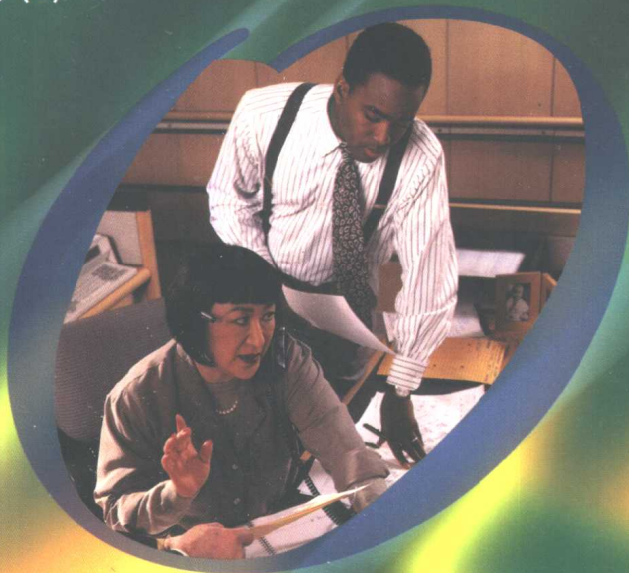


“十五”国家重点电子出版物规划项目 计算机知识普及和软件开发系列
体验办公新工具丛书 (6)

双色印刷



Master programming with
Visual C++.NET

深入
编程



北京希望电子出版社 总策划
杨竞锐 张连卫 王贵新 编写



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

“十五”国家重点电子出版物规划项目 计算机知识普及和软件开发系列
体验办公新工具丛书 (6)

双色印刷



Master programming with
Visual C++.

**深入
编程**



北京希望电子出版社 总策划
杨竟锐 张连卫 王贵新 编写



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

内 容 简 介

这是一本关于 Visual C++ .NET 编程技术书, 全书共由 10 章组成。主要内容包括 Visual Studio.NET 简介、Visual C++ .NET 的开发环境; 如何创建简单的 Visual C++ 应用程序(应用程序的类型、访问应用程序的代码以及编译和运行应用程序); 语言概述, 其中的语法和函数可供读者在编制程序的时候参阅; Windows 下几乎所有程序都会用到的对话框, 并就如何在 .NET 的编译环境下创建这些对话框进行了详细的讲解; .NET 提供的 Visual C++ 控件; 在 .NET 环境下的程序设计是本书的一个亮点, 介绍了编制 OpenGL 程序, 这对于游戏创作者和演示程序编制者等非常有用; 高级编程部分——多媒体编程。这里对数据库、网络等程序的编制加以详细的分析; 网络编程 Web Service 编程, 是 Visual Studio.NET 的新增部分, 它是 .NET 的重要组成部分。

本书的特点是由浅入深: 从编制程序所需要的语法和函数, 到深层次的数据库和网络编程。

内容丰富: 从 C# 编程、OpenGL 程序、到 Web Service 编程, 为从事多种类型的编程人员均提供了参考。

讲练结合: 范例与软件功能相结合, 易于读者理解、掌握、上手。

本书面向初、中级编程人员, 但对高级程序员也有一定的参考价值。

系 列 盘 书 : “十五”国家重点电子出版物规划项目 计算机知识普及和软件开发系列
体验办公新工具(6)

盘 书 名 : Master Programming with Visual C++ .NET 深入编程

文 本 著 者 : 杨竞锐 张连卫 王贵新

C D 制 作 者 : 希望多媒体开发中心

C D 测 试 者 : 希望多媒体测试部

责 任 编 辑 : 周 艳

出 版、发 行 者 : 北京希望电子出版社

地 址 : 北京中关村大街 26 号, 100080

网 址 : www.bhp.com.cn E-mail: lwm@hope.com.cn

电 话 : 010-62562329, 62541992, 62637101, 62637102, 62633308, 62633309

(图书发行) 010-62613322-215 (门市) 010-62547735 (编辑部)

经 销 : 各地新华书店、软件连锁店

排 版 : 希望图书输出中心 董淑红

C D 生 产 者 : 北京中新联光盘有限责任公司

文 本 印 刷 者 : 北京媛明印刷厂

开 本 / 规 格 : 787 毫米×1092 毫米 16 开本 30.375 印张 708.4 千字

版 次 / 印 次 : 2002 年 1 月第 1 版 2002 年 1 月第 1 次印刷

印 数 : 0001—5000 册

本 版 号 : ISBN 7-980008-45 -6 /TP · 45

定 价 : 46.00 元 (本版 CD)

说明: 凡我社产品如有残缺, 可执相关凭证与本社调换。

前 言

微软终于推出了.NET 计划最重要的一环——开发语言 Visual Studio.NET，程序员们终于松了一口气，微软的 Visual Studio.NET 没有让人觉得无法接受，而基于.NET 计划的网络开发功能在继承了全世界 300 万开发人员使用习惯的基础上，得到大大的加强，微软又一次用最简洁的方式将自己的.NET 理念通过 Visual Studio.NET 去影响全世界的微软开发人员。现在微软程序员可以用最简便的方式写出最好的程序，大多数人不必记得数量庞大的 API 函数，也可以放弃前辈程序员对系统底层的足够研究，甚至也不必对算法和数据结构有太多的了解，因为微软已经用 Visual Studio.NET 把所有基于.NET 计划的复杂理念变得如此简单。

这一次的变化是巨大的，作为全球最有影响的语言 C/C++，微软不愿意在 VC6 的基础上增加一些新特性，而是重新设计了该产品，使其用来设计像 Web 和企业级多层系统那样的分布式应用程序更加容易。Visual Studio.NET 有两个新的窗体包——Windows 窗体 (WindowsForms) 和 Web 窗体 (WebForms)。一个新版的 ADO 增加了访问离线数据源功能，改进了基本语言，去掉了以往版本遗留的关键字，增强了类型安全，并且应高级开发者需求揭示了函数方法实现的低级结构 (像 Delphi 一样)。另外还不能不提到微软 ASP 体系的发展，尽管以前版本的 ASP 开发十分方便，但还有许多不如人意的地方。以前的 ASP 只能使用 VBScript 或 JavaScript 等脚本语言来开发，现在我们可以使用更强大编程语言来开发 ASP.NET，比如 VB，当然还可以使用先锋语言 C#。而最令 Web 程序员开心的是，现在的 ASP.NET 使用 VS.NET 作为统一的调试器，同时已有了若干的小改动，ADO 可以直接与 DataGrid 元件相连，以前的 asp.dll，现在升级到统一的 .NET Framework 类库。ASP.NET 还封装了通用 HTML 接口的各种控件和针对 HTML 不足而设计的 Web 控件。

微软这样做的原因是由于网络的发展，它希望统一所有的网络开发语言。VB，VC 现在使用统一的继承开发环境，统一的调试器，统一的面向对象、异步、层次结构的可扩展类库 (即 .NET Framework 类库)。这一切基于公共语言运行时 (CLR)，程序员可以实现跨语言继承、纠错处理以及程序调试，而且 .NET 并不只是统一了微软本身的语言。据微软称，除了他自己的语言外还包括了其他 17 种语言都可以在 .NET 环境下运行。这意味着在 .NET 环境下网络上存在已久的 PERL 这样的语言也会被微软收编在自己的体系内。也意味着不但可以用不同的语言来开发组件控件，自己开发的类还可以继承其他语言编写的类，只要使用了交叉语言继承特性。调试器也是统一的，还意味着也能充分用网络上存在的各种语言代码，程序员们只需把需要的多语言代码串在一起，调试出多语言混合编写的应用程序就可以了，而不需要考虑它们能否在本地或远程机器上运行。总之，不管使用什么语言，微软 .NET Framework 都将提供一套丰富的统一的 Windows API 和关于 Internet 的 API。

MS/28/05

在 Visual Studio.NET 上微软充分利用了泊坞窗，所有的窗口都可以缩回到左右两边和下边的窗口条上，即使打开所有窗口也不会影响视线。这样就最大限度地利用了屏幕空间，在设计用户界面时视野更开阔、更直观。微软还为开发者增加了任务窗口、宏一览、连线帮助等等。任务窗口在升级旧版本工程时十分有用，它会教你一步一步完成任务列表上的升级服务，还有新建的工程需要做的初始化工作，调试出需要修改的地方，还有注释等都会在任务列表中显示并且鼠标双击就会定位到任务处，另外也可以定制任务，以“TODO”标记。连线帮助直接连接 Internet 上的技术资源，给开发者提供更及时和详尽的帮助。

飞速发展的信息时代，使得学习 Visual Studio.NET 已经刻不容缓，请从现在开始，从本书开始。

本书由邵谦谦总策划，杨竞锐、王贵新、李磊和张连卫共同执笔编写。张忠东、李晓裳、范智育、王宏生、李毅、王瑾、吴浩、李炎、刘伟、刘华刚、朱峰、赵晓燕、李晓、马苍、郝春容、韦勇、成美华、萧峰；李菊、张浩然、李欣、张浩、李想、朱大成、张诚华等同志在整理材料方面给予了作者很大的帮助。在这里，要特别感谢程梅女士在本书编著过程中给予的大力支持。

编者

目 录

第1章 Visual Studio.NET 简介..... 1	4.3.2 变量..... 42
1.1 Visual Studio.NET 的新功能..... 1	4.4 表达式..... 43
1.2 为什么要使用 Visual C++..... 4	4.4.1 赋值运算符和赋值表达式..... 44
1.3 Visual C++.NET 的运行条件..... 5	4.4.2 算术运算符和算术表达式..... 45
第2章 Visual C++.NET 的开发环境... 7	4.4.3 关系运算符和关系表达式..... 45
2.1 启动 Visual C++.NET..... 7	4.4.4 逻辑运算符和逻辑表达式..... 46
2.2 Visual Studio.NET 的开发环境..... 7	4.4.5 逗号运算符和逗号表达式..... 47
2.2.1 菜单栏..... 8	4.4.6 条件运算符和条件表达式..... 47
2.2.2 工具栏..... 17	4.4.7 位运算符和位运算表达式..... 47
2.2.3 Solution 浏览器..... 18	4.4.8 sizeof 运算符..... 50
2.2.4 类浏览器..... 21	4.4.9 运算符的优先级和结合顺序..... 51
2.2.5 资源浏览器..... 22	4.5 数据类型转换..... 52
2.2.6 属性窗口..... 22	4.5.1 数据类型的自动转换..... 52
2.3 Visual Studio.NET 的界面设置..... 24	4.5.2 数据类型的强制转换..... 53
第3章 创建简单的 Visual C++ 应用程序..... 26	4.6 数据的基本输入输出..... 54
3.1 了解应用程序的类型..... 26	4.7 语句..... 55
3.2 创建一个应用程序..... 27	4.7.1 语句概述..... 55
3.3 访问应用程序的代码..... 32	4.7.2 程序控制语句..... 56
3.3.1 使用 Class View (类视图) 访问代码..... 32	4.8 函数..... 71
3.3.2 使用 Object View (对象浏览器) 访问代码..... 33	4.8.1 概述..... 71
3.3.3 使用 Solution Explorer (解决方案 资源管理器) 访问代码..... 34	4.8.2 函数的定义、参数和返回值..... 72
3.4 编译和运行应用程序..... 34	4.8.3 函数的调用..... 75
3.4.1 编译应用程序..... 35	4.8.4 函数的嵌套调用和递归调用..... 77
3.4.2 运行应用程序..... 35	4.8.5 函数的引用调用..... 80
第4章 C/C++/C#语言概述..... 36	4.8.6 内联函数..... 83
4.1 字符集和词法记号..... 36	4.8.7 参数可以为缺省值的函数..... 84
4.1.1 字符集..... 36	4.8.8 函数重载..... 86
4.1.2 词法记号..... 36	4.8.9 函数模板..... 88
4.2 基本数据类型..... 38	4.9 数组..... 90
4.3 常量和变量..... 39	4.9.1 一维数组..... 90
4.3.1 常量..... 39	4.9.2 二维数组及多维数组..... 93
	4.9.3 字符数组..... 96
	4.9.4 数组作为函数参数..... 98
	4.10 指针..... 101
	4.10.1 指针的概念..... 101

4.10.2	指针变量的定义和引用	101
4.10.3	指针和函数	104
4.10.4	指针和数组	109
4.10.5	指向指针的指针	110
4.11	结构体、联合体和枚举类型	111
4.11.1	结构体	111
4.11.2	联合体	113
4.11.3	枚举类型	114
4.12	类	114
4.12.1	类的概述	115
4.12.2	定义类类型	115
4.12.3	无名类类型	116
4.12.4	类的定义点	116
4.12.5	空类	117
4.13	类的继承	117
4.13.1	单一继承	118
4.13.2	多重继承	121
4.14	对象	121
4.15	流式输入输出	122
4.15.1	什么是一个流	122
4.15.2	输出流	123
4.15.3	使用插入运算符和控制格式	124
4.15.4	输入流	126
4.16	函数名重载和操作符重载	127
4.16.1	重载参量类型	127
4.16.2	重载函数的限制	128
4.16.3	重载运算符	129
4.16.4	运算符重载的一般规则	130
4.17	构造函数和析构函数	132
4.17.1	构造函数的作用	132
4.17.2	说明构造函数	132
4.17.3	调用构造函数	133
4.17.4	构造函数与数组	134
4.17.5	析构函数	134
4.17.6	说明析构函数	135
4.17.7	使用析构函数	135
4.17.8	显式的析构函数的调用	135
4.18	虚拟函数	136
4.19	C#语言介绍	140

第5章	对话框	147
5.1	在.NET编译环境中使用资源编辑器 编辑对话框	147
5.2	创建有模式对话框	151
5.3	创建无模式对话框	153
5.4	.NET环境中的消息映射	156
5.5	.NET环境中对话框的数据交换和 数据检查	160
5.6	使用.NET的通用对话框	163
5.6.1	CFileDialog文件选择对话框 的使用	164
5.6.2	CFontDialog字体选择对话框 的使用	164
第6章	Visual Studio.NET提供的 Visual C++控件	166
6.1	概述	166
6.1.1	什么是控件	166
6.1.2	控件分类	166
6.1.3	在.NET的新环境下创建控件	167
6.1.4	控件和对话框的数据交换	168
6.2	Visual Studio.NET环境中的Windows 标准控件	171
6.2.1	按钮控件(CButton类)	171
6.2.2	列表框控件(CListBox类)	175
6.2.3	组合框控件(CComboBox类)	179
6.2.4	文本框控件(CEdit类)	181
6.2.5	静态控件(CStatic类)	184
6.2.6	滚动条(CScrollBar类)	184
6.2.7	标准控件的使用	185
6.3	常用控件	188
6.3.1	进度条控件	188
6.3.2	滑块控件	188
6.3.3	微调器控件	189
6.3.4	图像列表控件	190
6.3.5	列表视图控件	191
6.3.6	树形视图控件	196
6.3.7	Rich编辑控件	201
6.4	自定义控件	203
6.4.1	Windows类与自定义类	203

6.4.2 用于核实信息的编辑控件	204	7.4.3 用基本图元绘制命令进行基本 图元的绘制	295
6.4.3 时钟静态控件	206	7.4.4 源程序清单	305
6.5 ActiveX 控件	213	7.5 坐标变换	319
6.5.1 历史回顾	213	7.5.1 通用坐标变换	319
6.5.2 OLE 控件	215	7.5.2 造型变换	320
6.5.3 ActiveX 控件	215	7.5.3 视图变换	321
6.5.4 开发策略	216	7.5.4 投影变换	323
6.6 新环境下使用 MFC 建立控件的方法	220	7.5.5 视口变换	323
6.6.1 建立控件框架	220	7.6 光照编程	324
6.6.2 控件的注册	222	7.6.1 OpenGL 的光照效果	324
6.6.3 增加属性	223	7.6.2 创建一个光源	325
6.6.4 增加方法	227	7.6.3 OpenGL 的光照模型	327
6.6.5 完善控件功能	230	7.6.4 定义材质的特性	328
6.6.6 增加事件	234	第 8 章 多媒体编程	332
6.6.7 使用属性表	238	8.1 直接使用 Active 制作媒体播放器	332
6.6.8 序列化	242	8.1.1 建立工程	332
6.6.9 ActiveX 控件的字体与图片	244	8.1.2 添加代码	333
6.7 优化控件	252	8.2 MIDI、WAV 及 CD 的播放	335
6.7.1 优化绘制	252	8.2.1 MIDI 的播放	335
6.7.2 其他特性	254	8.2.2 WAV 文件的播放	337
6.8 使用控件	257	8.2.3 CD 的播放	338
6.8.1 组件	257	8.2.4 大型 WAV 文件的播放	339
6.8.2 与 Web 有关的新特性—— 插入网页	257	8.3 视频捕获编程	345
6.8.3 从 Windows 控件中继承	258	8.3.1 Video for Windows 简介	345
第 7 章 Visual Studio .NET 环境下的 OpenGL 程序设计	259	8.3.2 AVICap 编程简介	346
7.1 前言	259	8.3.3 AVICap 编程示例	347
7.1.1 计算机图形学与 OpenGL 编程	259	第 9 章 数据库编程	351
7.1.2 计算机图形学	259	9.1 Microsoft ODBC 数据库管理	351
7.1.3 OpenGL 概述	260	9.1.1 概述	351
7.2 预备知识	261	9.1.2 MFC 提供的 ODBC 数据库类	354
7.2.1 OpenGL 的函数	261	9.1.3 应用 ODBC 编程	354
7.2.2 OpenGL 绘图	262	9.1.4 总结	359
7.3 一个在 .NET 环境中运行的 OpenGL 例子	263	9.2 实例	359
7.4 基本图元的绘制	292	9.2.1 数据库的建立	359
7.4.1 绘图的初始化命令	292	9.3 为数据库指定数据源	360
7.4.2 基本图元的绘制	293	9.3.1 生成应用程序框架	362
		9.3.2 编辑应用程序界面	365
		9.3.3 编辑程序变量	365

9.4 Microsoft Data Access Objects		10.2 利用 WinSock 进行无连接的通信	427
数据库管理	372	10.3 利用 WinSock 进行有连接的通信	429
9.4.1 DAO 特性概述	372	10.4 如何利用 MAPI 实现简单的	
9.4.2 DAO 数据库应用程序分析	373	电子邮件功能	432
9.5 总结	389	10.5 使用 Visual C++ 编写 CGI 程序	435
9.5.1 Visual C++ 开发数据库技术的		10.5.1 CGI 输入输出原理	435
特点	390	10.5.2 应用 WinCGI 系统开发	
9.5.2 Visual C++ 开发数据库技术	390	CGI 程序	439
9.5.3 使用 ODBC API	392	10.5.3 利用 ISAPI 开发 CGI 程序	441
9.5.4 应用 ODBC API 建立应用程序	395	10.5.4 利用 WinInet 开发 Internet 程序	445
9.5.5 SQL 操作	400	10.6 实战网络编程	447
9.5.6 ODBC API 编程总结	409	10.6.1 编写 Ftp 客户程序	447
9.5.7 使用 MFC 访问 ODBC 数据源	410	10.6.2 控制远程计算机拨号上网并	
9.5.8 使用 DAO	412	获取其 IP 地址	453
9.5.9 使用 OLE DB	414	10.7 Web Service 编程	461
9.5.10 使用 ADO	417	10.7.1 如何创建简单的 Web Service	
9.5.11 如何选择合适的数据库访问类型	420	程序	461
9.6 .NET 的新特性, 在 Visual C++ 中		10.7.2 Web Service 和 SOAP	464
使用 SafeArrays	421	10.7.3 为 SOAP 建立基础设施	466
第 10 章 网络编程	423	10.7.4 一个具体的 soap 例子	468
10.1 WinSock 介绍	423		

第 1 章 Visual Studio.NET 简介

1.1 Visual Studio.NET 的新功能

Visual Studio.NET 的全称是 Microsoft Visual Studio.NET, 这是微软执行 .NET 战略的第一个重点产品。该软件集成了 Visual C++ 和 Visual Basic 以及其他和 Internet 相关的编程调试工具。用户可以方便快捷的创建各种 Internet 应用程序。

和以前版本相比, 软件的界面发生了很大的改变, 用户可以方便的调出所需要的对话框和浏览器。而且, “新建项目”对话框集成了所有的项目创建, 用户无需调用特别的程序完成相应的功能, 如图 1.1。

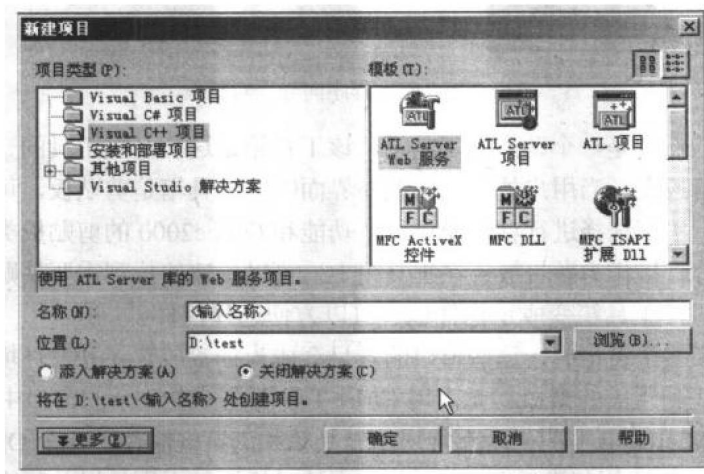


图 1.1 “新建项目”对话框

并且, 该对话框创建的应用程序向导也和上一版本不同, Visual C++6 的应用程序向导是一步一步的往下做, 最终创建出应用程序框架。而这个版本将所有的步骤集成到了一个对话框中, 用户可以迅速的切换到所需更改的项目, 不用一步一步的选择了。在该对话框中作出的选择都会发生实时的效果, 对话框会根据选择改变选项的设置和内容, 如下图 1.2。

在用户对该对话框完成选择后, 并不像以前版本会出现一个综述表, 详细列出用户将创建的程序的各项特性, 所以用户必须在按下 Finish (完成) 按钮之前仔细检查每项的设置, 避免出现问題后只能重新建立应用程序。

当完成了上面对话框操作并按下了 Finish (完成) 按钮后, 就出现了 Visual Studio.NET



的开发界面，该开发界面没有太大的改变，只是在左下角增加了属性列表框，该属性列表框类似于 VB 的属性框，用户可以方便的在这里更改控件的属性。不仅仅如此，该列表框还可以设置相关控件的事件，这也只是在以前版本的 VB 出现的功能，该项功能极大的方便了初级用户，这些用户无需对程序结构有清楚认识，就可以方便的增加或是删除函数和方法。

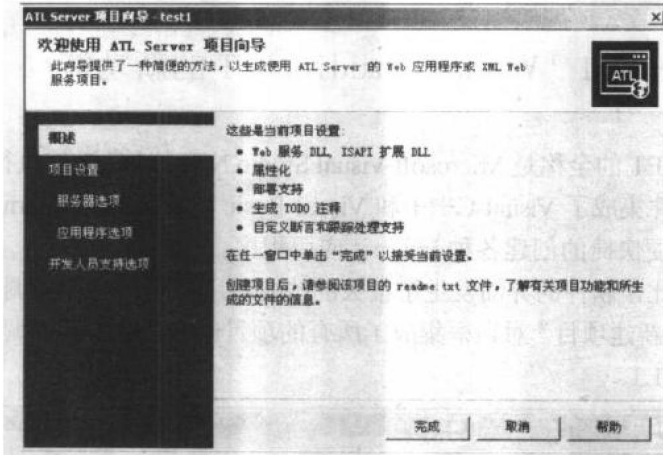


图 1.2 “应用程序向导”对话框

在开发界面的右方是一个动态的工具箱，该工具箱是这套开发工具的一个亮点，它会随着用户的操作环境改变。当用户处于代码编写界面时，工具箱是剪贴板，可以在板上放置多项文字内容，然后根据选择进行粘贴操作。该功能和 Office2000 的剪贴板类似，不过功能更为强大，文字内容直接作为题目放置在工具箱上，可以快速的找到需要粘贴的内容。而当用户在编辑对话框时，工具箱变成了控件箱，可以方便选择控件。

当用户打开了多个窗口时，这些窗口的题目会成为标签放置在开发环境的上方，这样就可以用鼠标快速找到所需的窗口，无需用 Ctrl+Tab 键在一个一个的窗口中循环查找了。

在工具栏上 Microsoft 专门为所有常用的工具箱和浏览框制作了一个 Object Browser（对象浏览器）按钮，通过该按钮可以迅速开关相应的对话框。不用像以前版本一样需要在各项菜单中寻找所需的工具和浏览框了。该浏览器的可以从 View（视图）菜单下其他窗口中调出，弹出菜单如下图 1.3 所示：

从 View（视图）菜单可以看出，包括最常用的 Class View（类视图）、Resource View（资源视图）等都出现在这里。值得注意的是这里出现了一个旧版本没有的 Web 浏览器菜单项，从这里也可以看出这套产品和 Internet 是紧密相关的。并且这套软件还集成了浏览器，在工具栏上可以找到前进和后退按钮。

在创建对话框时，虽然把相关的工具，例如左右对齐，上下对齐等放在了主界面的工具栏上，但是取消了这些操作的快捷键，可能是用户花在界面上的时间相对于代码编写调试时间较少的缘故。

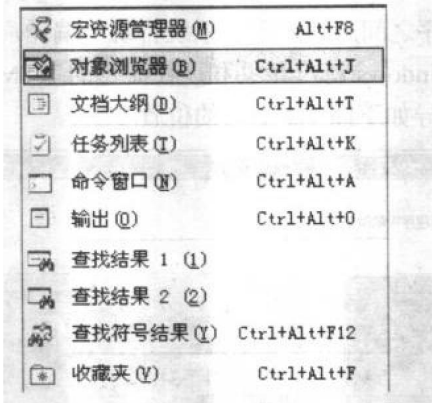


图 1.3 Object Browser (对象浏览器) 按钮的弹出菜单

上述界面上的改观虽然在很大程度上方便了用户的操作，但是也带来了问题。其中最主要的问题，但是也是一个微不足道的问题，就是对显示器的要求增加了，如果显示器的分辨率太低，将无法同时放置如此多的列表框和浏览器，不但无法享受新界面带来的方便和快捷，而且还不得不为了节约空间不断的把某些列表框开开关关。

上面介绍了 Visual Studio.NET 界面上的改变，接着介绍内部变化。

最大的内部变化当属于增加了一种新的语言 C# (读作 C Sharp)，该语言是基于 C 和 C++ 上发展而来的一种语言，在安全性和语法上都较原来的 C 语言有了很大的提高。原来的 C 和 C++ 在有的地方不是很符合语言的习惯，但是由于客观条件所限制而不得不把有的语法做得比较不符合语言规范，所以微软全新推出的 C# 解决了这些问题，不过用户也需要花费相当的时间去学习。但不是所有的用户都需要 C# 语言。微软于是在 New Project 对话框中把 C# 语言单独列出，可以选择所开发程序是否基于该语言。

另一个改变对于习惯于使用 VC6 的用户需要注意，那就是 Visual Studio.NET 取消了 Class Wizard。原来的大部分用户都习惯于通过 Class Wizard 创建类，并用它来为控件批量的分配变量等等。现在这部分用户不得不习惯于使用右键菜单来完成相关操作。虽然说在一个控件上单击鼠标右键只能弹出和该控件相关的变量设置更具有针对性也更易于操作，不会发生变量分配错误等失误，也不会当分配变量时忘了应该对应哪个控件。但是如果控件增多，那么用户将不得不一个一个控件的设置，没有了原来统一设置的方便快捷。

如果需要新建函数和变量，那么在 Class View 列表框中选择相应的类，然后单击鼠标右键，选择 New 菜单项就能选择创建新的函数和变量了。并且，要创建新的对话框类时，就需要使用右键菜单选择 New Class 菜单项来创建。

还有，用户可以在创建应用程序时就可以选择是否需要帮助文件的支持，在编程环境的内部就集成了帮助文件的编辑器和编译器。感觉上编辑器制作得不是很方便，但是完全可以用其他的工具，例如 FrontPage 来弥补。而提供的编译器的确在非常大的程度上方便了用户，现在用户不需要应用第三方的软件进行帮助文件的编译。集成了编译器的另一个好处就是不

容易在帮助文件和应用程序之间产生矛盾，系统将统一编译，大大减少了兼容方面的问题。而且帮助文件还可以在 Windows 标准帮助和现在流行的 HTML 帮助之间选择。是否需要帮助的选择创建应用程序向导如下图 1.4 所示的位置。

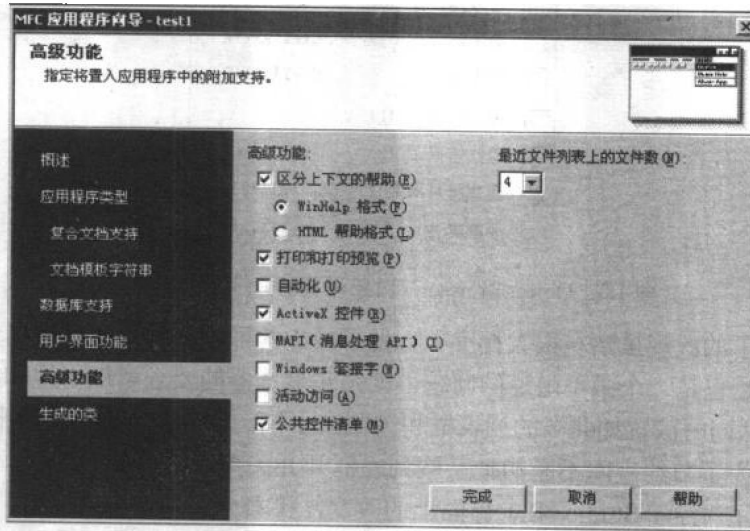


图 1.4 在创建应用程序向导中选择帮助文件的支持

1.2 为什么要使用 Visual C++

当然，使用 Visual C++ 的前提是用户需要编制在 Windows 下运行的程序，而编制 Windows 程序可以有很多的选择，其中最主要的选择是在 Java 和 VB 和 VC 中进行比较。首先，一种编译语言总比一种解释性的语言运行快。一个带有大量公式和宏的电子表格如果用解释性的语言来处理的话，在每次装载的同时都必须解释该程序，更何况解释后的代码还不能和经过编译器优化的代码相比。

除了执行的速度外，对系统的访问也成为了一个衡量语言是否应用广泛的标准，在这个方面，VB 和 Java 需要通过较为复杂的方法或是根本就不能访问系统的底层。出于安全性的考虑，Java 是不能对计算机的硬件进行操作的（除非是通过了授权的可信程序）。

此外，Windows 的程序接口也是一个值得考虑的问题，当然最好的应用程序接口就是由 Microsoft 提供的 API，Microsoft 提供的接口就是 MFC 库，而 Visual C++ 历来都对该类库提供了最全面的支持。

1.3 Visual C++.NET 的运行条件

Visual C++.NET 由于是微软 .NET 战略的一个主打产品,所以它必须安装相应的 Internet 组件才能运行。下列组件就是要求安装的。

1. Microsoft Windows Installer 1.1 (9x)
2. Microsoft Windows Management Infrastructure
3. Setup Runtime Files
4. Microsoft Office Shared Components
5. Internet Explorer Web Forms QFE
6. Microsoft XML Parser (MSXML) v3.0
7. Microsoft Data Access Components
8. Microsoft Jet 4.0 Service Pack 3
9. Microsoft .NET Framework SDK
10. Microsoft Internet Explorer 5.5 and Internet Tools

如果操作系统不能满足这些需要,那么安装程序将会出现如下图 1.5 的对话框提醒用户缺少的组件。

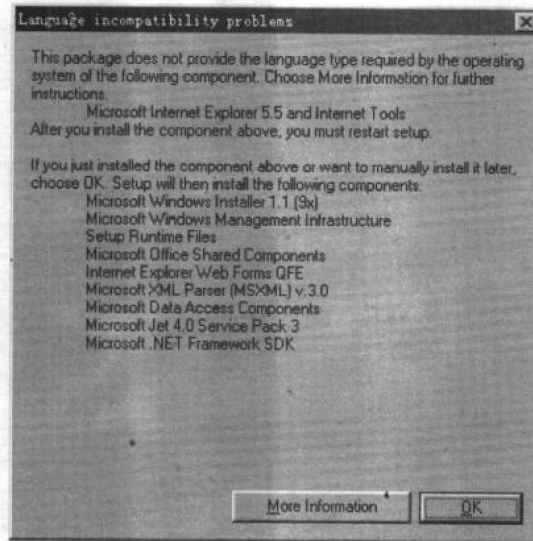


图 1.5 Language incompatibility problems 对话框

用户可以根据自已的情况安装相应的组件,从而使安装程序能够继续进行。如果用户不能满足安装条件,安装将不能进行。

虽然微软把 Visual Studio.NET 设定为在几乎所有的现行 Windows 平台上都能运行,其中



包括 Windows NT 4.0 以及 Windows 95, 但是就现在的条件来看, 运行 WinNT4.0 和 Win95 的计算机的配置都不高, 而 Visual Studio.NET 是内存杀手, 即使仅仅安装了而没有运行, 都会导致系统可用资源的下降。微软并没有说明这套软件需要在何种硬件条件下运行, 但是就使用情况来看, 以下为基本配置:

主频在 500 以上的中央处理器, 64MB 的内存, 有 700MB 以上的空余容量的硬盘用于安装 Visual Studio.NET 及基本的 MSDN 文件。

建议配置:

主频在 800 以上的中央处理器, 128MB 的内存, 1.5G 的硬盘空间用于安装 Visual Studio.NET 以及所有的 MSDN 文件。安装所有 MSDN 文件的好处是在查阅附带的程序实例时不必使用光盘。有条件的情况下, 可以使用 17 英寸以上的显示器, 这样可以获得较好的开发环境布局。

第 2 章 Visual C++ .NET 的开发环境

2.1 启动 Visual C++ .NET

Visual C++ .NET 和 Visual Basic .NET 都被集成于 Visual Studio .NET 内，所以启动了 Visual Studio .NET 就等于是启动了 VC。在开始菜单中选择了 Visual Studio .NET 项后就可以启动开发环境了。并且，当安装完毕后，Visual Studio .NET 会自动建立和相关文件的连接，当选择并运行了相关文件后就会启动 Visual Studio .NET 打开该文件。

2.2 Visual Studio .NET 的开发环境

运行 Visual Studio .NET 后，出现了如下图 2.1 所示的开发环境界面，注意，该环境是处于 1280×1024 分辨率下的状态，从图中可以看出窗体的布局正好合适，如果降低分辨率将会导致编写代码的区域大大缩小。

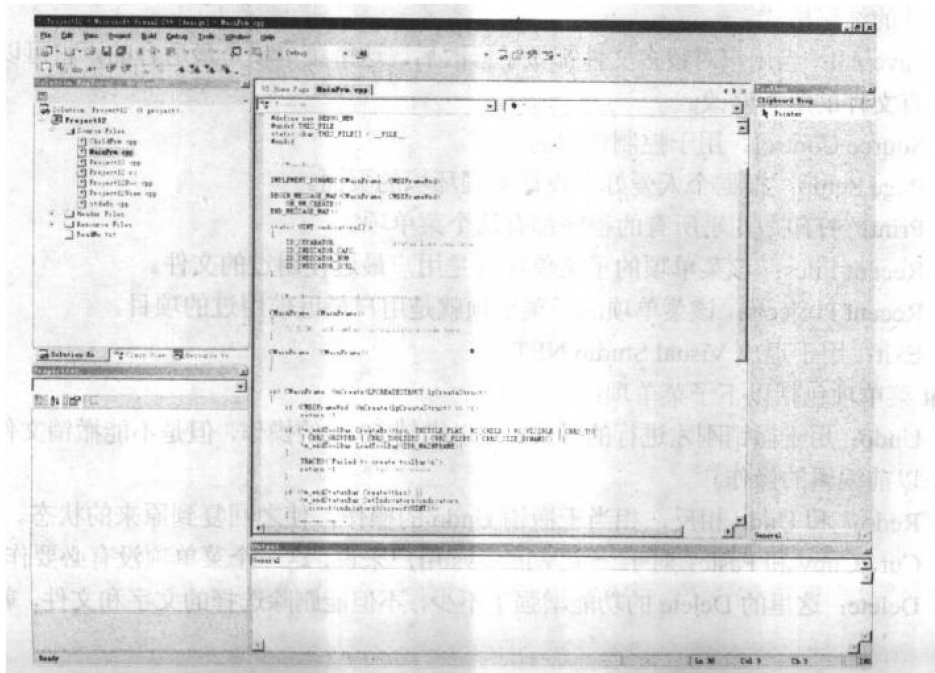


图 2.1 处于 1280×1024 分辨率下的开发环境



和以前版本相同的是，保留了备受好评的类列表和资源列表，但是原来的文件列表改名为 Solution 浏览器，当然功能是大大增强了，甚至可以在该浏览器内直接选择资源文件或是其他文件，程序都会自动调出相应的工具栏来编辑。值得推荐的是开发环境右方的工具箱，这是一个新增的功能，该功能实际上是把一些常用的对话框集成在一起，还增加了一些 Office 2000 的编辑功能，例如多个剪贴板就是 Office 2000 中的功能。用户可以在该工具箱中迅速找到自己所需的功能。下面将对该环境作一介绍。

2.2.1 菜单栏

由于 Visual Studio 的菜单项非常的多，如果逐一介绍将会占去较大篇幅，故只介绍常用的和新增并有特色的。

File 菜单包括如下子菜单项：

- **New:** 用于创建新的工程文件或其他与编程相关的文件。
- **Open:** 打开用 New 创建的或是以前存在的文件。
- **Close:** 关闭当前的子窗口。
- **Add Project:** 加入新的项目。
- **Open Solution:** 这个菜单项替换了原来的 Open Workspace，用于打开一个编程平台。
- **Close Solution:** 当用户需要打开一个新的 Solution 时，可以用该菜单项关闭打开的 Solution。
- **Save 和 Save As:** 用来保存对单个文件的修改。
- **Advanced Save Options:** 这个菜单项也是新增的，它用于设置保存所用的语言以及文件的结尾格式。
- **Save All:** 当用户对很多文件都做了修改后，这个菜单项是最适合的，它可以保存所有文件的所有修改。
- **Source Control:** 用于控制资源的共享。
- **Page Setup:** 根据个人爱好来设计编程环境的页面。
- **Print:** 打印，几乎所有的程序都有这个菜单项。
- **Recent Files:** 该菜单项的子菜单项就是用户最近使用过的文件。
- **Recent Projects:** 该菜单项的子菜单项就是用户最近使用过的项目。
- **Exit:** 用于退出 Visual Studio.NET。

Edit 菜单项包括以下子菜单项：

- **Undo:** 用于撤销刚才进行的操作，可以撤销 50 次的操作，但是不能撤销文件保存或以前编辑的操作。
- **Redo:** 和 Undo 相反，相当于撤销 Undo 的操作，使之回复到原来的状态。
- **Cut, Copy 和 Paste:** 对于一个 Windows 用户来说，这三个菜单项没有必要作介绍了。
- **Delete:** 这里的 Delete 的功能增强了不少，不但能删除选择的文字和文件，就连变量