CSC 7700: Scientific Computing

Module D: Scientific Visualization

Lecture 1: Overview

Dr. Werner Benger

Overview Module D

Grading:

10% Homework

1st Assignment due Oct 5th

2nd Assignment due Oct 27th

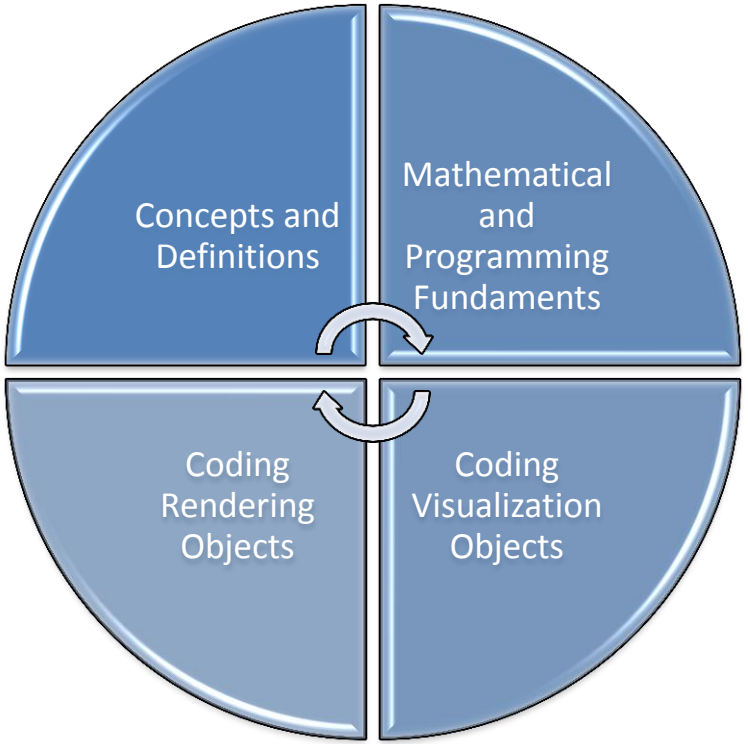
10% Project or Exam

Will decide by end of 4th lecture.

Office hours:

CCT JH, Room 211

Tuesday, Thursday 2pm-3pm



Overview of Course

- Definition and Scope of Scientific Visualization
- Concept of the Visualization Pipeline
- Data Flow Graph / Control Flow Graph
- Visual Programming
- Overview of Visualization Software
- Introduction to the "Vish" Visualization Shell
- Introduction to Data Types
- File formats - HDF5
- Homework

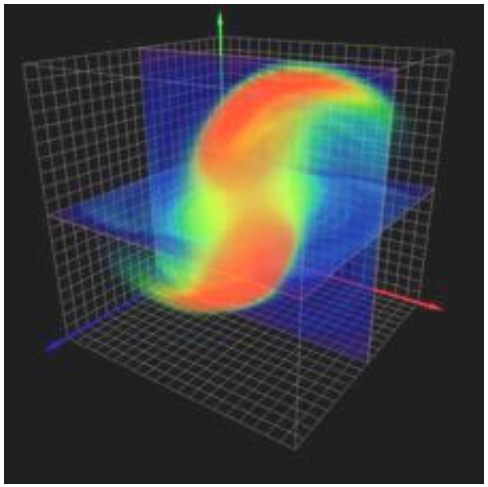
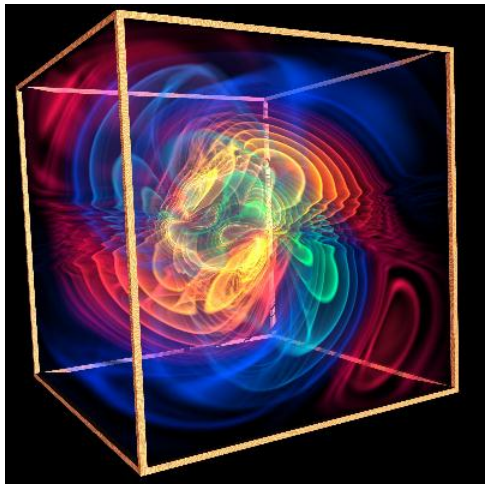
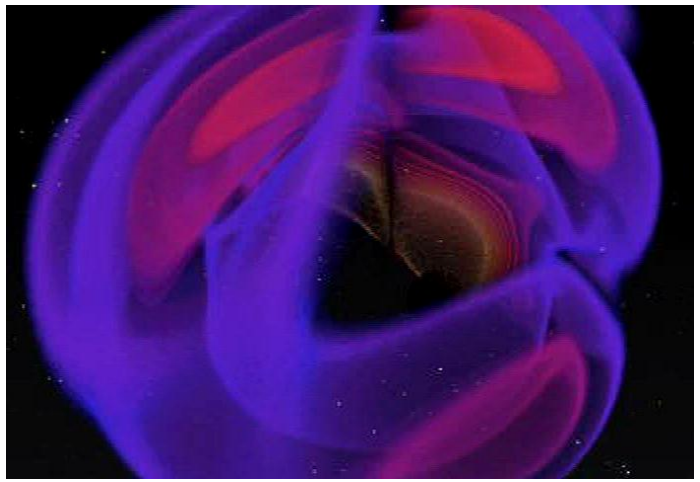
Definition "Scientific Visualization"

Scientists need an alternative to numbers. The use of images is a technical reality nowadays and tomorrow it will be an essential requisite for knowledge. The ability of scientists to visualize calculations and complex simulations is absolutely essential to ensure the integrity of analyses, to promote scrutiny in depth and to communicate the result of such scrutiny to others... The purpose of scientific calculation is looking, not enumerating. It is estimated that 50% of the brain's neurons are associated with vision. Visualization in a scientific calculation is aimed at putting this neurological machinery to work.

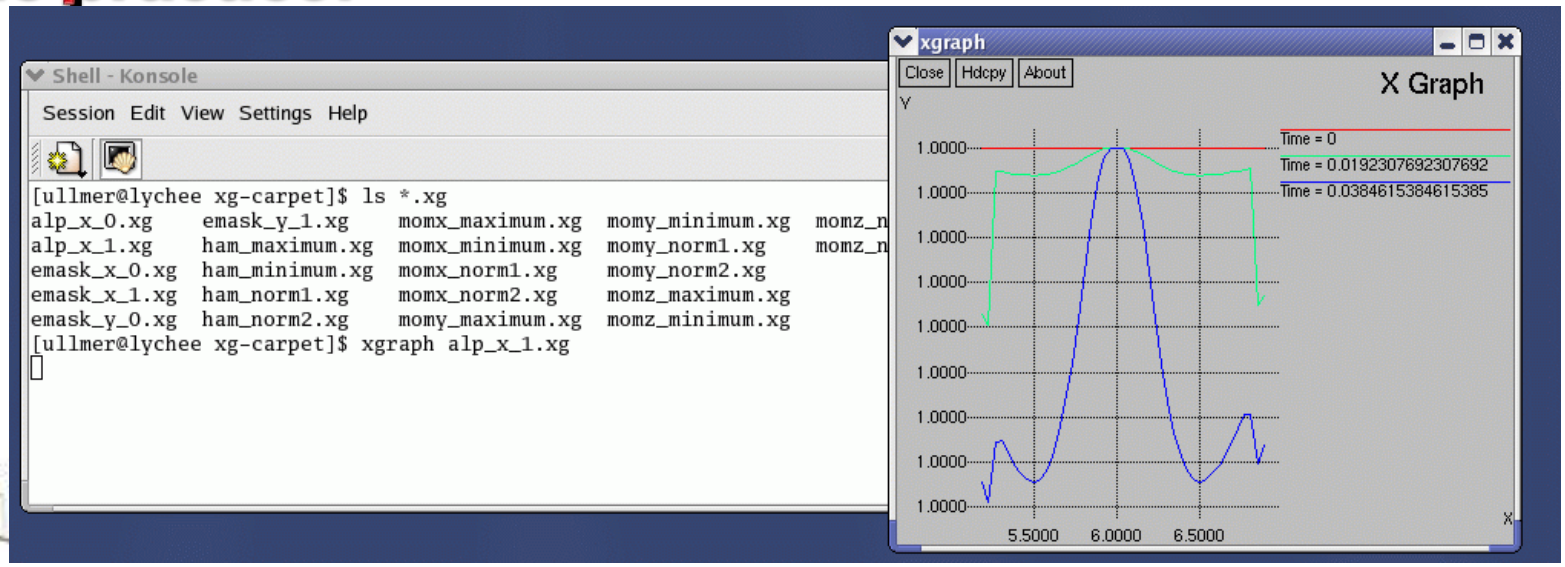
McCormick et al., 1987

Working context : visualization

The potential:

The practice:



Scientific visualization - Wikipedia, the free encyclopedia - Mozilla Firefox

File Edit View History Bookmarks Tools Help

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

Facebook TigerTrails SomaFM Vish LEO MeetingMaker Translator HTML PHP Manua http://www.youtube.... SciViz@CCT Forum sciviz phpMyAdmin Satellites Most Visited Lates...

Scientific visualization - Wikipedia, t...

Try Beta Log in / create account

article discussion edit this page history

WIKIPEDIA
The Free Encyclopedia

navigation
- Main page
- Contents
- Featured content
- Current events
- Random article

search
Go Search

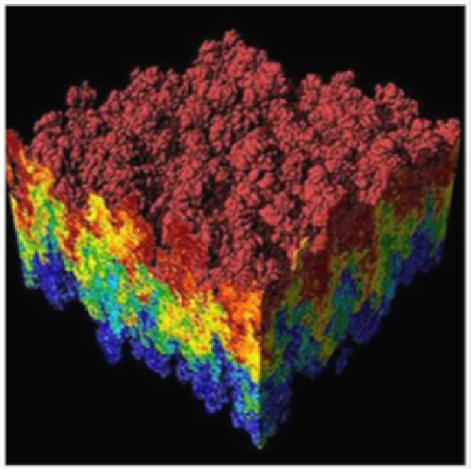
interaction
- About Wikipedia
- Community portal
- Recent changes
- Contact Wikipedia
- Donate to Wikipedia
- Help

toolbox
- What links here
- Related changes
- Upload file
- Special pages
- Printable version
- Permanent link
- te this page

Scientific visualization

From Wikipedia, the free encyclopedia

Scientific visualization (also spelled **scientific visualisation**) is an interdisciplinary branch of science according to Friendly (2008) "primarily concerned with the **visualization** of three dimensional phenomena (architectural, meteorological, medical, biological, etc.), where the emphasis is on realistic renderings of volumes, surfaces, illumination sources, and so forth, perhaps with a dynamic (time) component".[2]



A scientific visualization of an extremely large simulation of a Rayleigh–Taylor instability caused by two mixing fluids.[1]

Contents [hide]

Scientific visualization topics [edit]

Computer animation

Scientific visualization - Wikipedia, the free encyclopedia - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://en.wikipedia.org/wiki/Scientific_visualization#In_the_natural_science

Facebook TigerTrails SomaFM Vish LEO MeetingMaker Translator HTML PHP Manua http://www.youtube.... Sciviz@CCT Forum sciviz phpMyAdmin Satellites Most Visited Lates...

Scientific visualization - Wikipedia, t...

Capture a window or desktop image

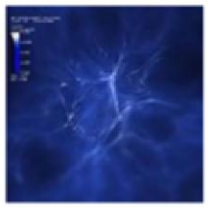
Press F1 for more help

large data storage. Advances in hardware and software are generalizing volume visualization as well as real time performances.[7]

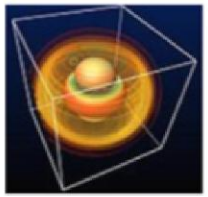
Scientific visualization applications [edit]

This section will give a series of examples how scientific visualization can be applied today.[8]

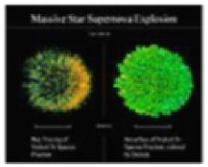
In the natural science [edit]



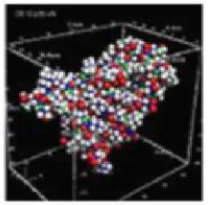
Star formation[9]



Gravity waves[10]



Massive Star Supernovae Explosions



Molecular rendering

Star formation: The featured plot is a Volume plot of the logarithm of gas/dust density in an Enzo star and galaxy simulation. Regions of high density are white while less dense regions are more blue and also more transparent.

Gravity waves: Researchers used the Globus Toolkit to harness the power of multiple supercomputers to simulate the gravitational effects of black-hole collisions.

Massive Star Supernovae Explosions: In the image three Dimensional Radiation Hydrodynamics Calculations of Massive Star Supernovae Explosions The DJEHUTY stellar evolution code was used to calculate the explosion of SN 1987A model in three dimensions.

Molecular rendering: Vislt's general plotting capabilities were used to create the molecular rendering shown in the featured visualization. The original data was taken from the Protein Data Bank and turned into a VTK file before rendering.

In geography and ecology [edit]

Data Visualization

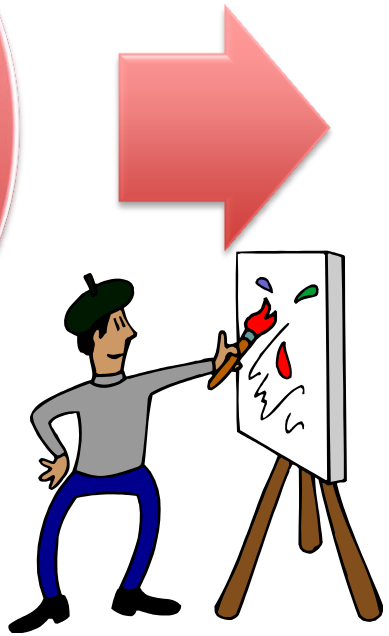
The main goal of data visualization is to communicate information clearly and effectively through graphical means. It doesn't mean that data visualization needs to look boring to be functional or extremely sophisticated to look beautiful. To convey ideas effectively, both aesthetic form and functionality need to go hand in hand, providing insights into a rather sparse and complex data set by communicating its key aspects in a more intuitive way. Yet designers often fail to achieve a balance between design and function, creating gorgeous data visualizations which fail to serve their main purpose — *to communicate information*.

Friedman, 2008

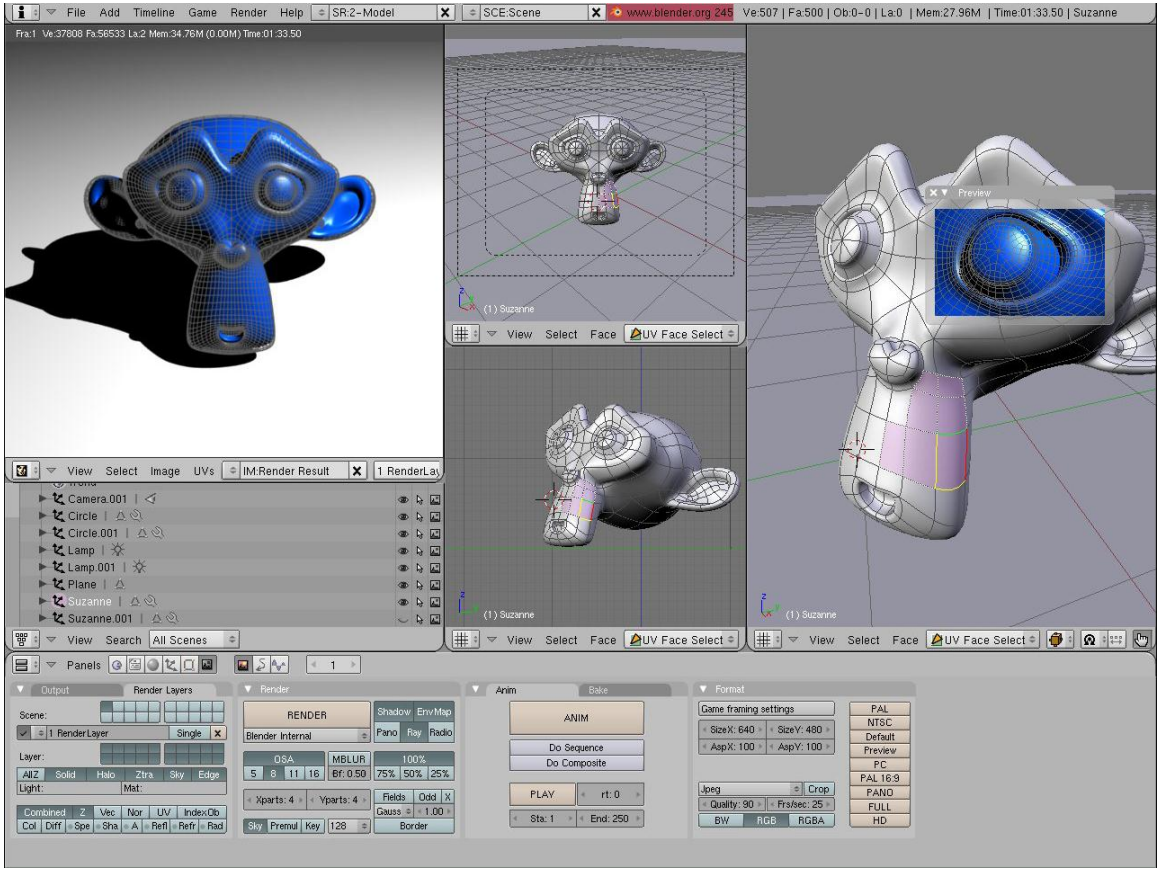
http://en.wikipedia.org/wiki/Data_visualization

Computer Graphics

Artist



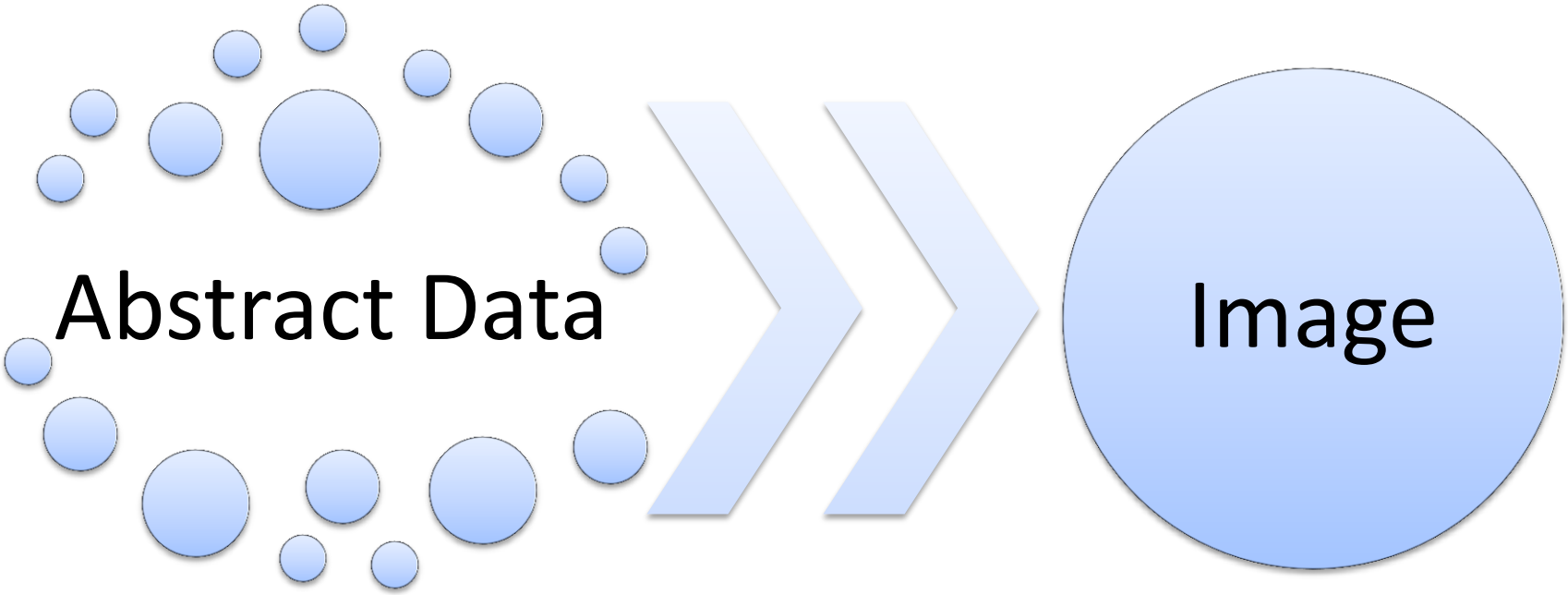
Image

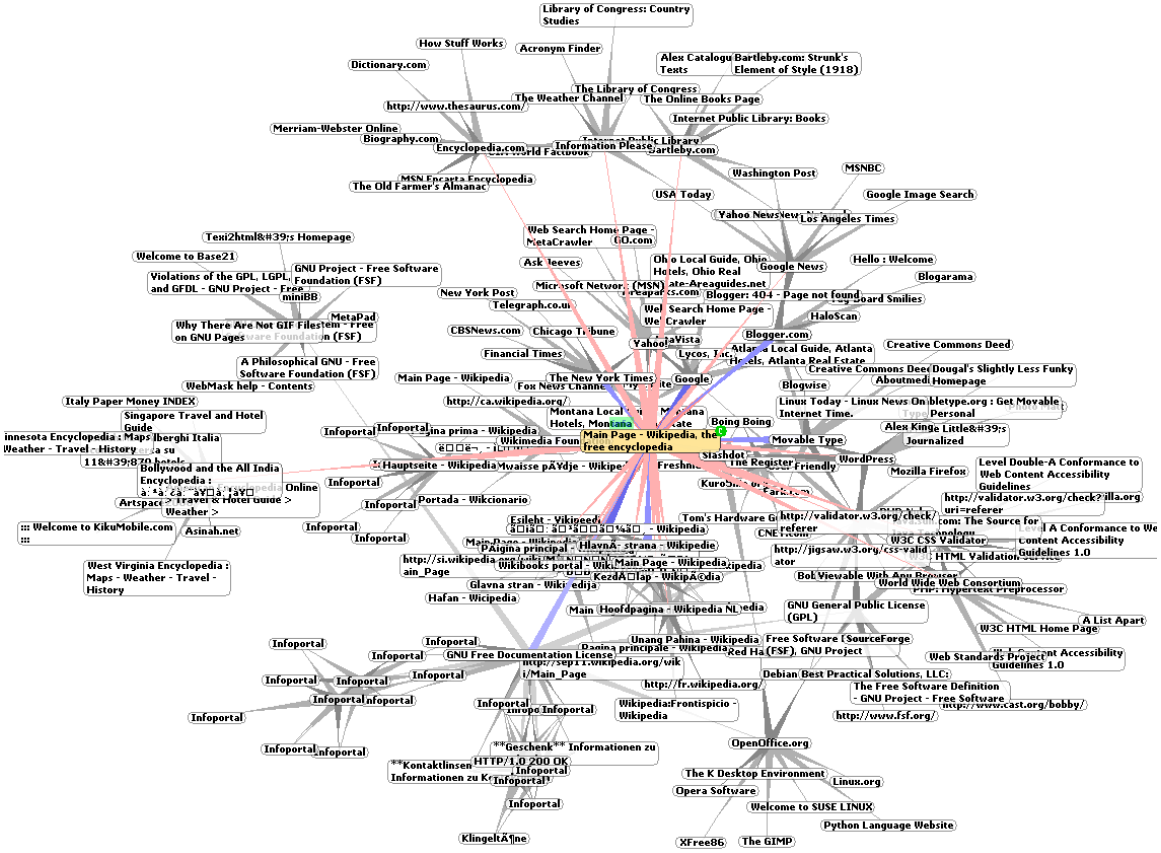


Computer Graphics: Modeling Data

Source: http://en.wikipedia.org/wiki/Computer_graphics

Information Visualization (InfoViz)

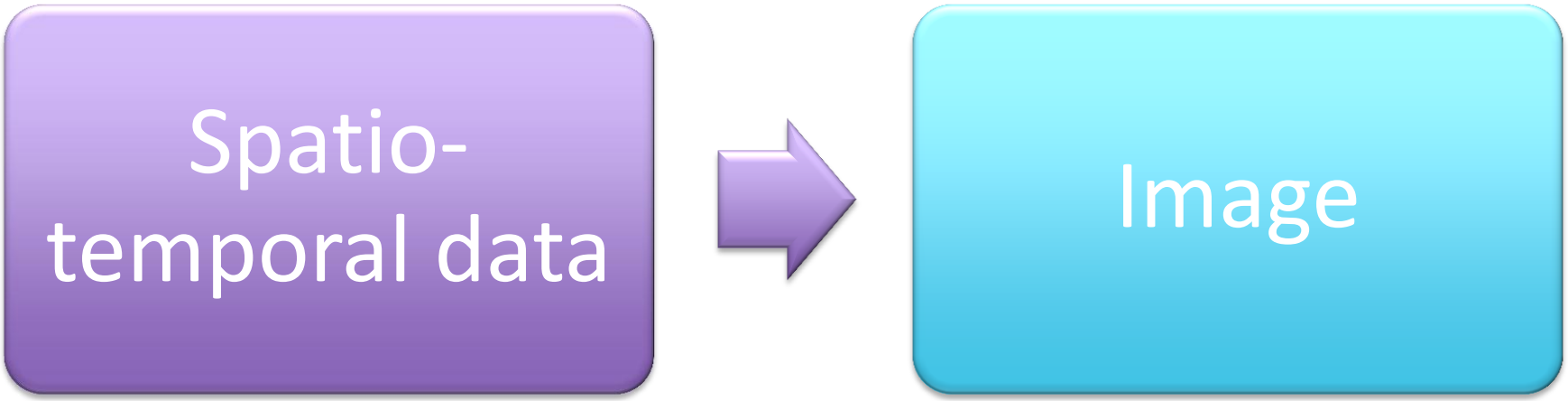




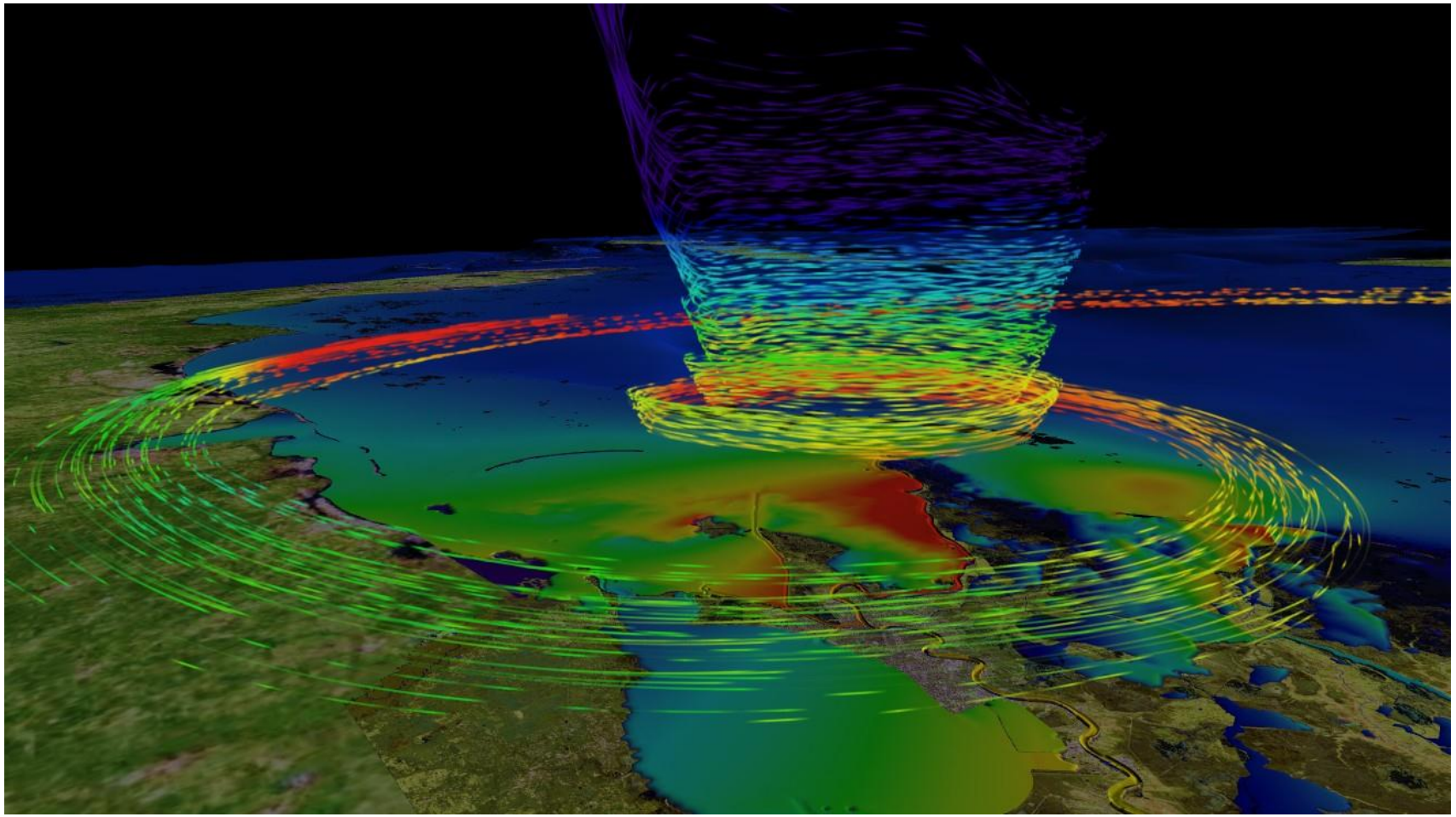
InfoViz: Visualization of Abstract Data

Source: http://en.wikipedia.org/wiki/Information_visualization

Scientific Visualization (SciViz)

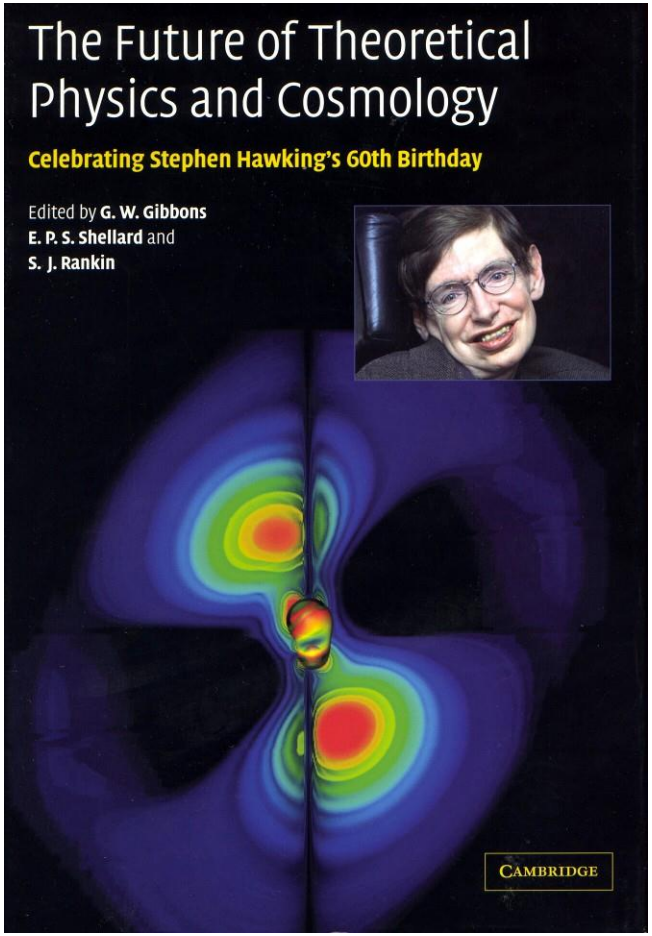


SciViz Example: Hurricane Katrina

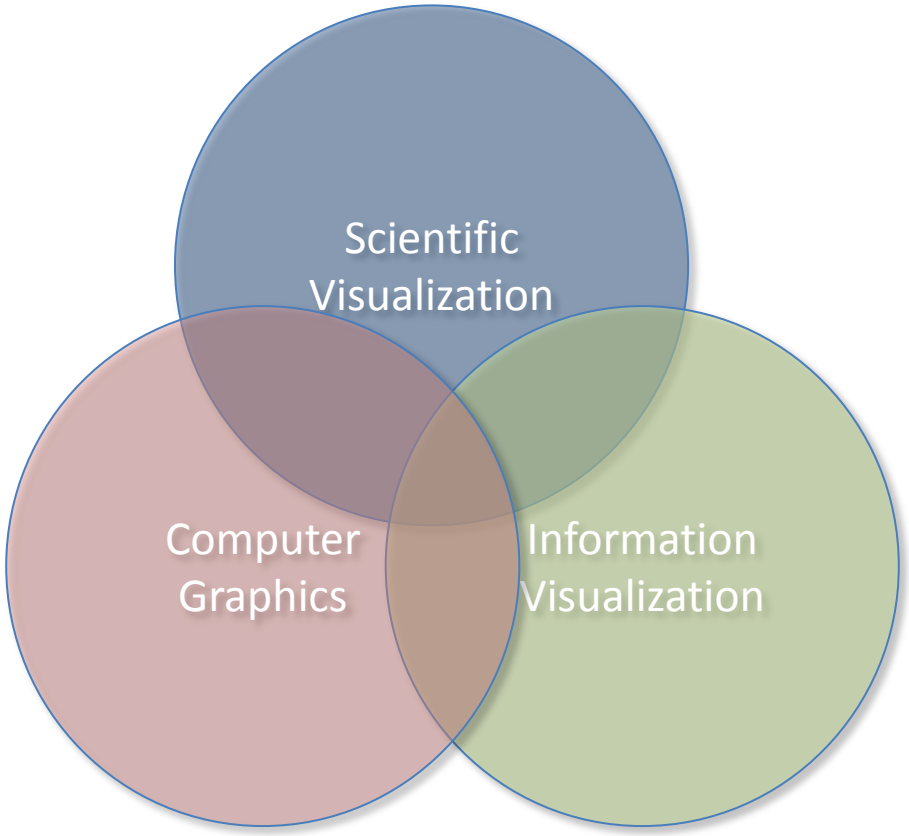


Example:
Scientific Visualization of Gravitational Waves

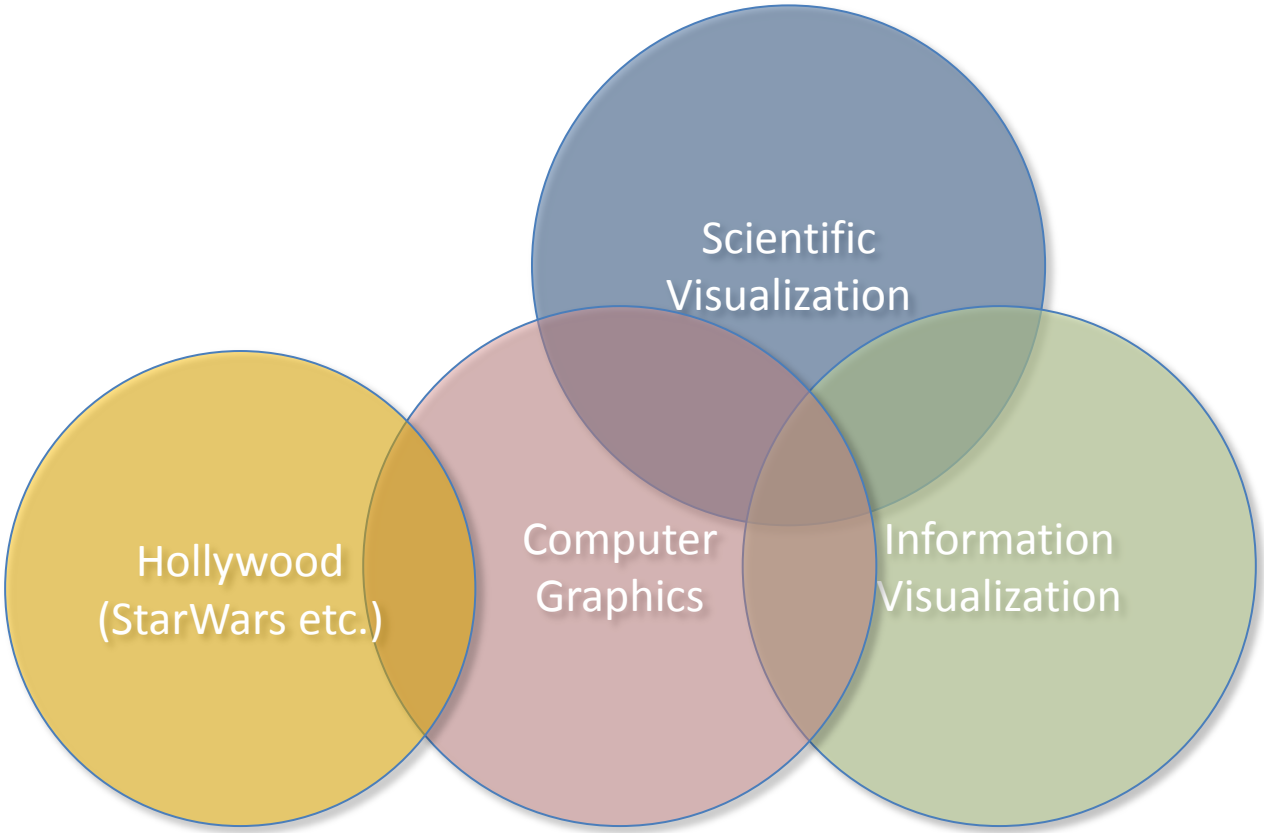
Visually Communicate Scientific Content of physical phenomena



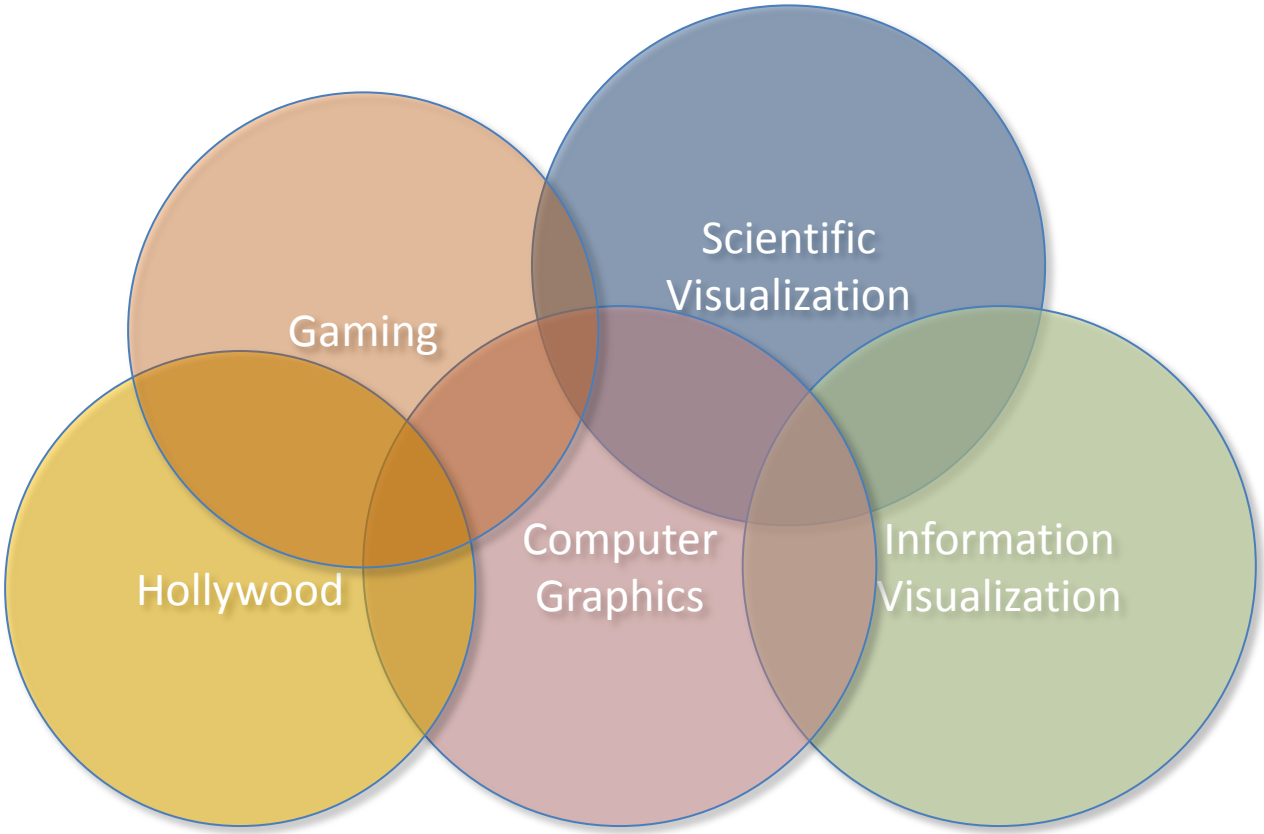
Visualization & Computer Graphics



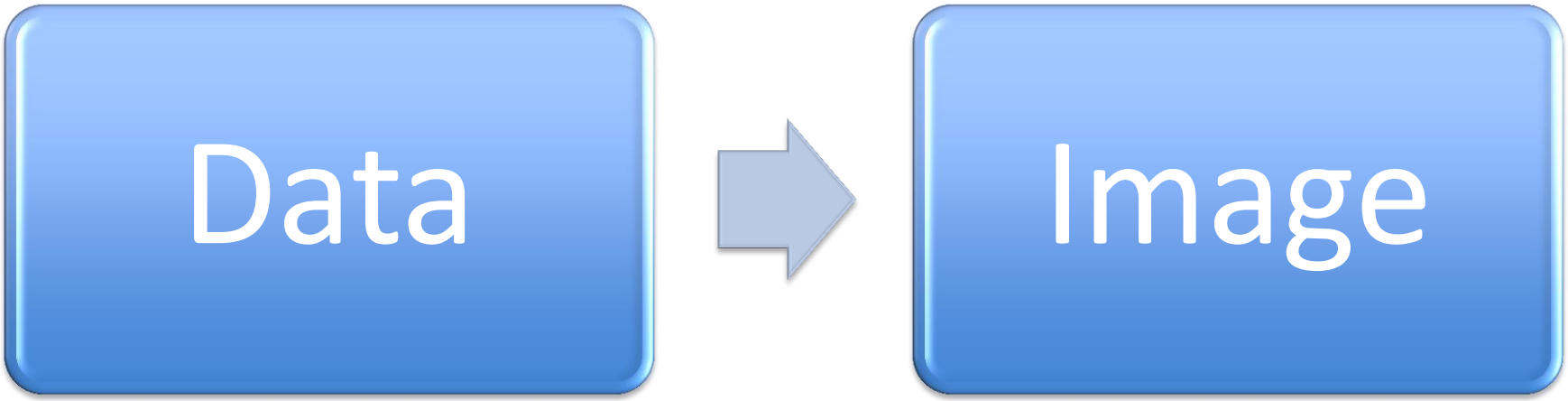
Visualization & Multimedia



Visualization & Gaming



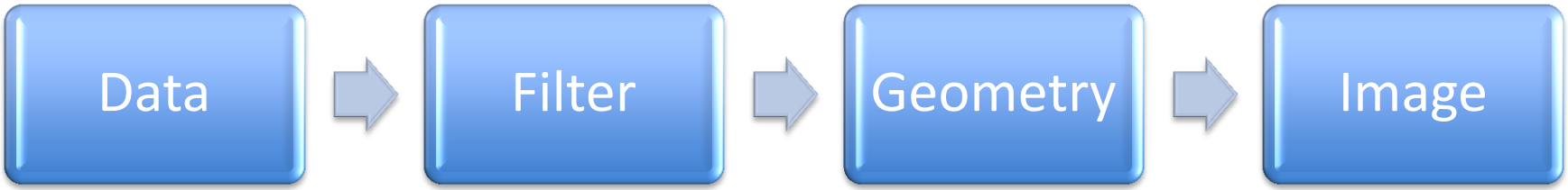
Data Visualization



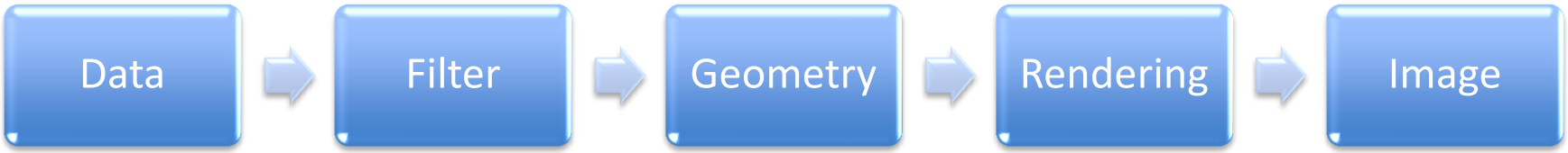
Visualization Pipeline

A concept of how visualization works
Haber & McNabb, 1990

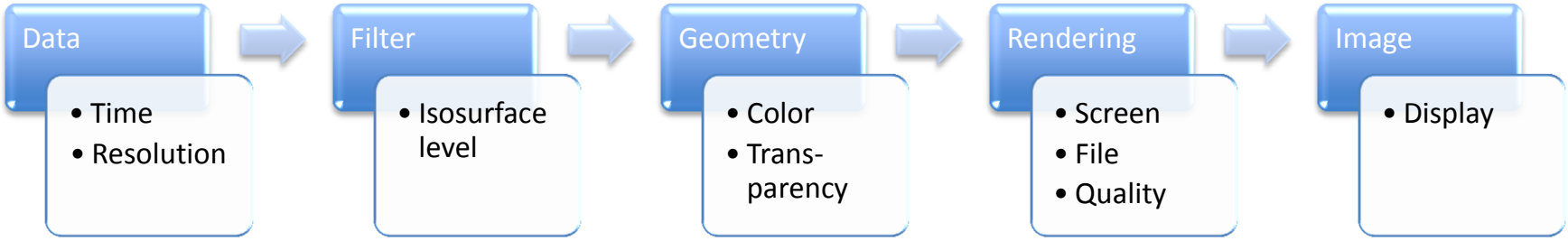
Visualization Pipeline



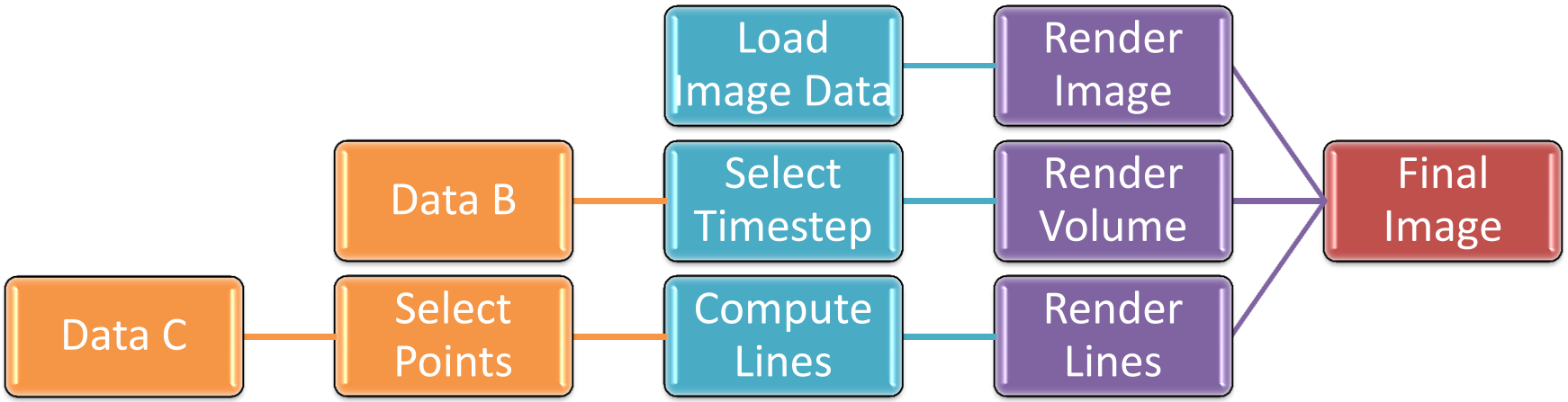
Visualization Pipeline



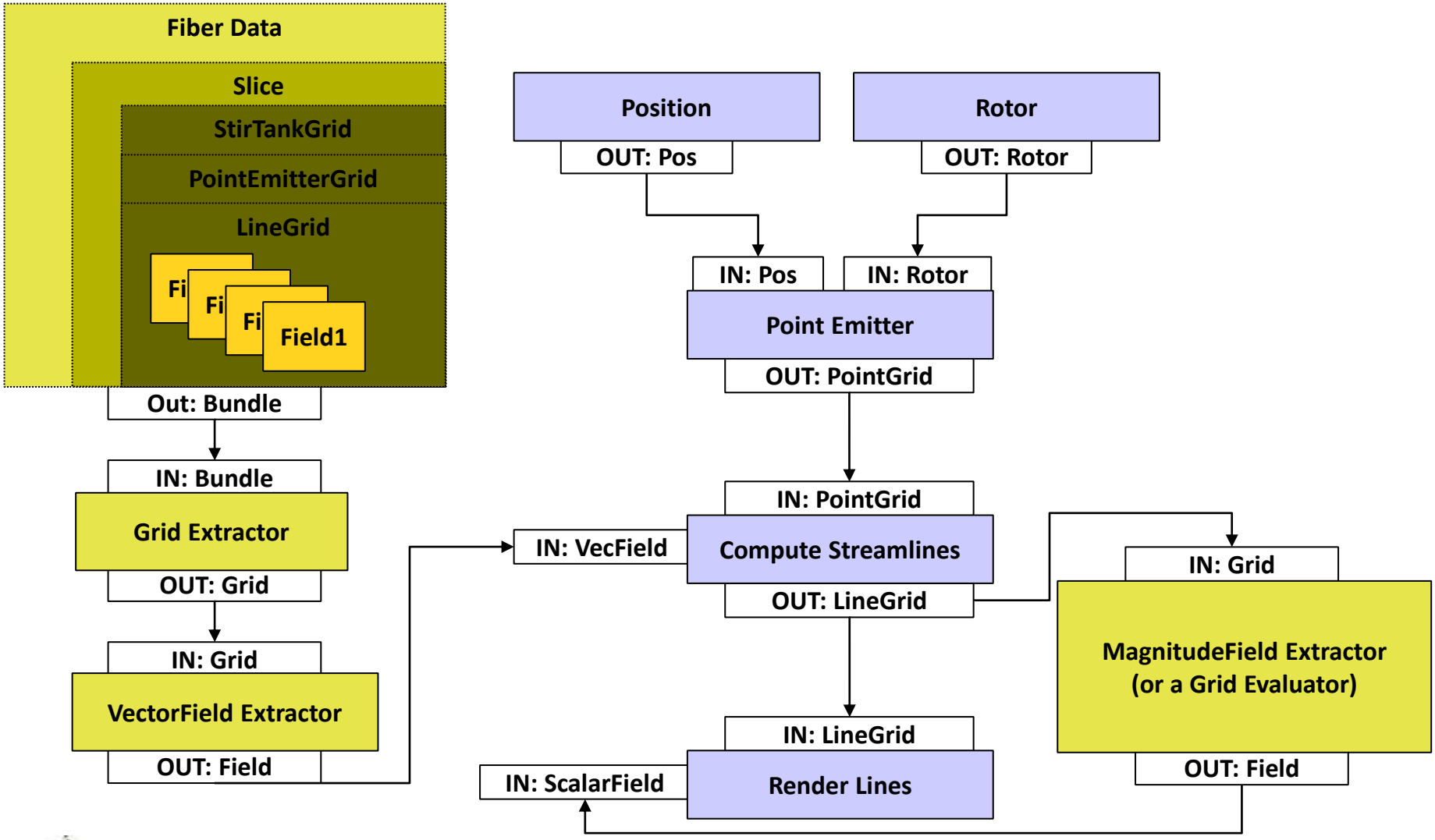
Visualization Pipeline Parameters



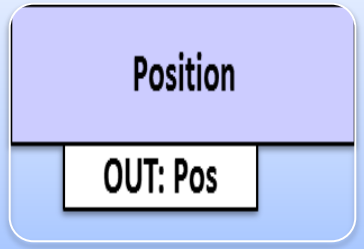
Visualization Network



Complex Viz Network



“Atomic Elements” of a Viz Network

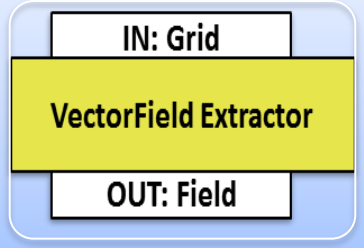


Data Sources

- Only output

Data Sink

- Only input



Data Filters

- Input and outputs

Examples

Data Sources	Data Sinks	Data Filters
• File • Network stream • Numerical computation • …	• Display • File • Printer • Network • …	• Timestep • Subsampling • Numerical Operation • …

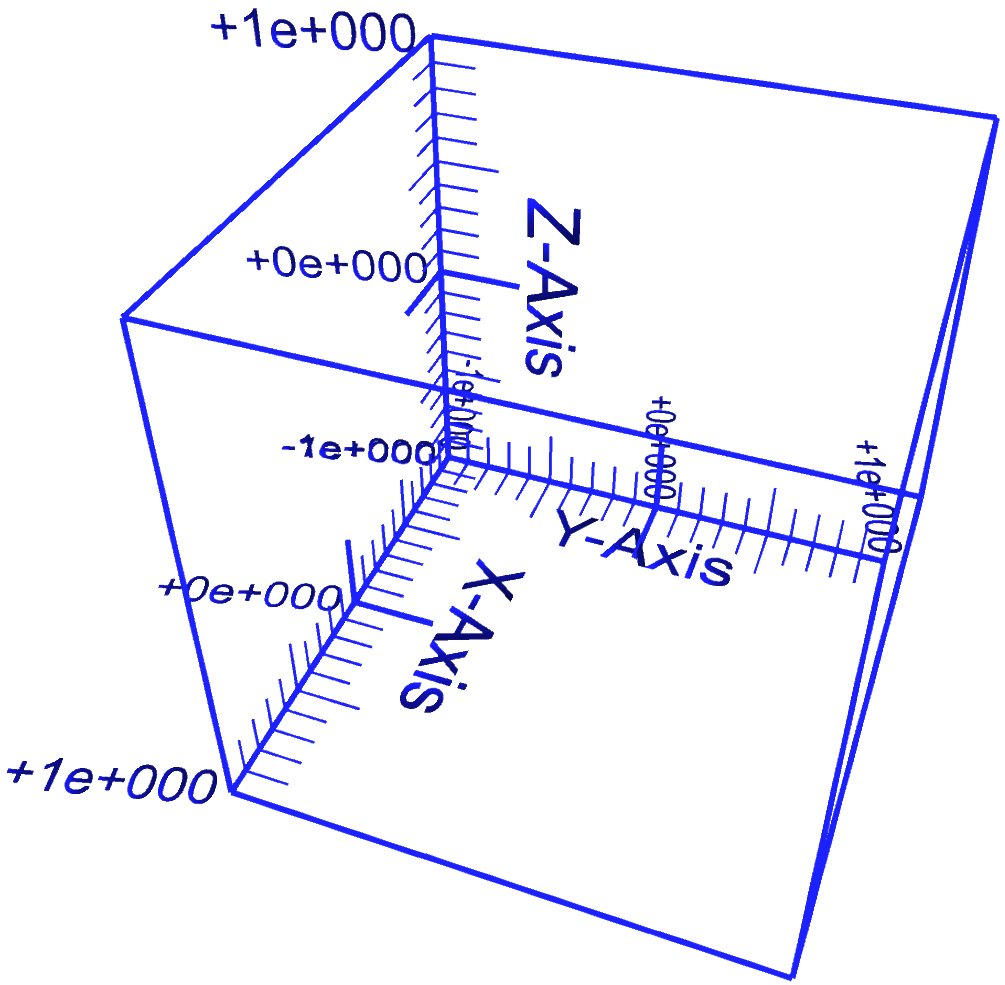
Application Example

Given a box with (Cartesian) coordinates

(-1,-1,-1) - (+1,+1,+1)

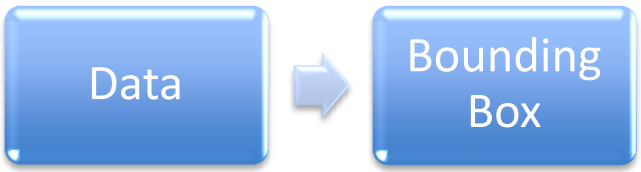
Given a scalar field in a volume (box)

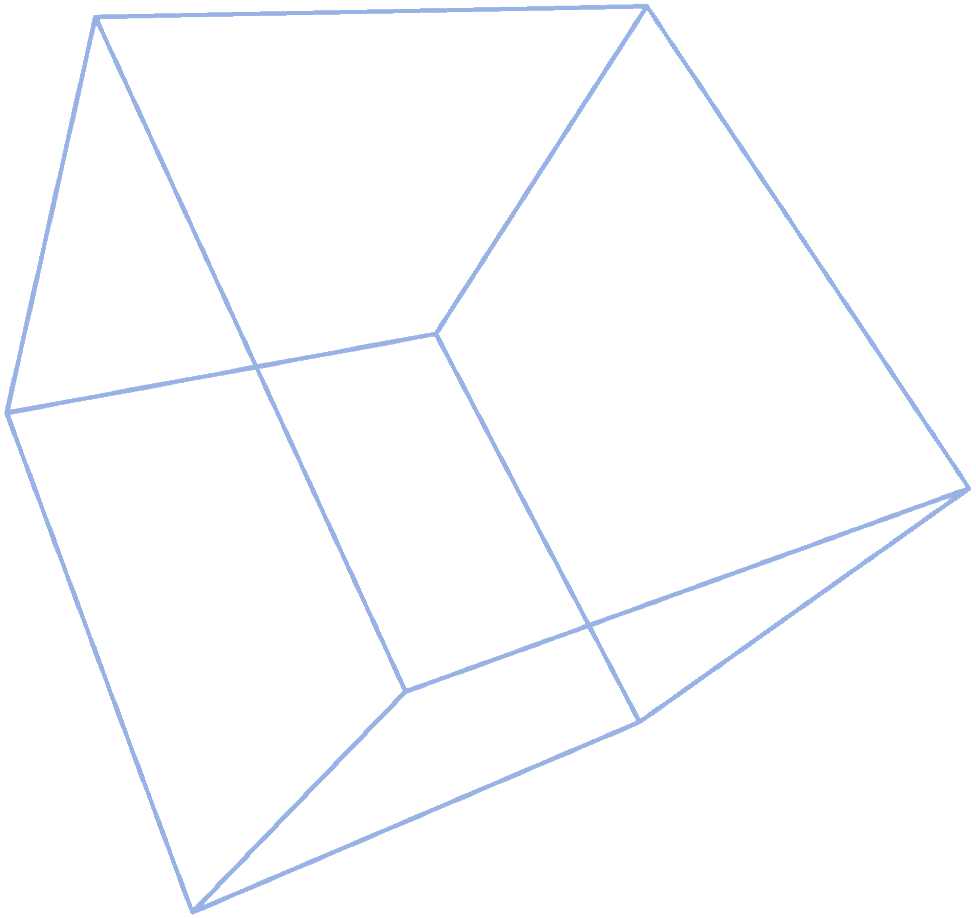
$f(x,y,z) = x^2 + y^2 + z^2$



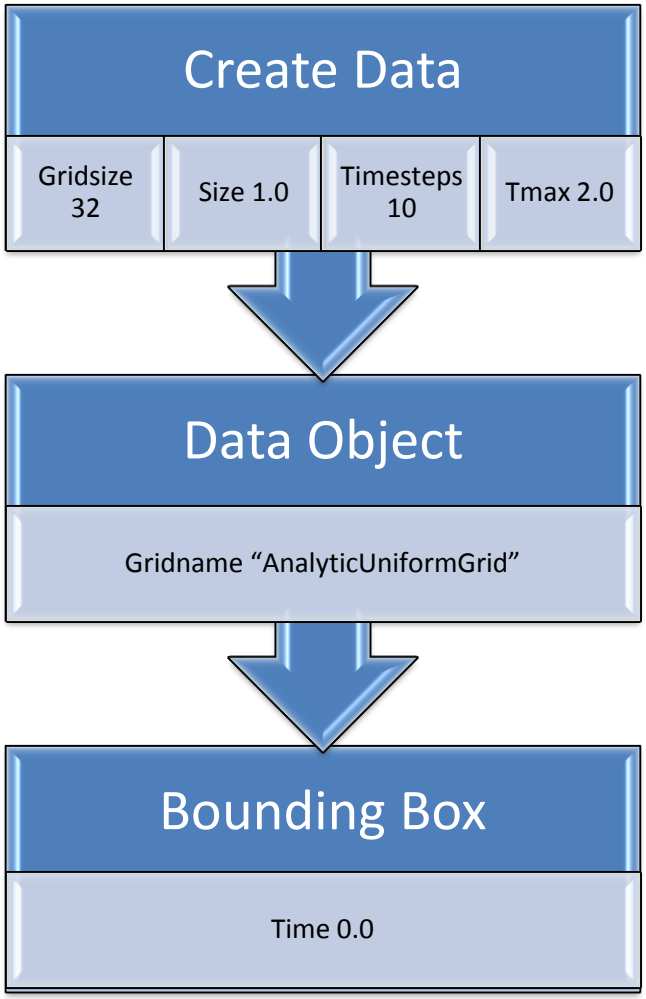
"Visualize it"

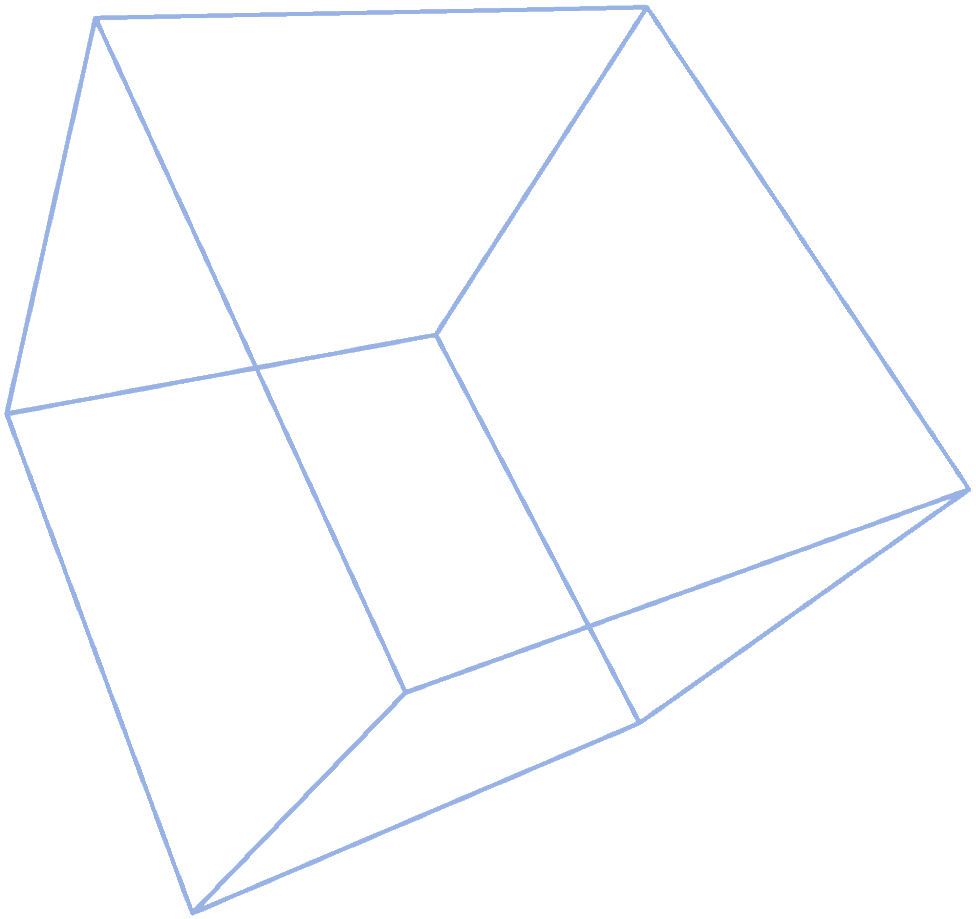
First Step: Bounding Box



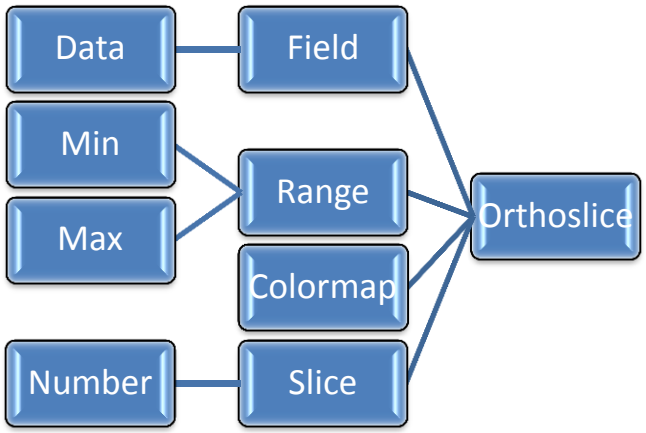


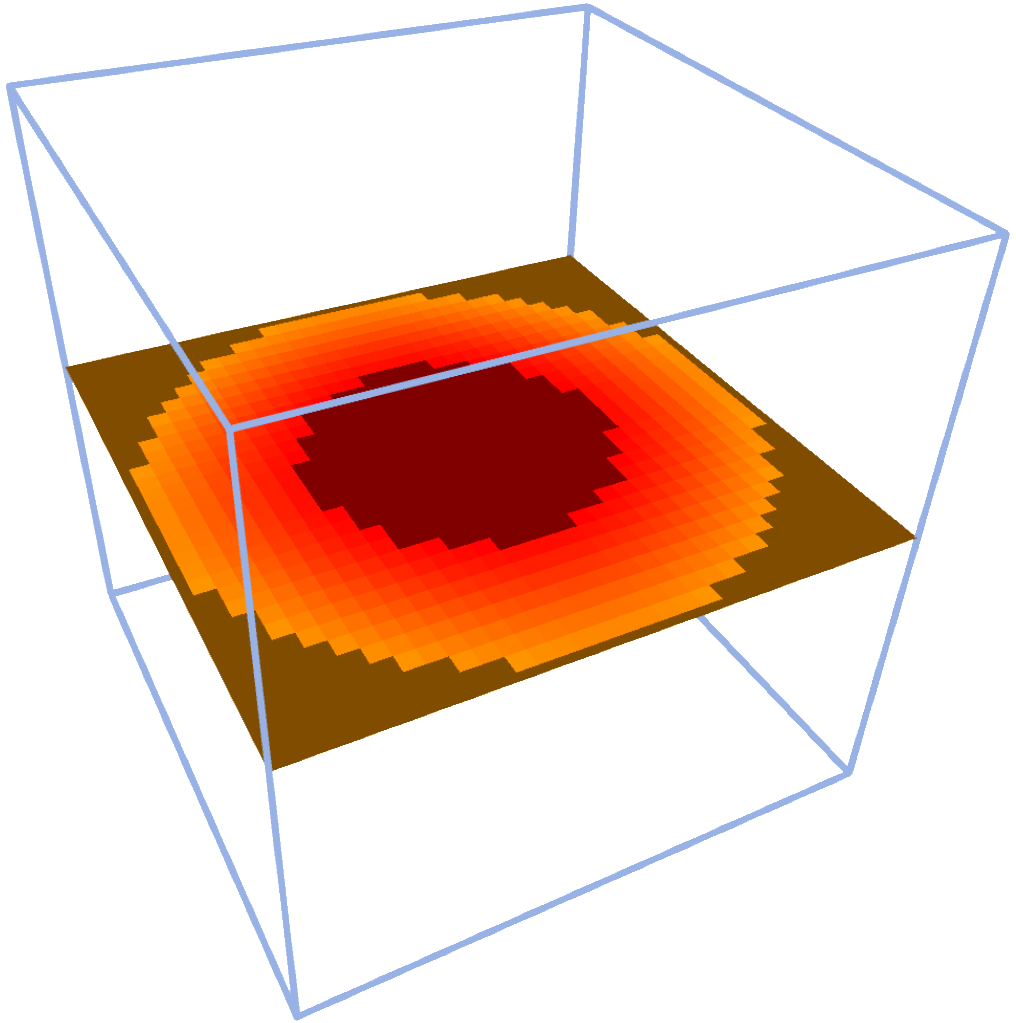
In Practice:


Create Data
Gridsize 32
Size 1.0
Timesteps 10
Tmax 2.0
Data Object
Gridname “AnalyticUniformGrid”
Bounding Box
Time 0.0

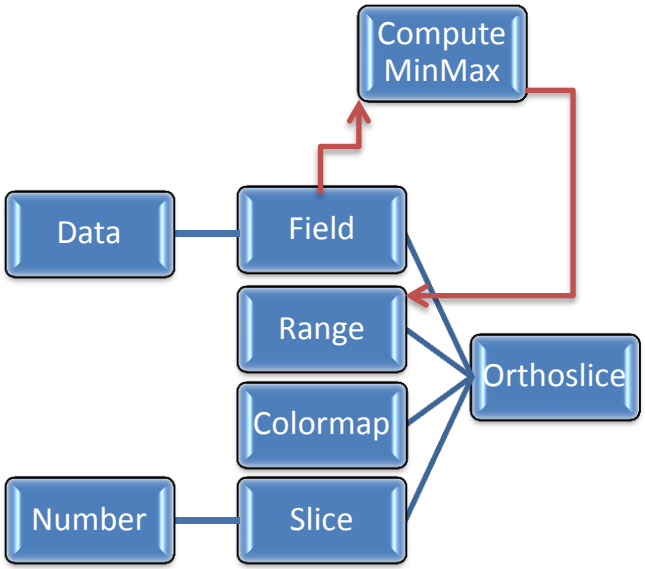


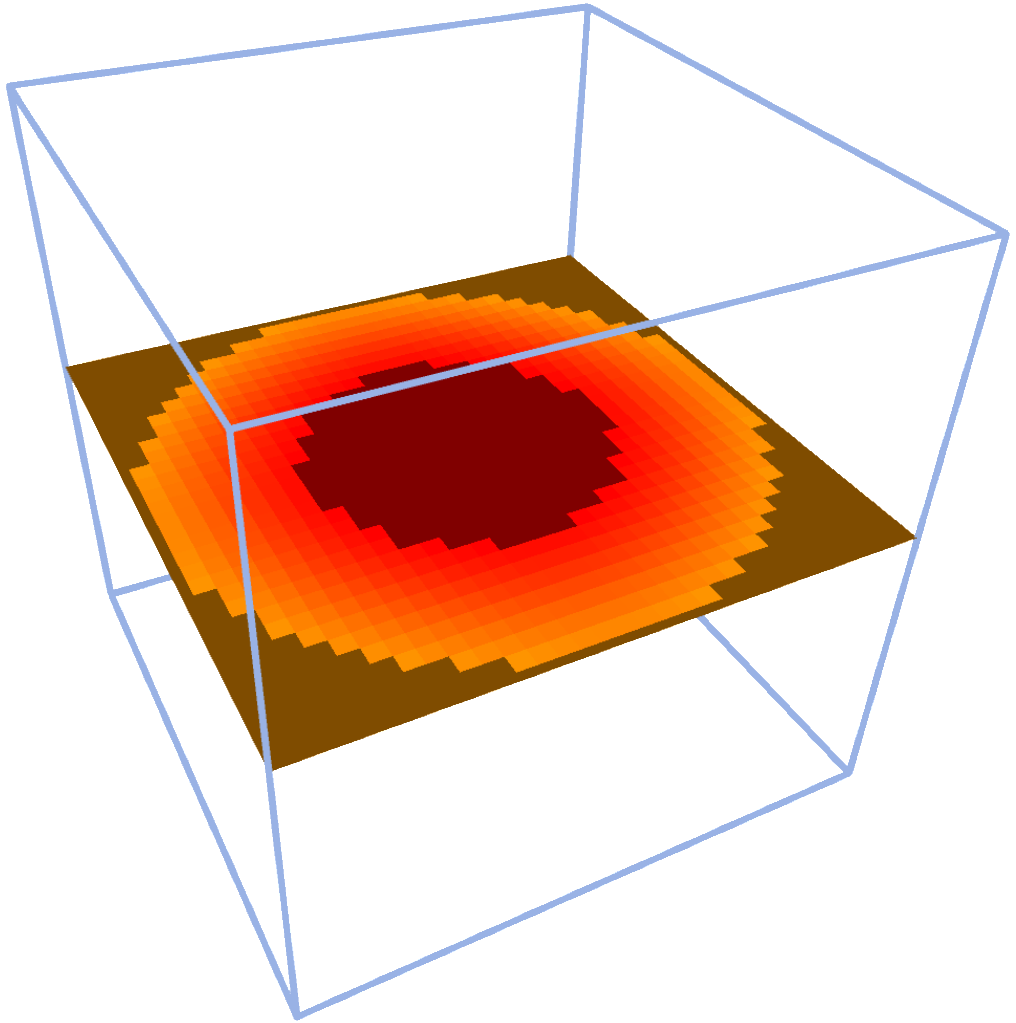
Display a Slice



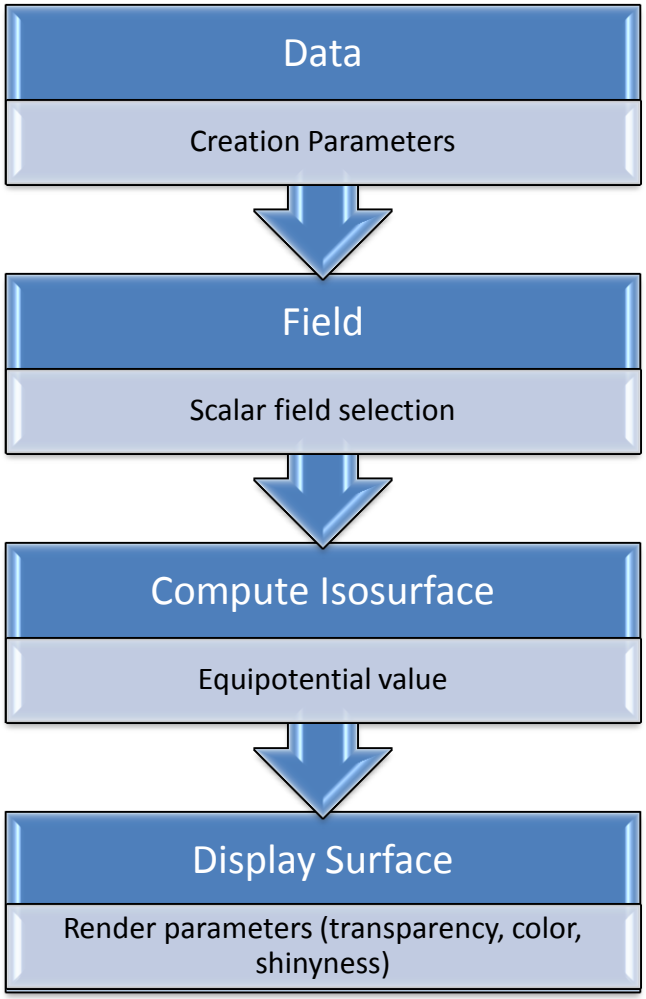


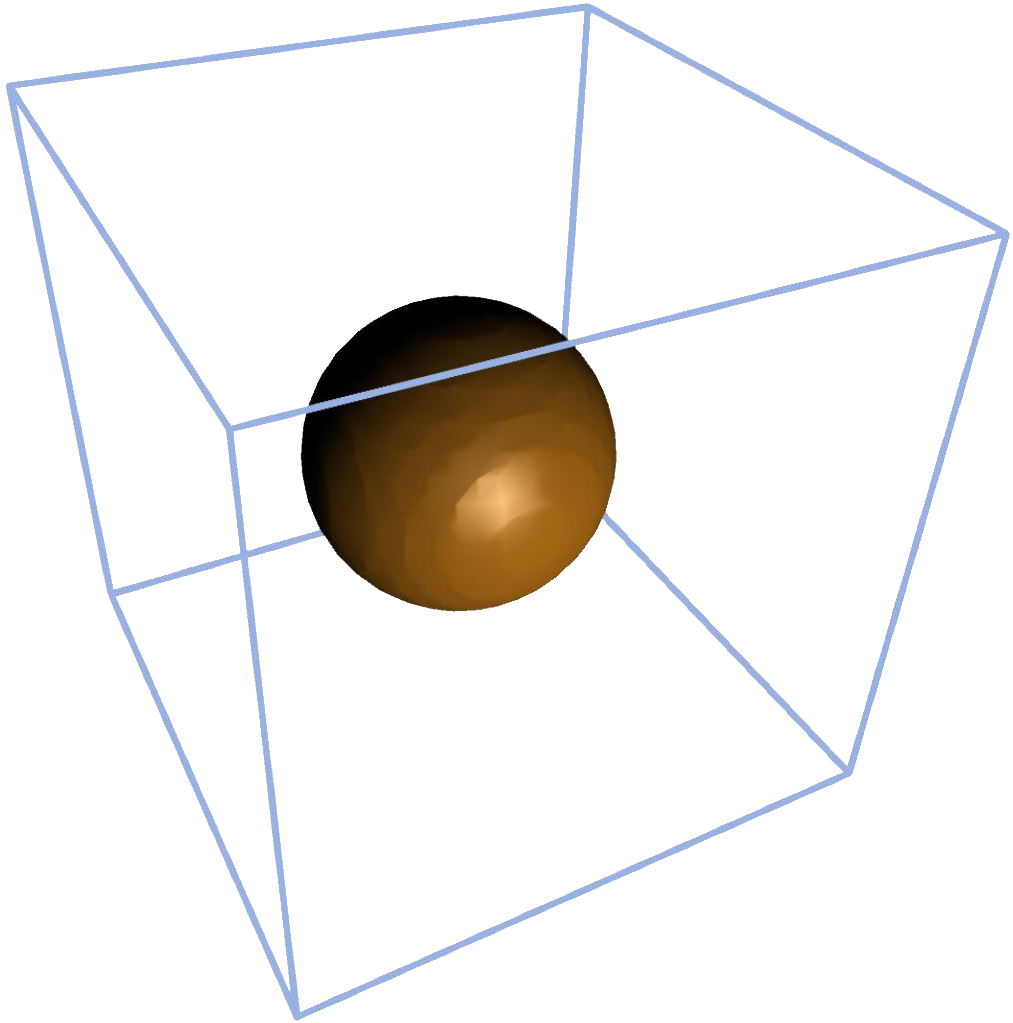
Display a Slice



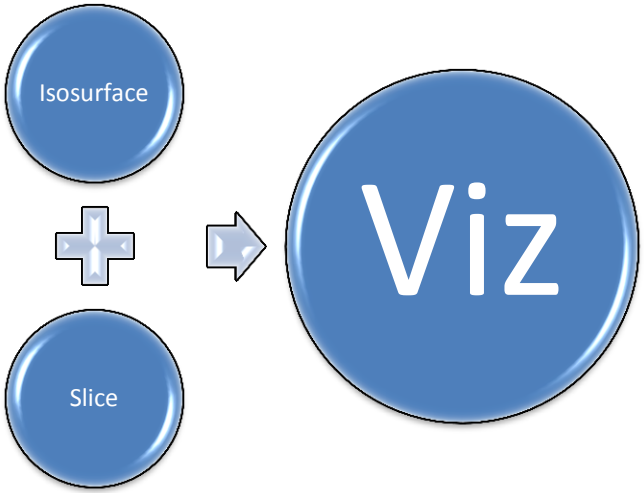


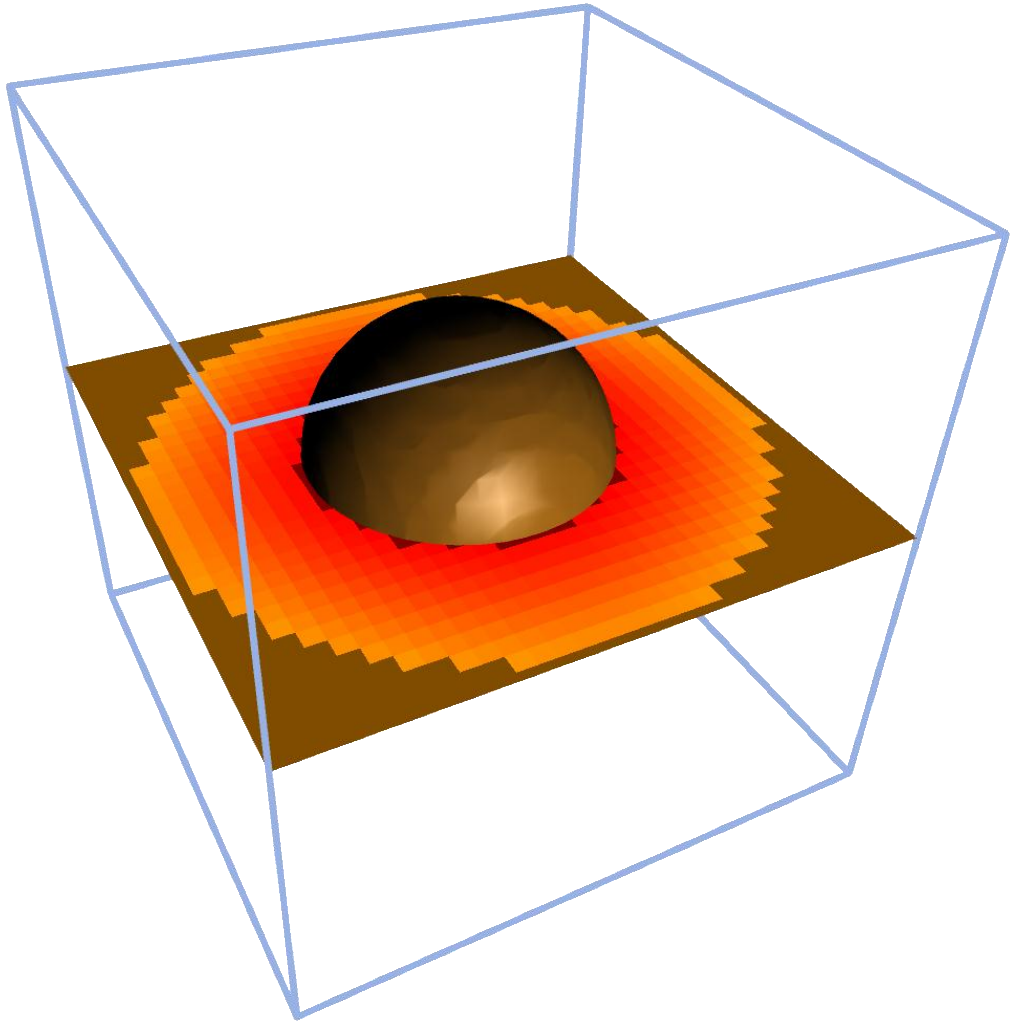
Display an Isosurface (Equipotential surface)





Combined Visualization





Building Visualization Networks

Built from elementary objects

- Data Sink
- Data Filter
- Data Source

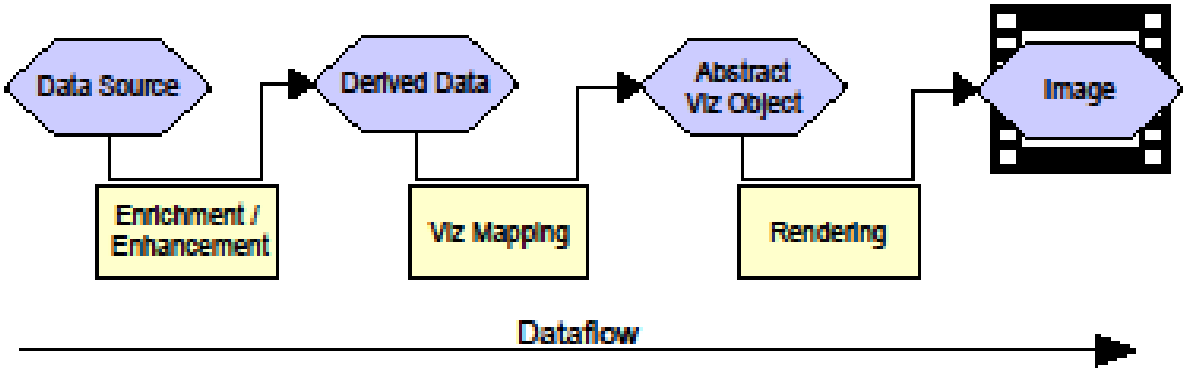
Forms a graph, communication via *types*

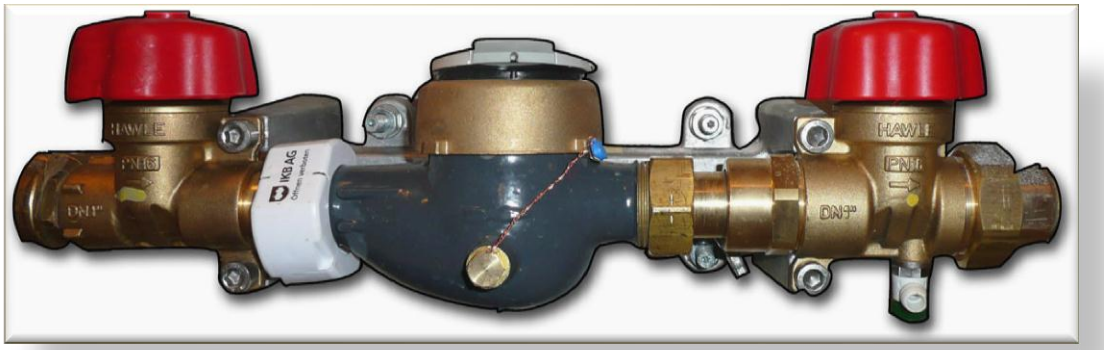
- One or multiple inputs
- One or multiple outputs
- Usually not cyclic

Objects have parameters

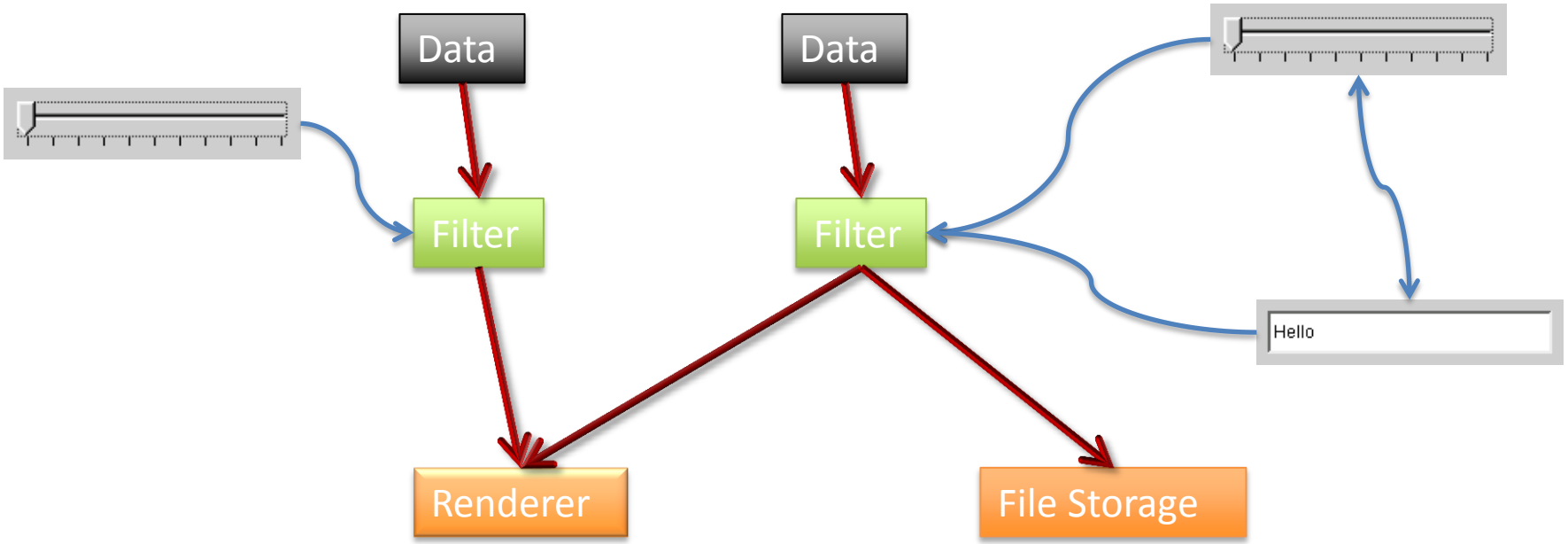
- Controls Properties of objects
  - int, float, 3D points, …

Data flow in a vis network





Interacting with a Viz Network



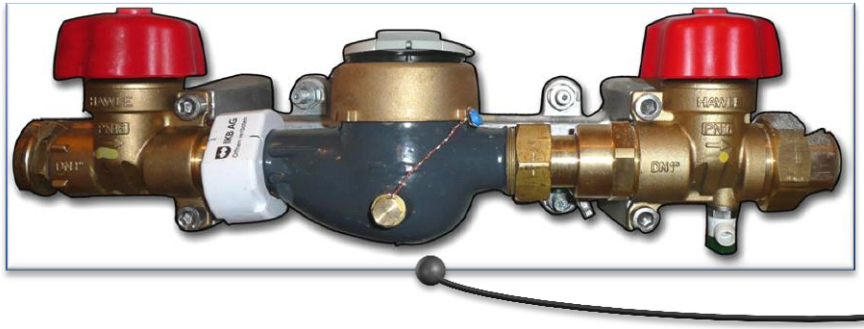
Data Flow vs. Control Flow

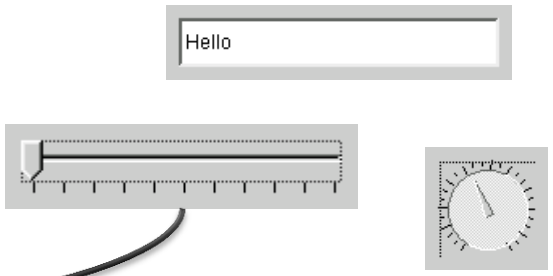
Data Flow

- Large data (volumetric data, meshes, images)
- Filtered and reduced to finally yield Pixel
- May be asynchronous with user interaction
- Can be complex
- Central to visualization software

Control Flow

- Small parameters
- Control elements of a GUI (sliders, buttons, …)
- Synchronous with user interaction
- Much simpler graph
- Often implicit to visualization software





Different ways on how to traverse a visualization network and user interaction

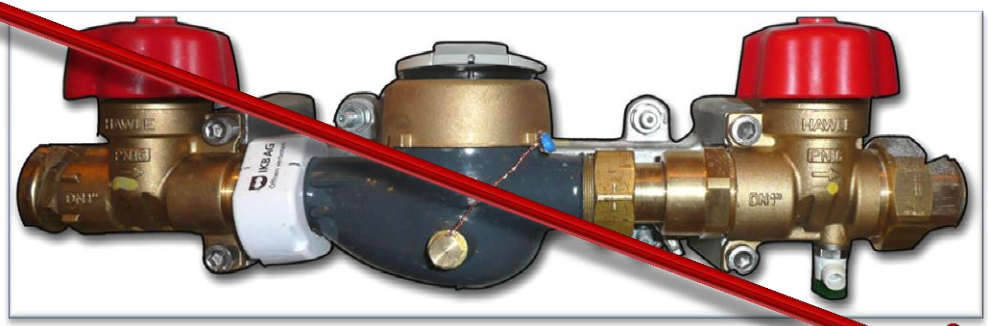
UPSTREAM VS. DOWNSTREAM

Push Model

- Data at the source is pushed downstream through filters to the data sink

Action

File reading,
network transfer,
computation, …

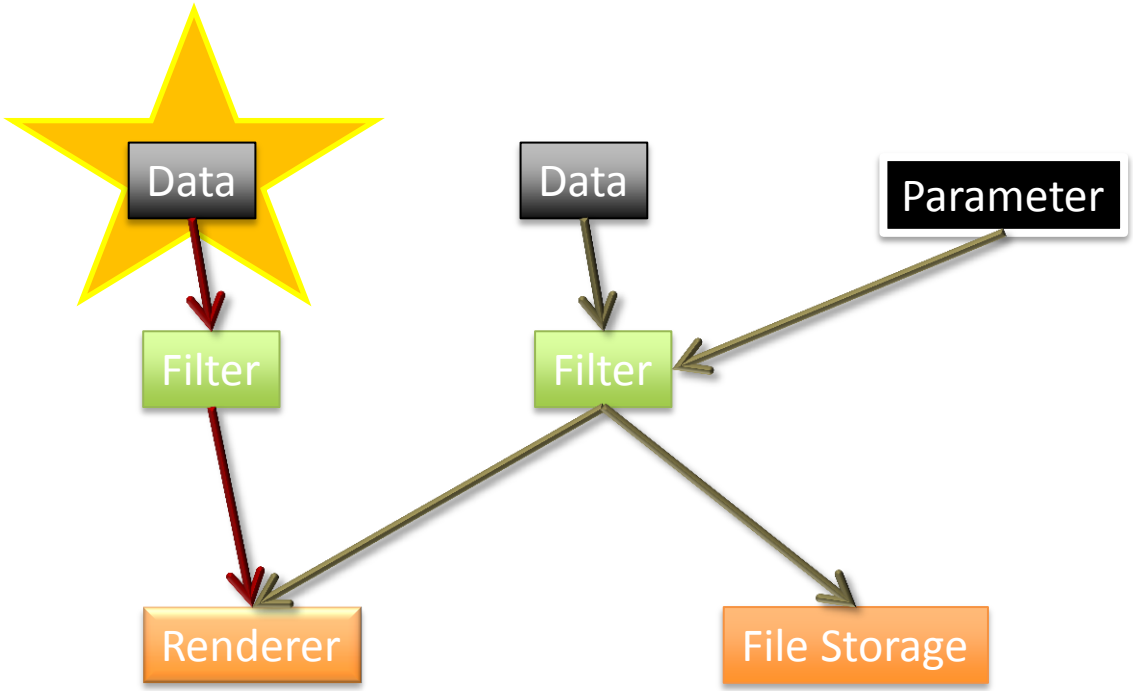


Result

Image Rendering

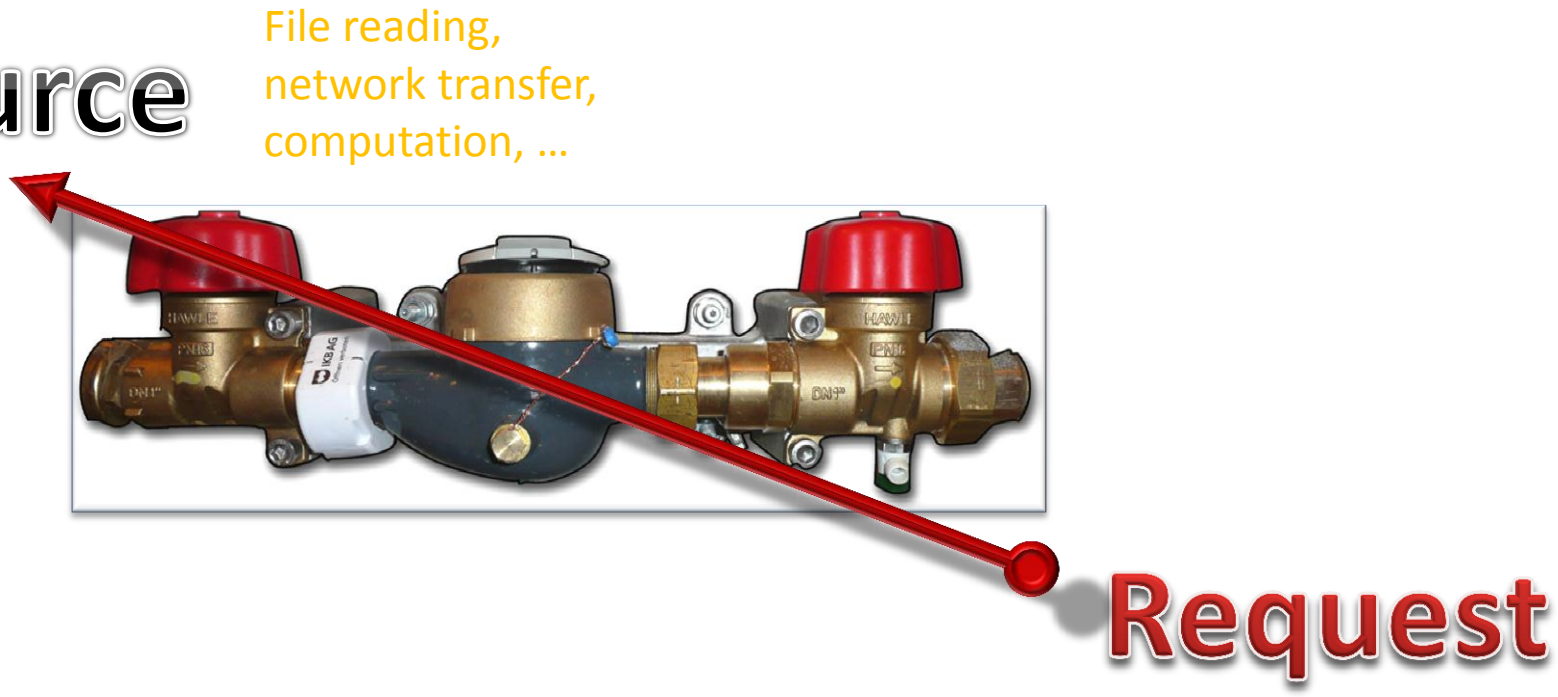
Push Model (downstream)

- AVS, Amira, …



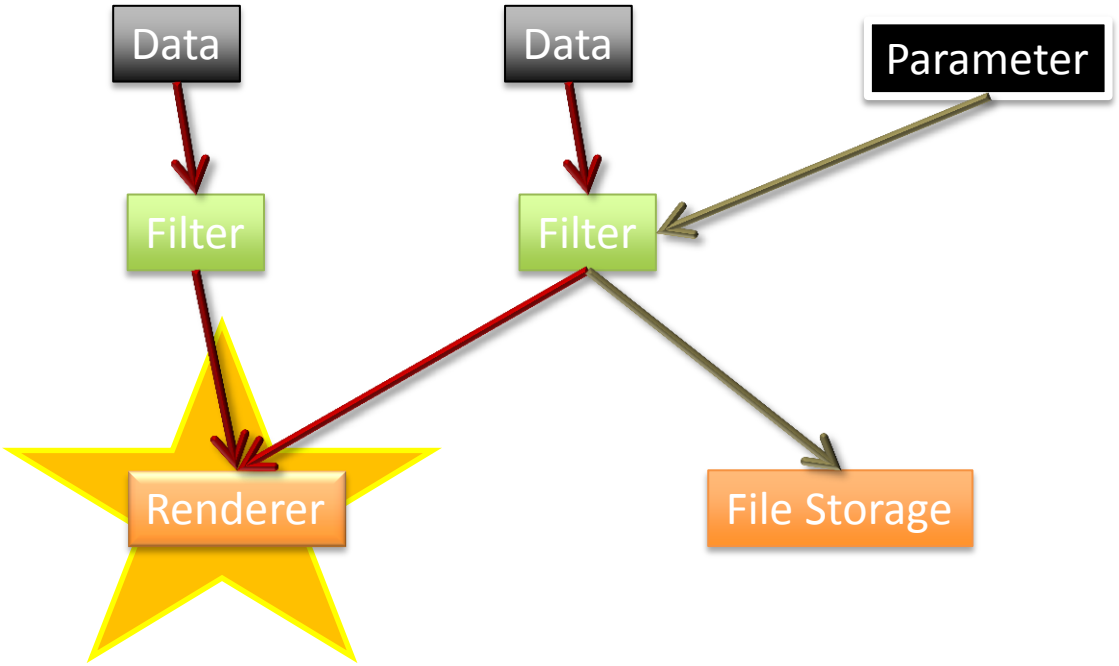
Pull Model

- Data sink requests run upstream through filters requesting data from the source source



Pull Model (upstream)

- OpenDX, VTK, Vish, …



Push vs. Pull

Push model

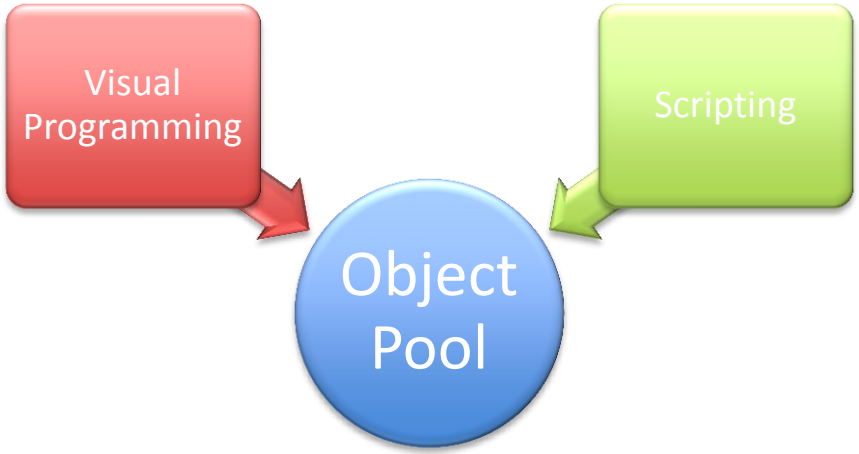
- Data are made available as soon as possible
- Traverses viz pipeline after loading/creation
- Filter modules have information about data available at early stage
- Loads data even if not used

Pull model

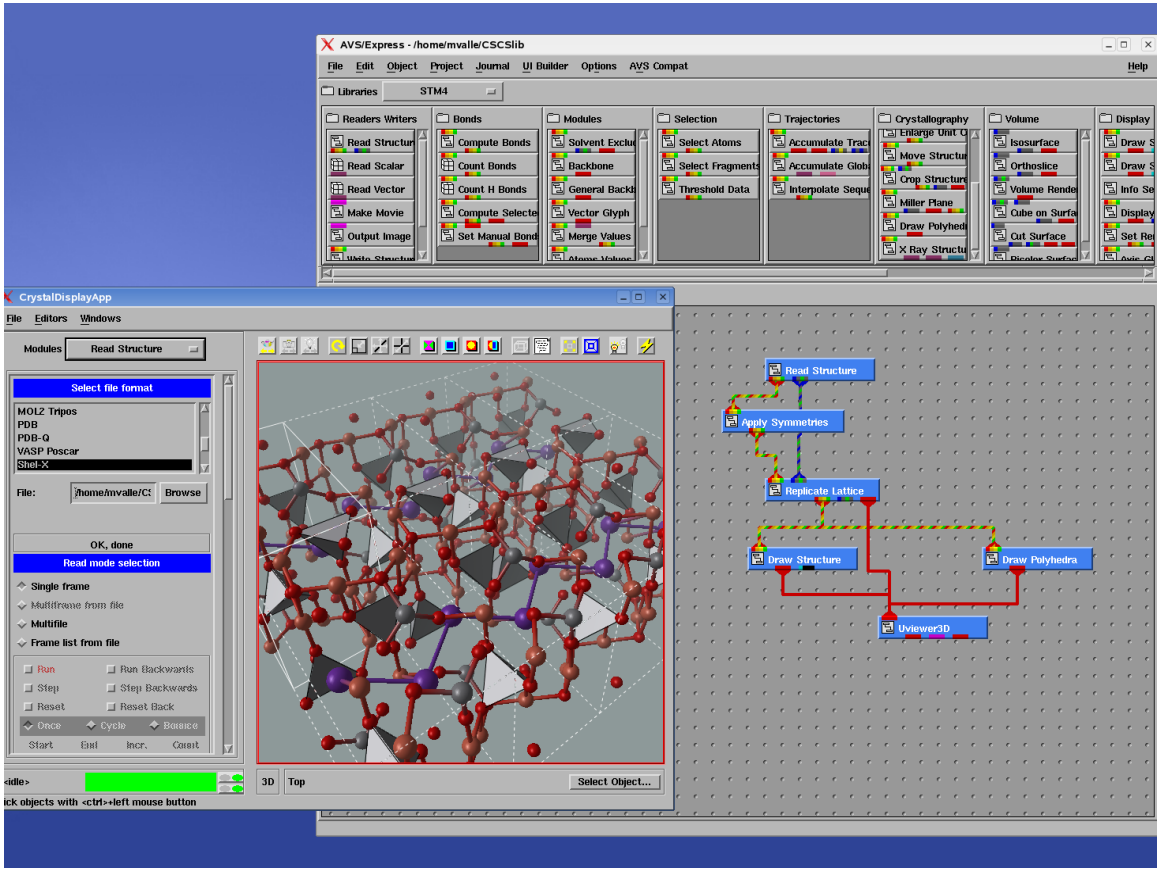
- Data are made available as late as possible
- Traverses viz pipeline at rendering time
- Filter modules don't know all about data until output is requested
- Loads only data when used

Visual Programming

- Interacting with Visualization Networks
- Alternative to Scripting
- Provided by most visualization software
- Create Viz network graphically, store as script
- Create Viz network by a script, display visualization network



EXAMPLES OF VISUALIZATION SOFTWARE

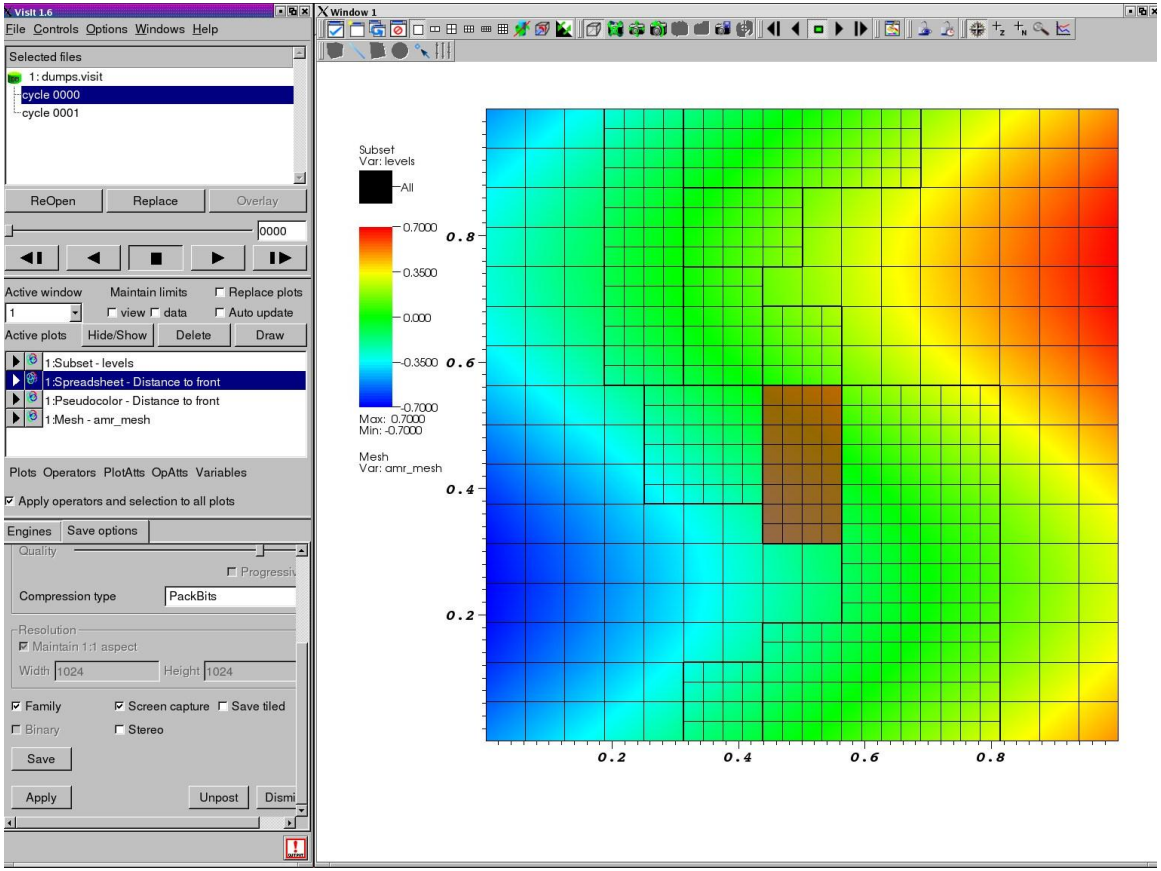


AVS/Express

http://personal.cscs.ch/~mvalle/STM4/

http://www.avs.com/software/soft_t/examples.html

Commercial

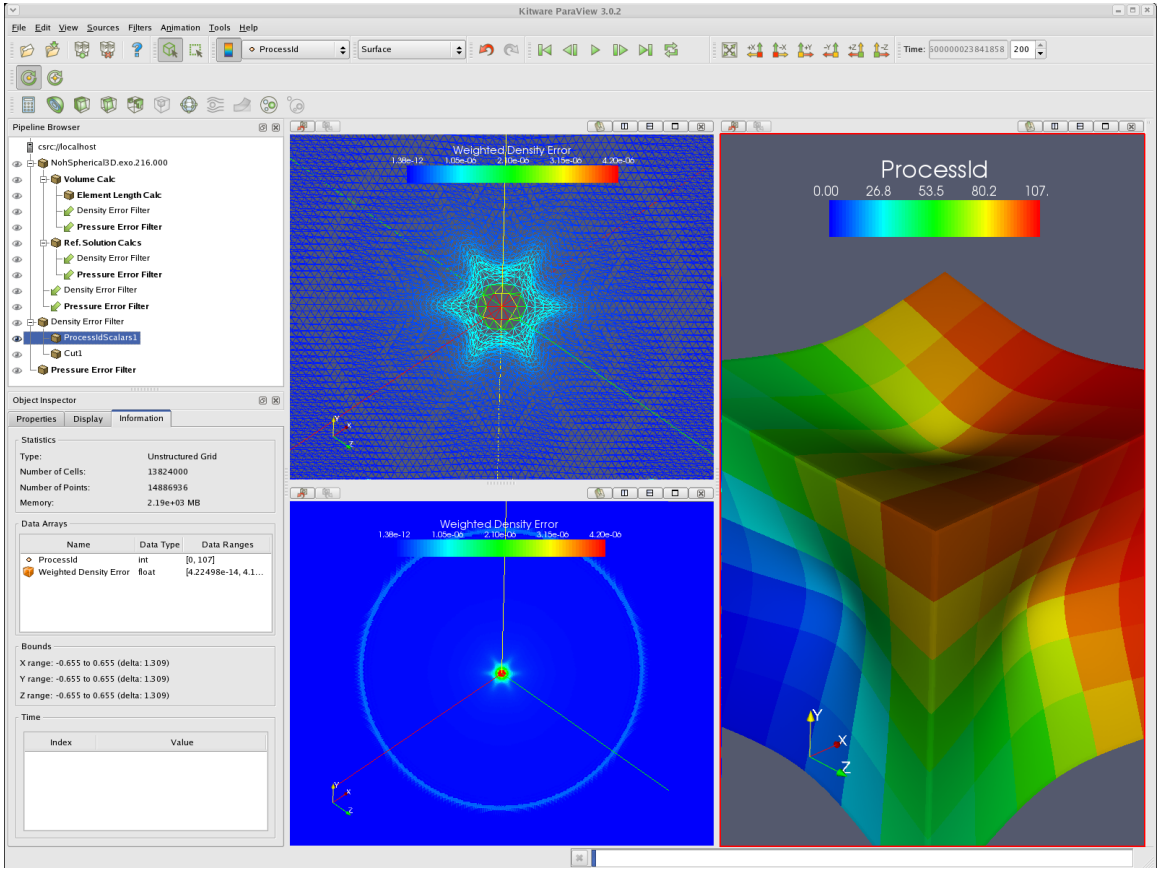


Visit

https://wci.llnl.gov/codes/visit/

Base on VTK (Visualization Toolkit), OpenSource

Image courtesy of LLNL

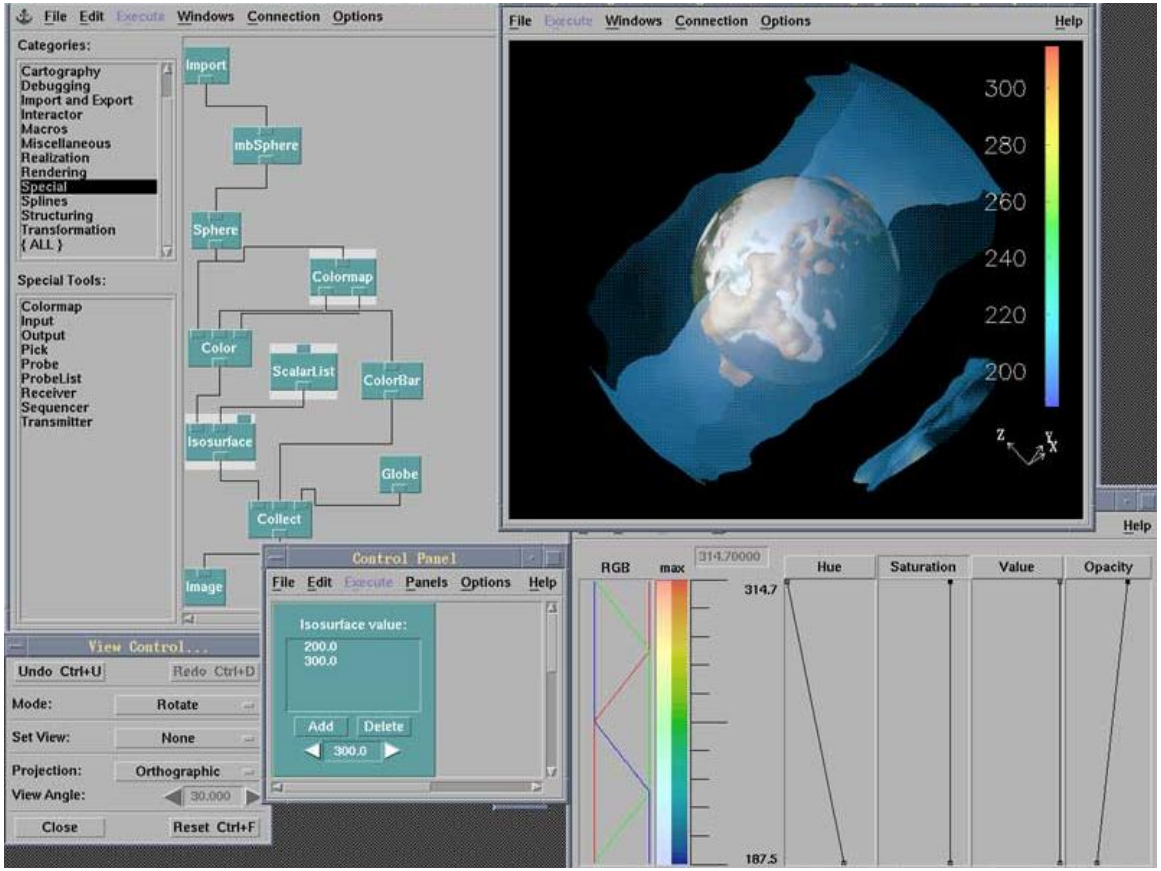


Paraview

http://www.paraview.org/

Based on VTK (Visualization Toolkit, Kitware.com), Open Source, Commercial extensions

Image courtesy of Sandia National Labs.

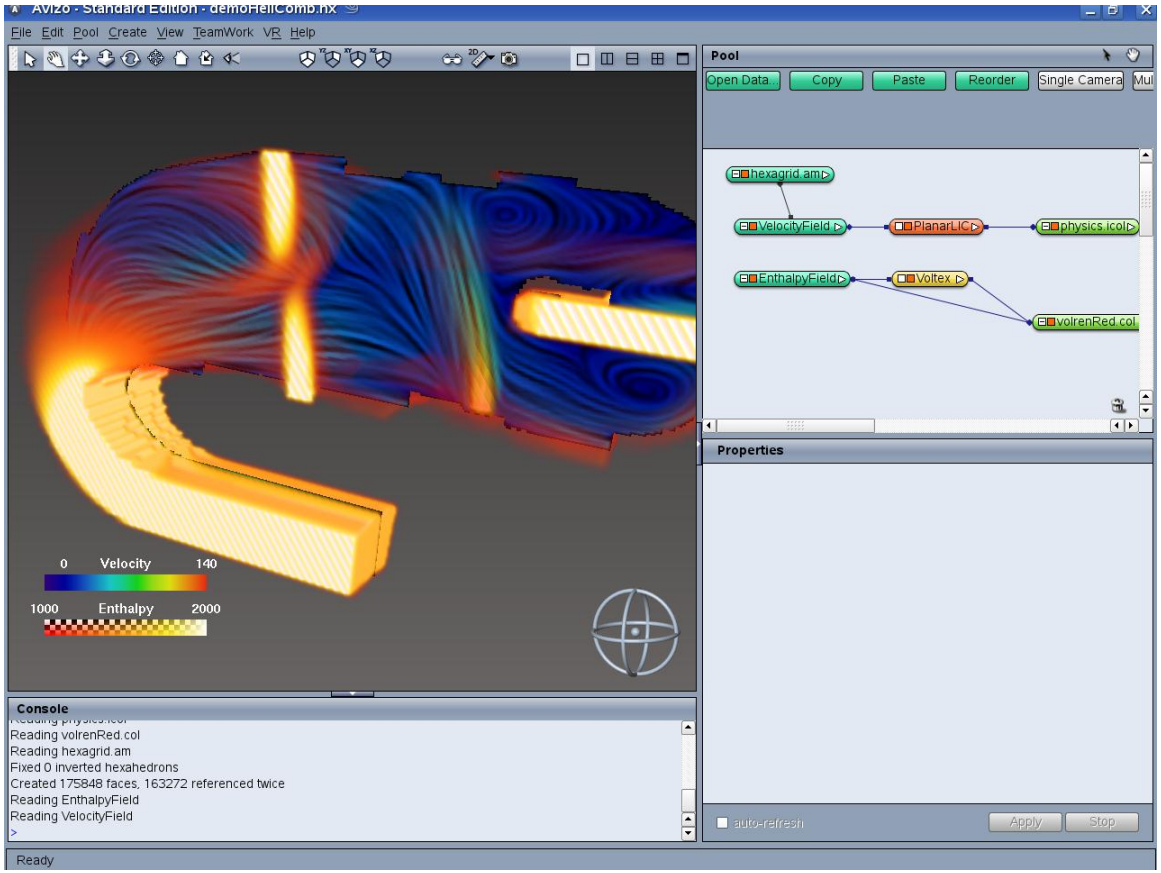


Open DX

Former IBM Data Explorer

http://www.opendx.org/

OpenSource

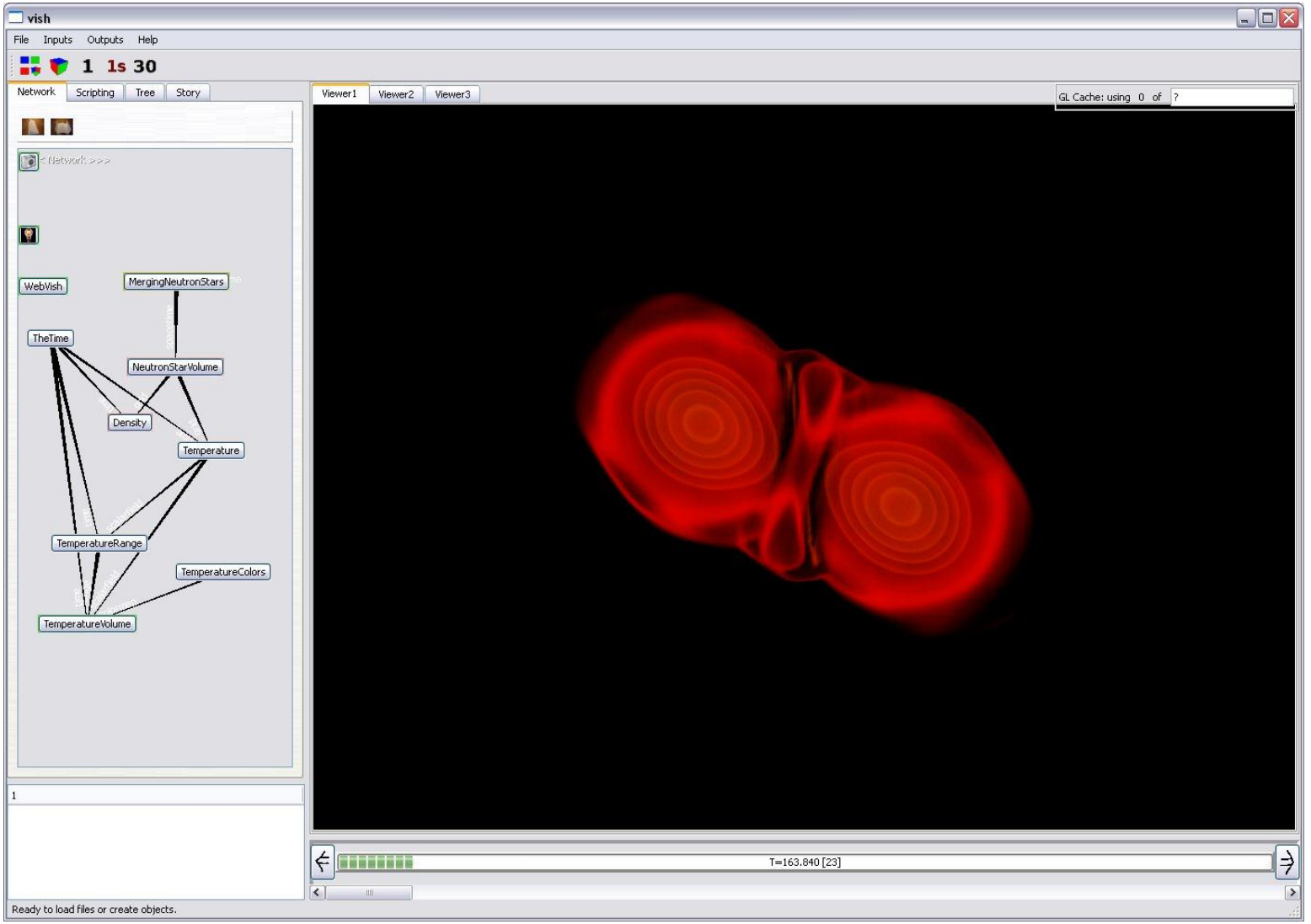


Amira / Avizo

http://www.amiravis.com/

http://amira.zib.de/

Commercial software



VISH - Visualization Shell

Developed at CCT and elsewhere: http://sciviz.cct.lsu.edu/projects/vish/

Academic open source licensing (not formally open source, but freely available to academic community)

Introduction to the Visualization Shell

VISH

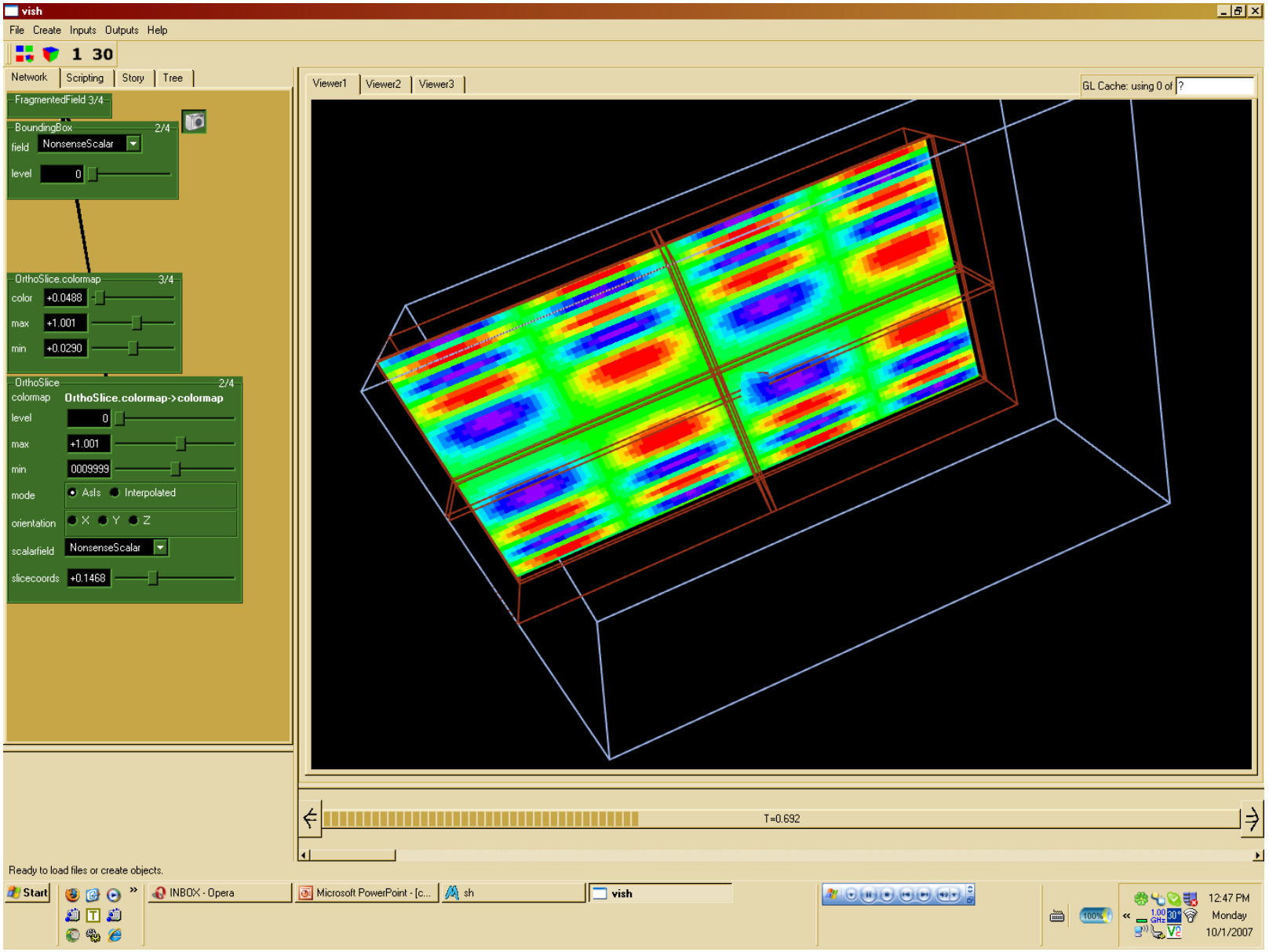
What is VISH?

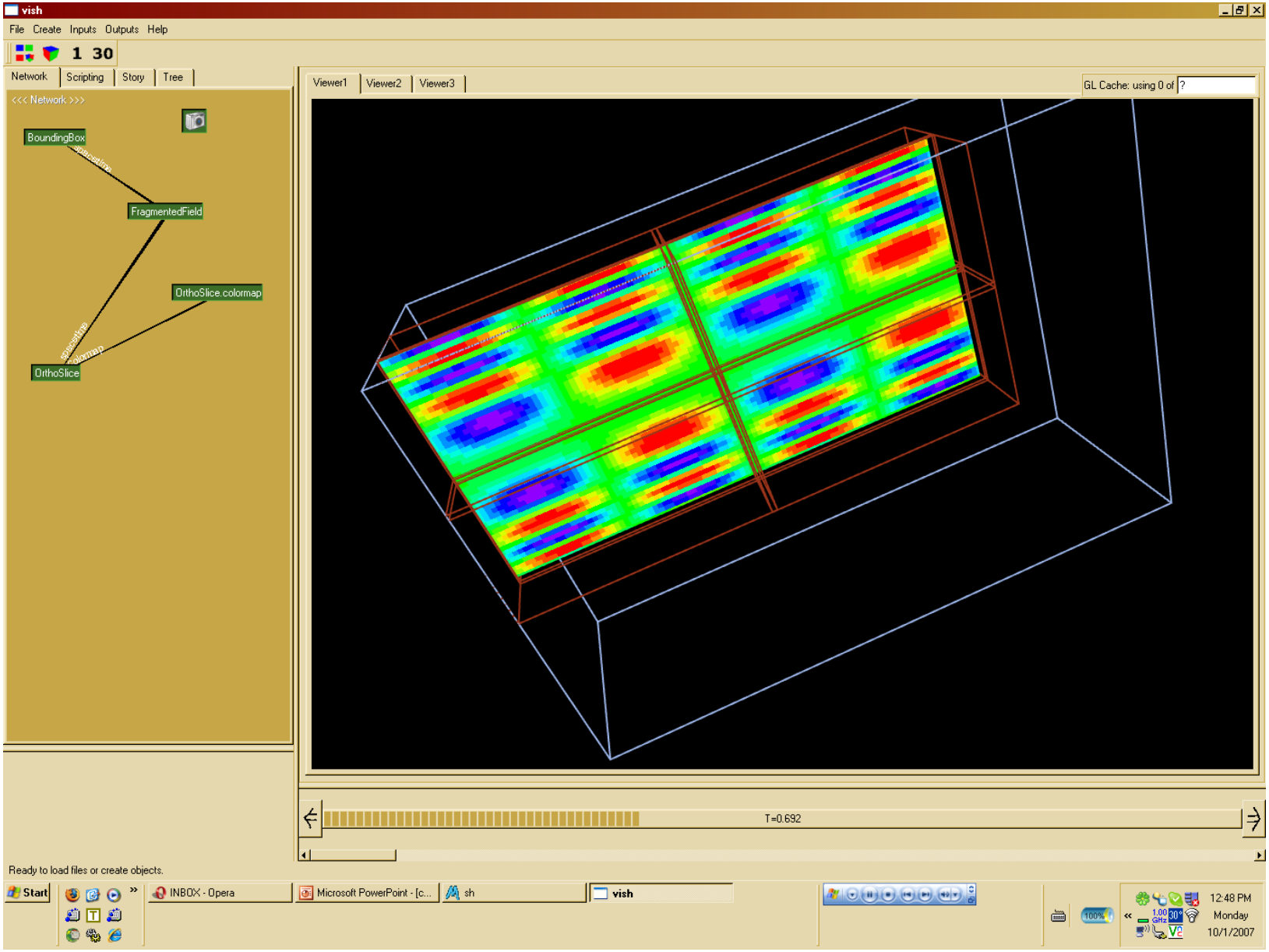
A highly modular infrastructure to implement visualization (and more) algorithms

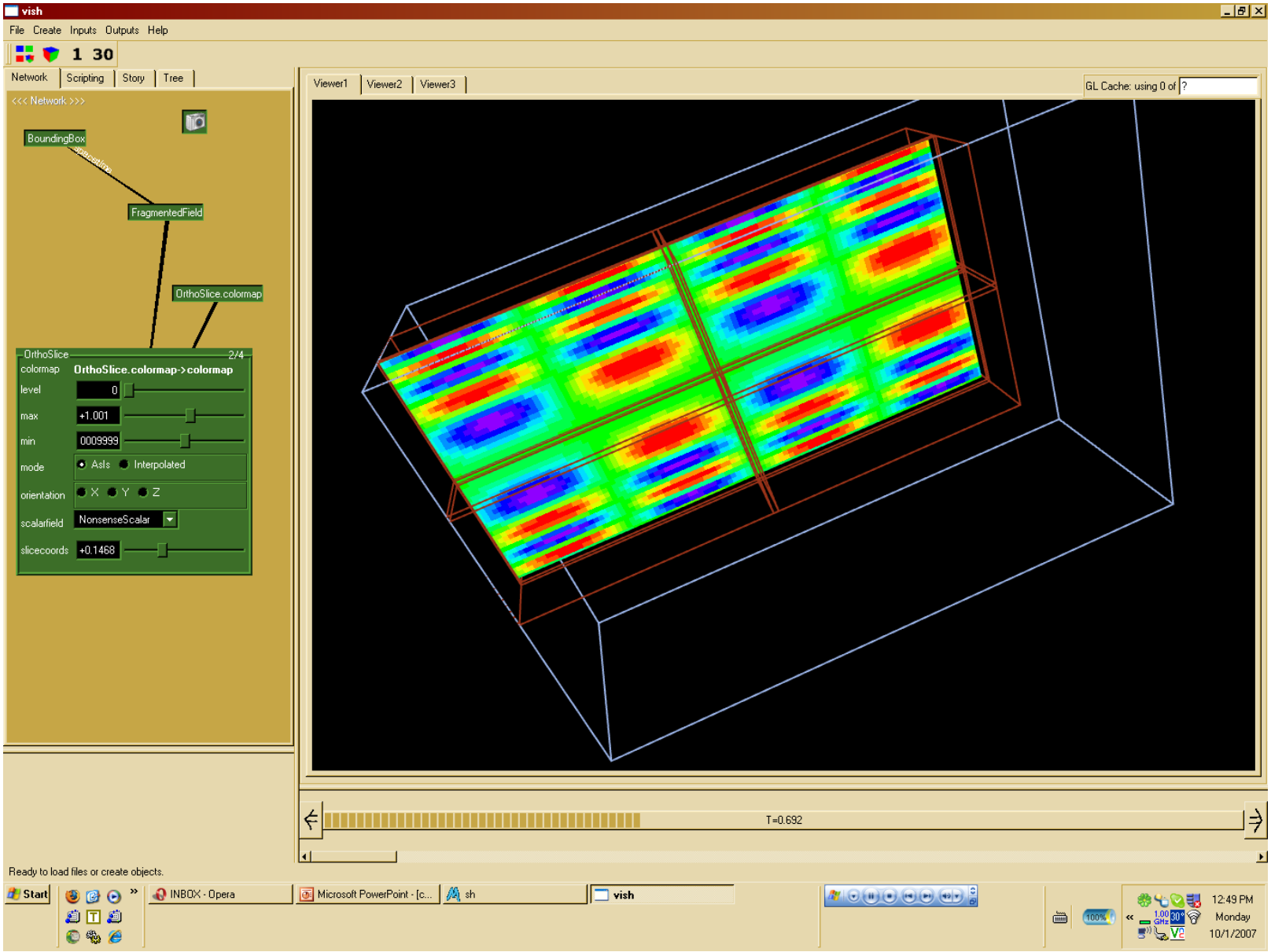
- Strong encapsulation between components
- Abstract interface that allows to integrate VISH components into existing applications
- Systematic approach to visualization concepts
- Pool of Visualization Objects ("Vishes")

Everything is a plugin

- "microkernel" – defines objects and their relationships
- "plugins" – OpenGL rendering, data I/O layers, GUI


vish
File Create Inputs Outputs Help
1 30
Network Scripting Story Tree
FragmentedField 3/4
BoundingBox 2/4
field NonsenseScalar
level 0
OrthoSlice.colormap 3/4
color +0.0488
max +1.001
min +0.0290
OrthoSlice 2/4
colormap OrthoSlice.colormap->colormap
level 0
max +1.001
min 0009999
mode Asls Interpolated
orientation X Y Z
scalarfield NonsenseScalar
slicecoords +0.1468
Viewer1 Viewer2 Viewer3
GL Cache: using 0 of ?
T=0.692
Ready to load files or create objects.
Start
INBOX - Opera
Microsoft PowerPoint - [c...
sh
vish
12:47 PM
Monday
10/1/2007


vish
File Create Inputs Outputs Help
1 30
Network Scripting Story Tree
<<< Network >>>
BoundingBox
FragmentedField
OrthoSlice.colormap
OrthoSlice
Viewer1 Viewer2 Viewer3
GL Cache: using 0 of ?
T=0.692
Ready to load files or create objects.
Start
INBOX - Opera
Microsoft PowerPoint - [c...
sh
vish
12:48 PM
Monday
10/1/2007


vish
File Create Inputs Outputs Help
1 30
Network Scripting Story Tree
<<< Network >>>
BoundingBox
FragmentedField
OrthoSlice.colormap
OrthoSlice 2/4
colormap OrthoSlice.colormap->colormap
level 0
max +1.001
min 0009999
mode As Is Interpolated
orientation X Y Z
scalarfield NonsenseScalar
slicecoords +0.1468
Viewer1 Viewer2 Viewer3
GL Cache: using 0 of ?
T=0.692
Ready to load files or create objects.
Start
INBOX - Opera
Microsoft PowerPoint - [c...
sh
vish
12:49 PM
Monday
10/1/2007

VISH Objects

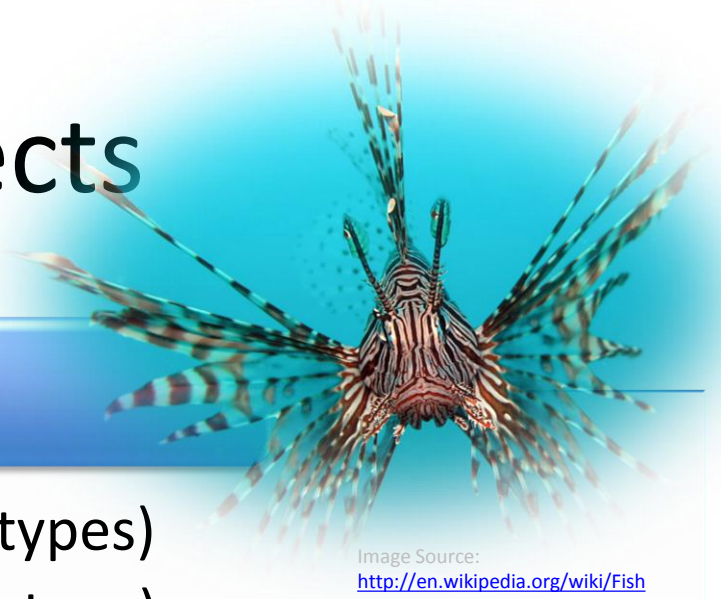


Image Source:
http://en.wikipedia.org/wiki/Fish

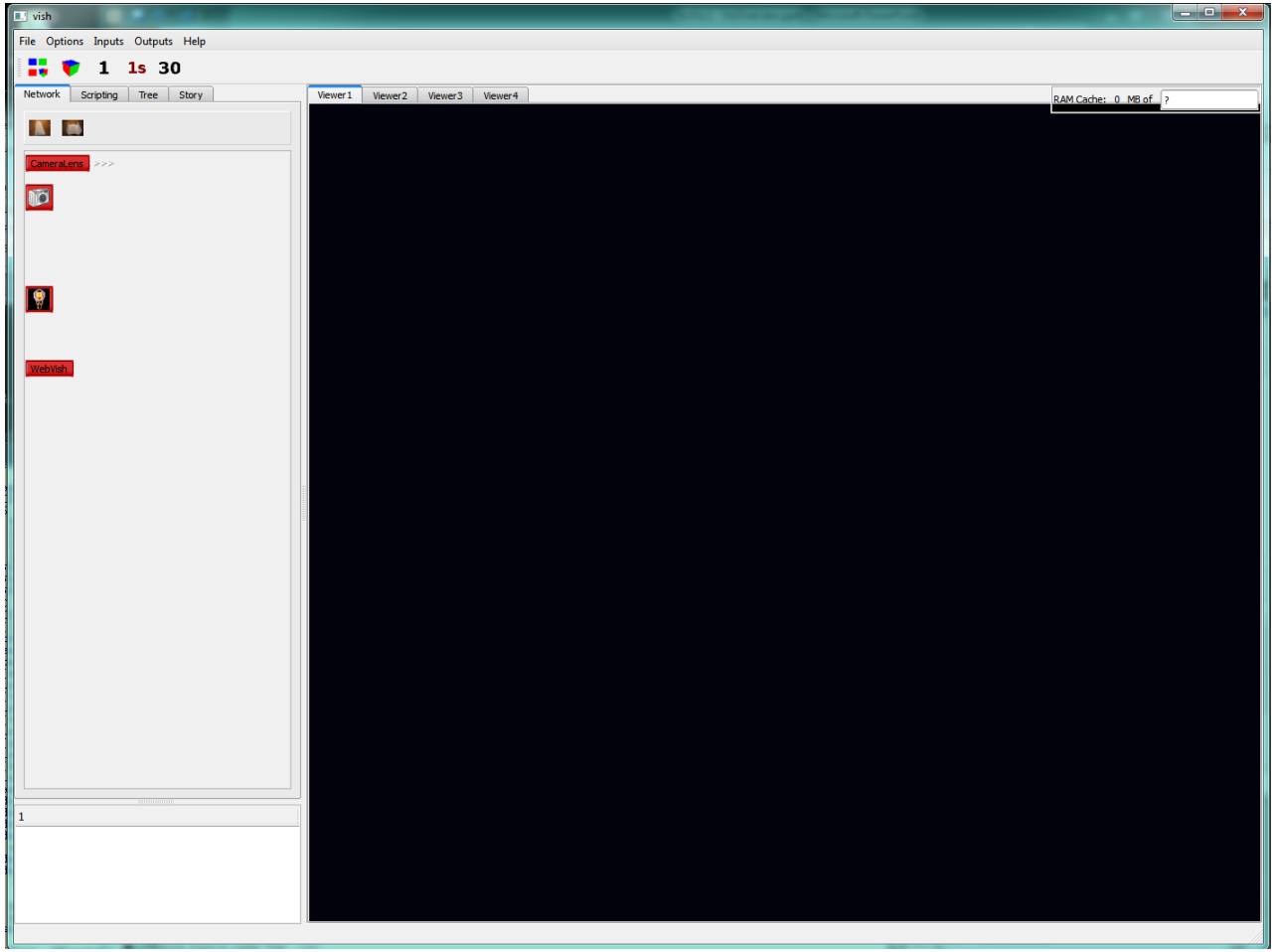
Abstract objects that optionally

- Request input types (native C++ types)
- Provide output types (native C++ type)
- Execute some procedure (virtual function)

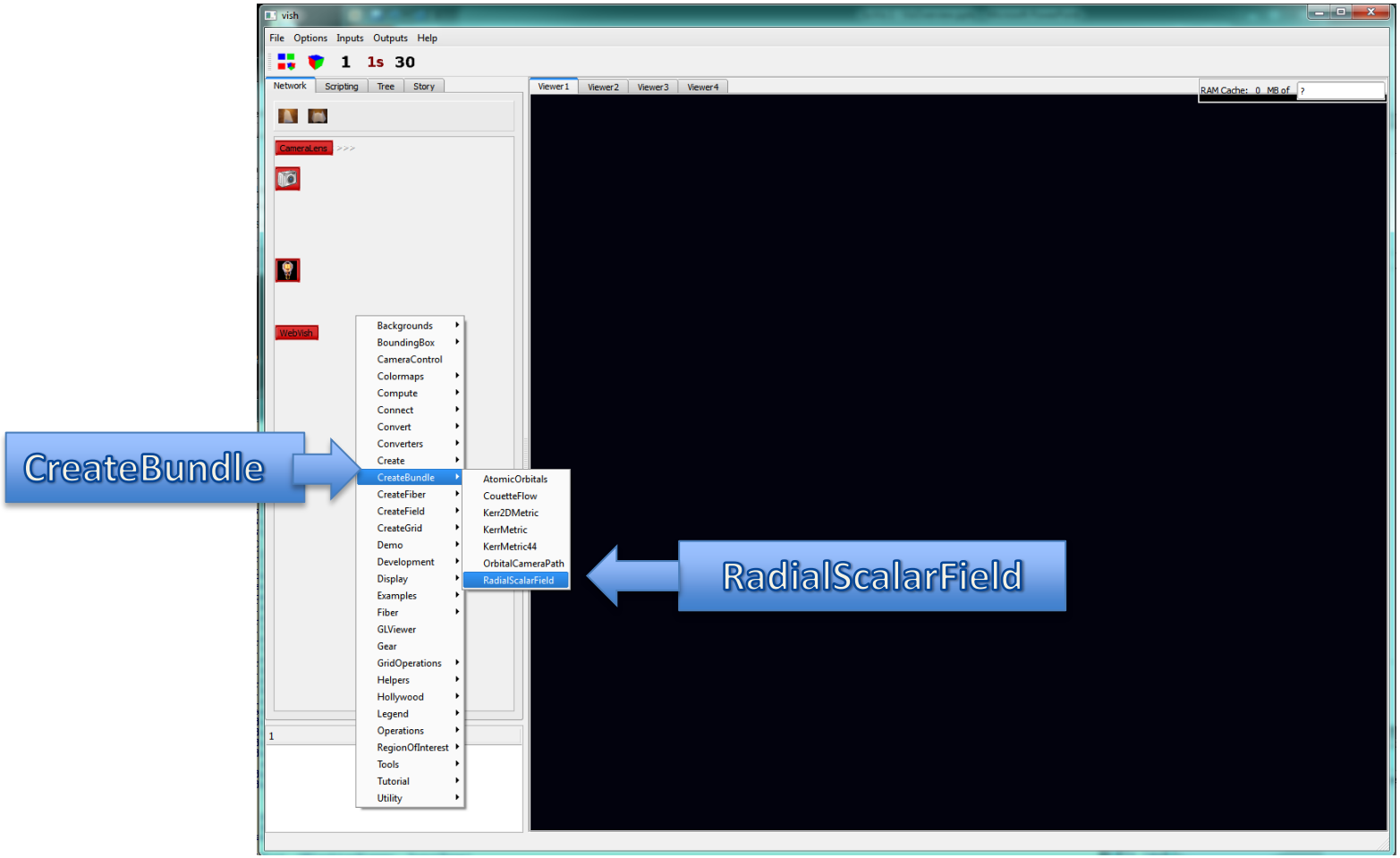
Input/output types are *Parameters*

- Create as source
- Use as input
- Share (attach)

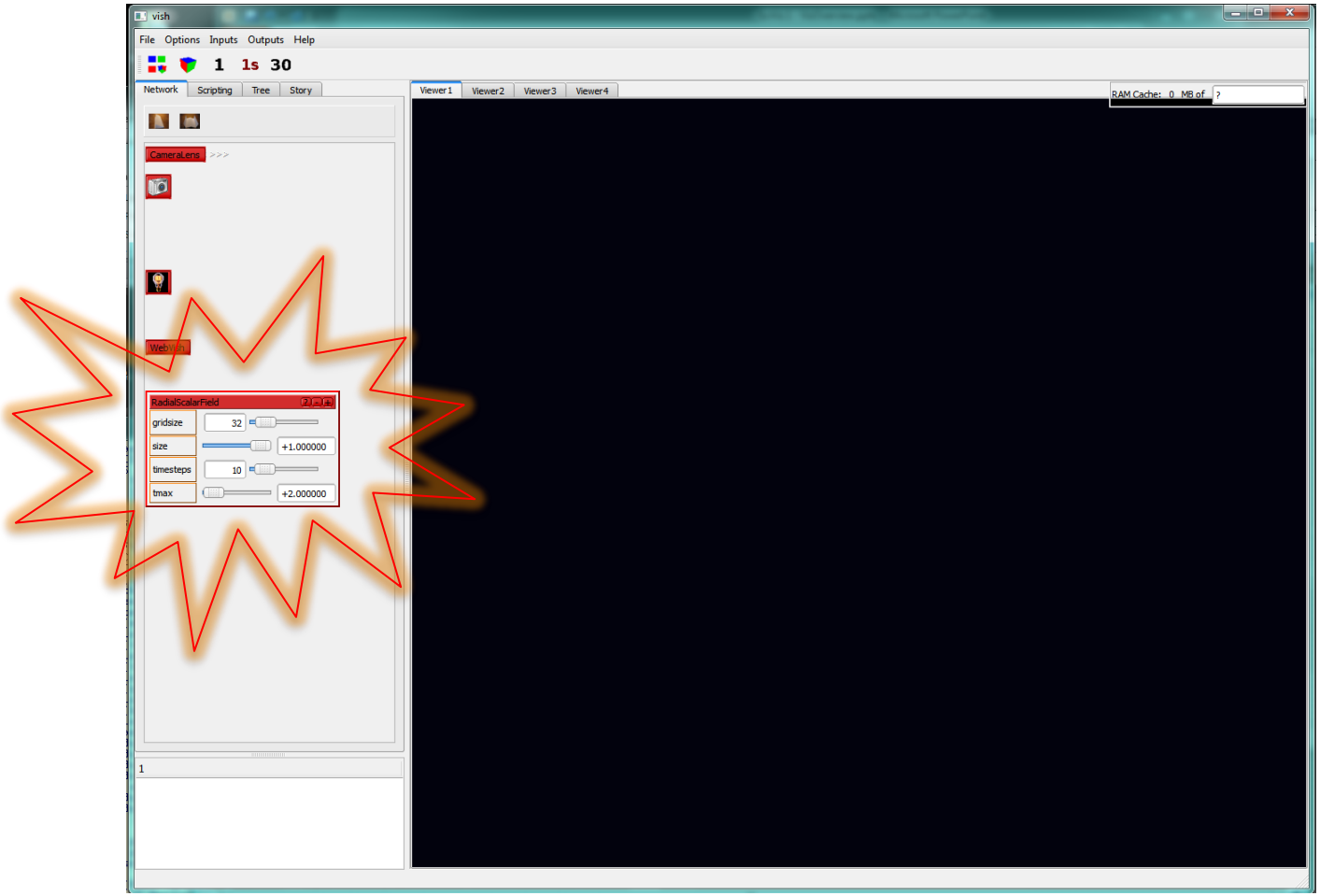
Practice: Using Vish



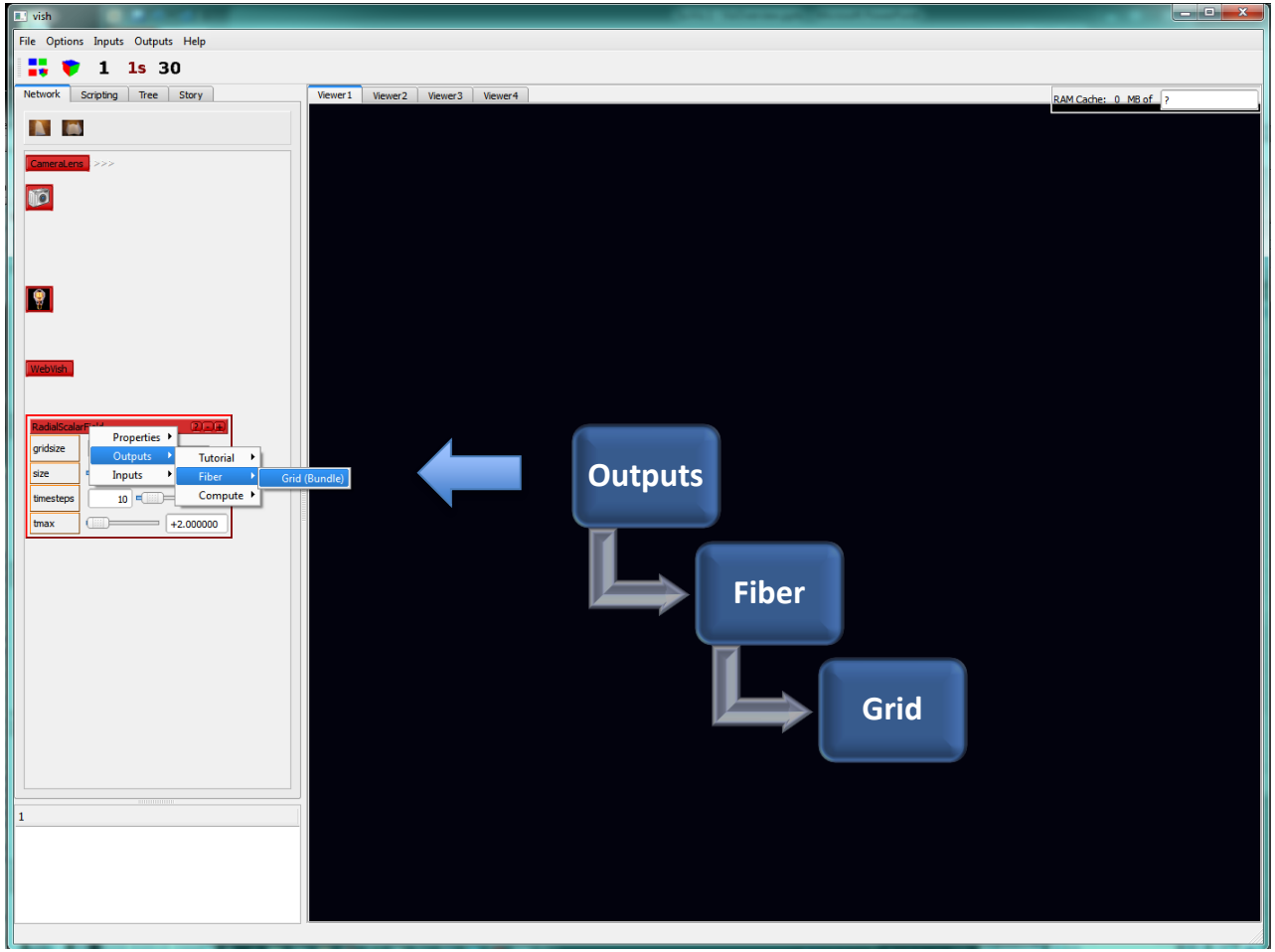
CreateBundle/RadialScalarField



Vish Object “RadialScalarField”

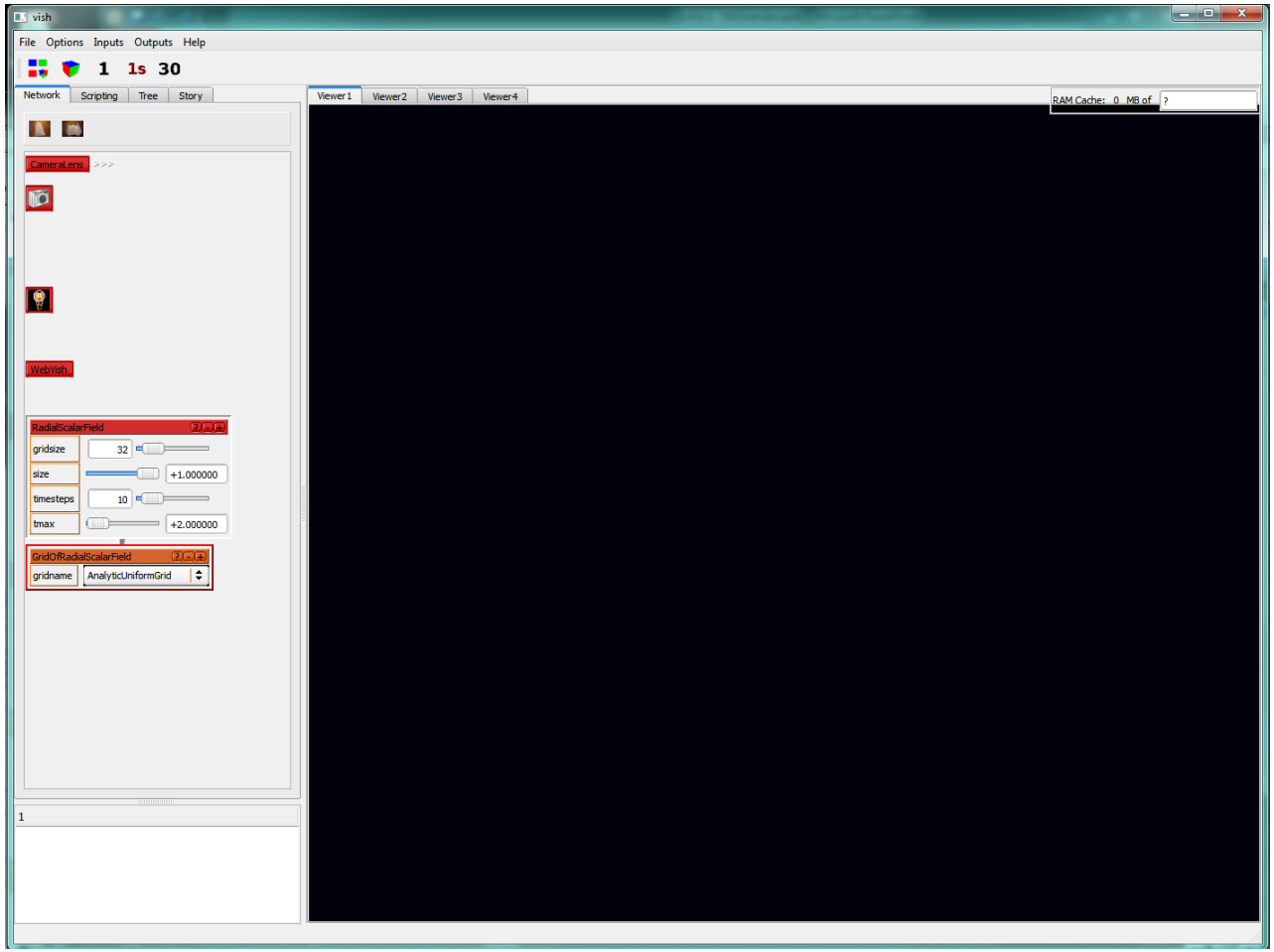


Outputs: Fiber/Grid

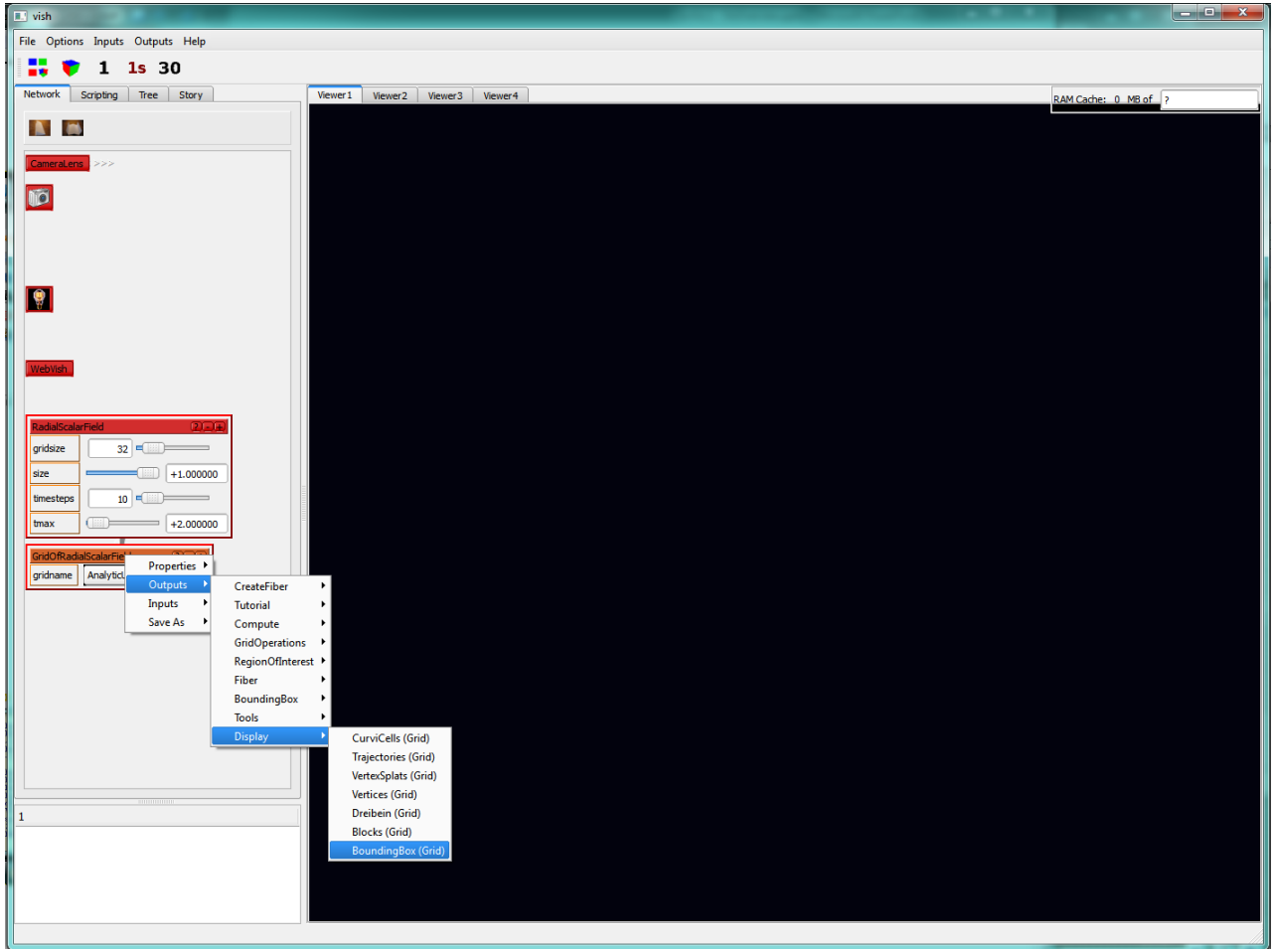


Data

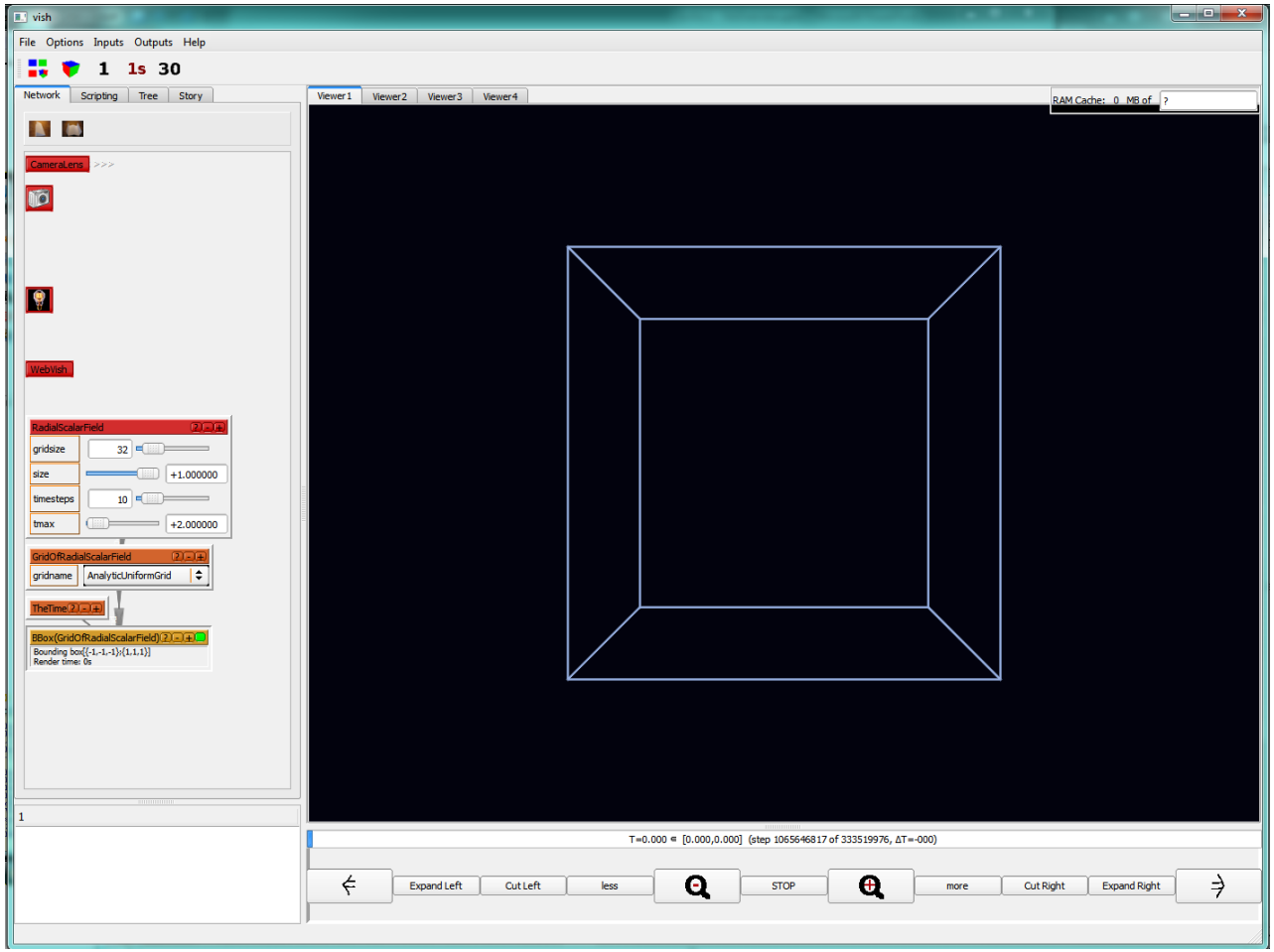
Selection



Outputs: Display/BoundingBox



Data → Selection → Display

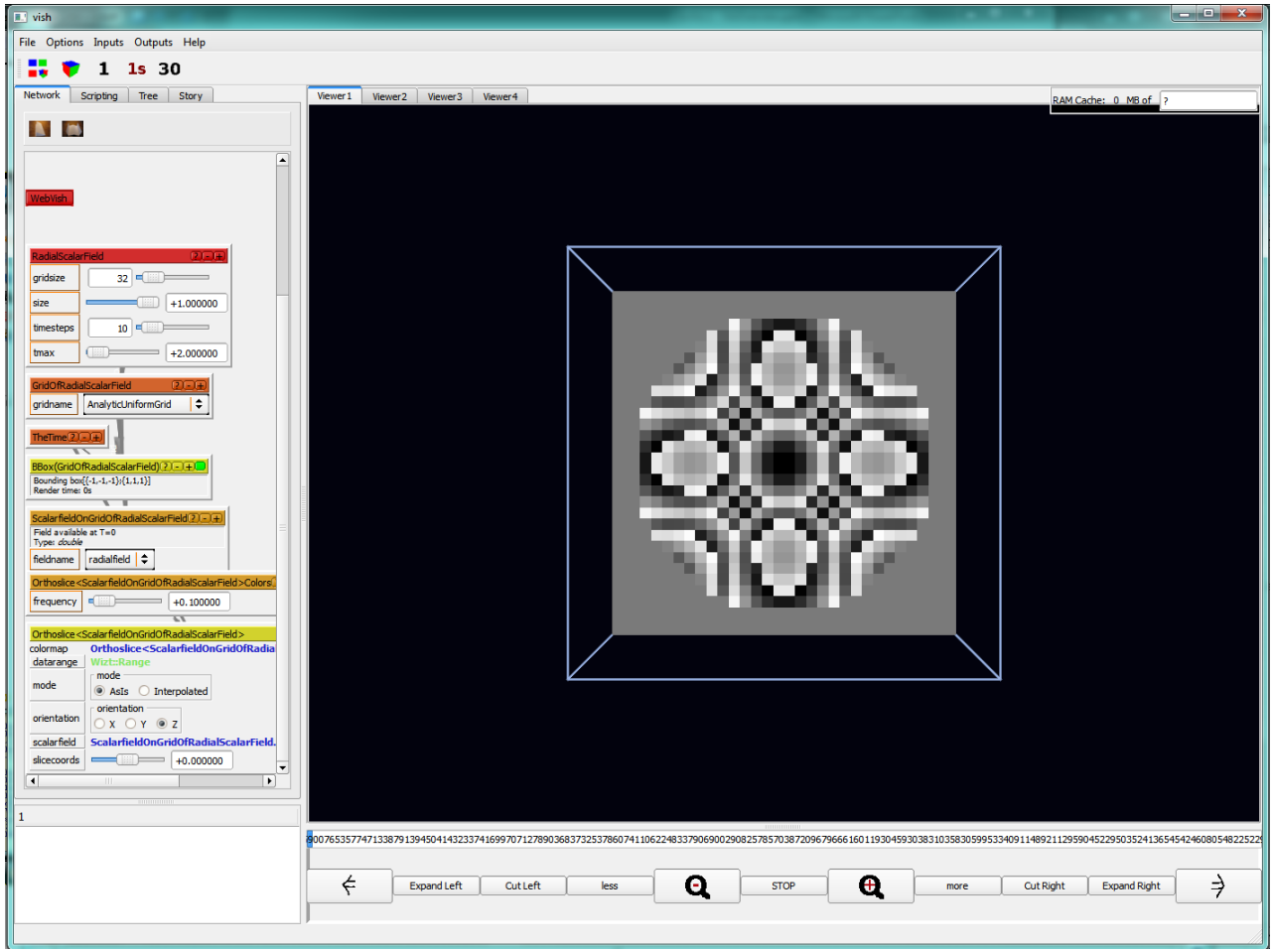


Fiber

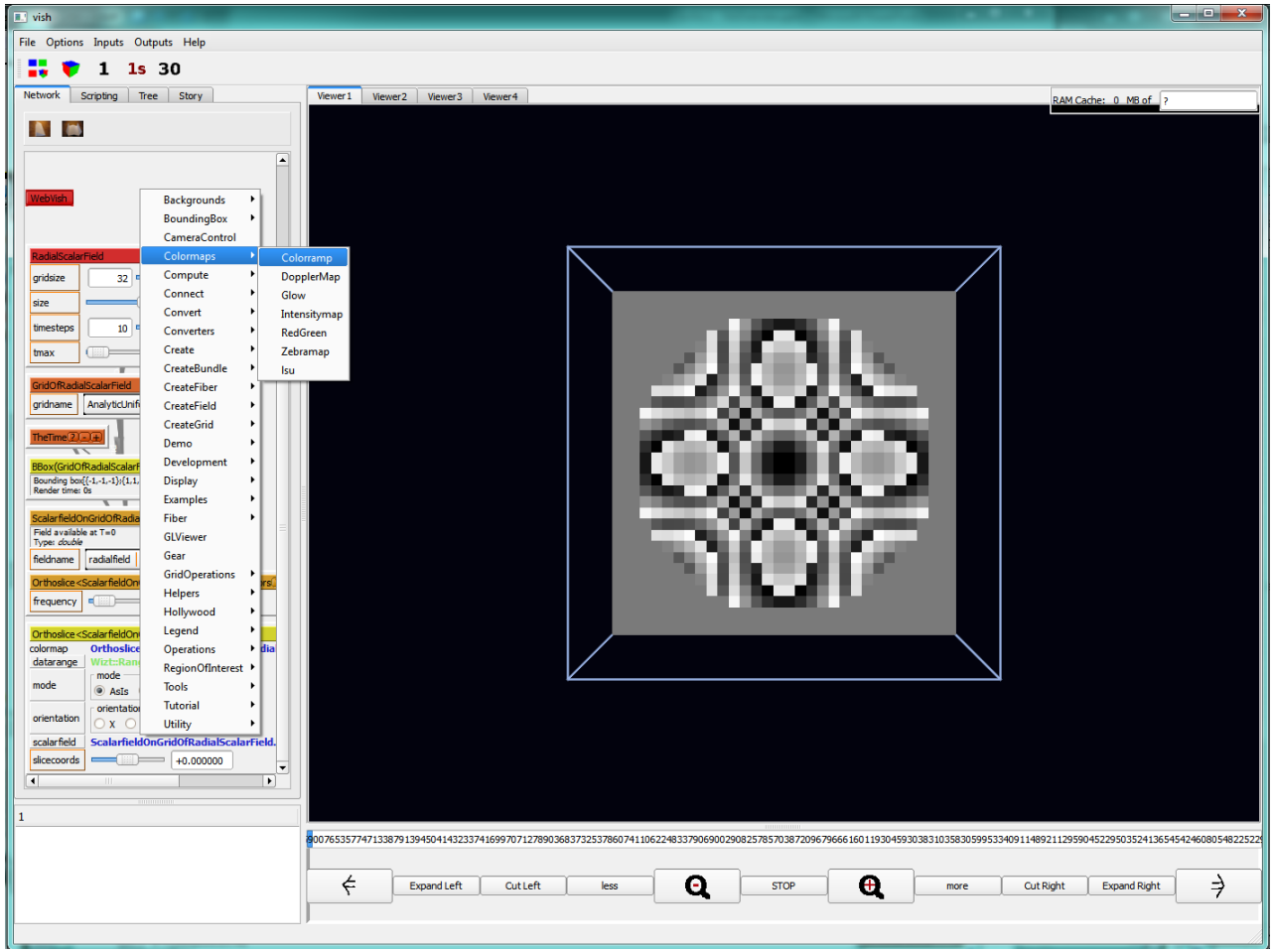
- Field

Display

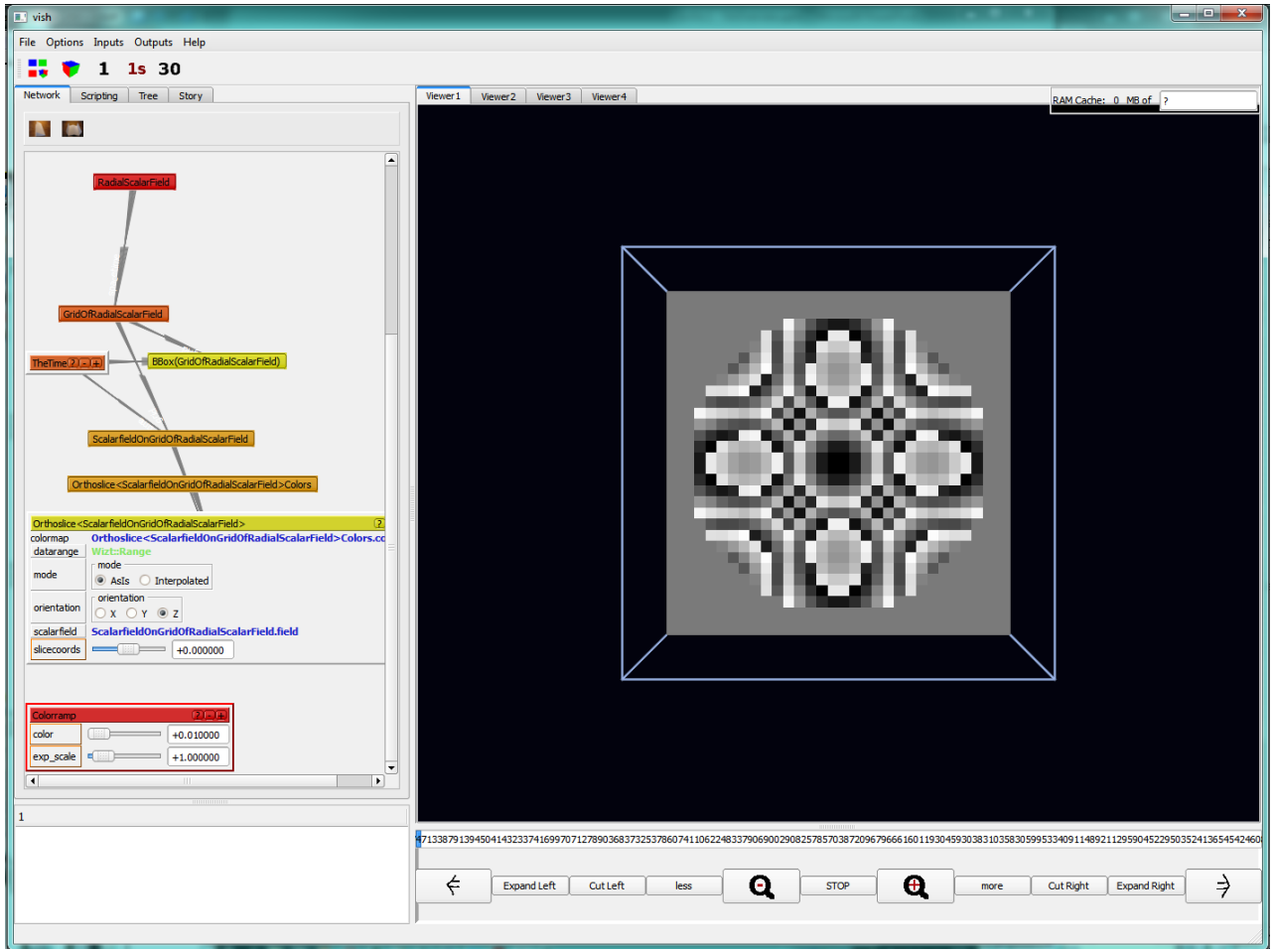
- OrthoSlice

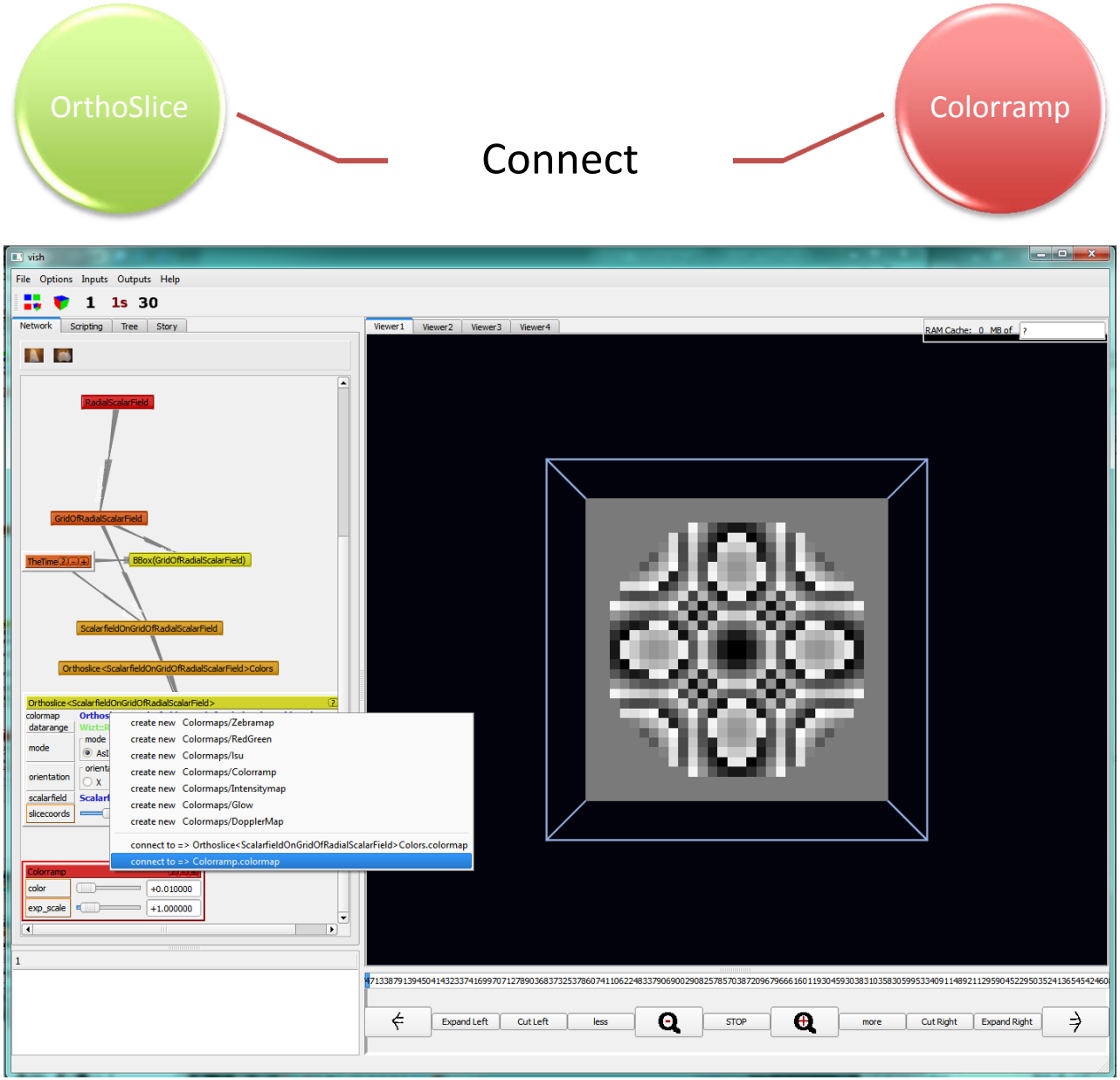


Colormaps/Colorramp

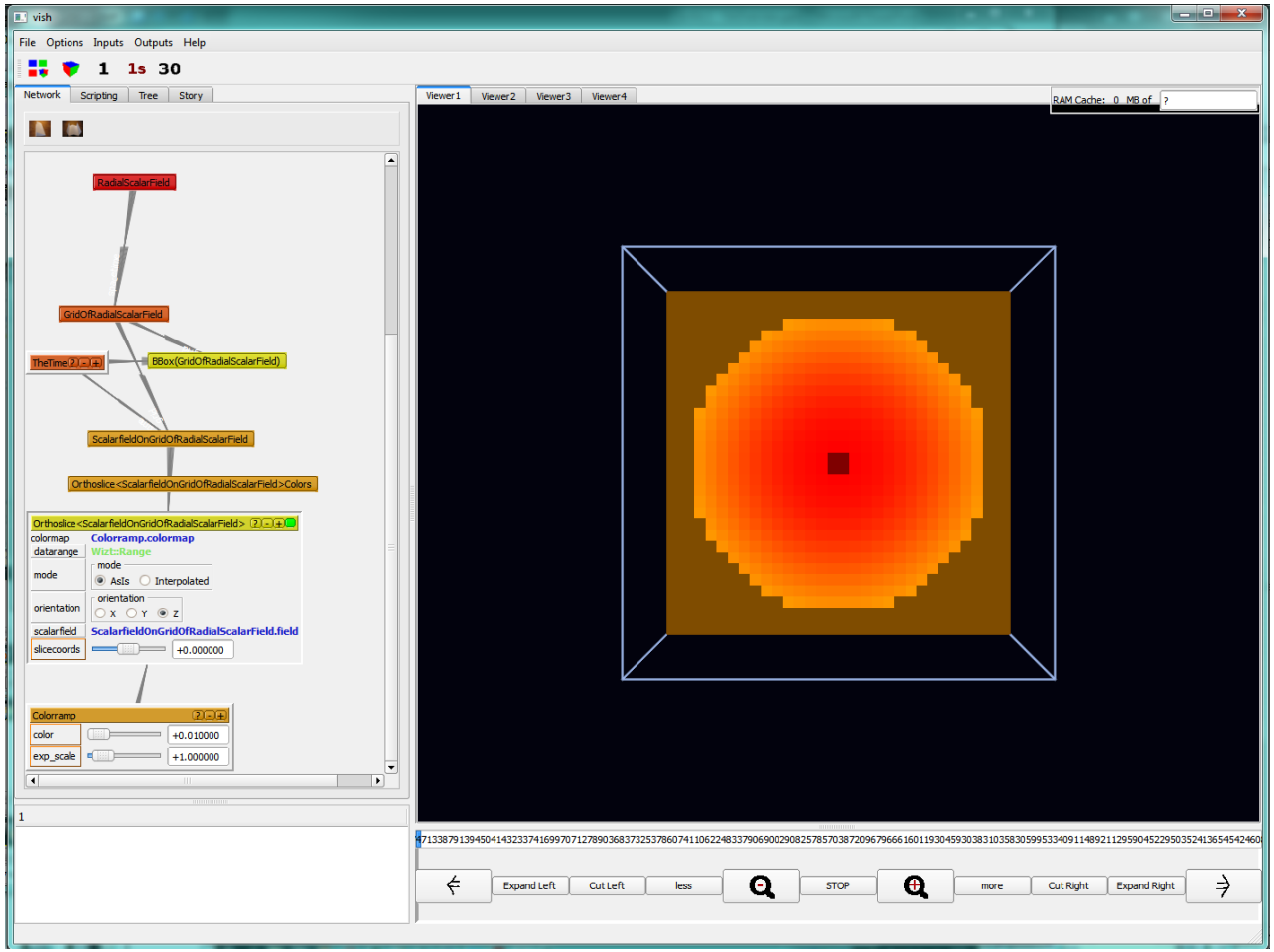


Iconize Objects

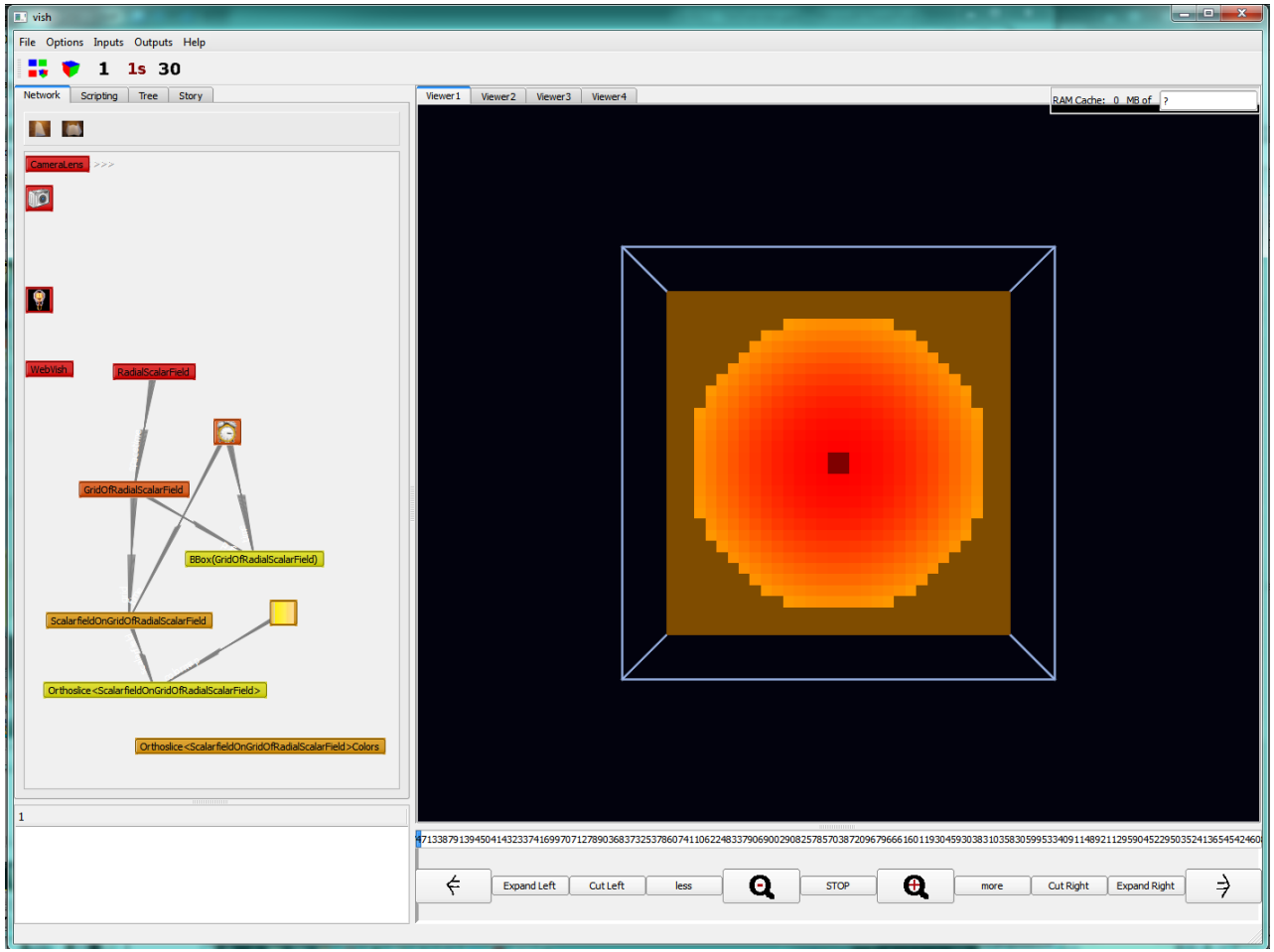



OrthoSlice
Connect
Colorramp
vish
File Options Inputs Outputs Help
Network Scripting Tree Story
Viewer1 Viewer2 Viewer3 Viewer4
RadialScalarField
GridOfRadialScalarField
BBox(GridOfRadialScalarField)
ScalarfieldOnGridOfRadialScalarField
Orthoslice<ScalarfieldOnGridOfRadialScalarField>Colors
create new Colormaps/Zebramap
create new Colormaps/RedGreen
create new Colormaps/Isu
create new Colormaps/Colorramp
create new Colormaps/Intensitymap
create new Colormaps/Glow
create new Colormaps/DopplerMap
connect to => Orthoslice<ScalarfieldOnGridOfRadialScalarField>Colors.colormap
connect to => Colorramp.colormap
Colorramp
color
exp_scale
Expand Left
Cut Left
less
STOP
more
Cut Right
Expand Right

Parameterized Visualization Network



Iconification shows Data Flow



As Vish Script
(save as “myscript.vis”)

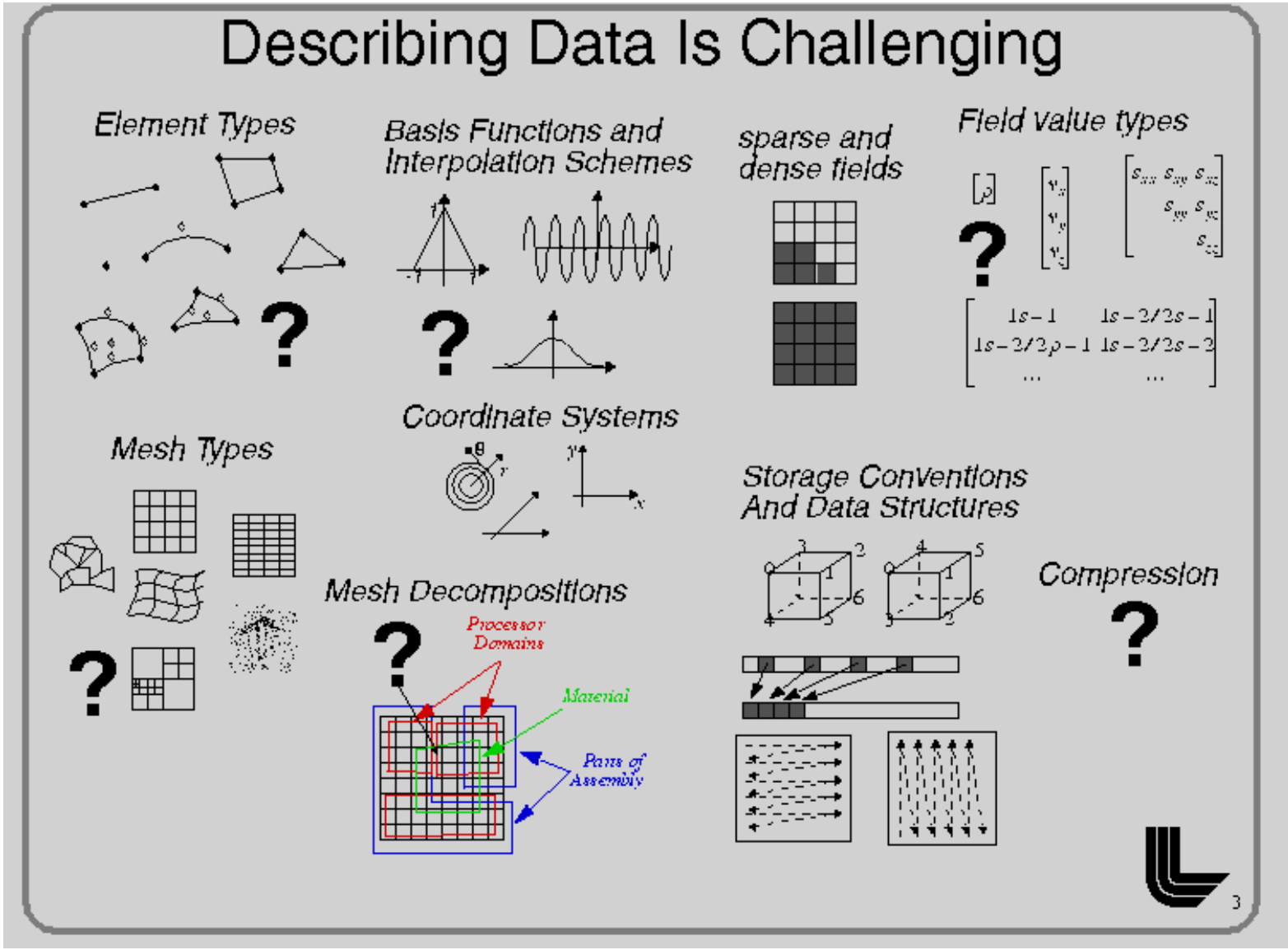
```
CreateBundle/RadialScalarField MyData

Fiber/Grid MyGrid
MyGrid=>MyData

Display/BoundingBox MyBBox
MyBBox=>MyGrid

Fiber/Scalarfield MyScalarfield
MyScalarfield=>MyGrid

Colormaps/Colorramp MyColors

Display/OrthoSlice MySlice
MySlice.colors=>MyColors
MySlice=>MyScalarfield
```

Note:
VishScript is keyword-free!

2D, 3D,4D

- Images
- Surfaces
- Meshes
- Volumetric data

INTRODUCTION TO DATA TYPES



Thanks to Mark Miller, LLNL

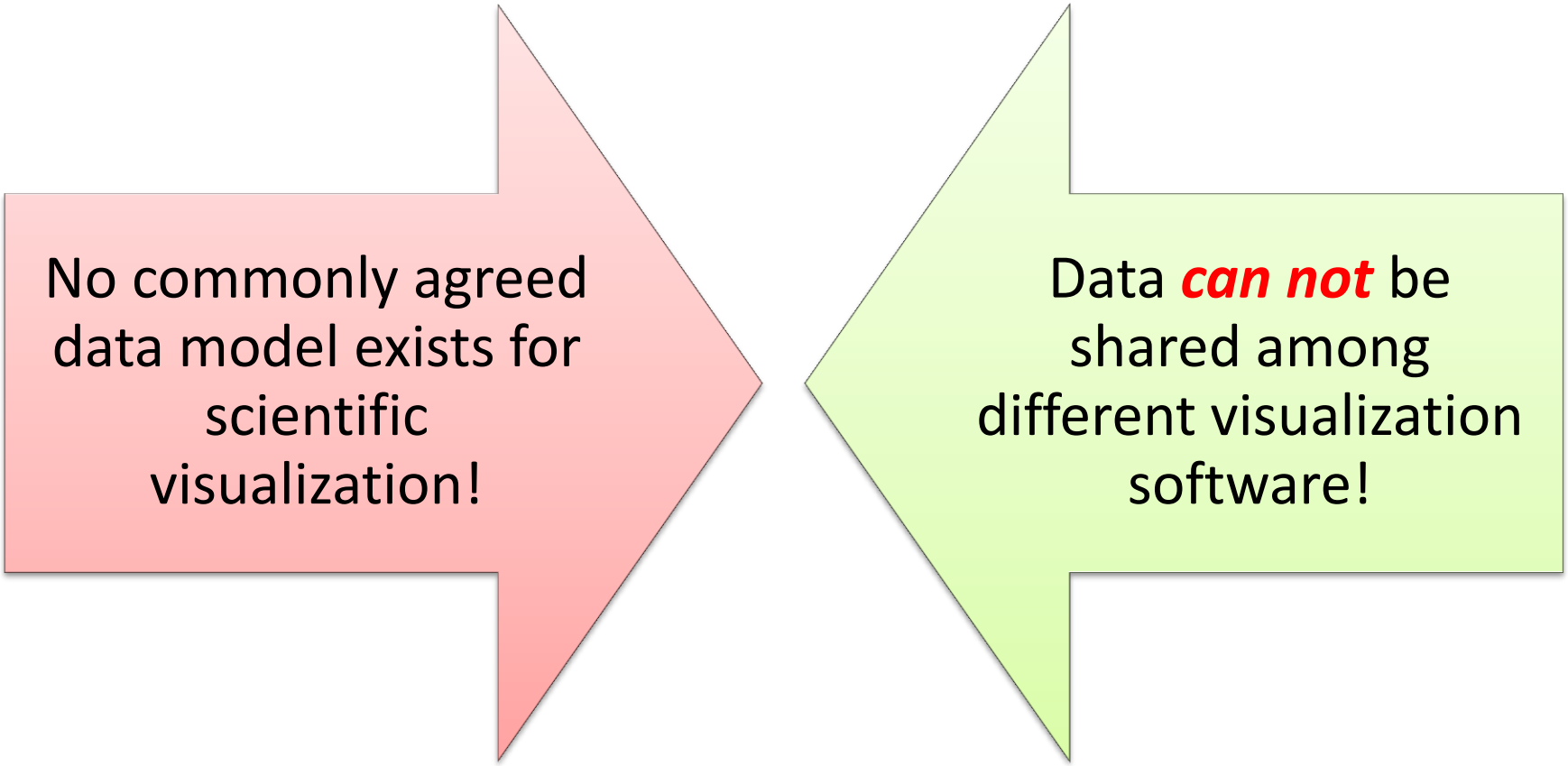
Data Model

A **data model** in software engineering is an abstract model that describes how data are represented and accessed. Data models formally define data elements and relationships among data elements for a domain of interest.

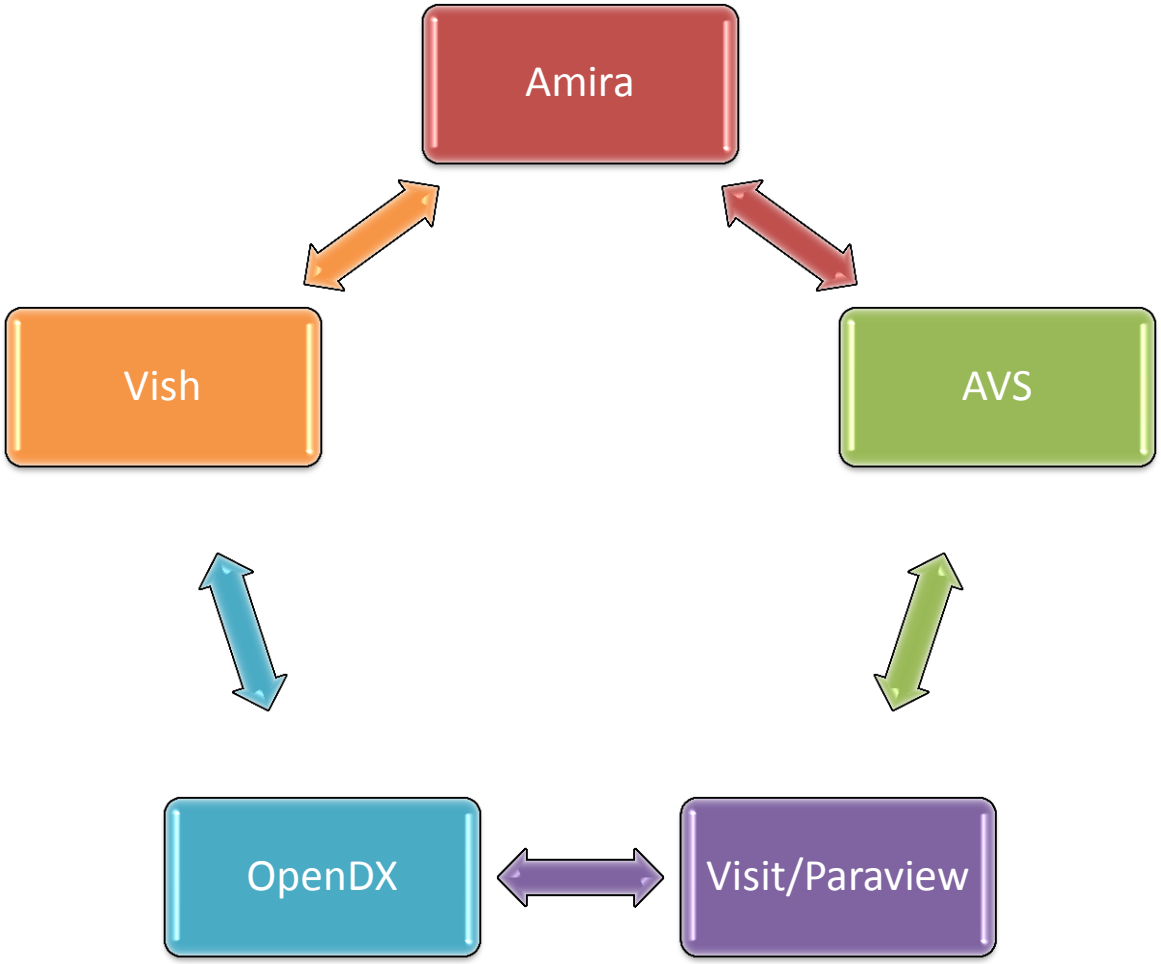
Required in: Databases, Data Visualization, Information systems

Source: http://en.wikipedia.org/wiki/Data_model

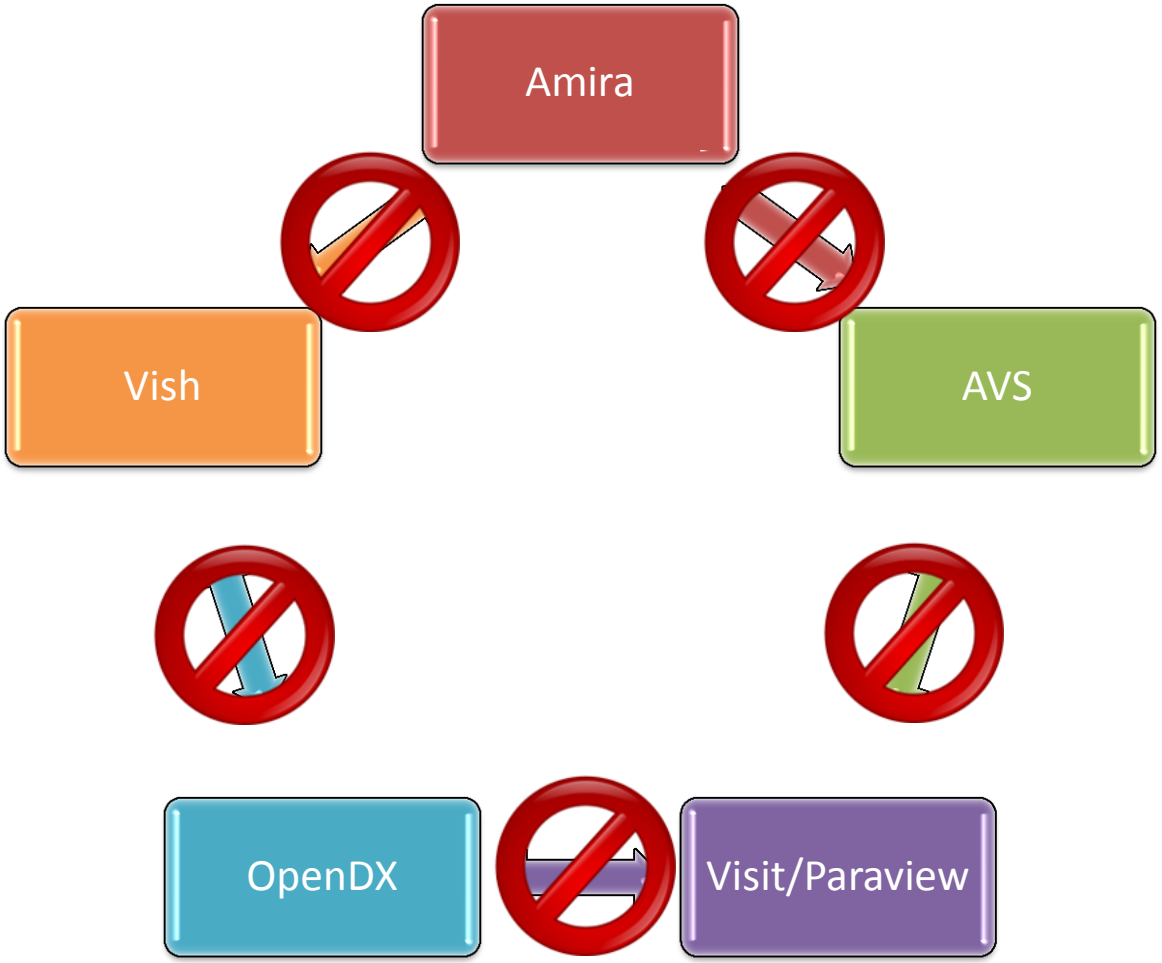
Data Model in SciViz



Interoperability?



Interoperability… wishful thinking

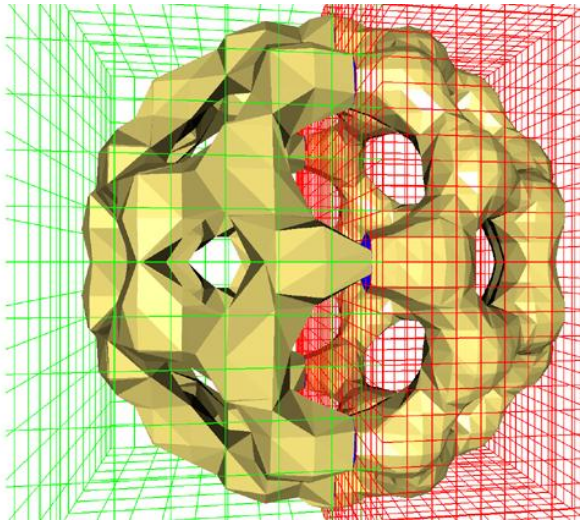


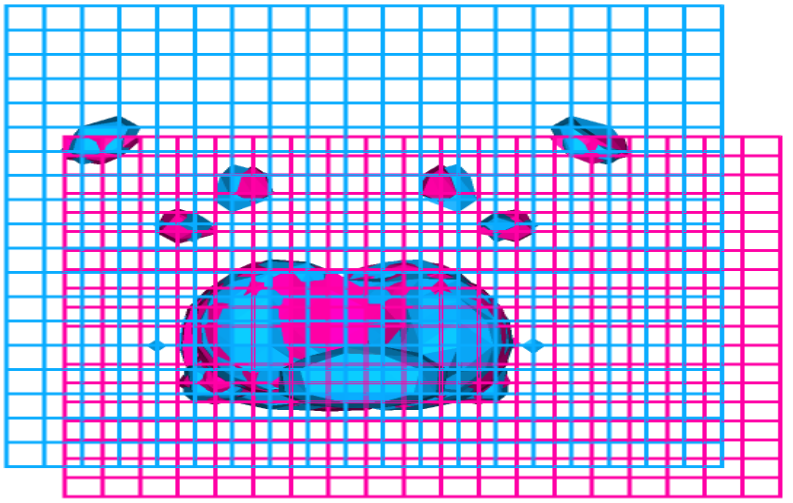
Challenge of a Common Data Model

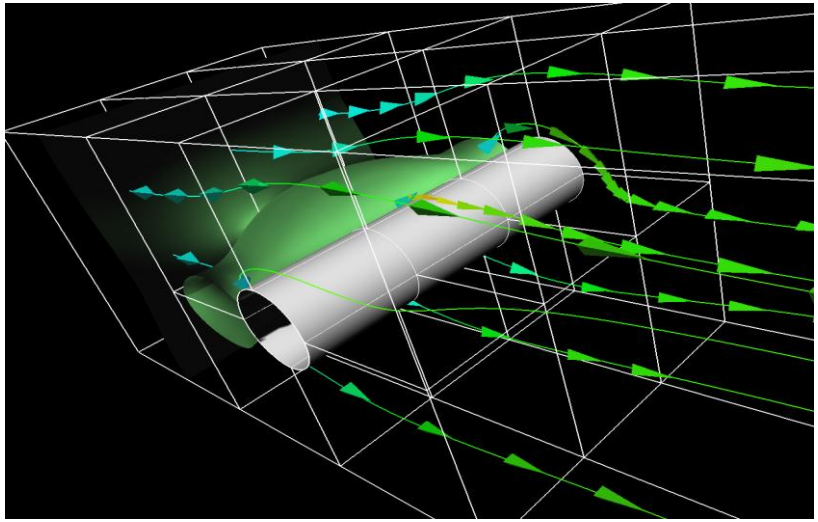
"The proper abstractions for scientific data are known. We just have to use them."

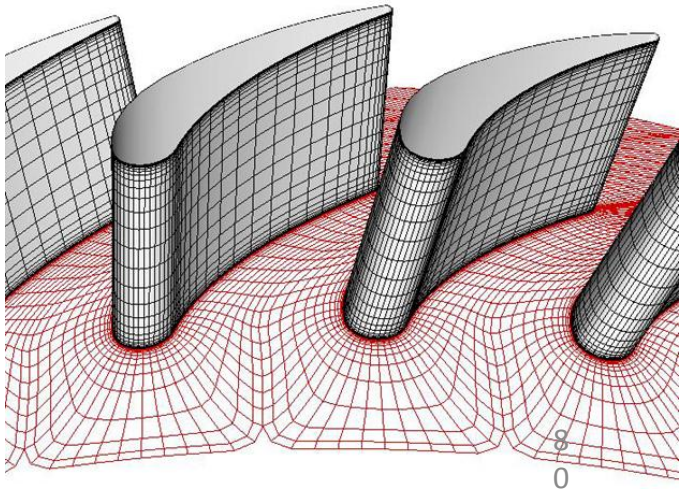
Butler & Pendley, 1989.

Fiber Bundles:
Unification of data types



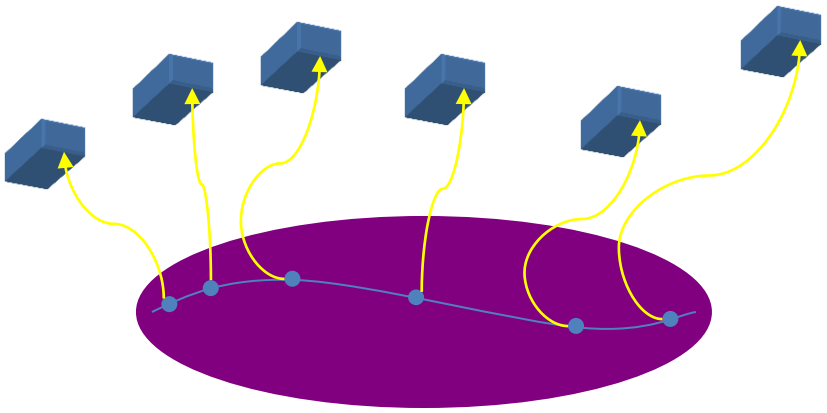






Modelling Data as “Fiber Bundle”

- The total space E can be written locally as a product space B×F of the base space B with some space F.

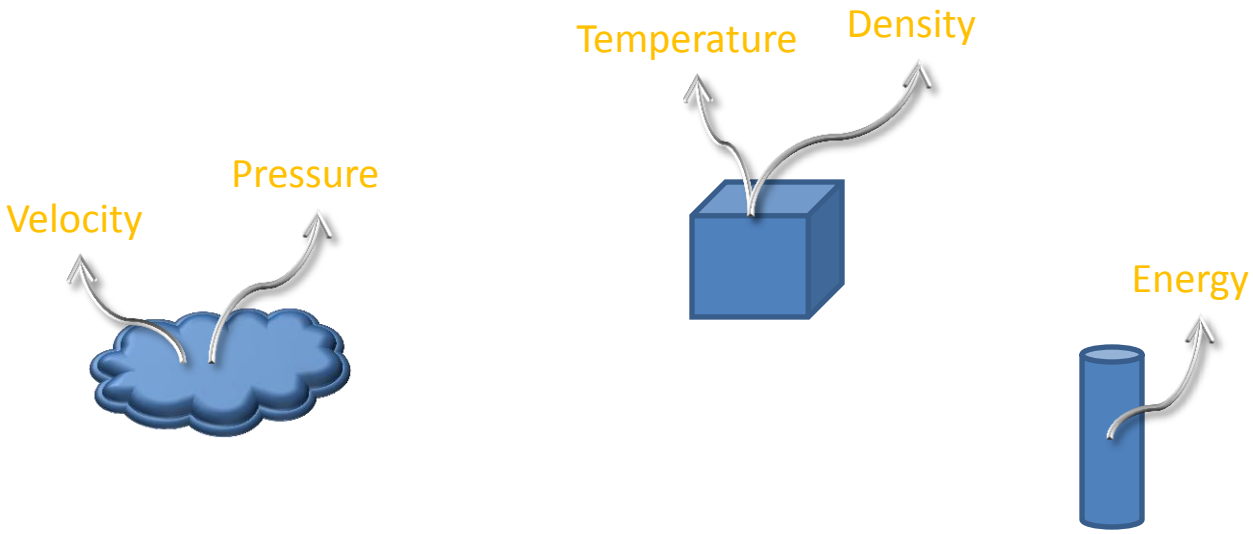


fiber space at each point of base space

base space

Grids and Fields

End-user operates on Grid and Field objects



High-Performance I/O

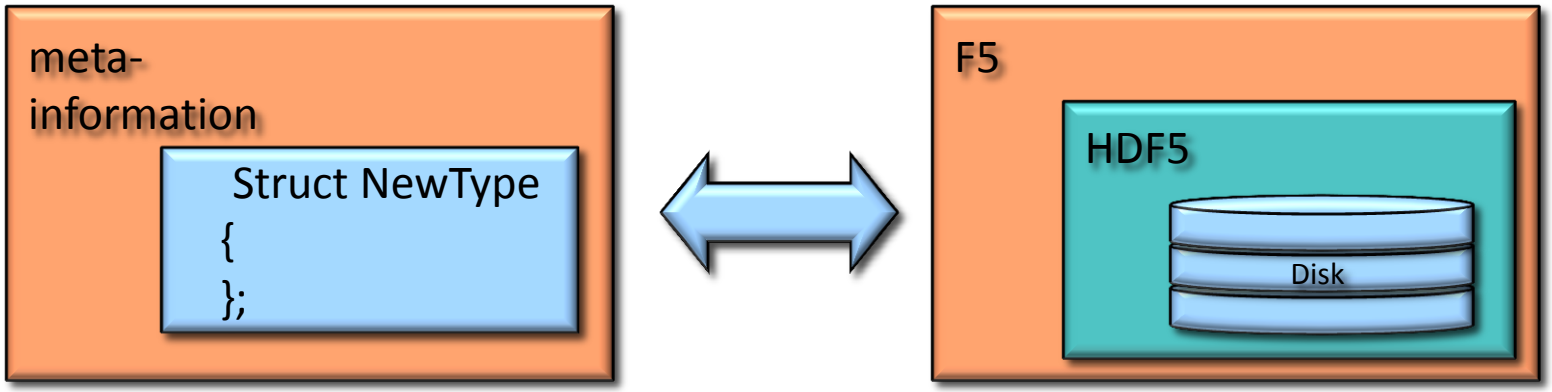
Generic I/O

Persistent Storage

FILE FORMATS: HDF5

Persistent Storage

- Map memory objects to disk objects
  - Combine and integrate with a fast and powerful I/O layer.



HDF5 as underlying technology

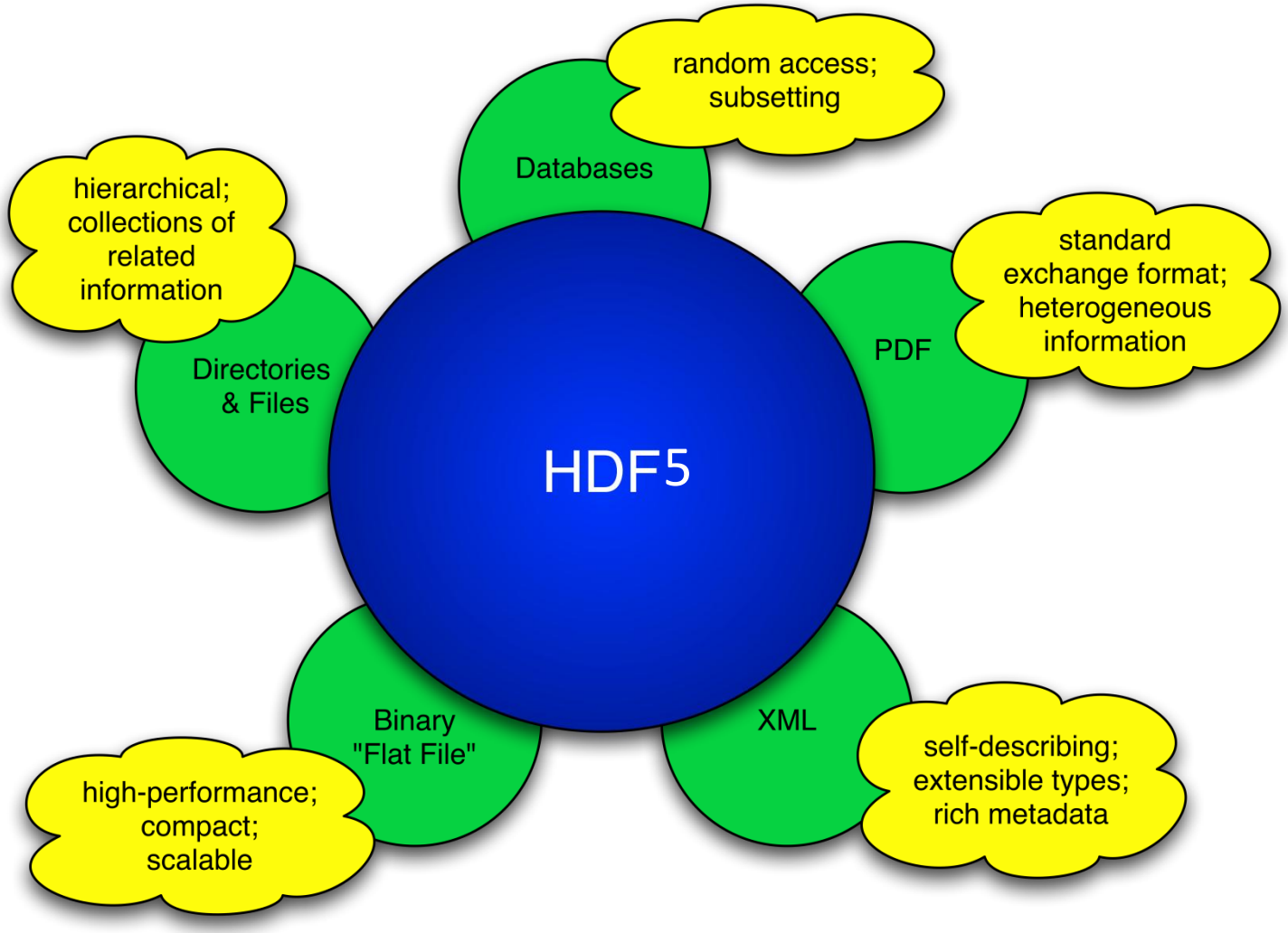
HDF = Hierarchical Data Format

- A file format for managing any kind of data
- Software system to manage data in the format

- HDF5 is the second HDF format ("first" version was HDF4)
  - Development started in 1996
  - First release was in 1998

- Designed for HPC simulations
  - High-performance, large data, long-term data preservation (archival), portability
  - Designed for high volume or complex data
  - Designed for every size and type of system
  - Open format and software library, tools

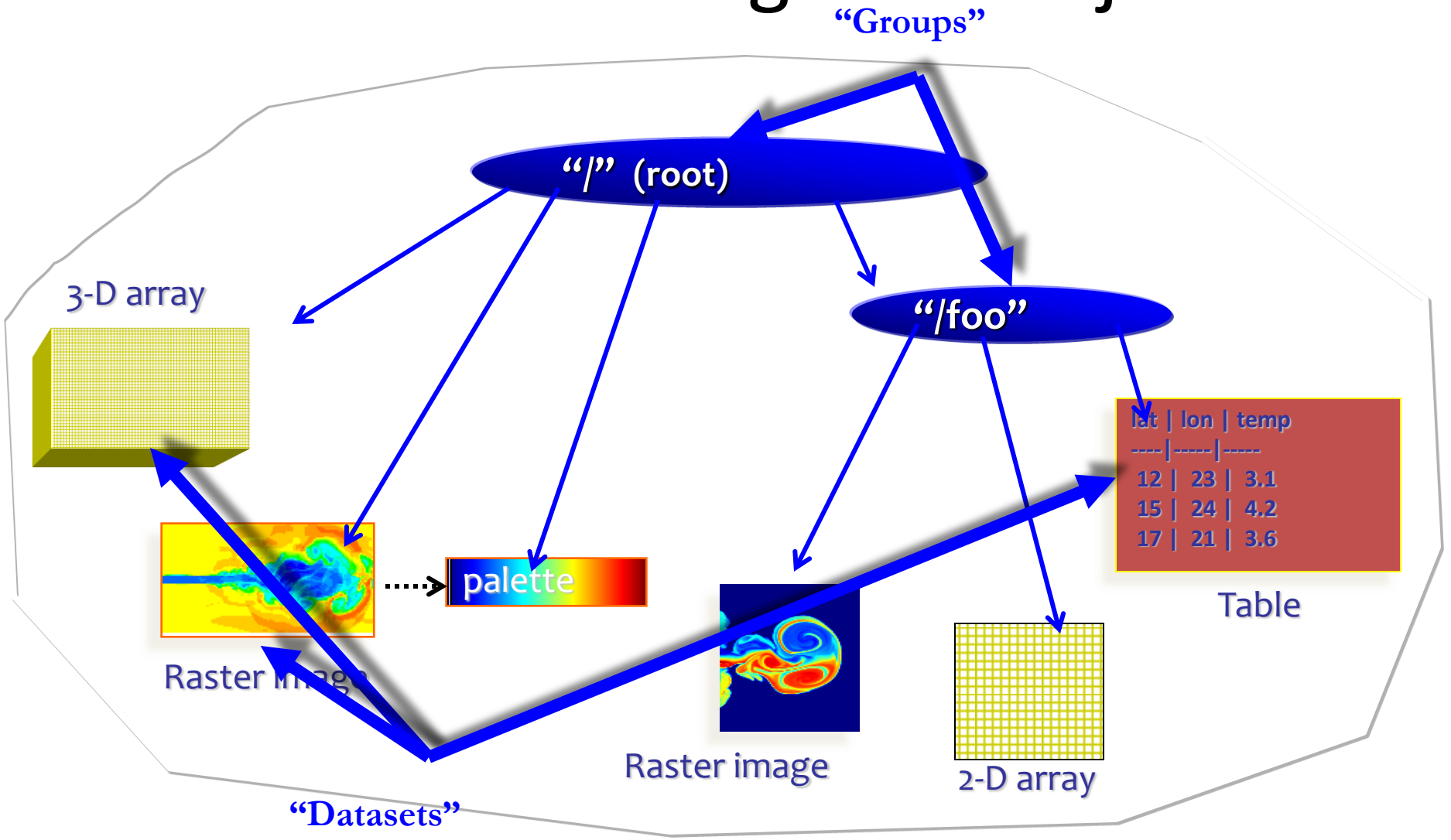
HDF5 is like…


HDF5
Databases
random access; subsetting
Directories & Files
hierarchical; collections of related information
PDF
standard exchange format; heterogeneous information
Binary "Flat File"
high-performance; compact; scalable
XML
self-describing; extensible types; rich metadata

An HDF5 “file” is a container…

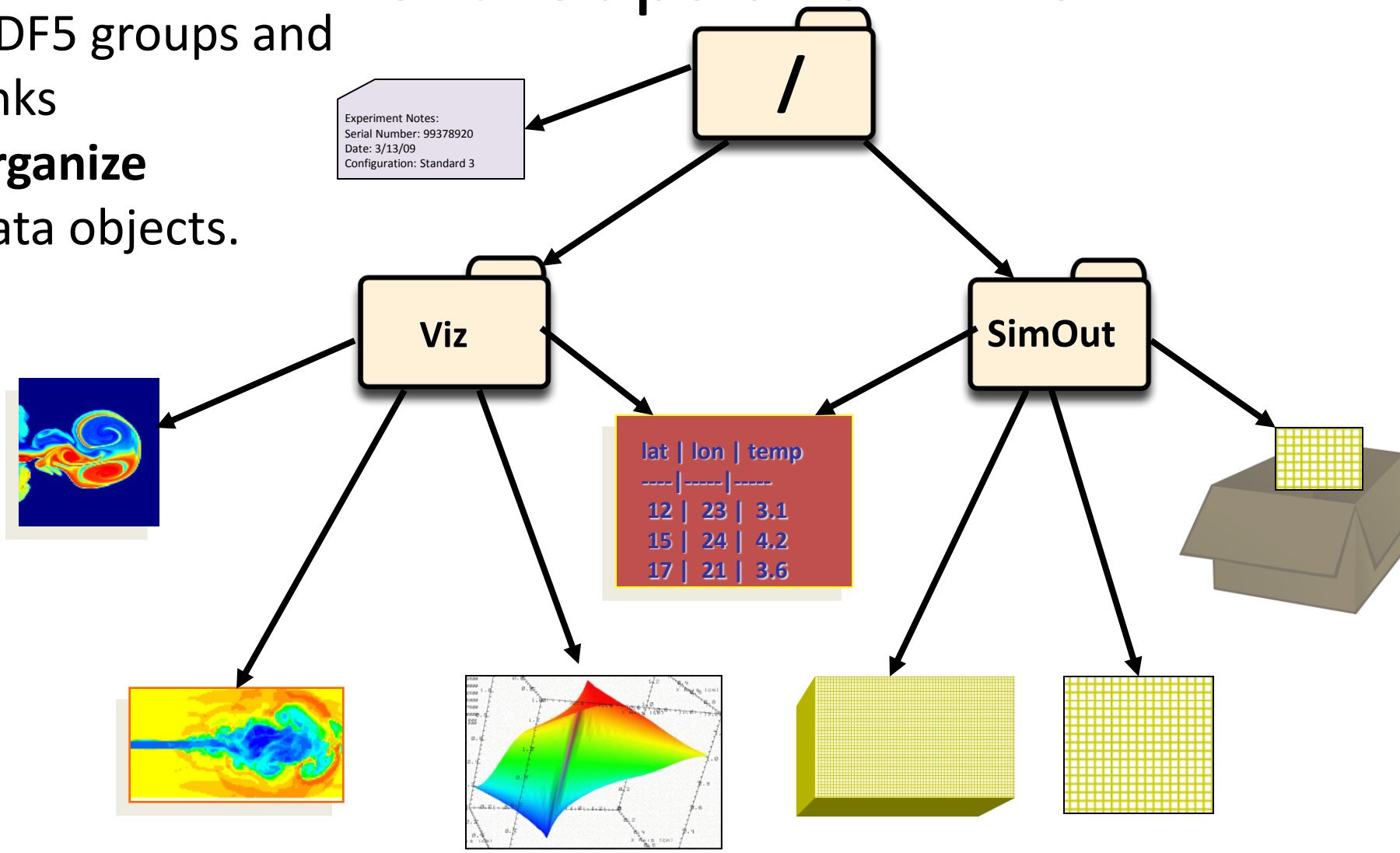
…into which you can put your data objects

Structures to organize objects



HDF5 Groups and Links

HDF5 groups and links **organize** data objects.



HDF5 model

Groups – provide structure among objects

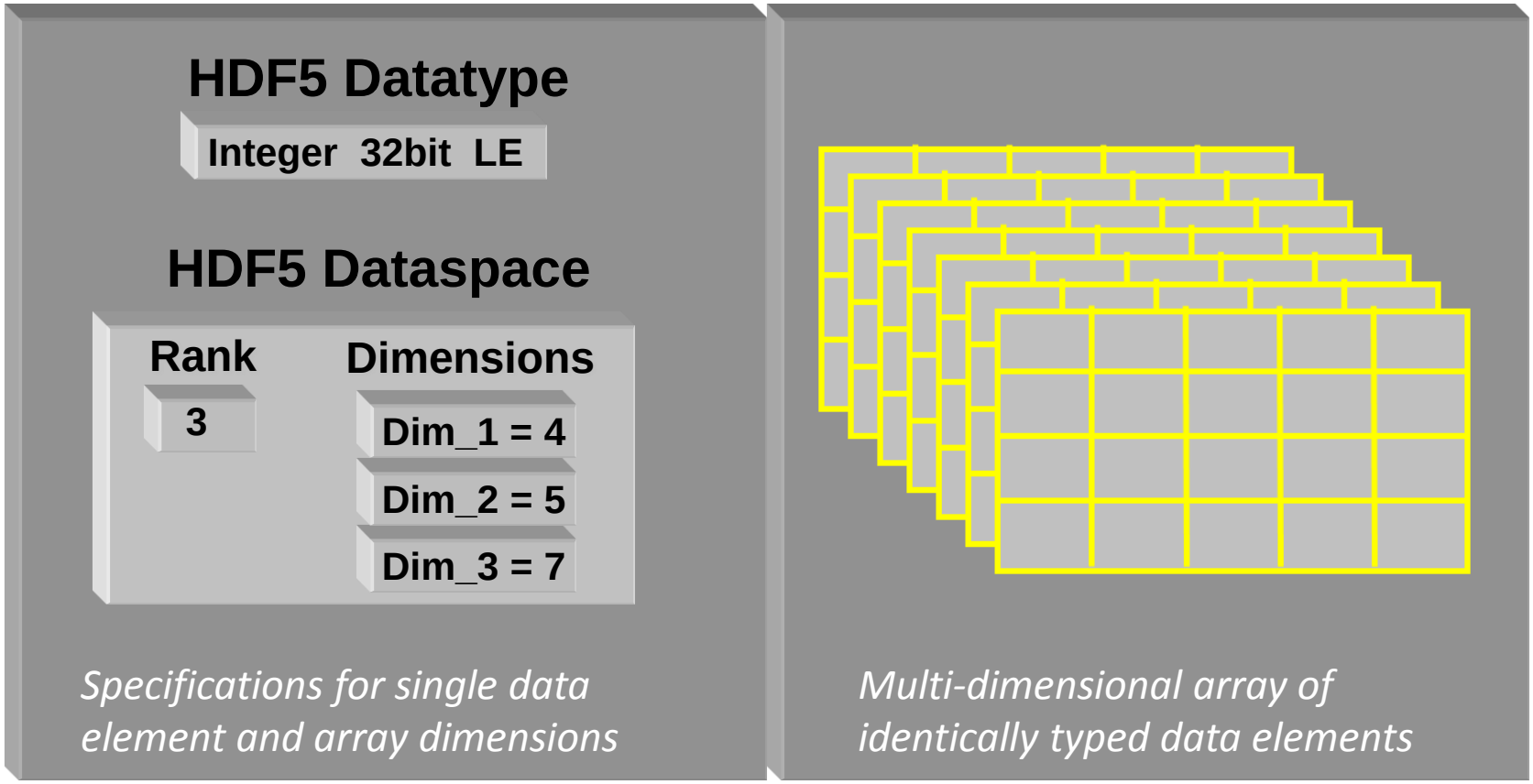
- Contain Links (point to object in the file or in another HDF5 file)

Datasets – where the primary data goes

- Data arrays
- Rich set of datatype options
- Flexible, efficient storage and I/O

Attributes, for metadata

HDF5 Dataset



- HDF5 datasets **organize and contain** "raw data values".
  - HDF5 datatypes describe individual data elements.
  - HDF5 dataspaces describe the logical layout of the data elements.

HDF5 Datatype

HDF5 atomic types include

- normal integer & float
- user-definable (e.g., 13-bit integer)
- variable length types (e.g., strings)
- references to objects/dataset regions
- enumeration - names mapped to integers
- array

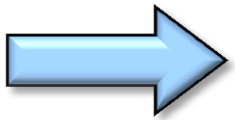
HDF5 compound types

- Comparable to C structs (“records”)
- Members can be atomic or compound types

Mapping Structs to HDF5

- Storing Vector Types in HDF5
  - Using the H5T compound data type:

```
struct CartesianVector
{
   double x,y,z;
};
```



```
hid_t id = H5Tcreate(H5T_COMPOUND,
                     sizeof(CartesianVector) );
H5Tinsert( id, "x", 0, H5T_DOUBLE);
H5Tinsert( id, "y",   sizeof(double), H5T_DOUBLE);
H5Tinsert( id, "z", 2*sizeof(double), H5T_DOUBLE);
```

This creates a HDF5 internal memory layout for the 'new' type.

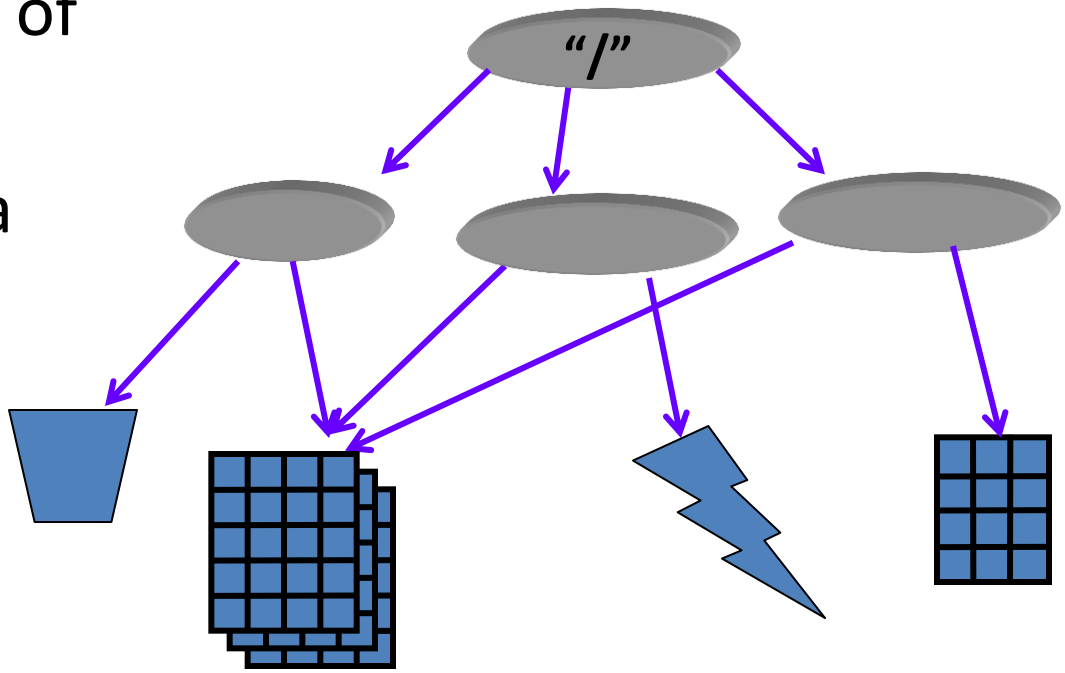
When reading or writing data HDF5 requires a type identifier and the void pointer to the memory location containing the data.

HDF5 Attributes

- An HDF5 attribute has a name and a value
- Attributes typically contain user metadata
- Attributes may be associated with
  - HDF5 groups
  - HDF5 datasets
  - HDF5 named datatypes
- An attribute's value is described by a datatype and a dataspace – analogous to a dataset.
  - Attributes are limited to < 64KB.

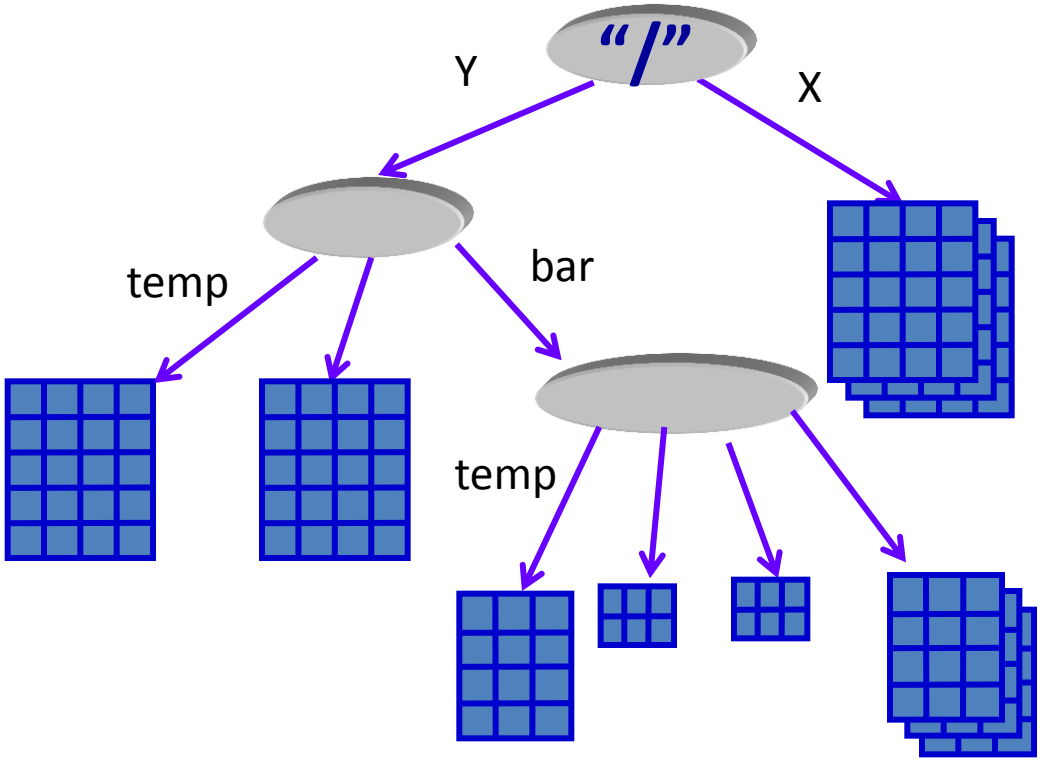
HDF5 Group

- A mechanism for organizing collections of related objects
- Every file starts with a root group
- Similar to UNIX directories
- Can have attributes



Path to HDF5 object in a file

/ (root)
/X
/Y
/Y/temp
/Y/bar/temp



Example Dataset

- The internal HDF5 file organisation made visible using the h5ls command: h5ls –rv file.h5

```
/Block00001 Dataset {5/5, 13/13, 9/9}
    Location: 1:15768
    Links:      1
    Storage:    7020 allocated bytes
    Type:       struct {
                    "x"       +0    native float
                    "y"       +4    native float
                    "z"       +8    native float
                } 12 bytes
    Data:
      (0,0,0) {0.210951, -0.0406732, 0.0611351},
              {0.210204, -0.0443333, 0.0611199},
              {0.209324, -0.0483009, 0.0611070},
              {0.208286, -0.0525892, 0.0610958}
      (0,0,4) {0.207065, -0.0571980, 0.0610863},
              {0.205640, -0.0621138, 0.0610815},
```

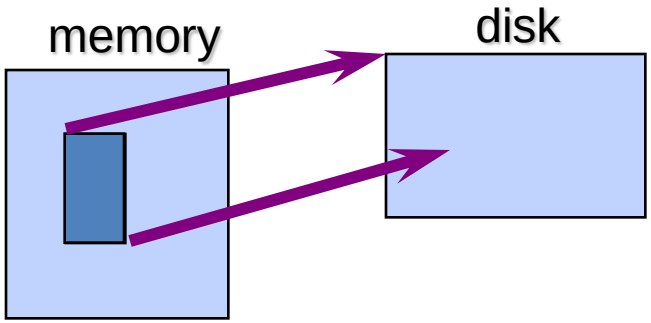
HDF5 allows easily to attach names to the numerical values.

HDF5 takes care of endianess, double to float conversion and different layouts, e.g. {x,y,z} vs {z,x,y}

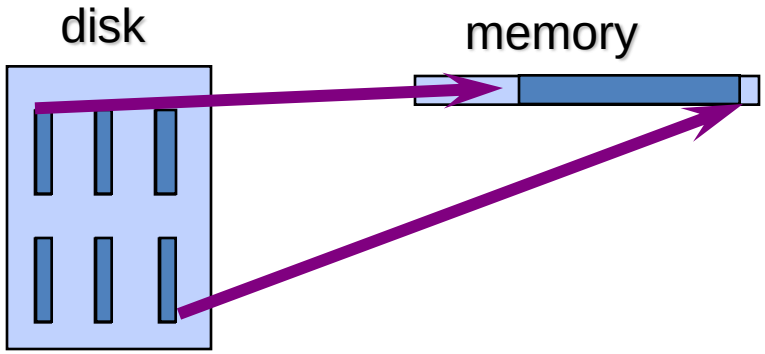
The naming scheme can also serve to identify the coordinate system rel. to which the numbers are stored e.g. {x,y,z} vs {r, phi, theta}

HDF5 Hyperslabs: Partial I/O

Move just part of a dataset



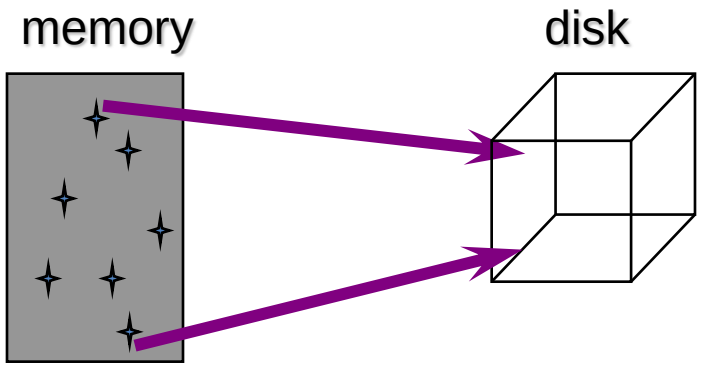
(a) Hyperslab from a 2D array to the corner of a smaller 2D array



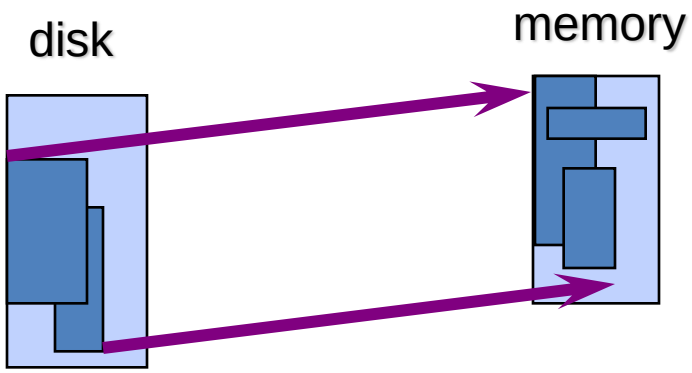
(b) Regular series of blocks from a 2D array to a contiguous sequence at a certain offset in a 1D array

Partial I/O

Move just part of a dataset



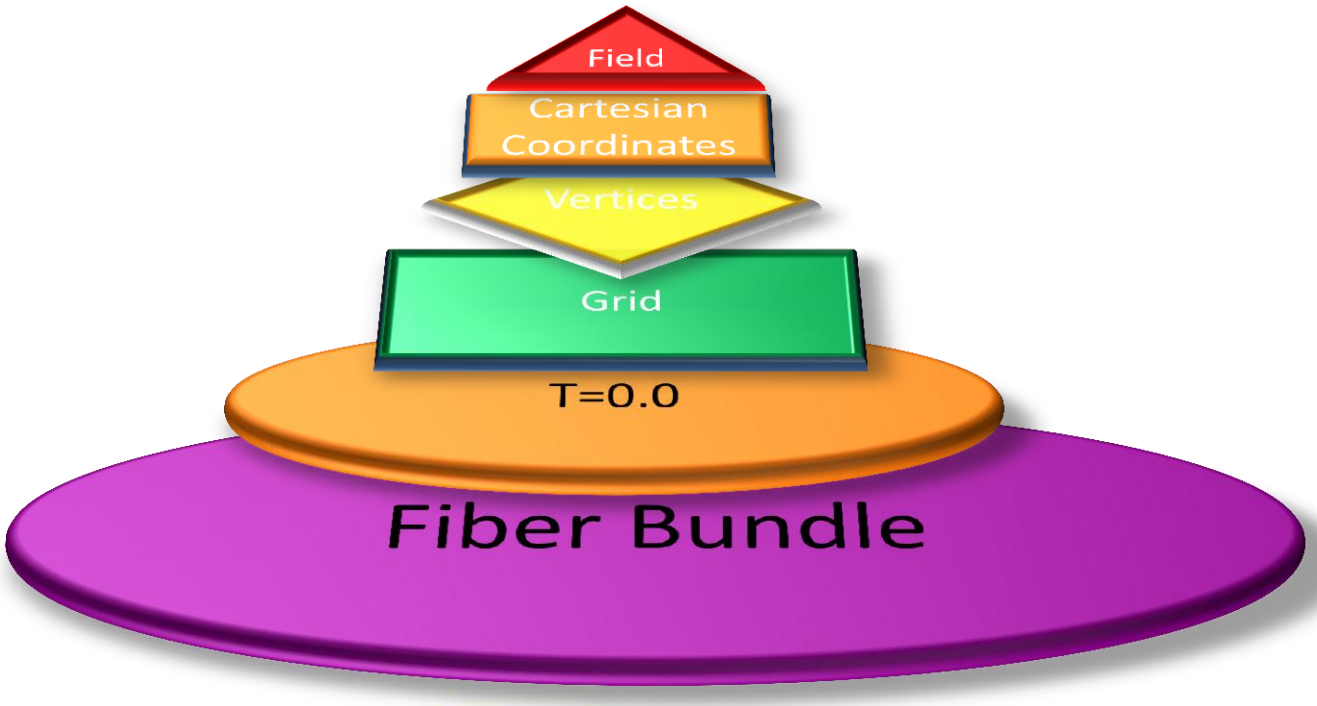
(c) A sequence of points from a 2D array to a sequence of points in a 3D array.



(d) Union of hyperslabs in file to union of hyperslabs in memory.

The F5 Data Model

- Metadata Layer on top of HDF5



1. **Field**
2. Representation
3. Topological Skeleton
4. **Grid object**
5. (Time) Slice
6. **Bundle**

The HDF Group

Homework: Install Vish and simple Viz network (grading 100%)

Optional Advanced: Explore Volume rendering (no grading, only influence on borderline grades)

HOMEWORK D-1 ASSIGNMENTS (DUE OCT 7TH)

Install Vish

- Install Vish as a working environment
  - Select an appropriate platform
    - Computers in the LSU Middleton Library
    - Your laptop
  - Download and compile

    http://sciviz.cct.lsu.edu/projects/vish/

    http://sciviz.cct.lsu.edu/projects/vish/CSC7700/Vish.tar.gz

Build a Visualization Network

- Create a visualization of a volumetric data set
  - Include an Isosurface rendering to the example
- Load Data from disk file
  - Small: Run “make mkdata” in Vish directory, use file data/TimeDependentScalar.f5
  - HUGE (16GB): go to http://sciviz.cct.lsu.edu/data/, download “MergingNeutronStars.f5”
- Provide Vish Script and Screenshots

Volume Rendering

- Find Volume Rendering Display object on data field
- Setup parameters (colormap, range, …)
  - Demonstrate that volume rendering is done by slicing the data object
- Find “VideoRecorder”
  - Make a movie (image sequence)
  - Post to a web page

Homework D1
Clarifications & Grading

- Ideally submit a .pdf or .doc/.docx file with text & images included
- Describe which platforms you used to install vish, which problems you encountered and how you solved them - 40%
- Include a screenshot of your viz network and the Isosurface of the dataset - 20%
- Include a screenshot of the TimeDependentScalar.f5 dataset and a listing of the viz network that *you* created; describe *how* you built this network. It needs to be loadable from Vish (e.g. identifiers must not include special characters such as "<", ">") - 20%
- Include a screenshot of the MergingNeutronStars.f5 dataset and a listing of the viz network that *you* created - 20%
- Optional volume rendering: include screenshots and discuss which parameters you explored to create them - (+50%)