

Re: Outlook Redemption

Source: <http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework/2004-12/0966.html>

From: Dmitriy Lapshin [C# / .NET MVP] (*x-code_at_no-spam-please.hotpop.com*)

Date: 12/17/04

Date: Fri, 17 Dec 2004 10:19:21 +0200

Dmitry,

I guess you're the author of Outlook Redemption :-)

See, the original poster says the DLL works fine on his development computer – and it stops working when deployed to another computer. As we know, the GUIDs cannot change if a DLL is just being copied to another machine. Therefore, it is unlikely that the problem is with hard-coded GUIDs. On the other hand, .NET has specific rules on where to search for an assembly. Once a COM-visible assembly is registered without the /codebase option (which probably happens in the original poster's installer), the Fusion loader might be unable to find it when an instance of an object from this assembly is requested. The /codebase option guarantees the loader is always able to find and load the assembly and to create a COM-Callable Wrapper for the requested object instance.

--

Sincerely,

Dmitriy Lapshin [C# / .NET MVP]

Bring the power of unit testing to the VS .NET IDE today!

<http://www.x-unity.net/teststudio.aspx>

"Dmitry Streblichenko (MVP)" <dmitry@dimastr.com> wrote in message news:eI5f%23364EHA.1392@tk2msftngp13.phx.gbl...

> Since you customized the dll to use custom class names and GUIDs, you need

> to be careful about how you create the objects: if you use the regular

> "new", the compiler hardcodes the class GUID, which is obviously missing.

> Try to use something like

>

> Type t = Type.GetTypeFromProgID("MyDll.MyMailItem");

> SafeMailItem sItem = (SafeMailItem) Activator.CreateInstance(t);

>

>

> Dmitry Streblichenko (MVP)

> <http://www.dimastr.com/>

> OutlookSpy - Outlook, CDO

> and MAPI Developer Tool

>

>

>

> "Glyn Meek" <gjmeek@earthlink.net> wrote in message

> news:Sw8wd.576\$RH4.284@newsread1.news.pas.earthlink.net...

>>

>>

microsoft.public.dotnet.framework: Re: Outlook Redemption

```
>> We have just purchased a development tool called Redemption for Outlook
>> which lets us deal with that vagaries of interfacing to all the different
>> versions of Outlook. Works wonderfully well (although the documentation
>> is
>> predictably DREADFUL!!).
>>
>>
>> Our problem: When we install and then run our software on other
>> computers,
>> we get a runtime exception "Cannot create ActiveX control" in the
>> routines
>> where we are creating a new redemption object (like
>> "SafeContactItem".renamed desktop_ContactITEM in our dll.). Again, this
> is
>> code works fine on a system where the desktop.dll HAS been registered -
> ie
>> my development system. We are using Visual Studio/2003 and Visual Basic,
> and
>> in the VS installation/Setup project, I have tried making the desktop.dll
>> file register as vsdraCOM and vsdraCOMRelativePath.
>>
>> Does anyone else have experience of using this tool (we can't get hold of
>> their support folks to save our lives, even though it cost us $199 for
>> the
>> software!!!!). It seems to be a problem putting their dll into the
>> registry!!
>>
>> HELPPPPPPP anyone?
>>
>>
>>
>> Glyn Meek
>>
>>
>
>
```