Subject: Upp Widgets in a dll
Posted by luoganda on Thu, 25 Apr 2024 18:13:10 GMT
View Forum Message <> Reply to Message

This is a known issues for a long time
and this post is here just to be clear about this,
since it would be very neat to write dlls/so's in upp.

A few days ago i made a simple demo
of a basic uppApp that uses uppDll - done on win11 with upp17221.

I tried to compile it in debugVer but it doesn't compile(in release it does)
and i think there was something about '...memoryleaks... not declared' that usually
goes in hand when checking 'UseMAlloc' or 'CheckInit' appFlags.

As i saw in uppsrc UseMalloc is always on for dll, so it seems here - this is not a problem,
since it's the same without UseMalloc - dllApp does not work as it should in both cases.

TestCode:
1) you can check and compile in Debug(in my case i can't) and check why it doesn't compile.
2) when one runs this - app starts as expected, but when pressing 1stButton(which just call's simple PromptOK in dll),
   and pressing ok - app crashes without any notice.

ADDED:
just a simple typo in DllPlugTest.7z,
simply edit DllPlugTest/DllPlugTest.upp and correct from DllPlugTest.h into DllPlugTest.hpp.

ADDED2:
i tried to compile same code with upp17190 in linux debug and it compiles.
i overlooked that widgets can't be declared globally, so just correct the
line in DllCode from 'Array<Ctrl> ctrls' to 'Array<Ctrl> &ctrls=Single<Array<Ctrl>>()'.
I left the flags DLL(fromwin11) and added so flag - which is probably ok,
and corrected from 'DLL_APP_MAIN' to 'GUI_APP_MAIN' - since linux doesn't have SO_APP_MAIN.
Even if i selected 'All static' in 'Clang debug' Output mode,
i still have to put some(not all) so files in same dir as DllPlugTest elf(with of course DllCode.so),
else DllCode.so doesn't have all dependencies satisfied. This should not happen since upper static flag was used,
altough messages says -shared is throwed into compiler flags somewhere.
I used extra compiler flags in linux: -fPIC -fdeclspec.
Ok - here compiles and runs, but after hitting 1st button terminal says:
'...assertion failed... GUI widgets cannot be global variables', altough
i corrected that mistake - which is written above.

File Attachments

1) DllPlugTest.7z, downloaded 96 times

Subject: Re: Upp Widgets in a dll
Posted by Oblivion on Sat, 27 Apr 2024 10:17:27 GMT
View Forum Message <> Reply to Message

Minor technical details -in this context- between global variables and singletons aside, you are in fact declaring the GUI widgets as global variables:

Quote:i overlooked that widgets can't be declared globally, so just correct the
line in DllCode from 'Array<Ctrl> ctrls' to 'Array<Ctrl> &ctrls=Single<Array<Ctrl>>()'.

This will open a can of worms, since their destruction time or order is still left rather undefined. Hence the assertion failure. You need to declare your widgets in a local scope. Singletons won't help you here. Your widgets should belong somewhere. (At least, you should first construct them in your main code)

IMO, you can use the dll as a widget factory & layout generator instead. (i.e you have the array of widgets (ctrls) in your main code and let the dll just create the right type of ctrls on request in that array (since Upp:Array allows polymorphism.)

Best regards,
Oblivion

Subject: Re: Upp Widgets in a dll
Posted by luoganda on Sat, 27 Apr 2024 21:20:58 GMT
View Forum Message <> Reply to Message

Quote:
Minor technical details -in this context- between global variables and singletons aside, you are in fact declaring the GUI widgets as global variables:

Quote:
i overlooked that widgets can't be declared globally, so just correct the
line in DllCode from 'Array<Ctrl> ctrls' to 'Array<Ctrl> &ctrls=Single<Array<Ctrl>>()'.
Yes, but i read the docs and it says this should work.
Also, Singletons or fn(){static Widg&r;ret r;} are kindof global vars, but
this test code works in regular guiApp(but not in so - i am not sure for dll, could be that some kind of INITBLOCK would help):

```
Array<Ctrl> &ctrls=Single<Array<Ctrl>>();

GUI_APP_MAIN{
 ctrls.Create<TopWindow>();
 ((TopWindow*)&ctrls.Top())->Run();
 ctrls.Clear();
 Ctrl::ProcessEvents();
 PromptOK("This was not killed! nor assertion failed for GlobalWidgets");
```

Quote:
IMO, you can use the dll as a widget factory & layout generator instead. (i.e you have the array of widgets (ctrls) in your main code and let the dll just create the right type of ctrls on request in that array (since Upp:Array allows polymorphism.)

Yes i could, but i mean here for the broadest range of usage of widgets in dll, in general - so to speak.
So as you proposed(Array<Ctrl> in app and factoryCreator in so/dll) and like it's done in DllCode/main.cpp).
Both ways(and maybe others not mentioned here) should work for dynamic usage of using widgets via so/dll.

Subject: Re: Upp Widgets in a dll
Posted by luoganda on Wed, 01 May 2024 10:49:00 GMT
View Forum Message <> Reply to Message

here is also a linux version.

But - if one copies so file to elfBinary and uses
ldd DllCode.so there - there are some missing shared libs even if
so was linked with AllStatic(altough it seems somewhere -shared is thrown in at the compiler).
More info in 1stPost.

File Attachments

1) DllPlugTest-linux.7z, downloaded 98 times
