

集成

1 准备环境

在开始集成 ZEGO Express SDK 前，请确保开发环境满足以下要求：

- Android Studio 2.1 或以上版本。
- Android SDK 25、Android SDK Build-Tools 25.0.2、Android SDK Platform-Tools 25.x.x 或以上版本。
- Android 4.1 或以上版本，且支持音视频的 Android 设备或模拟器（推荐使用真机）。
- Android 设备已经连接到 Internet。

2 集成 SDK

2.1 （可选）新建项目

► 此步骤以如何创建新项目为例，如果是集成到已有项目，可忽略此步。

2.2 导入 SDK

目前支持的平台架构包括：armeabi-v7a、arm64-v8a、x86、x86_64。

开发者可通过以下任意一种方式实现集成 SDK。

方式一：使用 JCenter 自动集成 SDK

1. 进入 “app” 目录，打开 “build.gradle” 文件，在 “dependencies” 中添加 `implementation 'im.zego:express-video:x.y.z'`。 “x.y.z” 为 SDK 的版本号，请参考 [下载 SDK 包](#) 中的发布历史获取。

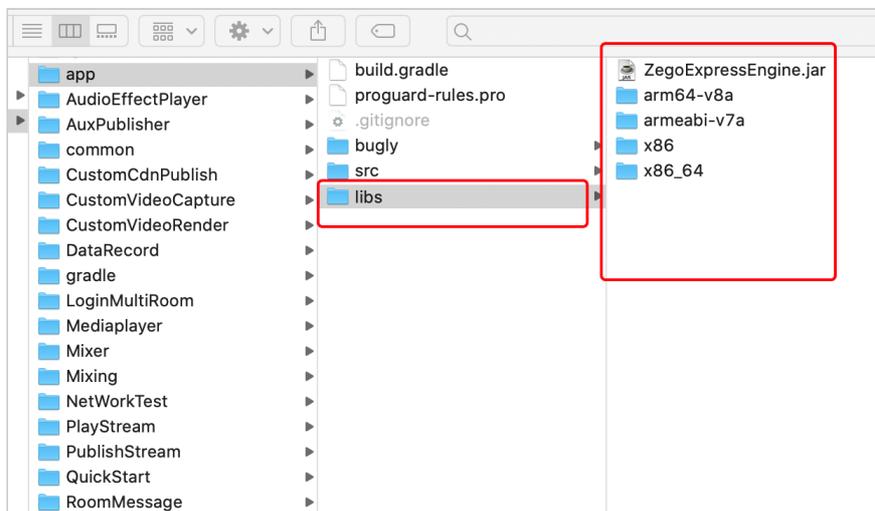
若需要使用带白板功能的 Express-Video SDK，则使用以下命令：`implementation 'im.zego:express-video-whiteboard:x.y.z'`。

从 1.11.0 版本开始，依赖的命令从 `implementation 'im.zego:express-engine-video:x.y.z'` 改为 `implementation 'im.zego:express-video:x.y.z'`。使用 1.11.0 以下的版本不受影响，但后续不再从“express-engine-video”里更新，建议所有使用旧版本的开发者切换到 1.11.0 或以上 的版本进行集成。

```
...
dependencies {
    ...
    // x.y.z 请填写具体版本号，如：1.11.0
    // 可通过 SDK 发布历史取得最新版本号
    implementation 'im.zego:express-video:x.y.z'
}
```

方式二：复制 SDK 文件手动集成

1. 请参考 [下载 SDK 包](#)，下载最新版本的 SDK。
2. 解压 SDK 至项目目录，如“app/libs”。



3. 添加 SDK 引用，进入到“app”目录，打开“build.gradle”文件。
 - 在“defaultConfig”节点添加“ndk”节点，指定支持的平台类型。

```
android {
    compileSdkVersion 30
    buildToolsVersion "30.0.2"

    defaultConfig {
        applicationId "im.zego.express.example.video"
        minSdkVersion 15
        targetSdkVersion 30
        versionCode getAppVersionCode()
        versionName "1.17.0"

        ndk {
            //设置支持的so架构
            abiFilters 'arm64-v8a', 'armeabi-v7a', 'x86', 'x86_64'
        }

        multiDexEnabled true
    }
}
```

```
ndk {
    abiFilters 'armeabi-v7a', 'arm64-v8a', 'x86', 'x86_64'
}
```

- 在“android”节点添加“sourceSets”节点，指定“libs”所在目录。

```
android {
    compileSdkVersion 28
    defaultConfig {...}

    sourceSets {
        main {
            jniLibs.srcDirs = ['libs']
        }
    }
}
```

```
sourceSets {
    main {
        jniLibs.srcDirs = ['libs']
    }
}
```

- 在 “dependencies” 节点引入 “libs” 下所有的 jar。

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

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

3 设置权限

根据实际应用需要，设置应用所需权限。

进入 “app/src/main” 目录，打开 “AndroidManifest.xml” 文件，添加权限。

```
<!-- SDK 必须使用的权限 -->  
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />  
<uses-permission android:name="android.permission.RECORD_AUDIO" />  
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />  
<uses-permission android:name="android.permission.CAMERA" />  
<uses-permission android:name="android.permission.BLUETOOTH" />  
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS" />  
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />  
  
<!-- App 需要使用的部分权限 -->  
<uses-permission android:name="android.permission.READ_PHONE_STATE" />  
<uses-permission android:name="android.permission.WAKE_LOCK" />  
  
<uses-feature  
    android:glEsVersion="0x00020000"  
    android:required="true" />  
  
<uses-feature android:name="android.hardware.camera" />  
<uses-feature android:name="android.hardware.camera.autofocus" />
```

因为 Android 6.0 在一些比较重要的权限上要求必须申请动态权限，不能只通过 “AndroidManifest.xml” 文件申请静态权限。因此还需要参考执行如下代码，其中 “requestPermissions” 是 “Activity” 的方法。

```
String[] permissionNeeded = {
    "android.permission.CAMERA",
    "android.permission.RECORD_AUDIO"};

if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
    if (ContextCompat.checkSelfPermission(this, "android.permission.CAMERA") != PackageManager.PERMISSION_GRANTED ||
        ContextCompat.checkSelfPermission(this, "android.permission.RECORD_AUDIO") != PackageManager.PERMISSION_GRANTED) {
        requestPermissions(permissionNeeded, 101);
    }
}
```

4 防止混淆代码

在“proguard-rules.pro”文件中，为 SDK 添加 `-keep` 类的配置，防止混淆 SDK 公共类名称。

```
-keep class **.zego.**{*};
```