

“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

编程新概念丛书 7

MFC 程序设计

Using Visual C++ 6.0

北京希望电子出版社 总策划

唐克 编 著



北京希望电子出版社
Beijing Hope Electronic Press
www.bhep.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

编程新概念丛书 7

MFC 程序设计

Using Visual C++ 6.0

北京希望电子出版社 总策划

唐克 编 著



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 MFC 程序设计的实用指导书。全书通过 21 个精彩典型的实例，详细介绍了 MFC 编程的规范、方法和技巧。

全书共分为三篇，初级篇、中级篇和高级篇。初级篇包括：Visual C++集成开发环境和 Win32 编程基础，共 2 章内容。中级篇包括：MFC 应用程序基本框架、纵览 MFC、MFC 的消息处理模式、窗口与视图、深入文档/视图结构和 MFC 与对话框共 6 章内容。高级篇包括：MFC 与控件栏、属性表与向导、MFC 与 DLL、MFC 与 Internet 和调试应用程序共 5 个专题。

本书概念清晰、内容丰富、层次分明、结构合理，典型实例与知识要点紧密结合，边讲边练，学习轻松，上手容易。本书面向 MFC 初、中级用户，高校相关专业师生和各类 MFC 初、中级培训班。

本书使用中的技术问题请与 tookoo@eyou.com 联系。

本书源代码可在 www.b-xr.com 免费下载。

- 系 列 书 名 : “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
编程新概念丛书(7)
- 书 名 : MFC 程序设计—Using Visual C++ 6.0
- 总 策 划 : 北京希望电子出版社
- 文 本 著 者 : 唐克
- 责 任 编 辑 : 杨如林
- 出版、发行者 : 北京希望电子出版社
- 地 址 : 北京市海淀区知春路甲63号卫星大厦三层 100080
网址: www.bhp.com.cn
E-mail: zwb@bhp.com.cn
电 话 : 010-62520290,62521724,62528991,62630301,62524940,62521921,82610344
(发行) 010-82675588-202(门市) 010-82675588-501,82675588-201(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 张月岭
- 文 本 印 刷 者 : 北京媛明印刷厂
- 开本 / 规格 : 787毫米×1092毫米 1/16 29印张 688千字
- 版次 / 印次 : 2002年10月第1版 2002年10月第1次印刷
- 印 数 : 0001-3000册
- 本 版 号 : ISBN 7-900118-60-8
- 定 价 : 46.00元

说明: 凡我社产品如有缺损, 可执相关凭证与本社调换。

前言

首先需要声明，本书不是 MFC 知识的百科全书，如果您想看这方面的资料，MSDN 也许是最好的选择。您是一个 MFC 程序员吗？您希望成为一个 MFC 程序员吗？那太好了，本书肯定适合您。如果您不愿意循规蹈矩地编写 MFC 程序，希望“标新立异”，编写具有自己特色的程序，本书绝对可以成为您的最佳选择之一。

毋庸置疑，C++ 是当今世界上应用最广泛的编程语言，这一切主要得益于当年深入人心的 C 语言。在 C++ 逐渐强大的同时也伴随着编程复杂度的大大增加，要想成为 C++ 开发者比成为 VB 或 Java 开发者需要付出更多的努力。由于操作系统的影响，Windows 环境下的 C++ 编程无疑更困难一些。幸好，Microsoft 开发的 VC 系列产品中提供了一整套减小开发强度的措施，其中最关键、最振奋人心的一环就是 MFC——微软基础类库。

MFC 是 Visual C++ 的核心。尽管我们知道，使用 Visual C++ 进行编程不一定要使用 MFC，Visual C++ 提供了许多的编程手段；使用 MFC 也不一定就要使用 Visual C++，许多开发工具都提供了对 MFC 的支持（例如 Borland C++）。但事实上，在大多数情况下，我们提到 MFC 指的就是使用 Visual C++ 进行开发，甚至也可以说大多数情况下，提到 Visual C++ 指的就是 MFC。

□ 为什么要使用 MFC

标准的 C++ 语言几乎没有附带任何库，仅提供了一些语言规则。开发者要么自己编写出适宜的类库，要么使用别人已经编好的。实际上，您可以把 MFC 当作一个第三方（Microsoft）提供的 C++ 标准库。

一个完整的 Windows 应用程序首先要具有良好的用户界面，这要涉及窗体、对话框、控件、按钮、菜单等多方面的知识等，虽然您可以通过调用 Windows 的 API 函数来实现这一切，但有过 API 编程经验的人都很清楚，这是一个极为痛苦的过程，代码复杂冗长。而 MFC 则可以大大简化这一过程，它位于任何 Windows API 之上，能够使开发者的工作更加高效和轻松。同时它还提供了所有一般 C++ 编程的优点，因此 MFC 具有良好的通用性和可移植性。

MFC 借助 Visual C++ 提供的工具使程序设计更加清晰，例如应用程序向导等，能帮助您编写出那些真正“有用”的代码。MFC 中还提出了许多独特的思想，例如借助 MFC 消息映射机制，您可以方便地实现对消息的处理。更令人兴奋的是，MFC 利用 C++ 的封装功能使开发者摆脱了 Windows 中各种句柄的困扰，而只需要面对 C++ 中的对象，这样使开发更接近编程语言而远离系统。

这么优秀的工具摆在面前，有什么理由不去尽情使用呢？

□ MFC 的局限

和其他任何事物相同，MFC 也有它的局限性。虽然您有可能还没意识到，但 MFC 对开发人员的限制主要表现在以下方面：

- MFC 有一个独特的结构，其许多部分都不得不依赖该结构；
- MFC 的工具只适用于特定的程序，在创建其他程序时无法使用它们；

- MFC 类库版本更新较慢（至少目前是这样），不能紧跟时代发展的步伐。

本书介绍的许多内容就是为了帮助您打破这种限制，虽然这有时意味着大量的工作，但如果您知道该如何去做，一切也会变得很容易。在任何情况下，只要您想做，就可以用 MFC 达到自己的目的。

□ 本书适用对象

本书适用于所有有志于使用 MFC 进行程序开发的人们，但为了阅读方便，您需要具备以下基础：

- 能够熟练地使用 Windows 操作系统；
- 有 C/C++ 的相关编程经验；
- 最好使用过可视化编程工具。

□ 本书内容特点

1. 不求面面俱到，但求通俗易懂

希望通过一本书来涵盖 MFC 编程的所有内容显然是一件不现实的事情。至少，本书志不在此。作者从 MFC 中选取了最基本、最核心、最具特点的部分组成了本书的内容框架，并结合自己的编程经验对其进行分析、探讨，给出了一些独到的见解。

为了让您能够尽快地掌握 MFC 编程，进行 Windows 程序开发，本书在内容安排方面尽可能地做到从易到难，由浅入深。希望您能够通过本书的学习，逐步进入 MFC 编程的殿堂，这正是本书写作目的之所在。

2. 突出内容重点，兼顾高级应用

本书对 MFC 类库进行了重点介绍，并以其为主线，对一些基础、关键的内容，如可视化操作、文档/视图结构、对话框与控件等进行了详细介绍。这些内容是使用 MFC 进行编程的基础，是重中之重。

为了您学习提高的需要，本书把 MFC 的一些高级编程知识分成不同的主题进行了详细讲解。通过对这一部分的认真学习，相信您可以达到一个更高的层次。

3. 内容结合实例，方便学习开发

纸上谈兵是没有意义的，也是不具有说服力的。本书中介绍的所有重要内容和关键技术都有具体的实例与之相配合。为了初学者更好地理解 MFC 编程的基本思想，没有强求例题达到太多的难度，但都有很强的代表性，并提供了很好的编程模板。读者可以在充分理解了相关内容的基础上，对其进行丰富、扩充，编写出真正属于自己的 MFC 程序。

□ 本书章节安排

本书章节是按照一般读者的学习进程安排的，先介绍基础知识，然后逐步深入，由浅入深。全书共分三篇——初级篇、中级篇、高级篇。

1. 初级篇

- 第 1 章详细讲述了 Visual C++ 集成开发环境。“工欲善其事，必先利其器”，熟悉 Visual C++ 的集成开发环境（IDE）可以让您的程序设计收到事半功倍的效果。
- 第 2 章介绍了 Win32 编程的基础知识，重点阐述了进程、线程、动态链接库等基

本概念。这是以后各章学习的基础。

2. 中级篇

- 第3章详细介绍了MFC应用程序向导的使用,并对向导自动生成MFC程序的主要框架进行了分析,介绍了程序中主要的类库和文件的作用与用途。
- 第4章介绍了MFC的演变历史与MFC类库层次,并对MFC7.0的新增功能进行了详细阐述,使读者对MFC类库有一个整体而全面的了解。
- 第5章介绍了MFC处理消息的方法与流程。理解这部分内容是进行MFC编程的关键。
- 第6章介绍了MFC的可视化操作,包括对程序中菜单的处理;不同方式的绘图操作等内容。
- 第7章详细讨论了文档、视图、文档模板以及三者之间的关系,并通过对CArchive类和CFile类深入介绍了文档序列化和文件的相关操作。
- 第8章详细介绍了对话框与控件的相关知识,在介绍消息对话框、模式对话框、非模式对话框、通用对话框、对话框控件的同时,介绍了MFC为它们提供的类库支持和操作函数,这部分内容是本书的重要组成部分。

3. 高级篇

高级篇以专题的形式介绍了五个方面的内容。

- 第9章介绍了工具栏、状态栏、对话框、Rebar等控件栏的使用。在应用程序中控件栏的处理可以极大地丰富程序界面。
- 第10章中介绍了一类特殊的对话框——属性表,以及一种特殊的属性表——向导的相关知识。在MFC程序中,属性表与向导的使用非常简单,难的是找到一个不使用它们的理由。
- 第11章中介绍了动态连接库方面的相关知识,以及使用MFC编写DLL程序的方法。
- 第12章对Internet程序设计进行了详细的介绍,内容包括WinSock, WinInet, MAPI以及ISAPI等多方面的内容。这是作者最感兴趣的内容之一,因此,这部分内容在整本书中占据了很大的比例,是本书的另一个重点内容。
- 第13章介绍了应用程序调试方面的相关内容,帮助读者排忧解难,更地地编写程序。

4. 附录

- 本书的附录对C++语言面向对象的相关内容进行了概括性介绍,如果您对书中程序代码的语法、格式有所疑问的话,随时都可以查阅这部分内容。

□ 学习指南

由于Visual C++是一个比较复杂的开发工具,如果您在第一次阅读时对本书的某些章节理解不够透彻,可以先跳过相关章节,继续阅读。等到对Visual C++有了更深一步了解后,再阅读相关部分将有助于对本书内容的理解。在阅读本书的过程中要坚持学习与编程相结合的原则,只有通过编程实践,才能把知识融会贯通,变为自己的财富。

本书主要由唐克编写。在本书的编写过程中,得到了单珊、张威、唐志平、朱玉芬、

唐军、单纪宽、赵凤楼、芦海涛、孙鑫、谢茂强、全雄文、孙旭、吴水华、郭兰、陈丽华、黄克颖、徐浩庆、王勇志、胡松、向琨等人的帮助与支持，最后还要特别感谢刘庆文、王瑛、刘丽蓉，没有他们，就不可能有这本书稿的顺利完成。

由于作者水平有限，错误和疏漏之处，恳请广大读者批评指正。

作者

目 录

第 1 部分 初 级 篇

第 1 章 Visual C++集成开发环境.....	3
1.1 开发环境	4
1.2 IDE 菜单介绍.....	4
1.3 资源编辑器	12
1.4 Visual C++工具集	16
1.5 本章小结	22
第 2 章 Win 32 编程基础.....	23
2.1 Windows 操作系统的历史与特点	24
2.2 Win32 编程思想.....	26
2.3 匈牙利命名法	31
2.4 用 API 编写一个 Windows 程序.....	33
2.5 本章小结	37

第 2 部分 中 级 篇

第 3 章 MFC 应用程序基本框架.....	41
3.1 使用 AppWizard 生成一个应用程序.....	42
3.2 程序中的文件	49
3.3 MFC 程序的主要成员类.....	52
3.4 本章小结	59
第 4 章 纵览 MFC.....	60
4.1 MFC 的演变历史.....	61
4.2 MFC 类库层次.....	62
4.3 MFC 的全局函数与宏	71
4.4 MFC 数据类型	72
4.5 MFC7.0 新增功能.....	73
4.6 本章小结	75
第 5 章 MFC 的消息处理模式.....	76
5.1 消息的分类	77
5.2 消息映射	79
5.3 消息的处理函数	81
5.4 消息传递	88
5.5 本章小结	92

第6章 MFC的可视化技术——窗口与视图.....	93
6.1 菜单与加速键	94
6.2 基本的菜单操作	95
6.3 MFC的高级菜单操作.....	99
6.4 设备上下文与图形设备接口	111
6.5 设备上下文类——CDC	111
6.6 CGdiObject 及其派生类.....	121
6.7 映射模式	132
6.8 编程实例	135
6.9 本章小结	154
第7章 深入文档/视图结构.....	155
7.1 文档/视图结构简介	156
7.2 文档/视图结构的核心类	157
7.3 创建文档、窗口与视图	165
7.4 CArchive 类与序列化.....	167
7.5 CFile 类与文件	171
7.6 编程范例	173
7.7 本章小结	179
第8章 MFC 与对话框.....	180
8.1 消息对话框	181
8.2 对话框的基本知识	183
8.3 CDialog 与对话框.....	184
8.4 对话框数据交换和验证机制	188
8.5 对话框控件	192
8.6 通用对话框	228
8.7 对话框例程	245
8.8 本章小结	262

第3部分 高级篇

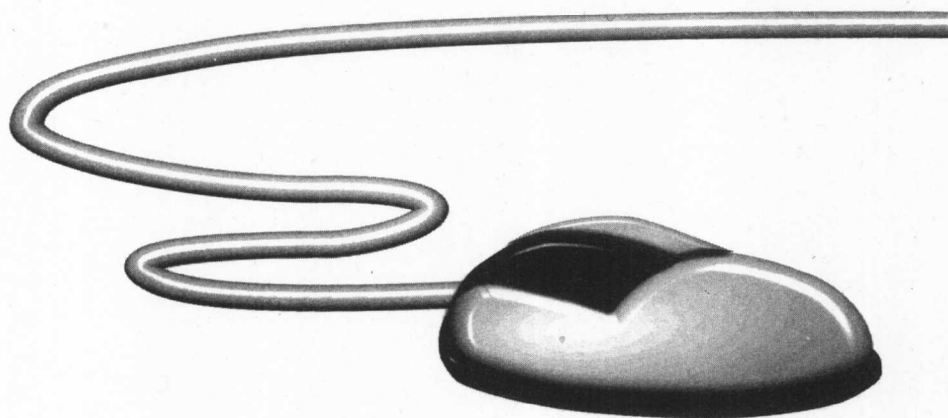
第9章 MFC 与控件栏.....	265
9.1 控件栏概述	266
9.2 工具栏	266
9.3 状态栏	272
9.4 对话框	275
9.5 Rebar.....	277
9.6 例程	280
9.7 本章小结	286

第 10 章	属性表与向导	287
10.1	属性表和属性页	288
10.2	创建属性表	288
10.3	属性表的相关操作	289
10.4	向导	294
10.5	例程	296
10.6	本章小结	303
第 11 章	MFC 与 DLL	304
11.1	DLL 基本理论	305
11.2	从 DLL 导出	306
11.3	导入到应用程序	310
11.4	程序链接	310
11.5	DLL 的入口点—DllMain 函数	311
11.6	Win32 DLL	313
11.7	MFC DLL	314
11.8	编程范例	318
11.9	本章小结	331
第 12 章	MFC 与 Internet	332
12.1	Windows Socket 编程	333
12.2	MAPI	346
12.3	WinInet	350
12.4	ISAPI	364
12.5	CHtmlView 类	373
12.6	例程	376
12.7	本章小结	407
第 13 章	调试应用程序	408
13.1	程序调试基本思想	409
13.2	Visual C++ 调试工具	410
13.3	高级调试技术	417
13.4	本章小结	425
附录 A	C++ 面向对象程序设计	426
A.1	类	427
A.2	继承性	436
A.3	多态性	440
A.4	小结	446



第1章 Visual C++集成开发环境

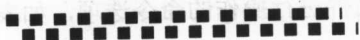
第2章 Win 32 编程基础



第1章 Visual C++ 集成开发环境

“工欲善其事，必先利其器”，熟悉 Visual C++ 的集成开发环境 (IDE) 可以使程序设计收到事半功倍的效果。

Visual C++ 6.0 继承了 Visual C++ 以前版本的主要特性，为用户提供了一套优秀的可视化集成开发环境。这其中主要包括文本编辑器、资源编辑器、项目创建工具、Debugger 调试器等等。用户可以在集成开发环境中创建新项目、打开项目，新建、打开、编辑文件，编译、链接、运行、调试应用程序。





1.1 开发环境

从程序菜单中选择运行 Visual C++6.0，即可进入 Visual Studio 集成开发环境。每次启动 Visual C++6.0 后，都会出现与图 1-1 类似的界面。

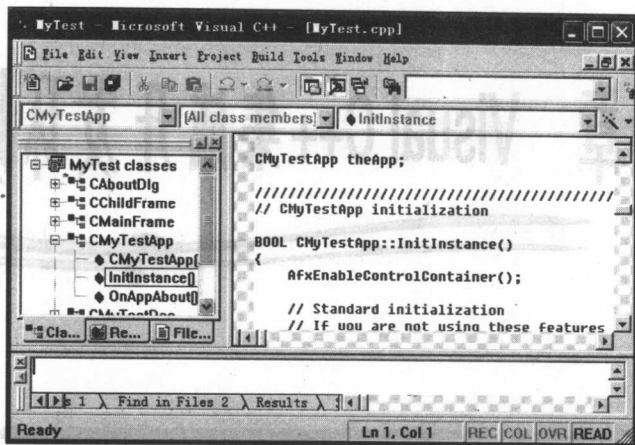


图 1-1 Visual C++6.0 集成开发环境

Visual C++6.0 集成开发环境包括标题栏、菜单栏、工具栏、工作区、源代码窗口、输出窗口等。主框架窗口的最上端为标题栏，用来显示应用程序名和当前打开的文件名。标题栏的下方通常是菜单栏和工具栏。工具栏的下面是工作区、源代码窗口、输出窗口等。在工作区中可以通过文件、类、资源三种方式来了解项目的有关信息，源代码窗口中显示当前打开的文件，可以通过它来编辑项目的源代码文件，输出窗口主要用于输出程序的编译、链接和调试信息。

1.2 IDE 菜单介绍

在上节中对集成开发环境的主框架窗口进行了介绍，在本节中将要具体介绍各菜单的功能。菜单栏由多个顶级菜单项组成，用户可以通过鼠标或者键盘来选择顶级菜单项。选中菜单项后，即弹出下拉式菜单，每个下拉式菜单又有多个菜单项，用于执行相应的功能和命令。

有些菜单项右面有相应的加速键（如 Save All 的加速键是 Ctrl+S），按加速键组合后能直接响应菜单命令，可以避免进入多层菜单的麻烦，从而方便快捷地执行菜单功能。有些菜单项后面带有省略号[...],表示选择该菜单项后将弹出一个对话框；有些菜单项后面带有三角符号▶,表示鼠标指向该菜单项时将弹出子菜单；有些菜单项被灰色显示，表示这些菜单项当前不可用。

1.2.1 File 菜单

File（文件）菜单中主要包括编程过程中对文件进行操作的命令选项，如表 1-1 所示。



表 1-1 File 菜单

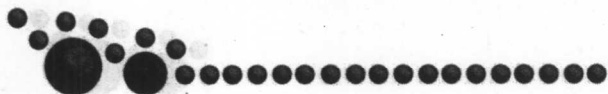
菜单项	功 能
New	弹出 New 对话框，可以新建一个项目、文件、空的工作区文件或其他文件
Open	弹出 Open 对话框，用于在当前窗口中打开一个文件
Close	关闭已经打开的文件，如果有多个已经打开的文件，则关闭当前的活动窗口。若即将关闭的文件还未保存，系统会提示用户是否保存文件
Open Workspace	打开已有的工作区文件，默认为打开 .dsw 格式的文件
Save Workspace	保存当前的工作区文件
Close Workspace	关闭已经打开的工作区文件
Save	保存当前活动窗口中的文件
Save As	用新的文件名来保存打开的文件，如果不想覆盖原文件，则可选择此方式
Save All	保存所有窗口中的文件内容，如果某一文件还从未被保存过，会自动提醒为该文件输入有效地文件名
Page Setup	页面设置，对要求输出的文档进行输出效果的控制
Print	打印当前活动窗口中的内容
Recent Files	显示最近打开过的文件，用户可以对显示的文件数量进行设置
Recent Workspaces	显示最近打开过的工作区文件，用户可以对显示的文件数量进行设置
Exit	退出 Visual C++6.0 集成开发环境

1.2.2 Edit 菜单

Edit（编辑）菜单主要用于文本文件的编辑、查找和替换，具体功能见表 1-2。

表 1-2 Edit 菜单

菜单项	功 能
Undo	取消最近一次的编辑修改操作
Redo	取消最近一次的 Undo 操作，即抵消了 Undo 命令的效果
Cut	将选定的内容复制到剪贴板，然后从当前窗口删除选定内容
Copy	将选定内容复制到剪贴板，但不从当前窗口中删除选定内容
Paste	将剪贴板中内容插入到当前光标所在的位置
Delete	删除选定的内容
Select All	选定当前窗口中的全部内容
Find	在当前窗口文件中寻找指定内容
Find in Files	在多个文件中寻找指定内容
Replace	在文件中对指定内容进行替换
Go to	弹出 Go to 对话框，将光标移动到指定行
Bookmarks	书签功能，有五个子菜单，通过对不同子菜单的选择来完成书签的设置、清除，以方便对文档、程序的访问



续 表

菜单项	功 能
	实现一些高级控制的功能,但并不常用
	Format Selection: 格式化选定区域的内容
	Tabilify Selection: 用制表符替换选定区域的空格
	Untabilify Selection: 用空格替换选定区域的制表符
Advanced	Make Selection Uppercase: 使选定区域内的所有字母大写
	Make Selection Lowcase: 使选定区域内的所有字母小写
	Delete Horizontal white Space: 删除选定区域内的所有空白
	View White Space: 突出显示空格
	- Incremental Search: 渐进式搜索,选中后,对任意位置出现变量和对象均提示较详细的说明
Breakpoints	弹出 Breakpoints 对话框,用于设置、删除以及查看断点
List Member	Visual C++6.0 新增功能,如果在变量名称后键入“.”或“—>”,将显示一个列表显示当前所有有效的数据成员名称,可以从中进行选择
Type Info	Visual C++6.0 新增功能,在源代码窗口中将鼠标指针指向某一语法标志时,将显示该标志的语法特征
Parameter Info	Visual C++6.0 新增功能,输入程序代码时,如果在函数名称后键入左括号,将显示该函数的函数原型,并突出显示即将输入的参数
Complete Word	Visual C++6.0 新增功能,自动完成当前语句其他部分的输入,即使不能自动完成,也会给出相应的提示来帮助用户完成

1.2.3 View 菜单

View (视图) 菜单主要用于对集成开发环境中一些窗口的控制,下面将介绍一些经常使用的窗口。

1. ClassWizard (类向导)

ClassWizard (如图 1-2 所示) 是 Visual C++6.0 中一个非常重要的工具,用户可以通过它来建立消息映射,添加成员变量,进行自动化操作,添加 ActiveX 事件,设置类库信息等。对 ClassWizard 的使用贯穿整本书的始终,由于它的功能十分强大,因而使用也比较复杂,因此将在今后的学习过程中逐步地介绍 ClassWizard 的使用。

2: FileView (文件视图窗口)

Visual C++的文件视图窗口 (如图 1-3 所示) 是 Visual C++6.0 工作区的一部分,

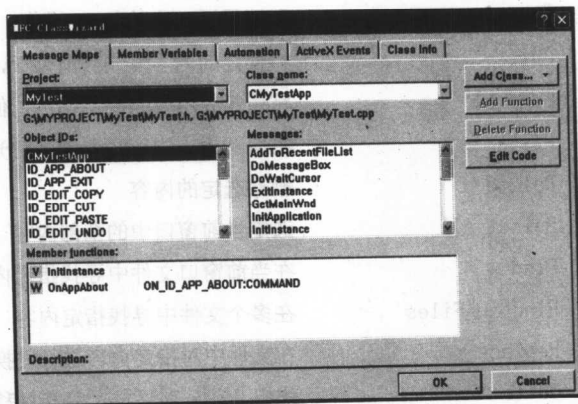


图 1-2 ClassWizard (类向导) 的界面

可以通过从 View (视图) 菜单中选择“工作区”菜单项或者使用加速键组合 Alt+0 调出工作区而将其调至前台。它提供了项目及其文件的有组织的视图, 显示了项目文件之间的关系, 同时提供了对项目 and 文件相关命令的便捷访问, 允许用户管理这些项目文件。例如, 可以通过双击文件来查看文件内容, 通过鼠标拖曳可以把文件添加到项目中或者在两个项目之间的移动或复制。

值得注意的是, 文件视图窗口中的中树形视图显示了工作区与项目、文件之间的逻辑关系, 并不一定表示它们之间的物理存储关系。

3. ClassView (类视图窗口)

类视图窗口 (如图 1-4 所示) 是 Visual C++6.0 工作区的一部分, 它显示了当前项目中所有的类以及这些类中的函数、变量、常量等。它为项目提供一种以代码为中心的视图, 可以通过它检查并迅速定位到项目中的符号, 指示符号之间的包容关系, 以方便对项目的控制。

类视图中显示了若干图标, 每个代表不同类型的符号。每个类成员的左边都会有一个或多个图标, 这些图标表明了类成员的附加信息, 即成员是变量、常量还是函数, 以及成员是保护类型、私有类型还是公有类型, 具体如表 1-3 所示。

表 1-3 类视图中的符号

图 标	说 明	图 标	说 明
	类		命令空间
	函数		运算符
	接口		属性
	结构		联合
	枚举		TypeDef
	映射		全局
	常数		枚举项
	映射项		外部声明
	宏		模板
	位置或错误		保护类型
	私有类型		友元
	内部类型		快捷方式

其中, 表格中的后五项是信号图标, 用于表明表中其他符号的访问权限, 如果符号前没有信号图标, 则表明为公有类型。

双击类或其成员, 可以查看相应的源代码。例如, 双击一个类, 就会打开类的头文件;

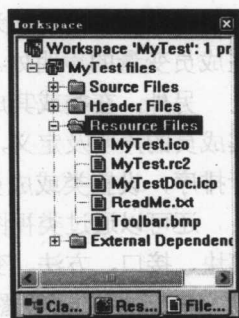


图 1-3 FileView 窗口

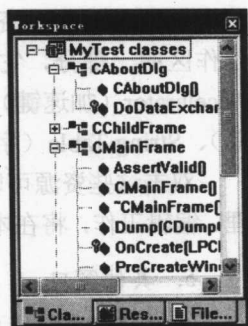


图 1-4 ClassView 窗口