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.