Semi-Automatic Information and Knowledge Systems:
Hierarchical Data Visualization & Ontology Visualization

Monika Lanzenberger

Information Visualization

InfoVis is ...

... the process of transforming data, information, and knowledge into visual form making use of humans’ natural visual capabilities.

... the computer-assisted use of visual processing to gain understanding.

... providing the user with an overview first and then details on demand (<-> text).

... based on pre-attentive features (< 200ms).

Outline

- Information Visualization
- Hierarchical Data Visualization Techniques
- Ontology Visualization
- Alignment Visualization

Information Visualization is ...

... based on pre-attentive features (< 200ms).
InfoVis: Using space

- Visualization of abstract data (e.g., financial transactions, insurance risks, etc.) means to find spatial representations (2D, 3D).

- No inherent spatial structure available, so the designer/user needs to decide which dimensions are represented by space: Mapping.

InfoVis: Kinds of Data

- Entities (e.g., people, terms) and relations (e.g., part-of, is-a)
- Both can have sets of attributes (duration, color, time, etc.)

- Types of attributes
  1. nominal, ordinal, interval, ratio
  2. Category data (nominal), integer data (ordinal), real-number data (interval & ratio)

- High-frequency versus high-structural

Classification

- Slaving
- Linking & Brushing

Linking & Brushing

Coupling views by:

- **Slaving**
  movements in one view are automatically propagated in the other views

- **Linking**
  connects the data items of one view with the data items of the other views e.g., done by **brushing**: user selects and highlights items in one view and the corresponding items are highlighted automatically
Visual Encoding Techniques

Different ways in encoding information visually:

- **Space**
  (See details next slide)

- **Marks (in space)**
  Points, lines, areas, volumes

- **Connections & enclosures**

- **Retinal properties**
  Crispness, shape, resolution, transparency, color, grayscale

- **Temporal changes**

- **Viewpoint transformations**

---

**Composition**
The orthogonal placement of axes, creating a 2D metric space

**Alignment**
The repetition of an axis at a different position in the space

**Folding**
The continuation of an axis in an orthogonal direction
Outline

• Composition
  The orthogonal placement of axes, creating a 2D metric space
• Alignment
  The repetition of an axis at a different position in the space
• Folding
  The continuation of an axis in an orthogonal direction
• Recursion
  The repeated subdivision of space

Hierarchical Techniques

Basic Idea: Visualization of data using a hierarchical partitioning into subspaces

Examples are:

- Dimensional Stacking [LeBlance et al. 1990]
- Worlds-within-Worlds [Feiner & Beshers 1990]
- Treemaps [Shneiderman 1992; Johnson, 1993]
- Cone/Cam Trees [Robertson, Mackinlay, Card 1991]
- Cheops [Beaudoin et al., 1996]
- InfoCube [Rekimoto & Green 1993]
Treemap

Screen-Filling Methods
- Hierarchical partitioning of the screen depending on the attribute values
- Overcoming space limitations

Alternative Partitioning
- x- and y-dim of the screen

Attributes - User-Defined
- for partitioning and their ordering

Color Correspond to Add. Attributes

Overview over
- Large amount of hierarchical data (e.g., file system)
- Data with multiple ordinal/quant. attributes (e.g., census data)

Trees:
- Ordered
- Acyclical
- Hierarchical

Horizontal vs. Vertical

Horizontal
- Corresponding to Text

Vertical
- Traditional

Standard Representations

Figure 5: A Standard 2D Tree.

Figure 6: A Rotated 2D Tree.
**Treemaps: Venn-Diagrams**

**Nested Treemap**

![Nested Treemap Diagram](image)

**Example: File Structure to Tree**

**File System:**
- 3 Folders
- 6 Files

**Root --> whole Screen**

**Example:**
- Cutting - according to the size (30% and 70% of the space)
- Iteration: folder and subfolder

[Shneiderman 1992; Johnson, 1993] ML
Example: File Structure to Tree

File System:

- 3 Folders
- 6 Files

One Solution

Treemap: View Large Trees with Node Values

+ Space filling
+ Color coding
+ Size coding
- Requires learning

Treemaps: Layouts

Figure 1: Top-Down, Size by Weight
Figure 2: Slice-and-Dice, Size by Unit
Figure 3: Slice-and-Dice, Size by Weight
Figure 4: Slice-and-Dice, no offsets

Treemaps Variants

Figure 7: Treemap UNIX Experiment Results
Figure 6: 2 1/2-D Treemaps
3D Hyperbolic Space

Hierarchical Clustering Explorer
for Interactive Exploration of Multidimensional Data
http://www.cs.umd.edu/hcil/multi-cluster

Outline

- Information Visualization
- Hierarchical Data Visualization Techniques
- Ontology Visualization
- Alignment Visualization

Protégé plug-ins
- OntoViz tab [7]
- Jambalaya [8]
- TGViz [9]
- OWLViz [10]
Ontology Visualization Tools (2)

Ontology tools applying unconventional visualization techniques

- The cluster map [2] applied in Autofocus [1], Spectacle [2], the DOPE project [3], and SWAP [4].

- Ontorama [5] is a hyperbolic-style browser designed to render RDF files derived from a web-accessible ontology server called WEBKB-2, which contains descriptions of over 74,500 object types from WORDNET.

- Ontobroker [6] utilizes a hyperbolic tree view and is an ontology-based semantic indexing and instance querying technology for the WWW.

Ontology Visualization Tools (3)

Graph-based visualization tools:

- WebODE [12] uses the tool called OntoDesigner to graphically edit ontologies using common node/edge to represent the concepts and the relations in a tree.

- Tadzebao [13], which is a tool for collaborative development of ontologies, includes the tree-tool WebOnto.

- FCA [14] uses simple node-link visualizations of the inherent structure.


- Vizigator [17] represents topic maps using the Touchgraph technology [18].

- ViSWeb [19] is an OPM-based (Object-Process Methodology) layer on top of XML/RDF/OWL to express knowledge visually and in natural language.

Ontology Visualization Tools (4)

Graph-based visualization tools:

- ORIENT (Ontology engineering Environment) [20] is an Eclipse-based system using RDF-graphs and includes ontology building, mapping, evolution, evaluation and visualization.

- RDFAuthor [21] supports the creation of RDF instance data by dragging the data into a graph and binding it together using a graphical and quite simple interface.

- FRODO RDFSViz tool [24], which provides class models of ontologies represented in RDF Schema using GraphViz.

- Building ontology-based queries with different levels of guidance is the aim of GODE [25] (Graphical Ontology Design Environment).

Ontology Visualization Tools (5)

Graph-based visualization tools:

IsaViz [22] relies on GraphViz [23] to browse and author RDF models presented as graphs.
Why Visualization?

Visualization techniques support by:
- direct manipulation of the classifications / concepts / instances
- providing with overview
- appropriate presentation of semantically rich query results
- visual support for exploration and querying
- focus on structure (metadata) or on data: different points of view
- efficiently comparing ontologies
- supporting creation of ontologies based on standards

Aduna AutoFocus

- Brings semantic, multi-dimensional information visualization (cluster map) to everyone’s desktop
- Lets users oversee and access the overwhelming amount of information
- Integration of different sources: local files, emails, websites, intranet resources
- Using a local Sesame RDF Repository + Aduna Metadata Server for sharing Metadata in enterprise environments
- Metadata: file type, size, date, author(s), sender, keywords
- http://www.aduna-software.com/home/overview.view

Outline

- Information Visualization
- Hierarchical Data Visualization Techniques
- Ontology Visualization
- Alignment Visualization
Introduction: Relations Among Concepts 49

<table>
<thead>
<tr>
<th>OWL Ontology Construct</th>
<th>Comparison Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concept</td>
<td>Equal</td>
<td>URI's equal.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Class member instances equal.</td>
</tr>
<tr>
<td>Syntactically equal</td>
<td></td>
<td>Labels are the same.</td>
</tr>
<tr>
<td>Similar</td>
<td></td>
<td>Superclasses are the same.</td>
</tr>
<tr>
<td>Broader than</td>
<td>Subclass superclass comparison.</td>
<td></td>
</tr>
<tr>
<td>Narrower than</td>
<td>Subclass subclass comparison.</td>
<td></td>
</tr>
<tr>
<td>Different</td>
<td>Class is different from all classes of the second ontology.</td>
<td></td>
</tr>
</tbody>
</table>

Tasks in Visual Ontology Alignment 50

- Read / assess / correct alignment result
- Examine the context of entities for both source ontologies
- Manipulate source ontologies (change labels, URIs, etc.)

Small World Graphs 51

- Neither completely regular nor completely random: Regular graphs 'rewired' to introduce increasing amounts of disorder
- Two characteristic features: clustering coefficient high and average path length short
- Variety of edge lengths, with shorter lengths for edges in tight clusters, longer lengths for random edges between clusters

Small World Graphs 52

Small-world phenomenon: according to Milgram each actor in a social network is linked to any other with a maximum of 6 intermediaries. Experiment in 1967 suggested that two random US citizens were connected on average by a chain of six acquaintances.

Smaller communities, such as mathematicians, are densely connected. Mathematicians use the Erdös number to describe their distance from Paul Erdös based on their shared publications. The Erdös Number Project: http://www.oakland.edu/enp/
Small World Graphs

- Based on a spring-embedded algorithm that position tightly coupled groups of nodes closely together and loosely coupled groups of nodes far apart
- Uses clusters to group the nodes of a graph according to the selected level of detail (degree of abstraction DOA ∈ [0, 1])

- Distance between two clusters of nodes is inversely proportional to their coupling (LinLog)
- Average link uses the average distance between all members

Users' Goals:
- Are there any distinct groups of items that are strongly interconnected (i.e. graph clusters)?
- How do these split into separate clusters?
- How do these clusters relate?

All spring-embedded algorithms bear the problem of high computational complexity - usually O(N^3), Optimization: O(N^2 Log(N))

Clustering the graph improves program's interactivity: On average there are only O(Log(N)) clusters visible

AlViz Implementation

- Tab widget plug-in for Protégé 3.2
- AlViz links four views in order achieve a better integration of overview and details
- Represents the entities linked together according to selected mutual properties such as IsA, IsPart, IsMember, locatedIn, hasOwner, isMadeBy, ...
- Color encodes alignment type

- Reduced saturation indicates mixed clusters
- Different levels of detail (degree of abstraction)
- Shape and size of cluster represents number of nodes
- Implementation: 2D graphs (based on implementation from Stephen Ingram)
Small World Graphs: Subgraphs of Tourism Ontology

- Focus on a certain entity, visualization the entity and its context
- Small world graph visualizations of two ontologies in the tourism domain: the focus of the graph is on the entity 'Urlaub' showing all related concepts for both ontologies
- Labeling is activated
- This view includes all sub-entities (transitive relation) and directly related entities (non-transitive relation), supplemented with all relations and entities among them within a beforehand defined number of hops (relations)
- The nodes are not clustered meaning each node of the graph represents one entity

The edges represent three different types of relations in tourismA: the depicted relations are:
- IsA, hatReisedauer, hatZiel, hatReisemittel
- In tourismB: IsA, hatEineDauer, manBenoetigtAusruestung, hatEinZiel
- The IsA paths are shorter than the other because we gave these edges a higher weight
- To distinguish different types of relations such as functional, transitive, or non-transitive we apply different weights, which can be modified by the user according to the exploration needs
• By moving the cluster sliders next to the graph
the user can zoom in or out
• The number of aggregated entities
is shown next to the label
• This example shows the clustering along the `IsA`
relations - transitive relations are clustered first

Partly Clustered Subgraphs of Tourism Ontology

• Clustering emphasizes the structure of the ontology
• An iterative process of zooming in and out allows to explore the ontology on
different levels of detail.
• Here clustering fades out the `IsA` relationships among the entities
  focusing on the non-transitive relations of the central entity `Urlaub`
• In tourism A `Urlaub` is related to
  `Gebiet`, `Fortbewegungsmittel`, and `Zeitraum`
• In tourism B the related entities are: `Ausruestung`, `Root`, and `Zeitraum`

AlViz

Many strengths:
• Location:
  Where do most of the mappings between ontologies occur?
• Impact:
  Do the mapping choices directly or indirectly affect parts of the
  ontology the user is concerned about?
• Type:
  What kinds of mappings occur between the ontologies?
• Reason:
  Why do this mappings exist?

... open issues ...
• Include focus+context techniques (e.g., distortion or SDOF)

• Labeling / Coloring of edges

• Stronger integration of AlViz and the alignment algorithm: re-calculate alignments?

• Detailed user testing

Some References


Some References


Some References


Thanks to ... Silvia Miksch for making nice slides on hierarchical data visualization available.