AVPro QuickTime™
Unity Plugin

Fast playback of HD video and audio content.

Version 2.8 - Released 12 September 2013

Contents

1. Introduction
2. System Requirements
3. Installation
4. Features
5. Demos
6. Unity Components
7. Tips
8. FAQ
9. Version History
10. Support
11. About RenderHeads Ltd

1. Introduction

AVPro QuickTime™ is a plugin for Unity that allows playback of supported QuickTime content in a fast and easy manner.

The plugin is aimed at the high-end user group that require video playback features beyond Unity's built-in video support.

We see this plugin being useful in the following areas:

- Interactive Installations
- Serious Games
- Kiosks
- Video Apps

QuickTime is a trademark of Apple Inc., registered in the U.S. and other countries.

2. System Requirements

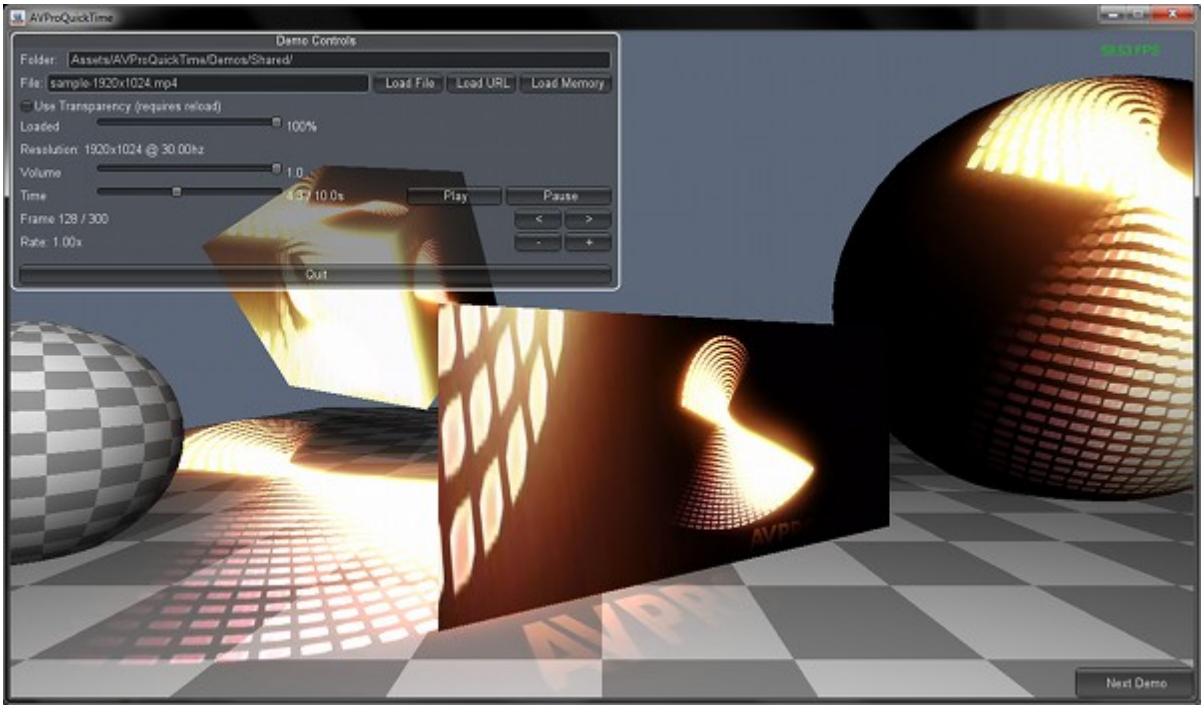
- Windows or Mac.
- Unity Pro 3.5 or Unity Pro 4.0+
- QuickTime 7 or above must be installed on the machine.
- Note: doesn't support the 64-bit Windows build option yet (but still runs on 64-bit Windows machines)

When playing back high resolution videos a decent CPU and GPU must be available. Integrated chipsets may not handle HD content well, especially when using the DirectX rendering path with Unity 3.5.

3. Installation

1. Import the unitypackage file into your Unity project.
2. You may need to **move the "Plugins" folder into the root of your project.**
3. For Windows users: make sure QuickTime is installed on your system

4. Features



4.1 General

- Play multiple QuickTime videos (or audio) simultaneously.
- Audio volume control.
- Frame by frame playback.
- Seeking.
- Playback rate control, including reverse.
- Playback from memory allows loading from a byte[] array.
- Streaming from URL

4.2 Unity Integration

The “AVPro QuickTime” plugin provides an API for playing QuickTime content. Additionally some helpful Unity components have been created to allow drag and drop use of the plugin without any scripting. See the “Unity Components” section below.

Movies are uploaded to standard Unity Texture2D objects.

4.3 Alpha / Transparent Video Support

The plugin supports video codecs that support an alpha channel, allowing playback of transparent videos - something that Unity's native Ogg Theora codec doesn't allow.

Codecs with alpha channel support include:

1. QuickTime native Animation
2. QuickTime native JPEG-2000
3. QuickTime native PNG
4. Resolume DXV (http://www.resolume.com/software/dxv.php)
5. HAP (http://vdmx.vidvox.net/blog/hap)

There codecs also support alpha but they are generally too slow to use for real-time:

1. Apple ProRes4444 http://en.wikipedia.org/wiki/ProRes_422
2. DNxHD Codec
   http://en.wikipedia.org/wiki/DNxHD_codec
   http://www.avid.com/US/industries/workflow/DNxHD-Codec

4.4 Advanced File Loading & Streaming

The plugin supports loading files from:

1. Local file system

   Loading content dynamically from disk allows content to be replaced and updated without relying on having Unity installed. This is especially useful when creating an application that must be maintained/updated by a third party, or for live applications where content is being created while the application is running.

   Another benefit of loading dynamically instead of importing into Unity is the time it takes to import assets. If you have a lot of video content, importing the assets can take a very long time. In this case, loading them dynamically provides a much better workflow.

2. Memory

   Loading from memory allows you to hide your movie content from users. One way to do this is to use Unity's TextAsset loader. The data can then be loaded from a byte[] array.

3. URL (beta)

   Loading from URL allows videos to be streamed. When streaming videos it's

important to have them encoded correctly using the “Internet Fast Start” option and a suitable bit-rate. QuickTime encoder has an option to enable “Fast Start”, as does the FFMPEG command-line tool. URLs can have the prefix “http://” for files hosted on a web server and “file:///” for local files. FFMPEG can be used to add “Fast Start” to an MP4 file via this command:

ffmpeg.exe -i "input.mp4" -c:v libx264 -tune fastdecode -g 12 -crf: 25 -pix_fmt yuv420p -movflags +faststart "output.mp4"

4.5 Video Codec Choice

Video codecs can be chosen to suit the content and playback requirements. Typical requirements:

- Lossless / HQ encoding

  Codecs like PNG and ProRes allow for perfect or near-perfect video encoding.

- Transparency

  See the list of codecs supporting transparency above.

- File size

  Codecs like H.264 give great file size reduction, however not everyone is worried about file size and can choose another codec that is more suited.

- Fast scrubbing / seeking

  Codecs without inter-frame dependencies allow for fast seeking and scrubbing. Typical codecs are: DV, Motion JPEG, PNG, ProRes

- CPU usage

  Some codecs (like H.264) use a lot of CPU during decoding. Other codecs can be used (usually at the expense of disk space) to allow for less CPU usage. XVid is generally provides a good balance of CPU usage and file size.

4.6 Audio & Video Formats

Playback of most of the file formats that QuickTime supports. Including:

Video / Codecs	Image	Audio

QuickTime Movie (mov)	JPEG	MP3
MPEG4 (mp4)	PNG	AAC
Animated GIF (gif)	BMP	WAV (uncompressed)
H.264	JPEG 2000	AU
DV	TGA	MIDI
Microsoft AVI (limited)	TIFF	Apple Lossless
Motion JPEG	GIF	AIFF
3GP & 3G2		
Animation		

4.7 Fast Playback of Full HD 1080p Content

The plugin has been optimised to run as fast as possible to allow for smooth playback of HD content. Various methods have been used to achieve optimal performance.

- **Fast OpenGL Rendering Path**

  Using direct GPU hardware updates, the plugin is able to render very quickly using little CPU power. This path is only available when Unity is running in OpenGL mode.

  You can run your Unity editor in OpenGL mode by adding the "-force-opengl" switch to the shortcut. Builds can also be forced to run in OpenGL mode by using the "-force-opengl" switch on the generated executable files.

- **Fast DirectX Rendering Path**

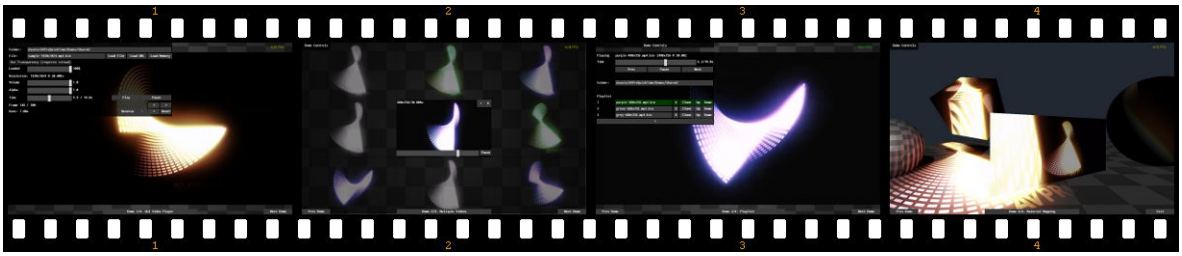
  The DirectX rendering path isn't as fast as the OpenGL path if you're using Unity version lower than 4.0, however we have made it as fast as possible using various techniques. Note: using the DirectX path a 1280x720 video will use use a 2048x1024 texture, as will a 1920x1080 video, so there is often not much difference between them in terms of system load.

In summary here is a table that shows how this plugin performs with different versions of Unity:

	Unity 4	**Unity 3.4 - 3.5 DirectX**	**Unity 3.4 - 3.5 OpenGL**	**Unity < 3.4**
Speed	Fast	Slower	Fast	Slower

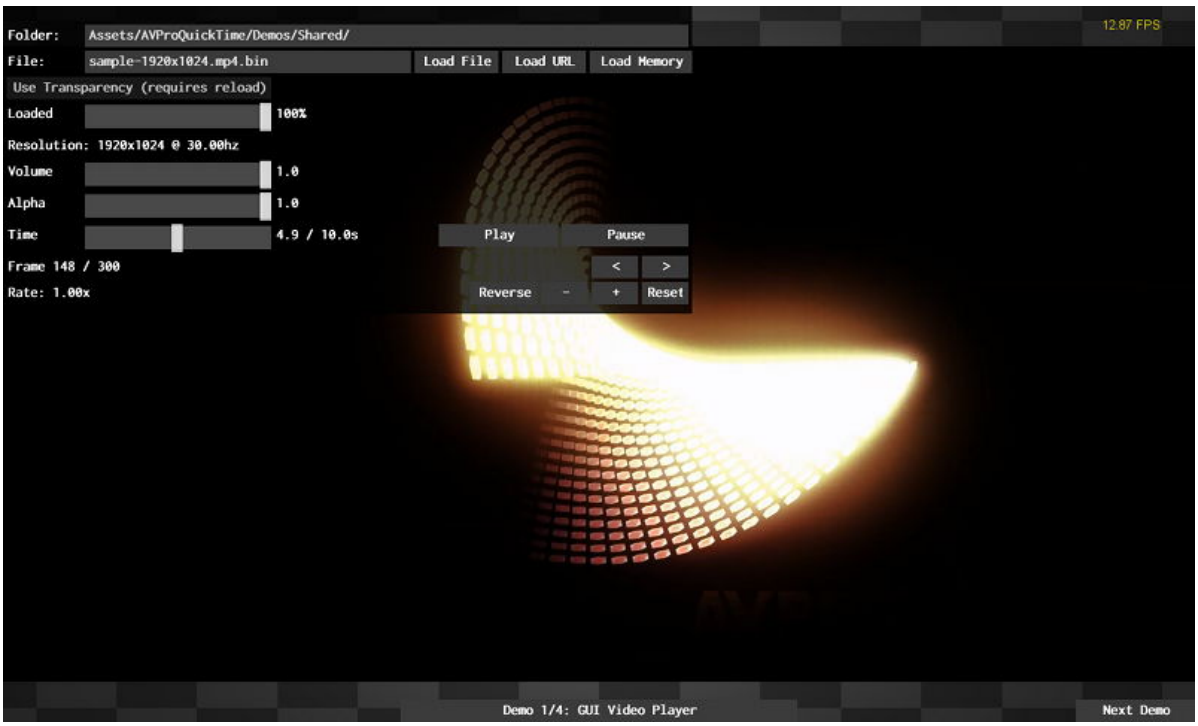
Frame Upload Method	Direct GPU access	Unity Texture2D	Direct GPU access	Unity Texture2D

5.0 Demos

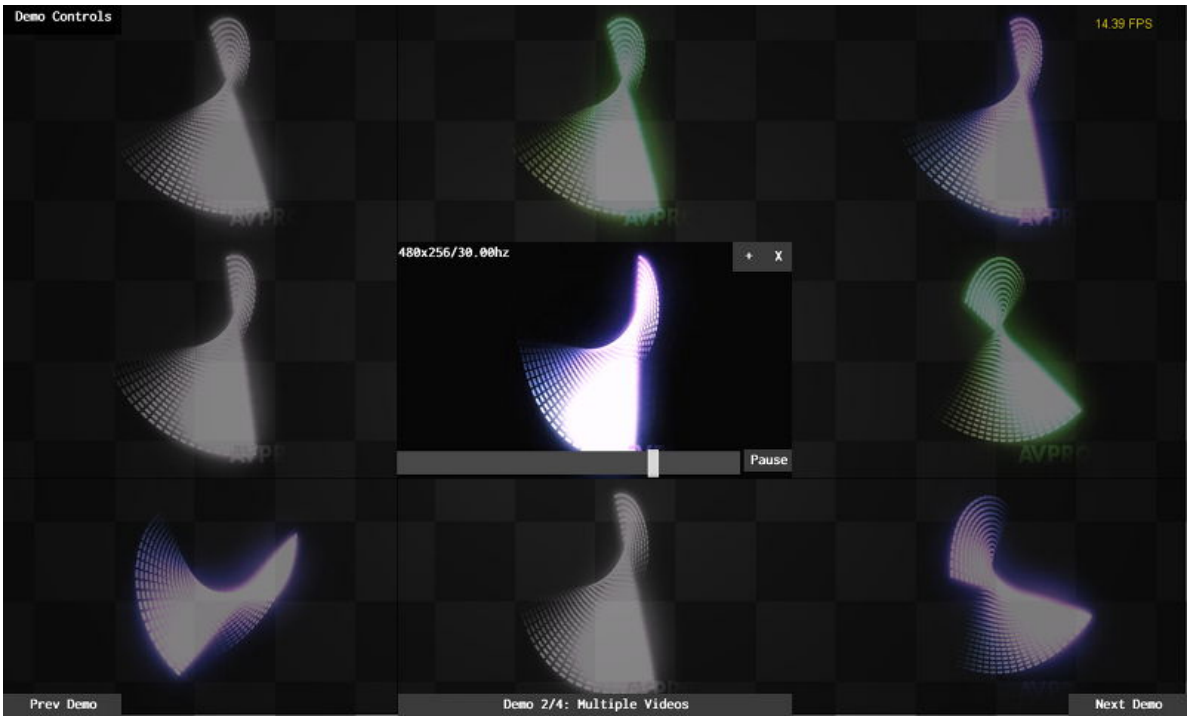


The package comes with 5 demo scenes to get you started.

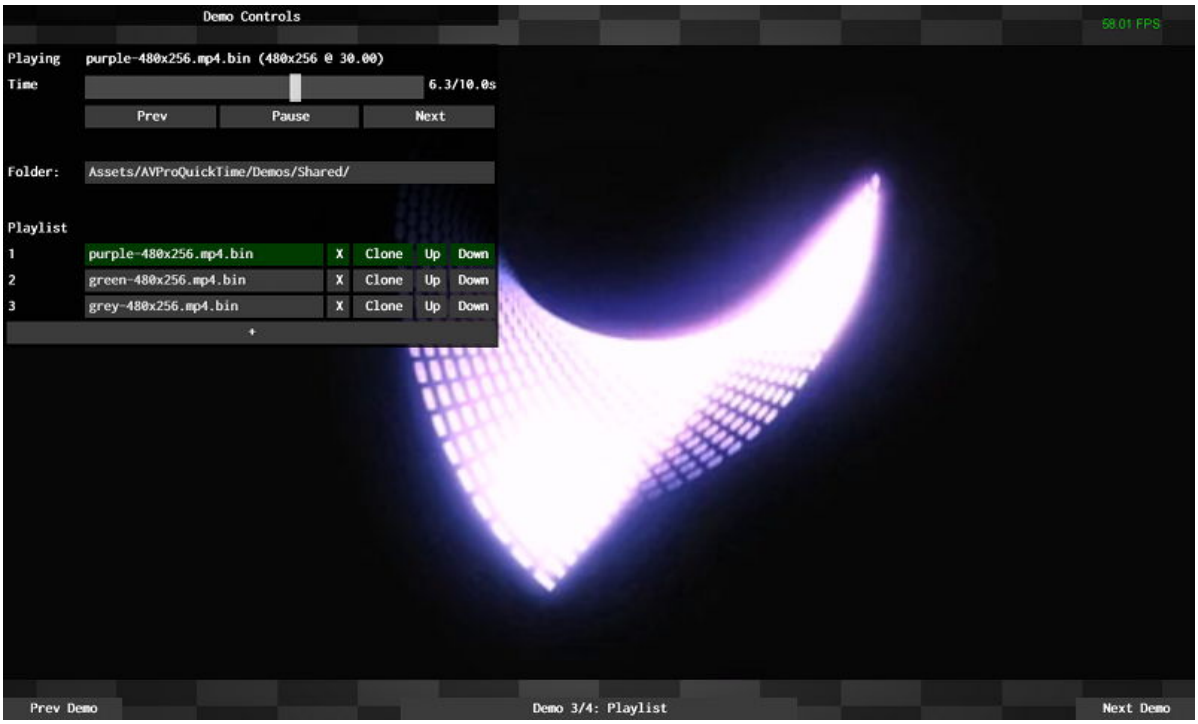
5.1 GUI Video Player Demo



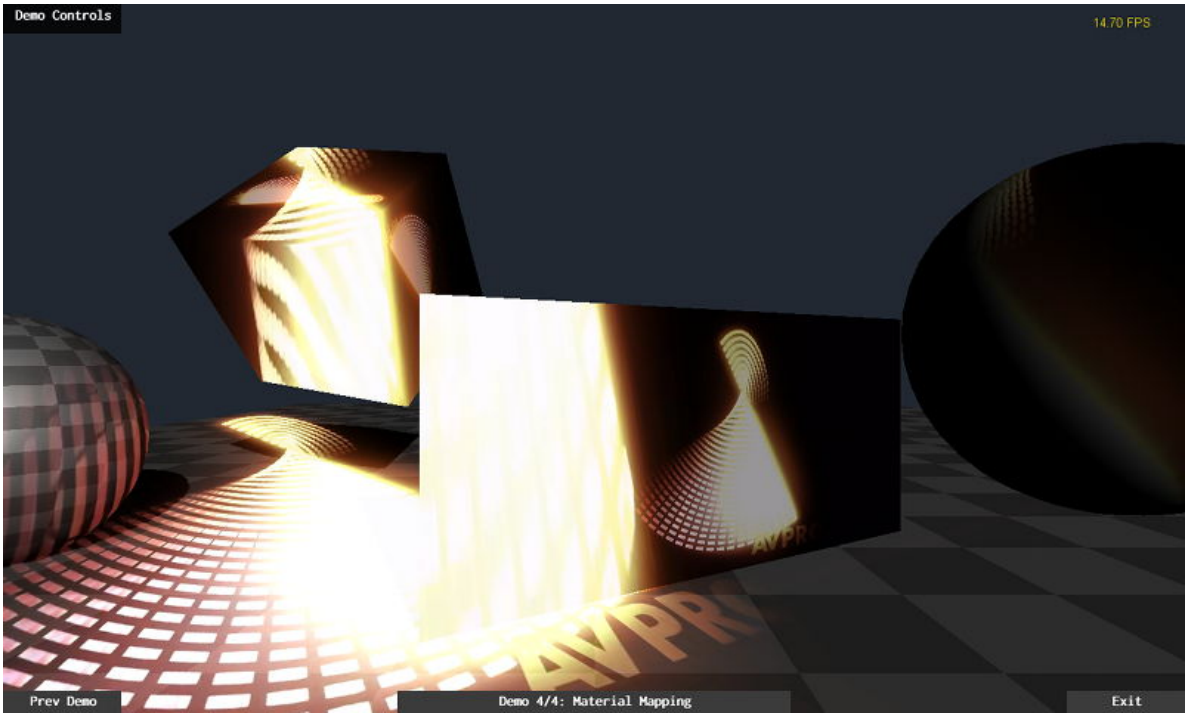
5.2 Multiple Videos Demo



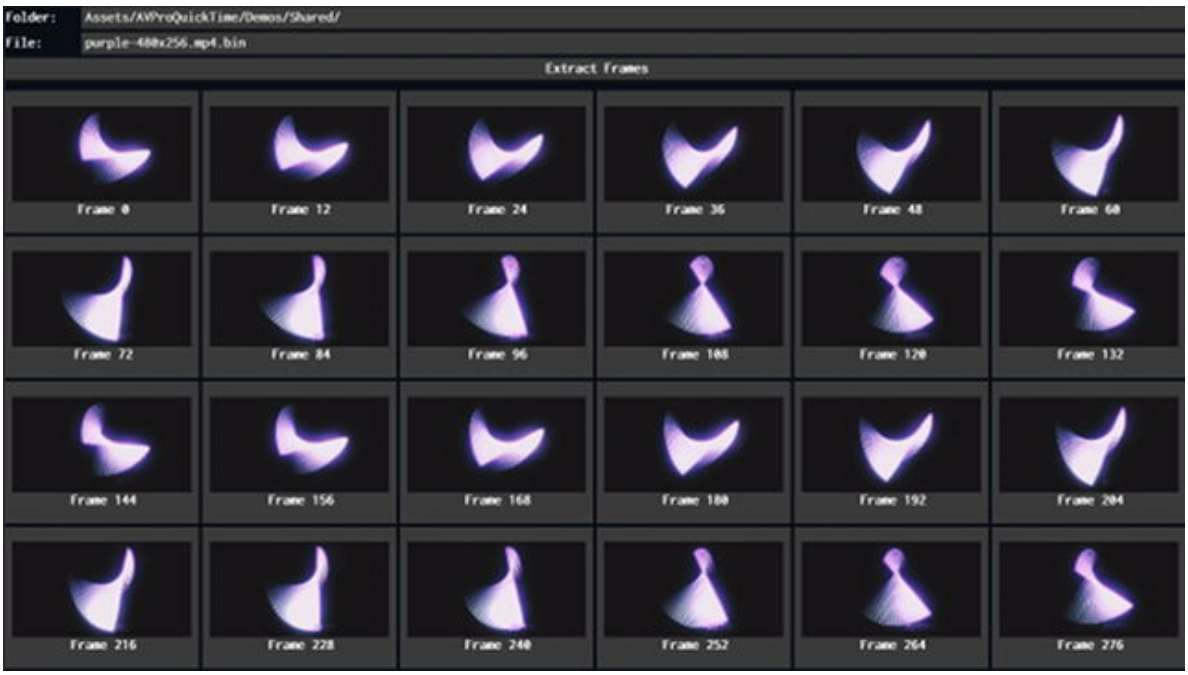
5.3 Playlist Demo



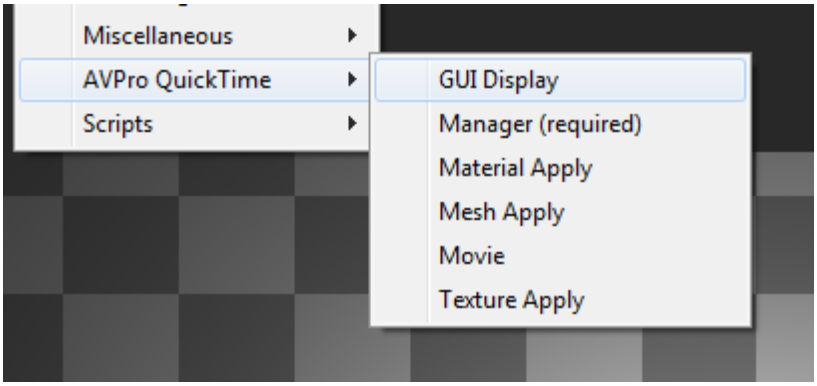
5.4 Material Mapping Demo



4.5 Frame Extract Demo



5. Unity Components



This asset includes a number of Unity script components that allow use of the asset without any scripting.

5.0 Script Order

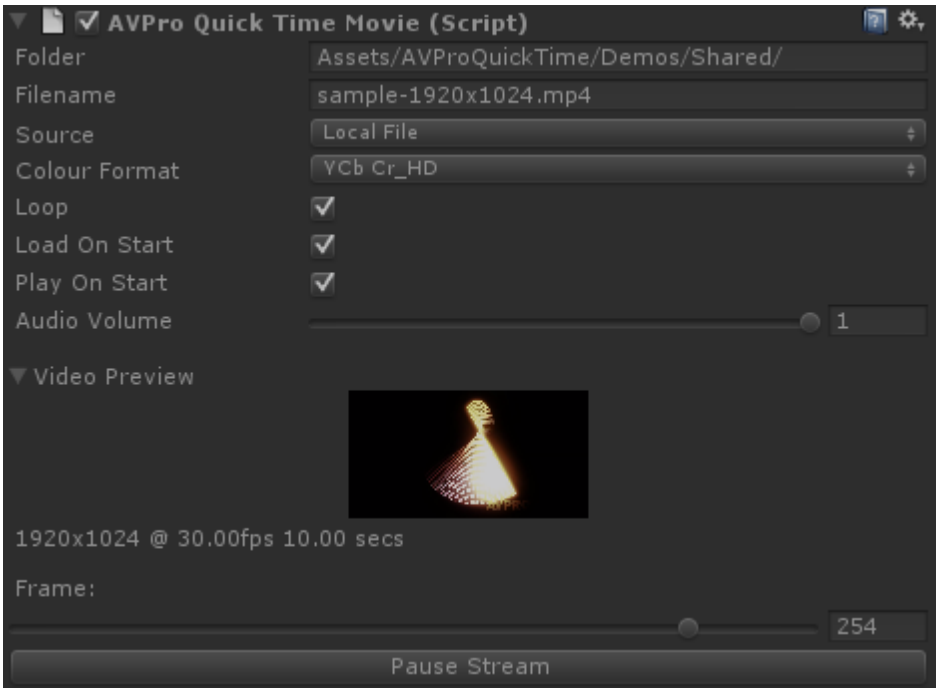
Sometimes the script execution order is important and we recommend this order for our component scripts:

The most important is the Manager script which should always be one of the first in your list. Any of our own scripts that refer to the AVPro QuickTime scripts may have to have their script order explicitly set so they run after the AVPro QuickTime scripts.

5.1 AVProQuickTimeManager

There must always be exactly one **AVProQuickTimeManager** in your scene when you use this plugin. It is also important that this component starts before the other **AVProQuickTimeMovie** components (controllable via Script Execution Order setting). There is nothing to configure in this component.

5.2 AVProQuickTimeMovie



This component represents a single piece of media (video or audio) that can be loaded and played. The colour format is the internal format that is used to play the video and can affect quality and performance. the options are:

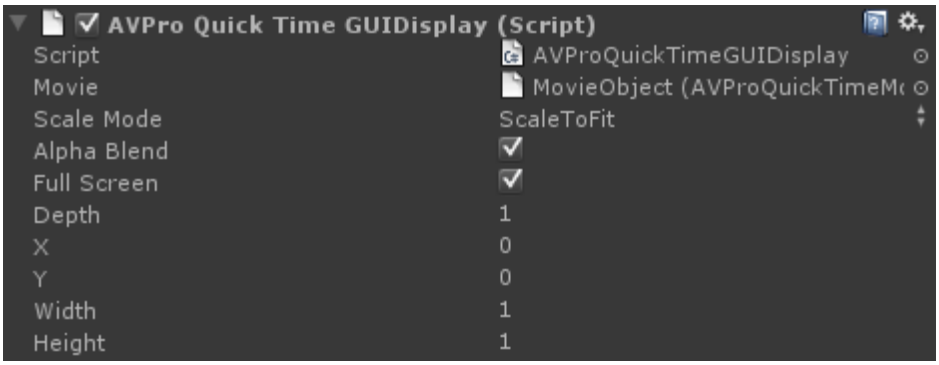
- “YCbCr_HD” (default): Fast playback using the Rec. 709 YUV colour conversion.
- “YCbCr_SD”: Fast playback using the Rec.601 YUV colour conversion.
- “RGBA32”: This mode is slower but it allows for videos with alpha channel.

The colour format cannot be changed once the video is playing.

When the editor is playing additional controls are displayed showing you the contents of the video which is useful during development. Note that having the video preview visible can affect frame rate as it forces update of the Unity UI.

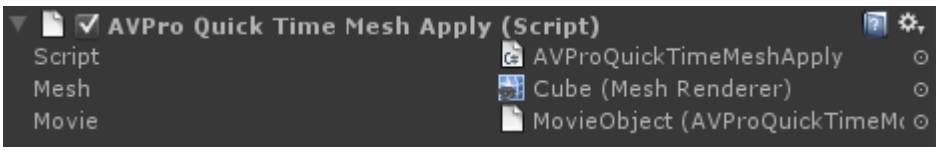
This component simply plays the movie and doesn’t display it on the screen. For display take a look at the components below.

5.3 AVProQuickTimeGUIDisplay



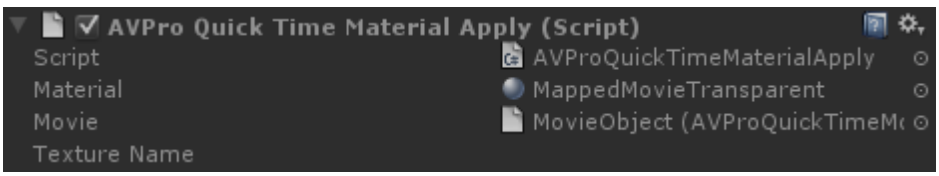
This component displays an AVProQuickTimeMovie on the screen using Unity's GUI system. Simply select the **AVProQuickTimeMovie** component you want to display in the "Movie" option. Next you can set the placement of the item on the screen or use the fullscreen default.

5.4 AVProQuickTimeMeshApply



Use this component to apply an **AVProQuickTimeMovie** to all of the materials used by a mesh in your scene.

5.5 AVProQuickTimeMaterialApply



Use this component to apply an **AVProQuickTimeMovie** to a material in your scene. Optionally a texture name can be specified to override a specific texture slot in the material.

6. Tips

For best results we recommend:

1. If your Unity version is lower than 4.0 then run your application or Unity with “-force-opengl” switch as this plugin runs much faster in OpenGL.

2. If you need to create and destroy many AVProQuickTimeMovie components, it’s best to reuse the existing components as this is faster and involves less memory reallocations.

3. If your Unity version is lower than 4.2 then when using 1080p videos it’s actually best if your input video has a maximum height of <= 1024 as this means the plugin can use a texture with a height of maximum 1024 instead of 2048 in the case of true 1080p.

7. FAQ (Frequently Asked Questions)

1. How do I fix the error: “DLLNotFoundException”?

You need to move/copy the “Plugins” folder from your “AVProQuickTime” folder into the root of your folder structure. This means the “Plugins” folder should be moved to your “Assets” folder. Unfortunately this is a limitation in the way Unitys Asset Store handles plugins.

If you are trying to make a Windows-64bit build you will also get this error message as the plugin doesn’t support 64-bit builds yet.

2. How do I make a 64-bit Windows build?

The plugin doesn’t support 64-bit builds yet.

3. My movie appears too bright or desaturated, how can I fix it?

Use the “YCbCr_HD” colour format instead of “YCbCr_SD”, this will force the plugin to use a different colour conversion routine that is more suitable for your movie.

4. My H.264 encoded video doesn’t play smoothly,, how can I make it play smoother?

H.264 videos (often in a .mov or .mp4 container) are often highly compressed and this can make them slower to play back. The h.264 encoder has many many options that can be used to tune how the videos are compressed and how they play. We’ve found that the tool FFMPEG can be used to convert videos for faster playback with the following command-line:

ffmpeg -i input.mp4 -c:v libx264 -pix_fmt yuv420p -preset veryslow -tune fastdecode

-profile:v main -coder 0 -g 6 -crf 20 -flags -loop output.mp4

5. **Creating a standalone build on the Mac fails to load my movies, however "Load from Memory" works, what's wrong?**

   It seems QuickTime and .Net (used by Load from Memory) look for files in different places by default. QuickTime searches relative to the actual .app folder package and .Net searches relative to the folder the .app is in. To load from disk you need to place your files INSIDE the .app content package.

6. **I keep getting "`System.IO.File' does not contain a definition for `ReadAllBytes'" or similar missing .NET function messages, how can I fix them?**

   This usually happens when your project is set to build a Web Player or another platform that doesn't support certain .NET features. You need to go to File->Build Settings and change the platform to "PC and Mac Standalone".

7. **Which Unity video-playback plugin is better: AVPro Windows Media or AVPro QuickTime?**

   If you need easy cross-platform (PC-Mac) video support then AVPro QuickTime is the only way to go. It is possible to use both plugins together and get the best of both worlds but this would require some scripting to create a basic wrapper to encapsulate both plugins.

   If you are focusing on Windows PC only and need high performance then we recommend using AVPro Windows Media as we have found QuickTime support on PC to be lacking (especially in multi-threaded video codec performance) since the QuickTime engine for PC hasn't been updated in some time.

   QuickTime though can be easier as it only requires a single install whereas on the PC you need to know what codecs you need to support and potential install multiple codecs.

   You can contact us with your requirements if you're not sure. We also have downloadable demos on the website which you can use for testing. Here is a table to help you decide:

	AVPro QuickTime	AVPro Windows Media
Windows PC Support	Yes	Yes
Mac Support	Yes	No
Great performance playing	No	Yes

multiple HD videos on Windows PC		
Requires additional installs	Only on Windows PC - you must install the QuickTime player	Only if you want to support codecs that Windows doesn't natively expose to DirectShow
MP4 / H.264 Playback	Supported	Must install a codec

8. **I have compiled the scripts into a DLL and am now experiencing some unexpected behaviour.**

   Some of our scripts have Unity version-specific preprocessor defines which determine how they compile (eg UNITY_4_0). Usually when you build an external DLL these defines are missing and so the incorrect version of the code can be compiled. You need to add the appropriate compiler defines to your build.

9. **How do I play back HAP encoded videos?**

   Playback of HAP videos requires Unity 4.x or Unity 3.5 (OpenGL mode ONLY). In the AVProQuickTimeMovie component disable "Allow YUV".

8. Version History

- **Version 2.x - Possible future features/enhancements**
  - Add 3D movie support?
  - Add a simple demo that parses media for information and a few frames
  - Add ability to query a movie format to determine whether it has an alpha channel. Then allow the all to select the most appropriate conversion format.
  - ← Your suggestion here

- **Version 2.8 - 12 September 2013**
  - Added support for all 3 high-speed HAP codecs
  - Added automatic transparency detection
  - Simplified colour format selection
  - Added Unity 4.2 and Unity 4.3 support
  - Added Unity 4.1 non-pow2 texture support - optimisation
  - Added new demo that plays multiple videos simultaneously
  - Added new demo that extracts video frames
  - Added Script Order to documentation
  - Fixed alt-tab issue
  - Fixed some issues with streaming radio
  - Fixed issue with releasing video handles

- **Version 2.5.2 - 18 March 2013**
  - Unity 4.1 support added
  - Fixed a few platform #if issues

- **Version 2.5 - 11 March 2013**
  - Added a new demo that plays a queue of videos
  - Improved AVProQuickTimeManager
  - Optimised and improved pixel format conversion
  - Added dynamic setting of loop state.
  - Added path resolve for relative paths when working path isn't that of the EXE
  - Fixed GL.IssuePluginEvents() conflict bug with other AVPro plugins.
  - Fixed Apple App Store submission issues
  - Fixed small memory leak when playing movies from memory.
  - Fixed several bugs

- **Version 2.4 - 20 December 2012**
  - Unity 4.0 support including native DirectX texture updates
  - Improved streaming from URL.
  - Improved performance.
  - Removed overlay mode rendering

- **Version 2.2 - Friday 28 September 2012**
  - Added overlay rendering mode.

    - Improved performance.
    - Improved handling of seeking, frame stepping.
    - Added FPS display in inspector for each movie.
    - Texture no longer updates when video has completed playing.
    - Added option to load the first frame of the video.
    - Fixed bug that caused ProRes codec to crash on exit.
    - Fixed volume slider bug.

- **Version 2.0 - Tuesday 17 July 2012**
    - UTF-8 support added.
    - Easier components for non-scripter.
    - Components have unique names.
    - Improved memory cleanup.
    - Added in-editor preview of videos.
    - Improved demo scenes.
    - Fixed a bug in OpenGL alpha videos.

- **Version 1.9.4 - Monday 18 June 2012**
    - Fixed a bug in GetFrameCount()

- **Version 1.9.2 - Wednesday 3 May 2012**
    - Fixed texture update bug in OpenGL with Unity3.5.
    - Added support for BT709 conversion.

- **Version 1.9 - Friday 17 Feburary 2012**
    - Plugin now also built for Mac.
    - Playback from memory.
    - Playback from URL (beta).
    - Better performance switching between videos of the same resolution.
    - Fixed a resource leak.

- **Version 1.8 - Wednesday 18 January 2012**
    - Fixed a memory leak.

- **Version 1.7 - Wednesday 11 January 2012**
    - Added playback rate control, including reverse.

- **Version 1.5 - Thursday 22 December 2011**
    - Improved stability.
    - Tidied up demos.

- **Version 1.4 - Tuesday 30 November 2011**
    - Added frame-by-frame playback.
    - Updated logo graphics.
    - Added FPS counter to benchmark demo.

- **Version 1.2 - Thursday 17 November 2001**

    - Accepted at Asset Store.
    - Fixed seeking bug.
    - Fixed OpenGL state bug.
    - Improved demo controls.

- **Version 1.0 - Wednesday 26 October 2011**
    - Initial release submitted to Asset Store.

9. Support

If you are in need of support or have any comments/suggestions regarding this product please contact us.

Website: http://www.renderheads.com/contact/
Email: contact@renderheads.com

If you are reporting a bug please include any relevant files so that we may remedy the problem as fast as possible.

10. About RenderHeads Ltd

RenderHeads is an award winning creative and technical company that has been designing and building cutting edge technology solutions since its formation in 2006.

10.1 Services

- Unity plugin development
- Unity game / interaction / augmented reality development
- Unity consulting

10.2 Our Unity Plugins

AVPro QuickTime

AVPro Windows Media

AVPro Movie Capture

AVPro Live Camera