

MFC Handout

Purpose: Handout to understand the basic concepts of MFC (Microsoft Foundation Classes)

Intended Audience: Students interested to learn advanced OOPS programming using RAD (Rapid Application Development) tools.

Prerequisites: C++, Visual Studio (VB, VC++)

MFC Overview:

The Microsoft Foundation Class Library (MFC) is an "application framework" for programming in Microsoft Windows. Written in C++, MFC provides much of the code necessary for managing windows, menus, and dialog boxes; performing basic input/output; storing collections of data objects; and so on. All you need to do is add your application-specific code into this framework.

Class Overview

MFC contains well over 200 classes. These can be divided into several categories. A number of classes are involved in creating a graphical application user interface, such as frames, views, menus, dialogs and dialog controls. Other categories include graphics drawing classes, file and socket classes, database access classes, thread support and synchronization classes, OLE (Object Linking and Embedding) support classes, Internet support classes, collection classes such as arrays, lists, and maps, and a large number of other support classes.

The CObject Base Class

Most MFC classes are subclasses of CObject. This class provides a number of important services. The first of these is object creation with a CRuntimeClass object. An instance of CRuntimeClass contains the method CreateObject() that uses a function pointer data member to create an instance of some CObject subclass that supports dynamic creation. The CObject subclasses contain a CRuntimeClass reference, so that the exact subclass of any object can be determined at run time.

Another important concept supported by CObject is serialization, the ability to store and retrieve instances of the class to and from external storage, such as files. This can be used to implement object persistence, OLE, and other useful mechanisms. Also, non-homogenous, type-safe collections of CObject subclasses are supported.

The Document/View Architecture

All graphical MFC applications have to be based on MFC's document/view structure. The application defines one or more document templates (CDocTemplate subclass) that contain CRuntimeClass references the following three classes:

- a document class (CDocument subclass),
- a frame class (CFrameWnd subclass),
- a view class (CView subclass).

The document class contains the internal representation of the application data. An instance is created by the framework for each new or opened document. The frame class describes the user interface of document windows of the application, typically multiple-document interface (MDI) frames. The view class shows a graphical representation of the document type. All classes as CObject subclasses.

A number of overridable methods in each of these classes allow the application to represent almost any kind of document. This basic structure is supported by a large number of support services, such as transparent printing and print preview support and OLE support.

Graphics Drawing Support

The graphics drawing support classes are modeled on the Win32 Graphics Device Interface (GDI). A number of convenience classes as well as overloaded constructors provide more flexible tools for drawing than the Win32 API, although the basic concepts are the same in MFC. Allocating and deallocating GDI resources and other error-prone tasks are simplified as the standard overridable methods in the CView class provide a default structure for implementing drawing accepting user input in a graphical application.

Dialog Support

MFC contains numerous classes for creating forms and other dialogs. The standard Windows controls are supported as well as the standard Windows common dialogs. Also, the Windows 95 new common controls and common dialogs are supported. MFC defines a standard dialog data exchange and dialog data validation (DDX/DDV) mechanism that provides a default structure for manipulating dialog data entered by the user. The record set and record view classes work with the database access classes, simplifying the display and editing of database rows presented on forms.

Other User Interface Features

The menu, toolbar, tabbed dialog, and status bar classes implement the latest Windows look and feel. The visual editing server and container classes, automation server and client classes, and other OLE support classes help implement object linking and embedding, the Windows standard data exchange between applications. The basic structure for an OLE support is present in these and the document/view classes, but a lot of work is left for the application code

MFC Hierarchy:

CObject*	MFC common base class
CGdiObject*	Graphical device interface class
CPen	Class used for drawing patterns and colors
CBrush	Class that represents fill patterns and colors
CFont	Font class
CBitmap	Bitmap class
CDC*	Device context base class
CClientDC	Client area device context class
CPaintDC	Painting area device context class
CMenu	Menu class
CCmdTarget*	Event-messages target base class
CWnd*	Window base class
CDialog	Dialog box window class
CStatic	Static control class(for text labels)
CButton	Button control class
CListBox	List box control class
CComboBox	Combo box control class
CEdit	Edit box control class
CFrameWnd	Frame window class
CWinThread*	Task thread base class
CWinApp	Windows application class

MFC Example:

Creating a MFC Project using VB.NET

Step 1: Select File → New → Project → MFC Application and press OK. As shown below