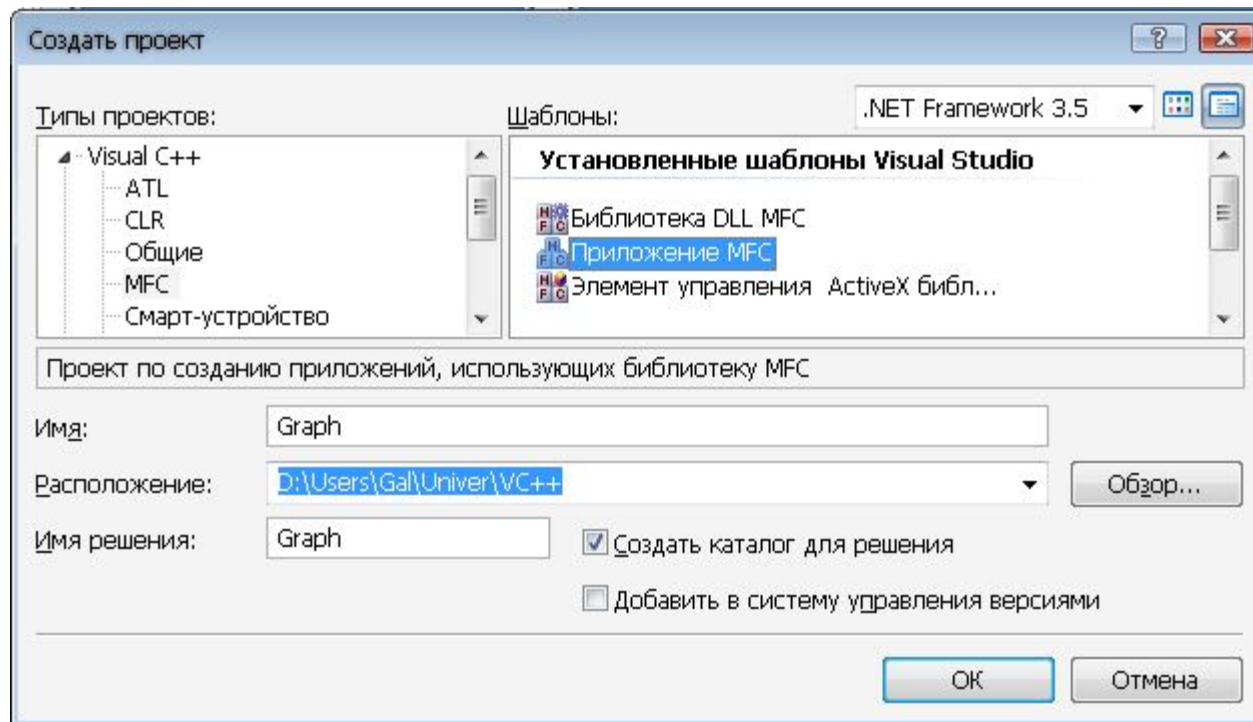


Создание пользовательских интерфейсов средствами **MFC**

- Пакет **Microsoft Foundation Classes** (MFC) — библиотека на языке C++, разработанная Microsoft и призванная облегчить разработку GUI-приложений (**Graphical User Interface**) для Microsoft Windows путем использования богатого набора библиотечных классов.

Создание проекта. Шаг 1



Создание проекта

В простейшем случае программа, написанная с помощью библиотеки MFC, содержит два класса, порождаемые от классов иерархии библиотеки: класс, предназначенный для создания приложения, и класс, предназначенный для создания окна.

```
class CTestGraphApp : public CWinApp
{
...
};
```

```
class CTestGraphDlg : public CDialog
{
};
```

Создание проекта. Шаг 2

Мастер приложений MFC - Graph

Тип приложения

Обзор

Тип приложения

Поддержка составных документов

Строки шаблона документов

Поддержка базы данных

Свойства интерфейса пользователя

Дополнительные параметры

Созданные классы

Тип приложения:

☐ Один документ

☐ Несколько документов

☒ На базе диалоговых окон

☐ Диалоговое окно HTML

☐ Несколько документов верхнего уровня

☒ Поддержка архитектуры Document/View

Стиль проекта:

☐ Проводник Windows

☒ Стандарт MFC

Использование MFC:

☐ Использовать MFC в общей DLL

☒ Использовать MFC в статической библиотеке

Связь с MFC DLL.

Язык ресурсов:

Русский

☒ Использовать библиотеки с поддержкой Юникода

< Назад

Далее >

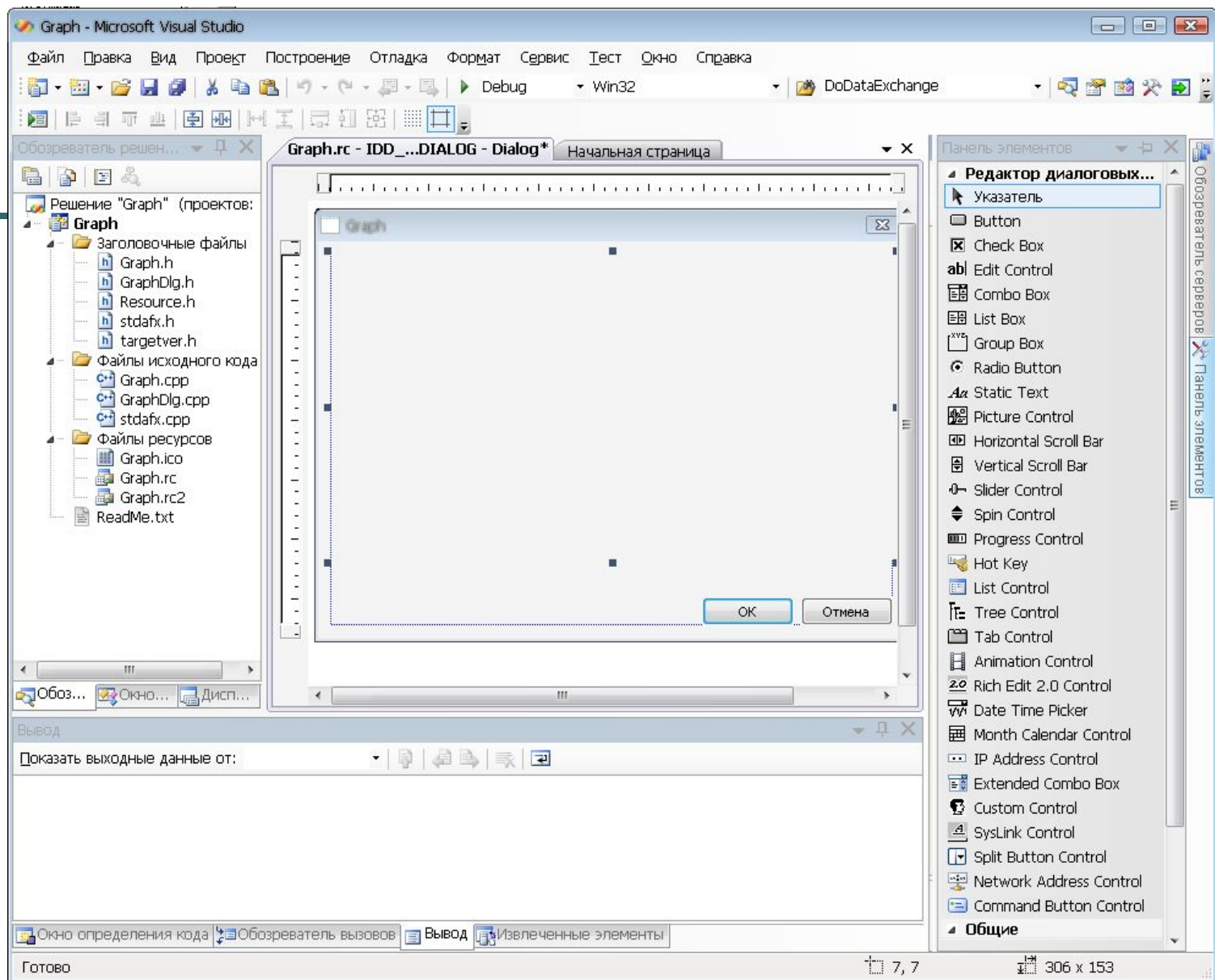
Готово

Отмена

Создание проекта. Шаг 3

- Помещаем на диалог элемент, в котором будет рисоваться график (н-р Static Text)
- В окне свойств задаем ему уникальный ID **IDC_GRAPH**
- Добавляем в класс **IDC_GRAPH** переменную типа CStatic **m_DrawArea**;
- Связываем переменную **m_DrawArea** и элемент **IDC_GRAPH**:

`DDX_Control(pDX, IDC_GRAPH, m_DrawArea);`
в методе `DoDataExchange`



Создание проекта. Шаг 4

Мастер добавления переменной-члена - Graph

Добро пожаловать в мастер добавления переменной-члена

Доступ:
public

Тип переменной:
CStatic

Имя переменной:
m_DrawArea

☒ Переменная элемента управления

Идентификатор элемента управления:
IDC_Graph

Тип элемента управления:
LTEXT

Минимальное значение:

Максимальное количество знаков:

Максимальное значение:

Файл .h:
...

Файл .cpp:
...

Комментарий (нотация // не требуется):

Готово Отмена

Создание проекта. Шаг 5

- Добавляем на диалоговое окно кнопку, при нажатии на которую будет происходить отрисовка графика
- Двойным щелчком по кнопке создаем соответствующий метод

Контекст устройств

Графический ввод-вывод в Windows унифицирован для работы с различными физическими устройствами. Для этого предусмотрен специальный объект, называемый контекстом устройства (Device context). Рисование на некотором абстрактном DC. Если DC связать с окном на экране, то рисование будет происходить в окне; если связать его с принтером – то на принтере; если с файлом – то, соответственно, в файл.

Класс **CClientDC** – разновидность контекстов устройств; позволяет выводить графику в рабочей области окна.

Для рисования в некоторой функции (н-р, обработчике события нажатия кнопки), нужно получить контекст устройства. Это делается так: `CClientDC dc(this);`

Отрисовка графика

-
- void CGraphDlg::OnBnClickedDraw()
- {
- *// TODO: добавьте свой код обработчика уведомлений*
- *//Создаем контекст, в котором будем рисовать*
- CClientDC dc(&m_DrawArea);
-
- *//Узнаем размеры прямоугольника*
- CRect rc; *//Графический объект*
- m_DrawArea.GetClientRect(&rc);
- int w = rc.Width();
- int h = rc.Height();
-
- int x_start = 10;
- int y_start = h-10;
-

//Отрисовка ...

*CPen pnPenBlack(PS_SOLID,1,RGB(0,0,0)); //Графический
//объект. Устанавливаем гр. объект в контекст устройства*

```
CPen * pOldPen = dc.SelectObject(&pnPenBlack);
dc.FillSolidRect(rc,RGB(255,255,255));
dc.MoveTo(x_start - 5,y_start);
dc.LineTo(x_start + w-15, y_start);
dc.MoveTo(x_start,y_start+5);
dc.LineTo(x_start, y_start-h+15);
CPen pnPenRed(PS_SOLID,1,RGB(255,0,0));
dc.SelectObject(&pnPenRed);
dc.MoveTo(x_start, y_start);
for(int i = 3; i < w-x_start-2; i+=3)
{
    dc.LineTo(x_start + i, y_start - int(h/3*(1 - sin((float)i))));
}
dc.SelectObject(pOldPen);
}
```

