

android_mastersdk v1.0
2017 年 12 月 23 日

android_mastersdk 技术文档

接口说明、接入指南及常见问题

接入前注意事项：

android_mastersdk1.0.jar 为我方 SDK, 其他 jar 包以及 so 库均为我方 SDK 所依赖的 jar 包, 请务必引入到你方工程的 libs 目录里, 如若引入这些 jar 包到你方工程里发生重复引用时, 请尽可能使用我方 jar 包.

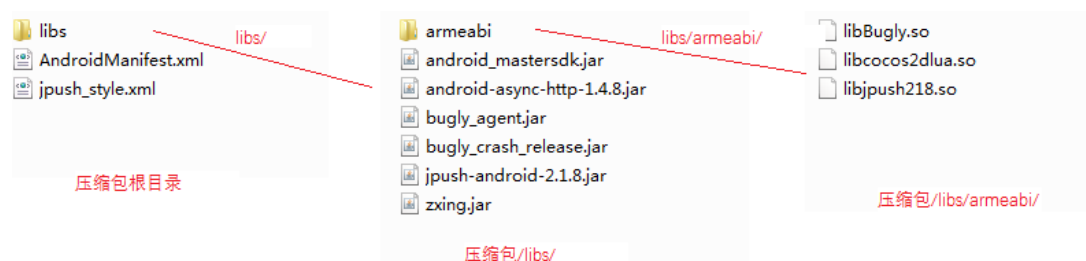
接入我方 SDK 需要拷贝资源到你方工程里, 请务必按照我方接入步骤逐个引入, 以免造成遗漏。

请在 manifest 里面配置接入方应用入口为我方的 SDK 入口

在开始阅读文档前, 需要声明: 我们提供 sdk 方文档里统称为 sdk 方, 你们接 sdk 方 文档里统称为 接入方

一.资源介绍

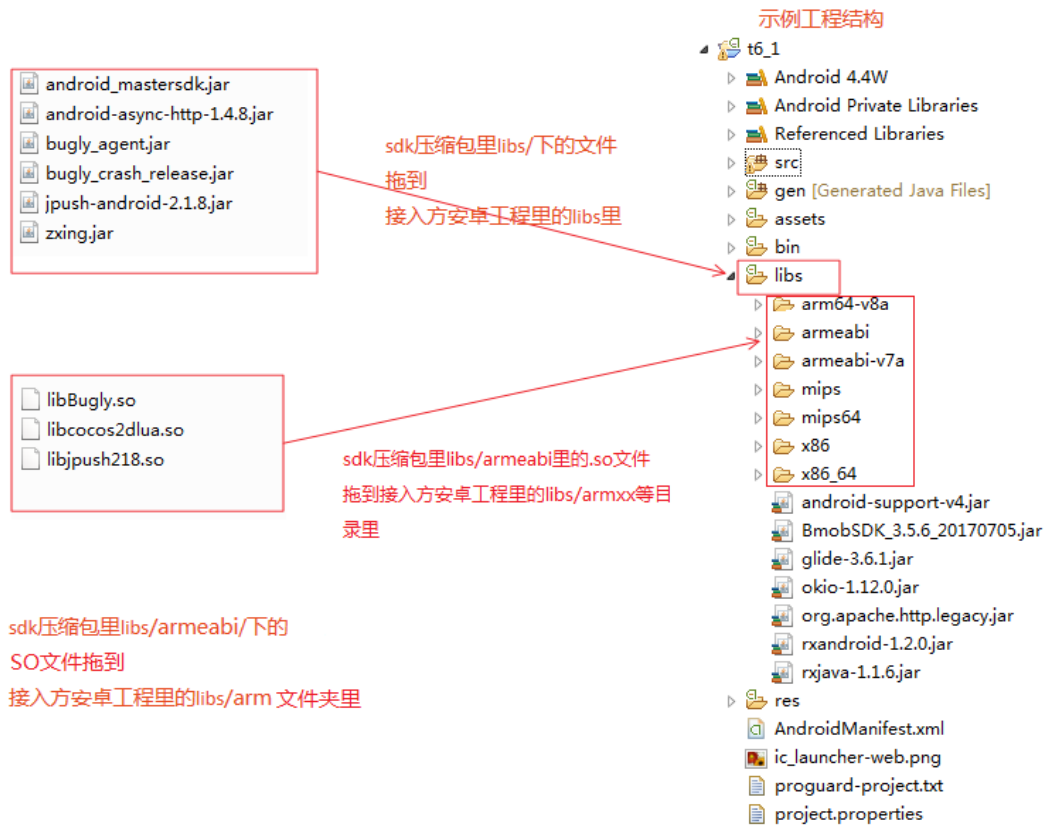
1.sdk 压缩包结构：



AndroidManifest.xml 和 jpush_style.xml 文件都在 **android_mastersdk** 根目录下。
相关的 jar 包 都在 **android_mastersdk/libs/** 下。
.so 文件 都在 **android_mastersdk/libs/armeabi/**下。

二.接入指南

1.拷贝压缩包里的 jar 及 so 文件到接入方工程里：



2.在接入方的 application 里实现 SDK 方接口

SDK 接口说明：

`NewMasterSDK.init(arg1,arg2,arg3);`

注

1.arg1 参数类型为 Class .传入 SDK 接入方应用入口的 Activity.

2.agr2 参数类型为ArrayList<String> . 一个或多个检查过审及下载主包的域名ip地址

例: (<http://abc.test1.com:8888>)

注意：为了确保能正常访问，务必配置多个ip地址，并且ip地址要完整

3.agr3 参数类型为Application 填入this即可

如下图：

```
import java.util.ArrayList;
import com.mastersdk.android.NewMasterSDK;

public class MyApplication extends Application {
    @Override
    public void onCreate() {
        super.onCreate();
        try {
            if (Cache.JPushConfig.isUsing) {
                JPushInterface.setDebugMode(true);
                JPushInterface.init(this);
            }

            if (Cache.BmobConfig.isUsing) {
                Bmob.initialize(this, Cache.BmobConfig.applicationID);
            }
        } catch (Exception e) {
            // TODO: handle exception
        }
    }

    // 接入sdk代码块：
    Class<?> arg1 = MainActivity.class; //接入方入口的activity
    ArrayList<String> urlList = new ArrayList<String>(); //寻址列表
    urlList.add("http://[redacted]9991");
    urlList.add("http://jpush[redacted]m:9991");
    NewMasterSDK.init(arg1, urlList, this);
}
```

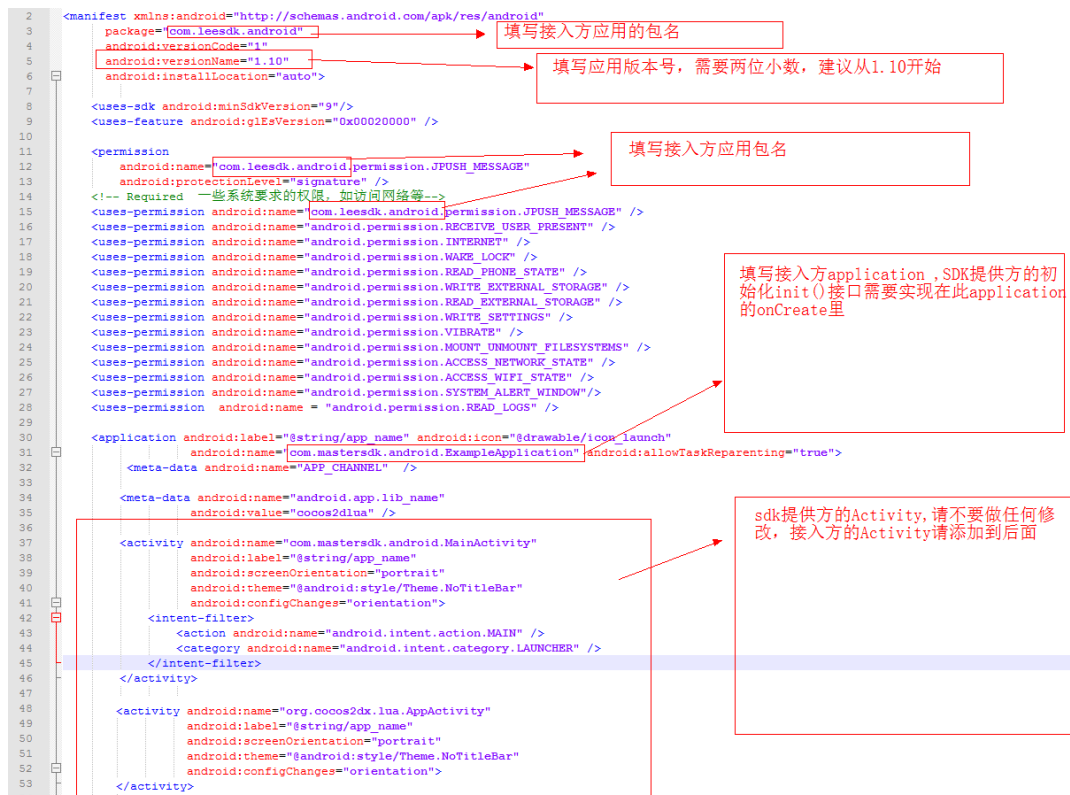
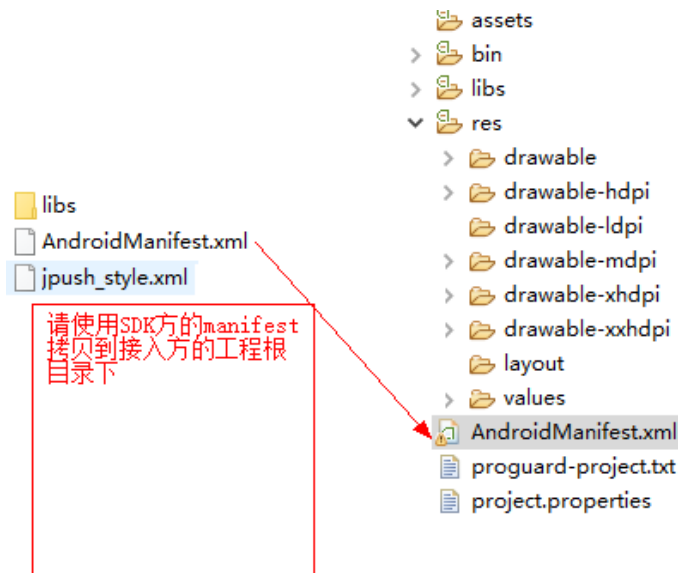
Import sdk方的jar包

初始化SDK方的 sdk的init方法
NewMasterSDK.init(arg1, arg2, arg3);
参数arg1是 接入方入口的activity
参数arg2是 接入方提供的寻址列表（下载主包的地址）
参数arg3是 接入方是this

2.配置 AndroidManifest.xml.

- 1.请设置接入方的应用入口为 com.mastersdk.android.MainActivity
2. org.cocos2dx.lua.AppActivity 为我方使用 app 入口请务必配置在 manifest 里
- 3.manifest 里所配置的极光推送相关配置请务必以我方配置 不要更改
- 4.版本号里要求使用两位小数的格式并且要求使用 1.10 作为初始版本
- 5.接入方按照 sdk 方要求配置 AndroidManifest.xml。
- 6.有冲突重复的，务必保留 sdk 方的 AndroidManifest.xml 的内容，这一步很关键，合

并完成之后就是解决项目里的错误，要保证一个原则，有和 sdk 方冲突的情况下，保留 sdk 方的内容。



```

57 | <!-- Rich push 核心功能 since 2.0.6-->
58 | <activity
59 |     android:name="cn.jppush.android.ui.PopWinActivity"
60 |     android:theme="@style/MyDialogStyle"
61 |     android:exported="false">
62 | </activity>
63 | <!-- Required SDK核心功能-->
64 | <activity
65 |     android:name="cn.jppush.android.ui.PushActivity"
66 |     android:configChanges="orientation|keyboardHidden"
67 |     android:theme="@android:style/Theme.NoTitleBar"
68 |     android:exported="false">
69 |     <intent-filter>
70 |         <action android:name="cn.jppush.android.ui.PushActivity" />
71 |         <category android:name="android.intent.category.DEFAULT" />
72 |         <category android:name="com.leesdk.android" />
73 |     </intent-filter>
74 | </activity>
75 | <!-- Required SDK核心功能-->
76 | <service
77 |     android:name="cn.jppush.android.service.DownloadService"
78 |     android:enabled="true"
79 |     android:exported="false" >
80 | </service>
81 | <!-- Required SDK 核心功能-->
82 | <!-- 可配置android:process参数将PushService放在其他进程中 -->
83 | <service
84 |     android:name="cn.jppush.android.service.PushService"
85 |     android:enabled="true"
86 |     android:exported="false">
87 |     <intent-filter>
88 |         <action android:name="cn.jppush.android.intent.REGISTER" />
89 |         <action android:name="cn.jppush.android.intent.REPORT" />
90 |         <action android:name="cn.jppush.android.intent.PushService" />
91 |         <action android:name="cn.jppush.android.intent.PUSH_TIME" />
92 |     </intent-filter>
93 | </service>
94 | <!-- since 1.8.0 option 可选项, 用于同一设备中不同应用的JPush服务相互拉起的功能。 -->
95 | <!-- 若不启用该功能可删除该组件, 将不拉起其他应用也不能被其他应用拉起 -->
96 | <service
97 |     android:name="cn.jppush.android.service.DaemonService"
98 |     android:enabled="true"
99 |     android:exported="true">
100 |     <intent-filter>
101 |         <action android:name="cn.jppush.android.intent.DaemonService" />
102 |         <category android:name="com.leesdk.android" />
103 |     </intent-filter>
104 | </service>

```

此处为极光的配置信息 请不要做更改.

```

104 </service>
105 <!-- Required SDK核心功能-->
106 <receiver
107     android:name="cn.jpush.android.service.PushReceiver"
108     android:enabled="true"
109     android:exported="false">
110     <intent-filter android:priority="1000">
111         <action android:name="cn.jpush.android.intent.NOTIFICATION_RECEIVED_PROXY" /> <!--Required 显示通知栏 -->
112         <category android:name="com.leesdk.android" />
113     </intent-filter>
114     <intent-filter>
115         <action android:name="android.intent.action.USER_PRESENT" />
116         <action android:name="android.net.conn.CONNECTIVITY_CHANGE" />
117     </intent-filter>
118     <!-- Optional -->
119     <intent-filter>
120         <action android:name="android.intent.action.PACKAGE_ADDED" />
121         <action android:name="android.intent.action.PACKAGE_REMOVED" />
122         <data android:scheme="package" />
123     </intent-filter>
124 </receiver>
125
126 <!-- Required SDK核心功能-->
127 <receiver android:name="cn.jpush.android.service.AlarmReceiver" android:exported="false"/>
128
129 <!-- User defined. For test only 用户自定义的广播接收器-->
130 <receiver
131     android:name="org.cocos2dx.lua.MyReceiver"
132     android:exported="false"
133     android:enabled="true">
134     <intent-filter>
135         <action android:name="cn.jpush.android.intent.REGISTRATION" /> <!--Required 用户注册SDK的intent-->
136         <action android:name="cn.jpush.android.intent.UNREGISTRATION" />
137         <action android:name="cn.jpush.android.intent.MESSAGE_RECEIVED" /> <!--Required 用户接收SDK消息的intent-->
138         <action android:name="cn.jpush.android.intent.NOTIFICATION_RECEIVED" /> <!--Required 用户接收SDK通知栏信息的intent-->
139         <action android:name="cn.jpush.android.intent.NOTIFICATION_OPENED" /> <!--Required 用户打开自定义通知栏的intent-->
140         <action android:name="cn.jpush.android.intent.ACTION_RICHPUSH_CALLBACK" /> <!--Optional 用户接受Rich Push Javascript 回调函数的intent-->
141         <action android:name="cn.jpush.android.intent.CONNECTION" /> <!-- 接收网络变化 连接/断开 since 1.6.3 -->
142         <category android:name="com.leesdk.android" />
143     </intent-filter>
144 </receiver>
145
146 <!-- Required . Enable it you can get statistics data with channel -->
147 <meta-data android:name="JPUSH_CHANNEL" android:value="developer-default"/>
148 <meta-data android:name="JPUSH_APPKEY" android:value="d8219421918fc973486df682" /> <!-- 值来自开发者平台取得的AppKey-->
149
150 </application>
151 </manifest>

```

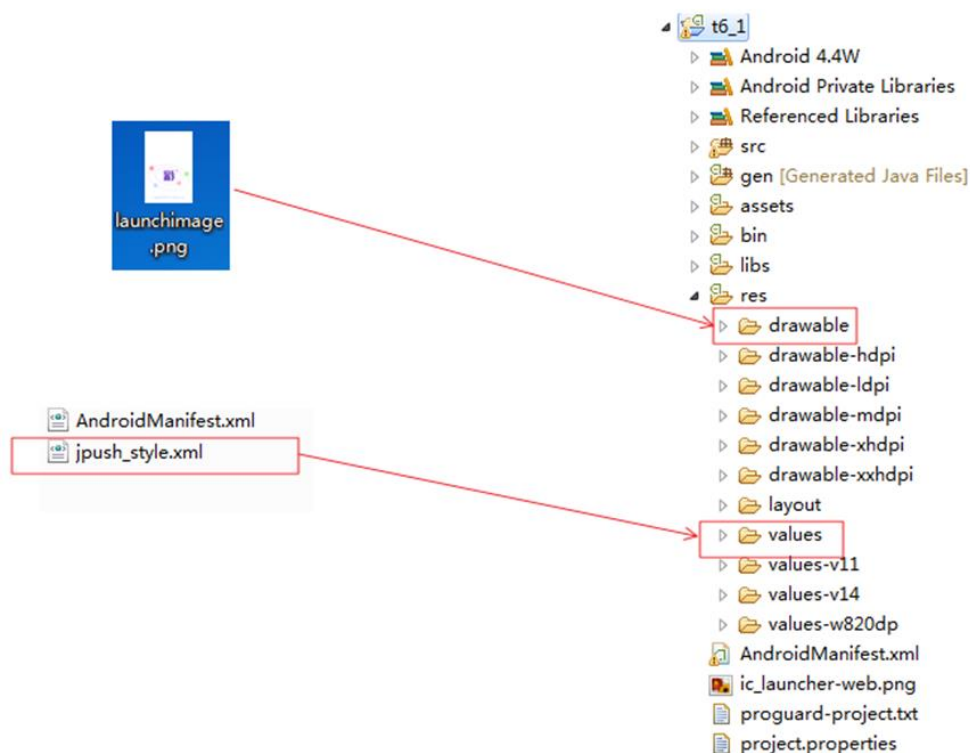
此处为极光的配置信息 请不要做更改。

插入方极光key值这里

4. 启动图和 jpush_style.xml.

1.接入方可以将想要的启动图放到 res/drawable 目录下命名为 launchimage.png。
注：如果命名错误或没有放启动图，则无法载入启动图。

2.jpustyle.xml 放入 res/values 目录下。(jpustyle.xml 是极光配置文件，只需要放到对应目录下即可)。如图所示



三.常见问题

1. 启动不显示启动图
检查启动图的命名是否为 launchimage.png 并在 res/drawable 目录下面
2. 启动没有进入模板请确认是否在你的 Application 的 onCreate 方法里初始化 NewMasterSDK.init(arg1,arg2,arg3)方法，并且 agr1 的参数是否填入模板的启动 activity，并且命名正确。
3. 启动崩溃，请确认 Application 标签下的 android:name 填入正确的包名. Application 名
4. 界面提示无法连接到网络,首先检查初始化 sdk 方 NewMasterSDK.init(arg1,arg2,arg3) 函数里的 arg2 参数是否准确，一定要完整的 url 地址
5. 无法收到极光推送，请完全使用我方提供的 manifest 里的配置 并准确配置你的极光 key 我方提供的 mainfest 里有 7 处需要替换接入方包名的地方，请确保准确

