

Rendering in manual strips

An intermediate/advanced Apophysis tutorial

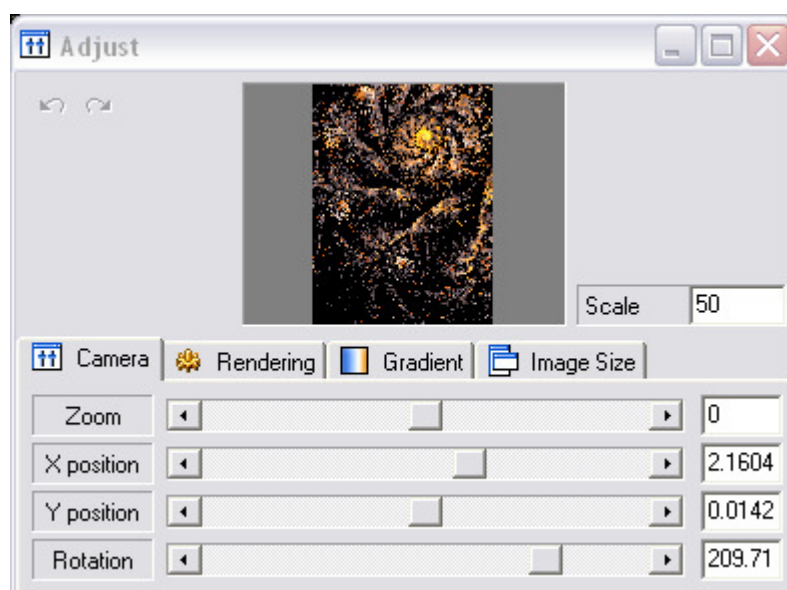
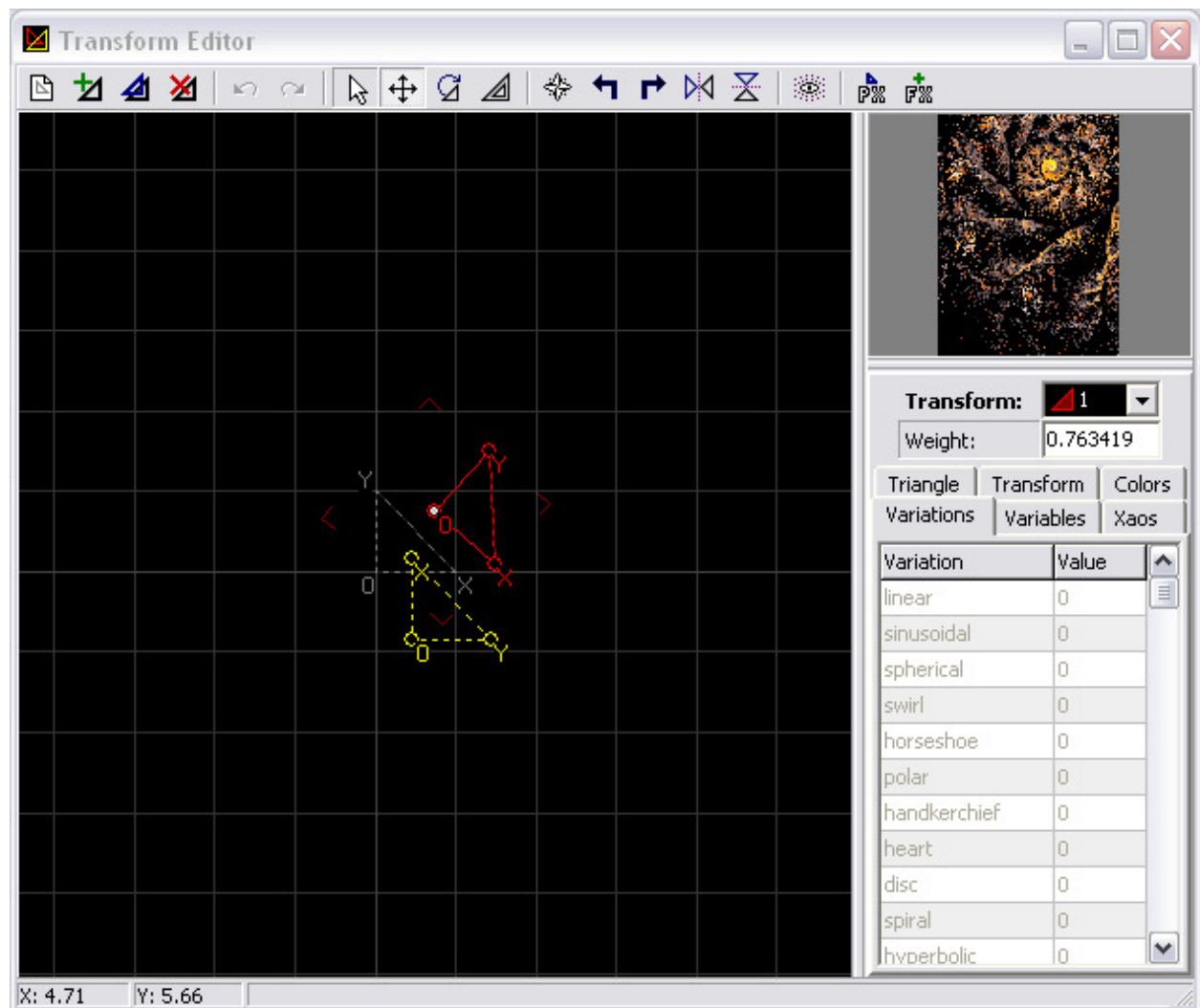
Sometimes, you'd really like to render a flame large, maybe for a special print, perhaps to fulfil specific requirements for a contest. If you're working with an older PC and/or OS, chances are the setup can't manage the render in a single strip, so you think to use Apo's memory limiter and render in strips. Sometimes this works, others not, and it's a huge CPU time investment gone if it fails at the final fence.

Due to the iterative process of flame fractals, each strip renders the entire fractal, discarding values outside the bounds of the strip. This means that rendering in 4 strips takes 4 times as long! Well not any more: I offer a methodology plus tools to bring large-scale rendering within the grasp of most artists.

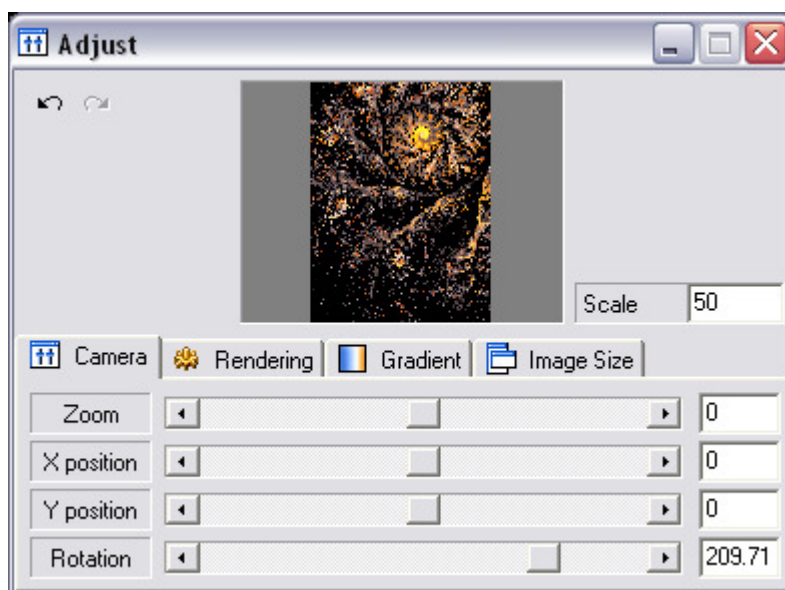
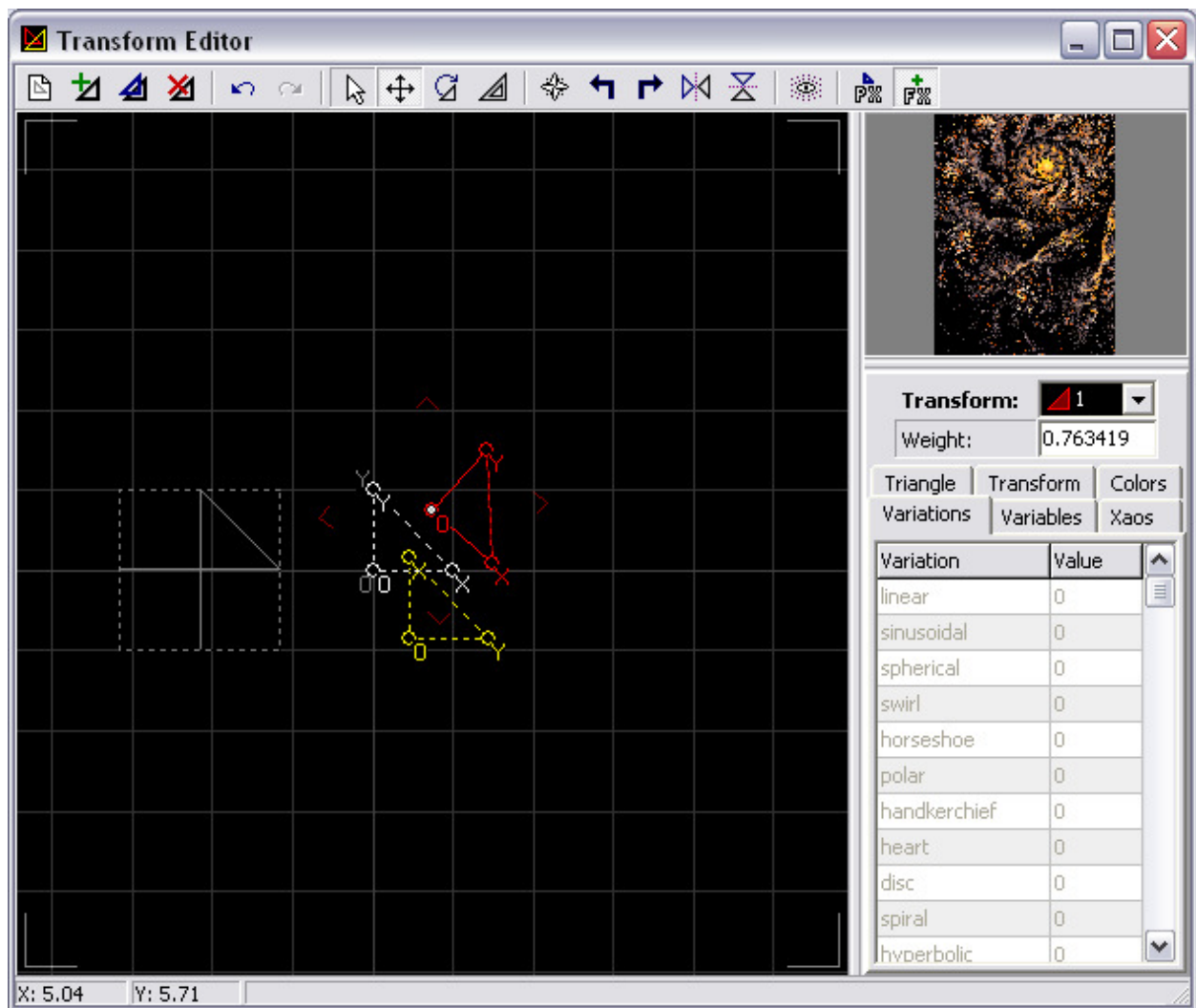
The stumbling block to this had always been flame rotation coupled with off-centre positioning. The scripting language doesn't seem to allow for rotation of the final transform about the World Pivot, never mind its post transform. However, this crucial stage of 'neutralising' the flame rotation by transferring it to the final post transform can be achieved in two simple steps, and the breakdown of the flame into strips for separate rendering is accomplished using a script.

Neutralising flame rotation

Illustrating with an example, note the settings at each stage within the Transform and Adjust Editors. Skip this stage if the Rotation setting on the Camera tab is zero.

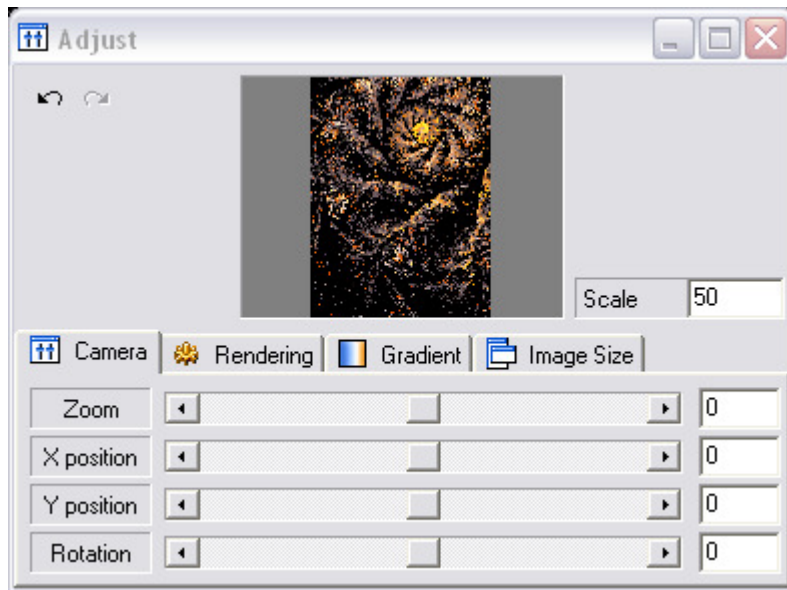


Load and run the script **ID_Position-with-final.asc**.



The X and Y positions are now zero. Just the rotation to take care of:

-
- Transform Editor**
- Transform: **Final**
- Weight: n/a
- Variations Variables Xaos
Triangle Transform Colors
- X: 2.37900 Y: 0.109012
O: 1.88347 1.05836
- 209.7
0.1
150
- Pivot Point
0 0
R World Pivot P
- X: 4.55 Y: 5.71 Zoom: 0.85



Preparing and rendering the strips

This has always been possible, but I'd never seen a mathematical treatment of the subject before. This has now all been taken care of within the script: the only user input required is to provide the desired render width and the number of strips. Note that the number of strips must divide exactly into the height. Obviously, this must be large enough to divide the memory requirement to within that available. Also, it is assumed that the currently loaded flame is the flame of interest, and that its aspect ratio is that desired.

Working through with some example figures:

preview picture: 640 x 480 px

desired width: 12000 px

of strips: 3

filename: example

Running the script **ID_Strips.asc** with suitable inputs generates a file called example.flame in the install folder. This file contains three parameter sets labelled example-1 to example-3 whose specified render dimensions are 12000 x 3000 px.

These should be rendered using the same values for quality, oversample and filter radius. This may be accomplished in a single operation using the Render All Flames command from the Flame menu item (also Ctrl-Alt-R). You'll need to render to .jpg for this size of image.

Assembling the final image

Instructions here pertain to IrfanView, an essential piece of software for any image-fancier:

- **Image, Create Panorama image...**
- select **Vertical**
- **Add images** then navigate to their folder
- select the required set, **Open**
- ensure that the sort order is correct, that is the -1 suffix at the bottom – this is achieved by **Sort images** and selecting **Name (descending)**
- **Create image** – your image should be ready in seconds
- save the file, and you're done!

Theorists may note that due to the stochastic nature of the iterative process, there may be slight differences at the joins. To those I say "Try a small test and then attempt to locate the joins."