

obs-ffgl — user guide

Run FreeFrame (FFGL) video plugins natively inside OBS Studio.

OBS cannot load FFGL plugins, and the usual way round that is a network round trip: send a source out over NDI, run the effect in another application, bring it back. This removes the trip. The plugin runs in OBS's own OpenGL context, so there is no second render, no copy back over the network, and no extra frame of latency.

What you need

- **macOS.** Windows and Linux are not built.
- **OBS Studio on its OpenGL renderer.** FFGL is an OpenGL format. If OBS is running Metal or Direct3D the module says so in the log and registers nothing, rather than adding a filter that produces black. On macOS OBS uses OpenGL 4.1, which is the same profile Resolume gives these plugins.
- **Some FFGL plugins.** Any FFGL 2.x bundle works, not only ours.

Installing

Copy `obs-ffgl.plugin` into:

```
~/Library/Application Support/obs-studio/plugins/
```

Restart OBS. You should see this in the log:

```
[obs-ffgl] loaded (plugins are scanned when first needed)
```

Using it

There are two entries, and which one you want depends on the plugin.

FFGL Effect is a filter, for plugins that transform a picture. Select any source, open **Filters**, press **+**, choose *FFGL Effect*, then pick a plugin from the list.

FFGL Source is an input, for plugins that generate a picture on their own — digital rain, screensavers, animated shapes. Press **+** in the Sources dock and choose *FFGL Source*. Only plugins that declare themselves generators appear in its list, because an effect placed there would have no input and render nothing.

A source has no picture to take its size from, so it has **Width** and **Height** fields. Leave both at 0 and it follows your canvas, which is usually what you want.

Parameters

Every parameter the plugin declares appears, in the plugin's own groups and order, with its own names — the same controls Resolume would show. Dropdowns are real dropdowns carrying the plugin's own entries.

Values are only sent to the plugin when you change them. That is deliberate: a plugin sets its own defaults when it starts, and pushing our reading of those back at it can quietly replace working values with wrong ones.

Where plugins are found

In order, first match winning:

```
$OBS_FFGL_PATH (colon-separated)
~/Library/Application Support/obs-studio/plugin_config/obs-ffgl/plugins
~/Library/Graphics/FreeFrame Plug-Ins
/Library/Graphics/FreeFrame Plug-Ins
~/Documents/Resolume Arena/Extra Effects
~/Documents/Resolume Avenue/Extra Effects
```

Resolume's own folders are included so you do not have to keep a second copy of plugins you already have. The properties dialog lists every directory it searched — hover the plugin dropdown — so a plugin that does not appear tells you where to look.

The **first** time you open the dialog, macOS may ask for permission to read your Documents folder. That is the Resolume paths above. Answer it and the scan takes about a tenth of a second thereafter. The scan deliberately does not happen when OBS starts, so that prompt never appears behind a splash screen.

Troubleshooting

No FFGL Effect in the filter list. OBS is not on its OpenGL renderer, or the plugin did not load. Check the log for `[obs-ffgl]`.

The plugin dropdown is empty. Nothing was found in the directories above. Drop a bundle into `~/Library/Graphics/FreeFrame Plug-Ins`, or set `OBS_FFGL_PATH` to where yours live, and reopen the dialog.

A plugin is listed but the picture does not change. Some plugins need a file before they can draw anything — Flipbook wants a sprite sheet, Cartridge wants a ROM. Set the file parameter. If it still does nothing, the log will carry either `FF_INSTANTIATE_GL failed` or `ProcessOpenGL failed` for that bundle, which means the plugin itself is refusing rather than being refused.

A generator is not in the FFGL Source list. It declares itself an effect rather than a source. Add it as a filter instead.

Everything went strange after adding the filter. Please report it. FFGL plugins are entitled to leave OpenGL in whatever state suits them, and this plugin puts it all back before OBS draws anything else — but a plugin that trips that is worth knowing about.

What it does not do

- Audio-reactive parameters (`FF_TYPE_BUFFER`) are not fed.
- The plugin's clock is wall time; OBS does not drive it.
- Windows and Linux are not built.

Licence

The source is MIT. A binary that links libobs is subject to OBS Studio's GPL-2.0 terms, so the plugin you install is GPL. See `ATTRIBUTIONS.md` .

obs-ffgl · guide updated 2026-08-06 · stoatworks-labs.com/software/obs-ffgl/

Latest version of this guide: stoatworks-labs.com/software/obs-ffgl/guide/ · Source and issues:

<https://github.com/stoatworks-labs/obs-ffgl>