

云点广告sdk接入文档

注意

1. 同一设备上只能有一个apk集成sdk，多个集成sdk的应用运行时不正常
2. 一定要设置设备的位置信息

1. 简介

云点广告sdk适用于Android操作系统版本（4.2~8.0）系统版本快速接入云点广告系统。云点广告sdk以下简称广告sdk。

广告sdk需要接入方使用云点平台生成商户渠道 id，接入sdk时集成到app应用中。

广告sdk主要包含以下功能：

- 支持定位或第三方定位信息传入
- 支持空闲轮播资源播放，包含视频(mp4)及图片（png,jpg）
- 支持轮播资源播放顺序及时间
- 支持线上广告资源（计费广告）在5分钟内动态安排播放
- 支持线上广告资源（计费广告）自动播放，即广告到达时立即播放线上广告
- 支持线上广告开启及停止播放

2. 广告sdk接入

2.1 配置根目录build.gradle

设置工程根目录下**build.gradle**配置

- 2.1.1 写入ext配置插件版本

```
ext{  
  
    //版本3.0.xx为使用第三方位置传入库  
    shellAssetVersion = '3.0.64'  
    //广告sdk版本  
    advSdkVersion = '1.50.0.53'  
    //播放器版本  
    shellAdvPlayerVersion = "3.0.15"  
  
}
```

- 2.1.2 配置maven 库地址

```
maven { url "http://archiva.cpo2o.com:8000/repository/internal" }
```

- 2.1.3 demo中build.gradle目录

```
// Top-level build file where you can add configuration options common to all sub-
projects/modules.

buildscript {

    repositories {
        google()
        jcenter()
    }
    dependencies {
        classpath 'com.android.tools.build:gradle:3.0.1'

        // NOTE: Do not place your application dependencies here; they belong
        // in the individual module build.gradle files
    }
}

// 2.1.1 写入ext配置插件版本
ext{

    //版本3.0.xx为使用第三方位位置传入库, 3.1.xx使用sdk自带定位
    shellAssetVersion = '3.0.64'
    //广告sdk版本
    advSdkVersion = '1.50.0.53'
    //播放器版本
    shellAdvPlayerVersion ="3.0.15"

}

allprojects {
    repositories {
        // 2.1.2 配置maven 库地址
        maven { url "http://archiva.cpo2o.com:8000/repository/internal" }
        google()
        jcenter()
    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}
```

2.2 配置应用build.gradle

- 2.2.1 配置comipleOptions

```
compileOptions {  
    sourceCompatibility JavaVersion.VERSION_1_8  
    targetCompatibility JavaVersion.VERSION_1_8  
}
```

- 2.2.2 配置插件依赖

```
// 广告sdk插件  
implementation "com.cloudpoint.plugins:adv-sdk:${advSdkVersion}"  
// 服务插件  
implementation "com.cloudpoint.plugins:shell-asset:${shellAssetVersion}"  
// 播放器插件  
implementation "com.cloudpoint.plugins:shell-adv-player:${shellAdvPlayerVersion}"
```

- 2.2.3 demo中应用**build.gradle**

```

apply plugin: 'com.android.application'

android {
    compileSdkVersion 26
    buildToolsVersion "26.0.2"
    defaultConfig {
        applicationId "com.cloudpoint.app.adv.sdk.demo"
        minSdkVersion 17
        targetSdkVersion 22
        versionCode 150001
        versionName "1.50.0.1"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
    // 2.2.1 配置comipleOptions
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}

dependencies {
    implementation fileTree(include: ['*.jar'], dir: 'libs')

    //noinspection GradleCompatible
    implementation 'com.android.support:appcompat-v7:26.1.0'
    implementation 'com.android.support.constraint:constraint-layout:1.1.3'
    implementation 'com.android.support:design:26.1.0'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.1'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.1'

    //2.2.2 配置插件依赖
    // 广告sdk插件
    implementation "com.cloudpoint.plugins:adv-sdk:${advSdkVersion}"
    // 服务插件
    implementation "com.cloudpoint.plugins:shell-asset:${shellAssetVersion}"
    // 播放器插件
    implementation "com.cloudpoint.plugins:shell-adv-player:${shellAdvPlayerVersion}"
}

```

3. 广告sdk集成

广告sdk集成成应用app需要以下几个方面的接入：

- 界面集成播放器视图布局
- Application 进行sdk初始化加载
- 扩展`IAdvertisementEvent`接口，接收计费信息或广告信息进行手动控制播放
- Activity中绑定视图及解绑视图

3.1 集成视图布局

在界面布局中加入广告sdk的播放器

```
<com.cloudpoint.plugins.sdk.adv.AdvView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/adv_view"
    app:localPlayer="false"
/>
```

demo中的activity布局

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android
.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.cloudpoint.app.adv.sdk.demo.AdvViewActivity">

    <com.cloudpoint.plugins.sdk.adv.AdvView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/adv_view"
        app:localPlayer="false"
        />

    </com.cloudpoint.plugins.sdk.adv.AdvView>

</android.support.constraint.ConstraintLayout>
```

3.2 在应用中集成

接入广告sdk过程中有以下步骤：

1. 设置是否输出debug日志
2. 初始化广告sdk
 - 写入接入渠道编码
 - 设置播放模式
 - 设置是否启用轮播图
3. 增加回调检测是否加载播放器
4. 设置**IAdvertisementEvent**处理计费或播放模式
5. 设置位置信息（第三方提供位置信息），写入DeviceLocation对象

```
CPAdvSdk.setLocation(loc)
```

6. 设置位置信息（使用sdk定位） 使用sdk定位还需要修改以下文件：
 - 设置根目录下*build.gradle*中 **shellAssetVersion** 版本为 *3.1.xx*
 - 修改百度定位sdk 的密钥 ,配置AndroidManifest.xml

```
<!--集成百度定位sdk, 请输入-->
<meta-data
    android:name="com.baidu.lbsapi.API_KEY"
    android:value="8y4LVGZuSQMHDms3FK0sGjVDN5ChcQf0"
    tools:replace="android:value" />

<service
    android:name="com.baidu.location.f"
    android:enabled="true"
    android:process=":baidLocation" />
```

- 设置位置信息为null

```
CPAdvSdk.setLocation(null)
```

7. 集成代码至Application中。

```
//1. 设置是否输出debug日志
CPAdvSdk.setDebug(BuildConfig.DEBUG);

// 2. 初始化应用
boolean initialized = CPAdvSdk.init(AdvApplication.this,
    "10000", //接入渠道编码
    true, //是否启用手动控制播放
    true //是否启用轮播图
);

if(initialized) {
```

```

// 3. 检测是否初始化播放器
CPAdvSdk.setPlayerListener(new IAdvPlayer() {
    @Override
    public void onPlayerError() {
        System.out.println("播放器未加载! ");
    }
});

//4. 处理计费或手动播放线上广告
CPAdvSdk.setAdvertisementEvent(new AdvEventListener());

//5. 设置位置信息
DeviceAddress address = new DeviceAddress();
address.setAddress("中国北京市朝阳区北苑路229号");
address.setCity("北京市");
address.setDistrict("朝阳区");
address.setLocationDescribe("在金泉港附近");
address.setProvince("北京市");
address.setStreet("北苑路");

DeviceLatitudeLongitude gps = new DeviceLatitudeLongitude();
gps.setCoorType("bd0911");
gps.setErrorCode(161);
gps.setLangtitude(116.423292);
gps.setLatitude(40.010727);
gps.setRadius(50.499428f);

DeviceLocation location = new DeviceLocation(gps, address);
Gson g = new Gson();
String loc = g.toJson(location);
//
// {"location":{"address":"中国北京市朝阳区北苑路229号","city":"北京市","di
sriect":"朝阳区","location_desc":"在金泉港附近","province":"北京市","street":"北苑路"},"
gps":{"coor_type":"bd0911","error_code":161,"langtitude":116.423292,"latitude":40.
010727,"radius":50.499428}}
//6. 设置 location信息
//loc = null; // 使用集成百度定位sdk时, 将loc设置为空
CPAdvSdk.setLocation(loc);
}

```

- demo中AdvApplication

```

public class AdvApplication extends Application {

    @Override
    public void onCreate() {
        super.onCreate();
    }
}

```

```

//设置是否输出debug日志
CPAdvSdk.setDebug(BuildConfig.DEBUG);

// 4. 初始化应用
boolean initialized = CPAdvSdk.init(AdvApplication.this,"10000",true,true)
;

if(initialized) {
    // 5. 检测是否初始化播放器
    CPAdvSdk.setPlayerListener(new IAdvPlayer() {
        @Override
        public void onPlayerError() {
            System.out.println("播放器未加载! ");
        }
    });

    //处理计费或手动播放线上广告
    CPAdvSdk.setAdvertisementEvent(new AdvEventListener());

    //设置位置信息
    DeviceAddress address = new DeviceAddress();
    address.setAddress("中国北京市朝阳区北苑路229号");
    address.setCity("北京市");
    address.setDistrict("朝阳区");
    address.setLocationDescribe("在金泉港附近");
    address.setProvince("北京市");
    address.setStreet("北苑路");

    DeviceLatitudeLongitude gps = new DeviceLatitudeLongitude();
    gps.setCoorType("bd0911");
    gps.setErrorCode(161);
    gps.setLangtitude(116.423292);
    gps.setLatitude(40.010727);
    gps.setRadius(50.499428f);

    DeviceLocation location = new DeviceLocation(gps, address);
    Gson g = new Gson();
    String loc = g.toJson(location);
    //
    // {"location":{"address":"中国北京市朝阳区北苑路229号","city":"北京市","di
sstrict":"朝阳区","location_desc":"在金泉港附近","province":"北京市","street":"北苑路
"},"gps":{"coor_type":"bd0911","error_code":161,"langtitude":116.423292,"latit
ude":40.010727,"radius":50.499428}}
    //6. 设置 location信息
    //loc = null; // 使用集成百度定位sdk时, 将loc设置为空
    CPAdvSdk.setLocation(loc);
}

```



```

}

@Override
public void onTerminate() {

    super.onTerminate();

}
}

```

4.绑定及解绑定视图

广告sdk在绑定视图时启动服务连接服务器，将视图设置为空时服务关闭

```

public class AdvViewActivity extends Activity {

    @Override
    public void onAttachedToWindow() {

        this.getWindow().addFlags(
            WindowManager.LayoutParams.FLAG_FULLSCREEN
            | WindowManager.LayoutParams.FLAG_SHOW_WHEN_LOCKED
            | WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON
            | WindowManager.LayoutParams.FLAG_DISMISS_KEYGUARD
        );

        super.onAttachedToWindow();

    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_adv_view);
    }

    AdvView advView;
    private void bindView(){
        advView = findViewById(R.id.adv_view);
    }
}

```

```

        //绑定视图，之后会启动服务，设置视图为null，停止服务
        CPAdvSdk.bindView(advView);

    }

    @Override
    protected void onPause() {
        super.onPause();
    }

    @Override
    protected void onResume() {
        super.onResume();
        // 3. 绑定view
        bindView();
    }

}
}

```

5. 计费及模式处理

5.1 手动模式处理

广告sdk设置为手动模式时，在*setAdv*方法中接收到服务器传入的广告信息，调用

```
AdvPlayerHandler.getInstance().play(adv);
```

执行播放线上推送广告。三方接入客户可以选择在其它业务结束后展示广告。需要在5分钟内完成广告展示。在接收及展示过程中不能对播放器视图进行解绑定操作。

5.2 自动模式处理

自动模式时不需要处理*setAdv*方法中的*Adv*数据。

5.3 计费及统计

notifyAdvTrackerStates 方法回调广告id及播放状态，三方统计时需要进程去重处理。具体状态为：

- 0： 播放开始

- 1: 播放结束
- 2: 计费结束
- -1 或其它: 播放失败

```
public class AdvEventListener implements IAdvertisementEvent {

    private static final String TAG = "AdvEventListener";

    private void d(String message){
        Log.d(TAG,message);
    }

    @Override
    public IAdvertisementPlayerListener getPlayer() {
        return null;
    }

    /**
     *
     *
     * 接收广告信息
     *
     * @param adv 接收到服务器推送广告
     * @param b true:线上广告
     */
    @Override
    public void setAdv(final Adv adv, boolean b) {
        d(adv.toString());

        //收到广告准备好的信息了。

        //开始播放广告，必须在广告超时前完成播放。
        AdvPlayerHandler.getInstance().play(adv);
    }

    /**
     *
     * 播放器开始播放
     */
    @Override
    public void onStart() {

        d(" onStart");

    }
}
```

```
/**
 * 播放器播放失败
 */
@Override
public void onError() {

    d(" onError");

}

/**
 *
 * 播放器播放结束
 */
@Override
public void onEnd() {
    d(" onEnd");

}

/**
 *
 *
 * 计费结束
 */
@Override
public void onCleanup() {
    d(" onCleanup");

}

/**
 *
 * 通知广告状态变化, 包含广告id,
 *
 *
 * @param s 广告id, 唯一标识
 * @param i 广告状态
 *
 *
 * 0: 播放开始
 * 1: 播放结束
 * 2: 计费结束
 * -1 或其它: 播放失败
 */
@Override
public void notifyAdvTrackerStates(String s, int i) {
```

```
d(" advId" + s + " , code:"+i);
```

```
}
```

```
}
```