



# RC Converter

## User Guide

## Steps to convert a Resource file

1. Choose the resource file which you wish to convert. Please, note that you can not select an .exe file, as we support only standard windows MFC or ATL resource files.
2. Click Next.
3. Select the folder where the .NET resource files will be generated. We recommend an empty folder to avoid the warning messages you will get during the conversion process if RC Converter finds existing files with duplicate file names.
4. Change the string resource file name to a unique name so that there will not be any conflict when you try to add this resource into your existing .NET project.
5. Choose the prefix for the various controls if you follow fixed naming conventions; otherwise leave the default settings.
6. Click Next.
7. Enter your class name prefix and suffix. Otherwise leave the default value.
8. Enter your namespace value. If you are going to include these files in your existing .NET project then you may want to use the same namespace for the project, otherwise you can choose any namespace value.
9. Enter your standard menu name suffixes and prefixes or leave the default values.
10. We recommend leaving the naming options check boxes unchecked. For instance, if you check 'Keep Dialog Names' and a dialog ID is IDD\_DIALOG, then your class name will be IDD\_DIALOG. If you check 'Keep Control Names' then your control names will be the same as the MFC ID, e.g. IDD\_EDIT. In the above example the Windows Form resource standard suggests FrmDialog as the class name and txtEdit as the control name. If you check these check boxes then you will also encounter a few controls with blank names because MFC supports having blank ID's in the case of popup menus. You will also encounter duplicate names of Label controls because MFC lets you use IDC\_STATIC as the ID for all static controls on a dialog or view. You will need to manually fix these issues in the converted .NET resource files.
11. Click Next.

12. Select your language of choice (C#, VB.NET or C++/CLI)
13. Select Visual Studio .Net 2003 or Visual Studio .NET 2005 as the development platform for which you wish to generate Windows Form resources.
14. We recommend not changing the X, Y, Width and Height scale factors until you see the converted dialog.
15. Click Next.
16. Choose the string resources which you wish to include in the conversion process. Update any ID's and the text of the string resources before conversion if you wish to do so.
17. Click Next.
18. By default, we associate menus with views and dialogs if you have the corresponding .h (header) and .cpp files in the same folder as your resource file. However, you have the flexibility to choose any menu item for a dialog or Form View.
19. Change the class name if you wish.
20. Change the namespace name if you wish.
21. Change the file name if required to make it unique.
22. Choose the dialogs which you would like to convert. By default, all dialogs will be included for conversion (except in trial mode).
23. Click the Generate .Net Resources button on the left corner at the bottom of the wizard.
24. If we found more menus in the resource file than have been attached, you will get a message asking if you wish to use the remaining menu items. You can say "No" and go ahead with the conversion or say "Yes" and associate additional menus with selected dialogs or form views.
25. Wait a bit while RC Converter converts your MFC or ATL resources to Windows Form resources.
26. Once the process is done you can see a log file specific to any dialog by clicking View Log button in the grid. You might also want to export the log to Excel or print it with your notes for future reference.

27. Click Next.
28. Check out the Success, Failure and Exception logs. You can export all three of them to Excel or print them with a customizable header for future reference.
29. Click Finish.
30. We recommend that you add the generated files to your project in the following manner: Copy all the files into your .NET project folder. From the Project menu item select the "Show All files" option (or on the Solution Explorer window toolbar press the "Show All files" icon). Then select all the files you want to include in the Solution Explorer, right click and select "Include in Project". If you try to add these files from the "Add | Existing item" menu option, then Visual Studio 2005 seems to have a problem when the number of files are greater than 15 or so. When adding from the "Add Existing Item" menu option, we recommend not adding the resx and designer.vb or designer.cs files directly as these will automatically be included when you add the form's vb or cs file.
31. If using C++/CLI, Dudelabs generates header files (h files) that the designer can read. However, by default, Visual Studio 2005 flags the imported h file as a "C++ Header file". You must manually set the "File Type" for the file properties to "C++ Form". If you select all of the imported h files, the file type can be set for all of the files at one time.
32. Good Luck with your migration to .NET .

Use DudeLabs .NET programming services and other advanced IT services if you need expert help in your conversion or new .NET development project.

Please visit our services page at <http://www.dudelabs.com/services.aspx>