

常见问题

1、应用启动数据未采集或缺失生命周期方法。

1.若 AndroidManifest.xml 文件中注册了自定义的 MyBaseActivity，请删除它，或对其配置过滤嵌码。过滤需要在 tingyun.properties 文件中添加以下内容：

```
#等号“=”后替换为需要过滤听云嵌码的类的路径，如有多个路径，使用英文逗号隔开  
noInstrumentedClasses=com/nbs/ui/base/MyBaseActivity
```

2.应用需实现自定义的 Application 类

2、用户操作数据采集少。

听云 SDK 编译时对点击事件（如 onClick、onItemClick、ItemSelected、PageSelected 等方法）嵌码，仅支持点击数据采集，暂不支持触摸、滑动等操作。

3、未采集到崩溃或采集的崩溃数量偏少。

1.听云 SDK 自 2.11.1 版本开始支持 Native Crash 异常采集，之前的版本仅采集 Java Crash。2.听云 SDK 初始化前的崩溃无法采集。3.检查报表的崩溃采集开关是否开启。4.若应用中存在应用自身异常捕获或第三方崩溃采集逻辑，需放在听云 SDK 之前初始化。

4、TCP、首包数据为0。

1.听云 SDK 需在应用自身网络请求发起之前初始化 2.暂不支持安卓8.1以上 TCP 数据采集。3.使用 Http 代理时无法采集 TCP 数据。4.WebView 跨域资源响应头未配置 Timing-Allow-Origin: *

5、WebView数据未采集或只采集到了网络错误、HTTP错误。

1.未在 WebChromeClient 的 onProgressChanged 方法中调用 NBSWebChromeClient.initJSMonitor(view, newProgress); 2.不支持 SPA 单页面应用数据采集。3.不支持使用 loadDataWithBaseURL 加载 html 代码的数据采集。

6、URL过滤、错误过滤未生效。

检查报表上配置是否正确，可以通过在线正则工具检查，需注意报表配置正则的前后需要加斜杠“/”。

7、如项目使用MultiDex分包，安卓5.0以下设备可能会出现 java.lang.NoClassDefFoundError异常。

需将报错的类分配至主 dex 中，可参考 Google 提供的解决方案<https://developer.android.com/studio/build/multi-dex.html?hl=zh-cn>

8、Sophix补丁工具生成补丁失败。

听云 SDK 为了采集应用启动数据，会在编译时对 Application 类的 attachBaseContext() 及 onCreate() 方法嵌码。而阿里 Hotfix 补丁工具自 3.2.1 版本开始，要求在 initialize 之前，不能调用非 Android 官方 API，导致生成补丁失败。

【解决方案】

为了通过补丁工具的检查，听云不能在编译时 Application 类嵌码。可以在听云 tingyun.properties 配置文件中对 Application 类过滤。这样会影响听云采集应用启动数据，故需要对 Application 类手动嵌码。

- 若项目中使用了 SophixStubApplication 类，请按以下方法配置。
 - 1、在 tingyun.properties 配置文件(没有则新建，在项目根目录和主 module 下各一份)中，配置对 Application 类过滤嵌码：

```
noInstrumentedClasses=com/nbs/app/SophixStubApplication
```

- 2、在项目的 RealApplication 类的 attachBaseContext() 及 onCreate() 方法的首行、末行中分别添加以下代码：

```
NBSAppInstrumentation.attachBaseContextBeginIns(base);
NBSAppInstrumentation.attachBaseContextEndIns();
NBSAppInstrumentation.applicationCreateBeginIns();
NBSAppInstrumentation.applicationCreateEndIns();
```

示例如下：

```
@Override
protected void attachBaseContext(Context base) {

com.networkbench.agent.impl.instrumentation.NBSAppInstrumentation.attachBaseContextBeginIns(base);
    super.attachBaseContext(base);

com.networkbench.agent.impl.instrumentation.NBSAppInstrumentation.attachBaseContextEndIns();
}

@Override
public void onCreate() {

com.networkbench.agent.impl.instrumentation.NBSAppInstrumentation.applicationCreateBeginIns();
    super.onCreate();
    //...其他页面代码

com.networkbench.agent.impl.instrumentation.NBSAppInstrumentation.applicationCreateEndIns();
}
```

- 若项目中未使用 SophixStubApplication 类，请按以下方法配置。

1、在 `tingyun.properties` 文件（没有则新建，在项目根目录和主module下各一份）中，配置对 `Application` 类过滤嵌码，其中 `"#="` 后为清单文件中配置的 `application` 类的全路径：

```
noInstrumentedClasses=com/nbs/app/MyApp
```

2、`Sophix` 需在 `attachBaseContext()` 方法中初始化，`Sophix` 初始化后，调用以下代码：

```
//在 Application 的 attachBaseContext() Sophix 初始化后调用以下方法
NBSAppInstrumentation.attachBaseContextBeginIns(base);
NBSAppInstrumentation.attachBaseContextEndIns();
//在 Application 的 onCreate() 方法的首行、末行分别调用以下方法
NBSAppInstrumentation.applicationCreateBeginIns();
NBSAppInstrumentation.applicationCreateEndIns();
```

示例如下：

```
@Override
protected void attachBaseContext(Context base) {
    super.attachBaseContext(base);
    //如果需要使用 Multidex，需在此处调用
    Multidex.install(this);
    initSophix();//初始化 Sophix

    com.networkbench.agent.impl.instrumentation.NBSAppInstrumentation.attachBaseContextBeginIns(base);

    com.networkbench.agent.impl.instrumentation.NBSAppInstrumentation.attachBaseContextEndIns();
}

@Override
public void onCreate() {

    com.networkbench.agent.impl.instrumentation.NBSAppInstrumentation.applicationCreateBeginIns();
    super.onCreate();
    //...其他页面代码

    com.networkbench.agent.impl.instrumentation.NBSAppInstrumentation.applicationCreateEndIns();
}
```

【参考资料】

Hotfix 补丁工具报错排查步骤请参见阿里云的介绍：

https://help.aliyun.com/knowledge_detail/59422.html?spm=a2c4g.11186623.4.3.NocMNz

9、编译时出现 `java.util.zip.ZipEexeption` 异常。

SDK jar 包重复引用，检查 libs 目录及 build.gradle 文件，确保只有一份 nbs.newlens.agent.jar 引用。

10、项目使用Lambda时编译失败。

将听云插件放在 lambda 后面

```
apply plugin: 'me.tatarka.retrolambda'
apply plugin: 'newlens'
```

11、SDK编译耗时问题。

听云SDK的原理是字节码插桩（使用 Transform API），在一定程度上会影响项目的编译耗时。我们提供了一个在开发期间听云 SDK 不参与编译的配置。若您的项目较大，在集成听云SDK后感觉到明显的编译时间增长，可按以下方法配置：

(1) 打开 app 目录下的 build.gradle 文件 (2) 添加以下内容

```
newlensExt {
    newlensEnabled = false //设为false时不对项目嵌码，提交测试及上线前需设置为true
}
```

12、同一设备在报表出现多个did

听云 SDK 自 2.14.2 版本不再采集 IMEI 等设备信息，在应用首次启动时由服务器下发 did，并在本地缓存。若测试时应用卸载重新安装，或对应用进行“清除数据”操作，都会使清除本地缓存的 did，再次启动应用会重新下发新的 did。