Chapter 11
Information Visualization and Policy Modeling

Kawa Nazemi
Fraunhofer Institute for Computer Graphics Research (IGD), Germany

Dirk Burkhardt
Fraunhofer Institute for Computer Graphics Research (IGD), Germany

Martin Steiger
Fraunhofer Institute for Computer Graphics Research (IGD), Germany

Jörn Kohlhammer
Fraunhofer Institute for Computer Graphics Research (IGD), Germany

ABSTRACT

Policy design requires the investigation of various data in several design steps for making the right decisions, validating, or monitoring the political environment. The increasing amount of data is challenging for the stakeholders in this domain. One promising way to access the "big data" is by abstracted visual patterns and pictures, as proposed by information visualization. This chapter introduces the main idea of information visualization in policy modeling. First abstracted steps of policy design are introduced that enable the identification of information visualization in the entire policy life-cycle. Thereafter, the foundations of information visualization are introduced based on an established reference model. The authors aim to amplify the incorporation of information visualization in the entire policy design process. Therefore, the aspects of data and human interaction are introduced, too. The foundation leads to description of a conceptual design for social data visualization, and the aspect of semantics plays an important role.

INTRODUCTION

The policy modeling process and lifecycle respectively is characterized by making decisions. The decision making process involves various stakeholders, that may have diverse roles in the policy making process. The heterogeneity of the stakeholders and their "way of work" is a main challenge for providing technologies for supporting the decision making as well as technologies to involve various stakeholder in the process. Stakeholders in this context may be citizens too, whereas often the term "eParticipation" is used in this context. Information visualization techniques provide helpful instruments for the various stages of decision making. To elaborate the different

DOI: 10.4018/978-1-4666-6236-0.ch011

stages of policy making and the role of visualization in each stage, we have developed three-stepped design process for the roles of visualizations in the policy modeling lifecycle (Kohlhammer et al. 2012). The model propagates the steps of information foraging, policy design and impact analysis, where various visualization techniques can be applied to. These steps are investigated in particular for the FUPOL project, where the information foraging stage covers the visual representation of various data and data formats to get a comprehensible and understandable view on the given masses of information without losing the context and targeted task. The impact analysis step will use and cover both, the outcomes of the simulation activities of FUPOL. The outcomes of the statistical data mining methods will be covered to support both, the active and passive involvement of the citizens and to provide a kind of "public mood" about a certain topic.

For decision making in the policy life cycle, Data, information, and knowledge are crucial and important resources. Beside storing, managing and retrieving data, one important factor is the access to the increasing amount of data. A promising discipline facing the information-access challenge by investigating the areas of human perception, human-computer interaction, data-mining, computer vision, etc. is information visualization. One main goal of information visualization is the transformation of data to visual representations that provides insights (Keim et al. 2010) to users and enable the acquisition of knowledge. The access to data is provided by interactive "pictures" of knowledge domains and enables solving various knowledge and information related policy tasks. These "pictures" are generated through transformation and mapping of data (Card et al. 1999) to visual variables (Bertin 1983) that are perceived by human to solve tasks (Shneiderman 1996). Different approaches on creating this "picture" of data provide various ways of perceiving visual representation of data and interacting with them. The most popular way is to get first an overview of the entire domain knowledge in an abstracted way, followed by zooming and getting more detailed information about the knowledge-of-interest (Shneiderman 1996). This top-down approach (*Information Seeking Mantra*), proposed by Shneiderman (Shneiderman 1996) makes use of our natural interaction with real world. Getting into a new situation forces us to build association of known or similar situations and create an overview of the context. Further interactions in this situation are more goal-directed and detailed. The complementary bottom-up approach, premises that we are able to verbalize a problem or direction. The visual representation is then generated by the results of a search query. Based on the amount and complexity of the results various visualizations may provide abstracted views or detailed visual knowledge representations.

The process of information search can be further optimized by the technologies and methods of formalized semantics and ontologies, in particular in context of the Semantic Web.

Semantic Web targets on a machine-readable annotation of data to provide a "meaning" by defined and formalized relationships between resources on web. (Kohlhammer 2005) While Semantic We focuses on the machine-readability, Information visualization focuses on the maximization of our perceptual and cognitive abilities (Chen 2004).

In context of Information Visualization the aspects of data, user and tasks are of great importance. For designing Information Visualization tools the question: which data to what kind of users and for solving which tasks may provide an adequate design process. In this context the recent research investigates in particular the feedback loop to the data in Visual Analytics, the model-based visual knowledge representation in Semantics Visualization and the cognitive-complexity reduction of users in Adaptive Information Visualizations (AIV).

This chapter introduces information visualization as a solution for enabling the human information access to the heterogeneous data that are necessary during the policy modeling process. Therefore we first identify the steps of policy design, where information visualizations are required based on an established policy life-cycle model. Thereafter a foundational overview of information visualization will be given, investigating beside visualization techniques, the entire spectrum of data to visualization. In this context data and interaction methods will be introduced. We will conclude this chapter with a conceptual example of visualizing social data in the domain of policy modeling.

ABSTRACT POLICY MODELING STEPS

Policies are usually defined as principles, rules, and statements that assist in decision-making and that guide the definition and adaptation of procedures and processes. Typically, government entities or their representatives create public policies, which help to guide governmental decision-making, legislative acts, and judicial decisions.

Some policy-modeling researches emphasize theoretical respectively formal modeling techniques for decision-making, whereas applied research focuses on process-driven approaches. These approaches determine effective workflows through clearly defined processes whose performance is then monitored (for example, as in business process modeling). This applied-research approach is widely seen as one way to effectively create, monitor, and optimize policies. One aspect of process-driven policy making is the clear definition of the sequence of steps in the process. This ensures the consideration of the most relevant issues that might affect a policy's quality, which is directly linked to its effectiveness.

Ann Macintosh published a widely used policy-making life cycles; it comprises these steps (Macintosh 2004):

1. Agenda setting defines the need for a policy or a change to an existing policy and clarifies the problem that triggered the policy need or change.
2. Analysis clarifies the challenges and opportunities in relation to the agenda. This step's goals are examining the evidence, gathering knowledge, and a draft policy document.
3. Policy creation aims to create a good workable policy document, taking into consideration a variety of mechanisms such as risk analysis or pilot studies.
4. Policy implementation can involve the development of legislation, regulation, and so on.
5. Policy monitoring might involve evaluation and review of the policy in action.

The general process model of Macintosh was applied to identify the need and advances of information visualization in the entire process (Kohlhammer et al. 2012). Therefore the model was abstracted to a highest level for identifying general and abstract information visualization steps: The need for a policy, the policy design, and impacts of the designed policy are shown in Figure 1.

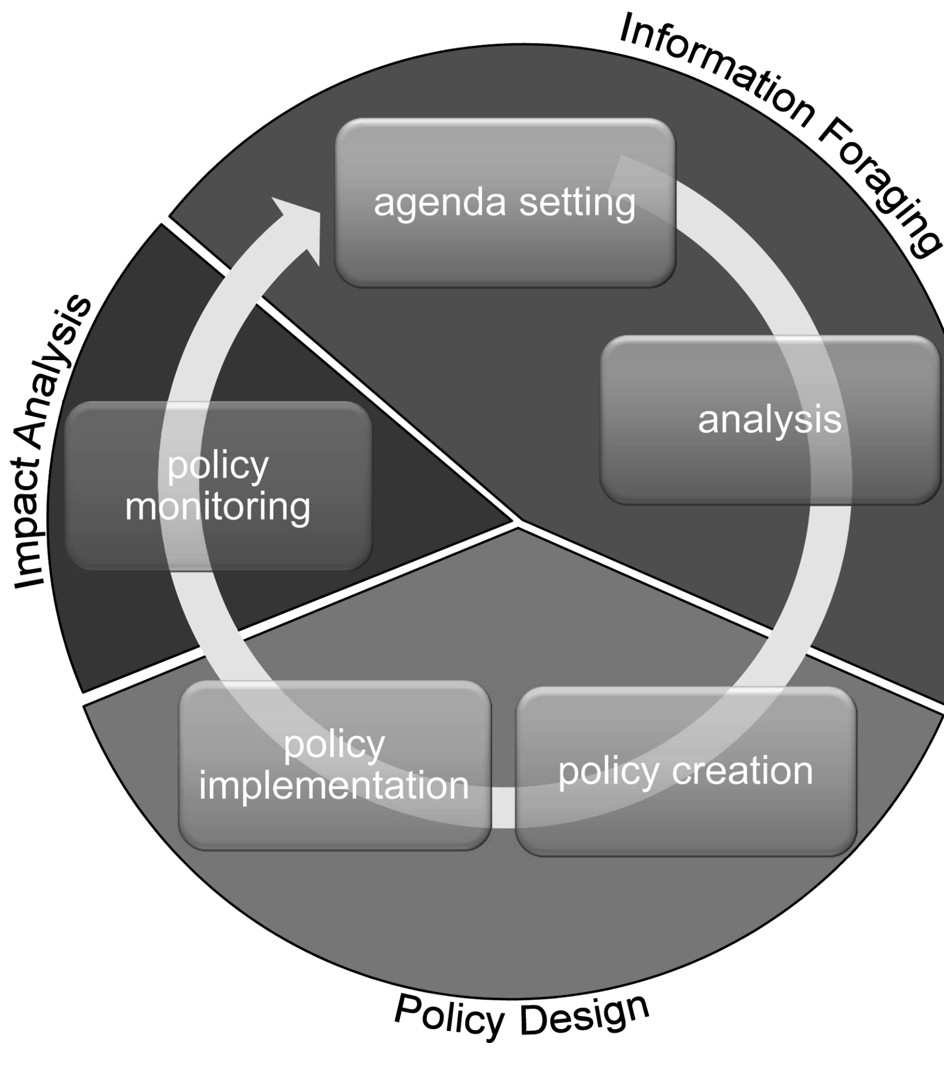
For adopting visualization in policy making, we simplified the general model and introduced three iterative stages (Kohlhammer et al 2012):

Figure 1. Abstract Policy steps

1. **Information Foraging:** Supports policy definition. This stage requires visualization techniques that obtain relations between aspects and circumstances, statistical information and policy-related issues. Such visualized information enables optimal analysis of the need for a policy.
2. **Policy Design:** Visualizes the correlating topics and policy requirements to ensure a new or a revised functional interoperability of a policy.
3. **Impact Analysis:** Evaluates the potential or actual impact and performance of a designed policy, which must be adequately visualized to support the further policy improvement.

All phases involve heterogeneous data sources to allow the analysis of various viewpoints, opinions, and possibilities. Without visualization and interactive interfaces, handling of and access to such data is usually complex and overwhelming. The key is to provide information in a topic-related,

Figure 2. Mapping of the five policy steps to the simplified model of information visualization in the policy making process (adapted from Macintosh 2004 and Kohlhammer et al. 2012). (Own drawing).



problem-specific way that lets policy makers better understand the problem and alternative solutions.

Today, many data sources support policy modeling. For example, linked open government data explicitly connects various policy-related data sources[1]. Linked data provides type-specific linking of information, which facilitates information exploration and guided search to get an overview and a deeper understanding of a specific topic. Further data sources may be the massive and growing statistical data provided by various institutions, including the EC[2].

Current policy modeling approaches do not use visualizations intensively neither for the general process nor for the entire identified stages.

The gap between information need and information access can be efficiently closed via information visualization techniques. The next sections will introduce some main aspects of information visualization independently from policy making and design. This should amplify actors in policy design to investigate information visualization as an instrument for the information provision process.

FOUNDATIONS OF INFORMATION VISUALIZATION

Model of Information Visualization

One of the most influential model in information visualization is the model of Card, Mackinlay and Shneiderman. It is a data flow diagram that models the data processing from its raw form into a visual representation. The visualization is described as a series of partly independent transformations. Its main contribution is that the complexity of the visualization process is split into smaller sub-processes. This is why it still serves as a basis for many visualization system architectures today. Usually, scientific contributions in the information visualization domain can be mapped precisely onto particular parts of the pipeline. Another important aspect of their work is the idea of user interaction in the pipeline. A visualization technique is not static process. Every component along the data processing pipeline serves as a basis for process control mechanisms.

The pipeline starts off with the transformation of the raw input data into data formats that are suitable for the visualization. This standardization is necessary if more than one data source should be attached to the process or if a single data source is used for different visualization techniques. This transformation aims at a data representation that is normalized in terms of content and structure so that the visualization can be decoupled from the input data. This is an important strategy that permits to adapt techniques to different scenarios and data sets. It might involve trivial operations like converting one data format into another, but in many cases it is also necessary to identify and deal with incomplete, imprecise or erroneous data. Depending on the application the outcome of this step is well-defined data for the visualization.

The second step in Card's visualization pipeline is the mapping of standardized, but raw data into the visual space. This mapping can be considered as the core transformation that forms the actual visualization. That is why the different visualization techniques can be differentiated in thispart of the pipeline. The visual space is described by a series of visual attributes which inherently represent the basic tools of the visualization techniques. Ware identified several groups of these attributes: form, color, animation and space (Ware 2013). While the second part of the pipeline describes the transformation into the visual space, the third block is about transformations within the visual space, the view transformation. In almost any case the transformation also takes place within the value set of a single visual attribute. This includes, for example, rotation, zoom and other camera settings as well as modifications of the color map for an attribute.

Card's model of the visualization pipeline is a also a model for a technical realization of visualization techniques and processes. Together with Mackinlay and Shneiderman he also develops a model for what he calls "Knowledge Crystallization Process". Instead of describing the data flow through the technical components they model the path from input data to application-dependent, domain-specific knowledge. This crystallization resembles the classification of analytic artifacts as done by Thomas & Cook (Thomas and Cook 2005). It models a cyclic process that repeats of the following steps:

- Forage for data.
- Search for schema.
- Instantiate schema.
- Problem-solve.
- Author, decide or act.

The proposed sequential cycle can be altered by several feedback and feed-forward loops that are the main characteristics for this model. Whether or not these loops are executed depends strongly on the application scenario. In most cases, human-interaction is required whenever a decision has to be made. In order to do that the human must be able to judge the available results. This task can be performed through automatic analysis if the judging process can be explicitly formalized.

The model of Card and Thomas et al. complement one another in the sense that the model for the knowledge crystallization process is independent of the technical realization. The single steps solely describe the way knowledge is gained and the tasks that perform in each step. The model of Thomas et al. is still valid if the interactive visualization techniques are replaced by automatic analysis methods, as done, for example in data mining.

Thomas & Cook define as the principle of knowledge crystallization as analytic deduction but focus on different aspects. Analytic artifacts appear in knowledge crystallization only implicitly whereas the transformation process and their application is put in the foreground. In many cases, the approaches for the theory and the models in information visualization can be assigned to one of two groups. These are "data-centered" and "decision- or user-centered" tasks. They differ mainly by the information that is available in the design phase. Amar and Stasko (Amar and Stasko 2005) put those two principles in juxtaposition in the context of information visualization. Visualization in data-centered approaches aims at a realistic representation of data and its structure. In its most consequent form, this idea is completely independent of the human user and the tasks that should be solved using that visualization. Its main goal is to create an identical replication of the input data in the mental model of the user. Viewing the data is an elementary low-level process. It is supported through visualization, but it does not support the user in solving a high-level task. According to Amar, the static connection between analytic activities is based on the assumption that the aims of the user are also formulated in a static and explicit manner. They find it necessary to link the user tasks on different abstraction layers through information visualization, i.e. low-level and high-level tasks.

In the following sections we will present two parts of the Card pipeline: the visual mappings and the interaction techniques. Mappings can be partitioned in five different groups that map fundamentally different structures into the visual space. Interaction techniques can be roughly classified by the part of the visualization pipeline they control. In this manner, the differentiation is performed through technical criteria. However, it would also be possible to separate the visualizations by the task they support. Although many techniques are advertised through the tasks they claim to solve, comprehensive studies that compare many different techniques is not yet available in the literature. Wherever possible, we will present reviews as found in the literature and express our own opinion where appropriate.

DATA FOUNDATIONS

The information visualizations model that was described in the previous part always starts with the transformation of data in their raw form. Heterogeneous data types need to be investigated for the transformation process. Shneiderman (Shneiderman 1996) introduced a taxonomy of data types, which distinguishes data types in one-, two- and three-dimensional data, temporal and multidimensional data, and tree and network data. We will shine light on these categories in this section of the chapter. Together with an independent taxonomy of analysis tasks, Shneiderman also presented a matrix of visualization techniques, which provides solutions for specific tasks and data. It has to be stated, however, that it is quite common that a given dataset falls into more than one of these categories of the taxonomy. The term "dimensionality" may either refer to the dimension of the actual data, or to the dimension of the display. In some cases, if the data set has a "native" dimensionality (as is the case with most geo-spatial datasets) the preferred visualization techniques map this data onto its native space. Also note that most of the visualization systems presented here employ one or more navigation and interaction concepts that were described in the previously, without being mentioned here. We make a clear distinction between publications introducing basic technology and visualization techniques of the "second generation", in which most of these technologies are implemented as a quasi-standard an in nearly all cases used in combination. The work of Keim (Keim 2002) gives a contemporary survey on the basis of Shneiderman's taxonomy.

One-Dimensional/Temporal Data

Tables with two columns are a typical example for one-dimensional datasets. If they contain at least one temporal component in their structure, they are referred to as temporal dataset and form a special subclass of 1-dimensional data. Shneiderman also includes textual documents, program source-code, lists and all other kind of sequentially arranged data to the category of one-dimensional data. Whether text documents actually belong to this category depends on the perspective and task. If the central focus lies on the individual items in the sequence (as for searching words in a document), the corresponding space is one-dimensional. If the focus lies on the sequence as a whole (as in document analysis and classification), the data space actually is multidimensional. Given the usual complexity of input data sets, they do not fall in the category of one-dimensional data alone. In this paragraph we present a number of visualization approaches which emphasize the temporal / one-dimensional components of the datasets.

Havre presents a visualization technique called ThemeRiver as part of a document analysis of news reports (Havre et al. 2000). It maps the change of headline stories in the news onto a time scale. The basis of this technique is the appearance of a specific keyword appearing in a number of articles and shows how specific themes may appear at the same time (though not on a granularity level of a single article). Card et al. describe a type of visualization (Card et al., 2006) that maps the temporal data is also onto a single axis, a time-line. This visualization couples temporal and hierarchical data. For the problem of mapping temporal data to a visual aspect, which is neither a time-line nor an animation, no convenient solution exists. In most cases, one of these variants is chosen, because they can be intuitively understood.

The work of Hochheiser and Shneiderman (Hochheiser and Shneiderman 2004) lies in the tradition of a number of tools which refine the dynamic queries technique. As in the other visualization techniques, the temporal information is mapped onto the timeline. The use of so-called TimeBoxes covers a spatial and temporal interval to intuitively define a number of data filters to identify time-series, which share a common behavior. Timebox queries are combined to form conjunctive queries of arbitrary complexity. These techniques are conceptually not restricted to temporal data. Every temporal dataset that is used in these techniques can be replaced with one-dimensional data of any other (ordinal) type. Lin et al. give a survey on the different techniques for the analysis of the same kind of data, including Timebox-Queries, calendar based visualization techniques. The authors also contribute VizTree, which interactively visualizes a similarity analysis in a number of data graphs, producing similarity trees (Lin et al., 2005).

Hao et al. (Hao et al. 2005) propose another combination of clustered / hierarchical data together with a large time-series data set. In their application scenario, the time-series entities show intrinsic hierarchical relationships. This technique combines tree-map properties with the ability to show temporal development of stock-market prices. The hierarchical properties of the underlying data are used to match the level of interest and importance in the layout.

The approach proposed by Voinea at el. (Voinea et al. 2005) in the field of collaborative document creation management deals with a completely different kind of data. The authors focus on software development source code files which require significantly different processing than plain text documents. The creation process of the software is clearly separated in a one-dimensional aspect (the position of lines added to the source-code), and the temporal aspect (the development of the source over time), both of which are combined in a two-dimensional overview. Different parts of the code can be identified by their author(s), such as stability and other aspects.

Two- and Three-Dimensional Data

The mapping of abstract two and three-dimensional data has by far the longest tradition. All kinds of geospatial information visualization can be identified as a mapping from data in a two-dimensional space (geographical maps) or three-dimensional space (a virtual model of our physical world). Every atlas can be considered a collection of physical data and geographic metadata which accounts for most of the earliest efforts in actual information visualization. Embedding abstract data into a representation of our physical world is one of the most powerful metaphors, because humans are attuned to organize and arrange mental mappings while copying our physical world. Hence, many visualization techniques for this embedding have been developed. Over the years, this concept evolved from plain satellite image visualization to a collaborative platform for which the (virtual) world serves as a common frame of reference to contribute, search and analyze large amounts of additional geographic metadata. Not surprisingly, many visualization techniques have been developed that use this platform as a basis for their data (Chen and Zhu 2007). With a special focus to the spread of avian flu, Proulx et al. combine the embedding of spatial, temporal and other metadata to actually formulate and test hypothesis on the basis of “events” (Proulx et al. 2006). Events serve as metadata containers which are used to bind the information to a place, time, etc.

One of the most prominent mappings of abstract data into two-dimensional space is the scatterplot technique, which appears in a large number of variants (North 2000). Despite the fact, that the native display space is only two-dimensional (although three-dimensional scatterplots exist), they are often used in combination as scatterplot matrices or with other techniques to be used in multidimensional data analysis. Scatterplots work best for numerical data (which can be mapped on the x and y coordinates respectively), and is of limited usage to convey purely semantic information. Because of their simple metaphors (points in n-dimensional space become points in 2-dimensional space), they are most conveniently used to visualize projections between n-dimensional data-space and display-space, which usually are supported by numerical methods just as factor analysis, matrix decomposition and similar methods.

One field of two- and three-dimensional mappings has been left out on purpose: Scientific visualization as is separated from information visualization by the data that is displayed. By definition, it deals with physical data which inherently lies in physical space rather than abstract information and metadata. Consequently, the techniques of scientific visualization are out of scope.

Multidimensional Data

Most of the techniques presented here involve data which covers more than three independent dimensions. Visualization techniques for multidimensional (or multivariate) data explicitly address the problem to visualize and identify inherent dependencies in the datasets, which cannot be expressed by simple correlations. Hidden relations may incorporate ten or more dimensions of data, and one of the major goals in all of these techniques is to display a sufficiently large number of dimensions in (2-dimensional) screen space to make these correlations visible. Defining "Visual Data Mining" as a concept the work of Keim gives a survey on a number of visualization techniques for multi-dimensional databases (Keim 1996). Aside from graph-based visualizations for networks and hierarchies, two classes of techniques evolved over the years to become prominent representatives for the visualization of multi-dimensional data: The first one is the so-called parallel coordinates technique, the other one falls into the category of pixel-oriented layouts. It has to be noted, that all of these techniques virtually never appear in their "pure" (i.e. conceptual) form. Most of the recent frameworks and techniques derive their improvements from an adequate combination of different basic techniques – in some cases in the same display. This holds true especially for glyphs, which also constitute a group of multidimensional visualization techniques, but does not refer to the layout (i.e. the positioning of visual objects in screen space) but on the appearance of objects. Basically every single visual object that conveys more information than its position can be considered as a glyph.

The parallel coordinate technique, as the name suggests, has all axis in the display arranged in a row of parallel lines. Basically this technique can be used for nominal, ordinal or numerical axis, but it works best for ordinal and numerical data. A "point" in the n-dimensional space is drawn as a poly-line connecting the (coordinate-) values on every axis. While the basic idea is relatively old, contemporary studies on parallel coordinates emphasize their use for the analysis of datasets (Siirtola 2000). In many cases, this technique is tightly coupled with the generation of dynamic queries. Both of these techniques illustrate the identification of data clusters by visual/manual methods (Siirtola 2000) and a method the display the data at different structural levels (Fua et al. 1999).

Complementary to that, the general idea of pixel based methods is to use the screen space in the most efficient way possible: Every pixel in the display area is used to convey different information: The use of "non-data-ink" is reduced to a minimum. Pixel-based techniques must cope with the layout problem of an adequate mapping of the (multidimensional) data-space onto the screen space. In many cases there is no strict correspondence between the similarity of the data items and their distance. (Keim 1995, Keim 1996 and Keim 2000) provide a good overview over the general idea of these techniques.

Tree and Network Data

Graph visualization has become an important topic in information visualization area over the past years. The display of networks helps to analyze of relationships between entities rather than the entities themselves. Graph visualization is used in many different application areas. For example, the site maps of web sites as well as the browsing history of a web browser can be displayed in a directed graph. In biology and chemistry, graphs are applied to evolutionary trees, molecule structures, chemical reactions or biochemical pathways. In computing, data flow diagrams, subroutine-call graphs, entity relationship diagrams (e.g., UML and database structures) and semantic networks and knowledge-representation diagrams are the main application fields. Furthermore, document management systems profit from document structure and relationship visualization. Social networks visualization has also become a popular application of graph visualization methods.

The key issues in graph visualization are the graph structure (directed vs. undirected graphs, trees vs. cyclic graphs) and their size. A survey of graph visualization techniques for different graph types can be found, for example, in the work of Herman (Herman et al. 2000). The graph display is driven by its layout. There are different graph layout techniques suited for different graph types.

For trees (graph in which any two vertices are connected by exactly one path) the classic layouts will position children nodes "below" their common ancestor (Reingold and Tilford 1981), in 3D a cone layout is used (Robertson et al. 1991). For large graphs the high node and link overplotting requires new visualization and clustering techniques, For example, 3D hyperbolic space layouts (Munzner 1997) or *treemaps* (van Wijk et al. 1999).

VISUALIZATION TECHNIQUES

As described in the previous section, a visual mapping is a transformation of the data flow performed by visual techniques and will be used for their classification. It is important to note here that visualization techniques contain almost never a visual mapping in pure form. Especially newer techniques are often combinations of older approaches. Some of them are explicitly mentioned in a separate sub-chapter at the end of this chapter.

Each of the following sub-chapters presents a category of visualization- and interaction techniques with a focus put on newer approaches. The classification we performed is similar to the multi-dimensional visualization technique classification done by Keim (Keim 2000) which we extend with a class that deals with projection methods. Whenever possible, the techniques are presented independently of their application domain. Where ratings of a technology are provided, then these are typically related to the technique's ability to solve a particular task rather than the type of data they display.

Instead of describing iconic data like Keim does, we focus on projection methods, because they are tightly coupled with methods from data-mining. Moreover, the class of pure iconic techniques has lost importance during the past couple of years. Today, the results of this domain are reused particularly in glyph-based designs. Glyphs are singular symbols for data objects that represent one or more attributes.

Keim provides a classification survey of visualization techniques combined with a comparison regarding different characteristics of the data, the tasks and inherent properties of the visualization itself.

The survey can be separated into three independent task groups: task-related, data-related and visualization-related characteristics. The associated questions are: Which tasks can be solved? What kind of data is suitable? What are the inherent properties of the technique?

Keim starts his evaluation by testing task-related capabilities of the techniques. The first task is the support for cluster identification in a dataset ("clustering") and describing the distribution/cumulation of points in high-dimensional space ("multivariate hotspots"). Data-related capabilities comprise the number of attributes, the number of data objects and the possibility to faithfully map nominal scales ("categorical data"). Among others, the inherent properties of the technique comprise effective use of available space measured through the overlapping area of the visual items ("visual overlap"). The last criterion is the experienced difficulty learning a technique ("learning curve").

Geometric Methods

Every visualization technique that maps a data element directly on a visual attribute that is more complex that a single pixel (e.g. lines, glyphs, etc.) belongs to the group of geometric methods. It is highly heterogeneous and contains many hybrids that also belong to class of projection methods. Most of the classical diagrams like starplots, pie charts, bar charts, line charts, histograms, etc. as well as geographic maps, parallel coordinates, scatterplots and scatterplot matrices. As an example, scatterplots can also be considered as a projection method.

One of the most important visualization techniques are line charts. They display one-dimensional functions like time-series in many application areas. Hochheiser and Shneiderman. (Hochheiser and Shneiderman 2004) present a Timebox-Widget that allows for interactive selection and dynamic filtering of the displays data sets. It is based on the older technique "Dynamic Queries" combined with a new visualization approach. The user defines a box selection implicitly through one or more intervals of attribute values that are mapped to the x- or y-axis in the display. These intervals define the data sets that lie completely within these data sets.

Equally important are geo-related data mappings. Every atlas can be seen as a collection of geo-data and geographic metadata. Embedding this abstract information in a geographic representation is one of the most abundant metaphors possible, because the reference to a location is one of the most important relations people use to organize information. Proulx et al. (Proulx 2007) display geo-data together with a time-based mapping in order to combine the two natural reference frames (space and time).

An example for an interesting combination of techniques is presented by Bendix et al. (Bendix et al. 2005). It has been chosen, because it combines two of the most popular techniques – parallel coordinates and dynamic queries. Compared to most other techniques, the parallel coordinates approach excels in that as the number of attributes is only limited through the amount of available screen space. Every attribute is mapped on its own axis which is parallel to every other axis. One element of a data set is thus represented as a polyline that intersects with all axes at that point that represents the value of the respective attribute. Data clusters and correlations can be easily identified if the attributes are adjacent. Bendix et al. put their focus on the search of describing expressions rather than the data set itself. This search of expressions is, apart from the search of patterns, a major aspect in visual data analysis. Technically spoken, they deal with the mapping of nominal data types. As they do not have a natural ordering, they display the relations between different classes instead of the data set itself. (Bendix et al. 2005)

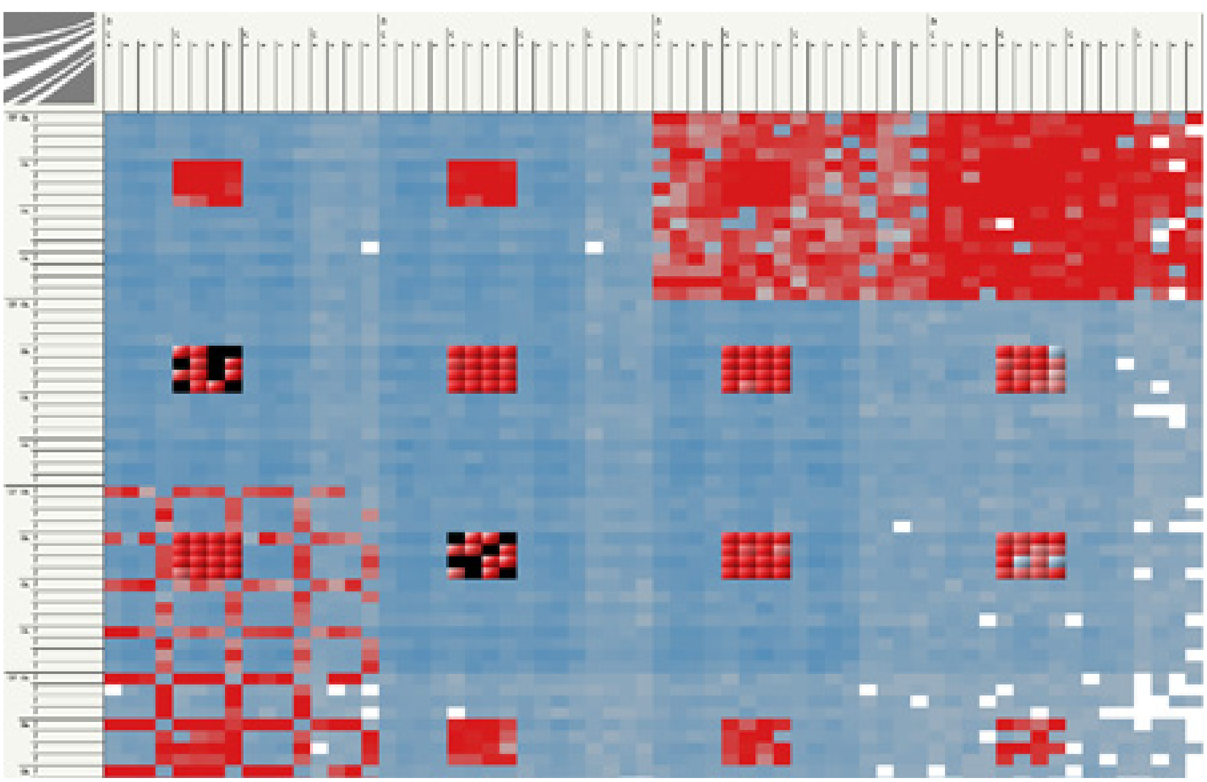
Pixel-Based Techniques

A visualization technique belongs to the group of pixel-based methods if the number of used visual attributes comprises only the position and color of a single pixel. Consequently, every pixel represents a data element which permits to display a maximum number of data elements at the same time. Pixel-based methods impose two design-problems. The value set of an attribute must be mapped to the range of available colors, but this is a problem that persists in most visualization techniques (Wijffel 2008).The second problem is about arranging the pixels related to the data set. The visualization can be seen as a function that values from high-dimensional space on the 2D screen.

A definition of pixel-based methods and a more formal description and can be found in the work of Keim (Keim 2000). The function that maps data elements in the visual space can be seen as the result of an optimization process. Assuming that the data set is ordered, this optimization must ensure that the one-dimensional ordering is kept also in the two-dimensional display. Equally important is the selection of the display area that ensures that the average distance between pixels that belong to the same dataset is minimal. The purpose of that is to aid the user in finding relations between different attributes in a data set.

May et al. present a visualization technique that maps multiple attributes on the same display. Every single pixel stands for a range of values that covers several data objects at the same time. The aggregation of the data values defines the final pixel color (May et al. 2008). In contrast to many other techniques the interesting information is hereby contained in frequencies. Pixels that relate to similar value sets can be, but do not need to be contiguous. Repetitions in well-defined horizontal

Figure 3. Pixel-based visualization from May et al. 2008 (with permission)



or vertical distances also indicate correlations. The human recognition is able to detect patterns in complex structures even if the data is distorted by noise. While pattern detection is easy, interpreting their meaning is often challenging.

Pixel-based techniques are often suitable for explorative analysis of patterns and distinctive features. Displaying previously found relations is a different task that is usually performed by different visualizations. More formally, the data model that describes the input data structure is linked but not equal to the analytic model that describes relations in the data set. Accordingly, different tasks often require different perspectives.

Hierarchies and Trees

Trees describe binary relations between differentiable elements can be described in a finite set. Most approaches in terms of visualization expose the hierarchy as dominant structure although several other attributes of the elements are present in the visualization. As the hierarchy does not impose a particular spatial structure, visualization techniques can be separated in two distinct parts. The first group deals with the design of visual mappings, i.e. the selection of attributes and metaphors for the display of elements and their connections. The element position in the 2D space does not play a major role for them. The second group is dedicated to different layout algorithms that map the elements according to one or more properties into the visual space.

Keim et al. present two space-filling methods that display hierarchies in different manners (Hao et al. 2005) and Mansmann 2007). The first one displays child nodes in their own separate space whereas the latter uses – similar to a treemap – the space of the parent node. Among others, the importance of leaves compared to inner nodes has influence on which one of the two methods makes more sense. The treemap puts the focus on the leaves of the tree. In contrast, the hierarchical layout highlights nodes that are close to the root node and less dominant in the treemap.

The nodes are displayed as simple rectangles in both cases which leaves room to show additional information. They can be used as a basis for a visualization of its own. The only restriction is that the amount of available screen space is defined by the tree layout. However, practically all visualizations for trees and graphs have in common that their ability to query and to display details is rather limited and often insufficient. This is why they are often combined with other methods, e.g. graph visualizations. Holten (Holten 2006) gives an example of such a combination. A node-link diagram is shown on top of a hierarchy with different aspects of the data. The edges between nodes are gathered in bundles in order to reduce the overdrawing and thus increase the readability of the graph.

A simple variation of node-link diagrams is the traditional Dendrogram. It is characterized by the fact that all nodes of a hierarchy level are in the same line. This significantly improves the visual arrangement of the tree. The simplicity of the structure and the display allows more complex information presentation. Up to a certain point it is possible to create abstractions of the components and use more or less independent techniques to display nodes, edges and the structure itself. The arising number of combinations is thus a source of new designs even without fundamental novelties.

Facing aesthetic, scientific and task-related aspects, designs tend to become overly complex which is conflicting with the user's need for easy-to-understand interfaces. A good visualization provides the relevant information on first sight without need for the user to actively search for it. This conflict has been actively discussed in the scientific community in the past years (Lorensen04) and (van Wijk 2005). The task defines, which data should be displayed, but it inherently defines as well which data should be hidden from the user as well. The data types impose a natural limitation on the repertoire of visual mappings. Today, scientists debate about the basic properties of visual mappings that are required support specific tasks with specific data sets in an adequate manner.

Graphs and Networks

Even if trees are only a specific subgroup of graphs they are typically depicted by very different techniques. Visualizations for trees exploit their simple structure, especially the fact they typically describe orderings. Compared to that, the placement of nodes in an arbitrary graph layout that fulfills certain optimality constraints is more complex, or mathematically spoken: NP-hard (Brandes et al. 2003).

Most graph visualizations are variations of node-link-diagrams. Some examples have already been given in the previous sub-chapter. As with trees and hierarchies the publications can be split in two categories: the graph layout on one side and the visualization of nodes and edges on the other side. The quality of a layout is measured in different criteria which often impose conflicting constraints. It is, for example, desirable to be able to see the most significant structures and clusters. But it is also desirable to minimize the spatial distance of related partitions. This makes it per se difficult to find a layout that is optimal for all demands.

Technically, the layout is often computed by mass-spring-simulations, so called "spring-embedders". They model the optimality criteria as an energy function. The simulation then tries to find a global minimum for that function. In a mathematical sense, layout algorithms are related to non-linear or local-linear projection methods.

One fundamental problem in graph visualization is the sheer amount of nodes many datasets contain. The number of nodes that can be displayed on the screen is rather limited. Considering that the focus of the user is either on the global structure or on a particular group of nodes it often makes sense to hide a large part of the data set. Balzer and Deussen (Balzer and Deussen 2007) create a visual abstraction on the basis of existing node hierarchies. It can be, for example, generated by hierarchical clustering algorithms. The nodes and the edges of a cluster are then combined into one single graphical element. A variation of this has been presented by Henry et al. (Henry 2007) who model this graphical element as an adjacency matrix. Their main contribution, however, is to provide interaction tools for the user.

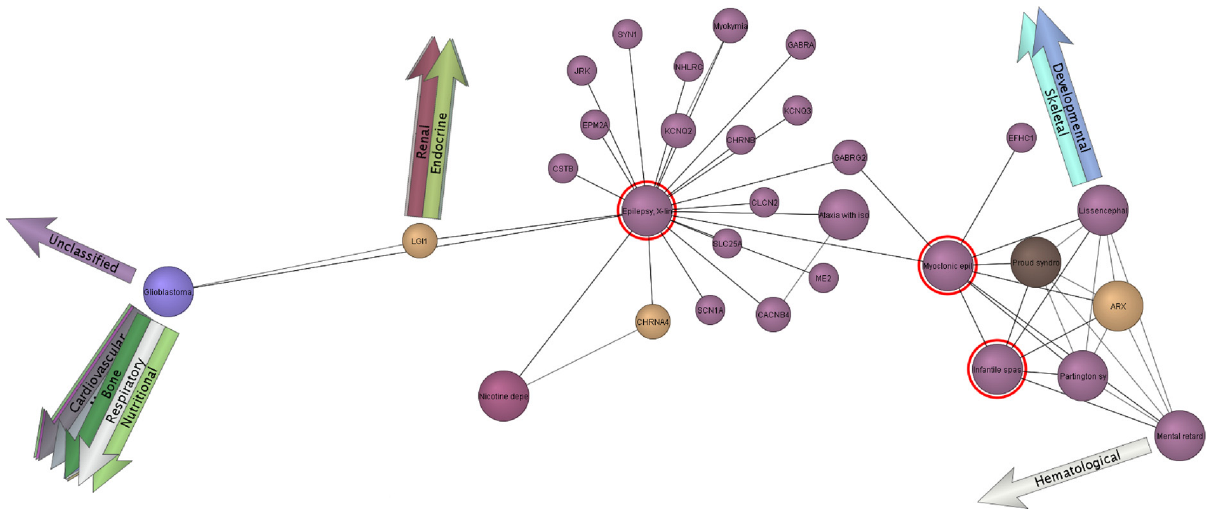
A system that is dedicated to navigate in large graphs has been developed by Abello et al. (Abello et al. 2006). The basis for that is again a given node hierarchy. It is used to display an overview on the graph that is used for navigation. At the same time, it acts as filter for the nodes that are displayed in a detailed view. Depending on the level of detail, sub-trees are expanded or collapsed.

Van Ham (van Ham 2009) faced the same problem from the opposite side. Based on an initial node pick, only a small region around a focus node is displayed. This idea has been picked up by May et al. (May et al. 12) whose system allows for more than just one focus node. It also add landmarks as graphical cues to give information on the context of the visible sub-graph. The arrows point along the shortest-path to regions in the graph that might be worth exploring.

Many combinations of techniques for the graph structure and the detail view are possible. Displaying details in the graph makes sense only if the information can be classified and processed on first sight, for example a mapping on a color scale.

The number of currently available visualization indicates already that there is no single best visualization, neither for the graph layout nor for displaying nodes and edges. The complexity of network graphs is often distributed on many structural levels. Many techniques assume that it has an inherent hierarchy. They exploit that

Figure 4. Signposts for navigation in large graphs (from May et al., 12, with permission)



by computing and using hierarchical structures for the display. Even if a visualization technique is able to switch between different levels in the hierarchy, it is probably not able to display all levels of the structure this at the same time. This does not work, because the user's visual ability to focus is limited to one or two levels. The essential task of graph visualization is thus to display one structural level as good as possible and to support user-controlled switches between different levels if necessary.

Projection Methods

This part of the book deals with projection methods. They project the data space onto the 2D visual space. This transformation is performed prior to the visual mapping. Originally, projection methods can be compared to methods from data mining domain even if the projections are of higher degree. The data space describes the set of all possible combinations of different data set attributes. Every element is represented by one point in this space. The projection tries to map the information that is inherent in this high-dimensional space into 2D. As with graphs, the focus is on the distribution rather than on accurate representation of single data elements.

Scatterplots are projection methods that are rather easy to understand. Basically, two attributes, typically numeric scales, are mapped onto the vertical and horizontal axes of a diagram. The main advantage compared to other techniques is their simplicity and the fact that most users know the concept already from math courses in school. The drawback is that only two attributes can be compared at the same time as the projection is linear along with the axes of the coordinate system. Elmquist et al. overcome this limiting with a Scatterplot-Matrix. It displays all possible scatterplots with a given number of attributes of a dataset in a matrix. (Elmquist et al. 2008) Every entry in that matrix is a miniaturized scatterplot. These small scatterplots give a first idea if and how two attributes are linked. The matrix display provides an overview and helps the user to find interesting attribute combinations, but it also solves the coherence problem for the scatterplot: modifying parameters (in this case the selected axes) modifies the user perspective in a way that the user cannot comprehend. The display before and after the modification differ so much that the user is not able to recognize the influence of the modified parameter. Animated transitions between those settings are an often used strategy to fight that problem.

A linear projection can be described as an optimization process that tries to find an optimal direction. As most optimizations do, the Principal Component Analysis (PCA) (Müller 2006) tries to minimize an objective function. For PCA, it describes the variance of points along an arbitrary axis in space. Linear projections screen all information along one projection axis, but highlight structures that are orthogonal to that axis. In case, a dataset contains structures that become manifest along several (in the worst case perpendicular) axes, linear projections fail to display the dataset properly.

Schreck at el. present a projection method that is based on self-organizing maps (sometimes also referred as Kohonen maps, named after Teuvo Kohonen) (Schreck at el. 2008). As the name already inclines, the maps are self-organizing neuronal networks that map high-dimensional attribute space in the two-dimensional display space. In contrast to other methods, the display space is discrete rather than continuous. Every discrete element corresponds to a set of classes and every data element is represented by an element that belongs to exactly one of the classes. Every class contains one element that represents the class as a whole. The classes can then be put in relation with each other in terms of similarity, or simply spoken, similar classes lie close to each other in the map.

With the exception of scatterplots, all linear projection methods work with numerical data only. Non-linear projection methods are able to work with other data types if the spatial distance between two data elements is metrically defined.

Above all, projections describe the data distribution in a multi-dimensional space. As a result, the points are mapped so that elements that are close in the data space are also close in the 2D space. Thus, these methods are particularly useful for clustering, similarity detection and outlier detection.

VISUAL INTERACTION

Many different information visualization techniques for interaction and navigation within the abstract data space exist. Hearst considers the following as the most important ones: brushing and linking, panning and zooming, focus and context, magic lenses, animation and as an additional combination overview plus detail (Hearst 1999). These techniques can be seen as the fundamentals (together with the visualization metaphors) for the design and implementation for visualization techniques.

Brushing and Linking

The interaction technique “brushing and linking” describes a connection between two or more views of the same data, based on a user-defined selection. Selecting a certain representation in one view affects the representation in other views as well. This requires that the raw data is mapped not only to one view at a time, but to several views. More specifically, brushing refers to the idea that the user picks a subset of the original data whereas linking refers to the visual highlighting in different complementary views. This Highlighting can occur in a number of forms. They all have in common that the selected item(s) can be distinguished in an intuitive way from the unselected items. This naturally limits the number of scalar dimensions which can be used in the same display. The work of Ware gives an overview on visualization features and presents how different visualization can be used to judge whether groups of objects belong together or not (Ware 2013). The basic feature classes presented are form, color, motion and spatial position. His work on preattentive perception gives important information which types of features can be used which each other, and which types of features should not be used for different information. Examples include using a different color, font, background or symbol, and adding additional labels for highlighted items (Eick and Karr 2000; Wills 1995). Depending on the sources, the brushing and linking technique is either considered as a change of the visual mapping (Hearst 1999) or as a technique which modifies the data transformation (Card et al. 1999). Most importantly, every visual mapping is required to provide an inverse mapping, by which visual structures can be remapped to a common data reference.

An example for a system implementing brushing and linking for the visualization of search results is the INQUERY-based 3D-visualization by Allan (Allan 1997).

Panning and Zooming

The view transformation from visual structure to views is often controlled by panning and zooming operations. Changing the viewpoint of the user alters the portion of the displayed part of the visual structures. Hearst uses the metaphor of a movie camera (Hearst 1999). Card et al. use the term "panning and zooming" in their listing of interaction techniques (Card et al. 1999). Their equivalent is camera movement and zooms. In contrast to simple panning, camera movement includes the third dimension, when dealing with three-dimensional visualizations. In both cases, zooming includes possible changes of the level of details displayed, when changing the zoom-factor – the virtual distance to an object of interest. An interesting contribution on zooming is the "single-axis-at-a-time-zooming", discussed by (Jog and Shneiderman 1995). While normal zooming can be explained by using a camera metaphor, this fails to work, when only the scale of one of the axes is changed.

The camera metaphor for movement in virtual (3D-)space is better-known from virtual world and games. However, a classical example for a system implementing panning and zooming for the visualization of browsing and searching is Pad++ (Bederson et al. 1996). One of the central characteristics of this system is the fact that scale is added as a first class parameter to all items displayed. In addition to implementing simple panning and zooming, Pad++ goes far beyond this interface technique. It also offers focus-plus-context views as well as overview plus detail, which are described later. In general, at least simple forms of panning and zooming are today one of the general techniques implemented in many of the available visualization systems.

Focus-Plus-Context

An inherent problem of zooming is that the higher the zooming factor is, the more details can be shown about particular items or the better the separation between close up items, but less can be perceived from the surroundings or the overall structure of the information. Focus-plus-Context techniques mitigate this problem by presenting more details about the items in focus, and less about the context while trying to avoid that the context of the information in the focus is completely hidden. Card et al. list three points as premises for focus plus context (Card et al. 1999):

- The user needs both overview and detailed information simultaneously.
- Information needed in the overview may be different that that needed in detail.
- These two types of information can be combined in a single (dynamic) display.

Overview-plus-Detail (Furnas 1981; Furnas 1986) methods can be used to cope with the mentioned problem of zooming and at least the first two of the premises, but overview plus detail does not combine both types of information in a single display. Hearst describes a fisheye camera lens as a metaphor for focus-plus-context (Hearst

1999). The trailblazers for fisheye views were two publications of Furnas (Furnas 1981; Furnas 1986) on "Degree of Interest" (DOI) functions and Sarkar (Sarkar 1992) with their extensions for graphical fisheye views. Card et al. list the following techniques for selective reduction of information for the contextual area: Filtering, selective aggregation, micro-macro readings, highlighting and last but not least distortion (Card et al. 1999). They interpret focus-plus-context as a data transformation, whereas for zooming, where a sort of filtering can also occur, they categorized the complete technique as working on the view transformation.

Examples for systems using focus-plus-context for the visualization of search results or browsing are the document lens, the table lens or the Pad++ system. The document lens (Robertson and Mackinlay 1993) is a component of the Information Visualizer system. It is a 3D tool for large rectangular presentations of documents or web page collections, like the web-book. The pages of the document of a collection are exploded out, so that all pages are available simultaneously and can be viewed using a rectangular lens magnifying the page in focus, and therefore distorting all the other pages. Another component, also using a lens metaphor, is the table lens (Rao et al. 1994). The table lens can be used for viewing of result lists or other lists in tabular form, and includes functions for magnifying lines or groups of lines whilst keeping the rest of the table viewable in compressed form. An entirely different method for a focus-plus-context, which uses semantic information technique, is presented in (Kosara 2001). Blurring is used for highlighting relevant information, without compromising the ability to show an overview of the situation.

Semantic Zooming

In contrast to ordinary zooming techniques, semantic zoom does not only change the parameters of a graphical representation, but modifies the selection and structure of the data that is displayed. Graphical zooming usually affects the displayed size of an object and – if applicable – also affects the graphical level of detail of a given object representation (i.e. the number and complexity of graphical primitives shown), based upon some distance measure. Semantic zooming, on the other hand, changes or enhances the actual type of information conferred in the graphical object(s). Usually additional graphical objects, just as annotations, flags or similar metaphors appear in the display while zooming. For every type of entity and every level of detail the structural information has to be defined. Semantic zooming is a technique for details-on-demand to avoid display cluttering in the panoramic view, while retaining all information for a more local field of interest.

Boulos presents a survey about the use of graphical map for browsing metadata resources (Boulos 2003). Map-based visualization techniques provide a natural frame of reference, by which an intuitive search strategy can be imposed to the user: The mapping defines the spatial topology – especially the "similarity in the abstract space" between points, mapped into their mutual distance. Modjeska gives an extensive survey about the navigation in virtual information worlds (Modjeska 1997). Semantic zooming can be developed for hypermedia and spatial worlds with a variety of information structures. It uses semantic information to change the physical representation of objects according to viewing scale. In their early work, Ahlberg et al. present a coupling of the semantic zooming technique and dynamic query technique in a starfield display. (Ahlberg et al. 1994)

The Magic Lens is a special form of a semantic zoom which connects the interaction method with a lens metaphor. Magic lenses allow to select an area of the view port (of either fixed or arbitrary size), and to manipulate this area with specific operators. They can be overlapped on items, and change their applied to the underlying data (Hearst 1999).

Animation

While the other techniques described so far affect data transformations, visual mappings, and/or view transformations, animation does not influence these conversions, but is affected by them. For a discussion about animation in the larger context of motion and the general usage of motion see the work of Bartram (Bartram 1997). Animation is used more and more in information visualization systems to help users keeping their orientation when transformations or changes of mappings occur. In the transition between images of the same data objects, animation is used to keep the path of an individual object coherent to human perception. The cognitive load on the user is reduced by providing object constancy and exploiting the human perceptual system (Robertson et al. 1993). Animation is used in a number of information-seeking systems like the Information Visualizer*, the Navigational View Builder*, Pad++*, or SPIRE. In the Information Visualizer*, animation is used in several ways, like for example animation rotations of Cone Trees to track substructure relationships without thinking about it (Robertson et al. 1991). In addition to animate changes (Bryan and Gershman 2000) used movement in their "aquarium" interface for a large online store to reinforce the absence of structure in the displayed items.

Especially in the context of semantic information, is has to be noted that animation is also used in the Prefuse toolkit (Heer et al. 2005) for the animation of graphs and networks. Depending on the field of interest, a different part of the structure must be centered in the viewport. This usually requires the movement of the different nodes in the network for the new arrangement. In most cases, this motion is animated to keep the mental image of the network structure consistent (Abello 2006). Animation can also be used to display actual – usually time-dependent – data (Tekusova and Kohlhammer 2007), which can also be used to add a new data-dimension to the display. This can be exploited to spot significant transition patterns over time.

Overview Plus Detail

For Overview-Plus-Detail, two or more levels of linked visualizations with different zoom factors are used. In contrast to semantic zooming, where different zooming levels are used in the same display, two or more separated displays are used. The technique helps users, while looking at a portion of the data at a detailed level, keeping an overview of the whole structure. Card et al. differentiate between time multiplexed overview plus detail displays, and space multiplexed ones (Card et al. 1999). Time multiplexing means, that overview and details are shown one at a time. Spatial multiplexing means, that overview and details are shown both at the same time at different locations on the screen. Time multiplexed overview plus detail views are conceptually not far away from simple zooming. Overview plus detail is sometimes also called map view concept (Beard and Walker 1990). Card et al. report that typical zoom factors (that is the relation between the size of the shown area in the two displays) range from 5 to 15, and that there is a limit for effective zoom factors of about 3 to 30.

Examples for systems using overview plus detail for the visualization of search results or browsing are the Harmony VRWeb 3D scene viewer, or the pre-VIR prototype by (Bekavac 1999). The Harmony VRWeb 3D scene viewer (Andrews 1995) uses a 2D-map for navigation in an information landscape. pre-VIR uses Overview plus detail in a horizontal tree view of the graph of the search results to ease navigation through the graph.

Dynamic Queries

The dynamic query technique has been presented in some foundational work on information visualization (Shneiderman 1994; Ahlberg 1994). Accessing information in databases is a major activity of knowledge workers. Unfortunately, traditional database query languages trade off ease of use for power and flexibility. The dynamic query technique is a convenient visualization of local database queries, with a simple, intuitive, interactive query refinement method. The basic idea of this technique is to generate moderate to complex queries on a database by purely visual means and to ensure that there is an instant feedback in the display showing the search results. One or more selectors control the value range of one or more attributes. Viewing a graphical database representation, users manipulate the selectors to explore data subsets rapidly and easily. In a navigational environment, dynamic queries may offer a useful way to reveal attributive information, which can facilitate way finding.

Direct Manipulation

Direct Manipulation basically manifests in two slightly different ways, depending on the relation of the manipulated object to the data displayed in the display. The graphical user interface provides elements and metaphors (buttons, sliders, etc.) which can be manipulated. In many techniques, including the dynamic query techniques presented above, the manipulation of the GUI elements may control the actual visualization, as is the case with most dynamic query techniques. This is some sort of direct manipulation regarding the GUI elements, but it is indirect with regards to the actual visualization. The means of manipulation do not necessarily correspond to the effect they cause. SHNEIDERMAN presents techniques by which this mental gap can be bridged to design intuitive interfaces (Shneiderman 2004).

SEMANTICS VISUALIZATION

Knowledge as Semantics Data

Since the announcement of the idea of Semantic Web (Berners-Lee 2010) the interest for semantic technologies and semantic data management increased. Berners-Lee et al. describe this idea as a new form of web content that provides meaning for computers systems and unleashes a reformation of new possibilities in the "web of data" (Berners-Lee et al. 2001). In this description two scientific developments joined and formed the understanding of semantic data: the developments of the World Wide Web and the semantics formalisms. These formalisms where predominantly subject in the field of artificial intelligence. (Berners-Lee et al. 2001)

In artificial intelligence formalisms for formal semantics where elaborated as knowledge base. Typically this knowledge base was designed for a specific application scenario. Hence the possibilities of reuse were limited. To overcome this limitation web-based semantic markup languages emerged in the Semantic Web. In the first step this markup language had been an extension inside HTML code to assign metadata, as semantic, to data fragments, like e.g. a telephone number. These machines enable to the interpretation the data fragments, e.g. as a base for calculating the relevance of a data fragment for solving an information need of the user. But here the interpretation logic is nested within the machines. Therefore, shortly afterwards the first semantic extension of websites, the trend moved to formalize also the interpretation logic within the data representation. Thus the web-based semantic markup languages provide the representation of semantics metadata, formal implications, restrictions etc.

The semiotic triangle describes an interpretation of semantic markup languages. In the semiotic triangle a sign invokes a concept. The concept in turn identifies an abstract or concrete thing in the world (Guarino et al. 2009). The formalized semantics is designed to be used for representing a data fragment's potential usage. The metadata captures part of the meaning of data (Antoniou et al. 2008). This formalization enables data reusability, machine-readability, inference mechanisms and semantic interoperability (Gómez-Pérez 2010).

Formalisms for Representing Semantics

Semantics formalisms describe the metadata as machine-readable formal semantics (knowledge representation paradigm). Semantic networks, frame-based logics, and description logics can be mentioned as most common existing formalisms. (Hitzler et al. 2008)

Semantic Networks describe data entities as nodes, which are connected among each other if a semantic relation exists (Fensel et al. 2003). Each of these connections is labeled to express the pragmatic idea behind this link. But in semantic networks the labeled link has to be interpreted if the underlying semantic is important. A well-known example for semantic networks is the *Resource Description Framework* (RDF). (Hitzler et al. 2008)

In addition frame-based logics may be used, which represent each named object as a frame. Frames have data slots in which a property or attribute of the object is represented. Slots can have one or more values and furthermore these values may be pointers to other frames (Fensel et al. 2003). The extension of RDF, the RDF Schema (RDFS), is a frame-based layer extending the expressiveness of RDF.

Another semantic formalism is the so called *Description Logics*. These allow constructing more expressive semantics, in terms of quantitative (numeric) and qualitative (structural) limitations, formal implications and restrictions. Substantially description logics constitute fragments of first-order logic, restricted to a certain complexity class to allow the construction of a high expressive language (Hitzler et al. 2008). Using description logics the semantics is represented as a terminological box (TBox) and an assertional box (ABox). In the TBox abstract information for concepts are specified. Information assigned to a concepts hold for all individuals (ABox) of this concept, thus this knowledge describes general properties of concepts. In the ABox the described real world objects are represented as individuals (Gómez-Pérez et al. 2010). Description logics based semantics formalisms are e.g. the Web Ontology Language (OWL) and OWL2.

Semantics data representations consist of concepts, concept taxonomies, relationships or roles between concepts, and properties describing the concepts. Thus on the concept level mainly concept taxonomies are described. Therefore semantics data representations consisting of these components are called lightweight formal semantics.

On the other hand heavyweight formal semantics allow representing more formal implications. This enables to model restrictions on domain semantics by adding formal axioms, functions, rules, procedures and constraints to lightweight formal semantics (Gómez-Pérez et al. 2010).

There are important relations and implications between the knowledge components (concepts, roles, etc.) used to build the formal semantics, the formal semantics formalism, used to represent the components, and the language, used to implement the semantics data (Gómez-Pérez et al. 2010).

Semantics Visualizations

Semantics Visualization plays a key-role in enlightening various relationships between data entities. Furthermore the relationships enable to gather information and adopt knowledge. Semantically annotated data can be visualized with semantics visualization, commonly known as "Ontology visualizations". The following section gives an overview about existing visualization techniques for representing semantically enriched data.

TGVizTab (TouchGraph Visualizaiton Tab) is the *TouchGraph* (Alani 2003) visualization Technique in the Protégé (Noy e al. 2000) ontology management tool. It provides different level of details by choosing variable radius of visibility. The user can navigate through graph by visualizing the parts of the graph gradually. The users can also rotate the graph to see the graph from different perspectives. Furthermore, She can also switch the graph to hyperbolic tree. It offers the also personalization features, which allows the user to choose focal point, color for the nodes, fonts and visibility of nodes. The ontology is also presented as tree structure on the left (Class Browser). It is a desktop solution, which the favorite ontology management tool for the experts. It is does not allow the aspects like brows and editing in "one-single-view", role based editing and collaboration. The GUI and UE design is suitable for the experts and does not meet the needs of the average user.

OntoTrack (Liebig and Noppens 2004) is a browsing and editing "in-one-view" ontology authoring tool for OWL lite ontologies. It offers a user friendly Graphical User Interface (GUI), which allows the users navigation and manipulation of large ontologies. It offers also intuitive User Experience Design concepts e.g. miniature branches or selective detail views to handle and manipulate ontologies in one-view. The system is based on *SpaceTree* (Plaisant et al. 2002). It is desktop application and it is not available as a web-based solution. It supports the scalability issues but does not provide features like personalization, role based view and collaboration.

TM-Viewer (Topicmap Viewer) (Godehardt and Bhatti 2008) is topic map based ontology visualization tool. TM-Viewer offers fields or sectors, which can be extracted from the ontology. The concepts in each field are represented with specific icons, lines between the knowledge concepts represent the associations and the levels represent the abstraction level of the concepts (inner level show generic concepts). Furthermore, the graphical metaphor with special icons for each sector supports the user to recognize the concept and navigate through the map easily. *TM-Viewer* allows to user to personalize the GUI with the help of configuration file completely. The user can choose not only the color for sectors or association, but also change the icons. It is web-based solution, but it does not allow role based and collaborative ontology visualization.

The visualization of huge number of knowledge items e.g. more than 100 topics can overstress the user. That is why, TM-Viewer uses cluster concept to keep the visualization manageable for the users. According to the cluster concept all the topics, which have same sibling will be clustered as it is shown in fig. The History component helps the user to keep the track of their navigation through the Topicmap. (Godehardt and Bhatti 2008)

COE (Hayes et al. 2003) is an RDF/OWL ontology viewing, composing and editing tool built on top of the IHMC *CmapTools* concept mapping software suite. Concept maps provide a human-centered interface to display the structure, content, and scope of *an ontology*. Concept mapping software solutions are used in educational settings, training, and knowledge capturing.

COE uses concept maps to display, edit and compose OWL, in an integrated GUI combining Cmap display with concept search and cluster analysis. COE imports OWL/RDFS/RDF ontologies from XML files (or URIs using http) and displays them as a new concept map. Layout is automatic. Stored ontology Cmaps can be modified and archived using Cmap Tools.

CropCircles (Parsia et al. 2005 and Wang and Parsia 2006)is an ontology visualization which represents the class hierarchy tree as a set of concentric circles. *CropCirces* aims to provide users intuitions on the complexity of a given class hierarchy at glance. Nodes are given the appropriate space in order to guarantee enclosure of all the sub trees. If there is only one child, it is placed as a concentric circle to its parents, otherwise the child - circles are placed inside the parent node from the largest to the smallest.

In order to navigate the ontology structure, the user may click on a circle to highlight it and see a list of its immediate children on a selection pane. The selection pane can let the user drill down the class hierarchy level-by-level and it also support user browsing history. The user may also select which top level nodes to show in the visualization.

Jambalaya (Storey et al. 2001) is a visualization plug-in for the Protégé ontology tool (Noy et al-2000) that uses the *SHriMP* (Simple Hierarchical Multi-Perspective) 2D visualization technique to visualize regular Protégé and OWL knowledge-bases. *SHriMP* is a domain-independent visualization technique designed to enhance how people browse and explore complex information spaces.

SHriMP uses a nested graph view and the concept of nested interchangeable views. It provides a set of tools including several node presentation styles, configuration of display properties and different overview styles.

OntoRama (Eklund 2002 and Eklund et al. 2002) is a RDF browser used for browsing the structure of an ontology with a hyperbolic – type visualization.

The hyperbolic visualization is motivated mainly by two arguments. Firstly an order of magnitude more nodes of a tree can be rendered in the same display space and secondly the focus of attention is maintained on the central vertex and its neighborhood. This means that the hyperbolic view is particularly useful for hierarchical diagrams with large numbers of leaves and branches and where neighborhood relationships are meaningful.

Unfortunately, *Ontorama* currently does not support "forest structures", which are sub-hierarchies neither directly nor indirectly connected to the root. It uses cloning of nodes that are related to more than one node, in order to avoid cases where the links become cluttered. It can support different relation types. Apart from the hyperbolic view, it also offers a windows explorer – like tree view.

OntoSpere3D (Bosca et al. 2005) is a *Protégé plug-in* for ontologies navigation and inspection using a 3-dimensional hyper-space where information is presented on a 3D view-port enriched by several visual cues (as the color or the size of visualized entities).

OntoSphere proposes a node – link tree type visualization that uses three different ontology views in order to provide overview and details according to the user needs. The OntoSphere3D user interface is quite simple; mouse centered, and supports scene manipulation through rotation, panning and zoom. It is strongly bound to the "one hand" interaction paradigm, allowing to browse the ontology as well as to update it, or to add new concepts and relations. Ontology elements are represented as follows: concepts are shown as spheres, instances are depicted as cubes, literals are rendered as cylinders and the relationships between entities are symbolized by arrowed lines where the arrow itself is constituted by a cone.

User interface features direct manipulation operations such as zooming, rotating, and translating objects in order to provide an efficient and intuitive interaction with the ontology model being designed. Since the tool aims at tackling space allocation issues the visualization strategy exploits dynamic collapsing mechanisms and different views, at different granularities, for granting a constant navigability of the rendered model.

Concepts and instances within scenes are click-able with the following outcomes: (1) Left clicks perform a focusing operation, shifting the currently visualized scene to a more detailed view, i.e. clicking on a concept in the tree view leads to a detailed view of such a concept. (2) Central clicks are used to expand collapsed elements. The actual behavior of the central click is slightly different from scene to scene: in the Main Scene it simply expands a concept replacing it with his children; in the Tree View expands a collapsed sub tree and collapses the others; in the Concept Focus Scene when clicking the central concept it shows/hides its children. (3) Right clicks, instead, open a contextual menu offering a set of alternatives dependent on the current scenes and the element properties.

When between 2 concepts in a scene occurs more than a single relation and a single line represents them all, no relation label is explicitly reported and arrow-head cones can be clickable as well. In that case the cone is depicted in white and left clicking onto it lists such relations.

A certain degree of scene personalization in terms of sizes of graphical components, distances between them and colors is supported through a proper option panel that is evocable by a button localized in the sx panel of the plugin.

Furthermore logical views can be defined on this hyper-space in order to easily manage interface complexity when the represented data gets huge, and thousands of concepts and/or relations must be effectively visualized.

The *3D Hyperbolic Tree* visualization was created for web site visualization but has been used as a file browser as well (Munzner 1997 and Munzner 1998).

It presents a tree in the 3D hyperbolic space in order to achieve greater information density. The nodes of the tree are placed at a hemisphere of a sphere. The graph structure in 3D hyperbolic space shows a large neighborhood around a node of interest. This also allows for quick, fluid changes of the focus point. Additionally, it offers animated transitions when changing the node on-focus.

IsaViz (Pietriga 2001) is a visual environment for browsing and authoring RDF ontologies represented as directed graphs.

It presents a 2D user interface allowing smooth zooming and navigation in the graph. Graphs are visualized using ellipses, boxes and arcs between them. The nodes are class and instance nodes and property values (ellipses and rectangles respectively), with properties represented as the edges linking these nodes.

IsaViz enables user import ontologies of RDF/XML, Notation 3 and N-Triple formats and export of RDF/XML, Notation 3 and N-Triple export, but also SVG and PNG formats.

OntoViz (Sintek 2003) is a Protégé (Noy 2000) visualization plug-in using the GraphViz library to create a very simple 2D graph visualization method.

The ontology structure is presented as a 2D graph with the capability for each class to present, apart from the name, its properties and inheritance and role relations. The user can pick a set of classes or instances to visualize part of an ontology. The instances are displayed in different color.

It is possible for the user to choose which ontology features will be displayed (for example slot and slot edges), as well as prune parts of the ontology from the "config" Panel on the left. Right-clicking on the graph allows the user to zoom – in or zoom – out

Grokker is a system to display the knowledge maps. It offers graphical representation for the search results or a file search. It uses a graphical metaphor for documents, clusters and category circles. The size of cluster and category circle shows the number of contained documents i.e. larger category circles contain more documents or results. The right panel offers further details about search results and allows the users to create own working list or tag to del.icio.us. The left panel offers filtering mechanism by date or domain and search within shown map. It is web-based solution and offers a user friendly and easy to GUI and UE design concept. It does not support role based, aspect oriented and collaboration aspects.

Kartoo is a search engine, which displays the search results with topographical interface. It displays the results closer to each other, if they have close relationship. The keywords show the relationship between the search results. The users can also click the keywords to navigate through the map. *Kartoo* uses different icons as graphical metaphor for different type of results e.g. documents, website etc. On the left side, all the topics are listed and serve as additional view of all displayed results. Furthermore, the description and thumbnail of the results are shown by roll over on the left side. It offers a user friendly and easy to GUI and UE design concept and it is a web-base solution. It does not support role based, aspect oriented and collaboration aspects.

Webbrain allows the visualization of the search results and organization of information. The organization in the *Webbrain* is associative instead of hierarchical. The users can organize the information by defining the association between the information items. The information in *Webbrain* is thoughts and they can be all type of documents like website, word or pdf-files. When the user chooses one thought, then it moves to the center and related thoughts to the selected thought branching out around it. The Company "The brain" offers different versions "Personal Brain (Desktop)", "*Webbrain* (Web)" and "Enterprise Knowledge Platform". The enterprise solution allows the collaboration as well. It offers a very easy and intuitive GUI and UE concept. It can also be used as *Mindmap*.

A more recent approach for visualizing complex semantics and ontologies is the *SemaVis* visualization technology. (Nazemi et al. 2011) *SemaVis* provide a more comprehensive view on heterogeneous semantics structures and uses several visualization techniques as described in previous chapters for graphically presenting semantic data. The main goal of *SemaVis* is to provide a core-technology for heterogeneous semantic data, different users and user groups and support heterogeneous tasks. Therefore a three layered model was developed, based on the model of Card et al. (Card et al. 1999), to provide a fine granular adaptation at different levels of abstractions. *SemaVis* subdivides the visualization layer into the layers *Semantics*, *Layout* and *Presentation*. With its modular characteristics several visualization techniques can be chosen while working with the visualization to present different views on the same data.

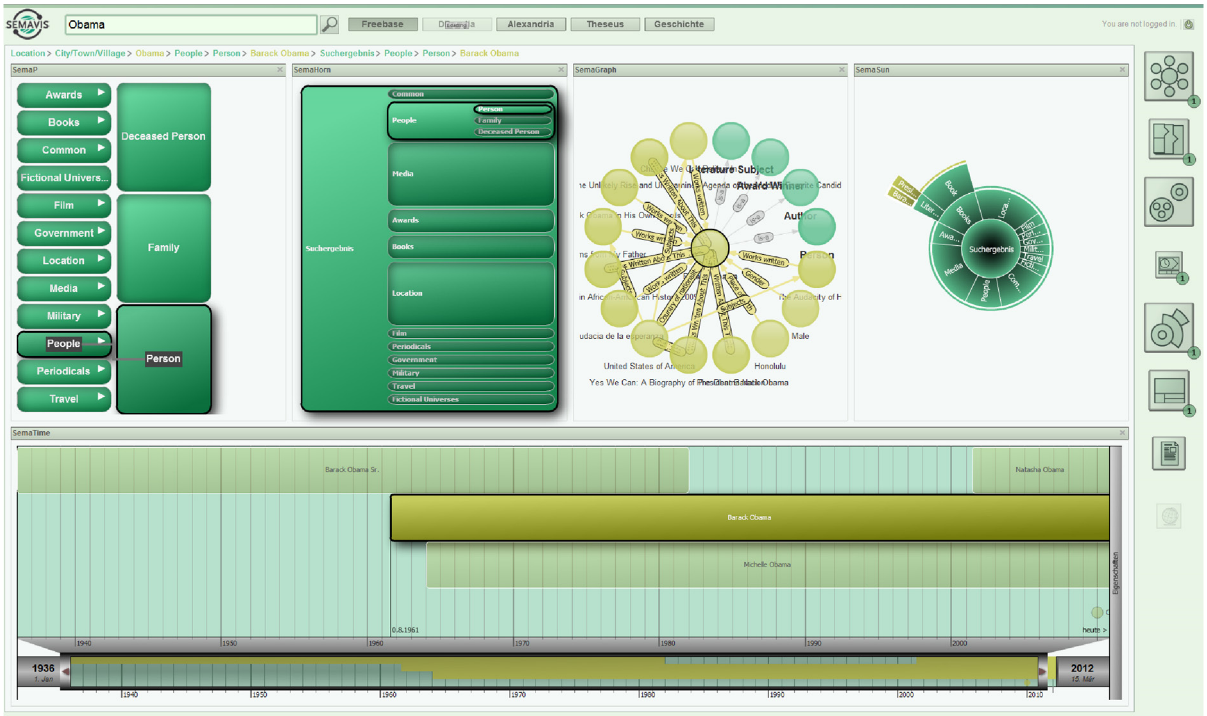
With an integrated *Visualization Cockpit* (Nazemi et al. 2010) the vies can be combined to solve different visualization tasks, e.g. exploring knowledge, comparing data structures etc. (Nazemi et al. 2011)

VISUALIZATION OF SOCIAL DATA AS SEMANTIC INFORMATION

The involvement of citizens, their opinions, discussions etc. in the policy modeling domain plays an increasing role. The web provides mass amounts of social data, which can be used to identify problems and involve the citizens' opinions in the policy creation process. The masses of information are very difficult to handle. Everyday new opinions, discussions etc. and there with new data are available. In FUPOL various technologies faces this challenge from different point of views. The crawling of data, the extraction of topics-of-relevance (hot-topics) and their causal relationships are investigated in FUPOL. From the users' point of view, the visualization of that data would provide an efficient way to acquire knowledge, e.g. for identifying problems and impacts of policies.

FUPOL provides therefore a visualization model that applies a top-down explorative metaphor for gathering knowledge in problem identification, impact analysis on social (subjective) level etc. The top-down approach integrates various overview visualizations, which first give an overview of topics in categorical, temporal and geographical aspects and provide further a faceting to reduce the information amount on relevant aspects. On the visualization level "details-on-demand" and graph-based visualizations provide a comprehensible view on the information relationships. With various visualization techniques the level of detail may prove fine granular or textual information. A

Figure 5. Visualization Cockpit of SemaVis (Nazemi et al. 2011)



model of data analysis for (data-based) adaptation provides an adaptable and adaptive multi-visualization view on the data. This approach enables the detection of policy related issues easier (more time efficient).

The benefit is to apply quantitative data analysis and visual mapping mechanisms in the domain of policy modeling to bridge the gap between masses of information entities (instances) and users' task. Therefore the quantity and attributes of the underlying data are analyzed and visualized in combined multi-visualization user interfaces. The analysis provides further a data adaptive visual representations, which may be integrated with further adaptation rules. The here proposed top-down (overview to detail) visualization cockpit provides another scientific value that is not investigated so far in context of social data for policy modeling.

This section describes an exemplary conceptual design for social data visualization based on semantics. In order to get the most suitable solution when analyzing social network data the appropriate visualization needs to fulfill some informational requirements. This means the visualization needs the capability to communicate available information as the result of the analysis process to the user interacting with the system. Depending on the available context information of the interaction (e.g. the actual task of the user, the political question to be solved) the visualization may also support the visual highlight of the relevant information artifacts. In this step of the conceptualization of the social web visualization predominantly the fulfillment of the identified informational requirements will be addressed. Further an overview-to-detail approach with combined semantics and quantitative visualizations will be introduced by investigating the FUPOL social data ontology. We propose in this chapter a solution to gather the semantic information in different levels of visual abstraction and provide therewith a new integrated approach of user experience with social data. In particular, the domain of policy modeling and the requirements on the policy process act as foundation.

Informational Requirements

The first and most obvious requirement is the illustration of the structural information or social groups, which predominantly is given by the relations between the nodes. Structures of the social network can depict interesting direct or indirect relations leading or promising to a specific result. Furthermore the nature of the connection structure provides informative bases where groups of people or topic-related documents occur and illustrate their impact to the neighbored nodes. Thus this structural requirement also represents social groups and social relations. To name some, these structures may be cliques, clustered cliques per clustering index, paths and different graph patterns.

In contrast to the structural information concerning multiple nodes and their intermediate relations, the second informational requirement corresponds to the position of a single node within the network. The social position describes for example the influence of a single user to other community members.

The third informational requirement describes the nodes itself. A user interacting with the social web visualization has always to be aware of the type of node which is connected with other entities. The most common node type may be a single person or a group of persons, which interacts in a social community. But also the type of posting or way of participating may be the information of interest.

To ensure the user may interpret the information type, glyphs are used to represent the nodes content type. In social web the nodes' content may be e.g. a single person, a group of persons, text, files, configuration files, audio, video or statements/messages. This can be indicated by icons or visual variables e.g. color or shape.

Due to the fact that the informational requirement information referee may vary in general (variable), but for a concrete situation or view of interest is a defined fix subset, the visualization concept needs to be designed appropriately. There are two types of information referee: the node references a concrete thing which can be pointed to (e.g. a person, a group of persons, a street, a building) and a non-physical object (e.g. an opinion, a topic) which can only be described textually but holds for multiple nodes (in terms of persons/actors).

The first type of information referee will be depicted in an information panel metaphor which displays the associated information like name, textual information and location. Users are able to interact and interpret this type of presentation, and the detailed information does not disturb the general interaction and navigation process within the visualization. Thus this visual metaphor corresponds to Shneidermans visual information seeking mantra: (1) overview first, then (2) zoom and (3) filter and present (4) details on demand. To ensure this mantra, details on demand will be displayed the information panel only if a user hovers over the visual element with the mouse; or if the user does an active interaction like a click on it.

Thus the second type of information referee will be visualized as a statistical distribution using a well-known visualization type, the pie chart. This pie chart is positioned directly behind the person, group of person or document discussing about the topics, like depicted in the subsequent picture. Due to their closeness this statistical information is directly associated to the node (law of closure, Gestalt psychology).

The social position represents the influence of a person or a group of persons to others. In addition here the social position will also be interpreted as the influence of a document, a statement, etc. with regard to the readers or the opinion generation process. Therefore this measure can be visually represented so that users are able to interpret the relations in a correct manner.

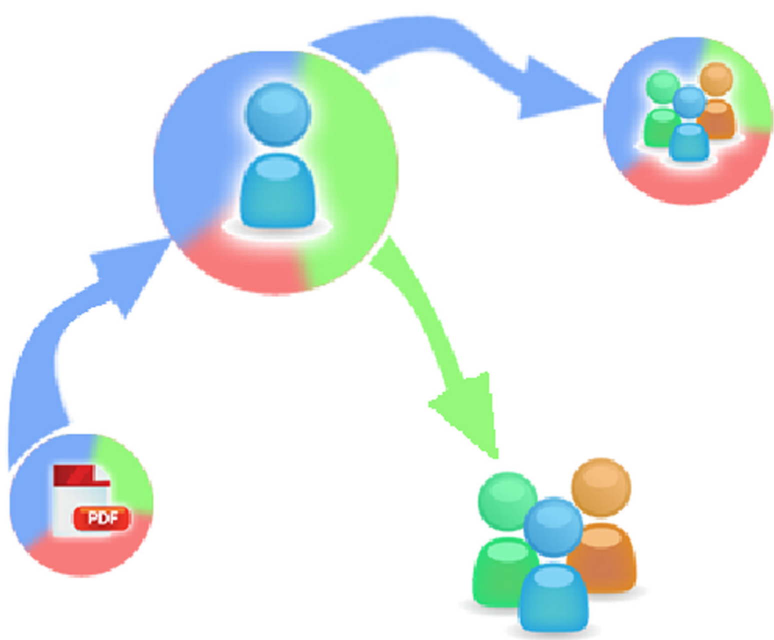
Social positions or social influences can be measured with two different measures. The first is a kind of measurement concerning only one element, which means this node has the power of x which is higher/lower than the power of y of another node. The second type is the resulting effect of the social position to other nodes, for example statements of a person influence others behavior/opinion with an intensity of value z. Due to the fact that the effect of social positions or social influence always relates to two or more nodes, usually in a directed way with a specific intensity this informational requirement will be visually represented using the edges between the entities. In contrast, social position or influence measures concerning only one node will be represented per size of the visual element itself.

Information attributes are important for the meaningful interpretation of the social data. The requirement information attributes summarize time stamps and trends (evolution/progress) and the consequential actuality of the information. In addition, geographic data, e.g. in which country a person lives, is addressed by this requirement. In order to visually meet the powerfulness of these information attributes, different visual layouts will be presented in this section.

The standard view on a social network is a graph view or linked network between the nodes. The nodes differ in their information type, so documents, movies, and persons build up the whole network structure.

To investigate the time stamps and time trends of the social data a timeline-based visualization is used. Here a stacked graph is appropriate for

Figure 6. Example of visualizing influencing relations



trend visualizations (e.g. topic evolutions) or a timeline-based bar chart if the data entries have the attributes of start and end times. In the subsequent picture an example is given for a stacked graph visualization presenting topic evolutions. The same concept can be applied to topics discussed by groups or single persons, like depicted in the subsequent picture.

Visualization of Social Data as Semantic Information

The work of the visualizations in context of social data is to provide a sufficient tool for identifying:

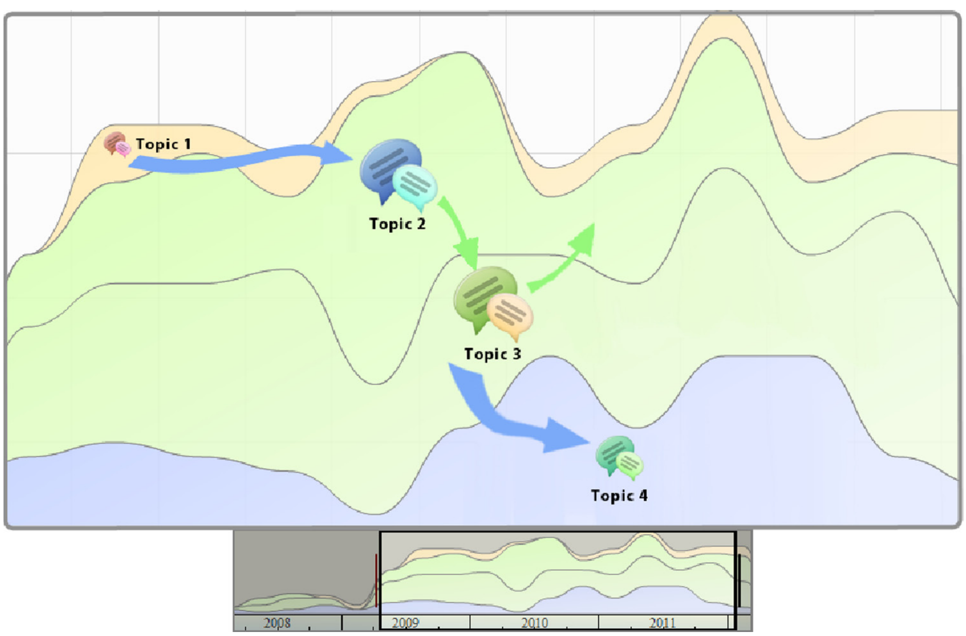
- Opinion-makers, opinion-leaders and influencing persons.
- Topics-of-interest (hot topics) in context of political discussions.
- Geographical influenced areas in context of identified topics.
- Temporal spread of hot topics.
- Relevance of hot topics after a policy implementation (impact).
- Further relevant aspects of social data analysis in context of policy modeling.

Formal Semantic Description of Social Data

The formalization of the crawled social web data is provided in FUPOL as a light-weight ontological representation. The technologies provide feature extractions based on statistical models. The extracted features are then formalized in a semantic relationship model, based on SIOC and FOAF, whereas FUPOL-specific classes are enhancing the ontology (see FUPOL Ontology).

Although the ontology provides a formalized and accessible way of the masses of social data, the problem still remains that only a low hierarchy is provided with masses on instances in each concept, e.g. the class "Topic" may contain a large number of topics. This is in particular a challenge

Figure 7. Topic evolution in a social web visualization



for web-based visualizations and a transparent and comprehensible on the data. To face this problem following approaches will be developed:

- Overview visualization of the ontology as a temporal, geographical or/and categorical spread.
- Details-on-Demand visualization on graph-based structures.
- Combined visualization of the Overview+Detail in a multi-visualization user interface.
- Use of the described visualization primitives in the various levels of visual representation.

Overview Visualization

The main challenge of visualizing the social data is the masses of instances in the described semantic representation. We have elaborated two ideas of partner technologies to face this problem on the data level, but beside a solution reducing the amount of instances per class/concept, the challenge of visualizing a mass amount of data still remains. An adequate way of facing this challenge on the visualization-level is the appliance of Shneiderman's Information Seeking Mantra (Shneiderman 1996). Shneiderman proposed a three-level seeking mantra containing the following steps: overview first, zoom and filter then details-on-demand. In the context of visualizing the social information the overview aspect plays a key role. In particular, we identify in context of social data visualization three main views on this information-level:

- Overview on categorical level,
- Overview on temporal level,
- Overview on geographical level.

The levels of overview visualizations are not distinct and can be combined to view on different information aspects.

Overview on Categorical-Level

The thematic arrangement enables a visual overview definition of "categories-of-interest", whereas all are some part of information are visualized interactively. We apply in this context two main visualization types to visualize the computed relevance and the result of a quantitative analysis on the user request. The different informational requirements are then visualized on the presentation level by using the visual variables. The size of a graphical entity will provide quantitative information whereas the relevance is visualized by their color.

We provide as the first categorical visualization an hierarchical treemap that uses the thematic hierarchy of the ontology as one visual indicator, the relevance of the topics as another visual indicator and the size as a third indicator for providing an overview of a topic on categorical level.

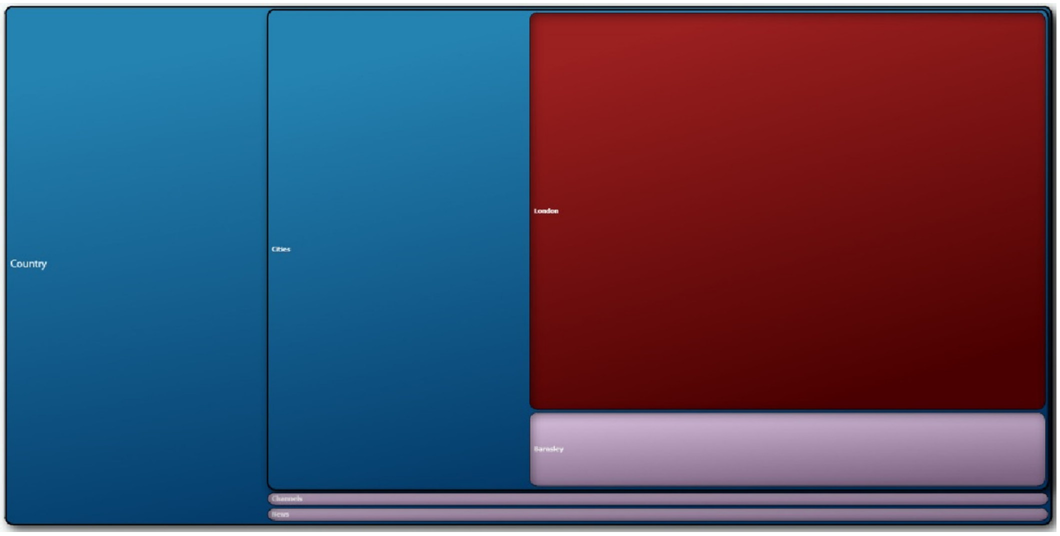
The following figure illustrates a very simple example of the described view. The parameters are abstracted to highest level. The hierarchy is simplified visualized as an overlapping (superimposing) and integrating spatial spaces. The size is illustrating the quantity and the color the relevance, as shown in Figure 8.

In contrast to that very simple visual view, a graph-based layout will be integrated that targets on the same information values. Therefore the size of circle will be used as the indicator for the quantity of information in one category, the hierarchy will be displayed as smaller integrated circles, and the color will be used for the computed relevance. We are dismissing any semantic relationships in this view, to not confuse the user with too many information.

Overview on Temporal Level

Another way of visualizing an overview of the whole spectrum of information is the consideration of the temporal attributes. With visualizing the temporal overview and providing a faceting in

Figure 8. Simplified abstract illustration of the hierarchical treemap (own development)



time another dimension of the data is investigated. We propose that the temporal view is the most beneficial way to:

- View the trend of upcoming social opinions.
- Interacting with and filtering semantic data for topic-of-relevance based on time.

Here we propose the use of a stacked graph with the using the following informational requirements on the information dimensions:

- **Size:** Quantity of topics, terms or extracted features.
- **Color:** Relevance based on the computed relevance.
- **X-Axis:** Temporal spread.

Overview on Geographical Level

The overview aspect can be investigated from the geographical point of view too. This visual representation here investigates the geographical spread. This visualization is beneficial when the data can be assigned to geographical attributes and the temporal space is set to a specific value, e.g. today's topics-of-relevance in Barnsley. The quantitative value cannot be considered in this view. It visualizes the geographical spread of topics on a map. The color indicates the topic related to a hot-area and this area can be named by the identified topics.

Details-on-Demand Visualization on Graph-Based Structures

The next step after the overview is a more detailed view with relational information. Therefore the existing graph-based visualizations will be extended to visualize the dependencies between actors and topics, between actors themselves and between topics themselves. This step can be done after a refinement on the overview visualization or based on a specific search that contains a comprehensible number of entities.

We propose to use a force-directed visual graph algorithm with quantitative analysis for this issue. In this case the size of a circle indicates the number of entities, the color the relevance, the size of entities the number and/or relevance of a topic or actor himself and the relations the semantic relationship design in the FUPOL social data ontology.

The detailed visualizations can further provide more information by requesting more details on demand. For example in the figure, we see one actor with a greater size than the others. With this information we can assume that this actor is an opinion maker, because either he has many postings or the postings are read by many people (regarding to the underlying data and goal). By clicking on this actor the visual representation will first give more information about him and further provide detailed information (as far as available) about the person.

In all the steps we have defined different visualization types that are appropriate to meet the informational requirements from the social data part of view. One of the main contributions in this task is that the visual change of the steps from overview to details and vice versa is recognized and appropriate visualizations are provided in combined user interfaces.

The categorical, temporal, and geographical view can be combined in various ways to provide a sufficient view on the social data. One promising way to provide a fruitful way for visualizing the different informational requirements of social data and statistical data respectively is the juxtaposed orchestration of visualizations (Nazemi et al. 2010) as illustrated exemplary in Figure 10.

Figure 9. Graph-based detail visualization (own development)

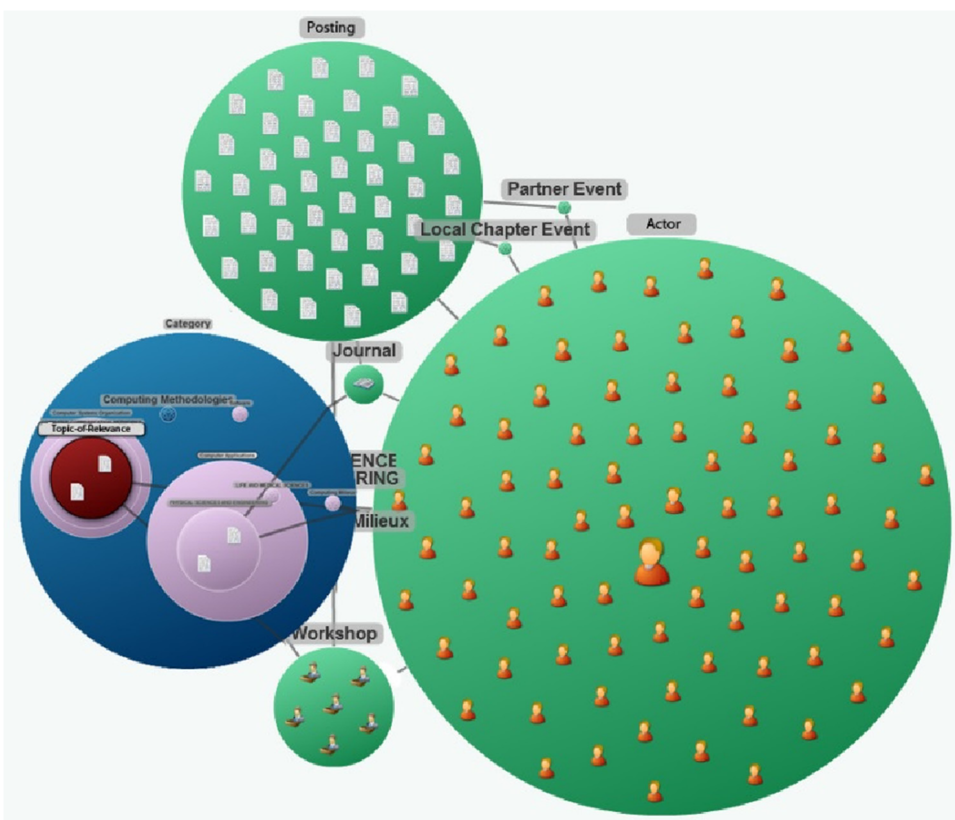


Figure 10. Visualization orchestration

FUTURE RESEARCH DIRECTIONS

Social media, linked-data and data on web provide masses of information that may help to find out the intentions of citizens and ease the decision making process in the entire policy creation process. The access to the information on web is getting more and more difficult, due to the growing amount of data. One promising way to illustrate the data and interact with them are information visualization tools, whereas commonly information visualization is either too simple (pie and bar charts) or too complex (analyst tools from visual analytics). This aspect is a great challenge in particular for the policy and eGovernment community. Thus, although the visualization techniques provide promising ways to interact with knowledge, they are not really accepted in that domain. Future research topics should cover more the human factor and the human-centered design of visualization systems. In particular adaptive and intelligent visualizations, incorporating machine learning algorithms for recognizing and modeling users' behavior, tasks to be solved, and the underlying data will play an essential role for the acceptance of complex visual systems.

CONCLUSION

The policy creation and modeling cycle is characterized by the need of information in particular to have a valid foundation for making decisions. In this context various kinds of information plays a key-role: social-data enable mining opinions and identify opinion leaders, while ground truth statistical data helps to identify policy indicators and therewith enables monitoring, validating or identifying policy needs and changes. The main problem in this context is the mass amount of data, especially on web. One promising way to face the challenge of "big data" is the use of interactive visual representations. Information visualization provides here various techniques for visualization.

But not only the visualizations themselves play an important role, the way how human interacts with the visualizations and the way how data are modeled, transformed and enriched gains more and more intention.

We introduced in this chapter information visualization as a solution for enabling the human information access to the heterogeneous data that are necessary during the policy modeling process. Therefore we first started to identify the steps of policy design, where information visualizations are required based on an established policy life-cycle model. Thereafter a foundational overview of information visualization was given, investigating beside visualization techniques, the entire spectrum of data to visualization. In this context data and interaction methods were introduced too. We concluded the chapter with an conceptual example of visualizing social data in the domain of policy modeling.

REFERENCES

Abello, J., van Ham, F., & Krishnan, N. (2006). ASK-Graphview: A Large Scale Graph Visualization System. IEEE Transactions on Visualizations and Computer Graphics, 12(5).

Ahlberg, C., & Shneiderman, B.Visual Information Seeking. (1994). *Tight Coupling Of Dynamic Query Filters with Starfield Displays*. Boston, MA: ACM Transactions on Human Factors in Computing Systems.

Alani, H. (2003). TGVizTab: An ontology visualisation extension for Protégé. In *Proceedings of Knowledge Capture (K-Cap'03), Workshop on Visualization Information in Knowledge Engineering*. Sanibel Island, FL: K-Cap.

Amar, R. A., & Stasko, J. T. (2005). Knowledge Precepts For Design and Evaluation of Information Visualizations. *IEEE Transactions on Visualization and Computer Graphics*, *11*(4). doi:10.1109/TVCG.2005.63 PMID:16138553

Andrews, K. (1995). Information Visualization in the Harmony Internet Browser. In *Proceedings of IEEE Information Visualization*. Atlanta, GA: IEEE.

Antoniou, G., & Van Harmelen, F. (2008). *A Semantic Web Primer*. MIT Press.

Balzer, M., & Deussen, O. (2007). Level-of-Detail Visualization of Clustered Graph Layouts. In *Proceedings of the Asia-Pacific Symposium on Visualization* (APVIS). APVIS.

Bartram, L. (1997). Can motion increase user interface bandwidth in complex systems?. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*. IEEE.

Beard, D. V., & Walker, J. Q. (1990). Navigational techniques to improve the display of large two-dimensional space. Behaviour and Information Technology, 9(6).

Bederson, B. B., Hollan, J. D., & Perlin, K. et al. (1996). Pad++: A zoomable graphical sketchpad for exploring alternate interface physics. *Journal of Visual Languages and Computing*, 7.

Bekavac, B. (1999). *Hypertextgerechte Suche und Orientierung im WWW: Ein Ansatz unter Berücksichtigung hypertextspezifischer Struktur- und Kontextinformation*. (Dissertation). Universität Konstanz, Fachgruppe Informatik und Informationswissenschaft.

Bendix, F., Kosara, R., & Hauser, H. (2005). Visual Analysis Tool for Categorical Data – Parallel Sets. In *Proceedings of IEEE Symposium on Information Visualization* (INFOVIS). IEEE.

Berners-Lee, T. (2010). *Weaving the Web: The original Design and Ultimate Destiny of the World Wide Web by its Inventor*. HarperBusiness.

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web. *Scientific American*, *284*(5), 34. doi:10.1038/scientificamerican0501-34 PMID:11396337

Bertin, J. (1983). *Semiology of Graphics: Diagrams, Networks, Maps* (W. J. Berg, Trans.). Madison, WI: University of Wisconsin Press.

Bosca, A., Bomino, D., & Pellegrino, P. (2005). OntoSphere: More than a 3D ontology visualization tool. In *Proceedings of SWAP, the 2nd Italian Semantic Web Workshop.* CEUR. Retrieved from http://ceur-ws.org/Vol-166/70.pdf

Boulos, K. (2003). The use of interactive graphical maps for browsing medical/health internet information resources. *Internal Journal of Health Geographics, 2* (1).

Brandes, U., Gaertler, M., & Wagner, D. (2003). Lecture Notes in Computer Science: Vol. 2832: *Experiments on Graph Clustering Algorithms*. Springer Verlag.

Bryan, D., & Gershman, A. (2000). The Aquarium: A Novel User Interface Metaphor For Large Online Stores. In *Proceedings 11th International Workshop in Database and Expert Systems Applications*. Academic Press.

Card, S. K., Mackinlay, J. D., & Shneiderman, B. (Eds.). (1999). *Readings in Information Visualization - Using Vision to Think*. San Francisco, CA: Morgan Kaufman Publishers, Inc.

Card, S. K., Robertson, G. G., & Mackinlay, J. D. (1991). The Information Visualizer, an information workspace. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 181—186). ACM Press.

Chen, C. (2004). *Information Visualization: Beyond the Horizon*. London: Springer.

Chen, C., Zhu, W., Tomaszewski, B., & MacEachren, A. (2007). Tracing Conceptual and Geospatial Diffusion of Knowledge. In D. Schuler (Ed.), Online Communities And Social Computing, (HCII 2007), (pp. 265-274). Springer-Verlag.

Eick, S. G., & Karr, A. F. (2000). *Visual Scalability* (Technical Report Number 106). National Institute of Statistical Sciences.

Eklund, P. (2002). Visual Displays for Browsing RDF Documents. In *Proceedings of the 7th Australasian Document Computing Symposium*. Sydney, Australia: Academic Press.

Eklund, P. W., Roberts, N., & Green, S.P. (2002). OntoRama: Browsing an RDF Ontology using a Hyperbolic-like Browser. In *Proceedings of the First International Symposium on CyberWorlds* (CW2002). IEEE Press.

Elmquist, N., Dragicevic, P., & Fekete, J.-D. (2003). Rolling the Dice: Multidimensional Visual Exploration using Scatterplot Matrix Navigation. IEEE Transactions on Visualization and Computer Graphics, 14(6).

Fua, Y., Ward, M. O., & Rundensteiner, E. A. (1999). Hierarchical Parallel Coordinates For The Exploration Of Large Datasets. In *Proceeding of the 10th IEEE Visualization Conference* (Vis ’99). IEEE.

Furnas, G. W. (1981). *The FISHEYE View: A New Look at Structured Files*. Murray Hill, NJ: AT&T Bell Laboratories.

Furnas, G. W. (1986). Generalized Fisheye views. In *Proceedings Human Factors in Computing Systems Conference*. Boston, MA: Academic Press.

Garland, K. (1994). *Mr Beck's Underground Map – A history*. Middlesex, UK: Capital Transport Publishing.

Gershon, N., & Eick, S. G. (1995). Visualisation's new tack: Making sense of information. *IEEE Spectrum*. doi:10.1109/6.469330

Godehardt, E., & Bhatti, N. (2008). Using Topic Maps for Visually Exploring Various Data Sources in a Web-based Environment. In *Proceedings of Scaling Topic Maps: Third International Conference on Topic Map Research and Applications* (TMRA 2007). Berlin: Springer.

Gómez-Pérez, A., Fernández-López, M., & Corcho, O. (2010). *Ontological Engineering*. Springer.

Guarino, N., Oberle, D., & Staab, S. (2009). What Is an Ontology?. In Handbook on Ontologies, International Handbooks on Information Systems, (pp. 1-17). Springer.

Hao, M. C., Dayal, U., Keim, D. A., & Schreck, T. (2005). Importance-Driven Visualization Layouts For Large Time-Series Data. In *Proceedings of IEEE Symposium On Information Visualization*. Minneapolis, MN: IEEE.

Havre, S., Hetzler, B., & Nowell, L. (2000). ThemeRiver: Visualizing Theme Changes Over Time. In *Proceedings of IEEE Symposium on Information Visualization*. IEEE.

Hayes, P., Saavedra, R., & Reichherzer, T. (2003). *A Collaborative Development Environment for Ontologies*. CODE.

Hearst, M. A. (1999). User Interfaces and Visualization. In B. Ribeiro-Neto (Ed.), *Modern Information Retrieval*. Addison Wesley Longman.

Heer, J., Card, S., & Landay, J. (2005). *Prefuse – A toolkit for interactive information visualization*. Portland, Oregon: Conference On Human Factors In Computing. doi:10.1145/1054972.1055031

Henry, N., Fekete, J.-D., & McGuffin, M. J. (2007). NodeTrix: A Hybrid Visualization of Social Networks. *IEEE Transactions on Visualization and Computer Graphics*, *13*(6). doi:10.1109/TVCG.2007.70582 PMID:17968078

Herman, I., Melancon, G., & Marshall, M. S. (2000). Graph Visualization and Navigation in Information Visualization: A Survey. IEEE Transactions on Visualzation and Computer Graphics, 6, 24-43.

Hitzler, P., Krötzsch, M., Rudolph, S., & Sure, Y. (2008). *Semantic Web: Grundlagen*. Springer.

Hochheiser, H., & Shneiderman, B. (2004). *Dynamic Query Tools for Time Series Data sets: Timebox Widgets for Interactive exploration* (Vol. 3). Information Visualization, Palgrave MacMillan.

Holten, D. (2006). Hierarchical Edge Bundles. *IEEE Transactions on Visualization and Computer Graphics*, *12*(5). doi:10.1109/TVCG.2006.147 PMID:17080795

Jog, N. K., & Shneiderman, B. (1995). Starfield Information Visualization with Interactive Smooth Zooming. In *Proceedings of the 3rd IFIP 2.6 Working Conference on Visual Database Systems Conference*. Lausanne, Switzerland: IFIP.

Keim, D., Kohlhammer, J., May, T., & James, J. T. (2006). Event Summary of the Workshop on Visual Analytics. *Computers & Graphics, 30*(2).

Keim, D., Kohlhammer, J., Ellis, G., & Mansmann, F. (Eds.). (2010). *Mastering the Information Age: Solving Problems with Visual Analytics*. Eurographics Association.

Keim, D. A. (2000). Designing Pixel-oriented Visualization Techniques: Theory and Applications. *IEEE Transactions on Visualization and Computer Graphics*, *6*(1), 59–78. doi:10.1109/2945.841121

Keim, D. A. (2002). Information Visualization and Visual Data Mining. *IEEE Transactions on Visualization and Computer Graphics*, *8*(1). doi:10.1109/2945.981847

Keim, D. A., & Kriegel, H. P. (1996). Visualization Techniques For Mining Large Databases – A Comparison. *IEEE Transactions on Knowledge and Data Engineering*, *8*(6). doi:10.1109/69.553159

Keim, D. A., Kriegel, H.P., & Ankerst, M. (n.d.). Recursive Pattern: A Technique for Visualizing Very Large Amounts Of Data. In *Proceedings of Visualization 95*. Academic Press.

Kohlhammer, J. (2005). *Knowledge Representation for Decision-Centered Visualization*. Herdecke, Germany: CGA-Verlag.

Kohlhammer, J., Nazemi, K., Ruppert, T., & Burkhardt, D. (2012). Towards Visualization in Policy Modeling. IEEE *Journal of Computer Graphics and Applications*.

Kosara, R. (2001). *Semantic Depth of Field – Using Blur for Focus + Context Visualization*. (PhD Thesis). Institute of Computer Graphics and Algorithms, Vienna University of Technology, Vienna, Austria.

Liebig, T., & Noppens, O. (2004). OntoTrack: Combining Browsing and Editing with Reasoning and Explaining for OWL Lite Ontologies. In *Proceedings of the 3rd International Semantic Web Conference ISWC 2004*. Hiroshima, Japan: Academic Press.

Lin, J., Keogh, E., & Lonardi, S. (2005). Visualizing and Discovering Non-Trivial Patterns In Large Time Series Databases. *Palgrave Macmillan Journal On Information Visualization*, *4*(2), 61–82. doi:10.1057/palgrave.ivs.9500089

Macintosh, A. (2004). Characterizing E-Participation in Policy-Making. In *Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences* (HICSS'04), (Vol. 5). IEEE Computer Society.

Mansmann, F., Keim, D. A., North, S. C., Rexroad, B., & Sheleheda, D. (2007). Visual Analysis of Network Traffic for Resource Plannung, Interactive Monitoring and Interpretation of Security Threats. IEEE Transactions on Visualization and Computer Graphics, 13(6).

May, T., & Kohlhammer, J. (2008). Towards closing the analysis gap: Visual Generation of Decision Supporting Schemes from Raw Data. *Eurographics IEEE-VGTC Symposium on Visualization (EuroVis), 27*(3).

May, T., Steiger, M., Davey, J., & Kohlhammer, J. (2012). Using Signposts for Navigation in Large Graphs. *Computer Graphics Forum*, *31*(2-3), 985–994.

Modjeska, D. (n.d.). Navigation in electronic worlds: A Research Review (Technical Report). *Computer Systems Research Group, University of Toronto*.

Müller, W., Nocke, T., & Schumann, H. (2006). Enhancing the Visualization Process with Principal Component Analysis to Support the Exploration of Trends. In Proceedings of the APVIS 2006. APVIS.

Munzner, T. (1997). H3: Laying Out Large Directed Graphs in 3D Hyperbolic Space. In *Proceedings of the 1997 IEEE Symposium on Information Visualization*. Phoenix, AZ: IEEE.

Munzner, T. (1998). Exploring Large Graphs in 3D Hyperbolic Space. *IEEE Computer Graphics and Applications*, *18*(4), 18–23. doi:10.1109/38.689657

Nazemi, K., Breyer, M., Burkhardt, D., & Fellner, D. W. (2010). Visualization Cockpit: Orchestration of Multiple Visualizations for Knowledge-Exploration. *International Journal of Advanced Corporate Learning*, *3*(4).

Nazemi, K., Breyer, M., Forster, J., Burkhardt, D., & Kuijper, A. (2011). Interacting with Semantics: A User-Centered Visualization Adaptation Based on Semantics Data. In Human Interface and the Management of Information: Part I: Interacting with Information. Berlin: Springer.

Nazemi, K., Stab, C., & Kuijper, A. (2011). A Reference Model for Adaptive Visualization Systems. In J. A. Jacko (Ed.), *Human-Computer Interaction: Part I: Design and Development Approaches* (pp. 480–489). Berlin: Springer. doi:10.1007/978-3-642-21602-2_52

Nightingale, F. (1857). *Mortality of the British Army*. London, UK: Harrison and Sons.

North, C. L. (2000). *A User Interface for Coordinating Visualizations based on Relational Schemata: Snap-Together Visualization*. (PhD Dissertation). University of Maryland Computer Science Dept.

Noy, N. F., Fergerson, R. W., & Musen, M. A. (2000). The knowledge model of Protege-2000: Combining interoperability and flexibility. In *Proceedings of 2nd International Conference on Knowledge Engineering and Knowledge Management* (EKAW 2000). Juanles-Pins, France: EKAW.

Parsia, B., Wang, T., & Goldbeck, J. (2005). Visualizing Web Ontologies with CropCircles. In *Proceedings of the 4th International Semantic Web Conference*. Academic Press.

Pietriga, E. (2001). *IsaViz*. Retrieved from http://www.w3.org/2001/11/IsaViz/

Plaisant, C., Grosjean, J., & Bederson, B. B. (2002). SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation. In *Proceedings of IEEE Symposium on Information Visualization*. Boston: IEEE.

Playfair, W. (2005). The Commercial and Political Atlas and Statistical Breviary. Cambridge University Press.

Proulx, et al. (2006). Avian Flu Case Study with nSpace and GeoTime. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology*. Baltimore, MD: IEEE.

Rao, R., & Card, S. K. (1994). The Table Lens, Merging graphical and symbolic Representations in an interactive Focus+Context visualization for Tabular Information. In *Proceedings Human Factors in Computing Systems*. Boston, MA: ACM.

Reingold, E.M., & Tilford, , T. (1981). Drawing of Trees. *IEEE Transactions on Software Engineering*, *7*(2), 223–228. doi:10.1109/TSE.1981.234519

Robertson, G. C., Card, S. K., & Mackinlay, J. D. (1993). Information Visualization Using 3-D Interactive Animation. *Communications of the ACM*, *36*(4). doi:10.1145/255950.153577

Robertson, G. G., & Mackinlay, J. D. (1993). The Document Lens. In Proceedings *of 6th ACM Symposium on User Interface Software and Technology*. ACM Press.

Robertson, G. G., Mackinlay, J. D., & Card, S. K. (1991). Cone Trees: animated 3D visualizations of hierarchical information. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Reaching through Technology*, (pp. 189 – 194). ACM.

Sarkar, M., & Brown, M. H. (1992). Graphical Fisheye Views of Graphs, Digital SRC Research Report 84a. In *Proceedings of ACM CHI'92*. ACM.

Schreck, T., Bernard, J., Tekušová, T., & Kohlhammer, J. (2008). Visual Cluster Analysis of Trajectory Data with Interactive Kohonen Maps. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology* (VAST). IEEE.

Shneiderman, B. (1994). Dynamic Queries for Visual Information Seeking. *IEEE Software*, *11*(6), 70–77. doi:10.1109/52.329404

Shneiderman, B. (1996). The Eyes Have It: A Task By Data Type Taxonomy For Information Visualization. IEEE Visual Languages, 336-343.

Shneiderman, B. (1999). *Dynamic Queries, Starfield Displays and the path to Spotfire, Report Human-Computer Interaction Lab*. University of Maryland.

Shneiderman, B. (2007). *Designing the user interface: Strategies for effective human-computer-interaction* (4th ed.). Addison-Wesley.

Siirtola, H. (2000) Direct Manipulation Of Parallel Coordinates. In *Proceedings of the International Conference On Information Visualization*. Academic Press.

Siirtola, H. (2007). *Interactive Visualization of Multidimensional Data. Dissertations in Interactive Technology, Number 7*. University of Tampere.

Sintek, M. (2003). *Ontoviz tab: Visualizing Protégé ontologies in online resource*. Retrieved from http://protege.stanford.edu/plugins/ontoviz/ontoviz.html

Storey, M.-A., Mussen, M., Silva, J., Best, C., Ernst, N., Fergerson, R., & Noy, N. (2001). Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in Protégé. In *Proceedings of Workshop on Interactive Tools for Knowledge Capture* (K-CAP-2001). Victoria, Canada: K-CAP.

Tekusova, T., & Kohlhammer, J. (2007). Applying Animation to the Visual Analysis of Financial Time-Dependent Data. In *Proceedings of the IEEE Conference on Information Visualization* (IV07). Zurich, Switzerland: IEEE.

Thomas, J., & Cook, K. (2005). *Illuminating the Path: Research and Development Agenda for Visual Analytics*. IEEE Press.

Tufte, E. R. (1983). *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press.

van Ham, F., & Perer, A. (2009). Search, Show Context, Expand on Demand: Supporting Large Graph Exploration with Degree-of-Interest. *IEEE Transactions on Visualization and Computer Graphics*, *15*(6), 953–960. doi:10.1109/TVCG.2009.108 PMID:19834159

van Wijk, J. J. (2005). The Value of Visualization. In *Proceedings of the IEEE Visualization*. IEEE.

van Wijk, J. J., & van de Wetering, H. (1999). Cushion Treemaps: Visualization of Hierarchical Information. In *Proceedings of IEEE Symposium on Information Visualization* (INFOVIS'99). San Francisco, CA: IEEE.

Voinea, S. L., Telea, A., & Chaudron, M. (2005). Version-Centric Visualization Of Code Evolution. In *Proceedings of the IEEE Eurographics Symposium on Visualization* (EuroVis'05). IEEE Computer Society Press.

Wang, T., & Parsia, B. (2006). Cropcircles: Topology sensitive visualization of owl class hierarchies. In *Proceedings of International Semantic Web Conference*. Academic Press.

Ware, C. (2013). *Information Visualization - Perception for Design* (3rd ed.). Morgan Kaufmann.

Wijffel, A. M., Vliegen, van Wijk, & van der Linden. (2008). Generating Color Palettes using Intuitive Parameters. In *Proceedings of IEEE-VGTC Symposium on Visualization* (EUROVIS). IEEE.

ADDITIONAL READING

Brusilovsky, P. wook Ahn, J.; Dumitriu, T. & Yudelson, M. (2006) Adaptive Knowledge-Based Visualization for Accessing Educational Examples Information Visualization, 2006. IV 2006. Tenth International Conference on, 2006, 142-150

Card, S. K., Mackinlay, J. D., & Shneiderman, B. (1999) Readings in Information Visualization: Using Vision to Think, Morgan Kaufmann, 1999.

Card, S. K., Robertson, G. G., & Mackinlay, J. D. (1991). The Information Visualizer, an information workspace. In CHI'91: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 181—186). ACM Press.

Kohlhammer, J., Nazemi, K., Ruppert, T., & Burkhardt, D. (2012): Towards Visualization in Policy Modeling. In Journal of Computer Graphics and Applications, IEEE, Sept.-Oct.2012.

Kohlhammer, J. Proff, D.U. Wiener, A. (2013) Visual Business Analytics: Effektiver Zugang zu Daten und Informationen. dPunkt Verlag.

Macintosh, A. Characterizing E-Participation in Policy-Making. In Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04), Vol. 5. IEEE Computer Society, Washington, DC, USA, 2004.

Nazemi, K., Stab, C., & Kuijper, A. (2011). A Reference Model for Adaptive Visualization Systems. In J. A. Jacko (Ed.), *Human-Computer Interaction: Part I: Design and Development Approaches* (pp. 480–489). Berlin, Heidelberg, New York: Springer. doi:10.1007/978-3-642-21602-2_52

Shneiderman, B. (1996) The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations, *VL, 1996*, 336-343

Ward, M., Grinstein, G., & Keim, D. (2010). *Interactive Data Visualizations Foundations*. Natick, Massachusetts: Techniques, and Applications A K Peters, Ltd.

Ware, C. (2013). *Information Visualization - Perception for Design* (3rd ed.). Morgan Kaufmann.

KEY TERMS AND DEFINITIONS

Adaptation: Adaptation in human-computer interfaces is the automatic and system-driven changes on content, structure, and presentation of system-behavior that involve some form of learning, inference, or decision making based on one or many influencing factors to support users.

Adaptive Visualizations: Adaptive visualizations are interactive systems that adapt autonomously the visual variables, visual structure, visualization method, or the composition of them by involving some form of learning, inference, or decision making based on one or many influencing factors like users' behavior or data characteristics to amplify cognition and enable a more efficient information acquisition.

Information Visualization: It is the interactive visual representation of data to amplify cognition and support information and knowledge acquisition.

Semantics Visualization: Semantics visualizations are computer-aided interactive visualizations for effective exploratory search, knowledge domain understanding, and decision making based on semantics.

Semantics: Semantic can be defined as data with meaningful relations of at least two information or data entities, to provide in best case a disambiguated meaning.

SemaVis: SemaVis is an adaptive semantics visualization technology developed by Fraunhofer Institute for Computer Graphics Research.

Visual Analytics: Visual Analytics is the interactive coupling of data analysis and information visualization to provide insights and knowledge.

ENDNOTES

[1] http://data.gov.uk/linked-data

[2] http://epp.eurostat.ec.europa.eu