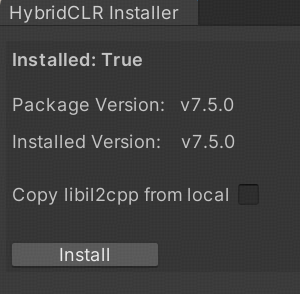
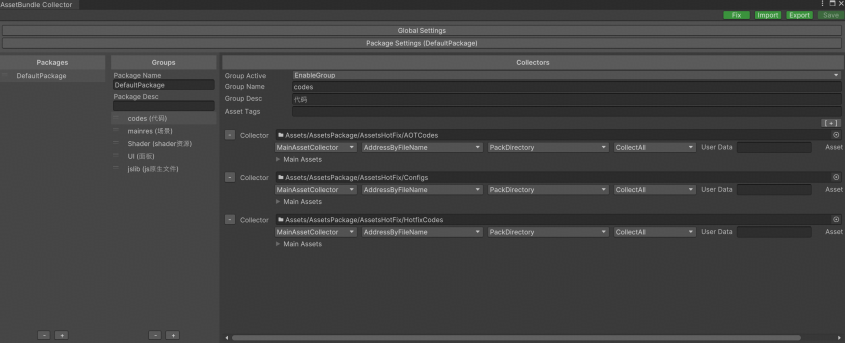
快速开始

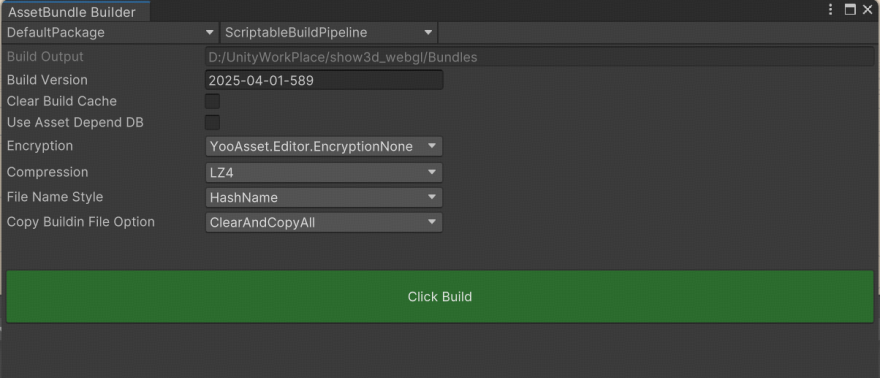
1. 发布热更新框架包
2. 找到工具栏-->HybridCLR-->Installer-->Install，安装HybridCLR，确认已安装；



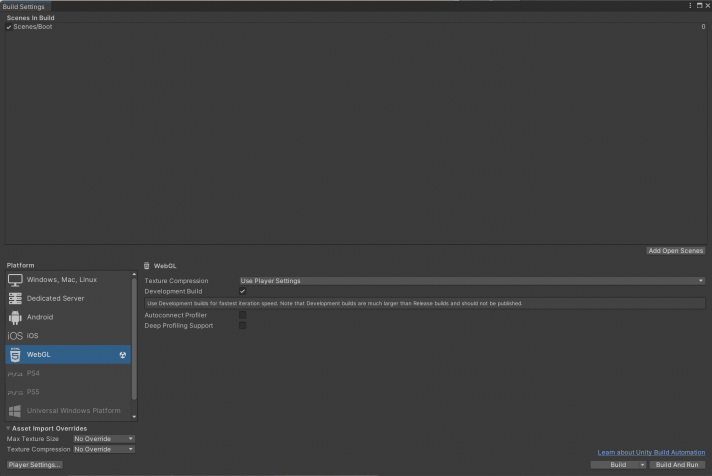
1. 初始化生成框架数据 HybridCLR -->Generate-->All;
2. 生成link.xml链接文件，避免代码过度裁剪，HybridCLR-->Generate-->LinkXml;
3. 配置初始热更资源包（必须），可根据需要设置空包，初始包等；工具栏YooAsset-->AssetBundle Collector，根据需要配置初始资源包；



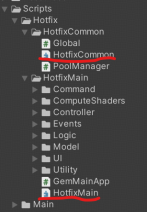
1. 资源打包，工具栏YooAsset-->AssetBundle Builder，打包管线推荐使用ScriptableBuildPipeline；初始需要拷贝资源到StreamingAssets，CopyBuildinFileOption设置为ClearAndCopyAll，然后打包；



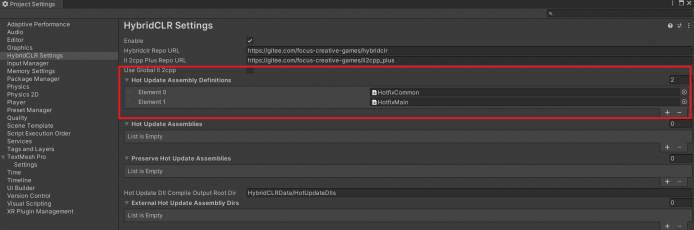
1. 发布框架包，选择发布平台（和打包资源匹配），Build；



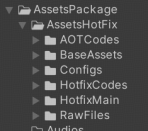
1. 发布热更资源包
2. 需要热更的代码生成Assembly Definition，在代码文件夹右键Create-->Assembly Definition;



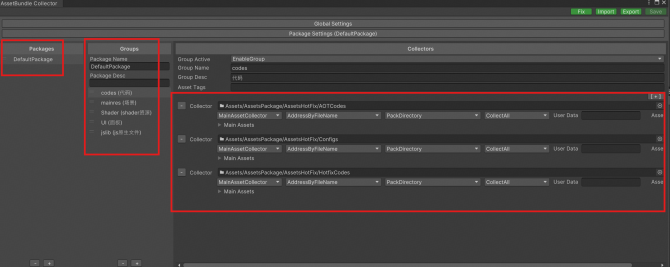
1. 配置需要热更的代码，HybridCLR-->Setting-->Hot Update Assembly Definitions;



1. 配置热更新资源（代码和需要热更的prefab/UI/Shder/Sound等），YooAsset-->AssetBundle Collector；



上述为资源文件夹分类，分别为AOT元数据，通用素材文件，代码数据记录配置文件，热更代码文件，HotfixMain为更新的主资源包括场景/prefab/UI/shader和shader变体等；

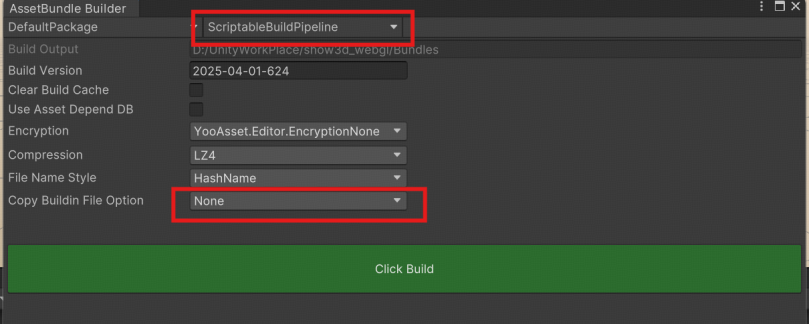


根据需要可自定义文件夹及资源设置；

PS：Shader变体收集，可使用YooAsset官方提供的工具，在Package Manager中找到YooAsset包导入Sample；

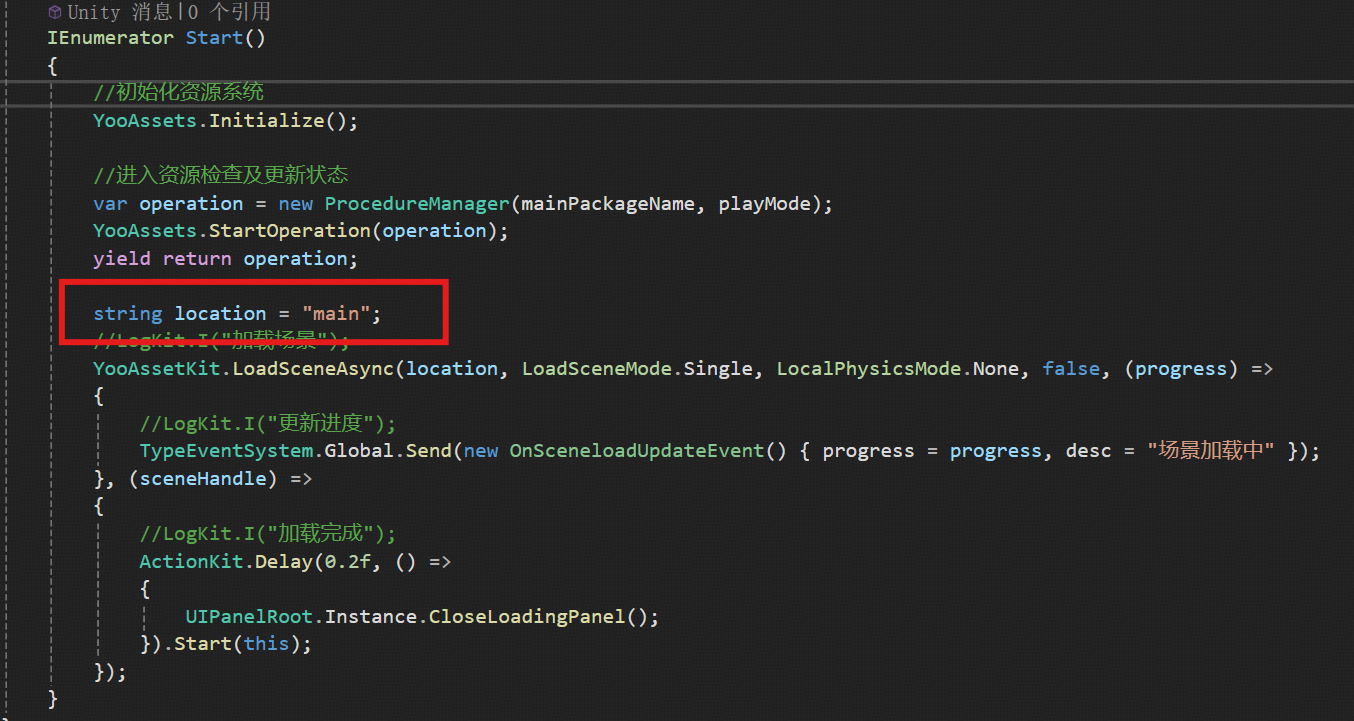
友情提示:资源包和执行程序包可以分为两个工程，避免混乱；

1. 着色器变体收集，工具栏Tools-->着色器变种收集器，将收集好的shader变体配置在上述资源包；
2. 热更新代码配置，HybridCLR-->CompileDll-->发布的目标平台，执行代码生成；
3. 热更代码收集，工具栏Build-->BuildAssetsAndCopyToAssetsPackage;代码收集完成，确认步骤3中有收集代码文件夹；
4. 打资源包，打包完成后将资源包放在cdn服务器，完成！



场景执行

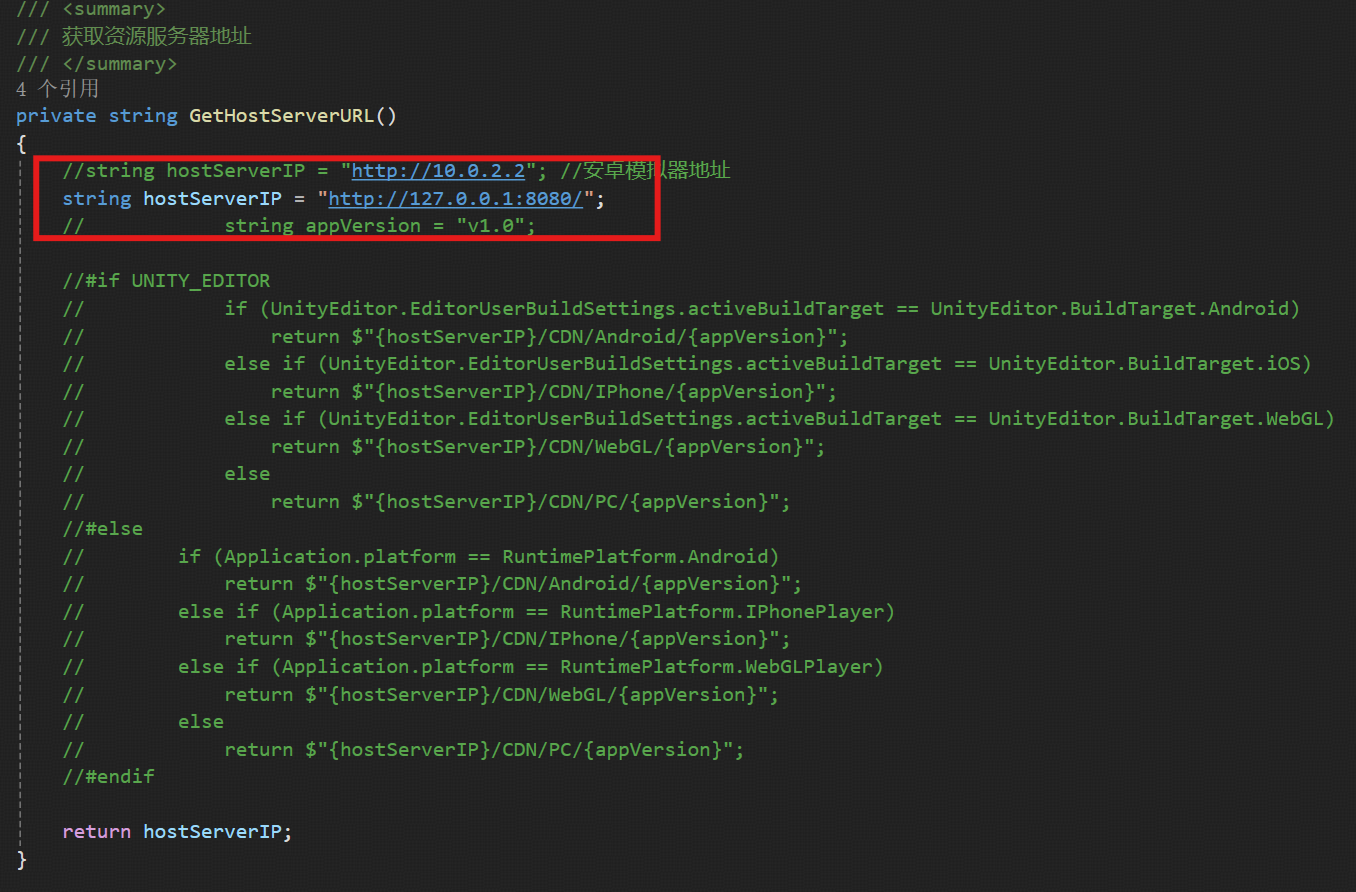
1. 启动场景Boot，在Boot脚本中加载你的热更场景，场景名为YooAsset模糊加载；



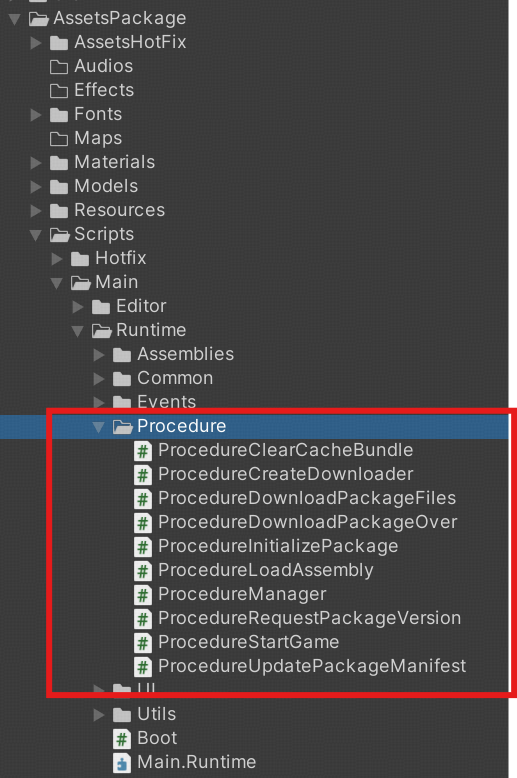
1. 初始化执行流程，查看ProcedureManager，具体执行流程为：



1. 修改服务器地址，在ProcedureInitializePackage流程，也可自定义获取方法实现。

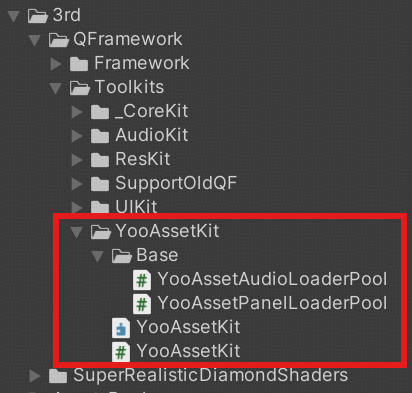


1. 流程文件位置如何，可根据需要修改或添加配置，此内容为框架公共代码，随执行程序打包发布；



1. YooAsset加载资源资源API，已集成在QFramework中，调用请使用YooAssetKit，根据需要调用对应API；

YooAsset加载UI和音频已重写，可以直接使用QF中的UIKit.OpenPanel<T>()打开和加载对应UI界面，可以使用AudioKit加载对应音频（实际是通过YooAsset加载资源）；



1. 请注意资源的加载和及时卸载，合理管理内存。