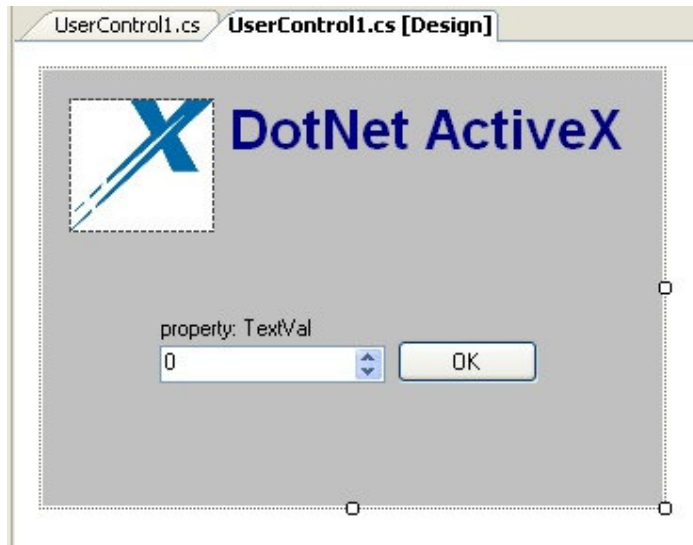


Creating an ActiveX using Csharp Dotnet UserControl

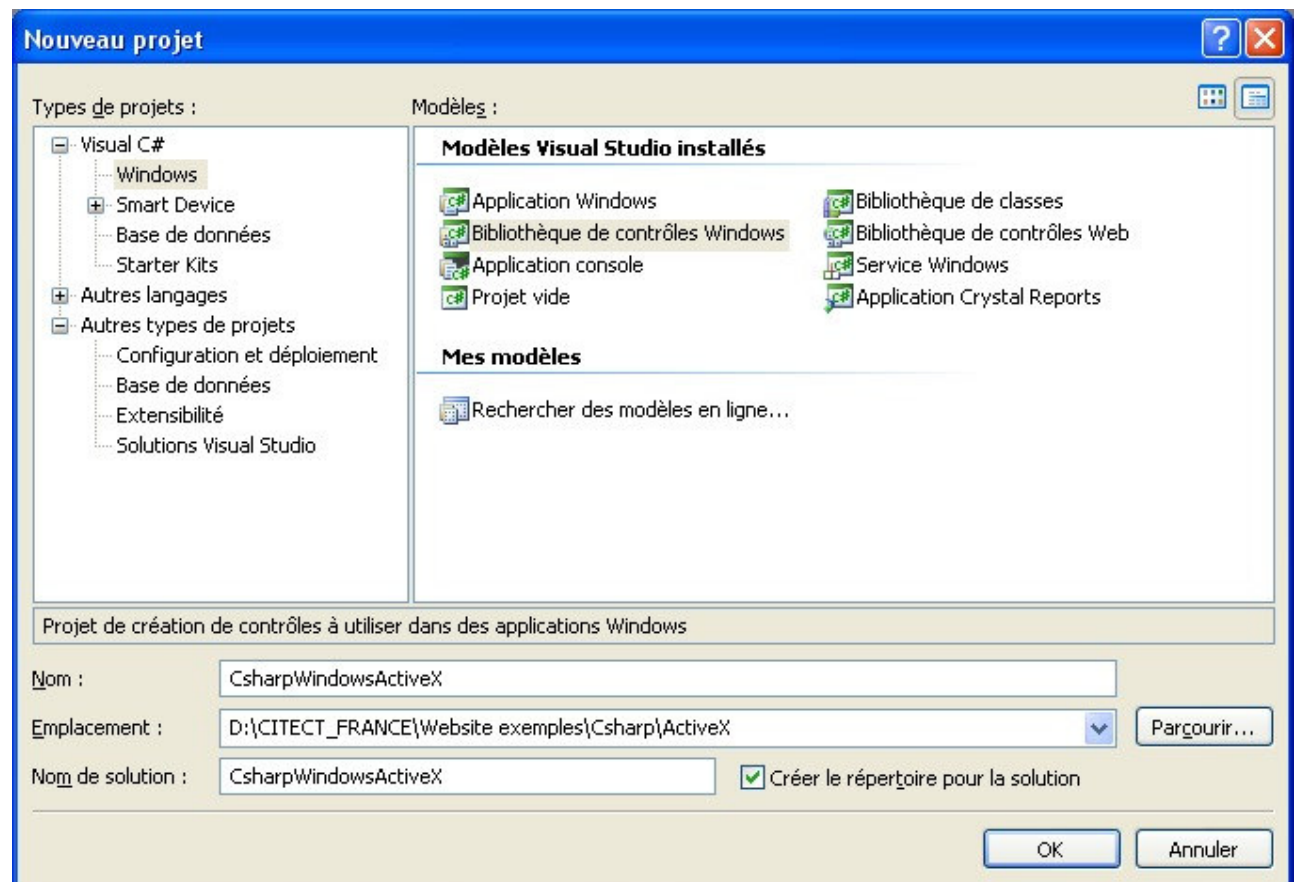
This article is about creating ActiveX controls using a DotNet UserControl in Csharp. You can design all ActiveX features like: properties, methods and events.

Environment: Visual Studio(2005) Csharp

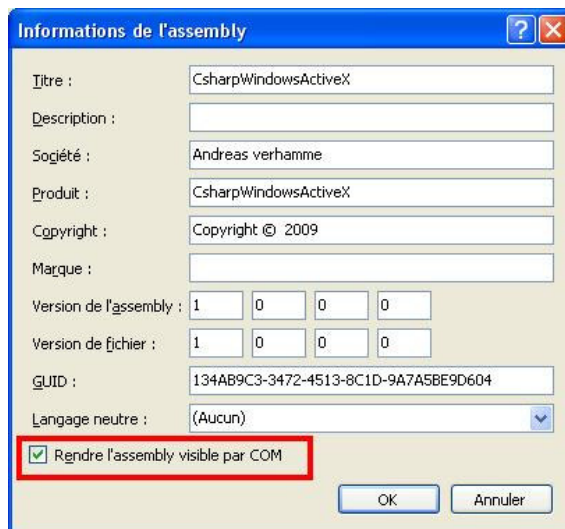


How to Do This

1. Create a Usercontrol using Visual Studio (C#):



2. Configure the project properties:



Informations de l'assembly

Titre : CsharpWindowsActiveX

Description :

Société : Andreas verhamme

Produit : CsharpWindowsActiveX

Copyright : Copyright © 2009

Marque :

Version de l'assembly : 1 0 0 0

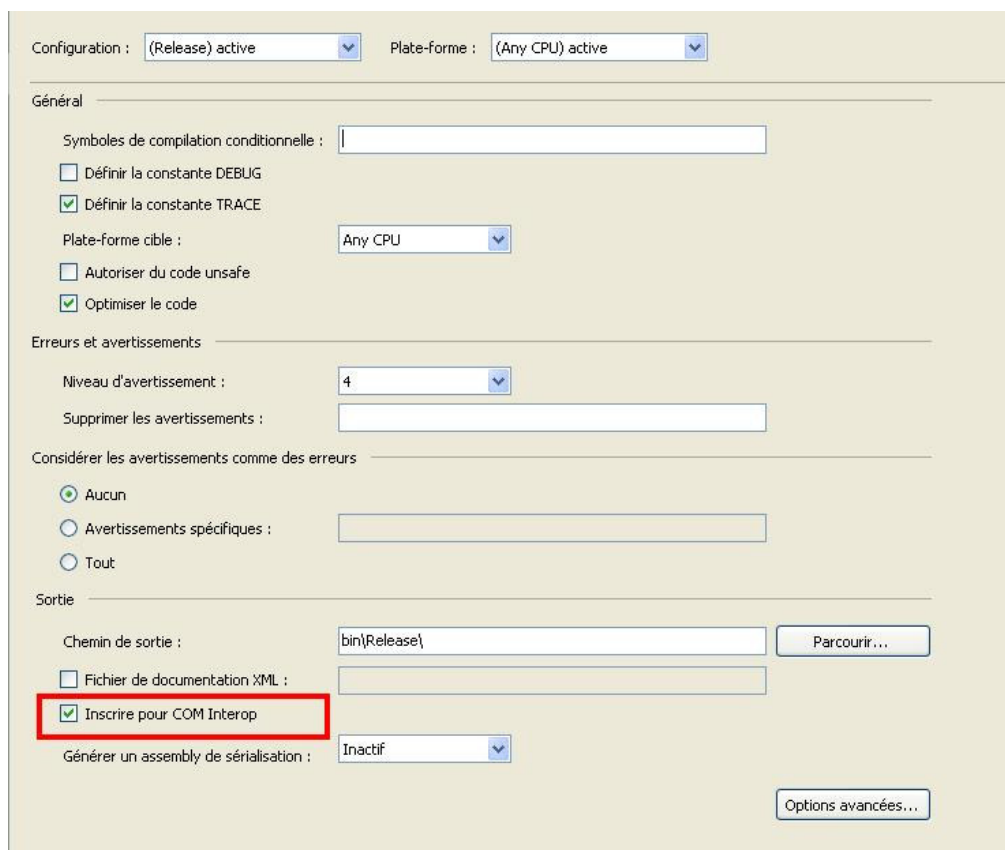
Version de fichier : 1 0 0 0

GUID : 134AB9C3-3472-4513-8C1D-9A7A5BE9D604

Langage neutre : (Aucun)

☒ Rendre l'assembly visible par COM

OK Annuler



Configuration : (Release) active Plate-forme : (Any CPU) active

Général

Symboles de compilation conditionnelle :

☐ Définir la constante DEBUG

☒ Définir la constante TRACE

Plate-forme cible : Any CPU

☐ Autoriser du code unsafe

☒ Optimiser le code

Erreurs et avertissements

Niveau d'avertissement : 4

Supprimer les avertissements :

Considérer les avertissements comme des erreurs

☒ Aucun

☐ Avertissements spécifiques :

☐ Tout

Sortie

Chemin de sortie : bin\Release\ Parcourir...

☐ Fichier de documentation XML :

☒ Inscrire pour COM Interop

Générer un assembly de sérialisation : Inactif

Options avancées...

3. Modify the usercontrol interface:

Note: Make sur you added the EVENTS class:

[ClassInterface(ClassInterfaceType.AutoDual), ComSourceInterfaces(typeof(UserControlEvents))]

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Drawing;
using System.Data;
using System.Text;
using System.Windows.Forms;
// Add references
using System.Runtime.InteropServices;
using System.Reflection;
using Microsoft.Win32;

namespace CsharpWindowsActiveX
{
    [ProgId("CsharpWindowsActiveX.ActiveXUserControl")]
    [ClassInterface(ClassInterfaceType.AutoDual), ComSourceInterfaces(typeof(UserControlEvents))]

    public partial class ActiveXUserControl : UserControl
    {
        public ActiveXUserControl()
        {
            InitializeComponent();
        }

        .....
    }
}
```

4. Add the register/unregister section in the source code

```
// register COM ActiveX object

[ComRegisterFunction()]
public static void RegisterClass(string key)
{
    StringBuilder skey = new StringBuilder(key);
    skey.Replace(@"HKEY_CLASSES_ROOT", "");
    RegistryKey regKey = Registry.ClassesRoot.OpenSubKey(skey.ToString(), true);
    RegistryKey ctrl = regKey.CreateSubKey("Control");
    ctrl.Close();
    RegistryKey inprocServer32 = regKey.OpenSubKey("InprocServer32", true);
    inprocServer32.SetValue("CodeBase", Assembly.GetExecutingAssembly().CodeBase);
    inprocServer32.Close();
    regKey.Close();
}

[ComUnregisterFunction()]
public static void UnregisterClass(string key)
{
    StringBuilder skey = new StringBuilder(key);
    skey.Replace(@"HKEY_CLASSES_ROOT", "");
    RegistryKey regKey = Registry.ClassesRoot.OpenSubKey(skey.ToString(), true);
    regKey.DeleteSubKey("Control", false);
    RegistryKey inprocServer32 = regKey.OpenSubKey("InprocServer32", true);
    regKey.DeleteSubKey("CodeBase", false);
    regKey.Close();
}
```

5. Add an ActiveX property:

```
// ActiveX properties (Get/Set) ////////////////////////////////////////  
  
private int ptextVal;  
public int TextVal  
{  
    get  
    {  
        ptextVal = (int)(numericUpDown1.Value);  
        return ptextVal;  
    }  
    set  
    {  
        ptextVal = value;  
        numericUpDown1.Value = ptextVal;  
    }  
}
```

6. Add an ActiveX method:

```
// ActiveX methods/functions ////////////////////////////////////////  
  
public interface ICOMCallable  
{  
    int GetTextBoxValue();  
}  
  
public int GetTextBoxValue()  
{  
    int i = (int)(numericUpDown1.Value);  
    MessageBox.Show("ActiveX method: GetTextBoxValue " + i.ToString());  
    return (i);  
}
```

7. Add an ActiveX event:

```
// Eventhandler interface //////////////////////////////////////
```

```
public delegate void ControlEventHandler(int NumVal);  
[Guid("0A415E38-372F-45fb-813B-D9558C787EA0")]  
[InterfaceType(ComInterfaceType.InterfaceIsIDispatch)]
```

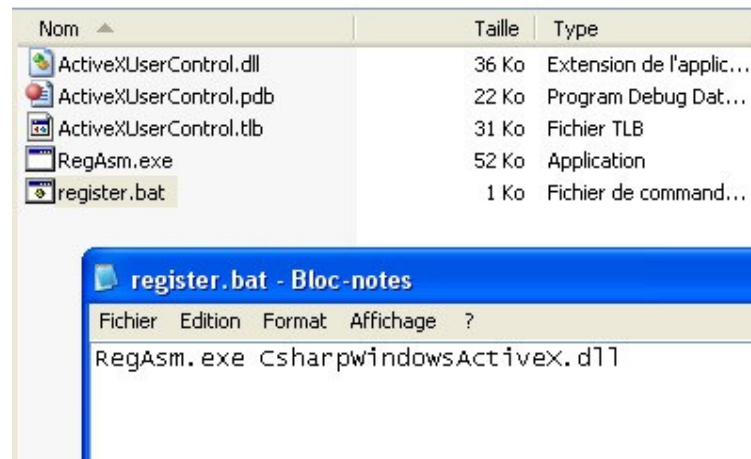
```
public interface UserControlEvents  
{  
    [DispId(0x60020001)]  
    void OnButtonClick(int NumVal);  
}
```

```
public event ControlEventHandler OnButtonClick;
```

```
private void buttonOK_Click(object sender, EventArgs e)  
{  
    int NumVal;  
    if (OnButtonClick != null)  
    {  
        NumVal = (int)(numericUpDown1.Value);  
        OnButtonClick(NumVal);  
    }  
}
```

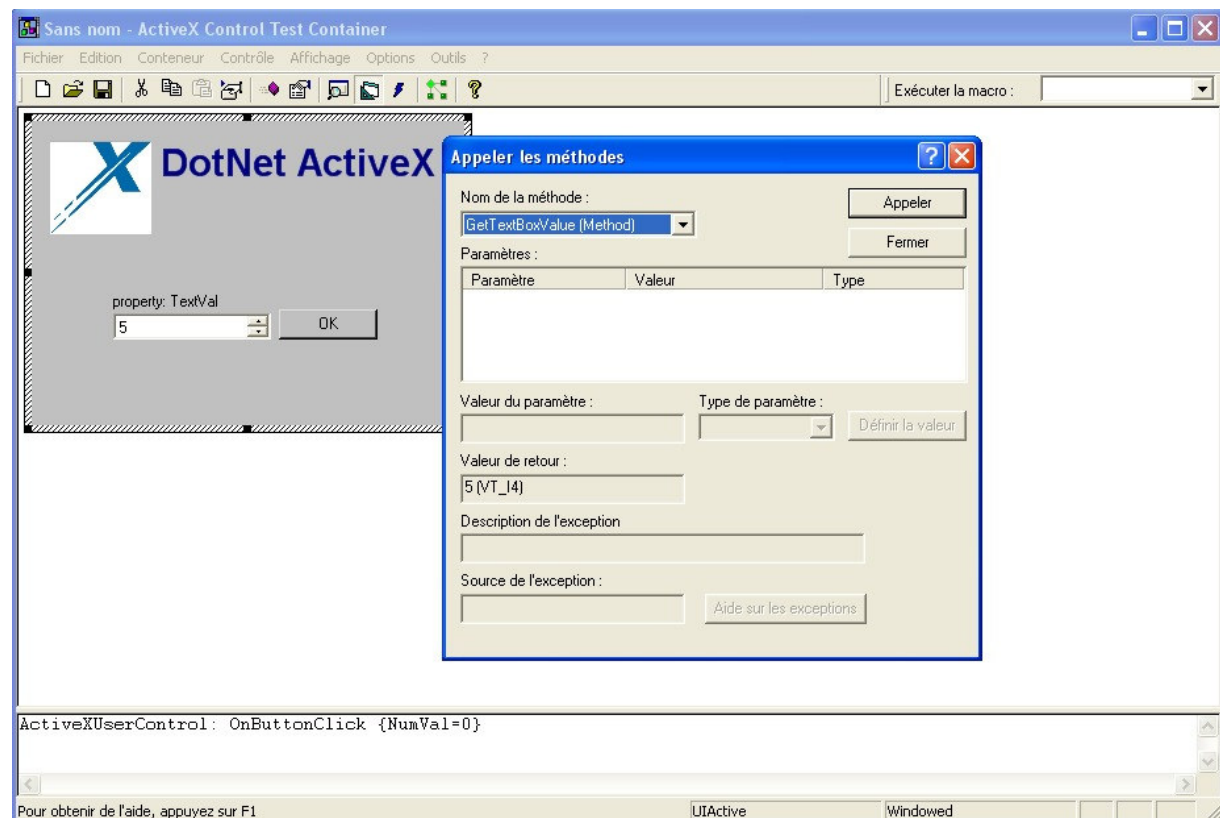
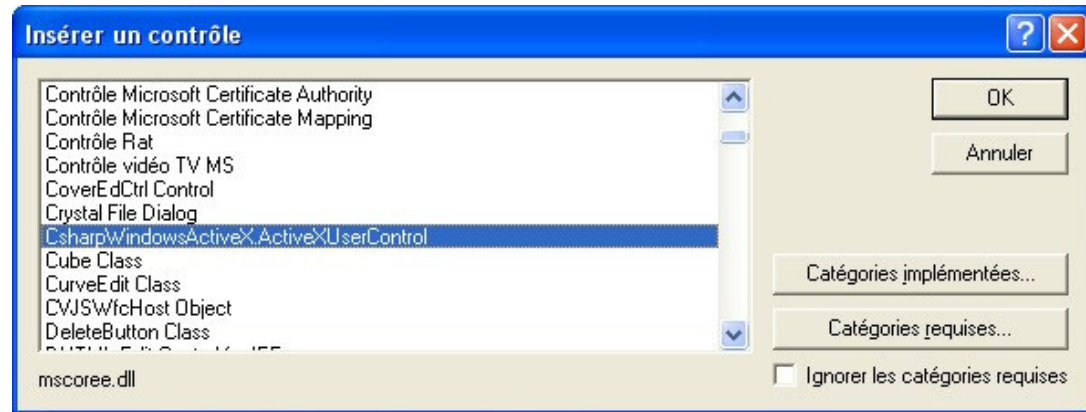
8. Register the ActiveX on your PC:

Register the new ActiveX on your computer using the command: RegAsm.exe CsharpWindowsActiveX.dll



9. Test your ActiveX:

Use the TSTCon32.exe tool from Visul Studio to test the ActiveX:



Andreas Verhamme