

Podcast #128: We chat with Kent C Dodds about why he loves React and discuss what life was like in the dark days before Git. [Listen now](#).

Could not find a part of the path ... bin\roslyn\csc.exe

Asked 4 years, 1 month ago Active 25 days ago Viewed 318k times



632

I am trying to run Asp.net MVC project retrieved from TFS source control. I have added all assembly references and I am able to build and compile successfully without any error or warning.

But I get the following error in the browser:

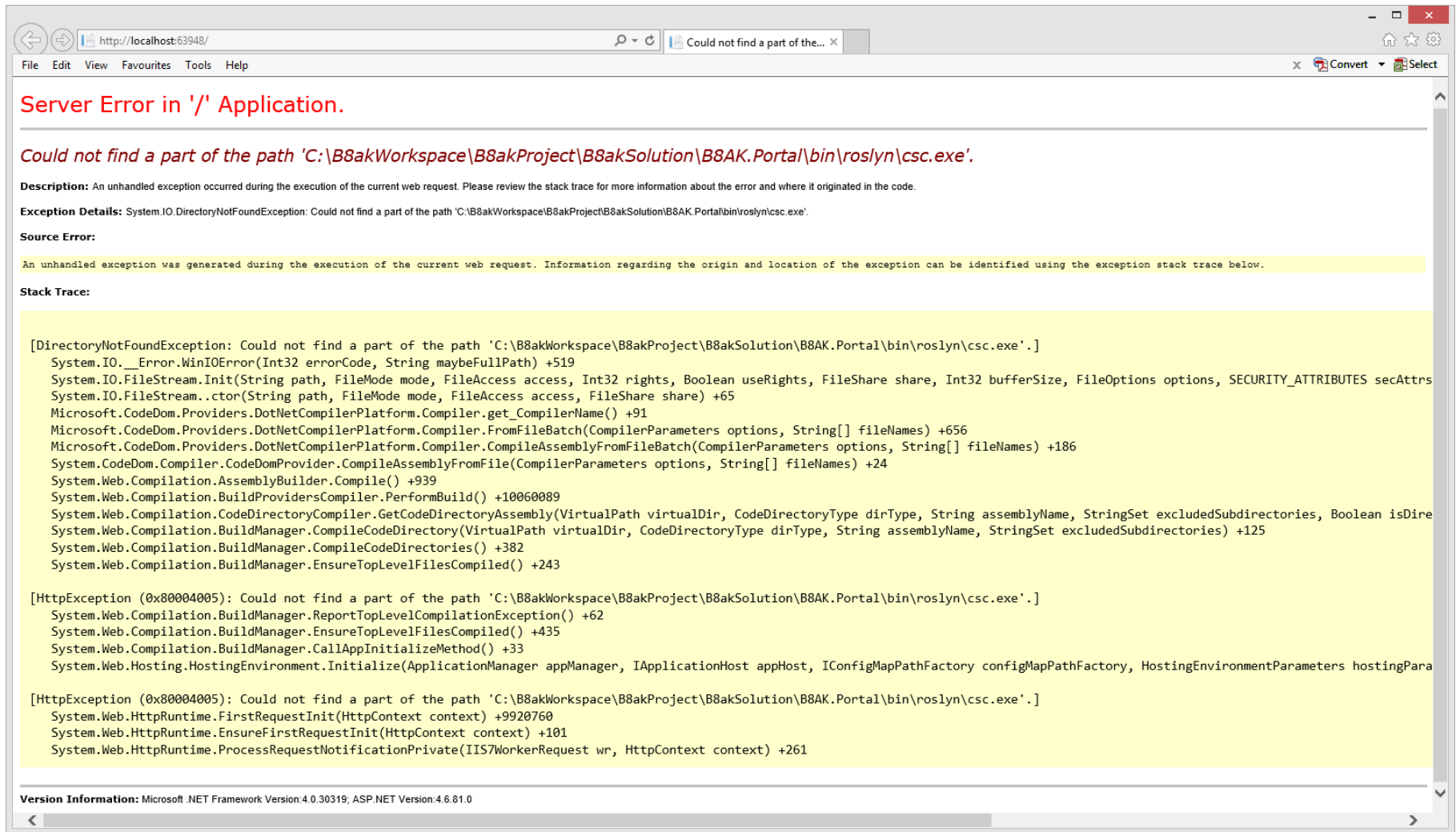


139

Could not find a part of the path 'C:\B8akWorkspace\B8akProject\B8akSolution\B8AK.Portal\bin\roslyn\csc.exe'.

Here is a full screenshot of the error page.

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



After few days of research, I understood that [Roslyn](#) is .Net compiler platform that offers advance compiling features. However, I do not understand why my build is trying to find \bin\roslyn\csc.exe because I did not configure anything related to Roslyn nor I intend to use Roslyn in my project.

c#

asp.net

.net


asp.net-mvc

roslyn

edited Oct 28 at 6:21

asked Sep 25 '15 at 10:36

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

- 10 Can anyone explain why this is necessary as part of running a compiled ASP.NET application? What is it using csc.exe for? – [gregmac](#) Nov 12 '15 at 20:11 
- 1 I think this explains roslyn involvement: blogs.msdn.microsoft.com/webdev/2014/05/12/... – [andy250](#) Feb 10 '16 at 13:09
- 1 roslyn folder was not getting copied in bin folder i fixed it by installing the below Install-Package Microsoft.CodeDom.Providers.DotNetCompilerPlatform – [Abdullah Tahan](#) Oct 12 '18 at 23:24
- 5 Reinstalling Microsoft.CodeDom.Providers.DotNetCompilerPlatform solved my issue. – [SurenSaluka](#) Feb 4 at 5:59
- 1 I was having this after opening my project in VS 2019. It previously worked on VS 2017. I found that simply downgrading Microsoft.CodeDom.Providers.DotNetCompilerPlatform to any previous version then back to the latest version solved the issue. It corrected some issues in my .csproj file. – [Neo](#) May 6 at 14:30

49 Answers

1 2 next



353



The problem with the default VS2015 templates is that the compiler isn't actually copied to the tfr\bin\roslyn\ directory, but rather the {outdir}\roslyn\ directory

Add this code in your .csproj file:

```
<Target Name="CopyRoslynFiles" AfterTargets="AfterBuild"
Condition="!$(Disable_CopyWebApplication) And '$(OutDir)' != '$(OutputPath)'">
  <ItemGroup>
    <RoslynFiles Include="$(CscToolPath)\*" />
  </ItemGroup>
  <MakeDir Directories="$(WebProjectOutputDir)\bin\roslyn" />
  <Copy SourceFiles="@$(RoslynFiles)"
DestinationFolder="$(WebProjectOutputDir)\bin\roslyn" SkipUnchangedFiles="true"
Retries="$(CopyRetryCount)" RetryDelayMilliseconds="$(CopyRetryDelayMilliseconds)" />
</Target>
```

edited Nov 7 '17 at 10:35


answered Sep 25 '15 at 10:44





[Mitchell](#)

4,162 2 12 13

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

Please notice that when I create a new MVC project in VS2015 I don't see the mentioned configuration in .csproj and it runs perfectly fine in the browser – [Eyad](#) Sep 25 '15 at 10:59 

- 4 Thanks. I am now able to build and run the project in the browser after I downloaded the Roslyn directory and placing it in the /bin folder. I did not put the PstBuildEvent mentioned above and it still works. Maybe you want to edit your answer above and mention the need to manually placing the Roslyn files and to better reflect the solution. – [Eyad](#) Sep 25 '15 at 14:20 
- 2 Well, in a normal situation; you should have the compiler inside your \$(OutDir)roslyn*. * folder, so this script will copy the compiler to the binfolder of your project. Apparently your installation of vs2015 did not include the compiler. – [Mitchell](#) Sep 25 '15 at 15:09
- 5 I found updating Microsoft.CodeDom.Providers.DotNetCompilerPlatform to 1.0.8 and Microsoft.Net.Compilers 2.6.1 helped me a lot. I didnt need to add this extra target. Looks like something similar was added in a later version of the tooling: github.com/aspnet/RoslynCodeDomProvider/commit/... – [Ian Robertson](#) Feb 13 '18 at 16:01 
- 3 I have updated the version of Microsoft.Net.Compilers to version 2.10.0 from the Nuget and it has been the solution for me. I'm using targetFramework="4.6.2" – [juanytuweb](#) Nov 16 '18 at 12:39

▲
0

None of the other answers worked for me. After doing a folder compare before/after vs my expected committed files, i discovered that GIT was ignoring a required folder. If you are tracking the compiler in a repository, make sure the BUILD folder is tracked. If it's not, the compiler won't ever be built and will throw this exact error after publish. I added this line to my **.gitignore** file:

▼

```
!*/packages/Microsoft.CodeDom.Providers.DotNetCompilerPlatform.2.0.1/build/
```

and now it's deploying to other computers properly.

edited Oct 17 at 15:37

answered Oct 17 at 15:08



[John Lord](#)

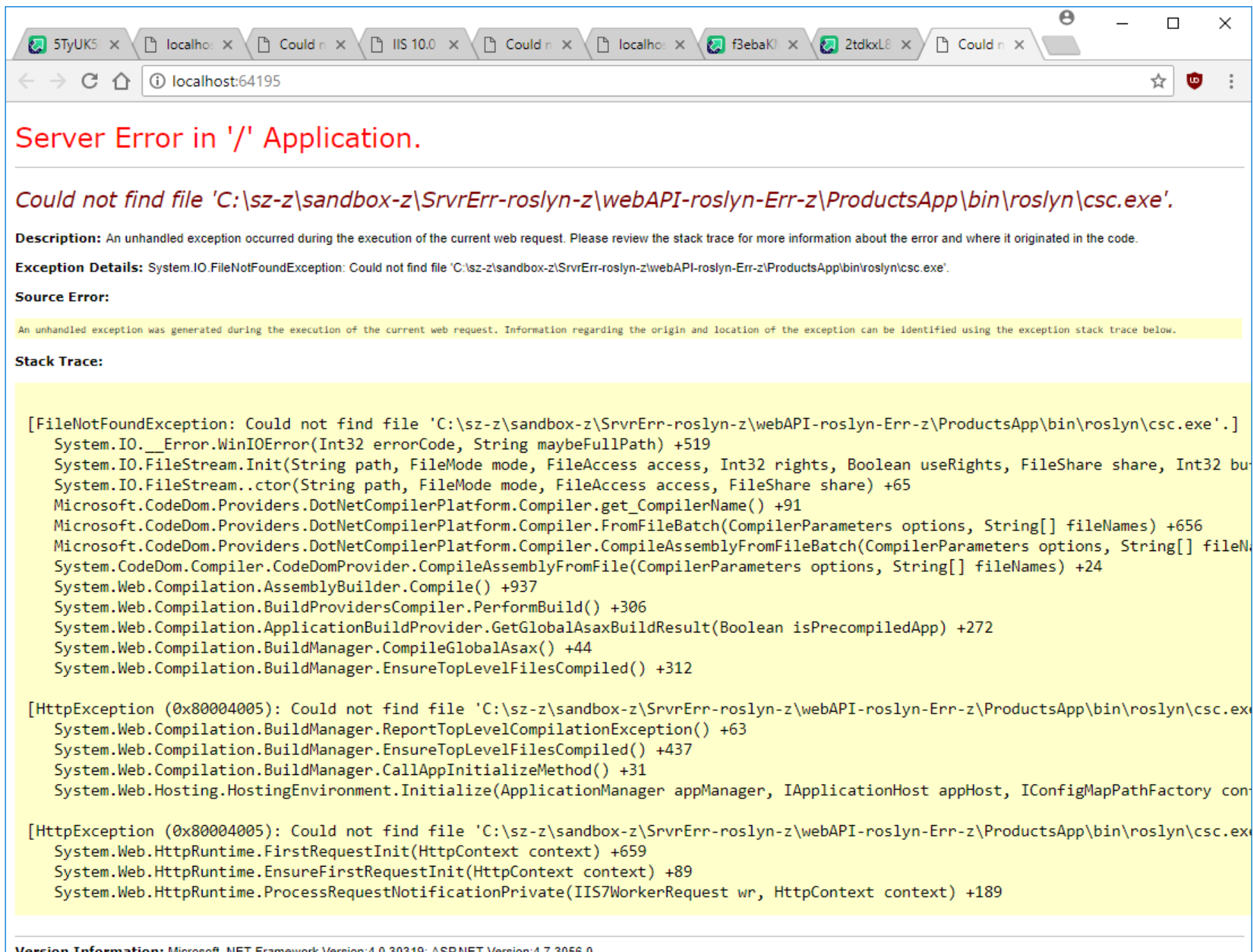
712 1 17

▲
3

First of all, here is code that reproduces the error: <http://schulze.000webhostapp.com/vs/SvrErr-reproduce.zip> (Originally from <https://github.com/aspnet/AspNetDocs/tree/master/aspnet/web-api/overview/advanced/calling-a-web-api-from-a-net-client/sample/server/ProductsApp>)

▼

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



Server Error in '/' Application.

Could not find file 'C:\sz-z\sandbox-z\SrvrErr-roslyn-z\webAPI-roslyn-Err-z\ProductsApp\bin\roslyn\csc.exe'.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.IO.FileNotFoundException: Could not find file 'C:\sz-z\sandbox-z\SrvrErr-roslyn-z\webAPI-roslyn-Err-z\ProductsApp\bin\roslyn\csc.exe'.

Source Error:

An unhandled exception was generated during the execution of the current web request. Information regarding the origin and location of the exception can be identified using the exception stack trace below.

Stack Trace:

```
[FileNotFoundException: Could not find file 'C:\sz-z\sandbox-z\SrvrErr-roslyn-z\webAPI-roslyn-Err-z\ProductsApp\bin\roslyn\csc.exe'.]
  System.IO.__Error.WinIOError(Int32 errorCode, String maybeFullPath) +519
  System.IO.FileStream.Init(String path, FileMode mode, FileAccess access, Int32 rights, Boolean useRights, FileShare share, Int32 bufferSize, FileOptions options, SecurityAttributes securityAttributes) +107
  System.IO.FileStream..ctor(String path, FileMode mode, FileAccess access, FileShare share) +65
  Microsoft.CodeDom.Providers.DotNetCompilerPlatform.Compiler.get_CompilerName() +91
  Microsoft.CodeDom.Providers.DotNetCompilerPlatform.Compiler.FromFileBatch(CompilerParameters options, String[] fileNames) +656
  Microsoft.CodeDom.Providers.DotNetCompilerPlatform.Compiler.CompileAssemblyFromFileBatch(CompilerParameters options, String[] fileNames) +24
  System.Web.Compilation.AssemblyBuilder.Compile() +937
  System.Web.Compilation.BuildProvidersCompiler.PerformBuild() +306
  System.Web.Compilation.ApplicationBuildProvider.GetGlobalAsaxBuildResult(Boolean isPrecompiledApp) +272
  System.Web.Compilation.BuildManager.CompileGlobalAsax() +44
  System.Web.Compilation.BuildManager.EnsureTopLevelFilesCompiled() +312

[HttpException (0x80004005): Could not find file 'C:\sz-z\sandbox-z\SrvrErr-roslyn-z\webAPI-roslyn-Err-z\ProductsApp\bin\roslyn\csc.exe'.]
  System.Web.Compilation.BuildManager.ReportTopLevelCompilationException() +63
  System.Web.Compilation.BuildManager.EnsureTopLevelFilesCompiled() +437
  System.Web.Compilation.BuildManager.CallAppInitializeMethod() +31
  System.Web.Hosting.HostingEnvironment.Initialize(ApplicationManager appManager, IApplicationHost appHost, IConfigMapPathFactory configMapPathFactory) +111

[HttpException (0x80004005): Could not find file 'C:\sz-z\sandbox-z\SrvrErr-roslyn-z\webAPI-roslyn-Err-z\ProductsApp\bin\roslyn\csc.exe'.]
  System.Web.HttpRuntime.FirstRequestInit(HttpContext context) +659
  System.Web.HttpRuntime.EnsureFirstRequestInit(HttpContext context) +89
  System.Web.HttpRuntime.ProcessRequestNotificationPrivate(IIS7WorkerRequest wr, HttpContext context) +189
```

Version Information: Microsoft .NET Framework Version: 4.0.30319; ASP.NET Version: 4.7.3056.0

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

Rather than using package manager (as suggested by andy250/Smolj r) to run

Update-Package Microsoft.CodeDom.Providers.DotNetCompilerPlatform -r

you can remove an attribute of your project's `Web.config` file.
(`Web.config` is in the same directory as your `.csproj` file.)

Open the `Web.config` file in a text editor (or inside Visual Studio).

- In the tag `configuration | system.codedom | compilers | compiler language="c#;cs;csharp"` , completely remove the `type` attribute.

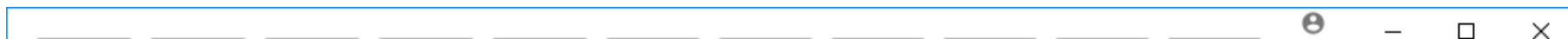
```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <!-- ... -->
  <system.codedom>
    <compilers>
      <compiler language="c#;cs;csharp" extension=".cs"
        type="Microsoft.CodeDom.Providers.DotNetCompilerPlatform.CSharpCodeProvider,
Microsoft.CodeDom.Providers.DotNetCompilerPlatform, Version=1.0.5.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"
        warningLevel="4" compilerOptions="/langversion:default /nowarn:1659;1699;1701"/>
      <compiler language="vb;vbs;visualbasic;vbscript" extension=".vb"
        type="Microsoft.CodeDom.Providers.DotNetCompilerPlatform.VBCodeProvider,
Microsoft.CodeDom.Providers.DotNetCompilerPlatform, Version=1.0.5.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"
        warningLevel="4" compilerOptions="/langversion:default /nowarn:41008
/define:_MYTYPE=\&quot;Web\&quot; /optionInfer+"/>
    </compilers>
  </system.codedom>
</configuration>
```

In short, remove the line that starts with `type="Microsoft.CodeDom.Providers.DotNetCompilerPlatform.CSharpCodeProvider, Microsoft .`

(Presumably, the same fix works for Visual Basic as well as for Csharp, but I have not tried it.)

Visual Studio will take care of the rest. No more `Server Error in '/' Application .`

In the example code I provided in the zip file above you will now get `HTTP Error 403` when you hit `Ctrl + F5`.



By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

HTTP Error 403.14 - Forbidden

The Web server is configured to not list the contents of this directory.

Most likely causes:

- A default document is not configured for the requested URL, and directory browsing is not enabled on the server.

Things you can try:

- If you do not want to enable directory browsing, ensure that a default document is configured and that the file exists.
- Enable directory browsing.
 1. Go to the IIS Express install directory.
 2. Run `appcmd set config /section:system.webServer/directoryBrowse /enabled:true` to enable directory browsing at the server level.
 3. Run `appcmd set config ["SITE_NAME"] /section:system.webServer/directoryBrowse /enabled:true` to enable directory browsing at the site level.
- Verify that the configuration/system.webServer/directoryBrowse@enabled attribute is set to true in the site or application configuration file.

Detailed Error Information:

Module	DirectoryListingModule	Requested URL	http://localhost:64195/
Notification	ExecuteRequestHandler	Physical Path	C:\sz-z\sandbox-z\SrvrErr-roslyn-z\webAPI-roslyn-Err-z\ProductsApp
Handler	StaticFile	Logon Method	Anonymous
Error Code	0x00000000	Logon User	Anonymous

More Information:

This error occurs when a document is not specified in the URL, no default document is specified for the Web site or application, and directory listing is not enabled for the Web site or application. This setting may be disabled on purpose to secure the contents of the server.

[View more information »](#)

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<ArrayOfProduct xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://schemas.datacontract.org/2004/07/ProductStore.Models">
  <Product>
    <Category>Groceries</Category>
    <Id>1</Id>
    <Name>Tomato soup</Name>
    <Price>1.39</Price>
  </Product>
  <Product>
    <Category>Toys</Category>
    <Id>2</Id>
    <Name>Yo-yo</Name>
    <Price>3.75</Price>
  </Product>
  <Product>
    <Category>Hardware</Category>
    <Id>3</Id>
    <Name>Hammer</Name>
    <Price>16.99</Price>
  </Product>
</ArrayOfProduct>
```

As a provocation, I even tried removing the whole `package` directory of my Visual Studio solution. It was automatically and silently recreated as soon as I (re-)built it.

edited Oct 4 at 15:19

answered Oct 4 at 14:45



Henke

99 1 11

I had this issue on the server I was deploying to, and determined that I did not need

2

Microsoft.CodeDom.Providers.DotNetCompilerPlatform

So, I uninstalled it via nuget, and removed the reference in the web config. No more issues.

I originally tried to add target node to the `.proj` file as mentioned in some of the other answers, but that just led to another error where the `msbuild` could not copy the `pagefile.sys` which seemed from what I read to be a bug in the nuget package.

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



Popo

2,327

5

27

50

-
- 1 Thanks for this. That worked for me. Just wondering if you know why DotNetCompilerPlatform was added in the first place, what package usually adds this? – [russelrillema](#) Nov 6 at 4:50
-

▲ Reboot Windows.

4

This is the only solution that worked for me after trying rebuild, delete contents of `bin` and rebuild, restart Visual Studio.

▼ It's yet another example of how terrible C#/.NET build tools are.

I think (after reading many of the answers), the overall conclusion is that the cause and solution of this problem heavily depends on the setup and project, so if one answer does not work, just try another. Try non-intrusive/destructive solutions, such as restarting Visual Studio, rebooting, rebuilding, etc., FIRST, before messing with NuGet packages or reinstalling development tools. Good luck!

(NOTE: Using Visual Studio 2019, and project file was originally created in Visual Studio 2015. Maybe this helps someone investigate the issue)

(EDIT: Could this be caused by not rebooting after installing/modifying the Visual Studio installation or updating Visual Studio when the installer prompts to reboot?)

edited Aug 12 at 14:07

answered Jul 1 at 13:41



Florian Winter

2,419

1

21

35

▲ In my situation, our team don't want to keep 'packages' folder, so we put all dlls in other directory like 'sharedlib'.

0

I used build event to solve this problem.

▼

```
if "$(ConfigurationName)" == "Release" (  
  goto :release  
) else (  
  
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
:release
if not exist $(TargetDir)\roslyn mkdir $(TargetDir)\roslyn

copy /Y "$(ProjectDir)..\..\Shared
Lib\Microsoft.CodeDom.Providers.DotNetCompilerPlatform.2.0.1\tools\Roslyn45\*"
"$(TargetDir)\roslyn"
goto :exit

:exit
```

answered Jul 26 at 3:23

[Ray Chung](#)

11 2



Other than deleting the Bin directory from all projects inside the solution, delete the obj folders too.

3

In the main solution directory remove the folder .vs



Worked for me when trying to bring an already done project into a blank solution created on git.

edited Jul 1 at 18:39

answered Mar 13 '18 at 17:11

[Erik Silva](#)

31 5



In my case by just Deleting everything inside the bin folder and recompiling did all the work for me.

5

Good luck to anyone having this problem.



answered May 9 at 20:26

[Juan Martí](#)

81 2 3

2 I think this is more simple and effective. Usually I get this error after cloning a project from a repository into a new computer. – [Jonathan Ortega](#) May 24 at 14:29

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



27



As noted in [an issue in the Roslyn project on GitHub](#), a solution (that worked for me) is to simply unload and reload the project in Visual Studio.

The "bin\roslyn" folder wasn't created on build or rebuild until I reloaded the project.

answered May 3 at 7:23



[Christian Davén](#)

12k 9 49 64

7 Can someone get this man a chicken dinner, please? – [joelmddev](#) Jun 7 at 5:44



1



A lot of these answers are referring to the Nuget packages and/or cleaning and reloading your project.

If you have WCF service references and invalid endpoints, you can also get this error message. Make sure your endpoints are correct and update the service configuration with the correct endpoint in the .config and when you configure the service reference from the GUI.

- Service Reference Settings

Client

Address:

Access level for generated classes: Public

☒ Allow generation of asynchronous operations

☒ Generate task-based operations

☐ Generate asynchronous operations

Data Type

☐ Always generate message contracts

Collection type: System.Array

Dictionary collection type: System.Collections.Generic.Dictionary

☒ Reuse types in referenced assemblies

☒ Reuse types in all referenced assemblies

☐ Reuse types in specified referenced assemblies:

- ☐ Antlr3.Runtime
- ☐ Microsoft.CodeDom.Providers.DotNetCompilerPlatform
- ☐ Microsoft.CSharp
- ☐ Microsoft.Web.Infrastructure
- ☐ mscorlib
- ☐ Newtonsoft.Json
- ☐ PagedList
- ☐ PagedList.Mvc
- ☐ RestSharp

OK Cancel

answered Apr 30 at 17:13



GibraltarTop

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



111



A clean and rebuild worked for me!

edited Apr 22 at 17:25



Community ♦

1 1

answered Jan 5 '17 at 11:55



[pipedreambomb](#)

1,534 1 13 15

-
- 4 I dont think the clean is required. According to this discussion of the issue a **rebuild** not a regular build will always put the roslyn file back.
github.com/dotnet/roslyn/issues/15556 – [leemicw](#) Mar 28 '17 at 13:26 ✎
-
- 8 Just rebuild didn't worked for me.. After Clean + Rebuild the error disappeared. – [Bruno Miquelin](#) Oct 22 '18 at 13:19
-
- 1 DotNetCompilerPlatform 1.0.3, Microsoft.Net.Compilers 1.3.2, VS Pro 2017 15.9.4 here. A clean/rebuild didn't work for me, even before and after restarting Visual Studio. Finally, a Build > Batch Build... > Rebuild All did the trick. It must have whispered just the right sweet-nothing to get VS to see it was missing the bin/roslyn directory in the output. – [Johann](#) Dec 14 '18 at 19:26 ✎
-



12



1. Clean Solution

2. Rebuild Solution ,These two steps worked for me.

answered Apr 2 at 18:42



[nisha](#)

136 1 3

-
- 1 Works. I accidentally pressed `Ctrl C` when I was in the middle of checking out a branch in `git` and it screwed my repo up. `git reset --hard` didn't work, so I `git clean -xdf` and had to rebuild the project. However I ran into this error so I simply cleaned and rebuilt the project again and it worked for me. – [Paul Carlton](#) Jun 4 at 21:46 ✎
-



8



In my case I have had issue in Jenkins when it tried to deploying it in Octopus with following error:

```
MSBUILD : OctoPack error OCT-1676060969: Failed to build the path for
'\bin\roslyn\csc.exe' relative to 'T:\workspace\machine.engine\Machine.engine.Test':
Invalid URI: The format of the URI could not be determined.. See the inner exception for
more details.
[T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj]
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```

more details. ---> System.UriFormatException: Invalid URI: The format of the URI could
not be determined.
[T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj]
MSBUILD : OctoPack error OCT-1676060969: at System.Uri.CreateThis(String uri, Boolean
dontEscape, UriKind uriKind)
[T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj]
MSBUILD : OctoPack error OCT-1676060969: at System.Uri..ctor(String uriString)
[T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj]
MSBUILD : OctoPack error OCT-1676060969: at
OctoPack.Tasks.Util.OctopusPhysicalFileSystem.GetPathRelativeTo(String fullPath, String
relativeTo) in
Z:\buildAgent\workDir\20ba9f2e0d5e4022\source\OctoPack.Tasks\Util\OctopusPhysicalFileSyste
211 [T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj]
MSBUILD : OctoPack error OCT-1676060969: --- End of inner exception stack trace ---
[T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj]
MSBUILD : OctoPack error OCT-1676060969: at
OctoPack.Tasks.Util.OctopusPhysicalFileSystem.GetPathRelativeTo(String fullPath, String
relativeTo) in
Z:\buildAgent\workDir\20ba9f2e0d5e4022\source\OctoPack.Tasks\Util\OctopusPhysicalFileSyste
224 [T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj]
MSBUILD : OctoPack error OCT-1676060969: at
OctoPack.Tasks.CreateOctoPackPackage.AddFiles(XContainer nuSpec, IEnumerable`1
sourceFiles, String sourceBaseDirectory, String targetDirectory, String relativeTo) in
Z:\buildAgent\workDir\20ba9f2e0d5e4022\source\OctoPack.Tasks\CreateOctoPackPackage.cs:line
443 [T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj]
MSBUILD : OctoPack error OCT-1676060969: at
OctoPack.Tasks.CreateOctoPackPackage.Execute() in
Z:\buildAgent\workDir\20ba9f2e0d5e4022\source\OctoPack.Tasks\CreateOctoPackPackage.cs:line
190 [T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj]
Done Building Project
"T:\workspace\machine.engine\Machine.engine.Test\Machine.engine.Test.csproj" (default
targets) -- FAILED

```

Cause

After spending some time, I was using an internal developed component that was using `Microsoft.Net.Compilers`. The reason the internal component was using `Microsoft.Net.Compilers` was to overcome this issue ([C#: throw invalid expression compilation](#)) and was solved this way ([How to use C# 7 with Visual Studio 2015?](#)). This result in, when I installed the component on the main program, the `Microsoft.Net.Compilers` get added it selves automatically.

Solution

```

"Microsoft.Net.Compilers" <PackageReference Include="Microsoft.Net.Compilers" Version="1.2.3" />

```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
PM> Uninstall-package Microsoft.CodeDom.Providers.DotNetCompilerPlatform
PM> Uninstall-package Microsoft.Net.Compilers
```

And chose C# 7 compiler in Jenkins instead of C# 6 and rebuild, this is to ensure everything is working and building correctly.

Than finally in my main program I tried to update my internal component. And everything than build again. It has built without any problems or issues.

edited Mar 12 at 21:09

answered Feb 2 '18 at 11:30



maytham-waeyliew

14.8k 7 55 81



0



The answer for this is different for Website project and Web application Project. The underlying issue is same that NuGet package is behaving differently on different machine. It may be rights issue or some execution policy which stops it from copying to Bin folder As you know Roslyn is new compiler. you should have it in Bin folder for these Projects Go to your website NuGet Packages check this Folder Microsoft.CodeDom.Providers.DotNetCompilerPlatform.2.0.0
 \code\packages\Microsoft.CodeDom.Providers.DotNetCompilerPlatform.2.0.0 Do you see it ? Can you see
 code\packages\Microsoft.CodeDom.Providers.DotNetCompilerPlatform.2.0.0\tools\RoslynLatest in it Now as part of compiling this Folder should get copied to your website under bin like this. \code\WebSite1\Bin\Roslyn some how that is not happening for you . Try running Visual studio as Admin . Copy Roslyn folder manually. Try uninstall and install of NuGet Package. Remember this package compile your folder and if its not there you cannot compile anything and so you cannot add anything too. Try copying this package to offline version tools -> options-> nuget package Manager->Package source-> Microsoft Visual Studio Offline Packages C:\Program Files (x86)\Microsoft SDKs\NuGetPackages

answered Feb 28 at 19:08



Jin Thakur

1,462 11 10



14



- Right click on your project and select Manage Nuget Packages
- Find "Microsoft.CodeDom.Providers.DotNetCompilerPlatform"
- Simply Update to an older or newer version (doesn't matter which), and then update again back to your original version.

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

Microsoft.CodeDom.Providers.DotNetCompilerPlatform

Microsoft

23.5M

downloads

Replacement CodeDOM providers that use the new .NET Compiler Platform ("Roslyn") compiler as a service APIs.

v2.0.0

v2.0.1

Microsoft

69.9K

downloads

Microsoft.CodeDom.Providers.DotNetCompilerPlatform.BinFix

If you tried to install 'Microsoft.CodeDom.Providers.DotNetCompilerPlatform' package into non-APS.NET project you most likely got an 'Could not find a part of the path ... bin\roslyn\csc.exe' error.

v1.0.0

Microsoft

10.2K

downloads

Microsoft.CodeDom.Providers.DotNetCompilerPlatform.NoInstall

Replacement CodeDOM providers that use the new .NET Compiler Platform ("Roslyn") compiler as a service APIs.

v1.0.3.2

Microsoft

2.0.0

Uninstall

Version

Latest stable 2.0.1

Update

Options

Description

Replacement CodeDOM providers that use the new .NET Compiler Platform ("Roslyn") compiler as a service APIs. This provides support for new language features in systems using CodeDOM (e.g. ASP.NET runtime compilation) as well as improving the compilation performance of these systems.

This package was built from the source at <https://github.com/aspnet/...>

answered Feb 26 at 15:55



Bojan

163 1 5



In my case I had this issue when i was running two visual studio IDE simultaneously. So the solution was to clean the project and close the other instance.

2



answered Feb 12 at 23:37



Badr Bellaj

4,011 1 20 21



I followed these steps and it worked perfectly

22



- Delete all the bin and obj folders
- Clean solution and rebuild
- Run this command in powershell

Update-Package Microsoft.CodeDom.Providers.DotNetCompilerPlatform -r

edited Jan 25 at 16:54

answered Jan 24 at 7:40

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



827



In my case the solution was to re-install/upgrade Nuget packages:

- Microsoft.Net.Compilers 1.1.1
- Microsoft.CodeDom.Providers.DotNetCompilerPlatform 1.0.1

Then I looked into .csproj and made sure that the paths to packages are correct (in my case ..\..\packages*.*) inside tags

<ImportProject> on top and in <Target> with name "EnsureNuGetPackageBuildImports" on the bottom. This is on MVC 5 and .NET Framework 4.5.2.

Short answer - run this in the Package Manager Console:

```
Update-Package Microsoft.CodeDom.Providers.DotNetCompilerPlatform -r
```

edited Nov 22 '18 at 11:24



Gabriel Smoljář

1,132 2 12 27

answered Dec 21 '15 at 8:23



andy250

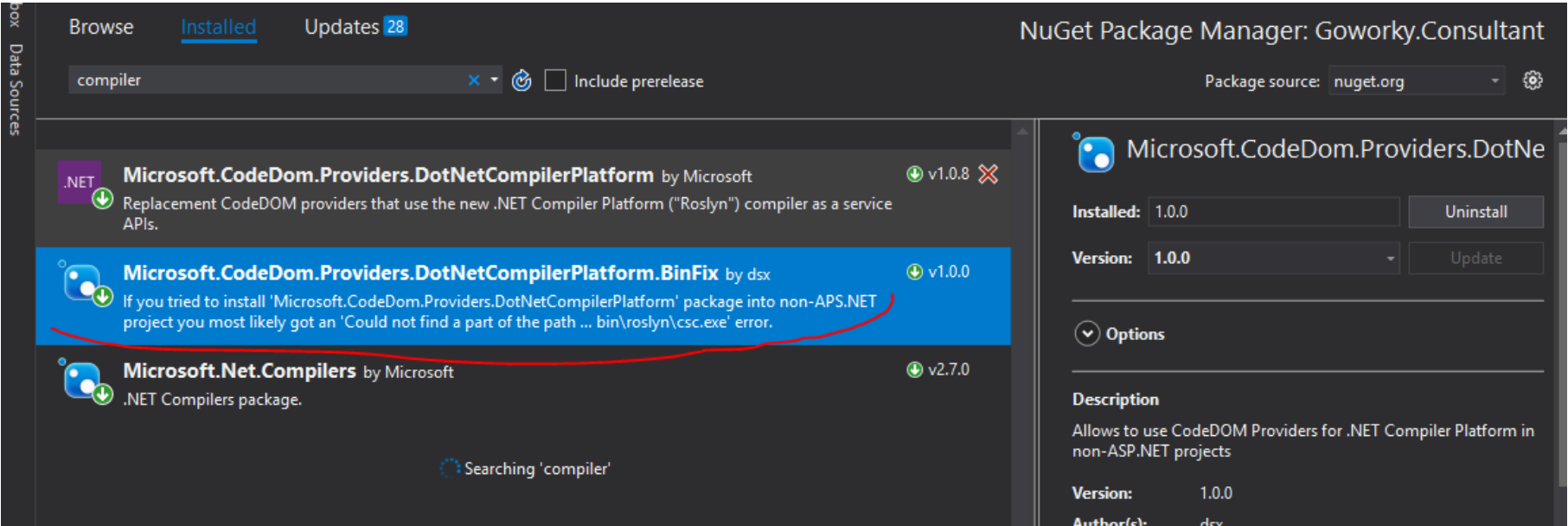
8,657 2 7 19

- 9 This was my problem - the bin/roslyn folder is there when a project is created, however, if you delete it or, like source control, it is not copied, then it doesn't get rebuilt. I think there is a bit of a "sync" issue with the versions, once 1.0.1 is installed and update the Import in proj file to correct version, a build will then automatically copy the Roslyn folder - no need for any of those post build commands. – [Jester](#) Jan 3 '16 at 20:06
- 14 I'm pretty sure this is the best solution...trying this myself with an update-package -reinstall -projectname myprojectname – [cloudstrifebro](#) Jun 2 '16 at 21:18
- 4 After doing all of this, nothing changed for my project. The bin folder is still flattened. – [brianary](#) Oct 20 '16 at 21:02
- 8 One note: version 1.0.3 of the Microsoft.CodeDom.Providers.DotNetCompilerPlatform Nuget package works for me, but version 1.0.6 causes the error in this question. – [Daniel Neel](#) Aug 4 '17 at 21:41
- 5 1.0.6 has a bug: github.com/aspnet/RoslynCodeDomProvider/issues/13 – [akatakritos](#) Aug 14 '17 at 12:50



15

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



You need to install Microsoft.CodeDom.Providers.DotNetCompilerPlatform.BinFix, was especially created for that error

edited Oct 23 '18 at 9:17



OffHeGoes

1,519 18 41

answered Mar 19 '18 at 13:57



Adrian Berca

183 1 4

7 I am not installing random packages from random person with 'dsx' as nickname. That is big security no... — Mateusz Apr 12 at 21:27



After trying all of the fixes with no cigar I fixed it by updating this Nuget Package in Visual Studios:

21

Microsoft.CodeDom.Providers.DotNetCompilerPlatform



Mine was from 1.0.0 to 2.0.0 for reference (The error no longer shows)

answered Jul 12 '18 at 14:34



josh.thomson

845 7 23

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

2 This solved it for me. I downgraded which must've fixed the issue and then I updated back to the latest. – [Daniel Jackson](#) Oct 9 '18 at 18:56

I have webproject without csproj file and solutions mentiond here did not work for me.

3

Changing target .NET framework, reinstalling packages (`Update-Package -reinstall`) and then building the project worked for me. You can even change target framework back after this operation(make suere you reinstall nuget packages again after).

answered Mar 15 '18 at 12:20



[Miroslav Adamec](#)

554 6 18

I had this error for `Microsoft.CodeDom.Providers.DotNetCompilerPlatform 1.0.6` but also with `1.0.7` that worked for @PrisonerZERO. However when Microsoft released `1.0.8` 2017-10-18 it finally started working for me again and I did not have to downgrade.

1

<https://www.nuget.org/packages/Microsoft.CodeDom.Providers.DotNetCompilerPlatform/>

answered Feb 14 '18 at 13:02



[Ogglas](#)

22.6k 12 133 171

my solution is using Nuget to update below items to latest version: - Microsoft.Net.Compilers - Microsoft.CodeDom.Providers.DotNetCompilerPlatform Then rebuilt the project. Since my project is a website so no *.csproj file. The error above appears when I tried to view a cshtml in browser.

1

The error fixed after the two items above updated to latest version. I am in VS2015 and windows7 SP1

answered Jan 28 '18 at 10:09



[Penny](#)

363 2 12

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

What is Roslyn and Who added them(packages) in project : If you're using .net Framework 4.5.2 to create projects using VS2015, you might have noticed that the project templates use Roslyn by default. Actually Roslyn is one of [open-source](#) compilers for .NET languages from Microsoft.

Why should we delete Roslyn : If your project has Roslyn references and you are interested to deploy it no server, you will get unwanted errors on the website as many hosting providers still have not upgraded their servers and hence do not support Roslyn. To resolve this issue, you will need to remove the Roslyn compiler from the project template.

if you are not interested in using Roslyn, follow steps bellow to delete it

1. Remove Nuget packages, use the following commands from Nuget Package Console

```
PM> Uninstall-package Microsoft.CodeDom.Providers.DotNetCompilerPlatform
PM> Uninstall-package Microsoft.Net.Compilers
```


2. After you do this, your web.config file should be auto-updated. In case it is not, look for the below code in web.config file and if it is found, delete this piece of code.

```
<system.codedom>
  <compilers>
    <compiler language="c#;cs;csharp" extension=".cs"
type="Microsoft.CodeDom.Providers.DotNetCompilerPlatform.CSharpCodeProvider,
Microsoft.CodeDom.Providers.DotNetCompilerPlatform, Version=1.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" warningLevel="4" compilerOptions="/langversion:6
/nowarn:1659;1699;1701"></compiler>
    <compiler language="vb;vbs;visualbasic;vbscript" extension=".vb"
type="Microsoft.CodeDom.Providers.DotNetCompilerPlatform.VBCodeProvider,
Microsoft.CodeDom.Providers.DotNetCompilerPlatform, Version=1.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" warningLevel="4" compilerOptions="/langversion:14
/nowarn:41008 /define:_MYTYPE=\&quot;Web\&quot; /optionInfer+"></compiler>
  </compilers>
</system.codedom>
```

edited Jan 3 '18 at 7:35

answered Jul 29 '16 at 22:20

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

-
- 24 This isn't really a solution if you actually *want* to use the new compiler and the new features. – [Matti Virkkunen](#) Apr 11 '17 at 12:11
-
- 14 You're advocating against the future. – [cchamberlain](#) Apr 12 '17 at 0:56
-
- 2 @cchamberlain why is it the future? I just think it should be straightforward to use, but it looks many people are having troubles with it. – [Alisson](#) Dec 26 '17 at 2:09
-
- 1 @Alisson - Roslyn is the direction things are going in. It contains newer language features, is more performant, cross-platform, and open source. It came out after the other tools - hence future. Nothing says you need to use it, most upgrades incur some cost. See the "Why Roslyn compilation in ASP.NET?" section: blogs.msdn.microsoft.com/webdev/2014/05/12/... – [cchamberlain](#) Jan 3 '18 at 6:26 
-

▲ This can be done in the following easy way:-

0

- Create a new project of similar type anywhere in your system. Build it and copy over the roslyn folder to your bin directory.

answered Nov 26 '17 at 17:20



[Sriman Saswat Suvankar](#)

365 3 13

▲ FYI...

1

As of 8/31/2017 upgrading to Microsoft.CodeDom.Providers.DotNetCompilerPlatform 1.0.7 is works.

answered Aug 31 '17 at 18:28



[Prisoner ZERO](#)

11.9k 17 71 113

▲ I had the same problem after updating DotNetCompilerPlatform. Solved by Restarting Visual Studio > Clean Project > Build Project.

1

answered Aug 23 '17 at 17:26



[Uhha](#)

21 3

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



10

This is a [known issue](#) with [Microsoft.CodeDom.Providers.DotNetCompilerPlatform](#) 1.0.6. Downgrading to 1.0.5 fixed this for me.

edited Aug 16 '17 at 15:36

answered Aug 11 '17 at 13:25

[jrummell](#)

39.7k 14 103 161



1

I ran into this problem after updating some packages through NuGet. A rebuild (instead of a normal build) has worked for me.

answered Aug 14 '17 at 9:45

[Helen](#)

11 3



8

Per a comment by Daniel Neel above :

version 1.0.3 of the Microsoft.CodeDom.Providers.DotNetCompilerPlatform Nuget package works for me, but version 1.0.6 causes the error in this question

Downgrading to 1.0.3 resolved this issue for me.

answered Aug 9 '17 at 15:43

[Jason Coyne](#)

5,900 7 32 62

3 1.0.6 has a bug: github.com/aspnet/RoslynCodeDomProvider/issues/13 – [akatakritos](#) Aug 14 '17 at 12:51

2 1.0.7 is still affected in certain scenarios github.com/aspnet/RoslynCodeDomProvider/issues/17 – [altso](#) Aug 18 '17 at 16:51

1 1.0.5 is the most recent version that worked for me (1.0.6 and 1.0.7 are generating the error) – [Patrick](#) Sep 26 '17 at 10:30

1 2 next

Thank you for your interest in this question. Because it has attracted low-quality or spam answers that had to be removed, posting an answer now requires 10 [reputation](#) on this site (the [association bonus does not count](#)).

Would you like to answer one of these [unanswered questions](#) instead?

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).