

CinemaVideoPlayer Component API

© 2006/07 TUFaT.com, All Rights Reserved

To use the CinemaVideoPlayer component, you must first install him by opening CinemaVideoPlayer.mxp. You must have the Macromedia Extension Manager, which you can download from www.macromedia.com. After successful installation, CinemaVideoPlayer component should be available in the Flash IDE Components Panel.

This component was by TUFaT.com, in conjunction with the RuAnSoft development team. It is designed to mimic the product demo video feeds of Adobe.com.

- **Resize component:**

Usage

```
cinemavideoplayer_instance.setSize(WIDTH, HEIGHT);
```

Sample

```
cinemavideoplayer_instance.setSize(300, 300);
```

- **Sets the video content to be played. First parameter is the path to the video file, and the second parameter is the video buffer size, in seconds.**

Usage

```
cinemavideoplayer_instance.setVideo(string_value, number_value);
```

Sample

```
cinemavideoplayer_instance.setVideo("video.flv", 5);
```

- **Sets the volume for the video. This parameter must be an integer from 1 to 100:**

Usage

```
cinemavideoplayer_instance.setSoundVolume(number_value);
```

Sample

```
cinemavideoplayer_instance.setSoundVolume(50);
```

- **Stop video:**

Usage

```
cinemavideoplayer_instance.pauseVideo();
```

- **Play video:**

Usage

```
cinemavideoplayer_instance.playVideo();
```

- **Play video from start:**

Usage

```
cinemavideoplayer_instance.rewindVideo();
```

- **Set AutoPlay option for video:**

Usage

```
cinemavideoplayer_instance.setAutoPlay(boolean_value);
```

Sample

```
cinemavideoplayer_instance.setAutoPlay(true);
```

- **Set string that will be visible when video is buffering:**

Usage

```
cinemavideoplayer_instance.setBufferingString(string_value);
```

Sample

```
cinemavideoplayer_instance.setBufferingString(" Buffering...");
```

- **Set string that will be visible when loading/connecting the video:**

Usage

```
cinemavideoplayer_instance.setConnectingString(string_value);
```

Sample

```
cinemavideoplayer_instance.setConnectingString("Connecting...");
```

- **Set array to show “CC” text by time intervals (show caption X at second Y, show caption M at second N, etc.):**

Usage

```
cinemavideoplayer_instance.setClosedCaptions(array_value);
```

Sample

```
var inClosedCaptions=new Array();  
inClosedCaptions.push({caption: "<B>Hello</B>", seconds: 0});  
inClosedCaptions.push({caption: "so<I>me</I> words", seconds: 6});  
cinemavideoplayer_instance.setClosedCaptions(inClosedCaptions);
```

- **Set logo image. This must be a *non-progressive .jpg* image. Be sure to get your relative path correct.**

Usage

```
cinemavideoplayer_instance.setLogo(string_value);
```

Sample

```
cinemavideoplayer_instance.setLogo("logo.jpg");
```

- **Set shadow width. This must be an integer from 2 to 10:**

Usage

```
cinemavideoplayer_instance.setShadowWidth(number_value);
```

Sample

```
cinemavideoplayer_instance.setShadowWidth(5);
```

CinemaVideoPlayer parameters:

AUTO_START	Autostart video
BORDER_SHADOW_COLOR	Setup shadow color
BORDER_SHADOW_WIDTH	Setup shadow width
BUFFER_TIME	Video buffer time in seconds
BUFFERING_STRING	String that will be shown during video buffering
BUFFERING_STRING_BOLD	Buffering string bold style
BUFFERING_STRING_COLOR	Buffering string color
BUFFERING_STRING_ITALIC	Buffering string italic stile
BUFFERING_STRING_SIZE	Buffering string font size
CLOSED_CAPTIONS	Setup array for show text by time intervals
CONNECTING_STRING	Setup string to show while connecting to video
CONTROLS_BAR_COLOR	Control bar color
LOGO_APPEARANCE	Logo alignment
LOGO_PATH	Path to logo file (non-progressive jpg)
LOOPING	Video looping
SCRUBBER_BACKGROUND_COLOR	Progress bar color background
SCRUBBER_COLOR	Scrubber color
VIDEO_FILE_PATH	Path to video file
VOLUME	Sets sound volume

Event handler summary for the CinemaVideoPlayer:

	Invoked when video stream starts playing.
<i>onPlay</i>	<pre>Usage: video_mc.onPlay = function() { // actions go here } Sample: player.onPlay = function() { trace('play');</pre>
	Invoked when video stream is paused.
<i>onPause</i>	<pre>Usage: video_mc.onPause = function() { // actions go here } Sample: player.onPause = function() { trace('pause');</pre>
<i>onEnd</i>	Invoked when video stream ends.

```
Usage: video_mc.onEnd = function() {  
  // actions go here  
}  
Sample: player.onEnd = function() {  
  trace('end');  
}
```

Return broadcasting player time in 0.00 seconds format.

onTime

```
Usage: video_mc.onTime = function(obj:Object) {  
  // actions go here  
}  
Sample: player.onTime = function() {  
  trace('time '+obj.time);  
}
```