

Re: cyclic dependency

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2005-12/msg04753.html>

- *From:* "Pohihihi" <noemail@xxxxxxxxxxx>
 - *Date:* Fri, 23 Dec 2005 22:30:08 -1000
-

Nicholas,

Yes, I agree, but we are limited to what is giving as other team do not agree to change anything other than adding one line for our part. Only thing I wish is that managers would understand more of the programming than deadlines.

Thanks.

"Nicholas Paldino [.NET/C# MVP]" <mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message news:uSyQ5MFCGHA.2908@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

> Po,
>
> Forgive me for saying so, but I see cyclical references between
> assemblies as a sign of bad design. If they really have a need to have
> that level of familiarity with each other, then I don't see why they can't
> be in the same assembly.
>
> If you really need this, the only way to do it is the way that Jay
> suggested, using interfaces. Otherwise, you will have to place the
> classes in the same assembly.
>
> On top of that, I don't think that you will find circular references in
> .NET anytime soon. I think it was considered at one point, but shot down
> pretty quickly.
>
> Hope this helps.
>
>
> --
> - Nicholas Paldino [.NET/C# MVP]
> - mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
>
> "Pohihihi" <noemail@xxxxxxxxxxx> wrote in message
> news:O8Tau7DCGHA.336@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Re: cyclic dependency

> This is a report from VSS newsgroup in hope of getting some more
> suggestions.
>
> Thanks,
> Po
>
> -----
> I guess I am not getting the whole picture here. How does this apply to
> the
> condition when there is a cyclic reference for the binaries (dlls) in
> dotNet
> !!
> What I am thinking is that to use the real ProjectB object I will still
> need
> to create a object from ProjectB for mapping in ProjectA, which again puts
> me in loop.
>
> Two interfaces can be made for 2 different project but still for mapping
> unless I use dynamic assmebly invoke I do not see how we can do somethng
> like following ----
>
> // 1st physical binary (MyProject.Home.dll
> namespace MyProject.Home
> {
> public class ProjectA: System.Windows.Forms.Form
> {
> public ProjectA()
> { // create object to use later}
>
> public void MyProjA_Method()
> { // do something }
>
> public void MyProjAMethodCallsProjBMethod()
> {
> int i = (new ProjectB()).CalBalls(5);
> }
> }
> }
>
> //2nd physical binary (MyProject.Work.dll
> namespace MyProject.Work
> {
> public class ProjectB: System.Windows.Forms.Form
> {
> public void MyMethodInB()
> {
> ProjectA pA = new ProjectA();
> pA.Show();
> }
>
> public int CalBalls(int i)

Re: cyclic dependency

Re: cyclic dependency

```
> { // return some calculation as int }
> }
> }
>
>
>
> "Jay B. Harlow [MVP – Outlook]" <Jay_Harlow_MVP@xxxxxxxxxxxxxx> wrote in
> message news:uX7Dn%238BGHA.2036@xxxxxxxxxxxxxxxxxxxxxxxxxxxx
>> Pohihihi,
>> Normally when I have two projects that need to refer to each other I use
>> a
>> Separated Interface Pattern.
>>
>> http://www.martinfowler.com/eaCatalog/separatedInterface.html
>>
>> Depending on the needs of the solution will decide if I use 2 or 3
>> projects
>> to implement the Separated Interface Pattern.
>>
>> The "easiest" way is to put the interfaces in the first project, then
>> reference the first project from the second. Alternatively you can put
>> the
>> interfaces in a project by themselves, and reference the interface
>> project
>> from both of the other projects.
>>
>> Something like:
>>
>> ' Project A
>>
>> Public Interface InterfaceA
>> Sub SomeMethod()
>> End Interface
>>
>> Public Class Something
>>
>> Sub DoSomething(anA As InterfaceA)
>> anA.SomeMethod()
>> End Sub
>>
>> End Class
>>
>> ' Project B
>> ' references Project A
>>
>> Public Class SomethingElse
>> Implements InterfaceA
>>
>> Sub DoSomething()
>> Dim something As New Something
>> something.DoSomething(Me)
```

Re: cyclic dependency

Re: cyclic dependency

>> End Sub
>>
>> Sub SomeMethod() Implements InterfaceA.SomeMethod
>> ...
>> End Sub
>>
>> End Class
>>
>> --
>> Hope this helps
>> Jay [MVP – Outlook]
>> .NET Application Architect, Enthusiast, & Evangelist
>> T.S. Bradley – <http://www.tsbradley.net>
>>
>>
>> "Pohihihi" <noemail@xxxxxxxxxxxx> wrote in message
>> news:edk95d5BGHA.1544@xxxxxxxxxxxxxxxxxxxxxxxxxxxx
>> Hello,
>>
>> I have 2 project A & B belonging to same solution. Now what I want is
>> that
>> few forms I am accessing from A in B and few I am doing that other way
>> around. What should I do to not get the error while building solution, I
>> mean by the build sequencing.
>>
>> Thanks,
>> Po
>>
>>
>

• **References:**

◆ **[Re: cyclic dependency](#)**

◇ *From:* Nicholas Paldino [.NET/C# MVP]

- Prev by Date: **[Re: Thread safety and events](#)**
- Next by Date: **[Re: Another question about performance](#)**
- Previous by thread: **[Re: cyclic dependency](#)**
- Next by thread: **[Thread safety and events](#)**
- Index(es):
 - ◆ **[Date](#)**
 - ◆ **[Thread](#)**