

# Введение

Поздравляю с удачным приобретением! Благодаря этой книге вы сможете стать сведущим программистом на C++. В Visual Studio 2008 – новейшей системе разработки от Microsoft – поддерживаются две отдельных, но тесно связанных между собой разновидности языка C++, а именно – исходная версия C++, представляющая собой стандарт ISO/ANSI, а также новая версия C++ под названием C++/CLI, которая была разработана Microsoft и теперь утверждена в стандарте ECMA. Эти две версии C++ дополняют друг друга и выполняют совершенно разные роли. Версия ISO/ANSI C++ предназначена для написания высокопроизводительных приложений, способных функционировать на компьютере “родным” образом, в то время как версия C++/CLI была специально разработана для написания приложений, ориентированных на .NET Framework. Настоящая книга позволит вам научиться писать приложения на обеих этих версиях C++.

При написании программ на ISO/ANSI C++, конечно, довольно во многом будет помогать генерируемый автоматический код, но свой код на C++ все равно писать придется и немало. А это требует основательного понимания приемов объектно-ориентированного программирования вообще и программирования для Windows – в частности. Хотя язык C++/CLI и ориентирован на .NET Framework, он еще также является инструментом для программирования приложений Windows Forms, который позволяет разрабатывать такие приложения с написанием минимального объема кода, а в некоторых случаях даже и вообще без явного написания кода. Конечно, в тех случаях, когда в приложение Windows Forms все-таки требуется добавлять код, пусть даже и в небольших объемах, без глубокого знания языка C++/CLI никак не обойтись. Многие профессионалы продолжают отдавать предпочтение ISO/ANSI C++, но скорость разработки, которую дает C++/CLI и приложения Windows Forms, тоже заслуживает внимания. По этой причине в настоящей книге охватываются обе эти разновидности языка C++.

## Для кого предназначена эта книга

Целью этой книги является научить вас написанию приложений C++ для операционной системы Microsoft Windows с применением Visual C++ 2008 или любой версии Visual Studio 2008. Никаких предположений касательно исходного уровня знаний читателя в области программирования с помощью того или иного языка не делается. Поэтому данное пособие подходит для всех тех, кто:

- ❑ обладает хотя бы минимальным опытом программирования на каком-то другом языке, например, на языке BASIC, и хочет изучить язык C++ и приобрести практические навыки программирования под Microsoft Windows;
- ❑ имеет некоторый опыт работы на C или C++, но не в контексте Microsoft Windows, и хочет усовершенствовать свои навыки для получения возможности выполнять программирование в среде Windows с использованием новейших инструментов и технологий;
- ❑ знаком с C++ и желает научиться работать еще и с C++/CLI;

- является новичком в программировании, но серьезно намерен глубоко изучить C++, в случае чего, для успешного изучения изложенного в этой книги материала, должен иметь хотя бы грубое представление о том, как работает компьютер, и в частности, о том, каким образом функционирует память и как хранятся данные и команды.

## О чем эта книга

Целью, которую автор преследовал при написании этой книги, было научить читателя основами программирования на C++ с применением обеих поддерживаемых в Visual C++ 2008 технологий. Поэтому данная книга, по сути, и получилась детальным пособием по обоим разновидностям языка C++, а именно: по разработке “родных” приложений Windows на ISO/ANSI C++ с применением библиотеки Microsoft Foundation Classes (MFC) и по разработке приложений Windows на C++/CLI с использованием Windows Forms.

Из-за важности и широкого распространения в наши дни технологий баз данных в этой книге еще также приводится краткое описание основных приемов, которыми можно пользоваться для получения доступа к источникам данных как в приложениях MFC, так и в приложениях Windows Forms. Приложения MFC требуют написания большего количества кода, чем приложения Windows Forms. Объясняется это тем, что последние создаются в Visual C++ 2008 с помощью высокоразвитого средства визуального проектирования, которое позволяет компоновать весь графический интерфейс пользователя (GUI) для приложения в интерактивном визуальном режиме и генерирует весь необходимый для его создания код автоматически. Именно поэтому в данной книге программированию MFC и посвящено больше страниц, чем программированию Windows Forms.

## Как построена эта книга

Содержимое настоящей книги организовано следующим образом.

- В **главе 1** приводится описание базовых концепций, в которых необходимо обязательно разбираться для того, чтобы программировать на C++ как родные приложения Windows, так и приложения .NET Framework, а также описание ключевых понятий, которые используются в среде разработки Visual C++ 2008. В частности, здесь рассказывается о том, как нужно использовать возможности Visual C++ 2008 для создания различных видов приложений C++, которые будут рассматриваться в остальных главах.
- **Главы 2–9** посвящены изучению обеих версий языка C++. Содержимое всех этих глав имеет схожую структуру: в первой части каждой из них рассматриваются вопросы, касающиеся ISO/ANSI C++, а во второй — вопросы, касающиеся C++/CLI.
- В **главе 10** рассказывается о способах применения библиотеки STL (Standard Template Library — стандартная библиотека шаблонов), которая представляет собой мощный и обширный набор инструментов для манипулирования данными в родных программах на C++. Эта библиотека не предусматривает никаких предпочтений среди приложений, благодаря чему может применяться в огромном количестве самых разнообразных контекстов. Помимо этого, в главе 10 еще также рассказывается и о способах применения библиотеки STL/CLR, которая является новой в Visual C++2008 и предназначена специально для разработки приложений на C++/CLI.

- ❑ **Глава 11** посвящена описанию приемов, которыми можно пользоваться для осуществления поиска ошибок в программах на C++.
- ❑ В **главе 12** рассказывается о том, как структурируются приложения Microsoft Windows, а также описываются и демонстрируются наиболее важные элементы, встречающиеся в каждом приложении Windows. Здесь приводятся примеры создания элементарных Windows-приложений с использованием ISO/ANSI C++, Windows API и MFC, а также пример создания базового приложения Windows Forms на C++/CLI.
- ❑ В **главах 13–18** подробно описываются предоставляемые MFC возможности для создания графического пользовательского интерфейса (GUI) и способы использования эквивалентных средств в программе, ориентированной на .NET Framework. В частности, здесь рассказывается о том, как создавать и использовать типичные элементы управления для построения графического интерфейса приложения, и как обеспечивать обработку событий, которые будут возникать в результате взаимодействия пользователя с этим приложением. Кроме того, демонстрируется процесс разработки реального рабочего приложения на C++ и программы с почти точно таким же функциональными возможностями на C++/CLI. Помимо приемов по созданию графического пользовательского интерфейса, по мере разработки этих приложений также иллюстрируются и способы вывода на печать и сохранения на диске документов.
- ❑ В **главе 19** приводится основной материал, в котором необходимо разбираться для того, чтобы создавать с помощью MFC свои собственные библиотеки. В частности, здесь рассказывается о различных видах библиотек, которые можно создавать, и демонстрируются примеры выполнения этой задачи для приложения, разработанного в предыдущих шести главах.
- ❑ В **главе 20 и 21** рассматриваются возможные способы получения доступа к источникам данных из приложения MFC. В частности, сначала здесь рассказывается о том, как можно получать доступ к базе данных в режиме только для чтения, а затем описываются основные приемы программирования, которыми можно пользоваться для обновления базы данных с применением MFC. В приводимых примерах фигурирует база данных Northwind, которую можно загрузить из Internet, но описываемые приемы также вполне могут применяться и к какому-нибудь другому источнику данных.
- ❑ В **главе 22** на примере разработки конкретного приложения демонстрируются возможные способы использования Windows Forms и C++/CLI для создания, настройки и применения элементов управления Windows Forms. Постепенная разработка этого второго приложения на C++/CLI по ходу главы позволяет обрести необходимый практический опыт.
- ❑ В **главе 23** на основе материала, изложенного в главе 22, показывается, как работают предназначенные для доступа к данным элементы управления и как их можно настраивать. Помимо этого, здесь еще также рассказывается и о том, как можно создавать приложения для получения доступа к базе данных практически без самостоятельного написания кода.

В состав всех глав входит множество рабочих примеров, демонстрирующих описываемые в них приемы программирования. Каждая глава завершается перечнем ключевых моментов, а многие главы еще также включают в конце упражнения, позволяющие применить проверенные знания на практике и тем самым проверить себя. Правильные решения этих упражнений вместе с кодами всех примеров доступны для загрузки на Web-сайте издательства.

Для обучения языку C++ в качестве примеров используются консольные программы с простым вводом и выводом в командной строке. Такой подход позволяет изучить различные возможности C++, не вдаваясь в сложности, связанные с программированием графических пользовательских интерфейсов Windows. Программирование для Windows становится посильной задачей только после глубокого изучения самого языка программирования.

Дабы процесс обучения протекал как можно проще, можно сначала заняться изучением только приемов программирования на ISO/ANSI C++. В каждой из глав, посвященных языку C++ (в главах 2–9) в первую очередь рассматриваются именно конкретные аспекты ISO/ANSI C++, и только потом – аналогичные новые средства C++/CLI. Причина подобной организации материала состоит в том, что C++/CLI создавался в виде расширения стандартного языка ISO/ANSI C++, и потому его изучение обязательно требует предварительного изучения ISO/ANSI C++. Благодаря этому, сначала можно прочитать только касающиеся ISO/ANSI разделы в главах 2–21, пропустив следующие за ними разделы по C++/CLI, и тем самым разобраться сначала в способах разработки приложений Windows с помощью ISO/ANSI, не удерживая в голове информацию об особенностях обеих версий языка, а затем вернуться к изучению C++/CLI уже тогда, когда в ISO/ANSI C++ все будет понятно. Конечно, можно также и читать все подряд и тем самым постепенно пополнять свои знания по обеим версиям языка C++.

## Что необходимо для работы с этой книгой

Для изучения предлагаемого в этой книге материала требуется иметь одну из версий Visual Studio 2008 Standard Edition, Visual Studio 2008 Professional Edition или Visual Studio 2008 Team System. Версия Visual C++ Express 2008 не подходит, поскольку в ее состав не входит MFC.

Для использования Visual Studio 2008 необходимо иметь установленной одну из операционных систем Windows XP (x86 или x64) с пакетом обновлений Service Pack 2, Windows Server 2003 с пакетом обновлений Service Pack 1 или Windows Vista в любой из версий, кроме Starter Edition. Для установки любой из трех названных версий Visual Studio 2008 необходимо иметь процессор с тактовой частотой 1 ГГц, минимум 348 Мбайт памяти (в случае Windows Vista – минимум 768 Мбайт) и хотя бы 2,2 Гбайт свободного пространства на жестком диске. Для установки всей предоставляемой с продуктом документации MSDN потребуются еще дополнительные 1,8 Гбайт на диске.

Во всех приводимых в настоящей книге примерах, связанных с получением доступа к базам данных, фигурирует база данных Northwind Traders. Отыскать доступную для загрузки версию этой базы данных можно путем выполнения на сайте <http://msdn.microsoft.com> поиска ключевых слов Northwind Traders. Конечно, при желании все эти примеры можно переделать и под свою собственную базу данных.

Наиболее важным, однако, для извлечения максимальной пользы из этой книги, является желание учиться и серьезное намерение овладеть наиболее эффективным на сегодня средством для программирования Windows-приложений. Необходима полная готовность вводить и прорабатывать все примеры и выполнять все упражнения, которые предлагаются в этой книге. Это звучит немного “сложнее”, чем есть на самом деле, а в действительности читатель, скорее всего, удивится тем, сколько ему удастся достичь за довольно небольшой период времени. Главное помнить о том, что каждый, кто изучает приемы программирования, время от времени запутывается, но если не останавливаться, то постепенно все начнет проясняться, и в конечном итоге станет полностью понятным. Настоящая книга поможет начать не бояться самостоятельно экспериментировать, а затем, соответственно, стать успешным программистом на C++.

## Использование классической темы Windows Classic

Если вы работаете с Visual Studio 2008 в среде Windows Vista, то, видимо, отметили великолепный внешний вид пользовательского интерфейса. Прозрачность, которую обеспечивает интерфейс Aero Glass, на первый взгляд (да и на второй) выглядит потрясающе. После включения всех визуальных эффектов, которые возможны в Vista, вы можете не понять, как кто-то может возражать против такой замечательной рабочей среды. Но после нескольких часов разглядывания раскрывающихся окон и протирая усталые глаза, вполне может возникнуть желание настроить дисплей так, чтобы он меньше утомлял зрение. И что более важно, вы можете заметить существенное снижение производительности, т.к. все эти украшения существенно загружают процессор.

Украшательства — это хорошо, но после того проведения целого дня за написанием кода хочется чего-то привычного. Это одна из причин, по которой в настоящей книге для демонстрации окон Visual Studio 2008 используется тема Windows Classic. Другая причина — если вы все еще работаете в Windows XP, то красивые окна Vista мало что значат для вас. Тема Windows Classic имеется в обеих операционных системах и, несомненно, она более приятна для глаз, чем интерфейс Aero Glass.

Если вы работаете в Vista, я предлагаю опробовать различные имеющиеся темы и решить, подходят ли они вам. Но если вы захотите применить в Vista такую же тему, которую я использовал в своей книге, можете установить ее, выполнив следующие шаги.

1. Щелкните на рабочем столе правой кнопкой мыши и выберите в контекстном меню пункт Персонализация. Появится окно Персонализация, показанное на рис. 0.1. Из этого окна возможен доступ ко всем параметрам дисплея, которые нужны для получения представления Windows Classic.

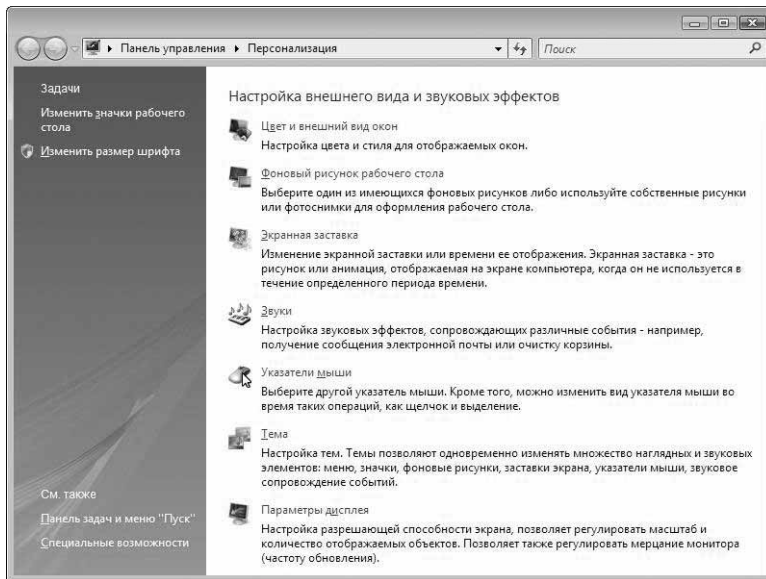


Рис. 0.1. Окно Персонализация

- Щелкните на пункте Тема – появится диалоговое окно Параметры темы, показанное на рис. 0.2.

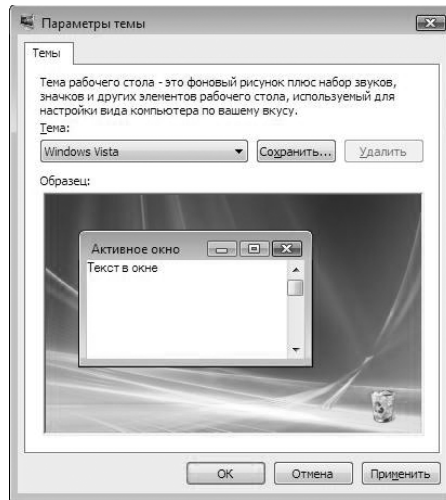


Рис. 0.2. Диалоговое окно Параметры темы

- В списке Тема выберите вариант Классическая и щелкните на кнопке ОК. После этого ваш дисплей станет очень похож на мой. Конечно, нужно еще разобраться со всеми спецэффектами. Так что следующие шаги будут посвящены им.
- Закройте окно Персонализация.
- Щелкните в панели управления на значке Система – появится окно, показанное на рис. 0.3.

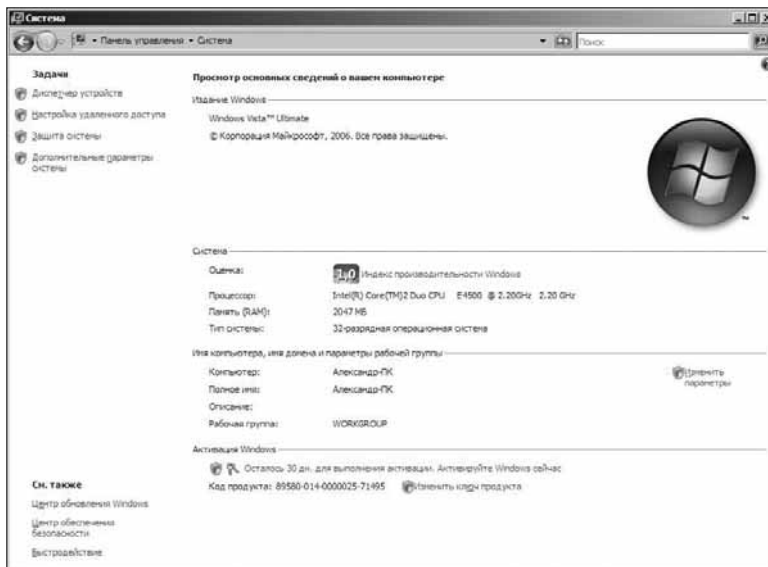


Рис. 0.3. Окно Система

6. Щелкните на ссылке **Дополнительные параметры системы** – появится диалоговое окно **Свойства системы**, раскрытое на вкладке **Дополнительно** (рис. 0.4).

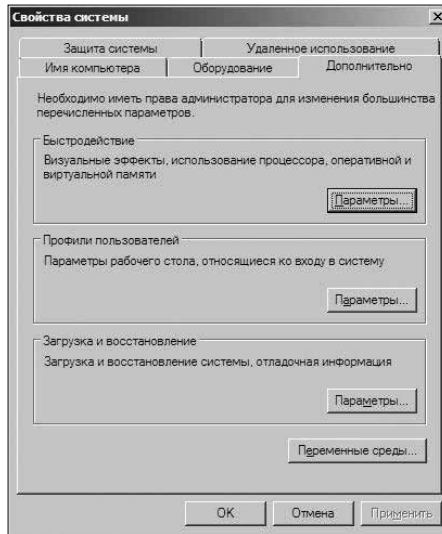


Рис. 0.4. Диалоговое окно **Свойства системы**

7. Щелкните на кнопке **Параметры** в разделе **Быстродействие** – появится диалоговое окно **Параметры быстродействия**, показанное на рис. 0.5.

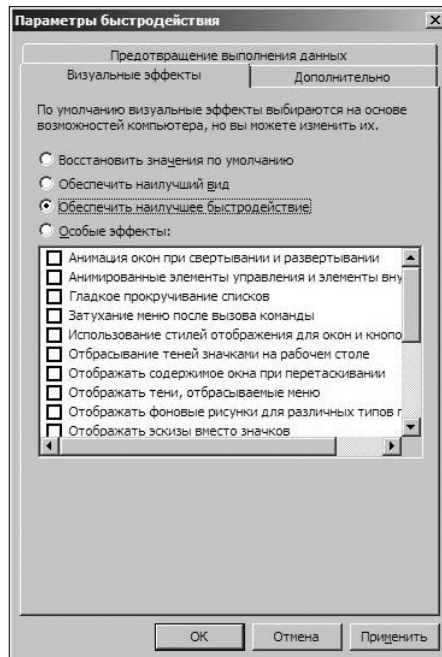


Рис. 0.5. Диалоговое окно **Параметры быстродействия**

8. Выберите вариант Обеспечить наилучшее быстродействие, как показано на рис. 0.5. (Если вы работаете с жидкокристаллическим монитором, то лучше все-таки установить флажок Сглаживать неровности экранных шрифтов. — *Прим. перев.*) Два раза щелкните на кнопках ОК, чтобы закрыть диалоговые окна. Теперь ваша система настроена на такое же представление, что и у меня, а также на максимальное графическое быстродействие.

Возможно, вы решите также привести к классическому виду меню Пуск, чтобы было легче находить нужные приложения и выполнять другие задачи. Для этого щелкните правой кнопкой на панели задач и выберите в контекстном меню пункт Свойства. Перейдите на вкладку Меню "Пуск". Выберите вариант Классическое меню "Пуск", щелкните на кнопке Настроить, чтобы произвести дополнительные настройки, а затем щелкните на кнопке ОК. Теперь ваша среда оптимально настроена для работы с Visual Studio.

Разумеется, если вы работаете в Windows XP и хотите, чтобы изображения на экране выглядели так же, как и в данной книге, можете также воспользоваться классической темой. Просто щелкните на рабочем столе Windows и в появившемся меню выберите пункт Свойства — появится диалоговое окно, показанное на рис. 0.6. На вкладке Темы диалогового окна Свойства: Экран выберите в списке вариант Классическая. Затем щелкните на кнопке ОК — вот и все.

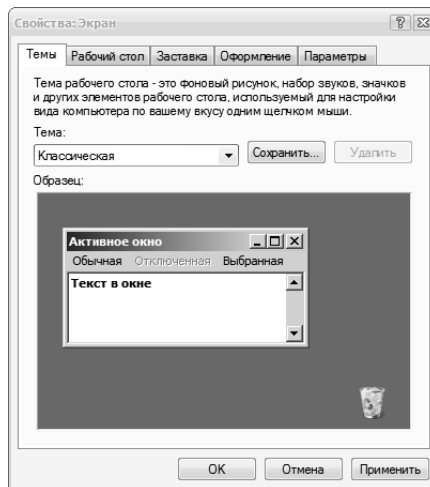


Рис. 0.6. Диалоговое окно Свойства: Экран

## Соглашения

Для предоставления возможности получения максимальной пользы из излагаемого материала и понимания о чем идет речь, везде в этой книге используются определенные соглашения.

### Практическое занятие

Под таким заголовком приводятся упражнения, содержащие рабочий пример, который требуется создать, скомпилировать и выполнить, следуя указанным в тексте инструкциям. Выходные данные рабочего примера выделяются моноширинным шрифтом:

Так выглядят выходные данные примера.



## Описание полученных результатов

После каждого раздела “Практическое занятие” идет раздел с таким заголовком, в котором приводится подробное описание введенного ранее кода.

В таких врезках содержится важная информация, которую следует запомнить и которая имеет непосредственное отношение к материалу, внутри которого она находится.

*Примечания, советы, подсказки, нестандартные трюки и прочие сведения, предлагаемые в качестве дополнения к основному тексту, идут отдельно и выделяются курсивом, как показано здесь.*

Ниже описаны стили текста книги.

- Новые термины и важные слова при первом упоминании выделяются **полуужирным**.
- Клавиатурные комбинации приводятся в таком виде: <Ctrl+A>.
- Имена файлов, URL-адреса и строки кода внутри текста выделяются следующим образом: `persistence.properties`.
- Код выделяется двумя разными способами:  
Моноширинным шрифтом, если этот код уже приводился ранее или был создан автоматически.  
**Моноширинным шрифтом с полуужирным начертанием, если код является новым или был изменен.**

## Исходные коды

Исходные коды примеров и упражнений вместе с решениями доступны для загрузки на Web-сайте издательства.

## От издательства

Вы, читатель этой книги, и есть главный ее критик и комментатор. Мы ценим ваше мнение и хотим знать, что было сделано нами правильно, что можно было сделать лучше и что еще вы хотели бы увидеть изданным нами. Нам интересно услышать и любые другие замечания, которые вам хотелось бы высказать в наш адрес.

Мы ждем ваших комментариев и надеемся на них. Вы можете прислать нам бумажное или электронное письмо, либо просто посетить наш Web-сервер и оставить свои замечания там. Одним словом, любым удобным для вас способом дайте нам знать, нравится или нет вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас.

Посылая письмо или сообщение, не забудьте указать название книги и ее авторов, а также ваш обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию последующих книг.

Наши координаты:

E-mail: [info@dialektika.com](mailto:info@dialektika.com)

WWW: <http://www.dialektika.com>

Информация для писем из:

России: 127055, г. Москва, ул. Лесная, д. 43, стр. 1

Украины: 03150, Киев, а/я 152