1. Android studio:

主体采用一个主工程，里面有几个子工程,父工程和子工程都会有gradle文件，这个文件是配置文件，比如下载链接要挂在父工程的gradle里，类似于

buildscript { repositories { jcenter() maven { url 'https://dl.bintray.com/umsdk/release' } } dependencies { classpath 'com.android.tools.build:gradle:2.3.3' // NOTE: Do not place your application dependencies here; they belong // in the individual module build.gradle files }}allprojects { repositories { jcenter() mavenCentral() maven { url 'https://dl.bintray.com/umsdk/release' } }}task clean(type: Delete) { delete rootProject.buildDir}

子项目的gradle可以配置在线集成以及配置本地引用

例如

dependencies { compile fileTree(include: ['\*.jar'], dir: 'libs') //PushSDK必须依赖基础组件库，所以需要加入对应依赖 //compile 'com.umeng.umsdk:common:2.1.0' //PushSDK必须依赖utdid库，所以需要加入对应依赖 //compile 'com.umeng.umsdk:utdid:1.1.5.3' //PushSDK //compile 'com.umeng.umsdk:push:6.0.1' compile 'com.umeng.umsdk:xiaomi-push:3.6.17' compile 'com.umeng.umsdk:xiaomi-umengaccs:1.1.0' compile project(':push')}

前面几个是在线包，最后一个compile project(':push')指的是引用同项目的子项目push.

1.打包代码，在要打包的子项目中添加

task makeJar(type: Copy) { //如果什么都不写就生成在默认的路径下}makeJar.dependsOn(build)

然后在下面Terminal命令行中输入，gradle makeJar即可，中间报错百度

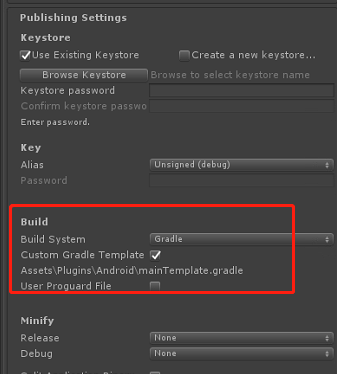
1. 打apk包时，gradle最上面需要是apply plugin: 'com.android.application'
2. 打jar包或者被引用时时，gradle最上面需要是apply plugin: 'com.android.library'
3. 打apk时，这点很不一样，这次栽坑也是因为这，他会把所有在线集成的和引用到的和自己的manifest文件进行合并，这次离线集成跑不通，就是因为在线集成下载的包里有manifest。
4. 如果想sync，则view->Toolbar,然后
5. Build中是当前项目编译进度，有错误都会显示在这里
6. 一个安卓项目，有一个Application，里面有很多Activity，需要把一个Activity设为启动
7. 打apk的gradle中需要applicationId "com.zgame.qlywz"(不确定需不需要，但是有的manifest中会${ applicationId })

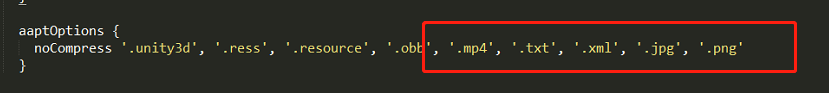
时隔很久第二次用as，依然一脸懵逼，之前写的并不详细，当时理解了，但是隔了很久看很多东西并没有看懂，这次主要目的是打个jar，写下流程：

1. 刚开始一直想新建一个纯jar的项目，但是查了下好像不行，只能先建一个空的android项目，然后要么改app，要么再new一个module
2. 这次踩坑的重点，as需要把java类写到src->main->java->包名下，而不是像eclipse直接建文件夹就行，我特么把之前的带包名的文件夹直接拷贝到src里了，怎么都不好使
3. 完事之后，还是上面的打包步骤，但是gradle不好使，用gradlew好使了，暂时没深究
4. 打包的时候报错说什么task代理类问题，把根目录的build.gradle里的classpath版本改成3.0.0就好使了，之前好像是3.1.14

Unity gradle打包报错，需要升级jdk为jdk-8u251-windows-x64

Gradle未压缩数量限制，其实是限制的种类，但是unity把整个StreamingAssets文件夹文件算进去了，修改方法，勾选这个，生成文件，然后修改aaptOptions





这些是修改的，吧StreamingAssets包括去掉了，把里面包含的后缀都列出来了

1. 引用aar包，先放到libs里，然后对应小项目的gradle里加

android {  
repositories {  
flatDir {  
dirs 'libs'  
}  
}  
}

dependencies {  
compile(name: 'lyx-library-1.0.3', ext: 'aar')  
}