

XPAC Migration

- **How to update a program to XP-8x3x-CE6**

How to update a program to XP-8x3x-CE6

Summary for updating a program to XP-8X3X-CE6

- The platform development SDK package installed on PC for developing the program for XP-8x3x-CE6 must be **version 1.4.5 or later**.
- The PACSDK.dll/PACSDK.lib used for XP-8x3x-CE6 must be **version 4.4.0.1 or later**.
- The VC program must be re-built with PACSDK.lib by Visual Studio.
- The C#/VB program aren't re-built by Visual Studio.

Platform Comparison

CPU	LX800/Atom	R3600
Models	XP-8x4x/ XP-8x4x-Atom	XP-8x3x
OS Version	WinCE 6.0	WinCE 6.0
Development Tools	VS2005/VS2008	VS2005/VS2008
Embedded SQL Server	3.5	3.5
.Net Compact Framework	3.5	3.5

Development Tool Requirements

Ensure that the necessary tools and the corresponding PAC SDKs are installed in your computer. The table below provides a summary of the supported development tools and languages required for developing PAC applications.

Development Tool		Language	Visual Basic .Net	Visual C#	Visual C++
Visual Studio 2005 or earlier	Any version except Professional		--	--	--
	Professional		√	√	√
Visual Studio 2008	Any version except Professional		--	--	--
	Professional		√	√	√
Visual Studio 2012 or later	Any version except Professional		--	--	--
	Professional		--	--	--

√: Supported, --: Unsupported



Visual Studio 2005/2008 Professional

Visual Studio takes full advantage of the .NET Compact Framework, which uses public Internet standards to enable integration with new and existing applications running on any platform. Supported languages include Visual C#, Visual C++ and Visual Basic .NET.

If you have an MSDN subscription, VS2005/VS2008 Pro can be downloaded from <https://msdn.microsoft.com/en-us/subscriptions/securedownloads/hh442898> Otherwise, you will need to purchase the software from your regular distributor.

Installation Procedure for VS2005

Step 1: Download the necessary installation files and updates

1. Visual Studio 2005 Professional
<https://www.dreamspark.com/Product/Product.aspx?productid=1>
2. Visual Studio 2005 Service Pack 1
<http://www.microsoft.com/en-us/download/details.aspx?id=10986>

Installation Procedure for VS2008

Step 1: Download the necessary installation files and updates

1. Visual Studio 2008 Professional
<https://www.dreamspark.com/Product/Product.aspx?productid=1>
2. Visual Studio 2008 Service Pack 1
<http://www.microsoft.com/en-us/download/details.aspx?id=10986>

Step 2: Install the platform development SDK package

The PACSDK_CE_X.X.X_VS2008 msi file contains the C header files, the C libraries and the software development kit used to develop applications for the PAC series on the PC.

The SDK package can be found in the CD:\SDK\PlatformSDK\ folder on the companion CD that was provided with the package, or the latest version can be downloaded from the ICP DAS web site.

<http://ftp.icpdas.com/pub/cd/xp-8x3x-ce6/sdk/platformsdk/>

File name: PACSDK_CE_X.X.X_VS2008.msi, where X.X.X is the released version of the platform sdk. **The platform sdk of version 1.4.5 or later to support XP-8X3X-CE6.**

Step 3: Execute the PACSDK_CE_X.X.X_VS2008.msi file (for VS2008) or PACSDK_CE_X.X.X_VS2005.msi file (for VS2005)

Step 4: Follow the prompts until the installation process is complete

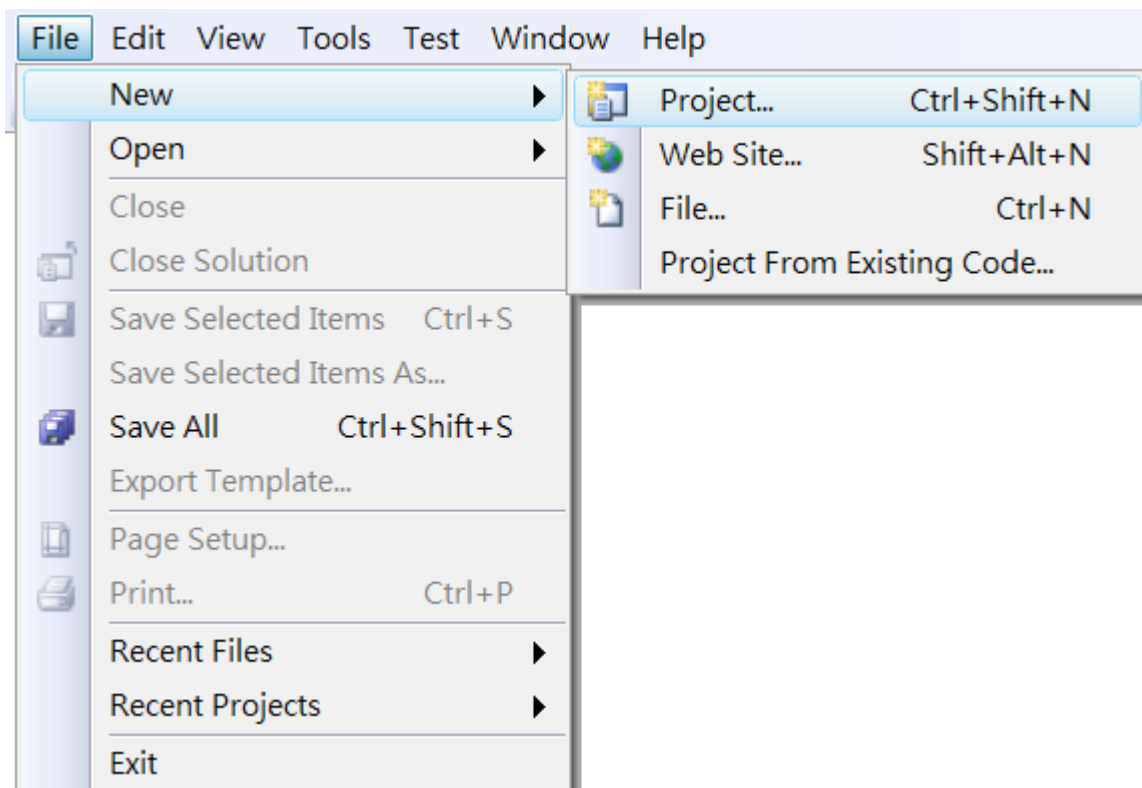
Creating a new project

The Visual C++ project template is a composite control that is used in this example to create a new project.

Step 1: Start Visual Studio 2008



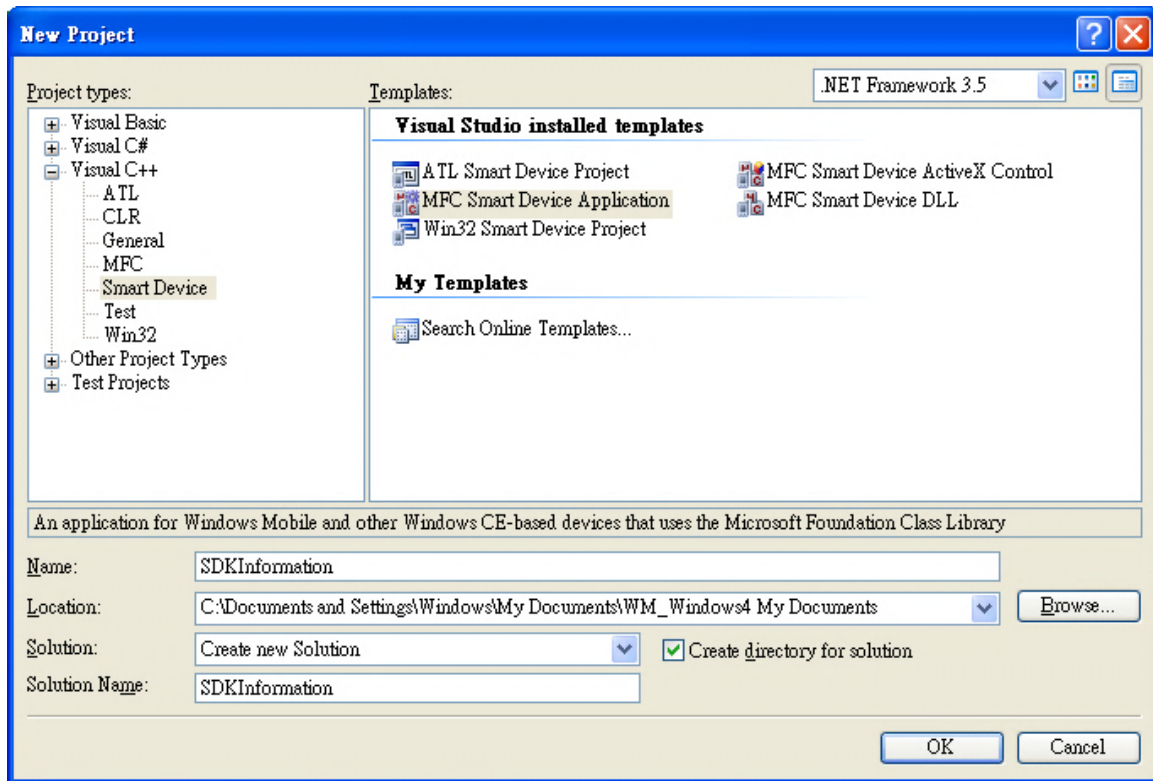
Step 2: On the File menu, point to New, and then click Project to open the New Project dialog



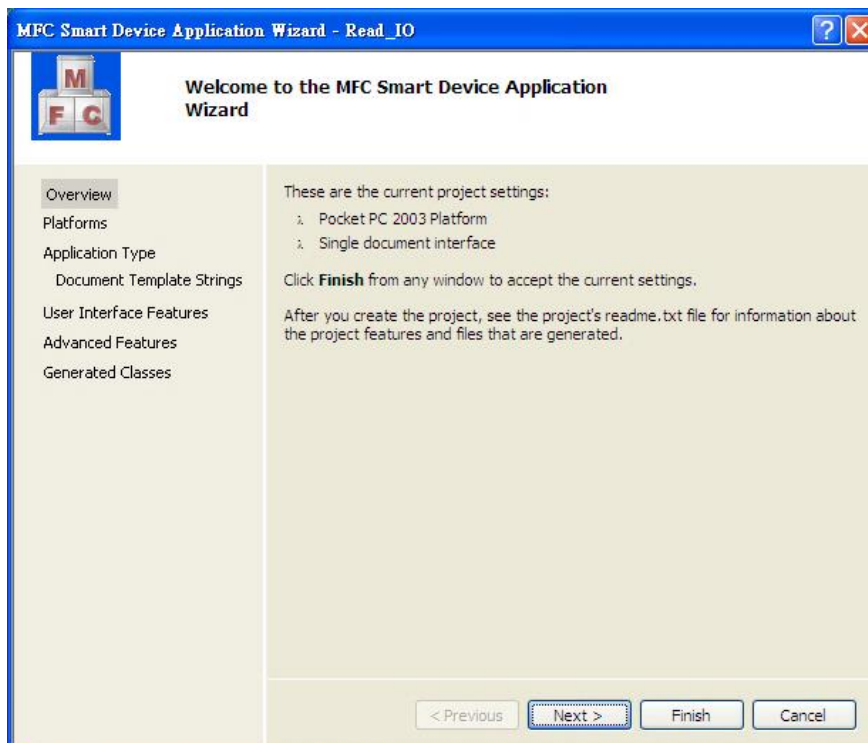
Step 3: In the Project types pane, expand the Visual C++ node and then select Smart Device

Step 4: In the list of Templates, select the MFC Smart Device Application item

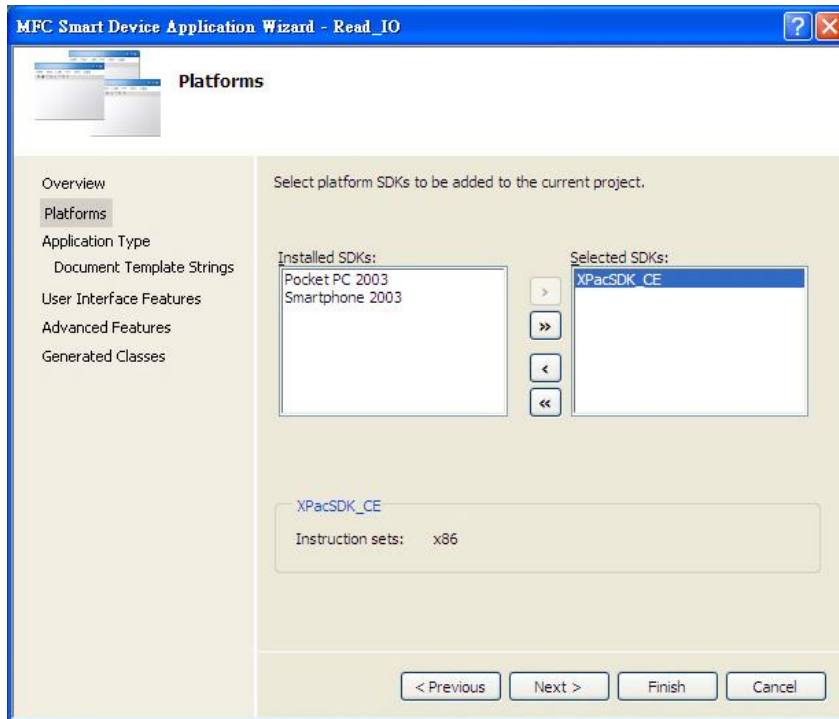
Step 5: Specify a name and a location for the application, and then click OK



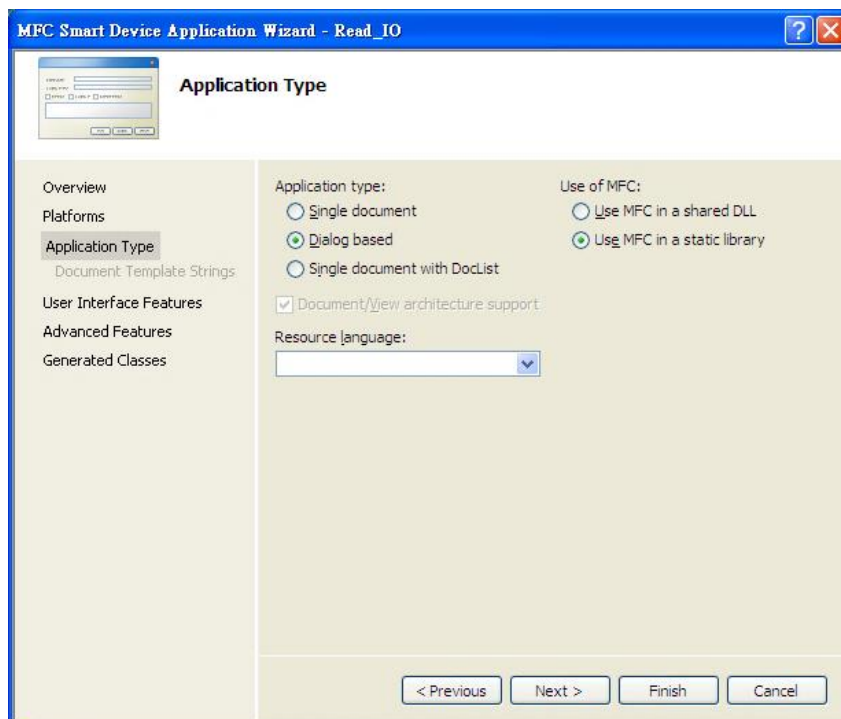
Step 6: When the following windows appears, click Next



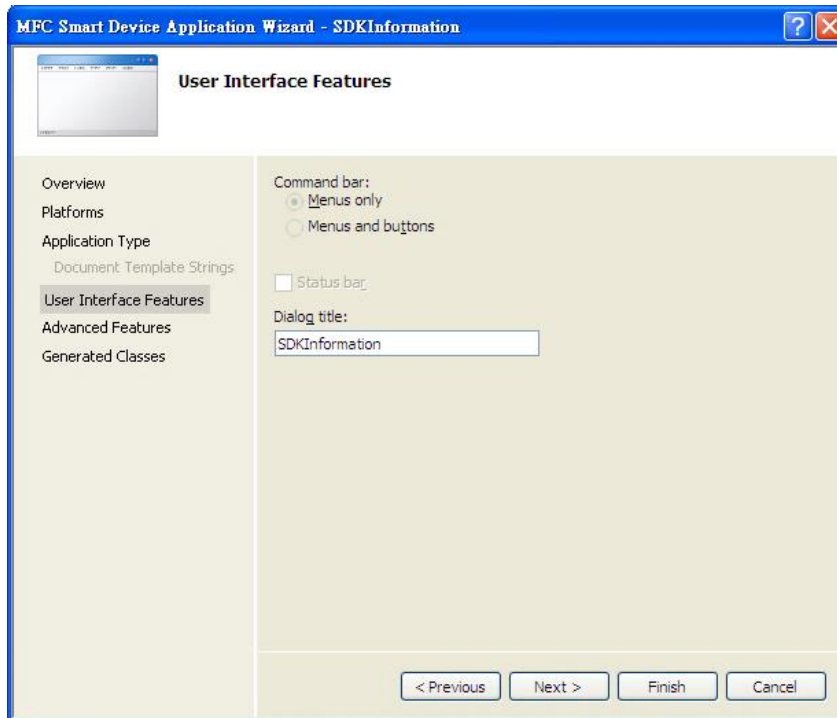
Step 7: On the Platforms screen, select the platform to be added to the project, XPacSDK_CE and then click Next



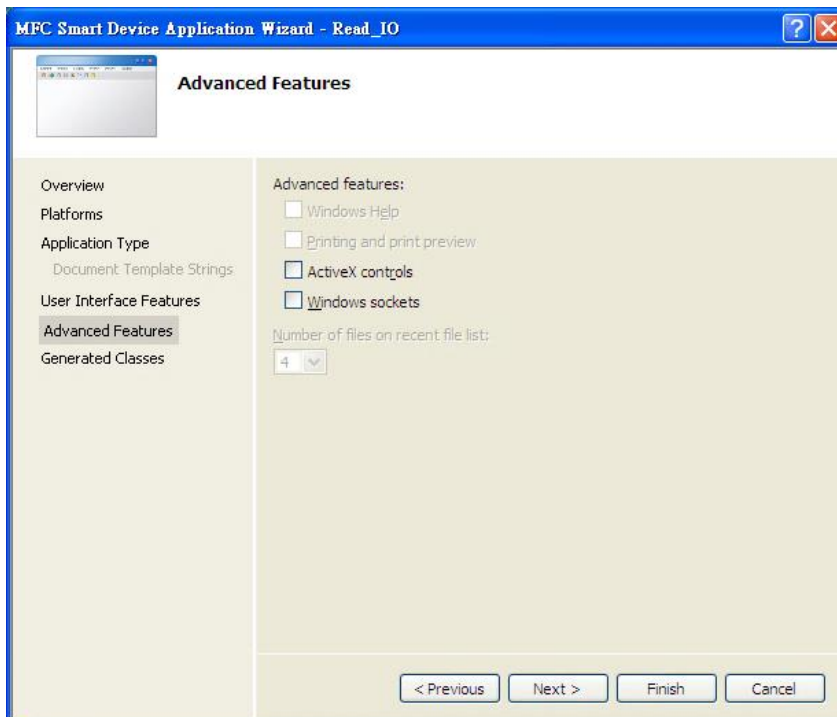
Step 8: On the Application Type screen, select the Dialog based option, and then click Next



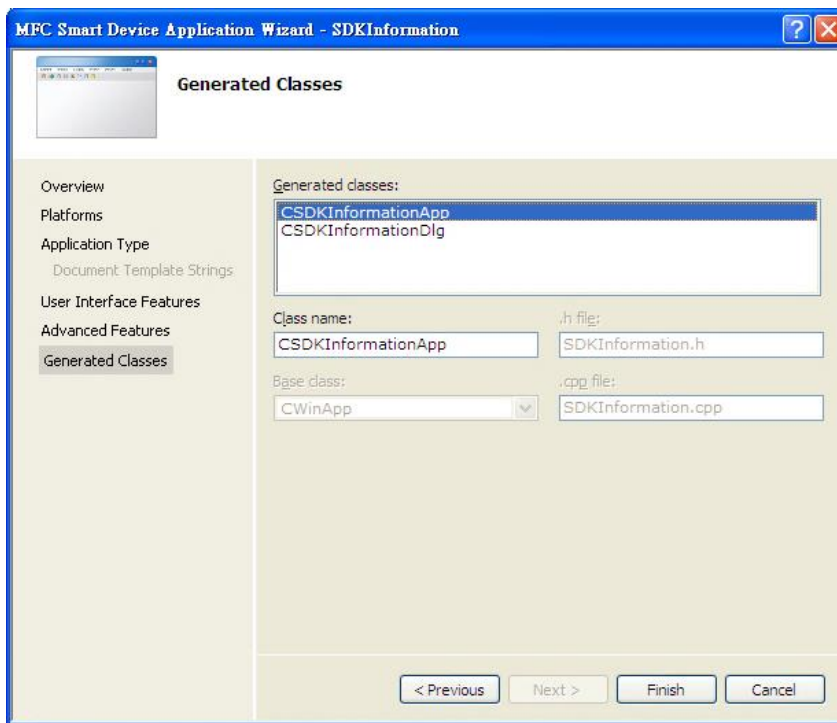
Step 9: Click Next on the User Interface Features screen



Step 10: Click Next on the Advanced Features screen



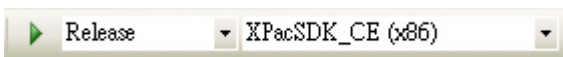
Step 11: Click Finish on the Generated Classes screen



Configuring the Platform

When developing applications using Visual C++, the Platform must be configured to indicate which platform and device the application will be downloaded to. Before deploying the project, ensure that the platform is corrected.

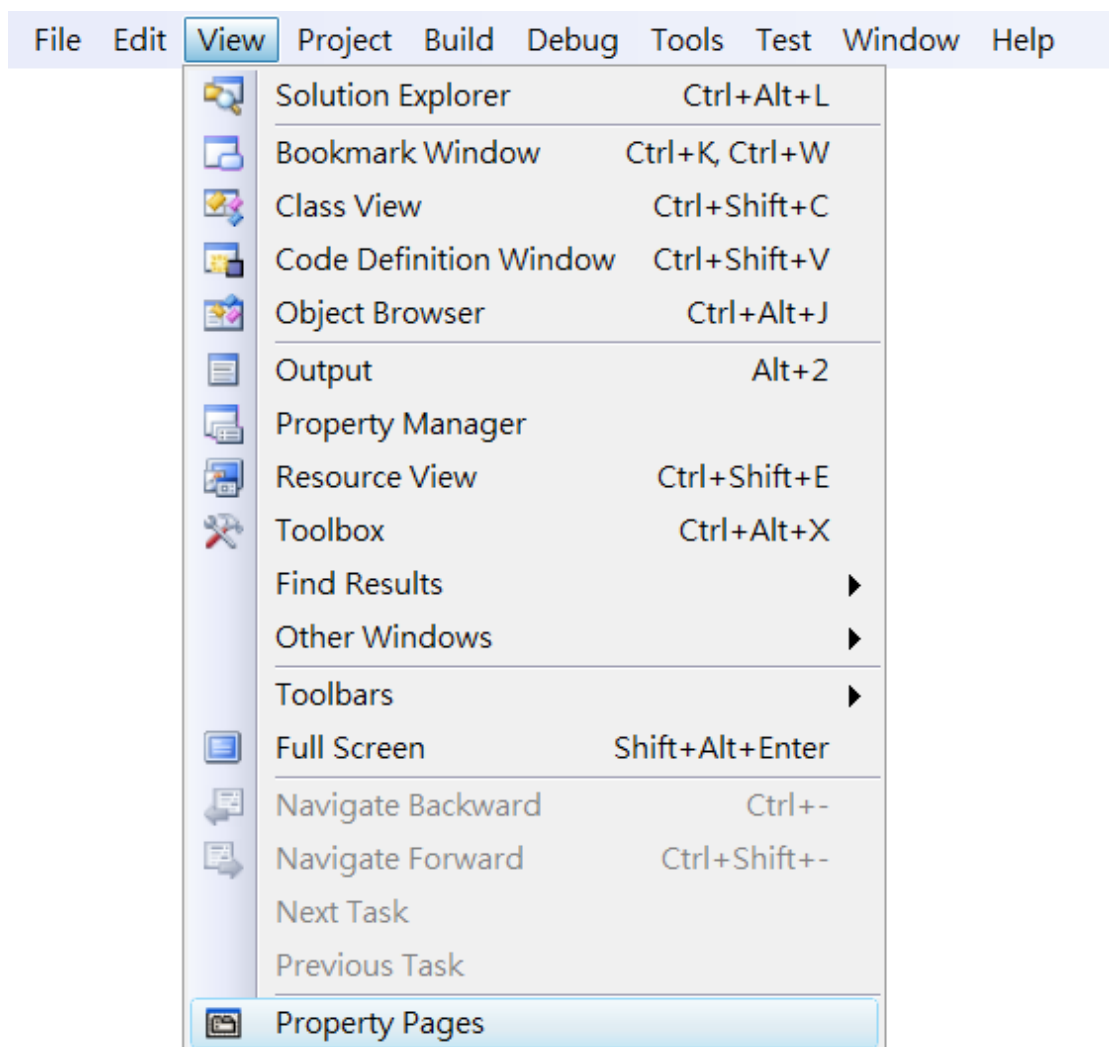
On the Debug configuration toolbar, select the Release option and then select XPacSDK_CE (x86), as illustrated below.



Specifying the PAC SDK Libraries

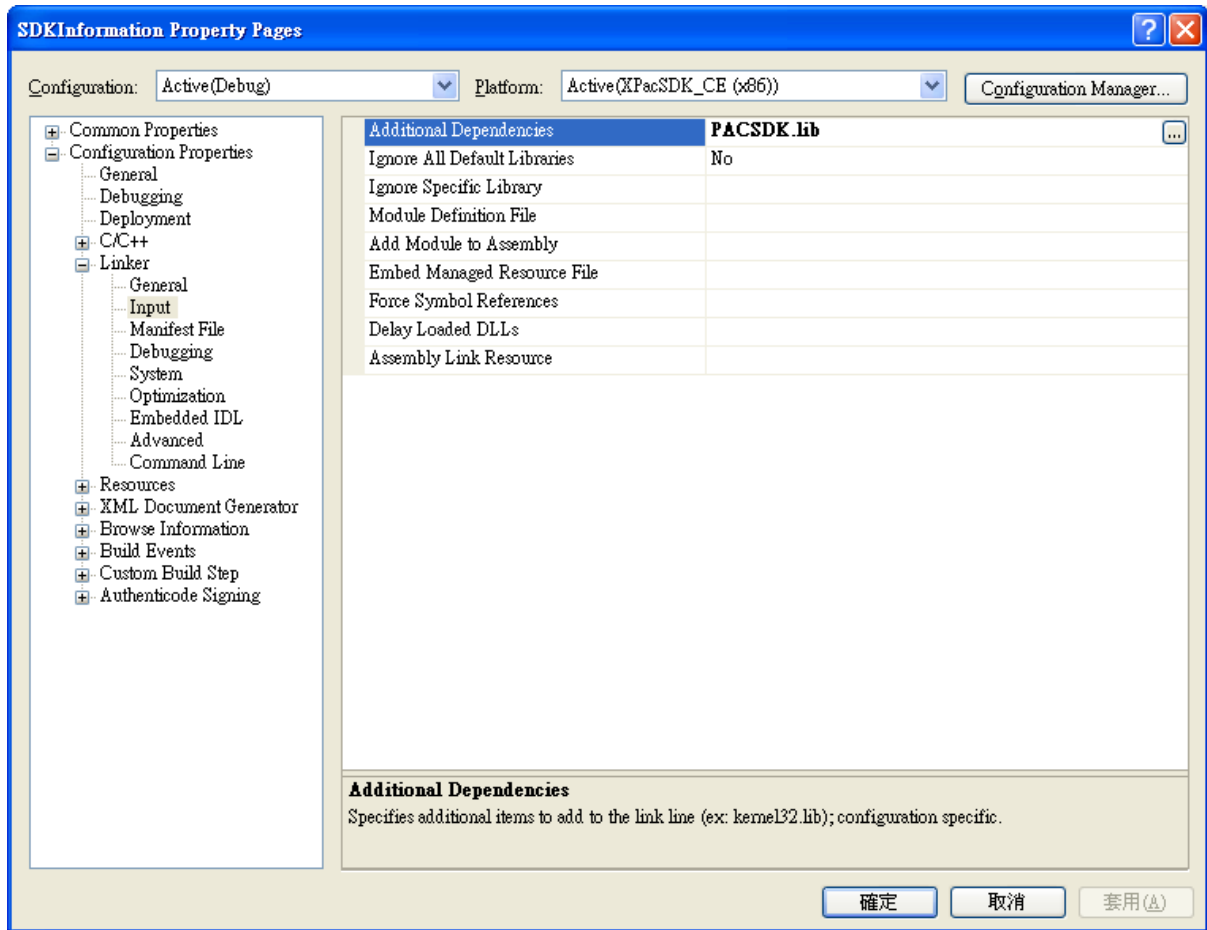
The PACSDK libraries for XP-8000-CE6 series module are provided in the PACSDK, and are compatible with C++. In order to use a component in an VC application, It is must to link PACSDK.lib in the project first.

Step 1: On the View menu, click the Property Pages option



Step 2: In left pane of the Property Pages dialog, expand the Linker node, and then click Input

Step 3: In the right pane of the Property Pages dialog, enter PACSDK.lib in the Additional Dependencies item



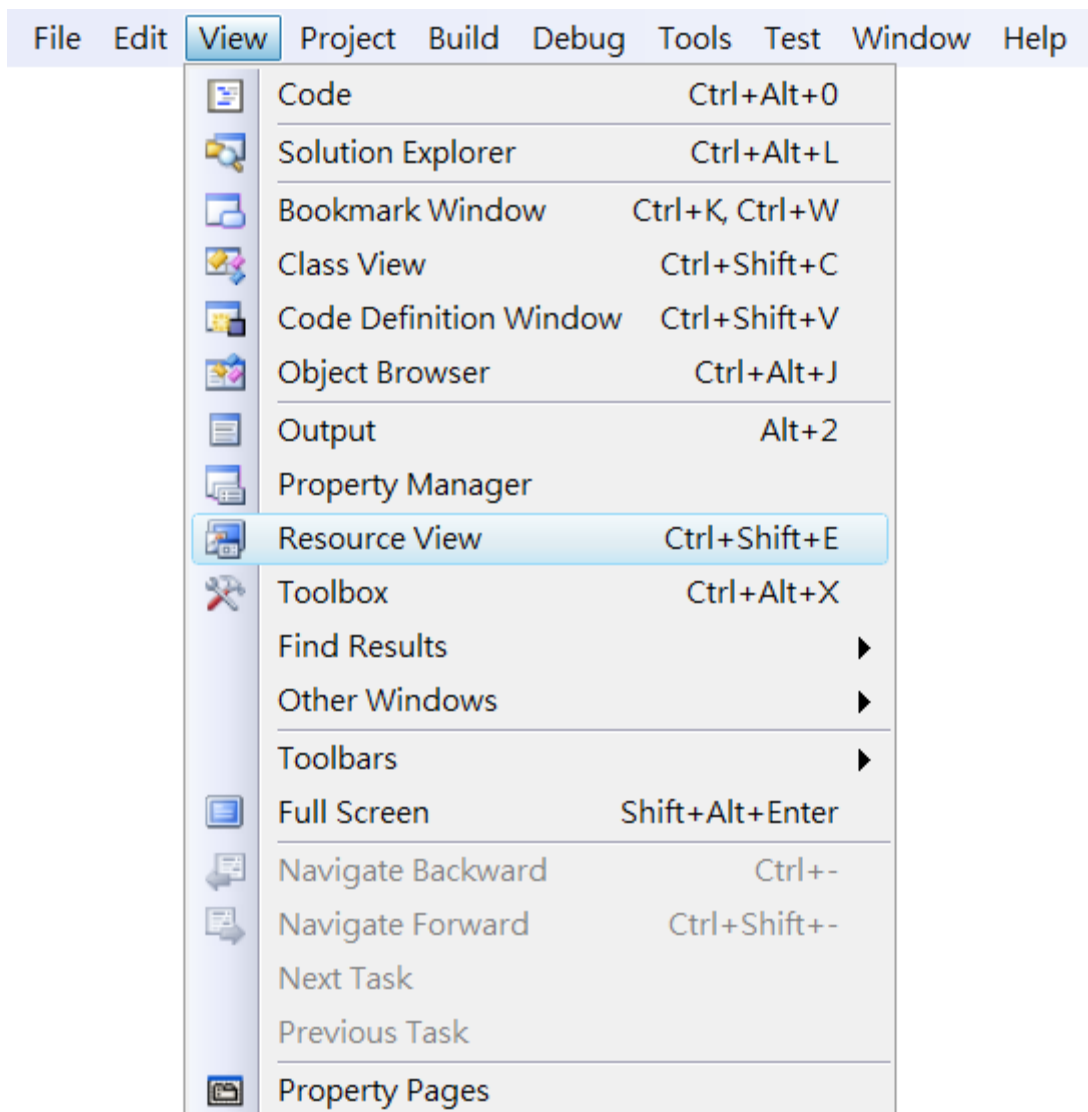
Step 4: Click the OK button to exit the Property Pages dialog

Adding a control to the form

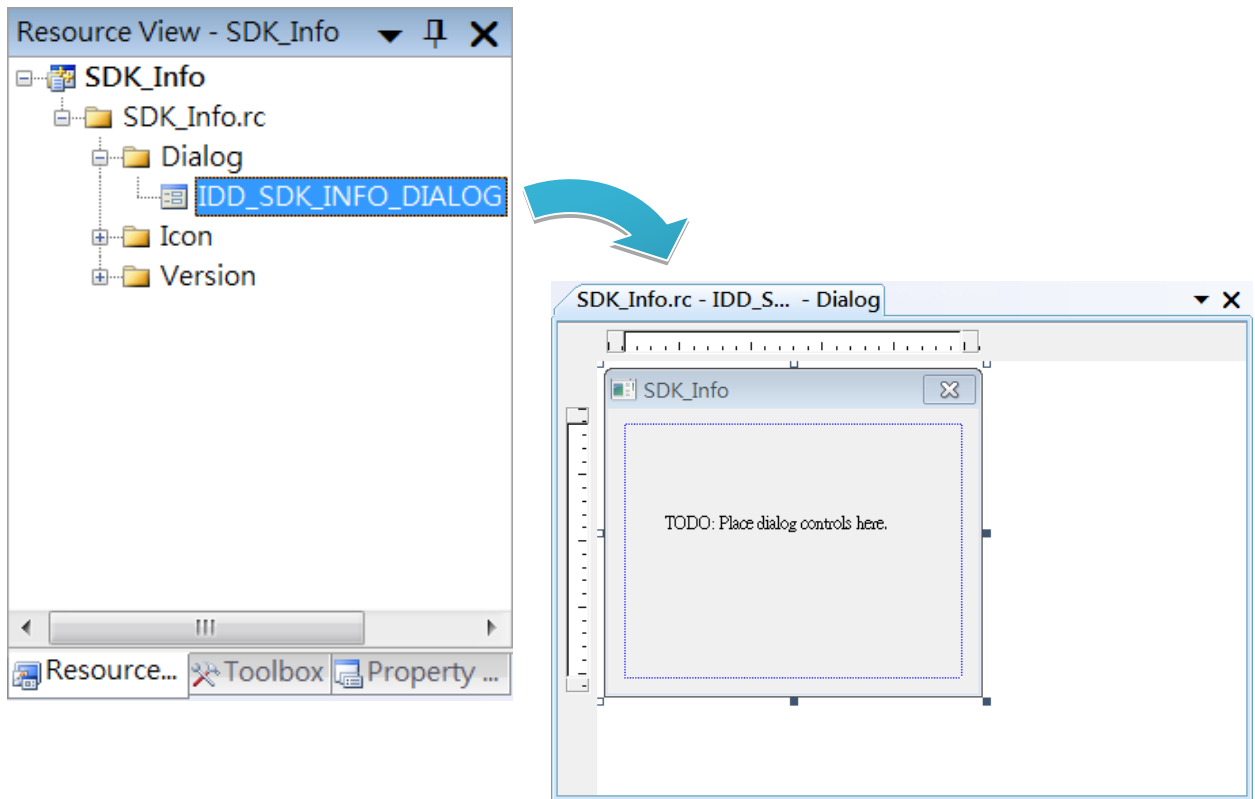
A variety of controls can be dragged from the Toolbox and placed onto the form. These controls, however, are not really "live"; they are simply images that can be used as a convenient method of moving an object to a precise location on the form.

After adding a control to the form, use the Properties dialog to set the parameter values for the control, such as the background color and the default text, etc. The values that are specified in the Properties dialog are the initial values that will be assigned to that property when the control is created at run time.

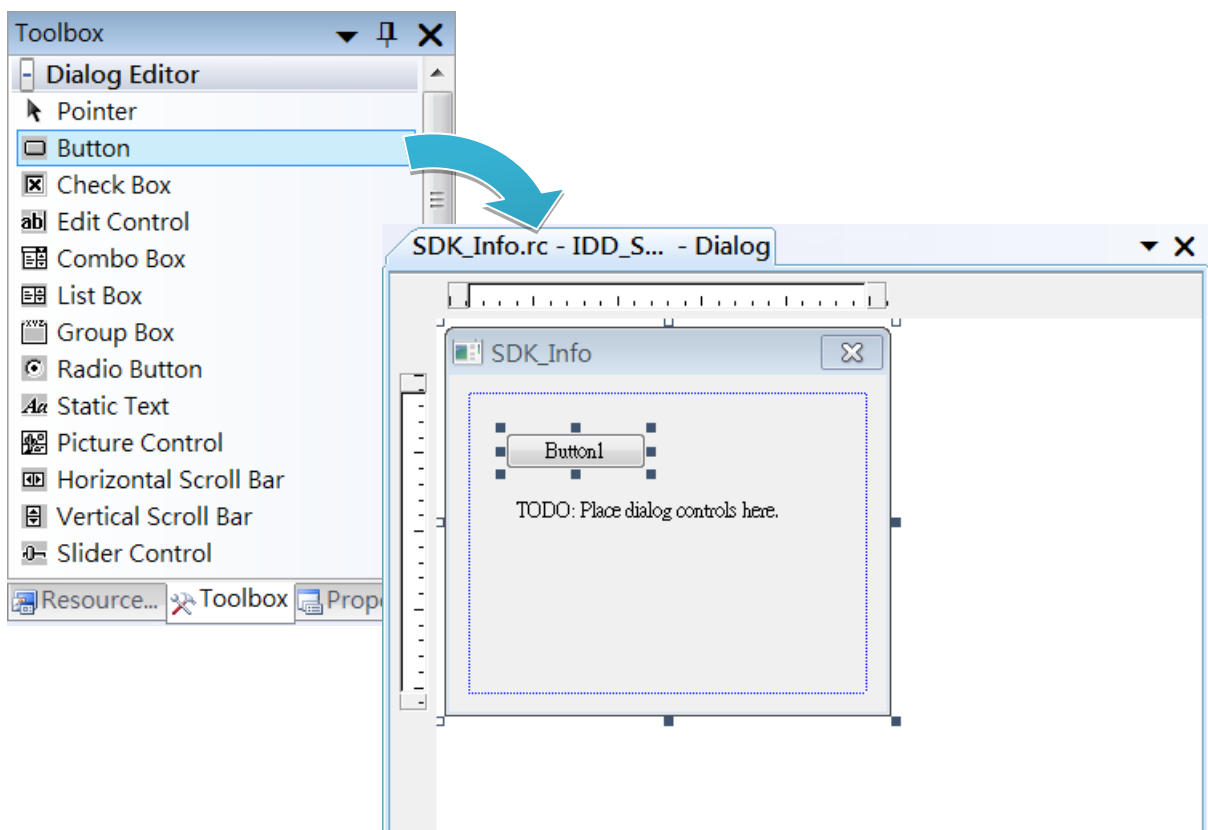
Step 1: On the View menu, click the Resource View option



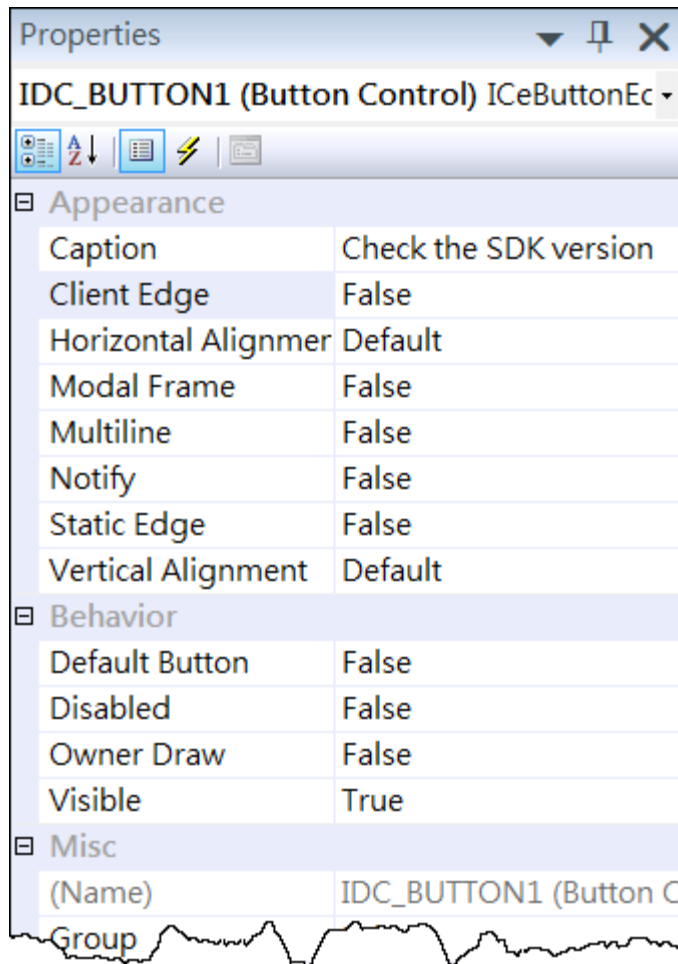
Step 2: In the Resource View Panel, expand the [Project name].rc file node, expand the Dialog node, and then click the plug-in dialog item



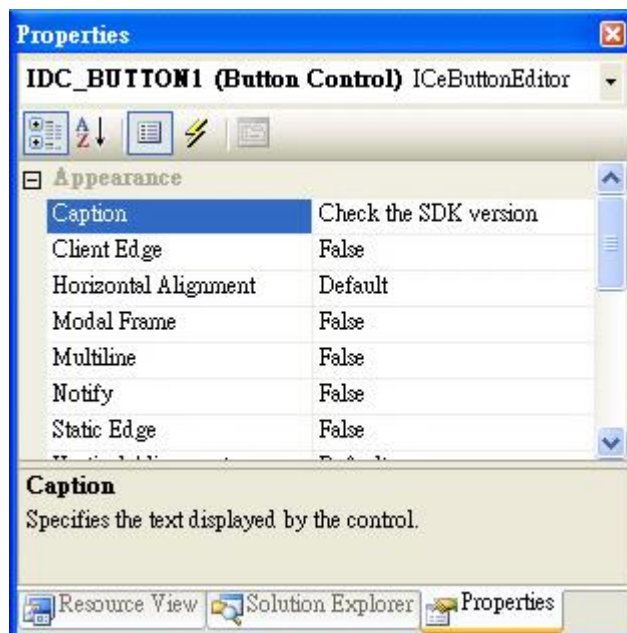
Step 3: Drag a Button control from the Toolbox panel onto the form



Step 4: On the Properties panel, type Check the SDK version in the Caption field



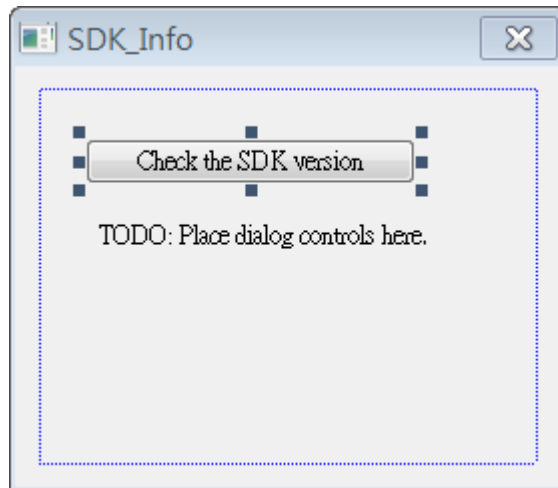
Step 5: In the Properties window, type Check the SDK version, and press ENTER to set the Text property



Adding the event handling for the control

Once the design stage for the application is complete, code can be added to provide the functionality for the program.

Step 1: Double-click the button on the form



Step 2: Insert the following code

```
char sdk_version[32];
TCHAR buf[32];
pac_GetSDKVersion(sdk_version);
pac_AnsiToWideString(sdk_version, buf);
MessageBox(buf,0,MB_OK);
```

```
void CSDK_InfoDlg::OnBnClickedButton1()
{
    // TODO: Add your control notification handler code here
    char sdk_version[32];
    TCHAR buf[32];
    pac_GetSDKVersion(sdk_version);
    pac_AnsiToWideString(sdk_version, buf);
    MessageBox(buf,0,MB_OK);
}
```

Step 3: Insert the following code into the header section

```
#include "PACSDK.h"
```

```
▣ // SDK_InfoDlg.cpp : implementation file
//
#include "stdafx.h"
#include "SDK_Info.h"
#include "SDK_InfoDlg.h"
#include "PACSDK.h"
▣ #ifdef _DEBUG
```