Loxodon Framework Fody

(中文版)

Developed by Clark

Requires Unity 2018.4 or higher.

This is a plugin for static weaving code that integrates Fody into a Unity project.

PropertyChanged.Fody：Injects code which raises the PropertyChanged event, into property setters of classes which implement INotifyPropertyChanged.

ToString.Fody：Generates ToString method from public properties for class decorated with a [ToString] Attribute.

Installation

Install via OpenUPM (recommended)

OpenUPM can automatically manage dependencies, it is recommended to use it to install the framework.

Requires nodejs's npm and openupm-cli, if not installed please install them first.

```
# Install openupm-cli,please ignore if it is already installed.
npm install -g openupm-cli

#Go to the root directory of your project
cd F:/workspace/New Unity Project

#Install loxodon-framework-fody-propertychanged
openupm add com.vovgou.loxodon-framework-fody-propertychanged

#Install loxodon-framework-fody-tostring
openupm add com.vovgou.loxodon-framework-fody-tostring
```

Install via Packages/manifest.json

Modify the Packages/manifest.json file in your project, add the third-party repository "package.openupm.com"'s configuration and add "com.vovgou.loxodon-framework-fody" in the "dependencies" node.
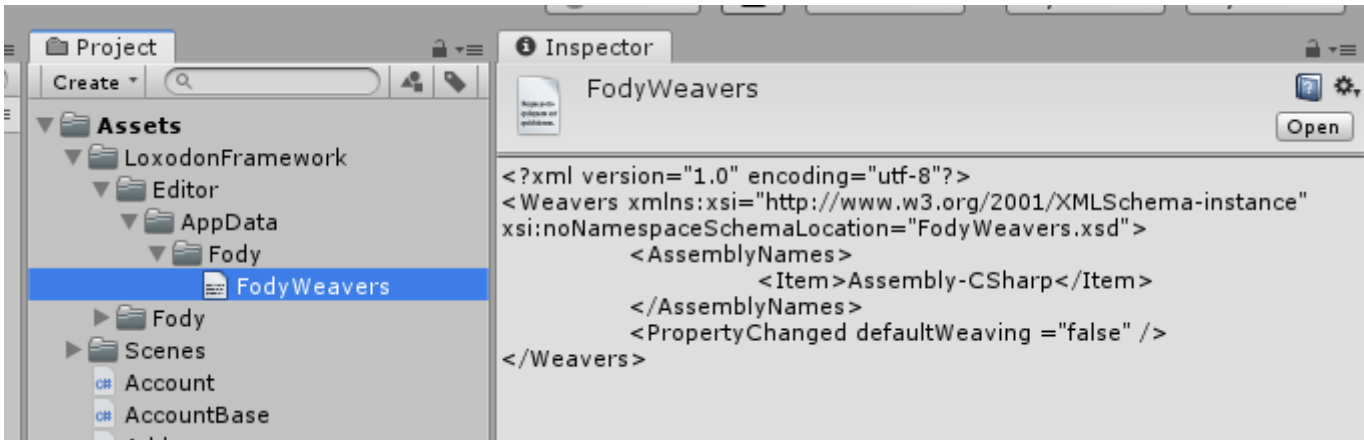
Installing the framework in this way does not require nodejs and openm-cli.

```
{
  "dependencies": {
    ...
    "com.unity.modules.xr": "1.0.0",
    "com.vovgou.loxodon-framework-fody": "2.4.9",
    "com.vovgou.loxodon-framework-fody-propertychanged": "2.4.9",
    "com.vovgou.loxodon-framework-fody-tostring": "2.4.9"
  },
  "scopedRegistries": [
    {
      "name": "package.openupm.com",
      "url": "https://package.openupm.com",
      "scopes": [
        "com.vovgou",
        "com.openupm"
      ]
    }
  ]
}
```

Quick Start

PropertyChanged.Fody

Add the assembly filenames that need to automatically weave "Notifity" code to the configuration file.

FodyWeavers.xml

```xml
<Weavers xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="Fo
        <AssemblyNames>
                <Item>Assembly-CSharp</Item>
        </AssemblyNames>
        <PropertyChanged defaultWeaving ="true" />
</Weavers>
```

Create a User class in the project and add annotation "AddINotifyPropertyChangedInterface" to the class, as follows:

```csharp
[AddINotifyPropertyChangedInterface]
public class User
{
    public string FirstName { get; set; }

    public string LastName { get; set; }

    public string FullName => $"{FirstName} {LastName}";
}
```

After the project is compiled, use the ILSpy tool to view the assembly.the code of the User class is as follows, "PropertyChanged.Fody" weaves the INotifyPropertyChanged interface and related code for User.cs

```csharp
public class User : INotifyPropertyChanged
{
        public string FirstName
        {
                [CompilerGenerated]
                get
                {
                        return FirstName;
                }
                [CompilerGenerated]
                set
                {
                        if (!string.Equals(FirstName, value, StringComparison.Ordinal))
                        {
                                FirstName = value;
                                <>OnPropertyChanged(<>PropertyChangedEventArgs.FullName);
                                <>OnPropertyChanged(<>PropertyChangedEventArgs.FirstName);
                        }
                }
        }

        public string LastName
        {
                [CompilerGenerated]
                get
                {
                        return LastName;
                }
                [CompilerGenerated]
                set
                {
                        if (!string.Equals(LastName, value, StringComparison.Ordinal))
                        {
                                LastName = value;
                                <>OnPropertyChanged(<>PropertyChangedEventArgs.FullName);
                                <>OnPropertyChanged(<>PropertyChangedEventArgs.LastName);
                        }
                }
        }

        public string FullName => FirstName + " " + LastName;

        [field: NonSerialized]
        public event PropertyChangedEventHandler PropertyChanged;

        [GeneratedCode("PropertyChanged.Fody", "3.4.1.0")]
        [DebuggerNonUserCode]
        protected void <>OnPropertyChanged(PropertyChangedEventArgs eventArgs)
        {
                this.PropertyChanged?.Invoke(this, eventArgs);
```

```
        }
    }
```

ToString.Fody

Add `<ToString/>` to FodyWeavers.xml

```xml
<Weavers xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="Fc
    <AssemblyNames>
        <Item>Assembly-CSharp</Item>
    </AssemblyNames>
    <ToString />
</Weavers>
```

Your Code

```csharp
[ToString]
public class User
{
    public string FirstName { get; set; }

    public string LastName { get; set; }

    public string FullName => $"{FirstName} {LastName}";
}
```

What gets compiled

```
public class User
{
    public string FirstName { get; set; }

    public string LastName { get; set; }

    public string FullName => $"{FirstName} {LastName}";

        [GeneratedCode("Fody.ToString", "1.11.1.0")]
        [DebuggerNonUserCode]
        public override string ToString()
        {
                return string.Format(CultureInfo.InvariantCulture, "{T: 'User', FirstName: '{0}'
                {
                        FirstName ?? "null",
                        LastName ?? "null",
                        FullName ?? "null"
                });
        }
}
```

Contact Us

Email: yangpc.china@gmail.com

Website: https://vovgou.github.io/loxodon-framework/

QQ Group: 622321589