

А. Мешков, Ю. Тихомиров



www.bhv.ru

www.bhv.kiev.ua

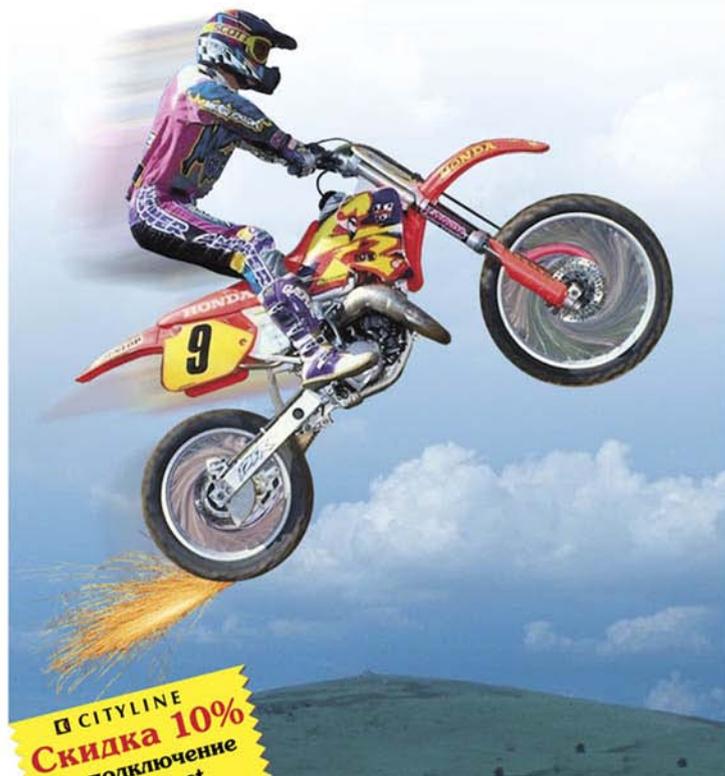
Visual C++

и MFC

2 издание

МАСТЕР

РУКОВОДСТВО ДЛЯ ПРОФЕССИОНАЛОВ



CITYLINE
Скидка 10%
на подключение
к Internet



Дискета содержит
файлы всех
примеров

Андрей Мешков
Юрий Тихомиров

Visual C++ и MFC

Издание второе



Дюссельдорф Киев
Москва Санкт-Петербург

УДК 681.3.06

Книга посвящена объектно-ориентированному программированию для 32-разрядных систем Windows 95/98 и Windows NT с использованием компилятора Visual C++ 6 и библиотеки классов MFC версии 4.23.

Книга содержит исчерпывающую информацию о библиотеке классов MFC (Microsoft Foundation Classes) и методах создания приложений Windows на основе классов этой библиотеки, а также большое количество наглядных и подробно прокомментированных примеров приложений, полные тексты которых приведены на сопроводительной дискете.

*Для программистов, владеющих языками C и C++
и имеющих опыт программирования в системе Windows*

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зав. редакцией	<i>Наталья Таркова</i>
Ответственный редактор	<i>Татьяна Темкина</i>
Компьютерная верстка	<i>Наталья Смирнова</i>
Корректурa	<i>Светлана Журавина</i>
Дизайн обложки	<i>Наталья Смирнова</i>
Производство	<i>Николай Тверских</i>

Мешков А. В., Тихомиров Ю. В.

Visual C++ и MFC. — 2-е изд. — СПб.: БХВ — Санкт-Петербург, 1999. — 1040 с.: ил.

ISBN 5-8206-0073-8

© Мешков А. В., Тихомиров Ю. В., 1999

© Оформление, издательство "БХВ — Санкт-Петербург", 1999

Лицензия ЛР № 065953 от 15.06.98. Подписано в печать 26.10.99.

Формат 70×100¹/₁₆. Печать офсетная. Усл. печ. л. 83,85.

Тираж 3000 экз. Заказ

"БХВ — Санкт-Петербург", 198005, Санкт-Петербург, Измайловский пр., 29.

Гигиеническое заключение на продукцию, товар, № 77.99.1.953.П.950.3.99 от 01.03.1999 г. выдано Департаментом ГСЭН Минздрава России.

Отпечатано с готовых диапозитивов
в ордена Трудового Красного Знамени ГП "Техническая книга"
Комитета Российской Федерации по печати.
198005, Санкт-Петербург, Измайловский пр., 29.

Содержание

ВВЕДЕНИЕ	1
ЧАСТЬ I. ОБЗОР VISUAL C++ 6	9
ГЛАВА 1. ЗНАКОМЬТЕСЬ — VISUAL C++ 6	9
Что нового в этой версии?	9
НАСТРОЙКИ ПРОЕКТА.....	13
Настройка опций компилятора.....	14
Настройка опций компоновщика	16
ГЛАВА 2. ИНТЕГРИРОВАННАЯ СРЕДА РАЗРАБОТКИ (IDE)	19
ЧАСТЬ II. ПРОГРАММИРОВАНИЕ ПОД MS WINDOWS	31
ГЛАВА 3. ЗНАКОМСТВО С WINDOWS	33
ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС	33
Окна, элементы управления, модальные и немодальные блоки диалога	34
Ресурсы приложения	39
СИСТЕМА, УПРАВЛЯЕМАЯ СООБЩЕНИЯМИ	40
ИНТЕРФЕЙС ПРИКЛАДНОГО ПРОГРАММИРОВАНИЯ	42
УПРАВЛЕНИЕ ГРАФИЧЕСКИМ ВЫВОДОМ	44
Графические объекты	44
Аппаратно-независимый графический вывод	45
Контекст устройства	45
Графические режимы	48
Работа со шрифтами	49
APISHELL — ПРИМЕР ПРИЛОЖЕНИЯ	50
ГЛАВА 4. СТРУКТУРА ПРИЛОЖЕНИЙ ДЛЯ WINDOWS	59
СТАНДАРТНЫЙ ЦИКЛ ОБРАБОТКИ СООБЩЕНИЙ.....	61
ОКОННАЯ ПРОЦЕДУРА	62
ОСНОВНЫЕ ТИПЫ ОКОННЫХ СООБЩЕНИЙ WINDOWS	63
APIWin — пример приложения.....	64

ЧАСТЬ III. БИБЛИОТЕКА КЛАССОВ MICROSOFT FOUNDATION CLASSES (MFC)	75
ГЛАВА 5. ОБЗОР БИБЛИОТЕКИ MFC	77
ИЕРАРХИЯ КЛАССОВ MFC.....	78
СОВЖЕСТ — ОСНОВА ВСЕХ КЛАССОВ.....	78
КЛАССЫ, ОПРЕДЕЛЯЮЩИЕ АРХИТЕКТУРУ ПРИЛОЖЕНИЯ	84
Классы приложения и поддержки потоков	85
Классы маршрутизации команд.....	86
Классы документов.....	87
Классы шаблонов документов.....	87
ОКНА, БЛОКИ ДИАЛОГА И ЭЛЕМЕНТЫ УПРАВЛЕНИЯ	88
Классы фреймов окон	88
Классы представлений	89
Классы блоков диалога.....	90
Классы элементов управления	92
Классы панелей элементов управления	94
КЛАССЫ ДЛЯ ВЫВОДА НА ЭКРАН И ПЕЧАТЬ.....	95
Классы контекста устройства	95
Классы графических объектов.....	95
КЛАССЫ ПРОСТЫХ ТИПОВ ДАННЫХ.....	96
МАССИВЫ, СПИСКИ И АССОЦИАТИВНЫЕ СПИСКИ.....	97
Классы шаблонов массивов, списков и ассоциативных списков	97
Готовые к использованию классы массивов	98
Готовые к использованию классы списков	98
Готовые к использованию классы ассоциативных списков	98
КЛАССЫ ДЛЯ РАБОТЫ С ФАЙЛАМИ И БАЗАМИ ДАННЫХ	99
Классы файлов ввода/вывода	99
Классы для работы с DAO.....	100
Классы для работы с ODBC	100
КЛАССЫ ДЛЯ РАБОТЫ В INTERNET И СЕТЯХ	101
Классы ISAPI.....	101
Классы сокетов Windows.....	102
КЛАССЫ ДЛЯ РАБОТЫ С OLE.....	102
Контейнерные классы OLE	103
Серверные классы OLE.....	103
Классы OLE для передачи данных и drag-and-drop.....	104
Классы автоматизации OLE	104
Классы, связанные с OLE.....	104
КЛАССЫ ДЛЯ ОТЛАДКИ И ОБРАБОТКИ ИСКЛЮЧЕНИЙ	105
Классы поддержки отладки	106
Классы обработки исключений.....	106

ГЛАВА 6. МАКРОСЫ, ГЛОБАЛЬНЫЕ ФУНКЦИИ И ПЕРЕМЕННЫЕ	107
Типы данных.....	107
Получение информации о приложении.....	108
Модель объекта времени выполнения.....	109
Диагностика объектов.....	111
Основные макросы.....	111
Форматирование строк и окно сообщения.....	112
ЧАСТЬ IV. ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ БИБЛИОТЕКИ MFC	115
ГЛАВА 7. ОСНОВНЫЕ СОСТАВЛЯЮЩИЕ ПРИЛОЖЕНИЯ НА БАЗЕ БИБЛИОТЕКИ КЛАССОВ MFC	117
Соглашения об именах MFC.....	121
Включаемые файлы.....	122
Функция WinMain.....	122
Класс CWinApp.....	125
Минимальная программа для Windows.....	131
Регистрация класса окна.....	134
Создание окна.....	139
ГЛАВА 8. ОБРАБОТКА СООБЩЕНИЙ В БИБЛИОТЕКЕ КЛАССОВ MFC	141
Цикл обработки сообщений MFC.....	141
Категории сообщений.....	143
Карта сообщений.....	144
Компоненты карты сообщений.....	146
Класс CCmdTarget.....	150
Стандартный маршрут команды.....	151
Команды обновления и класс CCmdUI.....	154
Функции для работы с сообщениями.....	155
ЧАСТЬ V. СОЗДАНИЕ SDI- И MDI-ПРИЛОЖЕНИЙ НА БАЗЕ БИБЛИОТЕКИ MFC	159
ГЛАВА 9. КЛАССЫ ОКОН БИБЛИОТЕКИ MFC	161
Окна, определенные в ОС Windows.....	161
Класс CWnd.....	166
Класс CFrameWnd.....	169
Создание окна и библиотека MFC.....	171
Создание главного окна SDI-приложения.....	185
Создание дочерних окон.....	191

Ограничение размеров окна	198
Функция GetSystemMetrics	199
Ограничение доступа к окну	200
ГЛАВА 10. СОЗДАНИЕ МНОГОДОКУМЕНТНЫХ ПРИЛОЖЕНИЙ.....	202
КЛАСС CMDIFRAMEWND.....	203
КЛАСС CMDICILDWND.....	206
MDI — ПРИМЕР ПРИЛОЖЕНИЯ.....	207
Обработка команд меню	226
Unicode	227
Обработка команд меню (продолжение).....	229
Обработка сообщения WM_PAINT	232
ЧАСТЬ VI. ЭЛЕМЕНТЫ ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ	237
ГЛАВА 11. ПАНЕЛИ ЭЛЕМЕНТОВ УПРАВЛЕНИЯ.....	239
КЛАСС CCONTROLBAR	239
КЛАСС CTOOLBAR	243
КЛАСС CTOOLBARCTRL.....	249
КЛАСС CSTATUSBAR.....	257
КЛАСС CSTATUSBARCTRL	261
КЛАСС CDIALOGBAR	265
КЛАСС CREBAR.....	266
КЛАСС CREBARCTRL.....	268
CONTROLBARS — ПРИМЕР ПРИЛОЖЕНИЯ	275
ГЛАВА 12. МЕНЮ	293
ОСНОВНЫЕ ТИПЫ МЕНЮ.....	293
КЛАСС CMENU	296
СОЗДАНИЕ МЕНЮ НА ОСНОВЕ ШАБЛОНА	306
ДИНАМИЧЕСКОЕ СОЗДАНИЕ МЕНЮ	311
Динамическое изменение меню.....	313
Системное меню	320
Контекстное меню	323
Создание собственных маркеров состояния.....	326
Самоотображение элементов меню	328
ГЛАВА 13. АКСЕЛЕРАТОРЫ, ГРАФИЧЕСКИЕ И ДРУГИЕ РЕСУРСЫ	330
АКСЕЛЕРАТОРЫ	330
ГРАФИЧЕСКИЕ РЕСУРСЫ.....	331
СТРОКОВЫЕ РЕСУРСЫ	333
РЕСУРСЫ, ОПРЕДЕЛЯЕМЫЕ ПРОГРАММИСТОМ	334

ЧАСТЬ VII. БЛОКИ ДИАЛОГА	335
ГЛАВА 14. МОДАЛЬНЫЕ И НЕМОДАЛЬНЫЕ БЛОКИ ДИАЛОГА.....	337
КЛАСС CDIALOG	340
ТЕМПЛАТЕ — ПРИМЕР ПРИЛОЖЕНИЯ.....	352
БЛОК ДИАЛОГА В КАЧЕСТВЕ ГЛАВНОГО ОКНА ПРИЛОЖЕНИЯ	367
ОБМЕН ДАННЫМИ С БЛОКОМ ДИАЛОГА	368
СОЗДАНИЕ БЛОКА ДИАЛОГА НА ОСНОВЕ ШАБЛОНА В ПАМЯТИ	372
ГЛАВА 15. СТАНДАРТНЫЕ БЛОКИ ДИАЛОГА	377
КЛАСС CFILEDIALOG	377
КЛАСС CCOLORDIALOG	390
КЛАСС CFONTDIALOG	395
КЛАСС CFINDREPLACEDIALOG.....	401
КЛАСС CPRINTDIALOG	405
КЛАСС CPAGESETUPDIALOG.....	414
ЧАСТЬ VIII. ГРАФИЧЕСКИЙ ИНТЕРФЕЙС УСТРОЙСТВ (GDI)	421
ГЛАВА 16. КЛАССЫ ГРАФИЧЕСКОГО ИНТЕРФЕЙСА	423
КОНТЕКСТЫ УСТРОЙСТВ	424
ГРАФИЧЕСКИЕ ОБЪЕКТЫ	426
ГЛАВА 17. КЛАССЫ КОНТЕКСТОВ УСТРОЙСТВ.....	430
КЛАСС CDC	430
Создание объекта класса и его связь с контекстом устройства	430
Настройка процесса рисования.....	434
Режимы отображения	440
Функции рисования.....	446
Отображение битовых массивов.....	452
Управление выводом текста.....	456
Дополнительные функции	462
ГЛАВА 18. КЛАССЫ ГРАФИЧЕСКИХ ОБЪЕКТОВ.....	463
КЛАСС CGDIОБЪЕСТ	463
КАРАНДАШ (КЛАСС CPEN)	468
КИСТЬ (КЛАСС CBRUSH)	474
ШРИФТ (КЛАСС CFONT).....	475
БИТОВЫЙ МАССИВ (КЛАСС СВИТМАР).....	482
РЕГИОН (КЛАСС CRGN).....	489
Создание регионов.....	489
Комбинирование регионов	491

PQClock — ПРИМЕР ПРИЛОЖЕНИЯ.....	494
Класс приложения.....	494
Класс окна.....	497
ЧАСТЬ IX. АРХИТЕКТУРА "ДОКУМЕНТ/ПРЕДСТАВЛЕНИЕ" (DOCUMENT/VIEW).....	511
ГЛАВА 19. ОСНОВНЫЕ ПОЛОЖЕНИЯ	513
МЕСТО ОБЪЕКТА-ПРИЛОЖЕНИЯ В АРХИТЕКТУРЕ "ДОКУМЕНТ/ПРЕДСТАВЛЕНИЕ"	516
Класс CDocTemplate.....	516
Класс CSingleDocTemplate	521
Класс CMultiDocTemplate.....	522
РОЛЬ ФРЕЙМОВ В АРХИТЕКТУРЕ "ДОКУМЕНТ/ПРЕДСТАВЛЕНИЕ"	530
СЕРИАЛИЗАЦИЯ.....	535
Класс CArchive	541
ГЛАВА 20. ДОКУМЕНТ И ЕГО ПРЕДСТАВЛЕНИЯ.....	544
ДОКУМЕНТЫ.....	545
Класс CDocument.....	547
ПРЕДСТАВЛЕНИЯ.....	555
Класс CView.....	558
Класс CCtrlView	562
Класс CEditView	562
Классы CListView и CTreeView.....	568
Класс CScrollView	568
Класс CSplitterWnd	573
ГЛАВА 21. ПЕЧАТЬ И ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР ДОКУМЕНТОВ.....	583
ВЫБОР И НАСТРОЙКА ПАРАМЕТРОВ ПРИНТЕРА.....	585
СОЗДАНИЕ КОНТЕКСТА УСТРОЙСТВА.....	588
ПЕЧАТЬ ДОКУМЕНТОВ И БИБЛИОТЕКА MFC.....	588
ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР ДОКУМЕНТА.....	598
ЧАСТЬ X. ЭЛЕМЕНТЫ УПРАВЛЕНИЯ	607
ГЛАВА 22. ОБЗОР СТАНДАРТНЫХ ЭЛЕМЕНТОВ УПРАВЛЕНИЯ.....	609
СОЗДАНИЕ ЭЛЕМЕНТОВ УПРАВЛЕНИЯ.....	610
Создание элементов управления в редакторе ресурсов.....	610
Создание элементов управления в тексте приложения.....	610
ГЛАВА 23. КЛАССЫ СТАНДАРТНЫХ ЭЛЕМЕНТОВ УПРАВЛЕНИЯ ...	612
СТАТИЧЕСКИЕ ЭЛЕМЕНТЫ УПРАВЛЕНИЯ.....	612

Создание	612
Отображение битового массива.....	615
Отображение значка	616
Отображение курсора	616
Отображение расширенных метафайлов.....	617
Изменение цвета	617
КНОПКИ	619
Создание	622
Определение и изменение текущего состояния кнопок	622
Определение и изменение стиля кнопки.....	624
Назначение изображений.....	624
Переопределяемые функции	625
ПРОСТЕЙШИЙ ТЕКСТОВЫЙ РЕДАКТОР	625
Инициализация	626
Выделение текста	628
Работа со строками	629
Работа с блоками текста.....	631
Параметры и флаги.....	632
Дополнительные функции	633
СПИСОК.....	634
Инициализация	635
Изменение параметров списка.....	636
Функции для работы с содержимым списка	638
Операции над элементами списка	641
Переопределяемые функции	643
Пример самоотображаемого списка	646
КОМБИНИРОВАННЫЙ СПИСОК.....	649
Инициализация	650
Основные операции с комбинированным списком	652
Операции со строками и виртуальные функции	654
ПОЛОСА ПРОКРУТКИ.....	655
Создание	657
Работа с полосой прокрутки.....	658

ГЛАВА 24. КЛАССЫ ЭЛЕМЕНТОВ УПРАВЛЕНИЯ С РАСШИРЕННЫМИ ВОЗМОЖНОСТЯМИ 663

КНОПКИ С НАНЕСЕННЫМ БИТОВЫМ ИЗОБРАЖЕНИЕМ	663
Кнопки в блоке диалога.....	664
Кнопки, созданные программным путем	665
Функции класса	665
СПИСОК, ИМЕЮЩИЙ ФЛАЖКИ.....	667
Инициализация	667
Функции для работы с расширенным списком	667
Виртуальные функции.....	668

КОМБИНИРОВАННЫЙ СПИСОК С ПОДДЕРЖКОЙ СПИСКА ОБРАЗОВ.....	674
Создание комбинированного списка	674
Основные операции.....	675
Функции работы с атрибутами.....	676
СПИСОК С ВОЗМОЖНОСТЬЮ ПЕРЕТАСКИВАНИЯ ЭЛЕМЕНТОВ	678
Ограничения на использование	678
Создание списка	678
Функции, предназначенные для перемещения элементов	679
Методика обхода ограничений.....	680
Реализация копирования элементов списка.....	680
ГЛАВА 25. ВСПОМОГАТЕЛЬНЫЕ ЭЛЕМЕНТЫ УПРАВЛЕНИЯ	683
ПРОСМОТР ВИДЕОКЛИПОВ	683
Элемент управления "анимация"	684
Класс CAnimateCtrl.....	684
Уведомления	686
"ГОРЯЧАЯ КЛАВИША"	688
Глобальные "горячие клавиши".....	691
Поточно-ориентированные "горячие клавиши".....	691
Пример использования класса CHotKeyCtrl	693
ИНДИКАТОР.....	698
Использование индикатора в панели диалога.....	700
ЛИНЕЙКА С ПОЛЗУНКОМ.....	703
Уведомления	703
Создание элемента управления	704
Параметры элемента управления	706
Положение меток.....	706
Положение ползунка	707
СЧЕТЧИК.....	707
Уведомления	708
Автоматическое изменение.....	710
Создание счетчика	710
Параметры элемента управления	711
ВСПЛЫВАЮЩАЯ ПОДСКАЗКА	713
Оконная поддержка	714
Возвращаемся к классу CToolTipCtrl.....	716
ГЛАВА 26. КОМПЛЕКСНЫЕ ЭЛЕМЕНТЫ УПРАВЛЕНИЯ	720
СПИСОК ИЗОБРАЖЕНИЙ.....	720
Создание списка	721
Изменение содержимого списка	722
Рисование изображений.....	723
Перемещение изображений	724
Информация о списке.....	727

ЗАГОЛОВОК	727
Создание элемента заголовка	728
Атрибуты элементов заголовка	732
Обработка извещений элемента заголовка	733
Элемент заголовка, определяемый пользователем	735
ПРОСМОТР СПИСКА	736
Режимы вывода	736
Создание просмотра списка	737
Работа со столбцами	741
Параметры просмотра списка	743
Функции для работы со списком в целом	744
Рабочие области просмотра списка	746
Списки изображений	746
Виртуальные списки	748
Основные и дополнительные поля	748
Записи по запросу	748
Поиск и сортировка записей	751
Редактирование надписей записей	752
Обработка уведомлений	753
Просмотр списка с возможностью перемещения записей	754
Переопределяемые функции	755
ListCtrl — пример приложения	755
ПРОСМОТР ДЕРЕВА	763
Создание просмотра дерева	763
Списки изображений	764
Изменение содержимого дерева	765
Поиск и сортировка записей дерева	770
Обработка уведомлений	772
КЛАСС CMONTHCALCTRL	774
Обработка уведомлений	777
КЛАСС CDATEIMECTRL	777
Обработка уведомлений	780
ГЛАВА 27. РАСШИРЕННЫЙ ТЕКСТОВЫЙ РЕДАКТОР	781
КЛАСС CRICHEDITCTRL	781
Создание объекта класса CRichEditCtrl	782
Изменение параметров редактора	783
Настройка окна редактора	785
Размер содержимого редактора	786
Работа со строками	787
Выделение содержимого редактора	789
Работа с буфером обмена	791
Печать содержимого редактора	792
Уведомляющие сообщения	793

Расширенные возможности	795
РАСШИРЕННЫЙ РЕДАКТОР В РАМКАХ АРХИТЕКТУРЫ	
"ДОКУМЕНТ/ПРЕДСТАВЛЕНИЕ"	802
Класс CRichEditView	804
Класс CRichEditDoc	810
ГЛАВА 28. НАБОРЫ СВОЙСТВ	811
Создание набора свойств	812
Создание страницы свойств	813
Создание объекта "набор свойств"	814
Настройка окна набора свойств	815
Добавление страниц	816
Создание модального окна свойств	818
Создание немодального окна набора свойств	818
Обмен данными	819
Операция Apply	820
КЛАСС CPROPERTYSHEET	821
Операции над набором свойств	825
Вспомогательные функции	827
Изменение параметров отображения	828
КЛАСС CPROPERTYSHEETEX	830
КЛАСС CPROPERTYPAGE	831
Создание страницы свойств	834
Переопределяемые функции	835
КЛАСС CPROPERTYPAGEEX	836
ВКЛАДКИ	838
Создание вкладок	838
Вспомогательные функции	841
Изменение вкладок	842
МАСТЕРА	845
Создание мастеров	846
Переопределяемые функции	846
ЧАСТЬ XI. ПОДДЕРЖКА МНОГОЗАДАЧНОСТИ	849
ГЛАВА 29. ВЫТЕСНЯЮЩАЯ МНОГОЗАДАЧНОСТЬ	851
ОБЩИЕ ПОЛОЖЕНИЯ	851
КЛАССЫ ПРОЦЕССОВ И ПРИОРИТЕТЫ ПОТОКОВ	853
Класс Idle	854
Класс Normal	854
Класс High	855
Класс Realtime	855
Относительные приоритеты потоков	855

ВЗАИМОДЕЙСТВИЕ ПРОГРАММ.....	858
СИНХРОНИЗАЦИЯ ПОТОКОВ.....	859
Критические секции.....	859
Ожидающие функции.....	860
Объекты синхронизации.....	861
БИБЛИОТЕКА MFC И МНОГОЗАДАЧНОСТЬ.....	864
Класс и функции для поддержки многопоточковых приложений.....	865
Классы и функции для поддержки взаимодействия потоков и процессов.....	866
ГЛАВА 30. СОЗДАНИЕ ПОТОКОВ.....	868
ПРИМЕР СОЗДАНИЯ НАСЛЕДУЕМЫХ ПОТОКОВ.....	871
СОЗДАНИЕ ИНТЕРФЕЙСНЫХ ПОТОКОВ.....	874
КЛАСС CWinThread.....	875
СОЗДАНИЕ ИНТЕРФЕЙСНЫХ ПОТОКОВ (ПРОДОЛЖЕНИЕ).....	880
CUIThreadApp — класс всего приложения.....	881
CUIThreadFrame — класс основного окна.....	881
CLineThread — класс дополнительного потока.....	886
CLineWnd — класс окна "бегущая строка".....	888
CEditWnd — класс окна редактора.....	893
СОЗДАНИЕ РАБОЧИХ ПОТОКОВ.....	895
Управляющая функция и ее параметр.....	895
Пример управляющей функции.....	896
ЗАВЕРШЕНИЕ ПОТОКОВ.....	897
Нормальное завершение.....	897
Экстренное завершение.....	899
Код завершения.....	900
Права доступа к потоку.....	901
ГЛАВА 31. ВЗАИМОДЕЙСТВИЕ ПОТОКОВ.....	903
ВЗАИМОИСКЛЮЧЕНИЕ ДОСТУПА.....	903
Организация доступа к объектам классов MFC.....	905
Списки дескрипторов объектов Windows.....	906
Graphics — пример приложения.....	907
ИСПОЛЬЗОВАНИЕ КЛАССОВ ОБЪЕКТОВ СИНХРОНИЗАЦИИ.....	915
Создание класса разделяемых объектов.....	916
Использование классов объектов синхронизации.....	921
КЛАССЫ ОБЪЕКТОВ СИНХРОНИЗАЦИИ.....	922
Класс CSyncObject — базовый класс объектов синхронизации.....	922
Критическая секция (класс CCriticalSection).....	925
Взаимоисключение (класс CMutex).....	926
Событие (класс CEvent).....	927
Семафор (класс CSemaphore).....	929
КЛАССЫ, ОБЕСПЕЧИВАЮЩИЕ СИНХРОНИЗАЦИЮ.....	930

TERMINAL — МНОГОПОТОКОВОЕ ПРИЛОЖЕНИЕ, ВЗАИМОДЕЙСТВУЮЩЕЕ С ПОСЛЕДОВАТЕЛЬНЫМ ПОРТОМ	934
Класс блока диалога и функции потоков	934
Окно редактора	942
Работа с последовательным портом.....	943
Кольцевой буфер общего доступа.....	944
ЧАСТЬ XII. INTERNET И БАЗЫ ДАННЫХ	947
ГЛАВА 32. РАБОТА С ИНФОРМАЦИОННЫМ СЕРВЕРОМ INTERNET	949
ОБЗОР ISAPI (API СЕРВЕРА INTERNET).....	950
КЛАССЫ ISAPI	953
Класс CHttpServer	954
Класс CHttpServerContext	955
Карта разбора	956
СОЗДАНИЕ РАСШИРЕНИЯ ISAPI	957
ГЛАВА 33. КЛАССЫ БИБЛИОТЕКИ MFC ДЛЯ РАБОТЫ С БАЗАМИ ДАННЫХ	961
КЛАССЫ ДЛЯ РАБОТЫ С ODBC	961
Класс CDatabase	961
Класс CRecordset.....	965
ГЛАВА 34. ИСПОЛЬЗОВАНИЕ РАСШИРЕНИЯ ISAPI.....	982
ОПИСАНИЕ СОПРОВОДИТЕЛЬНОЙ ДИСКЕТЫ	993
ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ	996

Введение

И снова программирование для Windows. Какой язык программирования вы бы ни выбрали, очевидно, что базой для этого является Windows API (Application Programming Interface — интерфейс прикладного программирования), освоив его, можно писать программы любой степени сложности, используя все возможности, предоставляемые операционной системой. Но сложность современных приложений так возросла, что разработка коммерческого программного обеспечения с использованием только Windows API и языка С уже не может удовлетворить программиста. Естественно, что технология программирования "не могла остаться в стороне", и в настоящее время предлагается два подхода к построению сложного программного обеспечения. С одной стороны, это использование систем визуального программирования. А с другой — применение библиотек классов, которые самостоятельно выполняют массу черновой работы и при этом гораздо в большей степени, нежели система Windows сама по себе, "ведут" программиста, структурируя и облегчая процесс создания программного обеспечения. Не будем сравнивать — каждый вправе сам решить, что ему больше подходит. Скажем только, что основная проблема, с которой сталкивается программист, начиная работать в некоторой операционной системе, состоит в понимании фундаментальных принципов и моделей, включенных в ее архитектуру. И только проникнувшись "образом мышления Windows", начинаешь понимать, что программирование под Windows не сложнее любого другого. А какой из подходов больше "прячет" эти принципы и модели, пусть каждый решает сам.

Мы не ставили своей целью написать традиционный учебник по программированию в Windows — об этом написано (и пишется) достаточно много хороших (и не очень) книг. Основным объектом нашего внимания является библиотека классов MFC. И именно *детальному* знакомству с ней посвящена основная часть книги. Но, с другой стороны, классы библиотеки настолько тесно связаны с системой Windows, а многие непосредственно инкапсулируют ее объекты, что мы не могли не описать саму систему Windows, предоставляемые ею возможности и те механизмы, без понимания которых попросту невозможно создать Windows-приложение. Тем не менее, мы отдаем себе отчет в том, что эти разделы не могут заменить литературу, посвященную программированию в Windows с использованием функций API.

У библиотеки классов MFC — основного предмета данной книги — две основные цели. Во-первых, скрыть от программиста то, что является скорее вспомогательным, чем обязательным аспектом при написании Windows-программ — всю "черновую работу". В частности, при использовании MFC

не нужно переносить из приложения в приложение практически не изменяющуюся функцию WinMain, а можно непосредственно приступить к решению своей конкретной задачи.

Во-вторых, обеспечить разработчиков средством, которое, в отличие от Win32 API, более структурированно и понятно. Материал книги как раз и посвящен представлению классов и методов их использования, которое позволяет существенно упростить создание как простейших приложений утилитарного характера, так и сложных систем.

Несмотря на то что сама библиотека классов сложна и многогранна, и на сегодняшний день существует более десяти ее версий, для начала работы с ней вполне достаточно владеть языком программирования C++. Новичкам даже не нужно знать принципы программирования в Windows. Дело в том, что в компилятор Visual C++ включено специальное средство, позволяющее автоматизировать процесс создания приложения. Это средство — MFC AppWizard — создаст для вас работающее приложение. Конечно, это будет только шаблон, и наполнять его конкретным содержанием придется самостоятельно. Но первоначальное знакомство со сгенерированным кодом принесет новичкам значительную пользу.

Должны сразу оговориться, что в одной книге невозможно осветить все аспекты программирования в Windows, как с использованием, так и без использования библиотеки классов MFC — это слишком обширная тема и очень трудно определить, где следует остановиться. Тем не менее, в предлагаемую книгу мы постарались включить достаточное количество информации, чтобы не только помочь начать писать собственные приложения для Windows, но и предоставить опытным программистам разнообразный материал не только по стандартным и уже не раз описанным в литературе возможностям (хотя и со своей точки зрения), но и по тем "изюминкам", которые до сих пор нигде не описаны и были найдены авторами как в процессе работы, так и при написании книги.

В этом издании мы рассматриваем библиотеку MFC версии 4.23, предназначенную для 32-разрядных операционных систем Windows 95/98 и Windows NT и включенную в шестую версию Visual C++.

Материала по предлагаемой теме так много, что от рассмотрения многих вопросов пришлось отказаться. Настоящее издание, в отличие от предыдущего, выходит в одном томе, поэтому мы убрали некоторые главы, в основном, по программированию с использованием Win32 API. Тем не менее, мы добавили часть, посвященную современным вопросам программирования для Internet и баз данных. Следует учитывать также, что изложение материала, в силу специфики Windows, не может быть последовательным. Долгие поиски привели к оптимальной, на наш взгляд, структуре изложения. За основу мы взяли три типа стандартных приложений Windows: приложения, поддерживающие SDI (Single Document Interface — однодокументный ин-

терфейс) и MDI (Multiple Document Interface — многодокументный интерфейс), а также приложение на базе блока диалога.

- *Часть I (главы 1 и 2)* посвящена краткому знакомству с Visual C++ 6 и интегрированной средой разработки — IDE, общим принципам их организации и тому новому, что появилось в шестой версии.
- В *Части II (главы 3 и 4)* приведен краткий обзор ОС Windows, а также рассмотрена общая структура приложений для этих операционных систем.
- *Часть III (главы 5 и 6)* посвящена обзору классов библиотеки MFC. Подробно описан базовый класс библиотеки — CObject и кратко описаны все компоненты библиотеки. В главе 6 представлены глобальные библиотечные функции и макросы, без которых не обходится ни одно приложение, написанное с использованием даже базовых возможностей библиотеки.
- *Часть IV (главы 7 и 8)* содержит описание требований к структуре и составляющим каждого приложения, созданного с использованием библиотеки классов MFC, а также рассматривается класс приложения CWinApp (глава 7). Глава 8 посвящена подробному изложению одного из основных механизмов Windows — работы с сообщениями, естественно с точки зрения библиотеки MFC.

Эти четыре части представляют собой некоторое теоретическое введение в Windows и библиотеку MFC, но при этом не забыты и практические аспекты программирования.

- *Часть V (главы 9 и 10)* посвящена основным объектам Windows — окнам и созданию на их основе однодокументных и многодокументных приложений. В главе 9 подробно рассмотрен базовый класс для всех типов и видов окон — класс CWnd, а также механизм создания однодокументных приложений. Глава 10 целиком посвящена описанию процесса создания многодокументных приложений.
- Элементы интерфейса пользователя подробно рассмотрены в *Части VI (главы 11—13)*. Глава 11 знакомит читателя с мощными средствами создания и использования панелей элементов управления, таких как панель инструментов и строка состояния. Меню, графические и другие ресурсы подробно рассматриваются в главах 12 и 13.
- *Часть VII (главы 14 и 15)* полностью охватывает третий тип стандартных приложений Windows — приложения на базе блока диалога. В главе 14 рассматриваются основные принципы создания и работы с блоками диалога и класс CDialog, предназначенный для решения этих задач. Системные (или стандартные) блоки диалога описаны в главе 15. Они позволяют настраивать практически все параметры (открытие и сохранение файла, выбор шрифта и цвета, поиск и т. д.), которые могут понадобиться создателям приложений.

- *Часть VIII (главы 16—18)* содержит практические аспекты процесса графического вывода. Знакомство с графическими возможностями Windows и библиотеки классов MFC в дальнейшем упростит понимание того, например, как следует создавать самоотображаемые (owner-draw) элементы управления. Описываемые классы практически полностью инкапсулируют соответствующие объекты Windows, не добавляя ничего нового. Однако функции этих классов группируют функции Win32 API, которые так или иначе вызываются в реализации классов. Одного этого было бы достаточно, чтобы создать и использовать классы взамен функций Win32 API. Сама реализация класса уже сводит воедино все или практически все необходимые возможности. Описывая функции классов, мы раскрыли все, что поддерживается в Windows. Таким образом, эту часть можно рассматривать как всеобъемлющее руководство по использованию возможностей графического вывода. Здесь также представлен большой набор примеров функций и приложений, который позволит даже новичку в программировании для Windows разобраться в том, как следует осуществлять графический вывод.
- В *Части IX (главы 19—21)* сосредоточено все, что необходимо знать о поддержке со стороны библиотеки MFC архитектуры "документ/представление" (Document/View architecture). Данная архитектура напрямую не поддерживается Windows, поэтому только описания классов и функций явно недостаточно. В части представлена взаимосвязь различных классов, последовательность вызовов тех или иных функций. Эти знания позволят вам в дальнейшем легко, с минимальными накладными расходами переопределять только те функции, новое содержание которых наполнит нужным вам смыслом разрабатываемые приложения.
- *Часть X (главы 22—27)* посвящена классам, инкапсулирующим элементы управления Windows. В главах 22—24 сосредоточена информация о стандартных элементах управления (controls) и большая часть теории, касающейся всех элементов управления. Главы 25 и 26 посвящены новым общим элементам управления (common controls). В главе 27 описывается элемент управления Windows — текстовый редактор формата RTF, который позволяет легко и быстро создать приложение наподобие WordPad, входящего в комплект поставки Windows. В этой части не только подробно излагаются и описываются соответствующие классы, но также содержится информация по самим элементам управления и описывается ряд тонких моментов, которые вряд ли можно найти в документации.
- *Часть XI (глава 28)* завершает знакомство с классами, инкапсулирующими элементы управления Windows. Здесь собрана информация, которая позволит создавать наборы свойств, состоящие из нескольких блоков диалога с вкладками, и мастера, также представляющие собой наборы блоков диалога. Различие между ними заключается в том, что в наборе свойств пользователь переходит от одного блока диалога к другому, вы-

бирая соответствующую вкладку, доступную в произвольный момент времени, а в мастере переход от одного к другому может осуществляться только последовательно. Но несмотря на различия, эти элементы управления по способу работы с ними очень схожи и базируются на одних и тех же классах — `CPropertySheet(Ex)`, `CPropertyPage(Ex)` и `CTabCtrl`.

- И Windows 95/98, и Windows NT — многозадачные системы и, естественно, мы не могли обойти вниманием этот важный вопрос и его поддержку в библиотеке классов MFC. Поэтому в *Части XII (главы 29—31)* подробно рассматриваются вопросы многозадачности и создания потоков.
- Одним из мощных средств современных языков программирования и операционных систем является встроенная обработка исключений. В *Части XIII (главы 32—36)* подробно рассматриваются обработка исключений в языке C++ и структурная обработка исключений, а также поддержка механизма обработки исключений в библиотеке классов MFC.
- Хотя часть материала предыдущего издания пришлось сократить, мы решили дополнить книгу информацией о базах данных и Internet. Каждая тема сама по себе чрезвычайно велика, интересна и не может быть рассмотрена в рамках одной книги. Поэтому материал, представленный в *Части XIV (главы 37—39)*, "идет от конкретного примера", раскрывая один из аспектов программирования для Internet — работа с базами данных и предоставление результатов пользователю.
- Примеры приложений иллюстрируют то, что можно найти на страницах книги. Сопроводительная дискета содержит полные исходные тексты и исполняемые файлы всех рассматриваемых примеров. Мы надеемся, что они будут полезны читателям.

В заключение хотелось бы сказать несколько добрых слов в адрес нашего редактора Татьяны Темкиной. То, что книга не только содержит полезный материал, но и, на наш взгляд, достаточно легко читается, во многом именно ее заслуга. Мы искренне признательны ей за это.

Все замечания и пожелания можно присылать по адресу:

- Издательство "БХВ — Санкт-Петербург", 199397, С.-Петербург, а/я 194
- E-mail: bhv@mail.nevalink.ru



Часть I

ОБЗОР VISUAL C++ 6

Глава 1

Знакомьтесь — Visual C++ 6



Новый 32-разрядный компилятор Microsoft Visual C++ 6.0 для операционных систем Windows (Windows 95/98 и Windows NT) — одна из составных частей мощного пакета компиляторов Developer Studio, в который входят новейшие версии таких известных программных продуктов, как Visual Basic, Visual J++, Visual FoxPro, Visual InterDev, Visual SourceSafe и других. Прежде чем переходить к деталям, окинем взглядом пакет в целом.

Что нового в этой версии?

Надо сказать, что изменений по сравнению с предыдущей (пятой) версией не так много. Основные изменения коснулись, как впрочем и ожидалось, работы с базами данных. Напомню, что основные отличия пятой версии Visual C++, в основном, касались программирования для Internet. Естественно, все лучшее, что было реализовано там, перешло и в новую версию. Следует сказать, что выпуск шестой версии произошел столь стремительно, что многие еще не успели как следует поработать с пятой. Поэтому коснемся и тех возможностей, которые появились в пятой версии.

Как мы только что отметили, Internet в этих версиях продукта играет едва ли не самую заметную роль. Все начинается непосредственно с установки. Одно из окон мастера установки специально проверяет, установлена ли на вашем компьютере версия Internet Explorer 4. И попробуйте отказаться от его установки. Не тут-то было — фирма Microsoft верна себе — если не хочешь пользоваться ее продуктами, не пользуйся ничем.

Примечание

Тех, кто работает в Windows NT 4, ждет еще один сюрприз — вы не сможете установить Visual C++, пока не установите как минимум Service Pack 3.

Поддержку Internet можно найти практически везде, включая IDE (Integrated Development Environment — интегрированная среда разработки). Ниже приведен обзор этих возможностей:

- ❑ *Классы сервера Internet.* Фирма Microsoft добавила пять новых классов, которые поддерживают ISAPI (Internet Server Application Programming Interface — Интерфейс программирования приложений сервера Internet). Эти классы помогают создавать фильтры сервера Internet и расширения в форме специальных DLL. Для облегчения работы предусмотрен специальный мастер ISAPI Extension Wizard (Мастер расширения ISAPI).
- ❑ *Win32 Internet.* Эти классы позволяют вам добавить поддержку Internet к любым приложениям; включает HTTP, FTP и сервисы протокола распространения электронной информации (gopher services).
- ❑ *Active X Controls Wizard (Мастер элементов управления Active X).* Эти элементы хорошо вам знакомы — фирма Microsoft произвела некоторые изменения, которые позволяют обычным элементам управления работать в Internet. Можно также создавать Active X Documents (документы Active X), которые предоставляют множество интерфейсов, что позволяет видеть документ внутри сетевого браузера.
- ❑ *ATL (Active Template Library, Библиотека активных шаблонов).* В Visual C++ 6.0 вы найдете обновленную версию ATL.
- ❑ *Cluster Resource Type Wizard (Мастер типа кластерных ресурсов).* Создает два проекта, отвечающих за создание динамически подключаемой библиотеки ресурса и динамически подключаемой библиотеки администратора кластеров (рис. 1.1).

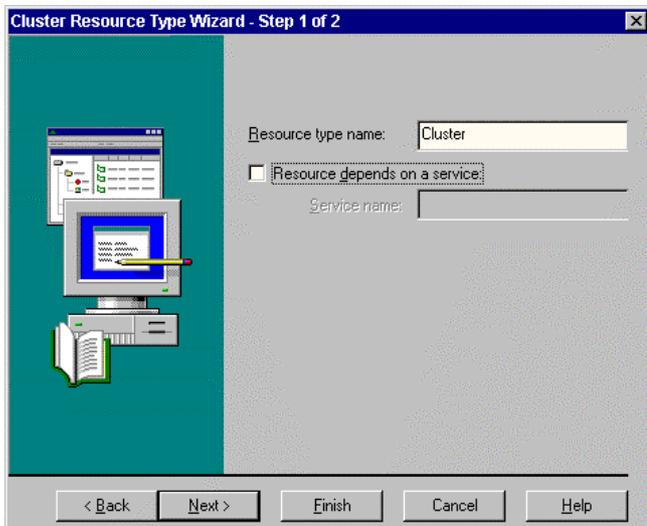


Рис. 1.1. Первое окно мастера Cluster Resource Type Wizard

Перечисленные изменения в основном относятся к пятой версии компилятора. Они охватывают ту область программирования, которая чем дальше, тем требуется все больше. Управление базами данных — это целый комплекс решений задач по программированию. Visual C++ 6.0 содержит новые возможности для проектирования баз данных, которые действительно нужно знать, а именно: что они собой представляют и как можно их использовать:

- ❑ *Multiithreading — ODBC (Многопоточный режим обработки)*. Теперь можно создавать множество исполняемых потоков для приложений ODBC. К несчастью, фирма Microsoft при этом не дает гарантии безопасности потоков.
- ❑ *Класс CDatabase* предоставляет улучшенную поддержку транзакций. *Два новых метода GetCursorComitBehavior и GetCursorRollbackBehavior* исключают необходимость сохранения курсора. *Третий новый метод OpenEx* теперь предпочтителен для инициализации объекта CDatabase перед тем, как вы будете создавать объекты CRecordset.
- ❑ *Класс CRecordset предлагает сразу несколько усовершенствований*. Например, теперь можно получить и установить закладки, используя компоненты GetBookmark и SetBookmark. (Эти два компонента относятся к классу CDBVariant.) Используя класс CRecordset, можно также производить выборку строк больших объемов. Компонент Open тоже предоставляет несколько новых опций. И, наконец, CRecordset больше не имеет чисто виртуальных функций. Это означает, что можно непосредственно создавать объект CRecordset.

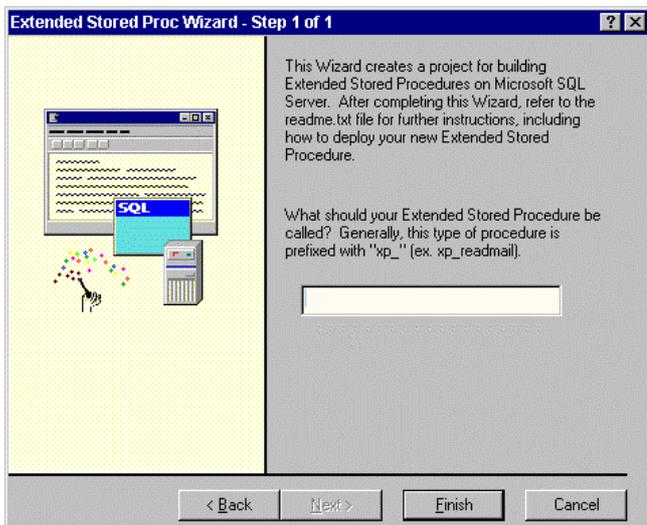


Рис. 1.2. Первое окно мастера Extended Stored Proc Wizard

- ❑ *Поддержка DAO*. Теперь MFC поддерживает версию DAO 3.5. Это та версия, которая поставляется с Microsoft Office 97.
- ❑ *OLE DB*. Это множество стандартных интерфейсов, которые позволяют совместно использовать данные из множества источников информации без написания отдельного кода для каждого из них. Интерфейсы предоставляют уровень поддержки, соответствующий источнику данных, который вы хотите использовать.
- ❑ *Extended Stored Proc Wizard (мастер расширенных сохраненных процедур)*. Использование этого мастера значительно облегчает создание расширенных сохраненных процедур для SQL Server (рис. 1.2).

Изменения коснулись и собственно языка программирования. Эти изменения помогают сохранить соответствие Visual C++ текущему стандарту:

- ❑ *bool, false, true (булевский, "ложь", "истина")*. Ключевое слово `bool` позволяет вам определить переменную со значением "истина", либо "ложь". Вы можете перекодировать эту переменную в число, при этом вы получите значение 0, если значение переменной ложно, или 1, если оно истинно. Ключевое слово `true` устанавливает переменную в значение "истина", тогда как ключевое слово `false` устанавливает ее в значение "ложь".
- ❑ *Explicit (формальный)*. Спроектированное для работы с объявлениями конструктора внутри класса, это ключевое слово делает невозможным использование конструктора в неявных преобразованиях.
- ❑ *Mutable (изменяемый)*. Если вы объявляете, что компонент данных является видоизменяемым, вы можете присвоить ему значение из постоянного метода.
- ❑ *Type name (имя типа)*. Это ключевое слово спроектировано для использования в определениях шаблонов. Оно сообщает компьютеру, что некоторый идентификатор является типом.
- ❑ *_declspec*. Это ключевое слово — новшество специфического языка фирмы Microsoft и используется для квалификации информации о классе хранения. Оно упрощает и стандартизирует специфические расширения Microsoft до языков C и C++. Существует множество различных атрибутов класса хранения, специфических для Microsoft, включая `allocate`, `thread`, `naked`, `dllimport`, `dllexport`, `nothrow`, и `selectany`.

Visual C++ предоставляет еще некоторые дополнительные обновления и изменения, которые нелегко отнести к какой-либо определенной области:

- ❑ *Поддержка механизма прокрутки с помощью мыши (Mouse Scroll Wheel Support)*. Класс `CScrollView` теперь обеспечивает поддержку механизма прокрутки для мыши. Эта поддержка позволяет пользователю прокручивать документ без использования полосы прокрутки. Пользователь запрашивает прокрутку с помощью события `WM_MOUSEWHEEL`.

- ❑ *Утилита ERRLOOK.* Эту служебную программу можно использовать, чтобы отыскивать ошибочные запросы системы, которые включают HRESULT OLE, по их значению. Формат значений может быть либо десятичным, либо шестнадцатеричным, кроме того, можно вставлять полученный текст в приложения, если это необходимо. Эта утилита включает контекстно-зависимую справку, которая показывает вам, как ее использовать.
- ❑ *Сервер транзакций Microsoft.* Он предоставляет все, что вам необходимо для написания приложений на основе транзакций. Microsoft Transaction Server поддерживает программирование баз данных через интерфейс ODBC.
- ❑ *Визуальные инструменты работы с базами данных.* Любой, кто работал с Microsoft Access, знает о времясберегающем аспекте использования инструментов визуального проектирования при создании приложений баз данных. Визуальные инструменты для разработки баз данных, предоставляемые Enterprise Edition Visual C++, включают как Query Designer, так и Database Designer. Query Designer помогает получить или обновить данные в базе данных путем использования операторов SQL. Database Designer помогает создать множество различных объектов баз данных. Его устанавливают для работы с Microsoft SQL Server версии 6.5 (или выше).
- ❑ *Data View Pane (подокно отображения данных).* Это специальная область просмотра (подокно) в окне Workspace (рабочее пространство), которое позволяет вам работать со множественными источниками данных ODBC. Вы можете использовать Data View Pane для выполнения множества различных задач, подобных добавлению или удалению источника данных из проекта, а также восстановлению локальной копии источника данных из исходной версии. Data View Pane также подходит для таких действий, как выполнение сохраненной процедуры или сброс одного из источников данных.
- ❑ *Debugger Performance (производительность отладчика).* Debugger Performance позволяет улучшить эксплуатационные характеристики во время отладки соединения баз данных.
- ❑ *Расширенная поддержка типов баз данных SQL.* Теперь можно просматривать все типы данных SQL, включая текст и образ.

Перечислив основные глобальные особенности Visual C++ 6.0, рассмотрим некоторые параметры компилятора.

Настройки проекта

Для того чтобы определить настройки проекта, в меню **Project** (проект) выберите пункт **Settings** (настройки). На экране появится блок диалога **Project Settings** (настройки проекта) (рис. 1.3).

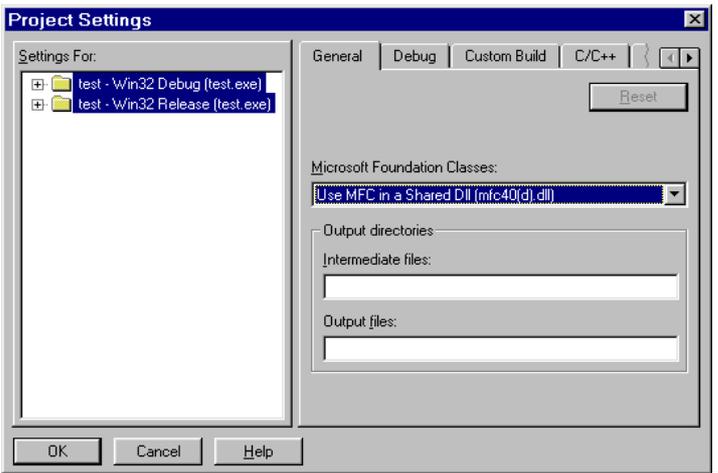


Рис. 1.3. Блок диалога **Project Settings**

Перейдите на вкладку с названием типа опций, которые вы хотите установить. Доступны вкладки **General** (основные), **Debug** (отладка), **C/C++** (настройки компилятора), **Link** (настройки компоновщика), **Resources** (ресурсы), **MIDL**, **Browser Info** (информация программы просмотра), **Custom Build** (обычное построение), **Pre-link step** (действия перед компоновкой) и **Post-build step** (действия после построения). Выбрав вкладки **C/C++** или **Link**, можно установить необходимые опции для, соответственно, компилятора и компоновщика.

Настройка опций компилятора

При выборе в раскрывающемся списке **Category** (категория) (рис. 1.4) категории опций, в блоке диалога будут отображаться опции, относящиеся к этой категории. Определены следующие категории опций:

- General** — наиболее часто используемые опции
- Language** — определяет представление указателей C++ на компоненты классов, управление обработкой исключений и управление созданием скрытых виртуальных конструкторов/деструкторов в классах, базирующихся на виртуальных
- Code Generation** — определяет тип процессора, библиотеку времени выполнения, соглашения о вызовах и выравнивание элементов структур
- Customize** — отключает расширения языка Microsoft, допускает компоновку на уровне функций, устраняет дублирование строк, допускает минимальное перепостроение проекта, пошаговую компиляцию, отключает сообщения при запуске

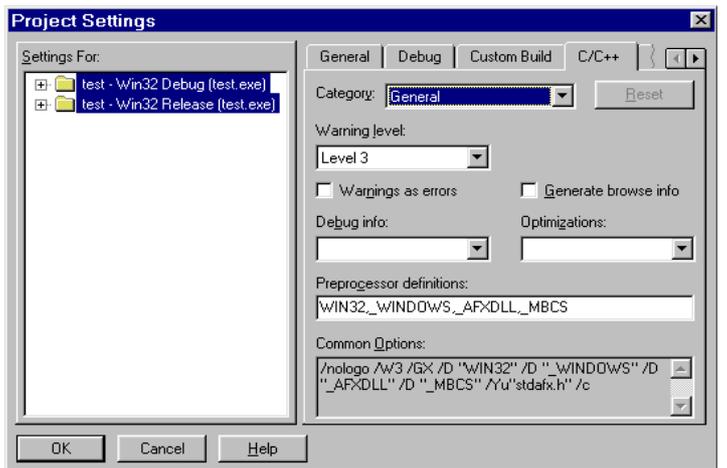


Рис. 1.4. Вкладка C/C++ блока диалога Project Settings

- **Listing Files** — создает файлы с информацией для просмотра (BSC) и файлы с листингом кода
- **Optimizations** — определяет, как компилятор настраивает скоростные показатели работы прикладной программы
- **Precompiled Headers** — позволяет ускорить время компиляции, разрешая прекомпиляцию кода C или C++
- **Preprocessor** — управляет символами, макросами и включает пути, используемые препроцессором C/C++

Ниже описана категория наиболее часто используемых опций — **General**:

- **Warning level** — устанавливает уровень выдачи предупреждений об ошибках (только для файлов с исходными текстами, не для объектных (OBJ) файлов)
 - **None** — отключает все предупреждения. Параметр командной строки /W0 или /w
 - **Level 1** — выводит только строгие предупреждения. Параметр командной строки /W1
 - **Level 2** — выводит менее строгие предупреждения, такие как использование функций без объявления типа возвращаемого значения, ошибка передачи возвращаемого значения для функции, тип возвращаемого значения которой не void, ошибки преобразования данных, которые могут привести к потере данных или точности. Параметр командной строки /W2
 - **Level 3** — выводит еще менее строгие предупреждения, такие как вызов функций до их объявления. Параметр командной строки /W3