

计·算·机·技·术·入·门·与·提·高·系·列



Microsoft
Visual C++ 6.0



Enterprise Edition

Visual C++ 6.0

入门与提高

r u m e n y u t i g a o

◆ 求是科技 王超龙 陈志华 编著

恒

毅

下

门

入

人民邮电出版社

计算机技术入门与提高系列



Visual C++ 6.0

入门与提高

r u m e n y u t i g a o

◆ 求是科技 王超龙 陈志华 编著

人民邮电出版社

图书在版编目(CIP)数据

Visual C++ 6.0 入门与提高/王超龙编著. —北京: 人民邮电出版社, 2002.7
ISBN 7-115-10399-2

I. V... II. 王... III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2002)第 045217 号

内容提要

本书全面讲解了 Visual C++ 6.0 应用程序开发方面的内容, 包括集成开发环境 IDE、C++ 语言的基本语法和面向对象技术、微软基础类库 (MFC)、MFC 应用程序框架、AppWizard 和 ClassWizard 的使用方法、各种对话框的使用方法、常用窗体控件和高级控件的详细用途及其使用方法、文件处理、文字和图形操作、网络编程和多线程技术等。

本书的特点是对每一个重要的知识点除了讲述原理外, 一般都配有对应的精彩实例演示, 这些实例从易到难, 深入浅出。本书适合 Visual C++ 初、中级读者系统地学习, 也可供软件开发人员参考。

计算机技术入门与提高系列 Visual C++ 6.0 入门与提高

- ◆ 编 著 求是科技 王超龙 陈志华
责任编辑 张立科
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67180876
北京汉魂图文设计有限公司制作
北京密云春雷印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 27.25
字数: 663 千字 2002 年 7 月第 1 版
印数: 1-6 000 册 2002 年 7 月北京第 1 次印刷

ISBN7-115-10399-2/TP · 2945

定价: 35.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

前 言

近年来面向对象技术发展迅猛，它被广泛地应用于计算机研究与应用的各个方面，如文件处理、操作系统设计、多媒体技术、网络与数据库应用等。用面向对象技术进行程序设计、开发软件已经成为一种时尚。这种技术从根本上改变了人们以往设计软件的思维方式，从而使程序设计者可以最大限度地摆脱繁琐的数据格式和冗长的研发过程，将精力集中在对象的设计和 research 上，大大提高了程序的开发效率。

Visual C++ 6.0 是 Microsoft 公司开发的面向 Windows 程序设计的一整套开发环境，Visual Studio 中的一种开发工具，和该套环境中的另一种开发工具 Visual Basic 一起成为 Microsoft 公司的面向 Windows 应用软件的可视化、面向对象的高效快速开发工具的两个拳头产品，目前在全球有大量的程序员正在运用这两种工具来完成面向多种应用的应用软件开发任务。

经过微软公司不断的完善，Visual C++ 在开发速度、程序执行效率、程序大小、与系统的集成性方面都有极大的提高，这使得它不但适合开发一般的应用软件包括数据库应用软件，而且更适合算法（图形图像、压缩等）的编程。另外，如果要进行驱动程序的开发，除了一些专用的驱动程序开发软件外，Visual C++ 也是一个不错的选择。

本书从基础知识出发，全面讲解了 Visual C++ 6.0 各方面的知识，包括 C++ 语言、MFC 类库、文件操作、图形图像编程、网络知识和网络编程、多线程知识和多线程编程等，由浅入深、从易到难、深入浅出，是一本学习 Visual C++ 6.0 的不可多得的好书。

本书共分 14 章。

第 1 章“进入 Visual C++ 世界”，讲述 Visual C++ 的基础知识，将使读者对 Visual C++ 有一个初步、大概的了解，初步认识 Visual C++ 6.0 的集成开发环境，并解答读者为何要使用 Visual C++ 6.0 的疑问。

第 2 章“C++ 编程基础”，讲述 Visual C++ 采用的编程语言——C++ 的相关知识，包括 C++ 类及其主要成员、C++ 的优点（如封装性、继承性和多态性等）、运算符重载特性以及 C++ 的一些编程规范。

第 3 章“第一个 Visual C++ 程序——鼠标追踪”，讲述了 Visual C++ 6.0 的集成开发环境中程序生成向导(AppWizard)的具体用法，并通过一个具体的简单实例演示了利用 AppWizard 完成程序设计的一般步骤。

第 4 章“微软类库(MFC)的基础”，讲述微软基础类库(MFC)的相关知识，并对一些重要类进行了讲解，如应用程序框架结构类、程序窗口类、图形打印类、文件类、Internet 类、集合类、异常和调试支持类等。

第 5 章“MFC 应用程序框架”，讲述了如何在 Visual C++ 6.0 中进行单文档和多文档程序的设计，并对 Visual C++ 的一大特色结构——文档/视图结构做了详细介绍。

第 6 章“类的设计魔棒——ClassWizard”，主要讲述 Visual C++ 中类的设计工具 ClassWizard 的使用，详细讲解了各个选项卡的作用，并演示如何利用 ClassWizard 实现类的常用操作，如增加新类、增减成员变量、增减成员函数、定位程序代码等。

第7章“程序菜单、工具栏和状态栏”，讲述了如何设计应用程序的菜单、工具栏和状态栏，并通过一个时钟程序做了具体演示。

第8章“输入输出与对话框”，讲述了如何响应 Windows 的基本消息，如键盘输入消息、鼠标输入消息、时钟触发消息，简述了对话框的使用方法，并分类讲解 Visual C++ 中各种对话框的使用，如消息对话框、文件对话框、颜色对话框、字体对话框、打印对话框等。

第9章“常用窗体控件”，讲述 Visual C++ 中常用窗体控件的使用方法，对每种控件都做了详细讲解，并配以具体的实例说明。

第10章“高级控件”，讲述 Visual C++ 中高级控件的使用方法，对多种高级控件做了详细讲解，并用一个综合实例演示了这些高级控件的使用。最后将高级控件的使用方法和常用窗体控件的使用方法作了一个详细比较。

第11章“文件的存取”，讲述了 Windows 编程中进行文件操作的各种 API 函数的使用，并对 MFC 中封装了这些函数的两个重要的文件类 CFile 类和 CStdioFile 类作了详细介绍并配以实例演示。

第12章“文字和图形操作”，讲述了 Windows 图形图像编程中的一些基本概念，如图形设备接口、设备上下文、映射模式等，然后通过具体实例演示了文本和图形图像输出中常用类 CFont、CPen、CBrush 的使用以及一些绘图 API 函数的用法。

第13章“网络编程”，讲述了网络编程的基础知识，如 TCP/IP 协议、UDP 协议等知识，介绍了 Windows Sockets 的特性及其库函数用法，讲述了 MFC 中用于网络编程的两个最重要的类 CAsyncSocket 和 CSocket 的成员函数及其用法，并配以具体实例做了演示。

第14章“多线程技术”，讲解了 Windows 的多线程机制以及在 Windows 中进行多线程编程常用的技术，如线程间通信和线程同步等。另外，对两种线程：工作者线程与用户界面线程也做了简单比较。

由于作者水平有限，书中难免有不足和疏忽之处，恳请读者和各位同仁批评指正。本书所附的代码可到 <http://www.ucbook.com/zvc.zip> 下载，作者的 E-mail 为 hl_zhang@163.com，欢迎联系。

编者

2002 年 7 月

目 录

第 1 章 进入 Visual C++ 世界	1
1.1 初识 Visual C++	1
1.2 集成开发环境	1
1.2.1 Visual C++ 主窗口	1
1.2.2 标题栏和工具栏	1
1.2.3 菜单	4
1.3 本章小结	10
第 2 章 C++ 编程基础	11
2.1 什么是 C++	11
2.2 C++ 类及其主要成员	12
2.2.1 定义 C++ 类	12
2.2.2 成员变量	13
2.2.3 构造函数	14
2.2.4 析构函数	14
2.2.5 虚函数	14
2.3 C++ 的优点	15
2.3.1 封装性	15
2.3.2 继承性	15
2.3.3 多态性	15
2.4 静态变量	16
2.5 运算符重载	17
2.6 C++ 编程规范	19
2.6.1 命名规范	19
2.6.2 注释规范	20
2.6.3 格式规范	20
2.7 本章小结	21
第 3 章 第一个 Visual C++ 程序——鼠标追踪	22
3.1 简介程序生成向导 (AppWizard)	22
3.1.1 启动 AppWizard	22
3.1.2 文件 (Files) 选项卡	22
3.1.3 工程 (Projects) 选项卡	23
3.1.4 工作区 (Workspaces) 选项卡	24
3.1.5 其他文档 (Other Documents) 选项卡	25
3.2 用 AppWizard 生成鼠标追踪程序	25

3.2.1	步骤 1——创建一个新的 Visual C++ 工程	26
3.2.2	步骤 2——确定程序名称和路径	26
3.2.3	步骤 3——选择程序类型	26
3.2.4	步骤 4——确定数据库来源	27
3.2.5	步骤 5——设置所支持的文档类型	28
3.2.6	步骤 6——设定程序特征项	29
3.2.7	步骤 7——确定工程样式	30
3.2.8	步骤 8——调整最终的类信息	31
3.2.9	步骤 9——生成程序初始框架	32
3.3	程序的框架说明	32
3.3.1	程序的组成文件	32
3.3.2	程序生成类及其简介	34
3.4	完善鼠标追踪程序	35
3.4.1	编写并保存代码	36
3.4.2	编译并运行程序	37
3.5	本章小结	38
第 4 章	微软类库 (MFC) 基础	39
4.1	应用程序框架结构类	39
4.1.1	应用程序及线程类	40
4.1.2	文档类	41
4.1.3	文档模板类	41
4.2	程序窗口类	42
4.2.1	窗口支持类	42
4.2.2	框架窗口类	42
4.2.3	对话框类	43
4.2.4	视图类	43
4.2.5	控件类	44
4.2.6	控制栏类	45
4.2.7	窗体分割类	45
4.3	图形打印类	46
4.3.1	设备输出类	46
4.3.2	图形工具类	46
4.4	文件类	47
4.5	Internet 类	48
4.5.1	ISAPI 类	48
4.5.2	Windows Sockets 类	48
4.5.3	Win32 Internet 类	49
4.6	集合类	50
4.6.1	数组类	50
4.6.2	列表类	50

4.6.3	数据对象映像类	51
4.7	异常和调试支持类	52
4.8	本章小结	52
第 5 章	MFC 应用程序框架	54
5.1	最简单的单文档程序——“新年好!”	54
5.1.1	应用程序对象	56
5.1.2	主框架窗口对象	62
5.1.3	文档对象	66
5.1.4	视图对象	69
5.1.5	文档模板	72
5.2	多文档程序框架	73
5.2.1	与单文档程序的框架区别	74
5.2.2	特有类——子框架窗口类	74
5.3	文档/视图结构	76
5.3.1	文档与视图的关系	76
5.3.2	两者相互作用的函数	77
5.3.3	视图类操作文档对象——CView::GetDocument()	77
5.3.4	视图类操作文档对象——CView::OnUpdate()	78
5.3.5	视图类操作文档对象——CView::OnInitialUpdate()	78
5.3.6	文档类操作视对象——CDocument::UpdateAllView()	78
5.3.7	文档类影响视对象——CDocument::DeleteContents()	78
5.4	具有文档/视图结构的程序示例	79
5.4.1	单视图程序	79
5.4.2	双视图程序	80
5.5	本章小结	82
第 6 章	类的设计魔棒——ClassWizard	83
6.1	初识 ClassWizard	83
6.1.1	启动 ClassWizard	83
6.1.2	消息映射 (Message Maps) 选项卡	84
6.1.3	成员变量 (Member Variables) 选项卡	85
6.1.4	自动控制 (Automation) 选项卡	86
6.1.5	ActiveX 控件事件 (ActiveX Events) 选项卡	87
6.1.6	类信息 (Class Info) 选项卡	87
6.2	ClassWizard 的常用功能与操作	88
6.2.1	增加新类	90
6.2.2	增减成员变量	91
6.2.3	增减成员函数	92
6.2.4	定位程序代码	92
6.3	本章小结	95
第 7 章	程序菜单、工具栏和状态栏	96

7.1	设计和使用程序菜单	96
7.1.1	可视化设计菜单	97
7.1.2	响应菜单消息	99
7.1.3	程序控制菜单	100
7.2	设计和使用程序工具栏	102
7.2.1	可视化设计工具栏	103
7.2.2	程序加载工具栏	103
7.2.3	程序控制工具栏	104
7.2.4	改变工具栏风格	105
7.3	设计和使用程序状态栏	106
7.3.1	加载状态栏	106
7.3.2	设置状态栏参数	106
7.4	综合示例——小小时钟	107
7.5	本章小结	116
第 8 章	输入输出与对话框	117
8.1	响应 Windows 基本消息	117
8.1.1	键盘输入消息	117
8.1.2	鼠标输入消息	120
8.1.3	时钟触发消息	121
8.2	显示提示信息	124
8.2.1	AfxMessageBox 函数	124
8.2.2	CWnd::MessageBox 函数	125
8.3	对话框概述	126
8.3.1	对话框的分类	126
8.3.2	为程序增加对话框类	126
8.3.3	为对话框增加控件	127
8.3.4	显示对话框	127
8.3.5	存取对话框数据	128
8.3.6	常用对话框操纵函数	128
8.3.7	对话框示例——登录窗口	129
8.4	通用对话框	131
8.4.1	文件对话框	131
8.4.2	颜色对话框	132
8.4.3	字体对话框	133
8.4.4	打印对话框	134
8.5	本章小结	135
第 9 章	常用窗体控件	136
9.1	控件初步	136
9.1.1	向对话框添加控件	136
9.1.2	设置控件属性	138

9.1.3 将控件与变量关联	139
9.2 按钮控件及范例	153
9.2.1 下压按钮控件及范例	153
9.2.2 成组框控件简单介绍	161
9.2.3 单选按钮控件及范例	161
9.2.4 复选按钮控件及范例	166
9.3 静态控件及范例	170
9.4 编辑控件及范例	177
9.5 列表控件及范例	184
9.6 组合框控件及范例	196
9.7 滚动控件及范例	207
9.8 本章小结	211
第 10 章 高级控件	212
10.1 Win32 高级控件	212
10.1.1 传统 Windows 标准控件的通知消息	212
10.1.2 Win32 控件的通知消息	213
10.2 数字滚选按钮控件	215
10.3 滑尺控件	218
10.4 进度条控件	220
10.5 树形视图控件	221
10.6 列表视图控件	225
10.7 动画控件	230
10.8 标签控件	233
10.9 范例——高级控件的使用	237
10.10 控件使用技术总结	250
10.10.1 所有的控件都是窗口	250
10.10.2 控件的创建方法	250
10.10.3 访问控件的方法	251
10.10.4 控件及控件对象的删除	252
10.10.5 控件通知消息	252
10.11 本章小结	252
第 11 章 文件的存取	253
11.1 API 函数及其范例	253
11.1.1 文件的打开和关闭	253
11.1.2 文件的同步和异步读写	256
11.1.3 文件加锁	265
11.1.4 文件的查找	266
11.1.5 文件的其他操作	270
11.1.6 串口的操作	271
11.2 CFile 类及其范例	280

11.2.1	构造类成员函数	281
11.2.2	存取类成员函数	285
11.2.3	定位类成员函数	285
11.2.4	锁定类成员函数	286
11.2.5	状态操作类成员函数	287
11.2.6	静态成员函数	289
11.3	CStdioFile 类及其范例	290
11.3.1	成员变量	290
11.3.2	构造函数	290
11.3.3	存取成员函数	291
11.4	本章小结	292
第 12 章	文字和图形操作	293
12.1	图形设备接口和设备上下文	293
12.2	映射模式及其范例	295
12.2.1	映射模式简介	295
12.2.2	映射模式单位转换范例	297
12.2.3	自定义的坐标系统及其范例	304
12.3	文本输出及其范例	309
12.3.1	CFont 类	309
12.3.2	LOGFONT 结构	310
12.3.3	字体和文本函数	312
12.3.4	枚举系统中所有字体以及使用字体对象范例	313
12.3.5	创建特殊的字体效果	321
12.4	输出图形及其范例	330
12.4.1	与绘图相关的简单数据类	330
12.4.2	CPen 类及其范例	333
12.4.3	CBrush 类及其范例	337
12.4.4	绘制点、线、弧和圆及其范例	341
12.5	本章小结	346
第 13 章	网络编程	347
13.1	TCP/IP 简介	347
13.1.1	IP (Internet Protocol) 协议	347
13.1.2	TCP (Transmission Control Protocol) 协议	352
13.1.3	UDP (User Datagram Protocol) 协议	354
13.2	Windows Sockets 简介	354
13.2.1	什么是 Windows Sockets	354
13.2.2	Windows Sockets 的适用范围	355
13.2.3	Windows Sockets 的版本	355
13.2.4	使用 Windows Sockets 进行网络编程	356
13.2.5	Windows Sockets 1.1 库函数概览	363

13.2.6	Windows Sockets 2 的扩展特性	367
13.3	Windows Sockets 类	378
13.3.1	CAsyncSocket 类和 CSocket 类	378
13.3.2	范例——利用 CSocket 进行面向连接的通信	381
13.4	本章小结	386
第 14 章	多线程技术	387
14.1	Windows 多线程机制	387
14.1.1	线程的基础知识	387
14.1.2	何时创建线程	388
14.1.3	何时不能创建线程	389
14.1.4	编写一个线程函数	390
14.1.5	线程的创建	390
14.1.6	线程的终止	395
14.1.7	暂停和恢复线程的运行	397
14.1.8	切换到另一个线程	398
14.1.9	线程的运行时间	399
14.1.10	线程的优先级	400
14.2	线程间通信	401
14.2.1	使用全局变量通信	401
14.2.2	使用用户自定义消息通信	403
14.2.3	使用 Event 对象通信	406
14.3	线程同步	410
14.3.1	使用临界区 (Critical Section)	410
14.3.2	使用互斥对象 (Mutex)	416
14.3.3	使用信号量 (Semaphore)	417
14.4	工作者线程与用户界面线程	421
14.5	本章小结	422

第 1 章 进入 Visual C++ 世界

1.1 初识 Visual C++

Visual C++ 是 Microsoft 公司的重要产品之一——Developer Studio 工具集的重要组成部分。它用来在 Windows（包括 Windows 95、Windows 98、Windows NT、Windows 2000 等）环境下开发应用程序，是一种功能强大、行之有效的可视化编程工具。Visual C++ 以可视化技术为基础，以 C++ 语言为蓝本，以众多的集成工具为骨架，在计算机领域的诸多方面都发挥着重要的作用。

Visual C++ 提供了简洁实用的开发环境，集成的工具集能够让用户高效率地开发应用程序。

Visual C++ 提供了 MFC 类库，一般性工作可以全部交给 Visual C++ 来完成，用户只要在其基础上做出自己想要实现的功能即可。与 Turbo C 等工具相比，Visual C++ 完成目标（尤其对于图形界面的程序）所花费的时间要少得多。

1.2 集成开发环境

Visual C++ 是一种集成开发环境（Integrated Development Environment，IDE），它拥有友好的可视化界面。在 Visual C++ 中进行的操作，包括建立、打开、浏览、编辑、保存、编译等。下面一一介绍各窗口、菜单和工具栏。

1.2.1 Visual C++ 主窗口

Visual C++ 安装完毕，在 Windows【开始】菜单的【程序】菜单中可以找到【Microsoft Visual Studio 6.0】子菜单项，在其弹出的子菜单中单击【Microsoft Visual C++ 6.0】，即可启动 Visual C++。进入集成开发环境的主窗口，其界面形式如图 1-1 所示。

缺省的 IDE 主窗口自上而下依次为：标题栏【Title bar】、菜单【Menu】、工具栏【Tool bar】、工作区窗口（左）【Workspace】、代码编辑窗口（右）【Edit】、输出窗口【Output】、状态栏【Status bar】。

1.2.2 标题栏和工具栏

标题栏位于主窗口的最上方，它显示了应用程序的标题名称。

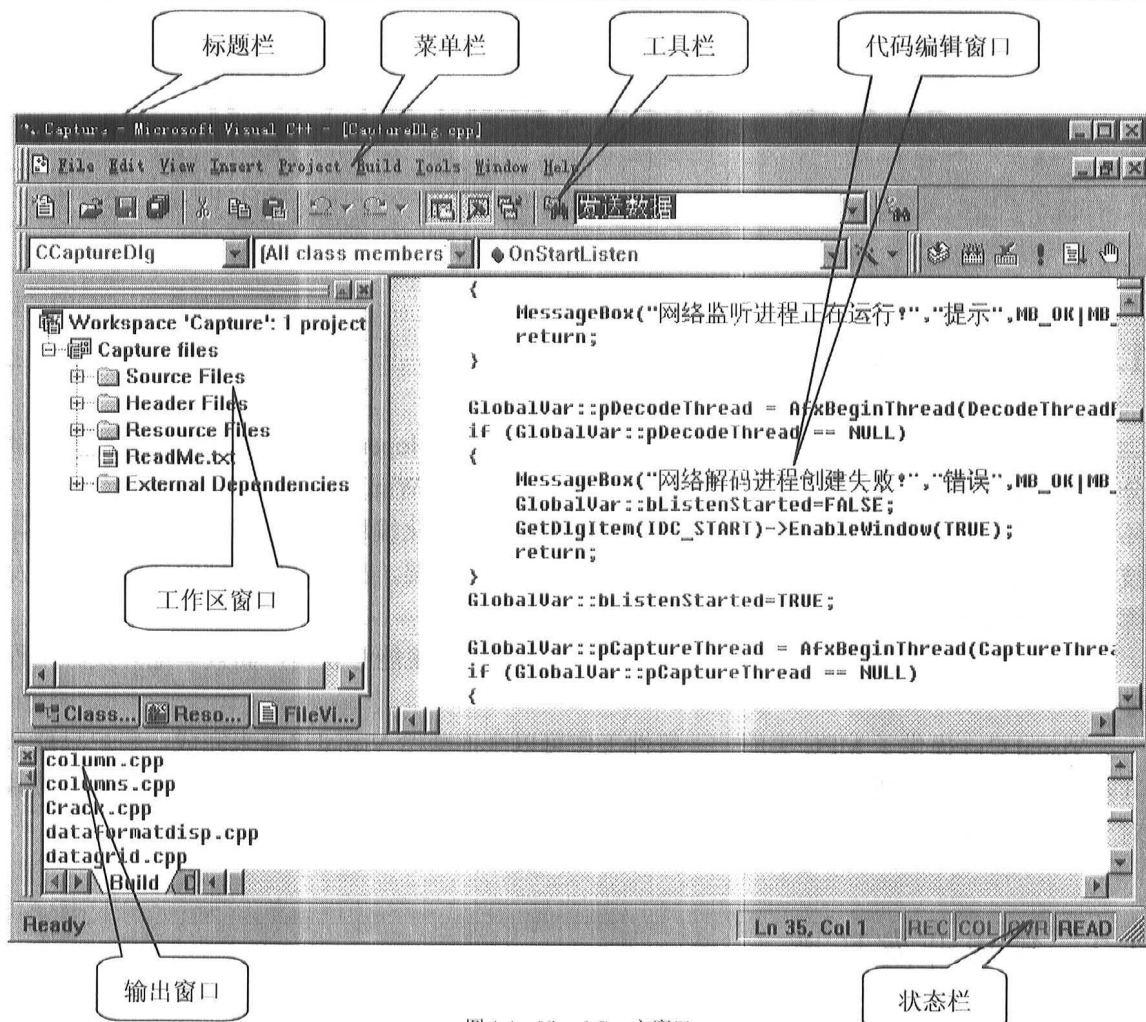


图 1-1 Visual C++主窗口

工具栏是 Visual C++ 集成开发环境的重要组成部分, 包括 Standard 工具栏、Build 工具栏、Resource 工具栏、WizardBar 工具栏、Debug 工具栏等。

Standard 工具栏用于建立项目工作区及项目, 如图 1-2 所示。

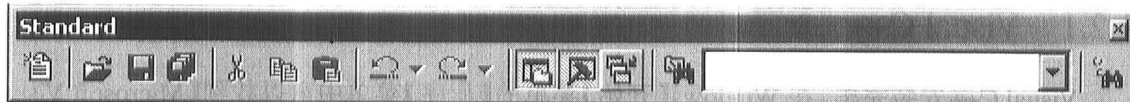


图 1-2 Standard 工具栏

将鼠标停在其中一个图标上, 就能出现关于该图标功能的说明文字。下面分别介绍:

- New Text File: 创建新的文本文件。
- Open: 打开文档。
- Save: 保存文档。
- Save All: 保存所有打开的文档。
- Cut: 剪切选定的内容。
- Copy: 复制选定的内容。
- Paste: 粘贴选定的内容。

- Undo: 取消上一步操作。
- Redo: 重复上一步操作
- Workspace: 显示/隐藏工作区窗口。
- Output: 显示/隐藏输出窗口。
- Windows list: 窗口管理。
- Find in Files: 在多个文件中搜索。
- Find: 查找字符串。
- Search: 搜索联机文档。

Build Minibar 工具栏用于编译代码、连接目标文件和调试运行程序，如图 1-3 所示。



图 1-3 Build Minibar 工具栏

各图标功能介绍如下：

- Compile: 编译文件。
- Build: 建立项目。
- Stop Build: 停止建立。
- Execute Program: 运行程序。
- Insert/Remove Breakpoint: 插入或删除断点。

Build 工具栏比 Build Minibar 多两项功能：选择项目和配置，如图 1-4 所示。

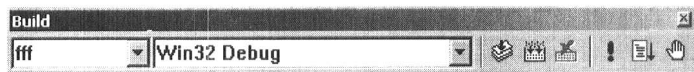


图 1-4 Build 工具栏

各图标功能介绍如下：

- Select Active Project: 选择活动项目。
- Select Active Configuration: 选择活动配置 (Debug 或 Release)。

Debug 工具栏用于调试程序，如图 1-5 所示。

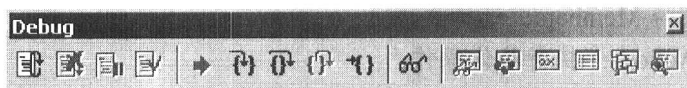


图 1-5 Debug 工具栏

各图标功能介绍如下：

- Restart: 重新执行程序，处于调试状态。
- Stop Debugging: 停止调试程序。
- Break Execution: 停止运行程序。
- Apply Code Change: 应用改变后的代码。
- Show Next Statement: 显示下一条语句。
- Step Into: 进入函数内部单步执行。
- Step Over: 单步执行 (跳过函数)。

- Step Out: 跳出函数。
- Run to Cursor: 运行到光标处。
- Quick Watch: 快速察看调试状态。
- Watch: 察看变量信息。

Resource 工具栏用于控制资源的创建, 如图 1-6 所示。

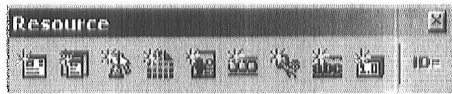


图 1-6 Resource 工具栏

各图标功能介绍如下:

- New Dialog: 创建新的对话框。
- New Menu: 创建新的菜单。
- New Cursor: 创建新的光标。
- New Icon: 创建新的图标。
- New Bitmap: 创建新的位图。
- New Toolbar: 创建新的工具栏。
- New Accelerator: 创建新的快捷键。
- New String Table: 创建新的字符串表。
- New Version: 创建新的版本信息。
- Resource Symbols: 获得资源符号。

WizardBar 工具栏用于进行类的操作, 如图 1-7 所示。



图 1-7 WizardBar 工具栏

各图标功能介绍如下:

- WizardBar C++ Class: 选择类。
- WizardBar C++ Filter: 选择过滤事件。
- WizardBar C++ Members: 选择类成员。
- WizardBar Actions: 执行具体操作。

Visual C++ 6.0 还提供了其他几种工具栏, 如 Browse、Database、Edit、ATL、Source Control 等, 它们的功能这里就不一一说明了。

1.2.3 菜单

菜单是 Visual C++ 集成开发环境的另一重要组成部分, 包括【File】菜单、【Edit】菜单、【View】菜单、【Insert】菜单、【Project】菜单、【Build】菜单、【Debug】菜单、【Tools】菜单、【Windows】菜单和【Help】菜单。它们几乎包含了常用的所有操作功能。

【File】菜单用于文件的相关操作。打开 File 菜单, 可以看到如下常用命令 (如图 1-8 所示):

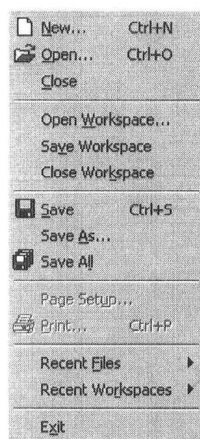


图 1-8 File 菜单

各菜单项功能介绍如下：

- New: 创建新工程或新文件
- Open: 打开现有文件
- Close: 关闭文件
- Open Workspace: 打开现有工作
- Save Workspace: 保存当前工作
- Close Workspace: 关闭当前工作
- Save: 保存当前文件
- Save As: 文件另存为
- Save All: 保存所有已打开的文件
- Page Setup: 设置文件页面
- Print: 打印文件
- Recent Files: 近期的文件
- Recent Workspace: 近期的工作
- Exit: 退出

【Edit】菜单用于代码编辑，如图 1-9 所示，很多功能与工具栏中的图标功能对应。

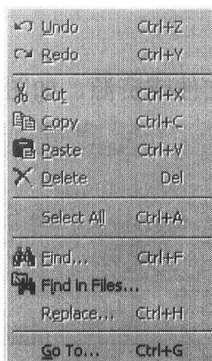


图 1-9 Edit 菜单

各菜单项功能介绍如下：