# VitoSDK使用说明

### v1.3.0张林博

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本更新记录 | 版本修改人 | 文档修改人 | 修改内容 | 上一版本号 |  |
| v1.3.0 | 张林博 | 张林博 | 1. 增加自定义统计功能案例 2. 增加移动端电量统计   3.修改兼容版中个人信息栏显示两个姓名为显示姓名和编号 | V1.2.5 | 新功能说明在文档最后功能点4之后 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

### 概述

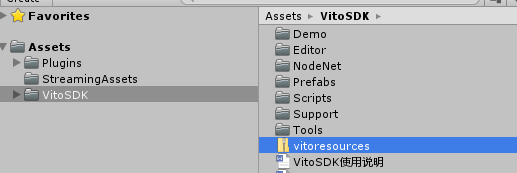
* 1. 通过VitoSDK ，第三方应用商可以将自己的VR应用生成为一个教师端(PC或平板,非VR)和一个学生端(VR模式，可以为HTC，Gear,Iealens等)，通过教师端控制台可以控制学生端的操作，观察学生当前视角，场景切换，问题提问，问题统计等功能。
  2. 教师端和学生端为同一个项目文件，但因为发布平台可能不同，建议做两份项目文件(一份为PC平台(教师端，HTC)，一份为Android平台（Gear,Idealens）)，通过项目管理工具（SVN，git等）保证两份代码的一致性。
  3. VitoSDK 内置了Gear，HTC，Idealens的一些基本操作的接口，但并没有包含HTC、Oculus、Idealens等的SDK，但都通过宏进行控制，此处把用到的宏定义和使用列举如下

|  |  |
| --- | --- |
| 宏定义 | 说明 |
| HTC VIVE | HTCVIVE的宏定义 |
| GEARVE | GEARVR的宏定义，非VR模式，移动端VR均需要开启 |
| IVR | Idealens的宏定义，Idealens设备需要开启”GEARVR;IVR”两个宏 |
| DPN | 大朋VR 的宏定义，大朋设备需要”GEARVR;DPN”两个宏 |

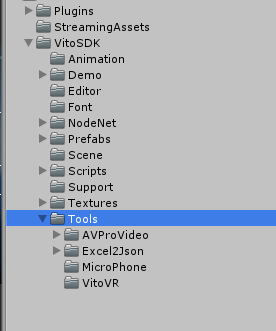
注：PC端教师控制台需要开启GEARVR的宏，其他宏关闭(历史原因，之后会修改掉)

注2：Idealens端需要开启”GEARVR;IVR”两个宏，其他宏关闭

## 文件结构说明



vitoresources下为之后需要用到的配置文件，可以从项目下拿出来，之后放到指定的路径下



Plugins下为SDK内使用到的一个视频插件依赖到的底层库，可以不用关心

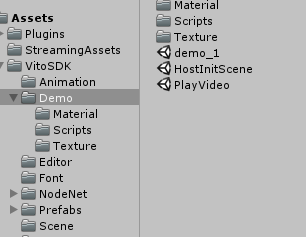
StreamingAssets下为本地保存的数据，提问的问题，配置的场景信息，历史学生信息等

Vito SDK/Tools/AVProVideo 为视频插件

Vito SDK/Tools/Excel2Json/为导表工具,不需要

Vito SDK/Tools/MicroPhone/为语音工具

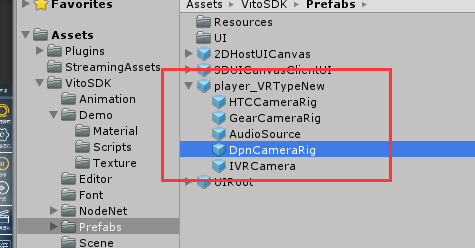
VitoSDK/Tools/VitoVR/下为VR模式的一些常用接口，但是和网络功能无关，用不到可以不用关心



VitoSDK/Demo/HostInitScene，为初始化场景，作为启动时的第一个场景

VitoSDK/Demo/PlayVideo为360视频播放场景，场景名不可修改

VitoSDK/Demo/demo\_1为网络功能的demo，简单展示了物体移动缩放旋转和摄像机位置角度的同步



VitoSDK/Prefabs/player\_VRTypeNew为集成了HTC、Gear、大朋、Idealens的一个预制体，之后介绍

其他目录暂无

### demo场景打包idealens/大朋一体机/GearVR等平台说明(重要)

V1.2.5修改

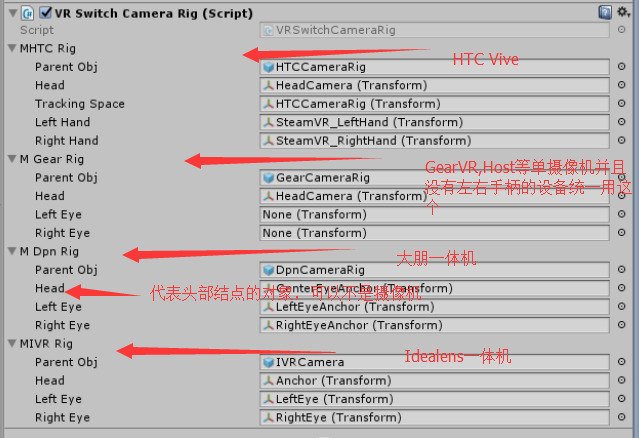
**注1：所有场景的代表玩家的摄像机都要替换为Player\_VRTypeNew 。替换方式为：**

**例如原本场景是大朋VR的DpnCameraRig，就添加player\_VRTypeNew到场景内，并删除player\_VRTypeNew下的DpnCameraRig对象，然后，把场景内的DpnCameraRig放到player\_VRTypeNew下，保证DpnCameraRig的local坐标为P(0,0,0),R(0,0,0),S(0,0,0),移动player\_VRTypeNew的位置来调整玩家的初始位置。同时注意场景内只需要一个audiolistener.**

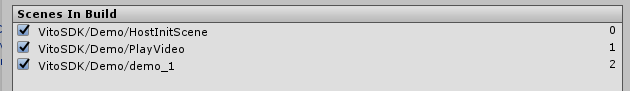
如图



**注2：VitoSDK已经做了多平台集成的功能，开发者只需要导入对应平台的SDK，然后替换或者下图中对应的部分就可以，如果有其他平台，也可以修改VRSwitchCameraRig.cs代码，添加新的支持平台。如果不需要同时支持多平台，也可以删除对应平台的内容，或者直接留空也可以。**

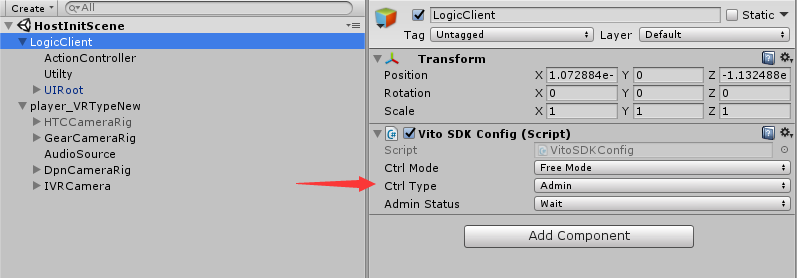


1.新建一个空项目，导入VitoSDK.unitypackage 导入IVRSDK.unitypackage（idealens SDK）

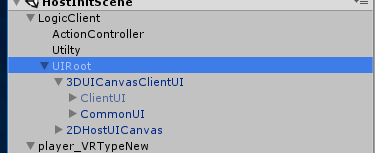
2.将如图场景加入打包列表

3.配置控制台：

在HostInitScene场景中，确保以下设置



CtrlType为Admin代表为老师端

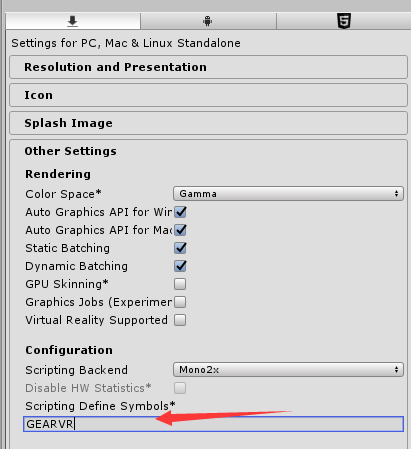


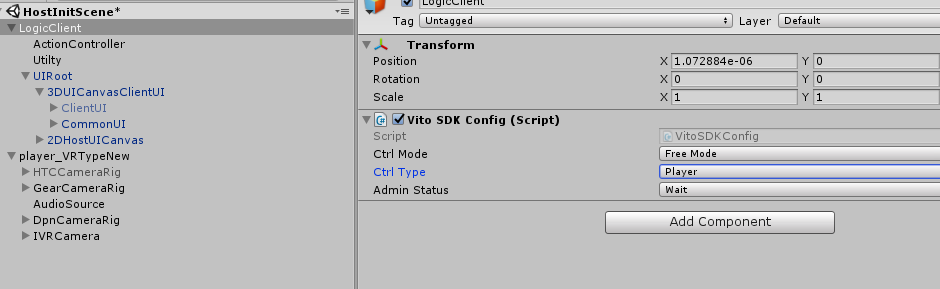
UIRoot下ClientUI关掉，这是在只学生段展示的UI，2DHostUICanvas开启，这是老师端展示的UI

V1.2.5修改

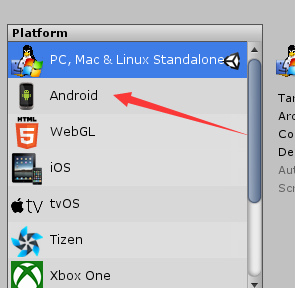
**3DUICanvasClientUI 以及它的子节点需要设置层级为3DMobileUI ,这里是场景加载进度条UI和问题提问UI，否则会显示不正确,开发者需要检查，确保层级正确**

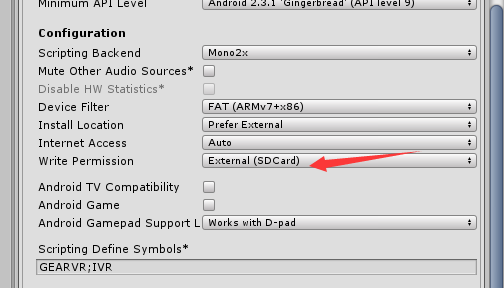
**开发者也可以修改UILoadingBar.cs代码中的3DMobileUI为其他内容，对应的这里的层级也可以为其他内容。**

1. 在PC平台下确保GEARVR宏开启，然后打包，生成教师控制台版本
2. 将刚刚新建的项目复制一份，作为移动端的项目，通过svn或者git保持两个项目内容一致，然后同样打开HostInitScene,修改CtrlType为Player，代表学生端



关闭2DHostUICanvas，并打开Client UI,可以看到如下界面说明设置正确

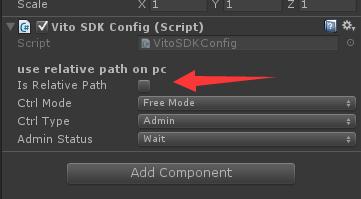
1. 修改目标平台为Android,并配置GEARVR;IVR的宏回车
2. 开启sd卡读写权限，否则不能读配置文件以及视频文件



1. 打包apk。

9.配置配置文件GlobalSetting.xml的位置

**V1.2.5版本新增，控制台视频和场景配置文件的位置可以选择绝对路径或者相对路径。如果使用绝对路径就是GlobalSetting.xml中配置的盘符加上路径名称。如果使用相对路径则是放在exe文件的同级目录加上路径名称。开发者也可以修改VitoSDKConfig.cs代码来自定义路径配置文件路径。**

**可以通过如图设置使用相对路径或者绝对路径，默认情况下为绝对路径。**

注：在editor模式下，GlobalSetting.xml的路径是Assets/目录的上一级目录，即项目目录。生成exe执行文件后GlobalSetting.xml的位置为exe文件的同级目录

Android设备下GlobalSetting.xml的路径是sdcard/vitoresources/目录下

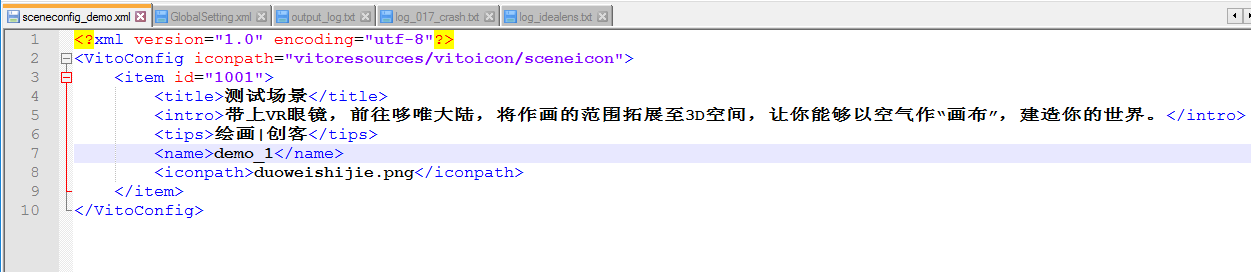
注：1.1版本sdk 配置文件改为xml文件， GlobalSetting.xml ，场景和视频描述的配置不再放到服务器，而是在本地，方便B端用户自己添加视频内容。

GlobalSetting.xml ，各节点意义可参看注释

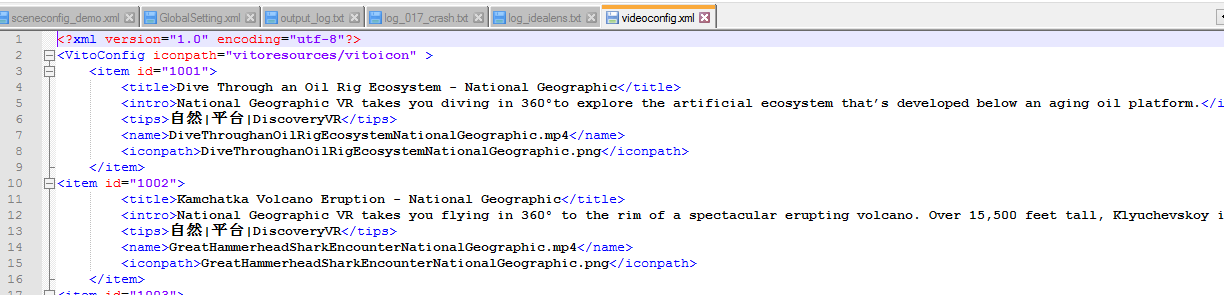


然后视频和场景配置文件路径在GlobalSetting.xml中有配置

场景文件sceneconfig\_demo.xml如下

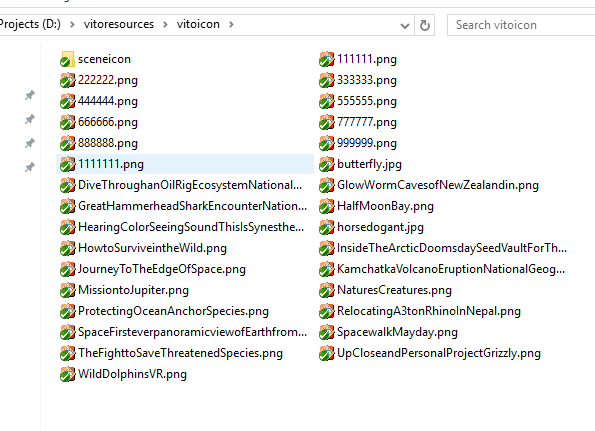


视频文件videoconfig.xml与场景文件的配置基本一致示例如下



同时这两个文件里都有配置对应的icon的路径iconpath属性(相对于globalsetting.xml中配置的根目录)

示例如下



注：”sdcard/”代表安卓设备的sd卡根目录

Pc版GlobalSetting.xml放在exe文件的同级目录下，android放在sd卡下的vitoresources目录下

注：pc和安卓设备需要预先把视频放在GlobalSetting.xml指定的目录下

8，通过控制台测试配置文件是否正确，如果能够切换场景或视频说明配置正确。

9，安装apk到设备可以开始测试了

### VitoSDK 使用说明

1. 场景切换功能

控制台显示的自定义资源列表(场景)是通过GlobalSetting.xml 中sceneconfigfile字段指定的配置文件来显示的，该文件内填的场景名称scenename要和unity中的场景文件的名字一致。

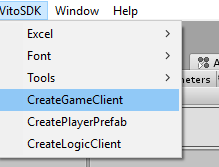
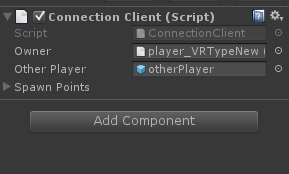
同样的设置方式也适用于视频列表，不同的是视频配置文件中videoname要是视频文件的名字

如果想要自己定义UI界面，可以通过如下接口

VitoPluginLoadScene.instance.OnRequestChangeScene(“场景名称”).调用场景切换

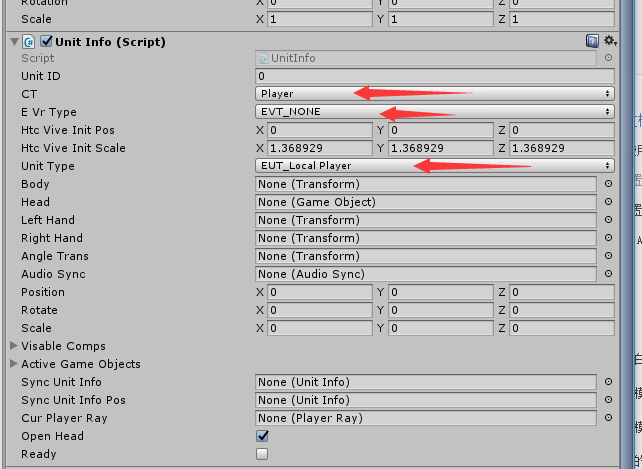
注册Action<bool> VitoPluginLoadScene.showLoadingBarAction(bool show),和VitoPluginLoadScene.updateLoadingBarAction(float value) ，来获取显示或隐藏自定义进度条以及更新进度条进度

可以参考预置的UILoadingBar.cs实现进度条功能。

1. 为自己的场景添加同步功能，每一个独立的场景都需要重复下述步骤
   * 1. 点击CreateGameClient，创建联网组件
     2. 设置联网组件如图，联网组件里Owner代表玩家对象，就是下一步所说的player\_VRTypeNew对象，OtherPlayer代表其他玩家，是一个名称为otherPlayer的预制体
     3. 点击CreatePlayerPrefab,或者直接拖拽player\_VRTypeNew预制体到场景中， 创建人物预制，调整Player到需要的位置， 删除场景原来的摄像机。

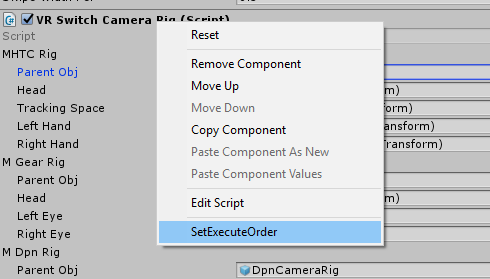
，如图，人物预制结构如下，IVRCamera为idealens的摄像机GearCameraRig为教师端或者Gearvr下的相机对象。

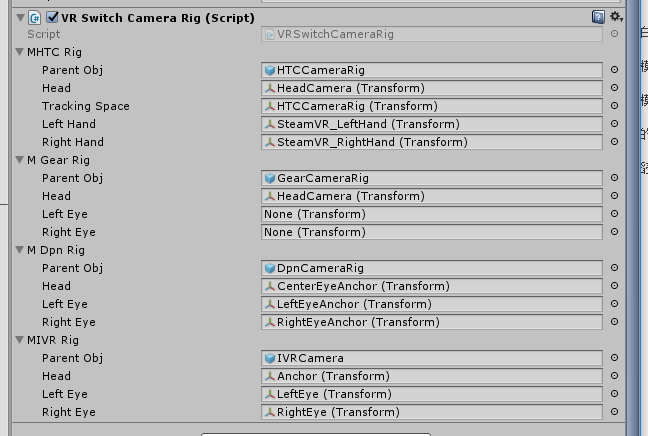
人物预制体上以下设置要与下图一致

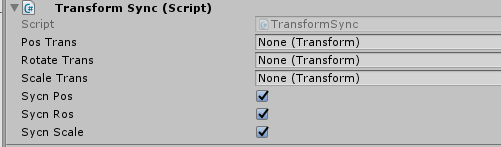


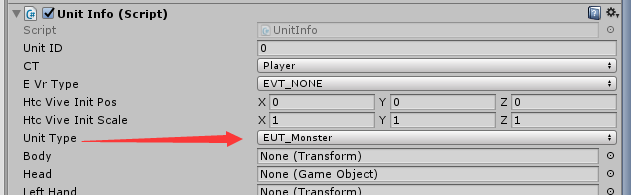
VRSwitchCameraRig.cs用来做多平台适配，需要把对应的物体在Inspector面板对应赋值，在运行时会删掉非当前平台的对象。

建议在VRSwitchCameraRig上右键 执行一遍 SetExecuteOrder，设置该脚本最早执行





1. 同步非玩家对象，场景内物体如果需要同步，要挂在TransformSync.cs脚本，，需要同步的数据postion,rotation,scale，对应打勾，不打勾不同步。默认要同步的transfrom,是脚本所在对象，三个transfrom不需要赋值。添加该组件之后，会自动添加UnitInfo.cs脚本，需要把UnitType对象设置如下，代表非玩家对象。



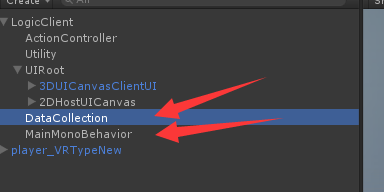
注意：需要客户端往服务器同步的非玩家对象，需要根据当前的状态判断是否执行位移旋转缩放等逻辑。例如：如果教师端逻辑在每帧设置物体A位置，同步逻辑也在每帧同步教师当前观察的学生的物体A的位置，这时两个逻辑就会冲突

所以需要判断当前需要同步的物体有没有执行权限：

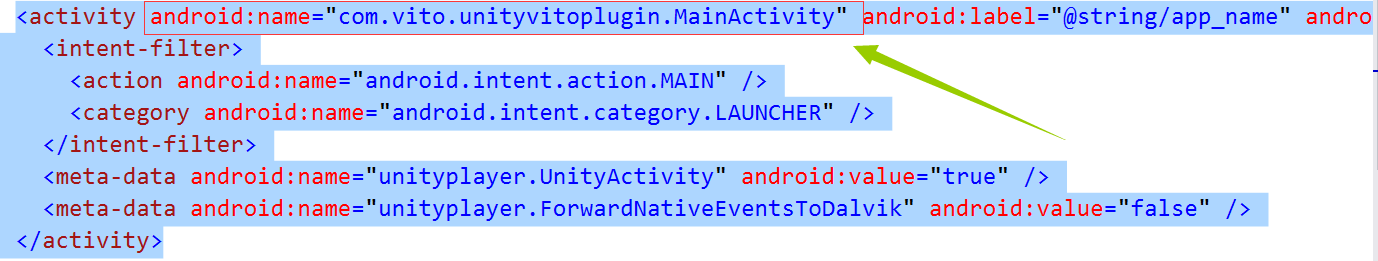
VitoPlugin.ClientHasPermission =true代表客户端有执行权限(单机模式，或联网状态下 自由模式以及被观察时)

4.控制台的UI逻辑和其他一些逻辑都是开源的，开发者可以自行修改、添加和删除功能。

1. V1.3.0功能说明：
   1. 
   2. 如上图，点击用户信息头像可以显示详细信息页面，详细信息页面有收集的与该用户有关的信息。Demo中该信息为随机生成的。实际使用用开发者可参考demo的结构，自定义信息内容和显示方式。
      * + 1. 开发者看demo代码是需要注意一下内容：

DataCollection上挂载的为数据收集管理类DataCollectionManager.cs 开发者需要自己修改里边的demo代码和数据结构以满足自己应用需求。

MainMonoBehaviour上挂载为调用android java代码用于获取用户电量的VitoAndroidSDK.cs，开发者需要把android启动项修改为com.vito.unityvitoplugin.MainActivity 并且添加libvitosdk.jar包到Plugins/Android/目录下



否则不能获取电量，可参考demo中Plugins/ Android/AndroidManifest.xml

1. 开发者自定义统计的数据中最终返回给控制台的参数中deviceid为用户设备号，该设备号和用户信息一对一对应，可以参考或修改UserInfoManger.cs中的方法来更新用户信息和UI
2. VitoPlugin.cs包含为注册和发送自定义消息的静态函数，开发者可以参考平台内现有的注册和发送自定义消息的方式，新增自定义消息
3. 关于全局配置文件GlobalSetting.xml ,该配置文件的内容和格式以及路径是可以开发者自定义修改的，可以参考VitoSDKConfig.cs内方法ReadGlobalConfigFile(),开发者可以在这里修改这些配置文件的存放路径。