



# AdView Android SDK 开 发 者 手 册

# 目录

<b>I.</b>	<b>注册并配置 SDK-KEY .....</b>	<b>3</b>
<b>II.</b>	<b>添加 SDK .....</b>	<b>5</b>
<b>III.</b>	<b>AndroidManifest.xml 文件配置.....</b>	<b>5</b>
	3.1 增加权限代码.....	5
	3.2 增加 Activity 声明.....	6
<b>IV.</b>	<b>获取广告配置.....</b>	<b>6</b>
<b>V.</b>	<b>创建横幅广告(AdBannerActivity).....</b>	<b>7</b>
	5.1 增加代码来增添广告 .....	7
	5.2 AdViewInterface 侦听接口 .....	8
<b>VI.</b>	<b>创建插屏广告.....</b>	<b>8</b>
	6.1 创建插屏.....	8
	6.2 插屏接口调用.....	9
	6.3 创建自定义样式插屏.....	9
<b>VII.</b>	<b>创建开屏广告.....</b>	<b>10</b>
	7.1 创建开屏广告 .....	10
	7.2 开屏接口调用.....	10
	7.3 自定义开屏顶部倒计时通知样式.....	11
<b>VIII.</b>	<b>创建原生广告.....</b>	<b>11</b>
	8.1 创建原生广告 .....	11
	8.2 原生接口调用.....	12
<b>IX.</b>	<b>创建视频广告.....</b>	<b>12</b>
	9.1 创建视频广告 .....	12
	9.2 视频接口调用.....	13
<b>X.</b>	<b>添加自定义广告平台.....</b>	<b>13</b>
	10.1 参考代码.....	14
	10.2 自定义函数实现.....	14
<b>XI.</b>	<b>指定应用渠道.....</b>	<b>16</b>
<b>XII.</b>	<b>常见问题处理.....</b>	<b>17</b>
	12.1 如果应用想混淆（ProGuard）怎么办? .....	17
	12.2 如果有问题，和谁联系? .....	22
	12.3 如何使用 sample .....	22
	12.3.1 加载 sample 工程，方式一 .....	22
	12.3.2 加载 sample 工程，方式二 .....	24

## I. 注册并配置 SDK-KEY

1. 访问 AdView 网站 <http://www.adview.cn> 注册 Adview 帐号。
2. 登录后在“我的产品”页面选择“发布应用”。
3. 应用平台选择“Android”,根据提示完成相关信息,您将获取唯一的 SDK key。
4. 在“应用管理”页面,配置应用:“未设置”中输入该平台的广告 id, 打开“开关”,设置各平台投放量总和为 100%, 点击“保存”。如需更多平台, 点击右上角的“添加广告平台”。一般建议开启平台数为 1-3 个。

yyz\_3 SDK-KEY: SDK20151516030130pj9h5ywnfyqacuu

横幅广告
全屏/插屏广告
开屏广告
跳转到

地域优化 ON
广告补余 ON

国内用户配置 + 添加广告平台

广告平台	评分	设置	开关	投放量	优先级	删除平台
竞价广告	7.69	<a href="#">设置</a>	<input checked="" type="checkbox"/>	<input type="text" value="100 %"/>		
广点通	6.91	<a href="#">未设置</a>	<input type="checkbox"/>	<input type="text" value="-- %"/>	<a href="#">+</a> <a href="#">-</a>	<a href="#">删除</a>
直投广告	4.55	<a href="#">设置</a>	<input type="checkbox"/>	<input type="text" value="-- %"/>	<a href="#">+</a> <a href="#">-</a>	
交换广告	2.00	<a href="#">设置</a>	<input type="checkbox"/>	<input type="text" value="-- %"/>	<a href="#">+</a> <a href="#">-</a>	

投放总量: **100 %**

国外用户配置 + 添加广告平台

广告平台	评分	设置	开关	投放量	优先级	删除平台
竞价广告	7.69	<a href="#">设置</a>	<input type="checkbox"/>	<input type="text" value="50 %"/>		

投放总量: **0 %**

保存

5. 首页-> Android SDK 下载, 或者应用管理-> Android SDK 下载, 能获取 AdView SDK 包, 包括文档和 Demo;

Android SDK(横幅&全/插屏广告)
版本号: 3.2.4
更新时间: 2016-04-25
下载

[查看更新日志](#)
[查看在线文档](#)

6. 开启竞价或补余时, 如果需要广告点击时弹出二次确认框, 可通过后台进行如下设置: 1) 点击“编辑”; 2) 打开“确认提示”开关。不需要此过程的, 可以略过。

设备	应用名称	展示数(今/昨)▲	点击数(今/昨)▲	竞价补余状态 ?	创建日期	操作
Android	test	0/0	0/0	未审核	2016-05-25	<span style="border: 1px solid red; padding: 2px 5px;">编辑</span>   <a href="#">配置</a>   <a href="#">删除</a>

刷新时间：

地域优化： ☐ OFF

确认提示： ☒ ON



### 说明：

1. 我们提供的SDK可以让您自由选择您喜欢的广告公司，除了竞价补余以外，其他广告平台均为第三方，所需的广告App Key，需要您到这些平台官网上注册申请，并将此配置到AdView后台对应的平台下
2. 如果你是一个新手，对广告平台不怎么了解，也不知道选择哪家广告平台，或者不知道哪家广告平台收入稳定，**Adview**建议你**可以先使用竞价看看效果**。
3. 竞价和补余广告需要在后台补充市场信息，通过审核后方能获取正式广告，开始费用，在此之前都是测试广告，不产生计费
4. 开关：只有开关打开的那家平台的广告才会在应用中显示。
5. 投放量：只有开关打开的投放量才有意义，投放量各广告平台的请求优先级，一般是比例高的，优先请求。切换广告平台应同时调整投放量，所有广告状态为on的应用投放量总额应为100，否则无法正确保存
6. 横幅广告、全屏/插屏广告、开屏广告等，每页下面均有一个“保存”键，每修改完一种广告形式，都要点击该页的“保存”键，否则所做的修改无效
7. 地域优化：开启地域优化的功能是指定在国内的手机显示国内配置的广告，在国外的手机显示国外配置的广告；最大限度的满足不同的需求。当地域优化关闭的时候，就不区分国内国外了，设置为开关打开的都可以显示。
8. 如果使用视频广告，如图创建视频广告位：（原生广告同理）

**注意：**①完善视频广告位后，一定要点击“保存”

②创建成功的应用会生成“视频广告位 ID”（使用聚合功能中的视频竞价广告不需要填写广告位 ID）。视频广告位 ID 位置：应用管理→编辑→视频广告位模块

视频广告位 如果您要使用“视频广告”，请创建视频广告位

### 创建视频广告位

是否创建： ☒ 是 ☐ 否

视频尺寸：

系统会根据您选定尺寸的接近宽高比进行适配，而不是严格按照尺寸

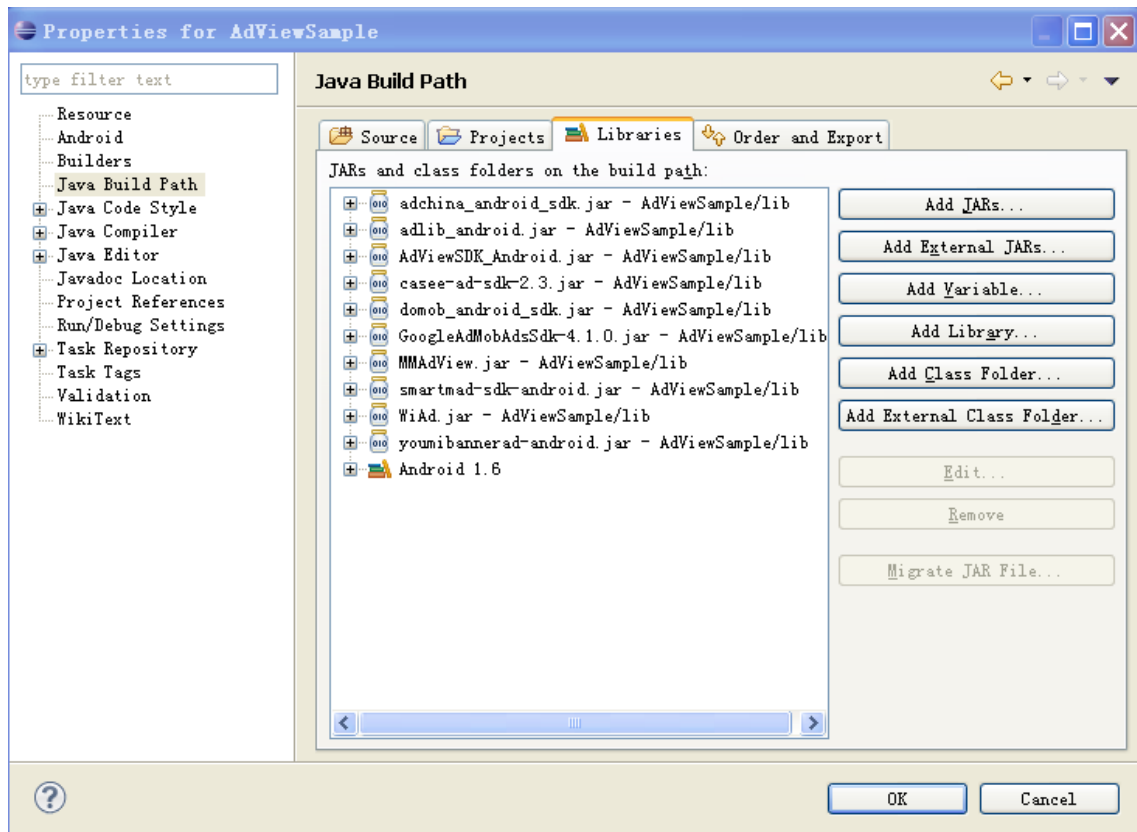
广告最长时长： ☐ 15s ☐ 30s ☐ 45s ☐ 60s ☒ 不限

视频格式： ☒ MP4 ☒ FLV ☒ SWF

视频类型： ☒ 非贴片视频 ☐ 贴片视频

## II. 添加 SDK

1. 将从网站获取 AdView SDK 开发包解压，libs 文件夹里包含所有广告平台的 SDK。(README.pdf 中有每个 jar 对应的广告平台的说明)
2. 将 AdViewSDK\_Android.jar, android-support-v4.jar 放入到您的应用工程中。
3. 将应用程序可能用到的其他广告平台的 SDK 用同样的方式添加进去。(只能使用 **adview** 提供的 **jar**，使用其他途径取得的会导致广告不能展示)



注：如果集成友盟(Umeng)SDK，需要把umeng\_res目录下的资源文件放到工程的res；如果集成微云(wiyun)SDK，需要把wiyun\_res目录下的资源文件放到工程的res,并且添加相应权限

## III. AndroidManifest.xml 文件配置

### 3.1 增加权限代码

需要添加的必备权限（完整代码可参考Demo中AndroidManifest）

```
<!-- AdView SDK mandatory or important permissions -->
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
```

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
```

备注:

INTERNET: 允许访问网络, 这是必须的

ACCESS\_NETWORK\_STATE: 允许访问网络的状态, 这是必须的

READ\_PHONE\_STATE: 允许访问手机的各种状态, 这是必须的

ACCESS\_COARSE\_LOCATION: 允许一个程序访问CellID或WiFi热点来获取粗略的位置

ACCESS\_FINE\_LOCATION: 允许一个程序访问精确位置(如GPS)

ACCESS\_WIFI\_STATE: 允许程序访问Wi-Fi网络状态信息

WRITE\_EXTERNAL\_STORAGE: 允许程序访问外部存储设备, 可以缓存广告

READ\_EXTERNAL\_STORAGE: 允许程序访问外部存储设备

### 3.2 增加 Activity 声明

某些广告平台需要声明 activity 才能正常工作, 在 application 标签中包含; 详情请参考 Demo 中 AndroidManifest;

其中竞价广告须添加配置:

```
<service android:name="com.kyview.DownloadService" />
<activity android:name="com.kyview.AdviewWebView" />
<activity android:name="com.kyview.AdActivity" />
<!-- Adivew 竞价视频 -->
<activity android:name="com.kuaiyou.video.vast.activity.VASTAdActivity"
    android:hardwareAccelerated="true"
    android:screenOrientation="landscape"/>
<activity android:name="com.kuaiyou.video.AdviewWebView"/>
```

注: 如果不聚合触控、有米、指点等广告请务必将该广播接收器删除!

## IV. 获取广告配置

```
// 请求广告之前请务必先进行初始化, 否则将无法使用广告
// 设置广告请求配置参数, 可使用默认配置InitConfiguration.createDefault(this);
InitConfiguration initConfig = new InitConfiguration.Builder(this)
    .setUpdateMode(UpdateMode.EVERYTIME) // 实时获取配置, 非必须写
    .setBannerCloseble(BannerSwitcher.CANCLOSED) // 横幅可关闭按钮
    .setInstlCloseble(InstlSwitcher.CANCLOSED) // 插屏可关闭按钮
    .setRunMode(RunMode.TEST) // 测试模式下log更多, 应用上线后请删除
    .setHtml5Switcher(Html5Switcher.NONSUPPORT) // 为默认情况, 设置后html5和非
html5广告都接收; Html5Switcher.SUPPORT, 只接收html5广告
    .setInstlDisplayMode (InstlDisplayMode.DIALOG_MODE) // 为默认情况, 设置插
屏展示模式, popupwindow模式可设置窗体外可点击
    .build();
```

```
// 分别请求横幅、插屏、原生、开屏的广告配置，keySet可为一个或多个key
AdViewBannerManager.getInstance(this).init(initConfig,MainActivity.keySet);
AdViewInstlManager.getInstance(this).init(initConfig,MainActivity.keySet);
AdViewNativeManager.getInstance(this).init(initConfig,MainActivity.keySet);
AdViewSpreadManager.getInstance(this).init(initConfig,MainActivity.keySet);
```

注：

- 1.InitConfiguration 为全局作用，只需要调用一次即可。
- 2.以上的 set 方法均为可选项，非必须添加。
- 3.从 3.2.4 版本开始，SDK 支持一个 app 中设置多个广告位(SDK-KEY)，Demo 中 keySet 中以三个广告位举例，因为个别 app 希望在多个 Activity 中分别设置不同的广告位，根据广告的展示量来统计每个 Activity 的用户访问量等。如果一个广告位能满足 app 的要求，就不需要再申请多个广告位了。

## V. 创建横幅广告(AdBannerActivity)

### 5.1 增加代码来增添广告

```
// 初始化之后请求横幅广告
AdViewBannerManager.getInstance(this).requestAd(this, key, this);
// 获取当前请求的横幅View，加载到自己的布局中
View view = AdViewBannerManager.getInstance(this).getAdViewLayout(this, key);
layout.addView(view);
```

增加部分平台设置尺寸接口：

平台	常量值	实例
InMobi	INMOBI_AD_UNIT_728x90 INMOBI_AD_UNIT_468x60 INMOBI_AD_UNIT_320x50	InitConfiguration.setAdInMobiSize(AdInMobiSize.INMOBI_AD_UNIT_320x50);
AdMob	BANNER LARGE_BANNER MEDIUM_RECTANGLE FULL_BANNER LEADERBOARD SMART_BANNER	InitConfiguration.setAdMobSize(AdMobSize.BANNER);
O2omobi 百灵欧拓	AD_SMART_BANNER AD_SIZE_728x90 AD_SIZE_468x60 AD_SIZE_320x50	InitConfiguration.setAdO2omobiSize(AdO2omobiSize.AD_SMART_BANNER);
Youmi 有米	SIZE_320x50 SIZE_468x60 SIZE_728x90 FIT_SCREEN	InitConfiguration.setAdYoumiSize(AdYoumiSize.FIT_SCREEN);
SamrtAd	AUTO_AD_MEASURE	InitConfiguration.setAdSmartAdSize(Ad

亿动智道	TABLET_AD_MEASURE_728X90 TABLET_AD_MEASURE_468X60	SmartAdSize. <a href="#">AUTO_AD_MEASURE</a> );
Gdt 广点通	BANNER BANNER_SMALL SMART_BANNER	InitConfiguration.setAdGdtSize(AdGdtSize. <a href="#">BANNER</a> );
Adview 竞价、补余	BANNER_AUTO_FILL BANNER_SMART BANNER_320X50 BANNER_480X75 BANNER_728X90	InitConfiguration.setAdSize(AdSize. <a href="#">BANNER_SMART</a> );

## 5.2 AdViewInterface 侦听接口

```
public interface AdViewBannerListener{
    /**
     * 当广告被点击时调用该函数
     */
    public void onAdClick(String key);
    /**
     * 当广告被显示时调用该函数
     */
    public void onAdDisplay(String key);
    /**
     * 当广告被关闭时调用该函数
     */
    public void onAdClose(String key);
    /**
     * 当广告被异常、失败，中断时才会调用
     */
    public void onAdFailed(String key);
    /**
     * 实验性功能
     */
    public void onAdReady(String key);
}
```

## VI. 创建插屏广告

### 6.1 创建插屏

```
// 初始化之后请求插屏广告，请求与展示广告，单独使用
AdViewInstlManager.getInstance(this).requestAd(this, MainActivity.key2);
// 请求广告成功之后，调用展示广告
AdViewInstlManager.getInstance(this).showAd(this, MainActivity.key2);
```

注：因插屏广告有一定的生存周期，请求后请不要间隔太久才去调用插屏展



示方法，以免广告失效

## 6.2 插屏接口调用

```
public interface AdViewInstlListener {  
    /**  
     * 点击事件通知  
     */  
    public void onAdClick(String key);  
    /**  
     * 展示事件通知  
     */  
    public void onAdDisplay(String key);  
    /**  
     * 消失事件通知  
     */  
    public void onAdDismiss(String key);  
    /**  
     * 请求成功通知  
     */  
    public void onAdReceived(String key);  
    /**  
     * 请求失败通知  
     */  
    public void onAdFailed (String key);  
}
```

## 6.3 创建自定义样式插屏

```
//初始化时需要设置用户托管模式，设置之后必须要手动调用展示  
InitConfiguration.setInstlControlMode(InstlControlMode.USERCONTROL);  
// 初始化之后请求插屏广告  
AdViewInstlManager.getInstance(this).requestAd(this, MainActivity.key2);  
//广告需要被展示时需要调用，返回不为null(View)则表示有广告返回，反之则没有得到广告  
//可将返回的view放置在自定义的容器中展示，如dialog  
AdViewInstlManager.getInstance(this).getInstlView (key);  
//当成功展示时需要调用展示汇报方法（必须）  
AdViewInstlManager.getInstance(this).reportImpression(key);  
//当广告被点击时需要调用点击事件处理方法，否则无响应（必须）  
AdViewInstlManager.getInstance(this).reportClick(key);
```

## VII. 创建开屏广告

### 7.1 创建开屏广告

```
// 设置开屏下方LOGO，可不设，可传本地图片或者图片链接
AdViewSpreadManager.getInstance(this).setSpreadLogo(R.drawable.spread_Logo)
;
// 设置开屏背景颜色，可不设置
AdViewSpreadManager.getInstance(this).setSpreadBackgroudColor(
    Color.WHITE);
// 请求开屏广告
AdViewSpreadManager.getInstance(this).request(this, MainActivity.key2,
    (RelativeLayout) findViewById(R.id.spreadlayout), this);
```

### 7.2 开屏接口调用

```
public interface AdViewSpreadListener {
    /**
     * 当广告被显示时调用该函数。
     */
    public void onAdDisplay(String key);
    /**
     * 当广告请求成功时调用该函数。
     */
    public void onAdReceived(String key);
    /**
     * 点击回调。
     */
    public void onAdClick(String key);
    /**
     * 当广告请求失败时调用该函数。
     */
    public void onAdFailed(String key);
    /**
     * 当广告被关闭时调用该函数。
     */
    public void onAdClose(String key);
    /**
     * 自定义回调
     */
    public void onAdNotifyCustomCallback(String key,ViewGroup view,int
ruleTime,int delayTime);
}
```

## 7.3 自定义开屏顶部倒计时通知样式

```
// 设置后顶部会显示跳过按钮
AdViewSpreadManager.getInstance(this).setSpreadNotifyType(this,
AdSpreadManager.NOTIFY_COUNTER_NUM);
// 默认值, 不显示任何通知
public final static int NOTIFY_COUNTER_NULL = 0;
// 设置后顶部显示倒计时
public final static int NOTIFY_COUNTER_NUM = 1;
// 设置后顶部显示为跳过按钮, 在规定展示时间之后才会出现
public final static int NOTIFY_COUNTER_TEXT = 2;
// 设置后将调用 onAdNotifyCustomCallback(String key,ViewGroup view,int
ruleTime,int delayTime) 接口, 可在其中自定义通知样式
public final static int NOTIFY_COUNTER_CUSTOM = 3;
```

注：开屏广告请保证广告曝光时间充足，否则将影响收入。可参照 demo 中 SpreadScreenActivity 中的代码。

## VIII. 创建原生广告

### 8.1 创建原生广告

```
//初始化原生广告, 需要提前定义广告布局, 并在应用后台申请原生广告位id
AdViewNativeManager.getInstance(this).requestAd(this,MainActivity.key2, 2,
this); //设置原生回调接口
@Override
public void onReceivedAd(String key,List arg0) {
    for (int i = 0; i < arg0.size(); i++) {
        Data data = new Data();
        NativeAdInfo nativeAdInfo = (NativeAdInfo) arg0.get(i);
        data.descript = nativeAdInfo.getDescription();
        data.icon = nativeAdInfo.getIconUrl();
        data.title = ((NativeAdInfo) arg0.get(i)).getTitle();
        data.adInfo = (NativeAdInfo) arg0.get(i);
        ((NativeAdInfo) arg0.get(i)).getIconHeight();
        data.isAd = true;
        Log.i("原生信息 ", "data.descript: " + data.descript + "\ndata.icon: "
            + data.icon + "\ndata.title:" + data.title);
        list.add(3, data);
        ((NativeAdInfo) arg0.get(i)).onDisplay(new View(
            AdNativeActivity.this));
    } }
/**
 * 当广告请求失败时调用该函数。
```

```

*/
@Override
public void onFailedReceivedAd(String arg0) {
}
/**
 * 为下载广告时调用 返回现下载内容状态。
 */
@Override
public void onAdStatusChanged (int arg0) {
}
});

```

## 8.2 原生接口调用

```

public interface AdViewNativeListener {
    /**
     * 当广告请求成功时调用该函数。
     */
    public void onAdReceived(String key ,List<NativeAdInfo> adMaps);
    /**
     * 当广告请求失败时调用该函数。
     */
    public void onAdReceived(String key);
    /**
     * 当广告状态改变。
     */
    public void onAdStatusChanged (String key ,int status);
}

```

# IX. 创建视频广告

## 9.1 创建视频广告

```

//初始化之后请求视频广告，请求与展示广告要单独使用
AdViewVideoManager.getInstance(this).requestAd(this,MainActivity.key3,this)
; //设置视频回调接口
// 请求广告成功之后，调用展示广告
AdViewVideoManager.getInstance(this).playVideo(this, MainActivity.key3);

```

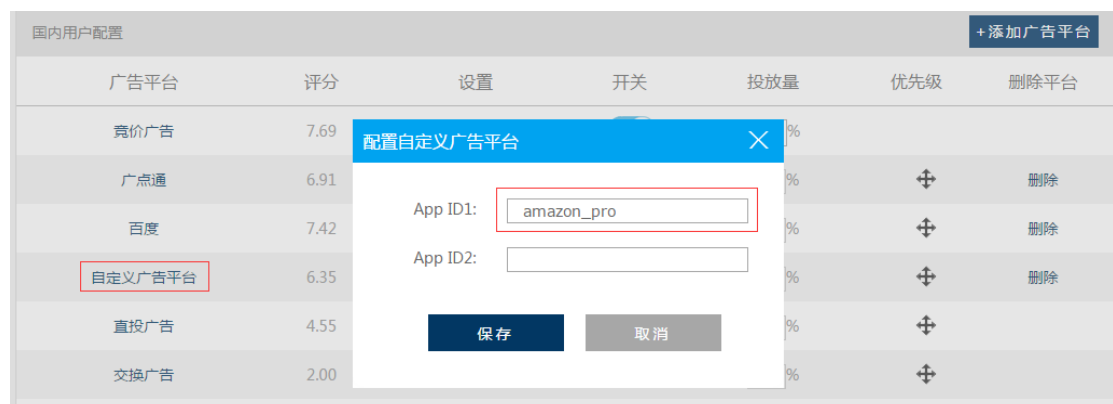
## 9.2 视频接口调用

```
public interface AdViewVideoListener{
    /**
     * 播放开始事件通知
     */
    public void onAdPlayStart(String key);
    /**
     * 播放结束事件通知
     */
    public void onAdPlayEnd(String key, Boolean isEnd);
    /**
     * 关闭事件通知
     */
    public void onAdClose(String key);
    /**
     * 请求成功通知
     */
    public void onAdReceived(String key);
    /**
     * 请求失败通知
     */
    public void onAdFailed (String key);
}
```

## X. 添加自定义广告平台

开发者有时想添加一个不是聚合里面的广告平台；Adview 提供了这种需求的实现；

添加广告平台中有个“自定义广告平台”；



开发者需要填写App ID1，这是个函数名字，需要客户端实现这个函数；这个函数的功能是调用广告平台的接口；App ID2，随便填点东西，否则最终不能保存。

## 10.1 参考代码

Demo中提供的例子是amazon广告的实现：

## 10.2 自定义函数实现

```
// 可以访问https://developer.amazon.com/sdk/mobileads.html
// 方法必须携带final AdViewAdapter adapter, final String key 这两个参数
// 否则会导致自定义广告失败
public void amazon_proc(final AdViewAdapter adapter, final String key) {
    // TODO Auto-generated method stub
    AdRegistration.enableLogging(this, true);
    AdRegistration.enableTesting(this, true);
    AdRegistration.setAppKey(this, "sample-app-v1_pub-2");

    // 创建amazon的adview实例
    adView = new AdLayout(this, AdSize.AD_SIZE_320x50);
    // 指定侦听接口
    adView.setListener(new AdListener() {
        @Override
        public void onAdLoaded(AdLayout arg0, AdProperties arg1) {
            Log.d("AdViewSample", arg1.getAdType().toString()
                + " Ad loaded successfully.");

            // 广告请求成功之后，启动定时器，到时后请求下一个广告
            adapter.reportImpression(key);
            adapter.rotateDelayedAd(key);
        }

        @Override
        public void onAdExpanded(AdLayout arg0) {
            // TODO Auto-generated method stub
        }

        @Override
        public void onAdCollapsed(AdLayout arg0) {
            // TODO Auto-generated method stub
        }

        @Override
        public void onAdFailedToLoad(AdLayout arg0, AdError arg1) {
            // TODO Auto-generated method stub
            Log.w("AdViewSample",
                "Ad failed to load. Code: "
                    + arg1.getResponseCode() + ", Message: "
                    + arg1.getResponseMessage());

            // 失败之后开始请求下一个广告
        }
    });
}
```

```
        adapter.rotatePriAd(key);
    }
});
AdViewBannerManager.getInstance(AdBannerActivity.this).addSubview(
    AdViewBannerManager.getInstance(AdBannerActivity.this)
        .getAdViewLayout(AdBannerActivity.this, key),
    adView, key);
AdTargetingOptions adOptions = new AdTargetingOptions();
adView.loadAd(adOptions);
}
```

## XI. 指定应用渠道

开发者在Manifest文件中加

```
<meta-data
    android:name="AdView_CHANNEL" android:value="GFAN"></meta-data>
```

可以配置渠道；（上传机锋时必须标明否则审核不过）；之前的AdViewTargeting.setChannel接口已经作废；

目前adview支持的渠道如下：

EOE (优亿市场)  
GOOGLEMARKET (谷歌电子市场)  
APPCHINA (应用汇)  
HIAPK (安卓市场)  
GFAN (机锋)  
GOAPK (安智)  
NDUOA (N多网)  
91Store (手机91)  
LIQUCN (历趣)  
WAPTW (天网)  
ANDROIDCN (安卓中国)  
GGDWON (G友网)  
ANDROIDAI (安卓之家)  
STARANDROID (安卓星空)  
ANDROIDD (安致)  
YINGYONGSO (应用搜)  
IMOBILE (手机之家)  
SOUAPP (搜应用)  
MUMAYI (木蚂蚁)  
MOBIOMNI (欧米)  
PAOJIAO (泡椒网)  
AIBALA (爱扒拉市场)  
COOLAPK (酷安网)  
ANFONE (安丰)  
APKOK (乐致网)  
360MARKET (360市场)

如果不配置，或配置其他的值，一律作为“OTHER”处理；  
在移动广告观察(<http://t.adview.cn/>)中有各个市场的链接；



## XII. 常见问题处理

### 12.1 如果应用想混淆（ProGuard）怎么办？

AdView是动态调用，不必对其进行混淆，广告公司代码基本已经独立混淆过，如果应用自己代码需要混淆，可以在文件proguard.cfg开头如下增加，详情可参考sample中的代码（下面的code可以去sample中copy）：

```
#The below is used for AdView SDK settings,only instead for your app
-dontwarn

# for google play service
-libraryjars '/libs/android-support-v4.jar'

-keep public class com.kyview.** {*; }
-keep public class com.kuaiyou.** {*; }
-keepclassmembers class * {public *; }
-keep public class * {public *; }
-keep class com.five.adwoad.** {*; }
-keep public class com.wooboo.** {*; }
-keep public class cn.aduu.android.**{*; }
-keep public class com.wqmobile.** {*; }
-keep class com.baidu.mobads.** {
    public protected *;
}
-keep public class com.google.android.gms.ads.** {
    public *;
}
-keep public class com.google.ads.** {
    public *;
}
-keep public class com.millennialmedia.android.* {
    <init>(...);
    public void *(...);
    public com.millennialmedia.android.MMJSResponse *(...);
}

-keep class com.qq.e.** {
    public protected *;
}
-keep class com.tencent.gdt.**{
    public protected *;
}

-dontoptimize
```

```

-dontwarn
-keep class com.mobisage.android.** {*; }
-keep interface com.mobisage.android.** {*; }
-keep class com.msagecore.** {*; }
-keep interface com.msagecore.** {*; }

#使用 触控 广告时使用
-keepattributes Exceptions
-keepattribute Signature
-keepattribute Deprecated
-keepattributes *Annotation*
#-dontwarn com.chance.**
#-dontwarn com.android.volley.NetworkDispatcher
-flattenpackagehierarchy com.chance.v4

-keep class * extends com.chance.ads.Ad {
public *;
}
-keep class com.chance.ads.AdActivity {
public *;
}
-keep class com.chance.recommend.RecommendActivity {
public *;
}
-keep class com.chance.ads.AdRequest {
public *;
}
-keep class com.chance.ads.MoreGameButton {
public *;
}
-keep class com.chance.ads.OfferWallButton {
public *;
}
-keep class com.chance.ads.RecommendButton {
public *;
}
-keep class com.chance.ads.OfferWallData {
public *;
}
-keep class com.chance.ads.OfferWallAdInfo {
public *;
}
-keep class com.chance.ads.OfferWallAdDetail {
public *;
}

```

```
}
-keep class com.chance.response.TaskInfo {
public *;
}
-keep class com.chance.exception.PBException {
public *;
}
-keep class com.chance.listener.AdListener {
public *;
}
-keep class com.chance.listener.PointsChangeListener {
public *;
}
-keep class com.chance.listener.QueryPointsListener {
public *;
}
-keep class com.chance.listener.GetAdDetailListener {
public *;
}
-keep class com.chance.listener.GetAdListListener {
public *;
}
-keep class * extends android.app.Service {
public *;
}
-keep class com.chance.report.ReportData {
public *;
}
-keep class com.chance.engine.DownloadData {
public *;
}
-keep class com.chance.engine.DownloadSubData {
public *;
}
-keep class com.chance.data.DownloadCompleteInfo {
public *;
}
-keep class com.chance.data.AppInfo {
public *;
}
-keep class com.chance.util.PBLog {
public *;
}
-keep class com.chance.recommend.** {*;}
```

```

-keep class com.chukong.android.crypto.** {*; }
-keep class com.chance.d.** {*; }

#触控 混淆结束

-keep class com.suizong.mobile.** {*; }
-keep class com.go2map.mapapi.** {*; }

-keep public class cn.Immob.sdk.** {*; }
-keep public class cn.Immob.sdk.controller.** {*; }

-keep class net.youmi.android.** {*; }

-keepclassmembernames cn.smartmad.ads.android
-keepclassmembernames I
-keep class cn.smartmad.ads.android.* {*; }
-keep class I.* {*; }

-keep public class MobWin.*
-keep public class MobWin.cnst.*
-keep class com.tencent.lbsapi.*
-keep class com.tencent.lbsapi.core.*
-keep class LBSAPIProtocol.*
-keep class com.tencent.lbsapi.core.QLBSJNI {
*;
}

-keepclassmembernames com.adchina.android.ads
-keepclassmembernames com.adchina.android.ads.controllers
-keepclassmembernames com.adchina.android.ads.views
-keepclassmembernames com.adchina.android.ads.animations
-keep class com.adchina.android.ads.*{*; }
-keep class com.adchina.android.ads.controllers.*{*; }
-keep class com.adchina.android.ads.views.*{*; }
-keep class com.adchina.android.ads.animations.*{*; }

-optimizationpasses 5
-dontusemixedcaseclassnames
-dontskipnonpubliclibraryclasses
-dontpreverify
-verbose

```

```

-optimizations !code/simplification/arithmetic,!field/*,!class/merging/*

-keep public class * extends android.app.Activity
-keep public class * extends android.app.Application
-keep public class * extends android.app.Service
-keep public class * extends android.content.BroadcastReceiver
-keep public class * extends android.content.ContentProvider
-keep public class * extends android.app.backup.BackupAgentHelper
-keep public class * extends android.preference.Preference
-keep public class com.android.vending.licensing.ILicensingService

-keepclasseswithmembers class * {
    native <methods>;
}

-keepclasseswithmembers class * {
    public <init>(android.content.Context, android.util.AttributeSet);
}

-keepclasseswithmembers class * {
    public <init>(android.content.Context, android.util.AttributeSet, int);
}

-keepclassmembers enum * {
    public static **[] values();
    public static ** valueOf(java.lang.String);
}

-keep class * implements android.os.Parcelable {
    public static final android.os.Parcelable$Creator *;
}
-keep class com.media.v.** {*;};
-keep class org.adver.score.**{*;};

-keep class com.easou.ecom.mads.** {
    public protected *;
}
-keep class com.adroi.sdk.** {
    public protected *;
}
-keep class com.mopanspot.sdk.** {
    *;
}
-keep class com.imopan.plugin.spot.** {

```

```

*;
}
-keep class com.jd.**{
    *;
}
-keep class cn.pro.ad.sdk.*

```

目前Adview SDK混淆支持proguard4.6以上的版本，开发者可以去proguard官方网站<http://sourceforge.net/projects/proguard/files/proguard/>下载4.6或以上版本；如果要升级的话，用下载的版本直接替换掉“android-sdk-windows\tools\proguard”就可以了；

## 12.2 如果有问题，和谁联系？

用户可以登录AdView，在首页底部，有服务邮箱，客服热线，企业QQ客服：

合作伙伴













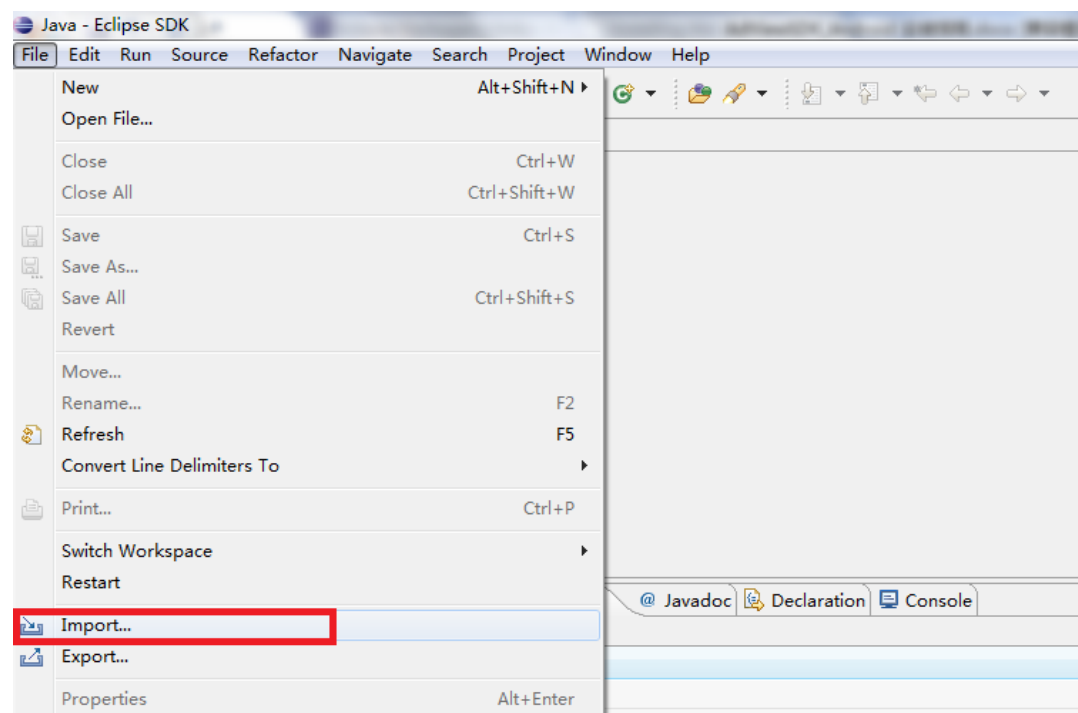
 7x24服务热线：4000-1314-00
  客服邮箱：service@kuaiyou.com
  企业客服：4000-1314-00

产品合作	开发者中心	帮助中心	关注AdView微信	关注AdView微博
解决方案	Android SDK下载	AdView使用流程	 	
商务合作	iOS SDK下载	在线帮助		
合作伙伴	开源SDK下载	联系我们		
移动广告观察	在线文档			

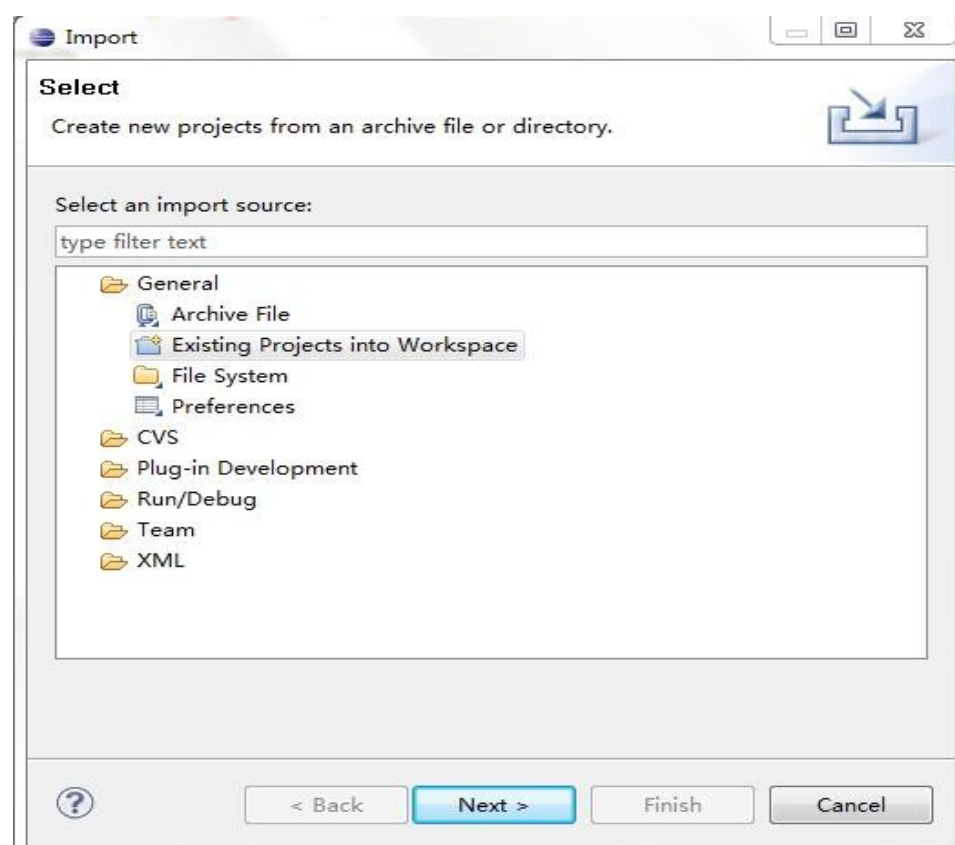
## 12.3 如何使用 sample

### 12.3.1 加载 sample 工程，方式一

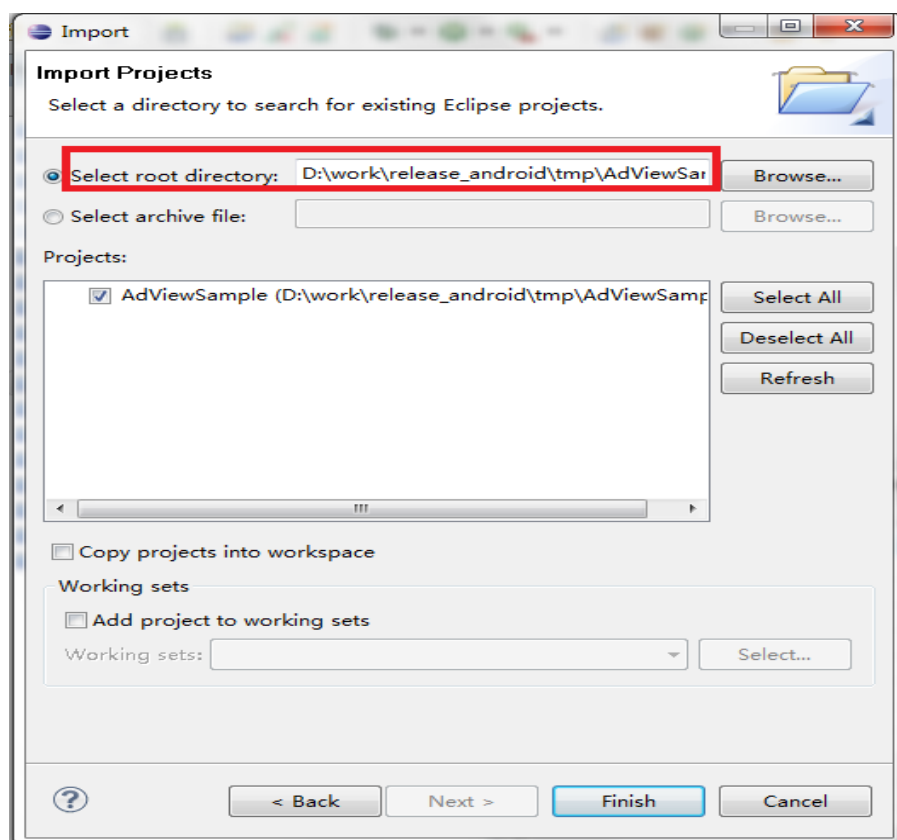
步骤一：



步骤二:



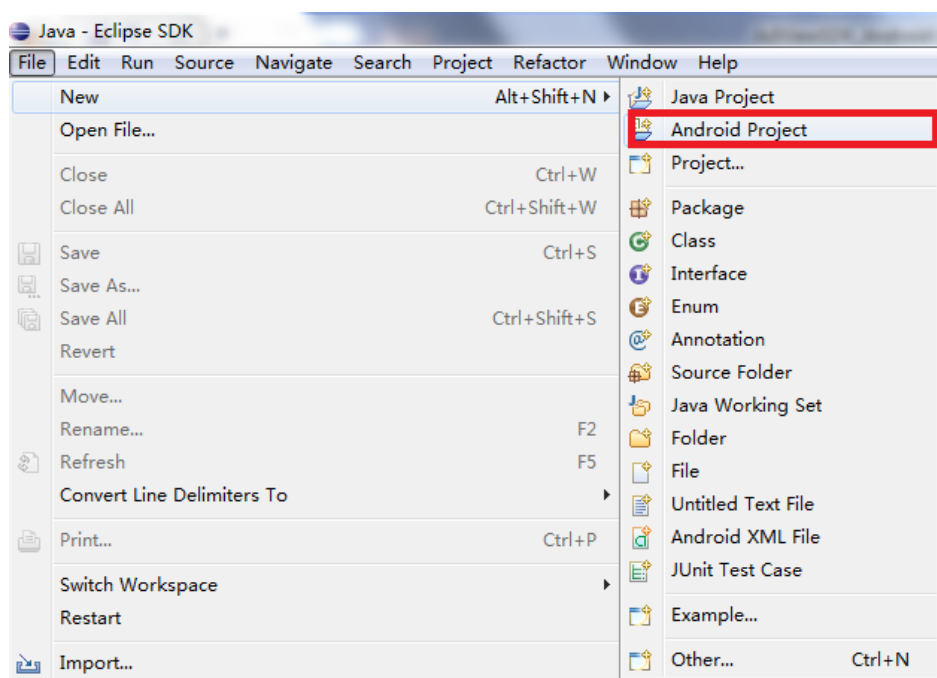
步骤三:



路径选择 sdk 包中的 sample 目录；

## 12.3.2 加载 sample 工程，方式二

步骤一：





步骤二:

路径选择 sdk 包中的 sample 目录;

默认 target 是 android1.6;

