**安卓SDK接入**

# 环境

Unity5.3.5

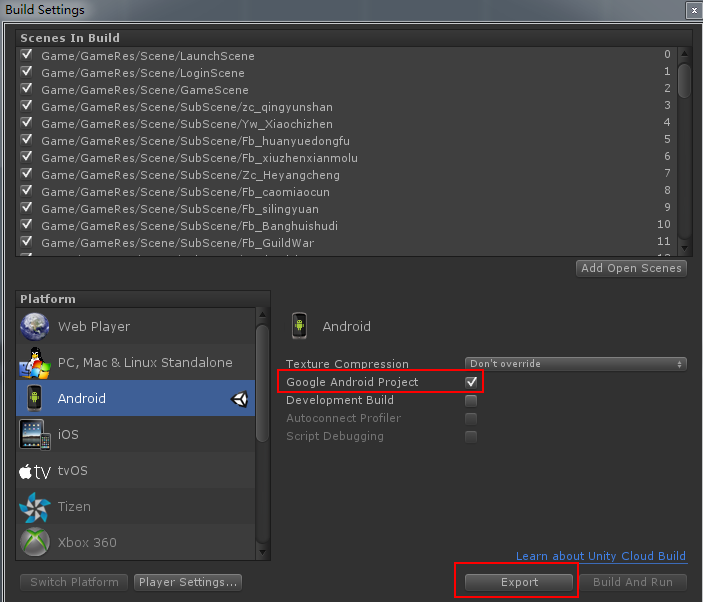
64位jdk：jdk-8u60-windows-x64.exe。默认位置C:\Program Files\Java\jdk1.8.0\_60

android-sdk：解压android\_sdk.rar到硬盘。例如D: \android-sdk

解压eclipse.rar

SDK接入就是c#和java相互调用的过程，将一些登录充值等操作放到java这边来实现，并将实现结果回调给c#客户端。下面讲讲登录流程，其他的类似登录。

# 开发



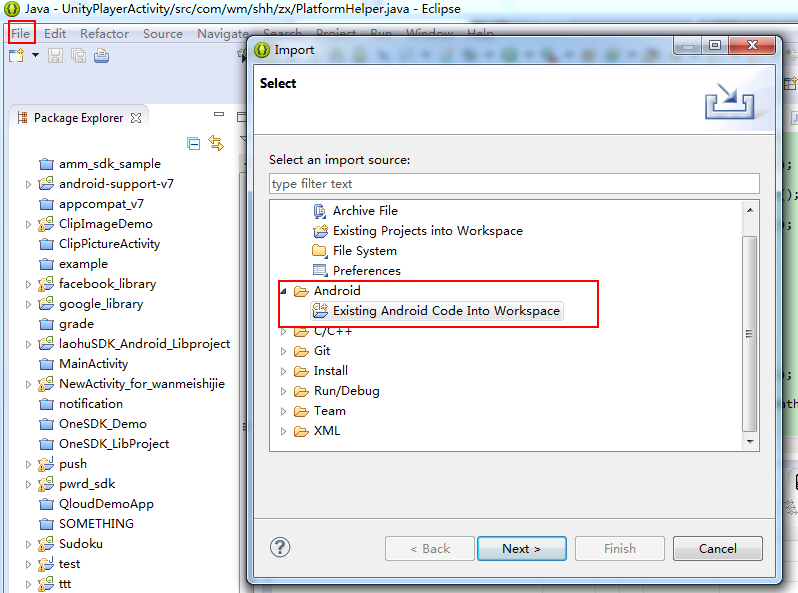
Unity-file-BuildSetting，导出一个android工程：UnityPlayerActivity。此步骤只需一次。

目前使用的安卓公测已经建立，路径是

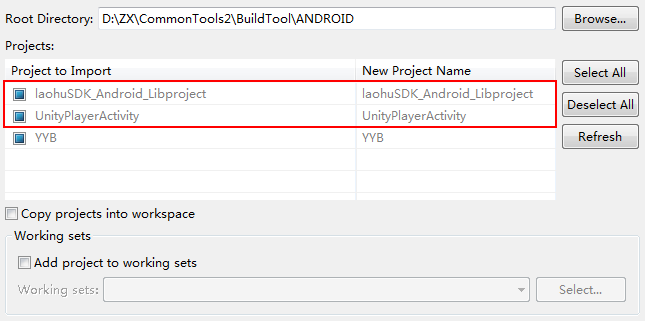
svn://10.17.5.13/Zhuxian/CommonTools/BuildTool/ANDROID/UnityPlayerActivity

打开eclipse并导入上面的工程

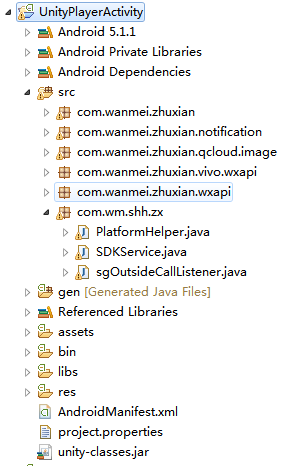
导入方法：eclipse左上角 File-Import



选中工程目录



最后点击finish即可，如下图：



Src里面是java的类，逻辑操作都在里面

Assets是放外部资源的目录

Libs放第三方库

Res放系统的资源，比如游戏图标，一些自定义字段

AndroidManifest.xml 安卓的入口文件，定义了游戏的主类和一些SDK用到的界面类

回到Unity，定义了两个文件：

AndroidHelper.cs – 封装c#调用java的函数

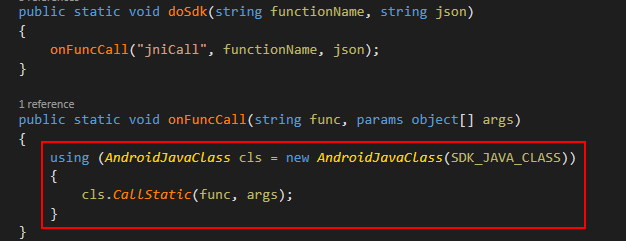
AndroidCallback.cs – 接收java调用c#的函数

将这两个文件绑定到一个空的gameobject，挂到场景里。

在AndroidHelper里定义了java的一个类：

SDK\_JAVA\_CLASS = "com.wm.shh.zx.SDKService"

C#调用java的方式为：



红框里是找到java的类，然后调用里面的某个方法。

例如登录操作，当点击登录按钮的时候，只需调用AndroidHelper.doSdk("doLogin","");

Android工程里的com.wm.shh.zx.SDKService.java 这个类就会接收到这个调用，我们在里面实现相应的doLoing方法即可：



方法里面的内容是SDK提供，具体实现方法根据对方提供的文档进行。这里实现的是ONESDK提供的登录调用方法，这样就会弹出第三方登录界面进行登录，如果登录成功，将返回的信息userid和token返回给unity，即红框里面的内容。

此时unity的脚本会收到这个调用，我们在AndroidCallback里实现onLogin函数即可：

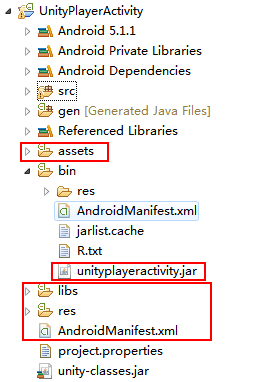


将java返回的userid和token解析出来发送到服务器进行验证。

到此，登录功能客户端部分完成。

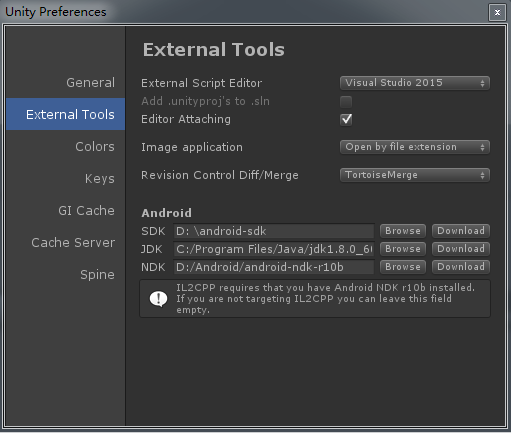
# 打包

集成好SDK之后，需要将Android工程的资源拷贝回unity进行打包

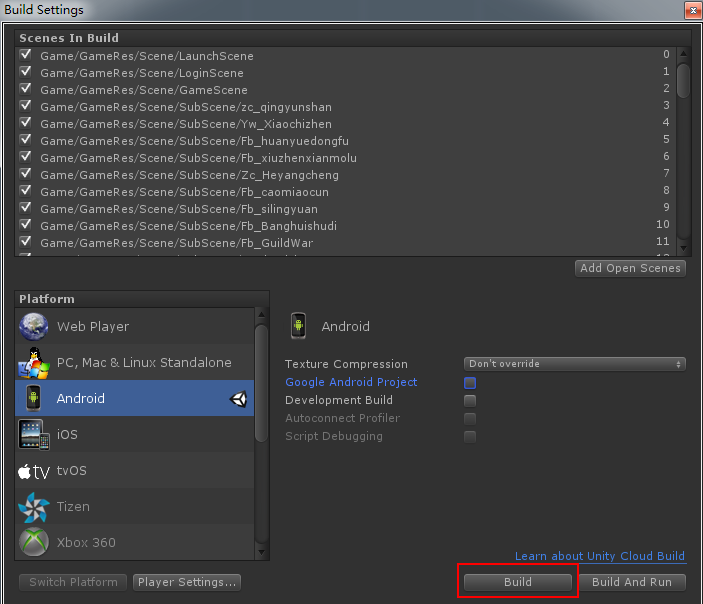


将红框的文件拷贝到Client\Assets\Plugins\Android目录下

在unity-edit-Preferences中配置以上两个路径



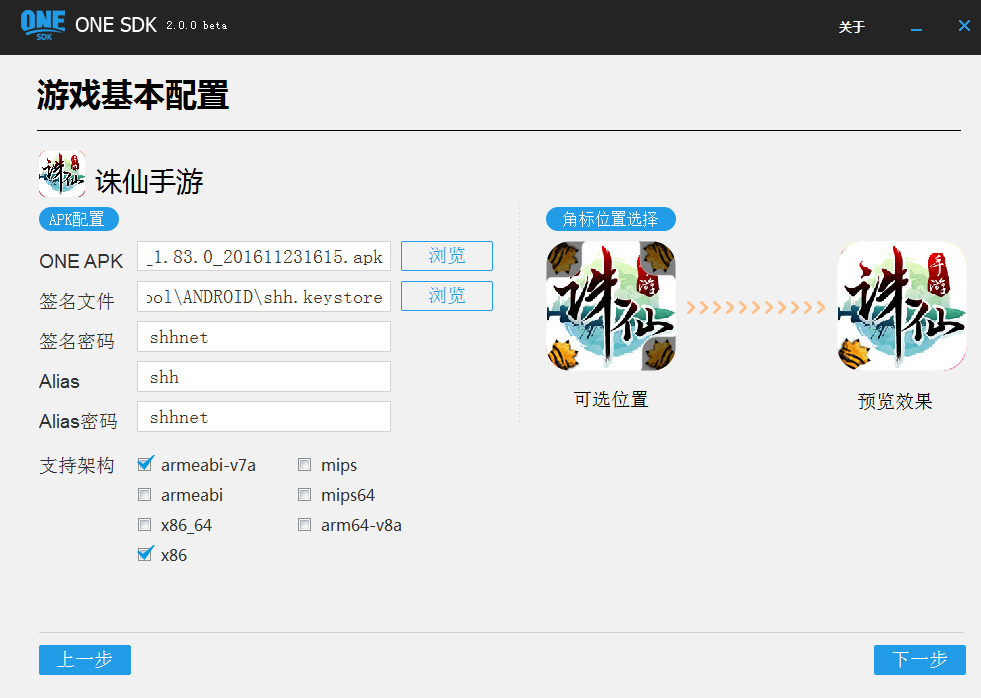
然后打开Unity-file-BuildSetting-Build，即可导出apk安装文件。



打包过程已经封装成函数，Unity菜单：Game-build-BuildAndroid 打包带SDK的版本，具体代码可参照GameBuild.cs

打出的apk路径：Client\Build\Game\_Android

至此打出的包母包，然后使用ONESDK的聚合工具，可以打出各个渠道的分包。







以上配置按相关需求配置好，最后点击开始打包。