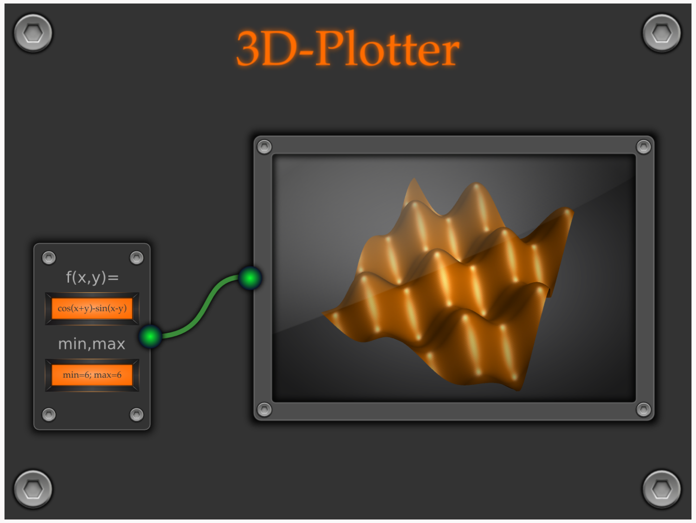
Creating Amazing Scientific Visualization Tools with JavaFX 8



f(x,y)= cos(x+y)-sin(x-y)

min,max min=6; max=6

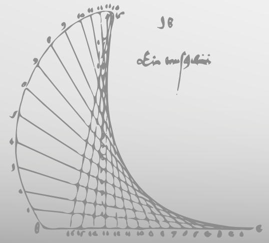
JavaOne 2013

(Sept. 22-26, 2013)

Michael Hoffer

G-CSC

Goethe University Frankfurt

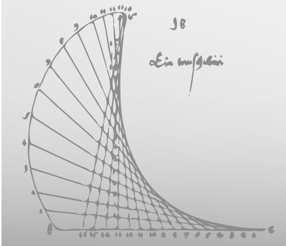


About Me

- **Doing my PhD at the G-CSC, University of Frankfurt**
- **Research interests:**
  developing Visual Programming Concepts
- **Software Projects: VRL-Studio, VWorkflows, JFXtras**

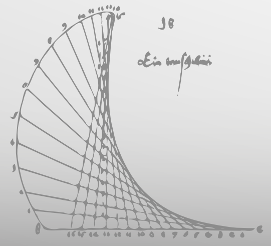
Twitter: @mihosoft

Web: mihosoft.eu



Outline

- Why Choose JavaFX?
- Introduction to Functions
- Creating a 2D & 3D Function Plotter
- Combining 2D & 3D Visualizations
- Loading and Visualizing 3D Geometries
- But it does move!
- Visualizing Simulation Workflows

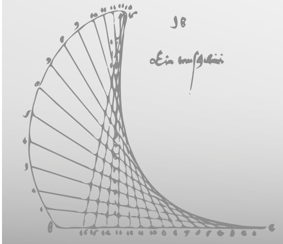


Why Choose JavaFX?

Explain current situation:

Swing+Java3D/OpenGL
(not easy to integrate!)
or
SWT, Qt...

It's a zoo of different options...

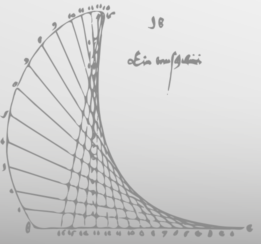


Why Choose JavaFX?

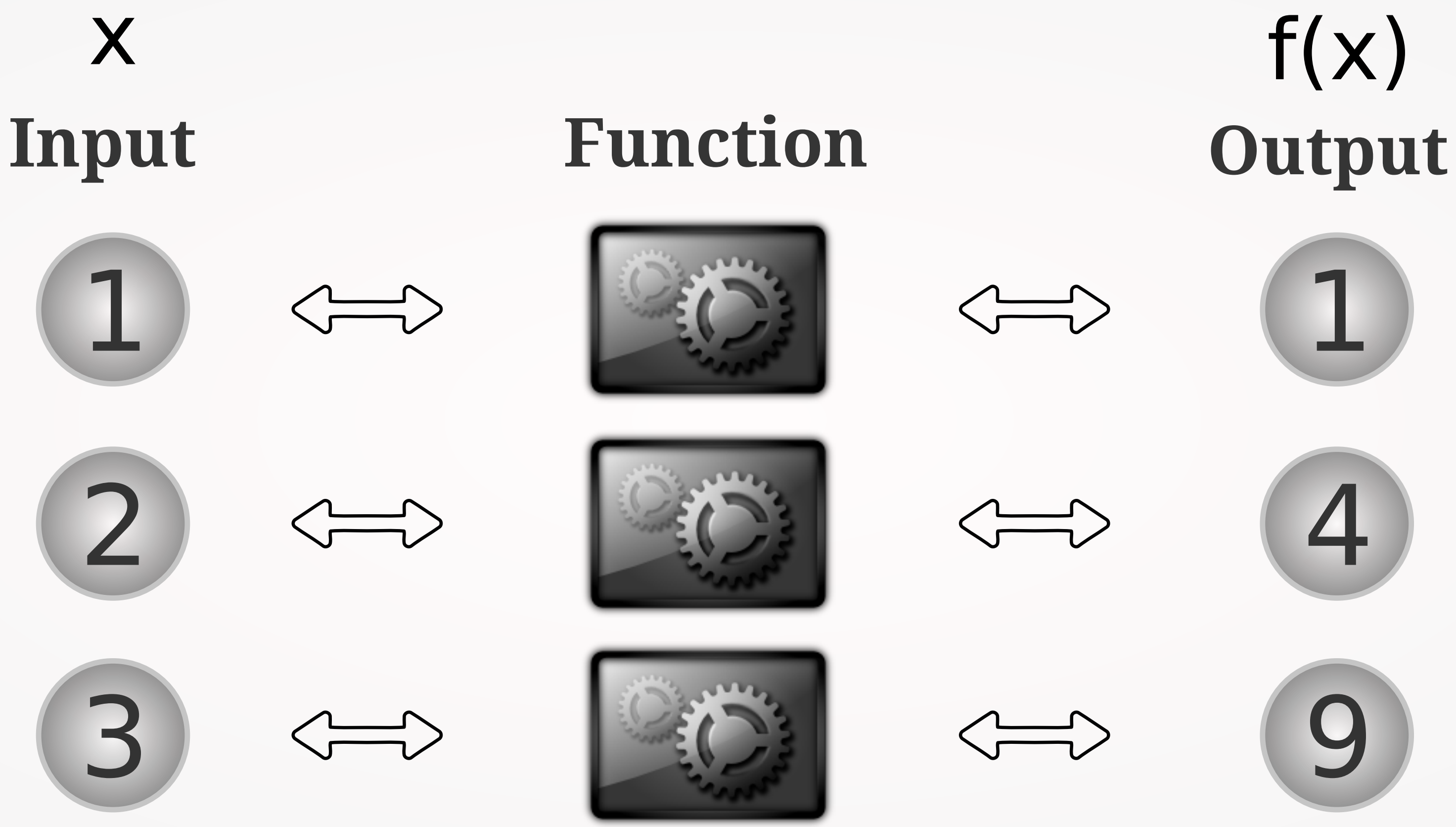
One To Rule Them All:

JavaFX:

2D API (Controls, Charting API)
and
3D API (Primitives, Meshes, Light,...)

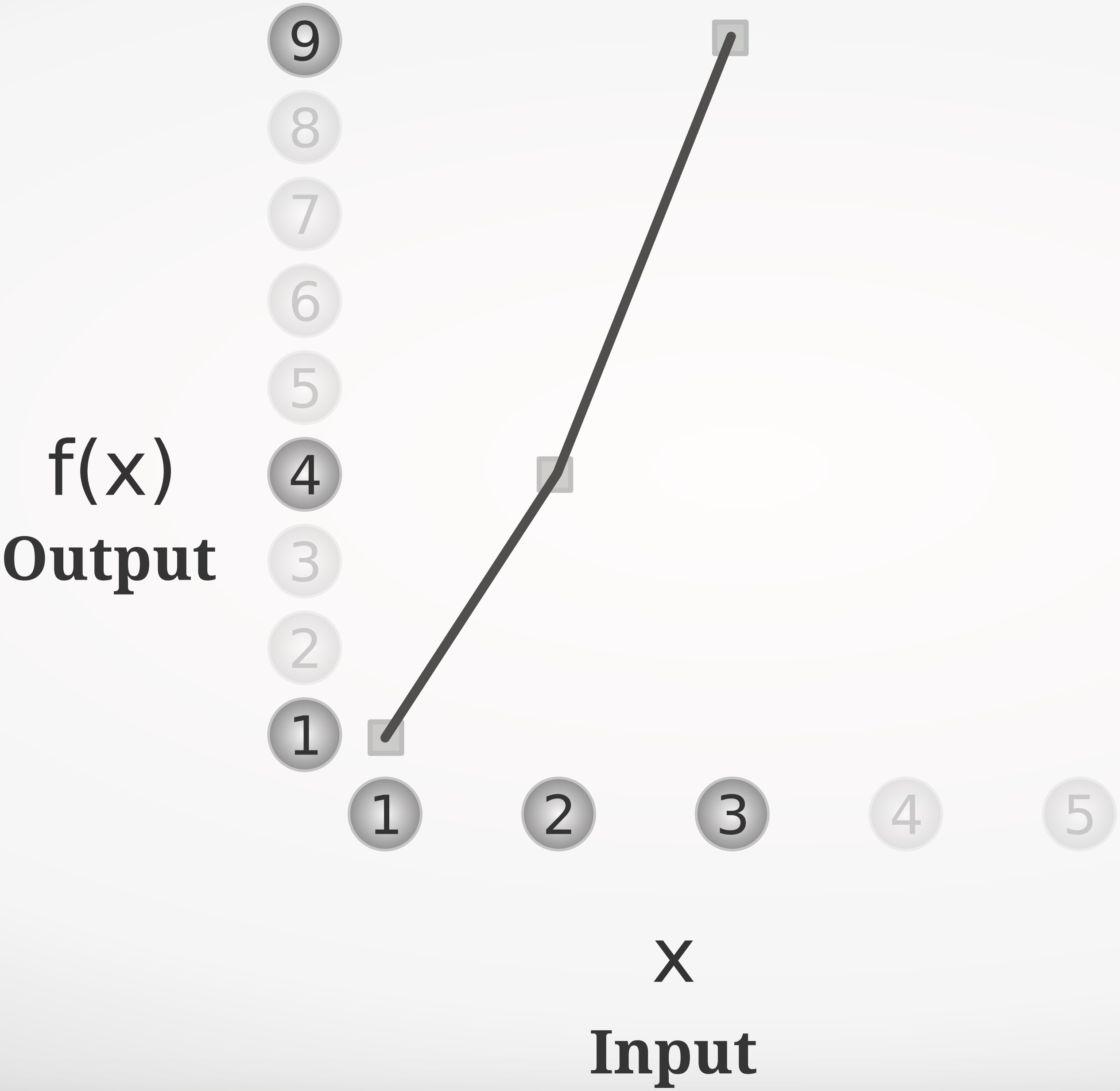


Introduction To Functions



x Input		Function		f(x) Output
1	⟷		⟷	1
2	⟷		⟷	4
3	⟷		⟷	9

Introduction To Functions



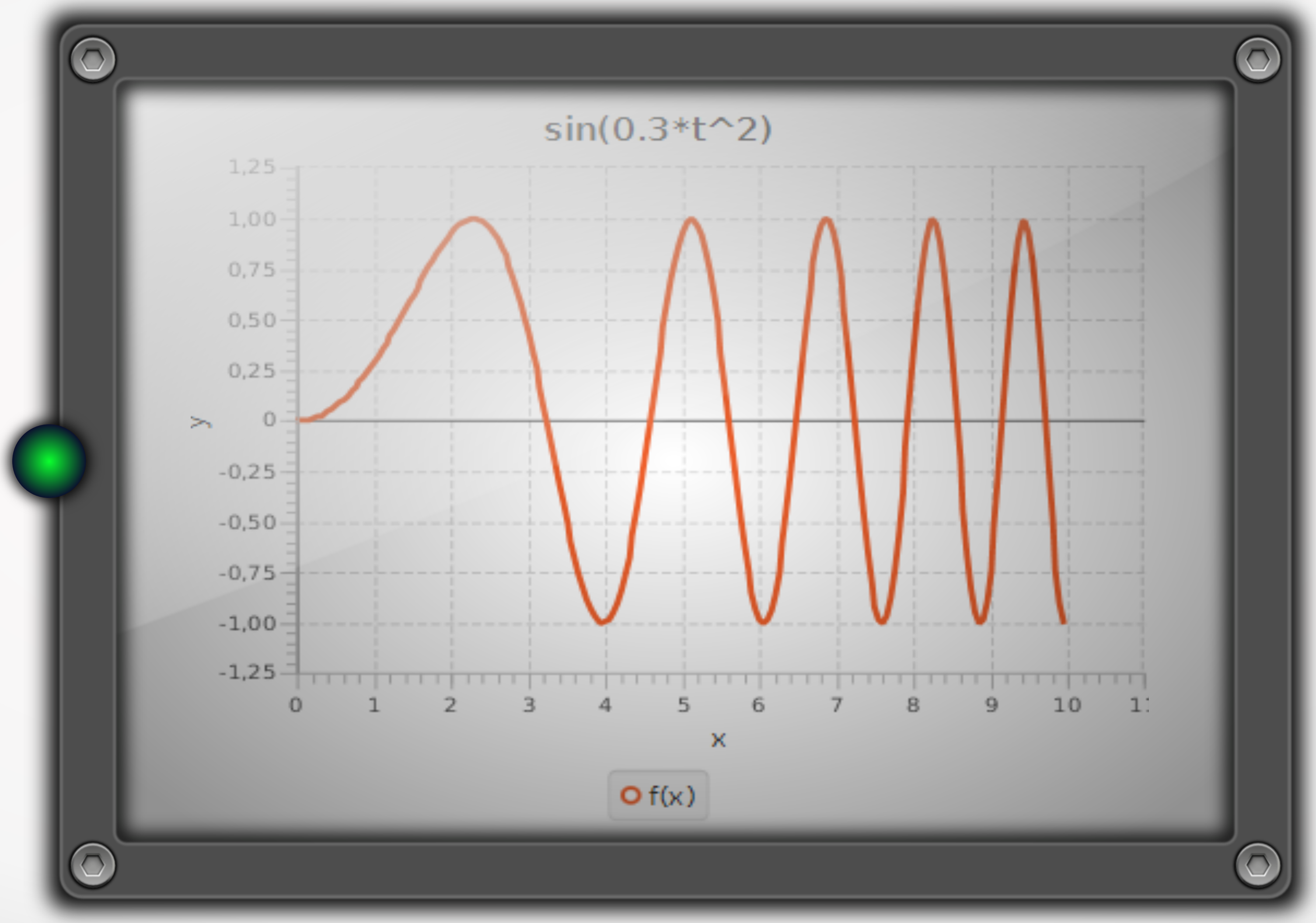
f(x)

Output

x

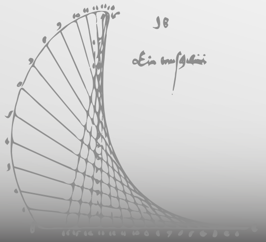
Input

Creating a 2D Function Plotter



Creating a 2D Function Plotter

Introduction To GroovyShell for evaluating Expressions



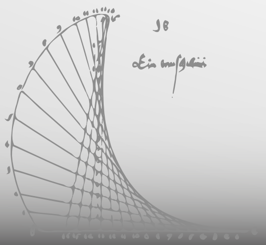
Creating a 2D Function Plotter

Explaining LineChart API
XYCharts.series()
Axes, Ticks, CSS etc.

Creating a 2D Function Plotter

Converting evaluated values to XYChart.Series()

Show the LineChart as child of a ScalableContentPane (from JFxtras)

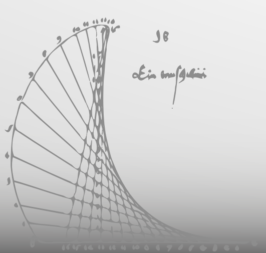


Creating a 2D Function Plotter

How to make the plot interactive?

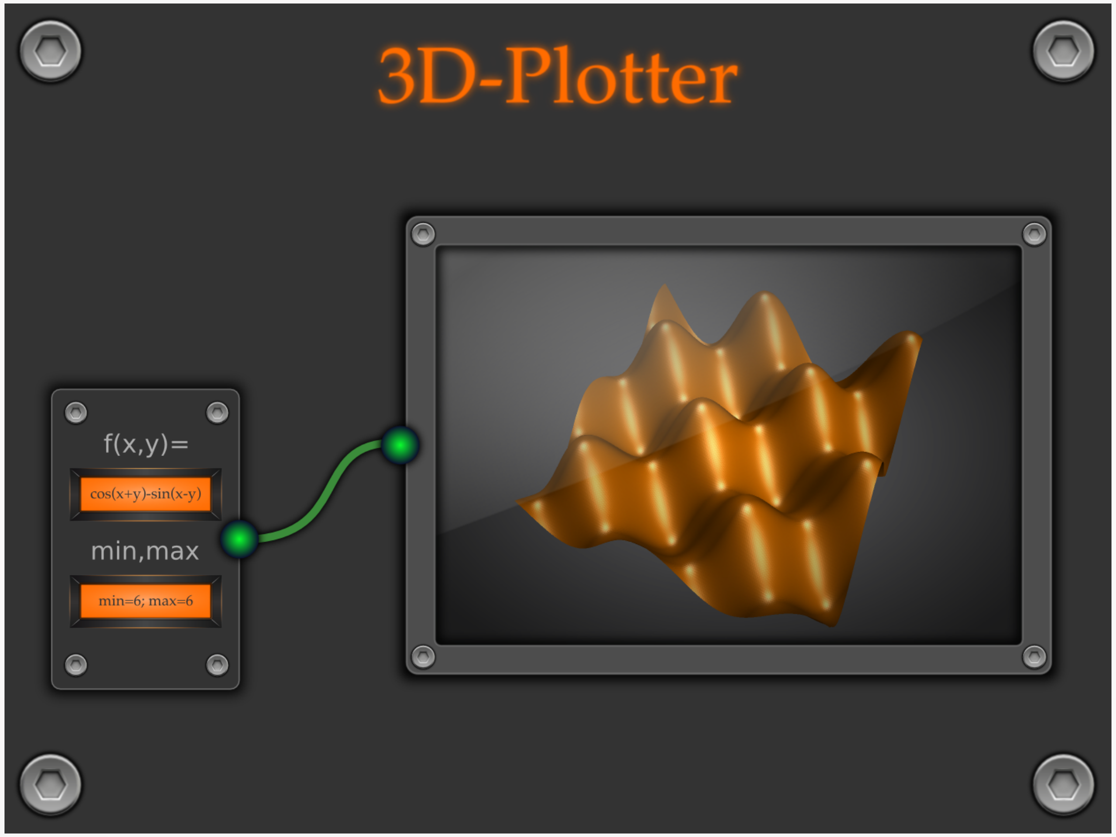
- add parameter sliders via binding (evaluates function and updates plot)

- click on plot/line to see exact values



Code / Demo

Creating a 3D Function Plotter

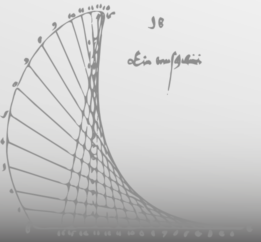


f(x,y)= cos(x+y)-sin(x-y)

min,max min=6; max=6

Creating a 3D Function Plotter

Revisiting Function Evaluation (now with 2 Parameters)

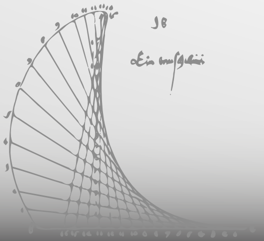


Creating a 3D Function Plotter

Explaining concept of representing geometries:

Meshes, TriangleMesh, …

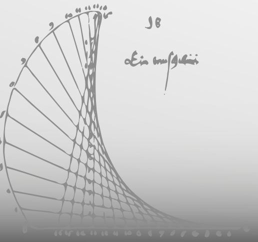
…



Creating a 3D Function Plotter

Short Introduction to
scenes, lights & materials

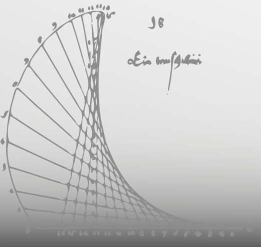
showing demo code that creates a
basic scene with lights, persp. cam etc.



Creating a 3D Function Plotter

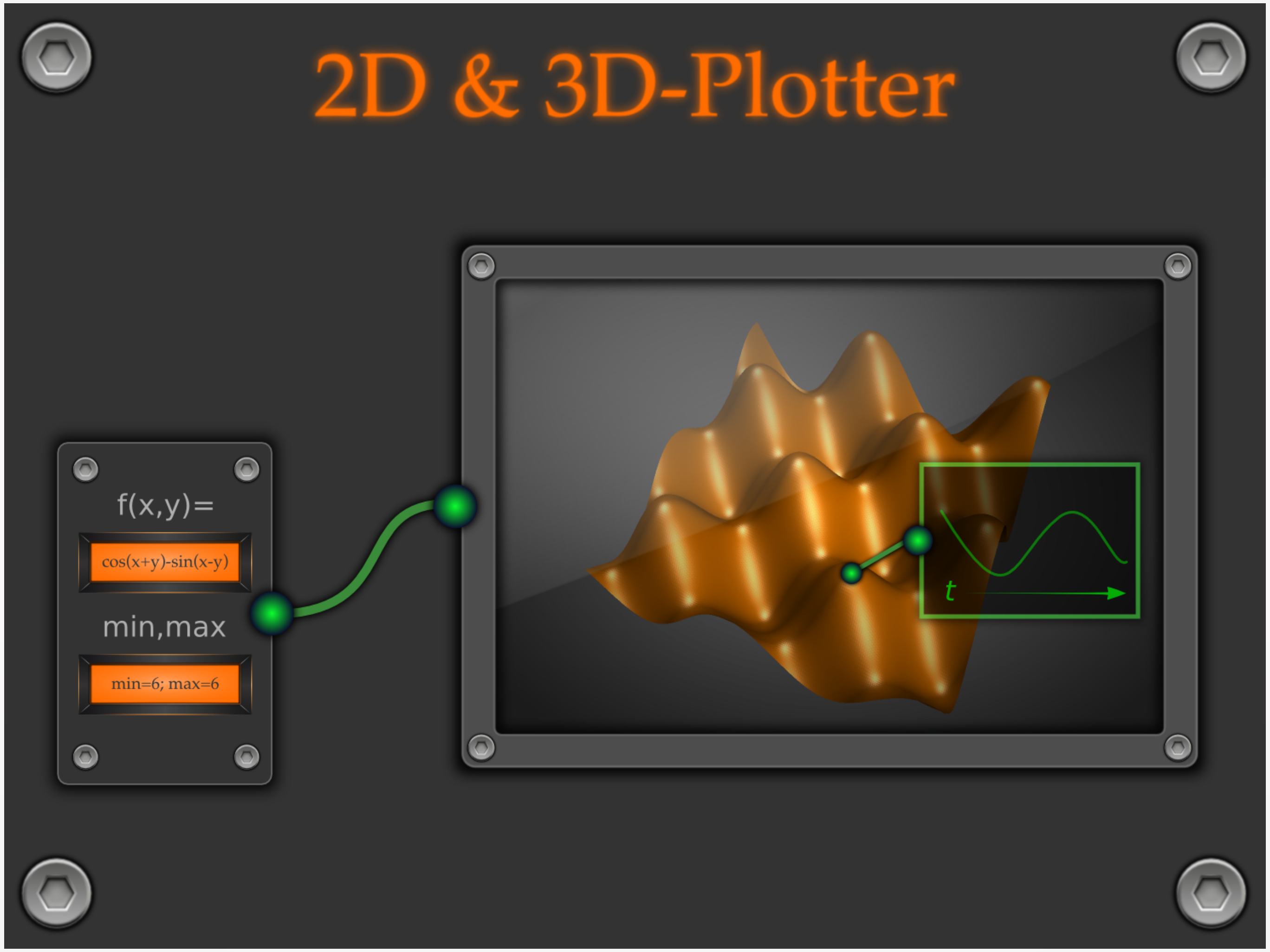
Adding the geometry that has been derived from the 3D function evaluator to the scene

Demo codes uses Window control from JFXtras/VWorkflows to show input & output (3D scene)



Code / Demo

Combining 2D & 3D



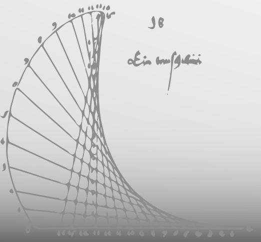
f(x,y)= cos(x+y)-sin(x-y)

min,max min=6; max=6

Combining 2D & 3D

Emphasize that 3D nodes provide 2D node API!

Use ray picking to select points in 3D geometry and plot value change over time with previously developed 2D plotter!



Combining 2D & 3D

Short Discussion on other possible use cases:

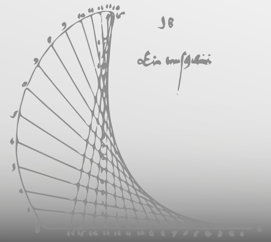
selecting parts for defining simulation parameters…

Code / Demo

Loading & Saving Geometries

Starting with simple .txt format:

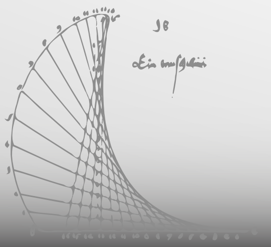
```
#nodes # triangles
node index node_x node_y node_z
.
.
.
triangle_index node_index_1 node_index_2 node_index_3
.
.
.
```



Loading & Saving Geometries

Explaining subset of .obj format

example code can load simple models
from .txt and .obj
and save them as well
(triangles)



Code / Demo

But It Moves

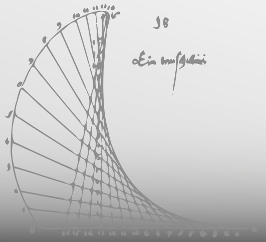
Introduction to snapshot functionality

Creating images from 2D and 3D scenes

Changing function prams over time!

Simulation Workflow

Using prepared api
(part of sample code)
to create .mov file (uncompressed)
by adding writable images as
frames

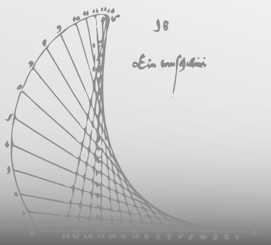


Simulation Workflow

Demonstrate full simulation example
numerics code consists of
ODE & PDE solver

examples are fully prepared
(no in-depth code discussion due to time restrictions)
they include all visualizations that have
been previously developed

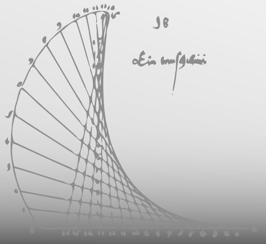
(no fear, no introduction to numerics)



Simulation Workflow

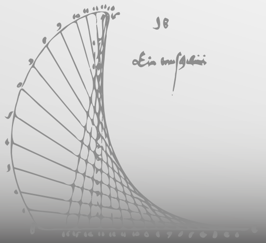
All numerics & visualization done with just one platform
Java 8

again, one to rule them all!



Code / Demo

Thank you for your attention!



Q & A