

Persistent Frames

A collection of classes that make MFC SDI and MDI applications to remember the positions and sizes of their main frame and child frame windows.

This article was contributed by **Stefan Chekanov**.

Platform: VC 5.0 & VC 6.0

- [Introduction](#)
- [How to use Persistent Frames classes](#)
- [How do Persistent Frames work?](#)

Introduction

A usual MFC SDI application has main frame window and MFC MDI application has a main frame and additional child frame windows. The classes presented here make these frame windows to remember their last positions on the screen, their sizes and their minimized/maximized states. So, next time the application starts, its frame windows will restore their positions, sizes and minimized/maximized states.

Features:

- Save / Restore of the position, size and minimized/maximized state of the frame window
- Save/Restore of the position of docked control bars
- Information is stored in the application's registry or .INI file - `CWinApp::WriteProfileString()` and `CWinApp::WriteProfileInt()` methods are used
- Each window can have its own set of properties saved in its own registry/.INI file section
- If the user changes the screen resolution, the next time he/she starts the application the frames are resized to fit in new screen size.

How to use Persistent Frames classes

You should include the files `PersistFrameImpl.cpp`, `PersistFrameImpl.h`, `PersistMDIChildFrame.cpp`, `PersistMDIChildFrame.h`, `PersistMDIFrame.cpp`, `PersistMDIFrame.h`, `PersistSDIFrame.cpp`, `PersistSDIFrame.h` and `PersistFrames.h` in your project.

I usually include this line in my `StdAfx.h` file.

```
#include "PersistFrames.h"
```

to make all classes available everywhere in the project.

To make your frame window persistent just follow these steps:

1. Inherit your frame window class from the appropriate [Persistent Frames base class](#). I usually let MFC Application or Class Wizard to make the class and then using the feature Find&Replace replace all occurrences of the base class in both `.h` and `.cpp` files.
2. In the constructor of your frame window class call `SetProfileHeading(LPCTSTR szHeading)` and specify the registry section that will be used to store this frame window properties. By default this is set to `Window size!`
3. In the constructor of your frame window class call `SetManageBarStates(bool bManage)` and specify whether positions of control bars should be saved/restored too. By default this is set to false.

Depending on the type of your frame window you should inherit it from different classes:

Persistent Frames base classes	
CPersistSDIFrame	A base class for SDI frame windows.
CPersistMDIFrame	A base class for MDI main frame windows.
CPersistMDIChildFrame	A base class for MDI child frame windows.

That's it. Now compile and run your application.

The accompanying demo .zip file contains a VC 6.0 workspace with two projects - the one is MFC SDI and the other is MFC MDI application.

How do Persistent Frames work?

There is a class called **CPersistFrameImpl** which is used in all of the other classes. Its main methods are **Load()** which loads the properties from the registry and **Save()** which saves the properties into the registry. Each of **CPersistSDIFrame**, **CPersistMDIFrame** and **CPersistMDIChildFrame** have an object of **CPersistFrameImpl** and call its **Load()** and **Save()** where appropriate.

That's all.