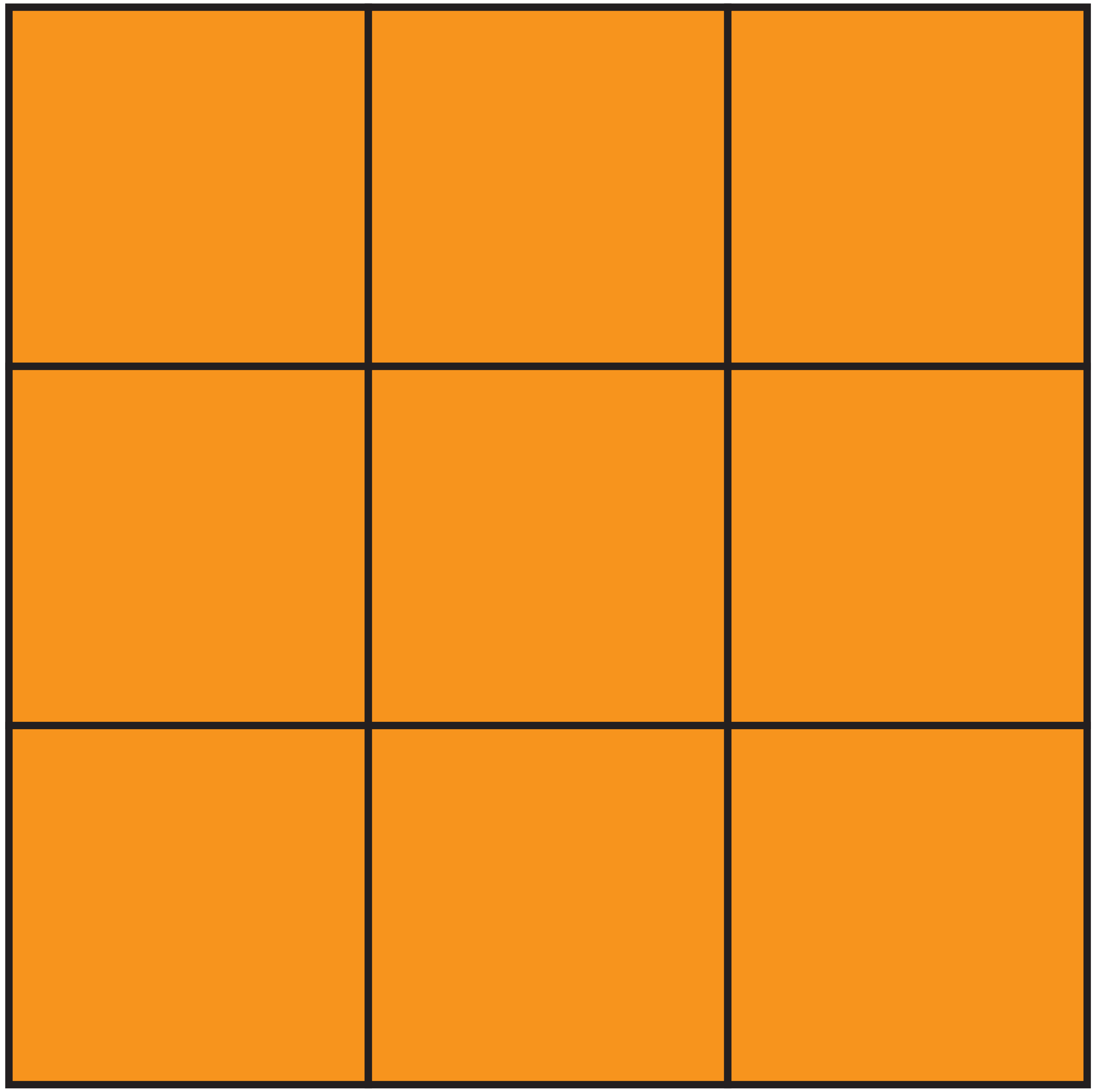
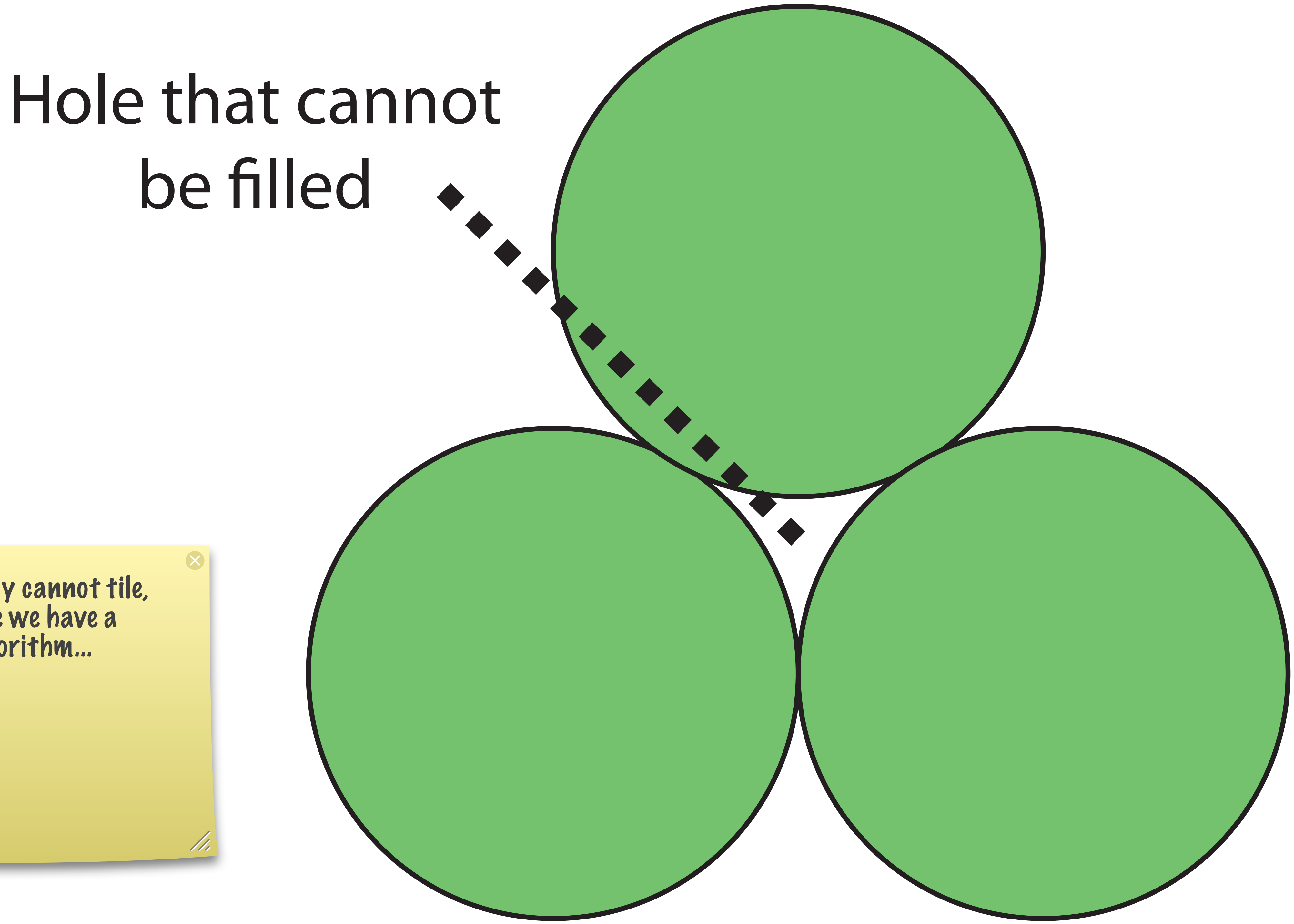
The Mysteries of Aperiodic Tiles

Edmund Harriss, University of Leicester
www.mathematicians.org.uk/eoh
maxwelldemon.com
@gelada on twitter



This can tile, periodically...

What about another, a circle...

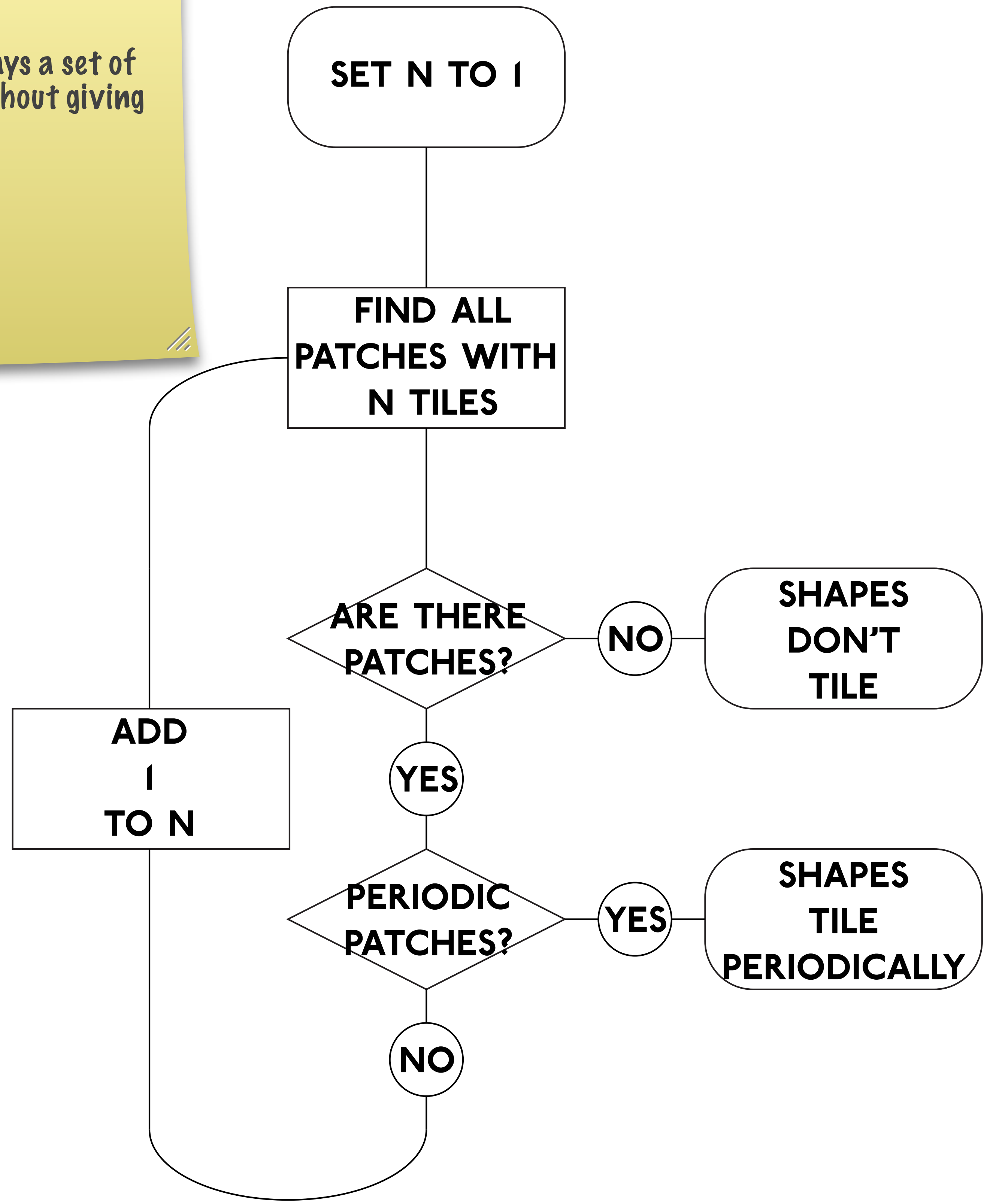


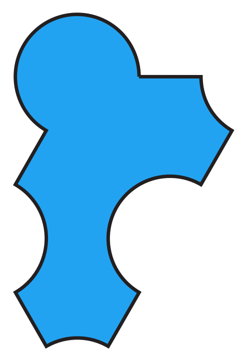
This clearly cannot tile, but maybe we have a simple algorithm...

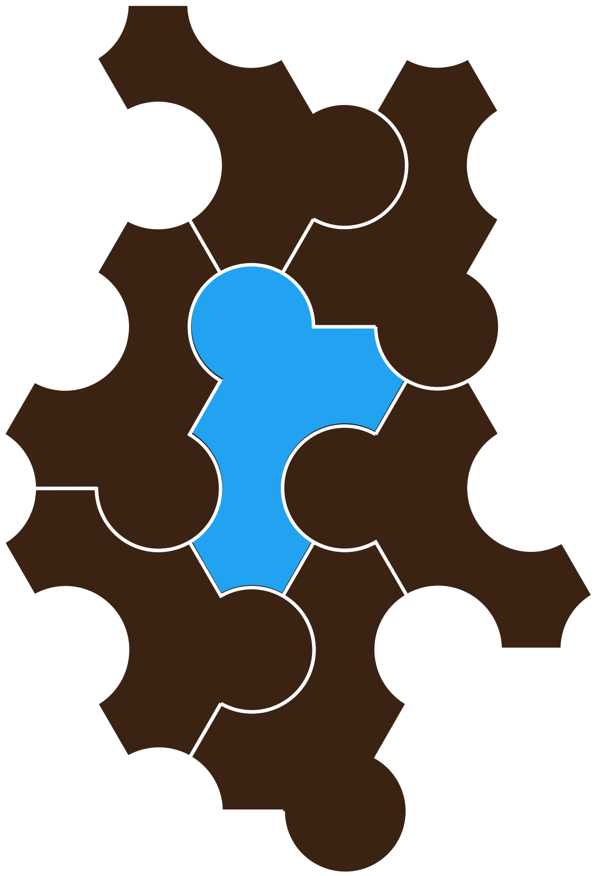
This will run around the loop forever.

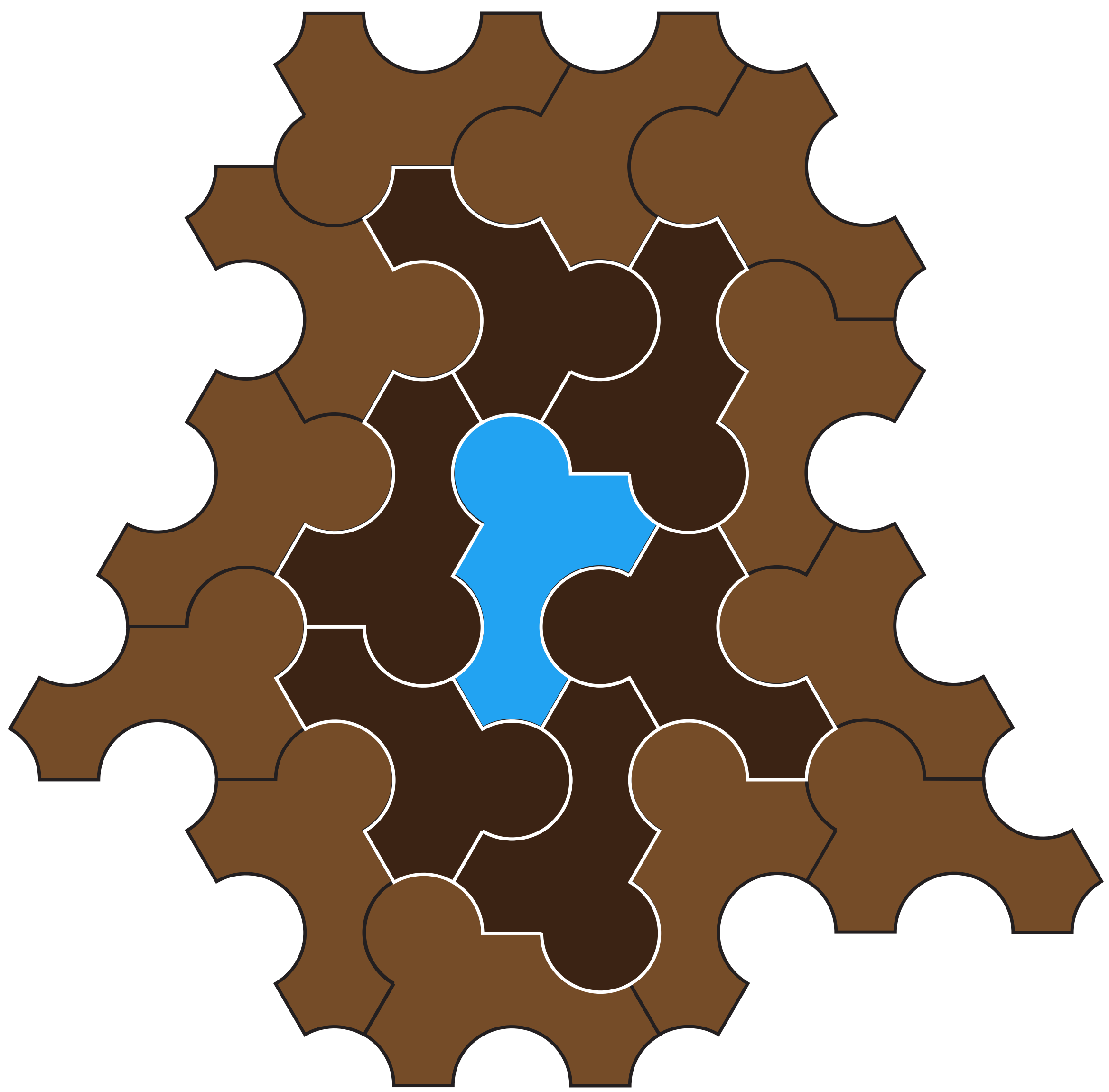
It gets worse. The question is undecidable

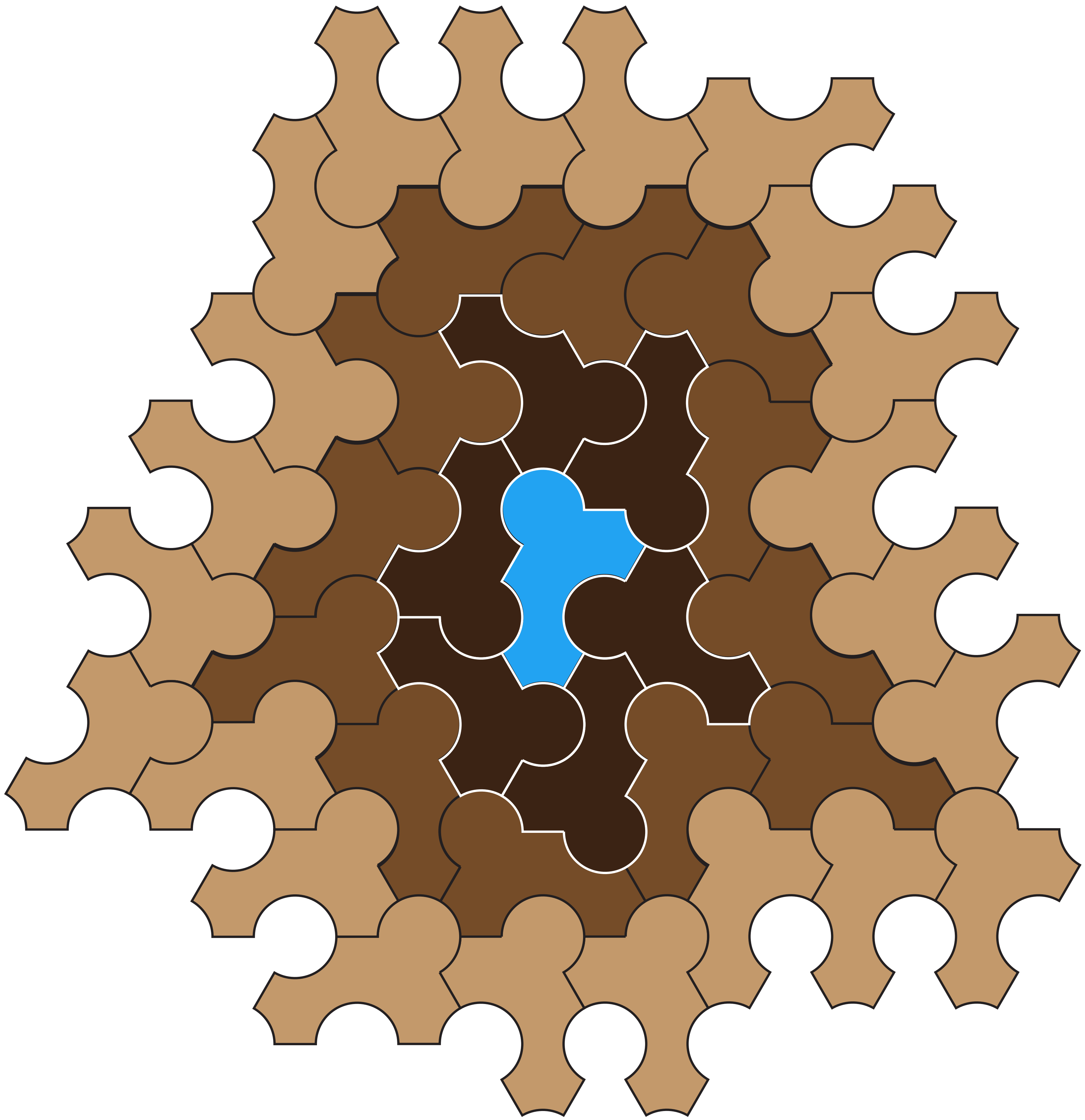
Whatever algorithm you find, there is always a set of shapes that will cause it to run forever without giving an answer...

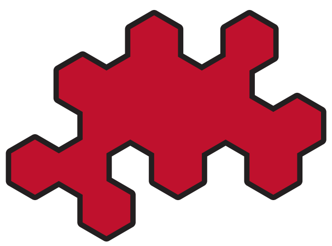


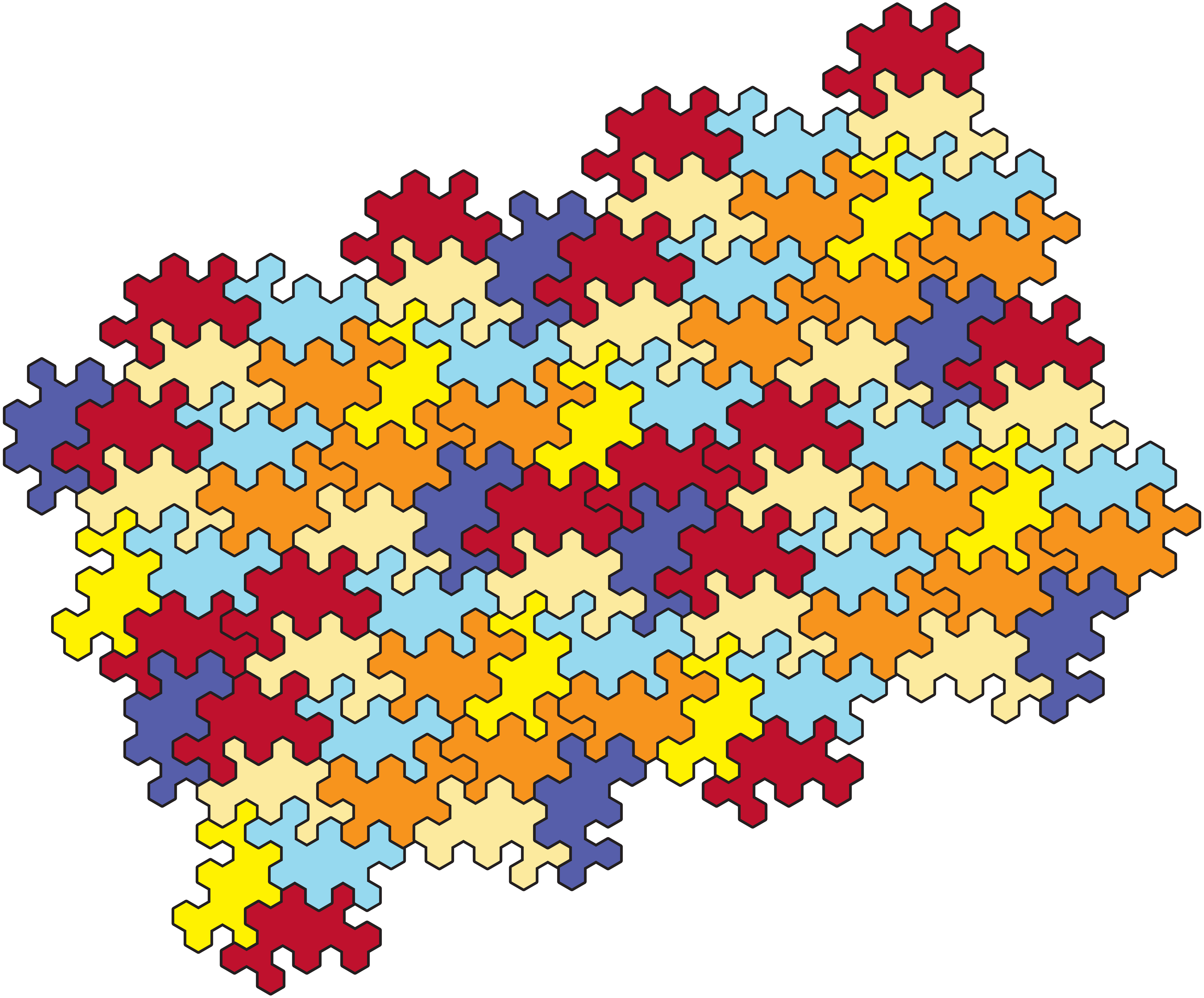


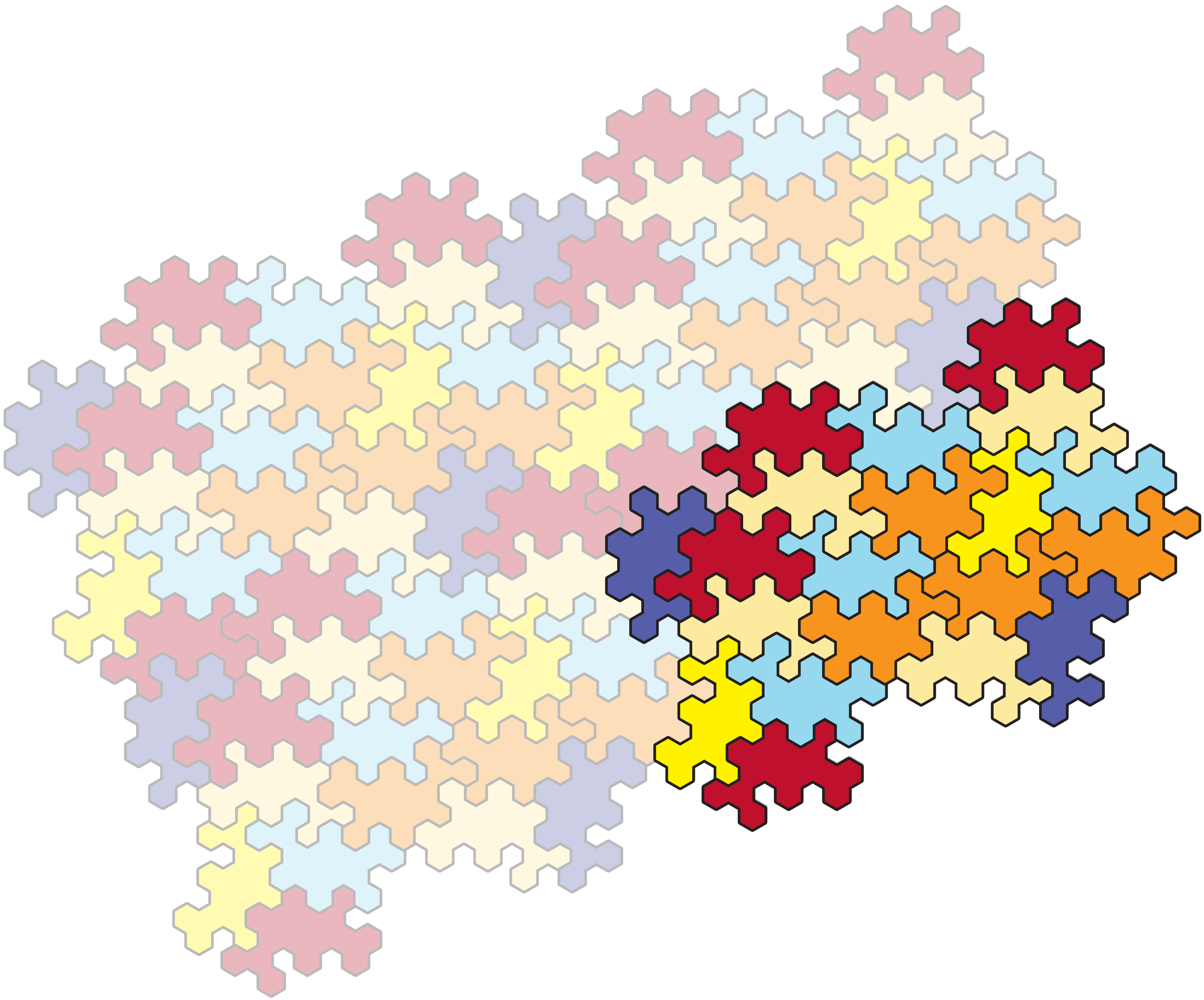


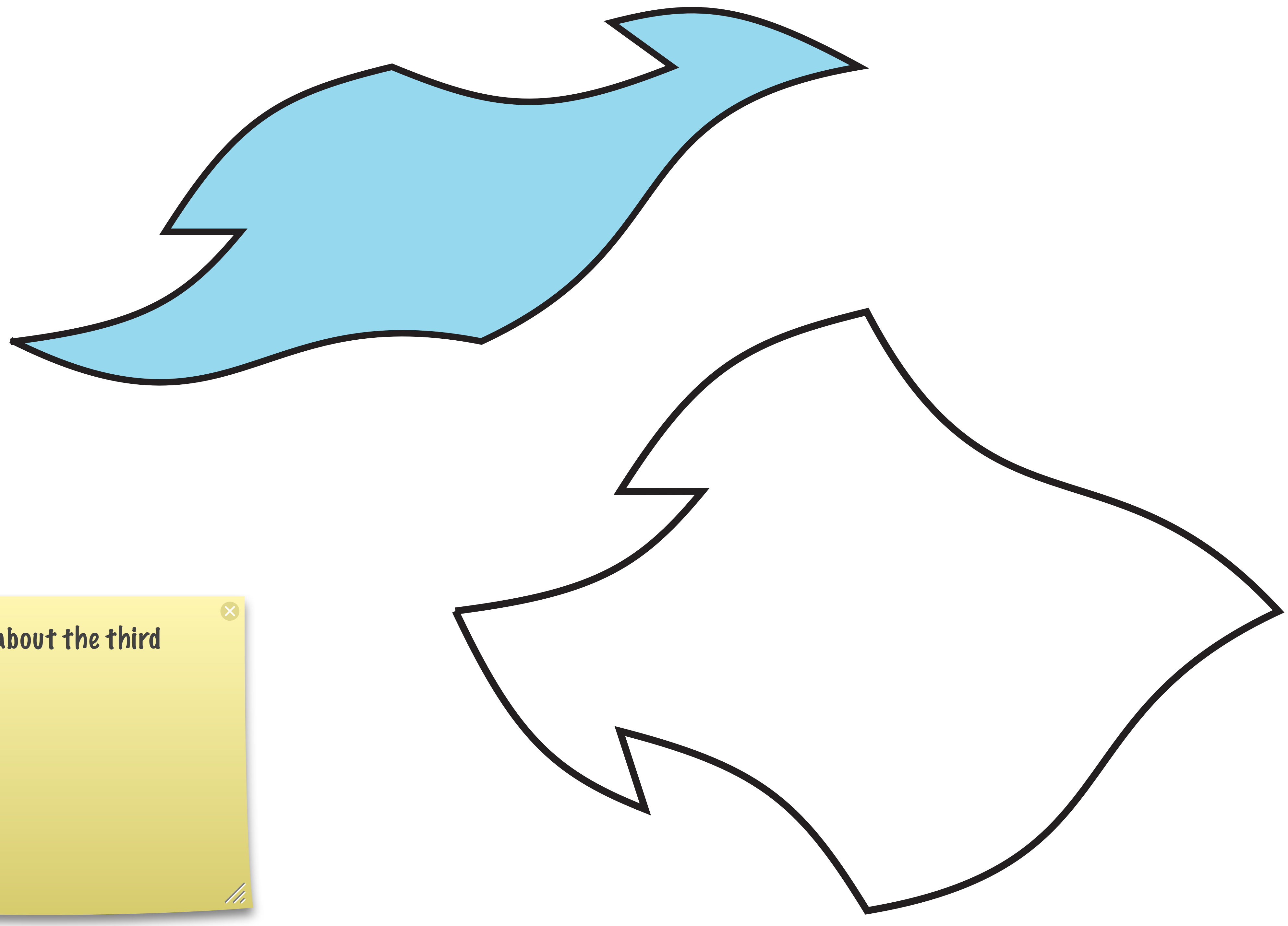






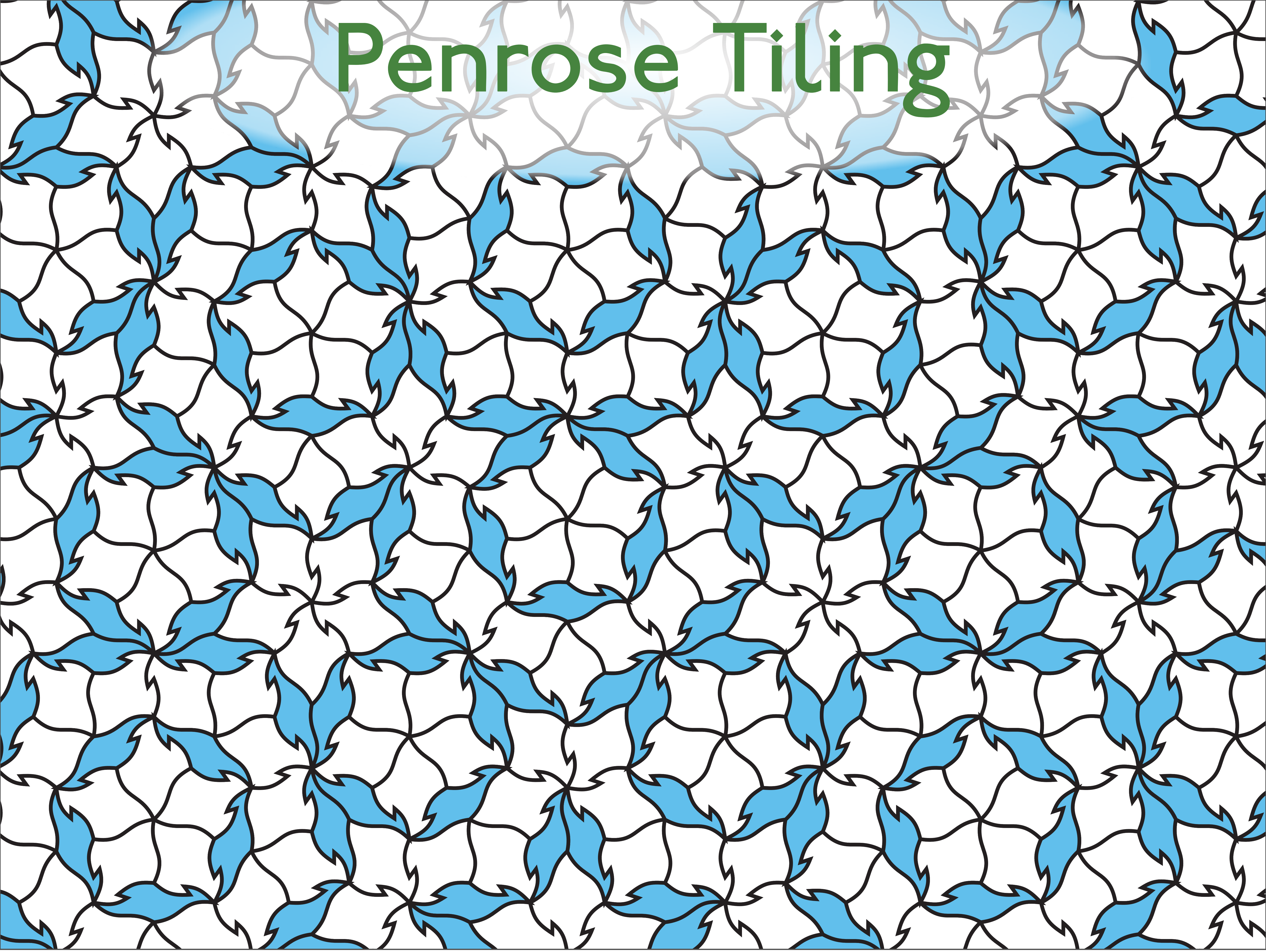


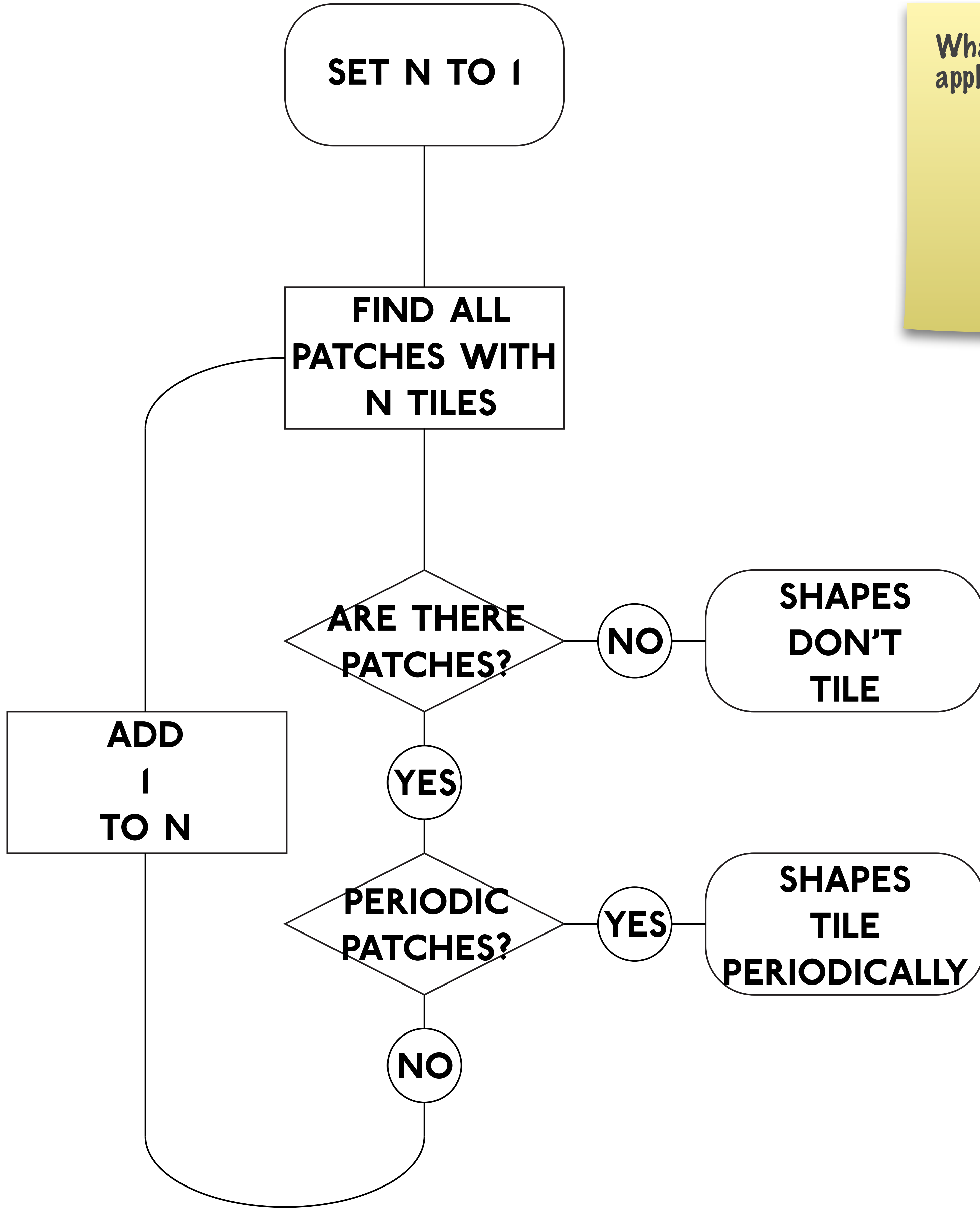




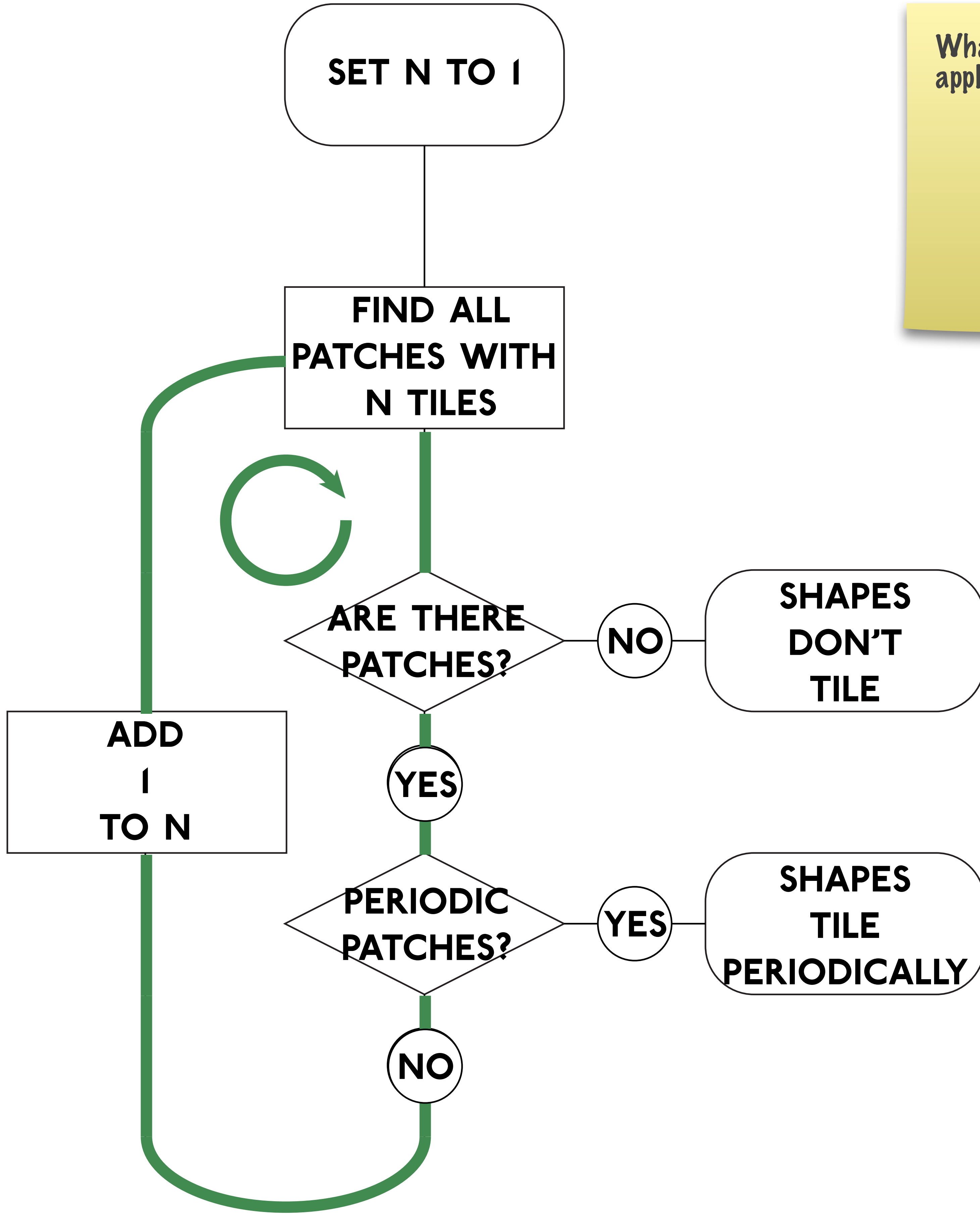
What about the third set...

Penrose Tiling





What happens when we apply this to our shapes...



What happens when we apply this to our shapes...

Berger PhD Thesis

Simpler Proof: Robinson

Current state of the art:
5 Tiles Ollinger

This is a reminder of how deep undeciability cuts into mathematics.

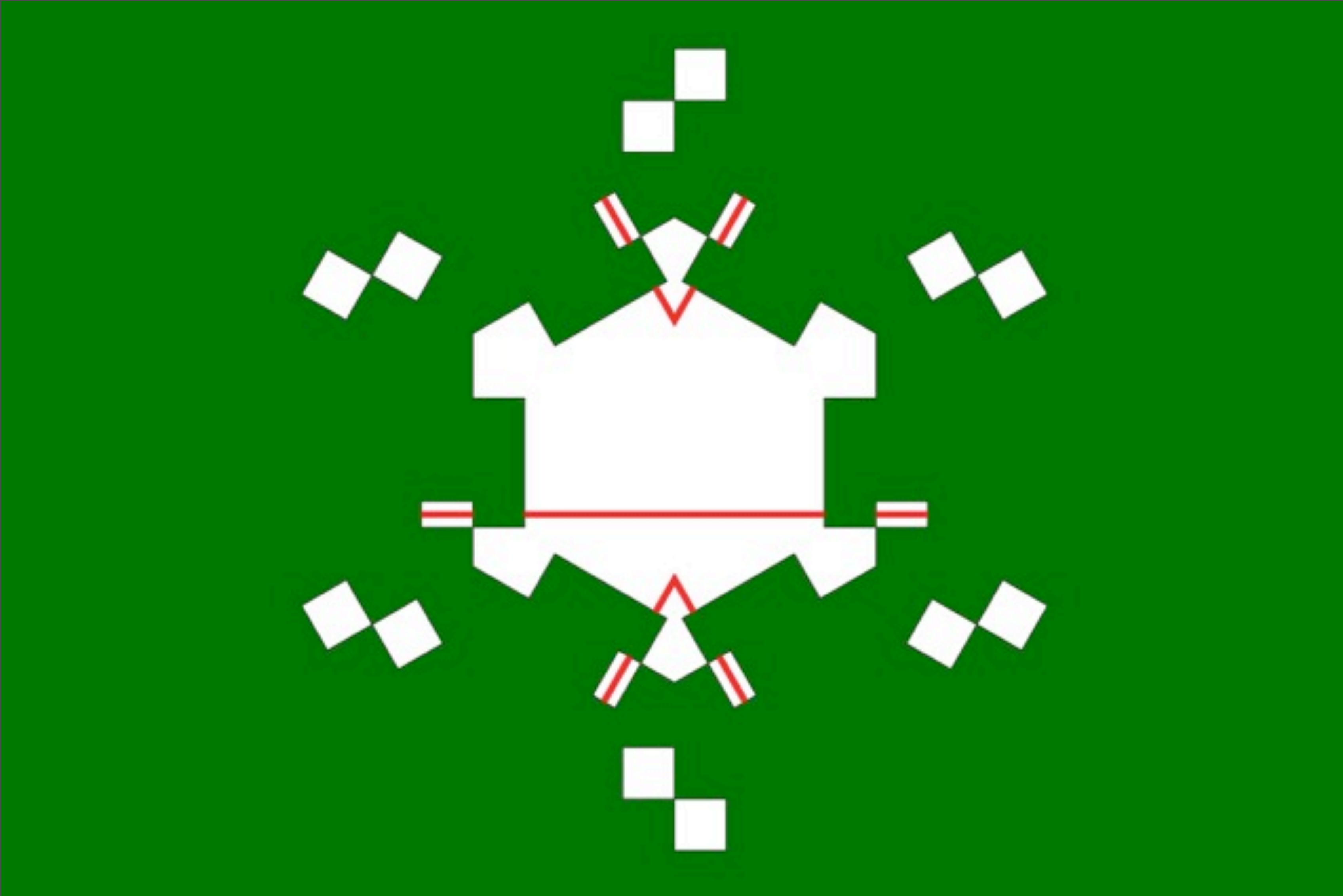
(of course it is also exciting: simple questions about tilings will always yield interesting new ideas:

Like Aperiodicity...

Robert Berger, *The undecidability of the domino probler*
Memoirs of the AMS 66, 1966

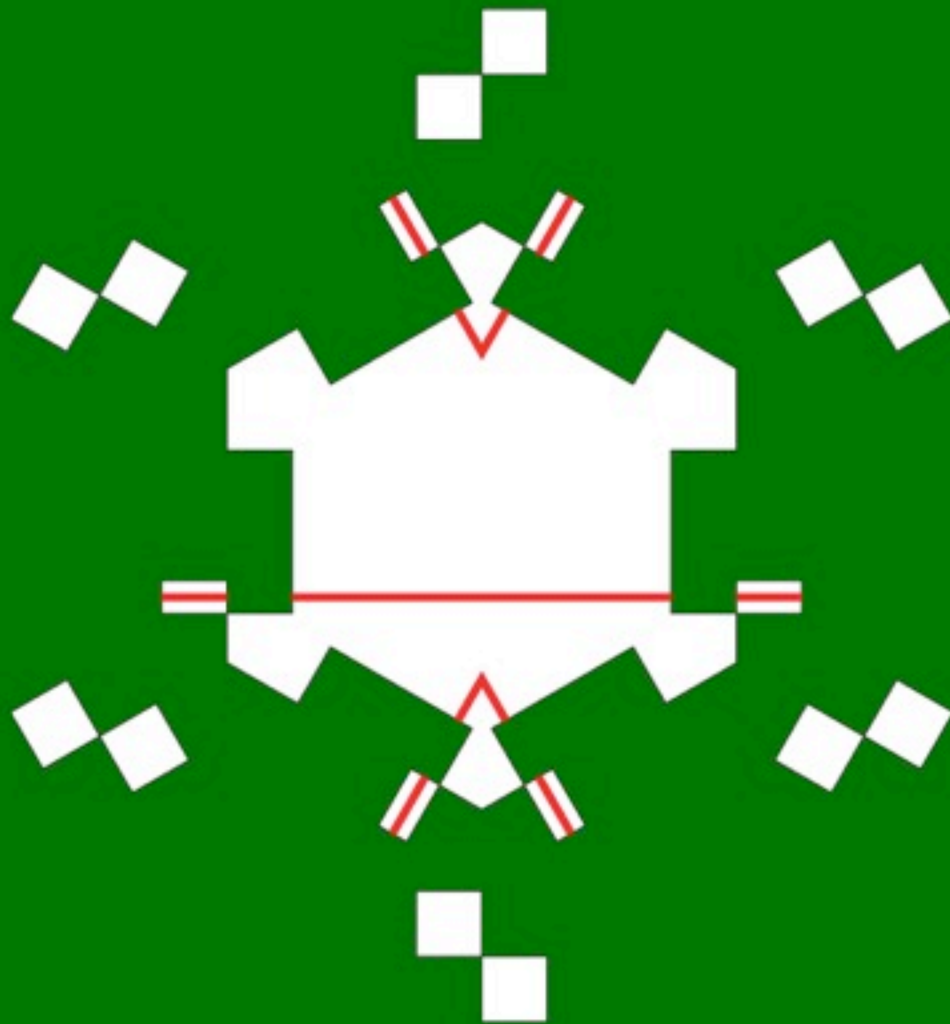
Raphael Robinson, *Undecidability and nonperiodicity fo*
Inventiones Mathematicae 12, 1971, pp. 177-209

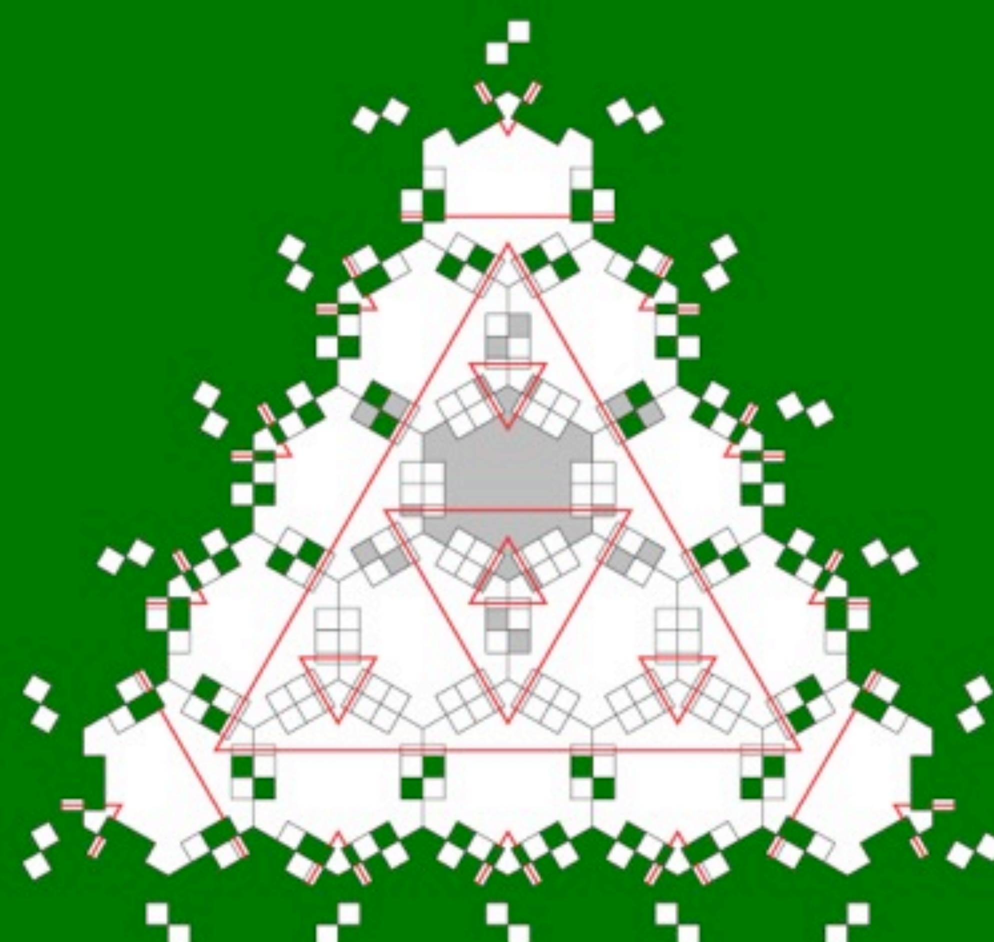
Nicolas Ollinger: *Tiling the Plane with a Fixed Number of Polyominoes*. Proceedings of LATA 2009, Lecture Notes in Computer Science 5457, Springer 2009, pp. 638-647.



Joshua Socolar and Joan Taylor,
An aperiodic hexagonal tile,
preprint: arXiv:1003.4279v1

Joan Taylor,
Aperiodicity of a Functional Monotile,
preprint:
www.math.uni-bielefeld.de/sfb701/preprints/sfb10015.pdf





Joshua Socolar and Joan Taylor,
An aperiodic hexagonal tile,
preprint: arXiv:1003.4279v1

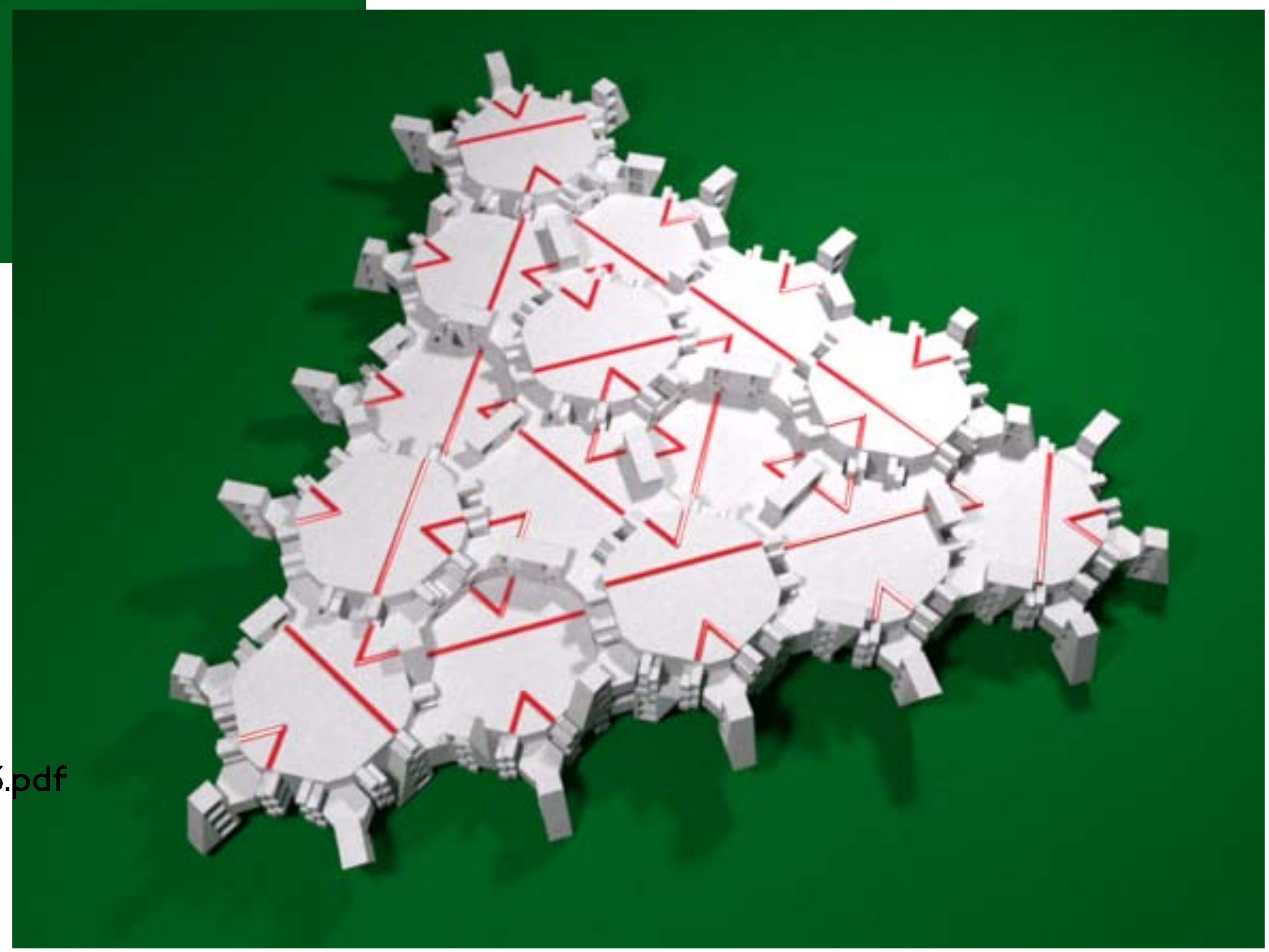
Joan Taylor,
Aperiodicity of a Functional Monotile,
preprint:
www.math.uni-bielefeld.de/sfb701/preprints/sfb10015.pdf

3d version, 1 periodic direction

Mentioned in New Scientist

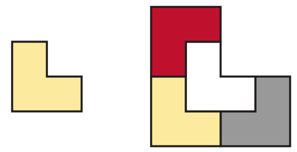
How can you tell if these shapes tile at all?

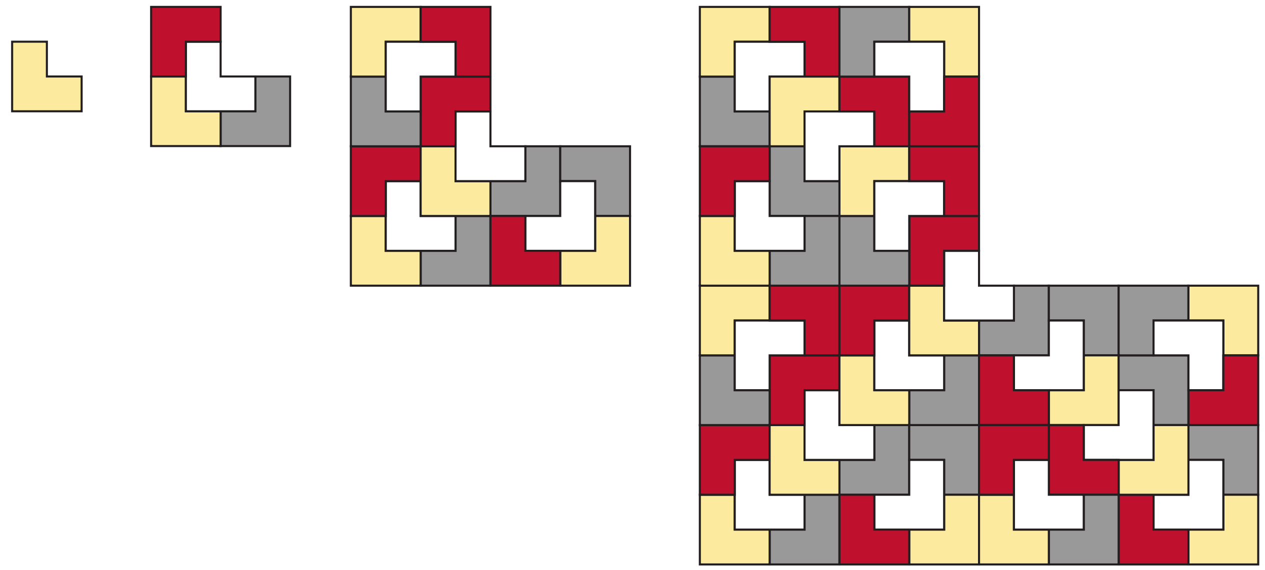
The answer is a substitution rule.

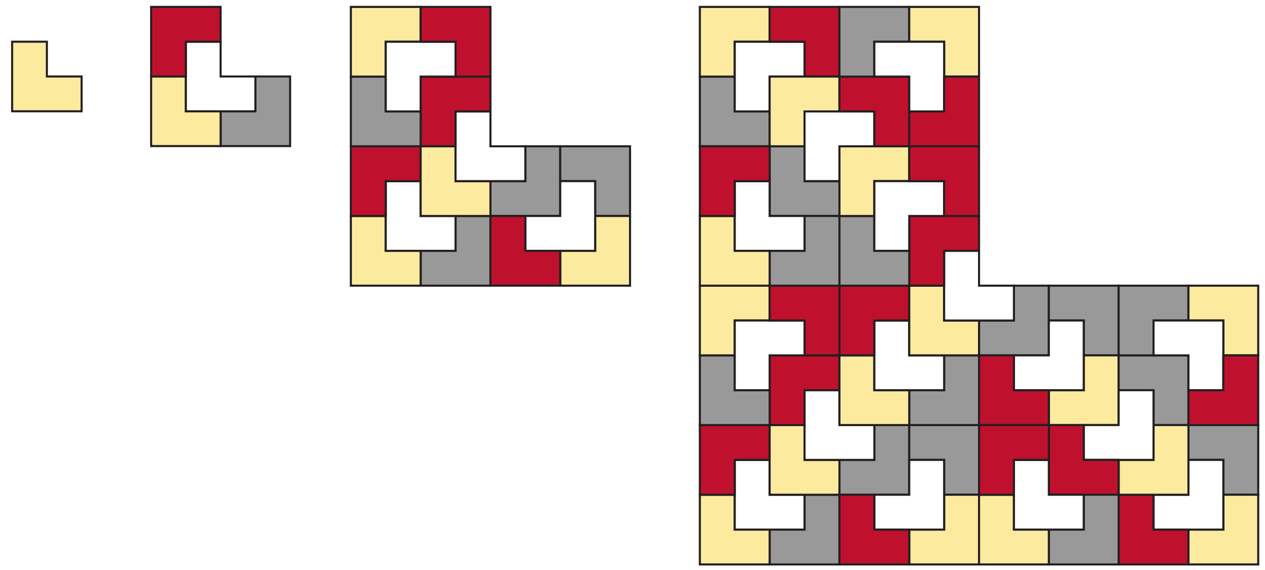


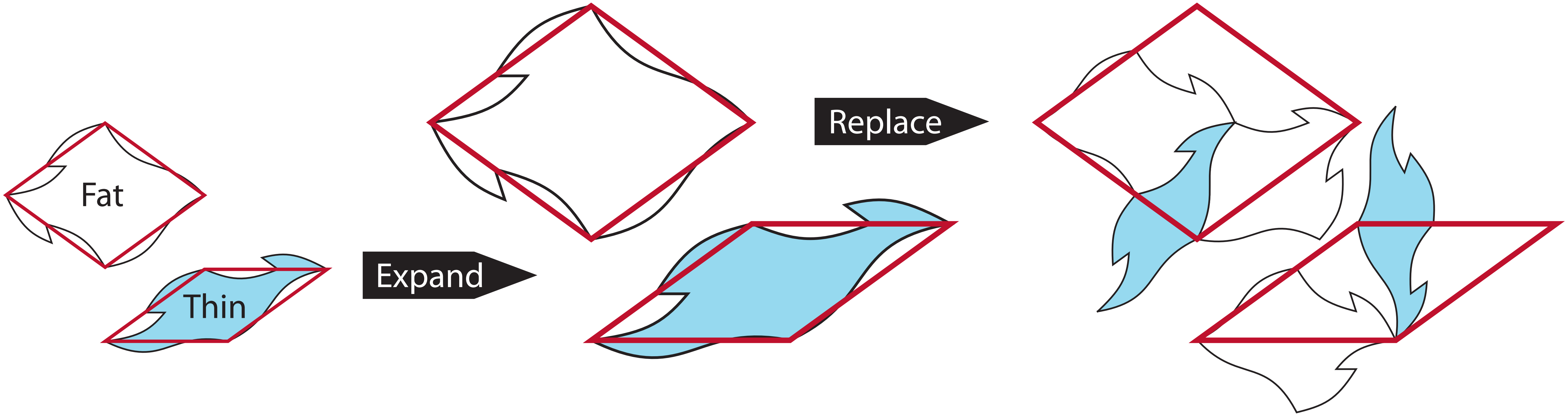
Joshua Socolar and Joan Taylor,
An aperiodic hexagonal tile,
preprint: arXiv:1003.4279v1

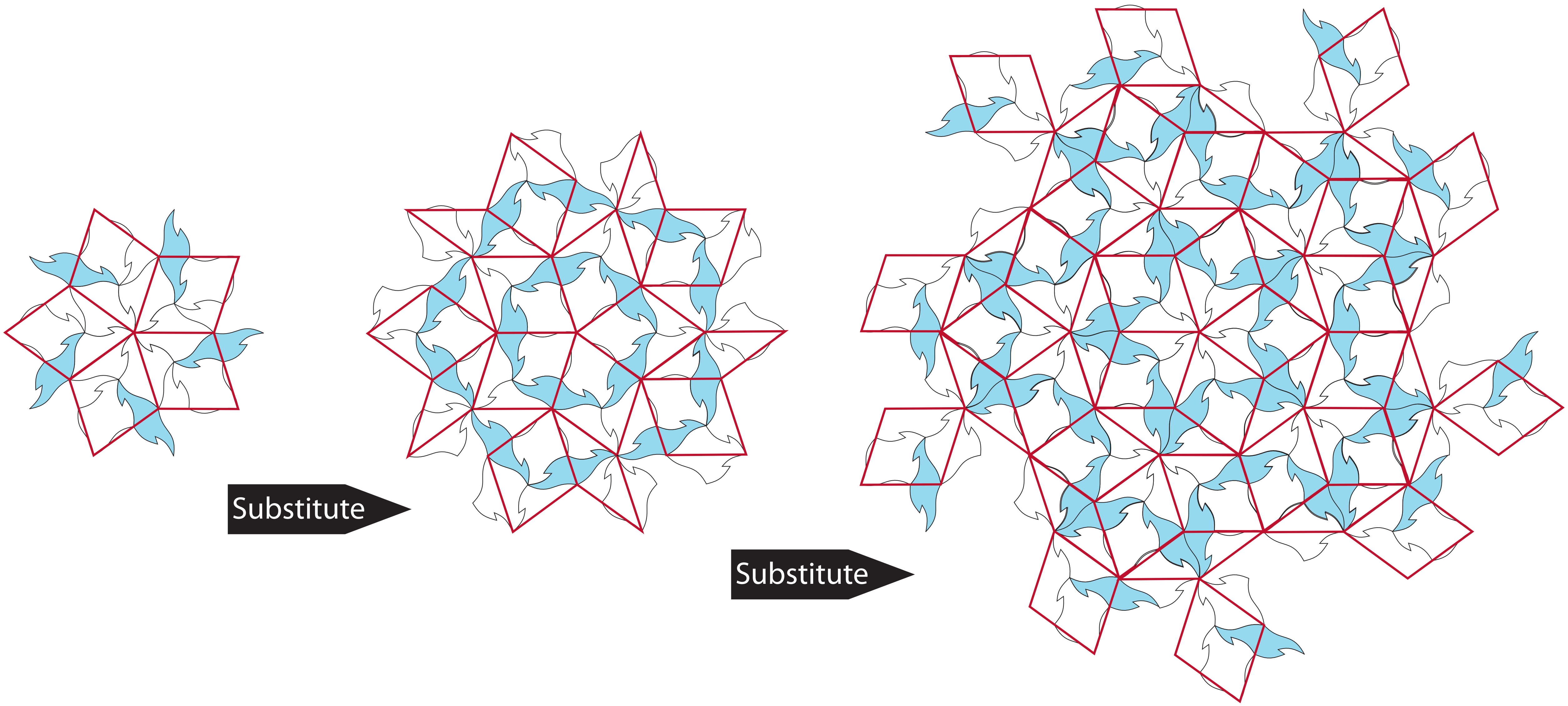
Joan Taylor,
Aperiodicity of a Functional Monotile,
preprint:
www.math.uni-bielefeld.de/sfb701/preprints/sfb10015.pdf

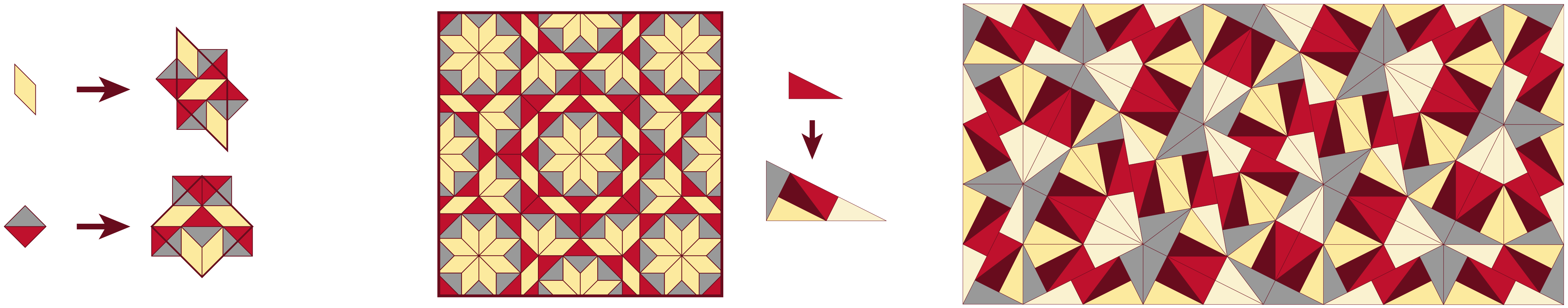


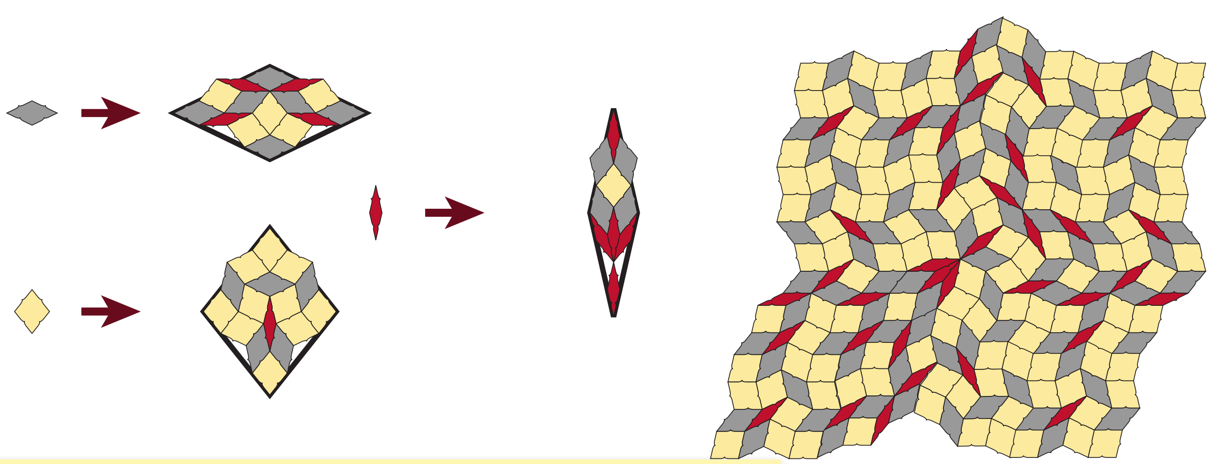


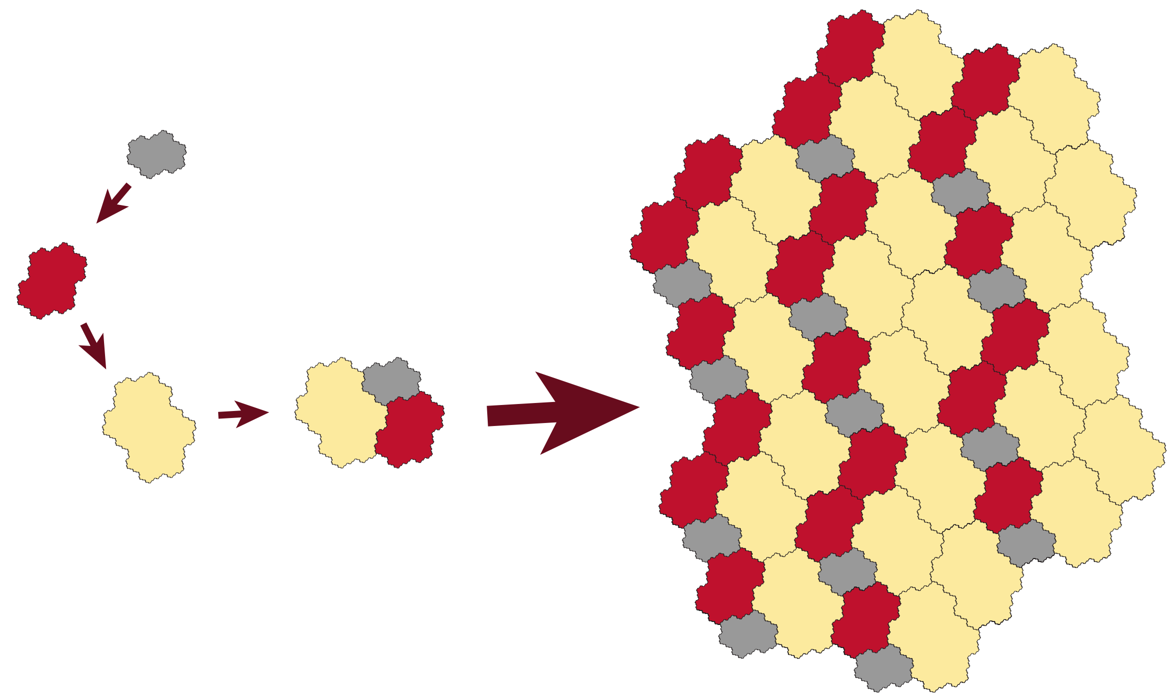



Fat
Thin
Expand
Replace


Substitute
Substitute







My work has been in creating and attempting to characterize substitution tilings.

The process of taking a substitution tiling

EG Penrose rhombs

and changing the edges to give a set of aperiodic tiles became known as matching rules...

Raphael Robinson, *Undecidability and nonperiodicity for tilings of the plane*, Inventiones Mathematicae 12, 1971, pp. 177-209

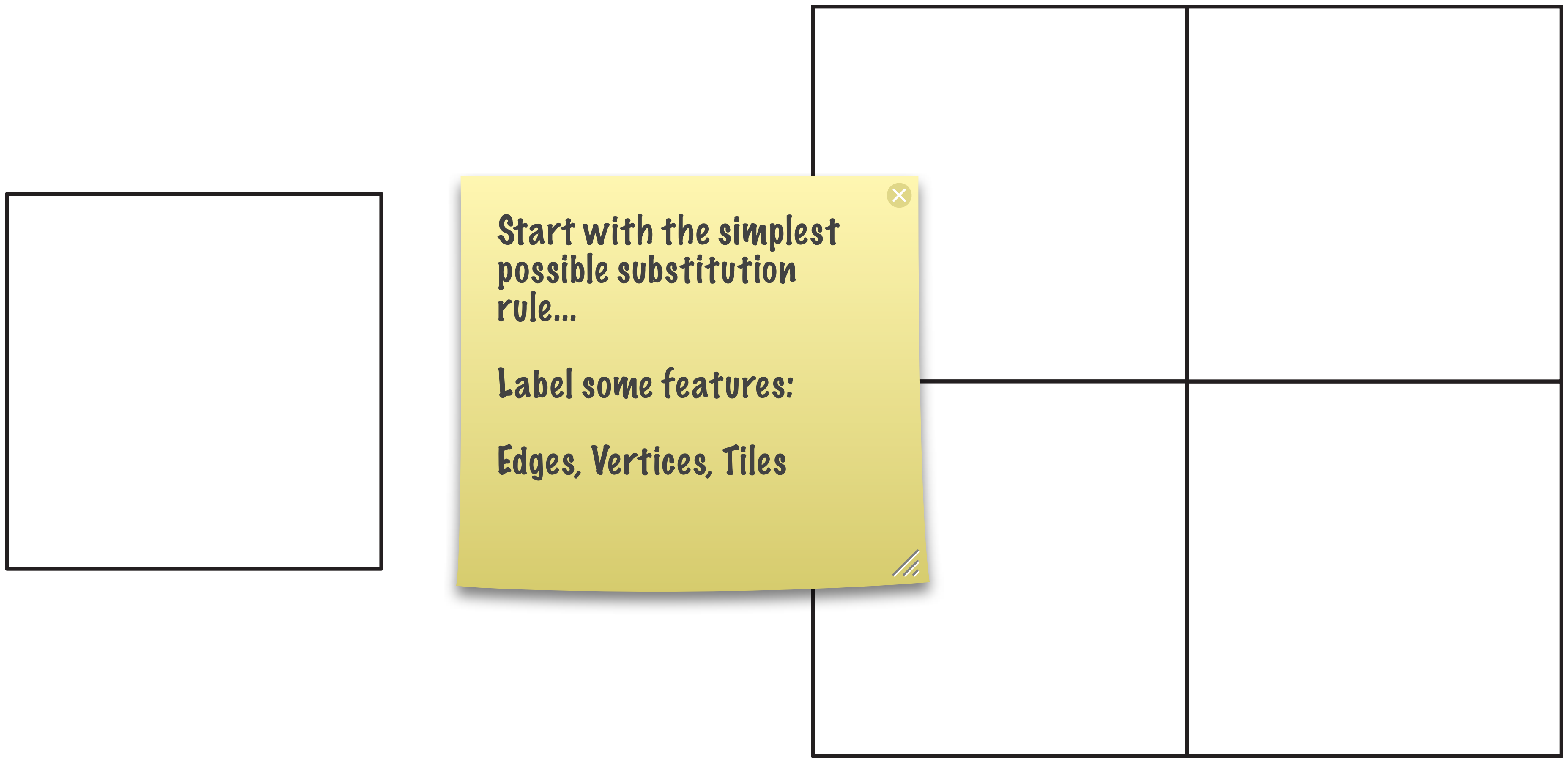
Sharhar Mozes, *Tilings, substitution systems and dynamical systems generated by them*, J. D'Analyse Math. 53, 1989, pp.139-186

Chaim Goodman-Strauss, *Matching rules and substitution tilings*, Annals of Mathematics 147 No. 1, 1998, pp. 181-223

A series of papers leading to a result that shows that we can get an aperiodic set of shapes from any substitution tiling...

Q: How?

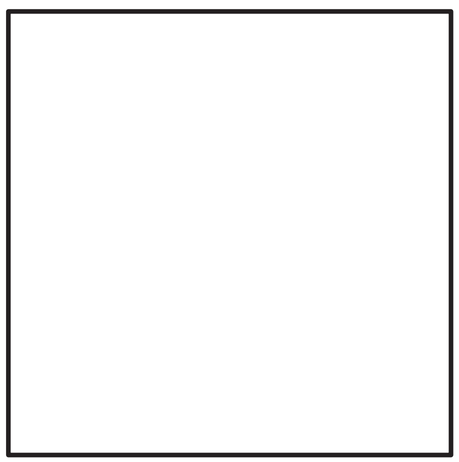
This is an important result, but not well understood. So now...



Start with the simplest possible substitution rule...

Label some features:

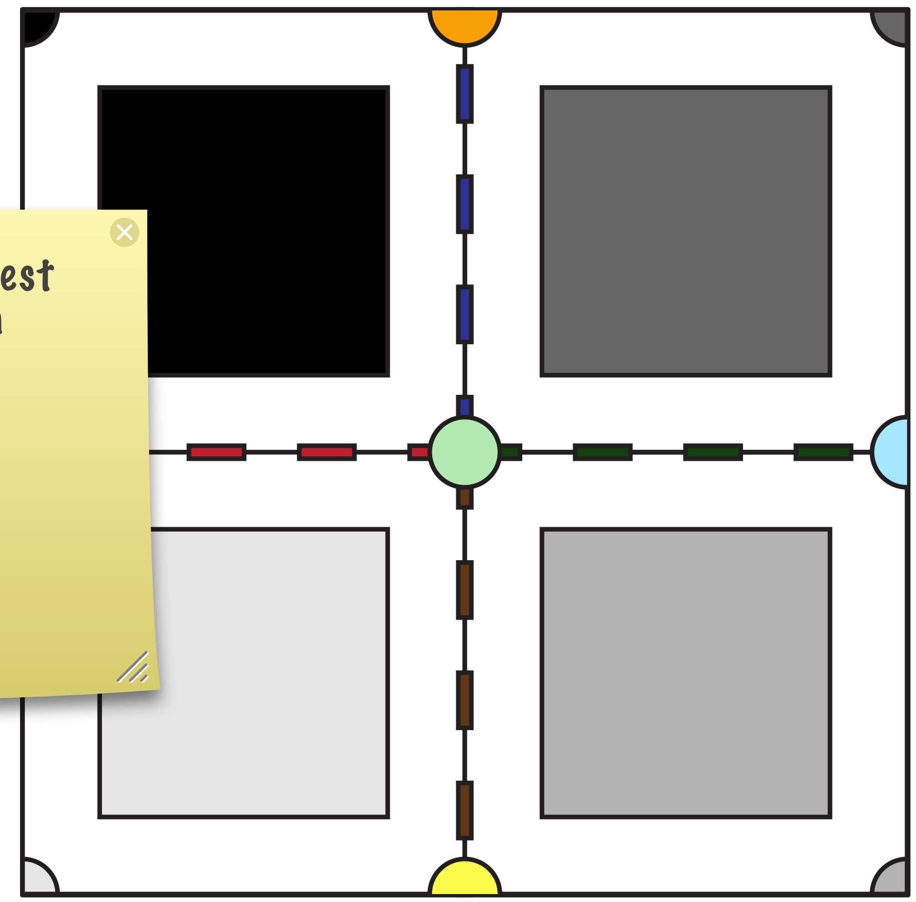
Edges, Vertices, Tiles



Start with the simplest possible substitution rule...

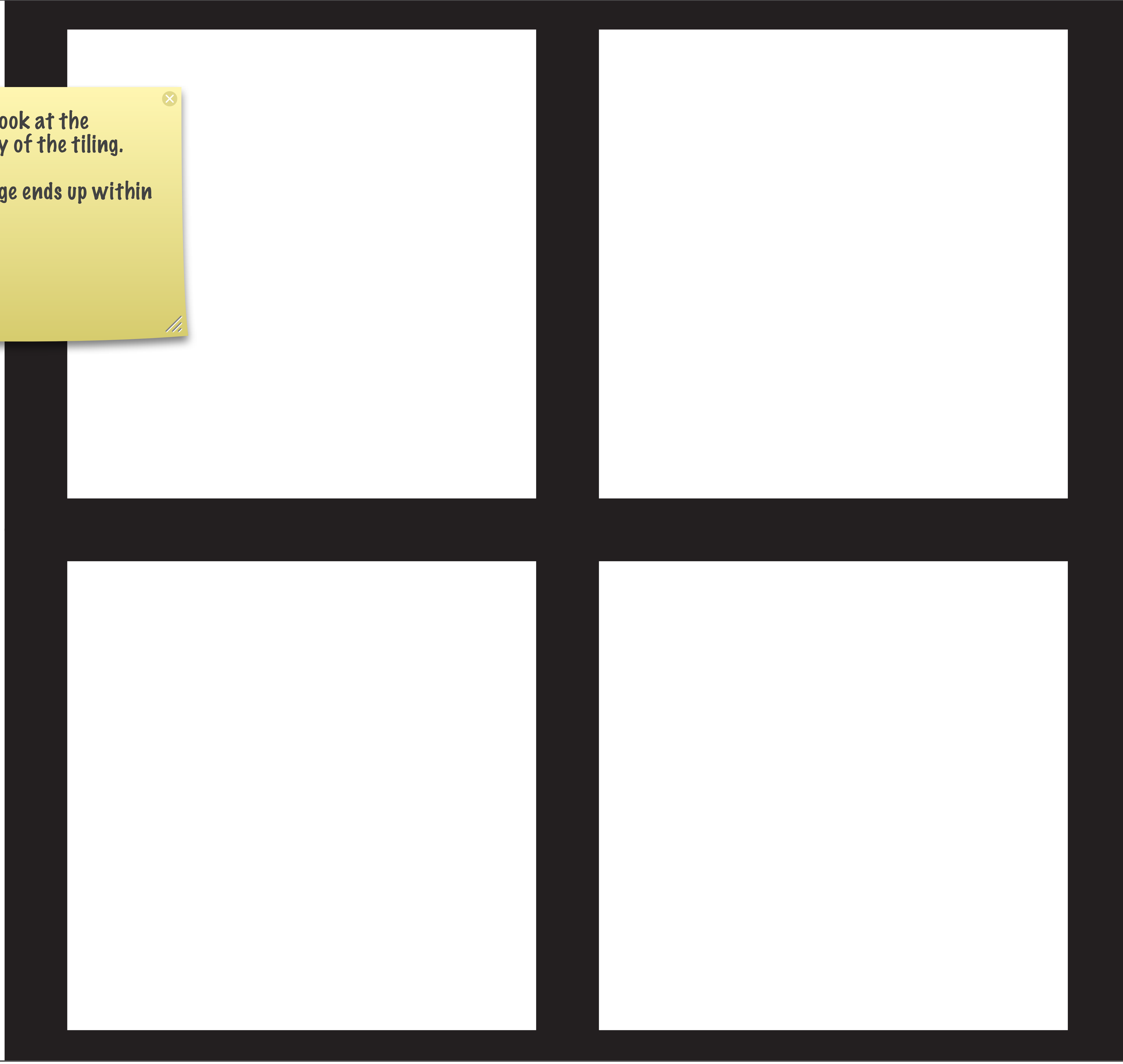
Label some features:

Edges, Vertices, Tiles



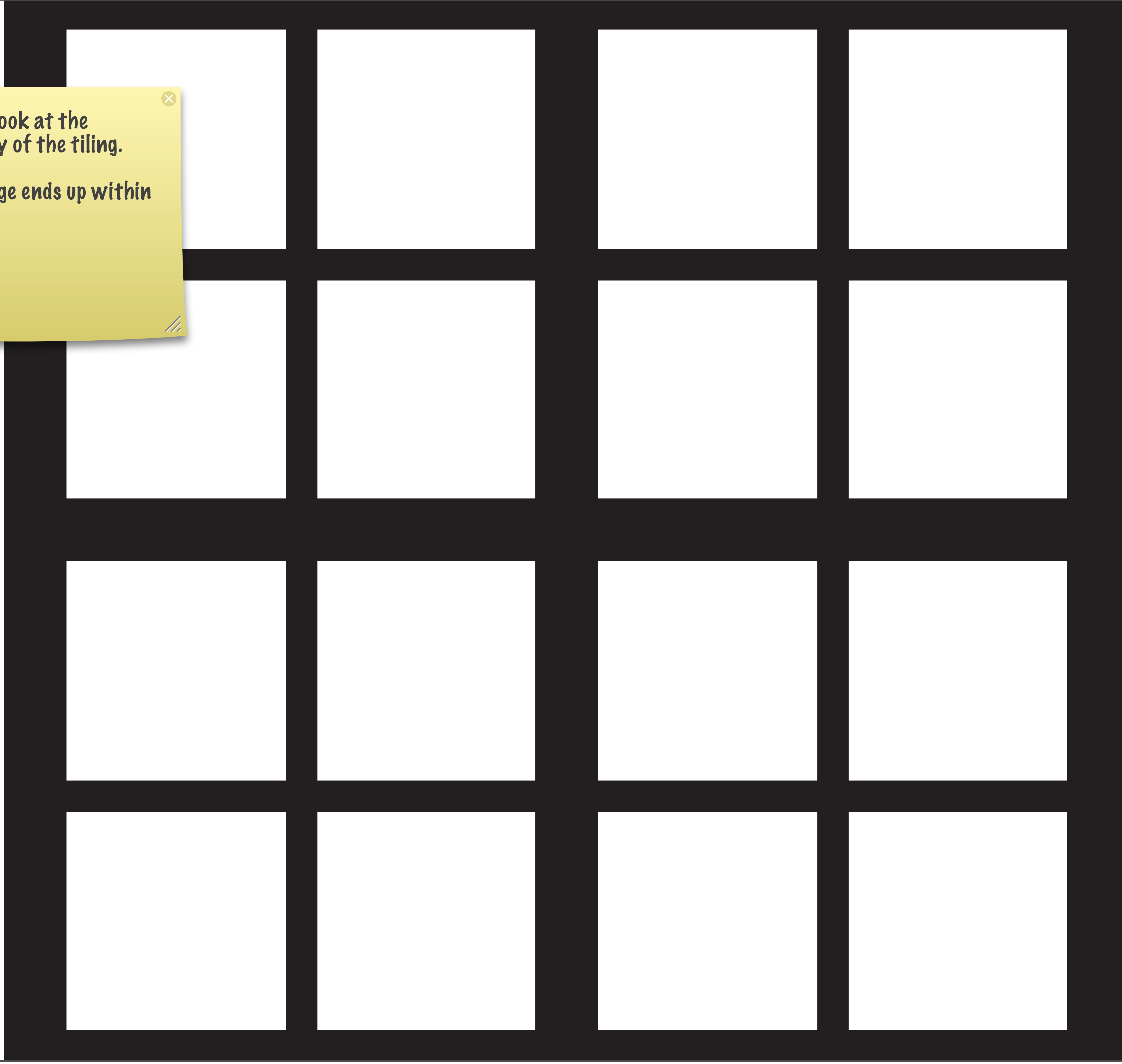
We can look at the hierarchy of the tiling.

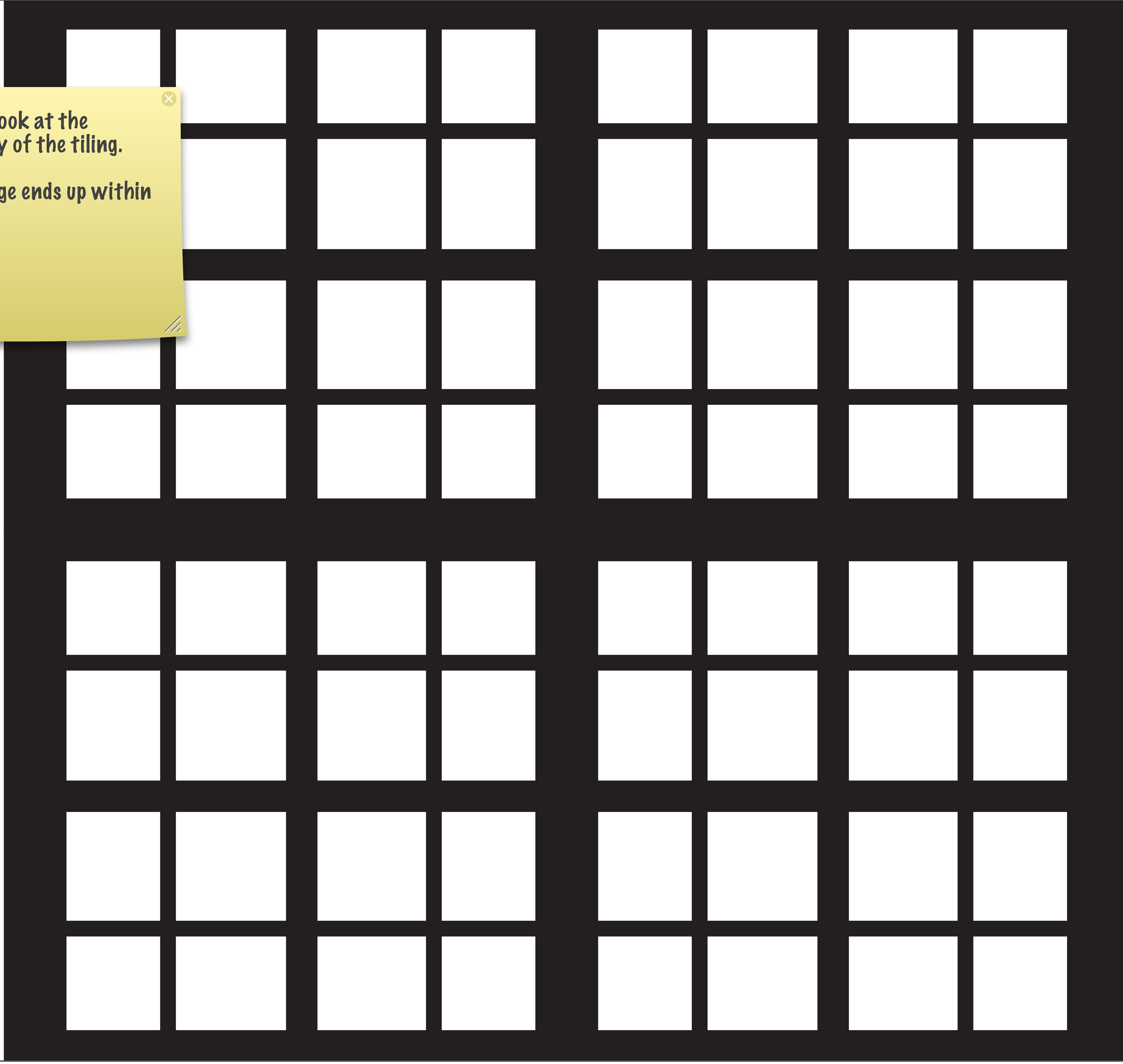
Every edge ends up within a tile



We can look at the hierarchy of the tiling.

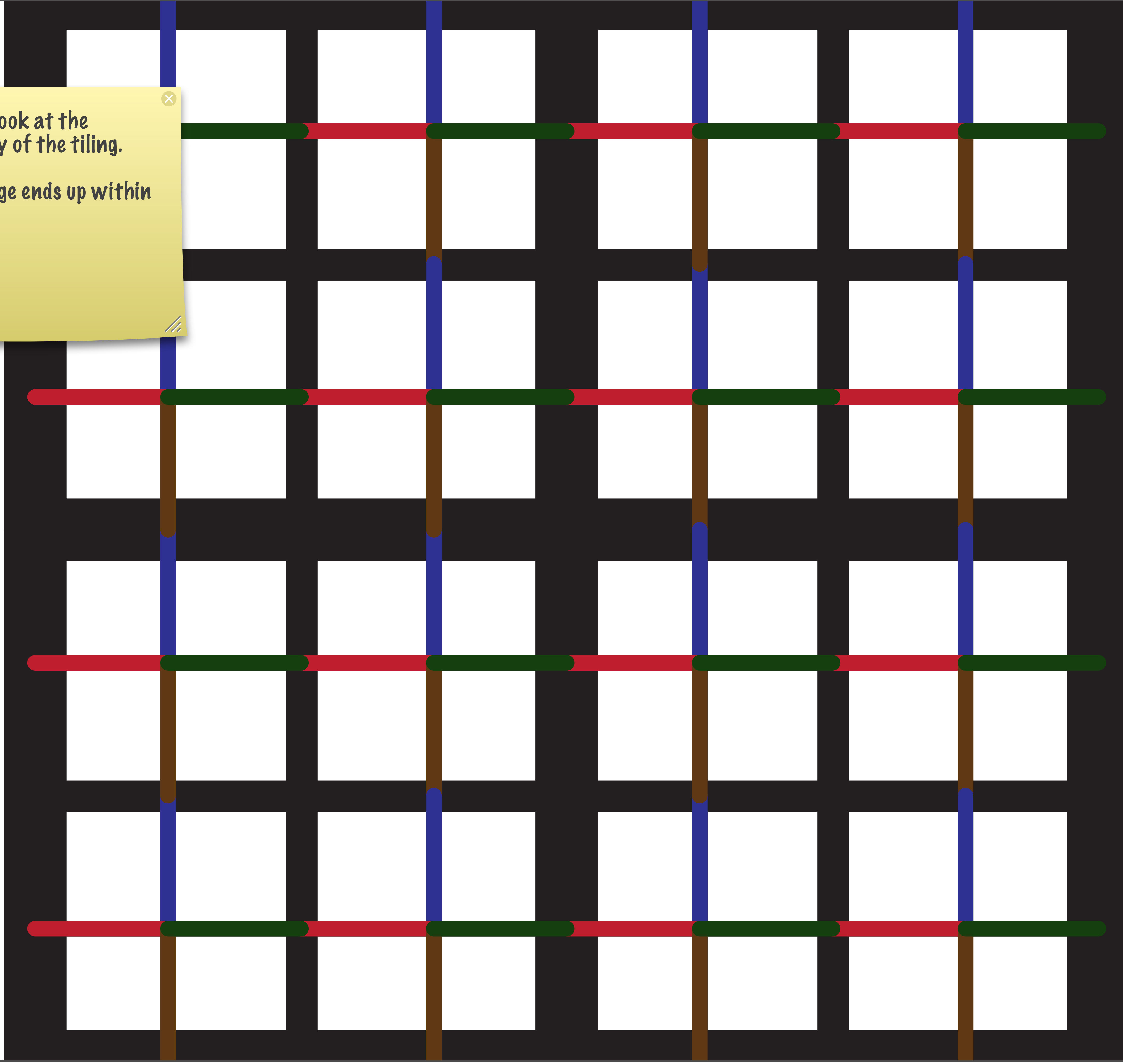
Every edge ends up within a tile





We can look at the hierarchy of the tiling.

Every edge ends up within a tile

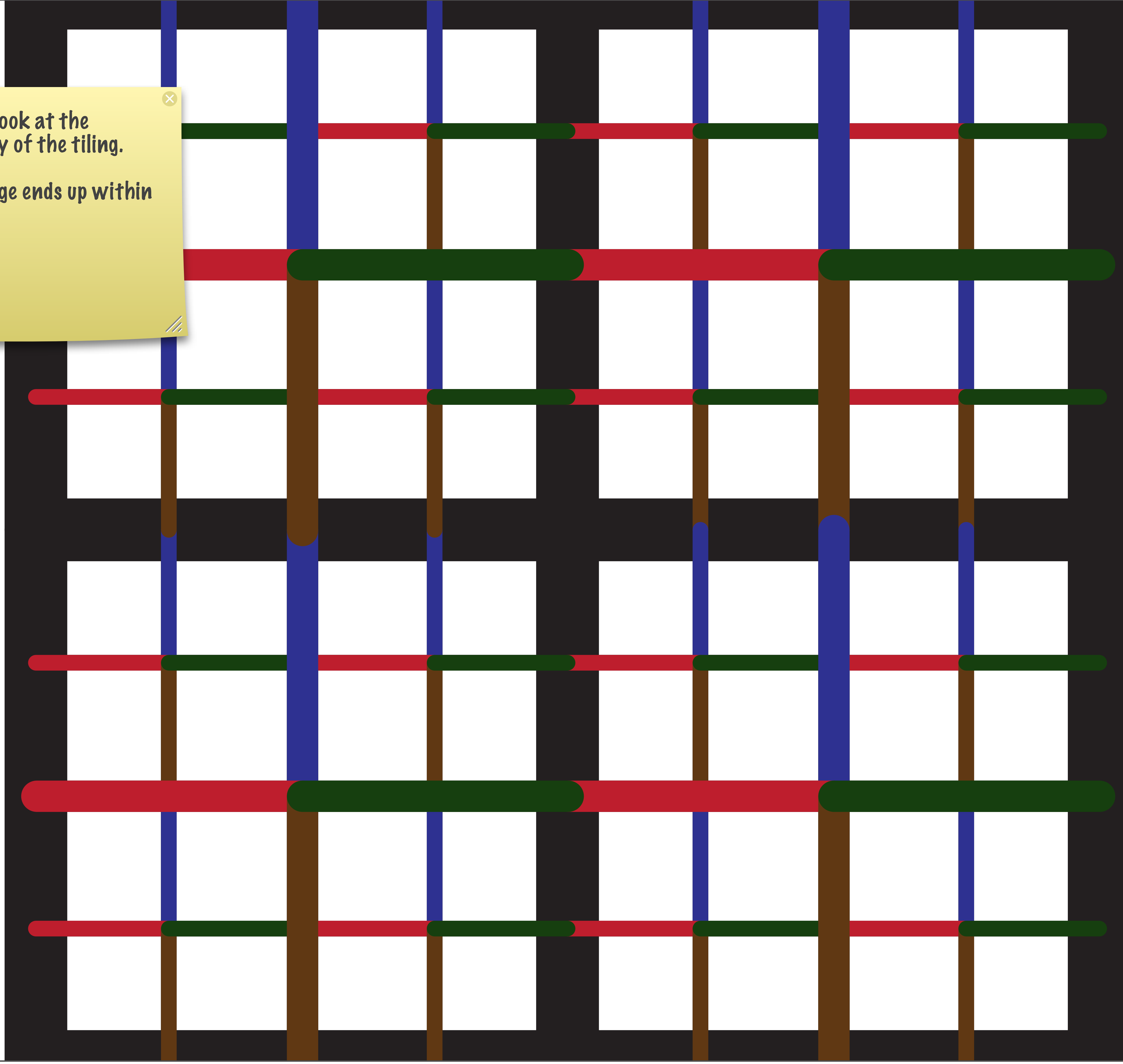


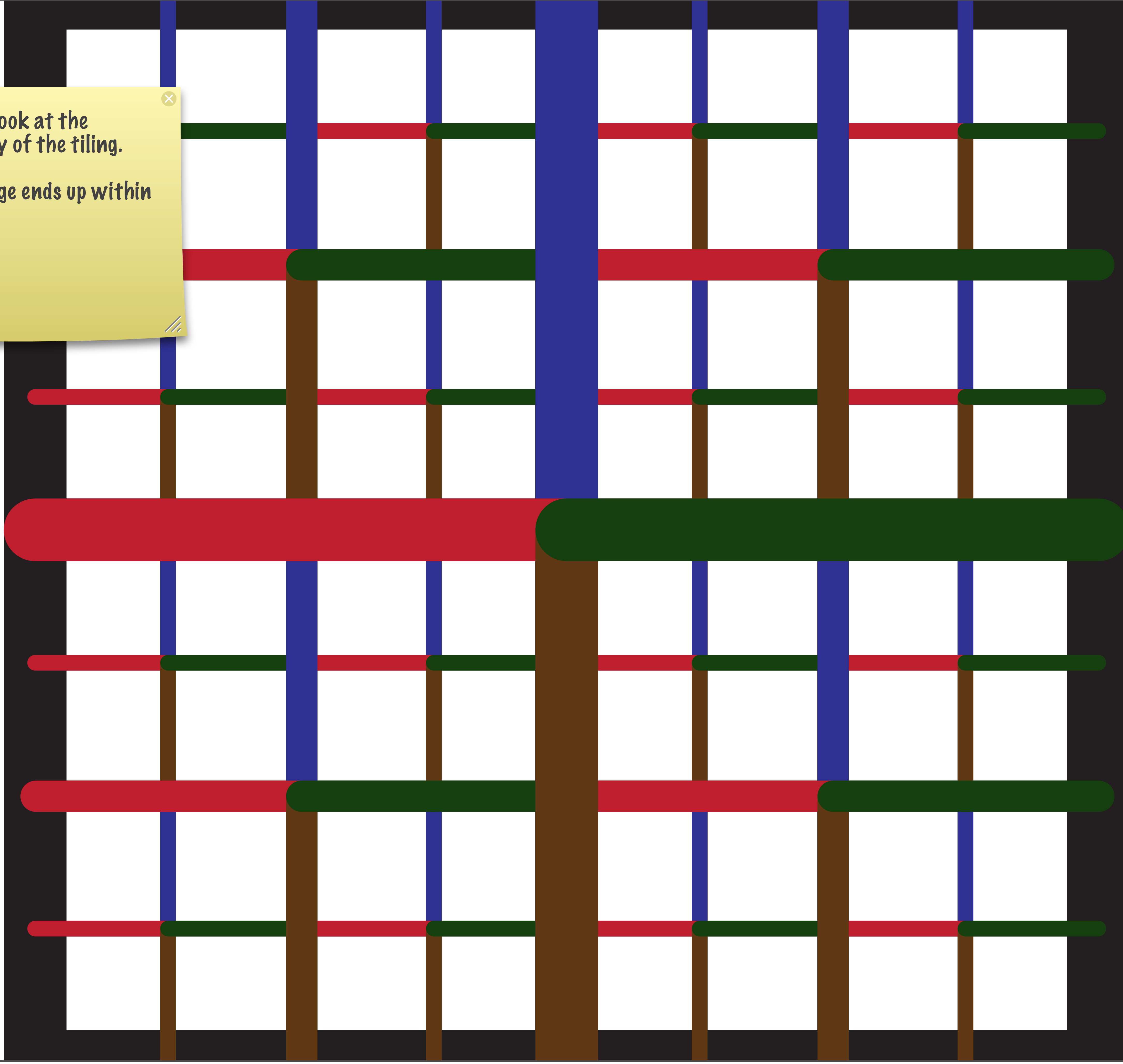
We can look at the hierarchy of the tiling.

Every edge ends up within a tile

We can look at the hierarchy of the tiling.

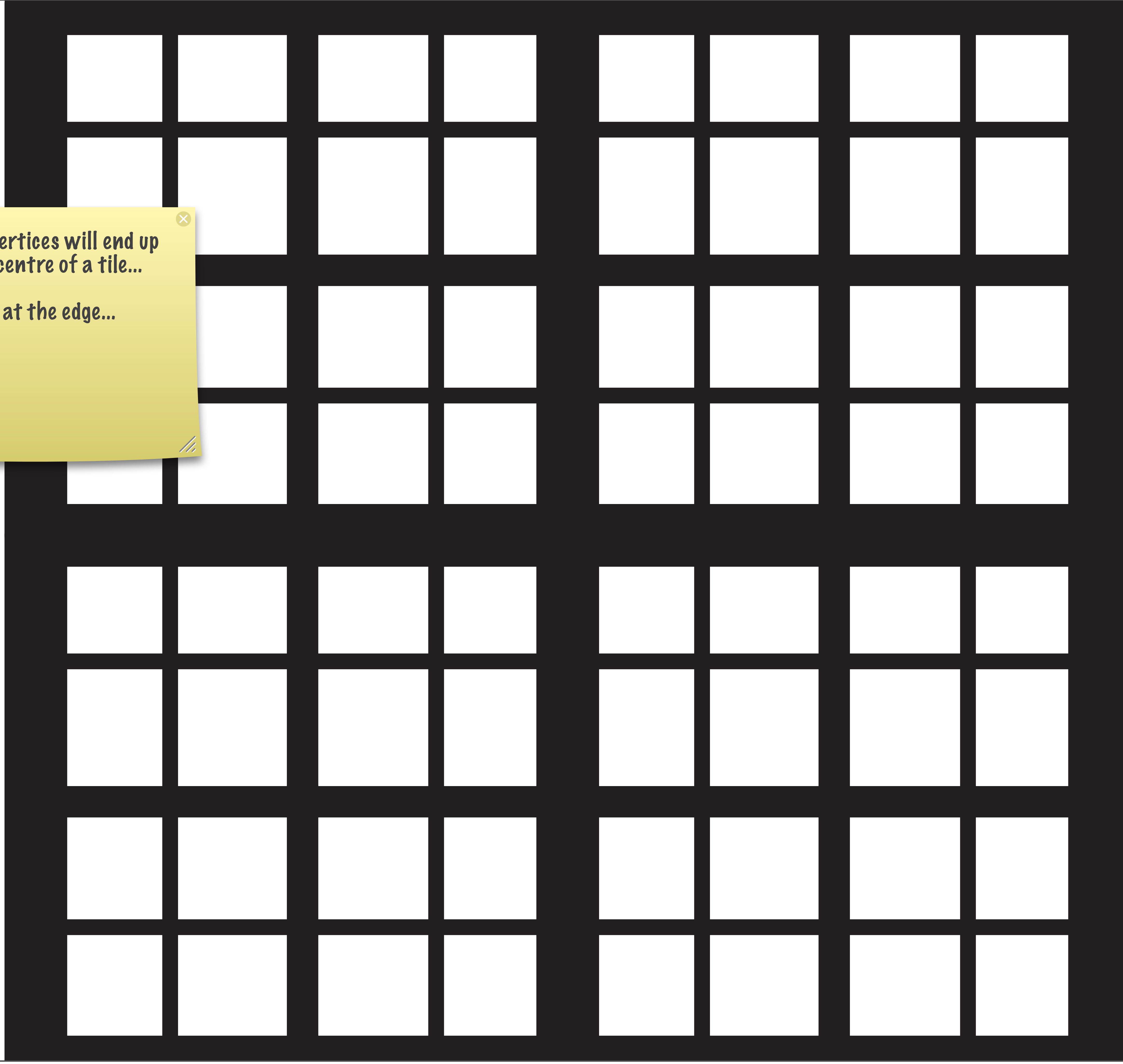
Every edge ends up within a tile





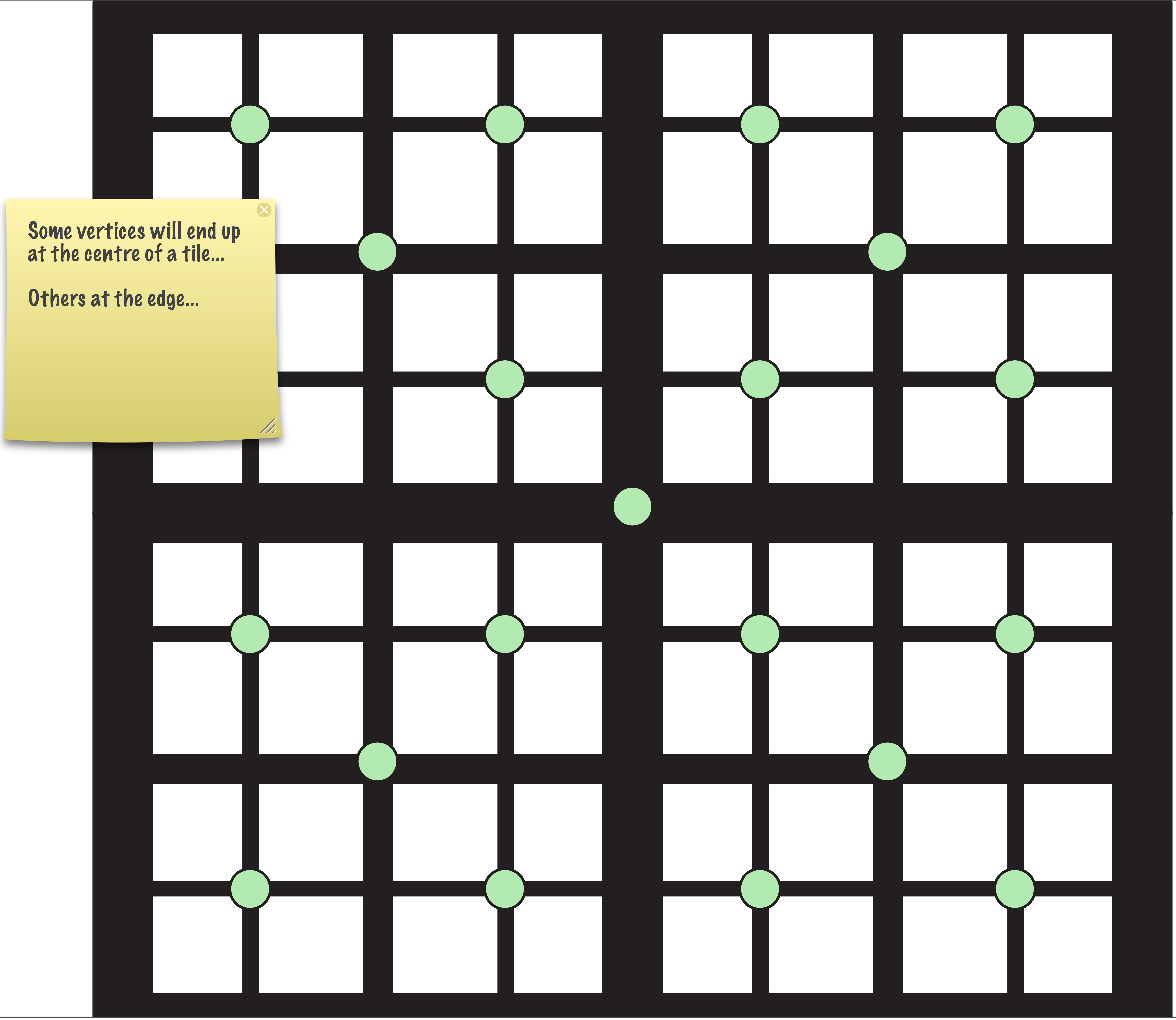
We can look at the hierarchy of the tiling.

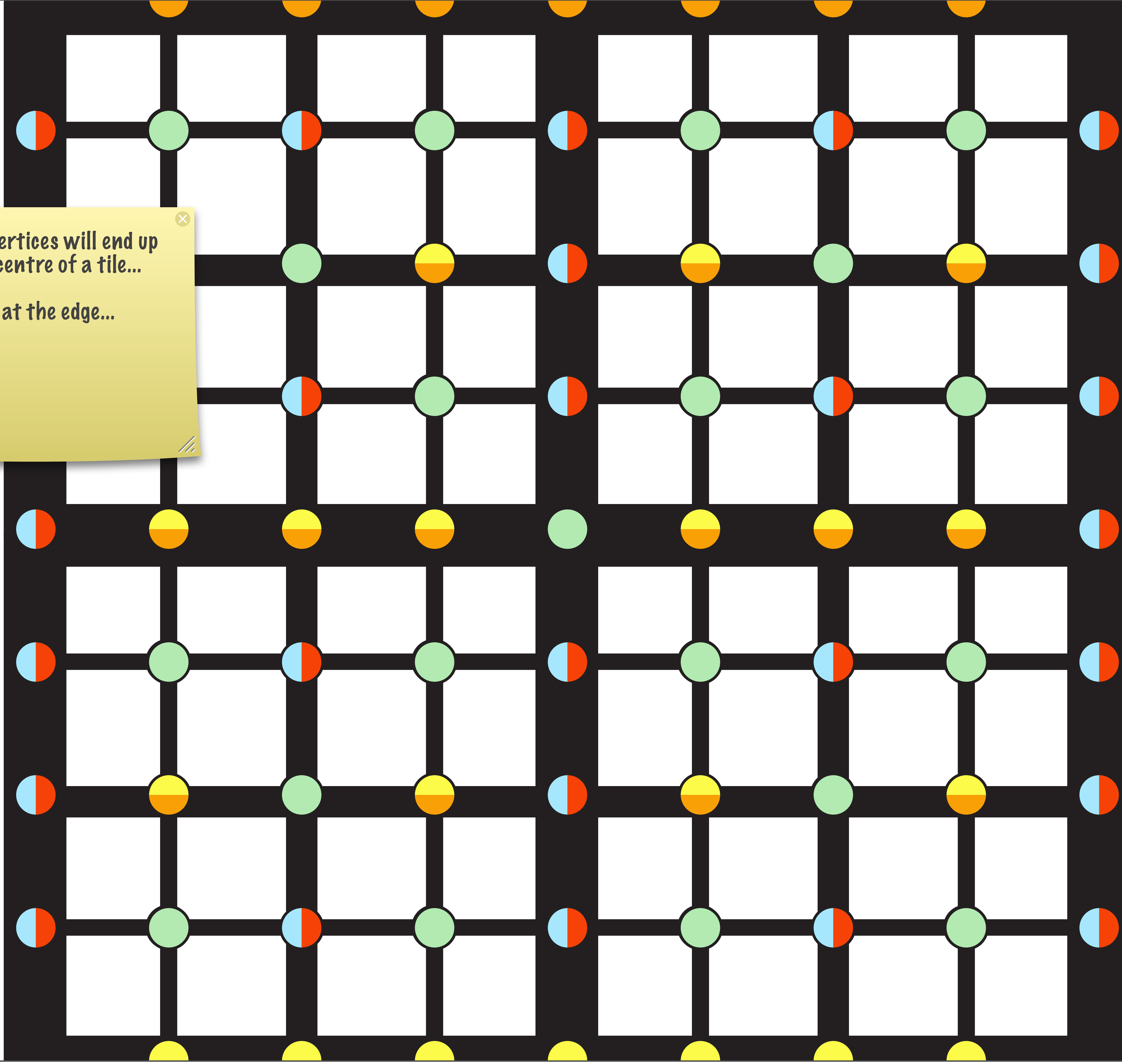
Every edge ends up within a tile



Some vertices will end up at the centre of a tile...

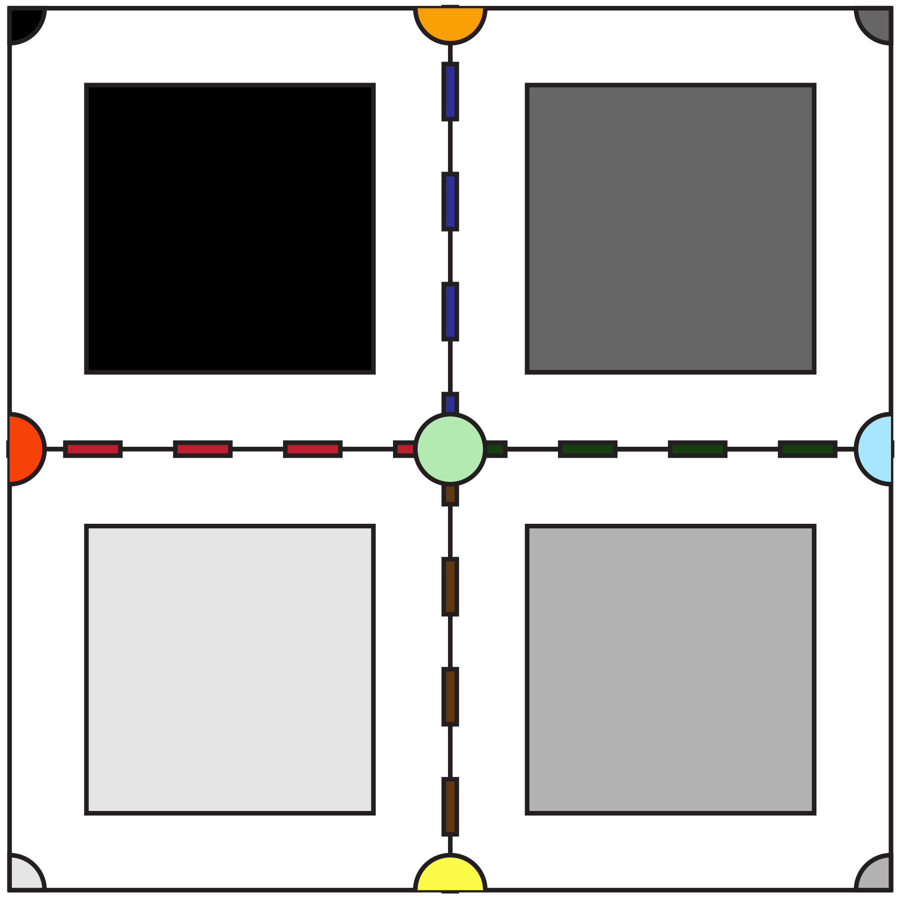
Others at the edge...


Some vertices will end up at the centre of a tile...
Others at the edge...

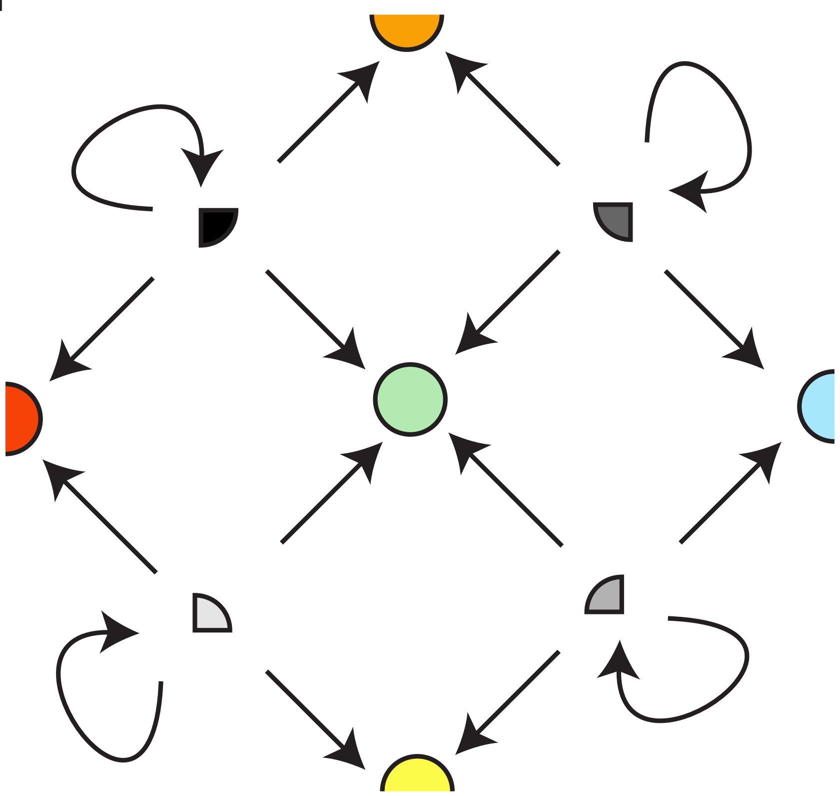


Some vertices will end up at the centre of a tile...

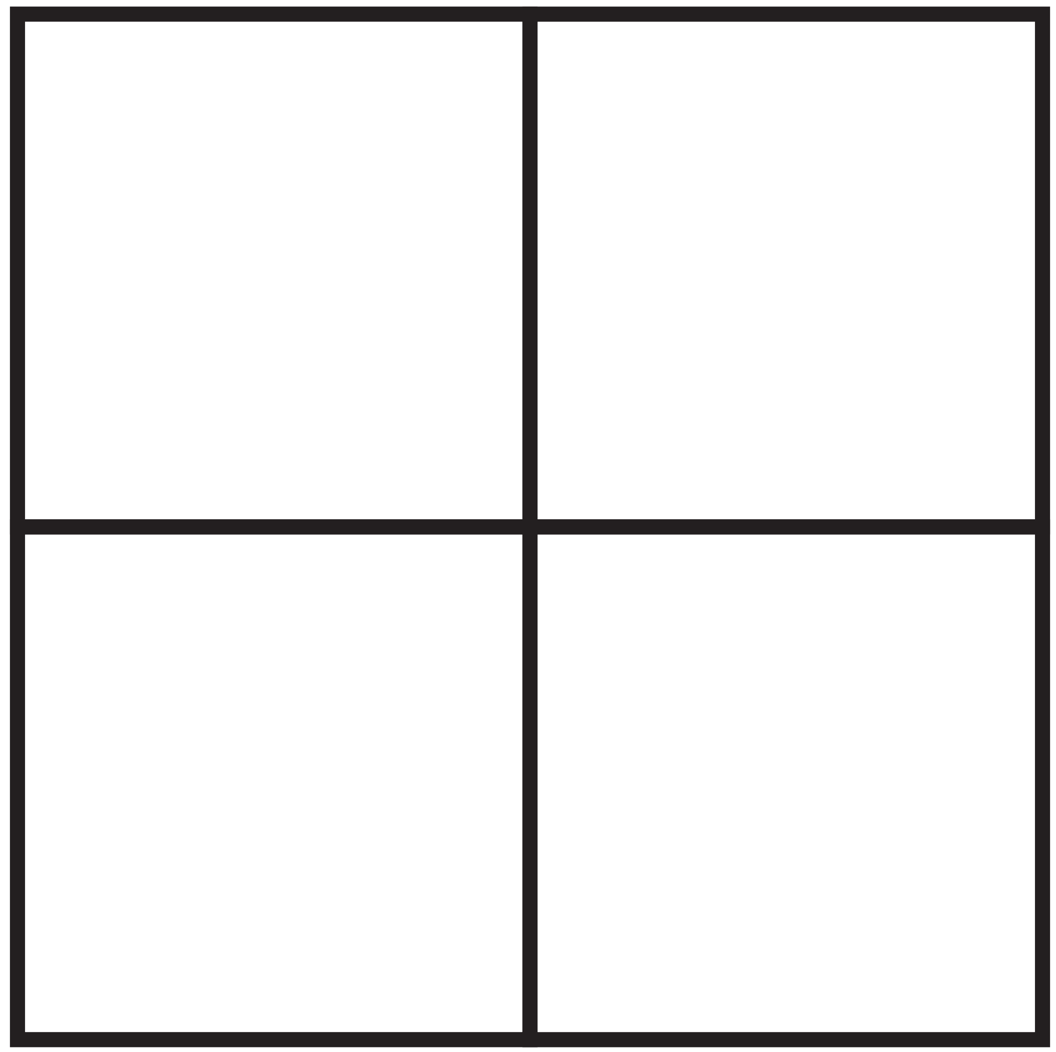
Others at the edge...



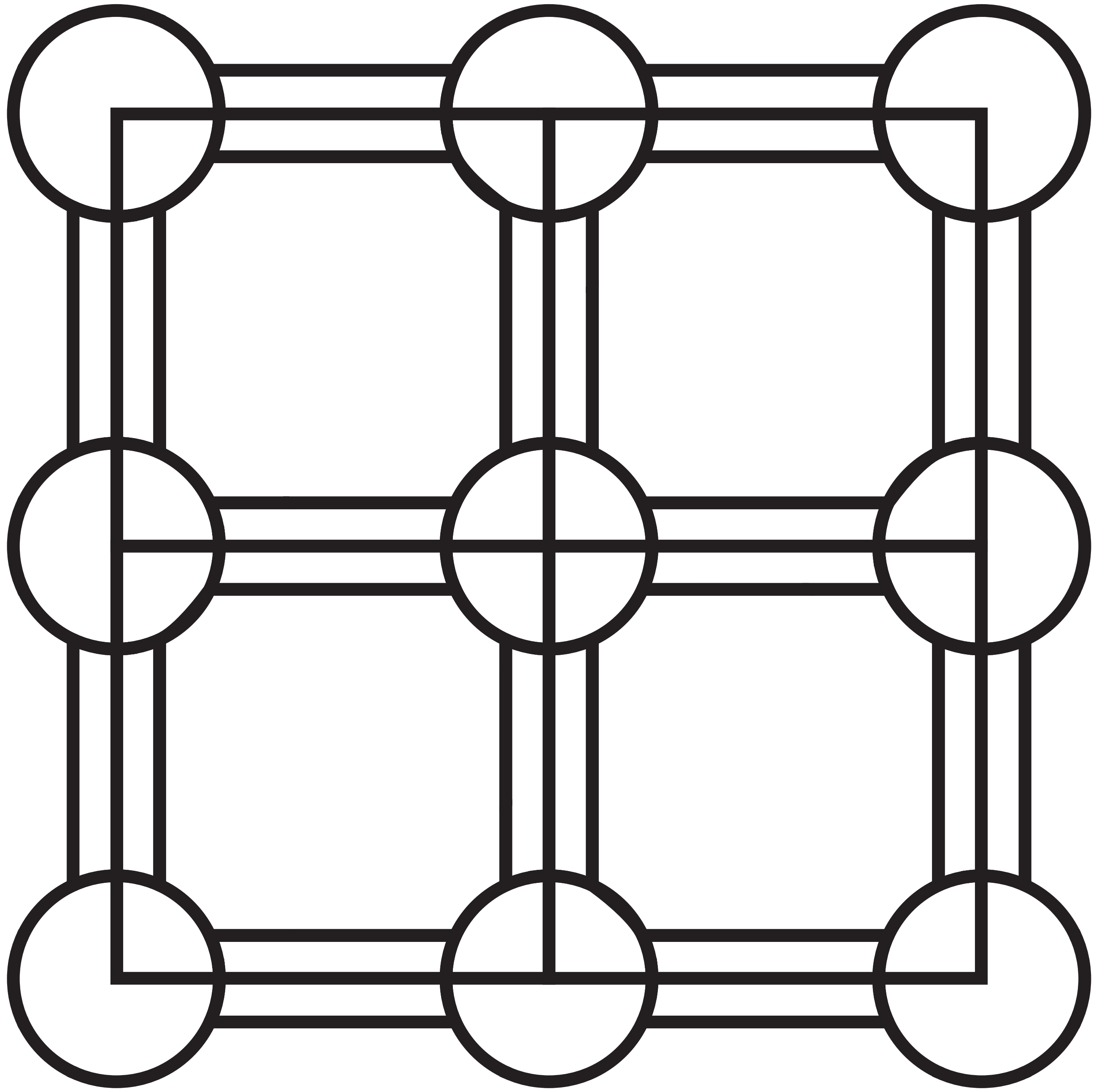
We can build a graph to show the possible roles that a vertex can take.



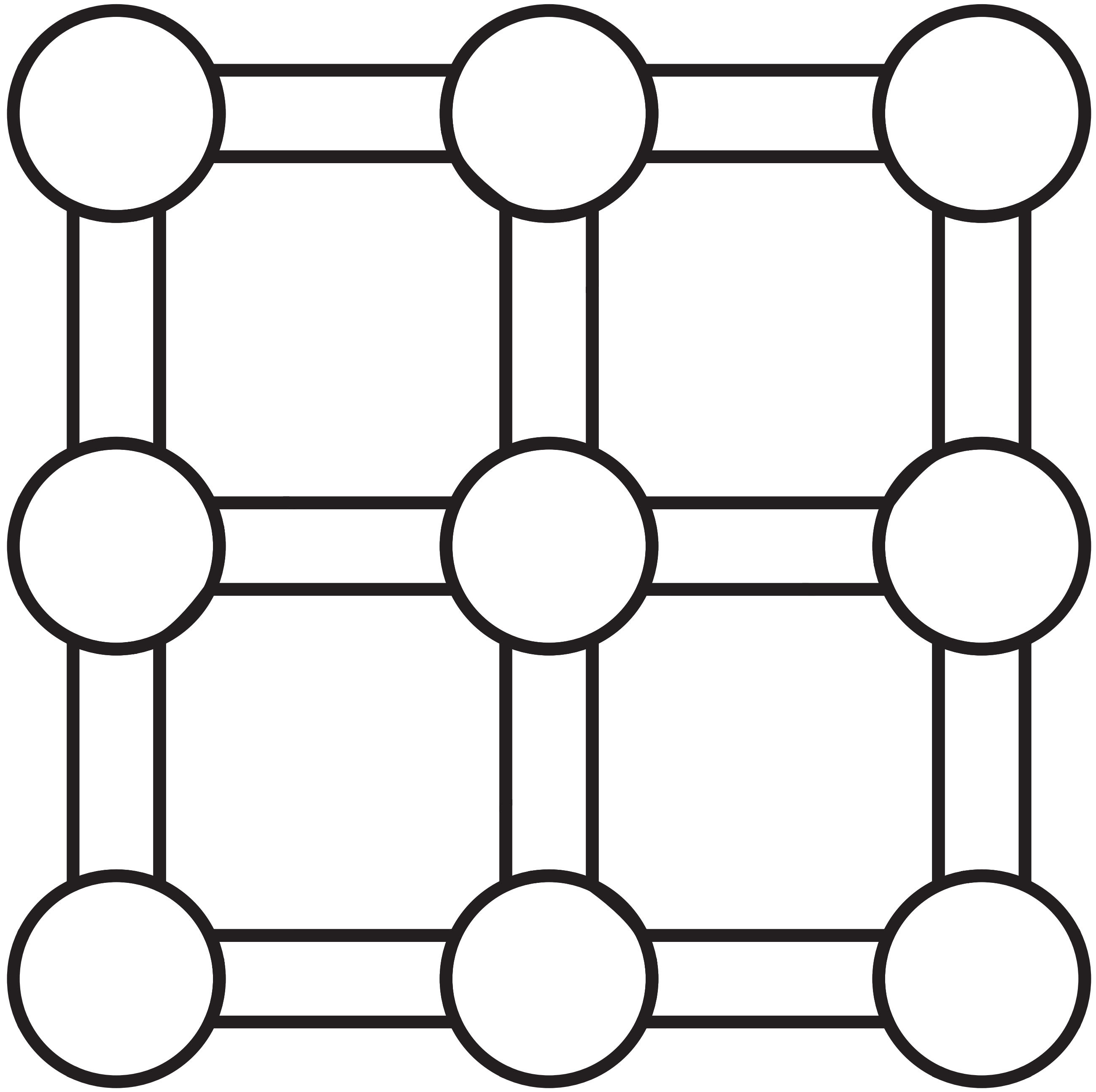
The tiles can be cut up to give the edges and vertices shape



The tiles can be cut up to give the edges and vertices shape



The tiles can be cut up to give the edges and vertices shape



Every Tile knows:

- Its tile type
- The eventual type of it special vertex

Every Edge knows:

- Its eventual type
- What supertile it lies in:
  - The tile type
  - The eventual type of its special vertex

Every Vertex knows

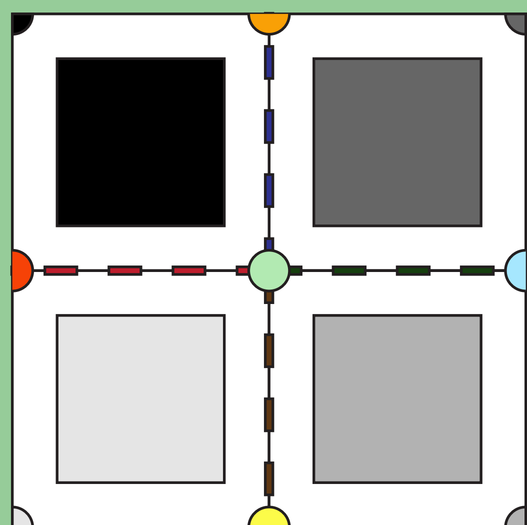
- Its eventual type
- What edges join it
- What supertile it lies in:
  - The tile type
  - The eventual type of its special vertex

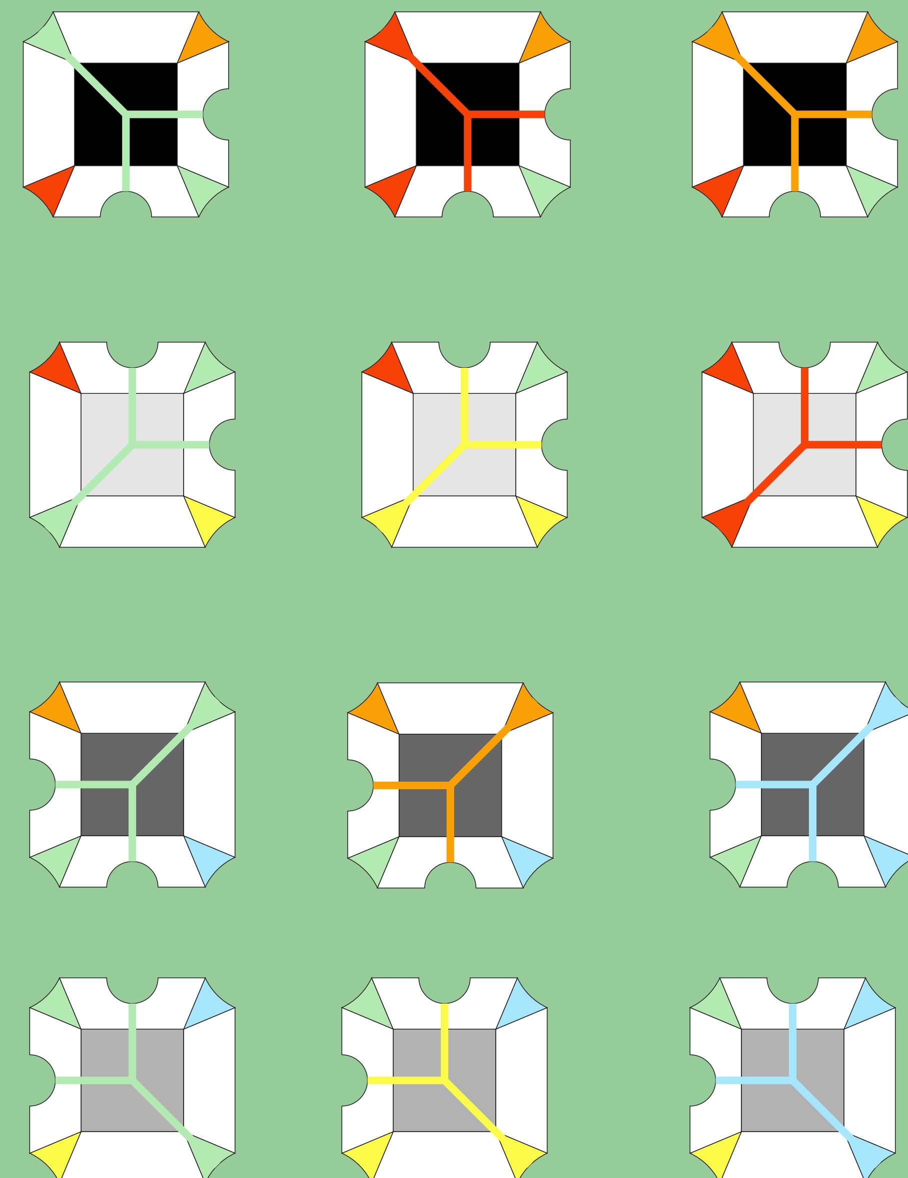
We want this information on the objects. The key is edges, they can grow transporting the information around the tiling.

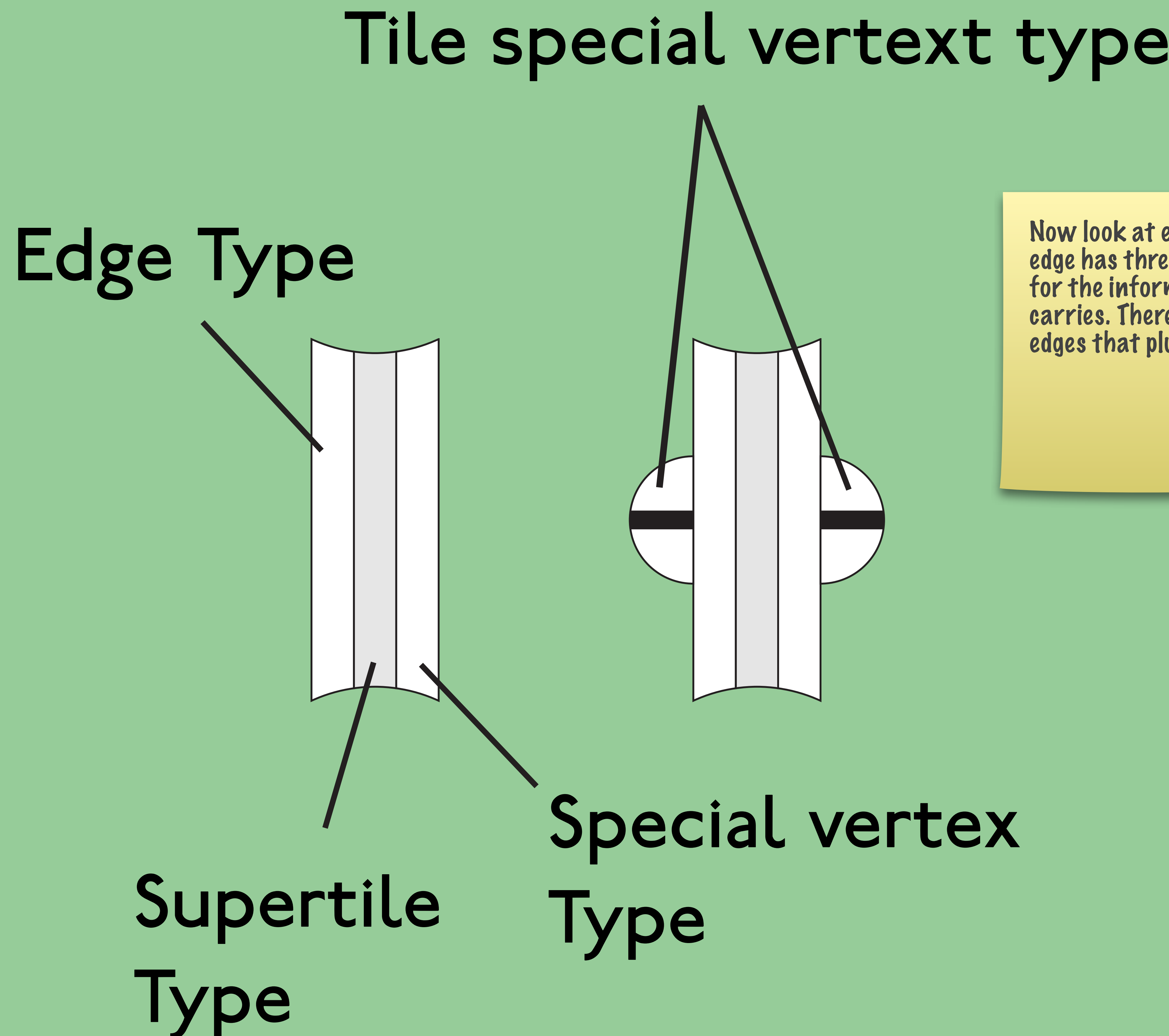
Now

We can start with the tiles. Each knows its type and the type of its special vertex.

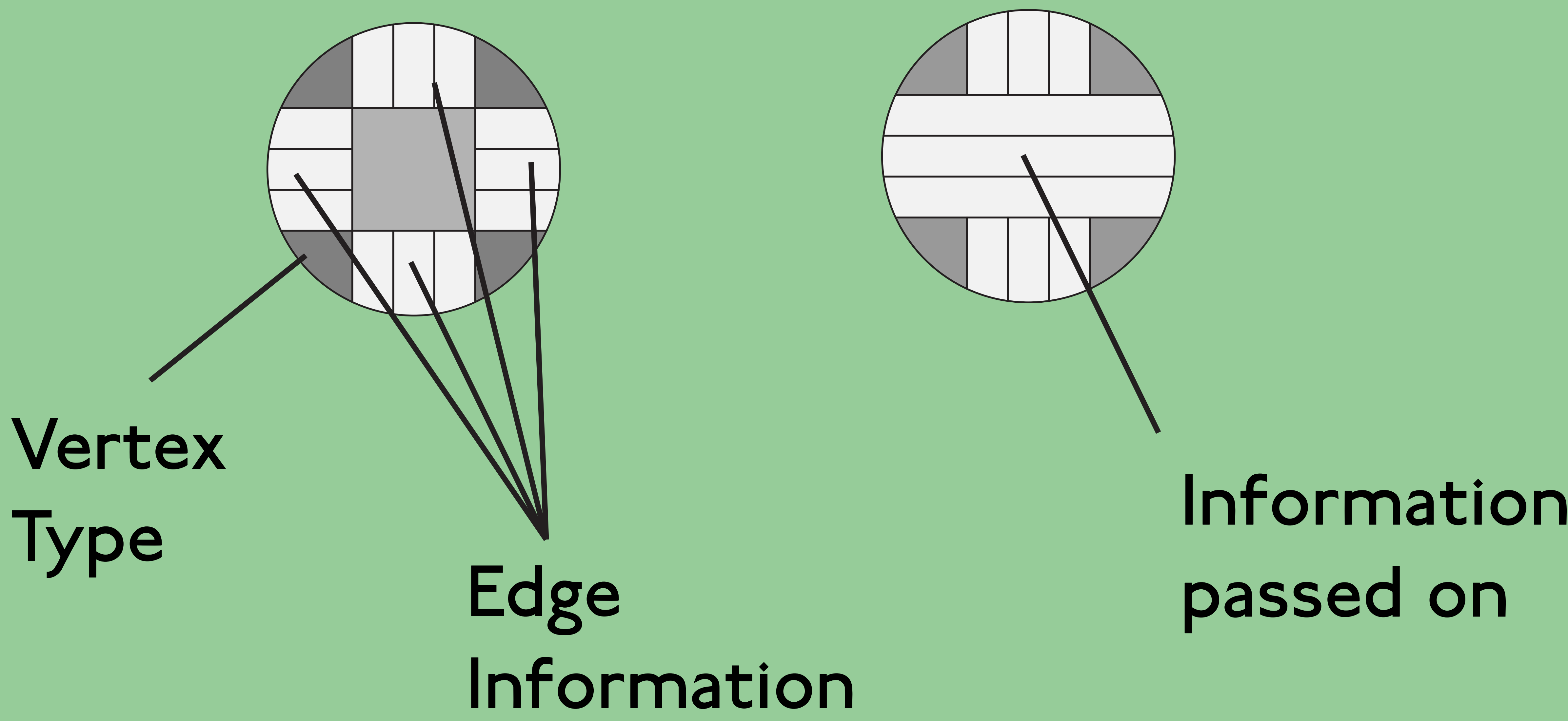
The edges of the supertile will also need to know the type of the special vertex, so the information is passed up to the internal edges.







Now look at edges. Each edge has three channels for the information it carries. There are also edges that plug into tiles.


Vertex
Type
Edge
Information
Information
passed on

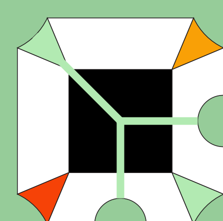
Lets build up a patch of tiling...

Note how the special vertex type is communicated up the hierarchy.

Thus each eleemnt can have finite information so there are a finite number of tiles...

but...

there are quite a few choices so...



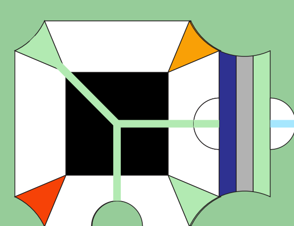
Lets build up a patch of tiling...

Note how the special vertex type is communicated up the hierarchy.

Thus each eleemnt can have finite information so there are a finite number of tiles...

but...

there are quite a few choices so...



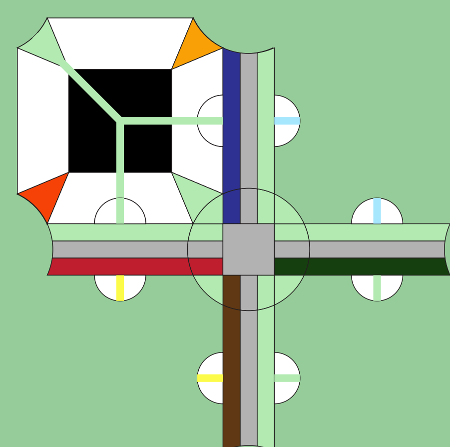
Lets build up a patch of tiling...

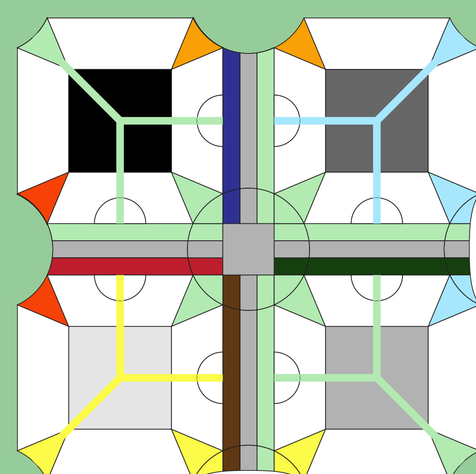
Note how the special vertex type is communicated up the hierarchy.

Thus each eleemnt can have finite information so there are a finite number of tiles...

but...

there are quite a few choices so...





Lets build up a patch of tiling...

Note how the special vertex type is communicated up the hierarchy.

Thus each eleemnt can have finite information so there are a finite number of tiles...

but...

there are quite a few choices so...

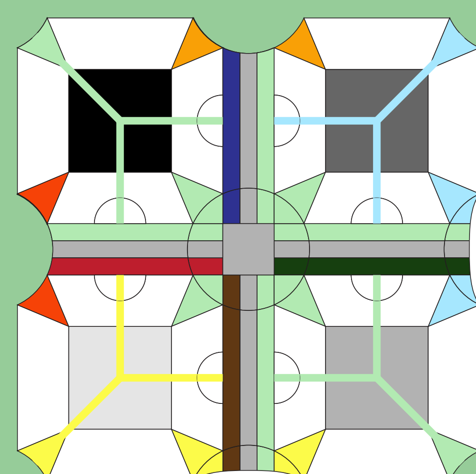
Lets build up a patch of tiling...

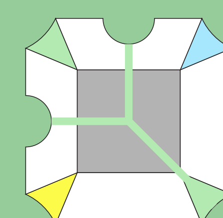
Note how the special vertex type is communicated up the hierarchy.

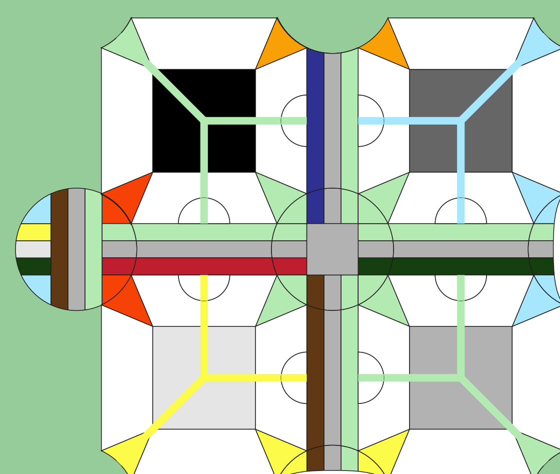
Thus each eleemnt can have finite information so there are a finite number of tiles...

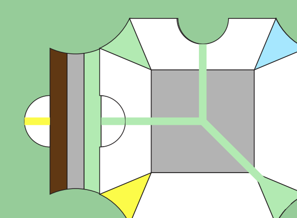
but...

there are quite a few choices so...









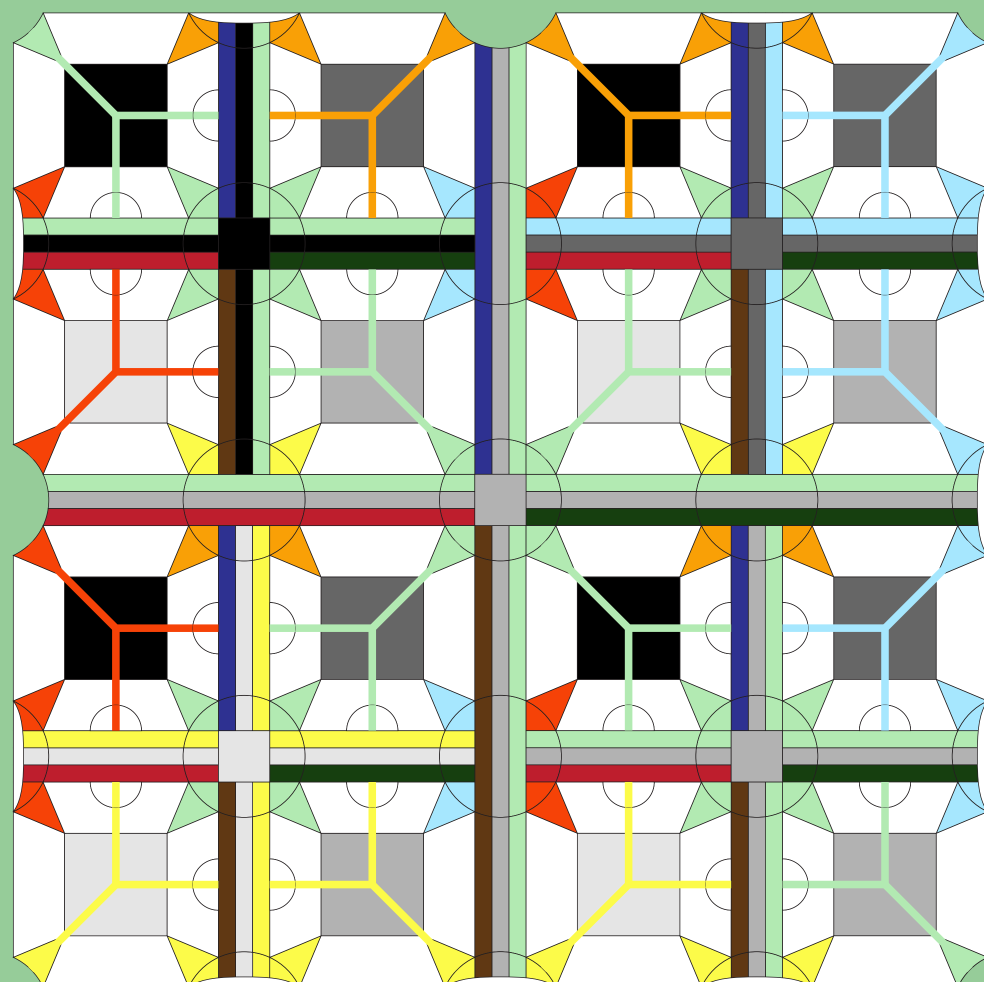
Lets build up a patch of tiling...

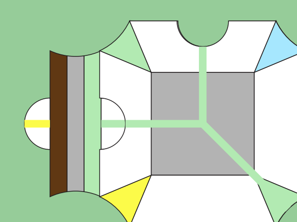
Note how the special vertex type is communicated up the hierarchy.

Thus each eleemnt can have finite information so there are a finite number of tiles...

but...

there are quite a few choices so...





Lets build up a patch of tiling...

Note how the special vertex type is communicated up the hierarchy.

Thus each eleemnt can have finite information so there are a finite number of tiles...

but...

there are quite a few choices so...

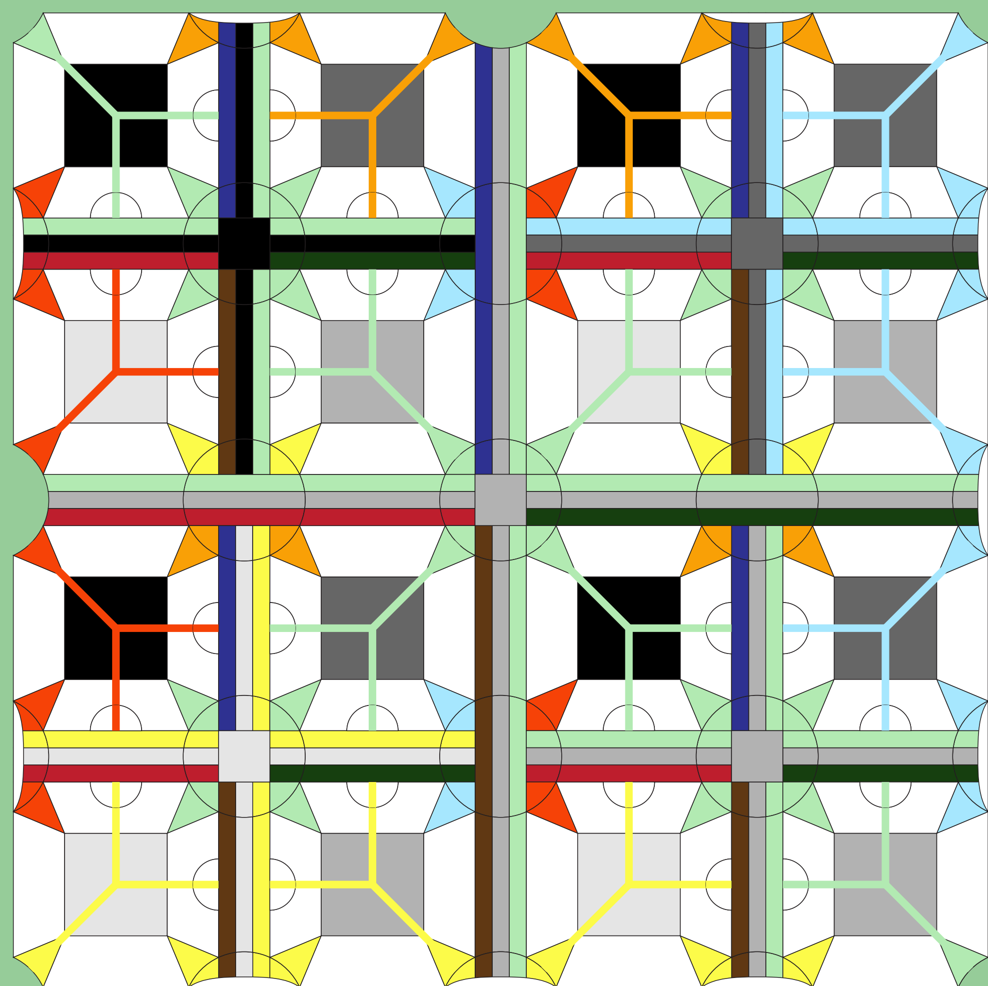
Lets build up a patch of tiling...

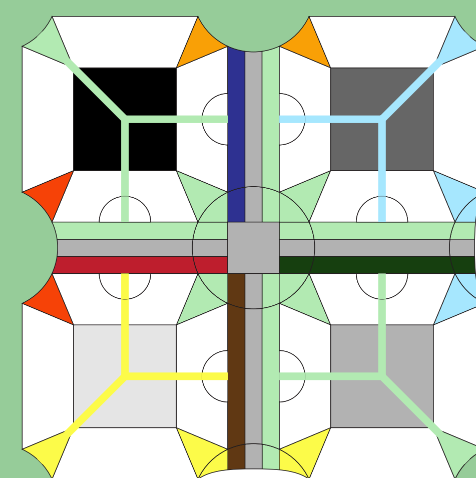
Note how the special vertex type is communicated up the hierarchy.

Thus each eleemnt can have finite information so there are a finite number of tiles...

but...

there are quite a few choices so...





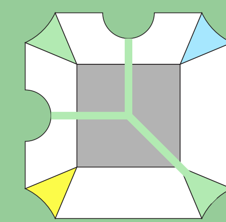
Lets build up a patch of tiling...

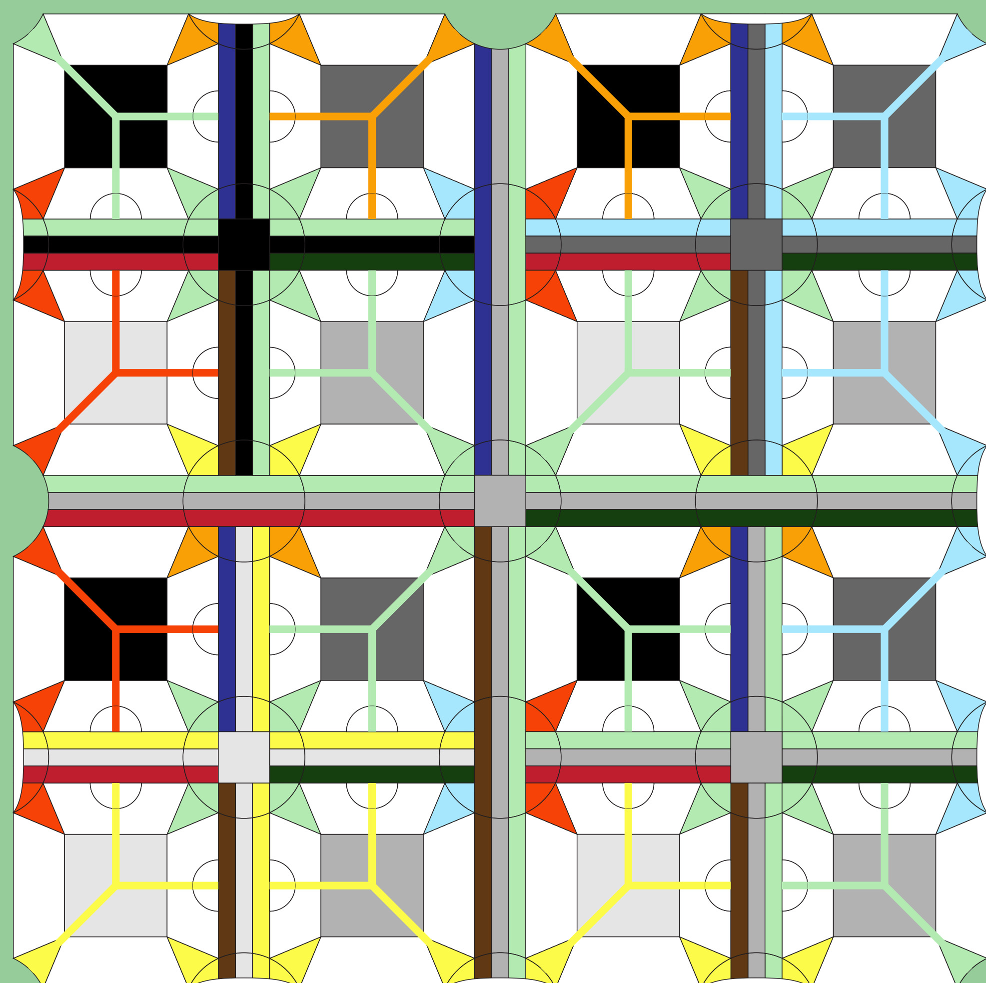
Note how the special vertex type is communicated up the hierarchy.

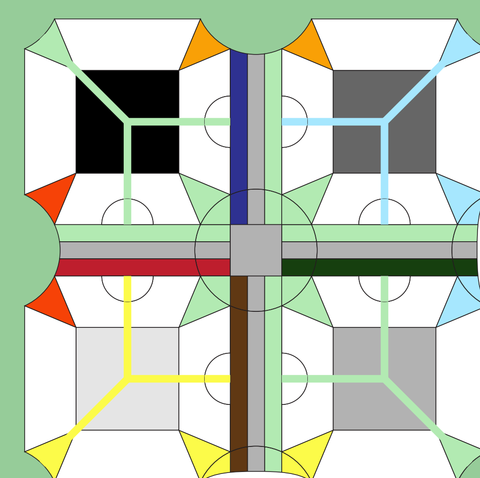
Thus each eleemnt can have finite information so there are a finite number of tiles...

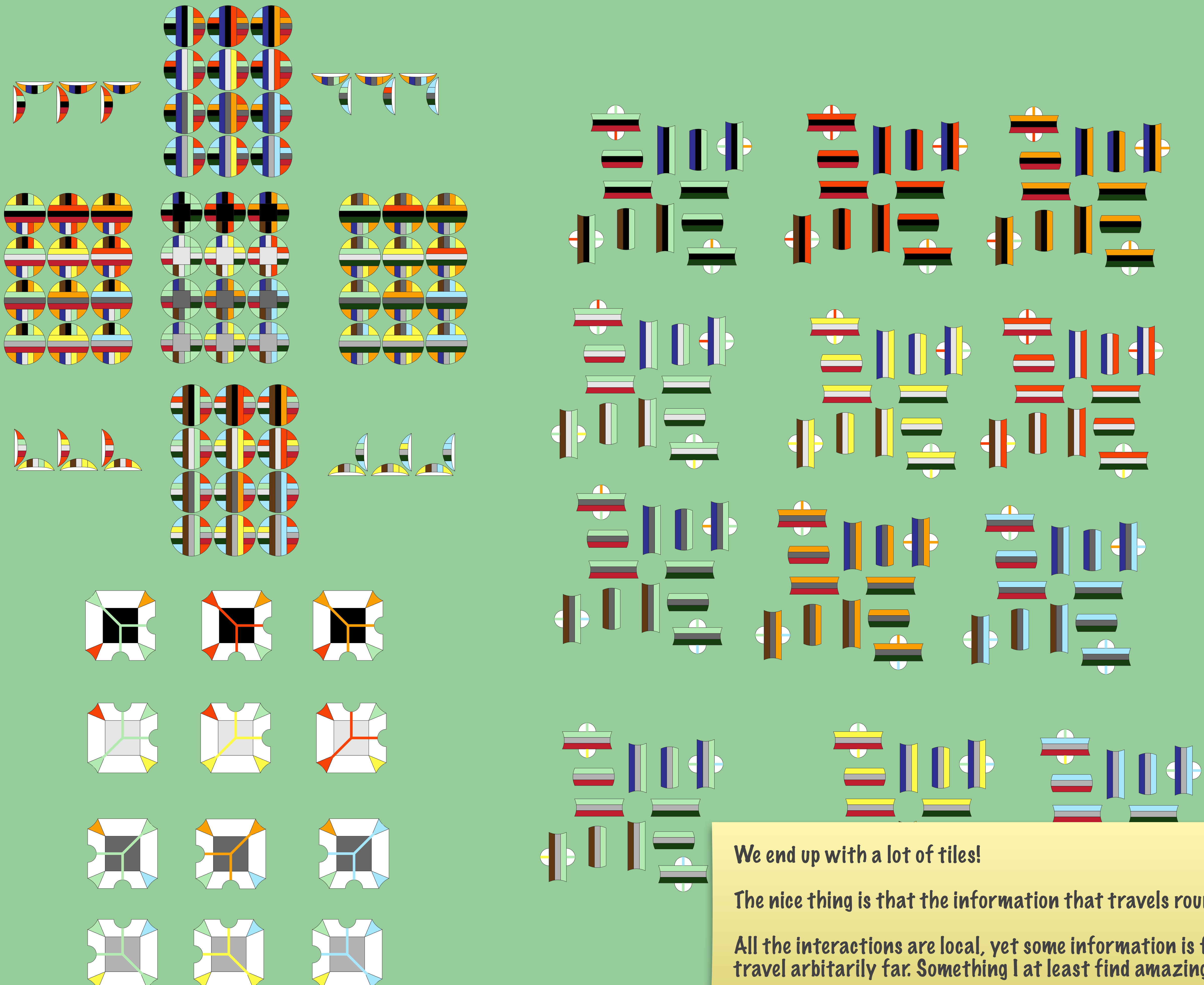
but...

there are quite a few choices so...









We end up with a lot of tiles!

The nice thing is that the information that travels round is explicit.

All the interactions are local, yet some information is forced to travel arbitarily far. Something I at least find amazing.