

腾讯云视频点播服务

Web 播放器 SDK 开发指南



版本：1.3.3
时间：2015.11

目录

1. 概述.....	3
2. 阅读对象.....	3
3. 准备.....	3
4. 开始 API 使用	4
4.1. 方法总览.....	4
4.2. 详细方法.....	5
5. 视频文件上传功能.....	12
6. 视频文件操作 API 说明	12
7. SDK 升级历史信息和文档变更	12
8. 反馈和建议.....	13

1. 概述

腾讯云视频点播播放器 Web SDK 解决方案，可帮助腾讯云视频用户直接使用经过验证的视频播放能力，通过灵活的接口，快速同自有 Web 应用集成，以实现桌面应用播放功能。

该 SDK 所播放的文件受限于全局防盗链功能定义。详细内容请查看[官网 FAQ_安全功能](#)相关说明。

2. 阅读对象

该文档面向考虑使用腾讯云视频点播播放器 Web SDK 进行开发并具备 javascript 语言基础的开发人员。

3. 准备

- 注册腾讯云点播并上传视频
- 页面引入初始化脚本

```
<script src=http://qzonestyle.gtimg.cn/open/qcloud/video/h5/h5connect.js  
charset=" utf-8" ></script>
```

4. 开始 API 使用

4.1. 方法总览

- 构造类

构造播放器	qcVideo.Player
是否自动播放	auto_play
播放器宽度设置	width
播放器高度设置	height
屏蔽全屏播放	disable_full_screen
禁止拖动	disable_drag
拉升视频匹配播放器	stretch_full
限制播放时间	stop_time
是否从上次停止处继续播放	remember

- 获取参数和状态

获取当前音量	getVolume
获取当前视频总时长	getDuration
获取当前播放位置	getCurrentTime
当前播放状态“加载中”查询	isSeeking
当前播放状态“暂停中”查询	isSuspended
当前播放状态“播放中”查询	isPlaying
当前播放状态“播放结束”查询	isPlayEnd
获取当前播放器宽度	getWidth
获取当前播放器高度	getHeight

获取当前视频清晰度	<u>getClarity</u>
获取当前视频清晰度	<u>getAllClaritys</u>

● 设置和动作

设置当前播放器宽度高度	<u>resize</u>
播放指定时长点	<u>play</u>
暂停当前播放的视频	<u>pause</u>
恢复播放视频	<u>resume</u>
更换视频清晰度	<u>setClarity</u>
动态更换视频	<u>changeVideo</u>
添加弹幕	addBarrage
关闭弹幕	closeBarrage

4.2. 详细方法

● 构造类

qcVideo.Player(id,option,listener)

功能：播放器构造函数

参数：

- 1 id: String ; 必选参数 ; 页面放置播放器的容器 ID
- 2 option: Object; 必选参数 ; 播放参数
- 3 listener: function; 可选参数 ; 播放状态变化回调函数

示例一 (播放腾讯云上保存的视频，需提供视频 ID 和应用 ID)：

```
var player = new qcVideo.Player(  
    //页面放置播放位置的元素 ID  
    "element_id" ,  
    {
```

```

//视频 ID (必选参数)
"file_id" : "16092504232103514290" ,
//应用 ID (必选参数), 同一个账户下的视频, 该参数是相同的
"app_id" : "1251132611" ,
//是否自动播放 默认值0 (0: 不自动, 1: 自动播放)
"auto_play" : "0" ,
//播放器宽度, 单位像素
"width" : 640,
//播放器高度, 单位像素
"height" : 480,
//屏蔽全屏播放标识,默认值0 (0: 支持全屏播放,1: 禁用全屏播放)
"disable_full_screen" : 0,
//禁止拖动标识,默认值0 (0: 允许拖拽,1: 禁止拖拽)
"disable_drag" :1,
//如视频尺寸小于播放器尺寸, 拉伸视频至播放器大小, 默认值0 (0: 不拉伸,1: 拉伸全屏)
"stretch_full" :1,
//限时播放时间, 单位秒, 设定后, 播放至该时间即停止播放, 默认值0 ( 0 : 为不限制播放时间 )
"stop_time " : 10,
//是否从上次停止处继续播放, 默认值0 ( 0 : 从头播放 ; 1 : 继续从上次停止处播放 )
"remember" :1 ,
//防盗链专用配置信息, 可选
//如开启防盗链后, 必须配置下面的含防盗链参数 URL 视频地址, 才可支持播放器播放
//注意, 下面所有的地址必须是上面对应的 file_id 的视频资源地址, 否则不会生效
"videos" :[
    "http://testdomain.com/mymp4.....testpath.1.mp4?....." ,
    "http://testdomain.com/mymp4.....testpath.2.mp4?....." ,
    "http://testdomain.com/mymp4.....testpath.3.mp4?....."
],
"Wmode" : "window" ,//默认 window 不支持其他页面元素覆盖在上面, 如需要可以修改为 opaque 或其他 flash Vmode 的参数值
"stretch_patch" : false,//默认为 false ,设置为 true 支持将开始、暂停、结束时的图片贴片, 铺满播放器
},
{
    //全屏/退出全屏操作 ,isFullScreen: true→全屏 ; false→ 退出全屏
    'fullScreen' : function(isFullScreen){
        //console.debug( 'out listener isFullScreen == ',isFullScreen);
    },
    //播放状态
    'playStatus' : function(status){

```

```

        /*status 可为 {ready: "播放器已准备就绪",seeking:" 搜索",
        suspended:" 暂停", playing:" 播放中", playEnd:" 播放结束"} */
        //console.debug( 'out listener status == ',status);
    },
    //拖动播放位置变化 ; second 拖动播放的位置 ( 单位秒 )
    'dragPlay' : function(second){
        //console.debug( 'out listener dragPlay == ',second);
    }
}
);

```

示例二(播放非腾讯云上保存的视频，例如用户自有视频，或存放在其他第三方网站视频，需提供视频 URL)：

```

var player = new qcVideo.Player(
    //页面放置播放位置的元素 ID
    "element_id" ,
    {
        //是否自动播放 默认值0 (0: 不自动, 1: 自动播放)
        "auto_play" : "0" ,
        //播放器宽度, 单位像素
        "width" : 640,
        //播放器高度, 单位像素
        "height" : 480,
        //屏蔽全屏播放标识,默认值0 (0: 支持全屏播放,1: 禁用全屏播放)
        "disable_full_screen" : 0,
        //禁止拖动标识,默认值0 (0: 允许拖拽,1: 禁止拖拽)
        "disable_drag" :1,
        //如视频尺寸小于播放器尺寸, 拉伸视频至播放器大小, 默认值0 (0: 不拉伸,1: 拉伸全屏)
        "stretch_full" :1,
        //限时播放时间, 单位秒, 设定后, 播放至该时间即停止播放, 默认值0 ( 0 : 为不限制播放时间 )
        "stop_time " : 10,
        //是否从上次停止处继续播放, 默认值0 ( 0 : 从头播放 ; 1 : 继续从上次停止处播放 )
        "remember" :1 ,
        "Wmode" : "window" ,//默认 window 不支持其他页面元素覆盖在上面, 如需要可以修改为 opaque 或其他 flash Vmode 的参数值
        "stretch_patch" : false, //默认为 false ,设置为 true 支持将开始、暂停、结束时的图片贴片, 铺满播放器
        //注意, 下面所有的地址必须包含 duration ( 时长 )
        "third_video" :{
            'duration' : 20, //视频时长(单位秒)

```

//10/210->手机, 20/220->标清, 30/230->高清, 40/240->超清; (至少包含一个地址)

```
    'urls' : {
        10: "手机视频地址"
        ,20: "标清视频地址"
        ,30: "高清视频地址"
        ,40: "超清视频地址"
        ,210: "hls 手机视频地址"
        ,220: "hls 标清视频地址"
        ,230: "hls 高清视频地址"
        ,240: "hls 超清视频地址"
    }
}
},
{
    //全屏/退出全屏操作 ,isFullScreen: true→全屏 ; false→ 退出全屏
    'fullScreen' : function(isFullScreen){
        //console.debug( 'out listener isFullScreen == ',isFullScreen);
    },
    //播放状态
    'playStatus' : function(status){
        /*status 可为 {ready: "播放器已准备就绪" ,seeking:" 搜索" ,
        suspended:" 暂停" , playing:" 播放中" , playEnd:" 播放结束" } */
        //console.debug( 'out listener status == ',status);
    },
    //拖动播放位置变化 ; second 拖动播放的位置 ( 单位秒 )
    'dragPlay' : function(second){
        //console.debug( 'out listener dragPlay == ',second);
    }
}
);
```

- **获取参数和状态**

getVolume()

功能：取当前音量

返回：Number，取值范围（0 到 1）;如 0.5

getDuration()

功能：取当前视频总时长

返回：int，单位秒

getCurrentTime()

功能：取当前播放位置

返回：int，单位秒

isSeeking()

功能：当前播放状态是否“加载中”

返回：Boolean；true 为“加载中”

isSuspended()

功能：当前播放状态是否“暂停中”

返回：Boolean；true 为“暂停中”

isPlaying()

功能：当前播放状态是否“播放中”

返回：Boolean；true 为“播放中”

isPlayEnd()

功能：当前播放状态是否“播放结束”

返回：Boolean；true 为“播放结束”

getWidth()

功能：取当前播放器宽度

返回：int

getHeight()

功能：取当前播放器高度

返回：int

getClarity()

功能：取当前视频的清晰度

返回：int (1: “手机”，2: “标清”，3: “高清”，4: “超清”)

getAllClaritys()

功能：取当前视频全部的清晰度

返回：Array<int>(1:“手机” ,2: “标清” ,3: “高清” ,4: “超清”)

● 设置和动作

resize(width,height)

功能：设置当前播放器宽度高度

参数：

1 width :int; 播放器宽度

2 height :int; 播放器高度

play(second)

功能：开始播放指定时长点

参数：int ; second 单位秒

返回：int ; 操作结果返回码

pause()

功能：暂停当前播放的视频

返回：int ; 操作结果返回码

resume()

功能：恢复播放视频

返回：int ; 操作结果返回码

setClarity(clarity)

功能：更换视频清晰度

参数：int ; clarity 清晰度 取值范围（ 1 : “手机” ,2 : “标清” ,3 : “高清” ,4 : “超清” ）

注意：clarity 指定的清晰度，请确保当前视频具备该清晰度，否则将按播放器默认规则选择一个清晰度播放

返回：int ; 操作结果返回码

changeVideo(opt)

功能：动态更换视频

参数：opt Object ; 包含将要播放的视频的基本信息，和构造函数第二个参数基本一致;

返回：int；操作结果[返回码](#)

示例：

```
player.changeVideo({
  //视频 ID (必选参数)
  "file_id" : "16092504232103514290" ,
  //应用 ID (必选参数)
  "app_id" : "1251132611" ,
  //是否自动播放 默认值0 (0: 不自动, 1: 自动播放)
  "auto_play" : "0" ,
  //屏蔽全屏播放标识,默认值0 (0: 支持全屏播放,1: 禁用全屏播放)
  "disable_full_screen" : 0,
  //开启防盗链后 支持输入视频地址播放,格式同构造方法输入
  "videos" : null,
  //非点播用户 , 切换视频源使用下面参数, 同构造方法输入
  "third_video" :null
});
```

addBarrage(barrage)

功能：添加弹幕

参数：Array；barrage 弹幕信息

返回：int；操作结果返回码

示例：

```
player.addBarrage([
  {
    "type" : " content" ,//消息类型 ,content:普通文本 (必 选)
    "content" : " hello world" ,//文本消息 (必 选)
    "time" : " 1.101" ,//单位秒 ,表示 距离当前调用添加字幕的时间多久后 ,开始 显示该条字幕
    (必选) },
    { "type" : " content" ,//同上条 (必 选)
      "content" : " MD35号字居中" ,//同上条 (必 选)
      "time" : " 2.101" ,//同上条 (必 选)
      "style" : "C64B03;35" ,// 分号分割 ,第一项颜色值 ,第二项字体 (可 选)
      "postion" : " center" //固定位置 center: 居中 , bottom: 底部 , up: 顶上 (可 选) }, ... ])
```

closeBarrage () 功能：关闭弹幕，调用 addBarrage 后，弹幕将继续开启 返回：int；操作结果返回码

closeBarrage ()

功能：关闭弹幕，调用 addBarrage 后，弹幕将继续开启

返回：int；操作结果[返回码](#)

- **返回码**

- 200: 操作成功
- 0: 播放器未完全初始化
- -1: 动态更换失败视频，缺少必要参数
- -2: 未知操作命令
- -3: 播放时间超出有效播放范围

5. 视频文件上传功能

用户可使用点播上传 Web SDK，以帮助腾讯云视频用户通过 Web 上传视频文件。

该 SDK 当前支持 HTML5 上传（不支持 HTML5 的浏览器暂不支持上传）

具体操作方法请阅：<http://video.qcloud.com/sdk/upload.html>

6. 视频文件操作 API 说明

关于如何对视频文件进行操作，包括上传，发布等内容，请参考官网链接：

<http://wiki.qcloud.com/wiki/v2/CreateVodTags>

7. SDK 升级历史信息和文档变更

版本号	发布日期	说明
1.0	2015.5	第一版本，基本播放功能；
1.1	2015.6	新增 视频拖动播放位置 事件
1.1	2015.6	新增 视频全屏/退出全屏 事件

1.1	2015.6	新增 支持防盗链 调用播放器
1.2	2015.6.23	新增 支持非点播用户使用播放器
1.3	2015.7	新增 文件上传功能；播放器禁止拖动和拉伸视频至播放器大小
1.3.1	2015.8.24	新增 限时播放功能，以及 是否继续上次播放进度播放 功能 增加试用配置 / 播放记忆功能
1.3.2	2015.08.31	增加试看结束回调事件
1.3.3	2015.09.15	增加弹幕展示/增加修改 flash 显示层级 Wmode/增加拉伸图片贴片高宽至播放器大小 stretch_patch/增加关闭弹幕的接口
1.3.3	2015.11.23	点播用户使用防盗链传 URL 的参数

8. 反馈和建议

腾讯云用户可以通过提交工单或联系客服的方式，对使用中发现的问题和意见建议进行反馈，我们的技术人员会同您联系。如果上面的 API 方法不能满足您的需求，如果您有好的建议和想法，都可以邮件我们的攻城狮 trumpli@tencent.com。

—— 文档结束 ——