

## Overview

This tutorial explains how to set up the editor (or game) to capture video from the CRYENGINE which will be output as single frames and if required can also output audio (in stereo or 5.1 surround sound) in .wav format. This material can later be put together with any video editing software.

## Preparation

The process of capturing video footage requires some commands to be typed into the console, in order to start both media streams.

To speed up the process you can create configuration files that will execute all necessary commands at once. This way, you can start and stop the process quite quickly. To start the recording you need to set up a few things which will be discussed below.

## Video Settings

### Frame Size

The height and width of the captured frames in the editor is normally set to exactly the size of your rendered perspective window. You can easily resize the view size by re-scaling the perspective window or by right clicking in the top right of the perspective viewport where the size of the frame is displayed.

There have recently been some changes to the capture process and you can now capture higher than rendered images from Sandbox and Launcher.

The console variables that are now used in conjunction with Capture Frames are:

- `r_CustomResHeight=N`
- `r_CustomResWidth=M`

Where N stands for the desired frame height and M for the desired frame width in pixels.

The next few commands you will want to be aware of as well:

- `r_CustomResMaxSize=4096`
- `r_CustomResPreview=1`

The `r_customresmaxsize` command defines the maximum resolution that the engine will render the frames at.

The `r_customrespreview` command enables and disables preview of custom resolution rendering in viewport where 0 = no preview, 1 = scaled to match viewport, 2 = custom resolution clipped to viewport.

### Frames Per Second

The captured frames are all full frames, also called progressive frames.

Define the amount of frames per second you need. An NTSC standard video is approximately 30 frames per second, which is a good compromise between quality and file size.

A high quality video can have up to 60 frames per second – the difference in quality of higher values is barely noticeable, but will still take up a lot of file space. Motion will not look smooth with less than 24 fps (cinema standard).

### Fixed Time Step

To force a fixed frame rate of a certain speed use the command: `t_fixedstep N`.

Where N specifies time step, which is calculated as follows: time step = 1 second / amount of frames.

Example: 1 second / 30 frames = 0.033333333

0.016666667 would be 60 frames per second. If you want to record a standard PAL speed video (25 fps), use a value of 0.04.

## File Format Selection

The captured pictures can be in several different file formats. A good choice for average quality is the .jpeg format, whereas .tga or .bmp are better for higher quality, and HDR for pictures in high dynamic range quality.

## Capture File Format

- [Overview](#)
- [Preparation](#)
- [Video Settings](#)
  - [Frame Size](#)
  - [Frames Per Second](#)
  - [Fixed Time Step](#)
- [File Format Selection](#)
  - [Capture File Format](#)
  - [File Location](#)
  - [Starting and Ending the Recording](#)
- [Audio Settings](#)
  - [Deactivating the Sound System](#)
  - [Reactivating the Sound System](#)
- [Creating Configuration Files](#)
  - [Executing the Config Files](#)

Use the console command: ***capture\_file\_format N***

In order to select the desired format, N should be replaced with either jpg, bmp, tga or hdr.

## File Location

The recorded frames will be stored either in the default folder called "CaptureOutput" and located in the root folder, or within a custom folder, defined using the following command: ***capture\_folder N***

To choose the correct folder, N should be replaced with the name of your folder (e.g. scene12\_take1).

Be aware that when you start a recording, the captured frames will be placed in the currently defined folder, overwriting any existing files with the same name.

For each new recording you should either create a new folder or move the existing files to another folder to avoid losing any work.

## Starting and Ending the Recording

When everything is set up you can start the recording with the command: ***capture\_frames N***

Set N to 1 to start recording, and 0 to stop.

## Audio Settings

First of all, decide if you need the audio in stereo or 5.1 surround. You will need to change your audio settings in the Windows control panel. Go to "Sounds and Audio Devices" select the Volume tab, click the button "Advanced" and select your choice of output device.

## Deactivating the Sound System

After loading the desired level, you will need to deactivate the sound system before you can redirect the sound output to a file. To deactivate it use the command: ***#Sound.DeactivateAudioDevice()***

Now, the sound output will be redirected to the root folder and saved as a .wav file. The sound will not run in realtime but be linked precisely to the set time step. You won't hear anything after activating the sound again, as long as you record it.

To write the sound use the command: ***s\_OutputConfig N***

N should be set to 3 to activate the non-realtime writing of the .wav file or 0 to switch it back to the default setting (auto-detection).

## Reactivating the Sound System

To reset the sound system use the following command: ***#Sound.ActivateAudioDevice()***

Now a .wav file will be created in the root folder of the game. The file will continue recording until the writing is deactivated with the following combination of commands:

- ***#Sound.DeactivateAudioDevice()***
- ***s\_OutputConfig 0***
- ***#Sound.ActivateAudioDevice()***

Although the whole sound system is reset using the above commands, some sounds won't restart until they are correctly triggered again. This applies particularly to looped sounds.

To get the correct sounds to play, it is recommended to start the recording of video and sound first, and then enter any area that triggers your looped sounds for your recording.

## Creating Configuration Files

To capture several recordings with the same setting, it may be convenient to set up a configuration file containing the parameters required for recording in order to ensure all captured files are of the same format.

A setting config file could look like this:

```
sys_spec = 4

Fixed_time_step 0.0333333333

Capture_file_format jpg

Capture_folder myrecording

r_width 1280
r_height 800
```

The command `sys_spec = 4` sets the game graphic settings to very high to get the best look possible.

To speed up the process to start and stop the recording, it's convenient to create two configuration files, one to start and one to stop the video.

To start recording, you need a config file that looks something like this:

```
#Sound.DeactivateAudioDevice()

s_OutputConfig 3

#Sound.ActivateAudioDevice()

Capture_frames 1
```

To stop recording, you need a config file that looks something like this:

```
Capture_frames 0

#Sound.DeactivateAudioDevice()

s_OutputConfig 0

#Sound.ActivateAudioDevice()
```

## Executing the Config Files

To activate the config file, open the console and type: **Exec N**

Where N is the name of the config file.