1 Microsoft .NET 包含的重要技术

1.1 Microsoft .NET Framework 概述

Microsoft.NET Framework 是一个程序语言中立 (Language Neutrality) 的执行环境, 包括以下技术 (图 1.2):

- Common Language Runtime (CLR)。
- Microsoft.NET Framework 类库。
- ADO.NET。
- ASP.NET。

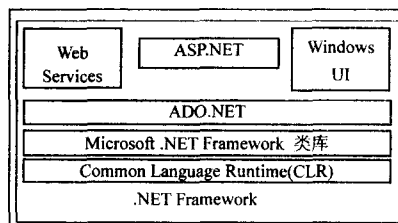


图 1.2 Microsoft.NET Framework 包括的技术

1.2 CLR

CLR 提供了一个跨语言 (Cross-Language) 的程序语言中立执行环境, 它可以使对象更容易跨语言 and 应用程序互动。不同语言所编写的对象可以彼此继承使用, 例如, 可以使用 C# 定义类, 而由 Visual Basic.NET 继承使用。

CLR 具有以下特点:

- 面向对象设计。
- 强大的类型安全 (Type Safety) 机制提供安全的执行环境。
- 自动内存管理。
- 支持多重语言, 特别是跨程序语言继承。
- 易于使用其他语言开发组件。
- 简化应用程序部署。

CLR 包括以下组件:

- Class Loader: 管理类的加载。
- Code Manager: 管理程序代码的执行。
- COM Marshaler: 提供 COM 的编组。
- Debug Engine: 提供应用程序调试环境。
- Exception Manager: 提供结构化的异常处理机制。
- Garbage Collector: 提供对象内存自动回收的管理机制。
- Microsoft Intermediate Language (MSIL): 将 MSIL 转换为 JIT (Just In Time) 机器码。
- Security Engine: 提供应用程序安全性验证环境。
- Thread: 提供多任务执行应用程序的类及界面。
- Type Checker: 验证类型安全性。

1.3 Microsoft.NET Framework 类库概述

Microsoft.NET Framework 类库为 Microsoft .NET Framework 的另一个重要特色, 类库是可再用 (Reusable) 类及类型的集合, 并与 CLR 紧密结合。

Microsoft.NET Framework 类库是完全以面向对象 (Object Oriented) 方式设计的。面向对象技术是继结构化技术之后, 系统开发上的另一新思维典范。面向对象的观念起源于仿真语言, 以对象 (Object) 模式来描述真实系统, 并将数据抽象化 (Data Abstraction)、封装 (Encapsulation)、继承 (Inheritance) 与多态性 (Polymorphism) 的观念融入于对象系统开发中。

面向对象包括如下方面:

- 对象。
- 类。

- 继承。
- 封装。
- 多态性。

1. 对象

对象是面向对象的基本单位，对象是一个具有状态（State）、行为（Behavior）与识别（Identity）的实体（Entity）或抽象化概念（Abstract Concept），且其行为会影响其状态。

对象包括如下方面：

- 名称。
- 属性。
- 操作或方法。

2. 类

类是具有相同结构及行为的对象所组成的集合，一个类是一种定义（Definition）、模板（Template）或模型（Mold），它是许多对象共同特征的一种描述。

类是对象经分类或抽象化后所得的结果。也就是将对象抽象化，剔除对象间的差异而只考虑其相同的性质后，将这些对象组成一个群体，则称为类。

3. 继承

继承是类之间的关系，在此关系中某类的数据结构与行为可供其关系中的类分享。

继承的特性可利用一般化（Generalization）与特殊化（Specialization）的原则，取相关子类（Subclass）的相同属性和操作，并将之归为一个父类（Superclass）来达成。

4. 封装

面向对象技术将数据及操作此数据的所有方法包装成一个对象，称为封装。

封装所形成的对象，其结构可分为两部分，一是定义对象外观行为的界面（Interface）部分；另一个则是存放抽象化结果以及如何达成外观行为的实现（Implementation）部分。

封装将对象的实现细节隐藏起来，只允许该对象所包含的操作修改其信息，称为信息隐藏（Information Hiding）。

使用对象时，只需知道对象提供何种操作，而不需要知道其内部信息或行为是如何表达或执行的。外界对象只能通过信息传递要求该对象提供服务，而无法亲自改变该对象的数据内容。

封装特性使面向对象的系统难于维护。

5. 多态性

多态性是指“多种形式”，简称多态。即利用相同的操作，以不同的方式处理不同类的数据。运用多态性特性，可允许不同的对象定义相同的操作名称，待程序执行时，根据动态绑定（Dynamic Binding）的方法判断参数的种类与类型，以决定运行的对象。

在面向对象上，Microsoft.NET Framework 类库包括如下方面：

- 继承：通过类来定义，以作为衍生其他类的基础。衍生类可继承延伸基础类的属性（Properties）、事件（Event）及方法，衍生类也可以新的方式覆盖所继承的方法。
- 覆写（Overriding）：覆写可让衍生出来的对象覆写从基础对象所衍生而来的特性，被覆写的方法具有相同的参数，但与基础对象的方法有不同的实现。
- 重载（Overloading）：重载是指利用相同的名字、不同的参数及不同的数据类型定义属性及方法。重载化程序的目的是希望以相同的界面处理不同的对象，以简化系统发展的复杂性并增加其弹性。
- 界面：界面说明类的属性与方法，但与类不同的是，界面并没有提供实现。
- 委派（Delegation）：委派是对象声明其他对象方法的行为，委派被形容为面向对象安全类型的功能，程序可利用委派指定事件处理的方法。
- 成员共享（Member Sharing）：成员共享是指类中所有成员共享的属性及字段。

Microsoft .NET Framework 类库提供的功能包括：

- 对象管理。
- 绘图。
- 网络。
- 数据存取。
- 安全性。
- 加密。
- XML。
- 网络服务。

Microsoft.NET 提供如表 1.1 所示的名字空间。

表 1.1 Microsoft.NET 提供的名字空间

名字空间	说明
System.codeDom	包含可用于表示源程序代码文档的元素与结构的类
System.collections	定义对象集合的类，如列举、行列、数组、哈希表和字典等
System.ComponentModel	提供用于实现组件和控件的运行时及设计阶段行为的类，包括实现属性、类型转换、数据源绑定及授权组件的基类与界面
System.Configuration	存取.NET Framework 的功能设置
System.Data	基本上由 ADO.NET 构建的类组成，用来管理数据存取
System.Diagnostics	处理系统过程、事件日志，以及与性能计数器进行交互
System.DirectoryServices	提供存取 Active Directory 的类，此类使用 Active Directory Services Interface (ADSI) 技术，使管理员能够相对轻松地查找和管理网络资源，不论网络规模有多大
System.Drawing	提供对 GDI+绘图功能的访问
System.Globalization	支持区域功能，如语言、国家/地区、日期、货币、数字格式及字符串的排序等
System.IO	提供对数据流和文件进行同步和非同步存取与管理，如文件、存储器等
System.Management	提供信息和事件管理的类
System.Messaging	支持 Microsoft Message Queue (MSMQ) 的管理与非同步信息传送和接收

续表

名 字 空 间	说 明
System.Net	支持通信协议网络应用
System.Reflection	提供已加载类型、方法和字段的托管视图的类，并具有动态创建和调用类型的能力
System.Resources	支持资源管理
System.Runtime.CompilerServices	支持 Runtime 编译器
System.Runtime.InteropServices	提供 CLR 与 COM、Unmanaged Code 的互通服务管理
System.Runtime.Remoting	支持建立分散式应用程序
System.Runtime.Serialization	支持对象序列化
System.Security	支持 CLR 的安全性原则及权限
System.Security.Cryptography	提供数据加密编译服务，如数据编码、解码、哈希处理、验证、数字签名等
System.Serviceprocess	支持 Windows 服务应用程序
System.Text	支持有关字节、字符串处理的类
System.Threading	支持多线程程序
System.Timers	提供定时间隔引发事件的 Timer 组件
System.Web	支持 Web 应用程序开发，如 ASP.NET、WebForm
System.Web.Services	支持 SOAP 网络服务应用程序开发
System.Windows.Forms	支持创建基于 Windows 的应用程序的类，这些应用程序可以充分利用 Microsoft Windows 操作系统中的丰富用户界面功能
System.Xml	处理 XML

1.4 安装 Visual Studio.NET 2003

本节将介绍如何安装 Visual Studio.NET 2003。

1. 系统需求

① 操作系统：Microsoft Windows 2000、Windows XP 或 Windows.NET Server 2003（表 1.2）。

表 1.2 支持 Microsoft Visual Studio.NET 的操作系统

操 作 系 统	至 少	建 议
Windows 2000 Professional	96 MB	128 MB
Windows 2000 Server	192 MB	256 MB
Windows XP Home Edition	96 MB	160 MB
Windows XP Professional Edition	192 MB	256 MB
Windows.NET Server 2003	192 MB	256 MB

② 处理器：Pentium II 450 MHz（建议 Pentium III 600 MHz）。

③ 内存：128 MB（建议 256 MB）。

④ 硬件空间（表 1.3）。

表 1.3 硬件空间

Visual Studio.NET 版本	可用硬件空间	系统硬件空间
Visual Studio.NET Enterprise Architect	3.8 GB	750 MB
Visual Studio.NET Enterprise Developer	3.7 GB	750 MB
Visual Studio.NET Professional	3.6 GB	750 MB
Visual Basic.NET	2.3 GB	750 MB
Visual C#.NET	2.0 GB	750 MB
Visual C++.NET	2.0 GB	750 MB

⑤ 硬盘空间足够大。

⑥ CD-ROM 或 DVD-ROM 光驱。

⑦ 显示器：800×600，256 色（建议全彩 16 位）。

⑧ 为确保执行效率，.NET Framework 需配备如表 1.4 所示的操作系统。

表 1.4 .NET Framework 需配备的操作系统

项 目	操 作 系 统
服务器端	Windows 2000 Professional with Service Pack 2.0+
	Windows 2000 Server with Service Pack 2.0+
	Windows 2000 Advanced Server with Service Pack 2.0+
	Windows XP Professional
用户端	Windows 2000 Professional
	Windows 2000 Server
	Windows 2000 Advanced Server
	Windows XP Home Edition
	Windows XP Professional

2. 安装 Visual Studio.NET 2003

① 放入光盘后，Visual Studio.NET 安装程序会自动检测并建议安装 Visual Studio.NET 所需要的 Windows 组件，如图 1.3 所示。

② 选择安装程序 1——Visual Studio.NET Prerequisites，并放入 Visual Studio.NET 的 Prerequisites 光盘（图 1.4）。

③ Visual Studio.NET Prerequisites 会自动检查系统所需安装和更新的组件，如图 1.5 所示。根据系统的不同会有所差异，大致有以下几种。

- Windows 2000 Service Pack 3。
- Microsoft FrontPage 2000 Web Extensions Client。
- Setup Runtime Files。
- Microsoft Internet Explorer 6。

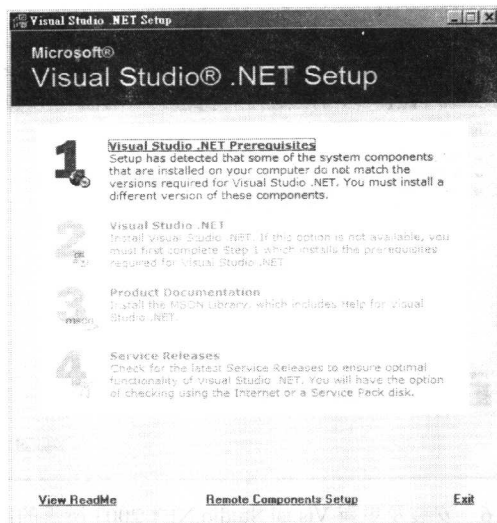


图 1.3 安装 Visual Studio.NET

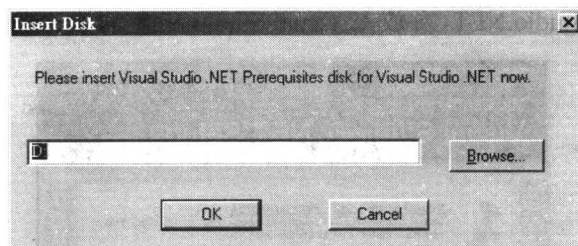


图 1.4 Insert Disk 对话框

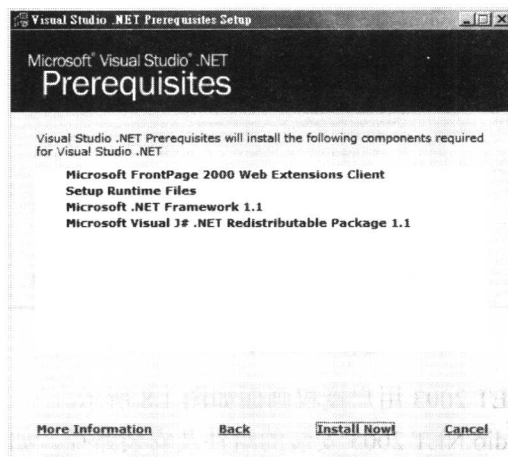


图 1.5 Visual Studio.NET Prerequisites 自动检查系统所需安装和更新的组件

- Microsoft Data Access Components 2.7。
 - Microsoft.NET Framework 1.1。
 - Microsoft Visual J# .NET Redistributable Package 1.1。
- ④ 安装及更新 Visual Studio.NET 2003 所需的组件如图 1.6 所示。

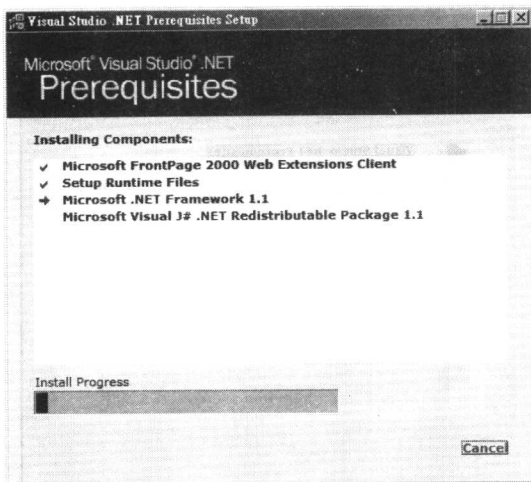


图 1.6 安装及更新 Visual Studio.NET 2003 所需的组件

⑤ 安装及更新 Visual Studio.NET 所需组件后，需要重新启动计算机。接着选择安装程序 2——Visual Studio.NET，并放入 Visual Studio.NET 的第一张光盘，如图 1.7 所示。

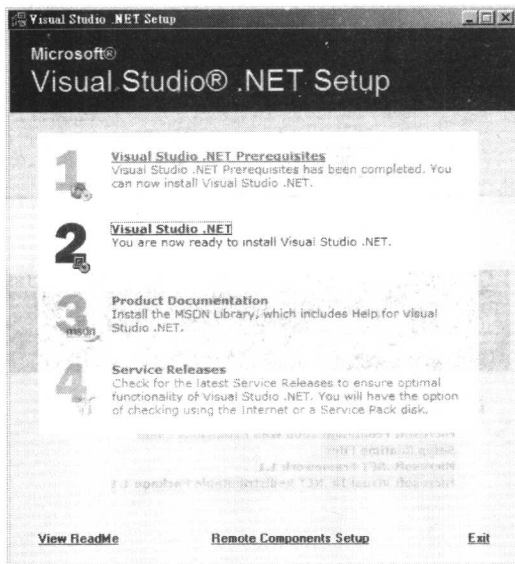


图 1.7 选择安装程序 Visual Studio.NET

- ⑥ Visual Studio.NET 2003 用户授权画面如图 1.8 所示。
- ⑦ 选择 Visual Studio.NET 2003 安装的组件及安装路径，如图 1.9 所示。
- ⑧ 安装过程如图 1.10 所示。
- ⑨ 安装完成之后，可继续安装 MSDN Library for Visual Studio.NET 2003，如图 1.11 所示。

Visual Studio.NET 2003 已正式将 Microsoft Visual J#.NET 列入其中，以支持 Microsoft 版的 Java 程序开发，并附有 Microsoft Visual J# .NET Redistributable Package 1.1 及 Java Language Conversion Assistant(JLCA)，以协助将 Sun Micro 版的 Java 程序转换至 Microsoft

版的 VisualJ#或 C#程序。

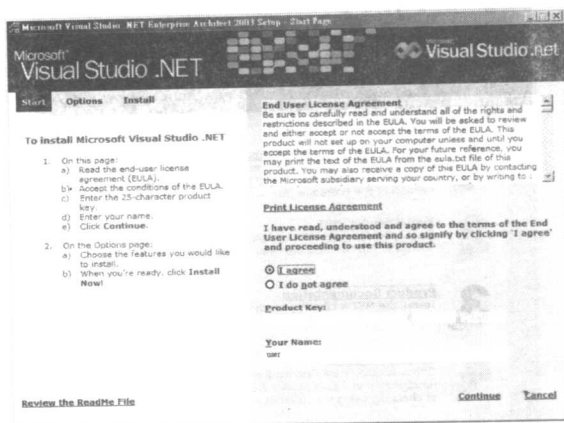


图 1.8 Visual Studio.NET 2003 用户授权画面

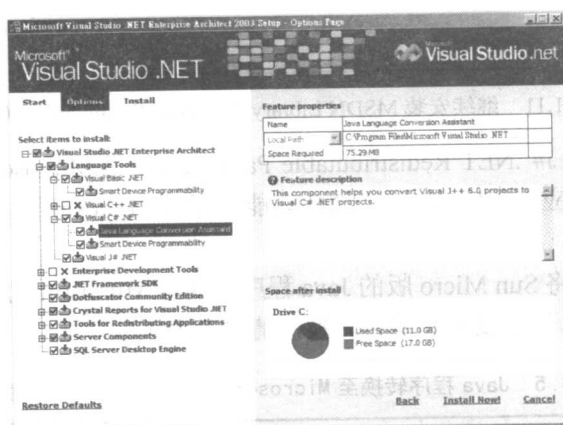


图 1.9 选择 Visual Studio.NET 2003 安装的组件及安装路径

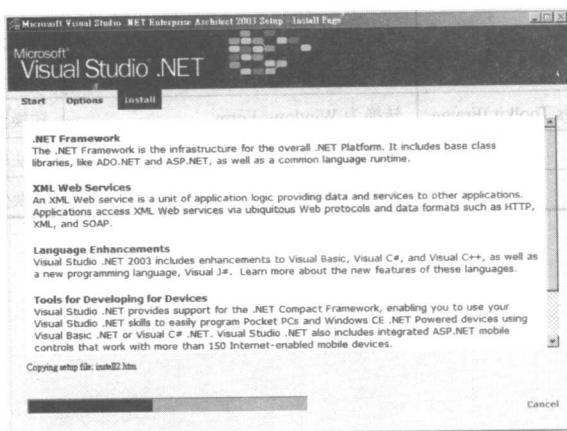


图 1.10 安装过程

Microsoft Visual J#是协助原 Java 开发人员在 Microsoft.NET Framework 上建立 Microsoft 版的 Java 应用程序的 IDE 开发工具。Microsoft Visual J#开发的程序只能在 Microsoft.NET