

DOUBLY CURVED SURFACES MODELING

Giulio Piacentino 2008
mail@giuliopiacentino.com

PROJECT WORKFLOW

software used in current architectural discourse



Interaction POINTS, LINES, MESH



MESH,
NURBS

Parametrics

feature based modeling/parametric modeling



Modeling
constructive solid geometry

NURBS, MESH, SUBDIVS

Rendering tools



MESH

CNC, 3D Prints



MESH

WORKSHOP SCHEDULE

Nov 28

morning
2D/3D
interface
patterns
curves

afternoon
surfaces
construction

Intro,
paper
model

Dec 5

morning
Complex
surfaces
manipulate
solids

afternoon
your project
help with
choices

Solids
modeling

Before we start:

feel free to **stop** me
whenever you feel it is
necessary or you don't have
everything **clear!**

THE RHINO INTRO

Rhinoceros Help

Nascondi Precedente Stampa Opzioni

Contenuto | **Indice** | Cerca | Preferiti

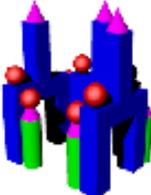
Digitare la parola chiave da cercare:

Elenca argomenti

Selezionare l'argomento da visualizzare:

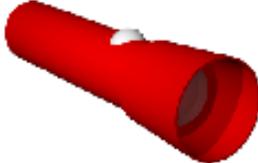
Visualizza

Tutorials



Introduction

You will use Rhino's basic navigation tools and shaded viewports to create basic geometric objects. These short tutorials give you a start using Rhino's navigation, editing and surface creation tools.



Build a model with solids

While making a simple model of a flashlight you will create accurately sized solids, combine the solids together, and render.



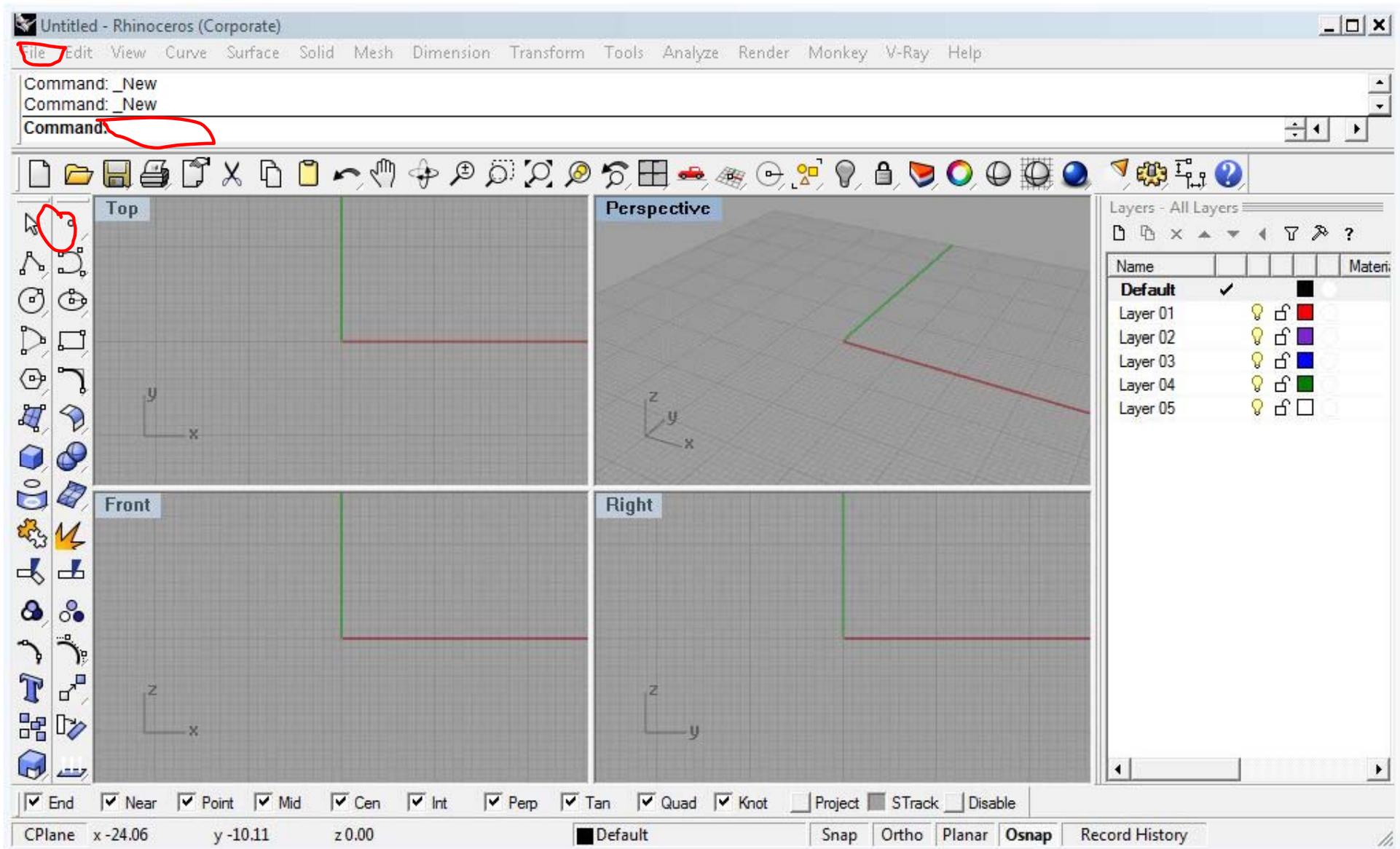
Create organic shapes

Focusing on flexible, free-form shapes, you will create and edit surfaces by manipulating control points, draw curves, blend between surfaces to create a rubber duck toy and then render the model with lights.

Next >

INTERFACE

INTERFACE



You can give instructions to Rhino by: menu items, command line and buttons.

For architectural practice:

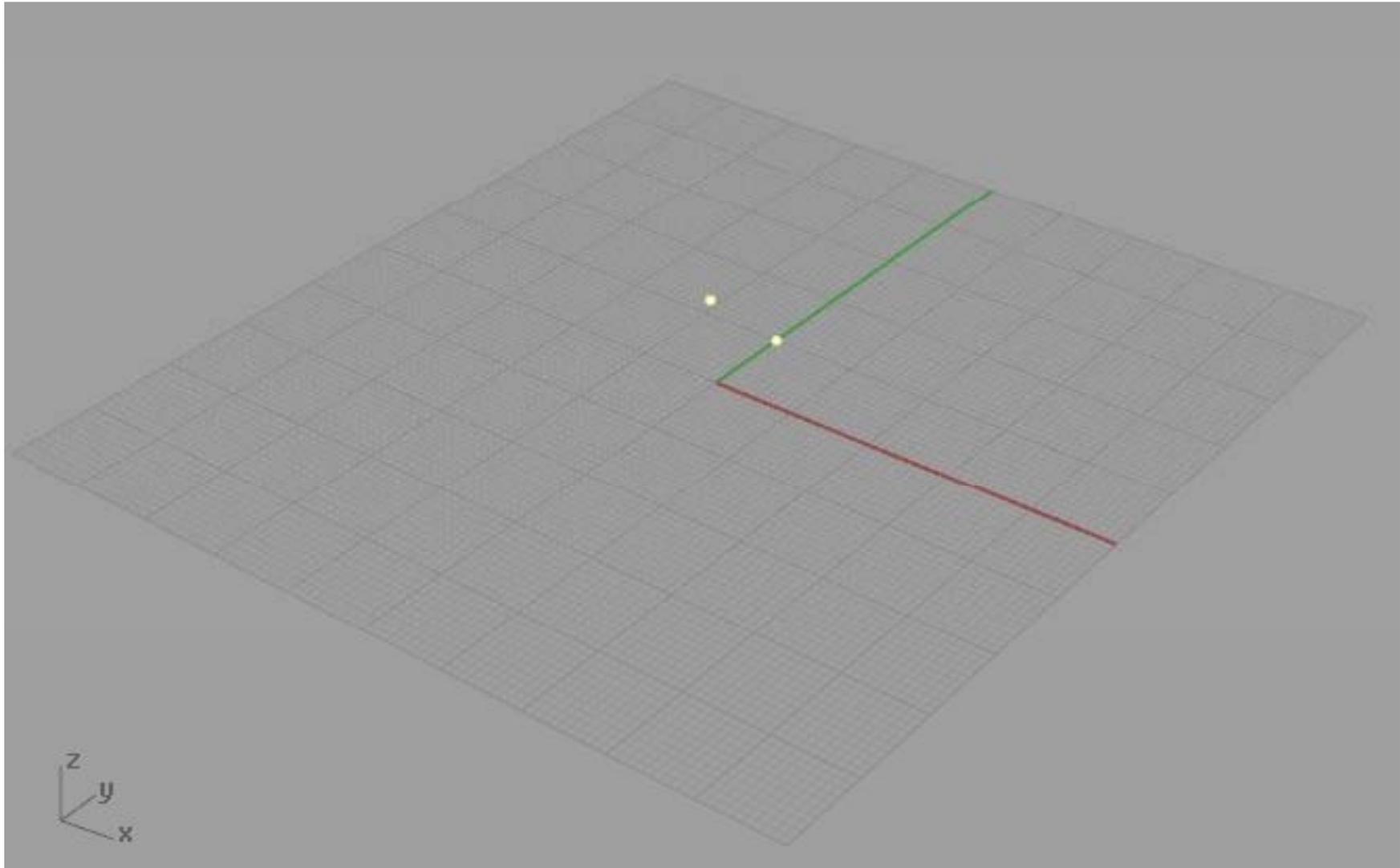
use the **meter** modelspace
with **small** objects
(recommanded),

or the **centimeter**
modelspace with **large** objs.

Thirdly, **mm** (large).

INTERFACE

Construction planes (**Cplane**) are view-dependent (one in each window)



GLOBAL: symbols

LOCAL: X is RED and Y is GREEN Z through right-hand rule

SHORTCUTS

F1 (help)

F2 (commandHistory)

F3 (properties)

F6 (show camera)

F8 (ortho)

F10 (pointsOn)

F11 (pointsOff)

Alt

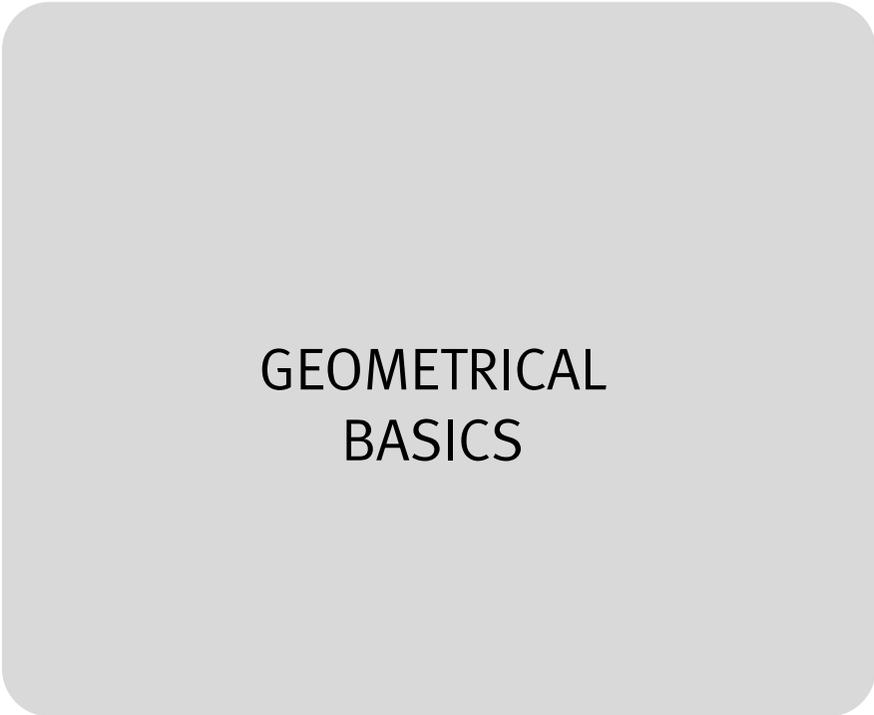
- temporarily disable oSnap
- copy object instead of drag

Ctrl

- elevator mode

Shift

- temporarily toggle ortho



GEOMETRICAL
BASICS

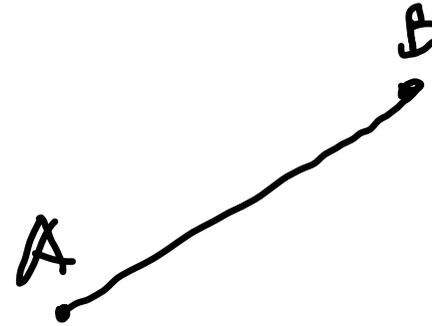
POINTS AND LINES

points and straight lines are the easiest set of objects in rhino

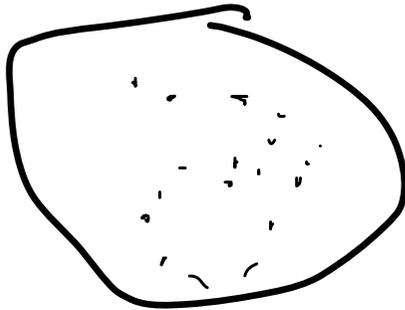
A



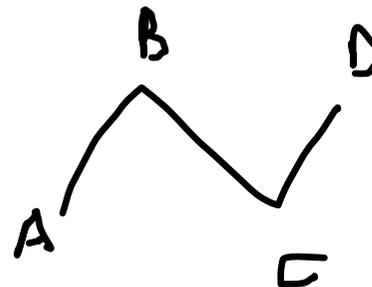
B



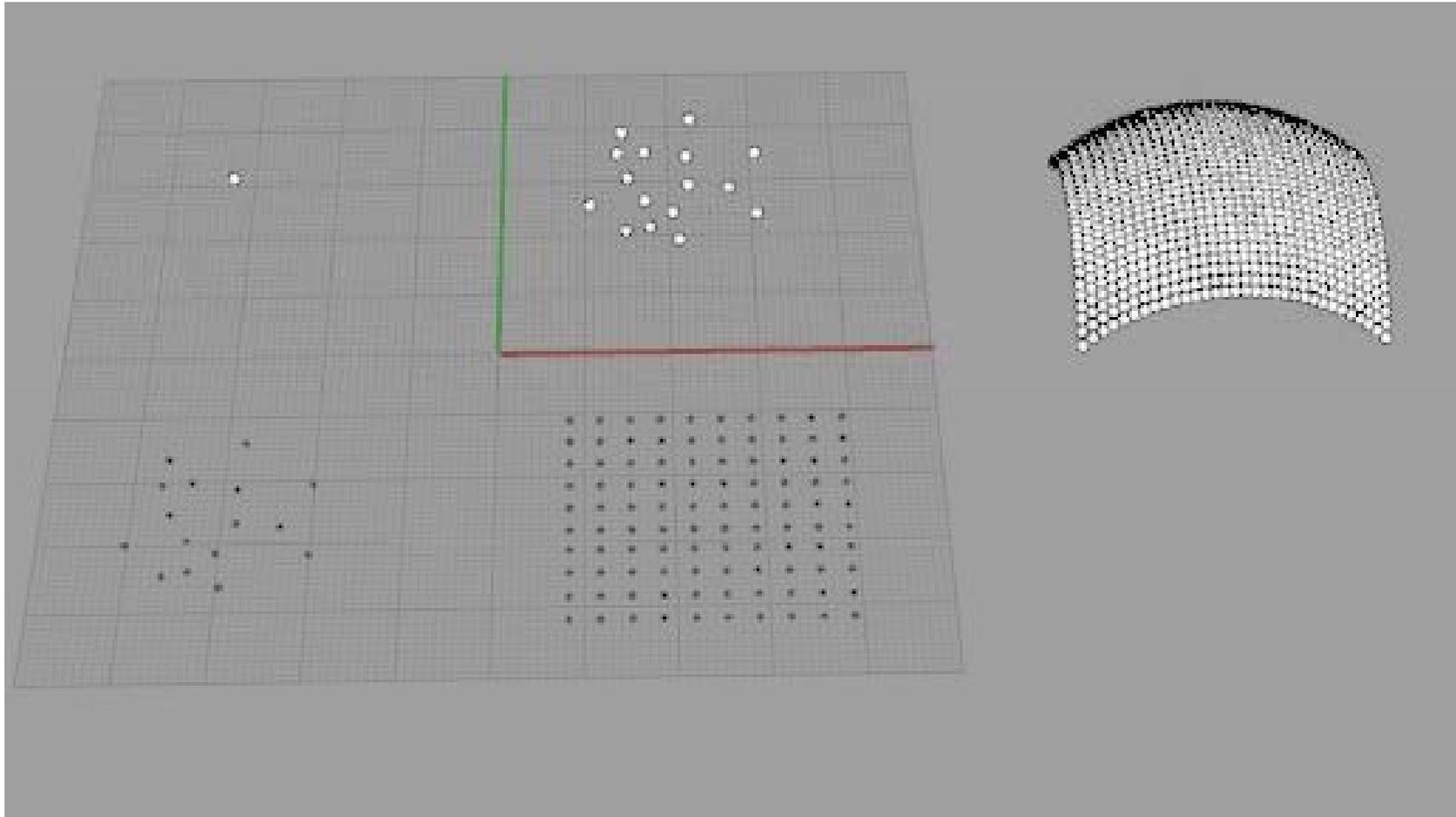
points can be grouped into pointClouds



lines can be joined into polylines.
polylines are still **MANY** lines.



POINTS AND LINES



point
pointCloud

points
pointGrid

drapePt

POINTS

```
0,0,0  
0,0,10 0,0,20 0,0,30
```

```
0,0,0  
r0,0,10 r0,0,10 r0,0,10
```

When you give coordinates , you can add an **r** to make the coordinate relative to the last given point.

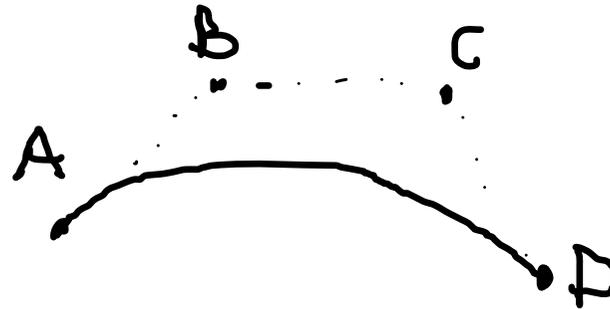
CURVES

curves are renders of mathematical equations
curves in rhino are NURBS curves (Non-Uniform Rational B-Splines)

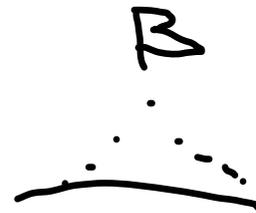
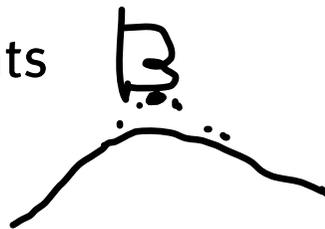
they have: a degree

X^10

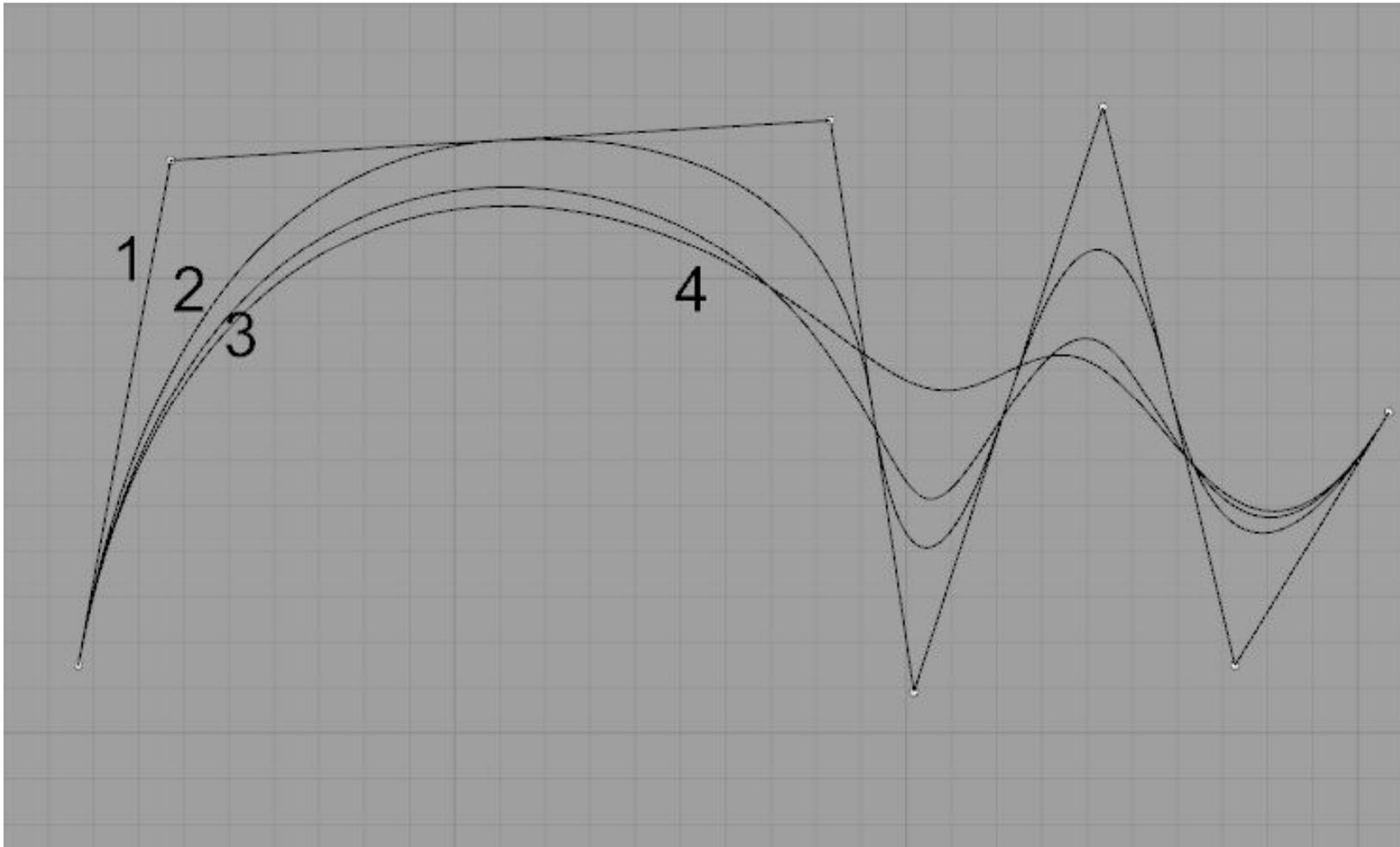
they have: control points



they have: weights

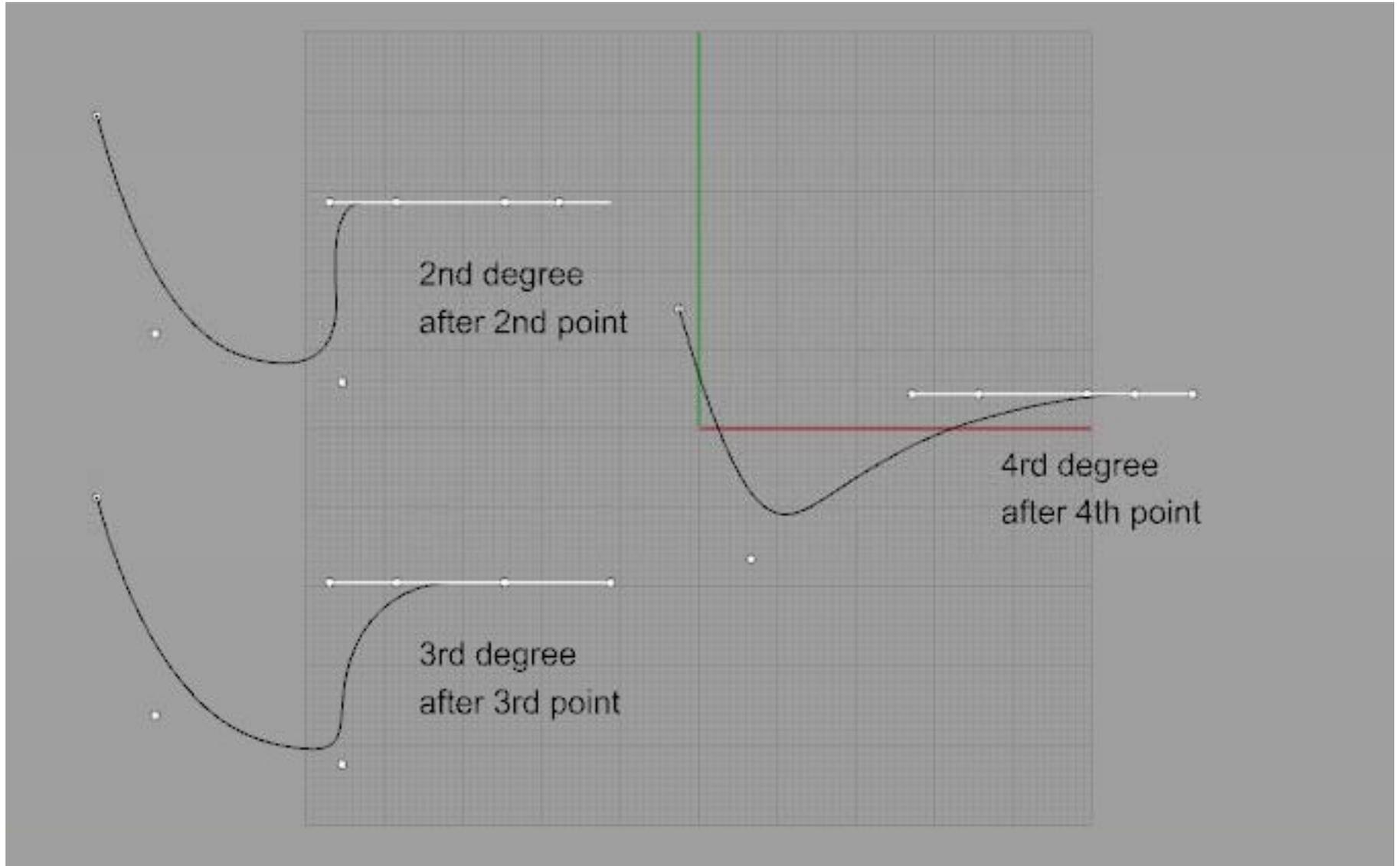


CURVES



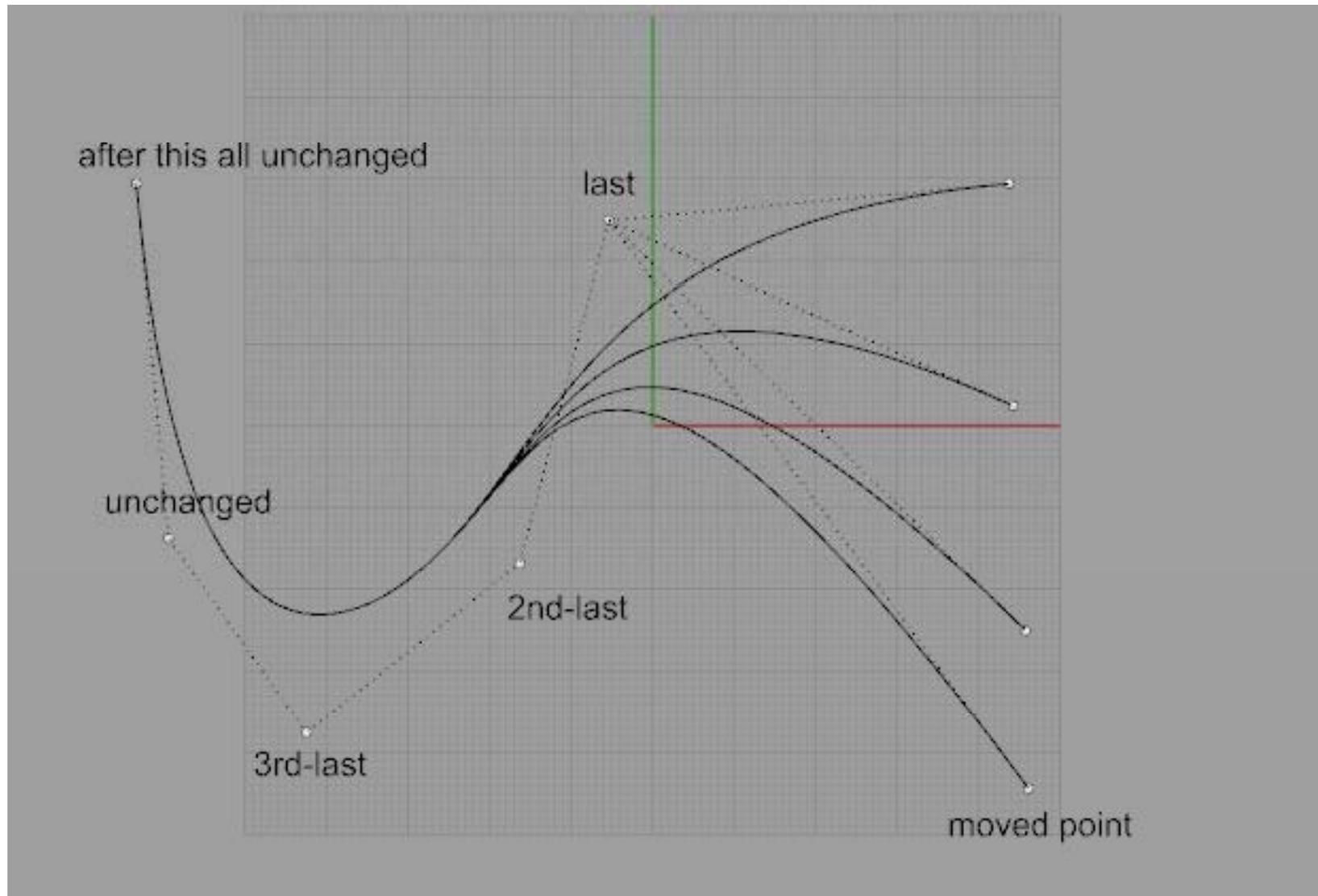
draw 3 or more curves with the same control points and different degrees.

CURVES



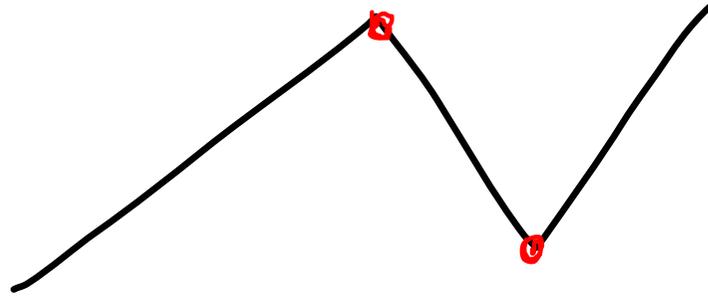
you get a straight line after... as many points edit point in a row as the degree of the curve.

CURVES

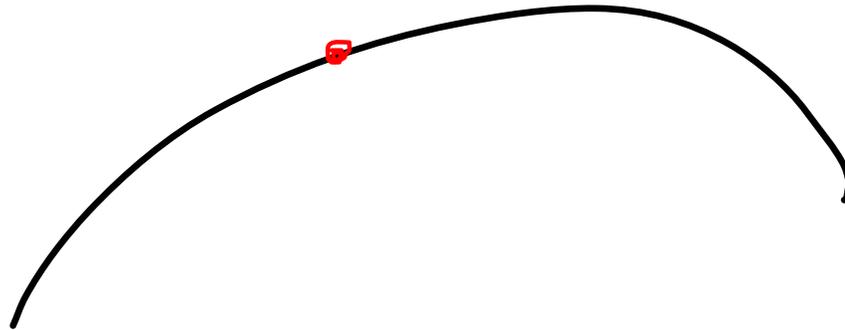


If we use any third degree NURBS curve,
and we modify any set of points,
the first curve segment to coincide again will be the one between the 2nd-last and the 3rd last modified point. After that, all will coincide again.

CURVES

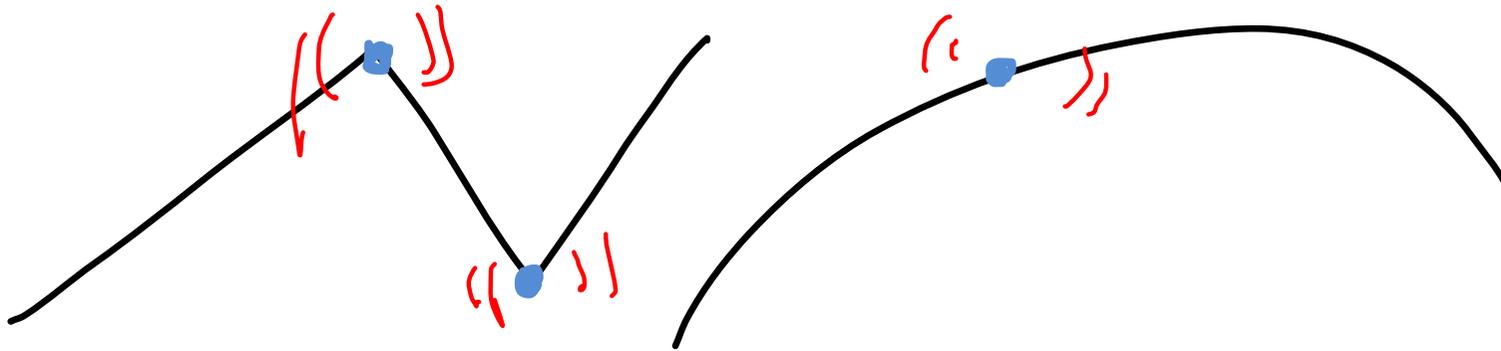


If the curve has kinks, you can **explode** it into smaller parts



Otherwise you can divide it with **split**

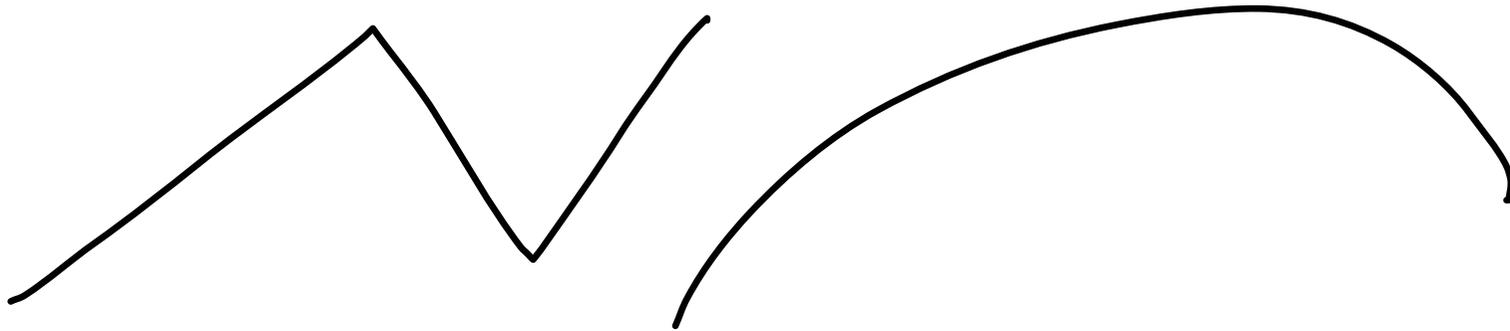
CURVES



You can join them back together, with **Join**, but they will be transformed into **polyLines** and **polyCurves** (depending on the degree).

Try the command **Properties** to inspect the outcome.

CURVES

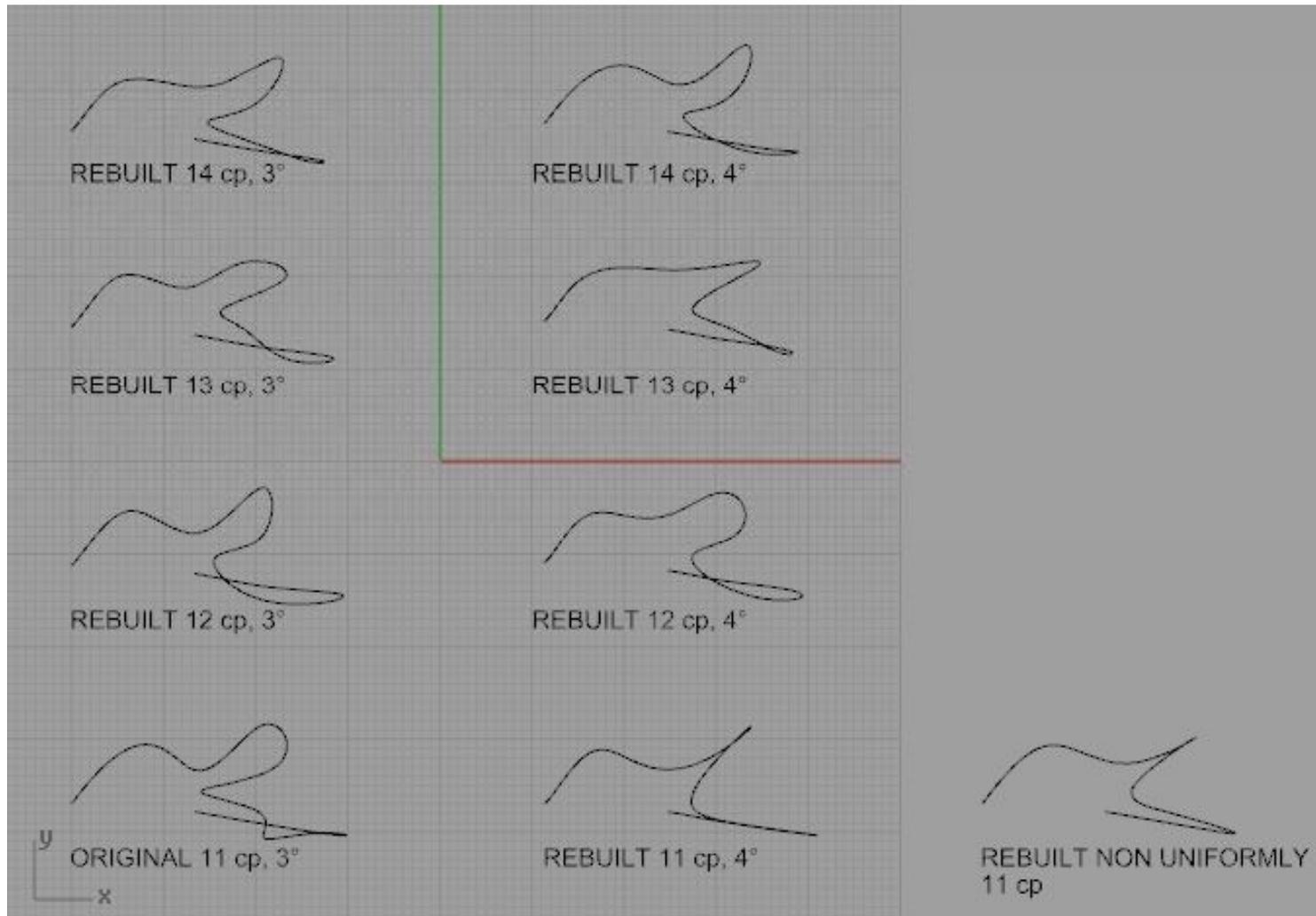


In some situations we may need to **rebuild** these polycurves, so that they can return to be a single entity.

A similar command to try is also **rebuildCrvNonUniform**.

Both these commands reconstruct selected curves or surfaces to a specified degree and with a specified number of control points.

CURVES



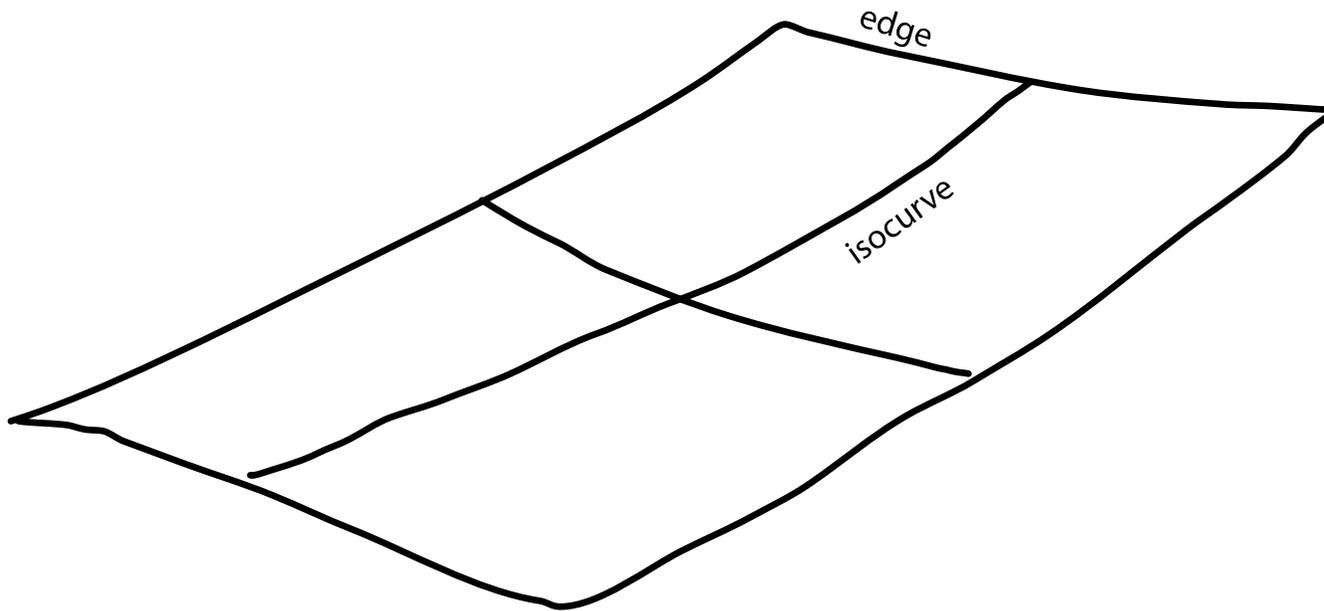
Not always it is easy to rebuild a curve to something similar, if we don't add many more points.

IN A INSTANT

	continuity type	adjacency analysis	inheritance	math. analysis	aestetical impression
disjoint (no continuity)			-	$f(x) \neq g(x)$ also: domains are not adjacent	you need to move the end points to join.
position			Go	$f(x) = g(x)$	curves will be continuous, but will probably have a visible crease.
tangency (position variation)			$G1 \gg Go$	$f'(x) = g'(x)$	curves will appear smooth. their reflection, though, could have sudden changes.
curvature (tangency variation)			$G2 \gg G1 \gg Go$	$f''(x) = g''(x)$	reflections will change smoothly as well. ambient shadows will be gradual.
variation (generally)			$Gn \gg Gn-1 \gg Gn-2 \gg \dots$	$f^{(n)}(x) = g^{(n)}(x)$	//

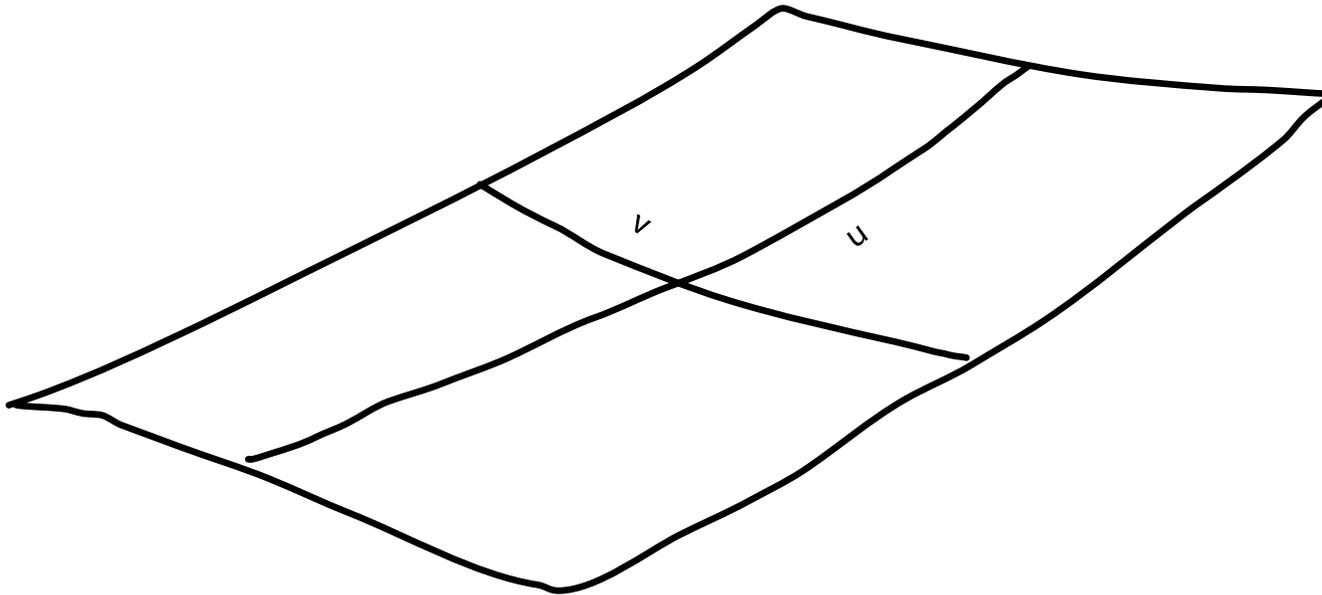
SURFACES

besides points and curves,
NURBS surfaces are the basics of geometry in Rhino and can be used to draw both flat things and objects



An isoparametric curve (**isocurve**) is a curve of constant u - or v -value on a surface. Rhino uses isocurves and surface edge curves to visualize the shape of a NURBS surface. By default isocurves are drawn at knot locations. If the surface is a single knot-span surface like a simple rectangular plane, isocurves are drawn also in the middle of the surface.

SURFACES



NURBS are inherently rectangular (U / V)

also surfaces can be rebuilt.

SURFACES

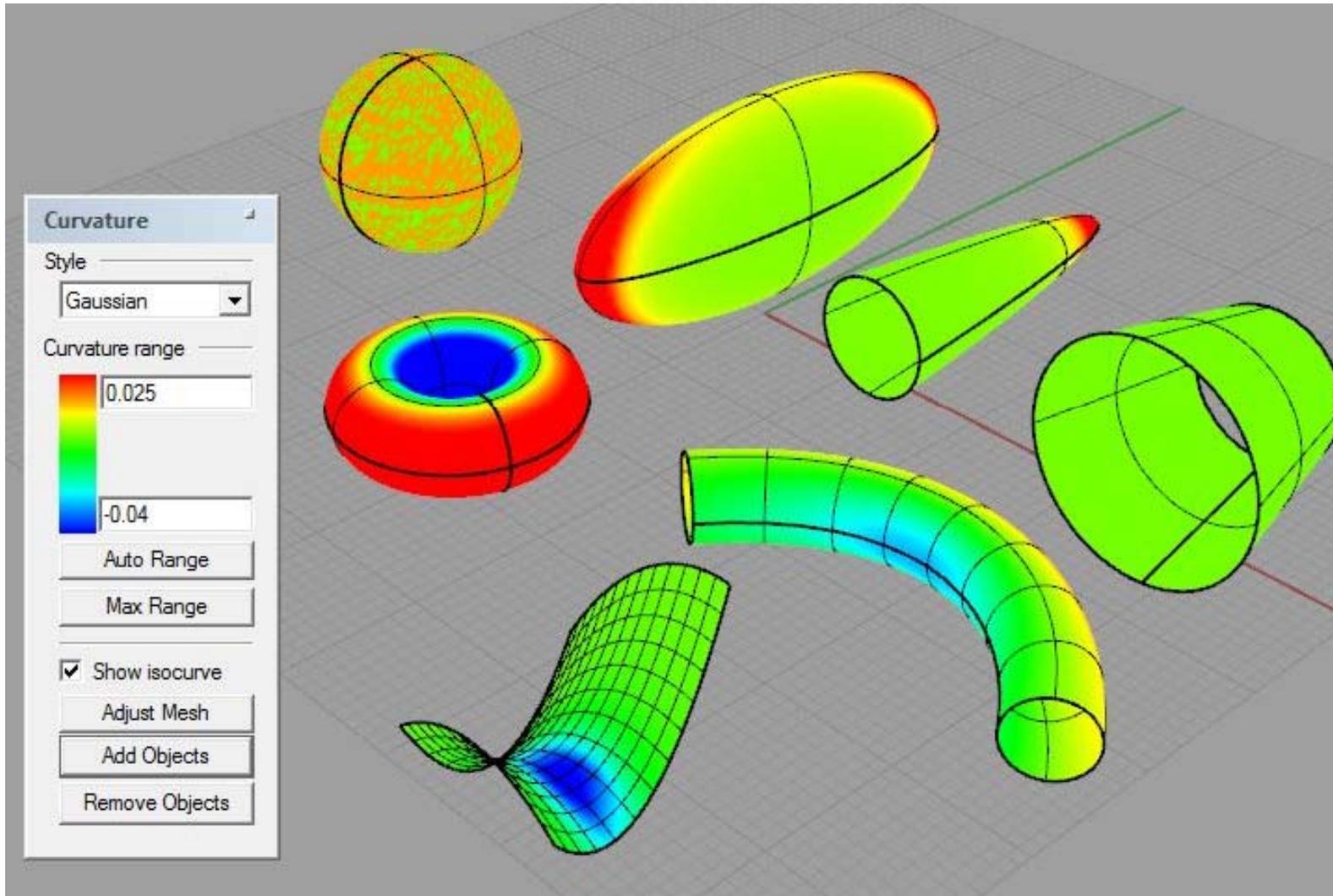
Platonic solids and regular surfaces

default.solid toolbar



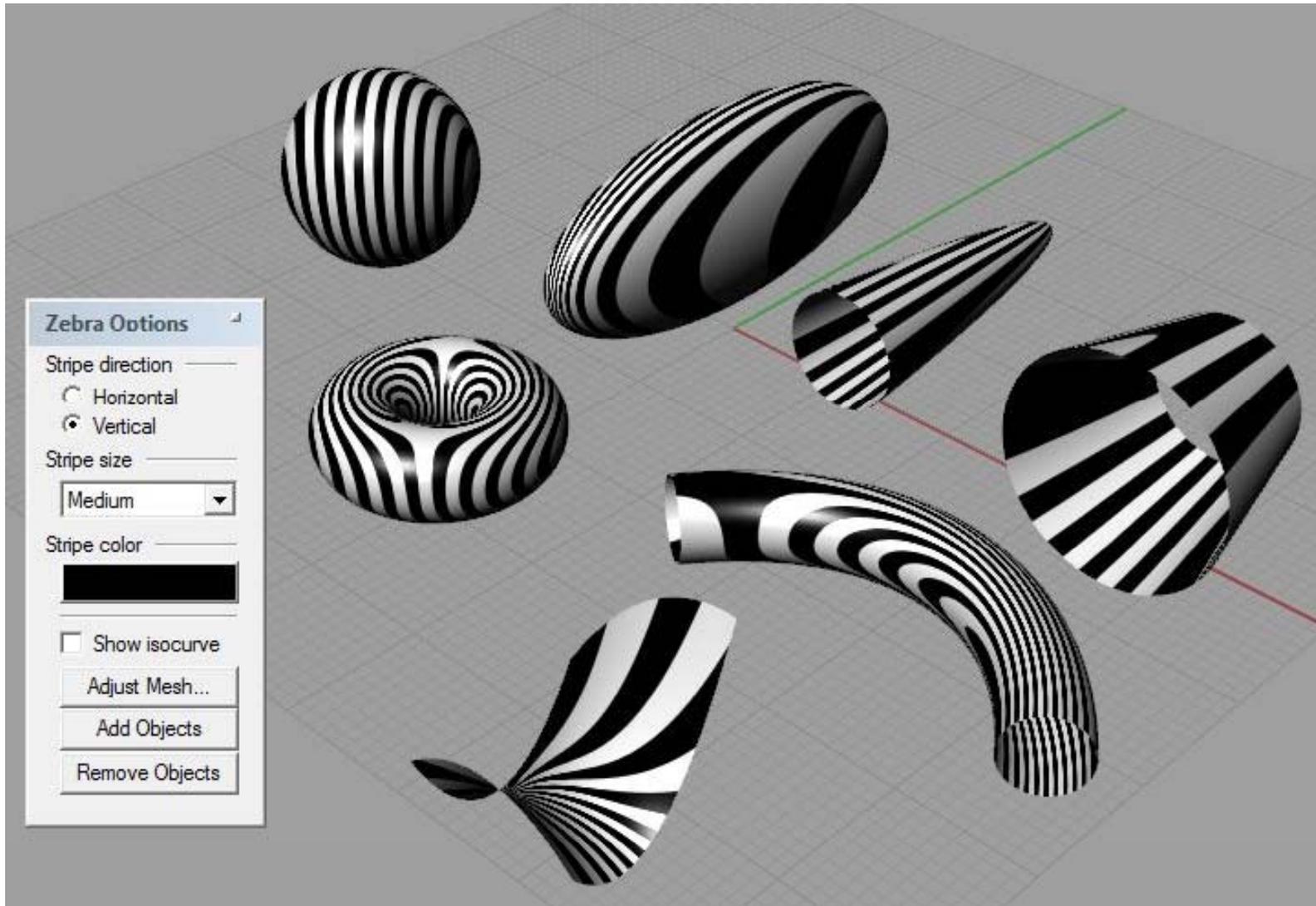
some of these commands create singular surfaces

SURFACES



curvature analysis of some singular NURBS surfaces

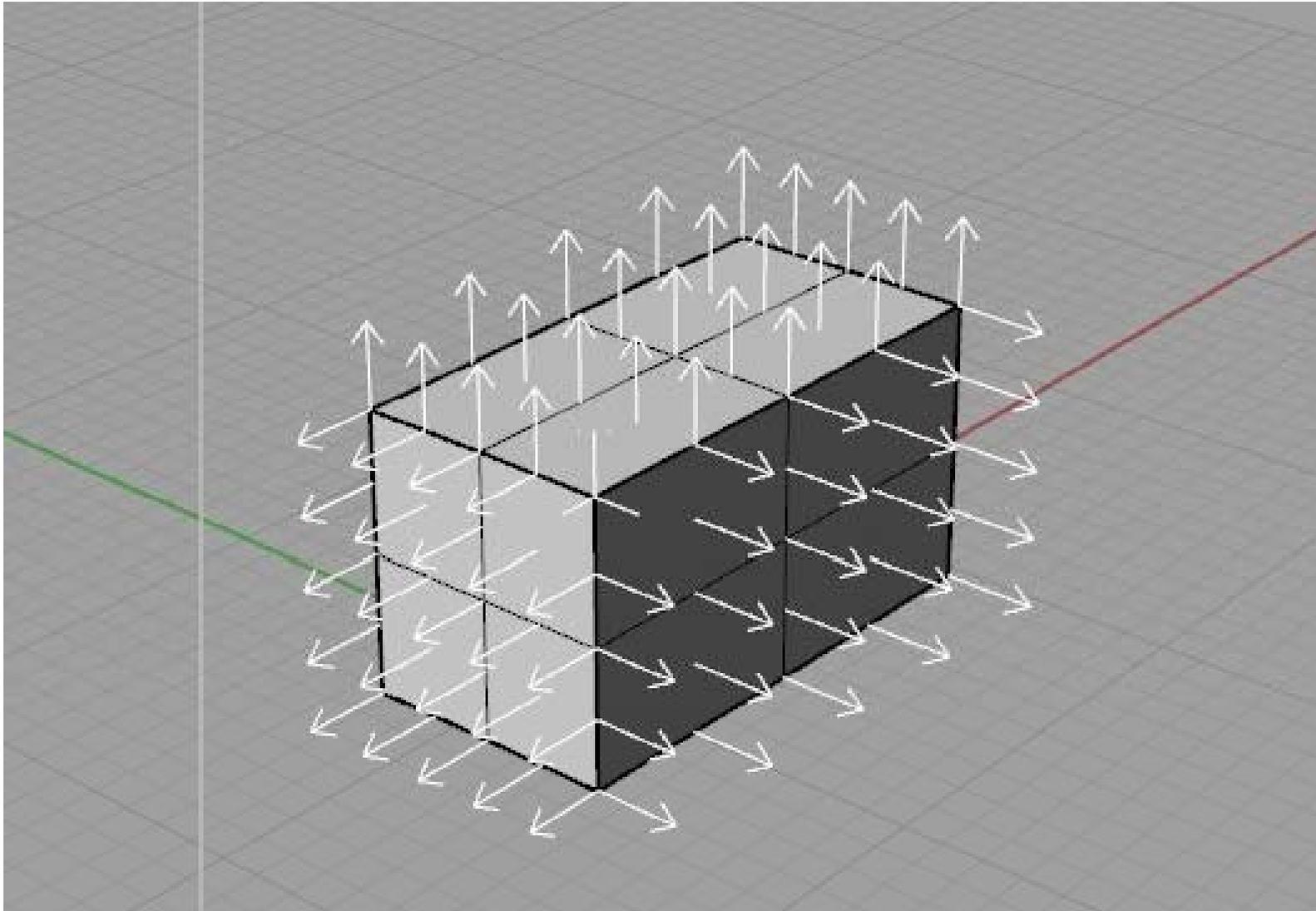
SURFACES



zebra (to visualize position, tangency and curvature continuity)

write help for more information about it

SURFACES



`dir` (to visualize and change direction of orientation), with `crvs` & `srfs` solids always have the normal toward the interior.
`flip` directly turns the direction of `crvs` and `srfs`.

SURFACES

10 basic methods of surface creation in Rhino

SrfPt
corner
points

**ExtrudeCr
v**
one curve
and (one

Revolve
profile
(crv)
axis (2

Sweep1
rail (crv)
profile
(crv)

Loft
sections
(curves)

EdgeSrf
2,3,4
edge
curves

BlendSrf
two
surfaces

RailRevolve
rail (curve
profile) (crv
axis)

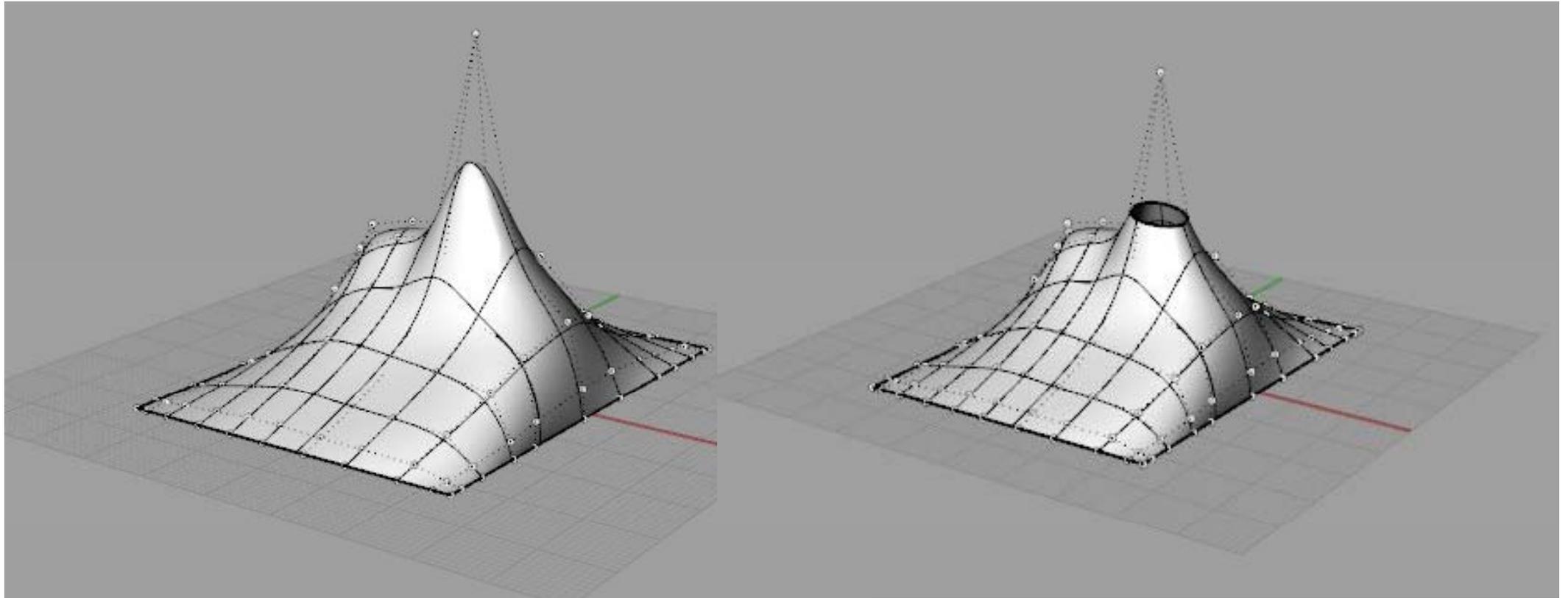
Sweep2
rail 1 (crv)
rail 2 (crv)
profile (crv)

NetworkSrf
4 or more
crvs

* More options of input are available

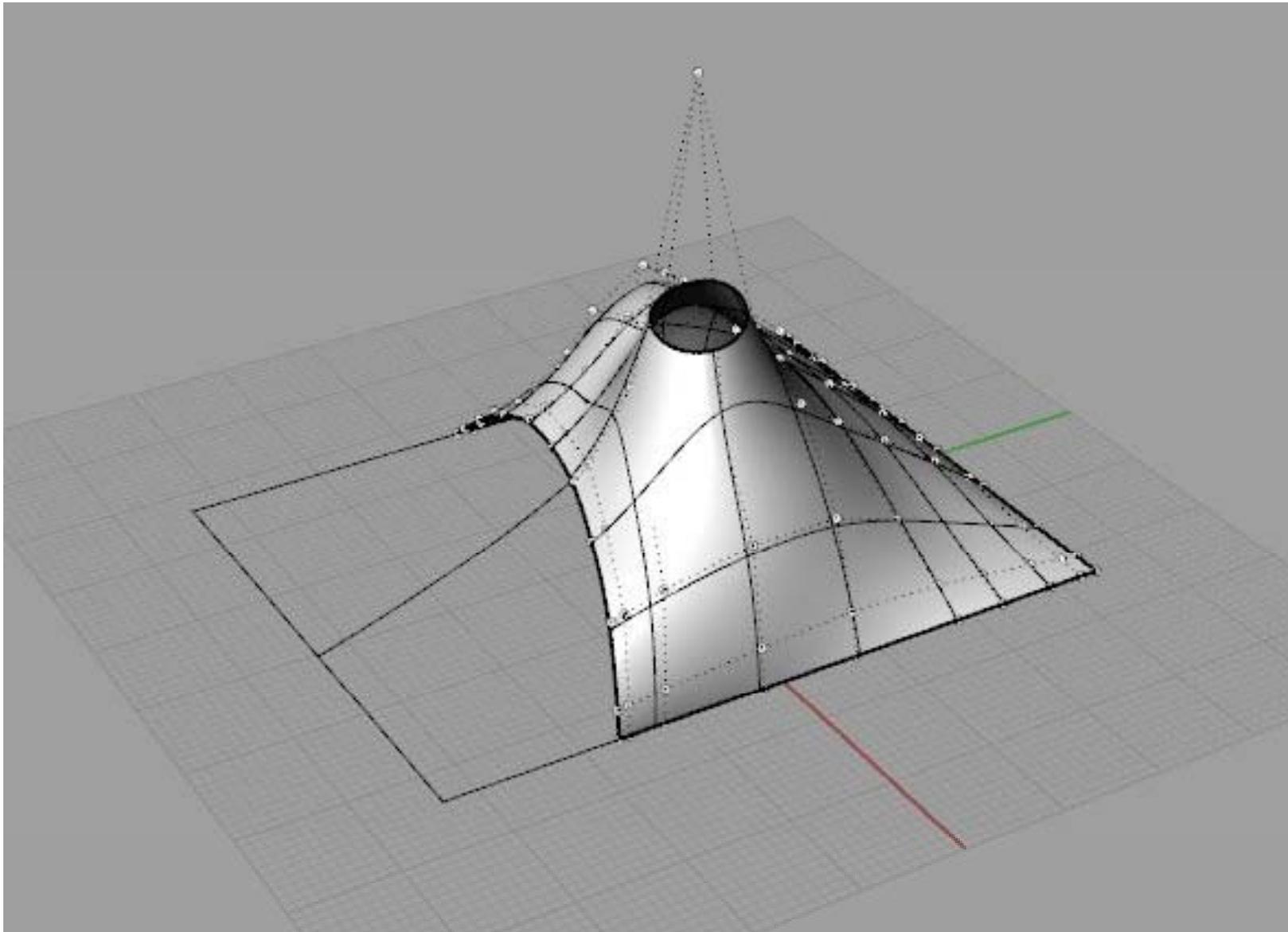


SURFACES



trim will cut the surface. But the control points remain.

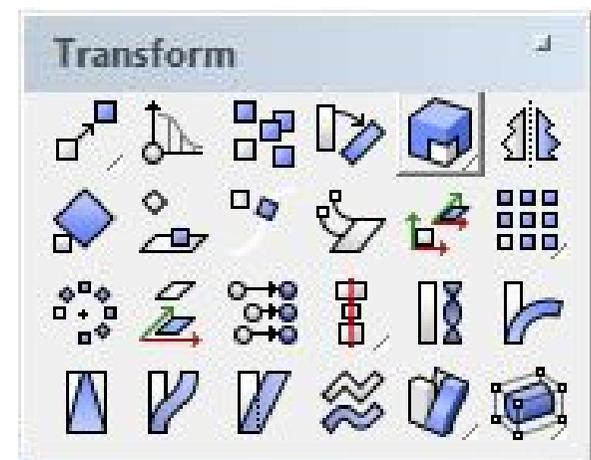
SURFACES



in some exceptional cases, we can use the command `shrinkTrimmedSrf`.
splitting along an isocurve will also automatically shrink the surface.

TRANSFORM TOOLS

If you are familiar with any modeling software, you'll find out that you can guess what each of these buttons will do.



VARIOUS OTHERS

in the next exercises you might find also the following other commands useful:

- **divide** a curve
- **offset** a curve

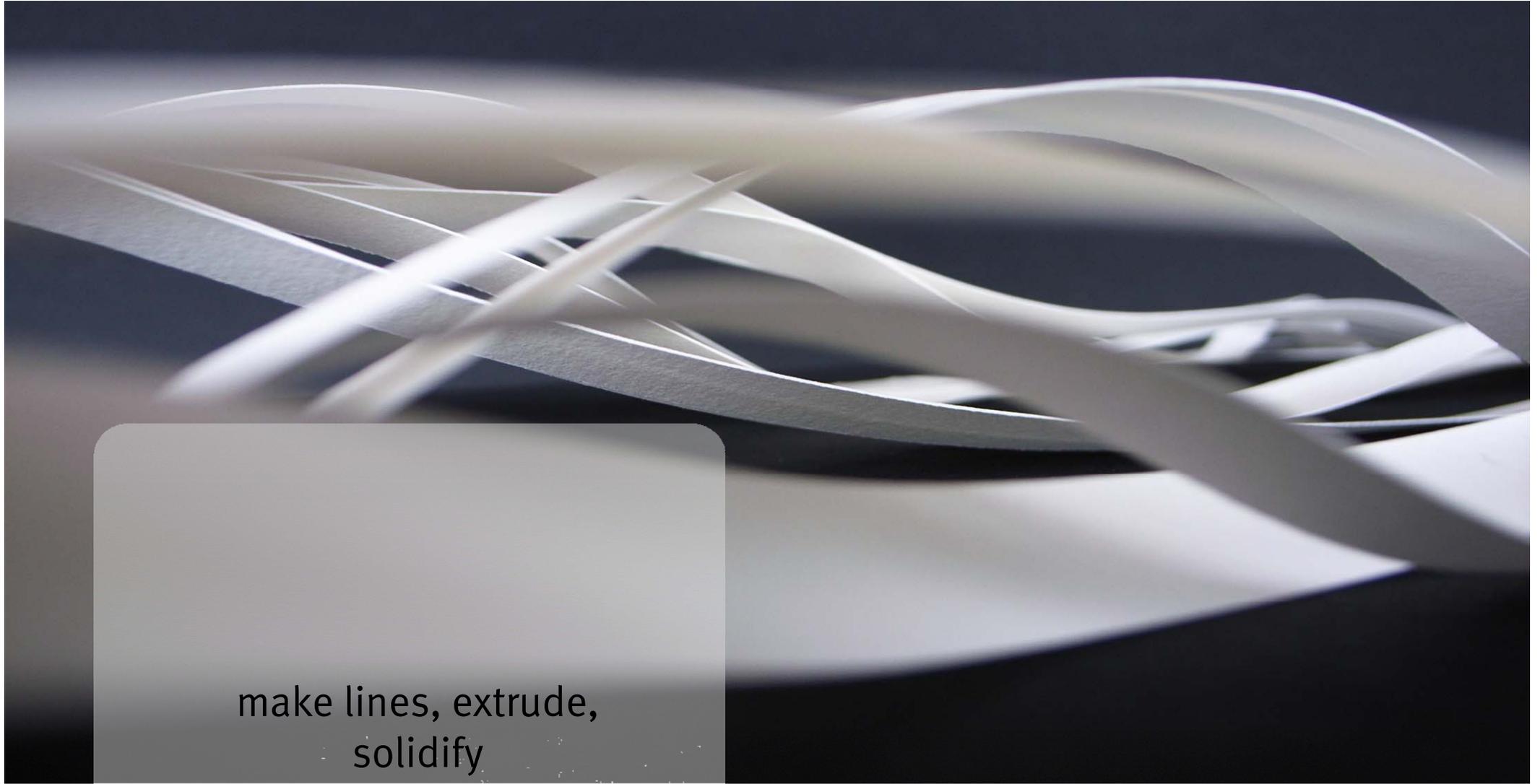
- **project** (parallel) onto a surface
- **pull** a curve onto a surface

- **extrudeSrf** to solidify a surface (straight movement)
- **offsetSrf** to solidify a surface (parallel movement)

A MODEL

An abstract composition of several white, glossy ribbons or strips of paper that are intertwined and looped against a solid black background. The ribbons create a complex, organic pattern of curves and crossings. In the lower right quadrant, there is a semi-transparent grey rectangular box containing the text "which rules can you use?".

which rules can you use?



make lines, extrude,
solidify

FIND YOUR WAY

Your
paper
model

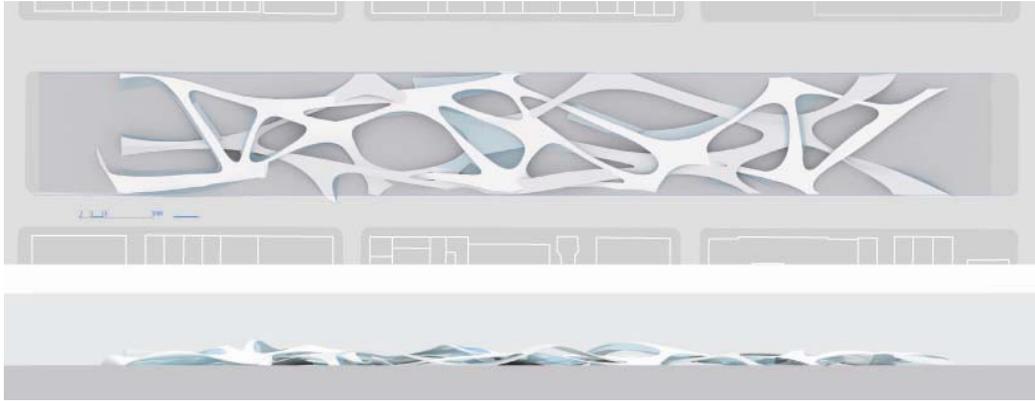
bent, twisted, cut

Your
3D
model

bend!, twist!, trim!

PROJECTS WITH RHINO

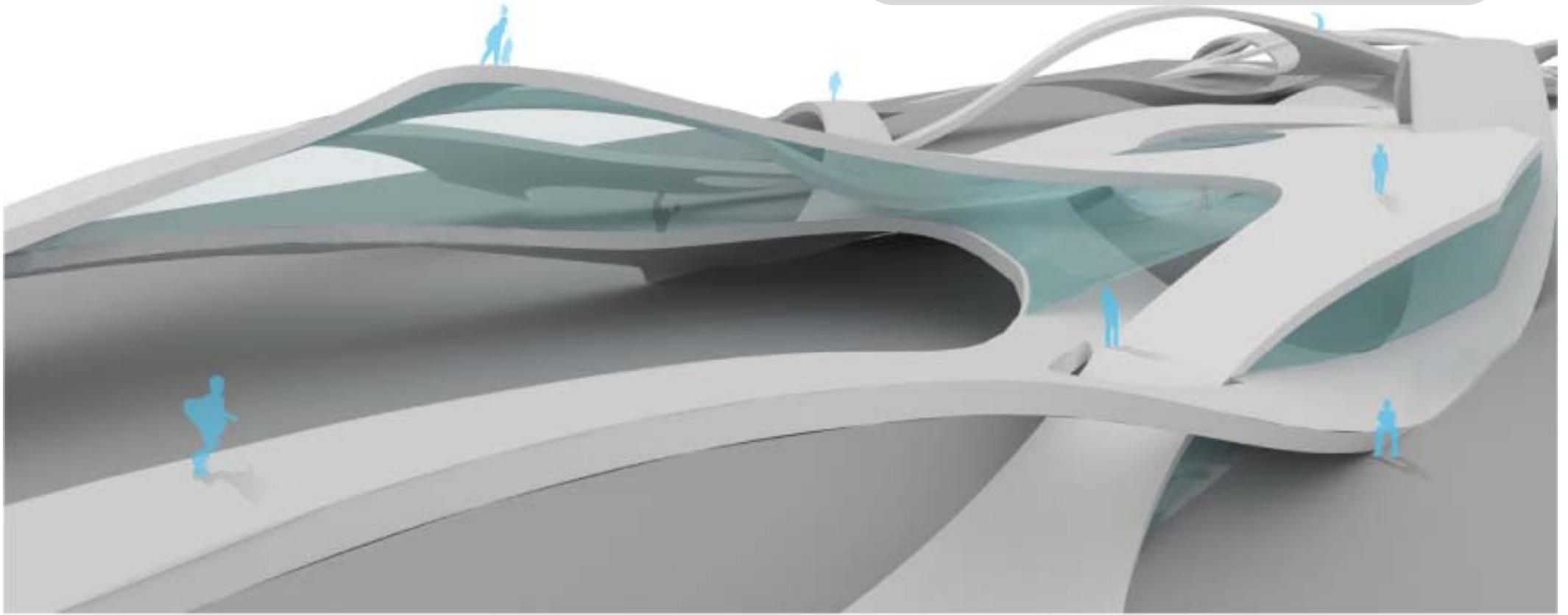
PROJECTS WITH RHINO



RYOKO IKEDA

Social facilitation building in New York

Graduation Project Border Condition 2008



Copyright Giulio Piacentino 2008

*All rights reserved. The author may grant a different
license in writing. mail@giuliopiacentino.com.*