

直播 SDK

API 文档

产品文档



腾讯云

【版权声明】

©2013-2024 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

文档目录

API 文档

iOS

概览

推流

V2TXLivePusher

V2TXLivePusherObserver

TXBeautyManager

拉流

V2TXLivePlayer

V2TXLivePlayerObserver

示例

Android

概览

推流

V2TXLivePusher

V2TXLivePusherObserver

TXBeautyManager

拉流

V2TXLivePlayer

V2TXLivePlayerObserver

示例

Web

TXLivePusherObserver

Flutter

概览

推流

V2TXLivePusher

V2TXLivePusherObserver

拉流

V2TXLivePlayer

V2TXLivePlayerObserver

示例

错误码表

API 文档

iOS

概览

最近更新时间：2022-12-23 17:44:05

V2TXLivePlayer

视频播放器

请参见 [V2TXLivePlayer](#)。

主要负责从指定的直播流地址拉取音视频数据，并进行解码和本地渲染播放。

播放器包含如下能力：

- 支持 RTMP、HTTP-FLV、TRTC 以及 WebRTC 协议。
- 屏幕截图，可以截取当前直播流的视频画面。
- 延时调节，可以设置播放器缓存自动调整的最小和最大时间。
- 自定义的视频数据处理，您可以根据项目需要处理直播流中的视频数据后，再进行渲染以及播放。

SDK 基础函数

API	描述
setObserver	设置播放器回调。

播放基础接口

API	描述
setRenderView	设置播放器的视频渲染 View。
startLivePlay	10.7 版本开始， <code>startPlay</code> 变更为 <code>startLivePlay</code> ，需要通过 <code>V2TXLivePremier#setLicence</code> 或者 <code>TXLiveBase#setLicence</code> 设置 Licence 后方可成功播放，否则将播放失败（黑屏），全局仅设置一次即可。直播 License、短视频 License 和视频播放 License 均可使用，若您暂未获取上述 License，可 快速免费申请测试版 License 以正常播放，正式版 License 需 购买 。
stopPlay	停止播放。
isPlaying	是否正在播放。

视频相关接口

API	描述
setRenderRotation	设置播放器画面的旋转角度。
setRenderFillMode	设置画面的填充模式。
pauseVideo	暂停播放器的视频流。
resumeVideo	恢复播放器的视频流。
snapshot	截取播放过程中的视频画面。
enableCustomRendering	设置视频自定义渲染回调。

音频相关接口

API	描述
pauseAudio	暂停播放器的音频流。
resumeAudio	恢复播放器的音频流。
setPlayoutVolume	设置音量。
enableVolumeEvaluation	启用播放音量大小提示。

更多实用接口

API	描述
setCacheParams	设置播放器缓存自动调整的最小和最大时间 (单位：秒)。
showDebugView	是否显示播放器状态信息的调试浮层。

V2TXLivePlayerObserver

腾讯云直播播放的回调通知。

SDK 基础回调

API	描述
onError	直播播放器错误通知，播放器出现错误时，会回调该通知。

API	描述
onWarning	直播播放器警告通知。
onConnected	已经成功连接到服务器通知。

视频相关回调

API	描述
onVideoPlaying	视频播放事件通知。
onVideoLoading	视频加载事件通知。
onVideoResolutionChanged	直播播放器分辨率变化通知。
onSnapshotComplete	截图回调。
onRenderVideoFrame	自定义视频渲染回调。

音频相关回调

API	描述
onAudioPlaying	音频播放事件通知。
onAudioLoading	音频加载事件通知。
onPlayoutVolumeUpdate	播放器音量大小回调。

统计回调

API	描述
onStatisticsUpdate	直播播放器统计数据回调。

V2TXLivePusher

直播推流类

请参见 [V2TXLivePusher](#)。

主要负责将本地的音频和视频画面进行编码，并推送到指定的推流地址，支持任意的推流服务端。

推流器包含如下能力：

- 自定义的视频采集，让您可以根据项目需要定制自己的音视频数据源。
- 美颜、滤镜、贴纸，包含多套美颜磨皮算法（自然&光滑）和多款色彩空间滤镜（支持自定义滤镜）。
- Qos 流量控制技术，具备上行网络自适应能力，可以根据主播端网络的具体情况实时调节音视频数据量。
- 脸形调整、动效挂件，支持基于优图 AI 人脸识别技术的大眼、瘦脸、隆鼻等脸形微调以及动效挂件效果，只需要购买 **优图 License** 就可以轻松实现丰富的直播效果。

SDK 基础函数

API	描述
setObserver	设置推流器回调。

推流基础接口

API	描述
setRenderView	设置本地摄像头预览 View。
startPush	开始音视频数据推流。
stopPush	停止推送音视频数据。
isPushing	当前推流器是否正在推流中。

视频相关接口

API	描述
setVideoQuality	设置推流视频编码参数。
setRenderRotation	设置本地摄像头预览画面的旋转角度。
setRenderMirror	设置本地摄像头预览镜像。
startCamera	打开本地摄像头。
stopCamera	关闭本地摄像头。
startVirtualCamera	开启图片推流。
stopVirtualCamera	关闭图片推流。
startScreenCapture	开启屏幕采集。
stopScreenCapture	关闭屏幕采集。

API	描述
snapshot	截取推流过程中的本地画面。
setWatermark	设置推流器水印。默认情况下，水印不开启。
setEncoderMirror	设置视频编码镜像。
enableCustomVideoCapture	开启/关闭自定义视频采集。
sendCustomVideoFrame	在自定义视频采集模式下，将采集的视频数据发送到 SDK。
enableCustomVideoProcess	开启/关闭自定义视频处理。
sendSeiMessage	发送 SEI 消息。

美颜相关接口

API	描述
getBeautyManager	获取美颜管理对象 TXBeautyManager，美颜的设置通过 TXBeautyManager 来设置。

音频相关接口

API	描述
startMicrophone	打开麦克风。
stopMicrophone	关闭麦克风。
setAudioQuality	设置推流音频质量。
enableVolumeEvaluation	启用采集音量大小提示。

音效相关接口

API	描述
getAudioEffectManager	获取音效管理对象。

设备管理相关接口

API	描述
getDeviceManager	获取设备管理对象。

更多实用接口

API	描述
setProperty	调用 V2TXLivePusher 的高级 API 接口。
setMixTranscodingConfig	设置云端的混流转码参数。
showDebugView	显示仪表盘。

V2TXLivePusherObserver

SDK 基础回调

API	描述
onError	直播推流器错误通知，推流器出现错误时，会回调该通知。
onWarning	直播推流器警告通知。

视频相关回调

API	描述
onPushStatusUpdate	推流器连接状态回调通知。
onSnapshotComplete	截图回调。
onProcessVideoFrame	自定义视频处理。
onGLContextDestroyed	SDK 内部的 OpenGL 环境的销毁通知。
onCaptureFirstVideoFrame	首帧视频采集完成的回调通知。

音频相关回调

API	描述
onCaptureFirstAudioFrame	首帧音频采集完成的回调通知。
onMicrophoneVolumeUpdate	麦克风采集音量值回调。

混流回调

API	描述
onSetMixTranscodingConfig	设置云端的混流转码参数的回调。

统计回调

API	描述
onStatisticsUpdate	直播推流器统计数据回调。

推流

V2TXLivePusher

最近更新时间：2022-10-13 11:40:06

功能

腾讯云直播推流器。

介绍

主要负责将本地的音频和视频画面进行编码，并推送到指定的推流地址，支持任意的推流服务端。

推流器包含如下能力：

- 自定义的视频采集，让您可以根据项目需要定制自己的音视频数据源。
- 美颜、滤镜、贴纸，包含多套美颜磨皮算法（自然&光滑）和多款色彩空间滤镜（支持自定义滤镜）。
- Qos 流量控制技术，具备上行网络自适应能力，可以根据主播端网络的具体情况实时调节音视频数据量。
- 脸形调整、动效挂件，支持基于优图 AI 人脸识别技术的大眼、瘦脸、隆鼻等脸形微调以及动效挂件效果，只需要购买 **优图 License** 就可以轻松实现丰富的直播效果。

SDK 基础函数

initWithLiveMode

初始化直播推流器。

```
- (instancetype) initWithLiveMode: (V2TXLiveMode) liveMode
```

参数

参数	类型	含义
liveMode	V2TXLiveMode	推流协议类型：RTMP/ROOM 协议，默认值：RTMP。

setObserver

设置推流器回调。通过设置回调，可以监听 V2TXLivePusher 推流器的一些回调事件，包括推流器状态、音量回调、统计数据、警告和错误信息等。

```
- (void) setObserver: (id<V2TXLivePusherObserver>) observer
```

参数

参数	类型	含义
observer	V2TXLivePusherObserver	推流器的回调目标对象，更多信息请参见 V2TXLivePusherObserver 。

推流基础接口

setRenderView

设置本地摄像头预览 View，本地摄像头采集到的画面，经过美颜、脸形调整、滤镜等多种效果叠加之后，最终会显示到传入的 View 上。

```
- (V2TXLiveCode) setRenderView:(TXView *)view
```

参数

参数	类型	含义
view	TXView *	本地摄像头预览 View。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startPush

开始音视频数据推流。

```
- (V2TXLiveCode) startPush:(NSString *)url
```

参数

参数	类型	含义
url	NSString *	推流的目标地址，支持任意推流服务端。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
 - V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，URL 不合法。
 - V2TXLIVE_ERROR_REFUSED：RTC 不支持同一设备上同时推拉同一个 StreamId。
-

stopPush

停止推送音视频数据。

```
- (V2TXLiveCode) stopPush
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
-

isPushing

当前推流器是否正在推流中。

```
- (int) isPushing
```

返回

是否正在播放。

- 1：正在推流中。
 - 0：已经停止推流。
-

视频相关接口

setVideoQuality

设置推流视频编码参数。

```
- (V2TXLiveCode) setVideoQuality: (V2TXLiveVideoEncoderParam *)param;
```

参数

参数	类型	含义
param	V2TXLiveVideoEncoderParam	视频编码参数。

setRenderRotation

设置本地摄像头预览画面的旋转角度。

```
(V2TXLiveCode) setRenderRotation: (V2TXLiveRotation) rotation
```

参数

参数	类型	含义
rotation	V2TXLiveRotation	旋转角度，默认值：V2TXLiveRotation0。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveRotation 枚举值

取值	含义
V2TXLiveRotation0	不旋转。
V2TXLiveRotation90	顺时针旋转90度。
V2TXLiveRotation180	顺时针旋转180度。
V2TXLiveRotation270	顺时针旋转270度。

setRenderMirror

设置本地摄像头预览镜像。

```
(V2TXLiveCode) setRenderMirror: (V2TXLiveMirrorType) mirrorType
```

参数

参数	类型	含义
mirrorType	V2TXLiveMirrorType	摄像头镜像类型，默认值：V2TXLiveMirrorTypeAuto。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveMirrorType枚举值

取值	含义
V2TXLiveMirrorTypeAuto	默认镜像类型. 在这种情况下，前置摄像头的画面是镜像的，后置摄像头的画面不是镜像的。
V2TXLiveMirrorTypeEnable	前置摄像头和后置摄像头都切换为镜像模式。
V2TXLiveMirrorTypeDisable	前置摄像头和后置摄像头都切换为非镜像模式。

startCamera

打开本地摄像头。

说明：

startVirtualCamera、startCamera 和 startScreenCapture 同一 Pusher 实例下，仅有一个能上行，三者为覆盖关系。例如先调用 startCamera，后调用 startVirtualCamera。此时表现为暂停摄像头推流，开启图片推流。

```
- (V2TXLiveCode) startCamera:(BOOL) frontCamera
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopCamera

关闭本地摄像头。

```
- (V2TXLiveCode) stopCamera
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startVirtualCamera

开启图片推流。

说明：

startVirtualCamera、startCamera 和 startScreenCapture 同一 Pusher 实例下，仅有一个能上行，三者为覆盖关系。例如先调用 startCamera，后调用 startVirtualCamera。此时表现为暂停摄像头推流，开启图片推流。

```
- (V2TXLiveCode) startVirtualCamera:(TXImage *) image
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopVirtualCamera

关闭图片推流。

```
- (V2TXLiveCode) stopVirtualCamera
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startScreenCapture

开启屏幕采集。

说明：

- iOS 端暂不支持通过此接口开启屏幕采集。
- startVirtualCamera、startCamera 和 startScreenCapture 同一 Pusher 实例下，仅有一个能上行，三者为覆盖关系。例如先调用 startCamera，后调用 startVirtualCamera。此时表现为暂停摄像头推流，开启图片推流。

1. 需要通过 iOS Broadcast Upload Extension 来开启屏幕采集。
2. 然后设置 enableCustomVideoCapture 开启自定义采集支持。
3. 最后通过 sendCustomVideoFrame 把 Broadcast Upload Extension 中采集到的屏幕画面送出去。

```
- (V2TXLiveCode) startScreenCapture: (NSString *) appGroup
```

参数

参数	类型	含义
appGroup	NSString *	主 App 与 Broadcast 共享的 Application Group Identifier，可以指定为 nil，但按照文档设置会使功能更加可靠。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopScreenCapture

关闭屏幕采集。

```
- (V2TXLiveCode) stopScreenCapture
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

snapshot

截取推流过程中的本地画面。

说明：

返回值成功后可以在 `V2TXLivePusherObserver#onSnapshotComplete:(TXImage *) image` 回调中获取截图图片。

```
- (V2TXLiveCode) snapshot
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_REFUSED：已经停止推流，不允许调用截图操作。

setWatermark

设置推流器水印。默认情况下，水印不开启。

```
- (V2TXLiveCode) setWatermark:(TXImage *) image x:(float)x y:(float)y scale:(float) scale
```

参数

参数	类型	含义
image	TXImage *	水印图片。如果该值为null，则等效于禁用水印。
x	float	水印的坐标。
y	float	水印的坐标。
scale	float	水印的缩放比例。

setEncoderMirror

设置视频编码镜像。

```
- (V2TXLiveCode) setEncoderMirror: (BOOL) mirror
```

参数

参数	类型	含义
mirror	BOOL	是否镜像。默认值：NO。

enableCustomVideoCapture

开启/关闭自定义视频采集。在自定义视频采集模式下，SDK 不再从摄像头采集图像，只保留编码和发送能力。

```
- (V2TXLiveCode) enableCustomVideoCapture: (BOOL) enable
```

参数

参数	类型	含义
enable	BOOL	是否开启自定义采集。默认值：NO。

sendCustomVideoFrame

在自定义视频采集模式下，将采集的视频数据发送到 SDK。

```
- (V2TXLiveCode) sendCustomVideoFrame: (V2TXLiveVideoFrame *) videoFrame
```

参数

参数	类型	含义
videoFrame	V2TXLiveVideoFrame *	向 SDK 发送的视频帧数据。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

- V2TXLIVE_ERROR_INVALID_PARAMETER：发送失败，视频帧数据不合法。
- V2TXLIVE_ERROR_REFUSED：发送失败，您必须先调用 `enableCustomVideoCapture` 开启自定义视频采集。

enableCustomVideoProcess

开启/关闭自定义视频处理。

```
- (V2TXLiveCode)enableCustomVideoProcess:(BOOL)enable pixelFormat:(V2TXLivePixelFormat)pixelFormat bufferType:(V2TXLiveBufferType)bufferType
```

参数

参数	类型	含义
enable	BOOL	是否开启自定义视频处理。默认值：NO。
pixelFormat	V2TXLivePixelFormat	视频帧的像素格式。
bufferType	V2TXLiveBufferType	视频数据包装格式。

V2TXLivePixelFormat 枚举值

取值	说明
V2TXLivePixelFormatUnknown	未知。
V2TXLivePixelFormatI420	YUV420P I420。
V2TXLivePixelFormatTexture2D	OpenGL 2D 纹理。

V2TXLiveBufferType 枚举值

取值	说明
V2TXLiveBufferTypeUnknown	未知。
V2TXLiveBufferTypeByteBuffer	DirectBuffer，装载 I420 等 buffer，在 native 层使用。
V2TXLiveBufferTypeByteArray	byte[]，装载 I420 等 buffer，在 Java 层使用。
V2TXLiveBufferTypeTexture	直接操作纹理 ID，性能最好，画质损失最少。

sendSeiMessage

发送 SEI 消息。播放端 [V2TXLivePlayer](#) 通过 [V2TXLivePlayerObserver](#) 中的 `onReceiveSeiMessage` 回调来接收该消息。

```
- (V2TXLiveCode) sendSeiMessage:(int)payloadType data:(NSData *)data;
```

参数

参数	类型	含义
payloadType	int	数据类型，支持 5、242。推荐填：242
data	NSData *	待发送的数据

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

美颜相关接口

getBeautyManager

获取美颜管理对象 [TXBeautyManager](#)。

通过美颜管理，您可以使用以下功能：

- 设置“美颜风格”、“美白”、“红润”、“大眼”、“瘦脸”、“V脸”、“下巴”、“短脸”、“小鼻”、“亮眼”、“白牙”、“祛眼袋”、“祛皱纹”、“祛法令纹”等美容效果。
- 调整“发际线”、“眼间距”、“眼角”、“嘴形”、“鼻翼”、“鼻子位置”、“嘴唇厚度”、“脸型”。
- 设置人脸挂件（素材）等动态效果。
- 添加美妆。
- 进行手势识别。

```
- (TXBeautyManager *)getBeautyManager
```

音频相关接口

startMicrophone

打开麦克风。

```
- (V2TXLiveCode) startMicrophone
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopMicrophone

关闭麦克风。

```
- (V2TXLiveCode) stopMicrophone
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setAudioQuality

设置推流音频质量。

```
- (V2TXLiveCode) setAudioQuality:(V2TXLiveAudioQuality) quality
```

参数

参数	类型	含义
quality	V2TXLiveAudioQuality	音频质量。

V2TXLiveAudioQuality 枚举值

取值	说明
----	----

取值	说明
V2TXLiveAudioQualitySpeech	语音音质。采样率：16k、单声道、音频码率：16kbps。 适合语音通话为主的场景，例如在线会议，语音通话。
V2TXLiveAudioQualityDefault	默认音质。采样率：48k、单声道、音频码率：50kbps。 SDK 默认的音频质量，如无特殊需求推荐选择之。
V2TXLiveAudioQualityMusic	音乐音质。采样率：48k、双声道 + 全频带、音频码率：128kbps。 适合需要高保真传输音乐的场景，例如 K 歌、音乐直播等。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_REFUSED：推流过程中，不允许调整音质。

enableVolumeEvaluation

启用采集音量大小提示。

说明：

开启后可以在 `V2TXLivePusherObserver#onMicrophoneVolumeUpdate:(NSInteger)volume` 回调中获取到 SDK 对音量大小值的评估。

```
- (V2TXLiveCode)enableVolumeEvaluation:(NSInteger)intervalMs;
```

参数

参数	类型	含义
intervalMs	NSInteger	决定了 onMicrophoneVolumeUpdate 音量大小回调的触发间隔，单位为 ms，最小间隔为 100ms，如果小于等于0则会关闭回调，建议设置为 300ms。默认值：0，不开启。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

音效相关接口

getAudioEffectManager

获取音效管理对象 [TXAudioEffectManager](#)。

通过音效管理，您可以使用以下功能：

- 调整麦克风收集的人声音量。
- 设置混响和变声效果。
- 开启耳返，并设置耳返的音量。
- 添加背景音乐，调整背景音乐的播放效果。

```
- (TXAudioEffectManager *)getAudioEffectManager
```

设备管理相关接口

getDeviceManager

获取设备管理对象 [TXDeviceManager](#)。

通过设备管理，您可以使用以下功能：

- 切换前后摄像头。
- 设置自动聚焦。
- 设置摄像头缩放倍数。
- 打开或关闭闪光灯。

```
- (TXDeviceManager *)getDeviceManager;
```

更多实用接口

setProperty

调用 V2TXLivePusher 的高级 API 接口。

```
- (V2TXLiveCode) setProperty:(NSString *)key value:(NSObject *)value
```

参数

参数	类型	含义
key	NSString *	高级 API 对应的 key。
value	NSObject *	调用 key 所对应的高级 API 时，需要的参数。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，key 不允许为空。

setMixTranscodingConfig

设置云端的混流转码参数。如果您在 [实时音视频控制台](#) 中的功能配置页开启了“启用旁路推流”功能，每一路画面都会有一个默认的直播 [CDN 地址](#)。一个直播间中可能有不止一位主播，而且每个主播都有自己的画面和声音，但对于 CDN 观众来说，他们只需要一路直播流，所以您需要将多路音视频流混成一路标准的直播流，这就需要混流转码。

```
- (V2TXLiveCode) setMixTranscodingConfig:(V2TXLiveTranscodingConfig *)config
```

参数

参数	类型	含义
config	V2TXLiveTranscodingConfig *	云端混流（转码）配置。

V2TXLiveTranscodingConfig

取值	说明
V2TXLiveBufferTypeUnknown	未知。
V2TXLiveBufferTypeByteBuffer	DirectBuffer，装载 I420 等 buffer，在 native 层使用。
V2TXLiveBufferTypeByteArray	byte[]，装载 I420 等 buffer，在 Java 层使用。
V2TXLiveBufferTypeTexture	直接操作纹理 ID，性能最好，画质损失最少。

showDebugView

显示仪表盘。

```
- (void)showDebugView:(BOOL)isShow
```

参数

参数	类型	含义
isShow	BOOL	是否显示，默认值：NO。

V2TXLivePusherObserver

最近更新时间：2022-10-13 11:40:06

功能

腾讯云直播的推流的回调通知。

介绍

可以接收 [V2TXLivePusher](#) 推流器的一些推流通知，包括推流器连接状态、音视频首帧回调、统计数据、警告和错误信息等。

SDK 基础回调

onError

直播推流器错误通知，推流出现错误时，会回调该通知。

```
- (void)onError:(V2TXLiveCode)code message:(NSString *)msg extraInfo:(NSDictionary *)extraInfo
```

参数

参数	类型	含义
code	V2TXLiveCode	错误码。
msg	NSString *	错误信息。
extraInfo	NSDictionary *	扩展信息。

onWarning

直播推流器警告通知。

```
- (void)onWarning:(V2TXLiveCode)code message:(NSString *)msg extraInfo:(NSDictionary *)extraInfo
```

参数

参数	类型	含义
----	----	----

参数	类型	含义
code	V2TXLiveCode	警告码。
msg	NSString *	警告信息。
extraInfo	NSDictionary *	扩展信息。

视频相关回调

onPushStatusUpdate

直播推流器连接状态回调通知。

```
- (void)onPushStatusUpdate:(V2TXLivePushStatus) status
message:(NSString *)msg
extraInfo:(NSDictionary *)extraInfo
```

参数

参数	类型	含义
status	V2TXLivePushStatus	状态码。
msg	NSString *	连接状态信息。
extraInfo	NSDictionary *	扩展信息。

V2TXLivePushStatus 枚举值

取值	含义
V2TXLivePushStatusDisconnected	与服务器断开连接。
V2TXLivePushStatusConnecting	正在连接服务器。
V2TXLivePushStatusConnectSuccess	连接服务器成功。
V2TXLivePushStatusReconnecting	重连服务器中。

onSnapshotComplete

截图回调。

```
- (void)onSnapshotComplete:(TXImage *)image
```

参数

参数	类型	含义
image	TXImage *	已截取的视频画面。

onProcessVideoFrame

自定义视频处理回调。

说明：

调用 `V2TXLivePusher#enableCustomVideoProcess:(BOOL)enable pixelFormat:(V2TXLivePixelFormat)pixelFormat bufferType:(V2TXLiveBufferType)bufferType` 开启自定义视频处理后，会收到这个回调通知。

```
- (void)onProcessVideoFrame:(V2TXLiveVideoFrame * _Nonnull)srcFrame dstFrame:(V2TXLiveVideoFrame * _Nonnull)dstFrame
```

参数

参数	类型	含义
srcFrame	V2TXLiveVideoFrame *	用于承载未处理的视频画面。
dstFrame	V2TXLiveVideoFrame *	用于承载处理过的视频画面。

onGLContextDestroyed

自定义视频处理 GL 环境销毁回调。

```
- (void)onGLContextDestroyed
```

onCaptureFirstVideoFrame

首帧视频采集完成的回调通知。

```
- (void)onCaptureFirstVideoFrame
```

音频相关回调

onCaptureFirstAudioFrame

首帧音频采集完成的回调通知。

```
- (void)onCaptureFirstAudioFrame
```

onMicrophoneVolumeUpdate

麦克风采集音量值回调。

```
- (void)onMicrophoneVolumeUpdate:(NSInteger) volume
```

统计回调

onStatisticsUpdate

直播推流器统计数据回调。

```
- (void)onStatisticsUpdate:(V2TXLivePusherStatistics *)statistics
```

参数

参数	类型	含义
statistics	V2TXLivePusherStatistics *	推流器统计数据。

混流回调

onSetMixTranscodingConfig

设置云端的混流转码参数的回调。

说明：

调用 `V2TXLivePusher#setMixTranscodingConfig:(V2TXLiveTranscodingConfig *)config` 设置云端混流转码参数后，会收到这个回调通知。

```
- (void)onSetMixTranscodingConfig:(V2TXLiveCode)code message:(NSString *)msg
```

参数	类型	含义
code	V2TXLiveCode	0 表示成功，其余值表示失败。
msg	NSString *	具体错误原因。

TXBeautyManager

最近更新时间：2022-10-13 11:40:06

美颜、美容、动效挂件类基础函数。

setBeautyStyle

设置美颜类型。

```
- (void) setBeautyStyle: (TXBeautyStyle) level
```

参数

参数	类型	含义
level	TXBeautyStyle	美颜风格，TXBeautyStyleSmooth表示光滑；TXBeautyStyleNature表示自然；TXBeautyStylePitu表示朦胧。

setFilter

设置指定素材滤镜特效。

```
- (void) setFilter: (TXImage *) image
```

说明：

滤镜图片一定要用 png 格式，Demo 用到的滤镜查找表图片位于
TXLiteAVDemo/Resource/Beauty/filter/FilterResource.bundle 中。

setSpecialRatio

设置滤镜浓度。

```
- (void) setFilterStrength: (float) strength
```

参数

参数	类型	含义
strength	float	取值范围0 - 1的浮点型数字，取值越大滤镜效果越明显，默认取值0.5。

介绍

在美女秀场等应用场景里，滤镜浓度的要求会比较高，以便更加突显主播的差异。我们默认的滤镜浓度是0.5，如果您觉得滤镜效果不明显，可以使用下面的接口进行调节。

setGreenScreenFile

设置绿幕背景视频（商业版有效，其它版本设置此参数无效）。

```
- (void)setGreenScreenFile:(NSString *)file
```

参数

参数	类型	含义
file	NSString *	视频文件路径。支持 MP4；nil 表示关闭特效。

介绍

此处的绿幕功能并非智能抠背，它需要被拍摄者的背后有一块绿色的幕布来辅助产生特效。

setBeautyLevel

设置美颜级别。

```
- (void)setBeautyLevel:(float) level
```

参数

参数	类型	含义
level	float	美颜级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setWhitenessLevel

设置美白级别。

```
- (void)setWhitenessLevel:(float) level
```

参数

参数	类型	含义
level	float	美白级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setRuddyLevel

设置红润级别。

```
- (void) setRuddyLevel: (float) level
```

参数

参数	类型	含义
level	float	红润级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setEyeScaleLevel

设置大眼级别（商业版有效，其它版本设置此参数无效）。

```
- (void) setEyeScaleLevel: (float) level
```

参数

参数	类型	含义
level	float	大眼级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setFaceSlimLevel

设置瘦脸级别（商业版有效，其它版本设置此参数无效）。

```
- (void) setFaceSlimLevel: (float) level
```

参数

参数	类型	含义
level	float	瘦脸级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setFaceVLevel

设置 V 脸级别（商业版有效，其它版本设置此参数无效）。

```
- (void) setFaceVLevel: (float) level
```

参数

参数	类型	含义
----	----	----

参数	类型	含义
level	float	V 脸级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setChinLevel

设置下巴拉伸或收缩（商业版有效，其它版本设置此参数无效）。

```
- (void)setChinLevel:(float)level
```

参数

参数	类型	含义
level	float	下巴拉伸或收缩级别，取值范围-9 - 9；0表示关闭，小于0表示收缩，大于0表示拉伸。

setFaceShortLevel

设置短脸级别（商业版有效，其它版本设置此参数无效）。

```
- (void)setFaceShortLevel:(float)level
```

参数

参数	类型	含义
level	float	短脸级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setNoseSlimLevel

设置瘦鼻级别（商业版有效，其它版本设置此参数无效）。

```
- (void)setNoseSlimLevel:(float)level
```

参数

参数	类型	含义
level	float	瘦鼻级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setEyeLightenLevel

设置亮眼（商用企业版有效，其它版本设置此参数无效）。

```
- (void) setEyeLightenLevel:(float) level
```

参数

参数	类型	含义
level	float	亮眼级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setToothWhitenLevel

设置白牙（商用企业版有效，其它版本设置此参数无效）。

```
- (void) setToothWhitenLevel:(float) level
```

参数

参数	类型	含义
level	float	白牙级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setWrinkleRemoveLevel

设置祛皱（商用企业版有效，其它版本设置此参数无效）。

```
- (void) setWrinkleRemoveLevel:(float) level
```

参数

参数	类型	含义
level	float	祛皱级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setPouchRemoveLevel

设置祛眼袋（商用企业版有效，其它版本设置此参数无效）。

```
- (void) setPouchRemoveLevel:(float) level
```

参数

参数	类型	含义
level	float	祛眼袋级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setSmileLinesRemoveLevel

设置祛法令纹（商用企业版有效，其它版本设置此参数无效）。

```
- (void)setSmileLinesRemoveLevel:(float)level
```

参数

参数	类型	含义
level	float	祛法令纹级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setForeheadLevel

设置发际线（商用企业版有效，其它版本设置此参数无效）。

```
- (void)setForeheadLevel:(float)level
```

参数

参数	类型	含义
level	float	发际线级别，取值范围0 - 9；0表示关闭，1 - 9值越大，发际线越向下移。

setEyeDistanceLevel

设置眼距（商用企业版有效，其它版本设置此参数无效）。

```
- (void)setEyeDistanceLevel:(float)level
```

参数

参数	类型	含义
level	float	眼距级别，取值范围0 - 9；0表示关闭，1 - 9值越大，眼间距越小。

setEyeAngleLevel

设置眼角（商用企业版有效，其它版本设置此参数无效）。

```
- (void)setEyeAngleLevel:(float)level
```

参数

参数	类型	含义
----	----	----

参数	类型	含义
level	float	眼角级别，取值范围0 - 9；0表示关闭，1 - 9值越大，外眼角越向上，内眼角越向下。

setMouthShapeLevel

设置嘴型（商用企业版有效，其它版本设置此参数无效）。

```
- (void)setMouthShapeLevel:(float)level
```

参数

参数	类型	含义
level	float	嘴型级别，取值范围0 - 9；0表示关闭，1 - 9值越大，嘴越小。

setNoseWingLevel

设置鼻翼（商用企业版有效，其它版本设置此参数无效）。

```
- (void)setNoseWingLevel:(float)level
```

参数

参数	类型	含义
level	float	鼻翼级别，取值范围0 - 9；0表示关闭，1 - 9值越大，鼻翼越小。

setNosePositionLevel

设置鼻子位置（商用企业版有效，其它版本设置此参数无效）。

```
- (void)setNosePositionLevel:(float)level
```

参数

参数	类型	含义
level	float	鼻子位置级别，取值范围0 - 9；0表示关闭，1 - 9值越大，鼻子位置越向下移。

setLipsThicknessLevel

设置嘴唇厚度（商用企业版有效，其它版本设置此参数无效）。

```
- (void)setLipsThicknessLevel:(float)level
```

参数

参数	类型	含义
level	float	嘴唇厚度级别，取值范围0 - 9；0表示关闭，1 - 9值越大，嘴唇越厚。

setFaceBeautyLevel

设置脸型（商用企业版有效，其它版本设置此参数无效）。

```
- (void)setFaceBeautyLevel:(float)level
```

参数

参数	类型	含义
level	float	脸型级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setMotionTpl

选择使用哪一款 AI 动效挂件（商业版有效，其它版本设置此参数无效）。

```
- (void)setMotionTpl:(nullable NSString *)tplName inDir:(nullable NSString *)tplDir
```

参数

参数	类型	含义
tplDir	NSString *	动效所在目录。
tplName	NSString *	动效名称。

setMotionMute

设置动效静音（商业版有效，其它版本设置此参数无效）。有些挂件本身会有声音特效，通过此 API 可以关闭这些特效播放时所带的声音效果。

```
- (void)setMotionMute:(BOOL)motionMute
```

参数

| 参数 | 类型 | 含义 |

|-----|-----|-----|·

| motionMute | BOOL | YES表示静音；NO表示不静音。 |

拉流

V2TXLivePlayer

最近更新时间：2022-10-20 15:57:07

功能

腾讯云直播播放器。

主要负责从指定的直播流地址拉取音视频数据，并进行解码和本地渲染播放。

介绍

播放器包含如下能力：

- 支持 RTMP、HTTP-FLV、TRTC、WebRTC 协议。
- 屏幕截图，可以截取当前直播流的视频画面。
- 延时调节，可以设置播放器缓存自动调整的最小和最大时间。
- 自定义的视频数据处理，您可以根据项目需要处理直播流中的视频数据后，再进行渲染以及播放。

SDK 基础函数

setObserver

设置播放器回调。通过设置回调，可以监听 V2TXLivePlayer 播放器的一些回调事件，包括播放器状态、播放音量回调、音视频首帧回调、统计数据、警告和错误信息等。

```
- (void)setObserver:(id<V2TXLivePlayerObserver>) observer
```

参数

参数	类型	含义
observer	V2TXLivePlayerObserver	播放器的回调目标对象，详情请参见 V2TXLivePlayerObserver 。

播放基础接口

setRenderView

设置播放器渲染 View，该控件负责显示视频内容。

```
- (V2TXLiveCode) setRenderView: (TXView *)view
```

参数

参数	类型	含义
view	TXView *	播放器渲染 View。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startLivePlay

开始播放音视频流。

```
- (V2TXLiveCode) startLivePlay: (NSString *)url
```

说明：

10.7 版本开始，`startPlay` 变更为 `startLivePlay`，需要通过 `V2TXLivePremier#setLicence` 或者 `TXLiveBase#setLicence` 设置 Licence 后方可成功播放，否则将播放失败（黑屏），全局仅设置一次即可。直播 License、短视频 License 和视频播放 License 均可使用，若您暂未获取上述 License，可 [快速免费申请测试版 License](#) 以正常播放，正式版 License 需 [购买](#)。

参数

参数	类型	含义
url	NSString *	音视频流的播放地址，支持 RTMP、HTTP-FLV、TRTC、WebRTC 协议。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，URL 不合法。
- V2TXLIVE_ERROR_REFUSED：RTC 不支持同一设备上同时推拉同一个 StreamId。

stopPlay

停止播放音视频流。

```
- (V2TXLiveCode) stopPlay;
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

isPlaying

播放器是否正在播放中。

```
- (int) isPlaying
```

返回

是否正在播放。

- 1：正在播放中。
- 0：已经停止播放。

视频相关接口

setRenderRotation

设置播放器画面的旋转角度。

```
- (V2TXLiveCode) setRenderRotation:(V2TXLiveRotation) rotation
```

参数

参数	类型	含义
rotation	V2TXLiveRotation	旋转角度，默认值：0。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveRotation 枚举值

取值	含义
V2TXLiveRotation0	不旋转。
V2TXLiveRotation90	顺时针旋转90度。
V2TXLiveRotation180	顺时针旋转180度。
V2TXLiveRotation270	顺时针旋转270度。

setRenderFillMode

设置画面的填充模式。

```
– (V2TXLiveCode) setRenderFillMode:(V2TXLiveFillMode) mode
```

参数

参数	类型	含义
mode	V2TXLiveFillMode	画面填充模式，默认值：V2TXLiveFillModeFit。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveFillMode 枚举值

取值	含义
V2TXLiveFillModeFit	图像适应屏幕，保持画面完整，但如果图像宽高比不同于屏幕宽高比，会有黑边的存在。
V2TXLiveFillModeFill	图像铺满屏幕，不留黑边，如果图像宽高比不同于屏幕宽高比，部分画面内容会被裁剪掉。

pauseVideo

暂停播放器的视频流。

```
– (V2TXLiveCode) pauseVideo
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

resumeVideo

恢复播放器的视频流。

```
- (V2TXLiveCode) resumeVideo
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

snapshot

截取播放过程中的视频画面。

说明：

返回值成功后可以在 `[V2TXLivePlayerObserver onSnapshotComplete: image:]` 回调中获取截图图片。

```
- (V2TXLiveCode) snapshot
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_REFUSED：播放器处于停止状态，不允许调用截图操作。

enableCustomRendering

设置视频自定义渲染回调。

通过该方法，可以获取解码后的每一帧视频画面，进行自定义渲染处理，添加自定义显示效果。

说明：

开启成功后可在 `[V2TXLivePlayerObserver onRenderVideoFrame:frame:]` 回调中获取视频帧数据。

```
- (V2TXLiveCode)enableCustomRendering:(BOOL)enable
pixelFormat:(V2TXLivePixelFormat)pixelFormat
bufferType:(V2TXLiveBufferType)bufferType
```

参数

参数	类型	含义
enable	BOOL	是否开启自定义渲染。默认值：NO。
pixelFormat	V2TXLivePixelFormat	自定义渲染回调的视频像素格式。
bufferType	V2TXLiveBufferType	自定义渲染回调的视频数据格式。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_NOT_SUPPORTED：像素格式或者数据格式不支持。

V2TXLivePixelFormat 枚举值

取值	含义
V2TXLivePixelFormatUnknown	未知。
V2TXLivePixelFormatI420	YUV420P I420。
V2TXLivePixelFormatNV12	YUV420SP NV12。
V2TXLivePixelFormatBGRA32	BGRA8888。
V2TXLivePixelFormatTexture2D	OpenGL 2D 纹理。

V2TXLiveBufferType 枚举值

取值	含义
V2TXLiveBufferTypeUnknown	未知。

取值	含义
V2TXLiveBufferTypePixelFormat	直接使用效率最高，iOS 系统提供了众多 API 获取或处理 PixelBuffer。
V2TXLiveBufferTypeNSData	会有一定的性能消耗，SDK 内部是直接处理 PixelBuffer 的，所以会存在 NSData 和 PixelBuffer 之间类型转换所产生的内存拷贝开销。
V2TXLiveBufferTypeTexture	直接操作纹理 ID，性能最好。

音频相关接口

pauseAudio

暂停播放器的音频流。

```
- (V2TXLiveCode) pauseAudio
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

resumeAudio

恢复播放器的音频流。

```
- (V2TXLiveCode) resumeAudio
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setPlayoutVolume

设置音量。

```
- (V2TXLiveCode) setPlayoutVolume:(NSInteger) volume;
```

参数

参数	类型	含义
volume	NSNumber	音量大小，取值范围0 - 100，默认值：100。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

enableVolumeEvaluation

启用播放音量大小提示。

说明：

开启后可以在 `[V2TXLivePlayerObserver onPlayoutVolumeUpdate:volume:]` 回调中获取到 SDK 对音量大小值的评估。

```
– (V2TXLiveCode)enableVolumeEvaluation:(NSNumber)intervalMs
```

参数

参数	类型	含义
intervalMs	NSNumber	<code>onPlayoutVolumeUpdate</code> 音量大小回调的触发间隔，单位为 ms，最小间隔为 100ms。如果小于等于0则会关闭回调，建议设置为 300ms。默认值：0，不开启。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

更多实用接口

setCacheParams

设置播放器缓存自动调整的最小和最大时间（单位：秒）。

```
– (V2TXLiveCode)setCacheParams:(CGFloat)minTime maxTime:(CGFloat)maxTime
```

参数

参数	类型	含义
minTime	CGFloat	缓存自动调整的最小时间，取值需要大于0。默认值：1。
maxTime	CGFloat	缓存自动调整的最大时间，取值需要大于0。默认值：5。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，minTime 和 maxTime 需要大于0。
- V2TXLIVE_ERROR_REFUSED：播放器处于播放状态，不支持修改缓存策略。

showDebugView

是否显示播放器状态信息的调试浮层。

```
- (void)showDebugView:(BOOL)isShow
```

参数

参数	类型	含义
isShow	BOOL	是否显示，默认值：NO。

V2TXLivePlayerObserver

最近更新时间：2022-10-13 11:40:06

功能

腾讯云直播的播放器回调通知。

介绍

可以接收 [V2TXLivePlayer](#) 播放器的一些回调通知，包括播放器状态、播放音量回调、音视频首帧回调、统计数据、警告和错误信息等。

SDK 基础回调

onError

直播播放器错误通知，播放器出现错误时，会回调该通知。

```
- (void)onError:(id<V2TXLivePlayer>)player  
code:(V2TXLiveCode) code  
message:(NSString *)msg  
extraInfo:(NSDictionary *)extraInfo
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
code	V2TXLiveCode	错误码。
msg	NSString *	错误信息。
extraInfo	NSDictionary *	扩展信息。

onWarning

直播播放器警告通知。

```
- (void)onWarning:(id<V2TXLivePlayer>)player  
code:(V2TXLiveCode) code
```

```
message:(NSString *)msg
extraInfo:(NSDictionary *)extraInfo
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
code	V2TXLiveCode	警告码。
msg	NSString *	警告码信息。
extraInfo	NSDictionary *	扩展信息。

onConnected

已经成功连接到服务器通知。

```
- (void)onConnected:(id<V2TXLivePlayer>)player
extraInfo:(NSDictionary *)extraInfo
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
extraInfo	NSDictionary *	扩展信息。

视频相关回调

onVideoPlaying

视频播放事件通知。

```
- (void)onVideoPlaying:(id<V2TXLivePlayer>)player
firstPlay:(BOOL)firstPlay
extraInfo:(NSDictionary *)extraInfo
```

参数

参数	类型	含义
----	----	----

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
firstPlay	BOOL	第一次播放标志。
extraInfo	NSDictionary *	扩展信息。

onVideoLoading

视频加载事件通知。

```
- (void)onVideoLoading:(id<V2TXLivePlayer>)player  
extraInfo:(NSDictionary *)extraInfo;
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
extraInfo	NSDictionary *	扩展信息。

onVideoResolutionChanged

直播播放器分辨率变化通知。

```
- (void)onVideoResolutionChanged:(id<V2TXLivePlayer>)player  
width:(NSInteger)width  
height:(NSInteger)height;
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
width	NSInteger	视频宽
height	NSInteger	视频高

onSnapshotComplete

截图回调。

```
- (void)onSnapshotComplete:(id<V2TXLivePlayer>)player image:(TXImage *)image
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
image	TXImage *	已截取的视频画面。

onRenderVideoFrame

自定义视频渲染回调。

说明：

调用 `[V2TXLivePlayer enableCustomRendering:pixelFormat:bufferType:]` 开启自定义渲染之后，会收到这个回调通知。

```
- (void)onRenderVideoFrame:(id<V2TXLivePlayer>)player  
frame:(V2TXLiveVideoFrame *)videoFrame
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
videoFrame	V2TXLiveVideoFrame *	视频帧数据。

音频相关回调

onAudioPlaying

音频播放事件通知。

```
- (void)onAudioPlaying:(id<V2TXLivePlayer>)player  
firstPlay:(BOOL)firstPlay  
extraInfo:(NSDictionary *)extraInfo;
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
firstPlay	BOOL	第一次播放标志。
extraInfo	NSDictionary *	扩展信息。

onAudioLoading

音频加载事件通知。

```
- (void)onAudioLoading:(id<V2TXLivePlayer>)player  
extraInfo:(NSDictionary *)extraInfo;
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
extraInfo	NSDictionary *	扩展信息。

onPlayoutVolumeUpdate

播放器音量大小回调。

```
- (void)onPlayoutVolumeUpdate:(id<V2TXLivePlayer>)player volume:(NSInteger)volume
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
volume	NSInteger	音量大小，取值范围：0 - 100。

统计回调

onStatisticsUpdate

直播播放器统计数据回调。

```
- (void)onStatisticsUpdate:(id<V2TXLivePlayer>)player
statistics:(V2TXLivePlayerStatistics *)statistics
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
statistics	V2TXLivePlayerStatistics	播放器统计数据。

示例

最近更新时间：2022-10-20 15:57:07

针对开发者的接入反馈的高频问题，腾讯云提供有直观易懂的 **API-Example** 工程，方便开发者可以快速的了解相关 API 的使用，欢迎使用。

工程地址

所属平台	GitHub 地址
iOS	Github

目录说明

在这个示例项目中包含了以下场景：

- **基础功能：**
 - [摄像头推流](#)
 - [录屏推流](#)
 - [直播拉流](#)
 - [连麦互动](#)
 - [连麦 PK](#)
- **进阶功能：**
 - [自定义视频采集](#)
 - [第三方美颜](#)
 - [RTC 连麦 + 超低延时播放](#)

说明：

目前的工程结构跟标准的 xcproject 工程在名称大小写上可能有略微的差异，主要目的是方便大家在网页上看到此工程时，名称意义更加清晰。

应用图示

示例主页	推流示例	拉流示例
Tencent Cloud MLVB API Example Basic Features Publishing from Camera Including rtmp push stream, rtc push stream function Publishing from Screen Including rtmp push stream, rtc push stream function Playback Including standard protocols rtmp, flv, hls streaming and rtc streaming Webrtc Playback Co-anchoring Competition Advanced Features Custom Video Capturing Third-Party Beauty Filters RTC Co-anchoring + Ultra-low-latency Playback Picture in Picture Webrtc Auto Bitrate Ultra-Low Latency	MLVB Publishing (Camera) Stream ID 351242 Audio Quality Default Speech Music LVB Playback Tencent Cloud's proprietary RTC protocol features lower latency and better adaptability to poor network conditions when compared with traditional live streaming protocols, and can be used worldwide. For details, visit https://intl.cloud.tencent.com/document/product/1071/39888. Currently only supported in China, continuing to develop. RTC Publishing	MLVB Playback Stream ID 517375 Tencent Cloud's proprietary RTC protocol features low latency and strong adaptability to poor network conditions, and can be used worldwide. It is suitable for application scenarios such as e-commerce streaming and sports streaming. For details, visit https://intl.cloud.tencent.com/document/product/1071/39888. Currently only supported in China, other regions are continuing to develop. RTMP Playback FLV Playback HLS Playback RTC Playback

Android

概览

最近更新时间：2022-12-23 17:47:39

V2TXLivePlayer

视频播放器

请参见 [V2TXLivePlayer](#)。

主要负责从指定的直播流地址拉取音视频数据，并进行解码和本地渲染播放。

播放器包含如下能力：

- 支持 RTMP、HTTP-FLV、TRTC 以及 WebRTC 协议。
- 屏幕截图，可以截取当前直播流的视频画面。
- 延时调节，可以设置播放器缓存自动调整的最小和最大时间。
- 自定义的视频数据处理，您可以根据项目需要处理直播流中的视频数据后，再进行渲染以及播放。

SDK 基础函数

API	描述
setObserver	设置播放器回调。

播放基础接口

API	描述
setRenderView	设置播放器的视频渲染 TXCloudVideoView。
setRenderView	设置播放器的视频渲染 TextureView。
setRenderView	设置播放器的视频渲染 SurfaceView。
startLivePlay	10.7 版本开始， <code>startPlay</code> 变更为 <code>startLivePlay</code> ，需要通过 <code>V2TXLivePremier#setLicence</code> 或者 <code>TXLiveBase#setLicence</code> 设置 License 后方可成功播放，否则将播放失败（黑屏），全局仅设置一次即可。直播 License、短视频 License 和视频播放 License 均可使用，若您暂未获取上述 License，可 快速免费申请测试版 License 以正常播放，正式版 License 需 购买 。
stopPlay	停止播放。

API	描述
isPlaying	是否正在播放。

视频相关接口

API	描述
setRenderRotation	设置播放器画面的旋转角度。
setRenderFillMode	设置画面的填充模式。
pauseVideo	暂停播放器的视频流。
resumeVideo	恢复播放器的视频流。
snapshot	截取播放过程中的视频画面。
enableObserveVideoFrame	设置视频自定义渲染回调。

音频相关接口

API	描述
pauseAudio	暂停播放器的音频流。
resumeAudio	恢复播放器的音频流。
setPlayoutVolume	设置音量。
enableVolumeEvaluation	启用播放音量大小提示。

更多实用接口

API	描述
setCacheParams	设置播放器缓存自动调整的最小和最大时间 (单位：秒)。
showDebugView	是否显示播放器状态信息的调试浮层。

V2TXLivePlayerObserver

腾讯云直播播放的回调通知

SDK 基础回调

API	描述
onError	直播播放器错误通知，播放器出现错误时，会回调该通知。
onWarning	直播播放器警告通知。
onConnected	已经成功连接到服务器通知。

视频相关回调

API	描述
onVideoResolutionChanged	直播播放器分辨率变化通知。
onVideoLoading	视频加载事件。
onVideoPlaying	视频播放事件。
onSnapshotComplete	截图回调。
onRenderVideoFrame	自定义视频渲染回调。

音频相关回调

API	描述
onAudioLoading	音频加载事件。
onAudioPlaying	音频播放事件。
onPlayoutVolumeUpdate	播放器音量大小回调。

统计回调

API	描述
onStatisticsUpdate	直播播放器统计数据回调。

V2TXLivePusher

直播推流类

请参见 [V2TXLivePusher](#)。

主要负责将本地的音频和视频画面进行编码，并推送到指定的推流地址，支持任意的推流服务端。

推流器包含如下能力：

- 自定义的视频采集，让您可以根据项目需要定制自己的音视频数据源。
- 美颜、滤镜、贴纸，包含多套美颜磨皮算法（自然&光滑）和多款色彩空间滤镜（支持自定义滤镜）。
- Qos 流量控制技术，具备上行网络自适应能力，可以根据主播端网络的具体情况实时调节音视频数据量。
- 脸形调整、动效挂件，支持基于优图 AI 人脸识别技术的大眼、瘦脸、隆鼻等脸形微调以及动效挂件效果，只需要购买 **优图 License** 就可以轻松实现丰富的直播效果。

SDK 基础函数

API	描述
setObserver	设置推流器回调。

推流基础接口

API	描述
setRenderView	设置本地摄像头预览 View。
setRenderView	设置本地摄像头预览 View。
setRenderView	设置本地摄像头预览 View。
startPush	开始音视频数据推流。
stopPush	停止推送音视频数据。
isPushing	当前推流器是否正在推流中。

视频相关接口

API	描述
setVideoQuality	设置推流视频编码参数。
setRenderRotation	设置本地摄像头预览画面的旋转角度。
setRenderMirror	设置本地摄像头预览镜像。
startCamera	打开本地摄像头。
stopCamera	关闭本地摄像头。

API	描述
startVirtualCamera	开启图片推流。
stopVirtualCamera	关闭图片推流。
startScreenCapture	开启屏幕采集。
stopScreenCapture	关闭屏幕采集。
snapshot	截取推流过程中的本地画面。
setWatermark	设置推流器水印。默认情况下，水印不开启。
setEncoderMirror	设置视频编码镜像。
enableCustomVideoCapture	开启/关闭自定义视频采集。
sendCustomVideoFrame	在自定义视频采集模式下，将采集的视频数据发送到 SDK。
enableCustomVideoProcess	开启/关闭自定义视频处理。
sendSeiMessage	发送 SEI 消息。

美颜相关接口

API	描述
getBeautyManager	获取美颜管理对象 TXBeautyManager ，美颜的设置通过 TXBeautyManager 来设置。

音频相关接口

API	描述
startMicrophone	打开麦克风。
stopMicrophone	关闭麦克风。
setAudioQuality	设置推流音频质量。
enableVolumeEvaluation	启用采集音量大小提示。

音效相关接口

API	描述
-----	----

API	描述
getAudioEffectManager	获取音效管理对象。

设备管理相关接口

API	描述
getDeviceManager	获取设备管理对象。

更多实用接口

API	描述
setProperty	调用 V2TXLivePusher 的高级 API 接口。
setMixTranscodingConfig	设置云端的混流转码参数。
showDebugView	显示仪表盘。

V2TXLivePusherObserver

SDK 基础回调

API	描述
onError	直播推流器错误通知，推流器出现错误时，会回调该通知。
onWarning	直播推流器警告通知。

视频相关回调

API	描述
onPushStatusUpdate	推流器连接状态回调通知。
onSnapshotComplete	截图回调。
onProcessVideoFrame	自定义视频处理。
onGLContextCreated	SDK 内部的 OpenGL 环境的创建通知。
onGLContextDestroyed	SDK 内部的 OpenGL 环境的销毁通知。

API	描述
onCaptureFirstVideoFrame	首帧视频采集完成的回调通知。

音频相关回调

API	描述
onCaptureFirstAudioFrame	首帧音频采集完成的回调通知。
onMicrophoneVolumeUpdate	麦克风采集音量值回调。

混流回调

API	描述
onSetMixTranscodingConfig	设置云端的混流转码参数的回调。

统计回调

API	描述
onStatisticsUpdate	直播推流器统计数据回调。

推流

V2TXLivePusher

最近更新时间：2022-10-13 11:40:06

功能

腾讯云直播推流器。

介绍

主要负责将本地的音频和视频画面进行编码，并推送到指定的推流地址，支持任意的推流服务端。

推流器包含如下能力：

- 自定义的视频采集，让您可以根据项目需要定制自己的音视频数据源；
- 美颜、滤镜、贴纸，包含多套美颜磨皮算法（自然&光滑）和多款色彩空间滤镜（支持自定义滤镜）。
- Qos 流量控制技术，具备上行网络自适应能力，可以根据主播端网络的具体情况实时调节音视频数据量。
- 脸形调整、动效挂件，支持基于优图 AI 人脸识别技术的大眼、瘦脸、隆鼻等脸形微调以及动效挂件效果，只需要购买 **优图 License** 就可以轻松实现丰富的直播效果。

SDK 基础函数

setObserver

设置推流器回调。通过设置回调，可以监听 [V2TXLivePusher](#) 推流器的一些回调事件，包括推流器状态、音量回调、统计数据、警告和错误信息等。

```
public abstract void setObserver(V2TXLivePusherObserver observer);
```

参数

参数	类型	含义
observer	V2TXLivePusherObserver	推流器的回调目标对象，更多信息请参见 V2TXLivePusherObserver 。

推流基础接口

setRenderView

设置本地摄像头预览 View，本地摄像头采集到的画面，经过美颜、脸形调整、滤镜等多种效果叠加之后，最终会显示到传入的 View 上。

```
public abstract int setRenderView(TXCloudVideoView view);
```

参数

参数	类型	含义
view	TXCloudVideoView	本地摄像头预览 View。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setRenderView

设置本地摄像头预览 View，本地摄像头采集到的画面，经过美颜、脸形调整、滤镜等多种效果叠加之后，最终会显示到传入的 View 上。

```
public abstract int setRenderView(SurfaceView view);
```

参数

参数	类型	含义
view	SurfaceView	本地摄像头预览 View。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setRenderView

设置本地摄像头预览 View，本地摄像头采集到的画面，经过美颜、脸形调整、滤镜等多种效果叠加之后，最终会显示到传入的 View 上。

```
public abstract int setRenderView(TextureView view);
```

参数

参数	类型	含义
view	TextureView	本地摄像头预览 View。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startPush

开始音视频数据推流。

```
public abstract int startPush(String url);
```

参数

参数	类型	含义
url	String	推流的目标地址，支持任意推流服务端。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，URL 不合法。
- V2TXLIVE_ERROR_REFUSED：RTC 不支持同一设备上同时推拉同一个 StreamId。

stopPush

停止推送音视频数据。

```
public abstract int stopPush();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

isPushing

当前推流器是否正在推流中。

```
public abstract int isPushing();
```

返回

是否正在播放。

- 1：正在推流中。
- 0：已经停止推流。

视频相关接口

setVideoQuality

设置推流视频编码参数。

```
public abstract int setVideoQuality(V2TXLiveVideoEncoderParam param);
```

参数

参数	类型	含义
param	V2TXLiveVideoEncoderParam	视频编码参数。

setRenderRotation

设置本地摄像头预览画面的旋转角度。

```
public abstract int setRenderRotation(V2TXLiveRotation rotation);
```

参数

参数	类型	含义
----	----	----

参数	类型	含义
rotation	V2TXLiveRotation	旋转角度，默认值：V2TXLiveRotation0。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveRotation 枚举值

取值	含义
V2TXLiveRotation0	不旋转。
V2TXLiveRotation90	顺时针旋转90度。
V2TXLiveRotation180	顺时针旋转180度。
V2TXLiveRotation270	顺时针旋转270度。

setRenderMirror

设置本地摄像头预览镜像。

```
public abstract int setRenderMirror(V2TXLiveMirrorType mirrorType);
```

参数

参数	类型	含义
mirrorType	V2TXLiveMirrorType	摄像头镜像类型，默认值：V2TXLiveMirrorTypeAuto。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveMirrorType枚举值

取值	含义
----	----

取值	含义
V2TXLiveMirrorTypeAuto	默认镜像类型. 在这种情况下，前置摄像头的画面是镜像的，后置摄像头的画面不是镜像的。
V2TXLiveMirrorTypeEnable	前置摄像头和后置摄像头都切换为镜像模式。
V2TXLiveMirrorTypeDisable	前置摄像头和后置摄像头都切换为非镜像模式。

startCamera

打开本地摄像头。

说明：

startVirtualCamera、startCamera 和 startScreenCapture 同一 Pusher 实例下，仅有一个能上行，三者为覆盖关系。例如先调用 startCamera，后调用 startVirtualCamera。此时表现为暂停摄像头推流，开启图片推流。

```
public abstract int startCamera(boolean frontCamera);
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopCamera

关闭本地摄像头。

```
public abstract int stopCamera();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startVirtualCamera

开启图片推流。

说明：

startVirtualCamera、startCamera 和 startScreenCapture 同一 Pusher 实例下，仅有一个能上行，三者为覆盖关系。例如先调用 startCamera，后调用 startVirtualCamera。此时表现为暂停摄像头推流，开启图片推流。

```
public abstract int startVirtualCamera(Bitmap image);
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopVirtualCamera

关闭图片推流。

```
public abstract int stopVirtualCamera();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startScreenCapture

开启屏幕采集。

说明：

startVirtualCamera、startCamera 和 startScreenCapture 同一 Pusher 实例下，仅有一个能上行，三者为覆盖关系。例如先调用 startCamera，后调用 startVirtualCamera。此时表现为暂停摄像头推流，开启图片推流。

```
public abstract int startScreenCapture();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopScreenCapture

关闭屏幕采集。

```
public abstract int stopScreenCapture();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

snapshot

截取推流过程中的本地画面。

说明：

返回值成功后可以在 `V2TXLivePusherObserver.onSnapshotComplete` 回调中获取截图图片。

```
public abstract int snapshot();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_REFUSED：已经停止推流，不允许调用截图操作。

setWatermark

设置推流器水印。默认情况下，水印不开启。

```
public abstract int setWatermark(Bitmap image, float x, float y, float scale);
```

参数

参数	类型	含义
image	Bitmap	水印图片。如果该值为null，则等效于禁用水印。
x	float	水印的坐标。
y	float	水印的坐标。
scale	float	水印的缩放比例。

setEncoderMirror

设置视频编码镜像。

```
public abstract int setEncoderMirror(boolean mirror);
```

参数

参数	类型	含义
mirror	Boolean	是否镜像。默认值：false。

enableCustomVideoCapture

开启/关闭自定义视频采集。在自定义视频采集模式下，SDK 不再从摄像头采集图像，只保留编码和发送能力。

```
public abstract int enableCustomVideoCapture(boolean enable);
```

参数

参数	类型	含义
enable	Boolean	是否开启自定义采集。默认值：false。

sendCustomVideoFrame

在自定义视频采集模式下，将采集的视频数据发送到 SDK。

```
public abstract int sendCustomVideoFrame (V2TXLiveVideoFrame videoFrame);
```

参数

参数	类型	含义
videoFrame	V2TXLiveVideoFrame	向 SDK 发送的视频帧数据。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：发送失败，视频帧数据不合法。
- V2TXLIVE_ERROR_REFUSED：发送失败，您必须先调用 `enableCustomVideoCapture` 开启自定义视频采集。

enableCustomVideoProcess

开启/关闭自定义视频处理。

```
public abstract int enableCustomVideoProcess(boolean enable, V2TXLivePixelFormat pixelFormat, V2TXLiveBufferType bufferType);
```

参数

参数	类型	含义
enable	Boolean	是否开启自定义视频处理。默认值：false。
pixelFormat	V2TXLivePixelFormat	视频帧的像素格式。
bufferType	V2TXLiveBufferType	视频数据包装格式。

V2TXLivePixelFormat 枚举值

取值	说明
V2TXLivePixelFormatUnknown	未知。
V2TXLivePixelFormatI420	YUV420P I420。
V2TXLivePixelFormatTexture2D	OpenGL 2D 纹理。

V2TXLiveBufferType 枚举值

取值	说明
V2TXLiveBufferTypeUnknown	未知。
V2TXLiveBufferTypeByteBuffer	DirectBuffer，装载 I420 等 buffer，在 native 层使用。
V2TXLiveBufferTypeByteArray	byte[]，装载 I420 等 buffer，在 Java 层使用。
V2TXLiveBufferTypeTexture	直接操作纹理 ID，性能最好，画质损失最少。

sendSeiMessage

发送 SEI 消息。播放端 [V2TXLivePlayer](#) 通过 [V2TXLivePlayerObserver](#) 中的 `onReceiveSeiMessage` 回调来接收该消息。

```
public abstract int sendSeiMessage(int payloadType, byte[] data);
```

参数

参数	类型	含义
payloadType	int	数据类型，支持 5、242。推荐填：242
data	byte[]	待发送的数据

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

美颜相关接口

getBeautyManager

获取美颜管理对象 [TXBeautyManager](#)。

通过美颜管理，您可以使用以下功能：

- 设置“美颜风格”、“美白”、“红润”、“大眼”、“瘦脸”、“V脸”、“下巴”、“短脸”、“小鼻”、“亮眼”、“白牙”、“祛眼袋”、“祛皱纹”、“祛法令纹”等美容效果。

- 调整“发际线”、“眼间距”、“眼角”、“嘴形”、“鼻翼”、“鼻子位置”、“嘴唇厚度”、“脸型”。
- 设置人脸挂件（素材）等动态效果。
- 添加美妆。
- 进行手势识别。

```
public TXBeautyManager getBeautyManager()
```

音频相关接口

startMicrophone

打开麦克风。

```
public abstract int startMicrophone();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopMicrophone

关闭麦克风。

```
public abstract int stopMicrophone();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setAudioQuality

设置推流音频质量。

```
public abstract int setAudioQuality(V2TXLiveAudioQuality quality);
```

参数

参数	类型	含义
quality	V2TXLiveAudioQuality	音频质量。

V2TXLiveAudioQuality 枚举值

取值	说明
V2TXLiveAudioQualitySpeech	语音音质。采样率：16k、单声道、音频码率：16kbps。 适合语音通话为主的场景，例如在线会议，语音通话。
V2TXLiveAudioQualityDefault	默认音质。采样率：48k、单声道、音频码率：50kbps。 SDK 默认的音频质量，如无特殊需求推荐选择之。
V2TXLiveAudioQualityMusic	音乐音质。采样率：48k、双声道 + 全频带、音频码率：128kbps。 适合需要高保真传输音乐的场景，例如 K 歌、音乐直播等。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_REFUSED：推流过程中，不允许调整音质。

enableVolumeEvaluation

启用采集音量大小提示。

说明：

开启后可以在 `V2TXLivePusherObserver#onMicrophoneVolumeUpdate(int)` 回调中获取到 SDK 对音量大小值的评估。

```
public abstract int enableVolumeEvaluation(int intervalMs);
```

参数

参数	类型	含义
----	----	----

参数	类型	含义
intervalMs	int	决定了 onMicrophoneVolumeUpdate 音量大小回调的触发间隔，单位为 ms，最小间隔为 100ms，如果小于等于0则会关闭回调，建议设置为 300ms。默认值：0，不开启。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

音效相关接口

getAudioEffectManager

获取音效管理对象 [TXAudioEffectManager](#)。

通过音效管理，您可以使用以下功能：

- 调整麦克风收集的人声音量。
- 设置混响和变声效果。
- 开启耳返，并设置耳返的音量。
- 添加背景音乐，调整背景音乐的播放效果。

```
public TXAudioEffectManager getAudioEffectManager ()
```

设备管理相关接口

getDeviceManager

获取设备管理对象 [TXDeviceManager](#)。

通过设备管理，您可以使用以下功能：

- 切换前后摄像头。
- 设置自动聚焦。
- 设置摄像头缩放倍数。
- 打开或关闭闪光灯。

```
public abstract TXDeviceManager getDeviceManager();
```

更多实用接口

setProperty

调用 V2TXLivePusher 的高级 API 接口。

```
public abstract int setProperty(String key, Object value);
```

参数

参数	类型	含义
key	String	高级 API 对应的 key。
value	Object	调用 key 所对应的高级 API 时，需要的参数。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，key 不允许为空。

setMixTranscodingConfig

设置云端的混流转码参数。如果您在 [实时音视频控制台](#) 中的功能配置页开启了“启用旁路推流”功能，每一路画面都会有一个默认的直播 [CDN 地址](#)。一个直播间中可能有不止一位主播，而且每个主播都有自己的画面和声音，但对于 CDN 观众来说，他们只需要一路直播流，所以您需要将多路音视频流混成一路标准的直播流，这就需要混流转码。

```
public abstract int setMixTranscodingConfig(V2TXLiveTranscodingConfig config);
```

参数

参数	类型	含义
config	V2TXLiveTranscodingConfig	云端混流（转码）配置。

V2TXLiveTranscodingConfig

取值	说明
V2TXLiveBufferTypeUnknown	未知。
V2TXLiveBufferTypeByteBuffer	DirectBuffer，装载 I420 等 buffer，在 native 层使用。
V2TXLiveBufferTypeByteArray	byte[]，装载 I420 等 buffer，在 Java 层使用。
V2TXLiveBufferTypeTexture	直接操作纹理 ID，性能最好，画质损失最少。

showDebugView

显示仪表盘。

```
public abstract void showDebugView(boolean isShow);
```

参数

参数	类型	含义
isShow	boolean	是否显示，默认值：false。

V2TXLivePusherObserver

最近更新时间：2022-10-13 11:40:07

功能

腾讯云直播的推流的回调通知。

介绍

可以接收 [V2TXLivePusher](#) 推流器的一些推流通知，包括推流器连接状态、音视频首帧回调、统计数据、警告和错误信息等。

SDK 基础回调

onError

直播推流器错误通知，推流出现错误时，会回调该通知。

```
public void onError(int code, String msg, Bundle extraInfo)
```

参数

参数	类型	含义
code	int	错误码。
msg	String	错误信息。
extraInfo	Bundle	扩展信息。

onWarning

直播推流器警告通知。

```
public void onWarning(int code, String msg, Bundle extraInfo)
```

参数

参数	类型	含义
code	int	警告码。

参数	类型	含义
msg	String	警告码信息。
extraInfo	Bundle	扩展信息。

视频相关回调

onPushStatusUpdate

直播推流器连接状态回调通知。

```
public void onPushStatusUpdate(V2TXLivePushStatus status, String msg, Bundle extraInfo)
```

参数

参数	类型	含义
status	V2TXLivePushStatus	状态码。
msg	String	连接状态信息。
extraInfo	Bundle	扩展信息。

V2TXLivePushStatus 枚举值

取值	含义
V2TXLivePushStatusDisconnected	与服务器断开连接。
V2TXLivePushStatusConnecting	正在连接服务器。
V2TXLivePushStatusConnectSuccess	连接服务器成功。
V2TXLivePushStatusReconnecting	重连服务器中。

onSnapshotComplete

截图回调。

```
public void onSnapshotComplete(Bitmap image)
```

参数

参数	类型	含义
image	Bitmap *	已截取的视频画面。

onProcessVideoFrame

自定义视频处理回调。

说明：

调用 `V2TXLivePusher#enableCustomVideoProcess(boolean, V2TXLiveDef.V2TXLivePixelFormat, V2TXLiveDef.V2TXLiveBufferType)` 开启自定义视频处理后，会收到这个回调通知。

```
public void onProcessVideoFrame(V2TXLiveVideoFrame srcFrame, V2TXLiveVideoFrame dstFrame)
```

参数

参数	类型	含义
srcFrame	V2TXLiveVideoFrame	用于承载未处理的视频画面。
dstFrame	V2TXLiveVideoFrame	用于承载处理过的视频画面。

onGLContextCreated

自定义视频处理 GL 环境创建回调。

```
public void onGLContextCreated()
```

onGLContextDestroyed

自定义视频处理GL环境销毁回调。

```
public void onGLContextDestroyed()
```

onCaptureFirstVideoFrame

首帧视频采集完成的回调通知。

```
public void onCaptureFirstVideoFrame()
```

音频相关回调

onCaptureFirstAudioFrame

首帧音频采集完成的回调通知。

```
public void onCaptureFirstAudioFrame()
```

onMicrophoneVolumeUpdate

麦克风采集音量值回调。

```
public void onMicrophoneVolumeUpdate(int volume)
```

统计回调

onStatisticsUpdate

直播推流器统计数据回调。

```
public void onStatisticsUpdate(V2TXLivePusherStatistics statistics)
```

参数

参数	类型	含义
statistics	V2TXLivePusherStatistics	推流器统计数据。

混流回调

onSetMixTranscodingConfig

设置云端的混流转码参数的回调。

说明：

调用

```
V2TXLivePusher#setMixTranscodingConfig(V2TXLiveDef.V2TXLiveTranscodingConfig)
```

设置云端混流转码参数后，会收到这个回调通知。

```
public void onSetMixTranscodingConfig(int code, String msg)
```

参数	类型	含义
code	int	0 表示成功，其余值表示失败。
msg	String	具体错误原因。

TXBeautyManager

最近更新时间：2022-10-13 11:40:07

美颜、美容、动效挂件类基础函数。

setBeautyStyle

设置美颜类型。

```
void setBeautyStyle(int beautyStyle)
```

参数

参数	类型	含义
beautyStyle	int	美颜风格，0表示光滑，1表示自然，2表示朦胧。

setFilter

设置指定素材滤镜特效。

```
void setFilter(Bitmap bmp)
```

参数

参数	类型	含义
bmp	Bitmap	滤镜图片。

说明：

滤镜图片一定要用 png 格式，demo 用到的滤镜查找表图片位于 app/src/main/res/drawable-xxhdpi/ 中。

setFilterStrength

设置滤镜浓度。

```
void setFilterStrength(float strength)
```

参数

参数	类型	含义
strength	float	取值范围0 - 1的浮点型数字，取值越大滤镜效果越明显，默认取值0.5。

介绍

在美女秀场等应用场景里，滤镜浓度的要求会比较高，以便更加突显主播的差异。我们默认的滤镜浓度是0.5，如果您觉得滤镜效果不明显，可以使用下面的接口进行调节。

setGreenScreenFile

设置绿幕背景视频（商业版有效，其它版本设置此参数无效）。

```
boolean setGreenScreenFile(String path)
```

参数

参数	类型	含义
path	String	视频文件路径。支持 MP4；null 表示关闭特效。

介绍

此处的绿幕功能并非智能抠背，它需要被拍摄者的背后有一块绿色的幕布来辅助产生特效。

setBeautyLevel

设置美颜级别。

```
void setBeautyLevel(int beautyLevel)
```

参数

参数	类型	含义
beautyLevel	int	美颜级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setWhitenessLevel

设置美白级别。

```
void setWhitenessLevel(int whitenessLevel)
```

参数

参数	类型	含义
whitenessLevel	int	美白级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setRuddyLevel

设置红润级别。

```
void setRuddyLevel(int ruddyLevel)
```

参数

参数	类型	含义
ruddyLevel	int	红润级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setEyeScaleLevel

设置大眼级别（商业版有效，其它版本设置此参数无效）。

```
void setEyeScaleLevel(int eyeScaleLevel)
```

参数

参数	类型	含义
eyeScaleLevel	int	大眼级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setFaceSlimLevel

设置瘦脸级别（商业版有效，其它版本设置此参数无效）。

```
void setFaceSlimLevel(int faceSlimLevel)
```

参数

参数	类型	含义
faceSlimLevel	int	瘦脸级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setFaceVLevel

设置 V 脸级别（商业版有效，其它版本设置此参数无效）。

```
void setFaceVLevel(int faceVLevel)
```

参数

参数	类型	含义
faceVLevel	int	V 脸级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setChinLevel

设置下巴拉伸或收缩（商业版有效，其它版本设置此参数无效）。

```
void setChinLevel(int chinLevel)
```

参数

参数	类型	含义
chinLevel	int	下巴拉伸或收缩级别，取值范围-9 - 9；0 表示关闭，小于0表示收缩，大于0表示拉伸。

setFaceShortLevel

设置短脸级别（商业版有效，其它版本设置此参数无效）。

```
void setFaceShortLevel(int faceShortlevel)
```

参数

参数	类型	含义
faceShortlevel	int	短脸级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setNoseSlimLevel

设置瘦鼻级别（商业版有效，其它版本设置此参数无效）。

```
void setNoseSlimLevel(int noseSlimLevel)
```

参数

参数	类型	含义
----	----	----

参数	类型	含义
noseSlimLevel	int	瘦鼻级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setEyeLightenLevel

设置亮眼（商用企业版有效，其它版本设置此参数无效）。

```
void setEyeLightenLevel(int eyeLightenLevel)
```

参数

参数	类型	含义
eyeLightenLevel	int	亮眼级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setToothWhitenLevel

设置白牙（商用企业版有效，其它版本设置此参数无效）。

```
void setToothWhitenLevel(int toothWhitenLevel)
```

参数

参数	类型	含义
toothWhitenLevel	int	白牙级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setWrinkleRemoveLevel

设置祛皱（商用企业版有效，其它版本设置此参数无效）。

```
void setWrinkleRemoveLevel(int wrinkleRemoveLevel)
```

参数

参数	类型	含义
wrinkleRemoveLevel	int	祛皱级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setPouchRemoveLevel

设置祛眼袋（商用企业版有效，其它版本设置此参数无效）。

```
void setPouchRemoveLevel(int pouchRemoveLevel)
```

参数

参数	类型	含义
pouchRemoveLevel	int	祛眼袋级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setSmileLinesRemoveLevel

设置祛法令纹（商用企业版有效，其它版本设置此参数无效）。

```
void setSmileLinesRemoveLevel(int smileLinesRemoveLevel)
```

参数

参数	类型	含义
smileLinesRemoveLevel	int	祛法令纹级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setForeheadLevel

设置发际线（商用企业版有效，其它版本设置此参数无效）。

```
void setForeheadLevel(int foreheadLevel)
```

参数

参数	类型	含义
foreheadLevel	int	发际线级别，取值范围0 - 9；0表示关闭，1 - 9值越大，发际线越向下移。

setEyeDistanceLevel

设置眼距（商用企业版有效，其它版本设置此参数无效）。

```
void setEyeDistanceLevel(int eyeDistanceLevel)
```

参数

参数	类型	含义
eyeDistanceLevel	int	眼距级别，取值范围0 - 9；0表示关闭，1 - 9值越大，眼间距越小。

setEyeAngleLevel

设置眼角（商用企业版有效，其它版本设置此参数无效）。

```
void setEyeAngleLevel(int eyeAngleLevel)
```

参数

参数	类型	含义
eyeAngleLevel	int	眼角级别，取值范围0 - 9；0表示关闭，1 - 9值越大，外眼角越向上，内眼角越向下。

setMouthShapeLevel

设置嘴型（商用企业版有效，其它版本设置此参数无效）。

```
void setMouthShapeLevel(int mouthShapeLevel)
```

参数

参数	类型	含义
mouthShapeLevel	int	嘴型级别，取值范围0 - 9；0表示关闭，1 - 9值越大，嘴越小。

setNoseWingLevel

设置鼻翼（商用企业版有效，其它版本设置此参数无效）。

```
void setNoseWingLevel(int noseWingLevel)
```

参数

参数	类型	含义
noseWingLevel	int	鼻翼级别，取值范围0 - 9；0表示关闭，1 - 9值越大，鼻翼越小。

setNosePositionLevel

设置鼻子位置（商用企业版有效，其它版本设置此参数无效）。

```
void setNosePositionLevel(int nosePositionLevel)
```

参数

参数	类型	含义
nosePositionLevel	int	鼻子位置级别，取值范围0 - 9；0表示关闭，1 - 9值越大，鼻子位置越向下移。

setLipsThicknessLevel

设置嘴唇厚度（商用企业版有效，其它版本设置此参数无效）。

```
void setLipsThicknessLevel(int lipsThicknessLevel)
```

参数

参数	类型	含义
lipsThicknessLevel	int	嘴唇厚度级别，取值范围0 - 9；0表示关闭，1 - 9值越大，嘴唇越厚。

setFaceBeautyLevel

设置脸型（商用企业版有效，其它版本设置此参数无效）。

```
void setFaceBeautyLevel(int faceBeautyLevel)
```

参数

参数	类型	含义
faceBeautyLevel	int	脸型级别，取值范围0 - 9；0表示关闭，1 - 9值越大，效果越明显。

setMotionTpl

选择使用哪一款 AI 动效挂件（商业版有效，其它版本设置此参数无效）。

```
void setMotionTpl(String motionPath)
```

参数

参数	类型	含义
motionPath	String	动效所在路径。

setMotionMute

设置动效静音（商业版有效，其它版本设置此参数无效）。有些挂件本身会有声音特效，通过此 API 可以关闭这些特效播放时所带的声音效果。

```
void setMotionMute(boolean motionMute)
```

参数

参数	类型	含义
motionMute	boolean	true表示静音；false表示不静音。

拉流

V2TXLivePlayer

最近更新时间：2022-10-20 15:57:07

功能

腾讯云直播播放器。

主要负责从指定的直播流地址拉取音视频数据，并进行解码和本地渲染播放。

介绍

播放器包含如下能力：

- 支持RTMP、HTTP-FLV、TRTC、WebRTC。
- 屏幕截图，可以截取当前直播流的视频画面。
- 延时调节，可以设置播放器缓存自动调整的最小和最大时间。
- 自定义的视频数据处理，您可以根据项目需要处理直播流中的视频数据后，再进行渲染以及播放。

SDK 基础函数

setObserver

设置播放器回调。通过设置回调，可以监听 V2TXLivePlayer 播放器的一些回调事件，包括播放器状态、播放音量回调、音视频首帧回调、统计数据、警告和错误信息等。

```
public abstract void setObserver(V2TXLivePlayerObserver observer);
```

参数

参数	类型	含义
observer	V2TXLivePlayerObserver	播放器的回调目标对象，更多详情请参见 V2TXLivePlayerObserver 。

播放基础接口

setRenderView

设置播放器渲染 View，该控件负责显示视频内容。

```
public abstract int setRenderView(TXCloudVideoView view);
```

参数

参数	类型	含义
view	TXCloudVideoView	播放器渲染 View。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setRenderView

设置播放器渲染 View，该控件负责显示视频内容。

```
public abstract int setRenderView(SurfaceView view);
```

参数

参数	类型	含义
view	SurfaceView	播放器渲染 View。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setRenderView

设置播放器渲染 View，该控件负责显示视频内容。

```
public abstract int setRenderView(TextureView view);
```

参数

参数	类型	含义
view	TextureView	播放器渲染 View。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startLivePlay

开始播放音视频流。

```
public abstract int startLivePlay(String url);
```

说明：

10.7 版本开始，`startPlay` 变更为 `startLivePlay`，需要通过 `V2TXLivePremier#setLicence` 或者 `TXLiveBase#setLicence` 设置 License 后方可成功播放，否则将播放失败（黑屏），全局仅设置一次即可。直播 License、短视频 License 和视频播放 License 均可使用，若您暂未获取上述 License，可 [快速免费申请测试版 License](#) 以正常播放，正式版 License 需 [购买](#)。

参数

参数	类型	含义
url	String	音视频流的播放地址，支持 RTMP、HTTP-FLV、TRTC、WebRTC 协议。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，URL 不合法。
- V2TXLIVE_ERROR_REFUSED：RTC 不支持同一设备上同时推拉同一个 StreamId。

stopPlay

停止播放音视频流。

```
public abstract int stopPlay();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

isPlaying

播放器是否正在播放中。

```
public abstract int isPlaying();
```

返回

是否正在播放。

- 1：正在播放中。
- 0：已经停止播放。

视频相关接口

setRenderRotation

设置播放器画面的旋转角度。

```
public abstract int setRenderRotation(V2TXLiveRotation rotation);
```

参数

参数	类型	含义
rotation	V2TXLiveRotation	旋转角度，默认值：V2TXLiveRotation0。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveRotation 枚举值

取值	含义
V2TXLiveRotation0	不旋转。
V2TXLiveRotation90	顺时针旋转90度。
V2TXLiveRotation180	顺时针旋转180度。

取值	含义
V2TXLiveRotation270	顺时针旋转270度。

setRenderFillMode

设置画面的填充模式。

```
public abstract int setRenderFillMode(V2TXLiveFillMode mode);
```

参数

参数	类型	含义
mode	V2TXLiveFillMode	画面填充模式，默认值：V2TXLiveFillModeFit。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveFillMode 枚举值

取值	含义
V2TXLiveFillModeFit	图像适应屏幕，保持画面完整，但如果图像宽高比不同于屏幕宽高比，会有黑边的存在。
V2TXLiveFillModeFill	图像铺满屏幕，不留黑边，如果图像宽高比不同于屏幕宽高比，部分画面内容会被裁掉。

pauseVideo

暂停播放器的视频流。

```
public abstract int pauseVideo();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

resumeVideo

恢复播放器的视频流。

```
public abstract int resumeVideo();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

snapshot

截取播放过程中的视频画面。

说明：

返回值成功后可以在 `V2TXLivePlayerObserver.onSnapshotComplete` 回调中获取截图图片。

```
public abstract int snapshot();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_REFUSED：播放器处于停止状态，不允许调用截图操作。

enableObserveVideoFrame

设置视频自定义渲染回调。通过该方法，可以获取解码后的每一帧视频画面，进行自定义渲染处理，添加自定义显示效果。

说明：

开启成功后可在 `V2TXLivePlayerObserver.onRenderVideoFrame` 回调中获取视频帧数据。

```
public abstract int enableObserveVideoFrame(  
    boolean enable,  
    V2TXLivePixelFormat pixelFormat,  
    V2TXLiveBufferType bufferType);
```

参数

参数	类型	含义
enable	Boolean	是否开启自定义渲染。默认值：false。
pixelFormat	V2TXLivePixelFormat	自定义渲染回调的视频像素格式。
bufferType	V2TXLiveBufferType	自定义渲染回调的视频数据格式。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_NOT_SUPPORTED：像素格式或者数据格式不支持。

V2TXLivePixelFormat 枚举值

取值	说明
V2TXLivePixelFormatUnknown	未知。
V2TXLivePixelFormatI420	YUV420P I420。
V2TXLivePixelFormatTexture2D	OpenGL 2D 纹理。

V2TXLiveBufferType 枚举值

取值	说明
V2TXLiveBufferTypeUnknown	未知。
V2TXLiveBufferTypeByteBuffer	DirectBuffer，装载 I420 等 buffer，在 native 层使用。
V2TXLiveBufferTypeByteArray	byte[]，装载 I420 等 buffer，在 Java 层使用。
V2TXLiveBufferTypeTexture	直接操作纹理 ID，性能最好，画质损失最少。

音频相关接口

pauseAudio

暂停播放器的音频流。

```
public abstract int pauseAudio();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

resumeAudio

恢复播放器的音频流。

```
public abstract int resumeAudio();
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setPlayoutVolume

设置音量。

```
public abstract int setPlayoutVolume(int volume);
```

参数

参数	类型	含义
volume	int	音量大小，取值范围0 - 100，默认值：100。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

enableVolumeEvaluation

启用播放音量大小提示。

说明：

开启后可以在 `V2TXLivePlayerObserver.onPlayoutVolumeUpdate` 回调中获取到 SDK 对音量大小值的评估。

```
public abstract int enableVolumeEvaluation(int intervalMs);
```

参数

参数	类型	含义
intervalMs	int	onPlayoutVolumeUpdate 音量大小回调的触发间隔，单位为 ms，最小间隔为 100ms，如果小于等于0则会关闭回调，建议设置为 300ms。默认值：0，不开启。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

更多实用接口

setCacheParams

设置播放器缓存自动调整的最小和最大时间（单位：秒）。

```
public abstract int setCacheParams(float minTime, float maxTime);
```

参数

参数	类型	含义
minTime	float	缓存自动调整的最小时间，取值需要大于0。默认值：1。
maxTime	float	缓存自动调整的最大时间，取值需要大于0。默认值：5。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，minTime 和 maxTime 需要大于0。
- V2TXLIVE_ERROR_REFUSED：播放器处于播放状态，不支持修改缓存策略。

showDebugView

是否显示播放器状态信息的调试浮层。

```
public abstract void showDebugView(boolean isShow);
```

参数

参数	类型	含义
isShow	boolean	是否显示，默认值：NO。

V2TXLivePlayerObserver

最近更新时间：2022-10-13 11:40:07

功能

腾讯云直播的播放器回调通知。

介绍

可以接收 [V2TXLivePlayer](#) 播放器的一些回调通知，包括播放器状态、播放音量回调、音视频首帧回调、统计数据、警告和错误信息等。

SDK 基础回调

onError

直播播放器错误通知，播放器出现错误时，会回调该通知。

```
public void onError(V2TXLivePlayer player, int code, String msg, Bundle extraInfo)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
code	int	错误码。
msg	String	错误信息。
extraInfo	Bundle	扩展信息。

onWarning

直播播放器警告通知。

```
public void onWarning(V2TXLivePlayer player, int code, String msg, Bundle extraInfo)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
code	int	警告码。
msg	String	警告码信息。
extraInfo	Bundle	扩展信息。

onConnected

已经成功连接到服务器通知。

```
public void onConnected(V2TXLivePlayer player, Bundle extraInfo)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
extraInfo	Bundle	扩展信息。

视频相关回调

onVideoResolutionChanged

直播播放器分辨率变化通知

```
public void onVideoResolutionChanged(V2TXLivePlayer player, int width, int height)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
width	int	视频宽。

参数	类型	含义
height	int	视频高。

onVideoLoading

视频加载事件。

```
public void onVideoLoading(V2TXLivePlayer player, Bundle extraInfo)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
extraInfo	Bundle	扩展信息。

onVideoPlaying

视频播放事件。

```
public void onVideoPlaying(V2TXLivePlayer player, boolean firstPlay, Bundle extraInfo)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
firstPlay	boolean	第一次播放标志
extraInfo	Bundle	扩展信息。

onSnapshotComplete

截图回调。

```
public void onSnapshotComplete(V2TXLivePlayer player, Bitmap image)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
image	Bitmap *	已截取的视频画面。

onRenderVideoFrame

自定义视频渲染回调。

说明：

调用 `[V2TXLivePlayer enableCustomRendering:pixelFormat:bufferType:]` 开启自定义渲染之后，会收到这个回调通知。

```
public void onRenderVideoFrame(V2TXLivePlayer player, V2TXLiveVideoFrame videoFrame)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
videoFrame	V2TXLiveVideoFrame	视频帧数据。

音频相关回调

onAudioLoading

音频加载事件。

```
public void onAudioLoading(V2TXLivePlayer player, Bundle extraInfo)
```

参数

参数	类型	含义
----	----	----

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
extraInfo	Bundle	扩展信息。

onAudioPlaying

音频播放事件。

```
public void onAudioPlaying(V2TXLivePlayer player, boolean firstPlay, Bundle extraInfo)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
firstPlay	boolean	第一次播放标志
extraInfo	Bundle	扩展信息。

onPlayoutVolumeUpdate

播放器音量大小回调。

```
public void onPlayoutVolumeUpdate(V2TXLivePlayer player, int volume)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
volume	int	音量大小，取值范围：0 - 100。

统计回调

onStatisticsUpdate

直播播放器统计数据回调。

```
public void onStatisticsUpdate(V2TXLivePlayer player, V2TXLivePlayerStatistics statistics)
```

参数

参数	类型	含义
player	V2TXLivePlayer	回调该通知的播放器对象。
statistics	V2TXLivePlayerStatistics	播放器统计数据。

示例

最近更新时间：2022-10-20 15:57:07

针对开发者的接入反馈的高频问题，腾讯云提供有直观易懂的 **API-Example** 工程，方便开发者可以快速的了解相关 API 的使用，欢迎使用。

工程地址

所属平台	GitHub 地址
Android	Github

目录说明

在这个示例项目中包含了以下场景：

- **基础功能：**
 - [摄像头推流](#)
 - [录屏推流](#)
 - [直播拉流](#)
 - [连麦互动](#)
 - [连麦 PK](#)
- **进阶功能：**
 - [动态切换渲染组件](#)
 - [自定义视频采集](#)
 - [第三方美颜](#)
 - [RTC 连麦 + 超低延时播放](#)

说明：

目前的工程结构跟标准的 **Android Studio** 工程在名称大小写上可能有略微的差异，主要目的是方便大家在网页上看到此工程时，名称意义更加清晰。

应用图示

示例主页	推流示例	拉流示例
5:17 Tencent Cloud MLVB API Example Basic Features Publishing from Camera Including rtmp push stream, rtc push stream function Publishing from Screen Including rtmp push stream, rtc push stream function Playback Including standard protocols rtmp, flv, hls streaming and rtc streaming Webrtc Playback Co-anchoring Competition Advanced Features Custom Video Capturing Third-Party Beauty Filters RTC Co-anchoring + Ultra-low-latency Playback Picture in Picture Webrtc Auto Bitrate Ultra-Low Latency	5:17 < Back MLVB Publishing (Camera) Stream ID 351242 Audio Quality Default Speech Music LVB Playback Tencent Cloud's proprietary RTC protocol features lower latency and better adaptability to poor network conditions when compared with traditional live streaming protocols, and can be used worldwide. For details, visit https://intl.cloud.tencent.com/document/product/1071/39888. Currently only supported in China, RTC Publishing	5:17 < Back MLVB Playback Stream ID 517375 Tencent Cloud's proprietary RTC protocol features low latency and strong adaptability to poor network conditions, and can be used worldwide. It is suitable for application scenarios such as e-commerce streaming and sports streaming. For details, visit https://intl.cloud.tencent.com/document/product/1071/39888. Currently only supported in China, other regions are continuing to develop. RTMP Playback FLV Playback HLS Playback RTC Playback

MLVB Publishing (Camera)

Stream ID

351242

Audio Quality

Default

Speech

Music

LVB Playback

Tencent Cloud's proprietary RTC protocol features lower latency and better adaptability to poor network conditions when compared with traditional live streaming protocols, and can be used worldwide. For details, visit <https://intl.cloud.tencent.com/document/product/1071/39888>. Currently only supported in China, other regions are continuing to develop.

RTC Publishing

MLVB Playback

Stream ID

517375

Tencent Cloud's proprietary RTC protocol features low latency and strong adaptability to poor network conditions, and can be used worldwide. It is suitable for application scenarios such as e-commerce streaming and sports streaming. For details, visit <https://intl.cloud.tencent.com/document/product/1071/39888>. Currently only supported in China, other regions are continuing to develop.

RTMP Playback

FLV Playback

HLS Playback

RTC Playback

Web

TXLivePusherObserver

最近更新时间：2022-10-13 11:40:07

腾讯云直播推流的回调通知，回调包括推流器状态，统计信息，警告以及错误信息。

onError

直播推流器错误通知，推流器出现错误时，会回调该通知。

```
onError(code: number, msg: string, extraInfo: Object): void;
```

参数

- `code`：错误码，请参见 [推流错误码](#)。
- `msg`：错误信息。
- `extraInfo`：扩展信息，目前未用到。

onWarning

直播推流器警告通知。

```
onWarning(code: number, msg: string, extraInfo: Object): void;
```

参数

- `code`：错误码，请参见 [推流错误码](#)。
- `msg`：错误信息。
- `extraInfo`：扩展信息，目前未用到。

说明

打开摄像头、麦克风、屏幕录制失败时返回的错误信息，可参见 [getUserMedia 异常](#) 和 [getDisplayMedia 异常](#)。

onCaptureFirstAudioFrame

首帧音频采集完成的回调通知。

```
onCaptureFirstAudioFrame(): void;
```

onCaptureFirstVideoFrame

首帧视频采集完成的回调通知。

```
onCaptureFirstVideoFrame(): void;
```

onPushStatusUpdate

推流器连接状态回调通知。

```
onPushStatusUpdate(status: number, msg: string, extraInfo: Object): void;
```

参数

- `status`：连接状态码，请参见 [推流状态码](#)。
- `msg`：连接状态信息。
- `extraInfo`：扩展信息，目前未用到。

onStatisticsUpdate

直播推流器统计数据回调，主要是 WebRTC 相关的统计数据，不同浏览器返回的数据可能不一致。

```
onStatisticsUpdate(statistics: Object): void;
```

参数

`statistics`：推流统计数据，请参见 [推流统计数据](#)。

推流错误码

[onError](#) 和 [onWarning](#) 中用到的错误码如下：

枚举值	数值	描述
TXLIVE_ERROR_WEBRTC_FAILED	-1	WebRTC 接口调用失败
TXLIVE_ERROR_REQUEST_FAILED	-2	请求服务器推流接口返回报错
TXLIVE_WARNING_CAMERA_START_FAILED	-1001	打开摄像头失败

枚举值	数值	描述
TXLIVE_WARNING_MICROPHONE_START_FAILED	-1002	打开麦克风失败
TXLIVE_WARNING_SCREEN_CAPTURE_START_FAILED	-1003	打开屏幕录制失败
TXLIVE_WARNING_MEDIA_FILE_START_FAILED	-1004	打开本地媒体文件失败
TXLIVE_WARNING_CAMERA_INTERRUPTED	-1005	摄像头被中断（设备被拔出或者权限被用户取消）
TXLIVE_WARNING_MICROPHONE_INTERRUPTED	-1006	麦克风被中断（设备被拔出或者权限被用户取消）
TXLIVE_WARNING_SCREEN_CAPTURE_INTERRUPTED	-1007	屏幕录制被中断（Chrome 浏览器单击自带的停止共享按钮）

推流状态码

[onPushStatusUpdate](#) 用到的状态码如下：

枚举值	数值	描述
TXLIVE_PUSH_STATUS_DISCONNECTED	0	与服务器断开连接
TXLIVE_PUSH_STATUS_CONNECTING	1	正在连接服务器
TXLIVE_PUSH_STATUS_CONNECT_SUCCESS	2	连接服务器成功
TXLIVE_PUSH_STATUS_RECONNECTING	3	重连服务器中

推流统计数据

通过 [onStatisticsUpdate](#) 回调接口可以监听直播推流过程中 WebRTC 相关的统计数据。SDK 会以一秒一次的频率统计数据，返回的对象基本结构如下：

字段	说明
timestamp	数据采样的时间，自1970年1月1日（UTC）起经过的毫秒数
video	视频流相关的数据
audio	音频流相关的数据

video

字段	说明
bitrate	视频码率，单位：bit/s
framesPerSecond	视频帧率
frameWidth	视频宽度，Firefox下不存在
frameHeight	视频高度，Firefox下不存在
framesEncoded	编码帧数
framesSent	发送帧数，Firefox下不存在
packetsSent	发送包数
nackCount	NACK（Negative ACKnowledgement）数
firCount	FIR（Full Intra Request），关键帧重传请求数
pliCount	PLI（Picture Loss Indication），视频帧丢失重传数
frameEncodeAvgTime	平均编码时间，单位：ms，Safari / Firefox 下不存在
packetSendDelay	数据包发送之前本地缓存的平均时间，单位：ms，Firefox 下不存在

audio

字段	说明
bitrate	音频码率，单位：bit/s
packetsSent	发送包数

Flutter

概览

最近更新时间：2022-10-20 15:57:08

V2TXLivePlayer

视频播放器

请参见 [V2TXLivePlayer](#)。

主要负责从指定的直播流地址拉取音视频数据，并进行解码和本地渲染播放。

播放器包含如下能力：

- 支持 RTMP、HTTP-FLV、TRTC 以及 WebRTC 协议。
- 屏幕截图，可以截取当前直播流的视频画面。
- 延时调节，可以设置播放器缓存自动调整的最小和最大时间。
- 自定义的视频数据处理，您可以根据项目需要处理直播流中的视频数据后，再进行渲染以及播放。

SDK 基础函数

API	描述
addListener	设置播放器回调。

播放基础接口

API	描述
setRenderViewID	设置播放器的视频渲染 ViewID。
startLivePlay	10.7 版本开始， <code>startPlay</code> 变更为 <code>startLivePlay</code> ，需要通过 <code>V2TXLivePremier#setLicence</code> 设置 License 后方可成功播放，否则将播放失败（黑屏），全局仅设置一次即可。直播 License、短视频 License 和视频播放 License 均可使用，若您暂未获取上述 License，可 快速免费申请测试版 License 以正常播放，正式版 License 需 购买
stopPlay	停止播放。
isPlaying	是否正在播放。

视频相关接口

API	描述
setRenderRotation	设置播放器画面的旋转角度。
setRenderFillMode	设置画面的填充模式。
pauseVideo	暂停播放器的视频流。
resumeVideo	恢复播放器的视频流。
snapshot	截取播放过程中的视频画面。

音频相关接口

API	描述
pauseAudio	暂停播放器的音频流。
resumeAudio	恢复播放器的音频流。
setPlayoutVolume	设置音量。
enableVolumeEvaluation	启用播放音量大小提示。

更多实用接口

API	描述
setCacheParams	设置播放器缓存自动调整的最小和最大时间 (单位：秒)。
showDebugView	是否显示播放器状态信息的调试浮层。

V2TXLivePlayerObserver

腾讯云直播播放的回调通知。

SDK 基础回调

API	描述
onError	直播播放器错误通知，播放器出现错误时，会回调该通知。
onWarning	直播播放器警告通知。

API	描述
onConnected	已经成功连接到服务器通知。

视频相关回调

API	描述
onVideoPlaying	视频播放事件通知。
onVideoLoading	视频加载事件通知。
onVideoResolutionChanged	直播播放器分辨率变化通知。
onSnapshotComplete	截图回调。
onRenderVideoFrame	自定义视频渲染回调。

音频相关回调

API	描述
onAudioPlaying	音频播放事件通知。
onAudioLoading	音频加载事件通知。
onPlayoutVolumeUpdate	播放器音量大小回调。

统计回调

API	描述
onStatisticsUpdate	直播播放器统计数据回调。

V2TXLivePusher

直播推流类

请参见 [V2TXLivePusher](#)。

主要负责将本地的音频和视频画面进行编码，并推送到指定的推流地址，支持任意的推流服务端。
推流器包含如下能力：

- 自定义的视频采集，让您可以根据项目需要定制自己的音视频数据源。

- 美颜、滤镜、贴纸，包含多套美颜磨皮算法（自然&光滑）和多款色彩空间滤镜（支持自定义滤镜）。
- Qos 流量控制技术，具备上行网络自适应能力，可以根据主播端网络的具体情况实时调节音视频数据量。
- 脸形调整、动效挂件，支持基于优图 AI 人脸识别技术的大眼、瘦脸、隆鼻等脸形微调以及动效挂件效果，只需要购买 **优图 License** 就可以轻松实现丰富的直播效果。

SDK 基础函数

API	描述
addListener	设置推流器回调。

推流基础接口

API	描述
setRenderViewID	设置本地摄像头预览 View。
startPush	开始音视频数据推流。
stopPush	停止推送音视频数据。
isPushing	当前推流器是否正在推流中。

视频相关接口

API	描述
setVideoQuality	设置推流视频编码参数。
setRenderRotation	设置本地摄像头预览画面的旋转角度。
setRenderMirror	设置本地摄像头预览镜像。
startCamera	打开本地摄像头。
stopCamera	关闭本地摄像头。
startVirtualCamera	开启图片推流。
stopVirtualCamera	关闭图片推流。
startScreenCapture	开启屏幕采集。
stopScreenCapture	关闭屏幕采集。
snapshot	截取推流过程中的本地画面。

API	描述
setWatermark	设置推流器水印。默认情况下，水印不开启。
setEncoderMirror	设置视频编码镜像。
enableCustomVideoCapture	开启/关闭自定义视频采集。
sendCustomVideoFrame	在自定义视频采集模式下，将采集的视频数据发送到 SDK。
enableCustomVideoProcess	开启/关闭自定义视频处理。
sendSeiMessage	发送 SEI 消息。

美颜相关接口

API	描述
getBeautyManager	getBeautyManager ，美颜的设置通过 TXBeautyManager 来设置。

音频相关接口

API	描述
startMicrophone	打开麦克风。
stopMicrophone	关闭麦克风。
setAudioQuality	设置推流音频质量。
enableVolumeEvaluation	启用采集音量大小提示。

音效相关接口

API	描述
getAudioEffectManager	获取音效管理对象。

设备管理相关接口

API	描述
getDeviceManager	获取设备管理对象。

更多实用接口

API	描述
setProperty	调用 V2TXLivePusher 的高级 API 接口。
setMixTranscodingConfig	设置云端的混流转码参数。
showDebugView	显示仪表盘。

V2TXLivePusherObserver

SDK 基础回调

API	描述
onError	直播推流器错误通知，推流器出现错误时，会回调该通知。
onWarning	直播推流器警告通知。

视频相关回调

API	描述
onPushStatusUpdate	推流器连接状态回调通知。
onSnapshotComplete	截图回调。
onProcessVideoFrame	自定义视频处理。
onGLContextDestroyed	SDK 内部的 OpenGL 环境的销毁通知。
onCaptureFirstVideoFrame	首帧视频采集完成的回调通知。

音频相关回调

API	描述
onCaptureFirstAudioFrame	首帧音频采集完成的回调通知。
onMicrophoneVolumeUpdate	麦克风采集音量值回调。

混流回调

API	描述
-----	----

API	描述
onSetMixTranscodingConfig	设置云端的混流转码参数的回调。

统计回调

API	描述
onStatisticsUpdate	直播推流器统计数据回调。

推流

V2TXLivePusher

最近更新时间：2022-10-20 15:57:08

功能

腾讯云直播推流器。

介绍

主要负责将本地的音频和视频画面进行编码，并推送到指定的推流地址，支持任意的推流服务端。

推流器包含如下能力：

- 自定义的视频采集，让您可以根据项目需要定制自己的音视频数据源。
- 美颜、滤镜、贴纸，包含多套美颜磨皮算法（自然&光滑）和多款色彩空间滤镜（支持自定义滤镜）。
- Qos 流量控制技术，具备上行网络自适应能力，可以根据主播端网络的具体情况实时调节音视频数据量。
- 脸形调整、动效挂件，支持基于优图 AI 人脸识别技术的大眼、瘦脸、隆鼻等脸形微调以及动效挂件效果，只需要购买 **优图 License** 就可以轻松实现丰富的直播效果。

SDK 基础函数

initWithLiveMode

初始化直播推流器。

```
V2TXLivePusher(V2TXLiveMode liveMode)
```

参数

参数	类型	含义
liveMode	V2TXLiveMode	推流协议类型：RTMP/ROOM 协议，默认值：RTMP。

addListener

设置推流器回调。通过设置回调，可以监听 V2TXLivePusher 推流器的一些回调事件，包括推流器状态、音量回调、统计数据、警告和错误信息等。

```
void addListener(V2TXLivePusherObserver func)
```

参数

参数	类型	含义
observer	V2TXLivePusherObserver	推流器的回调目标对象，更多信息请参见 V2TXLivePusherObserver 。

推流基础接口

setRenderViewID

设置本地摄像头预览 View，本地摄像头采集到的画面，经过美颜、脸形调整、滤镜等多种效果叠加之后，最终会显示到传入的 View 上。

```
Future<V2TXLiveCode> setRenderViewID(int viewID)
```

参数

参数	类型	含义
viewID	int	本地摄像头预览视图 V2TXLiveVideoWidget.onViewCreated(int viewID)。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startPush

开始音视频数据推流。

```
Future<V2TXLiveCode> startPush(String url)
```

参数

参数	类型	含义
url	String	推流的目标地址，支持任意推流服务端。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，URL 不合法。
- V2TXLIVE_ERROR_REFUSED：RTC 不支持同一设备上同时推拉同一个 StreamId。

stopPush

停止推送音视频数据。

```
Future<V2TXLiveCode> stopPush()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

isPushing

当前推流器是否正在推流中。

```
Future<V2TXLiveCode> isPushing()
```

返回

是否正在播放。

- 1：正在推流中。
- 0：已经停止推流。

视频相关接口

setVideoQuality

设置推流视频编码参数。

```
Future<V2TXLiveCode> setVideoQuality(V2TXLiveVideoEncoderParam param)
```

参数

参数	类型	含义
param	V2TXLiveVideoEncoderParam	视频编码参数。

setRenderRotation

设置本地摄像头预览画面的旋转角度。

```
Future<V2TXLiveCode> setRenderRotation(V2TXLiveRotation rotation)
```

参数

参数	类型	含义
rotation	V2TXLiveRotation	旋转角度，默认值：v2TXLiveRotation0。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveRotation 枚举值

取值	含义
v2TXLiveRotation0	不旋转。
v2TXLiveRotation90	顺时针旋转90度。
v2TXLiveRotation180	顺时针旋转180度。
v2TXLiveRotation270	顺时针旋转270度。

setRenderMirror

设置本地摄像头预览镜像。

```
Future<V2TXLiveCode> setRenderMirror(V2TXLiveMirrorType mirrorType)
```

参数

参数	类型	含义
mirrorType	V2TXLiveMirrorType	摄像头镜像类型，默认值：v2TXLiveMirrorTypeAuto。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveMirrorType枚举值

取值	含义
v2TXLiveMirrorTypeAuto	默认镜像类型. 在这种情况下，前置摄像头的画面是镜像的，后置摄像头的画面不是镜像的。
v2TXLiveMirrorTypeEnable	前置摄像头和后置摄像头都切换为镜像模式。
v2TXLiveMirrorTypeDisable	前置摄像头和后置摄像头都切换为非镜像模式。

startCamera

打开本地摄像头。

说明：

startVirtualCamera、startCamera 和 startScreenCapture 同一 Pusher 实例下，仅有一个能上行，三者为覆盖关系。例如先调用 startCamera，后调用 startVirtualCamera。此时表现为暂停摄像头推流，开启图片推流。

```
Future<V2TXLiveCode> startCamera (bool frontCamera)
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopCamera

关闭本地摄像头。

```
Future<void> stopCamera ()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startVirtualCamera

开启图片推流。

说明：

startVirtualCamera、startCamera 和 startScreenCapture 同一 Pusher 实例下，仅有一个能上行，三者为覆盖关系。例如先调用 startCamera，后调用 startVirtualCamera。此时表现为暂停摄像头推流，开启图片推流。

```
Future<V2TXLiveCode> startVirtualCamera(String type, String imageUrl)
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

参数	类型	含义
type	String	network(网络资源), file(本地资源)
imageUrl	String	资源地址

stopVirtualCamera

关闭图片推流。

```
Future<V2TXLiveCode> stopVirtualCamera()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startScreenCapture

开启屏幕采集。

说明：

- iOS 端暂不支持通过此接口开启屏幕采集。

- startVirtualCamera、startCamera 和 startScreenCapture 同一 Pusher 实例下，仅有一个能上行，三者为覆盖关系。例如先调用 startCamera，后调用 startVirtualCamera。此时表现为暂停摄像头推流，开启图片推流。

1. 需要通过 iOS Broadcast Upload Extension 来开启屏幕采集。
2. 然后设置 enableCustomVideoCapture 开启自定义采集支持。
3. 最后通过 sendCustomVideoFrame 把 Broadcast Upload Extension 中采集到的屏幕画面送出去。

```
Future<V2TXLiveCode> startScreenCapture(String appGroup)
```

参数

参数	类型	含义
appGroup	String	主 App 与 Broadcast 共享的 Application Group Identifier，可以指定为 nil，但按照文档设置会使功能更加可靠。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopScreenCapture

关闭屏幕采集。

```
Future<V2TXLiveCode> stopScreenCapture()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

snapshot

截取推流过程中的本地画面。

说明：

返回值成功后可以在 `V2TXLivePusherObserver.onSnapshotComplete` 回调中获取截图图片。

```
Future<V2TXLiveCode> snapshot()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_REFUSED：已经停止推流，不允许调用截图操作。

setWatermark

设置推流器水印。默认情况下，水印不开启。

```
Future<V2TXLiveCode> setWatermark(String type, String image, double x,  
double y, double scale)
```

参数

参数	类型	含义
type	String	network(网络资源), file(本地资源)
image	String	资源地址
x	double	水印的坐标。
y	double	水印的坐标。
scale	double	水印的缩放比例。

setEncoderMirror

设置视频编码镜像。

```
Future<V2TXLiveCode> setEncoderMirror(bool mirror)
```

参数

参数	类型	含义
mirror	bool	是否镜像。默认值：false。

enableCustomVideoCapture

开启/关闭自定义视频采集。在自定义视频采集模式下，SDK 不再从摄像头采集图像，只保留编码和发送能力。

```
Future<V2TXLiveCode> enableCustomVideoCapture(bool enable)
```

参数

参数	类型	含义
enable	bool	是否开启自定义采集。默认值：false。

sendCustomVideoFrame

在自定义视频采集模式下，将采集的视频数据发送到 SDK。

```
Future<V2TXLiveCode> sendCustomVideoFrame(V2TXLiveVideoFrame videoFrame)
```

参数

参数	类型	含义
videoFrame	V2TXLiveVideoFrame	向 SDK 发送的视频帧数据。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：发送失败，视频帧数据不合法。
- V2TXLIVE_ERROR_REFUSED：发送失败，您必须先调用 enableCustomVideoCapture 开启自定义视频采集。

enableCustomVideoProcess

开启/关闭自定义视频处理。

```
Future<V2TXLiveCode> enableCustomVideoProcess(bool enable,  
V2TXLivePixelFormat pixelFormat, V2TXLiveBufferType bufferType)
```

参数

参数	类型	含义
enable	bool	是否开启自定义视频处理。默认值：false。

参数	类型	含义
pixelFormat	V2TXLivePixelFormat	视频帧的像素格式。
bufferType	V2TXLiveBufferType	视频数据包装格式。

V2TXLivePixelFormat 枚举值

取值	说明
v2TXLivePixelFormatUnknown	未知。
v2TXLivePixelFormatI420	YUV420P I420。
v2TXLivePixelFormatTexture2D	OpenGL 2D 纹理。

V2TXLiveBufferType 枚举值

取值	说明
v2TXLiveBufferTypeUnknown	未知。
v2TXLiveBufferTypeByteBuffer	DirectBuffer，装载 I420 等 buffer，在 native 层使用。
v2TXLiveBufferTypeByteArray	byte[]，装载 I420 等 buffer，在 Java 层使用。
v2TXLiveBufferTypeTexture	直接操作纹理 ID，性能最好，画质损失最少。

sendSeiMessage

发送 SEI 消息。播放端 [V2TXLivePlayer](#) 通过 [V2TXLivePlayerObserver](#) 中的 `onReceiveSeiMessage` 回调来接收该消息。

```
Future<V2TXLiveCode> sendSeiMessage(int payloadType, Uint8List data)
```

参数

参数	类型	含义
payloadType	int	数据类型，支持 5、242。推荐填：242
data	Uint8List	待发送的数据

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

美颜相关接口

getBeautyManager

获取美颜管理对象 [TXBeautyManager](#)。

通过美颜管理，您可以使用以下功能：

- 设置“美颜风格”、“美白”、“红润”、“大眼”、“瘦脸”、“V脸”、“下巴”、“短脸”、“小鼻”、“亮眼”、“白牙”、“祛眼袋”、“祛皱纹”、“祛法令纹”等美容效果。
- 调整“发际线”、“眼间距”、“眼角”、“嘴形”、“鼻翼”、“鼻子位置”、“嘴唇厚度”、“脸型”。
- 设置人脸挂件（素材）等动态效果。
- 添加美妆。
- 进行手势识别。

```
TXBeautyManager getBeautyManager()
```

音频相关接口

startMicrophone

打开麦克风。

```
Future<V2TXLiveCode> startMicrophone()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

stopMicrophone

关闭麦克风。

```
Future<V2TXLiveCode> stopMicrophone()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setAudioQuality

设置推流音频质量。

```
Future<V2TXLiveCode> setAudioQuality(V2TXLiveAudioQuality quality)
```

参数

参数	类型	含义
quality	V2TXLiveAudioQuality	音频质量。

V2TXLiveAudioQuality 枚举值

取值	说明
v2TXLiveAudioQualitySpeech	语音音质。采样率：16k、单声道、音频码率：16kbps。 适合语音通话为主的场景，例如在线会议，语音通话。
v2TXLiveAudioQualityDefault	默认音质。采样率：48k、单声道、音频码率：50kbps。 SDK 默认的音频质量，如无特殊需求推荐选择之。
v2TXLiveAudioQualityMusic	音乐音质。采样率：48k、双声道 + 全频带、音频码率：128kbps。 适合需要高保真传输音乐的场景，例如 K 歌、音乐直播等。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_REFUSED：推流过程中，不允许调整音质。

enableVolumeEvaluation

启用采集音量大小提示。

说明：

开启后可以在 `V2TXLivePusherObserver.onMicrophoneVolumeUpdate` 回调中获取到 SDK 对音量大小值的评估。

```
Future<V2TXLiveCode> enableVolumeEvaluation(int intervalMs)
```

参数

参数	类型	含义
intervalMs	int	决定了 onMicrophoneVolumeUpdate 音量大小回调的触发间隔，单位为 ms，最小间隔为 100ms，如果小于等于0则会关闭回调，建议设置为 300ms。默认值：0，不开启。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

音效相关接口

getAudioEffectManager

获取音效管理对象 [TXAudioEffectManager](#)。

通过音效管理，您可以使用以下功能：

- 调整麦克风收集的人声音量。
- 设置混响和变声效果。
- 开启耳返，并设置耳返的音量。
- 添加背景音乐，调整背景音乐的播放效果。

```
TXAudioEffectManager getAudioEffectManager()
```

设备管理相关接口

getDeviceManager

获取设备管理对象 [TXDeviceManager](#)。

通过设备管理，您可以使用以下功能：

- 切换前后摄像头。
- 设置自动聚焦。
- 设置摄像头缩放倍数。
- 打开或关闭闪光灯。

```
TXDeviceManager getDeviceManager()
```

更多实用接口

setProperty

调用 V2TXLivePusher 的高级 API 接口。

```
Future<V2TXLiveCode> setProperty(String key, String value)
```

参数

参数	类型	含义
key	String	高级 API 对应的 key。
value	String	调用 key 所对应的高级 API 时，需要的参数。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，key 不允许为空。

setMixTranscodingConfig

设置云端的混流转码参数。如果您在 [实时音视频控制台](#) 中的功能配置页开启了“启用旁路推流”功能，每一路画面都会有一个默认的直播 [CDN 地址](#)。一个直播间中可能有不止一位主播，而且每个主播都有自己的画面和声音，但对于 CDN 观众来说，他们只需要一路直播流，所以您需要将多路音视频流混成一路标准的直播流，这就需要混流转码。

```
Future<V2TXLiveCode> setMixTranscodingConfig(V2TXLiveTranscodingConfig? config)
```

参数

参数	类型	含义
config	V2TXLiveTranscodingConfig	云端混流（转码）配置。

V2TXLiveTranscodingConfig

取值	说明
v2TXLiveBufferTypeUnknown	未知。
v2TXLiveBufferTypeByteBuffer	DirectBuffer，装载 I420 等 buffer，在 native 层使用。
v2TXLiveBufferTypeByteArray	byte[]，装载 I420 等 buffer，在 Java 层使用。
v2TXLiveBufferTypeTexture	直接操作纹理 ID，性能最好，画质损失最少。

showDebugView

显示仪表盘。

```
Future<V2TXLiveCode> showDebugView (bool isShow)
```

参数

参数	类型	含义
isShow	bool	是否显示，默认值：false。

V2TXLivePusherObserver

最近更新时间：2022-10-20 15:57:08

功能

腾讯云直播的推流的回调通知。

介绍

可以接收 [V2TXLivePusher](#) 推流器的一些推流通知，包括推流器连接状态、音视频首帧回调、统计数据、警告和错误信息等。

SDK 基础回调

onError

直播推流器错误通知，推流出现错误时，会回调该通知。

```
V2TXLivePusherListenerType.onError
```

参数

参数	类型	含义
code	V2TXLiveCode	错误码。
msg	String	错误信息。
extraInfo	Map	扩展信息。

onWarning

直播推流器警告通知。

```
V2TXLivePusherListenerType.onWarning
```

参数

参数	类型	含义
code	V2TXLiveCode	警告码。

参数	类型	含义
msg	String	警告信息。
extraInfo	Map	扩展信息。

视频相关回调

onPushStatusUpdate

直播推流器连接状态回调通知。

```
V2TXLivePusherListenerType.onPushStatusUpdate
```

参数

参数	类型	含义
status	V2TXLivePushStatus	状态码。
msg	String	连接状态信息。
extraInfo	Map	扩展信息。

V2TXLivePushStatus 枚举值

取值	含义
V2TXLivePushStatusDisconnected: 0	与服务器断开连接。
V2TXLivePushStatusConnecting: 1	正在连接服务器。
V2TXLivePushStatusConnectSuccess: 2	连接服务器成功。
V2TXLivePushStatusReconnecting: 3	重连服务器中。

onSnapshotComplete

截图回调。

```
V2TXLivePusherListenerType.onSnapshotComplete
```

参数

参数	类型	含义
image	Uint8List	已截取的视频画面。

onProcessVideoFrame

自定义视频处理回调。

说明：

调用 `V2TXLivePusher.enableCustomVideoProcess(bool enable, V2TXLivePixelFormat pixelFormat, V2TXLiveBufferType bufferType)` 开启自定义视频处理后，会收到这个回调通知。

```
V2TXLivePusherListenerType.onProcessVideoFrame
```

参数

参数	类型	含义
srcFrame	Map	用于承载未处理的视频画面。
dstFrame	Map	用于承载处理过的视频画面。

onGLContextDestroyed

自定义视频处理 GL 环境销毁回调。

```
V2TXLivePusherListenerType.onGLContextDestroyed
```

onCaptureFirstVideoFrame

首帧视频采集完成的回调通知。

```
V2TXLivePusherListenerType.onCaptureFirstVideoFrame
```

音频相关回调

onCaptureFirstAudioFrame

首帧音频采集完成的回调通知。

```
V2TXLivePusherListenerType.onCaptureFirstAudioFrame
```

onMicrophoneVolumeUpdate

麦克风采集音量值回调。

```
V2TXLivePusherListenerType.onMicrophoneVolumeUpdate
```

统计回调

onStatisticsUpdate

直播推流器统计数据回调。

```
V2TXLivePusherListenerType.onStatisticsUpdate
```

参数

参数	类型	含义
statistics	Map	推流器统计数据。

混流回调

onSetMixTranscodingConfig

设置云端的混流转码参数的回调。

说明：

调用 `V2TXLivePusher.setMixTranscodingConfig(V2TXLiveTranscodingConfig config)` 设置云端混流转码参数后，会收到这个回调通知。

```
V2TXLivePusherListenerType.onSetMixTranscodingConfig
```

参数	类型	含义
----	----	----

参数	类型	含义
code	V2TXLiveCode	0 表示成功，其余值表示失败。
msg	String	具体错误原因。

拉流

V2TXLivePlayer

最近更新时间：2022-10-20 15:57:08

功能

腾讯云直播播放器。

主要负责从指定的直播流地址拉取音视频数据，并进行解码和本地渲染播放。

介绍

播放器包含如下能力：

- 支持 RTMP、HTTP-FLV、TRTC、WebRTC 协议。
- 屏幕截图，可以截取当前直播流的视频画面。
- 延时调节，可以设置播放器缓存自动调整的最小和最大时间。
- 自定义的视频数据处理，您可以根据项目需要处理直播流中的视频数据后，再进行渲染以及播放。

SDK 基础函数

addListener

设置播放器回调。通过设置回调，可以监听 V2TXLivePlayer 播放器的一些回调事件，包括播放器状态、播放音量回调、音视频首帧回调、统计数据、警告和错误信息等。

```
void addListener(V2TXLivePlayerObserver func)
```

参数

参数	类型	含义
observer	V2TXLivePlayerObserver	播放器的回调目标对象，详情请参见 V2TXLivePlayerObserver 。

播放基础接口

setRenderViewID

设置播放器渲染 View，该控件负责显示视频内容。

```
Future<V2TXLiveCode> setRenderViewID(int viewID)
```

参数

参数	类型	含义
viewID	int	播放器渲染 V2TXLiveVideoWidget.onViewCreated viewId。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

startLivePlay

开始播放音视频流。

```
Future<V2TXLiveCode> startLivePlay(String url)
```

说明：

10.7 版本开始，`startPlay` 变更为 `startLivePlay`，需要通过 `V2TXLivePremier#setLicence` 或者 `TXLiveBase#setLicence` 设置 License 后方可成功播放，否则将播放失败（黑屏），全局仅设置一次即可。直播 License、短视频 License 和视频播放 License 均可使用，若您暂未获取上述 License，可 [快速免费申请测试版 License](#) 以正常播放，正式版 License 需 [购买](#)。

参数

参数	类型	含义
url	String	音视频流的播放地址，支持 RTMP、HTTP-FLV、TRTC、WebRTC 协议。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，URL 不合法。
- V2TXLIVE_ERROR_REFUSED：RTC 不支持同一设备上同时推拉同一个 StreamId。

stopPlay

停止播放音视频流。

```
Future<V2TXLiveCode> stopPlay()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

isPlaying

播放器是否正在播放中。

```
Future<int> isPlaying()
```

返回

是否正在播放。

- 1：正在播放中。
- 0：已经停止播放。

视频相关接口

setRenderRotation

设置播放器画面的旋转角度。

```
Future<V2TXLiveCode> setRenderRotation(V2TXLiveRotation rotation)
```

参数

参数	类型	含义
rotation	V2TXLiveRotation	旋转角度，默认值：0。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveRotation 枚举值

取值	含义
v2TXLiveRotation0	不旋转。
v2TXLiveRotation90	顺时针旋转90度。
v2TXLiveRotation180	顺时针旋转180度。
v2TXLiveRotation270	顺时针旋转270度。

setRenderFillMode

设置画面的填充模式。

```
Future<V2TXLiveCode> setRenderFillMode(V2TXLiveFillMode mode)
```

参数

参数	类型	含义
mode	V2TXLiveFillMode	画面填充模式，默认值：v2TXLiveFillModeFit。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

V2TXLiveFillMode 枚举值

取值	含义
v2TXLiveFillModeFit	图像适应屏幕，保持画面完整，但如果图像宽高比不同于屏幕宽高比，会有黑边的存在。
v2TXLiveFillModeFill	图像铺满屏幕，不留黑边，如果图像宽高比不同于屏幕宽高比，部分画面内容会被裁剪掉。

pauseVideo

暂停播放器的视频流。

```
Future<V2TXLiveCode> pauseVideo()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

resumeVideo

恢复播放器的视频流。

```
Future<V2TXLiveCode> resumeVideo()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

snapshot

截取播放过程中的视频画面。

说明：

返回值成功后可以在 `V2TXLivePlayerObserver.onSnapshotComplete` 回调中获取截图图片。

```
Future<V2TXLiveCode> snapshot()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_REFUSED：播放器处于停止状态，不允许调用截图操作。

音频相关接口

pauseAudio

暂停播放器的音频流。

```
Future<V2TXLiveCode> pauseAudio()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

resumeAudio

恢复播放器的音频流。

```
Future<V2TXLiveCode> resumeAudio()
```

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

setPlayoutVolume

设置音量。

```
Future<V2TXLiveCode> setPlayoutVolume(int volume)
```

参数

参数	类型	含义
volume	int	音量大小，取值范围0 - 100，默认值：100。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

enableVolumeEvaluation

启用播放音量大小提示。

说明：

开启后可以在 `V2TXLivePlayerObserver.onPlayoutVolumeUpdate` 回调中获取到 SDK 对音量大小值的评估。

```
Future<V2TXLiveCode> enableVolumeEvaluation(int intervalMs)
```

参数

参数	类型	含义
intervalMs	int	onPlayoutVolumeUpdate 音量大小回调的触发间隔，单位为 ms，最小间隔为 100ms。如果小于等于0则会关闭回调，建议设置为 300ms。默认值：0，不开启。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。

更多实用接口

setCacheParams

设置播放器缓存自动调整的最小和最大时间（单位：秒）。

```
Future<V2TXLiveCode> setCacheParams(double minTime, double maxTime)
```

参数

参数	类型	含义
minTime	double	缓存自动调整的最小时间，取值需要大于0。默认值：1。
maxTime	double	缓存自动调整的最大时间，取值需要大于0。默认值：5。

返回

返回值 V2TXLiveCode：

- V2TXLIVE_OK：成功。
- V2TXLIVE_ERROR_INVALID_PARAMETER：操作失败，minTime 和 maxTime 需要大于0。
- V2TXLIVE_ERROR_REFUSED：播放器处于播放状态，不支持修改缓存策略。

showDebugView

是否显示播放器状态信息的调试浮层。

```
Future<void> showDebugView (bool isShow)
```

参数

参数	类型	含义
isShow	bool	是否显示，默认值：false。

V2TXLivePlayerObserver

最近更新时间：2022-10-20 15:57:08

功能

腾讯云直播的播放器回调通知。

介绍

可以接收 **V2TXLivePlayer** 播放器的一些回调通知，包括播放器状态、播放音量回调、音视频首帧回调、统计数据、警告和错误信息等。

SDK 基础回调

onError

直播播放器错误通知，播放器出现错误时，会回调该通知。

```
V2TXLivePlayerListenerType.onError
```

参数

参数	类型	含义
code	V2TXLiveCode	错误码。
msg	String	错误信息。
extraInfo	Map	扩展信息。

onWarning

直播播放器警告通知。

```
V2TXLivePlayerListenerType.onWarning
```

参数

参数	类型	含义
code	V2TXLiveCode	警告码。
msg	String	警告码信息。

参数	类型	含义
extraInfo	Map	扩展信息。

onConnected

已经成功连接到服务器通知。

```
V2TXLivePlayerListenerType.onConnected
```

参数

参数	类型	含义
extraInfo	Map	扩展信息。

视频相关回调

onVideoPlaying

视频播放事件通知。

```
V2TXLivePlayerListenerType.onVideoPlaying
```

参数

参数	类型	含义
firstPlay	bool	第一次播放标志。
extraInfo	Map	扩展信息。

onVideoLoading

视频加载事件通知。

```
V2TXLivePlayerListenerType.onVideoLoading
```

参数

参数	类型	含义
extraInfo	Map	扩展信息。

onVideoResolutionChanged

直播播放器分辨率变化通知。

```
V2TXLivePlayerListenerType.onVideoResolutionChanged
```

参数

参数	类型	含义
width	int	视频宽
height	int	视频高

onSnapshotComplete

截图回调。

```
V2TXLivePlayerListenerType.onSnapshotComplete
```

参数

参数	类型	含义
image	Uint8List	已截取的视频画面。

onRenderVideoFrame

自定义视频渲染回调。

说明：

调用 `V2TXLivePlayer.enableCustomRendering(pixelFormat:bufferType:)` 开启自定义渲染之后，会收到这个回调通知。

```
V2TXLivePlayerListenerType.onRenderVideoFrame
```

参数

参数	类型	含义
videoFrame	Map	视频帧数据。

音频相关回调

onAudioPlaying

音频播放事件通知。

```
V2TXLivePlayerListenerType.onAudioPlaying
```

参数

参数	类型	含义
firstPlay	bool	第一次播放标志。
extraInfo	Map	扩展信息。

onAudioLoading

音频加载事件通知。

```
V2TXLivePlayerListenerType.onAudioLoading
```

参数

参数	类型	含义
extraInfo	Map	扩展信息。

onPlayoutVolumeUpdate

播放器音量大小回调。

```
V2TXLivePlayerListenerType.onPlayoutVolumeUpdate
```

参数

参数	类型	含义
volume	int	音量大小，取值范围：0 - 100。

统计回调

onStatisticsUpdate

直播播放器统计数据回调。

```
V2TXLivePlayerListenerType.onStatisticsUpdate
```

参数

参数	类型	含义
statistics	Map	播放器统计数据。

示例

最近更新时间：2022-10-20 15:57:08

针对开发者的接入反馈的高频问题，腾讯云提供有直观易懂的 **API-Example** 工程，方便开发者可以快速的了解相关 API 的使用，欢迎使用。

工程地址

所属平台	GitHub 地址
Flutter	Github

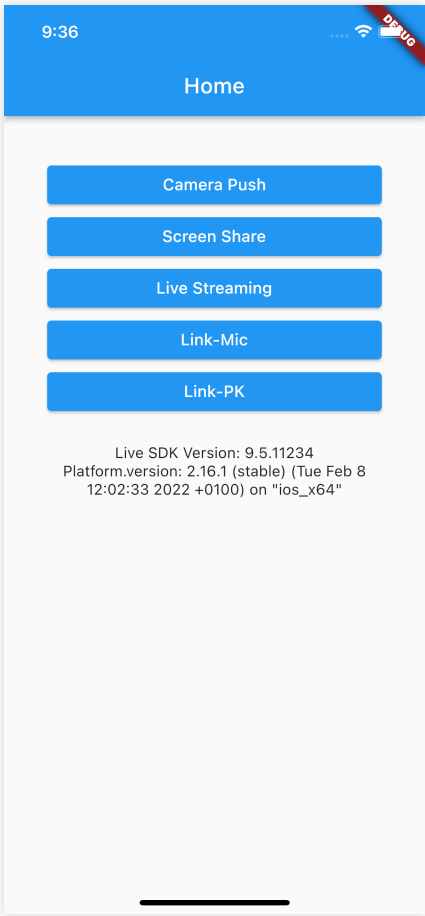
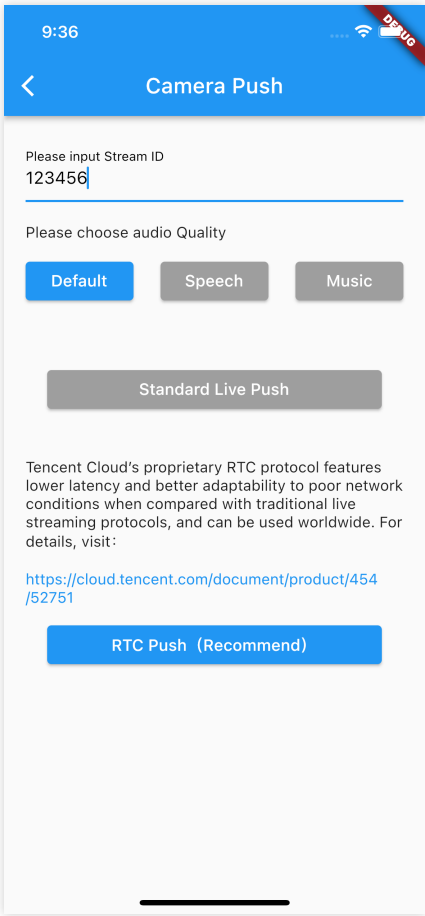
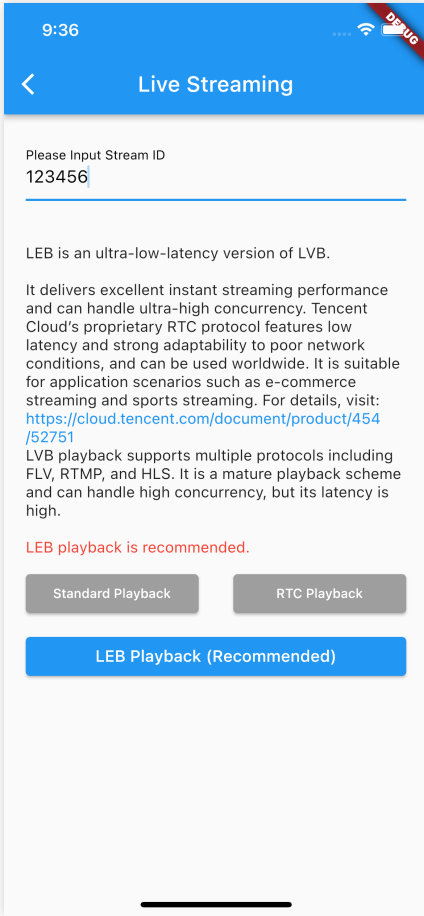
目录说明

在这个示例项目中包含了以下基础功能场景：

- [摄像头推流](#)
- [录屏推流](#)
- [直播拉流](#)
- [连麦互动](#)
- [连麦 PK](#)

应用图示

示例主页	推流示例	拉流示例
----------------------	----------------------	----------------------

示例主页	推流示例	拉流示例
		

错误码表

最近更新时间：2022-10-20 15:57:08

onError 严重错误

SDK 发现了一些严重问题，推流无法继续了，例如 License 不合法，调用失败。

注意：

视频编码失败并不会直接影响推流，SDK 会做自行处理以保证后面的视频编码成功。

code	事件定义	含义说明
0	V2TXLIVE_OK	没有错误
-1	V2TXLIVE_ERROR_FAILED	暂未归类的通用错误
-2	V2TXLIVE_ERROR_INVALID_PARAMETER	调用 API 时，传入的参数不合法
-3	V2TXLIVE_ERROR_REFUSED	API 调用被拒绝
-4	V2TXLIVE_ERROR_NOT_SUPPORTED	当前 API 不支持调用
-5	V2TXLIVE_ERROR_INVALID_LICENSE	Licence 不合法，调用失败。注：在 <code>startLivePlay</code> 之前，需要通过 <code>V2TXLivePremier#setLicence</code> 设置 License 后方可成功播放，否则将播放失败（黑屏），全局仅设置一次即可。直播 License、短视频 License 和视频播放 License 均可使用，若您暂未获取上述 License，可 快速免费申请测试版 License 以正常播放，正式版 License 需 购买
-6	V2TXLIVE_ERROR_REQUEST_TIMEOUT	请求服务器超时
-7	V2TXLIVE_ERROR_SERVER_PROCESS_FAILED	服务器拒绝请求
-8	V2TXLIVE_ERROR_DISCONNECTED	连接断开

code	事件定义	含义说明
-2304	V2TXLIVE_ERROR_NO_AVAILABLE_HEVC_DECODERS	找不到可用的 HEVC 解码器，当抛出此错误码，说明当前设备不支持 H265 解码，请切换至 H264 数据流进行播放

onWarning 警告事件

大部分的警告事件会触发一些重试性的保护逻辑或者恢复逻辑，很大概率能够由 SDK 自行恢复。部分警告事件需要由开发者处理。警告的目的主要用于提示开发者或者最终用户。

code	事件定义	含义说明
1101	V2TXLIVE_WARNING_NETWORK_BUSY	网络状况不佳：上行带宽太小，上传数据受阻
2105	V2TXLIVE_WARNING_VIDEO_BLOCK	当前视频播放出现卡顿
-1301	V2TXLIVE_WARNING_CAMERA_START_FAILED	摄像头打开失败
-1316	V2TXLIVE_WARNING_CAMERA_OCCUPIED	摄像头正在被占用中，可尝试打开其他摄像头
-1314	V2TXLIVE_WARNING_CAMERA_NO_PERMISSION	摄像头设备未授权，通常在移动设备出现，可能是权限被用户拒绝了
-1302	V2TXLIVE_WARNING_MICROPHONE_START_FAILED	麦克风打开失败
-1319	V2TXLIVE_WARNING_MICROPHONE_OCCUPIED	麦克风正在被占用中，例如移动设备正在通话时，打开麦克风会失败
-1317	V2TXLIVE_WARNING_MICROPHONE_NO_PERMISSION	麦克风设备未授权，通常在移动设备出现，可能是权限被用户拒绝了
-1309	V2TXLIVE_WARNING_SCREEN_CAPTURE_NOT_SUPPORTED	当前系统不支持屏幕分享
-1308	V2TXLIVE_WARNING_SCREEN_CAPTURE_START_FAILED	开始录屏失败，如果在移动设备出现，可能是权限被用户拒绝了
-7001	V2TXLIVE_WARNING_SCREEN_CAPTURE_INTERRUPTED	录屏被系统中断