

# Unity Export 导出到 Android Studio 再打包大致过程

deepwaterooo

December 18, 2022

## Contents

<b>1 UnityPlayerActivity 中 onDestroy() 时 mUnityPlayer 会杀死整个进程的问题 [爱表哥, 爱生活]</b>	<b>1</b>
1.1 相关部分的源码:	1
1.1.1 UnityPlayerActivity	1
1.1.2 UnityPlayer	2
1.2 解决办法: 网上搜索到几种解决方法, 及其相关后续可能存在的问题	2
1.2.1 自定义 UnityPlayer, 绕过杀进程的部分	2
1.2.2 因为还没有如此实现, 不知道还会有哪些后续的问题	3
1.2.3 与上面第一种方法同理, 接上面的 (继上面两步之后), 别人还使用过的方法包括:	3
1.2.4 另一种: 特制结束游戏端活动广播; 调用安卓端活动以借其发送结束游戏端活动广播; 安卓端特制类专门负责接收广播, 并结束游戏端活动, 感觉很绕, 性能?	3
1.2.5 把 UnityPlayerActivity 放在独立的进程, 也会带来后续多进程数据交互相关的问题?	4
<b>2 安卓导.jar .aar 进 unity 几个细巧点</b>	<b>5</b>
2.1 项目的 build.gradle	5
2.2 修改 AndroidManifest.xml	6
2.3 导文件 (打.jar 包给 unity 用)	6
<b>3 some commands use as reference for later debugging</b>	<b>7</b>
<b>4 导出的 unity 项目文件大致是这样的</b>	<b>7</b>
<b>5 Android 启动运行 unity</b>	<b>8</b>
5.1 在 unity 的 AndroidManifest.xml 文件	8
5.2 在 app 的 AndroidManifest.xml 文件里, 在图中位置加入这两行代码:	9
5.3 在 app 的 build.gradle 里加入这行代码。	9
5.4 在 app 的 main->res->values->strings.xml 里加入这行代码	10
5.5 点击按钮启动 unity(画蛇添足)	11
5.6 在 build.gradle 中申明包裹类名称	12
<b>6 启动运行</b>	<b>13</b>
<b>7 Android Studio 类库中重复类的修复</b>	<b>13</b>
<b>8 安卓 Android Studio 库包中有依赖的库包的解决方案 7.2.2</b>	<b>14</b>
8.1 比较好一点的, 是如下: 在项目的根目录的 build.gradle 里申明类库 unityLibrary 的依赖的文件路径就可找到	14
8.2 下面的只是一种解决方案, 可能还不是很好	16
8.3 在你工程根目录下新建一个文件夹 <b>unitylibs</b> , 将你的 aar 文件放入, 然后在该目录下新建一个 build.gradle 文件	16





- 我因为还没有如此实现, 不知道还会有哪些后续的问题
- 其它网友提到过的还存在的问题包括: **[这些, 需要用自己的游戏来验证一下]**
  - 项目中最近在做一个 AR 扫行标 (银行 Logo) 抽奖的活动, 动画模型依托 Unity 环境来展现的。Android 集成 Unity 很简单 (比 iOS 方便太多), 但是在使用过程中还是需要到一点小麻烦。
  - 需求中要求扫描页面可以反复进出, 在 @Override protected void onDestroy () 生命周期的回调方法中需要调用 mUnityPlayer.quit(); 来达到退出 Unity 的目的。但是发现 mUnityPlayer.quit(); 方法会杀死当前进程,
  - 不调用 mUnityPlayer.quit(); 方法, 控制台中又会提示一个服务没有被清理, 会发生内存泄漏,
  - 网上的做法是集成 UnityPlayer 写一个新类, 重写掉 kill() 方法, 不杀死进程, 但是实验了之后发现没有效果, 依然会有内存泄漏的提示, 虽然大部分机型可以正常使用, 但是测试中小米和其他品牌的部分机型在反复进出几次 Unity 程序后, 会闪退。**[感觉这里的问题没有追查到底, 是资源完全没能释放, 还是部分释放了? 对于出问题的手机型号, 没有提供或作进一步的征查资源释放过程中的问题. 如果我能拿到日志, 会想到分析得再彻底一点儿]**
- 自己的项目暂时只实现到这里, 会留意是否会有后续问题, 再考虑是否游戏端活动放独立进程

### 1.2.3 与上面第一种方法同理, 接上面的 (继上面两步之后), 别人还使用过的方法包括:

- 原理同样是绕过 unity 游戏端的桥接类 UnityPlayerActivity UnityPlayer 中杀掉进程的这一步, 只是是通过调用安卓端实现的方法来结束 unity 游戏端活动
- 在 unity 中调用 android 接口返回上级 activity

```
private static void BackToShelfFromAndroid() {
    using (AndroidJavaClass jc = new AndroidJavaClass("com.dps.core.UnityInterface")) {
        jc.CallStatic("backToShelf");
    }
}
```

- 实现 android 接口:

```
public static void backToShelf() {
    Log.i("unity", "finish unityPlayer ");
    UnityPlayer.currentActivity.finish(); // 这里就是正常安卓活动的 finish() 方法
}
```

### 1.2.4 另一种: 特制结束游戏端活动广播; 调用安卓端活动以借其发送结束游戏端活动广播; 安卓端特制类专门负责接收广播, 并结束游戏端活动, 感觉很绕, 性能?

- 再看最后一个游戏端放独立进程的作比较, 就会发现这里安卓端的发送广播与接收广播都是很绕的自找麻烦的浪费性能的过节, 这里应该完全可以不用多一步广播的定义, 注册发送与接收
- 创建一个 UnityReaderActivity 继承 UnityPlayerActivity, 在此注册一个广播接收, 接收到广播后 finish 此 UnityReaderActivity。在 onPause 中用 super.onDestroy() 替换掉 super.onPause()。

// 这个类只做一件事: 专门负责接收游戏端发送过来, 要求结束游戏端活动的广播, 接收后结束游戏端活动 (绕过游戏端的杀进程)

```
public class UnityReaderActivity extends UnityPlayerActivity {
    // 自身的引用
    private BroadcastReceiver receiver;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        registBroadcast();
    }
}
```



- 接下来，按照下面步骤进行：[这可以尝试作为另一种方法的实现，不是加法，是或者]
  - 第一步，在 activity 里定义一个 unityFinish 方法。
  - 第二步，在 unity 脚本中调用 activity 里定义好的 unityFinish 方法。(如不知道怎么调用请移步到：Android 与 Unity 的交互) **[同上面的安卓端的广播定义注册发送与接收相比，省性能了]**
  - 第三步，activity 的 unityFinish 方法里处理一些结束操作，最后调用 Activity 的 finish()。 **[可以成功绕过那个杀死自己进程的步骤]**
  - 第四步，在 activity 的 destroy 方法里调用 UnityPlayer 的 quit()。 **[在必要需要的进候，仍然进行了必要的资源释放，只是不知道能否在有限制时间里清理干净，还是说会自动延迟，或是程序员给足它时间清理干净呢?]**
- 这样你需要进行结束要处理的事情放到了自己定义的方法里进行了处理，处理完后调用 Activity 的 finish()，这样 activity 开始走自己结束的方法，在 Activity 的 destroy 方法里调用 super.onDestroy() 之前调用 mUnityPlayer.quit()，这个时候会先走 UnityPlayer 的 quit() 处理后事，这样即使把自己的宿主进程 kill 了，用户也感知不到了。
- 缺点：后面的四个步骤感觉逻辑还比较清晰，可是感觉作者可能也没能理解透彻，步骤有多余，当游戏端活动处于独立的进程，那杀进程就杀呀，它反正是在独立的进程，死了那个游戏端活动的进程，不影响其它安卓端进程呀，它的前后两部分其实应该可以独立到两种不同的实现方法里去的，而不是同时处理两种方法???
- 独立进程可能带来的问题: <https://www.jianshu.com/p/046590c1e6b3>

```
<activity
    android:name="cn.com.global_vision_ar.UnityPlayerActivity"
    android:screenOrientation="portrait"
    android:process=":unityAR"
/>
```

- 这个办法解决了闪退问题，但是还没有结束，UnityPlayerActivity 中需要接收 AR 程序的回调方法，如：public void getGift()，在这个方法中请求后台接口，发起抽奖。Unity 程序改成用新进程启动后，发起抽奖接口时，发生了会话丢失的情况（抽奖是需要登录的，后台会话是基于 Cookie 中的 JsessionId）。我猜是因为不在同一个进程，Cookie 没有同步，导致送到后台的 JsessionId 不一致（不是登录成功后返回的 JsessionId）。
- 解决办法是使用 Broadcast 跨进程通信，把抽奖的请求写在主进程，要抽奖时发起广播进行抽奖，抽奖结束后得到抽奖结果的数据，再通过广播的方式把数据传回 UnityPlayerActivity，使用 UnityPlayer.UnitySendMessage 方法通知 Unity 程序。

## 2 安卓导.jar .aar 进 unity 几个细巧点

### 2.1 项目的 build.grale

```
plugins {
    id 'com.android.library' // <----- 可以直接把 app 改成早类库就可以了，不同再导一个
}
android {
    namespace 'com.deepwaterooo.dwsdk' // <----- 这个需要
    compileSdk 32
    defaultConfig {
        minSdk 25
        targetSdk 32
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
        }
    }
}
```

```
    }  
    compileOptions {  
        sourceCompatibility JavaVersion.VERSION_1_8  
        targetCompatibility JavaVersion.VERSION_1_8  
    }  
}  
  
dependencies {  
    compileOnly files('libs\\classes.jar') // <<<<<<<<<<<<<< compileOnly Unity 里的这些类是分 Mono il2cpp 的，要区分  
    implementation 'androidx.appcompat:appcompat:1.5.1'  
    implementation 'com.google.android.material:material:1.7.0'  
    implementation 'androidx.constraintlayout:constraintlayout:2.1.4'  
    testImplementation 'junit:junit:4.13.2'  
    androidTestImplementation 'androidx.test.ext:junit:1.1.4'  
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.5.0'  
}
```

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.deepwaterooo.dwsdk">
    <!-- 上面的包名名称 -->

    <uses-permission android:name="android.permission.READ_PRIVILEGED_PHONE_STATE" />

    <application>
        <!-- 这下面原本是 application 里的，都不要了 -->
        <!-- android:allowBackup="true" -->
        <!-- android:dataExtractionRules="@xml/data_extraction_rules" -->
        <!-- android:fullBackupContent="@xml/backup_rules" -->
        <!-- android:icon="@mipmap/ic_launcher" -->
        <!-- android:label="@string/app_name" -->
        <!-- android:roundIcon="@mipmap/ic_launcher_round" -->
        <!-- android:supportRtl="true" -->
        <!-- android:theme="@style/Theme.MyApplication" -->
        <!-- tools:targetApi="31" -->

        <!-- 这个类的名字要写全，不写全有可能会不会出现找不到类 -->
        <activity
            android:name="com.deepwaterooo.dwsdk.MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
            <!-- <meta-data -->
            <!-- android:name="android.app.lib_name" -->
            <!-- android:value="" /> -->
            <!-- 将上面缺省的，改成下面的，这里的原理还没有弄懂，为什么一定要这么写 -->
            <meta-data
                android:name="unityplayer.UnityActivity"
                android:value="true" />

        </activity>
    </application>
</manifest>
```

- 最后-选择菜单栏 **Build -> Make Moudle** ‘你的项目名称’，build 完成后会在项目中创建一个 build 文件夹，在 **build/intermediates/aar\_main\_jar/debug/** 选择 **classes.jar** 包，用解压软件打开，删除里面 unity3d 文件夹，避免在 unity 发布时重复。
- 然后重新打成是 jar 包，导入 unity 中；将 **AndroidManifest.xml** 也导入 unity。
- 如果导入的是 aar，有以下几点需要注意：这个需要自己打一次测试一下
  - 删除 unity\_classes.jar 冲突文件，在 aar 的 lib 里
  - 删除包根目录下面的 Android 配置文件（先复制一份出来）
  - 删除 aar 跟目下的 classes.jar 里面的 BuildConfig.class 文件



### 3 some commands use as reference for later debugging

```
jar cvf classes.jar -C classes/ .

# java 8 的打包工具地址
/mnt/c/Program Files/java/jdk1.8.0_351/bin/jar.exe cvf classes.jar -C classes/ .

cp classes.jar /mnt/f/tetris3D/trunk/client/trunk/Assets/deepwaterooo/Plugins/Android

jar cvf deepwaterooo_activity.jar -C classes/ .

cp deepwaterooo_activity.jar /mnt/f/unityProjects/dwsdkTest/Assets/Plugins/Android

zip -r dwsdk-debug.zip ./*

cp dwsdk-debug.aar /mnt/f/tetris3D/trunk/client/trunk/Assets/Plugins/Android/

cp deepwaterooo_activity.jar /mnt/f/tetris3D/trunk/client/trunk/Assets/Plugins/Android/

jar cvf com.android.support.support-compat-26.1.0.aar -C com.android.support.support-compat-26.1.0/ .

# gf 首先检查是否 crash 掉了
grep -nE "FATAL EXCEPTION:" log1.log

# lst 抓取最后一个
tac log1.log | awk '!flag; /FATAL EXCEPTION: main/{flag = 1};' | tac > cur.log

# tl 抓取从某个时间点开始的
tac log1.log | awk '!flag; /:00/{flag = 1};' | tac > cur.log

grep -nE "com.unity3d.player|UnityPlayerActivity|GameApplication" cur.log > tmp.log







grep -nE "com.unity3d.player | UnityPlayerActivity" cur.log

grep -nE "com.defaultcompany.trunk | UnityPlayerActivity" cur.log
```

### 4 导出的 unity 项目文件大致是这样的

- 大致过程记一下, 用作参考, 原理还没有吃透, 细节又比较多, 容易忘记. 作个笔记记一下, 给自己用作参考

F (F:) > tmp > prevTetris19 >

Name	Date modified
 launcher	11/23/2022 1
 unityLibrary	11/23/2022 1
 build.gradle	11/23/2022 1
 gradle.properties	11/23/2022 1
 local.properties	11/23/2022 1
 settings.gradle	11/23/2022 1

- 下面是 2019 年的版本可以打出两个文件夹, 一个主工程, 一个类库的导出包,2017 年我用的版本打不出来, 还需要想得再深一点多点儿, 到可以按照这个笔记过程打包才行



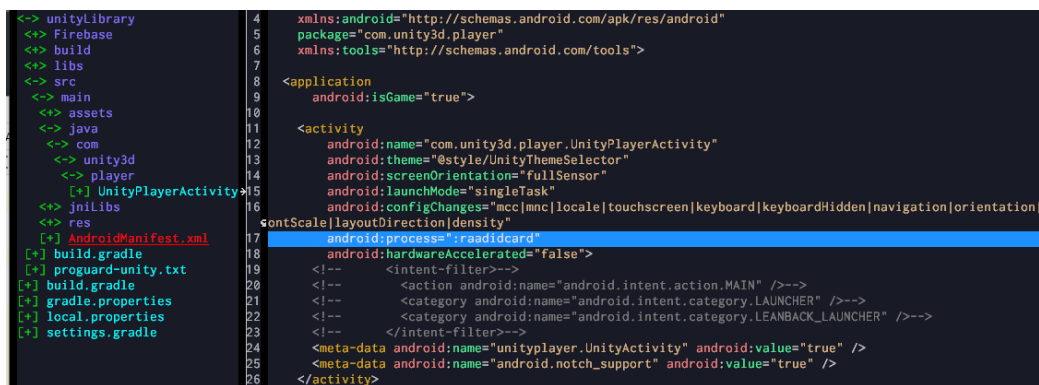
- 原始自己参考的项目是用 2017 版本的, 当时没有吃透这里的构建关系, 当时以为只能用 2017 的 unity 和 2017 的 Visual Studio 才能开发. 现在知道 2019 的版本能够导出自己可以调试的 Android Studio 项目, 而 unity 2017 版本的导出来自己还仍不知道该如何从 Android Studio 打包, 那么就暂时先用 2019 的版本, 先试图打出在安卓设备上可运行的包, 才能 move on.

## 5 Android 启动运行 unity

### 5.1 在 unity 的 AndroidMainfest.xml 文件

- 把 <intent-filter>-> 删掉或者注释掉, 留着的话, 当我们把程序运行到手机或者模拟机上时会有两个图标。
- 其次是在 <activity> 里加入这行代码, 实现多线程, 避免在从 unity 返回 Android 时也将 Android 界面也结束了。

android:process=":raadidcard"



### 5.2 在 app 的 AndroidMainfest.xml 文件里, 在图中位置加入这两行代码:

xmlns:tools="http://schemas.android.com/tools"

tools:replace="android:icon,android:theme,android:allowBackup"

- 可以成片复制的代码如下:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.unity3d.player">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        tools:replace="android:icon,android:theme,android:allowBackup"
        android:theme="@style/Theme.Test"
        tools:targetApi="31">

        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

```

        <meta-data
            android:name="android.app.lib_name"
            android:value="" />
    </activity>

</application>
</manifest>

```

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3
4     xmlns:tools="http://schemas.android.com/tools"
5     package="com.unity3d.player">
6
7     <application
8         android:allowBackup="true"
9         android:dataExtractionRules="@xml/data_extraction_rules"
10        android:fullBackupContent="@xml/backup_rules"
11        android:icon="@mipmap/ic_launcher"
12        android:label="@string/app_name"
13        android:roundIcon="@mipmap/ic_launcher_round"
14        android:supportRtl="true"
15        tools:replace="android:icon,android:theme,android:allowBackup"
16        android:theme="@style/Theme.Test"
17        tools:targetApi="31">
18        <activity
19            android:name=".MainActivity"
20            android:exported="true">
21            <intent-filter>
22                <action android:name="android.intent.action.MAIN" />
23
24                <category android:name="android.intent.category.LAUNCHER" />
25            </intent-filter>
26
27            <meta-data
28                android:name="android.app.lib_name"
29                android:value="" />
30        </activity>

```

### 5.3 在 app 的 build.gradle 里加入这行代码。

```

ndk {
    abiFilters 'armeabi-v7a'
}

```

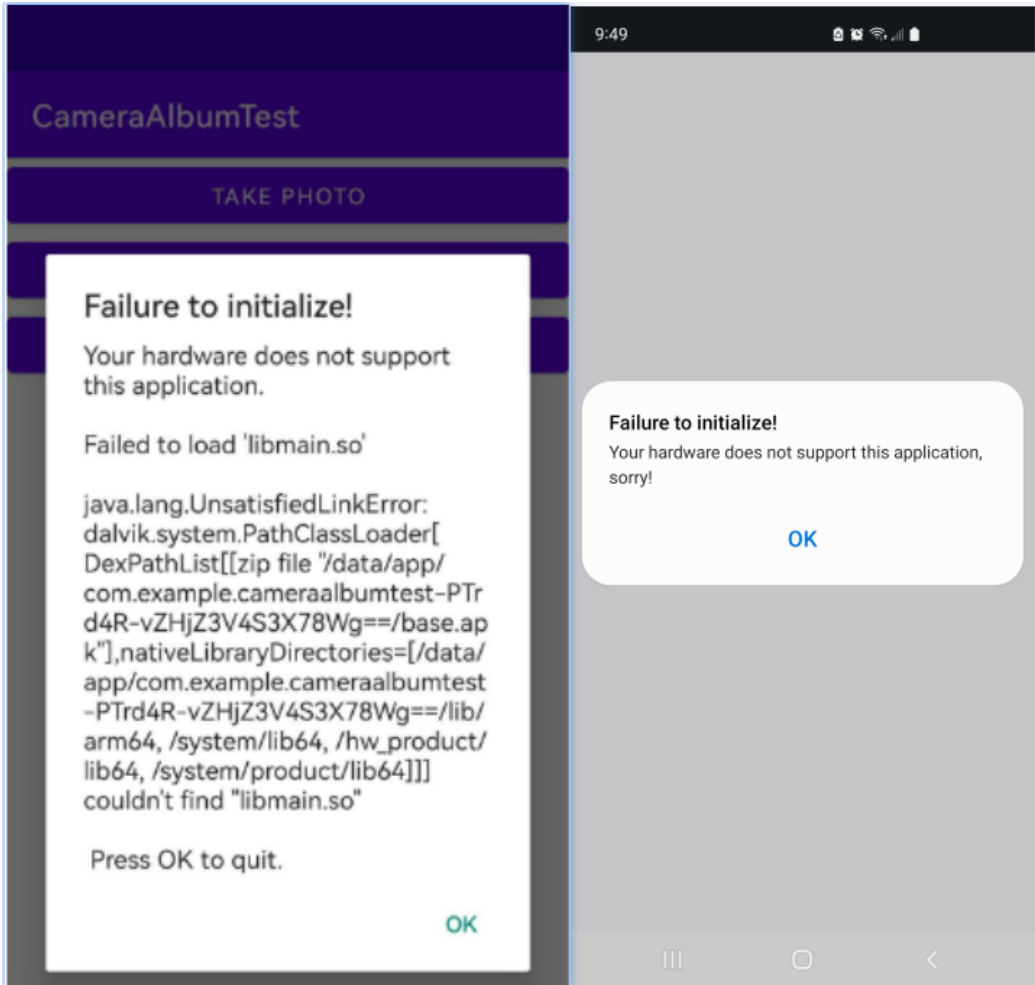
```

1 id 'com.android.application'
2
3 }
4
5 android {
6     namespace 'com.unity3d.player'
7     compileSdk 32
8
9     defaultConfig {
10        applicationId "com.unity3d.player"
11        minSdk 25
12        targetSdk 31
13        versionCode 1
14        versionName "1.0"
15
16        ndk {
17            abiFilters 'armeabi-v7a'
18        }
19
20        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
21
22        buildTypes {
23            release {
24                minifyEnabled false
25                proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
26            }
27        }
28    }

```

### 5.4 在 app 的 main->res->values->strings.xml 里加入这行代码

- 都还没有去想, 这句话能起到什么作用, 应该是关系不大, 或是可以跳过绕过的小细节
- 进行这两步操作的原因是, 我在运行到手机时, 他显示硬件不支持或者闪退。加入上面两个代码后就可以正常启动 unity。
- 我个人认为真正起作用的是上一步关于手机架构的设置的 ndk 那三行, 与上面字符串无关, 应该是无关的



## 5.5 点击按钮启动 unity(画蛇添足)

- 感觉这个连接过程对于自己的项目就是画蛇添足. 可是如何既能避开这一步, 又能两者很好的平滑交互呢? 对于现在的自己, 是个问题和挑战
- 在主工程的 `activity_main.xml` 文件里添加一个按钮。`MainActivity.java` 里加入启动事件, 如果在这里 `layout` 标红的话, 就把鼠标移到 `layout` 下面, 建立一个 `layout` 就行, 我分析是主工程的问题, 这个影响不大

```
<Button
    android:id="@+id/showUnityBtn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Show Unity"/>
```

```

f:/tmp/test/
<=> app
<=> build
<=> libs
<=> src
<=> androidTest
<=> main
<=> java
<=> res
    <=> drawable
    <=> drawable-v24
    <=> layout
        [+] activity_main.xml
    <=> mipmap-anydpi-v26
    <=> mipmap-hdpi
    <=> mipmap-mdpi
    <=> mipmap-xhdpi
    <=> mipmap-xxhdpi
    <=> mipmap-xxxhdpi
    <=> values
        [+] colors.xml
        [+] strings.xml
        [+] themes.xml
    <=> xml
    [+] AndroidManifest.xml
<=> test

```

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <!-- <androidx.constraintlayout.widget.ConstraintLayout -->
3 <LinearLayout
4     xmlns:android="http://schemas.android.com/apk/res/android"
5     xmlns:app="http://schemas.android.com/apk/res-auto"
6     xmlns:tools="http://schemas.android.com/tools"
7     android:layout_width="match_parent"
8     android:layout_height="match_parent"
9     tools:context=".MainActivity">
10
11     <!-- <TextView -->
12     <!--     android:layout_width="wrap_content" -->
13     <!--     android:layout_height="wrap_content" -->
14     <!--     android:text="Hello World!" -->
15     <!--     app:layout_constraintBottom_toBottomOf="parent" -->
16     <!--     app:layout_constraintEnd_toEndOf="parent" -->
17     <!--     app:layout_constraintStart_toStartOf="parent" -->
18     <!--     app:layout_constraintTop_toTopOf="parent" /> -->
19
20     <Button
21         android:id="@+id/showUnityBtn"
22         android:layout_width="match_parent"
23         android:layout_height="wrap_content"
24         android:text="Show Unity"/>
25 </LinearLayout>
26 <!-- </androidx.constraintlayout.widget.ConstraintLayout -->
27

```

- MainActivity.cs 里的回调设置

```

Button btn = (Button)findViewById(R.id.showUnityBtn);
btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

```

```

// <----- UnityPlayerActivity <= com.unity3d.player 这里就是刚刚那个包名奇怪的地方，要不然 找不到 下面的 UnityPla
Intent intent = new Intent(MainActivity.this, UnityPlayerActivity.class); // <----- UnityPlayerAc

startActivity(intent);
}
});

```

```

f:/tmp/test/
<=> app
<=> build
<=> libs
<=> src
<=> androidTest
<=> main
<=> java
<=> com
    <=> unity3d
    <=> player
        [+] MainActivity.java
<=> res
    <=> drawable
    <=> drawable-v24
    <=> layout
        [+] activity_main.xml
    <=> mipmap-anydpi-v26
    <=> mipmap-hdpi
    <=> mipmap-mdpi
    <=> mipmap-xhdpi
    <=> mipmap-xxhdpi
    <=> mipmap-xxxhdpi
    <=> values
        [+] colors.xml
        [+] strings.xml
        [+] themes.xml

```

```

1 package com.unity3d.player; // <-----
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import androidx.constraintlayout.widget.ConstraintLayout;
5 import androidx.core.app.ActivityCompat;
6 import androidx.core.content.ContextCompat;
7 import androidx.core.view.WindowCompat;
8 import androidx.core.view.WindowInsetsCompat;
9
10 public class MainActivity extends AppCompatActivity {
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15
16         Button btn = (Button)findViewById(R.id.showUnityBtn);
17         btn.setOnClickListener(new View.OnClickListener() {
18             @Override
19             public void onClick(View view) {
20                 // <----- UnityPlayerActivity <= com.unity3d.player 这里就是刚刚那个包名奇怪的地方，要不然 找不到 下面的 UnityPlayerActivity 类
21                 Intent intent = new Intent(MainActivity.this, UnityPlayerActivity.class); // <----- UnityPlayerAc
22
23                 startActivity(intent);
24             }
25         });
26     }
27 }

```

## 5.6 在 build.gradle 中申明包裹类名称

- 说是现在在 AndroidManifest.xml 里申明包裹名称已经过时了，要在配置文件里申明，于是我在这里申明的：

```

android {
    namespace 'com.unity3d.player'
}

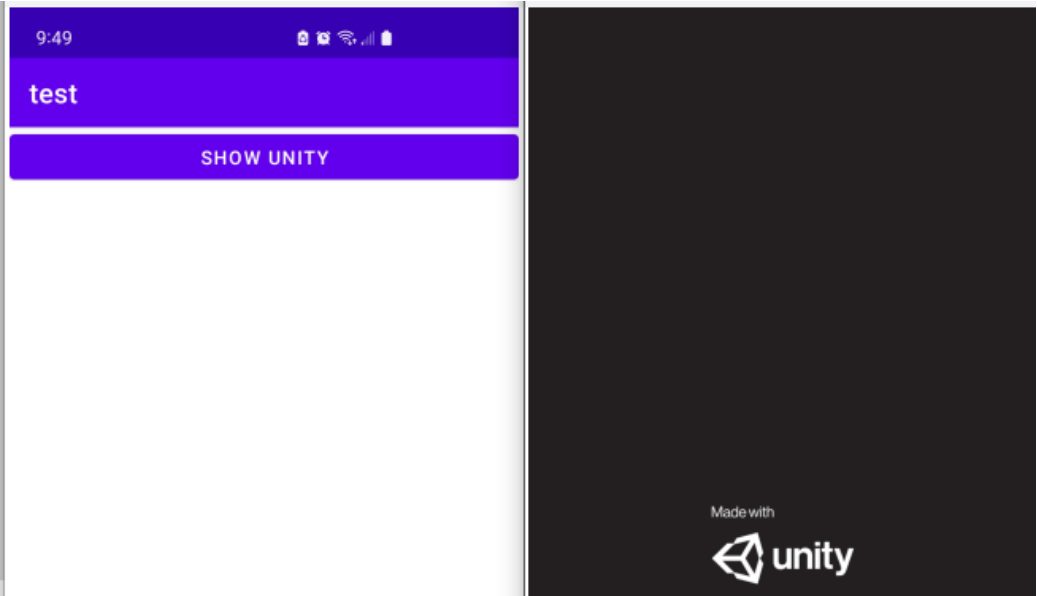
```

```

<-> unitylibrary
<-> Firebase
<-> libs
<-> src
<-> main
<-> assets
<-> java
<-> jnilibs
<-> res
[+] AndroidManifest.xml
[+] build.gradle
[+] proguard-unity.txt
[+] build.gradle
[+] gradle.properties
[+] local.properties
[+] settings.gradle
42 android {
43     namespace 'com.unity3d.player'
44
45     compileSdkVersion 32
46     buildToolsVersion '31.0.0'
47
48     compileOptions {
49         sourceCompatibility JavaVersion.VERSION_11
50         targetCompatibility JavaVersion.VERSION_11
51     }
52
53     defaultConfig {
54         minSdkVersion 31
55         targetSdkVersion 31
56         ndk {
57             abiFilters 'armeabi-v7a'

```

## 6 启动运行



## 7 Android Studio 类库中重复类的修复

```

Duplicate class com.airbnb.lottie.LottieAnimationView found in modules jetified-MiyataOpenUISdk-1.0.2-run
Duplicate class com.airbnb.lottie.LottieAnimationView$SavedState found in modules jetified-MiyataOpenUISdk-1.0.2-run
Duplicate class com.qq.e.ads.nativ.widget.NativeAdContainer found in modules jetified-GDTSdk.unionNormal
Go to the documentation to learn how to Fix dependency resolution errors.

```

- 如果新导入的依赖库发生了 Duplicate class android.xx.xx 这种类型的报错可能就是两个库导入了重复的类，这时候只需要把 build.gradle 中新导入的依赖做如下处理

```

implementation ('com.xxx.xxx.xx:xx:1.0.0'){
    exclude group: "com.xxxx.xxxx"
}

```
- 上面这个方法我还没有试，下面的试过了可行
- 对，就是把新导入的依赖库的后面加上大括号并把重复导入包名填入相应的位置就可以解决了，有时候可能会好几个依赖库都重复了，这就比较难判断了
- 1. 把 MiyataOpenUISdk-1.0.2.aar 改后缀成 zip，得到解压后的 MiyataOpenUISdk-1.0.2 文件夹，里面包含 classes.jar 和 res 等。

F (F:) > tmp > test > unityLibrary > libs > tmp > com.android.support.support-core-utils-26.1.0 >

Name	Date modified	Type	Size
aidl	9/13/2017 5:08 AM	File folder	
assets	9/13/2017 5:09 AM	File folder	
jni	9/13/2017 5:09 AM	File folder	
libs	9/13/2017 5:09 AM	File folder	
res	9/13/2017 5:08 AM	File folder	
AndroidManifest.xml	9/13/2017 5:08 AM	XML File	2 KB
classes.jar	9/13/2017 5:09 AM	Executable Jar File	93 KB
R.txt	9/13/2017 5:08 AM	TXT File	5 KB

- 2. 同理把 classes.jar 改后缀成 zip，解压后得到 classes 文件夹，找到冲突的包，直接删除整个文件夹，如图
- 3. 使用 jar 命令重新对 classes 文件夹打包成 jar，并替换掉之前的 classes.jar。

```
jar cvf classes.jar -C classes/ .
```

- 4. 同理，使用 jar 命令重新对 MiyataOpenUISdk-1.0.2 文件夹打包成 aar，得到的 newMiyataOpenUISdk.aar 即可使用。

```
jar cvf com.android.support.support-compat-26.1.0.aar -C com.android.support.support-compat-26.1.0/ .
```

## 8 安卓 Android Studio 库包中有依赖的库包的解决方案 7.2.2

Direct local .aar file dependencies are not supported when building an AAR.

- 在高版本的 AndroidStudio 并且使用了版本的 gradle 出现了上述问题可以按着如下引用

### 8.1 比较好一点的, 是如下: 在项目的根目录的 build.gradle 里申明类库 unityLibrary 的依赖的文件路径就可找到

```
allprojects {
    buildscript {
        repositories {
            google()
            jcenter()
        }

        dependencies {
            classpath 'com.android.tools.build:gradle:7.2.2'
        }
    }

    repositories {
        google()
        jcenter()
        flatDir {
            dirs "${project(':unityLibrary').projectDir}/libs"
        }
    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}
```

```

f:/tmp/test/
<=> app
<=> gradle
<=> unityLibrary
<=> unitylibs
[+] build.gradle
[+] gradle.properties
[+] local.properties
[+] settings.gradle

1 allprojects {
2     buildscript {
3         repositories {
4             google()
5             jcenter()
6         }
7
8         dependencies {
9             classpath 'com.android.tools.build:gradle:7.2.2'
10        }
11    }
12
13    repositories {
14        google()
15        jcenter()
16
17        flatDir {
18            dirs "${project(':unityLibrary').projectDir}/libs"
19        }
20    }
21
22    task clean(type: Delete) {
23        delete rootProject.buildDir
24    }
25
26    // Top-level build file where you can add configuration options common to all
27    // plugins {
28
29        lintOptions {
30            abortOnError false
31        }
32
33        aaptOptions {
34            ignoreAssetsPattern = "!svn:!git:!ds_store:!*.scc:.*!CVS:!thumbs.db:!pico"
35        }
36
37        packagingOptions {
38            doNotStrip '*/*armeabi-v7a/*.so'
39        }
40
41        dependencies {
42            implementation fileTree(dir: 'libs', include: ['*.jar'])
43
44        }
45    }
46
47    ndk {
48        abiFilters 'armeabi-v7a'
49    }
50
51    testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
52
53    buildTypes {
54        release {
55            minifyEnabled false
56            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'),
57                'proguard-rules.pro'
58        }
59    }
60
61    compileOptions {
62        sourceCompatibility JavaVersion.VERSION_1_8
63        targetCompatibility JavaVersion.VERSION_1_8
64    }
65
66    dependencies {
67
68        implementation 'androidx.appcompat:appcompat:1.5.1'
69        implementation 'com.google.android.material:material:1.7.0'
70        implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
71
72        implementation project(path: ':unityLibrary')
73
74        testImplementation 'junit:junit:4.13.2'
75    }
76
77    }
78
79    }
80
81    }
82
83    }
84
85    }
86
87    }
88
89    }
90
91    }
92
93    }
94
95    }
96
97    }
98
99    }
100
101    }
102
103    }
104
105    }
106
107    }
108
109    }
110
111    }
112
113    }
114
115    }
116
117    }
118
119    }
120
121    }
122
123    }
124
125    }
126
127    }
128
129    }
130
131    }
132
133    }
134
135    }
136
137    }
138
139    }
140
141    }
142
143    }
144
145    }
146
147    }
148
149    }
150
151    }
152
153    }
154
155    }
156
157    }
158
159    }
160
161    }
162
163    }
164
165    }
166
167    }
168
169    }
170
171    }
172
173    }
174
175    }
176
177    }
178
179    }
180
181    }
182
183    }
184
185    }
186
187    }
188
189    }
190
191    }
192
193    }
194
195    }
196
197    }
198
199    }
200
201    }
202
203    }
204
205    }
206
207    }
208
209    }
210
211    }
212
213    }
214
215    }
216
217    }
218
219    }
220
221    }
222
223    }
224
225    }
226
227    }
228
229    }
230
231    }
232
233    }
234
235    }
236
237    }
238
239    }
240
241    }
242
243    }
244
245    }
246
247    }
248
249    }
250
251    }
252
253    }
254
255    }
256
257    }
258
259    }
260
261    }
262
263    }
264
265    }
266
267    }
268
269    }
270
271    }
272
273    }
274
275    }
276
277    }
278
279    }
280
281    }
282
283    }
284
285    }
286
287    }
288
289    }
290
291    }
292
293    }
294
295    }
296
297    }
298
299    }
300
301    }
302
303    }
304
305    }
306
307    }
308
309    }
310
311    }
312
313    }
314
315    }
316
317    }
318
319    }
320
321    }
322
323    }
324
325    }
326
327    }
328
329    }
330
331    }
332
333    }
334
335    }
336
337    }
338
339    }
340
341    }
342
343    }
344
345    }
346
347    }
348
349    }
350
351    }
352
353    }
354
355    }
356
357    }
358
359    }
360
361    }
362
363    }
364
365    }
366
367    }
368
369    }
370
371    }
372
373    }
374
375    }
376
377    }
378
379    }
380
381    }
382
383    }
384
385    }
386
387    }
388
389    }
390
391    }
392
393    }
394
395    }
396
397    }
398
399    }
400
401    }
402
403    }
404
405    }
406
407    }
408
409    }
410
411    }
412
413    }
414
415    }
416
417    }
418
419    }
420
421    }
422
423    }
424
425    }
426
427    }
428
429    }
430
431    }
432
433    }
434
435    }
436
437    }
438
439    }
440
441    }
442
443    }
444
445    }
446
447    }
448
449    }
450
451    }
452
453    }
454
455    }
456
457    }
458
459    }
460
461    }
462
463    }
464
465    }
466
467    }
468
469    }
470
471    }
472
473    }
474
475    }
476
477    }
478
479    }
480
481    }
482
483    }
484
485    }
486
487    }
488
489    }
490
491    }
492
493    }
494
495    }
496
497    }
498
499    }
500
501    }
502
503    }
504
505    }
506
507    }
508
509    }
510
511    }
512
513    }
514
515    }
516
517    }
518
519    }
520
521    }
522
523    }
524
525    }
526
527    }
528
529    }
530
531    }
532
533    }
534
535    }
536
537    }
538
539    }
540
541    }
542
543    }
544
545    }
546
547    }
548
549    }
550
551    }
552
553    }
554
555    }
556
557    }
558
559    }
560
561    }
562
563    }
564
565    }
566
567    }
568
569    }
570
571    }
572
573    }
574
575    }
576
577    }
578
579    }
580
581    }
582
583    }
584
585    }
586
587    }
588
589    }
590
591    }
592
593    }
594
595    }
596
597    }
598
599    }
600
601    }
602
603    }
604
605    }
606
607    }
608
609    }
610
611    }
612
613    }
614
615    }
616
617    }
618
619    }
620
621    }
622
623    }
624
625    }
626
627    }
628
629    }
630
631    }
632
633    }
634
635    }
636
637    }
638
639    }
640
641    }
642
643    }
644
645    }
646
647    }
648
649    }
650
651    }
652
653    }
654
655    }
656
657    }
658
659    }
660
661    }
662
663    }
664
665    }
666
667    }
668
669    }
670
671    }
672
673    }
674
675    }
676
677    }
678
679    }
680
681    }
682
683    }
684
685    }
686
687    }
688
689    }
690
691    }
692
693    }
694
695    }
696
697    }
698
699    }
700
701    }
702
703    }
704
705    }
706
707    }
708
709    }
710
711    }
712
713    }
714
715    }
716
717    }
718
719    }
720
721    }
722
723    }
724
725    }
726
727    }
728
729    }
730
731    }
732
733    }
734
735    }
736
737    }
738
739    }
740
741    }
742
743    }
744
745    }
746
747    }
748
749    }
750
751    }
752
753    }
754
755    }
756
757    }
758
759    }
760
761    }
762
763    }
764
765    }
766
767    }
768
769    }
770
771    }
772
773    }
774
775    }
776
777    }
778
779    }
780
781    }
782
783    }
784
785    }
786
787    }
788
789    }
790
791    }
792
793    }
794
795    }
796
797    }
798
799    }
800
801    }
802
803    }
804
805    }
806
807    }
808
809    }
810
811    }
812
813    }
814
815    }
816
817    }
818
819    }
820
821    }
822
823    }
824
825    }
826
827    }
828
829    }
830
831    }
832
833    }
834
835    }
836
837    }
838
839    }
840
841    }
842
843    }
844
845    }
846
847    }
848
849    }
850
851    }
852
853    }
854
855    }
856
857    }
858
859    }
860
861    }
862
863    }
864
865    }
866
867    }
868
869    }
870
871    }
872
873    }
874
875    }
876
877    }
878
879    }
880
881    }
882
883    }
884
885    }
886
887    }
888
889    }
890
891    }
892
893    }
894
895    }
896
897    }
898
899    }
900
901    }
902
903    }
904
905    }
906
907    }
908
909    }
910
911    }
912
913    }
914
915    }
916
917    }
918
919    }
920
921    }
922
923    }
924
925    }
926
927    }
928
929    }
930
931    }
932
933    }
934
935    }
936
937    }
938
939    }
940
941    }
942
943    }
944
945    }
946
947    }
948
949    }
950
951    }
952
953    }
954
955    }
956
957    }
958
959    }
960
961    }
962
963    }
964
965    }
966
967    }
968
969    }
970
971    }
972
973    }
974
975    }
976
977    }
978
979    }
980
981    }
982
983    }
984
985    }
986
987    }
988
989    }
990
991    }
992
993    }
994
995    }
996
997    }
998
999    }
1000
1001    }
1002
1003    }
1004
1005    }
1006
1007    }
1008
1009    }
1010
1011    }
1012
1013    }
1014
1015    }
1016
1017    }
1018
1019    }
1020
1021    }
1022
1023    }
1024
1025    }
1026
1027    }
1028
1029    }
1030
1031    }
1032
1033    }
1034
1035    }
1036
1037    }
1038
1039    }
1040
1041    }
1042
1043    }
1044
1045    }
1046
1047    }
1048
1049    }
1050
1051    }
1052
1053    }
1054
1055    }
1056
1057    }
1058
1059    }
1060
1061    }
1062
1063    }
1064
1065    }
1066
1067    }
1068
1069    }
1070
1071    }
1072
1073    }
1074
1075    }
1076
1077    }
1078
1079    }
1080
1081    }
1082
1083    }
1084
1085    }
1086
1087    }
1088
1089    }
1090
1091    }
1092
1093    }
1094
1095    }
1096
1097    }
1098
1099    }
1100
1101    }
1102
1103    }
1104
1105    }
1106
1107    }
1108
1109    }
1110
1111    }
1112
1113    }
1114
1115    }
1116
1117    }
1118
1119    }
1120
1121    }
1122
1123    }
1124
1125    }
1126
1127    }
1128
1129    }
1130
1131    }
1132
1133    }
1134
1135    }
1136
1137    }
1138
1139    }
1140
1141    }
1142
1143    }
1144
1145    }
1146
1147    }
1148
1149    }
1150
1151    }
1152
1153    }
1154
1155    }
1156
1157    }
1158
1159    }
1160
1161    }
1162
1163    }
1164
1165    }
1166
1167    }
1168
1169    }
1170
1171    }
1172
1173    }
1174
1175    }
1176
1177    }
1178
1179    }
1180
1181    }
1182
1183    }
1184
1185    }
1186
1187    }
1188
1189    }
1190
1191    }
1192
1193    }
1194
1195    }
1196
1197    }
1198
1199    }
1200
1201    }
1202
1203    }
1204
1205    }
1206
1207    }
1208
1209    }
1210
1211    }
1212
1213    }
1214
1215    }
1216
1217    }
1218
1219    }
1220
1221    }
1222
1223    }
1224
1225    }
1226
1227    }
1228
1229    }
1230
1231    }
1232
1233    }
1234
1235    }
1236
1237    }
1238
1239    }
1240
1241    }
1242
1243    }
1244
1245    }
1246
1247    }
1248
1249    }
1250
1251    }
1252
1253    }
1254
1255    }
1256
1257    }
1258
1259    }
1260
1261    }
1262
1263    }
1264
1265    }
1266
1267    }
1268
1269    }
1270
1271    }
1272
1273    }
1274
1275    }
1276
1277    }
1278
1279    }
1280
1281    }
1282
1283    }
1284
1285    }
1286
1287    }
1288
1289    }
1290
1291    }
1292
1293    }
1294
1295    }
1296
1297    }
1298
1299    }
1300
1301    }
1302
1303    }
1304
1305    }
1306
1307    }
1308
1309    }
1310
1311    }
1312
1313    }
1314
1315    }
1316
1317    }
1318
1319    }
1320
1321    }
1322
1323    }
1324
1325    }
1326
1327    }
1328
1329    }
1330
1331    }
1332
1333    }
1334
1335    }
1336
1337    }
1338
1339    }
1340
1341    }
1342
1343    }
1344
1345    }
1346
1347    }
1348
1349    }
1350
1351    }
1352
1353    }
1354
1355    }
1356
1357    }
1358
1359    }
1360
1361    }
1362
1363    }
1364
1365    }
1366
1367    }
1368
1369    }
1370
1371    }
1372
1373    }
1374
1375    }
1376
1377    }
1378
1379    }
1380
1381    }
1382
1383    }
1384
1385    }
1386
1387    }
1388
1389    }
1390
1391    }
1392
1393    }
1394
1395    }
1396
1397    }
1398
1399    }
1400
1401    }
1402
1403    }
1404
1405    }
1406
1407    }
1408
1409    }
1410
1411    }
1412
1413    }
1414
1415    }
1416
1417    }
1418
1419    }
1420
1421    }
1422
1423    }
1424
1425    }
1426
1427    }
1428
1429    }
1430
1431    }
1432
1433    }
1434
1435    }
1436
1437    }
1438
1439    }
1440
1441    }
1442
1443    }
1444
1445    }
1446
1447    }
1448
1449    }
1450
1451    }
1452
1453    }
1454
1455    }
1456
1457    }
1458
1459    }
1460
1461    }
1462
1463    }
1464
1465    }
1466
1467    }
1468
1469    }
1470
1471    }
1472
1473    }
1474
1475    }
1476
1477    }
1478
1479    }
1480
1481    }
1482
1483    }
1484
1485    }
1486
1487    }
1488
1489    }
1490
1491    }
1492
1493    }
1494
1495    }
1496
1497    }
1498
1499    }
1500
1501    }
1502
1503    }
1504
1505    }
1506
1507    }
1508
1509    }
1510
1511    }
1512
1513    }
1514
1515    }
1516
1517    }
1518
1519    }
1520
1521    }
1522
1523    }
1524
1525    }
1526
1527    }
1528
1529    }
1530
1531    }
1532
1533    }
1534
1535    }
1536
1537    }
1538
1539    }
1540
1541    }
1542
1543    }
1544
1545    }
1546
1547    }
1548
1549    }
1550
1551    }
1552
1553    }
1554
1555    }
1556
1557    }
1558
1559    }
1560
1561    }
1562
1563    }
1564
1565    }
1566
1567    }
1568
1569    }
1570
1571    }
1572
1573    }
1574
1575    }
1576
1577    }
1578
1579    }
1580
1581    }
1582
1583    }
1584
1585    }
1586
1587    }
1588
1589    }
1590
1591    }
1592
1593    }
1594
1595    }
1596
1597    }
1598
1599    }
1600
1601    }
1602
1603    }
1604
1605    }
1606
1607    }
1608
1609    }
1610
1611    }
1612
1613    }
1614
1615    }
1616
1617    }
1618
1619    }
1620
1621    }
1622
1623    }
1624
1625    }
1626
1627    }
1628
1629    }
1630
1631    }
1632
1633    }
1634
1635    }
1636
1637    }
1638
1639    }
1640
1641    }
1642
1643    }
1644
1645    }
1646
1647    }
1648
1649    }
1650
1651    }
1652
1653    }
1654
1655    }
1656
1657    }
1658
1659    }
1660
1661    }
1662
1663    }
1664
1665    }
1666
1667    }
1668
1669    }
1670
1671    }
1672
1673    }
1674
1675    }
1676
1677    }
1678
1679    }
1680
1681    }
1682
1683    }
1684
1685    }
1686
1687    }
1688
1689    }
1690
1691    }
1692
1693    }
1694
1695    }
1696
1697    }
1698
1699    }
1700
1701    }
1702
1703    }
1704
1705    }
1706
1707    }
1708
1709    }
1710
1711    }
1712
1713    }
1714
1715    }
1716
1717    }
1718
1719    }
1720
1721    }
1722
1723    }
1724
1725    }
1726
1727    }
1728
1729    }
1730
1731    }
1732
1733    }
1734
1735    }
1736
1737    }
1738
1739    }
1740
1741    }
1742
1743    }
1744
1745    }
1746
1747    }
1748
1749    }
1750
1751    }
1752
1753    }
1754
1755    }
1756
1757    }
1758
1759    }
1760
1761    }
1762
1763    }
1764
1765    }
1766
1767    }
1768
1769    }
1770
1771    }
1772
1773    }
1774
1775    }
1776
1777    }
1778
1779    }
1780
1781    }
1782
1783    }
1784
1785    }
1786
1787    }
1788
1789    }
1790
1791    }
1792
1793    }
1794
1795    }
1796
1797    }
1798
1799    }
1800
1801    }
1802
1803    }
1804
1805    }
1806
1807    }
1808
1809    }
1810
1811    }
1812
1813    }
1814
1815    }
1816
1817    }
1818
1819    }
1820
1821    }
1822
1823    }
1824
1825    }
1826
1827    }
1828
1829    }
1830
1831    }
1832
1833    }
1834
1835    }
1836
1837    }
1838
1839    }
1840
1841    }
1842
1843    }
1844
1845    }
1846
1847    }
1848
1849    }
1850
1851    }
1852
1853    }
1854
1855    }
1856
1857    }
1858
1859    }
1860
1861    }
1862
1863    }
1864
1865    }
1866
1867    }
1868
1869    }
1870
1871    }
1872
1873    }
1874
1875    }
1876
1877    }
1878
1879    }
1880
1881    }
1882
1883    }
1884
1885    }
1886
1887    }
1888
1889    }
1890
1891    }
1892
1893    }
1894
1895    }
1896
1897    }
1898
1899    }
1900
1901    }
1902
1903    }
1904
1905    }
1906
1907    }
1908
1909    }
1910
1911    }
1912
1913    }
1914
1915    }
1916
1917    }
1918
1919    }
1920
1921    }
1922
1923    }
1924
1925    }
1926
1927    }
1928
1929    }
1930
1931    }
1932
1933    }
1934
1935    }
1936
1937    }
1938
1939    }
1940
1941    }
1942
1943    }
1944
1945    }
1946
1947    }
1948
1949    }
1950
1951    }
1952
1953    }
1954
1955    }
1956
1957    }
1958
1959    }
1960
1961    }
1962
1963    }
1964
1965    }
1966
1967    }
1968
1969    }
1970
1971    }
1972
1973    }
1974
1975    }
1976
1977    }
1978
1979    }
1980
1981    }
1982
1983    }
1984
1985    }
1986
1987    }
1988
1989    }
1990
1991    }
1992
1993    }
1994
1995    }
1996
1997    }
1998
1999    }
2000
2001    }
2002
2003    }
2004
2005    }
2006
2007    }
2008
2009    }
2010
2011    }
2012
2013    }
2014
2015    }
2016
2017    }
2018
2019    }
2020
2021    }
2022
2023    }
2024
2025    }
2026
2027    }
2028
2029    }
2030
2031    }
2032
2033    }
2034
2035    }
2036
2037    }
2038
2039    }
2040
2041    }
2042
2043    }
2044
2045    }
2046
2047    }
2048
2049    }
2050
2051    }
2052
2053    }
2054
2055    }
2056
2057    }
2058
2059    }
2060
2061    }
2062
2063    }
2064
2065    }
2066
2067    }
2068
2069    }
2070
2071    }
2072
2073    }
2074
2075    }
2076
2077    }
2078
2079    }
2080
2081    }
2082
2083    }
2084
2085    }
2086
2087    }
2088
2089    }
2090
2091    }
2092
2093    }
2094
2095    }
2096
2097    }
2098
2099    }
2100
2101    }
2102
2103    }
2104
2105    }
2106
2107    }
2108
2109    }
2110
2111    }
2112
2113    }
2114
2115    }
2116
2117    }
2118
2119    }
2120
2121    }
2122
2123    }
2124
2125    }
2126
2127    }
2128
2129    }
2130
2131    }
2132
2133    }
2134
2135    }
2136
2137    }
2138
2139    }
2140
2141    }
2142
2143    }
2144
2145    }
2146
2147    }
2148
2149    }
2150
2151    }
2152
2153    }
2154
2155    }
2156
2157    }
2158
2159    }
2160
2161    }
2162
2163    }
2164
2165    }
2166
2167    }
2168
2169    }
2170
2171    }
2172
2173    }
2174
2175    }
2176
2177    }
2178
2179    }
2180
2181    }
2182
2183    }
2184
2185   
```



8.2 下面的只是一种解决方案,可能还不是很好

8.3 在你工程根目录下新建一个文件夹 **unitylibs** , 将你的 **aar** 文件放入, 然后在该目录下新建一个 **build.gradle** 文件

```
f:/tmp/test/
<> app
<> gradle
<> unityLibrary
<> unitylibs
[+] build.gradle
[-] build.gradle
[+] gradle.properties
[+] local.properties
[+] settings.gradle

configurations.maybeCreate("default")
artifacts.add("default", file('android.arch.lifecycle.runtime-1.0.0.aar'))
artifacts.add("default", file('com.android.support.support-compat-26.1.0.aar'))
artifacts.add("default", file('com.android.support.support-core-ui-26.1.0.aar'))
artifacts.add("default", file('com.android.support.support-core-utils-26.1.0.aar'))
artifacts.add("default", file('com.android.support.support-fragment-26.1.0.aar'))
artifacts.add("default", file('com.android.support.support-media-compat-26.1.0.aar'))
artifacts.add("default", file('com.android.support.support-v4-26.1.0.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-ads-identifier-16.0.0.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-base-16.0.1.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-basement-16.2.0.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-flags-16.0.1.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-measurement-16.5.0.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-measurement-api-16.5.0.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-measurement-base-16.5.0.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-measurement-impl-16.5.0.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-measurement-sdk-16.5.0.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-stats-16.0.1.aar'))
artifacts.add("default", file('com.google.android.gms.play-services-tasks-16.0.1.aar'))
artifacts.add("default", file('com.google.firebase.firebase-analytics-16.5.0.aar'))
artifacts.add("default", file('com.google.firebase.firebase-app-unity-6.0.0.aar'))
artifacts.add("default", file('com.google.firebase.firebase-auth-17.0.0.aar'))
artifacts.add("default", file('com.google.firebase.firebase-auth-interop-17.0.0.aar'))
artifacts.add("default", file('com.google.firebase.firebase-auth-unity-6.0.0.aar'))
artifacts.add("default", file('com.google.firebase.firebase-common-17.0.0.aar'))
artifacts.add("default", file('com.google.firebase.firebase-database-17.0.0.aar'))
artifacts.add("default", file('com.google.firebase.firebase-database-collection-16.0.1.aar'))
artifacts.add("default", file('com.google.firebase.firebase-database-unity-6.0.0.aar'))
artifacts.add("default", file('com.google.firebase.firebase-iid-17.1.2.aar'))
artifacts.add("default", file('com.google.firebase.firebase-iid-interop-16.0.1.aar'))
artifacts.add("default", file('com.google.firebase.firebase-measurement-connector-17.0.1.aar'))
artifacts.add("default", file('UnityAds.aar'))
artifacts.add("default", file('UnityAdsAndroidPlugin.aar'))
```

8.4 在 **settings.gradle** 导入该工程

include ':unitylibs

```
f:/tmp/test/
<> app
<> gradle
<> unityLibrary
<> unitylibs
[+] build.gradle
[-] build.gradle
[+] gradle.properties
[+] local.properties
[+] settings.gradle

1 pluginManagement {
2     repositories {
3         gradlePluginPortal()
4         google()
5         mavenCentral()
6     }
7 }
8 dependencyResolutionManagement {
9     repositoriesMode.set(RepositoriesMode.FAIL_ON_PROJECT_REPOS)
10    repositories {
11        google()
12        mavenCentral()
13    }
14 }
15 rootProject.name = "test"
16 include ':app'
17 include ':unityLibrary'
18 include ':unitylibs'
```

8.5 在你需要依赖的工程里面的 **build.gradle** 中增加依赖

- // 这里需要注意的是, unitylibs 是你 aar 库所在文件夹
- ```
implementation project(path: ':unitylibs')
```

```
<-> app
<-> build
<-> libs
<-> src
[+] build.gradle
<-> gradle
<-> unitylibrary
<-> build
<-> libs
<-> src
[+] build.gradle
[+] proguard-unity.txt
<-> unitylibs
[+] build.gradle
[+] build.gradle
[+] gradle.properties

27 // implementation(name: 'com.google.firebase.firebase-app-unity-6.0.0', ext:'aar')
28 // implementation(name: 'com.google.firebase.firebase-auth-17.0.0', ext:'aar')
29 // implementation(name: 'com.google.firebase.firebase-auth-interop-17.0.0', ext:'aar')
30 // implementation(name: 'com.google.firebase.firebase-auth-unity-6.0.0', ext:'aar')
31 // implementation(name: 'com.google.firebase.firebase-common-17.0.0', ext:'aar')
32 // implementation(name: 'com.google.firebase.firebase-database-17.0.0', ext:'aar')
33 // implementation(name: 'com.google.firebase.firebase-database-collection-16.0.1', ext:'aar')
34 // implementation(name: 'com.google.firebase.firebase-database-unity-6.0.0', ext:'aar')
35 // implementation(name: 'com.google.firebase.firebase-iid-17.1.2', ext:'aar')
36 // implementation(name: 'com.google.firebase.firebase-iid-interop-16.0.1', ext:'aar')
37 // implementation(name: 'com.google.firebase.firebase-measurement-connector-17.0.1', ext:'aar')
38 // implementation(name: 'UnityAds', ext:'aar')
39 // implementation(name: 'UnityAdsAndroidPlugin', ext:'aar')
40
41 implementation fileTree('Firebase')
42 implementation project(path: 'unitylibs')
43
```

- 当然如果你有很多 aar 库，那么你需要在根目录创建一个 LocalRepo 目录，然后将你不同的 aar 库放在不同文件夹下。在 setting.gradle 分别导入
- 下面它是这么说的, 可是我都把它们放在同一个类库里, 看不行的话再移. 为什么每个包都需要一个单独的类库呢? 解偶多个不同包之间的依赖性? 加载时的内存性能影响等?

## 9 安卓设备上资源包的存放位置, 以及是否本地存放有需要的资源包

This PC\HEYAN's S10+\Internal storage\Android\data\com.defaultcompany.trunk\files

## Android:

```
Application.dataPath           /data/app/package name-1/base.apk
Application.streamingAssetsPath jar:file:///data/app/package name-1/base.apk!/assets
Application.tmporaryCachePath  /storage/emulated/0/Android/data/package name/cache
Application.persistentDataPath /storage/emulated/0/Android/data/package name/files
```

看Android上的路径，跟iOS有点类似，简单说一下。Android的几个目录是apk程序包、内存存储(InternalStorage)和外部存储(ExternalStorage)目录。

**apk程序包目录**: apk的安装路径，/data/app/package name-n/base.apk，dataPath就是返回此目录。

**内部存储目录**: /data/data/package name-n/，用户自己或其它app都不能访问该目录。打开会发现里面有4个目录（需要root）

- cache 缓存目录，类似于iOS的Cache目录
- databases 数据库文件目录
- files 类似于iOS的Documents目录
- shared\_prefs 类似于iOS的Preferences目录，用于存放常用设置，比如Unity3D的PlayerPrefs就存放于此

**外部存储目录**: 在内置或外插的sd上，用户或其它app都可以访问，外部存储目录又分私有和公有目录。

**公有目录**是像DCIM、Music、Movies、Download这样系统创建的公共目录，当然你也可以像微信那样直接在sd卡根目录创建一个文件夹。好处嘛，就是卸载app数据依旧存在。

**私有目录**在/storage/emulated/n/Android/data/package name/，打开可以看到里面有两个文件夹cache和files。为什么跟内部存储目录重复了？这是为了更大的存储空间，以防内存存储空间较小。推荐把不需要隐私的、较大的数据存在这里，而需要隐私的或较小的数据存在内部存储空间。

下面是各路径对应的Java访问方法：

```
apk包内: AssetManager.open(String filename)
内部存储: context.getFilesDir().getPath() or context.getCacheDir().getPath()
外部存储: context.getExternalFilesDir(null).getPath() or context.getExternalCacheDir().getPath()
```

理解了Android存储的原理，最后来说开头提到的bug，Application.tmporaryCachePath/persistentDataPath返回空字符串。这其实因为权限的原因，app没有声明访问外部存储空间的权限，但是Application.tmporaryCachePath/ ApplicationpersistentDataPath却想返回外部存储的路径。这是Unity3D的bug，没有权限本应该抛出异常或者错误，让开发者知道原因。

经反复测试发现，有【外置SD卡】的设备上，如果声明读写外部存储设备的权限，会返回外部存储路径，不声明则会返回内部存储路径，这样不会有问题。而在【无外置SD卡】的设备上，不管是否声明读/写外部存储设备的权限，Application.tmporaryCachePath/persistentDataPath都返回外部存储路径，但是又没有权限，就可能会导致返回null了，所以说可能是因为这个bug不是必现，如果出现了设备重启之后就好了，怀疑是linux设备mount问题。但是出了问题，我们不能跟用户说你重启一下手机就好了。

```
1 | <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

```
Application.dataPath /data/app/package name-1/base.apk
Application.streamingAssetsPath jar:file:///data/app/package name-1/base.apk!/assets
Application.tmporaryCachePath /storage/emulated/0/Android/data/package name/cache
Application.persistentDataPath/storage/emulated/0/Android/data/package name/files
```

- 看 Android 上的路径，跟 iOS 有点类似，简单说一下。Android 的几个目录是 apk 程序包、内存存储 (InternalStorage) 和外部存储 (ExternalStorage) 目录。
- **apk 程序包目录**: apk 的安装路径，/data/app/package name-n/base.apk，dataPath 就是返回此目录。
- **内部存储目录**: /data/data/package name-n/，用户自己或其它 app 都不能访问该目录。打开会发现里面有 4 个目录（需要 root）
- cache 缓存目录，类似于 iOS 的 Cache 目录
- databases 数据库文件目录
- files 类似于 iOS 的 Documents 目录
- shared\_prefs 类似于 iOS 的 Preferences 目录，用于存放常用设置，比如 Unity3D 的 PlayerPrefs 就存放于此

- 外部存储目录: 在内置或外插的 sd 上, 用户或其它 app 都可以访问, 外部存储目录又分私有和公有目录。
- 公有目录是像 DCIM、Music、Movies、Download 这样系统创建的公共目录, 当然你也可以像微信那样直接在 sd 卡根目录创建一个文件夹。好处嘛, 就是卸载 app 数据依旧存在。
- 私有目录在/storage/emulated/n/Android/data/package name/, 打开可以看到里面有两个文件夹 cache 和 files。为什么跟内部存储目录重复了? 这是为了更大的存储空间, 以防内存存储空间较小。推荐把不需要隐私的、较大的数据存在这里, 而需要隐私的或较小的数据存在内部存储空间。
- 下面是各路径对应的 Java 访问方法:
  - apk 包内: AssetManager.open(String filename)
  - 内部存储: context.getFilesDir().getPath() or context.getCacheDir().getPath()
  - 外部存储: context.getExternalFilesDir(null).getPath() or context.getExternalCacheDir().getPath()

理解了 Android 存储的原理, 最后来说说开头提到的 bug, Application temporaryCachePath/persistentDataPath 返回空字符串。这其实因为权限的原因, app 没有声明访问外部存储空间的权限, 但是 Application temporaryCachePath/ Application persistentDataPath 却想返回外部存储的路径。这是 Unity3D 的 bug, 没有权限本应该抛出一个异常或者错误, 让开发者知道原因。

- 经反复测试发现, 有【外置 SD 卡】的设备上, 如果声明读/写外部存储设备的权限, 会返回外部存储路径, 不声明则会返回内部存储路径, 这样不会有问题。而在【无外置 SD 卡】的设备上, 不管是否声明读/写外部存储设备的权限, Application temporaryCachePath/persistentDataPath 都返回外部存储路径, 但是又没有权限, 就可能会导致返回 null 了, 之所以说可能是因为这个 bug 不是必现, 如果出现了设备重启之后就好了, 怀疑是 linux 设备 mount 问题。但是出了问题, 我们不能跟用户说你重启一下手机就好了。

```
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

## 10 那么现在就是说: 安卓 SDK 与 unity 的交互与打包基本没有问题了

- This PC's S10+storage.defaultcompany.trunk



SCORE LEVEL LINE  
0 1 0



SAVE GAME

IN

RESUME GAME

TUTORIAL

BACK TO MAIN  
MENU

CREDIT

- 但对自己更大的挑战是: 为什么 unity 里一个空物件挂载到热更新的过程, 我打包之后在安卓手机上运行不出来, 仍需要时间 debug 这个过程 (呵呵, 前面昨天还是前天已经想到问题的原因, 不到因为探讨其它的想法, 直到今天傍晚刚才整个过程才理清. 不过目前仍是用 unity 直接到包, 还有许多其它的细节小问题需要解决)
- 过程中遇到过, 还会遇到很多不懂的问题, 比如同样的某些 android studio 里加 android:exported="true" 各种标签等, 如果只用 unity 打包, 该如何实现呢? 两套不同的打包机制都得弄明白. 但都是这么一个学习的过程, 不会被轻易挫败.
- 相比之下, 安卓 SDK 的实现极其简单, 可以放在后面

## 10.1 FATAL EXCEPTION: main

```

8943 8943 E AndroidRuntime: FATAL EXCEPTION: main
8943 8943 E AndroidRuntime: Process: com.defaultcompany.trunk, PID: 8943
8943 8943 E AndroidRuntime: java.lang.RuntimeException: Unable to destroy activity {com.defaultcompany.trunk/com.defaultcompany.trunk.UnityPlayerActivity}: java.lang.NullPointerException: Attempt to invoke virtual method 'boolean android.os.Handler.sendMessage(android.os.Message)' on a null object reference
8943 8943 E AndroidRuntime: at android.app.ActivityThread.performDestroyActivity(ActivityThread.java:5950)
8943 8943 E AndroidRuntime: at android.app.ActivityThread.handleDestroyActivity(ActivityThread.java:5995)
8943 8943 E AndroidRuntime: at android.app.servertransaction.DestroyActivityItem.execute(DestroyActivityItem.java:47)
8943 8943 E AndroidRuntime: at android.app.servertransaction.ActivityTransactionItem.execute(ActivityTransactionItem.java:45)
8943 8943 E AndroidRuntime: at android.app.servertransaction.TransactionExecutor.executeLifecycleState(TransactionExecutor.java:176)
8943 8943 E AndroidRuntime: at android.app.servertransaction.TransactionExecutor.execute(TransactionExecutor.java:97)
8943 8943 E AndroidRuntime: at android.app.ActivityThread$H.handleMessage(ActivityThread.java:2438)
8943 8943 E AndroidRuntime: at android.os.Handler.dispatchMessage(Handler.java:106)
8943 8943 E AndroidRuntime: at android.os.Looper.loopOnce(Looper.java:226)
8943 8943 E AndroidRuntime: at android.os.Looper.loop(Looper.java:313)
8943 8943 E AndroidRuntime: at android.app.ActivityThread.main(ActivityThread.java:8663)
8943 8943 E AndroidRuntime: at java.lang.reflect.Method.invoke(Native Method)
8943 8943 E AndroidRuntime: at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:567)
8943 8943 E AndroidRuntime: at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:1135)
8943 8943 E AndroidRuntime: Caused by: java.lang.NullPointerException: Attempt to invoke virtual method 'boolean android.os.Handler.sendMessage(android.os.Message)' on a null object reference
8943 8943 E AndroidRuntime: at android.os.Message.sendToTarget(Message.java:468)
8943 8943 E AndroidRuntime: at com.unity3d.player.UnityPlayer$a(Unknown Source:8)
8943 8943 E AndroidRuntime: at com.unity3d.player.UnityPlayer$a(Unknown Source:2)
8943 8943 E AndroidRuntime: at com.unity3d.player.UnityPlayer.quit(Unknown Source:28)
8943 8943 E AndroidRuntime: at com.defaultcompany.trunk.UnityPlayerActivity.onDestroy(UnityPlayerActivity.java:44)
8943 8943 E AndroidRuntime: at android.app.Activity.performDestroy(Activity.java:8571)
8943 8943 E AndroidRuntime: at android.app.Instrumentation.callActivityOnDestroy(Instrumentation.java:1364)
8943 8943 E AndroidRuntime: at android.app.ActivityThread.performDestroyActivity(ActivityThread.java:5937)
8943 8943 E AndroidRuntime: ... 15 more

```

- 这个没有再出现了, 根据这里改的:<https://forum.unity.com/threads/android-crashes-after-1126979/>
- 但是游戏的界面仍然是渲染不出来, 还在找原因

```

@Override protected void onDestroy () {
    Log.d(TAG, "onDestroy() ");
    // mUnityPlayer.destroy();
    mUnityPlayer.removeAllViews();
    mUnityPlayer.quit();
    super.onDestroy();
}

```

## 10.2 类库包里的错误的修复问题

- 现在还不是很有经验, 或是还没有经历很好地锻炼怎么改类库包里的错误, 晚点儿再理会这些

```

<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:state_enabled="false"
        android:color="@color/common_google_signin_btn_text_dark_disabled" />
    <item
        android:state_pressed="true"
        android:color="@color/common_google_signin_btn_text_dark_pressed" />
    <item
        android:state_focused="true"
        android:color="@color/common_google_signin_btn_text_dark_focused" />
    <item
        android:color="@color/common_google_signin_btn_text_dark_default" />
</selector>

```

ERROR: F:\tmp\test\unity\libs\build\transforms\ac63488beaabb80e19f1e3c851bb20a\transformed\com.google.android.gms.play-services-base-16.0.1\

- 先只把这些有错误的类库包不连上