

魅力·实践·发现

Visual C++.NET(2008)

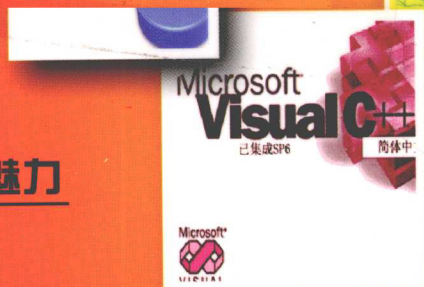
典型案例

郑阿奇 主编

展现精英高手发现之旅

站在流行平台开发实践

介绍流行软件神奇魅力



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

魅力·实践·发现

Visual C++.NET (2008)

典型案例

郑阿奇 主 编

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书以 Visual C++.NET (2008) 几个典型的应用实例为引导, 比较系统地介绍 Visual C++.NET (2008) 应用开发方法, 对学习 Visual C++.NET 开发实践具有明显的优势。其基本方法是把繁多和复杂的内容分散开来, 通过典型应用理解原理和方法。应用实例首先是实现任务目标, 然后是实现功能及设计思路, 再介绍需要的主要知识点, 在此基础上介绍程序实现过程, 最后是实例小结及实例扩展。实例扩展就是读者在理解的基础上自己完成。

本书可作为 Visual C++.NET 学习和应用开发的参考, 也可作为大学本科、高职高专 Visual C++.NET 课程、实习教材。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目 (CIP) 数据

Visual C++.NET (2008) 典型案例 / 郑阿奇主编. —北京: 电子工业出版社, 2011.1
(魅力·实践·发现)
ISBN 978-7-121-12170-8

I. ①V… II. ①郑… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2010) 第 213696 号

责任编辑: 郝黎明 特约编辑: 钟永刚

印 刷: 北京丰源印刷厂

装 订: 三河市鹏成印业有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 26 字数: 666 千字

印 次: 2011 年 1 月第 1 次印刷

印 数: 3000 册 定价: 45.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

Microsoft 公司最新 Visual C++.NET (2008) 是目前系统开发的最佳平台, 它支持两种不同但又紧密相关的 C++ 语言。第一种是原来 ISO/ANSI 标准 C++, Visual C++6.0 是其流行的开发平台之一。第二种是由 Microsoft 公司开发的 C++/CLI 的新版 ECMA 标准 C++, 专门为 .NET Framework 开发。为了让读者更快掌握 Visual C++.NET (2008) 开发应用的方法, 本书采用多个案例进行介绍, 这些案例既可方便学习, 又可在应用开发时模仿。

Visual C++.NET (2008) 典型案例具有如下特点:

(1) 先介绍 Visual C++.NET 应用开发和编程基础, 在此基础上构建案例;

(2) 应用案例既不大, 也不小; 每个应用案例中分布的知识点不太多, 也不少;

(3) 应用案例的顺序安排考虑分布的知识点要循序渐进, 逐步深入。能够覆盖 Visual C++.NET (2008) 的主要内容、主要方面。

(4) 在内容结构组织上, 首先是任务目标, 然后是实现功能及设计思路, 再介绍需要的主要知识点。在此基础上介绍程序实现过程, 最后是实例小结及实例扩展。实例扩展就是读者理解的基础上自己完成。

(5) Visual C++ 典型案例以 C++ 作为基础通过审读, 案例通过试做确认能够完成设计的功能。

(6) 在介绍 Visual C++.NET (2008) MFC 的同时, 兼顾 CLR 模式开发系统。

读者通过对本书的学习和上机操作练习, 能够基本掌握 Visual C++.NET (2008) 开发应用系统的方法。

本书由南京师范大学郑阿奇主编, 参加本套丛书编写的还有郑进、陶卫冬、邓拼搏、严大牛、韩翠青、王海娇、刘博宇、孙德荣、吴明祥、周何骏、徐斌、孙承龙、陈超、毛风伟等。

本书配有教学、学习课件和应用系统的所有源代码文件, 需要者可以在华信教育资源网 <http://www.hxedu.com.cn>, 免费下载。

由于我们的水平有限, 错误在所难免, 敬请广大师生、读者批评指正。

意见建议邮箱: easybooks@163.com

编 者

目 录

第 1 章 熟悉 Visual C++.NET 开发环境	1
1.1 Visual C++.NET 概述	1
1.1.1 了解 .NET 框架	1
1.1.2 托管 C++ 代码	3
1.2 Visual C++.NET 环境	4
1.2.1 窗口元素	4
1.2.2 开发组件	9
1.3 项目与解决方案	10
1.3.1 项目的创建	11
1.3.2 项目的组织	14
1.3.3 项目的属性	16
1.4 资源编辑器	17
1.4.1 添加新资源	18
1.4.2 加速键编辑器	18
1.4.3 对话框编辑器	19
1.4.4 菜单编辑器	19
1.4.5 图形编辑器	20
1.4.6 字符串表编辑器	21
1.4.7 版本编辑器	21
1.4.8 其他资源编辑器	22
1.5 环境设置选项	23
1.5.1 设置常用选项	23
1.5.2 自定义开发环境	24
1.6 调试程序	25
1.6.1 调试窗口	25
1.6.2 调试步骤	27
1.7 本章小结	30
第 2 章 Windows 编程基础	31
2.1 Windows 编程概述	31
2.1.1 简单的 Windows 应用程序	32
2.1.2 常用数据类型	37
2.1.3 消息机制	40

2.1.4	常见的消息类型	42
2.1.5	Win32 API 概述	45
2.2	MFC 编程概述	45
2.2.1	MFC 背景介绍	45
2.2.2	MFC 类库结构	46
2.2.3	MFC 项目类型	51
2.2.4	简单的 MFC 应用程序	53
2.2.5	MFC 中的宏	58
2.3	MFC 新用户界面	61
2.3.1	MFC 功能包概述	61
2.3.2	MFC 新增类	62
2.3.3	Office 视觉样式	67
2.3.4	Visual Studio 视觉样式	72
2.4	本章小结	75
第 3 章	设计“文件管理”程序	76
3.1	实现功能及设计思路	76
3.1.1	功能描述	76
3.1.2	设计思路	78
3.2	主要知识点	79
3.2.1	对话框概述	79
3.2.2	模式与无模式对话框	80
3.2.3	常用控件	83
3.2.4	DDX 和 DDV	88
3.2.5	通用对话框	89
3.3	程序实现过程	91
3.3.1	设计程序主界面	92
3.3.2	创建位图按钮及工具提示	96
3.3.3	更新树形控件中的项	98
3.3.4	更新列表控件中的项	105
3.3.5	响应树形控件的事件	108
3.3.6	响应列表控件的事件	110
3.3.7	选择地址栏中的路径	112
3.3.8	显示隐藏、系统文件	112
3.3.9	打开、重命名文件	113
3.3.10	复制、移动及删除文件	118
3.4	实例小结及扩展	119

3.4.1	实例小结	119
3.4.2	实例扩展	119
第 4 章	设计“通讯簿”程序	121
4.1	实现功能及设计思路	121
4.1.1	功能描述	121
4.1.2	设计思路	125
4.2	主要知识点	125
4.2.1	框架窗口	125
4.2.2	菜单、工具栏和状态栏	128
4.2.3	文档和视图	130
4.2.4	“一档多视”	135
4.2.5	文档序列化	136
4.3	程序实现过程	139
4.3.1	创建应用程序框架	139
4.3.2	定义数据结构	143
4.3.3	新建联系人信息	146
4.3.4	查看联系人信息	153
4.3.5	修改、删除信息	157
4.3.6	查找联系人信息	160
4.3.7	读取、保存数据	163
4.4	实例小结及扩展	165
4.4.1	实例小结	165
4.4.2	实例扩展	165
第 5 章	设计“画图”程序	167
5.1	实现功能及设计思路	167
5.1.1	功能描述	167
5.1.2	设计思路	167
5.2	主要知识点	168
5.2.1	设备环境类	168
5.2.2	坐标映射	170
5.2.3	GDI 对象	173
5.2.4	图形绘制	176
5.2.5	文本输出	180
5.2.6	位图、图标及光标	185
5.3	程序实现过程	187
5.3.1	创建 MyPaint 项目	188

5.3.2	创建 Ribbon 功能区	189
5.3.3	新建绘图类	192
5.3.4	响应绘图命令	198
5.3.5	实现图形绘制	200
5.3.6	实现文本输出	204
5.3.7	修改绘图光标	208
5.4	实例小结及扩展	209
5.4.1	实例小结	209
5.4.2	实例扩展	209
第 6 章	设计“网络聊天”程序	210
6.1	实现功能及设计思路	210
6.1.1	功能描述	210
6.1.2	设计思路	212
6.2	主要知识点	212
6.2.1	Internet 概述	212
6.2.2	WinSock 基础	217
6.2.3	MFC 套接字基础	221
6.2.4	MFC 多线程概述	223
6.3	程序实现过程	224
6.3.1	编辑对话框资源	226
6.3.2	添加套接字类	227
6.3.3	取得本地信息	232
6.3.4	获取用户信息	234
6.3.5	实现网络聊天	237
6.3.6	实现文件传输	239
6.4	实例小结及扩展	245
6.4.1	实例小结	245
6.4.2	实例扩展	245
第 7 章	设计“网络聊天”程序 (DLL 实现)	247
7.1	实现功能及设计思路	247
7.1.1	功能描述	247
7.1.2	设计思路	249
7.2	主要知识点	249
7.2.1	动态链接库概述	249
7.2.2	创建动态链接库	253
7.2.3	访问动态链接库	256

7.3	程序实现过程	258
7.3.1	创建 NetChat 项目	258
7.3.2	创建套接字类 DLL	260
7.3.3	创建文件传输 DLL	263
7.3.4	导入套接字输出类	265
7.3.5	实现文件传输功能	269
7.4	实例小结及扩展	271
7.4.1	实例小结	271
7.4.2	实例扩展	271
第 8 章	设计“图书管理”程序	272
8.1	实现功能及设计思路	272
8.1.1	功能描述	272
8.1.2	设计思路	277
8.2	主要知识点	278
8.2.1	ODBC 概述	278
8.2.2	ADO 概述	286
8.2.3	ADO 接口	288
8.2.4	几种数据类型	290
8.2.5	使用 ADO	292
8.3	程序实现过程	298
8.3.1	创建图书数据库	298
8.3.2	创建应用程序窗口	298
8.3.3	建立数据库连接	301
8.3.4	实现数据查询功能	302
8.3.5	实现借书、还书功能	311
8.3.6	实现图书管理功能	313
8.4	实例小结与扩展	316
8.4.1	实例小结	316
8.4.2	实例扩展	317
第 9 章	设计“图书管理”程序（打印功能）	318
9.1	实现功能及设计思路	318
9.1.1	功能描述	318
9.1.2	设计思路	320
9.2	主要知识点	320
9.2.1	MFC 打印概述	320
9.2.2	MFC 打印过程	321

9.2.3	CPrintInfo 类	322
9.2.4	MFC 打印预览	322
9.3	程序实现过程	323
9.3.1	添加框架/视图类	323
9.3.2	添加窗口元素	324
9.3.3	实现打印功能	326
9.4	实例小结与扩展	334
9.4.1	实例小结	334
9.4.2	实例扩展	334
第 10 章	设计“涂鸦板”程序	335
10.1	实现功能及设计思路	335
10.1.1	功能描述	335
10.1.2	设计思路	337
10.2	主要知识点	337
10.2.1	了解 C++/CLI 语法	337
10.2.2	Windows Forms 编程基础	342
10.2.3	了解 GDI+	349
10.2.4	了解 ADO.NET	353
10.3	程序实现过程	361
10.3.1	准备数据库	362
10.3.2	设计 Form 窗体	364
10.3.3	实现 GDI+绘图	366
10.3.4	保存、读取图像	374
10.4	实例小结与扩展	381
10.4.1	实例小结	381
10.4.2	实例扩展	381
附录 A	准备 MBOOK 数据库	382
附录 B	安装部署应用程序	396

熟悉 Visual C++.NET 开发环境

◇ 任务目标

本章主要通过了解 Visual C++ 中的各个组件来熟悉 Visual C++.NET 集成开发环境。首先,了解 Visual C++.NET 界面的组成元素和开发环境的相关组件,以及项目的概念和项目的创建、配置等。接下来,了解如何使用 Visual C++.NET 提供的各种资源编辑器来添加、编辑、删除资源等。然后,熟悉 Visual C++ 提供的配置选项,并介绍通过这些配置选项来定制开发环境。最后,熟悉 Visual C++ 开发环境中的调试器,以及调试器中的常用调试工具等。



1.1 Visual C++.NET 概述

随着计算机网络的发展及 Internet 的普及,基于 Web 浏览的应用程序变得越来越重要。为了能够在 Web 应用上占一席之地,Microsoft 于 2000 年推出了 .NET 计划,并不断发展 .NET 以计划将 Internet 作为构建操作系统的基础。Microsoft 的 .NET 平台提供创建整个 XML Web 服务的统一的开发环境 Microsoft Visual Studio.NET,它是用于快速建立企业级 Web 应用程序的集成开发环境。其中,该开发环境包含了 Visual C++、Visual Basic、Visual C# 和 Visual J# 等的开发工具。

1.1.1 了解 .NET 框架

.NET 是 Microsoft 面向 XML Web 服务的平台。XML Web 服务允许应用程序通过 Internet 进行通信和共享数据,而不管所采用的是哪种操作系统、设备和编程语言。



其中，.NET 框架（.NET Framework）是.NET 的核心。.NET 框架是由多个组件组成的庞大的框架，它执行应用程序和 Web 服务（包括类库，称为.NET 框架类库或 FCL），并提供安全性及许多其他的编程功能。使用.NET 开发的应用程序都需要在.NET 框架下运行。

.NET 框架的体系结构包括五大部分：程序设计语言及公共语言规范（CLS）、应用程序平台（ASP.NET、Windows 窗体等）、ADO.NET 及类库、公共语言运行时（CLR）和程序开发环境（Visual Studio.NET），如图 1.1 所示。

- 公共语言规范（Common Language Specification, CLS）：定义了各种语言间的互操作性规则，是许多应用程序互用的所有语言的通用功能，如图 1.2 所示。
- 应用程序平台：主要包括 ASP.NET 应用程序和 Windows Forms 应用程序，其中 ASP.NET 应用程序主要用于构建基于 Internet 的应用程序。而 Windows 窗体用于创建基于 CLR 的 Windows 图形用户界面的窗口应用程序。
- ADO.NET：主要用于操作存储在关系型数据库管理系统（DMBS）中的数据，它提供了对关系数据、XML 和应用程序数据的访问，是.NET 框架中的重要组成部分。
- 基础类库（Basic Class Library）：.NET 框架的基础类库提供许多类与接口，包括 ADO.NET、XML、IO、网络、调试、安全和多线程等，这些类库都是以命名空间的方式来组织的。
- 公共语言运行时（Common Language Runtime, CLR）：CLR 的作用是负责执行程序，并提供内存管理、线程管理、安全管理、异常处理、通用类型系统、生命周期监控等的核心服务。
- 编程工具开发环境：Visual Studio.NET 是一套多语言系列的开发工具，使用 Visual Studio.NET 可以创建 Windows 平台下的 Windows 应用程序和网络应用程序，以及创建网络服务、智能设备应用程序、Office 插件等。

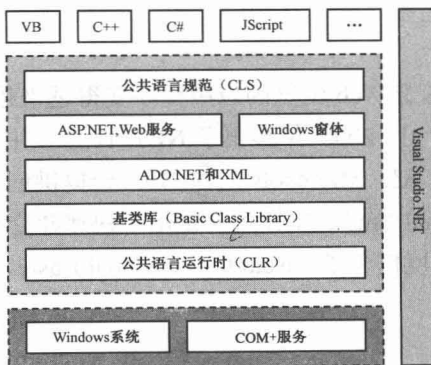


图 1.1 .NET 框架的体系结构

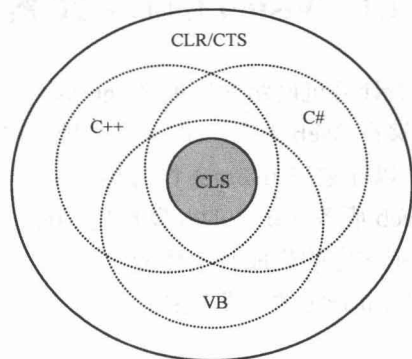


图 1.2 公共语言规范（CLS）

在.NET 框架之上，无论采用哪种编程语言编写应用程序，都首先被编译成微软中间语言（Microsoft Intermediate Language, IL），然后 IL 经过一个被称做 JIT 的（Just In Time）

编译器编译成本地 CPU 可执行的机器码。

随着 .NET 技术的不断发展, .NET Framework 的发展也经历了几个阶段, 从早期的 .NET Framework 1.0、1.1 发展到 .NET Framework 2.0 标志着 .NET 技术走向成熟, 功能也更加强大。在 2008 年, 随着 Microsoft 推出 Visual Studio 2008 开发平台, .NET Framework 又由 2.0 更新为 3.0 和 3.5。 .NET Framework 3.0、3.5 是在 .NET Framework 2.0 的基础上进行扩展, 增加了很多新特性, 如 WCF、WPF、WF、LINQ、AJAX 等。

1.1.2 托管 C++ 代码

托管是 .NET 中一个专门的概念, 它仅仅倡导一种新的编程理念。使用 .NET 框架编写程序代码通常被称为托管代码 (Managed Code), 而这些代码被编译后的程序在执行时需要依赖 CLR (如自动垃圾回收、运行库类型检查、安全支持等)。使用托管 C++ 代码, 就意味着这些代码能够被 CLR 管理, 并能开发出具有垃圾自动收集、程序间相互访问等新特性的 .NET 框架应用程序。

.NET 框架是由彼此独立而又相关的两部分组成的: CLR 和基础类库。其中, CLR 用于为应用程序提供一系列服务, 而基础类库提供了 CLR 功能的实现方法。 .NET 框架中的大部分特性 (如垃圾收集、版本控制、线程管理等) 都使用了 CLR 提供的服务。

这样, 为 CLR 编写及使用 CLR 服务的代码被称做托管代码, 而那些未使用 CLR 服务的代码被称做非托管代码。

当然, 应用程序可以完全采用托管代码进行实现, 这类应用程序的所有行为都需要依赖 CLR 及 .NET 框架类库的支持。另外, 应用程序也可以既使用托管代码又可以使用非托管代码来实现。托管应用程序的创建过程如图 1.3 所示。

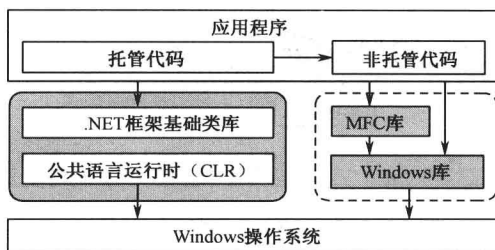


图 1.3 托管应用程序的创建过程

由于 C++ 并不是为 .NET 平台设计的语言, 所以为了能够在托管环境中应用 C++, Microsoft 引入了“托管扩展”的概念来扩充 C++, 也就是托管的 C++ 扩展。托管的 C++ 扩展在原 C++ 语言的基础上增加了一些新的关键字, 以说明使用这些扩展的 C++ 可以编写出 .NET 代码并调用 .NET 框架类库, 如 `__gc`、`__abstract`、`__value` 等。

然而, 以这种方式实现的托管 C++ 在语法上比较晦涩。例如, 在非托管 C++ 和托管 C++ 语法中的指针都通过 * 符号表示, 但是托管 C++ 在语法中的指针实际上是对象的一个引

用，它与非托管 C++ 中的指针在本质和行为上是完全不同的。由于这些原因，微软在原 C++ 的基础上提出了 C++/CLI 语言，以提供在 .NET 上进行托管编程的扩展。



1.2 Visual C++.NET 环境

Microsoft Visual Studio 2008 是一个集成开发工具，其中包含了 Visual Basic、Visual C++ 和 Visual C# 的开发组件。在集成开发环境（Integrated Develop Environment, IDE）中集成了编辑器、编译器、调试器等多个工具，使得在整个开发环境下能够进行创建、编译、调试和优化应用程序。

1.2.1 窗口元素


当启动 Visual Studio 2008 后，将显示如图 1.4 所示的启动界面。默认情况下，开发环境的窗口中包含菜单栏、工具栏、解决方案资源管理器、起始页、属性窗口、输出窗口等几个部分。单击各个窗口上的按钮  可以自动隐藏窗口，而当光标移动到窗口标签上时，将自动弹出该窗口。



图 1.4 Visual Studio 2008 启动界面

1. 菜单栏

默认情况下，开发环境窗口中包含有“文件”、“编辑”、“视图”、“工具”、“测试”、“窗口”及“帮助”菜单选项。当打开或新建一个项目后，菜单栏中会增加“项目”、“生成”及“调试”菜单选项。

- “文件”菜单包含了用于项目、文件的常用命令项，如新建、打开、保存等，以及打印和打印预览的菜单命令。
- “编辑”菜单提供了文档内容的常用操作，如剪切、复制、查找、替换等。其中还包含了一个“书签”菜单项，提供了类似浏览器的书签功能。

- “视图”菜单中主要提供了管理开发环境中各个窗口的菜单选项，可以根据实际需要选择常用的窗口及工具条。
- “项目”菜单中列出了开发项目的一些常用选项，如添加新类、添加资源、为项目添加文件、设置项目的属性等。
- “生成”菜单主要提供了生成项目的相关命令，这些菜单选项分别可以用于生产整个项目及项目中的单个文件等。
- “调试”菜单提供了运行、调试程序的菜单命令。其中主要包含了一些常用的调试命令及管理调试窗口的菜单命令等。
- “工具”菜单中提供了许多执行外部工具的菜单命令，以及定制开发环境和设置整个开发环境的菜单命令。
- “测试”菜单中主要用于创建单元测试的菜单命令，这些命令可以创建测试项目，并执行单元测试及查看测试的输出结果等。
- “帮助”菜单中提供了大量的 MSDN 帮助信息。其中包含了使用 Visual Studio 开发环境的大量项目开发技术及编程资料。

2. 工具栏

工具栏中列出了常用的工具按钮，而当打开不同的编辑器时，会出现相应的工具条。用户可以在工具栏上单击鼠标右键，在弹出一个如图 1.5 所示的工具栏菜单中，通过标记“勾号”的来选择需要显示的工具条。

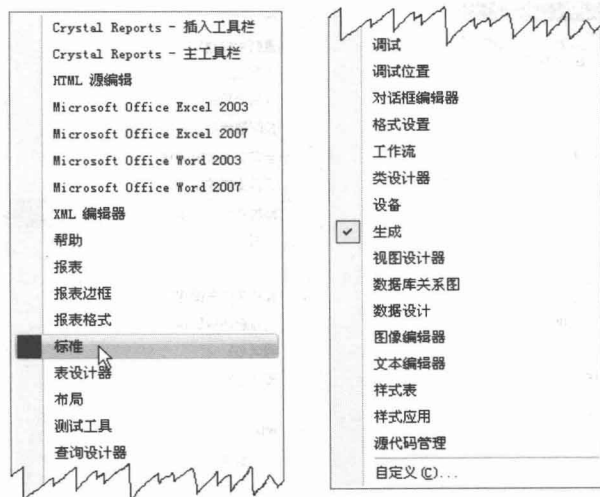


图 1.5 工具栏快捷菜单

默认情况下，工具栏上含有标准工具条和生成工具条。其中，标准工具条中包含一些常用的工具按钮及调试、窗口的管理工具按钮；生成工具条仅包含了生成项目的工具按钮。

另外，在 Visual C++.NET 开发环境中的工具栏具有“浮动”与“停泊”功能。如果将工具条拖动到窗口的中央处，则工具栏成为“浮动”的工具窗口，窗口的标题就是该工具



栏的名称。

当然，浮动和停泊两种状态可以进行切换。在“浮动”的工具窗口标题栏处双击鼠标左键或将其拖放到主窗口的四周，都能使其停泊在相应的位置处。在“停泊”工具栏的非按钮区域双击鼠标左键，可切换成“浮动”的工具窗口。

3. 解决方案资源管理器

解决方案资源管理器提供项目及其文件的有组织的视图，在开发应用程序时，该视图可以用来帮助管理解决方案中的项目和文件。其中的所有文件以层次关系被包含项目名下，并根据文件类型被划分为多个分组。其中，头文件项下包含了项目中所有的头文件；源文件项中包含项目中所有的源文件；资源项下是在项目中使用到的资源文件，而其他与项目无关的文件也被单独列在项目名下，如图 1.6 所示。

在解决方案资源管理器中，选中一个项目并单击鼠标右键，弹出如图 1.7 所示的快捷菜单。其中，“生成”和“重新生成”菜单命令将编译整个项目，“添加”菜单命令将在项目中新建一个文件或从外部添加文件，而选择“属性”菜单命令后将弹出该项目的属性设置，其中包括项目的设置、编译、链接信息等。

另外，在解决方案资源管理器中双击某个文件后，开发环境将以支持的编辑器自动打开该文件。用鼠标右键单击该文件，将弹出一个快捷菜单，选择“从项目中排除”菜单命令后将从项目中去除该文件，而如果选择“移除”菜单命令，则会弹出一个提示对话框并询问是否需要从磁盘上删除该文件，若单击“删除”按钮则从项目中去除该文件并删除磁盘上的文件。

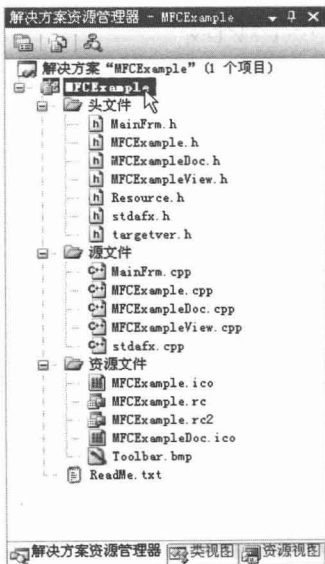


图 1.6 解决方案资源管理器

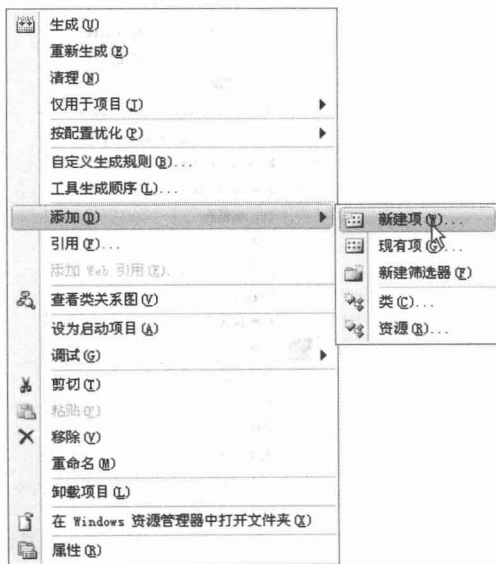


图 1.7 项目快捷菜单

4. 类视图、资源视图

在 Visual C++.NET 开发环境中包含了两个重要的视图：类视图和资源视图，它们分别

用来显示项目中所有的类、资源信息等，如图 1.8 和图 1.9 所示。



图 1.8 类视图

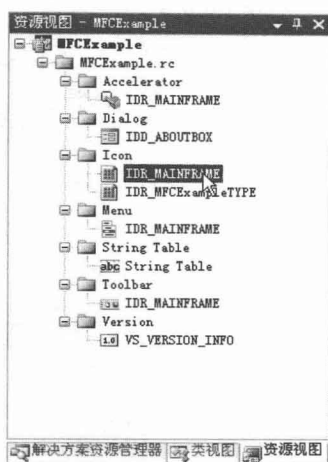


图 1.9 资源视图

类视图用于显示正在开发的应用程序中定义、引用或调用的符号。其中包含两个窗口格：上面部分是对象窗格，而下面部分是成员窗格。对象窗格包含一个可以展开的符号树，其顶级节点表示项目，而每个节点用于标识项目中使用的分层结构，如命名空间、类型、接口、枚举、类等。成员窗格中列出了属性、方法、事件、变量、常量和包含的其他项等。

资源视图显示项目中包含的资源文件。展开顶部文件夹将显示资源文件 (*.rc) 中所有资源的类型，展开每种资源类型将显示该类型的各个资源。这些资源是应用程序窗口元素的一部分，如菜单栏、工具栏、对话框等。每个资源都具有相应的资源类型以及对应的资源文件类型，并且都含有一个资源标识符，如 IDR_MAINFRAME。双击某个资源的标识符后，开发环境将使用资源编辑器自动打开这些资源，并可以修改、编辑这些资源。

5. 代码编辑窗口

当打开一个文件后，开发环境将自动打开代码编辑窗口以供编辑代码，如图 1.10 所示。代码编辑窗口中的文件标签显示当前打开的文件名，并且根据打开文件类型自动进行语法高亮显示，如以红色显示注释内容、蓝色显示关键字等。

当在代码编辑窗口中输入代码时，它会根据当前的输入自动弹出提示框。这些提示信息可以是类中的成员函数或成员变量、全局函数、函数声明等，用户可以在提示框中选择适合的内容以自动补全代码内容。

6. 工具箱

工具箱是一个浮动的树控件，它与 Windows 资源管理器的工作方式非常类似，但没有网格或连接线，如图 1.11 所示。然而工具箱中的控件并非一成不变，当打开其他编辑器或项目类型时，会更改当前工具箱中的选项卡，以及在选项卡中显示的项。例如，使用 Visual C++.NET 创建一个基于 CLR 的 Windows 窗体项目时，工具箱中的可用控件数明显增多。