VLC FOR UNITY INTEGRATION README
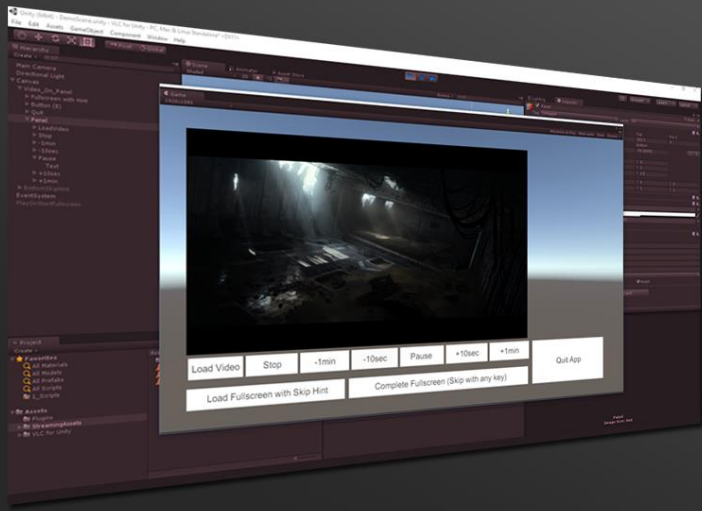
Thanks for purchasing this asset. Please read this file for an easy introduction on how everything works.
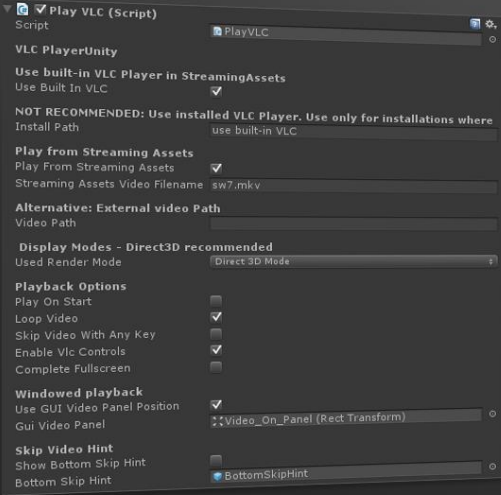

Introduction and Supported Media

VLC for Unity makes it easy to play any kind of video file in your Windows builds with the power of VLC. **You can use any video codec, filetype, bitrate or resolution** that VLC supports. Since the video will be played on your GPU, even 4K videos should be handled easily by most PCs.

Operating Systems supported (builds and editor): Windows 7, 8, 10. Other operating systems may be supported later on as updates.

Project Settings: These are the recommended project settings for the plugin.

Particularly important is *Running in Background* and using the *.NET 2.0* API compability setting – otherwise needed plugins can't be loaded.
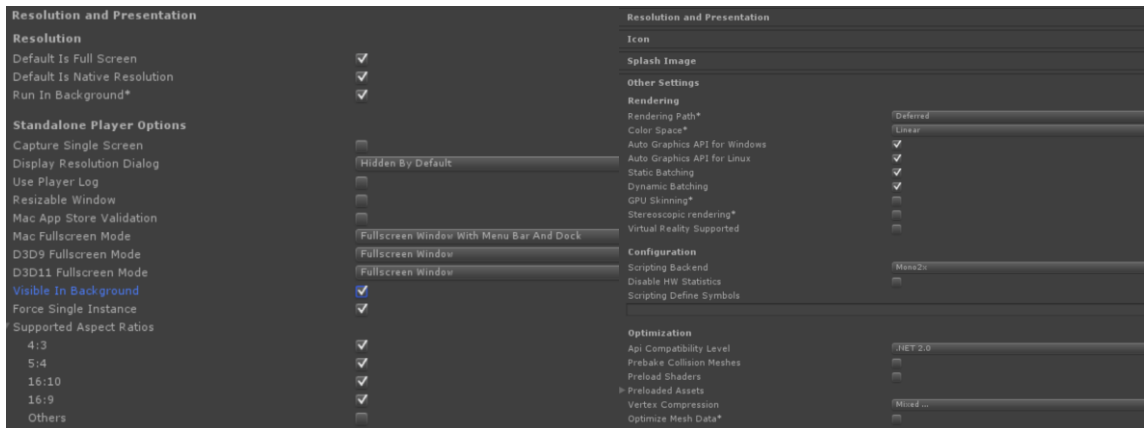
Requirements for proper playback in Editor

When testing your videos *windowed on GUI elements* in the Editor, please make sure of the following things. **This does not apply to builds, in your builds everything will look properly ALL THE TIME.**
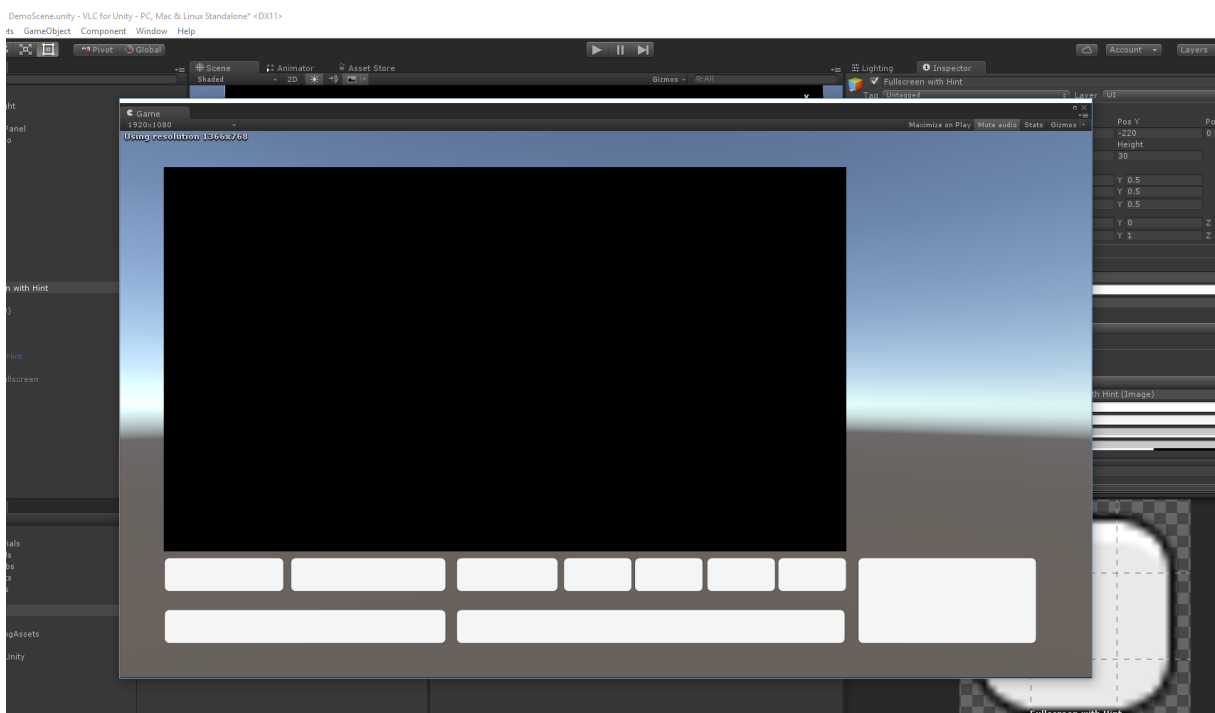
- Drag the game view from the editor.

- Make sure you set the resolution to **Free Aspect OR**..
- …if using a **specific resolution**, resize the window so **no gray overflow area is around** your game view in the game window.

If you don't do these steps, the windowed video *might be a bit off in the editor* view. I'll try to fix this issue for the editor, but again: everything will always look right in builds.

Play a video with VLC for Unity

To simply play a video:

- Assign the PlayVLC.cs Script to any GameObject.
- Make sure "Use Built in VLC" is checked
- Put any video in the StreamingAssets folder if you want to include the video in your build. Write the filename into the "Streaming Assets Video Filename"-Field and check "Play from Streaming Assets"-Box
- OR: Play the video from any location and paste the path to the video in the "Video Path"-Field while making sure "Play from Streaming Assets" is unchecked
- Check the "Play On Start" and "Skip Video With Any Key"-Box
- Start to play video

For a quick overview, open the **DemoScene** that comes with this plugin.

Playing a video on GUI Elements

ATTENTION: Make sure the *pivot is in the upper left corner* or use the prefab that comes with the package. This will be fixed later.

Follow these steps to easily place a video on UI-Panels in Unity:

- Create a UI Panel and put it to the desired location / size
- Assign the PlayVLC.cs script to the Panel or any other GameObject in your scene
- Make sure the **pivot** of the Panel is in the **upper left corner**, or use the Panel prefab that comes with the plugin
- Check these settings below and assign the **RectTransform** of the Panel to the Script
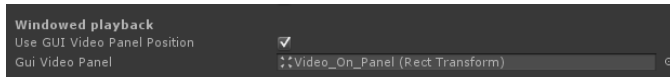
-
- Optional: Check "Enable VLC Controls" box and use UI Buttons to control the video
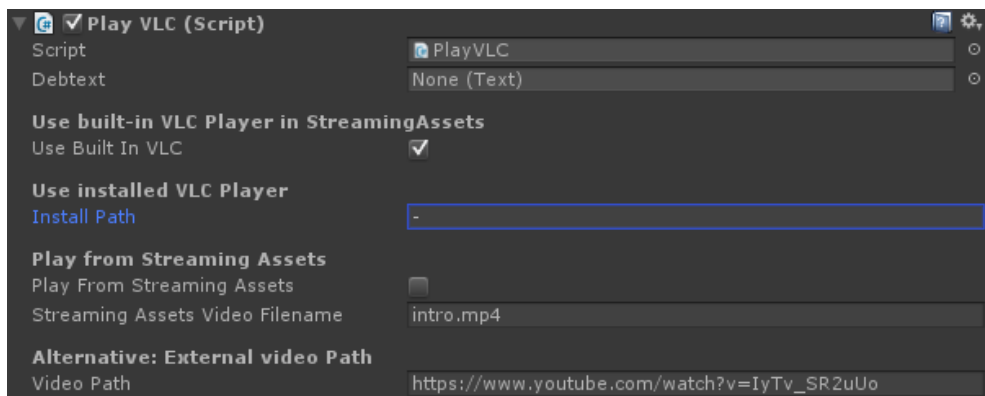
Just use a copied YouTube URL and put it into the Video Path field.

You can play a video behind your actual Unity window, so your UI and 3D Elements will remain visible and interactable, while a video plays in the background (pretty much like in the Fallout4 Main Menu).

ATTENTION: This works only in Builds at the moment!

To do this, check *VideoInBackground* and assign the *VideoInBackgroundCameraPrefab* to the PlayVLC.cs script instance.

Take a look at the **"DemoScene_VideoInBG"** and the "All Settings Overview"-Part of this readme below.

To play a video from script, you can call the *Play()* function of a PlayVLC-Script Instance.

To stop a video, call the stop *StopVideo()* function. The video stops automatically after playing, if looping is set to false. Otherwise you need to call the stop function or allow skipping by users.

If you enable video controls with the *"Enable Vlc Controls"* checkbox, you can additionally call the following functions from scripts or UI elements:

Pause(): pauses the video and resumes it when called again

SeekForwardShort(): jumps 10 seconds ahead

SeekForwardMedium(): jumps 1 minute ahead

SeekForwardLong(): jumps 5 mins ahead

SeekBackwardShort(): jumps 10 seconds back

SeekBackwardMedium(): jumps 1 minute back

SeekBackwardLong():   jumps 5 mins back

ToggleMute():   Turns audio on and off.

VolumeDown(): reduces audio volume by 5%

VolumeUp(): increases audio volume by 5%

```
More controls are coming! If you need something really fast, send me an email and I'll
see what I can do!
```

QuitVLCHelper.cs

When you want to kill all videos, you can use this helper script. Comes in handy if you want to:

- Change a scene while playing a video or multiple videos and you don't stop them yourself
- Quit the app while playing a video or multiple videos and you don't stop them yourself

Just call *QuitAllVideos()* and *QuitApplication()* for this.

All Settings Overview

---

bool UseBuiltInVLC: Use a special VLC version that will be included in your builds. This is **strongly recommended**, since you can't know if or which VLC your users will have installed.

string InstallPath: Put the install path to vlc.exe here (Most often C:\Program Files\VideoLAN\VLC\vlc.exe). This VLC player will be used if `UseBuiltInVLC` is **false**. Use only for installations where you can install a proper version that you can test yourself (many VLC builds have different issues that could interfere with the Unity Integration).

UPDATE: since 1.03, everything should work OK with all VLC versions installed.

bool PlayFromStreamingAssets : Play videos from StreamingAssets folder. These videos will be copied to your build automatically. Recommended option.

string StreamingAssetsVideoFilename : Name of the file to play – **"name.ext"**

string VideoPath: If *PlayFromStreamingAssets* is false, use this path to the video file. It can be located anywhere, but must of course be accessible for the user.

UsedRenderMode: You can choose from 3 Options
- **Direct3D**: Best option, highly recommended. This is the most performant solution.
- **Fullscreen Overlay on Primary Display**: Works only on primary display. **[In development]**
- **QT Interface:** When you need to use full VLC options with interface. **[In development]**

bool PlayOnStart: Play on startup? For intros.

bool LoopVideo: Loop video. Make sure the user is able to skip the video. If false, video stops automatically.

bool SkipVideoWithAnyKey Skip videos with any key on your keyboard or button on gamepad. You should use this for fullscreen videos, where you don't have any other possibility to skip. Note: videos skip automatically after playback, except when *LoopVideo* is true.

bool EnableVlcControls: Allows you to call control functions for the last active video. See demo scene and the explenations on the functions on the previous pages.

bool CompleteFullscreen: If this is set to **false**, fullscreen videos will be rendered atop of Unity's window. This means: if you're running in windowed mode, the video will be "fullscreen" atop the game window, but not fullscreen over the whole screen. If this is set to **true**, it will be forced to render across the complete screen. Make sure to activate *SkipVideoWithAnyKey* when using this, especially if *LoopVideo* is true!

bool UseGUIVideoPanelPosition: Render on top of this UI element. Makes it easy to position windowed videos.

GuiVideoPanel: The UI RectTransform that the video will be rendered on. **IMPORTANT**: The pivot must be in the **upper left corner**. Use the prefab that comes with this script if you want. Apply the poster image that you want to use on the Image component of the RectTransform if you wish.

bool ShowBottomSkipHint: Show a hint that tells the user how to skip the cutscene. If true, the BottomSkipHint will be shown.

BottomSkipHint: UI Element that will be shown under the video (you can use the prefab in the package, or use a custom one). See demo scene.

bool FlickerFix: Better performance at video start. Recommended.

bool VideoInBackground: Play video behind your scene. See the readme above.

VideoInBackgroundCameraPrefab: Place the prefab that comes with this package here, or use your own or modified prefab. This camera keys the background from your scene and makes the Unity window transparent there – which shows the video that is rendered beneath. See the demo scene.

bool UseVlc210OrHigher: Since VLC 2.0.8, there are a few bugs that have not yet been fixed by the VLC team. If you want to use a more recent version, like the built-in 2.2.1 in VLC for Unity 1.03, check this box to ensure proper functionality.

bool PinVideo: Pin the video to the UI its rendered upon, or to the Unity window. If you move either, the window will resize and use the best aspect to fill the new window or UI size automatically.

New versions, planned features, Support and Contact

1.03

- Pin Video to UI / Unity Window: You can now move/resize the video while playing, not just at video start. Best aspect ratio inside the available space is handled automatically.

- Youtube streaming: just put in the URL
- Audio Controls
- Replaced demo video with Youtube URLs in the scripts to save filesize of the plugin.

1.02

- Better performance at video start
- Play video in background.

1.01

- Minor fixes.

Here's what I'm working on and what's planned for the next updates:

- Additional Video controls: Open file, playback speed…
- Render Video to Texture2D
- Playlists
- Shuffle Media/Playlists
- Video Effects
- Add complete folders full of media
- Automatic detection of installed VLC media player and its version

I hope I can get everything working in the near future!

If you have feedback, need help or have ideas what I could improve please write me – I'd like to hear from you! I'll answer as soon as possible.

You can reach me by mail: mail@christianholzer.com

PS: Check out my other stuff on the Asset Store:
https://www.assetstore.unity3d.com/en/#!/search/page=1/sortby=popularity/query=publisher:5166

Thanks again for purchasing,

Christian