Rendering/Parametric Modelling 07
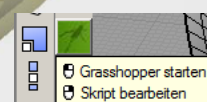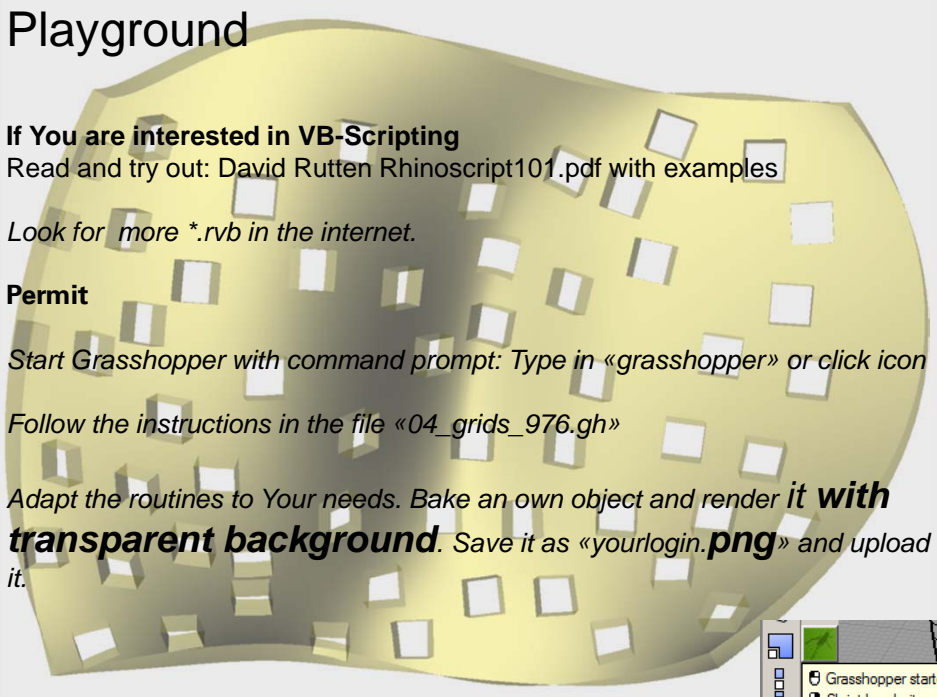
08:30 Rendering - History - Macros – Rhino Visual Basic

09:15 Grasshopper / Rendering

10:30 Playground

12:00 End

Playground

If You are interested in VB-Scripting
Read and try out: David Rutten Rhinoscript101.pdf with examples

Look for more *.rvb in the internet.

Permit

Start Grasshopper with command prompt: Type in «grasshopper» or click icon

Follow the instructions in the file «04_grids_976.gh»

Adapt the routines to Your needs. Bake an own object and render **it with transparent background**. *Save it as «yourlogin.***png***» and upload it.*

Rhino Expansions – Erweiterungen nur mit Windows
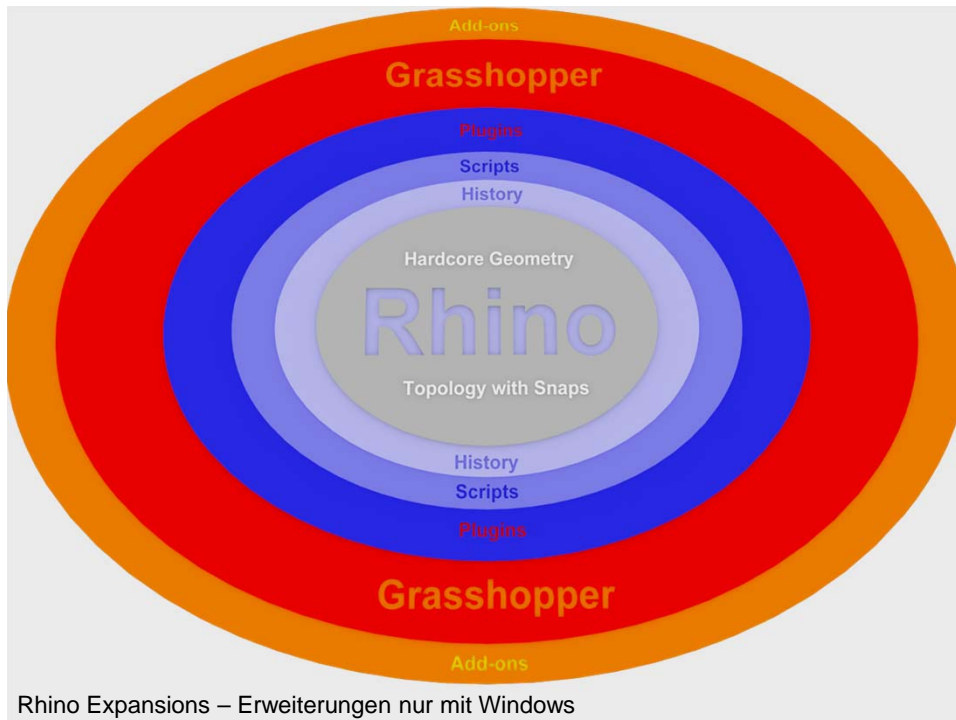
Macros: linear sequences of commands
You will find a list of commands in the Help

Syntax

! (Exclamation Mark) Deleting the previous command
_ (Underline) English command name
- (Hyphen) Supresses Options Window
' (Apostrophe) Next command is nested
; (Semicolon) Comment

Tools – Read Command File: *.txt
Or set a Shortcut in the Options - Keyboard - F4

Macros
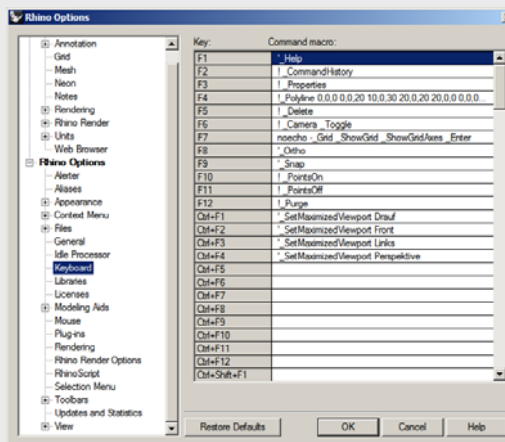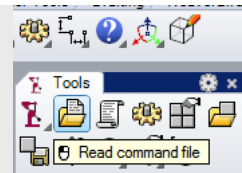
«Parenting» is a one-way relationship:
The child depends of the parents, the parents not depend of the children.

Activate „**Record History**" while using the (loft) command.
You now can modify the curves - the surface will follow.
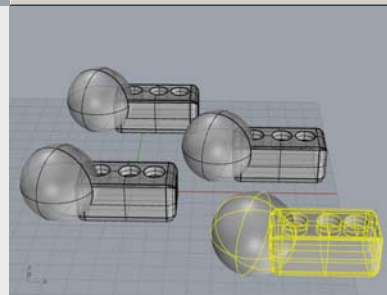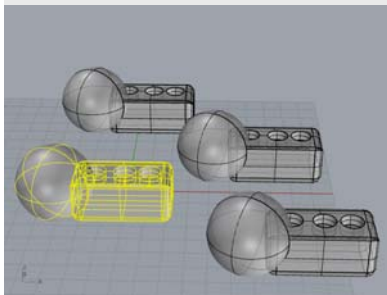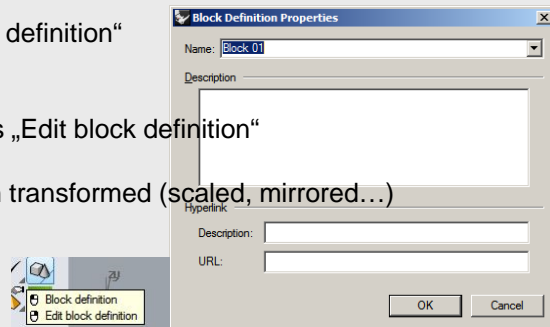When You move the surface, the relation will be broken.

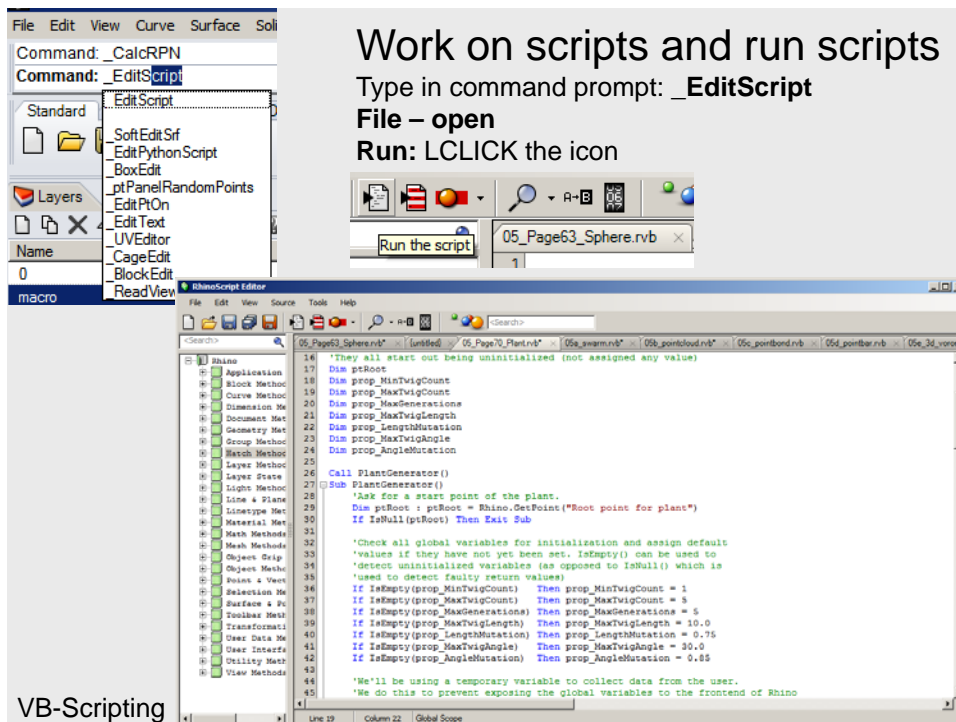Consult F1(Help) „History" to look which commands are „history – enabled"

History

«Blocks» is a two-way relationship:

Select an object and press „Block definition"
Give a name
Copy the object several times.
Select one of the objects an press „Edit block definition"
Edit it and confirm:
All blocks are adapted even when transformed (scaled, mirrored…)

Blocks

Work on scripts and run scripts
Type in command prompt: _**EditScript**
File – open
Run: LCLICK the icon

VB-Scripting

RhinoScript

This language is situated between macros and compiled (translated in computer language) Programs

Rhinoscript is based on Microsoft Visual Basic Script (VBScript):
BASIC-Family in 3. Generation
(**B**eginners **A**ll purpose **S**ymbolic **I**nstruction **C**ode, 1963)
And ist a
Object-oriented language (Opposite: Batch system, like macros)

The task of syntax is:

Handling variable data
Flow control
Input – output control
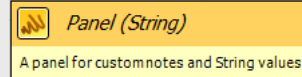
VB-Scripting: additional knowledge

Variables

The common variables are:

integer - a real number
double - a decimal place number
booleans - either true or false
strings - a set of characters

Panel (String)
A panel for custom notes and String values

Arrays are lists of variables.
They are retrieved with a zero-based counting system: the Index.
Dim arrExample(3)
arrExample(0) = 9
arrExample(1) = 11
arrExample(2) = 3
arrExample(3) = 7


```
                    {0}
0 {0.0, 0.0, 0.0}
1 {0.0, 1.0, 0.0}
2 {0.0, 2.0, 0.0}
3 {0.0, 3.0, 0.0}
4 {0.0, 4.0, 0.0}
5 {0.0, 5.0, 0.0}
6 {1.0, 0.0, 0.0}
7 {1.0, 1.0, 0.0}
8 {1.0, 2.0, 0.0}
9 {1.0, 3.0, 0.0}
```
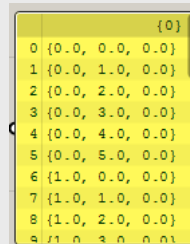
VB-Scripting : additional knowledge

Grasshopper

Readers
Zubin Khabazi: Generative Algorhithms.pdf
Mode Lab Grasshopper Primer Third Edition.pdf

Links
http://www.rhino3.de/
http://blog.rhino3d.com/
http://wiki.mcneel.com/rhino/home
http://wiki.mcneel.com/people/scottdavidson
http://www.grasshopper3d.com/
http://rhinotoday.com/
http://blog.rhino3d.com/2010/01/digital-toolbox-new-tutorial-site-for.html
http://eat-a-bug.blogspot.com/
http://www.generativedesigncomputing.net/
http://www.liftarchitects.com/
http://www.giuliopiacentino.com/grasshopper-tools/
http://www.designalyze.com/
http://www.food4rhino.com/
http://processing.org/