

Designed for
Microsoft®
Windows NT®
Windows 95



CD-ROM
Included

Visual C++[®]

技术内幕

第4版 修订版

Microsoft
Programming
Series

Microsoft Visual C++
编程的标准用书



David J. Kruglinski 著
潘爱民 王国印 译

清华大学出版社



Microsoft[®] Press

Visual C++技术内幕

(第4版)修订版

David J. Kruglinski 著
潘爱民 王国印 译

清华大学出版社
北京

Visual C++ 技术内幕(第4版)(修订版)

Inside Visual C++, 4th Edition

David J. Kruglinski

Copyright © 1997 by David J. Kruglinski

Original English language Edition Copyright © 1997 by David J. Kruglinski

Published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U. S. A.

本书中文版由 Microsoft Press 授权清华大学出版社出版。

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。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

Visual C++ 技术内幕: 第4版 / (美)小克鲁格林斯基(Kruglinski, D. J.)著;潘爱民,王国印译. —修订版. —北京:清华大学出版社, 2009.4

书名原文: Inside Visual C++, 4th Edition

ISBN 978-7-302-19721-8

I. V… II. ①小… ②潘… ③王… III. C++ 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 033206 号

责任编辑: 汤斌浩 龙啟铭

责任印制: 杨 艳

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 清华大学印刷厂

经 销: 全国新华书店

开 本: 185×230

印 张: 53.5

字 数: 1195 千字

(附光盘 1 张)

版 次: 2009 年 4 月第 1 版

印 次: 2009 年 4 月第 1 次印刷

印 数: 1~3000

定 价: 99.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: 010-62770177 转 3103 产品编号: 033087-01

十年后的修订说明

《Visual C++ 技术内幕(第4版)》是一本富有传奇色彩的计算机图书。它的作者不幸遇难,留下了第4版绝笔;它压倒了第5、6版的锋头,现在要重印这一版本也说明了这一点。对于像我这样的早期 Visual C++ 程序员,这本书就像一首经典的老歌,曾经伴随我们度过了多少个编程之夜。这几年我偶尔翻看仅存的一本样书,脑子里浮现的全是记忆。

1997年秋季开学后,我从王国印老师手中接过这本书的翻译工作,当时我由于健康原因无法再继续从事软件研发工作,想着最好还能做点有意义的事情,便开始了这本书的翻译。这是我第一次翻译技术图书,以前从没想过要做这样的事情,现在只求力所能及做一点,这是我当时的心态。尽管如此,我也不敢轻视,整个过程非常谨慎,力求译文准确。还好,技术方面几乎没有任何障碍,书中介绍的绝大多数技术我都用到过,可谓了然于胸。但身体确有些吃不消,记得那段时间我每天还要熬药,如果这不是用电子稿而用纸稿来完成的话,这本书的译稿一定充满了中药味。即便如此,我也不敢耽误时间,4个月左右便完成了整本书的翻译,每一章还检查了两遍。交稿以后不再挂念这本书,到1999年1月出版时,很是惊喜地看到了自己的第一部作品。

时隔十年,当2008年12月龙启铭编辑告诉我,出版社计划要重印《Visual C++ 技术内幕(第4版)》时,我曾经写过一篇博文,大意是更应该重印第6版,毕竟第6版针对 Visual C++ .NET(相当于 Visual C++ 7.0),而第4版针对 Visual C++ 5.0,而且,第6版新增加了部分内容。第6版也是我翻译的,所以我很清楚两者的内容。但是,从感情而言,我很愿意看到第4版重印。

十年前,开发 Windows 应用程序的可选工具和语言并不多,Visual C++ 和 MFC 当然是首选的开发环境和支持库。但今非昔比,随着 .NET 的普及和完善,Visual C++ 和 MFC 正在淡出主流。Visual C++ 的拥护者,譬如我,可以列举出足够多的理由来说明它的不可替代性。尽管周围越来越多的人在使用其他的语言开发 Windows 程序,MFC 也乏人问津,但是,从技术角度而言,Visual C++ 依然是最为强大的编程语言之一,特别是对于开发 Windows 程序,而 MFC 作为应用程序框架类库,我认为迄今仍无出其右者,它成了 Windows 程序设计的 C++ 封装典范。这十年中流行的各种设计模式,在 MFC 类库中几乎都能看到非常恰当的应用。MFC 类库在技术应用上是如此成功,以至于 Windows 程序的各种新颖界面元素,始终可以在 MFC 基础上实现。Microsoft 提供的 MFC-Next 是在 MFC 上的一个扩展包,利用 MFCNext 可以很方便地建立起拥有 Office、Inter-

net Explorer 或 Visual Studio 界面风格的 Windows 程序。

从学习的角度而言,我一直坚持这样的观点:学语言应该学 C++,要开发 Windows 应用程序应该看看 MFC。C++ 是程序语言的宝库,MFC 是开发 Windows 应用程序的技术宝库。学习 MFC 是否需要阅读 MFC 源代码呢?通常情况下不需要。由于 MFC 支持太多 Windows 程序设计功能,它已经非常庞大,甚至累赘了。如何有效地学习 MFC 呢?我认为,《Visual C++ 技术内幕(第 4 版)》可能是最佳的选择,它可以帮助读者系统地了解 MFC 的精华。尽管它缺少后续 Visual C++ 版本提供的一些开发支持,但 MFC 的核心都涵盖在这本书中了。我想这可能是很多人怀念或追寻这本书的原因吧。

说到这本书的翻译质量,无疑它远没有达到完美,但有一点我非常确信,早期的 Visual C++ 程序员确实使用书中的语言来描述和讨论问题,我翻译这本书时刚刚从软件开发一线上退下来,还保持着程序员语言的习惯。然而,个别地方的疏漏是有可能的,请读者谅解。

最后,借此重印机会,感谢读者们这十年来对这本书的厚爱。

潘爱民

2009 年 2 月

作者简介

David Kruglinski 是一位自学成才的、杰出的程序设计员、作家、教师和户外活动家。

自从 1966 年加入 Purdue 大学,他就开始了程序设计生涯;1976 年他的一位朋友从垃圾箱中发现了一块 8080 板,从那以后他就开始针对微机编写应用程序。

《Visual C++ 技术内幕》是 David 的第五本书。1992 年他为 Microsoft 出版社写了该书的第一版。该书赢得了很大的成功并受到了读者的广泛欢迎。作为最畅销的书,该书现在已经到第四版了;在为 Microsoft 出版社写作之前,David 编写了四本书,涉及的内容从微机数据库管理系统一直到 PC 通信。

由于他在写作和程序设计方面的出色表现,1991 年中期他与 Microsoft 基本类库(MFC) 1.0 版本的文档小组建立了正式合约关系。在大多数 MFC 参考材料,特别是对一些基本类如 CString 等的说明中,我们不难看到 David 的精辟论述。当 1992 年 Microsoft C/C++ 7.0 正式推出时,他离开了 Microsoft 语言开发小组,专门从事《Visual C++ 技术内幕》的写作。

David 被公认为 MFC 和 Visual C++ 语言这两方面技术的权威。他成功地发起组织了软件工具研究小组,并兼作 Microsoft Visual C++ 的顾问和指导。他经常驾驶着他的滑翔机,在各地授课、做报告、参加会议等。David 通过与 Microsoft 的密切合作,紧密跟踪着最新的软件发展方向。他经常去拜访 Visual C++ 开发小组的成员,从而得以不断跟踪新的发展,并深入研究 MFC 中的一些新技术,如 OLE、数据库和 Internet 等。

作为一个周游世界的旅行家和户外活动爱好者,David 经常徒步旅行和爬山,他从不闲着。从 80 年代后期他开始滑翔机运动,在过去的几年中,作为一个优秀的飞行员,他赢得了国际级的名誉。David 是一个大胆的、非传统的飞行员,他时而翱翔在高山上,时而低飞在山谷中,充分领略了从欧洲到北美洲的壮观美景。他不断刷新新的记录,包括空中停留时间、旅行距离、飞行高度。

“顺风去吧,Dave。”1997 年 4 月 17 日,Kruglinski 在华盛顿州 Okanogan 郡 Methow 峡谷飞行时不幸遇难,终年 49 岁。

“据说,当藏族人面对危险,或者当他们要穿过雪山和大河时,他们会喊,‘每个活着的人 100 年后都会死,所以又有什么可怕的呢?’Dave 就是这样的人,但他没有喊。”

Lowell Skoog,滑翔机飞行伙伴

译者前言

Visual C++ 自诞生以来,一直是 Windows 环境下最主要的应用开发系统。本书的第一版和第二版主要针对 16 位 Windows 应用程序的开发,以 MFC 2.0 和 2.5 为基础,向读者展示了 Visual C++ 软件系统的魅力,而且在第二版中,还介绍了 ODBC 数据库管理以及 OLE 与自动化方面的高级编程技术。

本书主要针对 Visual C++ 5.0 版本,以 Windows NT 4.0 与 Windows 95 或更高版本的 32 位 Windows 为操作系统平台,以 MFC 4.21 为基础,全面介绍了各种 MFC 类库应用程序的开发过程。由于 Win32 与 Win16 有着很大的差异,所以本书对以前的版本作了较大的修改,有些章节是完全重写的,如 Win32 内存管理、动态连接库编程、ActiveX 控件等,但是在 MFC 基本框架方面又以继承为主,这就保证了读者在从 16 位环境转到 32 位环境时程序设计的连续性,也保证了原来的 16 位代码可以很方便地被移植到 32 位系统中来。但本书中,在讲述应用程序框架和文档-视图结构的过程中,涉及到了很多 Win32 的知识,以及 Developer Studio 环境提供的很多实用工具,包括 AppWizard、ClassWizard 等,因此,即使是熟练的程序员也有必要再阅读这些内容。

除了基本的 MFC 应用框架知识外,本书还增加了一些新的内容。在数据库应用开发方面,Visual C++ 5.0 除了支持原来的版本中就已经有的 ODBC 接口外,又增加了新的 DAO(数据访问对象)接口,在 MFC 4.21 的支持下,用户可以快速建立数据库应用程序;上一版本介绍了 OLE 自动化的程序设计方法,随着这几年新技术的不断发展,Microsoft 在 COM 和 OLE 的基础上又引入了 ActiveX 技术,本书用较多的篇幅介绍了 ActiveX 技术,通过这部分内容,读者不仅可领略到 ActiveX 技术的概貌,还可以学习一些深入的程序设计特性;除此之外,本书还介绍了 Internet/Intranet 程序设计技术,包括服务器软件和客户软件的编写以及 IIS 编程;最后,本书介绍了 ActiveX 文档技术,如果读者在从事网络方面的工作,那么相信这些内容可以为您提供基本的技术支持。

Visual C++ 不仅仅是 C++ 语言的集成开发环境,而且与 Win32 紧密相连,所以,利用 Visual C++ 开发系统可以完成各种各样应用程序的开发,从底层软件直到上层直接面向用户的软件都可以用 Visual C++ 来完成开发;而且 Visual C++ 强大的调试功能也为大型复杂软件的开发提供了有效的排错手段。随着软件版本的不断升级,其功能也越来越强大,几乎包括了 Windows 应用的各个方面,所以 Visual C++ 成了 Windows 系统平台上最强大的应用程序开发系统。本书从程序员的角度系统地介绍了 Visual C++ 5.0 的各项编程技术,向读者揭

示了 Visual C++ 5.0 的技术要点,是 Visual C++ 5.0 的标准参考书,是一本不可多得的好书。

本书基本上保持了前两个版本的结构和风格,但由于译者水平有限,难免有不妥之处。尤其是一些新技术术语,在译法上力求正确反映其实际含义又符合中文习惯。若有不当之处,还请读者见谅。

在以前三个版本的基础上,本书全面覆盖了 Visual C++ 开发系统的各个方面,还涉及了 Windows 操作系统的 32 位编程的各项复杂技术。本书可以作为用 Microsoft Visual C++ 开发 Windows 应用程序的标准参考书。

本书从 Windows 和 Visual C++ 的基本技术出发,讲述了应用程序框架的技术细节,同时还包括以下一些技术:

- 基础——事件控制、GDI、对话框、内存管理、SDI 和 MDI、打印和帮助
- 高级话题——多线程、DIB、ODBC 和 DLL
- ActiveX 技术——ActiveX 控件、COM、自动化、统一数据传输、结构化存储和复合文档技术
- Internet 编程——Winsock、MFC WinInet、ISAPI 对 IIS 的扩展和 ActiveX 文档

* 本书随附的光盘包含所有示例程序的源代码。

译者

1998 年 5 月于北大蔚秀园

前言

在这本书即将出版之际,我坐飞机去 San Francisco 参加 Software Development 97 会议。对软件开发者来说,这是最主要的会议,同时也是一个内部演示会。在此之前,我期望 Java 语言在展示会上将会很重要,但当我到达后,我感觉到了“Java Revolution”(Java 革命)的对抗,有 10 000 多名热心者认为 Visual Basic 和 C++ 现在成了“自然馈赠的语言”(legacy language)。我是否要像许多其他的 C++ 作者那样转换到 Java 上来呢?我决定不做这种转换,这倒不是因为市场上已经有了成打的 Java 书籍,而是因为我相信像 C++ 这样的编译语言在将来会更重要。

C++ 和 Java

为什么 C++ 超过 Java? 首先,一种编译语言总是比一种解释语言要快。想象一个带有单元公式和宏的高性能电子数据表格。现在假定由 Java 虚拟机来解释公式和宏的代码,则翻译公式和宏不会很灵快,是不是? 用这种实时编译的方法,不仅在每次装载时都必须编译程序,而且编译代码是否真的能够跟 C++ 编译器的优化结果相比呢?

执行速度是一个因素;对操作系统的访问是另一个因素。因为安全的原因,Java 小应用程序(Java Applets)不能处理像写磁盘和串口访问这样的任务。为了做到平台的独立性,Java 应用程序被限制到操作系统性能的最低公共部分。而一个 Microsoft Windows 的 C++ 程序则更加灵活,因为它任何时候都可以调用所有的 Win32 函数。

Java 仍将是一种重要的语言,但我相信它只是另外一种语言而已,而不是一场革命。如果您需要一个 Internet 小程序或者一个真正与平台无关的应用程序,那就选择 Java;但如果您需要的是效率和灵活性,就应选择 C++。

本书的适用对象

产品名字“Visual C++”误导了很多入,他们认为自己买了一个完全可视的编程系统,类似于 Microsoft Visual Basic,并在刚开始的几天总这样幻想。然而不久,人们认识到他们必须实际编写和阅读 C++ 代码。Visual C++ 向导可以节约时间和提高准确性,但程序员必须理解向导所产生的代码,最主要的,还必须理解 Microsoft Foundation Class(MFC) Library(我们称为 MFC 库)的结构和 Windows 操作系统的内部工作方式。

Visual C++ 和它复杂的应用程序框架(application framework)是为专业程序员准备的,本

书也是如此。我假定读者已经精通了 C 语言——您不用参考手册就可以编一个 if 语句。我还假定您已经接触过 C++ 语言——至少学过这门课程或者读过 C++ 的书籍,但可能还没写过太多的代码。比较一下 C++ 语言的学习和法语的学习。您可以在学校里学习法语,但您不能说得很流利,除非您去法国开始与法国人谈话。而阅读这本书就像是您去法国旅行一样。

不过,我并不假定您已了解 Windows 编程。从我的教学经验来看,我知道精通 C 的程序员能够以 MFC 的方式学习 Windows。此外,虽然了解 C++ 比了解 Win32 API 更重要,但您还是应该知道如何运行 Windows 及运行基于 Windows 的应用程序。

即使您已经对 Win32 API 或者 MFC 库有经验了,这本书仍然可供您学习。首先,它可以帮助您转换到 Win32 编程上来,然后您可以学到一些新的特征,如 Data Access Objects(数据访问对象,简称为 DAO)、ActiveX 控件容器支持,和一些 Windows 95 提供的新控件。如果您尚未领会 Component Object Model(组件对象模型,简称为 COM),则本书介绍的一些重要理论会让您理解 ActiveX 控件。最后,您可以学习 Internet 上的 C++ 编程。

本书没有包括的内容

在一本书里涉及基于 Windows 的编程的所有方面是不可能的,因此我把一些依赖于特定用途的硬件和软件的主题排除了,如 MAPI, TAPI 和通讯口的访问等。我也排除了 ActiveX Template Library(ActiveX 模板库,简称为 ATL),因为它主要是针对 Apple Macintosh 计算机和源码控制(source code control)的。尽管我介绍了如何在一个应用程序中使用 ActiveX 控件,但编写 ActiveX 控件的内容我则推荐 Adam Denning 和他的书《ActiveX Controls Inside Out》(Microsoft Press, 1997)。我建议您从 32 位内存管理、DLL 理论和多线程编程技术开始,但如果您对这些内容缺少了解的话,您还需要参阅 Jeffrey Richter 写的第三版的《Advanced Windows》(Microsoft Press, 1997);另一本有用的书是 George Shepherd 和 Scot Wingo 写的《MFC Internals》(Addison-Wesley, 1996)。

怎样使用本书

当读者开始学习使用 Visual C++ 的时候,最好首先从头到尾阅读此书,它会给您提供最好的学习指导。然后读者就可以将此书作为一本参考手册,通过目录来经常查阅一些内容。由于许多应用程序框架元素之间是彼此紧密相关的,而且许多概念也难以按章节完全分离开来,因此本书不可能像百科全书那样包罗万象。在使用本书时,您最好打开联机帮助(online help),以备查找类和成员函数;如果您更喜欢使用印刷制品的话,那么就可以买一本《Microsoft Visual C++ MFC Library Reference》(Microsoft Press, 1997)。

如果您对 Visual C++ 的 Win16 版本很有经验的话,则可以浏览一下第一部分以对新特

征有个大致的了解,然后可以跳过第二部分的前三章,但请阅读第六章到第十一章,因为这些章里包含了针对 Win32 的一些内容。

本书的组织

正如本书的目录中所列出的那样,本书主要包括六大部分和附录。

第一部分: Windows、Visual C++ 和应用程序框架基础

在这部分中,既有理论阐述,又有实际应用,我力求使这两者达到一种平衡。在这部分中,我对 Win32 和 Visual C++ 的基本组成作了一番评论,进而逐步将应用程序框架以及文档-视图结构(document—view architecture)展现在读者面前。同时,在这部分中还给出了一个通过 MFC 类库中的类来构造的简单的“Hello, world!”程序,该程序只需 30 行代码。

第二部分: MFC 库中的视图类

在假定读者已经熟悉使用基本的 Windows API 的前提下,类库文档材料简捷而又完整地介绍了应用程序框架的基本元素。在第二部分中,我将把读者的注意力集中在应用程序框架的主要构成元素之一——“视图”(view)上,“视图”实际上是一个窗口。在这里,我们将从 C++ 和类库中的类的角度来学习和掌握熟练的 Windows 程序员早已熟知的有关窗口的一些特性。您还将使用 Visual C++ 工具,而这些工具避免了以前 Windows 程序员不得不忍耐的一些繁杂的代码编写工作。

第二部分介绍了很多内容,包括用位图进行图形编程、对话框数据交换、ActiveX 控件使用、32 位内存管理,以及多线程编程。其中的练习将帮助您编写比较复杂的基于 Windows 的程序,但这些程序并没有充分利用应用程序框架的高级特性。

第三部分: 文档-视图结构

这一部分介绍了应用程序框架的核心——文档-视图结构。在这部分中,读者将会了解什么是“文档”(document)(在这里,“文档”并不是指文字处理工作中通常所说的文档,它有更进一步的含义),也将学习如何将文档同第二部分中所介绍的视图联系起来。一旦读者动手编写出了一个自己的文档类,就会禁不住对类库简化文件 I/O 及打印的方式感到非常的惊讶。

另外,读者在这部分中也会接触到命令消息处理(command message processing)、工具栏(toolbar)、状态栏(status bar)、切分框架(splitter frame)以及上下文相关帮助(context-sensitive help)。同时,读者还将学习 Windows 的多文档界面(MDI),这是当前基于 Windows 的应用程序的标准。

第三部分还讨论了如何用 MFC 库编写动态连接库。您将知道在扩展 DLL(extension

DLL)和正规 DLL(regular DLL)之间的区别。如果您过去习惯于 Win16 的 DLL,那么请注意在移到 Win32 时会有一些变化。

第四部分: ActiveX:组件对象模型、自动化和 OLE

要介绍 COM 不是一本书能够完成的。第四部分从 MFC 的角度出发,使您开始学习基本的 COM 理论。接下来您将学习 Automation(自动化),这是连接 C++ 和 Visual Basic for Applications (VBA)的纽带。您还将熟悉统一数据传输(uniform data transfer)和结构化存储(structured storage),还要学习复合文档(compound document)和嵌入对象(embedded object)的基础知识。

第五部分: 数据库管理

基于 Windows 的程序常常要访问大型数据库的信息。Visual C++ 现在支持两种独立的数据库管理方式:Open Database Connectivity(开放数据库连接性,简称为 ODBC)和 Data Access Objects (DAO)。第五部分对每种方式各设有一章。您将学习对这两种方式的更广泛的 MFC 和向导支持,还将看到在 ODBC 和 DAO 之间的相似和不同之处。

第六部分: Internet 程序设计

这部分是第四版的新内容。它从 Internet 的技术指南开始,涉及了 TCP/IP 协议和 Winsock 及 WinInet API。您可以学到如何为 Internet 和 intranet(内联网)编写 C++ 服务端程序和客户端程序,您还可以学到如何编写 ISAPI DLL 以便扩展 Microsoft Internet Information Server。

附 录

附录 A 列出了消息映射宏和对应的消息控制函数原型的列表。ClassWizard 通常会为您生成这些代码,但有时您必须手工添加一些宏和函数条目。

附录 B 描述了 MFC 应用程序框架的运行时类信息和动态创建系统。这是独立于 RTTI(runtime type information,运行时类型信息)的特征,而 RTTI 现在已经成了 ANSI C++ 的一部分。

Win32 和 Win16

我再提一下关于 Win16 编程的问题,因为有很多老的过时的计算机仍然在使用 Windows 3.1。然而,再花费钱去为老的平台编写新程序有些不太值得。本版《Visual C++ 技术内幕》是关于 32 位编程的,使用的是 Microsoft Windows 95 和 Microsoft Windows NT 平台上的 Win32 API。如果您实在需要 16 位编程的话,您可以去看本书的第二版。

Windows 95 和 Windows NT

Visual C++ 5.0 以 Windows 95、Windows NT 4.0 或者更高版本作为平台,这些平台使用同样的用户界面。我建议您使用 Windows NT 作为开发平台,因为它的稳定性好——您可以长时间工作,几个月都不用重启动机器。如果您只用到了 MFC 编程接口,那么编译后的程序在 Windows 95 和 Windows NT 上都可以运行;但如果一个程序调用了特定的 Windows NT 的特性,那就只能在 Windows NT 上运行了。

对 Windows 的进一步探讨: “致 Win32 程序员”这一附栏的目的

本书并没有给出最近一些新的关于 Win32 的专门书籍中所提供的一些技巧和隐含特性。这些书籍大都是针对使用 C 语言的程序员编写的。为了能有效地使用这些书籍,读者必须理解和掌握底层 Win32 API 以及它和类库之间的关系。而且,您还必须懂得 Windows 的消息分发机制和窗口类(window classes)的概念。

书中的“致 Win32 程序员”这一附栏可帮助读者将类库同 Windows 的底层编程联系起来。这些特殊格式的方框中所包含的内容能够帮助熟练的 C 程序员将新的类库概念同他们所熟知的原理联系起来。如果读者还不熟悉底层编程,那么第一遍阅读此书时可跳过这些注释,但当第二遍阅读此书时则应该仔细阅读这些内容。例如,即使您并不需要用 WinMain 函数写一个底层的 Windows 程序,但知道 Windows 操作系统和您的程序是如何相互作用的还是有必要的。

随附的 CD-ROM 光盘的使用

在本书封底的内封还附带着一张光盘,本书中所有示例程序的源文件都在该光盘中。该光盘中还包含可执行文件,因此,您不必再去编译感兴趣的示例程序的可执行文件。为了安装光盘中的文件,请将该光盘插入 CD-ROM 驱动器中,运行 Setup 程序,然后只需按照屏幕提示进行操作即可。

说明: Setup 程序大约将拷贝 10MB 的文件到您的硬盘上。如果您喜欢,也可以手工拷贝单个项目的文件。简单地用目录拷贝(tree-copy)的方法把相应的目录连同其子目录一起从 CD-ROM 拷贝到 c:\vcpp32 下即可。因为每个项目单独在一个目录下,所以并不需要其他项目目录下的文件。(如果您用 Windows Explorer 或者 File Manager 拷贝文件的话,您还必须把文件的只读属性去掉。)

说明: CD-ROM 光盘上的大多数文件是长文件名的,如果您使用 Windows 95 且您的 CD-ROM 驱动器用的是实模式的驱动程序,则您可以看到这些文件名被截短了,不再是原来的文件名或者目录名。然而,Setup 程序仍然可以正常进行,它会从 CD-ROM 上特定的 \SETUP 目录下拷贝文件,并用长文件名重命名。您可以在硬盘上浏览这些文件。反过来,您也可以使用 8.3 别名浏览 CD-ROM 上 \SETUP 目录下的文件。

对于传统的用 C 语言编写的 Windows API 程序,源程序会告诉人们一切,但对于类库应用程序框架来说,事情就没有这么简单了。在类库应用程序框架中,大部分的 C++ 源代码都是由 AppWizard 产生的,而资源则是由资源编辑器生成的。前面几章的例子大都是一步一步地教读者如何使用有关工具来生成自己的源代码文件,建议读者最好能按照规定的步骤去试几个例子,在这些例子中需要用户自己动手编写的代码是很少的。中间几章的例子可以使用附带的光盘所提供的源代码,但也必须仔细阅读创建过程,以便对资源编辑器及 Wizard 的作用有更深的体会。在最后几章中则没有列出例子的所有源代码,读者只好从附带的光盘上去查找这些例子的完整源代码了。

致 Win32 程序员:Unicode

直到最近,基于 Windows 的程序还只是使用 ANSI 字符集,共包含 256 个单字节字符。而针对亚洲软件市场的开发者则正在往 Unicode 字符集上移植,Unicode 字符集包含了 65 536 个 2 字节(wide)字符。第三种选择是双字节字符集(Double-Byte Character Set, DBCS),包含了 1-字节字符和 2-字节字符,但 DBCS 并不受欢迎。

MFC 类库和运行时库(runtime library)支持 Unicode 应用程序。如果您定义了常量 _UNICODE,并按照联机文档中介绍的步骤操作的话,那么您所有的字符变量和常量字符串都是双字节(wide)的,编译器将产生双字节(wide-character)版本 Win32 函数的调用。这里假定您在声明字符指针和数组时用了特定的宏,例如 TCHAR 和 _T。

然而,如果您试着在 Windows 95 下运行 MFC Unicode 应用程序,则您可能会碰到问题,因为 Windows 95 内部并不支持 Unicode。即使 Windows 95 有双字节版本 Win32 函数,这些函数也只是返回错误码。而 Windows NT 的内部使用了 Unicode,并有两个版本的 Win32 函数来处理字符。如果您调用一个单字节版本的函数,则 Windows NT 会在函数的入口和出口做必要的双字节转换。

本书的示例程序都没有设置成 Unicode。所有的程序都使用单字节类型,如 char 和单字节字符串常量,且没有定义 _UNICODE。如果您在 Windows NT 下运行这些例子,操作系统会作必要的转换;而如果在 Windows 95 下运行,则接口是纯粹单字节的。

强制使用双字符的一个地方是 COM,所有的非 MFC 的 COM 函数(除了 DAO 函数)中的字符串和字符参数都要求双字节(OLECHAR)。如果您编写了一个非 Unicode 的程

序,则必须自己做转换,不过 MFC 的 CString 类和多个宏可以帮助您做这种转换。

如果您想编写 Unicode 的应用程序,请阅读 Jeffrey Richter 的《Advanced Windows》一书中的有关 Unicode 的章节,也可以参阅 Visual C++ 联机文档中关于 Unicode 的资料。

技术说明及示例程序

Developer Studio 的 InfoView 窗口中包含了本书提到的技术说明及示例程序。用数字进行标识的技术说明被包含在下面的标题中:

Developer Products
Visual C++
Microsoft Foundation Class Reference
MFC Technical Notes

Visual C++ 5.0 的 CD-ROM 中还包含了许多 MFC 示例程序,其中包括本书中所提到的,这些示例程序用名字来标识。这些程序在下面的标题中:

Developer Products
Visual C++
Visual C++ Samples
MFC Samples

支持信息

我们作了很大的努力来确保这本书和随附光盘上的内容的准确性。Microsoft 出版社通过后面的 Web 地址为本书提供了修正:<http://mspress.microsoft.com/mspress/support>。

如果您对本书和随附光盘有任何意见、问题或者好的建议,请通过普通邮件或电子邮件发送给 Microsoft Press:

Microsoft Press
Attn: Inside Visual C++ Editor
One Microsoft Way
Redmond, WA 98052-6399
MSPINPUT@MICROSOFT.COM

请注意,对产品的支持信息请不要发送到上述邮件地址。要获得 Microsoft Visual C++ 的支持信息,您可以打技术支持电话(425)635-7007,时间为每个工作日的上午 6 点到下午 6 点(美国西部标准时间)。Microsoft 还在网上提供了关于 Visual C++ 的信息 <http://www.microsoft.com/visualc/>, 和关于 Microsoft Developer Network 的信息 <http://www.microsoft.com/MSDN/>。

目 录

前 言	29
第一部分 Windows、Visual C++ 和应用程序框架基础	
第一章 Microsoft Windows 和 Visual C++	3
1.1 Windows 的编程模式	3
1.1.1 消息处理	3
1.1.2 Windows 的图形设备接口 (GDI)	4
1.1.3 基于资源的程序设计	4
1.1.4 内存管理	4
1.1.5 动态连接库 (DLL)	5
1.1.6 Win32 应用程序编程接口	5
1.2 Visual C++ 的组成	5
1.2.1 Microsoft Developer Studio 97 和创建过程	6
1.2.2 资源编辑器——Workspace ResourceView	8
1.2.3 C/C++ 编译器	8
1.2.4 资源编译器	9
1.2.5 连接器	9
1.2.6 调试器	9
1.2.7 AppWizard	9
1.2.8 ClassWizard	10
1.2.9 源程序浏览器	10
1.2.10 联机帮助	11
1.2.11 Windows 诊断工具	12
1.2.12 源代码控制	12
1.2.13 Gallery	12
1.2.14 Microsoft 基本类库 4.21 版	13
1.2.15 Microsoft ActiveX 模板库	13
第二章 Microsoft 基本类库应用程序框架	14
2.1 为什么要使用应用程序框架	14
2.1.1 学习曲线	17

2.2 什么是应用程序框架	18
2.2.1 应用程序框架和 MFC 库	18
2.2.2 应用程序框架示例	18
2.3 MFC 库消息映射	21
2.3.1 文档和视图	22

第二部分 MFC 库中的视图类

第三章 从“Hello, world!”着手学习 AppWizard	27
3.1 什么是视图	27
3.2 单文档界面(SDI)和多文档界面(MDI)	28
3.3 “不做任何事情”的应用程序	28
3.4 CEx03aView 视图类	32
3.5 在视窗内绘图——Windows 的图形设备接口	32
3.5.1 OnDraw 成员函数	33
3.5.2 Windows 设备环境	33
3.5.3 往 EX03A 程序中加入绘制代码	33
3.6 资源编辑器简介	35
3.6.1 ex03a.rc 的内容	35
3.6.2 运行对话框资源编辑器	36
3.7 Win32 Debug 目标和 Win32 Release 目标	37
3.8 允许诊断宏	37
3.9 对预编译头文件的理解	38
3.10 两种方法运行程序	39
第四章 基本事件处理、映射模式和滚动视图	40
4.1 获取用户输入——消息映射函数	40
4.1.1 消息映射	40
4.1.2 保存视图的状态-类数据成员	41
4.1.3 初始化视图类数据成员	42
4.1.4 使矩形区域无效	42
4.1.5 Windows 的客户区域	42
4.1.6 CRect, CPoint 和 CSize 类的运算	43
4.1.7 一个点是否在矩形内	43
4.1.8 CRect 的 LPRECT 操作符	43
4.1.9 一个点是否在椭圆内	44
4.2 EX04A 示例程序	44