

Developer

/ Documentation

/ Unreal Engine ▾

/ Unreal Engine 5.4 Documentation

/ Working with Media

/ Integrating Media

/ Bink Video for Unreal Engine

Bink Video for Unreal Engine

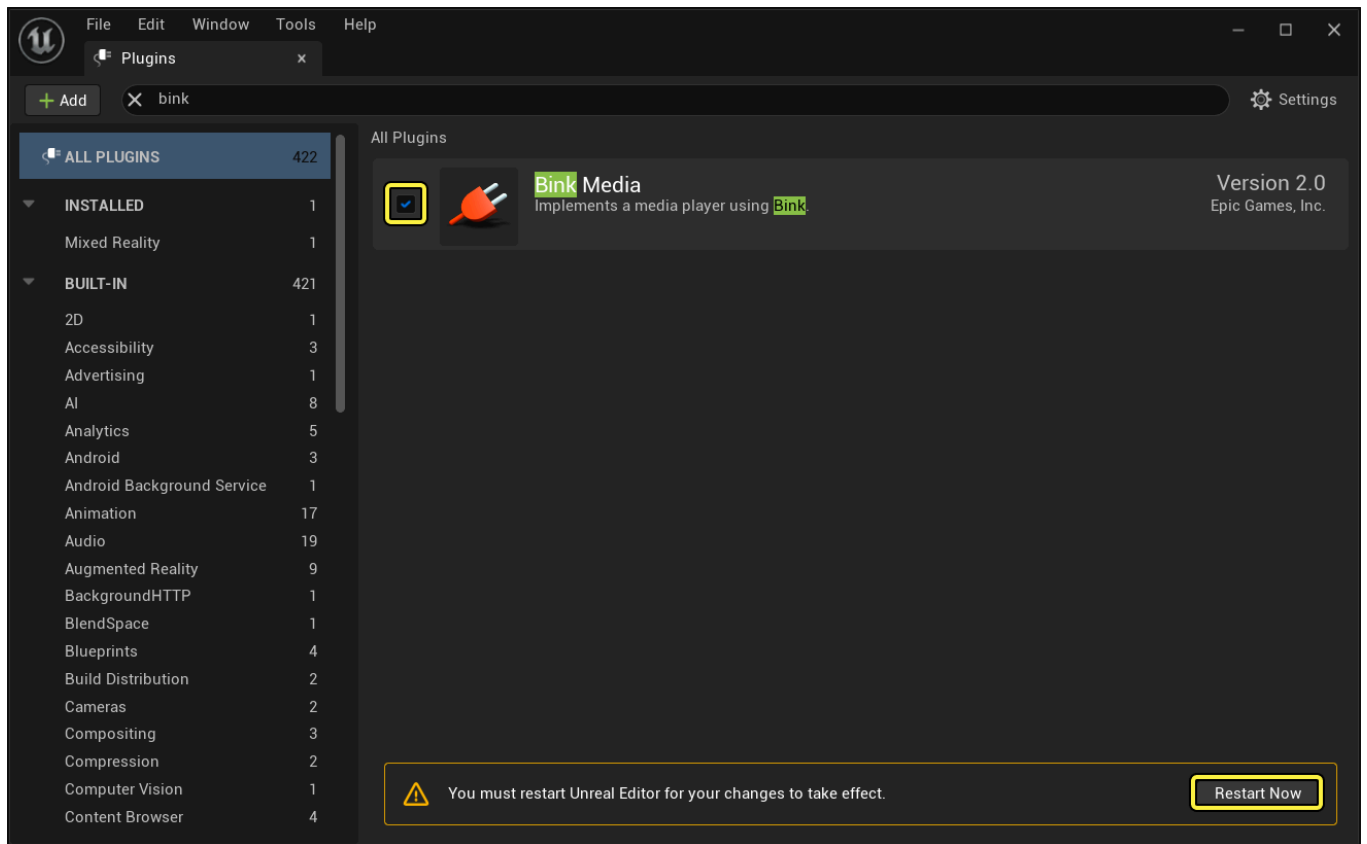
A short overview of using Bink Video to show movies.



The **Bink Media** plugin is built into Unreal Engine 5. All Unreal platforms (including NDA platforms) are supported.

Installation

1. Select **Edit > Plugins** in the Editor, search for **Bink**, and enable the plugin. Restart Unreal Engine as required.

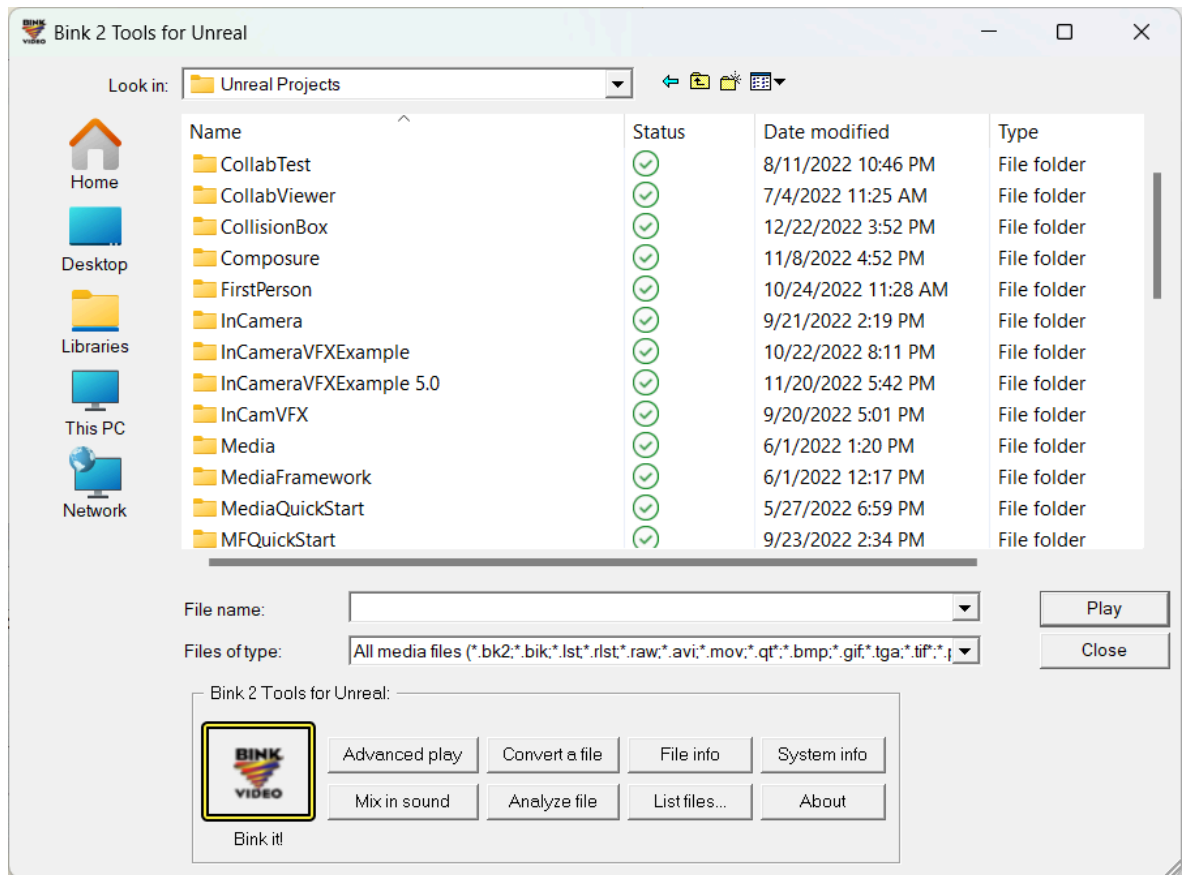


Click image to expand.

2. Copy your Bink video files into the **Contents/Movies** directory in your Unreal Engine Project directory.

You create Bink files using **Bink 2 Encoder for Unreal** found in the `Engine/Binaries/ThirdParty/Bink` directory (Bink2ForUnreal.exe). Double-click the executable, then select your video file to convert and click **Bink it!** then **Bink**. For most use cases, the automatic settings work fine.





- Double-click on the `.uproject` file for your project, and a warning dialog appears and asks if you want to rebuild the project due to missing plugins. Confirm executing the rebuild, and Unreal Engine restarts with the project loaded up.
- You can now play the Bink files in your Content/Movies directory.

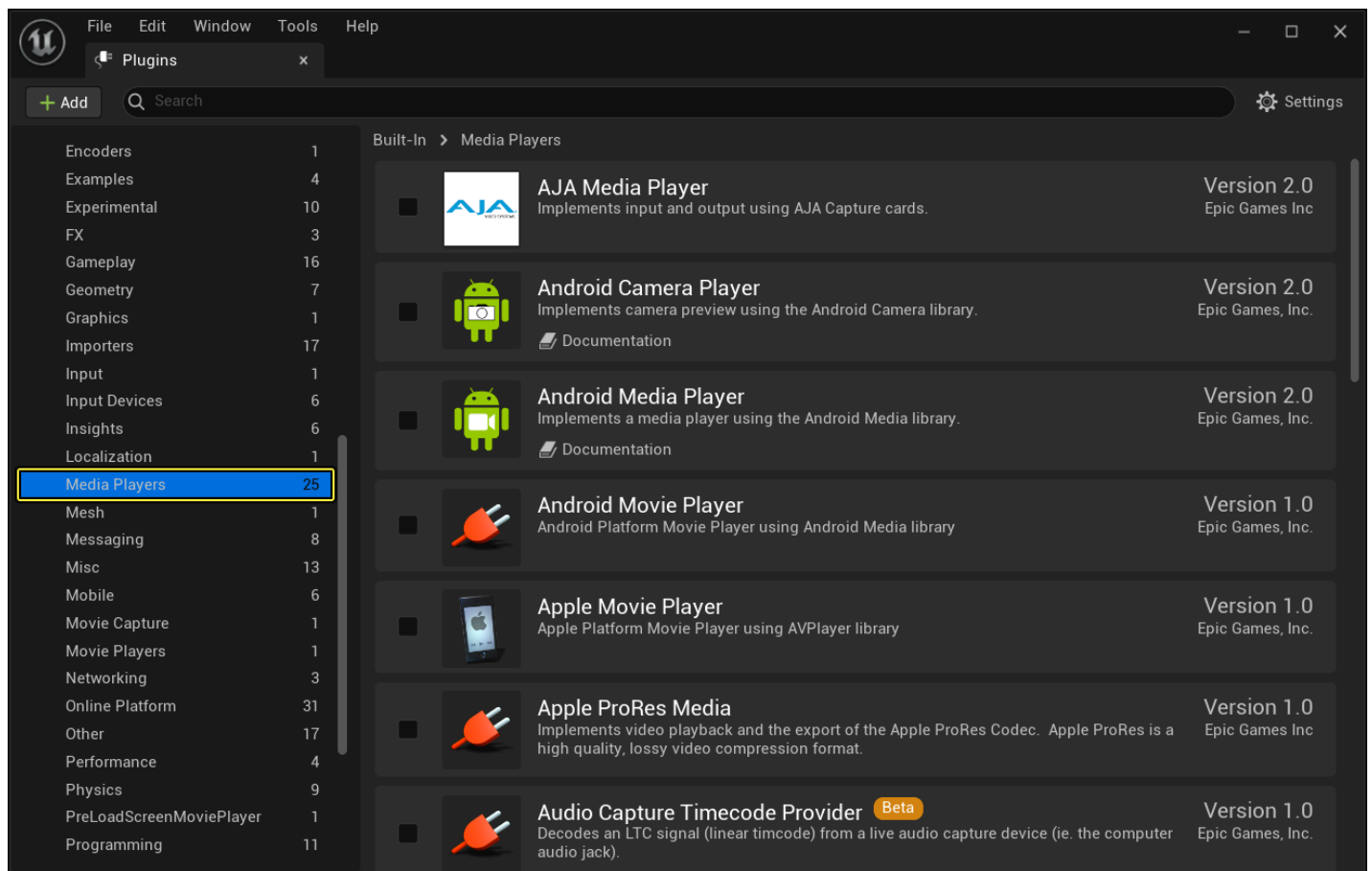
Bink for Unreal Engine Movie Types

Unreal Engine has two distinct types of movies: fullscreen startup movies, and in-game non-startup movies. You use each type a little differently, as described below.

Fullscreen Startup Movie Setup

First, you need to disable all other movie player plugins. Otherwise, any enabled MPEG-4 player tries to play the Bink movie files and silently fails.

To do this, go into **Edit > Plugins**:

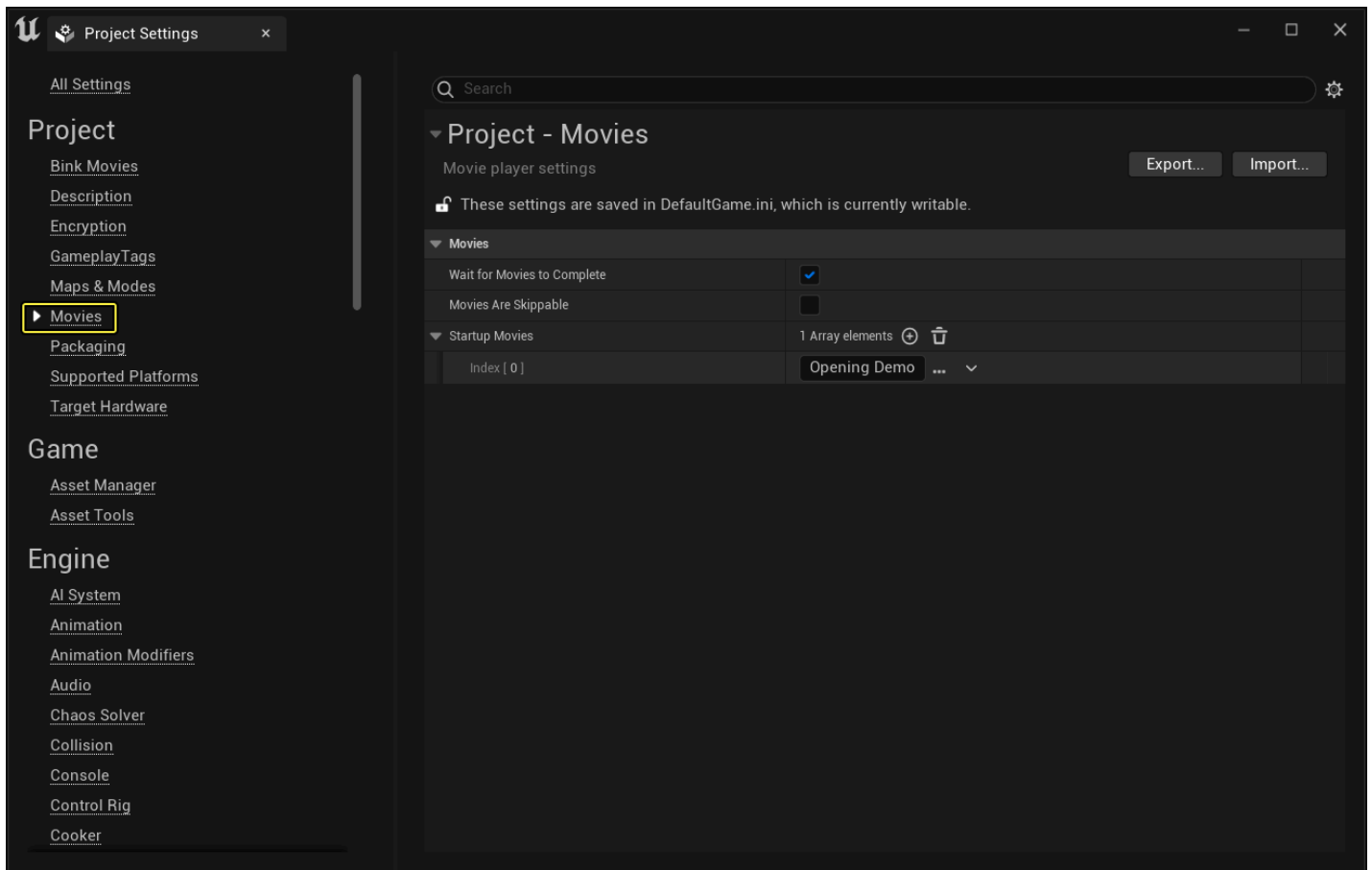


Click image to expand.

Disable all Movie Player plugins other than the Bink Media plugin.

Second, you need to configure some movies to play on startup.

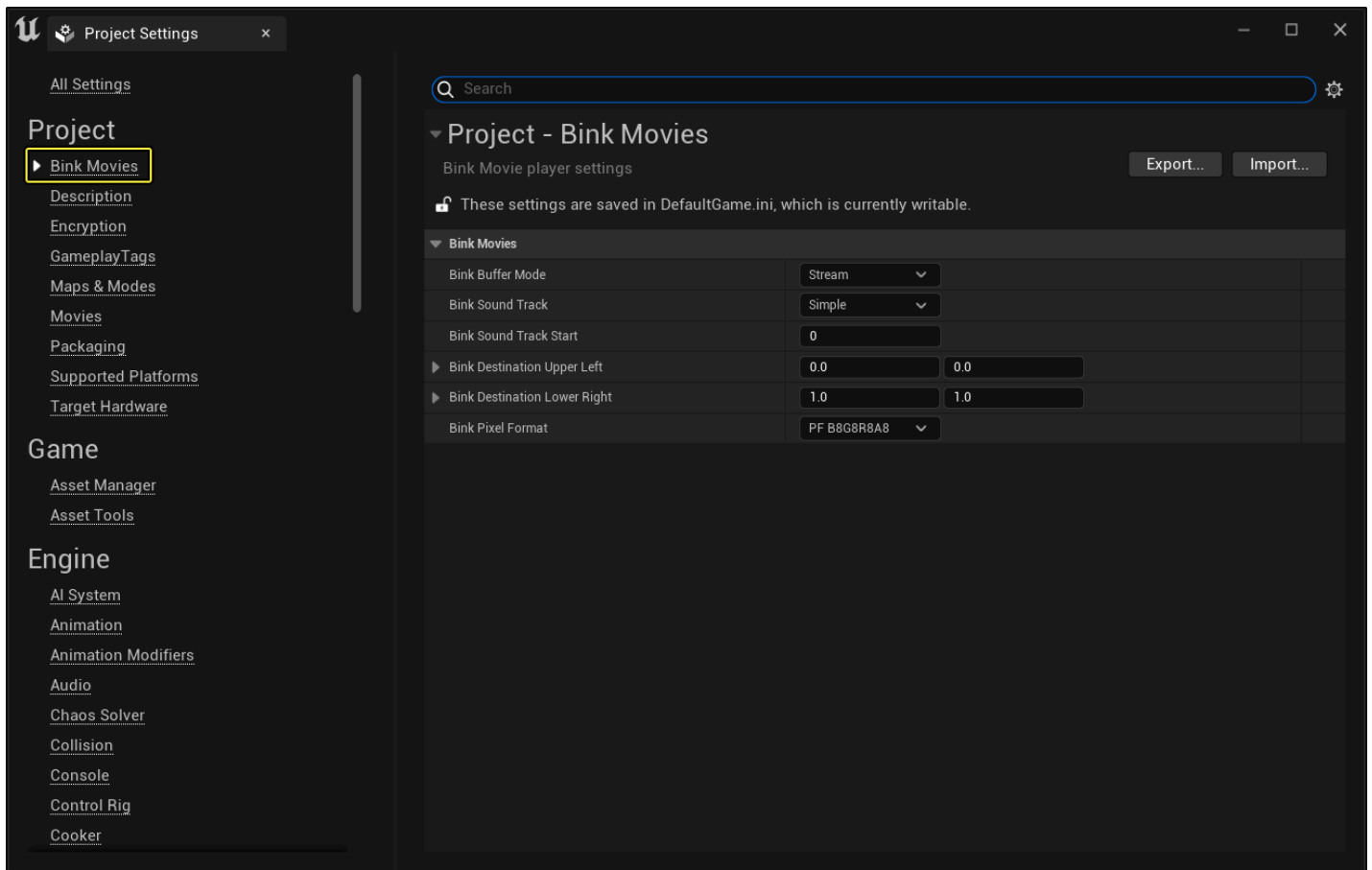
Go into **Edit > Project Settings:**



Click image to expand.

Click the **+** button next to **Startup Movies**, and then expand the list to show the movies. You can play multiple movies in sequence. Then, click the **...** button to select a file to play.

Go into **Edit > Project Settings > Bink Movies** to see Bink-specific options.



Click image to expand.

There are some Bink-specific options:

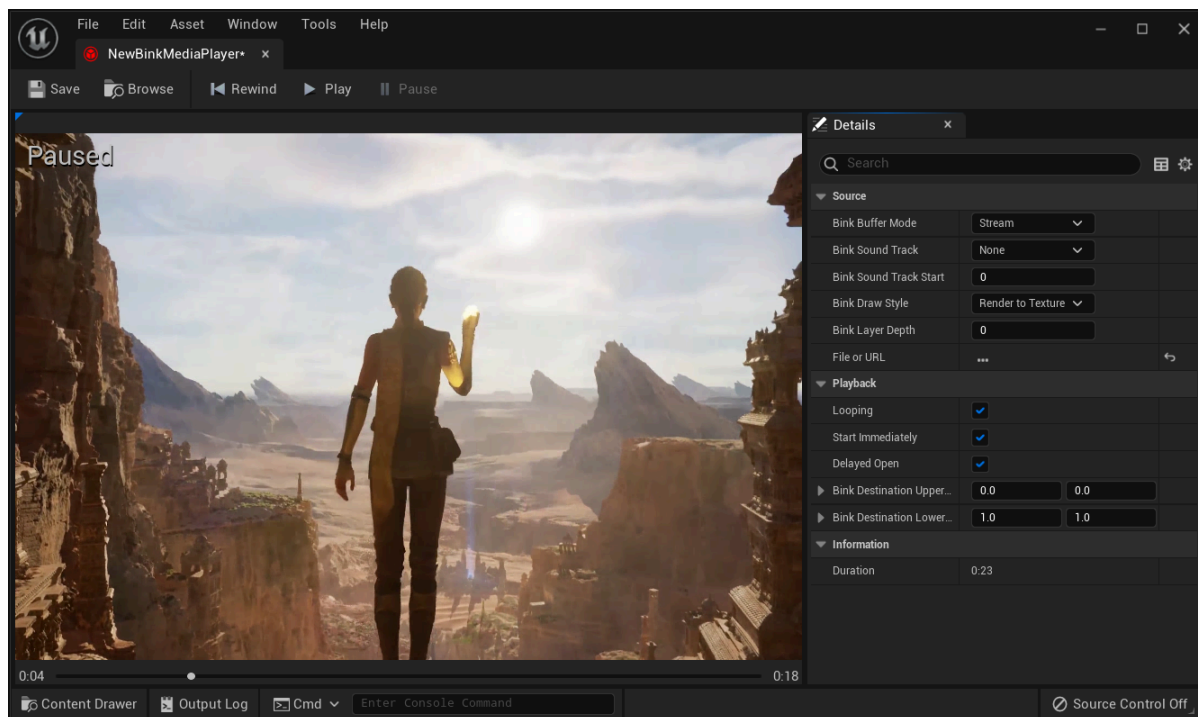
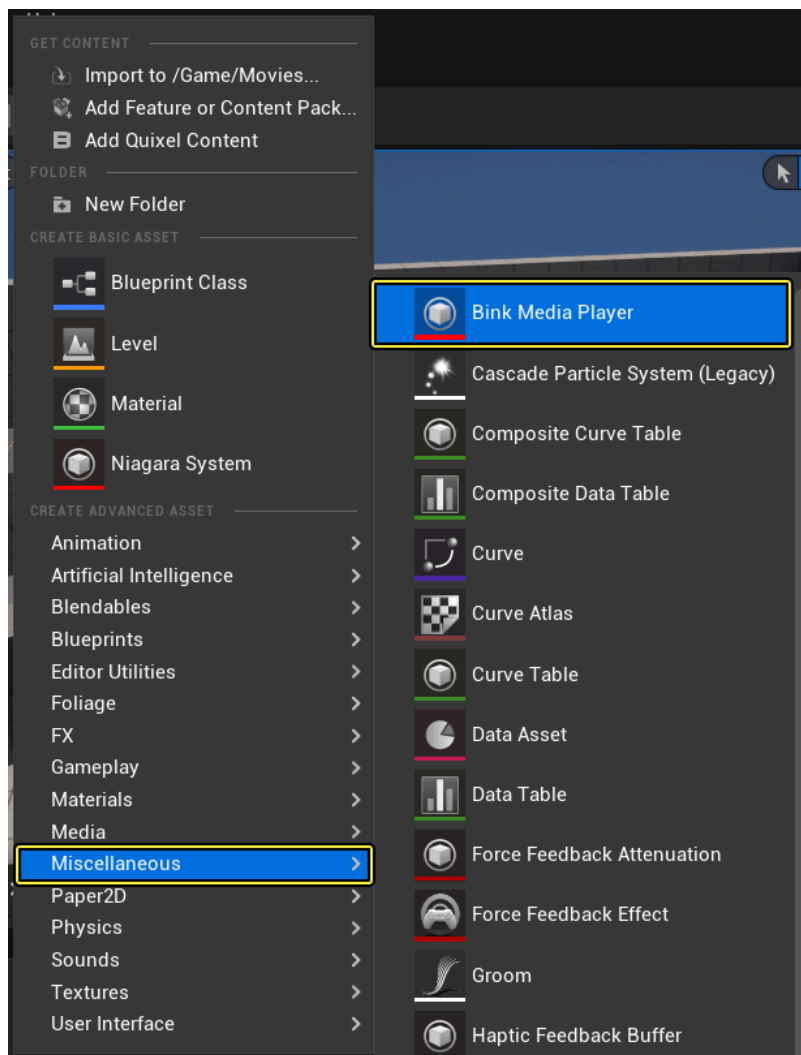
1. **Bink Buffer Mode:** This controls whether or not you want to stream a portion of the movie from disk, preload the whole movie before playing, or don't start streaming until the whole movie is resident.
2. **Bink Sound Track:** This determines how you want to play the sound. There are several options:
 - a. **SndNone:** don't play any sound in this Bink.
 - b. **Snd Simple:** Default. This option tries to figure out what you want based on the filename of the Bink file.
 - If the filename ends in `_51`, it uses the **Snd 51** option.
 - If the filename ends in `_51L`, it uses the **Snd 51 Language Override** option.
 - If the filename ends in `_71`, it uses the **Snd 71** option.
 - If the filename ends in `_71L`, it uses the **Snd 71 Language Override** option.
 Otherwise, it plays the Bink track specified by the **Sound Track Start** value (by default, track 0).
 - c. **Snd Language Override:** This option plays two tracks.
 - The audio in track 0, usually the background music / effects.

- The track specified by the **Sound Track Start** value, which is usually the language track.
- d. **Snd 51**: This option plays the six mono tracks starting at the track specified by the **Sound Track Offset** value (by default, 0) into the systems. So, if you have fully localized 5.1 tracks for four languages, you would mix in twenty-four Bink tracks, and use the **Sound Track Offset** setting to specify the right range of audio tracks to play.
 - e. **Snd 51 Language Override**: This option plays the six mono background / effect tracks starting at offset 0, and then the one mono track specified by the **Sound Track Start** value as the language track into the center channel. So, for four languages here, you would have ten audio tracks, the 5.1 background music / effects tracks and then four different center language tracks.
 - f. **Snd 71**: This option plays the eight mono tracks starting at the track specified by the **Sound Track Offset** value (by default, 0) into the systems. So, if you have fully localized 7.1 tracks for four languages, you would mix in twenty-eight Bink tracks, and use the **Sound Track Offset** setting to specify the right range of audio tracks to play.
 - g. **Snd 71 Language Override**: This option plays the eight mono background / effect tracks starting at offset 0, and then the one mono track specified by the **Sound Track Start** value as the language track into the center channel. So, for four languages here, you would have twelve audio tracks, the 7.1 background music/effects tracks and then four different center language tracks.
3. **Bink Sound Track Start**: A Bink movie can contain multiple different sound tracks. With this option you can specify which sound track you want to play.
 4. **Bink Destination Upper Left / Lower Right** This specifies the rectangle you want the movie to render to. You can use this to force a letter box, for example.

In-Game (Non-Startup) Movie Setup

You can also render movies during the game level itself. You can either render directly to a texture and then use that texture in game, or you can render directly to the screen. Bink renders after all the graphics but before the UI, so you can draw subtitles on top of the movie.

To do this, right click in the **Content Browser** and under the **Miscellaneous** section add a new **Bink Media Player**.



Click image to expand.

The Bink specific configuration options are:

1. **Bink Buffer Mode** This controls whether or not you want to stream a portion of the movie from disk, preload the whole movie before playing, or don't stop streaming until the whole movie is resident.
2. **Bink Sound Track** This determines how you want to play the sound. There are various options from simple stereo to 7.1 surround sound.
3. **Bink Sound Track Start** A Bink movie can contain multiple different sound tracks. With this option you can specify which sound track you want to play.
4. **Bink Draw Style** With this you can override the default UE5 functionality of rendering to a texture and instead render directly to the screen bypassing UE4 rendering.
5. **Bink Destination Upper Left / Lower Right** This specifies the rectangle you want the movie to render to. You can use this to force a letter box, for example.
6. **Bink Layer Depth** Allows you to render multiple videos at the same time and set the depth to control the order that they are drawn.



Bink movie files (`*.bk2`) should be placed in the (ProjectName)/Content/Movies folder, where (ProjectName) is the name of your project. This ensures that they are copied through to the final build properly in all configurations.

Right-click on the **BinkMediaPlayer** in the **Content Browser** and select **Create Media Texture** to generate a texture from the player. You can then create a default material using this texture by right-clicking the `BinkMediaPlayer_Tex` texture in the **Content Browser** and selecting **Create Material**. You can use this material and texture like you would use any other material or texture in Unreal Engine.

Other Bink for Unreal Notes

Multithreaded Decoding

The Bink Media plugin for Unreal Engine natively supports multithreaded decoding. It uses up to four threads during decompression, depending on the runtime CPU count.

By default, Bink 2 Encoder for Unreal Engine uses four video slices for best multithreaded performance.

Seeking

You can set a new playback position by calling the Bink plugin function, `BinkPluginGoto`, directly. Usually, you want to jump to a key frame, or all the intervening frames have to be decompressed along the way (the Bink plugin does this in the background for you). You control how much CPU Bink spends each frame seeking to the new frame with the `ms_per_process` parameter. Usually using 30 ms or so per frame seeks relatively quickly.

Video Depth (Sorting During Drawing)

If you are drawing lots of video overlays with the non-startup movie option, you can control the order in which they are drawn with the **Depth** option. This lets your videos stack correctly regardless of the order that Unreal Engine processes your videos.

For render targets, the **Depth** controls the order that the videos are drawn into the render target, but the render target is drawn into the scene normally by Unreal Engine.

Subtitles

Subtitles are supported via the [Subrip](#) .srt file format.

To use an srt file, include it in the same directory as the **Bink 2 Video bk2** file. Name the file using the following format: "`<name of Bink2 video file>_<language>.srt`". This file should contain the subtitles for the specified language.

For example, if you have a "**Movies/example.bk2**" file, you would then have an additional "**Movies/example_en.srt**" file alongside it containing the subtitles for the English language.

Stereo Video

Bink is fast enough to use for 3D video. To do this, use the render target path, and then draw to a screen aligned quad. For each eye, adjust the quad to fill the screen with either the top half of the quad or the bottom half of the video.

Bink Plugin API

There are a number of other controls to the Bink plugin API as well. You can call any of the functions in `binkplugin.h` when you need custom controls, such as seeking, pausing, and so on.