King Abdullah University of Science and Technology KAUST

CS 247 – Scientific Visualization
Lecture 3: The Visualization Pipeline; Data Representation, Pt. 1

Markus Hadwiger, KAUST

Reading Assignment #2 (until Feb 7)

Read (required):

- Data Visualization book, finish Chapter 2
- Data Visualization book, Chapter 3 until 3.5 (inclusive)
- Data Visualization book, Chapter 4 until 4.1 (inclusive)

- Continue familiarizing yourself with OpenGL if you do not know it !

Programming Assignments Schedule (tentative)

Assignment 0:	Lab sign-up: setup piazza + github account, get git repo	until	**Jan 31**
	Basic OpenGL example [we will offer a tutorial!]		
Assignment 1:	Volume slice viewer	until	**Feb 13**
Assignment 2:	Iso-contours (marching squares)	until	**Feb 27**
Assignment 3:	Iso-surface rendering (marching cubes)	until	**Mar 15**
Assignment 4:	Volume ray-casting, part 1	until	**Mar 31**
	Volume ray-casting, part 2	until	**Apr 7**
Assignment 5:	Flow vis, part 1 (hedgehog plots, streamlines, pathlines)	until	**Apr 21**
Assignment 6:	Flow vis, part 2 (LIC with color coding)	until	**May 5**

Programming Assignment #1: Slice Viewer

Basic tasks

- Download data into 3D volume texture
- Display three different axis-aligned slices using OpenGL texture mapping using the 3D volume texture

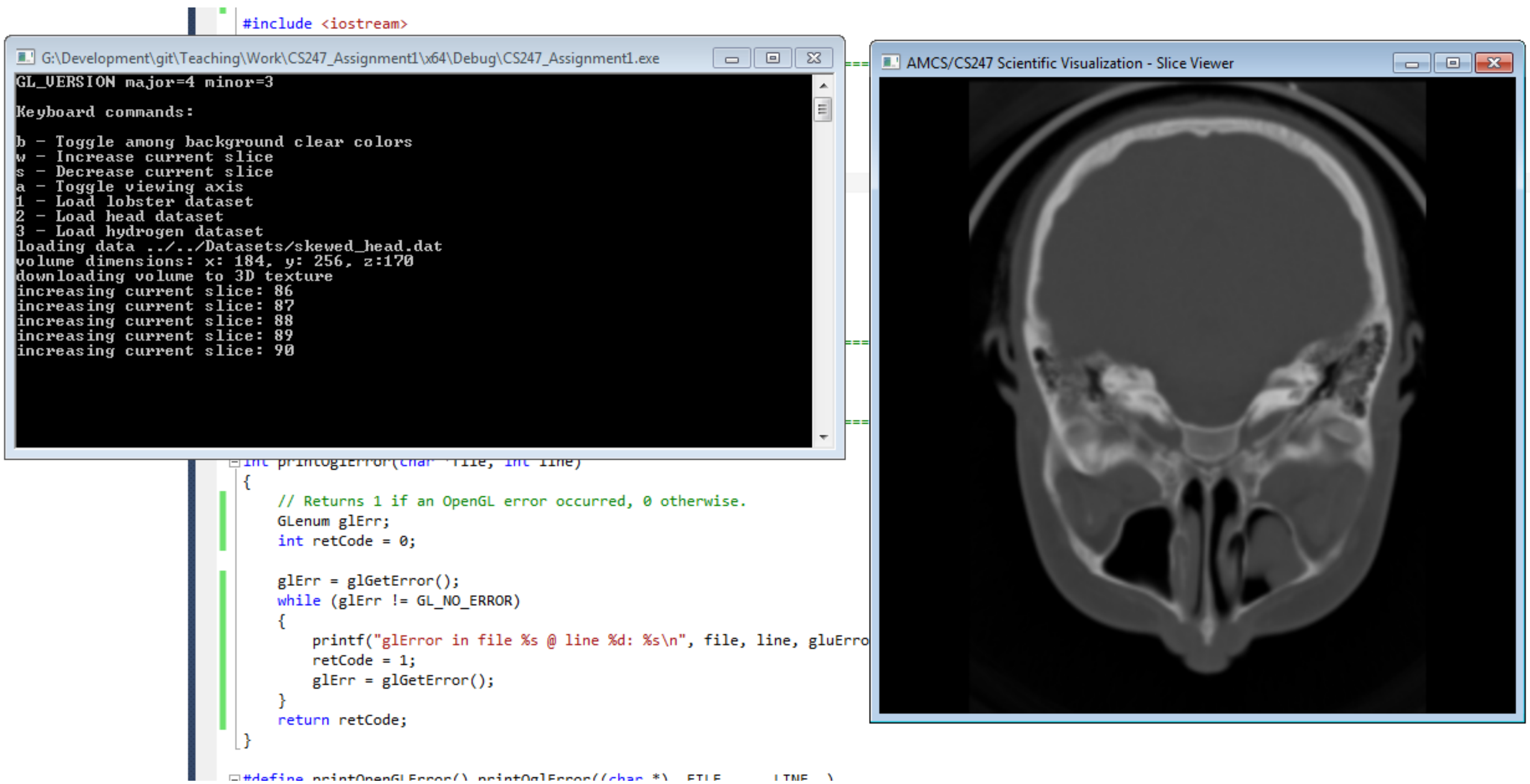
Minimum

- The slice position should be adjustable for each slice view.
- Make sure the aspect ratio of the shown slices is correct.
- If the window is resized, the slice is resized with the correct aspect ratio (no distortions)

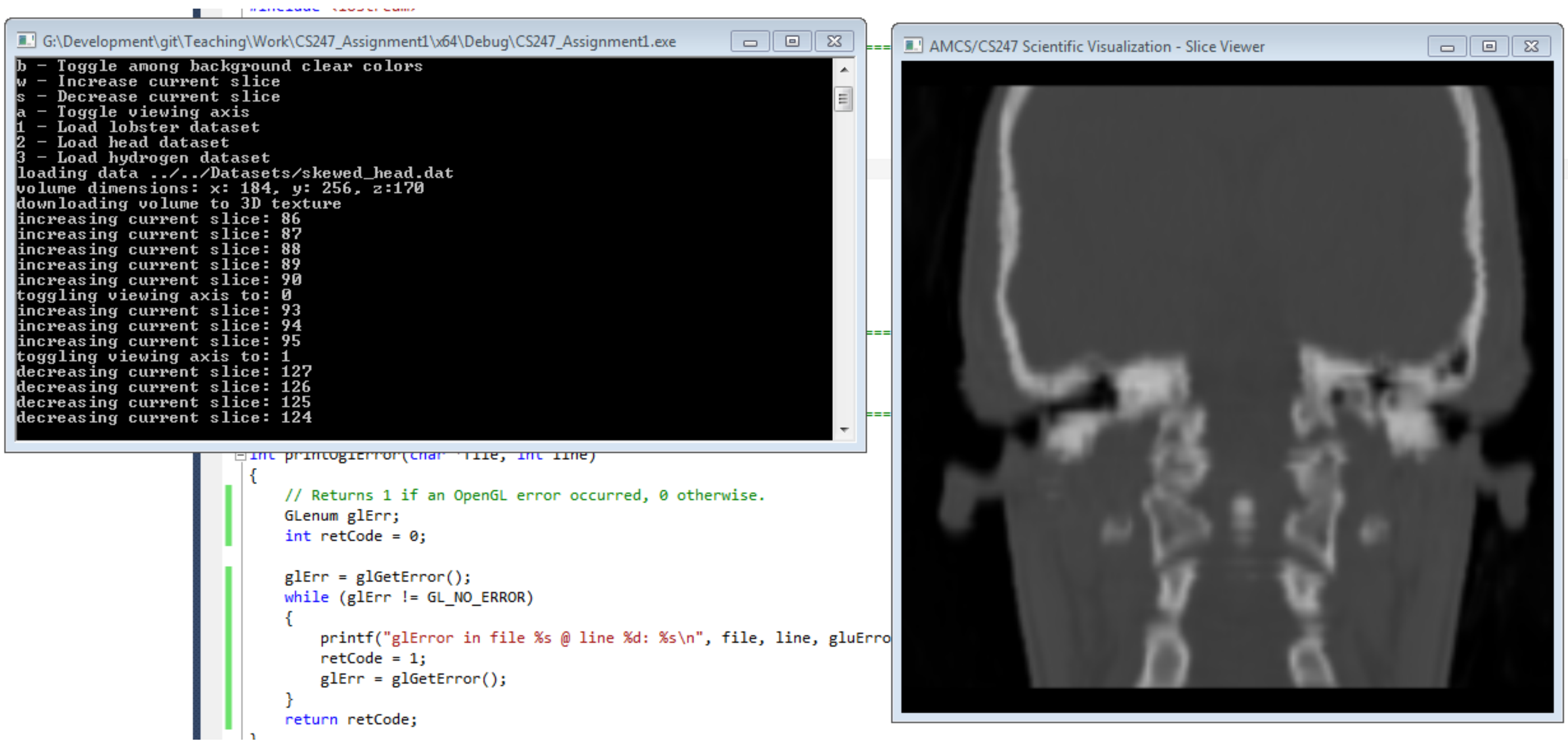
Bonus

- Show all three axis aligned slices at once
- Show arbitrarily aligned slices with an interface to change the arbitrary slice

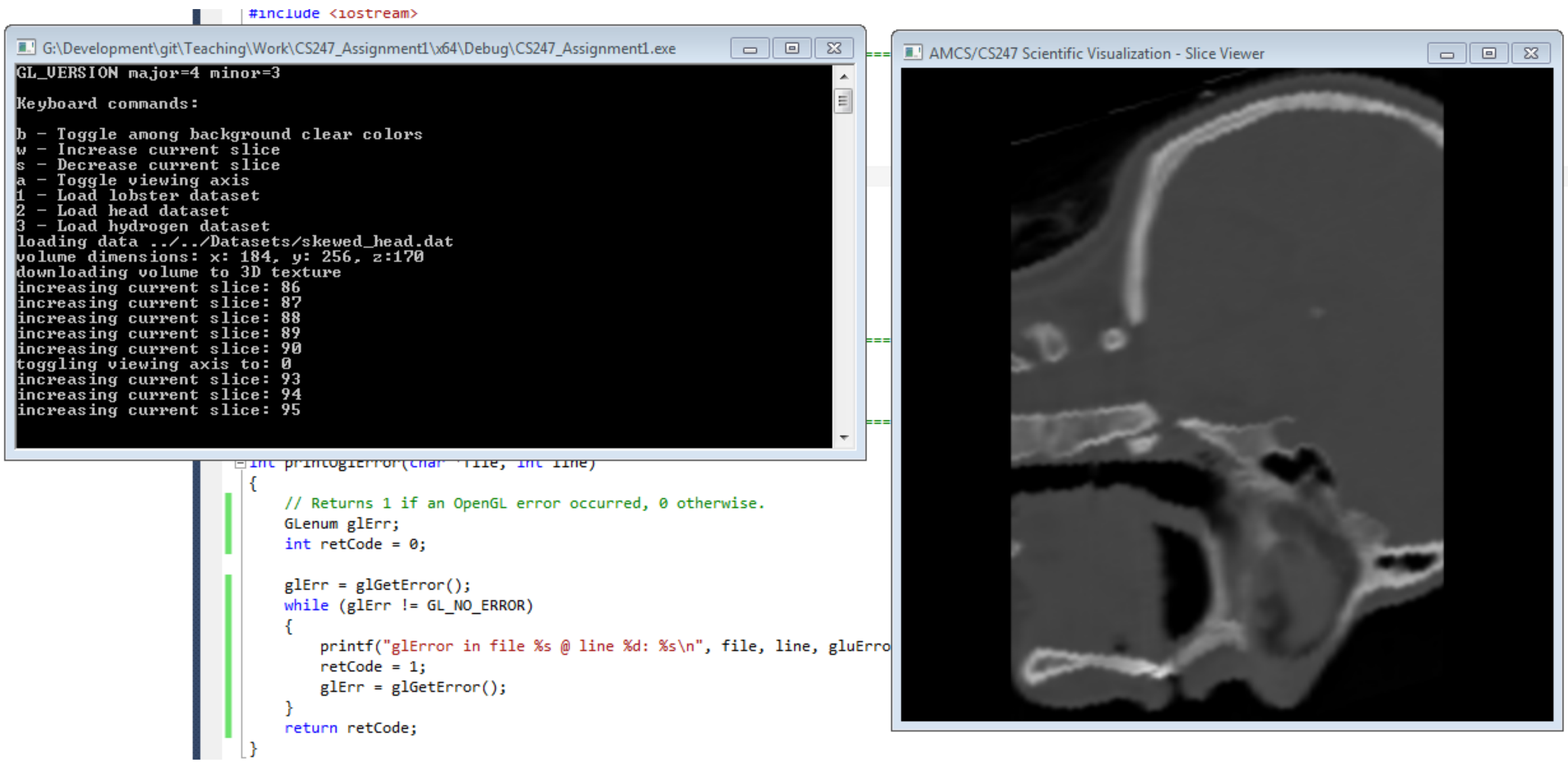
Programming Assignment #1 Example



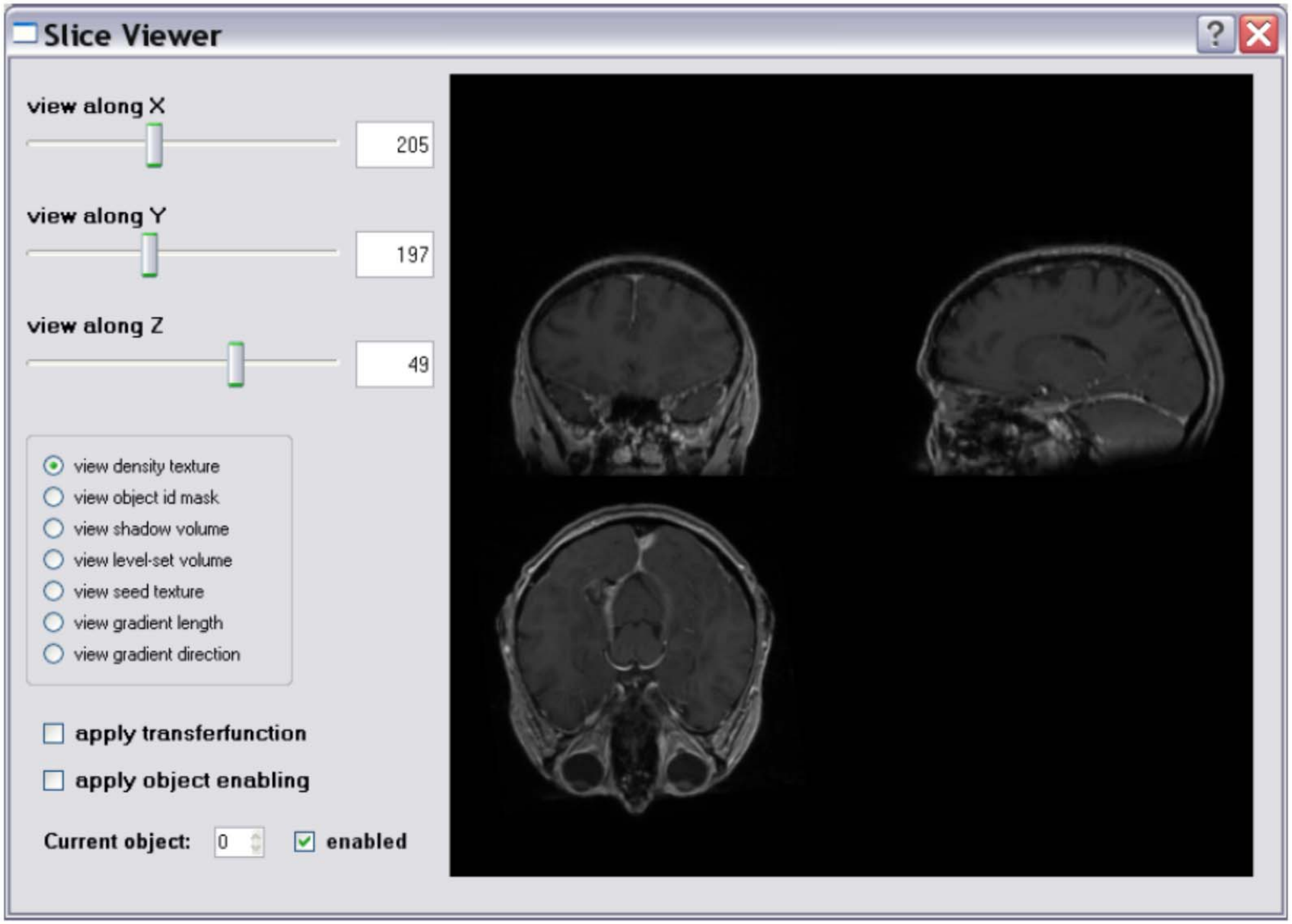
Programming Assignment #1 Example



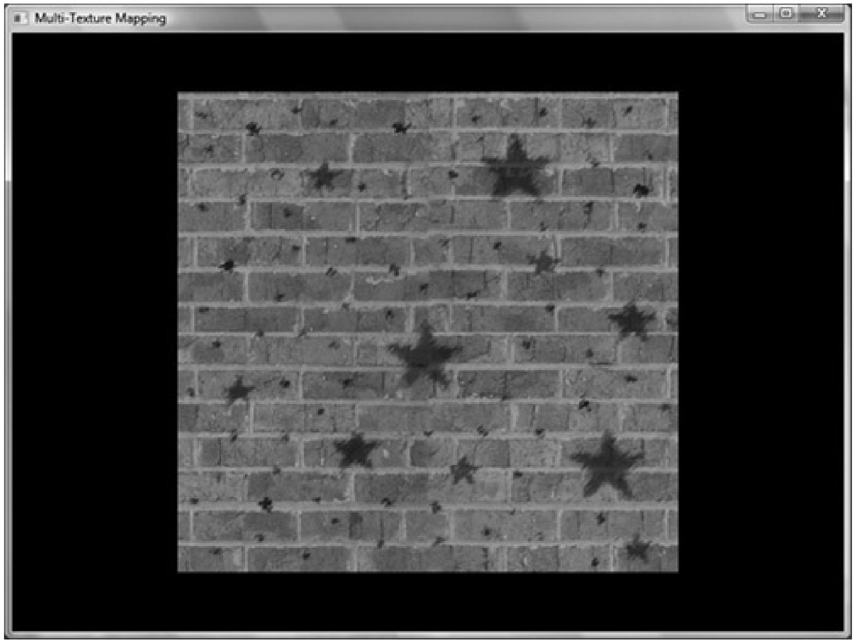
Programming Assignment #1 Example

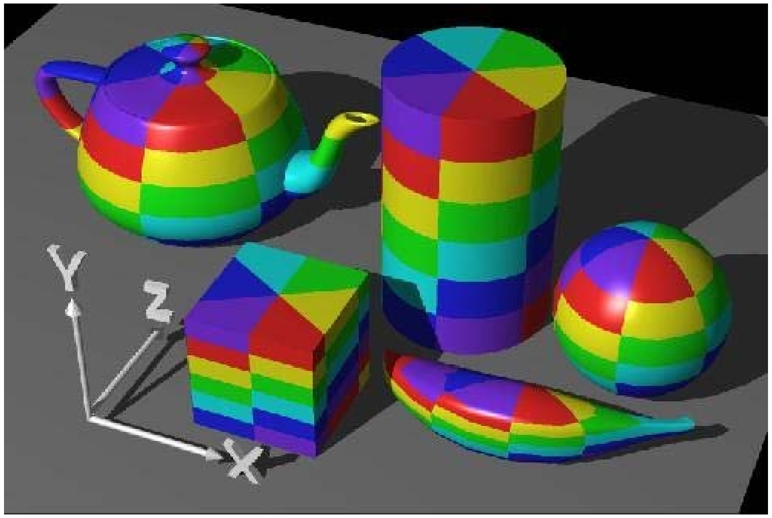


Programming Assignment #1 Example

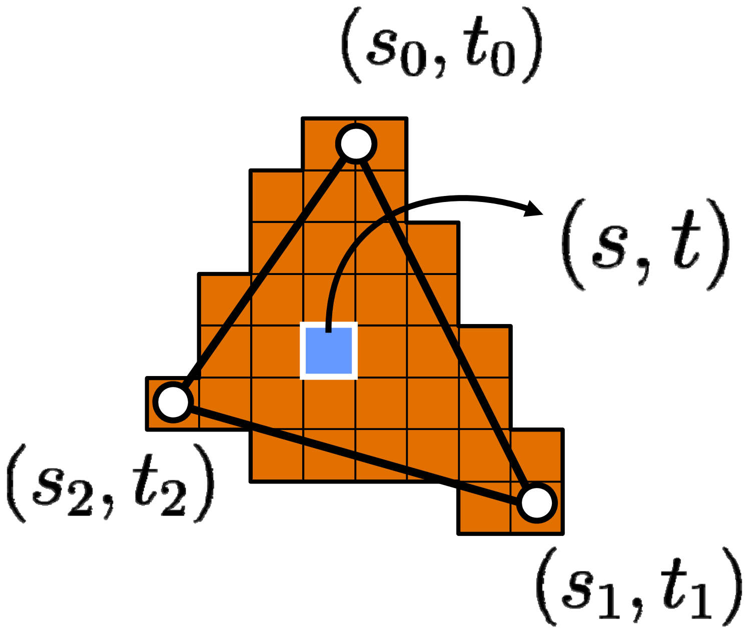


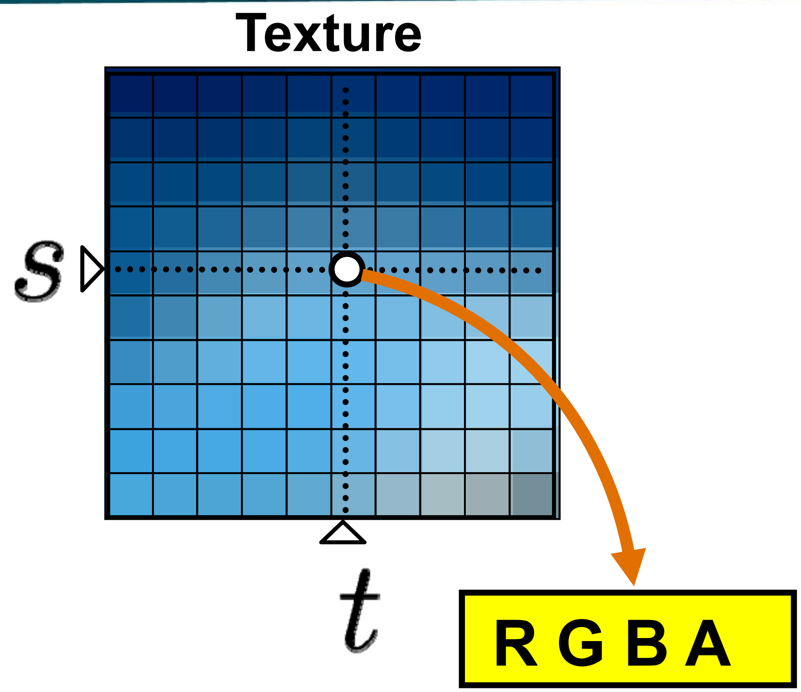
Texture Mapping





2D Texture Mapping

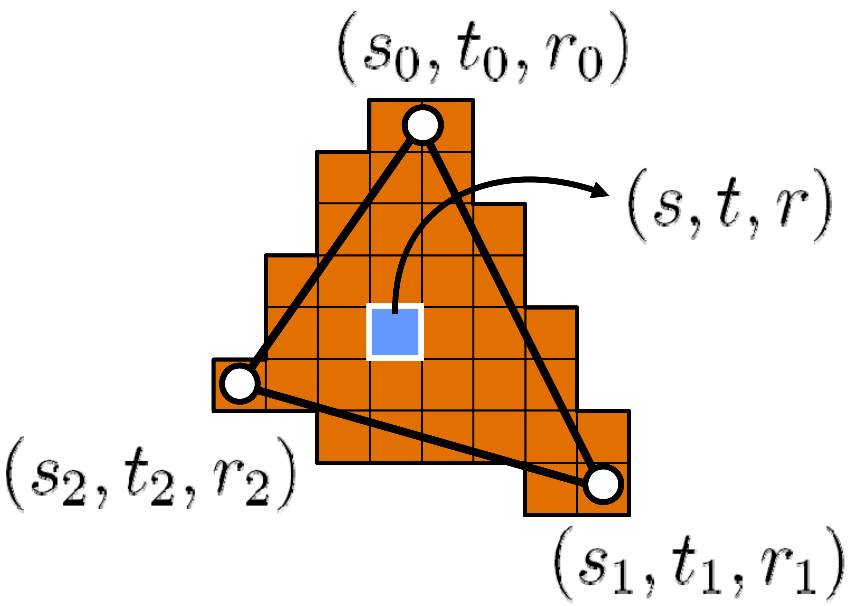




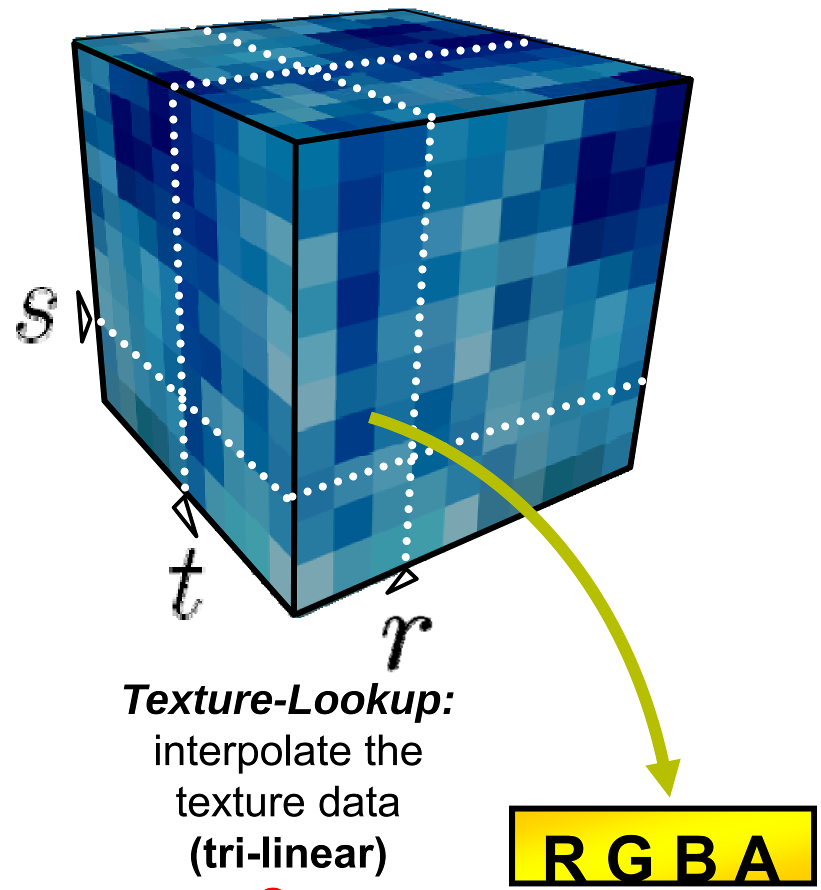
For each fragment:
interpolate the
texture coordinates
(barycentric)
Or:
Use arbitrary, computed coordinates

Texture-Lookup:
interpolate the
texture data
(bi-linear)
Or:
Nearest-neighbor for “array lookup”

3D Texture Mapping



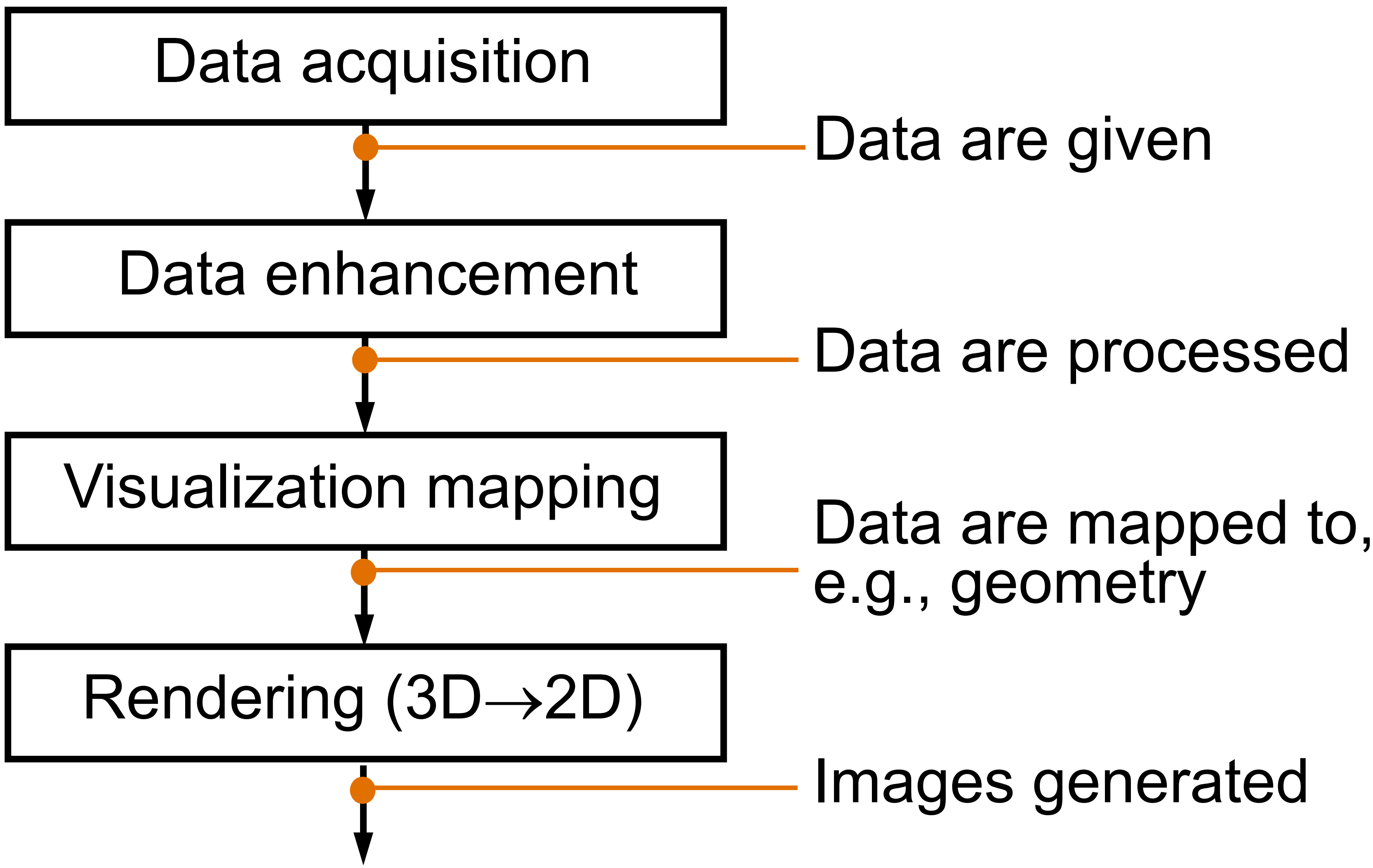
For each fragment:
interpolate the
texture coordinates
(barycentric)
Or:
Use arbitrary, computed coordinates



Texture-Lookup:
interpolate the
texture data
(tri-linear)
Or:
Nearest-neighbor for "array lookup"

The Visualization Pipeline

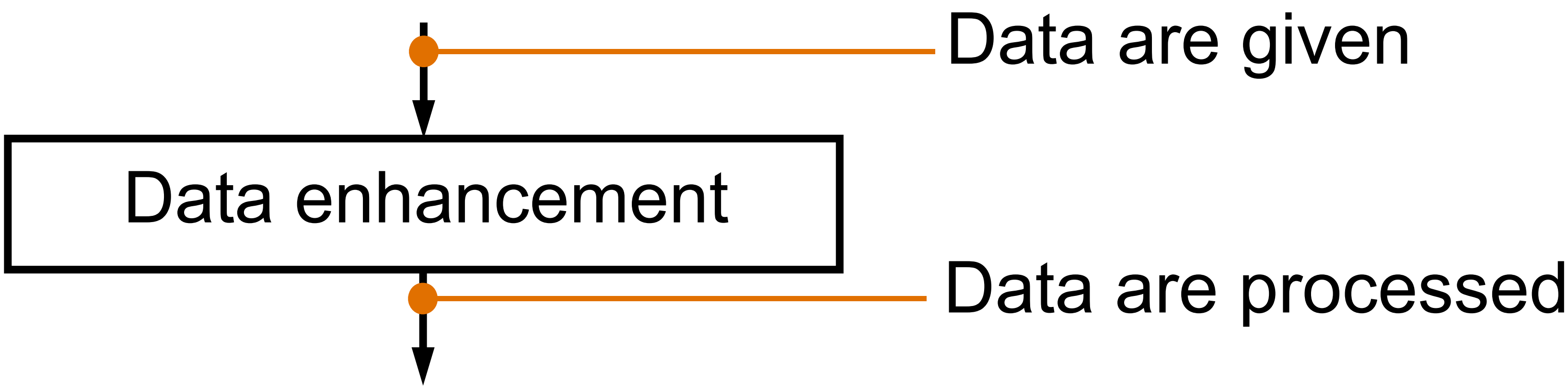
The Visualization Pipeline – Overview



The Visualization Pipeline – Stage 1

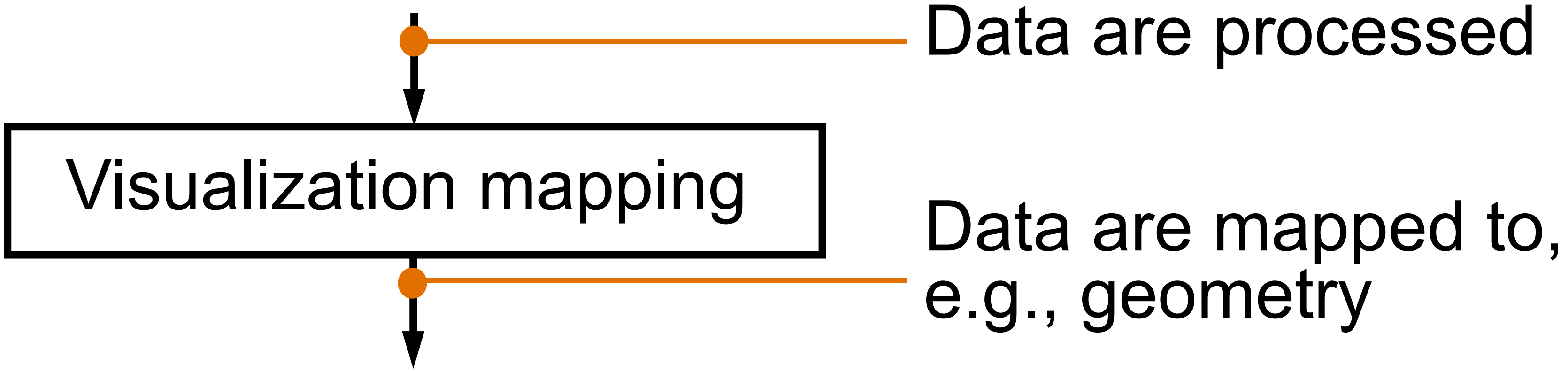
- Measurements, e.g., CT/MRI
- Simulation, e.g., flow simulation
- Modeling, e.g., game theory

The Visualization Pipeline – Stage 2



- Filtering, e.g, smoothing (de-noising, …)
- Resampling, e.g., on a different-resolution grid
- Data derivation, e.g., gradients, curvature
- Data interpolation, e.g., linear, cubic, …

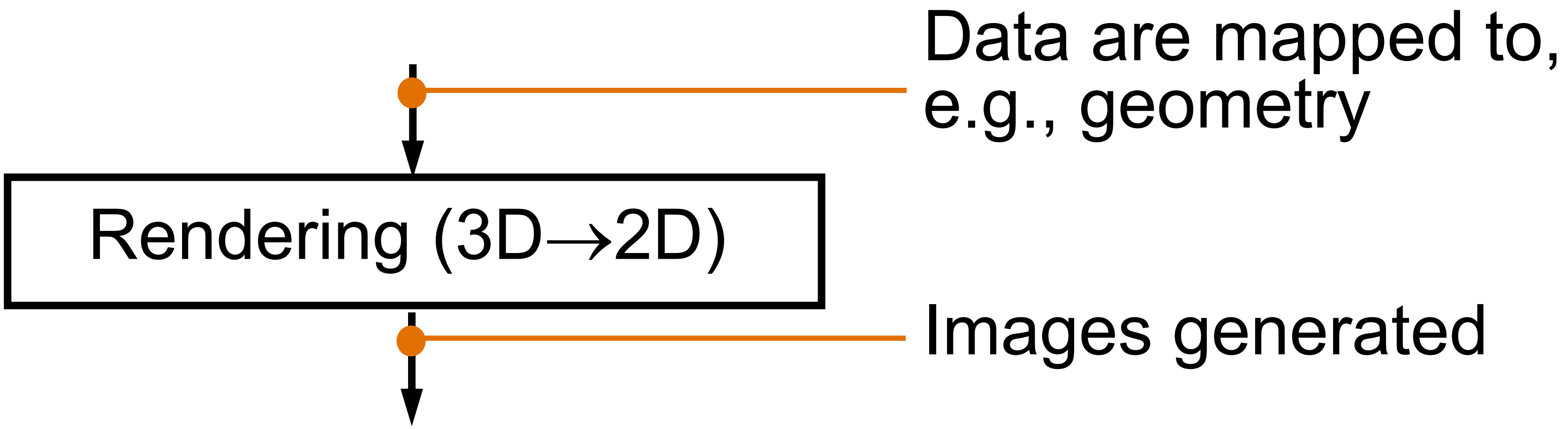
The Visualization Pipeline – Stage 3



Make data “renderable”

- Iso-surface calculation
- Glyphs, icons determination
- Graph-layout calculation
- Voxel attributes: color, transparency, …

The Visualization Pipeline – Stage 4



Rendering = image generation with computer graphics

- Visibility calculation
- Illumination
- Compositing (combine transparent objects, …)
- Animation

Data Representation

Data – General Information

Data:

- Focus of visualization,
  everything is centered around the data
- Driving factor (besides user) in choice and attribution of the visualization technique
- Important questions:
  - Where do the data “live” (**data space**)
  - **Type** of the data
  - Which **representation** makes sense
    (secondary aspect)

Data Space

Where do the data "live"?

- Inherent spatial domain (**SciVis**):
  - 2D/3D data space given
  - examples: medical data, flow simulation data, GIS data, etc.
- No inherent spatial reference (**InfoVis**):
  - abstract data,
    spatial embedding through visualization
  - example: data bases
- **Aspects**: dimensionality, domain, coordinates,
  region of influence of samples (local, global)

Data Type

What type of data?

- **Data types**:
  - Scalar = numerical value
    (natural, integer, rational, real, complex numbers)
  - Non-numerical (categorical) values (e.g., blood type)
  - Multi-dimensional values, i.e., codomain (n-dim. vectors, second-order ($n \times n$) tensors, higher-order tensors, ...)
  - Multi-modal values (vectors of data with varying type [e.g., row in a table])
- **Aspects**: dimensionality, codomain (superset of range/image)

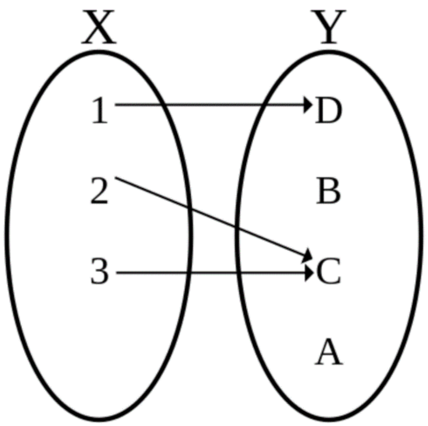
Data == Functions

Mathematical Functions

Associates every element of a set (e.g., X) with *exactly one* element of another set (e.g., Y)

Maps from domain (X) to codomain (Y)

$$f : X \to Y$$
$$x \mapsto f(x)$$



Also important: *range/image*; *preimage*; continuity, differentiability, dimensionality, ...

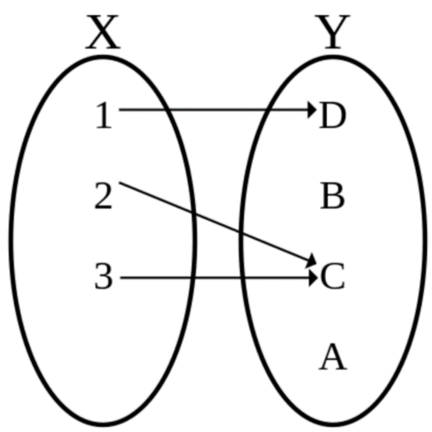
Graph of a function (mathematical definition):

$$G(f) := \{(x, f(x)) | x \in X\} \subset X \times Y$$

Mathematical Functions

Associates every element of a set (e.g., X) with *exactly one* element of another set (e.g., Y)

Maps from domain (X) to codomain (Y)



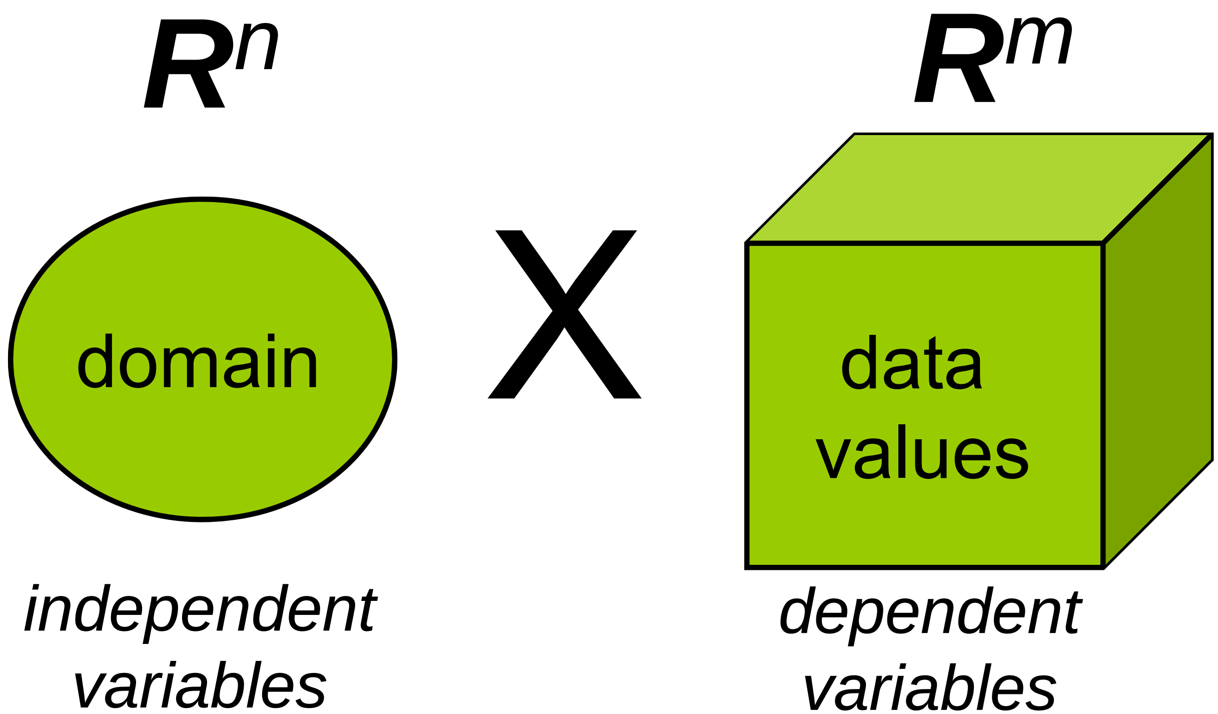
$$f : \mathbb{R}^n \to \mathbb{R}^m$$
$$x \mapsto f(x)$$

Also important: *range/image*; *preimage*; continuity, differentiability, dimensionality, ...

Graph of a function (mathematical definition):

$$G(f) := \{(x, f(x)) | x \in \mathbb{R}^n\} \subset \mathbb{R}^n \times \mathbb{R}^m \simeq \mathbb{R}^{n+m}$$

Data Representation



scientific data $\subseteq \boldsymbol{R}^{n+m}$

Example: Scalar Fields

2D scalar field

$$f : \mathbb{R}^2 \to \mathbb{R}$$
$$x \mapsto f(x)$$

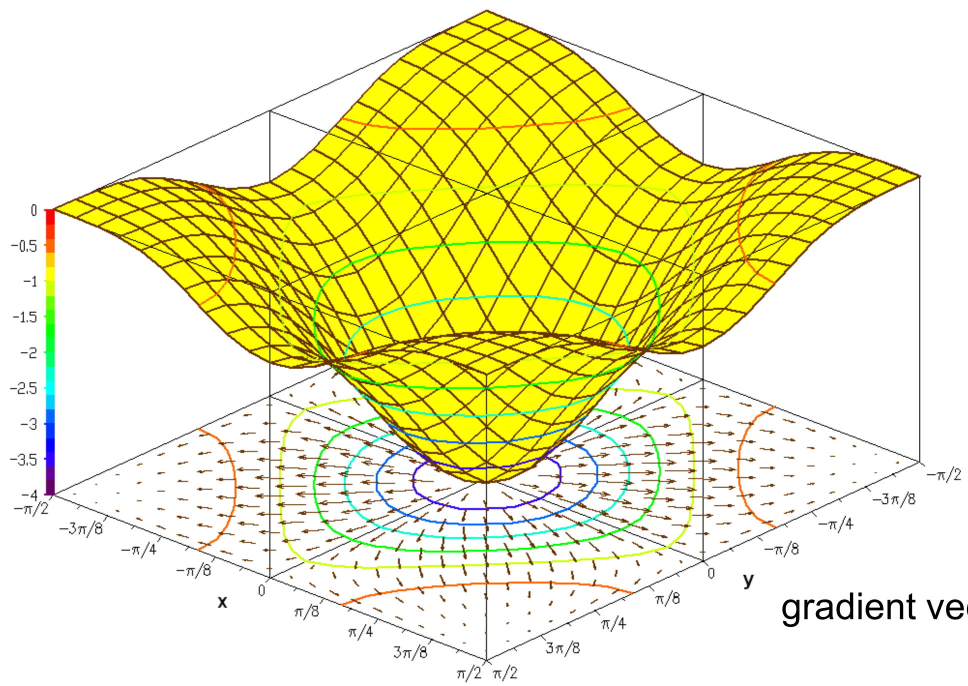
Graph: $G(f) := \{(x, f(x)) | x \in \mathbb{R}^2\} \subset \mathbb{R}^2 \times \mathbb{R} \simeq \mathbb{R}^3$

pre-image

$$S(c) := f^{-1}(c)$$

iso-contour

$$(\nabla f \neq 0)$$



gradient vector field ∇f

Example: Scalar Fields

3D scalar field

$$f : \mathbb{R}^3 \to \mathbb{R}$$
$$x \mapsto f(x)$$

Graph: $G(f) := \{(x, f(x)) | x \in \mathbb{R}^3\} \subset \mathbb{R}^3 \times \mathbb{R} \simeq \mathbb{R}^4$

pre-image

$$S(c) := f^{-1}(c)$$

iso-surface

$$(\nabla f \neq 0)$$

?

Visualization Examples

data	description	visualization example
$N^1 \rightarrow R^1$	value series	bar chart, pie chart, etc.
$R^1 \rightarrow R^1$	scalar function over R	(line) graph
$R^2 \rightarrow R^1$	scalar function over R^2	2D-height map in 3D, contour lines in 2D, false color map
$R^2 \rightarrow R^2$	2D vector field	hedgehog plot, LIC, streamlets, etc.
$R^3 \rightarrow R^1$	scalar function over R^3 (3D densities)	iso-surfaces in 3D, volume rendering
$R^3 \rightarrow R^3$	3D vector field	streamlines/pathlines in 3D

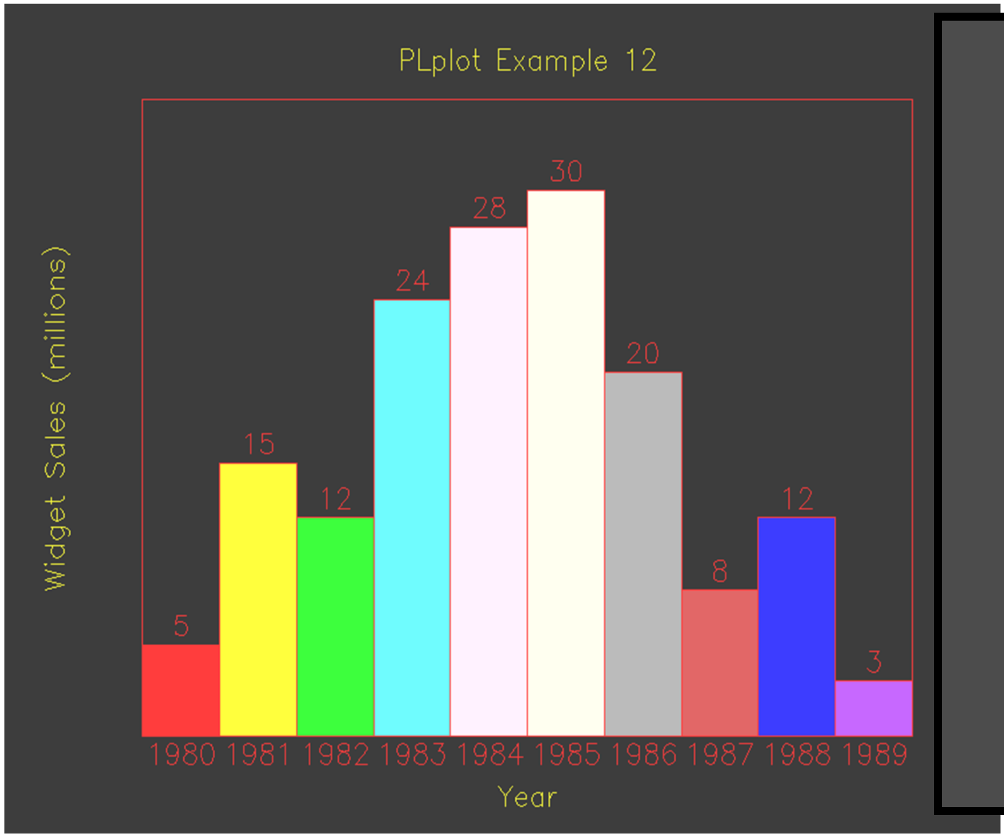
Data Space (Domain) vs. Data Type (Codomain)

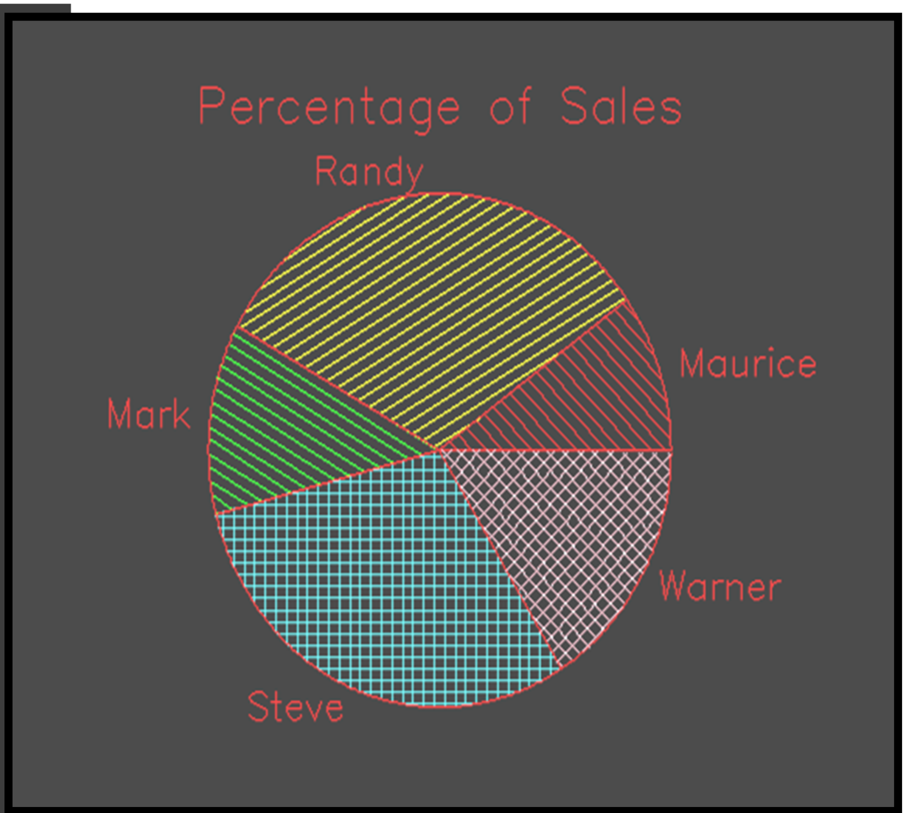
	1D	2D	3D
1D	y=f(x)		spatial curve **x**(t)
2D		2D flow **v**(**x**)	
3D	CT data s(**x**)		3D flow **v**(**x**)

Examples

Visualization Examples

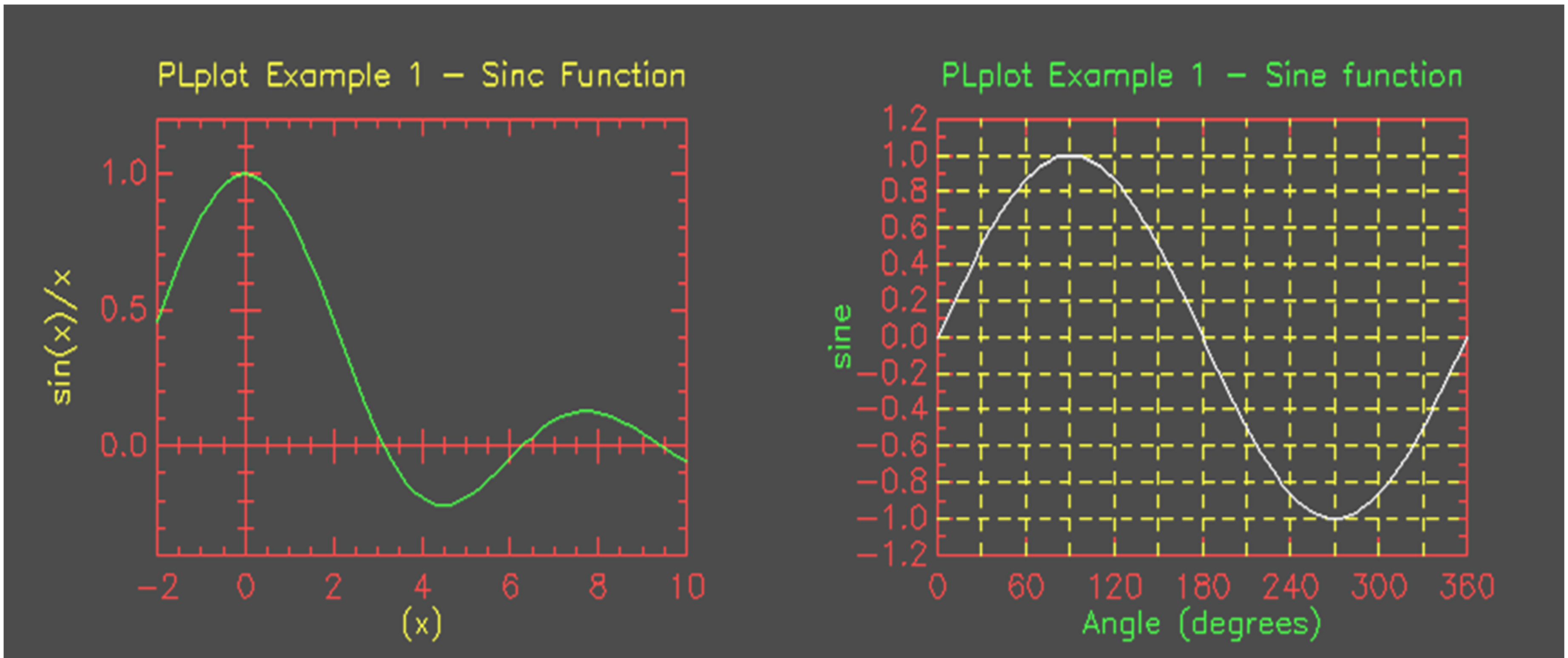
data	description	visualization example
$N^1 \rightarrow R^1$	value series	bar chart, pie chart, etc.





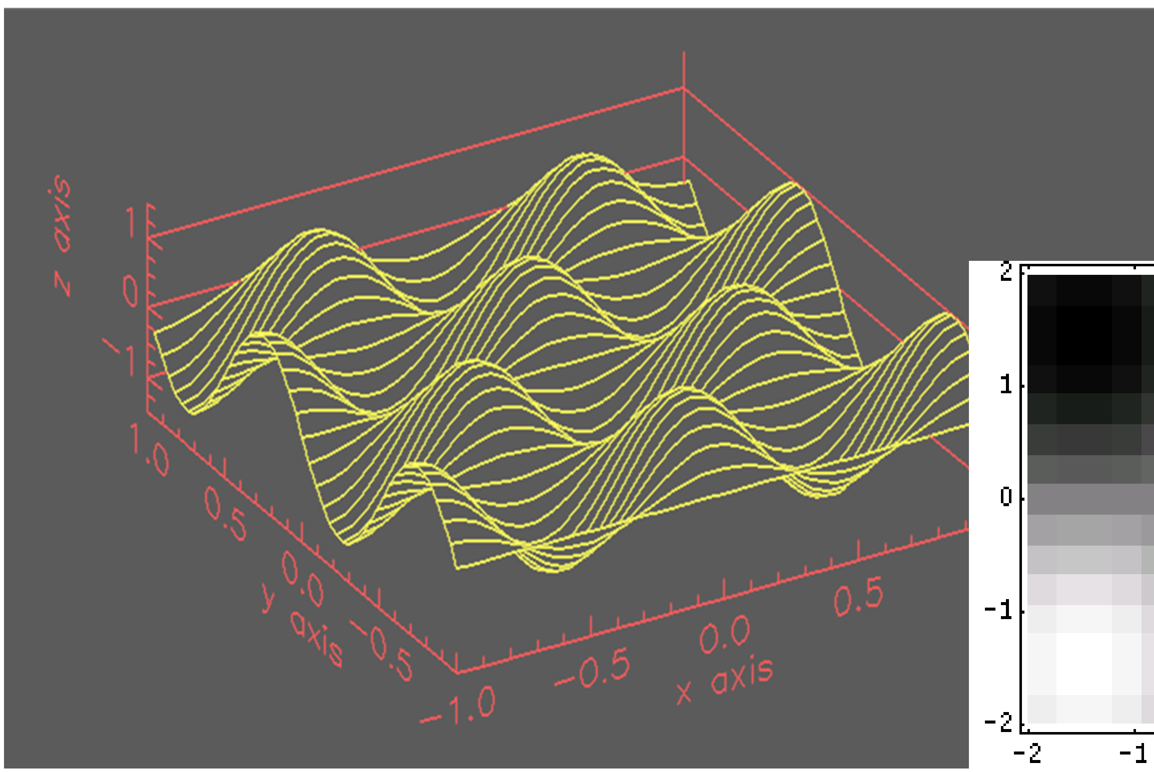
Visualization Examples

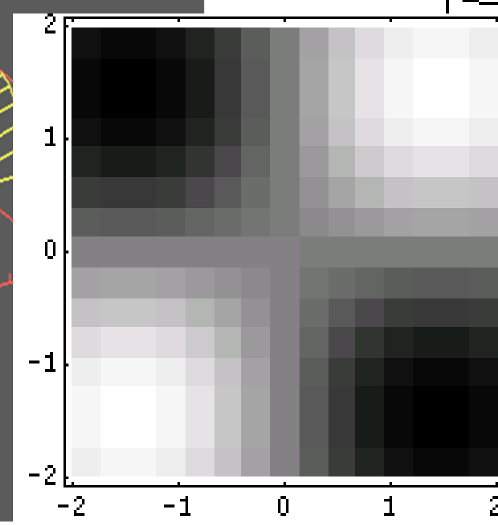
data	description	visualization example
$R^1 \rightarrow R^1$	function over R	(line) graph

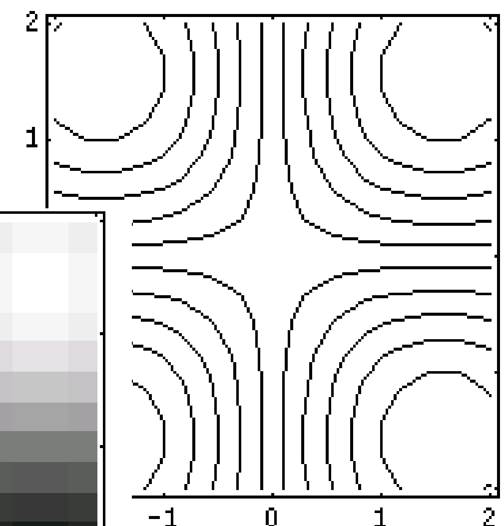


Visualization Examples

data	description	visualization example
$R^2 \rightarrow R^1$	function over R^2	2D-height map in 3D, contour lines in 2D, false colors (heat map)

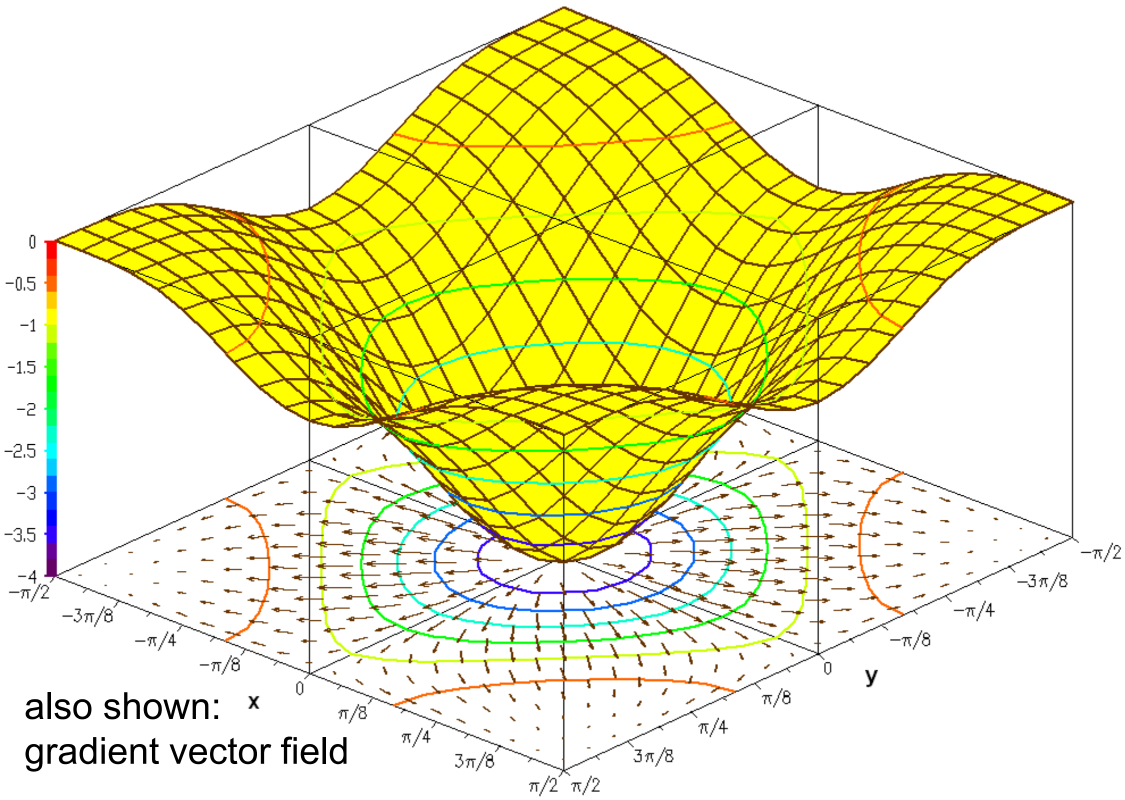




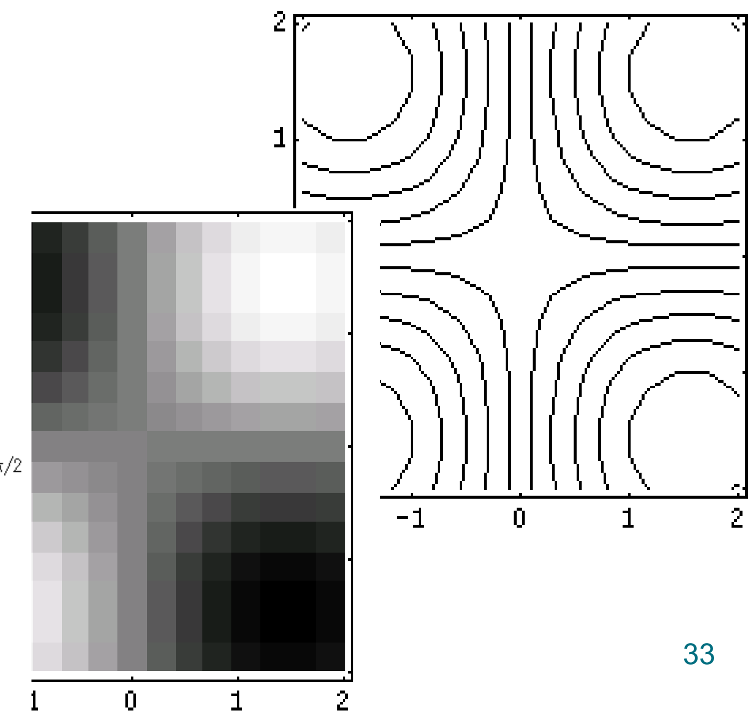


Visualization Examples

data	description	visualization example
$R^2 \rightarrow R^1$	function over R^2	2D-height map in 3D, contour lines in 2D, false colors (heat map)

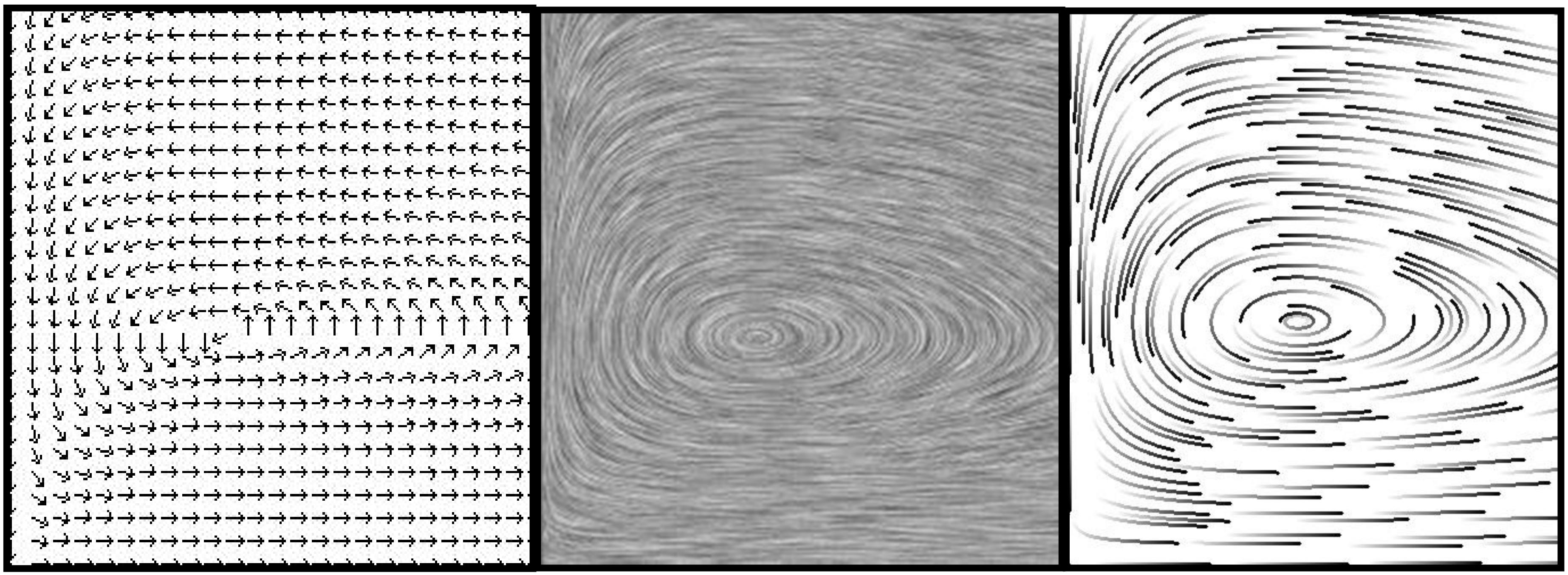


also shown:
gradient vector field



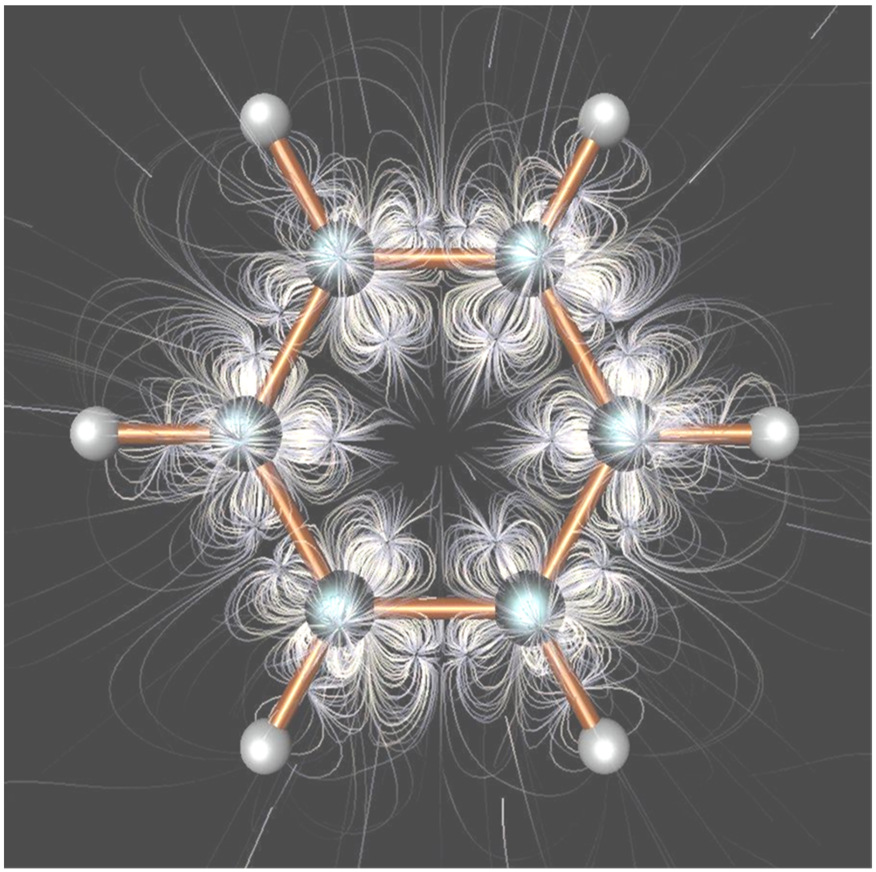
Visualization Examples

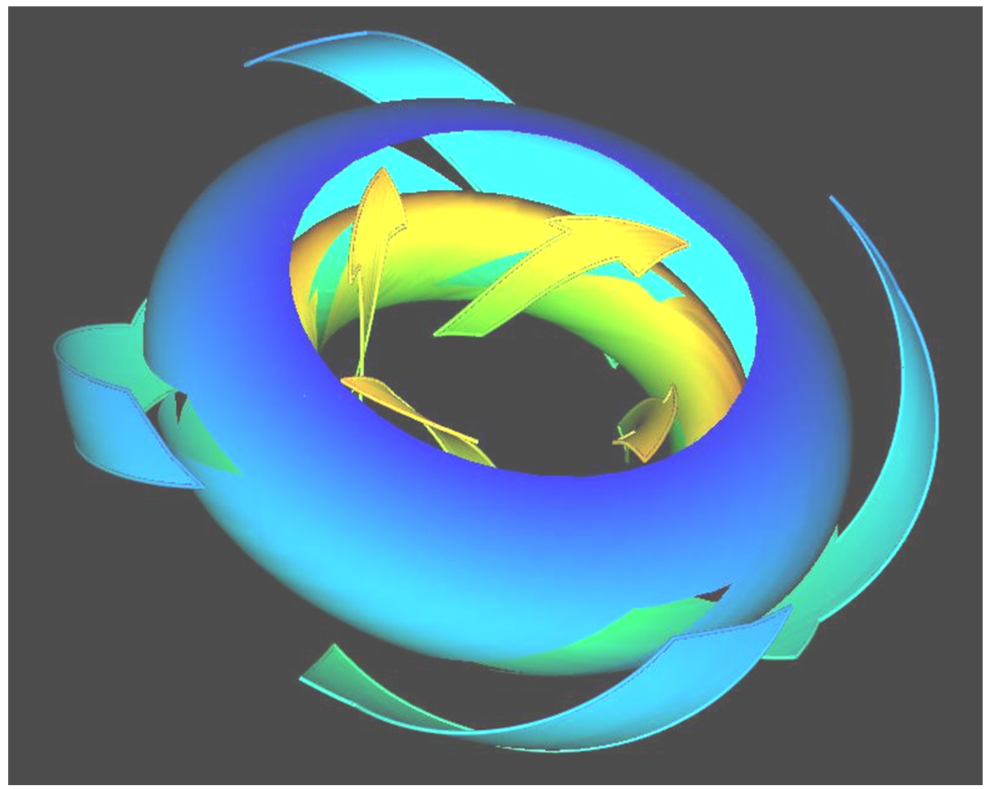
data	description	visualization example
$R^2 \rightarrow R^2$	2D-vector field	hedgehog plot, LIC, streamlets, etc



Visualization Examples

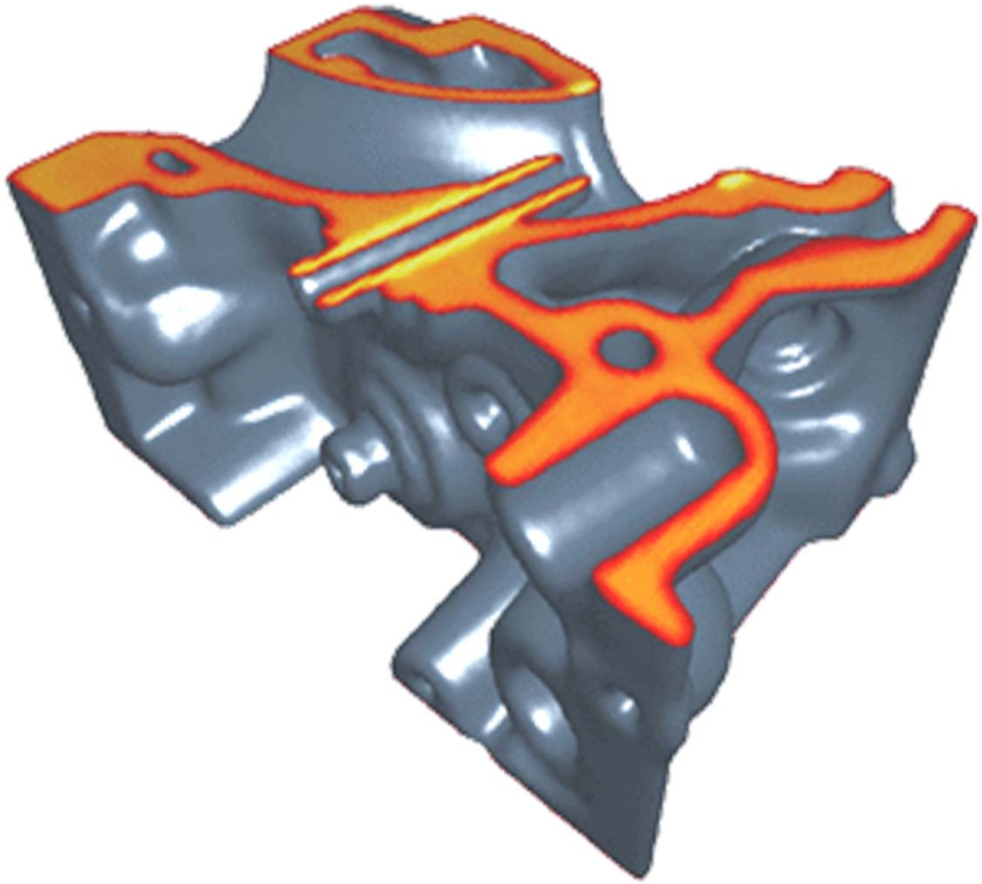
data	description	visualization example
$R^3 \rightarrow R^3$	3D-flow	streamlines, streamsurfaces

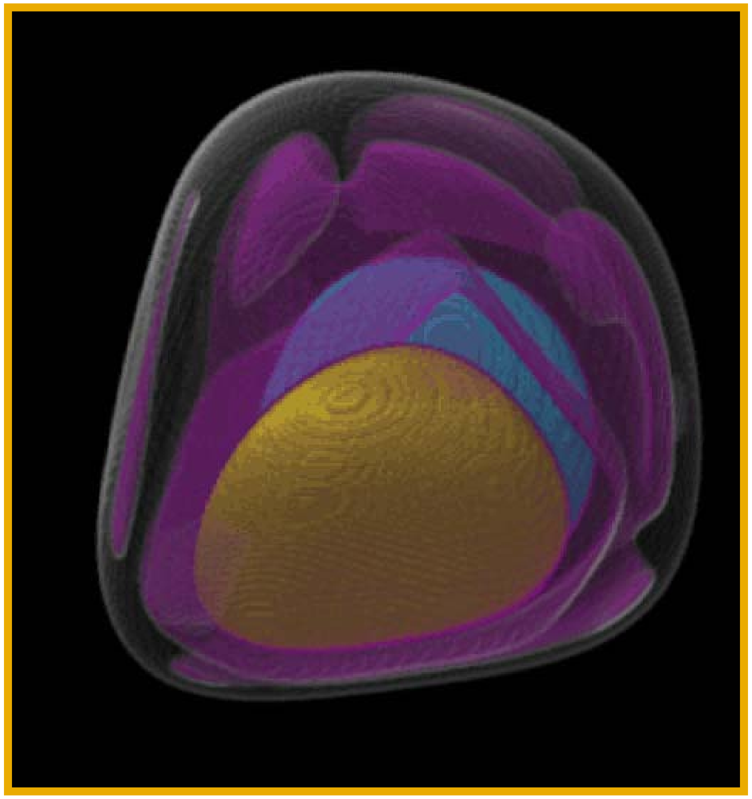




Visualization Examples

data	description	visualization example
$R^3 \to R^1$	3D-densities	iso-surfaces in 3D, volume rendering



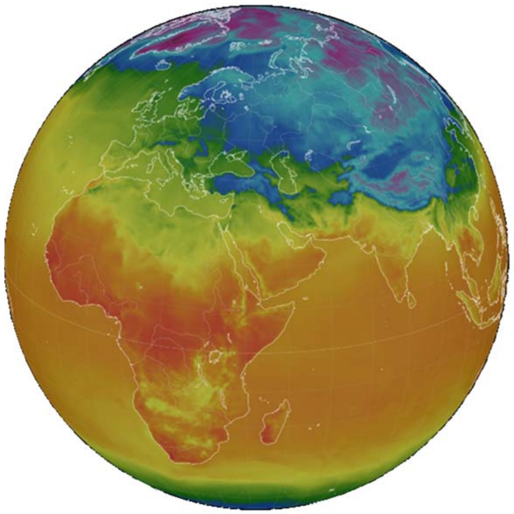


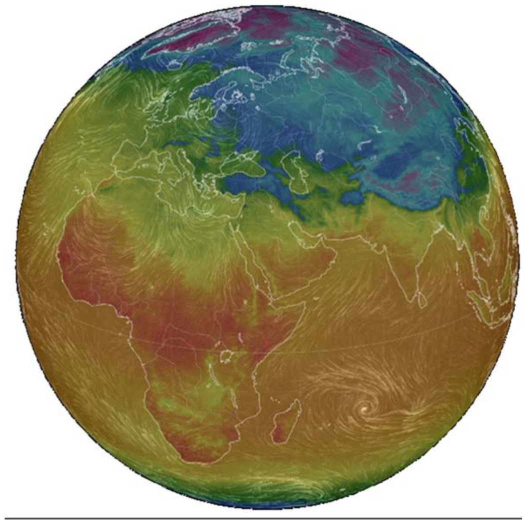
Domain Not Always Euclidean

Manifolds

…

- Scalar, vector, tensor fields on manifolds





Thank you.

Thanks for material

- Helwig Hauser
- Eduard Gröller
- Daniel Weiskopf
- Torsten Möller
- Ronny Peikert
- Philipp Muigg
- Christof Rezk-Salama