

Bugly NDK集成说明文档（重要）

从2.0.5版本开始，Bugly的SO库（libbugly.so）包含以下架构：

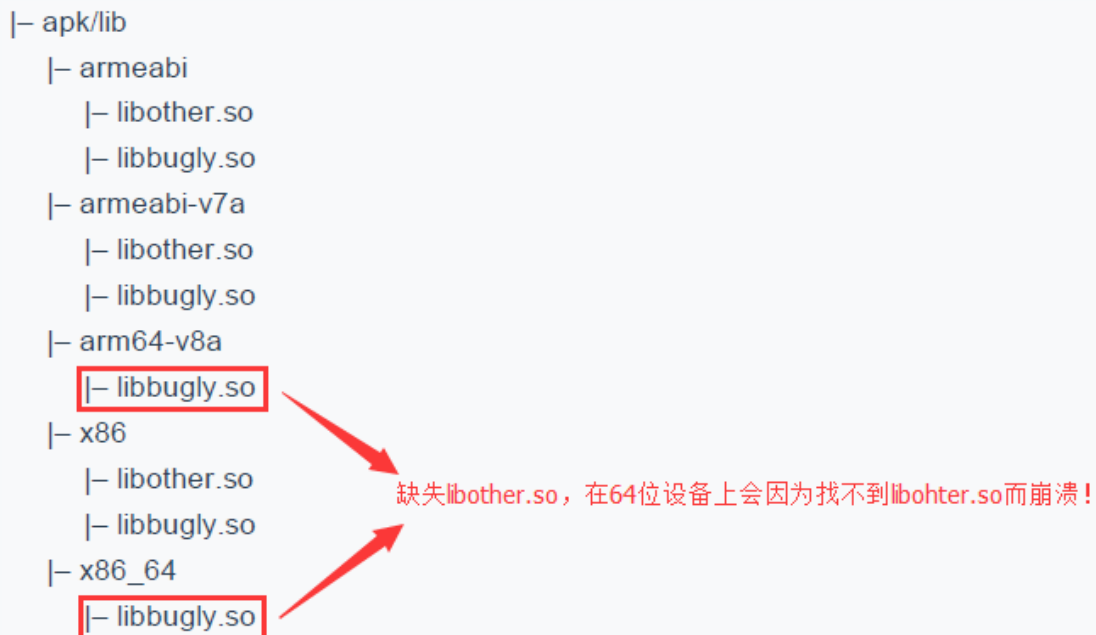
- armeabi
- armeabi-v7a
- arm64-v8a
- x86
- x86_64

除了Bugly SO库之外，如果还集成了**其他第三方SO库**，则一定要注意兼容性问题：

- 如果libs架构数与上述架构不一致时，可能会发生“[java.lang.UnsatisfiedLinkError](#)”异常！

例如，第三方SO库缺失了64位架构的库（缺失了“**arm64-v8a**”和“**x86_64**”），则编译出来的APK包中的lib目录结构如下：

```
|- apk/lib
  |- armeabi
    |- libother.so
    |- libbugly.so
  |- armeabi-v7a
    |- libother.so
    |- libbugly.so
  |- arm64-v8a
    |- libbugly.so
  |- x86
    |- libother.so
    |- libbugly.so
  |- x86_64
    |- libbugly.so
```



缺失libother.so，在64位设备上会因为找不到libother.so而崩溃！

因此，只需要保留支持的架构的最小子集！

例如，第三方SO库只包含“armeabi”和“armeabi-v7a”架构，而Bugly SO库包含了“armeabi”、“armeabi-v7a”、“arm64-v8a”、“x86”、“x86_64”，那么最小子集就是二者的交集，即“**armeabi**”和“**armeabi-v7a**”架构。此时需要把其他架构的Bugly SO库去掉。

Eclipse集成

- 复制Bugly NDK包的libs目录下**所需架构**到工程的libs目录中。

Android Studio集成

- 复制Bugly NDK包的libs目录到工程的libs目录中；
- 在Module的build.gradle文件中添加NDK的abiFilter配置：

```
android {  
    defaultConfig {  
        ndk {  
            //设置支持的SO库架构  
            abiFilters 'armeabi' //, 'x86', 'armeabi-v7a', 'x86_64', 'arm64-v8a'  
        }  
    }  
}
```

如果在添加“**abiFilter**”之后Android Studio出现以下提示：

NDK integration is deprecated in the current plugin. Consider trying the new experimental plugin.

则在项目根目录的**gradle.properties**文件中添加：

```
android.useDeprecatedNdk=true
```