# 全球首发—鸿蒙开源平台OpenGL

## 前言

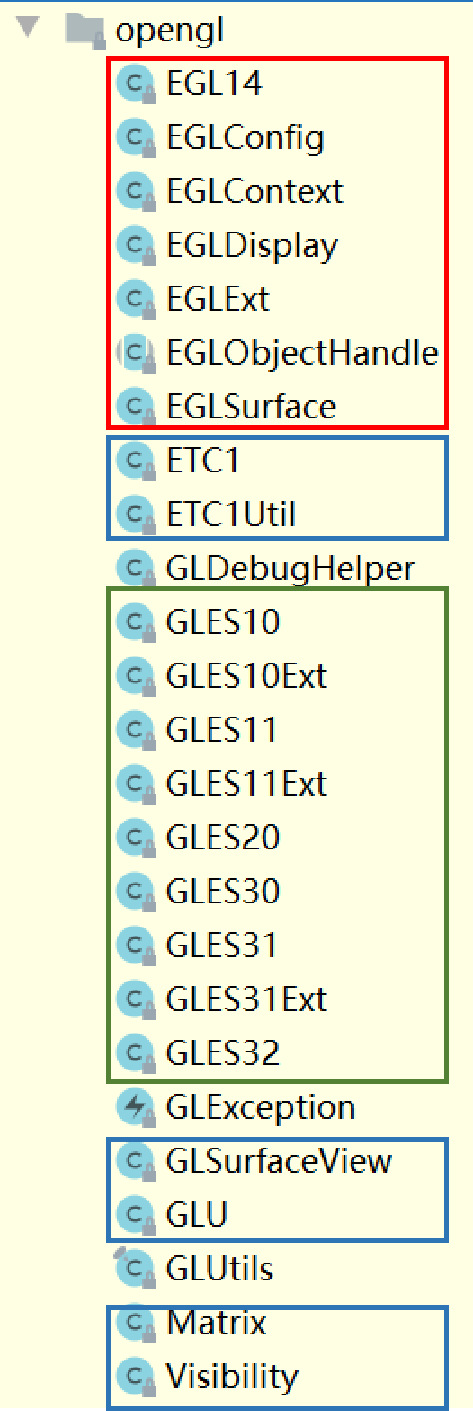
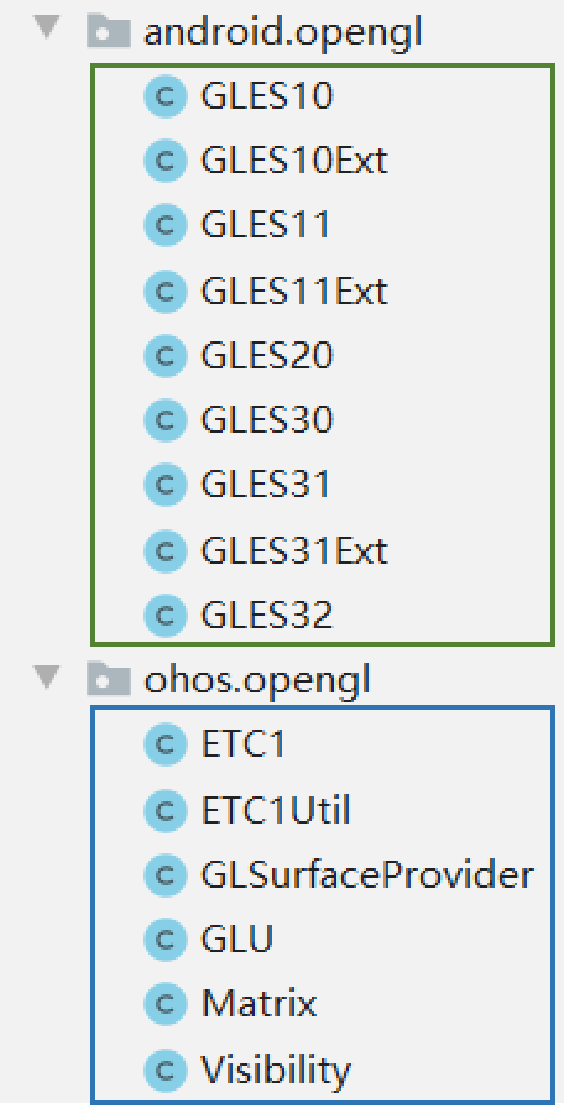
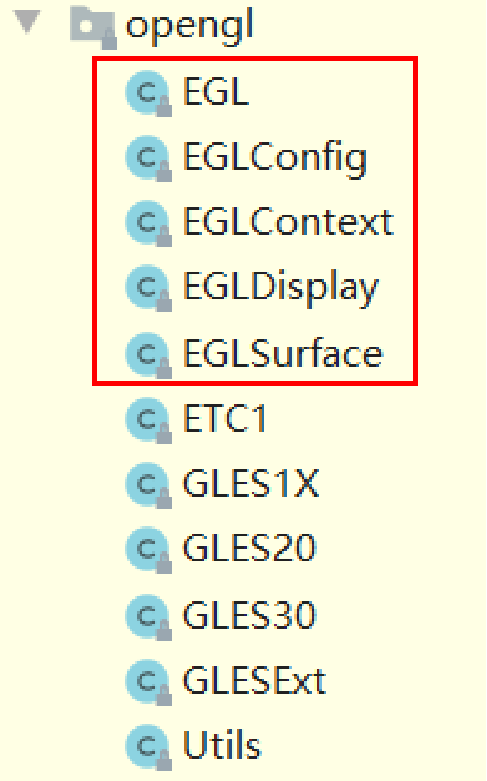
基于安卓平台的OpenGL(http://androidxref.com/9.0.0\_r3/)，实现功能的鸿蒙化迁移和重构，全球首发鸿蒙平台的OpenGL-ISRC，代码已经开源到（https://gitee.com/iscas-ohos/OpenGL\_ISRC/tree/main），欢迎各位下载使用并提出宝贵意见！

## 背景

OpenGL（Open Graphics Library）是用于渲染2D、3D矢量图形的跨语言、跨平台的应用程序编程接口（API），可绘制从简单的图形到复杂的三维景象，操作在GPU之上，实现硬件加速渲染 。目前，Microsoft、SGI、IBM、DEC、SUN、HP等大公司都采用了OpenGL做为三维图形标准，著名的动画制作软件Soft Image和3D Studio MAX、仿真软件Open Inventor、VR软件World Tool Kit、CAM软件ProEngineer、GIS软ARC/INFO等等都是以OpenGL为基础制作完成的。OpenGL是个与硬件无关的软件接口，可以在不同的平台之间移植。

OpenGLES是OpenGL的高性能版本，删减了其中的低效能的操作方式，Android就是使用的这种标准，因此鸿蒙平台的OpenGL-ISRC也是使用的这种标准。

## 鸿蒙OpenGL-ISRC的结构

(1) Android的OpenGL库 (2) **Harmony的OpenGL-ISRC库** (3)Harmony的OpenGL库

图1 ：OpenGL库对比

由图1可知，OpenGL-ISRC整体的结构和Android OpenGL类似，不同之处在于OpenGL-ISRC由两个包组成，分别是android.opengl和ohos.opengl。android.opengl里放置的是目前支持版本的OpenGLES标准，ohos.opengl放置的是常用的图像处理类。且类的数量少于Android的OpenGL。下面对这几点给出解释。

1. **为什么其中一个包名是android.opengl？**

采用C++实现接口的方式，通过so库的调用实现接口的调用。对于目前支持版本的OpenGLES标准这部分，Android已经开源了其生成的so库，为了避免功能重复开发，此处直接使用Android的so库，因此包名必须是android.opengl。

1. **OpenGL-ISRC为什么由两个包组成？**

对于图像处理类的实现，若继续采用Android so库调用的方式，需要将图像处理类放置于android.opengl包下，此时运行鸿蒙环境，会出现方法声明重复的错误，见图2，表明图像处理类中的方法与鸿蒙底层存在的某些方法存在冲突，因此继续使用Android so库调用的方式不再可行。

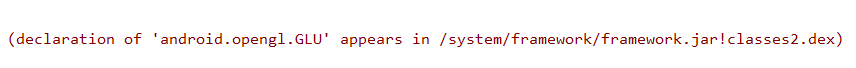


图2：图像处理类放置于android.opengl包下的报错情况

在OpenGL-ISRC中，图像处理类这一部分，依旧采用C++实现接口，通过so库调用实现接口调用的方式。与android.opengl里的类不同的是，我们对图像处理的类的接口进行了鸿蒙化的移植重构，生成了全新的so库进行调用，适配了鸿蒙底层的环境，也避免了此处对安卓的依赖。

1. **OpenGL-ISRC为什么缺少EGL类？**

鸿蒙SDK自带OpenGL库，见图1.(3),内部含有EGL类。OpenGL-ISRC是基于鸿蒙平台，所以直接使用了鸿蒙SDK自带OpenGL库的EGL类，避免了功能的重复开发。

1. **OpenGL-ISRC中的GLSuefaceprovider**

OpenGL-ISRC中的GLSuefaceProvider在功能上和Android的GLSurfaceView相同。因为鸿蒙中的SurfaceView命名为SuefaceProvider，根据命名一致原则，OpenGL-ISRC中的GLSurfaceView命名为GLSuefaceProvider。

## OpenGL-ISRC和鸿蒙SDK OpenGL的区别

OpenGL-ISRC是鸿蒙开源系统的功能相对完整的OpenGL ES库。从使用上来说，OpenGL-ISRC具有较大的独立性，与鸿蒙SDK OpenGL在实现方式、完善程度、功能提供等方面都存在较大不同。

（1） OpenGL-ISRC采用C++实现接口，通过so库调用实现接口调用的方式，而鸿蒙SDK OpenGL 的接口采用java实现的方式，二者的使用较为独立，不存在冲突；

（2）图1.(2)和(3)的对比中可以看出，OpenGL-ISRC的所提供的功能是相对完善的，鸿蒙SDK OpenGL目前还缺少很多标准类，已存在的标准类内部功能也有不完整的现象。

（3）OpenGL-ISRC封装使用了鸿蒙SDK OpenGL的EGL类，避免功能重复开发，因此二者是互相完善的关系；

（4）由于鸿蒙SDK OpenGL的不完整性，OpenGL-ISRC支持的OpenGLES标准类的实现使用了安卓so库的调用，没有使用鸿蒙SDK OpenGL的OpenGLES标准。

## OpenGL-ISRC的使用

1. **GlSurfaceProvider的使用示例**

此处创建了GlSurfaceProvider控件，add到layout中进行显示，如图3所示。为了增强GlSurfaceProvider的可视化效果，在GlRenderer里面设置了一个红色的三角图形（三角具体属性见开源代码：https://gitee.com/iscas-ohos/OpenGL\_ISRC/tree/main），显示效果如图4所示。



图3 GlSurfaceProvider的使用demo

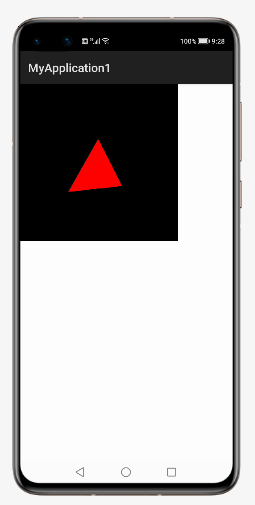


图4 GlSurfaceProvider显示效果

1. **EGL和ETC1的使用示例**

为了验证了EGL的显示效果，在initView()方法中给Viewport设置了紫色方块视图属性（见开源代码：https://gitee.com/iscas-ohos/OpenGL\_ISRC/tree/main）。

为了验证ETC1的有效性，将ETC1.getEncodedDataSize()的结果输出到一个Text中进行显示。

当EGL显示紫色方块，Text上显示ETC1.getEncodedDataSize()的结果时，证明EGL和ETC1类可成功使用，二者的显示效果如图6所示。

****

图5 EGL和ETC1的使用demo

****

图6：EGL的显示效果、ETC1的输出效果

## 项目贡献人

戴研 郑森文 朱伟 陈美汝