背景

网络上打包Dll的教程多如繁花，殊不知Unity定义一个AssemblyDefinition文件，dll工具集就躺在那儿了，如果还不知道这个小清新功能的，可以通读一下！

知识点

**一 . Assembly Definition**  
Unity 3D 2017.3 引入了一个新的概念，[Assembly Definition](https://links.jianshu.com/go?to=https%3A%2F%2Fdocs.unity3d.com%2FManual%2FScriptCompilationAssemblyDefinitionFiles.html" \t "_blank)。  
这个文件定义了它所在的文件夹及其子文件夹的所有脚本应该编译在哪一个程序集中。利用它就能将咱们工程的脚本划分成多个程序集，也就是编译成多个Dll。

**二 . 循环引用**  
如下图，Assembly Definition 与 预设程序集（Predefined Assemblies ，即Assembly-CSharp.dll等等）的关系是，这些预设程序集自动引用 用户使用Assembly Definition自定义程序集。

Unity程序集之间的引用关系

**Tips**

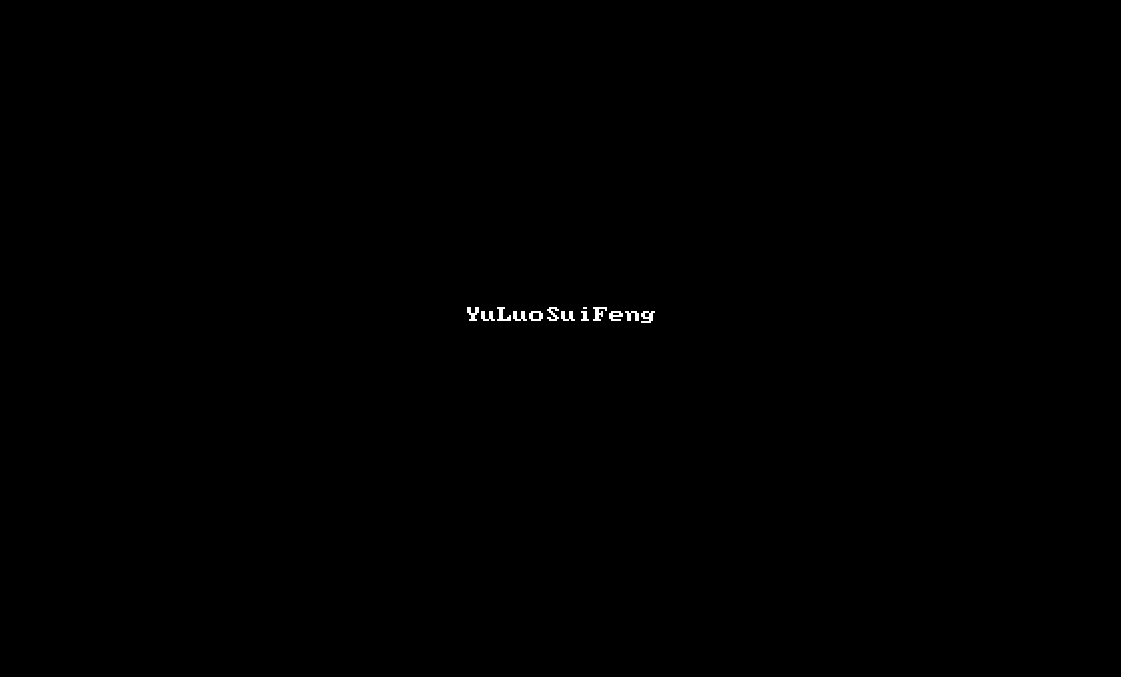
* Predefined Assembly→可以理解成用户脚本
* Assembly Definition定义的程序集→可以理解成你这个工具包
* 可以很清楚的看到，这种引用关系，很适合做自己的工具包：用户脚本依赖工具，工具不依赖用户脚本但依赖其他DLL插件。
* 当然，如果你想把用户脚本也指定一个Assembly Definition，额~ 必定产生很绕的引用关系，这种关系叫循环引用，在程序里面是不被允许的，除非你大刀阔斧的[规划一下你的文件夹结构](https://links.jianshu.com/go?to=http%3A%2F%2Fzh.esotericsoftware.com%2Fforum%2FUnity-2017-3-Assembly-Definition-Files-9270)（参考他editor+runtime理念）。
* **如果你的工具是引用了UnityEditor**，平台务必只能勾选Editor，或者引入宏指令达到editor和runtime通用(务必注意打包后使用宏指令包裹的Editor逻辑在生成的 dll 中将被剔除，编辑器下则不会有这个问题)。

实现

在本文，笔者将使用[这篇文章](https://www.jianshu.com/p/bf82beb41f7f" \t "_blank)里面的**事件系统**来做个简单的讲解：  
**首先**，找到你需要单独打包Dll的文件夹，创建Assembly Definition，写一个高Bigger的名称。

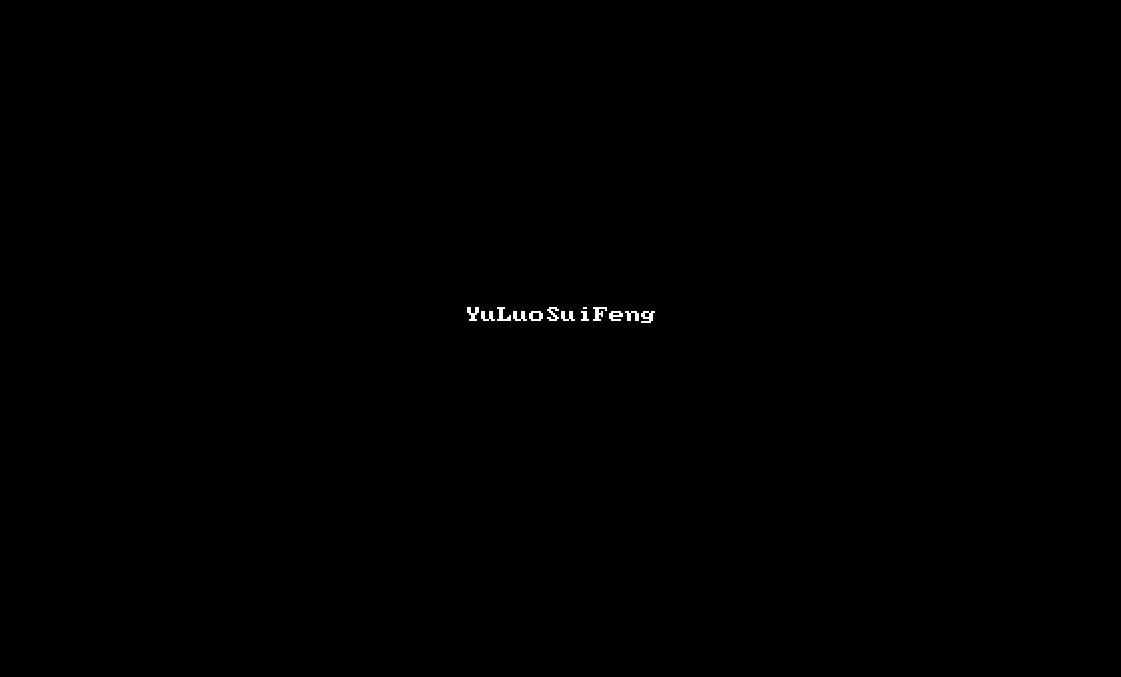
Ardez.event.utilityBigger该够了

**然后**，你的Unity右下角菊花开始转动，嗯，在这一步Unity就已经帮你把这个工具包（dll）打包好了，现在我们一起去看望看望她。



路径：工程根目录-Library-ScriptAssemblies-你定义的名字.dll

**最后**，我们测试一下这个Dll好不好用，首先我运行一遍源码的事件系统，证明能正常工作，然后会删掉这个源码并使用dll替换。



cool! 运行正常，工具封装成功咯

**结论**，完美运行，DLL工具封装成功，可以拷贝到其他工程里面用啦～

刚好讲到了DLL，笔者就顺势推荐一个反编译软件吧，如果还没用上的小伙伴就有福啦：

这主题百搭vs和unity哇

如果存在xml描述文件，熟悉的代码注释就回来了

扩展阅读

* [可能你需要升级Visual Studio Tools for Unity](https://links.jianshu.com/go?to=https%3A%2F%2Fforum.unity.com%2Fthreads%2Fsolved-assembly-definition-files-broken-on-new-machine.509798%2F)
* [为什么你定义的程序集不能引用用户脚本（Assembly-CSharp.dll）答案在第11楼](https://links.jianshu.com/go?to=https%3A%2F%2Fforum.unity.com%2Fthreads%2Fexample-project-assembly-definition-files.482313%2F)
* [Unity 使用VisualStuido将C#脚本封装打包DLL使用 - 简书](https://www.jianshu.com/p/7934f1c6374b)

写到最后：

这个Assembly Definition 的存在，使得每个功能块依存关系更加明确，功能块互相独立，如果我们把特殊的算法或者一些模块做成独立的dll拿来热更是不是很酷呢，这种由assembly definition 带来的热更便利，有兴趣的小伙伴可以研究下呢~

作者：雨落随风  
链接：<https://www.jianshu.com/p/6a1202f74222>  
来源：简书  
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。