

精通

Visual C++ .NET 2003

窗口程序设计

位元文化 编著

- ◆ 本书以范例为教学与讲解的主线，是学习MFC窗口程序设计的最佳入门教材。
- ◆ 详尽的程序注释与完整且清楚的概念说明，让窗口程序设计的技巧一览无遗。
- ◆ 如果您想轻松学习应用MFC开发窗口程序，快速深入到文档/视图（Doc/View）应用程序架构与对话框应用程序，本书是您的最佳选择。



清华大学出版社

精通 Visual C++ .NET 2003 窗口程序设计

位元文化 编著



清华大学出版社
北京

内 容 简 介

随着.NET平台的推出,微软也发布了新的窗口程序集成开发环境 Visual Studio .NET,且增加了 Visual C++ .NET、C#、Visual Basic .NET等新语言,而其中 C++语言窗口程序设计的主角还是 MFC。本书以范例为教学与讲解的主线,带领读者了解如何运用 MFC 开发窗口程序,是学习 MFC 窗口程序设计最佳入门教材。本书分为 5 篇,共 30 章,主要内容包括:窗口程序设计基础,程序资源的建立与基本绘图原理,深入应用程序构建,对话框、对话框应用程序与各种控件,网络及数据库程序设计等。

本书适合使用 Visual C++ .NET 进行窗口程序设计开发的初、中级用户参考,也可作为高等院校及计算机培训学校学习窗口程序设计的学生的教材或参考书,还适合于对窗口程序开发感兴趣的自学者自学使用。

本书繁体字版书名为《精通 Visual C++ .NET 2003 视窗程式设计》,由文魁资讯股份有限公司出版,版权属位元文化所有。本书简体字中文版由文魁资讯股份有限公司授权清华大学出版社独家出版。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内容。

北京市版权局著作权合同登记号 图字:01-2003-8764

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

精通 Visual C++ .NET 2003 窗口程序设计/位元文化编著. —北京:清华大学出版社,2006.11
ISBN 7-302-14108-9

I.精… II.位… III.C 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(2006)第 121801 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:桑任松

排版人员:朱 康

印 刷 者:北京市清华园胶印厂

装 订 者:三河市化甲屯小学装订二厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:44.5 字数:1052 千字

版 次:2006 年 11 月第 1 版 2006 年 11 月第 1 次印刷

书 号:ISBN 7-302-14018-9/TP·8422

印 数:1~4000

定 价:62.00 元

目 录

第一篇 窗口程序设计基础

第 1 章 简介 Visual C++ .NET 窗口程序设计	2
1.1 窗口程序设计的基本概念	2
1.1.1 事件、消息与窗口运行	2
1.1.2 窗口的基本构造	2
1.1.3 资源的概念	3
1.2 如何编写窗口程序	4
1.2.1 什么是应用程序框架	4
1.2.2 应用程序框架与面向对象	5
1.3 强大的集成程序开发环境——Visual C++ .NET	5
1.3.1 什么是集成程序开发环境	5
1.3.2 Visual C++ .NET 的画面介绍	5
1.3.3 MFC 的类继承结构	8
1.4 从 Visual C++ 到 Visual C++ .NET	8
第 2 章 Hello MFC——窗口程序设计的初步体验	10
2.1 建立窗口程序的基本概念	10
2.2 第一个窗口程序——Hello MFC	11
2.2.1 建立 Hello MFC	11
2.2.2 CWinApp 类与程序入口点	16
2.2.3 窗口框架对象	17
第 3 章 Visual C++ .NET 的操作(一)——建立项目与程序编译/调试	22
3.1 项目的介绍	22
3.1.1 Visual C++ .NET 使用的文件	22
3.1.2 项目/方案的打开、关闭与保存	23
3.1.3 项目中文件的操作	24
3.2 编译程序	26
3.3 排除错误——程序的调试	27
3.3.1 程序的调试	27
3.3.2 暂停程序的执行	28
3.3.3 变量值的查看	31
3.4 帮助文件	33
第 4 章 自定义窗口框架与资源文件的运用	37
4.1 自定义窗口框架对象	37
4.2 MyFrame 程序示例	37
4.3 自定义窗口框架类与资源文件	42
4.3.1 窗口框架对象的构建	42
4.3.2 自定义窗口框架与资源文件的建立	42
4.3.3 在窗口框架中建立菜单	43
第 5 章 窗口的消息处理	44
5.1 窗口消息的传递与处理	44
5.2 Message 程序示例	46
5.3 消息映射与响应函数的建立	48
5.4 利用鼠标绘图	49
5.5 消息框的使用与窗口的关闭	51
第 6 章 窗口应用程序架构	54
6.1 什么是文档/视图架构	54
6.2 以文档/视图为架构的窗口应用程序	55
6.2.1 Doc_View 程序示例	55

6.2.2 文档/视图的架构与 建立步骤	57	6.3 文档/视图架构的应用	63
6.2.3 单文档模板类的应用	60	6.3.1 窗口的重绘	63
6.2.4 CView 类与 CDocument 类的使用	62	6.3.2 repaint 程序示例	63
		6.3.3 文档/视图架构的运作机制	65
		6.3.4 窗口的重绘	67

第二篇 程序资源的建立与基本绘图原理

第 7 章 Visual C++ .NET 的操作(二) ——资源编辑器的使用	70	10.1.2 加载快捷键	115
7.1 资源编辑器的基本操作	70	10.2 工具栏、状态栏与字符串表 的使用	116
7.2 菜单	72	10.2.1 简介工具栏、状态栏 与字符串表	116
7.3 图标、位图、光标	75	10.2.2 controlbar 程序示例	116
7.4 工具栏	79	10.2.3 工具栏的建立	120
7.5 对话框	81	10.2.4 状态栏的建立	124
7.6 字符串表	83	第 11 章 图标、位图与鼠标 指针资源	129
7.7 标识符的管理	84	11.1 简介图标、位图与鼠标 指针资源	129
7.8 快捷键	86	11.2 图标、位图与鼠标指针资源 的使用	130
第 8 章 菜单的建立	87	11.2.1 resource 程序示例	130
8.1 菜单的简介	87	11.2.2 图标资源的使用	134
8.2 menu 程序示例	89	11.2.3 位图资源的使用	135
8.3 菜单的建立与设置	95	11.2.4 鼠标指针资源的使用	135
8.4 菜单的切换	96	11.3 创建 Painter 窗口	137
8.5 利用 CMenu 类控制菜单	96	11.3.1 简介 Painter 窗口	137
8.6 修改系统菜单与建立 快捷菜单	105	11.3.2 painter1 程序示例	138
第 9 章 UPDATE_COMMAND_UI 消息	108	11.3.3 painter1 的程序架构	143
9.1 什么是 UPDATE_COMMAND_UI 消息	108	11.3.4 painter1 的自定义类 说明	143
9.2 COMMAND_UI 程序示例	108	11.3.5 控制工具栏的隐藏与 显示	146
9.3 UPDATE_COMMAND_UI 消息 的接收与响应	111	第 12 章 基本绘图原理	148
9.4 UI 组件的修改	112	12.1 窗口的基本绘图原理	148
第 10 章 快捷键、工具栏、状态栏 与字符串表	113	12.1.1 绘图设备接口	148
10.1 快捷键的建立	113	12.1.2 设备环境	149
10.1.1 accelerators 程序 示例	113		

12.1.3 颜色的定义.....	149	12.4.4 painter2 的自定义类 说明.....	166
12.1.4 MFC 的 GDI 类.....	150	12.4.5 MyView 类所处理的 消息.....	172
12.1.5 建立画笔对象.....	150	12.4.6 Shape::draw 的奥妙—— 动态链接的应用.....	177
12.1.6 建立画刷对象.....	151		
12.2 CDC 类的介绍.....	152	第 13 章 图形对象的记录与重绘.....	179
12.2.1 CDC 类的派生类.....	152	13.1 记录图形对象与窗口重绘 的机制.....	179
12.2.2 CDC 类提供的形状 绘制函数.....	152	13.2 painter3 程序示例.....	180
12.3 画笔、画刷与绘图模式的控制.....	155	13.3 窗口重绘的机制.....	189
12.3.1 画笔与画刷的选取.....	155	13.4 painter3 的自定义类.....	189
12.3.2 绘图模式的控制.....	156	13.5 形状对象的重绘机制.....	191
12.4 建立 Painter 的绘图功能.....	157		
12.4.1 painter2 的绘图功能.....	157		
12.4.2 painter2 程序示例.....	158		
12.4.3 painter2 的程序架构.....	165		
第三篇 深入应用程序架构			
第 14 章 文件操作机制的建立.....	198		
14.1 MFC 的文件保存机制.....	198	16.2.3 建立动态分割窗口.....	233
14.2 painter4 程序示例.....	199	16.2.4 子窗口的同步更新.....	234
14.3 painter4 的自定义类.....	209	16.3 静态分割窗口的建立.....	237
14.4 建立保存数据的序列化 机制.....	210	16.3.1 如何建立静态分割 窗口.....	237
14.5 建立文件操作功能.....	213	16.3.2 StaticSplit 程序示例.....	238
		16.3.3 静态分割窗口的建立.....	249
		16.3.4 建立查看整体画布的 GlobalView 类.....	251
第 15 章 具有滚动条的窗口与 窗口部分重绘.....	216	16.3.5 子窗口的更新机制.....	254
15.1 具有滚动条的窗口.....	216		
15.2 ScrollView 程序示例.....	217	第 17 章 多文档窗口程序的建立.....	258
15.3 可滚动范围的设置.....	220	17.1 如何建立 MDI 窗口程序.....	258
15.4 客户区与文件间的坐标转换.....	222	17.2 MDI 程序示例.....	260
15.5 重绘部分窗口.....	226	17.3 MDI 窗口程序使用的资源.....	267
		17.4 MDI 窗口程序的 InitInstance 函数.....	268
第 16 章 分割窗口.....	229	17.5 CMDIFrame 类的功能.....	272
16.1 多窗口程序的建立.....	229	17.6 主/子窗口框架对象的取得.....	275
16.2 动态分割窗口的建立.....	230		
16.2.1 如何动态分割窗口.....	230	第 18 章 文本编辑器的建立.....	277
16.2.2 DynaSplit 程序示例.....	231	18.1 Editor 程序示例.....	277

18.2	多种文档编辑器.....	292	第 20 章 Visual C++ .NET 的操作(三)	
18.3	建立文本编辑器.....	293	——程序快速开发工具	
18.3.1	View 类的建立.....	293	的应用.....	323
18.3.2	Document 类的建立.....	301	20.1 辅助程序开发的工具.....	323
第 19 章 打印与预览.....		304	20.2 应用程序向导.....	324
19.1	MFC 的打印、预览		20.2.1 应用程序向导的	
	打印机制.....	304	使用说明.....	324
19.2	Print 程序示例.....	306	20.2.2 应用程序向导所产生	
19.3	建立打印机制需要		文件的说明.....	328
	加载的资源.....	315	20.3 程序代码向导的使用.....	345
19.4	坐标系统的转换.....	315	20.3.1 程序代码向导的用途.....	345
19.5	图文件与文字文件的打印及		20.3.2 新建类.....	347
	预览打印.....	319	20.3.3 新建类的函数.....	349
19.5.1	图文件的打印与预览		20.3.4 新建类的属性.....	352
	打印.....	319	20.3.5 事件响应函数的建立.....	353
19.5.2	文字文件的打印与		20.3.6 消息响应函数的建立.....	354
	预览打印.....	322	20.4 运用 Visual C++ .NET 工具	
			建立项目.....	355

第四篇 对话框、对话框应用程序及各种控件

第 21 章 模态对话框的建立.....		358	22.5 DDX 与 DDV 函数的说明.....	398
21.1 对话框简介.....		358	第 23 章 非模态对话框.....	401
21.1.1 对话框的种类与功能.....		358	23.1 非模态对话框的建立.....	401
21.1.2 对话框组成.....		359	23.2 painter7 程序范例.....	401
21.1.3 对话框资源与对话框			23.3 painter7 的自定义类.....	415
对象.....		360	23.4 非模态对话框的建立.....	416
21.2 painter5 程序示例.....		360	23.5 数据交换机制.....	420
21.3 painter5 的自定义类.....		373	23.6 其他更改的部分.....	422
21.4 painter5 的架构.....		373	第 24 章 Visual C++ .NET 的操作(四)	
21.5 SWidthDlg 对话框类.....		375	——对话框类的建立.....	423
21.6 模态对话框对象的建立.....		381	24.1 快速建立对话框类.....	423
第 22 章 对话框的数据交换与			24.1.1 程序代码向导建立	
检查机制.....		383	对话框类的机制.....	423
22.1 DDX 与 DDV 机制.....		383	24.1.2 建立对话框类.....	423
22.2 painter6 程序示例.....		384	24.1.3 建立数据交换与	
22.3 DDX 与 DDV 机制的建立.....		395	检查机制.....	425
22.4 数据交换与检查机制.....		396	24.1.4 其他相关资源的建立.....	426

24.2	快速建立对话框应用程序.....	429	25.5.5	初始状态的设置.....	508
第 25 章	对话框应用程序与控件		25.5.6	WM_HSCROLL 与 WM_VSCROLL 消息.....	510
	的使用	437	25.5.7	Slider 与 Scroll Bar 的 滚动.....	510
25.1	对话框应用程序的使用.....	437	25.5.8	Edit Box 的 EN_CHANGE 事件.....	513
25.2	Button 控件与 Static Text 控件.....	437	25.5.9	Check Box 的 BN_CLICKED 消息.....	514
25.2.1	Button 控件与 Static Text 控件简介.....	437	25.6	树状控件、列表控件与 单选按钮.....	515
25.2.2	计算器程序示例.....	438	25.6.1	树状控件、列表控件与 单选按钮简介.....	515
25.2.3	OnCommand 消息的 响应.....	448	25.6.2	DirView 程序示例.....	516
25.2.4	OnBnClickedClear 函数.....	453	25.6.3	文件信息的取得.....	531
25.3	Picture 控件.....	454	25.6.4	Image List 对象的建立.....	537
25.3.1	Timer(小时钟)程序示例.....	454	25.6.5	树状控件的建立.....	538
25.3.2	WM_TIMER 消息的 使用.....	464	25.6.6	列表控件的建立.....	544
25.3.3	系统时间的取得与 CTime 对象的使用.....	465	25.6.7	单选按钮的使用.....	551
25.4	List Box、Combo Box 控件与 多页对话框的建立.....	467	第 26 章	文件对话框与颜色	
25.4.1	List Box、Combo Box 与多页对话框简介.....	467		对话框的应用	554
25.4.2	ListProperty 程序示例.....	468	26.1	文件对话框的应用.....	554
25.4.3	多页对话框的建立.....	485	26.1.1	文件对话框的类型 与应用.....	554
25.4.4	List Box 的操作.....	489	26.1.2	FDiallog 程序示例.....	555
25.4.5	Combo Box 的操作.....	491	26.1.3	文件对话框的建立 与使用.....	560
25.5	Scroll Bar、Slider、Spin、Edit Box 控件与 Check Box 控件.....	493	26.2	颜色对话框的应用.....	564
25.5.1	滚动类控件简介.....	493	26.2.1	利用颜色对话框 设置颜色.....	564
25.5.2	Edit Box 与 Check Box 简介.....	495	26.2.2	CDiallog 程序示例.....	564
25.5.3	SldCtrl 程序示例.....	495	26.2.3	颜色对话框的建立与 使用.....	566
25.5.4	各控件间的关系.....	507			

第五篇 网络及数据库程序设计

第 27 章	进程与线程的概念	570	27.2	Thread 程序示例.....	571
27.1	什么是进程与线程.....	570	27.3	建立工作者线程.....	583
			27.3.1	建立线程函数.....	583

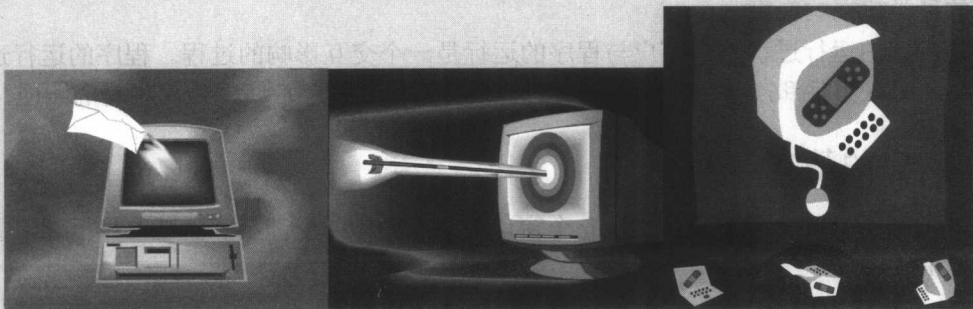
27.3.2 子线程的产生.....	586	第 30 章 MFC 与数据库	654
27.4 线程的执行.....	588	30.1 数据库管理系统的建立.....	654
27.4.1 暂停间隔的设置.....	588	30.1.1 简介数据库管理系统.....	654
27.4.2 暂停/恢复线程的执行.....	588	30.1.2 数据库操作程序的概念.....	654
27.4.3 优先执行权的设置.....	590	30.2 ODBC 程序范例.....	656
27.4.4 检查线程的执行状态.....	591	30.2.1 建立数据库.....	656
27.5 中断线程的执行.....	592	30.2.2 建立 ODBC 连接.....	658
27.6 自定义线程的结束执行消息.....	594	30.2.3 建立 ODBC 程序范例.....	660
第 28 章 FTP 程序的编写	597	30.3 RFX 数据交换机制.....	674
28.1 以 MFC 编写网络程序.....	597	30.3.1 记录集类的定义与 RFX 机制.....	674
28.1.1 因特网的简介.....	597	30.3.2 取得 RFX 机制的数据源 ...	677
28.1.2 MFC 中与建立网络 程序的相关类.....	598	30.4 CRecordset 类的使用.....	677
28.2 FTP 程序示例.....	599	30.4.1 建立/打开 CRecordset 对象.....	677
28.3 FTP 连接的建立与中断.....	619	30.4.2 数据的新增/删除/修改.....	680
28.3.1 FTP 联机的建立.....	619	30.4.3 设置数据筛选条件与 排序方式.....	682
28.3.2 联机错误的处理机制.....	623	30.5 CDatabase 类的使用.....	684
28.3.3 FTP 联机的中断.....	625	30.5.1 CDatabase 对象的打开 与关闭.....	684
28.4 取得服务器端的文件资料.....	627	30.5.2 执行 SQL 语句.....	685
28.5 利用线程执行文件下载.....	633	附录 A 本书所使用 MFC 函数索引	688
第 29 章 一个简单的浏览器	638		
29.1 简介 CHtmlView.....	638		
29.2 HViewer 程序示例.....	638		
29.3 CHtmlView 类的介绍.....	648		

精通 Visual C++ .NET 2003 窗口程序设计

第一篇 窗口程序设计基础

本篇介绍窗口程序设计的基础内容，包括第 1 章至第 6 章。各章的主题如下：

- 简介 Visual C++ .NET 窗口程序设计；
- Hello MFC——窗口程序设计的初步体验；
- Visual C++ .NET 的操作(一)——建立项目与程序编译/调试；
- 自定义窗口框架与资源文件的运用；
- 窗口的消息处理；
- 窗口应用程序架构。



第 1 章 简介 Visual C++ .NET

窗口程序设计

学习窗口程序设计，往往让人有一种难度颇高的感觉，这的确也是事实，相对于 DOS 模式的程序编写，编写窗口程序的确技术高深、原理艰难了些。但是这也是因为窗口程序拥有更人性化、更直观的操作界面。在这一章里，将简单地介绍编写窗口程序的一些基本概念，并介绍 Visual C++ .NET 这个好用的窗口程序集成开发环境。

本章要点：

- 窗口程序设计的基本概念；
- 如何编写窗口程序；
- 强大的集成程序开发环境——Visual C++ .NET；
- 从 Visual C++ 到 Visual C++ .NET。

1.1 窗口程序设计的基本概念

1.1.1 事件、消息与窗口运行

当编写窗口程序时，在脑海里，应该会不断地询问自己这样的问题：

用户会执行什么动作？程序如何响应这些动作？

为什么在设计窗口程序时，会思考这些问题呢？那是因为窗口的运行基础就是事件响应。相对于 DOS 模式下大部分以流程概念为主的程序编写方式，窗口程序以事件响应作为程序设计的基本机理，对于初学者来说是比较难适应的。不过，如果读者已经具备了面向对象的概念，在适应上应该不会太难。若读者并不懂什么是面向对象，建议最好先看看有关面向对象的入门书，学习一下面向对象的概念。

在以窗口为界面的软件操作环境下，用户会通过鼠标和键盘等设备，提出许多不同的事件要求，如单击按钮、选择菜单等。这些由用户触发的事件(不论是单击按钮，或者选择菜单)，都将产生消息传递给窗口。因此，在窗口运行的过程中，用户所产生的事件在程序里都将化身为消息，传递给窗口。而不同的事件将产生不同的消息，因此，在程序里，将依照消息的种类建立响应机制。

在窗口程序的运行过程里，用户与程序的运行是一个交互影响的过程。程序的运行过程将视用户所触发事件和产生消息的不同，进而运行不同的程序片段，做出相对的响应。

1.1.2 窗口的基本构造

图 1.1 所示是最基本的窗口形式，整个窗口大致可分为两个部分：一是窗口框架，二是客户区(Client Area)。

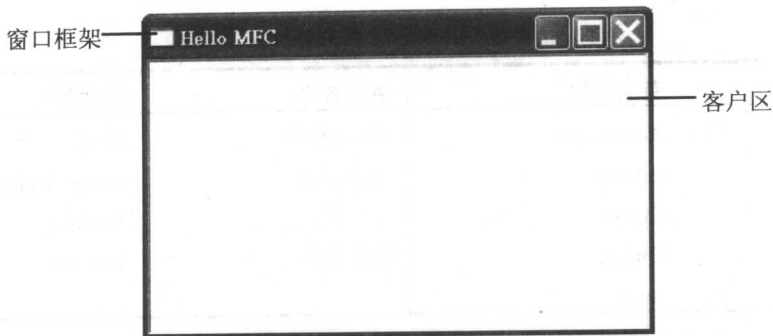


图 1.1

窗口框架用于容纳窗口界面的操作组件，一般窗口程序常见的窗口组件有菜单栏、工具栏、状态栏等。而客户区则是用户工作的区域，用户可以在其中输入文字、绘图等，因此，读者也可以将其称之为工作区。图 1.2 是文字编辑软件 Word 的窗口界面，在该窗口界面里，拥有一般常见的窗口组件与编辑文件用的客户区。

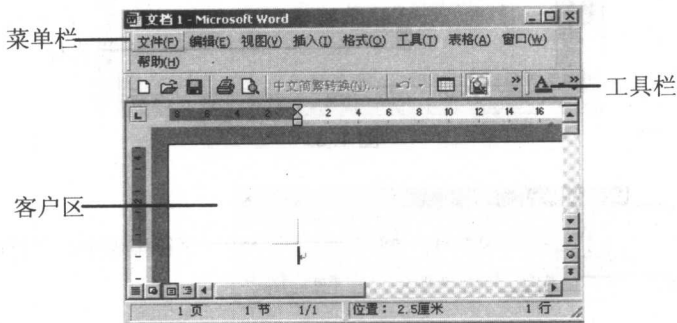


图 1.2

1.1.3 资源的概念

1. 什么是资源

在前面的说明里，介绍了消息在窗口程序运行过程中所扮演的角色。有了这样的基本认知，接着，再来谈谈窗口程序的资源概念。

相对于文字模式下运行的 DOS 程序，窗口程序的图形化界面是相当复杂且多变化的。图形化界面可以拥有菜单、工具栏、状态栏等窗口组件。这些协助用户操作程序的可视化对象，在程序编写时，将被视为程序建立窗口界面所使用的资源。

2. 资源文件的使用

前述窗口程序所使用的资源对象，在编写窗口程序时，将定义在资源文件里。表 1.1 中列出用 Visual C++ .NET 开发窗口程序时，在资源文件中所使用的各种窗口资源。

图 1.3 和图 1.4 将把表 1.1 所提及的资源，利用 Word 的窗口界面加以说明，相信更能让您了解资源在窗口应用程序中扮演的角色。

表 1.1 资源文件中所使用的各种窗口资源

中文名称	英文名称	中文名称	英文名称
加速键表	Accelerator	菜单(选单)	Menu
位图	Bitmap	字符串表	String Table
光标	Cursor	工具栏	Toolbar
对话框	Dialog	版本信息	Version
图标	Icon		

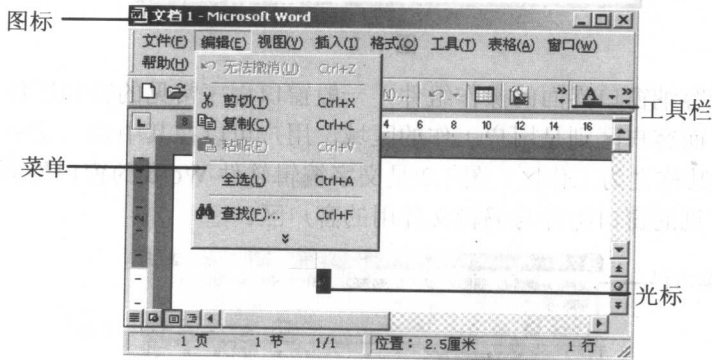


图 1.3

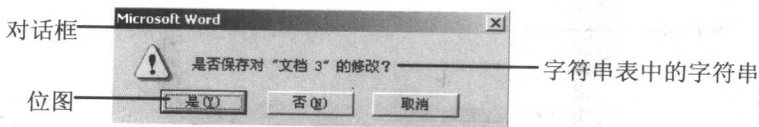


图 1.4

3. 资源文件的建立

对于资源的建立, 可运用 Visual C++ .NET 所提供的资源编辑器建立/编辑您的资源, 并自动产生资源文件(各资源编辑器的使用请参考第 7 章)。

1.2 如何编写窗口程序

1.2.1 什么是应用程序框架

想要编写窗口程序, 有许多方法, 其中一个方法, 是利用由软件厂商所提供的类库, 这种类库称之为应用程序框架(Application Frameworks, 以下简称为 AF)。什么是 AF 呢? 按照字面上的含意, 可以翻译成“应用程序框架或应用软件架构”。从字面上的意思可以清楚地了解, AF 的用途在提供建立窗口的软件架构。

当利用 AF 建立窗口程序时, AF 将提供所有窗口程序共有的部分, 具体地讲, 比如: 窗口、菜单、工具栏等, 以及消息的拦截机制。也就是说, AF 已经构建好窗口程序的基本架构。程序设计人员只要利用它, 窗口程序就可以拥有基本的窗口功能, 其余的心思只要

放在增加或者修改程序的功能，以符合自己的需求即可。

其实利用 AF 开发窗口程序与使用程序语言编写程序是一样的，都是利用别人的成果，开发符合自己需求的程序。只是使用程序语言编写程序时，利用的是前人定义出来的特定语法，这些特定语法将操作计算机的细节包装起来。就像在 C++ 中，要将数据输出到屏幕上时，只需要使用 cout 对象，但不必去管计算机究竟如何从程序得到数据，又如何将这些数据输出到屏幕上。同样的道理，使用 AF 建立窗口程序时，亦不必理会计算机究竟如何在屏幕上显示一个窗口，反正别人写好的拿来用就是了。

1.2.2 应用程序框架与面向对象

从以上的说明里，可以看出应用程序框架(简称 AF)实现了软件重用的思想。这跟面向对象技术有密不可分的关系，因为 AF 本身就是一个很复杂的类阶层，运用 AF 的方法就是利用类继承与聚合的概念。前者继承 AF 的类，以修改方式产生窗口，如继承文字窗口类；后者则是直接使用 AF 的类，建立窗口组件，如使用按钮类。所以，面向对象概念是学习窗口程序设计的基本能力之一，有了面向对象概念，才能充分利用 AF，设计出好的窗口程序。

1.3 强大的集成程序开发环境 —— Visual C++ .NET

1.3.1 什么是集成程序开发环境

当进行程序开发时，一个提供程序编辑与调试的良好程序开发环境，将可以提高程序设计人员的工作效率。对于开发窗口程序而言，更是这样，因为窗口程序除了程序代码以外，还有需要使用其他资源，如图标、菜单、对话框等。若没有一个能够提供建立和编辑这些程序资源的工具，那对于程序设计人员而言，开发窗口程序将会是一场梦魇。作为 Visual Studio .NET 的重要成员之一的 Visual C++ .NET 就是一个集成各种窗口程序开发所需工具的工作环境，它除了提供上述程序资源的建立工具，便利窗口程序的开发外，更提供方便的调试工具，可有效节省程序设计人员花费在为程序调试上的时间与精力。

除此之外，在窗口程序设计方面，Visual C++ .NET 更提供了向导，协助用户快速建立窗口程序的大致架构，让程序设计人员只需要编写部分程序代码，即可完成一个窗口应用程序。但是学习编写窗口程序的初期，并不建议您直接学习向导的应用，因为在不了解整个窗口程序建立的过程时，就学习应用向导，可能学了半天，还是不知道窗口程序的架构与原理。所以，在本书前半部里，将以手工方式建立窗口程序，到了后半部，才告诉您如何应用向导的协助建立窗口程序，相信这样的学习过程是比较适合初学者的。

1.3.2 Visual C++ .NET 的画面介绍

乍见 Visual C++ .NET 的窗口画面，相信大部分的人，都会被复杂的窗口工作画面吓住。不过，为给程序设计人员提供具备完整消息的工作环境，使用复杂的工作画面是可理解的。也因此，在 Visual C++ .NET 里，建立的每个应用程序，除了有源代码及机器代码外，还有

存储其他相关信息的文件，而这些文件将会被组织成一个方案/项目，方便我们利用 Visual C++ .NET 加以管理。图 1.5 所示是 Visual C++ .NET 的工作画面。

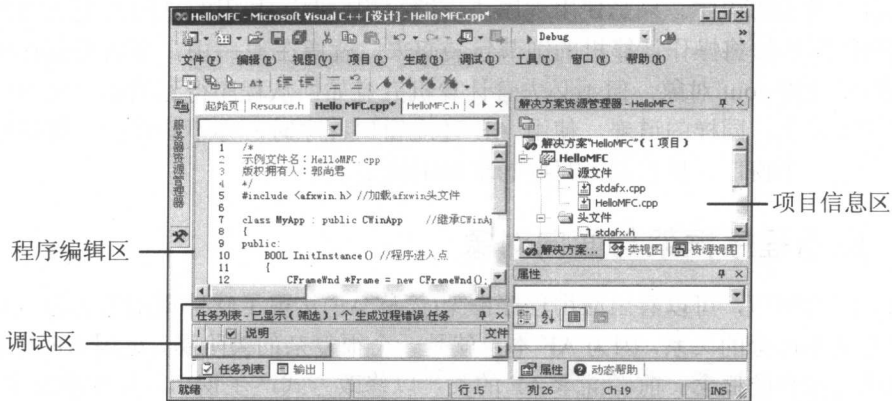


图 1.5

整个工作画面主要分为 3 个区域。

1. 程序编辑区

顾名思义，程序编辑区就是编写程序的区域。您将在这个区域里，一行一行地堆砌出窗口程序。另外，当要建立窗口程序所使用的资源时，该区则成为建立各种资源的编辑器。

2. 项目信息区

该区有 3 个视图，一是类视图(Class View)，如图 1.6 所示。切换至该视图后，您可以看到该项目中的所有类，单击某类前的 + 号，即可展开该类，观看该类的成员有哪些。若想要更改该类某成员的程序代码，更可直接在该成员上双击，左侧的程序编辑区便会切换到该类成员程序代码的位置。

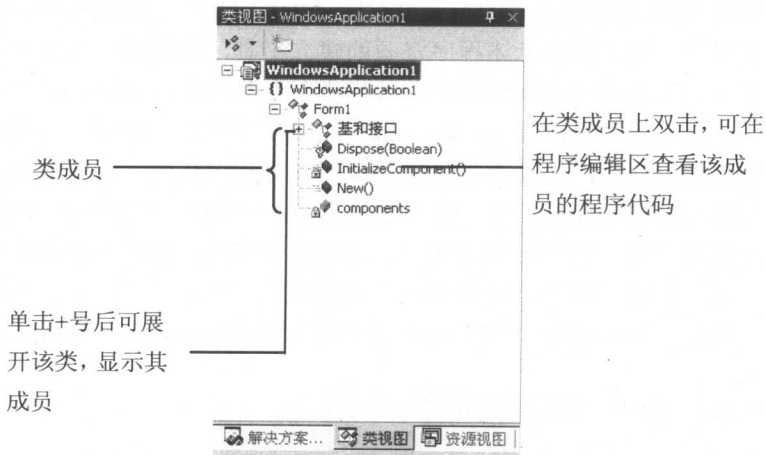


图 1.6

二是资源视图(Resource View)，如图 1.7 所示。切换至该视图里，将可查看程序所使用

的资源。

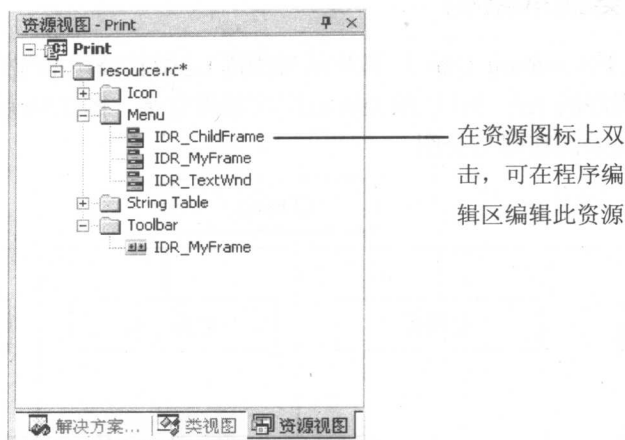


图 1.7

三是解决方案视图(File View)即解决方案资源管理器, 如图 1.8 所示。切换至该视图时, 可知道该项目中有哪些文件。在要查看的图标上双击, 将在左边的程序编辑区打开该项目。



图 1.8

3. 调试区

调试区在运行调试时, 将显示错误消息, 如图 1.9 所示。可以在要查看的错误消息上双击, 程序编辑区的画面便会切换至发生错误的程序片段。

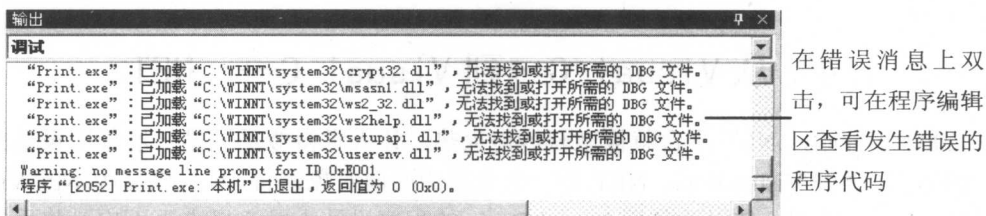


图 1.9

1.3.3 MFC 的类继承结构

MFC(Microsoft Foundation Class, 微软基础类库)是微软公司搭配在 Visual C++ .NET 中, 用于开发窗口程序的 AF。MFC 的类继承层次相当复杂, 图 1.10 就本书接下来将会介绍的相关类, 做一简要的整理与介绍。

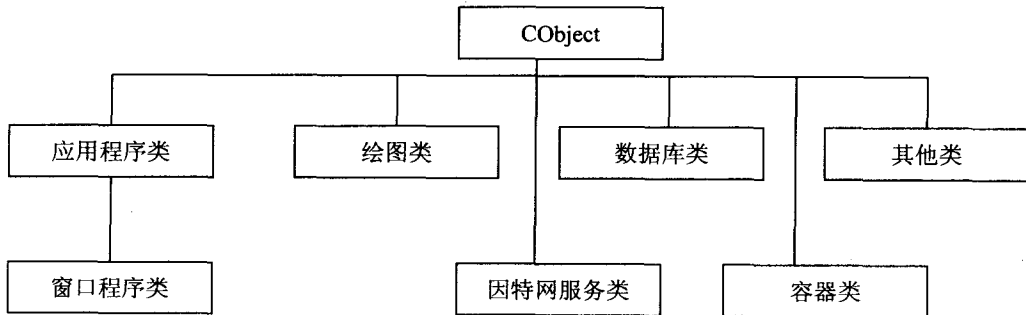


图 1.10

- **CObject:** CObject 类是 MFC 类库最基础的类。
- **应用程序类:** 应用程序类定义了建立应用程序可能需要利用的类, 如 CWinApp(窗口应用程序类)、CDocTemplate(文档模板类)、CDocument(文档类)等类。
- **窗口程序类:** 窗口程序类派生于应用程序类, 窗口程序类里定义了构造窗口所需的类, 如 CFrameWnd(窗口框架类)、CDialog(对话框类)、CView(浏览类)等类。
- **绘图类:** 用于绘图的类, 如 CDC(设备上下文或设备环境类)、CPen(画笔类)、CBrush(画刷类)等。
- **因特网服务类:** 用于建立网络连接的类, 可建立的连接种类有 FTP、Gopher、HTTP 三种。
- **数据库类:** 用于建立数据库连接, 取得数据库数据的类, 主要有 ODBC 数据库类、DAO 数据库类及 CRecordset 类(记录集类)。
- **容器类:** CList、CArray、CMap 等容器类。
- **其他类:** 除了上述几种类外, MFC 中还有异常处理类、文件操作类等。

除了继承自 CObject 的类外, MFC 还有几个本书将会使用到的独立类, 如用于数据储存的文件类 CArchive, 用于存储简单数据的 CPoint、CRect、CString 等。详细的类继承层次图, 请参考 Visual C++ 中有关 MFC 类库的说明。

1.4 从 Visual C++ 到 Visual C++ .NET

从 2002 年开始, .NET 平台就被微软炒得火热, 让人觉得似乎又是一场程序设计的大革命。微软一再强调 Visual Basic .NET 是一个全新的面向对象语言, 且与以往版本的 Visual Basic 相比有很大区别, 使以往使用 Visual Basic 的程序设计人员心里忐忑不安, 将面临重新学习的情境。再加上又出现了 .NET Framework、Common Language Runtime(CLR)等。技术三年一小修, 五年一大改, 真令人应接不暇。