

ActiveX

From VideoLAN Wiki

The Windows build of VLC includes an (optionally installed) **ActiveX control**. The ActiveX control enables VLC to be embedded in web browsers and third-party applications.

Contents

- 1 Important
- 2 Properties
- 3 Methods
- 4 Options
- 5 Samples
- 6 Installing
 - 6.1 Introduction
 - 6.2 Creating an VLC ActiveX Distribution
 - 6.3 .cab-file Integration
- 7 See Also

Important

The API described in this page only reflects VLC ActiveX controls prior to 0.8.5.1. This API will be removed soon.

It is **not advised** to use this JS API any longer.

Please use the VLC ActiveX v2 interface as described in the new documentation.

The new JS API interface is exactly the same for Internet Explorer, Mozilla/Firefox and Safari. Thus easing the maintenance and developing of webpages for the VLC browser plugins.

Properties

The ActiveX control includes the following properties:

<i>name</i>	<i>type</i>	<i>get or set</i>	<i>description</i>
Length	Integer	get	Returns length of the current clip.
playlistCount		get	Returns the count of items in the playlist
playlistIndex		get	Returns the index of the current item in the playlist.
AutoLoop	Boolean	get/set	Determines if the player should automatically loop when it finishes the current playlist.
AutoPlay	Boolean	get/set	Determines if the player should start playing a new file/playlist immediately upon being loaded.

Volume	Integer	get/set	Current volume (scaled from 0 to 100)
MRL	String	get/set	Presumably returns the MRL of the currently loaded file.
Time	Integer	get/set	Time elapsed in seconds playing current MRL, 0 for live feed
showdisplay	Boolean	get/set	show/hide control viewport
Playing	Boolean	get	Returns whether some MRL is playing
Position	'real'	get/set	Playback position within current MRL, scaled from 0.0 to 1.0. Live feed returns 0.0
VersionInfo	String	get	Returns version and build information.

Note: In Visual Basic, type "Long" should be used for properties listed with type "Integer".

Methods

The ActiveX control includes the following methods (functions):

<i>name</i>	<i>type</i>	<i>description</i>	<i>syntax (VB)</i>
setVariable	method	Assigns a value to a variable that is defined in libvlc.c	controlname.setVariable name as String, value
getVariable	method	Returns the contents of a variable that is defined in libvlc.c	x = controlname.getVariable(name as String)
pause	method	Pauses the currently playing clip	controlname.pause
play	method	Plays as in the normal player, if a clip is not loaded, does nothing.	controlname.play
playFaster	method	Makes the currently playing clip play faster.	controlname.playFaster
playSlower	method	Makes the currently playing clip play slower.	controlname.playSlower
stop	method	Makes the currently playing clip stop.	controlname.stop
shuttle	method	Moves the playback position a specified number of seconds in either direction.	controlname.shuttle(seconds as Long)
playlistClear	method	Clears the playlist	controlname.playlistClear
playlistNext	method	Goes to next item in the playlist	controlname.playlistNext
playlistPrev	method	Goes to previous item in the playlist	controlname.playlistPrev
addTarget	method	Adds a uri to the current playlist or replaces the current playlist with the uri.	controlname.addTarget(uri as String, options, mode as VLCPlaylistMode, Position as Long)
toggleMute	method	Toggles between the muted/unmuted state of the embedded player.	controlname.toggleMute
fullscreen	method	Toggles between fullscreen and non-fullscreen modes.	controlname.fullscreen

Options

The `addTarget` method accepts most of the command line options; including, but not limited to, the following :

```
:audio-track=index
:vout-filter=output-filter (deinterlace for example)
:deinterlace-mode=mode (linear for example)
```

Samples

Delphi Implementation

HTML Implementation

Installing

Introduction

VideoLAN is not a good source for the installation of VLC through an ActiveX control. At some point, Microsoft Internet Explorer stopped allowing the installation of ActiveX controls unless:

- they were signed to associate the software vendor's name with the file containing the ActiveX control, or
- they resided in an "trusted site" (from the user's perspective, running Internet Explorer)

The second option is only feasible where both server and client infrastructure are managed under the same umbrella, i.e. where the server providing the ActiveX control can be declared "trusted". This is not feasible in the open web. In a web configuration, a Certificate Authority (http://en.wikipedia.org/wiki/Certificate_authority) needs to digitally sign the ActiveX control. This comes at a cost which the VideoLAN project is not able to bear. The solution is to create one's own ActiveX control distribution, which in turn can be digitally signed by a Certificate Authority of your choice, if necessary.

Creating an VLC ActiveX Distribution

The following requisites are required:

CABSDK by Microsoft to create a Cabinet File (http://en.wikipedia.org/wiki/Cabinet_%28file_format%29) (.cab-file) containing the relevant ActiveX control and VLC files. MSDN has a good description how to package an ActiveX Control (<http://msdn2.microsoft.com/en-us/library/aa751974.aspx>) , including a link to the CABSDK download page.

The .cab-file is a special form of a .zip file, consisting of:

- axvlc.inf - The INF file, which would be called manifest in the modern age
- axvlc.dll - ActiveX DLL which bootstraps the VLC setup file
- vlc-0.8.6e-win32.exe - VLC setup file (refer to your version)

I am using the following code in the INF file:

```
;; Version number and signature of INF file.
;
[version]
```

```

:signature="$CHICAGO$"
:AdvancedINF=2.0

[Add.Code]
vlc-0.8.6e-win32.exe
axvlc.dll=axvlc.dll

[axvlc.dll]
FileVersion=0,8,6,0
clsid={9BE31822-FDAD-461B-AD51-BE1D1C159921}
RegisterServer=no
hook=nsiinstaller

[vlc-0.8.6e-win32.exe]
FileVersion=0,8,6,0
file-win32-x86=thiscab

[nsiinstaller]
run=%EXTRACT_DIR%\vlc-0.8.6e-win32.exe

```

Again, refer to your version of VLC.

You can extract axvlc.dll from the binary distribution of VLC. This is a zip file on VideoLAN's download page, version 0.8.6.e here (<http://downloads.videolan.org/pub/videolan/vlc/0.8.6e/win32/vlc-0.8.6e-win32.zip>) .

The regular VLC setup file is the "featured" download on VideoLAN's main page; a copy resides in the same directory as axvlc.dll, the VLC setup file of version 0.8.6.e here (<http://downloads.videolan.org/pub/videolan/vlc/0.8.6e/win32/vlc-0.8.6e-win32.exe>) .

With CABSDK installed and the three components in place, you can create the .cab-file using the CABARC tool from Microsoft's CABSDK, using the following command from a command prompt:

```
CABARC.EXE N axvlc.cab axvlc.inf axvlc.dll vlc-0.8.6e-win32.exe
```

This assumes the PATH (http://en.wikipedia.org/wiki/Path_%28computing%29) has been set to CABARC.EXE's directory. If you need to sign the .cab-file, you need to use the -s switch to allocate space for your digital certificate.

.cab-file Integration

The .cab file is integrated as described in the samples above (<http://wiki.videolan.org/ActiveX#Samples>) , using the <OBJECT> element in the case of HTML, for example. When the ActiveX Control is called for the first time, i.e. prior to VLC installation through any means, the setup program should start up. Unfortunately, users have to click through a few dialogs, "allow blocked contents" or deal with other inconveniences, depending on your configuration and the user's security settings of Internet Explorer. The user must have proper rights to install VLC on the user's computer; identical to a regular installation of VLC as required by the VLC setup program.

See Also

Official ReadMe (<http://git.videolan.org/?p=vlc.git;a=blob;f=projects/activex/README.TXT>)

Retrieved from "<http://wiki.videolan.org/ActiveX>"
Category: Embedded

This page was last modified on 27 May 2009, at 09:49.