Information Visualization Assignment

Societies continued reliance on information and communications technologies has resulted in organizations generating, gathering, and storing "raw data" at a growing rate. While much of the data under consideration is in the form of multi-dimensional or unstructured data some of it can be considered relational information.

Relational information is typically modeled in terms of a graph G where the atomic entities of the domain form the set of nodes N and the interrelationships form the set of edges E. This graph can be drawn using different paradigm which include, graph embeddings as a series of icons and lines (node-link) or as a two/three dimensional structure plot (matrix view, where a highlighted cell indicates an edge between the two nodes) or a city-plot view (where the strength of the relationship is extrapolated into the third dimension to produce a landscape view). The general problem is that the nodes and edges have no inherent geometry

Assignment 1

- The major part of this assignment is to implement the basic force-directed or spring-algorithm in 2D using OpenGl or VTK.
- Using your system provide drawings of: 1138 BUS, BFW398A, DWB512 and GRE512, WEST1505 from: http://math.nist.gov/MatrixMarket/matrices.html you can use the matrix read utilities from the Matrix Market website.

- Next, select one of the following options and implement it allowing the user to use the feature through an application flag or GUI selection.

1. Instead of randomly positioning the nodes to give the nodes their first geometric positions, implement an improved initial positioning algorithm (such as the wave front layout)
2. Implement the Kamada-Kawai variant of the force-directed algorithm.
3. Implement a magnetic spring variant of the force-directed algorithm.
4. Implement a snap-to-grid or snap-to-circular-grid location method
5. Implement a clustering method to reduce the number of nodes/edges in the graph to produce a new clustered graph with nodes, clustered nodes and edges.
6. Implement the layout in 3D and add an edge simplification method, which clusters nodes and edges depending on the current viewpoint and the distance to each node and edge.
7. Create a hybrid 2D graph layout, which uses a structure plot for portions of the graph and uses a force-directed layout for the other parts of the graph.
8. Implement a stopping criterion, which measures a graph drawing aesthetic (e.g. crossing, uniform edge length) and uses this to stop the layout algorithm looping.

The spring-algorithm is described in the slides but for more details see: http://www.cs.ucd.ie/staff/aquigley/home/downloads/aquigley-thesis-mar-02.pdf

Dr. Aaron Quigley, 1st March 2006