SDK打包自动化

**一、目前所接的渠道一般有的需求和修改**

1. 支付
2. 登录
3. 接入游戏大厅或悬浮条
4. 返回键
5. 暂停（从后台切换到前台显示）
6. 修改包名
7. 修改Icon
8. ChannelID
9. 设置短信支付方式

10、设置其相应的appid和appkey

我们游戏对于各个需求的修改：

1. 支付：现在所有游戏都是统一的了
2. 登录：现在所有游戏都是统一的了

3、游戏大厅：分为SDK自带按钮的（如91）和不自带按钮的(如oppo)：

自带按钮的：可以在初始化的时候直接调用，不需要修改c++代码；

不自带按钮的：需要修改一次我们游戏的c++代码，在大厅找一个地方添加，或者替换掉原来的一个按钮。目前需要接入游戏大厅的有：91，oppo

4、返回键：由于不是所有的渠道都要求用他们的返回键，所以现在的策略是，在c++代码对特定的渠道进行判断，需要修改一次c++代码，添加上相应的渠道枚举值，目前需要修改退出的有：91渠道,360渠道

5、暂停：目前只有91有这个需求，不需要修改我们c++代码

6、包名：目前需要修改包名的渠道有：oppo，91，xiaomi，蜗牛，点金

7、icon替换：目前需要修改icon的渠道有：oppo，91，蜗牛，点金

1. channelID一般特定渠道可以在Androidmanifest中进行修改

9、设置短信支付方式：在game.properties和gamedebug.properties中设置

10、设置其相应的appid和appkey：在game.properties和gamedebug.properties中设置

**二、脚本自动化：**

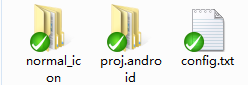
脚本实现了java层面的自动化，

1、在每个渠道下：都会有我们所有游戏的一个文件夹（如zjh，lhj，dn），

用来配置该渠道的一些配置项，

2、每个文件加下：都会有相应渠道的配置项，

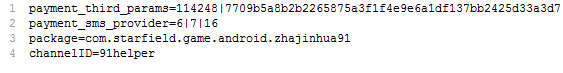
如91渠道的zjh文件夹有以下配置项



normal\_icon里面放的是正常的icon，在我们还原的时候需要用到这些icon，

proj.android里面放的是91需要我们替换的icon

config.txt里面是一些渠道具体的配置项：



分别对应：

渠道的appid|appkey

渠道支持的短信支付方式

包名

渠道名

不同的渠道会有相应的配置，每个游戏应该在其相应的文件夹（已经存在并且包含了需要配置的配置项）下，配置好其相应的配置

对于目前的第三方配置如下：

1、不需要任何配置的：移动，电信，联通，dpay，凌云支付，willpay

2、只需要配置params的：huawei，lenovo，xiaomi，360

3、需要配置params、图标和包名的：oppo，91，woniu

**每个游戏都应该，对照炸金花进行相应的配置**

脚本已经能完成自动化，命令为：

**./setup\_config ../../../game\_client/proj.android add 91pay/ zjh**

**zjh：已经存在的配置了的文件夹名字，也可以扩展为dn,lhj**

使用脚本的原则，先减后增，

需要改包名的，一开始的时候，包名必须和文件结构一致，不然，无法复制到路径中的文件，会导致修改包名不成功

**三、C++部分需要修改一次的：**

现在脚本实现了java部分的自动化，对于至少需要修改一次的需求，还是需要每个游戏进行在c++部分的一次修改，需要一次修改的有以下几点：

1、开始界面（lhj应为load界面）、大厅的退出，应该把keyback函数中添加退出相应渠道退出方式的调用：

void MainLayer::keyBackClicked()函数中修改

CCLog("MainLayer Keyback clicked!");

if(!PopupLayer::handleKeyBack() && !m\_broadcastNode->handleKeyBack())

{

Int iAccountProvider = RequestPaymentLoginProvider(); //取得渠道枚举值

if (QIHUBilling == iAccountProvider || NdStore == iAccountProvider) //判断是否为要求退出的渠道

{

PaymentServiceLogoutParams params;

params.provider = (PaymentServicePlatform)iAccountProvider;

params.pContext = this;

RequestPaymentServiceLogout(params.provider, &params, NULL);//调用退出，null表无回调

}

else

{

我们自己游戏的退出方式

}

}

开始界面也做相同的处理