Spatial Visualization: Grids

CS 5630/6630, Fall 2016
Alexander Lex
Guest Lecturer: Aaron Knoll

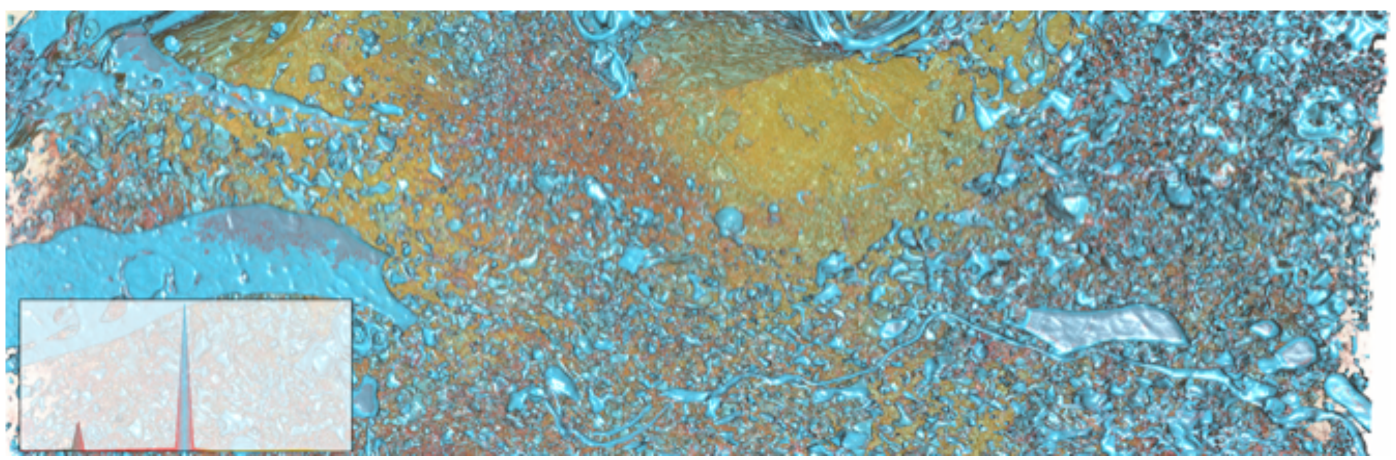
www.sci.utah.edu

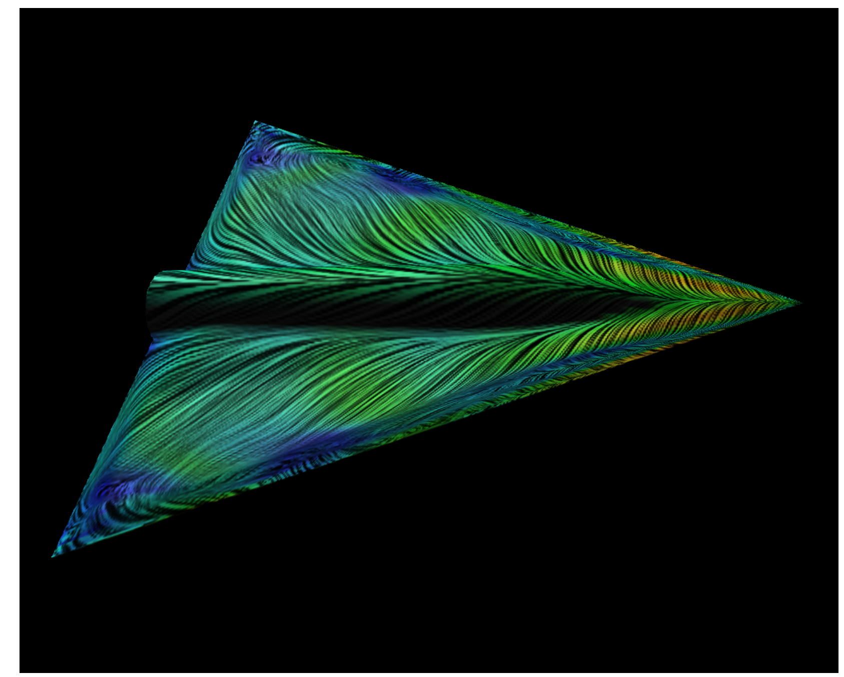
The traditional “branches” of visualization (IEEE Visweek)

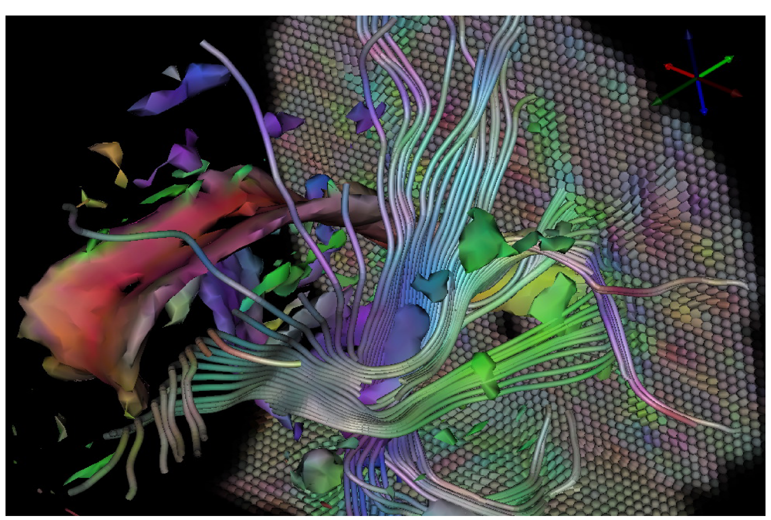
- Scientific (spatial) Visualization
- Information Visualization
- Visual Analytics

Scientific visualization

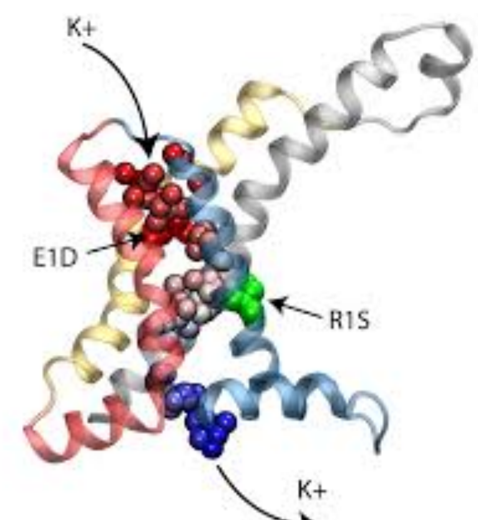
- Data have **spatial** context (usually from simulation or scan)
- Map spatial quantities to colors or geometry,
  f(space, time) -> rgba
- **2D or 3D graphics for visualization.**


Volume rendering


Flow visualization


Tensor field visualization

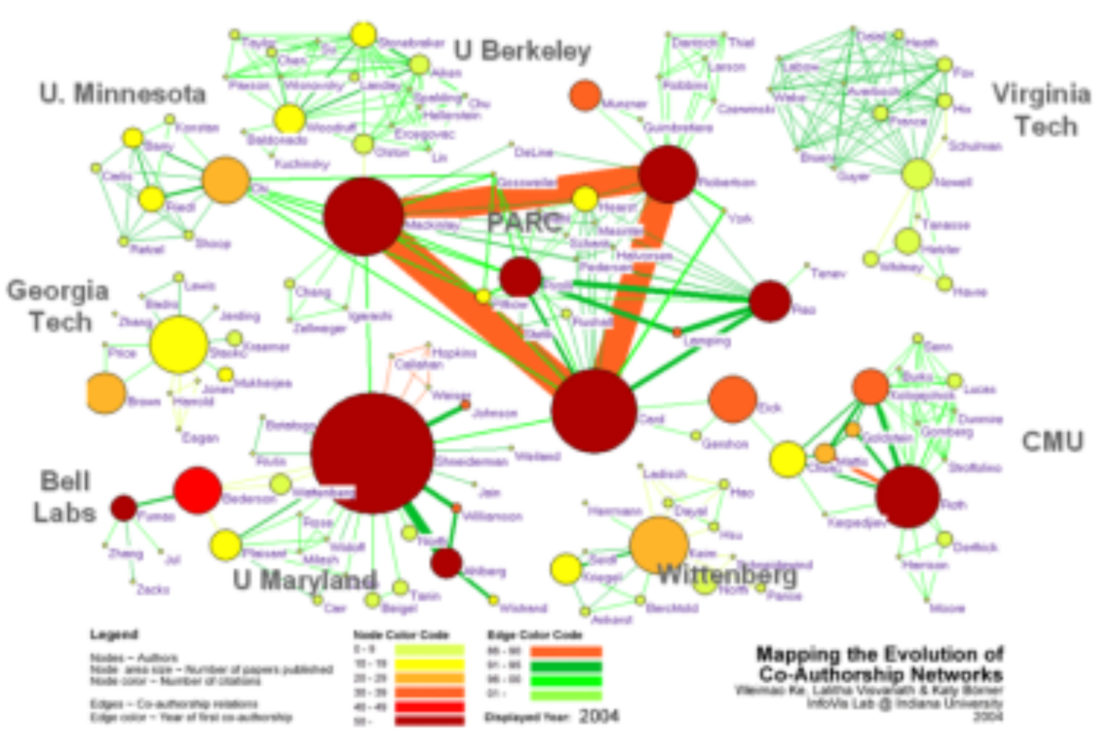

Map and GIS visualization



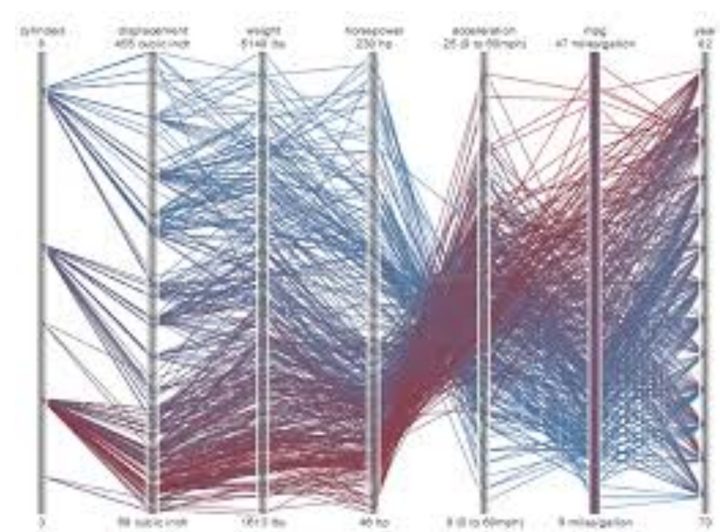
Molecular visualization

Information visualization

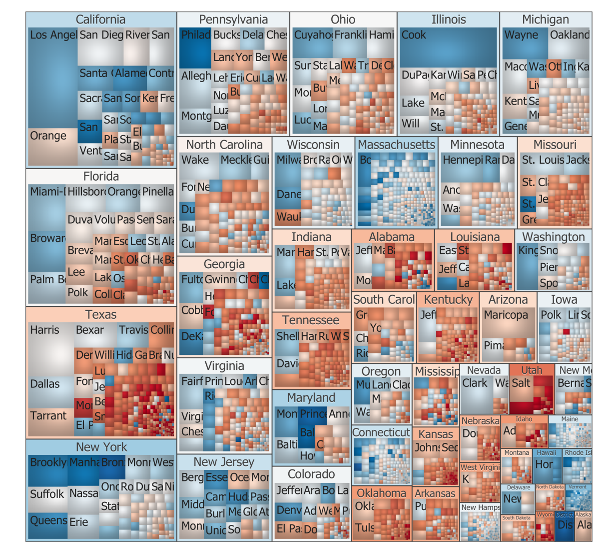
- Spatial position is secondary or non-existent.
- Illustrate relationships between abstract attributes.
- **Plots, charts, graphs, diagrams.**



Graph and network visualization

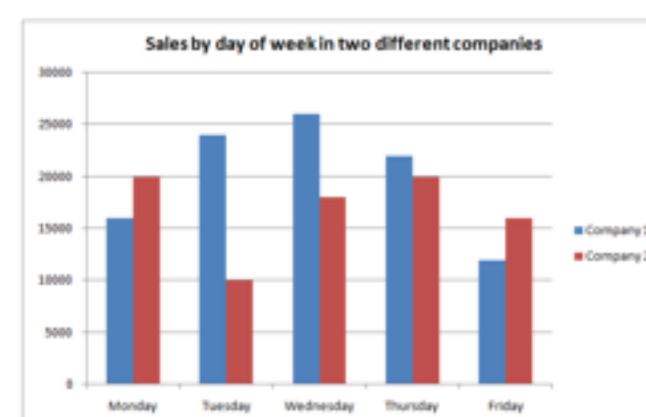


Parallel coordinates



Treemaps

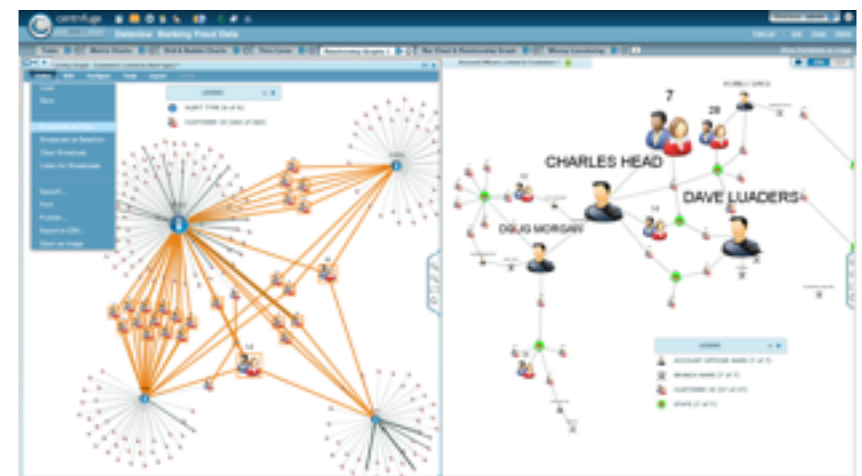
Scatterplots



Charts

Visual Analytics

- More about **interactive user interfaces** for data analysis.
- Uses techniques from both scientific visualization and information visualization, as well as statistics, perception, cognition.
- D3+Javascript, R, Matlab
- “Putting it all together”



Security visualization (Centrifuge)

Management Information Systems (SAS)

Genomics (Meyer et al. “Mizbee”)

www.sci.utah.edu

Scientific Visualization

- **“Sci-vis" is about interpreting and rendering spatial data.**
- Today:
  - where do spatial data come from? (science domains)
  - how are they represented (*grids*)
  - what can we do with them (*direct* and *indirect* vis)
  - interpolation
- Tues, Nov 17: Volumes
  Tues, Nov 29: Isosurfaces
  Thurs, Dec 1: Advanced topics: Vector and Tensor Fields

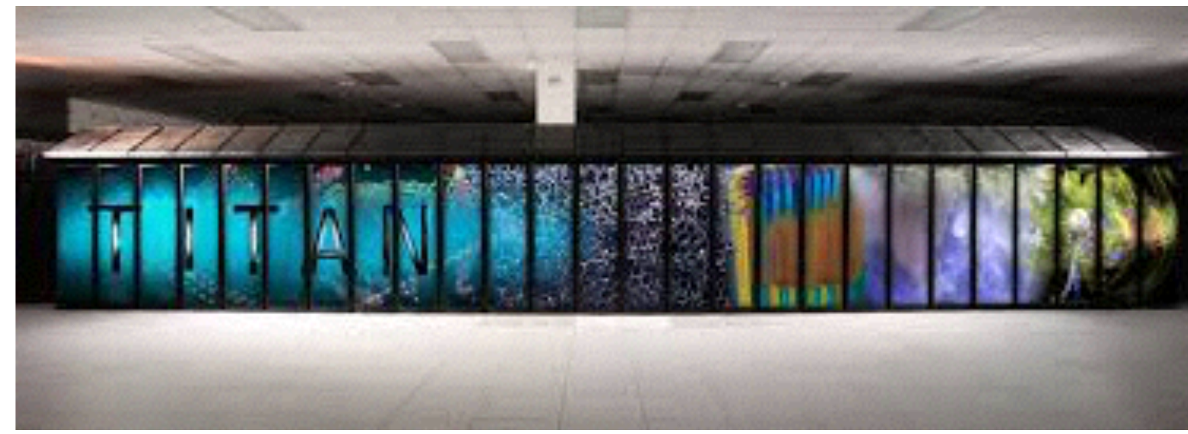
Scientific Visualization

- Data sources
- Data representation
  - fields
  - grids
- Data interpretation
  - The scientific visualization pipeline
  - Interpolation

Data sources

Computational Data

The output of scientific computing:
physics, chemistry, blood flow, neurophysiology, meteorology, climatology, astronomy…

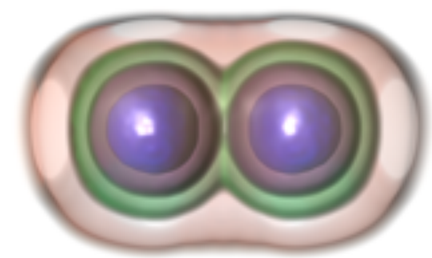


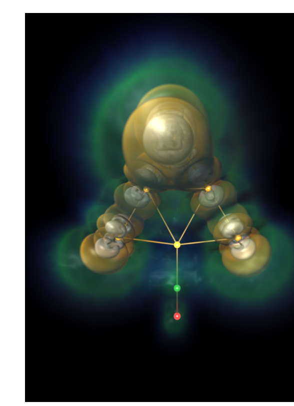
- Nuclear physics
- Quantum chemistry
- Molecular dynamics
- Computational fluid dynamics
- Rigid-body and structural mechanics
- Coarse-grained dynamics, agents simulations
- Meteorology
- Astrophysics
- Cosmology

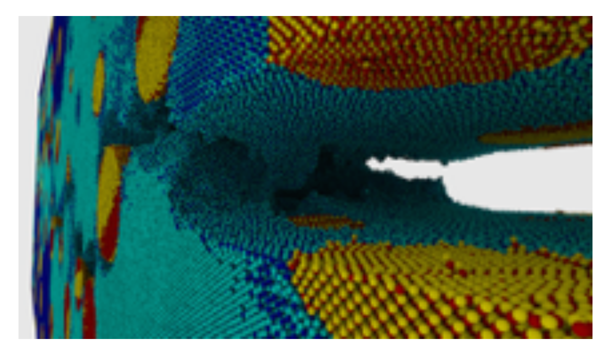
kilobytes

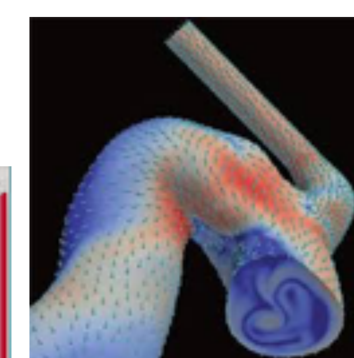
↓

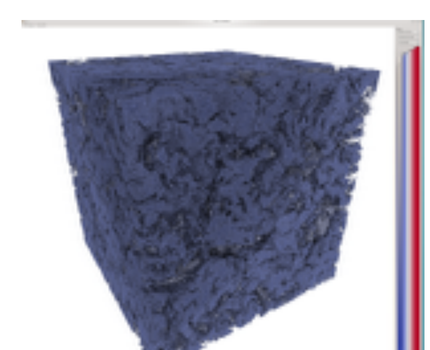
petabytes











www.sci.utah.edu

Scanned data

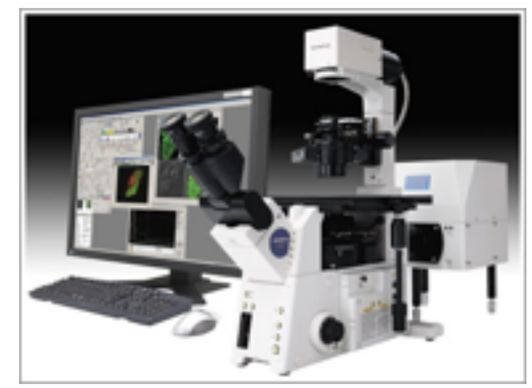
The output of instruments in medical imaging, microscopy, telescopy, GIS

- X-ray crystallography
- Synchrotron / radiation light sources
- Transmission electron microscopy
- Confocal microscopy
- Camera imagery
- Ultrasound
- Magnetic resonance imaging
- X-ray tomography
- Satellite
- Telescope

Angstroms

↓

Megaparsecs



www.sci.utah.edu

Data representation (grids)

Fields

- Mathematically, a *field* is a set of elements with addition, multiplication operators that satisfy the field axioms

name	addition	multiplication
associativity	$(a+b)+c=a+(b+c)$	$(a\,b)\,c=a\,(b\,c)$
commutativity	$a+b=b+a$	$a\,b=b\,a$
distributivity	$a\,(b+c)=a\,b+a\,c$	$(a+b)\,c=a\,c+b\,c$
identity	$a+0=a=0+a$	$a\cdot 1=a=1\cdot a$
inverses	$a+(-a)=0=(-a)+a$	$a\,a^{-1}=1=a^{-1}\,a$ if $a\neq 0$

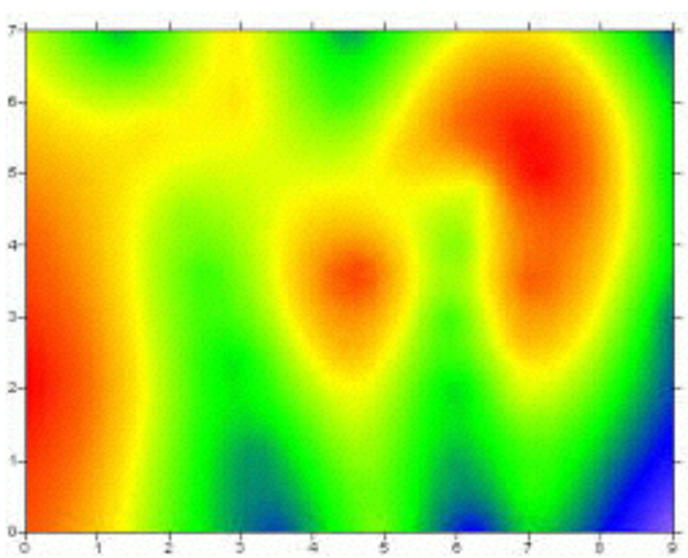
wolfram.com

- Intuitively, a field is a varying quantity defined continuously over space.

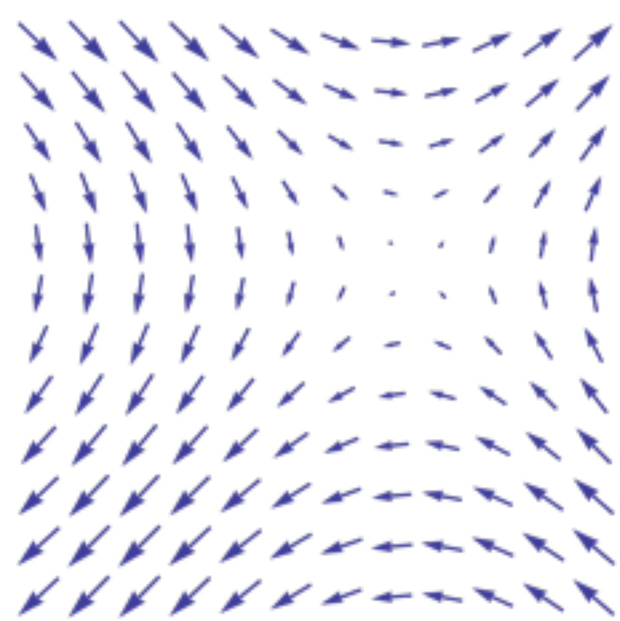
Fields

with a 2D domain

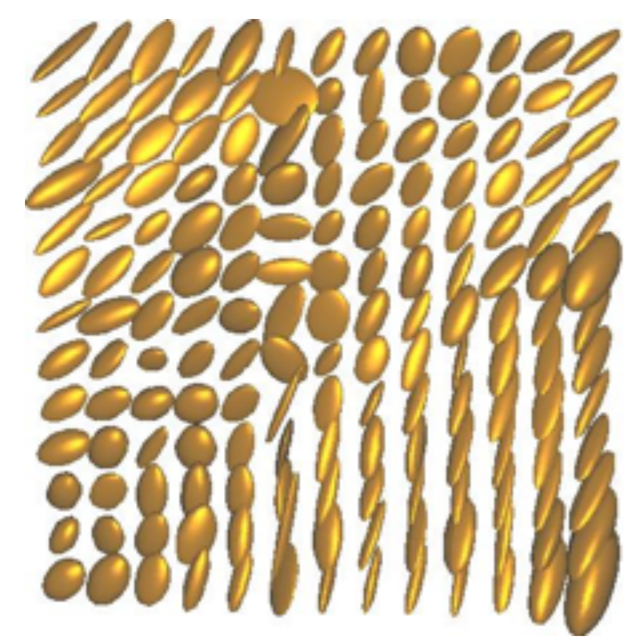
scalar field



vector field

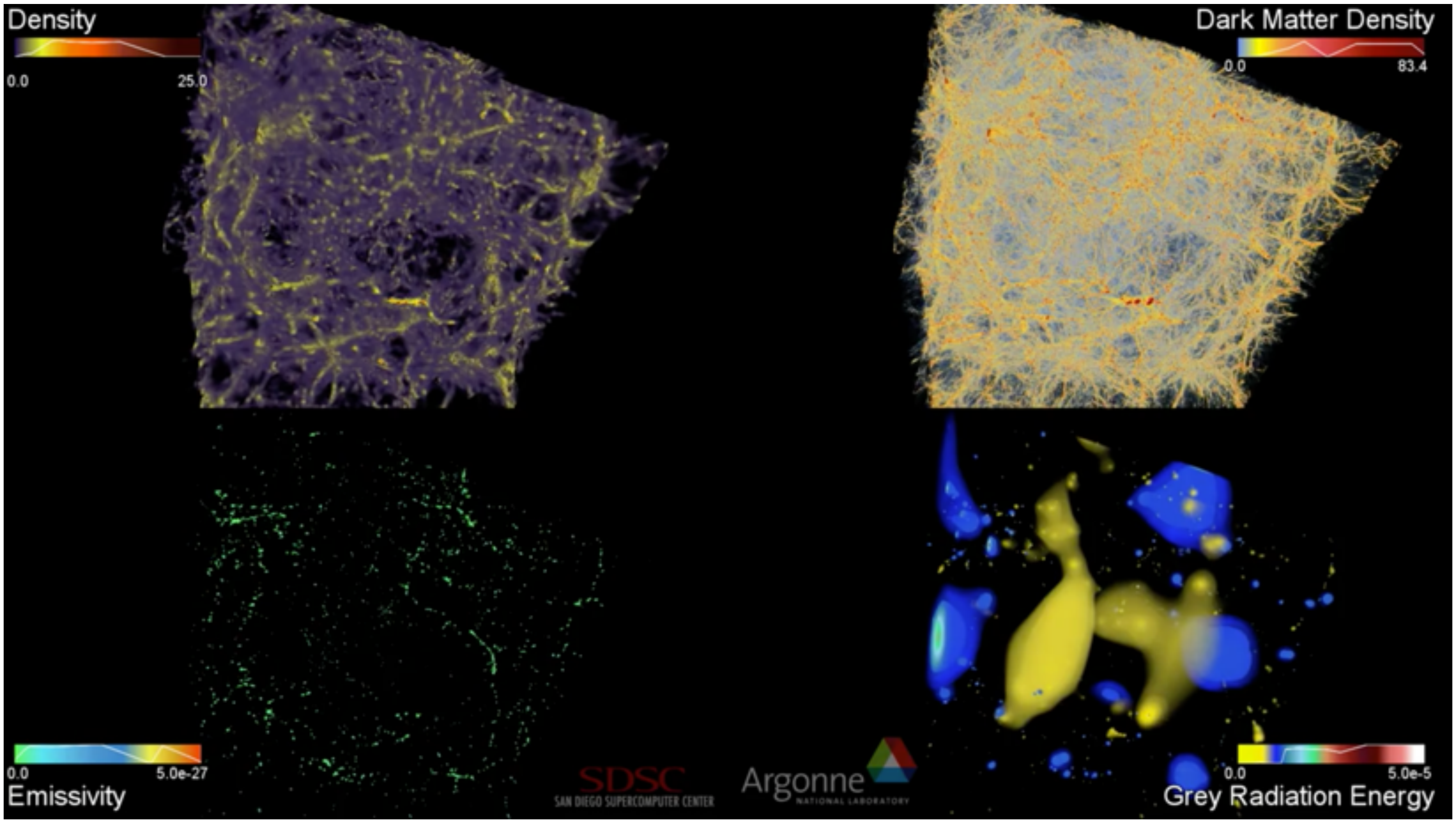


tensor field



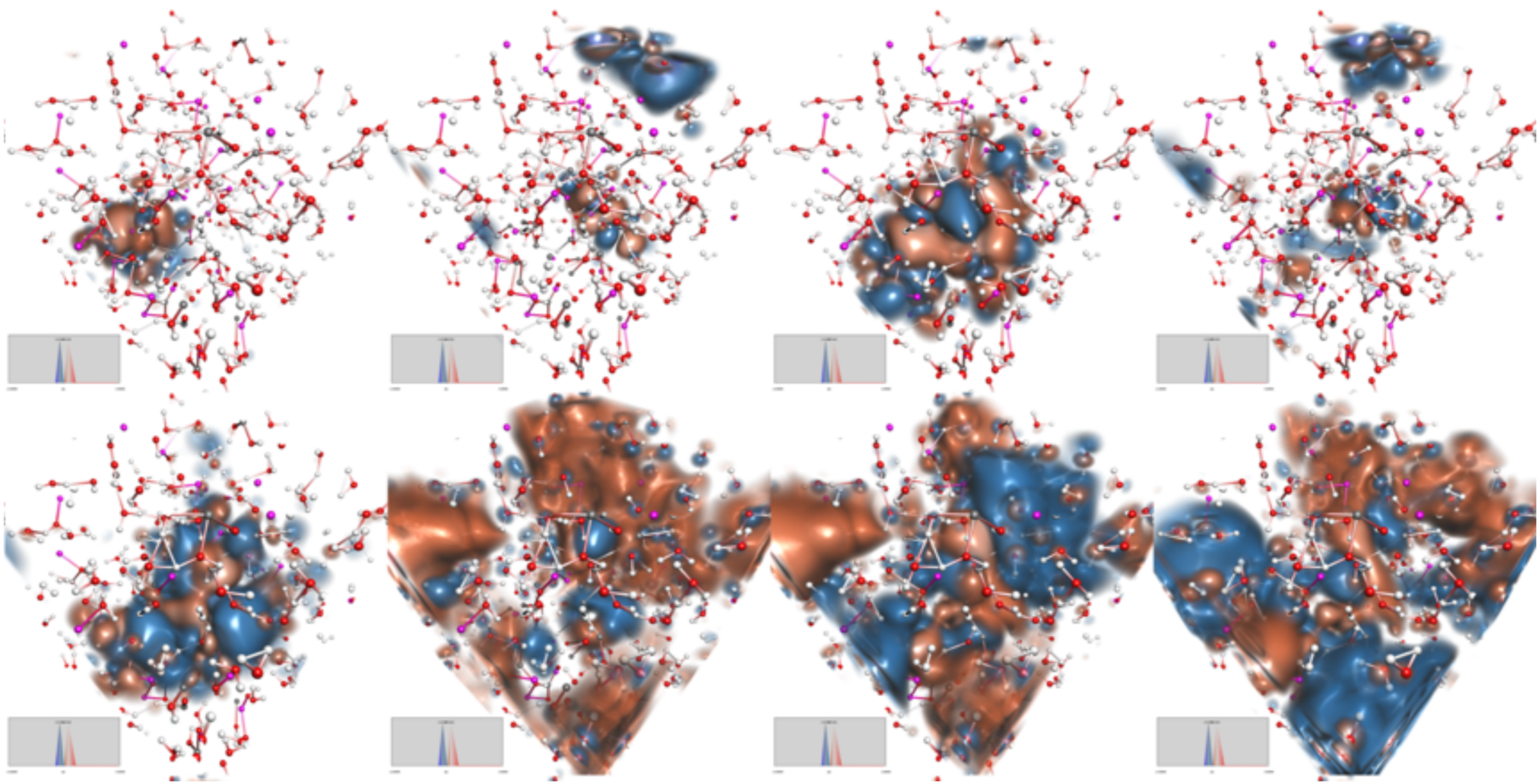
www.sci.utah.edu

Multifields



Radiation hydrodynamics in Enzo: Joe Insley (ANL), Rick Wagner (SDSC)
https://vimeo.com/17771397

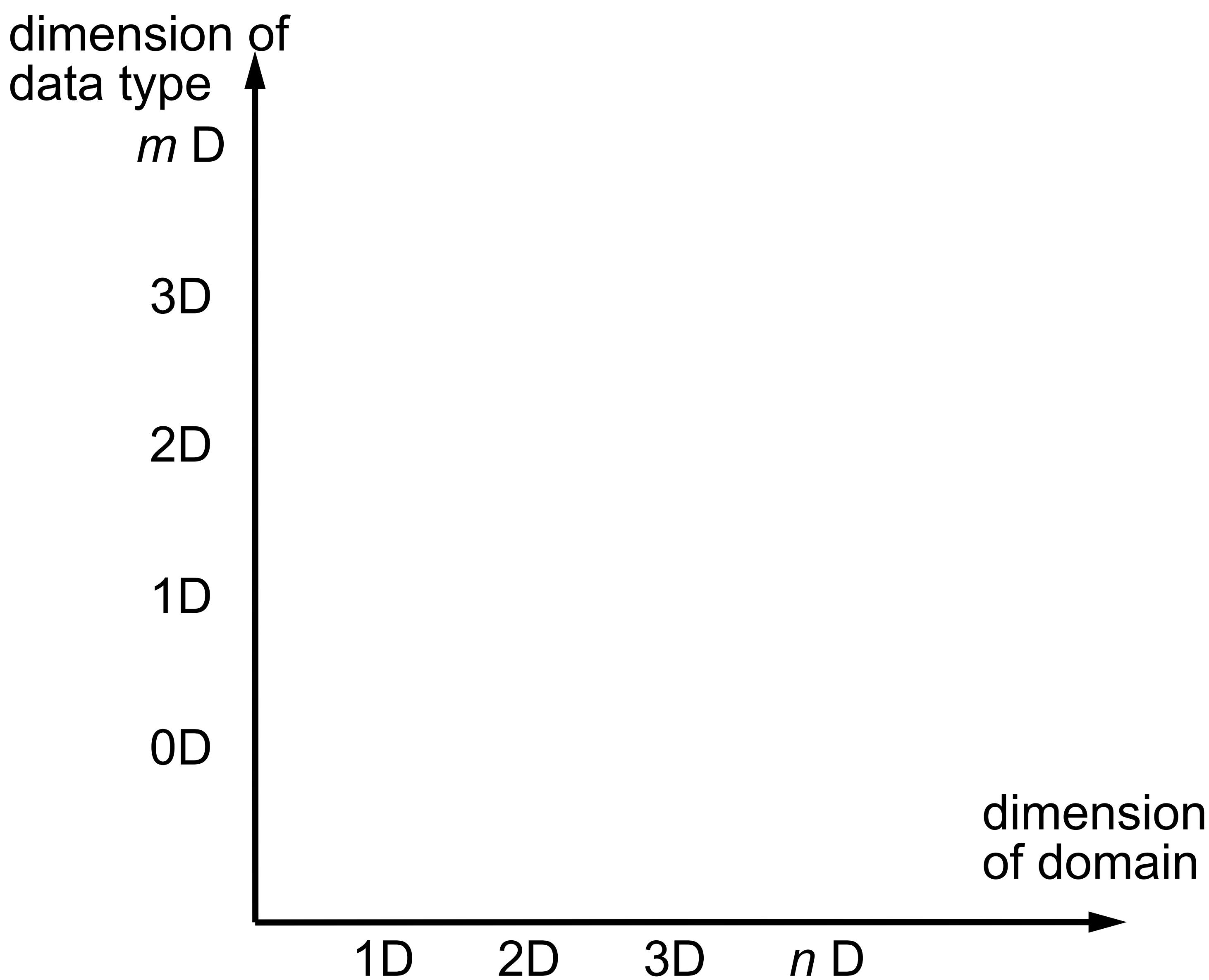
Multifields



8 molecular orbitals of a LiAlH2O DFT simulation, courtesy Aiichiro Nakano, University of Southern California

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

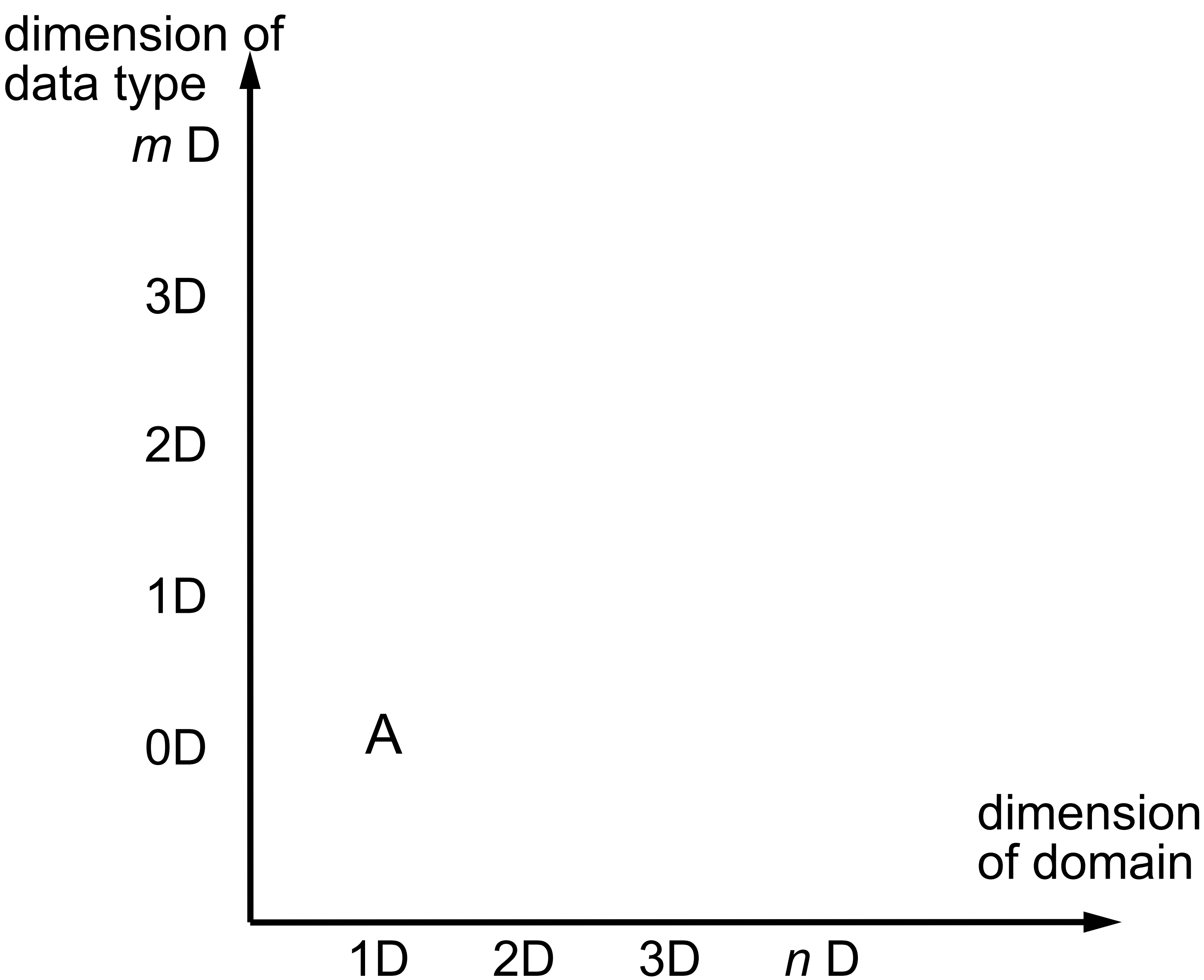


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

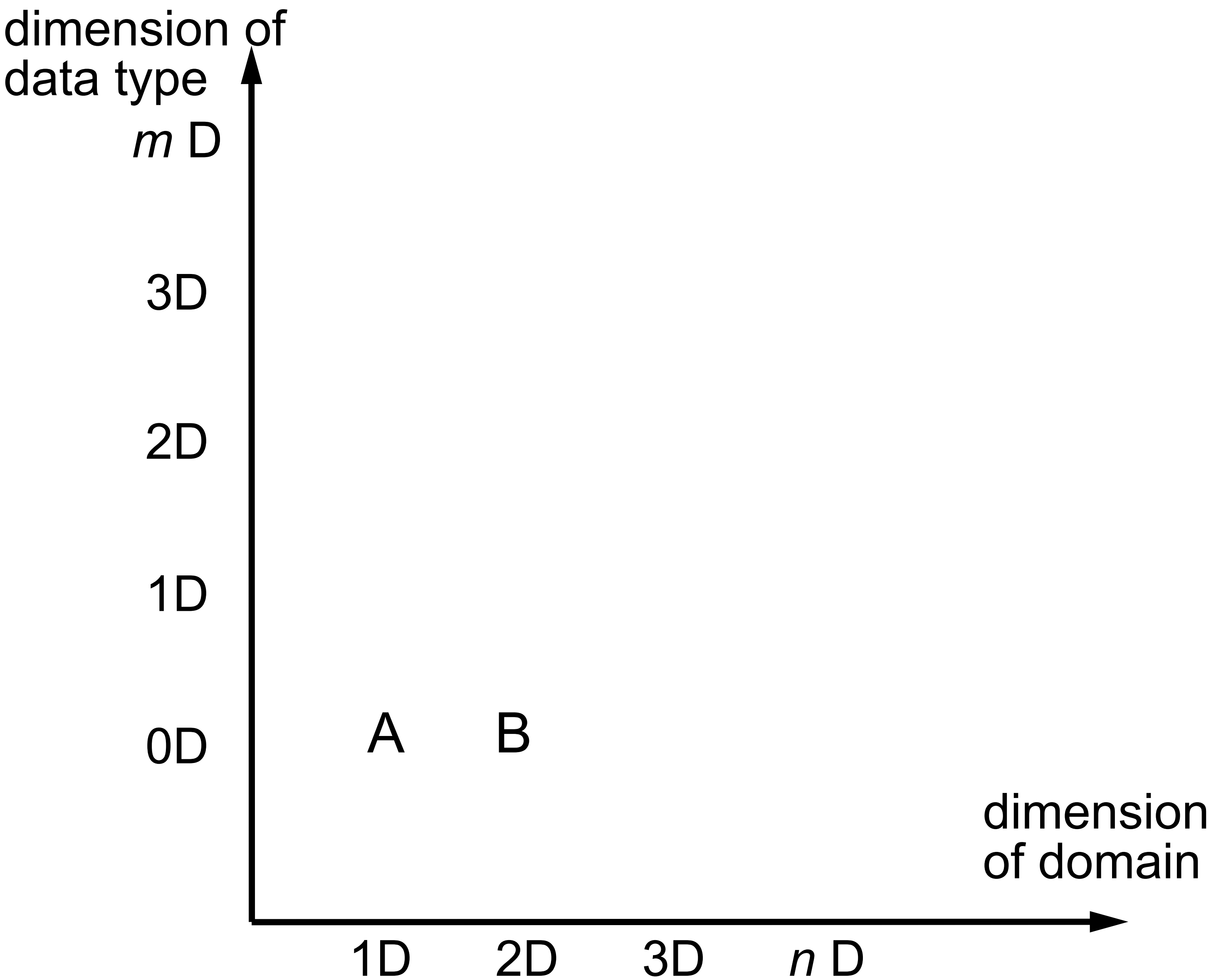


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

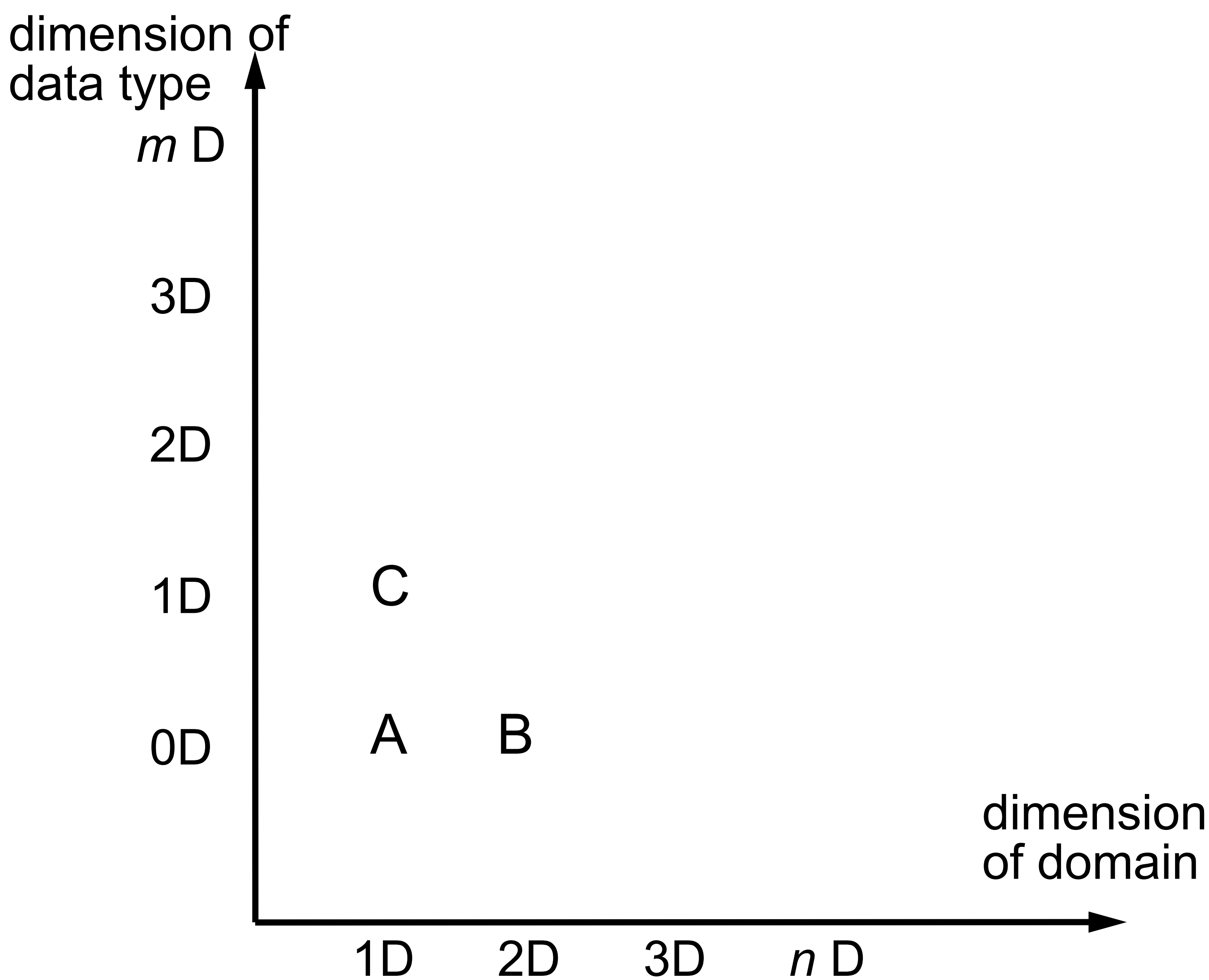


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

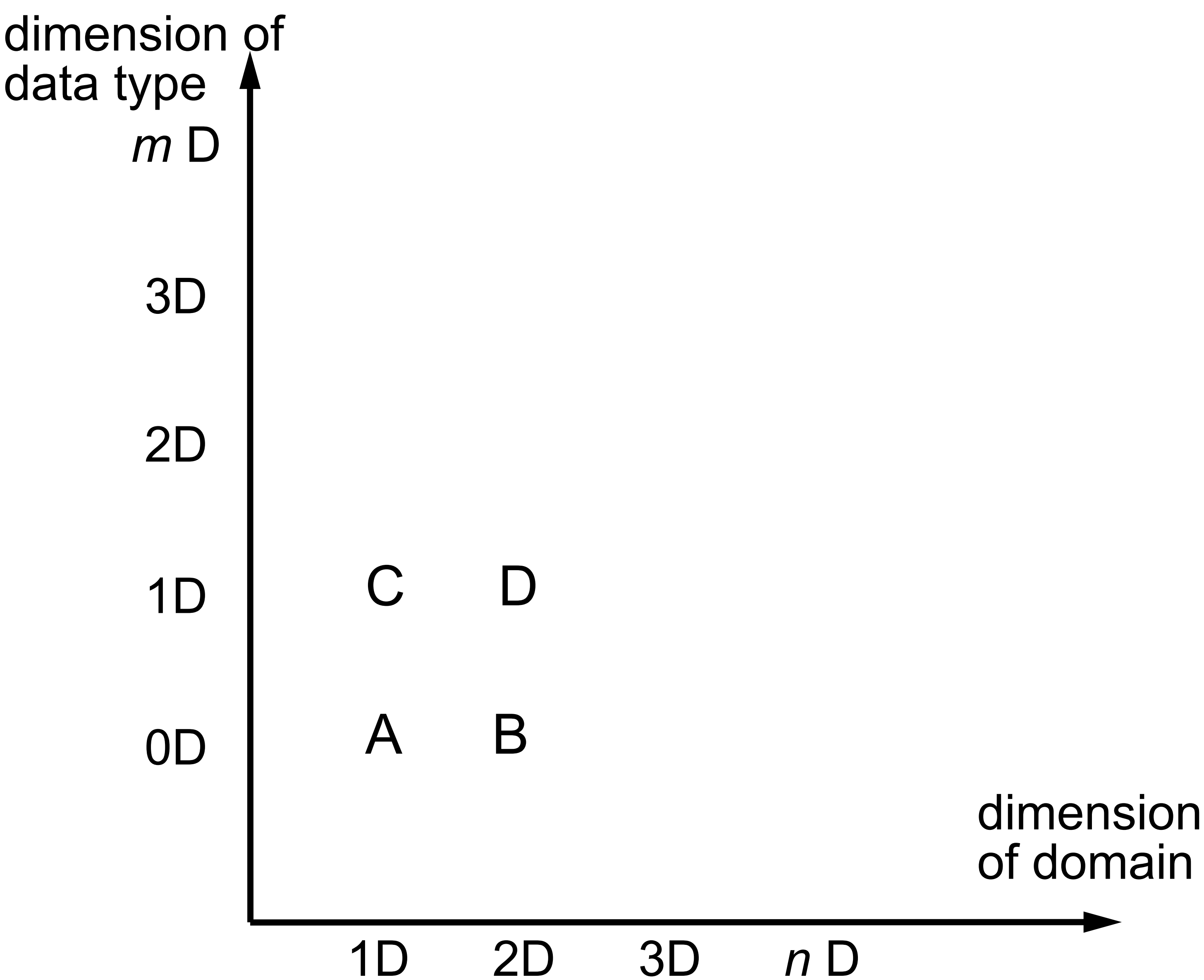


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

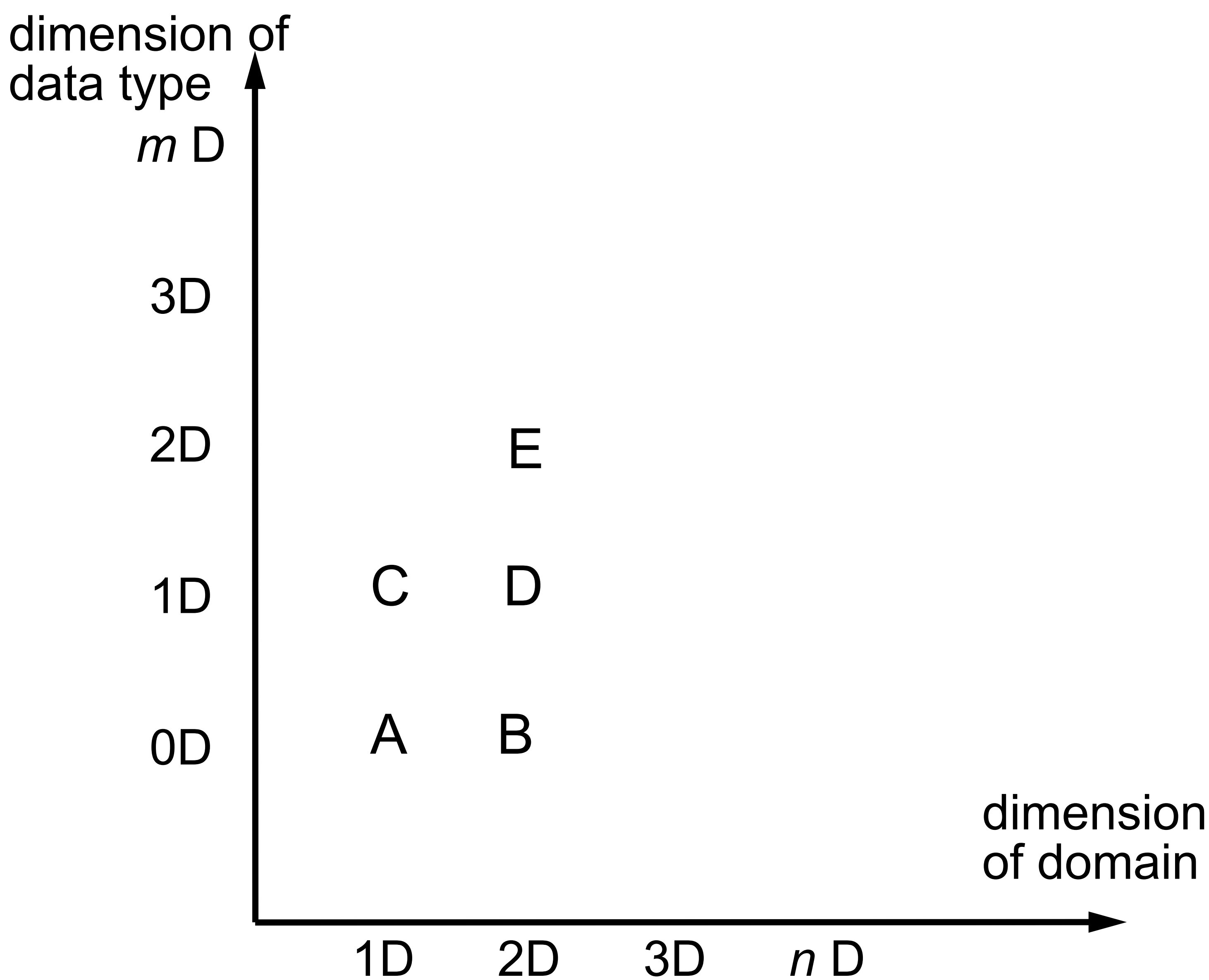


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

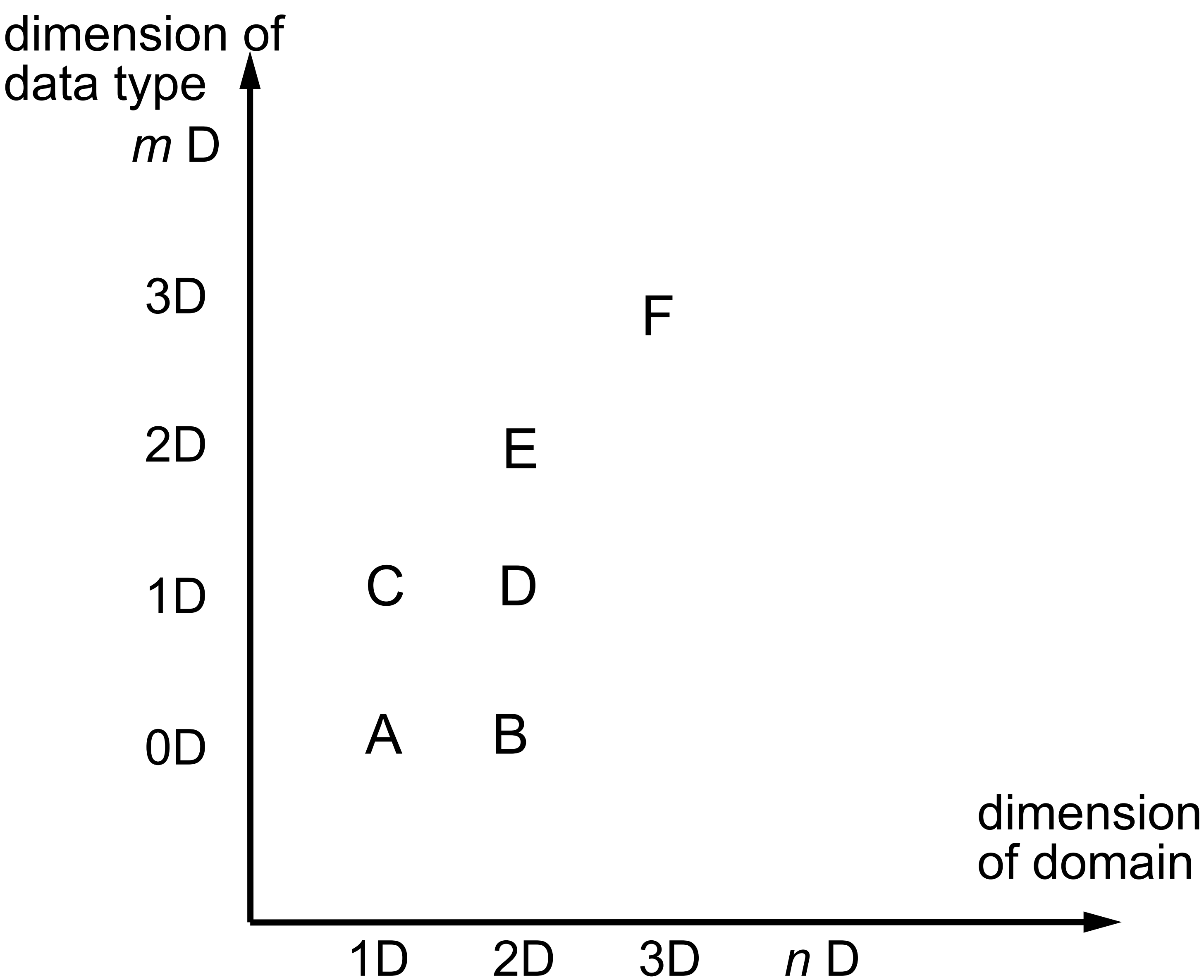


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

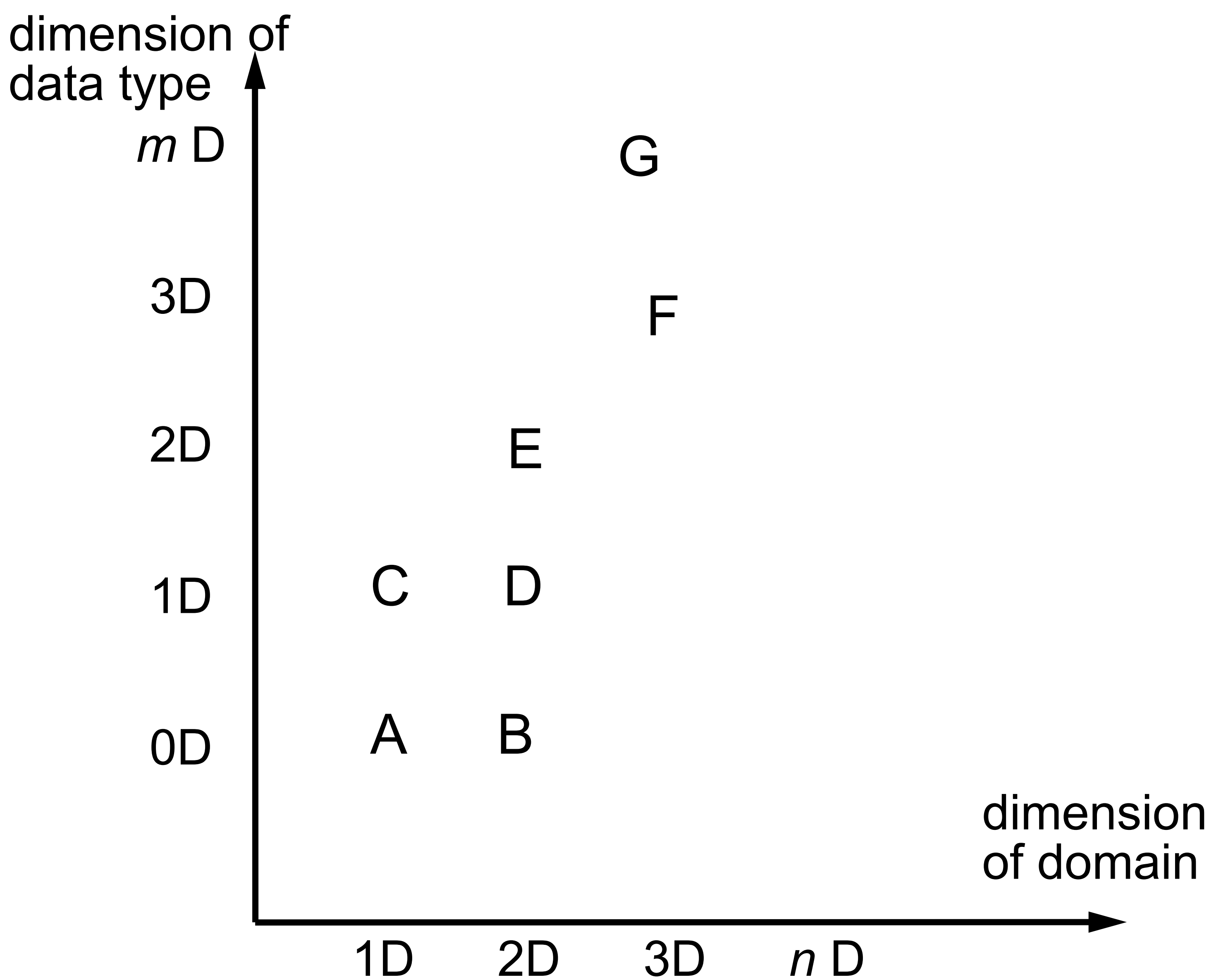


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

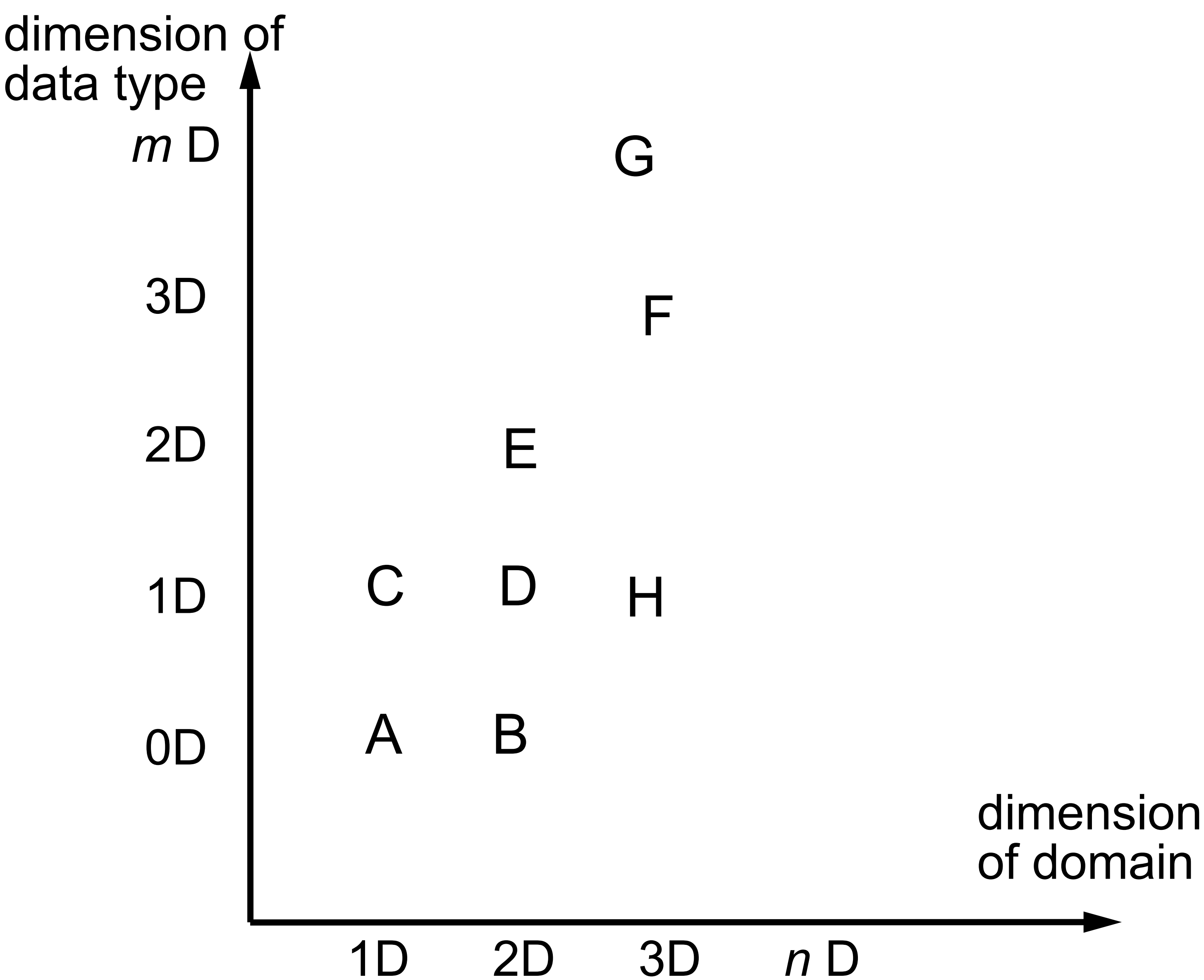


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

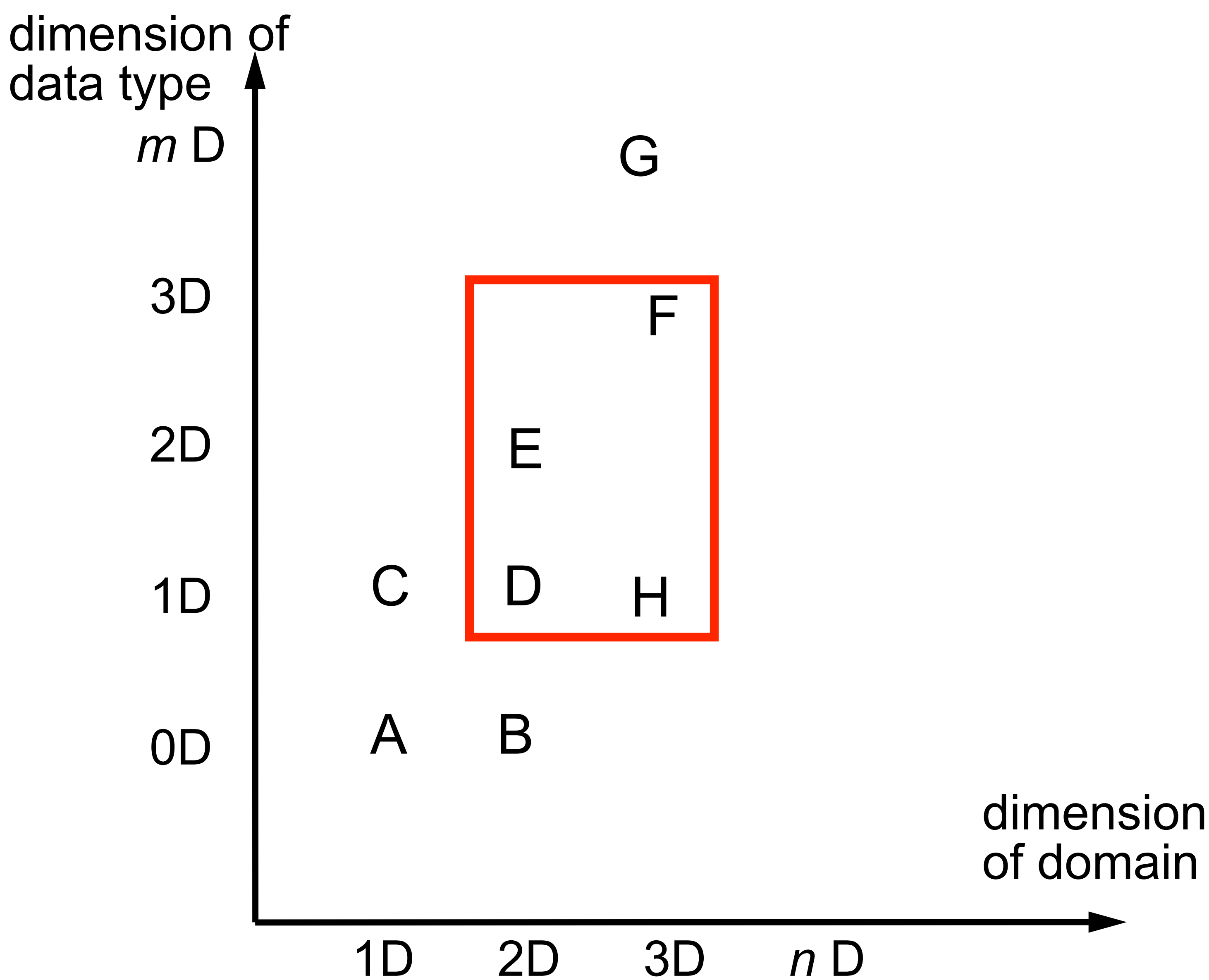


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

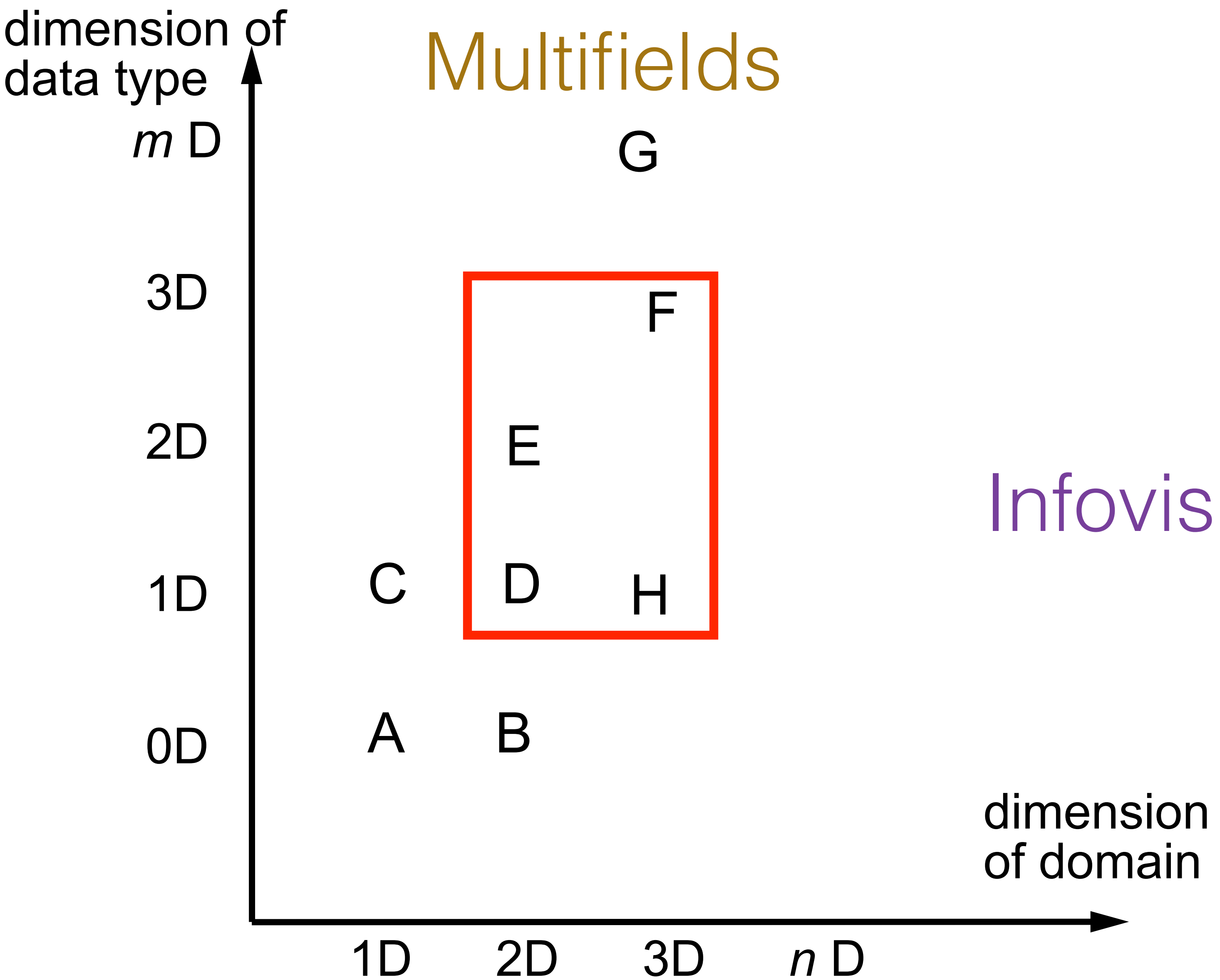


Examples:

A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

Types and Classification of Field Data

- dimension of domain (the field)
- dimension of the data to visualize (the geometry)

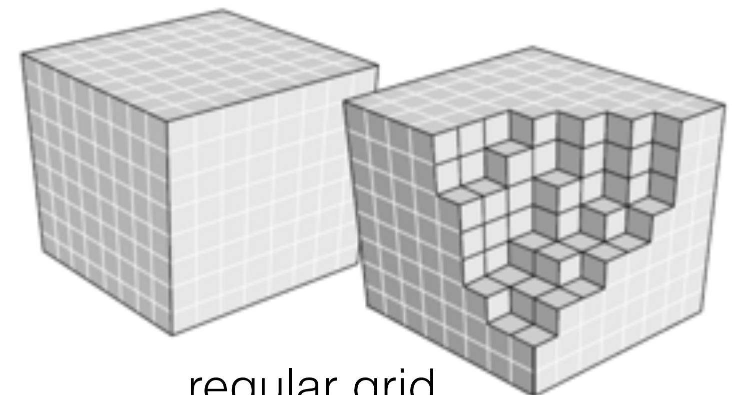


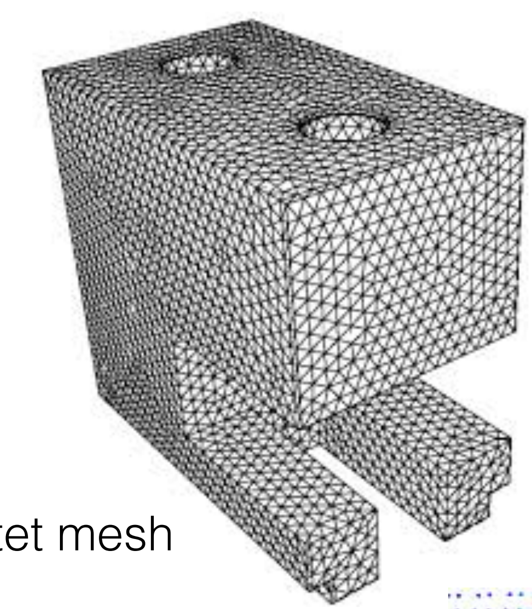
Examples:

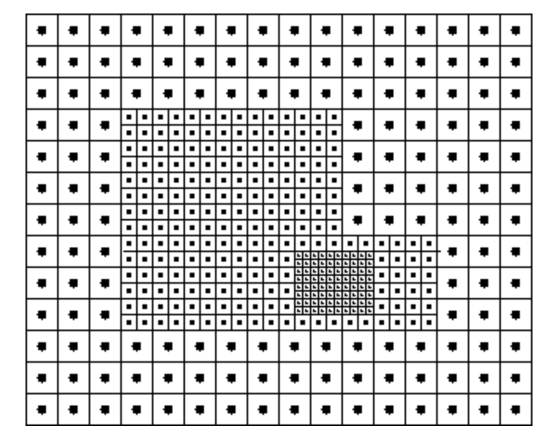
A: gas station along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a mechanical part
H: ozone concentration in the atmosphere

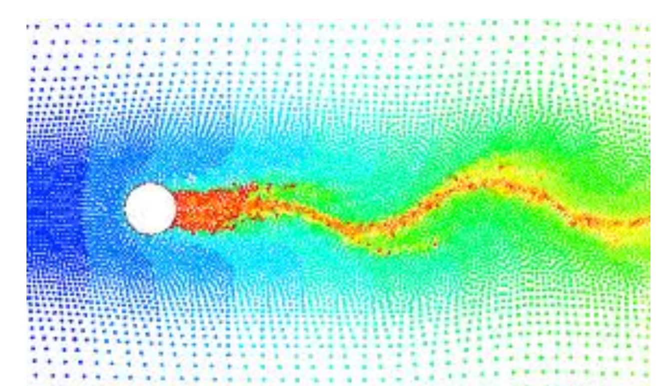
Grids

- Continuous fields are an illusion
- All data are discrete
- Meshes are chosen based on what is computationally efficient for the
- Visualization software must implement data models to handle a wide range of field and non-field data


regular grid


tet mesh

AMR



particles

Structured vs Unstructured

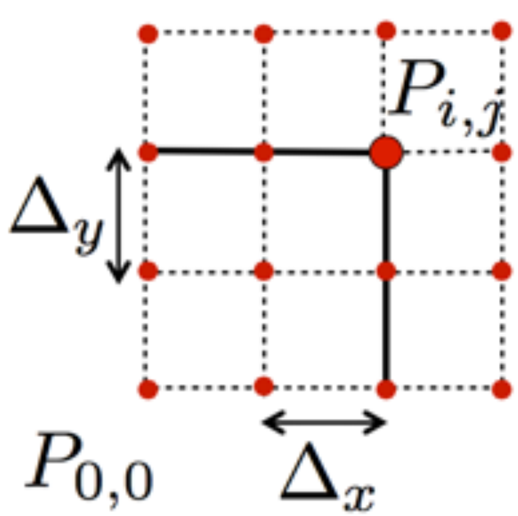
- In general, from the relational database world:
  - **Structured data** are data that are indexed, and can be accessed via a hash, array, or other query.
    - I.e., search time *O(1)* or *O(log N)*.
  - **Unstructured data** are not indexed — you have to brute-force search to find them.
    - I.e. search time *O(N)*
- In information/data visualization:
  - **structured** means data you've already indexed, organized (for example, in D3).
  - **unstructured** is everything else (i.e. text, imagery, video, foo) you have to search through.
- In scientific visualization, this can get a bit confusing…
- First we need to differentiate between geometry and topology.

Geometry vs Topology

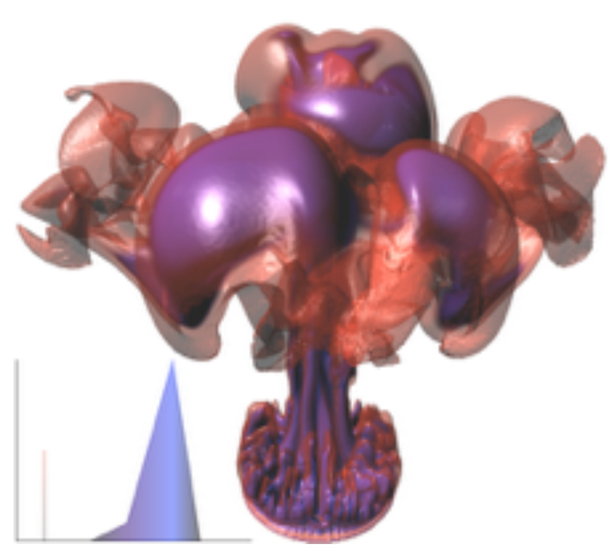
- Geometry
  - Position of vertices in Euclidean space
  - Can be uniform, structured or unstructured.
- Topology
  - Defines the “cells”, or connectivity of the vertices.
  - Can also be structured or unstructured.

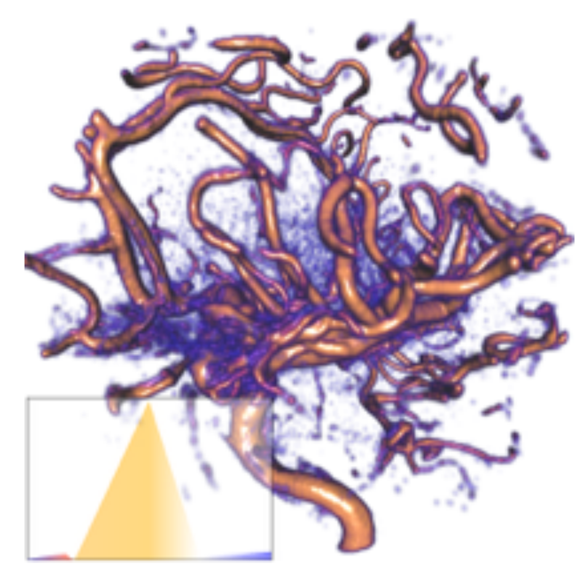
Uniform grid geometry

- Uniform spacing along the axes, also known as “raster data”.
- Most volume data look like this; structured data usually means this.
- You still need metadata to know the size of the axes!



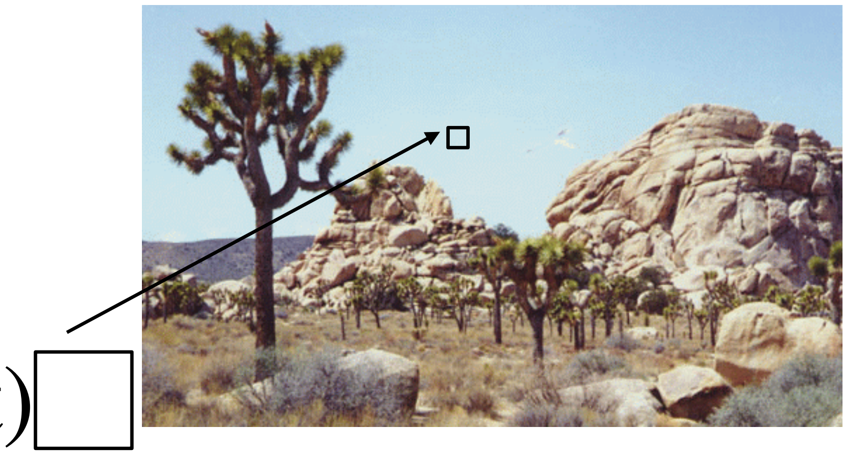
$$P_{i,j,k} = P_{0,0} + i\Delta_x \vec{e}_x + j\Delta_y \vec{e}_y$$



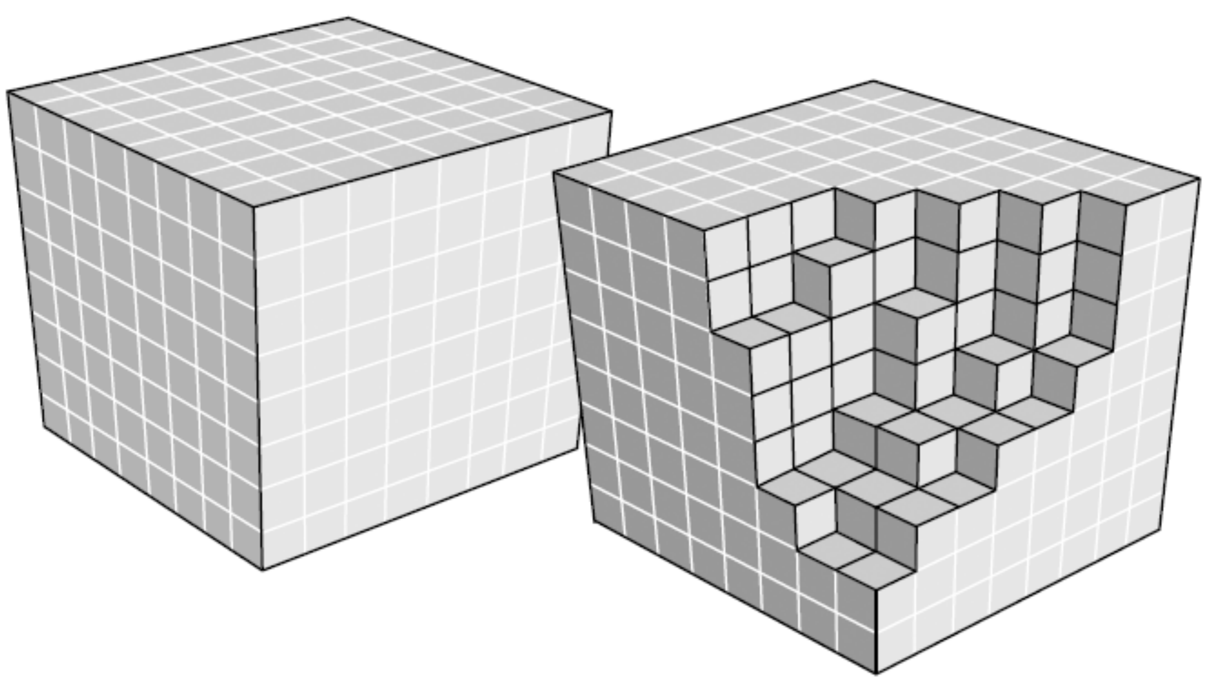


www.sci.utah.edu

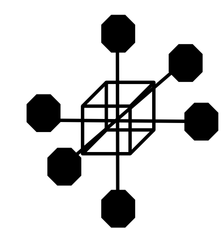
- Representation of scalar 3D data set $\Omega \in R^3 \rightarrow R$
- Analogy: pixel (picture element)

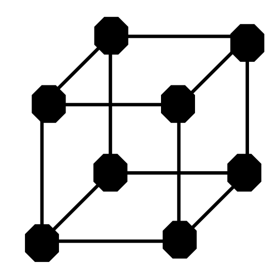


- Voxel (volume element), with two interpretations:
  - Values between grid points are resampled by interpolation



- Collection of voxels
- Uniform grid

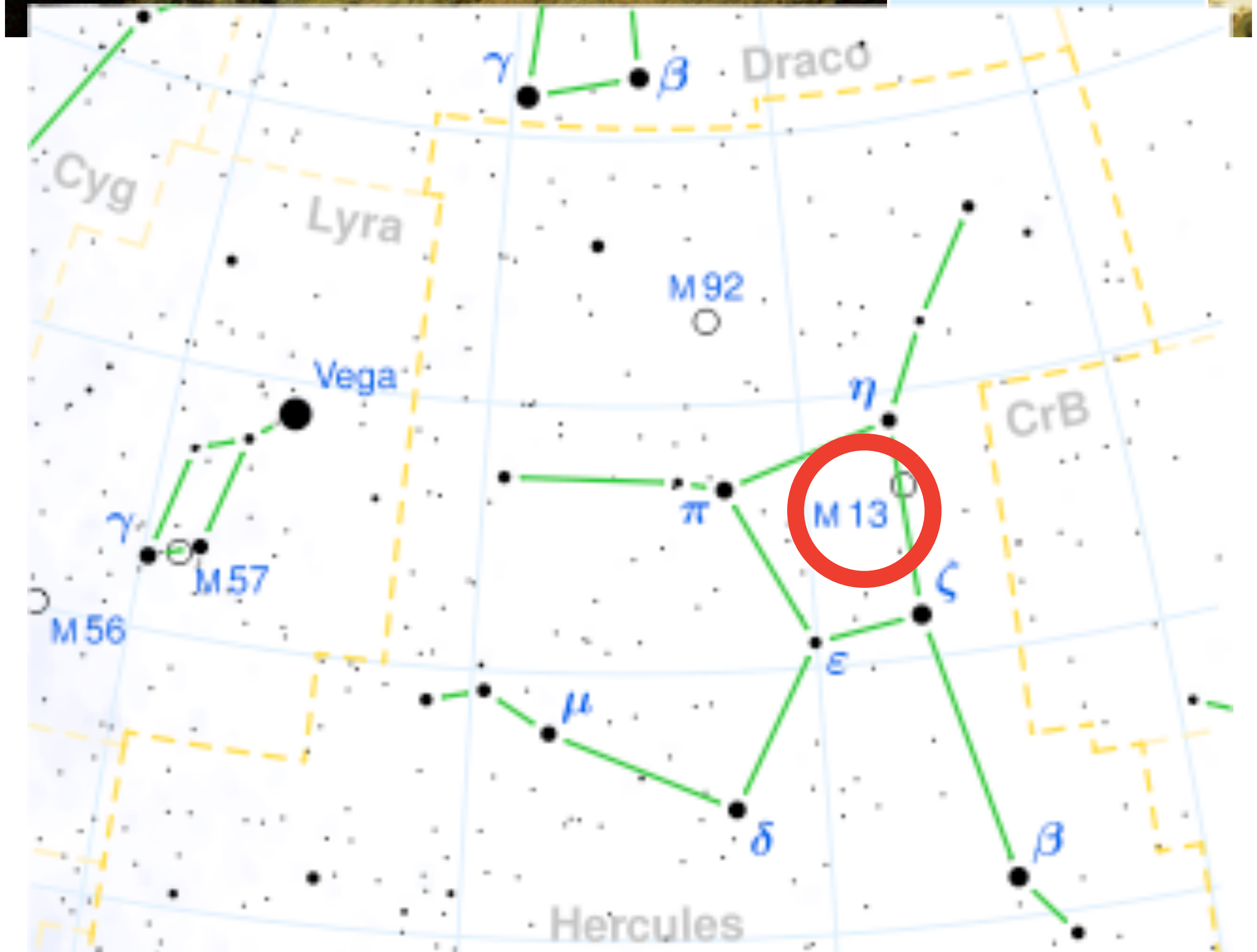




Arecibo Message

http://en.wikipedia.org/wiki/Arecibo_message

- Way of understanding mechanics of raster image representation
- Radio telecope in Puerto Rico
- built in 1964, renovated in 1974
- To celebrate: Frank Drake and Carl Sagan (Cornell University) sent message to M13 in Hercules (25,000 light years away)
- 1679 bits, frequency modulate 2380 MHz



The Message

http://www.physics.utah.edu/~cassiday/p1080/lec06.html

1679 bits were encoded as 2380MHz plus and minus some frequency

0 0 0 0 0 0 1 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 1 0 1 1 0 0 1 0 1 0 1
0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 1 1 1 0 0 0 1 1 0 0 0 0 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 1 1 0 0 1 0 0 0 0 1 1 0 1 0 0 0 1 1 0 0 0 1 1 0 0 0 0 1 1 0 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 1 1 0 0 0 1 1 0
0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 1 0 0 0 0 1 1 0 0 0 1 1 1 0 0 1 1 0 1 0 1 1 1 1 1 1 0 1 1 1 1 1 1 0 1 1 1 1 1 1 0 1 1 1 1 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0
0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 1 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0
0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0
0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 1 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 1 1 1 1 0 1 0 0 1 0 1 1 0 1 1 0 0 0 0 0 0 0 1 0 0 1 1 1 0 0 1 0 0 1 1 1 1 1 1 1 0
1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 1 1 0 1 1 0 0 1 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 1 0 0
0 0 0 0 1 0 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0
0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 1 0 1 0 0 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0
1 1 1 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 0 0 0 0 0 0 1 1 0 0 1 1 0 0 0 0 0 0 0 0 1 1 0
0 1 1 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 1 0 1 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 0 0 1 1 1 1 1 1 0 1 0 0 1 1 1 1 1 0 0 0

This is a **1-D** sequence of bits in time

How will an alien understand this list of bits?

(will have different symbols than “0” “1”)

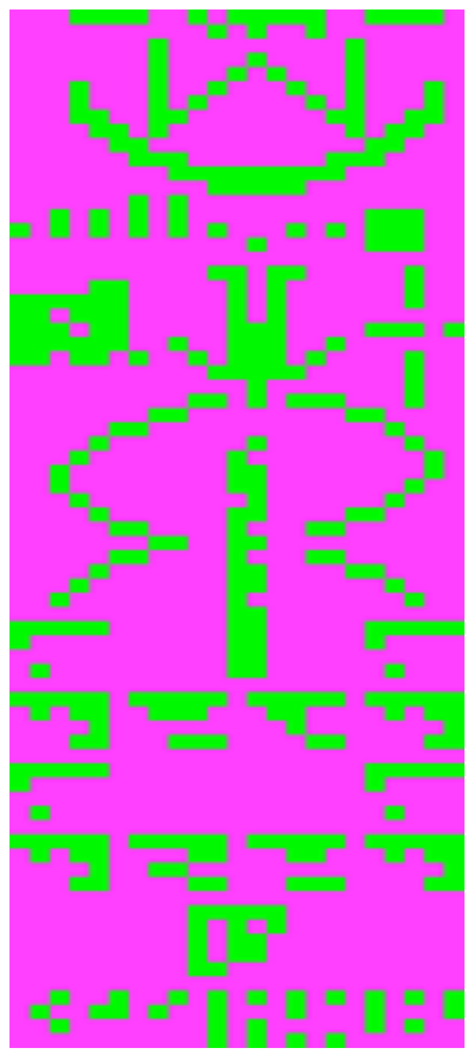
No meta-information!

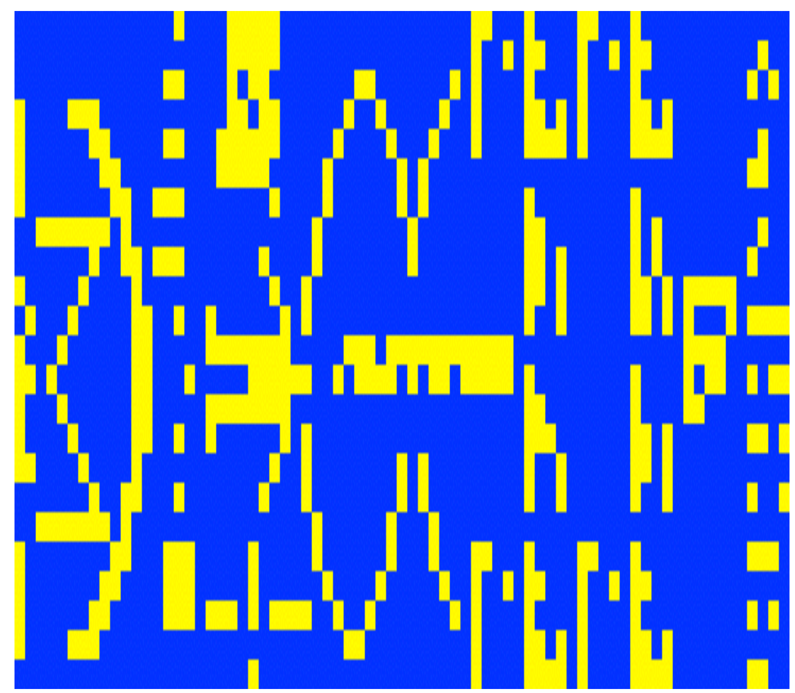
Understanding the message

- Perhaps some “visual” representation of bits

  - (what is black vs white?)
- Aliens notice 1679 = 23 x 73 (product of two primes)
- Perhaps its not a linear sequence: 2-D array
- Two ways of sequencing values in 2D array
- Various ways of laying them out in 2D space
- Then: have to decipher it!

73 x 23



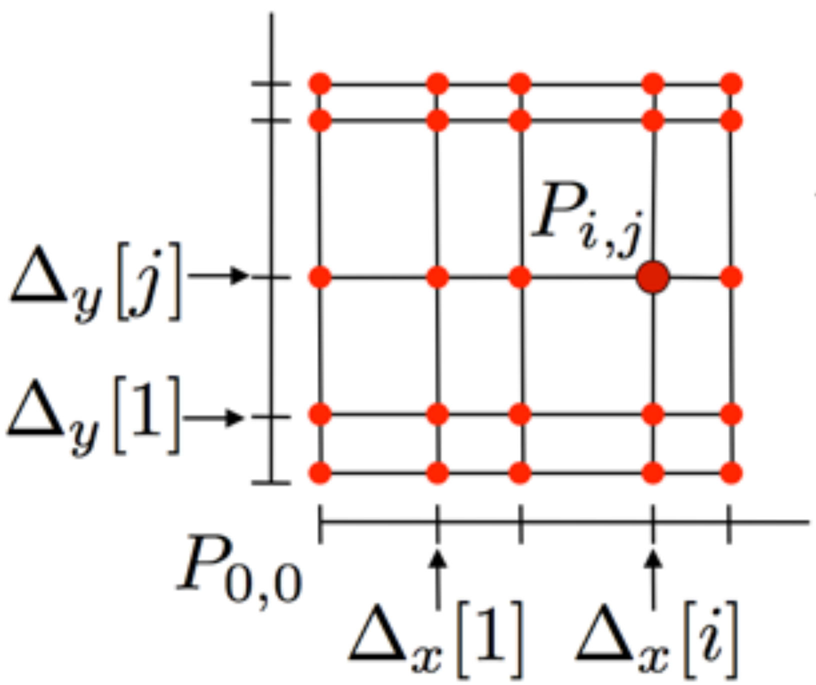


compare to:
http://en.wikipedia.org/wiki/Arecibo_message

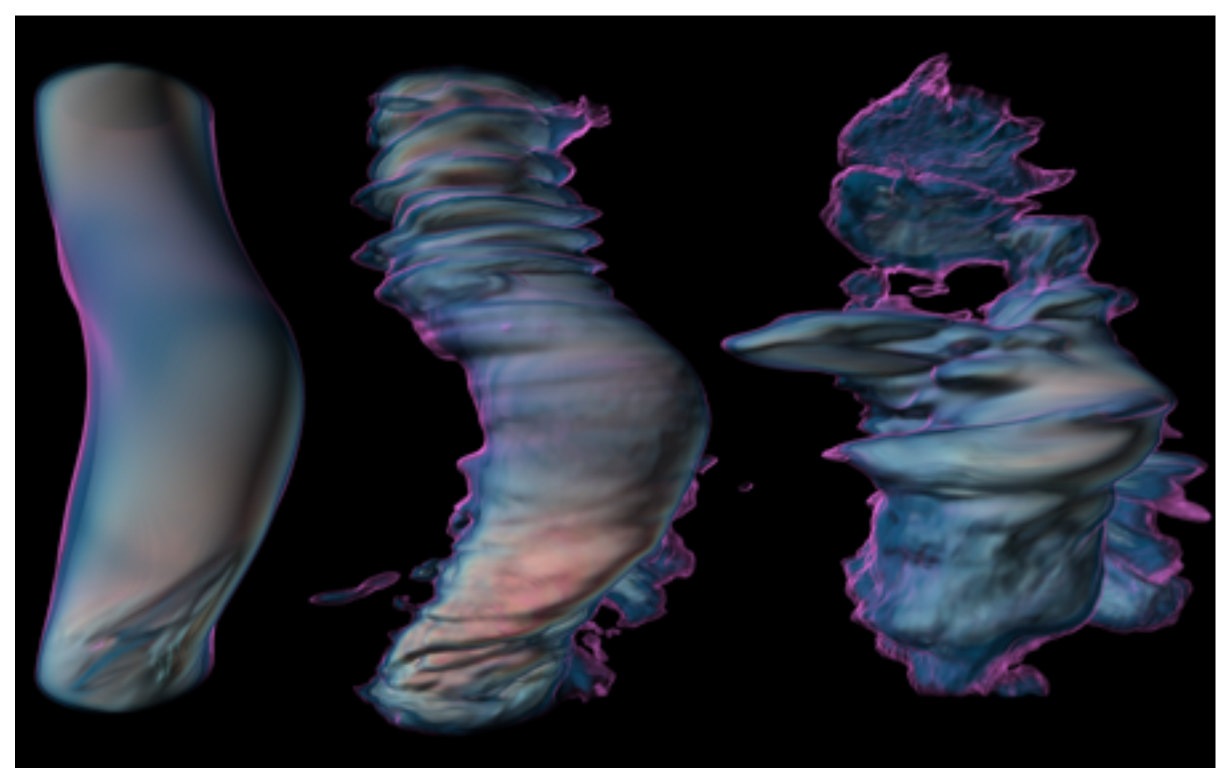
23 x 73: what was different?

Structured (rectilinear) grid geometry

- Still structured, but with non-uniform spacing along the axes.
- Positions can still be computed procedurally
- Some meteorology, climate CFD data like this.



$$P_{i,j,k} = P_{0,0} + \Delta_x[i]\vec{e}_x + \Delta_y[j]\vec{e}_y$$

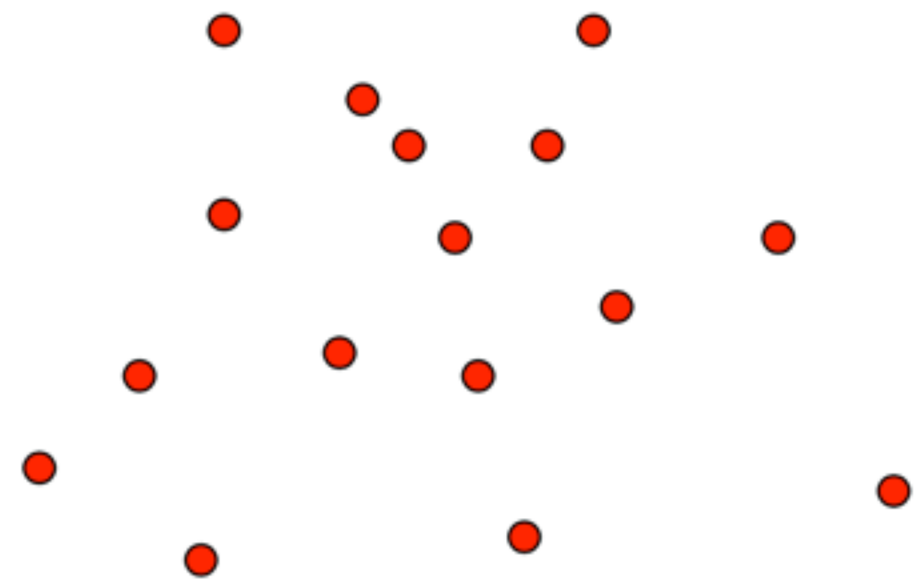


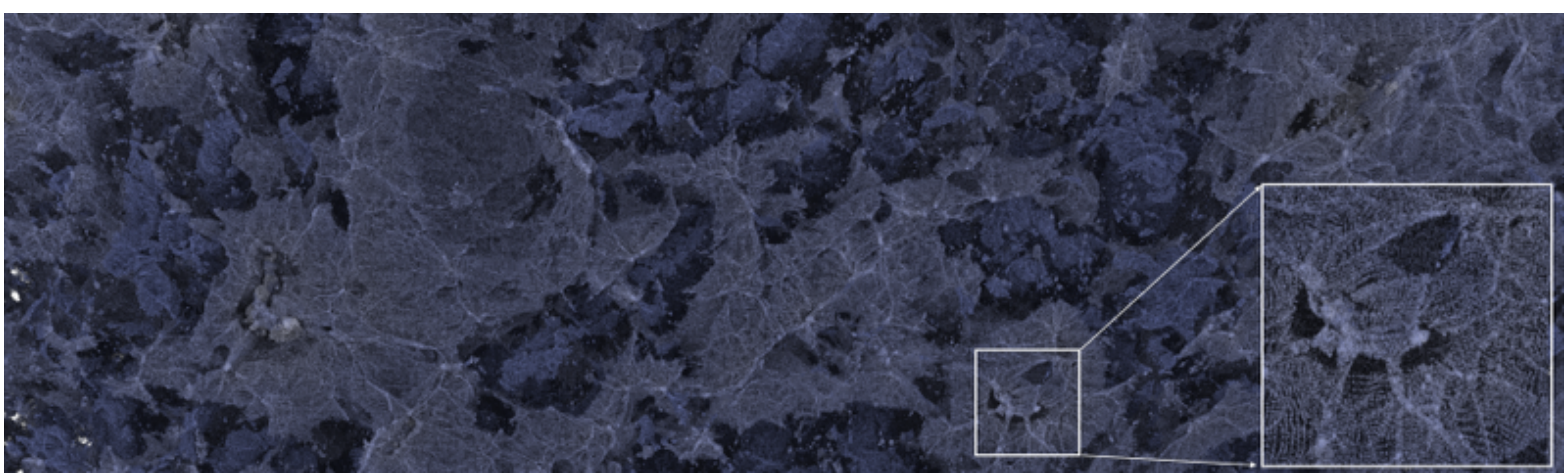
Turbulence in the Ionosphere - Greg Foss, TACC

www.sci.utah.edu

Unstructured geometry

- Raw, unstructured point data.
- You actually need to store the x,y,z positions of vertices.
- Some of the largest computational and scanned data
  - LiDAR, RGB-D point clouds, range scans
  - n-body codes — molecular dynamics, cosmology
- Note: this is unstructured topology, too!

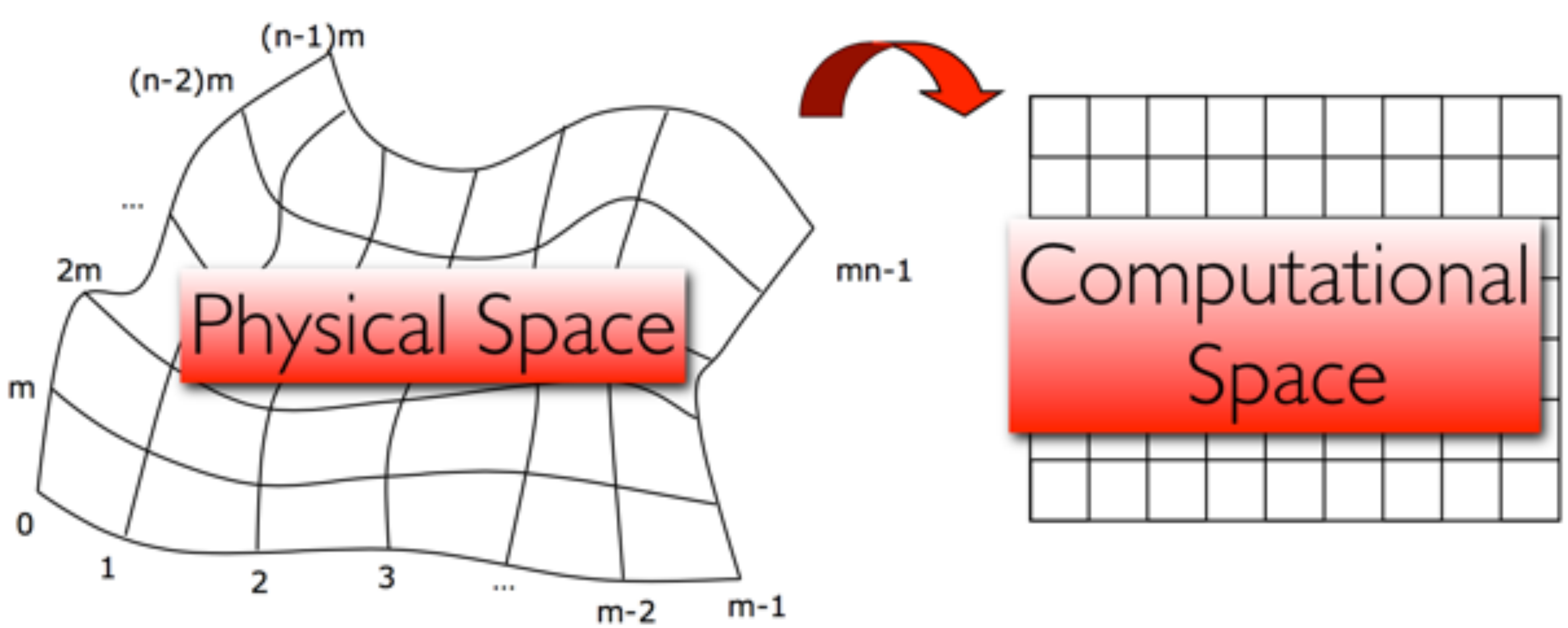


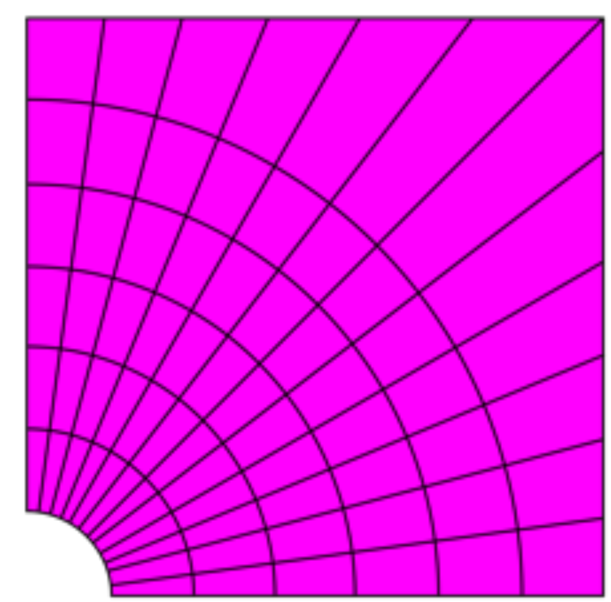


30-billion particle “Cosmic Web” data — Paul Shapiro, University of Texas at Austin
I Wald, A Knoll, G.P. Johnson, W. Usher, V. Pasccci & M.E. Papka: “CPU Ray Tracing Large Particle Data with P-k-d Trees” IEEE Visualization 2015

Structured grid topology

- You can have unstructured geometry but structured topology
  - Implicit definition of cells
  - Implicit connectivity between vertices
- More exotic options with structured grid topology:
  - Finite elements, finite differences on curvilinear grids
  - spectral F/E, some spline-based finite elements simulations
  - Good for precision-critical flow computations (blood flow, CFD)





Curvilinear grids

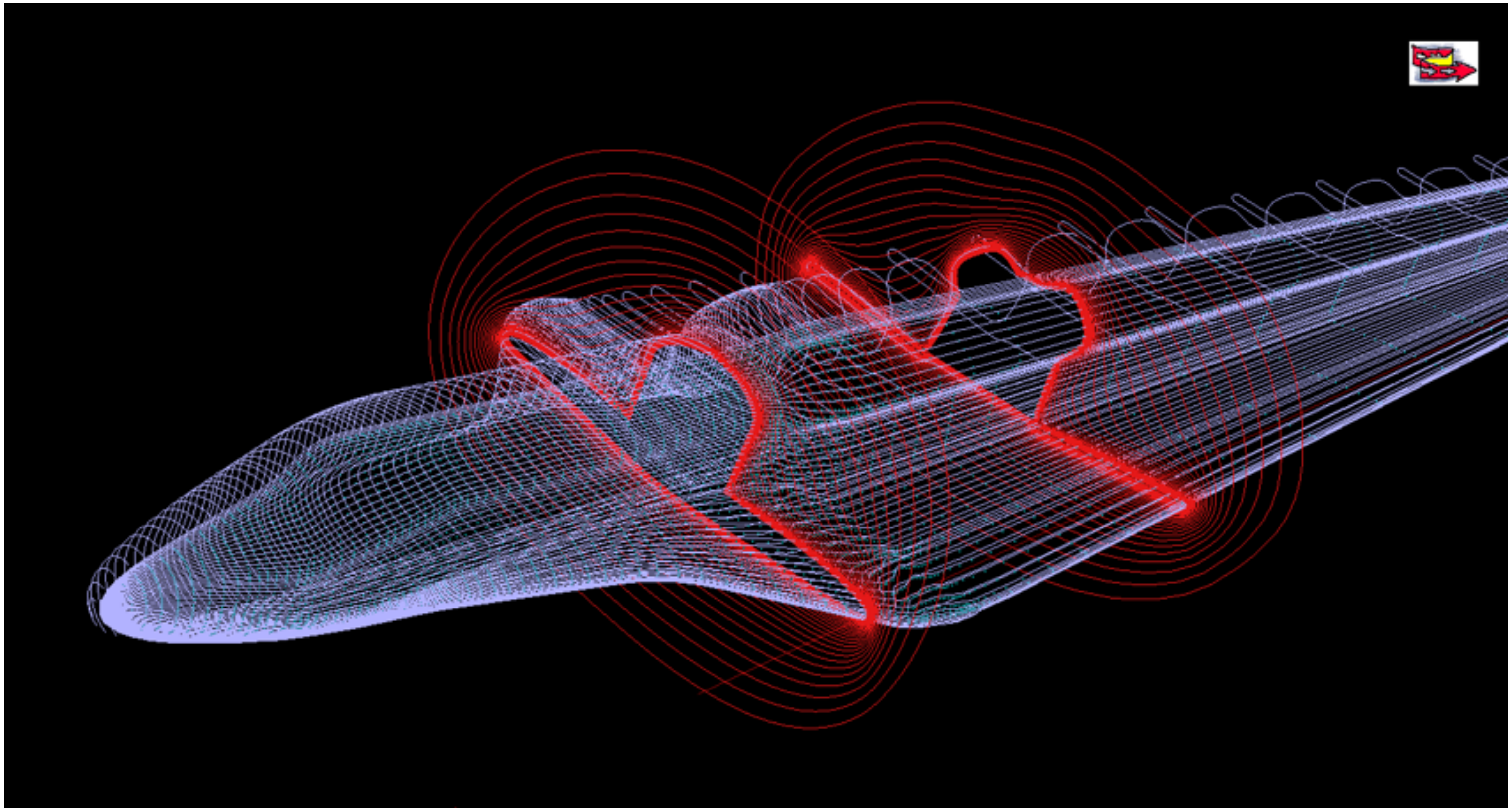
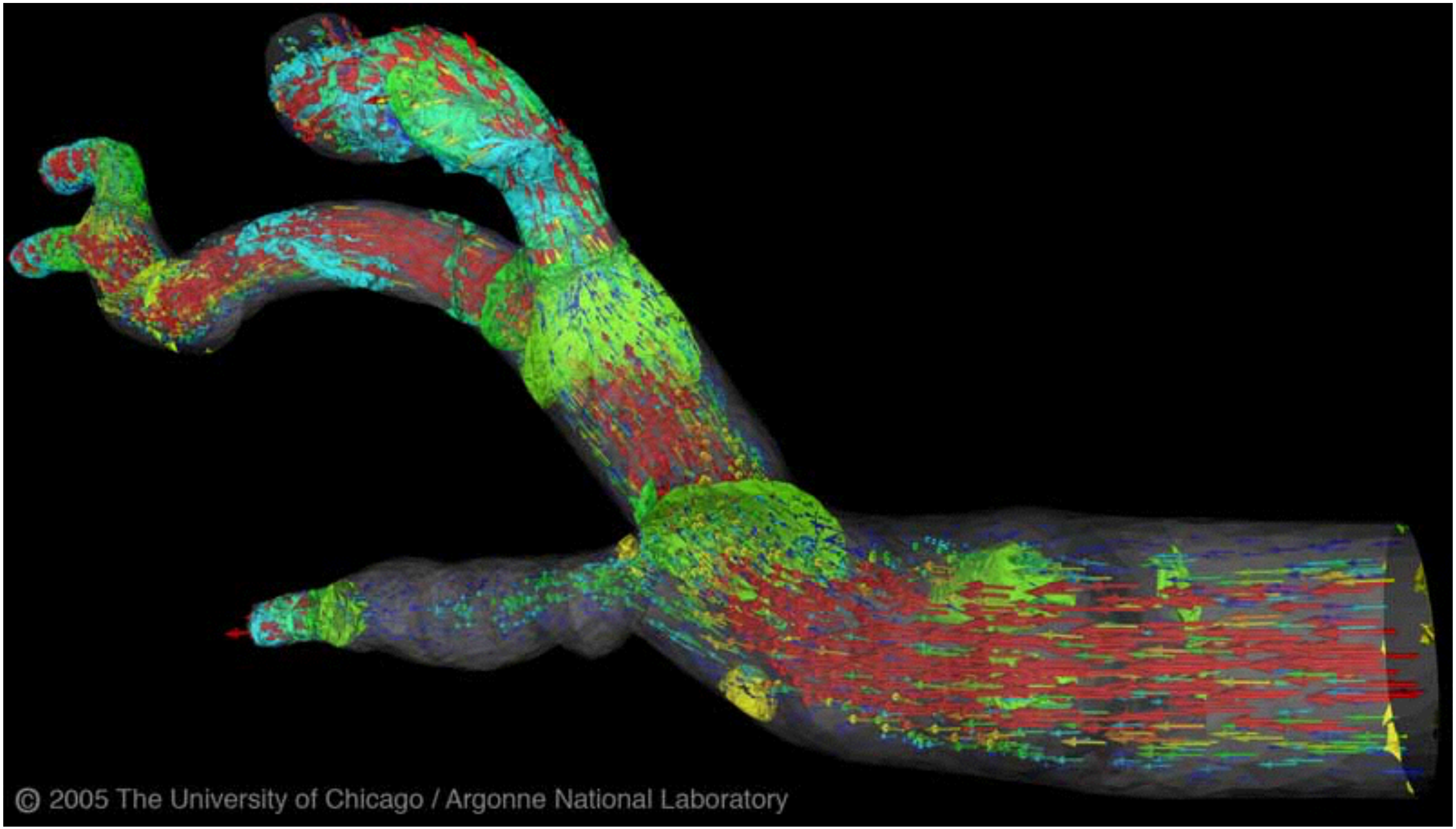


Image:T.U. Graz

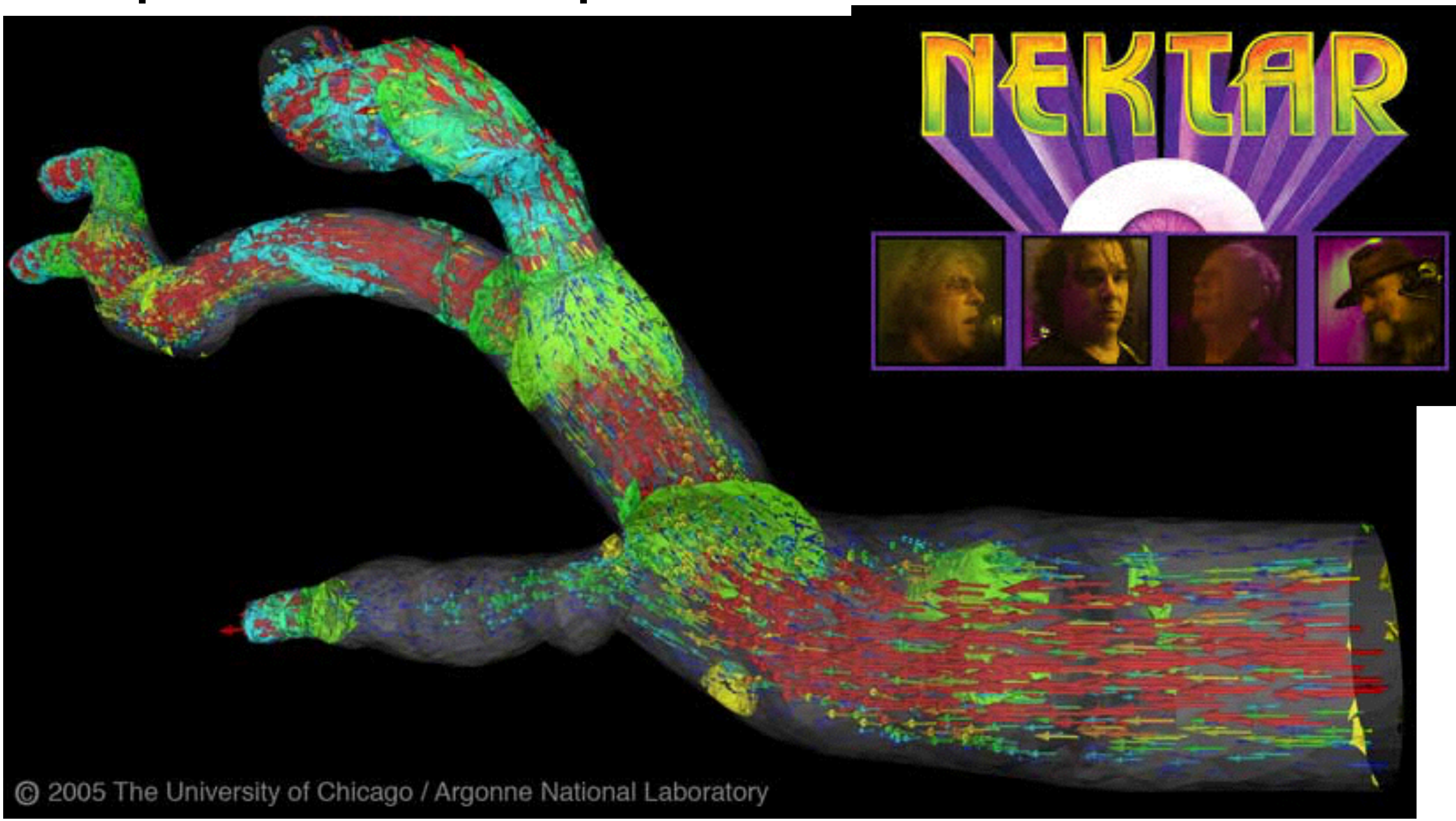
Spectral/hp finite elements



Data: George Karniadakis, Brown University. Visualization: Joe Insley, ANL

www.sci.utah.edu

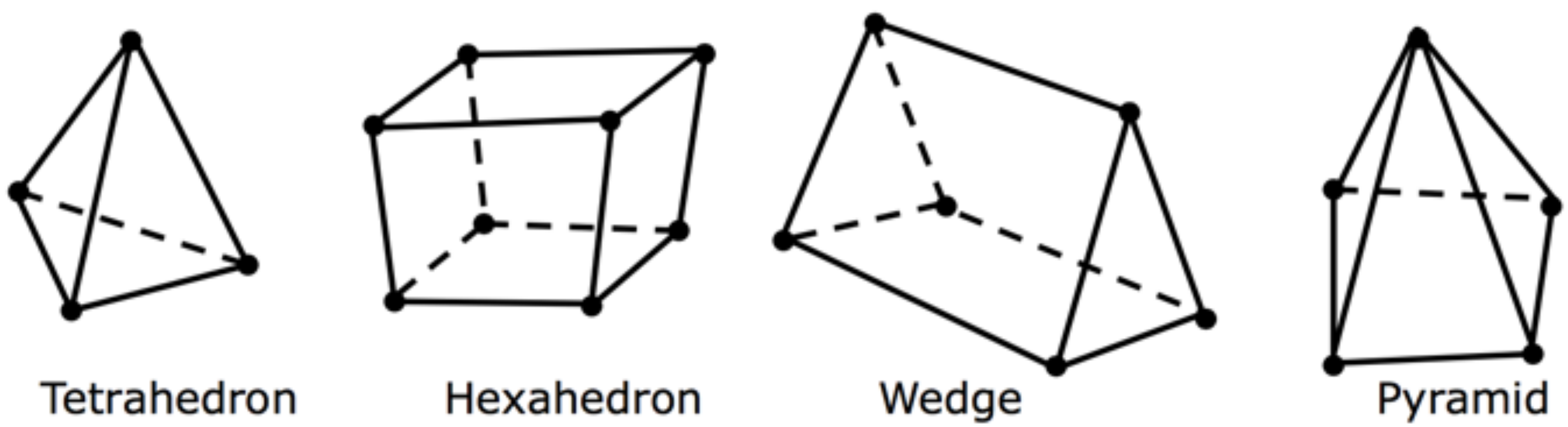
Spectral/hp finite elements

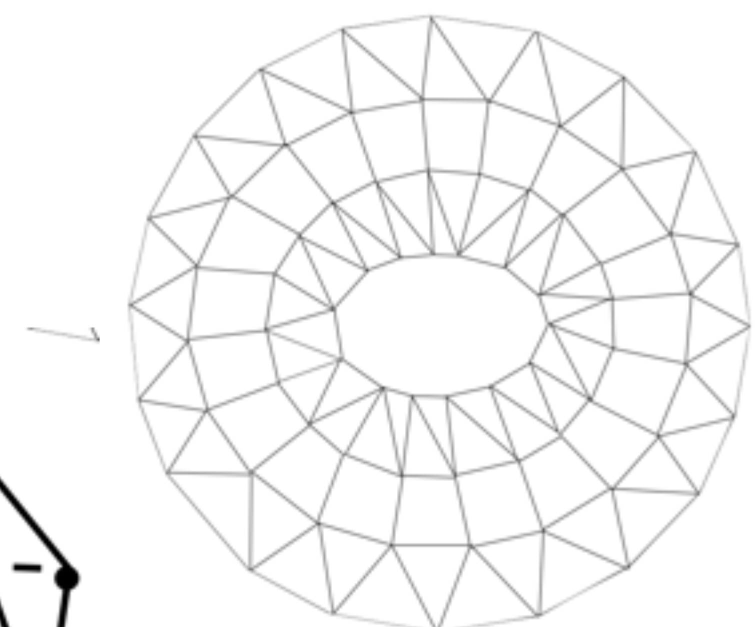


Data: George Karniadakis, Brown University. Visualization: Joe Insley, ANL

Unstructured grid topology

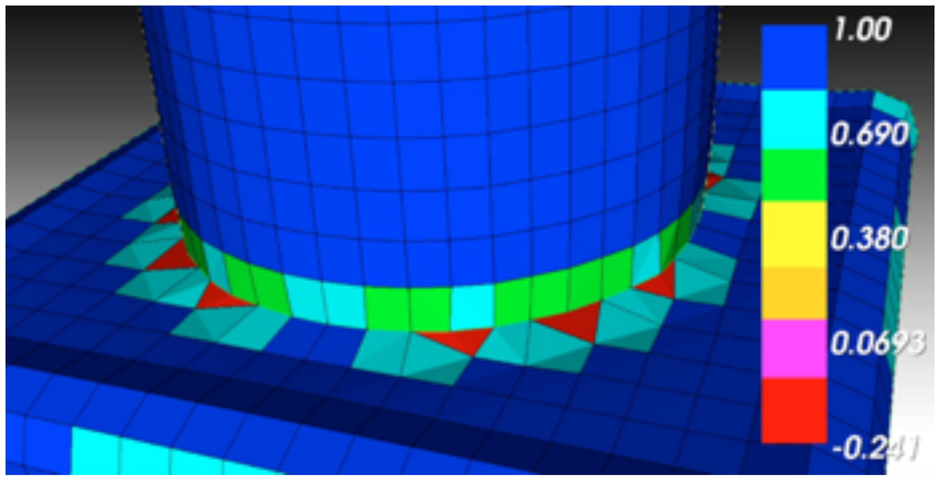
- Both uniform elements and "mixed elements" (allowing any cell type)
- Need to store vertices and indices separately
  - mixed elements: vertices, indices and count
- Many, many finite elements codes.
  - solid mechanics, CAD
  - bioelectric modeling



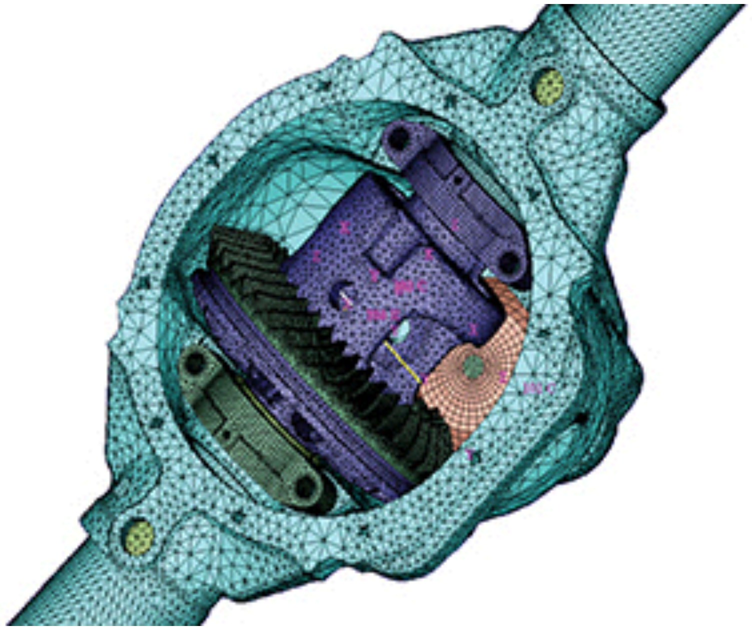


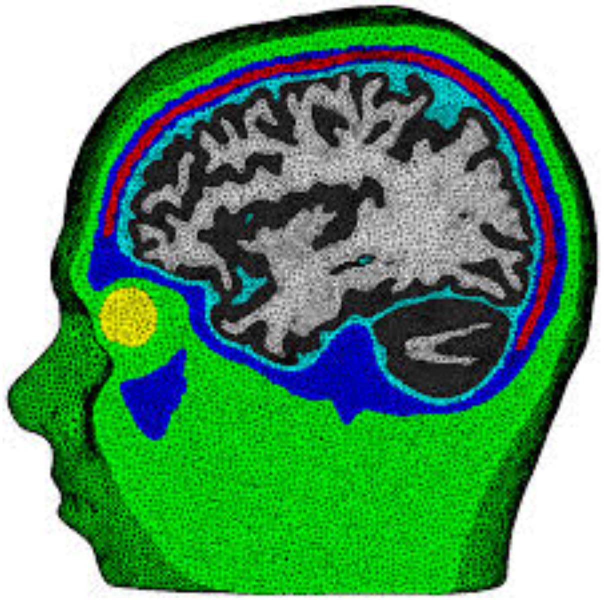
www.sci.utah.edu

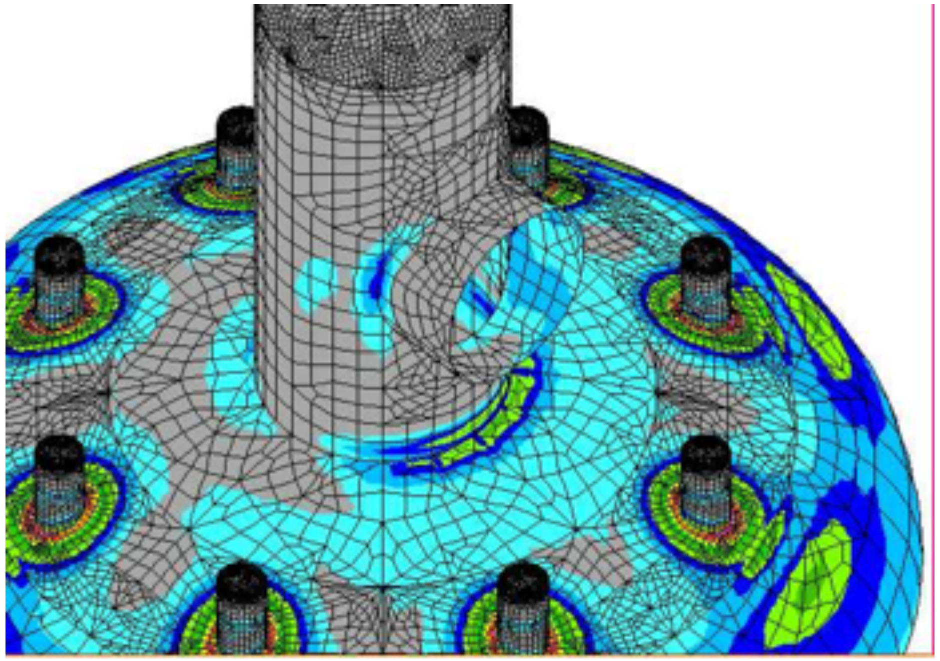
Finite elements

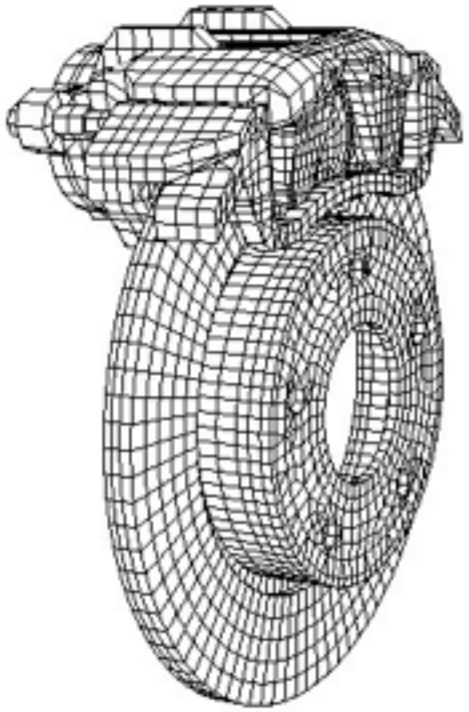


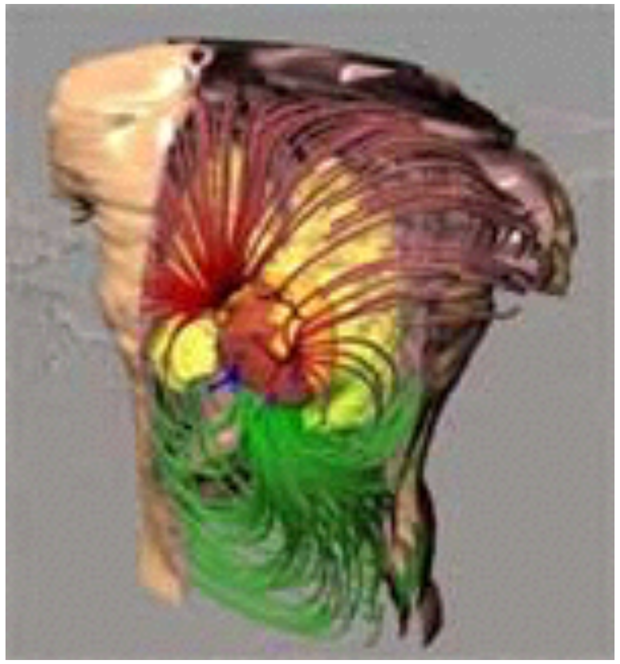
Visualization: Steve Owen, using Cubit











www.sci.utah.edu

Geometry vs topology

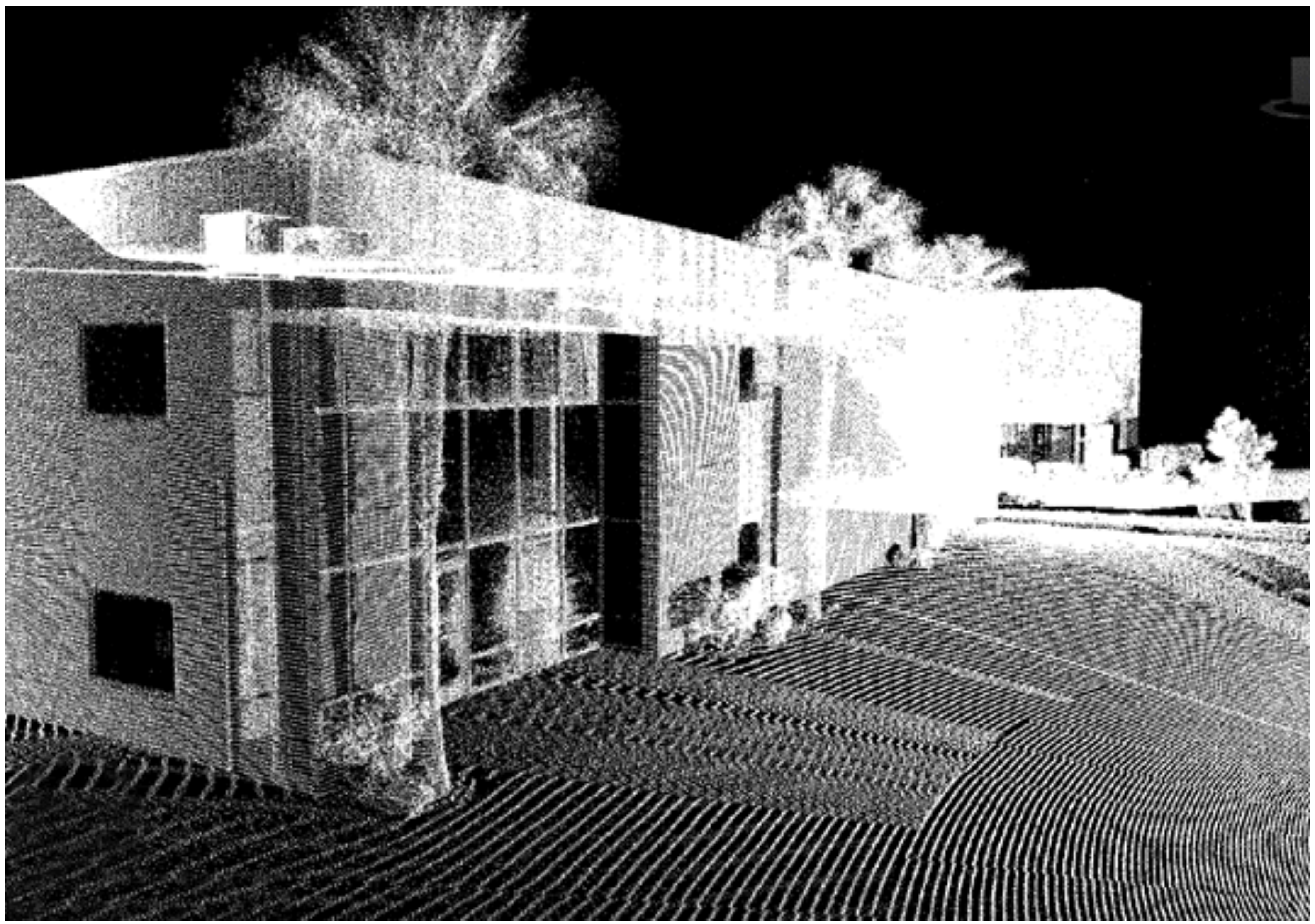
Geometry ↕ / Topology ↔	Structured	Unstructured
Uniform	**Image**	**Unstructured**
Structured	**Rectilinear**	**Unstructured**
Unstructured	**Curvilinear**	**Unstructured**

sci www.sci.utah.edu

Colloquially

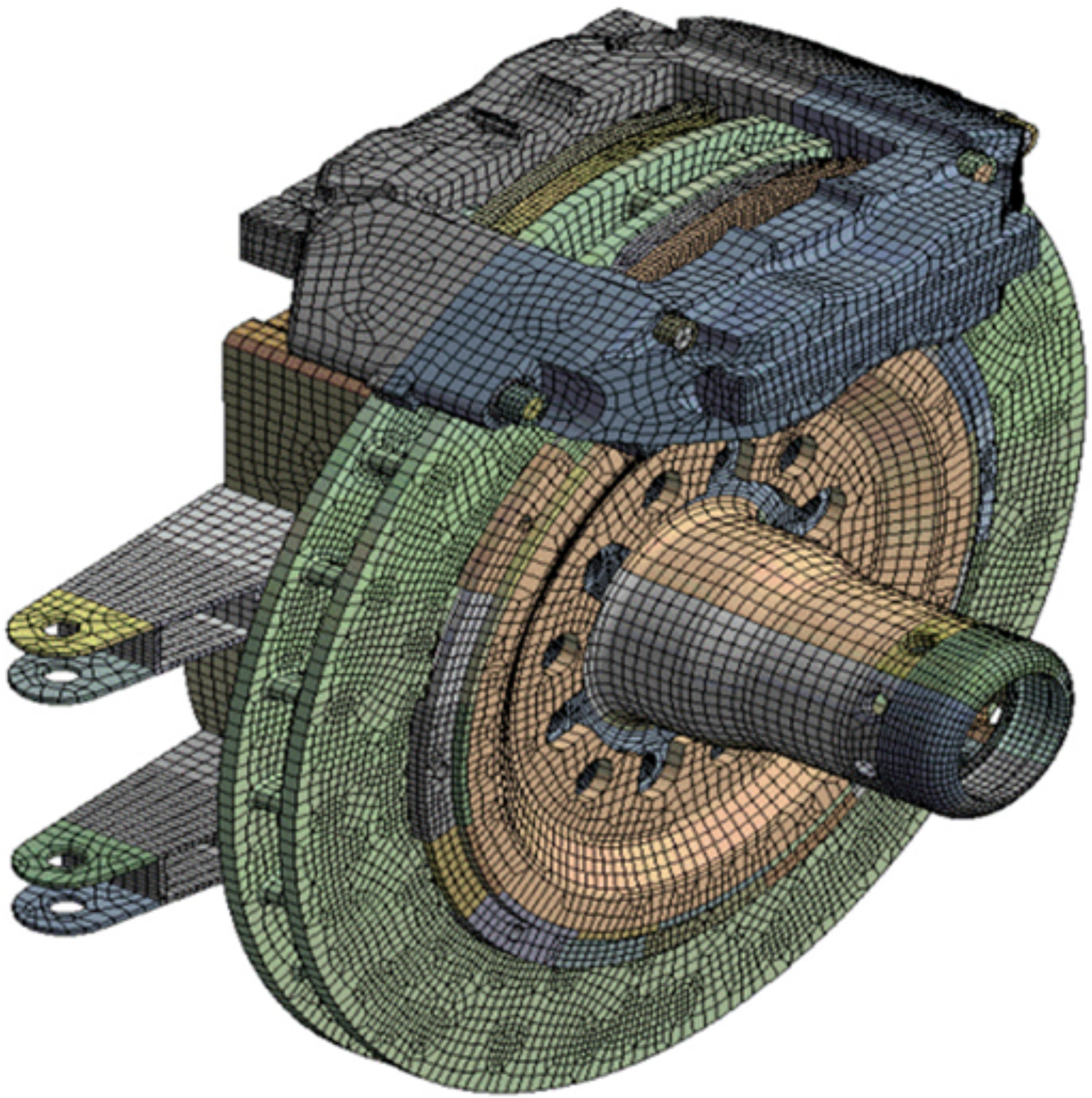
- In spatial (scientific) vis we usually talk about **geometry**, and:
  - **structured** means rectilinear grid (usually, but not always uniform).
  - **unstructured** means everything else (curvilinear grids, tetrahedra, hexahedra, points, etc.)

Is it (geometrically) structured or unstructured?



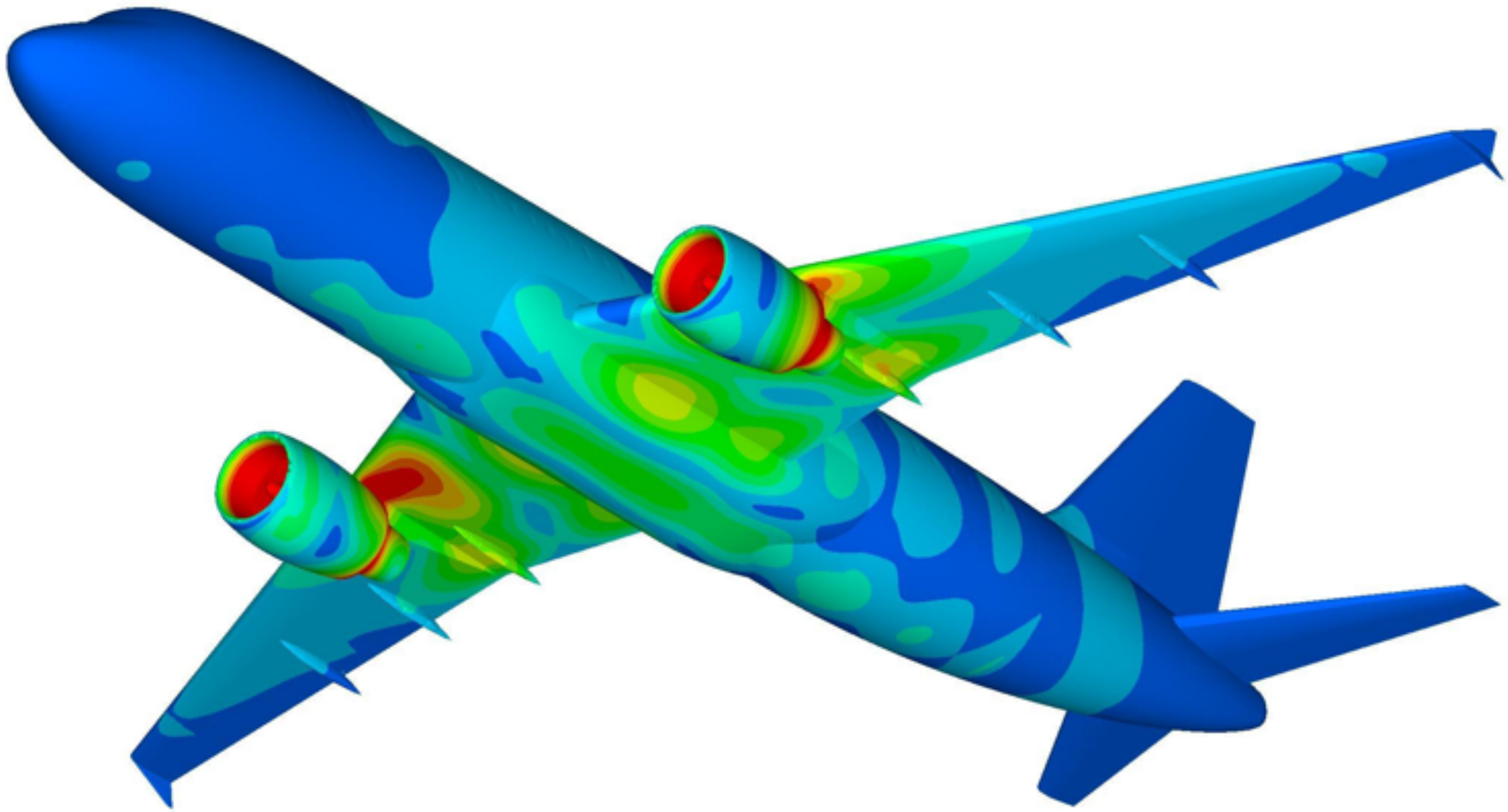
www.sci.utah.edu

Is it (geometrically) structured or unstructured?



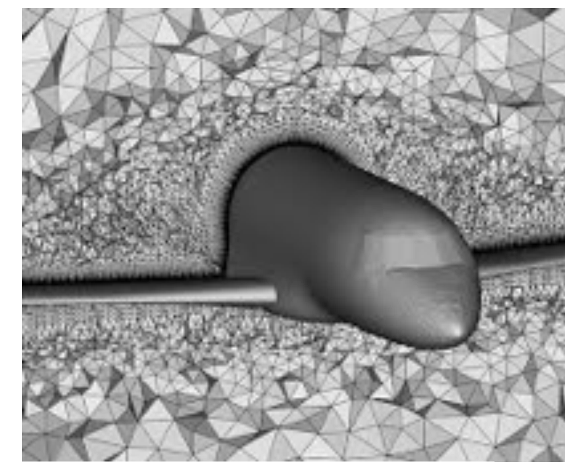
Is it (geometrically) structured or unstructured?

Is it (geometrically) structured or unstructured?

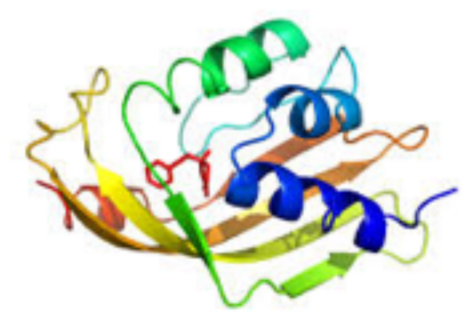


Non-field and other data

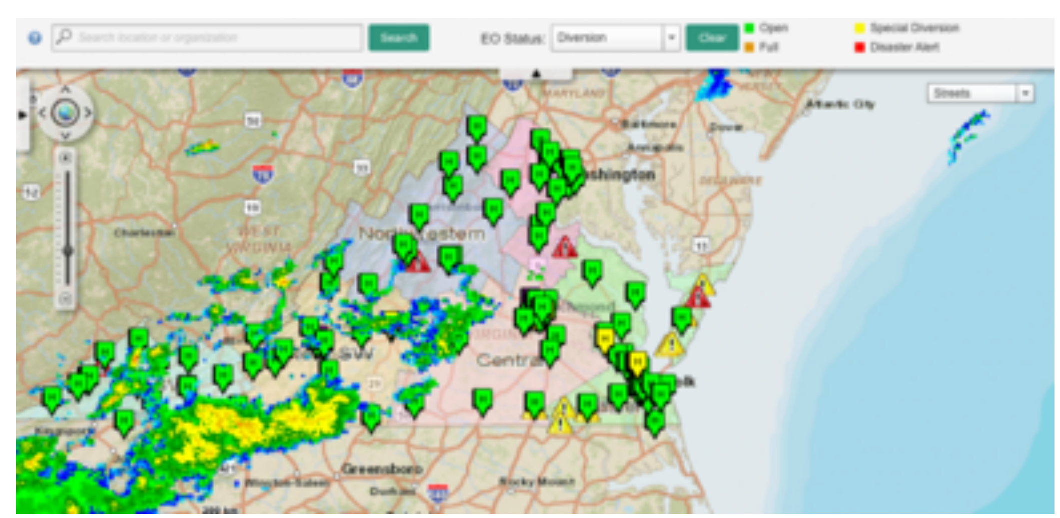
- In addition to structured/unstructured field data, you can have non-field geometry.



  - Boundary surface meshes
  - Atom positions, bonds, ribbons



- Non-geometric annotations
  - Especially in GIS.
- Visualization data models are complex!

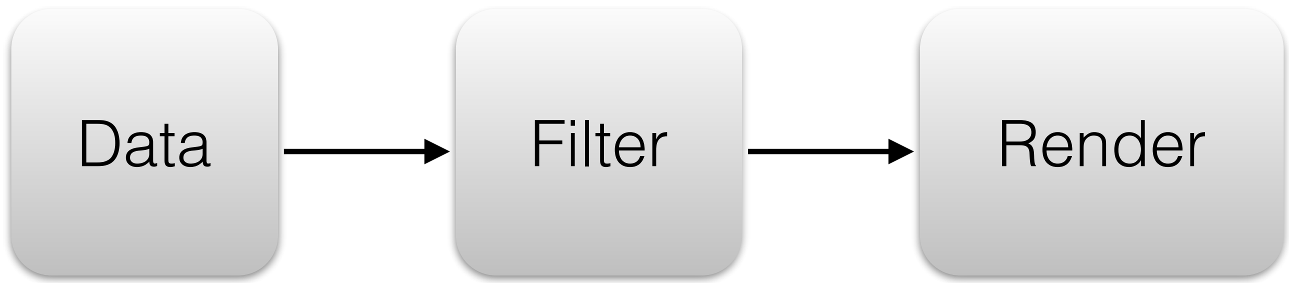


www.sci.utah.edu

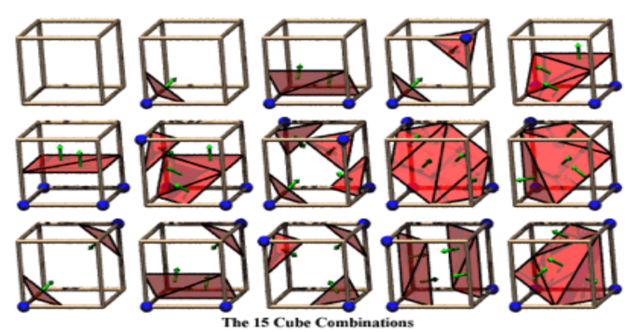
What do we do with these data?

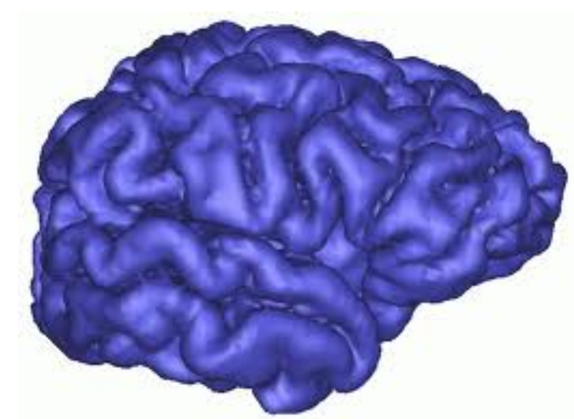
- In computer graphics, life is “easy”
  - Have a triangle mesh, render it!
- Visualization is more than just rendering.
- Two approaches:
  - direct visualization:
    i.e. render from a (usually 3D) field directly
  - indirect visualization:
    i.e. convert the field to triangles and render those (usually with GPU rasterization)

Indirect visualization



0	4	8	0
4	14	9	0
6	11	1	0
2	1	0	0

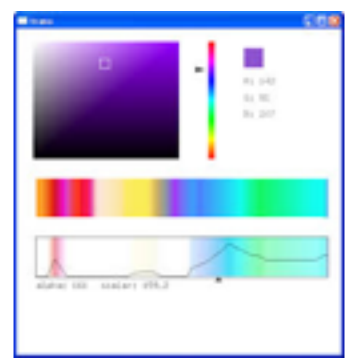


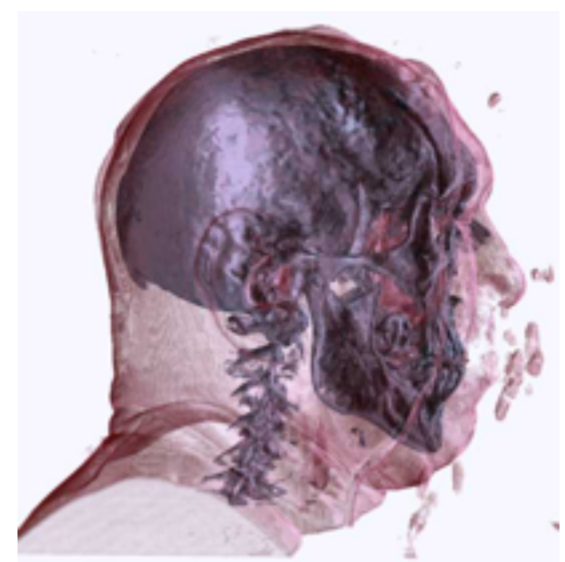


Direct visualization

Data → Filter + Render

0	4	8	0
4	14	9	0
6	11	1	0
2	1	0	0





The indirect-direct spectrum

- Most data must be processed somewhere after acquisition
- Direct/Indirect are not “absolute”, but relative.
  - Direct = less processing prior to rendering
  - Indirect = more processing prior to rendering
- Multiple questions:
  - how do the data need to be transformed for the desired analysis?
  - what does the target rendering engine / API require?
- There is no such thing as pure “direct” or “indirect” visualization; only “more direct” or “more indirect”.

The indirect-direct spectrum

isosurface extraction (marching cubes)
+ rasterization

direct isosurface ray casting

segmentation+filtering+
classification+rasterization pipeline

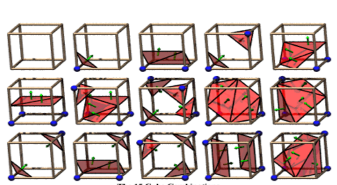
volume rendering
from raw data

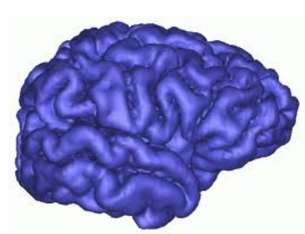
polygonal ⟷ splatting

Indirect

Data → Filter → Render

0	4	8	0
4	14	9	0
6	11	1	0
2	1	0	0

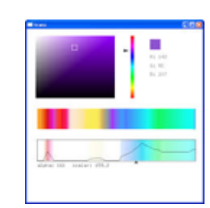


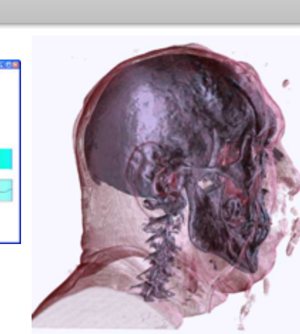


Direct

Data → Filter + Render

0	4	8	0
4	14	9	0
6	11	1	0
2	1	0	0

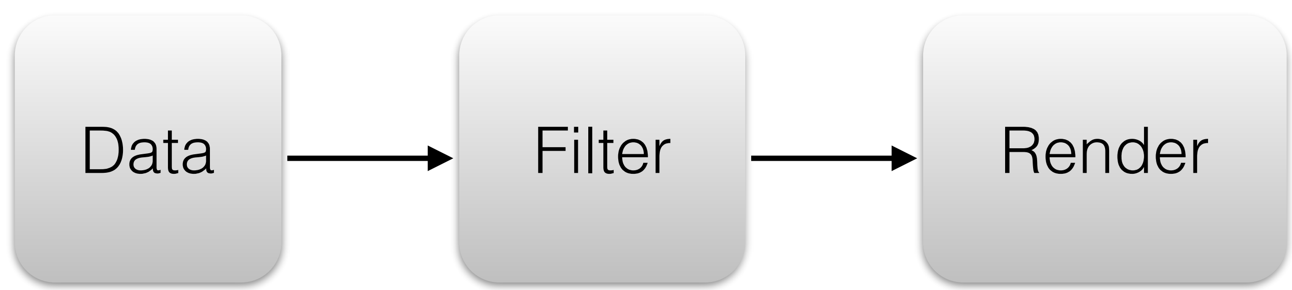




www.sci.utah.edu

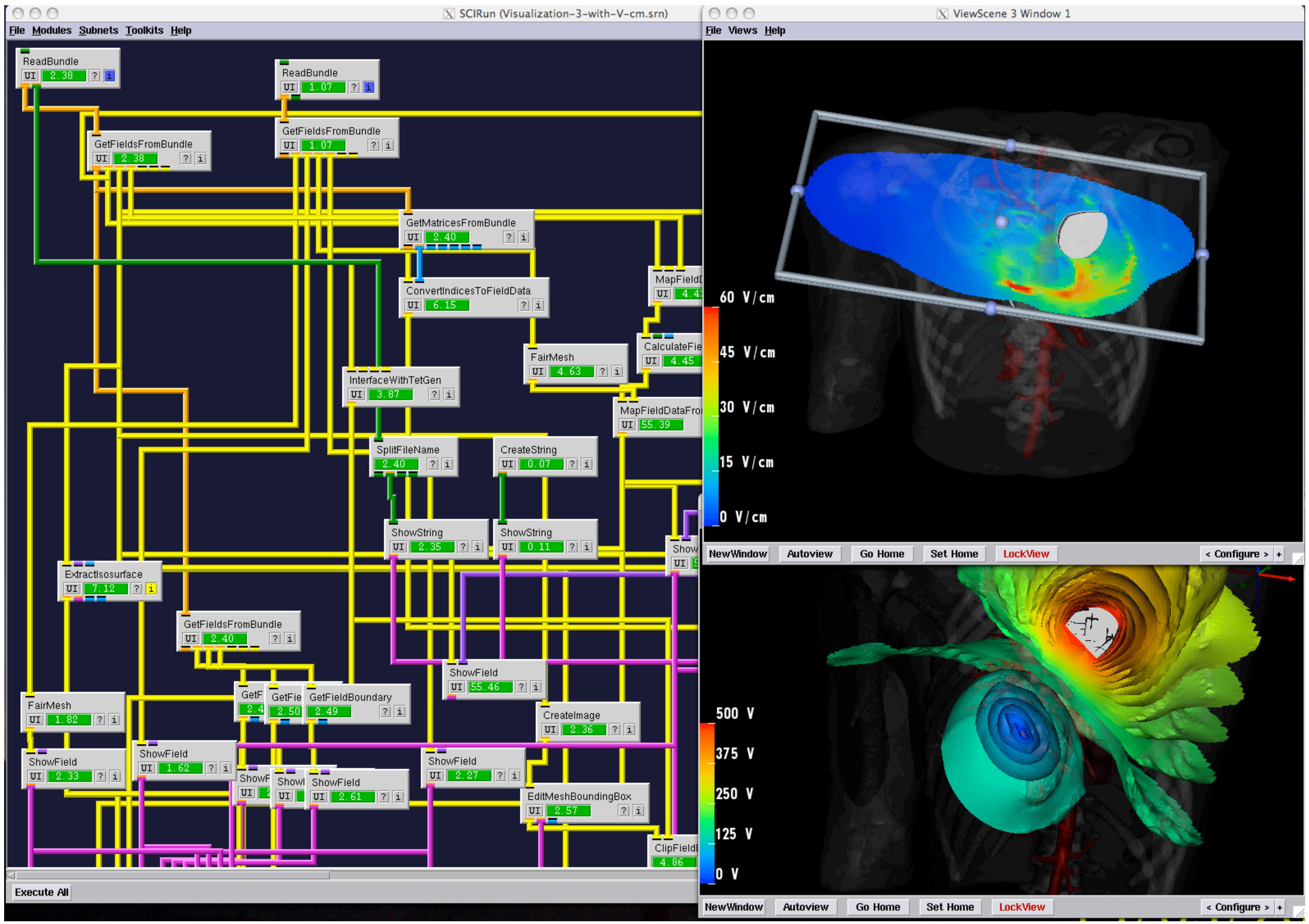
The visualization pipeline

- Even if we merge filtering and rendering, it is still helpful to think of them as a chain of operations.



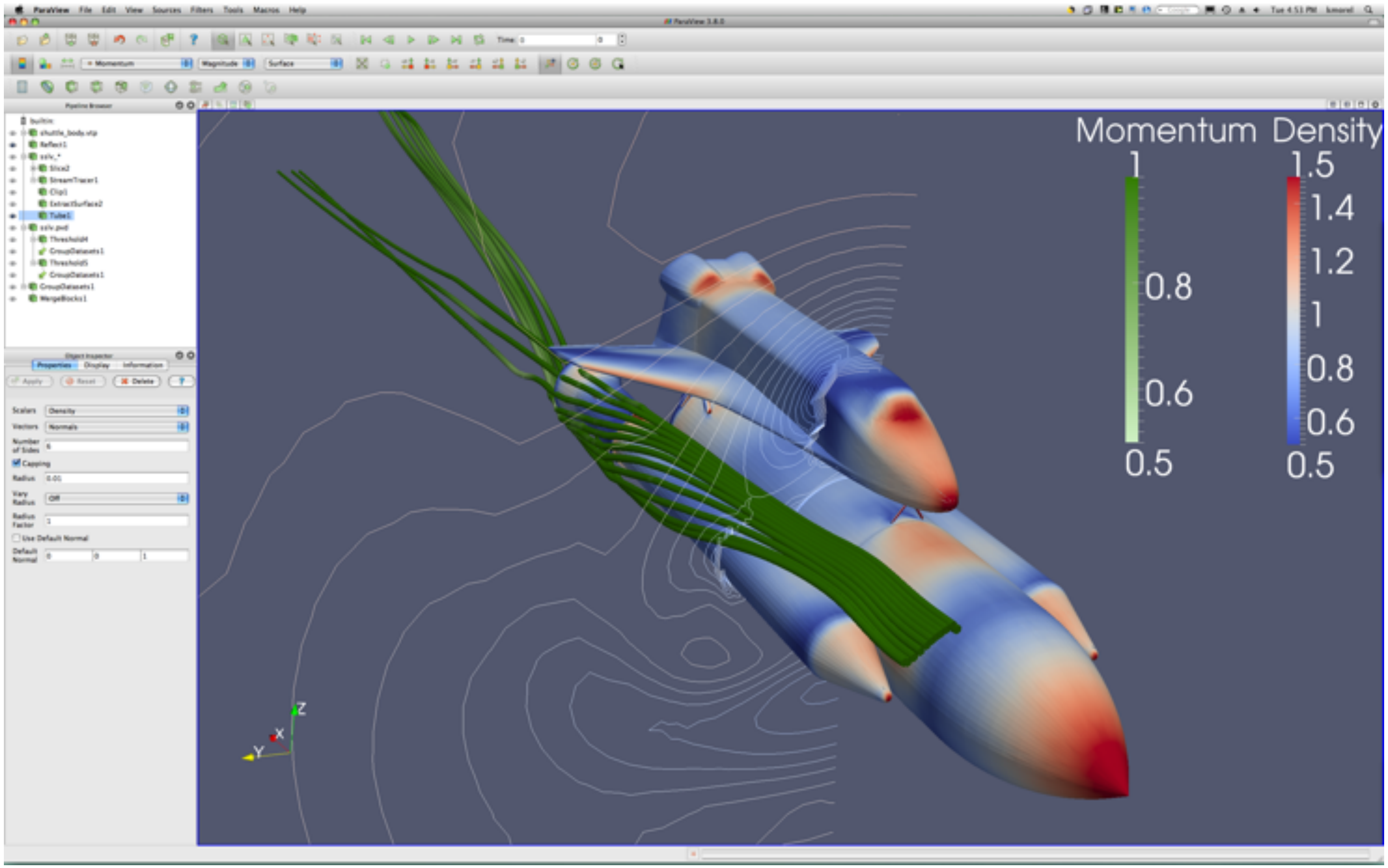
- Visualization workflows take the form of a flow chart, tree or network…

SCIRun



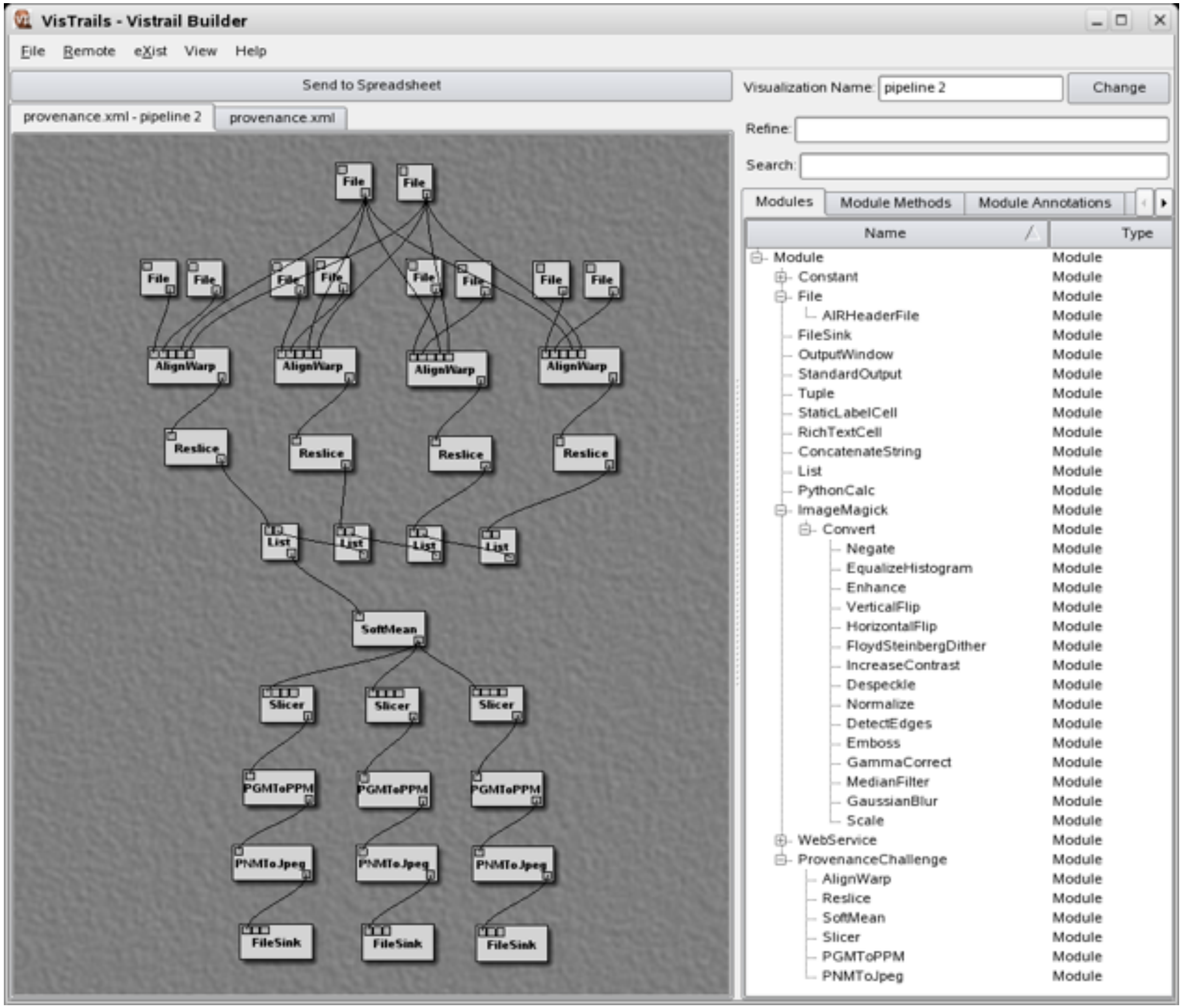
www.sci.utah.edu

ParaView



www.sci.utah.edu

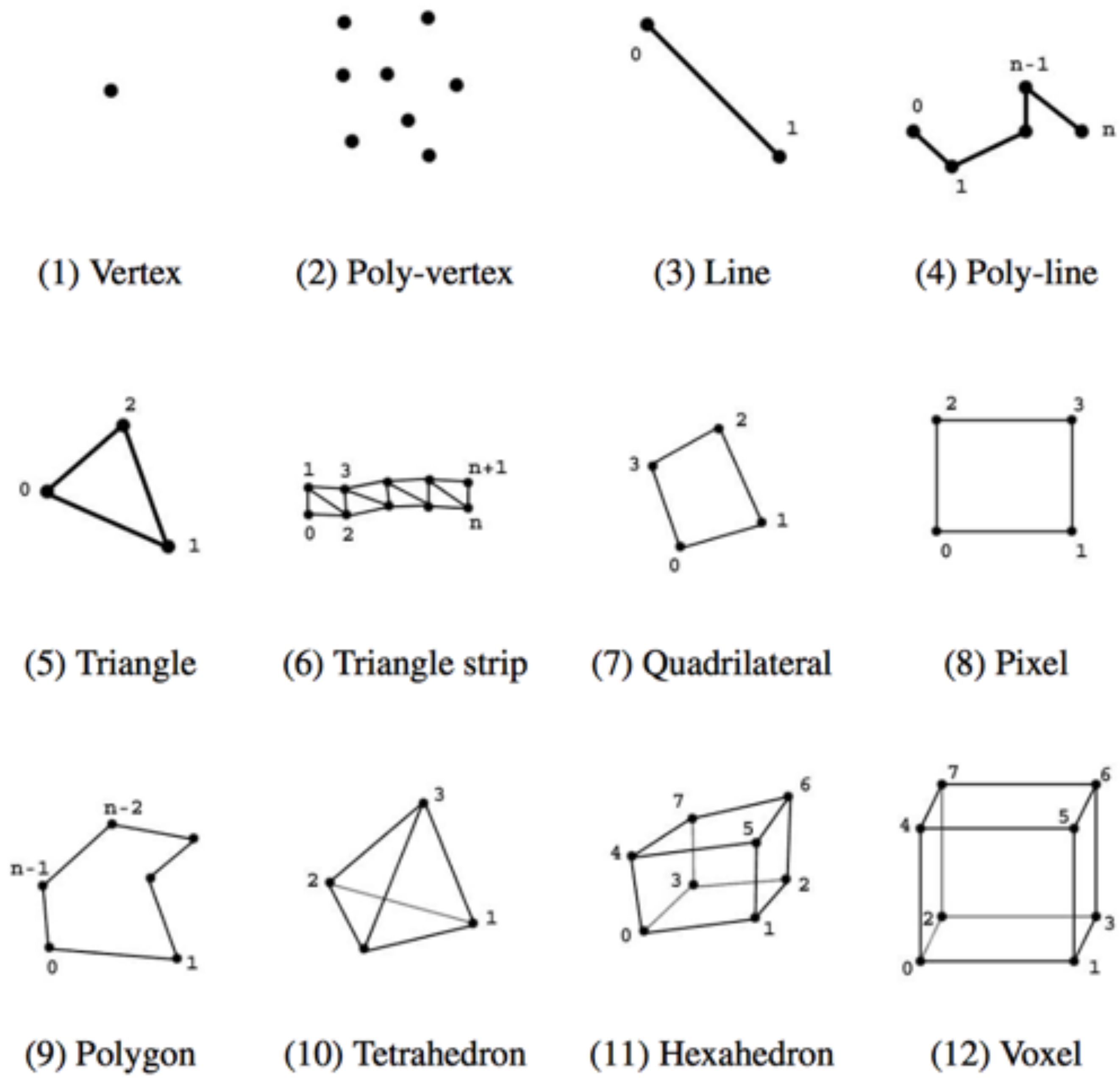
VisTrails


VisTrails - Vistrail Builder
File Remote eXist View Help
Send to Spreadsheet
provenance.xml - pipeline 2
provenance.xml
Visualization Name: pipeline 2
Change
Refine:
Search:
Modules
Module Methods
Module Annotations
Name
Type
Module Module
Constant Module
File Module
AIRHeaderFile Module
FileSink Module
OutputWindow Module
StandardOutput Module
Tuple Module
StaticLabelCell Module
RichTextCell Module
ConcatenateString Module
List Module
PythonCalc Module
ImageMagick Module
Convert Module
Negate Module
EqualizeHistogram Module
Enhance Module
VerticalFlip Module
HorizontalFlip Module
FloydSteinbergDither Module
IncreaseContrast Module
Despeckle Module
Normalize Module
DetectEdges Module
Emboss Module
GammaCorrect Module
MedianFilter Module
GaussianBlur Module
Scale Module
WebService Module
ProvenanceChallenge Module
AlignWarp Module
Reslice Module
SoftMean Module
Slicer Module
PGMToPPM Module
PNMToJpeg Module
File
AlignWarp
Reslice
List
SoftMean
Slicer
PGMToPPM
PNMToJpeg
FileSink

VTK

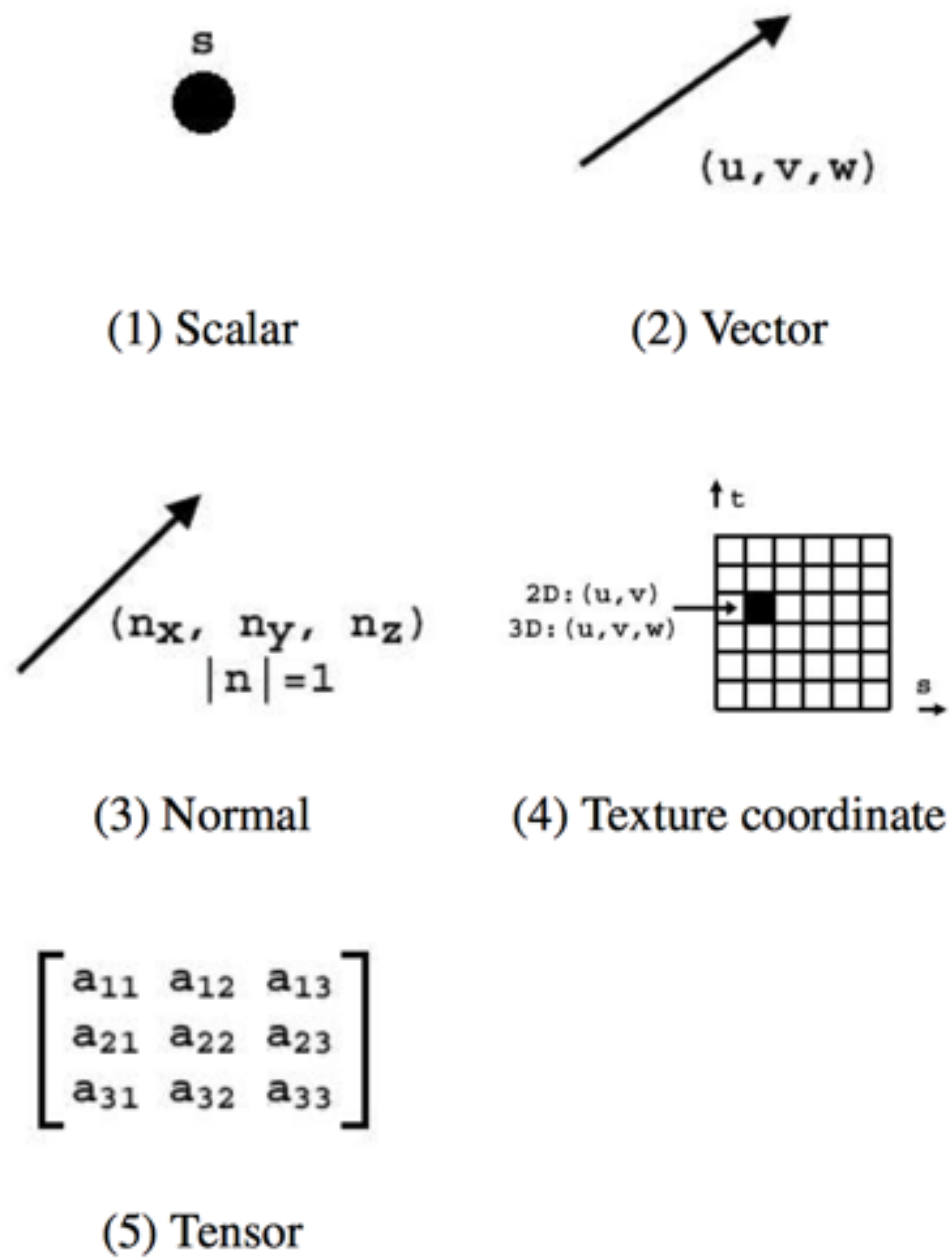
- http://www.vtk.org, open-source, developed and maintained by Kitware.
- The standard-bearer API for general-purpose scientific visualization
  - Full-fledged data model for structured, unstructured, particle data
  - Marching cubes, cut/clip planes, streamlines, etc.
  - Hundreds of other analysis filters
  - Numerous readers for common scientific formats
- Call as a library from C++, Java, Python, Tcl/Tk
- Limitations:
  - no UI — you need to code (or at least, script) your workflows.
  - data model can be “heavy”, memory-inefficient — but it nearly always works!

Cell types in VTK



S. Bruckner, “Data Structures in the Visualization Toolkit.”

Attribute types in VTK



S. Bruckner, "Data Structures in the Visualization Toolkit."

www.sci.utah.edu

Simple data flow in VTK

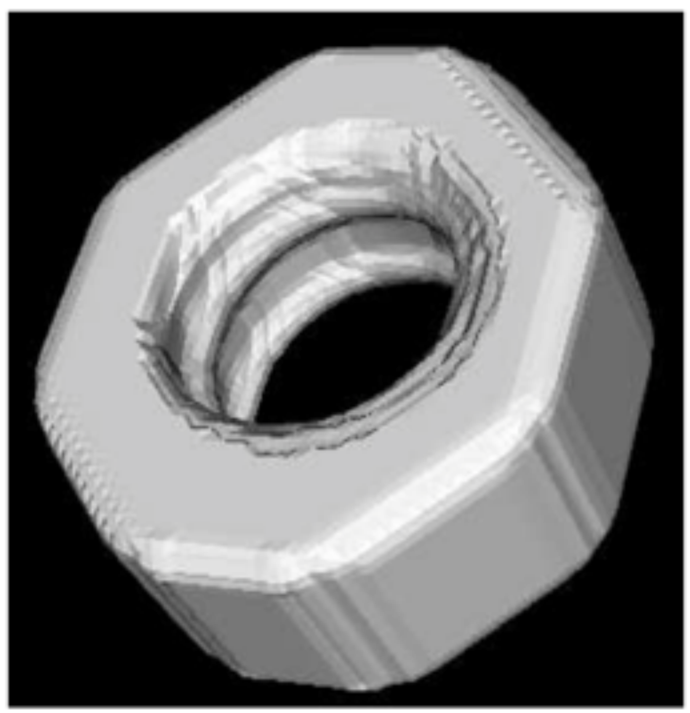


Figure 7: Resulting image for program 3.2

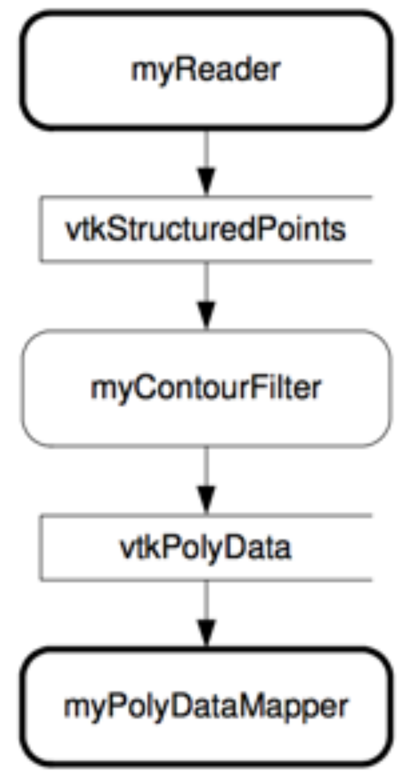


Figure 8: Data-flow chart for program 3.2

S. Bruckner, “Data Structures in the Visualization Toolkit.”

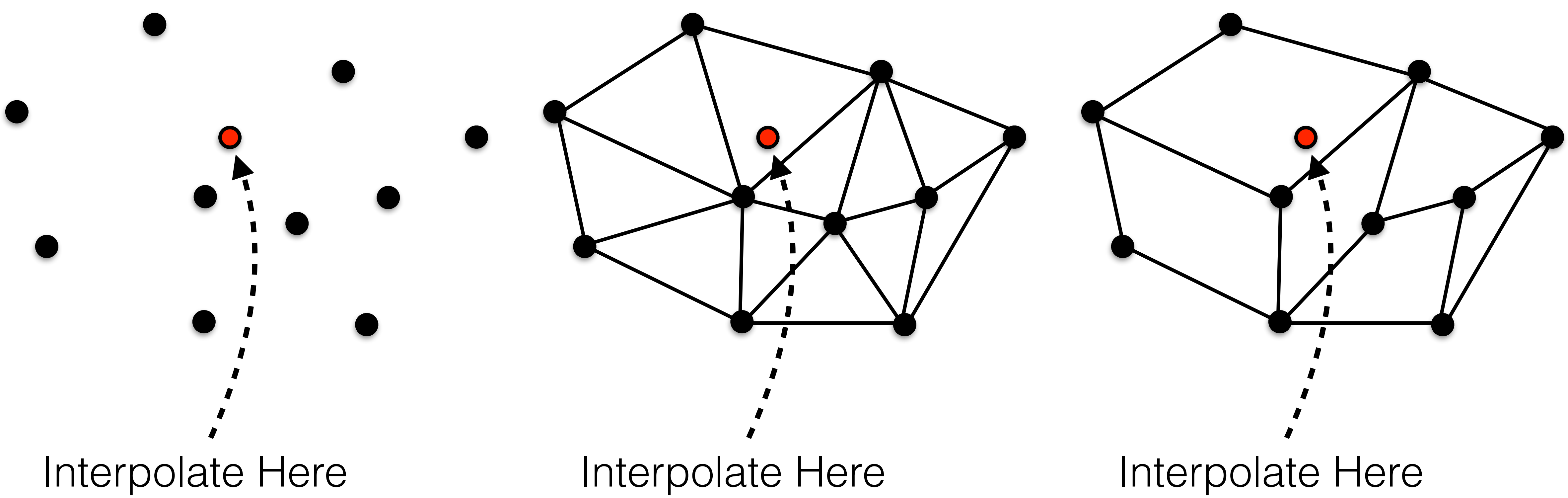
Interpolation

Interpolation

- Converting from discrete to continuous — i.e. **grid** to **field**.
- How do we find the values of points “inside” a grid/mesh?
    - Indirect visualization: how to find vertices of triangles
    - Direct visualization: how to find the value of samples in space
- i.e., from an explicit grid, we want to evaluate a field **f** at some point **x**
    - The way we do this interpolation is called the **filter kernel** of the field f.

Mesh Choice Impacts How the Continuous Data is Interpreted

- Two key questions:
  - Sampling, or the choice of where attributes are measured
  - Interpolation, or how to model the attributes in the rest of space



Interpolation

- **Continuous** reconstruction of **discrete** input data

position

$$F : \mathbb{R}^n \longrightarrow \mathbb{R}^m$$

$$(\mathbf{x}_i, f_i)$$

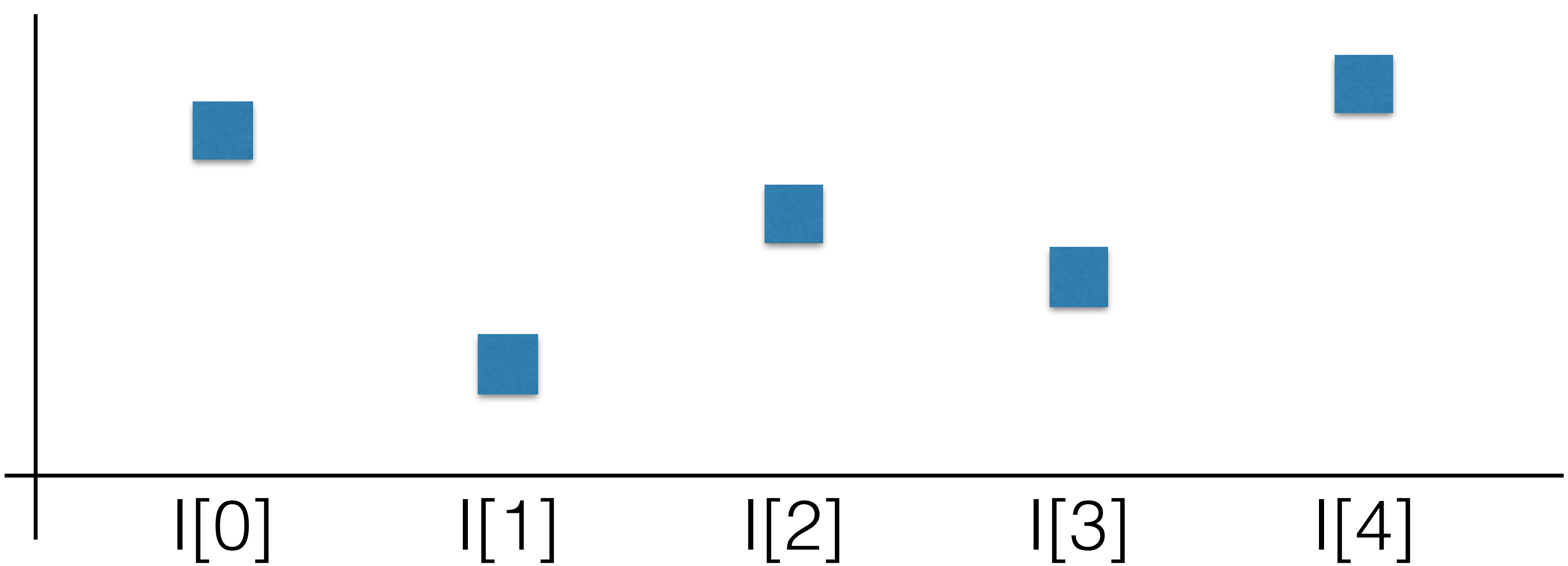
value

$$\forall i \in \{1, .., n\}, F(\mathbf{x}_i) = f_i$$

- Depends on grid structure (when available)
- Interpolation vs. approximation

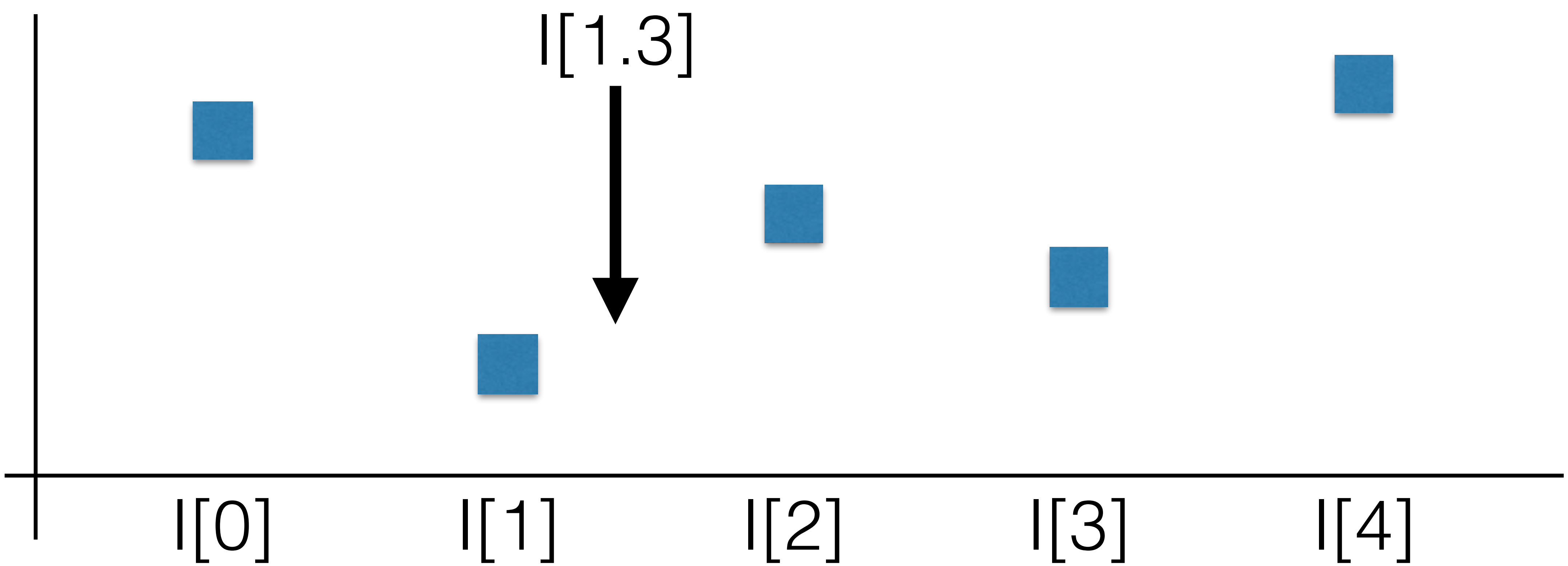
Nearest Neighbor Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?



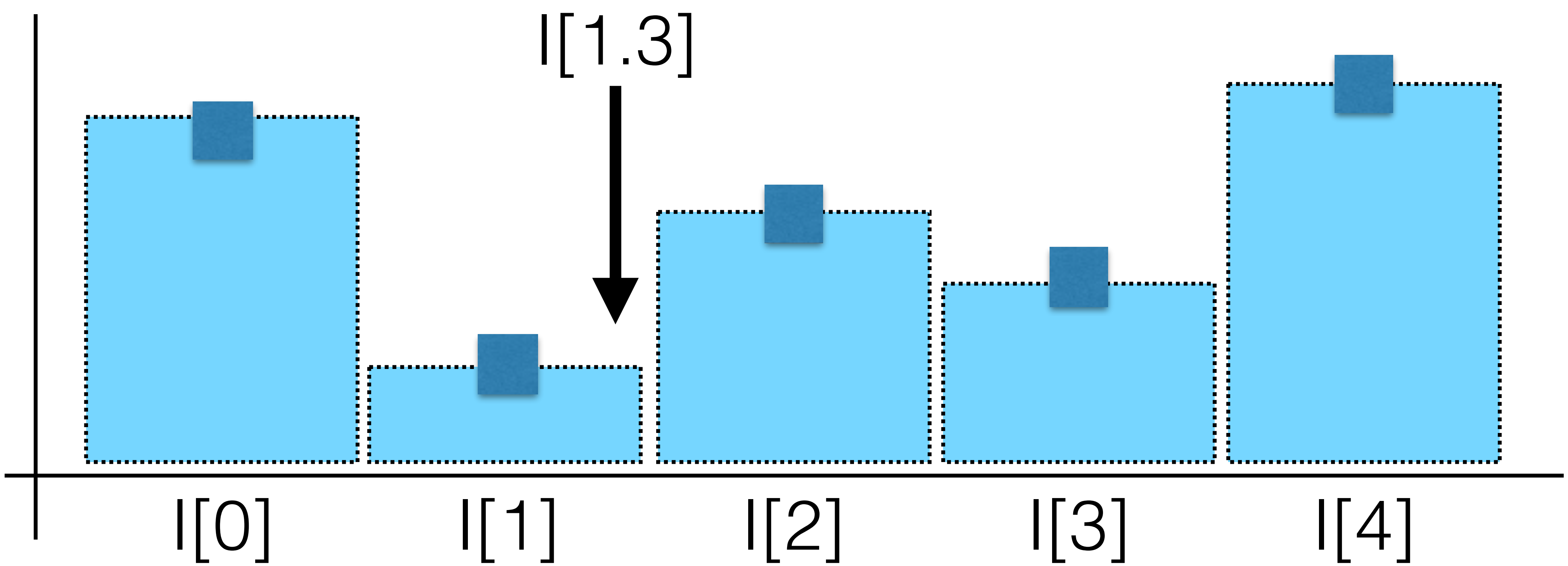
Nearest Neighbor Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?



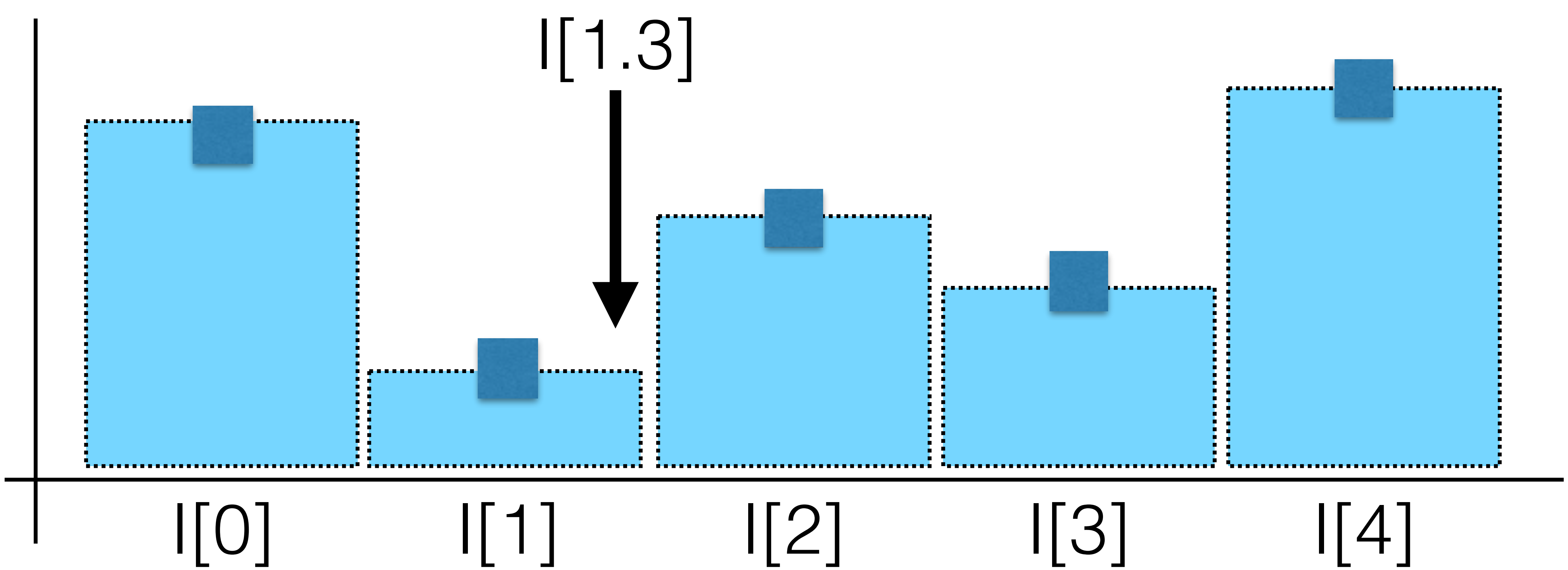
Nearest Neighbor Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?



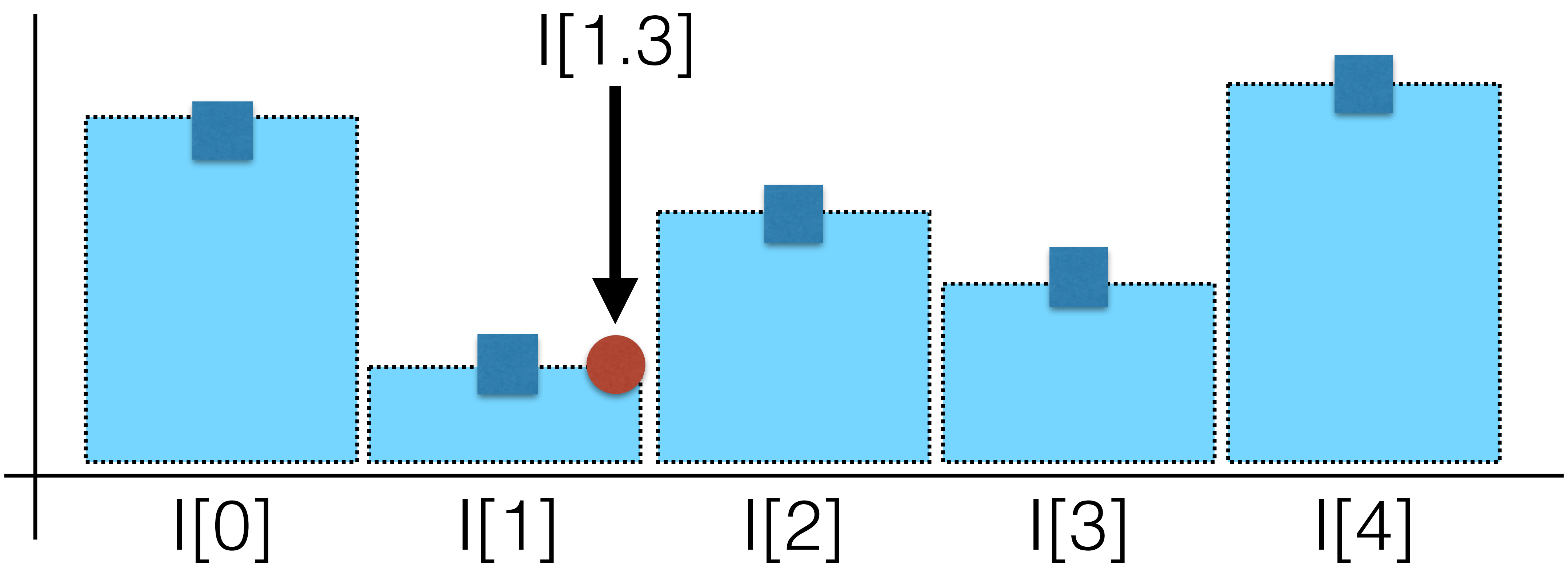
Nearest Neighbor Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?
  - I[1.3] = I[round(1.3)] = I[1]



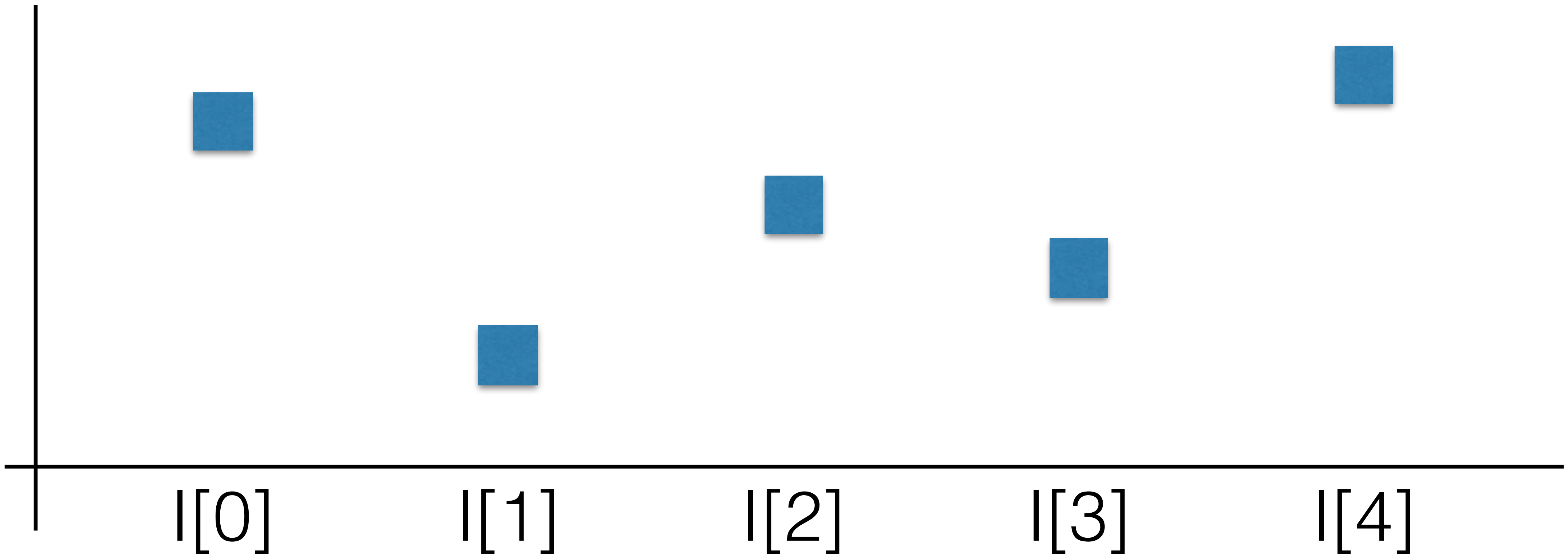
Nearest Neighbor Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?
  - I[1.3] = I[round(1.3)] = I[1]



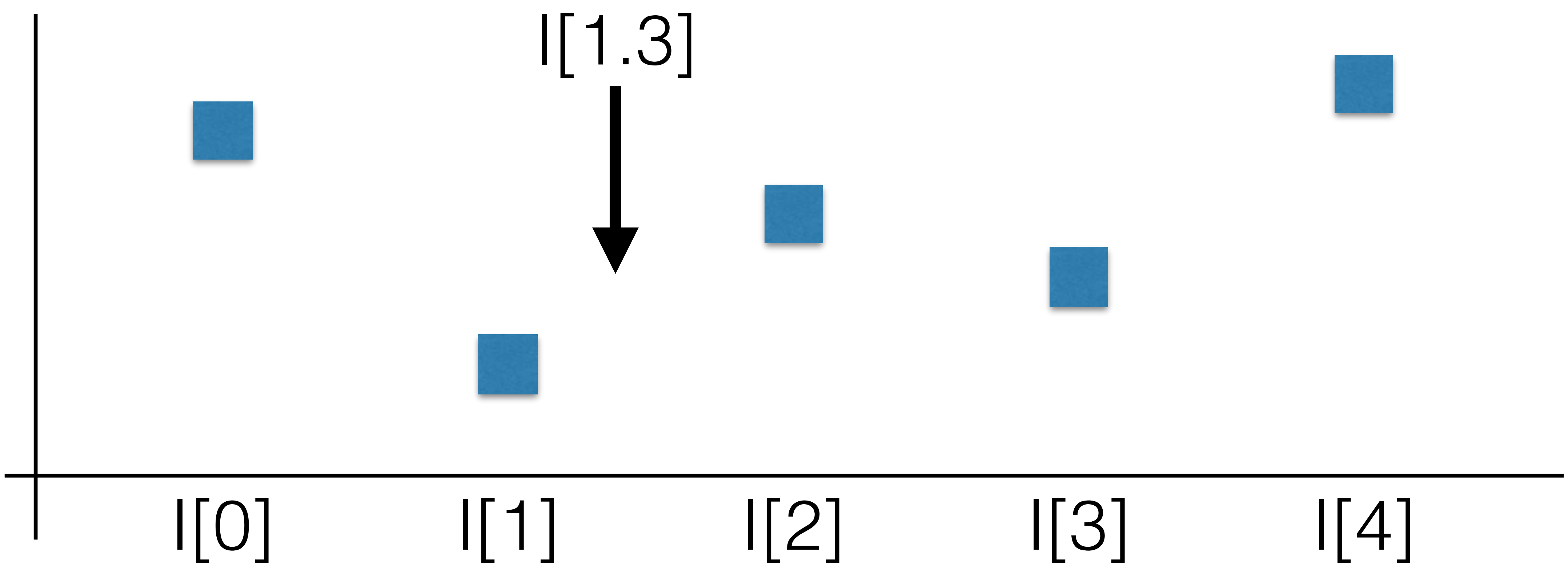
Linear Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?



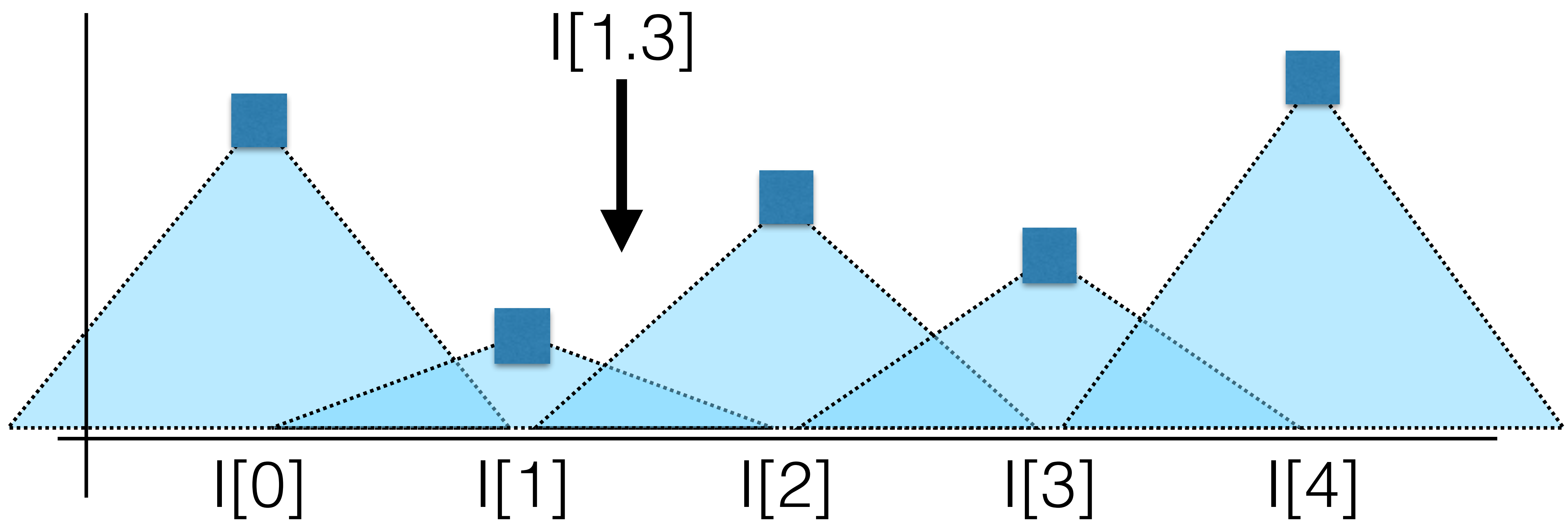
Linear Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?



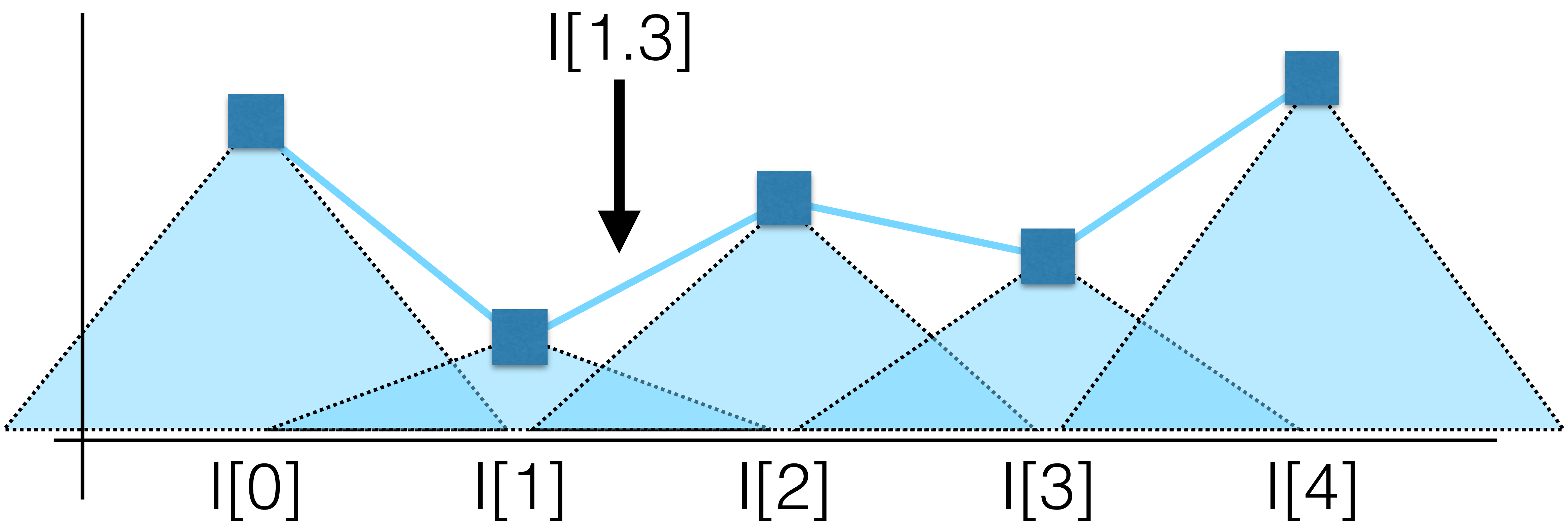
Linear Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?



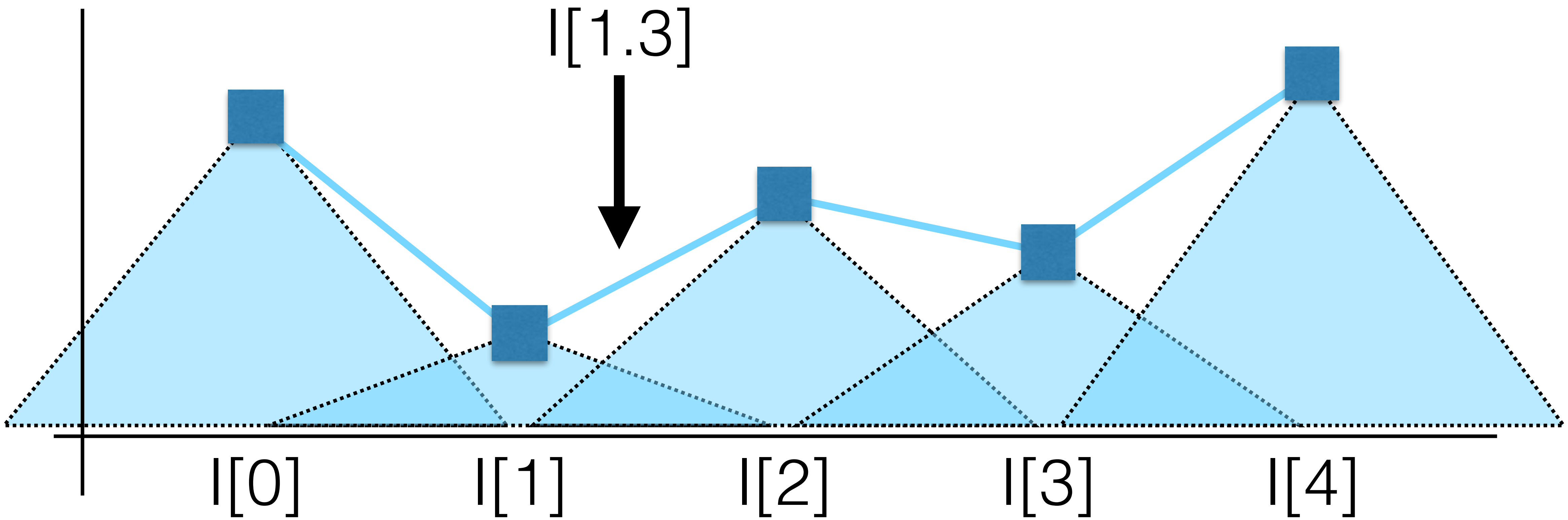
Linear Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?



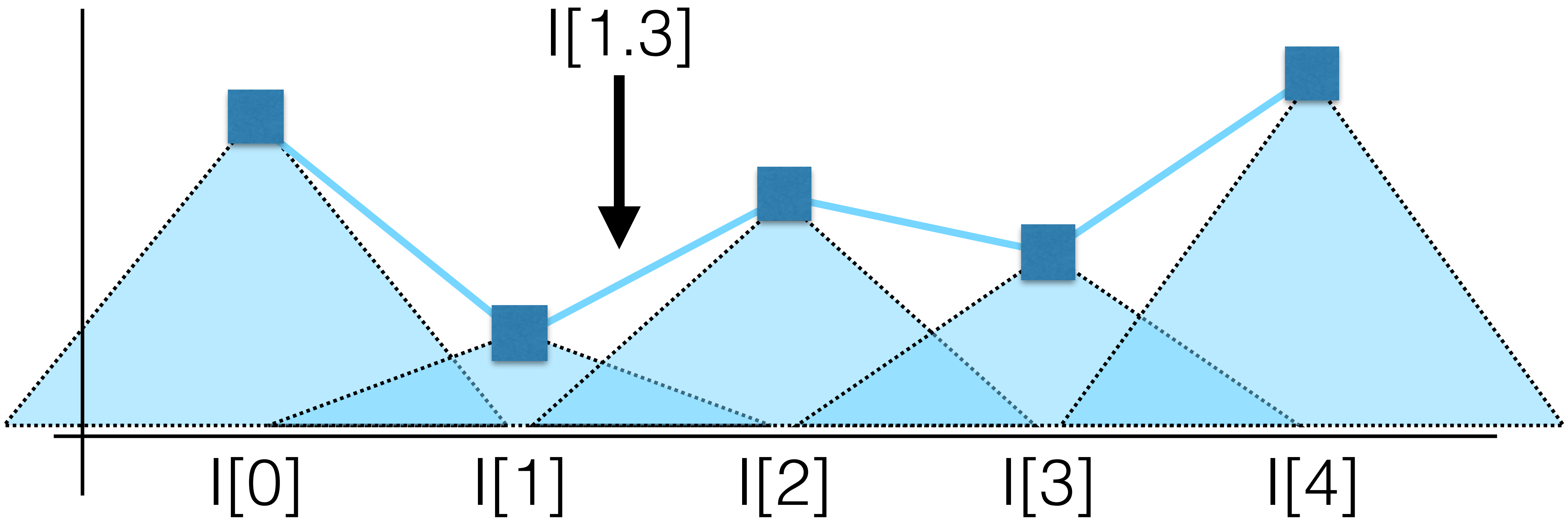
Linear Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?
  - Let s = 1.3 - round(1.3)



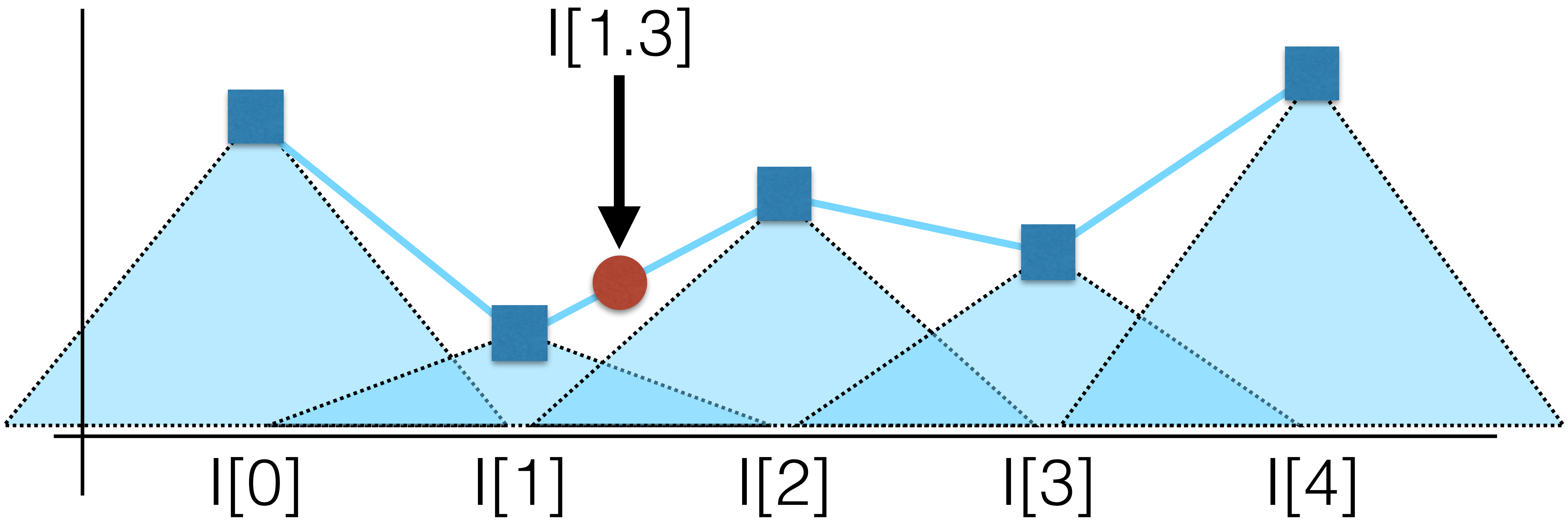
Linear Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?
  - Let s = 1.3 - round(1.3)
  - I[1.3] = 0.7*I[1] + 0.3*I[2] = (1-s)*I[1] + s*I[2]



Linear Interpolation

- Consider a 1-dimensional, grayscale image I spread horizontally
- What value is I[1.3] ?
  - Let s = 1.3 - round(1.3)
  - I[1.3] = 0.7*I[1] + 0.3*I[2] = (1-s)*I[1] + s*I[2]



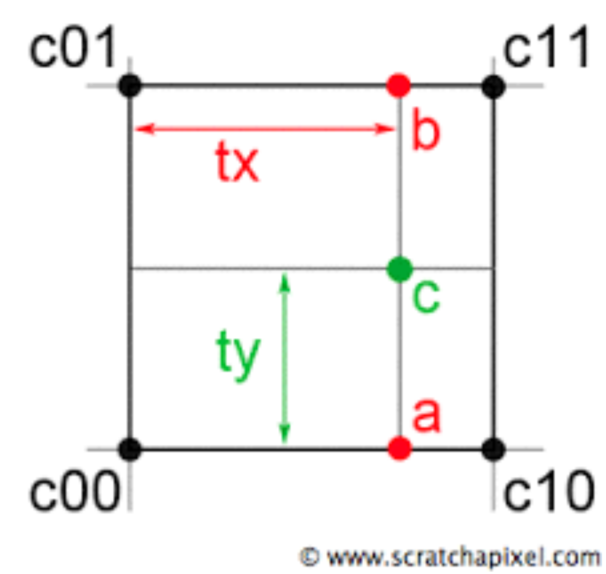
Linear interpolation

```
#define lerp(a,b,t)  (1-t) * a + t*b
```

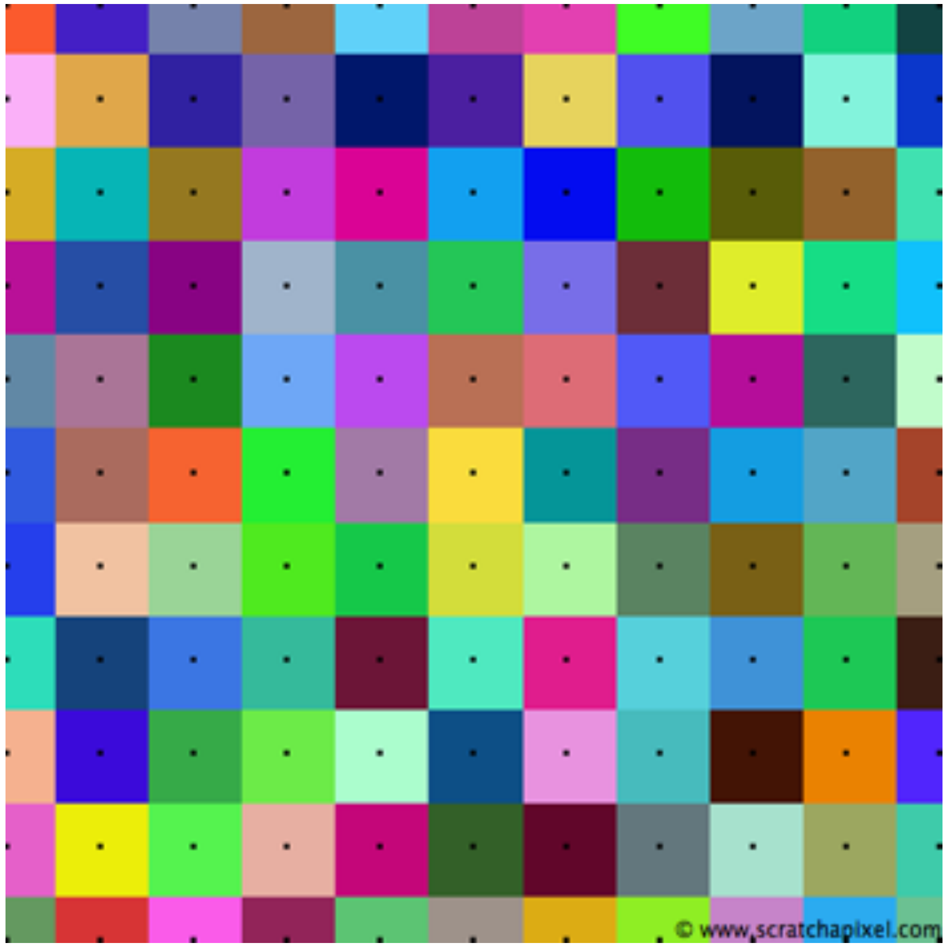
Bilinear interpolation

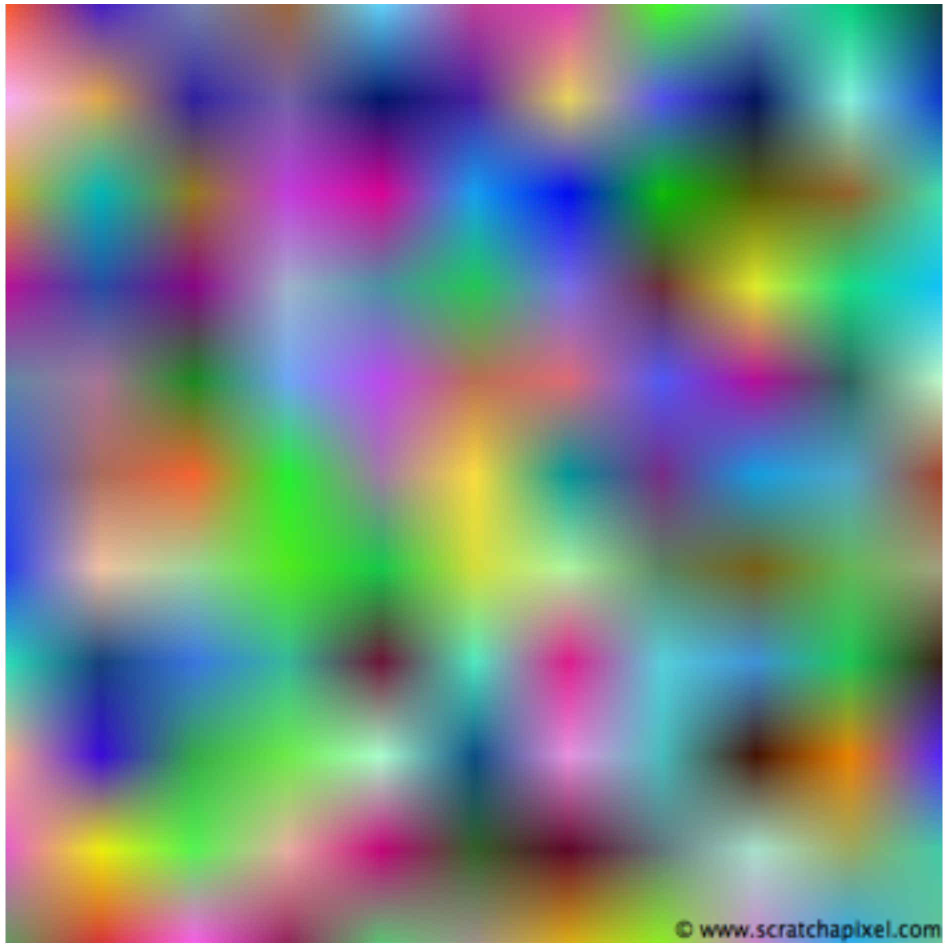
Just 3 linear interpolations

```
#define lerp(a,b,t)  (1-t) * a + t*b

//Given voxel vertices cXX and the (tx,ty) position within the voxel [0,1]^2

//lerp along y direction
float c00_y = lerp(c00, c01, ty);
float c10_y = lerp(c10, c11, ty);

//lerp along x direction
return lerp(c00_y, c10_y, tx);
```



Bilinear interpolation

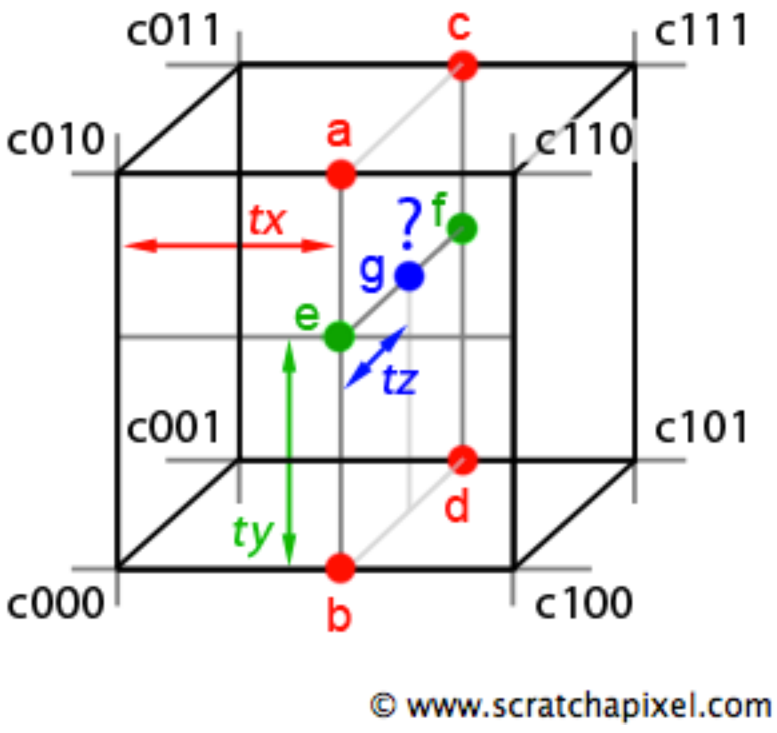




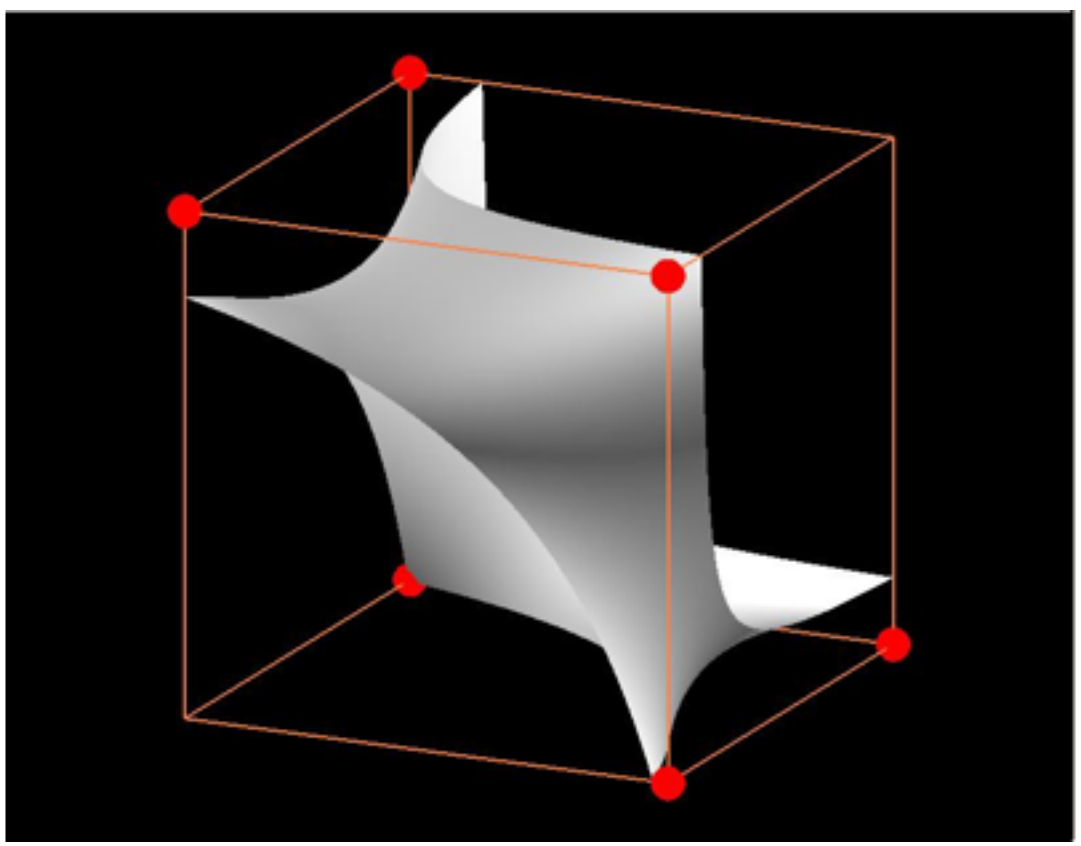
Trilinear interpolation

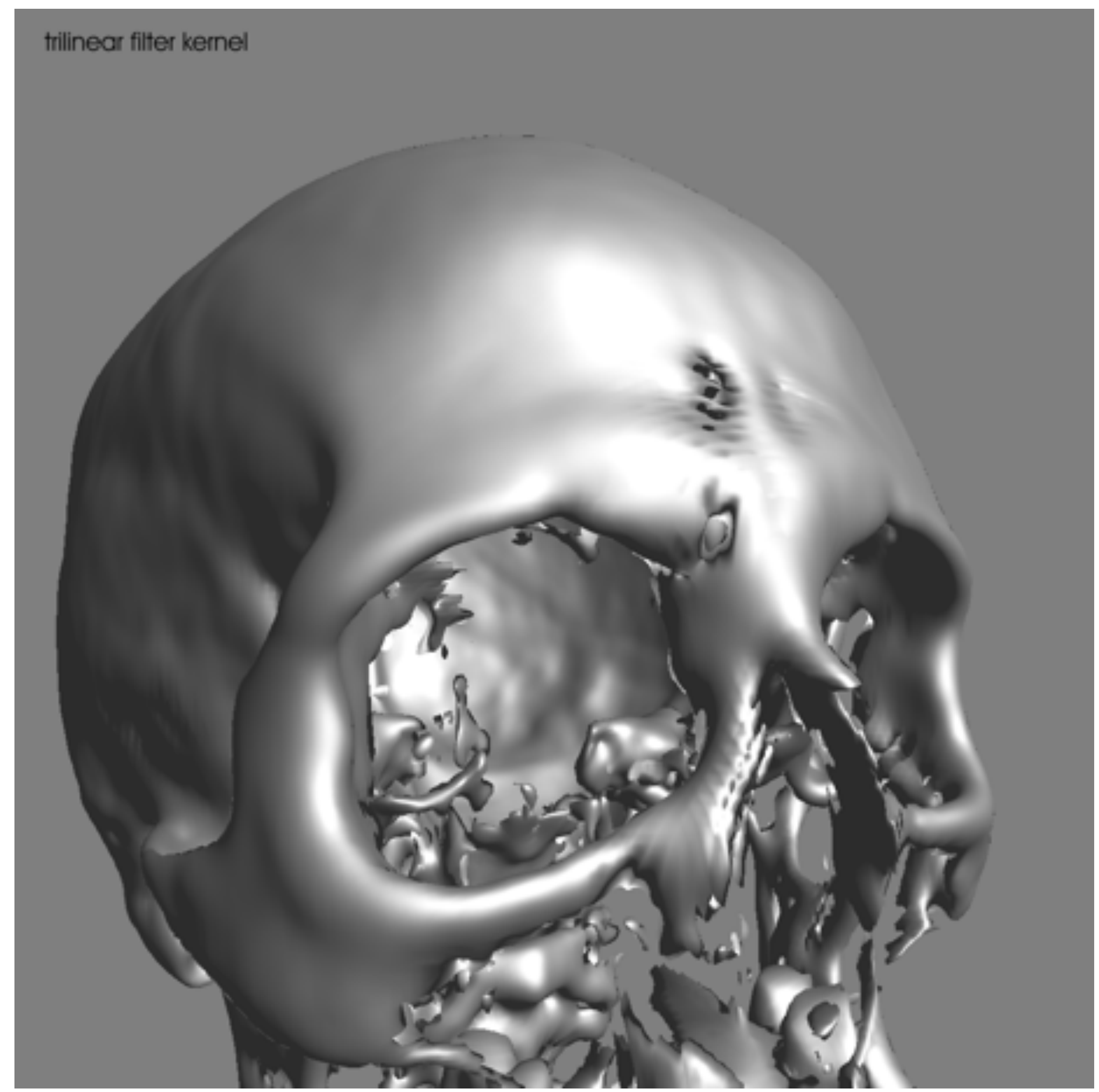
Just 7 linear interpolations!

```
#define lerp(a,b,t)  (1-t) * a + t*b

//Given voxel vertices cXXX and the (tx,ty,tz) position within the voxel [0,1]^3

//lerp along z direction.
float c000_z = lerp(c000, c001, tz);
float c010_z = lerp(c010, c011, tz);
float c100_z = lerp(c100, c101, tz);
float c110_z = lerp(c110, c111, tz);

//lerp along y direction
float c000_yz = lerp(c000_z, c010_z, ty);
float c100_yz = lerp(c100_z, c110_z, ty);

//lerp along x direction
return lerp(c000_yz, c100_yz, tx);
```



Isosurfaces from trilinear filter kernels





Another formula for trilinear interpolation (3D)

$$f(x,y,z) = \sum_{i,j,k=\{0,1\}} x_i y_j z_k \, v_{ijk}$$

Where $x_0 = i+1-x, x_1 = x-i$, ditto for y and z

And v_{ijk} is the value of the voxel at that vertex.

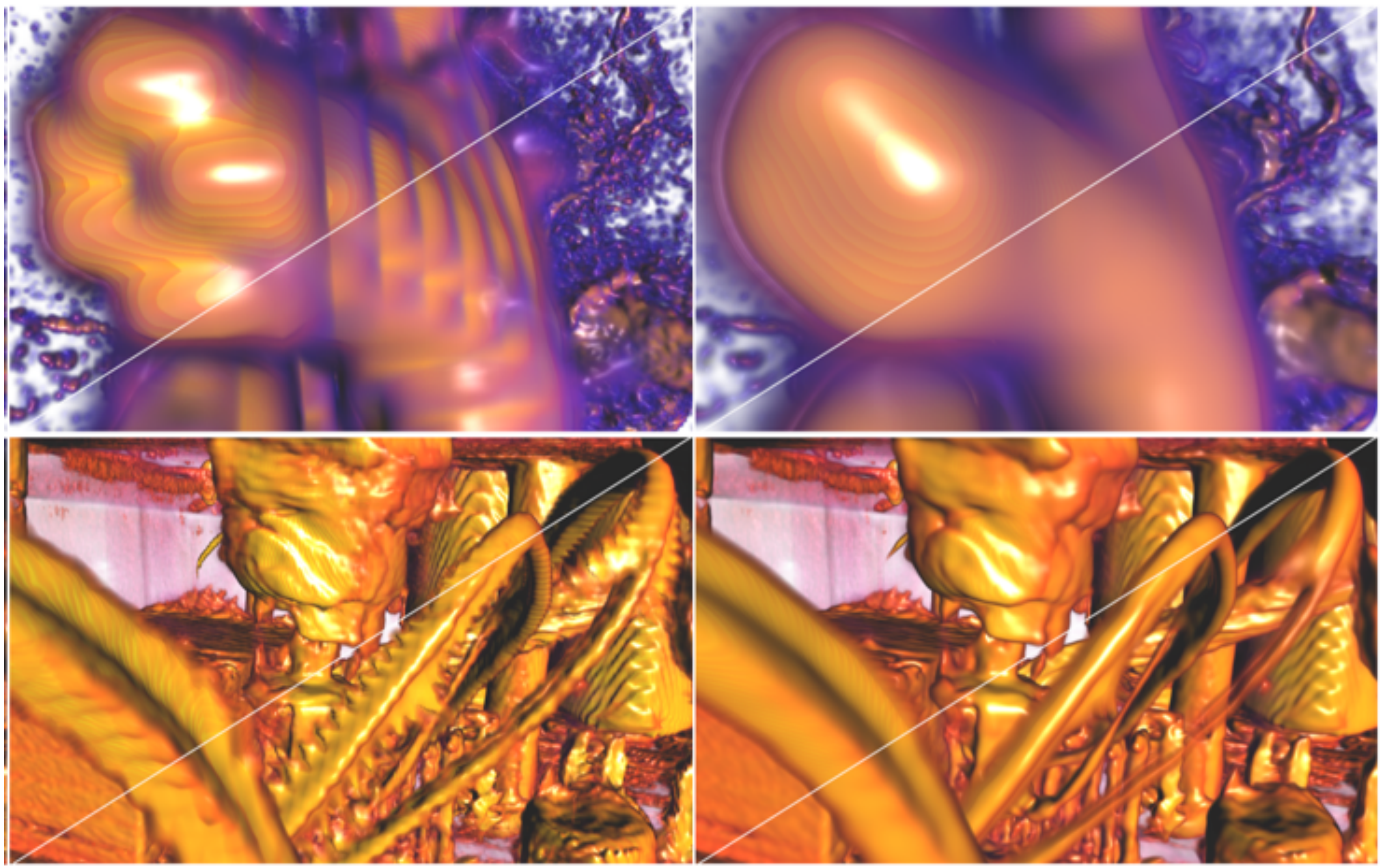
General interpolation (3D)

$$f(x, y, z) = \sum_{i,j,k} B_i(x) B_j(y) B_k(z) v_{ijk}$$

Where the B(x) is a general basis function, v is the voxel.

Trilinear vs B-spline filtering for volume rendering

trilinear tri - cubic-B-spline



Next Tuesday lecture

- Volume rendering.