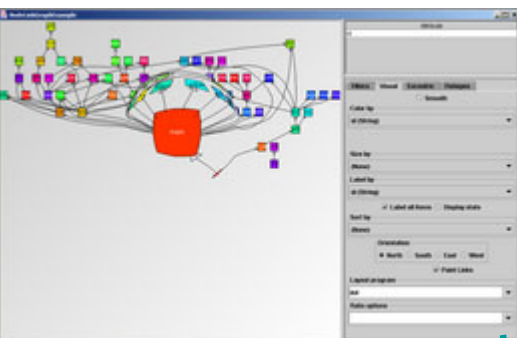


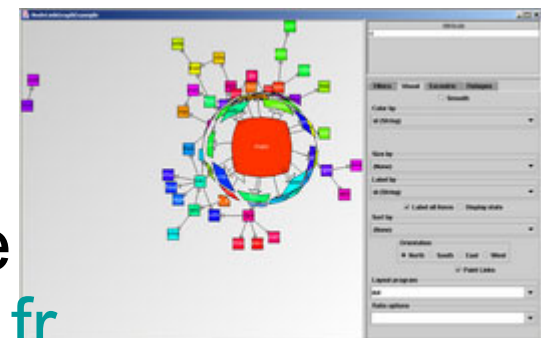
# The Infovis Toolkit



Jean-Daniel Fekete  
 INRIA Futurs/LRI, France

[Jean-Daniel.Fekete@inria.fr](mailto:Jean-Daniel.Fekete@inria.fr)

<http://www.lri.fr/~fekete/InfovisToolkit>



# The Future of Infovis

(Inspired by S. Greenberg on Enhancing Creativity)

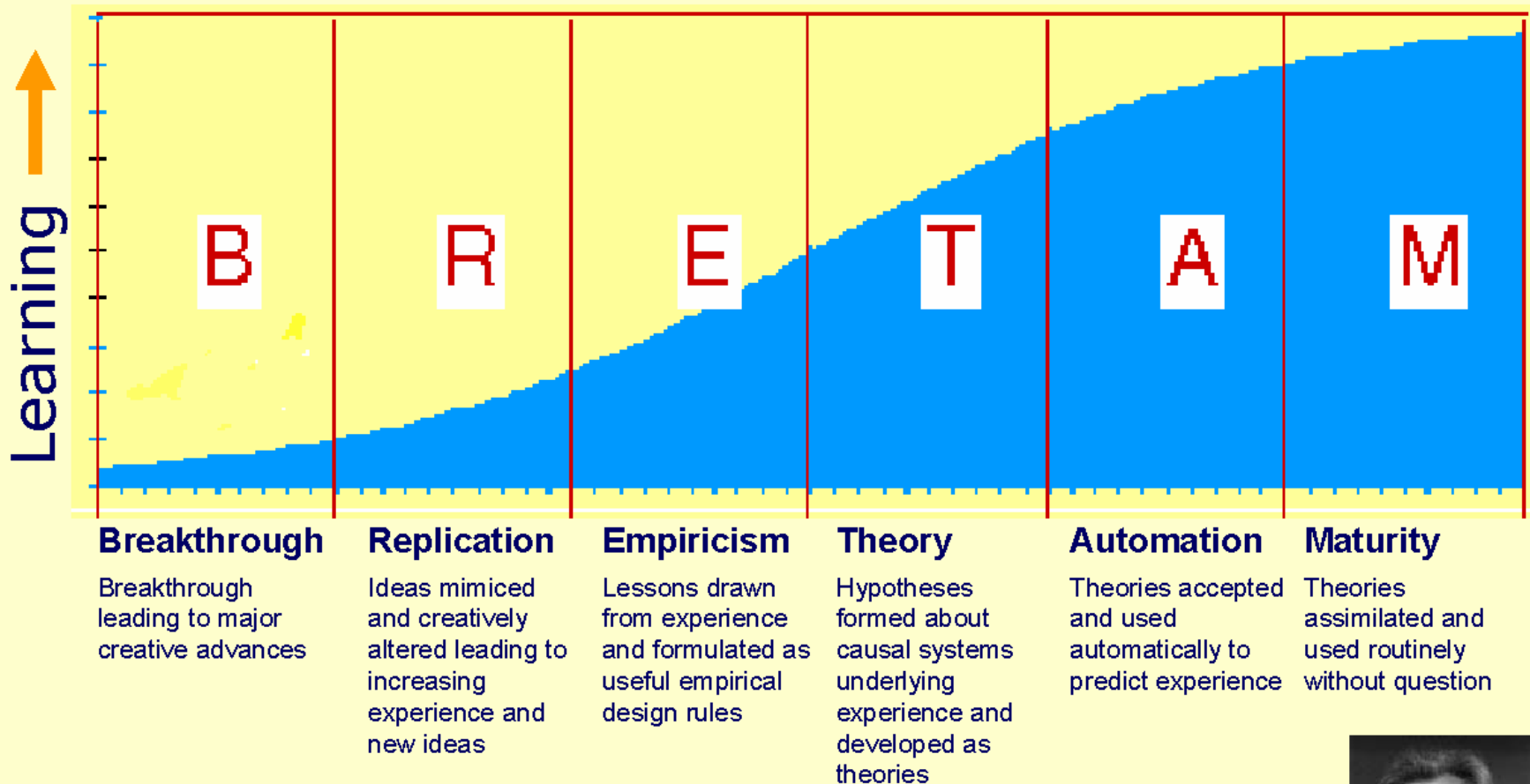
- Infovis is a live domain in research
- It needs to spread
- Creating new visualization techniques and applying them is hard and long
- We need new tools so that:
  - simple ideas become simple to do
  - innovative concepts become possible
  - Infovis will evolve

# Path to Evolution

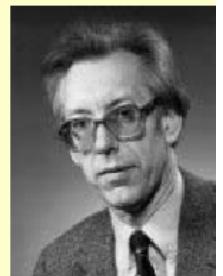
- Is the lack of tools the real bottleneck?
- In the 80's, Brian Gaines introduced a model on how science technology develops over time
- 6 Stages: Breakthrough, Replication, Empiricism, Theory, Automation, Maturity

# Brian Gaines's Model

Time →



Brian Gaines



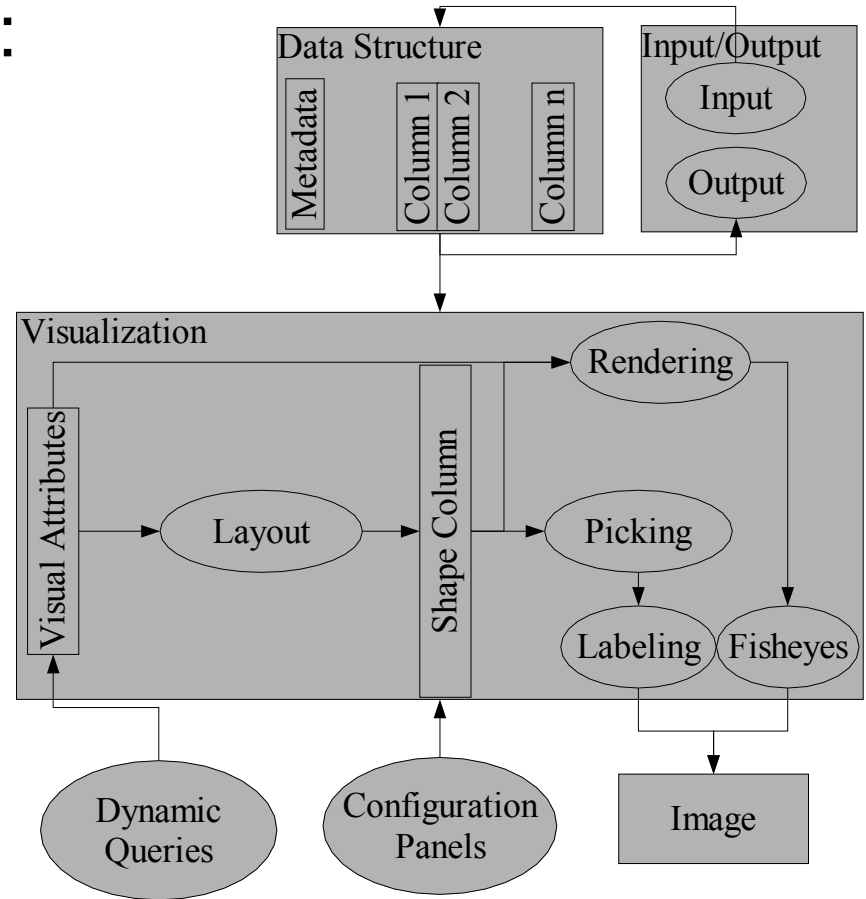
# A Java Toolkit for InfoVis

## Experimentation and Deployment

- The InfoVis Toolkit Enables Quick Replication
- The InfoVis Toolkit already contains:
  - A rich and extensible set of visualizations (time series, scatter plots, treemaps, node-link diagrams for trees and graphs, etc.)
  - Range sliders, standard control panels, Excentric Labeling, Fisheye Lenses
  - Readers and Writers for several popular file formats
  - Optionally an optimized rendering engine relying on OpenGL
- Focus on innovation:
  - Build new visualization techniques
  - Build new interaction techniques
  - Build new widgets for filtering
  - Rely on the InfoVis Toolkit for everything else!

# Architecture of InfoVis

- Separation of concerns:
- Data Structures
  - Table, Tree, Graph
- Visualizations
  - Layered
- Dynamic Queries
  - Generic
- Input/Output



# Data Structures

- In-memory database for speed
- Column-oriented, not Tuple oriented
  - Record = row over all the columns
  - Columns are homogeneous
    - IntColumn, StringColumn, ObjectColumn, BooleanColumn
  - Columns can be added at will in containers (new attributes added)
  - Columns can have missing value, implement notification
- Fast, low memory footprint, extensible
- Table (list of Columns), Tree (Table+topology columns), Graph (2 tables, VertexTable + LinkTable + topology columns)

# Visualizations

- Manage 1 table
- Can be stacked
  - e.g. Node Visualization over Link Visualization for Node-Link Diagrams
  - Excentric Labels Visualization on top
- Manage Fisheye in a generic way
- Specialized by data structures and visualization techniques
- Manage “Visual Attributes” connected to Columns
  - Size, Color, Alpha, X/Y position for some
  - Generalized to Selection, Filtering, Sorting order, DOI, etc.

# Connecting Table to Visualization

Table

Column1 (name)

Column2 (date)

Column3 (length)

Column4 (type)

Visualization

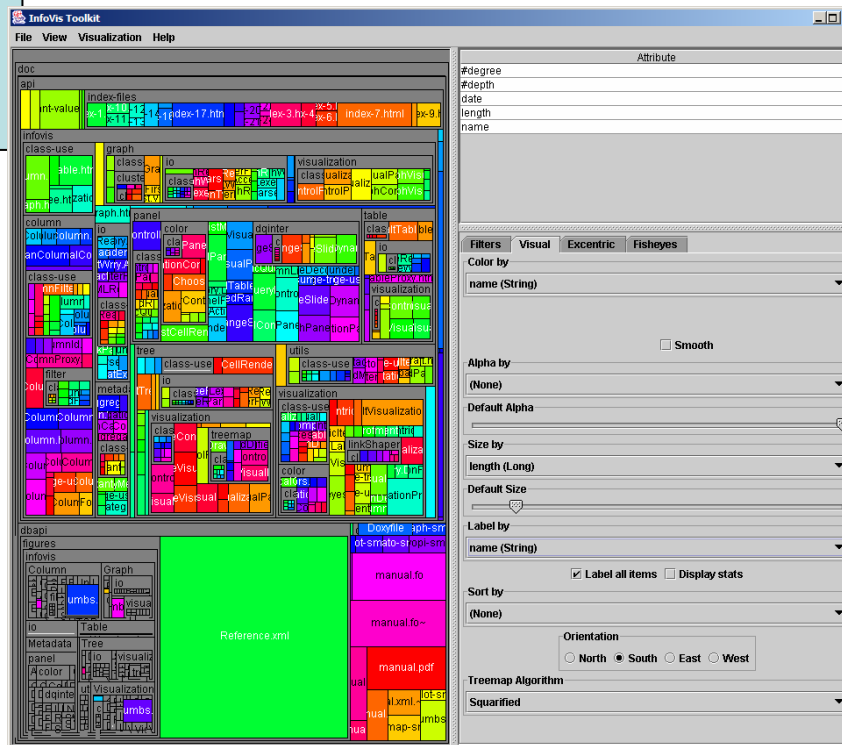
Visual Attribute1 (size)

Visual Attribute2 (color)

Visual Attribute3 (alpha)

Visual Attribute4 (label)

Visual Attribute5 (selection)



# Connecting Visualizations

Table

Column1 (name)

Column2 (date)

Column3 (length)

Column4 (type)

Visualization1

Visual Attribute1 (size)

Visual Attribute2 (color)

Visual Attribute3 (alpha)

Visual Attribute4 (label)

Visual Attribute5  
(selection)

Visualization2

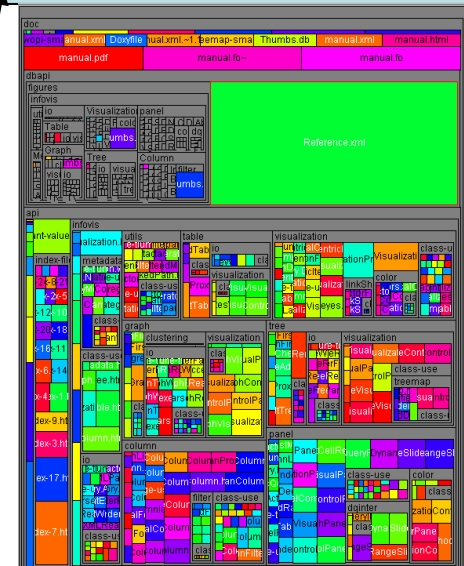
Visual Attribute1 (size)

Visual Attribute2 (color)

Visual Attribute3 (alpha)

Visual Attribute4 (label)

Visual Attribute5  
(selection)



# Demos

- Visualizations
  - + Excentric Labels
  - + Fisheyes
- Filtering
- Extensions
  - Links on Treemaps
  - Image on Treemaps

# Who should use it?

- Professors for teaching with a working support
- Students for creating innovative projects
- Practitioners for integrating InfoVis techniques to applications
- Researchers for focusing on novel techniques without re-creating everything
- The Toolkit is Open Source: Contribute!