The InfoVis Toolkit

Jean-Daniel Fekete
INRIA Futurs/LRI
Bat. 490
Université Paris-Sud
F91405 Orsay Cedex, France
Tel: 1-33-1-69-15-66-23
E-mail: Jean-Daniel.Fekete@inria.fr

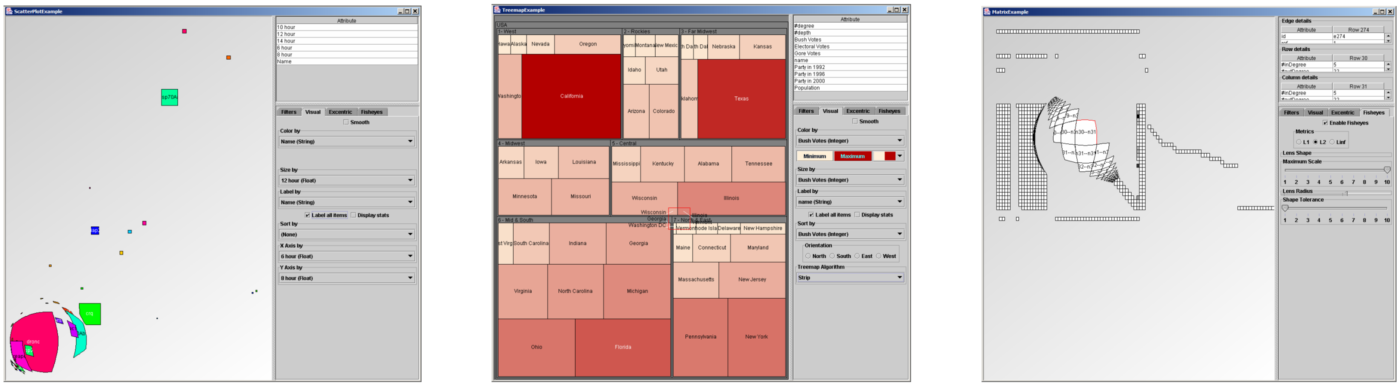

Figure 1: Examples of Scatter Plot, Treemap and Graph Visualizations Built with the InfoVis Toolkit

ABSTRACT

The *InfoVis Toolkit* is designed to support the creation, extension and integration of advanced 2D Information Visualization components into interactive Java Swing applications. The InfoVis Toolkit provides specific data structures to achieve a fast action/feedback loop required by dynamic queries. It comes with a large set of components such as range sliders and tailored control panels to control and configure the visualizations. Supported data structures currently include tables, trees and graphs. Supported visualizations include scatter plots, time series, Treemaps, node-link diagrams for trees and graphs and adjacency matrix for graphs. All visualizations can use fisheye lenses and dynamic labeling. The InfoVis Toolkit supports hardware acceleration when used with Agile2D, an OpenGL-based implementation of the Java Graphics API resulting in speedup factors of 10 to 200.

KEYWORDS: Information Visualization, Toolkit, Graphics.

INTRODUCTION

Despite their well understood potentials, information visualization applications are difficult to implement. They require a set of components and mechanisms that are not available in or not well supported by traditional GUI toolkits such as range sliders, fisheye lenses and dynamic queries.

The InfoVis Toolkit has been designed to quickly specialize existing information visualization techniques to specific applications, to design and test new visualization techniques and to experiment with new uses of visual attributes such as transparency and color gradients [2]. The *InfoVis Toolkit* key features are:

- Generic data structures suited to visualization;
- Specific algorithms to visualize these data structures;
- Mechanisms and components to perform direct manipulations on the visualizations;
- Mechanisms and components to perform well-known generic information visualization tasks;
- Components to perform labeling and spatial deformation.

STRUCTURE OF THE INFOVIS TOOLKIT

The InfoVis Toolkit is a Java library and software architecture organized in five main parts (Figure 2): tables, columns, visualizations, components and input/ output. It brings together several ideas from different domains and assembles them in a consistent framework, similar to [1,2] but using the Java/Swing libraries instead of C++/OpenGL which are more difficult to learn and to use.

The InfoVis toolkit provides a *unified underlying data structure* based on tables. Representing data structures with tables improves the memory footprint and performance, compared with ad-hoc data structures used by other specialized InfoVis applications. Any data structure can easily be implemented on top of tables and accessed using an object-oriented interface for ease of programming.

A table is a list of named columns plus metadata and user data. A column manages rows of elements of homogeneous type, i.e. integers, floating points or strings. The elements are indexed so columns are usually implemented with primitive arrays. Some rows can be undefined. This mechanism is important because in real data sets, values may be missing. Allowing undefined elements is also very useful for representing general data structures such as XML elements with attributes.

Columns also support the following features:

- they contain metadata, e.g. to express that an integer column contains categorical or numeral values;
- they can trigger notifications when their content is modified;
- they support formatting for input and output so, for example, dates can be stored in columns of “long integers” data types and still appear as dates when read or displayed.

Layout algorithms are encapsulated into *Visualization* components that map data structures into visual shapes. Visualizations natively support dynamic labeling [3] and fisheye views.

The InfoVis Toolkit currently supports three concrete data structures: tables, trees and graphs. For each data structure, it supports several visualizations: time series and scatter plots for tables, node-link diagrams and treemaps for trees, node-link diagrams and adjacency matrices for graphs.

Creating a new visualization technique such as the Icicle Tree (Figure 3a) requires 50 lines of Java code. Adding direct manipulation to Icicle trees for interactive clustering requires 18 additional lines of Java. Dynamic queries, dynamic labeling and fisheye views are immediately operational on this new visualization. Yet, all interactions can be tailored. Visualizations such as the Icicle tree can easily be used as a component, e.g. for controlling the clustering and permutations of a graph visualized as a matrix (Figure 3b).

The InfoVis Toolkit is distributed as free software under a liberal license (QPL) in the hope that the Information Visualization community will adopt it as a workbench for implementing new ideas within an already rich toolkit. It is available at: http://www.lri.fr/~fekete/InfovisToolkit and is currently used by several research projects in domains including biology, cartography and trace analysis. It has also proved very efficient for student projects, both in terms of development time and shared experience.

We are continuing the development of the InfoVis Toolkit and are looking forward to improvements and feedback from the Information visualization community.

REFERENCES

1. Bosch, R:, Stolte, C., Tang, D. Gerth, J. Rosenblum, M. and Hanrahan; P., Rivet: A Flexible Environment for Computer Systems Visualization, Computer Graphics 34(1), February 2000, pp. 68 – 73.
2. Fekete, J.-D. and Plaisant, C. Interactive Information Visualization of a Million Items Proceedings of IEEE Symposium on Information Visualization, 2002, Boston, October 2002, pp 117 -124.
3. Fekete, J.-D., and Plaisant, C. Excentric labeling: Dynamic neighborhood labeling for data visualization. In Proc. of CHI '9, ACM Press, May 1999, pp. 512-519

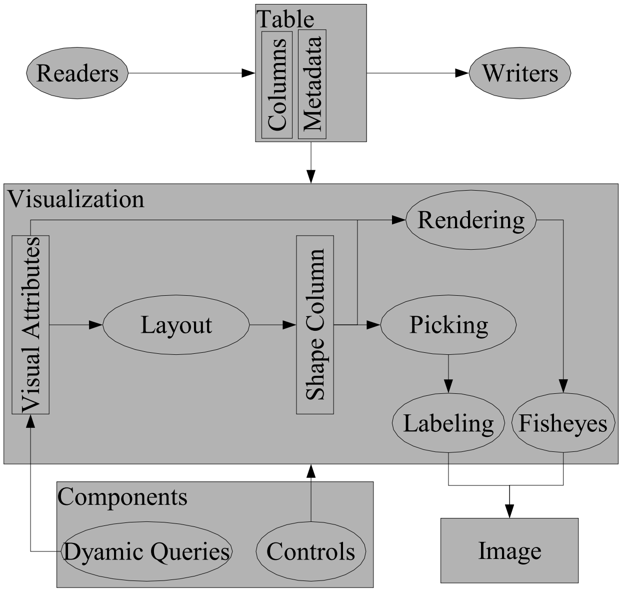

Figure 2: Internal structure of the InfoVis Toolkit. Squares represent data structures whereas ellipses represent functions.

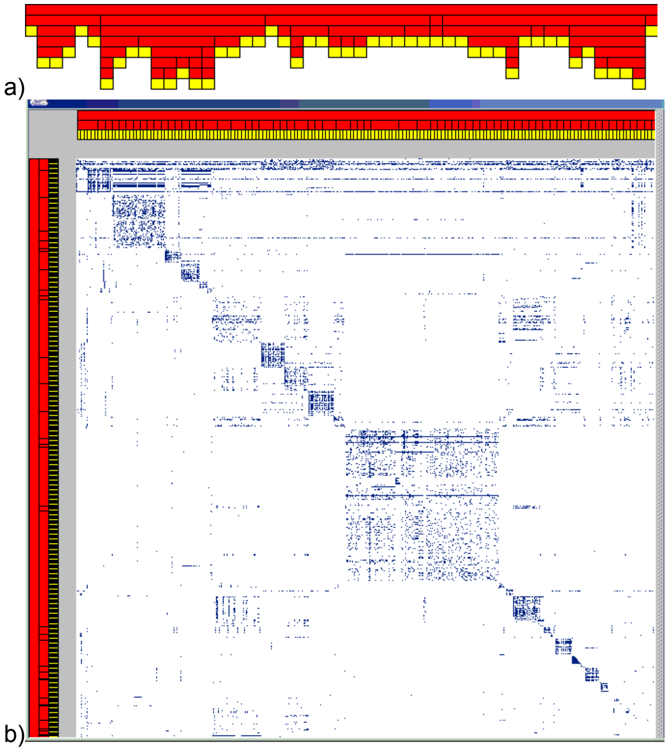

Figure 3: a) An irregular Icicle trees, b) Icicle trees as components for a clustered graphs showing a web site with 600 documents..