北京基调网络股份有限公司

移动 App 监测 SDK 部署说明书

Android SDK Eclipse 部署文档

Meng'yang Cheng

2016/5/5

目录

听云 APP 部署文档 2

1、设置应用名称 2

2、使用 ECLIPSE 构建 2

3、嵌码验证 16

4、性能数据采集 16

5、CRASH 反混淆 16

5.1、查找 Mapping File 文件 16

6、配置 TINGYUN.PROPERTIES 文件 18

6.1、新建配置文件 18

6.2、配置 Mapping File 自动上传 18

6.3、配置 OkHttp 控制开关 19

6.4、配置 WebView 控制开关 19

7、配置用户自定义附加信息 20

附录 20

听云 APP 支持协议类库 20

听云 TCP 支持设备版本 20

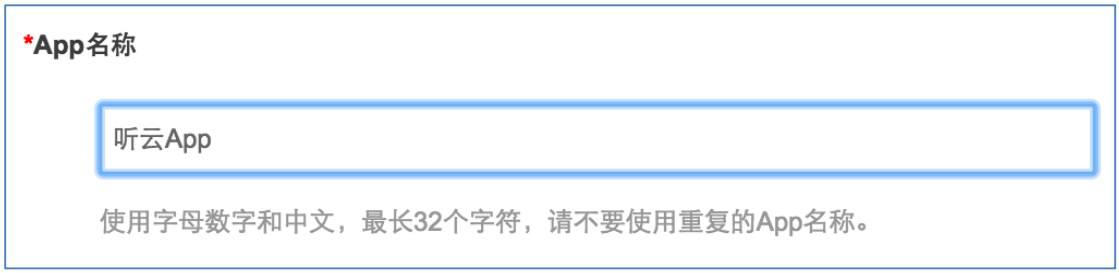
听云 App 部署文档

1、设置应用名称

请为需要监测的 App 设置一个易于识别的应用名称，该 App 名称将出现在 App 列表和相关的报表中来代表对应的 App。

App 名称可以使用字母数字和中文，最长 32 个字符。设置 App 名称时，尽量不要设置重复的 App 名称，以免给自己造成不必要的麻烦。

您的 App 名称可以在设置选项中随时修改



2、使用 Eclipse 构建

听云 App SDK Android 版本支持在 Eclipse 集成开发环境中直接使用和部署。

（1）查看 Eclipse 版本号

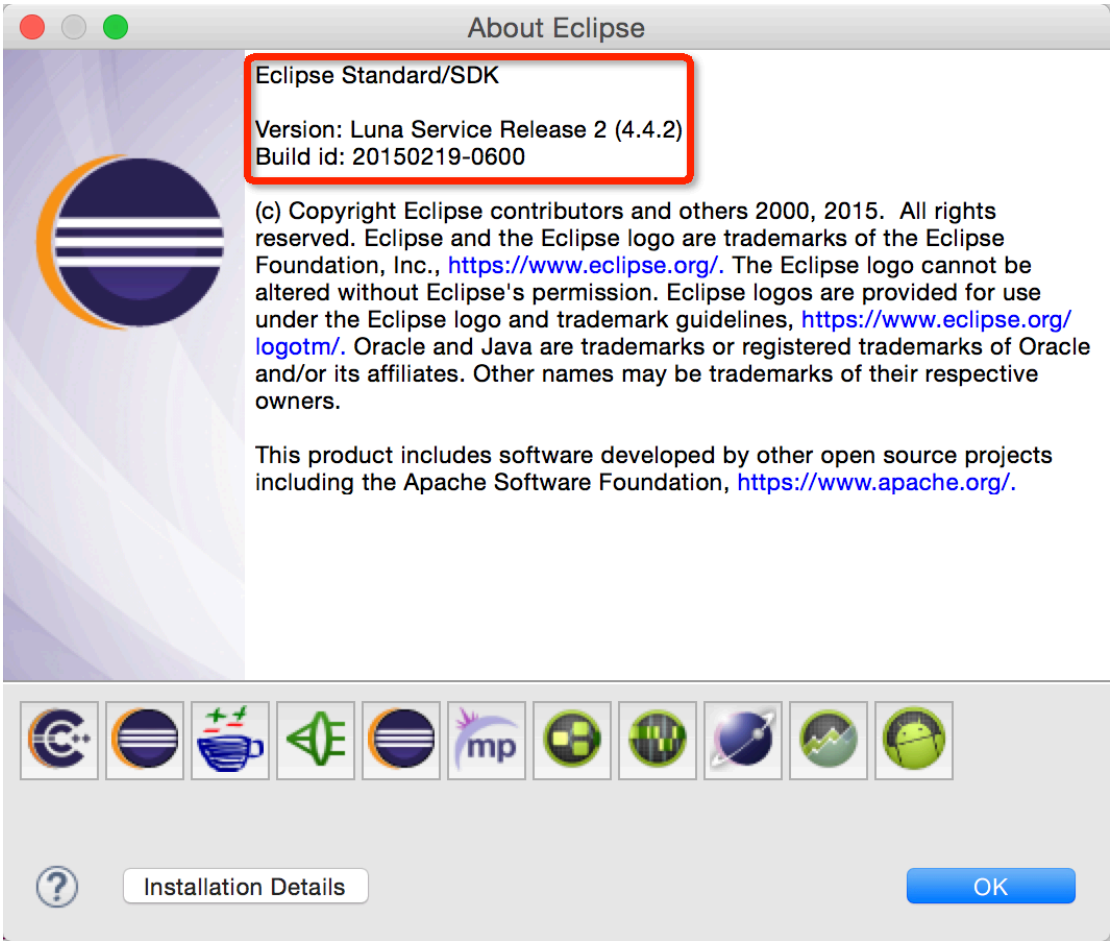
听云 App Eclipse Plugin 支持 Eclipse 3.8~4.4 版本，请于安装前确认您使用的 Eclipse 版本号

Mac OS 下

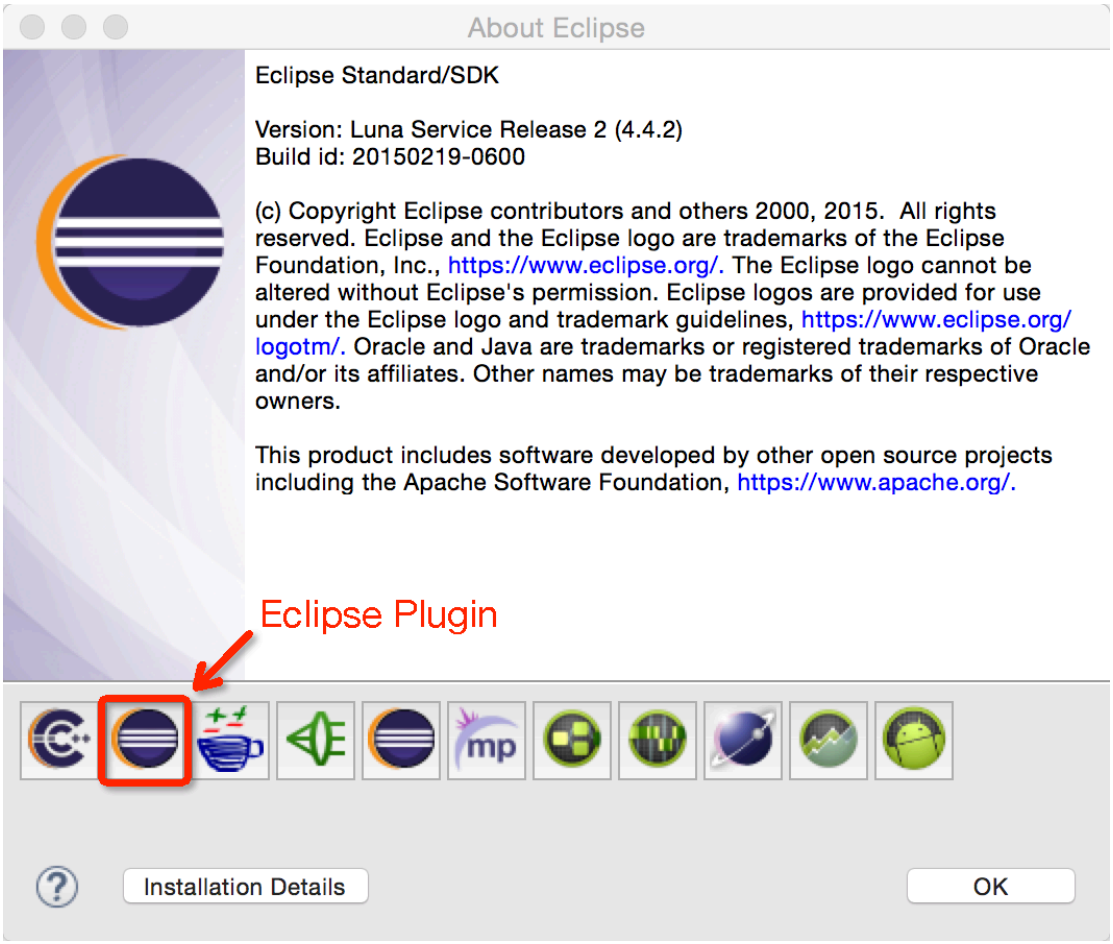
点击进入“关于 Eclipse”

Windows 下

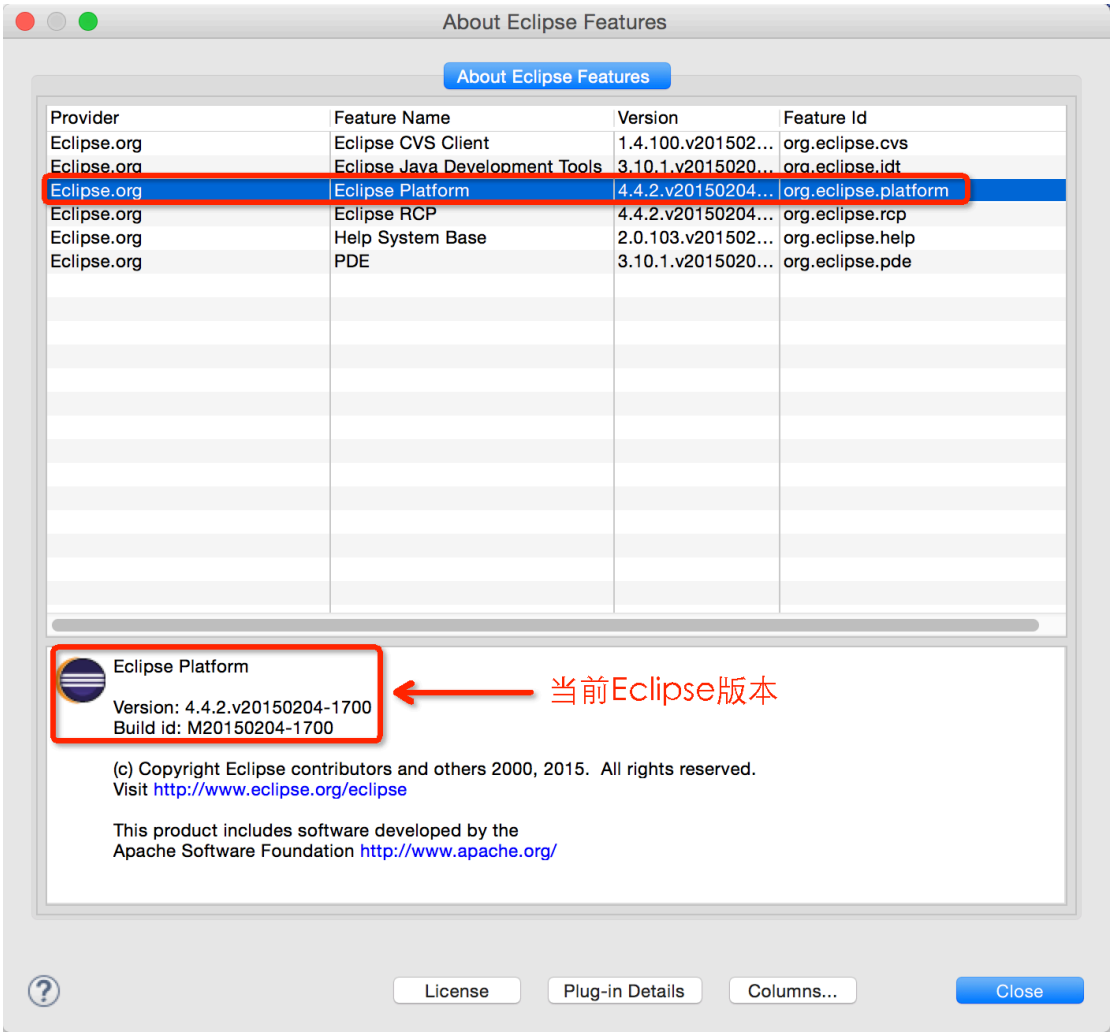
点击进入“about Eclipse”



i. 打开“关于 Eclipse”可查看当前版本号

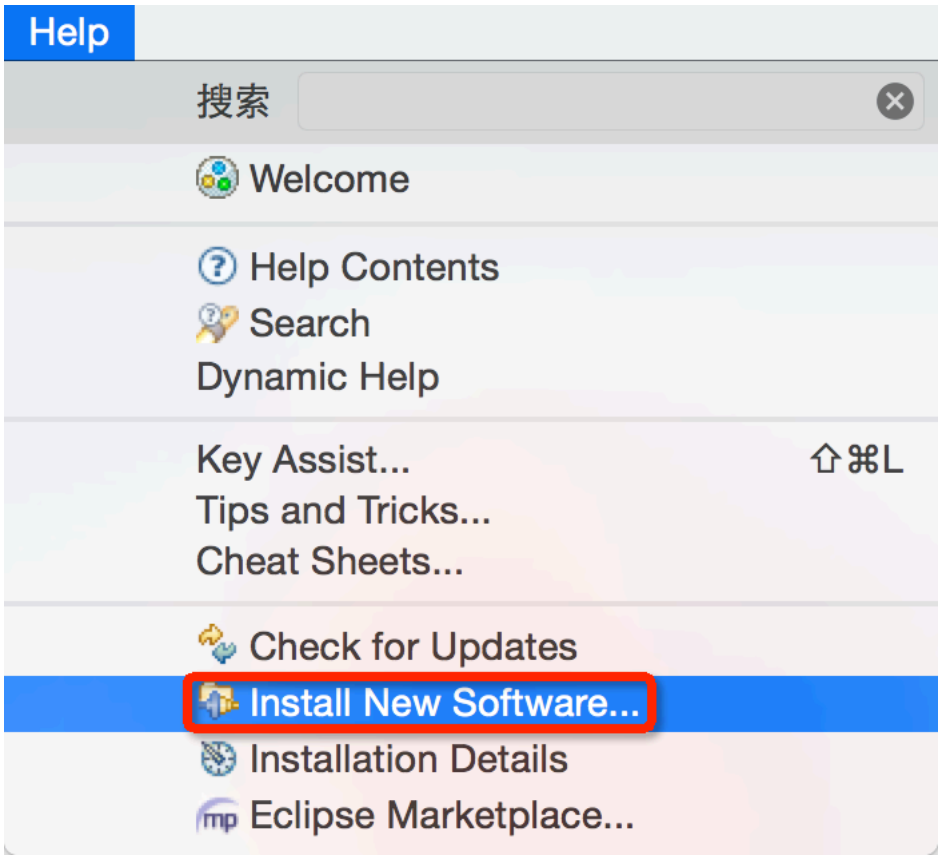


ii. 如未见详细版本号可点击进入“Eclipse Plugin”查看

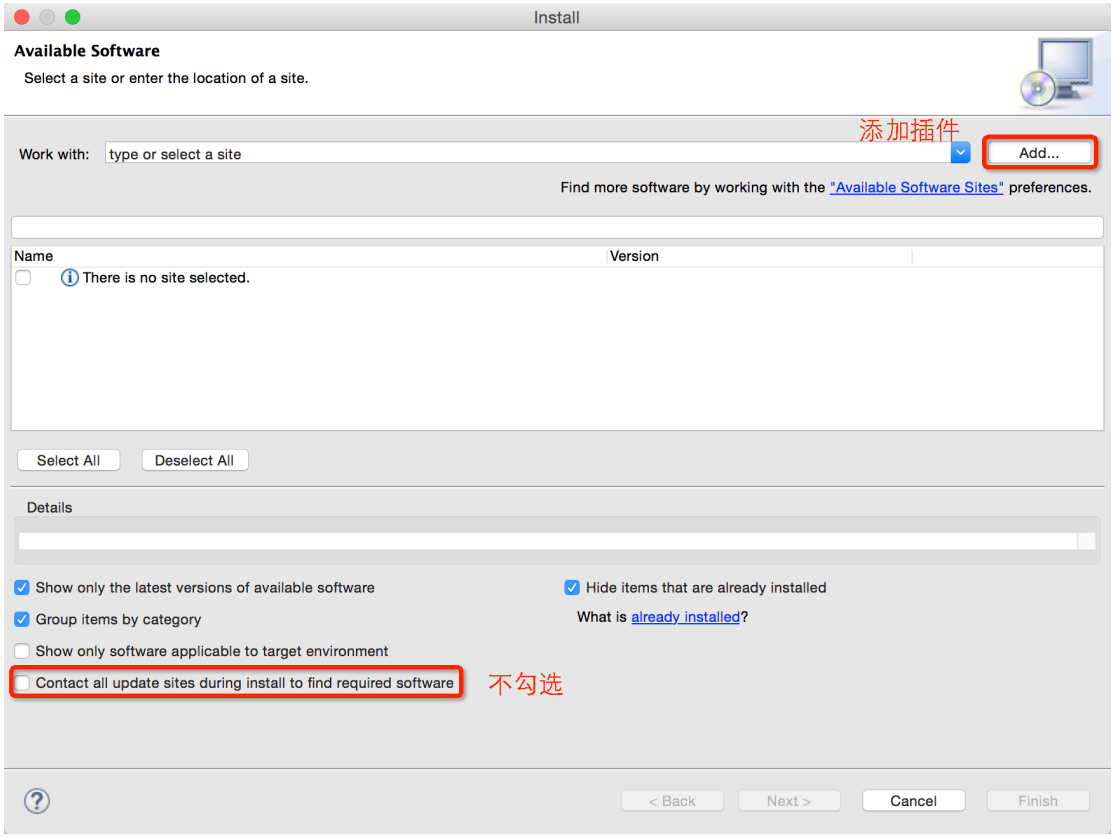


iii. 当前版本号为：4.4.2

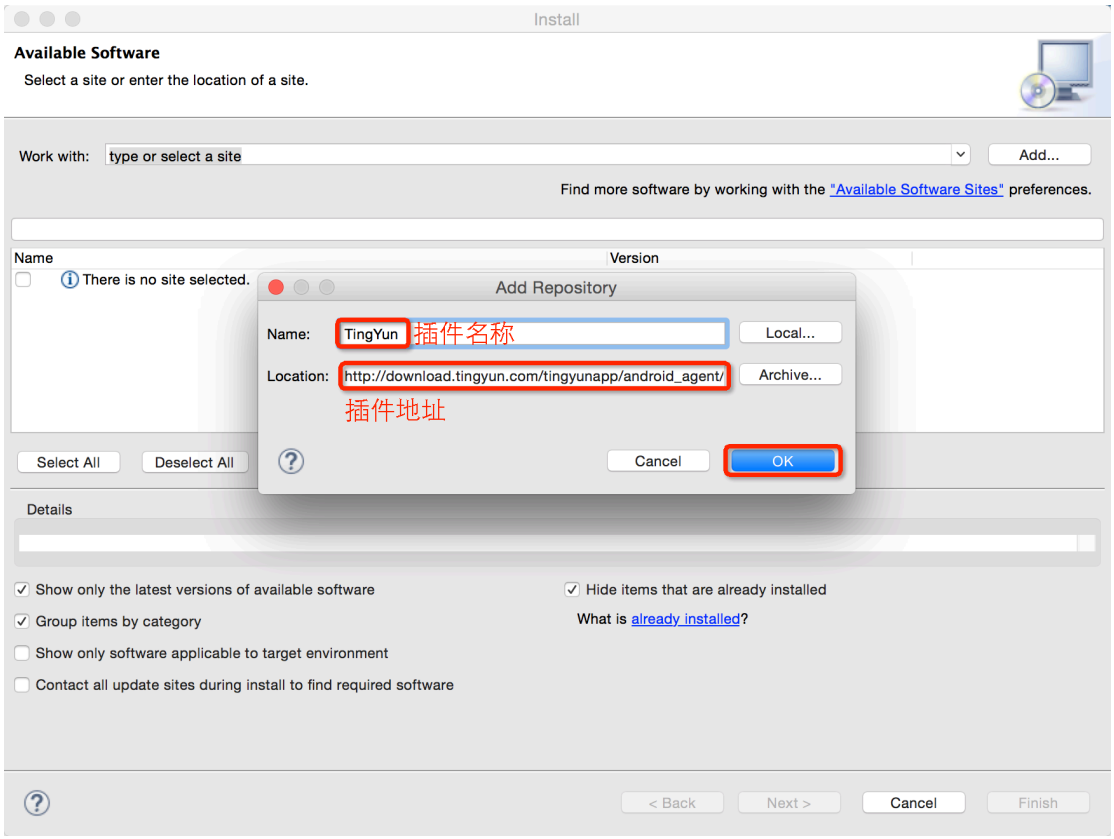
（2）安装 TingYun NewLens Android SDK 的 Eclipse Plugin



i. 在 Eclipse 集成开发环境中点击“Help”菜单，选择“Install New Software...”



ii. 点击 Work with 项右侧的“Add…”按钮来增加 TingYun 的插件资料库。



iii. 设置插件资料库的名称（例如：TingYun 等等），设置资料库的 URL 地址：

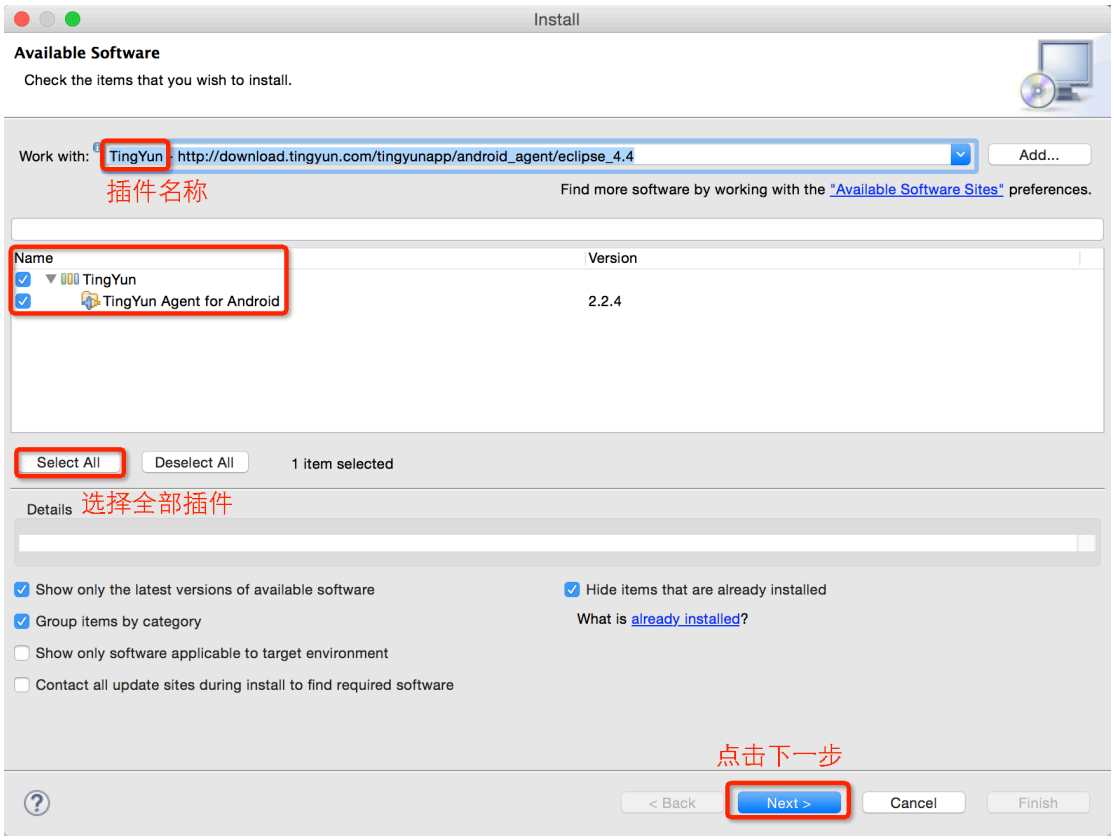
注：使用新版地址前，请先卸载老版本！

Eclipse 4.4 版本资料库 URL 地址为：

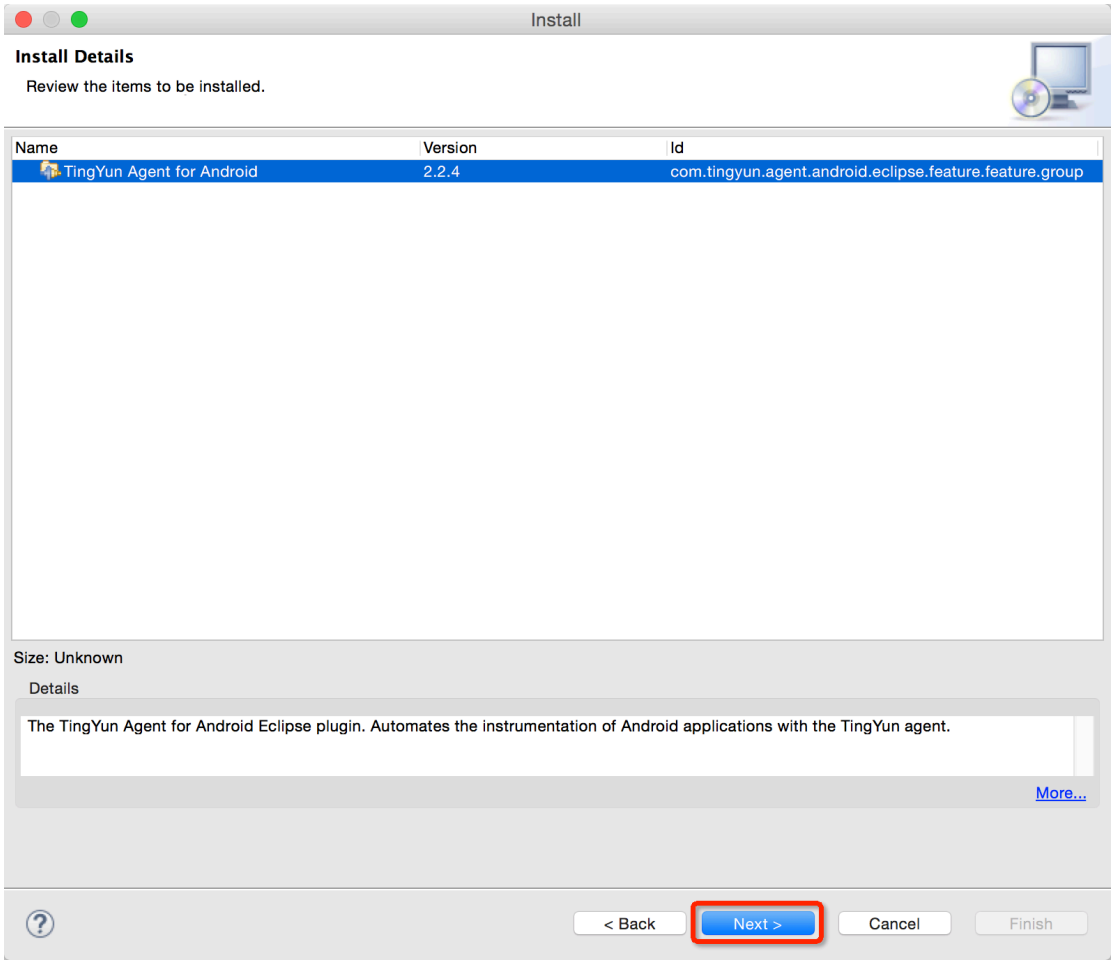
```
http://download.tingyun.com/tingyunapp/android_agent/eclipse_4.4
```

Eclipse 4.4 以下版本资料库 URL 地址为：

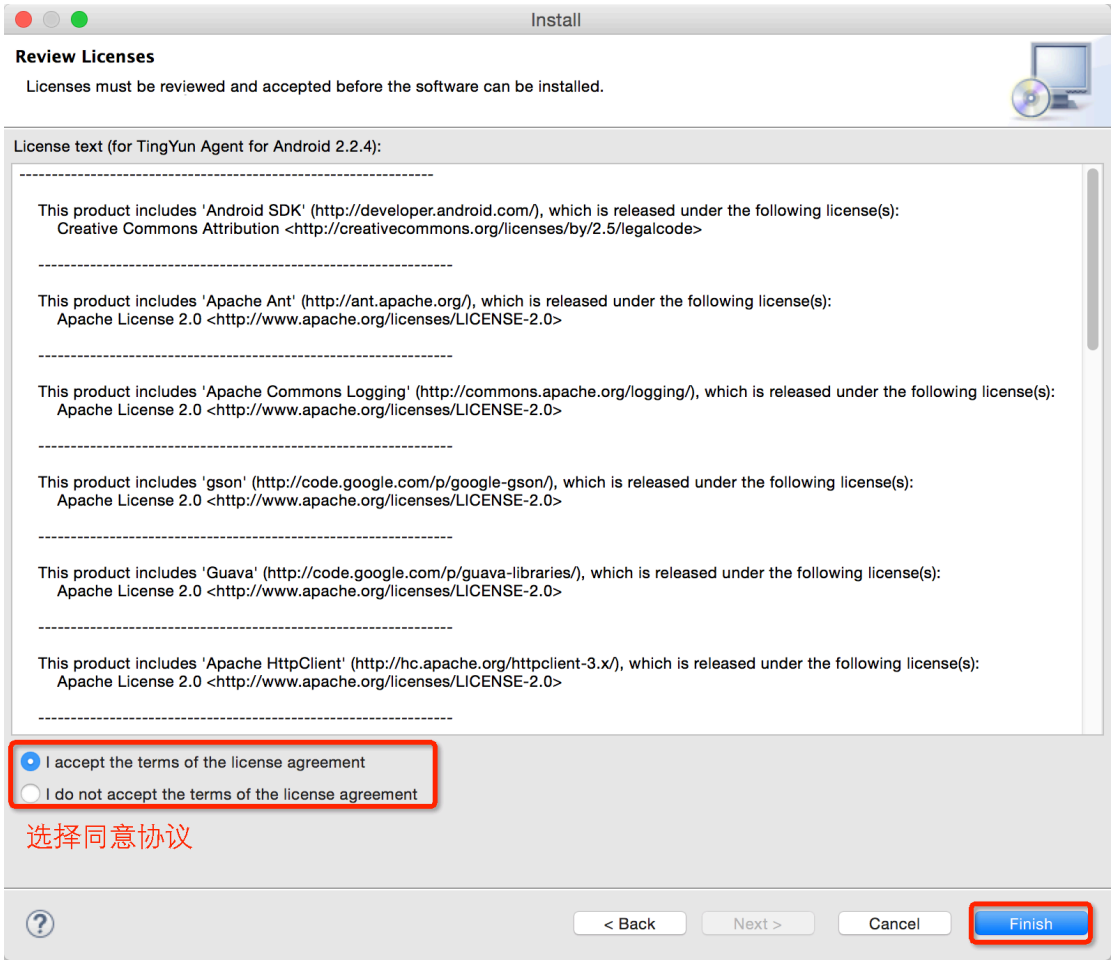
```
http://download.tingyun.com/tingyunapp/android_agent/eclipse
```



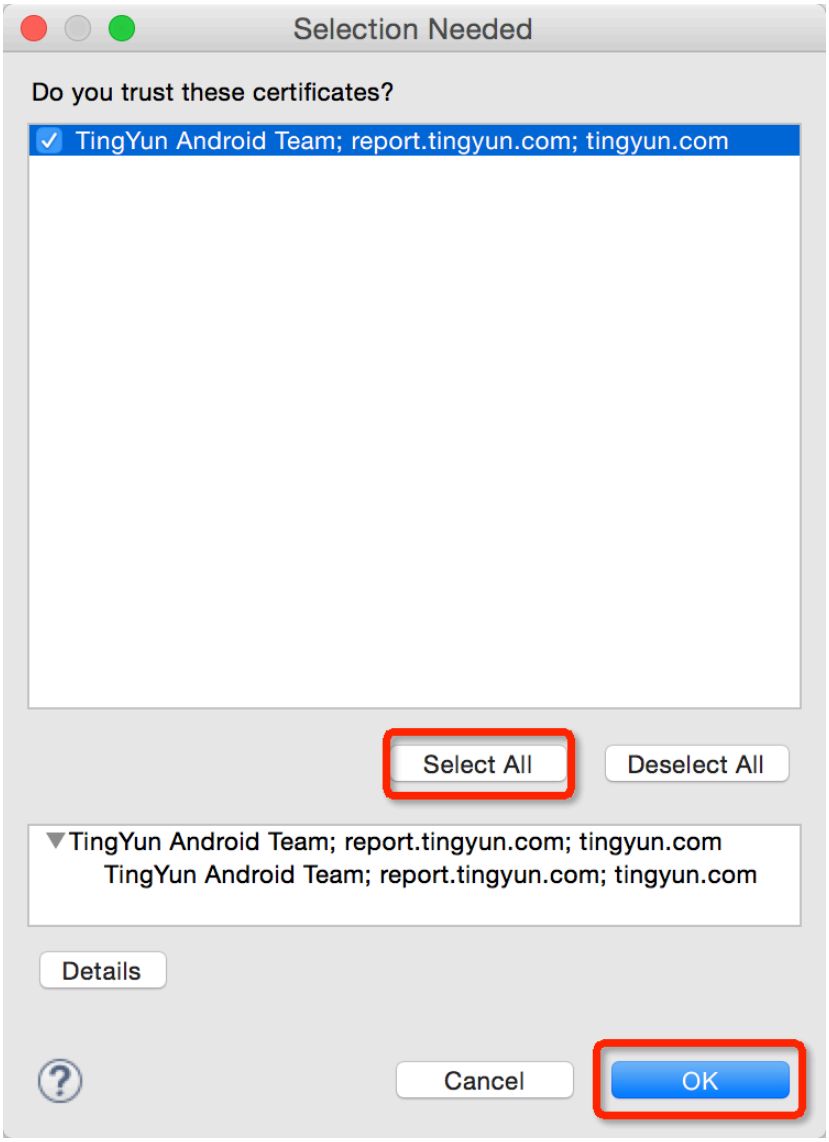
iv. 在 Work with 选项中选中刚才添加的资料库，在下方列表中点击“Select All”选中所有的插件。点击“Next >”到下一步。



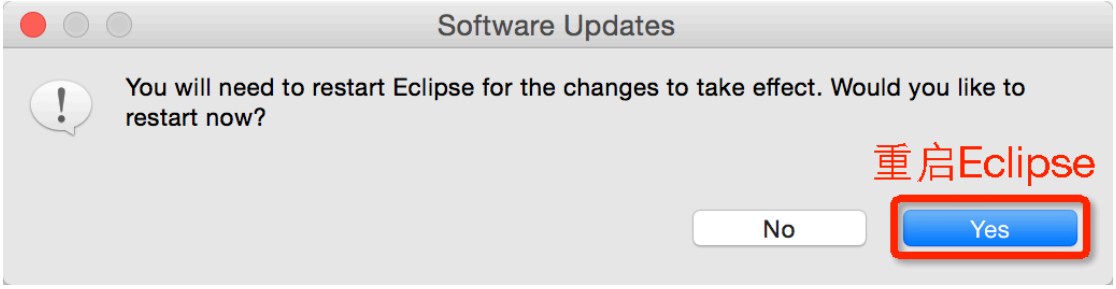
v. 查看插件描述并点击“Next >”进入下一步。



vi. 查看许可协议，选择“I accept the terms of the license agreement”，点击“Finish>”完成操作。

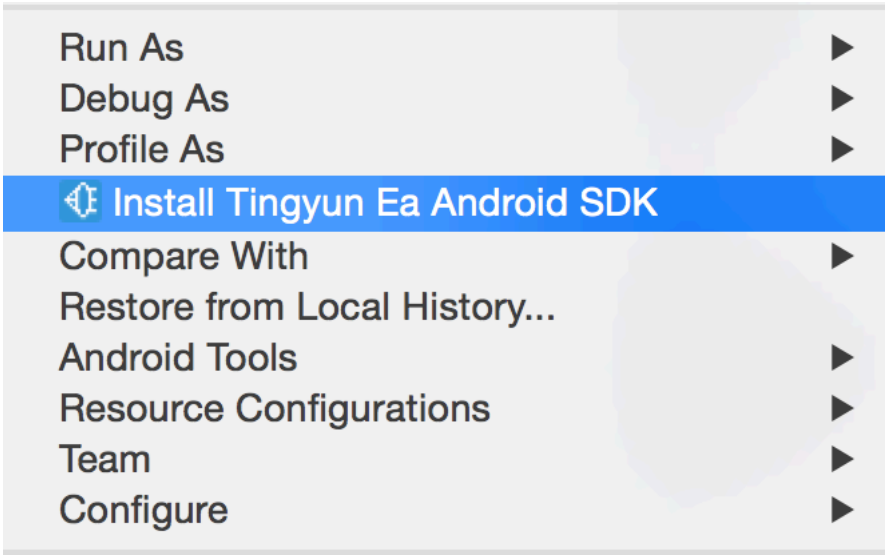


vii. 选择信任插件的签名证书，点击“OK”。

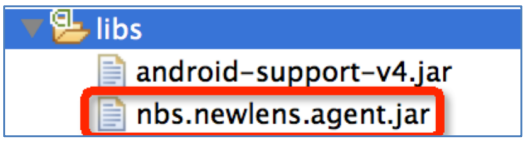


viii. 点击“Yes”重启 Eclipse 来完成插件的安装。

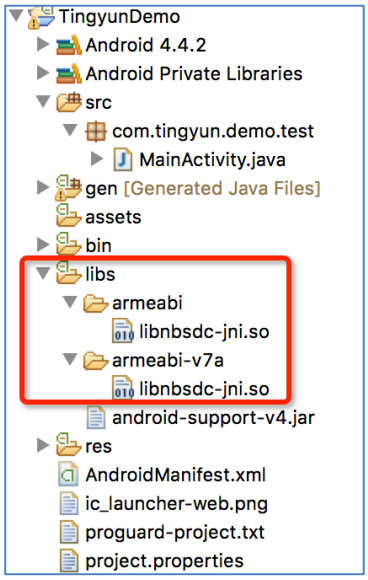
（3）插件安装完成后，右击需要监测的 App，选择“Install Tingyun Ea Android SDK”



Eclipse 会自动添加“nbs.newlens.agent.jar”包到 libs 目录下，若没有 libs 目录请新建一个。



注：若需采集 TCP 性能数据（主机 IP、DNS 时间、TCP 建联时间），请将 SO 文件添加到 libs 目录下的相应文件夹中，如下图所示：



SO 库下载地址为：https://report.tingyun.com/mobile/download/sdk

（4）设置应用程序权限

构建完成后，请在待监测的 App 工程的 AndroidMainfest.xml 文件中增加以下的权限：

```
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.READ_LOGS" />
<uses-permission android:name="android.permission.INTERNET" />
```

（5）插入启动 NBSAgent 的代码

在待监测 App 的主 Activity（Main Activity）源文件中导入听云的 Android 探针类：

```
import com.networkbench.agent.impl.NBSAppAgent;
```

在 onCreate()方法中（如果没有请新增 onCreate()方法）增加以下初始化听云 App 探针的调用：

```
NBSAppAgent.setLicenseKey("AppKey
").withLocationServiceEnabled(true).start(this.getApplicationContext());
```

注：如项目使用了腾讯 X5 框架，可调用 setX5Enable()方法

```
NBSAppAgent.setLicenseKey("AppKey
").withLocationServiceEnabled(true).setX5Enable(true).start(getApplicationContext());
```

若不需要采集地理位置信息（报表中将没有区域维度的数据）请使用以下代码：

```
NBSAppAgent.setLicenseKey("AppKey ").start(this.getApplicationContext());
```

其中“AppKey”从听云 App 系统中生成。

（6）设置混淆器

如果待监测的 App 使用 ProGuard 工具来优化发布的 apk 包，为了保证听云 App 监测 SDK 可以正常使用，请在所属项目的 proguard.cfg 文件中加入以下的内容：

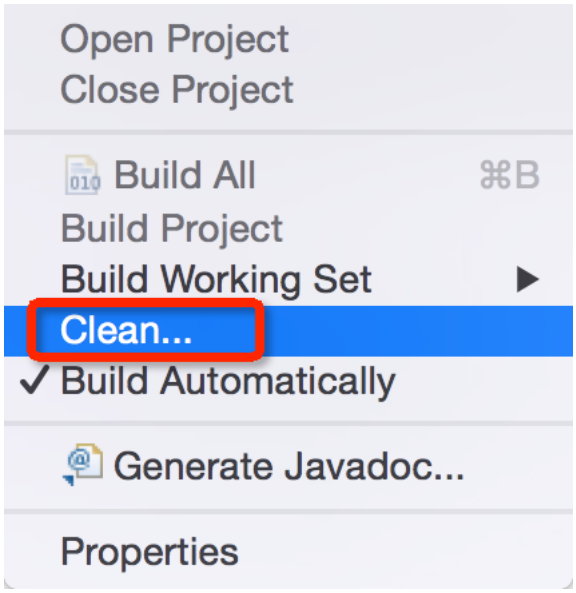
```
# ProGuard configurations for NetworkBench Lens
-keep class com.networkbench.** { *; }
-dontwarn com.networkbench.**
-keepattributes Exceptions, Signature, InnerClasses
# End NetworkBench Lens
```

如果您希望保留行号信息，建议您在 proguard.cfg 中添加如下代码：

```
-keepattributes SourceFile,LineNumberTable
```

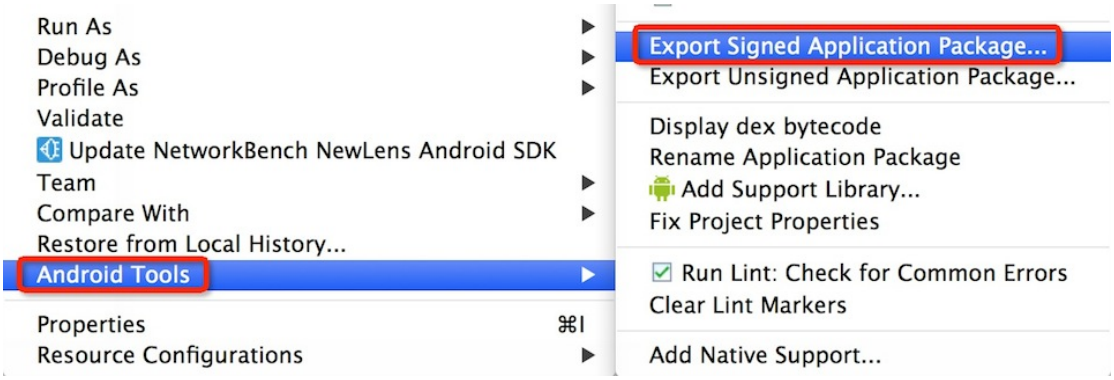
（7）clean 项目工程

建议打包前先 Clean 项目工程，以确保听云 App 配置生效



（8）Eclipse 打包编译

使用“Export Signed Application Package...”进行编译即可



（9）运行 apk

打包编译后，在模拟器或 Android 设备上运行待监测的 App，并进行适当的网络访问。

注意：
1、Eclipse 插件目前只支持 JDK（1.5 - 1.8）运行环境（不支持只有 JRE 的运行环境）
2、Window 安装 Eclipse 插件时，请在没有空格和特殊字符的路径安装 JDK

3、Eclipse 插件需要使用 JAVA_HOME 环境变量，请检查环境变量

3、嵌码验证

运行嵌码完成后的 APK，连接电脑使用 LogCat 过滤 TAG = NBSAgent 查看 log 日志

出现如下 log 表示嵌码成功：

```
NBSAgent start
NBSAgent enabled
NBSAgent V “TingYun_Version”
connect success
```

4、性能数据采集

若以上的步骤操作正确，听云 App Android SDK 探针就已经成功部署到待测 App。保持 App 运行，并访问网络，静待几分钟即可在听云报表看到嵌码 App 的性能数据。

5、Crash 反混淆

为了更好的帮助用户解析崩溃信息,我们提供了反混淆机制来帮助用户快速精准的定位崩溃发生位置。当您使用反混淆机制时,需要您上传当前版 本的 Mapping File 到听云服务器。

目前,我们提供了两种方式上传 Mapping File,您可以选择任意一种方式上传您的 Mapping File 此过程不会干扰正常用户代码和打包编译。

5.1、查找 Mapping File 文件

进入嵌码项目根目录，搜索 Mapping.txt。

5.1.1、报表上传对应版本的 Mapping File



i. 登陆听云报表，在 App 概览中选择嵌对应的 App 名称，进入 App 后选择左侧菜单栏中的“崩溃”按钮：



ii. 在右侧“崩溃历史记录列表”中选择并点击“dSYM / Mapping 文件管理”



iii. 在弹出窗口中选择上传平台类型，如：“Android”。选择对应的应用版本号，并点击右

侧上传按钮，手动上传 Mapping File

6、配置 tingyun.properties 文件

6.1、新建配置文件

进入 Eclipse 安装目录，在 Eclipse.ini 同级目录下新建 tingyun.properties 配置文件

6.2、配置 Mapping File 自动上传

将下列配置项添加到到 tingyun.properties 配置文件：

```
authKey=听云 API 账号授权 Key，由报表系统生成
appKey=听云 App 授权 Key
mapping_file_auto_upload=上传 Mapping 开关
```

1) 配置 AuthKey 授权码



i. 登陆报表，选择用户名称，点击修改授权码



ii. 复制“当前授权码”并黏贴到配置文件下

2) 配置 AppKey



i. 登陆报表，在 App 概览中选择嵌对应的 App 名称，如：Android_SDK。点击左侧配置按钮，选择“修改设置”



ii. 进入“修改配置”后，找到“安装 SDK”选择“App Key”复制到配置文件中

3) 配置 mapping_file_auto_upload 上传开关

如果您需要自动上传 Mapping File，请在配置文件中打开 Mapping File 控制开关：

```
mapping_file_auto_upload=true
```

注意：改开关只有在启用混淆器时才会生效，通过主机上传 Mapping File 到听云服务器，您可以在正式发布或需要时打开 Mapping File 上传开关，如应用未开启混淆器，听云 SDK 将不会启用配置文件

6.3、配置 OkHttp 控制开关

将下列配置项添加到到 tingyun.properties 配置文件：

```
okhttp3.0=true
```

注：听云默认对 OkHttp2.0 嵌码，如果您使用 Okhttp2.0 请忽略该设置，若使用的是 OkHttp1.0 请设置 okhttp2.0=false 即可

6.4、配置 WebView 控制开关

将下列配置项添加到到 tingyun.properties 配置文件：

```
webview=true
```

注：听云 WebView 需使用 setWebViewClient 方法，如未实现该方法，请添加如下代码

```
webview.setWebViewClient(new WebViewClient() {});
```

7、配置用户自定义附加信息

听云 App 提供用了户自定义附加信息接口，用户可以自定义配置附加信息。当应用发生 Crash 时上传到报表服务器。用户可以再初始化之后任意位置配置该接口。UserCrashMessage 最多可添加 10 条附加信息，每条附加信息最大支持 100 个字节。

用户自定义附加接口如下：

```
NBSAppAgent.setUserCrashMessage(String key,String value)
```

附录

听云 App 支持协议类库

- HttpURLConnection
- Android HttpClient 4.0
- Apache HttpClient (> 4.0)
- Volley+OkHttpClient
- OkHttp 2.0 (> 2.1)
- OkHttp 1.0
- Retrofit (> 1.9)

听云 TCP 支持设备版本

- 目前只支持 Android 4.0