<https://blog.csdn.net/yang8456211/article/details/51331358>

总体来说Unity与Android之间的互相调用，是通过Unity提供的classes.jar来实现的。

**1）Unity调用Android**

在java中写函数：

**public** String u3dCall(String param, **int** p1) {

Log.*i*("Unity", "u3d call, p1 = " + p1);

**return** "I'm results";

}

Unity中调用：

Debug.Log("call java start");

AndroidJavaClass jc = new AndroidJavaClass("com.unity3d.player.UnityPlayer");**// new了一个UnityPlayer类**

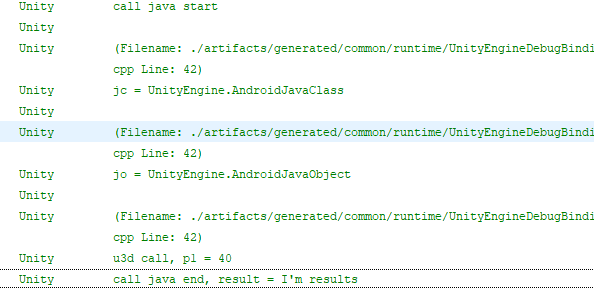
Debug.Log("jc = " + jc);

AndroidJavaObject jo = jc.GetStatic<AndroidJavaObject>("currentActivity");

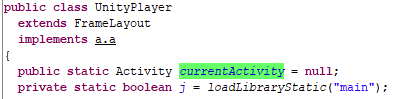
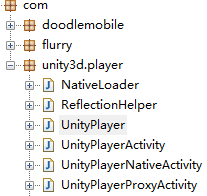
Debug.Log("jo = " + jo);

string result = jo.Call<string>("u3dCall", "nihao", 40);

Debug.Log("call java end, result = " + result);



我们通过apk的反编译可以看到UnityPlayer实际上是一个FrameLayout



Call是Unity3d调用Android所设置的API之一，还有一些常用的：



**2）Android调用Unity**

如果想要异步发送消息回Unity那么应该怎么处理呢？就是使用Unity3D的UnitySendMessage发送回去

在java中onCreate函数中调用：

mUnityPlayer.*UnitySendMessage*("Directional Light", "FromAndroid", "param from android");

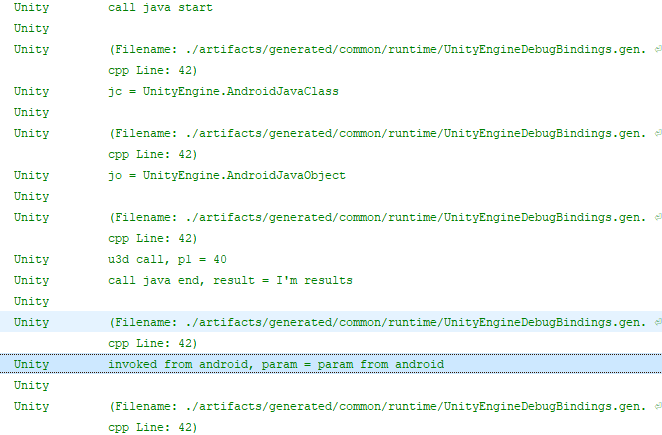
在Unity的Start中调用：

public void FromAndroid(string param)

{

Debug.Log("invoked from android, param = " + param);

}



调用的先后顺序

