

# Hypershift by tatasz on DeviantArt

This tutorial is part of [Structured IFS tutorial collection](#) and requires some previous hypertile knowledge:

[“Structured IFS tutorial collection”](#) 🙄🙄 well, as usual, i need your help with this bunch of tutorials.

[What should i write about? Is there anything you wanna know? Please ask, i'll write about it ^^](#)

[If you wrote / want to write cool tutorials about structured IFS, poke me, i'll add it to the collection.](#) I need feedback 🙄 [Is anything too crappily written? Is it messy / unclear? Do tell!](#)

## Update Log:

**2016 / 04 / 06 - Advanced Linear Tilings added**

[2016 / 03 / 20 - 2 new tutorials: Glynnsim and more on hypertiles.](#)

[2016 / 03 / 08 - 2 new Tutorials, and a bit of organization](#) 🙄

[2016 / 02 / 23 - Second blur tutorial added](#) [2015 / 11 / 23 - 3 tutorials added to list](#) 🙄

## XAOS

### Xaos:

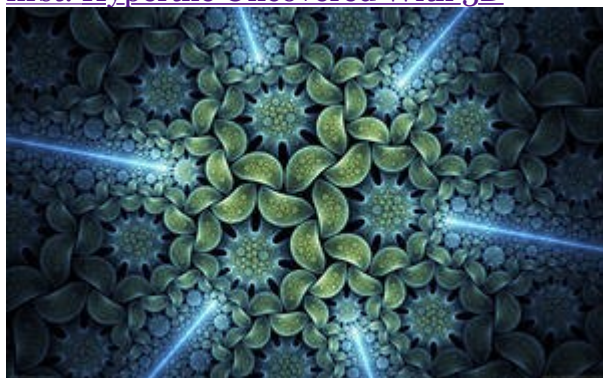
[Xaos Basics](#)

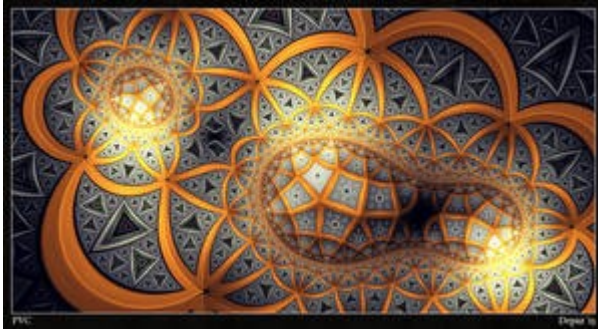
[Linked Transforms](#)

[Shared linked transforms](#) [Shared linked transforms - Examples”](#) 🖥️ **“Hypertile**

**Basics** First of all, special thanks to Zueuk who had the patience to explain me all the hypertile stuff 🙄

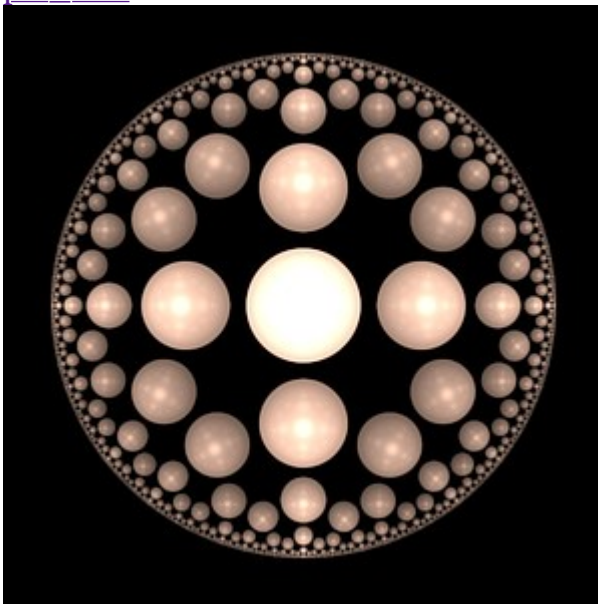
There is also an awesome hypertile tutorial that you should check out first: [Hypertile Uncovered With 3D](#)





### **Basics**

To make a basic hypertile, you will need 2 transforms: Basic hypertile  
hypertile2, rotated 180 degrees bubble (small amount, about 0.25 lets say) with  
pre\_blur



The exact bubble size to fill a hypertile can be calculated exactly, or you can just change the amount of bubble until it fits.

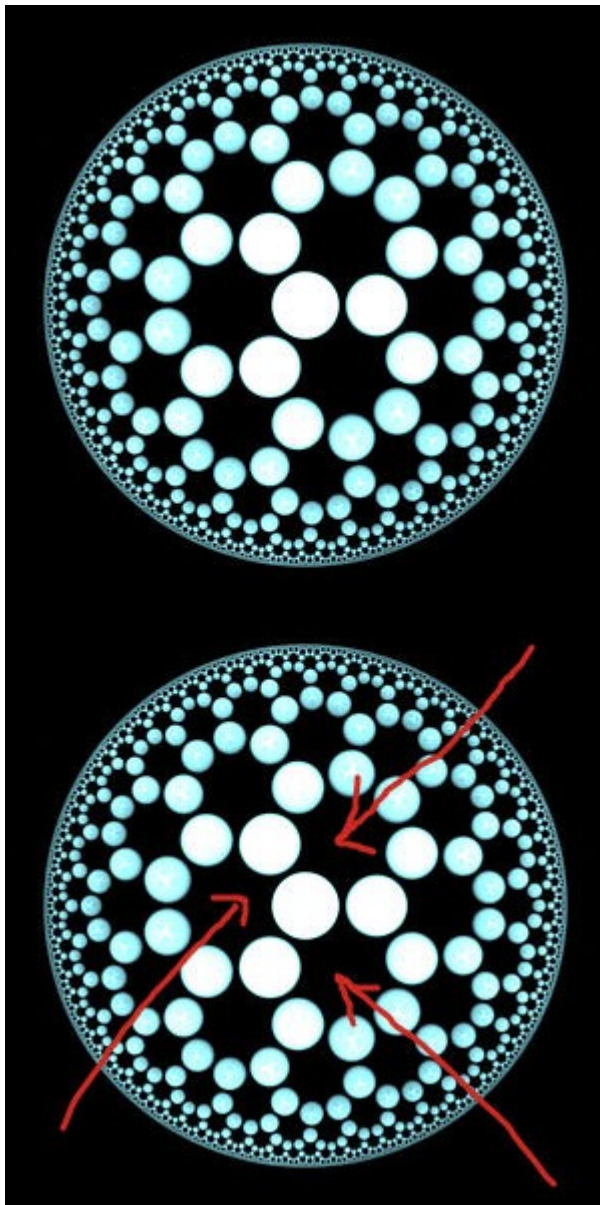
The hypertile has two parameters, p and q. Basically, this means it takes p-gons, with q polygons meeting at each vertex (this do"

This also requires an unreleased plugin by [zyOrg](#), called hypershift. As aposhackers have been spreading it from artist to artist, i just sneak out a link ^^ If [zyOrg](#) has anything against it, please lemme know, i'll take it down.

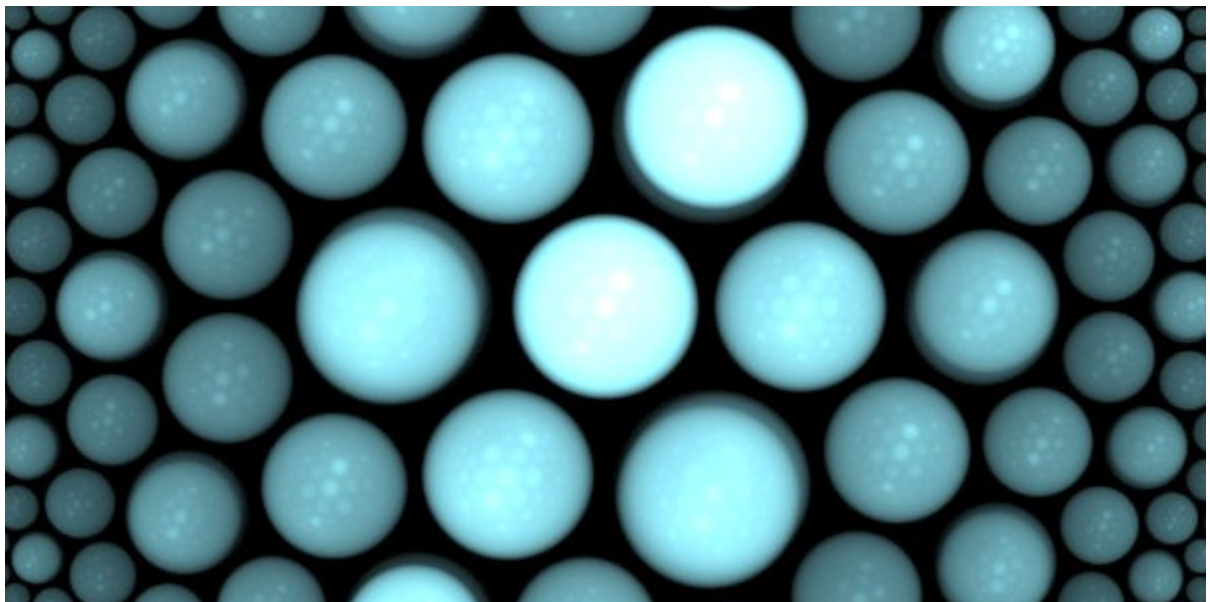
[dl.dropboxusercontent.com/u/12...](http://dl.dropboxusercontent.com/u/12...)

## **The problem**

Lets start with a basic hypertile,  $p = 3$  and  $q = 7$ , and fill it in with some blurry hemisphere:



We have those huge holes. They seem quite not easy to fill in. If you, for example, add a second hemisphere and move it around using a post transform, you still get some artifacts (notice that there is a shadow around the hemispheres that fill those holes).





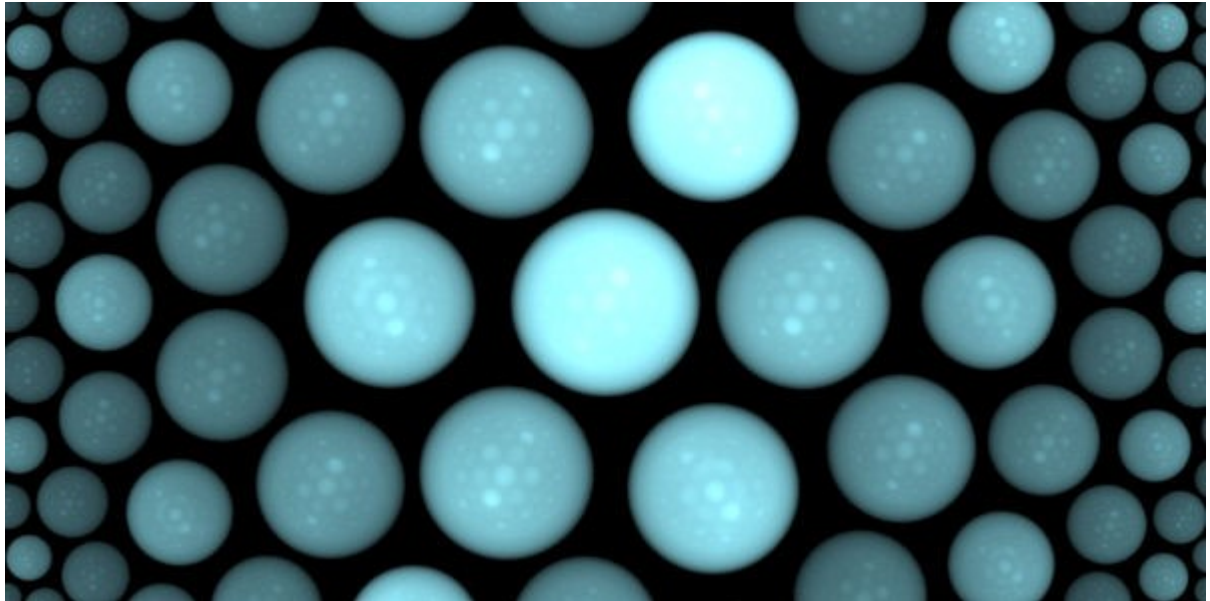
Those holes can be filled using poincare or hypershift. As hypershift works better with Chaotica, it is what i recommend.

## Hypershift magic

So, move the second hemi back to the origin, and lets do a magic trick:

1. Add to it a linked transform with hypershift 1
2. Set shift variable to 0.301
3. Rotate the post transform 60 degrees CCW

And here we have the result: no nasty artifacts and shadows.



The 0.301 value can be found by either doing some maths or just experimenting.

To find out the angle, do the following:

1. Remember that  $p = 3$
2. Take  $360 / 3 = 120$ . This is the angle between the holes we are filling.
3. As the horizontal axis goes in the middle of two holes, you only need to rotate half of it,  $120 / 2 = 60$  degrees.

For a more uniform result (above, you can see the filling hemispheres are uneven), try positioning one filler for each hole, so 3 transforms in this case, wit corresponding rotations ( $120 + 60$  CCW and  $2 * 120 + 60$  CCW, for example).

Some example params: [Hypershift Example Params](#), that look kinda like:

