

Re: Programaticly call a method from a different object?

Re: Programaticly call a method from a different object?

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.general/2007-01/msg00904.html>

- *From:* "Nick Malik [Microsoft]" <nickmalik@xxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Wed, 31 Jan 2007 00:58:13 -0800
-

You completely missed the point, I'm afraid.

In most cases, reflection is not necessary. A better design removes the need for the crutch. That's not to say that reflection is somehow bad, but when used as an alternative to simple, easily described patterns that use interfaces, declare intent, and are managed by the developer, and not 'self modifying code', reflection can open the door to a wide array of design issues that can make maintenance expensive.

I'm not criticizing the code. I'm expressing reservations about the need to use it.

If a gentleman comes to my door with his arms full of packages, opening the door is polite. If the packages are on fire, opening the door is quite inconsiderate and potentially self-destructive. You have asked me to open the door. I'm smelling smoke.

--- Nick Malik [Microsoft]
Enterprise Application Architect
<http://blogs.msdn.com/nickmalik>

Disclaimer: Opinions expressed in this forum are my own, and not representative of my employer.

I do not answer questions on behalf of my employer. I'm just a programmer helping programmers.

"schneider" <eschneider@xxxxxxxxxxxxxxxxxxxxxx> wrote in message news:e43bjAJRHHA.2076@xxxxxxxxxxxxxxxxxxxxxx

Look Nick the pattern is not the issue, I know what I want to do, and Ben seems to understand, it's a language detail.

Either read the prior messages or scram...

Re: Programaticly call a method from a different object?

"Nick Malik [Microsoft]" <nickmalik@xxxxxxxxxxxxxxxxxxxx> wrote in message
news:aaidnUhMBZ0J7iLYnZ2dnUVZ_uKknZ2d@xxxxxxxxxxxxxxxx

What's wrong with Visitor or Chain of Responsibility? You can set them up at runtime.

--- Nick Malik [Microsoft]
MCSD, CFPS, Certified Scrummaster
<http://blogs.msdn.com/nickmalik>

Disclaimer: Opinions expressed in this forum are my own, and not representative of my employer.
I do not answer questions on behalf of my employer. I'm just a programmer helping programmers.

"schneider" <eschneider@xxxxxxxxxxxxxxxxxxxx> wrote in message
[news:eYUV0o\\$QHHA.4252@xxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:eYUV0o$QHHA.4252@xxxxxxxxxxxxxxxxxxxxxxxx)

No, I know how to do a plug-in design, and already using it.

Trying to wire two methods together during runtime, similar to addhandler but method to be called is discovered by reflection.

Currently using the Ben pattern, I just thought I might be missing cleaner language feature...

Schneider

"Nick Malik [Microsoft]"
<nickmalik@xxxxxxxxxxxxxxxxxxxx> wrote in message
news:0_2dnajF7cO_pyPYnZ2dnUVZ_uKknZ2d@xxxxxxxxxxxxxxxx

What is the objective? Do you want to be able to develop a plug-in for your code, after the system is released, and have your system recognize it?

If so, that's covered loosely by traditional Design Patterns. I would recommend you investigate the Gang of Four Design Patterns, especially Strategy, Visitor, and Chain of Responsibility. (good reference on GoF patterns:
<http://home.earthlink.net/~huston2/dp/patterns.html>

Re: Programaticly call a method from a different object?

)

Once you understand the design patterns that lead to frameworks, go ahead and dive into a light framework like Spring.Net. Here's an article on Dependency Injection in .Net:
<http://msdn.microsoft.com/msdnmag/issues/05/09/DesignPatterns/default.aspx>

---- Nick Malik [Microsoft]
MCSD, CFPS, Certified Scrummaster
<http://blogs.msdn.com/nickmalik>

Disclaimer: Opinions expressed in this forum are my own, and not representative of my employer. I do not answer questions on behalf of my employer. I'm just a programmer helping programmers.

"schneider"
<eschneider@xxxxxxxxxxxxxxxxxxxxxx>
wrote in message
<news:%230j09yVOHHA.488@xxxxxxxxxxxxxxxxxxxxxxxxxx>

I'm looking for a way to programaticly call a method from a different object and associate the two objects at runtime.

Example: Object A exist and is unknow, I want object B to be able to call a method on object A whenever needed.

Like addhandler but I don't know what the method is and must discover it through reflection.

Maybe something like this:

```
Public Sub attach(ByVal valueA As Object)
```

```
Dim meth() As MethodInfo  
=
```

Re: Programaticly call a method from a different object?

```
value.GetType.GetMethods
```

```
Dim btnB As Button = New  
Button
```

```
Dim mitem As MethodInfo  
= meth(0) 'find the method I  
want somehow
```

```
AddHandler btnB.Click,  
mitem.[Selected_Method]  
'???
```

```
End Sub
```

Any ideas would be great.

Thanks,

Schneider