

图书基本信息

书名：<<Visual C++ 技术内幕 (第四版)>>

13位ISBN编号：9787302032434

10位ISBN编号：7302032432

出版时间：1999-1

出版时间：清华大学出版社

作者：David J. Kruglinski

译者：潘爱民/王国印

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

内容概要

在前三版的基础上，本书向读者详尽地介绍了Visual C++开发系统和复杂的32位Windows编程技术，全面诠释了Microsoft的这个强大而复杂的开发工具的功能。

本书从对Windows, Visual C++和应用程序框架的技术性概述入手，讨论了下列课题：基础知识——事件处理，GDI，对话框，内存管理，SDI和MDI，打印，以及帮助；高级课题——多线程，DIB，ODBC，DLL等等。

作者简介

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。

”

LowellSkoog，滑翔机飞行伙伴

书籍目录

- 目录
- 前言
- 第一部分 Windows、VisualC++ 和应用程序框架基础
- 第一章 MicrosoftWindows 和VisualC++
 - 1.1Windows的编程模式
 - 1.1.1消息处理
 - 1.1.2Windows的图形设备接口 (GDI)
 - 1.1.3基于资源的程序设计
 - 1.1.4内存管理
 - 1.1.5动态连接库 (DLL)
 - 1.1.6Win32应用程序编程接口
 - 1.2VisualC++ 的组成
 - 1.2.1MicrosoftDeveloperStudio97和创建过程
 - 1.2.2资源编辑器 WorkspaceResourceView
 - 1.2.3C/C++ 编译器
 - 1.2.4资源编译器
 - 1.2.5连接器
 - 1.2.6调试器
 - 1.2.7AppWizard
 - 1.2.8ClassWizard
 - 1.2.9源程序浏览器
 - 1.2.10联机帮助
 - 1.2.11Windows诊断工具
 - 1.2.12源代码控制
 - 1.2.13Gallery
 - 1.2.14Microsoft基本类库4.21版
 - 1.2.15MicrosoftActiveX 模板库
- 第二章 Microsoft基本类库应用程序框架
 - 2.1为什么要使用应用程序框架
 - 2.1.1学习曲线
 - 2.2什么是应用程序框架
 - 2.2.1应用程序框架和MFC库
 - 2.2.2应用程序框架示例
 - 2.3MFC库消息映射
 - 2.3.1文档和视图
- 第二部分 MFC 库中的视图类
- 第三章 从 “ Hello , world ! ” 着手学习AppWizard
 - 3.1什么是视图
 - 3.2单文档界面 (SDI) 和多文档界面 (MDI)
 - 3.3 “ 不做任何事情 ” 的应用程序
 - 3.4CEx03aView视图类
 - 3.5在视窗内绘图 Windows的图形设备接口
 - 3.5.1OnDraw成员函数
 - 3.5.2Windows设备环境

- 3.5.3往EX03A程序中加入绘制代码
- 3.6资源编辑器简介
 - 3.6.1ex03a.rc的内容
 - 3.6.2运行对话框资源编辑器
- 3.7Win32Debug目标和Win32Release目标
- 3.8允许诊断宏
- 3.9对预编译头文件的理解
- 3.10两种方法运行程序
- 第四章 基本事件处理、映射模式和滚动视图
 - 4.1获取用户输入 消息映射函数
 - 4.1.1消息映射
 - 4.1.2保存视图的状态 - 类数据成员
 - 4.1.3初始化视图类数据成员
 - 4.1.4使矩形区域无效
 - 4.1.5Windows 的客户区域
 - 4.1.6CRect, CPoint和CSize类的运算
 - 4.1.7一个点是否在矩形内
 - 4.1.8CRect的LPRECT操作符
 - 4.1.9一个点是否在椭圆内
 - 4.2Ex04A示例程序
 - 4.3对EX04A使用ClassWizard
 - 4.3.1同时使用AppWizard 和ClassWizard
 - 4.4映射模式
 - 4.4.1MM_TEXT映射模式
 - 4.4.2 “ 固定比例 ” 映射模式
 - 4.4.3 “ 可变比例 ” 映射模式
 - 4.4.4坐标变换
 - 4.4.5EX04B示例程序 转换到MM_HIMETRIC映射模式
 - 4.5滚动视图
 - 4.5.1窗口比人们所见到的要大
 - 4.5.2滚动条
 - 4.5.3滚动方式的选择
 - 4.5.4 OnInitialUpdate函数
 - 4.5.5接收键盘输入
 - 4.5.6EX04C滚动示例程序
 - 4.6其他Windows消息
 - 4.6.1WM_CREATE消息
 - 4.6.2WM_CLOSE消息
 - 4.6.3WM_QUERYENDSESSION消息
 - 4.6.4WM_DESTROY消息
 - 4.6.5WM_NCDESTROY消息
- 第五章 图形设备接口 (GDI)、颜色及字体
 - 5.1设备环境类
 - 5.1.1显示设备环境类CClientDC和CWindowDC
 - 5.1.2构造和析构CDC对象
 - 5.1.3设备环境的状态
 - 5.1.4CPaintDC类

5.2 GDI对象

5.2.1 GDI对象的构造与析构

5.2.2 跟踪GDI对象

5.2.3 库存的GDI对象

5.2.4 GDI选择的有效期

5.3 Windows的颜色映射

5.3.1 标准视频图形阵列 (V以) 显示卡

5.3.2 256色显示卡

5.3.3 16位颜色显示卡

5.3.4 24位颜色显示卡

5.4 字体

5.4.1 字体是GDI对象

5.4.2 选择字体

5.4.3 打印字体

5.4.4 显示字体

5.4.5 显示器的逻辑英寸和物理英寸

5.4.6 计算字符高度

5.5 EX05A示例程序

5.5.1 EX05A程序的组成元素

5.6 EX05B程序

5.6.1 EX05B程序的组成元素

5.7 EX05C示例程序 再次使用CScrollView

5.7.1 EX05C程序的组成元素

5.7.2 CScrollView的SetScaleToFitSize模式

5.7.3 在滚动视图中使用逻辑twips映射模式

第六章 模式对话框和Windows通用控件

6.1 模式和无模式对话框

6.2 资源和控件

6.3 模式对话框的编程

6.4 包罗万象的对话框 EX06A示例程序

6.4.1 创建对话框资源

6.4.2 ClassWizard和对话框类

6.4.3 将对话框与视图相连接

6.4.4 深入理解EX06A应用程序

6.5 对话框程序的改进

6.5.1 获得OnOK退出控制

6.5.2 OnCancel处理

6.5.3 和滚动条控件挂上钩

6.6 控件的标识: CWnd指针和控件ID

6.7 设置对话框的背景色和其中控件的颜色

6.8 在对话框窗口内绘图

6.9 在运行时加入对话框控件

6.10 利用其他的控件特性

6.11 Windows通用控件

6.11.1 进度指示器控件

6.11.2 滑杆条 (Trackbar) 控件

6.11.3 微调按钮控件

- 6.11.4列表控件
- 6.11.5树状控件
- 6.11.6WM NOTIFY消息
- 6.12EX06B示例程序
- 6.13其他Windows 通用控件
- 第七章 无模式对话框和Windows通用对话框类
- 7.1无模式对话框
 - 7.1.1创建无模式对话框
 - 7.1.2用户定义的消息
 - 7.1.3对话框的归属
 - 7.1.4无模式对话框的例子 EX07A
- 7.2CFormView类 另外一种可供选择的无模式对话框
- 7.3Windows通用对话框
 - 7.3.1直接使用CFileDialog类
 - 7.3.2从通用对话框类进行派生
 - 7.3.3嵌套对话框
 - 7.3.4使用CFileDialog的一个例子 EX07B
 - 7.3.5CFileDialog的其他可定制特性
- 第八章 使用ActiveX控件
- 8.1ActiveX控件和普通的Windows控件
 - 8.1.1普通的Windows 控件 参考体系
 - 8.1.2ActiveX控件与普通控件的相同点
 - 8.1.3ActiveX控件与普通控件的不同点 属性和方法
- 8.2安装ActiveX控件
- 8.3日历控件
- 8.4ActiveX控件包容器编程
 - 8.4.1属性访问
 - 8.4.2ClassWizard为ActiveX控件产生的C++类
 - 8.4.3AppWizard对ActiveX控件的支持
 - 8.4.4ClassWizard和包容器对话框
 - 8.4.5对话框类数据成员和控件类的用法
 - 8.4.6映射ActiveX控件事件
 - 8.4.7把ActiveX控件锁定在内存里
- 8.5EX08A示例程序 ActiveX控件对话框包容器
- 8.6ActiveX控件在HTML文件中的使用
- 8.7在运行时创建ActiveX控件
- 8.8EX08B示例程序 Web浏览器ActiveX控件
- 8.9EX08C示例程序 一个完整的双窗口浏览器
- 8.10图形属性
- 8.11约束属性 改变通知
- 8.12其他的ActiveX控件
- 第九章 Win32内存管理
- 9.1进程和内存空间
 - 9.1.1Windows95进程地址空间
 - 9.1.2WindowsNT进程地址空间
- 9.2虚拟内存如何工作
- 9.3VirtualAlloc函数 占用和保留内存

- 9.4 Windows 堆和GlobalAlloc 函数族
- 9.5 小内存堆, C++ 的new和delete操作符, _heapmin
- 9.6 内存映射文件
- 9.7 访问资源
- 9.8 管理动态内存的一些提示
- 9.9 常量数据的优化存储
- 第十章 位图
- 10.1 GDI位图和与设备无关的位图
- 10.2 彩色位图和单色位图
- 10.3 使用GDI位图
 - 10.3.1 从资源中装入GDI位图
 - 10.3.2 显示映射模式的作用
 - 10.3.3 对位进行伸缩处理
 - 10.3.4 EX10A示例程序
- 10.4 利用位图改进屏幕显示
 - 10.4.1 EX10B示例程序
 - 10.4.2 Windows动画
- 10.5 DIB和CDib类
 - 10.5.1 关于调色板编程的术语
 - 10.5.2 DIB、象素和颜色表
 - 10.5.3 BMP文件中DIB的结构
 - 10.5.4 DIB访问函数
 - 10.5.5 CDib类
 - 10.5.6 DIB显示性能
 - 10.5.7 EX10C示例程序
- 10.6 进一步使用DIB
 - 10.6.1 Load Image函数
 - 10.6.2 DrawDibDraw函数
- 10.7 在按钮上设置位图
 - 10.7.1 EX10D示例程序
 - 10.7.2 进一步使用位图按钮
- 第十一章 Windows消息处理和多线程编程
- 11.1 Windows消息处理
 - 11.1.1 单线程程序处理消息的过程
 - 11.1.2 交出控制
 - 11.1.3 计时器
 - 11.1.4 EX11A示例程序
- 11.2 空状态处理
- 11.3 多线程编程
 - 11.3.1 编写辅助线程函数和启动线程
 - 11.3.2 主线程如何和辅助线程通话
 - 11.3.3 辅助线程如何和主线程通话
 - 11.3.4 EX11B示例程序
 - 11.3.5 用事件使线程同步
 - 11.3.6 EX11C示例程序
 - 11.3.7 线程阻塞
 - 11.3.8 排斥区

- 11.3.9互斥体 (mutex) 和信号
- 11.3.10用户界面线程 (UI线程)
- 第三部分 文档 - 视图结构
- 第十二章 菜单、键盘加速键、复文本编辑控件和属性表
 - 12.1主框架窗口和文档类
 - 12.2Windows的菜单
 - 12.3键盘加速键
 - 12.4命令处理
 - 12.4.1派生类中的命令消息控制
 - 12.4.2更新命令用户界面消息控制函数
 - 12.4.3对话框发送的命令
 - 12.5应用程序框架的内含菜单项
 - 12.6菜单项的允许/禁用
 - 12.7MFC文本编辑选择
 - 12.7.1CE出tView类
 - 12.7.2CRichEditView类
 - 12.7.3CRichEditCtrl类
 - 12.8ex12A示例程序
 - 12.9属性表
 - 12.9.1创建属性表
 - 12.9.2属性表数据交换
 - 12.10改进EX12A示例程序
 - 12.11Apply按钮的处理过程
 - 12.12CMenu类
 - 12.13创建浮动的弹出式菜单
 - 12.14扩展命令处理
- 第十三章 工具栏和状态栏
 - 13.1控制栏和应用程序框架
 - 13.2工具栏
 - 13.2.1工具栏位图
 - 13.2.2按钮的状态
 - 13.2.3工具栏和命令消息
 - 13.2.4工具栏更新命令UI消息控制函数
 - 13.3工具提示
 - 13.4寻找主框架窗口
 - 13.5EX13A工具栏示例程序
 - 13.6状态栏
 - 13.6.1状态栏的定义
 - 13.6.2信息行
 - 13.6.3状态指示器
 - 13.6.4获得对状态栏的控制
 - 13.7EX13B状态栏示例程序
- 第十四章 可重用框架窗口基类
 - 14.1为什么可重用基类难以设计
 - 14.2CPersistentFrame类
 - 14.3CFrameWnd类和ActivateFrame成员函数
 - 14.4PreCreateWindow成员函数

- 14.5 Windows注册表
- 14.6 使用CString类
- 14.7 最大化窗口的位置
- 14.8 控制栏状态和注册表
- 14.9 静态数据成员
- 14.10 默认窗口矩形
- 14.11 EX14A 示例程序
- 14.12 MDI应用程序中的持续框架
- 第十五章 文档与视图的分离
- 15.1 文档 - 视图之间的相互作用函数
 - 15.1.1 CView GetDocument函数
 - 15.1.2 CDocument UpdateAllViews 函数
 - 15.1.3 CView OnUpdate函数
 - 15.1.4 CView OnInitialUpdate函数
 - 15.1.5 CDocument OnNewDocument函数
- 15.2 最简单的文档 - 视图应用程序
- 15.3 CFormView类
- 15.4 CObject类
- 15.5 诊断信息转储
 - 15.5.1 TRACE宏
 - 15.5.2 atxDump对象
 - 15.5.3 信息转储环境及CObjct类
 - 15.5.4 未被删除对象的自动信息转储
- 15.6 EX15A 示例程序
- 15.7 更加高级的文档 - 视图之间的相互作用
- 15.8 CDocument DeleteContents函数
- 15.9 CObList集合类
 - 15.9.1 对先入先出列表使用CObList类
 - 15.9.2 在CObList中进行迭代 POSITION变量
 - 15.9.3 CTypedPtrList模板集合类
 - 15.9.4 信息转储环境及集合类
- 15.10 EX15B 示例程序
 - 15.10.1 资源要求
 - 15.10.2 代码要求
 - 15.10.3 CStudentDoc
 - 15.10.4 CStudentView
 - 15.10.5 测试EX15B应用程序
- 15.11 留给读者的两个练习
- 第十六章 读和写文档 SDI应用程序
- 16.1 什么是序列化
 - 16.1.1 磁盘文件和归档 (Archives)
 - 16.1.2 使类可序列化
 - 16.1.3 编写Serialize函数
 - 16.1.4 从归档中载入 内嵌对象和指针
 - 16.1.5 使集合序列化
 - 16.1.6 Serialize函数和应用程序框架
- 16.2 SDI应用程序

- 16.2.1 Windows应用程序对象
- 16.2.2 文档模板类
- 16.2.3 文档模板资源
- 16.2.4 SDI文档的多视图
- 16.2.5 创建空文档 CWinApp OnFileNew函数
- 16.2.6 文档类的OnNewDocument函数
- 16.2.7 连接FileOpen与序列化代码 OnFileOpen函数
- 16.2.8 文档类的DeleteContents函数
- 16.2.9 将FileSave和FileSaveAs与序列化代码相连接
- 16.2.10 文档的“脏”标志
- 16.3 EX16A 序列化SDI示例程序
- 16.3.1 CStudent
- 16.3.2 CEx16aApp
- 16.3.3 CFrameWnd
- 16.3.4 CStudentDoc
- 16.3.5 CStudentView
- 16.3.6 测试EX16A应用程序
- 16.4 从Explorer启动及拖放
- 16.4.1 程序注册
- 16.4.2 在文档上双击鼠标
- 16.4.3 允许拖放
- 16.4.4 程序启动参数
- 16.4.5 试验用Explorer启动程序及拖放功能
- 第十七章 读和写文档 MDI应用程序
- 17.1 MDI应用程序
- 17.1.1 一个典型的具有MFC库风格的MDI应用程序
- 17.1.2 MDI应用程序对象
- 17.1.3 MDI文档模板类
- 17.1.4 MDI框架窗口和MDI子窗口
- 17.1.5 主框架和文档模板资源
- 17.1.6 创建空文档 CWinApp OnFileNew函数
- 17.1.7 为现有文档创建新的视图
- 17.1.8 载人和存储文档
- 17.1.9 多文档模板
- 17.1.10 Explorer启动及拖放
- 17.2 EX17A示例程序
- 17.2.1 CEx17aApp
- 17.2.2 CMainFrame
- 17.2.3 CChildFrame
- 17.2.4 测试EX17A应用程序
- 第十八章 打印和打印预览
- 18.1 Windows的打印
- 18.1.1 标准打印对话框
- 18.1.2 交互选择打印页码
- 18.1.3 显示页和打印页
- 18.2 打印预览
- 18.3 有关打印机的程序设计

- 18.3.1打印机设备环境和CViewOnDraw函数
- 18.3.2CView OnPrint函数
- 18.3.3准备设备环境 CView OnPrePareDC函数
- 18.3.4打印任务的开始和结束
- 18.4EX18A示例程序 一个所见即所得的打印程序
- 18.5读取打印机矩形框
- 18.6再看模板集合类 CArray类
- 18.7EX18B示例程序 多页打印程序
- 第十九章 切分窗口和多视图
- 19.1切分窗口
- 19.2多视图选择
- 19.3动态和静态切分窗口
- 19.4EX19A示例程序 具有单个视图类的SDI动态切分
- 19.4.1与切分有关的资源
- 19.4.2CMainFrame
- 19.4.3测试EX19A应用程序
- 19.5EX19B示例程序 双视图类SDI静态切分
- 19.5.1CHexView
- 19.5.2CMainFrame
- 19.5.3测试EX19B应用程序
- 19.6EX19C示例程序 切换没有切分的视图类
- 19.6.1资源要求
- 19.6.2CMainFrame
- 19.6.3测试Ex19C应用程序
- 19.7Ex19D示例程序 多视图类MDI应用程序
- 19.7.1资源要求
- 19.7.2CEx19dApp
- 19.7.3CMainFrame
- 19.7.4测试EX19D应用程序
- 第二十章 上下文相关帮助
- 20.1Windows 的WinHelp程序
- 20.1.1复文本格式 (RTF)
- 20.1.2编写简单的帮助文件
- 20.1.3改进的TableOfContents
- 20.2应用程序框架与WinHelp
- 20.2.1调用WinHelp
- 20.2.2使用搜索字符串
- 20.2.3从应用程序的菜单调用WinHelp
- 20.2.4帮助上下文别名
- 20.2.5确定帮助上下文
- 20.2.6F1帮助
- 20.2.7Shift - F1帮助
- 20.2.8消息框帮助 AfxMessageBox 函数
- 20.2.9一般的帮助
- 20.3帮助示例 不需要任何编程
- 20.4MAKEHEL处理
- 20.5帮助命令处理

- 20.5.1 F1处理过程
- 20.5.2 Shift - F1处理过程
- 20.6 帮助命令处理示例程序 EX20 B
 - 20.6.1 头文件要求
 - 20.6.2 CStringView
 - 20.6.3 CHexView
 - 20.6.4 资源要求
 - 20.6.5 帮助文件要求
 - 20.6.6 测试EX20B应用程序
- 第二十一章 动态连接库
 - 21.1 基本DLL理论
 - 21.1.1 导入如何与导出相匹配
 - 21.1.2 隐式连接和显式连接
 - 21.1.3 符号连接和序号连接
 - 21.1.4 DLL入口点 DllMain
 - 21.1.5 实例句柄 装载资源
 - 21.1.6 客户程序如何找到DLL
 - 21.1.7 调试DLL
 - 21.2 MFC DLL 扩展的和正规的
 - 21.2.1 共享的MFC DLL和Windows DLL
 - 21.2.2 MFC扩展DLL 导出类
 - 21.2.3 MFC扩展DLL资源搜索的顺序
 - 21.2.4 EX21A示例程序 MFC扩展DLL
 - 21.2.5 EX21B示例程序 DLL测试客户程序
 - 21.2.6 MFC正规DLL CWinApp派生类
 - 21.2.7 使用AFX_MANAGE_STATE宏
 - 21.2.8 MFC正规DLL资源搜索顺序
 - 21.2.9 EX21C示例程序 MFC正规DLL
 - 21.2.10 修改EX21B示例程序 加入代码以测试ex21c.dll
 - 21.3 自定义控件DLL
 - 21.3.1 什么是自定义控件
 - 21.3.2 自定义控件的窗口类
 - 21.3.3 MFC库和WndProc函数
 - 21.3.4 自定义控件通知消息
 - 21.3.5 发送给控件的用户定义消息
 - 21.3.6 EX21D示例程序 自定义控件
 - 21.3.7 再修改EX21B示例程序 加入代码以测试ex21d.dll
- 第二十二章 不带文档或视图类的MFC程序
 - 22.1 EX22A示例程序 一个基于对话框的应用程序
 - 22.1.1 应用程序类InitInstance函数
 - 22.1.2 对话框类和程序图标
 - 22.2 EX22B示例 一个SDI应用程序
 - 22.2.1 应用程序类InitInstance函数
 - 22.2.2 CMainFrame类
 - 22.3 EX22C示例程序 一个MDI应用程序
 - 22.3.1 应用类程序InitInstance函数
 - 22.3.2 CMainFrame类

- 22.3.3 CChildFrame类
- 22.3.4 资源要求
- 22.4 定制 (Custom) AppWizard
- 第四部分 ActiveX : 组件对象模型、自动化和OLE
- 第二十三章 组件对象模型
- 23.1 ActiveX技术背景
- 23.2 组件对象模型 (COM)
- 23.2.1 COM所解决的问题
- 23.2.2 COM的本质
- 23.2.3 什么是COM接口
- 23.2.4 IUnknown接口和QueryInterface成员函数
- 23.2.5 引用计数 : AddRef和Release函数
- 23.2.6 类厂 (classfactory)
- 23.2.7 CCmdTarget类
- 23.2.8 EX23A示例程序 一个模拟的COM
- 23.3 MFC库中实际的COM
- 23.3.1 COM的CoGetClassObject函数
- 23.3.2 COM和Windows注册表
- 23.3.3 运行时对象注册
- 23.3.4 COM客户程序如何调用进程内组件
- 23.3.5 COM客户程序如何调用进程外组件
- 23.3.6 MFC接口宏
- 23.3.7 MFC的COleObjectFactory类
- 23.3.8 AppWizard/ClassWizard对COM进程内组件的支持
- 23.3.9 MFCCOM客户程序
- 23.3.10 EX23B示例程序 MFCCOM进程内组件
- 23.3.11 EX23C示例程序 MFCCOM客户
- 23.4 包容、集合和继承
- 第二十四章 自动化
- 24.1 C + + 与VisualBasicforApplications (VBA) 之间的连接
- 24.2 自动化客户和自动化组件
- 24.3 MicrosoftExcel 一种比VisualBasic还好的VisualBasic
- 24.4 属性 (property)、方法 (method) 和集合
- 24.5 自动化所解决的问题
- 24.6 IDispatch接口
- 24.7 自动化程序设计的选择
- 24.8 MFCIDispatch的实现
- 24.9 一个MFC自动化组件
- 24.10 一个MFC自动化客户程序
- 24.11 使用编译器 # import指令的自动化客户程序
- 24.12 VARIANT类型
- 24.13 COleVariant类
- 24.14 Invoke的参数和返回值的类型转换
- 24.15 自动化示例
- 24.15.1 EX24A自动化组件EXE示例 不带任何用户界面
- 24.15.2 EX24B自动化组件DLL示例程序
- 24.15.3 EX24CSDI自动化组件EXE示例 带用户界面

- 24.15.4EX24D自动化客户示例程序
- 24.15.5EX24E自动化客户示例程序
- 24.16VBA早绑定
 - 24.16.1注册类型库
 - 24.16.2组件如何注册自己的类型库
 - 24.16.3ODL文件
 - 24.16.4Excel如何使用类型库
 - 24.16.5为什么使用早绑定
 - 24.16.6客户 - 组件的快速连接
- 第二十五章 统一数据传输 剪贴板传输以及OL拖放
 - 25.1IDataObject接口
 - 25.2IDataObject如何增强标准剪贴板的支持
 - 25.3FORMATETC和STGMEDIUM结构
 - 25.3.1FORMATETC
 - 25.3.2STGMEDIUM
 - 25.3.3IDataObject接口成员函数
 - 25.4MFC统一数据传输支持
 - 25.4.1COleDataSource类
 - 25.4.2COleDataObject类
 - 25.4.3MFC数据对象剪贴板传输
 - 25.5MFCCRectTracker类
 - 25.5.1CRectTracker类成员函数
 - 25.5.2CRectTracker类矩形坐标转换
 - 25.6EX25A示例程序 数据对象剪贴板
 - 25.6.1CMainFrame类
 - 25.6.2CE25aDoc类
 - 25.6.3CEx25aView类
 - 25.7MFC拖放
 - 25.7.1传输的源方
 - 25.7.2传输的目标方
 - 25.7.3拖放的顺序
 - 25.8EX25B示例程序 OLE拖放
 - 25.8.1CEx25bDoc类
 - 25.8.2CEx25bView类
 - 25.9Windows应用程序和拖放 CObjview
 - 第二十六章 结构化存储
 - 26.1复合文件
 - 26.2存储 (Storage) 和IStorage接口
 - 26.2.1获取IStorage指针
 - 26.2.2释放STATSTG内存
 - 26.2.3对存储对象的元素进行枚举
 - 26.2.4进程间共享存储
 - 26.3流 (Stream) 和IStream接口
 - 26.3.1IStream接口成员函数
 - 26.3.2IStream编程
 - 26.4ILockBytes接口
 - 26.5EX26A示例程序 结构化存储

- 26.5.1 菜单
- 26.5.2 CEx26aView类
- 26.5.3 辅助线程
- 26.6 结构化存储和持久COM对象
 - 26.6.1 IPersistStorage 接口
 - 26.6.2 IPersistStream接口
 - 26.6.3 IPersistStream编程
- 26.7 EX26B示例程序 一个持久DLL组件
- 26.8 EX26C示例程序 一个持久存储客户程序
- 26.9 复合文件碎片的处理
- 26.10 复合文件的其他优点
- 第二十七章 OLE嵌入组件和容器
 - 27.1 嵌入和实地激活 (可视编辑)
 - 27.2 最小服务器和完备服务器 (组件) 链接
 - 27.3 可视编辑前景暗淡的一面
 - 27.4 Windows 图元文件和内嵌对象
 - 27.5 组件程序的MFCOLE结构
 - 27.6 EX27A示例程序 MFC实地激活最小服务器
 - 27.6.1 MDI嵌入组件
 - 27.6.2 实地组件调整大小的思路
 - 27.7 容器 - 组件相互作用
 - 27.7.1 使用组件的IOleObject接口
 - 27.7.2 装入和保存组件自身的数据 组件文档
 - 27.7.3 剪贴板数据传输
 - 27.7.4 获得组件的图元文件
 - 27.7.5 进程内控制器的角色
 - 27.8 组件状态
 - 27.8.1 容器接口
 - 27.8.2 通报连接
 - 27.8.3 剪贴版图元文件
 - 27.9 接口简要说明
 - 27.9.1 IOleObject接口
 - 27.9.2 IViewObject2接口
 - 27.9.3 IOleClientSite接口
 - 27.9.4 IAdviseSink接口
 - 27.10 OLE辅助函数
 - 27.11 OLE嵌入容器应用程序
 - 27.11.1 MFC对OLE容器的支持
 - 27.11.2 一些容器限制
 - 27.11.3 容器特性
 - 27.12 EX27B示例程序 嵌入容器
 - 27.12.1 CEx27bView类
 - 27.12.2 CEx27bDoc类
 - 27.13 EX27C示例程序 OLE嵌入组件
 - 27.13.1 CEx27cView类
 - 27.13.2 CEx27cDoc类
- 第五部分 数据库管理

第二十八章 Microsoft ODBC 数据库管理

- 28.1 数据库管理的优势
- 28.2 结构化查询语言
- 28.3 ODBC 标准
 - 28.3.1 ODBC 结构
 - 28.3.2 ODBC SDK 编程
- 28.4 MFC ODBC 类 CRecordset 和 CDatabase
 - 28.4.1 记录集中关于行的计数
 - 28.4.2 处理 odbc 异常
- 28.5 学生注册数据库
- 28.6 EX28A 记录集示例程序
- 28.7 EX28A 示例程序中的元素
 - 28.7.1 记录集类与应用程序之间的连接
 - 28.7.2 CEx28aView 类的 OnInitialUpdate 成员函数
 - 28.7.3 CEx28aView 类的 OnDraw 成员函数
- 28.8 筛选器和排序字符串
- 28.9 连接两个数据库表
- 28.10 MFCCRecordView 类
- 28.11 EX28B 记录查看示例程序
- 28.12 多记录集
 - 28.12.1 EX28C 多记录集示例程序
 - 28.12.2 参数化查询
- 28.13 ODBC 多线程
- 28.14 大批行数据获取 (bulkrow fetch)
- 28.15 使用不带绑定的记录集

第二十九章 Microsoft Data Access Objects 数据库管理

- 29.1 DAO、COM 和 Microsoft Jet 数据库引擎
- 29.2 DAO 和 VBA
- 29.3 DAO 和 MFC
- 29.4 利用 DAO 可以打开什么样的数据库
- 29.5 按 ODBC 模式使用 DAO 快照和动态集
- 29.6 DAO 表类型记录集
- 29.7 DAO QueryDef 和 TableDef
- 29.8 DAO 多线程情况
- 29.9 在滚动窗口中显示数据库的行
 - 29.9.1 滚动显示方法
 - 29.9.2 一个行 - 视图类
 - 29.9.3 基类和派生类的分工处理
 - 29.9.4 CRowView 的纯虚成员函数
 - 29.9.5 其他的 CRowView 函数
- 29.10 动态记录集的编程
- 29.11 EX29A 示例程序
 - 29.11.1 CEx29aApp
 - 29.11.2 CMainFrame 和 CChildFrame
 - 29.11.3 CEx29aDoc
 - 29.11.4 CEx29aView
 - 29.11.5 CTableSelect

- 29.11.6CIsamSelect
- 29.12EX29A资源文件
- 29.13运行EX29A程序
- 第六部分 Internet 程序设计
- 第三十章 TCP/IP、Winsock和WinInet
- 30.1既针对COM也可以不需要COM
- 30.2Internet基础
- 30.2.1网络协议 层
- 30.2.2Internet协议
- 30.2.3用户报文协议
- 30.2.4IP地址格式 网络字节顺序
- 30.2.5传输控制协议
- 30.2.6域名系统
- 30.2.7HTTP基础
- 30.2.8FTP基础
- 30.2.9Internet和Intranet
- 30.3建立自己的价值 \$ 99的Intranet
- 30.3.1NT文件系统和文件分配表
- 30.3.2网络硬件
- 30.3.3对Windows进行网络配置
- 30.3.4Intranet的主机名 HOSTS文件
- 30.3.5测试Intranet Ping程序
- 30.3.6只有一台计算机的Intranet TCP/IP环形地址
- 30.4Winsock
- 30.4.1同步和异步Winsock编程
- 30.4.2MFCWinsock类
- 30.4.3BlockingSocket类
- 30.4.4一个简化了的Http服务器程序
- 30.4.5一个简化了的HTTP客户端程序
- 30.5用CHttpBlockingSocket创建一个Web服务器程序
- 30.5.1EX30A服务器限制
- 30.5.2Ex30A服务器结构
- 30.5.3使用Win32TransmitFile函数
- 30.5.4编译和测试EX30A
- 30.5.5使用Telnet
- 30.6用CHttpBlockingSocket创建一个Web客户端程序
- 30.6.1EX30AWinsock客户
- 30.6.2EX30A对代理服务器的支持
- 30.6.3测试EX30AWinsock客户
- 30.7WinInet
- 30.7.1WinInet相对于Winsock的优势
- 30.7.2MFCWinInet类
- 30.7.3Internet会话状态回调
- 30.7.4一个简化了的WinInet客户程序
- 30.8用MFCWinInet类创建一个Web客户
- 30.8.1Ex30AWinInetClient # 1 使用CHttpConnection
- 30.8.2测试WinInetClient # 1

- 30.8.3EX3 OAWinInetClient #2 使用OpenURL
- 30.8.4测试WinInetClient #2
- 30.9异步Moniker文件
 - 30.9.1Moniker
 - 30.9.2MFCCAsyncMonikerFile类
 - 30.9.3在程序中使用CAsyncMonikerFile类
 - 30.9.4 异步moniker文件和WinInet编程
- 第三十一章 MicrosoftInternetInformationServer编程
 - 31.1IIs的替代产品
 - 31.2MicrosoftIIS
 - 31.2.1安装和控制IIS
 - 31.2.2运行InternetServiceManager
 - 31.2.3IIS的安全性
 - 31.2.4IIS目录
 - 31.2.5IIS日志记录
 - 31.2.6测试IIS
 - 31.3ISAPI服务器扩展
 - 31.3.1通用网关接口和ISAPI
 - 31.3.2一个简单的ISSAPI服务器扩展GET请求
 - 31.3.3HTML窗体 Get和POST
 - 31.3.4编写ISAPI服务器扩展DLL
 - 31.3.5MFCISAPI服务器扩展类
 - 31.4一个实际的ISAPI服务器扩展 ex31a.dll
 - 31.4.1第一步 获取定单
 - 31.4.2第二步 处理确认窗体
 - 31.4.3编译并测试ex31a.dll
 - 31.4.4调试EX31ADLL
 - 31.5ISAPI数据库访问
 - 31.6使用HTTPCookie来连接事务
 - 31.6.1Cookie如何工作
 - 31.6.2ISAPI服务器扩展如何处理Cookie
 - 31.6.3使用cookie的一些问题
 - 31.7WWW鉴定
 - 31.7.1基本鉴定
 - 31.7.2WindowsNT问讯/应答鉴定
 - 31.7.3安全套接字层 (SSL)
 - 31.8ISAPI过滤器
 - 31.8.1编写ISAPI过滤器DLL
 - 31.8.2MFCISAPI过滤器类
 - 31.9ISAPI过滤器示例 ex31b.dll、ex31c.exe
 - 31.9.1选择通知
 - 31.9.2把事务数据发送到显示程序
 - 31.9.3显示程序
 - 31.9.4 编译并测试EX31BISAPI过滤器
- 第三十二章 ActiveX文档服务器和Internet
 - 32.1ActiveX文档理论
 - 32.1.1ActiveX文档服务器和OLE嵌入服务器

- 32.1.2从IE3中运行ActiveX文档服务器
- 32.1.3ActiveX文档服务器和ActiveX控件
- 32.1.4ActiveX文档服务器和包容器的OLE接口
- 32.1.5MFC对ActiveX文档服务器的支持
- 32.2ActiveX文档服务器示例程序EX32A
 - 32.2.1EX32A第一阶段 简单的服务器
 - 32.2.2调试ActiveX文档服务器
 - 32.2.3Ex32A第二阶段 加入WinInet调用
- 32.3ActiveX文档服务器示例程序EX32B
 - 32.3.1在程序控制下产生POST请求
 - 32.3.2EX32B视图类
 - 32.3.3编译并测试EX32B
 - 32.3.4ActiveX文档服务器和VBScript
- 32.4进一步使用ActiveX文档服务器
- 附录A Microsoft基本类 (MFC) 库中的消息映射函数
- 附录B MFC库运行时类识别型和动态对象创建

版权说明

本站所提供下载的PDF图书仅提供预览和简介, 请支持正版图书。

更多资源请访问:<http://www.tushu007.com>