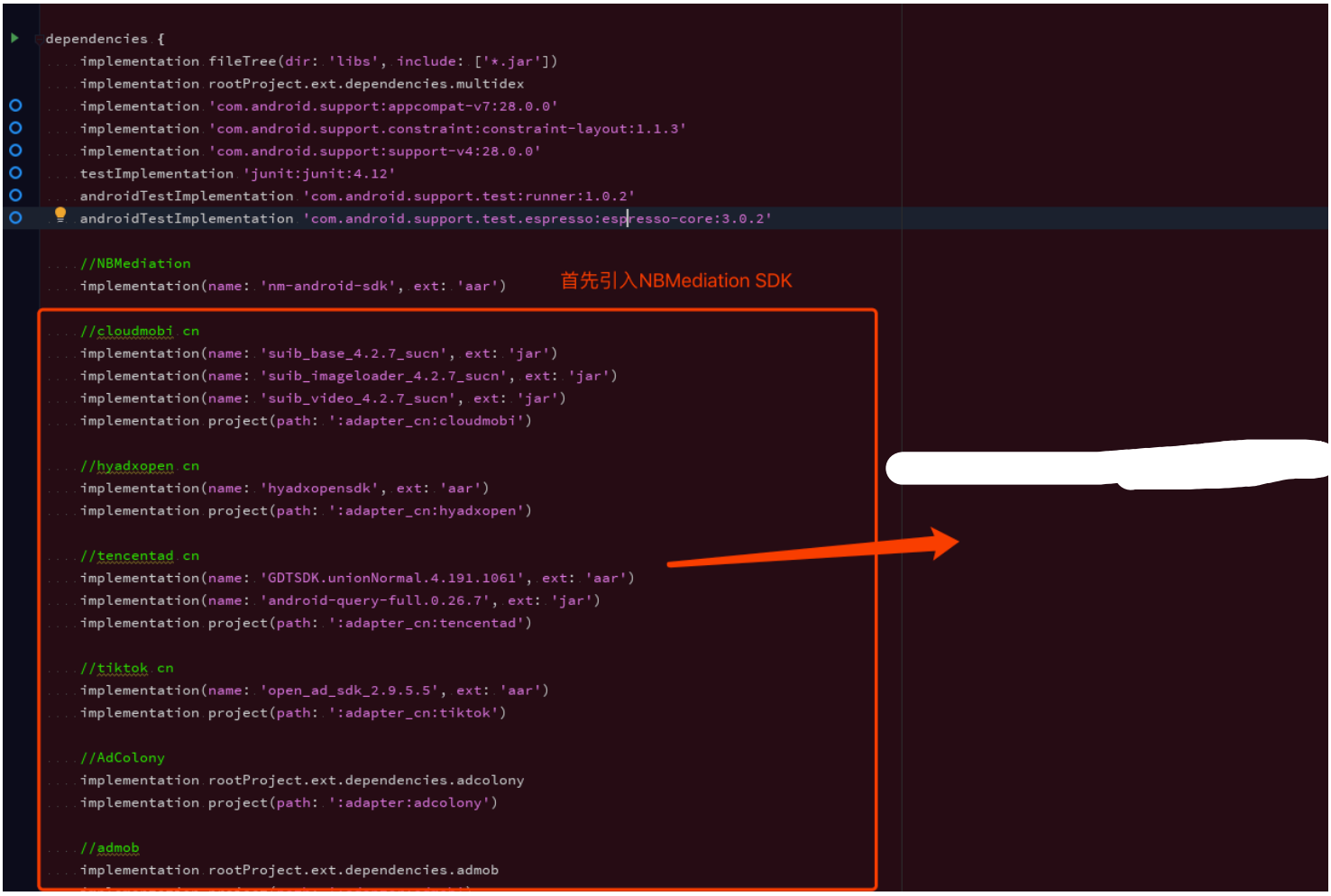
NBMediation 对接说明



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')
```

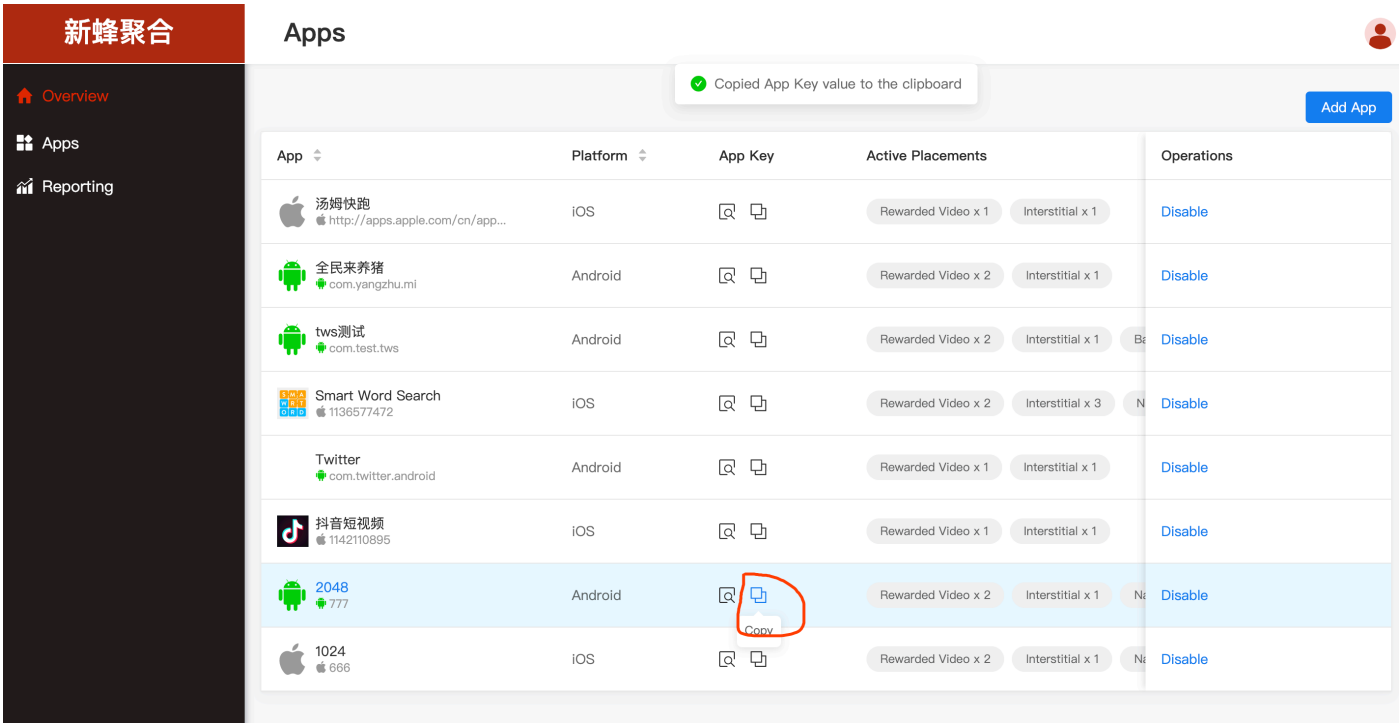
```
implementation(name: 'android-query-full.0.26.7', ext: 'jar')
implementation project(path: ':adapter_cn:tencentad')

//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

```
private void initSDK() {
    NewApiUtils.printLog( msg: "start init sdk");
    NmAds.init( activity: this, NewApiUtils.APPKEY, new InitCallback() {
        @Override
        public void onSuccess() {
            NewApiUtils.printLog( msg: "init success");
            setVideoListener();
            setInterstitialListener();
        }

        @Override
        public void onError(Error result) {
            NewApiUtils.printLog( msg: "init failed " + result.toString());

        }
    });
}
```

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);
}
```

```
public void showRewardVideo(View view) {
    RewardedVideoAd.showAd();
    setRewardVideoButtonStat(false);
}

public void showInterstitial(View view) {
    InterstitialAd.showAd();
    setInterstitialButtonStat(false);
}
```

设置全局激励视频的监听回调：

```
private void setVideoListener() {
    RewardedVideoAd.setAdListener(new RewardedVideoListener() {
        @Override
        public void onRewardedVideoAvailabilityChanged(boolean available) {
            if (available) {
                setRewardVideoButtonStat(true);
            }
        }

        @Override
        public void onRewardedVideoAdShowed(Scene scene) { NewApiUtils.printLog( msg: "onRewardedVideoAdShowed " + scene); }

        @Override
        public void onRewardedVideoAdShowFailed(Scene scene, Error error) { NewApiUtils.printLog( msg: "onRewardedVideoAdShowFailed " + scene); }

        @Override
        public void onRewardedVideoAdClicked(Scene scene) { NewApiUtils.printLog( msg: "onRewardedVideoAdClicked " + scene); }

        @Override
        public void onRewardedVideoAdClosed(Scene scene) { NewApiUtils.printLog( msg: "onRewardedVideoAdClosed " + scene); }

        @Override
        public void onRewardedVideoAdStarted(Scene scene) { NewApiUtils.printLog( msg: "onRewardedVideoAdStarted " + scene); }

        @Override
        public void onRewardedVideoAdEnded(Scene scene) { NewApiUtils.printLog( msg: "onRewardedVideoAdEnded " + scene); }

        @Override
        public void onRewardedVideoAdRewarded(Scene scene) { NewApiUtils.printLog( msg: "onRewardedVideoAdRewarded " + scene); }
    });
}
```

设置全局插屏的监听回调：

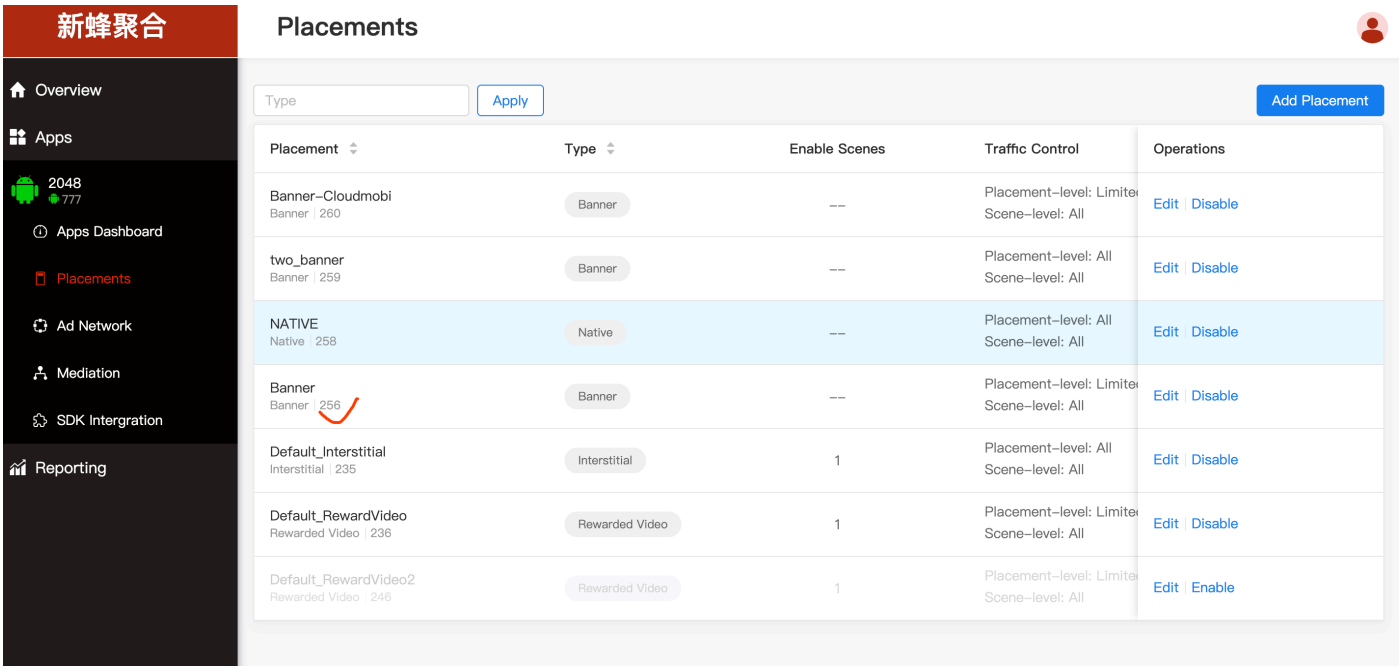
```
private void setInterstitialListener() {
    InterstitialAd.setAdListener(new InterstitialAdListener() {
        @Override
        public void onInterstitialAdAvailabilityChanged(boolean available) {
            if (available) {
                setInterstitialButtonStat(true);
            }
        }

        @Override
        public void onInterstitialAdShowed(Scene scene) { NewApiUtils.printLog( msg: "onInterstitialAdShowed " + scene); }

        @Override
        public void onInterstitialAdShowFailed(Scene scene, Error error) { NewApiUtils.printLog( msg: "onInterstitialAdShowFailed " + scene); }

        @Override
        public void onInterstitialAdClosed(Scene scene) { NewApiUtils.printLog( msg: "onInterstitialAdClosed " + scene); }

        @Override
        public void onInterstitialAdClicked(Scene scene) { NewApiUtils.printLog( msg: "onInterstitialAdClicked " + scene); }
    });
}
```

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() {
    @Override
    public void onAdReady(View view) {
        try {
            if (null != view.getParent()) {
                ((ViewGroup) view.getParent()).removeView(view);
            }
            adContainer.removeAllViews();
            RelativeLayout.LayoutParams layoutParams = new RelativeLayout.LayoutParams(
                    RelativeLayout.LayoutParams.WRAP_CONTENT, RelativeLayout.LayoutParams.WRAP_CONTENT);
            layoutParams.addRule(RelativeLayout.CENTER_IN_PARENT);
            adContainer.addView(view, layoutParams);
        } catch (Exception e) {
            Log.e( tag: "AdtDebug", e.getLocalizedMessage());
        }
        bannerButton.setEnabled(true);
        bannerButton.setText("Load And Show Banner Ad");
    }

    @Override
    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();
```

这里的广告位 ID 请去聚合后台查找



五、native 接入

```
nativeAd = new NativeAd( activity: this, NewApiUtils.P_NATIVE, new NativeAdListener() {
    @Override
    public void onAdFailed(String msg) {
        nativeButton.setEnabled(true);
        nativeButton.setText("Native Load Failed, Try Again");
    }

    @Override
    public void onAdReady(AdInfo info) {
        adContainer.removeAllViews();
        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);
```

```
            nativeAdView.setTitleView(title);
            nativeAdView.setDescView(desc);
            nativeAdView.setAdIconView(adIconView);
            nativeAdView.setCallToActionView(btn);
            nativeAdView.setMediaView(mediaView);


            nativeAd.registerNativeAdView(nativeAdView);
            RelativeLayout.LayoutParams layoutParams = new RelativeLayout.LayoutParams(
                    RelativeLayout.LayoutParams.WRAP_CONTENT, RelativeLayout.LayoutParams.WRAP_CONTENT);
            adContainer.addView(nativeAdView, layoutParams);
            nativeButton.setEnabled(true);
            nativeButton.setText("Load And Show Native Ad");
        }

        @Override
        public void onAdClicked() {

        }
    });
    nativeAd.loadAd();
```

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。

```
<activity android:name=".SplashAdActivity"
    android:theme="@android:style/Theme.DeviceDefault.NoActionBar"
    android:configChanges="keyboardHidden|screenSize|orientation"
    android:screenOrientation="portrait"
    tools:ignore="LockedOrientationActivity">
  <intent-filter>
    <action android:name="android.intent.action.MAIN" />
    <category android:name="android.intent.category.LAUNCHER" />
  </intent-filter>
</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>
```

删除

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）。

```
@Override
protected void onResume() {
    super.onResume();
    if (isClick) {
        startActivity(new Intent( packageContext: this, MainActivity.class));
        finish();
    }
}
```

```
public void toMainPage() {
    Log.e( tag: "SplashAdActivity",  msg: "---------- toMainPage ----------");
    startActivity(new Intent( packageContext: this, MainActivity.class));
    finish();
}
```

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；

7、开发者可在 SplashAdActivity 中加入 Launch Image 作为背景，默认为黑色背景。*f*