

# 程序员指南丛书

# Visual C++ .NET 高级编程

- Visual C++ .NET 是针对最新的 .NET 思想建立的、以 C++ 语言为基础的可视化集成开发工具,对 .NET 编程提供了全面的支持。
- 使用 Visual C++ .NET 既可以进行传统的 Windows 编程、COM/COM+ 和 ATL 组件编程,也可以进行 .NET 组件和应用程序编程。
- 本书详细介绍了使用 Visual C++ .NET 进行各种编程的基本方法,所附光盘包括了开发实例的所有程序源码。

齐舒创作室 编著  
毅 鸣 审校



清华大学出版社  
<http://www.tup.com.cn>



本书含光盘

# Visual C ++.NET 高级编程

齐舒创作室 编著  
毅 鸣 审校

清华大学出版社

**(京)新登字 158 号**

### **内 容 简 介**

Visual C++.NET 是 Microsoft 公司针对最新的 .NET 思想建立的、以 C++ 语言为基础的可视化集成开发工具,该工具完全集成到 Visual Studio.NET 中。本书循序渐进地介绍了使用 Visual C++.NET 进行各种编程的基本方法,并通过具体的实例使读者可以更好地理解所学内容。全书分为 4 篇共 14 章,内容包括 MFC 高级编程、组件编程、数据库编程和网络编程。随书光盘中包括了开发实例的所有程序源码。

本书旨在使得广大程序开发人员尤其是 C++ 编程人员可以快速了解和掌握 Visual C++.NET 的功能和操作方法,适用于有一定计算机编程经验的软件开发人员。

**版权所有,翻印必究。**

**本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。**

**书 名:** Visual C++.NET 高级编程  
**作 者:** 齐舒创作室 编著 毅鸣 审校  
**出 版 者:** 清华大学出版社(北京清华大学学研大厦,邮编:100084)  
<http://www.tup.tsinghua.edu.cn>  
**责任编辑:** 胡先福  
**印 刷 者:** 北京市清华园胶印厂  
**发 行 者:** 新华书店总店北京发行所  
**开 本:** 787×1092 1/16 **印张:** 32.5 **字数:** 812 千字  
**版 次:** 2002 年 11 月第 1 版 2002 年 11 月第 1 次印刷  
**书 号:** ISBN 7-900643-77-X  
**印 数:** 0001 ~ 4000  
**定 价:** 55.00 元

# 前 言

Visual C ++.NET 是 Microsoft 公司针对最新的 .NET 思想建立的、C ++ 语言为基础的可视化集成开发工具,该工具完全集成到 Visual Studio .NET 中。作为新一代产品。Visual C ++.NET 不仅继承了 Visual C ++ 的所有功能,还在增强 ATL 服务器编程功能的基础上,通过使用扩展受控 C ++ 对 .NET 编程提供了全面的支持。

使用 Visual C ++.NET 既可以进行传统的 Windows 编程、COM/COM + 和 ATL 组件编程,也可以使用扩展的受控 C ++ 进行 .NET 组件和应用程序编程。对于这样一个功能强大的全新开发工具,如何才能更好地掌握呢? 本书就是为了解决这个问题而编写的。书中采用了循序渐进的方法,由简单到复杂,由初级到高级,详细介绍了使用 Visual C ++.NET 进行各种编程的基本方法,并通过具体的实例使读者可以更好地理解所学内容。本书旨在使得广大程序开发人员尤其是 C ++ 编程人员可以快速了解和掌握 Visual C ++.NET 的功能和操作方法。

本书分为 4 篇共 14 章。

第 1 篇 MFC 高级编程:本篇包括前 3 章内容,主要介绍如何使用 Visual C ++.NET 开发基于文档视图、动态链接库和多线程的程序。第 1 章介绍了应用程序框架、文档模板和文档视图之间的关系,在这个基础上,详细讲述了怎样在应用程序中传递消息和怎样实现切分视图。第 2 章首先介绍有关动态链接库方面的基本理论,然后讲解如何使用 MFC 创建扩展 DLL 和正规 DLL。第 3 章则主要介绍进程和线程方面的有关知识,在理解进程的基础上详细介绍了多线程编程。

第 2 篇组件编程:这一篇包括第 4 ~ 7 章的内容。在第 4 章中介绍了 COM 的基本概念和创建 COM 组件的方法。第 5 章主要介绍 ATL 库和 ATL 控件的创建方法。第 6 章详细讨论了关于 ATL Server 和 ATL Web 服务的编程方法,以及 SOAP、XML 和 XSL 的基本概念。第 7 章主要介绍 .NET 编程的一些重要理论,以及如何使用 C ++ 托管扩展语言和 C # 语言来编写可操控组件及应用程序。

第 3 篇数据库编程:这一篇包括第 8 ~ 13 章的内容。第 8 章介绍了数据库的基本知识。第 9 章讲述了怎样使用 ODBC 开发数据库。第 10 章介绍了怎样使用 DAO 开发数据库。第 11 章首先介绍 SQL 语言,接着介绍怎样使用服务器资源管理器。第 12 和 13 章介绍有关 ADO 的知识,并详细讨论了怎样使用 ADO.NET 来开发数据库。

第 4 篇网络编程:即第 14 章,在这一章里介绍了怎样使用 Visual C ++.NET 开发网络应用程序。首先详细介绍了 Windows 的套接字和 Windows API 函数,然后讨论怎样使用 Winsock 的 MFC 类编写网络程序,最后介绍怎样使用 WinInet 进行网络开发。

本书所附光盘包括了开发实例的所有程序源码,这些程序源码都是在 Visual C ++.NET 上编译通过的。要运行本光盘上的实例,读者最好事先安装 Visual Studio .NET,以免出现运行不了的问题。然而对于 DLL、COM、ATL 和 ActiveX 的例程,读者必须将其先复制到硬盘中,再使用 Visual C ++.NET 进行编译,以便对这些组件程序进行注册,然后通过客户程序来

调用。为了不出现任何问题,最好的方法是读者自己创建一个新的工程,再将光盘中提供的源码复制到相应的文件中。

本书由齐舒创作室策划并编写,其中李剑、于亮和曲学军负责全书的设计、统稿和修改,并负责光盘内容的制作和例子测试,参与本书编写和指导工作的还有刘勇、孙仁、于勇、于伟、郭志鹏、李宏军、张世扬、赵天亮、吴齐、关汗羽、李治、李海涛、苏震、卢雪阳、张士华、陈辉、郑明峰、姚正思、沈毅明、汪宝川、高朴真、王立、顾仁、刘航、方东、许胜利、冯卫国、黄丽云、张梦天、程浩、高雪松和蒋才国等。由于作者水平和能力有限,本书编写网络程序,最后介绍了如何使用 WinInet 进行网络开发。

由于时间短促,再加之作者的能力和水平有限,本书的错误以及疏漏之处在所难免,望广大读者批评指正。

编著者  
2002年8月

# 目 录

## 第 1 篇 MFC 高级编程

<b>第 1 章 应用程序框架</b> .....	1
1.1 应用程序、框架、文档和视图 .....	1
1.1.1 基本介绍 .....	1
1.1.2 文档模板 .....	3
1.1.3 文档和视图 .....	4
1.1.4 最简单的文档/视图应用 .....	7
1.2 消息处理 .....	8
1.2.1 发送和寄送消息 .....	8
1.2.2 消息类型 .....	9
1.2.3 使用 MFC 处理接收到的消息 .....	12
1.2.4 创建自定义的消息 .....	13
1.2.5 消息的重定位 .....	17
1.3 切分窗口 .....	24
1.3.1 切分窗口概述 .....	24
1.3.2 动态切分窗口 .....	25
1.3.3 静态切分窗口 .....	26
1.3.4 滚动视图和 CScrollView 类 .....	27
1.4 单文档应用程序切分窗口编程举例——SDISplitter .....	28
1.4.1 创建工程和资源 .....	28
1.4.2 切分窗口 .....	31
1.4.3 添加功能代码 .....	34
1.4.4 编译并生成可执行文件 .....	35
1.5 多文档界面编程举例——MDISplitter .....	35
1.5.1 创建工程 .....	36
1.5.2 修改文档和视图类 .....	36
1.5.3 完成视图类 CEnterView .....	38
1.5.4 完成视图类 CShowView .....	41
1.5.5 完成 CSplitFrame 类 .....	43
1.5.6 添加模板资源 .....	44
1.5.7 编译并生成可执行文件 .....	45
<b>第 2 章 动态链接库</b> .....	46
2.1 动态链接库的基本理论 .....	46

2.1.1	导入函数和导出函数 .....	47
2.1.2	隐式连接和显式连接 .....	48
2.1.3	选择连接方式 .....	49
2.1.4	DLL 入口点——DllMain .....	50
2.1.5	动态链接库的定位 .....	50
2.2	使用 MFC 编写 DLL .....	51
2.2.1	MFC 扩展 DLL .....	51
2.2.2	MFC 扩展 DLL 的资源搜索顺序 .....	52
2.2.3	扩展 DLL 实例——ExtensionDll .....	52
2.2.4	测试扩展 DLL .....	55
2.2.5	MFC 规则 DLL .....	59
2.2.6	MFC 规则 DLL 的资源搜索顺序 .....	59
2.2.7	规则 DLL 实例——RegularDll .....	59
2.2.8	测试规则 DLL .....	61
<b>第 3 章</b>	<b>进程和线程 .....</b>	<b>62</b>
3.1	进程 .....	62
3.1.1	基本理论 .....	62
3.1.2	进程的创建 .....	63
3.1.3	进程的终止 .....	68
3.1.4	子进程 .....	70
3.2	多线程编程 .....	72
3.2.1	线程的基础知识 .....	72
3.2.2	线程的优先级 .....	73
3.2.3	创建辅助线程和用户界面线程 .....	75
3.2.4	一个简单的实例 .....	78
3.2.5	线程的挂起和唤醒 .....	83
3.2.6	线程的休眠 .....	83
3.2.7	线程的同步 .....	84
3.2.8	事件对象 .....	84
3.2.9	使用临界段 .....	89
3.2.10	使用互斥量 .....	91
3.2.11	使用信号量 .....	94
3.2.12	一个复杂的线程例子 .....	96
<b>第 2 篇 组件编程</b>		
<b>第 4 章</b>	<b>COM 编程 .....</b>	<b>111</b>
4.1	组件对象模型 COM 简介 .....	111
4.1.1	COM 所解决的问题 .....	111
4.1.2	COM 本质 .....	111

4.1.3	COM 对象类型	112
4.1.4	COM 的利与弊	113
4.2	COM 对象和接口	114
4.2.1	COM 对象	114
4.2.2	全球惟一标识符 GUID	114
4.2.3	COM 对象的特点	116
4.2.4	COM 接口	117
4.2.5	接口描述语言 IDL	120
4.2.6	接口的特点	121
4.3	IUnknown 接口	122
4.3.1	引用计数	123
4.3.2	接口查询	124
4.4	COM 的实现	126
4.4.1	COM 组件的实现规则	127
4.4.2	进程内组件和进程外组件	127
4.4.3	类厂	129
4.4.4	建立一个模拟 COM 组件服务器程序	131
4.4.5	COM 库	140
4.5	COM 客户	142
4.5.1	COM 对象创建函数	142
4.5.2	运行时的 COM 对象注册	144
4.5.3	COM 客户程序调用进程内组件	145
4.5.4	COM 客户程序调用进程外组件	146
4.6	MFC 对 COM 的支持	147
4.6.1	MFC 接口宏	148
4.6.2	MFC 对自动化的支持	152
4.6.3	MFC 对 ActiveX 控件的支持	152
4.6.4	MFC 对复合文档的支持	154
4.7	基于 MFC 的 COM 编程举例	154
4.7.1	使用 MFC 创建服务程序	155
4.7.2	创建 COM 客户程序	160
4.8	COM 的高级特性	162
4.8.1	包容和聚合	162
4.8.2	实现包容和聚合	164
4.8.3	列集	166
4.8.4	多线程特性	168
4.9	COM 扩展技术——COM+	172
4.9.1	COM 与 Windows DNA	172
4.9.2	组件服务配置	173



4.9.3	事务处理	175
4.9.4	排队组件	176
4.9.5	动态负荷平衡	177
4.9.6	对象池	178
<b>第 5 章</b>	<b>ATL 控件编程</b>	<b>179</b>
5.1	Active 技术概述	179
5.1.1	ActiveX 的用途	179
5.1.2	对象链接和嵌入	179
5.1.3	容器和服务	179
5.1.4	ActiveX 自动化编程	180
5.1.5	ActiveX 控件	180
5.2	ActiveX 控件编程	181
5.2.1	ActiveX 控件的事件	181
5.2.2	ActiveX 控件的方法	184
5.2.3	ActiveX 控件的属性	185
5.2.4	ActiveX 控件的属性页	186
5.3	ATL 基础	187
5.3.1	使用模板库	189
5.3.2	ATL 适用范围	189
5.3.3	何时使用 ATL	189
5.4	ATL 窗口类	189
5.4.1	使用窗口	190
5.4.2	实现一个窗口	190
5.4.3	实现一个对话框	192
5.4.4	使用容器窗口	193
5.5	连接点	194
5.5.1	连接点类	194
5.5.2	给对象增加一个连接点	194
5.5.3	连接点的例子	195
5.6	枚举器	195
5.7	代理	196
5.7.1	ATL 代理	196
5.7.2	生成连接点或快捷指针的代理	196
5.8	创建一个简单的 ATL 控件	196
5.8.1	创建工程	197
5.8.2	添加控件	198
5.8.3	增加属性	203
5.8.4	修改绘图代码	205

---

<b>第 6 章 ATL Serer 编程</b> .....	208
6.1 ATL Server 简介 .....	208
6.1.1 ATL Server 宏 .....	208
6.1.2 ATL Server 属性 .....	209
6.1.3 ATL Server 函数 .....	214
6.1.4 服务器响应文件 .....	215
6.1.5 ATL Server 程序结构 .....	215
6.1.6 创建 ATL Server 工程 .....	217
6.2 SOAP 规范 .....	222
6.2.1 设计目标 .....	222
6.2.2 符号约定 .....	223
6.2.3 SOAP 消息示例 .....	223
6.2.4 SOAP 消息交换模型 .....	224
6.3 XML 简介 .....	227
6.3.1 什么是 XML .....	227
6.3.2 XML 元素 .....	229
6.3.3 XML 元素的内容和属性 .....	230
6.3.4 一个简单的 XML 例子 .....	231
6.4 XSL 简介 .....	235
6.4.1 CSS:HTML 的样式表 .....	235
6.4.2 XSL:XML 的样式表 .....	235
6.4.3 XSLT:XSL 转换 .....	236
6.4.4 XSL 如何工作 .....	236
6.5 XSL 转换 .....	236
6.5.1 从 XML 文档开始 .....	236
6.5.2 创建一个 XSL 样式表 .....	237
6.5.3 将样式表连接到 XML 文档 .....	237
6.5.4 XSL 模板 .....	238
6.5.5 客户端 XSL .....	240
6.5.6 服务器端 XSL .....	241
6.6 ATL Web 服务编程简介 .....	242
6.6.1 Web 服务概述 .....	242
6.6.2 ATL Web 服务的基础结构 .....	243
6.6.3 服务描述 .....	243
6.6.4 创建 ATL Web 服务提供者程序 .....	243
6.6.5 创建 ATL Web 服务客户程序 .....	247
<b>第 7 章 .NET 框架编程</b> .....	252
7.1 初识 .NET .....	252

7.2	.NET 结构简介	253
7.2.1	虚拟对象系统	253
7.2.2	元数据	254
7.2.3	公用语言规范	254
7.2.4	虚拟执行系统	254
7.2.5	.NET 框架结构	255
7.3	理解 .NET 框架编程	255
7.3.1	公用语言运行时环境	255
7.3.2	深入理解公用语言规范	256
7.3.3	受控执行	257
7.3.4	程序集	257
7.3.5	元数据和自我描述组件	258
7.3.6	.NET 框架的基本单元——类型	259
7.4	托管扩展编程方法	266
7.5	何时使用 C++ 托管扩展语言	266
7.6	公用类型系统	267
7.7	C++ 托管扩展语言简介	267
7.7.1	受控类型	268
7.7.2	受控数组	269
7.7.3	C++ 托管扩展中的字符串	270
7.7.4	C++ 托管扩展中的代理	270
7.7.5	受控对象的属性	271
7.8	使用 C++ 托管扩展语言进行编程	272
7.8.1	在现有工程中添加对 C++ 托管扩展的支持	272
7.8.2	全局属性和扩展受控 C++	273
7.8.3	受控类型和 MFC	274
7.9	C# 简介	275
7.10	C# 语言的特点	275
7.10.1	简洁的语法	276
7.10.2	精心的面向对象设计	276
7.10.3	与 Web 的紧密结合	277
7.10.4	完全的安全性与错误处理	277
7.10.5	版本处理技术	277
7.10.6	灵活性和兼容性	278
7.11	创建托管的 C++ 应用程序	278
7.11.1	创建托管的 C++ 应用程序项目	278
7.11.2	实现主框架窗口	280
7.11.3	实现 Stroke 类	285
7.11.4	实现文档和视图	286

7.11.5	添加设置线宽对话框 .....	289
7.11.6	编译并运行应用程序 .....	291
7.12	使用 C# 编写 .NET 应用 .....	292
7.12.1	使用 C# 语言的原因 .....	293
7.12.2	HelloWorld 的 C# 版 .....	293
7.12.3	C# 程序的常规结构 .....	295
7.12.4	创建和使用 C# DLL .....	296

### 第 3 篇 数据库编程

第 8 章	用 Visual C++ .NET 开发数据库应用程序 .....	299
8.1	MFC 支持的数据库访问类型 .....	299
8.1.1	ODBC .....	300
8.1.2	DAO .....	300
8.1.3	比较 ODBC 与 DAO .....	301
8.1.4	OLE DB .....	302
8.2	开发数据库应用程序的方法 .....	303
8.2.1	通过 API 访问数据库 .....	304
8.2.2	通过数据对象接口访问数据库 .....	304
8.3	构造数据库 .....	305
8.3.1	数据库概述 .....	305
8.3.2	创建数据库 .....	307
第 9 章	用 ODBC 开发数据库应用程序 .....	312
9.1	ODBC 起源 .....	312
9.2	ODBC 的工作机制 .....	313
9.2.1	ODBC 标准 .....	313
9.2.2	ODBC 驱动程序管理器 .....	314
9.2.3	ODBC 驱动程序 .....	314
9.2.4	ODBC 数据源管理器 .....	314
9.2.5	ODBC 一致层 .....	315
9.2.6	ODBC 的两种版本 .....	315
9.3	MFC 封装的 ODBC 数据库类 .....	316
9.4	使用 CRecordset 类 .....	317
9.4.1	连接到数据库 .....	317
9.4.2	打开和关闭记录集 .....	317
9.4.3	浏览记录集 .....	318
9.5	添加、更新和删除记录 .....	319
9.5.1	确定记录集是否可更新 .....	319
9.5.2	将记录添加到记录集 .....	320
9.5.3	编辑记录集中的记录 .....	321

9.5.4 从记录集中删除记录 .....	321
9.6 利用书签和绝对位置定位记录 .....	322
9.7 排序与过滤 .....	323
9.7.1 对记录排序 .....	323
9.7.2 筛选记录 .....	323
9.8 创建 ODBC 应用程序 .....	325
9.8.1 注册数据库 .....	326
9.8.2 利用应用程序向导创建应用程序外壳 .....	328
9.8.3 显示数据库的记录 .....	332
9.8.4 添加、删除和修改记录 .....	336
9.8.5 定位到指定的记录 .....	340
9.8.6 排序与过滤 .....	346
<b>第 10 章 用 DAO 开发数据库应用程序 .....</b>	<b>352</b>
10.1 DAO、COM 和 Microsoft Jet 数据库引擎 .....	352
10.2 MFC 中的 DAO 类 .....	352
10.2.1 MFC 如何封装 DAO .....	353
10.2.2 DAO 对象和 MFC 类 .....	353
10.3 利用 DAO 可以打开的数据库 .....	354
10.4 DAO 记录集类型 .....	354
10.4.1 按 ODBC 模式使用 DAO .....	355
10.4.2 DAO 表类型记录集 .....	355
10.5 DAO 的其他特性 .....	356
10.6 使用 MFC DAO 类创建和运行查询 .....	356
10.7 使用 DAO .....	358
10.7.1 打开记录集 .....	358
10.7.2 动态绑定记录 .....	360
10.8 利用 DAO 构造数据库应用程序 .....	362
10.8.1 创建一个简单的 DAO 数据库访问应用程序 .....	363
10.8.2 进一步完善应用程序的功能 .....	372
<b>第 11 章 使用服务器资源管理器 .....</b>	<b>380</b>
11.1 什么是 SQL .....	380
11.2 SQL 语法基础 .....	381
11.2.1 数据控制语言语句 .....	381
11.2.2 数据定义语言语句 .....	381
11.2.3 数据操作语言语句 .....	383
11.3 在 C++ 中利用 SQL 操作数据库 .....	386
11.4 创建 Procedure 应用程序 .....	387
11.4.1 创建应用程序外壳 .....	387

---

11.4.2	数据连接 .....	389
11.4.3	服务器资源管理器 .....	392
11.4.4	查询设计器 .....	393
11.4.5	存储过程 .....	395
11.4.6	编写存储过程 .....	398
11.4.7	利用 C++ 代码调用存储过程 .....	400
11.5	利用服务器资源管理器管理数据库 .....	405
11.5.1	设计数据库 .....	405
11.5.2	使用数据库关系图 .....	408
11.6	Microsoft 事务处理服务器 .....	410
<b>第 12 章</b>	<b>用 ADO 开发数据库应用程序 .....</b>	<b>412</b>
12.1	ADO 概述 .....	412
12.1.1	OLE DB 与 ADO .....	412
12.1.2	ADO 和 RDO、DAO 的比较 .....	414
12.1.3	ADO 对象 .....	414
12.2	ADO.NET 简介 .....	416
12.2.1	为何使用 ADO.NET .....	417
12.2.2	ADO.NET 的体系结构 .....	418
12.2.3	ADO.NET 和 ADO .....	419
12.2.4	ADO、ADO.NET 与 OLE DB .....	421
12.3	如何使用 ADO DLL .....	424
12.3.1	引入 ADO DLL .....	424
12.3.2	建立到数据库的连接 .....	425
12.3.3	执行命令并获取数据 .....	425
12.3.4	浏览记录集 .....	427
12.3.5	访问字段值 .....	427
12.3.6	ADO 中的宏 .....	427
12.3.7	更新记录 .....	428
12.3.8	添加和删除记录 .....	428
12.3.9	关闭 Recordset 对象和 Connection 对象 .....	429
12.4	使用 ADO 创建数据库应用程序 .....	429
12.4.1	创建应用程序外壳 .....	430
12.4.2	创建定制的记录类 .....	431
12.4.3	连接和获取数据 .....	434
12.4.4	显示记录 .....	436
12.4.5	保存修改结果 .....	438
12.4.6	浏览记录集 .....	439
12.4.7	增加新记录 .....	443

12.4.8	删除当前记录 .....	445
12.4.9	进一步完善应用程序功能 .....	445
<b>第 13 章</b>	<b>使用 ADO.NET 开发数据库应用程序 .....</b>	<b>448</b>
13.1	ADO.NET 概述 .....	448
13.1.1	ADO.NET 对象模型 .....	448
13.1.2	记录集和数据集 .....	450
13.1.3	使用命名空间 .....	451
13.2	连接数据源 .....	452
13.3	用 DataSets 访问数据 .....	453
13.3.1	用 DataSets 读数据 .....	453
13.3.2	在 DataSet 中存储多个表 .....	456
13.3.3	DataTable 对象 .....	458
13.3.4	使用 DataView .....	460
13.3.5	在 DataGrid 控件中显示数据 .....	460
13.3.6	使用模板 .....	463
13.3.7	使用 Repeater 控件 .....	464
13.4	使用存储过程 .....	465
13.4.1	调用存储过程 .....	465
13.4.2	传递参数 .....	467
13.5	用 DataReaders 访问数据 .....	469
13.5.1	创建 DataReader .....	469
13.5.2	从 DataReader 读数据 .....	470
13.5.3	使用 DataSets 还是 DataReaders .....	472

## 第 4 篇 网络编程

<b>第 14 章</b>	<b>Internet 编程 .....</b>	<b>474</b>
14.1	网络基础 .....	474
14.1.1	OSI 网络模型 .....	474
14.1.2	OSI 网络模型概述 .....	475
14.2	Winsock 编程 .....	475
14.2.1	Windows 套接字 .....	476
14.2.2	Winsock 的版本和平台 .....	476
14.2.3	Winsock 函数 .....	479
14.2.4	MFC 的 Winsock 类 .....	489
14.3	WinInet 编程 .....	499
14.3.1	WinInet 概述 .....	499
14.3.2	使用 MFC WinInet 类编写 Internet 客户端应用程序 .....	499

# 第 1 篇 MFC 高级编程

## 第 1 章 应用程序框架

### 1.1 应用程序、框架、文档和视图

大多数 MFC 类是从三个基类(Basic Class)派生的: CObject、CCommandTarget 和 CWnd。CCommandTarget 是从 CObject 类派生来的,而 CWnd 是从 CCommandTarget 派生来的。从 CObject 派生出来的类具有在运行时获得对象名字的能力,从 CCommandTarget 派生的类能够处理命令消息,从 CWnd 派生的类可以控制应用程序的窗口。

#### 1.1.1 基本介绍

创建一个标准的 MFC 应用程序时,应用程序从四个基类派生:

CWinApp 是应用程序的“应用程序类”,负责初始化和运行应用程序。

CFrameWnd 是应用程序的“框架类”,负责显示和搜索用户命令。

CDocument 是应用程序的“文档类”,负责装载和维护文档。文档可以是文档到网络设备设置的任何内容。

CView 是应用程序的“视图类”,负责为文档提供一个或多个视图。

#### 1. 应用程序类(CWinApp)

应用程序类 CWinApp 是应用程序开始后创建的第一个对象,并且是在结束前最后一个执行的对象。程序启动时,应用程序类负责创建程序的其余对象。

应用程序使用文档模板来管理与视图和框架窗口相关的文档创建过程。MFC 提供了抽象基类 CDocTemplate 来实现文档模板的功能,应用程序不能直接使用,但可以把文档模板定义为 CDocTemplate 的派生类。在应用程序中,必须为每种类型的文档都建立一个文档模板。

对于一个单文档(SDI)的应用程序,应用程序类创建一个文档模板,然后用这个模板打开一个空文档。

对于一个多文档(MDI)的应用程序,在主框架类中,应用程序类创建多个文档模板然后用模板打开一个文档。

对于一个对话框应用程序,应用程序类用 CDialog 类创建一个对话框。

CWinApp 类自身是从 CWinThread 类派生的。CWinThread 类封装了系统中用来创建和维护应用程序线程的 Windows API 函数。实际上,可以通过创建 CWinThread 类的另一个实例实现应用程序多任务。在应用程序中,CWinApp 类代表了主要的执行线程。



### 2. 框架类 (CFrameWnd)

框架类 CFrameWnd 是应用程序运行时创建的第二个对象,负责显示和监督用户对应用程序其余部分的命令。

对于一个单文档应用程序,框架类是从 CFrameWnd 派生的;而对于一个多文档应用程序,框架类是从 CMDIFrameWnd 派生的。在多文档应用程序中,每个打开的文档有一个子框架类,每个子框架类是从 CMDIChildWnd 派生的。

对于一个对话框应用程序来说,没有框架类。在前面提过,一个对话框应用程序是由一个应用程序类和一个对话框类组成的。

### 3. 文档类 (CDocument)

通常,文档类 CDocument 是应用程序打开一个新文档或一个已存在的文档时创建的对象。文档类负责将一个文档赋给它的成员变量,并允许视图类编辑这些成员变量。一个文档包括从图形文件到可编程控制的任何内容。

### 4. 视图类 (CView)

在创建文档类的一个实例之后将创建视图类 CView 的一个实例。视图类负责描述文档类的内容,它也可以允许用户编辑文档。

窗口分区类 CSplitterWnd 允许文档有多视图,这些视图可以由相同视图类的一些实例创建,也可以由完全不同的视图类实例创建。

应用程序从 CView、CScrollView、CFormView 或 CEditView 等类派生视图类(也可直接用 CEditView 类),用于显示文档数据。

常用的应用程序是单文档和多文档应用程序。下面详细介绍一下这两个应用程序类型。

#### • 单文档应用程序

单文档应用程序包括以下类:从 CWinApp 派生的应用程序类,从 CFrameWnd 派生的框架类,从 CDocument 派生的文档类,以及每个文档类的一个或多个视图类,这些视图类是从一些派生于 CView 的视图类派生的,见图 1-1。

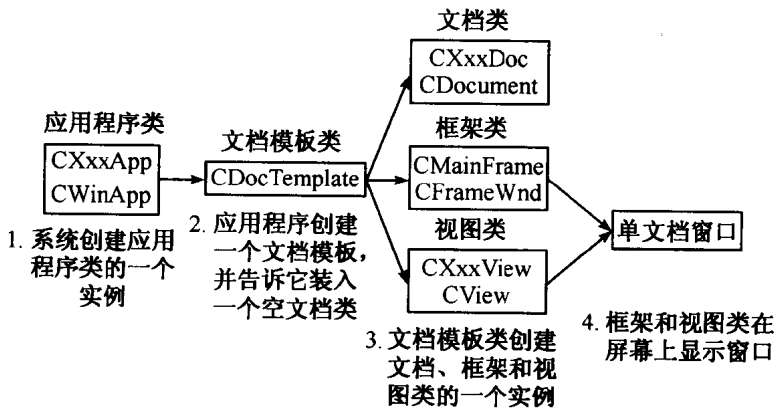


图 1-1 一个单文档应用程序由应用程序、框架、文档和视图类创建