Theory of Visualization

Curran Kelleher

Abstract

The scope of this work is to develop a comprehensive theoretical model for interactive visualization systems.

1 Scope

We begin with a set of functional requirements for an ideal visualization system:

- Data
  - Data can be published on the Internet by anyone.
  - Data can be consumed from the Internet by anyone.
  - Data has rich metadata describing itself.
  - Data from multiple sources can be automatically integrated properly.
  - Geospatial data is supported (points, lines, polygons).
  - Temporal data is supported (points, periods, absolute and relative).
  - The hierarchical data cube model is supported.
  - OLAP operations are supported: Slide, Dice, Drill Down, Drill Up, Roll-up, Pivot.
  - Arbitrary hierarchical selections can drive data views.
  - The relational model is supported.
  - Data flow networks are supported (including statistics and analysis operations).

  - Very large data sets (¿200 terabytes) are supported.

- Graphics
  - Every visualization contains a set of graphical objects for each represented object.
  - Interactive visualizations can be fully defined using a specification language.
  - Color maps are consistent across visualizations of the same data.
  - Graphics are scalable (for large high resolution displays or printed material).
  - Performance is high (the system never goes under 60 frames per second).

- Interaction
  - Each graphical object can be *selected*.
  - Brushed/linked selection is supported.
  - Selection of one record may trigger selection of related records.
  - Each graphical object can be *probed*.
  - Brushed/linked probing is supported.

- Meta-interaction
  - Traversing the session history graph is supported (undo, redo, go to state).
  - Synchronous collaboration is supported.
  - Asynchronous collaboration is supported.

Based on this set of feature requirements, it becomes clear that we are engaging in building a set of models, one for each component or realm of the system, which can be integrated into one larger model encompassing the entire system. The following system domains have been identified as canditates to be independently modeled:

- Data storage and retrieval

- Data reconstruction and integration
- Data transformation networks
- Color maps
- Normalization schemes
- Interactive graphical objects
- Brushed selection and probing
- Visualization specification
- Application session state
- Application session state transition network