

Envisaged Redux

Version 0.18.0

Table of Contents

About	1
Dependencies	2
Versioned Dependencies	3
Building Envisaged Redux	3
Auto Script	3
Manual	4
Getting Started	4
Installation	4
Quick Start Example	4
Examples	5
All Examples	5
API Documentation	7
Runtime Mounts	8
Live Preview	10
Templates	11
Color Groups	13
Runtime Environment Variables	14
Live Preview Environment Variables	15
Color Groups Environment Variables	16
Render Environment Variables	16
Gource Environment Variables	19
API Glossary	31
Troubleshooting	33

About

This is the official documentation for the **Envisaged Redux** project.

Envisaged Redux is a Docker container application to generate Gource visualizations and saving them as videos using FFmpeg. Currently **Envisaged Redux** supports generating Gource visualizations from Git repos, as well as an option to provide your own Gource custom log to render. More inputs may be supported in the future.



What makes **Envisaged Redux** stand out from other similar containerized applications is its portable-focused approach. **Envisaged Redux** can and will run on any platform that supports Docker, and it requires no extra hardware support to run (e.g. GPU's). This enables **Envisaged Redux** to be used in CI/CD chains and basic cloud VPS services without issue. This is achieved by leveraging the Gallium LLVMPipe Driver and Xvfb to render in software.

Head over to [Getting Started](#) for a fast track tutorial.

Official source repository for this project is on [GitLab](#) with a mirror on [GitHub](#). Docker images available on [Docker Hub](#).

Key features include:

- Git Support
- Web-based Live Preview
- Gource API (via Environment Variables)
- Render Templates
- Multi-repository Support
- Color Grouping Feature
- Save generated Gource log file
- Recursive Submodule Support

Envisaged Redux is a fork of the [Envisaged](#) project.

Dependencies

A number of core dependencies used by **Envisaged Redux** are:

- Git
- FFmpeg †
- Gource †
- libx264 †
- libx265 †
- lighttpd

- Xvfb
- Mesa (Gallium LLVMPipe)
- ImageMagick
- HLS.js

† These projects are GPL Licensed to their respective copyright owners. They are compiled from source during the `docker build` process and the source code is bundled with each image under `/gpl_sources`. Compile-time configurations are derivable from the `Dockerfile` at the root of this project's source repository.

If there is any issue regarding GPL compliance, please reach out to the maintainer of this project.

Versioned Dependencies

For stability purposes, some dependencies (including some build-time dependencies) are version controlled for stability purposes.

The current list of versioned dependencies for **Envisaged Redux** 0.18.0 are:

```
ENV VERSION_STABLE_GOURCE="0.51" \  
    VERSION_MESA="19.3.5" \  
    VERSION_LLVM="11vm9" \  
    VERSION_FFmpeg="4.2.2" \  
    VERSION_NASM="2.14.02" \  
    VERSION_YASM="1.3.0" \  
    VERSION_X264="20191217-2245" \  
    VERSION_X265="3.2.1" \  
    VERSION_HLS_JS="0.13.2"
```

Building Envisaged Redux

Prebuilt images of **Envisaged Redux** are available at [Docker Hub](#). If you wish to build and test Envisaged Redux locally (for example, to develop on), read on.

Auto Script

There is a convenience script that will work on Linux and OSX hosts located at `dev-tools/build_script.sh`. Running the script will perform 3 tasks:

1. Build a local copy of **Envisaged Redux** under the image name and tag `cartoonman/envisaged-redux:latest`.
2. Build the testing overlay image `cartoonman/test-envisaged-redux:latest`.
3. Run the test suite.

Manual

To manually build the image, run the following command from the root directory of the **Envisaged Redux** repo:

```
docker build -t cartoonman/envisaged-redux:latest ./
```

To build the test image, the **Envisaged Redux** image must have been already built. The command to build it is (from the root directory of the **Envisaged Redux** repo):

```
docker build -f tests/Dockerfile -t cartoonman/test-envisaged-redux:latest ./
```

To run the test suite, run the `tests/scripts/start.sh` script.

Alternatively, you can run the test suite with the command:

```
docker run --rm -t \  
  --name test-envisaged-redux \  
  cartoonman/test-envisaged-redux:latest
```

Getting Started

This section will fast track users to run a local instance of **Envisaged Redux** to experiment with. We will leverage one of the many example scripts designed to showcase the features of **Envisaged Redux**. The **Envisaged Redux** git repository itself will be our example repository.

Installation

Dependencies

Docker [Installation Instructions](#)

Git [Installation Instructions](#)

Quick Start Example

Fetching the Image

1. Pull the latest build of **Envisaged Redux** from Docker Hub

```
docker pull cartoonman/envisaged-redux:latest
```

2. Clone the git repo for **Envisaged Redux**

```
git clone https://gitlab.com/Cartoonman/Envisaged-Redux.git
```

Starting the Example

1. Navigate to the cloned repo directory and run the following command:

```
examples/quick_start_example.sh
```

2. At this point the container should be spawned. Open your browser and navigate to <http://localhost:8080>
3. Click on [Click here for Live Preview](#) (or navigate to <http://localhost:8080/preview.html>) to view the Live Preview feature of **Envisaged Redux**. The Live Preview shows in pseudo-realtime (with a slowdown factor applied to avoid jitter) what the visualization being rendered looks like †.
4. When the video is finished rendering, the main page will display the link to download the video from. When you are done, simply press Ctrl + C in the terminal running **Envisaged Redux** and this will close the application.

TIP

The Live Preview window is smart and can tell when the **Envisaged Redux** container is stopped, or when a new **Envisaged Redux** container is launched. A great way to experiment is to leave the Live Preview window up while playing with different runtime configurations. The Live Preview window will automatically show the live preview when available without requiring browser refreshes.

NOTE

† Certain visual compression artifacts present/absent in output file will not be present/absent in the Live Preview display. See [Live Preview](#) for more details regarding these caveats.

Examples

Included in the **Envisaged Redux** repository are various example scripts that highlight various features and capabilities of **Envisaged Redux**. Users are encouraged to use these scripts to build off of and experiment with their own runtime configurations.

All Examples

Included in the `examples/` directory are a number of scripts that showcase different runtime configuration examples. The scripts in `examples/` leverage helper functions and scripts to make it easy to build off of. They all call a common script, `baseline.sh`, and use common arguments and methods.

You can make your own scripts by leveraging the arguments and API of this script. By adding `--help` to the calls to these scripts, you will be given the available arguments.

Available Runtime Arguments for Example Scripts:

Args:

One of the following mounts is REQUIRED:

- `--vcs-source-dir [absolute/path/to/repo(s)_dir]`
The VCS repo (or directory of repos) you want Gource to render. Currently supports Git repositories only.

- `--custom-log [absolute/path/to/log_file]`
The path to the custom Gource log file to be used as the input to Gource to render the visualization.

The following are optional mount arguments:

- `--log-output-dir [path/to/dir]`
The path to the output directory where you would want Envisaged Redux to save its generated gource.log file.

- `--caption-file [absolute/path/to/caption_file]`
The path to a caption file for Gource to display captions during the video at predefined timestamps.

- `--avatars-dir [absolute/path/to/avatars_dir]`
A directory of images with filenames matching that of users in the generated or provided Gource log.

- `--logo-file [absolute/path/to/logo_image]`
A logo image file to be rendered in the video.

- `--background-image-file [absolute/path/to/image]`
An image file to be used as the background image of the Gource visualization.

- `--default-user-image-file [absolute/path/to/image]`
An image file to be used as the default image of users in the Gource visualization.

- `--font-file [absolute/path/to/font]`
A font file to be used to render the displayed text in the Gource visualization.

- `--video-output-dir [path/to/dir]`
If given, the video will be saved into this directory instead of available for download through web server.

Other args will be passed through to docker run command.
e.g. `-e RENDER_H265_CRF="0"`

NOTE

For convenience, all example scripts default to using **Envisaged Redux's** own repo as its selected repository directory. However you can override this by simply providing your own path with the `--vcs-source-dir` argument.

Most scripts also run with `LIVE_PREVIEW` enabled, so you can view the visualization in real time. In addition, all examples run with the `-p 8080:80` docker argument, so your local port `8080` will be accessible to reach the web interface of **Envisaged Redux**.

default_example

This is the bare-bones script that all other Example scripts are based on. This runs **Envisaged Redux** with no predefined environment variables. This can be considered a clean slate example that users may wish to build on top of as they experiment.

quick_start_example

If you followed the section [Getting Started](#), this is the example you ran. This showcases a general example of configurations tailored for visualizing **Envisaged Redux**'s development history.

alt_preview_example

If for any reason Live Preview does not work for you (e.g. browser not capable of [HLS](#)), this is a useful script for optimizing the fastest render time. Users can use this to quickly view the effects of their custom runtime configurations on the rendered visualization before scaling up to their target render settings.

alt_default_user_image_example

This example exercises the [Default User Image](#) feature of Gource, with a provided image replacing the default user image.

headless_example

For CD/VPS use cases, these will require a headless operation. This gives an example of how to run Envisaged Redux for headless operation mode.

By default it will save the output video file in the `examples/` directory.

NOTE

For proper headless operation, you may wish to remove the `-it` as well as the `-p 8080:80` docker args present in `common/baseline.sh`.

color_groups_example

This example exercises the [Color Groups](#) feature of Gource, where each directory branch is assigned it's own color.

border_template_example

This example utilizes the [Border Template](#) to render the visualization.

4k_example

For rendering 4k visualizations, this example sets certain optimal variables to ensure elements like text are properly sized at this scale.

API Documentation

This is the documentation detailing the API for **Envisaged Redux**. Environment Variables are

passed as inputs into **Envisaged Redux** through the `--env` or `-e` flag in `docker run`. For mount points, the examples provided give you the exact syntax needed. Through a combination of Environment Variables and Mounts, there is a wide array of configurations you can use to customize how **Envisaged Redux** renders your visualization.

Runtime Mounts

In **Envisaged Redux**, mounts act both as a data source/sink, as well as a trigger to enable certain features. This section will describe some of the mounts that **Envisaged Redux** supports, as well as an example of a proper mount.

For safety reasons, all mounts that provide data to **Envisaged Redux** should be marked as `readonly` when using the `--mount` directive in Docker. This will be reflected in the examples.

Required

One of the following mounts must be provided when running **Envisaged Redux**.

Single/Multi Repos

Mount Type	Read
Mount Example	<code>--mount type=bind,src=/path/to/vcs_source(s)_dir,dst=/visualization/resources/vcs_source,readonly</code>
Description	Mounts the path to a single VCS repo, or a directory of multiple VCS repos in subdirectories for multi-repo visualizations. Currently Envisaged Redux only supports Git repositories.
TIP	When running a multi-repo scenario, add <code>root</code> to <code>GOURCE_HIDE_ITEMS</code> to visually separate multiple repos.

Custom Log

Mount Type	Read
Mount Example	<code>--mount type=bind,src=/path/to/custom.log,dst=/visualization/resources/gource.log,readonly</code>
Description	Envisaged Redux will skip checks for a source repo and use the provided log file as the input to Gource. See Gource's Custom Log Format Page for more details on the file format, as well as examples.

Optional

These runtime mounts are not required, but will add features, capabilities, or modify the behavior of **Envisaged Redux**.

Save Generated Gource Log

Mount Type	Write
Mount Example	<code>--mount type=bind,src=/path/to/output_dir,dst=/visualization/output</code>
Description	Envisaged Redux will generate it's own <code>gource.log</code> based on all other options as usual, then save the resulting <code>gource.log</code> file into the provided mount directory. Envisaged Redux will promptly exit afterwards.
NOTE	Cannot be used when Custom Log is provided.
WARNING	When this is provided, Envisaged Redux will immediately exit upon saving the <code>gource.log</code> file into the output directory.

Captions

Mount Type	Read
Mount Example	<code>--mount type=bind,src=/path/to/captions.txt,dst=/visualization/resources/captions.txt,readonly</code>
Description	Gource will try given <code>captions.txt</code> file to render captions on video. See gource docs for supported caption format.

Avatars

Mount Type	Read
Mount Example	<code>--mount type=bind,src=/path/to/avatars_dir,dst=/visualization/resources/avatars,readonly</code>
Description	Gource will try given avatars directory to render user avatars on video. See gource docs for naming rules and supported image types.

Logo

Mount Type	Read
Mount Example	<code>--mount type=bind,src=/path/to/image.png,dst=/visualization/resources/logo.image,readonly</code>
Description	Gource will try given logo image file to render the logo in the lower right hand corner of the video.

Background Image

Mount Type	Read
Mount Example	<code>--mount type=bind,src=/path/to/image.png,dst=/visualization/resources/background.image,readonly</code>

Description Gource will try given background image file to use as the background of the visualization scene.

Default User Image

Mount Type Read

Mount Example `--mount type=bind,src=/path/to/image.png,dst=/visualization/resources/default_user.image,readonly`

Description Gource will try given image file to use as the default user image for all users in the visualization scene.

TIP

Add `GOURCE_COLOR_IMAGES` to have Gource apply a random color fill to each user with the given default image.

Font File

Mount Type Read

Mount Example `--mount type=bind,src=/path/to/font.ttf,dst=/visualization/resources/font,readonly`

Description Gource will use the provided font file as the text font used in the visualization. Compatible with most font types supported by FreeType, including `.ttf` and `.odf` formats.

Local Video Output

Mount Type Write

Mount Example `--mount type=bind,src=/path/to/output_dir,dst=/visualization/video`

Description If this mount is made, **Envisaged Redux** will save the rendered video in the mounted directory and immediately exit once rendering completes.

TIP

This is a necessary mount for any CD/VPS based headless runs.

Live Preview

Requirements

In order to use Live Preview, you must run the **Envisaged Redux** docker container with port 80 accessible or mapped to the host.

Live preview requires H.264 codec support and JavaScript enabled in your browser. A rough check can be done by visiting html5test.com to see if your browser supports H.264.

It is confirmed to work on the latest versions of Firefox, Chromium, and Edge, with likely support on Chrome, Safari and Opera. Since this works through the browser, it is inherently platform agnostic, and will work on any host operating system.

NOTE

If you are unable to use Live Preview, the `preview.sh` script is the best alternative to quickly check the effects of your configs before rendering a longer run.

How it works

Live preview works concurrently with the normal video rendering process, so at the end of the render you will still have the original video available to save.

The Live Preview mechanism splits the FFmpeg video rendering stream such that the user can view the output live as FFmpeg rendered it. This lets users quickly preview how the visualization looks before settling on a configuration list.

Caveats

Excluding `RENDER_H265_PRESET`, `RENDER_H265_CRF`, and `PREVIEW_SLOWDOWN_FACTOR` > 1, all API variables will have their effects represented in Live Preview exactly as what is rendered to `output.mp4`.

For the CRF and PRESET variables, the Live Preview mechanism uses a fixed CRF of 1 and does not use the PRESET variable, instead being subjected to a fixed max bitrate ceiling for performance. Because of this, visual artifacts (e.g. compression artifacts such as blockiness and fuzziness) seen in the live preview may not manifest in `output.mp4` and vice versa.

For `PREVIEW_SLOWDOWN_FACTOR`, any value that is greater than 1 will have the Live Preview render at a slower rate, but any speed adjustments made with this variable will not affect `output.mp4`. At a value of 1, the Live Preview frame speed will match `output.mp4`.

TIP

If you want to be absolutely sure `output.mp4` will look like what you expect from a compression artifact point of view, run a separate render with `GOURCE_STOP_AT_TIME` set to a low number of seconds so the render process ends early. An example of this usage can be seen in `examples/preview.sh`.

Templates

Templates are predefined structural configurations within **Envisaged Redux** that change the way Gource and FFmpeg work together.

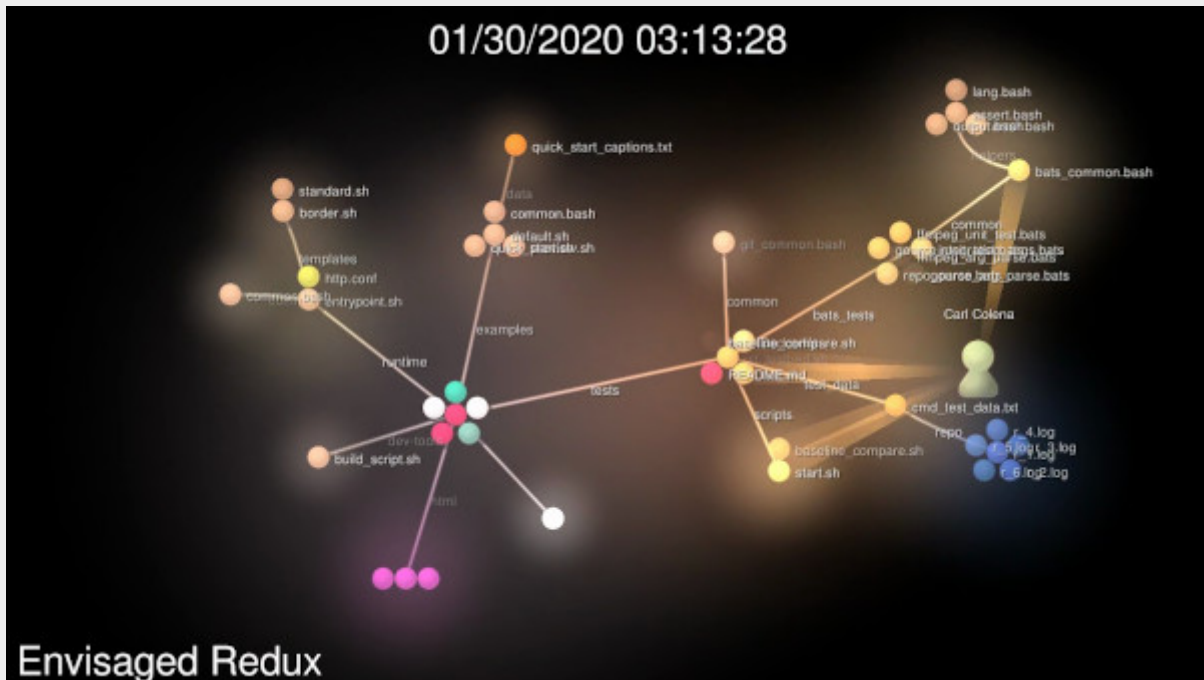
By selecting a template, **Envisaged Redux** will use it to render the output video. Certain templates may restrict, add, or modify the behavior of other configs.

Templates are configured through the `RUNTIME_TEMPLATE` Environment Variable.

Standard

This is the default template. The standard template is the simplest template. It does not employ any overlays or filters, and renders the Gource environment in full.

Standard Template Example



Border

The border template adds a border overlay with a dedicated left panel for the Gource file extension key. Due to the overlay, the user can individually set the Title and Date font sizes and colors.

The border template overrides the following general config variables:

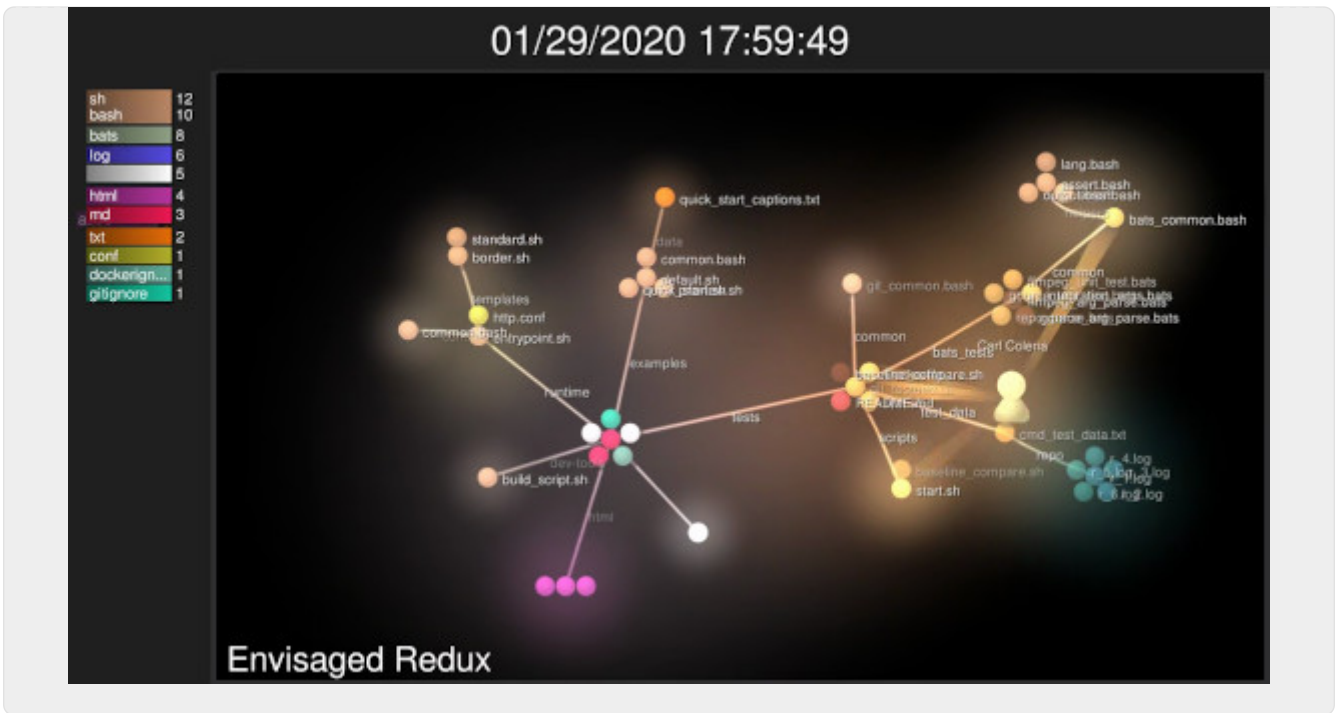
- [GOURCE_FONT_SIZE](#)
- [GOURCE_FONT_COLOR](#)
- [GOURCE_HIDE_ITEMS](#)
- [GOURCE_SHOW_KEY](#)

NOTE

See the **Overridden By** section in the documentation of each variable for details about the override.

In addition to overridden variables, there are additional specialized configuration variables for the Border template. See [Gource Variables for Templates: Border](#) for their documentation.

Border Template Example



Color Groups

NOTE This feature does not apply to [Custom Log](#) inputs.

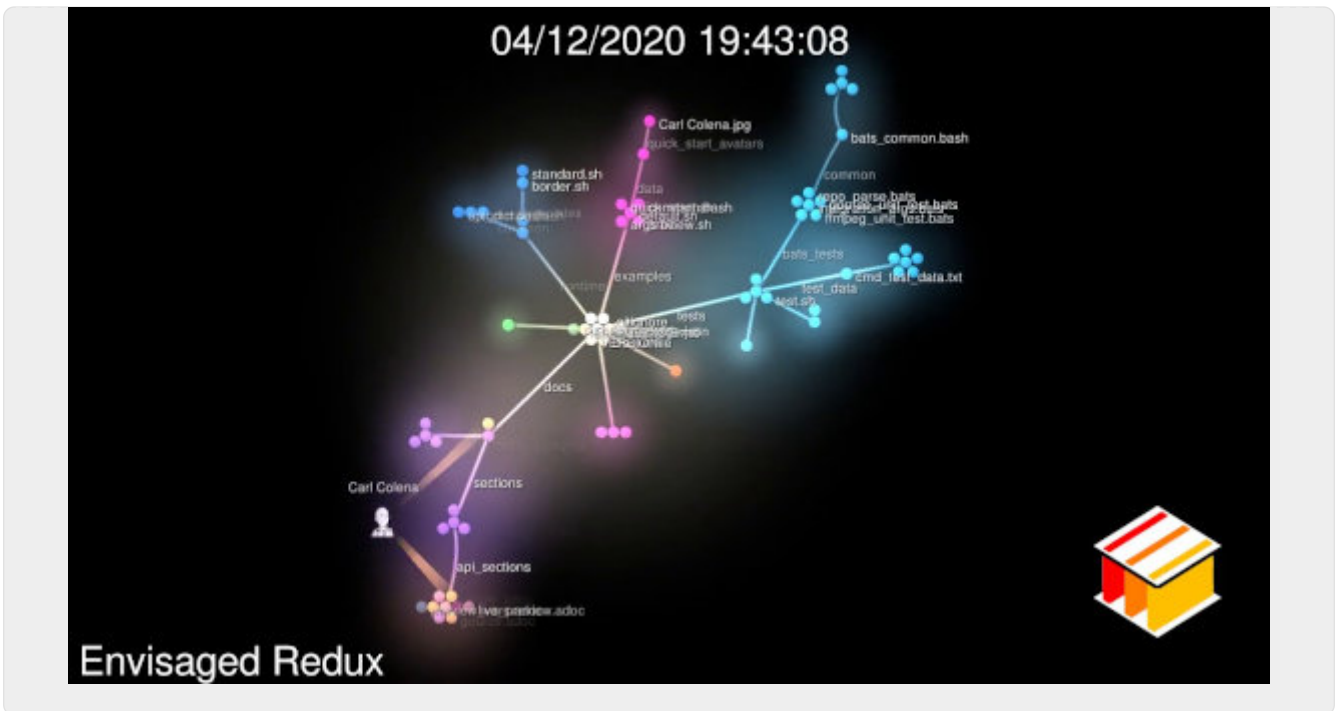
The Color Groups feature of **Envisaged Redux** colors each branch of the Gource tree to a pseudo-random color. Each branch will have its own color, and all sub branches and nodes will share the same color. For single-repo cases, each branch represents a parent directory in the repository. For multi-repo cases, each branch represents a repository.

TIP This feature is useful when you wish to highlight multiple groups working on various components of a project or system.

By default, if Color Groups is not enabled, each node in the Gource tree is colored based on the file type. Color Groups is enabled by setting the boolean variable `RUNTIME_COLOR_GROUPS` to `1`.

Additional variables to configure Color Groups can be found at the [Color Groups Environment Variables](#) section.

Color Groups Example



Modes

There are two modes of operation for Color Groups:

Random

By default this is the default mode of operation. This will generate close-to-true random colors to the coloring mechanism.

Seeded Random

For some cases, we may want to have the same color assignments for our branches. We can achieve this by providing a seed value to `COLOR_GROUPS_SEED`. This will mean that if you run the same visualization job with the same `COLOR_GROUPS_SEED` provided, you should get the same colors mapped to the same branches every time. This can provide a replicable way to have color groups in your visualization.

Runtime Environment Variables

These are general settings that configure and enable certain features of **Envisaged Redux**.

RUNTIME_TEMPLATE

Type	String
Example Value	border
Default	standard

Description	Specify a template to use. The implemented templates are: <ul style="list-style-type: none"> • standard • border
--------------------	--

RUNTIME_RECURSE_SUBMODULES

Type	Boolean
Example Value	1
Description	Flag to enable recursing through the repo(s) submodules.

RUNTIME_LIVE_PREVIEW

Type	Boolean
Example Value	1
Description	Flag to enable Live Gource preview through the web interface. See Live Preview for details.

RUNTIME_COLOR_GROUPS

Type	Boolean
Example Value	1
Description	Flag to enable Color Groups feature. See Color Groups for details.
	NOTE Does not apply when using Custom Log input.

RUNTIME_PRINT_VARS

Type	Boolean
Example Value	1
Description	Flag to have Envisaged Redux print the provided input Environment Variables. Useful for logging the runtime configuration on headless platforms.

Live Preview Environment Variables

These are general settings that configure the [Live Preview](#) feature in **Envisaged Redux**.

PREVIEW_SLOWDOWN_FACTOR

Type	Integer
Example Value	2
Description	Slowdown Factor is used to slow the preview stream to reduce buffer hangs from slow renders. 1 means no slowdown. Supported values are integers ≥ 1 . This setting will not affect the resultant <code>output.mp4</code> .

Color Groups Environment Variables

These are general settings that configure the [Color Groups](#) feature in **Envisaged Redux**.

COLOR_GROUPS_CENTER_COLOR

Type	Hexadecimal
Example Value	FFFFFF
Description	Override the default color (FFFFFF) for files at the root (center) of the visualization.

COLOR_GROUPS_SEED

Type	Integer
Example Value	577
Description	Seed value to provide as the input to the random color generator. The value MUST be a positive integer between 0 and 32768. If unset, the Color Groups will generate true random colors.

Render Environment Variables

The rendering process can be configured using a number of Render Environment Variable. These primarily adjust and effect the FFmpeg backend.

In addition to general Render settings, there are Codec-specific options available to configure:

- [H.264 Options](#)
- [H.265 Options](#)

RENDER_CODEC

Type	String
Example Value	h265
Default	h264
Description	Output video codec. Supported options are h264 and h265.

RENDER_VIDEO_RESOLUTION

Type	String
Example Value	2160p
Default	1080p
Description	Output video resolution. Supported options are 480p, 720p, 1080p, 1440p, and 2160p.

RENDER_FPS

Type	String
Example Value	60
Default	30
Description	Output video Frames per Second. Supported framerates are 25, 30, or 60 only.

RENDER_PROFILE

Type	String
Example Value	baseline
Description	The maximum compatibility profile to set for the chosen codec. See ffmpeg docs for details.

RENDER_LEVEL

Type	Float
Example Value	3.0
Description	The target level for the chosen codec. Note that there are fixed, specific supported levels, and you must specify a supported level. See ffmpeg docs for details.

RENDER_INVERT_COLORS

Type	Boolean
Example Value	1
Description	Applies a color inversion filter on the video output.

RENDER_VERBOSE

Type	Boolean
Example Value	1
Description	Enables verbose output from FFmpeg render step.

RENDER_NO_PROGRESS

Type	Boolean
Example Value	1
Description	When set to true, this disables the live progress bar during FFmpeg render step. Should be used on headless (non-interactive) renders to avoid output log spam.

H.264 Options

These are environment variables to configure the H.264 (x264) codec renderer.

RENDER_H264_PRESET

Type	String
Example Value	medium
Description	H.264 encoding preset. Refer to the FFmpeg H264 Docs for supported options.

RENDER_H264_CRF

Type	Integer
Example Value	21
Description	The Constant Rate Factor (CRF). Refer to FFmpeg H264 Docs for supported values.

H.265 Options

These are environment variables to configure the H.265 (x265) codec renderer.

RENDER_H265_PRESET

Type	String
Example Value	medium
Description	H.265 encoding preset. Refer to the FFmpeg H265 Docs for supported options.

RENDER_H265_CRF

Type	Integer
Example Value	21
Description	The Constant Rate Factor (CRF). Refer to FFmpeg H265 Docs for supported values.

Source Environment Variables

These variables give user access to Gource's runtime settings.

The currently supported Gource build is **v0.51**.

These are the subgroups of Gource Variables:

- [General API](#)
- [Caption API](#)
- [Templated API: Border](#)

General API

These are general variables that apply to all templates. There are certain templates however that will override some of these values. If applicable, a list of these templates will be noted in the [Overridden By](#) field.

GOURCE_TITLE

Assignment	<code>--title</code>
Type	String
Example Value	"Software Development"
Description	Title displayed at the lower left corner of the video

GOURCE_CAMERA_MODE

Assignment	<code>--camera-mode</code>
-------------------	----------------------------

Type	String
Example Value	overview
Description	Camera mode (Valid options are 'overview' or 'track').

GOURCE_START_DATE

Assignment	--start-date
Type	String
Example Values	<ul style="list-style-type: none"> "2016-01-30" "2004-09-22 04:15:00" "2012-11-15 13:00:00 +8"
Description	Start after given date (and optional time) (see gource docs for formats)

GOURCE_STOP_DATE

Assignment	--stop-date
Type	String
Example Values	<ul style="list-style-type: none"> "2016-01-30" "2004-09-22 04:15:00" "2012-11-15 13:00:00 +8"
Description	Stop after given date (and optional time) (see gource docs for formats)

GOURCE_START_POSITION

Assignment	--start-position
Type	Float OR String
Example Values	<ul style="list-style-type: none"> 0.25 random
Description	Begin at some position in the log (between 0.0 and 1.0 or 'random').

GOURCE_STOP_POSITION

Assignment	--stop-position
Type	Float
Example Value	0.75
Description	Stop at some position in the log (between 0.0 and 1.0)

GOURCE_STOP_AT_TIME

Assignment	<code>--stop-at-time</code>
Type	Integer
Example Value	20
Description	Stop after a specified number of seconds.

GOURCE_SECONDS_PER_DAY

Assignment	<code>--seconds-per-day</code>
Type	Float
Example Value	0.1
Description	Speed of simulation in seconds per day.

GOURCE_AUTO_SKIP_SECONDS

Assignment	<code>--auto-skip-seconds</code>
Type	Float
Example Value	3.0
Description	Skip to next entry if nothing happens for a number of seconds.

GOURCE_TIME_SCALE

Assignment	<code>--time-scale</code>
Type	Float
Example Value	1.0
Description	Change simulation time scale.

GOURCE_USER_SCALE

Assignment	<code>--user-scale</code>
Type	Float
Example Value	1.0
Description	Change scale of user avatars.

GOURCE_MAX_USER_SPEED

Assignment	<code>--max-user-speed</code>
Type	Integer
Example Value	<code>500</code>
Description	Max speed users can travel per second.

GOURCE_HIDE_ITEMS

Assignment	<code>--hide</code>
Type	List
Example Values	<ul style="list-style-type: none"><code>date</code><code>date,users,filenames</code>
Overridden By	<ul style="list-style-type: none">border<ul style="list-style-type: none">Only overrides <code>date</code> (Forces <code>date</code> to be set). User can set all other options.
Description	Hide one or more display elements from the list below: <div data-bbox="336 976 1455 1464" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"><pre>bloom - bloom effect date - current date dirnames - names of directories files - file icons filenames - names of files mouse - mouse cursor progress - progress bar widget root - root directory of tree tree - animated tree structure users - user avatars usernames - names of users</pre></div> <p>For Envisaged Redux, mouse and progress have no effect.</p>

GOURCE_FILE_IDLE_TIME

Assignment	<code>--file-idle-time</code>
Type	Float
Example Value	<code>0.0</code>
Description	Time in seconds files remain idle before they are removed or 0 for no limit.

GOURCE_MAX_FILES

Assignment	<code>--max-files</code>
-------------------	--------------------------

Type	Integer
Example Value	0
Description	Set the maximum number of files or 0 for no limit. Excess files will be discarded.

GOURCE_MAX_FILE_LAG

Assignment	<code>--max-file-lag</code>
Type	Float
Example Value	5.0
Description	Max time files of a commit can take to appear. Use -1 for no limit.

GOURCE_FILENAME_TIME

Assignment	<code>--filename-time</code>
Type	Integer
Example Value	2
Description	Duration to keep filenames on screen (>= 2.0).

GOURCE_FONT_SIZE

Assignment	<code>--font-size</code>
Type	Integer
Example Value	48
Overridden By	<ul style="list-style-type: none"> • border <ul style="list-style-type: none"> ◦ Superseded by <code>GOURCE_BORDER_TITLE_SIZE</code> and <code>GOURCE_BORDER_DATE_SIZE</code>.
Description	Font size for title and date.

GOURCE_FONT_COLOR

Assignment	<code>--font-colour</code>
Type	Hexadecimal
Example Value	FFFFFF
Overridden By	<ul style="list-style-type: none"> • border <ul style="list-style-type: none"> ◦ Superseded by <code>GOURCE_BORDER_TITLE_COLOR</code> and <code>GOURCE_BORDER_DATE_COLOR</code>.

Description	Font color for title and date in hex.
--------------------	---------------------------------------

GOURCE_BACKGROUND_COLOR

Assignment	<code>--background-colour</code>
-------------------	----------------------------------

Type	Hexadecimal
-------------	-------------

Example Value	<code>000000</code>
----------------------	---------------------

Description	Background color in hex.
--------------------	--------------------------

GOURCE_DATE_FORMAT

Assignment	<code>--date-format</code>
-------------------	----------------------------

Type	String
-------------	--------

Example Value	<code>"%m/%d/%Y %H:%M:%S"</code>
----------------------	----------------------------------

Description	Date Format (based on strftime format)
--------------------	--

GOURCE_DIR_NAME_DEPTH

Assignment	<code>--dir-name-depth</code>
-------------------	-------------------------------

Type	Integer
-------------	---------

Example Value	<code>3</code>
----------------------	----------------

Description	Draw names of directories down to a specific depth in the tree.
--------------------	---

GOURCE_BLOOM_MULTIPLIER

Assignment	<code>--bloom-multiplier</code>
-------------------	---------------------------------

Type	Float
-------------	-------

Example Value	<code>1.2</code>
----------------------	------------------

Description	Adjust the amount of bloom. (≥ 0.0)
--------------------	--

GOURCE_BLOOM_INTENSITY

Assignment	<code>--bloom-intensity</code>
-------------------	--------------------------------

Type	Float
-------------	-------

Example Value	<code>0.75</code>
----------------------	-------------------

Description	Adjust the intensity of the bloom. (≥ 0.0)
--------------------	---

GOURCE_PADDING

Assignment	<code>--padding</code>
Type	Float
Example Value	1.1
Description	Camera view padding (between 0.0-2.0 exclusive)

GOURCE_HIGHLIGHT_USERS

Assignment	<code>--highlight-users</code>
Type	Boolean
Example Value	1
Description	Keeps all user's names visible.

GOURCE_MULTI_SAMPLING

Assignment	<code>--multi-sampling</code>
Type	Boolean
Example Value	1
Description	Enables anti-aliasing multi-sampling for smoother edges

GOURCE_SHOW_KEY

Assignment	<code>--key</code>
Type	Boolean
Example Value	1
Overridden By	<ul style="list-style-type: none">• border<ul style="list-style-type: none">◦ Always enabled.
Description	Enables the file extension key legend

GOURCE_REALTIME

Assignment	<code>--realtime</code>
Type	Boolean
Example Value	1
Description	Runs the visualization at realtime playback speed.

GOURCE_ELASTICITY

Assignment	<code>--elasticity</code>
Type	Float
Example Value	1.4
Description	Sets the elasticity of nodes.

GOURCE_FOLLOW_USER

Assignment	<code>--follow-user</code>
Type	String
Example Value	<ul style="list-style-type: none">• John Doe• user45623
Description	Set the camera such that it follows the given user.

GOURCE_HIGHLIGHT_DIRS

Assignment	<code>--highlight-dirs</code>
Type	Boolean
Example Value	1
Description	Keeps the names of all directories visible for the duration of the visualization.

GOURCE_HIGHLIGHT_COLOR

Assignment	<code>--highlight-colour</code>
Type	Hexadecimal
Example Value	FFFFFF
Description	Sets the color of user names in hex.

GOURCE_SELECTION_COLOR

Assignment	<code>--selection-colour</code>
Type	Hexadecimal
Example Value	FFFFFF
Description	Sets the color of selected user names (names selected by GOURCE_FOLLOW_USER for example).

GOURCE_FILENAME_COLOR

Assignment	<code>--filename-colour</code>
Type	Hexadecimal
Example Value	<code>FFFFFF</code>
Description	Sets the color of filenames.

GOURCE_DIR_COLOR

Assignment	<code>--dir-colour</code>
Type	Hexadecimal
Example Value	<code>FFFFFF</code>
Description	Sets the color of directory names.

GOURCE_FILE_EXTENSIONS

Assignment	<code>--file-extensions</code>
Type	Boolean
Example Value	<code>1</code>
Description	Show filename extensions only.

GOURCE_USER_FRICTION

Assignment	<code>--user-friction</code>
Type	Float
Example Value	<code>1.0</code>
Description	Sets the time (in seconds) it takes for users to come to a stop.

GOURCE_DISABLE_AUTO_ROTATE

Assignment	<code>--disable-auto-rotate</code>
Type	Boolean
Example Value	<code>1</code>
Description	Disables automatic camera rotation. Camera will not rotate.

GOURCE_COLOR_IMAGES

Assignment	<code>--colour-images</code>
Type	Boolean
Example Value	<code>1</code>
Description	Colorizes the provided user images used in the Default User Image Mount .

GOURCE_NO_TIME_TRAVEL

Assignment	<code>--no-time-travel</code>
Type	Boolean
Example Value	<code>1</code>
Description	Use the time of the last commit if the time of a commit is in the past.

GOURCE_DIR_NAME_POSITION

Assignment	<code>--dir-name-position</code>
Type	Float
Example Value	<code>0.5</code>
Description	Position along edge of the directory name (between 0.1 and 1.0, default is 0.5).

GOURCE_FILE_EXTENSION_FALLBACK

Assignment	<code>--file-extension-fallback</code>
Type	Boolean
Example Value	<code>1</code>
Description	Use filename as extension if the extension is missing or empty. This applies for the file extension key legend, enabled by GOURCE_SHOW_KEY .

GOURCE_FONT_SCALE

Assignment	<code>--font-scale</code>
Type	Float
Example Value	<code>1.0</code>
Description	Scales the sizes of all fonts. 1.0 is default, < 1.0 is smaller, and > 1.0 is larger.

GOURCE_FILE_FONT_SIZE

Assignment	<code>--file-font-size</code>
Type	Integer
Example Value	14
Description	Font size of filenames.

GOURCE_DIR_FONT_SIZE

Assignment	<code>--dir-font-size</code>
Type	Integer
Example Value	14
Description	Font size of directory names.

GOURCE_USER_FONT_SIZE

Assignment	<code>--user-font-size</code>
Type	Integer
Example Value	14
Description	Font size of user names.

Caption API

These are specific Gource Variables that will only be used if a caption file was provided to **Envisaged Redux** at runtime. These settings allow the user to configure the effects of the captions displayed.

GOURCE_CAPTION_SIZE

Assignment	<code>--caption-size</code>
Type	Integer
Example Value	48
Description	Caption font size.

GOURCE_CAPTION_COLOR

Assignment	<code>--caption-colour</code>
-------------------	-------------------------------

Type	Hexadecimal
Example Value	FFFFFF
Description	Caption color in hex.

GOURCE_CAPTION_DURATION

Assignment	--caption-duration
Type	Float
Example Value	5.0
Description	The duration in seconds each caption will appear for.

GOURCE_CAPTION_OFFSET

Assignment	--caption-offset
Type	Integer
Example Value	3
Description	Caption horizontal offset. 0 centers the captions.

Templated API: Border

For the Border template, these are specialized Environment Variables for configuring Gource. These will only have an effect when used with the Border template, and will have no effect otherwise.

GOURCE_BORDER_TITLE_SIZE

Assignment	--font-size
Type	Integer
Example Value	48
Description	Font size for title.

GOURCE_BORDER_DATE_SIZE

Assignment	--font-size (overlay)
Type	Integer
Example Value	60

Description	Font size for date.
--------------------	---------------------

GOURCE_BORDER_TITLE_COLOR

Assignment	<code>--font-colour</code>
-------------------	----------------------------

Type	Hexadecimal
-------------	-------------

Example Value	FFFFFF
----------------------	--------

Description	Font color for title.
--------------------	-----------------------

GOURCE_BORDER_DATE_COLOR

Assignment	<code>--font-colour</code> (<i>overlay</i>)
-------------------	---

Type	Hexadecimal
-------------	-------------

Example Value	FFFFFF
----------------------	--------

Description	Font color for date.
--------------------	----------------------

API Glossary

The following is a detailed description of the labels used in the API documentation.

Assignment

This denotes the equivalent or assigned Gource runtime argument for the given Environment Variable.

For example, the Environment Variable `GOURCE_TITLE` has the assignment of `--title`. This means that when the user provides the input of `GOURCE_TITLE="My Project"`, then the argument that is forwarded to Gource is `--title "My Project"`.

This serves as a easy reference for further exploration if a user is curious what the details of a given configuration are. All Gource assignments can be looked up on the official [Gource docs](#)

Type

This denotes the expected datatype of the variable. By default all Environment Variables exist in an unset state save a few exceptions. If a Gource variable is unset, the Gource executable will use its own internal default handler. See [Gource's source code](#) for more details on the default assignments and handling of configuration variables.

In general, configuration variables will either be set to a default value, or remain unset in Gource.

The datatypes are:

Boolean

If set, they must be assigned the value `1` to be considered *true*.

If assigned any value other than `1`, it will be considered *false*.

By default, the state of being unset is equivalent to *false*.

Example boolean input: `GOURCE_SHOW_KEY="1"`.

String

String or string-based equivalent. May include whitespace.

Example string input: `GOURCE_TITLE="My Project Title"`

List

Identical to String, except may not include whitespace. Typically delimited with commas.

Example list input: `GOURCE_HIDE_ITEMS="mouse,date,filenames"`

Integer

Integer number. The range bounds if known will be described.

Example integer input: `GOURCE_DIR_NAME_DEPTH="3"`

Float

Floating point number. The range bounds if known will be described.

Example float input: `GOURCE_BLOOM_INTENSITY="0.75"`

Hexadecimal

6 digit Hexadecimal represented as 3 groups of 256 bit values.

For example, `030201` = $\langle 3, 2, 1 \rangle$. `AABBCC` = $\langle 170, 187, 204 \rangle$.

This is often used to represent 256 level RGB color space (0-255).

Example hexadecimal input: `GOURCE_FONT_COLOR="FFCC45"`

Mount Type

This specifies the type of mount the user should be expected to assign.

The mount types are:

Read

Read only mounts should be given then `readonly` specifier. This will guarantee that files that are passed to **Envisaged Redux** cannot be modified or overwritten by **Envisaged Redux**.

While **Envisaged Redux** has no reason to write anything to the repositories, it is best to be safe and restrict write access to the host mount.

Write

These are mounts that are expected to be written to. Generally it is a good idea to mount an empty directory to **Envisaged Redux** so it is clear to the user what files were written as a result of **Envisaged Redux**.

Example Value(s)

One or more example values that are considered valid inputs for the given Environment Variable.

Mount Example

An example of the syntax to have docker mount a directory to a specified location within **Envisaged Redux**. The highlighted portion is the section that the user must replace with their own file/folder paths, leaving the rest of the `--mount` flag the same.

It is important that the mount points within the container match the given example. **Envisaged Redux** will perform checks for these directory or file paths and will only enable functionality if the mount points are made at the right paths.

Default

If this is specified, the Environment Variable will use this assigned default value when not explicitly set at runtime.

For example, `RENDER_FPS` has a default value of `30`. This means that if **Envisaged Redux** is launched without the `RENDER_FPS` variable set, it will automatically assign `30` as the value of `RENDER_FPS`.

If `RENDER_FPS=60` is set at runtime, the `RENDER_FPS` variable will be `60` instead of the default `30`.

Environment Variables which do not have a default remain as an unset variable when not declared at runtime.

Overridden By

A list of one or more templates that fully or partially overrides the specified Environment Variable when used.

Description

A general description of what the variable configures.

Troubleshooting

Logo Error

If you receive this error:

```
convert: no decode delegate for this image format `IMAGE' @
error/constitute.c/ReadImage/556.
convert: no images defined `/visualization/resources/logo_txfrmed.image' @
```

```
error/convert.c/ConvertImageCommand/3273.
```

This means the image file passed into **Envisaged Redux** is not readable by the image converter onboard (ImageMagick). Try using another format or image.

Video can't be played because the file is corrupt

If you receive this error or something similar during a live preview playback:

```
Video can't be played because the file is corrupt
```

This is a known issue. Refreshing the page brings you right back to the latest live preview, and is the current workaround for this issue.