Fachhochschule Dortmund
University of Applied Sciences and Arts

Fachhochschule Köln
Cologne University of Applied Sciences

From Number Crunching to Visual Analytics

Tools for Effective Decision Making

Project Paper

Karolin Stefani

Student Number: 7083726 (11088489)

Studies: Master Wirtschaftsinformatik (Verbund)

Lecturer: Prof. Dr. Kristian Fischer

June 30, 2013

Contents

List of Figures

List of Tables

"The purpose of computing is insight, not numbers."

— Richard Hamming (1915–1998)

1 Introduction

Information has become one of the most critical factors for companies to maintain their competitiveness in times of rapidly changing market factors and customer needs. Having timely access to the right information enables companies to gain a competitive advantage. (Gabriel et al. 2009, p. 4)
Very often a huge amount of data that is only insufficiently used resides in the databases of the companies' operational information systems. Thus, valuable information is not accessible for decision makers at a timely and appropriate manner. (Gabriel et al. 2009, p. 4)
Additionally, there is more and more data relevant for decisions that exists in data sources which are external to the companies, e.g. social networks data, documents on the web or sensor data. Thomas & Cook (2005, p. 24) show that the amount of data produced and stored, regardless whether externally or internally, is growing rapidly at a rate of more than 30% a year. The challenge is to select the most relevant bits of information that are very often contained in only small parts of the overall information base.
The vast amount of data, distributed over multiple sources is the reason why there is an ongoing demand for appropriate access and analytical methods to make the information usable in decision making processes (Gabriel et al. 2009, p. 4).
Appropriate analysis techniques, visualization and interaction possibilities are essential for gaining meaningful insight from all the available data. Because of this immanent demand for analytical tools, the following project paper introduces different tools that can help decision makers in gaining insights from the data and making reasonable decisions. Next to traditional number crunching based tools, graph based tools and Formal Concept Analysis (FCA)[1] based tools, which belong to a rather new field of analytical research called Visual Analytics, will be introduced. In advance, the basics of analytical information systems, data types and visualization will be examined.

[1]Formal Concept Analysis is a mathematical method in order theory (see section 4.3.1)

2 Decision Support with Analytical Information Systems

2.1 Necessity and Architecture of Analytical Information Systems

Decision support is ubiquitous by nature (Hosack et al. 2012, p. 316), as decisions are made on all levels and areas of business (Gabriel et al. 2009, p. 4).
Problem solving processes require decision making. The nature of problems whether structured, semi-structured or unstructured influences the decision making tasks, related to these problems. The more structured a decision task is the higher the number of stable parameters and relationships a decision maker has to consider. Contrary, the more unstructured a decision task is the more unknown or shifting parameters and relationships exist. In both extremes information systems which deliver right access to the data should support the decision makers in effective decision making. (Hosack et al. 2012, p. 316)

The required information systems have to provide the necessary information in a suitable fashion regarding the situation, the persons and problems in the decision making stage. Those systems are called **Analytical Information Systems** or **Business Intelligence Systems**. (Gabriel et al. 2009, p. 4)
Many definitions of Business Intelligence (BI) exist in literature. Ghazanfari et al. (2011, p. 24ff.) build up a classification of BI definitions based on how BI is approached, differentiating between managerial, technical and system enabler approaches. For all approaches in common, they point out that the collection, analysis and distribution of information is the core of BI and supporting the strategic decision making process is its objective.
A rather technically oriented definition, which emphasizes the importance of decision support, can be found in Chaudhuri et al. (2011, p. 88):

> "BUSINESS INTELLIGENCE (BI) SOFTWARE is a collection of decision support technologies for the enterprise aimed at enabling knowledge workers such as executives, managers, and analysts to make better and faster decisions."

Chamoni & Gluchowski (2006, p. 11) emphasize that the term *Business Intelligence* has been excessively used in the past and it is imminent to dilute. Because of that, they prefer to use the term *Analytical Information Systems*. In the following paragraphs both terms will be used synonymously.

Figure 1 shows the basic architecture of a business intelligence system. It becomes apparent that a BI environment is typically composed of a set of different components ranging from the data sources (external and/or internal), data extraction, transformation and loading functionalities (i.e. ETL), a data staging layer i.e. a data warehouse, analytical data preparation components to numerous front-end

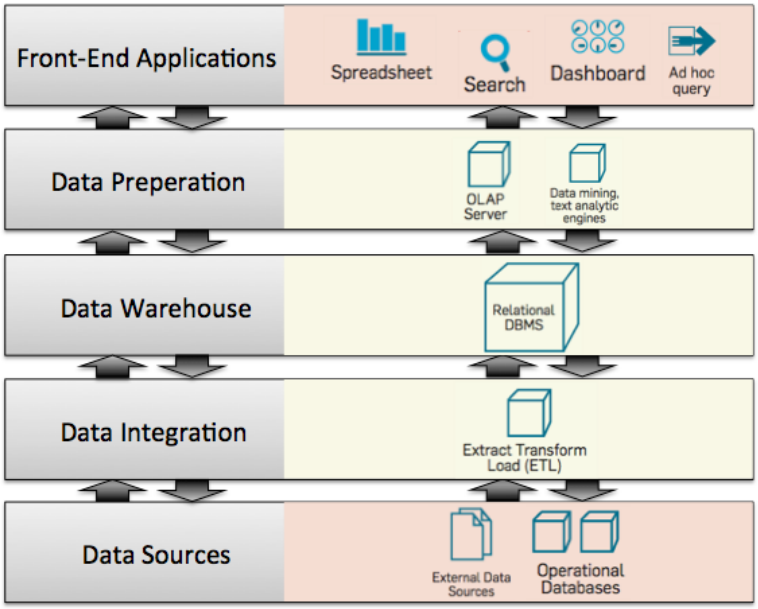

Figure 1: Typical business intelligence architecture (Bange 2006; Chaudhuri et al. 2011, adapted)

applications through which users access the data. The data flows through the system from the data sources to the analytical front-end tools, and is transformed according to the analytical information needs. The data can also flow the other way round when users enter new data in the front-end tools. This will be shortly covered in section 3.3. In this project paper the focus lies on the upper two layers including data preparation e.g. the OLAP engine and additional analysis methods residing in this layer, and especially the front-end layer with its possible visualizations and applications. For details about the lower layers please refer to Gabriel et al. (2009) or Chaudhuri et al. (2011).

2.2 Historical Development of Decision Support Systems

Decision Support Systems (DSS) have been a paradigm of interest from almost the beginning of information systems research. Hosack et al. (2012) give a comprehensive overview of the historical development of DSS showing that DSS research and practices have evolved along with the development of information technology:

In the 1960s, when mainframe computers were used for decision support, decision making was a centralized process mainly characterized through batch processing techniques. Mainframe computers were large, expensive and space-filling systems with less capacity than a mobile device today. Tabulating data and creating simple reports were costly calculation tasks. Data was processed in a fixed way with a fixed set of reports. Applying changes to the reports required the intervention of programming experts. Thus, the systems could not be used to conduct flexible scenario analysis in the increasingly dynamic business environments. Arnold (2013)

calls these systems which were rather static and required specialist intermediation *first-generation analytics*.
The emergence of smaller, less expensive microcomputers in the early to mid-1970s led to more distributed computing environments and decentralized decision making processes. Individual departments in organizations purchased their own systems. Non-programmed, rather unstructured decisions as opposed to the programmed routine problems solved in mainframe systems became the focus of DSS research. The introduction of the IBM PC in 1981 led to a common integration of microcomputers as business computing resources. The formerly dumb terminals connected to mainframe computers were replaced by microcomputers being able to run computer programs independently of larger corporate computing resources. The development of spreadsheet applications, like Lotus 1-2-3, enabled a manager to interactively work with the data and prepare reports without a need for intervention by a programming specialist. Arnold (2013) calls these systems *second-generation analytics*. The new paradigm led to the development of small scale decision models mainly supporting individual and functional area decision making.
The distributed computing environment and relative ease of individual decision making support were a problem for decision making processes at a corporate level as conflicting analyses of problem situations could be the result. Thus, the focus of research turned to group decision making processes. The importance of group decision support systems (GDSS) was accompanied by the evolving availability of local area networks. Research on GDSS evolved into research on computer supported cooperative work (CSCW).
In the 1990s, DSS research focused on data warehouses for the integration and harmonization of data from different sources. The Internet and web explosion of the late 1990s and early 2000s further increased the need for viable and flexible decision support systems. Decision support as a business service was expected to be always available. Furthermore, decision making responsibilities were pushed downward the organizational hierarchy to lower level employees or even outward to consumers or stakeholders.
Arnold (2013) calls the newest generation of analytical systems *third-generation analytics* in which the appliance of mathematical methods and intuitive and interactive interfaces should completely eliminate the need for expert intermediation. Moreover, in third-generation systems users are not confronted with the mass of underlying primary data as it is automatically transformed with analytical methods before it is visualized.

2.3 The Nature of Data To Be Analyzed

To make use of all the available data it has to be brought into a suitable data representation format. The quality of the underlying data representations has a severe impact on the quality of the visualizations used in analytical tools. This makes it necessary to transform the data into a representation that fits the different analytical tasks. (Thomas & Cook 2005, p. 9)
Before transforming the data, it is important to understand that raw data is present

in multiple data types, with different characteristics which have an influence on how the data is represented in analytical information systems and on how it can be used in the decision making process. Chaudhuri et al. (2011, p. 96) show that BI tasks often require to search different types of data.

Perspective	**Description**
Data type	Numeric or non-numeric or both
Level of structure	Structured, semi-structured or unstructured
Geospatial characteristics	Association with location or region information
Temporal characteristics	Temporal association, discrete or continuous

Table 1: Data characteristics from multiple perspectives

Thomas & Cook (2005, p. 108) try to characterize data by looking at it from multiple perspectives (see table 1). The perspectives can be described as follows:

Data Type Regarding the data type, the data can be divided into two larger categories: numeric and non-numeric data.
Numeric data mainly originates from other computer systems e.g. ERP systems and it is quantitative by nature. Depending on the analytical tool used or on the purpose of the analysis numeric data may or may not be manipulated or re-represented before visualization takes place. For a long time, numeric data have been the focus in analytics and many data representation techniques have been developed for this type of data. (Thomas & Cook 2005, p. 108f.)
The distinction whether data is numeric or not mainly originates from the fact that traditionally thousands records of numeric data are produced during business operations and stored in the operational information systems (see section 3.1).
All other types of data can be regarded as **non-numeric data**. These include for example textual data, image, audio or video data. However, categorizing all these types as non-numeric data in a binary categorization does not imply that the same processing techniques can be applied to all of these types of data, e.g. video and audio data need to be encoded and tagged with meta data for searching and interpretation purposes.
Nevertheless, these types have in common that they can include or be made of linguistic data. Examining linguistic data means paying attention to its different level of structure: phonological, morphological, syntactic, semantic and pragmatic level. With proper representations, all levels could support visualization. However, it is still a big challenge to create useful data representations of linguistic data that can be used for analytic purposes. (Thomas & Cook 2005, p. 109ff.)
According to Chaudhuri et al. (2011, p. 96) more and more valuable BI relevant data is available in the form of texts (e.g. E-mail, surveys, forums, blogs). It is also possible that numeric and non-numeric data is combined, e.g. with categorical data such as survey data (Thomas & Cook 2005, p. 113).

Level of Structure The level of structure ranges from well-structured, semi-structured to completely unstructured data. Numeric data is highly structured, textual data can be well-structured or not structured at all. E-mail is an example for semi-structured data as it contains well-structured elements and prosa text as well. Speaking of structure in this sense means the ability to automatically derive a structure from data and interpret it without human involvement. Unstructured does not mean that there is no structure at all, but only humans can interpret the data meaningfully. (Thomas & Cook 2005, p. 109)
With the evolvement of the Internet in the past decade managing semi- and unstructured data has become a focus of data management research, e.g. Florescu (2005) describes the problem of automatically processing semi-/unstructured data. Using metadata languages like XML, as proposed in Sperberg-McQueen (2005), can provide structural information on raw data.
Next to this rather syntactic perspective on structure, structure can also be viewed from a more semantically oriented perspective. Similar to XML, languages for describing the semantics of data have been developed e.g. OWL (Web Ontology Language). OWL can be used to describe data via classes, properties and relations, and thus enabling machine interpretability and automatic processing ability of data. The level of semantic structure determines how far machine processing can be applied to data without human involvement. (Breitman et al. 2006, p. 81) This is important as structure makes it possible to assign meaning to data and forms a basis for data representation. If no structure is available, it needs to be derived. Thus, non-structured data is transformed into structured data before it is analyzed.

Geospacial and Temporal Characterisics Geospacial and temporal characteristics can add additional valuable content to data analysis. Those characteristics compose information that is associated either with numeric data (e.g. seismic activity data) or non-numeric data (e.g. textual news report or transportation data). Adding geospacial and/or temporal information affects the types of possible analysis and also the data representation (e.g. raster grids and vectors for geospatial data). (Thomas & Cook 2005, p. 109)
As geospatial and temporal analytics will not be the focus within the following sections, please see Thomas & Cook (2005) for additional information.

2.4 Information Visualization in Analytics

Through visualization decision makers access data that need to be analyzed. According to Card et al. (1999, p. 6) decision making is one of the main goals of visualization next to explanation and discovery. Card et al. (1999, p. 9) define information visualization as:

> "The use of computer-supported, interactive, visual representations of abstract data to amplify cognition[2]."

[2]Cognition is the process of acquisition or use of knowledge (Card et al. 1999).

The authors use the term *abstract data* in their definition, as information visualization shows abstractions of raw data that have been transformed meaningfully. Regarding the handling of business data, they emphasize that visualizing non-physical data - like financial data, business information, collections of documents and abstract conceptions - is more challenging than visualizing data that allow obvious spatial mappings.

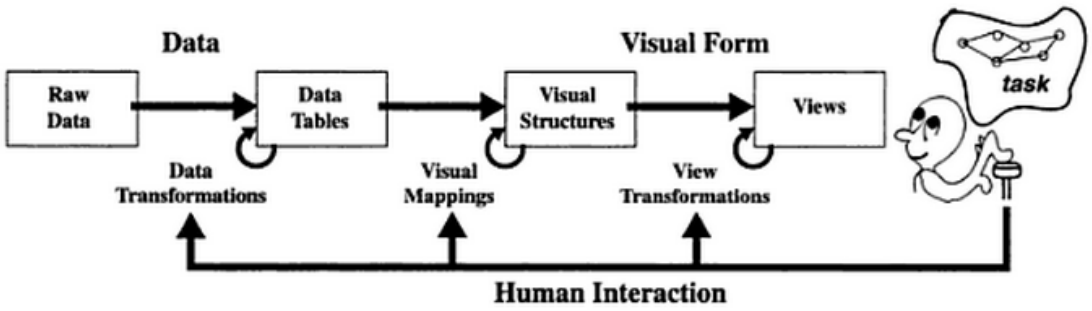

Figure 2: Reference model for visualization (Card et al. 1999)

Card et al. (1999, p. 9) propose a reference model for visualization as a basis for discussions about visualization systems (see fig. 2). A similar model that lays more emphasis on the data and the transformation steps is the data state model (or operator model) introduced by Chi & Riedl (1998). Both models are based on the so called visualization pipeline model which has been introduced by Haber & McNabb in 1990. The visualization pipeline is used to describe the different operations performed on data in the process of creating a visualization of the data.
In all models three stages can be identified: data transformations, visual mappings and view transformations. In the data transformation stage raw data is transformed into relations or set of relations (i.e. set of tuples) that are more structured and thus can be more easily visualized. The second step is the visual mapping stage, where characteristics of the transformed data are mapped to attributes of visual structures e.g. dimensions, transparency or color. In the last step, the view transformation, the visual structures are rendered for presentation. In this step, feedback mechanisms like rotation or scaling functionalities are added and made available to the analyst. (Card et al. 1999, p. 17ff.)
As shown in figure 2 an analyst can influence all stages of visualization through interaction and thus manipulate the view presented on the data.

3 Traditional Analytical Tools

3.1 Concept - Number Crunching

Traditional analytical tools are mainly based on number crunching, i.e. deriving information from numerical data. Many business relevant information is available

in form of numerical data or can be translated into measures, e.g. sales, budget, inventories, or revenues. Thus, for traditional tools data is stored in a measure-based and multidimensional structure. The combination of numerical data with different dimensions that are logical summarizations of data elements like products, segments, or places, transforms raw numbers into meaningful information as dimensions add context to each measure. The dimensions can be described by a set of attributes and hierarchies that provide additional information for analyzing the measures (see fig. 3). The aggregation of measures along the different dimensions, e.g. total sales by country and year, is another characteristic of the data model used in traditional analytics. (John et al. 2011, p. 43; Chaudhuri et al. 2011, p. 92)
As visualized in figure 3 and stressed by Chamoni & Gluchowski (2006, p. 15) these multidimensional data structures are often allegorized as cubes where the dimensions build the textualized edges. Each intersection of the dimensions represents the value of the measure relevant for this combination of dimensions. The overall volume of the cube represents all available measure values for the selected level of aggregation.

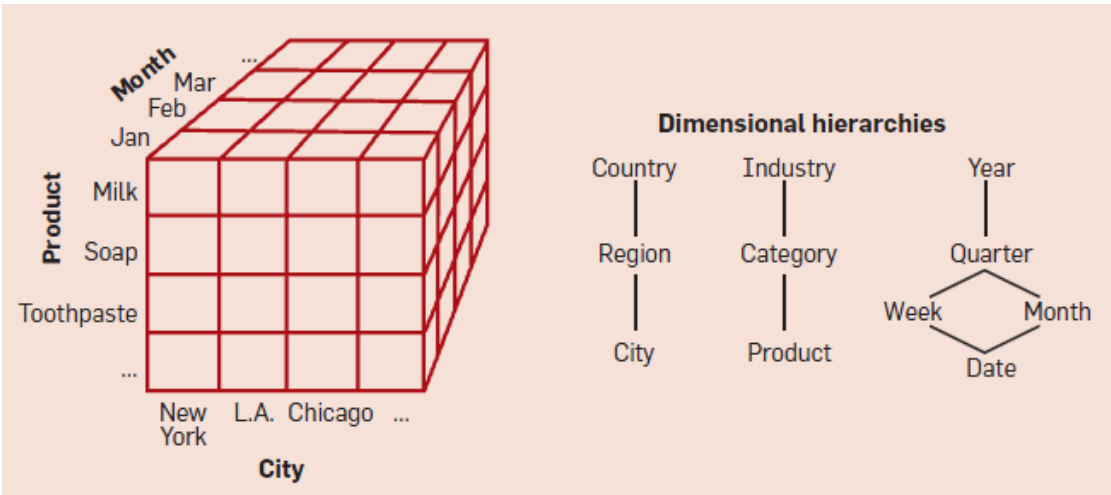

Figure 3: Multidimensional data (Chaudhuri et al. 2011)

Analytical engines called OLAP servers (Online Analytical Processing) are used to provide the multidimensional access on the data to the analytical tools and thus to the users. There are "classical" BI operations like pivoting, roll-up, drill-down, slicing and dicing that are enabled through OLAP engines. Those operations enable the user to navigate freely and interactively within the database. (Chaudhuri et al. 2011, p. 90; Chamoni et al. 2005, p. 18)
Despite the term OLAP is used by Chaudhuri et al. to describe the engine it is not only a calculation engine but also a common framework for building analytical information systems. It is mainly used for analyzing historical, consolidated data from a multidimensional perspective. Chamoni & Gluchowski (2006, p. 14) show that the possibility of multidimensional analysis is the main characteristic of

OLAP which distinguishes it from OLTP (Online Transaction Processing) used in operational information systems to support transaction-based business processes.

3.2 Visualization in Traditional BI Systems

The multidimensional view provided by the OLAP engine, influences how the data is mapped to visualizations and presented in the frontend tools (see visualization pipeline in section 2.4).

According to John et al. (2011, p. 52) two-dimensional tables, like shown in figure 4, are the most frequently used visualizations in traditional BI tools. One or more of the data model's dimensions are mapped to the horizontal and vertical axis. At the intersection of the axes the measures are displayed. The user can interact with the data by (ex-)changing the dimensions or elements shown in the axes. These interaction possibilities correspond with the pivoting, roll-up, slice or dice operations enabled by the OLAP engine, e.g. the user create a more aggregated view on the data by choosing node elements from within the dimension hierarchies. By navigating through the data and its dimensional levels, different questions can be answered that require different levels of data aggregation. Additionally, analyzing the data from multiple perspectives is possible by changing the dimensions and its order in the axes.

Sales (in €) per Store/Region (Quarter 1 2013)

	January 2013	February 2013	March 2013	Quarter 1 2013
Total Sales	**12.048.134,63**	**15.219.828,51**	**16.032.671,04**	**43.300.634,19**
Total Brandenburg	***2.916.099,31***	***3.719.655,92***	***3.564.948,48***	***10.200.703,70***
Store Cottbus	990.231,45	1.430.707,12	1.991.085,52	4.412.024,09
Store Potsdam	1.352.043,46	1.390.902,46	1.003.256,05	3.746.201,97
Store Frankfurt (Oder)	573.824,40	898.046,34	570.606,91	2.042.477,64
Total Sachsen	***3.309.511,12***	***4.474.076,05***	***4.472.766,78***	***12.256.353,95***
Store Dresden	1.476.283,37	1.992.811,87	1.489.062,87	4.958.158,11
Store Chemnitz	650.388,27	567.452,75	535.660,31	1.753.501,33
Store Leipzig	1.182.839,48	1.913.811,44	2.448.043,60	5.544.694,52
Total Sachsen-Anhalt	***1.970.420,06***	***2.298.536,04***	***2.097.009,97***	***6.365.966,06***
Store Magdeburg	839.291,39	880.828,29	697.708,27	2.417.827,96
Store Halle (Saale)	638.291,29	746.910,09	766.037,70	2.151.239,08
Store Naumburg	492.837,38	670.797,66	633.263,99	1.796.899,03
Total Thueringen	***1.779.810,73***	***2.587.165,26***	***3.595.832,98***	***7.962.808,98***
Store Erfurt	1.029.233,84	1.426.615,75	2.027.085,11	4.482.934,70
Store Gera	405.342,30	574.023,91	792.900,54	1.772.266,74
Store Weimar	345.234,59	586.525,61	775.847,34	1.707.607,53
Total Mecklenburg-Vorpommern	***2.072.293,41***	***2.140.395,24***	***2.302.112,84***	***6.514.801,49***
Store Rostock	993.782,23	1.033.514,88	1.516.806,50	3.544.103,61
Store Greifswald	683.923,43	553.801,34	338.934,30	1.576.659,07
Store Neubrandenburg	394.587,75	553.079,01	446.372,04	1.394.038,80

Figure 4: Data analysis with a two-dimensional table

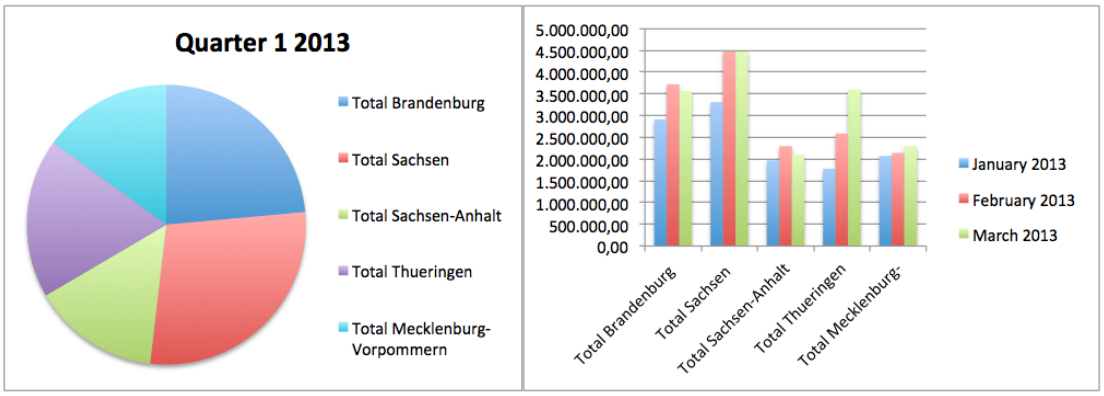

Figure 5: Pie chart and bar chart for data visualization

Sometimes, displaying information in table based visualizations can be overwhelming for the user who is not able to select the most relevant information from the whole pool of information. Fekete et al. (2008, p. 5) show examplarily how challenging or almost impossible it could be to derive valuable information from spreasheet data when answering complex questions without proper graphical visualizations.
Therefore, there are many visualizations in traditional analytical tools that illustrate the numerical data in graphics. This leads to higher attention, fosters data exploration and supports the user in gaining insight from the numerical data.
Typical non-table based visualizations in traditional BI tools are bar charts, pie charts, line diagrams and all varieties of theses types of diagrams. As with the table based visualizations, the characteristics of the multidimensional data model are mapped to the characteristics of the graphical diagrams, e.g. elements of a dimension are spread in a pie chart or form the bars in a bar chart. Furthermore, colors, transparency or other attributes can be used to categorize dimensional elements or the measures themselves. Depending on which diagrams are used and how the data model's dimensions and measures are mapped different information can be drawn from the visualizations, e.g. forecast the development of a measure by placing it in the data area of a time-associated line diagram, comparing a measure with regard to different dimension elements in a bar chart, or examining the share of different dimensional elements in a measure's sum value in a pie chart. This effect is shown in figure 5 where the data from figure 4 is visualized.

3.3 Application of Traditional BI Systems

Talking about applications requires to distinguish between the functional area of application and specific software applications. Depending on which functional area is supported different tools will be applied. According to Gluchowski & Chamoni (2006, p. 151) the scope of application for multidimensional analytical information systems comprises a wide range of functional areas and facets. An OLAP-based system can be applied everywhere where analytical tasks in organizations need to be solved. This ranges from a pure supply of professionals and executives with

information to offering a corpus of data for complex analyses, e.g. for calculations in market forecasting or investment decisions. Therefore, in virtually all business areas there is a requirement for analytical information systems.

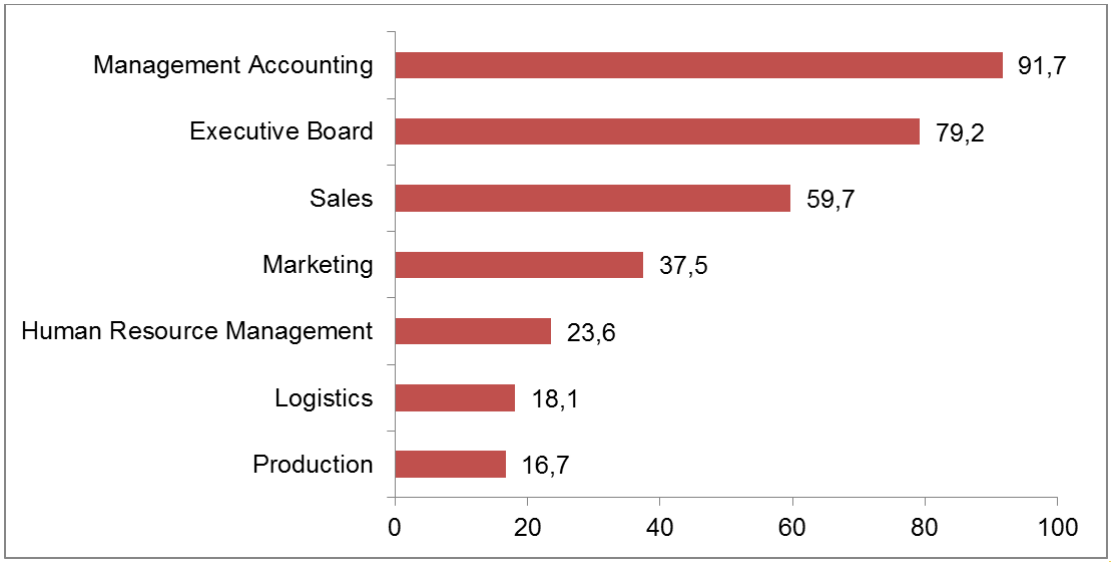

Figure 6: Applications of OLAP systems regarding functional areas (Gluchowski & Chamoni 2006)

As shown in figure 6, OLAP-based analytical information systems are mainly applied in management accounting and at the executive board's level followed by marketing and sales areas. In human resource management, logistics and production they still play a minor role.
Without regard to a special functional area, the main objective of analytical information systems is displaying and processing decision relevant information (see section 2.1). With this understanding Bange (2006, p. 97) suggests a classification of (software) applications for analytical OLAP-based information systems that distinguishes the applications according to its level of freedom in analysis for the user and the applications' complexity (see fig. 7).
The visualizations described in the previous section are integrated into the different types of applications.

With reference to Bange (2006, p. 98ff.) the classes of applications can be described as follows:

- **Cockpits and Scorecards** are used for a simplified and aggregated view on information and are published in enterprise portals or as stand-alone applications. They are often used to visualize key performance indicators and are popular tools at the executive board level.

- **Reporting** provides a static or dynamic view on the data according to the reports' definitions, formatting, and query possibilities. Very often table based views are combined with diagrams to visualize the data in reports.

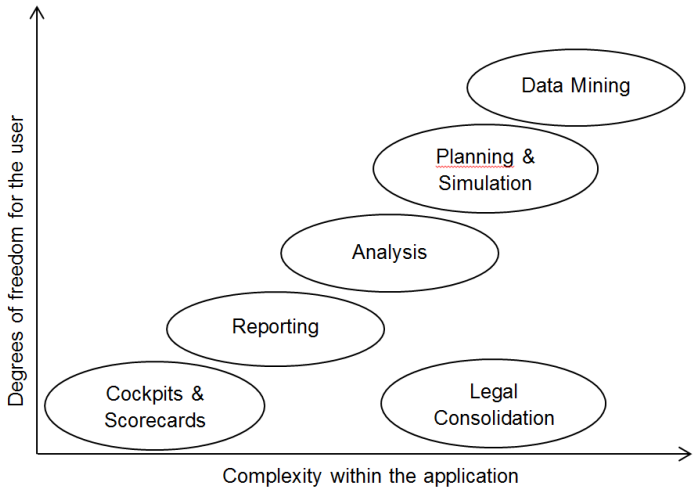

Figure 7: Classification of applications for analytical information systems (Bange 2006)

- **Analysis** extends reporting as it adds possibilities of navigating and restructuring the data, e.g. with additional calculation and simulation possibilities.
- **Planning & Simulation** applications offer functionalities for forecasting, planning and simulation. They differ from the previous application classes as they allow to access the data bi-directionally, i.e. they allow to read and write data. They are mainly used in management accounting but also in marketing, sales or human resource planning.
- **Legal Consolidation** applications are used to create a common view on corporations' financial data by integrating different accounting standards and chart of accounts. Due to the focus on finance the tools are mainly used in accounting and for information purposes of the executive level or external shareholders.
- **Data Mining** tools are used to analyze data with the purpose of discovering new structures and patterns in existing data by applying statistical methods, techniques of machine learning or artificial intelligence to the existing pool of data. Depending on where the data mining logic is implemented there could exist seperate data mining servers next to the OLAP engines (see fig. 1). Data mining applications are popular tools in marketing and sales analyses.

There is almost an infinite number of traditional, OLAP-based and/or data mining based BI software available at the market that can be assigned to one or more of the classes of applications.

4 Visual Analytics Tools

4.1 Concept - Visual Analytics

As already shown in section 1 there is a steadily increasing amount of data with different types and from multiple sources. Thomas & Cook belong to the pioneers who coined the term **Visual Analytics** to describe a new way of how these massive amounts of data could be consumed and analyzed. Next to the immense amounts of data, the often disparate, sometimes conflicting and dynamic nature of information that can be derived is a reason why Thomas & Cook (2005, p. 22) demand new methods of data analyses.
Additionally, Thomas & Cook (2005, p. 30) stress the importance of human judgment in the analysis process, as analytical reasoning on the available data is a key factor in gaining insight from the data by discovering unexpected relationships or revealing missing expected relationships . Lemieux (2011, p. 39) captures this idea by speaking of an "integrated approach combining visualization, human cognition, and data analysis".

Thomas & Cook (2005, p. 28) summarize the above mentioned key points in their definition of **Visual Analytics**:

> "Visual analytics is the science of analytical reasoning facilitated by interactive visual interfaces."

This definition reveals that Visual Analytics should rather be regarded as an autonomous research area than simply as a set of analytical tools. Figure 8 shows the different fields of research which simultaneously make up the components of Visual Analytics.

According to figure 8, fostering the analytical reasoning process could be seen as one of the main components of Visual Analytics and it is also one of its main purposes. This is realized by software that allows humans to maximize their capacity to perceive, understand and reason about complex and dynamic data situations. The underlying analytical reasoning process is an iterative process where an analyst iteratively collects and organizes information when advancing to judge about a question. Thomas & Cook (2005, p. 33f.) show that using interactive visualizations, rather than static visualizations, is essential in this iterative process, as they allow the user to gradually explore the data. They call this kind of dialog between an analyst and the data *analytic discourse*, in which the visual representation is simply the interface or view on the data.

As part of Visual Analytics different methods of computer analysis are used to transform data into representations that can be mapped to the necessary interactive visualizations (Lemieux 2011, p. 39). These methods are used to pre-process the data for first visualization but also process data in reaction to the user's interactions during the analytical reasoning process. The data that has been processed

Figure 8: The R&D agenda for Visual Analytics (Thomas & Cook 2005)

in reaction to a user's interaction is re-represented as visual representation and can be further analyzed. An example of such a method for computer analysis is Formal Concept Analysis (FCA) which is introduced in section 4.3.1.
In Visual Analytics data representation and transformation also occurs in lower layers of the analytical information system's architecture comparable to the ETL-process in traditional BI environments (see fig. 1). However, these steps of converting the different types of maybe conflicting and dynamic data will not be covered in this project paper.

Using tools that can be characterized as Visual Analytics tools do not eliminate the idea of number crunching as possible with traditional BI tools (see section 3.1). They rather allow another form of interaction with the data. Thus, the same visualizations as available in traditional BI tools, like e.g. bar charts, could also be used in Visual Analytics (Lemieux 2011, p. 39). As described by Thomas & Cook (2005, p. 69), the combinational use of visual representations and interactions to accelerate rapid insight into complex data is the key factor that distinguishes Visual Analytics tools from other analytical tools.

4.2 Graph Based Visual Analytics

4.2.1 Basics of Graphs and their Role in Visual Analytics

As mentioned in the previous section, the integration of computer analysis methods with visualization and interaction techniques to analyze large sets of data is the core of Visual Analytics. Von Landesberger et al. (2011, p. 1720) pick up this idea by taking a Visual Analytics perspective on the field of visual graph analysis, as graphs are popular data structures within Visual Analytics. They describe the

importance of applying graph algorithms as a pre-processing step and also a processing step in the interaction process to reveal new interesting aspects of data. This is visualized in figure 9 which shows the components of visual graph analysis. Next to representation and interaction it stresses the importance of algorithmic graph analysis and visualizes that an integrated approach is necessary for effective visual graph analysis.

The main aspect that distinguishes graph based analytics from traditional analytics is the importance of understanding structures in the graph or respectively in the underlying data. Thus, identifying connections between entities and clusters among them is the main focus of graph based analytics.

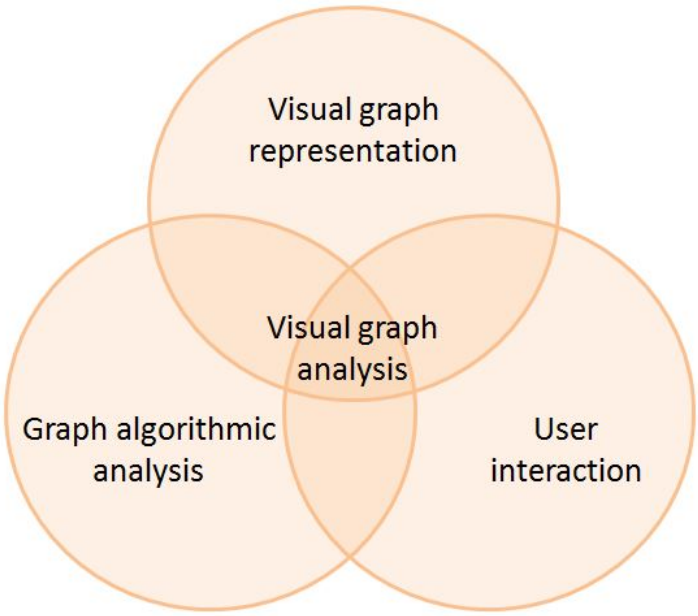

Figure 9: Main components of visual graph analysis (Von Landesberger et al. 2011)

To better understand graph visualizations a short abstract about graph theory follows:

A graph in general is a set of vertices V (i.e. nodes) and a set of edges E (i.e. links) that connect pairs of vertices (Diestel 2005, p. 2).
In addition, attributes can be attached to nodes and edges that provide additional information like their type, size etc. Graphs can either be *directed* or *undirected*. If to every edge e an initial vertex $init(e)$ and a terminal vertex $ter(e)$ are assigned, it is called a directed graph. There can be several edges between the same two vertices which are then called *multiple edges*. If there are no loops or multiple edges in a directed graph, it is called an *oriented graph*. (Diestel 2005, p. 28)

Several substructures can be identified in graphs. A *path* with length k is a sequence of distinct vertices x_i (with i=0,...,k) which are connected through edges $x_j x_l$ (with i=0,...,k-1 and l=1,...,k). The vertices x_0 and x_k are called *ends*. (Diestel 2005, p. 6)
If there is a closed path where x_0 equals x_k it is called a *cycle*. If it is an undirected graph with no cycles, it is called a *tree*. A tree is called *rooted* if a particular node is designated as a *root node*. Rooted trees are often considered as hierarchies where the length of the path from a vertex to the root determines its level in the hierarchy. (Diestel 2005, p. 13ff.)
A directed graph with weighted edges, e.g. associated flows of material or transportation costs, is called a *network*. In Visual Analytics the term network is also used for graphs whose vertices and edges are associated with attributes in general. It is possible to successively aggregate (or cluster) vertices in a graph by adding new *meta-nodes* or *super-nodes* and *meta-edges* between those nodes to form *aggregated graphs*. If graphs evolve over time, i.e. its attributes, the graph structure or both changes, they are called *dynamic graphs* in contrast to *static graphs*. (Von Landesberger et al. 2011, p. 1721)

Several algorithms for mapping and transforming data into graph structures exist. For more details on graph algorithms see the available literature about graph theory.
Due to their importance for graph visualization the graph pre-processing techniques graph filtering and graph aggregation are shortly described. Graph preprocessing in general is applied to simplify graph structures or highlight the most interesting parts of graphs, as visual inspection of large and complex graphs can be very difficult and time-consuming. *Graph filtering* means selecting only certain nodes and edges of a graph either stochastically or by applying deterministic algorithms. These algorithms can e.g. use the attributes assigned to nodes/edges or the topology of the graph to filter out parts of a graph that are less relevant for answering a certain question. The second technique called *graph aggregation* merges nodes and edges to reduce the size of graphs and reveal relationships among groups of nodes. (Von Landesberger et al. 2011, p. 1722)

4.2.2 Visualization of Graphs

Visual presentation through effective graph layouts, visual mappings and interaction techniques is essential to support the analysis of graphs. Very often graphs form complex structures which makes it impossible to analyze a complete graph with only one static view. Thus, interaction and graph processing techniques, as mentioned in the previous section, are required to navigate effectively through graphs. (Von Landesberger et al. 2011, p. 1719)
Many graph based visualizations and even more derivatives exist which cannot all be discussed in this project paper. Only a selection of well-known visualizations will be discussed. They have in common that basic graph properties, like number of nodes included, graph density and connectivity are used by graph based

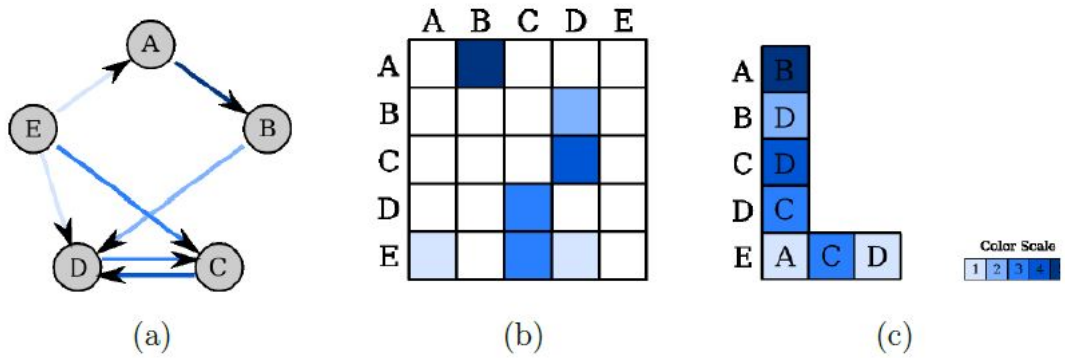

Figure 10: Three different representations for the same graph: (a) node-link diagram; (b) matrix representation; (c) list representation. (Burch 2009)

visualization algorithms to apply certain visualization methods. Those properties influence the choice and effectiveness of the applied methods. (Von Landesberger et al. 2011, p. 1721)

Talking about graphs in general without considering substructures as introduced in the previous section, Burch (2009, p. 17f.) names three basic types of graph visualizations (see fig. 10). One of the most prevalent representations is the node-link style. *Node-link-based representations* can be intuitively interpreted, as a set of nodes is connected by lines that represent some kind of relationship. However, the more nodes and the more links are displayed the harder it becomes to analyze the diagrams because of link crossings and seemingly vanishing level of detail. This problem is known as *visual clutter*.
A second type of graph representations is the *matrix representation*. Here the vertices are represented at both axes of a matrix. A marker at the intersection of both axes represents a link between elements. Edge weights can be represented with special color coding. With this form of representation link crossing can be avoided and links can be displayed very clearly, though the doubled presence of an element at the different axes makes tracking paths in the graph difficult.
A third type of graph representation is a *list-based representation*. In list-based representations for a vertex all vertices that are linked through an edge are added in a list, but there is no order which makes it difficult to read the visualization. Despite vertices with many outgoing edges can be easily identified, finding incoming edges or identifying the vertex with the most incoming edges becomes very difficult in large graphs without proper support through interactive features. Thus, tools that use list-based graph representations are rare at the market.

When trees as a subset of graphs are to be visualized, Burch (2009, p. 23) shows that next to the previously introduced visualizations additional types of visualizations can be used which make use of the planar characteristic of trees. *Planar* means that a tree can be visualized as a node-link diagram in a two dimensional plane without crossing edges.

Node-link diagrams for tree layouting can be top-down, left-to-right or radial oriented concerning the positioning of parent-child relationships (Herman et al. 2000, p. 4).

Two "non-traditional" so called space-filling graph visualizations used for tree visualization are tree-maps and sunburst diagrams (see fig. 11). In *tree-maps* rectangles/boxes form the vertices which are nested according to its sequence in the graph. Herman et al. (2000, p. 4) show that the size of the boxes is significant for interpreting the visualizations, as it displays certain attributes of the vertices or edges (e.g. file size in a hierarchical file system). Thus, Burch (2009, p.) emphasizes that tree-maps can be very effective to visualize the size of vertices but they are much more ineffective compared to e.g. node-link diagrams when analyzing the hierarchical structure of a graph.
Another form of space-filling visualization is the radial *sunburst diagram*. This type of diagram distinguishes from tree-maps as the parent nodes are not overlapped by child-nodes. The parent-child relationships are represented by placing the children next to the parent. Thus, attributes can be more easily displayed and analyzed, but the diagrams are not as dense as tree-maps. A non-radial, linear form of those diagrams is called *icicle plots*. (Von Landesberger et al. 2011, p. 1724)

4.2.3 Application of Graph Based Analytics

As shown in Herman et al. (2000, p. 1) graph visualization is used in many areas of applications ranging from displaying file hierarchies in computer systems, to organizational charts in companies, taxonomies in biology, object and flow structures in software development or semantic networks and knowledge-representation diagrams. To shorten this enumeration, graph visualization can be applied everywhere where the following question should be answered: *Is there an inherent relation among the data elements to be visualized?*

As the idea of graph visualization itself is much older than the notion of Visual Analytics which has been introduced by Thomas & Cook in 2005, the traditional area of applying graphs in science has also been one of the first areas of application for visual graph analysis (e.g. analyzing protein functions in biology). However, as mentioned by Lemieux (2011, p. 38), Visual Analytics and also graph analysis have moved to other areas like business intelligence, fraud detection and epidemiology.

In graph based analytical applications different granular analytical tasks can be performed by the analytical engine or through user interaction. Table 2 shows different tasks in graph analysis that form the basic analytical functionalities offered in graph based tools. For more details on the different tasks see Von Landesberger et al. (2011, p. 1735ff.).

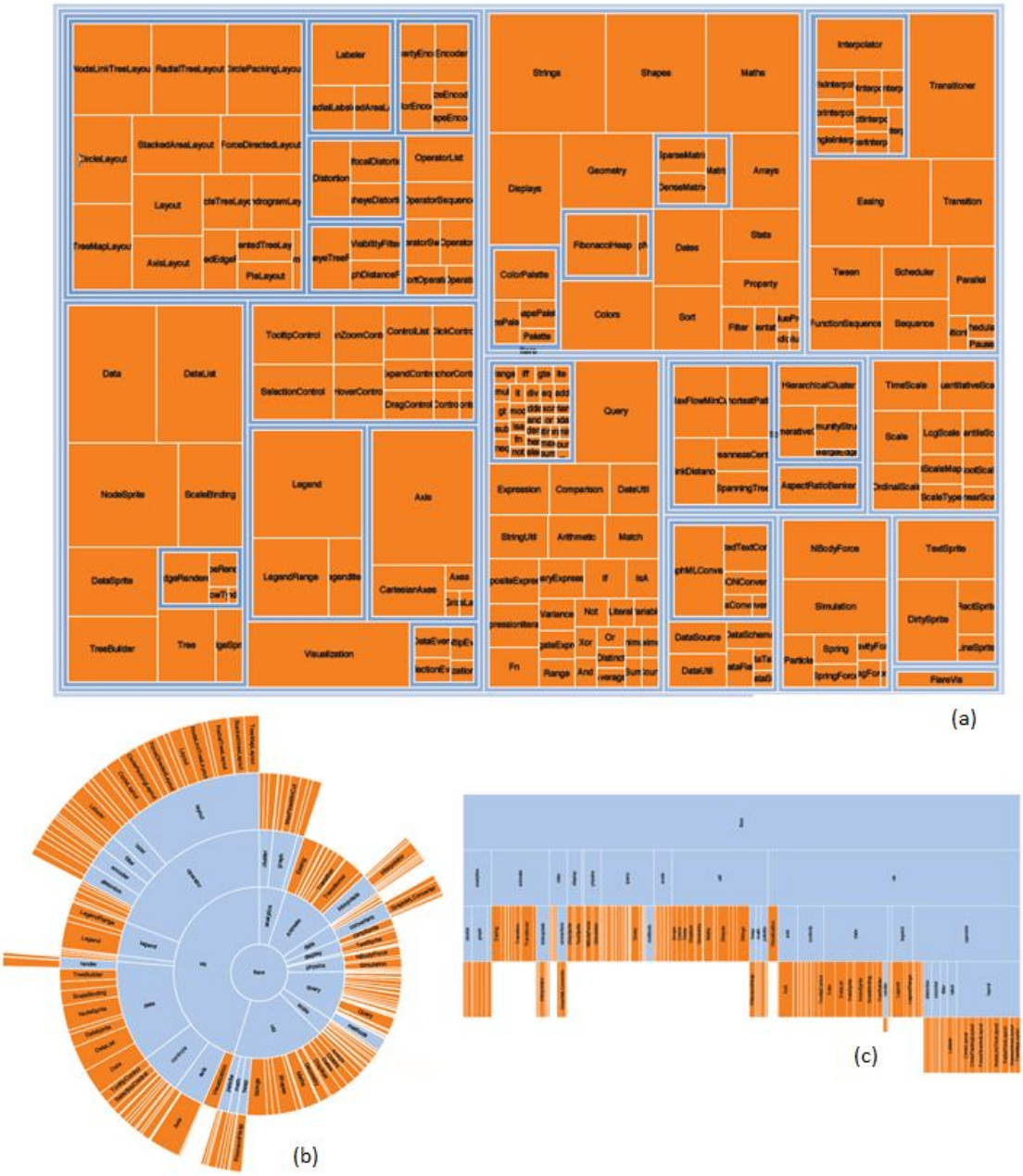

Figure 11: Different types of tree visualization: (a) tree-map; (b) sunburst diagram; (c) icicle plot (Heer et al. 2010)

Task	Focus of Analysis
Analysis of Graph Structure	Identification of important nodes Connections between nodes Graph substructures Graph structure on several aggregation levels Impact of graph changes on the structural properties
Graph Comparison	One-to-one node comparison of two graphs One-to-many nodes comparison of two graphs Structural differences between two graphs Structural similarity among multiple graphs

Table 2: Graph analysis tasks

The ongoing expansion of social networks and the richness of the data that can be derived (see section 1) constitute an important field of application for graph based Visual Analytics. Due to the emerging presence of companies in social networks like Facebook, and the availability of professional networks like XING or LinkedIn, companies encounter new possibilities of gathering valuable information for their business decisions, e.g. they are able to analyze connections between (potential) customers, markets, market trends, focus groups, potential employees etc. The analytical tasks mentioned in table 2 support decision makers in deriving meaningful information from social network data, e.g. the analysis of graph substructures helps to identify cliques or focus groups in social networks.
LinkedIn offers a graph analysis tool called *InMaps* to its users, which allows them to automatically cluster and visualize their connections in the network. Figure 12 shows exemplarily how a professional network of a LinkedIn employee looks like. Different color codings represent different affiliations and groups. In the example contacts from other companies are grouped. The label size symbolizes highly connected people in the network. The tool enables users to identify clusters of people, experts, new and maybe already forgotten connections. Thus, InMaps is a good example of how social network data can be visualized and explored with the help of graph analysis and visualization.

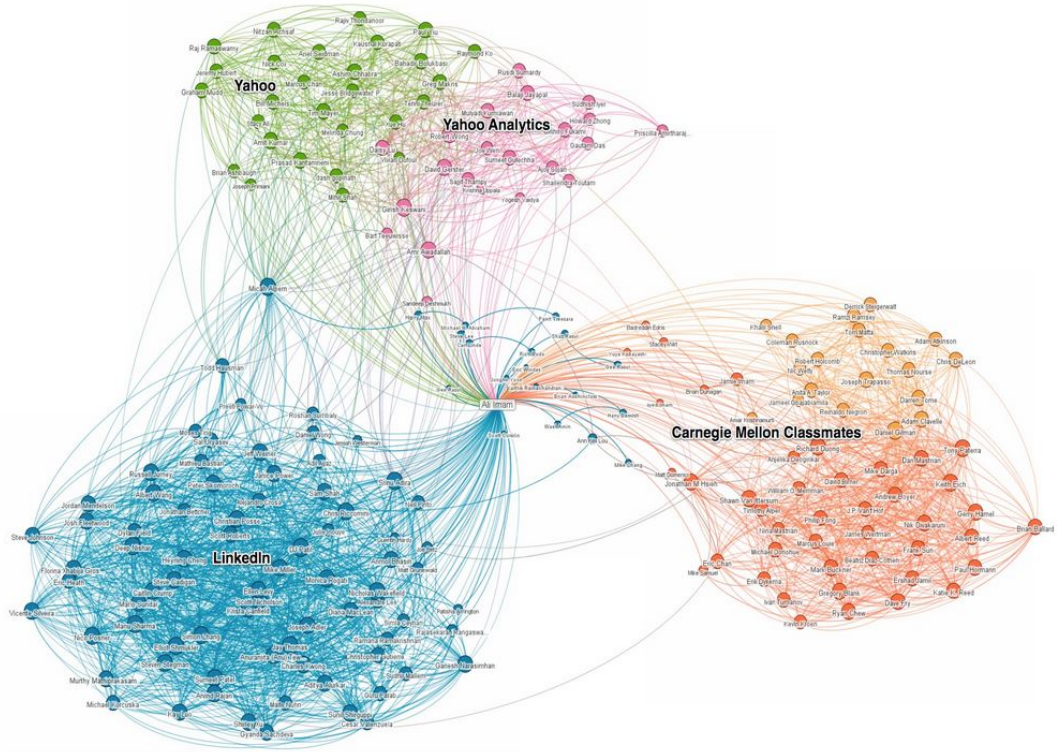

Figure 12: InMaps example for LinkedIn employee (Imam 2011)

4.3 FCA Based Visual Analytics

4.3.1 Basics of Formal Concept Analysis (FCA)

Though graph based analytics take a dominant role in Visual Analytics, a rather new approach of computer analysis in connection with BI systems called **Formal Concept Analysis (FCA)** will be introduced in the following sections. FCA itself has been introduced by Wille (1982). The idea to integrate FCA as an algorithmic engine in BI systems has been fostered by the EU funded research project named CUBIST (see section 4.3.2).
To understand the value of FCA as an analytical analysis method in BI some essential FCA fundamentals are described in the following paragraph.
As part of the methods name, the notion of a *concept* plays an important role in FCA. Philosophically, a concept can be defined as a unit of thoughts that consists of an *extension* and an *intension*. The extension comprises all objects that belong to the concept and the intension includes all attributes valid for all objects in the extension. The relation between the objects and the attributes is called *incidence relation* ("an object has an attribute"). (Wolff 1994, p. 2)

From a more formal perspective the elementary notions in Formal Concept Analysis are the terms *formal context* and *formal concept*.

ANIMALS	preying	flying	bird	mammal
LION	×			×
FINCH		×	×	
EAGLE	×	×	×	
HARE				×
OSTRICH			×	

Figure 13: A table of crosses representing a formal context (Wolff 1994)

According to Ganter & Wille (1996, p. 17) a *formal context* K can be defined as

$$K := (G, M, I)$$

where G is a set of objects, M a set of attributes and I the relation that relates objects in G to attributes in M.

A *formal concept* of a context (G, M, I) is a pair (A, B) with

$$A \subseteq G, B \subseteq M, A' = B \text{ and } B' = A.$$

Defined that $A' := \{m \in M \mid (g, m) \in I, \forall g \in A\}$ (the set of all common attributes in A) and $B' := \{g \in G \mid (g, m) \in I, \forall m \in B\}$ (the set of objects having all attributes of B).
A is called the *extent* and B is called the *intent* of a concept (A, B).

A natural hierarchical order called *subconcept-superconcept relation* exists between concepts in a context. In case (A_1, B_1) and (A_2, B_2) are concepts in a context, (A_1, B_1) is called a *subconcept* of (A_2, B_2) if $A_1 \subseteq A_2$ (or equivalent $B_2 \subseteq B_1$). Then, (A_2, B_2) is a *superconcept* of (A_1, B_1). This relationship is a *partial ordering relationship*, to be formalized as:

$$(A_1, B_1) \leq (A_2, B_2) \Leftrightarrow A_1 \subseteq A_2 \text{ and } B_2 \subseteq B_1$$

The thus ordered set of all concepts in (G, M, I) is called *concept lattice* of the context (G, M, I). The context can be completely reconstructed from the concept lattice.

Figure 13 shows an example of an animal context defined as a table of crosses. Focusing the FINCH and the EAGLE, a set X can be identified that includes a set Y of attributes (flying and bird). Thus, the pair (X, Y) forms a formal concept in the context of animals. The concept might be called "flying birds". In this example the concept "preying flying birds" (i.e. the EAGLE) is a subconcept of

the concept "flying birds" (i.e. the EAGLE and FINCH).
Next to this *object concept* view the *attribute concept* view can be taken. In the attribute concept (C, D) for an attribute m, C determines all objects of m and D is the set of all attributes valid for all objects in C. In the example, the attribute concept for "flying" comprises the objects FINCH and EAGLE which have the attributes "flying" and "bird" in common.

4.3.2 Visualization of FCA Results

FCA can be used to present data in a hierarchical form by analyzing a set of objects and associated attributes. As the mathematical representation introduced in the previous section is not feasible for appliance in (business) scenarios the most commonly used visualizations will be introduced.

Very often a concept lattice is visualized with a line-diagram (a so called *Hasse diagram*). In a Hasse diagram the formal concepts of a concept lattice are visualized by circles. Objects that belong to a subconcept of a superconcept are placed below the objects that belong to the superconcept. (Burmeister 2003, p. 15)
For example in figure 14, which is a line-diagram of the previously used animal concept lattice, the concept generated by the object EAGLE (with the attributes preying, flying, bird) is placed below the concept generated by FINCH (with the attributes flying, bird).
As depicted by Burmeister (2003, p. 18), by convention the object name is written below the circle representing the object. An attribute name is written above the circle which represents the attribute concept. The object circles are connected through edges with the attribute circles which represent the incidence relation. From the object concept perspective, the Hasse diagram can be read from the bottom to the top by following sequences of line segments. It is possible to move upwards as long as an object has the attribute at the opposite end of the sequence., e.g. when looking for the concept "flying birds" one can go from the EAGLE up to the FINCH but not up to the OSTRICH as it is placed above the attribute "flying", which implies that OSTRICH does not carry this attribute. Consequently, taking the attribute concept perspective one has to follow the sequences from the top to the bottom, starting at the respective attribute circle to identify all objects carrying the attribute.
It is possible that one circle represents an object and an attribute simultaneously if there is a unique relationship between the object and the attribute, taking the restrictions regarding sequences and placement of circles in the diagram into consideration. For example, the OSTRICH is the only object in this context that only carries the attribute "bird" and no additional attributes. Thus, the literals "bird" and "OSTRICH" are placed at a common circle in the diagram.

Without basic knowledge about how FCA results are mapped to the Hasse diagram it might be difficult to understand the diagram correctly. Furthermore, when contexts with a large number of concepts are visualized the visualization might be

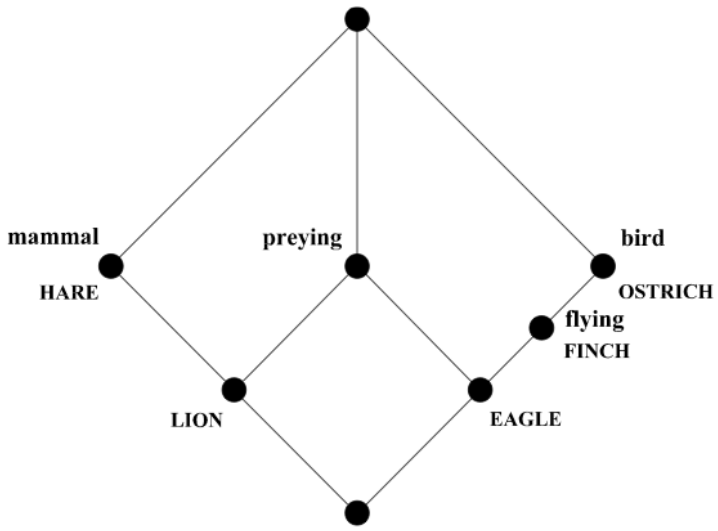

Figure 14: Line diagram for the animal concept lattice (Wolff 1994)

confusing. Melo et al. (2011, p. 262) propose an approach for extracting tree structures from concept lattices to visualize the data in tree based visualizations which are easily understandable and more commonly used. Their basic idea is to choose one single parent for each concept in a lattice. The main objective in this procedure is to preserve the most essential features of the original concept lattice structure as the transformation leads to information loss. Thus, Melo et al. propose different strategies for parent selection. After extracting the tree structures, tree-based visualizations as introduced in section 4.2.2 can be applied.

4.3.3 Application of Formal Concept Analysis

FCA in general is a formal method for data analysis and knowledge representation. Its formalism and application independence allows it to apply it in different functional areas. In comparison to graph based analytics, FCA is not mainly used to analyze inherent, maybe explicit relationships between entities but rather stresses the derivation of the meaning of data by creating clusters of objects through analyzing commonly shared or differing sets of attributes. In a second step, relationships between objects can be inferred.
A frequently mentioned area of application is analyzing the semantic structures of textual data. Conceptual models can be extracted through human natural language processing by applying FCA methods (Laukaitis & Vasilecas 2007, p. 5). By offering such a conceptual centric view on a domain of interest, a common understanding about concepts with less ambiguity can be created. Exemplarily, the research of Hashemi et al. (2004) can be stated who demonstrate how FCA has been applied on interview data and accident logs to conceptualize marine accidents for marine policy and regulation purposes.

Other possible areas of applications, extracted from Wolff (1994, p. 10) and Kang et al. (2009, p. 353), are listed below without raising claim to completeness:

- Psychoanalysis
- Industrial Engineering
- Bioinformatics
- Social Science
- Ontology Engineering
- Software (Re-)Engineering
- Knowledge Systems

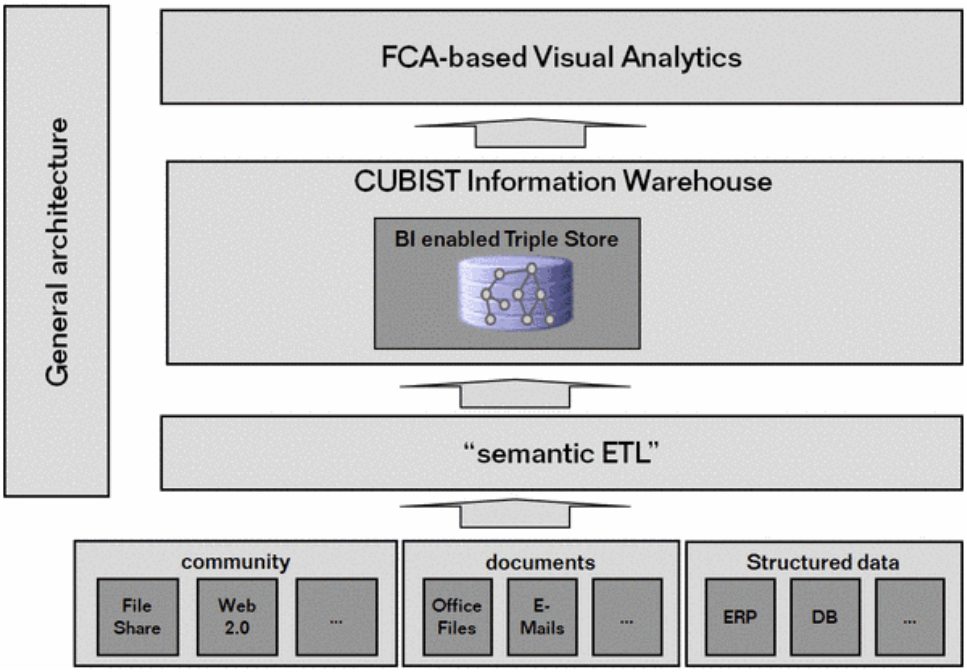

Figure 15: CUBIST architecture (CUB 2013)

A first approach of integrating FCA into BI systems is the prototypic application called CUBIST. The main idea of CUBIST is to integrate semantic technologies like the FCA method into business intelligence environments by focusing on the meaning of data and thus enabling the user to discover new (semantic) facts in the data which are not explicitly modeled in the database schemes. By fostering the analytical reasoning on data and having the meaning of data in place, the user should be better guided in the analysis process.

Figure 15 shows the basic architecture of CUBIST. The CUBIST architecture is very similar to the BI architecture introduced in section 2.1, including different datasources, an ETL layer and an information warehouse layer, which differ from traditional BI through the integration of semantic technologies, e.g. a triple-store is used instead of a relational database. The FCA engine and the visualizations are placed on top. In the course of the CUBIST project three use cases have been implemented to show the capabilities of the tool exemplarily. The use cases include the analysis of biomedical atlases, data of space control centres and recruitment data for the UK job market. For more details about the use cases see the CUBIST project website[3]. The analytical tool offered in CUBIST is a first approach of combining the different analytical methods (FCA, graph and traditional BI) in a common tool as visualized in figure 16. Next to Hasse diagrams, tree-based visualizations or traditional diagrams like bar charts can be used in CUBIST front-end providing different perspectives on the same data.

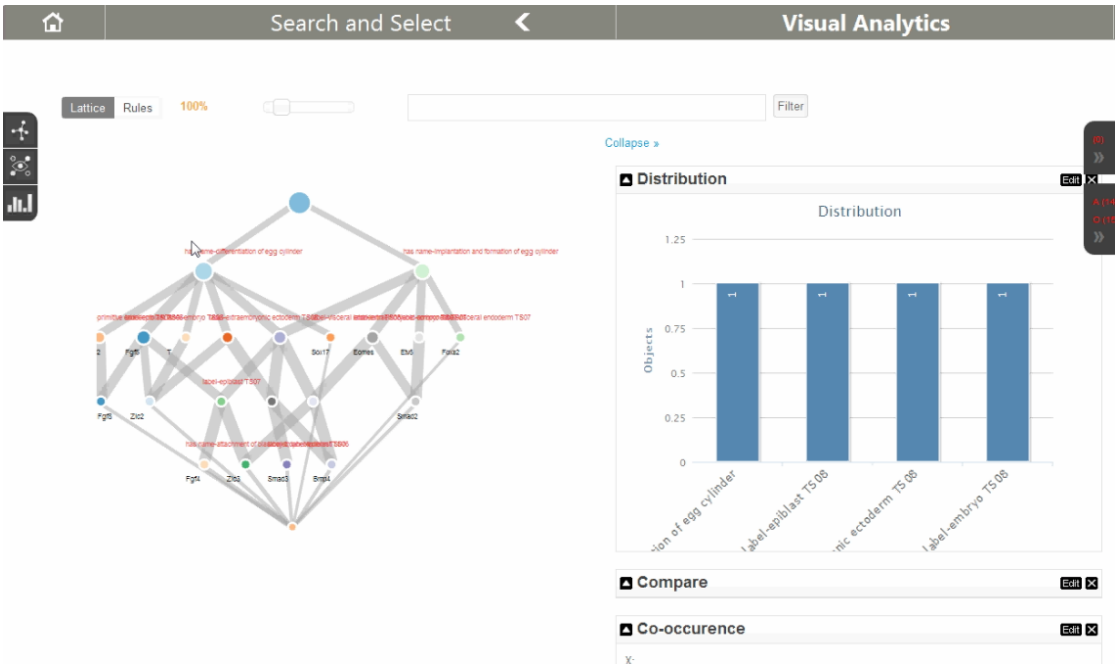

Figure 16: CUBIST prototype frontend

[3]http://www.cubist-project.eu/

5 Summary and Outlook

The previous sections have shown that each of the introduced approaches to analytical information systems does have its own impact on what kind of information can be extracted from raw data and how this information can be leveraged in decision making processes.
Traditional number crunching based tools allow to analyze mainly numeric data from multiple perspectives. Graph-based tools can be employed to visualize and analyze relationships among entities that can be extracted from the data. FCA-based tools can be used to describe the nature (i.e. the attributes) of entities in the data and thus assign meaning to them.
Next to the analytical basics of the different tools the most commonly used visualizations and applications have been introduced. However, the number of possible visualizations is almost infinite. Which visualization is best suitable for a certain analysis situation depends on various factors, e.g. the type of data to be analyzed, the questions to answer or the analyst's level of knowledge in the domain of interest. Thomas & Cook (2005, p. 30) emphasize that a single visual metaphor cannot meet all analytical needs but a combination can provide multiple complementary views on the information.
As introduced with the notion of Visual Analytics, next to visualization interaction plays an important role in the analytical reasoning process. Analytical interaction capabilities have not been discussed within this project paper. Nevertheless, for the analysis process they are as essential as visualization itself since interaction allows to explore the underlying data, navigate in the information space and reveal unexpected facts. Thus, interaction with analytical tools is another interesting field of research.
Considering future activities in analytical information systems research, a system that combines the different analytical approaches might be of interest. Such an integrated system should exploit the strengths of the different analytical approaches and minimize the weaknesses that exist when using each approach separately. This will require to systematically classify the questions answered with analytical information systems, derive the analytical requirements from such a classification, analyze the capabilities (strengths and weaknesses) of the analytical approaches, visualizations and interaction functionalities and match them to the requirements. Going further, next to analyzing the requirements and capabilities of such a multipurpose system, research on automating the selection of the analytical methods, visualizations and interaction functionalities depending on the specific situation in the analysis process will enable users to effectively utilize analytical information systems without having to care about the most appropriate analytical approach by themselves. This requirement is immanent in times of increasing amounts of data, various available types of data and analytical approaches.

References

CUB 2013
CUBIST. About the Project: Structure. `http://www.cubist-project.eu/index.php?id=439`. Version: 2013, Accessed: June 9, 2013

Arnold 2013
ARNOLD, Stephen E.: Analytics for Pain-Free Number-Crunching. In: *Information Today* 30 (2013), January, Nr. 1, pages 18–19

Bange 2006
BANGE, Carsten: Werkzeuge für analytische Informationssysteme. In: CHAMONI, Peter (Ed.); GLUCHOWSKI, Peter (Ed.): *Analytische Informationssysteme. Business Intelligence-Technologien und -Anwendungen.* 3. Springer Verlag, 2006, pages 90–110

Breitman et al. 2006
BREITMAN, Karin; CASANOVA, Marco A.; TRUSZKOWSKI, Walt: *Semantic Web: Concepts, Technologies and Applications (NASA Monographs in Systems and Software Engineering).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006

Burch 2009
BURCH, Michael: *Visualizing Static and Dynamic Relations in Information Hierarchies.* Universitätsring 15, 54296 Trier, Universität Trier, Diss., 2009

Burmeister 2003
BURMEISTER, Peter: *Formal Concept Analysis with ConImp: Introduction to the Basic Features.* `http://www.mathematik.tu-darmstadt.de/~burmeister/ConImpIntro.pdf`. Version: April 2003, Accessed: June 9, 2013

Card et al. 1999
CARD, Stuart K. (Ed.); MACKINLAY, Jock D. (Ed.); SHNEIDERMAN, Ben (Ed.): *Readings in information visualization: using vision to think.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999

Chamoni & Gluchowski 2006
CHAMONI, Peter; GLUCHOWSKI, Peter: Analytische Informationssysteme - Einordnung und Überblick. In: CHAMONI, Peter (Ed.); GLUCHOWSKI, Peter (Ed.): *Analytische Informationssysteme. Business Intelligence-Technologien und -Anwendungen.* 3. Springer Verlag, 2006, pages 3–22

Chamoni et al. 2005
CHAMONI, Peter; GLUCHOWSKI, Peter; HAHNE, Michael: *Business Information Warehouse.* Springer-Verlag, 2005

Chaudhuri et al. 2011
CHAUDHURI, Surajit; DAYAL, Umeshwar; NARASAYYA, Vivek: An overview

of business intelligence technology. In: *Commun. ACM* 54 (2011), August, Nr. 8, pages 88–98

Chi & Riedl 1998
CHI, Ed Huai-Hsin; RIEDL, John: *An Operator Interaction Framework for Visualization Systems.* In: *Proceedings of the 1998 IEEE Symposium on Information Visualization.* IEEE Computer Society, Washington, DC, USA, 1998 (INFOVIS '98), pages 63–70

Diestel 2005
DIESTEL, Reinhard: *Graph Theory (Graduate Texts in Mathematics).* 3. Springer, 2005

Fekete et al. 2008
FEKETE, Jean-Daniel; WIJK, Jarke J.; STASKO, John T.; NORTH, Chris: The Value of Information Visualization. In: KERREN, Andreas (Ed.); STASKO, John T. (Ed.); FEKETE, Jean-Daniel (Ed.); NORTH, Chris (Ed.): *Information Visualization.* Springer-Verlag, Berlin, Heidelberg, 2008, pages 1–18

Florescu 2005
FLORESCU, Daniela: Managing Semi-Structured Data. In: *Queue* 3 (2005), Nr. 8, pages 18–24

Gabriel et al. 2009
GABRIEL, Roland; GLUCHOWSKI, Peter; PASTWA, Alexander: *Data Warehouse & Data Mining.* W3L-Verlag, 2009

Ganter & Wille 1996
GANTER, Bernhard; WILLE, Rudolf: *Formale Begriffsanalyse - Mathematische Grundlagen.* Springer, 1996

Ghazanfari et al. 2011
GHAZANFARI, Mehdi; ROUHANI, Saeed; JAFARI, Mostafa; AKHAVAN, Peyman: Association Rule Approach for Evaluation of Business Intelligence for Enterprise Systems. In: *The IUP Journal of Computer Sciences* 5 (2011), April, Nr. 2, pages 21–38

Gluchowski & Chamoni 2006
GLUCHOWSKI, Peter; CHAMONI, Peter: Entwicklungslinien und Architekturkonzepte des On-Line Analytical Processing. In: CHAMONI, Peter (Ed.); GLUCHOWSKI, Peter (Ed.): *Analytische Informationssysteme. Business Intelligence-Technologien und -Anwendungen.* 3. Springer Verlag, 2006, pages 3–22

Haber & McNabb 1990
HABER, R. B.; MCNABB, D. A.: Visualization Idioms: A Conceptual Model for Scientific Visualization Systems. In: *Visualization in Scientific Computing.* IEEE Computer Society Press, 1990, pages 74–93

Hashemi et al. 2004
HASHEMI, Ray R.; LE BLANC, Louis A.; KOBAYASHI, Tetsuya: Formal concept analysis for investigation of normal accidents. In: *International Journal of General Systems* 33 (2004), Nr. 5, pages 469–484

Heer et al. 2010
HEER, Jeffrey; BOSTOCK, Michael; OGIEVETSKY, Vadim: A Tour through the Visualization Zoo. In: *Queue* 8 (2010), Nr. 5, pages 20–30

Herman et al. 2000
HERMAN, Ivan; MELANÇON, Guy; MARSHALL, M. S.: Graph Visualization and Navigation in Information Visualization: A Survey. In: *IEEE Transactions on Visualization and Computer Graphics* 6 (2000), Januar, Nr. 1, pages 24–43

Hosack et al. 2012
HOSACK, Bryan; HALL, Dianne; PARADICE, David B.; COURTNEY, James F.: A Look Toward the Future: Decision Support Systems Research is Alive and Well. In: *J. AIS* 13 (2012), Nr. 5, pages 315–340

Imam 2011
IMAM, Ali: *Visualize your LinkedIn network with InMaps.* `http://blog.linkedin.com/2011/01/24/linkedin-inmaps/`, 2011. – Accessed June 1, 2013

John et al. 2011
JOHN, Peter; KESSLER, Torsten; KNÖTZELE, Thilo; THALER-MIESLINGER, Karin; HEILIG, Loren: *SAP NetWeaver BW und SAP BusinessObjects. Das umfassende Handbuch.* Galileo Press, 2011

Kang et al. 2009
KANG, Yu-Kyung; HWANG, Suk-Hyung; YANG, Kyoung-Mo: FCA-based conceptual knowledge discovery in Folksonomy. In: *International Journal of Electrical and Electronics Engineering* 3 (2009), Nr. 6, pages 353–357

Laukaitis & Vasilecas 2007
LAUKAITIS, Algirdas; VASILECAS, Olegas: *Formal concept analysis and information systems modeling.* In: *Proceedings of the 2007 international conference on Computer systems and technologies.* ACM, New York, NY, USA, 2007 (CompSysTech '07), pages 17:1–17:6

Lemieux 2011
LEMIEUX, Victoria L.: Visual Analytics: A New Way to Manage Data Deluge in E-Discovery. In: *Information Management Journal* 45 (2011), March, Nr. 2, pages 38–40

Melo et al. 2011
MELO, Cassio; LE-GRAND, Benedicte; AUFAURE, Marie-Aude; BEZERIANOS,

Anastasia: *Extracting and Visualising Tree-like Structures from Concept Lattices.* In: *Proceedings of the 2011 15th International Conference on Information Visualisation.* IEEE Computer Society, Washington, DC, USA, 2011 (IV '11), 261–266

Sperberg-McQueen 2005
SPERBERG-MCQUEEN, C. M.: XML <and Semi-Structured Data>. In: *Queue* 3 (2005), Nr. 8, pages 34–41

Thomas & Cook 2005
THOMAS, James J.; COOK, Kristin A.: *Illuminating the Path: The Research and Development Agenda for Visual Analytics.* National Visualization and Analytics Ctr, 2005

Von Landesberger et al. 2011
VON LANDESBERGER, T.; KUIJPER, A.; SCHRECK, T.; KOHLHAMMER, J.; WIJK, J.J. van; FEKETE, J.-D.; FELLNER, D.W.: Visual Analysis of Large Graphs: State-of-the-Art and Future Research Challenges. In: *Computer Graphics Forum* 30 (2011), Nr. 6, pages 1719–1749

Wille 1982
WILLE, Rudolf: *Restructuring lattice theory: an approach based on hierarchies of concepts.* In: RIVAL, Ivan (Ed.): *Ordered sets.* Reidel, Dordrecht–Boston, 1982, pages 445–470

Wolff 1994
WOLFF, Karl E.: *A Fist Course in Formal Concept Analysis How To Understand Line Diagrams.* In: FAULBAUM, F. (Ed.): *SoftStat '93: Advances in Statistical Software 4.* Fischer, Stuttgart, 1994, pages 429–438