Bits & Atoms

Computer Aided Design

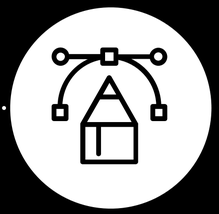
3rd Semester I 23rd of October 2017

Florian Wille & Verena Ziegler

Overview

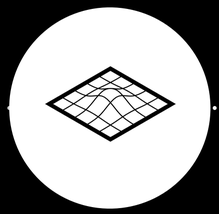
Bits & Atoms: Computer Aided Design

23.10.2017



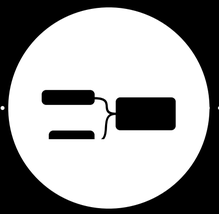
Rhino Basics

30.10.2017



Rhino to Grasshopper

6.11.2017



Parametric Design with Grasshopper

Overview

Bits & Atoms: Computer Aided Design

23.10.2017

Rhino Basics

30.10.2017

Rhino to Grasshopper

6.11.2017

Parametric Design with Grasshopper

EXERCISE 0

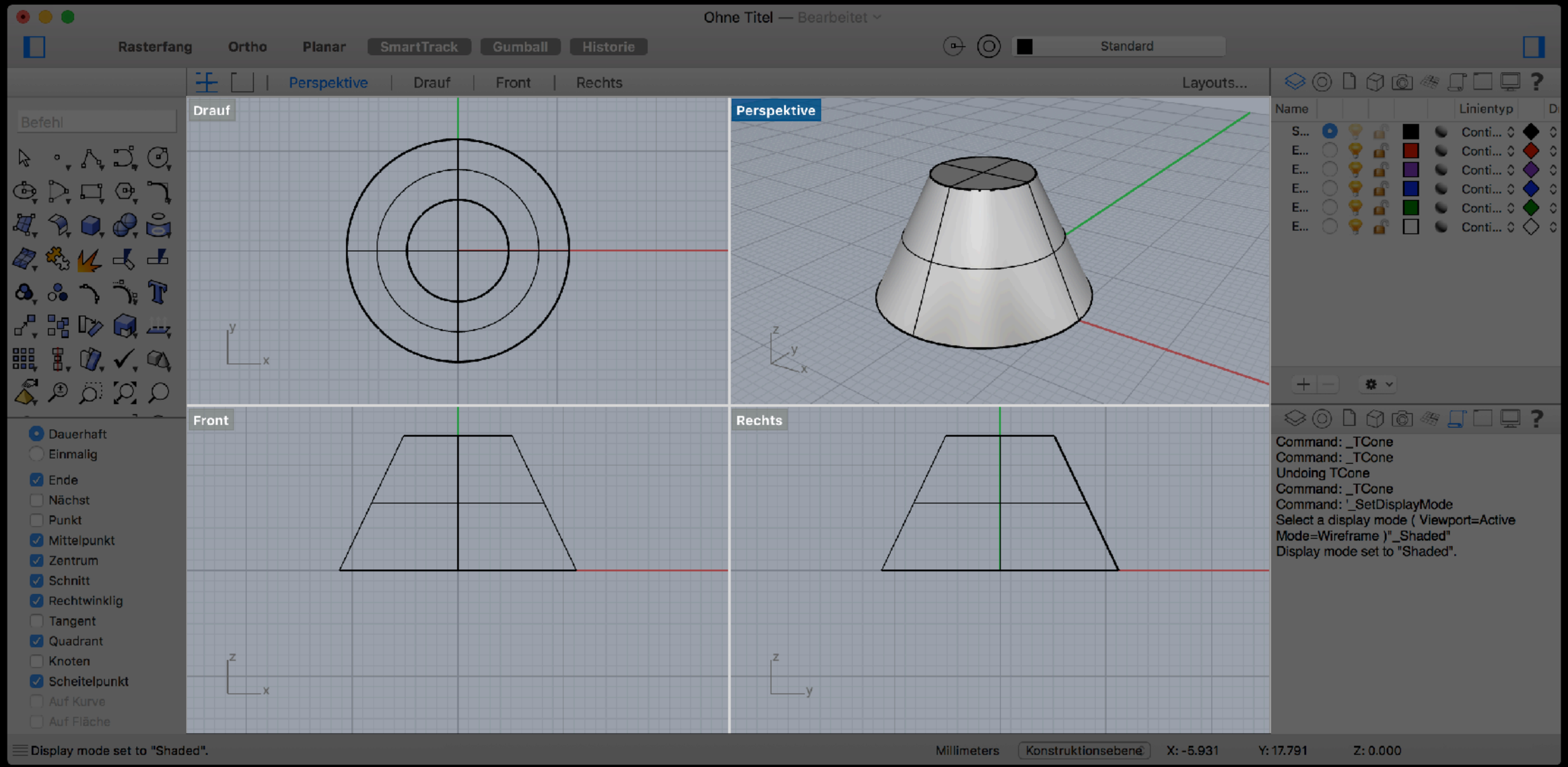
Install Rhino 3D WIP

1. **Install Rhino the Rhino3D for Mac Testversion first (valid for 90 days)**
   https://www.rhino3d.com/download/rhino-for-mac/5/evaluation
2. **Then download and install Rhino3D WIP - as this version includes Grasshopper.**
   https://www.rhino3d.com/download/rhino-for-mac/5/wip
   The WIP version requires a valid license, therefore you need to install the Testversion first.

Rhino 3D Interface

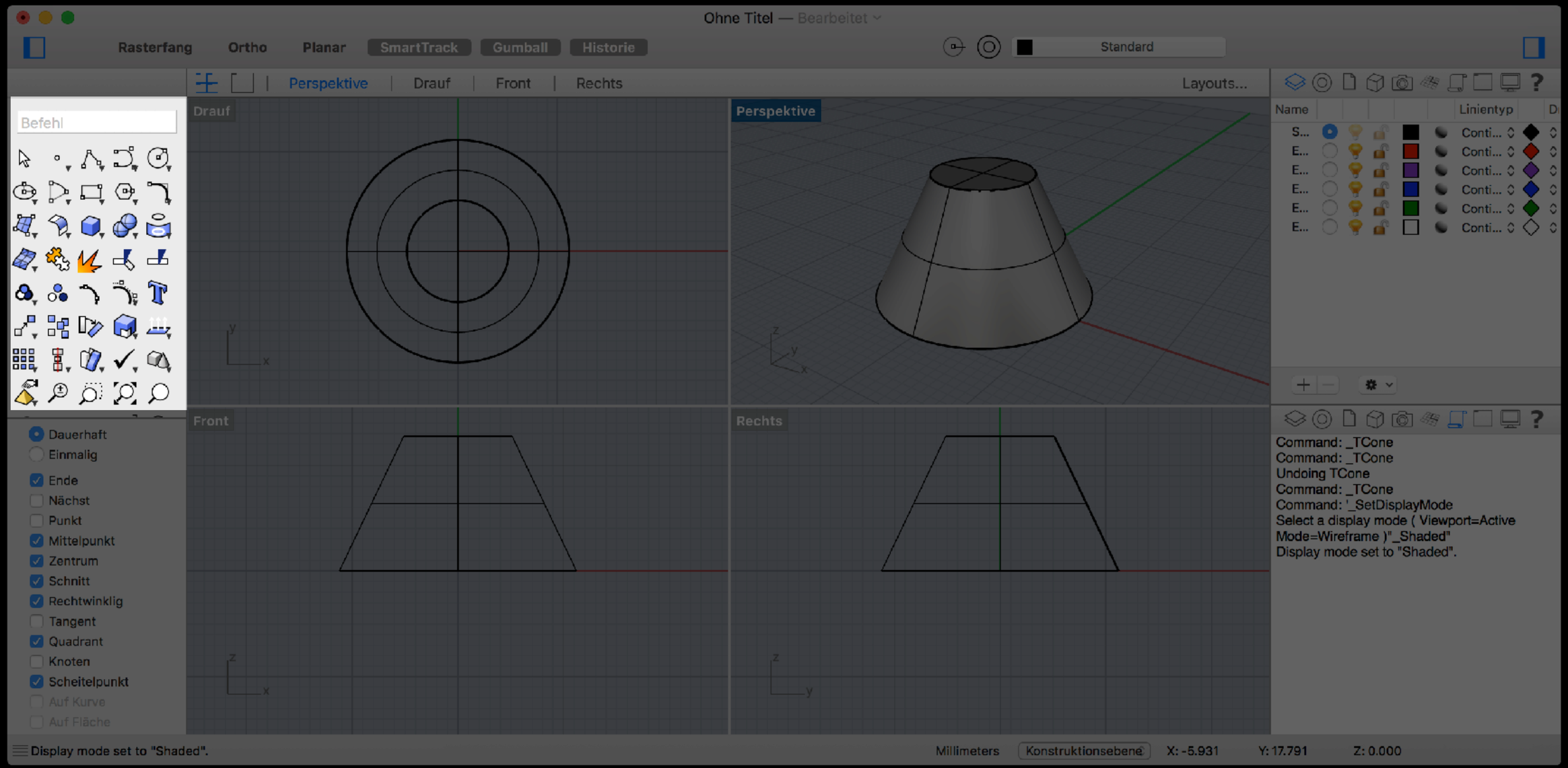
Rhino 3D

Interface - Viewports



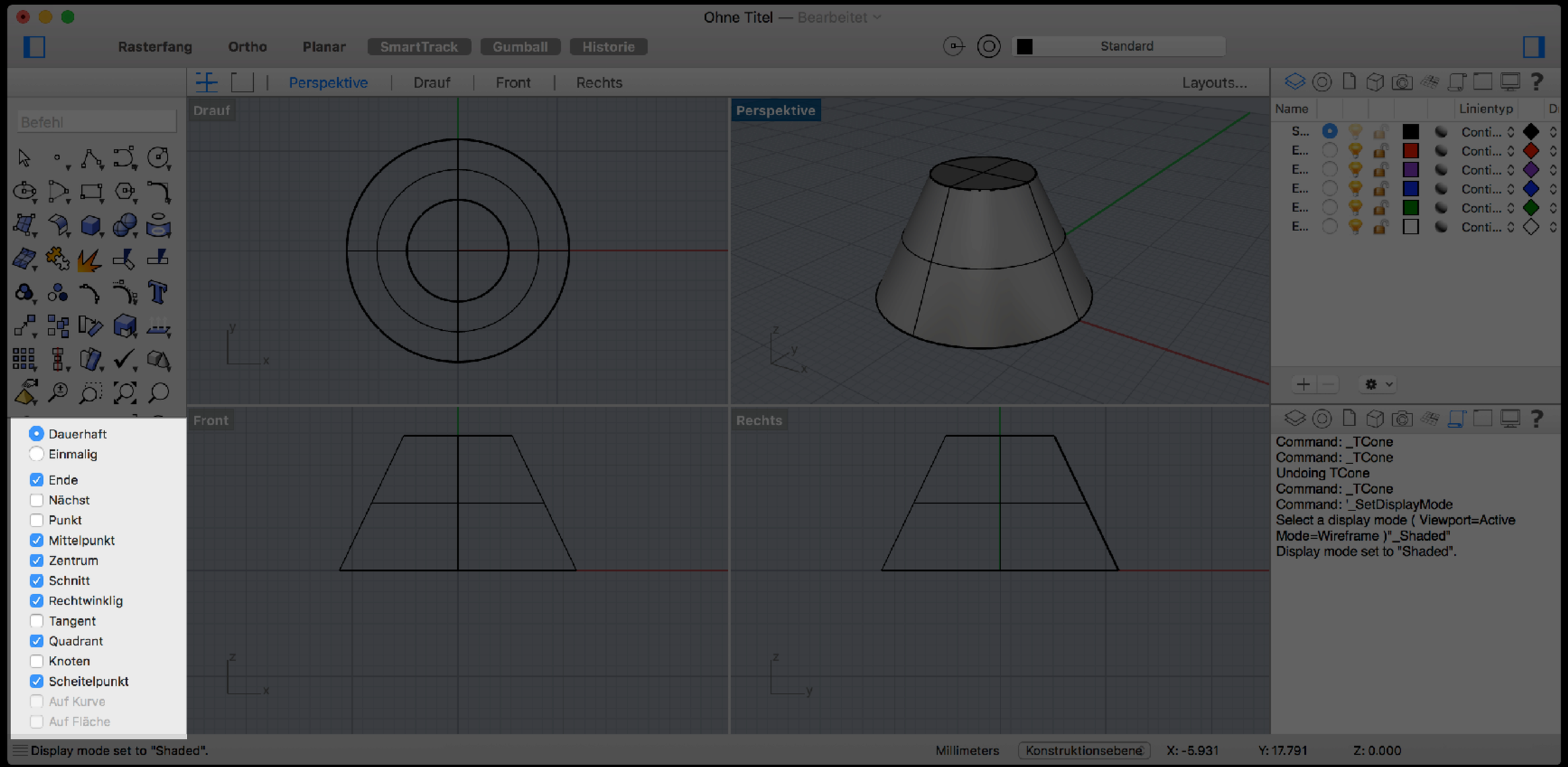
Rhino 3D

Interface - Toolbar



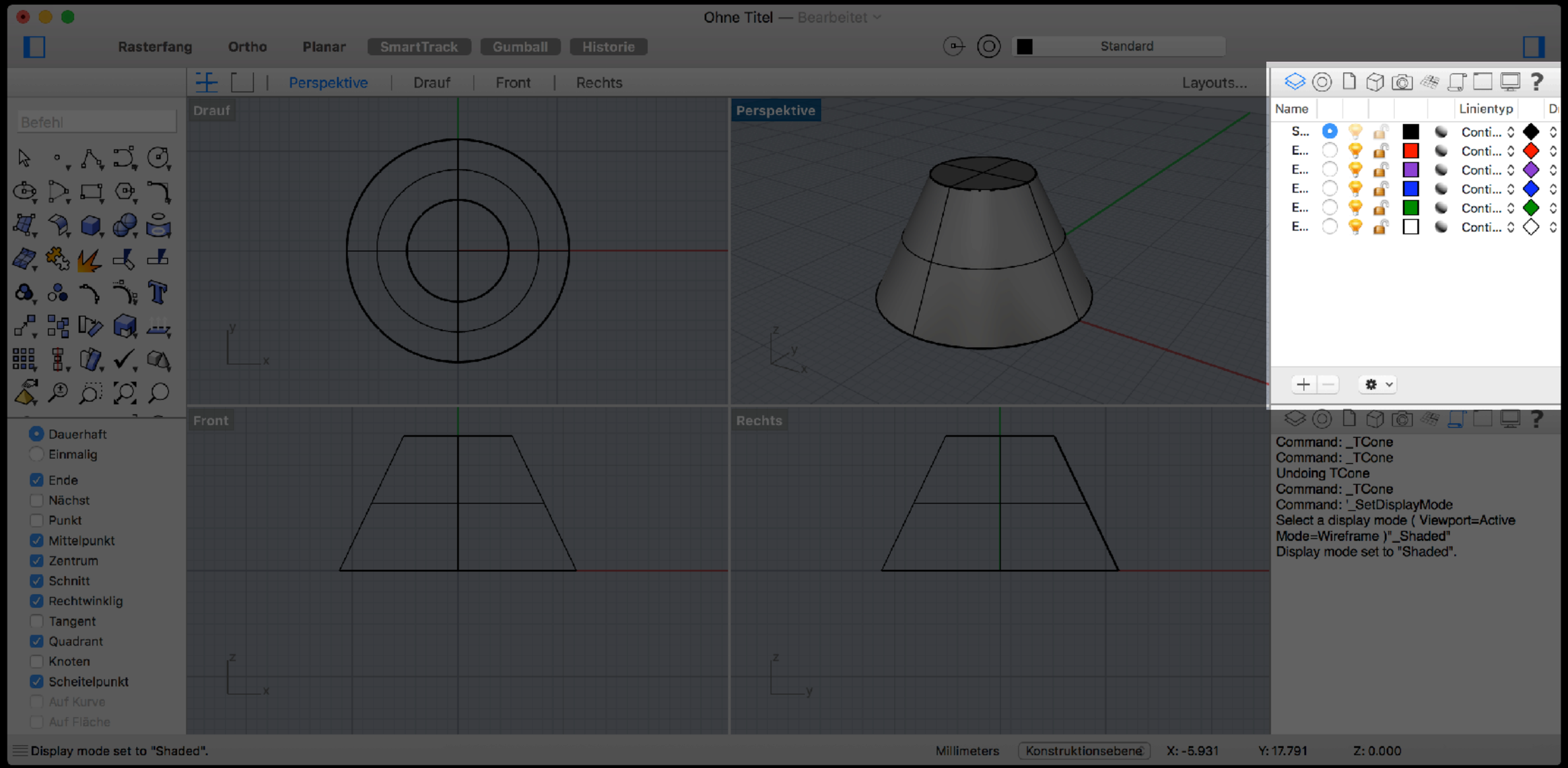
Rhino 3D

Interface - Object Snap



Rhino 3D

Interface - Layers

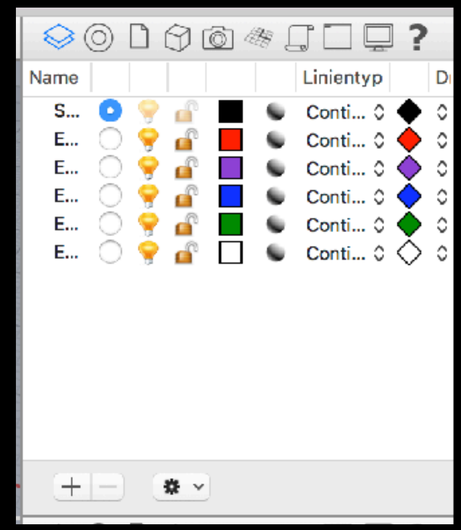


What are Nurbs?

What are Nurbs

Definition

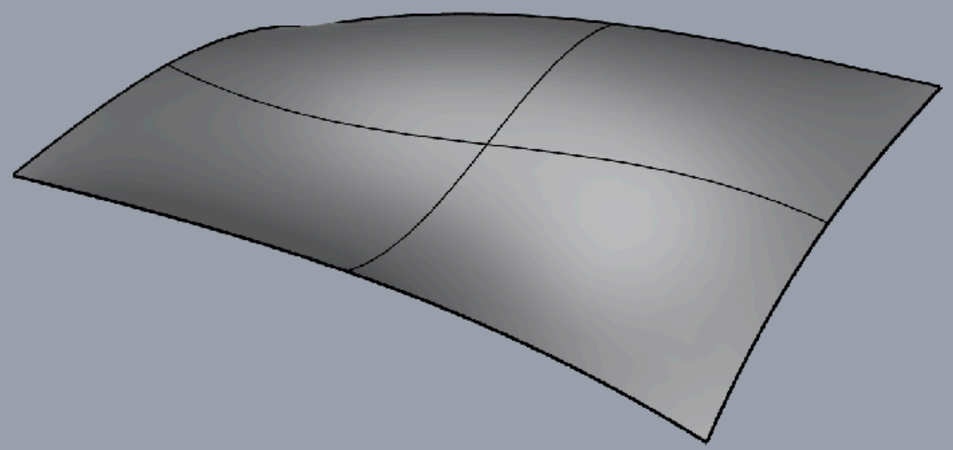
Non-uniform rational basis spline (NURBS) is a mathematical model commonly used in computer graphics for generating and representing curves and surfaces. It offers great flexibility and precision for handling both analytic (surfaces defined by common mathematical formulae) and modeled shapes. NURBS are commonly used in computer-aided design (CAD), manufacturing (CAM), and engineering (CAE) and are part of numerous industry wide standards, such as IGES, STEP, ACIS, and PHIGS. NURBS tools are also found in various 3D modeling and animation software packages.

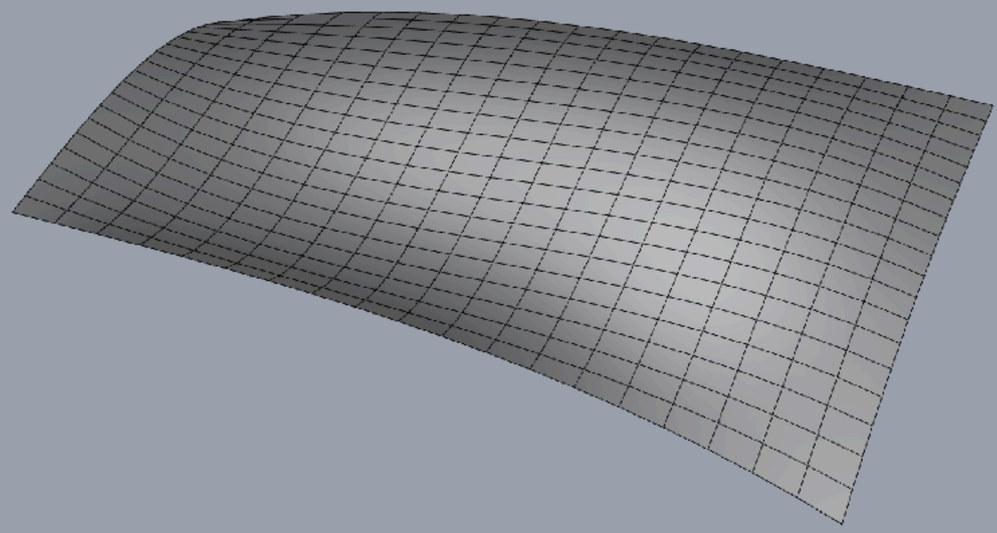


source: https://en.wikipedia.org/wiki/Non-uniform_rational_B-spline

Rhino 3D

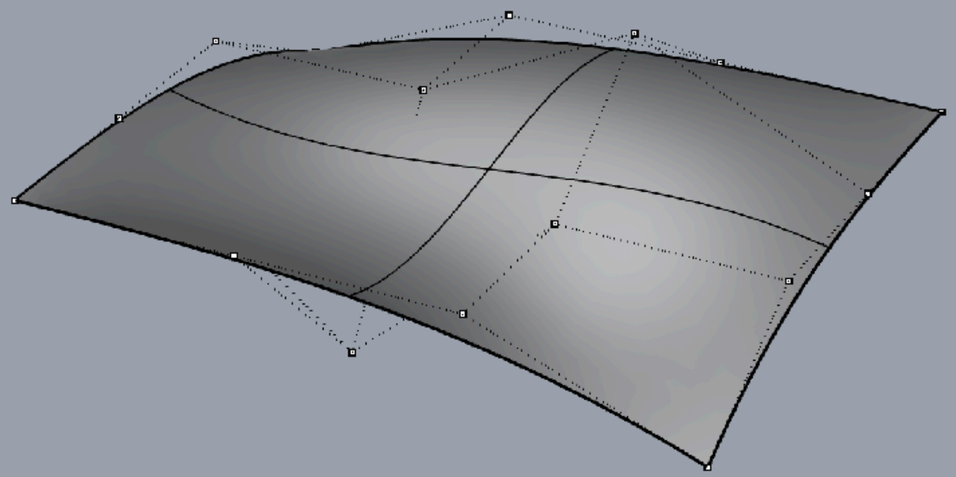
Nurbs vs. Polygons/Meshes

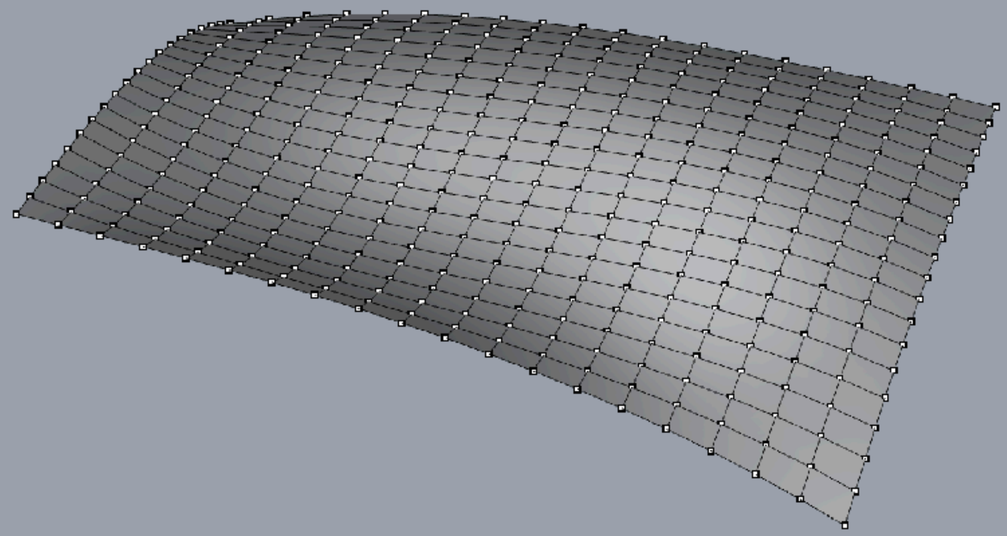




Rhino 3D

Nurbs vs. Polygons/Meshes

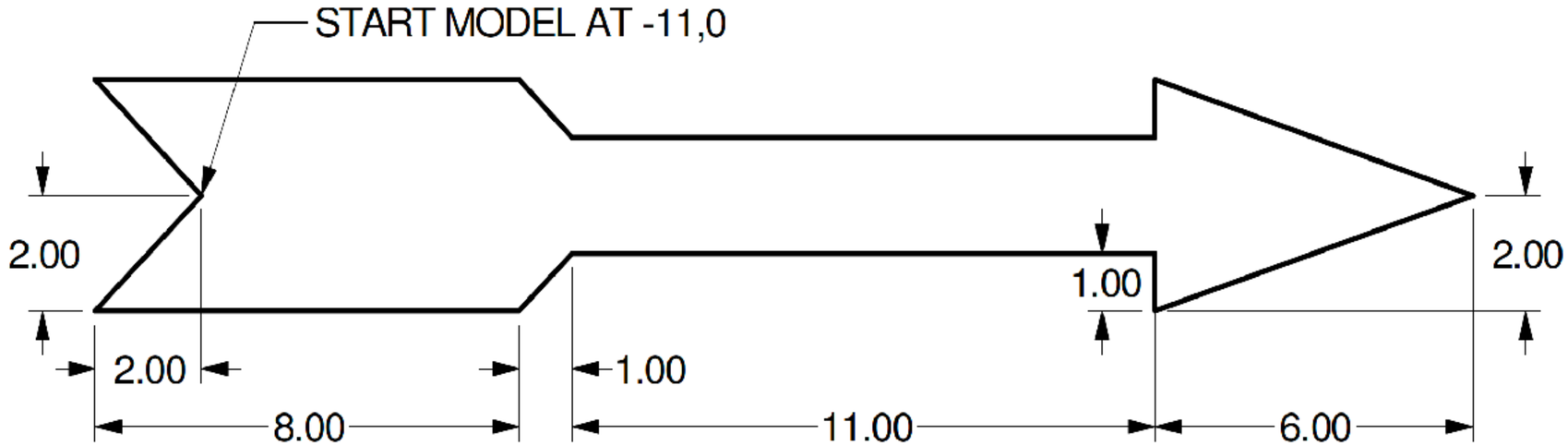




Create and Edit 2D Geometry

EXERCISE01

2D & 3D Creation and Editing



create & edit 2D:

_line, _circle, _curve, _arc

_move / _mirror / _trim / _extend

_EditPtOn

Use Osnap / Objektfang

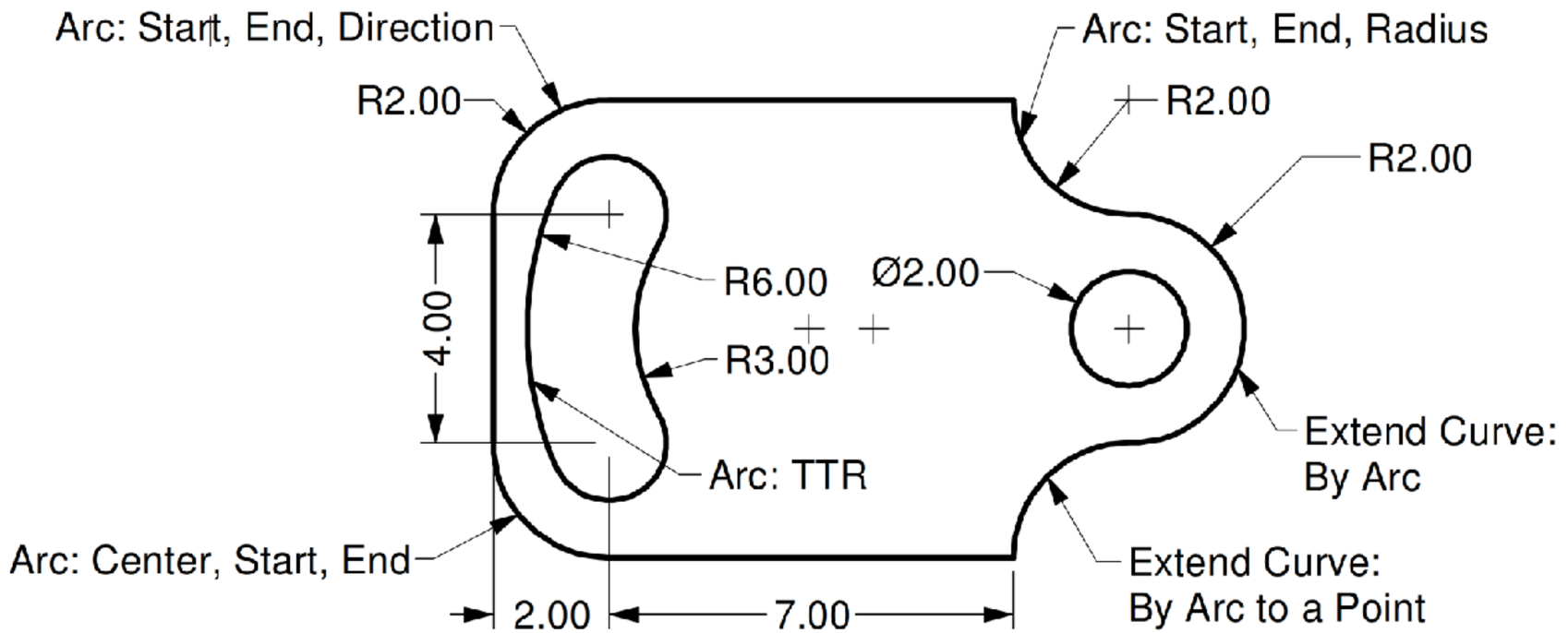
Use Gridsnap / Rasterfang

create & edit 3D:

_ExtrudeCrv

EXERCISE02

Circles and Arcs



create & edit 2D:

_line, _circle, _curve, _arc

_move / _mirror / _trim / _extend

_EditPtOn

Use Osnap / Objektfang

Use Gridsnap / Rasterfang

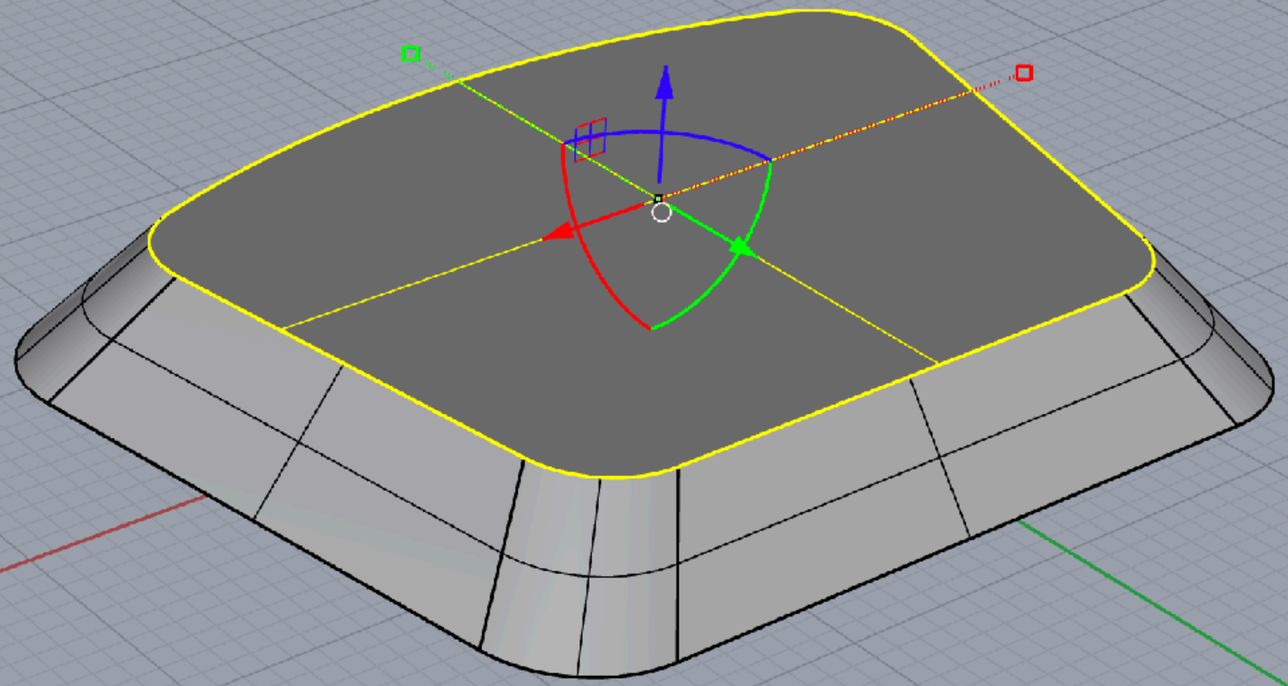
create & edit 3D:

_ExtrudeCrv

Create and Edit 3D Geometry

EXERCISE03

Modelling with Gumball



Gumball Commands:

move: arrow handles

scale: square handles

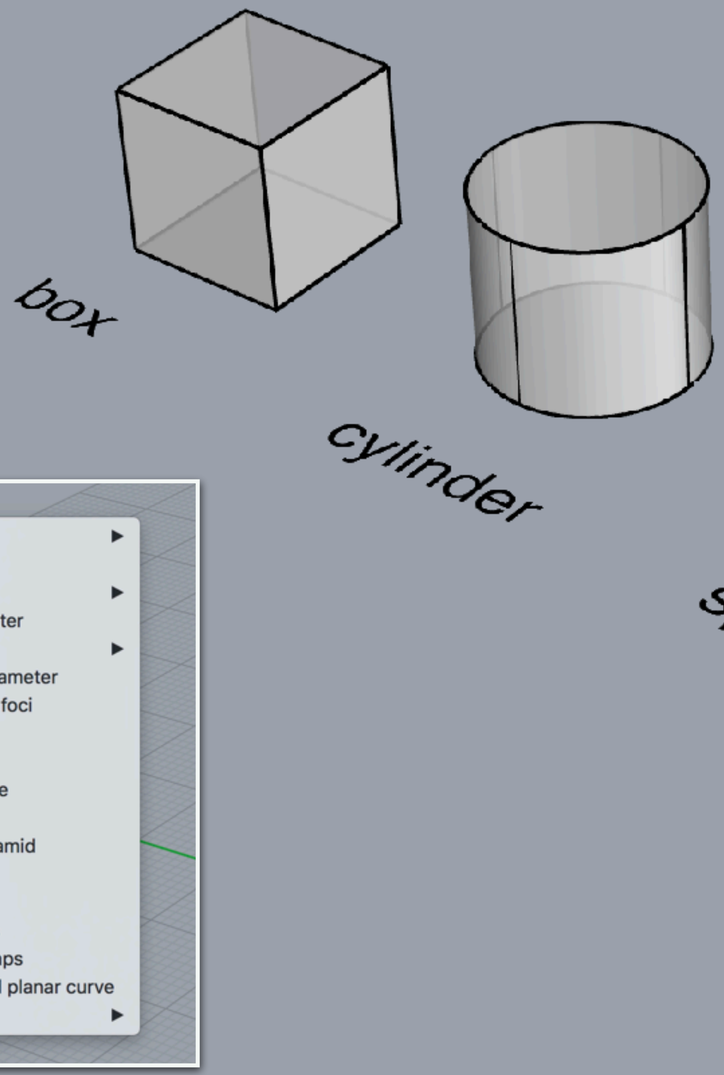
rotate: circle segment handles

extrude: click arrow handle > drag > press command&shift

select subsurface: command&shift + selecting surface or edge

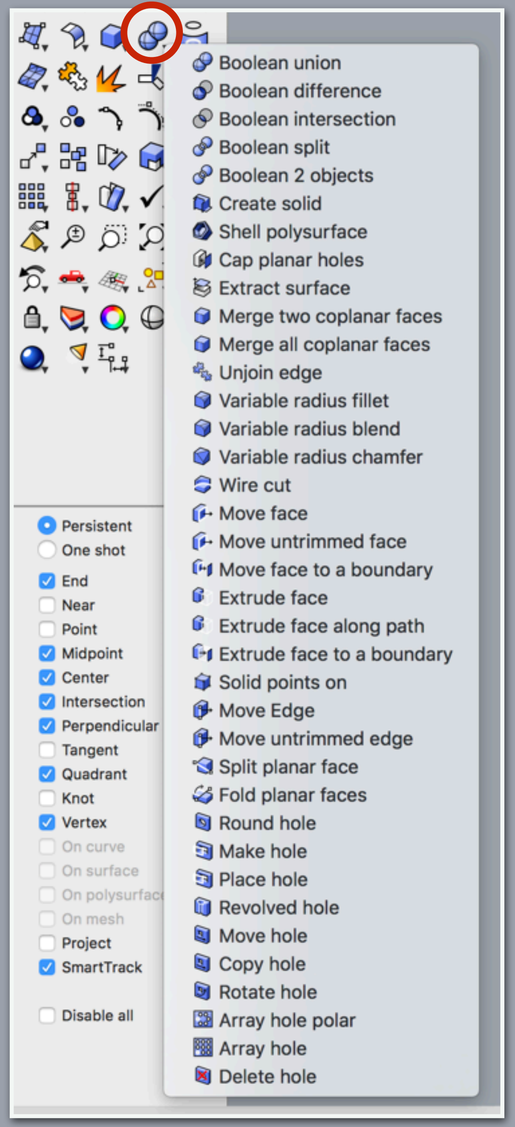
EXERCISE03

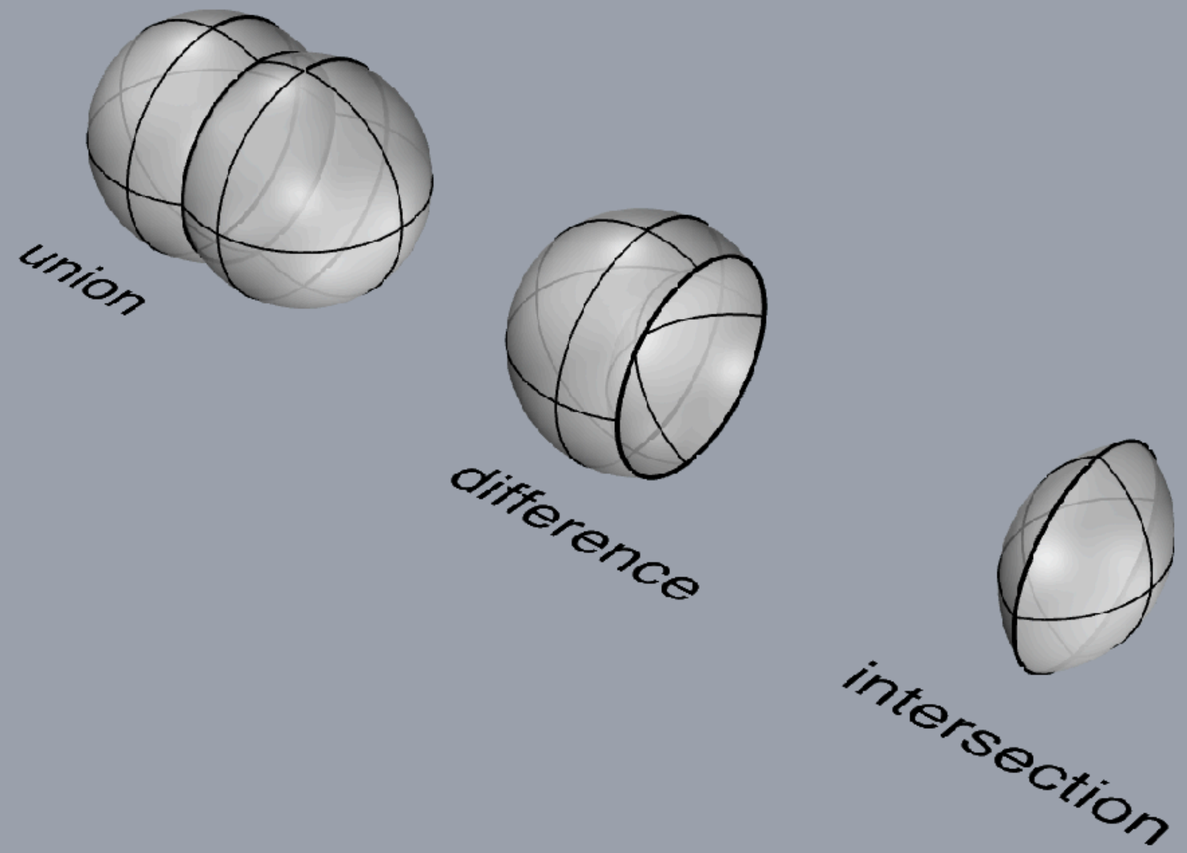
Create Solids



EXERCISE03.1

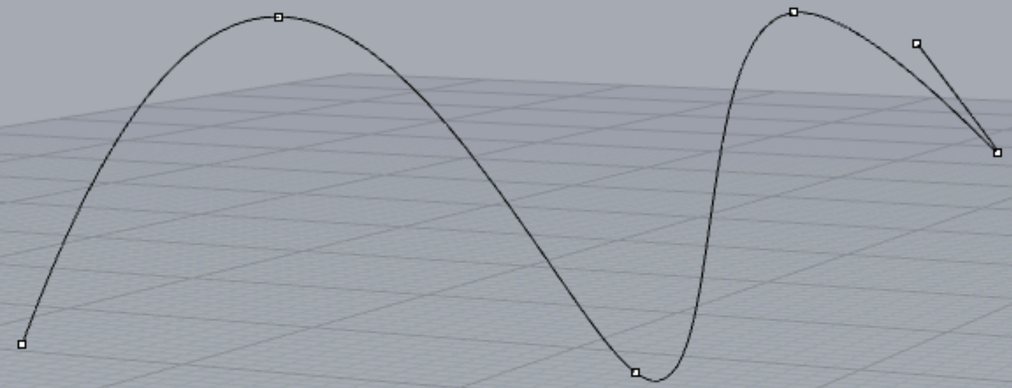
Edit Solids





EXERCISE04

CV Curve Creation and Editing



_curve

_move / _mirror / _trim / _extend

_EditPtOn

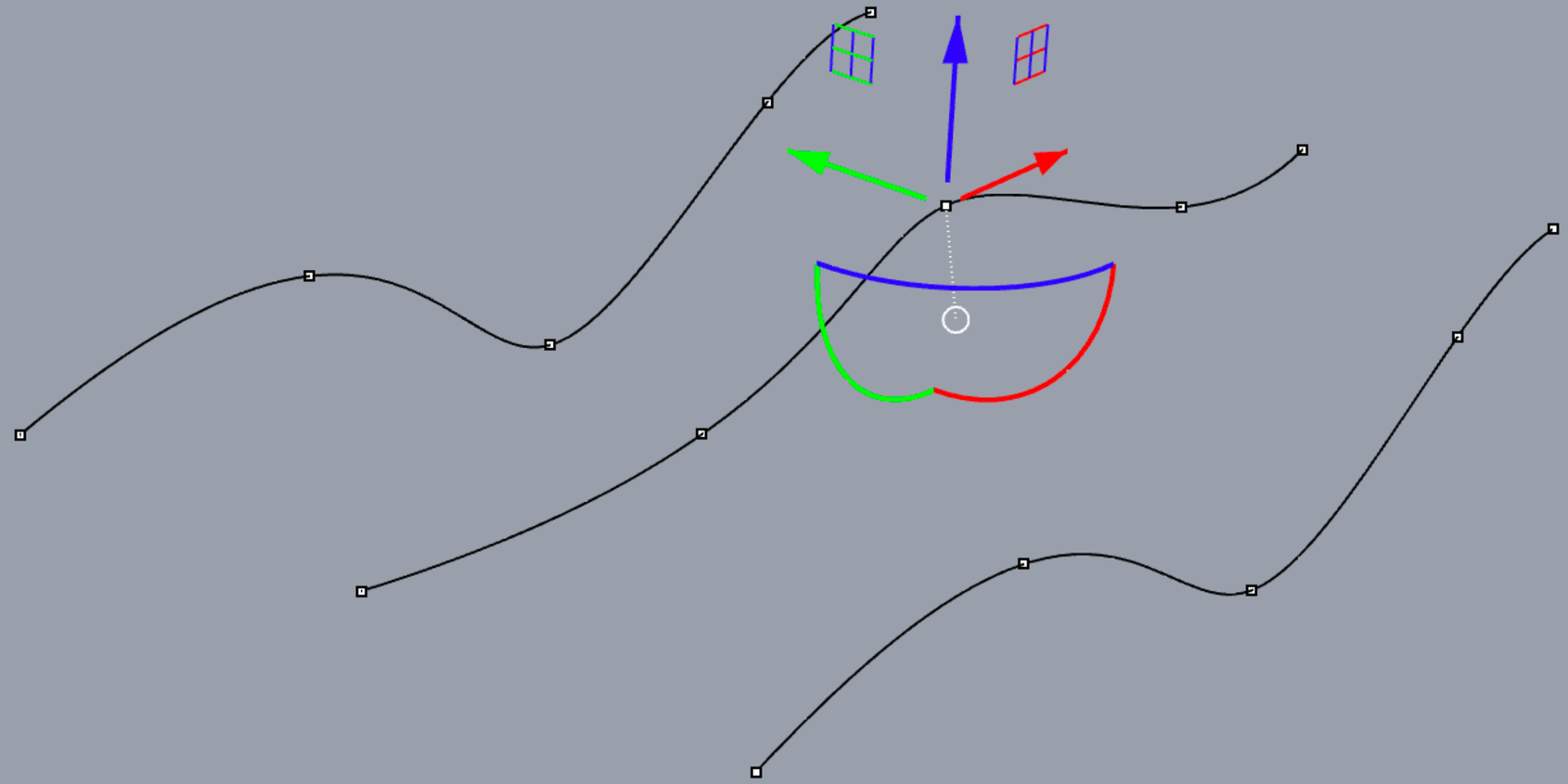
Use Osnap / Objektfang

Use Gridsnap / Rasterfang

Use gumball to to edit and move points

EXERCISE04

CV Curve Creation and Editing



_curve

_move / _mirror / _trim / _extend

_EditPtOn

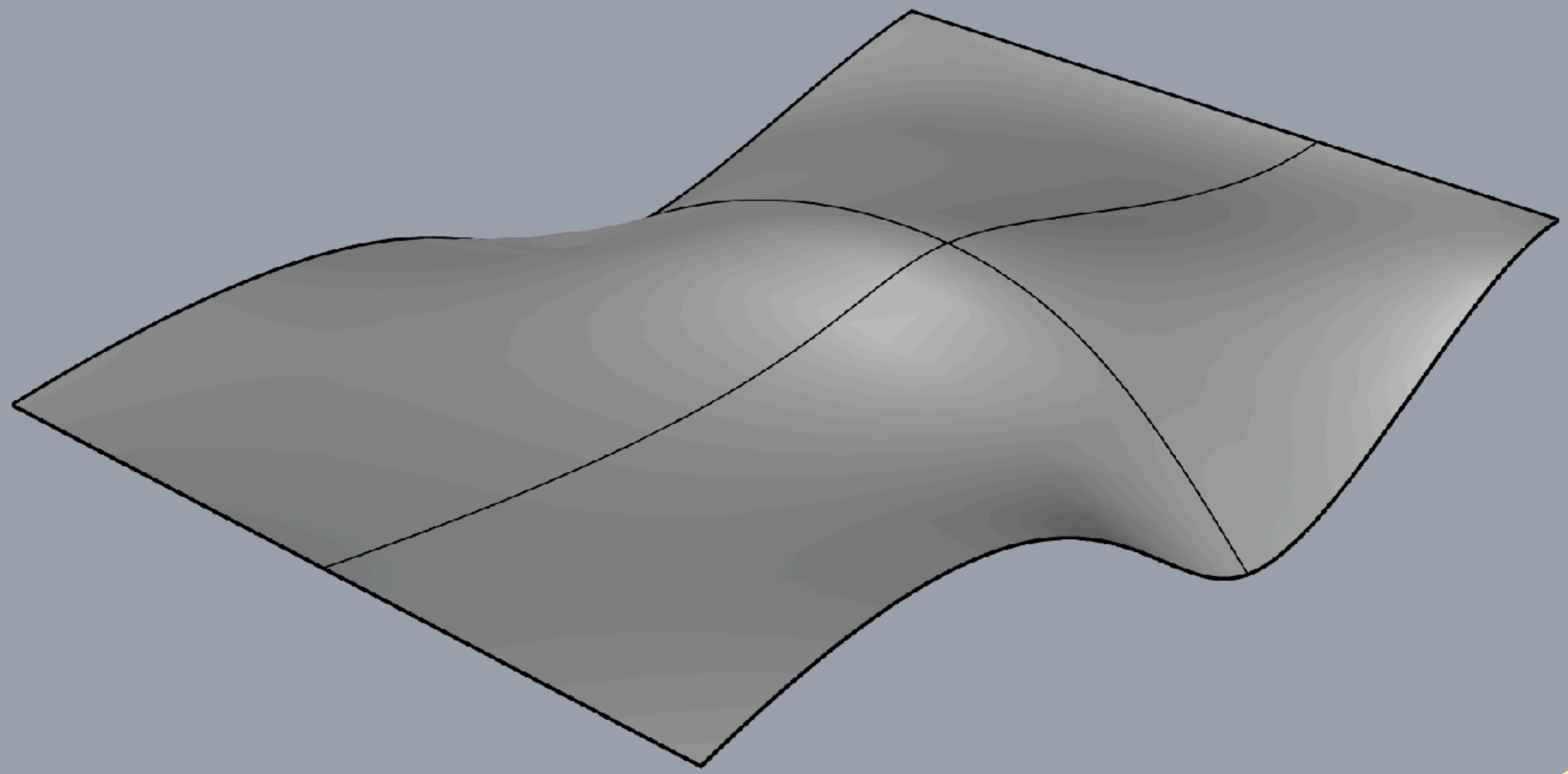
Use Osnap / Objektfang

Use Gridsnap / Rasterfang

Use gumball to to edit and move points

EXERCISE05

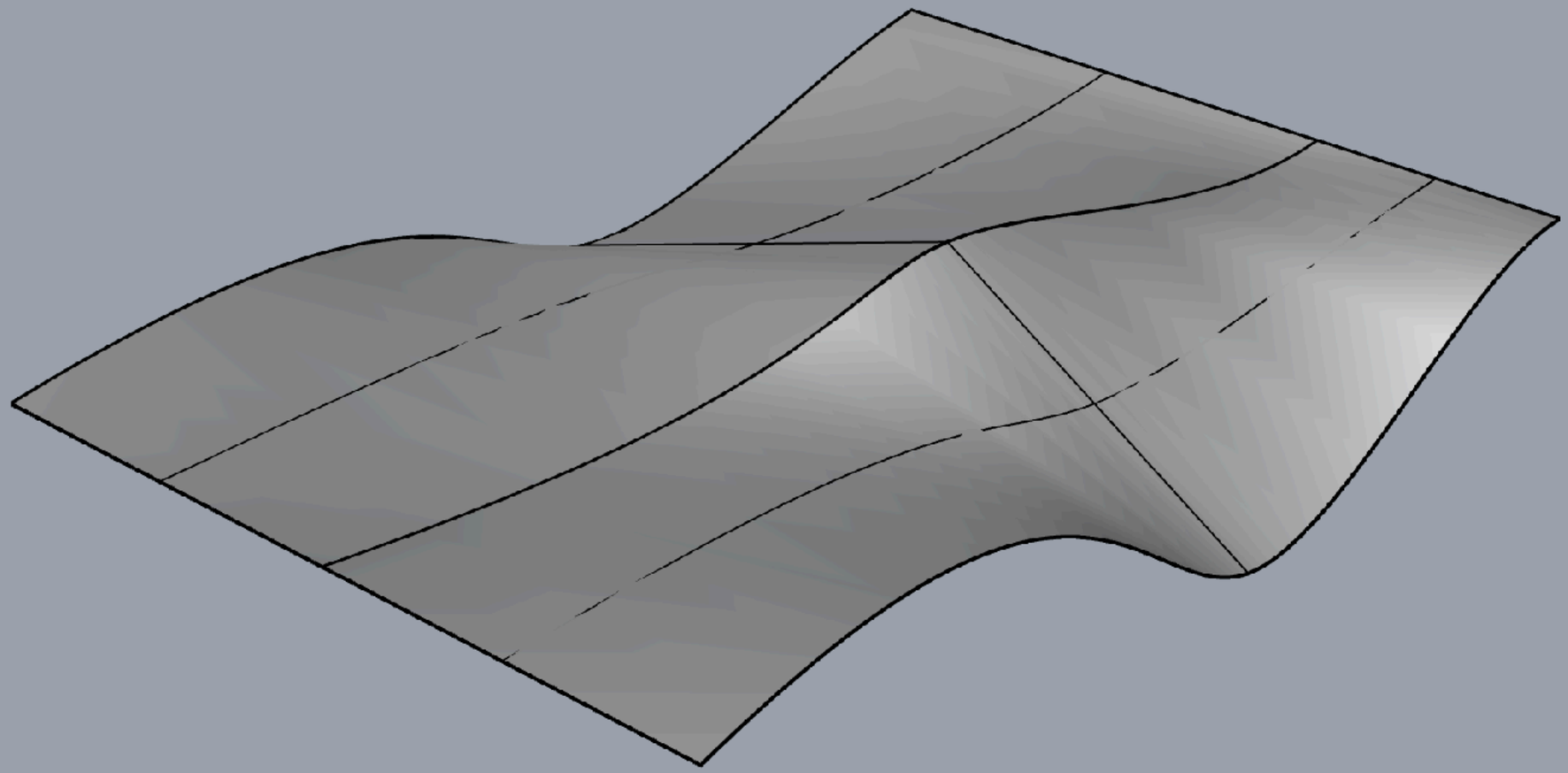
Create Surfaces - Loft



Loft > normal

EXERCISE05

Create Surfaces - Loft



Loft > straight sections

Overview

Bits & Atoms: Computer Aided Design

23.10.2017

Rhino Basics

30.10.2017

Rhino to Grasshopper

6.11.2017

Parametric Design with Grasshopper

Overview

Grasshopper

Grasshopper is a visual programming language and environment developed by David Rutten at Robert McNeel & Associates, that runs within the Rhinoceros 3D computer-aided design (CAD) application. The first version of Grasshopper was released in September 2007, and titled Explicit History. Grasshopper has become part of the standard Rhino toolset in Rhino 6.0 and later.

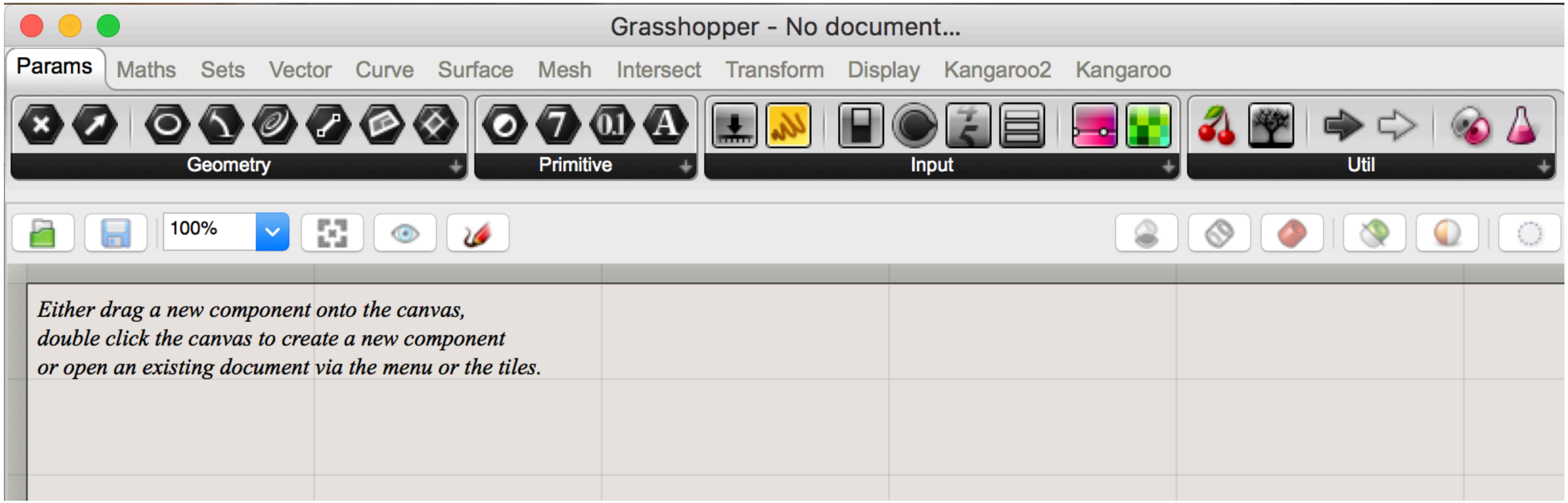
- Grasshopper is primarily used to build generative algorithms, such as for generative art. Many of Grasshopper's components create 3D geometry.

- Advanced uses of Grasshopper include parametric modelling for structural engineering, parametric modelling for architecture and fabrication, lighting performance analysis for eco-friendly architecture and building energy consumption.

Grasshopper is "The new Grasshopper environment provides an intuitive way to explore designs without having to learn to script.” (AEC Magazine)

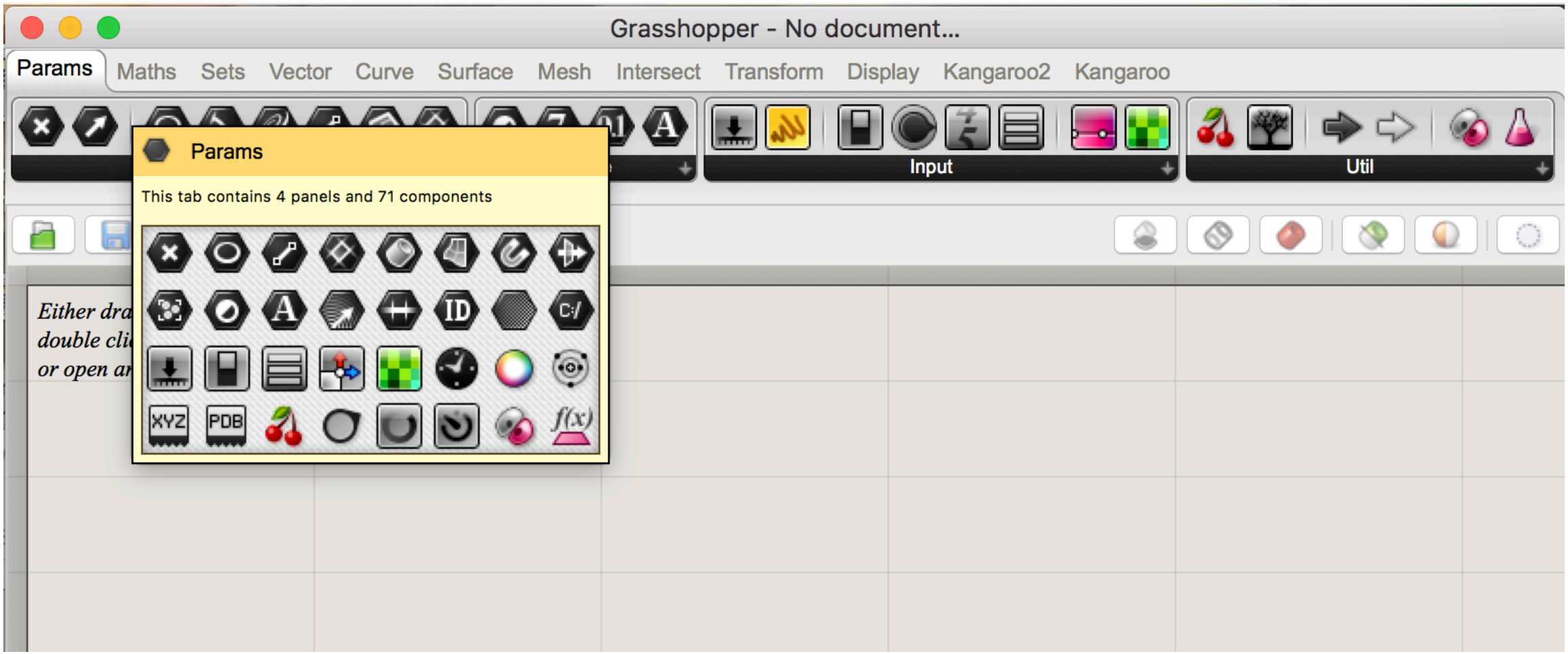
Overview

Component Panels (Container System)



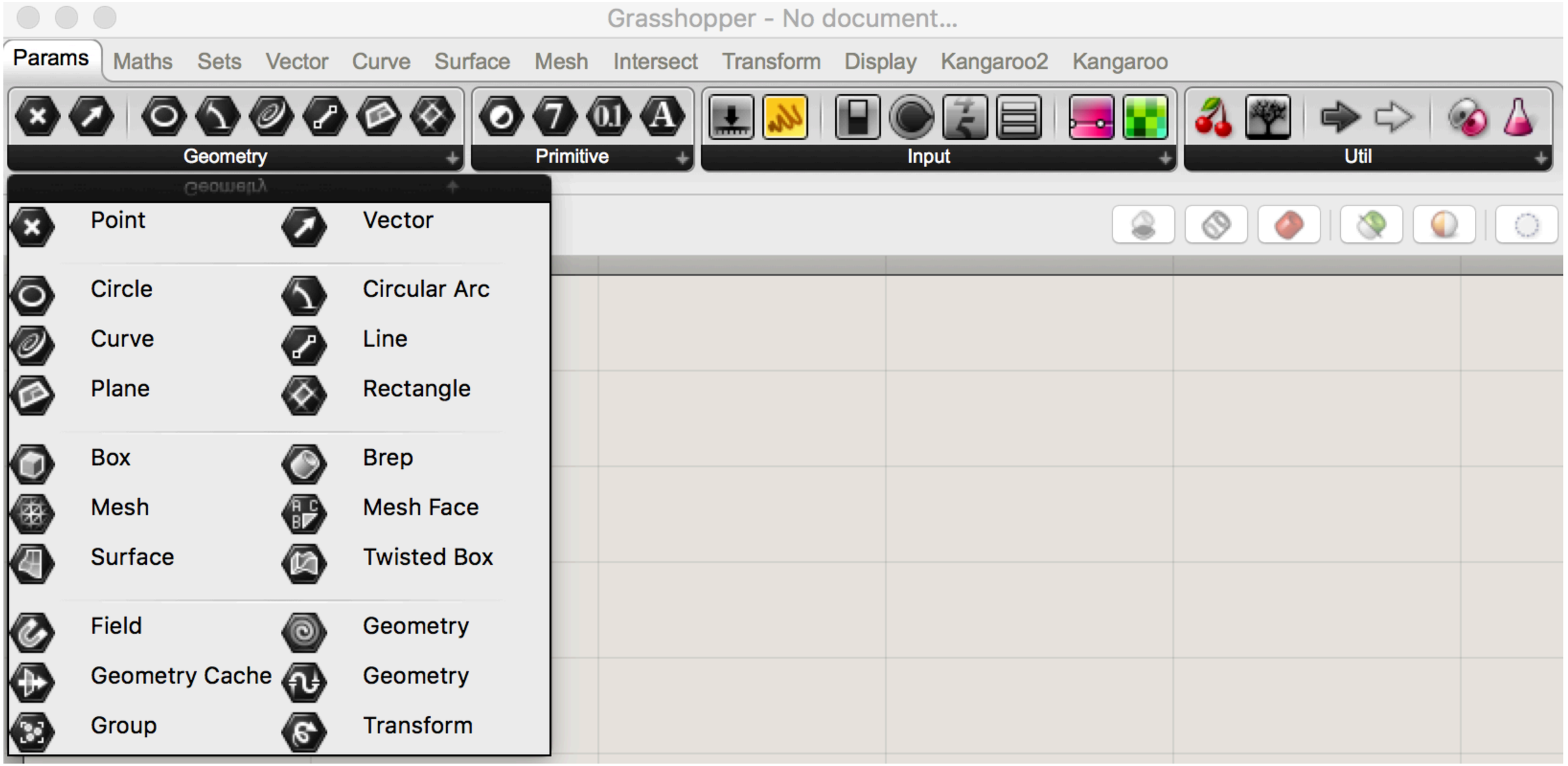
Overview

Component Panels (Container System)



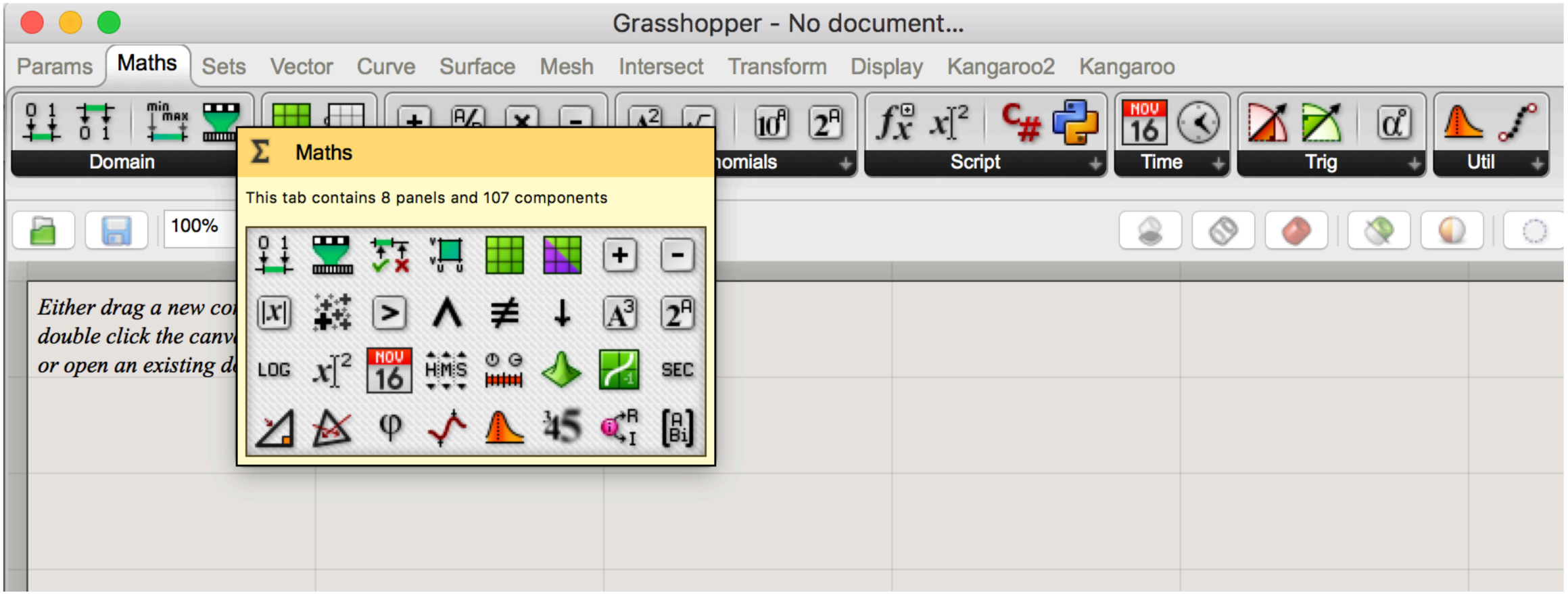
Overview

Component Panels (Container System)



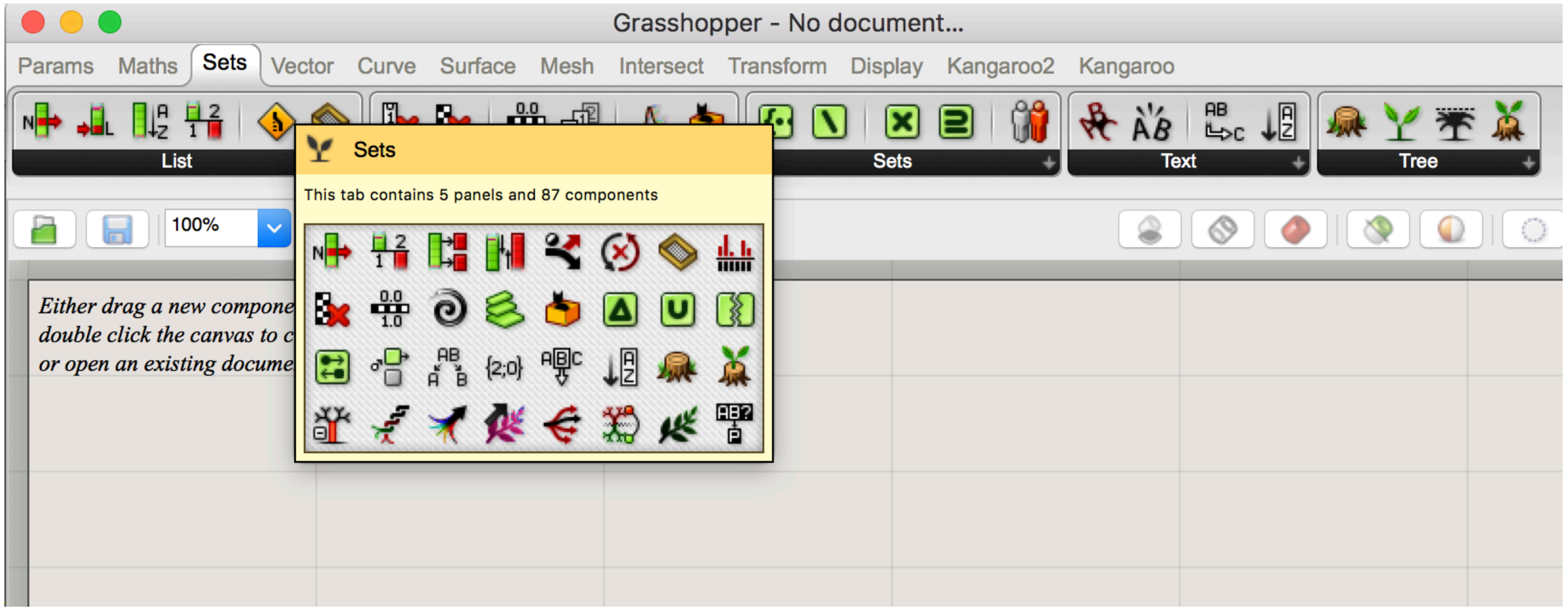
Overview

Component Panels (Container System)



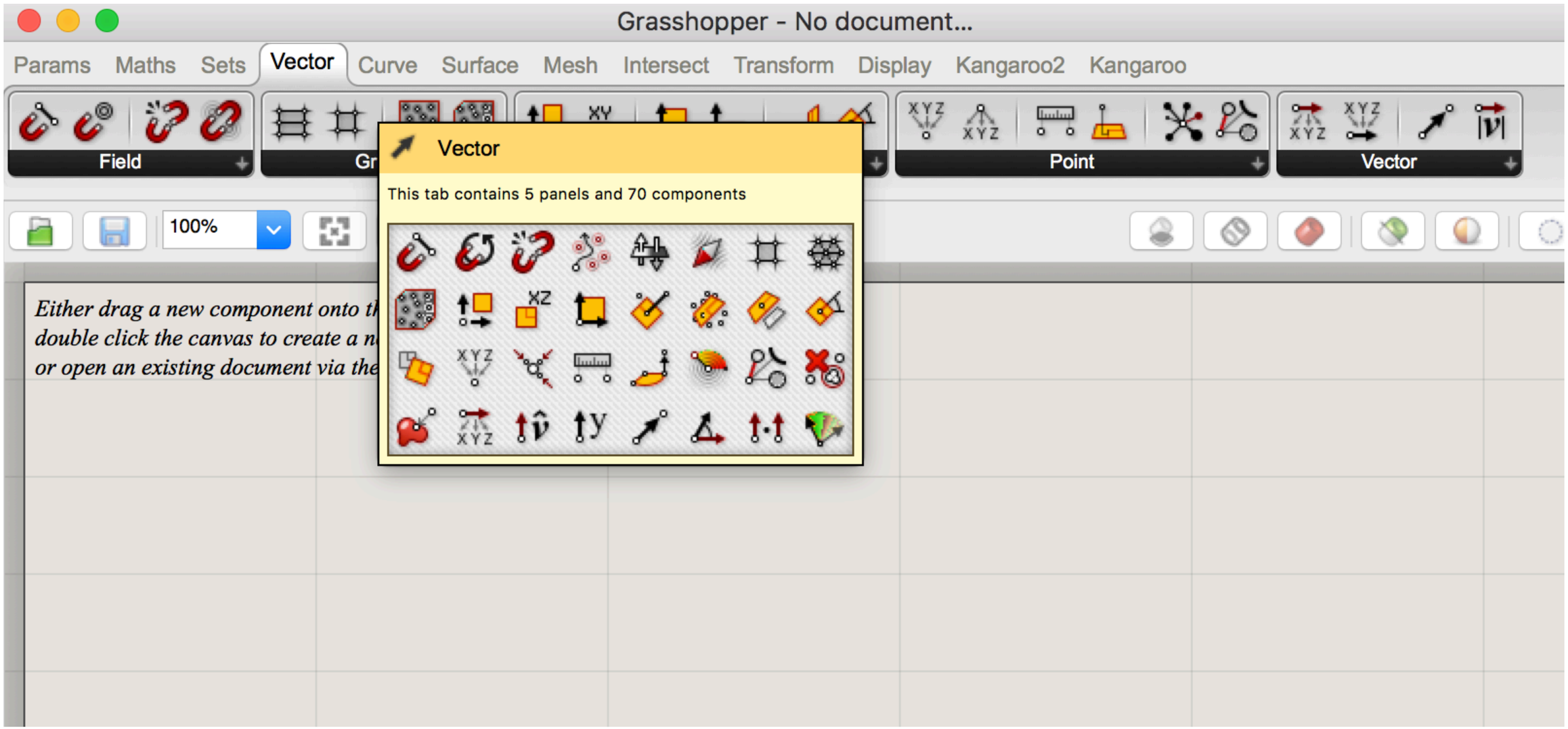
Overview

Component Panels (Container System)



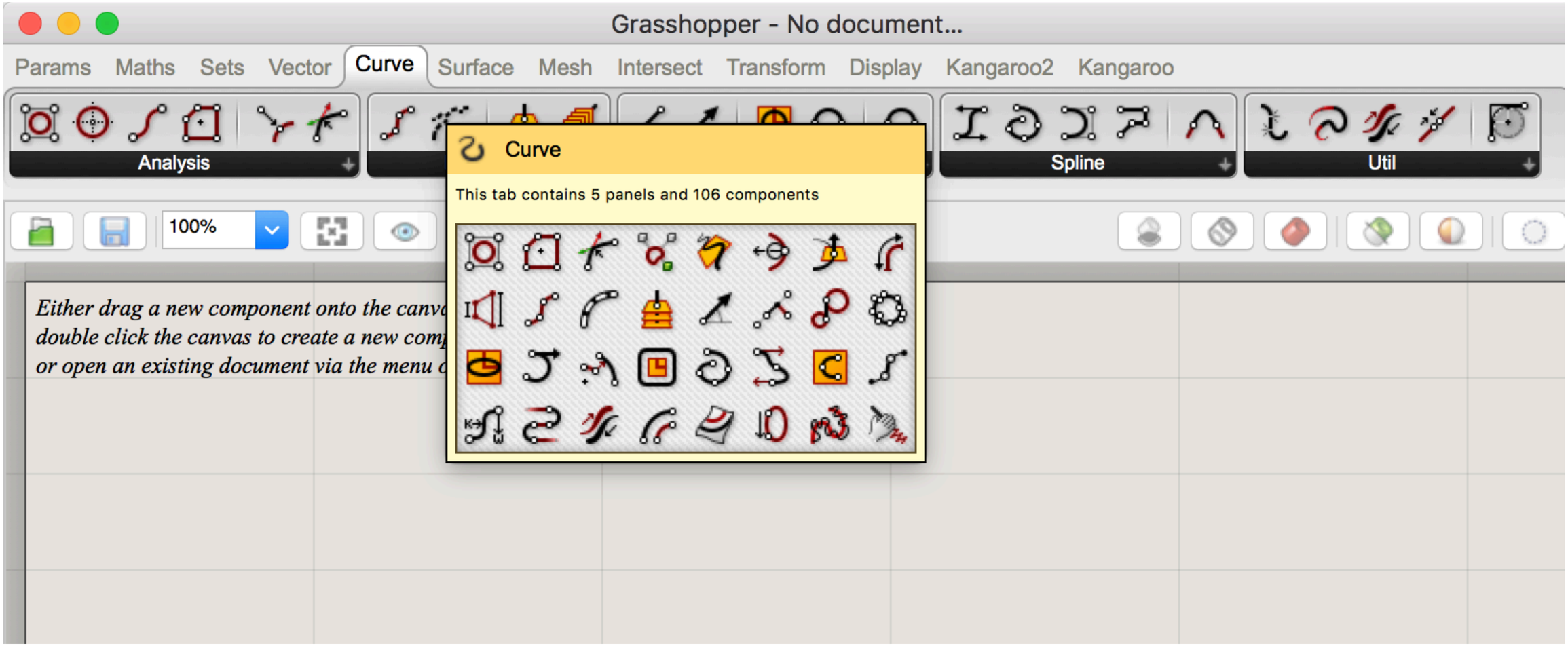
Overview

Component Panels (Container System)



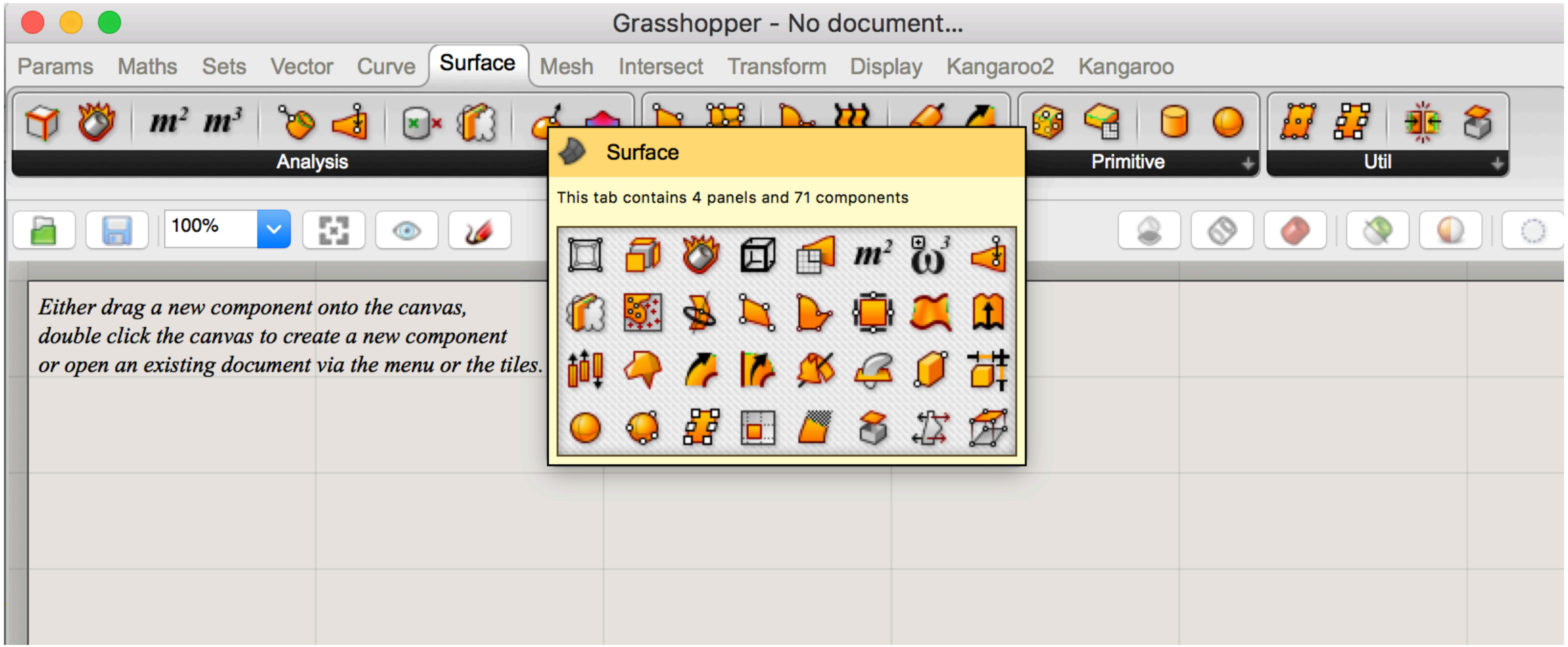
Overview

Component Panels (Container System)



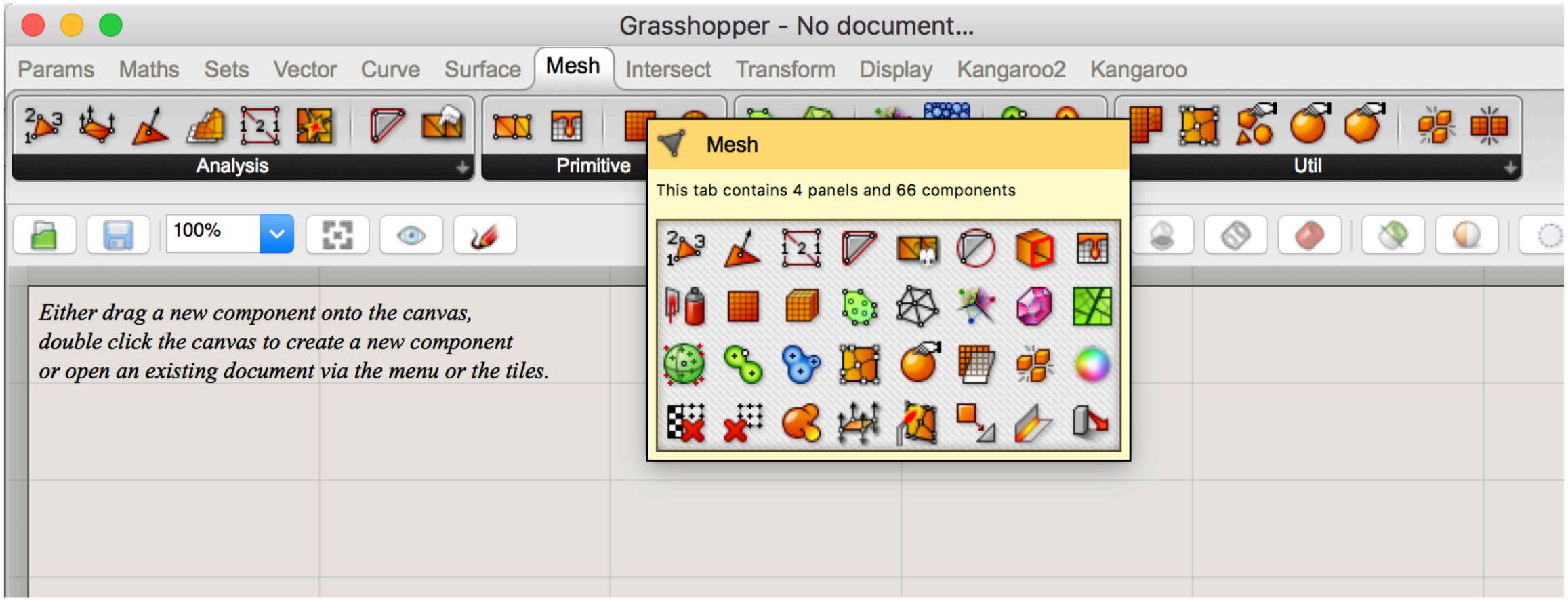
Overview

Component Panels (Container System)



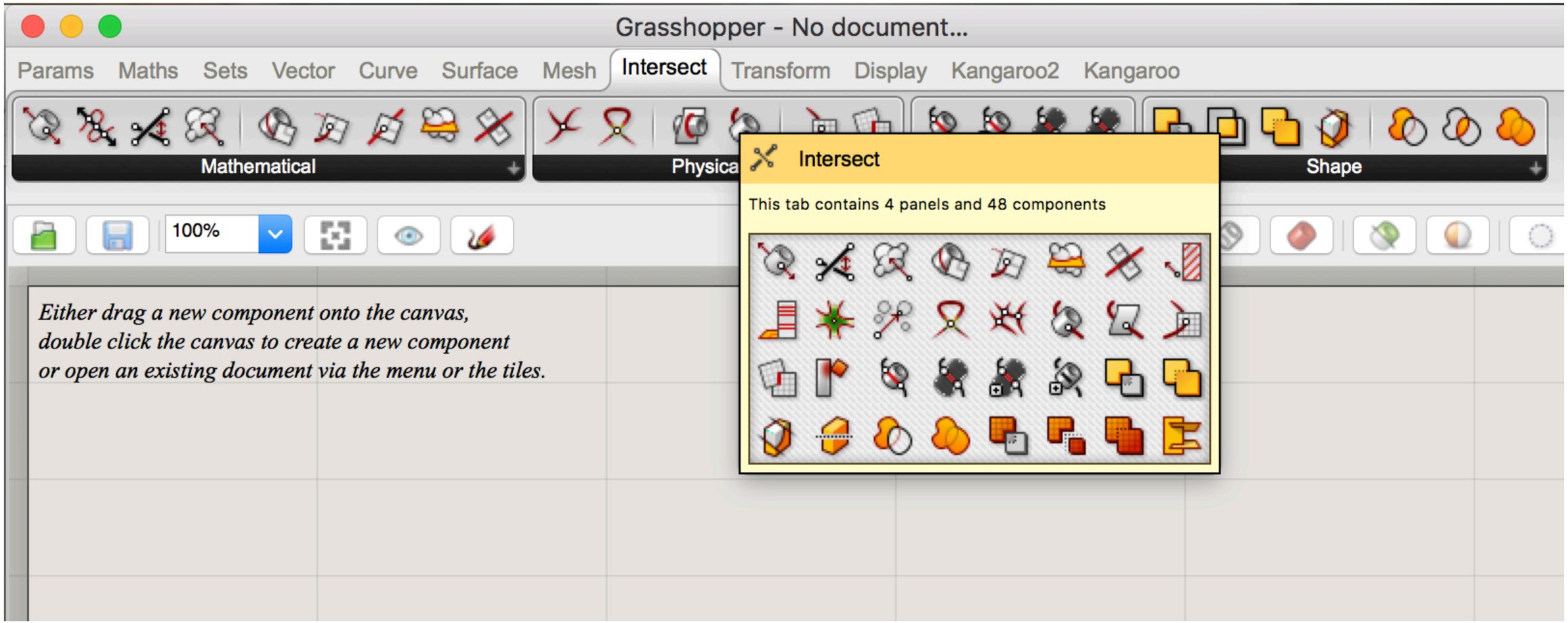
Overview

Component Panels (Container System)



Overview

Component Panels (Container System)

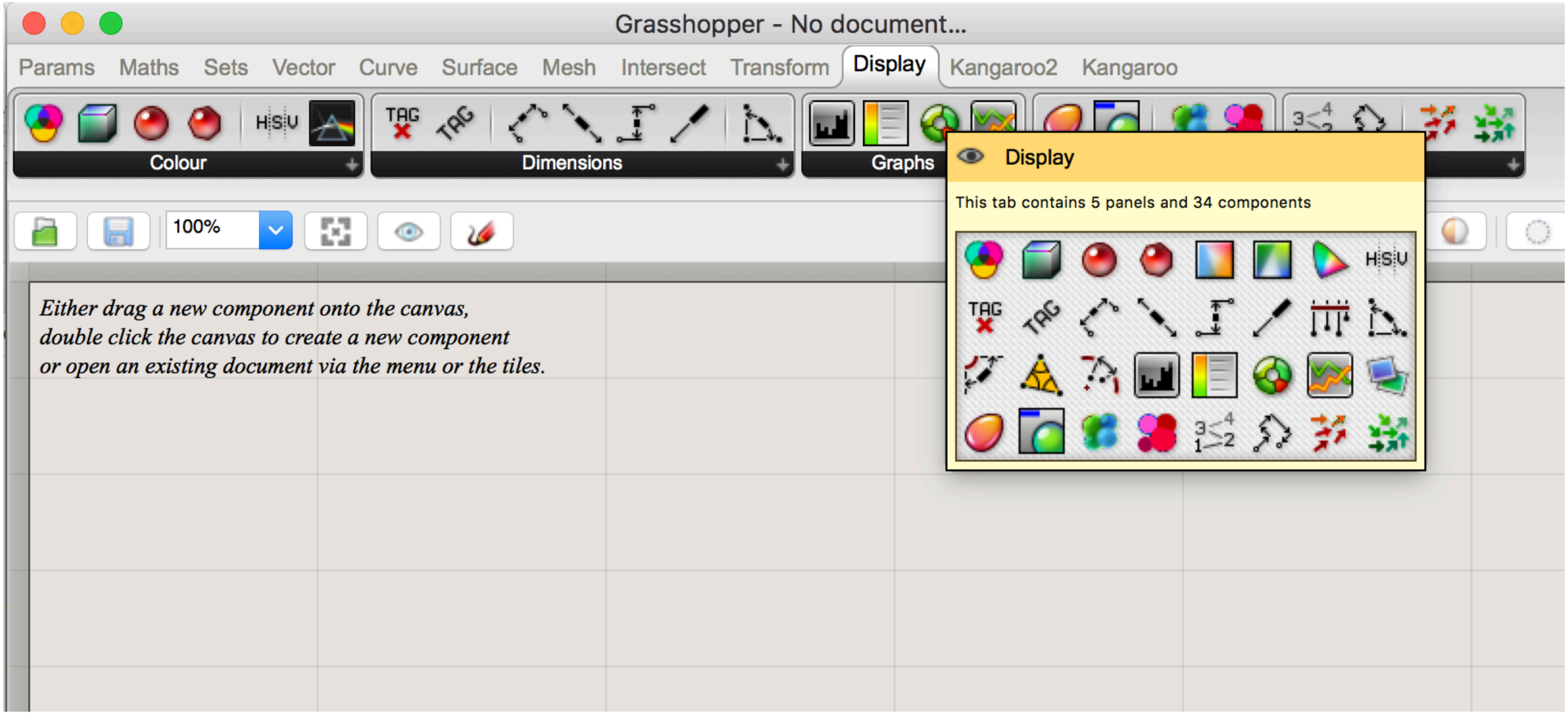


Overview

Component Panels (Container System)

Overview

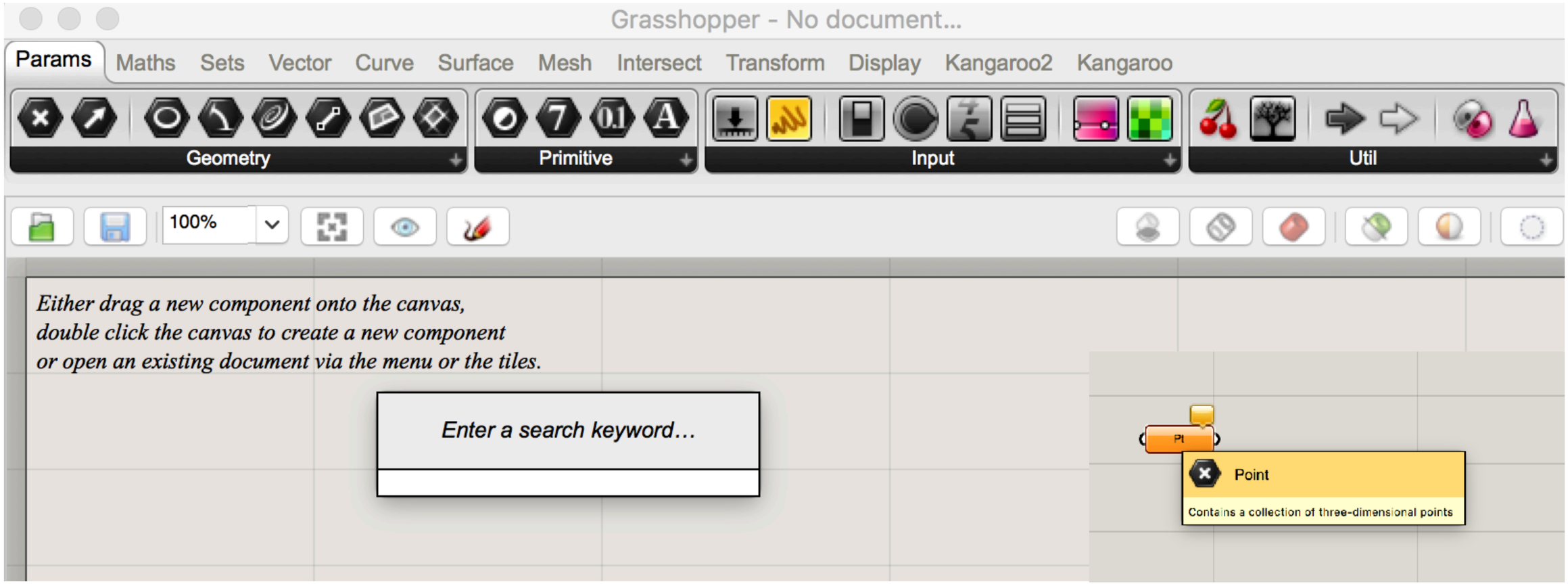
Component Panels (Container System)



Overview

Component Panels (Container System)

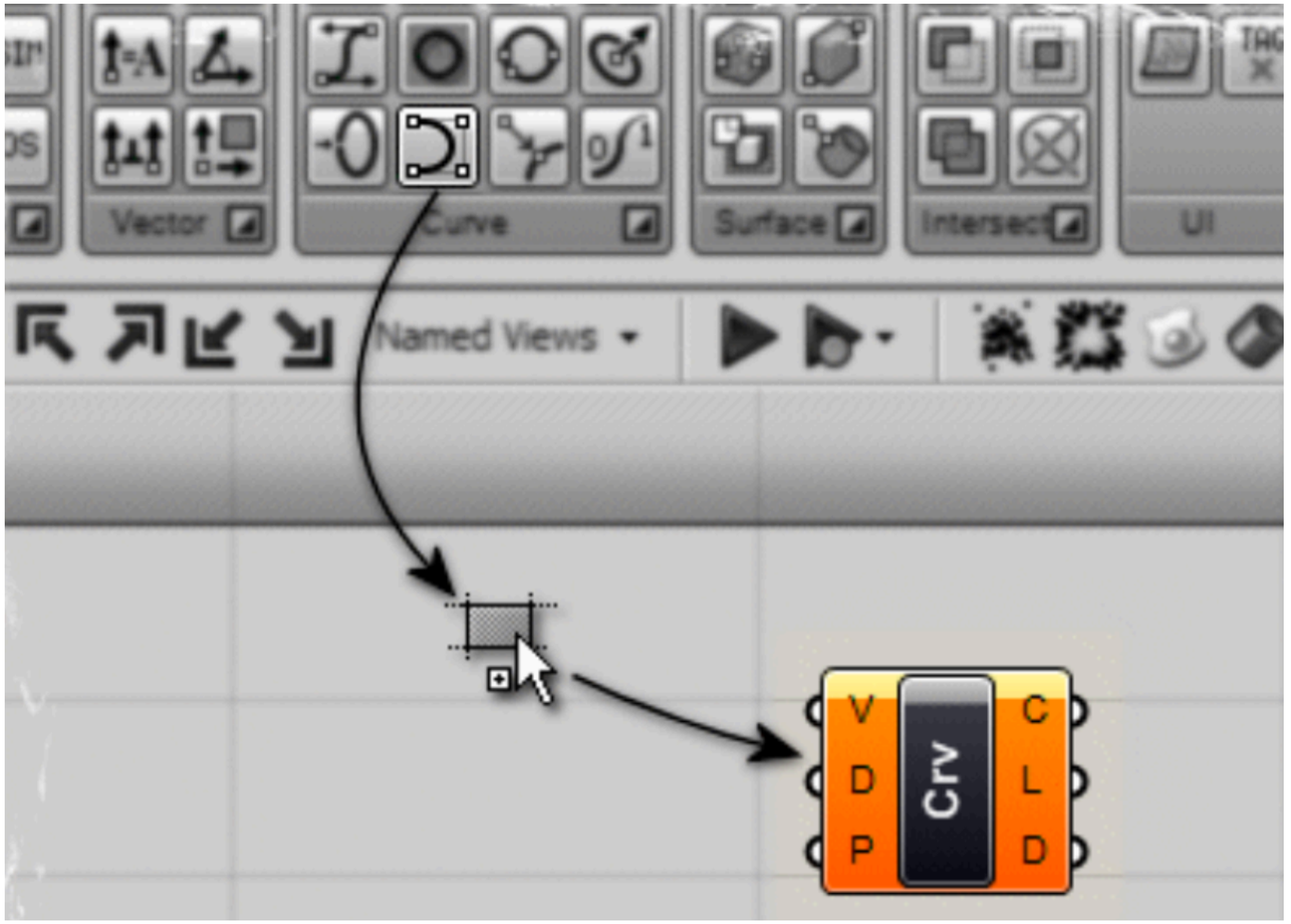
You can also find components by name, by double-clicking anywhere on the canvas; launching a pop-up search box. Type in the name of the component you are looking for and you will see a list of parameters or components that match your request.



Overview

Component Panels (Container System)

- The program gets created by dragging components onto a canvas. The outputs to these components are then connected to the inputs of subsequent components.



Overview

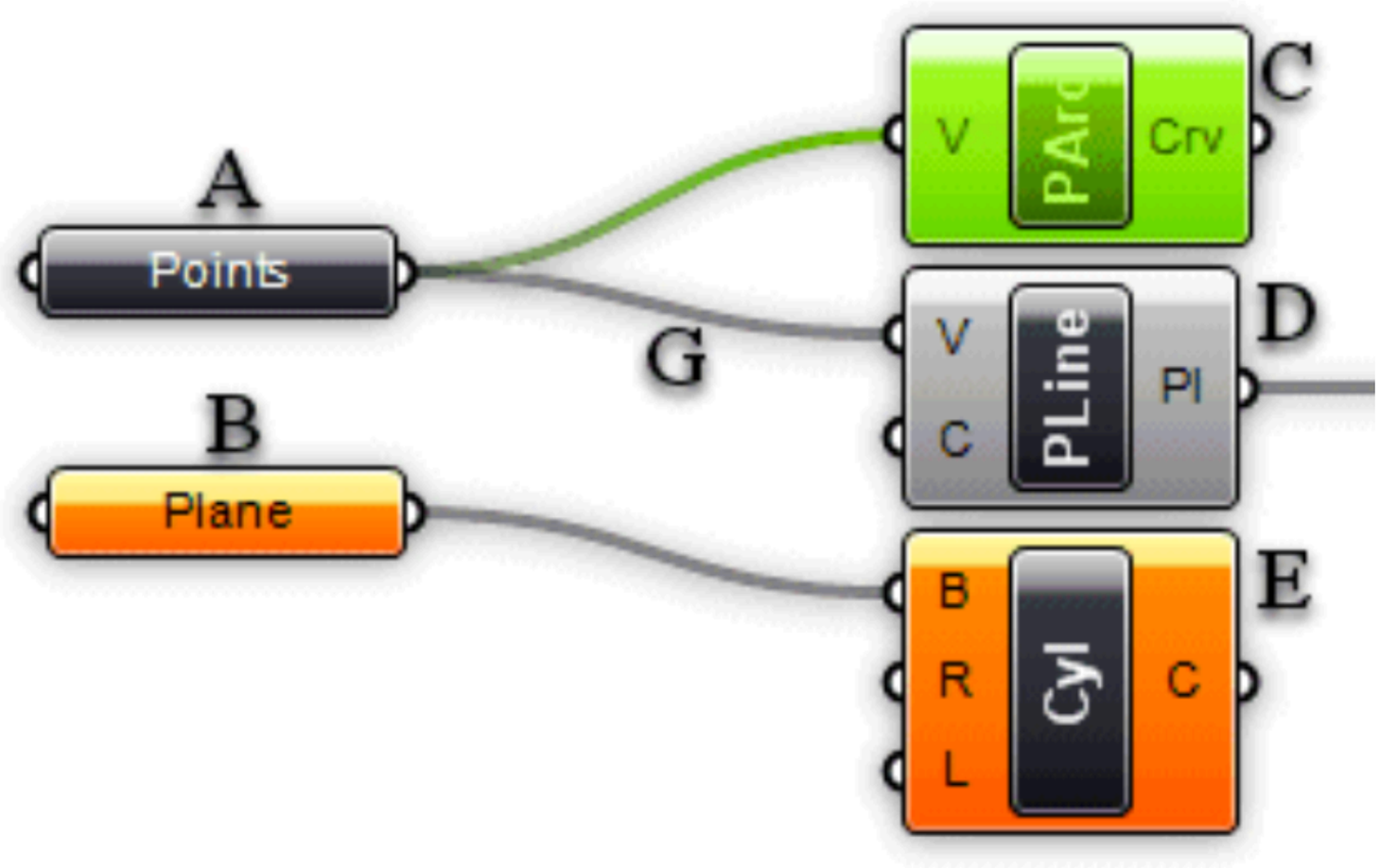
Component Panels (Container System)

Overview

Component Panels (Container System)

Parameters contain data, meaning that they store stuff.

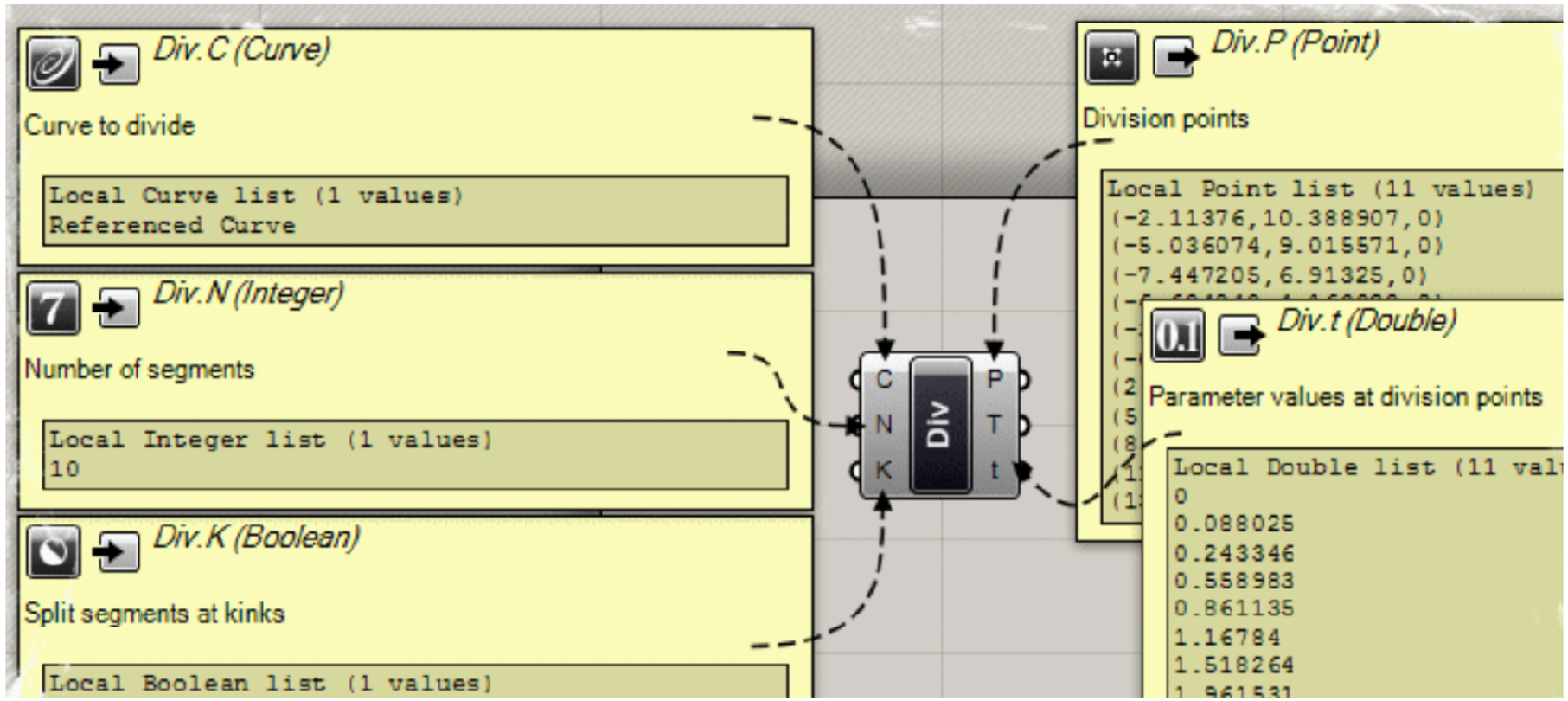
Components contain actions, meaning that they do stuff.



Overview

Component Panels (Container System)

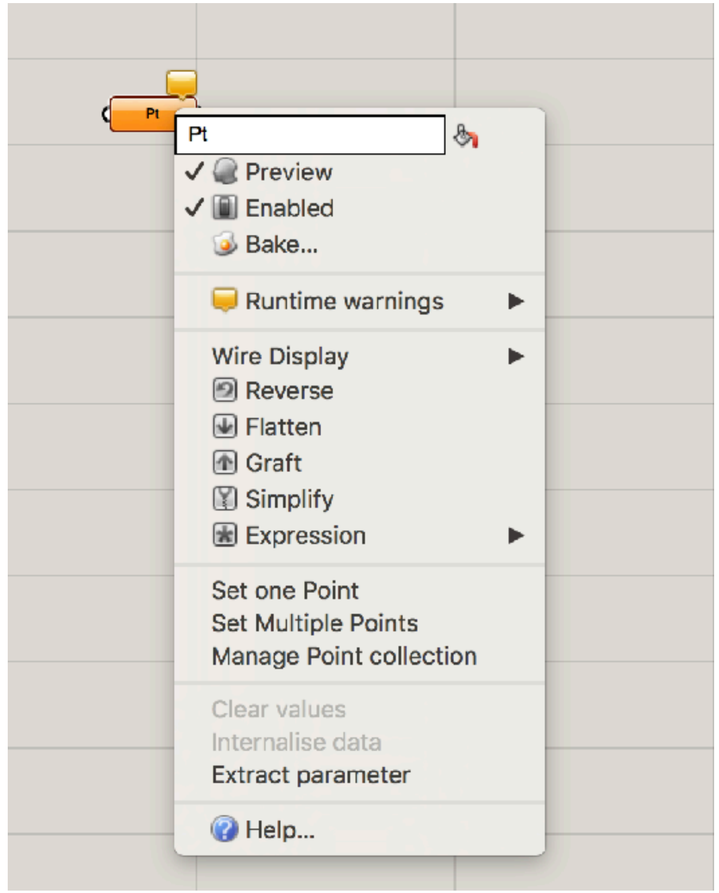
When you hover your mouse over the individual parts of a Component object, you'll see different tooltips that indicate the particular type of the (sub)object currently under the mouse. Tooltips are quite informative since they tell you both the type and the data of individual parameters:



Component Panels (Container System)

All objects on the Canvas have their own context menus that expose most of the features for that particular component.

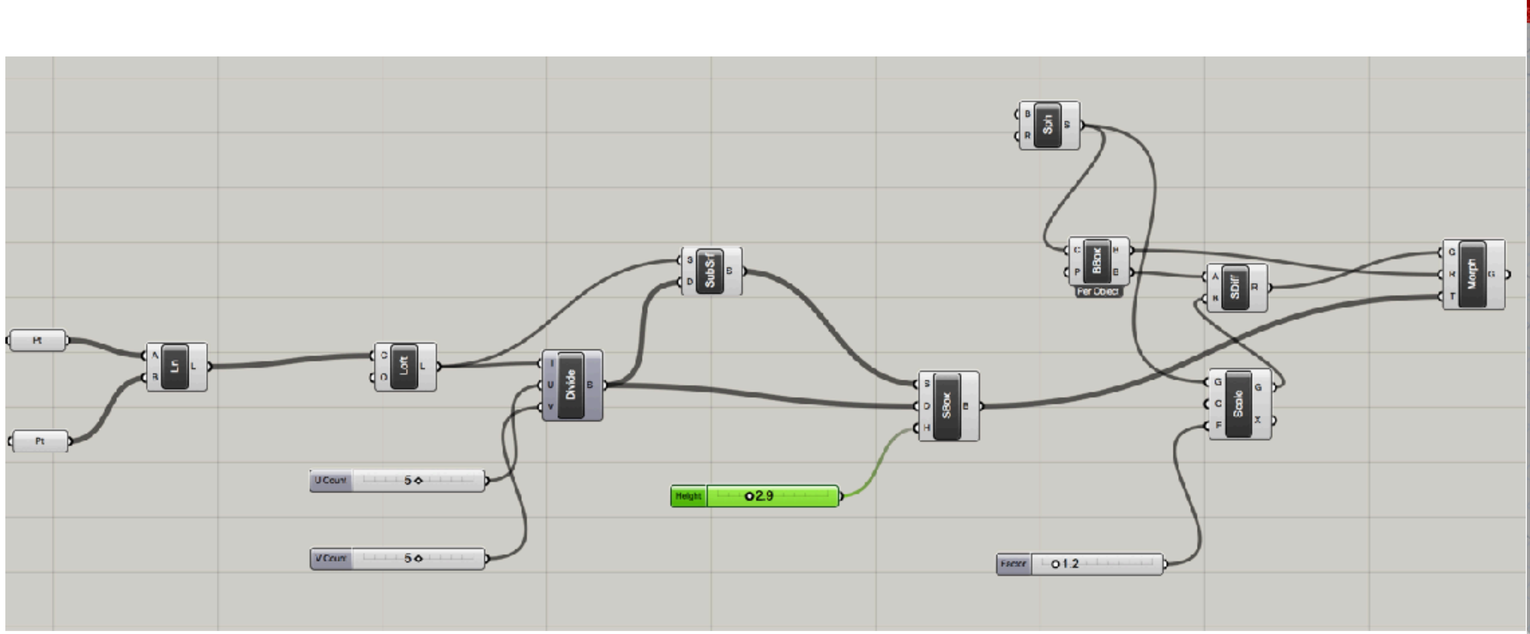
Right click on the Parameter or Component indicates those features:

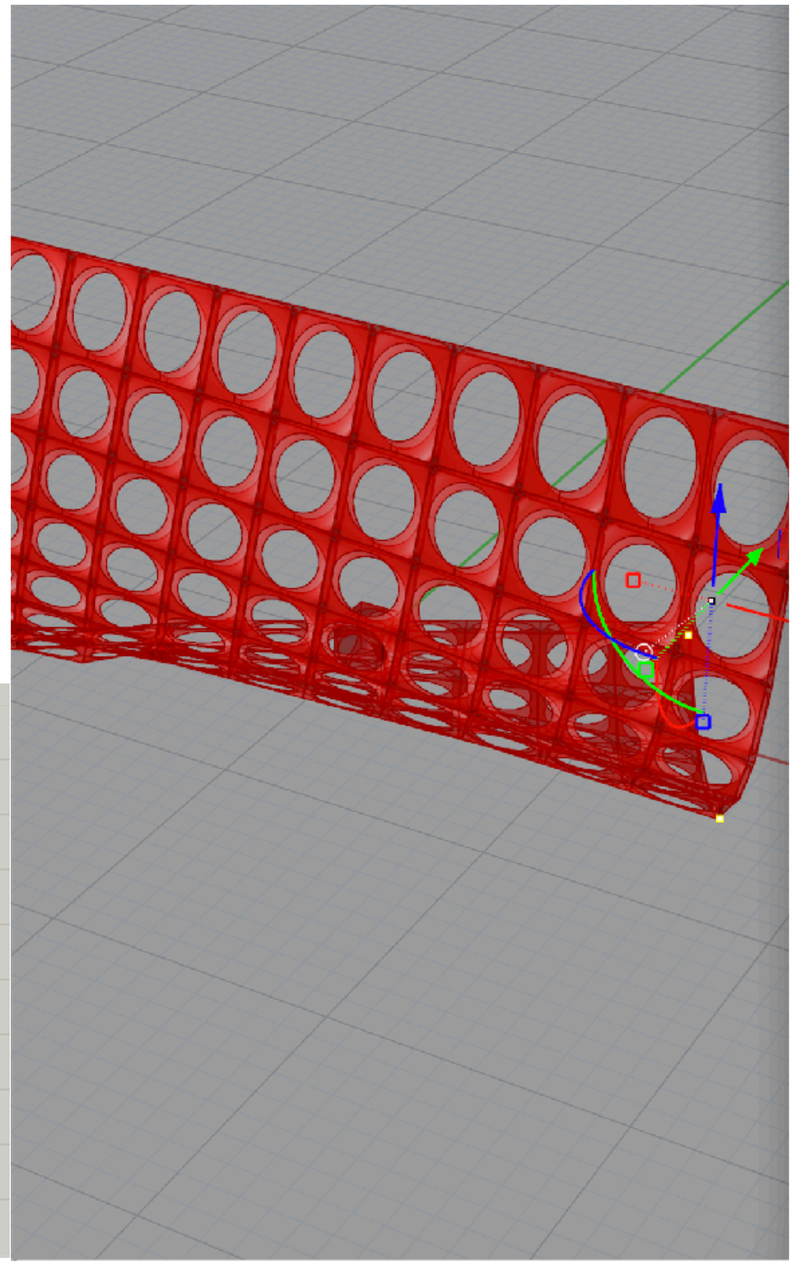


Or Blank space button shows the same features in symbols:

Exercise I

BoxMorph





Exercise I

BoxMorph

1. Parameter Points: Parameter wants to store stuff - still empty

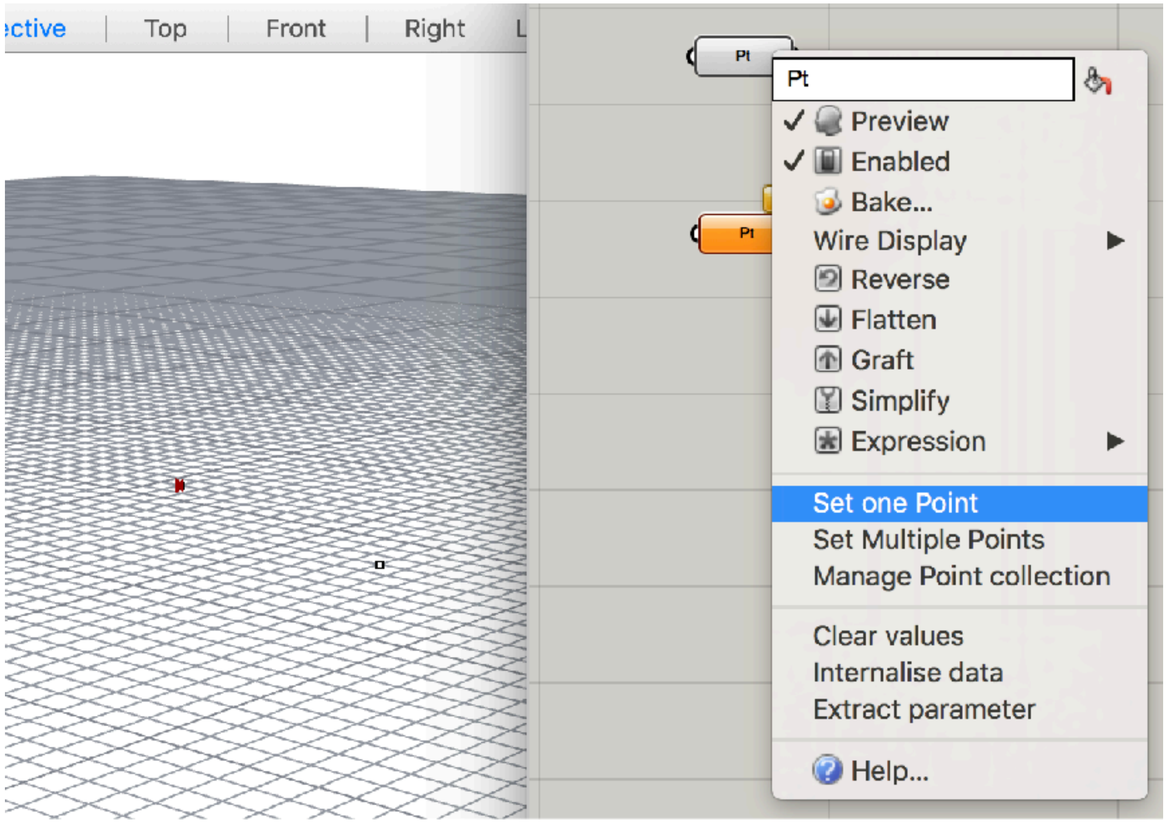
2. Connect Rhino and Grasshopper points:

In Grasshopper:

Right click, set one point

In Rhino:

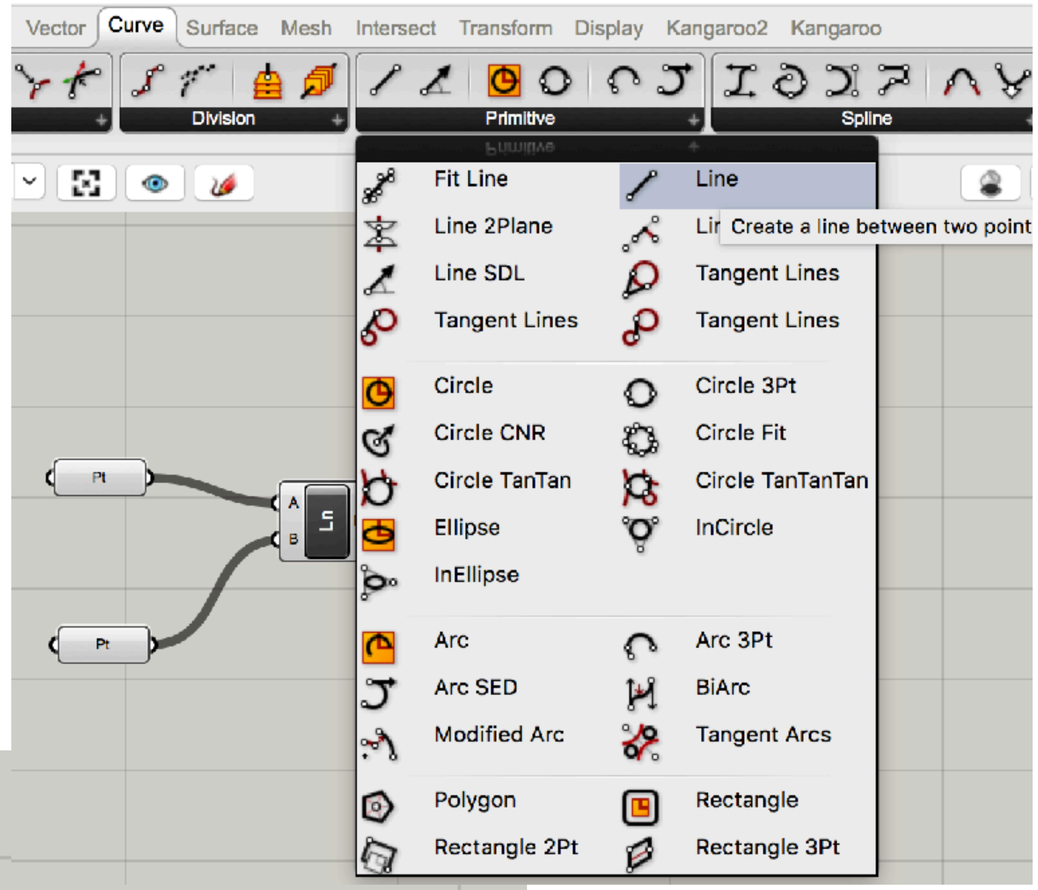
Click on point

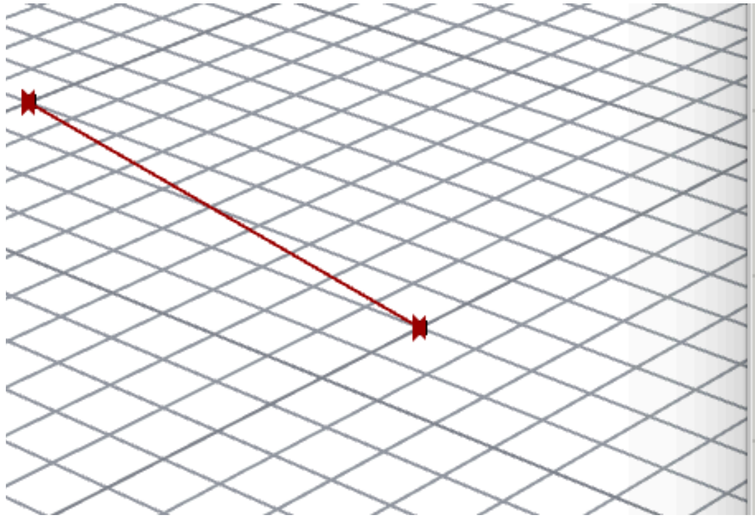


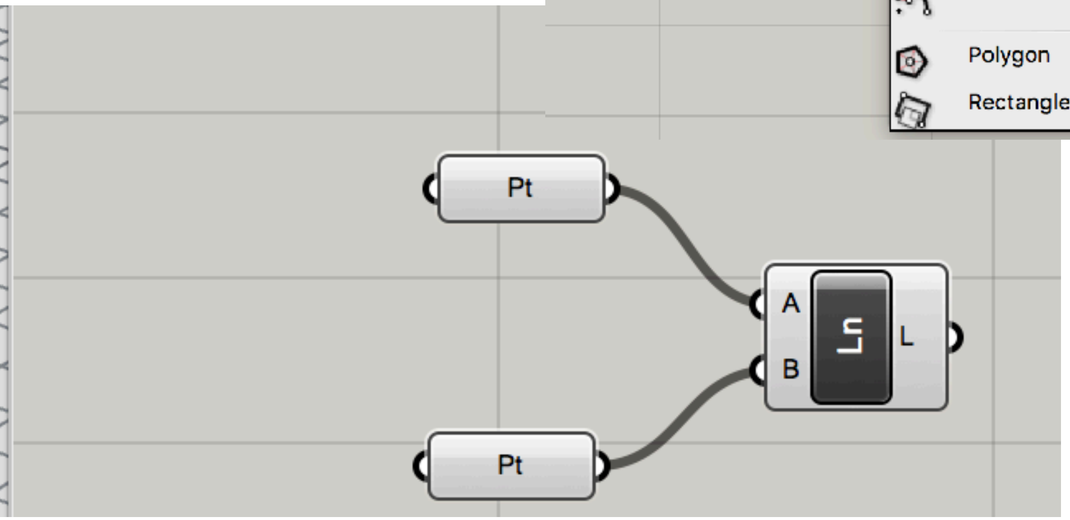
Exercise I

BoxMorph

Primitive: Line

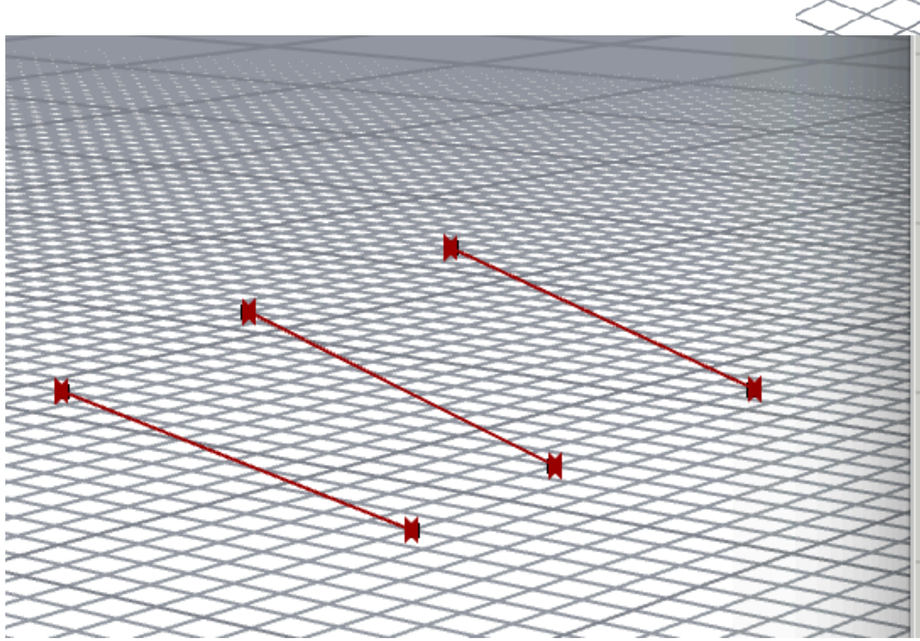


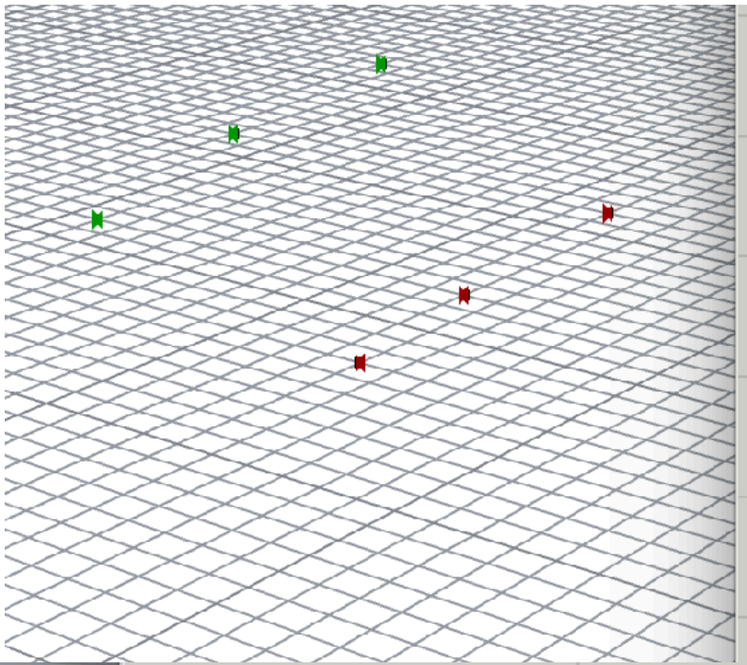


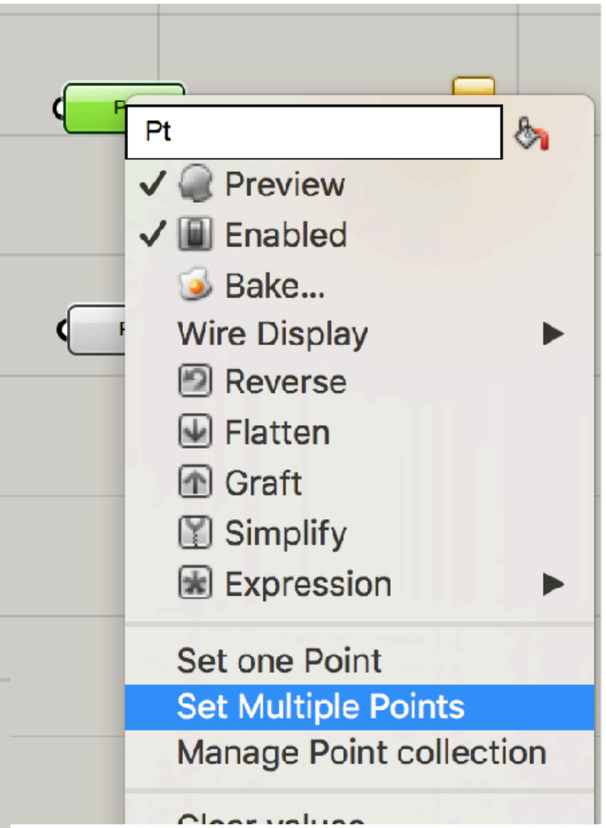


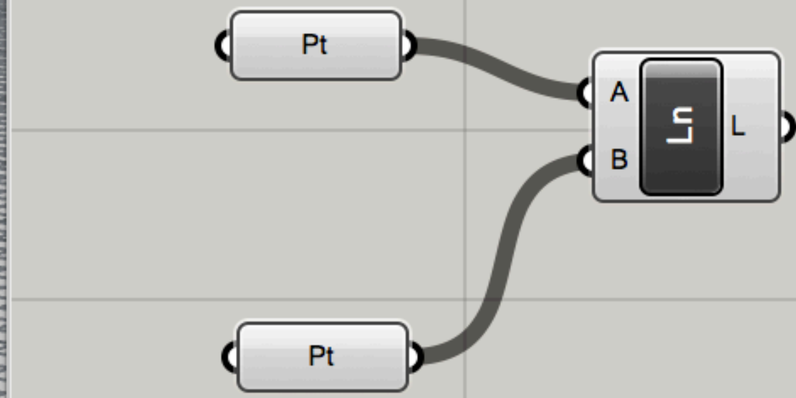
Exercise I

BoxMorph





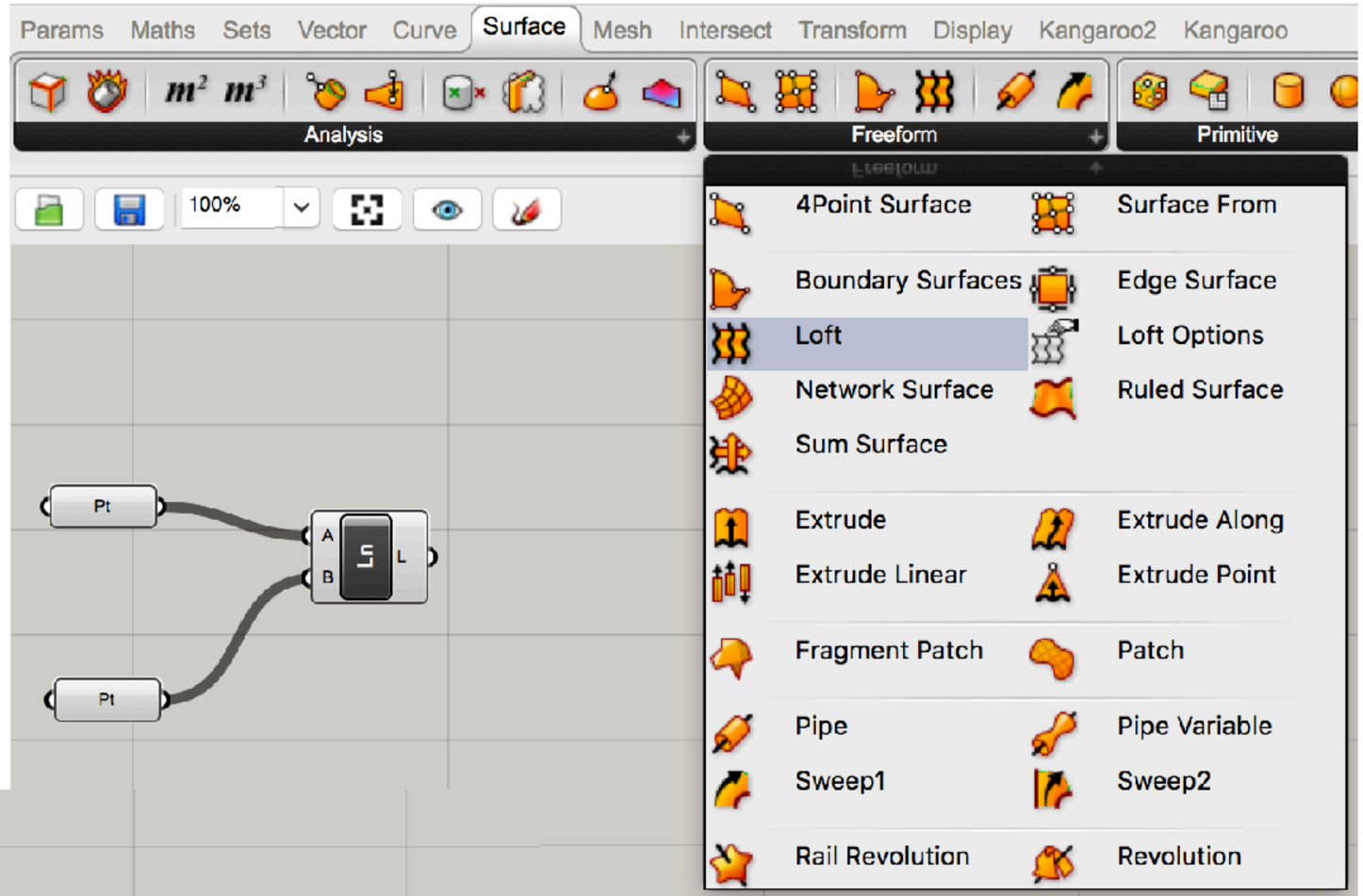


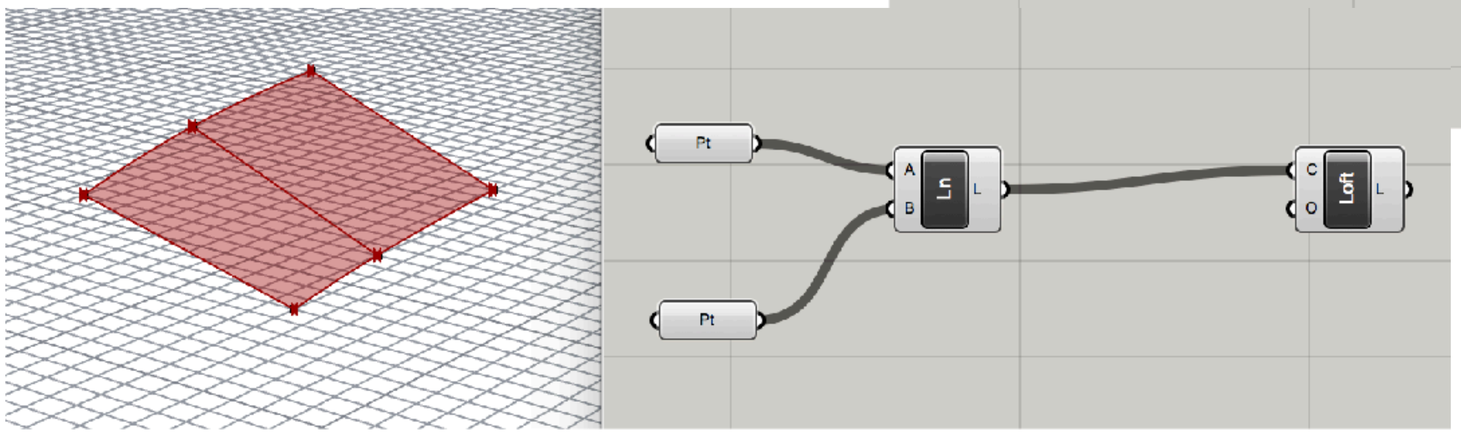


Exercise I

BoxMorph

Loft: To create a surface from joined lines

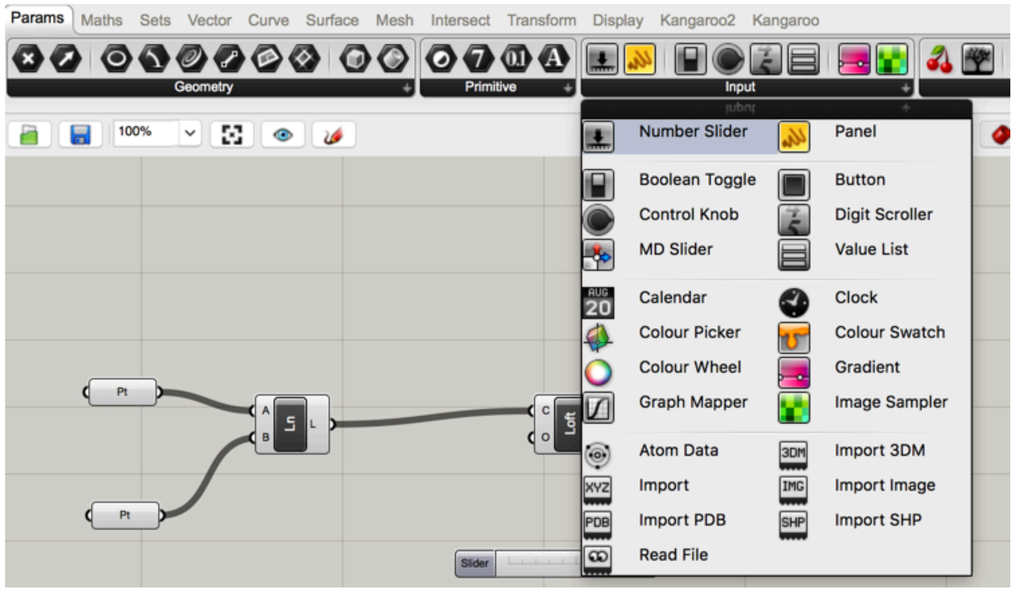




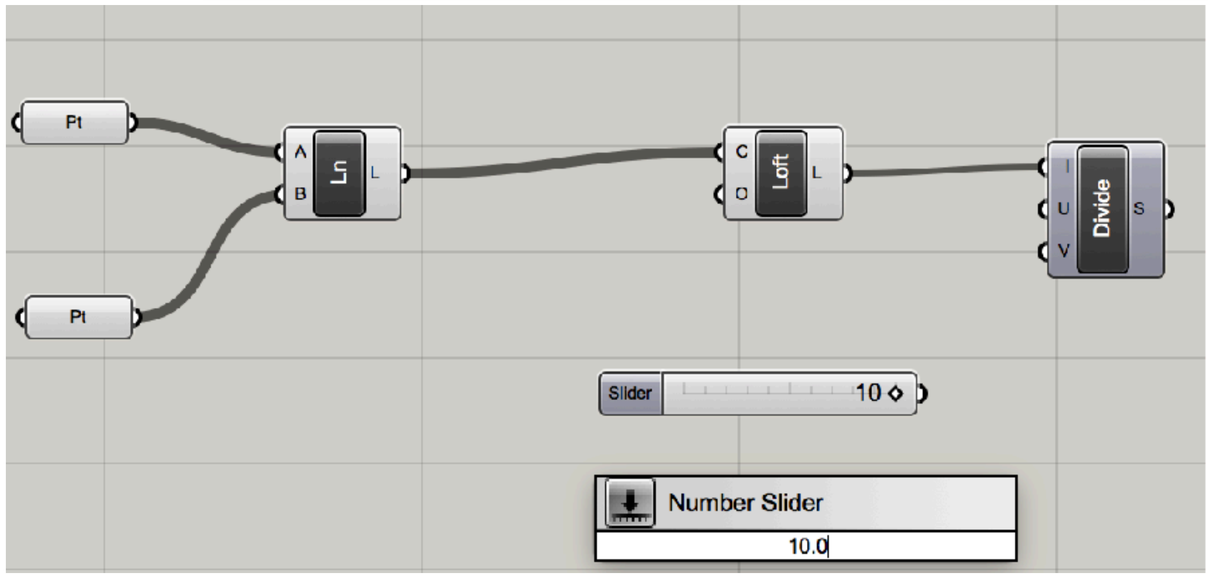
Exercise I

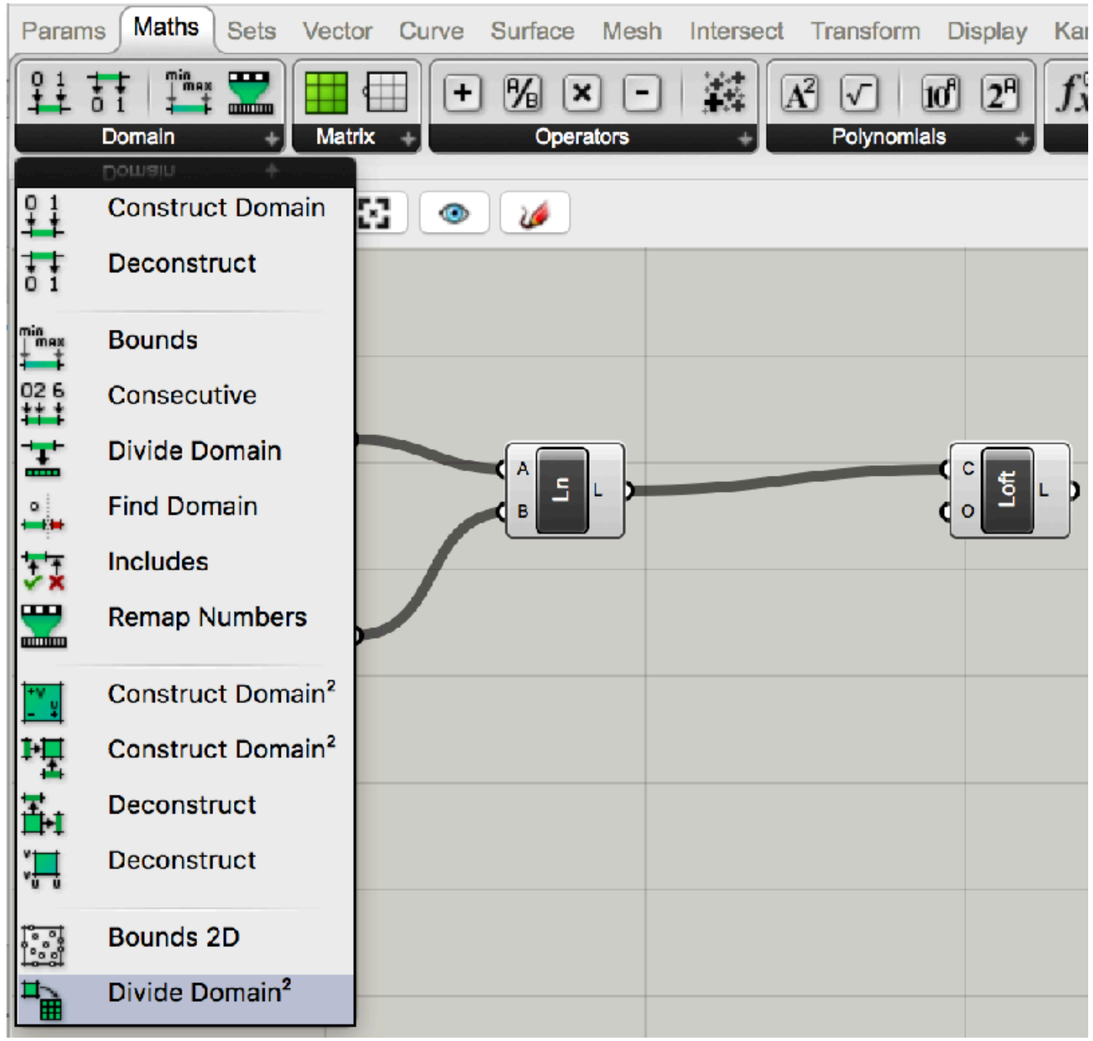
BoxMorph

a)



b)

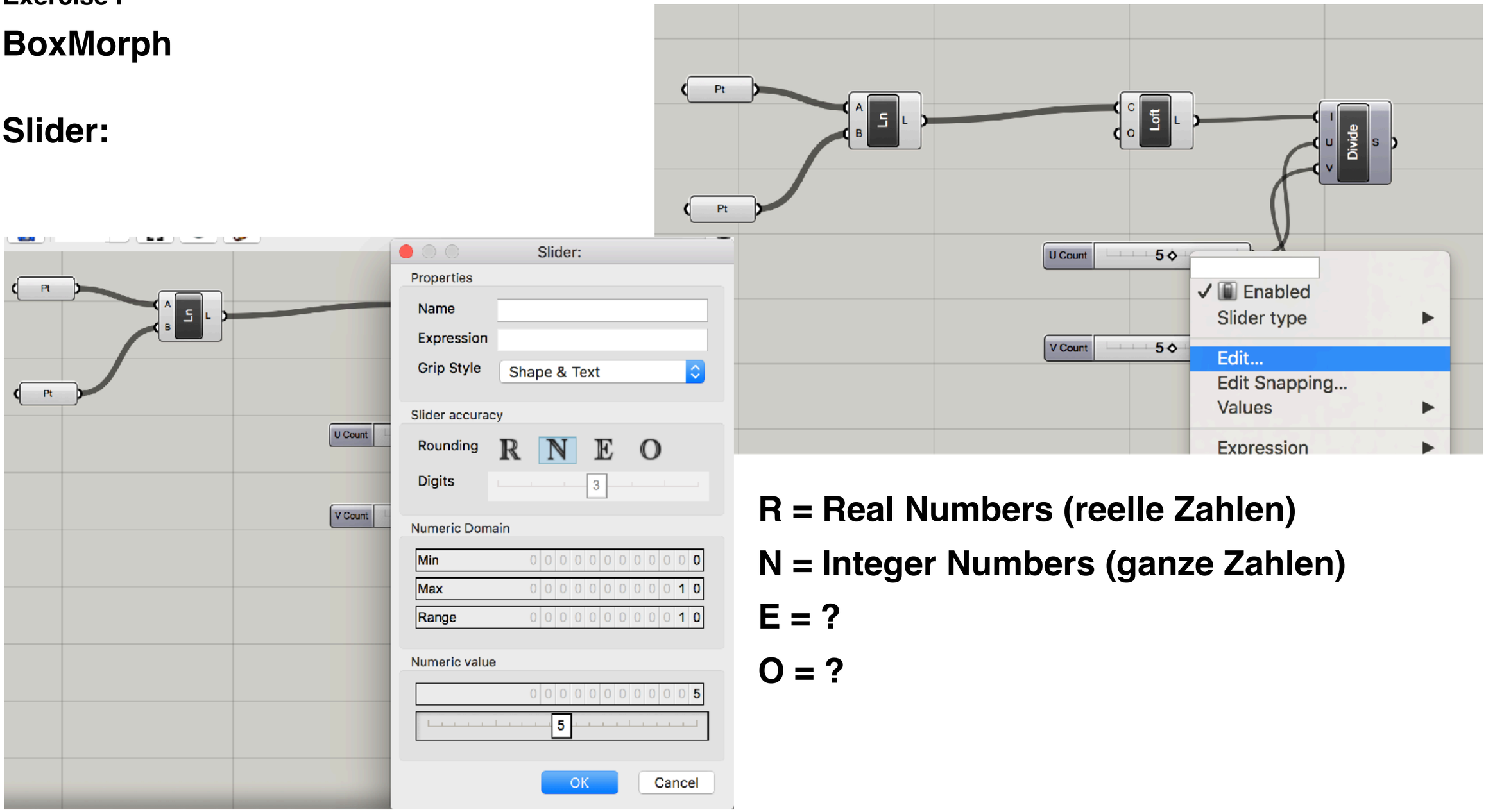




Exercise I

BoxMorph

Slider:



R = Real Numbers (reelle Zahlen)

N = Integer Numbers (ganze Zahlen)

E = ?

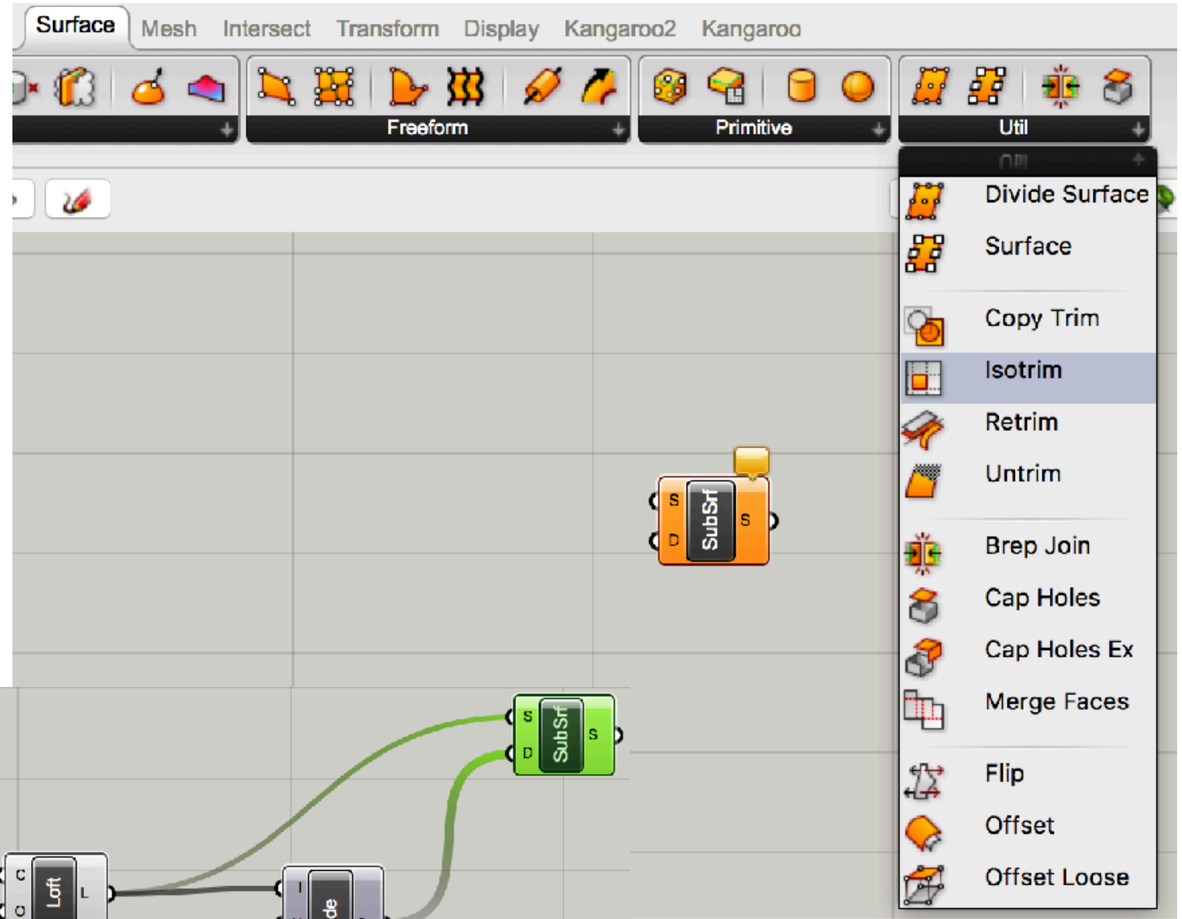
O = ?

Exercise I

BoxMorph

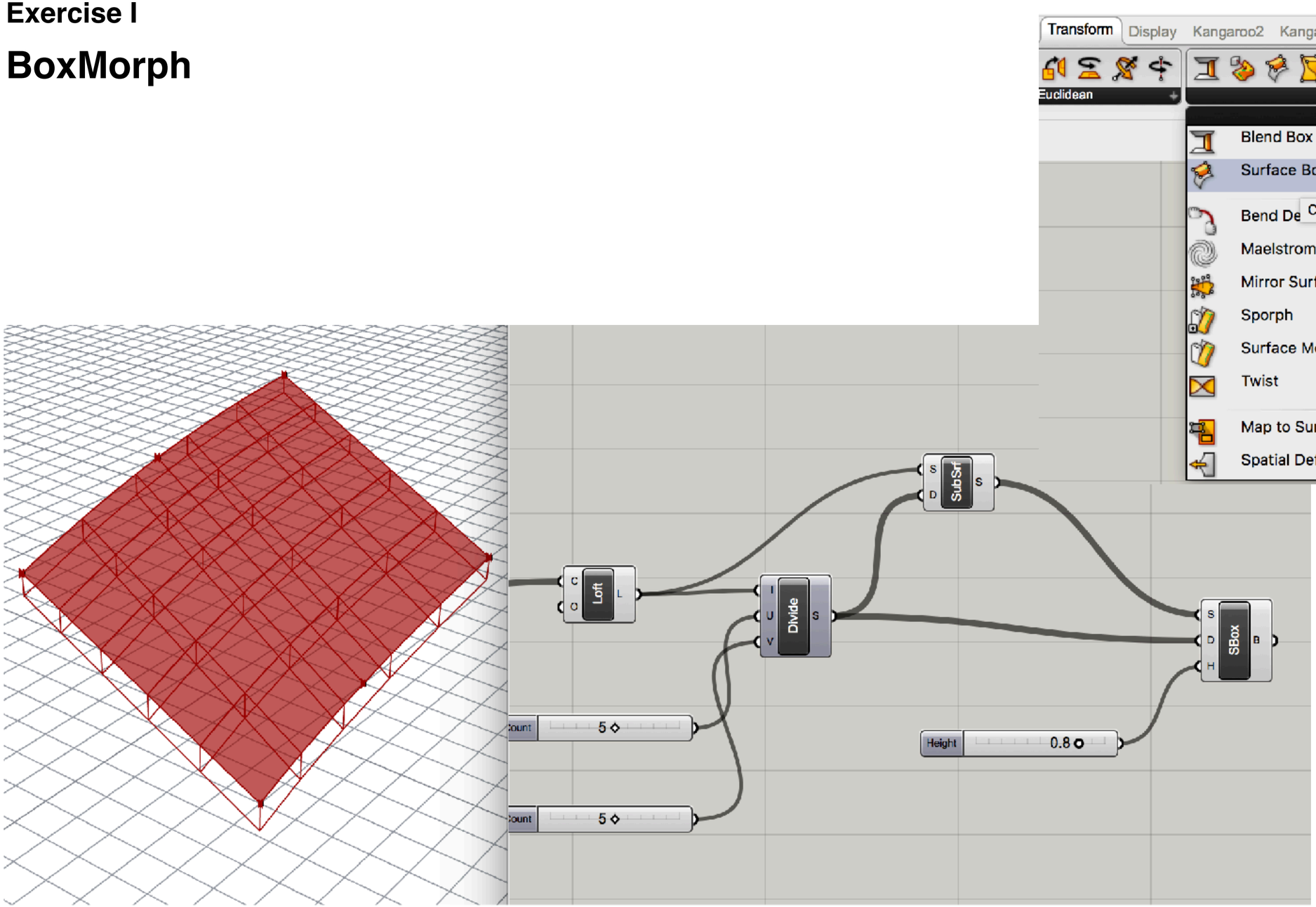
Creating SubSrf/Isotrim:

You can scale each resulting cell about it's center in relation to the overall surface.



Exercise I

BoxMorph

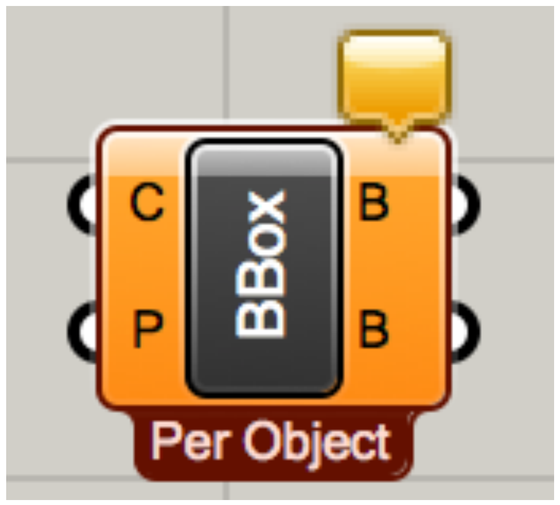


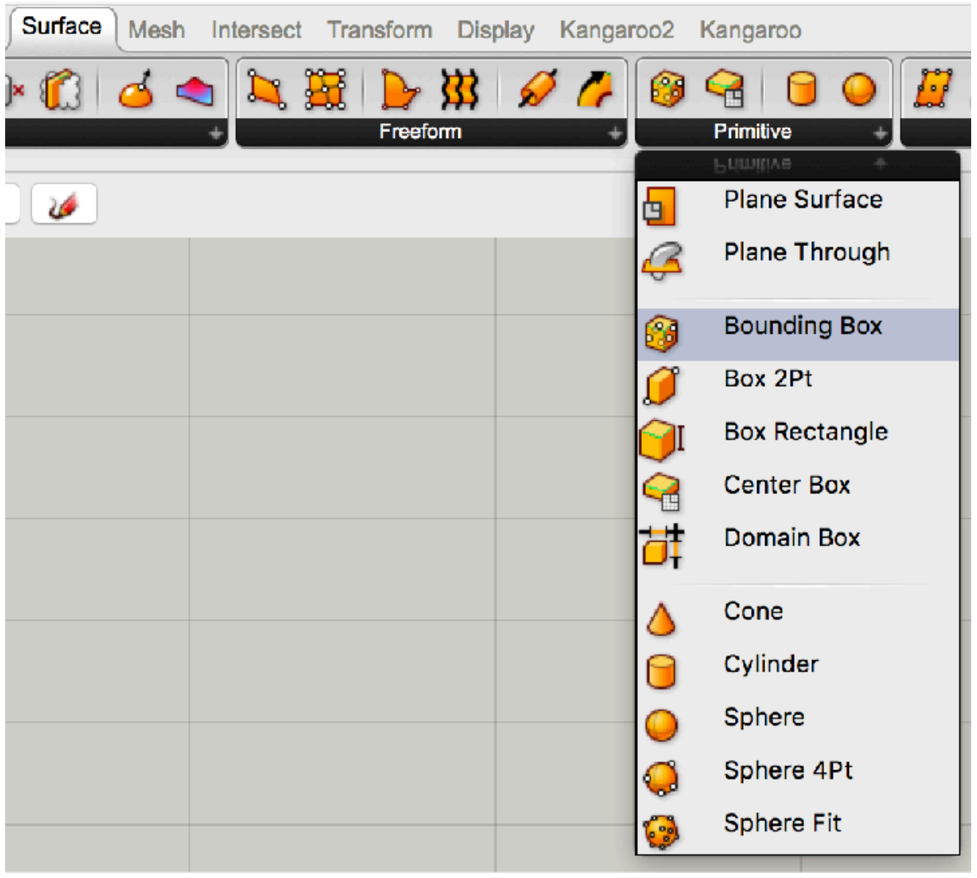
Exercise I

BoxMorph - Main Part

The Bounding Box is a reference Brep for the BoxMorph component. The Bounding Box creates a reference of the geometry that you wish to morph. i.e. the corners of that BBox will be mapped to the corners of the Target Boxes. Any Geometry within that BBox will then be mapped relative to the space of that box.

You can still change your base surface, pattern geometry, and the U and V subdivisions (sliders) to control any number of panels on a surface.



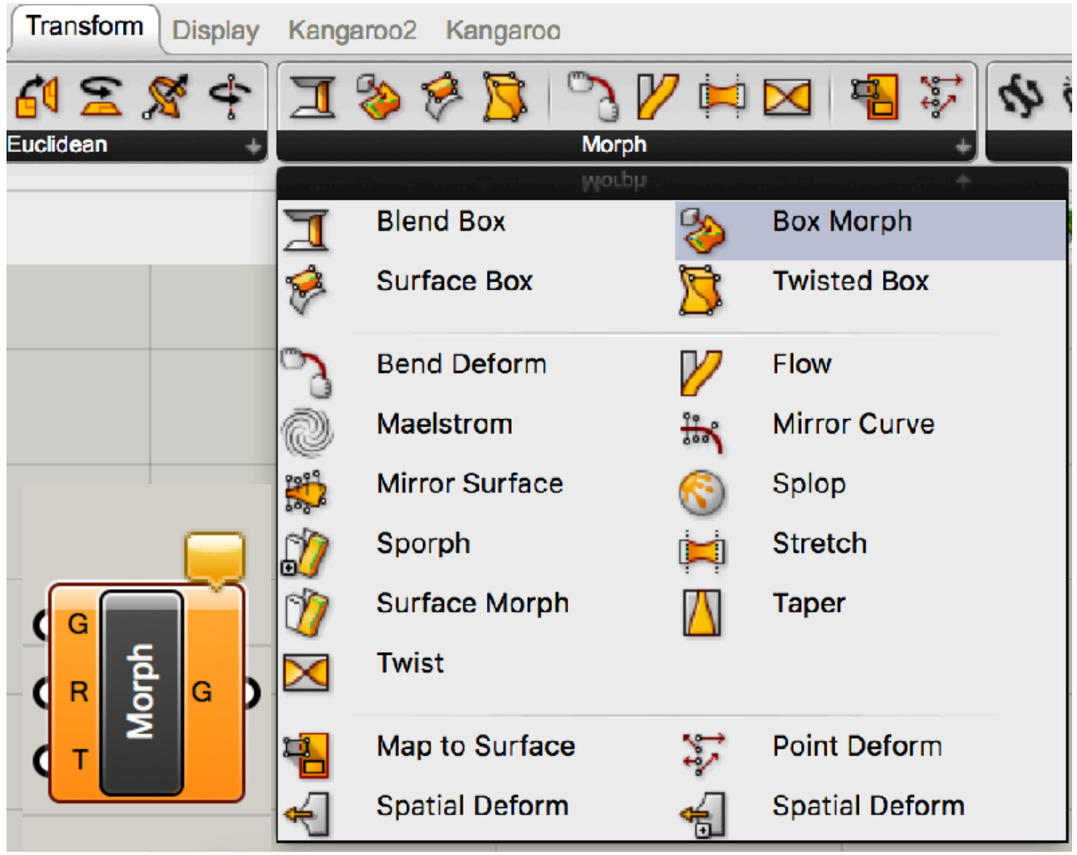


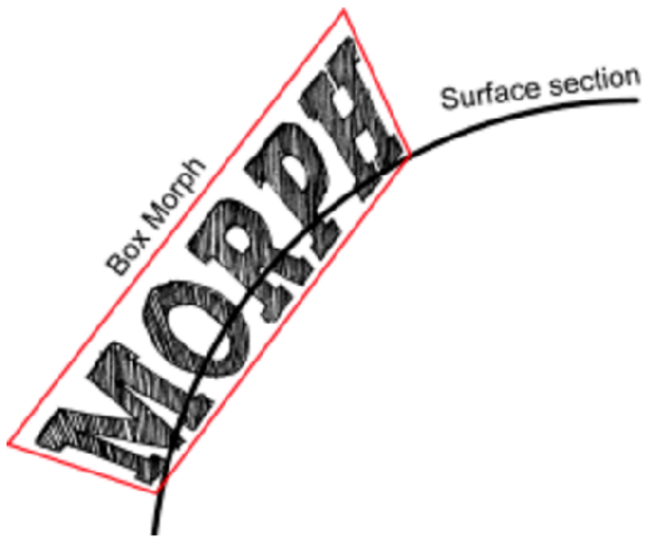
Exercise I

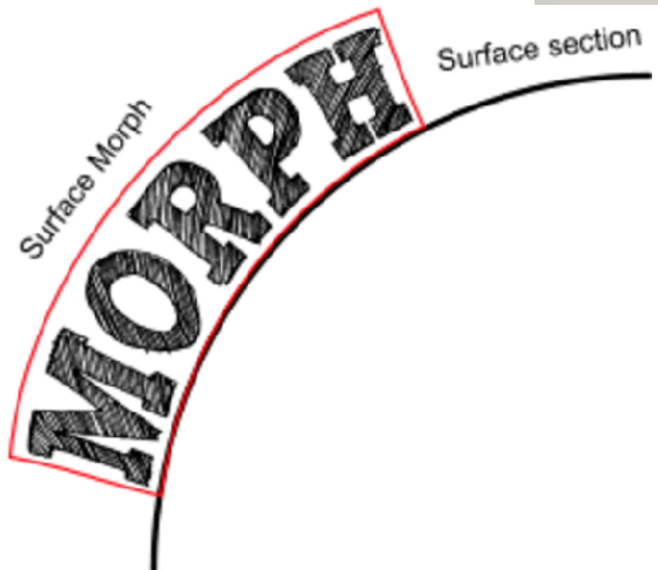
BoxMorph - Main Part

BoxMorph sorts the geometry to each containing box, so we can stretch and distort geometry in a very precise way.

The difference between Box Morph and Surface Morph is that a box morph only takes the 8 corners into account. It assumes a linear deformation between the corners. The Surface Morph component on the other hand actually pays attention to the surface curvature in



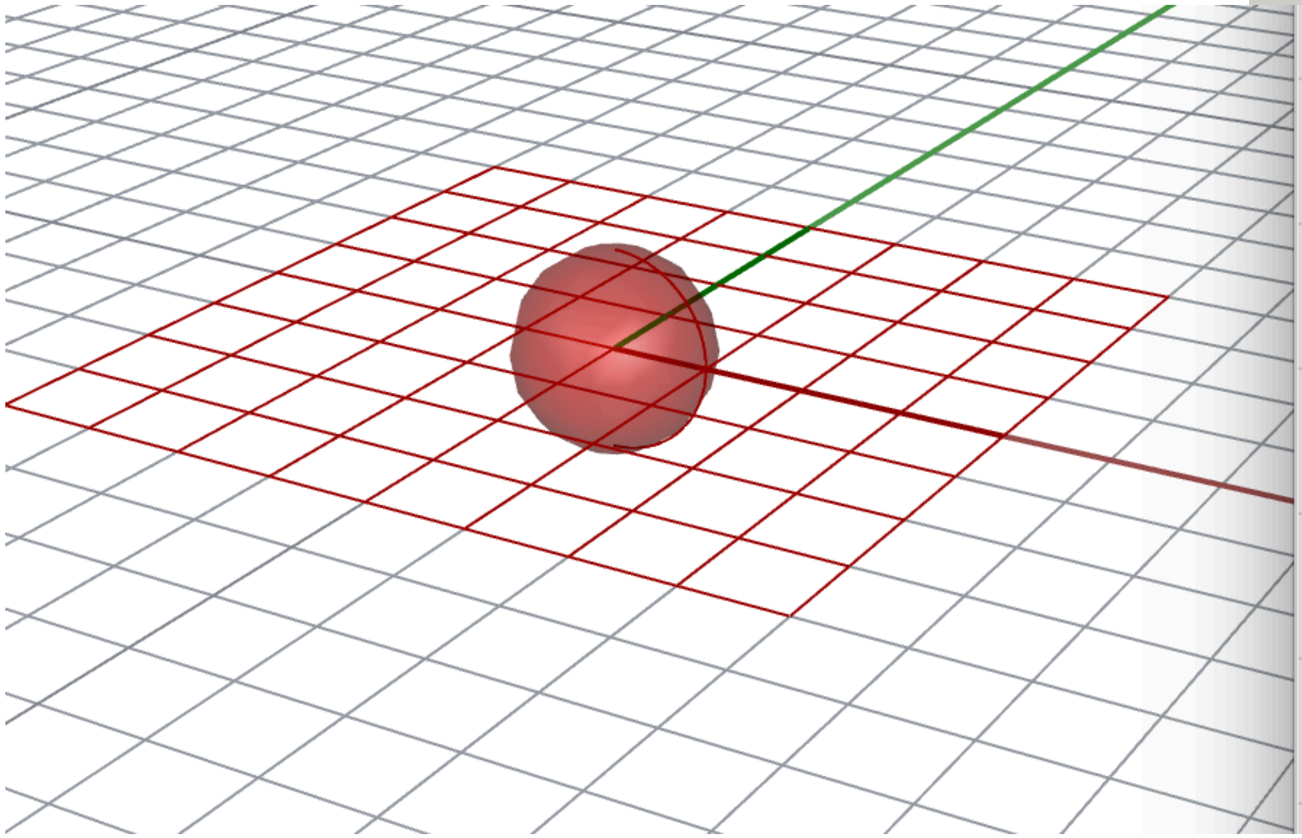


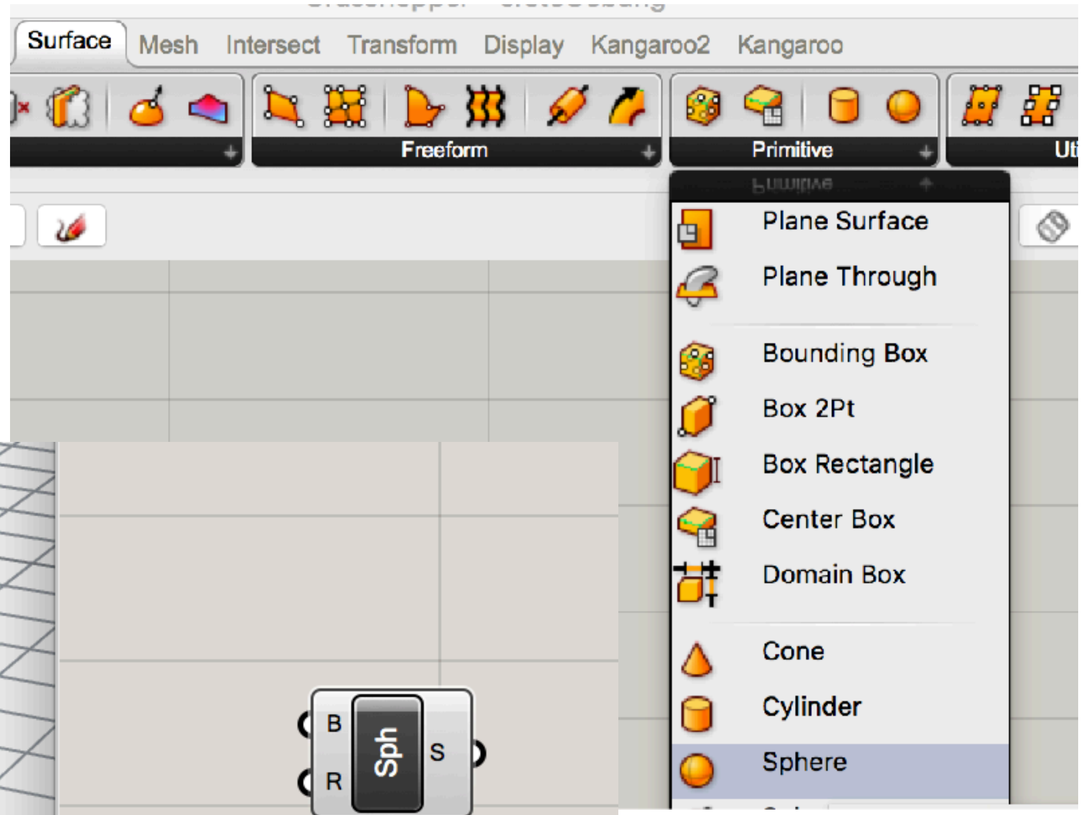


Exercise I

BoxMorph - Main Part

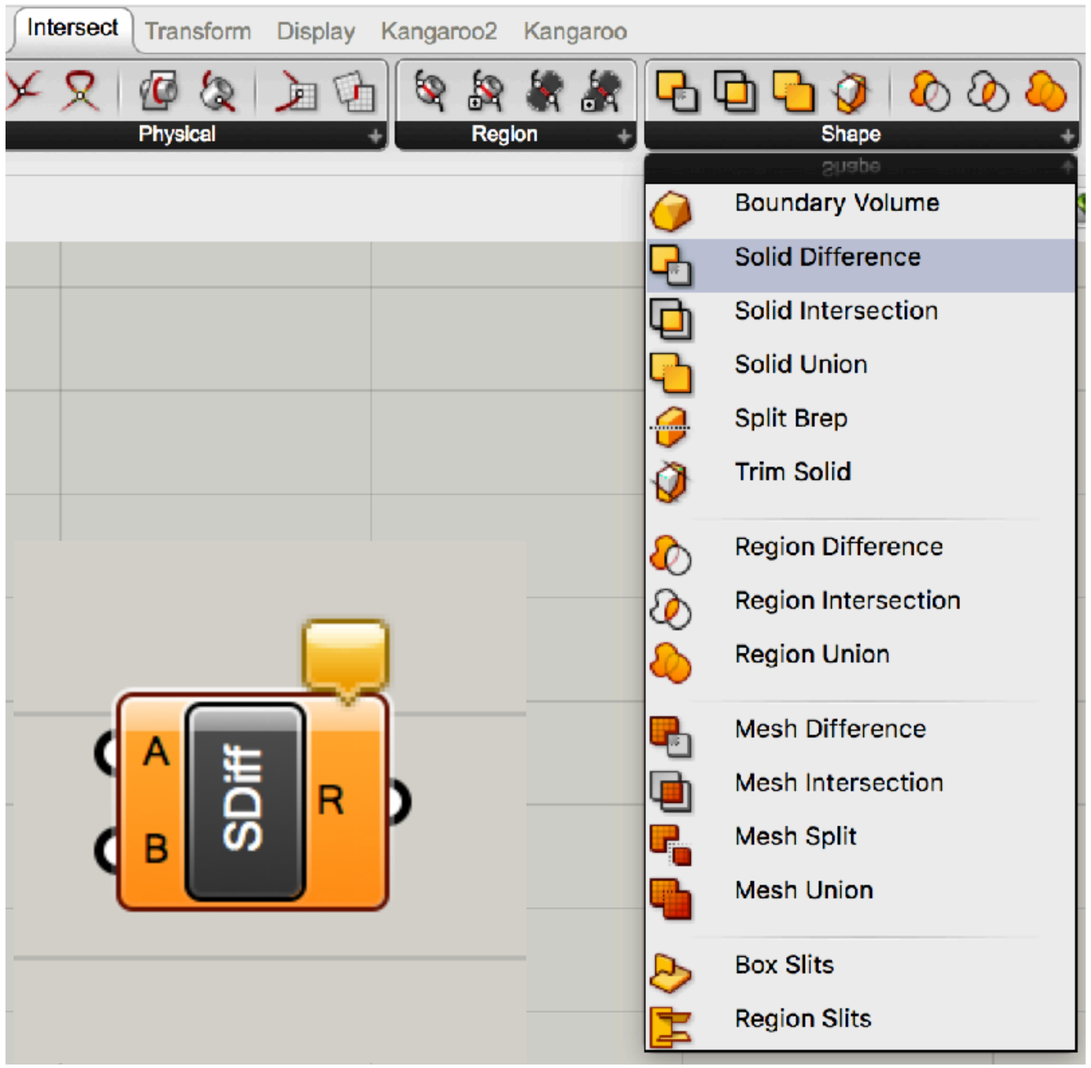
A Sphere is a definition to populate a surface.

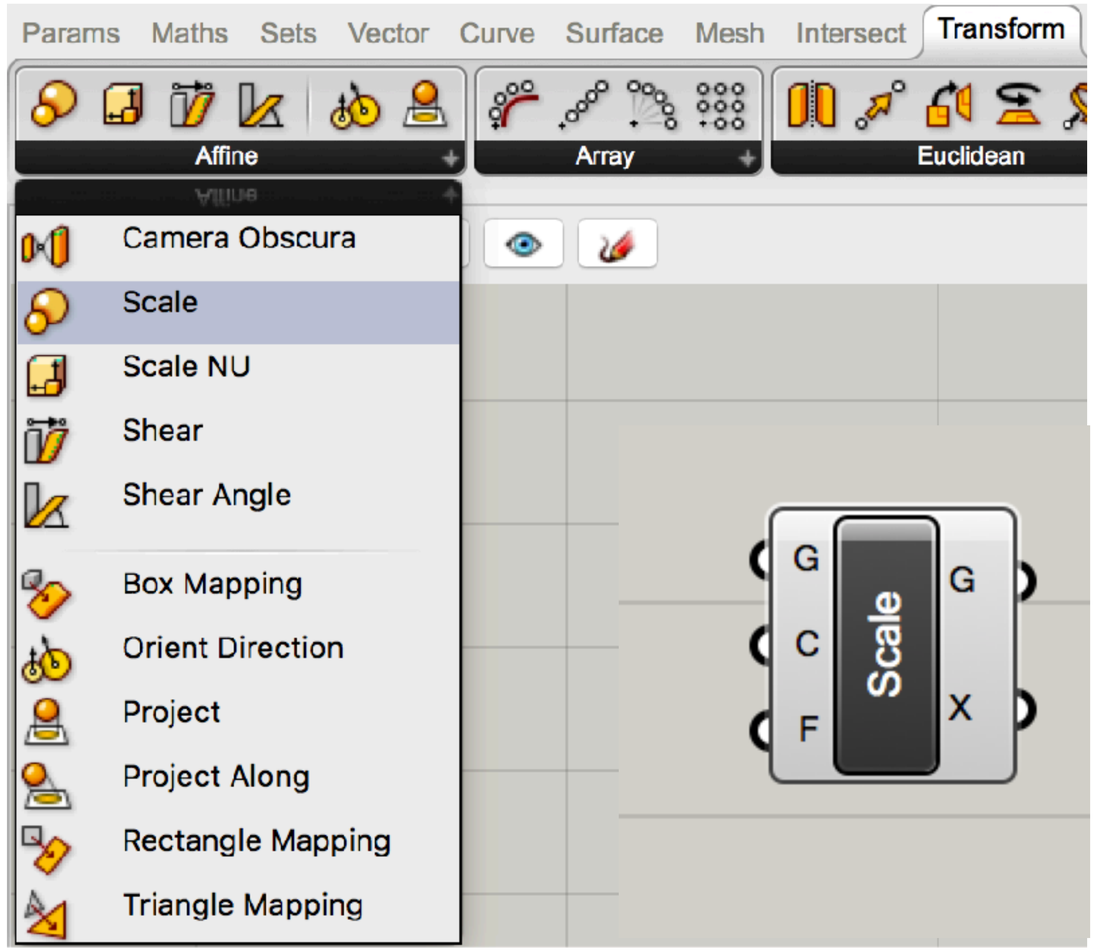




Exercise I

BoxMorph - Main Part



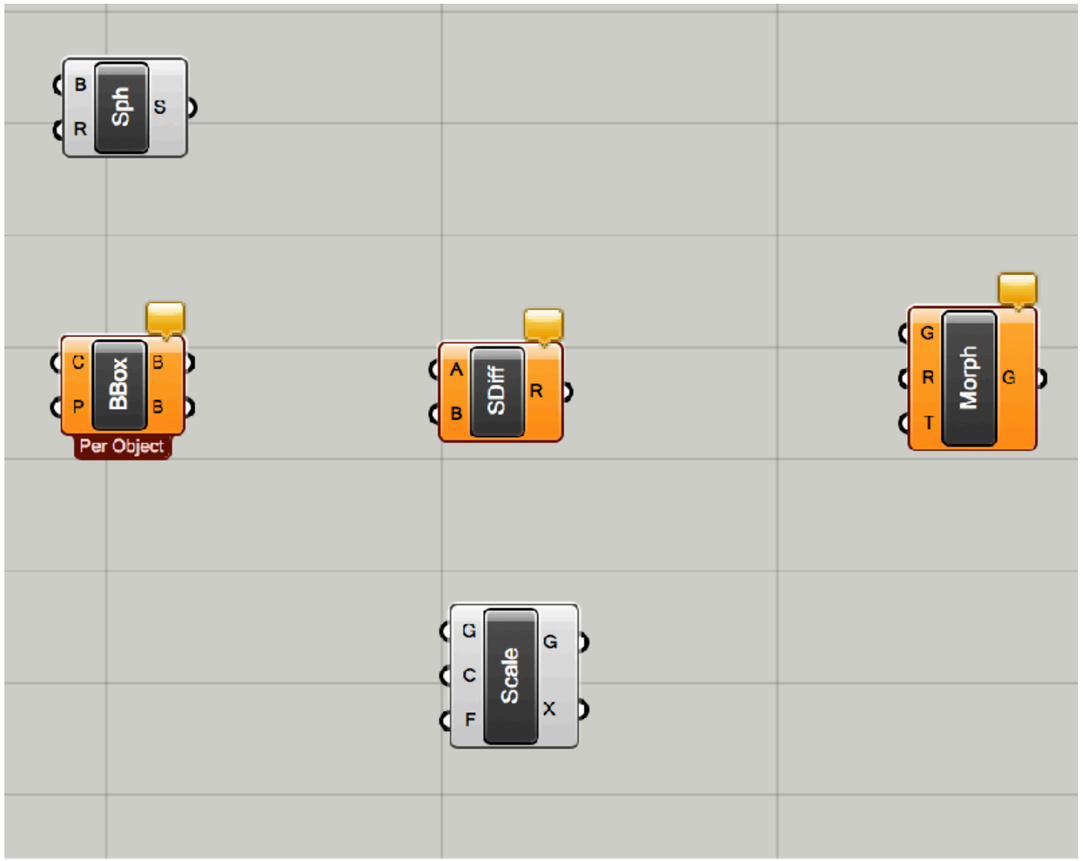


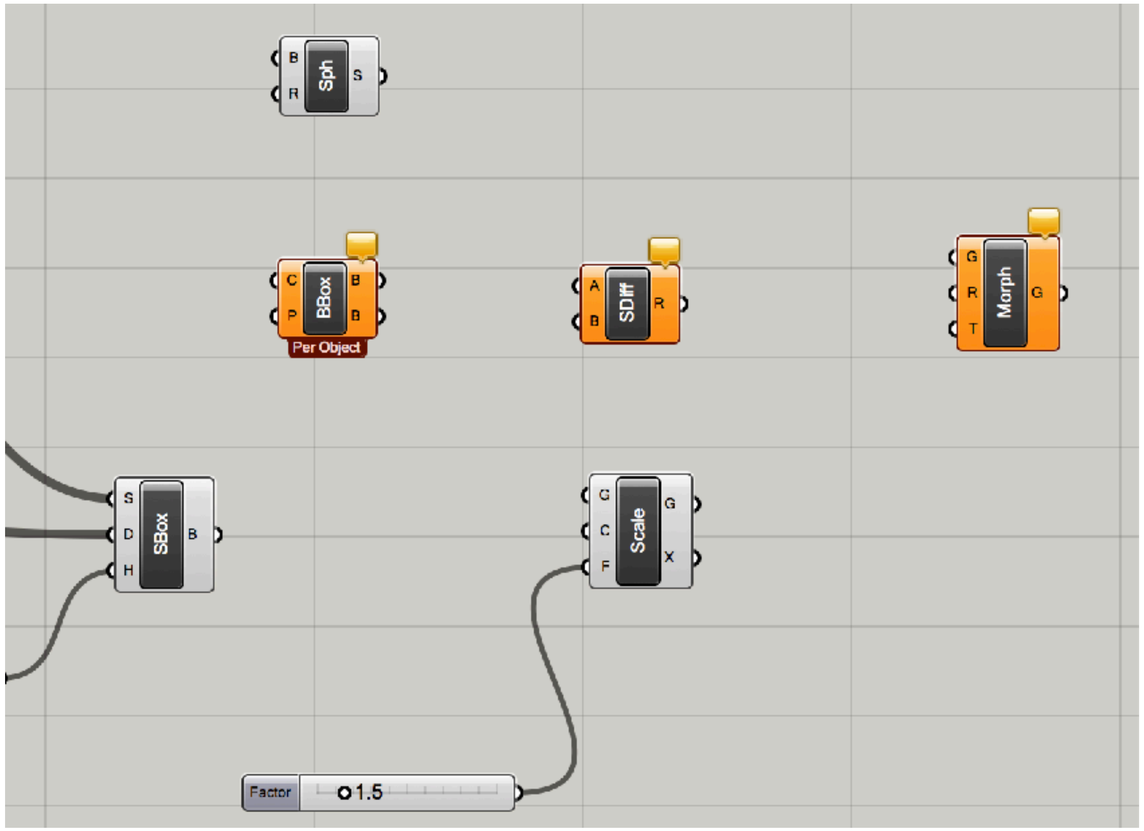
Exercise I

BoxMorph - Main Part

… that`s how your components should look like.

Now we add a Slider to the Scale.

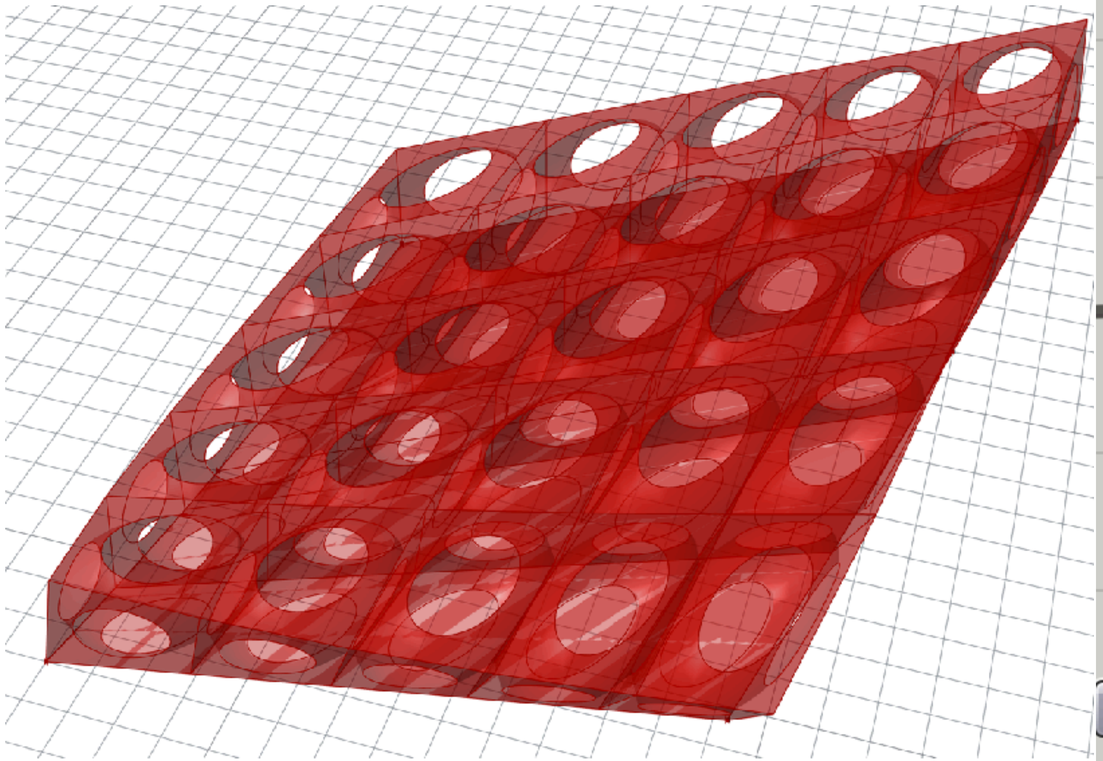


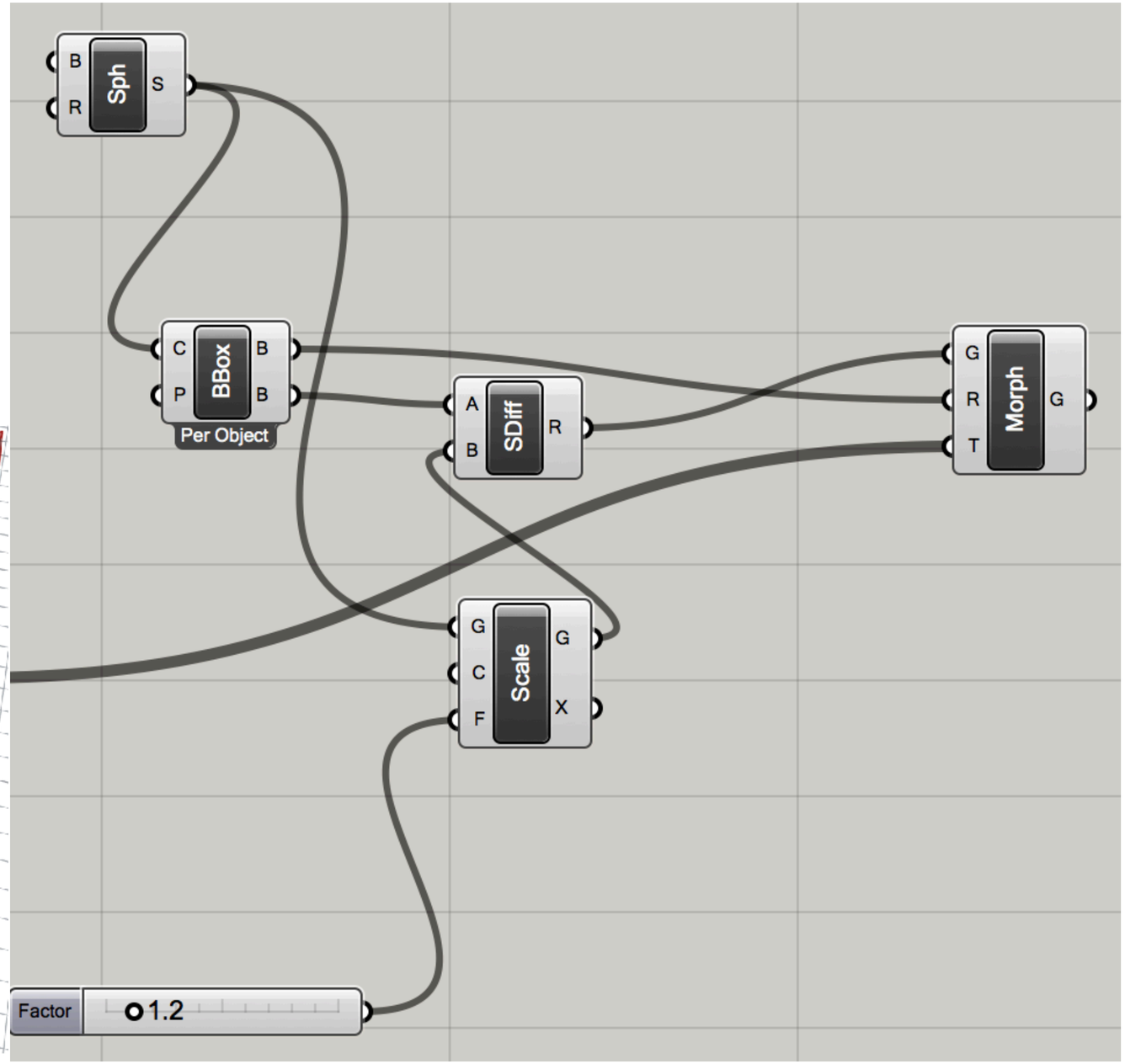


Exercise I

BoxMorph - Main Part

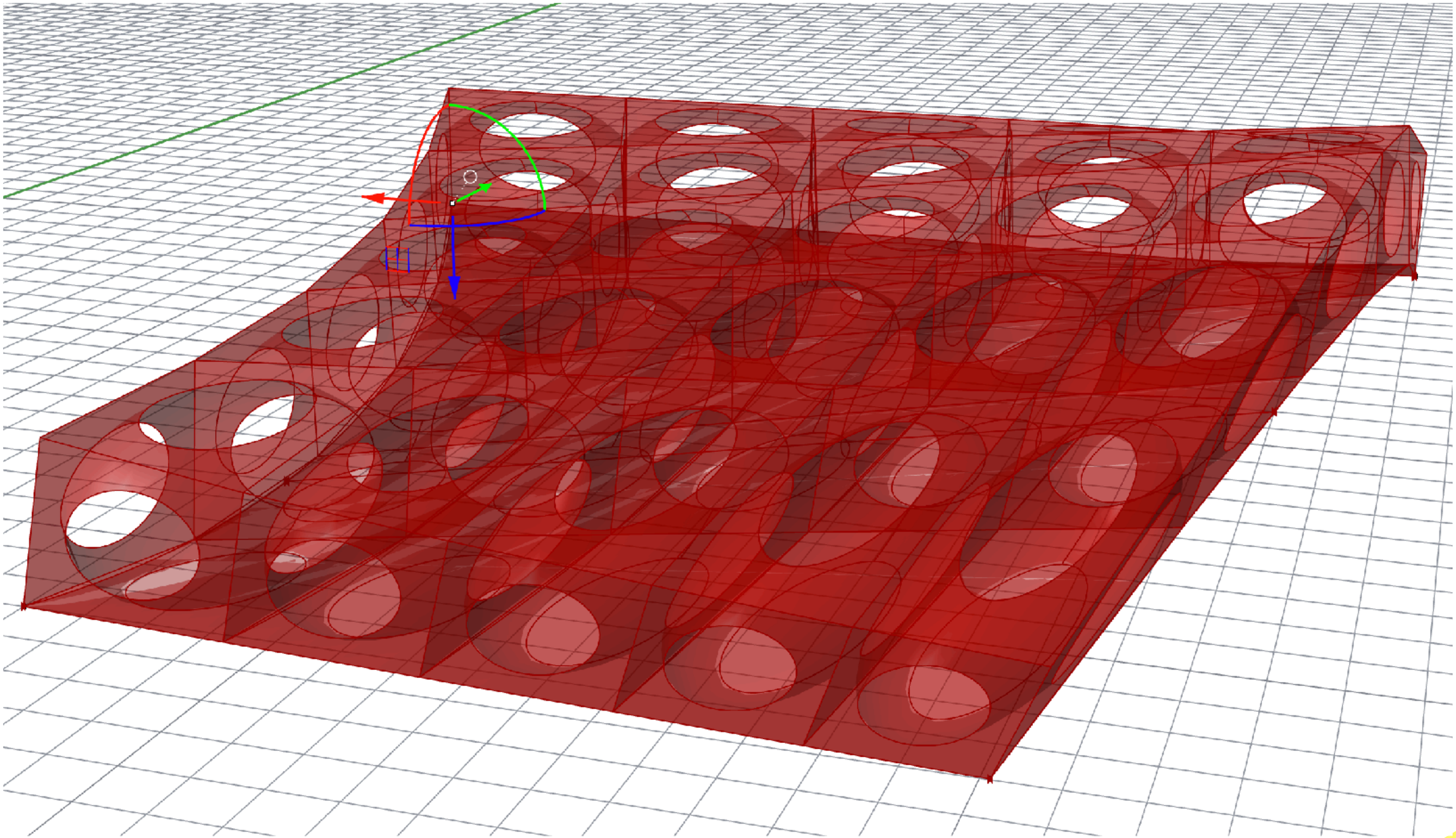
Finally we connect all the components with each other …





Exercise I

BoxMorph - Finish - play with Sliders, Gumball and X, Y, Z Coordinates



Overview

Bits & Atoms: Computer Aided Design

23.10.2017

Rhino Basics

30.10.2017

Rhino to Grasshopper

6.11.2017

Parametric Design with Grasshopper

Modelling Beispiele

Modelling

Bösendorfer GT

Übersicht

smartskull

Übersicht

Kaputt.R