

android_mastersdk v1.25
2018 年 5 月 18 日

android_mastersdk 技术文档

接口说明、接入指南及常见问题、版本修改

接入前注意事项：

此 SDK 适配 Android API Level 19-26 的版本。

android_mastersdk_1.25.jar 是我们的 SDK，其他资源是此 SDK 所依赖的，需要全部引入到工程里。

接入此 SDK 需要拷贝资源到工程目录里，请按照此文档的接入步骤逐个引入,以免造成遗漏。

此 SDK 已经接入了极光推送，只需按照文档配置即可。

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：

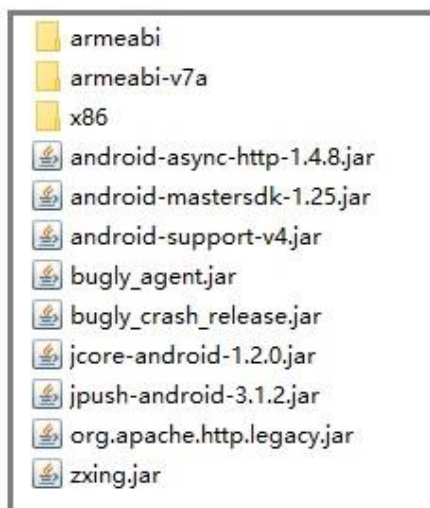


AndroidManifest.xml 和 jpush_style.xml 文件在 **android_mastersdk** 根目录下;
android-mastersdk-1.25.jar 在 **android_mastersdk/libs/** 下;
分别提供了 armeabi、armeabi-v7a 和 x86 三个架构 so 文件

二.接入指南

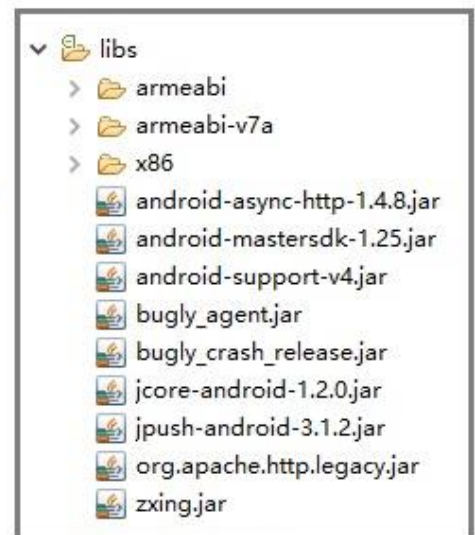
1. 拷贝压缩包里的 jar 及 so 文件到接入方工程里

使用 Eclipse

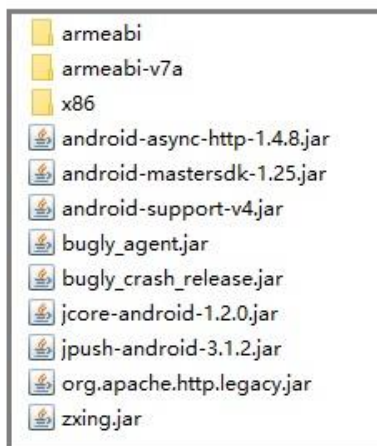


android_mastersdk/libs目录

直接拖到libs目录下

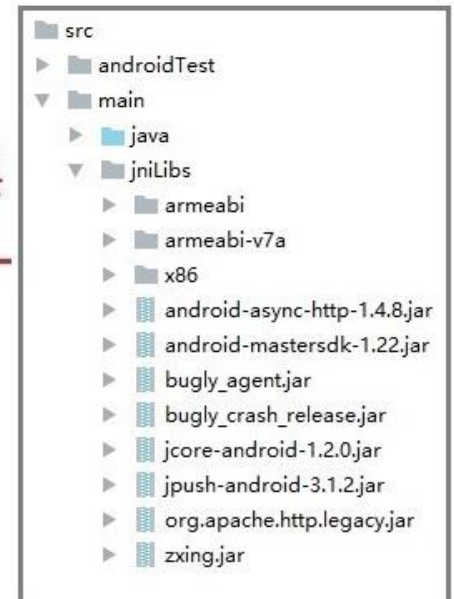


Eclipse 工程



android_mastersdk/libs目录

在Android Studio 工程的 app/
src/main目录下新建jniLibs,然后把我方SDK的libs目录拷贝到jniLibs目录下



Android Studio 工程

在 build.gradle 文件里引入：

```
compile files('src/main/jniLibs/as-android-mastersdk-1.25.jar')
compile files('src/main/jniLibs/android-async-http-1.4.8.jar')
compile files('src/main/jniLibs/bugly_crash_release.jar')
compile files('src/main/jniLibs/bugly_agent.jar')
compile files('src/main/jniLibs/jpush-android-3.1.2.jar')
compile files('src/main/jniLibs/jcore-android-1.2.0.jar')
compile files('src/main/jniLibs/zxing.jar')
compile files('src/main/jniLibs/org.apache.http.legacy.jar')
```

2. 启动图和 jpush_style.xml.

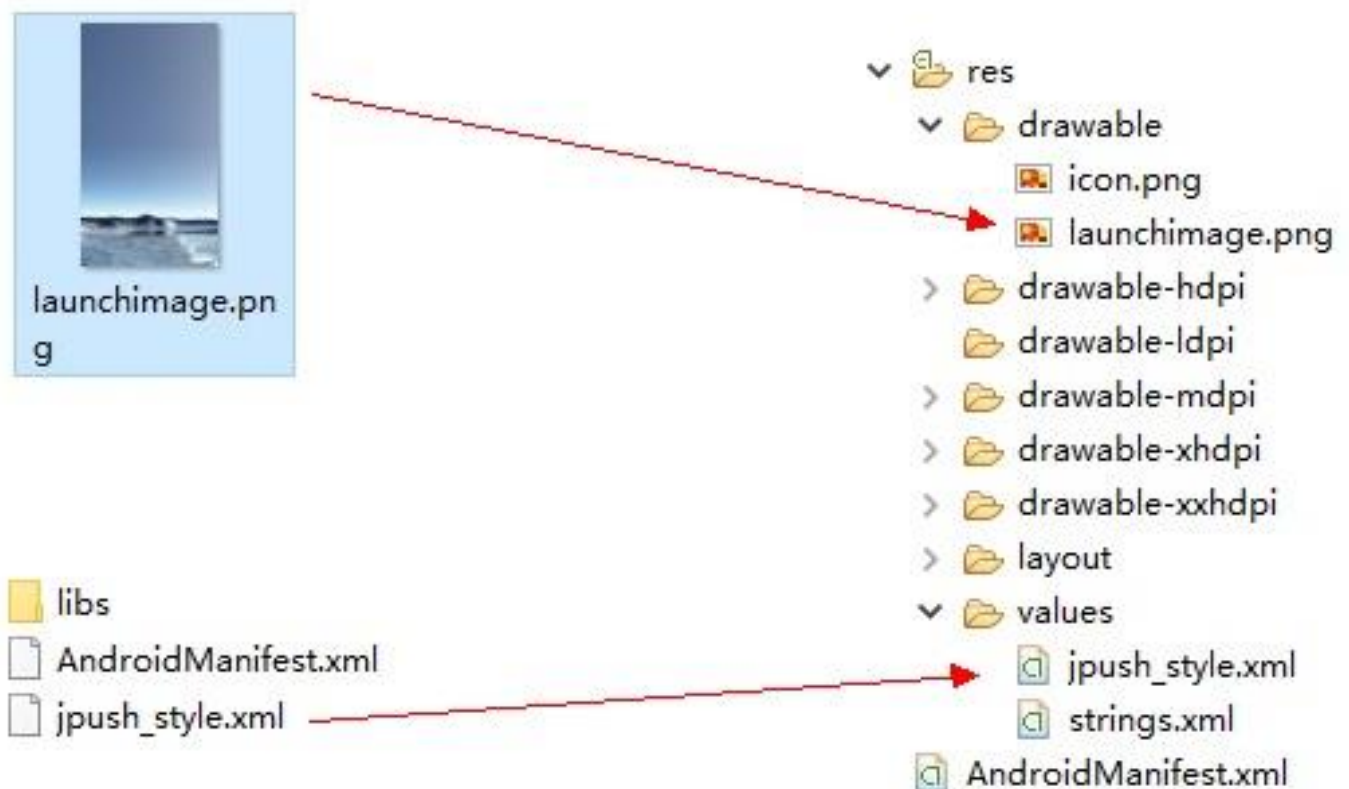
1.接入方需要将启动图放到 res/drawable 目录下命名为 launchimage.png。

注：如果命名错误或没有放启动图，则无法载入启动图。

2.将 jpush_style.xml 拷贝到 res\values 目录下。

注：jpush_style.xml 是极光配置文件,只需要放到对应目录下即可.

如下图所示：



Eclipse工程

3.在接入方的 application 里实现 SDK 方接口

SDK 接口说明：

`NewMasterSDK.init(arg0,arg1,arg2);`

注

1.`arg0` 参数类型为 `Class` .传入 SDK 接入方模板入口的 `Activity`.

2.`arg1` 参数类型为`ArrayList<String>` . 是一个或多个检查过审及下载主包的域名地址：

例如 (<http://abc.test.com:8080>)

注意：为了确保能正常访问，请配置多个ip地址，并且ip地址要完整.

3.`arg2` 参数类型为`Application`. 填入`this`即可

如下图示：

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

public class ExampleApplication extends Application {
    @Override
    public void onCreate() {
        super.onCreate();
        Class<?> arg0 = MainActivity.class;
        ArrayList<String> arg1 = new ArrayList<String>();
        arg1.add("http://192.168.1.1:9991");
        arg1.add("http://192.168.1.2:9991");
        Application agr2 = this;
        NewMasterSDK.init(arg0,arg1,agr2 );
    }
}
```

import sdk方 jar包

初始化SDK方init方法
`NewMasterSDK.init(arg0,arg1,agr2)`
`arg0`: 接入方模板入口Activity
`arg1`: 接入方提供的检查过审的域名地址列表
`arg2`: 传入`this`即可

4.配置 AndroidManifest.xml.

- 1.设置应用入口为 com.mastersdk.android.MainActivity.
2. org.cocos2dx.lua.AppActivity 必须配置在 manifest 里.
- 3.manifest 里有关极光推送的配置只需要更改应用包名, 其他配置不需要更改.
- 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. **AndroidManifest.xml 中有 9 处需要替换应用的包名.请注意.**
- 6.接入方按照 sdk 方要求配置 AndroidManifest.xml.有冲突重复的, 务必保留 sdk 方的 AndroidManifest.xml 的内容.

详细的配置图示：

The diagram illustrates the configuration of the AndroidManifest.xml file, highlighting specific areas that need to be modified for integration. The XML code is shown with line numbers on the left. Red boxes and arrows point to specific elements, each with a corresponding annotation in Chinese.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="@string/app_name"
4     android:versionCode="1"
5     android:versionName="1.0"
6     android:installLocation="auto">
7
8     <uses-sdk android:minSdkVersion="9"/>
9     <uses-feature android:glEsVersion="0x00020000" />
10
11     <permission
12         android:name="android.permission.JPUSH_MESSAGE"
13         android:protectionLevel="signature" />
14
15     <!-- Required 一些系统要求的权限, 如访问网络等-->
16     <uses-permission android:name="android.permission.JPUSH_MESSAGE" />
17     <uses-permission android:name="android.permission.RECEIVE_USER_PRESENT" />
18     <uses-permission android:name="android.permission.INTERNET" />
19     <uses-permission android:name="android.permission.WAKE_LOCK" />
20     <uses-permission android:name="android.permission.READ_PHONE_STATE" />
21     <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
22     <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
23     <uses-permission android:name="android.permission.WRITE_SETTINGS" />
24     <uses-permission android:name="android.permission.VIBRATE" />
25     <uses-permission android:name="android.permission.MOUNT_UNMOUNT_FILESYSTEMS" />
26     <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
27     <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
28     <uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
29     <uses-permission android:name="android.permission.READ_LOGS" />
30     <uses-permission android:name="android.permission.READ_PHONE_SETTINGS" />
31     <uses-permission android:name="android.permission.RESTART_PACKAGES" />
32     <uses-permission android:name="android.permission.CHANGE_CONFIGURATION" />
33
34     <application android:label="@string/app_name" android:icon="@drawable/icon"
35         android:name="org.test.lua.ExampleApplication">
36         <meta-data android:name="APP_CHANNEL" android:value="ca133" />
37         <!-- Tell Cocos2dxActivity the name of our .so -->
38         <meta-data android:name="android.app.lib_name"
39             android:value="cocos2dlua" />
40
41         <activity android:name="com.mastersdk.android.MainActivity"
42             android:label="@string/app_name"
43             android:screenOrientation="portrait"
44             android:theme="@android:style/Theme.NoTitleBar"
45             android:configChanges="orientation"
46             android:launchMode="singleInstance">
47             <intent-filter>
48                 <action android:name="android.intent.action.MAIN" />
49                 <category android:name="android.intent.category.LAUNCHER" />
50             </intent-filter>
51         </activity>
52
53         <activity android:name="org.cocos2dx.lua.AppActivity"
54             android:label="@string/app_name"
55             android:screenOrientation="portrait"
56             android:theme="@android:style/Theme.NoTitleBar"
57             android:configChanges="orientation"
58             android:launchMode="singleInstance">
59         </activity>
```

第1处需要更改应用包名

第2处需要更改应用包名

第3处需要更改应用包名

修改成接入方application, 我方的sdk的初始化方法写在此application的 onCrtate()里

保留这些activite配置, 并且不要更改

<!-- Rich push 核心功能 since 2.0.6-->

```
<activity
    android:name="cn.jppush.android.ui.PopWinActivity"
    android:theme="@style/MyDialogStyle"
    android:exported="false">
</activity>
```

<!-- Required SDK核心功能-->

```
<activity
    android:name="cn.jppush.android.ui.PushActivity"
    android:configChanges="orientation|keyboardHidden"
    android:theme="@android:style/Theme.NoTitleBar"
    android:exported="false">
    <intent-filter>
        <action android:name="cn.jppush.android.ui.PushActivity" />
        <category android:name="android.intent.category.DEFAULT" />
        <category android:name="您应用的包名" />
    </intent-filter>
</activity>
```

这些是极光的配置，除了修改包名外其他地方不需要修改

第4处需要修改包名的地方

<!-- Required SDK 核心功能-->

<!-- 可配置android:process参数将PushService放在其他进程中 -->

```
<service
    android:name="cn.jppush.android.service.PushService"
    android:process="mult"
    android:exported="false">
    <intent-filter>
        <action android:name="cn.jppush.android.intent.REGISTER" />
        <action android:name="cn.jppush.android.intent.REPORT" />
        <action android:name="cn.jppush.android.intent.PushService" />
        <action android:name="cn.jppush.android.intent.PUSH_TIME" />
    </intent-filter>
</service>
```

第5处需要修改包名的地方

<!-- since 3.0.9 Required SDK 核心功能-->

```
<provider
    android:authorities="您应用的包名.DataProvider"
    android:name="cn.jppush.android.service.DataProvider"
    android:exported="false"
/>
```

<!-- 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。 -->

<!-- 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 -->

```
<service
    android:name="cn.jppush.android.service.DaemonService"
    android:enabled="true"
    android:exported="true">
    <intent-filter>
        <action android:name="cn.jppush.android.intent.DaemonService" />
        <category android:name="您应用的包名" />
    </intent-filter>
</service>
```

第6处需要修改包名的地方

<!-- since 3.1.0 Required SDK 核心功能-->

```
<provider
    android:authorities="您应用的包名.DownloadProvider"
    android:name="cn.jppush.android.service.DownloadProvider"
    android:exported="true"
/>
```

第7处需要修改包名的地方

<!-- Required SDK核心功能-->

```
<receiver
    android:name="cn.jppush.android.service.PushReceiver"
    android:enabled="true"
    android:exported="false">
    <intent-filter android:priority="1000">
```



```

<!-- Required SDK核心功能-->
<receiver
    android:name="cn.jpush.android.service.PushReceiver"
    android:enabled="true"
    android:exported="false">
    <intent-filter android:priority="1000">
        <action android:name="cn.jpush.android.intent.NOTIFICATION_RECEIVED_PROXY" /> <!--Required 显示通知栏-->
        <category android:name="您的应用的包名" />
    </intent-filter>
    <intent-filter>
        <action android:name="android.intent.action.USER_PRESENT" />
        <action android:name="android.net.conn.CONNECTIVITY_CHANGE" />
    </intent-filter>
    <!-- Optional -->
    <intent-filter>
        <action android:name="android.intent.action.PACKAGE_ADDED" />
        <action android:name="android.intent.action.PACKAGE_REMOVED" />
    </intent-filter>
    <data android:scheme="package" />
</receiver>

<!-- Required SDK核心功能-->
<receiver android:name="cn.jpush.android.service.AlarmReceiver" android:exported="false"/>

<!-- User defined. For test only 用户自定义的广播接收器-->
<receiver
    android:name="org.cocos2dx.lua.MyReceiver"
    android:exported="false"
    android:enabled="true">
    <intent-filter>
        <action android:name="cn.jpush.android.intent.REGISTRATION" /> <!--Required 用户注册SDK的intent-->
        <action android:name="cn.jpush.android.intent.MESSAGE_RECEIVED" /> <!--Required 用户接收SDK消息的intent-->
        <action android:name="cn.jpush.android.intent.NOTIFICATION_RECEIVED" /> <!--Required 用户接收SDK通知栏信息的intent-->
        <action android:name="cn.jpush.android.intent.NOTIFICATION_OPENED" /> <!--Required 用户打开自定义通知栏的intent-->
        <action android:name="cn.jpush.android.intent.CONNECTION" /> <!-- 接收网络变化 连接/断开 since 1.6.3 -->
        <category android:name="您的应用的包名" />
    </intent-filter>
</receiver>

<!-- Required . Enable it you can get statistics data with channel -->
<meta-data android:name="JPUSH_CHANNEL" android:value="developer-default"/>
<meta-data android:name="JPUSH_APPKEY" android:value="您的应用的Appkey" /> <!-- </>值来自开发者平台取得的AppKey-->

</application>
</manifest>

```

这些是极光配置，除了修改包名外其他地方不需要修改

第8处需要修改包名的地方

第9处需要修改包名的地方

极光key配置

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接入方的 Application 的 onCreate 方法里初始化 NewMasterSDK.init(arg0,arg1,arg2)方法, 并且 agr0 的参数是否填入接入方模板入口的 activity, 并且命名正确。
3. 使用 Android Studio 打包的 apk,启动一直进模板, 检查 build.gradle 里配置的包名和版本号与 AndroidManifest.xml 里的包名和版本号是否一致, 否则造成的审核不通过而进入模板
4. 启动闪退
 - (1) 请确认在 AndroidManifest.xml 里的 Application 标签下的 android:name 填入正确的 Application
 - (2) 检查资源拷贝是否完整, 请参照本文档。
5. 启动黑屏, 请严格按照此文档接入 SDK,一般 SDK 初始化未成功会造成黑屏。
 - (1) 一定要写自己的 application 类, AndroidManifest.xml 里的 Application 标签下的 android:name 配置, SDK 的初始化需要写在此 application 的 onCreate 里, SDK 没有初始化或初始化失败会造成黑屏。
 - (2) 过审失败, 需要进入模板界面, 如果 SDK 初始化没有传入模板 activity 也会造成无法找到模板入口而进入黑屏。
6. 界面提示无法连接到网络, 请检查 SDK 初始化接口 NewMasterSDK.init(arg0,arg1,arg2) 函数里的 arg1 参数是否准确, 一定要完整的 url 地址.(例 : http://abc.test1.com:9991)
7. 极光推送问题
 - (1) 无法接收到推送消息
 - a. 请完全使用 SDK 方提供的 AndroidManifest.xml 里的配置 并准确配置你的极光 key, 在 AndroidManifest.xml 里有 9 处需要替换接入方包名的地方, 请确保准确

- b. **SDK 方已经接入了极光推送**，接入方只需要在 AndroidManifest.xml 里配置即可
如果重复接入可能造成无法收到极光推送的问题

(2) 极光推送闪退

- a. 检查 AndroidManifest.xml 里的 Application 标签下的 **android:icon** 是否配置或配置正确

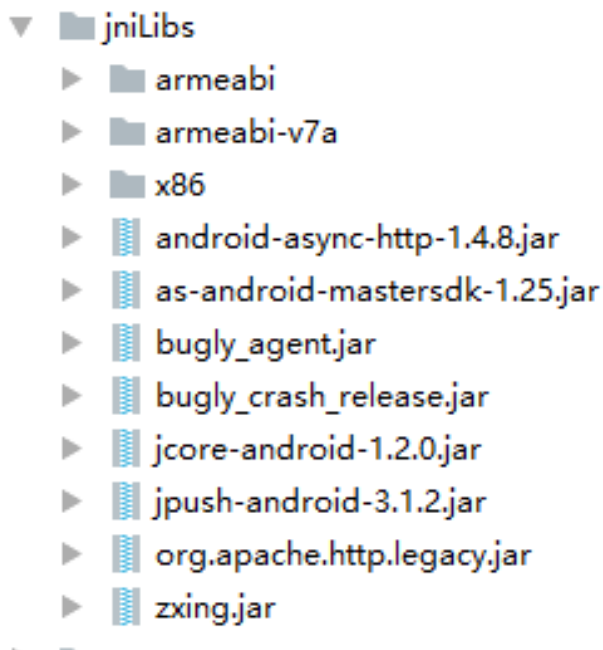
8. 编译错误

Error:Execution failed for task ':app:transformDexArchiveWithExternalLibsDexMergerForDebug'.
> java.lang.RuntimeException: com.android.builder.dexing.DexArchiveMergerException: Unable to merge dex

该编译错误是由于 grade 的依赖包的问题，查看依赖包中是否与 SDK 重复.

检查项目中是否存在重复包名

SDK 中包含所依赖的 jar 列表如图所示：



9.如果接入其他 SDK,请只保留 armeabi、armeabi-v7a、x86 三个 ABI 文件

目录,否则, debug 模式会弹出以下提示:

Error

工程内含有未被SDK支持的ABI架构文件夹,请检查.初始化失败,cocos2dlua.so不存在,请检查工程引入so文件是否正确.

退出