

应用云渲染 SDK 文档



腾讯云

【 版权声明 】

©2013–2024 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【 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。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。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。

【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或 95716。

文档目录

SDK 文档

- JavaScript SDK

- Android SDK

- iOS SDK

 - iOS SDK 下载

 - iOS SDK 接口

 - iOS 接入示例

SDK 文档

JavaScript SDK

最近更新时间：2023-11-28 10:56:21

应用云渲染提供 JavaScript SDK 用于 Web 接入。SDK 提供了丰富的接口，满足大部分接入需求。接入方法请参见 [快速入门](#) 及 [搭建示例](#)，并可通过 [SDK 接口说明文档](#)，获取更多功能指引。

SDK 下载

SDK	下载地址	SDK 说明文档	版本发布日志
JavaScript SDK	GitHub 下载	DOC	Release notes

安装说明

`import dist/tcg-sdk`，导出的为 umd 格式。

Android SDK

最近更新时间：2022-08-18 11:01:38

腾讯云云游戏提供 Android SDK 用于 Android 终端接入，支持 PC 应用和手机应用。SDK 提供了丰富的接口，满足大部分接入需求。接入方法和使用说明请参见 [GitHub](#)，仓库中包含完整的文档以及可快速集成的 Demo 工程示例。

iOS SDK

iOS SDK 下载

最近更新时间：2022-12-12 11:47:03

应用云渲染提供 iOS SDK 用于 iOS 端接入。SDK 提供了丰富的接口，满足大部分接入需求。接入方法请参见 [快速入门](#) 及 [接入示例](#)，并可通过接口概览页面，获取更多功能指引。

SDK	ZIP 包	GitHub	SDK 说明文档
iOS SDK	下载	GitHub 下载	DOC

iOS SDK 接口

最近更新时间：2022-04-21 20:17:08

iOS SDK 接口概览

生命周期相关接口

接口名称	接口描述
initWithParams	SDK 初始化
startGameWithRemoteSession	启动应用云渲染
stopGame	停止应用云渲染
TCGGamePlayerDelegate	生命周期回调

音视频控制接口

接口名称	接口描述
pauseResumeGame	暂停/恢复画面传输
setStreamBitrateMix	设置建议的帧率码率
setVolumeScale	设置应用音频 PCM 增益大小，默认1.0
videoView	获取应用画面渲染的图层
setVideoViewEnablePinch	是否允许双指缩放/拖动应用图层
setVideoViewInsets	设置缩放时画面边框的偏移大小，这能让应用画面划出手机边框。
onVideoSizeChanged	视频图像宽高发生变化
onVideoOrientationChanged	视频画面的朝向发生改变，进入应用前也会回调。

应用操控相关接口

接口名称	接口描述
gameController	获取应用操控辅助类
initWithFrame	创建鼠标视图（继承 UIView），支持将手指单击事件转换为 PC 上的鼠标操作
setCursorTouc	设置鼠标的操控模式

hMode	
setCursorIsShow	设置鼠标是否可见，配合 onCursorVisibleChanged : 回调一起使用
setCursorImage	更新鼠标的图标，配合 onCursorImageUpdated:frame : 回调一起使用
setCursorSensitive	设置鼠标相对移动时的灵敏度
setClickTypesLeft	设置鼠标触发单击的事件
moveCursorWithDiffX	以接口的方式移动鼠标指针
initWithFrame	创建一个虚拟按键视图（继承 UIView），支持将手指单击事件转换为键盘/手柄按键事件
showKeyGamepad	加载虚拟按键布局并生成按键，虚拟按键配置文件与 Android SDK、JS SDK 通用
needConnected	查询当前的按键布局是否需要主动通知云端，返回 YES 表示时需调用 enableVirtualGamepad 通知云端启用虚拟手柄。
initWithFrame	创建云端触控视图（继承 UIView），支持将手指单击事件传递到云端手机上
onCursorVisibleChanged	设置云端鼠标是否可见
onCursorImageUpdated	设置云端鼠标图片
clickKeyboard	发送键盘事件，推荐使用 TCGVKeyGamepad
clickGamepadKey	发送应用手柄按键事件，推荐使用 TCGVKeyGamepad
turnJoyStickX	转动应用手柄的（左/右）摇杆，推荐使用 TCGVKeyGamepad
enableVirtualGamepad	通知云端启用虚拟应用手柄
resetRemoteKeycode	清空云端的按键，用于处理键盘卡键的异常状态
remoteMobileBackClick	触发云端的返回动作，触发云端系统的物理返回键

数据通道交互接口

接口名称	接口描述
openCustomTransChannel	创建一条数据通道
sendData	给云端 UDP 端口发送数据
close	关闭数据通道
onConnSuccessAtRemotePort	连接成功，可以通过数据通道给云端应用发送消息
onConnFailed	连接失败
onReceiveData	接收来自云端的数据

调试及日志相关接口

接口名称	接口描述
setLogger	设置日志回调接收者及过滤等级
logWithLevel	日志打印回调接口
currentStatisReport	返回获取当前的性能统计数据
TCGMouseCursorTouchMode	鼠标的操控模式
TCGLogLevel	日志等级

生命周期相关接口

[TCGGamePlayer initWithParams:andDelegate:]

SDK 初始化。接口立即返回，异步执行初始化通过回调通知结果。

参数	类型	描述
params	NSDictionary	选填，预留暂未启用
listener	TCGGamePlayerDelegate	生命周期回调

[TCGGamePlayer startGameWithRemoteSession:error:]

启动应用云渲染。

参数	类型	描述

参数	类型	描述
remoteSession	NSString	Server Session, 需从云 API 获取

[TCGGamePlayer stopGame]

停止应用云渲染。

TCGGamePlayerDelegate

TCGGamePlayer 生命周期回调。

接口名称	接口描述
[TCGGamePlayerDelegate onInitSuccess:]	SDK 初始化成功回调
[TCGGamePlayerDelegate onInitFailure:msg:]	初始化失败
[TCGGamePlayerDelegate onConnectionFailure:msg:]	连接失败回调
[TCGGamePlayerDelegate onVideoShow]	首帧绘制时机

[TCGGamePlayerDelegate onInitSuccess:]

SDK 初始化成功回调。

参数	类型	描述
localSession	NSString	ClientSession 客户端会话, 其内容较长, 可能会超出 logcat 输出限制

[TCGGamePlayerDelegate onInitFailure:msg:]

初始化失败回调。

参数	类型	描述
errorCode	TCGErrorType	错误码
errorMsg	NSError	内部错误信息

[TCGGamePlayerDelegate onConnectionFailure:msg:]

连接失败回调。

参数	类型	描述
errorCode	TCGErrorType	错误码

errorMsg	NSError	内部错误信息
----------	---------	--------

[TCGGamePlayerDelegate onVideoShow]

连接成功，画面开始显示。

音视频控制接口

[TCGGamePlayer pauseResumeGame:]

暂停/恢复 画面传输。

参数	类型	描述
doPause	BOOL	YES 表示暂停画面传输，NO 表示恢复画面传输

[TCGGamePlayer setStreamBitrateMix:max:fps:]

设置建议的帧率和码率

⚠ 注意

该接口设置的仅是建议值，云端会根据实际网络情况动态调整。

参数	类型	描述
fps	int	建议帧率，范围[10, 60]，默认45
minBitrate	int	建议最小码率，范围[1000,15000]，默认1000kbps
maxBitrate	int	建议最大码率，范围[1000,15000]，默认3000kbps

[TCGGamePlayer setVolumeScale:]

设置应用音频 PCM 增益大小，默认1.0。

参数	类型	描述
scale	float	[0,10]之间的浮点数，大于 1 时可能会导致声音失真

[TCGGamePlayer videoView]

获取应用画面渲染的图层。

[TCGGamePlayer setVideoViewEnablePinch:]

是否允许双指缩放/拖动应用图层。

参数	类型	描述
isEnabled	BOOL	YES 表示允许双指缩放画面，NO 表示不允许

[TCGGamePlayer setVideoViewInsets:]

设置缩放时画面边框的偏移大小，这能让应用画面划出手机边框。

参数	类型	描述
insets	UIEdgeInsets	指定边缘间距，默认(100, 50, 50, 100)

[TCGGamePlayerDelegate onVideoSizeChanged:]

视频图像宽高发生变化。

参数	类型	描述
videoSize	CGSize	变化后的视频图像尺寸

[TCGGamePlayerDelegate onVideoOrientationChanged:]

视频画面的朝向发生改变，进入应用前也会回调。

⚠ 注意

这里的朝向指画面里的内容是竖屏或横屏内容（可理解为云端手机的朝向），而视频画面的宽高是不会发生改变的。

参数	类型	描述
orientation	UIInterfaceOrientation	视频画面的朝向

应用操控相关接口

⚠ 注意

所有按键相关接口必须在 [\[TCGGamePlayerDelegate onVideoShow\]](#) 回调发生之后调用才生效。

[TCGGamePlayer gameController]

获取应用操控辅助类。

[TCGVirtualMouseCursor initWithFrame:controller:]

创建鼠标视图（继承 UIView），支持将手指单击事件转换为 PC 上的鼠标操作。

❗ 说明

视图为透明视图，使用时添加为 `videoView` 的子视图，大小与 `videoView` 相同。

参数	类型	描述
frame	CGRect	视图的大小与位置
controller	TCGGameController	应用操控辅助类

[TCGVirtualMouseCursor setCursorTouchMode:]

设置鼠标的操控模式。

参数	类型	描述
mode	TCGMouseCursorTouchMode	鼠标的操控模式

[TCGVirtualMouseCursor setCursorIsShow:]

设置鼠标是否可见，配合 `onCursorVisibleChanged:` 回调一起使用。

参数	类型	描述
isShow	BOOL	YES 表示显示，NO 表示隐藏

[TCGVirtualMouseCursor setCursorImage:andRemoteFrame:]

更新鼠标的图标，配合 `onCursorImageUpdated:frame:` 回调一起使用。

参数	类型	描述
image	UIImage	鼠标指针图标
imageFrame	CGRect	view 的大小位置，云端主动下发

[TCGVirtualMouseCursor setCursorSensitive:]

设置鼠标相对移动时的灵敏度，`TCGMouseCursorTouchMode(RelativeTouch、RelativeOnly)` 下有效。

参数	类型	描述
sensitive	CGFloat	默认1.0与手动滑动的幅度相同

[TCGVirtualMouseCursor setClickTypeIsLeft:]

设置鼠标触发单击的事件，[TCGMouseCursorTouchMode\(AbsoluteTouch、RelativeTouch\)](#) 下有效。

参数	类型	描述
isLeft	BOOL	YES 表示触发鼠标左键，NO 表示触发鼠标右键

[TCGVirtualMouseCursor moveCursorWithDiffX:diffY:]

以接口的方式移动鼠标指针。

参数	类型	描述
diffX	CGFloat	在当前鼠标视图的 X 轴上，鼠标指针移动 diffX 个 point
diffY	CGFloat	在当前鼠标视图的 Y 轴上，鼠标指针移动 diffY 个 point

[TCGVKeyGamepad initWithFrame:controller:]

创建一个虚拟按键视图（继承 UIView），支持将手指单击事件转换为键盘/手柄按键事件。

⚠ 注意

需要选用 `TCGVkey.framework` 库。

参数	类型	描述
frame	CGRect	视图的大小与位置
controller	TCGGameController	应用操控辅助类

[TCGVKeyGamepad showKeyGamepad:]

加载虚拟按键布局并生成按键，虚拟按键配置文件与 Android SDK、JS SDK 通用。

参数	类型	描述
cfg	NSString	JSON 格式的配置内容

[TCGVKeyGamepad needConnected]

查询当前的按键布局是否需要主动通知云端，返回 YES 表示时需调用 [enableVirtualGamepad](#) 通知云端启用虚拟手柄。

[TCGRemoteTouchScreen initWithFrame:controller:]

创建云端触控视图（继承 UIView），支持将手指单击事件传递到云端手机上。

⚠ 注意

视图为透明视图，使用时添加为 `videoView` 的子视图，大小与 `videoView` 相同。

参数	类型	描述
frame	CGRect	视图的大小与位置
controller	TCGGameController	应用操控辅助类

[TCGGameController onCursorVisibleChanged:]

设置云端鼠标是否可见

参数	类型	描述
isVisble	BOOL	YES 表示云端鼠标可见，NO 表示云端鼠标不可见

[TCGGameController onCursorImageUpdated:frame:]

设置云端鼠标图片。

参数	类型	描述
image	UIImage	鼠标指针图标
imageFrame	CGRect	view的大小位置，云端主动下发

[TCGGameController clickKeyboard:isDown:]

发送键盘事件，推荐使用 [TCGVKeyGamepad](#)。

参数	类型	描述
keycode	int	按键值
isDown	BOOL	true 表示按下，false 表示抬起

[TCGGameController clickGamepadKey:isDown:]

发送应用手柄按键事件，推荐使用 [TCGVKeyGamepad](#)。

参数	类型	描述

keycode	int	按键值
isDown	BOOL	YES 表示发送按下消息，NO 表示发送抬起消息

[TCGGameController turnJoyStickX:y:isLeft:]

转动应用手柄的（左/右）摇杆，推荐使用 [TCGVKeyGamepad](#)。

参数	类型	描述
deltaX	CGFloat	范围[-1, 1]，最左端为-1 最右端为 1
deltaY	CGFloat	范围[-1, 1]，最左端为-1 最右端为 1
isLeft	BOOL	YES 表示转动左摇杆，NO 表示转动右摇杆

[TCGGameController enableVirtualGamepad:]

通知云端启用虚拟应用手柄。

参数	类型	描述
enable	BOOL	YES 表示发送启用的消息，NO 表示发送停用的消息

[TCGGameController resetRemoteKeycode]

清空云端的按键，用于处理键盘卡键的异常状态。

[TCGGameController remoteMobileBackClick]

触发云端的返回动作，触发云端系统的物理返回键。

数据通道交互接口

[TCGGamePlayer openCustomTransChannel:delegate:]

创建一条数据通道。

return: 数据通道对象，与云端连接的结果通过代理异步回调通知。

参数	类型	描述
remotePort	int	云端应用监听的端口号
channelDelegate	TCGCustomTransChannelDelegate	数据通道的代理，用于回调数据通道的连接状态和数据

[TCGCustomTransChannel sendData:]

给云端 UDP 端口发送数据。

return: 发送状态。0发送成功, ** -1** 数据超过长度, ** -2** 当前通道未连接成功。

参数	类型	描述
data	NSData	需要发送的数据内容, 单次发送大小限制在1200字节

[TCGCustomTransChannel close]

关闭数据通道。

[TCGCustomTransChannelDelegate onConnSuccessAtRemotePort:]

连接成功, 可以通过数据通道给云端应用发送消息。

参数	类型	描述
remotePort	int	云端应用监听的端口号

[TCGCustomTransChannelDelegate onConnFailed:atRemotePort:]

连接失败。

参数	类型	描述
remotePort	int	云端应用监听的端口号
error	NSError	失败的原因, code: ** -1**云端后台错误, 10009**连接超时

[TCGCustomTransChannel onClosedAtRemotePort:]

云端主动关闭了通道。

参数	类型	描述
remotePort	int	云端应用监听的端口号

[TCGCustomTransChannel onReceiveData:fromRemotePort:]

接收来自云端的数据。

参数	类型	描述
data	NSData	数据内容
remotePort	int	云端应用监听的端口号

调试及日志相关接口

[TCGGamePlayer setLogger:withMinLevel:]

设置日志回调接收者及过滤等级。

参数	类型	描述
logger	TCGLogDelegate	数据内容
minLevel	TCGLogLevel	云端应用监听的端口号

[TCGLogDelegate logWithLevel:log:]

日志打印回调接口。

参数	类型	描述
logLevel	TCGLogLevel	日志的等级
log	NSString	日志的内容

[TCGGamePlayer currentStatisReport]

返回获取当前的性能统计数据。

TCGMouseCursorTouchMode

鼠标的操控模式。

状态定义	说明
AbsoluteTouch	鼠标跟随手指移动，单击可以单击按键
RelativeTouch	手指滑动控制鼠标相对移动，轻触触发鼠标左键，长按触发按单击鼠标左键，滑动仅触发鼠标移动
RelativeOnly	触控，鼠标跟随手指移动，单击可以单击按键（鼠标左键或右键）

TCGLogLevel

日志等级。

等级	说明
TCGLogLevelDebug	调试信息，主要用于开发过程中打印一些运行信息

TCGLogLevelInfo	粗粒度信息，输出程序运行过程中的重要信息
TCGLogLevelWarning	警告，传递系统中的潜在错误信息
TCGLogLevelError	错误，传递系统或应用程序中出现的各种级别的错误
TCGLogLevelNone	不打印日志

iOS 接入示例

最近更新时间：2022-04-21 20:17:15

接入步骤

Podfile 中添加：

```
# 主SDK
pod 'TCGSDK', :git => "https://github.com/tencentyun/cloudgame-ios-sdk.git"
# 可选虚拟按键SDK
pod 'TCGVKey', :git => "https://github.com/tencentyun/cloudgame-ios-sdk.git"
```

调用示例

```
- (void)createGamePlayer {
    if (self.gamePlayer) {
        return;
    }
    // 初始化SDK
    self.gamePlayer = [[TCGGamePlayer alloc] initWithParams:nil andDelegate:self];
    self.gameController = self.gamePlayer.gameController;
    self.gameController.controlDelegate = self;
    // 添加video视图
    [self.gamePlayer.videoView setFrame:self.view.bounds];
    // 初始化鼠标视图
    self.mouseCursor = [[TCGVirtualMouseCursor alloc]
initWithFrame:self.gamePlayer.videoView.bounds
                controller:self.gameController];
    // 设置默认的鼠标指针图片，防止后台未及时下放时显示空白
    [self.mouseCursor setCursorImage:[UIImage imageNamed:@"default_cursor"]
andRemoteFrame:CGRectMake(0, 0, 32, 32)];

    [self.gamePlayer.videoView addSubview:self.mouseCursor];
    [self.view addSubview:self.gamePlayer.videoView atIndex:0];
}

- (void)startGameWithRemoteSession:(NSString *)remoteSession {
    NSLog(@"从业务后台成功申请到云端机器");
    NSError *error;
    // 开始应用
    [self.gamePlayer startGameWithRemoteSession:remoteSession error:&error];
    NSLog(@"start game %@", error);
}
```

```
}

- (void)stopGame {
    dispatch_async(dispatch_get_main_queue(), ^{
        if (self.gamePlayer == nil) {
            return;
        }
        [self.mouseCursor removeFromSuperview];
        self.mouseCursor = nil;
        [self.gamePlayer.videoView removeFromSuperview];
        [self.gamePlayer stopGame];
        self.gamePlayer = nil;
    });
}

#pragma mark --- TCGGamePlayerDelegate ---
- (void)onInitSuccess:(NSString *)localSession {
    NSLog(@"SimplePCDemo onInitSuccess, 本地初始化成功");
    // 开始向后台申请云端资源
    [self getRemoteSessionWithLocalSession:localSession];
}

- (void)onVideoSizeChanged:(CGSize)videoSize {
    NSLog(@"更新画面尺寸:%@", NSStringFromCGSize(videoSize));
    CGFloat newWidth = self.view.frame.size.width - [self.view safeAreaInsets].left -
    [self.view safeAreaInsets].right;
    CGFloat newHeight = self.view.frame.size.height;
    // 应用画面强制横屏、设置应用画面居中显示类似 UIViewContentModeScaleAspectFit
    if (newWidth/newHeight < videoSize.width/videoSize.height) {
        newHeight = floor(newWidth * videoSize.height / videoSize.width);
    } else {
        newWidth = floor(newHeight * videoSize.width / videoSize.height);
    }
    self.gamePlayer.videoView.frame = CGRectMake((self.view.frame.size.width -
    newWidth) / 2,
                                                (self.view.frame.size.height - newHeight) / 2,
                                                newWidth, newHeight);
    self.mouseCursor.frame = self.gamePlayer.videoView.bounds;
}

- (void)onVideoShow {
    NSLog(@"SimplePCDemo onVideoShow, 应用开始有画面");
    // 设置正确的鼠标显示与控制模式
    [self.gameController setCursorShowMode:TCGMouseCursorShowMode_Local];
}
```

```
[self.mouseCursor
setCursorTouchMode:TCGMouseCursorTouchMode_RelativeTouch];
}

- (void)onConnectionFailure:(TCGErrorType)errorCode msg:(NSError *)errorMsg {
    NSLog(@"SimplePCDemo onConnectionFailure");
    [self stopGame];
}

- (void)onInitFailure:(TCGErrorType)errorCode msg:(NSError *)errorMsg {
    NSLog(@"SimplePCDemo onInitFailure");
    [self stopGame];
}

- (void)onStartReConnectWithReason:(TCGErrorType)reason {
    NSLog(@"onStartReConnectWithReason 与云端链接断开, SDK内部重连中");
}

#pragma mark --- TCGGameControllerDelegate ---
- (void)onCursorImageUpdated:(UIImage *)image frame:(CGRect)imageFrame {
    [self.mouseCursor setCursorImage:image andRemoteFrame:imageFrame];
}

- (void)onCursorVisibleChanged:(BOOL)isVisible {
    NSLog(@"onCursorVisibleChanged 鼠标状态:%@", isVisible?@"显示":@"隐藏");
    [self.mouseCursor setCursorIsShow:isVisible];
}

- (void)onClickedTextField:(TCGTextFieldType)type {
    NSLog(@"onClickedTextField:%zu", type);
}
```

- 客户端获取的 localSession 作为参数传递给 App 后台（传递方式业务方可以自行实现），业务后台请求云 API 锁定机器并创建会话，拿到 remoteSession。
- 客户端通过 remoteSession 启动应用。
- SDK 启动应用之后会通过 TCGGamePlayerDelegate 把启动过程中的回调告知给客户端。
- videoView 的尺寸业务方根据自身场景调整尺寸，如果使用 SDK 内置虚拟鼠标，鼠标的视图最好设置成 video 视图的子视图且大小与之相同。