Subject: Re: Creating a com object
Posted by tojocky on Tue, 14 Jun 2011 18:15:38 GMT
View Forum Message <> Reply to Message

rylek wrote on Sun, 12 June 2011 23:54Hello Tojocky!

I've just had a quick look at your test example. I've tried to link it statically under VC71 and, even with the commented String lines, I'm unable to register the object using regsvr32; CComModule::RegisterServer fails here:

```
inline HRESULT CComModule::RegisterServer(BOOL bRegTypeLib /*= FALSE*/, const CLSID*
pCLSID /*= NULL*/) throw()
{
 HRESULT hr = S_OK;
 _ATL_OBJMAP_ENTRY* pEntry = m_pObjMap;
 if (pEntry != NULL)
```

Here, the m_pObjMap member is NULL and therefore registration bails out with an error code. Unfortunately I'm not a big MFC expert, I co-authored U++ among others in order not to have to become one, but it seems to me that the trouble is caused by MFC's DllMain not being called during OCX initialization (DllMainCRTStartup should invoke it through _pRawDllMain which gets initialized in MFC's atlmfc/src/mfc/dllmodul.cpp, but as I see it in the debugger it contains NULL).

Theoretically I can imagine that usage of U++ somehow brings in linking of standard C++ libraries before MFC libraries (as U++ uses standard C++ libraries but not MFC) and this causes the dummy DllMain stub from crt/src/dllmain.c to be linked into the application instead of the correct one from src/mfc/dllmodul.cpp. It might be possible to call the MFC's DllMain manually by declaring DllMain in your application and calling directly the RawDllMain function from the MFC package, but I'm afraid the obvious problem might be a mere after-effect of a deeper problem in MFC initialization in a DLL linked by U++ IDE (omission of a flag Developer Studio sets automatically or something like that); in my opinion it would be best to generate a dummy MFC OCX using Developer Studio, then try to re-build it using TheIDE and compare the linker command lines.

Regards

Tomas

Heloo Tomas,

Thank you very much for explanation.