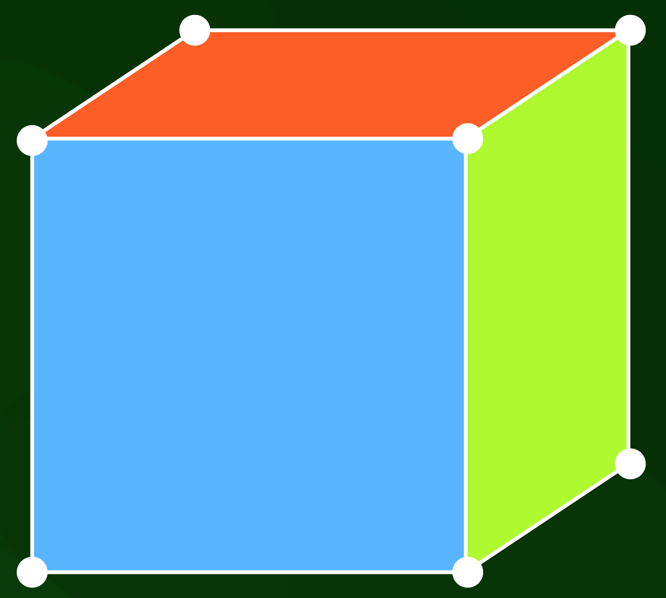
Generalised maps and combinatorial maps

GEO1004:
3D modelling of the built environment

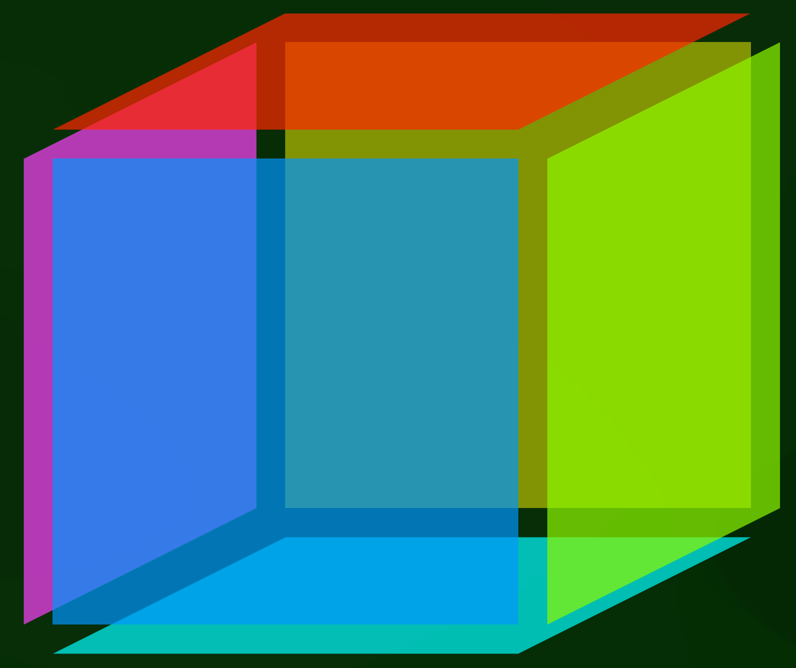
https://3d.bk.tudelft.nl/courses/geo1004

3D geoinformation
Department of Urbanism
Faculty of Architecture and the Built Environment
Delft University of Technology

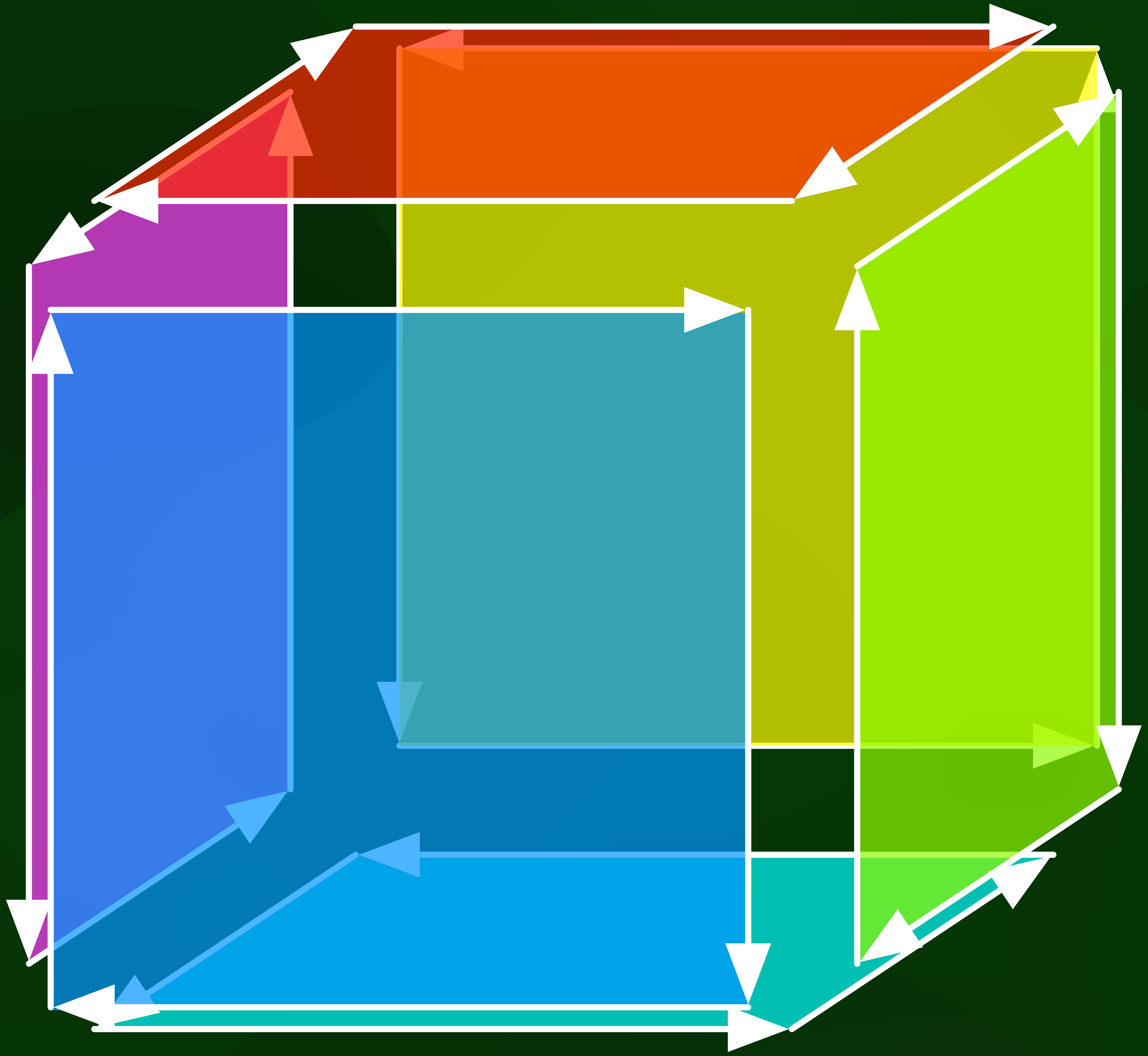
So far… (3D through b-rep)



3D object

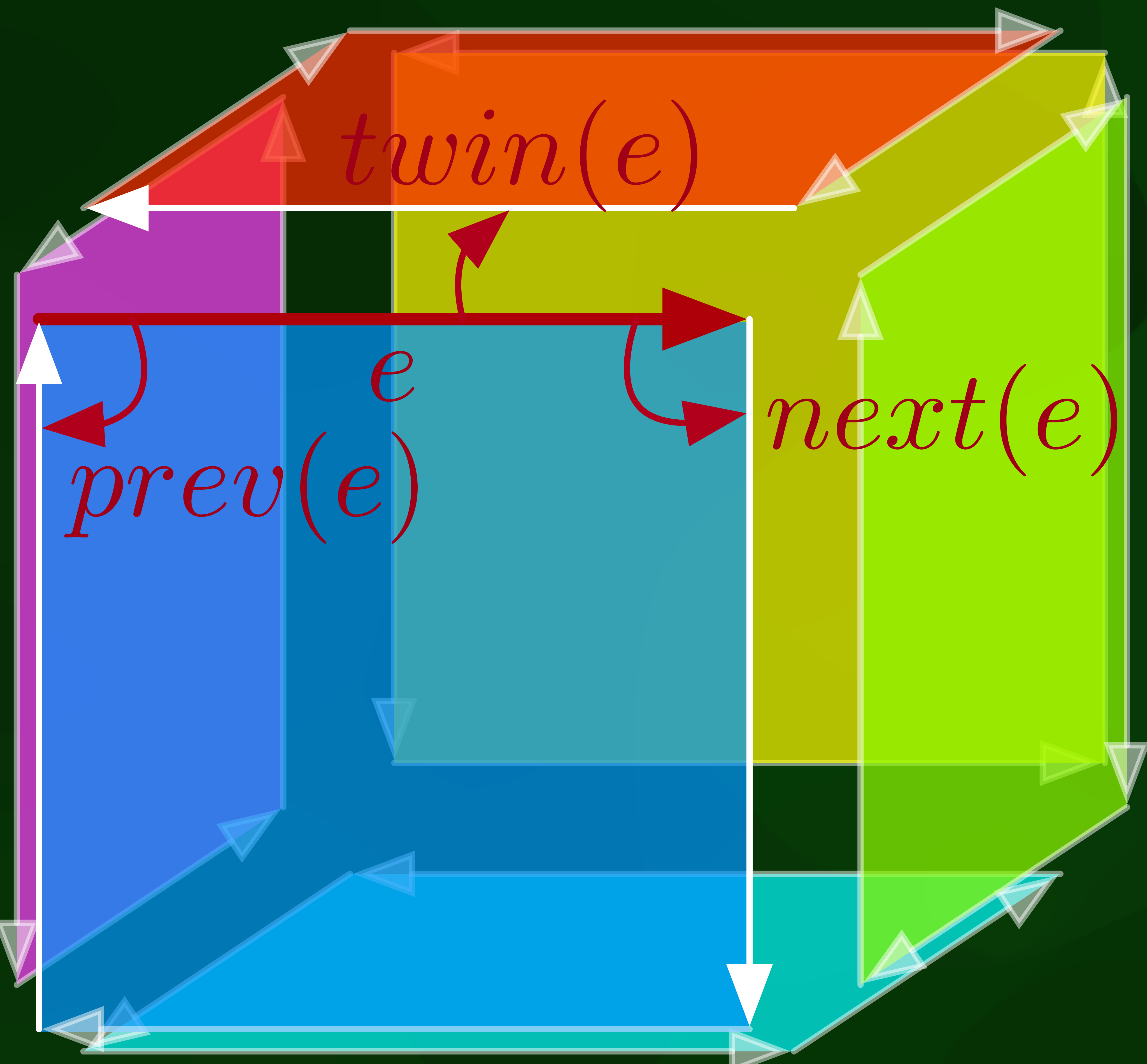


b-rep
(2D boundary)



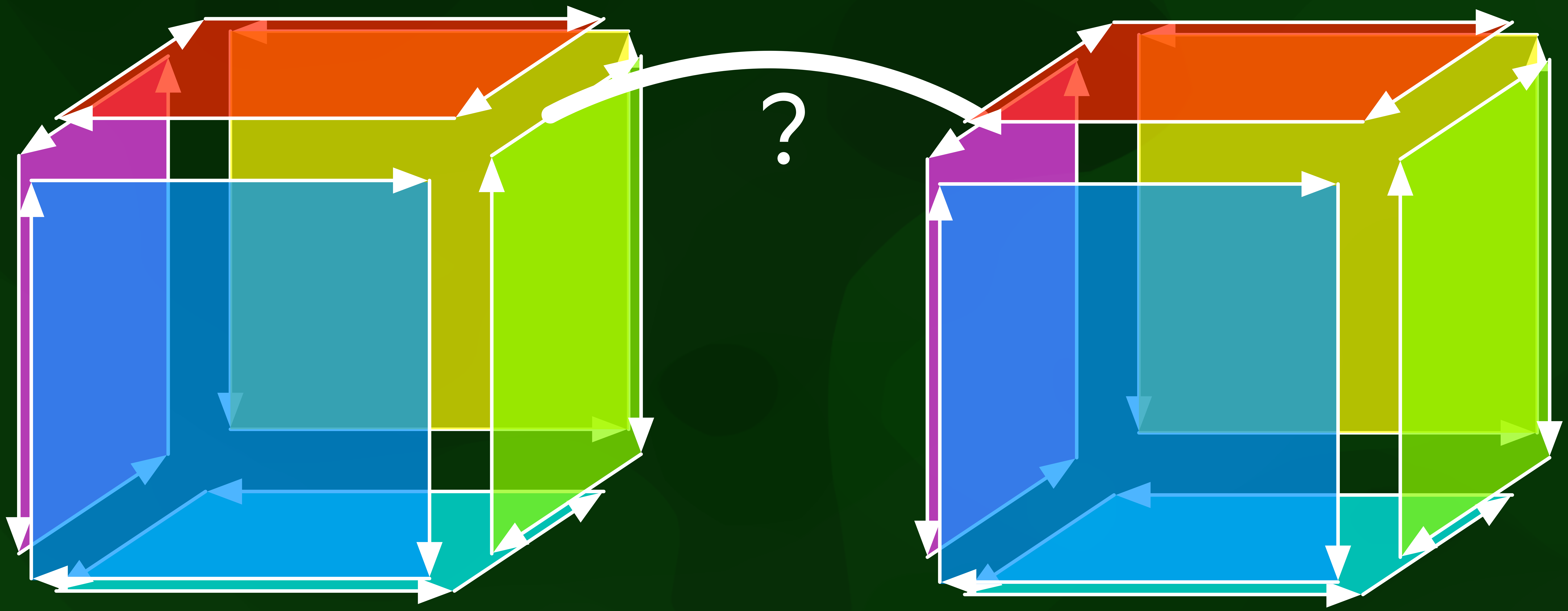
2D data structure

So far… (3D through b-rep)



links between 0D-2D elements

Links between 3D elements?



Drawbacks of b-rep approach

- Difficult to store:
  - Multiple volumes
  - Holes (2D and 3D)
  - Non-manifolds

Back to Jordan-Brouwer theorem

- In 2D, the Jordan curve theorem says: a closed curve separates the plane into two parts: an interior surface and an exterior surface
- In nD, the Jordan-Brouwer theorem, which in 3D says: a closed surface separates 3D space into two parts: an interior volume and an exterior volume.

Back to Jordan-Brouwer theorem (3D)

- Problems:
  - Holes: one more exterior per hole
  - Multiple volumes: one more interior per extra volume
  - Non-2-manifold: possibly one more interior per point in non-manifold part (depending on orientation)
- Note: can be fixed with bridges or by using a particular orientation

… but still links between 3D objects are missing

What are g-maps / c-maps?

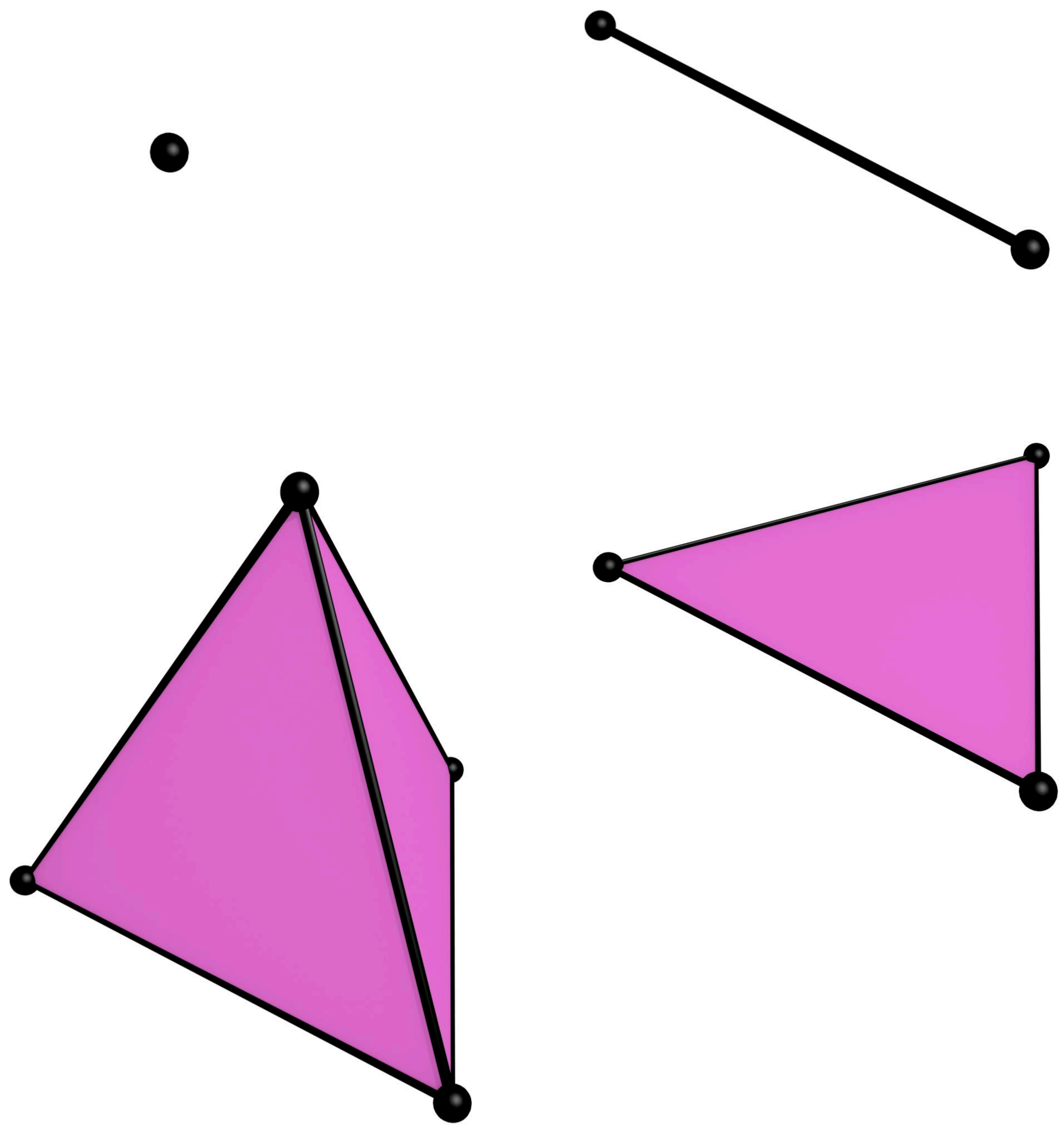
- In short, nD data structures, that is data structures that can store:
  - objects of any dimension
  - and the topological relationships between them
- c-maps: generalisation of half-edge to nD -> 2D c-maps is half-edge
- g-maps: c-maps where each element is split into two to avoid oriented edges

Why g-maps / c-maps?

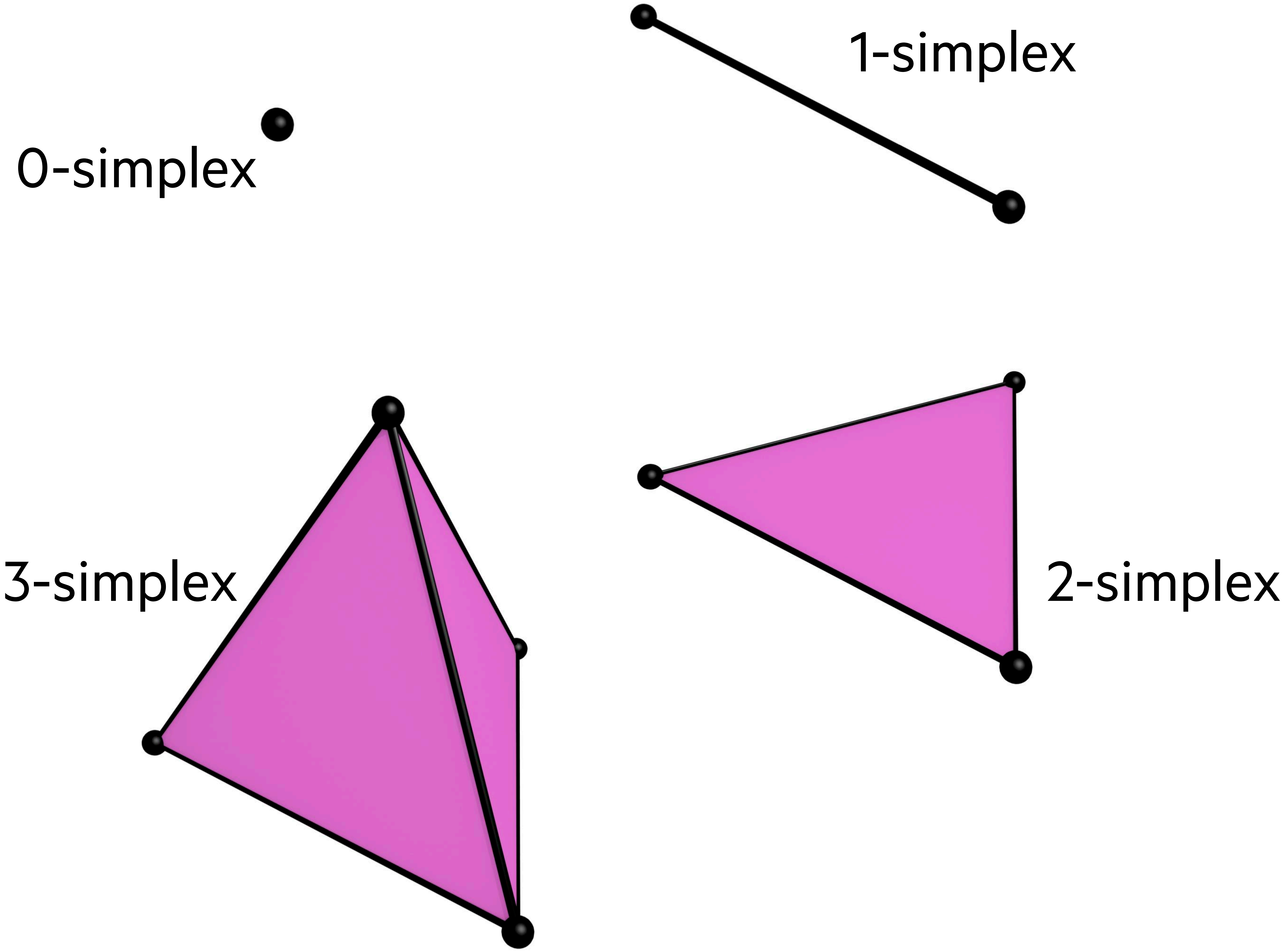
- In short:
  - Possibility to store links between nD elements (including 3D)
  - With g-maps: no orientation issues (e.g. in construction or with dangling/invalid objects)

Small background

Simplex



Simplex



Simplex properties

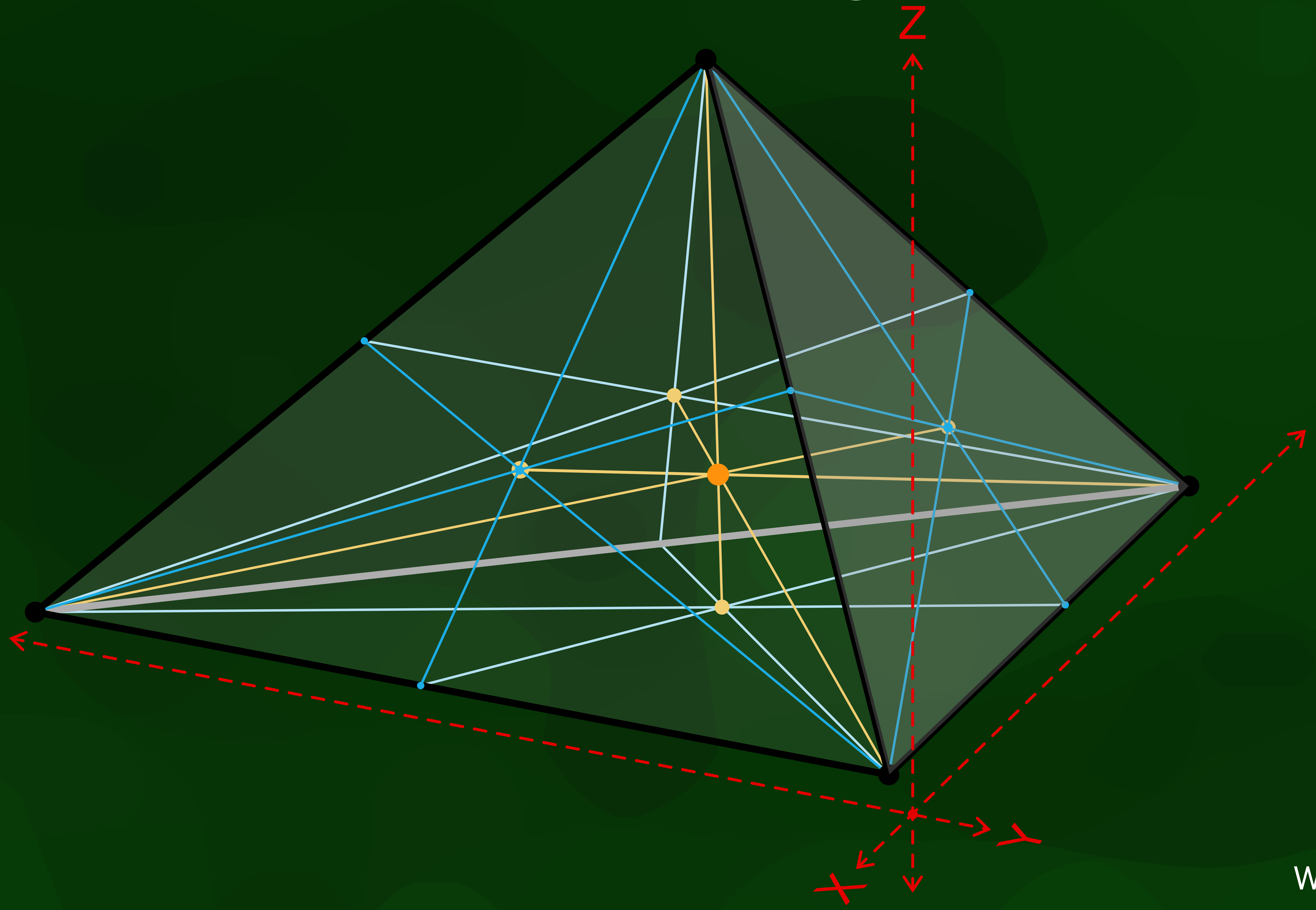
- An n-simplex in n-dimensional space:
  - is bounded by n+1 (n-1)-simplices
  - can have n+1 adjacent n-simplices, each of which shares a (n-1)-simplex on their common boundary

➔ Two adjacent n-simplices share all their vertices except for one

Cells

- 0-cell: vertex
- 1-cell: edge
- 2-cell: polygon
- 3-cell: polyhedron
- …

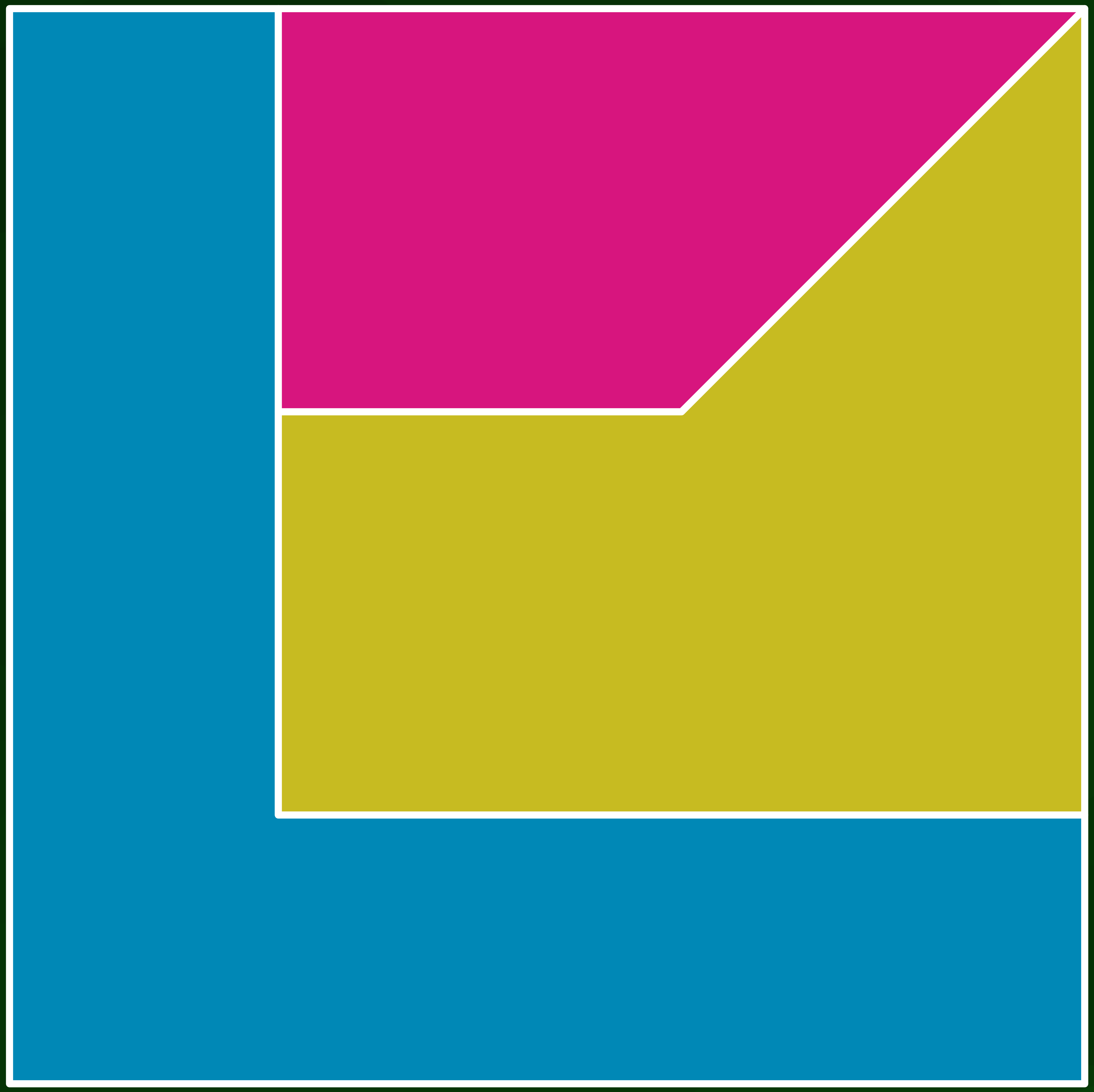
Barycentric triangulation



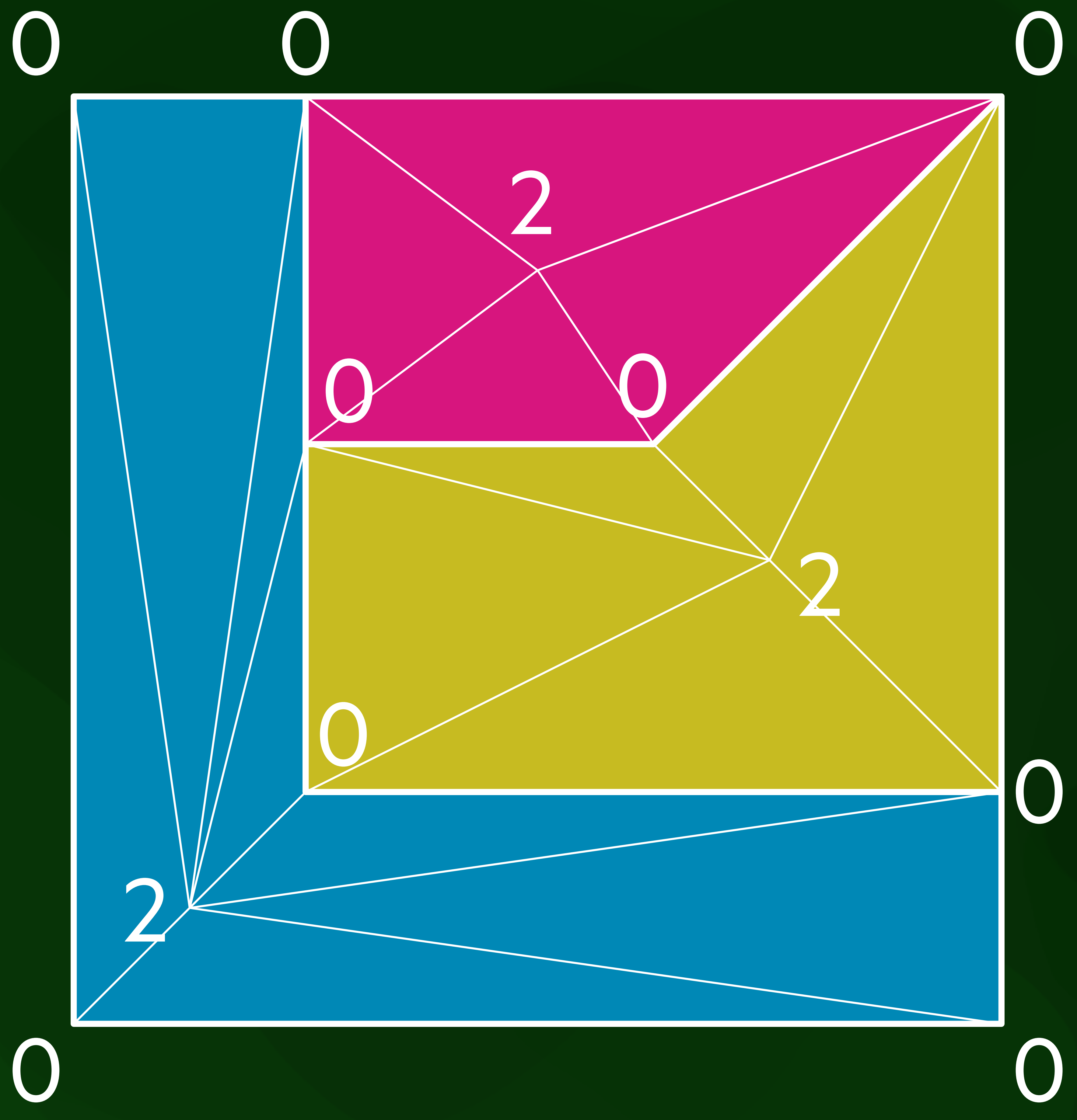
Building g-maps / c-maps

- nD c-maps (n-c-maps): barycentric triangulation n-cells from 2-cells
- nD g-maps (n-g-maps): barycentric triangulation of n-cells from 1-cells
- … where
  - n-simplices are called **darts** and have vertices that are linked to elements of a certain dimension, and
- when only 0-cells have a location in space and linear geometries are assumed between them, it is known as a **linear cell complex**.

Building g-maps / c-maps

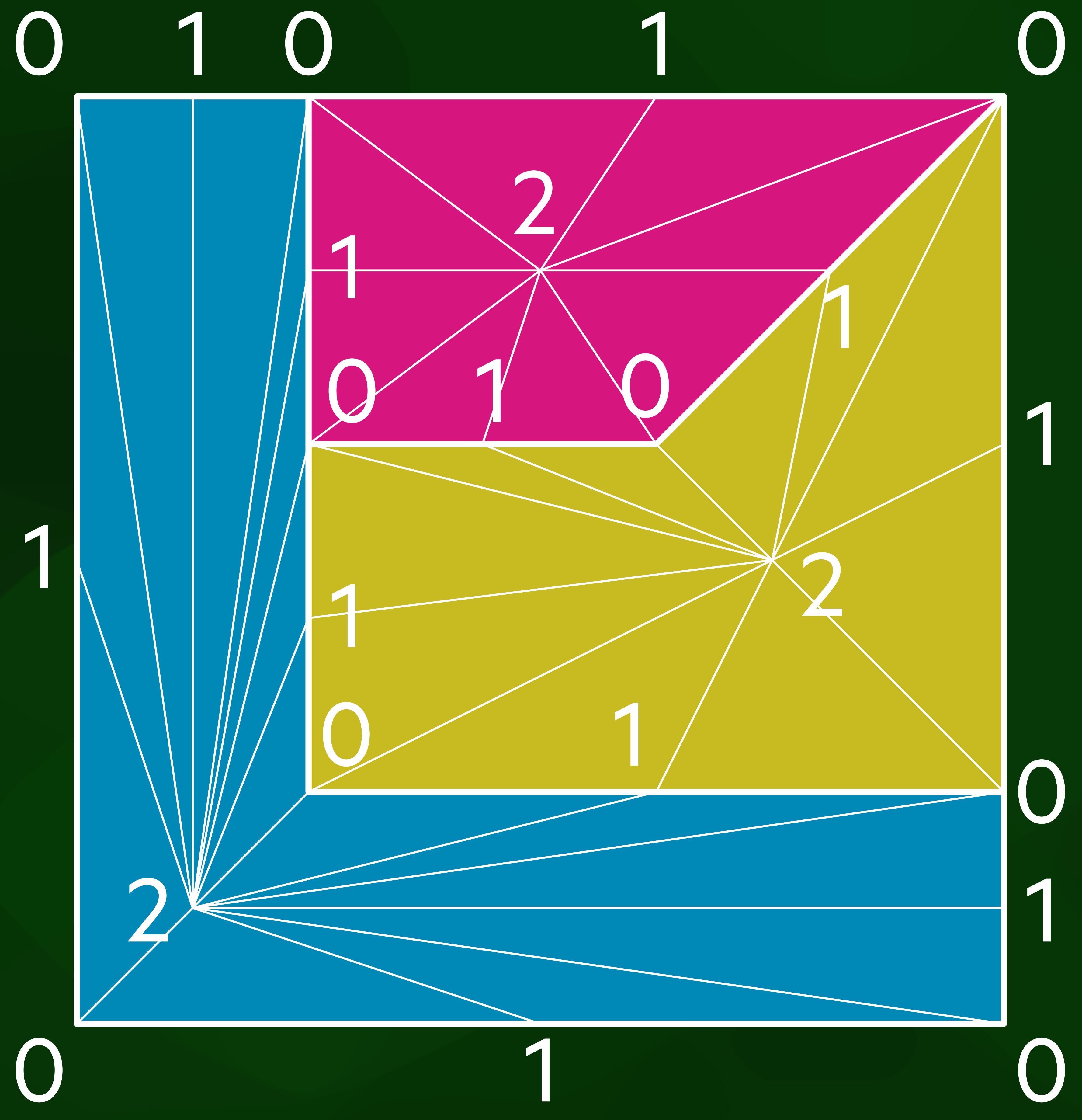


Building g-maps / c-maps



combinatorial map

(c-map)



generalised map

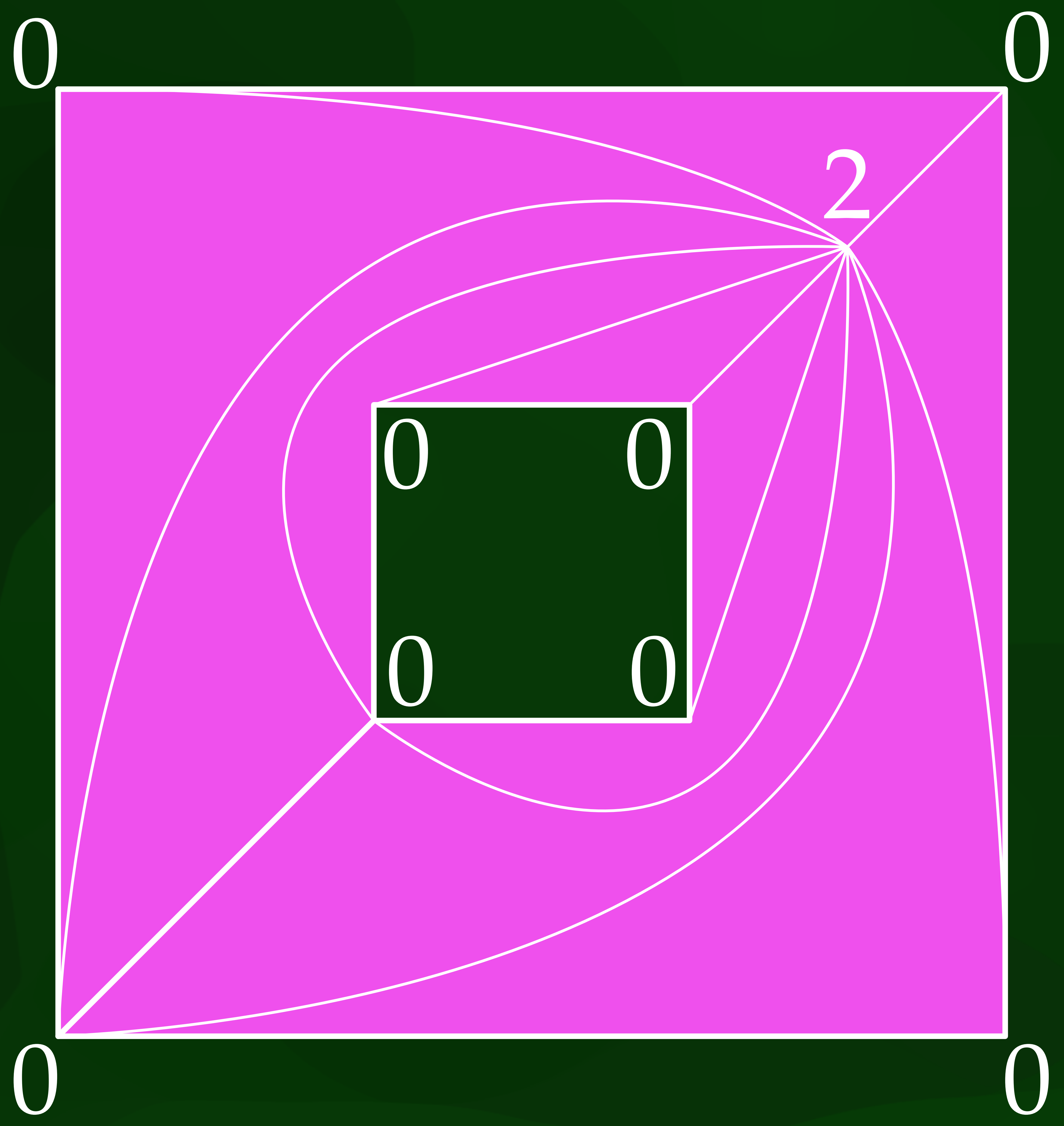
(g-map)

Intuitive meaning of a dart

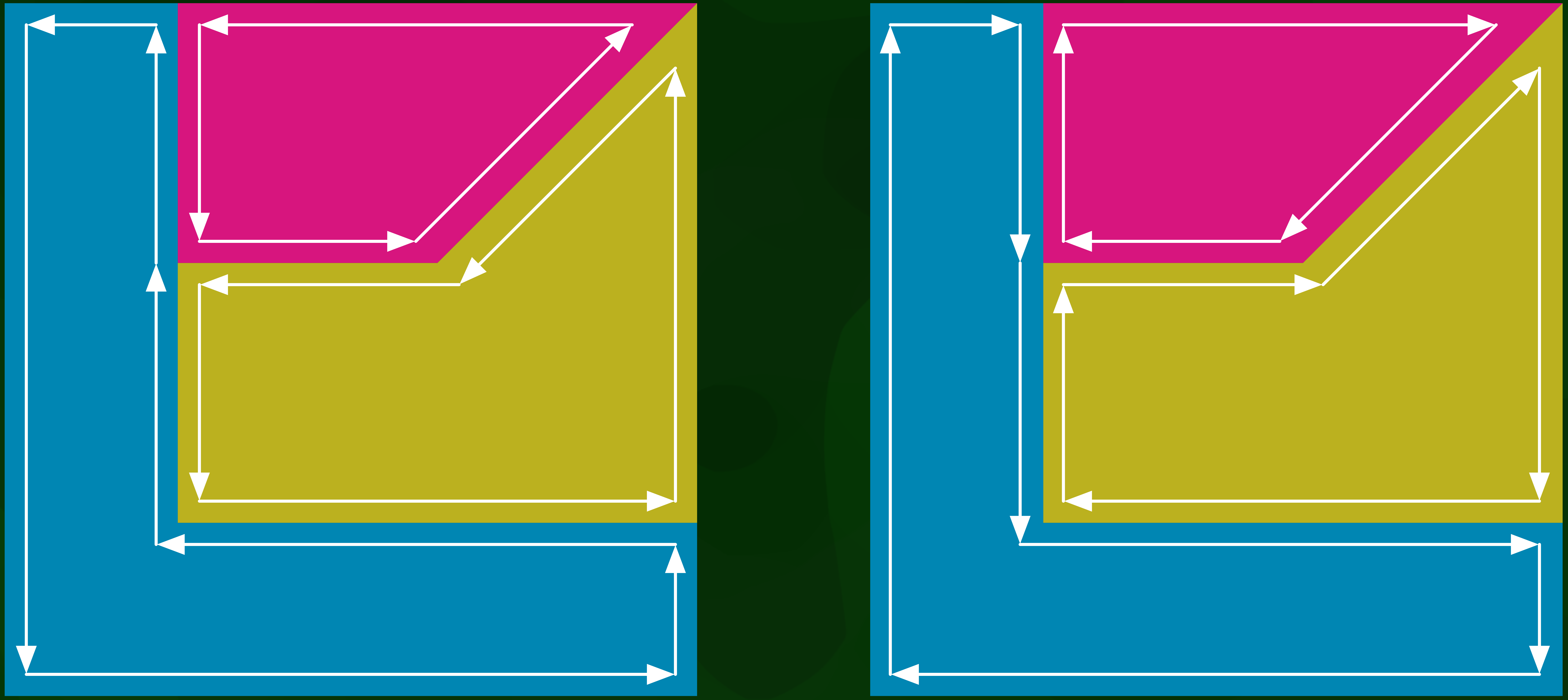
- Informally:
  - a generalised map dart is a unique combination of a cell of every dimension: vertex, edge, face, volume, …
  - a combinatorial map dart is a unique combination of a cell of every dimension from one upwards: edge, face, volume, …

Intuitive meaning of a dart

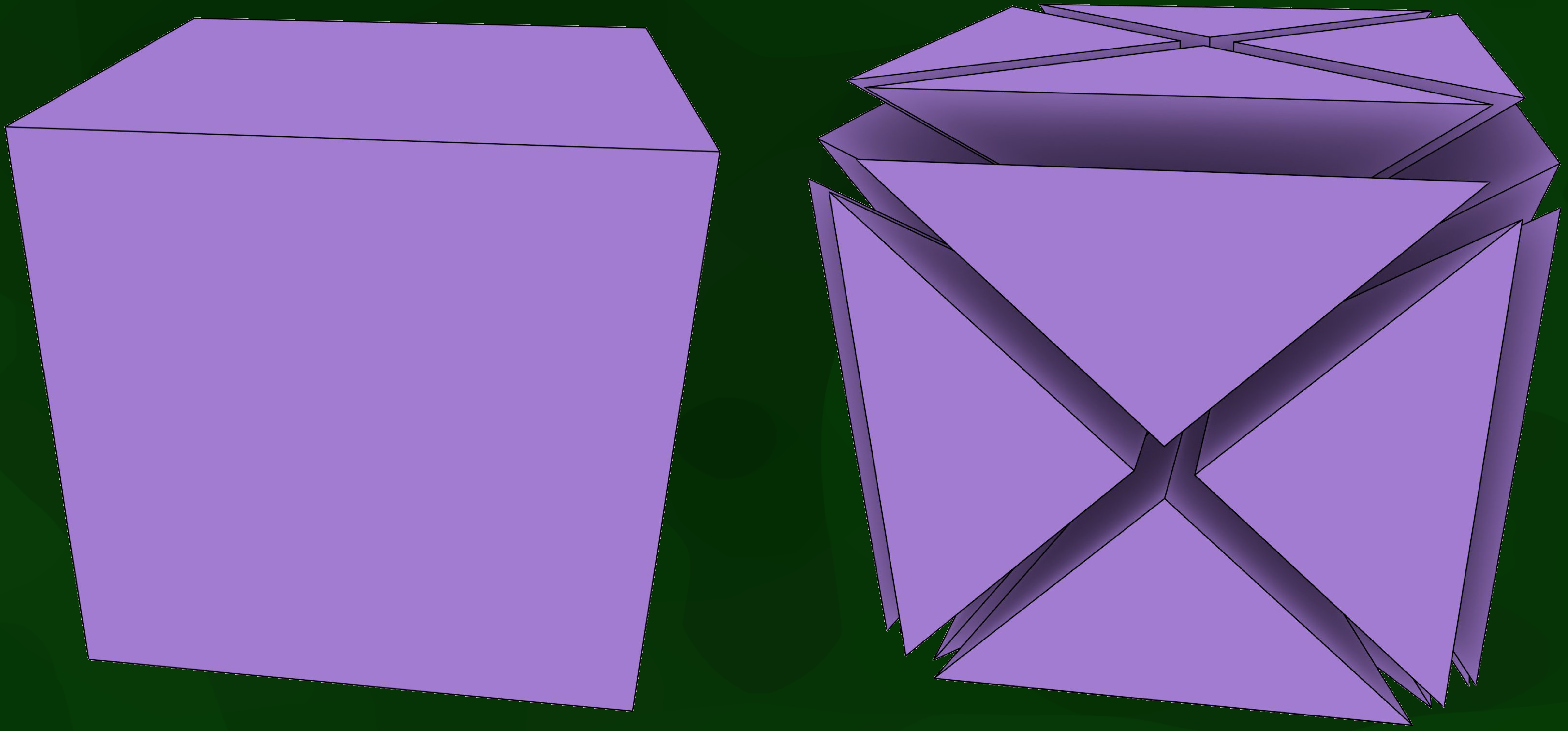
- Why informal? only given certain conditions, e.g.
    - linear embeddings (i.e. polygons, not curved surfaces)
    - no bridge edges



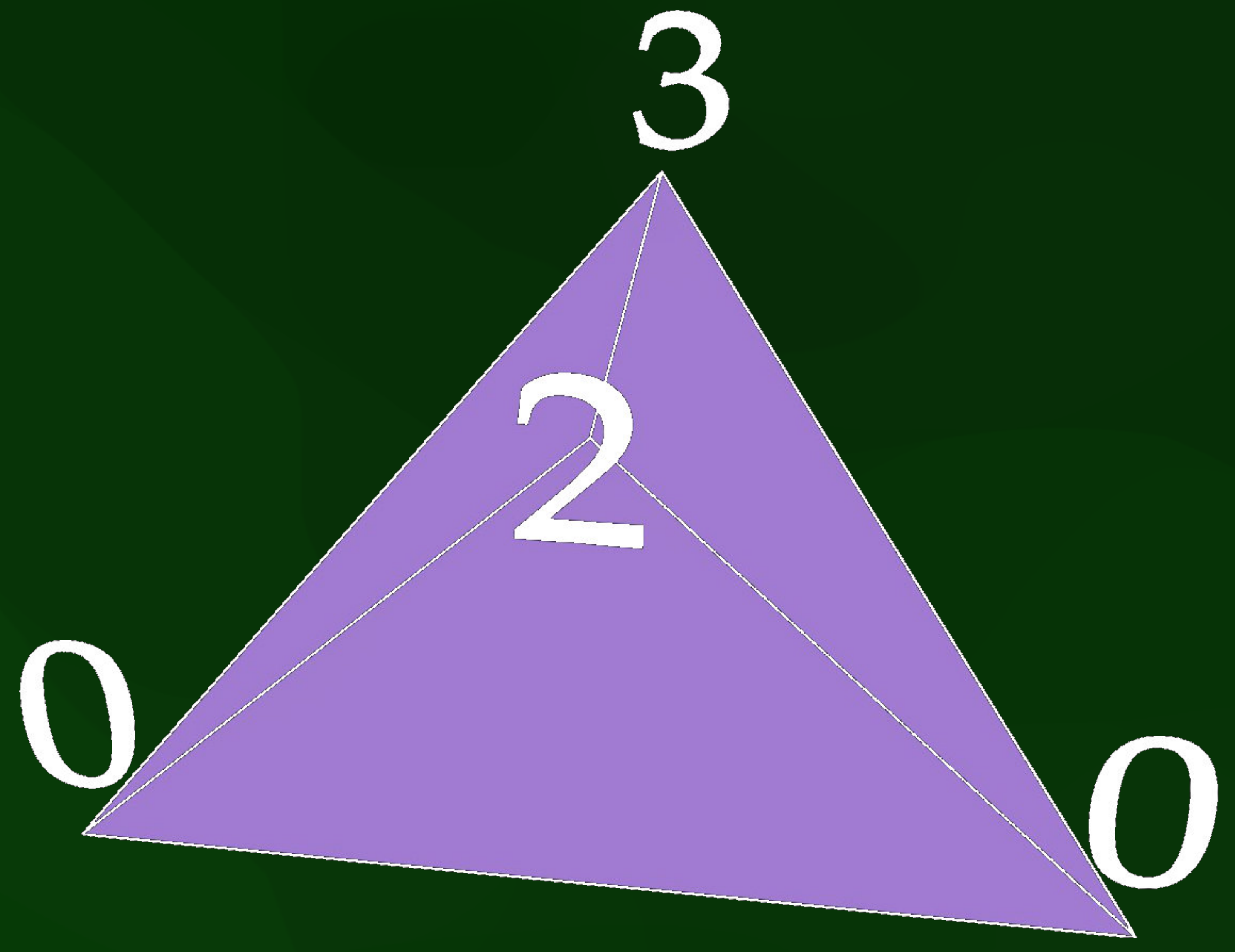
C-maps: orientation

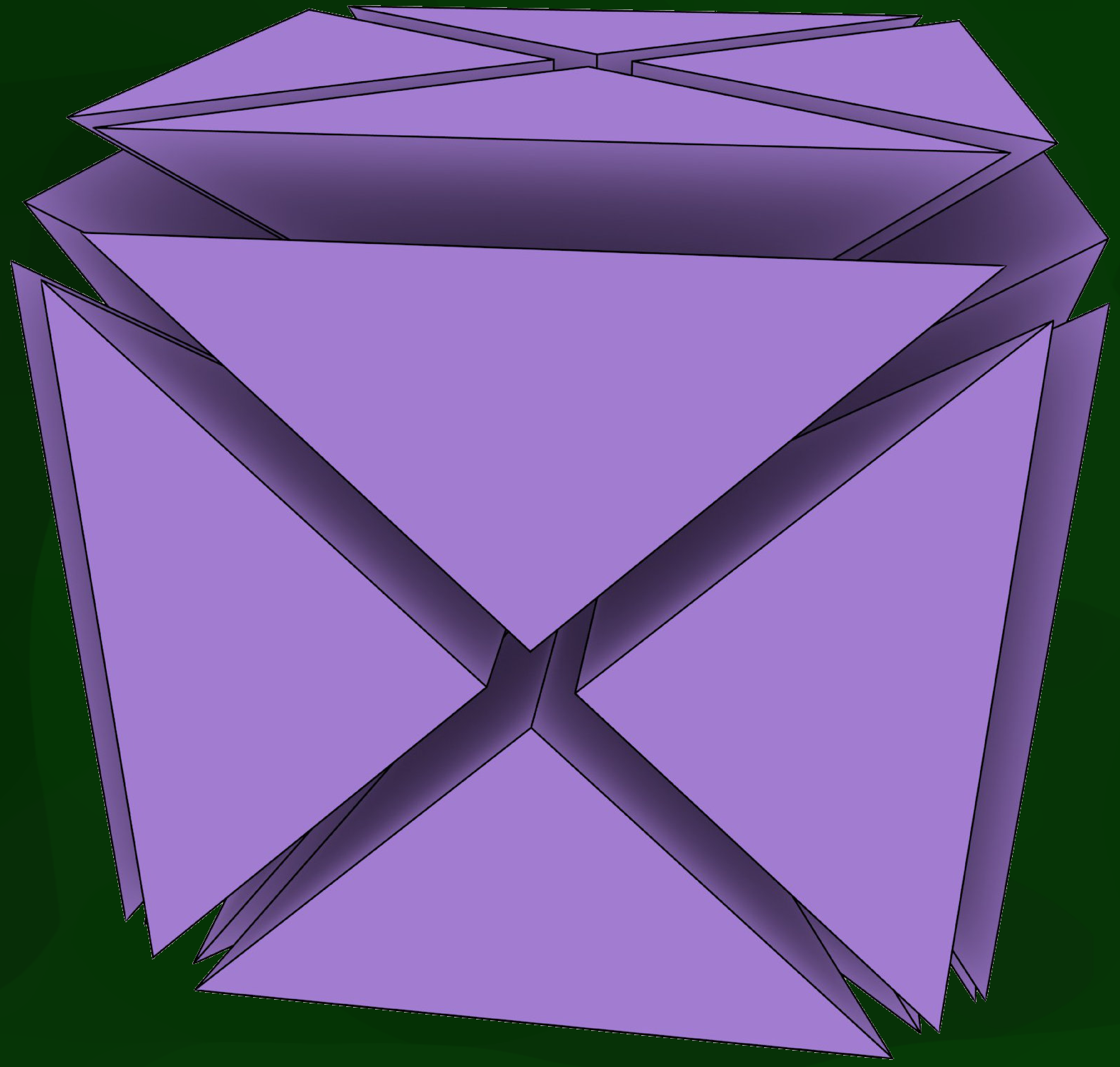


Building g-maps / c-maps



Building g-maps / c-maps





Traversing darts

- From the properties of simplices: a dart d in an nD combinatorial/generalised map has n+1 adjacent darts, each of which shares all but one of the vertices of d.

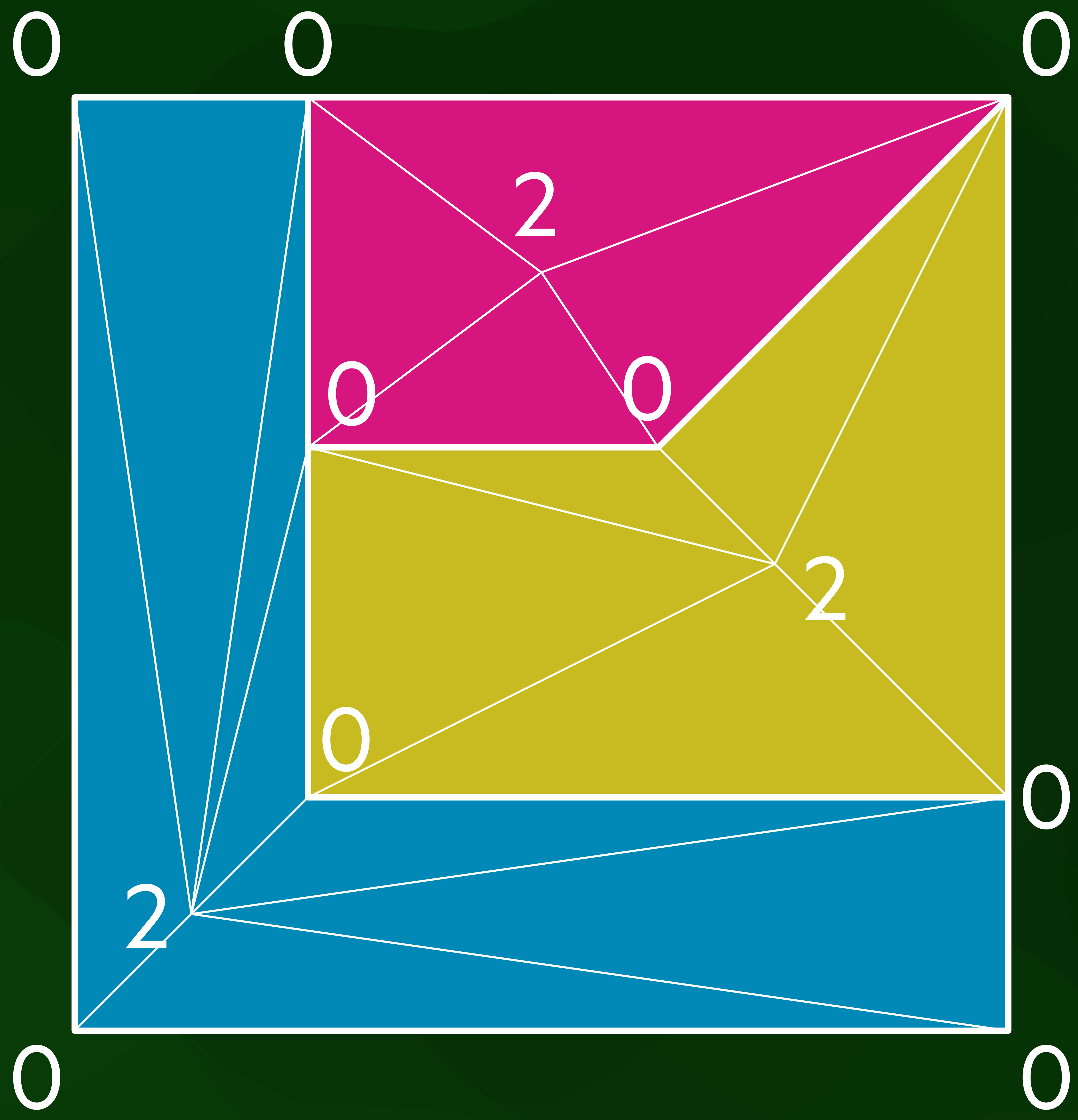
➔ Informally: Two adjacent darts share all of their cells except for one. e.g. if they differ in their edge, they share their vertex, face, volume, etc.

Traversing darts

- The link from an i-dimensional vertex of a dart d to the (other) i-dimensional vertex of its adjacent neighbour is called:
  - α_i in a generalised map
  - β_i in a combinatorial map
- Informally, it means switching the i-cell of d for the i-cell of its neighbour

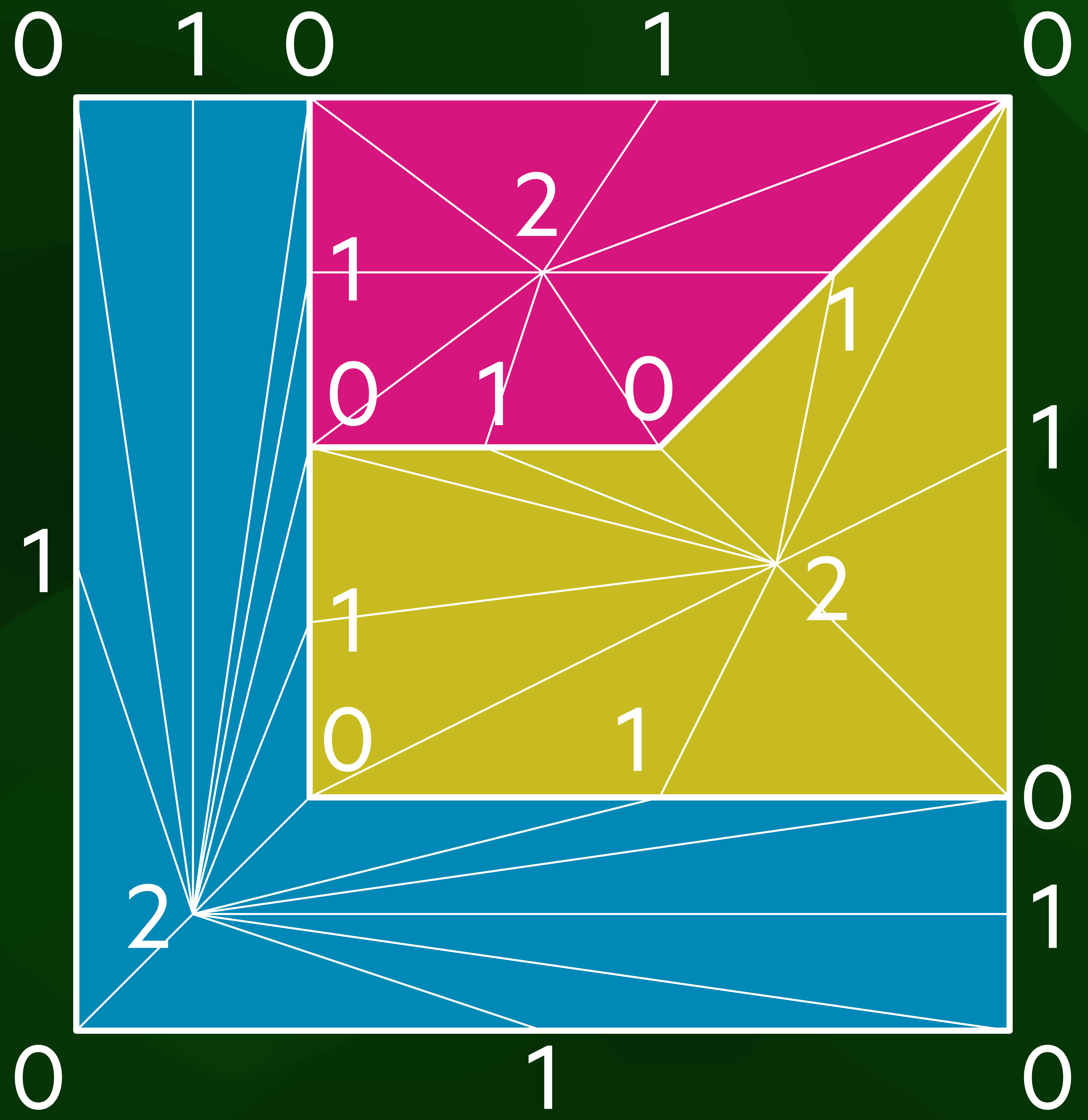
Traversing darts

- for all i, α_i is an involution
- for β > 1, β_i is also an involution
- but for for β = 1, β_i is a permutation



combinatorial map

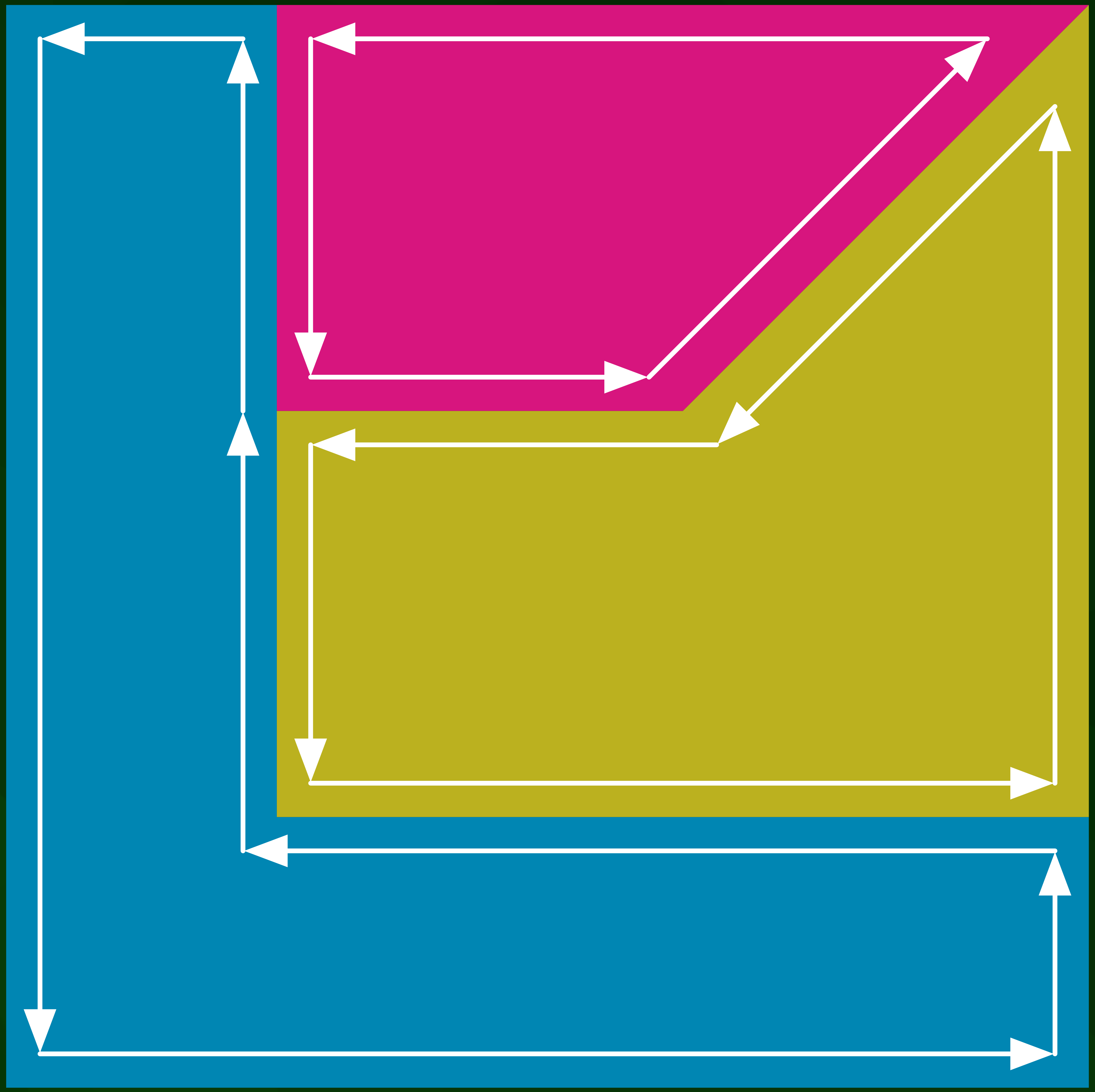
(c-map)



generalised map

(g-map)

C-maps vs. half-edge



- β_1 = next
- β_1^{-1} = prev
- β_2 = twin

Storage

- In a generalised map, a list of darts of the form:
  - $d = [[\alpha_0(d), \alpha_1(d), \alpha_2(d), ...],$

    $[a_0, a_1, a_2, ...]]$
  - where each α is a link (ID, pointer) to another dart, and

- In a combinatorial map, a list of darts of the form:
  - $d = [[\beta_1^{-1}, \beta_1, \beta_2, ...],$

    $[a_0, a_1, a_2, ...]]$
  - where each β is a link (ID, pointer) to another dart, and

- each a is an optional link to a data structure with the attributes for the i-cell of d, including the coordinates in a_0.

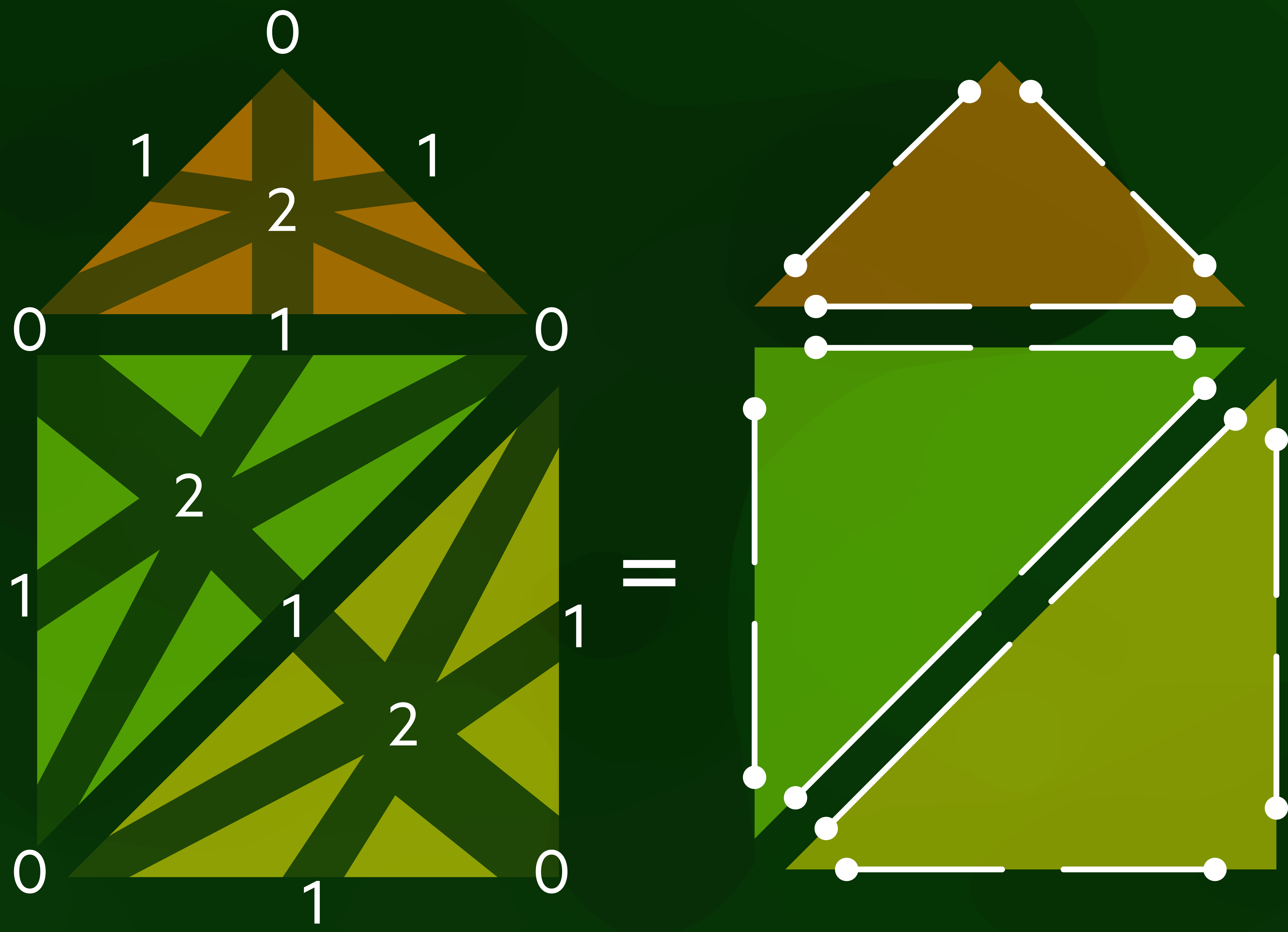
In practice? CGAL

https://doc.cgal.org/latest/Combinatorial_map/index.html#Chapter_Combinatorial_Maps

https://doc.cgal.org/latest/Generalized_map/index.html#Chapter_Generalized_Maps

https://doc.cgal.org/latest/Linear_cell_complex/index.html#Chapter_Linear_Cell_Complex

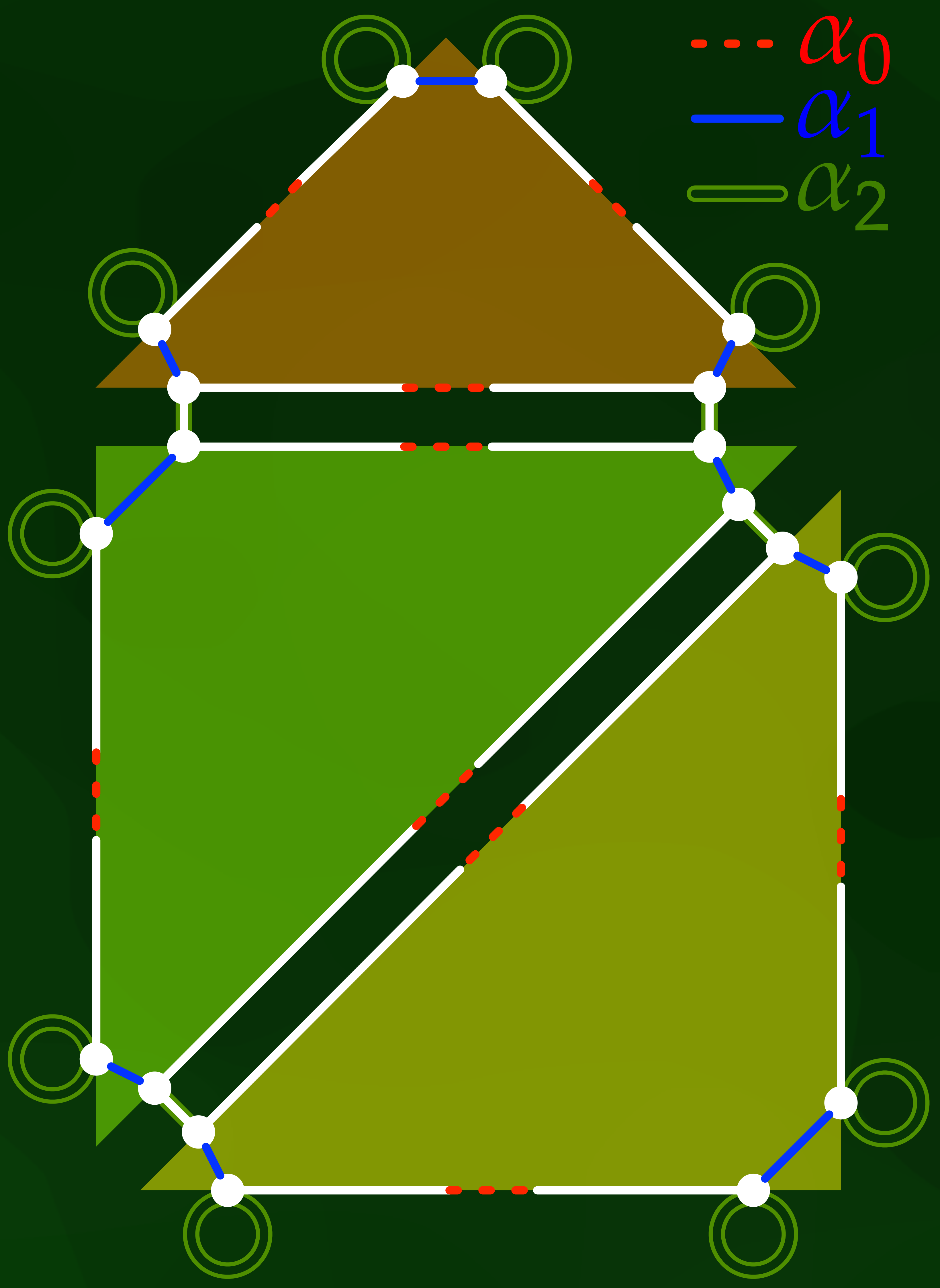
Simpler representation

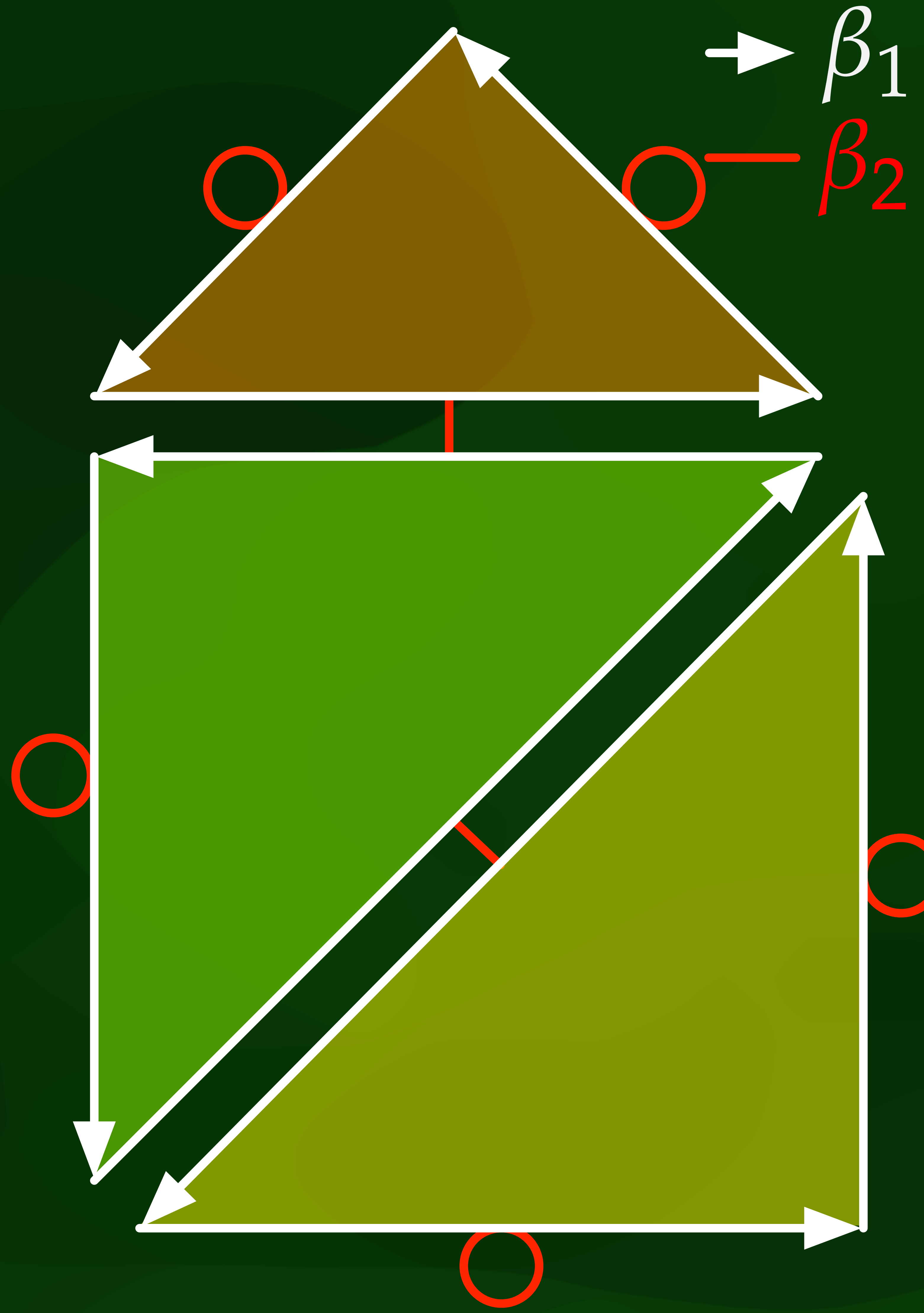


Dart as simplex

Dart as vertex-edge-face-...

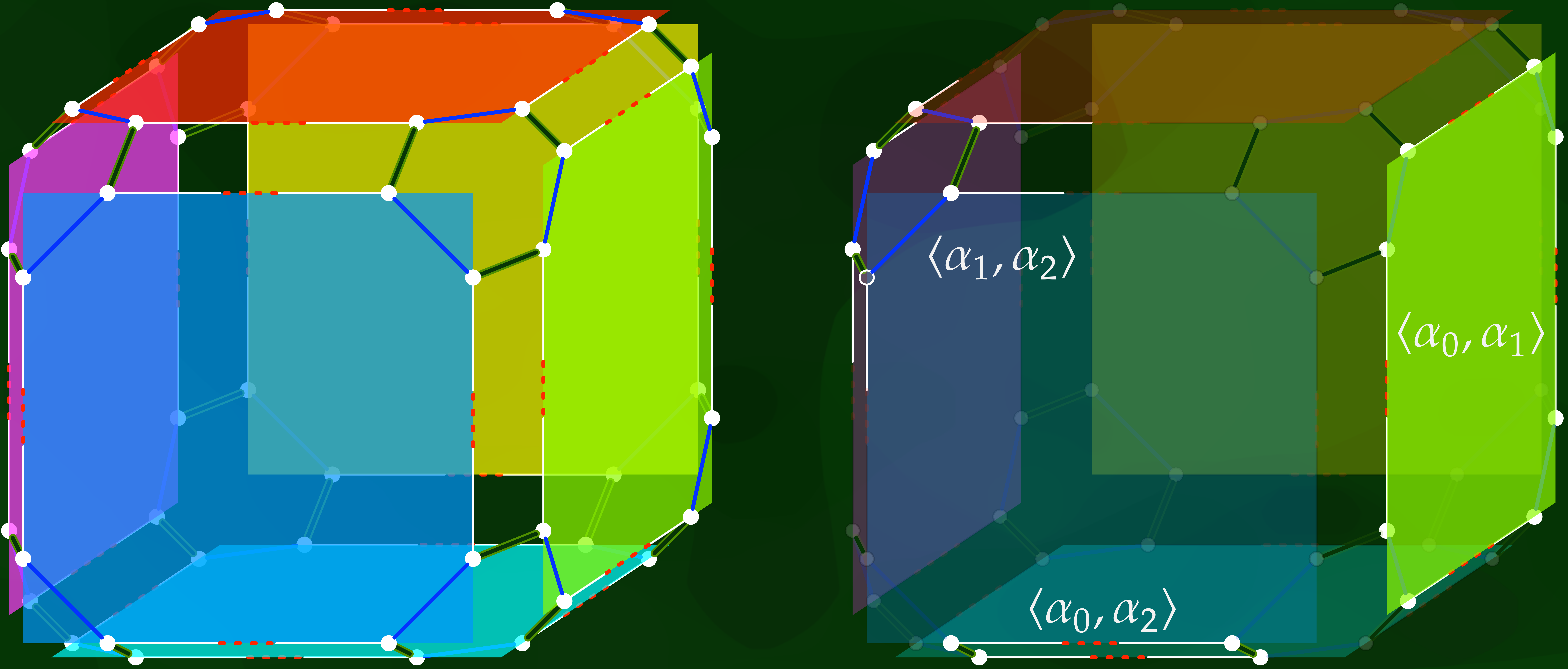
Involutions and permutations





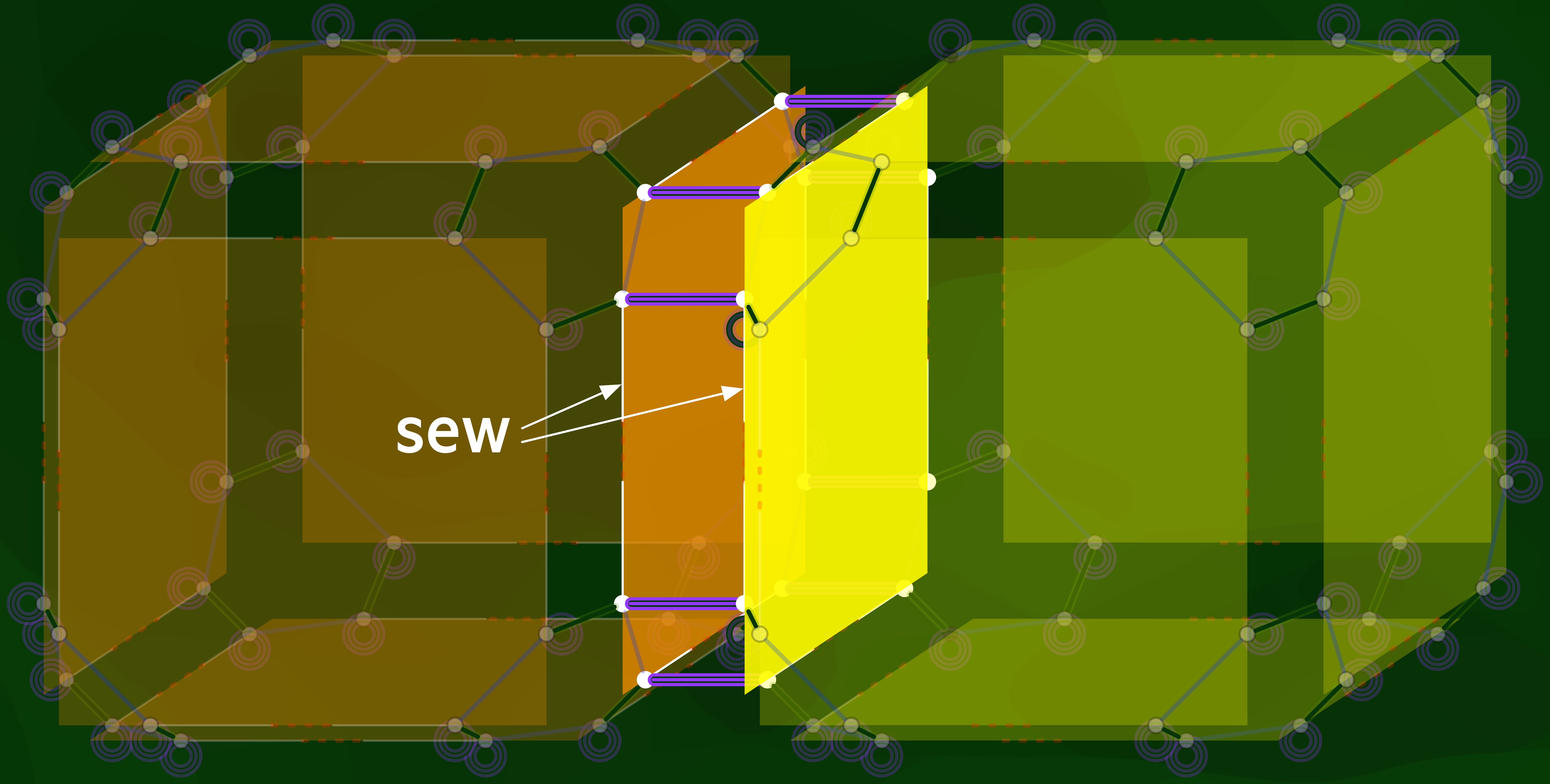
How to build more complex cmaps / gmaps?

Orbits



$\langle \alpha_1, \alpha_2 \rangle$
$\langle \alpha_0, \alpha_1 \rangle$
$\langle \alpha_0, \alpha_2 \rangle$

Sewing (3D)



Variations

- More complex embeddings for non-linear geometries, e.g. storing control points in edges or surfaces
- Storing non-manifolds with chains of maps

What to do next?

1. Today:
   - Continue with Homework 3
   - Go to geo1004 website and study today's lesson (3D book Chapter 8)
2. Monday: public holiday
3. Next Wednesday: final lecture on applications of 3D modelling

https://3d.bk.tudelft.nl/courses/geo1004

3D geoinformation
Department of Urbanism
Faculty of Architecture and the Built Environment
Delft University of Technology