

Introduction

This Unity editor extension (<https://www.assetstore.unity3d.com/#!/content/63294>) provides T4 text templates (<https://msdn.microsoft.com/en-US/library/bb126445.aspx>) processor. T4 template is a mixture of text blocks and control logic that can generate a text file. The control logic is written as fragments of program code in C#. The generated file can be text of any kind, such as resource file, source code or web page.

T4 template example:

```
<html><body>
    The date and time now is: <#= DateTime.Now #>
</body></html>
```

Use cases

- Repeating source code generation
- Project resources catalogization
(Assets/Editor/GameDevWare.TextTranform/Examples/FileList_Example.tt)
- Resource loading/reading code generation
((Assets/Editor/GameDevWare.TextTranform/Examples/ResourceAsyncLoad_Example.tt)
- Code generation by DSL (Assets/Editor/GameDevWare.TextTranform/Examples/DSL_Example.tt)
- Code generation for ORM (BLToolkit for example)
- Embedding environment information into project's build
(Assets/Editor/GameDevWare.TextTranform/Examples/EnvironmentInfo_Example.tt)

How to use

Create or copy file with **.tt** extension. Select this file in Project window (Window -> Project), then in Inspector window (Window -> Inspector) setup T4 template's parameters. Click "Generate" button. Inspector window for T4 template contains following parameters:

- **Output Type** – type of generated file
 - **Content** – normal template's output. It corresponds to "Design-time T4 text templates" in Microsoft's terminology.
 - **Generator** – generator-class which can generate content when TransformText() is called. It corresponds to "Run time T4 text templates" in Microsoft's terminology.
- **Output Path** – path to generated file. If not specified, generated file will have file name of template and file extension from *output* directive.

- **Auto-Gen Triggers** – list of events which trigger auto-generation.
 - **Code Compilation** – after each code compilation
 - **Asset Changes** – after watched assets are changed, look for **Assets to Watch**
- **Auto-Gen Delay (Ms)** – delay before triggered auto-generation starts
- **Assets to Watch** – list of assets and folders which trigger auto-generation

Details

T4 template can use *hostspecific=true* property (https://msdn.microsoft.com/en-us/library/bb126478.aspx#Anchor_4) to access *Host.ResolvePath* method, which maps path relatively to template's location.

By default *UnityEngine.dll* and *UnityEditor.dll* assemblies are referenced in all templates. You can reference project's assemblies *Assembly-CSharp* and *Assembly-CSharp-firstpass* by adding **assembly** directive (https://msdn.microsoft.com/en-us/library/bb126478.aspx#Anchor_3):

```
<#@ assembly name="Assembly-CSharp" #>
| <#@ assembly name="Assembly-CSharp-firstpass" #>
```

MSBuild Macros (<https://msdn.microsoft.com/en-US/library/c02as0cs.aspx>) are not available.

You can run template generation from your code with **UnityTemplateGenerator.RunForTemplate(templatePath)** call.

Contacts

Please send any questions at support@gamedevware.com

License

If you embed this package, you **MUST** provide a link (<https://www.assetstore.unity3d.com/#!/content/63294>) and warning about embedded package in the description of your package.

Asset Store Terms of Service and EULA ([LICENSE.md](#))

