

Microsoft Visual Studio Wizard for EntireX DCOM Wrapper

The Microsoft Visual Studio Wizard for EntireX DCOM Wrapper is a Software AG plug-in that enables Microsoft Visual Studio 2010 to generate Visual C# .NET and Visual Basic .NET Windows applications that use EntireX DCOM Wrapper objects. The Visual Studio Wizard creates a DCOM Wrapper object from a Software AG IDL file and adds a reference to the object into the new Windows application.

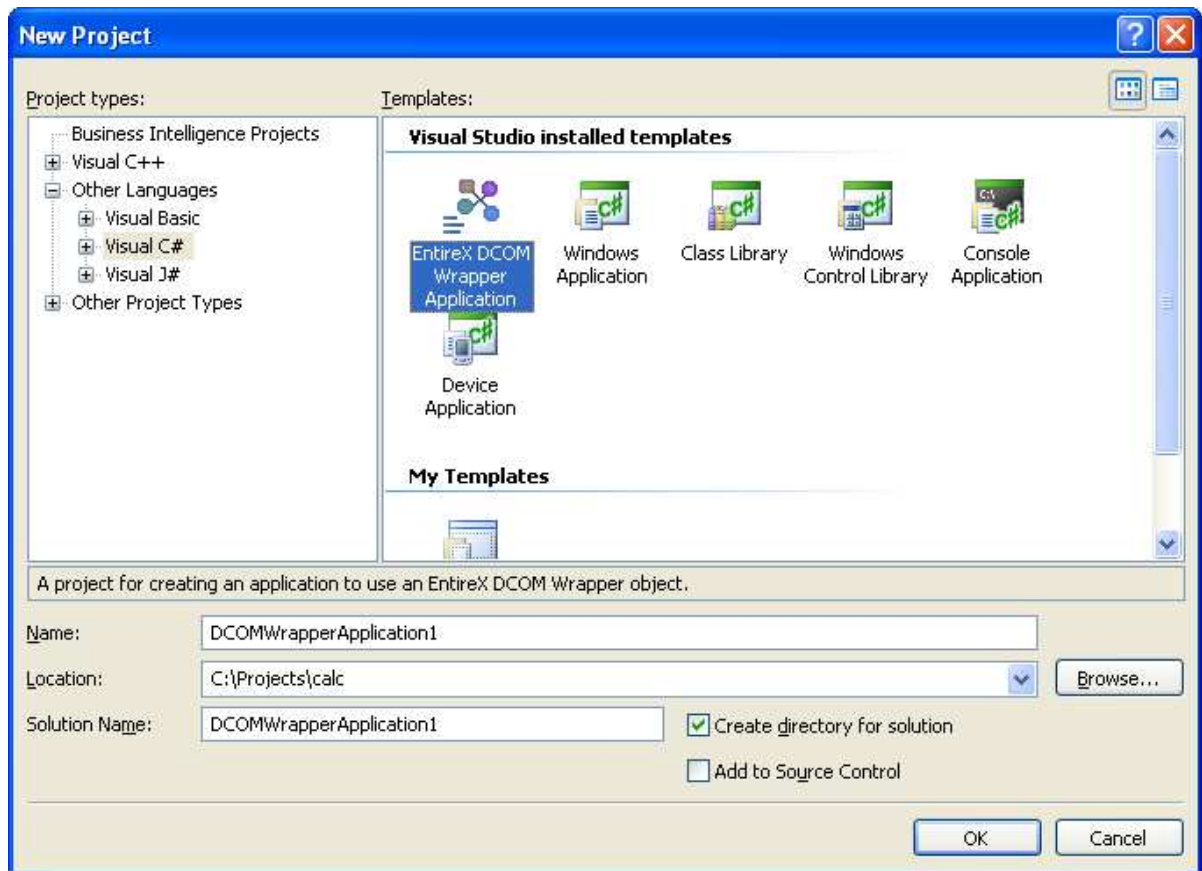
Notes:

1. The dialog boxes might differ depending on your active Visual Studio settings.
 2. Generation of a DCOM Wrapper proxy is not supported in the Visual Studio Wizard.
-

Creating a new DCOM Wrapper Project

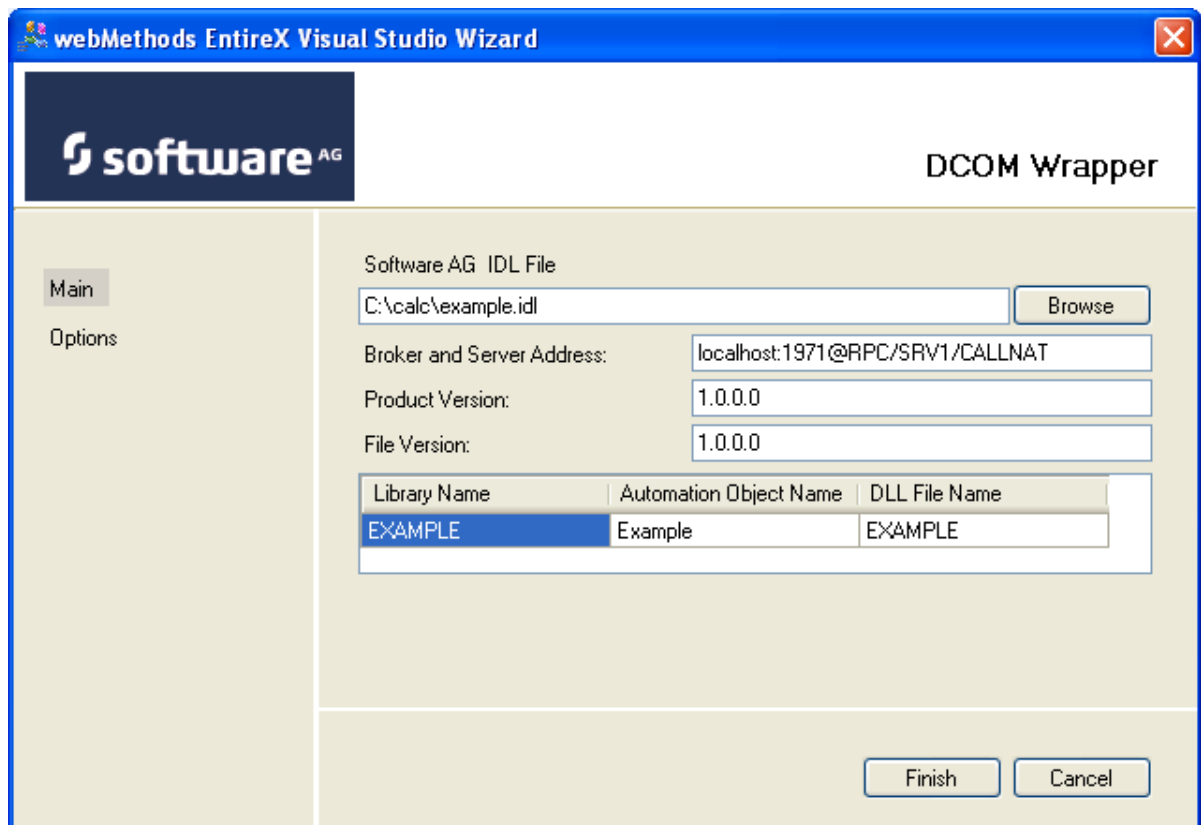
To create a new EntireX DCOM Wrapper Windows Application

1. Select the **File > New > Project** menu.
2. Choose your preferred programming language to create a new project. The Visual Studio Wizard is available for Visual C# and Visual Basic.
3. Select the Visual Studio Wizard icon, enter the name of the project and choose the project location.



4. Choose **OK**.

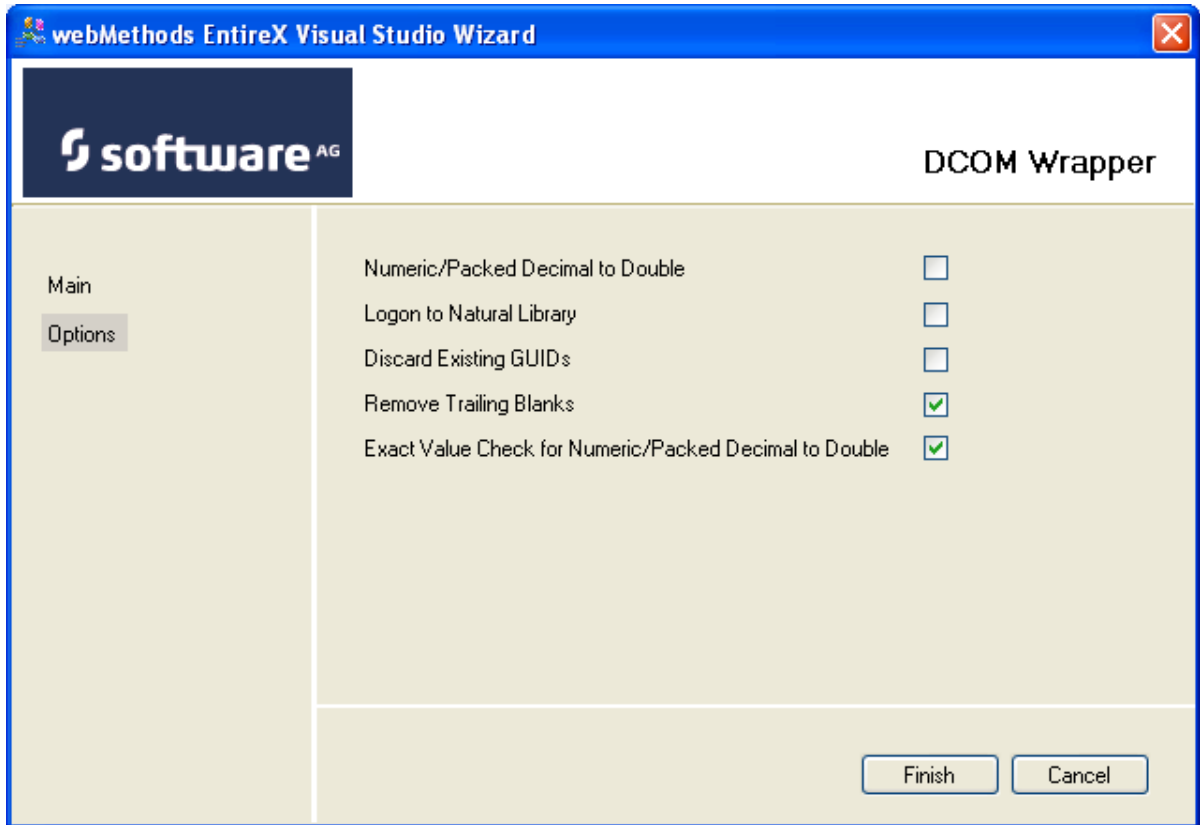
The Visual Studio Wizard starts and displays the main input panel.



5. Specify the Software AG IDL file.
The Software AG IDL file is mandatory in this dialog and the file you specify must be a valid Software AG IDL file.
6. Enter the broker and server address.
The broker address is by default set to localhost : 1971. The server address (class/server/service) is by default set to RPC/SRV1/CALLNAT. Please enter the required address of the broker and server for your application. Please use always the following notation for the broker and server address.
`<broker address>[:port]@<class>/<server>/<service>`
The port is optional if the broker runs on the default port 1971.
7. You can specify the product version and file version. This version information will be displayed in the properties of the generated DCOM DLL.
8. The library name is read from the library statement from the IDL file. The property is a read-only property in this dialog and cannot be changed.
9. The automation name is read from the alias statement of the library. If the alias statement is missing, the automation name is *EOL.<library name>*. You can change this property if required, and it will be used while generating the DCOM Wrapper object. Changes of the automation name in this dialog have no effect on the IDL file; they will not be saved into the IDL file.
10. The DLL file name is set by default to the library name. This value will be used while generating the DCOM Wrapper object and sets the output file name. You can set this value to generate a DLL with a name other than the library name. Please use this value without the extension *.dll*. The extension will be added automatically while generating the DCOM Wrapper DLL.

11. If you wish to specify further settings, choose the **Options** input panel on the left navigation panel, otherwise proceed to the next step.

The **Options** panel provides additional settings for generating the DCOM Wrapper object. See the *Setting DCOM Wrapper Properties* for a detailed description of these properties.



12. Choose **Finish** to start the generation of the DCOM Wrapper object. A process window will be displayed while generating the DCOM Wrapper object. When the generation process completes, a simple Visual Basic or Visual C# solution is available and ready to use.

The generated Visual Basic or Visual C# solution contains the reference to the new DCOM Wrapper object. If the Software AG IDL file has more than one library object, only the first library object will be shown in the example implementation, but all library objects will be added to the references of the solution.