

# 本地资源检测组件接入文档

本地资源检测服务用于检测项目内部Editor中各种资源、场景和代码相关的使用情况，旨在快速发现和解决开发过程中存在的资源问题。研发团队可通过简单的部署，即可将人力从繁冗复杂的操作中解放出来。该功能支持自动检测项目的资源、代码和一些设置问题，达到提高研发效率的目的。

同时建议研发团队新建一条流水线来使用此组件，设置定时任务，以达到定期体检的效果。

## 1. 介绍

资源组件下载解压后，有2个文件，如下图所示：



UwaProjScan.dll 中集成了所有的规则。

uwascan\_reluconfig.json文件用于所需要的规则，例如：

```
{
  "UwaProjScan.ScanRule.SceneCheck.Editor_MobileFog": "true",
  "UwaProjScan.ScanRule.SceneCheck.OcclusionCulling": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_UndefinedTag": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_MultipleAudioListeners": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_ShadowResolution": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_MeshCollider": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_StaticRigidBody": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_RigidBody": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_CanvasChildren": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_UIOutside": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_OcclusionCulling": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_PrefabType": "true",
  "UwaProjScan.ScanRule.ProjectAssets.VideoClip.Video_SizeLimit": "true",
  "UwaProjScan.ScanRule.ProjectAssets.Texture2D.Texture_AlphaAllOne": "false",
  "UwaProjScan.ScanRule.ProjectAssets.Texture2D.Texture_CompressionFormat": "false",
  "UwaProjScan.ScanRule.ProjectAssets.Texture2D.Texture_EdgeTransparent": "false",
  "UwaProjScan.ScanRule.ProjectAssets.Texture2D.Texture_FilterMode": "false",
  "UwaProjScan.ScanRule.ProjectAssets.Texture2D.Texture_PureColor": "false",
  "UwaProjScan.ScanRule.ProjectAssets.Texture2D.Texture_Resolution": "false",
  "UwaProjScan.ScanRule.ProjectAssets.Texture2D.Texture_RW": "false",
  "UwaProjScan.ScanRule.ProjectAssets.Texture2D.Texture_SpriteMipmap": "false",
  "UwaProjScan.ScanRule.ProjectAssets.Texture2D.Texture_WrapMode": "false",
}
```

上图中，true为使用该规则，false为不使用该规则。用户可根据自身项目需求自定义配置。该服务同时支持只针对打包场景进行扫描和自定义目录的扫描，如图：

```
{
  "scan_scene_mode": "buildonly",
  "scan_asset_mode": "targetonlywithdepends",
  "scan_asset_dirs": [
    "Assets"
  ],
  "scan_asset_filter_dirs": [
  ],
  "UwaProjScan.ScanRule.SceneCheck.Editor_MobileFog": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_UndefinedTag": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_MultipleAudioListeners": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_ShadowResolution": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_MeshCollider": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_StaticRigidBody": "true",
  "UwaProjScan.ScanRule.SceneCheck.Scene_RigidBody": "true",
}
```

默认情况下，插件会扫描项目中所有的场景。而在配置文件中添加 scan\_scene\_mode 字段，且把值设为“buildonly”时，插件只扫描参与打包的场景。

默认情况下，插件会扫描项目中所有的目录中的资源。可以在配置文件中添加

scan\_asset\_mode 、 scan\_asset\_dirs 和

scan\_asset\_filter\_dirs 字段，来自定义目录规则：

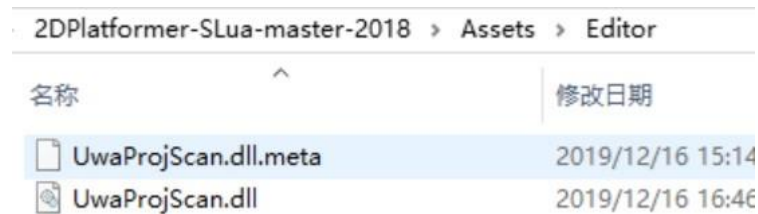
scan\_asset\_dirs：被扫描的路径（支持多个目录）；

scan\_asset\_filter\_dirs：不被扫描的路径（支持多个目录），同时位于 scan\_asset\_dirs 和 scan\_asset\_filter\_dirs 中路径下的资源，依然不被扫描；

scan\_asset\_mode：值为“targetonly”时，才会使用 scan\_asset\_dirs 中的路径；值为“targetonlywithdepends”时，除了使用scan\_asset\_dirs 中的路径，在收集资源时还会收集其依赖项（即使依赖项不在指定的路径下）。

## 2. 准备工作

- 2.1. 将 UwaProjScan.dll 文件放入工程目录中的某个 Editor 目录下，如下图所示：



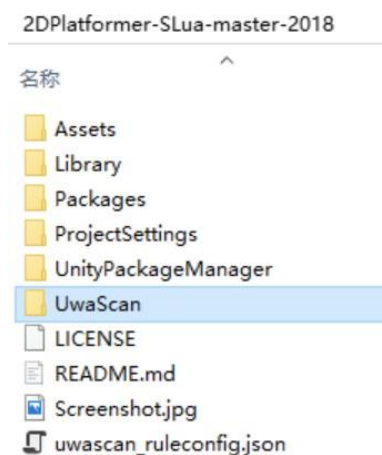
- 2.2. 编辑 uwascan\_reluconfig.json 文件，与 UwaProjScan.dll 文件放入同一个目录下，以确保使用选定的规则进行扫描。

- 2.3. 与 Jankinsfile 进行集成，在 Jenkins pipeline 的编排中加入一下命令，可以运行本地资源检测组件。

```
${UnityPath} -batchmode -projectPath ${ProjectPath} -executeMethod UwaProjScan.MainScan.DoTest  
-logFile  
${LogFile} -quit
```

- 2.4. 生成数据

执行完毕后，会在Assets的同级目录下生成UwaScan目录，如下图所示：



## 2.5. 数据同步

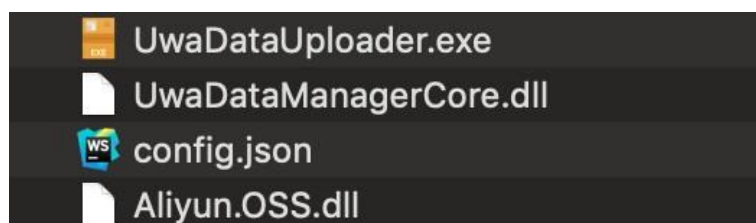
可以使用Jenkins的归档功能把数据同步到主节点（也可以使用scp等其它工具进行数据同步），示例如下：

```
1 pipeline{
2     AGENT xxx
3     STAGES {
4         STAGE ('...') {
5             steps{
6                 ARCHIVEARTIFACTS: '.../.../ASSETS/UWASCAN/' //数据生成的目录
7             }
8         }
9     }
10 }
```

检测完成后，如果有打包流程，打完的游戏包体也可以通过这个方式同步到Jenkins master节点，供主节点的组件进行上传。

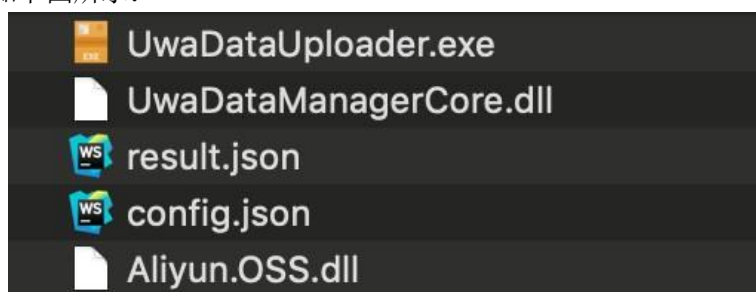
## 3. 提交上传组件

提交上传组件的具体文件如下图所示：



上传组件下载解压缩后，有4个文件，其中：

- 1) config.json是需要配置上传测试数据的UWA账号密码；
- 2) UwaDataUploader.exe运行完毕后会生成1个result.json的文件，包含了上传的结果以及Projectid，如下图所示：



该目录可放在Jenkins master节点的任意位置，用户可以通过UwaDataUploader.exe文件即可完成文件的提交和上传操作。

### 3.1. 使用方法

UwaDataUploader.exe 资源检测结果目录

在检测结束以后，检测结果会上传至UWA网站进行分析与展示，可以登录查看报告。

#### Windows

```
1 UwaDataUploader.exe "D:/.../UwaScan"
```

#### 类Unix

```
1 mono .../... /UwaDataUploader.exe /var UwaScan
```