

全景播放 SDK 使用手册

2016 年 3 月 24 日

行者悟空

目录

1. 播放框架介绍.....	4
1.1 简介.....	4
2. 播放功能与接口介绍.....	5
2.1 接口介绍.....	5
2.1.1 setDataSourceAsync	5
2.1.2 setActiveNetWorkType	5
2.1.3 setDecoderType.....	5
2.1.4 setView.....	6
2.1.5 play.....	6
2.1.6 pause.....	6
2.1.7 resume	7
2.1.8 stop.....	7
2.1.9 setPosition	7
2.1.10 getPosition.....	7
2.1.11 getBufferPercent.....	8
2.1.12 duration	8
2.1.13 bufferPercent	8
2.1.14 setVolume	8
2.1.15 bufferedBandWidth.....	9
2.1.16 release.....	9
2.1.17 setViewSize.....	9
2.1.18 videoSizeChanged.....	9
2.1.19 setProxyServerConfig	10
2.1.20 bufferedBandPercent.....	10
2.1.21 SetRendType	10
2.1.22 SetMotionEnable.....	11
2.1.23 SetTouchEnable.....	11
2.2 回调信息和参数.....	11
2.2.1 ENotifyPrepare = 1	11
2.2.1 ENotifyPlay = 2	11
2.2.3 ENotifyComplete = 3	11
2.2.4 ENotifyPause = 4	12
2.2.5 ENotifyClose = 5	12
2.2.6 ENotifyException = 6	12
2.2.7 ENotifyUpdateDuration = 7.....	12
2.2.8 ENotifyAssetReaderFail = 8	12
2.2.9 ENotifyStop = 9	12
2.2.10 ENotifyTimeReset = 10	12
2.2.11 ENotifySeekComplete = 11.....	12
2.2.12 ENotifyAudioFormatChanged = 12	13
2.2.13 ENotifyVideoFormatChanged = 13	13
2.2.14 ENotifyReserved = 14.....	13
2.2.15 ENotifyReserved = 15.....	13
2.2.16 ENotifyBufferingStart = 16.....	13
2.2.17 ENotifyBufferingDone = 17.....	13
2.2.18 ENotifyDNSDone = 18.....	13

2.2.19	ENotifyConnectDone = 19.....	13
2.2.20	ENotifyHttpHeaderReceived = 20	14
2.2.21	ENotifyPrefetchStart = 21	14
2.2.22	ENotifyPrefetchCompleted = 22.....	14
2.2.23	ENotifyCacheCompleted = 23	14
2.2.24	ENotifyMediaStartToOpen = 24	14
2.2.25	ENotifyMediaFirstFrame = 25	14
2.2.26	ENotifyMediaPreOpenStart = 26.....	14
2.2.27	ENotifyMediaPreOpenSucess = 27.....	14
2.2.28	ENotifyMediaPreOpenFailed = 28.....	15
2.2.29	ENotifyMediaChangedStart = 50.....	15
2.2.30	ENotifyMediaChangedSucess = 51	15
2.2.31	ENotifyMediaChangedFailed = 52	15
2.3	返回的错误 ID	15
3.	调用流程.....	17
3.1	在线音视频 加载失败.....	17
3.2	在线音视频加载时推出.....	17
3.3	在线音视频 播放.....	18
3.4	在线音视频加载时资源被抢占	19

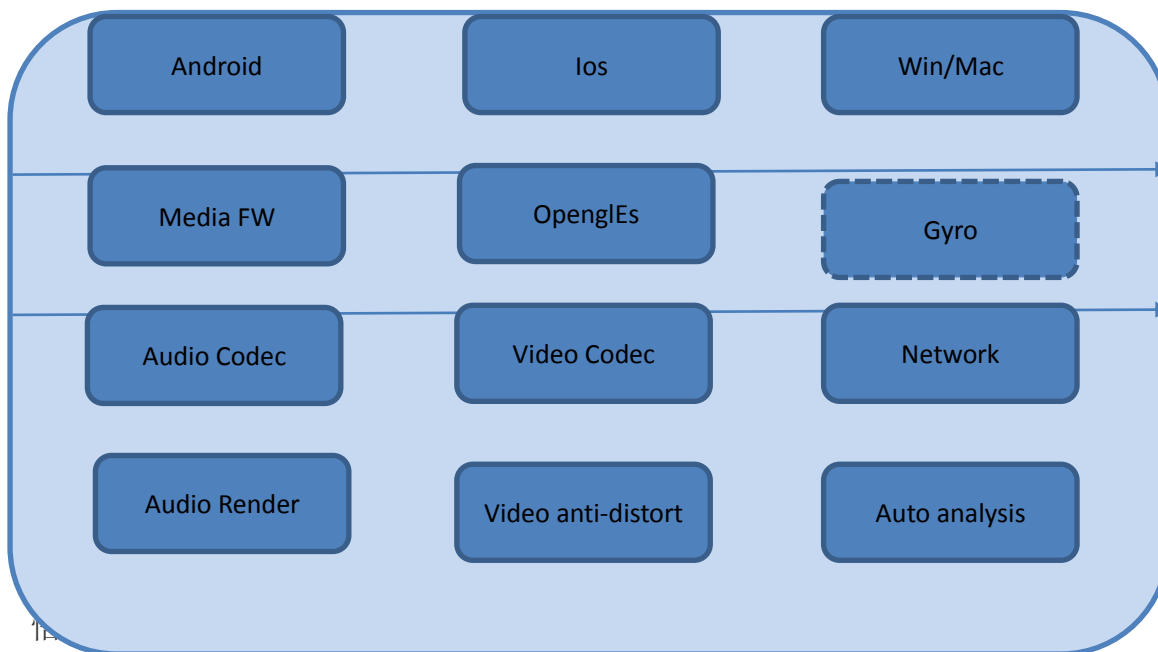
1. 播放框架介绍

1.1 简介

行者悟空播放器是国内首款带沉浸式全景音效的全景播放器，视频 360 度渲染，真正沉浸式观看视频。此播放框架可以播放本地，支持在线流，也支持直播流，支持 Flv，HLS 流，也支持在线 MP4 或者直播的 m3u8 流媒体播放过程中根据当前网络状况码率自动适配。

全部代码纯手动打造，没有开源代码，不依赖开源库，不依赖系统播放器，结构超级优化，增加功能方便快捷，查找问题更快速。

整个播放框架分上中下 3 层：



上层为应用层，根据提供的接口各个平台，根据产品需求开发

中间 MFW 是播放逻辑框架层，控制音视频播放，以及音视频同步，opengles 渲染，已经陀螺仪的平滑计算。

下面的就是音视频解码，还有，文件解析，及其网络下载模块。

支持文件格式有 m3u8，flv.支持的音频解码器有 aac，视频解码有 H264（android 4.1 以上，ios 是 ios8 以上，其他软解），HEVC（android 机器有硬解器）。其中对于软解，aac 解码，H264 解码，颜色空间转换，有对不同 CPU 有不同优化版本。

所有功能（播放，解析，解码库）大小在 2.5M，

对于性能和库大小，我们还会持续优化。

2. 播放功能与接口介绍

2.1 接口介绍

2.1.1 setDataSourceAsync

函数声明

```
public void setDataSourceAsync(String aUrl, int aflag);
```

函数作用

起步设置播放源,属于异步调用, ENotifyMediaStartToOpen 表示开始打开, ENotifyPrepare 事件表示打开成功。

参数说明

String aUrl 流路径, 本地文件路径或者网络连接

int aflag 打开参数属性, 具体属性有

int OPEN_DEFAULT = 0x00; //默认模式打开, 读文件或者写文件读文件模式

int OPEN_BUFFER = 0x01; //缓冲模式打开, 主要为了网络视频文件播放。

2.1.2 setActiveNetWorkType

函数声明

```
public void setActiveNetWorkType(int type);
```

函数作用

设置网络类型。

参数说明

int type 网络类型, 具体属性有

```
enum TActiveNetWorkType
```

```
{
```

```
    EActiveNetWorkNone = 0 //未知或者默认类型
```

```
    ,EActiveNetWorkGPRS = 1 //GPRS
```

```
    ,EActiveNetWorkWIFI = 2 //WIFI
```

```
};
```

2.1.3 setDecoderType

函数声明

```
public void setDecoderType(int type);
```

函数作用

设置视频解码类型。

参数说明

int type 视频解码类型，具体属性有
 int *EDecoderDefault* = 0x00; //自适应类型，尽量使用软件
 int *EDecoderSoft* = 0x01; //纯软件解码

2.1.4 setView

函数声明

public void setView (void* sv);

函数作用

设置当前视频播放 View

参数说明

void* sv 视频播放 View,不同平台有不同的 View

2.1.5 play

函数声明

public int play();

函数作用

开始播放，是播放器在播放状态，

参数说明

无

2.1.6 pause

函数声明

public int pause(boolean aFadeOut);

函数作用

暂停，是播放器在暂停状态

参数说明

boolean aFadeOut 是否需要做淡出效果

2.1.7 resume

函数声明

```
public int resume(boolean aFadeIn);
```

函数作用

继续播放，是播放器在继续播放状态

参数说明

boolean aFadeIn 是否需要做淡入效果

2.1.8 stop

函数声明

```
public int stop();
```

函数作用

停止播放，是播放器在停止播放状态，属于异步调用，事件 ENotifyClose 表示 stop 成功。

2.1.9 setPosition

函数声明

```
public int setPosition(int aPos, int flag);
```

函数作用

seek 操作，即拖动操作。

参数说明

int aPos seek 到的位置，单位毫秒

int flag seek 属性，具体属性有

 int SEEK_FAST = 0x00; //快速 seek，及关键帧 seek

 int SEEK_CORRECT = 0x01; //准确 seek

返回值

当前 seek 到的位置，单位毫秒，

如果 < 0，表示 seek 失败。

2.1.10 getPosition

函数声明

```
public int getPosition();
```

函数作用

获取播放位置

参数说明

无

返回值

当前播放位置，单位毫秒

2.1.11 getBufferPercent

函数声明

```
public float getBufferPercent();
```

函数作用

获取缓冲进度

返回值

获取当前缓冲进度，百分比，小数，相当于 `bufferedPercent()*0.01`

2.1.12 duration

函数声明

```
public int duration();
```

函数作用

获取音视频文件时长

参数说明

无

返回值

获取当前播放文件时长，单位毫秒，直播的时候是-1

2.1.13 bufferPercent

函数声明

```
public int bufferedPercent();
```

函数作用

获取缓冲百分比

返回值

获取当前缓冲进度，百分比

2.1.14 setVolume

函数声明

```
public void setVolume(float aLVolume, float aRVolume);
```

函数作用

设置音量

参数说明

float **aLVolume** 左声道

float **aRVolume** 右声道

2.1.15 bufferedBandWidth

函数声明

public int bufferedBandWidth();

函数作用

获得下载数据速度

返回值

当前网络下载速度， byte/sec

2.1.16 release

函数声明

public void release();

函数作用

释放资源，调用后，播放引擎不能使用

参数说明

无

2.1.17 setViewSize

函数声明

public void setViewSize(**int** width, **int** Height);

函数作用

设置观看视频的显示大小

参数说明

int width; //显示长

int height; //显示宽

2.1.18 videoSizeChanged

函数声明

public void videoSizeChanged(**int** width, **int** Height);

函数作用

提示视频长宽变化了

参数说明

```
int width;    //视频长
int height;   //视频宽
```

2.1.19 setProxyServerConfig

函数声明

```
public void setProxyServerConfig(String ip, int port, String authenkey,
boolean useProxy);
```

函数作用

设置网络连接代理

参数说明

```
String ip    //代理的 ip
int port     //代理的端口
String authenkey //代理授权码
boolean useProxy //是否用代理
```

2.1.20 bufferedBandPercent

函数声明

```
public int bufferedBandPercent ();
```

函数作用

获得缓冲时下载数据百分比

返回值

当前网络下载所需的数据百分比

2.1.21 SetRenderType

函数声明

```
public void setRenderType (int nRenderType);
```

函数作用

设置当前渲染的种类

渲染的具体种类有

ERenderDefault = 0 //普通模式

ERenderLR3D = 0x1 //左右分屏的源 3D 视频

ERenderUD3D = 0x2 //上下分屏的源 3D 视频

ERenderGlobeView = 0x4 // 360 全景视频

ERenderSplitView = 0x8 // 分屏显示

ERender180View = 0x10 // 180 度全景视频

ERenderMeshDistortion = 0x20 // mesh 反畸变

2.1.22 SetMotionEnable

函数声明

```
public void setMotionEnable (bool aEnable);
```

函数作用

是否启动陀螺仪

2.1.23 SetTouchEnable

函数声明

```
public void setTouchEnable (bool aEnable);
```

函数作用

是否启动手动触摸旋转渲染

2.2 回调信息和参数

2.2.1 ENotifyPrepare = 1

提示音视频已经准备好，就是音视频文件所需要的文件头信息已经解析，这时候可以拿到音频文件的文件时长，可以做前期 seek，或者开始播放。

2.2.1 ENotifyPlay = 2

提示音视频已经开始播放，本地文件就是正式开始播放，在线文件可能需要缓冲，边缓冲边解码。

2.2.3 ENotifyComplete = 3

提示音视频已经播放完成，可以播放下一首或者 seek 到文件前面的位置。

参数：

参数 1 0 表示正常结束，非 0 异常播放完成，具体错误 ID 看 2.3。

2.2.4 ENotifyPause = 4

提示音视频现在属于暂停状态。

2.2.5 ENotifyClose = 5

提示当前播放的音视频已经播放结束。

2.2.6 ENotifyException = 6

提示当前播放的音视频出现异常。

参数：

参数 1 异常原因 ID，具体看 2.3 错误 ID。

如果是网络相关，有参数 2 和 3

参数 2 异常时网络状态码

参数 3 异常时网络 ip

2.2.7 ENotifyUpdateDuration = 7

提示当前播放的音视频文件文件时长更新了。

2.2.8 ENotifyAssetReaderFail = 8

提示当前读取的音视频文件文件失败。

2.2.9 ENotifyStop = 9

提示当前播放的音视频是停止状态。

2.2.10 ENotifyTimeReset = 10

提示当前播放的音视频出现时间搓跳跃。

2.2.11 ENotifySeekComplete = 11

提示 seek 动作已经完成

2.2.12 ENotifyAudioFormatChanged = 12

提示音频的格式变化了，主要是 samplerate， channel 变化

2.2.13 ENotifyVideoFormatChanged = 13

提示视频的格式变化了，主要是视频的长和宽变化

2.2.14 ENotifyReserved = 14

保留 ID，以后更新

2.2.15 ENotifyReserved = 15

保留 ID，以后更新

2.2.16 ENotifyBufferingStart = 16

提示开始缓冲，表示网络数据不足，正在下载媒体数据。

2.2.17 ENotifyBufferingDone = 17

提示缓冲结束，表示网络数据已经足了，音视频文件开始正常播放。

2.2.18 ENotifyDNSDone = 18

提示在线音视频连接网络时 DNS 解析成功

2.2.19 ENotifyConnectDone = 19

提示在线音视频连接网络时接成功

2.2.20 ENotifyHttpHeaderReceived = 20

提示在线音视频连接网络时获得头信息成功。

2.2.21 ENotifyPrefetchStart = 21

提示在线音视频开始下载准备数据。

2.2.22 ENotifyPrefetchCompleted = 22

提示在线音视频下载准备数据完成。

2.2.23 ENotifyCacheCompleted = 23

提示在线音视频下载全部完成。

2.2.24 ENotifyMediaStartToOpen = 24

提示开始加载设置的音视频文件或连接。

2.2.25 ENotifyMediaFirstFrame = 25

提示视频文件第一帧已经显示了。

2.2.26 ENotifyMediaPreOpenStart = 26

提示开始预下载并与解析将要播放的下一个文件。

2.2.27 ENotifyMediaPreOpenSucess = 27

提示预加载下一个文件成功。

2.2.28 ENotifyMediaPreOpenFailed = 28

提示预加载下一个文件失败。

2.2.29 ENotifyMediaChangedStart = 50

提示开始进行码率切换。

2.2.30 ENotifyMediaChangedSucess = 51

提示码率切换成功。

2.2.31 ENotifyMediaChangedFailed = 52

提示码率切换失败。

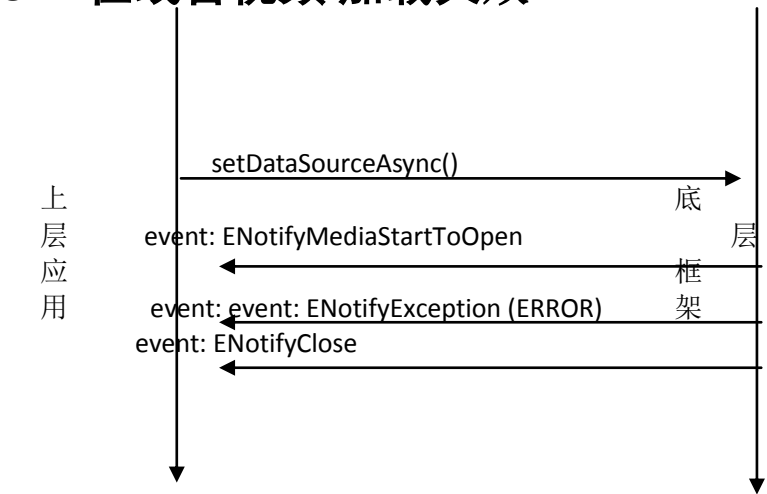
2.3 返回的错误 ID

```
#define KErrNone          0    //表示没有错误
#define KErrNotFound      -1   //需要加载的文件或者库没有找到
#define KErrGeneral       -2   //一般错误
#define KErrCancel        -3   //操作被取消
#define KErrNoMemory      -4   //内存不够，分配内存时失败
#define KErrNotSupported  -5   //不支持的文件或者特性属性
#define KErrArgument      -6   //错误的参数设置，空指针或者不支持类型
#define KErrTotalLossOfPrecision -7 //丢失精度错误
#define KErrBadHandle     -8   //无效的句柄
#define KErrOverflow       -9   //数据溢出失败
#define KErrUnderflow     -10  //数据不够失败
#define KErrAlreadyExists -11  //所要分配的类型或者其他已经存在
#define KErrPathNotFound  -12  //文件路径不存在
#define KErrDied          -13  //线程或者硬件等等已经退出
#define KErrInUse         -14  //所要的类型或者其他正在使用
```

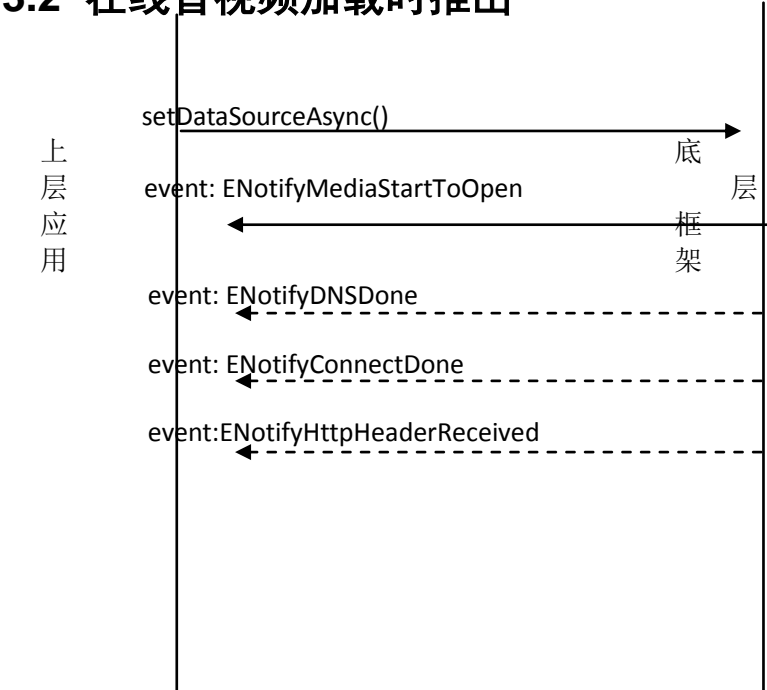
```
#define KErrServerTerminated    -15 //网络服务器已经关闭
#define KErrServerBusy -16 //网络服务器正在忙
#define KErrCompletion -17 //非正常结束
#define KErrNotReady -18 //数据或者网络等等还没有准备好
#define KErrUnknown -19 //未知错误, 超出范围内的错误
#define KErrCorrupt -20 //崩溃错误
#define KErrAccessDenied -21 //没有权限读取
#define KErrLocked -22 //所要的类型或者其他正被锁住
#define KErrWrite -23 //写文件或者缓存失败
#define KErrDisMounted -24 //mount 失败, 现在是 dismount 状态。
#define KErrEof -25 //读取文件到文件末尾了
#define KErrDiskFull -26 //内存或者硬盘已经满了
#define KErrBadDriver -27 //错误的驱动
#define KErrBadName -28 //错误的名字, 不支持的名字
#define KErrCommsLineFail -29 //命令行失败
#define KErrCommsFrame -30 //命令指令错误
#define KErrCommsOverrun -31 //命令指令太多
#define KErrCommsParity -32 //奇偶矫正失败
#define KErrTimedOut -33 //已经超时
#define KErrCouldNotConnect -34 //连接网络失败, 不能连接
#define KErrCouldNotDisconnect -35 //网络不能断开
#define KErrDisconnected -36 //网络已经非正常断开
#define KErrBadLibraryEntryPoint -37 //错误的指针接口
#define KErrBadDescriptor -38 //错误的描述
#define KErrAbort -39 //非正常退出
#define KErrTooBig -40 //所需要的数据太大
#define KErrDivideByZero -41 //被 0 除
#define KErrBadPower -42 //表示电量不足了
#define KErrDirFull -43 //操作文件夹失败, 已经满了或者其他
#define KErrHardwareNotAvailable -44 //硬件不能使用
#define KErrSessionClosed -45 //这次对话或者握手结束
#define KErrPermissionDenied -46 //没有权限, 错误操作
#define KErrExtensionNotSupported -47 //扩展的属性不支持
#define KErrCommsBreak -48 //命令行断开
#define KErrNoSecureTime -49 //不在保密时间内
#define KErrOperationInterrupted -50 //操作被中断
#define KErrStartAudioUnitFailed -51 //启动 audioUnit 失败
#define KErrFormatIsMP3 -52 //这个文件不是 MP3 文件
#define KErrFormatIsDTS -53 //这个文件不是 DTS 文件
#define KErrUrlInvalid -54 //无效的 url
#define KErrOnLineFormatNotSupport -55 //现在的媒体格式不支持
#define KErrFileParserException -56 //文件解析失败
#define KErrFileReadTimeout -57 //文件读取超时
#define KErrFileNotSupport -58 //不支持的媒体文件
#define KErrVFPNotSupport -59 //cpu 不支持浮点计算
#define KErrFilParseOutOfRange -60 //超出范围的文件解析, 解析失败, 可能不是一个媒体文件
#define KErrSyncReadErr -61 //读取文件信息头的时候失败, 可能不是一个媒体文件
#define KErrSyncEofIncomplete -62 //扫描全文件也没有文件信息头, 不是一个媒体文件
#define KErrNeedWait -63 //暂时拿不到数据, 需要等待
#define KErrNetWorkAbnormalDisconneted -64 //网络非正常断开
#define KErrFormatChanged -70 //视音频属性发生变化
```

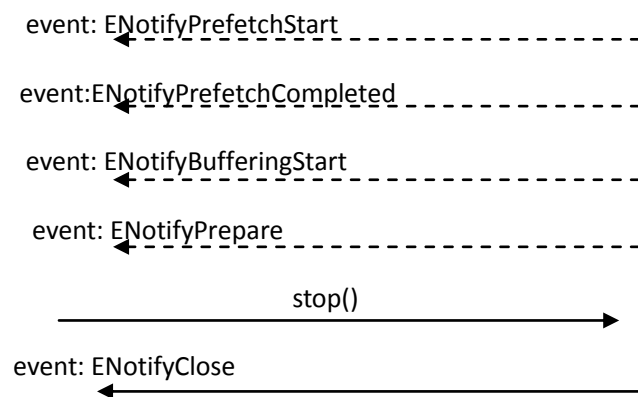

3. 调用流程

3.1 在线音视频 加载失败



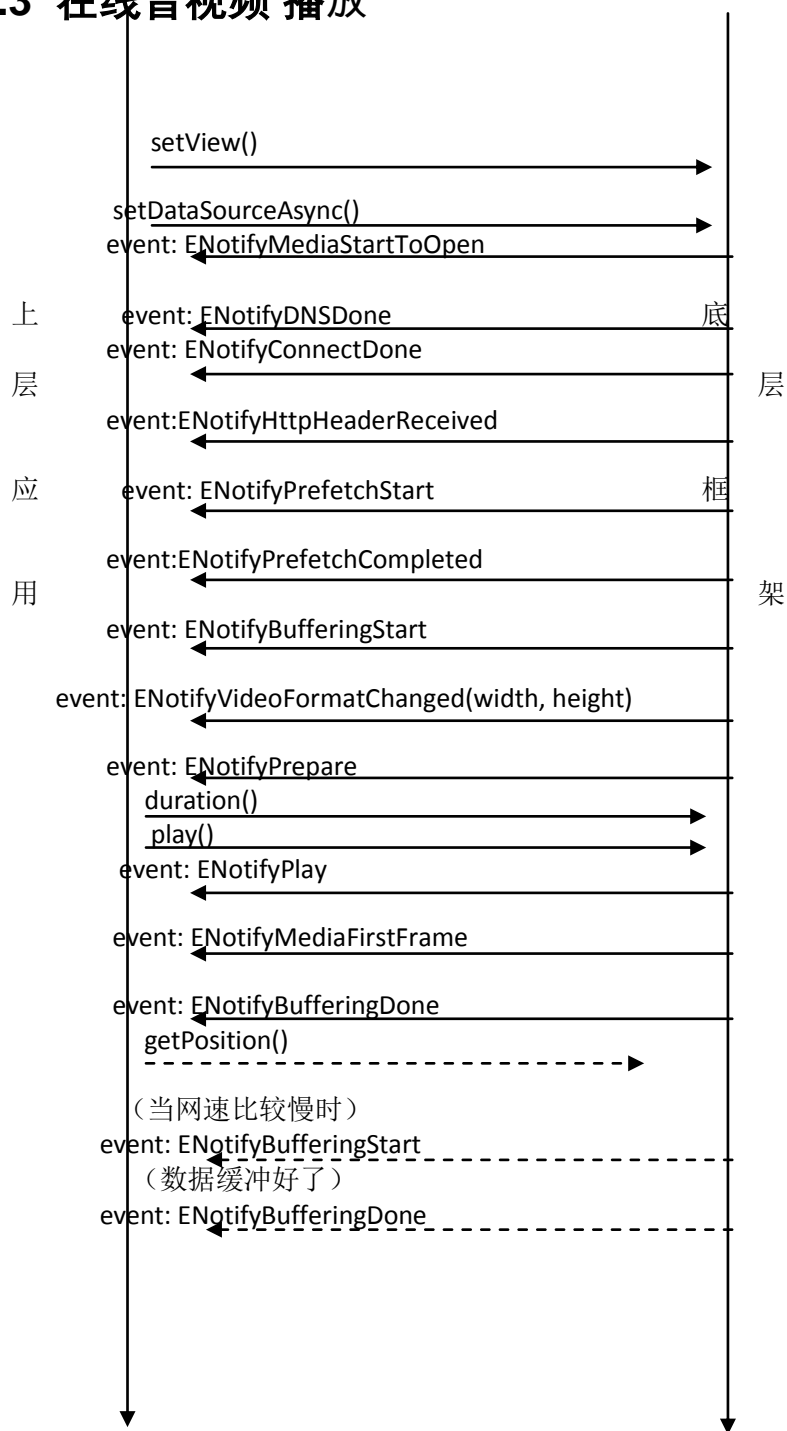
3.2 在线音视频加载时推出

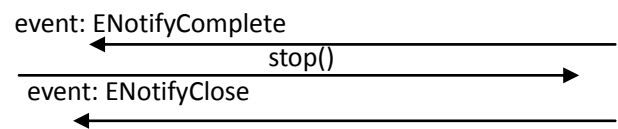




注：根据 stop 的时机，虚箭头 event 可能都有可能只有几个或者没有。

3.3 在线音视频 播放





注：当网速比较快的时候，事件 ENotifyBufferingDone 可能在 事件 ENotifyMediaFirstFrame 前面。

3.4 在线音视频加载时资源被抢占

