NBMediation 对接说明

```
implementation rootProject.ext.dependencies.multidex
  implementation 'com.android.support:appcompat-v7:28.0.0
   implementation.'com.android.support.constraint:constraint-layout:1.1.3'
  implementation 'com.android.support:support-v4:28.0.0'
  testImplementation 'junit:junit:4.12'
   androidTestImplementation 'com.android.support.test:runner:1.0.2'
androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
   implementation(name: 'nm-android-sdk', ext: 'aar')
  implementation(name: 'suib_base_4.2.7_sucn', ext: 'jar')
  implementation project(path: ':adapter_cn:cloudmobi')
  implementation(name: 'hyadxopensdk', ext: 'aar')
  implementation.project(path: ':adapter_cn:hyadxopen')
  //tencentad.cn
  implementation(name: 'GDTSDK.unionNormal.4.191.1061', ext: 'aar')
   implementation(name: 'android-query-full.0.26.7', ext: 'jar')
   implementation.project(path: ':adapter_cn:tencentad')
  implementation(name: 'open_ad_sdk_2.9.5.5', ext: 'aar')
  implementation rootProject.ext.dependencies.adcolony
   implementation.project(path: ':adapter:adcolony')
   implementation.rootProject.ext.dependencies.admob
```

所有的离线 aar 和 jar 包都放在 demo 项目根目录下的 libs 目录

一、引入 SDK 步骤

1、引入 NBMediation SDK

```
//NBMediation
implementation(name: 'nm-android-sdk', ext: 'aar')

2、引入聚合的 SDK 和相对应的 adapter
//cloudmobi cn
implementation(name: 'suib_base_4.2.7_sucn', ext: 'jar')
implementation(name: 'suib_imageloader_4.2.7_sucn', ext: 'jar')
implementation(name: 'suib_video_4.2.7_sucn', ext: 'jar')
implementation project(path: ':adapter_cn:cloudmobi')

//hyadxopen cn
implementation(name: 'hyadxopensdk', ext: 'aar')
implementation project(path: ':adapter_cn:hyadxopen')

//tencentad cn
implementation(name: 'GDTSDK.unionNormal.4.191.1061', ext: 'aar')
```

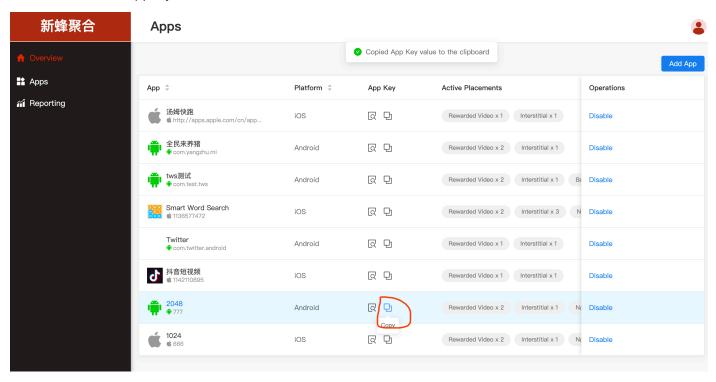
```
implementation(name: 'android-query-full.0.26.7', ext: 'jar')
implementation project(path: ':adapter_cn:tencentad')

//tiktok cn
implementation(name: 'open_ad_sdk_2.9.5.5', ext: 'aar')
implementation project(path: ':adapter_cn:tiktok')
```

二、聚合 SDK 初始化

请在 Application 的 onCreate 里面初始化聚合 SDK,或者在 SplashActivity 里初始化,接入开屏请参考第六章节

这里需要传入一个 AppKey, 请在聚合后台获取, 具体请看下图, 点击复制即可:



三、激励视频和插屏接入

这里通过下图接口来判断,返回 true 表示激励视频或者插屏已准备就绪,可以调用 showAd 方法展示广告

```
if (RewardedVideoAd.isReady()) {
    setRewardVideoButtonStat(true);
}
if (InterstitialAd.isReady()) {
    setInterstitialButtonStat(true);
}
```

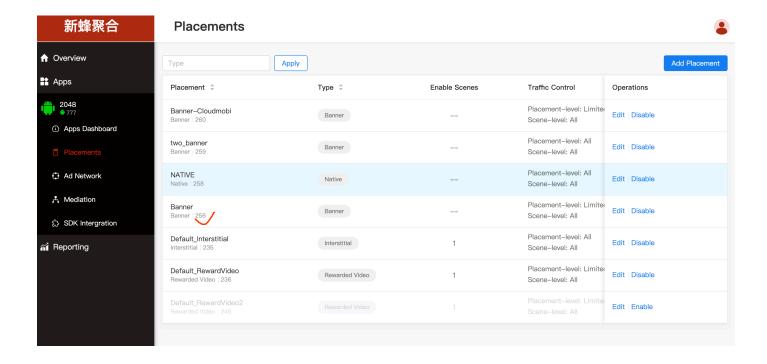
设置全局激励视频的监听回调:

设置全局插屏的监听回调:

具体请参考 DEMO 里的 com.nbmediation.sdk.demo.MainActivity 详情

四、Banner 接入

```
bannerAd = new BannerAd( activity: this, NewApiUtils.P_BANNER, new BannerAdListener() {
    public.void.onAdReady(View.view).{
            if.(null:!=.view.getParent()).{
                ((ViewGroup) view.getParent()).removeView(view);
           adContainer.removeAllViews();
           RelativeLayout.LayoutParams.layoutParams.= new RelativeLayout.LayoutParams(
            layoutParams.addRule(RelativeLayout.CENTER_IN_PARENT);
            adContainer.addView(view, layoutParams);
           Log.e( tag: "AdtDebug", e.getLocalizedMessage());
        bannerButton.setEnabled(true);
        bannerButton.setText("Load And Show Banner Ad");
   public.void.onAdFailed(String.error).{
        bannerButton.setEnabled(true);
        bannerButton.setText("Banner Load Failed, Try Again");
   @Override
   public void onAdClicked().{
bannerAd.setAdSize(AdSize.AD_SIZE_320X50);
bannerAd.loadAd();
```



五、native 接入

```
nativeAd = new NativeAd( activity: this, NewApiUtils.P_NATIVE, new NativeAdListener() {
       nativeButton.setEnabled(true);
       nativeButton.setText("Native Load Failed, Try Again");
   @Override
   public.void.onAdReady(AdInfo.info).{
       adView = LayoutInflater.from(MainActivity.this).inflate(R.layout.native_ad_layout, null);
       TextView.title = adView.findViewById(R.id.ad_title);
       title.setText(info.getTitle());
       TextView desc = adView.findViewById(R.id.ad_desc);
       desc.setText(info.getDesc());
       Button.btn.=.adView.findViewById(R.id.ad_btn);
       btn.setText(info.getCallToActionText());
       MediaView mediaView = adView.findViewById(R.id.ad_media);
       nativeAdView = new NativeAdView( context: MainActivity.this);
       AdIconView adIconView = adView.findViewById(R.id.ad_icon_media);
       DisplayMetrics displayMetrics = MainActivity.this.getResources().getDisplayMetrics();
       mediaView.getLayoutParams().height = (int) (displayMetrics.widthPixels / (1200.0 / 627.0));
       nativeAdView.addView(adView);
```

六、开屏接入

为保证广告效果,从原生上接入,将开屏 SplashAdActivity 作为 Launch Activity。

- 1、老版本用户,移除初始化 init 和回调,移除开屏 showSplash。
- 2、在 AndroidManifest.xml 中,添加 SplashAdActivity 并设置为启动 Activity。

3、去掉 MainActivity 的 intend-filter,不在作为 Launch Activity。

```
<activity
    android:name="org.cocos2dx.javascript.AppActivity"
    android:screenOrientation="sensorLandscape"
    android:configChanges="orientation|keyboardHidden|screenSize|screenLayout
    android: label="@string/app_name"
    android:launchMode="singleTask"
    android:taskAffinity=""
    android:theme="@android:style/Theme.NoTitleBar.Fullscreen" >
    <intent-filter android:order="1">
        <action android:name="android.intent.action.VIEW" />
        <category android:name="android.intent.category.BROWSABLE" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
    intent filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
   </intent-filter>
</activity>
```

4、引入 SplashAdActivity.java,负责初始化聚合,加载插屏,并返回 MainActivity(检查 SplashAdActivity.java 中跳转 MainActivity 的类是否和 app 匹配)。

- 5、修改 SplashAdActivit.java 中的 APPKEY。
- 6、在 init 的 onSuccess 回调函数中:
 setLoadTimeout (),设置开屏加载超时时间单位毫秒;
 setSplashAdListener(this),设置监听事件;
 setSize,设置请求宽高;