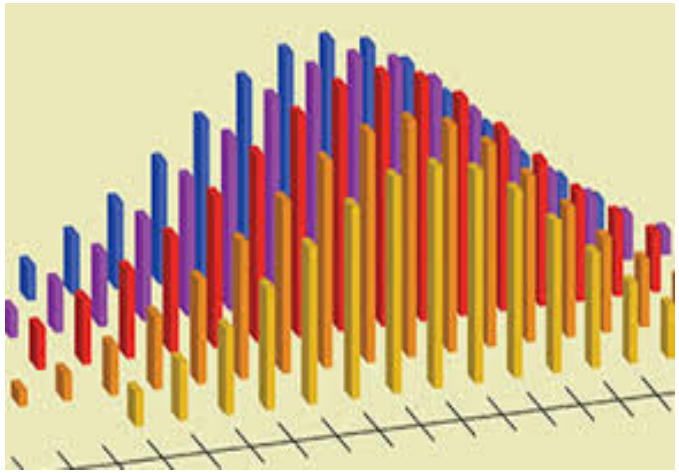
Prefuse: a toolkit for interactive information visualization

Jeffery Heer, UC Berkeley
Stuart K. Card, Palo Alto Research Center
James A. Landay, University of Washington

Infovis

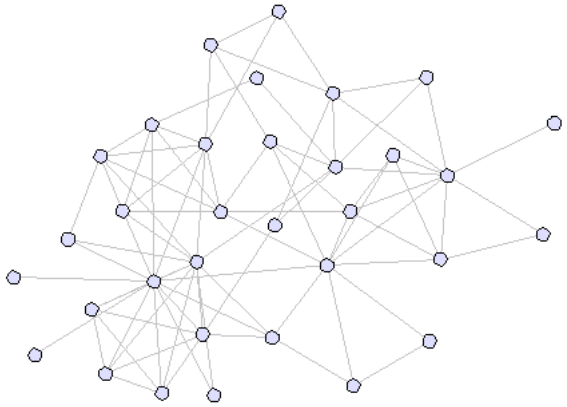
- What is information visualization (*infovis*)?

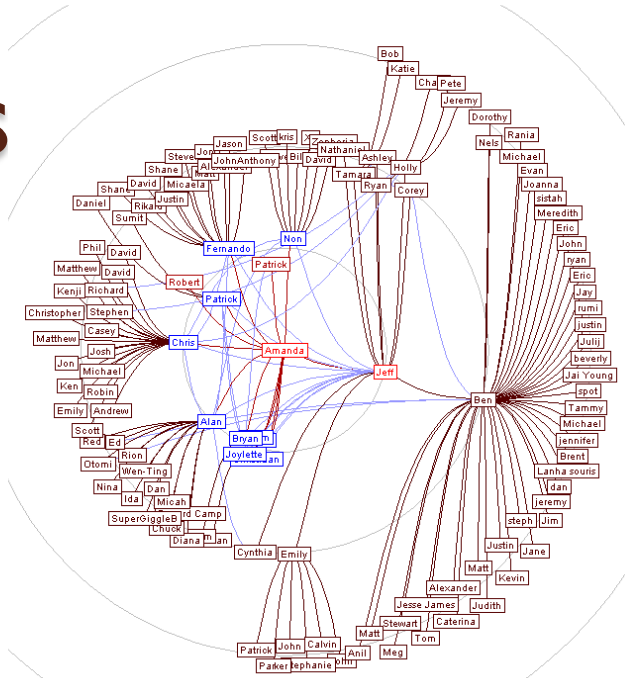
Dessert (100g serving)	Calories	Fat (g)	Carbs (g)	Protein (g)	Sodium (mg)	Calcium (%)	Iron (%)
Frozen yogurt	159	6.0	24	4.0	87	14%	1%
Ice cream sandwich	237	9.0	37	4.3	129	8%	1%
Eclair	262	16.0	24	6.0	337	6%	7%
Cupcake	305	3.7	67	4.3	413	3%	8%
Gingerbread	356	16.0	49	3.9	327	7%	16%
Jelly bean	375	0.0	94	0.0	50	0%	0%
Lollipop	392	0.2	98	0	38	0%	2%
Honeycomb	408	3.2	87	6.5	562	0%	45%
Donut	452	25.0	51	4.9	326	2%	22%
KitKat	518	26.0	65	7	54	12%	6%

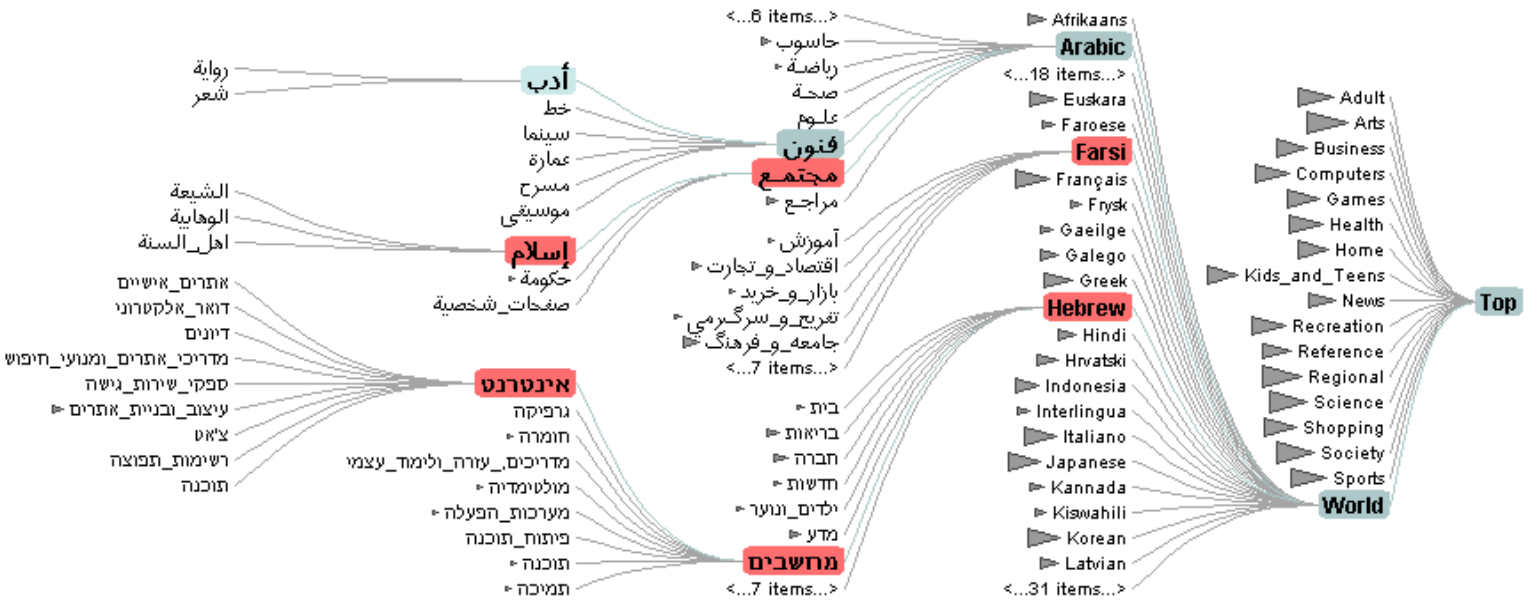


  - Visual representations of abstract information
  - Demystify data and reveal hidden patterns

Existing visualizations



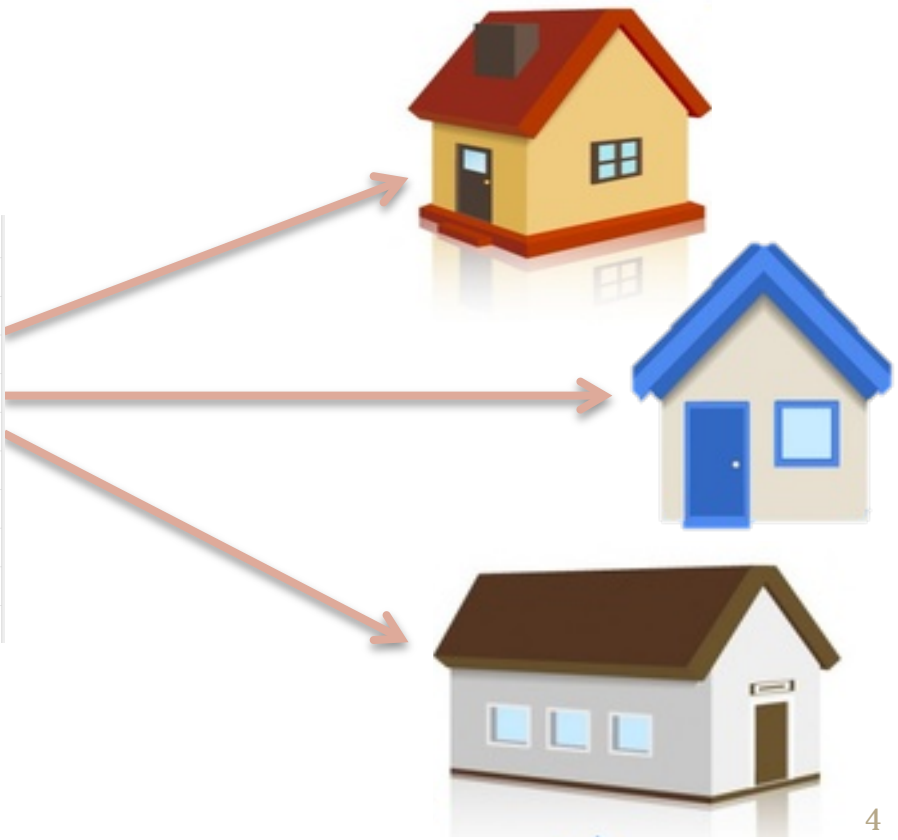




Existing visualizations

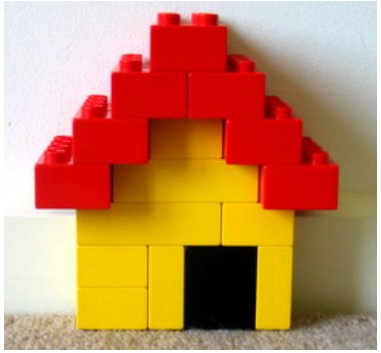
- **Constrained to one application domain**
- E.g. : Polaris, table-based
- DOITrees, tree-based

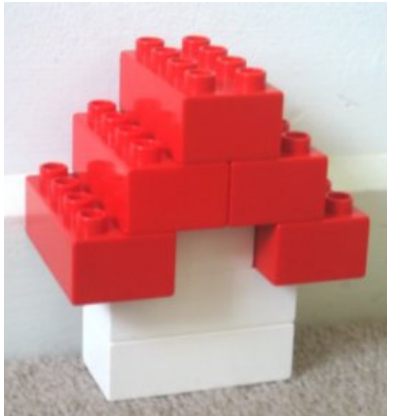
Dessert (100g serving)	Calories	Fat (g)	Carbs (g)	Protein (g)	Sodium (mg)	Calcium (%)	Iron (%)
Frozen yogurt	159	6.0	24	4.0	87	14%	1%
Ice cream sandwich	237	9.0	37	4.3	129	8%	1%
Eclair	262	16.0	24	6.0	337	6%	7%
Cupcake	305	3.7	67	4.3	413	3%	8%
Gingerbread	356	16.0	49	3.9	327	7%	16%
Jelly bean	375	0.0	94	0.0	50	0%	0%
Lollipop	392	0.2	98	0	38	0%	2%
Honeycomb	408	3.2	87	6.5	562	0%	45%
Donut	452	25.0	51	4.9	326	2%	22%
KitKat	518	26.0	65	7	54	12%	6%



What's new in Prefuse ?

What's new in Prefuse ?





composable & reusable

Prefuse:

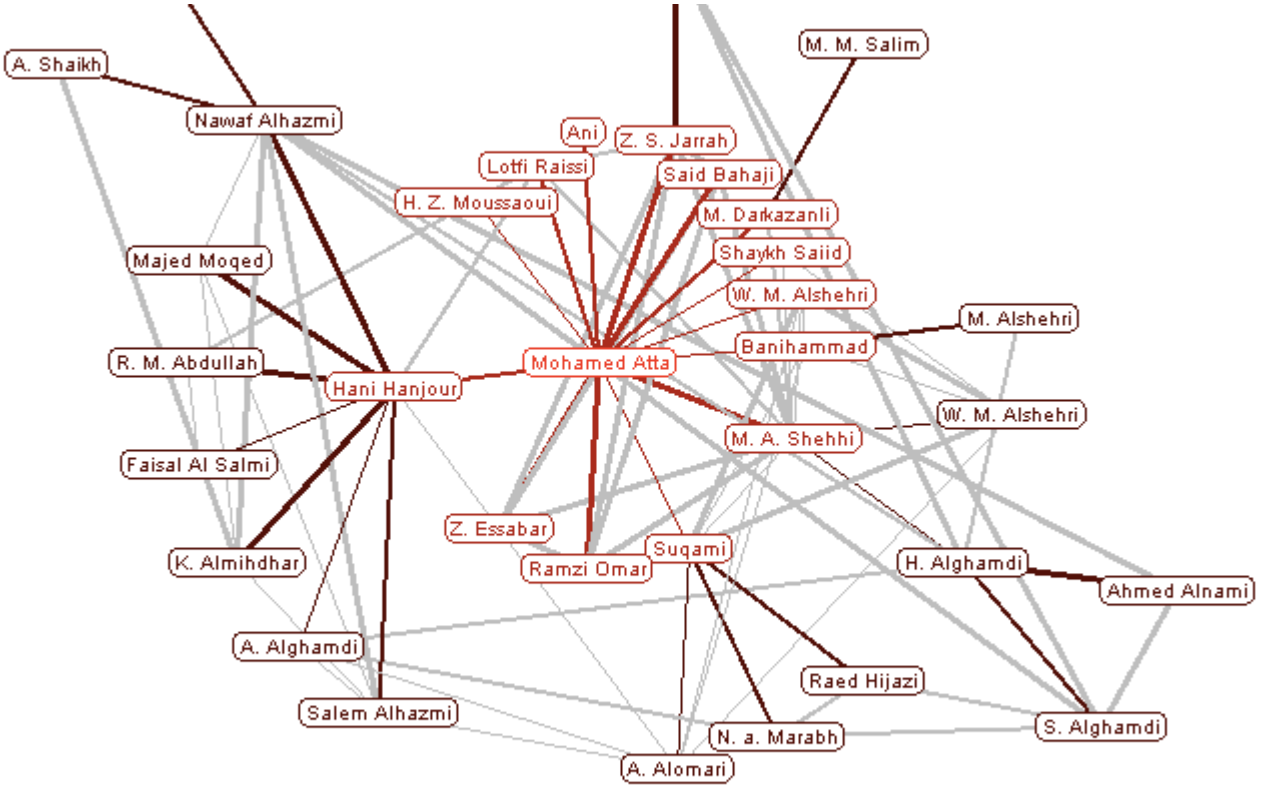
- **What?**
  - An extensible java user interface toolkit for constructing interactive information visualization applications
- **Why?**
  - Support customized visualization, animation, and interaction
- **How?**
  - Application building by stringing together fine-grained, reusable components

An example using Prefuse

An example using Prefuse

A tutorial:

- Implementing radial graph with Prefuse

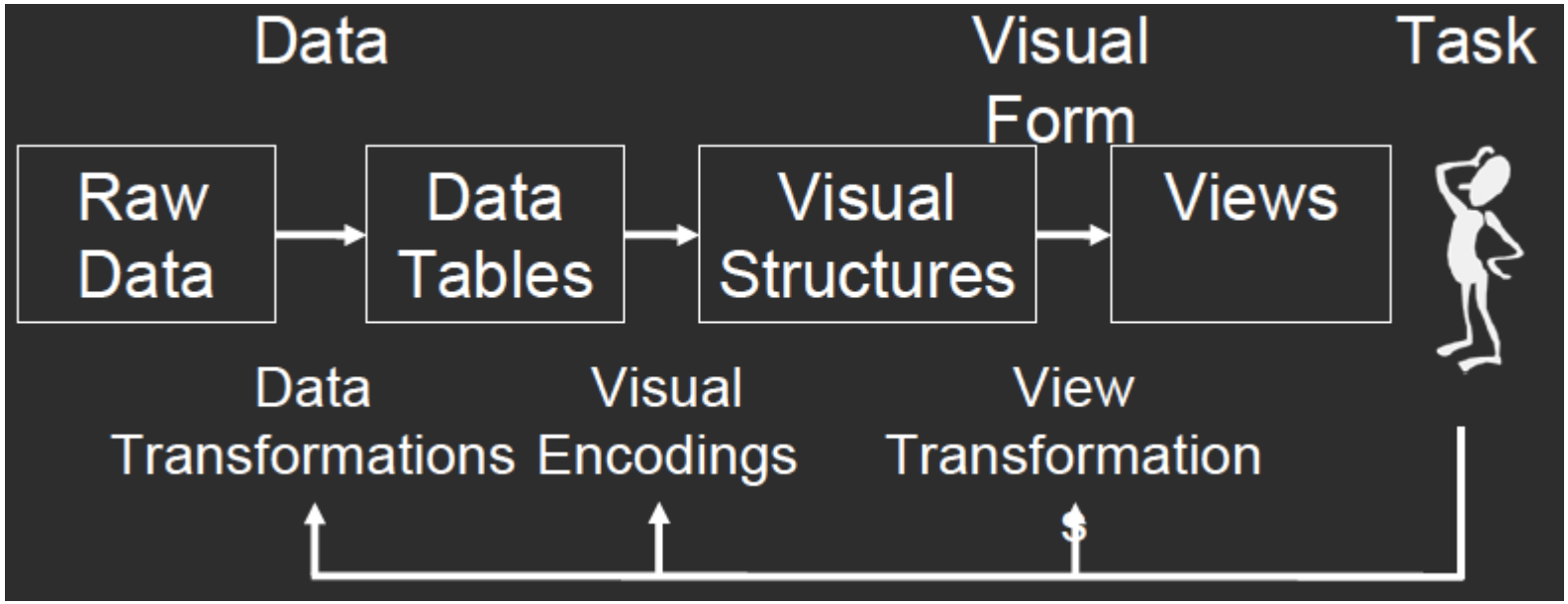


A code example

```
// create graph and registry
Graph g = new XMLGraphReader().loadGraph(datafile);
ItemRegistry registry = new ItemRegistry(g);

// intialize renderers
Renderer nodeR = new TextItemRenderer();
Renderer edgeR = new DefaultEdgeRenderer();
registry.setRendererFactory(
  new DefaultRendererFactory(nodeR, edgeR));

// initialize action lists
ActionList layout = new ActionList(registry);
layout.add(new TreeFilter(true));
layout.add(new RadialTreeLayout());
layout.add(new ColorFunction());

ActionList animate = new ActionList(registry,1500);
animate.setPacingFunction(new SlowInSlowOutPacer());
animate.add(new PolarLocationAnimator());
animate.add(new ColorAnimator());
animate.add(new RepaintAction());
animate.alwaysRunAfter(layout);

// initialize display
Display disp = new Display(registry);
disp.setSize(500,500);
disp.addControlListener(new DragControl());
disp.addControlListener(new FocusControl(layout));

// initialize enclosing window frame
JFrame frame = new JFrame("prefuse example");
frame.getContentPane().add(disp);
frame.pack(); frame.setVisible(true);

layout.runNow();
```

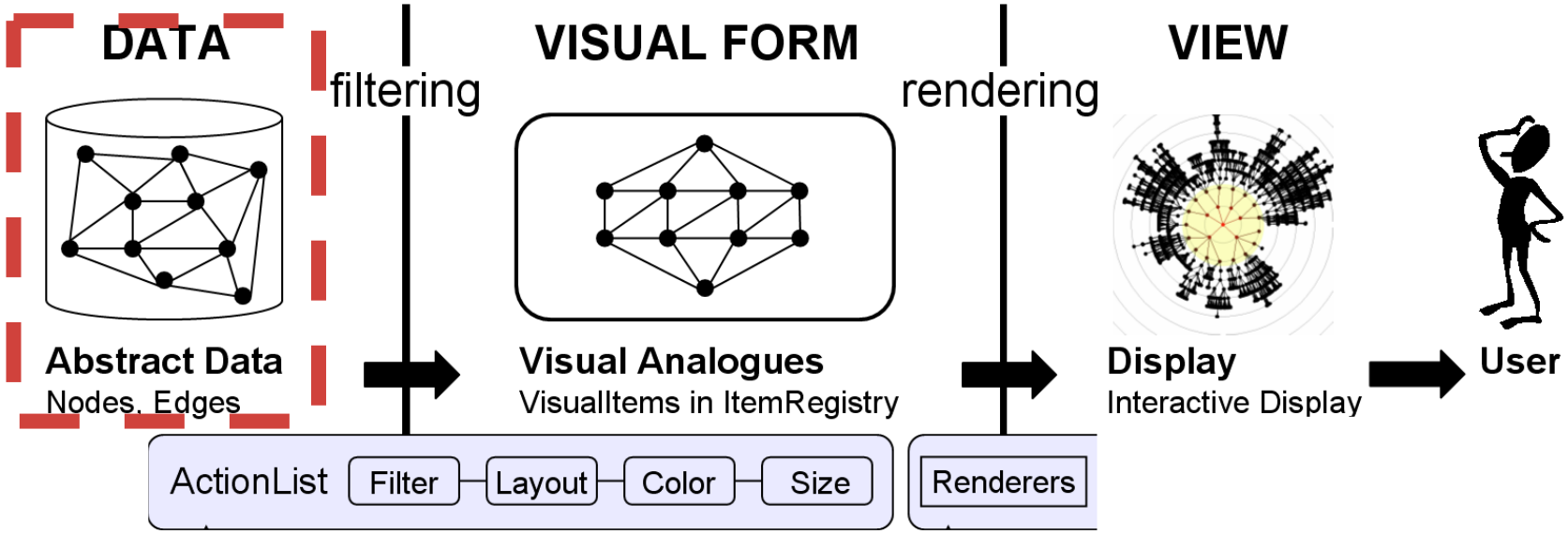
Code Sample 1: Radial Graph Explorer

A typical pipeline of *infovis*



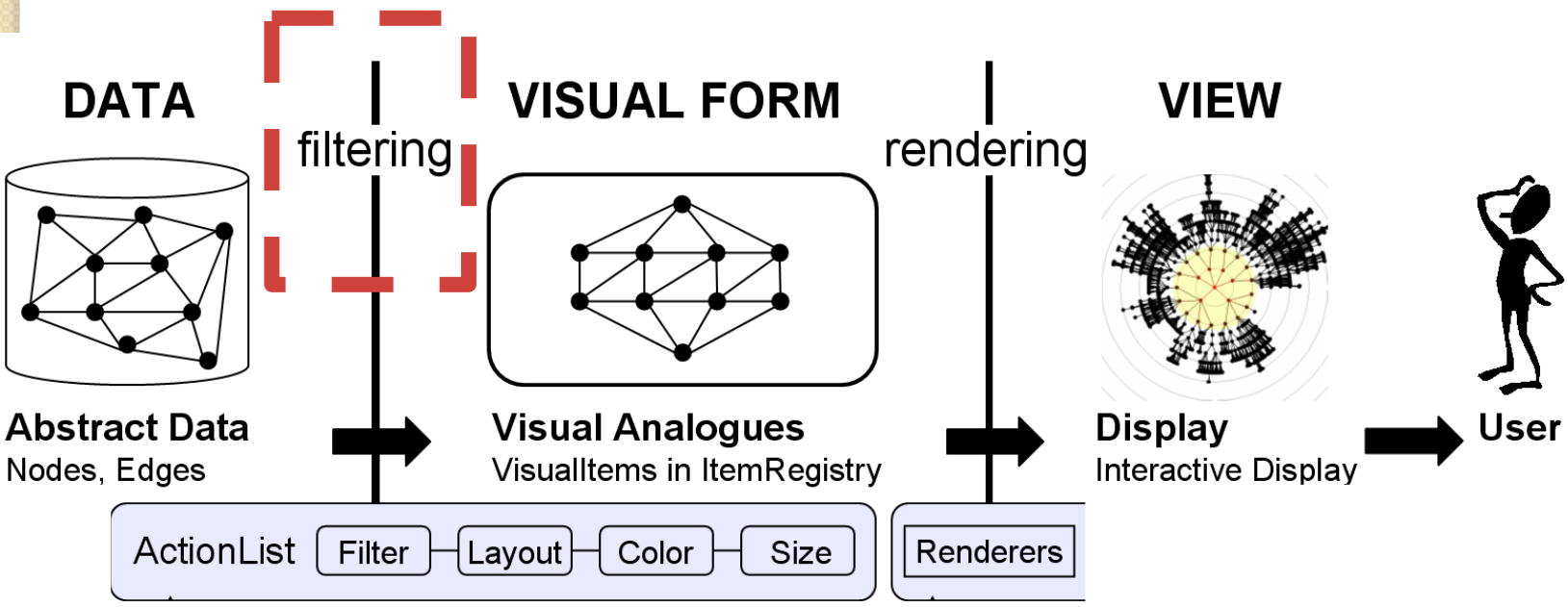
Design of Prefuse

- Prefuse provides interfaces and default implementations of data structures for unstructured, graph and tree data
- Abstract Data:
  - Data element: **entity** with attributes



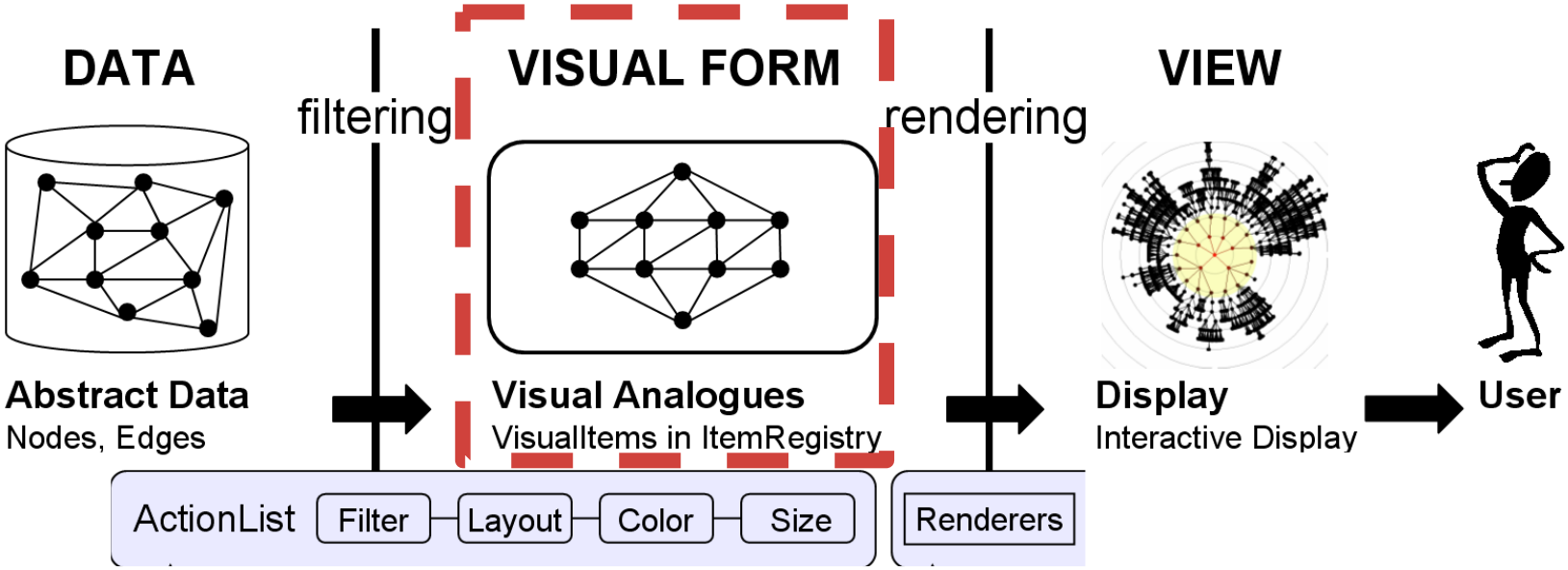
Design of Prefuse

- Filtering:
  - Process of mapping abstract data to a representation suitable for visualization then generate corresponding visual analogues



Design of Prefuse

- Visual Analogues:
  - To arrange data and stored in a centralized structure called ItemRegistry to house a specific visualization
- Prefuse provides several VisualItems to visualize different types of entities



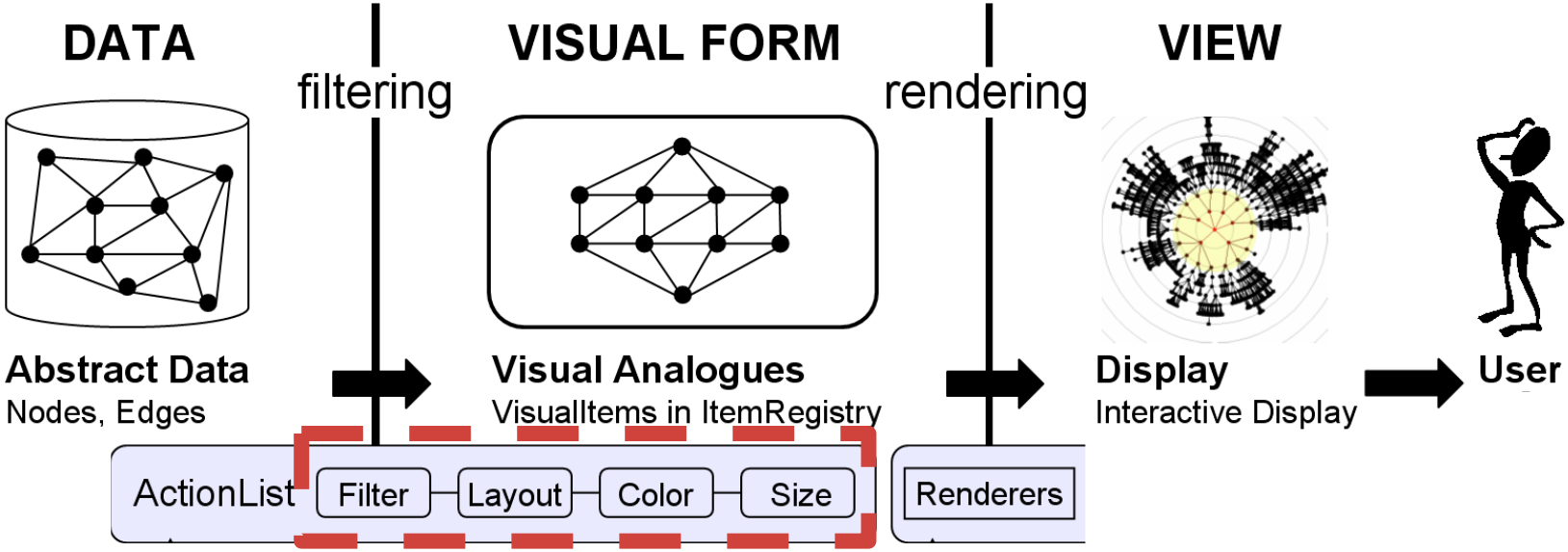
Writing apps – a code sample

```
// create graph and registry
Graph g = new XMLGraphReader().loadGraph(datafile);
ItemRegistry registry = new ItemRegistry(g);
```

- Load data
- Initialize ItemRegistry to house visualization

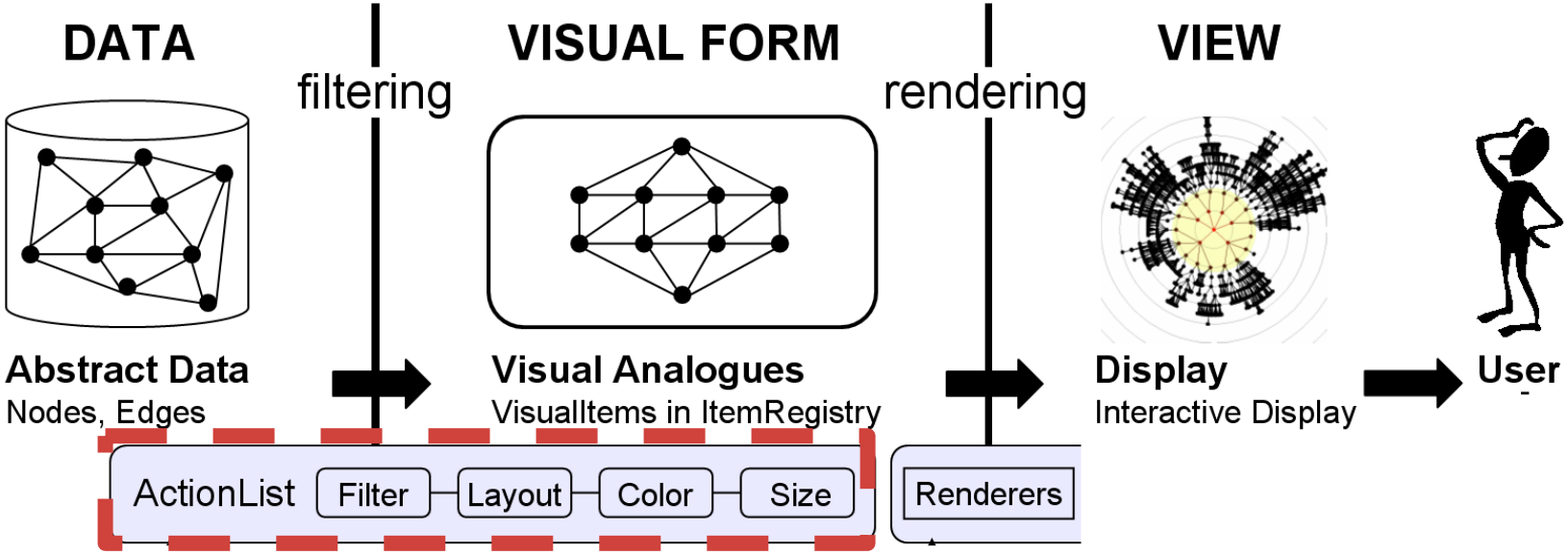
Design of Prefuse

- Actions:
  - Composable modules that update the VisualItems
  - Mechanism for selecting visualized data and setting visual properties, performing tasks such as filtering, layout, color assignment and sizing



Design of Prefuse

- ActionsList:
  - Configurable runnable Class that sequentially execute Actions



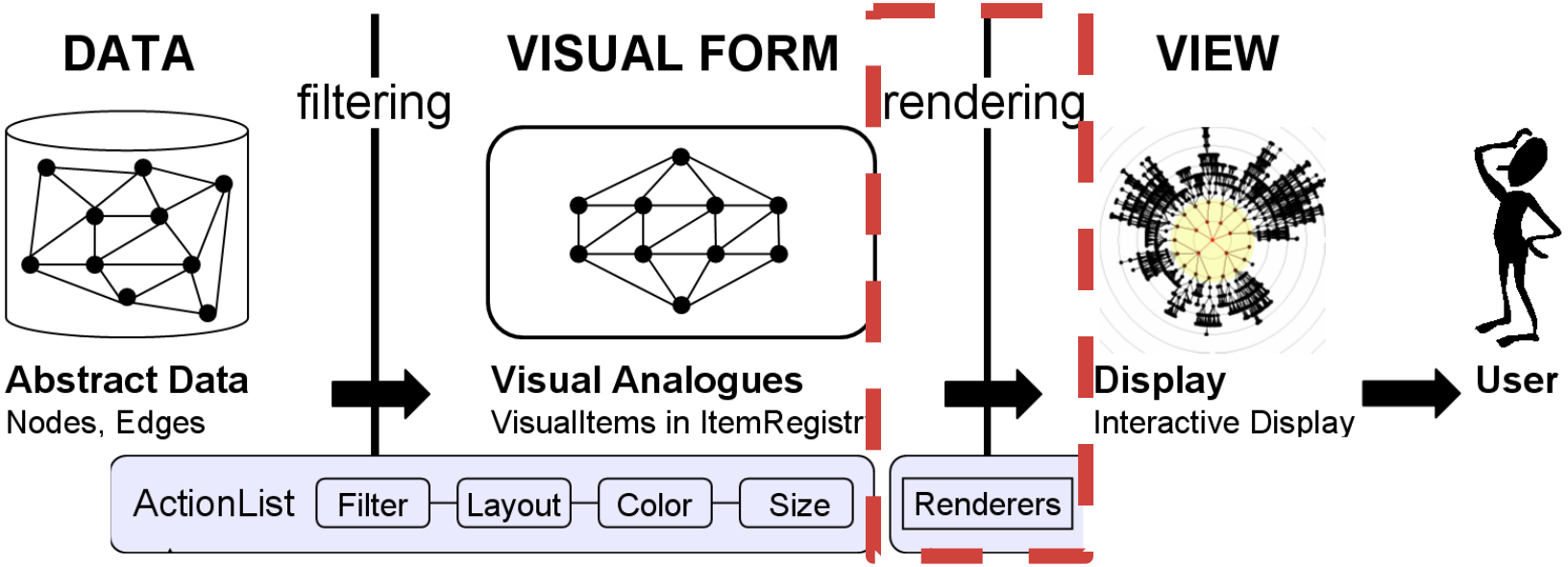
Writing apps – a code sample

```
// initialize action lists
ActionList layout = new ActionList(registry);
layout.add(new TreeFilter(true));
layout.add(new RadialTreeLayout());
layout.add(new ColorFunction());

ActionList animate = new ActionList(registry,1500);
animate.setPacingFunction(new SlowInSlowOutPacer());
animate.add(new PolarLocationAnimator());
animate.add(new ColorAnimator());
animate.add(new RepaintAction());
animate.alwaysRunAfter(layout);
```

- Specify two ActionList:
  - Filter data to tree structure, apply radial tree layout and assign colors to nodes.
  - Add an animation transition for when the focus of the visualization changes

Design of Prefuse

- Renderer:
  - Manage mappings between VisualItems and appearances



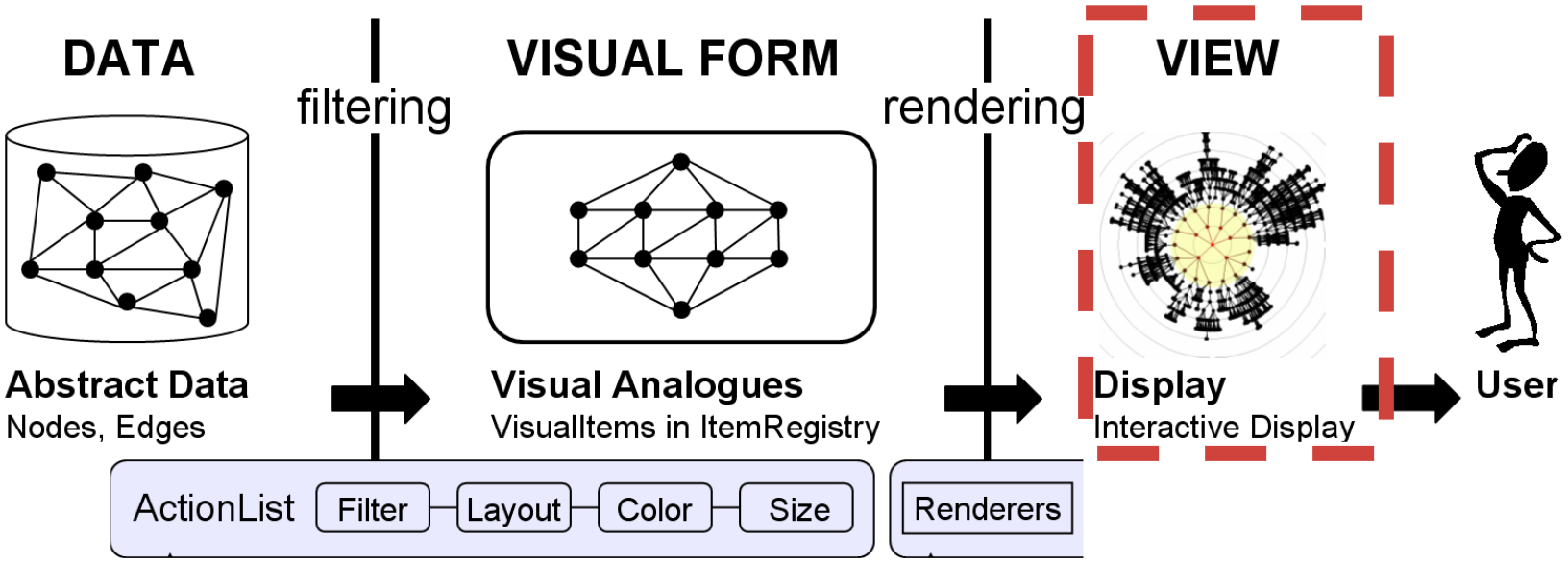
Writing apps – a code sample

```
// intialize renderers
Renderer nodeR = new TextItemRenderer();
Renderer edgeR = new DefaultEdgeRenderer();
registry.setRendererFactory(
  new DefaultRendererFactory(nodeR, edgeR));
```

- Initiate renderers
  - Assign renderers to appropriate items

Design of Prefuse

- Display:
  - Perform presentation of visualized data
    - Apply view transformations
    - Support interaction with visualized items



Writing apps – a code sample

```
// initialize display
Display disp = new Display(registry);
disp.setSize(500,500);
disp.addControlListener(new DragControl());
disp.addControlListener(new FocusControl(layout));
```

- Present visualization:
  - Enable user to reposition nodes and select new focus by clicking on a node

Radial Graph using Prefuse

Summary

- Prefuse:
  - A toolkit consists of composable, reusable units
  - Enables reuse and composition of visualization and interaction techniques

Evaluations

- How?

Summary

- Prefuse:
  - A toolkit consists of composable, reusable units
  - Enables reuse and composition of visualization and interaction techniques

Evaluations

- Application coverage
- Qualitative usability

Application Coverage

- Goal:
  - Test expressiveness and efficiency of the toolkit
- Approach:
  - Reimplement existing visualizations
    - e.g. Animated radial graphs, animated force-directed layout, the hyperbolic tree browser etc.
- Results:
  - Flexibility: implementations are greatly simplified
  - Efficiency: running time decreased from weeks or days to minutes

Qualitative Usability Study

- Goal:
  - Understand the learnability and usability of programming for other programmers
- Approach:
  - Given tutorial, observe 8 programmers of varying background and expertise
    - using the toolkit to build applications
    - interviewing them about experiences
- Results:
  - Programmers can use the toolkit to quickly build and tailor the visualizations

After Prefuse

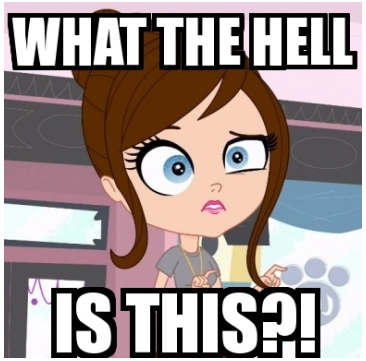
Jeffrey Heer

Follow

University of Washington
Human-Computer Interaction, HCI, Visualization, Information Visualization, Visual Analytics
Verified email at cs.washington.edu - Homepage

Title 1–20	Cited by	Year
D3: Data-Driven Documents M Bostock, V Ogievetsky, J Heer IEEE Transactions on Visualization and Computer Graphics 17 (12), 2301-2309	1216	2011
Prefuse: a toolkit for interactive information visualization J Heer, SK Card, JA Landay Proceedings of the SIGCHI conference on Human factors in computing systems ...	758	2005
Vizster: Visualizing online social networks J Heer, D Boyd IEEE Symposium on Information Visualization, 2005. INFOVIS 2005., 32-39	731	2005
Crowdsourcing graphical perception: using mechanical turk to assess visualization design J Heer, M Bostock Proceedings of the SIGCHI Conference on Human Factors in Computing Systems	394	2010

After Prefuse



Jeffrey Heer

Follow

University of Washington

Human-Computer Interaction, HCI, Visualization, Information Visualization, Visual Analytics

Verified email at cs.washington.edu - Homepage

Title 1–20	Cited by	Year
D3: Data-Driven Documents M Bostock, V Ogievetsky, J Heer IEEE Transactions on Visualization and Computer Graphics 17 (12), 2301-2309	1216	2011
Prefuse: a toolkit for interactive information visualization J Heer, SK Card, JA Landay Proceedings of the SIGCHI conference on Human factors in computing systems ...	758	2005
Vizster: Visualizing online social networks J Heer, D Boyd IEEE Symposium on Information Visualization, 2005. INFOVIS 2005., 32-39	731	2005
Crowdsourcing graphical perception: using mechanical turk to assess visualization design J Heer, M Bostock Proceedings of the SIGCHI Conference on Human Factors in Computing Systems	394	2010

D3 – Data-Driven Documents

- A JavaScript Library
- Help visualize data on web browser

D3 – not only for data scientists and data analysis

Interactive Data Analysis - Jeffrey Heer - May 23, 2013

caltech

Subscribed 42,234

D3 in European MTV Awards

Ending

jeffrey-heer.pdf

EXHIBITIONS

Apr-Sep 2005 **einfach komplex - Bildbäume und Baumbilder in der Wissenschaft**
Museum für Gestaltung, Zürich, Switzerland
Curators: Andres Janser, Marius Kwint, Barbara Bader
Exhibited images of the Degree-of-Interest Tree browser and other visualizations.

26 Mar 2005 **False Profit: LIQUIDATE**
False Profit, San Francisco, CA
Interactive installation of the Vizster social network browser.

8 Mar 2012 **The Art of Networks**
Florida Institute of Technology
Curator: Isabel Meirelles
Exhibition of the Stanford Dissertation Browser.

INVITED TALKS

Predictive Interaction
24 Feb 2016 Design@Large Seminar, UC San Diego *San Diego, CA*

Principles of Data Visualization
27 Jan 2016 Upper Columbia Science Conference *Wenatchee, WA*
5 Mar 2014 Keynote Address, Visualizing Biological Data (VIZBI) *Heidelberg, Germany*

Take away

- What is Prefuse and why it ?
- How to use it?
- How did we evaluate it?
- What did you learn from author's experience ?