Chap 15: Analysis Case Studies
Paper: D3

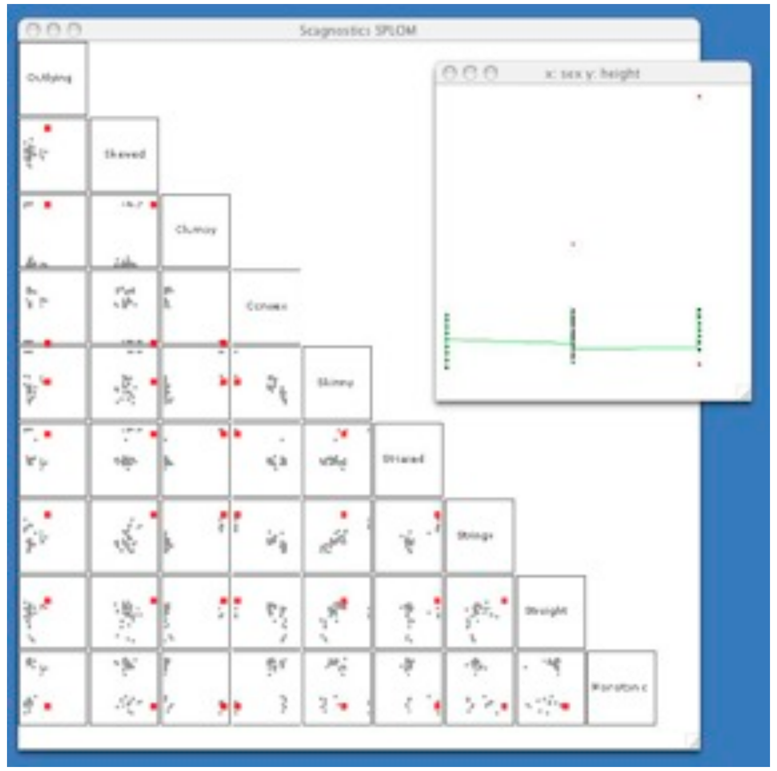
Tamara Munzner
Department of Computer Science
University of British Columbia

UBC CPSC 547: Information Visualization
Wed Nov 5 2014

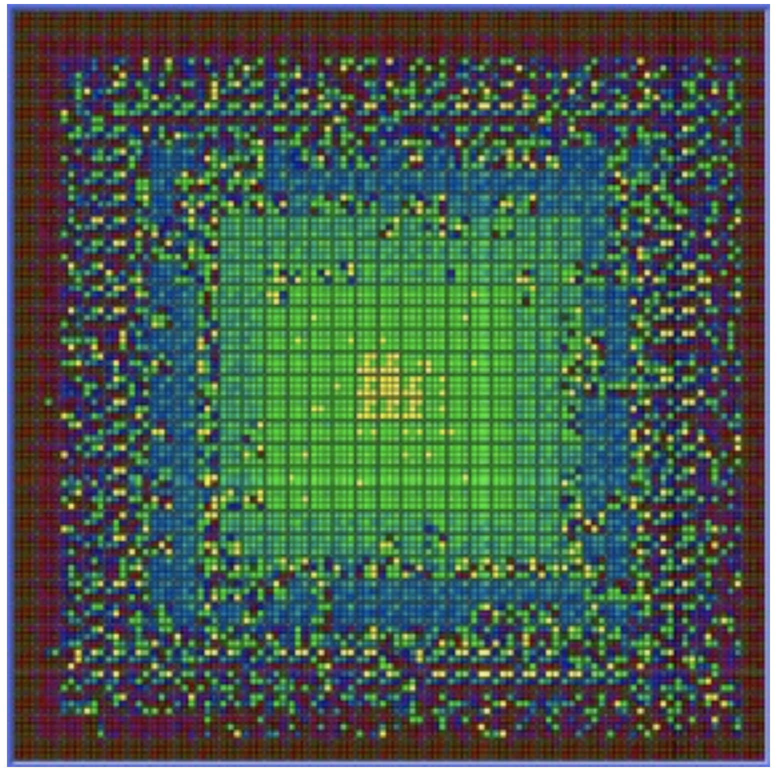
http://www.cs.ubc.ca/~tmm/courses/547-14/#chap15

Analysis Case Studies

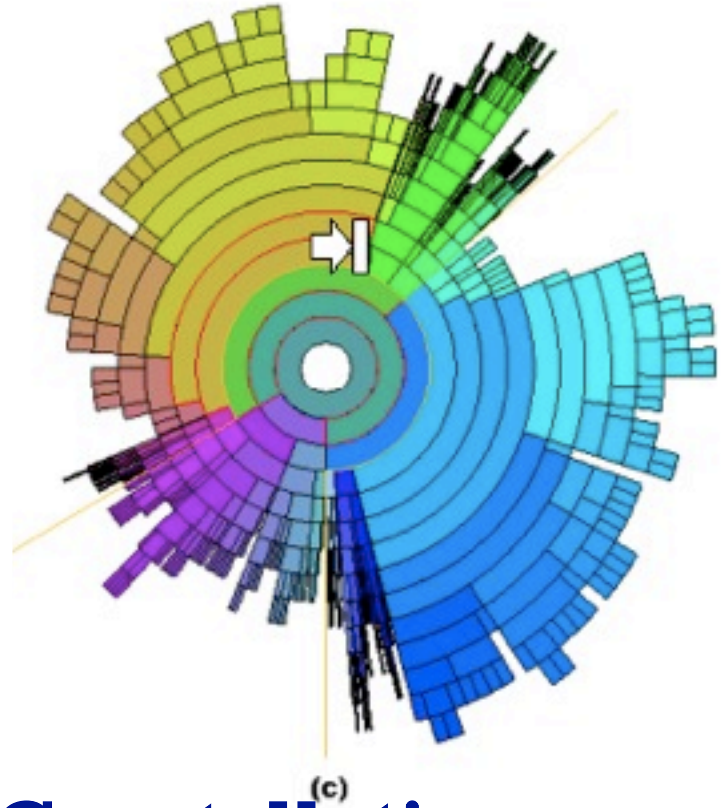
Scagnostics



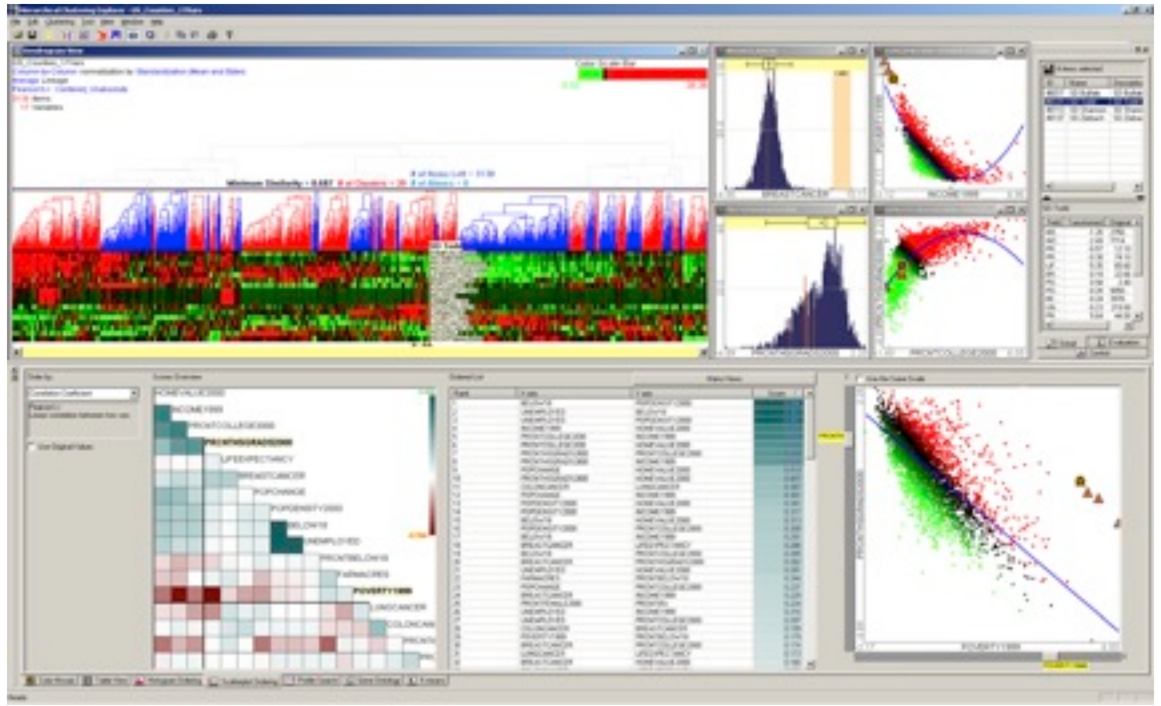
VisDB



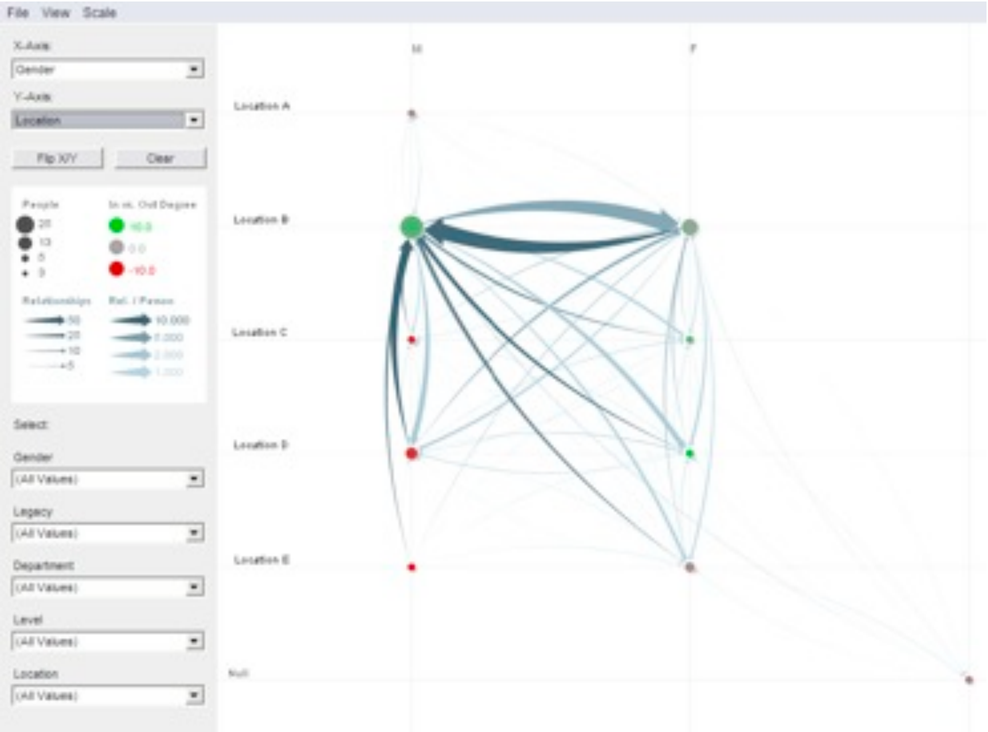
InterRing



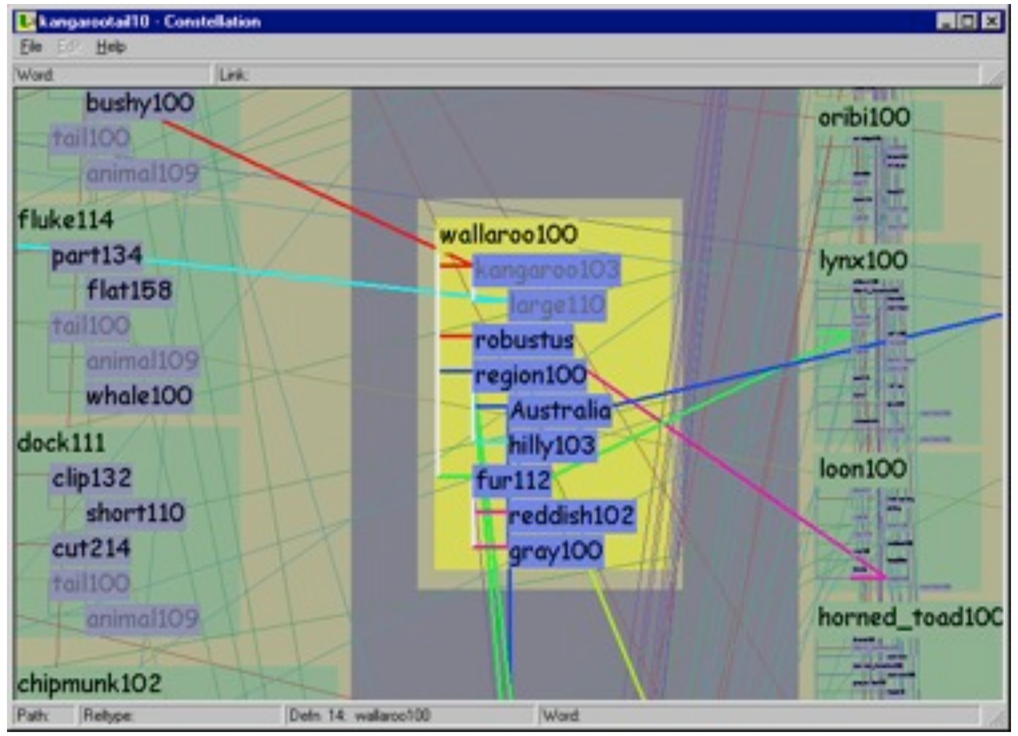
HCE



PivotGraph

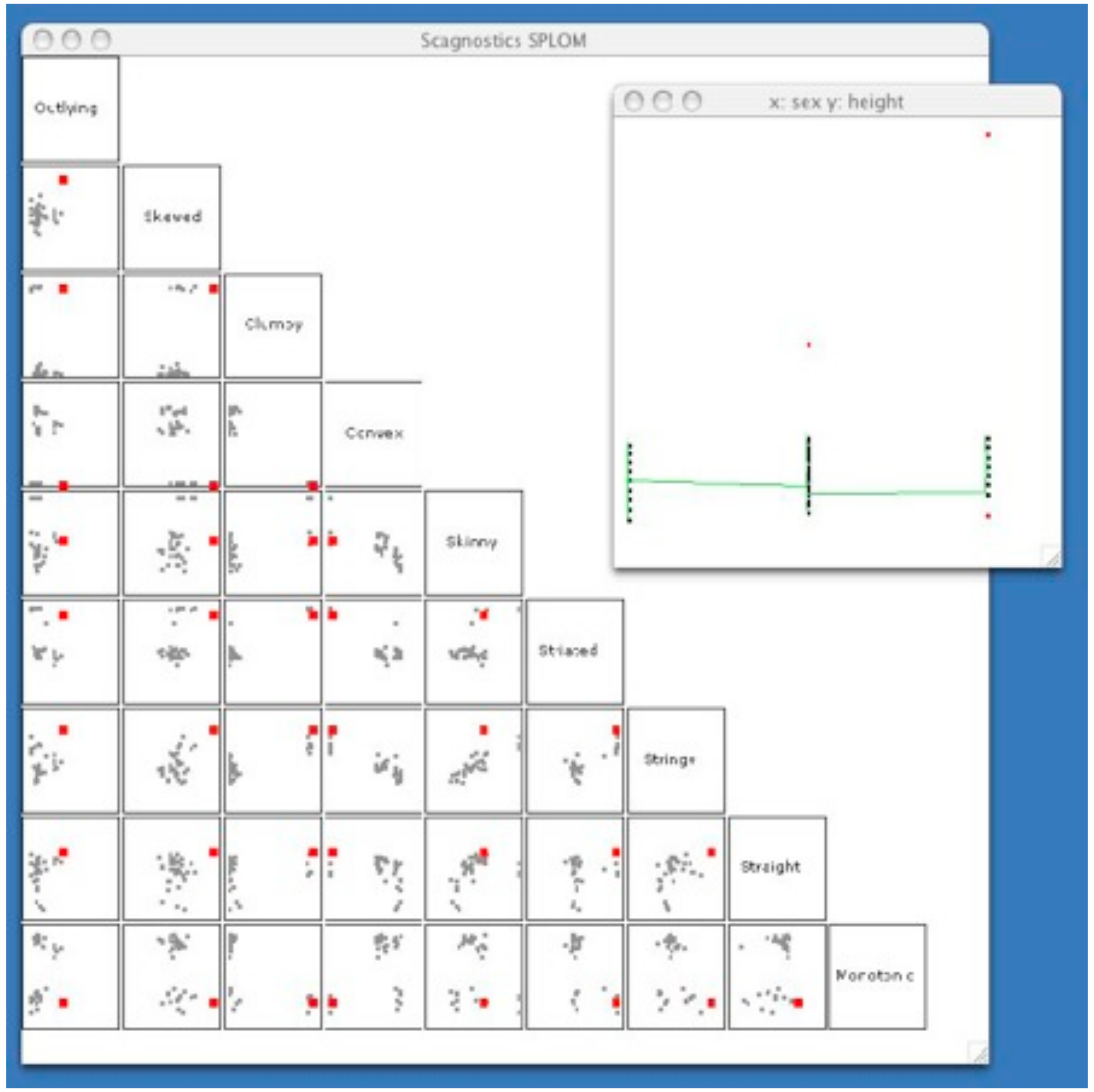


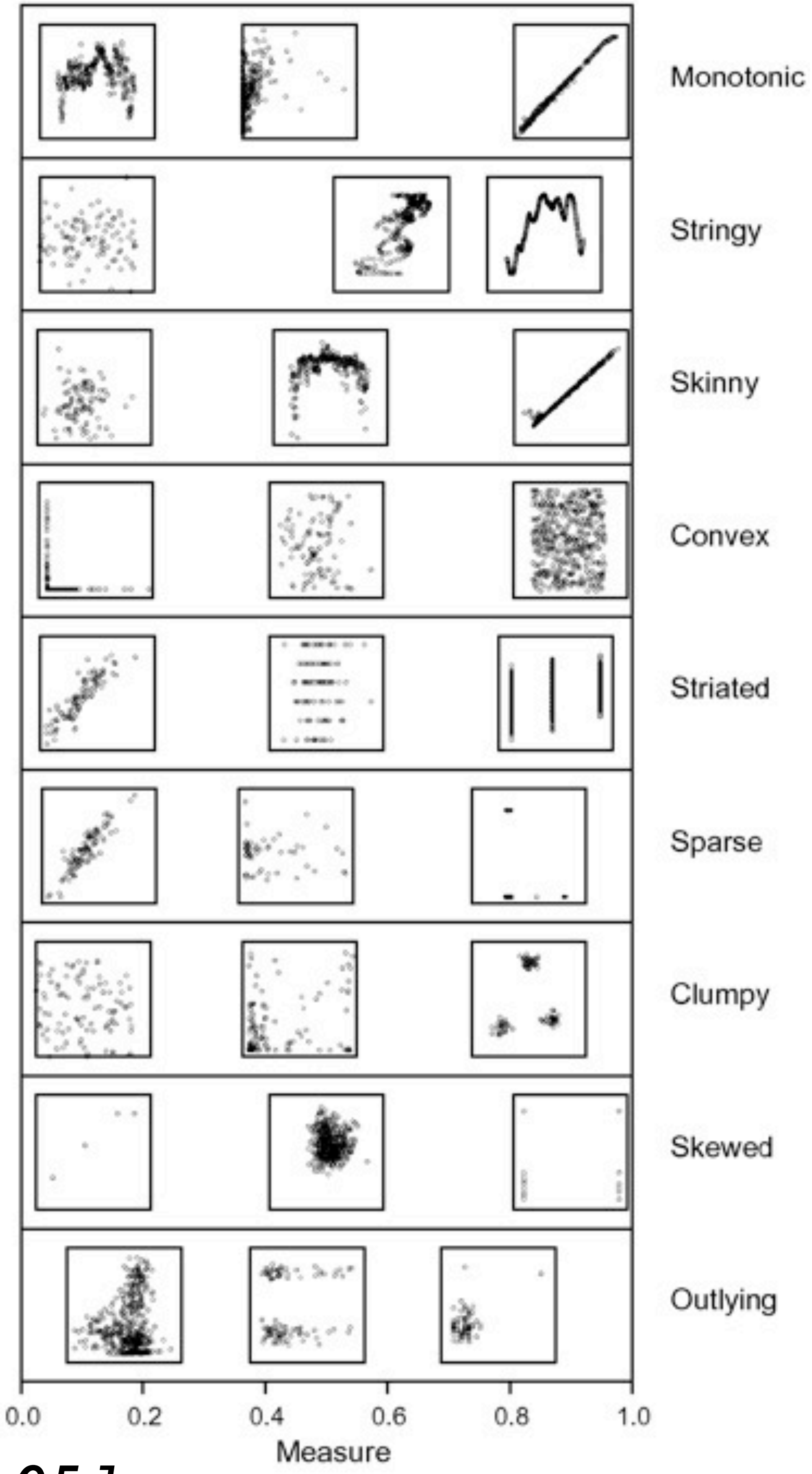
Constellation



Graph-Theoretic Scagnostics

- scatterplot diagnostics
  - scagnostics SPLOM: each point is one original scatterplot





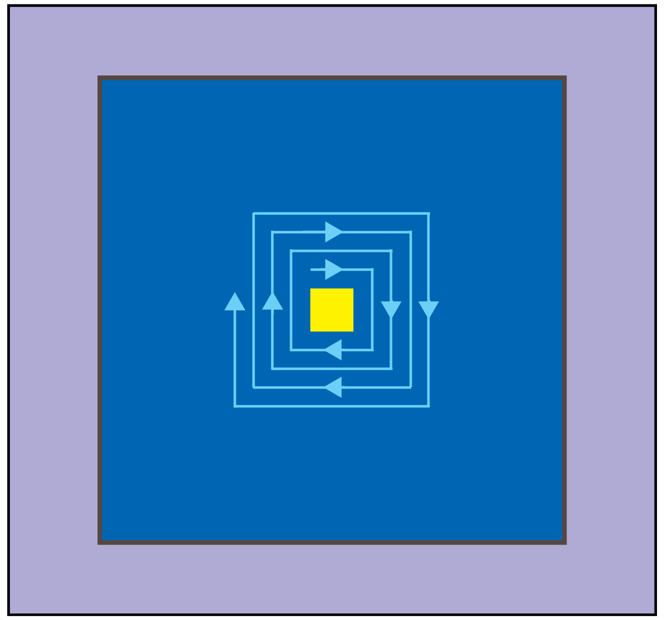
[Graph-Theoretic Scagnostics Wilkinson, Anand, and Grossman. Proc InfoVis 05.]

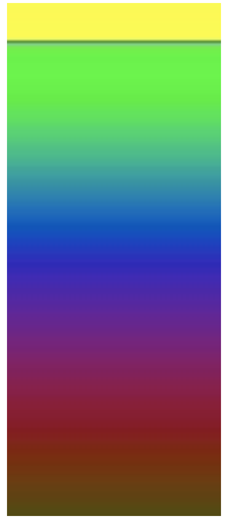
Scagnostics analysis

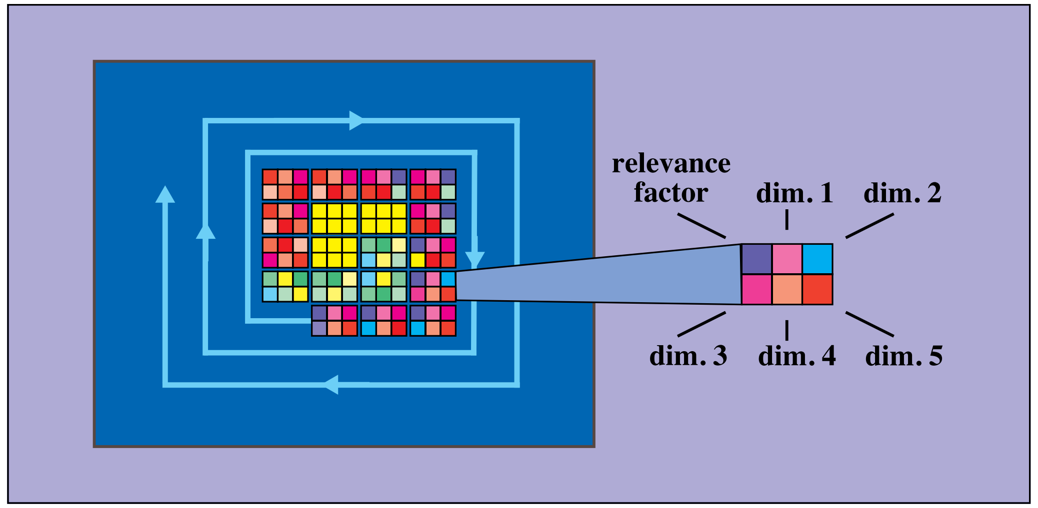
System	Scagnostics
What: Data	Table.
What: Derived	Nine quantitative attributes per scatterplot (pairwise combination of original attributes).
Why: Tasks	Identify, compare, and summarize; distributions and correlation.
How: Encode	Scatterplot, scatterplot matrix.
How: Manipulate	Select.
How: Facet	Juxtaposed small-multiple views coordinated with linked highlighting, popup detail view.
Scale	Original attributes: dozens.

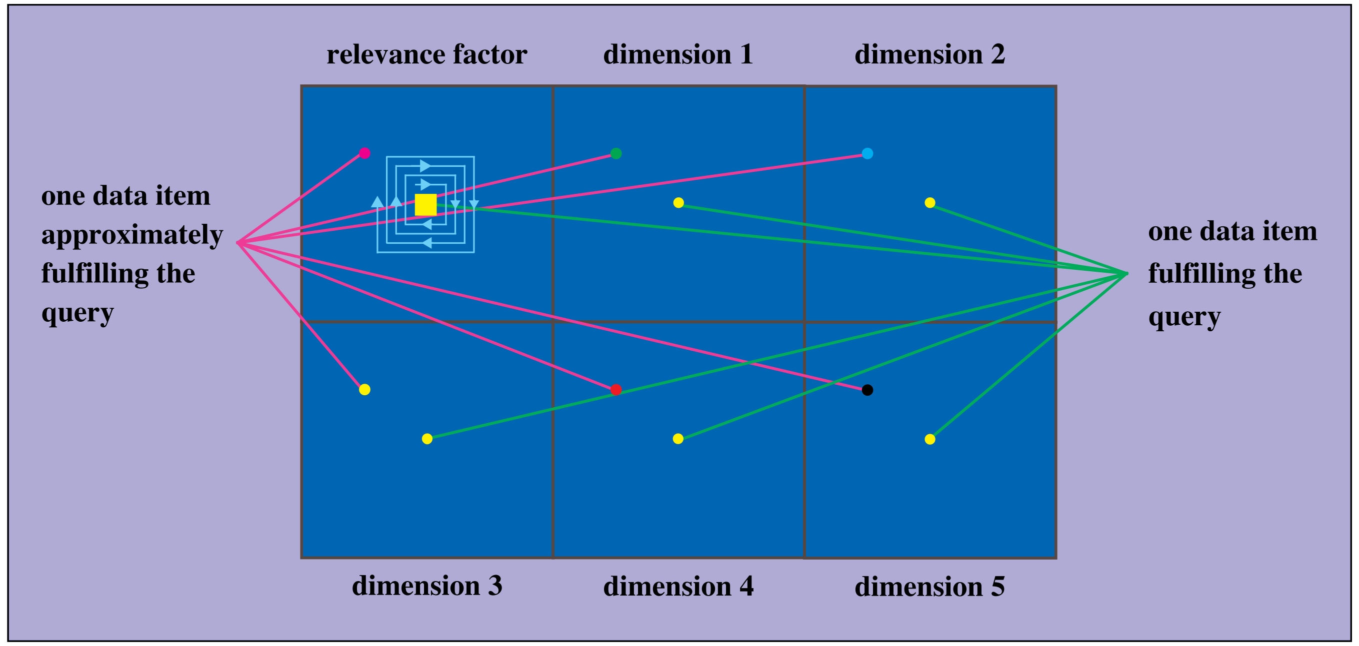
VisDB

- table: draw pixels sorted, colored by relevance
- group by attribute or partition by attribute into multiple views





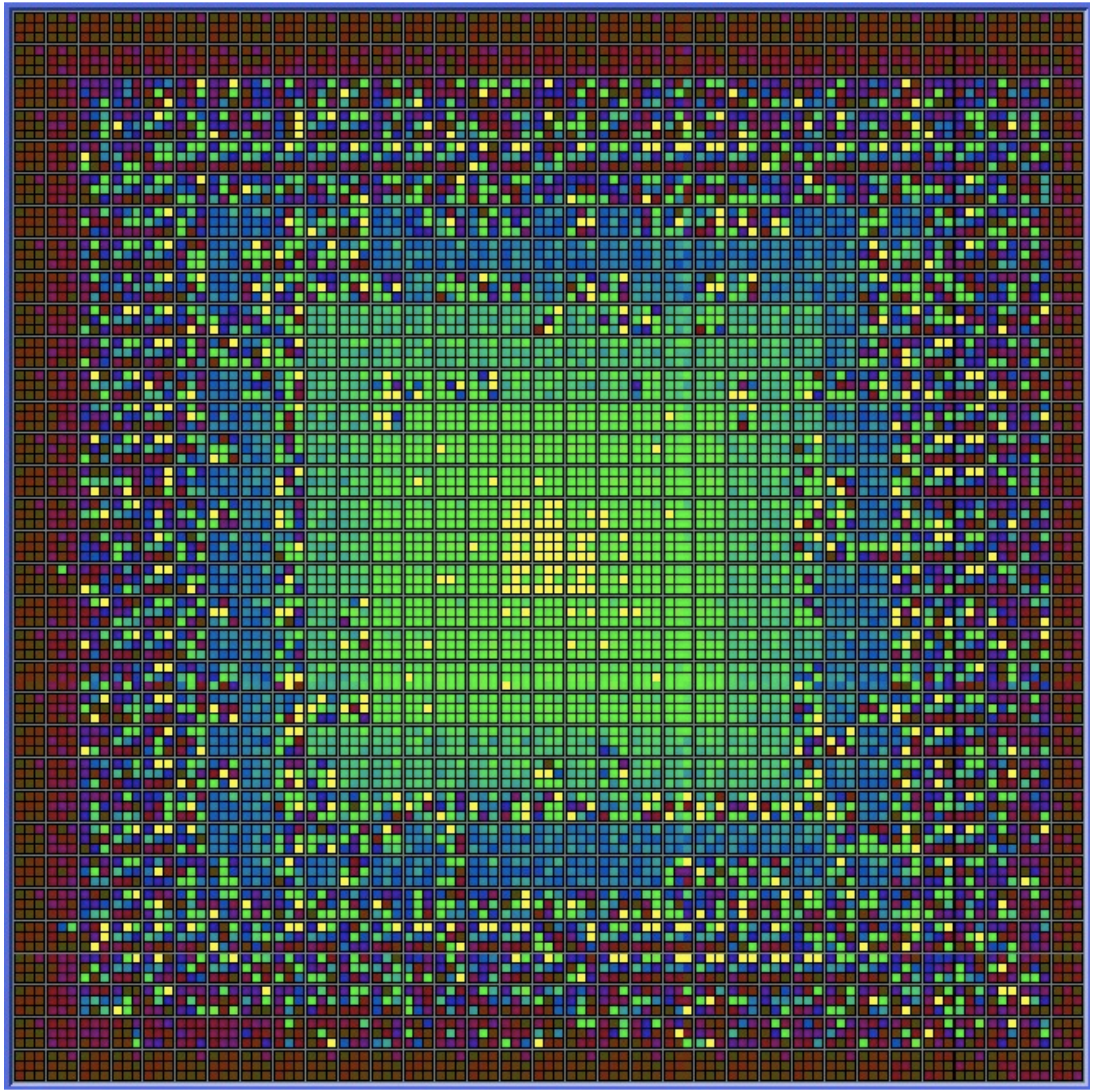




[VisDB: Database Exploration using Multidimensional Visualization, Keim and Kriegel, IEEE CG&A, 1994]

VisDB Results

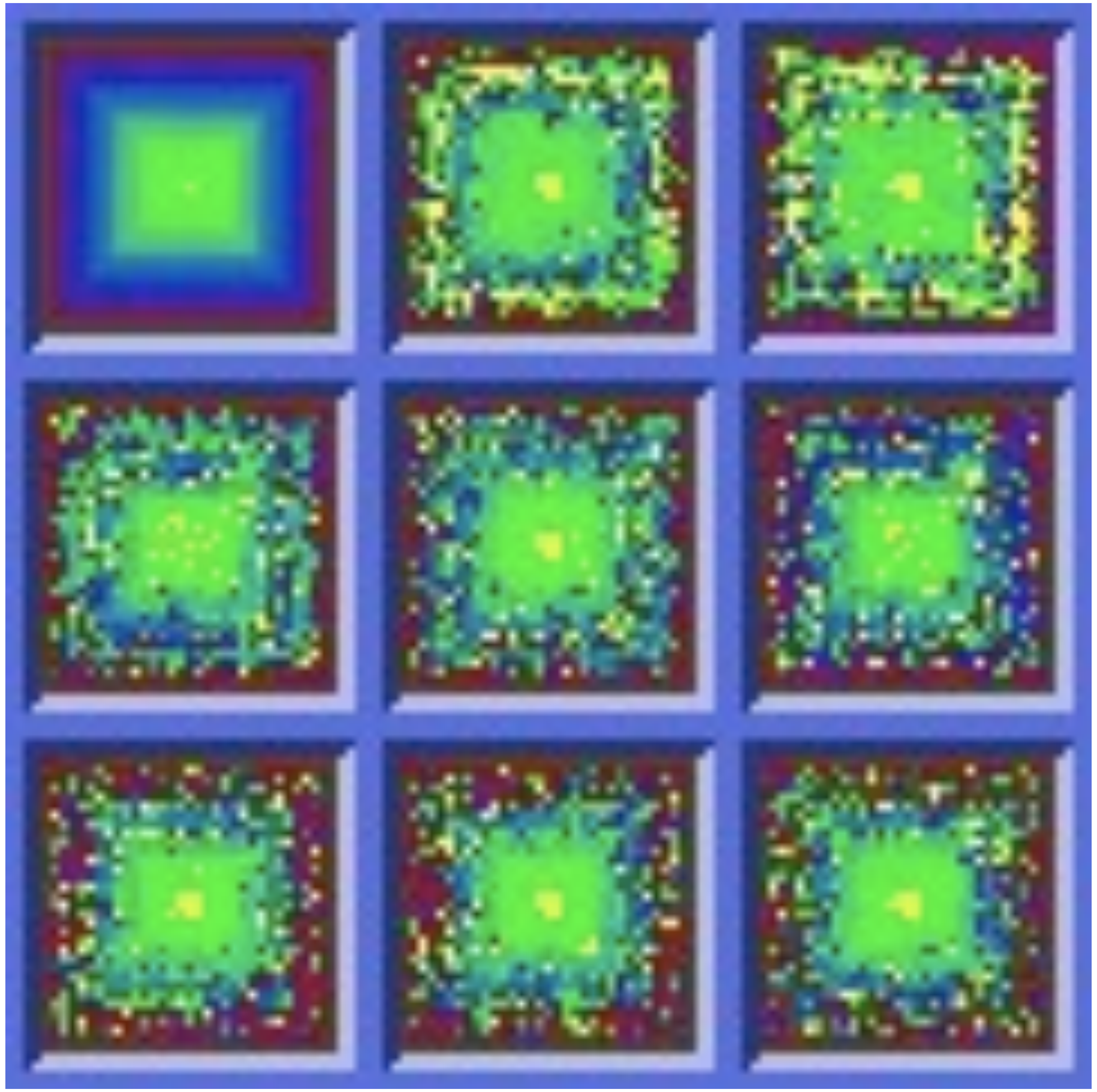
- partition into many small regions: dimensions grouped together



[VisDB: Database Exploration using Multidimensional Visualization, Keim and Kriegel, IEEE CG&A, 1994]

VisDB Results

- partition into small number of views
  - inspect each attribute



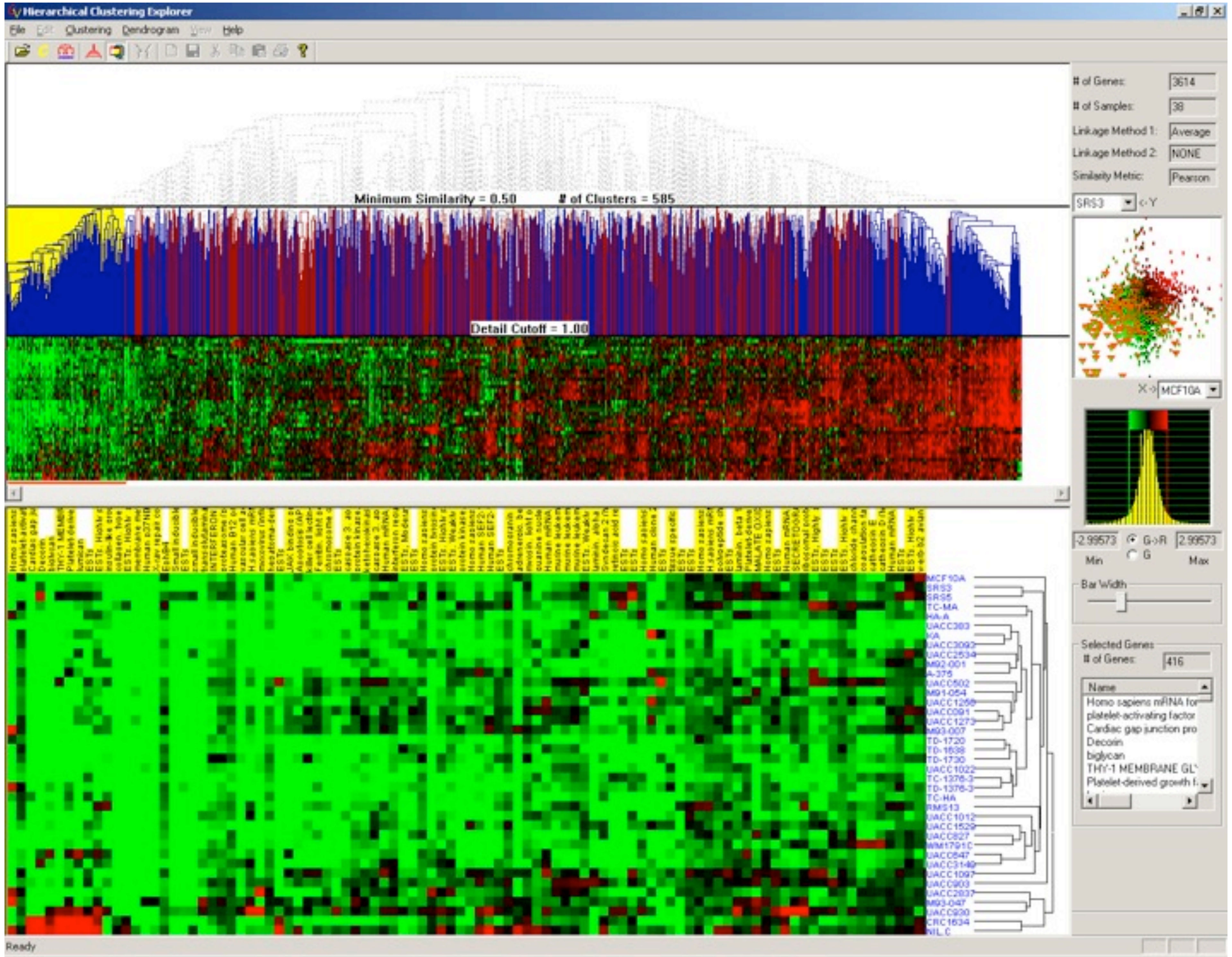
[VisDB: Database Exploration using Multidimensional Visualization, Keim and Kriegel, IEEE CG&A, 1994]

VisDB Analysis

System	VisDB
What: Data	Table (database) with k attributes; query returning table subset (database query).
What: Derived	$k+1$ quantitative attributes per original item: query relevance for the k original attributes plus overall relevance.
Why: Tasks	Characterize distribution within attribute, find groups of similar values within attribute, find outliers within attribute, find correlation between attributes, find similar items.
How: Encode	Dense, space-filling; area marks in spiral layout; colormap: categorical hues and ordered luminance.
How: Facet	Layout 1: partition by attribute into per-attribute views, small multiples. Layout 2: partition by items into per-item glyphs.
How: Reduce	Filtering
Scale	Attributes: one dozen. Total items: several million. Visible items (using multiple views, in total): one million. Visible items (using glyphs): 100,000

Hierarchical Clustering Explorer

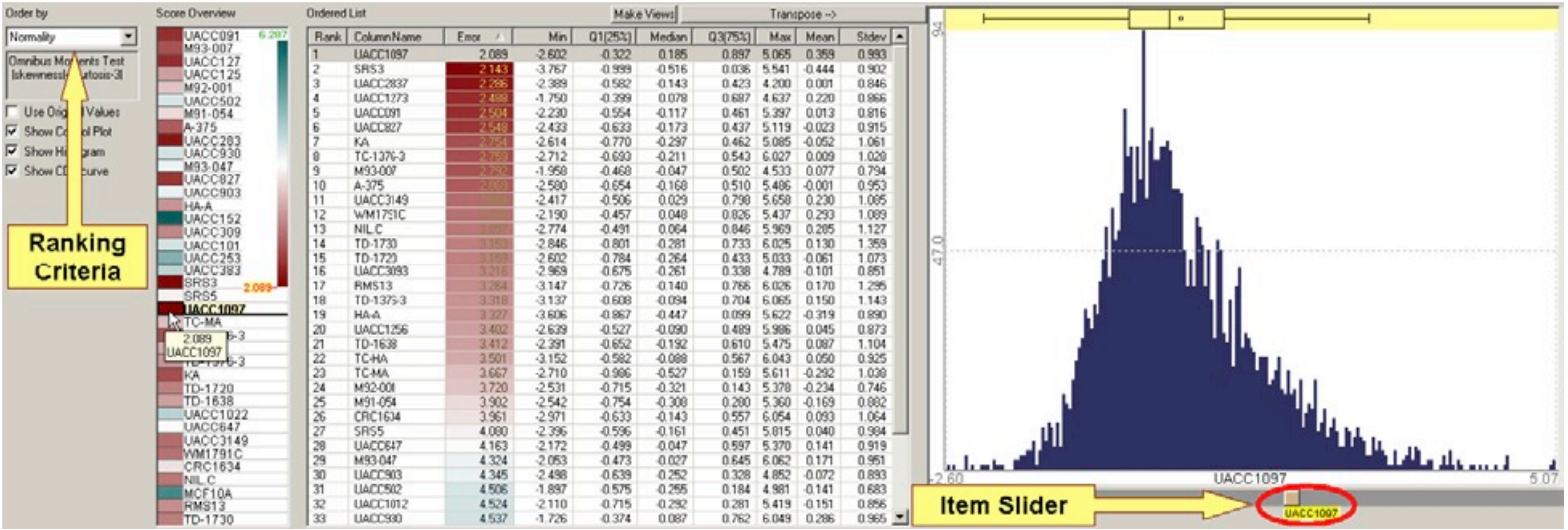
- heatmap, dendrogram
- multiple views

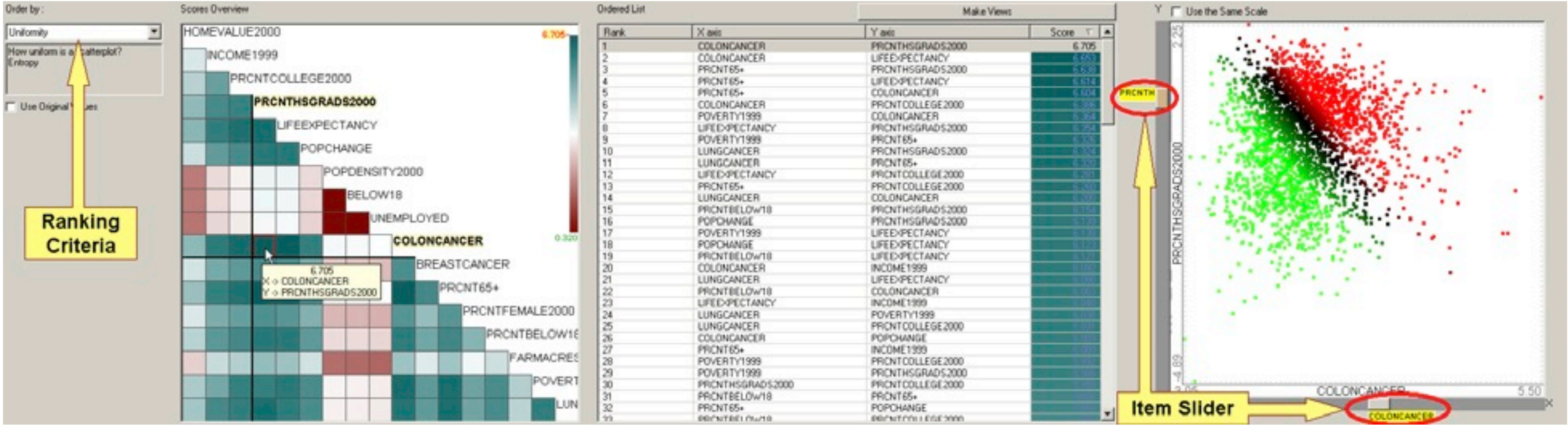


[Interactively Exploring Hierarchical Clustering Results. Seo and Shneiderman, IEEE Computer 35(7): 80-86 (2002)]

HCE

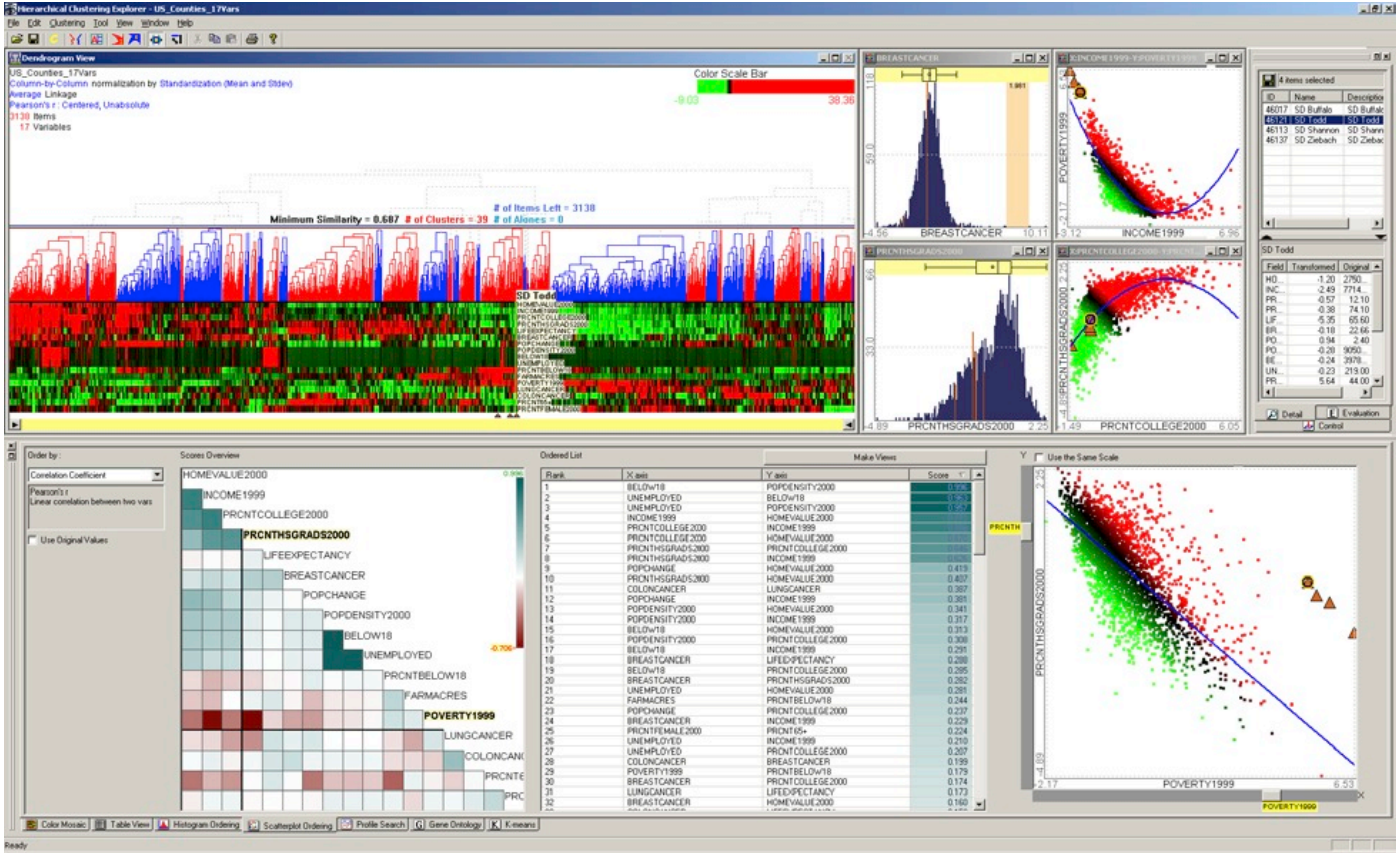
- rank by feature idiom
  - 1D list
  - 2D matrix





A rank-by-feature framework for interactive exploration of multidimensional data. Seo and Shneiderman. Information Visualization 4(2): 96-113 (2005)

HCE

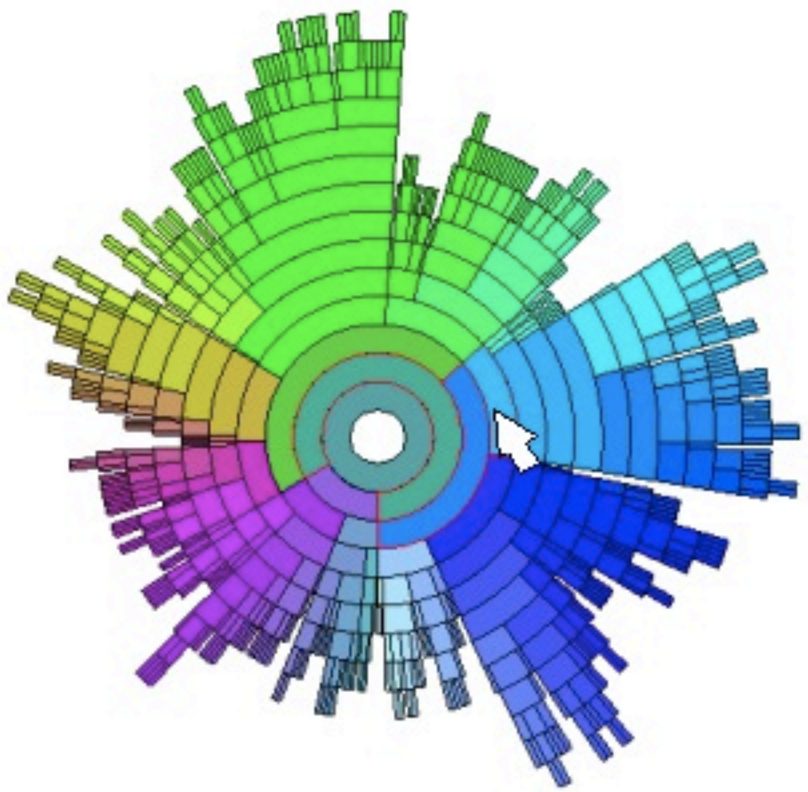


A rank-by-feature framework for interactive exploration of multidimensional data. Seo and Shneiderman. Information Visualization 4(2): 96-113 (2005)

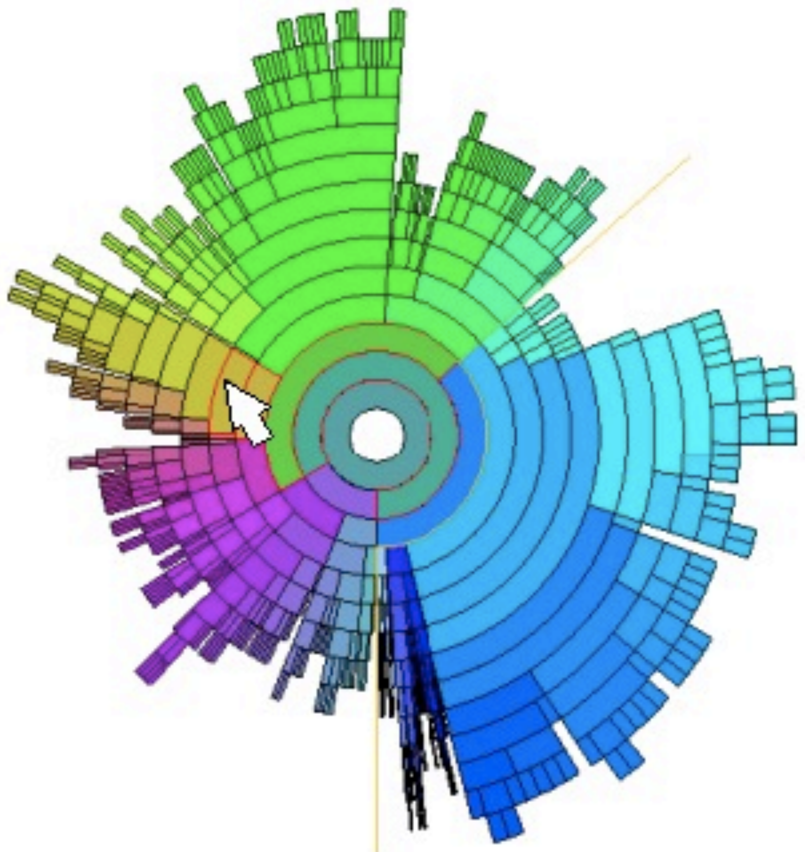
HCE Analysis

System	Hierarchical Clustering Explorer (HCE)
What: Data	Multidimensional table: two categorical key attributes (genes, conditions); one quantitative value attribute (gene activity level in condition).
What: Derived	Hierarchical clustering of table rows and columns (for cluster heatmap); quantitative derived attributes for each attribute and pairwise attribute combination; quantitative derived attribute for each ranking criterion and original attribute combination.
Why: Tasks	Find correlation between attributes; find clusters, gaps, outliers, trends within items.
How: Encode	Cluster heatmap, scatterplots, histograms, boxplots. Rank-by-feature overviews: continuous diverging colormaps on area marks in reorderable 2D matrix or 1D list alignment.
How: Reduce	Dynamic filtering; dynamic aggregation.
How: Manipulate	Navigate with pan/scroll.
How: Facet	Multiform with linked highlighting and shared spatial position; overview–detail with selection in overview populating detail view.
Scale	Genes (key attribute): 20,000. Conditions (key attribute): 80. Gene activity in condition (quantitative value attribute): $20{,}000 \times 80 = 1{,}600{,}000$.

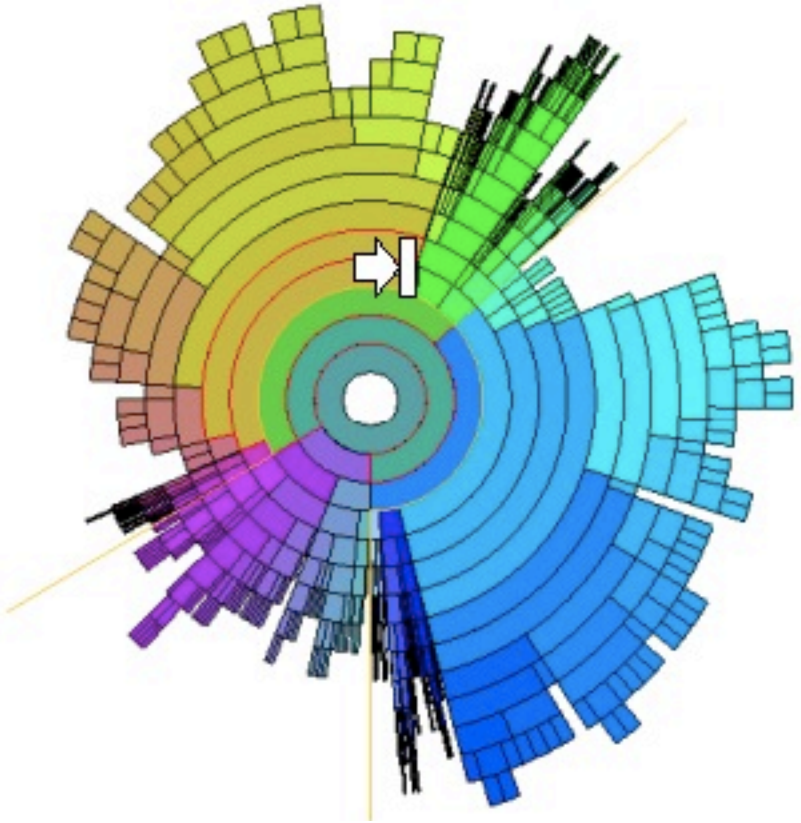
InterRing



original hierarchy



blue subtree expanded



tan subtree expanded

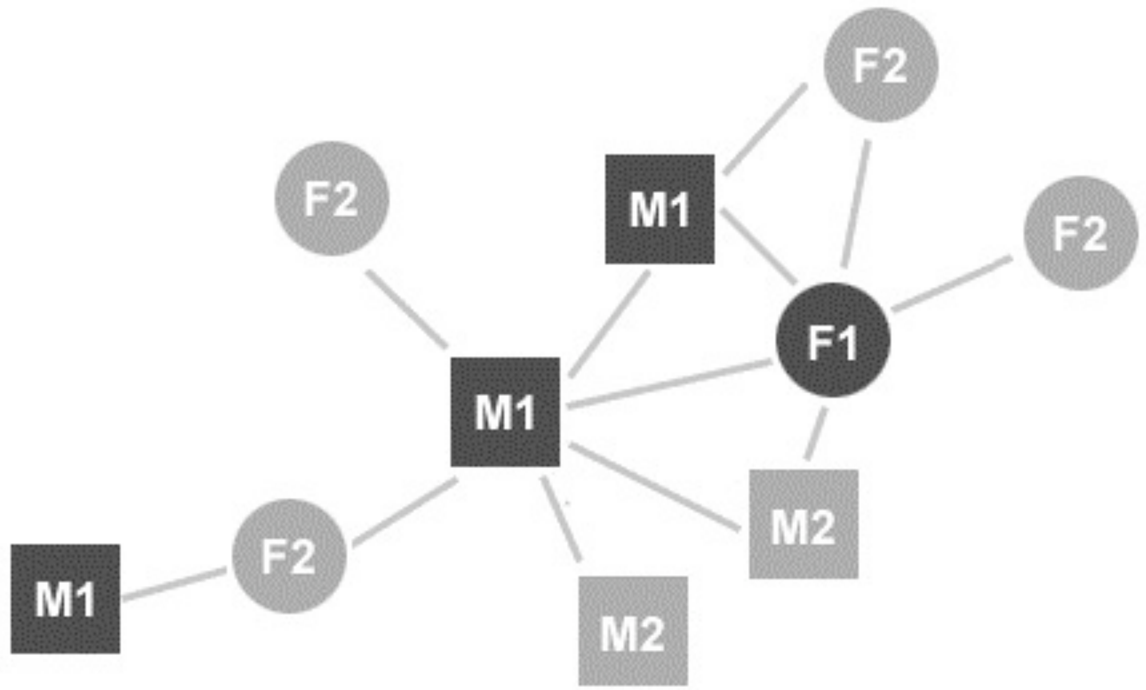
[InterRing: An Interactive Tool for Visually Navigating and Manipulating Hierarchical Structures. Yang, Ward, Rundensteiner. Proc. InfoVis 2002, p 77-84.]

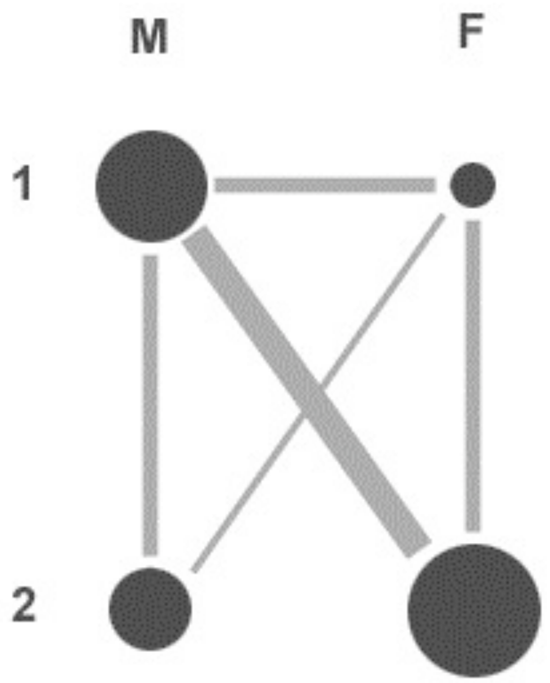
InterRing Analysis

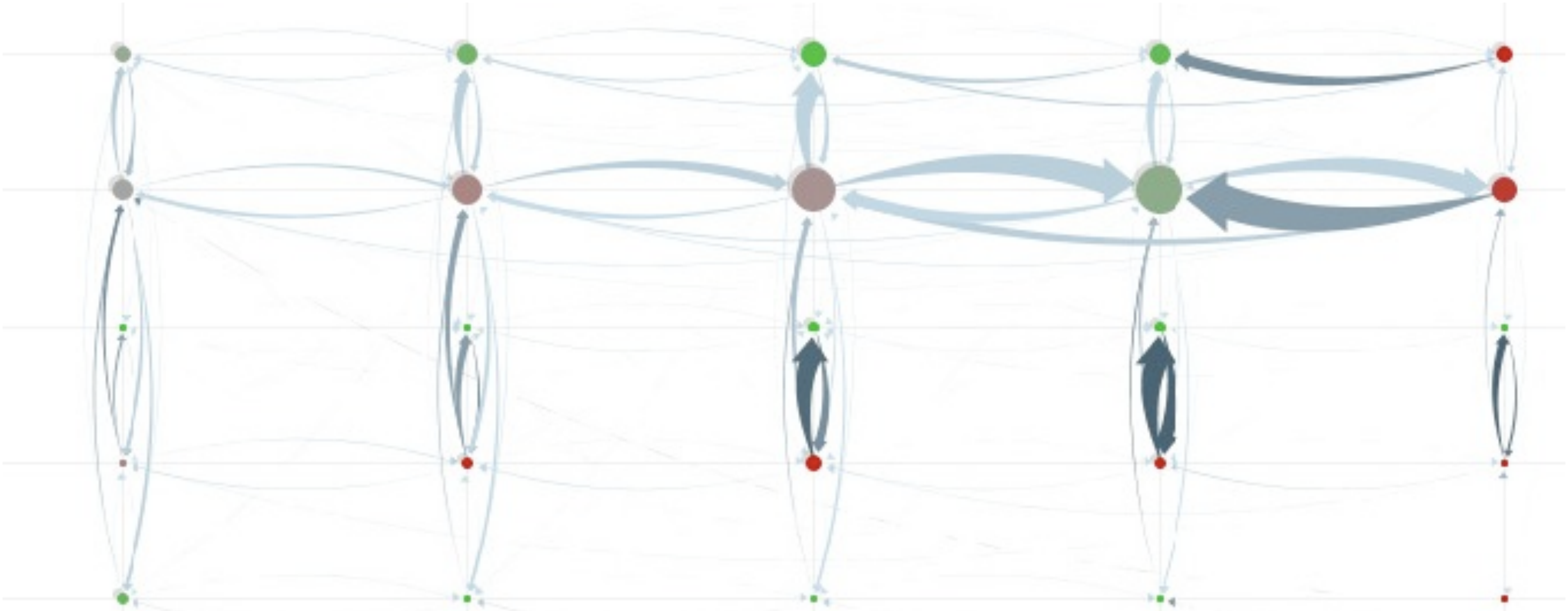
System	**InterRing**
What: Data	Tree.
Why: Tasks	Selection, rollup/drilldown, hierarchy editing.
How: Encode	Radial, space-filling layout. Color by tree structure.
How: Facet	Linked coloring and highlighting.
How: Reduce	Embed: distort; multiple foci.
Scale	Nodes: hundreds if labeled, thousands if dense. Levels in tree: dozens.

PivotGraph

- derived rollup network

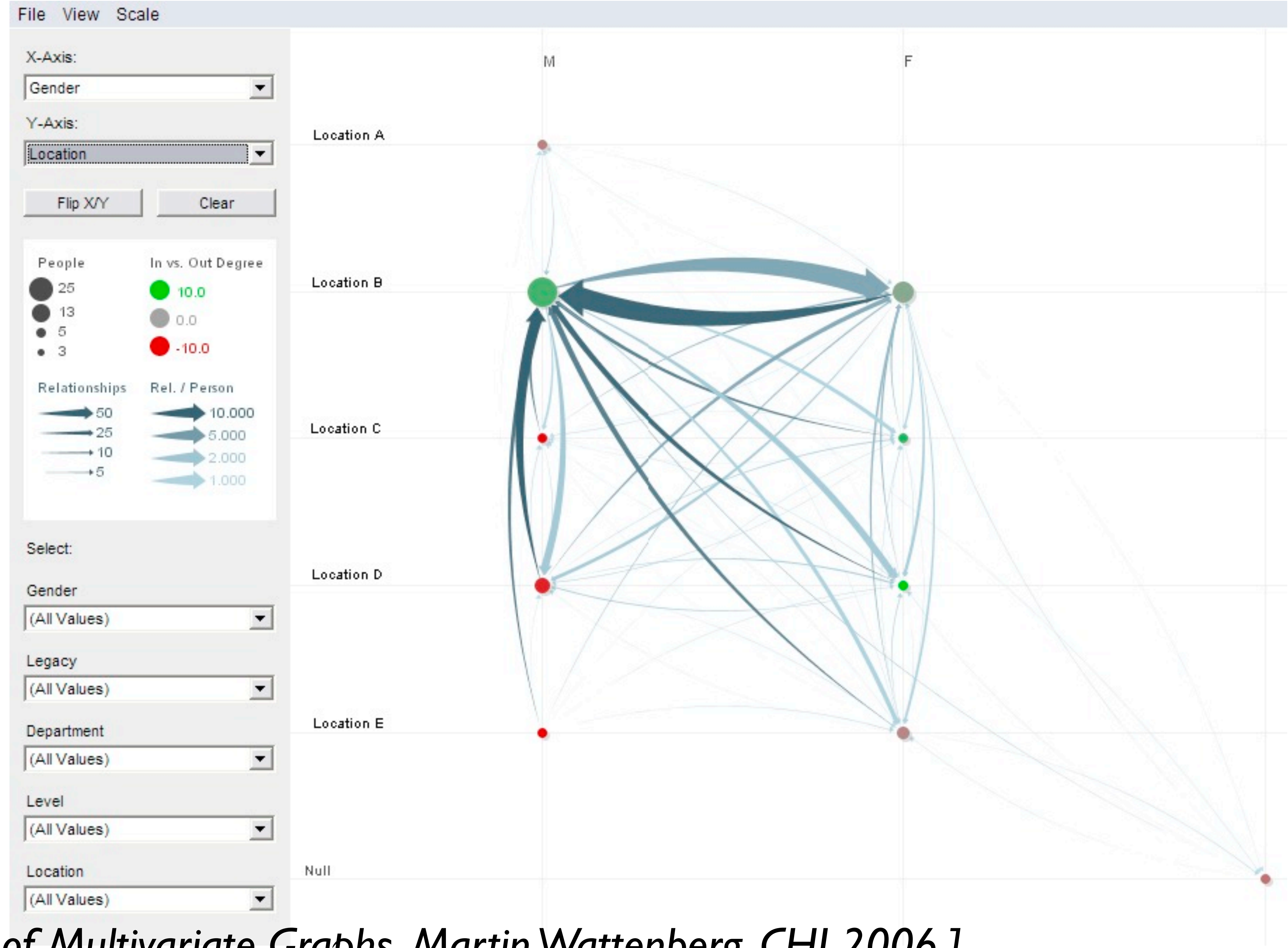






[Visual Exploration of Multivariate Graphs, Martin Wattenberg, CHI 2006.]

PivotGraph



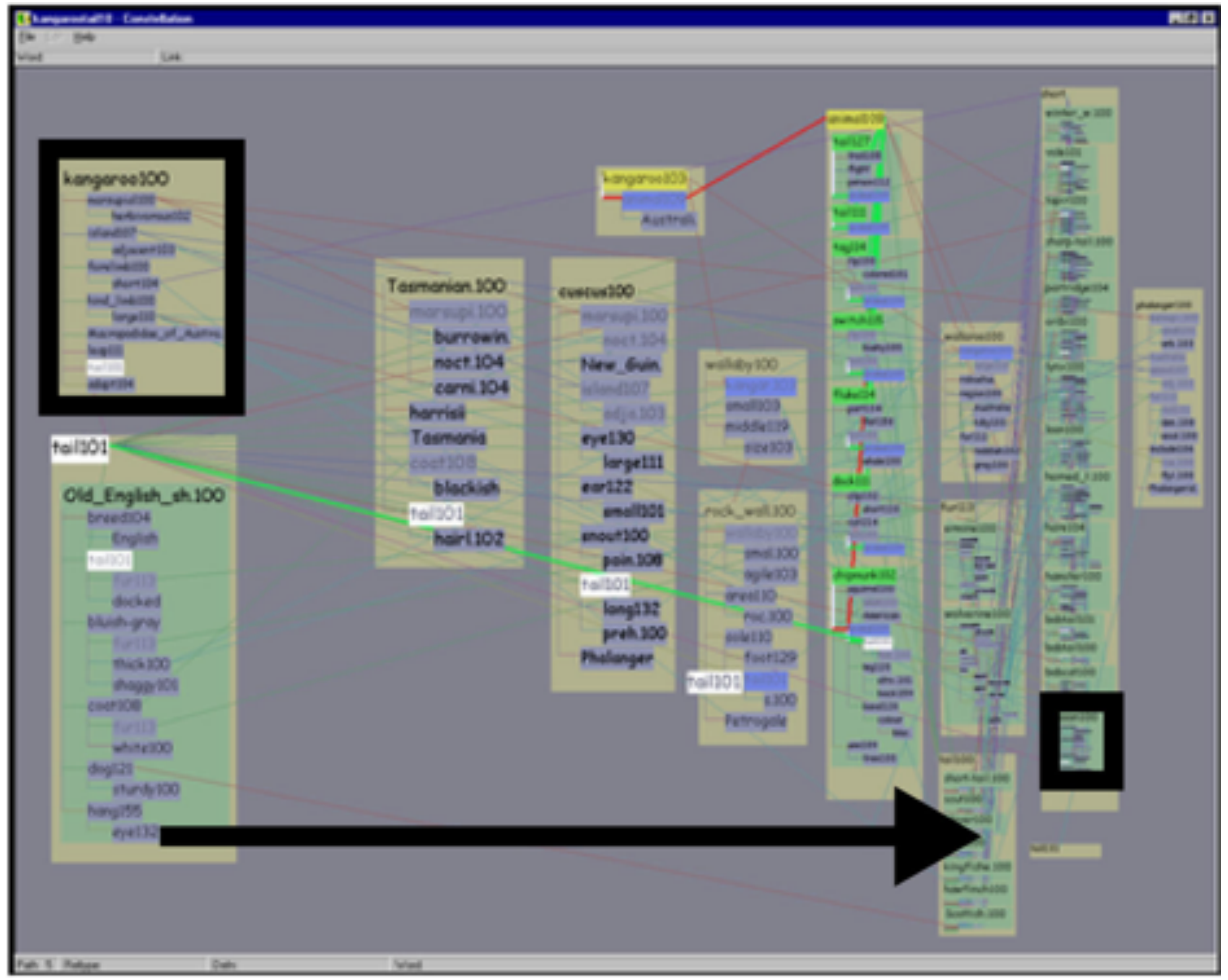
[Visual Exploration of Multivariate Graphs, Martin Wattenberg, CHI 2006.]

PivotGraph Analysis

Idiom	**PivotGraph**
What: Data	Network.
What: Derived	Derived network of aggregate nodes and links by roll-up into two chosen attributes.
Why: Tasks	Cross-attribute comparison of node groups.
How: Encode	Nodes linked with connection marks, size.
How: Manipulate	Change: animated transitions.
How: Reduce	Aggregation, filtering.
Scale	Nodes/links in original network: unlimited. Roll-up attributes: 2. Levels per roll-up attribute: several, up to one dozen.

Analysis example: Constellation

- data
  - –multi-level network
    - node: word
    - link: words used in same dictionary definition
    - subgraph for each definition
      - –not just hierarchical clustering
  - –paths through network
    - query for high-weight paths between 2 nodes
      - –quant attrib: plausibility

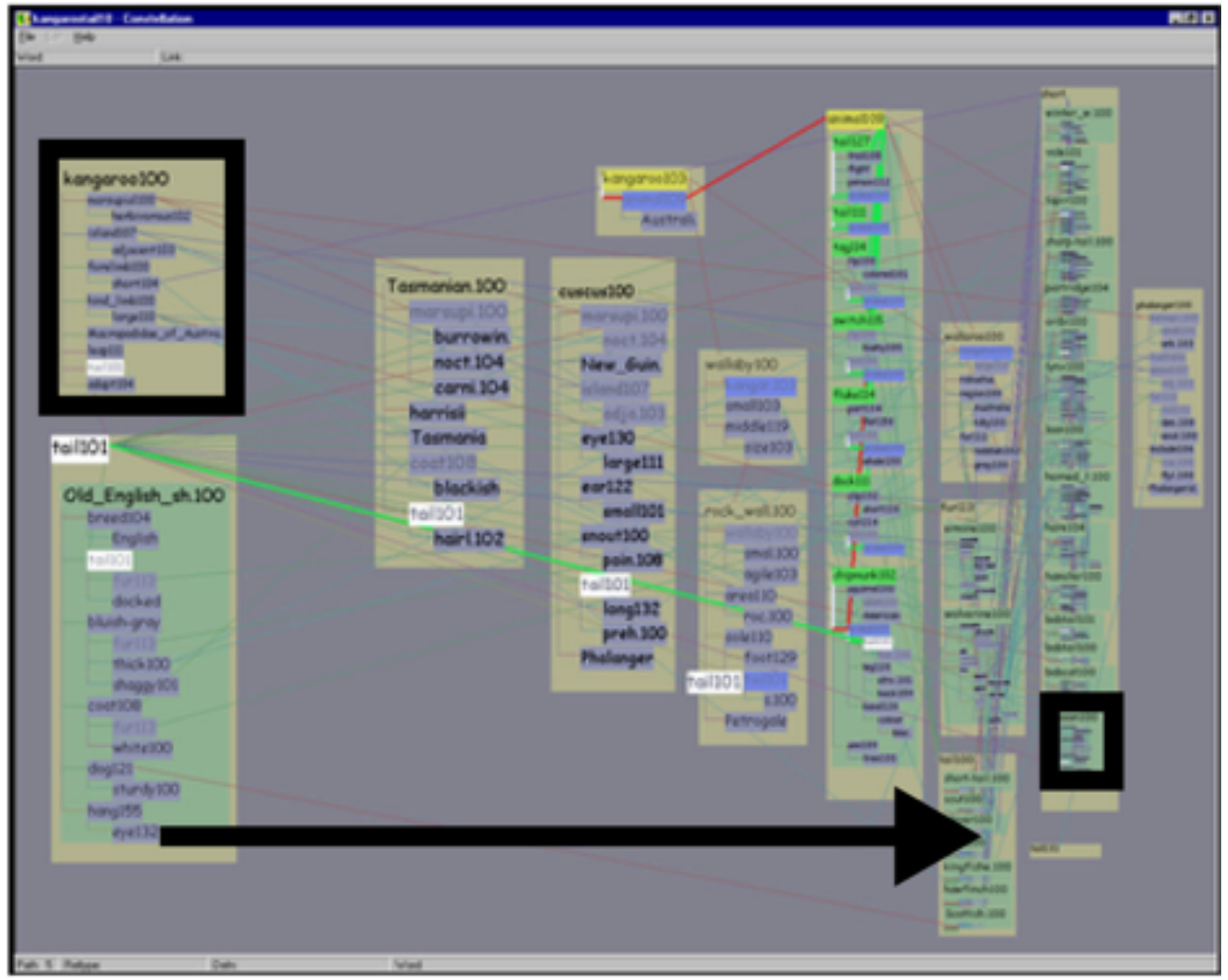


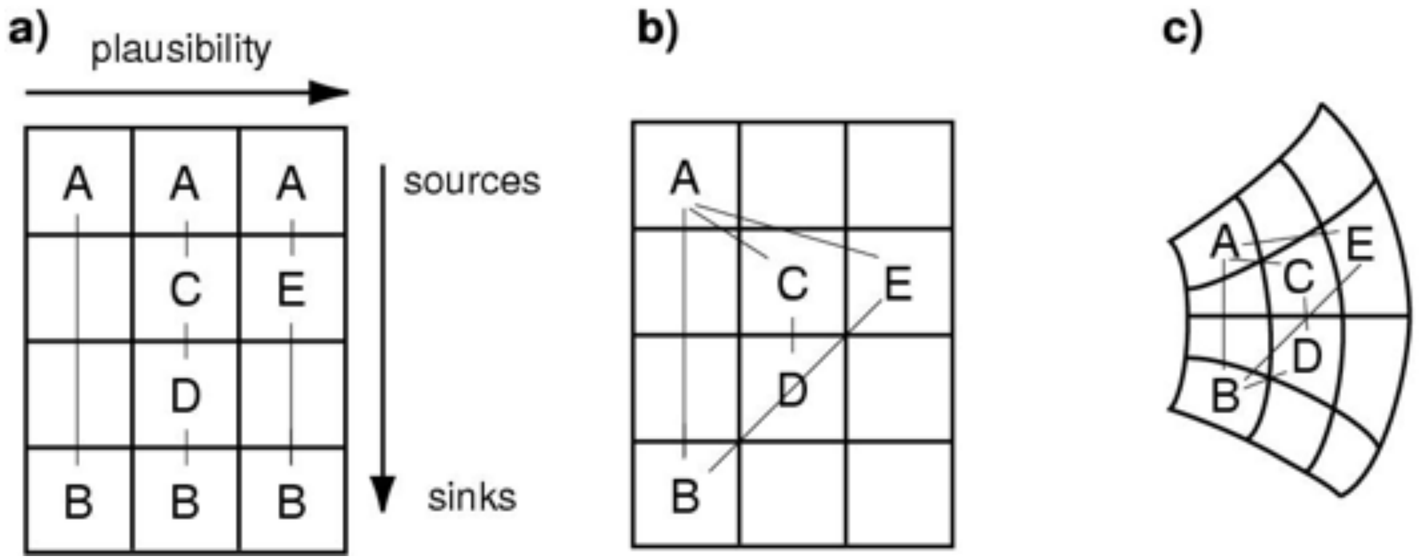
[Interactive Visualization of Large Graphs and Networks. Munzner. Ph.D. Dissertation, Stanford University, June 2000.]

[Constellation: A Visualization Tool For Linguistic Queries from MindNet. Munzner, Guimbretière and Robertson. Proc. IEEE Symp. InfoVis 1999, p.132-135.]

Using space: Constellation

- visual encoding
  - link connection marks between words
  - link containment marks to indicate subgraphs
  - encode plausibility with horiz spatial position
  - encode source/sink for query with vert spatial position
- spatial layout
  - curvilinear grid: more room for longer low-plausibility paths

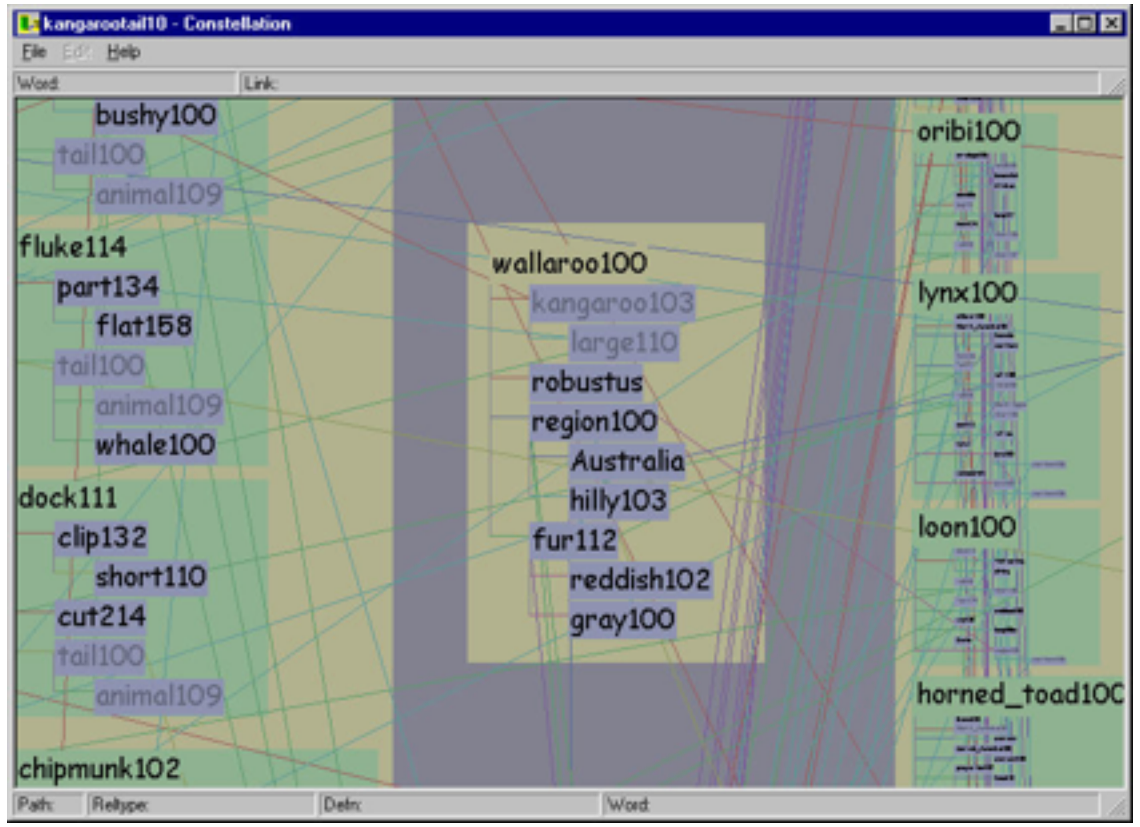


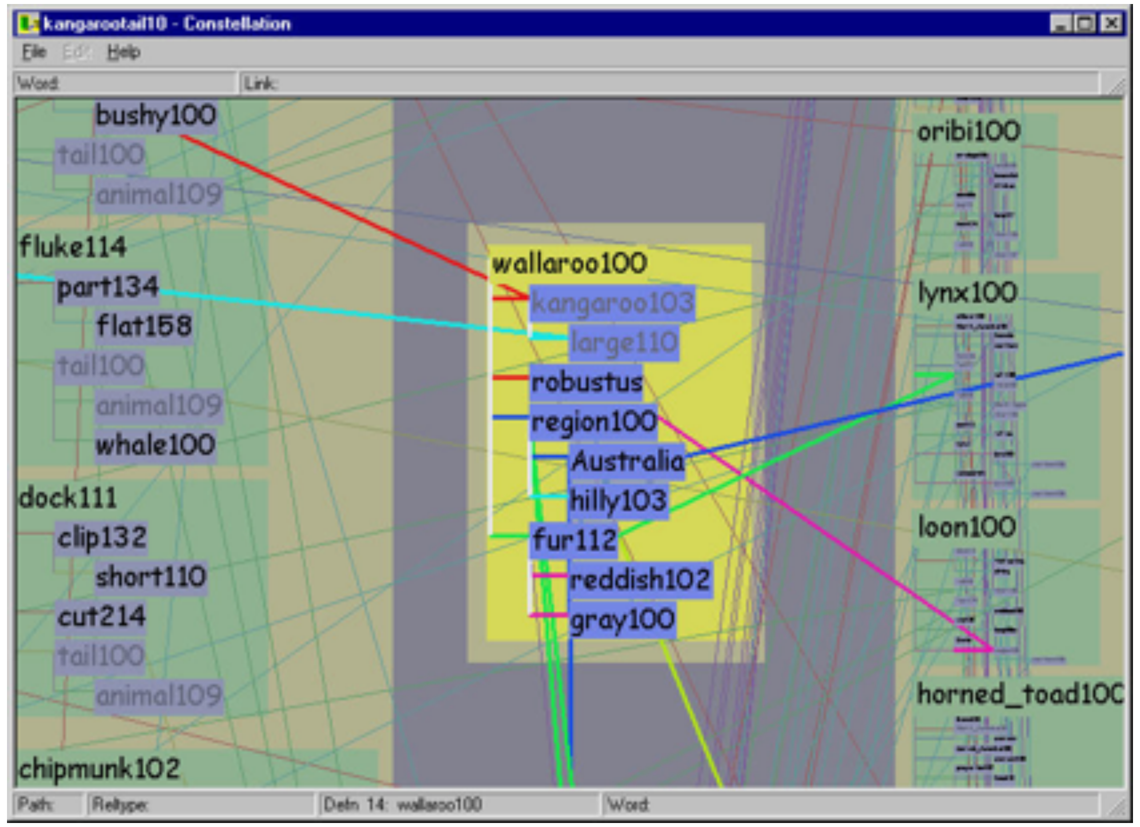


[Interactive Visualization of Large Graphs and Networks. Munzner. Ph.D. Dissertation, Stanford University, June 2000.]

Using space: Constellation

- edge crossings
  - cannot easily minimize instances, since position constrained by spatial encoding
  - instead: minimize perceptual impact
- views: superimposed layers
  - dynamic foreground/background layers on mouseover, using color
  - four kinds of constellations
    - definition, path, link type, word
      - not just 1-hop neighbors





[Interactive Visualization of Large Graphs and Networks. Munzner. Ph.D. Dissertation, Stanford University, June 2000.]

Constellation Analysis

System	Constellation
What: Data	Three-level network of paths, subgraphs (definitions), and nodes (word senses).
Why: Tasks	Discover/verify: browse and locate types of paths, identify and compare.
How: Encode	Containment and connection link marks, horizontal spatial position for plausibility attribute, vertical spatial position for order within path, color links by type.
How: Manipulate	Navigate: semantic zooming. Change: Animated transitions.
How: Reduce	Superimpose dynamic layers.
Scale	Paths: 10–50. Subgraphs: 1–30 per path. Nodes: several thousand.

Analysis

- expected in your paper/topic presentations
  - in addition to content summarization
- expected in your final projects

Paper: D3

- paper types
  - design studies
  - technique/algorithm
  - evaluation
  - model/taxonomy
  - **system**
    - today's emphasis

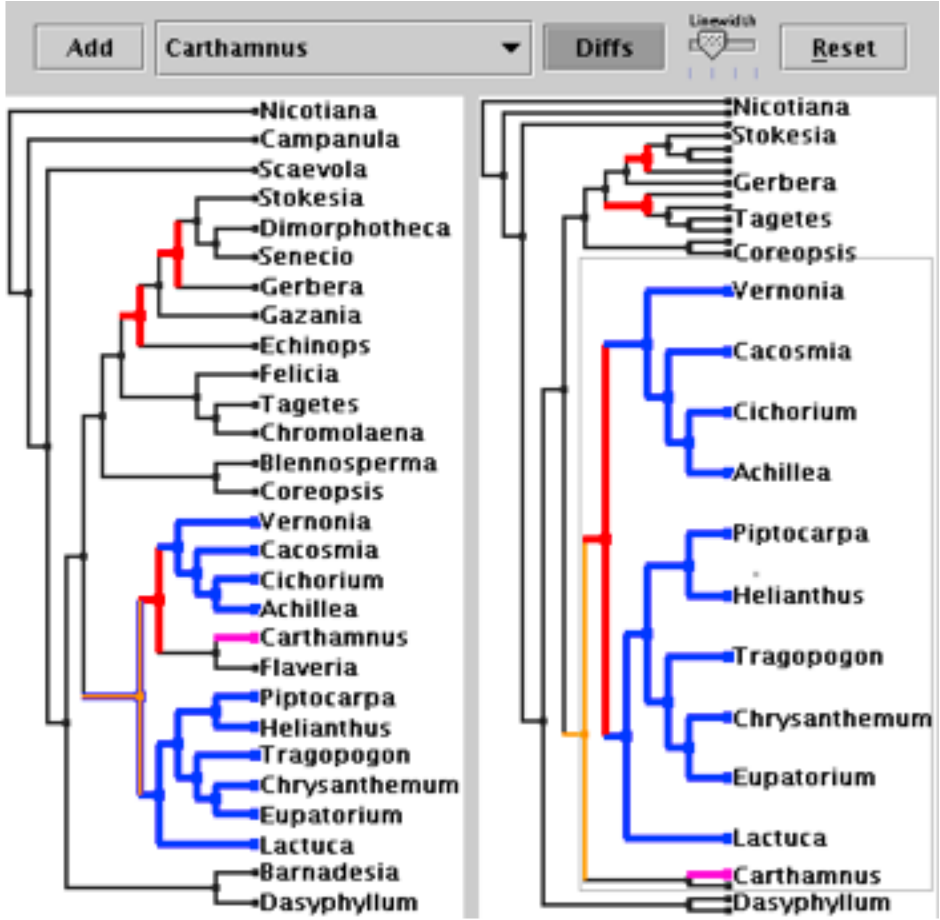
[D3: Data-Driven Documents. Bostock, Ogievetsky, Heer. IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis), 2011.]

Toolkits

- imperative: how
  - low-level rendering: Processing, OpenGL
  - parametrized visual objects: prefuse
    - also flare: prefuse for Flash
- declarative: what
  - Protoviz, D3, ggplot2
  - separation of specification from execution
- considerations
  - expressiveness
    - can I build it?
  - efficiency
    - how long will it take?
  - accessibility
    - do I know how?

OpenGL

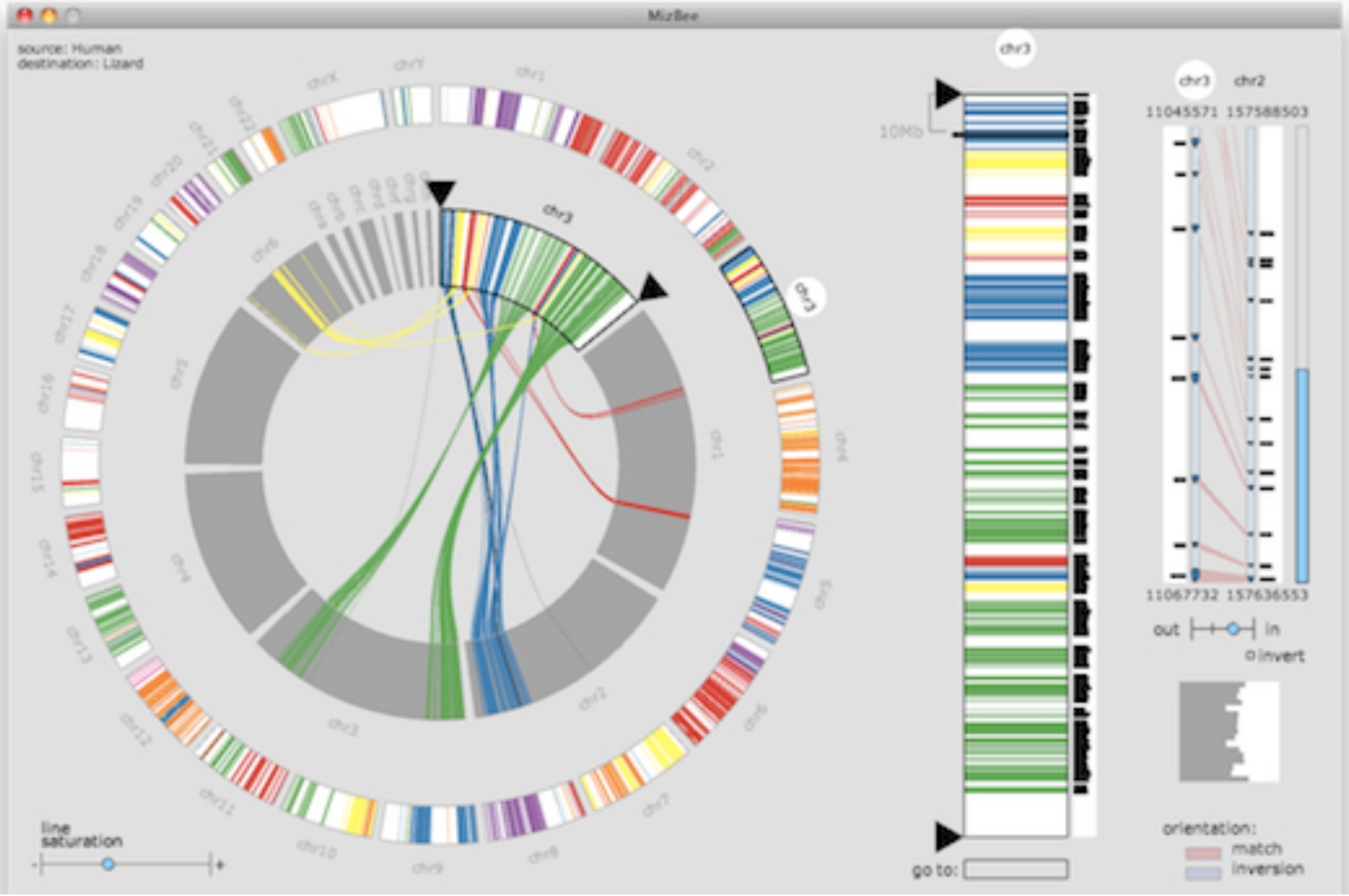
- graphics library
  - pros
    - power and flexibility, complete control for graphics
    - hardware acceleration
    - many language bindings: C, C++, Java (w/ JOGL)
  - cons
    - big learning curve if you don't know already
    - no vis support, must roll your own everything
  - example app: TreeJuxtaposer



[Fig 5. Munzner et al. TreeJuxtaposer: Scalable Tree Comparison using Focus+Context with Guaranteed Visibility. Proc SIGGRAPH 2003, pp 453-462.]

Processing

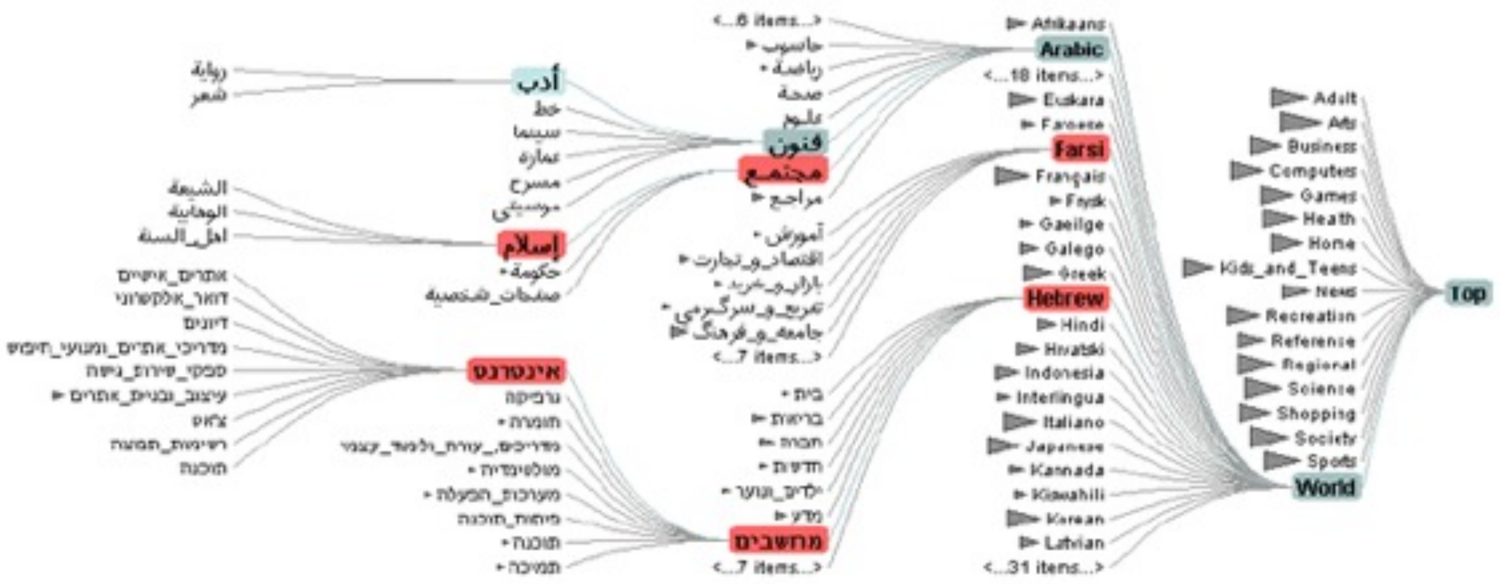
- layer on top of Java/OpenGL
- visualization esp. for artists/designers
- pros
  - great sandbox for rapid prototyping
  - huge user community, great documentation
- cons
  - poor widget library support
- example app: MizBee



[Fig 1. Meyer et al. MizBee: A Multiscale Synteny Browser. Proc. InfoVis 2009.]

prefuse

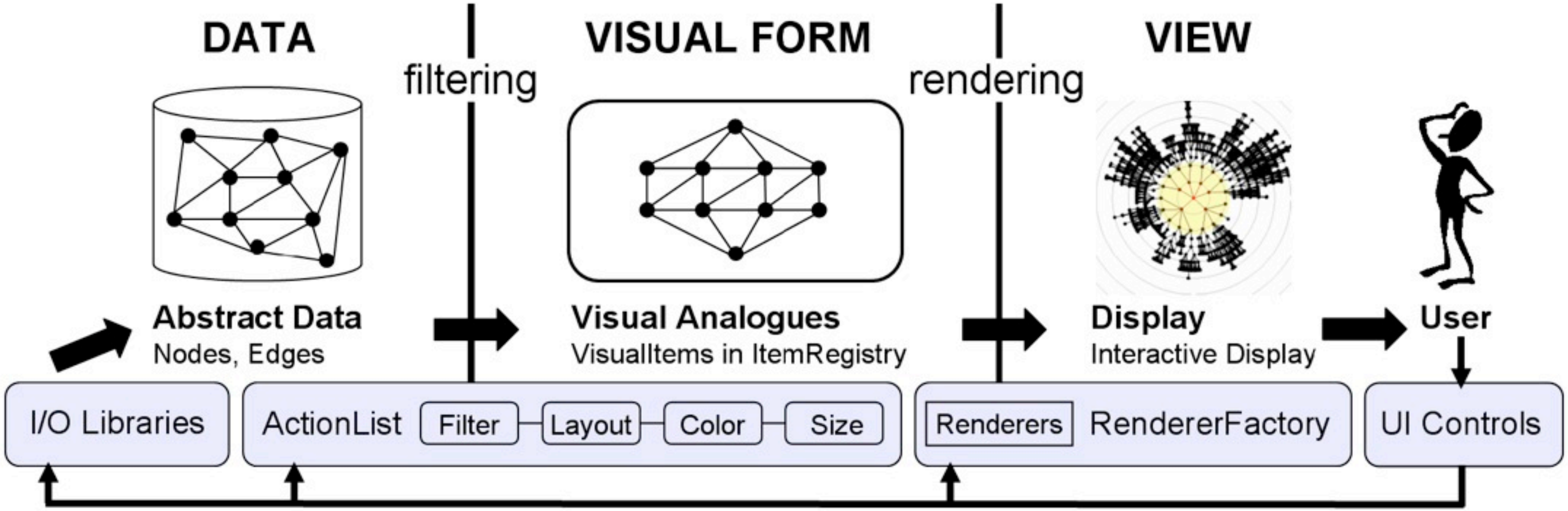
- infovis toolkit, in Java
- fine-grained building blocks for tailored visualizations
- pros
  - heavily used (previously)
  - very powerful abstractions
  - quickly implement most techniques covered so far
- cons
  - hasn't been under active development for
  - nontrivial learning curve
- example app: DOITrees Revisited



[DOITrees Revisited: Scalable, Space-Constrained Visualization of Hierarchical Data. Heer and Card. Proc. Advanced Visual Interfaces (AVI), pp. 421–424, 2004.]

prefuse

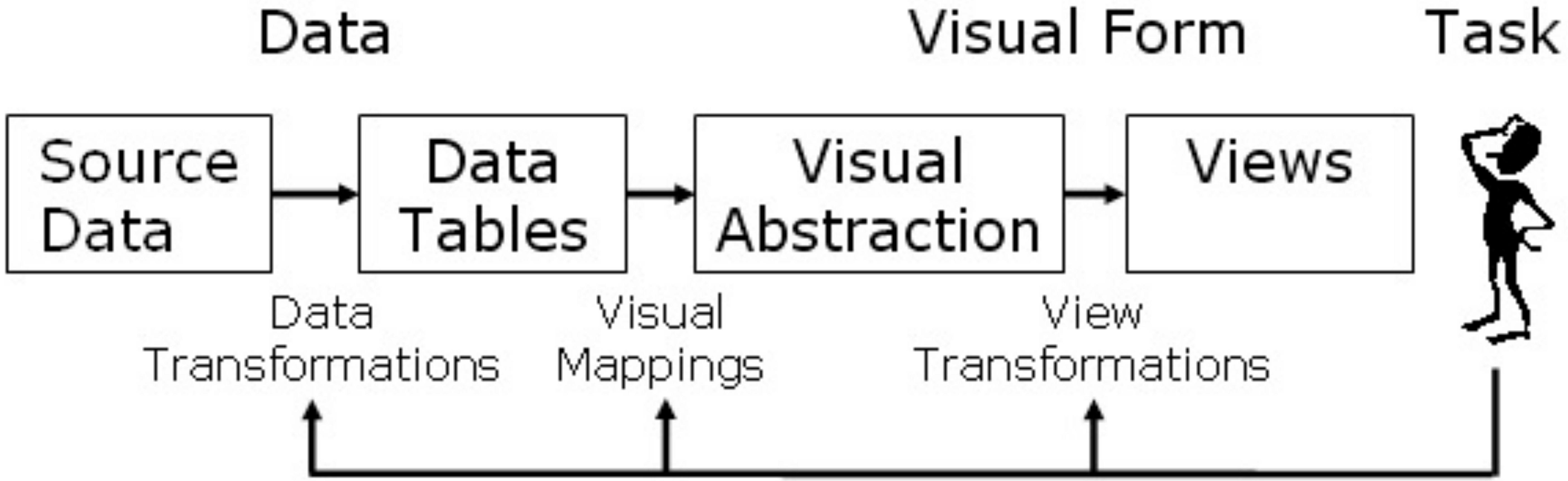
- separation: abstract data, visual form, view
  - data: tables, networks
  - visual form: layout, color, size, ...
  - view: multiple renderers



[Fig 2. Heer, Card, and Landay. Prefuse: A Toolkit for Interactive Information Visualization. Proc. CHI 2005, 421-430]

InfoVis Reference Model

- conceptual model underneath design of prefuse and many other toolkits
- heavily influenced much of infovis (including nested model)
  - aka infovis pipeline, data state model



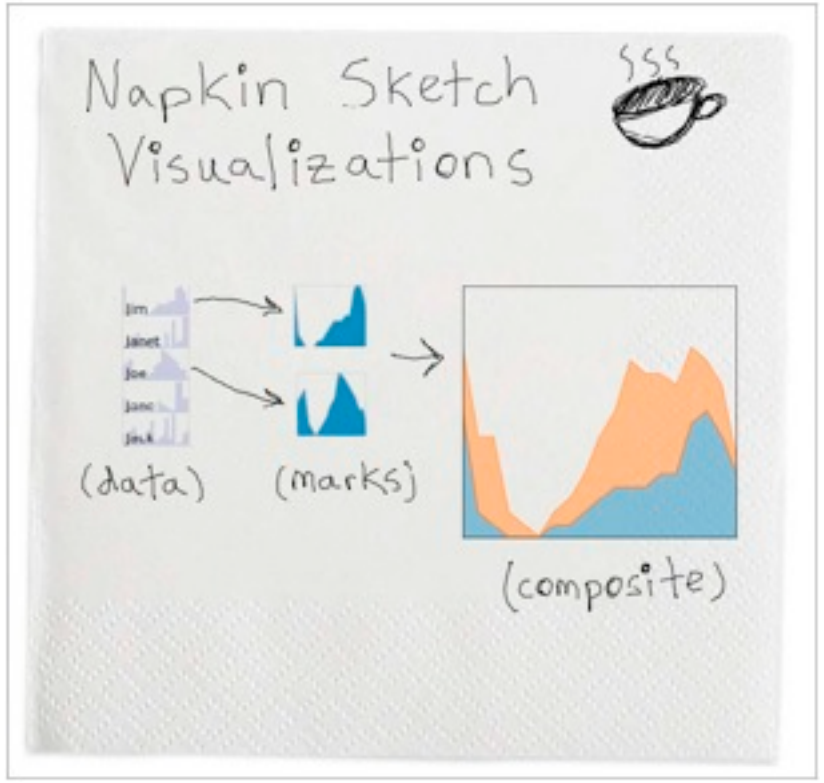
[Redrawn Fig 1.23. Card, Mackinlay, and Shneiderman. Readings in Information Visualization: Using Vision To Think, Chapter 1. Morgan Kaufmann, 1999.]

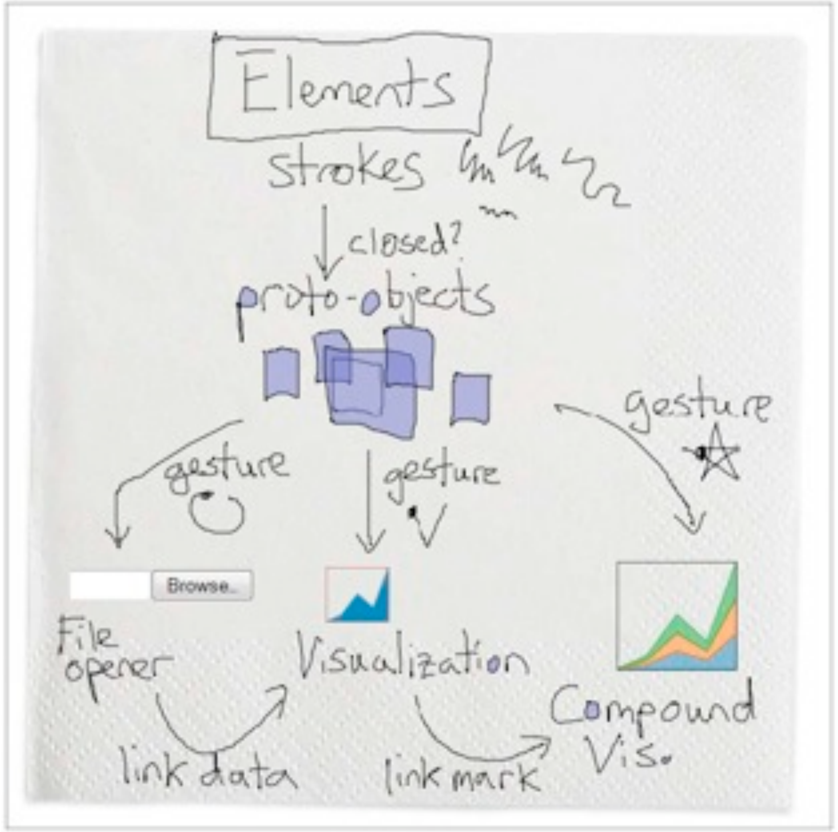
Declarative toolkits

- imperative tools/libraries
  - say exactly **how** to do it
  - familiar programming model
    - OpenGL, prefuse, ...
- declarative: other possibility
  - just say **what** to do
  - Protovis, D3

Protovis

- declarative infovis toolkit, in Javascript
  - also later Java version
- marks with inherited properties
- pros
  - runs in browser
  - matches mark/channel mental model
  - also much more: interaction, geospatial, trees,...
- cons
  - not all kinds of operations supported
- example app: NapkinVis (2009 course project)





[Fig 1, 3. Chao. NapkinVis. http://www.cs.ubc.ca/~tmm/courses/533-09/projects.html#will]

Protovis Validation

- wide set of old/new app examples
  - expressiveness, effectiveness, scalability
  - accessibility
- analysis with cognitive dimensions of notation
  - closeness of mapping, hidden dependencies
  - role-expressiveness visibility, consistency
  - viscosity, diffuseness, abstraction
  - hard mental operations

[Cognitive dimensions of notations. Green (1989). In A. Sutcliffe and L. Macaulay (Eds.) People and Computers V. Cambridge, UK: Cambridge University Press, pp 443-460.]

D3

- declarative infovis toolkit, in Javascript
- Protovis meets Document Object Model
- pros
  - seamless interoperability with Web
  - explicit transforms of scene with dependency info
  - massive user community, many thirdparty apps/libraries on top of it, lots of docs
- cons
  - even more different from traditional programming model
- example apps: many

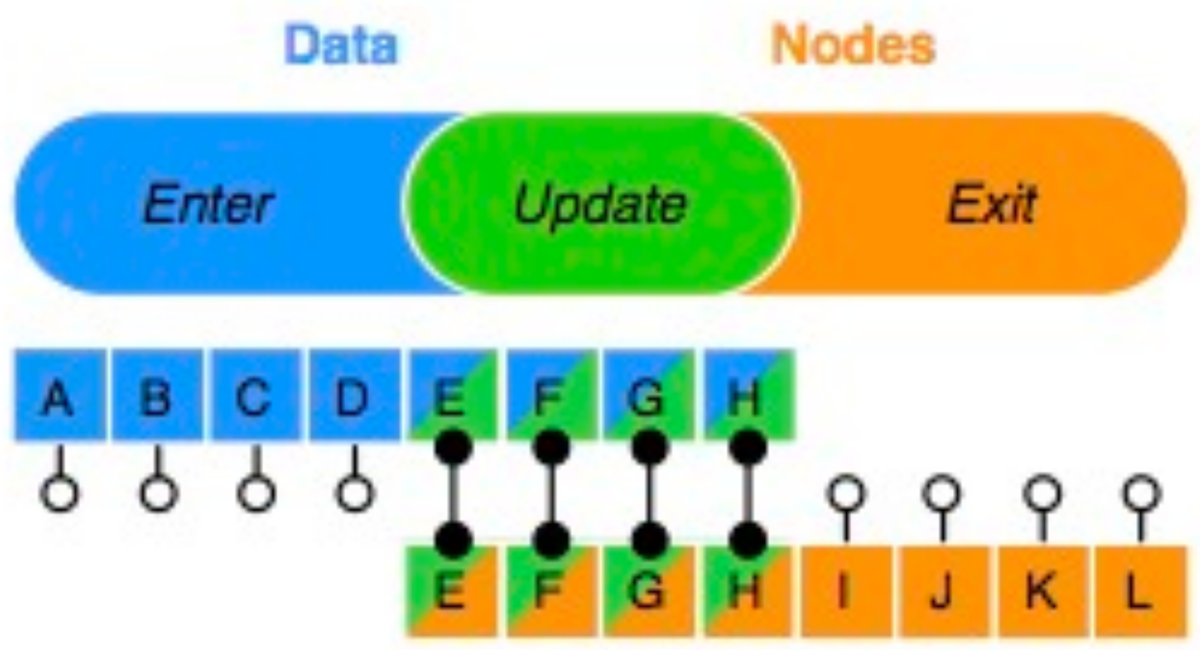
D3

- objectives
  - compatibility
  - debugging
  - performance
- related work typology
  - document transformers
  - graphics libraries
  - infovis systems
    - general note: all related work sections are a mini-taxonomy!

[D3: Data-Driven Documents. Bostock, Ogievetsky, Heer. IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis), 2011.]

D3 capabilities

- query-driven selection
  - selection: filtered set of elements queries from the current doc
    - also partitioning/grouping!
  - operators act on selections to modify content
    - instantaneous or via animated transitions with attribute/style interpolators
    - event handlers for interaction
- data binding to scenegraph elements
  - data joins bind input data to elements
  - enter, update, exit subselections
  - sticky: available for subsequent re-selection
  - sort, filter



[D3: Data-Driven Documents. Bostock, Ogievetsky, Heer. IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis), 2011.]

D3 Features

- document transformation as atomic operation
  - scene changes vs representation of scenes themselves
- immediate property evaluation semantics
  - avoid confusing consequences of delayed evaluation

- validation
  - performance benchmarks
    - page loads, frame rate
  - accessibility
    - everybody has voted with their feet by now!