Skip to main content

ThisAssembly by Daniel Cazzulino

Details

Info

> **INFO**
>
> Author: Daniel Cazzulino
>
> NuGet: https://www.nuget.org/packages/ThisAssembly
>
> You can find more details at https://www.clarius.org/ThisAssembly/
>
> Source : https://github.com/devlooped/ThisAssembly

About

> **NOTE**
>
> The ThisAssembly.Info allows you access to the Assembly Information as constants, instead of going to reflection each time.
>
> I found useful to see the assembly version right away in any project that I have.

How to use

Example (source csproj, source files)

CSharp Project | Program.cs

This is the CSharp Project that references ThisAssembly

```
<Project Sdk="Microsoft.NET.Sdk">
    <PropertyGroup>
        <OutputType>Exe</OutputType>
        <TargetFramework>netcoreapp7.0</TargetFramework>
    </PropertyGroup>
    <PropertyGroup>
        <EmitCompilerGeneratedFiles>true</EmitCompilerGeneratedFiles>

<CompilerGeneratedFilesOutputPath>$(BaseIntermediateOutputPath)\GeneratedX</CompilerG
    </PropertyGroup>
    <PropertyGroup>
        <Version>2023.5.7.800</Version>
    </PropertyGroup>
    <ItemGroup>
        <PackageReference Include="ThisAssembly" Version="1.2.14" OutputItemType="Ana
ReferenceOutputAssembly="false">
            <PrivateAssets>all</PrivateAssets>
            <IncludeAssets>runtime; build; native; contentfiles; analyzers; buildtran
        </PackageReference>
    </ItemGroup>
</Project>
```

This is the use of ThisAssembly in Program.cs

```
var strVersion = ThisAssembly.Info.Version;
System.Console.WriteLine(strVersion);
```

Generated Files

Those are taken from $(BaseIntermediateOutputPath)\GeneratedX

ThisAssembly.AssemblyInfo.g.cs **Branch.g.cs** **Commit.g.cs** **Root.g.cs**

Sha.g.cs **Url.g.cs** **ThisAssembly.Metadata.g.cs**

ThisAssembly.Property.g.cs **ThisAssembly.Resources.EmbeddedResource.cs**

ThisAssembly.Strings.g.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

using System.CodeDom.Compiler;
using System.Runtime.CompilerServices;

/// <summary>
/// Provides access to the current assembly information as pure constants,
```

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
using System.Globalization;

partial class ThisAssembly
{
    public static partial class Git
    {
        /// <summary>
        /// => @"[pending build]"
        /// </summary>
        public const string Branch = @"[pending build]";
    }
}
```

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
```

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
using System.Globalization;

partial class ThisAssembly
{
    public static partial class Git
    {
        /// <summary>
        /// => @"[pending build]"
        /// </summary>
        public const string Root = @"[pending build]";
    }
}
```

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
```

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
using System.Globalization;

partial class ThisAssembly
{
    public static partial class Git
    {
        /// <summary>
        /// => @"[pending build]"
        /// </summary>
        public const string Url = @"[pending build]";
    }
}
```

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

using System.CodeDom.Compiler;
using System.Runtime.CompilerServices;
```

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

using System.CodeDom.Compiler;
using System.Runtime.CompilerServices;

/// <summary>
/// Provides access to the current assembly information as pure constants,
///  without requiring reflection.
/// </summary>
partial class ThisAssembly
{
    /// <summary>
    /// Gets the project properties.
    /// </summary>
    [GeneratedCode("ThisAssembly.Project", "1.2.14")]
    [CompilerGenerated]
    public static partial class Project
    {
        /// <summary>AssemblyName = RSCG_Version</summary>
        public const string AssemblyName = @"RSCG_Version";

        /// <summary>RootNamespace = RSCG_Version</summary>
        public const string RootNamespace = @"RSCG_Version";

        /// <summary>TargetFrameworkIdentifier = .NETCoreApp</summary>
        public const string TargetFrameworkIdentifier = @".NETCoreApp";

        /// <summary>TargetFrameworkMoniker = .NETCoreApp,Version=v7.0</summary>
        public const string TargetFrameworkMoniker =
```

```
using System;
using System.IO;
using System.Linq;
using System.Reflection;

static class EmbeddedResource
{
    static readonly string baseDir =
Path.GetDirectoryName(Assembly.GetExecutingAssembly().Location) ?? "";

    public static string GetContent(string relativePath)
    {
        using var stream = GetStream(relativePath);
        using var reader = new StreamReader(stream);
        return reader.ReadToEnd();
    }

    public static byte[] GetBytes(string relativePath)
    {
        using var stream = GetStream(relativePath);
        var bytes = new byte[stream.Length];
        stream.Read(bytes, 0, bytes.Length);
        return bytes;
    }

    public static Stream GetStream(string relativePath)
    {
        var filePath = Path.Combine(baseDir, Path.GetFileName(relativePath));
        if (File.Exists(filePath))
            return File.OpenRead(filePath);

        var baseName = Assembly.GetExecutingAssembly().GetName().Name;
        var resourceName = relativePath
            .TrimStart('.')
            .Replace('/', '.')
            .Replace('\\', '.');
```

```
using System.Collections.Concurrent;
using System.Resources;
using System.Runtime.CompilerServices;

/// <summary>
/// Provides access to the current assembly information as pure
constants,
///  without requiring reflection.
/// </summary>
partial class ThisAssembly
{
    /// <summary>
    /// Access the strings provided by resource files in the project.
    /// </summary>
    [CompilerGenerated]
    public static partial class Strings
    {
        static ConcurrentDictionary<string, ResourceManager>
resourceManagers = new ConcurrentDictionary<string, ResourceManager>();

        static ResourceManager GetResourceManager(string resourceName)
            => resourceManagers.GetOrAdd(resourceName, name => new
ResourceManager(name, typeof(Strings).Assembly));
    }
}
```

Usefull

Download Example

Download Example ThisAssembly

Download PDF

Download PDF ThisAssembly

Share this page

https://ignatandrei.github.io/RSCG_Examples/v2/docs/ThisAssembly