SecWin Technical Document # 1
Subject : Creating Multi DLL applications

For the purposes of this discussion I will create 2 Dll's and an Exe. The Exe will reference both Dll's and the one Dll will reference the other. While the focus of this article is on implementing SecWin in such a setup, the general principles apply to all multi-dll projects. I will assume you are starting with a blank page - if you already have some Dll's then some of the work will already have been done.

The following files are associated with this document;
Root.App
Other.App
Main.App

SecWin
This document assumes that you have installed SecWin Version 1.7 or later, in it's 16 bit form. SecWin is availiable, in the 16 bit form, as FreeWare on both the Internat and the TopSpeed forum on Compuserve. To retrieve off the Internet point your Web browser at the IceTips Home Page (maintained by Arnor Baldvinson)

SecWin is a Template and Dll Extension which provides User Logins, Screen by Screen and Control by Control runtime user access control, across mutilple Exe's and Dll's. It is fully network compatible, and supports a central user database.

Copyright

Custom Business Software welcomes your comments, suggestions and critisisms. You can contact us in one of the following ways...
Internet : datasoft@iaccess.za
CIS : 10076,656
Fax : +27 21 797 4324
Phone : +27 21 797 9468
Mail : PO Box 18156, Wynberg, 7824, South Africa

Step 1 : Creating the root Dll.

The root Dll is the Dll we will use to store all the common variable and file declarations for the whole project. All the other Dll's and the Exe will reference this Dll.

1) Create a new App - I've called mine Root.Dll. You need to set the following application options;
a) Set the dictionary name to your Dictionary.
b) Change the destination to a Dll, not an Exe.
c) Change the first procedure name to something other than "Main". I called mine "RootMain" here.
d) Make sure the AppWizard check box is OFF
e) Click on OK to create the App.

2) Click on global, and set the following settings;
a) Click "Generate global data as External" OFF
b) Click on the "File Control Flags" tab, and click "Generate all file declarations" ON.
c) Click "Export all file declarations" ON.
d) Click on OK to return to the App tree.

3) Add the "Activate Security" extension template by clicking on "Global" then "Extensions" then "Insert" and then choose the "Activate Security" template. You can set the settings as follows
a) Set the "Unique application name" to the name of the whole project, not the name of the Dll. My whole project is called "MultiExample" so I've put that here. I'll use this name again later when we add the global extension to the other Dll and the Exe.
b) Click the ""Export SecWin data" check box ON.
c) Make sure the "SecWin data defied in another Dll" option is clicked OFF.
d) Click on Ok and then OK to get back to the App tree.

4) Import the Queues functions required by SecWin. To do this
a) Click on "Import text" in the file menu.
b) Choose the "Queues.Txa" file which should be in your \cw15\Libsrc directory.
c) You will see three functions added to your Application tree.

5) Create 2 global variables called AppNameDesc and AppNum which are declared as a string and long respectivly. You only need to do this in the Root Dll.
a) Click on Global.
b) Click on Data
c) Click on Insert
d) Type AppNameDesc for the variable name
e) Set the type to STRING.
f) Click on OK
g) Type AppNum for the variable name
h) Set the type to LONG.
i) Click on OK
j) Click on CANCEL, then CLOSE then OK to go back to the Application tree.

6) Click on the Compile button to compile the application. The following files will be created;
a) In the same directory as the App file will be the Dll file, in this case Root.Dll.
b) In the \cw15\Obj\ directory will be the associated Root.Lib file.

7) We are done with the root Dll, so click on "Ok" to save it and close the App file.

<u>Step 2 : Creating the "other" Dll.</u>

1) Create a new App - I've called mine Other.Dll. You need to set the following application options;
a) Set the dictionary name to your Dictionary.
b) Change the destination to a Dll, not an Exe.
c) Change the first procedure name to something other than "Main". I called mine "OtherMain" here.
d) Make sure the AppWizard check box is OFF
e) Click on OK to create the App.

2) Click on global, and set the following settings;
a) Click "Generate global data as External" ON
b) In the "File Attributes" section change the "External" option to read "All External".
c) Click the "All files declared in another App" option ON.
d) Click on OK to return to the App tree.

3) Add the "Activate Security" extension template by clicking on "Global" then "Extensions" then "Insert" and then choose the "Activate Security" template. You can set the settings as follows
a) Set the "Unique application name" to the name of the whole project, not the name of the Dll. My whole project is called "MultiExample" so I've put that here. I'll use this name again later when we add the global extension to the Exe.
b) Click the ""Export SecWin data" check box OFF. Note that this is the opposite to the root Dll.

c) Make sure the "SecWin data defied in another Dll" option is clicked ON. Note that this is opposite to the root Dll.
d) Click on Ok and then OK to get back to the App tree.

4) Add the required libraries to the project by...
a) Click on the Project button.
b) In the Library and Object files section add the Root.Lib and SecWin16.Lib files.
c) Click on OK to get back to the App tree.

5) Create as many functions here as you like. I'm going to create a Browse and a Form which we'll call later from the Exe application. I create these in the normal fashion by...
a) Clicking on the Procedure menu and choosing New.
b) Choosing the browse template, and letting the Wizard do the rest.

Notice the following;
Both the Browse and the Form have the Export attribute clicked on. As we are only going to call the browse function from the Exe, the form, strictly speaking doesn't need this attribute on, but I'm going to leave it on anyway.

6) Add SecWin security to the browse and form in the usual way, by using the Screen Security Extension template. In other words...
a) Click on the Browse Function, and then on "Properties", and then on "Extensions".
b) Click on "Insert" and choose the "User Screen Security" template.
c) Enter a unique screen name, I'll call mine "Browse Wages"
d) Add the access restrictions to the Insert, Change and Delete buttons.
e) Click on OK and OK to return to the App tree.

7) Click on the Compile button to compile the application. The following files will be created;
a) In the same directory as the App file will be the Dll file, in this case Other.Dll.
b) In the \cw15\Obj\ directory will be the associated Other.Lib file.

Step 3 : Creating the "main" Exe.

1) Create a new App - I've called mine Main.Exe. You need to set the following application options;
a) Set the dictionary name to your Dictionary.
b) Leave the destination type as an Exe.
c) Leave the name of the "First Procedure" as Main.
d) Make sure the AppWizard check box is OFF
e) Click on OK to create the App.

2) Click on global, and set the following settings;
a) Click "Generate global data as External" ON
b) In the "File Attributes" section change the "External" option to read "All External".
c) Click the "All files declared in another App" option ON.
d) Click on OK to return to the App tree.

3) Add the "Activate Security" extension template by clicking on "Global" then "Extensions" then "Insert" and then choose the "Activate Security" template. You can set the settings as follows
a) Set the "Unique application name" to the name of the whole project, not the name of the Dll. My whole project is called "MultiExample" so I've put that here.
b) Click the ""Export SecWin data" check box OFF. Note that this is the opposite to the root Dll.
c) Make sure the "SecWin data defied in another Dll" option is clicked ON. Note that this is opposite to the root Dll.
d) Click on Ok and then OK to get back to the App tree.

4) Add the required libraries to the project by...
a) Click on the Project button.
b) In the Library and Object files section add the Root.Lib, Other.Lib and SecWin16.Lib files.
c) Click on OK to get back to the App tree.

5) Add the required prototypes for the procedures in the Other.Dll by doing the following...
a) Create an "External Lib Module" by choosing "Insert Module" from the Application menu.
b) Choose "External DLL" as the Module Type.
c) Set the "Name" to be Other.Lib, then Click on Ok.
d) Add a new procedure to this module by clicking on "New" in the Procedure menu. Give it the same name as the name of the procedure in the Dll. In this case, "BrowseWages".

6) Now we're going to create the Frame procedure, Main, that will call the BrowseWages procedure by...
a) Click on the "Main" procedure on your App tree, and click "Properties".
b) Choose "Frame" from the list of templates.
c) Click on the "Window" button, then the "Menu" menu, then "Menu Editor".
d) Add an item to the "Edit" menu, call it "Browse Wages"
e) Set the Actions for this item to be "Call a Procedure", set the procedure to be "BrowseWages" and click "Initiate Thread" on.
f) Click on Close, then Exit, then Yes, then Ok to return to the App tree.

7) Add the SecWin Login feature to the frame by...
a) Click on the "Main" procedure and then click on properties.
b) Click on Extensions, then Insert, then choose the "User Login Here" template.
c) Set the logon area name. I called mine "MainLogon".
d) Set the other settings to your preference, I chose "Case Insensitive Logins", and "Allow 3 tries".
e) Click on Ok and then Ok to return to the application tree.

8) Click on the Compile button to compile the application. The following file will be created;
a) In the same directory as the App file will be the Exe file, in this case Main.Exe

9) Run the Main.Exe by clicking on the run button.

<u>Step 4 : Distributing your Exe</u>

You will need the following files when distributing your Exe.
a) Main.Exe
b) Other.Dll
c) Root.Dll
d) SecWin16.Dll

End od Document - Last updated 1 Feburary 1996