

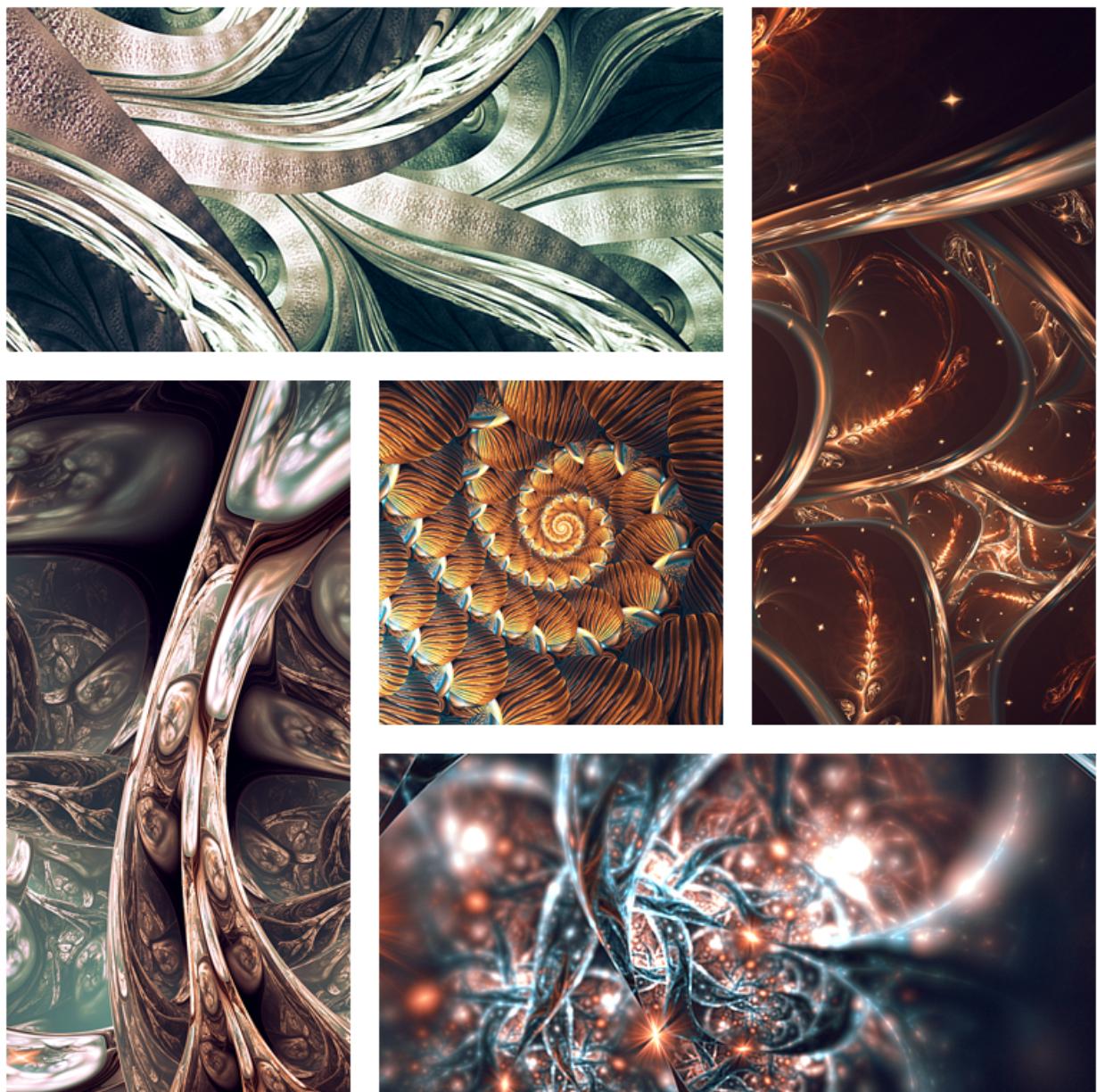
1 Introduction

Attention: this is an **INTERMEDIATE** to **ADVANCED** tutorial, and it requires some previous knowledge of how Apophysis / Chaotica and xoas / weights work. I will not go deep into explaining the interface and so on, so you may need to check some beginner tutorials first.

The goal of this tutorial is to go through a few of the "**half-plane**" transforms – transformations that map the whole plane to a half-plane, such as flux – and a few strategies of how to build structured IFS fractals with them.

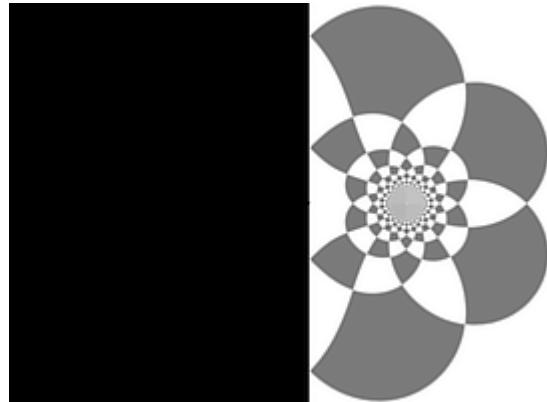
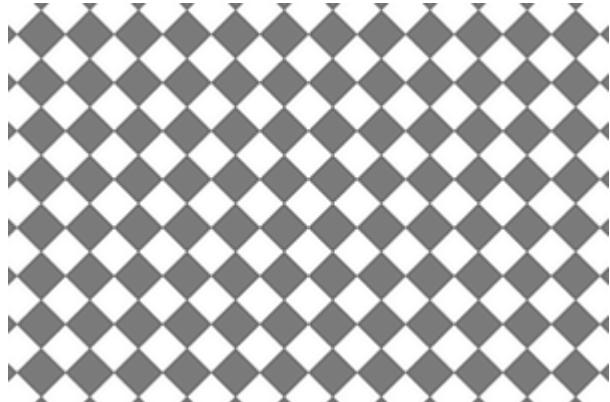
Most of the transforms / plugins mentioned in this tutorial can be found in the Aposhack Plugin Pack:
<https://phoenixkeyblack.deviantart.com/art/The-Aposhack-Plugin-Pack-98807426>

Below, you can see a few examples of what you can achieve using container transforms:



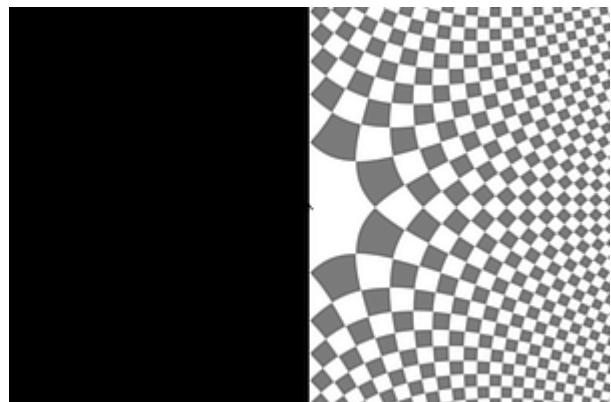
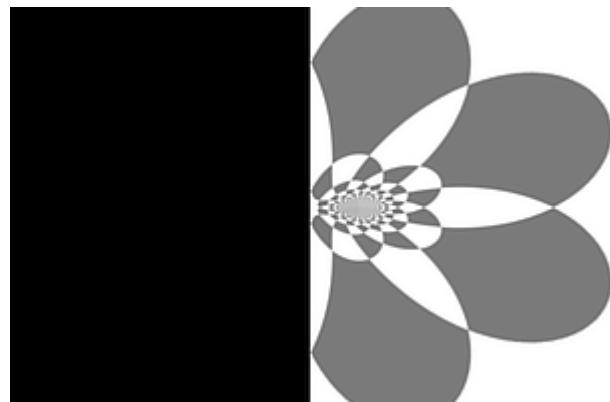
2 One transform Half-planes

The most common half-plane transform is **flux**. Below, you can see what happens when we apply flux on a check board pattern (check board, left, and flux, right):

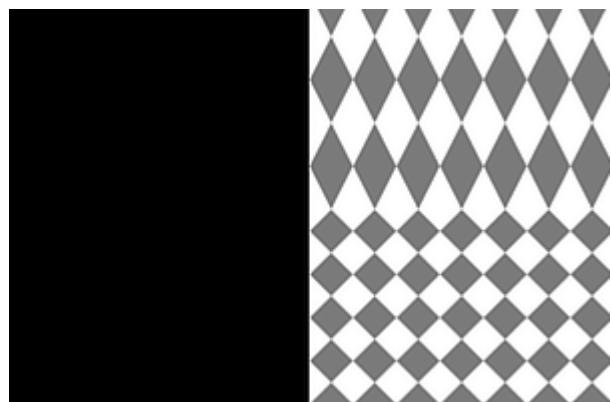
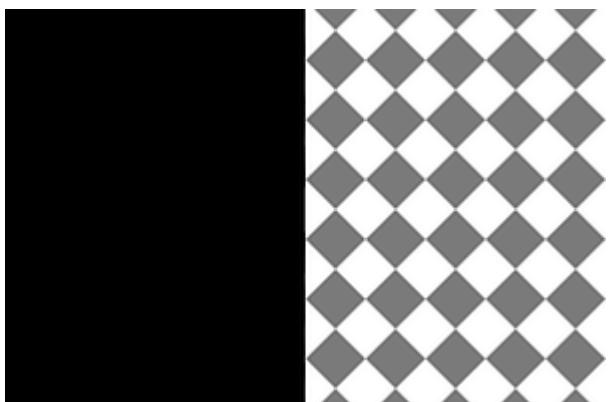


Observe how flux takes the whole plane, and maps it to the right half of the plane, leaving the left half empty.

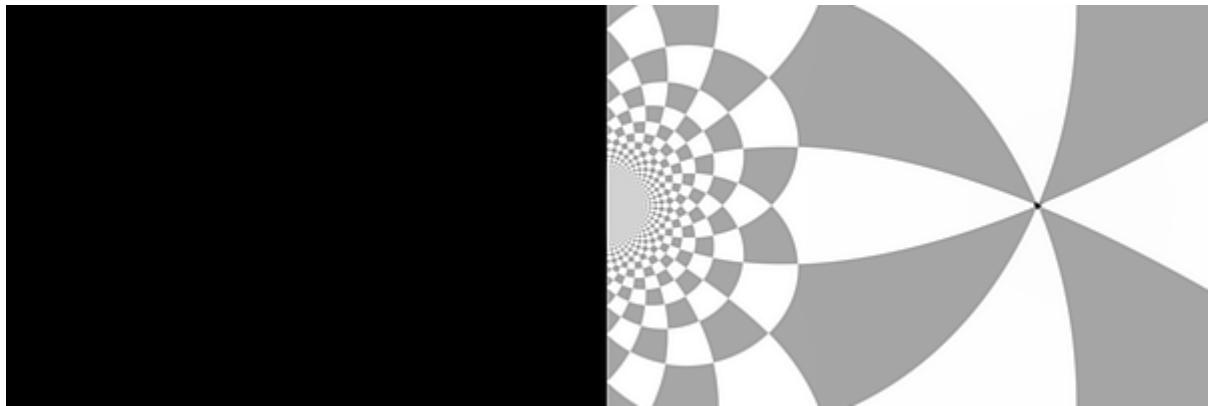
There are several other transforms that produce similar effect, among them **DeltaA** and **eScale** (with scale equal to 0.5). Note, that while they are somewhat different, half of the plane always remains empty.



Another transforms that produce a half plane are **crop** (there are several crop transformations available, but I strongly recommend using **smartcrop**) and **bent2** (with bent2x = 0 or bent2x = -1, for example):



Another example of one-transform half plane is **PlusRecip** by dark-beam (can be downloaded from his deviantart page).



Finally the last one is the polar + linear framework by Fardareismai
(<https://www.deviantart.com/fardareismai/art/Polinear-Examples-765766115>):



3 Basic filling

Now that we know how to get a half plane, we got to learn to fill the other half. The basic idea is pretty simple: the same that fills the right half fills also the left half.

3.1 Flipped Linear

The easiest way to fill a half plane is by using a mirrored linear, which means you just mirror whatever you have on one half to the other half. Follow the steps below:

1. Start with a blank flame
2. On first transform, replace linear with **flux**
3. Add another transform, and **flip it horizontally** (or rotate it 180 degrees)

The result should look like this:



If you had any troubles, you can download parameters here: <https://pastebin.com/BRJdcLJq>

Tweaking tips:

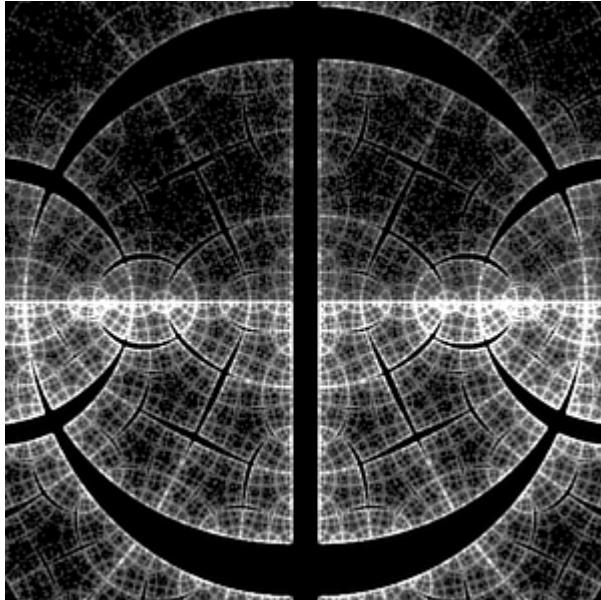
- Moving or rotating the linear will break the pattern
- You can get some really cool results by moving, scaling and rotating the flux transform
- Do not forget that flux has a variable, which you can edit change

3.2 Flipped Linear with Gap

You can "upgrade" this pattern by creating a central split and filling it with something. Basic step by step:

1. Move the **post transform** for flux 0.1 to right. This will create an empty vertical space in your pattern.
2. Add a new transform, replace linear with **cylinder** (use cylinder = 0.1)
3. To same transform, add **pre_blur** = 10
4. Switch to the transform tab and, for the post transform, change Y from 1 to 10
5. Finally, **reduce weights** to something between 0.01 - 0.05

Below, flux + linear framework after step 1 (left picture) and the parameters for steps 4 and 5 (right):



Weight:	0.02	
Variations	Variables	Xaos
Triangle	Transform	Colors

Transform

X	1	0
Y	0	1
O	0	0

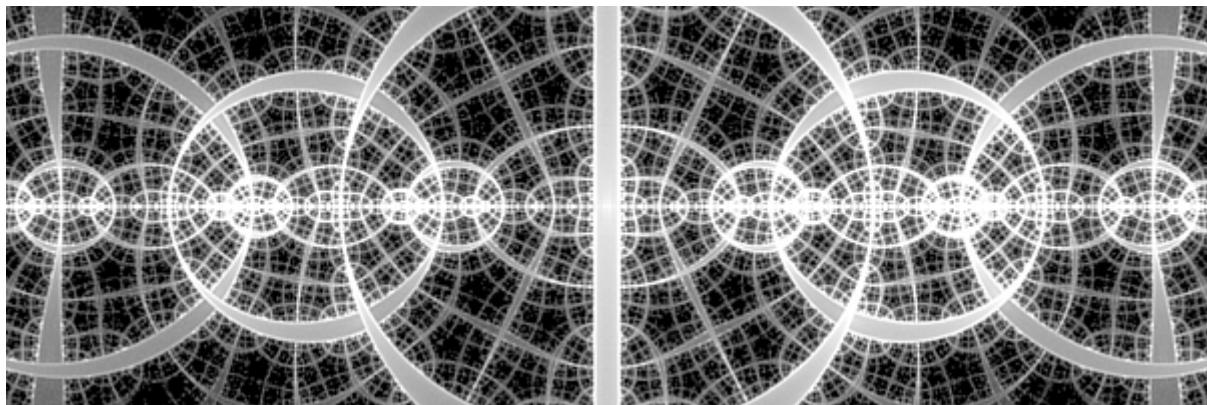
Reset transform

Post transform

X	1	0
Y	0	10
O	0	0

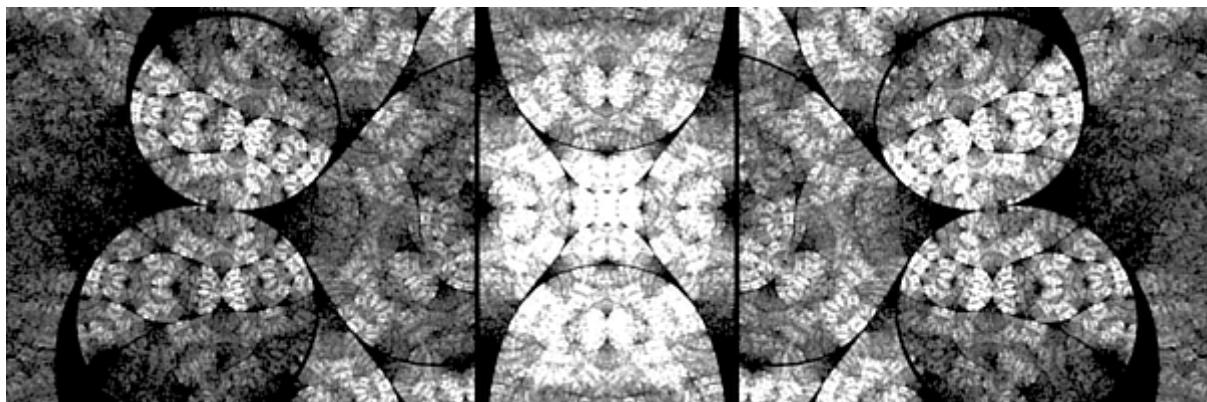
Reset post transform

Your fractal, after step 5, should look like this:



Advanced tweaking: if you want to make something even more complicated, you can fill the gap with any vertical stripe you can come up with. Transforms such as elliptic and bipolar, or maybe a vertically tiled square.

Below, flux + flipped linear with the gap filled with elliptic:



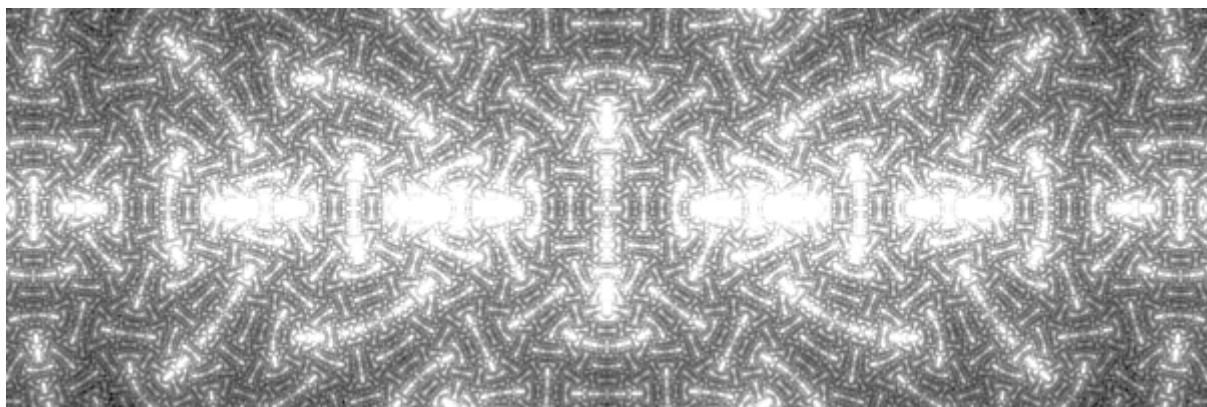
3.3 Flipped half plane

The flipped linear fills the empty half with the same pattern we have on the other side, without adding any additional distortion to it. Instead, we can use another half plane transform (either 2 of the same, or 2 different ones).

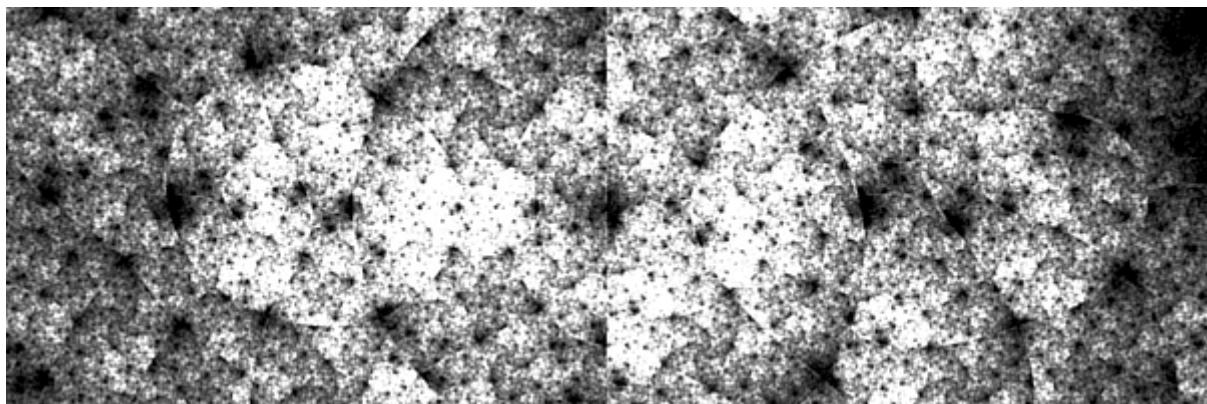
Here goes a step by step:

1. Start with a blank flame
2. Add a new transform, replace linear with **flux**
3. Add another transform, replace linear with **flux**.
4. Select the **post transform**, and **flip it horizontally** (or rotate it 180 degrees)

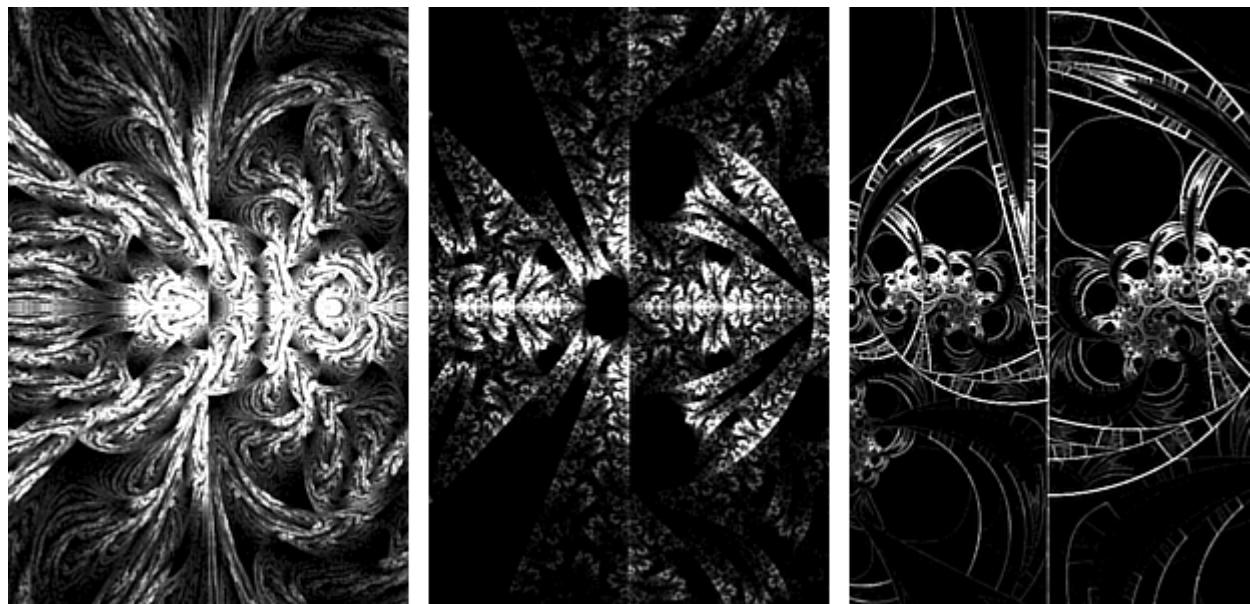
Does not look much different from previous, right?



Try moving around, rotating and scaling one of the flux transforms to see how this set up enables you to create asymmetric patterns, like the one below:



You can also replace flux with other half plane transforms. For example, below you can see deltaa, escale and bent2 combined with flux:



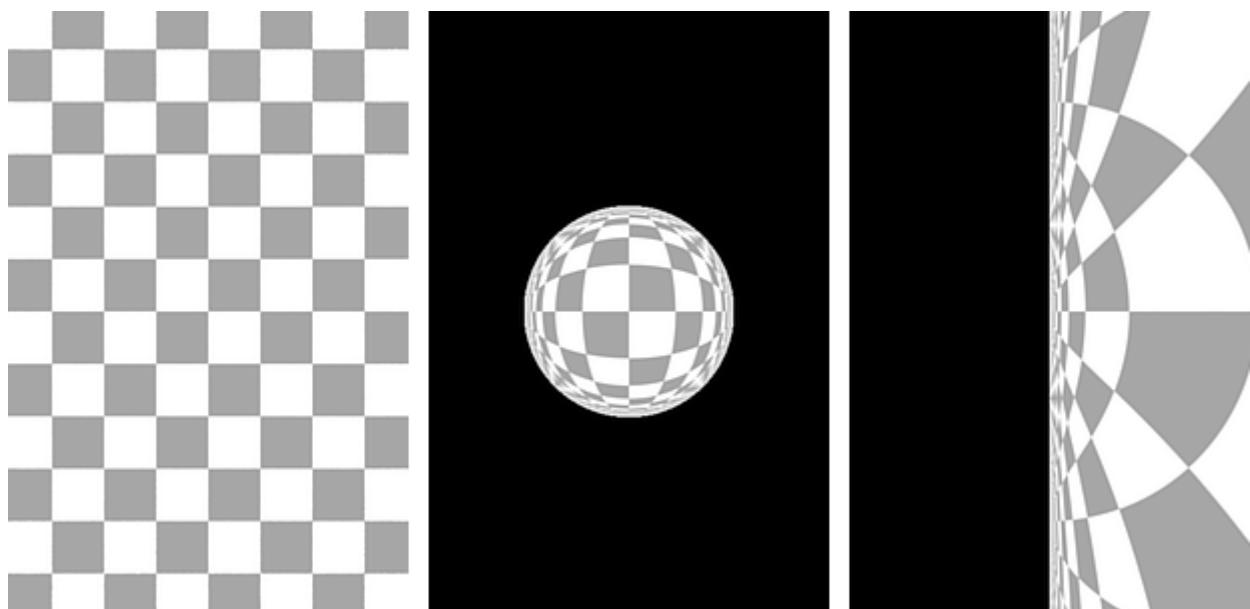
4 Circle to Halfplane

Another way to create a half plane is to take a unit disc and use curl to map it to halfplane. Unit discs can be produced by transforms such as hemisphere, scry or bubble (and many others) or frameworks such as disc and hypertile.

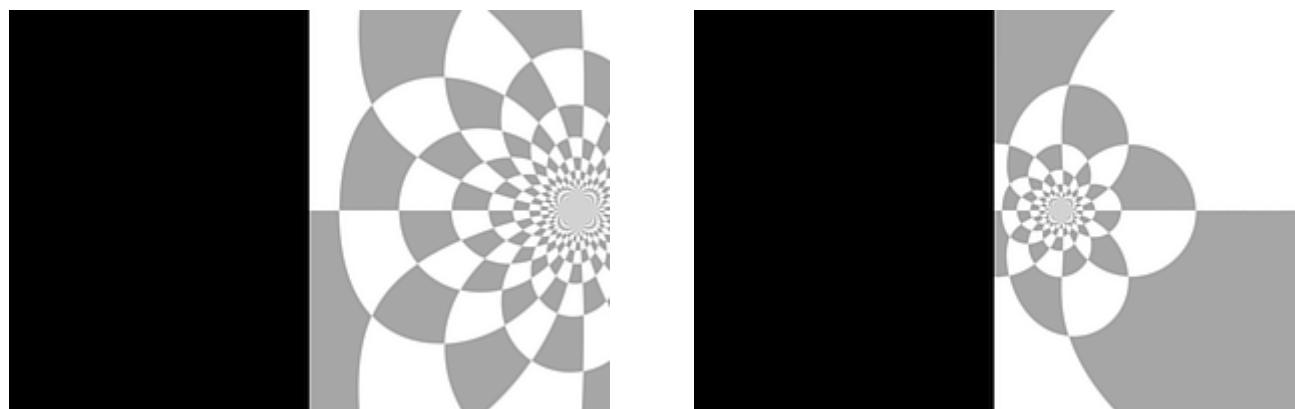
A classical example of this approach is the Double Bubble technique described here:
<https://www.deviantart.com/piethein21/art/Double-Bubble-Tut-535898713>

If you look closely, you will see that the tutorial features two half planes, both made with hemisphere and curl.

Below, you can see the whole plane (left) mapped first to a hemisphere (middle) and then, with curl, to a half plane (right).



It is important to observe that there is no unique way to map a disc to halfplane using curl. In this tutorial, I will present two different ways, leaving it to the user to explore the other possibilities. Below, two different setups for the scry + curl combination:

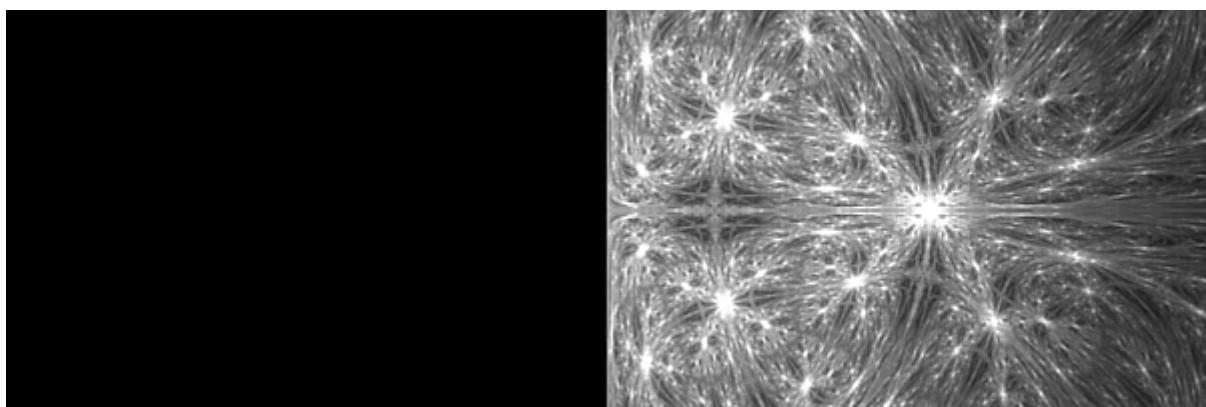


4.1 Curl Setup with `curl_c1 = -0.5`

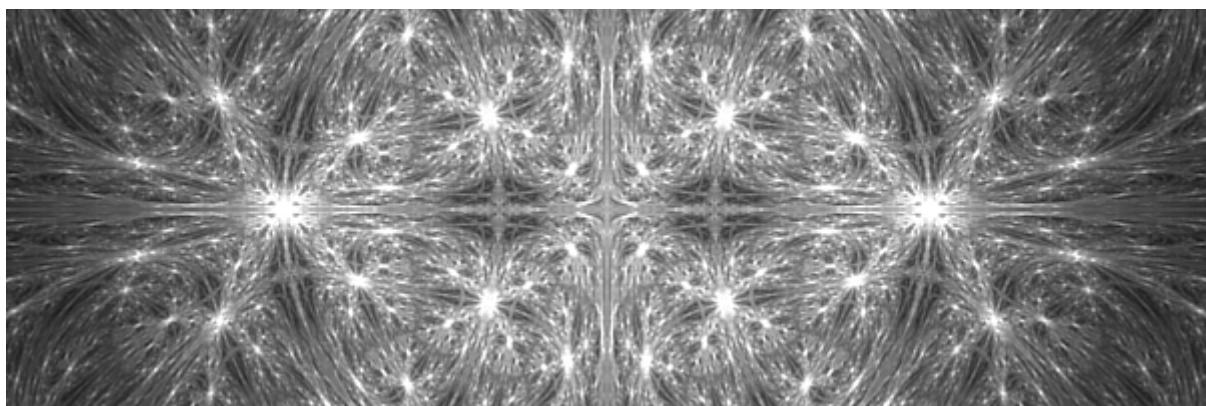
In this example, we will use a very simple setup, just to illustrate how the method works.

1. Start with a blank flame
2. On first transform, replace linear with `scry`.
3. Rotate it by 90 degrees CCW, and scale it down by 200% once
4. Add a linked transform to it, and replace linear with `curl`, set `curl_c1` to -0.5
5. Move this linked transform **1 unit right**
6. **Duplicate** this second transform
7. Flip the **post transform** horizontally

In this setup, each curl transform corresponds to a half plane. Here, I fill both halves with the same thing, but you can use something different, for example flux or a hemisphere with linked curl (it may require some xoas). This is how this fractal looks with only the transform 2 visible:



And this is the whole thing:



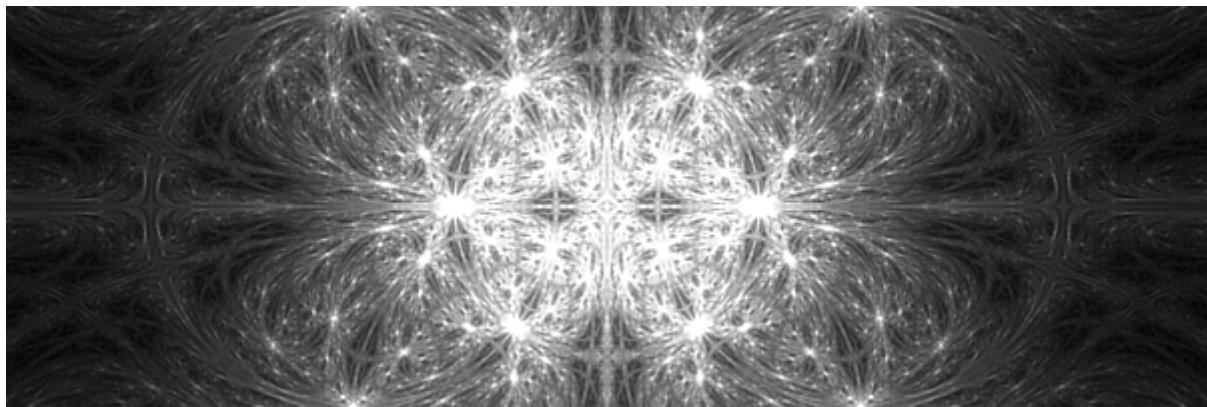
If you had any troubles, you can download parameters here: <https://pastebin.com/3ZMMnbhr>

4.2 Curl Setup with `curl_c1 = -1.0`

This is very similar to the previous example, the only difference is how we set up curl:

1. Start with a blank flame
2. On first transform, replace linear with `scry`.
3. Rotate it by 90 degrees CCW and scale up by 200%
4. Add a linked transform to it, and replace linear with `curl`, set `curl_c1` to -1
5. Move the **post transform** of this linked transform 0.5 unit right
6. Duplicate this second transform
7. Flip the **post transform** horizontally, then move it 1 unit left (so it is positioned at -0.5)

Your final result should look exactly like the previous one:



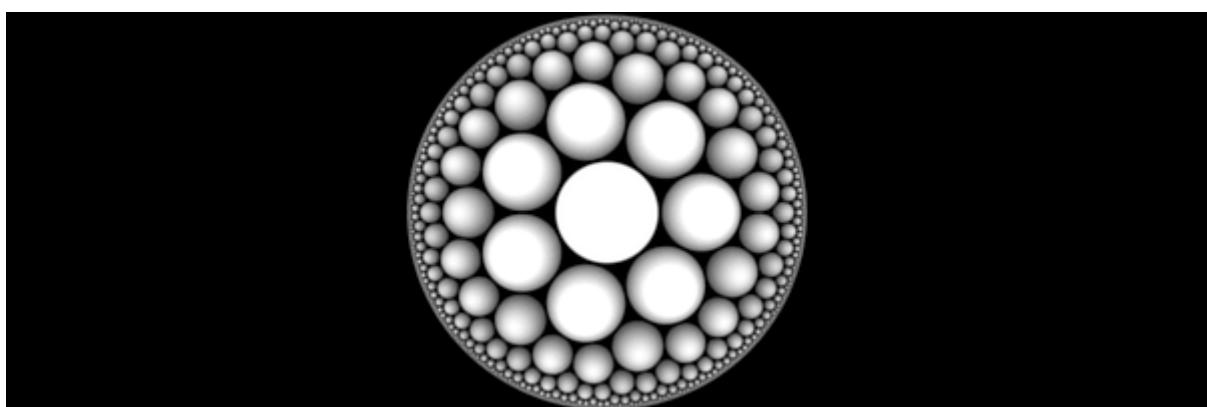
If you had any troubles, you can download parameters here: <https://pastebin.com/G0NEhjcD>

So, why all the hassle if it looks all the same? The answer is: this is actually useful when you are using some weird unit disc patterns, as it gives you more flexibility in how to distribute it over the half plane.

4.3 Filling Flux with Hypertile

Now, we will use the curl trick to fill flux with hypertile.

We will start with something like this:



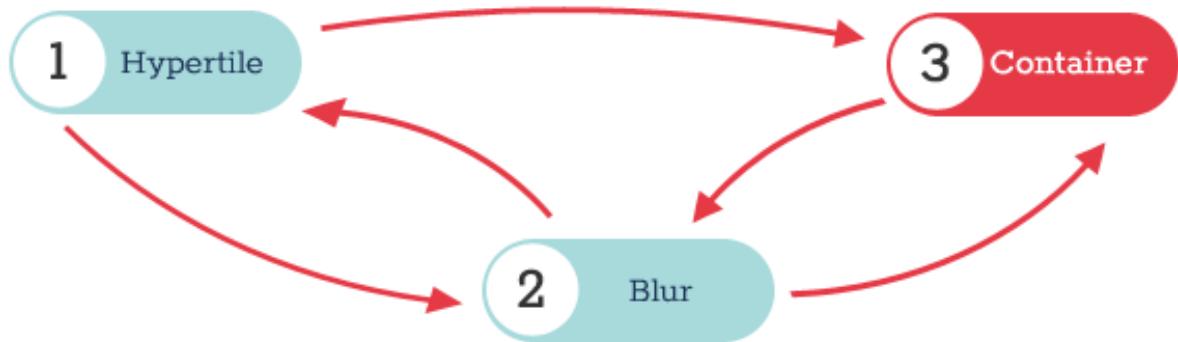
To make a basic hypertile filled with blurry circles, you may:

- Do it from blank all by yourself
- Follow this tutorial: <https://www.deviantart.com/tatasz/journal/Hypertile-Basics-535943140>
- Grab those params: <https://pastebin.com/AqD6R0KR>

And then:

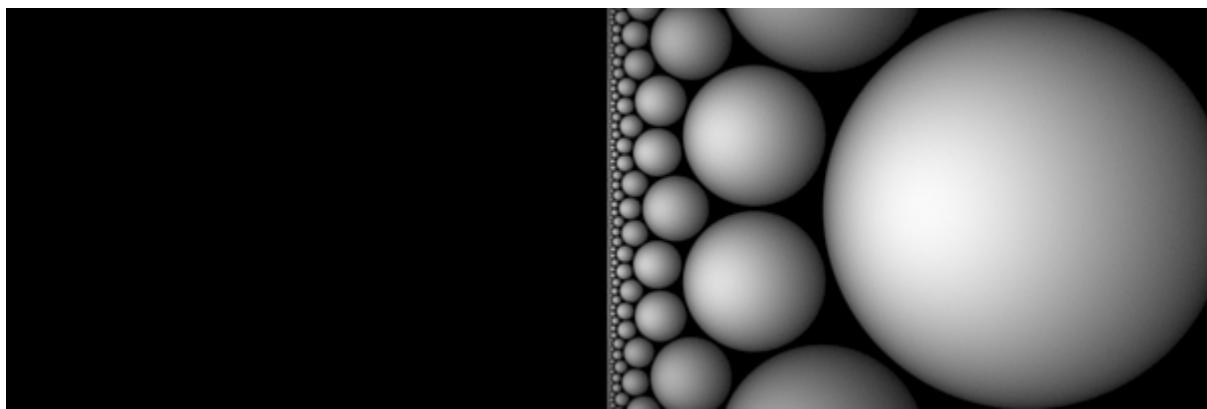
1. Start with setting the **opacity** of all the transforms to 0
2. Now, add a new transform, and set its **color speed** to 1
3. Go to the xaos tab and switch to "to" xaos. This transform should only send points **to** transforms with blur (transform 2 if you use my params)
4. Switch to "from" xaos. This transform receive points **from** both hypertile and the circle
5. Once you set everything up, play with weight of this transform to get a better look.

This is the xaos setup for steps 3 and 4:



6. On the container transform, replace linear with **curl**, set **curl_c1** to -0.5
7. Move this transform 1 unit right

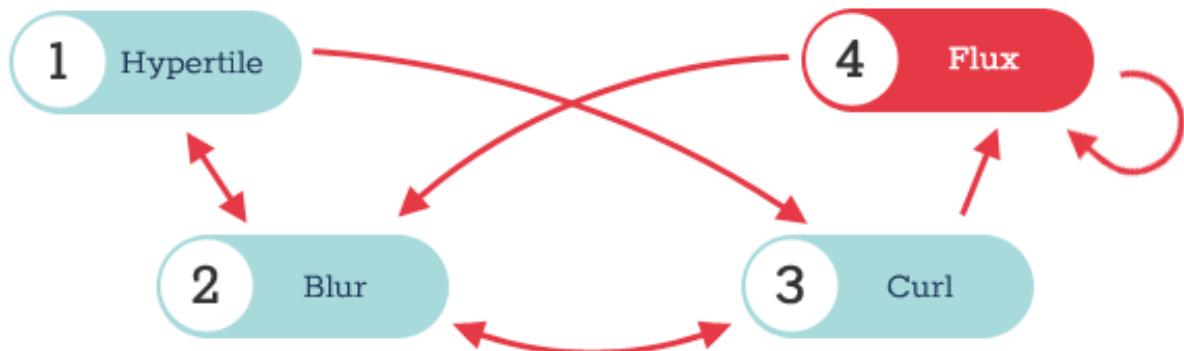
And here we go, a nice hypertile halfplane:



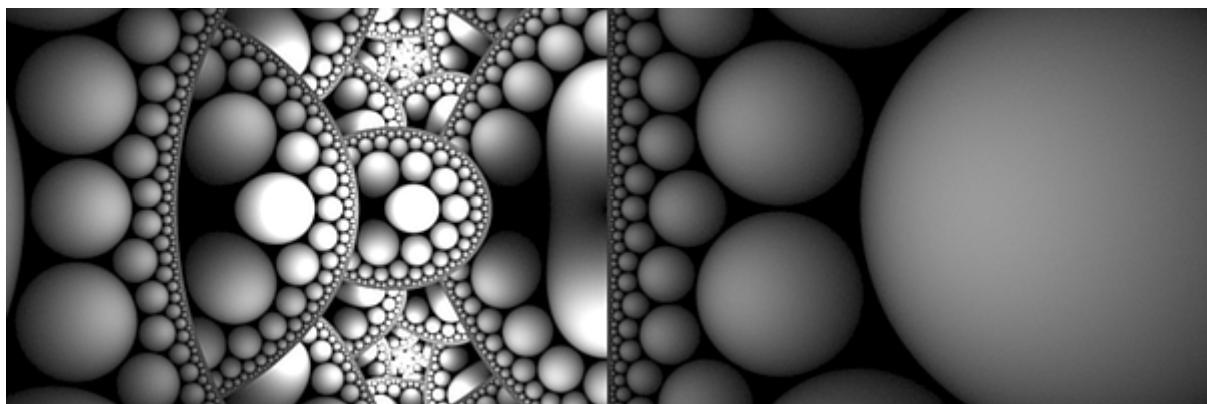
We just need to add a flux and it is done:

8. Add a new transform, replace linear with **flux**
9. Flip its **post transform** horizontally to fill the left halfplane
10. On xaos tab, set it to send points only **to** itself and the blur transform
11. And to receive points only **from** itself and the curl transform
12. Adjust the flux weight

Below, the xaos setup:

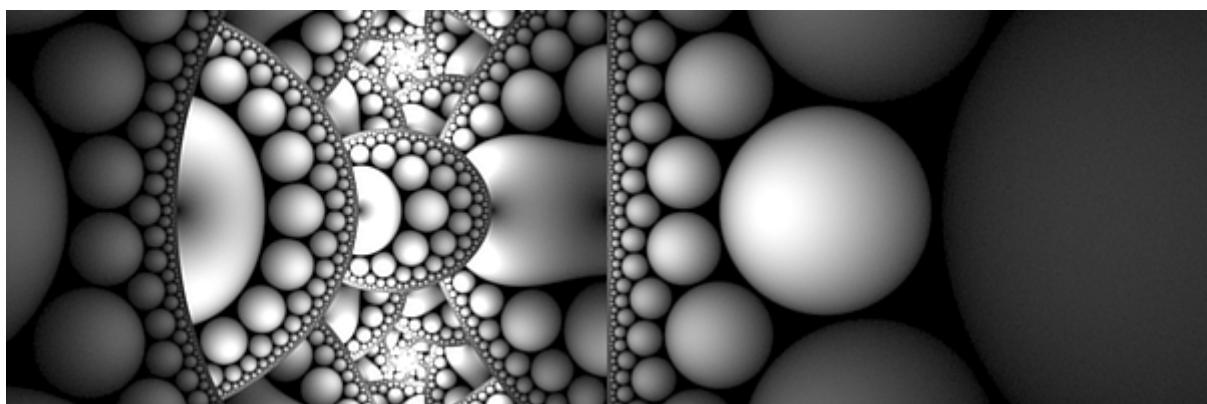


And the fractal should look like this:



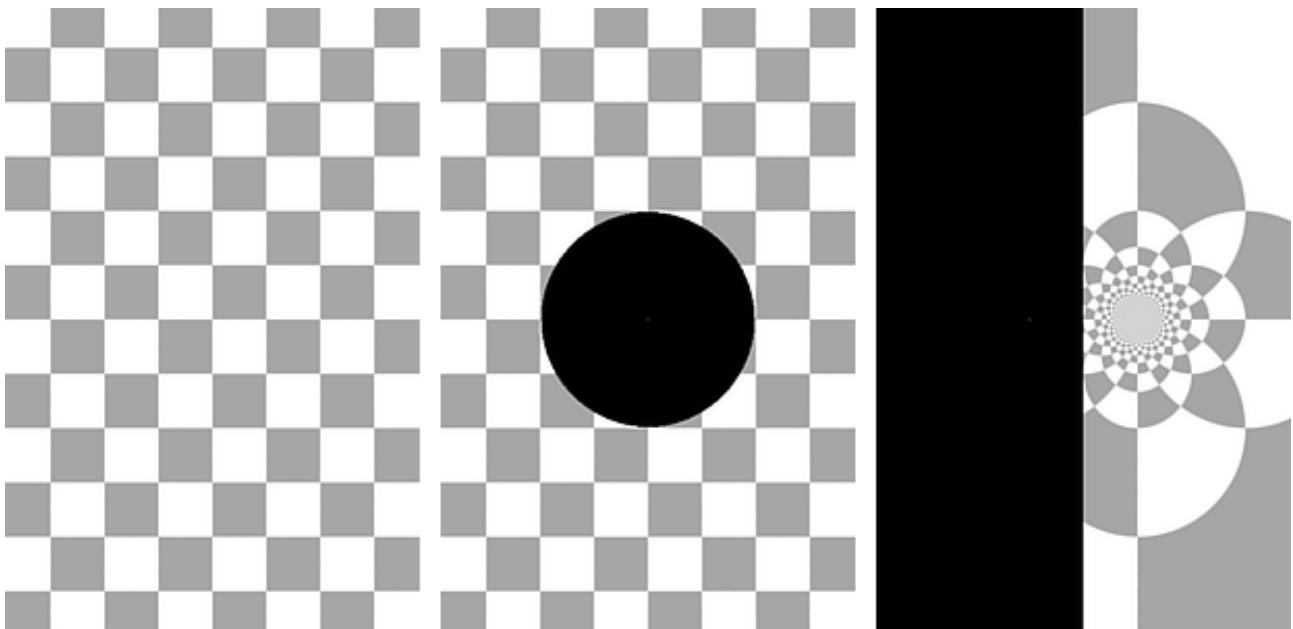
If you had any troubles, you can download parameters here: <https://pastebin.com/FNS4jDq6>

For comparison, the same fractal with a different curl setup (I used curl_c1 = 0.75):



4.4 Hole to halfplane

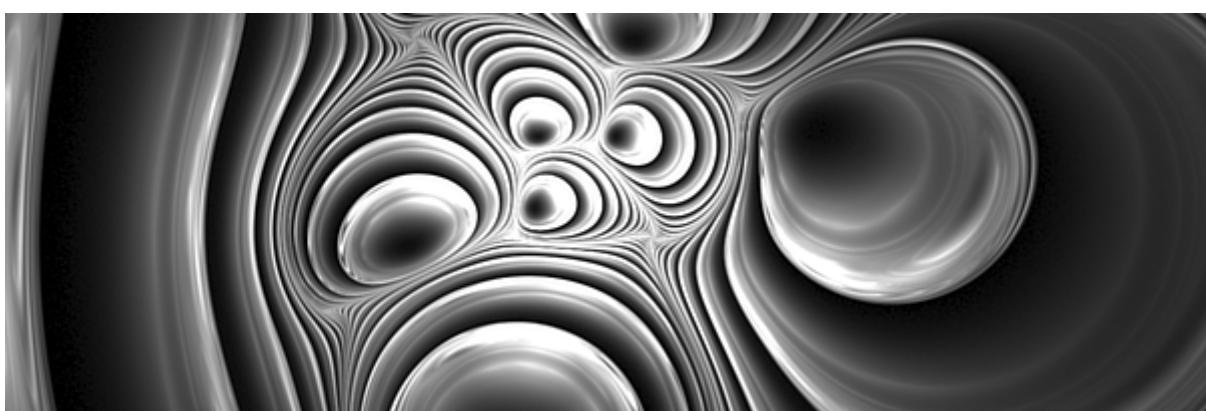
If we take the inverse of a circle, we can use the same technique from the previous sections to map it to a half plane. On the picture below, you can see a checkboard pattern (left), same pattern with a hole cut in it (middle) and, finally, curl applied to map it to halfplane (right).



While you can easily convert a hole to disc (using a linked spherical to invert it), sometimes using hole to halfplane is more natural.

To see how this works, we will map a plastic to halfplane.

1. Start with a blank flame
2. On first transform, replace linear with **spherical** = 1 and **eyefish** = 0.5
3. Add a new transform, replace linear with **hemisphere** and add **pre.blur** = 10 or so.
4. Now, back to transform 1, increase its **weight** to 5-10
5. **Scale** this transform down, **move** and **rotate** it, unless you get some nice shapes



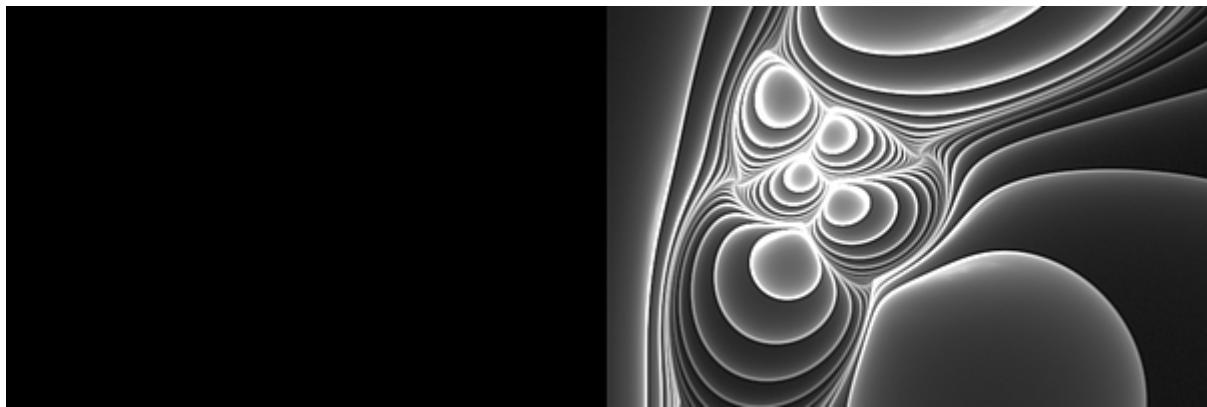
If you had any troubles, you can download parameters here: <https://pastebin.com/7mhganQn>

If you have troubles seeing where the plane with a circular hole is, set the hemisphere transform opacity to 0, and look only at the spherical + eyefish.

Now, time to half plane it:

6. Set all the **opacities** to 0.
7. Add a new transform, set its **color speed** to 0
8. Set it to send points only **to** the hemisphere + blur transform
9. And to receive points only **from** the spherical + eyefish transform
10. To avoid overlaps later on, set the hemisphere transform to receive points only **from** the last transform
11. Replace linear with **curl**, set curl_c1 to 0.5
12. **Move** the last transform 1 unit left

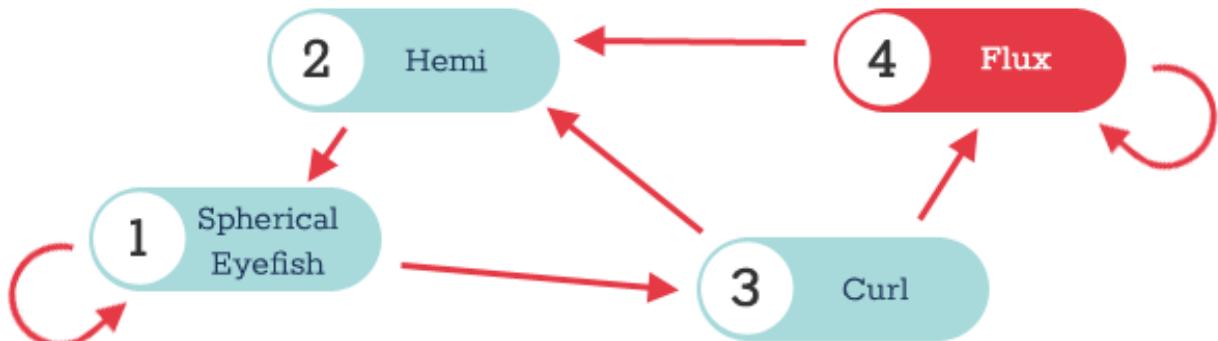
And we have a halfplane:



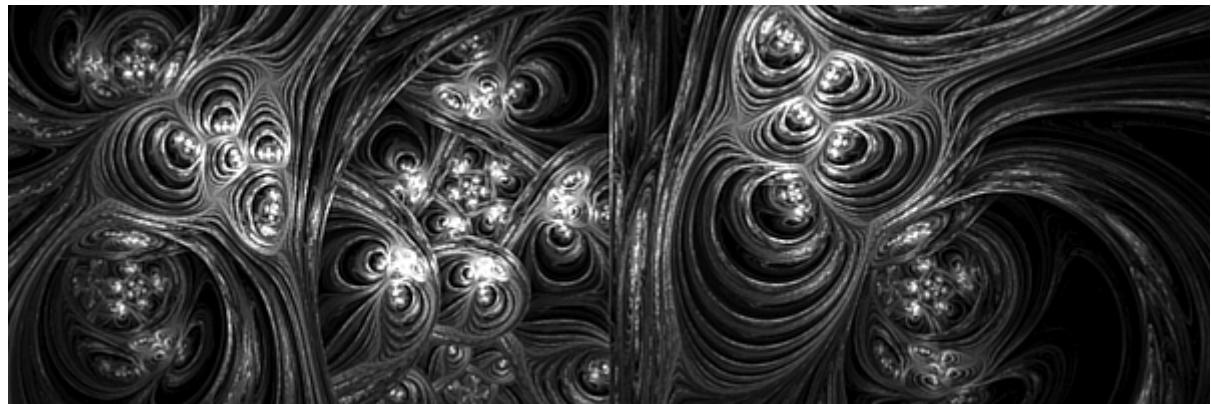
Time to fill the other half with a flux:

13. Add a new transform, replace linear with **flux**
14. Activate **post transform**, and **flip** it horizontally
15. Set it to send points only **to** the hemisphere + blur transform
16. And to receive points only **from** itself and the spherical + eyefish transform
17. Increase the weight a bit

This is how the xoas setup looks like:



The fractal should looks like this:



If you had any troubles, you can download parameters here: <https://pastebin.com/CXLTTApT>

5 Strip to Half Plane with Unpolar

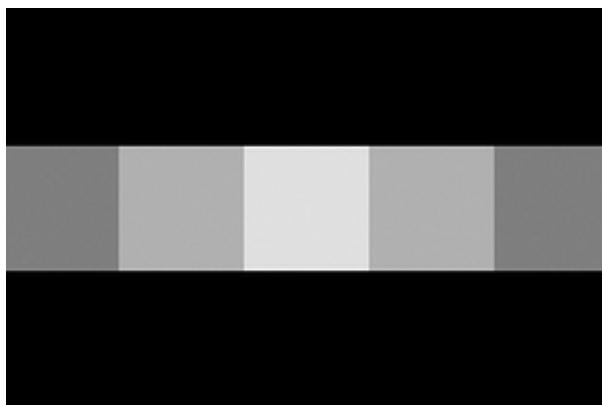
In this chapter, we will need a few plugins:

- Exp2: <https://www.deviantart.com/tatasz/art/Hyperbolic-Functions-684094800>
- Tile_hlp (by zyOrg): <https://app.box.com/s/1938fjgymb7vlwnnropvpbyuwglmaour>
- Tile_log (by zyOrg): <https://app.box.com/s/t9rdnferuxdk4uwono7oz36uzoikhk1ke>

We start with creating a stripe. I usually start with a blurry square, which i replace with something fun later on, when the main framework is ready.

1. Start with a blank flame
2. On first transform, replace linear with **square**
3. Add a linked transform to it, replace linear with **tile_hlp**
4. Add a linked transform to transform 2, replace linear with **tile_log**

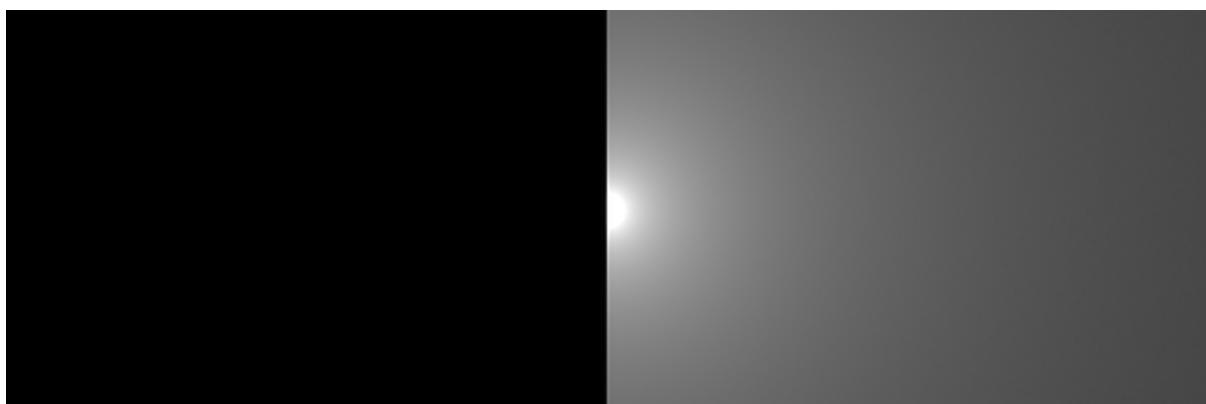
Tile_hlp smooths out your tile, while tile_log tiles the square into a horizontal stripe. Below, look at the output of step 4 (right) and the same stripe, with tile_hlp removed (left):



Now, we will use exp2. This transform is actually very same as unpolar, except it is multiplied by pi internally, which means it is much simpler to use and does not require scaling by multiples of pi.

5. Next, add a linked transform to transform 3
6. Replace linear with **exp2**

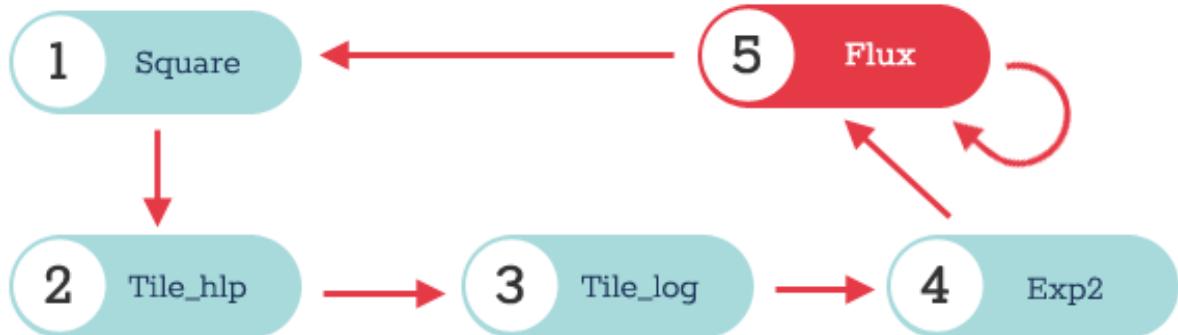
And we have a halfplane!



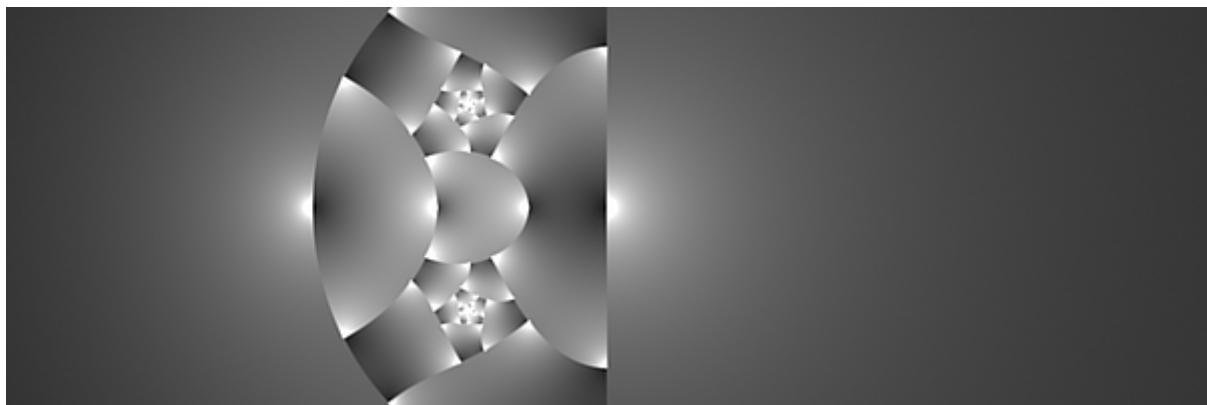
Time to fill the empty half plane:

7. Add a new transform, replace linear with **flux**
8. Flip the **post affine** horizontally
9. On xaos tab, set it to send points only **to** itself and the blur transform
10. And to receive points only **from** itself and the exp2 transform

This is the xaos setup:



And this is how your fractal should look at this point:

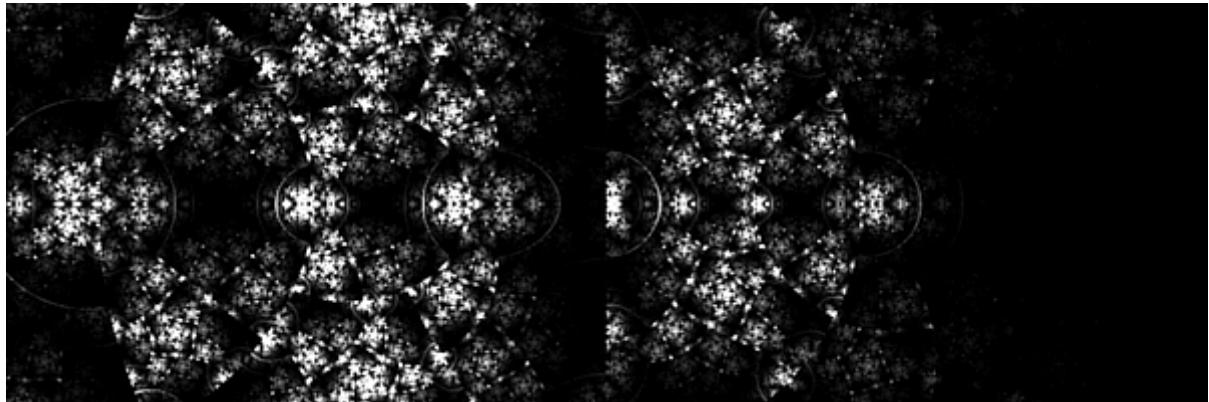


If you had any troubles, you can download parameters here: <https://pastebin.com/YyBORuRi>

And finally, as I promised, we will replace the boring square with something more fun (and that would be a cropped bipolar).

11. We will crop bipolar to a 2x2 square, which is bigger than the current 1x1 square we have. So first, switch to **post transform** and **scale down** by 200%
12. Replace square with **bipolar**
13. Add **post_smartcrop** (default settings are almost fine, just change static to 2)

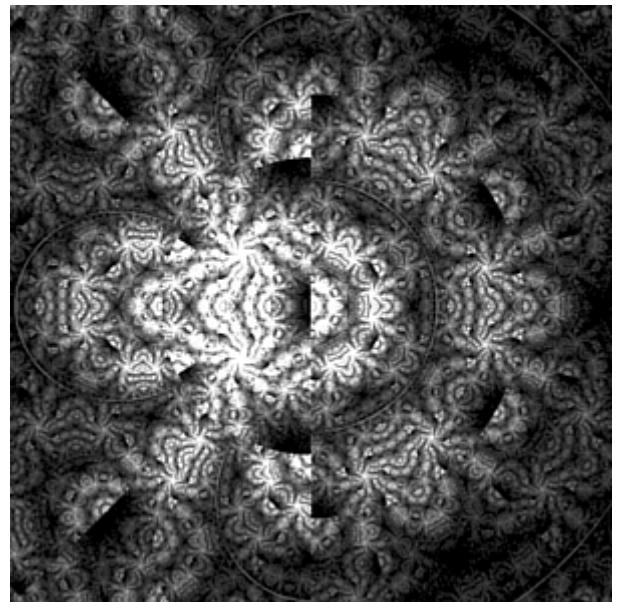
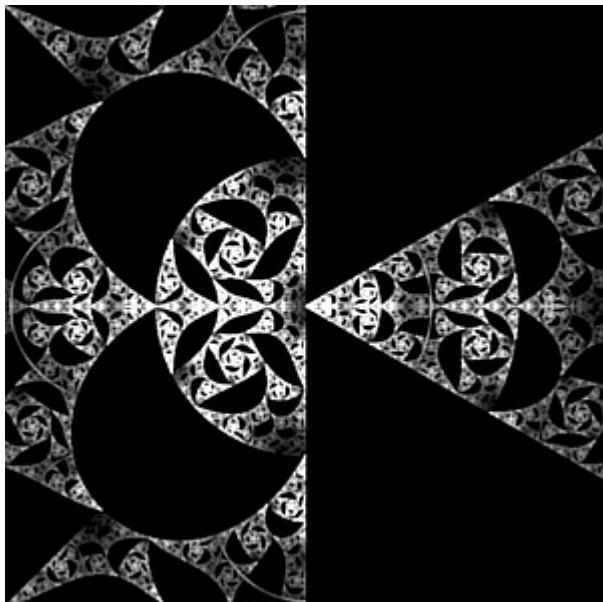
And we are nearly done:



We will now explore some possibilities of this setup. If you scale down the exp2 transform by 200%, you will have now a quarter plane. If we scale it down by 300%, we will have 1/6th plane, and so on. This means that we can scale it down, create several copies and then, with different rotations, still fill the half plane.

14. Scale down transform 4 (exp2) by 300
15. Duplicate it twice
16. Rotate first copy by 60 degrees CW
17. And the second one, by 60 degrees SWW

See how it looks after step 14 (left) and after step 17 (right):



If you had any troubles, you can download parameters here: <https://pastebin.com/Xc9E87ye>

6 Jacobi: square to halfplane

A bit different and a fairly fun setup, using jacobi plugins by dark-beam. They can be downloaded here:

- <https://www.deviantart.com/dark-beam/art/Jacobi-elliptic-sn-cn-and-dn-Apopugins-460783612>
- <https://www.deviantart.com/dark-beam/journal/misc-772683097>

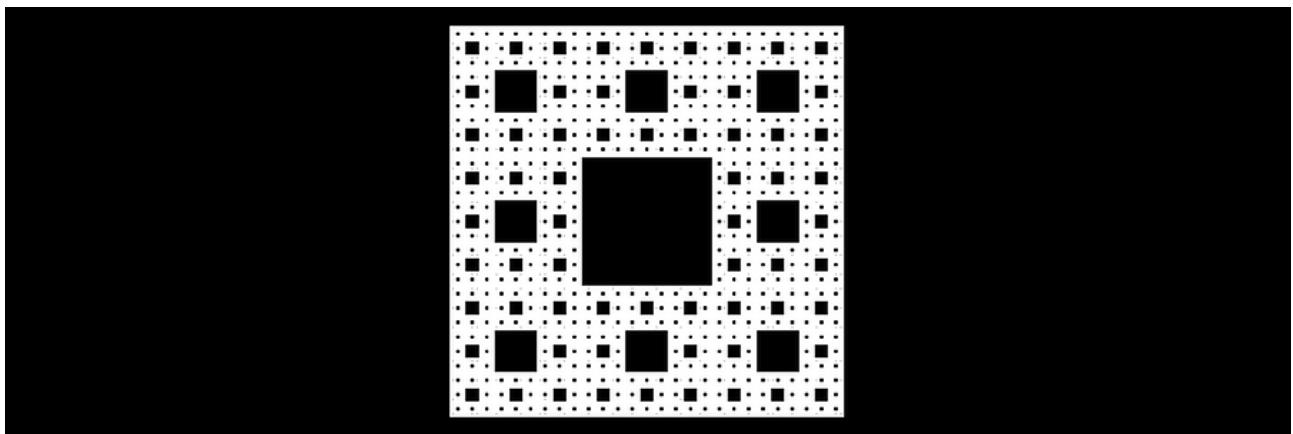
One of the possibilities that the jac_cn plugin offers is to map a unit square to halfplane.

For the unit square, we will use a menger sponge (but there are other possibilities, such as cropped bipolar and modulus tile). Then, we will map it to halfplane using jacobi, and feed into a flux. Both halfplanes will be cropped and sent back to fill the middle square.

First, we make the menger:

1. Start with a blank flame
2. On transform 1, set **linear** to 0.333333
3. **Duplicate** this transform 7 times, so you have a total of 8 transforms
4. Position those transforms at $(-1, 0)$, $(-1, 1)$, $(0, 1)$, $(1, 1)$, $(1, 0)$, $(1, -1)$, $(0, -1)$, $(-1, -1)$

Here we go, a nice menger sponge:



If you had any troubles, you can download parameters here: <https://pastebin.com/neD3D1u7>

We will start by adding the crop to it, so we do not have ugly artifacts along the way.

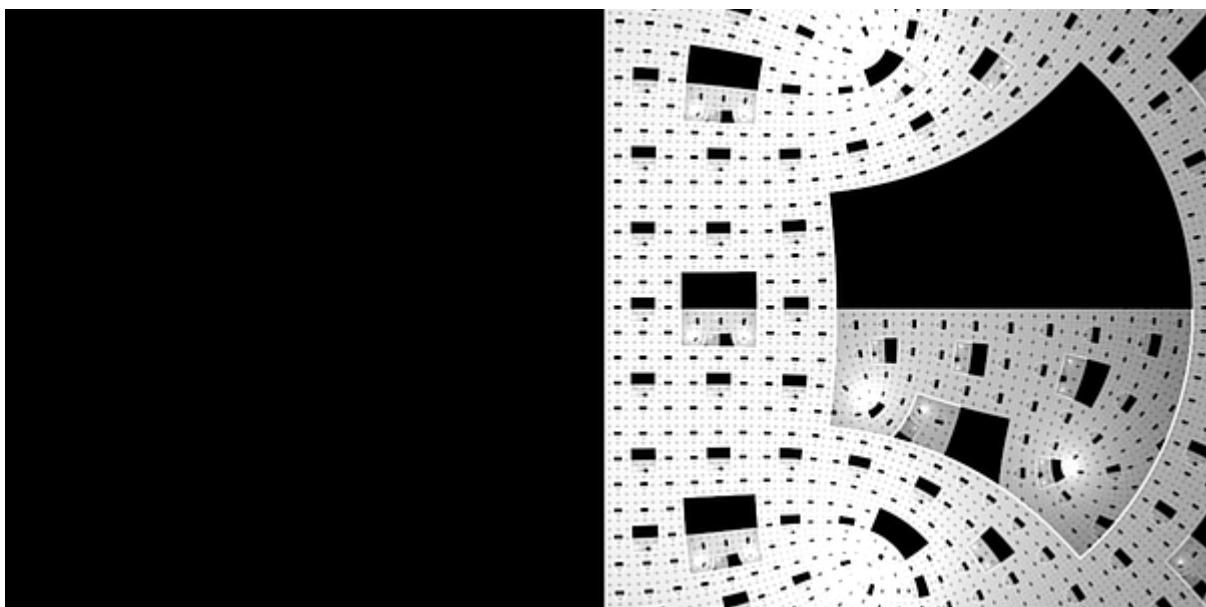
5. Add a new transform, add **post_smartcrop**
6. Set **static** to 1
7. Scale **post transform** down by 600%.
8. Set **color speed** to 1

What you should have at this point is a perfect square.

Time to use xaos and put it all into flux.

9. Set all **opacities** to 0, until there is just background visible
10. Add a **new transform**, set its color speed to 1
11. Replace linear with **jac_cn**
12. **Rotate** it by 45 degrees CW
13. **Move** it to -0.927, 0.927 (those are magic numbers)
14. **Scale** it up by 262.2 (another magic number)
15. Rotate the **post transform** by 45 degrees CW
16. On xaos tab, set the last transform to send points only **to** the previous transform, the one with smartcrop
17. Back on the smartcrop transform, set it to receive points only **from** jac_cn transform.

And we have a nice halfplane:



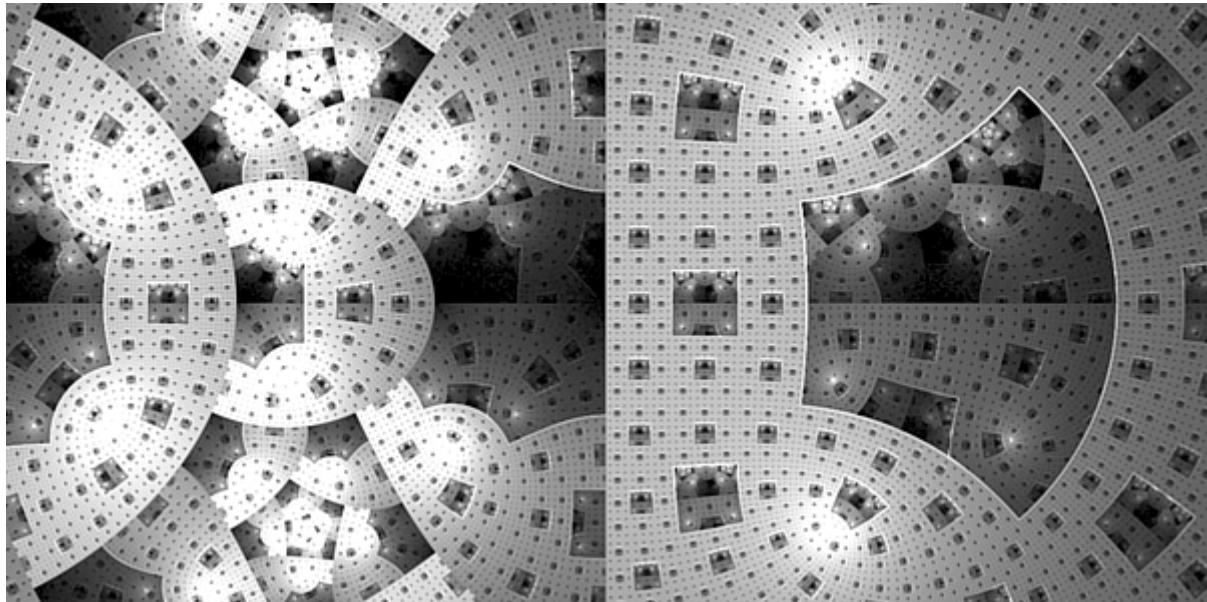
If you had any troubles, you can download parameters here: <https://pastebin.com/hkZB0UsC>

Finally, time to add flux.

18. Add a new transform
19. Go to the xaos tab and switch to "to" xaos. This transform should only send points **to** itself and the smartcrop transform
20. Switch to "from" xaos. This transform should receive points only **from** itself and jacobi transform
21. Set **color speed** to 1
22. Replace linear with **flux**
23. **Flip** the post transform
24. Increase its weight a bit (to something like 1-2)

It is ready:

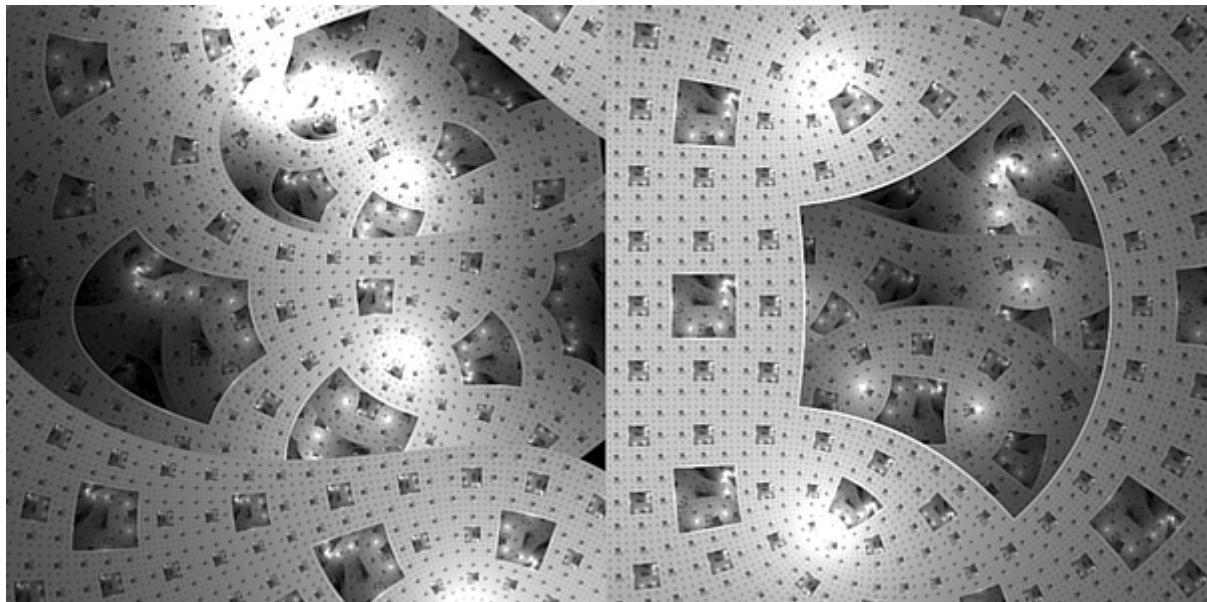
If you had any troubles, you can download parameters here: <https://pastebin.com/9J0iWKBV>



Tweaking tips:

- Move, scale and rotate foci
- On smartcrop transform, replace linear with spherical or curl
- Move, scale and rotate the smartcrop transform

With a little bit of tweaking, you can get something like this:



7 Tiled Strips

Another way to achieve a half plane is to take a vertical strip, and tile it horizontally in one direction. Like this:



Attention: this part of the tutorial may require some previous knowledge of xaos / weights.
Also, it requires some experience with certain frameworks, referenced in the next section.

7.1 Quick Review of Strips

In this section, we will quickly review some of the common vertical strips. I will not go

Elliptic transform from elliptic splits:

- <https://www.deviantart.com/tatasz/journal/Common-Frameworks-2-Elliptic-Splits-519706586>
- <https://www.deviantart.com/satania/art/Tutorial-Splits-elliptic-150422119>

Tiled bipolar:

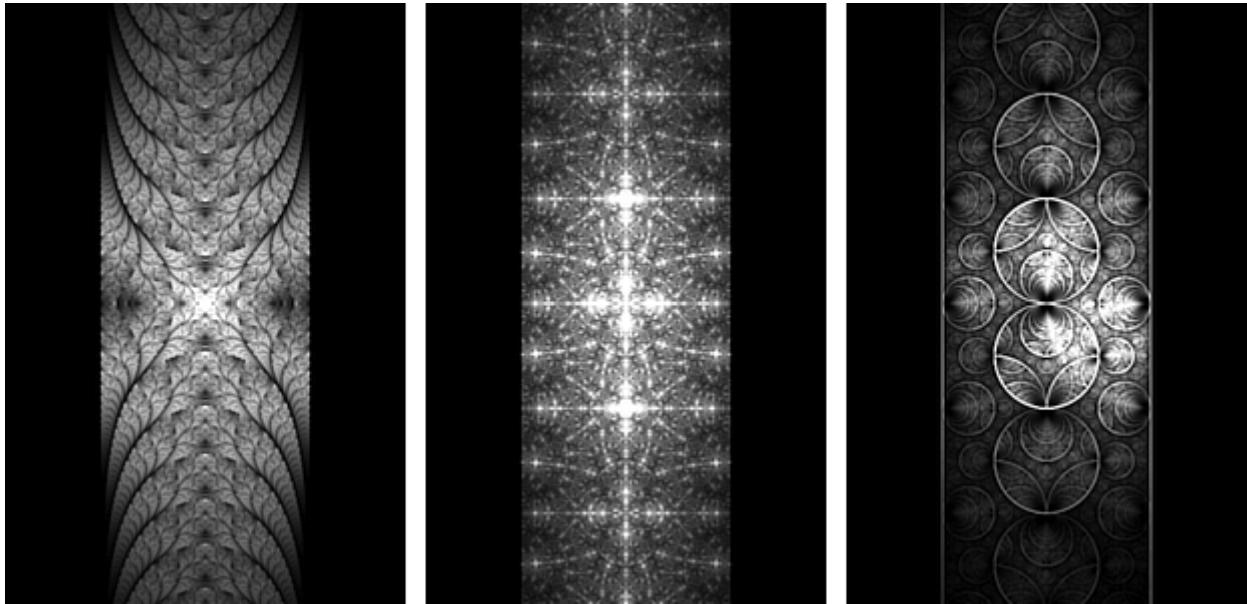
- <https://www.deviantart.com/satania/art/Tutorial-Bipolar-Shapes-213391274>
- <https://www.deviantart.com/fractaldesire/art/Tutorial-Making-of-Bipolar-design-276096394>

Spherical gasket:

- <https://www.deviantart.com/fiery-fire/art/Apo-Tuto-Sphericals-146647576>

Those are not the only possibilities, of course. For example, there is a whole gamma of tiled square objects. The setup for those is similar to that used for bipolar: you take a square and tile it into a vertical stripes using either linears, or plugins from the tile family.

Below, a preview of vertical stripes generated by elliptic, tiled bipolar and by the spherical gasket:



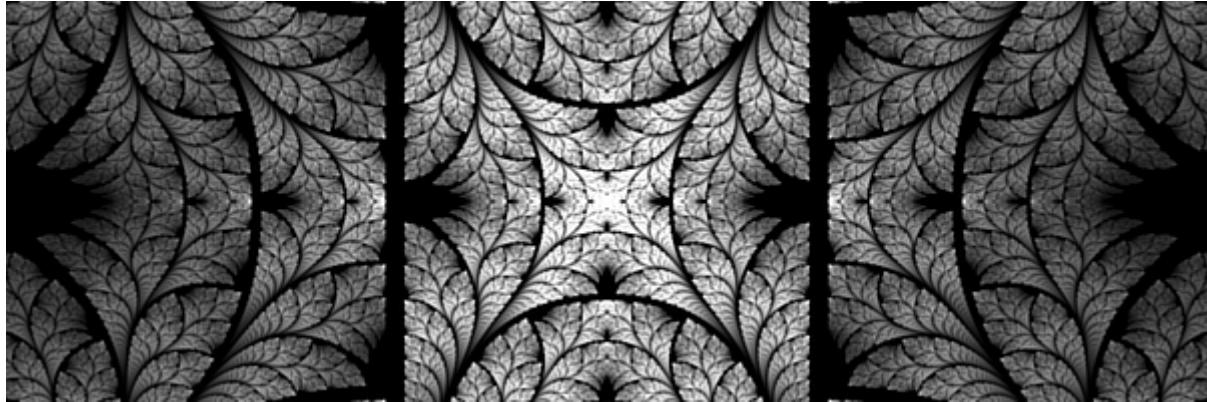
7.2 Making a tiled strip halfplane and filling it

Here, we will go through a simple example: elliptic splits tiled to halfplane, and then the other half plane filled with flux.

We start with making a basic elliptic splits setup:

1. Start with a blank flame
2. On transform 1, replace linear with **elliptic**. Rotate the transform 90 degrees CCW
3. Add a new transform, replace linear with **splits**. Set splits_x to 1.1
4. Scale up transform 2 by 200%
5. Add a new transform, replace linear with cylinder = 0.05 and pre_blur = 10
6. Now, switch to the transform tab, and, for the post transform, change Y from 1 to 10
7. Reduce the weight of transform 3 to something between 0.01 – 0.05
8. Duplicate transform 3
9. On transform 3, move the **post transform** 1.05 units right
10. On transform 4, move the **post transform** 1.05 units left

Your fractal should look like this after step 4:



And like this, after step 10:

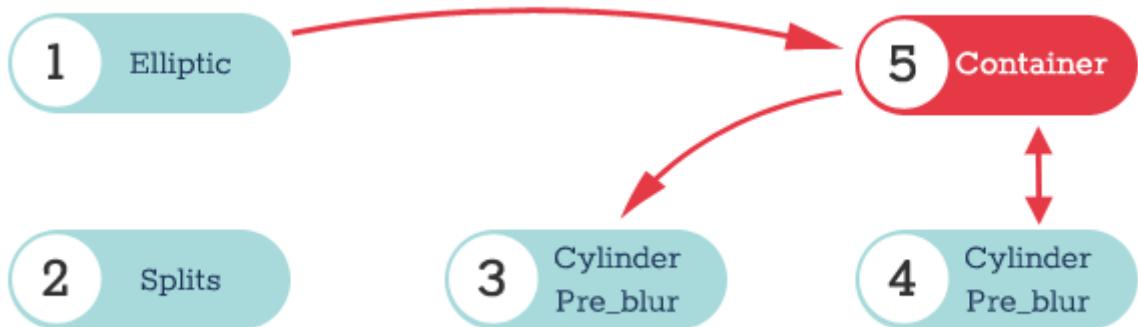


If you had any troubles, you can download parameters here: <https://pastebin.com/FE00PrPc>

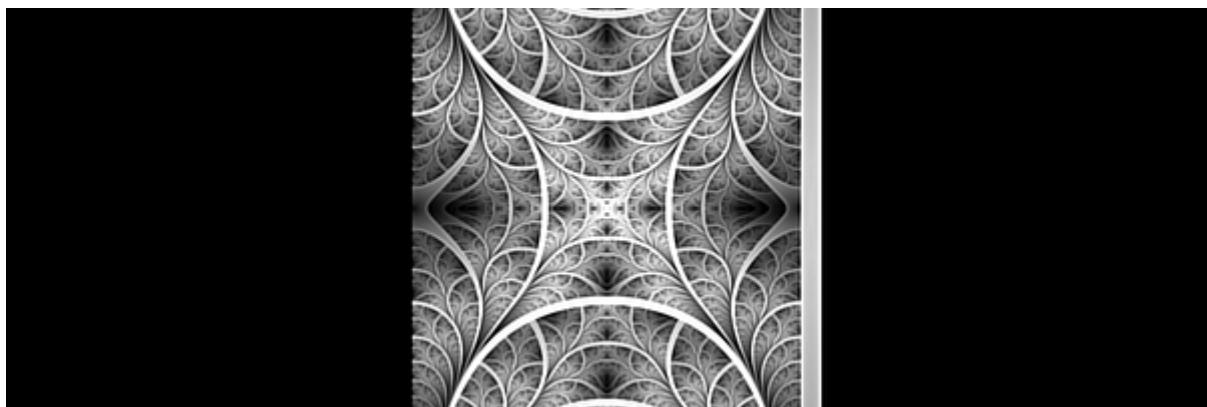
Next, we need to put the elliptic into a container to later tile it.

11. Start with setting the **opacity** of all the transforms to 0
12. Now, add a new transform, and set its color **speed to 1**
13. Go to the xaos tab and switch to "to" xaos. This transform should only send points **to** transforms with blur (transforms 3 and 4)
14. Switch to "from" xaos. This transform should receive points only **from** elliptic and one of the cylinders
15. Once you set everything up, play with weight of this transform to get a better look.

The xaos setup (steps 13 and 14) for the container should look like this:



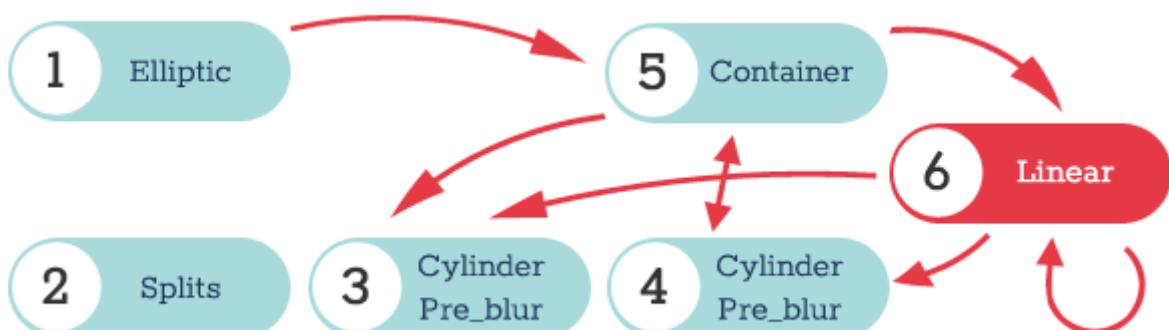
At this point, your fractal should look like this:



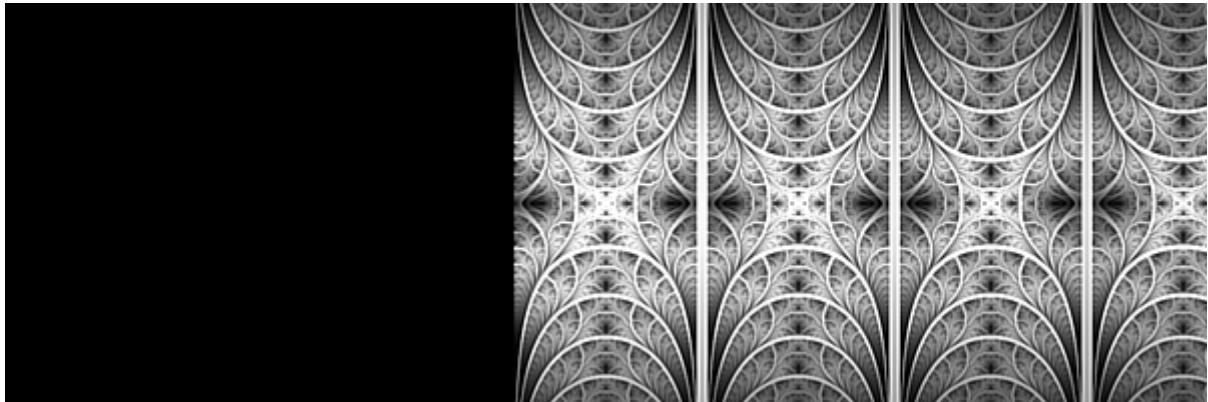
Now, we need to tile it.

16. First, we find the width of the tile. It is the width of the elliptic (2 units) plus the width of the cylinder (0.1 unit).
17. Now, add a new transform. Switch to the xaos tab. This transform should only send points **to** transforms with blur (transforms 3 and 4) and to itself.
18. Switch to "from" xaos. This transform should receive points only **from** the container (transform 5) and itself
19. Move the transform 2.1 unit to the right

This is the xaos setup for steps 17 and 18:



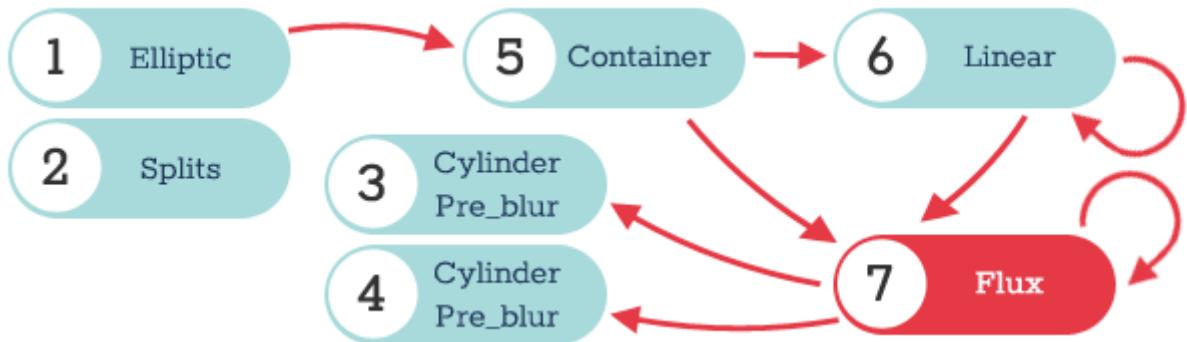
And, finally, we have a nice tiled halfplane



We got now to the very last part, where we fill the empty halfplane with something.

20. Add a new transform (duh), and replace linear with **flux**
21. Switch to the xaos tab. This transform should only send points **to** transforms with blur (transforms 3 and 4) and to itself.
22. Switch to "from" xaos. This transform should receive points only **from** the container (transform 5), the linear (transform 6) and itself
23. If you want to be extra neat, switch to transforms 3 and 4 (blurred cylinders), and set them to receive points only from the flux transform. This will improve the quality of the final render.

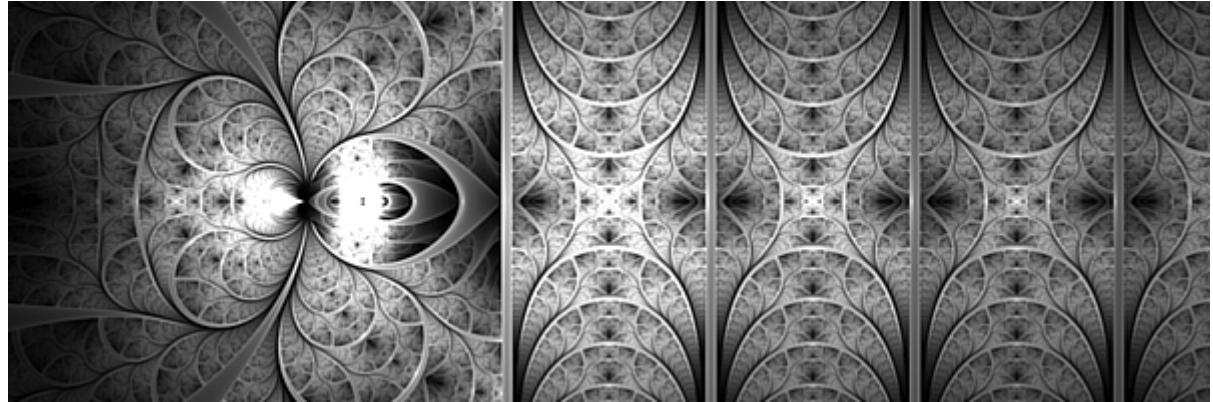
This is the xaos setup for steps 21, 22 and 23 (ouch getting complicated):



Back to work:

24. Select the **post transform** of the transform 7 and flip it horizontally
25. Our tile is not centered, so we will need to move the flux left a bit to avoid overlaps. Still with post transform, move the flux transform left by 1.05 or 1.1 (this depends on which cylinder you included into the tiled part) – if you have troubles, try both values and see which produces no overlaps and no empty spaces

And we are done!



If you had any troubles, you can download parameters here: <https://pastebin.com/8YiP74fc>