

目录

基本介绍.....	1
JS 调用方案.....	2
JS 文件存储路径.....	2
jQuery/Zepto 插件版调用方案	4
使用 HTML5 播放器的自设计 UI 皮肤	4
显示微信公众账号热门视频.....	6
播放器尺寸自适应(响应式).....	7
使用 seajs 或 requirejs 加载	7
demo 示例.....	8
对象和参数说明.....	8

基本介绍

腾讯视频通用播放器是一个使用 JS 封装好的 web 端视频播放器框架，提供视频播放的基础 API，方便调用方在任意网页嵌入腾讯平台的视频或者网络直播。

能够自适应：

设备类型：PC 和移动设备

支持的操作系统：windows、Mac、ios、Android

支持的浏览器：兼容 IE6\7\8\9\10\11，QQ 浏览器、TT 浏览器、遨游、Firefox、safari、opera、chrome 等主流浏览器

播放内核：FLASH、HTML5、P2P_控件

公司任何业务需要调用播放器，都可以直接引用 JS 代码即可，无需关心设备的播放特征，统一播放器能自动根据当前的操作系统、设备、浏览器选择最合适的播放器内核。



JS 调用方案

如果您不想关心细节，只想快速调用腾讯视频统一播放器，请使用[调用代码自动生成程序](#)

腾讯视频统一播放器调用主要分为两个步骤：

步骤 1：调用播放器的页面需要引入腾讯视频播放器 JS 文件；

步骤 2：创建腾讯视频统一播放器对象和视频对象，设置属性，输出播放器；

JS 文件存储路径

JS 存放路径

腾讯视频 JS 存放路径

JS 路径	说明	内核				播放类别	
		flash	HTML5	控件	MP4 链接	点播	直播
http://imgcache.gtimg.cn/tencentvideo_v1/tvp/js/tvp.player_v2.js	完全版(内核是异步按需加载)， 推荐使用	✓	✓	✓	✓	✓	✓
http://imgcache.gtimg.cn/tencentvideo_v1/tvp/js/tvp.player_v2_3g.js	m.v.qq.com 专用版		✓			✓	
http://imgcache.gtimg.cn/tencentvideo_v1/tvp/js/tvp.player_v2_mobile.js	移动设备点播版本	✓	✓			✓	
http://imgcache.gtimg.cn/tencentvideo_v1/tvp/js/tvp.player_v2_live.js	直播版本	✓	✓	✓			✓
http://imgcache.gtimg.cn/tencentvideo_v1/tvp/js/tvp.player_v2_jq.js	jQuery 插件版本(内核异步按需加载)，仅支持 jQuery1.5 以上版本	✓	✓	✓		✓	✓
http://imgcache.gtimg.cn/tencentvideo_v1/tvp/js/tvp.player_v2_zepto.js	Zepto 插件版本(内核异步按需加载)	✓	✓			✓	

说明：

整个步骤为：

1. 必须引入腾讯视频播放器 JS
2. 创建 `tvp.VideoInfo` 类（待播放的视频信息）的对象实例，并传入视频 `vid` 或者直播频道 `id`。
3. 创建 `tvp.Player` 类（播放器信息）的对象实例。
4. 通过播放器对象实例的 `create` 接口设置输出目标容器 ID 和播放器的属性。

点播调用方案

首先，先在页面添加一个 `div`，设置 `id`，当做播放器的容器，后面播放器将输出到这个 `div` 里：

```
<div id="mod_player"></div>
```

第 2 步，调用腾讯视频统一播放器 API 输出播放器：

```
var video = new tvp.VideoInfo();

video.setVid("i001216370f");
```

```
var player =new tvp.Player();

player.create({

    width:480,

    height:360,

    video:video,

    modId:"mod_player", //mod_player 是刚刚在页面添加的div 容器

    autoplay:true

});
```

注意：

ios 系统使用 HTML5 播放，这个基本没啥问题。而安卓由于其系统版本碎片化问题严重，每个厂商的系统都有自己的定制，所以 web 上的视频播放兼容性问题非常复杂，所以可能会有部分安卓设备无法在 web 播放视频，这个安卓适配问题我们一直在优化，但只能是逐渐解决，可能无法解决所有机型，这将是一个长期的工作。目前针对安卓：

- 1. 安卓 4.0 以下的设备，除 QQ 浏览器和微信 4.2+ 以外，全部使用 flash 播放
- 2. 安卓 4.0 上，优先使用 HTML5，如果当前设备接口告知不支持 HTML5，这继续使用 Flash
- 3. 如果您的设备满足以上条件，但仍然无法播放，那么请确认您的设备是否支持 flash 或者 HTML5。如有疑问，可以邮件 popotang，需要您同时反馈您访问的 URL，视频 vid，以及设备机型、设备 os 版本、浏览器版本，浏览器 userAgent。

直播调用方案

首先，先在页面添加一个 div，设置 id，当做播放器的容器，后面播放器将输出到这个 div 里：

```
<div id="mod_player"></div>
```

第 2 步，调用腾讯视频统一播放器 API 输出播放器：

```
var video = new tvp.VideoInfo();

video.setChannelId("18091377");

var player = new tvp.Player();

player.create({

    width: 480,

    height: 360,

    type: 1,

    video: video,

    modId: "mod_player",

    autoplay: true

});
```

注意：

- 1.ios 的 HTML5 直播是使用 HLS 技术，该技术只有 ios 系统才支持，所以用 chrome、safari 修改 userAgent 模拟是看不到效果的，可能会黑屏。
- 2.安卓手机基本不支持 HLS，则只能使用 flash 播放直播，如果您使用的安卓设备不支持 flash（能安装 flash 跟支持 flash 是两回事），则无法播放直播节目，可以找 wwjs 使用安卓 SDK 播放直播节目。

jQuery/Zepto 插件版调用方案

腾讯视频统一播放器自身可以作为 jQuery 或者 Zepto 的插件运行。插件版本有专门的 JavaScript文件。

JS 路径	编码格式	说明
http://imgcache.gtimg.cn/tencentvideo_v1/tvp/js/tvp.player_v2_jq.js	UTF-8	jQuery 插件版本
http://imgcache.gtimg.cn/tencentvideo_v1/tvp/js/tvp.player_v2_zepto.js	UTF-8	Zepto 插件版本

插件版调除了可以使用上述标准的调用以外，还可以使用建议的调用方案：

```
$(".mod_player").createTVP({  
  
    video:{vid:"i001216370f"},  
  
    width:450,  
  
    height:300  
  
});
```

使用 HTML5 播放器的自设计 UI 皮肤

由于 HTML5 的播放是使用 video 标签。而 video 标签的解析完全是有浏览器或者操作系统底层来控制，那么 video 标签表现出的 UI 自然也由于浏览器或者操作系统的不同而表现出各式各样的差异，比如有些浏览器控制栏是黑色，而有些浏览器的控制栏这则是蓝色或者橙色，交互体验也不尽相同，以及各自实现中的各种 bug 引起播放过程中各种各样的体验问题，同时由于缺少标题、清晰度切换等按钮，默认的系统皮肤也无法满足我们的特性需求。所以腾讯视频使用 div+css3 自己实现了一套控制栏皮肤。



要使用自定义的皮肤，只需要两步：

1. 在调用页面引入皮肤 CSS 文件：

```
http://imgcache.gtimg.cn/tencentvideo_v1/mobile/v2/style/play.css
```

2. 在 create 播放器的时候，将参数：isHtml5UseUI 设置为 true 即可，例如：

```
var player =new tvp.Player();  
  
player.create({
```

```
width:480,  
  
height:360,  
  
video:video,  
  
modId:"mod_player",  
  
isHtml5UseUI:true  
  
autoplay:true  
  
});
```

显示微信公众账号热门视频

统一播放器集成了播放器非全屏状态下暂停或结束时显示微信公众账号热门推荐视频功能。



要在播放器显示微信公众账号热门视频，首先场景是微信中的公众账号页面，根据页面 url 参数取得公众账号 id; 微信中支持检测是否安装腾讯视频 app，通过检测是否安装腾讯视频 app 来处理热门视频的跳转链接。然后在 create 播放器的时候，使用参数：plugins: {AppRecommend?: true}即可，例如：

```
var player = new tvp.Player();

player.create({

  width:480,

  height:360,

  video:video,
```



```
modId:"mod_player",

autoplay:true,

plugins: {

    AppRecommend: true

}

});
```

当然也可以同上边的 AppBanner?一样设置 text 参数，配置当前视频专属的文案。

注意：部分设备型号由于系统环境原因，无法检测 ended,pause 事件，暂不支持显示推荐视频广告。已知设备有: 1.小米 1+微信+andriod4

播放器尺寸自适应(响应式)

统一播放器 2.0 支持尺寸自适应，配合调用页面的响应式布局，只需要创建的时候将 width 和 height 设置为 100%即可。

注意：设置 100%以后，其尺寸由页面父元素控制，详情请参考 DOM 和 CSS 规范。

使用 seajs 或 requirejs 加载

统一播放器可以直接引用 JS 文件调用接口，也可以作为 commonjs 模块，通过 seajs 或者 requirejs 加载。

以 seajs 加载为例：

seajs alias 配置

```
seajs.config({

    alias:{

        "tvp":"http://imgcache.gtimg.cn/tencentvideo_v1/tvp/js/tvp.player_v2.js"

    }

});
```

我们创建一个 demo 模块，改模块通过 require 引用统一播放器

seajs_tvp_demo.js 定义：

```
define(function(require, exports, module) {

    var demo = {

        initplayer: function(modId) {

            var $P = require("tvp");

            var video = new $P.VideoInfo();

            video.setVid("c0013mhe2en");

            var myplayer = new $P.Player();

            myplayer.create({

                width: 420,

                height: 280,

                video: video,

                modId: modId,

                autoplay: false
```

```
        });

    }

};

module.exports = demo;

});
```

业务逻辑调用：

```
seajs.use("seajs_tvp_demo",function(obj){

    obj.initplayer("mod_player");

})
```

demo 示例

点播示例

在页面输出一个普通的播放器，播放指定的视频：[查看 demo](#)

在页面输出一个播放器，增加事件监听和回调，无刷新切换多个视频：[查看 demo](#)

直播示例

在页面输出直播播放器，播放指定的直播频道:[查看 demo](#)

对象和参数说明

腾讯视频播放器框架是完全采用面向对象的思想使用 javascript 编写，为了防止变量冲突，框架提供的 API 都统一在 window.tvp 命名空间下。 框架通过两个核心的对象接口实现对视频的播放。

这里简单介绍下统一播放器最常使用的两个核心对象和接口参数

tvp.VideoInfo 视频信息

tvp.VideoInfo 是视频信息对象，作用是对准备播放的视频信息属性的描述，例如视频的 vid、直播频道的频道 id、视频时长、视频起止时间、视频标题.....

接口列表：

函数名称	用途
setVid(vid::string)	设置视频的 vid
setChannelId(cnId::string)	设置直播频道的频道 id
getVid()	获取视频的 vid
getChannelId()	获取直播频道的频道 id
setTagStart(iStart::number)	设置看点开始时间
setTagEnd(iEnd::number)	设置看点结束时间
setTitle(sTitle::string)	设置视频标题
getVideoSnap()	获取视频截图，返回一个数组，包含了几种规格的视频截图 URL 地址

tvp.Player(width,height) 播放器

tvp.Player 是播放器对象，提供对播放器的描述和属性设置，例如播放器是直播还是点播、播放器类别（控件、Flash 还是 HTML5）、loading 动画、默认图片、皮肤.....

tvp.Player 接受两个参数作为构造函数，width 表示播放器的宽度，height 表示高度，单位为像素 px。

接口列表：

接口名称	描述
create(object::config)	创建播放器，其中 config 是传入的设置参数 JSON，具体参数列表见 这里
pause()	暂停播放
play()	play 接口无参数时，表示继续播放，一般用于 pause() 以后的继续播放
play(video::tvp.VideoInfo)	上面 play() 接口的多态形式，有参数时表示播放指定的视频，主要用于无刷新切换正在播放的视频
getPlaytime()	获取当前播放的时间，返回单位是秒
switchToOcx()	从 flash 播放器切换到控件播放器
switchToFlash()	从控件播放器切换到 flash 播放器

create 接口接受的配置参数列表

参数名称	描述
autoplay	是否自动播放
coverId	专辑 id
flashWmode	flash 播放器的 wmode
height	默认高度，单位像素
html5FeatureExtJS	HTML5UI 组件功能异步加载 JS 配置，有些组件由于不是必须，而代码量又很大，所以采用按需加载 配置是 JSON 格式， key 是组件名， value 是异步加载的 JS 文件路径
html5ForbiddenUIFeature	HTML5 播放器 UI 需要关闭的功能，跟上述的配置相反，这里列出的功能就不会展现
html5LiveUIFeature	HTML5 直播播放器 UI 组件
html5Preload	Html5 播放器 preload 属性 已废弃
html5VodUIFeature	HTML5 点播播放器 UI 组件
isHtml5AutoBuffer	Html5 播放器是否使用 autobuffer 属性
isHtml5ControlAlwaysShow	HTML5 播放器是否一直显示控制栏
isHtml5ShowPlayBtnOnPause	暂停的时候是否显示播放按钮
isHtml5ShowPosterOnChange	HTML5 播放器切换视频的时候是否要显示 Poster
isHtml5ShowPosterOnEnd	HTML5 播放器播放完毕是否显示 poster
isHtml5ShowPosterOnStart	HTML5 播放开始的时候是否显示 poster
isHtml5UseAirPlay	Html5 播放器是否使用 Airplay 功能，强烈建议开启
isHtml5UseFakeFullScreen	是否强制使用伪全屏
isHtml5UseUI	HTML5 播放器是否使用自设计的控制栏
isiPhoneShowPosterOnPause	iPhone 在暂停的时候是否显示 Poster 层
isLiveFlashShowCfg	直播播放器是否显示配置菜单
isLiveFlashShowConfigBtn	直播播放器是否显示设置按钮
isLiveflashShowFullBtn	直播播放器是否显示全屏按钮
isOcxHideControl	控件是否隐藏控制栏
isOcxShowPauseBtn	控件播放器暂停的时候是否在画面上显示暂停按钮
isVodFlashShowCfg	flash 点播播放器是否出现设置按钮

isVodFlashShowEnd	flash 点播播放器播放结束出现结束推荐
isVodFlashShowNextBtn	flash 点播播放器是否出现“下一个视频”按钮
isVodFlashShowSearchBar	flash 点播播放器是否出现搜索框
libpath	统一播放器框架的存放路径
liveFlashAppType	直播播放器皮肤类型，不传入则 flash 播放器内部会默认为 live weidiantai:微电台，weidianshi:微电视，live:腾讯直播，inlive:公司内部直播
liveFlashExtVars	直播播放器扩展 flashvars
liveFlashSwfType	直播播放器类型
liveFlashUrl	直播播放器 swf 的 url 地址
liveFlashWatermark	直播播放器右上角水印图片
loadingSwf	loading 动画的 swf 地址
modId	默认的 DOM 元素 ID
mute	是否静音
ocxControlBar	控件控制栏地址
ocxControlHeight	控件控制栏高度
ocxSkin	控件皮肤
oid	广告订单 id
pic	默认 loading 图片
playerid	播放器 id，不指定的话系统会自动分配一个，一般不需要配置
playerType	播放器类别
plugins	使用插件列表，如果配置在这里那么会在 write 之后自动调用这里列出的插件 当然，用户自己写的插件也可以不用列在这里，直接在外部调用 player.
pluginUrl	插件 JS 存放路径，key 是插件名，value 是插件的 JS 路径，跟下面的 libpath 组合成完成的 URL 地址 如果定义了这里的路径，那么会异步加载，否则会探测当前页面是否有对应的 build+插件名的函数
share	是否显示分享
type	播放器类别
typeId	分类 id
video	默认的视频镀锡 i 昂
vodFlashExtVars	flash 点播播放器扩展 flashvars 参数
vodFlashListType	点播 flash 播放器 listtype 参数
vodFlashSkin	flash 点播播放器皮肤地址
vodFlashType	flash 点播播放器类型
vodFlashUrl	flash 点播播放器地址
volume	默认音量
width	默认宽度，单位像素

其中：

create 接口 **pic** 参数可接受的域名白名单

vpic.video.qq.com

ossweb-img.qq.com
cache.tv.qq.com
imgcache.qq.com
im-img.qq.com
im.qq.com
img1.qq.com
mat1.qq.com
tt.qq.com
img1.gting.com
mat1.gting.com
i.gting.cn
qzonestyle.gting.cn
ctc.qzonestyle.gting.cn
cnc.qzonestyle.gting.cn
cm.qzonestyle.gting.cn
cn.qzonestyle.gting.cn
os.qzonestyle.gting.cn
edu.qzonestyle.gting.cn
img1.qqinternal.com

注意：白名单域名是固定的，暂不接受新增域名，且涵盖了公司绝大部分公司级的 **CDN** 域名，如有图片需要作为默认图显示，可以考虑将图片上传到以上任意域名。

create 接口 **vodFlashExtVars** 可扩展的配置参数

参数	参数值和说明
share	是否显示分享功能入口，默认显示。 0 为隐藏， 1 为显示。
shownext	是否显示下一集按钮，默认不显示。 1 为显示。
light	是否显示开关灯，默认显示。 0 为隐藏， 1 为显示。
popup	是否显示弹出小窗口按钮，默认显示。 0 为隐藏， 1 为显示。
favorite	是否显示收藏入口，默认显示。 0 为隐藏， 1 为显示。
showend	功能同上面的 isVodFlashShowEnd，强制指定是否显示播放结束界面 0 为不显示 1 为显示，默认为显示
searchbar	是否显示顶部搜索框； 1 显示， 0 不显示，默认为 1
searchpanel	是否显示包含搜索、分享等的浮层。 0 ：不显示， 1 ：显示，默认为显示。建议设置搜索框不显示时，结束推荐（showend）也设置不显示。

例如，想针对 **flash** 播放器设置播放完毕不显示相关推荐：

```
player.create({  
  
  vodFlashExtVars: {  
  
    share: 0,  
  
    favorite: 0  
  
  }  
})
```

```
});
```

[点击此处查看 demo](#)

tvp.Player()对象还对应各种事件

参数名称	描述
onwrite	当输出播放器时
oninited	当播放器初始化完毕时，flash 播放器在 swf 文件加载完毕触发，html5 播放器在视频载入完毕触发
onchange	播放器播放的视频发生变化时
onplay	播放器开始播放时
ongetnext	当播放器播放视频完毕，准备获取下一个视频时
onallended	播放器播放完毕时
onpause	播放器触发暂停时，目前只针对 HTML5 播放器有效
onfullscreen(isfull)	播放器触发全屏/非全屏时，参数 isfull 表示当前是否是全屏
onerror	当播放器发生错误时

例如：

```
var video = new tvp.VideoInfo();

video.setVid("i001216370f");

var player = tvp.create({

    width:480,

    height:360,

    video:video,

    modId:"mod_player",

    autoplay:true,

    onwrite:function(){alert("播放器显示完毕");}

});
```