

# Reflection Issues

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2006-06/msg02896.html>

---

- *From:* "Jason" <[Jason.Ipock@xxxxxxxxxxxxx](mailto:Jason.Ipock@xxxxxxxxxxxxx)>
  - *Date:* Mon, 19 Jun 2006 12:29:15 -0400
- 

Hello,

I'm trying to invoke a method using reflection on a DLL that is already within a project. However, I'm having a few problems and didn't know if anyone has any bright ideas. Please note, I've had lots of success with Reflection in the past, so I'm familiar with Reflection a little bit.

I have assembly X and Method Y I want to call. This assembly and related classes and assemblies are very complex to invoke, requiring database connections, complex constructors, etc. Method Y, however, is a property. It will return a simple string value, no parameters needed.

My problem is trying to activate the method. On the final line of code below, I get an error.

```
Type type = assembly.GetType("MainObj");
MethodInfo info = type.GetMethod("GetOperatingSystemVersion");
Object activator = Activator.CreateInstance(type);
object x = info.Invoke(activator, null);
```

The error is an "Argument Null Exception". Any advice on trying to track this down? Ironically, this Assembly is already referenced in the project and I am only using Reflection because I need to programmatically determine which method to call. If I put the `MainObj.GetOperatingSystemVersion()` in my code, I get an appropriate reply. Any advice on debugging this? Or using the Assembly already referenced by my program?

-J