

Special Edition Using Visual C++ 5



特版精品系列



(美) Kate Gregory 著
康博创作室 译

Visual C++ 5

开发使用手册



机械工业出版社

西蒙与舒斯特
国际出版公司



que

CMP

特版精品系列

Visual C++ 5 开发 使用手册

(美) Kate Gregory 著

康博创作室 译

文 珍 审校

机 械 工 业 出 版 社

西蒙与舒斯特国际出版公司

本书是一本极具参考价值的 Visual C++ 编程大全。书中通过对大量实例的详细分析,循序渐进、深入浅出地介绍了微软最新的 32 位应用程序可视化开发工具 Visual C++ 5.0。本书重点介绍了如何开发 Windows 95 或 NT 4.0 及以上版本环境下的应用程序以及面向 Internet 的应用程序,这一点与以前版本相比有很大不同。全书共由七部分组成,还有一个内容丰富的附录。其中第一部分到第三部分介绍了几乎每个 Windows 应用程序都会用到的概念。第四部分到第六部分介绍了一些高级技术包括如何改进用户界面、ActiveX 编程技术以及 Internet 编程技术。第七部分 Visual C++ 的高级编程技术包括数据库、多任务多线程以及模板开发等。附录给出了对全书的快速指南以及 VBScript 技术的快速参考。本书不仅适用于有一定 C++ 编程基础的初级程序设计和开发人员,更适合于熟悉 Visual C++ 技术的高级用户。利用本书提供的知识,读者可以迅速开发出功能强大的 Windows 应用程序。

Kate Gregory: Special Edition Using Visual C++ 5.

Authorized translation from the English language edition published by Que Corporation.

Copyright © 1997 by Que Corporation.

All rights reserved. For Sale in Mainland China only.

本书中文简体字版由机械工业出版社和美国西蒙与舒斯特国际出版公司合作出版,未经出版者书面许可,本书的任何部分不得以任何方式复制抄袭。

本书封面贴有 Prentice Hall 防伪标签,无标签者不得销售。

版权所有,翻印必究。

本书版权登记号:图字:01-97-1822

图书在版编目(CIP)数据

Visual C++ 5 开发使用手册/(美)格里高瑞(Gregory)等著;康博创作室译. - 北京:机械工业出版社,1998.5

(特版精品系列)

书名原文:Special Edition Using Visual C++ 5

ISBN 7-111-06154-3

I . V… II . ①格… ②康… III . C 语言-程序设计-手册 IV . TP312-62

中国版本图书馆 CIP 数据核字(98)第 06015 号

出 版 人:马九荣(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:蒋 克

昌平环球印刷厂·新华书店北京发行所发行

1998 年 5 月第 1 版第 1 次印刷

787mm×1092mm 1/16·49.5 印张

印数:0001-7000册

定价:102.00 元(含光盘)

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

译者序

Visual C++ 5.0 是 Microsoft 公司最新推出的开发 Windows 95 和 Windows NT 32 位应用程序的可视化开发工具,是 Developer Studio 97 工具集的重要组成部分。利用 Visual C++ 5.0,用户可以开发出规模更大、功能更加复杂的 Windows 应用程序。尽管如此,编程人员付出的工作量却在大大减少,因为 Visual C++ 5.0 提供了大量的 MFC 类和可视化的编辑工具,使得开发一个典型的 Windows 应用程序的框架简单到只需按几下鼠标即可完成。

与其以前的版本相比,Visual C++ 5.0 增加了许多新的功能,如 ActiveX 编程,包括建立 ActiveX 容器和服务,建立自动服务器以及创建 ActiveX 控件;Internet 编程包括 Internet 插口、Internet 下的 ActiveX 控件开发以及建立活动模板库;此外还介绍了防止内存泄漏,开发可重用代码和组件以及编写多线程的应用程序等最新技术。

本书是用 Visual C++ 进行程序开发的优秀参考书,也可以说是一本内容丰富的 Visual C++ 编程大全。本书主要目的是向读者介绍如何用 Visual C++ 5.0 开发 32 位 Windows 应用程序,包括数据库应用程序、Internet 应用程序和 ActiveX 应用程序等。因此,本书作者不但详细剖析了 Visual C++ 5.0 的新增功能,而且深入细致地介绍了用 Visual C++ 5.0 开发功能强大的应用程序的过程。本书根据不同层次读者的阅读需要,提供了大量实例以及对每个实例甚至是每一行代码的详细分析。同时还介绍了 Visual C++ 5.0 的一些高级技术及发展趋势,以使高级用户能迅速的掌握最新技术。全书内容丰富,编排新颖、实用,因此我们深信本书一定会成为广大 Visual C++ 爱好者的良师益友。

本书是康博创作室多人共同合作的结晶,参加本书翻译的具体人员有陈津、常征、陈安全、孙江宏、刘芝泉、刘莉、崔雨风、王峻飞、张杨、王春茫、周锐、蔡汇锦、王睿、赵文、周江华、孙连立、王卫林、黄琼、户东凯、田彦茹、李东升、刘伟、陈明、赵新军、何文、曹锋、李建民、孙锐、杨霞、郭莉、徐祖志、曹云、刘大伟、赵朴如、张达、郭俊、田学民等人。

在翻译过程中,我们本着严谨求实的态度,参考了各种相关书籍和资料。对于书中出现的新术语,我们都做了仔细的推敲与斟酌;对于疑难和分歧也进行了广泛的讨论,并最终达成了基本一致的意见。但是,由于时间仓促,加之本书涉及内容较新,疏漏和争议之处在所难免。我们热诚期望广大读者多提宝贵意见。

另外,本书英文原书附带光盘一张,对此感兴趣的读者可与机械工业出版社华章公司联系。

译者

1998 年 1 月

前 言

Visual C++ 是建立在 Window 95 和 Windows NT 32 位应用程序上的强有力的复杂工具。与 16 位 Windows 应用程序或者不使用图形用户界面的老程序相比,这些应用程序要大得多且更加复杂。然而,随着程序规模和复杂性增加,程序员需做的工作实际上在减少,至少对于使用适当工具的程序员来说是如此。

Visual C++ 是适当工具之一,借助于其生成代码的 Wizards,它能在数秒内生成可运行的 Windows 应用程序的外壳。Visual C++ 带有的类库,即 Microsoft Foundation Classes 类(MFC),已成为许多 C++ 编译器进行 Windows 软件开发的工业标准。可视化编辑工具使菜单和对话框的布局只要揪几下鼠标按钮即可。学习使用这一产品所花时间和精力在读者进行第一个 Windows 编程项目时将得到回报。

本书使用对象

本书将教读者如何使用 Visual C++ 建立 32 位 Windows 应用程序,包括数据库应用程序、Internet 应用程序以及开发 ActiveX 技术能力的应用程序。为了在 900 多页的篇幅里包含众多的内容,有一些内容必须略去,因此本书不包含下列内容:

- C++ 编程语言:读者应当已经熟悉 C++ 语言。本书偶尔也会提到有关 C++ 和面向对象的概念。原书附带光盘里包含了入门课程《C++ By Example》。

- 如何使用 Windows 应用程序:读者应当是个熟练的 Windows 用户,例如,能够缩放和移动窗口,双击,以及能认出熟悉的工具栏按钮。

- 如何把 Visual C++ 用作 C 编译器:如果读者已经使用 C 语言,那么可以用 Visual C++ 作为编译器。但是,新的开发人员应当使用 C++。

- 不采用 MFC 的 Windows 编程:这种编程技术对于熟悉它的人来说也是可行的,但既然有了 MFC,就不必去学了。

- ActiveX 编程的本质:这和 ActiveX 这一章有关,但该章仅告诉读者如何使它工作。

如果符合下列情况,读者应当阅读本书:

- 了解一些 C++ 和一些 Windows 编程技术,但不熟悉 Visual C++。如果读者只是试着写程序,了解该产品将比以前快得多。

- 已经在使用 Visual C++ 的老版本。用户经常以一种方式学习编程,而注意不到新产品功能。

- 使用 Visual C++ 5.0 已有一段时间,并且正开始觉得使用很困难。也许读者就是这样。

- 已经正式使用 Visual C++ 5.0,并且需要在自己的产品中增加一个功能。对于像 Help,打印和线程这样的任务,读者将会找到一个很好的起点。

在开始阅读之前

读者需要有 Visual C++ 5.0 的一个拷贝并已经安装它。安装过程是简单易行的,所以本

书不包含该内容。

在购买 Visual C++ 5.0 之前,需要 32 位 Windows 操作系统:Windows 95,Windows NT Server 或 Workstation。这意味着读者的计算机功能必须相当强和现代化:比如说 486 或更好的处理器,至少 16MB 的 RAM 以及 500MB 的硬盘空间,以及能进行 800×600 象素显示中更高分辨率的显示器。本书中的插图均是以 800×600 的分辨率制作的,并且,正如读者将看到的那样,有时也有些拥挤。

最后,读者必须保证,在阅读本书过程中需经常上机试验。如果不想键入所有代码,则不必键入,因为这些代码全部在本书英文原书附带的 CD 上供读者浏览。但读者应随时准备打开文件。

本书包含的内容

像利用 Visual C++ 进行 Windows 编程这样的主题包含的内容很广。本书分成学习章节(第 1 章~第 28 章)和附录(附录 A~附录 G)。读者应现在就浏览一下附录的标题,每当需要时,再回过头来仔细阅读它们,这些附录是:

- 附录 A,“Windows 编程回顾以及 CWnd 内幕”包含了 Windows 编程的具体细节,而这些细节在正文内被 CWnd 这样的 MFC 类所隐含。
- 附录 B,“Developer Studio 用户界面”解释了屏幕上所有菜单,工具栏,编辑区,以及快捷方式等,它们构成了用户和 Developer Studio 之间高度复杂和功能极其强大的界面。
- 附录 C,“调试”解释了与调试运行的应用程序有关的附加菜单,窗口,工具栏和命令。
- 附录 D,“MFC 宏和全局变量”总结了贯穿 Developer Studio Wizard 所生成代码的许多预处理宏和全局变量以及函数。
- 附录 E,“有用的类”描述了贯穿全书所使用的操纵日期,字符串,和对象集合的类。
- 附录 F,“Visual Basic Script 快速参考”为 C++ 用户提供了现成的 Visual Basic 参考。
- 附录 G,“光盘内容简介”是不需加以说明的,使用光盘即可。

根据读者的背景和读者在菜单下以及在联机帮助中浏览的意愿,读者可以对这些章节粗读一遍后不再返回查看,或者可以在这些章节内插满书签。尽管这些内容并没有向读者提供示例程序,但它们的确可以教会读者许多东西。

本书的主要内容是第 1 章到第 28 章。第 1 章教读者一个重要的编程任务,或两个紧密相关的任务,如创建一个任务栏或为应用程序增加 Help 功能。在每一章中,详细的说明告诉用户如何创建一个或几个可运行的应用程序。

前 9 章覆盖了几乎每个 Windows 应用程序都会用到的概念。其余章节的内容较不常用。下面是本书覆盖内容的一个快速浏览。

对话框与控件

什么 Windows 程序没有对话框?编辑框?按钮?对话框和控件对 Windows 用户界面来说是至关重要的,所有这些,甚至是简单按钮或一页静态文字,均是窗口。常用控件允许知道用户已投入到其他程序中的时间,以及开发人员已投入到操作系统的编程时间,允许像其他人一样利用相同的 File Open 对话框以及相同的分级树形控件等等。在第 2 章和第 3 章将学习有关这些控件的更多内容。

消息和命令

消息形成了 Windows 编程的核心。每当某个事件作用于 Windows 时,例如用户单击鼠标或按下某个键,就会触发一个消息并发送到一个或多个窗口,然后执行相应动作。Visual C++ 使用户易于编写捕捉这些消息并按消息进行动作的代码。第 4 章解释了消息的概念,以及 MFC 和 Visual C++ 的其他方面如何让用户处理这些消息。

视图/文档模型

示例是一个模型即观察事物的一种方式。MFC 的设计者假设每个程序均有一些东西想要存储在一个文件里,因此选择设计了一种框架。信息的集合称为“文档”,“视图”是观察文档的一种方式。把“文档”和“视图”分开有许多优点,这将在第 5 章进一步解释。MFC 提供了继承用户的文档类和视图类的类,因此像实现滚动条这样的普通编程任务不再是用户的事情。

在屏幕上绘图

不管 Windows 程序多么巧妙,如果不能在屏幕上显示一些文字或图形来告诉用户程序正在运行,那么无人会知道程序已经做了什么。大量的工作已由视图类自动完成(这是采用文档/视图范例模型的优点之一),但有时需要读者自己绘图。读者会在第 6 章学到关于设备环境、滚动以及更多的内容。

在纸上打印

为程序增加打印功能是非常容易的,只要重新利用一下绘制屏幕的代码即可。但如果包含一页以上的信息,事情就变得复杂了。第 7 章解释了这一切,以及映射模式,页眉,页脚和其他内容。

持久性和文件 I/O

有些美好的事物只是暂时的,如日历显示或联机闲谈窗口。但大多数程序能把文档保存到一个文件里,打开并加载该文件可重新创建某个已存储的文档,通过使用档案文件以及扩展 I/O 流运算符 > 和 < 的使用,MFC 使完成这一功能非常容易。读者会在第 8 章学到关于读写文件的所有内容。

ActiveX 编程

ActiveX 是 OLE 后继产品,正是这一技术促进了应用程序在对象层次上进行通信,允许用户在 Excel 电子表格中嵌入 Word 文档,或在任意 ActiveX 应用程序中嵌入几百种对象中的任意一个。ActiveX 内容在第 13、14、15、16 和 17 章介绍。

Internet 编程

Microsoft 公司意识到分布计算正变得越来越普及,其中,由两台或更多台不同计算机共同承担任务。程序需要相互对话,人们需要通过局域网(LAN)或环绕世界发送消息,MFC 具有支持这些通信方式的类。本书包含 Internet 内容的有第 18、19、20 和 21 章。

数据库访问

数据库编程变得越来越容易。ODBC(Microsoft 公司的开放数据库连接软件包)允许用户的代码调用 API 函数,以访问多种数据库文件(Oracle, DBase, Excel 电子表格,普通文本文件,利用 SQL 的老的主框架系统等)。用户调用标准的命名函数以及由数据库销售商或第三方提供的 API 来处理转换问题。详细内容参看第 22、23 章。

高级题材

本书还为已经掌握了基本内容的开发者专门提供了高级章节以提高编程技巧。利用第 24 章介绍的技术,读者将学会如何借助于第 24 章讨论的技术防止内存泄漏,查找瓶颈以及发现代码中的错误。

重用是目前软件开发中非常流行的概念,尤其对于那些寻求机会降低开发预算的管理人员来说更是这样。如果想编写可重用的代码和组件,请看第 25 章。

C++ 语言仍然是新颖的语言,它每年都发生改变。ANSI 的一个工作委员会正在制订 C++ 标准,并且随着朝最终定义方面发展,编译器销售商增加了全新的关键字和性能。读者将会在第 26 章学到语言本身的更新内容。

随着用户对高性能软件需求的不断上升,开发人员必须学习全新的技术,以编写出提供快速响应时间的强有力的应用程序。对于许多开发人员来说,编写多线程的应用程序是个关键技术。关于这个内容,请看第 27 章。

第 28 章介绍了如何生成控制台应用程序,利用和创建用户自己的 DLL 以及采用 Unicode。

本书使用的约定

本书附有很多代码,有时读者只需看一、两行即可,因此代码和文本是混在一起的,例如:

```
int SomeFunction( int x, int y);
{
    return x + y;
}
```

读者可从编排方式来区别代码和正文。有时,读者会看到一段代码太长而不能与正文混在一起的,则采用清单的方式,如:

清单 1 清单例子

```
CHostDialog dialog(m_pMainwnd);
if (dialog.DoModal() == IDOK)
{
    AppSocket = new CSocket();
    if (AppSocket->Connect(dialog.m_hostname,119))
    {
        while (AppSocket->GetStatus() == CONNECTING)
        {
            YieldControl();
        }
    }
}
```

```
        |
        | if (AppSocket->GetStatus() != CONNECTED)
        | {
        |     CString response = AppSocket->GetLine();
        |     SocketAvailable = TRUE;
        | }
    |
    | if (! SocketAvailable)
    | {
    |     AfxMessageBox("Can't connect to server. Please
    |     =>quit.", MB_OK | MB_ICONSTOP);
    | }
```

倒数第 2 行的字符(⇒)称为代码继续符。它表示:为了在一页上写完一行程序而必须打断的位置,但实际上该行并不在那里中断。如果读者键入该段代码,请不要在那儿中断,只要接着输入即可。如果从光盘上读出该段代码,当发现该行未在那儿中断时,请不要感到困惑。

记住:代码就在书中,以便读者理解正在进行的事情,而不是供读者键入的。所有代码也都在英文原书附带的 CD-ROM 上。有时,读者会通过开发一个应用程序而循序渐进,并且会看到一段代码的几个版本——最终的版本在 CD-ROM 上。

技巧:读者可能想知道的捷径或感兴趣的功能。

注意:它包含了微妙而重要的方面,即使忽略了技巧,也不要忽略注意。

警告:这是一个警告,并且是严重的。它警告读者错误步骤会造成的可怕后果,因此应确保阅读遇到的所有的警告。

疑难解答:

编程时如果遇到问题怎么办?任何人在编程时都会不时地碰到问题。疑难解答这一项预料到了一些问题——读者可从中了解到如何摆脱麻烦。

开始时刻

上面就是本书的内容预览。读者已了解到了开始所必须的一切,包括一些本书通篇用到的标记的忠告。现在可立即着手来学习用 MFC 编写 Windows 应用程序,然后就开始开发自己的一些程序!祝读者好运!

目 录

译者序
前言

第一部分 从头开始

第 1 章 建立第一个应用程序	1
1.1 生成一个 Windows 应用程序	1
1.1.1 决定应用程序支持多少文档	2
1.1.2 数据库	3
1.1.3 复合文档支持	4
1.1.4 外观和其他选项	5
1.1.5 其他选项	7
1.1.6 文件和类名	8
1.1.7 生成应用程序	8
1.1.8 试一试	9
1.2 生成基于对话的应用程序	10
1.3 生成 DLL, 控制台应用程序以及其他	13
1.3.1 ATL COM AppWizard	14
1.3.2 Custom AppWizard	14
1.3.3 Database Project	14
1.3.4 DevStudio Add-in Wizard	14
1.3.5 ISAPI Extension Wizard	14
1.3.6 Makefile	14
1.3.7 MFC ActiveX Control Wizard	14
1.3.8 MFC AppWizard(DLL)	15
1.3.9 New Database Wizard	15
1.3.10 Win32 Application	15
1.3.11 Win32 Console Application	15
1.3.12 Win32 Dynamic-Link Library	15
1.3.13 Win32 Static Library	15
1.4 改变 AppWizard 决定	15
1.5 理解 AppWizard 代码	16
1.5.1 单文档界面应用程序	16
1.5.2 其他文件	21
1.6 理解多文档界面应用程序	21
1.7 基于对话的应用程序	24
1.8 回顾 AppWizard 决定以及本书	26

1.9 小结	27
--------------	----

第二部分 用对话框和控件与应用程序交互

第 2 章 对话框和控件	29
2.1 理解对话框	29
2.2 创建对话框资源	30
2.2.1 定义对话框和控件的 ID	31
2.2.2 创建示例对话框	31
2.3 编写对话框类	32
2.4 使用对话框类	34
2.4.1 显示对话框	35
2.4.2 屏幕之后	36
2.4.3 使用列表框控件	37
2.4.4 使用单选按钮	41
2.5 小结	42
第 3 章 Windows 95 常用控件	43
3.1 Win95 控件应用程序	43
3.2 进度条控件	45
3.2.1 创建进度条	45
3.2.2 初始化进度条	46
3.2.3 操作进度条	46
3.3 滑块控件	46
3.3.1 创建跟踪条	47
3.3.2 初始化跟踪条	48
3.3.3 操作滑块	49
3.4 上下控件	49
3.5 图像列表控件	51
3.5.1 创建图像列表	52
3.5.2 初始化图像列表	52
3.6 列表视图控件	53
3.6.1 创建列表视图	56
3.6.2 创建列表视图的列	59
3.6.3 创建列表视图的项目	59
3.6.4 操作列表视图	61
3.7 树形视图控件	63
3.7.1 创建树形视图	64
3.7.2 创建树形视图的项目	66

3.7.3 操作树形视图	67	6.3.6 使用画刷	113
3.8 Rich 编辑控件	67	6.4 滚动窗口	115
3.8.1 创建 rich 编辑控件	68	6.5 小结	121
3.8.2 初始化 rich 编辑控件	69	第 7 章 打印和打印预览	122
3.8.3 操作 rich 编辑控件	71	7.1 理解 MFC 的基本打印和打印预览	122
3.9 小结	73	7.2 缩放	124
第 4 章 消息和命令	74	7.3 打印多页	126
4.1 消息传递	74	7.4 设置原点	131
4.2 消息循环	75	7.5 MFC 和打印	133
4.3 消息映射	76	7.6 小结	135
4.3.1 消息映射宏	77	第 8 章 持久性和文件 I/O	136
4.3.2 消息映射是如何工作的	78	8.1 对象和持久性	136
4.3.3 由 MFC 代码所捕获的消息	80	8.2 File Demo 应用程序	136
4.4 ClassWizard 帮助用户捕获消息	80	8.2.1 Document 类的回顾	138
4.4.1 带标签的 ClassWizard 对话框	80	8.2.2 File Demo 的源代码的快速浏览	138
4.4.2 添加 Windows 消息处理函数对话框	81	8.3 创建一个持久类	141
4.4.3 哪个类应捕获消息	82	8.3.1 File Demo 2 应用程序	142
4.5 消息列表	82	8.3.2 CMessage 类一览	142
4.6 命令	84	8.3.3 在程序中使用 CMessage 类	145
4.7 命令更新	84	8.4 直接读写文件	147
4.8 ClassWizard 帮助用户捕获命令和命令更新	86	8.4.1 File Demo 3 应用程序:使用 CFile	147
4.9 小结	87	8.4.2 CFile 类	147
第 5 章 文档和视图	88	8.4.3 开发 File Demo 3 应用程序	149
5.1 了解文档类	88	8.5 创建自己的 CArchive 对象	151
5.2 了解视图类	90	8.6 注册	152
5.3 建立 Rectangles 应用程序	93	8.6.1 Registry 是如何设置的	152
5.4 其他视图类	97	8.6.2 预定义的键	152
5.5 文档模板,视图和框架窗口	98	8.6.3 在 MFC 应用程序中使用 Registry	154
5.6 小结	100	8.6.4 重新访问 File Demo 2 应用程序	155
第三部分 由应用程序获取信息		8.7 小结	155
第 6 章 在屏幕上绘图	101	第 9 章 创建一个完整的应用程序:ShowString	157
6.1 理解设备环境	101	9.1 建立显示字符串的应用程序	157
6.2 Point1 应用程序介绍	102	9.1.1 用 AppWizard 创建一个空壳	157
6.3 创建 Paint1 应用程序	103	9.1.2 显示字符串	158
6.3.1 在一个 MFC 程序中画图	104	9.2 建立 ShowString 菜单	161
6.3.2 切换显示	105	9.3 创建 ShowString 对话框	164
6.3.3 使用字体	106	9.3.1 ShowString 的 About 对话框	164
6.3.4 改变窗口的大小和位置	110	9.3.2 ShowString 的选项对话框	165
6.3.5 使用画笔	111		

9.4 使菜单工作	166
9.4.1 对话框类	166
9.4.2 捕获消息	169
9.5 使对话框工作	170
9.6 向选项对话框添加外观选项	171
9.6.1 改变选项对话框	172
9.6.2 在对话框类中增加成员变量	173
9.6.3 在文档中增加成员变量	174
9.6.4 改变 OnToolsOptions()	175
9.6.5 改变 OnDraw()	176
9.7 小结	177

第四部分 改善用户界面

第 10 章 状态栏和工具栏	179
10.1 使用工具栏	179
10.1.1 删除工具栏按钮	179
10.1.2 添加按钮到工具栏	181
10.1.3 CToolBar 类的成员函数	185
10.2 使用状态栏	186
10.2.1 创建新的命令 ID	188
10.2.2 创建缺省字符串	188
10.2.3 添加 ID 到指示器数组	189
10.2.4 创建窗格的命令更新处理程序	190
10.2.5 设置状态栏的外观	191
10.3 小结	193
第 11 章 帮助	195
11.1 Help 的不同类型	195
11.1.1 获得 Help	196
11.1.2 显示 Help	196
11.1.3 使用 Help	196
11.1.4 编写 Help 程序	198
11.2 帮助系统的组件	198
11.3 AppWizard 支持的 Help	200
11.4 设计用户 Help 的方法	200
11.5 命令 Help 的编程	201
11.6 上下文 Help 的编程	202
11.7 编写 Help 文本	205
11.7.1 更改占位符字符串	207
11.7.2 添加主题	209
11.7.3 更改“How to Modify Text”主题	212
11.8 Contents 的调整	213
11.9 小结	215

第 12 章 属性页,属性表和属性向导	216
12.1 属性表介绍	216
12.2 创建属性表演示应用程序	217
12.3 运行属性表演示应用程序	227
12.4 在用户应用程序中添加属性表	227
12.5 把属性表变成 Wizards	228
12.5.1 运行 Wizard Demo 应用程序	229
12.5.2 创建 Wizard 页	230
12.5.3 显示 Wizard	230
12.5.4 设置 Wizard 的按钮	231
12.5.5 响应向导的按钮	232
12.6 小结	233

第五部分 ActiveX 应用程序和控件

第 13 章 ActiveX 概念	235
13.1 ActiveX 的目的	235
13.2 对象链接	238
13.3 对象嵌入	239
13.4 容器与服务器	240
13.5 更加直观的用户界面	241
13.6 组件对象模型	242
13.7 ActiveX Automation	243
13.8 ActiveX 控件	245
13.9 小结	245
第 14 章 创建 ActiveX 容器应用程序	246
14.1 改变 Showstring	246
14.1.1 由 AppWizard 生成的 ActiveX 容器代码	246
14.1.2 返回 ShowString 功能	262
14.2 移动、重设大小和跟踪	263
14.3 处理多个对象和对象选取	266
14.3.1 击中测试	266
14.3.2 画多个选项	267
14.3.3 处理单击	268
14.3.4 处理双击	271
14.4 实现拖放	271
14.4.1 实现拖动源	271
14.4.2 实现释放目标	273
14.4.3 注册视图作为释放目标	273
14.4.4 建立函数框架,添加成员变量	274
14.4.5 OnDragEnter	275
14.4.6 OnDragOver	277

14.4.7	OnDragLeave	279
14.4.8	OnDragDrop	279
14.4.9	测试和拖动目标	281
14.5	删除对象	281
14.6	小结	282
第 15 章	创建 ActiveX 服务器应用程 序	283
15.1	给 ShowString 添加服务器功能	283
15.1.1	AppWizard 的服务器示例程序	283
15.1.2	再次显示字符串	298
15.2	既是服务器又是容器的应用程序	302
15.2.1	创建另一个版本的 ShowString	302
15.2.2	嵌套和递归问题	303
15.3	ActiveX 文档	304
15.3.1	ActiveX Documents 做些什么	304
15.3.2	使 ShowString 成为 ActiveX 文 档服务器	305
15.4	小结	309
第 16 章	创建自动服务器	310
16.1	再次设计 ShowString	310
16.1.1	AppWizard 的自动示例程序	310
16.1.2	揭示属性	313
16.1.3	OnDraw() 函数	318
16.1.4	显示窗口	320
16.2	Visual Basic	322
16.3	类型库和 ActiveX 内核	325
16.4	小结	327
第 17 章	创建 ActiveX 控件	328
17.1	一个滚动的骰子控件	328
17.1.1	创建控件外壳	328
17.1.2	AppWizard 的代码	330
17.1.3	设计控件	333
17.2	显示当前值	333
17.2.1	添加属性	333
17.2.2	编写绘图代码	335
17.3	响应鼠标单击和滚动骰子	337
17.3.1	通知容器	338
17.3.2	滚动骰子	339
17.4	更好的用户界面	341
17.4.1	位图图标	341
17.4.2	显示圆点	341
17.5	属性表	345

17.5.1	数字与圆点	345
17.5.2	用户选择的颜色	349
17.6	按需滚动	354
17.7	进一步改进	355
17.7.1	允许与禁止滚动	355
17.7.2	具有非同寻常面数的骰子	355
17.7.3	骰子数组	355
17.8	小结	355

第六部分 Internet 编程

第 18 章	插口, MAPI 和 Internet	357
18.1	使用 Windows 插口	357
18.2	消息收发 API(MAPI)	360
18.2.1	何为 MAPI?	360
18.2.2	Win95 标志要求	361
18.2.3	MAPI 的高级应用	363
18.3	使用 Visual C++ 4.2 中的新的 Internet 类	365
18.4	使用 Internet 服务器 API(ISAPI) 类	366
18.5	小结	368
第 19 章	用 Winlnet 类进行 Internet 编程	370
19.1	设计 Internet 查询应用程序	370
19.2	创建查询对话框	372
19.3	查询 HTTP 站点	374
19.4	查询 FTP 站点	379
19.5	查询 Gopher 站点	381
19.6	使用 Gopher 发送 Finger 查询	383
19.7	使用 Gopher 发送一个 Whois 查询	386
19.8	未来的工作	388
19.9	小结	388
第 20 章	创建一个 Internet ActiveX 控件	389
20.1	在 Microsoft ExplorerWeb 页中嵌入 一个 ActiveX 控件	389
20.2	把一个 ActiveX 控件嵌入到 Netscape NavigatorWeb 页中	392
20.3	为了改编和初始化安全而注册	394
20.4	在 ActiveX 和 Java 之间进行选择	398
20.5	使用 AppWizard 来创建更快的 ActiveX 控件	398
20.6	用异步属性加速控件代码	402

20.6.1 属性	403
20.6.2 使用 BLOB	403
20.6.3 改变 Dieroll	404
20.6.4 测试和调试 Dieroll	410
20.7 小结	411
第 21 章 活动模板库	413
21.1 为何使用 ATL?	413
21.2 用 AppWizard 开始	414
21.3 使用 Object Wizard	415
21.3.1 在工程中加入一个控件	415
21.3.2 命名此控件	415
21.3.3 设置控件属性	416
21.3.4 支持 Stock 属性	418
21.4 给控件添加属性	418
21.4.1 由 Object Wizard 生成的代码	418
21.4.2 添加 ReadyState Stock 属性	421
21.4.3 增加定制属性	421
21.4.4 初始化属性	423
21.4.5 添加异步属性	424
21.5 绘制控件	427
21.6 持久性和属性页	431
21.6.1 增加一个属性页	431
21.6.2 连接属性页到 CDieRoll	435
21.6.3 在属性包中的持久性	435
21.7 在 Control Pad 中使用控件	435
21.8 增加事件	436
21.8.1 编辑 idl 文件	436
21.8.2 IConnectionPoint 接口的包装类	437
21.8.3 触发单击事件	438
21.8.4 触发 ReadyStateChange 事件	439
21.9 展示 DoRoll() 函数	439
21.10 Init Safe 和 Script Safe 的注册	440
21.11 准备在设计模式中使用的控件	441
21.12 最小化可执行文件大小	442
21.13 在 Web 页中使用控件	444
21.14 修正闪烁	445
21.15 小结	446

第七部分 高级编程技术

第 22 章 数据库访问	447
22.1 数据库概念	447
22.1.1 平面数据库模型	448
22.1.2 关系数据库模型	448

22.1.3 数据库访问	448
22.1.4 Visual C++ ODBC 类	450
22.2 创建 ODBC 数据库应用程序	450
22.2.1 注册数据库	450
22.2.2 创建基本的 Employee 应用程序	452
22.2.3 创建数据库显示	455
22.2.4 添加和删除记录	459
22.2.5 OnRecordAdd() 函数分析	464
22.2.6 OnMove() 函数分析	464
22.2.7 OnRecordDelete() 函数分析	465
22.2.8 排序与过滤	465
22.2.9 OnSortDept() 函数分析	470
22.2.10 DoFilter() 函数分析	470
22.3 在 ODBC 与 DAO 之间作出选择	471
22.4 OLE DB	472
22.5 小结	473
第 23 章 最新的企业版	474
23.1 SQL	474
23.2 在 C++ 中使用 SQL 操作数据库	475
23.3 开发出版应用程序	476
23.3.1 创建应用程序外壳	476
23.3.2 数据连接	479
23.3.3 查询设计器	481
23.3.4 存储过程	482
23.3.5 编写新的存储过程	484
23.3.6 存储过程与 C++ 代码的连接	486
23.4 使用数据库	489
23.4.1 数据库设计器	489
23.4.2 数据库图表	491
23.5 Microsoft 事务处理服务器	492
23.6 使用 Visual SourceSafe	493
23.7 小结	494
第 24 章 提高应用程序性能	495
24.1 ASSERT 和 TRACE	495
24.1.1 ASSERT: 检测逻辑错误	495
24.1.2 TRACE: 隔离用户程序中共 在问题的区域	496
24.2 只调试特征	497
24.3 封住内存泄漏	499
24.3.1 内存泄漏的共同原因	499
24.3.2 调试使用 new 和 delete 调用 的代码	500

24.3.3 自动指针	501	26.5.3 typename 关键字	540
24.4 优化	502	26.5.4 explicit 关键字	541
24.5 配置	504	26.6 小结	541
24.6 小结	504	第 27 章 Windows 线程的多任务处	
第 25 章 用 Gallery 和用户自己的		理	543
AppWizard 的实现重用	506	27.1 简单线程	543
25.1 书写可重用代码的优点	506	27.2 线程通信	547
25.2 使用 Component Gallery	507	27.2.1 利用全局变量通信	547
25.2.1 向 Gallery 中添加一个组件	507	27.2.2 利用用户定义的消息通信	549
25.2.2 在用户项目中使用 Gallery		27.2.3 利用事件对象通信	551
Components	508	27.3 使用线程同步	553
25.3 开发 Gallery	509	27.3.1 使用临界区	553
25.4 定制的 AppWizard 介绍	510	27.3.2 使用互斥	558
25.5 小结	513	27.3.3 使用信号量	559
第 26 章 例外、模板和 C++ 的最新		27.4 小结	562
补充	514	第 28 章 未来的开发方法	563
26.1 例外	514	28.1 创建控制台应用程序	563
26.1.1 简单的例外处理	515	28.1.1 创建一个控制台可执行程序	563
26.1.2 例外对象	516	28.1.2 编写一个面向对象的控制台	
26.1.3 放置 catch 块	517	应用程序	565
26.1.4 处理多种类型的例外	521	28.1.3 搭架起离散的计算	566
26.1.5 旧的例外机制	523	28.2 创建并使用一个 32 位的动态链	
26.2 使用模板	523	接库	567
26.2.1 模板介绍	524	28.3 制作一个 32 位的 DLL	569
26.2.2 创建函数模板	524	28.4 使用 32 位 DLLs	572
26.2.3 创建类模板	526	28.5 发送消息和命令	574
26.2.4 标准模板库	530	28.6 软件的国际化发展趋势	575
26.3 使用运行时类型信息	532	28.7 小结	577
26.3.1 RTTI 介绍	532		
26.3.2 实现安全的向下传播	533	第八部分 附 录	
26.3.3 获得对象信息	533	附录 A Windows 编程回顾以及 CWnd	
26.3.4 准备使用 RTTI	535	内幕	579
26.4 名字空间	536	附录 B Developer Studio 用户界面	588
26.4.1 定义一个名字空间	536	附录 C 调试	639
26.4.2 名字空间范围确定	537	附录 D MFC 宏和全局变量	651
26.4.3 未命名的名字空间	538	附录 E 有用的类	658
26.4.4 名字空间别名	538	附录 F Visual Basic Script 快速参考	681
26.5 新的关键字与数据类型	539	附录 G 光盘内容简介	777
26.5.1 bool 数据类型	539		
26.5.2 mutable 关键字	539		

第一部分 从头开始

第 1 章 建立第一个应用程序

本章要点:

- 利用 AppWizard 生成典型的应用程序
大多数人在考虑“应用程序”时期待着 EXE 文件。AppWizard 不到 1 分钟即可生成可执行的 Windows 程序的框架。
- AppWizard 可以生成的其他应用程序
其他生成向导的应用程序能生成 DLL, ActiveX 控件, 控制台应用程序, 库, 编译文件, Internet Server 扩展以及过滤器等。
- 如果改变主意的话
AppWizard 能为首次建立的应用程序增加许多功能。如果想为已经建立的应用程序增加一些功能则情况如何呢? 这一部分教读者怎样进行。
- AppWizard 代码
查看由 AppWizard 生成的代码并了解不同的应用程序类型。
- 其他 AppWizard 决定
这一部分把读者引导到本书中的相应章节, 在这些章节, 详细解释了 AppWizard 决定的结果, 如: 数据库支持, ActiveX 技术以及应用程序类型。

Visual C++ 不仅编译代码, 而且产生代码。读者在几分钟内即可生成一个 Windows 应用程序, 只要告诉 AppWizard 生成一个具有所有模板代码的“超始应用程序”即可。AppWizard 是非常有效的工具, 它把几乎所有 Windows 应用程序所需的代码复制到用户的应用程序中。毕竟, 读者不是第一个需要应用程序具有缩放边框, 最小最大化按钮, 带有 Open, Close, Print Setup, Print 和 Exit 选项的 File 菜单功能的程序员, 对吗?

1.1 生成一个 Windows 应用程序

AppWizard 能生成许多不同种类的应用程序, 但大多数人想要的(至少在开始)是可执行(.exe)程序。大多数人还要求 AppWizard 生成每个程序均必需的模板代码——类, 对象和函数。为了生成这样的程序, 请选择 File, New 并单击 New 对话框中的 Projects 标签, 如图 1-1 所示。

选择左边列表框中的 MFC AppWizard(.exe), 填上项目名, 并单击 OK。然后 AppWizard 通过许多步骤进行工作。读者在每一步上要作出关于想要何种应用程序的决定, 并单击 Next。在任何时候, 均可单击 Back 返回到前一个决定, 单击 Cancel 可放弃整个过程, 单击 Help 可了解更多的细节, 或单击 Finish 跳到最后一步生成应用程序而不用回答更多的问题(在完成最后

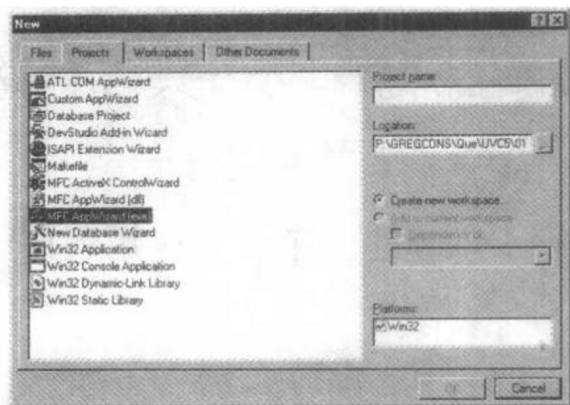


图 1-1 New 对话框的 Projects 标签处在欲建立的应用程序种类上一步之前不推荐这样做), 每一步骤均包含在下面的各部分内容里。

1.1.1 决定应用程序支持多少文档

与 AppWizard 通信的第一个决定(如图 1-2 所示)是应用程序是否为 MDI, SDI, 或基于对话的类型, AppWizard 为每一种类型的应用类型生成不同的代码和类。

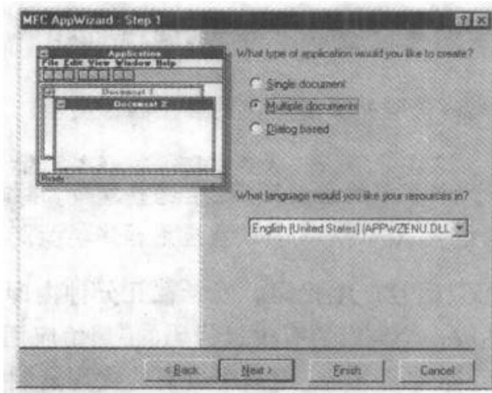


图 1-2 用 AppWizard 建立一个典型应用程序的第一步是选择界面

3 种可供选择的应用程序类型如下:

- 单文档界面(SDI)应用程序, 例如 Notepad, 一次只能打开一个文档。当选择 File, Open 时, 当前打开的文件在新文件打开之前关闭。
- 多文档界面(MDI)应用程序, 例如 Excel 或 Word, 一次能打开多个文档(通常是文件), 在 File 菜单上有 Window 菜单和 Close 项。下面是一句关于 MFC 的双关语: 如果希望在一个文档上有多个视图, 则必须建立 MDI 应用程序。
- 基于对话的应用程序, 例如 Windows 附带的 Character Map 实用程序(见图 1-3), 它根本没有文档, 也没有菜单。(如果想查看运行中的 Character Map, 则可通过单击 Start 在 Accessory 文件夹中找到它。读者可能需要利用 Control Panel 下的 Add/Remove 程序安装它。)