

高等院校计算机教育系列教材

Visual C++ .NET 程序设计教程

王恩德 编著

01



清华大学出版社

高等院校计算机教育系列教材

Visual C++ .NET 程序设计教程

王恩德 编著

清华大学出版社

北 京

内 容 简 介

本书内容由3部分组成:第1部分为基础部分(第1章~第5章),主要包括.NET技术概述、C++语言简介、面向对象的程序设计基础、Visual C++ .NET开发环境和Windows应用程序设计。第2部分为用户交互界面设计(第6章~第10章),主要介绍消息处理、对话框、菜单、工具栏与状态栏和图形设备接口与绘图。第3部分为开发新技术与应用(第11章~第14章),主要介绍文档与视图、组件编程、数据库编程和Internet编程。每章后面都有小结和习题。

作者在计算机专业教学和应用软件开发方面有丰富的经验,并对处理具体应用问题也有独到之处。全书讲解了Visual C++ .NET语言编程所涉及的许多领域中的应用,对于常见主题的讲解比较全面细致。

本书适合大学三年级或者三年级以上的学生使用,也可作为具有同等知识水平和能力的读者的自学用书。

版权所有,翻印必究。举报电话:010-62782989 13901104297 13801310933

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Visual C++ .NET 程序设计教程/王恩德编著.—北京:清华大学出版社,2004
(高等院校计算机教育系列教材)

ISBN 7-302-08886-1

I.V… II.王… III.C语言—程序设计—高等学校—教材 IV.TP312

中国版本图书馆CIP数据核字(2003)第059228号

出 版 者:清华大学出版社 地 址:北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编:100084

社 总 机:010-62770175 客 户 服 务:010-62776969

责任编辑:张 莉

封面设计:陈刘源

印 装 者:北京国马印刷厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:23.25 字数:450千字

版 次:2004年7月第1版 2004年7月第1次印刷

书 号:ISBN 7-302-08886-1/TP·6294

印 数:1~5000

定 价:28.00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)62770175-3103或(010)62795704

丛 书 序

人类进入新千年时, 计算机科学已经具有一块极其活跃的、崇尚发展与创新的领地, 并成为我们这一时代决定性的先驱技术。计算机是现代文化构成不可或缺的组成部分, 是现代科学技术发展的先导, 并且是世界经济巨量增长的根本引擎。同时, 计算机技术的发展日新月异, 它的快速发展对计算机科学与工程的教育产生了深刻的影响。因此, 计算机教育在世界各国备受重视, 计算机知识与能力已成为 21 世纪人才素质的基本要素之一。

伴随着计算机新技术的不断涌现, 已有技术几年内即变得陈旧。与此同时, 计算机教育也被其所在的专业、文化和社会范围的变化影响着。计算机学科已变得更宽广、内容更丰富, 其应用领域不断飞速增长。因此, 无论在教学体系、教学内容还是教学方法、教学手段上都必须进行深化改革, 与时俱进。CC2001 的出现, 无疑是对计算机学科课程体系一个崭新的完整的引导。我们工科院校无论计算机专业还是非计算机专业的计算机科学与工程的教育应该紧密有机地与工科学生的培养目标相结合, 注重知识、能力、素质教育三方面的综合教育, 加强计算机系统的认知、分析、设计和应用能力, 算法设计与分析能力, 程序设计能力以及计算思维能力等方面的培养。

原化工部部属高校计算机教育协作组结合工程教育的特点, 大力开展计算机教育协作与交流, 十年来开展了务实的、全方位的、卓有成效的教学研讨及教研观摩等活动, 极大地促进了交流并推动了各校计算机教育的发展。同时, 协作组不断地扩展, 吸收了许多其他领域的高校参加, 共同为我国计算机高等教育事业的发展与完善进行广泛的交流探索。

目前参加这个协作组的主要高校有:

清华大学	南京大学	天津工业大学	中国农业大学
北京化工大学	南京工业大学	青岛科技大学	郑州大学
武汉化工学院	沈阳化工学院	南京师范大学	华南理工大学
河北行政学院	南京工程学院	淮海工学院	北京石油化工学院
北京工业大学	沈阳化工学院	江苏石油化工学院	大连理工大学
吉林大学			

在清华大学出版社的大力支持下, 本协作组 2001 年年会决定组织出版一套最新的计算机系列教材, 第一期出版 14 部有关程序设计与软件应用方面的教材。它们是: 《Visual C++ .NET 程序设计教程》、《计算机导论》、《C 语言程序设计》、《Visual Basic 6.0 程序设计》、《Java 程序设计》、《面向对象程序设计——Visual C++》、《SQL Server 数据库原理及应用教程》、《C#编程及应用程序开发教程》、《组网技术与配置》、《数据库

理论及应用基础》、《现代语音技术基础与应用》、《计算机图形学基础教程》、《Win32 汇编语言实用教程》、《C/C++程序设计教程》和《数据结构教程》等。

本系列教材依据 CC2001 框架，精心策划、准确定位，概念清晰，例题丰富，深入浅出，内容翔实，体系合理，重点突出，是一套面向高等学校计算机和非计算机专业学生的计算机基础与应用系列教材，也可供从事计算机应用和开发的各类人员学习使用。

本系列教材源于十几所全国重点大学和普通高等院校计算机教育的教学改革与实践，凝聚了工作在教学第一线的任课教师的教学经验与研究成果。我们期望本系列教材的出版，并在教学实践中不断完善与更新，为我国高校计算机教育事业做出新的贡献。

编委会

2004年6月

编委会名单

主编：朱群雄

编委：张莉	闵华清	王晓峰
邵定宏	刘川来	彭四伟
刘斌	刘新民	张彦锋
吕纪国	刘焯	王相林
蔡莲红	孙正兴	冉林仓
彭波	王恩德	

前 言

作者多年来一直在高校从事计算机科学技术的相关教学与研究工作的，在长期的实践中深刻体会到，熟练掌握一门计算机语言也就是掌握了从事计算机技术应用研究的必备工具。这就如同各种门类的艺术家一样，通过他们所掌握的艺术工具，来描绘祖国的锦绣河山，歌唱时代的主旋律，为人们创作出无数的优秀艺术作品。确实，编程也是一门艺术。每当完成一项程序设计，就是一件杰出作品的诞生，往往为它的成功而享受着无穷的乐趣，因为它面对的是知识的海洋，它是计算机技术和逻辑思维的结合，凝聚了自己劳动的汗水和心血。

古语说：“工欲善其事，必先利其器”。选择好的编程工具，可达事半功倍之成效。随着计算机技术的发展，各种编程工具应有尽有。我和学生交谈时常说他们现在是多么的幸福，他们对这句话还不很理解，但是我深有体会，因为我曾亲身经历过手编程序和汇编程序的年代。根据我的经验，Visual C++ .NET 是编程工具的首选。

从整个 IT 产业的软件市场来讲，如果要开发高性能的 Windows 或 Web 应用程序，Microsoft Visual C++ 是一种开发效率最高的工具。几乎所有世界级的软件都是使用 Microsoft Visual C++ 开发系统生成的。在 2002 年年初，微软公司又推出了 Visual C++ 的最新版本——Visual C++ .NET，它继承了以往 Visual C++ 各版本的优点，具有强大的底层编程功能。作为一种编程语言，它完全继承了 C++ 语言面向对象的特性。同时作为一种编程工具，它又以其图形化的编程方式将 C++ 支持面向对象技术的特性完美地体现出来，使得开发软件的这一原本枯燥、难以理解的工作变得轻松快捷。此外，新版本中还增加了许多新的特性，使开发的能力更强、开发的效率更高，深受业内人士青睐。

为配合教学改革，更新教学内容，我为计算机专业的本科生开设过“面向对象的程序设计”课程，也为研究生开设过“应用软件开发新技术”课程，本书就是在相关教学的基础上编写而成的，内容由以下 3 部分组成：第 1 部分为基础部分(第 1 章~第 5 章)，主要包括 .NET 技术概述、C++ 语言简介、面向对象的程序设计基础、Visual C++ .NET 的开发环境和 Windows 应用程序设计。第 2 部分为用户交互界面设计(第 6 章~第 10 章)，主要介绍消息处理、对话框、菜单、工具栏与状态栏和图形设备接口与绘图。第 3 部分为开发新技术与应用(第 11 章~第 14 章)，主要介绍文档与视图、组件编程、数据库编程和 Internet 编程。

从以上介绍可知，本书从本质上讲是一本 MFC(Microsoft Foundation Class)书籍，而不是专论 .NET 的图书。为使读者对 .NET 有所了解，在第 1 章中就编排了相关内容。在涉猎微软公司的 MFC 后，作者为其包罗万象的内容所惊叹，它几乎覆盖了软件系统的各个领域，在一本书中根本无法囊括。但作为一本基础教材，本书力图将 Visual C++ .NET 的基本概念、基本方法和系统知识介绍清楚，同时将 Visual C++ .NET 的精彩之处及反映计算机发展潮流的新技术奉献给读者。

另外在书中的每个章节中，配合学习内容都有具体应用的例程加以说明。在介绍例程

之前，有该例程的设计思路；在讲解例程时或在介绍例程之后，也有例程的详细分析说明。这些例程比较短小精悍，针对性强，便于读者上机实际演练，也可供读者在实际应用中参考。

本书前两部分内容可作为与计算机有关的大学本科高年级学生的专业基础课，课时约为 54 学时；后一部分可作为选修课，课时约为 36 学时。

本书的编写，还得力于我的妻子和儿子所从事的专业与计算机有关。我的妻子高铁梅教授为本书编写提供了全力支持和无私奉献。她为编写大纲、资料搜集以及内容编排做了大量的工作，并认真审阅了全部内容。我儿子王斌是计算机专业的研究生，他不仅是本书的策划者，而且书中很多实例也经他调试，特别是第 3 部分的内容，他提出了许多修改意见，如果说是同他的合著，也不足为过。

本书成书之前曾作了几次修改和补充，但总觉得还有很多尚不完美的地方，留下了一些遗憾，衷心希望读者提出宝贵意见和批评。

作者

2004 年 5 月

目 录

第 1 章 .NET 与 Visual Studio .NET1	
1.1 .NET 介绍.....1	
1.1.1 .NET 概述.....1	
1.1.2 .NET 框架.....2	
1.2 Visual Studio .NET 产品.....4	
1.2.1 Visual Studio .NET 新特性.....4	
1.2.2 Visual Studio .NET 的安装.....5	
1.3 Visual C++ .NET 的开发范围.....8	
1.3.1 托管(managed)应用.....8	
1.3.2 非托管(unmanaged)应用.....9	
1.4 小结.....10	
1.5 习题.....10	
第 2 章 C++语言基础11	
2.1 最简单的 C++程序.....11	
2.2 C++基本数据类型.....12	
2.2.1 C++基本数据类型.....12	
2.2.2 类型安全性.....12	
2.3 常量和变量.....13	
2.3.1 常量.....13	
2.3.2 变量.....15	
2.4 表达式.....17	
2.4.1 数值表达式.....17	
2.4.2 逻辑表达式.....19	
2.4.3 运算符的运算次序.....19	
2.5 基本语句.....20	
2.5.1 空语句.....20	
2.5.2 块语句.....21	
2.5.3 条件语句.....21	
2.5.4 分支语句.....22	
2.5.5 循环语句.....23	
2.5.6 转移语句.....24	
2.5.7 输入/输出.....25	
2.5.8 预处理指令.....26	
2.6 函数.....27	
2.6.1 C++标准库.....28	
2.6.2 自定义函数.....28	
2.6.3 函数重载.....30	
2.6.4 函数调用.....31	
2.6.5 带默认参数的函数.....31	
2.6.6 函数和变量的作用域.....32	
2.6.7 函数和变量的存储类别.....33	
2.6.8 变量的引用类型.....35	
2.7 指针.....36	
2.7.1 指针的概念.....36	
2.7.2 指针的基本操作.....38	
2.8 扩展数据类型.....40	
2.8.1 数组.....40	
2.8.2 结构体.....42	
2.8.3 共用体.....43	
2.8.4 枚举类型.....44	
2.9 动态分配/撤消内存的运算符 new 和 delete.....44	
2.9.1 new 运算符.....44	
2.9.2 delete 运算符.....45	
2.10 小结.....46	
2.11 习题.....46	
第 3 章 面向对象程序设计基础50	
3.1 面向对象的基本概念.....50	
3.1.1 面向过程和面向对象的比较.....50	
3.1.2 面向对象的基本概念.....52	
3.2 类和对象在 C++中的定义.....56	
3.2.1 类的定义.....56	
3.2.2 对象的定义.....58	

3.2.3	对象指针	58	4.3.5	“生成”菜单	92
3.2.4	this 指针	60	4.3.6	“调试”菜单	93
3.3	构造函数与析构函数	60	4.3.7	“工具”菜单	93
3.3.1	构造函数	60	4.3.8	“窗口”菜单	94
3.3.2	析构函数	61	4.3.9	“帮助”菜单	95
3.3.3	拷贝构造函数	62	4.4	工具栏	95
3.4	继承和派生类	63	4.5	窗口	96
3.4.1	派生类的声明	63	4.5.1	窗口分类	96
3.4.2	保护成员	64	4.5.2	窗口管理	96
3.4.3	派生类的构造函数和 取析构函数	65	4.5.3	“解决方案资源管理器” 窗口	97
3.4.4	多重继承	66	4.5.4	“类视图”窗口	98
3.5	多态性	67	4.5.5	“资源视图”窗口	98
3.5.1	编译时的多态性和 运行时的多态性	67	4.5.6	“属性”窗口	99
3.5.2	函数重载	67	4.5.7	代码和文本编辑器窗口	99
3.5.3	虚函数	68	4.5.8	服务器资源管理器	100
3.6	友元	71	4.6	小结	101
3.6.1	友元函数	71	4.7	习题	101
3.6.2	友元成员	71	第 5 章 Windows 应用程序设计	102	
3.6.3	友元类	72	5.1	Windows 操作系统及其应用程序	102
3.7	模板	72	5.1.1	Windows 操作系统的特点	102
3.7.1	函数模板	73	5.1.2	Windows 应用程序的特点	102
3.7.2	类模板	73	5.2	MFC 基础	103
3.8	小结	75	5.2.1	MFC 的概念	103
3.9	习题	75	5.2.2	MFC 中类的组织结构	104
第 4 章 Visual C++ .NET 的开发环境	80		5.3	MFC 中基本类简介	104
4.1	Visual C++ .NET 的新特点	80	5.3.1	CObject 类	104
4.2	Visual Studio .NET 的 启动和退出	82	5.3.2	应用程序体系结构类	105
4.2.1	解决方案和项目	82	5.3.3	可视对象类	106
4.2.2	Visual Studio .NET 的启动	82	5.3.4	通用类	108
4.2.3	退出 Visual Studio .NET	86	5.3.5	OLE 类	109
4.3	菜单栏	86	5.3.6	ODBC 数据库类	109
4.3.1	“文件”菜单	86	5.4	快速设计 Windows 应用程序	110
4.3.2	“编辑”菜单	87	5.4.1	创建 MFC 应用 程序框架结构	110
4.3.3	“视图”菜单	89	5.4.2	加工和运行 MFC 应用程序	118
4.3.4	“项目”菜单	90			

5.4.3 MFC 中的应用	7.2.3 对话框的实现.....	154
程序框架类.....	7.2.4 标签式对话框设计.....	156
120	7.3 通用对话框.....	158
5.4.4 MFC 应用程序的	7.3.1 颜色对话框.....	158
控制流程.....	7.3.2 文件对话框.....	159
122	7.3.3 查找与替换对话框.....	161
5.5 小结.....	7.3.4 字体对话框.....	162
124	7.3.5 打印与打印设置对话框.....	163
5.6 习题.....	7.4 信息框.....	165
124	7.4.1 CWnd::MessageBox().....	165
第 6 章 消息处理	7.4.2 AfxMessageBox().....	167
125	7.5 小结.....	167
6.1 Windows 消息系统.....	7.6 习题.....	168
125	第 8 章 菜单设计	169
6.1.1 Windows 消息.....	8.1 静态菜单设计.....	169
125	8.1.1 创建菜单.....	169
6.1.2 消息分类.....	8.1.2 编写菜单处理函数.....	171
127	8.1.3 定义菜单项的加速键.....	173
6.1.3 消息的传递.....	8.2 动态菜单设计.....	174
128	8.2.1 有关动态菜单的函数.....	175
6.1.4 消息处理函数.....	8.2.2 动态菜单的实例.....	177
129	8.3 快捷菜单设计.....	182
6.1.5 消息系统的工作原理.....	8.4 小结.....	183
129	8.5 习题.....	183
6.2 消息映射.....	第 9 章 工具栏与状态栏设计	184
131	9.1 工具栏的创建和使用.....	184
6.2.1 消息映射和消息映射表.....	9.1.1 向导自动创建的工具栏.....	185
131	9.1.2 添加新的工具栏.....	187
6.2.2 消息映射项和消息	9.1.3 创建 ReBar.....	194
处理方法.....	9.1.4 对话框的创建和使用.....	195
132	9.2 状态栏的创建和使用.....	197
6.2.3 添加消息映射项.....	9.2.1 向导自动创建的状态栏.....	197
134	9.2.2 自定义状态栏.....	199
6.2.4 添加自定义消息.....	9.3 小结.....	201
134	9.4 习题.....	202
6.2.5 SendMessage()和	第 10 章 图形设备接口与绘图应用	203
PostMessage().....	10.1 设备环境和设备环境类.....	203
135		
6.3 程序实例.....		
136		
6.3.1 窗口消息处理实例.....		
136		
6.3.2 命令消息处理实例.....		
138		
6.3.3 处理控件通知消息实例.....		
139		
6.4 小结.....		
141		
6.5 习题.....		
141		
第 7 章 对话框设计		
142		
7.1 对话框基础知识.....		
142		
7.1.1 对话框的组成.....		
142		
7.1.2 对话框的种类.....		
143		
7.1.3 对话框的运行机制.....		
143		
7.1.4 对话框数据交换与验证.....		
145		
7.1.5 访问控件的方法.....		
146		
7.2 自定义对话框设计.....		
147		
7.2.1 对话框模板的设计.....		
147		
7.2.2 对话框类的定义.....		
150		

10.1.1	设备环境	203	12.1.4	COM 实现	262
10.1.2	设备环境类	204	12.2	OLE 技术编程	265
10.2	GDI 绘图对象及其管理	206	12.2.1	OLE 的基本概念	266
10.2.1	建设备环境	206	12.2.2	OLE 的应用实例	269
10.2.2	GDI 对象的分类	207	12.3	ActiveX 技术编程	273
10.2.3	CPen 类	207	12.3.1	ActiveX 控件概述	273
10.2.4	CBrush 类	209	12.3.2	ActiveX 控件事件	274
10.2.5	CFont 类	211	12.3.3	ActiveX 控件方法	277
10.2.6	CRgn 类	213	12.3.4	ActiveX 控件属性	280
10.2.7	CPalette 类	216	12.3.5	ActiveX 控件的属性页	281
10.2.8	CBitmap 类	217	12.3.6	绘制 ActiveX 控件	284
10.3	映射模式	218	12.3.7	ActiveX 控件实例	285
10.4	CDC 中的绘图操作及其特点	219	12.4	小结	290
10.4.1	设置绘图参数	219	12.5	习题	291
10.4.2	绘图函数	220	第 13 章	数据库编程	292
10.5	绘制文本	222	13.1	ODBC 数据库应用编程	292
10.6	实例	224	13.1.1	ODBC 体系结构	292
10.7	小结	230	13.1.2	MFC 和 ODBC	294
10.8	习题	231	13.1.3	ODBC 应用程序实例	300
第 11 章	文档与视图应用	232	13.2	ADO 数据库应用编程	312
11.1	文档类与视图类	232	13.2.1	OLE DB 技术简介	312
11.1.1	文档类(CDocument)	232	13.2.2	ADO 技术概述	313
11.1.2	视图类(CView)	239	13.2.3	ADO 模型结构	313
11.1.3	文档与视图之间的关系	241	13.2.4	ADO 的控件编程	315
11.2	文档视图结构	242	13.2.5	ADO 的数据绑定 对话框编程	318
11.2.1	文档模板	243	13.3	小结	327
11.2.2	框架窗口	245	13.4	习题	327
11.2.3	结构中各对象 之间的关系	245	第 14 章	Web 编程	329
11.3	文档视图应用	247	14.1	创建 Web 浏览器	329
11.4	小结	256	14.1.1	Web 浏览器简介	329
11.5	习题	256	14.1.2	CHtmlView 类	330
第 12 章	组件编程	257	14.1.3	创建 Web 浏览器实例	330
12.1	COM 技术知识	257	14.2	开发 Web 应用程序	333
12.1.1	COM 的发展历史	257	14.2.1	开发技术综述	333
12.1.2	COM 技术基础	259	14.2.2	ATL Server 结构	334
12.1.3	COM 对象的特点	261	14.2.3	开发环境支持	335

14.2.4 Web 应用程序示例.....	336	14.3.3 创建 Web 服务.....	346
14.3 Web 服务开发.....	342	14.3.4 访问 Web 服务.....	350
14.3.1 Web 服务概述.....	342	14.4 小结.....	353
14.3.2 Web 服务的优势.....	345	14.4 习题.....	354

第 1 章 .NET 与 Visual Studio .NET

Visual C++ .NET 是微软公司于 2002 年度推出的 Visual Studio .NET 中以 C++ 语言为基础的可视化集成开发工具，它是采用了 .NET 技术的新一代软件开发语言。我们所介绍的内容中涉及到 .NET 的概念，这是目前 IT 业界较为前沿的理念，特别是在基于 Internet 的分布式系统这一领域中，人们都听说、感受过 .NET 及它对应用程序开发所产生的革命性影响。由于 .NET 的题目相当广泛，因此本章只能对 .NET 技术和 .NET 框架进行简要介绍，然后在此基础上介绍 Visual Studio .NET 产品的有关内容。

1.1 .NET 介绍

下面我们先来介绍一下 .NET 的基本知识。

1.1.1 .NET 概述

什么是 .NET？这是一个很复杂的问题，很难通过几句话讲清楚，但我们可以通过微软所公布的 .NET 战略目标来理解它。

Windows 操作系统从诞生之初一直发展到今天，历经若干个版本的升级，从而不断带动着应用软件开发模式的更新与变革。API(应用程序编程接口)和 COM(组件对象模型)开发模式是当今 Windows 平台上的主流开发模式。这两种模式从诞生起发展至今，经历了长时间实际应用的考验，已日趋成熟，当今每个在 Windows 平台上进行软件开发的人员在实际工作中都会直接或间接地与它们打交道。既然微软已拥有了如此成熟的软件开发模式，为什么还要提出全新的 .NET 技术？这是一个值得思考的问题。

今天，Internet 发展的浪潮冲击着社会生活的方方面面，不难预见，人们对因特网的依赖将与日俱增。人们可以利用因特网进行购物、安排旅行、查看产品信息和彼此沟通，也可以利用因特网足不出户地完成个人商务上的联系。今天，因特网上已有的技术保障了上述需求的成功实现，但是，对于人们更为高级的要求，现有的技术还不能很好甚至根本无法实现。比如，在几分钟之内，通过网络为远道而来的亲戚在离居住地点很近的一家旅馆订一个房间；或者在即将到来的假期里，通过网络策划旅游行程并对这次旅行费用进行初步结算等。

因特网尚且不能实现人们愿望的主要原因有两个：一个原因是在因特网上还没有一个提供各种服务信息的统一标准。所有网站各自为营，使用自己的方式来描述他们的产品和服务，从而采用了不同的、私有的方式存储数据和发布数据(包括查询数据的方式)。如果要使用统一标准来描述在网上流通的数据，就需要有一致的表示数据类型的方式，同时为

了查询所有数据，至少在逻辑上需要将这些数据联系在一起。另一个原因是应用程序的开发还显得太复杂，编写提供这种统一服务的代码和框架将是一项繁重的工作。

出于以上各种原因，微软公司提出的战略目标是要通过 .NET 搭建第三代因特网平台，同时建立这个平台上的运营规则。这个网络平台将解决网站之间的协作，从而最大限度地获取信息。在 .NET 平台上，不同网站之间通过相关的协定联系在一起，网站之间可以自动交流、协同工作，从而提供最全面的服务。这种服务是使因特网成为 IT 产业发展和世界经济发展促进因素的必然方向，.NET 的核心目标就是对现有的计算模式和软件开发方式进行有针对性的调整，使之符合第三代因特网运营的需要，这也就是为什么我们要把 .NET 理解为一种新技术、新理念的原因。

1.1.2 .NET 框架

为了实现这种新技术、新理念，微软为开发人员提供了一个崭新的、具有划时代意义的开发平台，使应用程序的开发更为简单。这个开发平台就是我们要讨论的内容——.NET 框架。

.NET 框架是一个可以构造、发布并运行 Web 服务的开发环境。从概念上来讲，.NET 框架平台代表了一种崭新的软件开发模式，它与 API 或 COM 一样，都是把系统服务以接口形式提供给开发人员的软件开发平台。与以往不同的是，.NET 框架能够更好地完成代码重用、资源配置、多语言集成开发和安全管理等任务，在安全性、易用性以及开发效率等方面都远远超过了以前的开发模式。.NET 框架主要有以下一些特点。

- 提供了面向对象的交叉语言类库

.NET 框架的两个主要组件是 CLR 和基类库(Base Class Library)，基类库为所有 .NET 编程语言提供一致、面向对象的类集，大概所有的开发人员对它都不会感到陌生，因为从使用标准 C 语言开发工具开始，我们就接触过 C 运行时库(C Runtime Library)。后来使用的工具越来越强大，接触过的运行时库的种类也越来越多。像微软基本类库(MFC)、ActiveX 模板库(ATL)、Windows 模板库(WTL)和 VB 运行时库(VB Runtime Library)等。开发人员都很喜欢使用这些库，因为这些库将复杂的、重复性的东西都准备好了，供需要它们的人来随时调用，从而简化了应用程序的开发，使开发人员将精力集中在大局构造上，节省花费在琐碎技术细节上的时间。不过在以前，采用不同种类的语言编写同一企业应用程序的现象非常普遍，而各种语言的类库各不相同，因此人们使用时不得不在不同种类语言的类库之间进行切换。基类库(BCL)在所有 .NET 语言中都是通用的(包括这些类的方法、属性和字段)，这样就大大降低了开发人员编程的复杂程度。

- 应用程序一次运行就会永远运行

Windows 应用程序开发人员都知道“DLL 灾难(DLL hell)”这个术语，这个经典问题会时常给我们制造麻烦。这是由于将一个新的应用程序安装到机器上的时候，如果新安装的组件覆盖了原来程序依赖的组件，就可能导致原有应用程序的崩溃。.NET 通过内部跟踪每个应用程序运行所需的文件(及其版本号)而消除了这一问题。这个跟踪文件是在应用程序首次安装时声明的。之后，新版本的跟踪文件

在安装时不会覆盖旧版本的跟踪文件，通过这种方式，.NET 也就解决了“DLL 灾难”问题。

- 一致的运行环境

运行时(Runtime)是.NET 框架平台的另一个核心内容，它是.NET 可执行程序唯一的运行环境，这也就意味着在.NET 的一个版本下编译连接并执行的应用程序将运行在.NET 的所有版本下。但以前有许多 API 的实例分别要求运行在不同的 Windows 平台下，这给应用程序的通用带来了麻烦，从这方面，我们也可以理解对 Windows 系统开发进行改革的重要性与必要性。

- 通用的类型系统

随同通用语言规范(CLS)一起，.NET 还定义了通用类型系统(CTS)，其中定义了.NET 所支持的全部类型，还定义了实例化和管理工作这些类型的方法。这意味着所有语言都可访问相同的类型，以便能够在不同语言编写的模块间传递数据，而不必在每种语言支持的不同格式间来回转换数据。另外，由于运行时具有类型上的控制权(相对于每种语言)，因此它可确保将正确的参数个数和类型等传递给指定方法。

- 集成的编程语言

前面提到，企业应用程序通常是用多种语言编写的。这会带来许多麻烦，因为每种语言都有它自己的能力和局限性。加上充分体验过用不同语言编写的应用程序之间共享数据的内在复杂性以及调试此类程序的痛苦经历，我们肯定会欣赏所有.NET 语言必须遵守一个统一的标准这一事实。这使得用不同语言所编写的代码之间集成变得比以前更为容易。

- 内存和资源的自动管理

编程需要花费很长时间来学习，并且在实践中有很高的技巧性。在实际工作中，处理诸如文件、内存、屏幕区域、网络连接和数据库等资源时，更需要有清醒的认识和丰富的处理经验。否则，哪怕一个小小的内存泄漏都有可能让软件在某个时候运行不稳定。新开发平台的运行时引擎可以自动跟踪应用程序的资源使用情况，并确保不会出现资源泄漏。

- 统一的异常处理

在 Windows 平台上进行编程，一件很麻烦的事情就是错误提示信息的风格不统一。某些函数提示错误的方式是返回标准 Win32 错误代码，而另一些函数则返回 HRESULTS 值(用于 COM)，还有一些函数则会引发异常。在新的开发平台上，所有的错误处理都通过异常来报告，异常可以使开发人员从代码中去除那些复杂的对可能错误的判断，使代码的编写、阅读以及维护工作大大简化。最后需要强调的是，异常处理也可以跨语言边界。

- 大大简化了应用程序发布

发布 Windows 应用程序通常是一件痛苦的事，应用程序越复杂，在客户机器上部署起来也就越困难。事实上，这一工作近年来已变得很困难，以至于有许多成熟的产品用它们本身内置的编程语言来安装 Windows 应用程序。使用.NET 将不必再追踪注册入口，并且不再需要为组件创建特别的文件夹以便确保组件不被其他

应用程序覆盖。在新平台上，安装软件意味着直接将应用程序拷贝到目标机器上即可，而卸载应用程序则更为简单，只要删掉它们就是了。

- 先进的安全机制

实现安全性的问题一直是一个非常复杂而重要的问题。传统操作系统的安全机制在实现访问控制方面主要是面向个人用户账户的，在新平台中，这种模式得到了改进，其核心内容是一种信任机制。如果应用是从物理介质(比如 CD-ROM)或其他可信任服务器上安装的，那么这个应用程序是可以信任的。但是，对于那些来历不明的诸如 Web 脚本、从因特网上下载的可执行文件或者可疑的 E-mail 附件，系统都会使用代码访问安全机制来对它们进行控制。

1.2 Visual Studio .NET 产品

Visual Studio .NET(也可以称为 Visual Studio 7.0)是微软开发套件的最新版本。Visual Studio .NET 在原有语言工具的基础上作了大量的创新和升级，将许多变革特性加入其中。在这些新特性中，最引人注目的是 Visual Studio .NET 构筑在 .NET 框架之上，因此，它拥有 .NET 框架的所有关键特性，并可快速开发出符合 .NET 框架的应用程序。

Visual Studio .NET 是快速建立企业级 Web 应用程序和高性能桌面应用程序的强大工具包，包括基于组件技术的开发工具，如 Visual C# .NET、Visual Basic .NET 和 Visual C++ .NET，以及用于简化开发小组设计、开发和发布解决方案的附加工具，如 ASP.NET 等。

时至今日，尽管大多数应用程序仍然是基于客户端/服务器模式，但企业对基于 Web 的以浏览器为前端的多层结构应用程序的需求增长是最快的。开发人员必须要将他们的注意力转移到分布式应用程序开发上来。Visual Studio .NET 正是为了迎合这种需要而设计开发的，它给开发人员提供了开发强大 Web 应用程序的高效、快速的手段。

1.2.1 Visual Studio .NET 新特性

Visual Studio .NET 的新特性如下：

- 增强了开发人员的工作效率

决定一个软件项目能否成功的重要因素之一就是开发效率。企业要求项目开发必须加快进度、缩短开发周期，以尽快将业务系统投入使用来适应商业竞争的需要。Visual Studio .NET 为专业开发人员提供了完整的解决方案，这里包括统一、高度集成化的开发环境，可以让开发人员在这个环境中，完成包括编写代码、错误处理、测试和发布等所有环节的工作，因此也就大大增加了开发人员的工作效率。

- 强大的 Windows 和 Web 开发工具

Visual Studio .NET 为开发人员提供最为详尽的工具来开发各种应用程序，可以使用一种相似的编程手段来开发各种用户接口，包括浏览器、移动设备以及 Windows 客户程序。

- 改善了开发语言的相互协作

过去, 开发人员在选择开发语言和开发平台上面临着比较痛苦的抉择。为了获得软件运行的高效率和可靠性, 开发人员往往选择功能强大的 C++ 语言, 但是因为 C++ 语言功能过于强大、特性繁多, 不仅很难掌握, 而且完全是一种非 RAD(快速应用程序开发)式的开发方式, 从而使企业级应用程序的开发变得难度很大。微软提供的 RAD 产品 Visual Basic 虽然学习难度小、易于掌握, 开发速度也很快, 但是却不能胜任关键任务的开发, 所以在过去, 开发人员常常不得不在同一项目中使用多种开发语言工具, 或者在一个开发团队中将多方面开发人员集成在一起, 这种现象非常普遍。由于前者对开发人员的素质要求较高, 而后者对人员之间的配合与协调又会产生许多不可预期的开销与冲突, 所以, 为了彻底解决语言工具协作的问题, 微软公司作了大量创造性工作: 一方面创建了一种现代的开发语言——C#, 另一方面, 微软还创建了能够实现多语言无缝集成的通用语言框架——.NET 框架。前者作为一种现代语言, 博采众家之长, 将 Visual Basic 的简便易用和 C++ 语言的强大特性集于一身。而后者作为一个公共的运行和语言支持平台, 使所有的语言(包含第三方开发语言)在这个平台上不再有任何地位上的差异。

- 简化了基于服务的开发

分布式应用程序开发中最为复杂的部分大概就是服务端组件的开发了。服务端组件封装了项目当中的业务逻辑、担负着处理客户端服务请求、平衡访问负载和连接数据库等重要任务, 在多层架构中扮演着核心角色。Visual Studio .NET 为了简化基于服务的开发, 为开发人员提供了大量 RAD 手段来简化服务端组件的开发。

- 简化了 Web 服务的使用和创建

在下一代的因特网中, Web 站点之间的合作机制将大幅度加强, 不像过去, Web 服务都被设计和开发成相互隔离的孤岛。Web 站点一方面可以通过对外公开可编程接口为其他站点和开发人员提供服务; 另一方面, 开发人员也能够把因特网上已存 Web 服务直接整合到自己的应用中来, 以快速完成自身的开发需求。Visual Studio .NET 将使得这两方面的需求同时得到满足。

1.2.2 Visual Studio .NET 的安装

Visual Studio .NET 的安装对计算机配置要求相对较高, 具体的配置说明如表 1.1 所示。

表 1.1 Visual Studio .NET 的安装对计算机配置

配置项目	配置要求
处理器	PII 450 MHz(建议: PIII 733 MHz)
内存	128 MB(建议: 256 MB)
硬盘空间	大约 3 GB(C:盘 500 MB)
显示卡	支持 800×600 显示模式(建议: 1024×768 显示模式)
光驱	必备
操作系统	Windows XP、Windows 2000、Windows NT 4