12,343,999 members (74,915 online)   hezhihuang ▼   **1.5K**   Sign out ⊗

home   **articles**   **quick answers**   **discussions**   **features**   **community**   **help**

CAtlBitmapButton - ATL/WTL Ownerdraw Superclassed Bitmap Button

Amit Dey, 12 Jun 2001

★★★☆☆   3.29 (6 votes)   Rate:

An owner drawn ATL/WTL bitmap button

<!-- Download Links -->

Download source files - 84 Kb

Introduction

Recently, in one of my projects, I needed to build a simple user interface consisting of a series of bitmap buttons in a dialog. Something simple and probably easy to use. About the same time, I came across **David Pizzolato's** very nice article on skinned button at codeproject.com, that got me thinking. What came out of the whole endevour was CAtlBitmapButton - an ATL/WTL ownerdrawn superclassed bitmap button. The class is not really complete and represents work in progress. I'll be glad if any of you find this useful 😊 .

The CAtlBitmapButton class is very friendly and you can learn to use it in no time. The hardest part might be drawing the bitmaps (if you are as artistically challenged as I am !).

Now let's get down to the basics. We'll be building an ATL/WTL Dialog-based application so I assume you are slightly familiar with ATL/WTL and ATL Windowing.
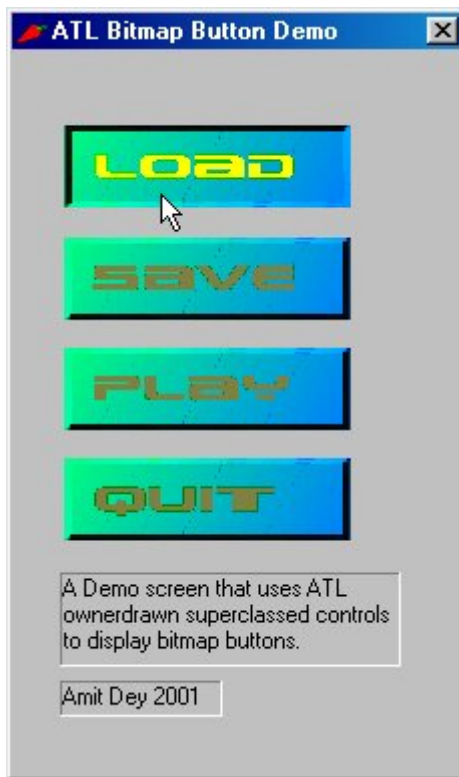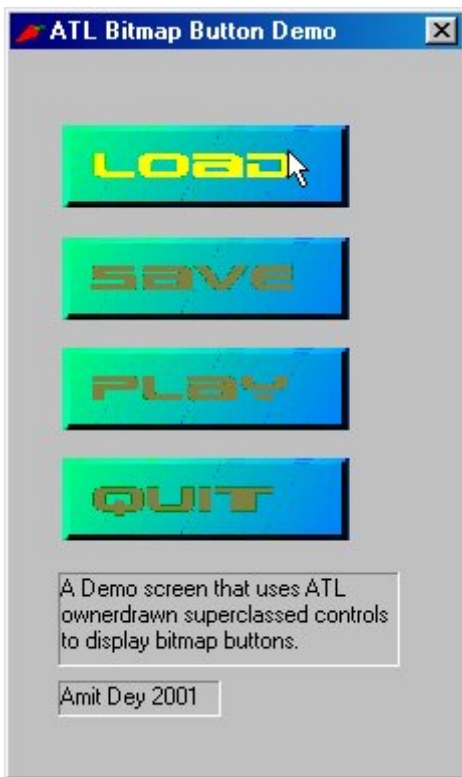
How to use

To build the client, fire up Visual C++ and create a new Win32 application . Next we shall rig up ATL support to the project. Since we'd like to have ATL Wizard support, just follow the instructions step-by-step. If you already know how to do this, you can skip this part. First, in project's stdafx.h file, replace

Hide　Copy Code

```
#include &ltwindows.h>
```

with

Hide　Copy Code

```
#define RICHEDIT_VER 0x0100
#include &ltatlbase.h>
extern CComModule _Module;
#include &ltatlcom.h>
#include &ltatlwin.h>
#include &ltatlapp.h>
#include &ltatlctrls.h>
```

Now add a new IDL file to the project that contains a blank library definition block like

Hide　Copy Code

```
library <Project Name>
{

};
```

Now, in the ClassView, right-click the IDL file you just added, and choose **Settings.** In the **General** tab of the project settings dialog, check the **Exclude file from build** option.

Next modify your projects .cpp file so that it looks like:

Hide　Copy Code

```
CComModule _Module;

BEGIN_OBJECT_MAP(ObjectMap)
END_OBJECT_MAP()

int APIENTRY WinMain(HINSTANCE  hInstance,
                     HINSTANCE  hPrevInstance,
                     LPSTR      lpCmdLine,
```

```
                        int      nCmdShow)
{
    // TODO: Place code here.
    _Module.Init(0, hInstance);


    _Module.Term();
    return 0;
}
```

Having rigged up ATL/WTL support, goto **Insert->New ATL Object**. In the **Miscellaneous** category, choose **Dialog** and click on **Next**.Enter the short name as **Dialog**.

In the dialog resource, add 4 buttons(IDC_BUTTON1,IDC_BUTTON2,IDC_BUTTON3 and IDC_BUTTON4) and set the **Ownerdraw** properties of these buttons to true. You would also need to add a few bitmaps to the project such that each button has three state bitmaps (Selected, Down and Over).

Add the file, *CAtlBitmapButton.h* to the project. In ClassView, right click the dialog class and add four member variables of type CAtlBitmapButton to it like

Hide Copy Code

```
CAtlBitmapButton m_button1,m_button2,m_button3,m_button4;
```

In the dialog's OnInitDialog(), add the following code :

Hide Copy Code

```
m_button1.SubclassWindow(GetDlgItem(IDC_BUTTON1));
m_button1.LoadStateBitmaps(IDB_LOADU, IDB_LOADD, IDB_LOAD);

m_button2.SubclassWindow(GetDlgItem(IDC_BUTTON2));
m_button2.LoadStateBitmaps(IDB_PLAYU, IDB_PLAYD, IDB_PLAY);

m_button3.SubclassWindow(GetDlgItem(IDC_BUTTON3));
m_button3.LoadStateBitmaps(IDB_SAVEU, IDB_SAVED, IDB_SAVE);

m_button4.SubclassWindow(GetDlgItem(ID_BUTTON4));
m_button4.LoadStateBitmaps(IDB_QUITU, IDB_QUITD, IDB_QUIT);
```

CAtlBitmapButton has a method LoadStateBitmaps() to load the state bitmaps. The last thing to do is to add the ATL macro REFLECT_NOTIFICATIONS() to the dialog's message map like:

Hide Copy Code

```
BEGIN_MSG_MAP(CDialog)
MESSAGE_HANDLER(WM_INITDIALOG, OnInitDialog)
COMMAND_ID_HANDLER(IDOK, OnOK)
COMMAND_ID_HANDLER(IDCANCEL, OnCancel)
COMMAND_ID_HANDLER(ID_QUIT, OnQuit)
REFLECT_NOTIFICATIONS()
END_MSG_MAP()
```

Build the project and run it. Check that the buttons are displaying the correct state bitmap. To handle button-clicks. use ATL macro COMMAND_ID_HANDLER() in the message map as shown in above code for the OK and Cancel button. OnCancel looks like:

Hide Copy Code

```
LRESULT OnCancel(WORD wNotifyCode, WORD wID, HWND hWndCtl, BOOL& bHandled)
{
    EndDialog(wID);
    return 0;
}
```

That's it. Yippee!
Have fun.

Acknowledgements

David Pizzolato.

License

This article has no explicit license attached to it but may contain usage terms in the article text or the download files themselves. If in doubt please contact the author via the discussion board below.

A list of licenses authors might use can be found here

Share

EMAIL

About the Author

Amit Dey
Web Developer
India 🇮🇳

Amit Dey is a freelance programmer from Bangalore,India. Chiefly programming VC++/MFC, ATL/COM and PocketPC and Palm platforms. Apart from programming and CP, he is a self-taught guitar and keyboard player.

He can be contacted at visualcdev@hotmail.com

You may also be interested in...

Reach: Intel® Edison based device is making highly precise GPS affordable for everyone

Looking Back: One Year of Microsoft Edge

A Lap Around @ChakraCore

Getting to Know the Arduino 101 Platform

Visual COBOL New Release: Small point. Big deal

Non-Tech Factors to Consider When Choosing Your Tech Stack

Comments and Discussions

Add a Comment or Question   ⊘

Search Comments [   ] Go

First   Prev   Next

Need Help 📌

Why not use the CBitmapButton class? 📌

Last Visit: 22-Jun-16 16:55   Last Update: 22-Jun-16 17:17

Refresh   1

📄 General   📰 News   💡 Suggestion   ❓ Question   🐞 Bug   📝 Answer   😄 Joke   👍 Praise   😠 Rant   ℹ️ Admin

Use Ctrl+Left/Right to switch messages, Ctrl+Up/Down to switch threads, Ctrl+Shift+Left/Right to switch pages.

Permalink | Advertise | Privacy | Terms of Use | Mobile   Layout: fixed | Article Copyright 2001 by Amit Dey
Web02 | 2.8.160621.1 | Last Updated 12 Jun 2001   fluid   Everything else Copyright © CodeProject, 1999-2016