

HTML5: Audio & Video API

- [浏览器支持](#)
- [API 参考](#)
- [DEMO](#)
- [注意事项](#)
- [兼容解决方案](#)
- [参考文献](#)

一、浏览器支持

1. Audio Codecs

Platform	MAC				WIN							
Browser	OPERA	FIREFOX	SAFARI	CHROME	OPERA	FIREFOX		SAFARI	IE			
Version	10.63	3.6	5	7	10.63	3.6	4.03	5	6	7	8	9
Audio: ogg/vorbis	√	√	-	√	√	√	√	-	-	-	-	-
Audio: mp3	-	-	√	√	-	-	-	√	-	-	-	√
Audio: wav	√	√	√	-	√	√	√	√	-	-	-	-
Audio: AAC	-	-	√	√	-	-	-	√	-	-	-	√

2. Video Codes

Platform	MAC	WIN
----------	-----	-----

Browser	OPERA	FIREFOX	SAFARI	CHROME	OPERA	FIREFOX		SAFARI	IE			
Version	10.63	3.6	5	7	10.63	3.6	4.03	5	6	7	8	9
Video: ogg/theora	√	√	-	√	√	√	√	-	-	-	-	-
Video: H.264	-	-	√	√	-	-	-	√	-	-	-	√
Video: WebM	√	-	-	√	√	-	√	-	-	-	-	-

二、API 参考

1. **API SPEC:** <http://www.whatwg.org/specs/web-apps/current-work/multipage/video.html>

- <audio />
- <video />
- <source />
- <track />

2. **HTML 写法:**

```
<video controls>
  <source src="video.mp4" type="video/mp4" />
  <source src="video.ogv" type="video/ogg" />
  <track kind=subtitles src=brave.en.srt srclang=en label="English">
  <track kind=captions src=brave.en.srt srclang=en label="English for the Hard of Hearing">
  <track kind=subtitles src=brave.fr.srt srclang=fr label="Français">
  <track kind=subtitles src=brave.de.srt srclang=de label="Deutsch">
  audio not supported
</video>

<audio controls>
  <source src="demo-audio.ogg" />
  <source src="demo-audio.mp3" />
```

```
audio not supported
</audio>
```

3.Script:

```
var audio = document.getElementsByTagName('audio')[0];
if( !audio.autoplay ) audio.autoplay = true;
```

(1)浏览器检测：

```
try {
    var hasVideo = !! (document.createElement('video').canPlayType);
} catch(msg) {}

audio 同 video。
```

(2)媒体控制：

Function	Behavior
load()	下载媒体源 除非是动态生成的元素，一般情况下并不需要这个来调用源。可以用其来进行预加载
play()	播放
pause()	暂停
canPlayType(type)	可以播放的类型 (MIME type) 检测是否支持 fooType 这样的 MIME type !! (document.createElement('video').canPlayType('fooType'))

(3)属性

属性	属性值	属性值
----	-----	-----

duration	媒体播放时间的长度。返回 Number，单位是秒；如果播放时间不确定，则返回 NaN。
paused	当媒体处于暂停状态，返回 true；当媒体未被播放，默认为 true。
ended	当媒体播放结束，返回 true。
startTime	通常返回 0.0。当是流式媒体，duration 不确定，媒体内容可能不在缓冲区内，返回媒体最早的开始点。
error	当发生错误时，发出警告。
currentSrc	返回一段代表浏览器选择的媒体源文件的 String。
autoplay	返回循环的状态 (Boolean)，或者设置自动播放。
loop	返回循环的状态 (Boolean)，或者设置循环。
currentTime	返回播放开始到现在的时间的秒数，或者设置一个值做为其播放点。
controls	显示是否有控制条 (Boolean)，或者设置隐藏/显示控制条。
volume	设置媒体的音量 (0.0 ~ 1.0)，或者获取音量的值。
muted	设置静音，或者获取当前是否静音的状态。
autobuffer	设置媒体是否在播放之前自动缓冲。如果媒体已经被设置自动播放，则些属性自动被忽略。
poster	返回媒体封面图片 (POSTER)，或者设置其封面图片地址。
width,height	设置或获取媒体的视觉宽高。
videoWidth videoHeight	获取媒体的原始 (intrinsic or natural) 宽高。

三、DEMO

[HTML5 Video](#)

四、注意事项

- 一般下载 video 都需要一段时间，建议给每一个 video 设置一个 Poster

六、兼容解决方案

在 video / audio 标签中包含 Flash。这是目前比较好的 Fallback 方案，如 video 我们通常这样做：

```
<video width="640" height="360" src="http://www.youtube.com/demo/google_main.mp4" autobuffer controls
poster="whale.png">
  <object classid="clsid:02bf25d5-8c17-4b23-bc80-d3488abddc6b" width="640"height="360"
codebase="http://www.apple.com/qtactivex/qtplugin.cab">
    <param value="http://www.youtube.com/demo/google_main.mp4">
    <param value="true">
    <param value="false">
    <embed src="http://www.youtube.com/demo/google_main.mp4" width="640"height="360"
autoplay="true" controller="false" pluginspage="http://www.apple.com/quicktime/download/">
    </embed>
  </object>
</video>
```

五、参考文献

- BOOK: 《Pro HTML5 Programming》
- [The video elemen - HTML5 Draft](#)
- [The Video Element](#)
- [Native Audio In The Browser](#)
- [Introduction to HTML5 video](#)
- [Everything you need to know about HTML5 video and audio](#)
- [HTML5 & CSS3 Support, Web Design Tools & Support](#)