

# VisMashup: Streamlining the Creation of Custom Visualization Applications\*

---

**Emanuele Santos**

University of Utah

joint work with

Lauro Lins, Juliana Freire and Cláudio Silva - *University of Utah*

James Ahrens - *Los Alamos National Lab*

\* Paper submitted to IEEE Vis 2009

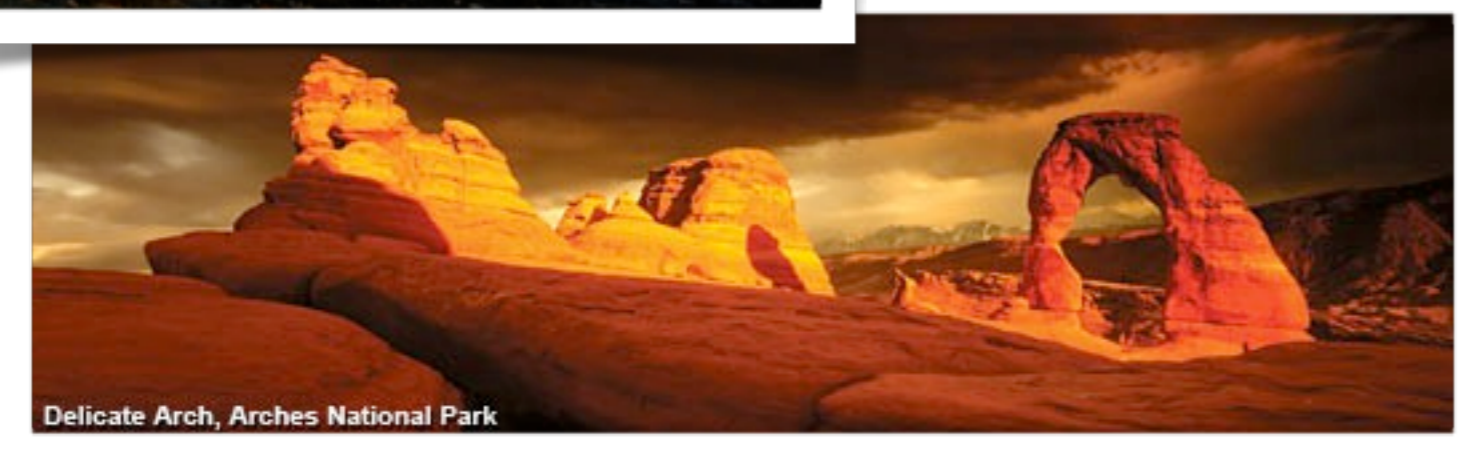
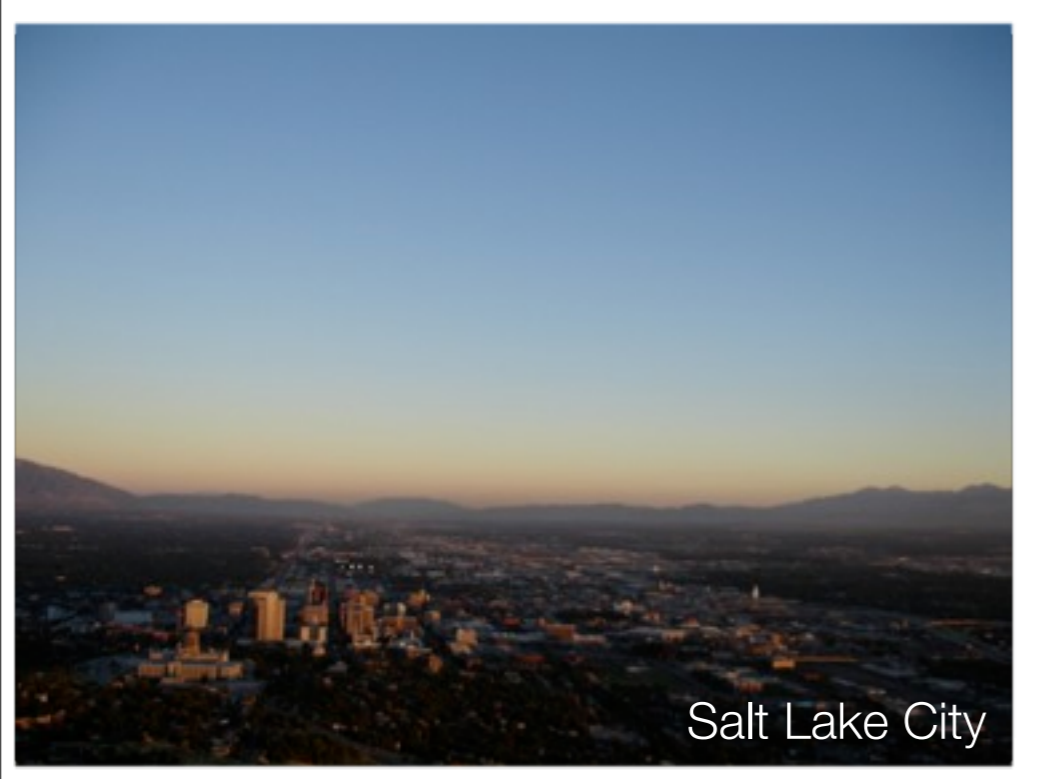
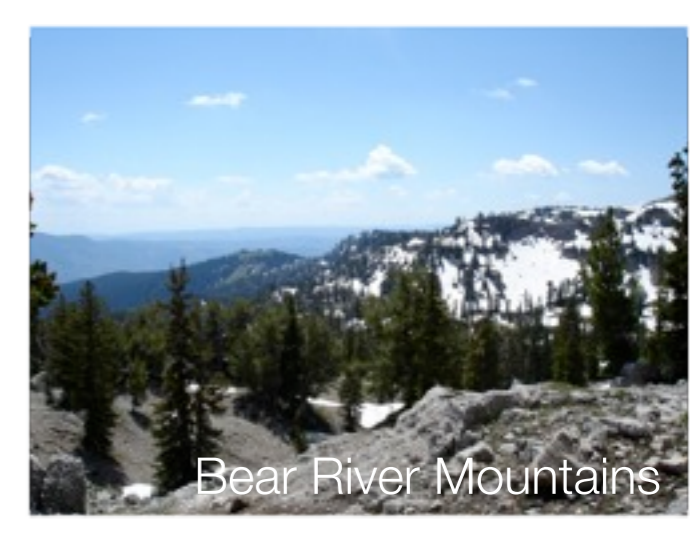
# Agenda

---

- A bit about Utah
- Introduction
- Model
- The VisMashup System
- Case Study: Sharing Astrophysics Analyses
- Limitations
- Conclusions and Future Work

# Utah

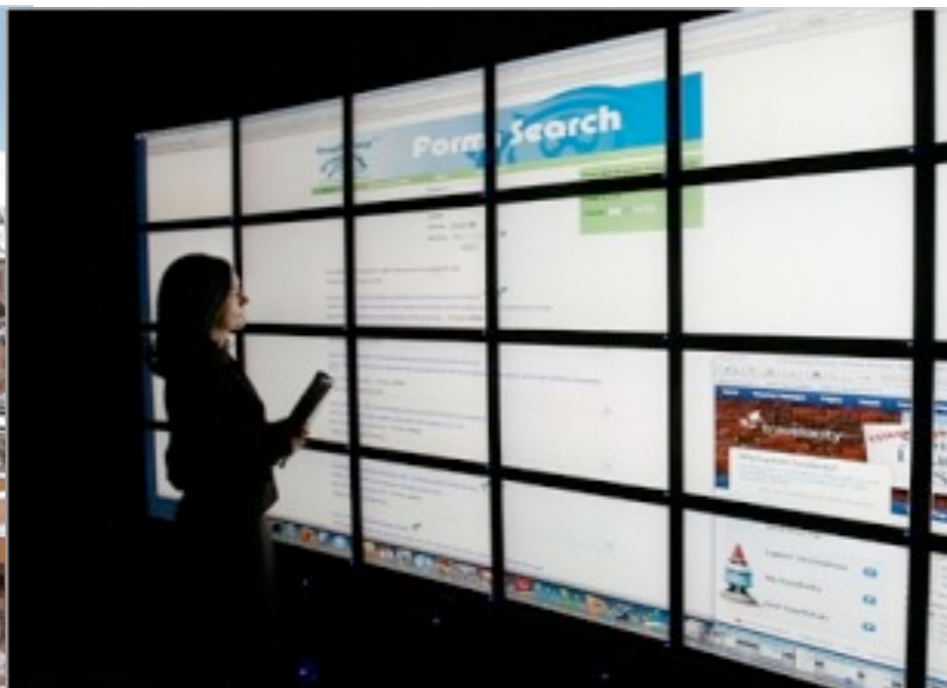
“The Greatest Snow on Earth”



# University of Utah

---

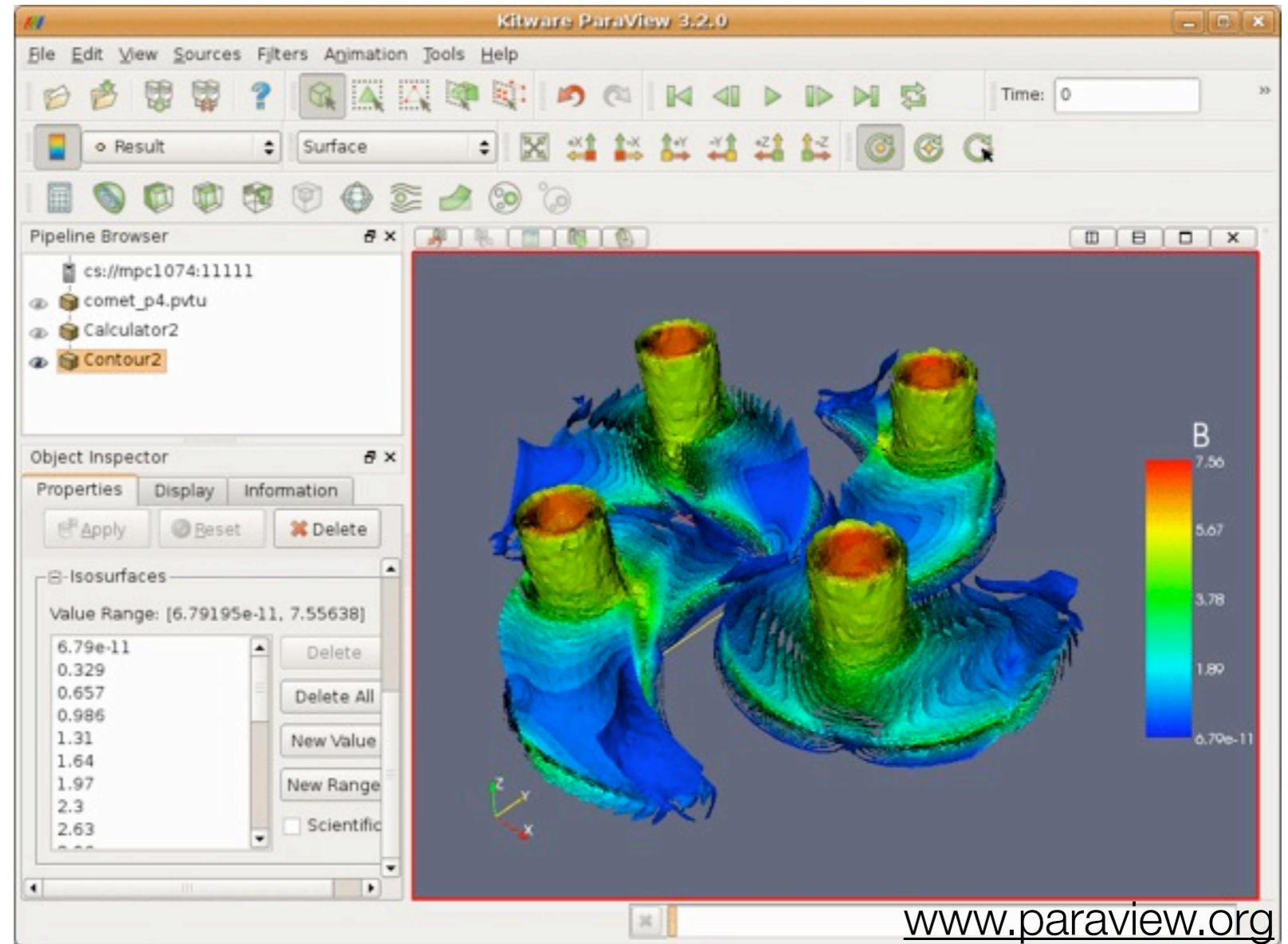
- Scientific Computing and Imaging (SCI) Institute
  - 15 faculty members and more than 100 students and post docs
- VGC Group: 3 post docs, 9 grads and 2 undergrads



# Introduction

---

# Visualization tools



Data exploration through visualization is an effective means to understand and obtain insights from large collections of data



# Visualization tools

www.vistrails.org

www.paraview.org

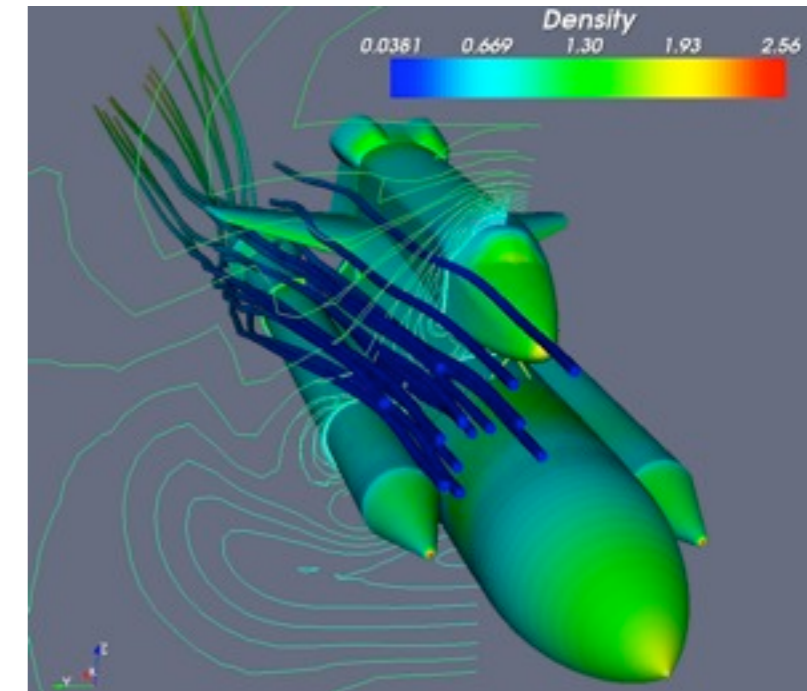
software.sci.utah.edu/scirun.html

Data exploration through visualization is an effective means to understand and obtain insights from large collections of data



# Creating visualizations

```
0265640 132304 113732 032051 037334 024721 015013 052226 001662
0265650 025537 064668 054596 043244 074076 124153 135216 125534
0265700 144210 056426 044790 042650 165230 137037 003655 006254
0265720 134453 124327 176005 027034 107624 170774 073702 063724
0265740 072451 007735 147620 061064 157435 113057 155356 114603
0265760 107204 102316 171451 046040 120228 001774 030477 046673
0266000 171817 116055 155117 134444 167210 041405 147127 050505
0266020 004137 046472 124015 134360 173550 053517 044635 021135
0266040 070176 047705 113754 175477 105532 076535 177366 056333
0266060 041023 074017 127113 003214 037026 037640 066171 123424
0266100 067701 037406 140000 165341 072410 100032 125455 056646
0266120 006716 071402 055672 132571 105645 170073 050376 072117
0266140 024451 007424 114200 077733 024434 012546 172404 102345
0266160 040223 050170 055164 164634 047154 126525 112514 032315
0266200 016041 176055 042766 025015 176314 017234 110060 014515
0266220 117154 030746 154234 125001 151344 163706 136237 164376
0266240 137055 062276 161755 115466 005322 132567 073216 002655
0266260 173466 126161 117155 065763 016177 014460 112765 055527
0266300 003767 175367 104754 036436 172172 150750 043643 145410
0266320 072094 000007 040627 070652 173011 002153 125132 140214
0266340 060115 014356 015164 067027 120206 070242 033065 131134
0266360 170601 170106 040437 127277 124446 136533 041462 115321
0266400 020343 005602 004146 121574 124651 006634 071381 102070
0266420 157504 160307 166330 074251 024520 114433 167273 030635
0266440 133614 106171 144160 030652 007365 026416 160716 100413
0266460 026630 007210 000630 121224 076033 140764 000787 003276
0266500 114060 042647 104475 110537 066716 104754 075447 112254
0266520 030374 144251 077734 035157 002513 173526 035531 150003
0266540 146207 015135 024446 130101 072467 040764 165513 156412
0266560 166410 067251 156160 106406 136770 030516 064740 022032
0266600 142166 123707 175121 071170 076357 037233 031136 015232
0266620 075074 016744 044055 102230 110063 033350 052765 172463
```



Raw Data

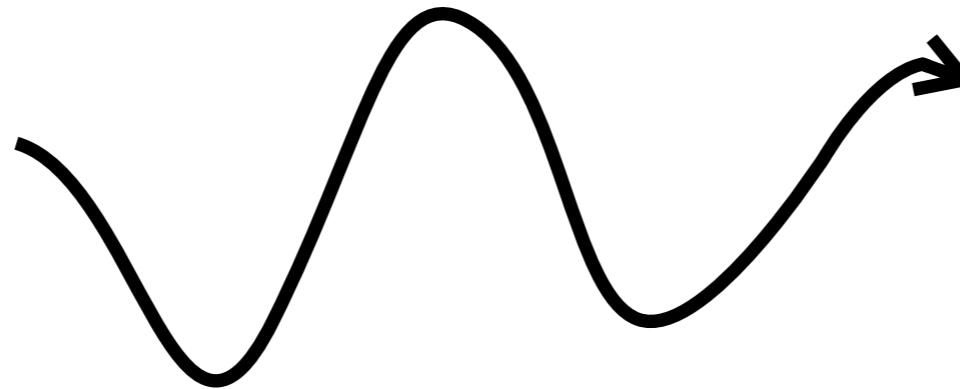
Insightful  
Visualization

# Creating visualizations

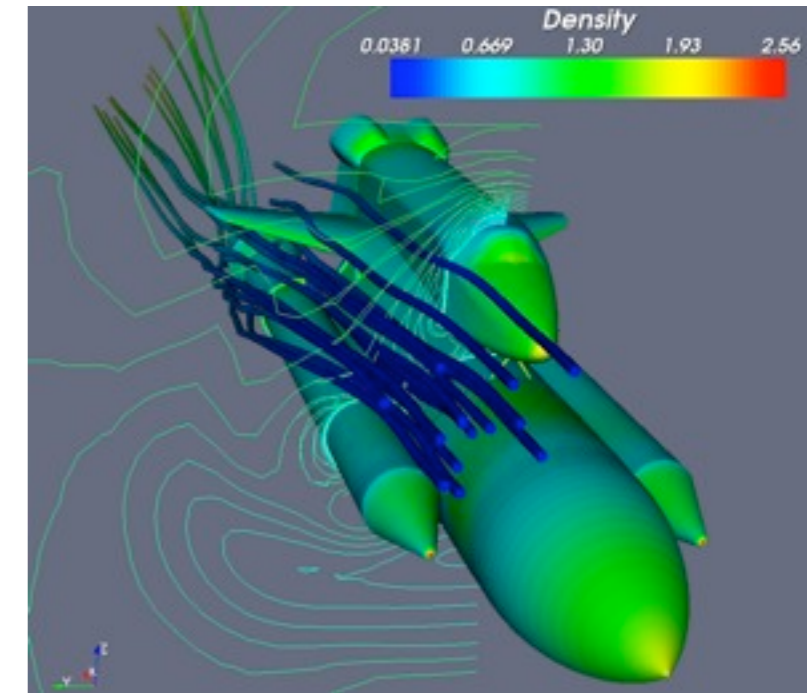
```
0265640 132304 113732 032051 037334 024721 015013 052226 001662
0265650 025537 064668 054596 043244 074076 124153 135216 125534
0265700 144210 056426 044790 042650 165230 137037 003655 006254
0265720 134453 124327 176005 027034 107624 170774 073702 063724
0265740 072451 007735 347620 061064 157435 113057 155356 114603
0265760 107204 102316 171451 046040 120228 001774 030477 046673
0266000 171817 116055 155117 134444 167210 041405 147127 050505
0266020 004137 046472 124015 134360 173550 053517 044635 021135
0266040 070176 047705 113754 175477 105532 076535 177366 056333
0266060 041023 074017 127113 003214 037026 037640 066171 123424
0266100 067701 037406 140000 165341 072410 100032 125455 056646
0266120 006716 071402 055672 132571 105645 170073 050376 072117
0266140 024451 007424 114200 077733 024434 012546 172404 102345
0266160 040223 050170 055164 164634 047154 126525 112514 032315
0266200 016041 176055 042766 025015 176314 017234 110060 014515
0266220 117154 030746 154234 125001 151344 163706 136237 164376
0266240 137055 062276 161755 115466 005322 132567 073216 002655
0266260 173466 126361 117155 065763 016177 014460 112765 055527
0266300 003767 175367 104754 036436 172172 150750 043643 145410
0266320 072094 000007 040627 070652 173011 002153 125132 148214
0266340 060115 014356 015164 067027 120206 070242 033065 131134
0266360 170601 170106 040437 127277 124446 136531 041462 115321
0266400 020243 005602 004146 121574 124651 006634 071381 102070
0266420 157504 169307 166330 074251 024520 114433 167273 030635
0266440 133614 106171 344160 030652 007365 026416 160716 109413
0266460 026630 007210 000630 121224 076033 140764 000787 003276
0266500 134060 042647 104475 110537 066716 104754 075447 112254
0266520 030374 144251 077734 035157 002513 173526 035531 150003
0266540 146207 015135 024446 130101 072467 040764 165513 156412
0266560 166410 067251 156160 106406 136770 030516 064740 022032
0266600 142166 123707 175121 071170 076357 037233 031136 015232
0266620 075074 016744 044055 102230 110063 033350 052765 172463
```

Raw Data

laborious



error-prone



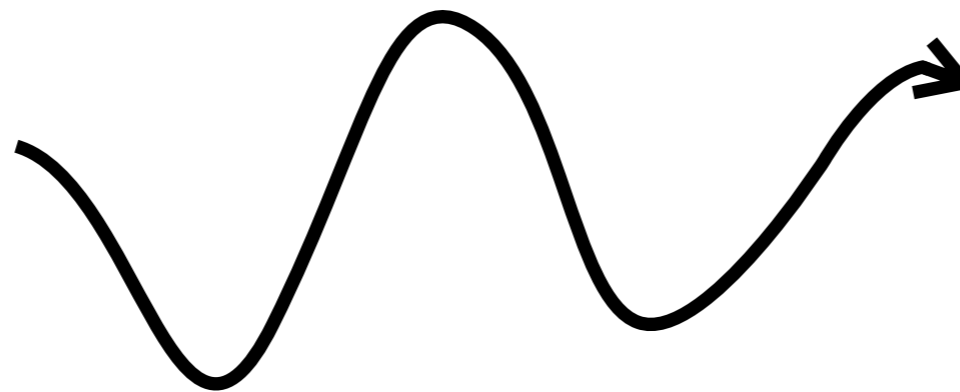
Insightful Visualization

# Creating visualizations

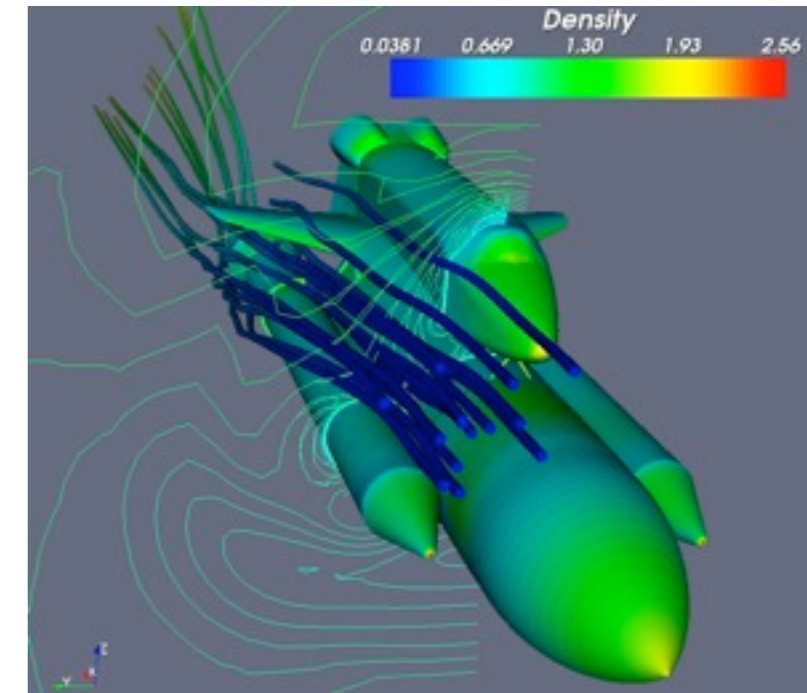
```
0265640 132304 113732 032051 037334 024721 015013 052226 001662
0265650 025537 064663 054596 043244 074076 124153 135216 125534
0265700 144210 056426 044790 042650 165230 137037 003655 006254
0265720 134453 124327 176005 027034 107624 170774 073702 063724
0265740 072451 007735 347620 061064 157435 113057 155356 114603
0265760 107204 102316 171451 046040 120223 001774 030477 046673
0266000 171817 116055 155117 134444 167210 041405 147127 050505
0266020 004137 046472 124015 134360 173550 053517 044635 021135
0266040 070176 047705 113754 175477 105532 076535 177366 056333
0266060 041023 074017 127113 003214 037026 037640 066171 123424
0266100 067701 037406 140000 165341 072410 100032 125455 056646
0266120 006716 071402 055672 132571 105645 170073 050376 072117
0266140 024451 007424 114200 077733 024434 012546 172404 102345
0266160 040223 050170 055164 164634 047154 126525 112514 032315
0266200 016041 176055 042766 025015 176314 017234 110060 014515
0266220 117154 030746 154234 125001 151344 163706 136237 164376
0266240 137055 062276 161755 115466 005322 132567 073216 002655
0266260 173466 126161 117155 065763 016177 014460 112765 055527
0266300 003767 175367 104754 036436 172172 150750 043643 145410
0266320 072094 000007 040627 070652 173011 002153 125132 148214
0266340 060115 014356 015164 067027 120206 070242 033065 131134
0266360 170601 170106 040437 127277 124446 136533 041462 115321
0266400 020243 005602 004146 121574 124651 006634 071381 102070
0266420 157504 169307 166330 074251 024520 114433 167273 030635
0266440 133614 106171 344160 030652 007365 026416 160716 109413
0266460 026630 007210 000630 121224 076033 140764 000787 003276
0266500 134060 042647 104475 110537 066716 104754 075447 112254
0266520 030374 144251 077734 035157 002513 173526 035531 150003
0266540 146207 015135 024446 130101 072467 040764 165513 156412
0266560 166410 067251 156160 106406 136770 030516 064740 022032
0266600 142166 123707 175121 071170 076357 037233 031136 015232
0266620 075074 016744 044055 102230 110063 033350 052765 172463
```

Raw Data

laborious



error-prone



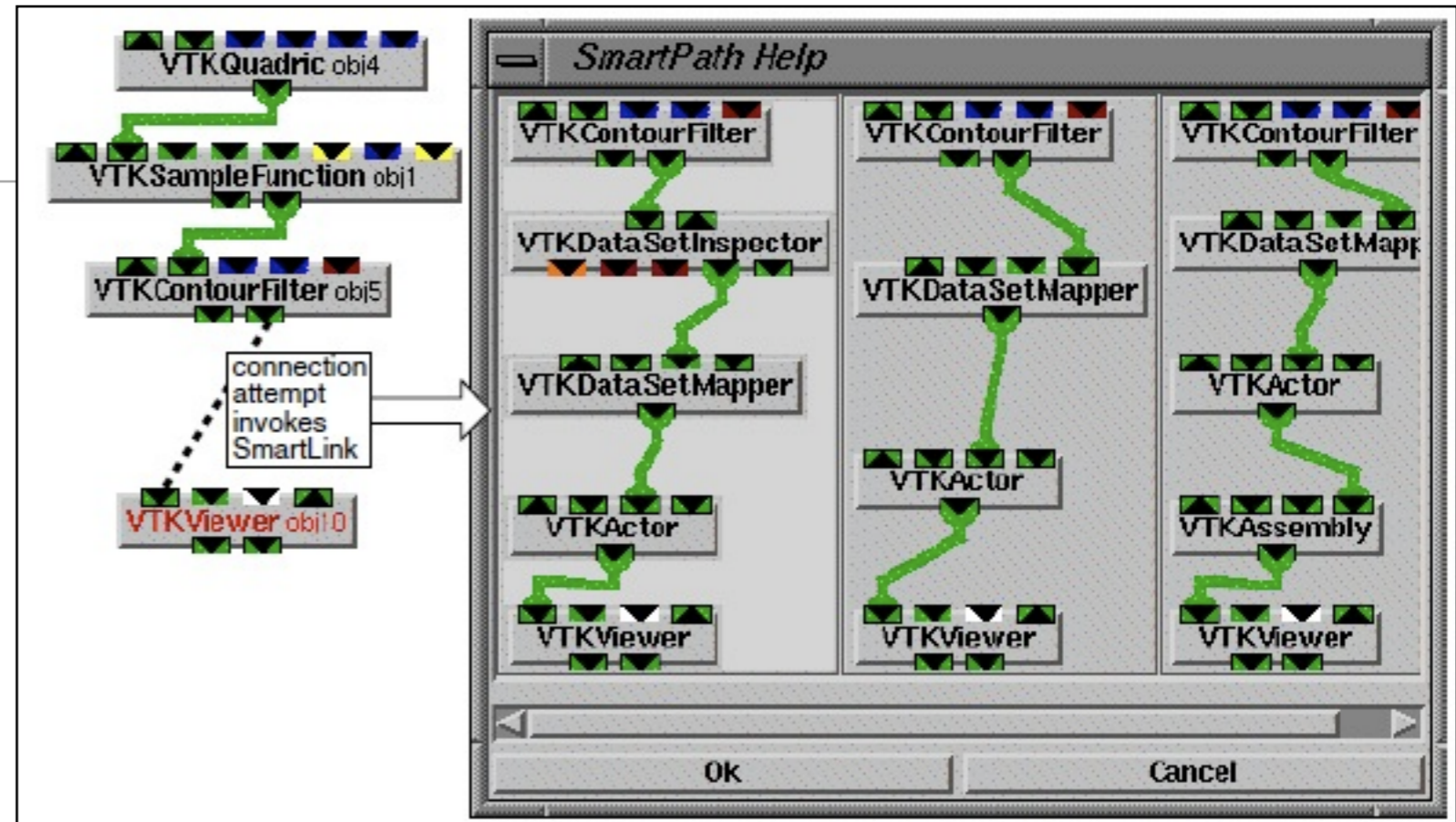
Insightful Visualization

The generation of visualizations is still best carried out by experts!

# Helping to create visualizations

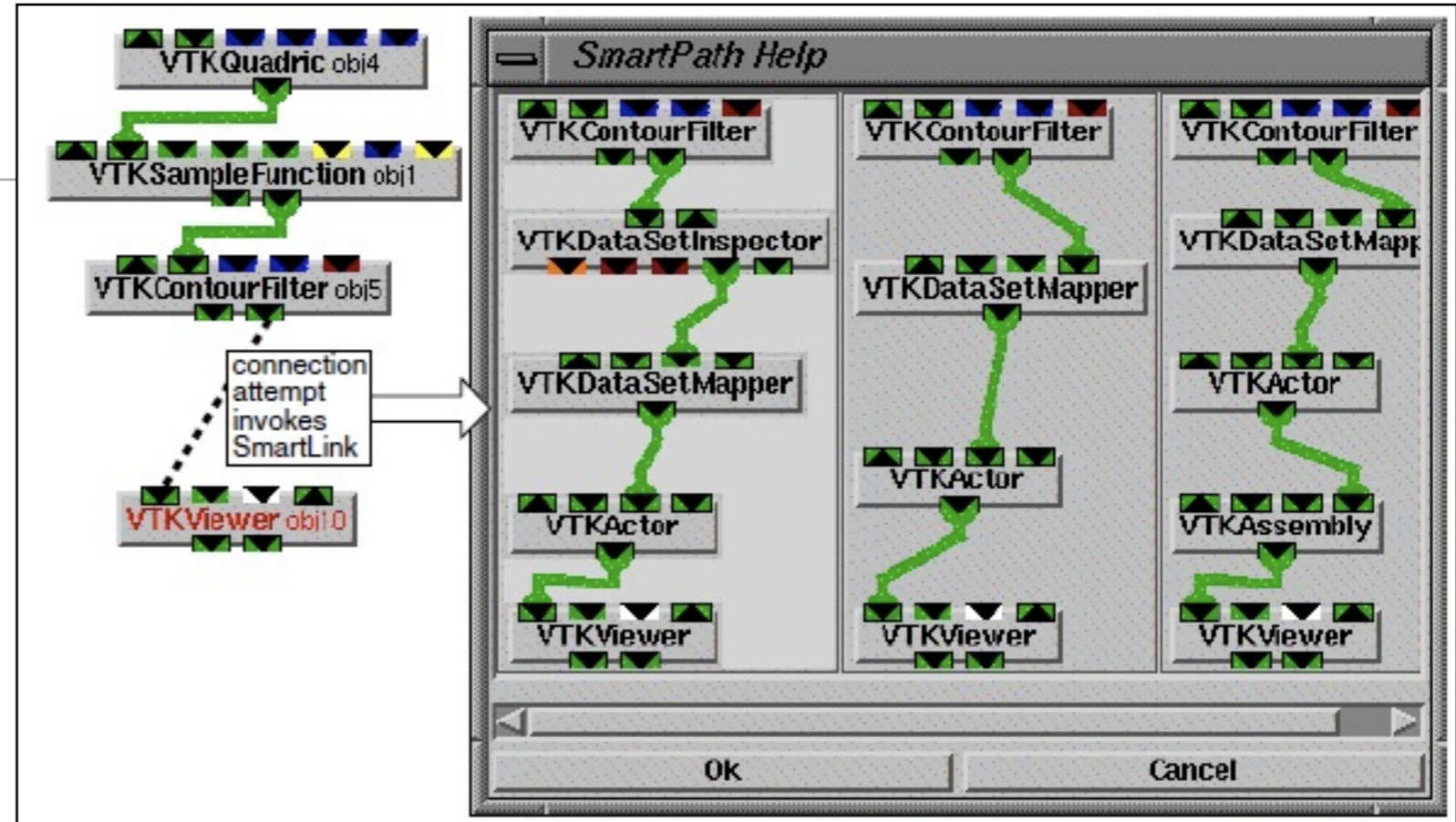
---

# Helping to create visualizations

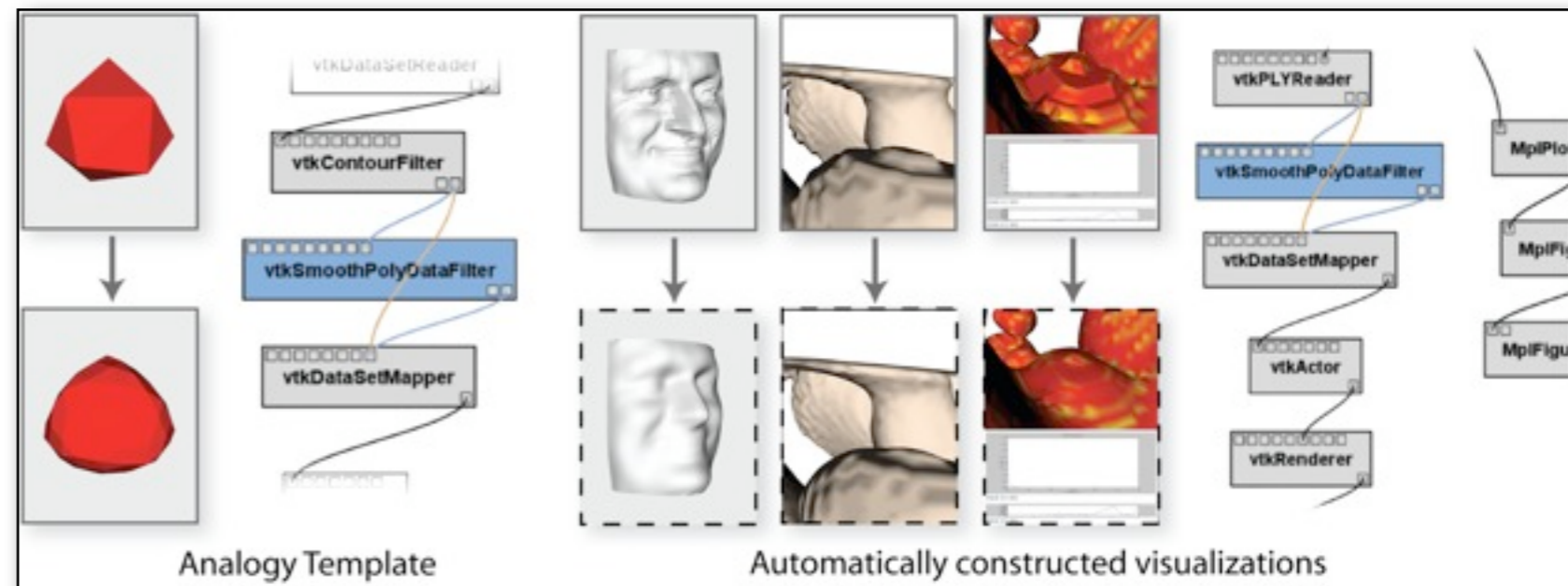


SmartLink: Telea & van Wijk, VisSym 1999

# Helping to create visualizations

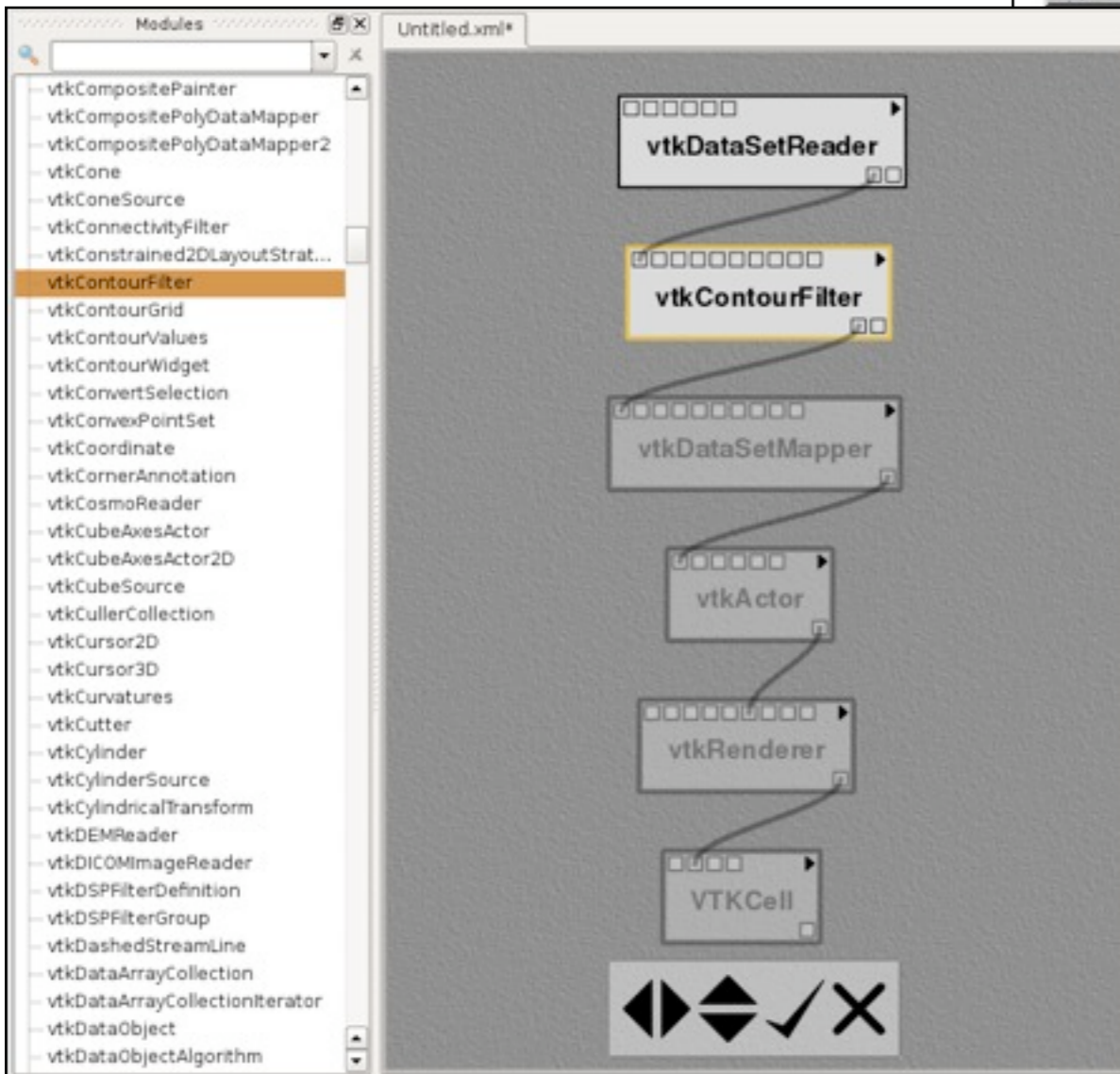


SmartLink: Telea & van Wijk, VisSym 1999

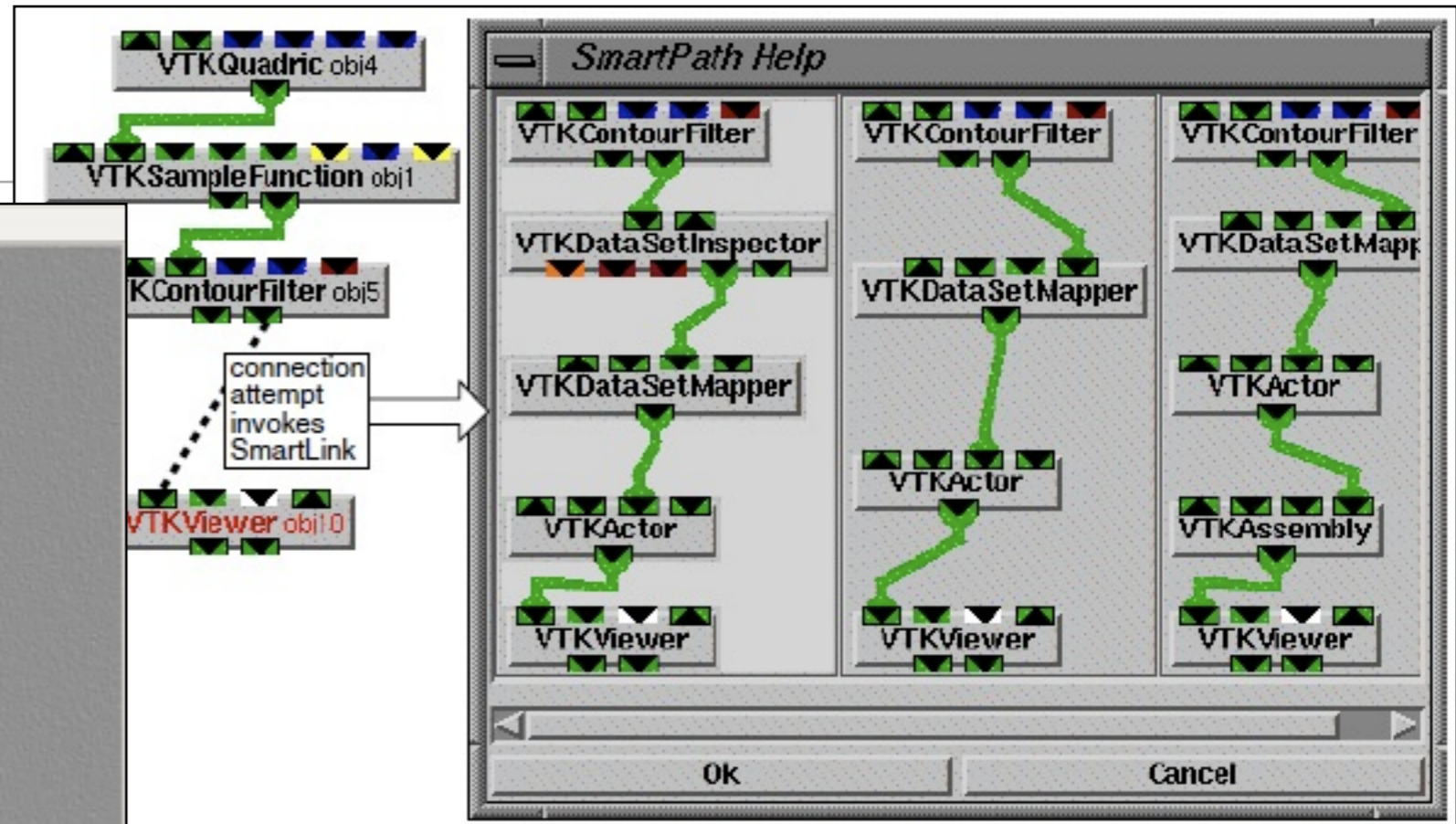


Vis by Analogy: Scheidegger et al., Vis 2007

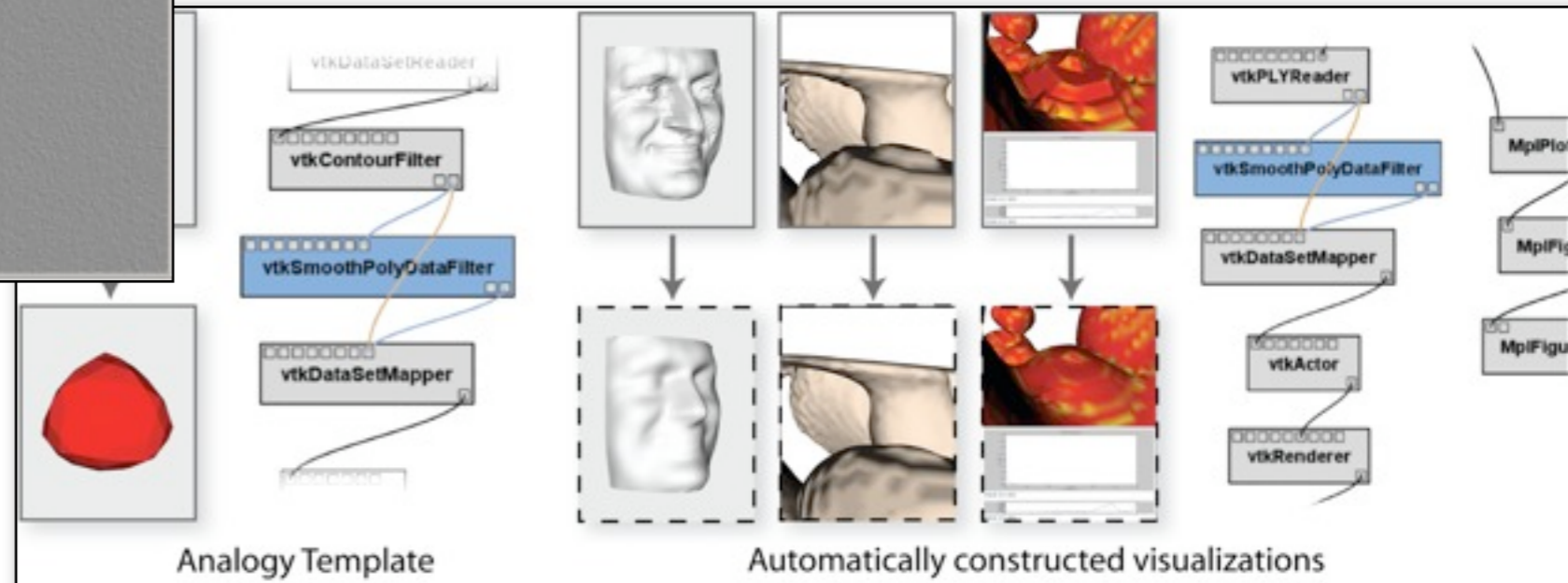
# Helping to create visualizations



VisComplete: Koop et al., Vis 2008

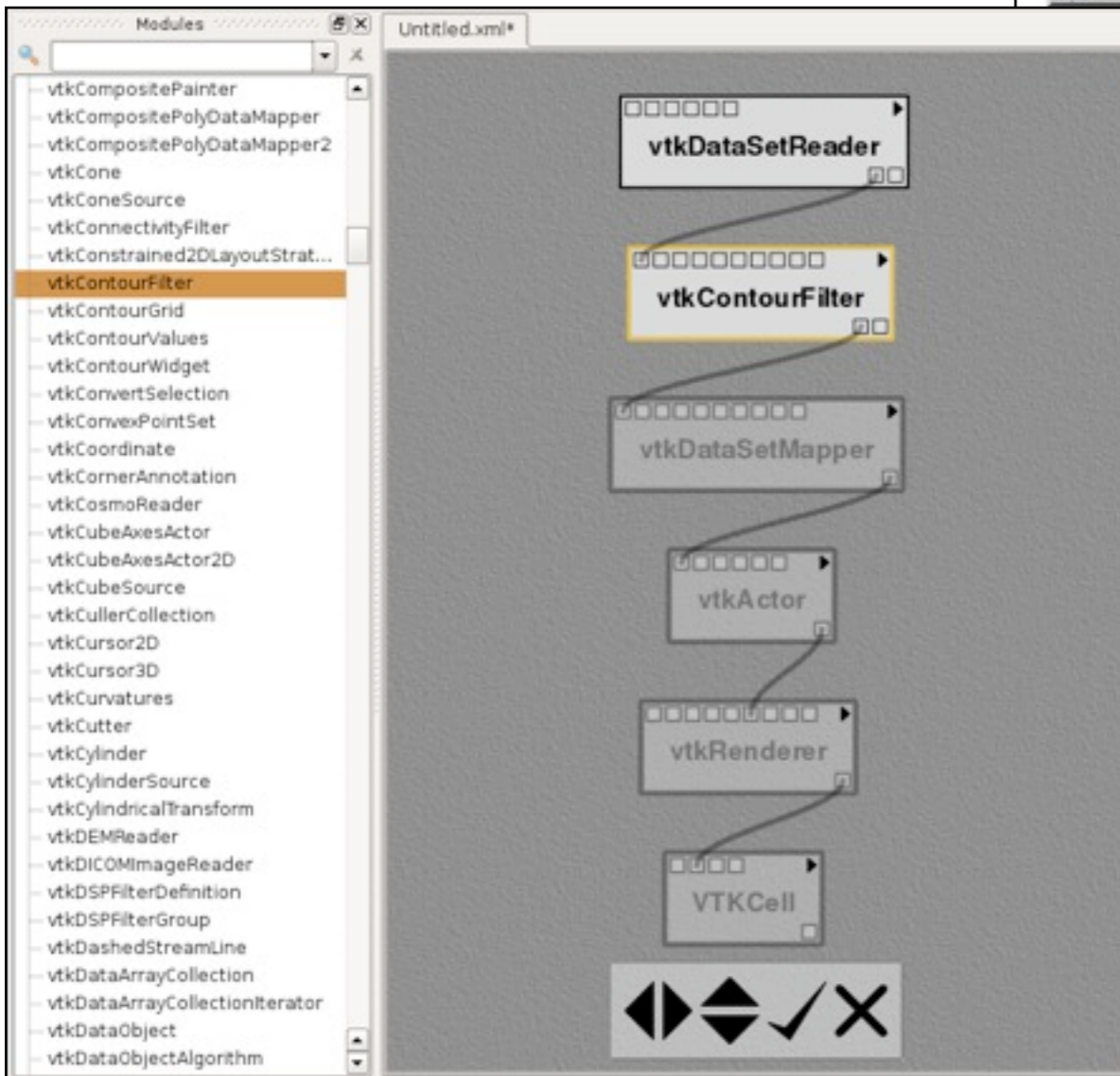


SmartLink: Telea & van Wijk, VisSym 1999



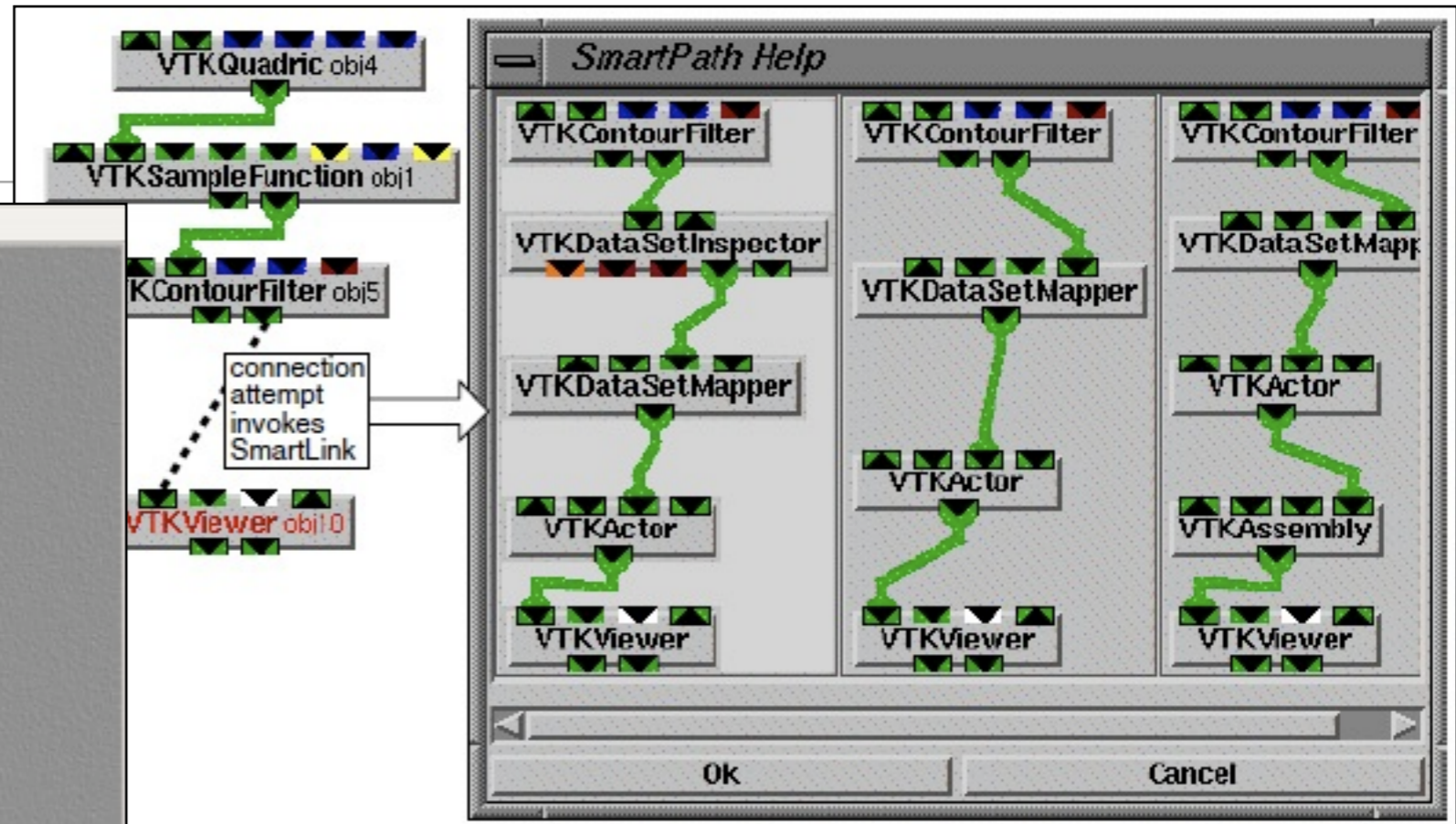
Vis by Analogy: Scheidegger et al., Vis 2007

# Helping to create visualizations

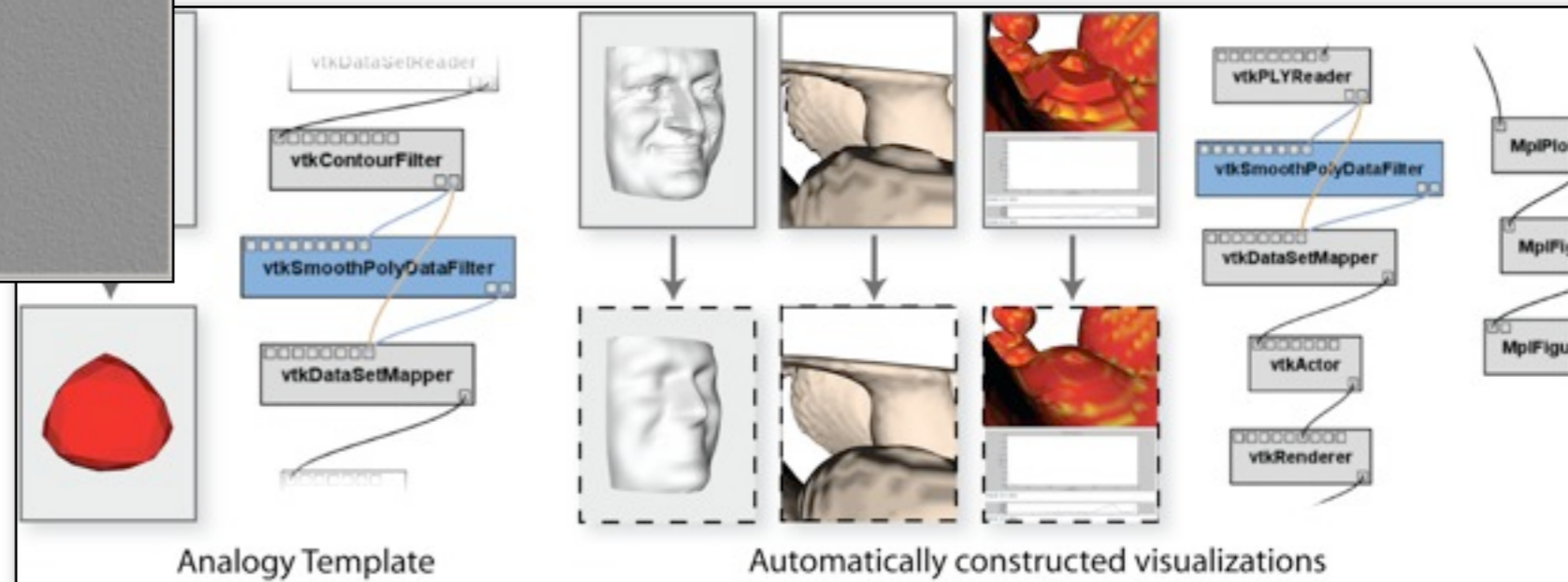


VisComplete: Koop et al., Vis 2008

Detailed knowledge of the underlying computational components is still necessary



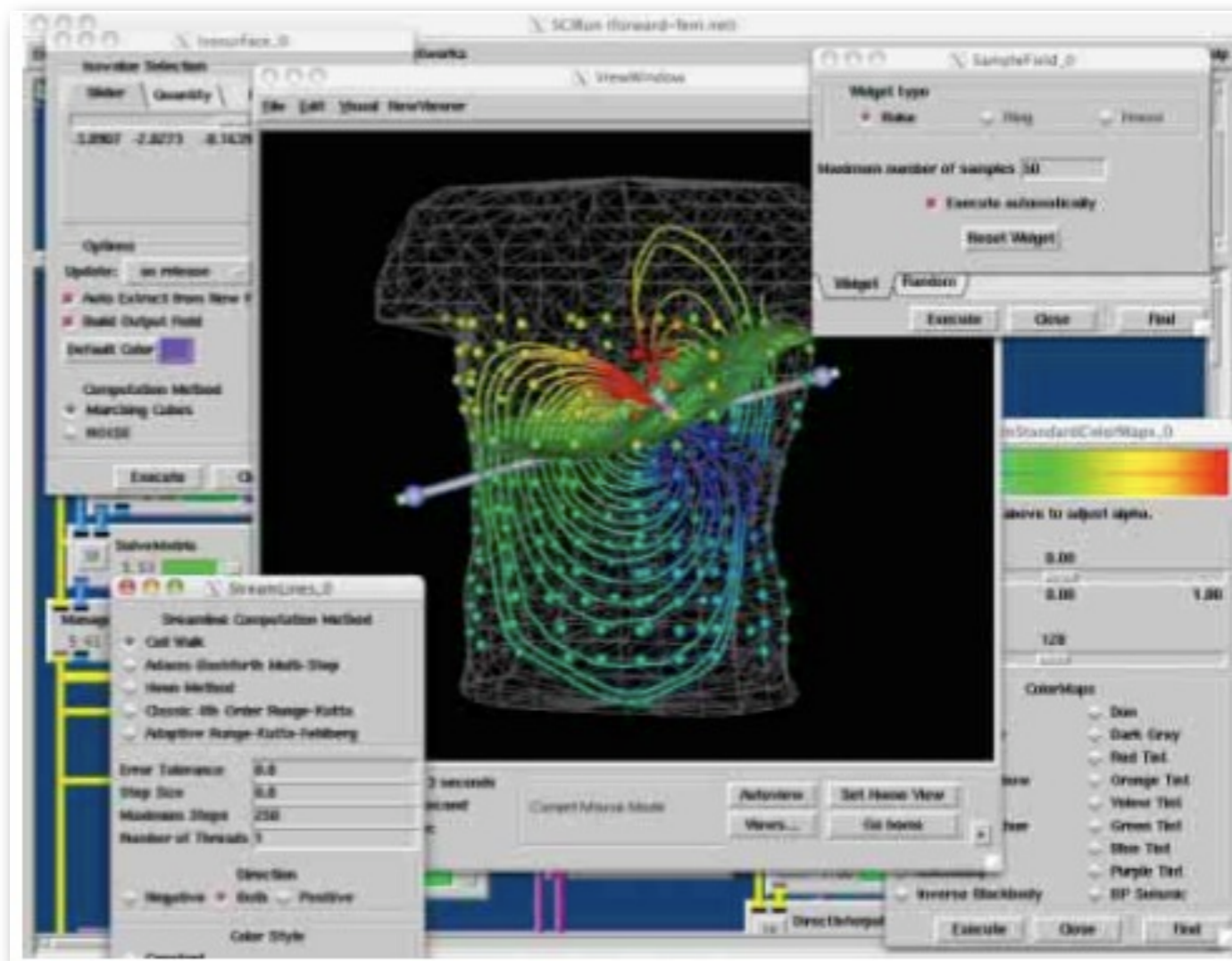
SmartLink: Telea & van Wijk, VisSym 1999



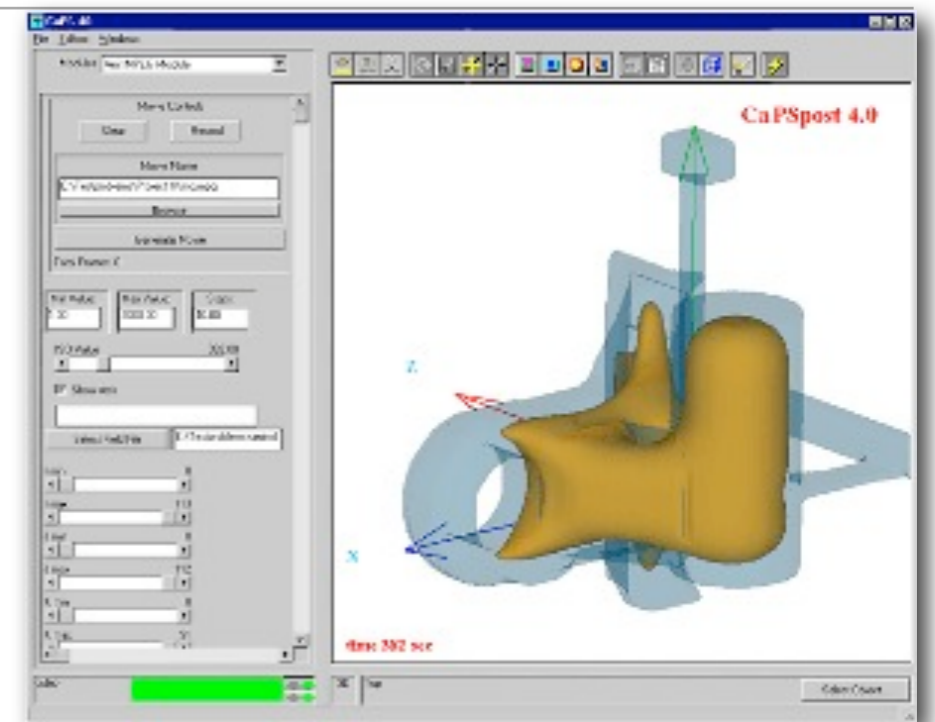
Vis by Analogy: Scheidegger et al., Vis 2007



# Helping to create visualizations: Custom Applications

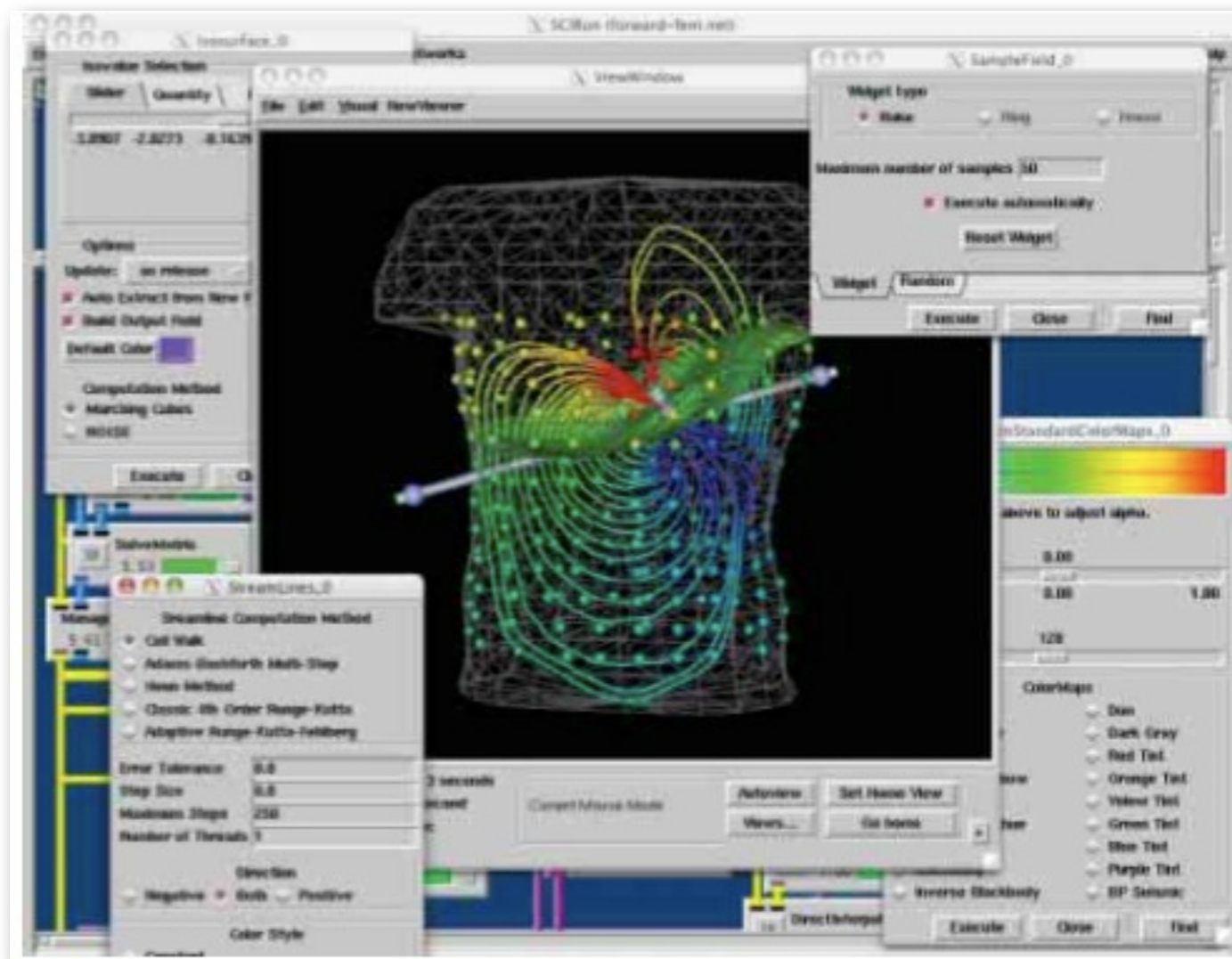


BioFEM: a SCIRun PowerApp

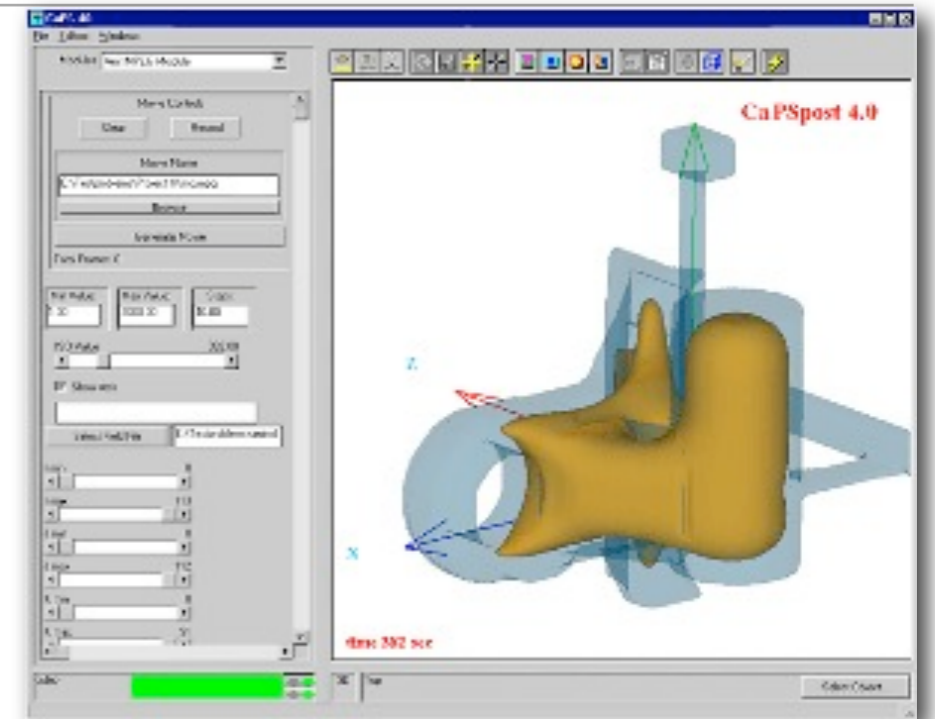


[www.avs.com](http://www.avs.com)

# Helping to create visualizations: Custom Applications



BioFEM: a SCIRun PowerApp

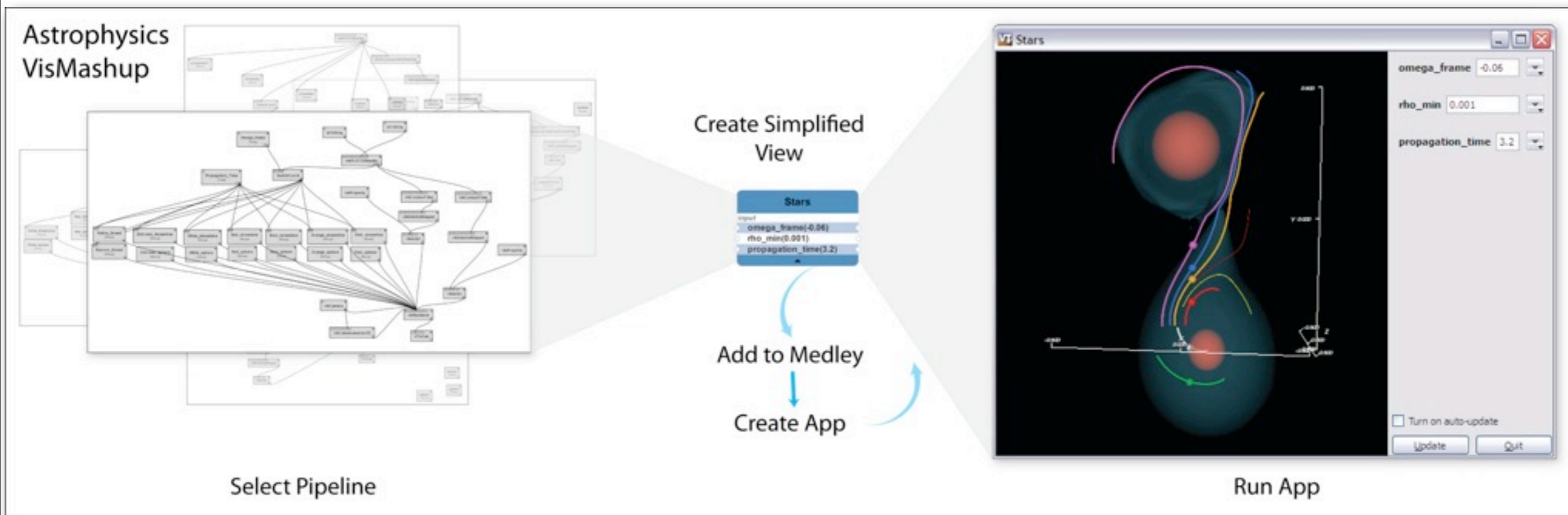


[www.avs.com](http://www.avs.com)

Applications are costly to develop,  
manually crafted for a given pipeline

# Our Approach: VisMashup

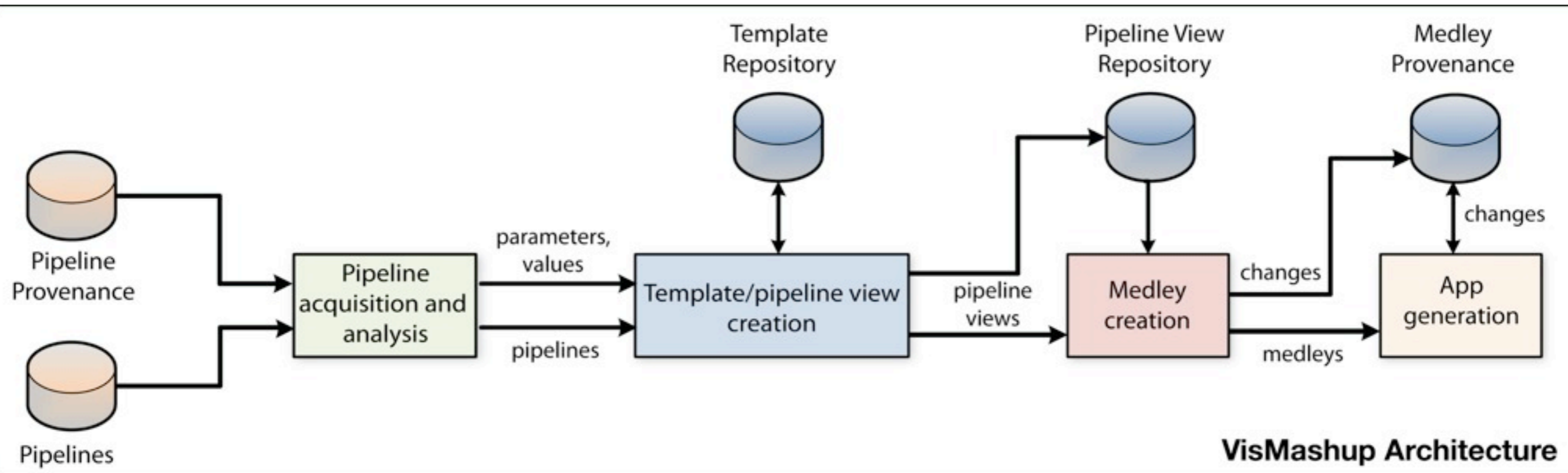
- Simplifies the creation, maintenance and use of customized visualization applications (mashups)
- Uses dataflows as the underlying model
- Keeps detailed provenance information of the application development process and use



# Model

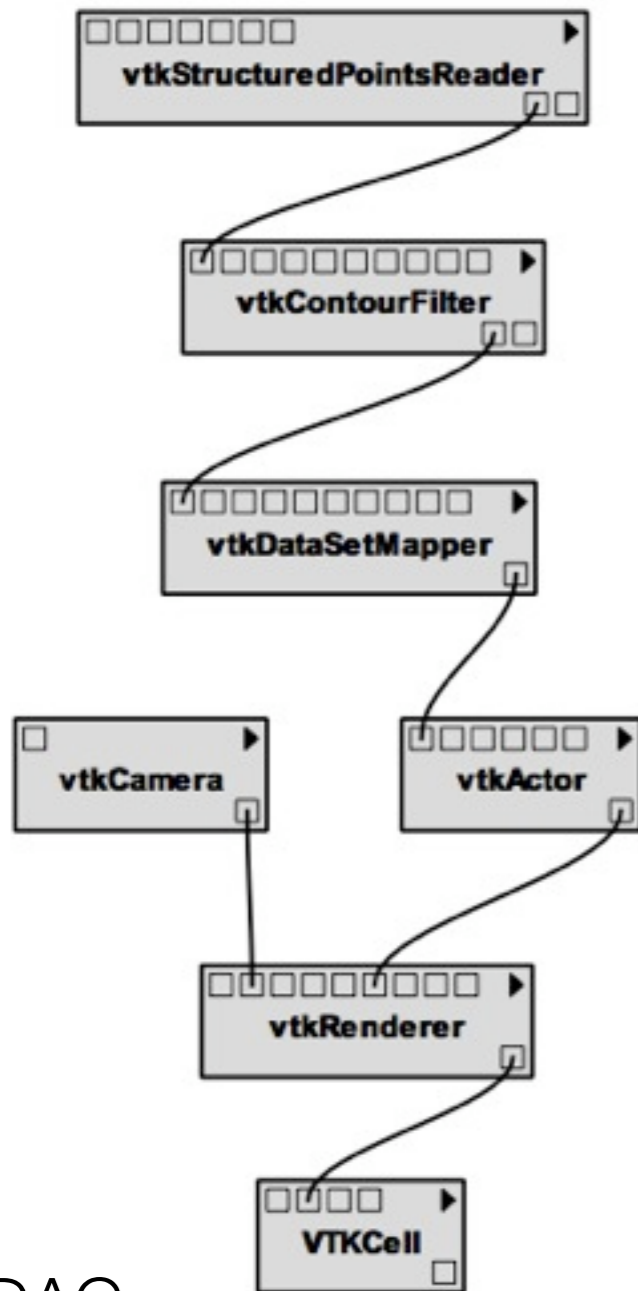
---

# The VisMashup Architecture

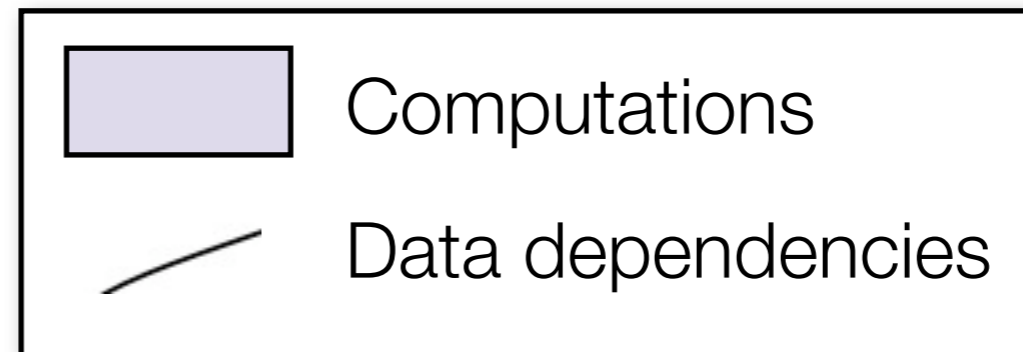


# Pipelines

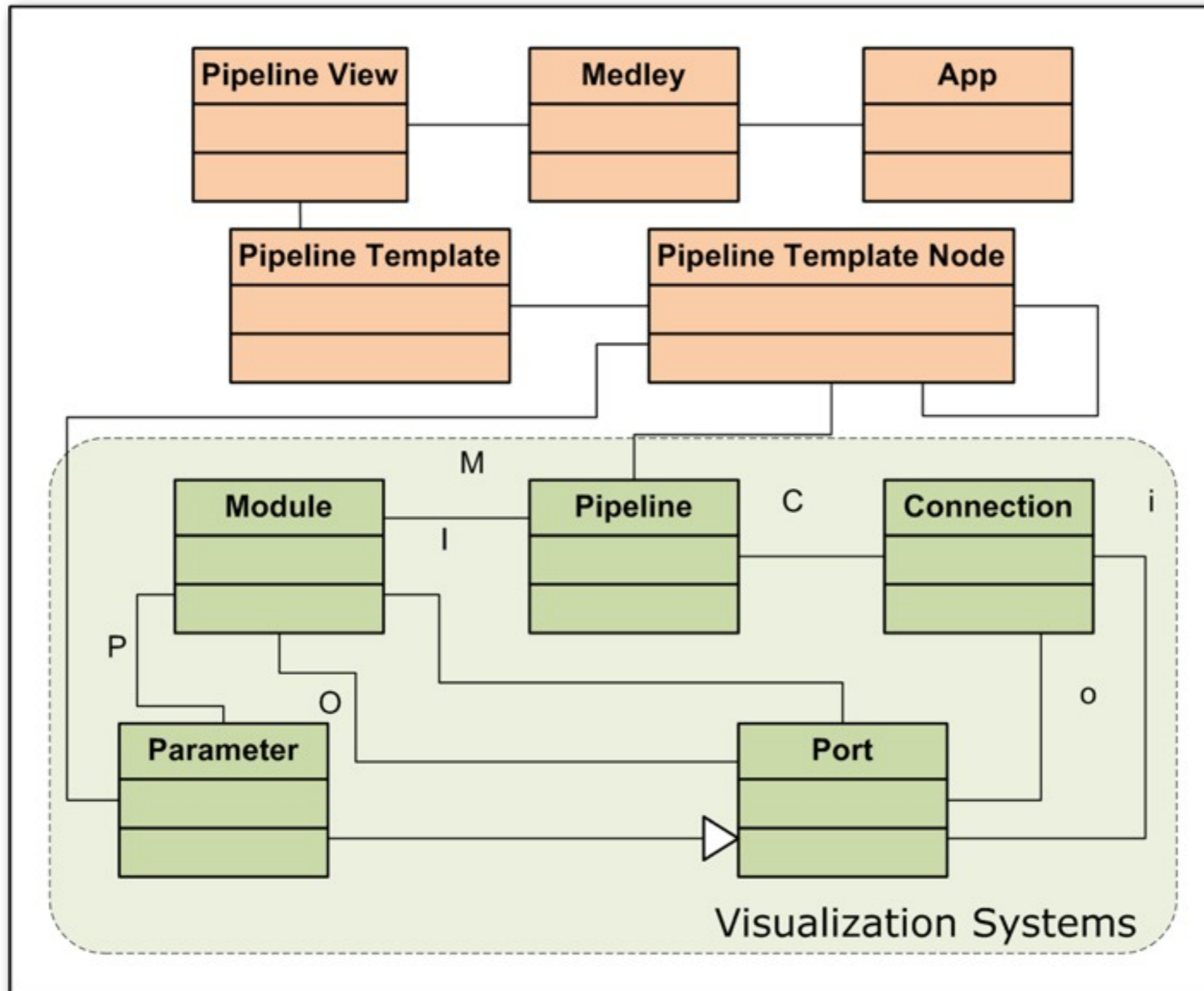
- Dataflow model is used to specify visualization pipelines



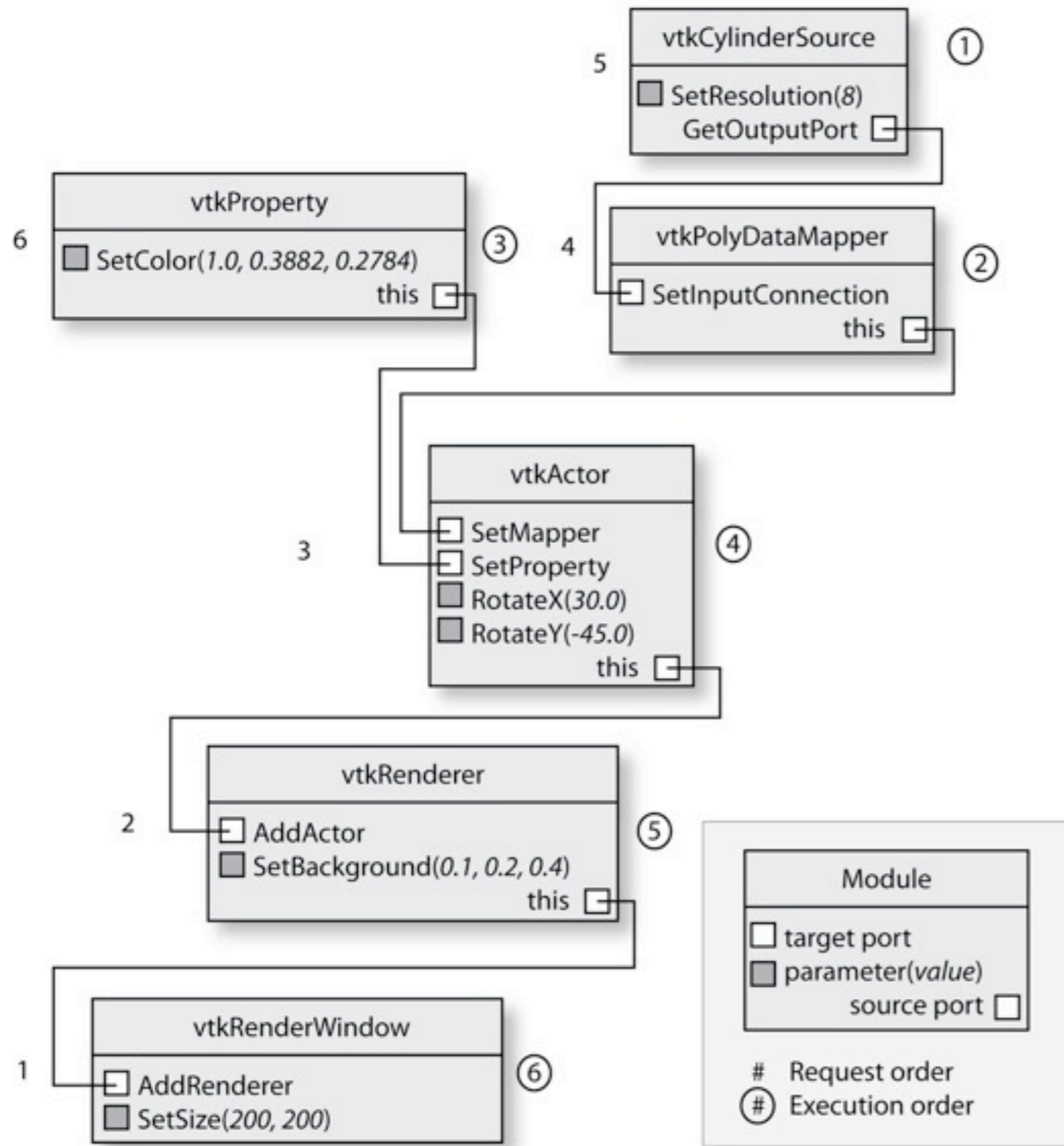
DAG



# Main concepts in VisMashup



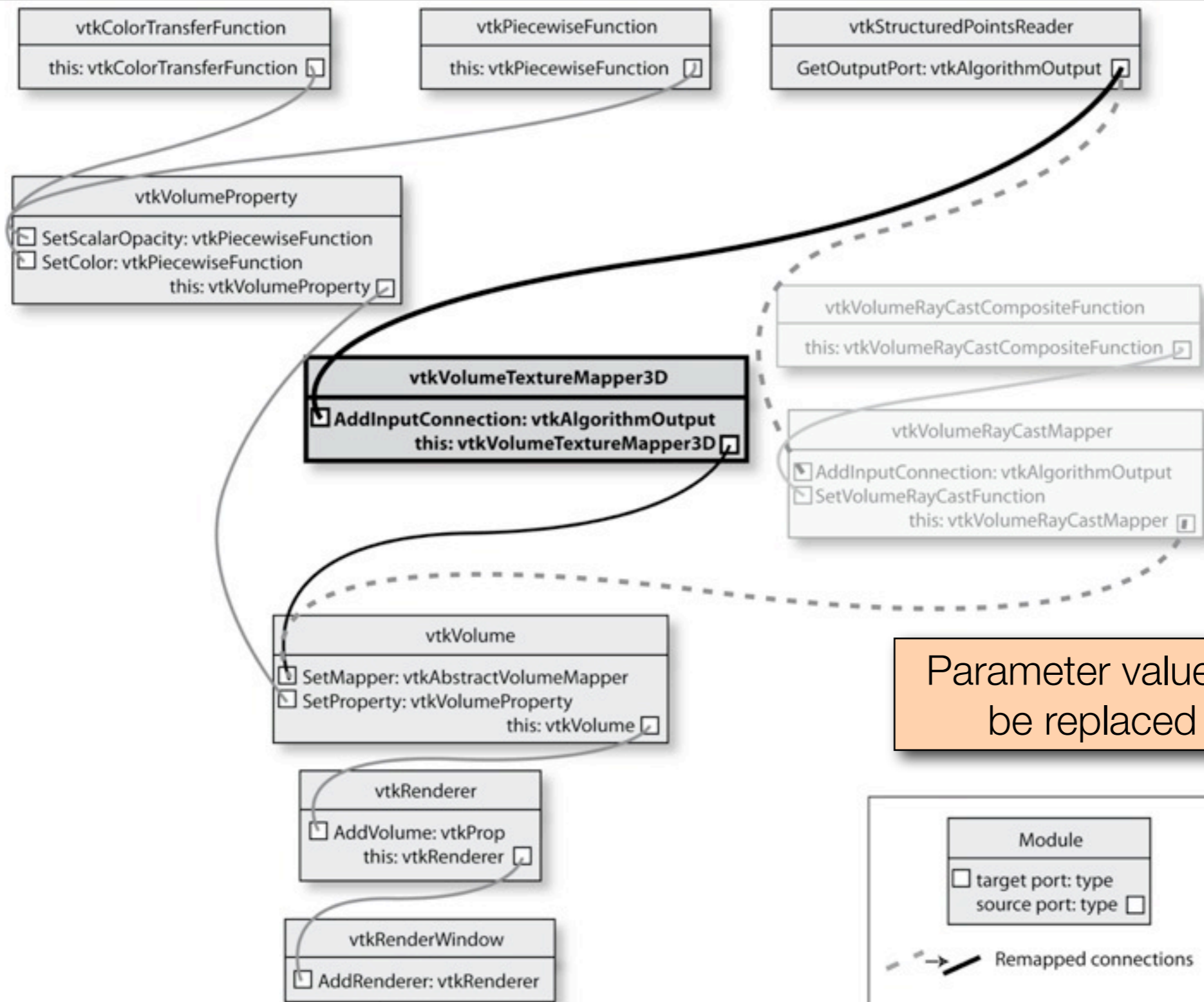
# Pipeline Operations: Run



The pipeline is executed in the order determined by the network of modules and connections, in a *demand-driven* fashion

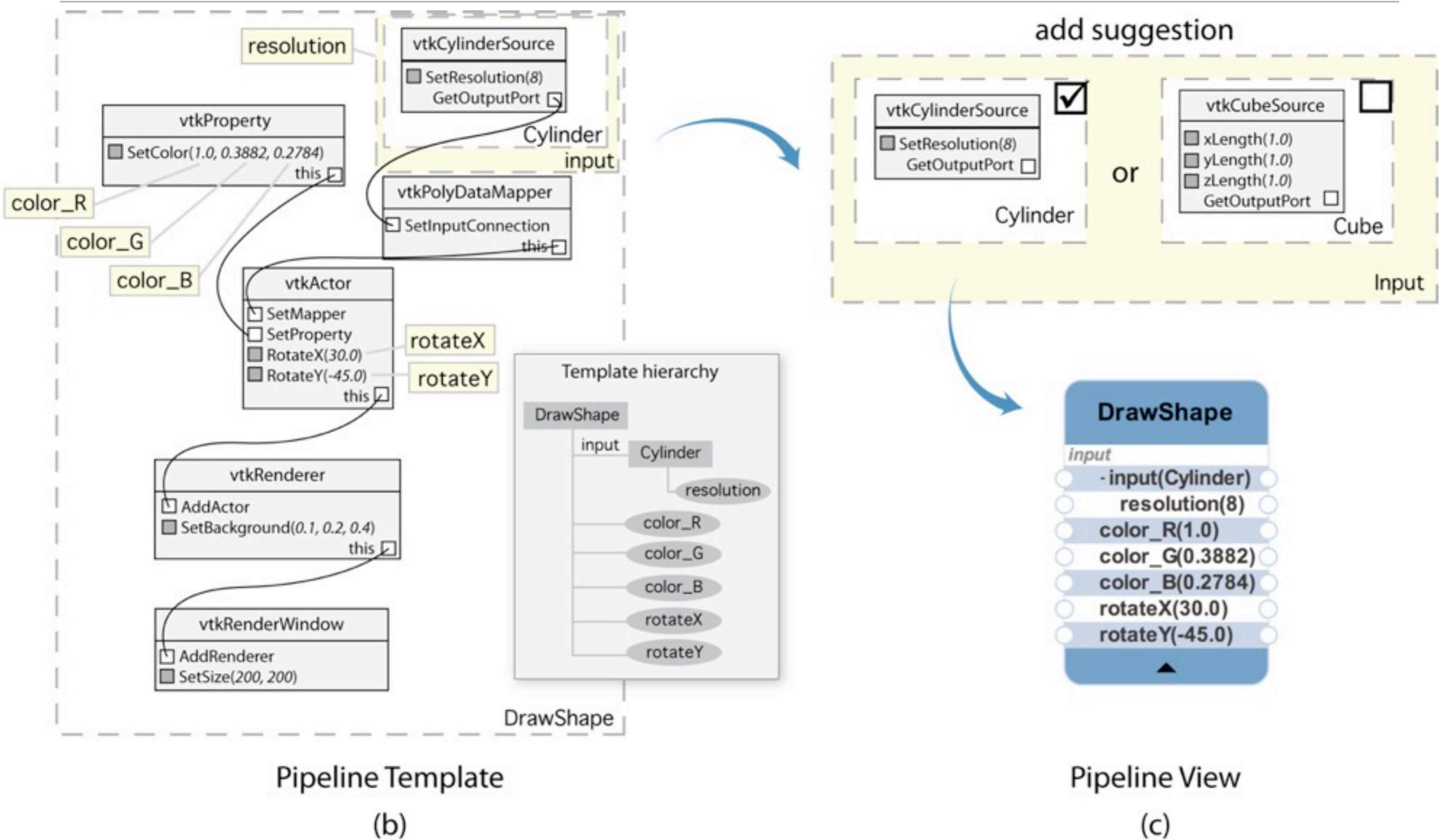


# Pipeline Operations: Substitution

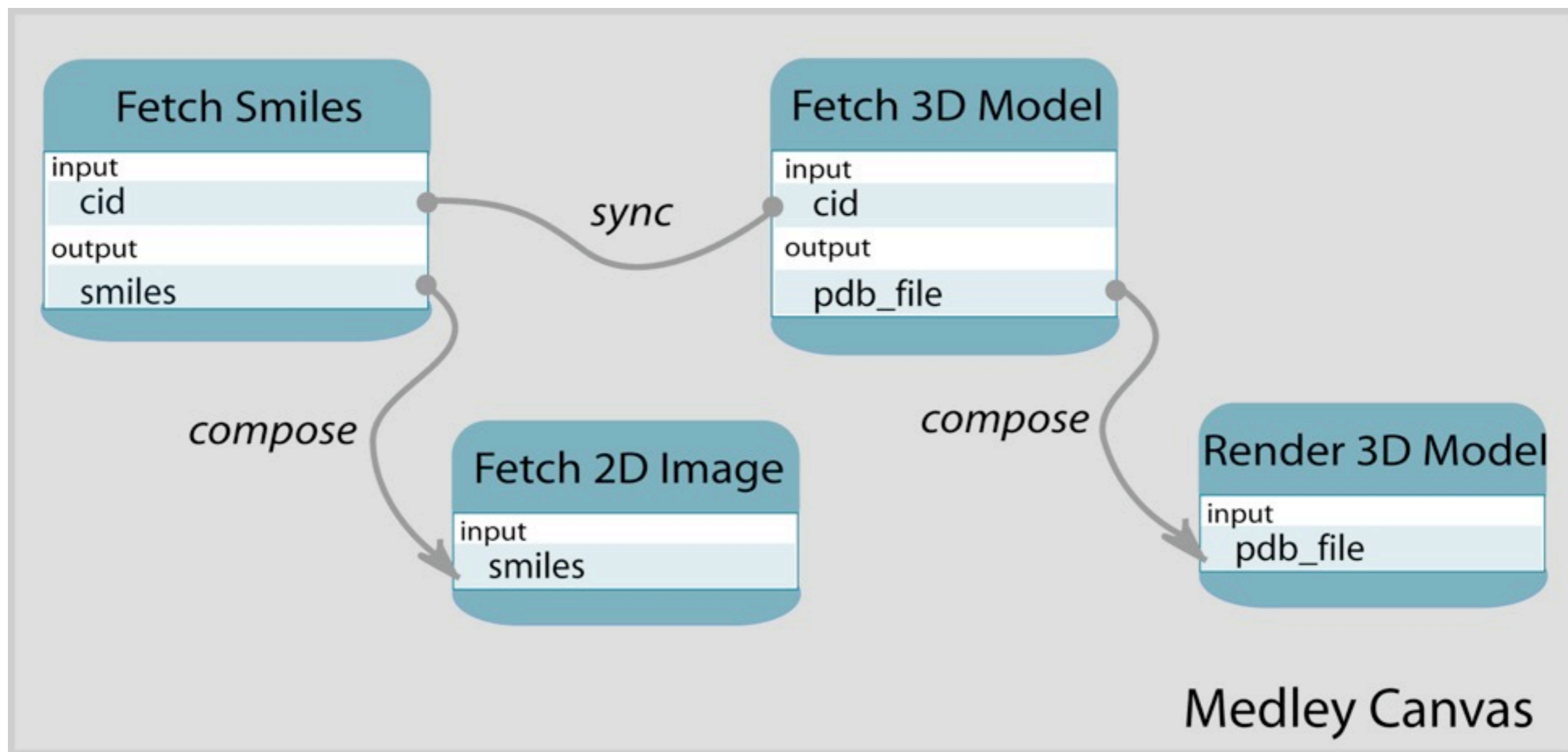


Parameter values and modules can be replaced within a pipeline

# Pipeline Templates and Pipeline Views



# Medleys: Collection of pipeline views

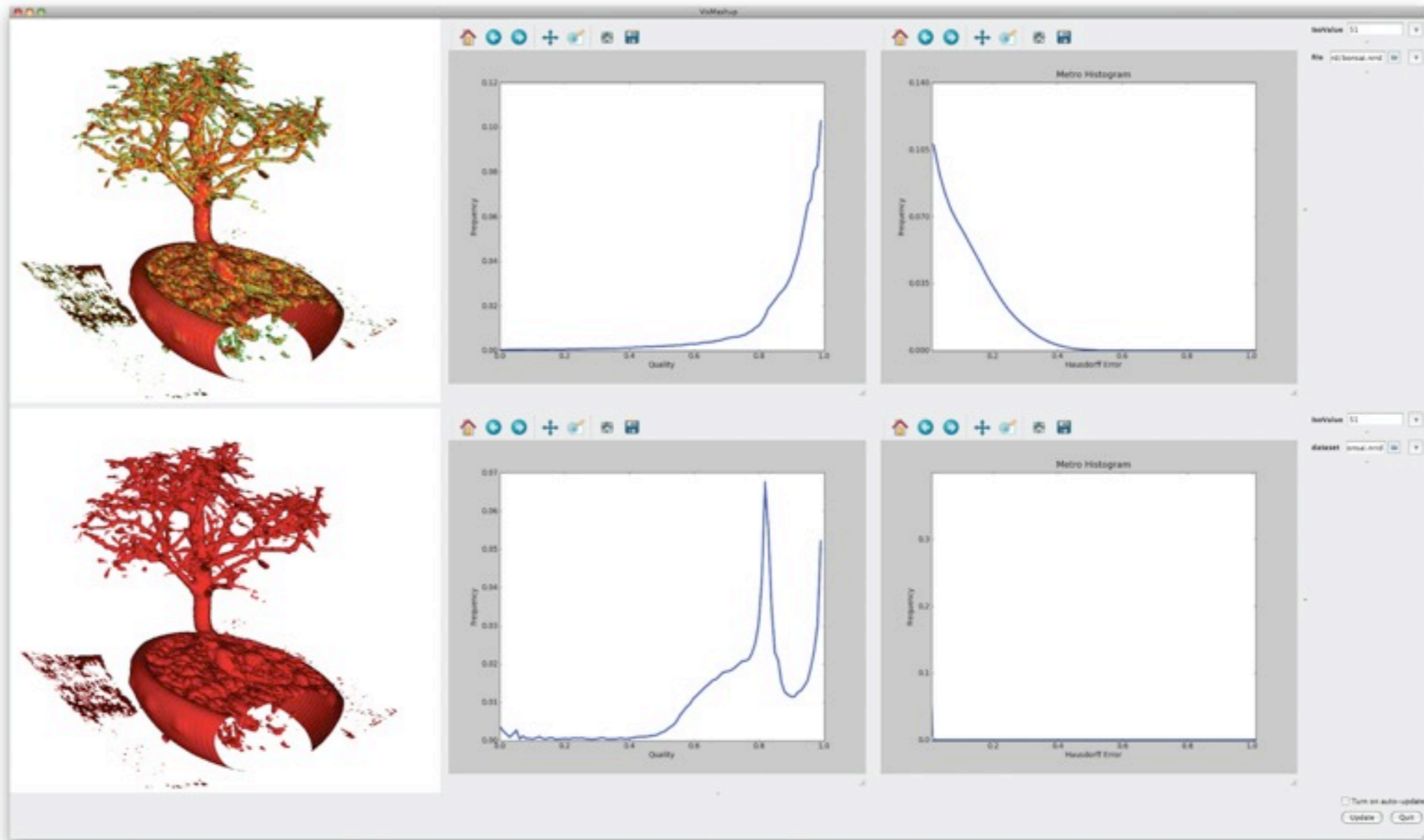
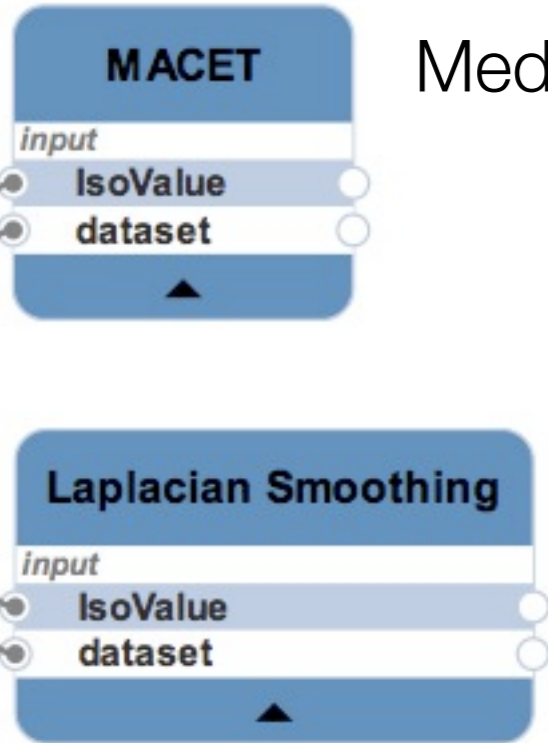


# VisMashup

Medley

VisMashup

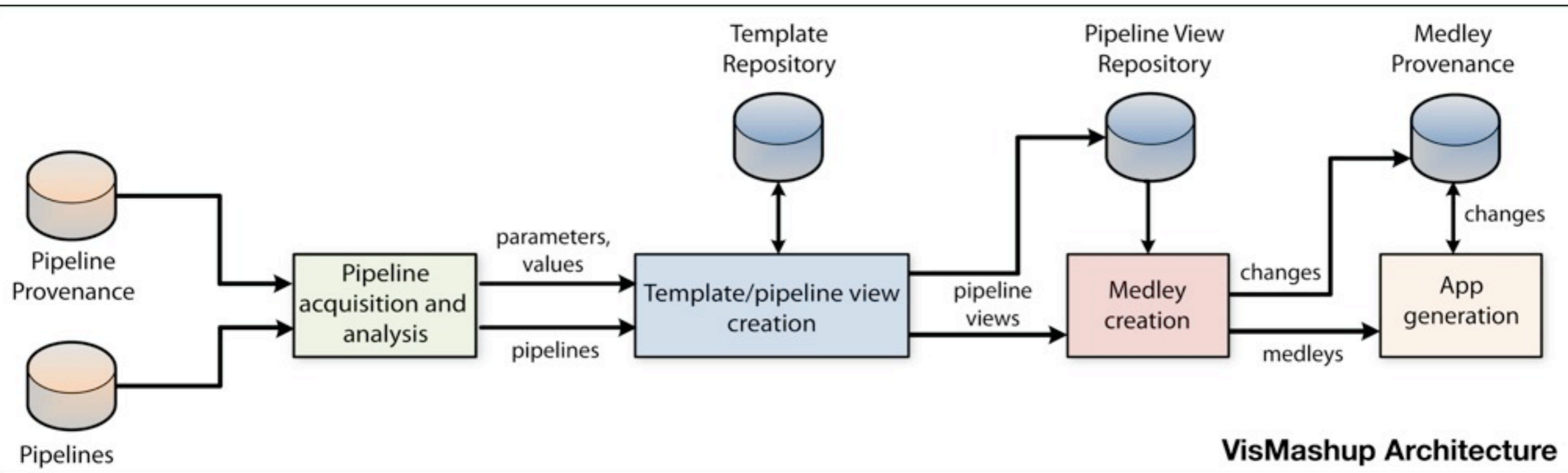
a flexible GUI automatically generated from a medley specification



# The VisMashup System

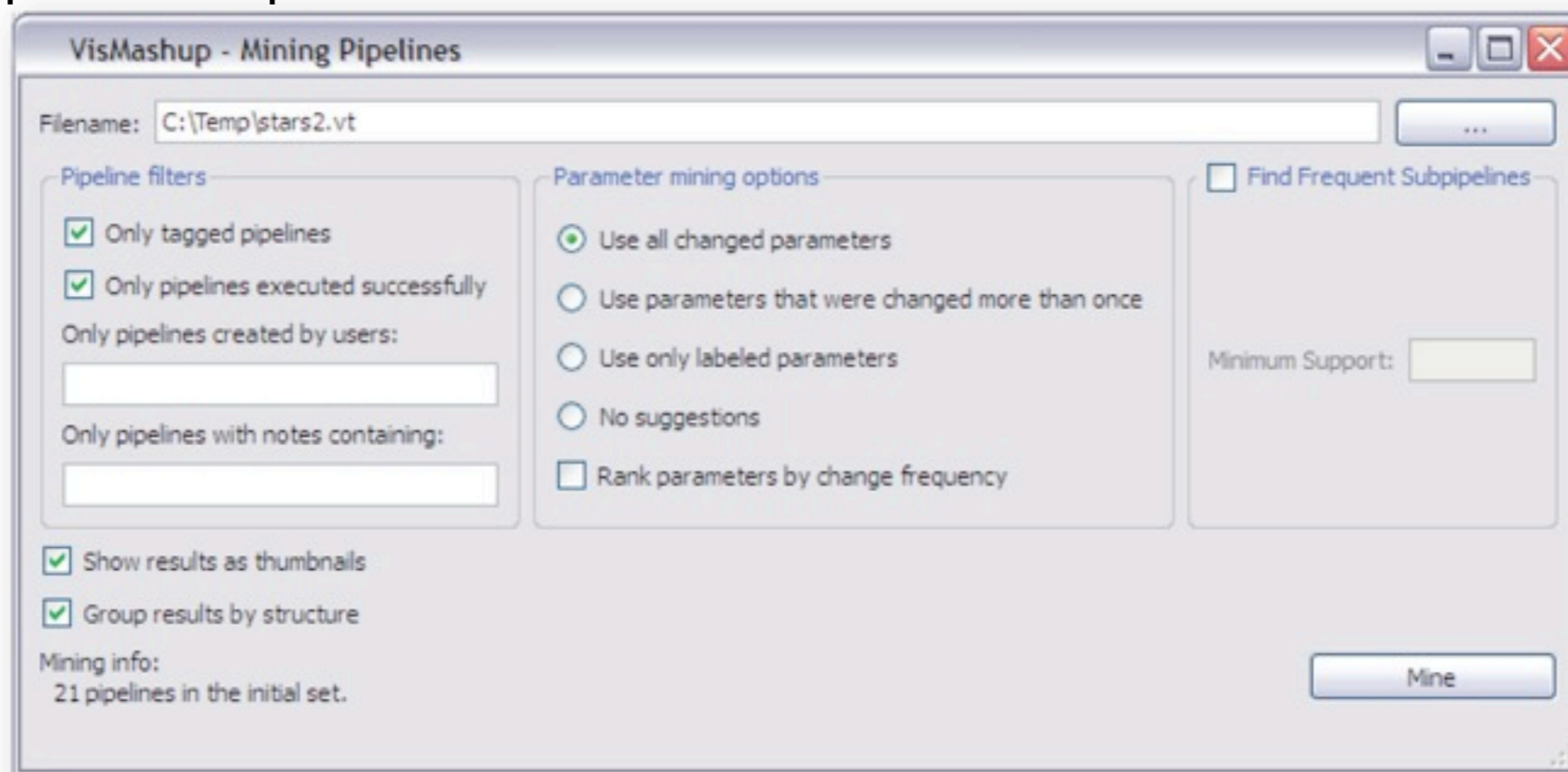
---

# The VisMashup Architecture



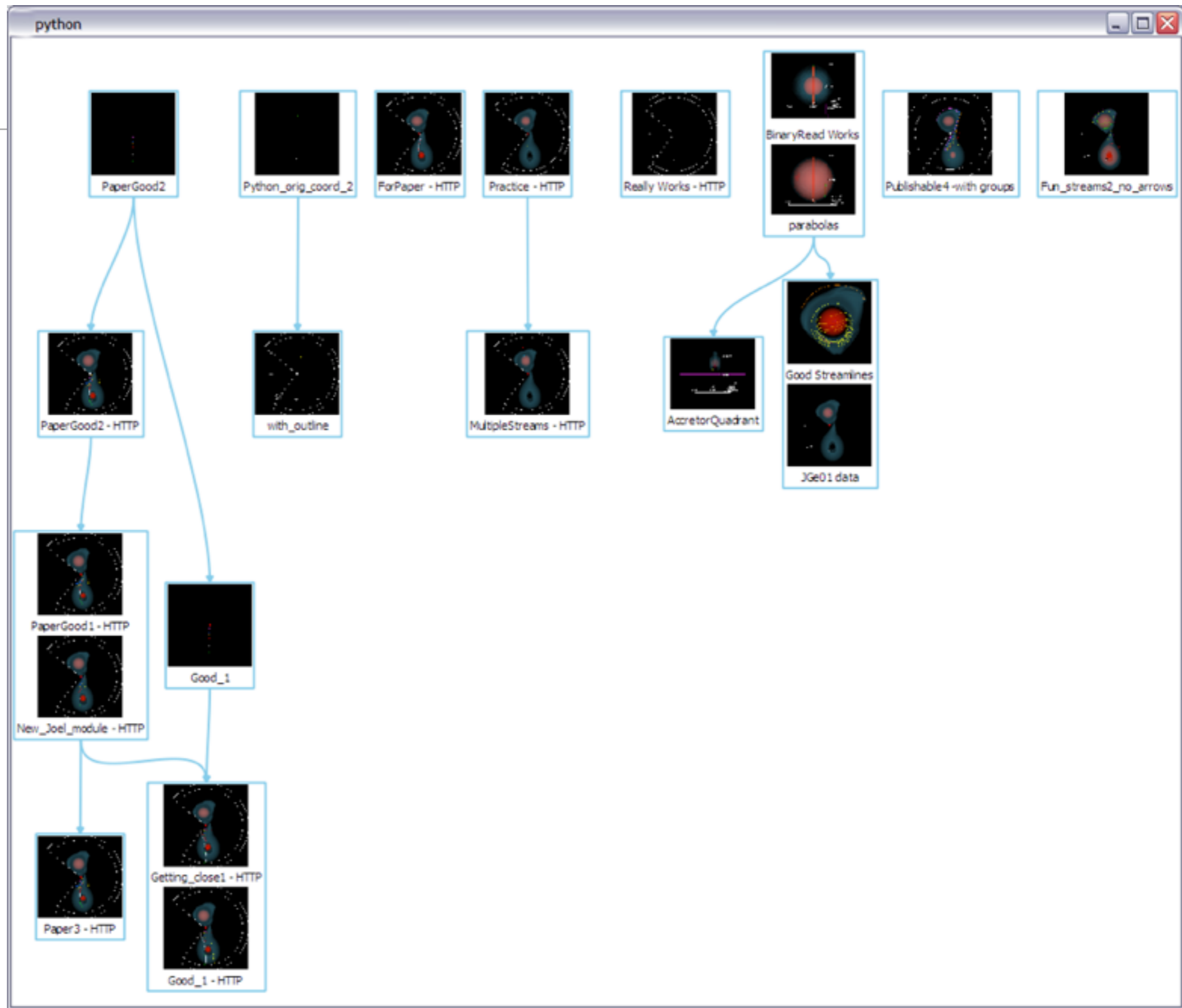
# Pipeline Acquisition and Analysis

- Pipelines
  - Can be built from scratch
  - Can be selected from a collection of pipelines
    - relevant pipelines
    - important parameters and associated values



# Hasse Diagram

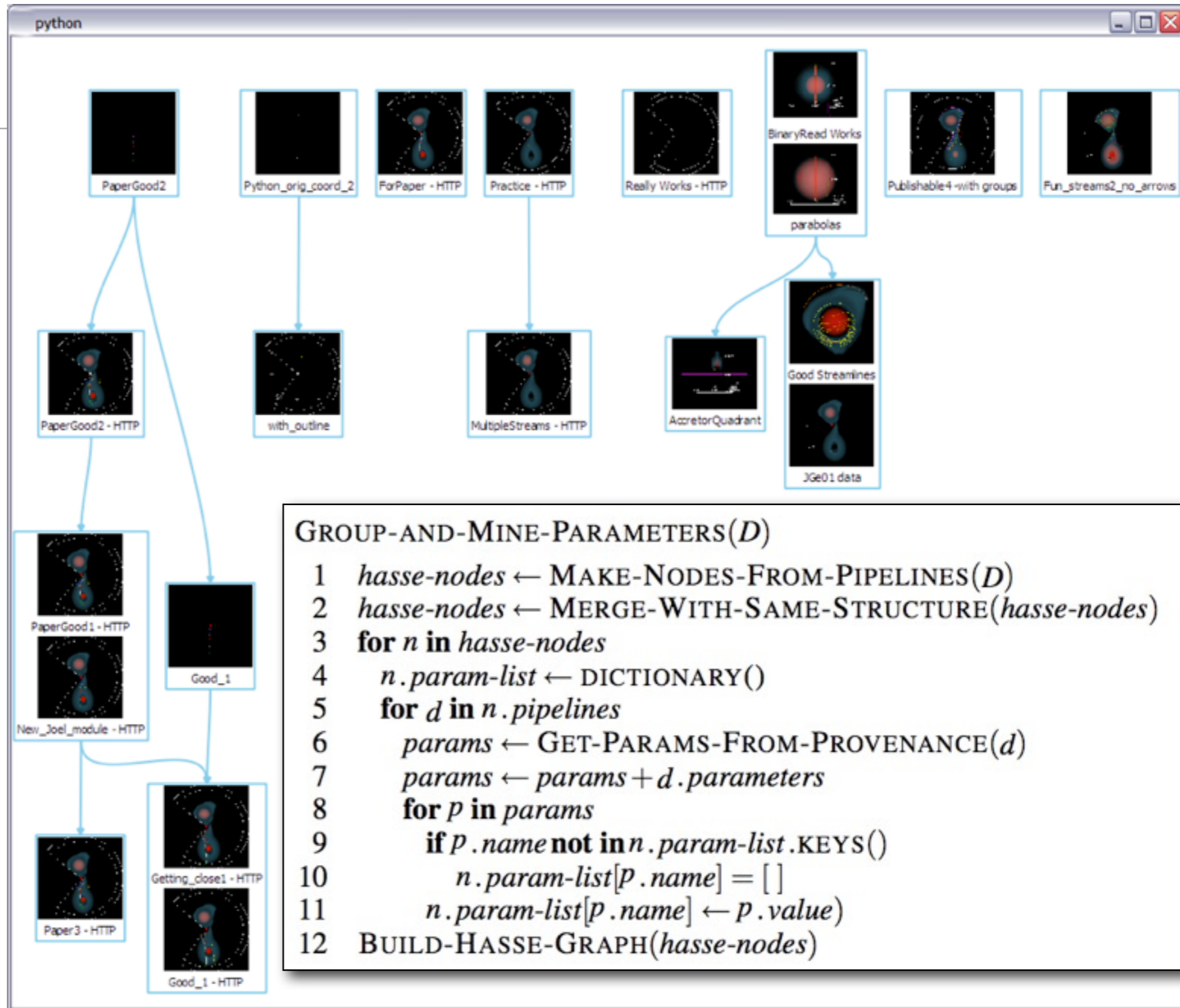
A Hasse diagram is a tree where each node corresponds to a group of isomorphic graphs, and edges between a parent and a child node indicate containment



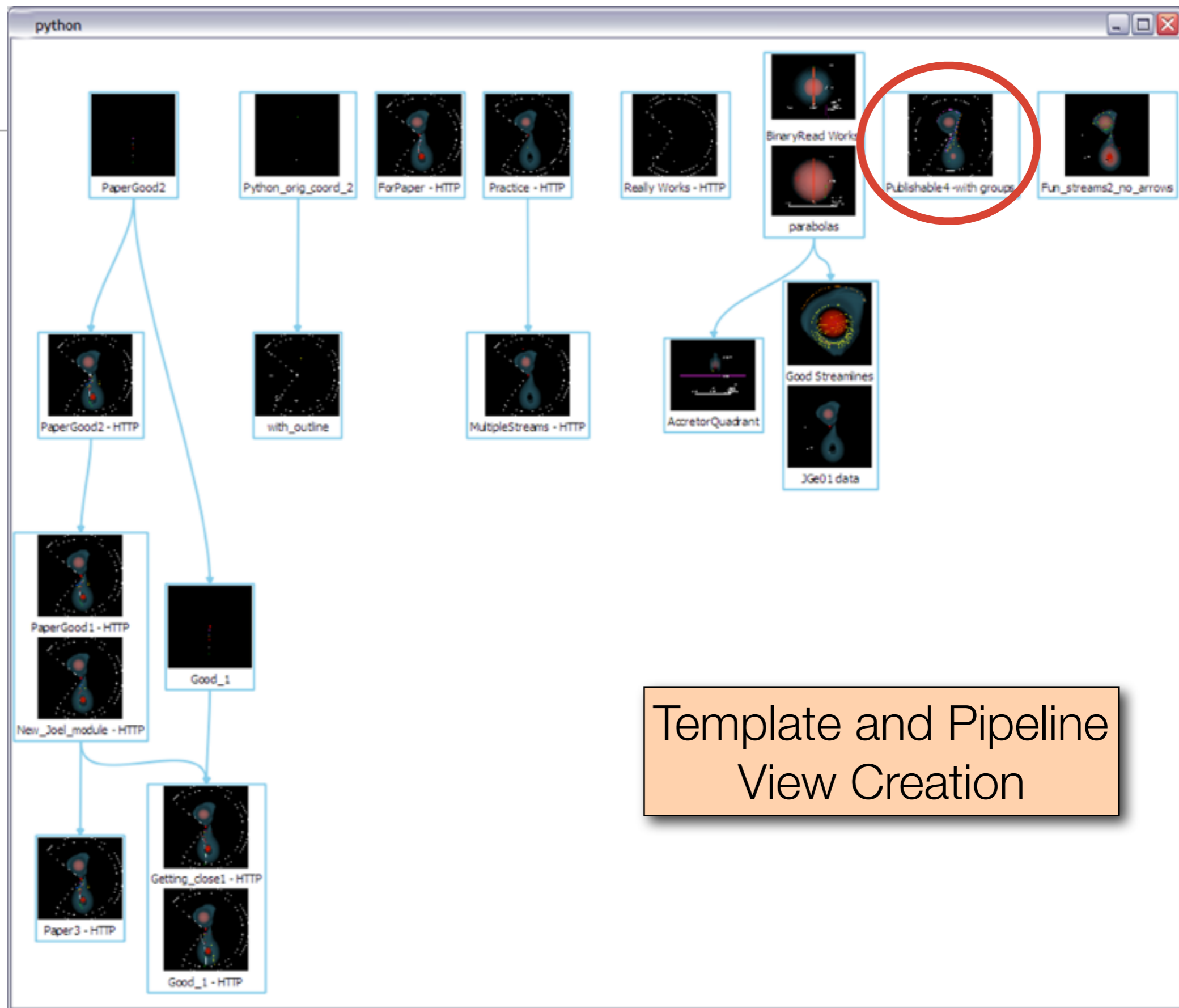


# Hasse Diagram

A Hasse diagram is a tree where each node corresponds to a group of isomorphic graphs, and edges between a parent and a child node indicate containment



# Hasse Diagram



### Pipeline View Preview

Parameters

	Name	Default value	Values list	Number of changes
(i) Float	omega_frame	-0.06	[-0.01, -0.012, -0.02, -0.03, ...]	178 time(s)
(i) Float	propagation_time	3.2	[0.1, 0.5, 1.0, 1.2, 1.5, 10, ...]	98 time(s)
(i) Float	vtkCamera6_Set...	0.0	[-1.0, 0.0, 100.0, 15.0, 25, ...]	8 time(s)
(i) Float	SwitchCoord77_...	0.001	[0.0001, 0.0005, 0.001, 1e, ...]	6 time(s)
(i) Float	vtkCamera6_Set...	4.0	[0.0, 3.0, 3.5, 4.0]	4 time(s)
(i) Float	vtkCamera6_Set...	0.0	[-1.0, 0.0]	2 time(s)
(i) String	HTTPFile762_url_...	n/JGe03XYZ.bin	[http://www.phys.lsu.edu/...	1 time(s)
(i) String	HTTPFile761_url_...	tion/JGe03q.bin	[http://www.phys.lsu.edu/...	1 time(s)

Create App

Template and Pipeline View Creation

### Pipeline View Preview

Parameters

	Name	Default value	Values list	Number of changes
(i) Float	omega_frame	-0.06	[-0.01, -0.012, -0.02, -0.03, ...]	178 time(s)
(i) Float	propagation_time	3.2	[0.1, 0.5, 1.0, 1.2, 1.5, 10, ...]	98 time(s)
(i) Float	vtkCamera6_Set...	0.0	[-1.0, 0.0, 100.0, 15.0, 25, ...]	8 time(s)
(i) Float	SwitchCoord77_...	0.001	[0.0001, 0.0005, 0.001, 1e, ...]	6 time(s)
(i) Float	vtkCamera6_Set...	4.0	[0.0, 3.0, 3.5, 4.0]	4 time(s)
(i) Float	vtkCamera6_Set...	0.0	[-1.0, 0.0]	2 time(s)
(i) String	HTTPFile762_url_...	n/JGe03XYZ.bin	["http://www.phys.lsu.edu/..."]	1 time(s)
(i) String	HTTPFile761_url_...	tion/JGe03q.bin	["http://www.phys.lsu.edu/..."]	1 time(s)

**Stars**

input

- omega\_frame(-0.06)
- rho\_min(0.001)
- propagation\_time(3.2)

▲

Create App

BinaryRead Works

parabolas

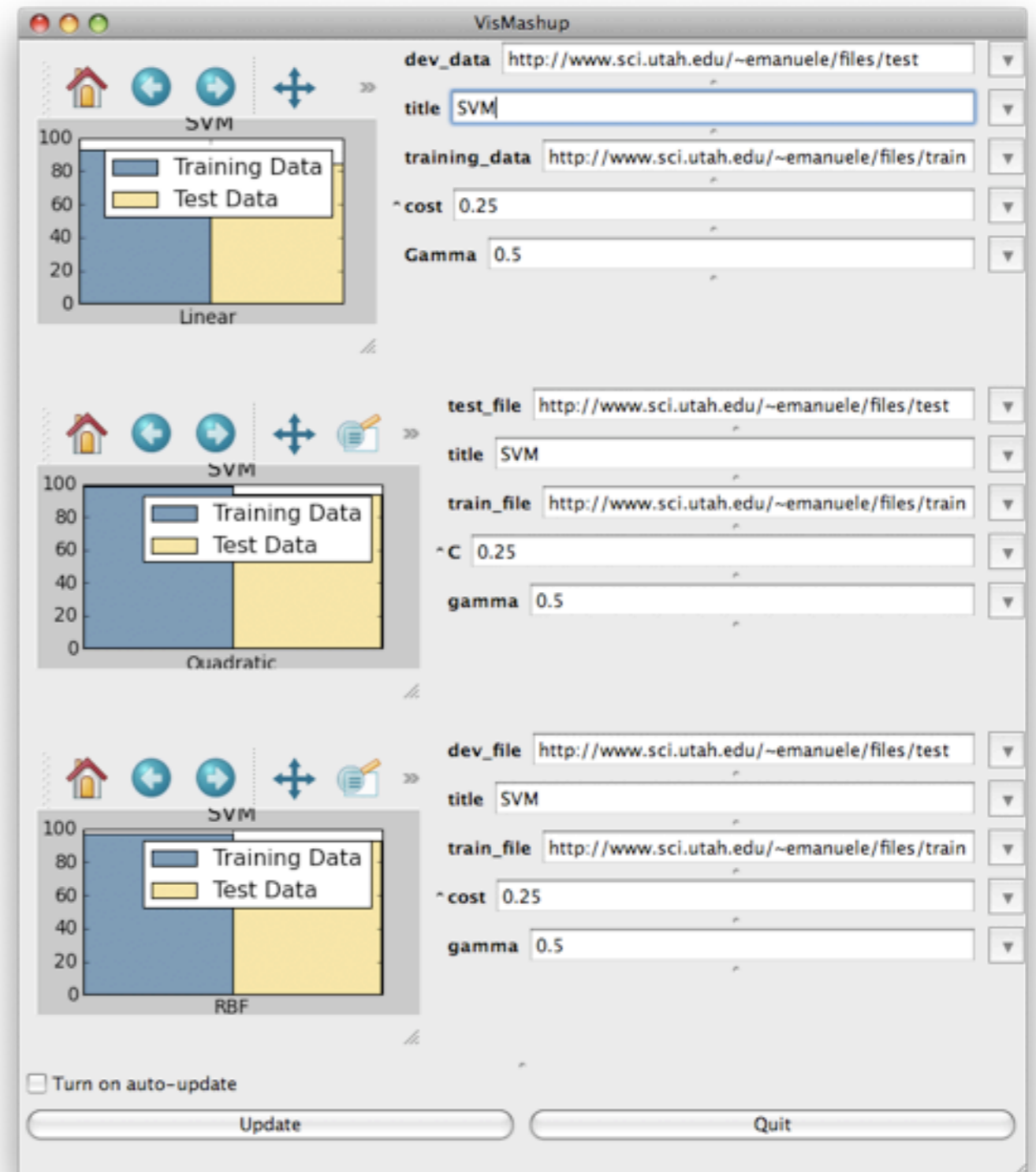
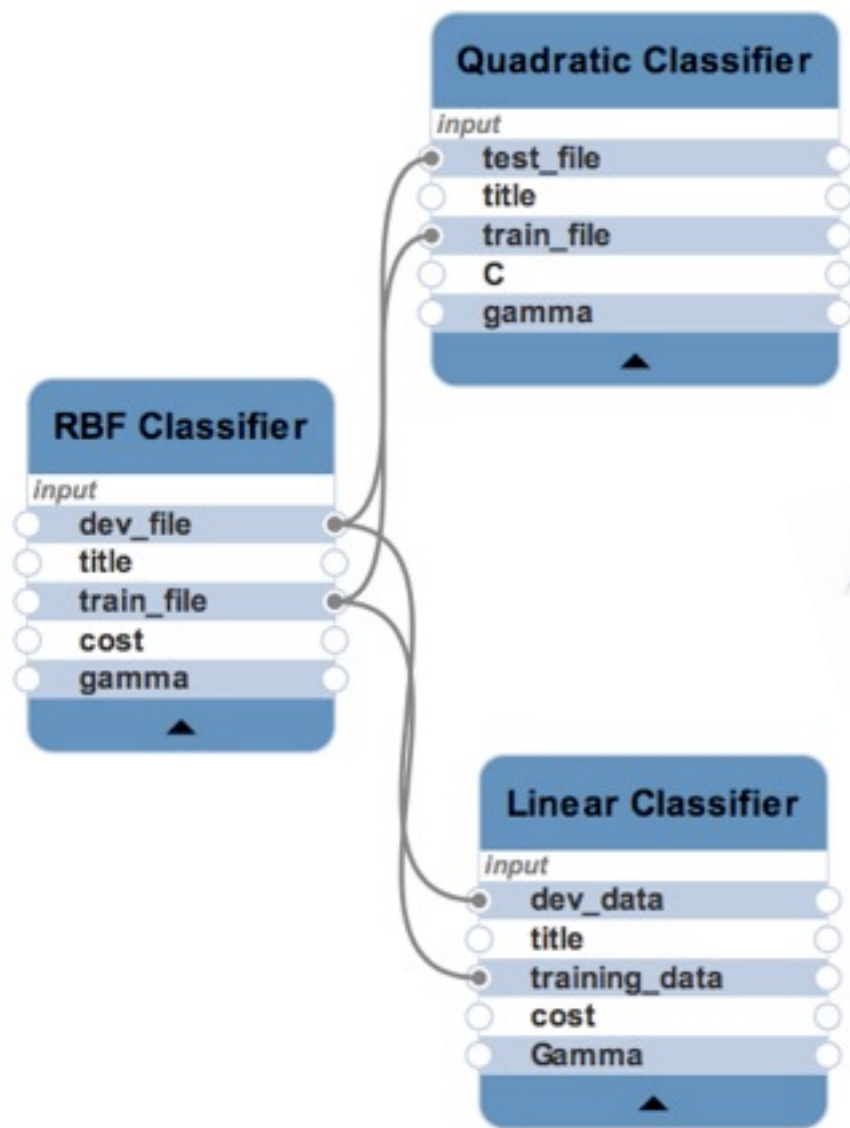
Good Streamlines

AccretorQuadrant

JGe01 data

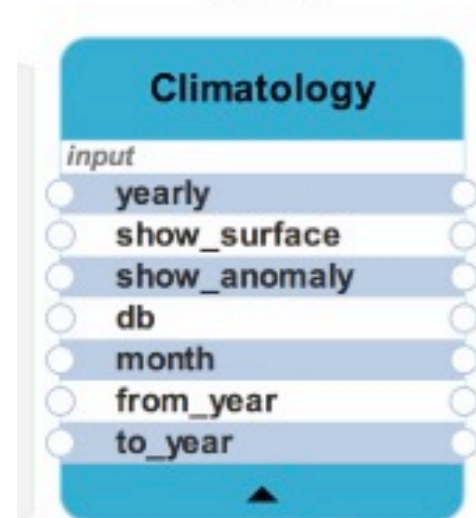
Template and Pipeline View Creation

# Medley Creation and Mashup Generation



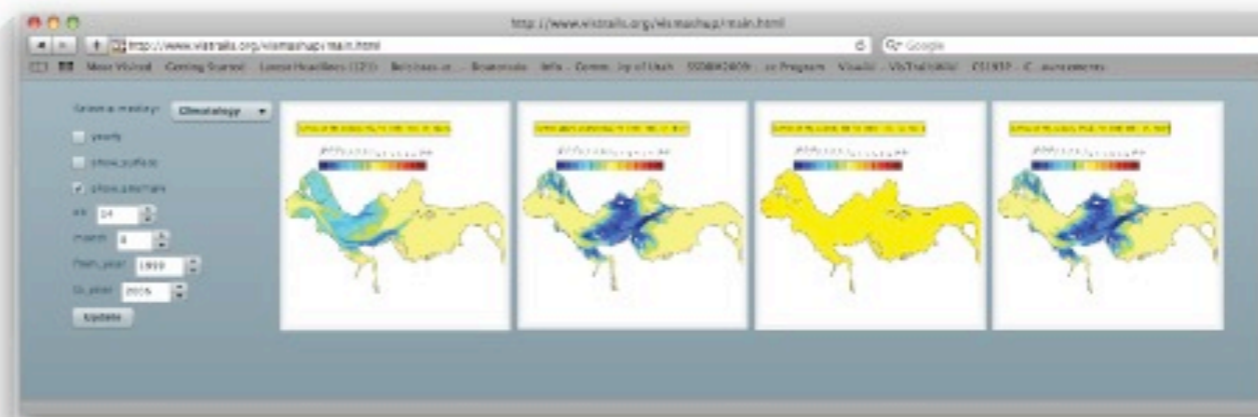
# Medley Creation and Mashup Generation

Create Simplified View

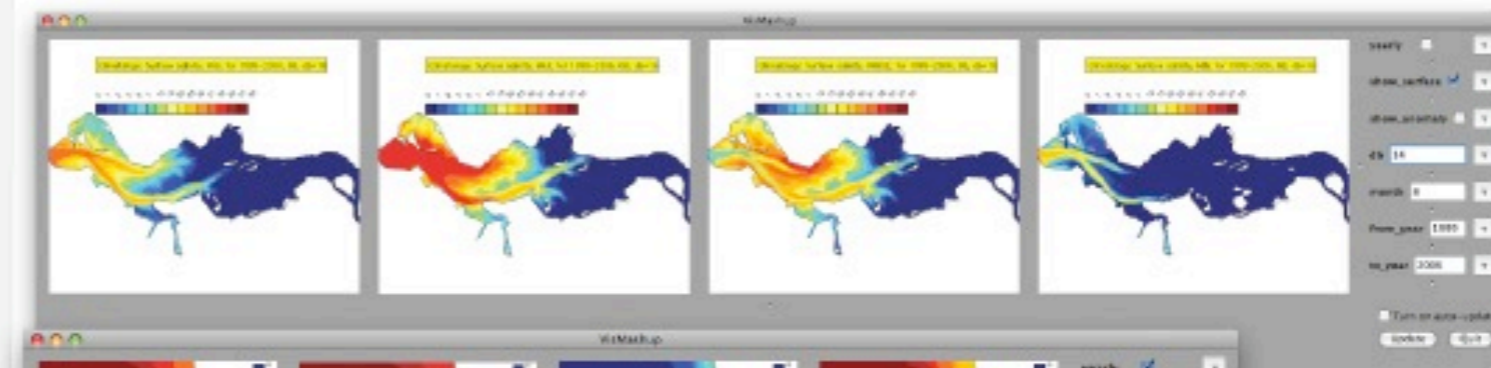


Add to Medley

Create App



Web



Desktop

Run App

# Case Study: Sharing Astrophysics Analyses

---

# Professor Joel Tohline's group Louisiana State University (LSU)

---

- Computational fluid dynamics (CFD) techniques are used to model various astrophysical phenomena
  - simulation of mass-transfer instabilities in binary star systems
- Visualization tools and techniques to help them explore the results of simulations (VisTrails: VTK + custom analysis modules)

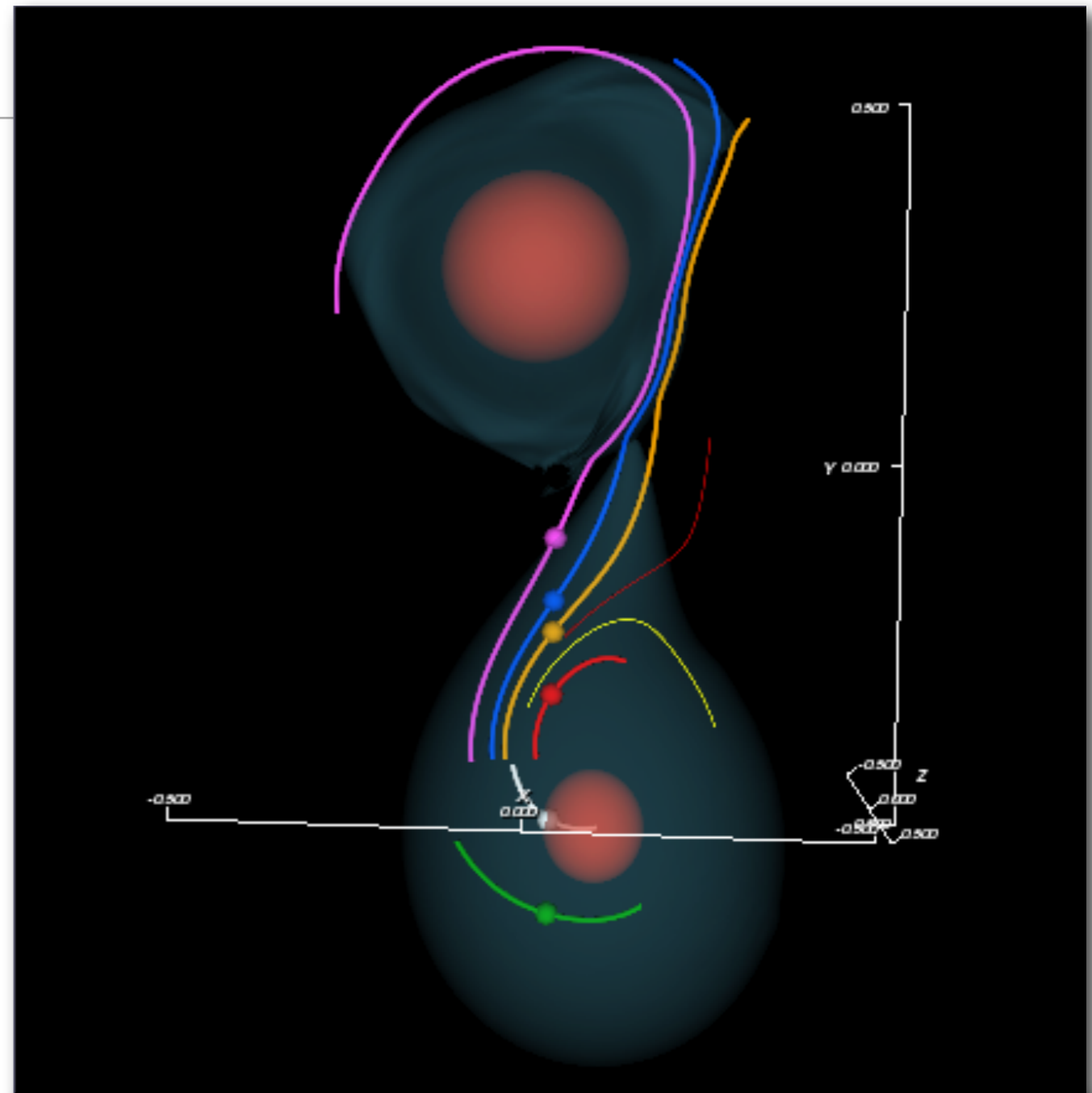


<http://www.phys.lsu.edu/~tohline/vistrails/>



# Binary Star System

- Two stars orbiting around a common center of mass with an orbital period  $P$
- When the system is viewed from a frame rotating with an angular frequency  $\Omega_{frame} = 2\pi/P$ , the system will appear stationary
- During a simulation,  $P$  and  $\Omega_{frame}$  are expected to vary



What is the best measure of the true orbital period of the binary system?

# Using VisTrails

LSU using VisTrails

http://www.phys.lsu.edu/~tohline/vistrails/ Joel Tohline

SRM - VIS09 LSU using VisTrails

## Learning How to use [VisTrails](#)

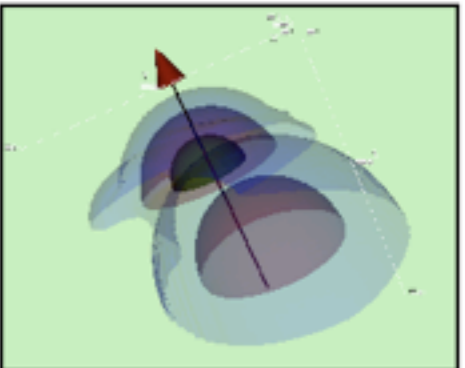
- **Part I:**

In July, 2007, Shangli Ou packaged all the material that is needed to run his 2D SCF code. Our idea is that this code could be effectively linked into VisTrails to provide a simple GUI for all potential users. The "Documentation" explains how to use the SCF code and it sketches the idea for developing a useful GUI.

  1. **SCF code:** 2007, July
    - [scf2d.vistrails.tar.gz](#)
    - [Documentation](#)
- **Part II:**

In August, 2008, Tohline and Z. Byerly began a more intense collaboration with Claudio Silva's research group at the University of Utah. Our objective is to use the capabilities of [VisTrails](#) to visualize and routinely analyze the results of astrophysics CFD simulations.

  1. **Example #1:** 2008, July 28
    - [jetOBJrenderer.vt](#)
    - [den1.obj](#) [0.64 MByte ASCII]
    - [den2.obj](#) [2.9 MByte ASCII]
    - [den3.obj](#) [5.3 MByte ASCII]
  2. **Example #2:** 2008, August 6 -- Files relevant to reading raw data files into VisTrails.
    - The following binary data files each contain one 3D array [178 × 256 × 146] of type real\*4
    - **big\_endian** binary files written from a Fortran program
      - [density](#)
      - [radial-momentum](#)
      - [angular-momentum](#)
      - [vertical-momentum](#)
    - **little\_endian** binary files written from a Fortran program
      - [density](#)
      - [radial-momentum](#)



# Using VisTrails

LSU using VisTrails

http://www.phys.lsu.edu/~tohline/vistrails/ Joel Tohline

SRM - VIS09 LSU using VisTrails

## Learning How to use VisTrails

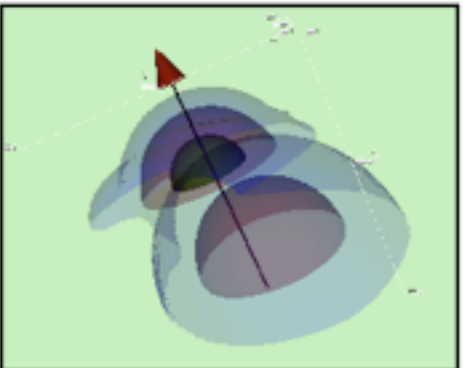
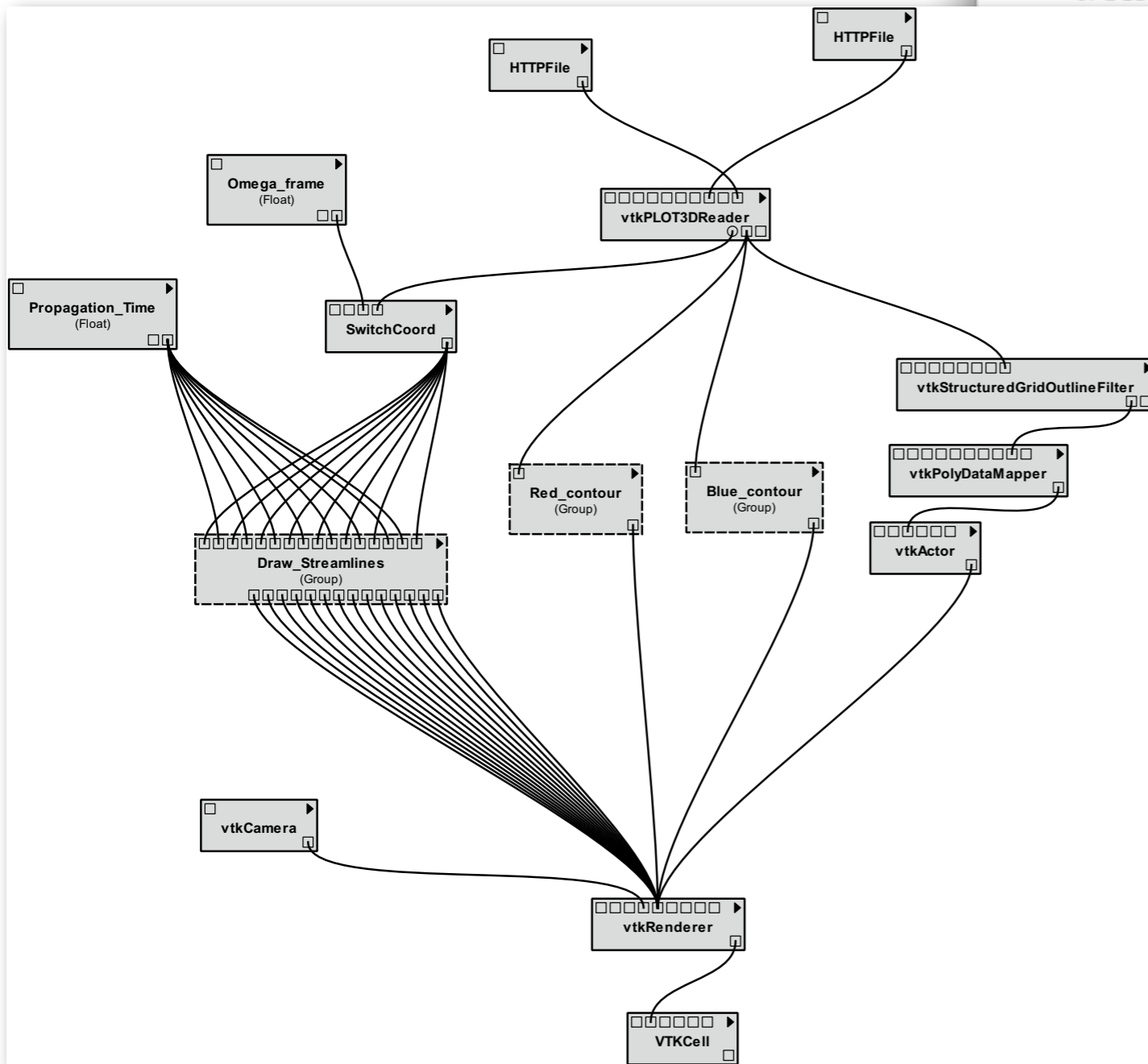
- Part I:**

In July, 2007, Shangli Ou packaged all the material that is needed to run his 2D SCF code. Our idea is that this code could be effectively linked into VisTrails to provide a simple GUI for all potential users. The "Documentation" explains how to use the SCF code and it sketches the idea for developing a useful GUI.

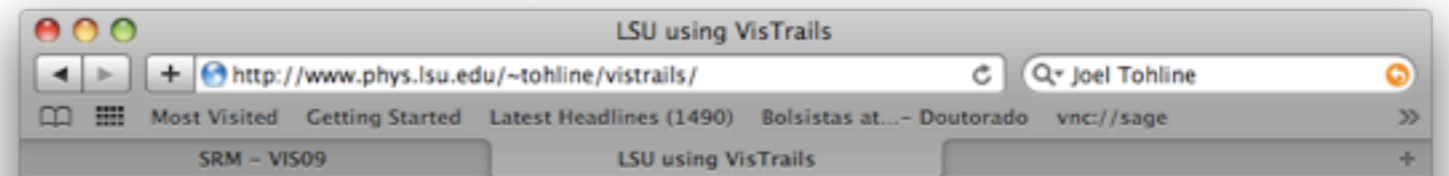
  - SCF code: 2007, July**
    - [scf2d.vistrails.tar.gz](#)
    - [Documentation](#)
  - 2008, Tohline and Z. Byerly began a more intense collaboration with Claudio Silva's research group at the University of Utah. Our objective is to use the capabilities of VisTrails to visualize and routinely analyze astrophysics CFD simulations.**
    - Example #1: 2008, July 28**
      - [jetOBJrenderer.vt](#)
      - [den1.obj](#) [0.64 MByte ASCII]
      - [den2.obj](#) [2.9 MByte ASCII]
      - [den3.obj](#) [5.3 MByte ASCII]
    - Example #2: 2008, August 6 -- Files relevant to reading raw files into VisTrails.**

The following binary data files each contain one 3D array [178 x 256 x 146] of type real\*4

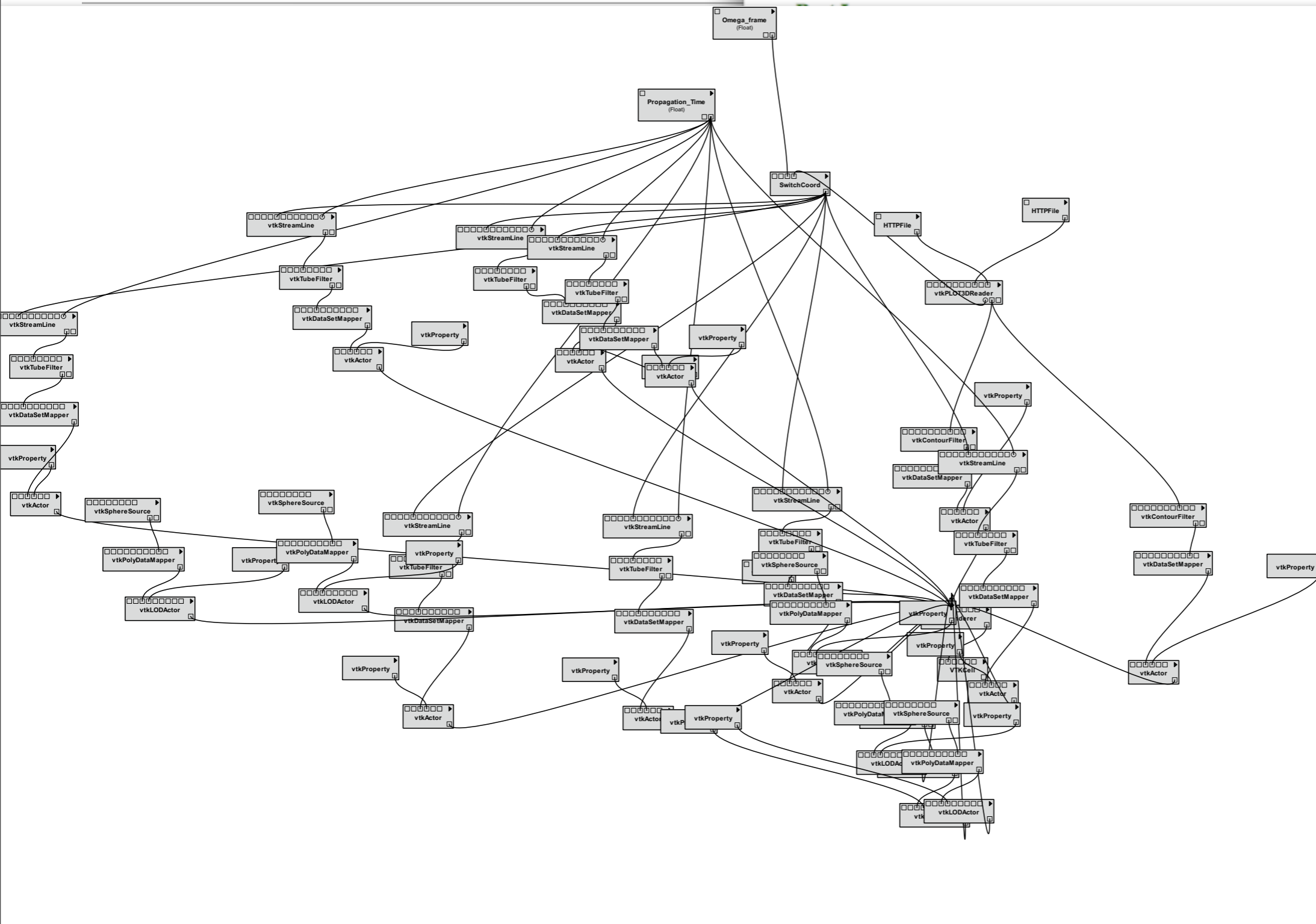
      - big\_endian** binary files written from a Fortran program
        - [density](#)
        - [radial-momentum](#)
        - [angular-momentum](#)
        - [vertical-momentum](#)
      - little\_endian** binary files written from a Fortran program
        - [density](#)
        - [radial-momentum](#)

# Using VisTrails

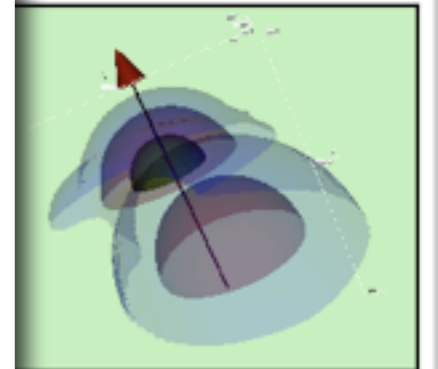


## Learning How to use VisTrails

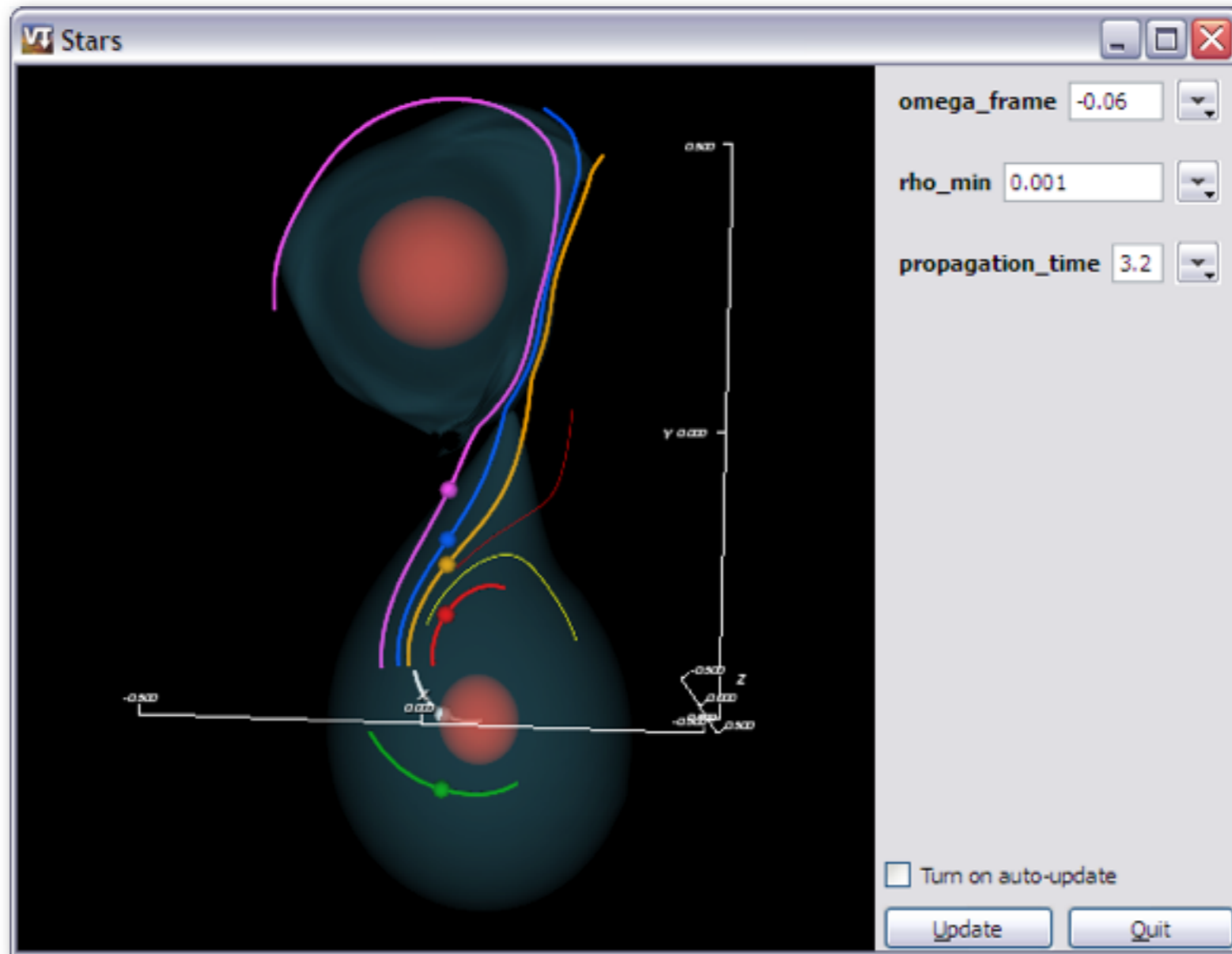


ED SCF code. Our idea is that this potential users. The for developing a useful GUI.

with Claudio Silva's research group to visualize and routinely analyze

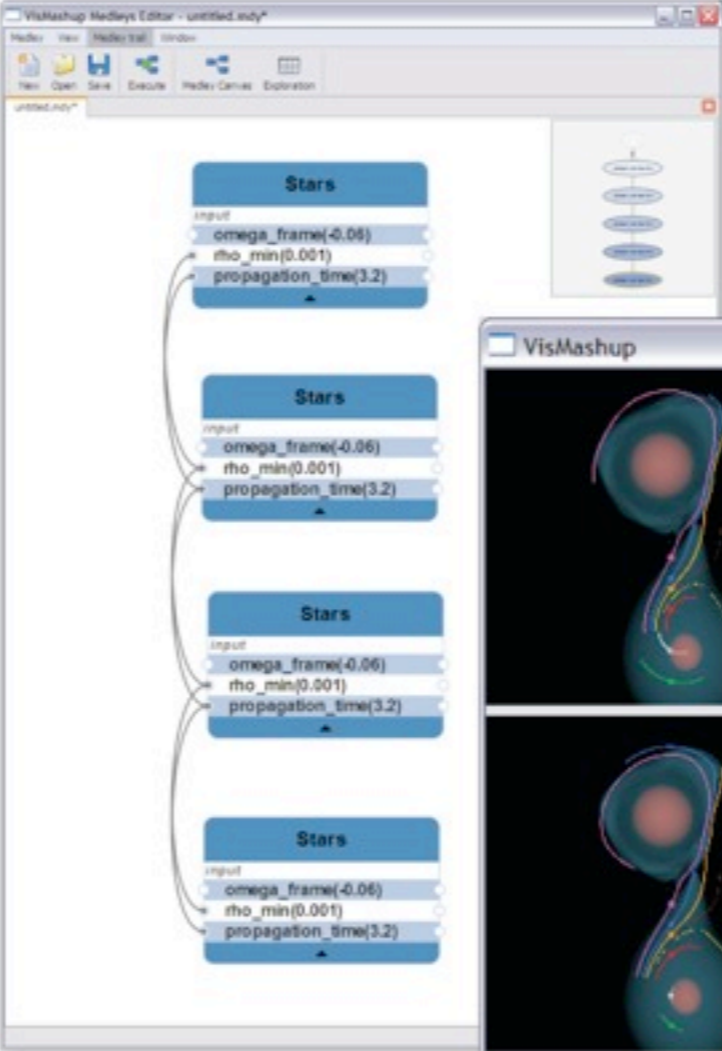


# Using VisMashups




# Using VisMashups

(a)



(b)



The image displays two windows from the VisMashup software. Window (a) is the 'Medleys Editor' showing a workflow with four 'Stars' blocks. Each block has three input parameters:  $\omega_{frame}$  (set to -0.06),  $\rho_{min}$  (set to 0.001), and  $\text{propagation\_time}$  (set to 3.2). Window (b) is the 'VisMashup' viewer, which shows four sequential frames of a star simulation. The parameters for each frame are: Frame 1:  $\omega_{frame} = -0.06$ ,  $\rho_{min} = 0.001$ ,  $\text{propagation\_time} = 3.2$ ; Frame 2:  $\omega_{frame} = -0.041$ ,  $\rho_{min} = 0.001$ ,  $\text{propagation\_time} = 3.2$ ; Frame 3:  $\omega_{frame} = -0.02$ ,  $\rho_{min} = 0.001$ ,  $\text{propagation\_time} = 3.2$ ; Frame 4:  $\omega_{frame} = 0.0$ ,  $\rho_{min} = 0.001$ ,  $\text{propagation\_time} = 3.2$ . The viewer also includes a 'Turn on auto-update' checkbox, 'Update', and 'Quit' buttons.

# Video

---

# Limitations

---

- Visualization systems must provide:
  - access to pipeline specifications
  - ability to identify and change pipeline components
  - ability to execute pipelines
- The produced mashups are not a substitute for more comprehensive, domain-specific applications such as CDAT
- The integration of different libraries can sometimes be complicated by a number of practical issues (GUI toolkit)
- Automatic layout does not guarantee the most appropriate and intuitive interface is created



# Conclusions and Future Work

---

# Lots of work to do! (-:

---

- VisMashup simplifies the creation of custom visualization applications
  - Developers can quickly assemble custom applications, leveraging an existing collection of visualization pipelines and their provenance
- Evaluation of effectiveness and usability
- Explore more sophisticated techniques for mining pipeline collections
- Build mashups collaboratively

# Acknowledgments

---



VACET



CAPES

Thanks to Joel Tohline for the Astrophysics case study