

## Using flam3 with Apophysis #1: basics

### Getting flam3

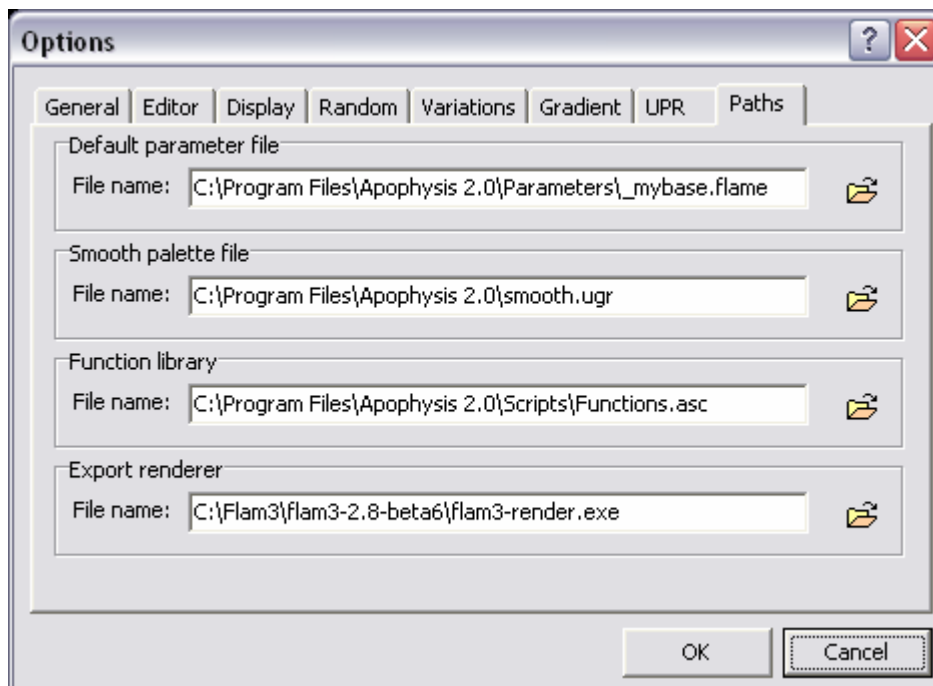
At time of writing, there are three current versions. The 2.8 beta series is recommended for use with Apophysis as it supports a significant proportion of plugin variations. Direct downloads for all:

Version	Download link
Sheep server 2.7	<a href="http://flam3.com/flam3-2.7.18.zip">http://flam3.com/flam3-2.7.18.zip</a>
2.8 beta 6 ( $\leq 2$ cores)	<a href="http://flam3.com/flam3-2.8-beta6.zip">http://flam3.com/flam3-2.8-beta6.zip</a>
2.8 beta 6 ( $\geq 4$ cores)	<a href="http://flam3.com/flam3-2.8-beta6-atomic.zip">http://flam3.com/flam3-2.8-beta6-atomic.zip</a>

### Installation

Simply unzip the package to your chosen location: a subfolder containing all relevant files is created. If a different location is desired, always move the entire folder.

To register with Apophysis, in Options on the Paths tab, under Export renderer, use the folder icon to open a dialogue then navigate to the folder and select flam3-render.exe:



## Compatibility

As already mentioned, the latest beta has much greater Apophysis compatibility. A complete list of supported variations is contained in the readme.txt file. If you don't use plugins, or only the 'official' Sourceforge pack\*, then full compatibility is assured.

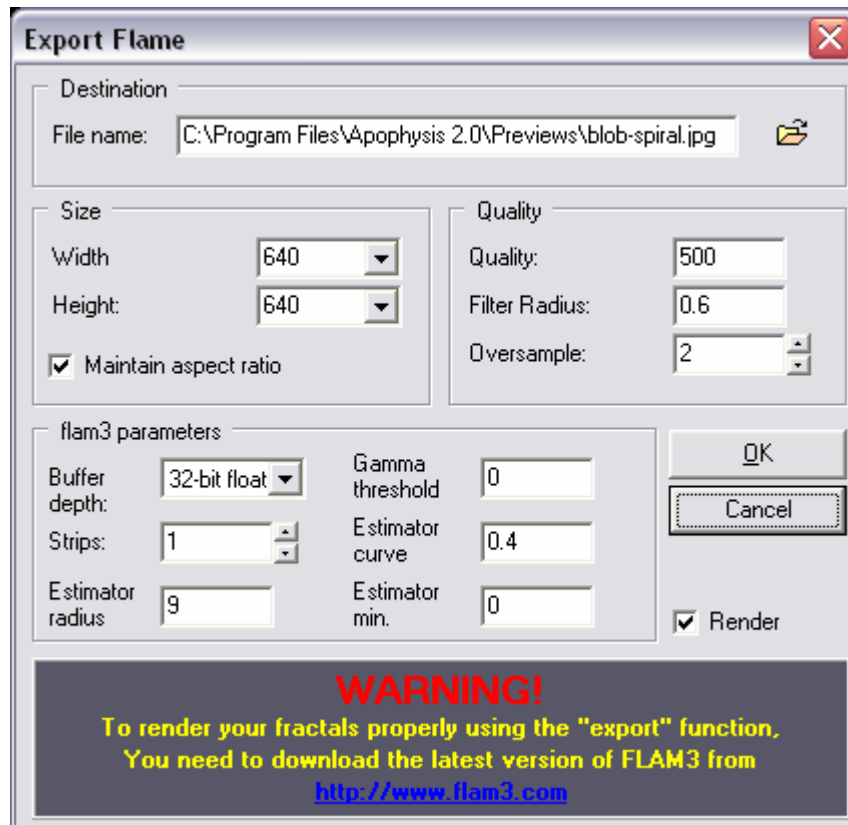
\*[http://sourceforge.net/project/showfiles.php?group\\_id=201476](http://sourceforge.net/project/showfiles.php?group_id=201476)

Beyond the built-in variations, supported plugins are:

pie	disc2	cpow	polar2	whorl	sinh
ngon	super_shape	curve	popcorn2	waves2	cosh
arch	flower	edisc	scry	exp	tanh
tangent	conic	elliptic	separation	log	sech
square	parabola	escher	split	sin	csch
rays	bent2	foci	splits	cos	coth
blade	bipolar	lazysusan	stripes	tan	auger
secant2	boarders	loonie	wedge	sec	
twintrian	butterfly	modulus	wedge_julia	csc	
cross	cell	oscilloscope	wedge_sph	cot	

## Use

Select File, Export Flame... or use Ctrl-X to open the rendering dialogue:

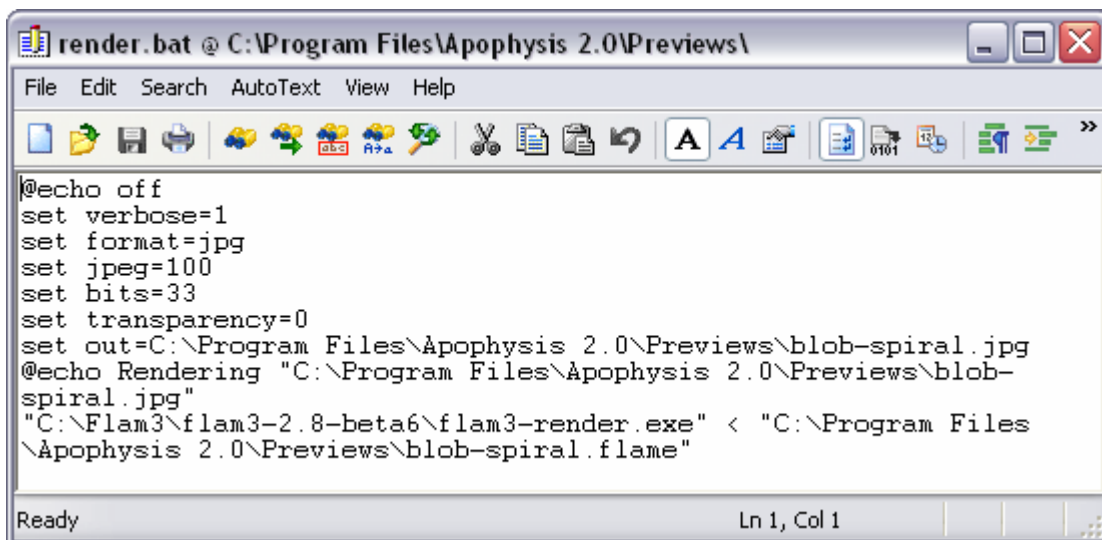


## Features and considerations

- Basic rendering parameters as internal renderer.
- Buffer depth recommended 32-bit float for a 32-bit OS, 64-bit for 64-bit OS.
- Strips: rendering in  $n$  strips multiplies the render time by  $n$ , hence always leave at 1. This shouldn't be an issue for anything barring poster-sized prints. There is a workaround (see 'Rendering in manual strips' at my resources page: <http://www.ultragnosis.com/fractals/resources.html>).
- Next are the four density estimation parameters. At this stage, simply leave the default values. Experience, knowledge and experimentation are required here – more to say in part 2.
- Checking 'Render' will automatically begin rendering. If a command prompt (DOS window) doesn't remain visible, use Task Manager to confirm that rendering is indeed occurring.
- Flam3 automatically uses all installed cores. Either use Task Manager to lower the process priority, or see next section.

## Editing the .bat file manually

Flam3 works by creating a single .flame file (without the <Flames> element) and a batch file (.bat) with instructions to pass this .flame file to the renderer and output as a picture file of selected format (.jpg, .png or .ppm). An example of a batch file:



```
@echo off
set verbose=1
set format=jpg
set jpeg=100
set bits=33
set transparency=0
set out=C:\Program Files\Apophysis 2.0\Previews/blob-spiral.jpg
@echo Rendering "C:\Program Files\Apophysis 2.0\Previews/blob-
spiral.jpg"
"C:\Flam3\flam3-2.8-beta6\flam3-render.exe" < "C:\Program Files
\Apophysis 2.0\Previews/blob-spiral.flame"
```

The crucial statements may be generalised thus:

```
set out=fullpath1\filename.ext
```

```
"path2\flam3-2.8-beta6\flam3-render.exe" < " fullpath1\ filename.flame"
```

Open the batch file with a text editor. To ensure that the command prompt remains visible during the rendering process, insert a line "set verbose=1" as seen above, if not already present – this should remain in place for all future renders. To amend the number of cores flam3 uses, insert a further line "set nthreads=c" where c is the desired number of cores – this will require setting each time, so don't check the render box if this is required.

### Another method

A more complete GUI for flam3 is provided by Fr0st:

<https://launchpad.net/fr0st/+download>

Currently in its first beta release, this provides some nice features re-hashed from the Apophysis GUI as well as easy access to some of flam3's more advanced parameters (Early Clip, nthreads setting and even filter kernel choice for the density estimation). A project well worth keeping abreast of.