**MediaPlayer java与jni代码**

**frameworks/base/media/java/android/media/MediaPlayer.java**

**frameworks/base/media/jni/android\_media\_MediaPlayer.cpp**

**A、System.loadLibrary("media\_jni");**

加载media jni动态库，执行jni代码的JNI\_OnLoad，调用register\_android\_media\_MediaPlayer，registerNativeMethods gMethod，映射mediaplayer java方法到jni 方法。比如下面的native\_init方法：

{"native\_init", "()V", (void \*)android\_media\_MediaPlayer\_native\_init}

**B、native\_init();**

jni中获取java类成员变量和方法到全局静态变量fields（static fields\_t fields），例如：postEventFromNative等。

mNativeContext中保存的是本地MediaPlayer对象地址。

jclass clazz;

clazz = env->FindClass("android/media/MediaPlayer");

fields.post\_event = env->GetStaticMethodID(clazz, "postEventFromNative","(Ljava/lang/Object;IIILjava/lang/Object;)V");

**C、EventHandler**

注册一个handler，绑定在应用进程looper中，接收native层抛上来的stop 、prepared 、seek 、complete 、start 、error 、info等播放器状态通知，并通过Listener callback机制通知app，app收到通知后执行相应的操作，比如收到onPrepared后执行start。

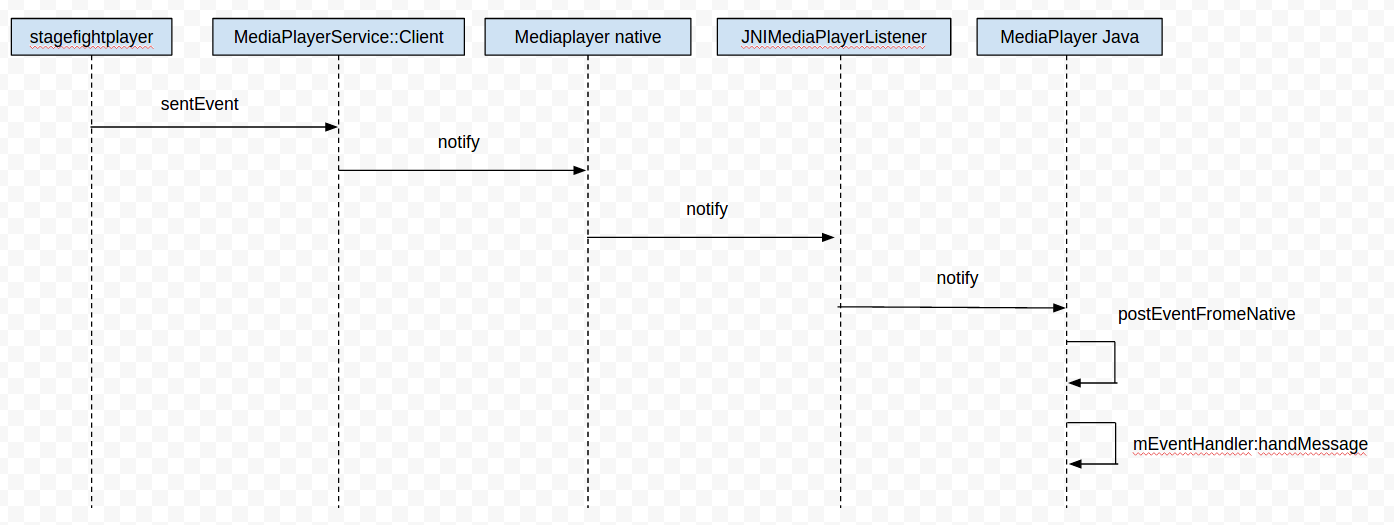
**D、TimeProvider**

？

**E、native\_setup**

1、把本地对象指针保存在java对象中，一一对应，new MediaPlayer，创建本地MediaPlayer类对象（C++），并保存到MediaPlayer java类的变量mNativeContext中，从java到jni后，jni中的每次调用都会去获取java MediaPlayer对象中记录的本地对象指针。

2、MediaPlayerListener接口类的派生类JNIMediaPlayerListener，注册该listener到本地MediaPlayer对象中作为（MediaPlayerListener）mListener，JNIMediaPlayerListener实现notify接口，notify中调用fields.post\_event（postEventFromNative，java MediaPlayer对象的静态方法），由于静态方法没有this指针，不能引用非静态方法和非静态数据成员；所以postEventFromNative参数中有一个弱引用java MediaPlayer对象，是native\_setup的参数(java MediaPlayer对象创建时对本身的一个弱引用)，通过此弱引用调用java MediaPlayer对象的所有public成员，比如mEventHandler，mEventHandler处理这些通知消息并发送到app。



**消息传递流程图**

**F、setVideoSurface**

App通过setDisplay和setSurface调用到setVideoSurface，jni部分code：根据jsurface创建一个Surface，Surface通过getIGraphicBufferProducer获取GraphicBufferProducer指针，保存在java类对象fields.surface\_texture中，并通过本地MediaPlayer对象的setVideoSurfaceTexture方法传递到具体的播放器实例中。

**G、Java MediaPlayer**通过new MediaPlayer()和MediaPlayer.create()创建的区别：create中封装了new MediaPlayer()，并执行了setDataSource、setDisplay、prepare等调用。

**H、其余接口setdatasource prepare pause stop等**，MediaPlayer的java和jni层code基本只是封装了natvie层MediaPlayer类的方法调用。