特效插件开发 for android

描述开发和使用android特效的原理和步骤

潘荣涛

德赛西威

[ 原理 3](#_Toc65846308)

[ 开发特效插件 4](#_Toc65846309)

[ JNI部分 4](#_Toc65846310)

[ 插件输出 5](#_Toc65846311)

[ 使用特效插件 7](#_Toc65846312)

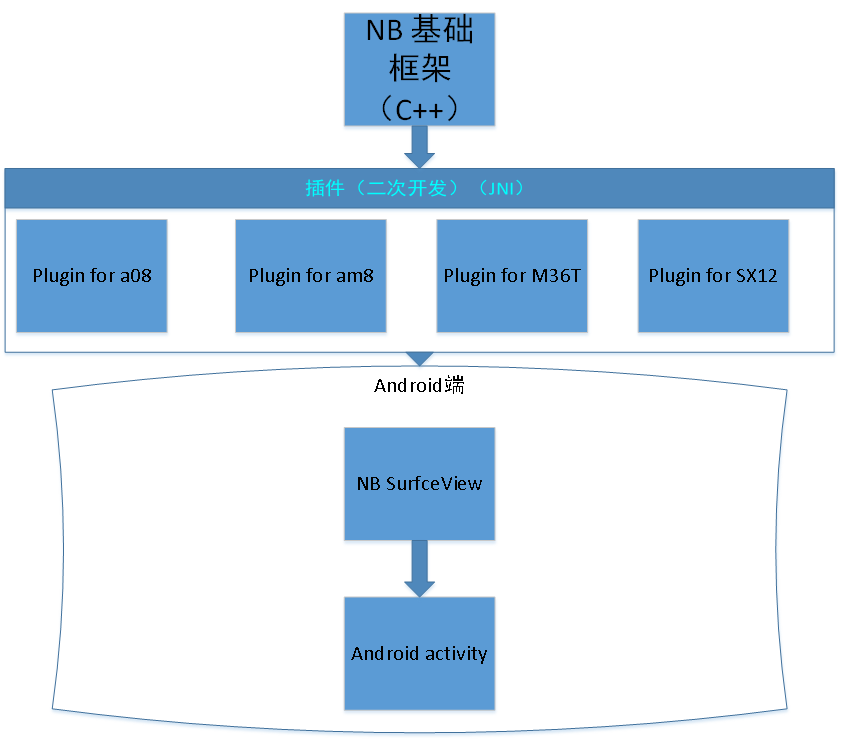
[ 环境配置 7](#_Toc65846313)

[ 显示 10](#_Toc65846314)

[ 交互 10](#_Toc65846315)

# 原理

为android平台开发一个特效组件，需要两方的配合。一方是基于NB框架开发的插件，另一方则是使用该插件的android app。使用JNI的方式达到两种语言的交互和配合。



从图中看到，NBCore是开发插件的基础，它提供二次开发插件的可能。而插件是根据每个项目不同需求所做的二次开发。因此，一般而言，NBCore的部分不怎么变，而插件则会依项目的不同需求千变万化，如Plugin for a08, Plugin for M36T等等。除此之外，NB也提供比较固定的NBSurfaceView.java供android app直接使用，隐藏了JNI开发的细节，android app开发者可以不需要熟悉JNI直接使用NBSurceView.java来呈现最终的效果，并与之交互。

其中，NBCore和插件都是c/c++完成，NBSurceView.java是android 语言完成的，这两部分都算是基于NB开发组件应该提供的东西。

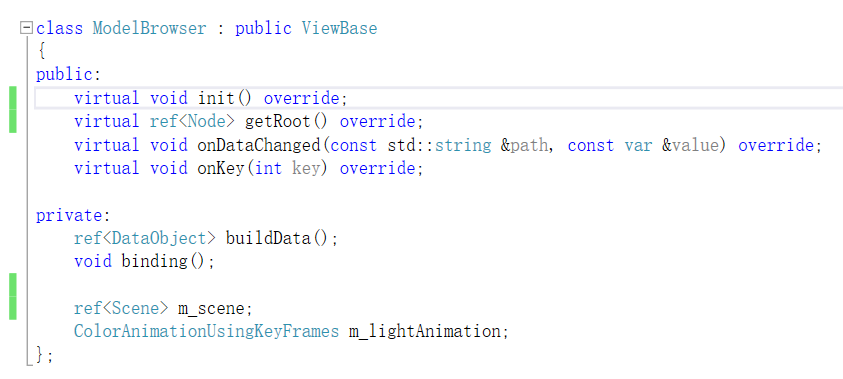
# 开发特效插件

## JNI部分

JNI的本质就是使用C/C++开发一个so供android端使用，只要接口定义符合android JNI方式即可。

为了隐藏JNI的一些规则和防止人为定义接口带来的错误，NB提供Common.h和Common.cpp，它固化了一些JNI开发的共性定义和调用，免去了开发JNI的繁杂步骤。

基于Common.h和Common.cpp，你只需新建一个类，继承ViewBase并实现ViewBase的虚函数即可。



其中，

virtual void init()是初始化组件的内容

virtual ref<Node> getRoot()返回根节点

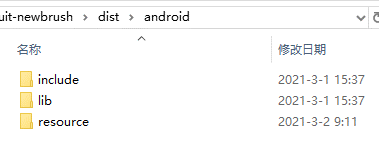
另外，还有一些onTouchDown、onTouchMove、onTouchUp、onDataChanged虚函数可按需重写

详细可查看projects下的各个工程。

编译后，会生成插件的so

## 插件输出

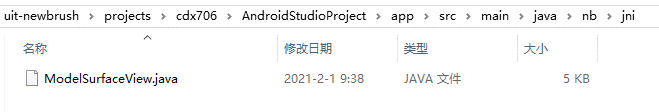
完整的插件包含所有依赖的so库、android java包以及使用到的图片资源等，它们在编译后自动输出到dist/android下。



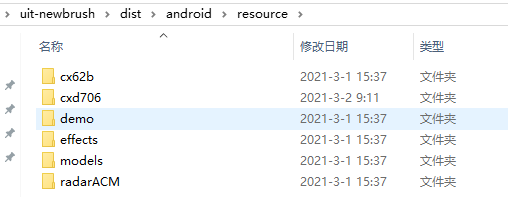
* so库



* java包



* 资源包（运行时需要，android端开发时不需要）

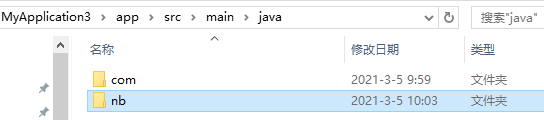


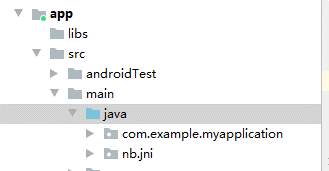
# 使用特效插件

## 环境配置

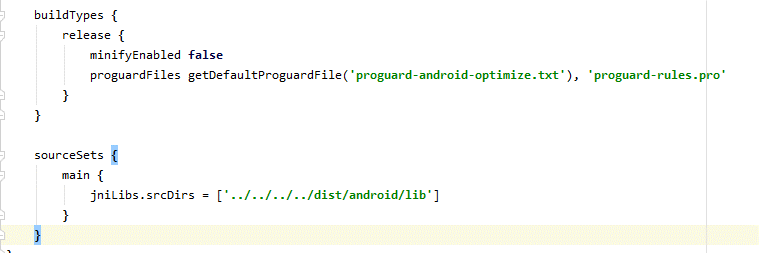
使用插件的一方是android app。插件开发方提供了so库，Java包和图片资源等，android开发者可以把这些插件开发发布的东西放在一个地方。

* 把nb提供的java包放在android工程的app/src/main/java目录下





* App/build.gradle添加以下内容，已让android工程能够识别插件路径



其中，jniLibs.srcDirs的值为插件so的路径，如果你放在d:/test下，这个值就改为”d:/test”。

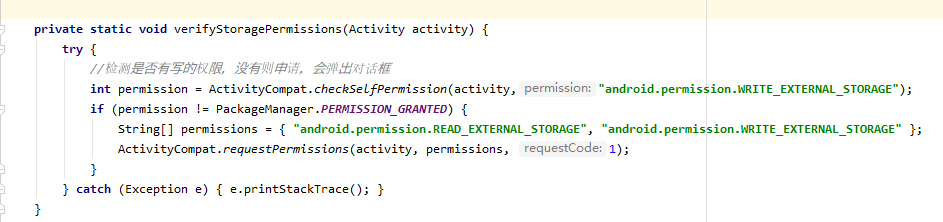
* 如果是android手机，访问资源可能需要权限，在app/src/main/AndroidManifest.xml文件添加

<**uses-permission android:name="android.permission.READ\_EXTERNAL\_STORAGE"**/>  
<**uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"** />

如下图：

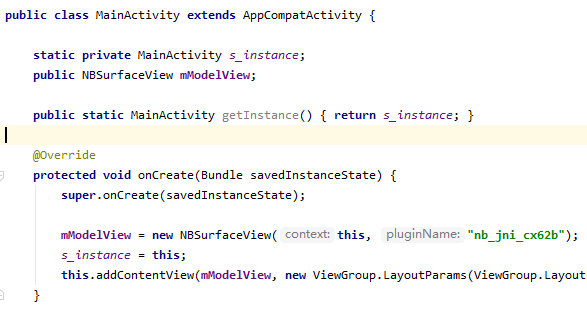


并在启动时申请权限



## 显示

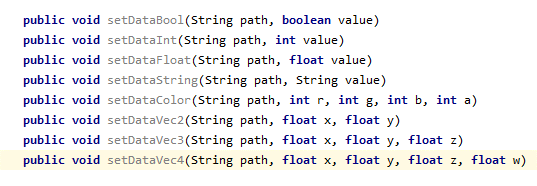
实例化NBSurfaceView，加载对应的插件，注意不同的插件名对应不同的插件



这样就完成了该插件的初始显示

## 交互

Android与插件的交互是通过NBSurface这个类来进行的，通过设置数据的方式来驱动插件的UI变化。



举例：如果插件是一个时钟插件，需要android传入“时间”这个数据以驱动UI的变化，则可约定一个类型为int型的数据”Time”来表示当前时间（秒数）。

setDataInt(“Time”, 1000);

详细请查看NBSurfaceView这个类。