

Die approbierte Originalversion dieser Dissertation ist an der Hauptbibliothek der Technischen Universität Wien aufgestellt (<http://www.ub.tuwien.ac.at>).

The approved original version of this thesis is available at the main library of the Vienna University of Technology (<http://www.ub.tuwien.ac.at/englweb/>).



Ph.D. Thesis

Gravi++
An Interactive Information Visualization
for High Dimensional, Temporal Data

Conducted for the purpose of receiving the academic title
'Doktor der technischen Wissenschaften'

Supervisor

Silvia Miksch

Institute of Software Technology & Interactive Systems (E188)

Submitted at Vienna University of Technology
Faculty of Informatics

by

KLAUS ALEXANDER HINUM

9625167

Dieselgasse 5a/13

A-1100 Wien

Vienna, December 12, 2006

GRAVI⁺

Abstract

Tracking and comparing psychotherapeutic data derived from questionnaires involves a number of highly structured, time-oriented parameters. Because of the small amount of data items, traditional statistical analysis are not suitable. Therefore, we decided to analyze this special kind of data with Information Visualization techniques. As a first step we searched for existing visualization methods that could be applied to our problem characteristics. We found out that there are no techniques that are able to perfectly cope with our kind of data and support the medical personnel in analyzing the psychotherapeutic data and, thus, we decided to develop a new visualization technique for this reason.

In this thesis we propose an interactive Information Visualization called Gravi++ to visualize highly structured, temporal, categorical data. It integrates a spring-based core visualization to display the multidimensional data set. To cope with the temporal dimension, we integrated animation and an additional view called Traces that maps the animation on a single image. Furthermore, the Star Plot view, Attraction Rings view, and the tooltips can give information on the exact values of each dimension. In addition to this, we added two overview visualizations called ListVis and TableVis to handle larger amounts of data and enable the user to select a subset. The different views are coordinated using linking and brushing and provide many different interaction possibilities.

In two different use cases we present some typical steps when analyzing data with Gravi++. One use case demonstrates interactions on the medical data set we used in our evaluation and the other one demonstrates the use of Gravi++ when visualizing a larger data set of Eurobarometer data (questionnaires to assess the attitude towards the European Union).

To get more information on the usefulness of Gravi++ we conducted an extensive evaluation of our visualization. The first step was a usability evaluation to preclude that usability problems are mixed up with problems with the visualization method as such. After that we conducted an insight study, where we wanted to find the strength and limitations of supervised Machine Learning (ML) techniques, Exploratory Data Analysis (EDA), and Gravi++ for our problem domain. It indicates that Gravi++ is a very good method for analyzing this kind of data and that using ML, EDA and visualization methods in conjunction will very likely contribute to a deeper comprehension of the data to explore.

Zusammenfassung

Die Analyse von psychotherapeutischen, auf Fragebögen basierenden Daten beinhaltet hochdimensionale, kategorische, zeit-orientierte Parameter. Die Daten stammen von einer klinischen Studie mit Patientinnen, die an Anorexie (Essensstörungen) leiden. Wegen der geringen Anzahl von Datenelementen, sind traditionelle statistische Methoden nicht anwendbar. Deshalb suchten wir nach existierenden passenden Methoden und Techniken die auf unsere Problemstellung anwendbar sind. Da jedoch keine passenden Methoden gefunden werden konnten, entschieden wir uns zur Entwicklung einer neuen Informationsvisualisierung für unsere Aufgabenstellung.

In dieser Dissertation zeigen wir unseren Ansatz zur Visualisierung hochdimensionaler, zeit-orientierter, kategorischer Daten. Wir integrierten eine Feder-basierte Kernvisualisierung zur gleichzeitigen Darstellung mehrerer Dimensionen. Die Dimension Zeit bilden wir einerseits mit Animation und andererseits mit einer zusätzlichen Visualisierung namens Traces ab. Ausserdem integrierten wir noch zwei Methoden zur Darstellung der genauen Datenwerte (Attraction Rings und Star Plot). Um dem / der Benutzer/in eine Übersicht über die Daten zu geben, ergänzten wir unseren Prototypen auch noch mit zwei Übersichtsvisualisierungen. Alle diese unterschiedlichen Methoden sind durch "Linking und Brushing" miteinander verbunden und bieten zahlreiche Interaktionsmöglichkeiten.

In zwei unterschiedlichen Anwendungsbeispielen zeigen wir typische Anwendungsschritte mit Gravi++. Zuerst demonstrieren wir verschiedene Interaktionsmöglichkeiten mit den medizinischen Daten aus unserer Studie. Das zweite Anwendungsbeispiel demonstriert die Verwendbarkeit von Gravi++ mit grösseren Datenmengen, die aus Umfragen zur europäischen Union stammen.

In einer ausführlichen Studie wollten wir weitere Informationen über die Nützlichkeit unserer Visualisierung erhalten. Der erste Teil war eine ausführliche Usability Untersuchung, die zur Verbesserung unseres Prototypen beigetragen hat. Danach führten wir die Hauptuntersuchung durch, deren Zweck es war die Vor- und Nachteile dreier verschiedener Methoden – Machine Learning (ML), Exploratory Data Analysis (EDA), und Gravi++ bezüglich unserer Problemstellung herauszufinden. Die Ergebnisse zeigen, daß Gravi++ eine gute Methode ist diese Art von hoch-strukturierten, zeit-orientierten Daten zu analysieren. Ausserdem dürfte eine Kombination von ML, EDA und Visualisierung zu einem tieferen Verständnis über die Daten beitragen.

Contents

1	Introduction	9
1.1	Motivation	9
1.2	Research Question	11
1.2.1	Main Question	11
1.2.2	Sub Questions	11
1.3	Goals	11
1.4	Our Approach	11
1.5	Dissemination	12
1.6	Overview	12
1.7	Conventions	12
2	Related Work	13
2.1	Visualization Principles	14
2.1.1	Cognitive Psychology	14
2.1.2	Visualization Frameworks	15
2.1.3	Interactions	15
2.1.3.1	Details on Demand	16
2.1.3.2	Linking and Brushing	16
2.1.3.3	Focus+Context	17
2.1.4	What is Time-Oriented Information	17
2.1.5	Animation	17
2.1.5.1	Perceptive basics	18
2.1.5.2	History	18
2.1.5.3	Temporal versus Non-Temporal Animation	18
2.1.5.4	Computer Animation for Exploration Tasks	18
2.1.5.5	Computer Animation for Verification	19
2.1.5.6	Computer Animation for Demonstration	19
2.1.5.7	Optimal Speed	19
2.1.5.8	Conclusion	19
2.1.6	Color	20
2.1.7	Metaphor	20
2.1.8	Small Multiples	21
2.1.9	2D Versus 3D Visualizations	21
2.1.10	Spring-Based Positioning	21
2.2	Visualizations	22
2.2.1	Visualization methods in Statistics and Mathematics	22
2.2.1.1	Exploratory Data Analysis (EDA)	22
2.2.1.2	Scatter Plot / Scatter Plot Matrix	23
2.2.1.3	Dimension Reduction Techniques	23
2.2.2	Geometric Techniques	25
2.2.2.1	Sequence Graph	25
2.2.2.2	Parallel Coordinates Plot (PCP)	25
2.2.2.3	ThemeRiver	26

2.2.2.4	Self Organizing Maps (SOMs)	27
2.2.2.5	Hierarchical Techniques	27
2.2.2.6	RADVIZ	28
2.2.2.7	FreeViz	28
2.2.2.8	Vibe System	28
2.2.2.9	Dust & Magnet	29
2.2.2.10	Star Graph	29
2.2.2.11	Interactive Stardimates	30
2.2.2.12	Star Coordinates	30
2.2.2.13	TimeWheel / Zeitrads	30
2.2.2.14	MultiComb	31
2.2.2.15	TimeHistogram	31
2.2.2.16	TimeHistogram 3D	31
2.2.2.17	Time-Tunnel	32
2.2.2.18	Worm Plots	32
2.2.2.19	Concentric Circles Technique (CCT)	32
2.2.2.20	Star Representation Technique (SRT)	33
2.2.2.21	Flocking Boids	33
2.2.2.22	Interactive Parallel Bar Charts (IPBC)	33
2.2.2.23	Table Lens	34
2.2.2.24	LifeLines	34
2.2.3	GLYPHS	35
2.2.3.1	Chernoff Faces	35
2.2.3.2	Stick Figures	35
2.2.3.3	InfoBUG	36
2.2.3.4	Timewheel Glyph	36
2.2.3.5	3D Wheel	36
2.2.3.6	VIE-VISU	37
2.2.4	Pixel-based / Pixel-oriented Visualizations	38
2.2.4.1	Recursive Pattern	38
2.2.4.2	Circle Segments	38
2.2.4.3	Multi-scale Temporal Behavior	39
2.2.4.4	Shape Coding	39
2.2.4.5	Color Icon	39
2.2.5	Coordinated Multiple Views (CMVs)	40
2.2.5.1	Time-Series Explorer	40
2.2.5.2	SpringView	40
2.2.5.3	Visual Data Mining Display (VDMD)	41
2.2.5.4	VisDB	41
2.2.5.5	LinkVis	41
2.3	Discussion	42
3	Medical Problem Domain	43
3.1	Definition of Anorexia Nervosa (AN)	43
3.2	Cognitive Behavioral Therapy	44
3.3	Cognitive-Behavioral Therapy for Eating Disorders	44
3.4	Aims of the Study	45
3.5	Project Description	45
3.5.1	Methods	45
3.5.2	Evaluation	45
3.5.2.1	Individual Treatment Measures	45
3.5.2.2	Family Measures	46
3.5.2.3	Therapists Measures	46

4	Gravi++	47
4.1	Data	47
4.1.1	Data Type	47
4.1.1.1	Categorical Data	47
4.1.1.2	Time-Oriented Data	48
4.1.1.3	High Dimensionality	48
4.1.2	Data File Format	48
4.1.2.1	Parameter Master Data	49
4.1.2.2	Conflation Data Set	50
4.1.2.3	Other Suitable Data	52
4.2	The Method	52
4.2.1	Spring-Based Core Visualization	53
4.2.2	Algorithm	56
4.2.2.1	Linear Algorithm	56
4.2.2.2	Iterative Algorithm	56
4.2.3	Interactions	57
4.2.3.1	Interactions on Question's Icons	57
4.2.3.2	Interactions on Person's Icons	59
4.2.3.3	General Interactions	61
4.2.4	Coordinated Visualizations	64
4.2.4.1	Attraction Rings	64
4.2.4.2	Star Plot (Glyph)	65
4.2.4.3	Traces	66
4.2.5	Overview	68
4.2.5.1	ListVis	68
4.2.5.2	TableVis	69
4.2.6	Missing Data	71
4.3	View Coupling	71
4.4	User Requirements	72
4.5	Benefits & Limitations	73
4.5.1	Benefits & Features	73
4.5.2	Shortcomings	74
4.6	Future Work	74
5	Implementation	77
5.1	System Architecture	78
5.2	Metaphor	78
5.3	Data Format	80
5.4	Property File	81
5.5	Menu XML File	88
5.6	Data Import	88
5.6.1	Import using LinkVis	89
5.6.2	Import using GenEdit	90
5.6.3	General Data Input	91
6	Use Cases	93
6.1	Use Case 1 - Anorexia Nervosa - How to Find a Predictor	93
6.2	Use Case 2 - Eurobarometer-Data	95
6.3	Conclusion	98

7 Evaluation	101
7.1 Usability Study	102
7.1.1 Methods	102
7.1.1.1 Usability Inspection / Guidelines Review	102
7.1.1.2 Heuristic Evaluation	103
7.1.1.3 Focus Groups	103
7.1.2 Lab Settings	103
7.1.3 Results	103
7.1.4 Conclusion	104
7.2 Study	104
7.2.1 Report System	106
7.2.2 Classification System	106
7.2.3 Results and Discussion	106
7.2.4 Conclusion	107
7.3 Transferability	108
7.4 Summary	108
Acknowledgments	109
Appendix	111
Bibliography	115
Curriculum Vitae	129

Chapter 1

Introduction

Information Visualization (InfoVis) can be defined as

“the use of computer-supported, interactive, visual representations of data to amplify cognition. ” [Card et al., 1998]

In other words,

“Information visualizations attempt to efficiently map data variables onto visual dimensions in order to create graphic representations.” [Gee et al., 2005]

This means that, InfoVis produces images from abstract data with the goal of making efficient use of our visual system. There are three main purposes for Information Visualizations [Schumann and Müller, 2000]:

1. Exploration: the user searches for visual relations and structures to get new insights into the data under investigation. A possible outcome can be a hypothesis which has to be analyzed.
2. Analysis: the user has a specific hypothesis about the data and tries to verify or falsify it during a visual investigation.
3. Presentation: data characteristics are communicated to other users by presenting them with an InfoVis technique visually.

Our focus with Information Visualizations lies in the exploration and analysis tasks because we want to interactively explore the data set to make use of the expert users background knowledge and enable them to gain new insights into the data.

1.1 Motivation

This dissertation is embedded in the in2vis (standing for Interactive Information Visualization) project. One main goal of this project was to develop adequate interactive visualization techniques for psychotherapeutic data, in particular, data derived from a study of anorexic girls. This medical data is highly structured with temporal dimensions.

We derive the importance of visualizing medical data from the book “To Err Is Human” [Kohn et al., 1999]. They describe that errors in healthcare are a leading cause of death and injury. In their studies they state that at least 44,000 and perhaps as many as 98,000 Americans die in hospitals each year as a result of medical errors. The visualization of medical data can help the medical staff to get further insights into the patients state during a therapy and, therefore, help to avoid medical errors. Another important characteristic of medical data is the high amount of data that has to be analyzed by the physicians.

“The capability of interactively mining patient clinical information is an increasingly urgent need in the clinical domain, due to the continuous growth in the number of parameters that can be automatically acquired and in the size of the databases where they accumulate.” [Chittaro, 2001]

Therefore, we can conclude the importance of assisting medical tools in the clinical domain for analyzing the huge amount of data and help the experts to extract the relevant information.

Another aspect of our problem space is the resulting highly structured data. Analyzing multi-dimensional data is a difficult task because of the resulting high amount of data that has to be differentiated. Nowadays, most scientific, engineering, and business data is multi-dimensional [Shahar, 2000]. InfoVis is one way to ease the understanding and interpretation of this kind of data. Still, visualizing highly structured data is difficult with well known visualization techniques.

“Despite many visualization techniques available, there are not too many of those that can visualize several features in the same graph, and, for instance, include scatterplot (two or three features, the later if plotted in 3D), parallel coordinates and RadViz (both for presentation of data using many features).” [Keim, 2002]

As Keim mentioned, a traditional Plot (or Scatterplot) can visualize only two dimensions at once (up to four dimensions with the use of color and shape). The limiting factors for visualizing multi-dimensional data are the characteristics of traditional 2-dimensional computer screens with limited screen space. Using color, shape, and texture we can extend the visualization possibilities, but still it remains a challenging process.

Another important aspect of our problem domain is the time dimension in the data. It is almost impossible to represent and analyze clinical data in the absence of a temporal dimension [Shahar, 2000]. As time is an inherent data dimension, using special methods for treating time is necessary in order to accomplish the important tasks of revealing trends, as well as identifying patterns and relationships in the data [Aigner, 2006]. This is also honored by Ben Shneiderman’s “Task by Data Type Taxonomy” [Shneiderman, 1996] who identifies temporal data as one of seven basic data types most relevant for InfoVis.

The main goal of our Information Visualization is to help the users in getting new insights. This is defined as:

“Insight: The capacity to discern the true nature of a situation; The act or outcome of grasping the inward or hidden nature of things or of perceiving in an intuitive manner.” [Merriam-Webste, 2006]

In other words, in respect to Information Visualization an insight can be described as the knowledge of what stands behind the data or as the relation between the data and the problem space. In addition to this, Chris North [North, 2006] listed some important characteristics of insight in Information Visualizations:

Complex. Insight is complex, involving all or large amounts of the given data in a synergistic way, not simply individual data values.

Deep. Insight builds up over time, accumulating and building on itself to create depth. Insight often generates further questions and, hence, further insight.

Qualitative. Insight is not exact, can be uncertain and subjective, and can have multiple levels of resolution.

Unexpected. Insight is often unpredictable, serendipitous, and creative.

Relevant. Insight is deeply embedded in the data domain, connecting the data to existing domain knowledge and giving it relevant meaning. It goes beyond dry data analysis, to relevant domain impact.

North stated that a default and implicit definition is that insight is equal with user tasks such as finding important facts in the data. We can see that building up an insight is a complex process that depends on various factors.

To conclude our motivation, the target users in our in2vis project are confronted with a huge amount of abstract, but highly structured data. They want to accomplish different kinds of tasks, where they need different interactions and navigations as well as different views of the same data to gain more insight into the data under investigation. The combination of high dimensionality, an inherent time dimension, and the medical aspect make it a challenging problem. Our approach is to provide an interactive InfoVis technique with different views to help the users analyzing the data and getting new insights.

1.2 Research Question

As a basis for our research we formulated a research questions.

1.2.1 Main Question

Can we find an interactive, exploratory, and task-oriented InfoVis method to analyze abstract and highly structured, temporal data appropriately?

1.2.2 Sub Questions

- Which already known techniques are suited to analyze this complex data type?
- Can we find a novel visualization that suits our needs for exploring and analyzing the medical data user and task appropriately and focusing on the particular problem characteristic?
- Are we able to help the medical experts to find new and explore existing predictors on the course of the therapy?
- How can we profit from our experts background knowledge on the problem domain?
- What are the advantages and drawbacks of our technique by assessing the usefulness of our approach?

1.3 Goals

The main goals of our work are to address the challenges stated in Section 1.1 and in our research question in Section 1.2. Specifically, we want to

- Tackle the complex data characteristics: The high dimensional, temporal, categorical data in conjunction with the low item count makes our data set very special.
- Focus on the tasks of our intended users: Our medical personnel want to mainly find predictors on the course of therapy and explore the progress over time of different patients.
- Center the design on the users: Find specific interactions and a visual metaphor that help the users to analyze and get insights into the data.

1.4 Our Approach

Our approach tries to cope with the problem of a high dimensional, temporal data space embedded in a medical environment. We considered temporal as well as logical data aspects and applied a user-centered development approach that was guided by user input gathered via a user study, design reviews, and prototype evaluations.

For visualizing the high dimensional data set, we decided to use a spring-based 2D visualization because it efficiently reduces the dimensionality and the findings can still be traced back to the original data values (with exceptions). The important time dimension is either visualized using animation or with a specific linked view called Traces, that displays the change over time in a single image.

To represent the individual characteristics of the data objects (the exact values) we integrated multiple linked views, such as the Star Plot and Attraction Rings in addition to the spring-based core visualization.

“A general problem for all displays representing individual objects is their poor suitability for large object sets.” [Andrienko and Andrienko, 2004]

Therefore, we decided to also include linked overview visualizations to cope with large object sets and give an overview on the whole data set. We integrated two overview visualizations to help the user reveal trends and patterns in the data and then analyze this subset with the main methods.

In order to support our users in getting new insights (according to the definition of Chris North [North, 2006] mentioned in Section 1.1) of the data, we provide:

- **Multiple views:** Because an insight is complex and involves a large amount of the given data and not individual data values.
- **Interactivity:** To support the user in creatively exploring the data set and questions that build up over time we integrated a set of interactive manipulation options.
- **User centered design:** We designed our visualization specifically for our medical problem domain because insights are deeply embedded in the data domain.

This specific design for our medical problem domain was also influenced by another design decision. The target user group are therapists with a profound background knowledge of the problem domain. Therefore, we decided to develop an interactive Information Visualization that lets the user explore the data set and formulate and examine hypotheses. This is done by letting the user experiment with the data set and provide a large set of interactions that are optimized for both the users and tasks.

1.5 Dissemination

Parts of this work were presented and published at the following conferences:

- Introduction of the method at the IDAMAP Workshop, Aberdeen, Scotland, [Hinum et al., 2005a]
- Publication of Gravi++ in the Journal for Universal Computer Science (J.UCS) [Hinum et al., 2005b]
- Papers about the Usability Examination of Gravi++ [Rester et al., 2005, Rester et al., 2006a]
- Papers about our insight study [Rester et al., 2006b, Rester, 2007]
- Forthcoming diploma thesis of Martina Osztoivits [Osztoivits, 2007] on GenEdit.

Furthermore, we held various presentations as part of the in2vis project.

1.6 Overview

In the next Chapter we will present related work to our problem space and visualization techniques we used or were inspired from. In the subsequent Chapter we will discuss the medical problem domain where our project and Gravi++ are embedded. Chapter 4 deals with our visualization Gravi++ in detail. The following Chapter deals with some implementation details and gives a brief insight, why we decided to use Macromedia Flash to build our prototype. Afterwards, we will present some typical use cases of Gravi++ and our data set. Finally, we will conclude all of our work and findings in Chapter ??.

1.7 Conventions

The work presented in this document was conducted in context of the in2vis project and the Information Engineering Group¹. Even though most design and architectural ideas were carried out by myself, the research and its results were shaped based on discussions and valuable input especially by my advisers and colleagues. That is why I decided to use the pronoun “we” rather than “I” throughout this thesis.

Direct quotations are denoted by the according reference in square brackets [Author, Year]. If ideas or concepts of others are referred to, the respective reference is put at the appropriate position within a sentence, or right after the end of a sentence if the whole paragraph is meant to be related to the given reference.

¹<http://ieg.ifs.tuwien.ac.at/>

Chapter 2

Related Work

As a first step in our project we conducted a state-of-the-art research for existing visualization methods that could be applied to our problem characteristics. Our data format is very complex because of the high dimensionality, time dimension, medical relatedness, and the ordinal categories of each dimension. Therefore, we also analyzed techniques and methods that only partly fit our problem domain.

In the first part of this Section we will discuss important visualization principles. After that we will list all visualization techniques and applications that are (partly) suited to our problem domain. Finally, we will discuss our findings and relate them to the development of Gravi++.

2.1 Visualization Principles

In this Section we want to present the most important visualization principles we have used for our visualization technique or are related to it. First, we will present some basic information about cognitive psychology because of its relevance for the perception of an Information Visualization. Next, we will shortly present some important visualization frameworks as they are an important and helpful starting point when creating a visualization. Thereafter, we will introduce the three most important interaction principles for our visualization, namely Details on Demand, Linking and Brushing, and Focus+Context. Time oriented information is the topic of the next part, as it is an outstanding dimension in our data set. Then, we will give an introduction to animation, as it is one of our ways to visualize time. After that, we will present the principles color, metaphor, and small multiples. Finally, we will first talk about 2D versus 3D visualizations and introduce the spring-based algorithm that we have used for our core visualization.

2.1.1 Cognitive Psychology

Information Visualizations depend mostly on visual perception and information processing. Therefore, cognitive psychology is very important for understanding the process from seeing coloured pixels to abstracting information and getting insight into the data which is one main principle of InfoVis techniques. Lanzemberger provides a good overview on related work in cognitive psychology [Lanzemberger, 2003]. Her work is mainly based on literature by Benjafield and Anderson [Benjafield, 1992, Anderson, 1989], a theory of visual perception by Gibson [Gibson, 1950, Gibson, 1986], and other important issues like attention, pre-attentive processes, automatic and controlled processes, perception, and associated concepts.

The Gestalt Laws [Koffka, 1935] (“Gestalt” is the German word for pattern) are seven basic rules that describe the way we see patterns in visual displays.

1. Proximity: Graphical objects that are close together are perceived as a group (Figure 2.1).
2. Similarity: Similar elements tend to be perceived as grouped (Figure 2.2).
3. Continuity: Smooth and continuous visual elements are more likely to be perceived as an entity than ones that contain abrupt changes in direction (Figure 2.3).
4. Symmetry: Symmetrical figures tend to be grouped to an object (Figure 2.4).
5. Closure: The human brain tends to close contours that have gaps in them.
6. Relative sizes: Smaller components of a pattern are perceived as objects whereas the bigger components are seen as background (Figure 2.5).
7. Figure and ground: During perceiving a visual field, some objects take a prominent role (the figures) while others recede into the background (the ground). The visual field is thus divided into these two basic parts (Figure 2.6).

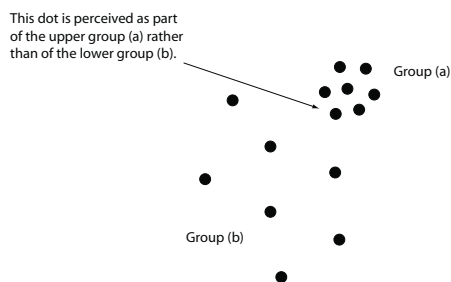


Figure 2.1: Gestalt Law - Proximity

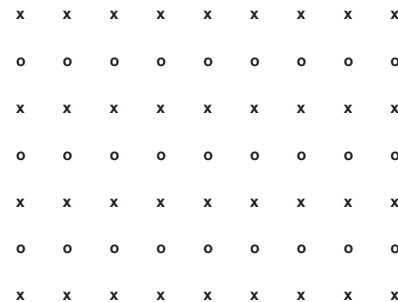


Figure 2.2: Gestalt Law - Similarity

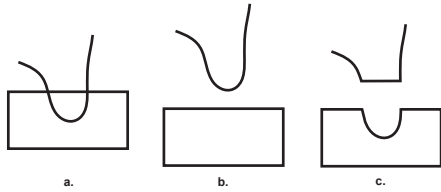


Figure 2.3: Gestalt Law - Continuity

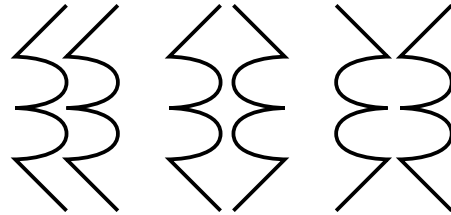


Figure 2.4: Gestalt Law - Symmetry

Further information about the Gestalt Laws can be found in Colin Ware’s book *Information Visualization* [Ware, 2000] Chapter 6 (page 203), Mads Soegaard’s article on *Interaction-Design.org* [Soegaard, 2005] and in my master thesis [Hinum, 2004].

Further basics about the human perception (especially about the cognitive processes) can be found in the book “*Dynamische Visualisierung*” [Buziek and Dransch, 2000].

2.1.2 Visualization Frameworks

Because of the dedicated time support, the TimeViz Framework [Aigner, 2006] (see Figure 2.7) is the most appropriate model to create an interactive time-based InfoVis application. It aims for supporting the interactive visual exploration of temporal data over a broad range of domains and applications. It is based on the Information Visualization Reference Model as presented in [Card et al., 1998] and extends it by the denotation of temporal attributes via meta data along a versatile time model.

Apart from this, other models of the visualization process exist for visualization in general [Schumann and Müller, 2000, Robertson and Ferrari, 1994] or for InfoVis in particular [Chi, 2000]. These models can be good starting points for the creation of a new Information Visualization.

The Visual Information Seeking Mantra [Shneiderman, 1996] by Ben Shneiderman summarizes many visual design guidelines and provides a framework based on a “task by data” type taxonomy. It is a useful starting point for designing user interfaces and interaction mechanisms in InfoVis tools. Craft and Cairns [Craft and Cairns, 2005] analyzed the Mantra and reviewed the current literature that references it. The results indicated a need for empirical validation of the Mantra.

2.1.3 Interactions

As our focus lies on developing an “interactive” Information Visualization, the interactions are of special interest. The Information Visualization Reference Model [Card et al., 1998] discriminates three kinds of interactions:

1. **Data Transformations:** Transform the raw data into one or more data tables (e.g., in a relational database) for further processing. Here selection of the data, filtering at an early stage, or changes on the data itself could be possible interactions for a user.
2. **Visual Mappings:** The user can define how the data is mapped to the visual objects. Interactions in this step could be selecting an appropriate visualization, choosing the dimensions or values which should be displayed or changing aspects (e.g., the color) of the final representation for specific data attributes.
3. **View Transformations:** The user can change the visual representation (for example navigate in 3D space or change the physical layout of the visual items).

A general introduction to human-computer interaction can be found in the book “*Interaction Design*” [Preece et al., 2002]. Specific information for interactions in Information Visualizations can be found

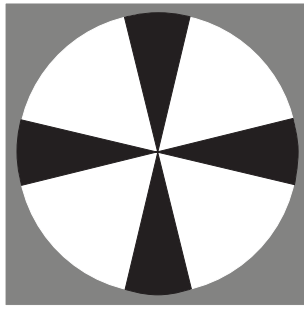


Figure 2.5: Gestalt Law - Relative sizes

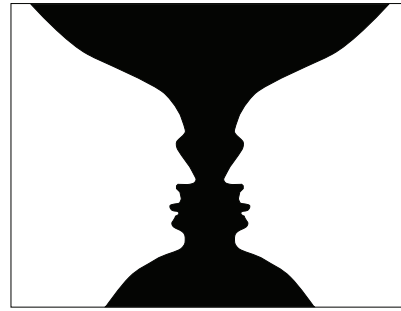


Figure 2.6: Gestalt Law - Figure and ground

in [Spence, 2001, Chapter 2] and [Ware, 2000, Chapter 10]. Particular examples are Focus+Context, Linking and Brushing, and Details on Demand, which we will present now in more detail.

2.1.3.1 Details on Demand

Details on Demand stands for techniques that provide more detail on the data set only after the user requested them. Usually the user gets a better overview on the data set because of the reduced amount of information but still is able to grasp the “details on demand”.

The last part of Shneiderman’s Visual Information Seeking Mantra [Shneiderman, 1996] is the “Detail on Demand” part. He states that, because of limited screen space and high data complexity visualizing every detail can be impractical.

“The details-on-demand technique provides additional information on a point-by-point basis, without requiring a change of view. This can be useful for relating the detailed information to the rest of the data set or for quickly solving particular tasks, such as identifying a specific data element amongst many, or relating attributes of two or more data points. Providing these details by a simple action, such as a mouse-over or selection (the “on-demand” feature) allows this information to be revealed without changing the representational context in which the data artifact is situated.” [Craft and Cairns, 2005]

A typical example for additional information on a point-by-point basis is a tooltip with additional information that is displayed when the user hovers over a data item in a visualization with the mouse cursor. With this method the user can acquire detailed information with a simple action and without changing the representational context (by overlaying the information under the mouse cursor for example).

2.1.3.2 Linking and Brushing

Linking is the highlighting of the same data-points in different views of the data. For example, displaying highlighted data points with the same color in all views. Brushing extends this concept a bit further, by letting the user interactively select the points to be highlighted (e.g., by mouse click) [of Standards and SEMATECH, 2006]. Lanzenberger [Lanzenberger, 2003] mentions two features of Linking and Brushing:

- “1. Insights are provoked by seeing an object or item in relation to different contexts. Linking and brushing allows to access items of the visualization according to group selections or individually. To really benefit from multiple views it is essential to make the data items graspable and distinguishable from the rest.
2. The user engages in a process of repeated selection and highlighting and learns about the connection of her or his actions in one view with the data in another view. So actions of the user are propagated in distinct views of the data. This process helps the user in exploring the structure of the data.” [Lanzenberger, 2003, Chapter 2, page 15]

Visualizations that integrate the concept of Linking and Brushing are also called Coordinated Multiple Views (see section 2.2.5) because of the linked views that are presented.

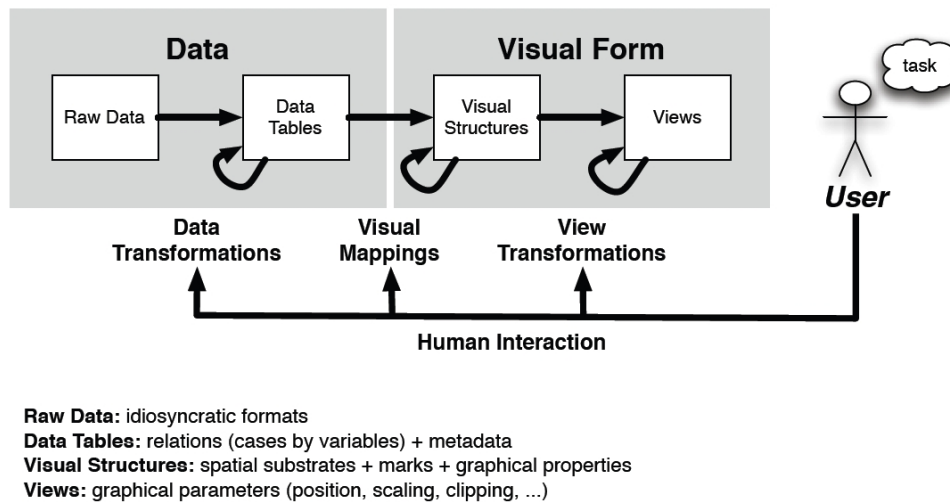


Figure 2.7: Information Visualization Reference Model [Card et al., 1998] - Visualization as a step-wise transformation from raw data into views - Image from [Aigner, 2006]

2.1.3.3 Focus+Context

Focus+Context (F+C) techniques enable the user to investigate specific details of the data while at the same time also providing an overview over the embedding of the data under investigation within the entire data set [Kosara, 2001]. Kosara gives a good overview on F+C techniques in his Ph.D. thesis [Kosara, 2001]. The most well known F+C techniques are Distortion Techniques [Leung and Apperley, 1994]. They are magnifying important parts while compressing less important ones. A typical example is the Fisheye Views technique [Furnas, 1986, Sarkar and Brown, 1992] that is based on the concept of fisheye lenses in photography. A fisheye lens is an ultra wide-angle lens that has an uneven distribution of magnification (objects in the middle appear larger than on the edge of the lens).

2.1.4 What is Time-Oriented Information

The dimension time differs fundamentally from the space dimension in terms of traversability and perceptibility. In the real world, it is only unidirectional - we cannot go back or forward in time - and it is perceived differently to the space dimensions. These facts make it a challenging task to visualize the invisible time.

General definitions about time-oriented information are given by C. Jensen and R. Snodgrass [Jensen and Snodgrass, 1999] as well as C. Daassi et al. [Daassi et al., 2004]:

“Temporal data is data that varies over time.” [Jensen and Snodgrass, 1999]

“A temporal data denotes the evolution of an object characteristic over a period of time.” [Daassi et al., 2004]

Aigner gives in his dissertation detailed information about time-oriented data and provides a model and framework for visualizing it. In his opinion

“... using special methods for treating time is necessary in order to accomplish the important tasks of revealing trends, as well as identifying patterns and relationships in the data.” [Aigner, 2006]

2.1.5 Animation

Animation is an important aspect of our Information Visualization. We will, therefore, discuss this topic more in detail. The word animation stems from the Latin word “animare” that means alive or animate. The term “Computer Animation” (CA) stands for the computerized creation of moving images.

We will now discuss some perceptive basics on the human perception of animation.

2.1.5.1 Perceptive basics

The human eye detects subsequent images as a change and not as discrete images. Wertheimer called this the Phi-Phenomenon [Wertheimer, 1912]. This phenomenon comes from the fact, that our eyes are not able to perceive a fast sequence of images as discrete images. The sequence is perceived as a continuous entity. This continuous change is also called “Scheinbewegung” (pseudo - movement) [Vernon, 1974, Marr, 1982, Rock, 1985]. Wertheimer showed that two simple stimuli (e.g., points or lines) in a short period of time on different locations were perceived frequently as a movement by a set of test persons. In addition to this, more complex pseudo movements like change of form can be perceived. For example, the change of form of a rhombus to a circle [Squires, 1959]. The perceivment of this pseudo movement depends on the distance and the time interval between the two stimuli [Korte, 1915]. In other words, some patterns in the data can only be perceived when the time interval between two stimuli is short enough [Dransch, 1997].

2.1.5.2 History

The following short list shows some highlights of the history of animation.

- 1831 - Joseph Antoine Plateau: First image animation on a “phenakistoskop” (spinning disc with different images).
- 1906 - First animated film “Humorous Phases of a Funny Face”
- 1951 - First computer animation of a jumping ball [Willim, 1989].
- 1982 - 15 minutes of the film “TRON” were fully generated by a computer.
- 1985+ - Increasing (consumer) computer power lead to a wider application of computer animation.

Especially in the last years, the graphics power even of the cheapest consumer computers is capable of displaying complex 3D graphics in real time (mainly because of the spread of computer games). This paved the way for the development of visually complex computer animations on consumer computers.

CA can be divided in two different classes - temporal and non-temporal animation.

2.1.5.3 Temporal versus Non-Temporal Animation

Temporal animation is the use of animation to show the differences in the data over time. That means a direct mapping of time to the animation time (linear mapping not obligatory). Non-temporal animation, on the other hand, is the visualization of other dimensions (than time) with the animation. Dransch [Dransch, 1997] defined these two terms in regard to spatial (cartographic) data, but the definition is also valid for non-spatial data. As our data set is transferred to spatial data (with a spring-based algorithm) the definition can also be used in our case without generalization.

This temporal or non-temporal animation can be used for exploration, verification and demonstration tasks. [Dransch, 1997]

2.1.5.4 Computer Animation for Exploration Tasks

The phenomenon of pseudo movement can be used to identify structures. Marr mentioned:

“Our visual system has remarkable powers to recover the shapes of unknown structures simply from the way their appearances change in the image.” [Marr, 1982]

Experiments by Ullmann [Ullmann, 1979] showed that moving points can be recognized as three dimensional structures. Johansson [Johansson, 1973] marked persons with points (light sources) and filmed them in the dark. The resulting point cloud was not identified as a still picture but with movement the human body was imaginable. Braddick [Braddick, 1980] used a sequence of varying images with unstructured points that formed clusters. The viewers saw a moving center, a not moving outer rim and a borderline between those two areas.

Kumiyo et al. [Nakakoji et al., 2001] argue that

“. animation has more power than just yet another dimension to 3D representations.”
[Nakakoji et al., 2001]

They explored what cognitive effects animated Information Visualization has and what interactions people would like to have with animated visualizations when exploring data. They found out that animation gets positive reactions for chunking data, interpreting data, expecting the following, comparing, and focusing / filtering, but not favored in aspects including grasping the whole and statistical analysis. Furthermore, post-experimental interviews showed, that animation was helpful in focusing on the changes of data, and in limiting a point of view in a single aspect.

From these findings we can conclude that Animation can help in the exploration of data sets by identifying structures over time. Especially the time dimension can be visualized using animation because of the inherent reference.

An example for the exploratory use of animation in Information Visualizations is the “Animator” by Barlow and Stuart [Barlow and Stuart, 2004], a tool for animating Parallel Coordinates. The paper presents an uncommon use for Parallel Coordinates in which animation is used to view multi-dimensional data sets over time. Another example for the use of animation is the Cichlid tool by Brown et al. [Brown and McGregor, 2000]. It allows the users to view data sets in 3D and shows changes over time using animation.

2.1.5.5 Computer Animation for Verification

As shown for exploration tasks, computer animation can also be applied for visualizing data. In contrast to the exploration, in the verification the parameters are specifically altered according to the formulated hypothesis. This is done to test the behavior of the system. Subsequent to this, statistical methods can be used for a more detailed examination.

2.1.5.6 Computer Animation for Demonstration

Koussoulakou and Kraak [Koussoulakou and Kraak, 1992] showed that spatial and time-related changes can be perceived faster with animation than on a static map. In addition to this, animation can support the human memory. Jacobs [Jacobs, 1992] made clear, that spatial and time related changes that were shown with an animation were remembered longer than the same information visualized on a static map. Jacobs ascribed that with the help of animation, dynamic mental images were created in the users mind. Therefore, the information is memorized as a whole (the whole process).

Animation can also be used to display complex information sequentially in a logical order. That way, the user is introduced step by step in the process. [Dransch, 1997]

Finally, animation can lead to an enhanced attention and consequently to an increased capacitance of the presented information of the user. Studies have shown that a homogenize stimulation without any variation leads to a decreased alertness and attention [Vernon, 1974, Dember and Earl, 1957]. Whereas, a varying stimulation increases the attention. For example, moving objects are attracting the attention of infants straight after their birth [Hershenson, 1964]. The variations during an animation lead to the necessary changes of the stimulation. Therefore, animation can lead to an increased attention of the viewer and is therefore very suited for demonstration purposes.

2.1.5.7 Optimal Speed

The optimal speed of a computer animation is an important factor for the perception of the data. Lind and Kjellin investigated in [Lind and Kjellin, 2005] the relationship between mean speed of animated objects' speed and the ease of learning a complex relationship defined between the moving objects. They found out a significant interaction between difficulty and speed. Their first results indicate that there seems to be a range of speeds that are optimal for discovering complex spatio-temporal patterns hidden in animations. Furthermore, this range of speeds is much higher than expected.

2.1.5.8 Conclusion

From the above presented facts we can conclude, that animation (if efficiently used) can be helpful in visualizing a further dimension (especially time). The optimal speed for presenting the animation

is a significant interaction between difficulty and speed [Lind and Kjellin, 2005]. Therefore, a general visualization should let the user decide at what speed the images should be presented.

2.1.6 Color

Color helps us to break camouflage and is excellent for labeling and categorization, but poor for displaying shape, detail, or space [Ware, 2000]. Labeling and categorization are two important aspects of Information Visualizations. Therefore, color also plays an important role for visualizations.

General information about colors and their impact on the viewer can be found in “Farben, Natur Technik Kunst” by Welsch and Liebmann [Welsch and Liebmann, 2003] They provide a good overview about the symbolic meanings of colors, the color models, the existence of color in nature, the human perception, and colors in physics and technics. Colin Ware also gives a good introduction to colors in his book “Information Visualization“ [Ware, 2000, Chapter 3 and 4] as well as Tufte’s “Envisioning Information” [Tufte, 1990].

Information Visualizations use color for two purposes

1. for user interface elements and
2. more important for visualization elements (the data representations).

Our main focus lies of course on using color for the visualization elements. Colin Ware proposes 12 different colors for labeling, which are reasonably far apart in color space and can therefore be easily distinguished. Furthermore, Robert Spence [Spence, 2001] gives a short introduction for the use of color in Information Visualizations.

A good general reminder for using color comes from Robert Spence, who states:

“Use of color should be undertaken with caution, however, and not with the hysterical abandon of a child discovering its first paint box.” [Spence, 2001, page 66]

2.1.7 Metaphor

Marcus [Marcus, 2002] defined a metaphor as:

“Metaphor: Fundamental concepts communicated via words, images, sounds, tastes, smells, and tactile experiences.

In computer operating systems, metaphors substitute for collections or individual elements and help users understand, remember, and enjoy the entities and relationships of computer-based communication systems. Metaphors can be overarching or communicate specific aspects of user interfaces. An example of an overarching metaphor is the desktop metaphor that substitutes for the computer’s operating system, functions, and data. Examples of specific concepts are the trashcan, windows and their controls, pages, shopping carts, chat rooms, and blogs (Weblogs, or Webbased diaries). The pace of metaphor invention, including neologisms “that is, verbal metaphor invention” is likely to increase because of rapid development and distribution, through the Web and mobile devices, of mutable products and services. Some researchers are predicting the end of the desktop metaphor era and the emergence of new fundamental metaphors.” [Marcus, 2002]

Benyon and Imaz discuss their article [Benyon and Imaz, 1999] where abstractions that designers employ come from and how such abstractions are related to the concepts that the users of these systems have. In particular, they use the theory of experientialism as the foundation for their analysis.

Further information about metaphors can be found in “Metaphors We Live By” [Lakoff and La, 1980]. Two examples for the use of metaphors in Information Visualizations can be found in [McGinn and Picking, 2003] and [Alty et al., 2000].

2.1.8 Small Multiples

Edward Tufte characterizes “Small Multiples” in the following way:

“Illustrations of postage-stamp size are indexed by category or a label, sequenced over time like the frames of a movie, or ordered by a quantitative variable not used in the single image itself.” [Tufte, 1990]

Tufte calls these small illustrations “Small Multiples” that are visually enforcing comparisons and reveal, all at once, a scope of alternatives, a range of options. Examples for Small Multiples are many Glyph based visualizations (see Section 2.2.3) and also map based visualizations (e.g., painting differently sized dots on a map for visualizing some sort of parameter).

Because of the small size, Small Multiples can often only convey an overview over the data. Combined with Detail on Demand they can still cover all levels of detail and enable full analysis of a large data set.

2.1.9 2D Versus 3D Visualizations

3D visualizations have the advantage of a third dimension to visualize data. The drawback is the additional effort to find the perfect perspective (that is given in 2D visualizations) [Wyeld, 2005]. 3D representations often show the inability to visualize all relationships between data which we know exists but are obscured. By changing the perspective we can detect obscured items but we have to remember what we saw first.

2.1.10 Spring-Based Positioning

Spring algorithms [Eades, 1984, Eades and Lin, 2000] are regarded as effective tools for visualizing undirected graphs. The basic idea behind spring algorithms for drawing graphs is to simulate a “balanced state” of a mechanical system. A spring system consists out of two parts:

1. The mechanical spring model for the graph
2. A technique for finding a minimal energy state of the spring system resulting in the final positioning of the graph items.

Most of the existing spring systems use a generalization of Hooke’s Law [Sears, 1970].

In physics, Hooke’s Law states that the amount by which a material body is deformed is linearly related to the force causing the deformation. Materials for which Hooke’s law is a useful approximation are known as linear-elastic (e.g., steel but not rubber) and they obey, that the extension produced is proportional to the load:

$$F = -kx$$

F ... Force exerted by the spring

k ... Spring constant (large for a stiff spring, small for a soft spring)

x ... Displacement from equilibrium position

2.2 Visualizations

In the following section we will present techniques that are (partly) suited to cope with our problem domain and the related data set. The visualization approaches are parted in the following five categories:

1. Visualization methods in Statistics and Mathematics
2. Geometric techniques
3. Glyphs
4. Pixel-based visualizations
5. Coordinated Multiple Views

This classification is leaned on Keim's categorization in [Keim and Kriegel, 1996], but leaving out the hierarchical and graph-based techniques because they are not suited for our data structure. The Coordinated Multiple Views are added because the intelligent combination of single techniques is especially interesting for our highly structured temporal-data. Parts of this work are taken from Aigner's comprehensive survey on available visualization techniques for time, temporal data, and temporal information [Aigner, 2006].

2.2.1 Visualization methods in Statistics and Mathematics

In this Section we present Exploratory Data Analysis [Tukey, 1977] by John W. Tukey. It is a well known and often used approach to get insight into data and one main part of our main study (see Section 7.2). Furthermore, we will discuss dimension reduction techniques because they are one traditional mathematical approach to cope with highly scattered data sets.

2.2.1.1 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) was introduced by John W. Tukey [Tukey, 1977] to analyze data when there is only a low level knowledge about its cause system as well as contextual information.

“Exploratory Data Analysis (EDA) is detective work - numerical detective work - or counting detective work - or graphical detective work ... unless exploratory data analysis uncovers indications, usually quantitative ones, there is likely to be nothing for confirmatory data analysis to consider ... [it] can never be the whole story, but nothing else can serve as the foundation stone - as the first step.” [Tukey, 1977, pages 1-3]

This definition by Tukey makes it clear that EDA is not identical to statistical graphics although it makes heavily use of them.

“EDA is not a mere collection of techniques; EDA is a philosophy as to how we dissect a data set; what we look for; how we look; and how we interpret.” [Filliben, 2006a]

The most well known and for our problem domain most important techniques are

- *Box Plot* (also known as Box-and-Whisker Diagram) [Chambers et al., 1983]: The Box Plot is a way of graphically depicting the five-number summary, which consists of the minimum, maximum, lower quartile, median, and upper quartile of the data. It can be used to visually compare different dimensions (e.g., timesteps, populations, ...) and help indicate variance, skew, and outliers.
- *Histogram* [Chambers et al., 1983, Harris, 1999]: Its purpose is to visualize the distribution of a data set. For this purpose the data set is specified into categories (usually non-overlapping intervals of a dimension). Bars are visualizing the proportion of cases falling into each category. A typical example for histograms is the use in photography to visualize the distribution of brightness or color values. The number of pixels for each level of brightness is counted and visualized as adjacent bars. Therefore, the photographer can easily see the distribution of brightness and maybe correct it with an appropriate program.

- *Scatter Plot* (see Section 2.2.1.2)
- *Mean Plot*: Used to compare means of between different groups of data. The mean values are simply plotted according to another dimension (e.g., time) on a plot. Using time on the x-axis we can see the changing mean value over time (e.g., for different groups different lines).
- *Stem-and-Leaf Plot* (or Stemplot): Similar to a Histogram but containing more information because the exact values are also visualized. The individual data values are displayed using a table-like format. The Stemplot is parted in the stem column on the left and leaf column on the right. In the stem column the first digits create categories. For example “1” standing for all values between “10” and “19”. In the leaf column the remaining digits are printed in order of increasing magnitude separated by a blank.

Due to the graphic possibilities of modern computers the histograms has mostly substituted the Stemplot.

Example: Visualize the values: 03, 05, 07, 23, 29, 33, 33, 34, 37, 39, 51

```

0 | 3 5 7
1 |
2 | 3 9
3 | 3 3 4 7 9
4 |
5 | 1

```

Other graphical EDA techniques like the Autocorrelation Plot or the Bihistogram can be found in the eHandbook of Statistical Methods [Filliben, 2006b]

The traditional EDA methods provide no interactivity and no inherent time support. That means visualizing time-based data leads to many plots that have to be analyzed. On the positive side is that the EDA methods are widely known and understood.

2.2.1.2 Scatter Plot / Scatter Plot Matrix

A Scatter Plot (Scatterplot) or Scatter Graph [Chambers et al., 1983] is a graph to visually compare two dimensions in the data. The data is displayed as points on an x- and y-axis according to their values on the two mapped dimensions. Because of this comparison the Scatter Plot reveals relationships or associations between the two dimensions. The Scatter Plot has been extended, for example, by Bertini and Santucci [Bertini and Santucci, 2005] to reduce clutter. This has been done by using pixel displacements and non uniform sampling.

The Scatter Plot Matrix displays a matrix of Scatter Plots next to each other to visualize more than two dimensions. The single Scatter Plots in the matrix contain all dimensions pairwise. That means, that if there are k dimensions to be visualized, the Scatter Plot Matrix will have k rows and k columns. Variable X_i and Y_j are then plotted against in the i -th row and j -th column.

While each off-diagonal plot maps a pair of non-identical dimensions, there is no prescribed mapping for the diagonal plots. This area can be used to display other relevant information (visualizations). For example, Cui et al. [Cui et al., 2006] describe how to integrate histograms, 1D plots and 2D plots (Raster Plots and Data Driven Plots) in the diagonal plots of the Scatter Plot Matrix.

2.2.1.3 Dimension Reduction Techniques

A traditional approach to examine multi-dimensional data sets is to reduce their dimensionality. This can be performed by applying feature extraction where the multidimensional space is mapped into a space of fewer dimensions. This can be done with transformations such as Principal Component Analysis (PCA),

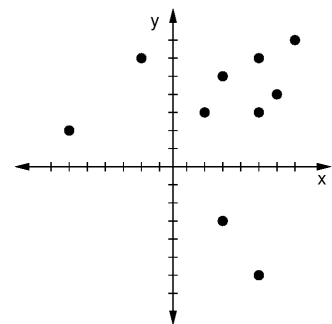


Figure 2.8: Scatter Plot [Chambers et al., 1983]

Factor Analysis, and Multidimensional Scaling. Many of these methods are subsumed under the term “Projection Pursuit” [Huber, 1985, Friedman and Tukey, 1974].

Principal Components Analysis (PCA) PCA is a classical statistical method for expressing two or more variables by a single factor. That means it is used to simplify data sets. It is one of the oldest and most prominent techniques for multivariate data analysis [Jolliffe, 1986]. PCA reduces the dimensionality of a data set while retaining as much variation as possible using so called “Eigenvalues”. For example an n -dimensional data set can be transformed to a 2-dimensional one using PCA and visualized afterwards with a Scatter Plot. This dimensionality reduction of the data set is done with the intention to retain the characteristics of the data set that contribute most to its variance.

“In essence, PCA seeks to reduce the dimension of the data by finding a few orthogonal linear combinations (the PCs) of the original variables with the largest variance.” [Fodor, 2002]

Factor Analysis (FA) Like PCA, FA is also a linear method. It was first suggested by psychologists and assumes that the measured variables depend on some unknown, and often immeasurable, common factors. The goal of FA is to uncover such relations, and thus it can also be used to reduce the dimensionality of data sets [Fodor, 2002].

Projection Pursuit (PP) Projection Pursuit [Huber, 1985, Friedman and Tukey, 1974] is also a linear method like PCA and FA, but incorporates higher than second-order information. Therefore, it is useful for non-Gaussian (distributed) data sets. It is also more computationally intensive than second-order methods. Given a projection index that defines the importance of a direction, PP looks for the directions that optimize that index. As the Gaussian distribution is the least interesting distribution (having the least structure), projection indices usually measure some aspects of non-Gaussianity [Fodor, 2002].

Multidimensional Scaling (MDS) Multidimensional Scaling [Borg and Groenen, 1997] can be considered to be an alternative to Factor Analysis. It attempts to arrange “objects” in a space with a particular number of dimensions to reproduce the observed distances [StatSoft, 2003]. In contrast to Factor Analysis, that requires that the similarities between objects are expressed in the correlation matrix, MDS can analyze any kind of similarity or dissimilarity matrix, in addition to correlation matrices. In technical terms, MDS uses a function minimization algorithm that evaluates different configurations with the goal of maximizing the goodness-of-fit.

Visualization of reduced feature dimensions data sets The resulting data from the dimension reduction algorithms is often visualized using geometric techniques. In many cases only a Scatter Plot is used to plot the data values as points on the plane and visualize clusters and outliers. For example, Yang et al. show in their visualization VisHD [Yang et al., 2005] a 2D projection of the corresponding vector of various dimension reduction techniques (PCA, FA, etc.).

Conclusion According to Crawford [Crawford and Fall, 1990] dimension reduction techniques have many limitations. For example, it is difficult to determine just what the solutions from automatic projection pursuit methods actually mean. In our point of view, these traditional techniques offer only limited time support and interaction possibilities and are therefore only partly suitable to analyze our data thoroughly.

2.2.2 Geometric Techniques

2.2.2.1 Sequence Graph

The Sequence Graph [Harris, 1999] (also known as Time Series Graph or Time Line Chart) is the most well known visualization for representing time and a data dimension. The main purpose of this graph is to graphically display time-related information. The x-axis is usually used to display the temporal ordering and is sometimes called chronological scale. The data values are mapped on the y-axis and are represented either by points alone, points with connecting lines or bars.

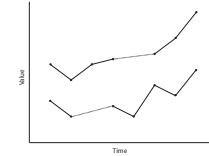


Figure 2.9: Sequence Graph [Harris, 1999]

The benefits of the Sequence Graph are that it is very well known and easily to perceive and understand. The drawbacks are that only few data dimensions can be visualized (using the y-axis, color, and shape).

2.2.2.2 Parallel Coordinates Plot (PCP)

The Parallel Coordinates Plot (PCP) or simply Parallel Coordinates [Inselberg, 1985, Inselberg et al., 1987, Inselberg and Dimsdale, 1990, Inselberg, 1998] is one of the most well known techniques used for visualizing multivariate data sets. Multiple parallel parameter axes are standing side by side on the screen. Each value on each axes of each parameter is connected by a line. That means that each data object is represented by polygonal lines connecting points (“coordinates”) on parallel axes, which represent attributes. These connecting lines can be painted according to class colors. Inselberg describes it as a “geometric problem solving technique”. The order of the axes has an important impact on the visual appearance. A history of Parallel Coordinates can be found in [Inselberg, 2006].

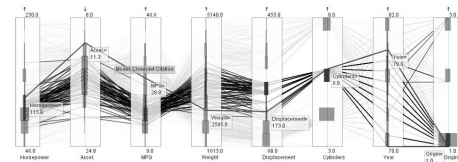


Figure 2.10: Parallel Coordinates [Hauser et al., 2002]

In early papers, various mathematical and algorithmic properties of parallel coordinates were presented [Inselberg, 1985, Inselberg et al., 1987]. Later Inselberg and Wegman described the parallel coordinates plot as a tool for EDA [Inselberg and Dimsdale, 1990, Inselberg, 1998, Miller and Wegman, 1991, Wegman, 1990].

Various interactive extensions exist for the PCP allowing manipulations, queries and added information like display of aggregated data into PCP and thereby make it more appropriate for large data sets. For example, Angular Brushing [Hauser et al., 2002] enables the user to highlight items according to angle changes of the lines. Siirtola [Siirtola, 2000] adds Box Plots, polyline averaging (summarize a set of polylines), and the correlation coefficients between polyline subsets to the PCP, helping the user to discover new information in the data set. Displaying Histograms on the axes [Pratt and Tschapek, 2003] or showing envelopes of subsets of lines [Inselberg, 1985, Inselberg et al., 1987] are two other extensions that convey further information about the data set and help to cope with a high amount of data.

Andrienko and Andrienko [Andrienko and Andrienko, 2004] suggested envelopes for clustered objects that are divided in stripes (into 10 equal-frequency subintervals) and adding ellipse plots (instead of Box Plots, which show medians and quartiles) on the axes. They use ellipses rather than boxes because two adjacent ellipses meet just in a single point and, hence, can be easier distinguished visually. A weakness of their approach is that the “striped” envelopes and ellipse plots only convey summary information for each attribute (axis) independently of others. Therefore, they are not suited to investigate relationships between the attributes and cannot explore the distribution of value combinations. Thus, they cannot convey information about the “typical profiles” of class members.

Novotny presented in [Novotny, 2004] a visual abstraction of PCP using clustering. He visualized the different clusters using colored polygonal areas representing the bounding box of the cluster (similar to Andrienkos’ approach [Andrienko and Andrienko, 2004]). His work is based on previous work by Fua et al. [Fua et al., 1999] about Hierarchical Parallel Coordinates.

Johansson et al. extended the Parallel Coordinates to the third dimension to simultaneously examine the relationships of a single data dimension with many others [Johansson et al., 2005]. They position one axis in the center that is connected with all other axes, which are aligned star shaped on the screen. To aid this method, they introduced a technique called relation spacing which is used to position the axes according to how interesting the different relations are. Furthermore, he also integrated clustering to reduce clutter.

Barlow and Stuart [Barlow and Stuart, 2004] animated the Parallel Coordinates to enhance the understanding of how objects within the multi-dimensional space change over time.

A problem of the parallel coordinates is the overlap of data items when too many data items are visualized. Keim showed this in [Keim and Kriegel, 1996] with 15.000 six-dimensional data items where only one of three clusters was visible (due to the overlapping). Many of the above mentioned extensions try to deal with this problem by adding further information on the data set to the PCP. Furthermore, the comparability of different dimensions is best for adjacent axes (because of the directly connecting lines). The time dimension can be integrated with an axis, different colors, or animation, whereas animation seems to be the best technique because of a reduced amount of visible lines at a time.

2.2.2.3 ThemeRiver

The ThemeRiver [Havre et al., 1999, Havre et al., 2000, Havre et al., 2002] visualization is used to analyze thematic changes of document collections over time. It utilizes the metaphor of a “river” that flows through time. The width of the river changes to reflect thematic changes of themes/topics which are represented as colored “currents”. As it tries to overcome the limitations of Histograms it is also possible to display other data values than document changes. Furthermore, ThemeRiver representations can be annotated, e.g. with time-related information, and raw data points with exact values can be shown. A modification of this technique using a 3D representation to visualize a further attribute has been developed by Imrich et al. [Imrich et al., 2003].

As the traditional Histogram the ThemeRiver is only capable of displaying a limited amount of data dimensions. Furthermore, our data set uses only a limited and discrete number of timesteps, therefore the fluid changes in the ThemeRiver could be interpreted wrong. In addition to this, the time-annotations are not important in our psychotherapeutic data domain.

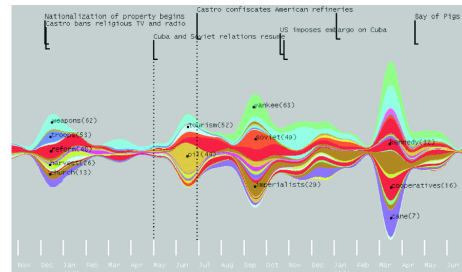


Figure 2.11: ThemeRiver [Havre et al., 2002]

2.2.2.4 Self Organizing Maps (SOMs)

The Self-Organizing Map (SOM) [Kohonen, 2001] is a data mining technique especially useful for clustering and visualizing relationships. The SOM performs a mapping to a two-dimensional map using an unsupervised neural network algorithm such that the pair wise similarities are preserved as faithfully as possible. Furthermore, there are methods to visualize the density in the SOM as well as groups of similar data patterns, statistical characteristics of the data distribution, and correlations between the variables [Pözlbauer et al., 2005]. The algorithm computes the mapping based on how similar the patterns in the variables are to each other.

A visualization of all data patterns mapped to the SOM is the Hit Histogram or Frequency Plot. For each position on the map (there is a fixed number of discrete map nodes, the axes are not continuous), a rough average value is provided that best describes the patterns that are projected to it and around it. Thus, a single variable can be plotted for the whole map. This kind of visualization is called Component Plane, and it can be learned from it which areas correspond to which values for a single variable. A visualization that summarizes many of the characteristics is the U-Matrix. It gives information on how dissimilar neighboring units are, enabling the observer to identify cluster boundaries (a cluster is a region of elements that are highly similar to each other and highly different to elements not within the cluster).

As the visualized maps are using the output of a data mining technique, the SOMs do not visualize the data itself, but the outcome of the algorithm. This is similar to the dimension reduction algorithms (described in Section 2.2.1). Therefore, similar problems such as interpreting the final results for non-experts arise. Furthermore, interactivity of the technique is limited.

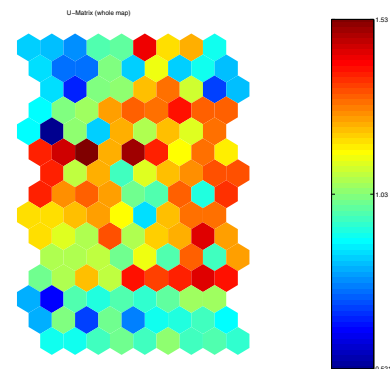


Figure 2.12: SOMs U-Matrix [Pözlbauer et al., 2005]

2.2.2.5 Hierarchical Techniques

Hierarchical techniques such as Dimensional Stacking [LeBlanc et al., 1990] (sometimes also called Stacked Display) share the basic idea of visualizing data using a hierarchical partitioning into subspaces. Dimension Stacking divides the n -dimensional attribute space in 2-dimensions which are “stacked” into each other. That means that the first two dimensions part the screen into n -regions (classes). In each region comes the next two dimensions. This process is repeated until all dimensions are visualized and the data point can be drawn on the screen (like in a Scatter Plot) [LeBlanc et al., 1990]. Dimensional Stacking is especially adequate for data with ordinal attributes of low cardinality.

Other hierarchical techniques are for example:

Worlds within Worlds [Feiner and Beshers, 1990]: A 3D representation where the 3D plots are positioned within a 3D plot (nested coordinates).

Treemaps [Shneiderman, 1992]: A 2D representation for visualizing trees with weights or sizes on the leaf nodes. These weights or sizes are represented by accordingly spaced, colored rectangles which are positioned recursively according to their position in the tree.

Cone Tree / Cam Tree [Robertson et al., 1991]: A 3D visualization of a tree structure, where all subnodes are arranged in a cone like position below (Cone Tree) or to the right (Cam Tree) of the upper node. Furthermore, interactive animation is used to shift some of the user’s cognitive load to the human perceptual system and shadows of cones are cast onto the virtual floor.

As our data set has no deep tree shape, recursive techniques are not that appropriate for our problem domain.

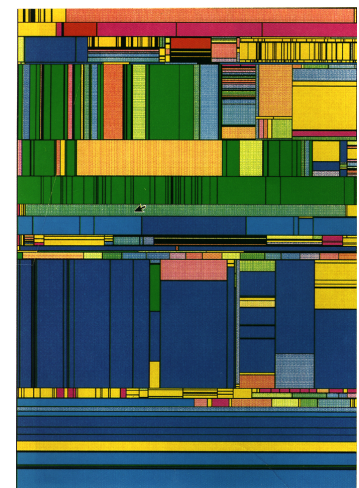


Figure 2.13: Hierarchical Techniques: Treemap [Shneiderman, 1992]

2.2.2.6 RADVIZ

The RADVIZ [Brunsdon et al., 1998] method uses a nonlinear mapping of a set of m -dimensional points onto two dimensional space. This is done with a spring-based algorithm (see Hooke's Law in Section 2.1.10). Each dimension is represented by a point in space that is connected to each data point with an invisible spring. The strength of the spring is depending on the data value in this dimension. The result is a plot of data points where their locations depicts the similarity in the overall measurement across the cases.

The RADVIZ technique seems appropriate for visualizing a highly structured data set as ours. The visualization has a similar result like visualizations of Dimension Reduction techniques but the advantage, that the user can conclude from the positions of the data values on the screen on the data values themselves. One drawback of the technique is that it is possible to get similar looking projections from different basic data properties. In addition to this, by placing the anchors evenly around a circle, RADVIZ assumes that the features are not correlated. But especially in our case, the user may want to group anchors because of their correlation. Furthermore, the RADVIZ paper does not describe how to integrate time-based data and regarding interactions only linking and brushing interaction techniques are discussed shortly.

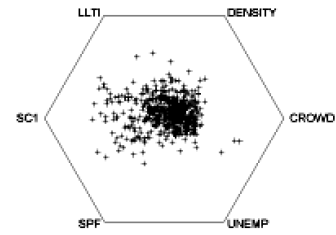


Figure 2.14: RADVIZ [Brunsdon et al., 1998]

2.2.2.7 FreeViz

FreeViz [Demsar et al., 2005] extends RADVIZ by overcoming the limitation of restricting the placement of anchors evenly spaced on a circle (similar to the approach of Gee et al. [Gee et al., 2005]). Furthermore, it integrates an optimization algorithm to automatically cluster the class-labeled data points by arranging the anchors freely. After this optimization, the user can analyze the position of the anchors to gain new insights in the data.

Despite all improvements compared to RADVIZ, FreeViz has still no dedicated support for the time dimension. Furthermore, the drawback of similar looking projections is also not solved in FreeViz.

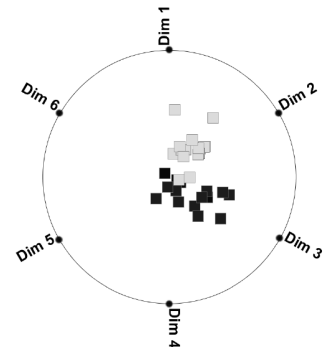


Figure 2.15: FreeViz [Gee et al., 2005]

2.2.2.8 Vibe System

The Vibe System [Olsen et al., 1993] positions document icons according to the occurrence of keywords with a spring-based system. Therefore, it is similar to RADVIZ and shares the same benefits and limitations.



Figure 2.16: Vibe System [Olsen et al., 1993]

2.2.2.9 Dust & Magnet

Dust & Magnet¹ [Yi et al., 2005] is similar to RADVIZ and is especially designed for use by people who struggle with interpreting the complex data of everyday problems. [Yi et al., 2005]

The technique uses a magnet metaphor where the individual data cases are represented as particles of iron dust and the dimensions are represented by magnets. When a magnet is dragged, dust particles are attracted to each magnet on the screen based on the value of the attribute corresponding to the magnets. In addition to this, sliders that filter the data according to the selected value are integrated in their prototype system.

For our problem domain, again, the dedicated visualization of the time dimension is missing. As animation is already used for conveying the data values (during dragging) it can not be used for visualizing changes over time. Furthermore, the interactive dragging principle has the disadvantage, that the iron dust has to be reset from time to time (because the dust is finding its final position at the corresponding magnets).

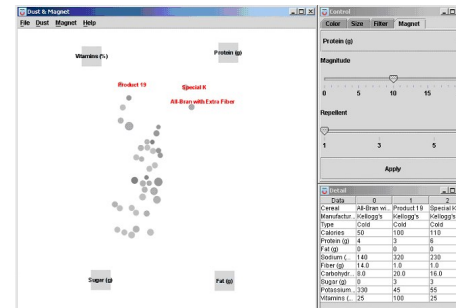


Figure 2.17: Dust & Magnet [Yi et al., 2005]

2.2.2.10 Star Graph

The Star Graph [Harris, 1999], also known as Radar Graph, Spider Graph, or Star Plot [Chambers et al., 1983], is a circular graph with three or more radial axes. The data values are plotted on the axes and connected with lines similar to the Parallel Coordinate Plot. Because of the radial axes, star shaped figures are created using this layout, that can be easily compared because of their shape and size. Therefore, it is primarily used as a comparative tool.

A lot of extensions of the Star Graph have been developed. For example the 2D Zoom Star [Noirhomme-Fraiture, 2002] integrates colored dots to deliver Histogram-like information. The Simple Star technique is an animated extension of the 2D Zoom Star and the 3D Zoom Star and Temporal Star techniques use the 3rd dimension to visualize further attributes (in case of the Temporal Star - time).

The Star Icon or Starglyph [Ward, 1994] is a miniature of the Star Graph typically without titles, labels, tick marks, and so on. The purpose of such icons is the easy comparability of many different icons (for example to provide an overview over the data).

A number of other Circular Graphs are presented in the book Information Graphics [Harris, 1999] that are similar to the Star Graph concept but use the area under the curve (Circular Area Graph), bars (Circular Bar Graph, Circular Column Graph), vectors (Vector Graph), or sectors (Sector Graph) to visualize the data.

The Star Graph and its extensions perform well in visualizing highly structured data. The disadvantages of the Star Graph are similar to Parallel Coordinates. Data lines with equal values cover each other and are, therefore, visualized only as one line. Furthermore, incomplete data is difficult to handle (adding an extra data point or not drawing this part of the polygon are two solutions). The biggest problem is the limited parameter space because of the limited number of axes. In case of using small multiples (Star Icon, Star Glyph, Stardimates) the number of polygons is limited by the screenspace. The important time character for our problem domain is not a dedicated feature.

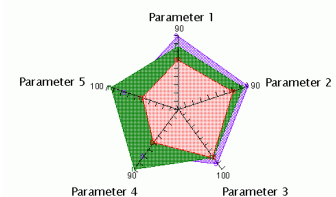


Figure 2.18: Spider Plot

¹<http://www.cc.gatech.edu/gvu/ii/dnm/> (retrieved at 2006/07/01)

2.2.2.11 Interactive Stardinates

The interactive Stardinates [Lanzenberger et al., 2003, Lanzenberger et al., 2005] by Monika Lanzenberger extended the Star Icons with interactivity and other useful features. In contrast to Star Plots this technique can display many records within one instance. The major advantage of the Stardinates are the interactive features such as Linking+Brushing, adding and removing axes, changing the orientation of axes, defining scales, constraining axes (minimum and maximum value), sliders for filtering, zooming, and the ability to change visual properties like color and line thickness which help to explore the visualized data set.

A comparative study of Stardinates and Parallel Coordinates was conducted by Lanzenberger et al. [Lanzenberger et al., 2005] and came to the following conclusions: The Parallel Coordinates performed better to convey information at a first glance. The authors suggested that this could be because of the high degree of familiarity of the Parallel Coordinates and that glyph-based visualizations are often confusing at the first glance. In contrast to this, the Stardinates were significantly better for finding crucial information. Furthermore, the Stardinates seemed to be more motivating for finding hypothesis.

The Stardinates were already applied to our problem space in Lanzenberger’s Ph.D. thesis [Lanzenberger, 2003] and the comparative study indicated that they are a more appropriate method for interpreting such highly structured data in detail.

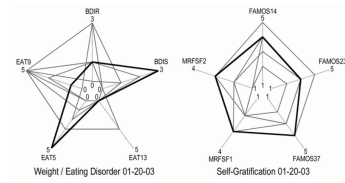


Figure 2.19: Interactive Stardinates [Lanzenberger et al., 2003]

2.2.2.12 Star Coordinates

Similar to a Star Graph, the Star Coordinates [Kandogan, 2000] arrange coordinates on a circle sharing the same origin at the center. In addition to representing each data point on each axis, the Star Coordinates also compute a single point on the screen similar to the RADVIZ visualization. The position of the single point is not computed by using a spring-based model, but by calculating vectors to the final position. Therefore, by using only two axes, the Star Coordinates visualization results in a simple Scatter Plot. Transforming a 3D Scatter Plot on the 2D Screen is the principle how Star Coordinates work (extended to dimensions higher than three). Multiple interaction possibilities such as moving and scaling the axis, highlighting data points (by filtering and selecting them), and displaying traces of the influence of selected axes or moved axis are implemented in the StarDeck system from the author.

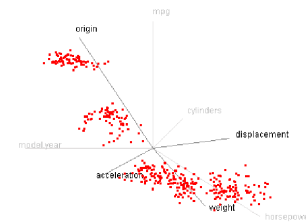


Figure 2.20: Star Coordinates [Kandogan, 2000]

For our problem domain the time-related data aspects are not supported natively. Furthermore, using more than three axes makes the interpretation a complex task for non-technical experts.

2.2.2.13 TimeWheel / Zeitradd

The TimeWheel / Zeitradd [Tominski et al., 2003a, Tominski et al., 2003b] was designed for visualizing temporal data and is conceptually similar to a multi-axes technique for Parallel Coordinates. The axes for the data dimensions are arranged around the time axis in the center. Like the Parallel Coordinates, lines are drawn connecting each axis with the centered time axis. Therefore, the axes oriented parallel to the time axis can be explored best. In order to support this task, the lines of these axes are drawn in full saturation whereas the neighboring axes are color faded. That makes it a Focus+Context technique (see Section 2.1.3.3). Various interaction possibilities are provided to scroll, add, remove, etc. the axes interactively.

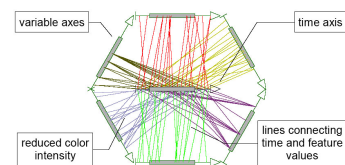


Figure 2.21: TimeWheel / Zeitradd [Tominski et al., 2003a]

The drawbacks of this technique are that only two dimensions (upper and lower to the time axes) can be explored in detail and too much information leads to clutter.

2.2.2.14 MultiComb

The authors of the TimeWheel/Zeitrad also developed the MultiComb technique [Tominski et al., 2003a] using timeseries plots that are arranged circular around the center where additional information or aggregated views are presented. The value axes can either be arranged outward from the center or on the polygon edges. In case the time axes are drawn on outward spokes, the individual time-series plots might be duplicated and put side by side with their neighbors to allow for direct comparison.

One drawback of the MultiComb technique is, that only two adjacent axes can be compared easily. Furthermore, the feature space is limited because of the radial arrangement of the axes.

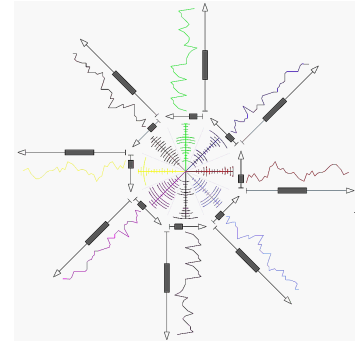


Figure 2.22: MultiComb [Tominski et al., 2003a]

2.2.2.15 TimeHistogram

The TimeHistogram [Kosara et al., 2004] has been developed in order to give an overview over complex time-varying data in the application context of computational fluid dynamics (CFD). A design goal of these techniques was to provide temporal information in static images while maintaining the easy readability of standard Histograms. In order to achieve this, the time axis has been integrated into a standard 2D Histogram in three different ways. In “standard mode”, the user can select and move the currently displayed time step and switch-on the sum of all histograms as contextual information. More information is provided in “point mode”, where the immediate temporal neighborhood is shown by using small colored disks (with decreasing size over time). In “line mode”, line graphs are drawn inside a box in each bin. The line graph represents the temporal development within each bin. A gray box is drawn to emphasize minimum and maximum values for each bar.



Figure 2.23: TimeHistogram [Kosara et al., 2004]

As the TimeHistogram is intended to give an overview, it is not suited for a more throughout analysis of our medical data. Furthermore, the high dimensionality of our data set would need many different plots.

2.2.2.16 TimeHistogram 3D

The TimeHistogram 3D [Kosara et al., 2004] was developed by the same authors to integrate the time-dimension into a standard Histogram. This is done by using the third dimension to represent time. Hence, for each time step a Histogram is drawn as a row of cuboids. Several interactive features such as brushing, scaling, and a 2D context display is shown in the background of the Histogram are part of this technique.

Like the TimeHistogram, the TimeHistogram 3D is intended to give an overview, it is not suited for a throughout analysis of the data. Furthermore, as it is a 3D technique additional navigational effort is required to change the view-point and avoid visual clutter.

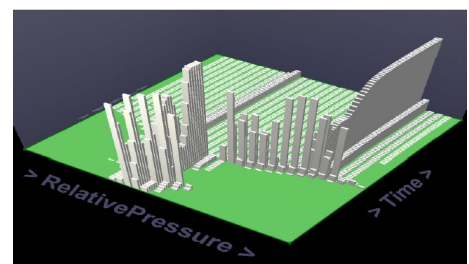


Figure 2.24: TimeHistogram 3D [Kosara et al., 2004]

2.2.2.17 Time-Tunnel

The Time-Tunnel [Akaishi and Okada, 2004] is a data analysis technique for visualizing a number of time-series plots in 3D. The individual plots are put onto semi-transparent planes (called data-wings) that are positioned around a central time-bar. The data-wings can be overlapped and due to their transparency easily compared. In addition to this, single time slices can be viewed as a Star Graph.

Because of the 3D presentation, the Time-Tunnel visualization can suffer from overlapping and clutter and forces the user to navigate in the 3D space to find an appropriate viewing angle. The high amount of dimensions in our psychotherapeutic data could also be problematic.

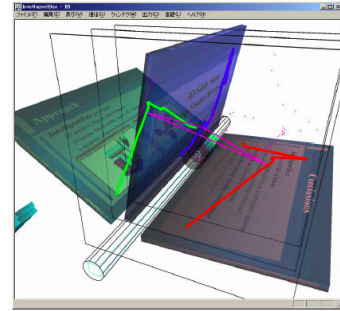


Fig. 6. Example as multimedia presentation tool.

Figure 2.25: Time-Tunnel [Akaishi and Okada, 2004]

2.2.2.18 Worm Plots

In the 3D technique Worm Plots [Matthews and Roze, 1997] groups of multivariate time series are visualized. Each time step is represented by a 2D plane in 3D space with every data group plotted as a circle whose center is the data's centroid (e.g. 2D mean) and whose radius is the average distance of the data points from the centroid. The circles of a group at neighboring time slices are then connected by cones to emphasize their change over time. This method is especially suited for exploring groups of data to get an overview of the depicted phenomena.

The negative impact of this visualization is display clutter, occlusion, and navigational effort in the 3D space.

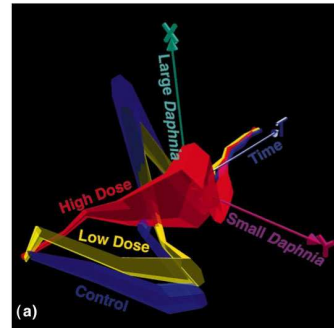


Figure 2.26: Worm Plots [Matthews and Roze, 1997]

2.2.2.19 Concentric Circles Technique (CCT)

The interactive representation technique Concentric Circles Technique (CCT) [Daassi et al., 2000] is designed to visualize one or two quantitative histories. Its interface consists out of a set of concentric circles, each one denoting the evolution of the visualized history. The data values are encoded by the height of a bar and its color intensity. A second dimension can be displayed on the opposite side of the corresponding circle for comparison purpose. Therefore, the technique is somewhat similar to the different Star Graph techniques. Furthermore, a timeline is provided to navigate through time and define an interval with sliders (for the Star Graph like display). A working area to compare interesting stills is provided on the right side of the circles.

As only two data dimensions can be visualized with one plot, the CCT is not perfectly suited for our problem domain. Furthermore, we only need to visualize a subset of the timesteps that can be visualized with this technique.

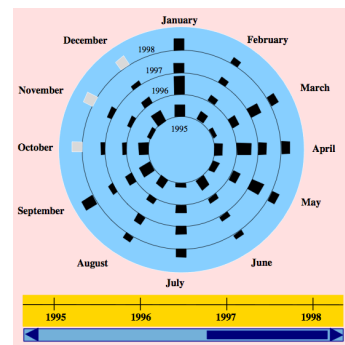


Figure 2.27: Concentric Circles Technique [Daassi et al., 2000]

2.2.2.20 Star Representation Technique (SRT)

The Star Representation Technique (SRT) [Daassi et al., 2004] uses two visualization techniques to display the temporal and structural dimension of the data set. The structural dimension is visualized using a Star Graph based visualization whereas the values on the adjacent axes are not connected by straight lines, but by using a filled “v-shape” which leads to a star-like appearance. To navigate through time the SRT has included the Focus+Context technique (see Section 2.1.3.3) Perspective Wall [Mackinlay et al., 1991]. Similar to the CCT, the SRT provides a reference (working) area that can be used for comparisons and performing mathematical operations (sum, average, subtract, multiply, divide) of Star Graphs.

Because of the filled “v-shapes” the exact values are hard to read and visualizing many different dimensions may lead to clutter.

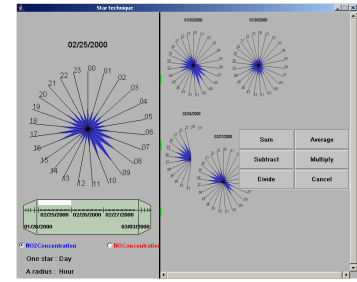


Figure 2.28: Star Representation Technique [Daassi et al., 2004]

2.2.2.21 Flocking Boids

The Flocking Boids technique [Moere, 2004] is a data-triggered, dynamic representation of multivariate time-related data. It represents data elements as moving particles in 3D with colored traces showing their evolution. The line color denotes positive or negative value changes. The movement of these 3D particles called boids (bird-objects) is controlled by a rule-based system that models their swarming behavior.

“Notably, this representation technique focuses on the representation of dynamic data alteration characteristics, or how reoccurring data objects change over time, instead of depicting the exact data values themselves.” [Moere, 2004]

As the Flocking Boids focus on the overview over the changes in the data set, the usage for our problem domain would be restricted to an overview tool or additional view.

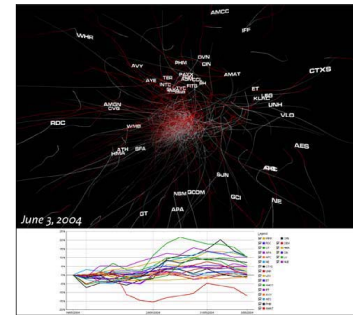


Figure 2.29: Flocking Boids [Moere, 2004]

2.2.2.22 Interactive Parallel Bar Charts (IPBC)

The Interactive Parallel Bar Charts (IPBC) technique [Chittaro et al., 2002] adopts bar charts as the basic visualization technique and augments them with several interactive features in 3D space. Each time-series is represented in a bar chart format where the x-axis is associated with time and the y-axis with the value (height of a bar). Now, the obtained bar charts are arranged side by side extending into the third dimension. To deal with the difficult 3D navigation, the IPBC restricts the navigation to pre-defined trajectories which can be useful for examining the visualization. Furthermore, the IPBC provides dynamic queries (using color coding) and a tide mode that helps comparing data with (time-varying linear thresholds). This means, the user can, for example, “flood” the visualization with an over-time increasing tide and watch which values stand out. To deal with occlusion, the IPBC can transform the parallel bar chart into a 2D matrix format that encodes the values only with color. Using multiple linked windows, more than one dimension can be visualized simultaneously.

Opposing to the major advantage that (because of using well known bar charts) the visualization is quickly learned and remembered by clinicians, the disadvantage of screenspace limitations for analyzing more than variables simultaneously would be a big drawback for our problem space.

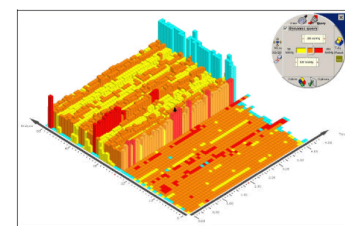


Fig. 7. Removing occlusions.

Figure 2.30: Interactive Parallel Bar Charts [Chittaro et al., 2002]

2.2.2.23 Table Lens

The Table Lens [Rao and Card, 1994] is a Focus+Context (fisheye) technique that works effectively on tabular information. It fuses symbolic and graphical representations into a single coherent view that can be fluidly adjusted by the user. The values can be either represented by the printed number or text (when in focus) or as a bar or pixel (when in context).

Because of the time dimension our target data set would be difficult to visualize efficiently using a Table Lens technique. Still an overview over the data could be very helpful using this approach.

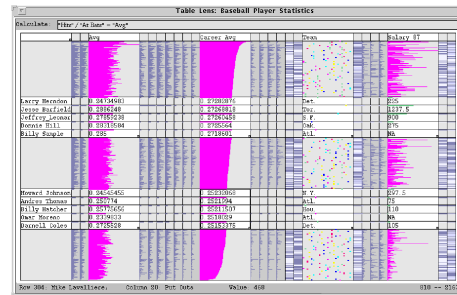


Figure 2.31: Table Lens [Rao and Card, 1994]

2.2.2.24 LifeLines

LifeLines [Plaisant et al., 1996, Plaisant and Rose, 1996, Plaisant et al., 1998] is the probably best known technique of “Timeline” visualizations. Horizontal bars are used to show the temporal location and duration of information elements. LifeLines were applied for representing personal histories and patient records. For organizing the elements, the information is grouped in so-called facets, which can be expanded and collapsed. When collapsed, only a very small and geometrically as well as semantically downscaled version without textual labels is shown. Furthermore, information can be encoded via the height as well as the color of the individual bars. Additional information might be provided on demand in a linked view, as for example x-ray images.

As we have only a limited number of timesteps (that are synchronized), a Timeline-based visualization (like the LifeLines) would not make sense as the main information “When was what?” (exact temporal location) is not very important for our users.

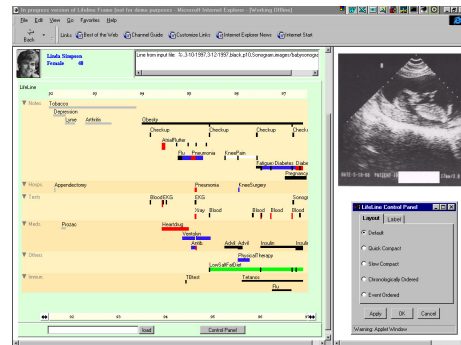


Figure 2.32: LifeLines [Plaisant et al., 1998]

2.2.3 GLYPHS

Glyphs or Icon-based visualizations are graphical objects, which represent data by using visual parameters such as shape, size, position, or color. They are especially well suited for displaying complex, multivariate data sets. Low-level tasks that are strongly supported by all techniques are the rate of change and sequence. A good overview on multivariate glyphs is presented by Matthew Ward [Ward, 2002].

One thing that has to be considered, is that the design of a suitable icon is a sensitive process. Metaphor-based icons (like the Chernoff Faces) can be problematic because they may convey some sort of information that does not come from the data set.

2.2.3.1 Chernoff Faces

Herman Chernoff first proposed the idea of using a cartoon face to represent multidimensional information. He coded all parameters in a Glyph which is called Chernoff Faces [Chernoff, 1973]. Up to 18 parameters can be mapped on the mouth, nose, eyes, eyebrows, and so on. By using Small Multiples (see Section 2.1.8) two further dimensions can be visualized (e.g., time and persons). The Chernoff Faces are especially suited to show trends in the data. Furthermore, as humans do very well in perceiving facial expressions (we may even have special neural hardware to deal with faces, see Colin Ware [Ware, 2000, page 253]), the glyphs are recognized and memorized very well.

A major drawback of Chernoff faces is that the subjective assignment of facial expressions to variables affects on the shape of the face. Chernoff and Rizvi [Chernoff and Rizvi, 1975] described that the permutations of the assignment of features caused an error rate of as high as 25 for the task of classifying faces into groups. It means that classifying two faces as "fairly similar" is greatly influenced by the assignment of variables to specific features. Furthermore, exact data values may be difficult to perceive because of only subtle changes in the visualization. Some dimensions are even perceived only as a unity. For instance, if seven data dimensions are mapped to properties of eyes and eyebrows: e.g., eye size, eye form, position of eyes, size of pupil, position of pupil, position of eyebrows, shape of eyebrows, we usually perceive the eyes as a whole and are unable to distinguish components adequately.

For our data set there is no direct, intuitive way of mapping all data values to facial properties. Therefore, the interpretation could be misleading (e.g., interpreting a happy face as a good therapy outcome).

2.2.3.2 Stick Figures

An iconic technique for visualizing large amounts of data is the Stick Figure technique [Pickett, 1970, Pickett and Grinstein, 1988]. As indicated by the name, the icon looks like a stick figure. It is a 2D, four arms figure where the data attributes are mapped to the angle and length of the limbs. Overall texture patterns (produced from large amounts of Stick Figures) show data characteristics.

As with Parallel Coordinates, Keim and Kriegel [Keim and Kriegel, 1996] showed that using 15.000 data items lead to overlapping icons and that may prevent the visualizations from being useful. They also mentioned that it is important which dimensions are assigned to the axes.

“If the right dimensions (e.g., the cluster dimensions) are chosen to be assigned to the axes, clusters may become visible due to the high density of Stick Figures in certain regions.” [Keim and Kriegel, 1996]

In their case the density of stick figures at certain locations revealed clusters (and that could also be visualized with a Scatter Plot diagram). To use the multidimensionality Keim claims that the overlapping has to be avoided.

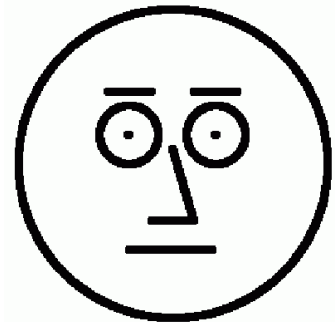


Figure 2.33: Chernoff Faces [Lanzenberger, 2003]

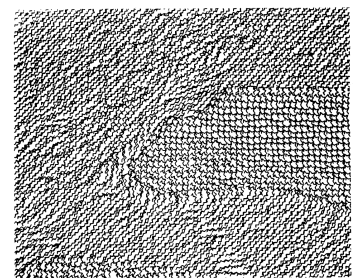


Figure 2.34: Stick Figures [Pickett, 1970]

In addition to these overlapping problems the Stick Figures do not support the time-dimension and sophisticated interactions. Therefore, this technique can only be used to provide some sort of overview of the data.

2.2.3.3 InfoBUG

The InfoBUG [Chuah and Eick, 1997a, Chuah and Eick, 1997b, Chuah and Eick, 1998] is a glyph for visualizing software projects or software families. The layout consists of the following graphical elements: body (representing component size), wings (code lines and errors), head (type of code), and tail (added and deleted lines of code). The various parameters make the representation a very information rich graphic that combines a multitude of different heterogeneous parameters and allows for easily deriving implications by relating parameters visually. Small multiples are used to visualize the development over time. Alternatively, the representation can be animated to follow the course of time.

The InfoBUG is a domain-specific glyph developed for software project management. It could be adapted for our data set by mapping the medical data to the different graphical layout components. Still not all dimensions could be mapped on the glyph. That means an interactive mapping possibility has to be provided, which leads to a high learnability. Furthermore, there are sometimes only small graphical differences between successive values which make it hard to read the exact values.

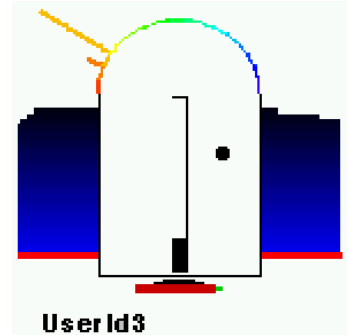


Figure 2.35: Single InfoBUG [Chuah and Eick, 1998]

2.2.3.4 Timewheel Glyph

The Timewheel Glyph [Chuah and Eick, 1997a, Chuah and Eick, 1997b, Chuah and Eick, 1998] contains less attributes than the InfoBUG visualization but pays more attention to the development of the data values over time. Small iconified versions of the TimeWheel/Zeitrad technique (see Section 2.2.2.13) are used to display global trends and patterns. In particular the Timewheel glyph makes it easy to identify two basic trends – an increasing trend and a decreasing trend of the individual time-series.

In respect to our data set, the Timewheel Glyph could be used for an overview to show the overall trend of the data set. Detailed analysis is restricted because exact values are hard to perceive and the comparability of different axes is difficult. As even less attributes are displayed than in the InfoBUG glyph, the same problems with our high dimensionality arise here.

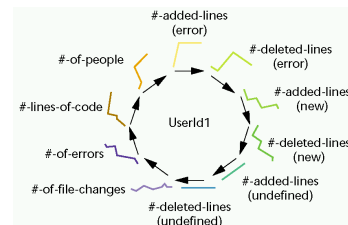


Figure 2.36: Timewheel Glyph [Chuah and Eick, 1998]

2.2.3.5 3D Wheel

The 3D Wheel [Chuah and Eick, 1997a, Chuah and Eick, 1997b, Chuah and Eick, 1998] extends the idea of the Timewheel to the third dimension. It uses the additional height dimension to represent time. It is particularly suited to visualize overall trends through time.

Because of the 3D representation the drawbacks of perspective and occlusion can not be denied.

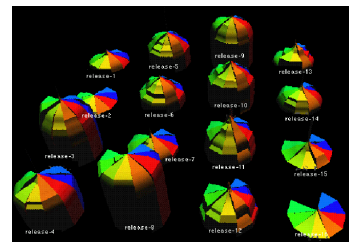


Figure 2.37: 3D Wheel [Horn et al., 1998]

2.2.3.6 VIE-VISU

VIE-VISU [Horn et al., 1998] is a data visualization method which uses multiples to present the change in a patient's status over time in graphic form. The glyph display helps to combine different measurements, maintain their relationships, show their development over time, and make specific situations easy to spot. 15 different patient parameters are combined to form this glyph. One display gives a metaphor graphics summary on a time scale of six hours to six days depending on the user's choice.

As with nearly all glyph-based representations, the complexity and learnability are a big problem because of the abstract representation. Our highly structured data set would be difficult to visualize because of the limited space for each of the different parameters. To compare different persons in detail over time is complex and can only be realized by placing the glyphs closely next to each other.

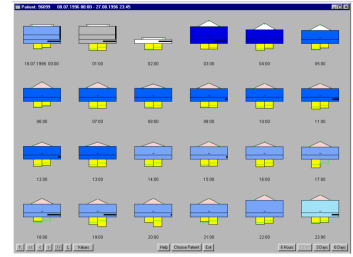


Figure 2.38: VIE-VISU [Horn et al., 1998]

by placing the glyphs closely

2.2.4 Pixel-based / Pixel-oriented Visualizations

In contrast to Glyphs, Pixel-based visualizations have no visual data-relatedness.

“The basic idea of Pixel-oriented techniques is to represent as many data items as possible at the same time. This is done by mapping each data value to a colored pixel and present the data values belonging to one attribute in separate window. This way the number of data values is only limited by the numbers of pixels of the screen.” [Keim and Kriegel, 1996]

Because each data value only needs one pixel on the screen space, a huge amount of data values can be visualized at once on the screen. The main question is how to arrange them. Keim and Kriegel [Keim and Kriegel, 1996] proposed to use different arrangements for different purposes, for example, according to the time dimension in the data. Instead of directly mapping the data values to color, query-dependent visualization techniques can be used. All Pixel-oriented techniques can partition the screen into multiple areas. This way n-dimensions can be mapped to n areas.

Related to Pixel-oriented techniques, where each pixel represents a separate data item, are matrix-based visualization techniques that use larger rectangular areas to display a single item (in each cell of the matrix or table respectively). The principle of mapping data values to color remains the same, therefore we will also present these techniques in this section.

2.2.4.1 Recursive Pattern

The best known pixel-oriented method is probably the Recursive Pattern Technique [Keim et al., 1995]. Each data value is mapped on the color of one pixel. The arrangement of the data items centers the most relevant data items in the middle of the window with a fully recursive algorithm (left to right, right to left, left to right), and less relevant data items are arranged in a spiral shape to the outside of the window. Multiple areas are used to visualize multiple dimensions. Furthermore, the technique is able to visualize query-dependent data in the context of a specific user query and gives the users feedback on their queries and direct their search. This method was also implemented in the VisDB system [Keim and Kriegel, 1995]. Keim and Kriegel mention in their article the possibility to arrange the pixels according to an inherent time aspect. The Matrix Visualization [Chittaro et al., 2002] that is part of the Interactive Parallel Bar Charts (IPBC) tool, the Agenda technique [Daassi et al., 2000] and the Tile Maps [Mintz et al., 1997] are examples, where the ordering is time related.

Therefore, this kind of visualization would be suited for our problem domain. Analysis would be done by recognizing patterns and clusters. Still, the high attribute space of our medical data would require many different windows (for each dimension one) that may lead to difficult comparability.

2.2.4.2 Circle Segments

Like Recursive Patterns, the Circle Segments technique [Ankerst et al., 1996] applies the basic principle to map data items to colored pixels. But in contrast, multiple data attributes can be shown in a single view. The whole data set is displayed in a circle that is divided into segments for each attribute. Within each segment, pixels are arranged using back and forward patterns starting from the circle’s center.

The drawback of the Recursive Pattern technique is solved here because of the segmentation but because of the circular arrangement, the comparability is limited.

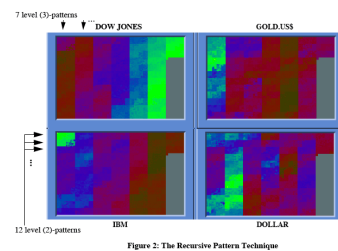


Figure 2.39: Recursive Pattern [Keim et al., 1995]

2.2.4.3 Multi-scale Temporal Behavior

The Multi-Scale Temporal Behavior technique [Shimabukuro et al., 2004] can be seen as a combination of pixel-oriented and matrix-based visualizations. It comprises different levels of granularity and aggregation to explore patterns at different temporal levels. The visualization is divided into three regions for the three scale levels daily, monthly, and yearly. The daily part uses single colored pixels for each day, whereas the middle and lower region aggregate the values to a single monthly or yearly value that is visualized in a matrix style using color. The problem of missing data is tackled visually and in terms of pre-processing.

For our medical problem, only two levels of granularity would make sense because of the few samples during the therapy.

2.2.4.4 Shape Coding

Shape Coding [Beddow, 1990] is a glyph approach for depicting higher-dimensional data set. Data is visualized using small arrays of fields (5x5 fields). Each field represents one value. The fields are arranged from left to right in a rectangular structure. The arrays can then be arranged, for example, according to a time attribute. Furthermore, the color mapping can be based on the mean and standard deviation of columns.

Because of the rectangular arrangement, the comparability of single values can be difficult. As the most Pixel- or Matrix-based visualizations the main usage is to present an overview on the data to find interesting patterns.

2.2.4.5 Color Icon

The Color Icon [Levkovitz, 1991] merges separable features by using color, shape, and texture perception to code multiple parameters into a single integrated image. The Color Icon is a rectangle with six linear features (lines from the edges through the center of the rectangle). These lines are colored according to the actual value of the feature. The space between the lines is interpolated. In another variant the fields between the lines are colored according to the mapped dimension. This yields a “Pie Chart” type of an icon, with eight parameter fields (“slices”). Furthermore, Levkovitz proposes to use three data parameters to construct a color value (red, green and blue component of the RGB model). By placing these Color Icons on a x,y-plane according to any two data-dimensions, an image is composed that can be analyzed for interesting patterns.

The drawback of the Color Icon approach is the learnability and comparability (especially using the interpolated data values in between the lines).

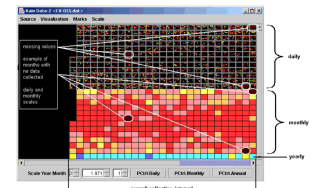


Figure 6. Multi-scale temporal behavior representation for handling missing and extraneous values

Figure 2.41: Multi Scale Temporal Behavior [Shimabukuro et al., 2004]

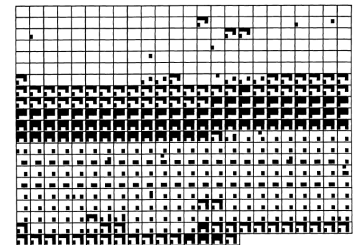


Figure 2.42: Shape Coding [Beddow, 1990]

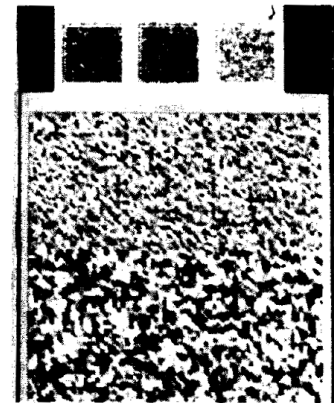


Figure 2.43: Color Icon [Levkovitz, 1991]

2.2.5 Coordinated Multiple Views (CMVs)

Each type of visualization emphasizes certain characteristics of the depicted data. Multiple View visualizations provide different visualizations to combine their characteristics and can therefore lead to a better insight into the data. Furthermore, they can support a new perception of known facts because of the transformation in unknown structures (through the observation from different perspectives).

“Visualization of data in a variety of ways can form part of a brainstorming which reveals anomalies or unexpected trends and patterns.” [Brodie et al., 1992, 16]

Baldonado et al. [Baldonado et al., 2000] give a good definition for multiple view-systems:

“A multiple view-system uses two or more distinct views to support the investigation of a single conceptual entity.” [Baldonado et al., 2000]

Coordinated Multiple Views (CMVs) can also be defined as multiple linked views with sophisticated coordination and interaction mechanisms of the same data. They are linked with interaction techniques such as Linking and Brushing (see Section 2.1.3.2) or Details on Demand (see Section 2.1.3.1) and can show the data with different levels of detail and different visualization techniques. The drawback of CMVs is the additional complexity for both designers and users. Fundamental research in formalizing the coordinated multiple views approach together with its incorporation into the visualization reference model has been carried out by J. C. Robert [Roberts, 1998, Roberts, 2000].

We will now present specific CMV applications that are related to our problem field.

2.2.5.1 Time-Series Explorer

The Time-Series Explorer [Craig et al., 2005] uses Coordinated Parallel Views for the exploratory analysis of Microarray time-course data. It uses three coordinated parallel views with two Time-Series Graphs and an animated Scatter Plot. What is so special about the Time-Series Explorer is the ability to highlight genes with a freehand selection tool. This can be done by clicking and dragging a freeform shape around their representations. Furthermore, an interval can be selected using dynamic sliders. Their evaluation showed that less dominant patterns in the data, such as those that occur over limited intervals of the time-course, cannot be revealed using existing established techniques (like the Heatmap [Eisen et al., 1998]) but can be analyzed with the Time-series Explorer.

Our medical data set would not perfectly fit into the Time-Series Explorer (e.g., different dimensions, additional dimension of different users, etc.) but the advanced selection methods combined with the coordinated views could be of value.

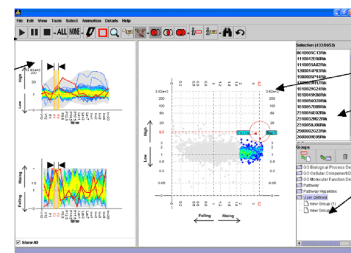


Figure 2.44: Time-Series Explorer [Craig et al., 2005]

2.2.5.2 SpringView

The SpringView system [Bertini et al., 2005] incorporates the visualization RADVIZ and Parallel Coordinates in a coordinated parallel manner. The authors argue that these two techniques are complementing one another.

“In particular, while the RADVIZ visualization lends itself to the detection of aggregates, Parallel Coordinates permit to explore finer details inside subsets, thus offering different data views.” [Bertini et al., 2005]

Furthermore, they integrated a simple clustering algorithm that colors the items in the visualization according to their position and therefore helps visually connecting them to the same data in the Parallel Coordinates.

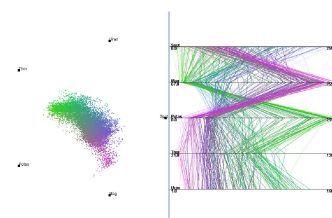


Figure 2.45: SpringView [Bertini et al., 2005]

The side by side view of both visualizations makes it a bit difficult to relate the dimensions of the visualized data. For our data, the time dimension would be hard to visualize without any change (would lead to much more data). Furthermore, more interaction techniques (as proposed in FreeViz [Demsar et al., 2005] for example) would enhance this system further.

2.2.5.3 Visual Data Mining Display (VDMD)

The Visual Data Mining Display (VDMD) [Ericson et al., 2005] visualizes a combination of statistical values with Parallel Coordinates. Furthermore, classifications of the currently selected data, as clusters and Histograms, can be turned on and off as layers on top of the Parallel Coordinates. Statistical values of selected dimensions are displayed in several two-dimensional coordinate systems, that are linked with main view. When a change occurs, the plots are animated and leave a visible vector as a track of the last movement.

The VDMD tool does not inherently support time-based data. In addition to this, as we want to compare visualizations against statistical methods, the use of statistical plots (such as the Histogram or the Plots) in the visualization would unbalance study settings, making the results less expressive.

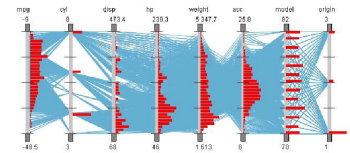


Figure 2.46: Visual Data Mining Display [Ericson et al., 2005]

2.2.5.4 VisDB

The VisDB system [Keim and Kriegel, 1995] combines traditional database querying and information retrieval with various data visualization techniques. It offers Pixel-oriented (Spiral Technique), a Geometric Projection (Parallel Coordinated), an Icon-based (Stick Figures), and Hierarchical techniques (Axes Technique) to explore and analyze large databases (supporting query based results). The two aspects distinguishing their visualization techniques from other approaches are the high number of data items that can be visualized (due to the Pixel-based approach) and the query-dependency of the displayed data. Furthermore, direct interaction (highlighting, change of the query range) on the visualizations is supported by the VisDB system.

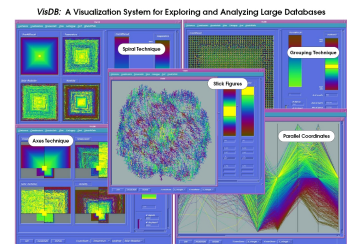


Figure 2.47: VisDB [Keim and Kriegel, 1995]

Especially the lack of inherent time support disqualifies the VisDB system for our data set and problem domain.

2.2.5.5 LinkVis

LinkVis [Herzog, 2004, Lanzenberger et al., 2005] is a multiple view framework and tool with a powerful data pre-processing part. The data preparation provides filtering, aggregation of values with functions and the conflation of records with different dates of measurement. The implemented tool visualizes the data with Chernoff Faces, Scatter Plots, and Parallel Coordinates. A scenario-based evaluation demonstrated the power of the LinkVis tool on real world problems and showed the data complexity.

Our domain experts adjudged, that LinkVis provides too less interaction possibilities for our complex data set and problem domain. Furthermore, the chosen visualizations (especially the Chernoff Faces) were not perfectly suited to analyze the data thoroughly. Still, the coordinated multiple views technique was a good approach to deal with the large feature space.

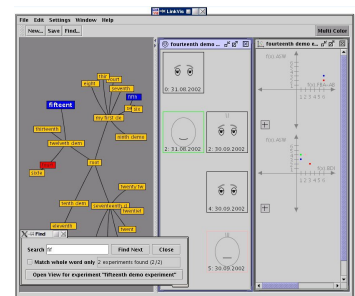


Figure 2.48: LinkVis [Herzog, 2004]

2.3 Discussion

Our problem is to visualize highly structured, temporal data with an interactive Information Visualization. Furthermore, we are looking for a technique that can be easily understood and used by non computer experts. None of the above described techniques met all of our requirements. Most often the dedicated support for the time dimension was missing. That time is an outstanding data dimension is also reflected by Ben Shneiderman's Task by Data Type Taxonomy [Shneiderman, 1996] who identifies temporal data as one of seven basic data types most relevant for Information Visualization. However, most visualization techniques do not reflect this fact and treat time just as any other data dimension. Hence, they do not take into consideration that time is perceived differently and also influences other attributes and physical dimensions quite uniquely. But using special methods for treating time is necessary in order to accomplish the important tasks of revealing trends, as well as identifying patterns and relationships in the data [Aigner, 2006].

In our presented overview of related work, we examined different techniques that promised to (at least partly) fulfill our requirements for our problem domain.

The visualization methods in Statistics and Mathematics are well known techniques but lack interactivity and time-support. The Dimension Reduction techniques have the big drawback of delivering a visual classification that can not easily be traced back to the individual data dimensions. This can lead to misinterpretations on the data set.

This is solved by most of the Geometric Techniques. They are well suited to visualize highly structured data but most techniques lack the support for the time dimension. Spiral-based techniques [Aigner, 2006] were not considered in our overview because of our non-cyclical data under investigation and the small number of timesteps in our data set that make the use of a spiral-based technique not suitable.

Using Glyphs for our problem domain (as LinkVis [Herzog, 2004] does) is problematic because of the high dimensionality of our data set. This would either mean constructing a glyph that can visualize all dimensions leading to an overcrowded Glyph where it is hard to depict subtle changes or require the interactive mapping of features. The interactive mapping can easily result in unrelated mappings which can lead to misinterpretations. For example, mapping medical data that predicts a bad therapy outcome to a smiling Chernoff Face [Chernoff, 1973]. Siirtola [Siirtola, 2005] discusses this data-relatedness in his paper and came to the conclusion that data-related glyphs are more accurate than unrelated glyphs.

In contrast to this, Pixel-based visualizations have no visual data-relatedness. They use colored pixels to represent the data set, which enables them to visualize a huge amount of data as a whole. The ordering is of much importance and decides which parameters can be compared easily.

Coordinated Multiple Views (CMVs) combine the characteristics of the used visualizations and offer different views on the data. The drawback is the higher complexity both for user and designer. A large data set that should be analyzed in detail can only be analyzed using CMVs offering an overview and detail information on the data.

As we can see, no existing technique is perfectly suited to meet all of our requirements. Geometric Techniques seem to visualize our categorical data set best and in conjunction with other techniques (using CMVs) the temporal data should be visualized best. Therefore, we chose a geometric technique (namely a spring-based dimension reduction similar to the RADVIZ method) combined with multiple views in order to meet most of our requirements.

Chapter 3

Medical Problem Domain

The medical data for developing and testing Gravi++ was prospectively collected during a psychotherapeutic clinical study, "Cognitive Behavioral Therapy (CBT) for adolescent girls with Anorexia Nervosa (AN) and their families" [Ohmann et al., 2004, Ohmann et al., 2005a, Ohmann et al., 2005b, Ohmann et al., 2005c]. The project was planned for an in-depth evaluation of a manualized multimodal group CBT program for anorexic girls. 30 patients were enrolled in the study. They were divided into four groups that received 40 therapeutic double sessions of 90 minutes. An extensive psychological test battery was used for assessing initial psychopathology and therapeutic progress during (at 3, 6, 9, and 12 months) and yearly after the therapy for a follow-up period of four years. The aim of the study was to find out at what time changes and clinical improvements occur in the therapeutic groups. Patients with a positive therapy refused to fill out long questionnaires. Therefore, only one follow-up data sets could be collected.

3.1 Definition of Anorexia Nervosa (AN)

Eating disorders in children and adolescents represent potentially life-threatening, debilitating conditions that impede physical, emotional, and behavioral growth and development [Robin et al., 1998]. Eating disorders are characterized by irrational fears of becoming fat, by an extremely restricted eating behavior and additional attempts to loose weight or prevent weight increases. There are at least three subtypes of varying severity: Anorexia Nervosa (AN), Bulimia Nervosa (BN) and a residual category of patients with symptoms of AN or BN but without meeting all necessary criteria.

AN is a common disease especially in adolescent girls with a prevalence of about one percent. It is characterized by low body weight as a result of loosing $> 25\%$ of the body weight by avoiding high caloric food, self induced vomiting, abuse of laxatives, appetite suppressants, or diuretics and compulsive exercising. Anorectics often experience body image distortions, absence of menstrual cycles, or a delay of pubertal development. Clinically, the "restricting" and the "binge eating/purging" subtypes may be differentiated.

Eating disorders evolve from typical patterns and can be more or less summarized in terms of three recurring characteristics: abnormal eating behavior, a negative body image, and either successful or unsuccessful attempts to lose weight. The multicausality model has generally been accepted and still figures as the most important etiological model [Jansen, 2001].

Problems associated with AN are mainly the loss of body mass and the change of body shape, consequences of inappropriate nutritional supply like osteoporosis, skin disorders, hormonal changes such as primary or secondary amenorrhoea, deterioration of physical fitness and performance, CNS changes associated with cognitive and affective disturbances, loss of interests, and social isolation.

An extreme need to control eating is the central feature of the disorder. The origins of cognitive behavioral theory of AN can be traced back to the clinical observations of Hilde Bruch. Bruch's ideas were further extended by Garner & Bemis [Garner and Bemis, 1985] who applied the principles of Beck's cognitive theory and therapy of depression to AN [Beck et al., 1979].

3.2 Cognitive Behavioral Therapy

Cognitive behavioral therapy (CBT) has become one of the most effective forms of short time psychotherapy for a wide range of problems and has been evaluated in various controlled randomized outcome studies [Vallis, 1998, Rachman, 1997]. Controlled treatment outcome studies of childhood disorders are scarce but those that exist indicate that CBT is an effective intervention for children and adolescents [Ronen, 1997, Baving and Schmidt, 201]. CBT is empirically supported for the treatment of

- Depression
- Anxiety Disorders
- Eating Disorders
- Pain
- Obsessive-Compulsive Disorder (OCD)
- Attention Deficit Hyperactivity Disorder
- Posttraumatic Stress Disorders
- Internalizing and Externalizing Disorders in children and adolescents
- Psychosis
- Personality disorders

People develop and maintain cognitive sets or schemata that allow their experiences to make sense [Beck et al., 1979, Lampert and Davidson, 1997, Meichenbaum, 1977]. Schemata are employed by social cognitive, information-processing and inferential processes as heuristics to screen, limit and organize perceptual stimuli. Psychopathology is thought to result from distortions in the formation or use of schemata [Davidson et al., 1998].

3.3 Cognitive-Behavioral Therapy for Eating Disorders

Cognitive therapy is a semi-structured and goal-directed approach to the treatment of eating disorders. The goals of the therapy are psychoeducation, modification, and correction of misperceptions in thinking. The focus is directed on the current aspects of the patients problems - the “here and now” focus. The process of identifying schemata helps to increase self-esteem and internal awareness and assists the patients to see when their self-deficits activate [Bowers, 2001].

CBT for AN has only insufficiently been studied. Its efficiency remains in question. By contrast, manual based CBT is the first-line treatment of choice for BN [Wilson, 1999, Porzelius et al., 1999, Whittal and Zaretsky, 1996, Fairburn, 1997, Robin et al., 1998, Latner and Wilson, 2000]. CBT with its multi-modal approach is one of the most effective therapeutic concepts for adults with eating disorders [deZwaan and Karwautz, 1996, Bowers, 2001]. Inpatient and outpatient models have successfully been described [Hsu, 1990, Wilson and Fairburn, 1993, Jacobi and Thiel, 1996, Benninghoven and Liebeck, 1997, Beisel and Leibl, 2001]. Detailed manuals providing step-by-step procedures for treating AN have not yet been published [Wilson, 1999]. There are several reasons why cognitive behavioral treatment of AN has been relatively neglected [Fairburn, 1997]: AN is more difficult to study than BN because it is less common, its treatment takes much longer, and the patients are much more difficult to engage in treatment.

Usually, cognitive interventions are more effective after a certain increase in weight - the lower the body weight, the greater the initial inertance against therapeutic approaches. CBT for patients with AN results in a more than 50% reduction of core eating-disorder psychopathology, depressive symptoms, and positive changes of illness-based schemata [Bowers, 2001].

3.4 Aims of the Study

Although behavioral therapy has been proven effective in inpatient and outpatient settings, standardized cognitive behavioral programs for adolescents with the diagnosis AN are still lacking. This study evaluates a manualized multimodal group CBT program to find out what therapeutic strategies and modules are effective during the therapeutic process. In addition, predictors of the relatively high drop-out and hospitalization rates and to understand the dynamics of complications occurring during the therapeutic process.

Further aims of this study are

1. To standardize the therapeutic process based on a clinical study.
2. To improve the communication of transparent therapy protocols.
3. To simplify the therapeutic supervision process.
4. To review and evaluate the process of cognitive behavioral therapy and to study the psychotherapeutic process in group therapy in a clinical study.

3.5 Project Description

A total of 30 adolescent girls with AN, aged 13 to 17 years (restrictive or binge/purging type) entered the study during the course of four years. Subjects had to fulfill DSM-IV or ICD-10 criteria for AN and the adapted weight criteria for age and gender related BMI-percentiles (body mass index) [Karwautz et al., 2001] at the initial assessment at the Eating Disorder outpatient clinic. The duration of illness and the number of previous treatment varied considerably. Patients were admitted if they freely consented to actively participate in the study and the follow-up investigations.

3.5.1 Methods

The clinical evaluates a manualized outpatient CBT group program that lasted for one school-year (two semesters, October to June). In case of a serious clinical need ($BMI < 11.5$), patients were admitted to an inpatient Eating Disorders Unit and the behavioral therapy was interrupted during the period of clinical residence. Assessments were conducted by a research clinical psychologist who was independent of the treatment team and interviewed the patient, her family and the therapists. Assessment was carried out before treatment, during the course of the treatment to allow for the evaluation of the process of change and at the end of treatment, and at follow-up of 12 months after the end of treatment. As mentioned in the introduction, practically only one assessment was collected on average after the therapy. At the beginning of each therapeutic session the patients were interviewed about their actual condition.

3.5.2 Evaluation

The evaluation of the clinical study was conducted using questionnaires that were filled out by the patients, parents, and therapists.

3.5.2.1 Individual Treatment Measures

All subjects completed a number of questionnaires - assessing

- Mood (Beck-Depressions-Inventar; [Hautzinger et al., 1995])
- Self-esteem (Skala zur allgemeinen Selbstwirksamkeit; [Schwarzer and Jerusalem, 1999])
- Social anxiety (Social Interaction Anxiety Scale; Social Phobia Scale; [Stangier et al., 1999])
- Alexithymia (Toronto-Alexithymie-Skala-26; [Kupfer et al., 2001])
- Mental state of condition (Befindlichkeitsskala; [von Zerssen, 1975])

- Cognitive schemata (Beck-Inventar zu kognitiven Schemata; [Fydrich et al., 2001])
- Motivational schemata (Fragebogen zur Analyse Motivationaler Schemata; [Holtforth and Grawe, 2002])
- Subjective image of the family (Subjektives Familienbild; [Mattejat and Scholz, 1994])
- Family functioning (Familienbögen; [Cierpka and Frevert, 1995])
- Hedonism (Marburger Untersuchungsinstrumentarium; [Lutz, 1999])
- Psychopathology (Youth Self- Report; [Döpfner et al., 1998])
- Treatment assessment (Fragebögen zur Beurteilung der Behandlung; [Mattejat and Remschmidt, 1999])
- As well as two inventories tapping aspects of eating disturbance (Eating Attitudes Test; [Garner and Garfinkel, 1979]; Eating Disorder Inventory; [Garner et al., 1983]).

A semi-structured interview and the Morgan-Russell Scale [Morgan and Hayward, 1988] were used to obtain details of the history and the clinical manifestations of the illness, the psychosocial functioning of the adolescent, the nutritional status, the menstrual functioning, mental state, psycho-sexual adjustment and socio economic status. Also the presence of bingeing, vomiting, abusing of laxatives, presence of mood disturbance or other psychopathologic comorbidity like obsessional symptoms and psychosomatic tension were obtained. A thorough medical examination assessed body weight (BW), body height (BH), body mass index (BMI), presence of other health related problems or neurological problems.

3.5.2.2 Family Measures

All family members completed the family assessment questionnaires “Familienbögen” [Cierpka and Frevert, 1995] and “Subjektives Familienbild” [Mattejat and Scholz, 1994], and the parents the CBCL/4-18 (Child Behavior Checklist) [Achenbach and Edelbrock, 1983]. During the course of our study the family members also completed the “Fragebögen zur Beurteilung der Behandlung” [Mattejat and Remschmidt, 1999]).

3.5.2.3 Therapists Measures

Group therapists also completed the treatment assessment questionnaires “Fragebögen zu Beurteilung der Behandlung” [Mattejat and Remschmidt, 1999] during the course of treatment.

To analyze the time-oriented, highly-structured, categorical data from the various assessed questionnaires, a new visualization technique had to be invented (as stated in the methodological overview in Chapter 2). Therefore, we developed Gravi++ that will be described in the following Chapter in detail.

Chapter 4

Gravi++

In the following part we will describe our Information Visualization named Gravi++ in detail. The name Gravi stands for “gravitation” and the two summation signs stand for magnetic energy. Both name parts stem from metaphors that were used during the development process for describing the spring-based core visualization.

First, we will present you the characteristics of the data type that can be used with Gravi++ in detail. After that, we will present the different methods used in our visualization in detail. Finally, we will discuss the benefits and limitations of Gravi++.

4.1 Data

Gravi++ is designed to visualize high dimensional, time-oriented data. As it was developed for the analysis of psychotherapeutic questionnaires, it is optimized for a limited number of weighted and ordered categorical data ranging from 0 to 6. There are different ways where the input data for Gravi++ might come from. A currently developed generic data editor called GenEdit [Osztoivits, 2007] is the main starting point. The data can be either manually entered or imported from a defined CSV (comma separated value) file. The next step can either be to export the values as a conflated data set directly to Gravi++ or use LinkVis [Herzog, 2004] to further modify and analyze the data. The synchronization of the data range can be either done in GenEdit (export option) or with LinkVis. In addition to this, LinkVis also lets the user define functions to concatenate questions to a single representing value (question). For example, whole questionnaires with a specific meaning can be summarized by a single score value that represents the meaning of the questionnaires. This way a manual dimension reduction can be applied on the data set. See Figure 4.1 for a detailed overview on the data workflow.

In the following we will concentrate on the data input format for Gravi++ in detail.

4.1.1 Data Type

As mentioned in Chapter 3 the medical data we want to analyze is categorical, time-oriented, and high dimensional.

4.1.1.1 Categorical Data

Categorical data means that the data is divided into groups. For example a data dimension “hair color” could have the characteristics “blond”, “brown”, and “other”. The values of a categorical data set can be unordered also called nominal (like the “hair color” example) or ordered also called ordinal (=, !=, <, and > relations). For our spring-based core visualization an ordered set with “<” or “>” relations can be analyzed best. Still unordered data sets can be visualized but is harder to analyze (see Section 4.2 for more details). A further restriction for easier analysis is that the data ranges should be synchronized. That means all data values should have an equal range of values because otherwise the visualization would be strongly unbalanced and not using all possible screen space. Our medical data set was synchronized in

a preprocessing step using ListVis [Herzog, 2004] to values ranging from 0 to 6. Alternatively, the export option of GenEdit [Osztoivits, 2007] can be used to synchronize the data.

4.1.1.2 Time-Oriented Data

As mentioned in Chapter 2, Section 2.1.4, time is an outstanding dimension that has to be treated in a special way. Gravi++ handles the time dimension separately and does not visualize it like the other dimensions. To achieve this, each data value has to be annotated with a time-stamp. This time value is split up in three fields “Day”, “Month”, and “Year”. In addition to this, the timesteps of a number of questions (usually a questionnaire) are concatenated to a timespan in so called data records. This concatenation is currently used to synchronize the questions in time.

4.1.1.3 High Dimensionality

One specialty of our data set is the high dimensionality. In our case, it is important to realize the coherence between the single dimensions and provide means to analyze them together. Looking at only one or two dimensions at a time makes it hard to reveal links between dimensions. Therefore, we decided to visualize as many dimensions as possible at a time to reveal correlations in the different dimensions.

To efficiently visualize the combination of the above mentioned characteristics is one of the main challenges of our visualization. To work with this challenging data we decided to use an XML based file format which will be introduced in the next section.

4.1.2 Data File Format

The data input files for Gravi++ are divided into two parts. One part is the master data, which describes the questions and questionnaires in detail. It is loaded using an XML file named

```
parameter_master_data.xml
```

The second part contains the data values themselves or in other words the answers to the specified questions. This file is also using XML and is named

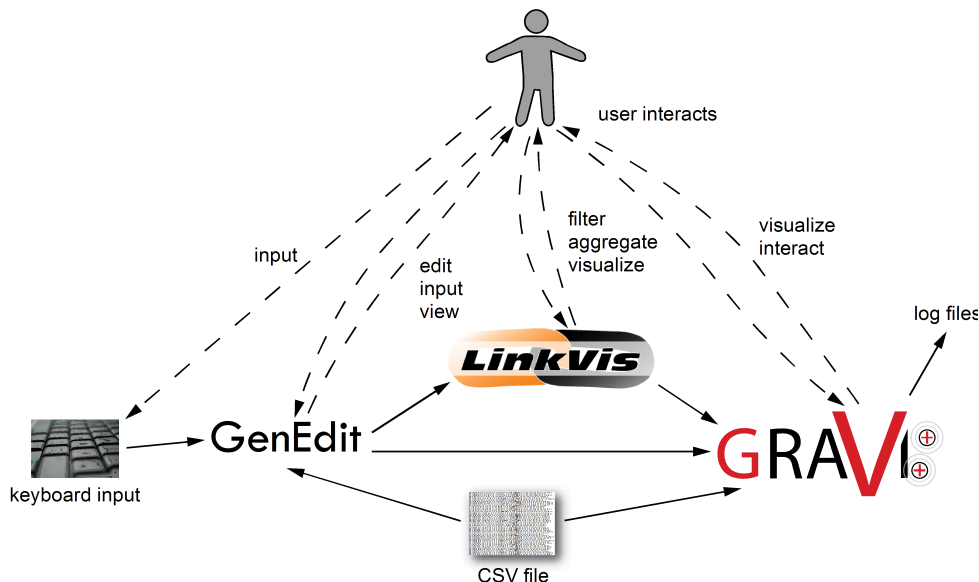


Figure 4.1: Gravi++ was designed for various data sources. The starting point can be the user manually entering the data or specifying a CSV file in the GenEdit data input tool. The next step can be LinkVis where various data manipulation tasks, like aggregation can be performed. Finally, the data is loaded into Gravi++ and analyzed there. Alternatively, a CSV file can directly be imported in Gravi++ by converting it to XML.

conflation_data_set.xml

Both data files are originating from LinkVis [Herzog, 2004] by Herbert Herzog. LinkVis is a visualization tool with sophisticated data transformation functions for our medical data. It was also developed especially for our problem domain and can therefore act as a preprocessing tool to select and modify the data that should be analyzed with Gravi++.

4.1.2.1 Parameter Master Data

The master data file consists merely of descriptive information on the questionnaires and their questions. The file contains the following information:

- Name of the questionnaires
- Name or identifier of the questions
- The formulation of the questions
- Data types of the answers
- Allowed answer ranges
- Mean values (neutral answer)

Each questionnaire consists of a number of questions and is identified by a unique name. The questions themselves are identified by the name of the questionnaire and a number. The data format allows the questions to have different data types. The questions can be of the types integer, float, and string. Gravi++ only supports the data types integer and float (technically).

The XML file format of the parameter_master_data.xml consists of three XML elements:

<linkvis_parameter_master_data> This is the root element of the document and it is only allowed once per file.

<parameter_group> The parameter_group element groups together parameters of the same subject. It features two attributes “id” and “count”. The attribute “id” gives the parameter group an unique name and the attribute “count” keeps the number of parameters in the parameter group.

Example:

```
<parameter_group id="FAQ" count="3">
```

This means a parameter_group named FAQ with three <parameter> elements were defined in the data set.

<parameter> Each parameter is described with the element <parameter>. It features six attributes “id”, “data_type”, “min_value”, “max_value”, “mean_value”, and “name”.

The attribute “id” is a unique identifier of the parameters within the parameter group. The attribute “data_type” indicate the data type of the expected measurement values. There are three different data types: integer, float and char. The parameters with the data_type=“integer” or “float” have optionally the attributes “min_value”, “max_value” and “mean_value”. The attribute “min_value” sets the allowed minimal value. The attribute “max_value” sets the allowed maximal value. The attribute “mean_value” sets an expected mean value of the parameter but it is not always the arithmetic mean value (could also be defined as a default value). The attribute “name” keeps the name or a formulation of a question. Each parameter can optionally list all allowed values between the minimal and maximal values. The sub element <allowed_values> lists with the sub elements <allowed> all allowed values. Each <allowed> element has an attribute value that holds one allowed value.

Example:

```
<parameter id="2" data_type="integer" name="How many good friends do you have?"
  min_value="0" max_value="30" mean_value="3"></parameter>
```

The example describes a question named “How many good friends do you have?” with `id="2"` that allows integer values between 0 and 30.

Example Data

```
<linkvis_parameter_master_data>
  <parameter_group id="FAQ" count="3">
    <parameter id="1" data_type="integer" min_value="1" max_value="5"
      mean_value="3" name="formulation of question one">
      <allowed_values>
        <allowed value="0"/>
        <allowed value="1"/>
        <allowed value="2"/>
        <allowed value="3"/>
        <allowed value="4"/>
        <allowed value="5"/>
      </allowed_values>
    </parameter>
    <parameter id="2" data_type="float" min_value="-5.5" max_value="+7.7"
      mean_value="0" name="formulation of question two">
    </parameter>
    <parameter id="3" data_type="char" name="formulation of question three">
    </parameter>
  </parameter_group>
</linkvis_parameter_master_data>
```

4.1.2.2 Conflation Data Set

The conflation data set consists of all answers on the questions. It contains only data from a specified experiment in LinkVis. This measurement data have the following information:

- Patient identifier
- Additional information about the patient: who has answered the questionnaire
- Recording date information (day, month, and year)
- Questionnaire identifier
- Question identifier (number for example)
- Answer

The patient identifier can be a number or string and helps to link to the patient data that is not visualized or included in the data files. The additional information about the patient informs us who has answered the questionnaire. This can be either the patient (encoded with a “C”), mother (“M”) or father (“F”). The recording date is the information on when the question was answered. The smallest unit in the data format is a day. Each answer consists of the answer value, the questionnaire name, and the question identifier. The question identifier and name are the link to the master data to get additional information on the question and questionnaire.

<linkvis_conflation_data_set> This is the root element of the document and it is only allowed once per file.

<data_record> The “data_records” element describes the record information without measurement values. It has the attributes “parameter_source_id”, “parameter_group_id”, “from_day”, “from_month”, “from_year”, “till_day”, “till_month”, “till_year”, “rowid”, and “label”. “parameter_source_id” (=patient identifier) and “parameter_source_flag” (=additional information about the person) are the combined unique identifier of each person. The attributes with “from” and “till” in the name describe the interval, in which the questions in the following parameter element were asked. These dates are stored in the three attributes: “*year”, “*month” and “*day” (* standing for “from” or “till”). The attribute “year” consists of 4 digits. The attribute “month” consists of 1 or 2 digits and the attribute “day” consists of 1 or 2 digits. “rowid” is the LinkVis record identifier, that is not used in Gravi++ (usually a combination of all attributes as a string). The tag “label” is storing the date label for the specified time span (string).

Example:

```
<data_record parameter_source_id="10" parameter_source_flag="C" from_day="01"
  from_month="08" from_year="2002" till_day="31" till_month="07" till_year="2003"
  rowid="10 C 2002-08-01 00:00:00 2003-07-31 00:00:00" label="31.07.2003">
```

This example shows a data record for person 10C in the time span from 2002/08/01 to 2003/07/31 and the label “31.07.2003”.

<parameter> The “parameter” element describes the measurement values with the exact measurement dates. This means the question identifier, answer and measurement date are stored in this element. It consists of the “group_id”, “id”, “value”, “day”, “month”, and “year” attributes. The “group_id” stands for the questionnaire and the “id” is the unique identifier of the question in the questionnaire. “value” contains the data value and “day”, “month”, and “year” contain the date of the measurement.

Example:

```
<parameter group_id="B-IKS" id="2" value="3" day="20" month="07" year="2003"/>
```

This example shows the question 2 in the questionnaire “B-IKS” that was answered with the value “2” at 2003/07/20.

Example Data

```
<?xml version="1.0" encoding="UTF-8"?>
<linkvis_conflation_data_set>
  <data_record parameter_source_id="10" parameter_source_flag="C" from_day="01"
    from_month="08" from_year="2002" till_day="31" till_month="07" till_year="2003"
    rowid="10 C 2002-08-01 00:00:00 2003-07-31 00:00:00" label="31.07.2003">
    <parameter group_id="B-IKS" id="2" value="3" day="20" month="07" year="2003"/>
    <parameter group_id="B-IKS" id="1" value="2" day="20" month="07" year="2003"/>
  </data_record>
  <data_record parameter_source_id="11" parameter_source_flag="C" from_day="01"
    from_month="08" from_year="2002" till_day="31" till_month="07" till_year="2003"
    rowid="11 C 2002-08-01 00:00:00 2003-07-31 00:00:00" label="31.07.2003">
    <parameter group_id="B-IKS" id="2" value="2" day="05" month="05" year="2003"/>
    <parameter group_id="B-IKS" id="1" value="2" day="05" month="05" year="2003"/>
  </data_record>
  <data_record parameter_source_id="12" parameter_source_flag="C" from_day="01"
    from_month="08" from_year="2002" till_day="31" till_month="07" till_year="2003"
    rowid="12 C 2002-08-01 00:00:00 2003-07-31 00:00:00" label="31.07.2003">
    <parameter group_id="B-IKS" id="2" value="4" day="27" month="06" year="2003"/>
    <parameter group_id="B-IKS" id="1" value="4" day="27" month="06" year="2003"/>
  </data_record>
  <data_record parameter_source_id="0" parameter_source_flag="C" from_day="01"
```

```

    from_month="08" from_year="2002" till_day="31" till_month="07" till_year="2003"
    rowid="0 C 2002-08-01 00:00:00 2003-07-31 00:00:00" label="31.07.2003">
    <parameter group_id="B-IKS" id="2" value="4" day="28" month="08" year="2002"/>
    <parameter group_id="B-IKS" id="1" value="5" day="28" month="08" year="2002"/>
  </data_record>
</linkvis_conflation_data_set>

```

Additional information on the data format and can be found in Herbert Herzogs master thesis [Herzog, 2004].

4.1.2.3 Other Suitable Data

In addition to the medical data set we used for our insight study (see Section 3 on the medical problem domain), Gravi++ can also be used for other domains and data sets. We demonstrate the analysis of our medical data and another data set (namely Eurobarometer data) in Chapter 6. In addition to this, the following data could be analyzed for example as its structure is similar to our medical data set:

School marks Usually pupils get marks on passed exams. These marks could be collected over time. In Austria for example the school marks range from 1 to 5. Therefore, the data would perfectly fit for our visualization. This way it could be interesting to see how the pupils of a class perform over time and if there are clusters of pupils who are good at some subjects and bad in others.

PISA scores Similar to school marks, the results of the PISA (Program for International Student Assessment) studies [OECD, 2006] from the OECD could be visualized using Gravi++.

Country statistics Different data values of different countries like population, income, crime rate, buying power, or inflation could be analyzed for clusters, outliers, and differences in their development. The different variables would have to be synchronized to show values of equal range. Otherwise, the visualization would not use up the whole screen space (as for example the population dimension would show very high values) which would result in a difficult analysis of the data. This preprocessing could be done with GenEdit [Osztoivits, 2007] or ListVis [Herzog, 2004].

Popularity of political parties The popularity of political parties could be evaluated and collected as scores. This score could be used to visualize the political orientation of voters and detect similar groups of voters that prefer related parties.

SOFA scores The SOFA (Sepsis-related Organ Failure Assessment) score [Vincent et al., 1996] is a rating mechanism of organ dysfunction in an intensive care unit. It was defined by the European Society for Intensive Care Medicine (ESICM). The SOFA score is a scoring system to determine the extent of a person's organ function or rate of failure. The score is based on six different scores based on the respiratory, cardiovascular, hepatic, coagulation, renal and neurological systems and is graded based on the system's failure and dysfunction. As the scores are collected over time and all have a similar range, they could be perfectly visualized using Gravi++.

4.2 The Method

Gravi++ consists of two overview visualizations, a main spring-based core visualization, and three additional views on the data. The two overview visualizations called ListVis and TableVis provide an overview of the whole data set and let the user choose an appropriate subset. This subset can then be analyzed in detail with the spring-based core visualization and the three additional views called Traces, Attraction Rings, and Star Plot (Star Glyph). These additional views provide further information that is difficult to grasp with the core visualization (for example the exact answers on each question). In Figure 4.2 you can see the structure of Gravi++.

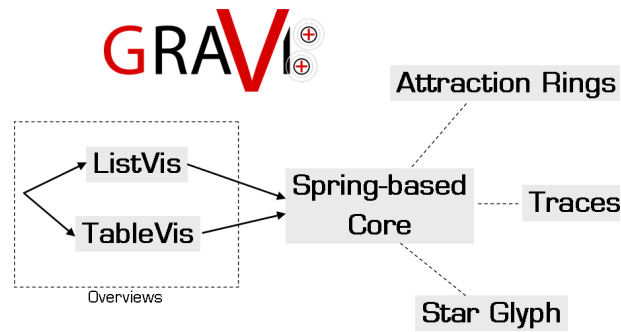


Figure 4.2: Structure of Gravi++. Gravi++ consists of two overview visualizations called ListVis and TableVis. With these two integrated visualizations the user can choose an appropriate subset and analyze the data with the spring-based core visualization. In addition to this, the user has the opportunity to add three additional views called Traces, Attraction Rings, and Star Plot (Star Glyph).

4.2.1 Spring-Based Core Visualization

The spring-based core visualization uses icons to visualize the data on a two dimensional plane (background of Gravi++). The visual parameters position, movement, size, color, structure, and shape are used to map the data on the screen. There are two kinds of icons (see Figure 4.3):

1. Rectangular icons representing questions.
2. Circular icons representing persons.

The position of the question icons can be chosen manually. They can be dragged around freely or placed on a circle. Depending on the position of the question icons, the person icons are positioned using a spring-based algorithm (see Section 2.1.10). The higher a question was answered by a person, the stronger the attraction force of the question icon to this person icon is. In our spring-based algorithm each person icon is connected with each question icon by an invisible spring. The higher the question was answered, the stronger the spring is. In Figure 4.4 one person icon is attracted by three question icons with different strengths. The person icon is positioned nearer to the questions that were answered with higher values. An important characteristic of a spring is, that the further away a person is from the question, the larger the attraction force is. The detailed algorithm that implements this spring-based approach can be found in Section 4.2.2.

As each person icon is connected with a (invisible) spring to each question, the person icons are dragged to a position where a balanced attraction in each direction is given. The user can now analyze the positions of the icons (both question and person) and conclude on the underlying data. Especially, clusters, outliers, and other patterns can give important hints on interesting data structures and lead to new insights in the data.

In Figure 4.5 four person icons form a cluster near question EAT13. Therefore, we can conclude that this question was answered with higher values than the other two questions. In addition to this, we can for example also conclude that “Person 3” has answered FAMOS14 only with 0 as the icon is positioned between the other two question icons.

To discriminate the different persons visually and to categorize the persons, different colors can be used to display the icons. For example, using green to visualize a good therapy outcome and red for the opposite can help in finding clusters of different therapy outcome and classify patients that are still in therapy (and therefore not classified yet). See Figure 4.6 for two differently colored clusters that also form a spatial cluster.

In addition to assigning a color value, Gravi++ displays the person icons with an adjustable transparency value. This way, overlapping icons can still be differentiated (up to a limited extent depending on the chosen level of transparency) and clutter is reduced.

To discriminate different questions without labels, Gravi++ supports to display small corresponding images on each question icon. These images can be user defined. Each questionnaire can be assigned a different image.

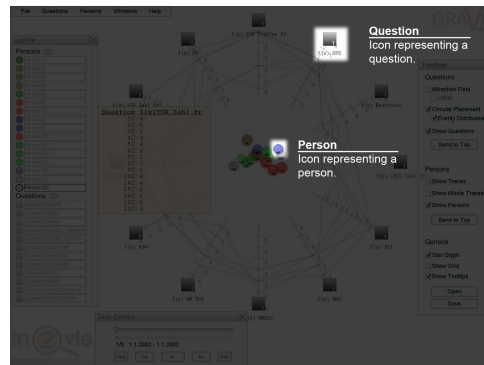


Figure 4.3: The round icons in the center represent the persons and the rectangular icons represent the questions.

The spring-based algorithm is a dimension reduction technique that reduces the n -dimensional problem space (for n questions) to two dimensions (the screen space). As every dimension reduction technique, this spring based approach also suffers from some loss of information. Person icons can be placed over each other although the questions on the screen have been answered differently. For example if a person answers all questions with “1” and another person answers all questions with “6”, both icons would be displayed on the same position on the screen (e.g., in the center if all question icons are placed on a circle with equal space between them). Gravi++ counters this limitation with the possibility to map the strength that is currently influencing the icon (from the springs) on the size of the person icons. For example the person that answered all questions with “6” would have a bigger sized icon than the person who answered all questions with “1”. In addition to this, the size of the persons can also be mapped on any numerical data dimension (for example the weight of the person) or the user can choose that the size should increase over time.

The size of the question icons is also mapped to an important interaction feature of Gravi++. The strength of all springs connected to a question can be changed manually to emphasize single questions. This can be done by multiplying the answer with a manually set strength value ($value_{ij}$ in the algorithm) of the question. The default strength value is “1” and can be altered for each question by the user in the interface (see Section 4.2.3 on the possible interactions). The changed strength value for a question results in a bigger sized question icon. Furthermore, the chosen strength value is displayed in the corner of the question icon. With these two graphical mechanisms, the user can visually control which question he or she has emphasized on. This is very important, as the chosen strength of each question has an essential impact on the interpretation of the person icons’ positions.

The starting-size that is taken as a root value for sizing question and person icons can be altered in the reference file.

Another possibility to overcome the limitation of overlapping data points, is to use one of the three additional views. Especially, the Attraction Rings and the Star Plot were designed to visualize the exact values. Therefore, both views do not suffer from information loss due to dimension reduction techniques.

In order to make the abstract time dimension visible, some kind of mapping has to be applied. For visual representation, two fundamental ways of mapping exist [Aigner, 2006]:

1. Time \rightarrow Time (Animation)
2. Time \rightarrow Space

Mapping “time to time” in visual terms means animation. This is probably the most natural form of graphically mapping the time dimension because there is no conversion of concepts needed in between. Mapping time to time is well suited for the tasks of keeping track of changes or following trends and movements. But for analytic and explorative tasks, this kind of mapping is not very well suited since no direct comparison of attributes between different points in time is possible [Aigner, 2006].

This means, the ideal solution for visualizing time would be to use both techniques. Therefore, we decided to implement both “time to time” and “time to space” mappings in Gravi++. “Time to time”

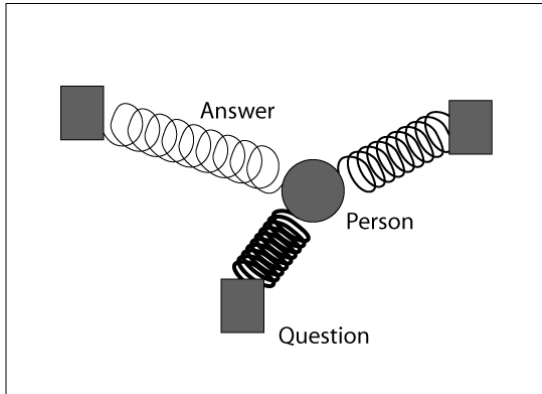


Figure 4.4: Each circular person icon is connected with the rectangular question icons by (invisible) springs. The strength on the spring depends on the answer that was given on this question. The lower question was answered with the highest value. Therefore, the attraction to it is higher than to the other questions. The right question was answered with the second largest value and, therefore, the person icon is positioned nearer to these two question icons than to the left one.

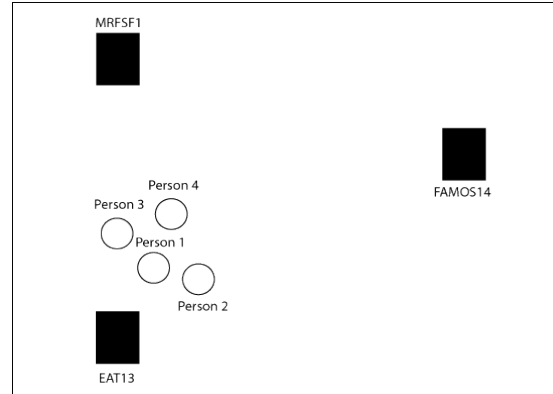


Figure 4.5: Four person icons form a cluster near question EAT13. Therefore, we can conclude that this question was answered with higher values than the other two questions.

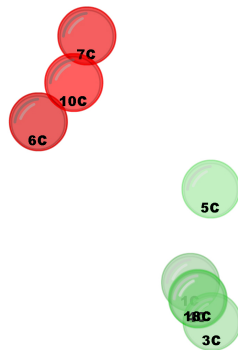


Figure 4.6: Two clusters of differently colored person icons.

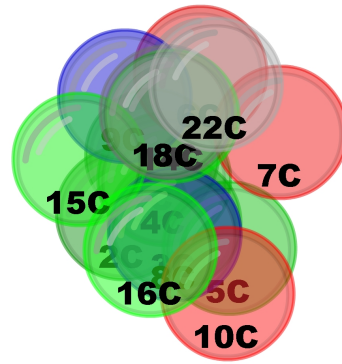


Figure 4.7: Gravi++ uses adjustable transparent colors for the person icons. This helps detecting overlapping icons.

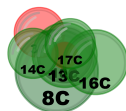


Figure 4.8: Gravi++ can map different parameters like strength, time, or a specific dimension by varying the size of the person icon.



Figure 4.9: Symbolic image of a person icon moving toward the next position (because of a change in time).

mapping is used for the core visualization. As the persons gave different answers over time, the attraction force of the spring-based algorithm also changes over time. This means, the position of the person icons is changing over time. The transition between the two positions is displayed by a linear animation (with controllable speed). When changing a timestep, all person icons move to the next position according to the new attraction force. To control the movement over time, Gravi++ offers a window with a timeline and video recorder like control buttons. This enables the user to move to a specific timestep, play automatically through time, and manually step forward and back through time.

In addition to the “time to time” mapping with the animation, the “time to space” mapping is visualized using the separate additional view called Traces (see Section 4.2.4.3). This way Gravi++ is suited for the tasks of keeping track of changes, follow trends and movements because of the animation and for analytic and explorative tasks because of the Traces.

4.2.2 Algorithm

To compute the attraction force of the questions on the persons, two types of algorithms can be used. The problem can either be solved algorithmically or with an iterative approach. Gravi++ was first using the iterative approach and later the linear algorithm (because of performance advantages).

4.2.2.1 Linear Algorithm

The algorithmic solution solves the equation to get the final position of each person for the two axis x and y. The force applied to each person i from each question j for each axis (in our case for the x and y axis) is $xforce_i = \sum_{j=0}^n strength_j \cdot value_{ij} \cdot f(qx)$ and $yforce_i = \sum_{j=0}^n strength_j \cdot value_{ij} \cdot f(qy)$, where $strength_j$ stands for the manually set strength multiplier of the question j (this can be altered in the user interface and ranges from 0 = no attraction to n = an arbitrary value). $value_{ij}$ stands for the answer of person i to question j . $f(qx)$ and $f(qy)$ stand for the attraction over the distance for each axis. Currently, we use a linear function that grows stronger over the distance: $f(qx) = qx_j - px_i$, $f(qy) = qy_j - py_i$. Here qx_j and qy_j stand for the x, y coordinates of the question j and px_i and py_i stand for the x, y coordinates of the person i .

Now we can analytically solve the equations by substituting the force with zero ($xforce_i = 0$ and $yforce_i = 0$), to find the final position for each person (px_i, py_i).

In our case (with the linear function) the following equations determine for the coordinates of the person i :

$$px_i = \frac{\sum_{j=0}^n strength_j \cdot value_{ij} \cdot qx_j}{\sum_{j=0}^n strength_j \cdot value_{ij}} \quad \text{and} \quad py_i = \frac{\sum_{j=0}^n strength_j \cdot value_{ij} \cdot qy_j}{\sum_{j=0}^n strength_j \cdot value_{ij}}$$

4.2.2.2 Iterative Algorithm

Alternatively, we can also use an iterative algorithm for solving the above equation. This concept could be used to enable the user to easily change the attraction functions $f(qx)$ and $f(qy)$ without the need to solve the equation. To calculate the exact position of each person, the following iterative algorithm is used:

For each person i do:

1. Calculate the force for each question j for each axis (in our case for the x and y axis):

$$xforce_i = \sum_{j=0}^n strength_j \cdot value_{ij} \cdot f(x)$$

$$yforce_i = \sum_{j=0}^n strength_j \cdot value_{ij} \cdot f(y)$$

Where $strength_j$ stands for the manually set strength multiplier of the question j (can be altered in the user interface and ranges from 0 = no attraction to n). $value_{ij}$ stands for the answer of person i on question j . $f(x)$ and $f(y)$ stands for the attraction over the distance for each axis. An easy case would be a linear function that grows stronger over the distance: $f(x) = x_j - dx_i$, $f(y) = y_j - dy_i$.

Here x_j and y_j stand for the coordinates of the question j and dx_i and dy_i stand for the coordinates of the person i .

2. Move one step left if $xforce_i < -\Delta$ or one step right if $xforce_i > \Delta$. Move one step up if $yforce_i < -\Delta$ or one step down if $yforce_i > \Delta$. Δ stands for the minimal threshold.

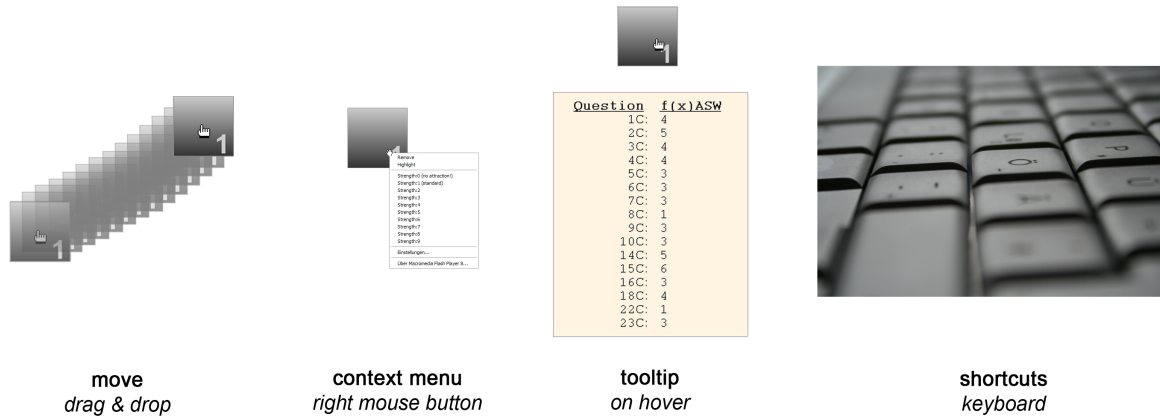


Figure 4.10: The interactions in Gravi++ can be accessed using four different interaction paradigms. Drag & drop (e.g., for moving a question), context menu (offering all possible options), details on demand – on hover (tooltip), and keyboard shortcuts (for a fast access of often used interactions) are offered.

3. If a step in any direction was done in step 2, go to step 1.

Both algorithms can be easily extended for a 3D visualization by just adding a third equation for the z-dimension (depth).

4.2.3 Interactions

Gravi++ was intended as an interactive Information Visualization that should make use of the expert users background knowledge and enable them to “play” with the data. Therefore, we tried to integrate as much helpful interactions as possible. The main interaction paradigm of Gravi++ is “drag and drop”. This means the user can click on graphical objects and drag them to another location. According to the new position, different actions may happen. For example changing the position of a question icon or removing it by dragging it to the overview visualization ListVis. In addition to drag and drop, Gravi++ also supports “on hover” effects. That means when the mouse cursor is placed over graphical objects, a tooltip containing further information appears. Furthermore, context menus can be accessed using the right mouse button and our visualization also supports keyboard input (short keys). In Figure 4.10 the four different interaction paradigms are illustrated.

All interactions can be classified in three categories: (1) interactions on question’s icons, (2) interactions on person’s icons, and (3) general interactions. In the following these categories are explained in detail.

4.2.3.1 Interactions on Question’s Icons

Add or remove a question A question can be added by simply dragging it from the overview ListVis to the background. Removing a question can be done the opposite way (by dragging it back to ListVis). In addition to this, a context menu entry that can be accessed on the question icon allows to add or remove the person. Removing all question icons can be achieved by using the appropriate option in the context menu on the background or by choosing this option in the top menu bar.

Change the position of a question Changing the position of a question can be done by simply dragging the icon to the desired position. The position can be changed either freely or arranged on a circle. Furthermore, the user can choose to evenly space the distance between the questions on the circle. If the option “arrange on a circle” is chosen, the icon can still be dragged around, but a shadow of the icon (icon with reduced brightness) is projected to the corresponding position on the circle, where the question icon will be placed (see Figure 4.11). If the additional option to evenly distribute the placement

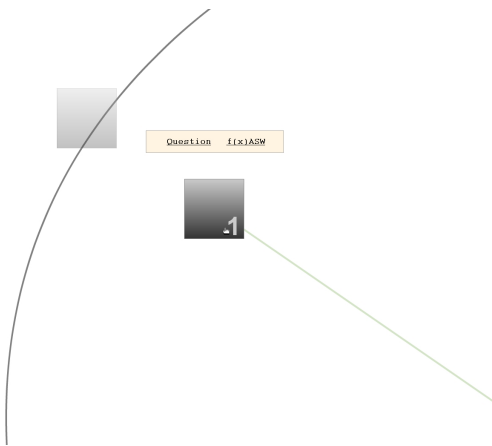


Figure 4.11: When the position of the question icons is restricted on a circle, Gravi++ displays a “shadow” of the final position when dragging a question icon around.

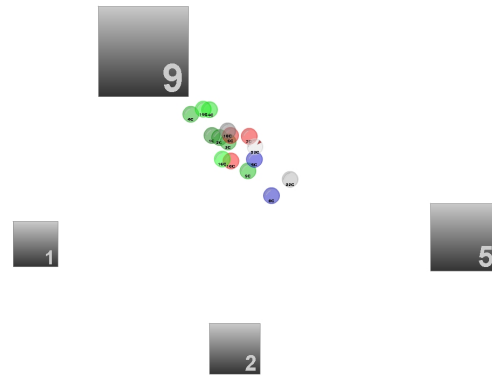


Figure 4.12: Four different question icons with different assigned strength values can be seen on this screenshot. The sizing of the question icons is based on this manually adjusted strength value and the gray number in the lower right corner shows the defined number. The strength value changes the influence of each question (strength of the spring) and lets the user therefore emphasize or de-emphasize questions.

on the circle is activated, the arrangement is calculated the following way. The questions are arranged evenly from the right position on the circle (90°) clockwise.

With this feature the user can interactively change the positions of the questions to find new clusters or other interesting visual structures. All other views are also updated during the dragging process. This feature is called ‘live preview’. This way the user can easily find interesting question constellations. See also Gee [Gee et al., 2005] for more information about the positioning of anchors in spring-based visualizations.

Change the influence of a question The influence or strength of the spring can be changed using the context menu. This enables the user to emphasize the influence of a chosen question. To visualize the current set strength of a question, the strength is shown as a number in the icon and in addition to this the icon is sized according to this manually adjusted influence.

In Figure 4.12 four different questions are shown with four different strength values. The upper question has the highest influence, therefore, it is visualized with the biggest icon and shows a high attraction of the person icons.

Add a complementary question By executing this option on a question, a new question with complementary values is created. This option can be used to analyze a single parameter (question) with the spring-based core visualization. It can be accessed in the context menu of any question icon.

Highlighting a question Highlighting a question, displays the corresponding question icon in red. When the Attraction Rings (see Section 4.2.4.1) are enabled, only the rings around highlighted questions are shown. If no question is highlighted, all rings are shown. Highlighting can be enabled in the overview visualization ListVis or in the context menu of the corresponding question icon.

Hide question icons All question icons can be hidden to reduce clutter and facilitate the analysis of the other views like Traces or the Star Glyph. This option can be accessed in the top menu bar and the toolbar window.

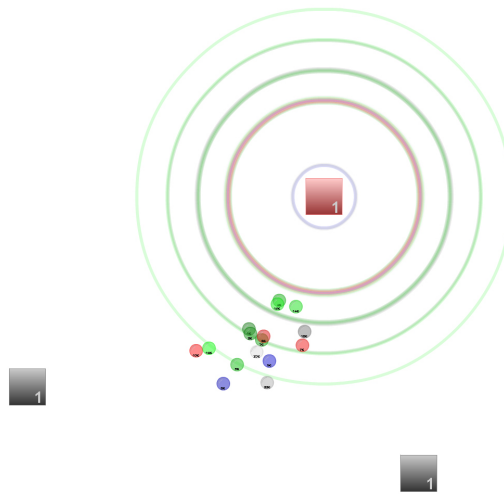


Figure 4.13: The highlighted question icon is shown with a red overlay. In the image the Attraction Rings are enabled. When a question is highlighted, only the Attraction Rings of the highlighted questions are visualized.

Fix question value over time This fixes the current values of the question over time. This can be useful if a question was only answered once through the whole timesteps but is always relevant. This option can be enabled in the top menu bar and the toolbar window.

Random question position The positions of the question icons can be randomized using this function. With this function the user can play around and find clusters or interesting structures or also verify hypothesis about clusters. The option can be enabled in the top menu bar or using the keyboard shortcut key “r”. This option also works when the questions are (evenly) arranged on a circle. The questions are placed on random locations on the screen and then transferred back to the circle. This process is not visible, but only the final positions do change.

Details on Demand Placing the mouse cursor over a question icon, displays a graphical tooltip containing the whole question name and the exact values of all answers that were given on this question in a tabular structure (see Figure 4.14). The tooltip is shown semi-transparent and is automatically positioned either above or beyond according to the position of the question icon. The tooltip can also be turned off in the top menu bar and the toolbar window to reduce clutter when displaying many questions (and big sized question icons).

Change the image of question icons By placing an image file with the name of the questionnaire in the main directory of Gravi++ and enabling the appropriate option in the resources file, Gravi++ automatically loads the image and displays them on the standard question icons.

4.2.3.2 Interactions on Person’s Icons

The following interactions can be performed with person icons.

Add or remove a person’s icon Similar to the question icons a person’s icon can be added and removed by dragging it between the overview ListVis and the background or by using the context menu on the corresponding icon (works also in ListVis). In addition to this, all person icons can be removed using the top menu bar option or the context menu on the background (see Figure 4.15).

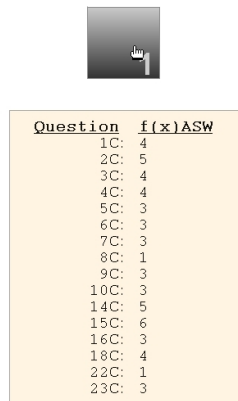


Figure 4.14: Placing the mouse cursor over a question icon shows the tooltip with detailed information on the exact values all persons have answered on this question. In addition to this, the whole question name is displayed in the semi-transparent tooltip.

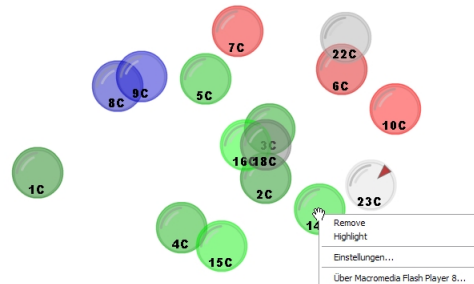


Figure 4.15: Removing and highlighting a person can be done by accessing the appropriate option in the context menu of the person icon.

Change the position of a person’s icon Changing the position of a person is automatically computed and displayed according to our spring-based algorithm. In addition to this, the user is able to drag the icon around. After releasing the mouse button, the person icon moves back to the computed position according to the current forces that affect the icon. Person icons without any attraction force at this timestep (e.g., because of missing values) can be dragged around without them moving back to the previous position. This feature can be used to drag apart overlaying icons, discriminate transparent and non-transparent icons (persons with data and without), and to separate persons with missing data from the others (and clearly see them moving as soon as they have data again).

Change the parameter representing the person icons size The parameter that is mapped on the person icons size can be changed in a menu that is accessible in the top menu bar. Four different parameters can be mapped on the size:

1. None: The size of all icons remains the same over time and is only depending on the person’s icon size in the resources file.
2. Attraction force: This option maps the “physical” force that is currently affecting the person icon (dragging it in the current position) on the size. This means that a question that is attracted by higher values is sized bigger than a question that is attracted by lower values. This helps discriminating persons that are mapped on the same position but have given different answers to the question. For example a person that answered all questions with “1” would be mapped on the same position as a person that answered all questions with “5” but the second icon would be sized bigger.
3. Increasing size over time: The person icons get bigger over time. This option makes most sense for analyzing the Traces (as this size parameter is also taken for the thickness of the Traces) as then the lines get thicker over time.
4. Data dimension (currently BMI (body mass index = weight factor) value): In principle every other data dimension can be mapped on the size of the persons. In our example we can map the BMI values on the size of the persons. The mapped dimension can also be a non-categorical number (integer or float) and does not need to be synchronized with the other dimensions (regarding the

value range). The minimum and maximum size that is used to visualize the size of the icons can be adjusted in the properties file.

As mentioned above, the size parameter is also used to calculate the thickness of the Traces view (see Section 4.2.4.3).

Highlight a person A specific person is singled out by using this option. The icon is visualized in red (similar to the question icons that are highlighted). Highlighting a person does also highlight the person in the overview visualization ListVis and in the other views (visualizing only the highlighted persons). Highlighting a person can be done in the context menu of the person’s icon, in the overview ListVis by clicking on the person (or also by using a context menu), or in the different views (by clicking on the representation for that person, e.g., on the corresponding Star Plot line).

Hide person icons All person icons can be hidden to reduce clutter and facilitate the analysis of other views like Traces or Star Glyphs. This option can be chosen in the top menu bar or in the toolbar for a quick access.

4.2.3.3 General Interactions

User interface interactions The overview visualizations, toolbar, status output, time control, help, save, and load windows do all provide the same interaction possibilities. The windows can be moved around by dragging the top border with the name in it. In addition to this, each window can be closed by clicking on the button with the “X” in it in the right upper corner. The windows can be opened again using the corresponding top menu bar entries. See Figure 4.17 for an example of the Time Control window that integrates these window interface elements.

Enable or disable views Using the top menu bar or the toolbar, the user can easily turn on or off the additional views Star Plot, Attraction Rings, or Traces.

Change the speed of movement Gravi++ lets the user adjust how fast the person icons are moving to their destination. The parameter can range from stop to fastest in four steps. Stop means that there is no movement at all. Lind and Kjellin investigated in [Lind and Kjellin, 2005] the relationship between mean speed of animated objects’ speed and the ease of learning a complex relationship defined between the moving objects. They found out a significant interaction between difficulty and speed. Therefore, we decided to let the user adjust the speed according to the complexity of his / her tasks manually. Kumiyo et al. [Nakakoji et al., 2001] also investigated interactions in animated visualizations.

“In viewing animation, all the subjects frequently used a control bar (the bar beneath the animation window) to change the speed of animation and to repeatedly view a certain segment of the animation.” [Nakakoji et al., 2001]

This shows the importance of allowing the user to change the speed of the animation like in Gravi++.

Show grid A grid with a marked center and a graphical circle can be blended in to ease the judgment of distances. The circle shows the possible positions of question icons, if they are positioned on a circle. Enabling the grid can be done with the top menu bar, the toolbar, and the context menu triggered on the background. See Figure 4.16 for an example image of the grid in Gravi++.

Save and load state Using the top menu bar or the toolbar the user is able to access the save and load windows. They both provide a list of all available saved states that can be selected. The save window also offers an input field to specify a name for the current state (that should be saved). Two buttons are located at the lower right bottom of the two windows. One to cancel the operation and close the window, and the other one to save or load the specified state.

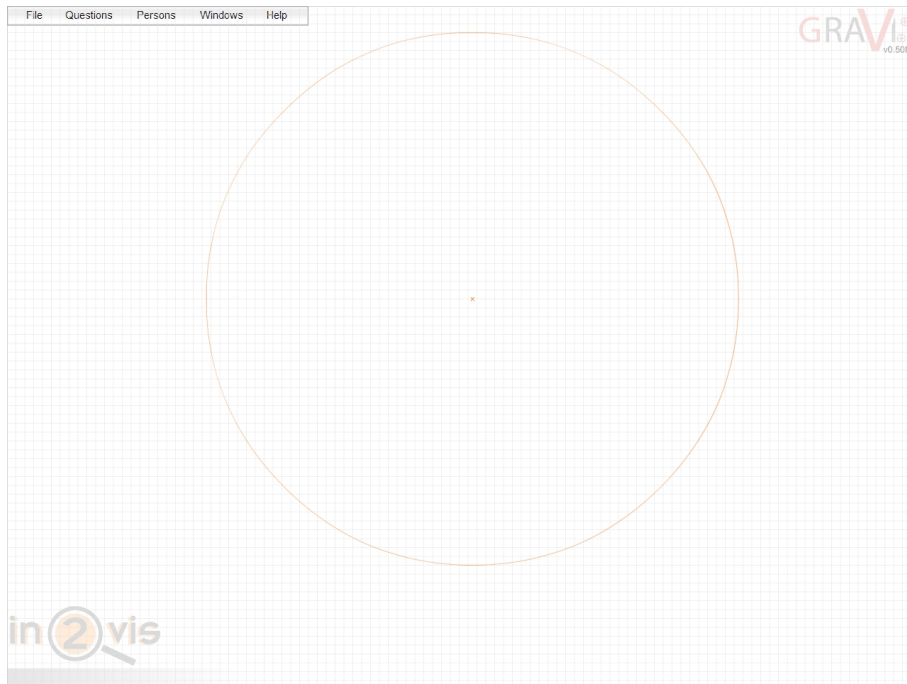


Figure 4.16: A grid with a marked center and an orange circle (on which the questions are positioned) can be displayed in the background to help the user judge distances.

Show help and about screen The top menu bar offers the option to display a window containing the help text. The help text is taken from the property file.

Reset window positions This option that is accessible through the context menu on the background or the top menu bar, resets all window positions to their preset locations.

Time control The time control window (see Figure 4.17) lets the user change the currently displayed timestep and gives feedback on the current position in time. It can be accessed in the top menu bar and offers all window interaction techniques (move and close).

The user can either select a specific timestep by clicking on the corresponding position of the slider or drag the arrow of the slider to the corresponding position. In addition to the interactions on the slider, Gravi++ offers five different buttons to navigate through time. This video recorder like controls offer (from left to right, see Figure 4.17) the options to move to the first timestep, move one timestep back, play through all timesteps automatically, move to the next timestep, and move to the last timestep. The different options to move through time are also presented in Figure 4.18.

Kumiyo et al. [Nakakoji et al., 2001] investigated what interactions of a time control people would like to have with animated visualizations when exploring data.

“Control over time: The subject (an expert programmer) very frequently held the play controller to fine-tuning the playing time and displaying certain frames of the animation. When the subject found interesting frames, they went back and forth between a certain timespan. Often the subject played a particular part of the animation extremely slowly by gradually moving a play-controller examining changes of data.” [Nakakoji et al., 2001]

“Differential representations: The subject depended on the presentations of the previous and next values.” [Nakakoji et al., 2001]

This shows the importance of providing interaction mechanism to go to the previous and next timestep and a mechanism play through all timesteps like Gravi++ supports.



Figure 4.17: The time control window in Gravi++ lets the user select a specific timestep, move through time, and gives feedback on the current position in time.

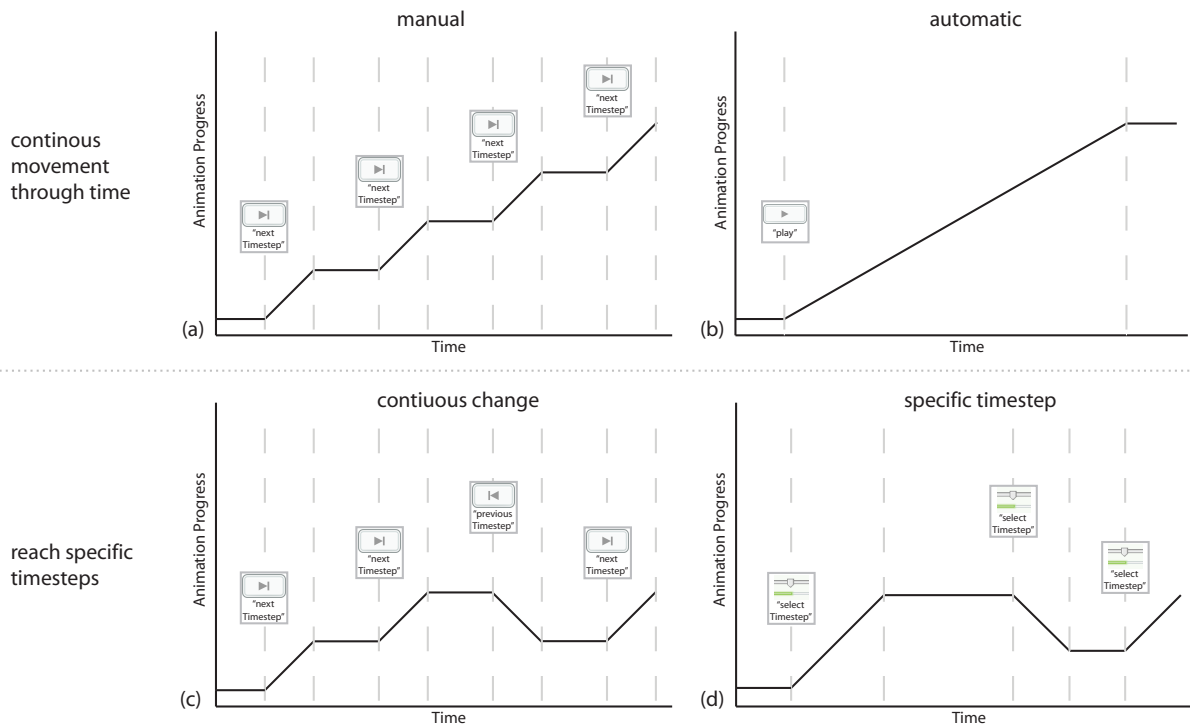


Figure 4.18: The time control window lets the user change the timestep with different interaction possibilities. He or she can either move through time (b), select a specific timestep (d), or step around with the “next” and “previous” buttons (a) / (c).

Zoom Zooming is one of the basic interaction techniques of Information Visualizations. The screen size and resolution limits the maximum amount of data items that can be visualized at a time. To overcome this limitation, zooming is an appropriate technique. Gravi++ supports geometric zooming, that means the user is able to magnify a specific area of interest. This area is then shown in an magnified image.

Like proposed in the Visual Information Seeking Mantra by Shneiderman [Shneiderman, 1996] and analyzed by Craft [Craft and Cairns, 2005], Gravi++ supports zooming.

“Zooming and filtering both involve reducing the complexity of the data representation by removing extraneous information from view and allowing for further data organization.” [Craft and Cairns, 2005]

Especially, when visualizing a large amount of data, zooming can help to cope with clutter and overlapping items.

In Gravi++, zooming can be achieved by calling the appropriate entries in the context menu of the background. The user has four different option:

- Magnify the view at the current position of the mouse cursor.
- Minimize the view at the current position of the mouse cursor (only available if the current view is already magnified).
- “Show everything” to reset the magnification to the standard view.
- 100% to show all visual items on the screen (this can be smaller than the standard view).

In addition to this, the user is able to navigate in the magnified area by dragging the background around which results in changing the current view.

4.2.4 Coordinated Visualizations

To complement the spring-based core visualization, Gravi++ offers a set of coordinated visualizations. The spring-based core offers low cluttering and a very good overview on one timestep but is less suited for providing quantitative information at a glance. To address this problem we integrated two views named Attraction Rings and Star Plot to visualize the exact values. In addition to this, we integrated a view called Traces to map the time dimension on the display space (in contrast to the core visualization that maps time to time using animation). This way, we can better support analytic and explorative tasks because it is possible to directly compare different points in time and watch the animation.

4.2.4.1 Attraction Rings

The Attraction Rings (see Figure 4.19) are a simple visualization to provide quantitative information on the currently visualized data. For this cause rings are displayed around the question icons for each answer. The ring size corresponds with the attraction to the question and the color is equal to the one used for the person icons. To avoid overlapping rings with the same value, they can be put closely side by side. This only makes sense when using a data set with discrete steps, like ours, and when visualizing a limited number of data items. If too many data items with the same value are visualized side by side, the space between two categories is getting too small. This means, that the question rings are then visualized closer to the next category than to the one they belong. A solution would be to automatically size the distances between the categories to ensure the comparability. Currently this is not integrated in the prototype but the space between the categories can be adjusted in the properties file and the rings are only painted in the space between two categories (if too much rings are painted, the last position gets overdrawn).

Another method against clutter is that the transparency value of each ring can be changed in the property file. This way overlapping circles can also be discriminated up to a given amount that depends on the adjusted alpha value. As Gravi++ visualizes temporal data, the Attraction Rings also have to support multiple timesteps. To help the user follow the answers of single persons, the circles are animated. During the movement of the person icons on the screen the circles change their size fluently to the next data value. This option can be turned off in the user interface as it is relatively resource demanding.

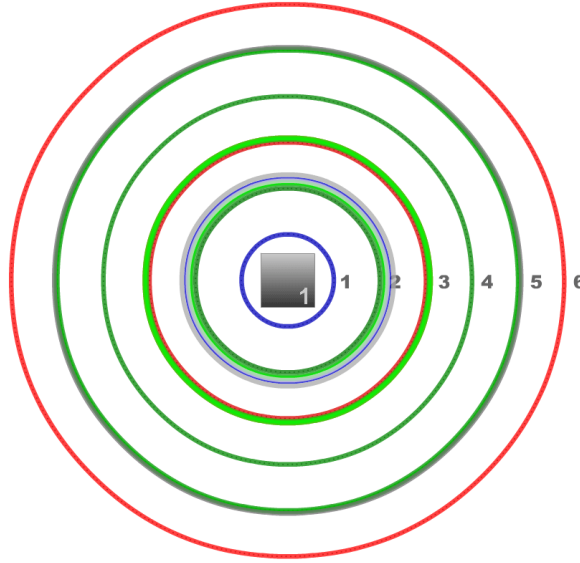


Figure 4.19: Attraction Rings are displayed around the corresponding question and give information on the exact answers that were given. If the rings are positioned next to each other, when having the same value, the distribution of discrete data sets can also be analyzed. In the above example we can see, that most patients answered this question with a “2”.

Interactions The Attraction Rings offer various interaction possibilities. Some interactions like add, remove and change of position are inherited from the question icon, where the Attraction Rings are anchored upon. In addition to this, they can be turned on and off at any time and a legend with dotted reference circles can be displayed to identify the value of each ring. On moving with the mouse cursor over a circle, a small tag with the corresponding name of the person and its value is shown next to the cursor (tool tip). Furthermore, clicking on any circle highlights the corresponding person. As soon as at least one person is highlighted only the selected ones are displayed (to reduce clutter). This makes it easy to compare two or more values. If a question is highlighted (using the overview visualization ListVis or the context menu) only the Attraction Rings of highlighted questions are shown (see Figure 4.13).

Benefits & Limitations The advantages of the Attraction Rings are clearly that they can easily visualize quantitative information of the answers for each question. The user is able to see the distribution of values for each question at a glance. This has proven to be helpful for finding questions that have not been answered in this timestep, group similar distributed questions, and find interesting questions. To help the user keep track of the data values of a single person (or more than one), the Attraction Rings offer the option to visualize the change between two timesteps with animation.

The biggest drawback of the method are the clutter when displaying a high amount of data. This limitation is countered by the possibilities to paint the rings next to each other in contrast to overlapping (works only with discrete data sets) or visualize the rings transparent. This way, the user can still discriminate a number of overlapping circles up to a given amount (depending on the adjusted alpha value). Still, visualizing the Attraction Rings for many questions and many persons leads to a lot of visual information in Gravi++ that may be distracting. Furthermore, when visualizing high values, the question rings get very big as they are currently not scaled (as our study data set has only values ranging from 0 to 6).

4.2.4.2 Star Plot (Glyph)

Another linked view to visualize the exact data values is the Star Plot (see Figure 4.20). It is a circular graph with radial axes based on a Star Graph (see Section 2.2.2.10). The axis are displayed from each question towards the center or away from the center (as defined in the resource file). The data values

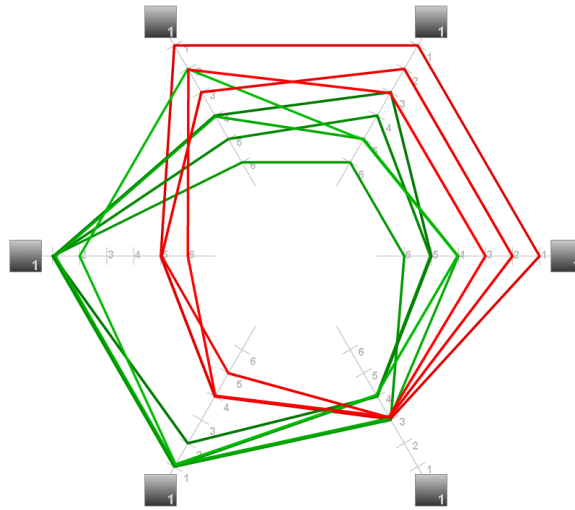


Figure 4.20: The above Star Plot displays two distinct clusters of red and green persons.

are plotted on the axes and connected with colored lines similar to the Parallel Coordinate Plot (see Section 2.2.2.2). The coloring is equal to the coloring of the Attraction Rings and question icons to help identifying the corresponding question. Like the Attraction Rings, the Star Plot also offers the options of transparency to deal with clutter and juxtaposed lines for items with equal values. Furthermore, animation is used to fluidly visualize the change over time. This helps the user in keeping track of the development over time.

Interactions Similar to the Attraction Rings the Star Plot and the axis legend can be turned on and off at any time and all interactions on the question icons are inherited. It also offers the same tool tip and highlighting functionality. For example highlighting some questions and persons in ListVis results in showing only the lines corresponding to the highlighted persons between the axis of the highlighted questions.

Benefits & Limitations As the Attraction Rings, the Star Plot offers similar benefits and limitations. It can easily visualize quantitative information for each question but as an advantage the exact data values of one person are displayed in a single shape and can be perceived more easily. Especially, comparing a number of (highlighted) Star Plot shapes shows differences much better than the Attraction Rings. This easy comparability is one of the main advantages of the Star Plot (also compared to other visualization techniques). For example, Lanzenberger et al. showed in [Lanzenberger et al., 2005] that Stardiates (can be compared to Star Plots) are more effective than Parallel Coordinates for the visualization of complex, multivariate data (e.g., psychotherapeutic data).

The biggest limitation is the resulting clutter when displaying a high amount of data. We countered this problem as described above with the option to use a specific degree of transparency or for special cases by juxtaposing the lines next to each other.

4.2.4.3 Traces

The Traces (see Figure 4.21) were inspired by the Phillips Curve [Tufté, 1990] and Dance Maps [Monmonier, 1990] and try to visualize the changing values over time with a single image. In other words, they map the time dimension on space and therefore enable the user to directly compare different points in time.

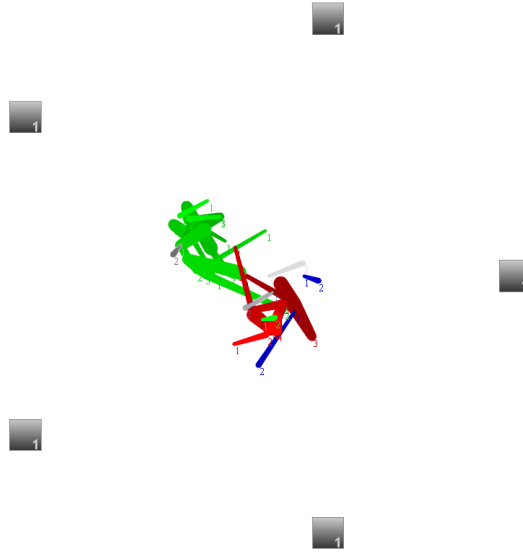


Figure 4.21: The Traces visualization displays the movement of the persons over time in a single image. The above image shows two clearly separated clusters where the Traces get bigger over time. This permits the visual analysis where the persons “move” over time.

The visualization displays the path of the persons over the course of time. As a metaphor, this could also be described as the footsteps of the person icons moving through time and space. Each position over time is connected by a line in the same color as the person icon is displayed. The thickness of the line can be mapped to different parameters (the same as on the question size). The option to map an increased size over time on the person icon and line thickness of the Traces is especially useful in this context. This enables the user to see the direction of movement in a single image.

The Traces lines can either reach only to the previous and next timestep (from the currently selected / viewed one) or to all timesteps. This second option is called “Whole Traces” and furthermore displays the corresponding timesteps as numbers next to the position on the line. As the Attraction Rings and Star Plot, the Traces are also visualized transparently (configurable in the properties file) to help recognizing overlapping segments and reduce clutter.

Interactions The Traces are directly embedded in the spring-based core visualization (displayed beneath the question and person icons) and are linked to the other views. They support highlighting a person by clicking on the specific line and display a tooltip with the name of the person when the user places the mouse cursor over a line. In addition to this, the user can interactively choose between displaying the “Whole Traces” (traces over all timesteps) or just the normal Traces that display lines to the next and previous point. Finally, the thickness of the traces is taken from the settings of the person icons.

Benefits & Limitations The major advantage of the Traces is that they can visualize the change over time with a single image. This helps the user to identify clusters and trends over time without moving through the time watching the animated icons. It seamlessly integrates with the spring-based core and enables the user to compare different persons and their development over time.

Despite the use of transparency, a major drawback of the Traces is the possible high clutter when visualizing many timesteps or a lot of persons. The visualization also bequeathed one problem of the spring-based core visualization. Two Traces may overlap on the screen but have different data values. This can happen because of the dimension reduction with the spring-based algorithm. Gravi++ tries to overcome this limitation by optionally mapping the strength, that is currently influencing the icons, on the thickness of the lines (equal of mapping the strength to the person icon’s size).

Future Work A possible enhancement of the Traces could be to “rounden” them. For example, the different points could be connected with Bezier or B-Spline curves leading to a smoother transition. The drawback of these techniques is, that the smooth transitions could visually suggest other data values than straight connecting lines.

4.2.5 Overview

As described by Shneiderman in his Visual Information Seeking Mantra “Overview first, zoom and filter, then details-on-demand.” [Shneiderman, 1996] the overview should be the first step in presenting the data set to the user. Craft and Cairns analyzed the mantra more in detail and argued “Why overview first?”:

“Overview provides a general context for understanding the data set; it paints a “picture” of the whole data entity that the information visualization represents. Patterns and themes in the data that may be helpful can often be seen only from a vantage point that comprises the whole view. From this perspective, major components and their relationships to one another are made evident. Simply the overall shape of the data itself can provide assistance in understanding the information that is encoded. Significant features can be discerned and selected for further examination. Such features might not be readily viewable from another part of the data representation or might be obscured from certain vantage points. Revealing these features at the outset can aid the user in filtering the extraneous information so that they can complete their task more efficiently by excluding unimportant aspects of the representation.” [Craft and Cairns, 2005]

To help the user get an overview on the data, see the general context, and see patterns and themes in the data we provide two different overview visualizations. Both of them visualize the whole data set and help the user in finding a specific subset, which then can be analyzed with the spring-based core visualization and the other views. The first overview is called ListVis, which is a simple list of all persons and questions in the data set. Its main aim is to let the user select a specific subset. The second overview is called TableVis and provides a visual overview of the data pattern. Its main aim is to help the user to find an interesting subset. Therefore, both overview visualizations complement each other as it is rather tedious to select a specific data set in TableVis and ListVis does not give any hints on the structure of the data.

4.2.5.1 ListVis

ListVis (see Figure 4.22) is one of the two overview visualizations that fully supports linking and brushing (see section 2.1.3.2). It consists of a scrollable list of all questions and persons that can be added to the spring-based core visualization. Its main aim is to provide a simple method for adding, removing, and highlighting a known subset (or all of the data) in the core visualization and additional views. For this cause ListVis offers a set of interactions.

- **Drag & Drop:** Adding or removing single items (questions or persons) can be done by simply dragging the visual item to the background (add item to the core visualization) or dragging the person or question icon anywhere on the ListVis windows (remove from the core visualization).
- Clicking on a question or person in ListVis highlights the corresponding data in each other visualization (core, Star Plot, Traces).
- Two dedicated buttons enable the user to add all persons or questions to the core with a single click.
- A context menu where all of the above interactions can be chosen is also available after a right click with the mouse anywhere on the ListVis window. In addition to this, an “add all” feature is offered only in the context menu to add all persons and questions at a time with a single click.

ListVis is implemented in a draggable window that can be positioned anywhere on the screen. Furthermore, it can be closed (hidden) to see only the core visualization.

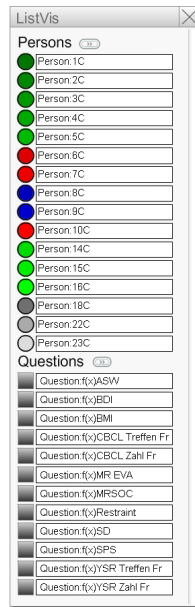


Figure 4.22: Screenshot of the ListVis overview visualization that shows a list of all persons and questions and offers a set of interactions like linking and brushing with the other visualizations.

4.2.5.2 TableVis

TableVis was developed by Christian Ritzinger for Gravi++. The main aim of this visualization is to provide an overview of the whole data set at a glance in order to find interesting patterns, clusters, questions and persons that she or he wants to analyze in detail with the spring-based core visualization.

For this purpose, TableVis visualizes highly structured, temporal data using small Bar Graphs [Harris, 1999] in a tabular structure (see Figure 4.23). These small juxtaposed plots are also called small multiples [Tufte, 1983]. Each bar visualizes one data value of a given person in relation to a given question (data value on y-axis). The x-axis can be mapped to any of the three dimensions (time, persons, or questions). Missing values are mapped to gray bars below the x-axis and values greater than six are painted in blue (with the height of six). This is done to visually mark data values out of the application’s scope. Values over six are currently marked as out of the scope as TableVis was specifically designed for our medical data set. The remaining two dimensions are mapped to the tabular structure where each entry is one Bar Graph (similar to a Scatterplot matrix).

To scroll the matrix and provide Focus+Context, TableVis contains a pixel-based overview visualization similar to the Matrix Visualization [Chittaro et al., 2002], that visualizes each Bar Graph as a gray shaded pixel. Charts with high values are displayed as a dark pixel, whereas charts with low values are shown as a bright pixel. The main aim of this overview scroll bar is to give the user the possibility to discover new patterns in the data and easily navigate to them.

To get details on demand, TableVis offers two enlarged Bar Graphs. The left one visualizes the currently selected cell of the matrix, whereas the other one shows the cell currently located below the mouse cursor. Because of the juxtaposition of the two charts the user can easily compare the different values and shapes. In Figure 4.24 you can see all three main elements of the TableVis visualization.

TableVis provides a set of interactions supporting the user to explore the data and to select an interesting subset for *Gravi++*:

- The rows and columns can be ordered freely using drag & drop. Dragging an item to a different position moves the column and row to the new position.
- The x- and y-axis of the matrix can be swapped.
- The parameter space can be restricted by the user.
- Displaying an enlarged version of the current Bar Chart where the mouse cursor is over.

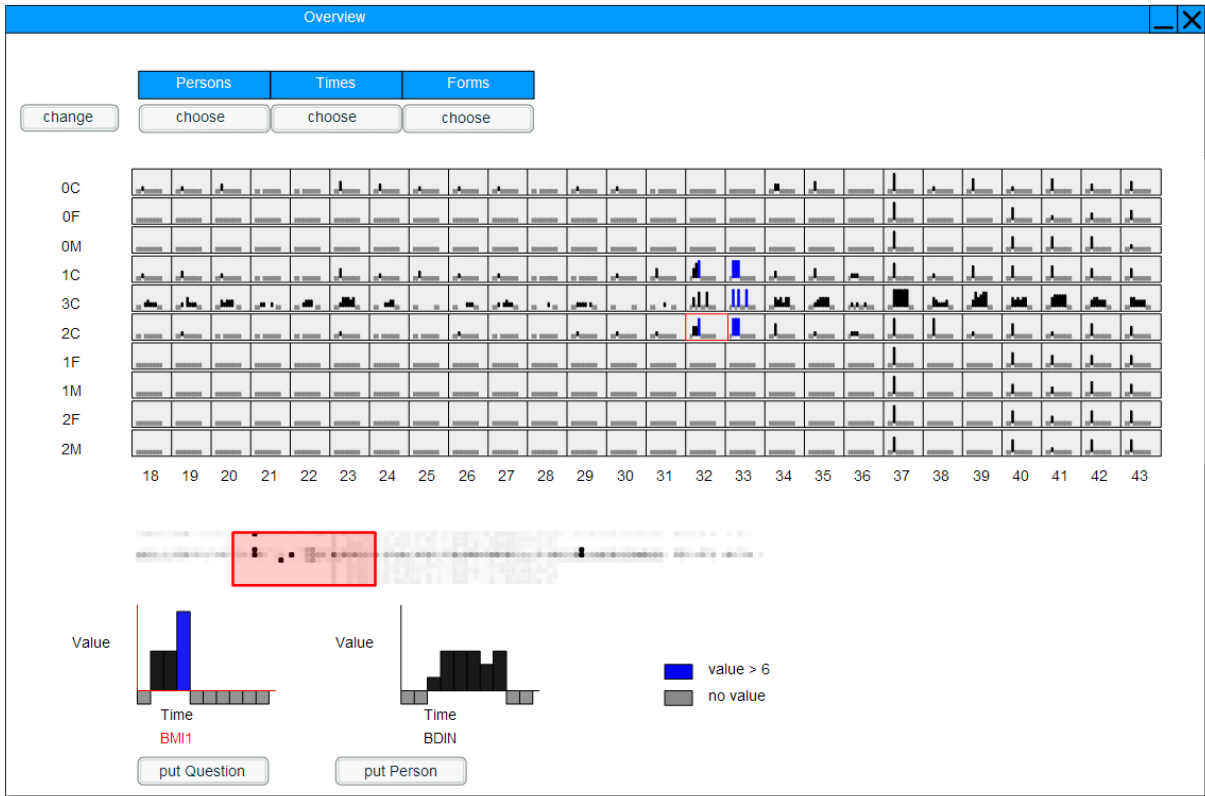


Figure 4.23: The overview visualization TableVis.

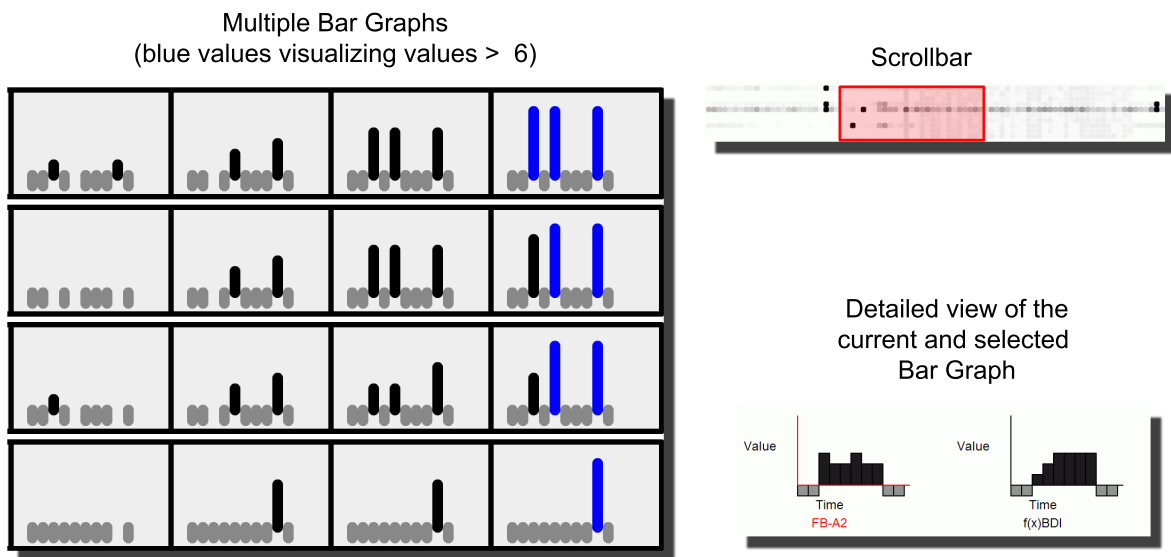


Figure 4.24: The three main visualization elements of TableVis.

- By clicking on a small multiple it is selected and visualized below as the second enlarged version.
- Adding the person or question of the currently selected bar chart to the spring-based core visualization. This can be achieved by two buttons. Dragging a small multiple to the spring based core to add the person and question is a future task.

4.2.6 Missing Data

Missing data is handled by Gravi++ in various ways as it is an important characteristic of “real data sets”. Especially, our medical data suffers from the phenomenon of missing data. The patients often do not fill out all questions of a questionnaire or even all questions of a timestep (e.g., because of an abort of the therapy) are missing. This characteristic has to be handled in a special way because especially in the spring-based core visualization missing data if not labeled correctly can lead to wrong insights. Questions that were not answered have no attraction on the person icons and are, therefore, dragged to another position than persons that answered the corresponding question. Therefore, the user could classify the person wrongly if it is not visually marked to be based on missing data. Gravi++ handles missing data in various ways:

- Persons that have no data in the current timestep are displayed fully transparent and can be dragged around freely because no force is attracting them.
- If not all data values are missing, the person icon is visualized with a red triangle in the upper red corner. A possible extension could be to place this triangle in the direction of the question with the missing data (and broaden it when more data is missing).
- If there is no data for the mapped size attribute of the person icon at the current timestep, a red dot displayed in the middle of the icon.
- The additional views Traces and Attraction Rings do not display any visual element if all data values are missing for a person. A possible improvement could be to visualize only partially missing data (for example only one question was not filled out) by another visual style (for example with dotted Traces).
- The Star Plot visualizes missing values by painting the corresponding line to the center of the Star Plot. This way it can be differentiated from answers with the value “0” and in addition to this, single axis with adjacent axis without any data values can still be identified.
- The overview TableVis does visualize missing values with gray bars that are painted below the x-axis symbolizing the value “-1”. This way the user can easily identify missing values even in the small multiples.

Figure 4.25 shows the five different visual representations of person icons regarding missing data.

In addition to this, the user has the interaction possibility to freeze the values of a selected question. This way, the user is able to propagate the attraction of a question through time, which makes sense, if this question is only asked once (but valid over time) or the questionnaire was not asked at a specific timestep.

4.3 View Coupling

The three additional views Traces, Attraction Rings, and Star Plot are tightly coupled with the spring-based core visualization and the overview visualization ListVis. The color scheme used for classifying the person icons is used in all views to help the user identify and relate the persons. This helps identifying interesting persons (for example outliers in the spring-based core visualization) in the other views (see Figure 4.26).

Linking and Brushing (see Section 2.1.3.2) is supported by all views. Highlighted data points are emphasized in all views (linking) and can be highlighted interactively in each view (brushing). If at least one person is highlighted, Gravi++ visualizes only the data values of this person in each view. The

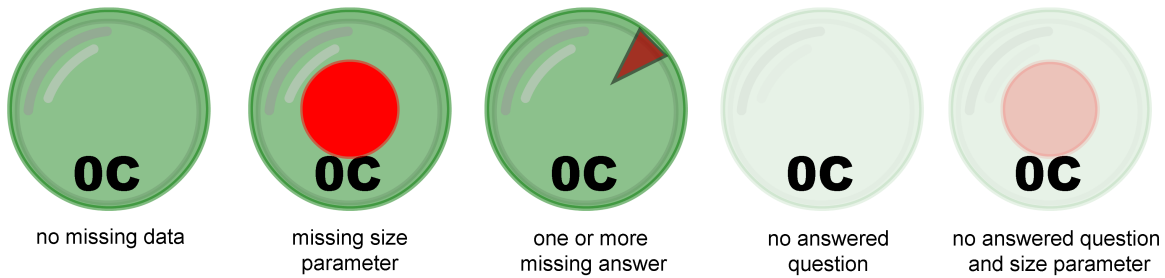


Figure 4.25: Different graphical representations of missing data in the spring-based core visualization.

highlighting can be done by clicking on the corresponding graphical object in the additional views, by using the context menu in the spring-based core visualization, or in the overview ListVis.

In addition to Linking and Brushing, Gravi++ also has a linked Time Control, that changes the timestep in all views at the same time. As the additional views are integrated in the spring-based core, interacting on the core visualization (e.g., changing the position of a question icon) is directly connected with the visualization of the additional views. For example, during dragging a question icon on the screen, the user can instantly see what impact the current position has on person icons and on the Traces, Star Plot and Attraction Rings.

4.4 User Requirements

Gravi++ was specifically designed for our medical problem and for the medical expert users who need to analyze the data. The clinicians main aim was to locate predictors in the data. These predictors should give an idea about a specific status of a patient and are used to predict the further development of the therapy. Especially, the first two timesteps (before the therapy and after three months of therapy) are important for the detection of predictors. Gravi++ does meet this requirement with the Traces view. Especially, the Traces to the next and previous timestep can visualize the development in the first timesteps. In addition to this, the animation and the possibility to move back and forth through time (using the time control window) helps analyzing the progress in specific time spans.

Another important user requirement was the possibility to explore the data. Therefore, we provide various views and visualizations to help the user find an appropriate view and generate answers and hypothesis on the data. Especially, the various interaction methods (see Section 4.2.3) help the user in the exploration process and let them “play” with the data set. In this context, we also emphasized the development on a visually appealing user interface and data visualization to keep the users interested in interacting with the data set.

The possibility of finding clusters and outliers of persons over time was also an important facet in our medical problem. Our spring-based core visualization in combination with the other views is especially good at visualizing clusters and outliers in multidimensional data. Clusters are distinct groupings in the data points which are displayed as visual groupings of person icons in the spring-based core. Outliers are one-off cases with usually unusual combinations of observed values (in respect to the other data points). They should be displayed in the spring-based core as single person icons with some distinct distance to the other items and can, therefore, visually be recognized easily.

One requirement of our in2vis project was to develop an interactive Information Visualization with the main aim on the visual discovery rather than the computational one. Therefore, we did not include any machine learning techniques in Gravi++. This way, we make use of the background knowledge of our expert users and do not “guide their route through the data”.

Finally, the detailed analysis of the exact values was one important requirement. For this task, we integrated tooltips and the additional views Attraction Rings and Star Plot that give detailed information on all the answers that were given in the currently selected timestep.

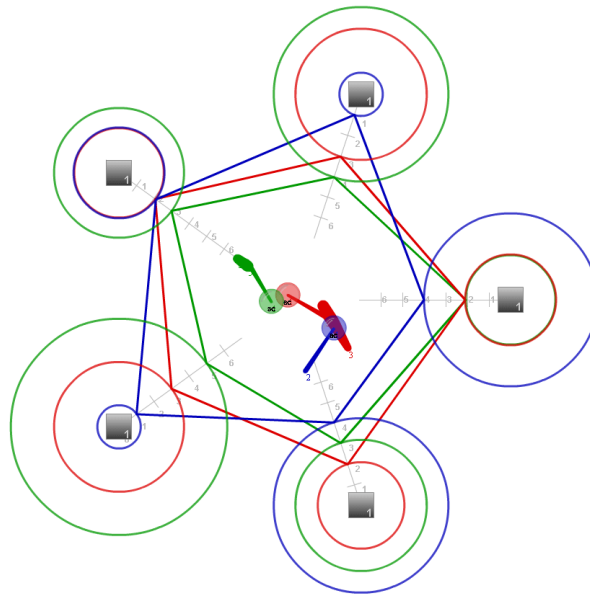


Figure 4.26: All three views are coupled with the spring-based core by using the same colors for visualizing the answers on the questions. Therefore, the user can easily relate the information in each view. Above image shows the spring based core with enabled views Traces, Attraction Rings, and Star Plot.

4.5 Benefits & Limitations

Gravi++ with the spring-based core visualization, the additional views and the two overviews offers a number of benefits and features. Still there are also some shortcomings of our approach. We will now discuss both benefits and limitations in detail.

4.5.1 Benefits & Features

From a cognitive perspective the visual elements of Gravi++ have several advantages [Card et al., 1999]. The Gestalt Laws “Proximity”, “Similarity”, and “Symmetry” (see Section 2.1.1) apply to our visual presentation for example. “Proximity” states that graphical objects that are close together are perceived as a group. This principle is essential for perceiving clusters of person icons, which is one main principle of Gravi++. Because, similar elements are perceived as grouped (principle of “Similarity”), the similar person icons and the similar question icons are perceived as two different groups which is essential for analyzing the visual presentation of the core visualization. As symmetrical figures tend to be grouped to an object, the symmetrical figures are perceived differently (principle of “Symmetry”). This fact can be useful when analyzing a Star Plot in Gravi++ where (depending on the arrangement of the question icons) a symmetrical figure stands for a special data constellation. Therefore, the user should arrange question icons with an inherent relation on opposite circular sides to make use of this phenomenon.

In addition to this, Gravi++ uses the twelve distinct colors Colin Ware proposes in his book “Information Visualization” [Ware, 2000] for color coding of different person icons.

“These colors have widely agreed-upon category names and are reasonably far apart in color space.” [Ware, 2000]

Therefore, the colors should be easily differentiable also in respect to color blind people (except for the red – green combination for our study).

The spring-based core visualization has the advantage of having the capability to visualize many dimensions at a single glance. Especially, finding clusters and outliers in coherent dimensions is the core feature of this dimension reduction method. Compared to mathematical dimension reduction techniques like Principal Component Analysis (PCA) (see Section 2.2.1.3), our visualization has the advantage that the user can relate the position of the person icon directly to the underlying data. The closer a person is

to a specific question, the higher it was answered (in respect to the other questions). This fact, can be very helpful when analyzing data similar to our medical data set.

To visualize time, the core visualization uses animation which is probably the most natural form of graphically mapping the time dimension. The Traces map the time dimension to space and, therefore, Gravi++ offers both techniques for visualizing time which should combine the advantages of both methods.

The multiple coordinated visualizations provide different views on the data with different levels of detail and emphases. This way the limitations of a single visualizations are overcome and the user is able to choose his or her favorite view to explore the data thoroughly. In Gravi++ the Traces overcome the limitation of the spring-based core to display only one timestep at a time. In addition to this, the Star Plot and Attraction Rings visualize the exact values, which are also hard to detect in the core and Traces visualization.

The interactive manipulation methods on the coordinated multiple views can help the user get new insights through the data. This can be used to formulate hypotheses on the data e.g., to find a new predictor.

4.5.2 Shortcomings

Nevertheless, some problems and shortcomings of Gravi++ are not solved yet. The spring-based core visualization suffers from incomplete data that can lead to incomparable person icons positions. This is countered by various options in Gravi++ (see Section 4.2.6) like visually emphasizing this data or supplying interaction methods and views to help the user to cope with this phenomenon.

An important drawback of our visualization is, that is not designed for huge amounts of data. Too many questions and persons lead to clutter, although we offer transparency, highlighting, and two overview visualizations to cope with overlapping data points and lines. If the user wants to analyze a high data space, further methods of data aggregation, dimension reduction, and filtering would be necessary.

4.6 Future Work

There are some open points and ideas that could enhance Gravi++.

A possible extension to Gravi++ could be an algorithm that automatically positions the questions to find clusters of persons. If the persons are classified, the arrangement could be automatically optimized to form clusters according to this classification. Demsar et al. proposed an algorithm in [Demsar et al., 2005] that could be extended for our temporal data set. Another approach could be to look at the data distribution of each question (over time) and arrange the icons of similar distributed questions next to each other. Both algorithms should be optionally to still make use of the expert users background knowledge and let the user explore the data. In our insight study we want to find strengths and limitations of a visualization, supervised Machine Learning techniques, and Explorative Data Analysis techniques to estimate how combinations of these methods can contribute to more in-depth reasoning processes. The combination of Gravi++ with an automatic positioning algorithm would lead to a mixture of a visualization with Machine Learning techniques and, therefore, could not be used in our study which is a main goal of our project.

Another possibly helpful extension of Gravi++ would be to integrate a technique to analyze the data frequencies. Especially, when analyzing a high number of categorical data, the Star Plot and Attraction Ring views are of limited use. Either all lines are overplotted on the view positions where the categories are mapped or when the option is enabled, that they are painted next to each other the distance between two categories is not sufficient. Even combining the use of transparent color and highlighting has limitations for huge data sets. A possible solution for this problem could be to extend the Star Plot with the Parallel Sets technique [Kosara et al., 2006]. This is a combination of a Mosaic Plot and Parallel Coordinates. The data points on the axis are extended to sections, that are sized according to the amount of data in this section. In addition to this, a Histogram can be shown for each section. The drawback of integrating this method would be an increased complexity.

In this context, the integration of other statistical methods (in addition to Histograms) as additional views or in a dedicated window could also help to get a better overview on the data set. This was also one

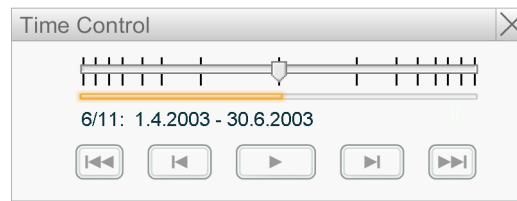


Figure 4.27: Possible implementation of a Time Control with Fisheye view, where it is enhanced with a Focus+Context visualization with the focus on the currently visualized timestep.

suggestion of the focus groups in our insight study (see Chapter 7). For example a Box Plot, Scatterplot, Histogram and additional descriptive values could prove to be useful.

In addition to a view with statistic values and plots, further InfoVis techniques could be integrated in views to give the user further alternatives to analyze the data. For example, a Fisheye view in the timeline for visualizing a lot of timesteps would make sense. This means, the timeline would be enhanced with a Focus+Context visualization with the focus on the currently visualized timestep. The rest of the timesteps would not be visualized with equal distances (or corresponding to the difference in time) but as the “context” getting smaller (or vanishing) the further the timestep is away from the currently visualized. A possible example is illustrated in Figure 4.27.

To visualize an additional dimension that for example informs about general trends in the whole data set, a subset of the color space could be used to color the background of the spring based core. For example in our medical data set the accumulated weight of all persons of the visualization could be visualized using the background color (with light colors). Therefore, the user could see overall trends easily.

A feature that was not implemented because of performance considerations with our Flash prototype was the “undo” function. It was the most wanted option in the usability assessment. Still, using the load and save function the user can manually specify points where he or she can return.

A small problem that we found using Gravi++ was during analyzing a high amount of data. The tooltips of question and person icons get very large. Placing the labels and tooltips at appropriate positions on the screen is a challenging problem because of the amount of information they contain. Gravi++ uses a simple algorithm that decides if the tooltip is displayed above or below the corresponding icon (depending the size of the tooltip and the position of the icon). This algorithm still leaves plenty of room for improvements.

According to the Gestalt Law “Continuity” (see Section 2.1.1) smooth and continuous visual elements are more likely to be perceived as an entity than ones that contain abrupt changes in direction. Currently our additional view, the Traces, contain abrupt changes in direction from one timestep to next. Therefore, the Traces could be enhanced by implementing continuous changes with B-Splines for example. The problem of this approach would be, that the paths of the Traces would not correspond to the movement of the person icons (this could also be changed to continuous changes) and that the persons and Traces could move nearer to question icons than the attraction force would permit (to enable the continuous changes). Because of this drawback, we chose to stick with the abrupt changes in direction.

A further improvement of the Traces could be to map the progress through time to different colors (or different shades of the used colors for the classification of the persons). This way, the movement through time may be seen with a single glimpse. For example, the Traces could have an increasing hue over time. This principle could also be applied for the Star Plot to display the Star Plots of all timesteps in a single image (and still being able to discriminate them).

As one of Gravi++’s main emphasis lies on interactions, integrating additional interaction techniques would be an important part of future work to enhance Gravi++. For example enhanced interactions on the timeline to highlight only a specific timespan or add and remove timesteps interactively could be added. Furthermore, advanced filtering methods could be integrated in a specific window to offer additional Focus+Context methods.

Finally, a solution to cope with the problem of overlapping person icons could be to integrate a similar technique like “liquid browsing” introduced in [Waldeck, 2005]. This technique works similar like

a Fisheye view on the data as it lets the data items float apart when the mouse cursor is over them. This means, overlapping icons would float apart when the mouse cursor is over them and could so be still differentiated.

With these additional techniques and ideas Gravi++ could be further improved to help the user get new insights in the data, which is one of its main goals.

In the next Chapter we will present some details on the implementation of our Gravi++ prototype.

Chapter 5

Implementation

Gravi++ including all views and the two overview visualizations were implemented in Macromedia Flash 7 and during the development ported to Flash 8.

“Flash® Professional 8 is the industry’s most advanced authoring environment for creating interactive websites, digital experiences and mobile content.” [Macromedia, 2006b]

As Macromedia states, Flash is a rapid development environment for interactive digital content. We first planned to use Flash only for early prototypes of our Information Visualization concept. But due to the relative powerful object-oriented ActionScript language behind Flash, the development of the early prototype did advance very well and a reimplementaion in e.g., Java did pose too less advantages in contrast for the time needed. Therefore, we implemented the whole Gravi++ system in Flash and noticed the following advantages:

Rapid visual prototyping and integration. Due to the integrated vector-oriented graphical editor and to some extend the possibility for visual programming, Flash enables the programmer to rapidly prototype and integrate new ideas. Changing symbols, icons, or user interface elements can be done with a minimum effort. Therefore, we were able to, for example, try out different metaphors fast and easily and discuss them with our project team.

Good availability of the Flash Player. The widespread distribution of the Flash Player (98% of desktops according to Macromedia’s web page [Macromedia, 2006a]) and the availability for all major operating systems (Windows, Linux, MacOS) were two major advantages of Flash (especially compared to the Macromedia Director authoring environment, that was e.g., not available for Linux at the time of the evaluation).

Possibility to integrate the Flash application in a web page. As Macromedia Flash was primarily intended to be integrated in a web page, the applications work without any problems embedded in HTML pages presented on the web.

State of the art graphical representation. The vector-based graphical rendering engine of Flash provides an appealing visual representation including anti-aliasing and integrated zooming functionality.

Easy integration of XML based data files because of the integrated data functions. Flash provides a threaded XML object, that enables the programmer to load and parse any XML data file with only a few lines of code.

All these advantages helped us to develop Gravi++ from “knowing which kind of data we want to analyze” to the prototype that was used in our evaluation. Especially, the constant changes and try-outs during the development could be done relatively easy and fast using Flash. Nevertheless, during the development we also noticed some major disadvantages:

Performance One of the major problems of Flash that we encountered during the development of Gravi++ was the slow performance when visualizing a high number of graphical objects (especially with Flash 7) . With the introduction of Flash 8 this got much better (the `cacheAsBitmap` method of the `MovieClip` object speeds up the visualization of many objects significantly) but still handling a large amount of data leads to a huge memory consumption and slow speed. One of the major problems in conjunction with this, is that the Flash plugin stops the execution and displays a warning message, if the script calculates for a given amount of time and doesn't change any frame. This means, implementing loops with long run times is problematic, as the user has to click away the warning message (script may have hung up warning) several times. We have countered this problem by distributing large loops over several frames (first frame initialization, second frame do things, third frame termination and incrementation of the counter).

Write to disk restriction Due to security reasons (Flash was mainly intended for web applications), applications written in Flash are not allowed to alter files on the computer they run on. This means no data files or configuration variables can be written on the harddisk. An alternative for this is the `SharedObject` in Flash. This object allows to write variables on the users computer like a web cookie. This way Gravi++ handles saving the log files and saved states. The actual information is stored in a `.SOL` file in a special directory on the user's computer. Using the flash configuration tool, the user can decline Flash cookies by domain as well as control the amount of data a site is allowed to store. By default, sites are permitted to store 100kB of information without prompting a user. On a computer with a German version of Windows XP the files are located in the following directory:

```
C:\Dokumente und Einstellungen\[UserName]\Anwendungsdaten\Macromedia\FIash Player\#Shared0bjects
```

No advanced IDE and debugging functionality Macromedia Flash Professional IDE is mainly focused on web development and animation support. Therefore, it does not deliver advanced functions for larger projects. Especially, the debugging possibilities are very limited which costs time during the development.

Still the overall impression during the development of Gravi++ was positive and especially now with the introduction of Flex2 (professional successor with Eclipse IDE support) the development of a visualization prototype with visual development environments like Flash or Flex2 can be a good choice.

5.1 System Architecture

Gravi++'s design architecture is based on two different principles. Our visualization was implemented in Macromedia Flash, which uses a timeline with changing frames that can be assigned code. Assigning code to the individual frames in the timeline is the first principle of Gravi++. The different states during the execution of the program are mapped on different frames. This can be seen in the Flow Chart in Figure 5.1. This principle of embedding code in frames, is also used to simulate loops (loops are distributed on three frames). This is done, because during execution of code fragments there is no possibility to update the screen. In addition to this, Flash brings up a warning if the execution of code in one frame exceeds a short time limit.

The second principle is the object-oriented architecture. Gravi++ uses separated class files that are instantiated and called in the code on the timeline. Each view, the overview visualizations and important parts of the core visualization are implemented in separated class files.

Therefore, the prototype is a mixture of an object-oriented architecture and a timeline oriented state machine. Detailed information on the implementation details of Gravi++ (like a Doxygen documentation) can be found in the "doc" directory of the Gravi++ prototype.

5.2 Metaphor

As defined by Marcus [Marcus, 2002] a metaphor helps the users to understand, remember, and enjoy the entities and relationships of computer-based communication systems. Therefore, we decided to employ a

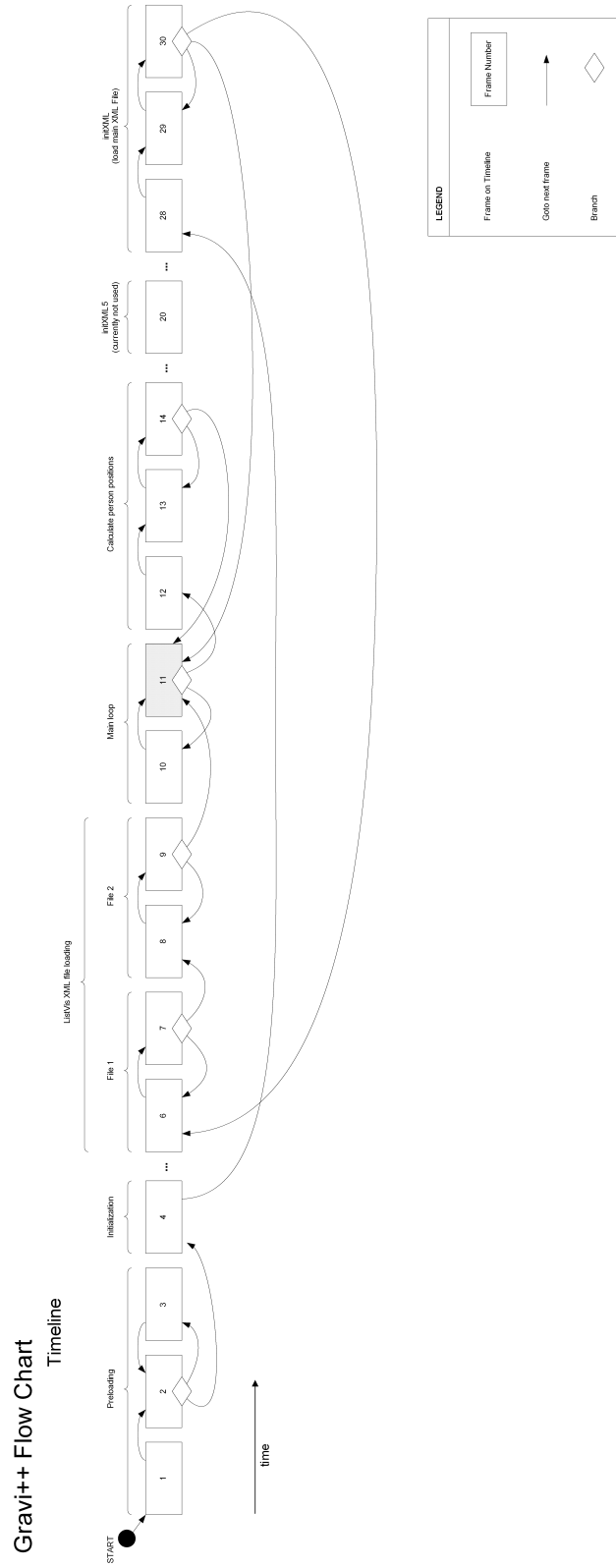


Figure 5.1: Flow Chart of the different frames in Gravi++ and how they are connected in the sequence of events.

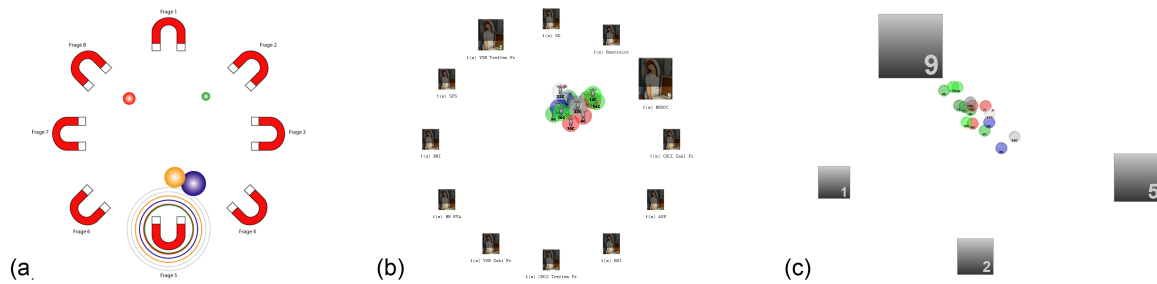


Figure 5.2: Three different metaphors that were introduced during the development of the Gravi++ prototype. Figure (a) shows the magnetic metaphor that was discarded because of interpretation problems with our medical data set. Figure (b) shows the implemented image gallery metaphor, where questions are represented by paintings in an art gallery and the persons are moving to the images they like best. Figure (c) shows the neutral version where simple spheres and rectangles are used to represent persons and questions.

visual metaphor to communicate the concepts that lie behind the spring-based core visualization. During the development of Gravi++ different metaphors were created and discussed with our project partners.

We started with a magnet metaphor (Figure 5.2(a)) for the spring-based core. The questions are visualized as magnets that attract iron balls that represent the persons. The metaphor does only partly match our physical spring model, as the magnetic attraction force gets smaller over the distance in contrast to a spring. But still it communicates the principle that the persons are attracted by the questions. In addition to this, the Attraction Rings can be interpreted as the magnetic fields. Furthermore, a gravity metaphor was also introduced as it provided a similar information on the attraction as the magnetic metaphor. Out of the conjunction of these two metaphors the name Gravi++ emerged. Gravi stands for gravitation and the two summation signs stand for the magnetic attraction.

As these metaphors were not appreciated by our therapists because of the interpretation of mechanical attraction which seemed to deterministic, we started a discussion to find a new metaphor where the persons are not mechanically attracted by the questions but move there more in a sense of free will. Out of this process the metaphor “image gallery” emerged (Figure 5.2 (b)). The questions are represented by paintings in an art gallery. The persons are displayed as person icons that move to the image (question) they like best.

In addition to this image gallery metaphor, we also implemented a neutral version where the persons are represented by simple spheres and the questions by shaded rectangles (Figure 5.2(c)). This clean and simple style lead to less clutter especially when visualizing a large amount of data (and using small icons).

Our discussions with the medical experts revealed the problem that some metaphors can interfere with the interpretation of the data. Therefore, it is important to consult experts of the problem domain to find a suitable representation of the data.

5.3 Data Format

To speed up the loading times and reduce the memory consumption, we decided to use transformed data files based on the conflation data set file described in Chapter 4, Section 4.1. This file is transformed by three different XSL scripts called

```
result-qpt.xml
result-qtp.xml
result-tpq.xml
```

The last file result-tpq is the main data input file used for Gravi++. The other two files are used for the TableVis overview visualization. The difference of the three files compared to the conflation_dataset.xml

is the different tree-like structure of the files. The result-qpt.xml file for example is ordered by questions, then persons, and then timesteps - therefore the “qpt”. The other files follow the same principle. To convert the conflation_dataset.xml, the user can use the following script calls:

```
java -jar saxon8.jar -o result-qpt.xml conflation_dataset.xml sort-qpt.xsl
java -jar saxon8.jar -o result-qtp.xml conflation_dataset.xml sort-qtp.xsl
java -jar saxon8.jar -o result-tpq.xml conflation_dataset.xml sort-tpq.xsl
```

Windows users can use the batch file “convert_data.bat” to convert the data file conflation_dataset.xml to the three single XML files.

5.4 Property File

The property file with the name “resources.xml” contains all options that cannot be changed in the user interface. In addition to this, all labels in Gravi++ are read from this file (to ease the translation of Gravi++). Each property can be found as the attribute “name” in the “prop” tag. For example, the property “test” with the value “testvalue” would be encoded in the following line:

```
<prop name="test" value="testvalue"/>
```

See Tables 5.1, 5.2, 5.3, 5.4, and 5.5 for a list of all possible attributes and values and the description of each value.

attribute	Possible values	Description
logging	true, false	This option enables the logging of all user interactions.
user	username	Sets the user name that is used for the logging.
inverted_center	Integer, e.g., 3	Number that should stay the same, when the values are inverted, eg., 3 ... , 3 -> 3, 4 -> 2, 2 -> 4, has to be $\geq \frac{\max}{2}$
metapher	sphere, default	Two possible visual metaphors can be chosen in this option. "sphere" enables the magnetic metaphor and any other string enables the neutral metaphor with rectangles and spheres.
loadChristianVis	true, false	Load the overview visualization TableVis (formerly known as ChristianVis). Without this option enabled, TableVis cannot be displayed in Gravi++.
general_cm_removepersons	"Remove Persons"	Label for the context menu on the background of Gravi++ to remove all person icons.
general_cm_removequestions	"Remove Questions"	Label of "Remove Questions" context menu entry on the background of Gravi++.
general_cm_removeall	"Remove Everything"	Label of the "remove all questions and persons" function context menu entry on the background of Gravi++.
general_cm_showgrid	"Show Grid"	Label of the function "show the grid on the background" context menu entry on the background of Gravi++.
general_cm_hidegrid	"Hide Grid"	Label of the function "hide the grid on the background" context menu entry on the background of Gravi++.
general_cm_resetwindows	"Reset Window Positions"	Label of the function "reset window position" context menu entry on the background of Gravi++.
general_cm_zoomin	"Semantic Zoom In"	Label of the function "semantic zoom in" context menu entry on the background of Gravi++.
general_cm_zoomout	"Semantic Zoom Out"	Label of the function "semantic zoom out" context menu entry on the background of Gravi++.
general_cm_random	"Random Question Position"	Label of the function "random question icon position" context menu entry on the background of Gravi++.
general_xsltransformedfiles	true	The current version of Gravi++ does only support the xsl transformed data files.
general_xsltransformeddatafile	"result-tpq.xml"	Filename of the xsl transformed data input file with the structure time - person -question.
general_tablevis_qtp_filename	"result-qtp.xml"	Filename of the xsl transformed data input file for TableVis with the structure question - time - person.
general_tablevis_qpt_filename	"result-qpt.xml"	Filename of the xsl transformed data input file for TableVis with the structure question - person - time.
general_experimentfilename	"experiment.xml"	The experiment XML file from LinkVis is not used anymore. Therefore, this option is not needed.
general_parameter_master_data_filename	"parameter_master_data.xml"	Filename of the parameter_master_data file with information about the questionnaires.

Table 5.1: General attributes of the Gravi++ property file, part one.

attribute	Possible values	Description
general_fulllivepreview	true, false	This option controls if the live preview is turned on or off for Traces and Star Plot views. Should be disabled when visualizing a high amount of data because of speed issues.
questionrings_distance	Integer, e.g., 30	Distance from the first question ring to the question center (default: 30).
questionrings_mfdistance	Integer, e.g., 30	Distance between the Attraction Rings / Star Plot lines (recommended: 30, otherwise labels are not fitting).
questionrings_thickness	Integer, e.g., 3	Thickness in pixel of the circles of the Attraction Rings view.
questionrings_alpha	0 - 100	Alpha value of the Attraction Rings (=transparency, possible values 0..100 (measured in %), 0 means no transparency).
mf_overlap	Integer, 0-n, e.g., 3	Distance between two overlapping Attraction Rings (recommended: 3 for thickness: 2, 0 for no overlapping).
general_show_tooltip	true, false	Show the tooltip on hover of question and person icons.
tooltip_fontsize	Integer, 0-n, e.g., 14	Size in points of the tooltip text (default: 14).
traces_alpha	0 - 100	Alpha value of the Traces lines (=transparency, possible values 0..100 (measured in %), 0 means no transparency).
starglyph_inverted	true, false	If this option is true the axis of the Star Plot point to the outside instead towards the center.
starglyph_legend	true, false	Display the legend / axis of the Star Plot.
starglyph_alpha	0 - 100	Alpha value of the Star Plot lines (=transparency, possible values 0..100 (measured in %), 0 means no transparency).
starglyph_thickness	Integer, 0-n, e.g., 3	Thickness in pixel of the Star Plot lines.
animate_sg_ar	true, false	If this option is true, the Star Plot lines and Attraction Rings are animated to fluently change the positions between two timesteps.
helptext	String	HTML Helptext of Gravi++ that is displayed in the help window.

Table 5.2: General attributes of the Gravi++ property file, part two.

Attribute	Possible values	Description
question_textlabel	String, e.g., "Question"	Question tag that is displayed in the tooltip and overview.
question_size	Real value 0-n, e.g., 100	Size in percent of the question icon (only 100 won't break the attraction ring labels).
question_show_label	true, false	Label containing the name of the question beneath the question icon.
question_nametag	true, false	Display the question name tag on the question icon.
question_cm_remove	String, e.g., Remove	Label for the question context menu to remove a question.
question_cm_highlight	String, e.g., Highlight	Label for the question context menu to highlight a question.
question_cm_unhighlight	String, e.g., Unhighlight	Label for the question context menu to unhighlight a question.
question_cm_complement	String, e.g., Add Complement Question	Label for the question context menu to add a complement question (question with complementary answers).
question_cm_strength0	String, e.g., "Strength:0"	Label for the question context menu to set the strength to 0.
question_cm_strength1	String, e.g., "Strength:1"	Label for the question context menu to set the strength to 1.
question_cm_strength2	String, e.g., "Strength:2"	Label for the question context menu to set the strength to 2.
question_cm_strength3	String, e.g., "Strength:3"	Label for the question context menu to set the strength to 3.
question_cm_strength4	String, e.g., "Strength:4"	Label for the question context menu to set the strength to 4.
question_cm_strength5	String, e.g., "Strength:5"	Label for the question context menu to set the strength to 5.
question_cm_strength6	String, e.g., "Strength:6"	Label for the question context menu to set the strength to 6.
question_cm_strength7	String, e.g., "Strength:7"	Label for the question context menu to set the strength to 7.
question_cm_strength8	String, e.g., "Strength:8"	Label for the question context menu to set the strength to 8.
question_cm_strength9	String, e.g., "Strength:9"	Label for the question context menu to set the strength to 9.
question_cm_strength10	String, e.g., "Strength:10"	Label for the question context menu to set the strength to 10.
question_cm_fixValue	String, e.g., fix / unfix	Label for the question context menu to fix / unfix the current values of the question over time (values do not change for this question when the timestep changes).
question_loadimages	true, false	Look for jpg images with the same name as the corresponding questionnaire that should be displayed on the question icon (if image metaphor is chosen).

Table 5.3: Question related attributes of the Gravi++ property file.

attribute	Possible values	Description
person_textlabel	String, e.g., "Person"	Person tag that is displayed in the tooltip and overview.
person_size	Real value 0-n, e.g., 100	Size in percent of the person icon.
bmi_timetosizelabel	String, e.g., "Map Size to Timestep"	Label of the mapping of time to the size of persons and questions (getting bigger over time).
bmi_strengthlabel	String, e.g., "Strength Indicator"	mapping of the strength from the questions to the size of the persons and traces
bmi_tracesthicknessbmi	Real, e.g., 1.3	Multiplier to map the additional dimension (e.g., BMI) on the thickness of Traces.
bmi_tracesthicknessbmi_min	Real, e.g., 12	Minimal value of the addition dimension (e.g., BMI) (used for the size of the person icons and Traces)
bmi_tracesthicknesstimetotime	Real, e.g., 1.5	Multiplier for mapping time to size on the Traces.
bmi_tracesthicknesstrength	Real, e.g., 1.0	Multiplier for mapping the strength on the Traces.
person_missing_data_alpha	0 - 100	Alpha value of the person icon on missing data (=transparency, possible values 0..100 (measured in %), 0 means no transparency).
person_cm_remove	String, e.g., "Remove"	Label for the person icon context menu to remove a person.
person_cm_highlight	String, e.g., "Highlight"	Label for the person icon context menu to highlight a person.
person_cm_unhighlight	String, e.g., "Unhighlight"	Label for the person icon context menu to unhighlight a person.
colors_array	List of all colors in hex values separated with ",", e.g., 0x007700, 0x8C993	List of all colors that are used for the person icons, Traces, Attraction Rings, and Star Plot.

Table 5.4: Person related attributes of the Gravi++ property file.

attribute	Possible values	Description
listvis_height	Integer, 0-n, e.g., 555	Height of the ListVis window in pixels.
listvis_heading_persons	String, e.g., "Persons"	Heading label of all persons.
listvis_heading_questions	String, e.g., "Questions"	Heading label of all questions.
listvis_cm_addquestions	String, e.g., "Add All Questions"	Label for the ListVis context menu to add all questions to the core.
listvis_cm_addpersons	String, e.g., "Add All Persons"	Label for the ListVis context menu to add all persons to the core.
listvis_cm_addall	String, e.g., "Add Everything"	Label for the ListVis context menu to add everything to the core.
listvis_cm_highlight_question	String, e.g., "Highlight Question"	Label for the ListVis context menu to highlight this questions.
listvis_cm_unhighlight_question	String, e.g., "Unhighlight Question"	Label for the ListVis context menu to unhighlight this questions.
listvis_cm_highlight_person	String, e.g., "Highlight Person"	Label for the ListVis context menu to highlight this person.
listvis_cm_unhighlight_person	String, e.g., "Unhighlight Person"	Label for the ListVis context menu to unhighlight this person.
listvis_cm_removepersonfromgravi	String, e.g., "Remove from Gravi++"	Label for the ListVis context menu to remove this person from the core.
listvis_cm_removequestionfromgravi	String, e.g., "Remove from Gravi++"	Label for the ListVis context menu to remove this question from the core.
listvis_cm_addquestiontogravi	String, e.g., "Add Question to Gravi++"	Label for the ListVis context menu to add this question to the core.
listvis_cm_addpersontogravi	String, e.g., "Add Person to Gravi++"	Label for the ListVis context menu to add this person to the core.

Table 5.5: ListVis related attributes of the Gravi++ property file.

The following code is an example for the current resources.xml file with the above listed attributes:

```
<?xml version="1.0" encoding="UTF-8"?>
<resources>
<!-- changeable properties of Gravi++ as options and labels
(e.g., for translating Gravi++ except the top menu which is in a separate file menu.xml) -->
  <properties>
    <!-- general -->
    <prop name="logging" value="false"/>
    <prop name="user" value="9625167"/>
    <prop name="inverted_center" value="3"/>
    <prop name="metapher" value="sphere"/>
    <prop name="loadChristianVis" value="false"/>
    <prop name="general_cm_removepersons" value="Remove Persons"/>
    <prop name="general_cm_removequestions" value="Remove Questions"/>
    <prop name="general_cm_removeall" value="Remove Everything"/>
    <prop name="general_cm_showgrid" value="Show Grid"/>
    <prop name="general_cm_hidegrid" value="Hide Grid"/>
    <prop name="general_cm_resetwindows" value="Reset Window Positions"/>
    <prop name="general_cm_zoomin" value="Semantic Zoom In"/>
    <prop name="general_cm_zoomout" value="Semantic Zoom Out"/>
    <prop name="general_cm_random" value="Semantic Zoom Out"/>
    <prop name="general_xsltransformedfiles" value="true"/>
    <prop name="general_xsltransformeddatafile" value="result-tpq.xml"/>
    <prop name="general_tablevis_qtp_filename" value="result-qtp.xml"/>
    <prop name="general_tablevis_qpt_filename" value="result-qpt.xml"/>
    <prop name="general_parameter_master_data_filename" value="parameter_master_data.xml"/>
    <prop name="general_experimentfilename" value="experiment.xml"/>
    <prop name="general_fulllivepreview" value="true"/>
    <prop name="questionrings_distance" value="0"/>
    <prop name="questionrings_mfdistance" value="30"/>
    <prop name="questionrings_thickness" value="3"/>
    <prop name="questionrings_alpha" value="100"/>
  </properties>
</resources>
```

```

<prop name="mf_overlap" value="1"/>
<prop name="general_show_tooltip" value="true"/>
<prop name="tooltip_fontsize" value="14"/>
<prop name="traces_alpha" value="100"/>
<prop name="starglyph_inverted" value="false"/>
<prop name="starglyph_legend" value="true"/>
<prop name="starglyph_alpha" value="100"/>
<prop name="starglyph_thickness" value="3"/>
<prop name="animate_sg_ar" value="false"/>
<prop name="helptext" value="Abstract:&lt;br&gt;Tracking and comparing psychotherapeutic
data derived from questionnaires involves a large number of
highly structured, time-oriented parameters. Descriptive and
other statistical methods are only suited for partial analysis.
Therefore, we invented a spring-based interactive Information
Visualization method for analysing these data more in-depth.
With our method the user is able to find new predictors for a
positive or negative course of the therapy due to the combination
of various visualization and interaction methods."/>

<!-- question -->
<prop name="question_textlabel" value="Question"/>
<prop name="question_size" value="100"/>
<prop name="question_show_label" value="false"/>
<prop name="question_nametag" value="false"/>
<prop name="question_cm_remove" value="Remove"/>
<prop name="question_cm_highlight" value="Highlight"/>
<prop name="question_cm_unhighlight" value="Unhighlight"/>
<prop name="question_cm_complement" value="Add Complement Question"/>
<prop name="question_cm_strength0" value="Strength:0 (no attraction!)" />
<prop name="question_cm_strength1" value="Strength:1 (standard)" />
<prop name="question_cm_strength2" value="Strength:2"/>
<prop name="question_cm_strength3" value="Strength:3"/>
<prop name="question_cm_strength4" value="Strength:4"/>
<prop name="question_cm_strength5" value="Strength:5"/>
<prop name="question_cm_strength6" value="Strength:6"/>
<prop name="question_cm_strength7" value="Strength:7"/>
<prop name="question_cm_strength8" value="Strength:8"/>
<prop name="question_cm_strength9" value="Strength:9"/>
<prop name="question_cm_strength10" value="Strength:10"/>
<prop name="question_cm_fixValue" value="fix/unfix current Timestep on this Question"/>
<prop name="question_loadimages" value="false"/>

<!-- person -->
<prop name="person_textlabel" value="Person"/>
<prop name="person_size" value="80"/>
<prop name="bmi_timetosizelabel" value="Map Size to Timestep"/>
<prop name="bmi_strengthlabel" value="Strength Indicator"/>
<prop name="bmi_tracethicknessbmi" value="1.3"/>
<prop name="bmi_tracethicknessbmi_min" value="12"/>
<prop name="bmi_tracethicknessstimesize" value="1.5"/>
<prop name="bmi_tracethicknessstrength" value="1.0"/>
<prop name="person_missing_data_alpha" value="20"/>
<prop name="person_cm_remove" value="Remove"/>
<prop name="person_cm_highlight" value="Highlight"/>
<prop name="person_cm_unhighlight" value="Unhighlight"/>
<prop name="colors_array" value=" 0x007700,0x008800,0x009900,0x00AA00,0x00BB00,0xDD0000,0xEE0000,
0x0000BB,0x0000CC,0xFF0000,0x00DD00,0x00EE00,0x00FF00,0x707070,
0xAAAAAA, 0xDDDDDD,0xEEEEEE,0xFFFFF,0x0085E6,0x7D00E6,0x11621F,
0x510000,0x0D0051,0x515000,0x003333,0xC1C1C1,0x3A3A3A,0xC9933,
0x1BC871"/>

<!-- listvis -->
<prop name="listvis_height" value="555"/>
<prop name="listvis_heading_persons" value="Persons"/>
<prop name="listvis_heading_questions" value="Questions"/>
<prop name="listvis_cm_addquestions" value="Add All Questions"/>
<prop name="listvis_cm_addpersons" value="Add All Persons"/>
<prop name="listvis_cm_addall" value="Add Everything"/>
<prop name="listvis_cm_highlight_question" value="Highlight Question"/>
<prop name="listvis_cm_unhighlight_question" value="Unhighlight Question"/>
<prop name="listvis_cm_highlight_person" value="Highlight Person"/>
<prop name="listvis_cm_unhighlight_person" value="Unhighlight Person"/>
<prop name="listvis_cm_removepersonfromgravi" value="Remove from Gravi++"/>
<prop name="listvis_cm_removequestionfromgravi" value="Remove from Gravi++"/>
<prop name="listvis_cm_addquestiontogravi" value="Add Question to Gravi++"/>
<prop name="listvis_cm_addpersontogravi" value="Add Person to Gravi++"/>
</properties>
</resources>

```

In addition to the resources.xml properties file of Gravi++, the overview visualization TableVis provides an own properties file called

props.xml

It contains only text labels and color values to customize the looks and the language settings of the TableVis overview visualization.

5.5 Menu XML File

The menu XML file contains the structure and name tags of the top menu bar. The “instanceName” attribute is not allowed to be changed as it is the name of the corresponding function that is called if this menu item is clicked in Gravi++. Here you can see an example of the menu.xml file:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<menu>
  <menuitem label="File">
    <menuitem label="Open..." instanceName="tm_load"/>
    <menuitem label="Save..." instanceName="tm_save"/>
  </menuitem>
  <menuitem label="Questions">
    <menuitem label="Remove Questions" instanceName="tm_removequestions"/>
    <menuitem type="separator"/>
    <menuitem label="Show Questions" instanceName="tm_hidemagnets" type="check" selected="true"/>
    <menuitem label="Show Star Glyph" instanceName="tm_stardinate" type="check"/>
    <menuitem label="Show Attraction Fields" instanceName="tm_magfield" type="check"/>
    <menuitem label="Show Attraction Field Label" instanceName="tm_magfieldlabel" type="check"/>
    <menuitem type="separator"/>
    <menuitem label="Randomize Question Position" instanceName="tm_random"/>
    <menuitem label="Circular Placement" instanceName="tm_circular" type="check" selected="true"/>
    <menuitem label="Evenly Distributed Placement" instanceName="tm_circulareven" type="check"/>
    <menuitem label="Animate StarGlyph and QuestionRings (slow!)"
      instanceName="tm_animate_sg_ar" type="check"/>
  </menuitem>
  <menuitem label="Persons">
    <menuitem label="Remove Persons" instanceName="tm_removepersons"/>
    <menuitem type="separator"/>
    <menuitem label="Show Persons" instanceName="tm_hidespheres" type="check" selected="true"/>
    <menuitem label="Show Traces" instanceName="tm_traces" type="check"/>
    <menuitem label="Show Whole Traces" instanceName="tm_wholetraces" type="check"/>
    <menuitem label="Show Grid" instanceName="tm_grid" type="check"/>
    <menuitem type="separator"/>
    <menuitem label="Speed">
      <menuitem label="stop" instanceName="tm_speed0" groupName="tm_speed" type="radio"/>
      <menuitem label="normal (standard)" instanceName="tm_speed1"
        groupName="tm_speed" type="radio" selected="true"/>
      <menuitem label="fast" instanceName="tm_speed5" groupName="tm_speed" type="radio"/>
      <menuitem label="fastest" instanceName="tm_speed10" groupName="tm_speed" type="radio"/>
    </menuitem>
    <menuitem label="Live Preview" instanceName="tm_live" type="check" selected="true"/>
    <menuitem label="Size..." instanceName="tm_spheresize"/>
  </menuitem>
  <menuitem label="Windows">
    <menuitem label="Overview ListVis" instanceName="tm_listvis" type="check" selected="true"/>
    <menuitem label="Overview TableVis" instanceName="tm_christianvis" type="check"/>
    <menuitem label="Toolbar" instanceName="tm_mainmenu" type="check" selected="true"/>
    <menuitem label="Time Control" instanceName="tm_timecontrol" type="check" selected="true"/>
    <menuitem label="Output" instanceName="tm_output" type="check" selected="false"/>
    <menuitem type="separator"/>
    <menuitem label="Show Tooltip" instanceName="tm_tooltip" type="check" selected="true"/>
    <menuitem type="separator"/>
    <menuitem label="Reset All Window Positions" instanceName="tm_resetwindows"/>
  </menuitem>
  <menuitem label="Help">
    <menuitem label="Help" instanceName="tm_help"/>
    <menuitem label="About" instanceName="tm_about"/>
  </menuitem>
</menu>
```

5.6 Data Import

There are currently two possible ways to import data to Gravi++ from the old data input tool called “Dateneingabetool”. The first is to use LinkVis [Herzog, 2004] to convert the data. The second option is to use the new GenEdit [Osztoivits, 2007] tool to import the data. Both options are now presented step by step.

Data Flowchart

From Questionnaires to Gravi++

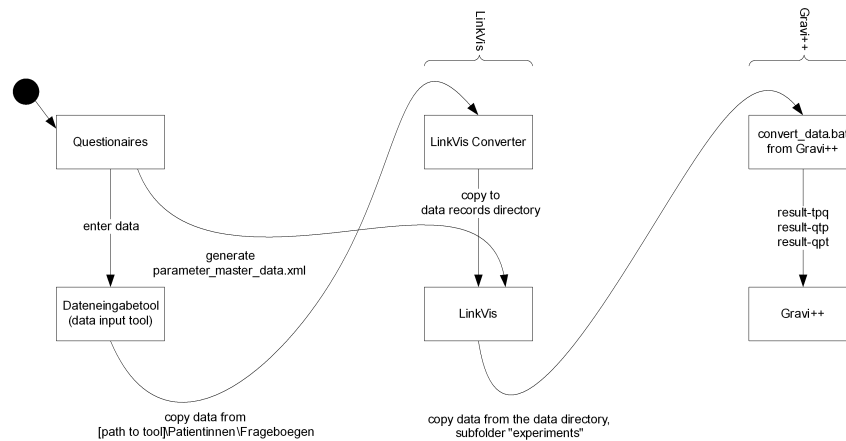


Figure 5.3: Flow Chart of the data import of the old "Dateneingabetool" to Gravi++.

5.6.1 Import using LinkVis

The following checklist is a detailed instruction on how to import data from the old "Dateneingabetool" to Gravi++. The sequence is also illustrated in Figure 5.3.

Step 1 Enter the data in the "Dateneingabetool" (data input tool). The data is stored in `[path to tool]\Patientinnen\Frageboegen` (e.g., `C:\anorexie\Patientinnen\Frageboegen`)

Step 2 Take the `parameter_master_data.xml` file (e.g., from LinkVis [Herzog, 2004] by Herbert Herzog). It contains information about all questionnaires, like the single answers and the data types.

Step 3 Convert the data using Herbert Herzog's "linkvisconvert" converter. The program is called with

```
java -jar [path to tool]\linkvisconvert.jar
--masterfile [path to parameter_master_file]\parameter_master_data.xml
--fbadir [path to original data]
--xmldir [path where output should be written (directory should not exist)]
```

For example:

```
java -jar C:\Projekte\visualisierung\LinkVisDemo_Download\LinkVisDemo\linkvisconvert.jar
--masterfile C:\Projekte\visualisierung\Anorexie_Daten_20060410\parameter_master_data.xml
--fbadir C:\Projekte\visualisierung\Anorexie_Daten_20060410\data_recods
--xmldir C:\temp\linkvis)
```

Look out for errors in the output of the converter. If any files could not be processed, edit the original file and look for XML errors (e.g., not well formed, empty files, ...).

Step 4 Copy the resulting data files from the output directory to the LinkVis data directory (in the sub directory `data_records`). Delete the old data values and insert the new ones. Don't remove the other sub directories like `cache`. After that, copy the `parameter_master_data.xml` in the LinkVis data directory (root - no sub directory).

Step 5 Start LinkVis and go to “Settings” - “Preferences” - “Data Storage” and enable the checkbox “Initialize data storage”. Furthermore, check if the data directory is correct and then press “OK”.

Step 6 Press the "New" button.

Step 7 Assign a new experiment name in the text field on the top.

Step 8 Check if the data source is set to “application data storage”.

Step 9 Select all Questionnaires and Questions (from left to right) of interest (in the parameter selection grayed out values mean that there is no data for this questionnaire).

Step 10 Press OK

Step 11 Press the save button

Step 12 In the data directory under the sub folder "experiments" search for the given experiment name (from step 7).

Step 13 Copy the conflation_ dataset.xml (all selected data) file to the Gravi++ data folder.

Step 14 Run convert_data.bat in the Gravi++ data folder (on Linux or MacOS open the convert_data.bat file and run the commands from this batch file on the console).

Step 15 Adjust some options in the resources.xml file (optional).

Step 16 Start Gravi++

5.6.2 Import using GenEdit

Step 1 Enter the data in the "Dateneingabetool" (data input tool). The data is stored in [path to tool]\Patientinnen\Frageboegen (e.g., C:\anorexie\Patientinnen\Frageboegen)

Step 2 Take the parameter_master_data.xml file (e.g., from LinkVis [Herzog, 2004] by Herbert Herzog). It contains information about all questionnaires, like the single answers and the data types.

Step 3 Convert the data using Herbert Herzog’s "linkvisconvert" converter. The program is called with

```
java -jar [path to tool]\linkvisconvert.jar
--masterfile [path to parameter_master_file]\parameter_master_data.xml
--fbadir [path to original data]
--xmldir [path where output should be written (directory should not exist)]
```

For example:

```
java -jar C:\Projekte\visualisierung\LinkVisDemo_Download\LinkVisDemo\linkvisconvert.jar
--masterfile C:\Projekte\visualisierung\Anorexie_Daten_20060410\parameter_master_data.xml
--fbadir C:\Projekte\visualisierung\Anorexie_Daten_20060410\data_recods
--xmldir C:\temp\linkvis)
```

Look out for errors in the output of the converter. If any files could not be processed, edit the original file and look for xml errors (e.g., not well formed, empty files, ...).

Step 3 Copy the data files and the “parameter_master_data.xml” file to the data directory of GenEdit or specify the paths in the GenEdit preference file.

Step 4 Start GenEdit and look out for possible errors in the console output (usually wrong parameter_master_data.xml settings).

Step 5 Export (a subset) as a conflated data set with a synchronized range directly to the Gravi++ data folder with the name “conflation_dataset.xml”.

Step 6 Run convert_data.bat in the Gravi++ data folder (on Linux or MacOS open the convert_data.bat file and run the commands from this batch file on the console).

Step 7 Start Gravi++

5.6.3 General Data Input

To visualize other data files that do not originate from the “Dateneingabetool”, GenEdit [Osztoivits, 2007] can be used to manually input the values or import CSV (comma separated values) files. To load this data into Gravi++ follow the above instructions provided in Section 5.6.2.

Chapter 6

Use Cases

In this section we will present two examples of using Gravi++ to analyze two different data sets. The first use case visualizes the medical data we presented in Section 3. The second example shows the application of our InfoVis technique to visualize questionnaires representing the attitude towards the European Union of citizens in Luxembourg.

6.1 Use Case 1 - Anorexia Nervosa - How to Find a Predictor

The following use case was conducted with our medical data of girls with eating disorders described in Section 3. Predictors allow to estimate the development of parameters after a certain treatment step had been performed. Therefore, we have to analyze the differences in the first answers and compare them with the differences over all timesteps. This can be done by watching the animations over time or analyzing the plotted traces of the persons' icons. Furthermore, the change over all timesteps can also show some sort of pattern that indicates a new predictor.

Experimenting sessions with our partners of the Department of Child and Adolescent Neuropsychiatry identified the following steps to find predictors. We will illustrate these steps using an example of our data set consisting of nine patients and 73 questions in six timesteps (see Figure 6.1-6.4):

1. Our domain experts chose due to the problem characteristics the questionnaires ASW (inverted)¹, BDI², MR EVA³, MR SOC⁴ and eight persons from the overview visualization ListVis, which we add to the spring-based core (see Figure 6.1). As the experts knew which questions to add, ListVis was the perfect choice to find the questions. In contrast to this, TableVis should be used when the user is not sure what subset she or he wants to analyze.
2. With the enabled option "Full Traces" we can analyze the traced paths of the persons over time (Figure 6.2).
3. By moving around the questions we try to find clusters of traces. In our example we have nine different persons. They are color coded the following way: The red icons and traces had a not favorable outcome in the therapy, whereas the green icons and traces had a positive therapy progress. This means we have to find clusters of evenly colored traces.
4. By positioning MR EVA and MR SOC on the one side and ASW (inverted) and BDI on the other side we can identify two clusters of red and green traces. In the upper screenshot of Figure 6.3 and 6.4 you can see the four questions after repositioning them (with two identifiable clusters of red and green person icons and Traces).

¹ASW: The Self-Efficacy Scale Index is a summary of the test items assessing self-esteem and strategies to cope with difficult situations independently.

²BDI: The Beck-Depression-Index describes the severity of depression.

³MR EVA: The Marburg Inventory Index specifies the feeling of pleasure in hedonistic behavior.

⁴MR SOC: The Sense of Coherence Scale Index indicates the intensity of comprehensibility, manageability and meaningfulness as the internal psychological mechanism mediating the effects of external stressors and resources on psychological dysfunction.

- By moving through the timesteps we can see that the randomly located persons in the first timestep, compose two clusters in the second and all later timesteps. Therefore, we can conclude that high values of BDI and the inverted ASW after three months therapy predict a bad outcome of the therapy, whereas high values of MR EVA and MR SOC predict a good outcome. Furthermore, we approve this claim by watching the movement over all remaining timesteps. In our example the persons with a positive therapy outcome still move further to the pole with MR EVA and the MR SOC and the persons with a negative outcome to the other pole. This means we can conclude that low BDI values and high MR EVA, MR SOC, and ASW values could be a predictor for a positive therapy outcome. This is confirmed by the clinical impression that patients who respond more rapidly to the primary therapeutic goals (enjoying pleasure and being in a good mood), have a better outcome.

This use case illustrated how to find a predictor using Gravi++. Especially, the animation and Traces view was useful for analyzing the therapy progress of the patients.



Figure 6.1: Four different questions and eight persons are added to the spring based core. By enabling the Attraction Rings view we can see the answer distribution of each question in the current timestep.

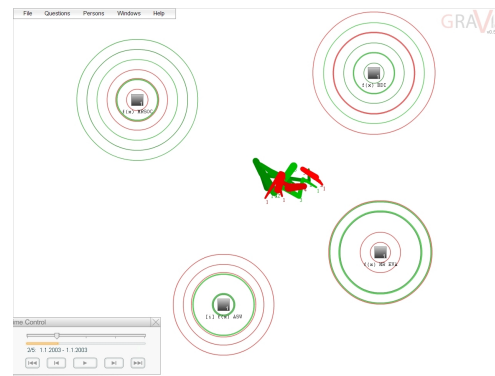


Figure 6.2: By enabling the Traces view in Gravi++, we are able to see the movement of the persons over time in a single image (corresponding to the therapy progress). In the above example no visible clusters of different persons can be seen.



Figure 6.3: Changing the position of the question icons leads to two clusters of green and red colored person icons in the current timestep.

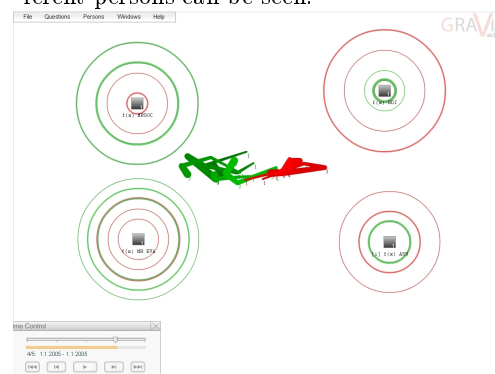


Figure 6.4: By enabling the Traces view, we can examine the two clusters over time. In the above screenshot, we can clearly see a separation of the two clusters after the first timestep.

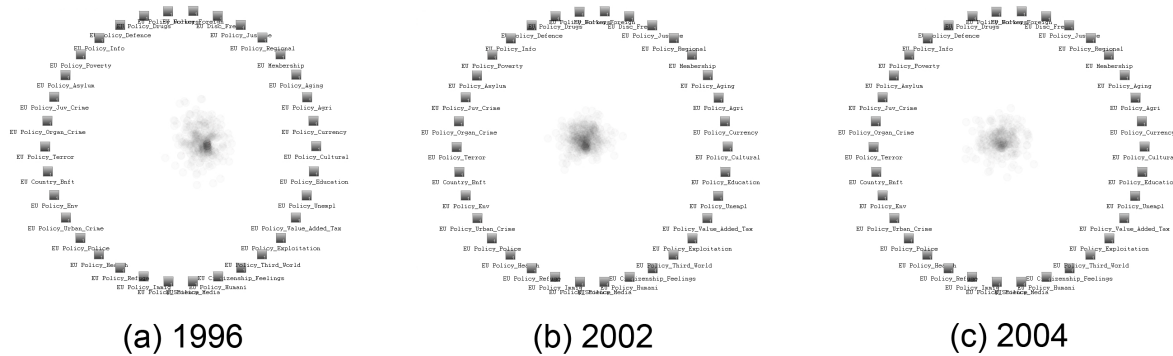


Figure 6.5: All questions and persons of the Eurobarometer data set are visualized over the three timesteps 1996 (a), 2002 (b), and 2004 (c). Due to missing data in some timesteps (questions that were not asked), the resulting person positions in the three image (a), (b), and (c) are not comparable.

6.2 Use Case 2 - Eurobarometer-Data

The Eurobarometer data originate from a survey that is commissioned by the European Commission in each European country to assess the attitude towards the European Union (EU). The sample size of each questionnaire lies between 400 to 2000 people (depending on the size of the country). The surveys are asked in telephone interviews on a voluntary basis. Each questionnaire consists out of trend questions and (for the country) specific questions and is conducted two times a year in each country. More information on the Eurobarometer data and some interpretations can be found in [Kritzinger, 2005, Kritzinger, 2003, Lutz et al., 2006, Dalton and Eichenberg, 1998, Gabel, 1998, Schmidberger, 1997, Wessels, 1998].

In this use case we chose a subset of this large data repository, namely only questionnaires from Luxembourg with about 600 persons and three timesteps (1996, 2002, and 2004). Luxembourg was chosen because of the smallest sample size of all countries. After cleaning the data (removing persons that did not answer questions and scaling the data) 426 persons were visualized with 19 questions (only these 19 questions were answered by all persons every year). Luxembourg is a very EU friendly country and not representative for the whole EU. For example, the answers of the population of Luxembourg are in many thematic sectors in contrast to the answers for example the Austrian people gave (not subject of this use case).

To visualize this high amount of structured data we needed to adjust the transparency value of the person icons and the Star Plot view to get useful results. After playing around with the settings, we realized that a one percent alpha value for the Star Plot lines and a five percent value for the person icon seem to be ideal for our data set. We will now illustrate the steps to analyze this data set:

1. In the first step we add all questions and persons from the overview visualization ListVis to the spring-based core visualization. The resulting three images of the three timesteps 1996, 2002, and 2004 can be seen in Figure 6.5.
2. Looking at the visualization, we can see that each icon is indicating that it is not attracted by each question (red triangle inside the person icon, although hardly visible because of the high transparency value). We now remove all questions that do not have answers in at least one timestep to ensure the comparability of the person icons over time. This can be done by looking at the tooltip of each question or person, or by looking at the additional views Star Plot or Attraction Rings.
3. We can now see that most questions (except four) are on different EU policy themes. Therefore, we decided to analyze these questions alone in a first step (and look at the remaining four questions later). These questions are about which political decisions should be either made by the national government or the EU government. The answers are ranging from 0 which means “National only” to 6 which stands for “European only”. The questions are asked for example: “Some people believe that certain areas of policy should be decided by the Luxembourg government, while other areas of policy should be decided jointly within the European Union. Which of the

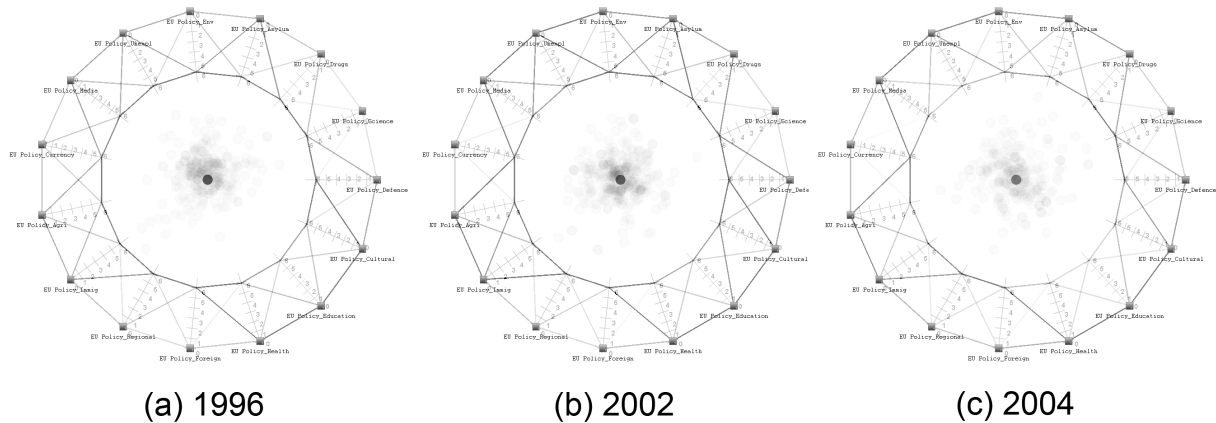


Figure 6.6: All policy questions are arranged on a circle with enabled Star Plot view to see the distribution of the exact answers on each question.

following areas of policy do you think should be decided by the Luxembourg government, and which should be decided jointly within the European Union? “

4. The next step is to activate the Star Plot view to analyze the distribution of answers of each question (see Figure 6.6). Now we can group similarly distributed questions using drag and drop operations on the question icons.
5. We will now group the questions with many answers of 0 (which means “National only”) on the right lower side of the circle (see Figure 6.7). As the used data is no panel data (different persons were asked over time), the animation and Traces view (see Figure 6.8 for an example) are only of limited use. Therefore, the single images of each timestep (looking at the distribution of the data) are more helpful.
6. Analyzing the three different images, Figure 6.7 (a) for 1996, Figure 6.7 (b) for 2002, and Figure 6.7 (c) for 2004, we can see one pole (Immigration, Health, Education and Cultural questions) that was mostly answered “National” over time. This can be seen by looking the positions of the person icons and the color of the Star Plot lines (because of the transparency - the darker the color of the icon or line, the more persons are represented with it). This finding that most people think that topics about immigration, health, education, and culture should be handled by the national government is also acknowledged by a domain expert.
Looking at the Star Plot lines over the time, we can see that the question on the policy of currency gets more European over time. This can be traced back on the introduction and support of the Euro currency.
Another finding is, that the people think that problems of unemployment should be dealt more by national governments as time progresses.
7. Looking at the person icons we can see a dark icon in the center of the visualization. It represents the majority of all persons (as it is very dark - which stands for many overlying icons because of the high transparency value of the single items). This means, most of the people answered all questions jointly over time, either all with “National only” or “European only”. Interestingly, this icon gets lighter over time. This means the answers get more differentiated. The domain expert related this finding to the fact that in 1996 there was more a permissive consensus of the answers. This changed over the years and became differentiated and clear opinions in 2002.
8. As the next step, we want to analyze the four questions that we removed to analyze the policy questions alone. Again we chose to use the Star Plot get further information on the value distribution of each question (see Figure 6.9). The four remaining questions are

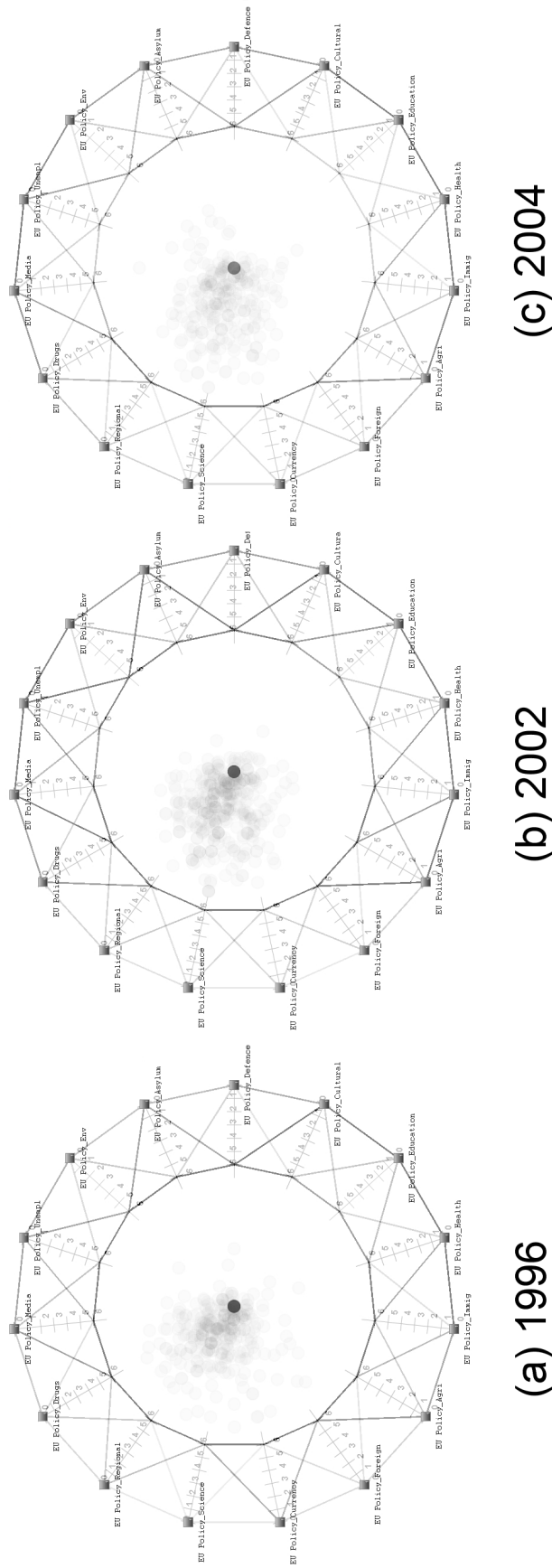


Figure 6.7: After sorting the policy questions according to their optical data distribution, we can see that the majority of person icons is visualized in the center and on the left side of the screenshots.

EU Disc_Freq: Political Discussion Frequency (0 ... frequently, 6 ... never): When you get together with friends, would you say you discuss political matters frequently, occasionally, or never?

EU Country_Bnft: EU Membership - Country Benefit (0 ... positive, 6 ... negative): Taking everything into consideration, would you say that Luxembourg has on balance benefited or not from being a member of the European Union?

EU Citizenship_Feelings: European Citizenship Future Feeling (0 ... Luxembourg, 6 ... European): In the future do you see yourself as ...?

EU Membership: EU Membership - Good / Bad (0 ... A good thing, 6 ... A bad thing): Generally speaking, do you think that Luxembourg's membership of the European Union is ...?

9. Looking at these four single questions over time, reveals us that most of the questions were answered the following way:

Political Discussion Frequency: Occasionally

EU Membership - Country Benefit: positive

European Citizenship Future Feeling: National and European

EU Membership - Good / Bad: A good thing

This overall trend was also confirmed by our domain expert. Interestingly, the citizenship feelings are getting more national over time. The domain expert confirmed this and explained this trend with the fact that the feelings got more national after the events of the terrorist attack at September 11, 2001 on the United States of America.

10. Looking at the person icons we can see an accumulation on the central axis. This means that the "Membership" and "Country Benefit" question were mostly answered with equal values. Most person icons are located a bit above the center. That means that these persons answered the questions with equal values. Other interesting positions are the darker icons above and beyond the center. They can be interpreted, that the persons answered "Country Benefit" and "Membership" equally (usually both positive) and either frequently discuss ("Discussion Frequency: frequent) and feel more European or discuss never and feel more national. This was also confirmed by our domain expert.

Concluding this use case, we can say that despite Gravi++ was not developed for large amounts of data, the visualization successfully visualized the data and showed us many different facets of the data. The domain expert validated all of our findings with her background knowledge on the data. Still the interactive component did suffer because of the limitations of our Macromedia Flash prototype. Long waiting periods did occur when adding or removing questions while visualizing all person icons. Therefore, we can conclude that our prototype is not suited for visualizing much more than the 500 persons in this example.

6.3 Conclusion

In our two presented use cases we demonstrated the usefulness of Gravi++ to visualize highly structured temporal data. The animation and the Traces view are very useful for tracking the changes of persons over time. In addition to this, the Star Plot view provides information on the exact values. When choosing an appropriate level of transparency, the Star Plot is also useful for visualizing a few hundred persons. Overall, the second use case demonstrated, that Gravi++ can even be applied for not perfectly fitting data. In our case, Gravi++ was not developed for visualizing different persons over time and a high amount of data. Although it proved to be useful, the user has to be careful interpreting this data, because the animation and Traces view can provide a wrong impression on this kind of data. All in all, Gravi++ provided many useful insight into the data which were also confirmed by our domain experts.

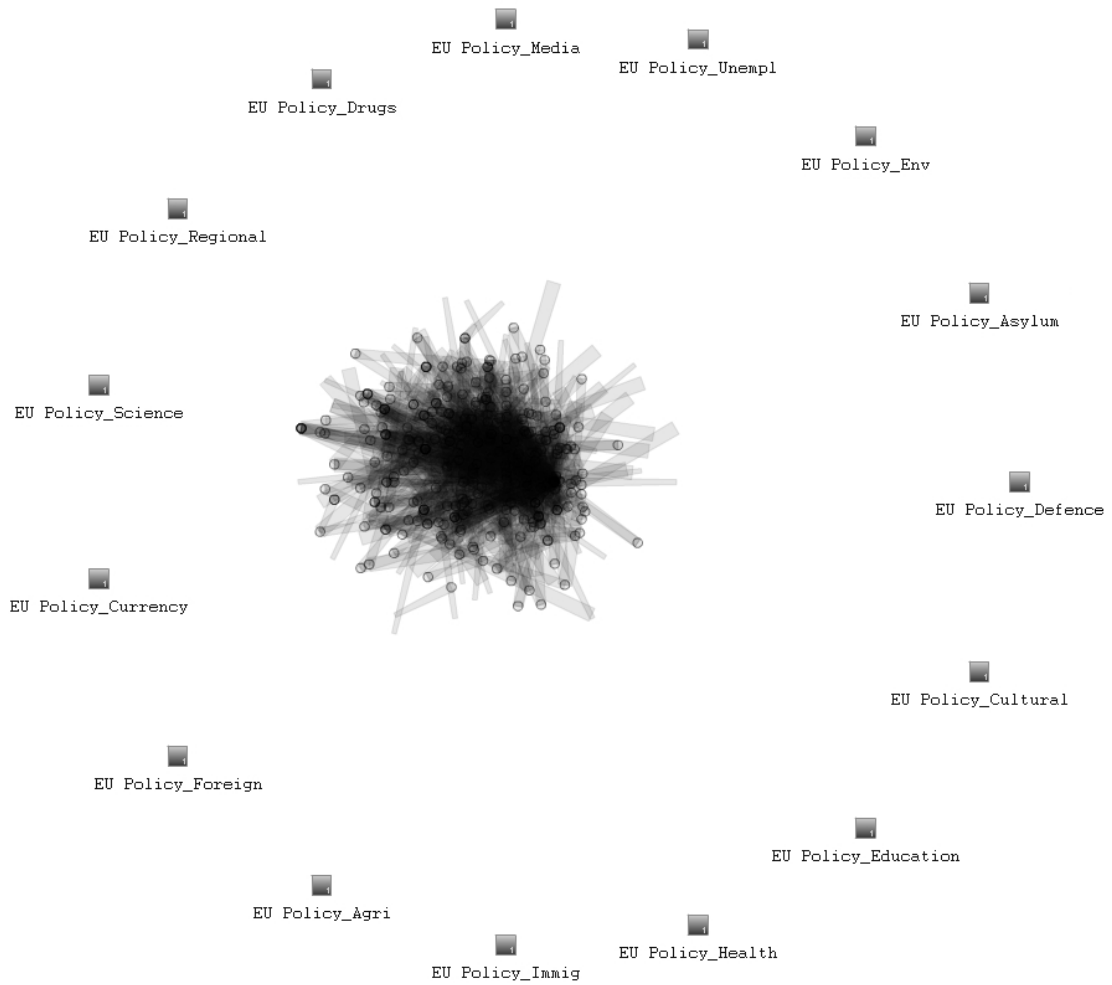


Figure 6.8: The Traces view shows the data values of all three timesteps at a glance but may be misleading because the data is not based on a panel questionnaire (different persons were asked in each timestep).

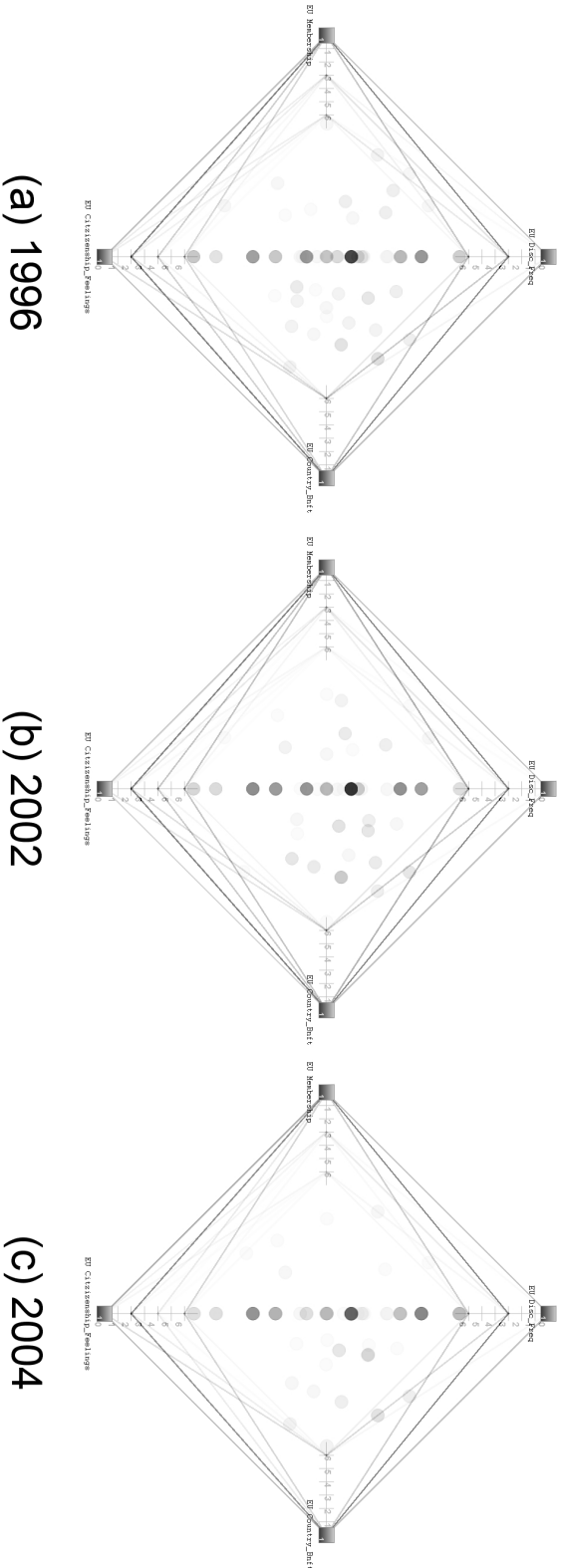


Figure 6.9: The four remaining questions in all three timesteps 1996 (a), 2002 (b), and 2004 (c) are shown above. We can see the similar answer distribution of the left and right question and by looking at the darkest Star Plot line we can conclude on the most given answers.

Chapter 7

Evaluation

Several authors have pointed out the importance of evaluation studies of InfoVis techniques (see e.g., [Chen, 2000, Chen, 2004, Plaisant, 2004, Tory and Möller, 2004]). In the past few years usability studies concerning visualization methods have become more frequent and valuable information about the design of such systems has been gathered. Nevertheless, as Spence [Spence, 2001] mentions, there is still too little systematic information about the specific strengths and weaknesses of the features of InfoVis techniques. On the basis of existing evidence it is still difficult to decide which InfoVis technique to use for which purpose. Therefore, evaluation studies are especially important to give the developers some insights into the usefulness of a given InfoVis technique for the intended area of application.

Stage	Method	Subjects	Aim	Outcome
Usability	usability inspection	usability expert	spot most obvious glitches	31 usability problems
	heuristic evaluation	27 students (semi-experts in usability)	in depth testing	447 reports documenting 576 problems (221 different)
	focus groups		additional usability assessment	no new problems BUT different perspective
Insight study (Gravi++, EDA, Machine Learning)	insight reports	33 students (domain novices)	patterns of insight & cognitive strategies	909 reports <i>work in progress</i>
	log files		used vis. options & exploration strategies	56055 log file entries <i>work in progress</i>
	focus groups		relativize findings & aids correct interpretation	transcription of 3x 100min <i>work in progress</i>
Case study	interviews	2 clinicians (real users)	feasibility & usefulness in real life	<i>work in progress</i>
	thinking aloud			<i>work in progress</i>
Transferability	interviews	20 experts of other domains	usefulness in other domains	transcription of 20x 60min <i>work in progress</i>

Table 7.2: Overview on all tasks of our evaluation of Gravi++.

Because of this relevance, we conducted an extensive evaluation of our visualization Gravi++ in different stages. Table 7.2 presents an overview on our whole evaluation that is still in progress (for a

detailed description see [Rester, 2007]). In the following we will focus on the results that are related to Gravi++, namely:

- A **usability evaluation** consisting of a usability inspection, a heuristic evaluation, and focus group interviews.
- An extensive **insight study** where we wanted to find the strength and limitations of three different methods including Gravi++.
- A **transferability assessment** to find other possible usage scenarios for our Information Visualization.

This evaluation was conducted by our project partners Markus Rester, Margit Pohl, and Sylvia Wiltner (see [Rester, 2007, Rester et al., 2005, Rester et al., 2006b, Rester et al., 2006a]) from the Institute of Design and Assessment of Technology, Vienna University of Technology with Gravi++ v0.300 without the TableVis view. In addition to this, the features “complement question” and “fix question” were not implemented or activated for the study.

7.1 Usability Study

As the first step of our study the usability of the Gravi++ system was tested to preclude that usability problems are mixed up with problems with the visualization method as such. The importance of a usability study for InfoVis is described in Chen’s article “Top 10 Unsolved Information Visualization Problems” [Chen, 2005] as the number one problem. He states that

“Although the overall growth of information visualization is accelerating, the growth of usability studies and empirical evaluations has been relatively slow.” [Chen, 2005]

and

“The complexity of the underlying analytic process involved in most information visualization systems is a major obstacle; end users cannot see how their raw data is magically turned into colorful images.” [Chen, 2005]

Therefore, we can conclude the need for a usability study to ensure that the user can actually make use of the possible advantages of a visualization and handle the complexity of the data set. In the following the results of the usability study are described.

7.1.1 Methods

Three different approaches in gathering information for the improvement of usability were adopted. They are targeting different possible insights: a usability inspection / guideline review, heuristic evaluation, and focus groups.

7.1.1.1 Usability Inspection / Guidelines Review

The usability inspection is an informal way of a usability evaluation to identify easy to find usability problems. The basis for this procedure were Tognazzini’s rather specific principles and guidelines for the design of interfaces [Tognazzini, 1992] and Nielsen’s very general heuristics [Nielsen and Mack, 1994, Mack and Nielsen, 1993]. Our usability expert Markus Rester performed the analysis with an early prototype of Gravi++.

7.1.1.2 Heuristic Evaluation

The second part of the study was a Heuristic Evaluation [Nielsen and Mack, 1994] which is one of the most recognized usability evaluation techniques. A small group of evaluators checks the compliance of given software against ten usability principles. These principles are of a general nature and therefore called “heuristics”. This assessment method was conducted with 27 students of thirteen different but mostly informatics-related studies. About half of the students already attended lectures on user interface design and/or usability engineering and all of them could be easily familiarized with the field of usability evaluation and therefore can be described as semi-experts. It is a generally accepted view that usually 3-5 expert evaluators are enough to find most usability problems [Holzinger, 2005]. Because of our higher number of (semi-expert) evaluators it makes sense that the subjects were students.

We chose the Heuristic Evaluation as one step in our usability assessment because it suits our needs in various ways:

- It serves as a foundation for the students to deal with the subject of usability and enables them to consider more abstract principles (in the later focus groups assessment).
- The heuristics were available to the students during the experiment both as handout and as part of the report-system and built an excellent link to the learned matter.
- The procedure encourages the students to compose concise and accurate reports on found usability problems and therefore ensures a high-quality evaluation even with semi-experts.

7.1.1.3 Focus Groups

At Focus Groups (see e.g., [Kuniavsky, 2003]), subjects discuss and comment on their experience with a given topic. They represent a form of a group interview with the difference that group interviews concentrate on questions of the researcher and responses of the subjects while focus groups are based upon the interaction between the subjects on a given topic. This interaction can lead to finding new problems, which would not have been individually anticipated by any of the participants. The very same characteristic holds a risk one has to deal with, the interactions can lead to group-thinking. Therefore, it is important to conduct at least two focus groups. Also the size of the groups should not be too big so that every participant can take active part in the discussion.

The focus groups part of our study was conducted with the same subjects who took part in the heuristic evaluation.

7.1.2 Lab Settings

The Heuristic Evaluation was conducted in sessions, lasting three hours each. In the beginning, the procedure was once more explained to the subjects. They also received handouts summarizing the heuristics, stating typical tasks performed with the visualization, and explaining the steps of evaluation in detail. In addition to that, the subjects could at any time ask the investigator if they needed any further help.

During conducting the usability test, the subjects were assisted in documenting their findings by a report system developed by Rester, Pohl, and Banovic [Rester et al., 2005]. With this system the subjects could report a problem by describing it, uploading a screenshot, and choose the appropriate principle of the heuristics (or specify any other usability guideline they knew of).

7.1.3 Results

All three methods for assessing the usability of our InfoVis method did show valuable results and did lead to a vast improvement of the prototype. In the following we will present you more detailed results of each usability method.

Rule	Mentions	Percentage
1. Visibility of system status	63	12.28 %
2. Match between system and the real world	40	7.80 %
3. User control and freedom	59	11.50 %
4. Consistency and standards	105	20.47 %
5. Error prevention	23	4.48 %
6. Recognition rather than recall	19	3.70 %
7. Flexibility and efficiency of use	32	6.24 %
8. Aesthetic and minimalist design	52	10.14 %
9. Help users recognize, diagnose, and recover from errors	12	2.34 %
10. Help and documentation	33	6.43 %
11. Other Rule	75	14.62 %
Sum	513	100.00 %

Table 7.3: Number of mentions of usability problems for each rule in the heuristic evaluation.

Usability Inspection The Usability Inspection revealed 31 usability problems based on the principles and heuristics by Tognazzini [Tognazzini, 1992] and Nielsen [Nielsen and Mack, 1994]. Most of them concerned wording, arrangement of options, and inconsistencies in the user interface issues. The problems were discussed and solved before the main Heuristic Evaluation to help the subjects in finding more complicated problems.

Heuristic Evaluation During the Heuristic Evaluation 447 reports were generated that documented 513 breaches of principles (see Table 7.3). The most noted rule was “Consistency and Standards” that was reported 105 times. After categorizing the reports, 221 different problems were noted. The 20 most noted problems can be found in Table 7.4. Noticeable is the high rate of bugs, performance problems, and easily identifiable problems (lacking help). Most of them are due to the early development stage of Gravi++ but still gave valuable hints for future development directions.

In addition to this, our usability expert tagged important and interesting reports for further pursuance. 128 reports were marked that way and evaluated to further enhance Gravi++.

Focus Groups The focus groups did not bring up any new problems but gave us a better insight into the found problems and also provided us with a different perspective on some of the found usability problems. 27 different problems in 46 total cases were noted during the three focus group sessions. The most often mentioned drawback was the missing undo/redo feature.

7.1.4 Conclusion

The Usability part of our study was very important for the development of Gravi++. It precluded that usability problems were mixed up with problems with the visualization method as such. In addition to this, it lead to a remarkable improvement of our InfoVis prototype. Especially, wording and consistency problems were found in the usability inspection and solved afterwards. The Heuristic Evaluation and the Focus Groups did give important feedback on the development prototype and also revealed some not so obvious bugs. Generally, many usability problems would have never been solved without the usability study because of the close involvement of the project team in the development of the prototype.

7.2 Study

As described in the introduction of this part, evaluation studies are especially important to give the developers insights into the usefulness of a given InfoVis technique for the intended area of application. The following study describes an investigation about how to support psychotherapists in their work (see Chapter 3 for more information on the medical problem domain).

We tested three possibilities how to support the therapists’ work:

Problems Found	Mentions
Bug: Numbers next to Full Traces are sometimes missing	19
Missing help	18
Bug: Disappearing question icons	17
Top menu items are visually cut	15
The output window is editable	14
Bug: Gravi++ not responding	13
Bug: Traces are displayed wrong	11
Flash player warns because of possible infinite loop	11
Bug: Wrong Star Plot spike in the left corner	10
Missing undo	10
Send to top option is unclear	10
Bug: Enabling the Star Plot is not synchronized in the toolbar and top menu	9
Bug: Linking of ListVis and the spring-based core is not always working.	8
Output window is unclear	8
Missing exit option in the top menu bar	7
Missing warning window when replacing a saved state	7
Bug: Sub menu in the top menu is cut off.	7
Unclear performance warning of the Flash player.	7
Some details of the questions are out of the display area	7
Status feedback of the timeline is unclear	7

Table 7.4: Most mentioned number of problems found in the heuristic evaluation of the usability of Gravi++.

- Gravi++, the interactive Information Visualization described in this thesis that was specifically developed for this purpose
- Exploratory Data Analysis (EDA) techniques (Histogram, Box Plot, Scatter Plot Matrix, descriptive Values - see Chapter 7.4 for an example) and
- supervised Machine Learning (ML) techniques (a C4.5 decision tree - J48 in WEKA and sequential minimal optimization (SMO) for training support vector machines - see Chapter 7.4 for an example).

An interactive Information Visualization was chosen because the characteristics of our data set (high dimensionality, time-dependency, small sample size, no linear progress in the data) indicates that it might be a good method to analyze the data. EDA seems to be an interesting and well known alternative because of its exploratory nature. ML takes the approach, to let the computer analyze the data and present the result (and the steps leading to this result) to the user. Our original assumption was not that one of these techniques might be the best but rather to find the specific strengths and weaknesses of these methods and how to combine the methods to get the best results.

Due to the complexity of domain and data, different scenarios were provided. Two scenarios were specifying meaningful subsets of data to explore (questions, patients, time steps) and two questions were stating concrete questions. The first two scenarios led to undirected exploration whereas the last two questions dealt with concrete tasks (data subset + question specified).

The intended types of investigations can be characterized as follows:

1. Scenario A: Realize the change over time of 16 patients in 5 dimensions and identify positive and negative predictors (e.g., all patients whose depression scores do not drop in the middle of the therapy are likely to have a negative therapy outcome).
2. Scenario B: Recognize the consistent / inconsistent answers of parents and patients in the first time step and their role as predictors.
3. Question 1: Analyze the effect of the therapy on one specific dimension over time.

Usability Problems (Total Mentions > 1)	Total Mentions	Corrected
Undo / redo missing	7	no (design decision)
Attraction Rings: Which circle corresponds to which person	3	yes (tooltip)
Performance problems	3	yes
Time Control feedback is confusion	3	yes
Bugs in the Traces	3	yes
Everything should be controllable via menu	2	yes
Help is missing	2	yes
Option to reset the positions of each window is missing	2	yes
Bugs in load and save	2	yes
No project files but saved states are used	2	no (design decision)
	29	

Table 7.5: Number of mentions of usability problems in the focus groups.

4. Question 2: Predict a positive or negative therapy outcome of a yet unclassified patient with the available data of the two time steps.

The major goal of this study is the answering of the following question:

- What are the types of insights gained with the different tools?
- Can the tools be used together in some order to maximize the comprehension of the data?
- What are the specific strengths and weaknesses of the three methods?

7.2.1 Report System

The subjects used a report system to document their findings during the exploration process. It is accessible with any conventional web browser and is implemented in Perl using a MySQL database to store the collected reports. The following data is collected: used material, description of findings, and confidence rating.

7.2.2 Classification System

For the amount of collected data we need a feasible system to support the analysis. We have two different views to access the stored data in the relational database: a list view with all important information of the documented reports and a detailed view of a specific report that offers multiple classification options. The classification contains the assigned insight with respective complexity, the plausibility of the documentation of this insight, and additional argumentation. The classification system was developed while classifying the reports in cooperation with our domain experts (they were for example asked for a list of possible insights into the data). The classification resulted in thirteen main categories that contained further sub categories (in a tree like structure). For example “data - n ident - pos neg - Restraint - t1” represents an “observation of nearly identical scores of patients with a positive therapy outcome and patients with a negative one on the questionnaire measuring the restraint on ingestion by the patients in time step one”. This hierarchy allowed us to summarize observations according to selected subsets. The complexity of each insight was rated on a three-step scale: low, medium, and high. The plausibility was also rated on a three-step scale: not plausible, moderately plausible, and very plausible.

7.2.3 Results and Discussion

32 subjects documented an overall of 876 reports in sessions of 155 minutes. The number of reports and assigned insights are listed in Table 7.6. An average of of 2.47 insights were assigned to a report. Pearson’s chi-square test shows significant differences of the number of reports in column one of Table 7.6. In addition to this, analyzing the whole Table 7.6 shows significant divergency regarding reports and

	Reports	Insights
EDA	375	846
Gravi++	235	711
ML	266	609
Sum	876	2166

Table 7.6: Number of documented reports and assigned insights of our study.

Insights	EDA	Gravi++	ML	Sum
High Complexity	453	427	270	1150
Medium Complexity	130	130	48	308
Low Complexity	263	154	291	708
Sum	846	711	609	2166

Table 7.7: Number of insights of EDA, Gravi++, and ML by complexity.

Insights	EDA	Gravi++	ML	Sum
Correct Arguments	244	195	137	576
Mediocre Arguments	585	474	433	1492
Wrong Arguments	17	42	39	98
Sum	846	711	609	2166

Table 7.8: Number of insights of EDA, Gravi++, and ML by complexity.

insights. There are fewer reports filled out for Gravi++ but at the same time more insights generated with it.

Another finding is, that the numbers of insights of the thirteen main categories differ significantly for the three used methods.

The sub category about “data” findings (reading off, comparing scores) showed that both decision tree J48 and SMO formula of the ML part cannot communicate individual scores of patients very well. Gravi++ and EDA differ significantly.

ML lacks high and medium complex insights and shows overly low complex insights. The opposite holds true for Gravi++. See Table 7.7 for more details.

As mentioned above, the plausibility of the recorded data was also noted during the classification. The number of insights and their plausibility classifications show no significant difference and all of them have their highest ratios in the very plausible category (79% EDA, 79% Gravi++, and 75% ML).

The last of the primary classification levels is whether an assigned insight has been elaborated in more detail. In this case we noticed a significant number of fewer wrong arguments with EDA and more wrong arguments with ML than expected (see Table 7.8).

7.2.4 Conclusion

Summing up the major findings of this study we can conclude that fewer reports were generated while using Gravi++ and at the same time more insights gained with Gravi++ than with the other two techniques. Observations of scores of individual patients are underrepresented in Gravi++, whereas findings based on comparisons resulting in “nearly identical” insights are overrepresented (conforms to [Pillat et al., 2005]). The subjects were systematically not deterred by any of the three methods. The underrepresentation of insights for the first scenario with the use of ML and at the same time an overrepresentation of insights for the second scenario and the second question may be due to the need for an extensive familiarization phase with this method. There were many low confidence ratings with ML and few low confidence ratings with Gravi++. ML lacked high and medium complex insights and showed overly low complex insights. The opposite held true for Gravi++. The numbers of insights and their plausibility classifications showed no significant difference. All of the three methods had their highest ratios in the “very plausible” category. There were fewer wrong arguments with EDA and more wrong arguments with

ML than expected. Summarizing, a combined usage of the three methods is indicated because of their different strengths and weaknesses. Any obvious (or maybe superficial) benefits and limitations of the different methods should not lead to an exclusive use of one technique. Used in conjunction all three methods will very likely contribute to a deeper comprehension of the data to explore.

7.3 Transferability

To date, 14 of a total of 20 planned interviews with experts of other domains (e.g., social sciences, intensive care medicine, science of history, online journalism, etc.) have been conducted. There is already ample indication for promising usage scenarios of Gravi++ in a lot of different domains. Transcription is currently undertaken and qualitative analysis will follow. One domain expert for example suggested the use of Gravi++ to visualize stock market data. Images of rated stocks should help the user to get information about stock behavior over time.

7.4 Summary

The evaluation of Gravi++ was conducted in two parts. First, an intensive usability study to enhance the visualization and preclude that usability problems are mixed up with problems with the visualization method as such. The usability tests did show valuable results and lead to a vast improvement of Gravi++. The second part was an intensive insight study, where we wanted to find the strength and limitations of three different methods to present our medical data to the user. The results indicated that Gravi++ is a proper tool to analyze our medical data and get insights into it. In addition to this, a combined usage of all three methods of our insight study (EDA, ML, and Information Visualization techniques) was indicated because of their different strengths and weaknesses.

Acknowledgments

First of all I would like to thank my supervisor **Silvia Miksch** for supporting me so well, encouraging, helping, and motivating me throughout the whole project. Furthermore, my second advisor **Christian Popow** deserves many thanks for his valuable comments and his support throughout this work.

Thanks to all further in2vis project members **Markus Rester**, **Susanne Ohmann**, and **Margit Pohl** for their feedback and influence during the development of Gravi++ and this thesis.

Special thanks to **Wolfgang Aigner** for his contributions, discussions, and criticisms during the development phase of Gravi++ and his valuable work that formed a basis for parts of this thesis.

In addition to this, I want to thank **Andreas Seyfang**, **Kathi Kaiser**, **Peter Votruba**, **Horst Kargl**, and **Hakan Kalinyaprak** for their ideas and contributions. Also, many thanks to **Sylvia Kritzinger** for providing me with the Eurobarometer data and taking the time to analyze it with me.

Moreover, I would like to thank **Wolfgang Aigner**, **Kathi Kaiser**, and **Michael Strommer** for proofreading my thesis and providing me with valuable feedback.

Finally, I want to thank my **mother**, **father** and especially my girlfriend **Katja** for their personal support.

This thesis has been carried out as part of the in2vis project which is supported by The Vienna Science and Technology Fund (WienerWissenschafts-, Forschungs- und Technologiefonds - WWTF), grant WWTF CI038.

Appendix

The following three images show a part of our study material (see Chapter 7) that was handed out to the subjects of the insight study. Illustration 7.1 and 7.2 show two sheets from the Exploratory Data Analysis (EDA) part and Illustration 7.3 shows the C4.5 tree from the supervised Machine Learning (ML) part.

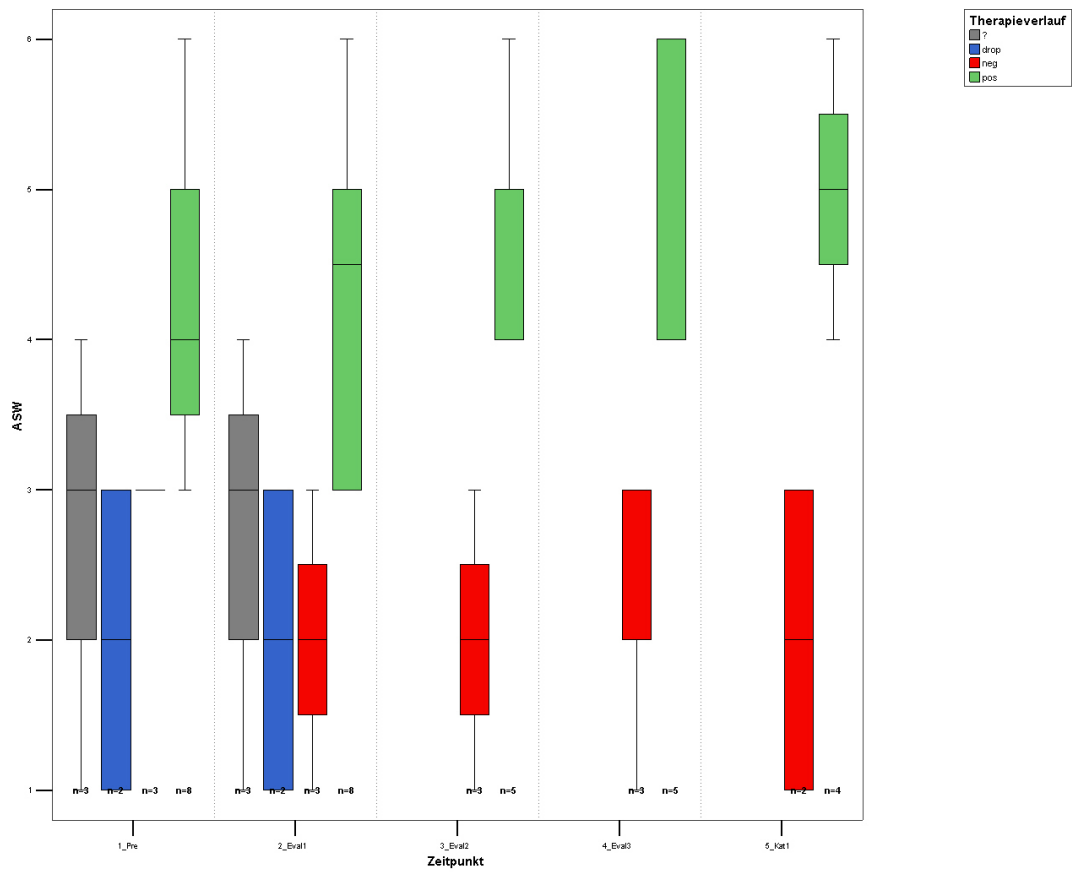


Figure 7.1: BoxPlot of question ASW and all timesteps.

Histogramm - Zeitpunkt 1 von 5

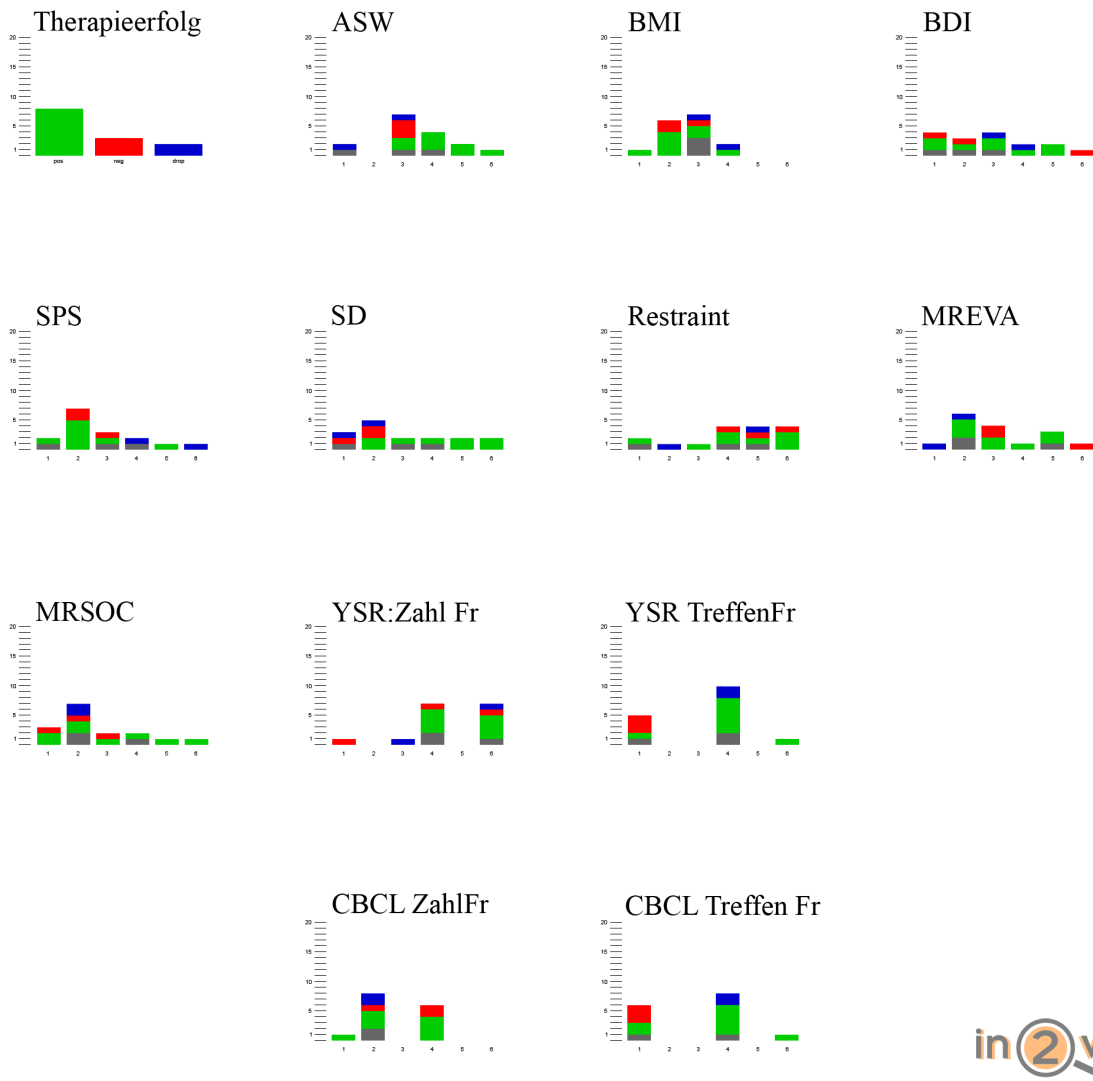
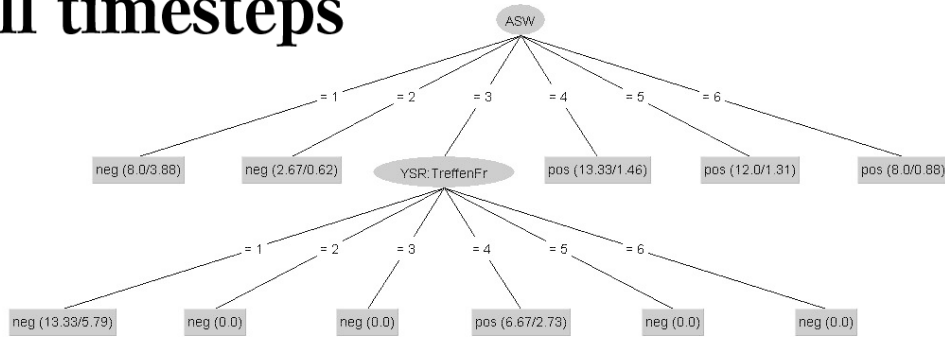
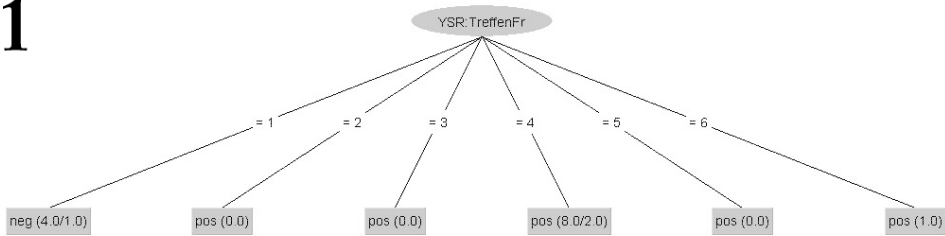


Figure 7.2: Histograms of all questions at timestep one.

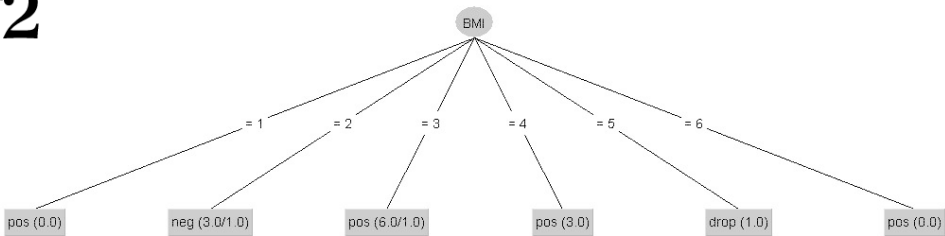
All timesteps



T1



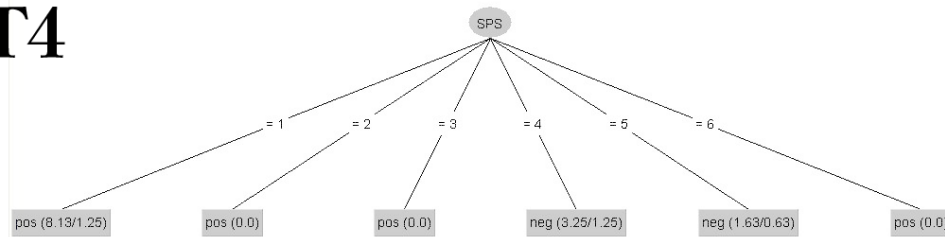
T2



T3

pos (13.0/5.0)

T4



T5

pos (12.0/5.0)

Figure 7.3: Machine Learning C4.5 trees of all timesteps.

Bibliography

- [Achenbach and Edelbrock, 1983] Achenbach, T. M. and Edelbrock, C. (1983). *Manual for the Child Behavior Checklist/4-18 and Revised Child Behavior Profile*. University of Vermont, Department of Psychiatry, Burlington.
- [Aigner, 2006] Aigner, W. (2006). *Visualization of Time and Time-Oriented Information: Challenges and Conceptual Design*. PhD thesis, Vienna University of Technology, Institute of Software Technology and Interactive Systems.
- [Akaishi and Okada, 2004] Akaishi, M. and Okada, Y. (2004). Time-tunnel : Visual Analysis Tool for Time-series Numerical Data and its Aspects as Multimedia Presentation Tool. In *Proceedings of the 8th International Conference on Information Visualisation (IV04)*, pages 456–461. IEEE.
- [Alty et al., 2000] Alty, J. L., Knott, R. P., Anderson, B., and Smyth, M. (2000). A framework for engineering metaphor at the user interface. *Interacting with Computers*, 13(2):301–322.
- [Anderson, 1989] Anderson, J. R. (1989). *Kognitive Psychologie, 2. Auflage*. Spektrum der Wissenschaft.
- [Andrienko and Andrienko, 2004] Andrienko, G. and Andrienko, N. (2004). Parallel Coordinates for Exploring Properties of Subsets. *Second International Conference on Coordinated and Multiple Views in Exploratory Visualization, Proceedings*, pages 93 – 104.
- [Ankerst et al., 1996] Ankerst, M., Keim, D. A., and Kriegel, H.-P. (1996). Circle segments: A Technique for Visually Exploring Large Multidimensional Data Sets. *Visualization, Hot Topic Session, San Francisco, CA (Vis '96)*.
- [Baldonado et al., 2000] Baldonado, M. Q. W., Woodruff, A., and Kuchinsky, A. (2000). Guidelines for Using Multiple Views in Information Visualization. In *AVI '00: Proceedings of the working conference on Advanced visual interfaces*, pages 110–119, New York, NY, USA. ACM Press.
- [Barlow and Stuart, 2004] Barlow, N. and Stuart, L. (2004). Animator: A Tool for the Animation of Parallel Coordinates. *Eighth International Conference on Information Visualisation (InfoVis '04)*, pages 725 – 730.
- [Baving and Schmidt, 201] Baving, L. and Schmidt, M. H. (201). Evaluated Treatment Approaches in Child and Adolescent Psychiatry II. *Zeitschrift für Kinder- und Jugendpsychiatrie und Psychotherapie*, 29:206 – 220.
- [Beck et al., 1979] Beck, A. T., Rush, A. J., Shaw, B. F., and Emery, G. (1979). *Cognitive Therapy of Depression*. The Guilford Press, NY. ISBN 0-89862-919-5.
- [Beddow, 1990] Beddow, J. (1990). Shape Coding of Multidimensional Data on a Microcomputer Display. In *Proceedings of the First IEEE Conference on Visualization*.
- [Beisel and Leibl, 2001] Beisel, S. and Leibl, C. (2001). *Stationäre Verhaltenstherapie bei Essstörungen*. Georg Thieme Verlag, Stuttgart.
- [Benjafield, 1992] Benjafield, J. G. (1992). *Cognition*. Prentice-Hall International Editions.

- [Benninghoven and Liebeck, 1997] Benninghoven, D. and Liebeck, H. (1997). *Ambulante kognitive Verhaltenstherapie der Eßstörungen*. Thieme, Stuttgart.
- [Benyon and Imaz, 1999] Benyon, D. and Imaz, M. (1999). Metaphors and Models: Conceptual Foundations of Representations in Interactive Systems Development. *Human-Computer Interaction, Lawrence Erlbaum Associates, Inc*, 14:159 – 189.
- [Bertini et al., 2005] Bertini, E., Dell’Aquila, L., and Santucci, G. (2005). SpringView: Cooperation of Radviz and Parallel Coordinates for View Optimization and Clutter Reduction. *Coordinated & Multiple Views in Exploratory Visualization 2005 (CMV 2005)*, pages 22–29.
- [Bertini and Santucci, 2005] Bertini, E. and Santucci, G. (2005). Improving 2D Scatterplots Effectiveness through Sampling, Displacement, and User Perception. In *IV ’05: Proceedings of the Ninth International Conference on Information Visualisation (IV’05)*, pages 826–834, Washington, DC, USA. IEEE Computer Society.
- [Borg and Groenen, 1997] Borg, I. and Groenen, P. (1997). *Modern Multidimensional Scaling: Theory and Applications*. Springer Series in Statistics, Springer-Verlag: New York.
- [Bowers, 2001] Bowers, W. A. (2001). Basic Principles for Applying Cognitive-Behavioral Therapy in Anorexia Nervosa. *Psychiatric Clinics of North America*, 24:293 – 303.
- [Braddick, 1980] Braddick, O. J. (1980). Low-level and High-level Processes in Apparent Motion. *Philosophical Transactions of the Royal Society of London.*, B(209):137–151.
- [Brodie et al., 1992] Brodie, K. W., Carpenter, L. A., and Earnshaw, R. A. (1992). *Scientific Visualization: Techniques and Applications*. Springer Verlag, Berlin, Heidelberg.
- [Brown and McGregor, 2000] Brown, J. and McGregor, A. (2000). Network performance visualization: Insight through animation. In *Proc. of PAM2000: Passive and Active Measurement Workshop, Hamilton, New Zealand*.
- [Brunsdon et al., 1998] Brunsdon, C., Fotheringham, A. S., and Charlton, M. E. (1998). An Investigation of Methods for Visualising Highly Multivariate Datasets. *Case Studies of Visualization in the Social Sciences. Joint Information Systems Committee / ESRC*, pages 55–80.
- [Buziek and Dransch, 2000] Buziek, G. and Dransch, D. (2000). *Dynamische Visualisierung*. Springer-Verlag.
- [Card et al., 1998] Card, S., MacKinlay, J., and Shneiderman, B. (1998). *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers, San Francisco.
- [Card et al., 1999] Card, S., Mackinlay, J. D., and Shneiderman, B. (1999). *Readings in Information Visualization*. Morgan Kaufman. Chapter 1.
- [Chambers et al., 1983] Chambers, J. M., Cleveland, W. S., Kleiner, B., and Tukey, P. A. (1983). *Graphical Methods for Data Analysis*. Chapman and Hall, New York.
- [Chen, 2000] Chen, C. (2000). Empirical Evaluation of Information Visualizations: An Introduction. *Int. J. Human-Computer Studies*, 53(5):631–635.
- [Chen, 2004] Chen, C. (2004). *Information Visualization. Beyond the Horizon*. Springer, London, Berlin, Heidelberg.
- [Chen, 2005] Chen, C. (2005). Top 10 Unsolved Information Visualization Problems. *Computer Graphics and Applications, IEEE*, 25(4):12–16.
- [Chernoff, 1973] Chernoff, H. (1973). The Use of Faces to Represent Points in k-dimensional Space Graphically. *Journal of the American Statistical Association*, 68:361–368.

- [Chernoff and Rizvi, 1975] Chernoff, H. and Rizvi, M. H. (1975). Effect on Classification Error of Random Permutations of Features in Representing Multivariate Data by Faces. *Journal of American Statistical Association*, 70:548–554.
- [Chi, 2000] Chi, E. H. (2000). A Taxonomy of Visualization Techniques Using the Data State Reference Model. In *INFOVIS '00: Proceedings of the IEEE Symposium on Information Visualization 2000*, pages 69–76, Washington, DC, USA. IEEE Computer Society.
- [Chittaro, 2001] Chittaro, L. (2001). Information visualization and its application to medicine. *Artificial Intelligence in Medicine Journal, Special Issue on Information Visualization in Medicine*, 22:81–88.
- [Chittaro et al., 2002] Chittaro, L., Combi, C., and Trapasso, G. (2002). Visual Data Mining of Clinical Databases: An Application to the Hemodialytic Treatment based on 3D Interactive Bar Charts. In *Proceedings of VDM 2002: 2nd International Workshop on Visual Data Mining*.
- [Chuah and Eick, 1997a] Chuah, M. C. and Eick, S. G. (1997a). Glyphs for Software Visualization. In *5th International Workshop on Program Comprehension (WPC'97)*, pages 183–191.
- [Chuah and Eick, 1997b] Chuah, M. C. and Eick, S. G. (1997b). Managing Software with New Visual Representations. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis97)*, pages 30–37.
- [Chuah and Eick, 1998] Chuah, M. C. and Eick, S. G. (1998). Information Rich Glyphs for Software Management Data. *IEEE Comput. Graph. Appl.*, 18(4):24–29.
- [Cierpka and Frevert, 1995] Cierpka, M. and Frevert, G. (1995). *Die Familienbögen (FB). Ein Inventar zur Einschätzung von Familienfunktionen*. Hogrefe, Göttingen.
- [Craft and Cairns, 2005] Craft, B. and Cairns, P. (2005). Beyond Guidelines: What Can We Learn from the Visual Information Seeking Mantra? *Ninth International Conference on Information Visualisation, 2005 Proceedings.*, pages 110 – 118.
- [Craig et al., 2005] Craig, P., Kennedy, J., and Cumming, A. (2005). Coordinated Parallel Views for the Exploratory Analysis of Microarray Time-course Data. In *Information Visualisation, 2005 Proceedings. Ninth International Conference on*.
- [Crawford and Fall, 1990] Crawford, S. L. and Fall, T. C. (1990). Projection Pursuit Techniques for Visualizing High-Dimensional Data Sets. *Visualization in Scientific Computing, (G.M. Nielson and B. Shriver, eds.)*, IEEE Computer Society Press, pages 94 – 108.
- [Cui et al., 2006] Cui, Q., Ward, M. O., and Rundensteiner, E. A. (2006). Enhancing Scatterplot Matrices for Data with Ordering or Spatial Attributes. In Erbacher, R. F., Roberts, J. C., Grünhagen, M. T., and Bülthner, K., editors, *Proceedings of SPIE – Volume 6060*, volume 6060, page 11.
- [Daassi et al., 2000] Daassi, C., Dumas, M., Fauvet, M., Nigay, L., and Scholl, P. (2000). Visual Exploration of Temporal Object Databases. In *Proceedings of 16 iemes Journées Bases de Données Avancées (BDA, French Conference on Databases)*, pages 159–178.
- [Daassi et al., 2004] Daassi, C., Nigay, L., and Fauvet, M. (2004). Visualization Process of Temporal Data. In Galindo, F., Takizawa, M., and Traummüller, R., editors, *15th International Conference on Database and Expert Systems Applications (DEXA04)*, volume 3180. Springer-Verlag Heidelberg.
- [Dalton and Eichenberg, 1998] Dalton, R. J. and Eichenberg, R. C. (1998). *Citizen Support for Policy Integration. European Integration and Supranational Governance*. Oxford University Press.
- [Davidson et al., 1998] Davidson, L., Lambert, S., and McGlashan, T. H. (1998). *Psychotherapeutic and Cognitive-Behavioral Treatments for Schizophrenia: Developing a Disorder-Specific Form of Psychotherapy for Persons with Psychosis*. Wiley.
- [Dember and Earl, 1957] Dember, W. N. and Earl, R. W. (1957). Analysis of Exploratory, Manipulatory and Curiosity Behavior. *Psychologic Review*, (64):91ff.

- [Demsar et al., 2005] Demsar, J., Leban, G., and Zupan, B. (2005). FreeViz - An Intelligent Visualization Approach for Class-Labeled Multidimensional Data Sets. *Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP) Workshop*, pages 61–66.
- [deZwaan and Karwautz, 1996] deZwaan, M. and Karwautz, A. (1996). Therapie von Essstörungen: Überblick über Befunde kontrollierter Psycho- und Pharmakotherapiestudien. *Psychotherapeut*, 41:275 – 287.
- [Döpfner et al., 1998] Döpfner, M., Plück, J., Bölte, S., Lenz, K., Melchers, P., and Heim, K. (1998). *Fragebogen für Jugendliche. Deutsche Bearbeitung des Youth Self-Report (YSR)*. Arbeitsgruppe Kinder-, Jugend- und Familiendiagnostik.
- [Dransch, 1997] Dransch, D. (1997). *Computer-Animation in der Kartographie: Theorie und Praxis*. Springer.
- [Eades, 1984] Eades, P. (1984). A Heuristic for Graph Drawing. *Congressus Numerantium*, 42:149–160.
- [Eades and Lin, 2000] Eades, P. and Lin, X. (2000). Spring algorithms and symmetry. *Theoretical Computer Science*, 240(2):379–405.
- [Eisen et al., 1998] Eisen, M. B., Spellman, P. T., Brownagger, P. O., and Botstein, D. (1998). Cluster Analysis and Display of Genome-Wide Expression Patterns. *Proceedings of the National Academy of Sciences of the United States of America*, 95(25):14863 – 14868.
- [Ericson et al., 2005] Ericson, D., Johansson, J., and Cooper, M. (2005). Visual Data Analysis using Tracked Statistical Measures within Parallel Coordinate Representations. In *Coordinated and Multiple Views in Exploratory Visualization, 2005. (CMV 2005). Proceedings. Third International Conference on*, pages 42 – 53.
- [Fairburn, 1997] Fairburn, C. G. (1997). *Eating Disorders*. Oxford: University Press.
- [Feiner and Beshers, 1990] Feiner, S. K. and Beshers, C. (1990). Worlds within Worlds: Metaphors for Exploring n-Dimensional Virtual Worlds. In *UIST '90: Proceedings of the 3rd annual ACM SIGGRAPH symposium on User interface software and technology*, pages 76–83, New York, NY, USA. ACM Press.
- [Filliben, 2006a] Filliben, J. J. (2006a). Nist/sematech e-handbook of statistical methods. URL, <http://www.itl.nist.gov/div898/handbook/eda/section1/eda11.htm>. viewed at: June 12, 2006.
- [Filliben, 2006b] Filliben, J. J. (2006b). Nist/sematech e-handbook of statistical methods - graphical techniques: Alphabetic. viewed at: <http://www.itl.nist.gov/div898/handbook/eda/section3/eda33.htm>. 12.
- [Fodor, 2002] Fodor, I. K. (2002). A Survey of Dimension Reduction Techniques. Technical report, Lawrence Livermore National Lab., CA (US).
- [Friedman and Tukey, 1974] Friedman, J. and Tukey, . (1974). A Projection Pursuit Algorithm for Exploratory Data Analysis. *IEEE Trans. Computers*, 23:881 – 890.
- [Fua et al., 1999] Fua, Y.-H., Ward, M. O., and Rundensteiner, E. A. (1999). Hierarchical Parallel Coordinates for Exploration of Large Datasets. In *VIS '99: Proceedings of the conference on Visualization '99*, pages 43 – 50, Los Alamitos, CA, USA. IEEE Computer Society Press.
- [Furnas, 1986] Furnas, G. W. (1986). Generalized Fisheye Views. In *CHI '86: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 16–23, New York, NY, USA. ACM Press.
- [Fydrich et al., 2001] Fydrich, T., Bodem, M., Schmitz, B., Bachmann, M., and Mestel, R. (2001). *Beck-Inventar zu Kognitiven Schemata (B-IKS)*. Hogrefe, Göttingen.
- [Gabel, 1998] Gabel, M. J. (1998). *Interests and Integration. Market Liberalization, Public Opinion, and European Union*. The University of Michigan Press.

- [Garner and Bemis, 1985] Garner, D. M. and Bemis, K. M. (1985). *Cognitive Therapy for Anorexia Nervosa*. Guilford Press, New York.
- [Garner and Garfinkel, 1979] Garner, D. M. and Garfinkel, P. E. (1979). The Eating Attitudes Test: an Index of the Symptoms of Anorexia Nervosa. *Psychological Medicine*, 10:647 – 656.
- [Garner et al., 1983] Garner, D. M., Olmstead, M. P., and Polivy, J. (1983). Development and Validation of a Multidimensional Eating Disorder Inventory for Anorexia Nervosa and Bulimia. *International Journal of Eating Disorders*, 2:15 – 34.
- [Gee et al., 2005] Gee, A. G., Yu, M., and Grinstein, G. G. (2005). Dynamic and Interactive Dimensional Anchors for Spring-Based Visualizations. Technical report, Computer Science, University of Massachusetts Lowell.
- [Gibson, 1950] Gibson, J. J. (1950). *The Perception of the Visual World*. Houghton Muffin.
- [Gibson, 1986] Gibson, J. J. (1986). *The Ecological Approach to Visual Perception*. Lawrence Erlbaum Associates.
- [Harris, 1999] Harris, R. L. (1999). *Information Graphics: A Comprehensive Illustrated Reference*. Oxford University Press.
- [Hauser et al., 2002] Hauser, H., Lederman, F., and Doleisch, H. (2002). Angular Brushing of Extended Parallel Coordinates. *IEEE Symposium on Information Visualization, INFOVIS 2002*, pages 127 – 130.
- [Hautzinger et al., 1995] Hautzinger, M., Bailer, M., Worall, H., and Keller, F. (1995). *Beck-Depressions-Inventar (BDI)*. Huber, Bern.
- [Havre et al., 1999] Havre, S., Hetzler, B., and Nowell, L. (1999). ThemeRiver™: In Search of Trends, Patterns, and Relationships. In *Proceedings of IEEE Symposium on Information Visualization (InfoVis99)*. IEEE.
- [Havre et al., 2000] Havre, S., Hetzler, E., and Nowell, L. (2000). ThemeRiver: Visualizing Theme Changes Over Time. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis00)*, pages 115–123, Salt Lake City, USA.
- [Havre et al., 2002] Havre, S., Hetzler, E., Whitney, P., and Nowell, L. (2002). ThemeRiver: Visualizing Thematic Changes in Large Document Collections. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20.
- [Hershenson, 1964] Hershenson, M. (1964). Visual Discrimination in the Human Newborn. *Journal of Comparative and Physiological Psychology*, (58):270–275.
- [Herzog, 2004] Herzog, H. (2004). Multiple view framework for exploring highly structured data. Master thesis, Vienna University of Technology, Vienna.
- [Hinum, 2004] Hinum, K. (2004). Human centred Design for Graphical User Interfaces. Master’s thesis, Technical University of Vienna, Austria.
- [Hinum et al., 2005a] Hinum, K., Miksch, S., Aigner, W., Ohmann, S., Popow, C., Pohl, M., and Rester, M. (2005a). Gravi++: Interactive Information Visualization of Highly Structured Temporal Data. In *Workshop IDAMAP 2005 Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP 2005)*, pages 67 – 72, Aberdeen, Schottland.
- [Hinum et al., 2005b] Hinum, K., Miksch, S., Aigner, W., Ohmann, S., Popow, C., Pohl, M., and Rester, M. (2005b). Gravi++: Interactive Information Visualization to Explore Highly Structured Temporal Data. *Journal of Universal Computer Science*, 11(11):1792–1805.
- [Holtforth and Grawe, 2002] Holtforth, M. G. and Grawe, K. (2002). *Fragebogen zur Analyse motivationaler Schemata (FAMOS)*. Hogrefe, Göttingen.

- [Holzinger, 2005] Holzinger, A. (2005). Usability Engineering Methods for Software Developers. *Commun. ACM*, 48(1):71–74.
- [Horn et al., 1998] Horn, W., Popow, C., and Unterasinger, L. (1998). Metaphor Graphics to Visualize ICU Data over Time. In *Proceedings of Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP-98), Workshop Notes of the ECAI-98 Workshop*.
- [Hsu, 1990] Hsu, L. K. G. (1990). *Eating Disorders*. Guilford Press, New York.
- [Huber, 1985] Huber, P. J. (1985). Projection Pursuit. *The Annals of Statistics, JSTOR*, 13(2):435 – 475.
- [Imrich et al., 2003] Imrich, P., Mueller, K., Imre, D., Zelenyuk, D., and Zhu, W. (2003). Interactive Poster: 3D ThemeRiver. In *Poster Compendium of IEEE Information Visualization Symposium (InfoVis03)*, Seattle, USA.
- [Inselberg, 1985] Inselberg, A. (1985). The Plane with Parallel Coordinates. *The Visual Computer*, pages 69 – 92.
- [Inselberg, 1998] Inselberg, A. (1998). Visual Data Mining with Parallel Coordinates. *Computational Statistics*, 13(1):47 – 63.
- [Inselberg, 2006] Inselberg, A. (2006). Home of Parallel Coordinates. URL, <http://www.cs.tau.ac.il/~aisreal/>. viewed at: July 1, 2006.
- [Inselberg and Dimsdale, 1990] Inselberg, A. and Dimsdale, B. (1990). Parallel Coordinates: A Tool for Visualizing Multi-Dimensional Geometry. In *Proceedings of the 1st conference on Visualization (Vis '90)*, pages 361–378. IEEE Computer Society Press.
- [Inselberg et al., 1987] Inselberg, A., Reif, M., and Chomut, T. (1987). Convexity Algorithms in Parallel Coordinates. *J. ACM*, 34(4):765 – 801.
- [Jacobi and Thiel, 1996] Jacobi, C. and Thiel, A. (1996). *Kognitive Verhaltenstherapie bei Anorexia und Bulimia Nervosa*. Psychologie Verlags Union, Weinheim.
- [Jacobs, 1992] Jacobs, R. (1992). Possible Perceptual and Cognitive Advantages of Map Animation. University of Nebraska at Omaha.
- [Jansen, 2001] Jansen, A. (2001). Towards Effective Treatment of Eating Disorders: Nothing is as Practical as a Good Theory. *Behaviour Research and Therapy*, 39:1007 – 1022.
- [Jensen and Snodgrass, 1999] Jensen, C. S. and Snodgrass, R. T. (1999). Temporal Data Management. *IEEE Transactions on Knowledge and Data Engineering*, 11(1):36–44.
- [Johansson, 1973] Johansson, G. (1973). Visual Perception of Biological Motion and a Model for its Analysis. *Perception and Psychophysics*, 14(2):201–211.
- [Johansson et al., 2005] Johansson, J., Cooper, M., and Jern, M. (2005). 3-Dimensional Display for Clustered Multi-Relational Parallel Coordinates. *Information Visualisation, 2005 Proceedings. Ninth International Conference on*, pages 188 – 193.
- [Jolliffe, 1986] Jolliffe, I. T. (1986). *Principal Component Analysis*. Springer.
- [Kandogan, 2000] Kandogan, E. (2000). Star Coordinates: A Multidimensional Visualization Technique with Uniform Treatment of Dimensions. In *Proceedings of the IEEE Information Visualization Symposium, Late Breaking Hot Topics*.
- [Karwautz et al., 2001] Karwautz, A., Vökl-Kernstock, S., Nobis, G., Hafferl-Gattermayer, A., Wöber-Bingöl, C., and Friedrich, M. H. (2001). Characteristics of Self-Regulation in Adolescent Patients with Anorexia Nervosa. *British Medical Journal*, 74:101 – 114.

- [Keim, 2002] Keim, D. A. (2002). Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8.
- [Keim and Kriegel, 1995] Keim, D. A. and Kriegel, H.-P. (1995). VisDB: A System for Visualizing Large Databases, System Demonstration. *Proceedings ACM SIGMOD International Conference on Management of Data, San Jose, CA*.
- [Keim and Kriegel, 1996] Keim, D. A. and Kriegel, H.-P. (1996). Visualization Techniques for Mining Large Databases: A Comparison. *IEEE Transactions on Knowledge and Data Engineering*, 8(6):923–938.
- [Keim et al., 1995] Keim, D. A., Kriegel, H.-P., and Ankerst, M. (1995). Recursive Pattern: A Technique for Visualizing Very Large Amounts of Data. *Visualization '95. Proceedings., IEEE Conference on Visualization*, pages 279 – 286, 463. Keim proposes to use a recursive pattern to arrange the pixels in a pixel based visualization.
- [Koffka, 1935] Koffka, K. (1935). *Principles of Gestalt Psychology*. Harcourt (New York).
- [Kohn et al., 1999] Kohn, L., Corrigan, J., and Donaldson, M. (1999). *To Err Is Human*. National Academy Press, Washington, D.C.
- [Kohonen, 2001] Kohonen, T. (2001). *Self-Organizing Maps, 3rd edition*. Springer, Berlin, Heidelberg.
- [Korte, 1915] Korte, A. (1915). Kinematoskopische Untersuchungen. *Zeitschrift für Psychologie*, 72:193ff.
- [Kosara, 2001] 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, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria.
- [Kosara et al., 2004] Kosara, R., Bendix, F., and Hauser, H. (2004). TimeHistograms for Large, Time-Dependent Data. In Deussen, O., Hansen, C., Keim, D., and Saupe, D., editors, *Proceedings of Joint Eurographics - IEEE TCVG Symposium on Visualization (VisSym04)*, pages 45–54.
- [Kosara et al., 2006] Kosara, R., Bendix, F., and Hauser, H. (2006). Parallel Sets: Interactive Exploration and Visual Analysis of Categorical Data. *Transactions on Visualization and Computer Graphics (TVCG)*, 12(4):558–568.
- [Koussoulakou and Kraak, 1992] Koussoulakou, A. and Kraak, M. J. (1992). Spatio-temporal Maps and Cartographic Communication. *The Geographic Journal*, 29:101–108.
- [Kritzinger, 2003] Kritzinger, S. (2003). The Role of the Nation State on Individual Support for the European Union. *European Union Politics*, 4(2):219 – 241.
- [Kritzinger, 2005] Kritzinger, S. (2005). European identity building from the perspective of efficiency. *Comparative European Politics*, 3:50 – 75.
- [Kuniavsky, 2003] Kuniavsky, M. (2003). *Observing the User Experience: A Practitioner's Guide to User Research*. Morgan Kaufmann Series in Interactive Technologies. Morgan Kaufmann.
- [Kupfer et al., 2001] Kupfer, J., Brosig, B., and Brähler, E. (2001). *Toronto-Alexithymie-Skala-26 (TAS-26)*. Hogrefe, Göttingen.
- [Lakoff and La, 1980] Lakoff, G. and La, M. J. (1980). *Metaphors We Live By*. The University of Chicago Press. ISBN: 0226468011.
- [Lampert and Davidson, 1997] Lampert, S. and Davidson, L. (1997). Cognitive-behavioral treatment of delusions in the context of community-based care: a case study. In *Manuscript under review*.
- [Lanzenberger, 2003] Lanzenberger, M. (2003). *The Interactive Stardates - An Information Visualization Technique Applied in a Multiple View System*. PhD thesis, Vienna University of Technology, Faculty of Science and Informatics.

- [Lanzenberger et al., 2003] Lanzenberger, M., Miksch, S., and Pohl, M. (2003). The Stardimates - Visualizing Highly Structured Data. In *Proceedings of the Seventh International Conference on Information Visualization (InfoVis '03)*, pages 47–52. IEEE Computer Society.
- [Lanzenberger et al., 2005] Lanzenberger, M., Miksch, S., and Pohl, M. (2005). Exploring Highly Structured Data A Comparative Study of Stardimates and Parallel Coordinates. In *Information Visualization, 2005. Proceedings. Ninth International Conference on*.
- [Latner and Wilson, 2000] Latner, J. D. and Wilson, G. T. (2000). Cognitive-Behavioral Therapy and Nutritional Counseling in the Treatment of Bulimia Nervosa and Binge Eating. *Eating Behaviors*, 1:3–21.
- [LeBlanc et al., 1990] LeBlanc, J., Ward, M. O., and Wittels, N. (1990). Exploring N-Dimensional Databases. In *VIS '90: Proceedings of the 1st conference on Visualization '90*, pages 230–237, Los Alamitos, CA, USA. IEEE Computer Society Press.
- [Leung and Apperley, 1994] Leung, Y. K. and Apperley, M. D. (1994). A Review and Taxonomy of Distortion-Oriented Presentation Techniques. *ACM Trans. Comput.-Hum. Interact.*, 1(2):126–160.
- [Levkovitz, 1991] Levkovitz, H. (1991). Color Icons-Merging Color and Texture Perception for Integrated Visualization of Multiple Parameters. In *Visualization, 1991. Visualization '91, Proceedings., IEEE Conference on*.
- [Lind and Kjellin, 2005] Lind, M. and Kjellin, A. (2005). Faster is better: Optimal speed of animated visualizations for decision makers. In *IV '05: Proceedings of the Ninth International Conference on Information Visualisation (IV'05)*, pages 896–900, Washington, DC, USA. IEEE Computer Society.
- [Lutz, 1999] Lutz, R. (1999). Marburger Untersuchungs-Instrumentarium (MUM).
- [Lutz et al., 2006] Lutz, W., Kritzinger, S., and Skirbekk, V. (2006). The Demography of Growing European Identity. *Science*, 314:425.
- [Mack and Nielsen, 1993] Mack, R. and Nielsen, J. (1993). Usability Inspection Methods: Report on a workshop held at CHI92, Monterey, CA, May 3 and 4, 1992. *SIGCHI Bull.*, 25(1):28–33.
- [Mackinlay et al., 1991] Mackinlay, J. D., Robertson, G. G., and Card, S. K. (1991). The Perspective Wall: Detail and Context Smoothly Integrated. In *Proceedings of ACM Conference on Human Factors in Computing Systems (CHI91)*, Information Visualization, pages 173–179.
- [Macromedia, 2006a] Macromedia (2006a). Flash player distribution. URL, <http://www.adobe.com/products/flash/flashpro/productinfo/buy/>. viewed at: Oktober 13, 2006.
- [Macromedia, 2006b] Macromedia (2006b). Flash professional. URL, <http://www.adobe.com/products/flash/flashpro/>. viewed at: Oktober 13, 2006.
- [Marcus, 2002] Marcus, A. (2002). Dare We Define User-Interface Design? *interactions*, 9(5):19–24.
- [Marr, 1982] Marr, D. (1982). *Vision*. Freeman Publishers.
- [Mattejat and Remschmidt, 1999] Mattejat, F. and Remschmidt, H. (1999). *Fragebogen zur Beurteilung der Behandlung (FBB)*. Hogrefe, Göttingen.
- [Mattejat and Scholz, 1994] Mattejat, F. and Scholz, M. (1994). *Subjektives Familienbild (SFB)*. Hogrefe, Göttingen.
- [Matthews and Roze, 1997] Matthews, G. and Roze, M. (1997). Worm Plots. *Computer Graphics and Applications, IEEE*, 17(6):17–20.
- [McGinn and Picking, 2003] McGinn, J. and Picking, R. (2003). The Argument-as Metaphor in Decision-making Visualisation. volume Seventh International Conference on Information Visualization (InfoVis '03), pages 596–599. IEEE Computer Society.

- [Meichenbaum, 1977] Meichenbaum, D. (1977). *Cognitive-behavior modification: an integrative approach*.
- [Merriam-Webste, 2006] Merriam-Webste (2006). Definition of Insight. URL, <http://www.m-w.com/dictionary/insight>. viewed at: October 26, 2006.
- [Miller and Wegman, 1991] Miller, J. J. and Wegman, E. J. (1991). Construction of line densities for Parallel Coordinate Plots. *Computing and Graphics in Statistics, Springer-verlag*, pages 107 – 123.
- [Mintz et al., 1997] Mintz, D., Fitz-Simons, T., and Wayland, M. (1997). Tracking Air Quality Trends with SAS/GRAPH. In *Proceedings of the 22nd Annual SAS User Group International Conference (SUGI97)*, pages 807–812.
- [Moere, 2004] Moere, A. V. (2004). Time-Varying Data Visualization Using Information Flocking Boids. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis04)*, pages 10–12.
- [Monmonier, 1990] Monmonier, M. (1990). Strategies for the Visualization Geographic Time-Series Data. *Cartographica*, 27(1):30–45.
- [Morgan and Hayward, 1988] Morgan, H. G. and Hayward, A. E. (1988). Clinical Assessment of Anorexia Nervosa. *The Morgan-Russell outcome assessment schedule. British Journal of Psychiatry*, 152:367 – 371.
- [Nakakoji et al., 2001] Nakakoji, K., Takashima, A., and Yamamoto, Y. (2001). Cognitive Effects of Animated Visualization in Exploratory Visual Data Analysis. In *IV '01: Proceedings of the Fifth International Conference on Information Visualisation (IV'01)*, pages 77–84. IEEE Computer Society.
- [Nielsen and Mack, 1994] Nielsen, J. and Mack, R. (1994). *Usability Inspection Methods*. John Wiley & Sons, Inc.
- [Noirhomme-Fraiture, 2002] Noirhomme-Fraiture, M. (2002). Visualization of Large Data Sets: The Zoom Star Solution. *International Electronic Journal of Symbolic Data Analysis*, 0(0).
- [North, 2006] North, C. (2006). Toward measuring visualization insight. *IEEE Computer Graphics and Applications*, 26(3):6–9.
- [Novotny, 2004] Novotny, M. (2004). Visually Effective Information Visualization of Large Data. *Proceedings of th 8th Central European Seminar on Computer Graphics (CESCG 2004)*, pages 41 – 48.
- [OECD, 2006] OECD (2006). Pisa. URL, <http://www.pisa.oecd.org/>. viewed at: September 27, 2006.
- [of Standards and SEMATECH, 2006] of Standards, N. I. and SEMATECH, T. N. . (2006). Nist/sematech e-handbook of statistical methods. URL, <http://www.itl.nist.gov/div898/handbook/>. viewed at: July 10, 2006.
- [Ohmann et al., 2005a] Ohmann, S., Popow, C., Schuch, B., Karwautz, A., Lanzenberger, M., Herzog, H., Hinum, K., and Miksch, S. (2005a). Evaluation of a Manual Based Cognitive Behavioral Group Therapy (CBT) Program for Adolescent Girls with Anorexia Nervosa (AN) and their Families. In *ECED European Council on Eating Disorders, Innsbruck, Book of Abstracts, ECED*, page 35.
- [Ohmann et al., 2005b] Ohmann, S., Popow, C., Schuch, B., Karwautz, A., Lanzenberger, M., Herzog, H., Hinum, K., and Miksch, S. (2005b). Evaluation of a Manual Based Cognitive Behavioral Group Therapy (CBT) Program for Adolescent Girls with Anorexia Nervosa (AN) and their Families. In *EDRS, Toronto, Poster*.
- [Ohmann et al., 2005c] Ohmann, S., Popow, C., Schuch, B., Karwautz, A., Lanzenberger, M., Herzog, H., Hinum, K., and Miksch, S. (2005c). Kognitive Verhaltenstherapie bei Mädchen mit Anorexia Nervosa (AN) und deren Familien – Evaluation eines Gruppentherapiemanuals. In *Kongress Essstoerungen, 13. Internationale Wissenschaftl. Tagung, Innsbruck, Book of Abstracts, Netzwerk Essstoerungen*, page 27.

- [Ohmann et al., 2004] Ohmann, S., Popow, C., Schuch, B., Karwautz, A., Lanzenberger, M., Herzog, H., and Miksch, S. (2004). Manual Based Cognitive Behavioral Group Therapy (CBT) for Adolescent Girls with Anorexia Nervosa (AN) and their Families. In Remschmidt, H. and Belfer, M., editors, *16th World Congress of the IACAPAP, International Association for Child and Adolescent Psychiatry and Allied Professions, Book of Abstracts*, page 313.
- [Olsen et al., 1993] Olsen, K. A., Korfhage, R. R., Sochats, K. M., Spring, M. B., and Williams, J. G. (1993). Visualization of a Document Collection: The VIBE System. *Information Processing & Management*, 29(1):69–81.
- [Osztoivits, 2007] Osztoivits, M. (forthcoming 2007). GenEdit - Generisches Eingabetool für Fragebögen am Beispiel der Anorexiestudie. Master's thesis, Technische Universität Wien.
- [Pickett, 1970] Pickett, R. M. (1970). *Picture Processing and Psycho-Pictorics*, chapter Visual Analyses of Texture in the Detection and Recognition of Objects. Academic Press, New York.
- [Pickett and Grinstein, 1988] Pickett, R. M. and Grinstein, G. G. (1988). Iconographic Displays For Visualizing Multidimensional Data. In *Proc. IEEE Conf. on Systems, Man and Cybernetics, IEEE Press*, pages 514–519.
- [Pillat et al., 2005] Pillat, R., Valiati, E., and Freitas, C. (2005). Experimental Study on Evaluation of Multidimensional Information Visualization Techniques. In *CLIH05: Proceedings of the 2005 Latin American Conference on Human-computer Interaction*, pages 20–30, New York, NY, USA. ACM Press.
- [Plaisant, 2004] Plaisant, C. (2004). The Challenge of Information Visualization Evaluation. In *Proceedings of Conference on Advanced Visual Interfaces (AVI04)*, pages 109–116.
- [Plaisant et al., 1996] Plaisant, C., Milash, B., Rose, A., Widoff, S., and Shneiderman, B. (1996). LifeLines: Visualizing Personal Histories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 221–ff. ACM Press.
- [Plaisant et al., 1998] Plaisant, C., Mushlin, R., Snyder, A., Li, J., Heller, D., and Shneiderman, B. (1998). LifeLines: Using Visualization to Enhance Navigation and Analysis of Patient Records. In *Proceedings of the 1998 American Medical Informatic Association Annual Fall Symposium*, pages 76–80.
- [Plaisant and Rose, 1996] Plaisant, C. and Rose, A. (1996). Exploring LifeLines to Visualize Patient Records. Technical Report CS-TR-3620, University of Maryland, College Park.
- [Pözlzbauer et al., 2005] Pözlzbauer, G., Dittenbach, M., and Rauber, A. (2005). A visualization technique for self-organizing maps with vector fields to obtain the cluster structure at desired levels of detail. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'05)*, pages 1558–1563, Montreal, Canada. IEEE Computer Society.
- [Porzelius et al., 1999] Porzelius, L. K., Berel, S., and Howard, C. (1999). *Cognitive Behavior Therapy - Handbook of Comparative Interventions for Adult Disorders (2nd ed.)*. Wiley, New York.
- [Pratt and Tschapek, 2003] Pratt, K. B. and Tschapek, G. (2003). Visualizing Concept Drift. *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining; Poster Session*, pages 735 – 740.
- [Preece et al., 2002] Preece, J., Rogers, Y., and Sharp, H. (2002). *Interaction Design*. John Wiley & Sons, Inc., New York, NY, USA.
- [Rachman, 1997] Rachman, S. (1997). *The Science and Practice of Cognitive Behaviour Therapy*, volume The Evolution of Cognitive Behaviour Therapy. Oxford: University Press.
- [Rao and Card, 1994] Rao, R. and Card, S. K. (1994). The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information. In *Proceedings of the ACM Conference Human Factors in Computing Systems (CHI '94)*, pages 318–322. ACM.

- [Rester, 2007] Rester, M. (forthcoming 2007). *Nachhaltige Evaluierung explorativer Informationsvisualisierung*. PhD thesis, Vienna University of Technology, Institute of Design and Assessment of Technology.
- [Rester et al., 2006a] Rester, M., Pohl, M., Banovic, S., Hinum, K., Miksch, S., Ohmann, S., and Popow, C. (2006a). Zum hohen Stellenwert einer Usability-Untersuchung in der Evaluierung einer interaktiven Informationsvisualisierung. In Kempter, G. and von Hellberg, P., editors, *Information nutzbar machen – Zusammenfassung der Beiträge zum Usability Day IV*, pages 143–147. Pabst Science Publishers, Lengerich, D.
- [Rester et al., 2005] Rester, M., Pohl, M., Hinum, K., Miksch, S., Ohmann, S., Popow, C., and Banovic, S. (2005). Assessing the Usability of an Interactive Information Visualization Method as the First Step of a Sustainable Evaluation. In *Vienna Usability Symposium: Empowering Software Quality: How can Usability Engineering reach these goals?, 1st Usability Symposium, HCI&UE Workgroup, Austrian Computer Society (OCG)*, volume 198, pages 31 – 43. Austrian Computer Society (OCG). 3-85403-198-X.
- [Rester et al., 2006b] Rester, M., Pohl, M., Hinum, K., Miksch, S., Popow, C., Ohmann, S., and Banovic, S. (2006b). Methods for the evaluation of an interactive infovis tool supporting exploratory reasoning processes. In Bertini, E., Plaisant, C., and Santucci, G., editors, *BELIV '06: Proceedings of the 2006 conference on BEyond time and errors: novel evaluation methods for Information Visualization*, pages 1–6, New York, NY, USA. ACM Press. pp. 32–37 in printed proceedings.
- [Roberts, 1998] Roberts, J. C. (1998). On Encouraging Multiple Views for Visualization. In Banissi, E., Khosrowshahi, F., and Sarfraz, M., editors, *Proceedings of the International Conference on Information Visualization (IV98)*, pages 8–14. IEEE Computer Society.
- [Roberts, 2000] Roberts, J. C. (2000). Multiple-View and Multiform Visualization. In Erbacher, R., Pang, A., Wittenbrink, C., and Roberts, J., editors, *Visual Data Exploration and Analysis VII, Proceedings of SPIE*, volume 3960, pages 176–185. IST and SPIE.
- [Robertson et al., 1991] Robertson, G. G., Mackinlay, J. D., and Card, S. K. (1991). Cone Trees: Animated 3D Visualizations of Hierarchical Information. In *CHI '91: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 189–194, New York, NY, USA. ACM Press.
- [Robertson and Ferrari, 1994] Robertson, P. and Ferrari, L. D. (1994). *Scientific Visualization - Advances and Challenges*, chapter Systematic Approaches to Visualization: Is a Reference Model Needed? Academic Press.
- [Robin et al., 1998] Robin, A. L., Gilroy, M., and Dennis, A. M. (1998). Treatment of Eating Disorders in Children and Adolescents. *Clinical Psychology Review*, 18:421 – 446.
- [Rock, 1985] Rock, I. (1985). *Vom visuellen Reiz vom Sehen zum Erkennen*. Spektrum-der-Wissenschaft-Verlagsgesellschaft Heidelberg.
- [Ronen, 1997] Ronen, T. (1997). *Cognitive Developmental Therapy with Children*. New York: Wiley.
- [Sarkar and Brown, 1992] Sarkar, M. and Brown, M. H. (1992). Graphical Fisheye Views of Graphs. In *CHI '92: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 83–91, New York, NY, USA. ACM Press.
- [Schmidberger, 1997] Schmidberger, M. (1997). *Regionen und europäische Legitimität. Der Einfluss des regionalen Umfeldes auf Bevölkerungseinstellungen zur EU*. Peter-Lang-Verlag.
- [Schumann and Müller, 2000] Schumann, H. and Müller, W. (2000). *Visualisierung - Grundlagen und allgemeine Methoden*. Springer Verlag.
- [Schwarzer and Jerusalem, 1999] Schwarzer, R. and Jerusalem, M. (1999). *Skalen zur Erfassung von Lehrern- und Schülermerkmalen. Dokumentation der psychometrischen Verfahren im Rahmen der wissenschaftlichen Begleitung des Modellversuchs Selbstwirksame Schulen*. Freie Universität Berlin.

- [Sears, 1970] Sears, F. W. (1970). *University Physics*. Addison Wesley Longman Publishing Co.
- [Shahar, 2000] Shahar, Y. (2000). Dimension of Time in Illness: An Objective View. *Annals of Internal Medicine*, 132:47–53.
- [Shimabukuro et al., 2004] Shimabukuro, M., Flores, E., de Oliveira, M., and Levkowitz, H. (2004). Coordinated Views to Assist Exploration of Spatio-Temporal Data: A Case Study. In *Proceedings of the 2nd International Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV04)*, pages 107–117. IEEE.
- [Shneiderman, 1992] Shneiderman, B. (1992). Tree Visualization with Tree-Maps: 2-d Space-Filling Approach. *ACM Trans. Graph.*, 11(1):92–99.
- [Shneiderman, 1996] Shneiderman, B. (1996). The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. *Proceedings of the IEEE Symposium on Visual Languages*, pages 336–343, Washington. IEEE Computer Society Press, pages 336 – 343.
- [Siirtola, 2000] Siirtola, H. (2000). Direct Manipulation of Parallel Coordinates. *Proceedings of the IEEE International Conference on Information Visualization*, pages 373 – 378.
- [Siirtola, 2005] Siirtola, H. (2005). The effect of data-relatedness in interactive glyphs. In *IV '05: Proceedings of the Ninth International Conference on Information Visualisation (IV'05)*, pages 869–876, Washington, DC, USA. IEEE Computer Society.
- [Soegaard, 2005] Soegaard, M. (2005). Gestalt principles of form perception. URL, http://www.interaction-design.org/encyclopedia/gestalt_principles_of_form_perception.html. viewed at: July 10, 2007.
- [Spence, 2001] Spence, R. (2001). *Information Visualization*. Addison-Wesley, ACM Press.
- [Squires, 1959] Squires, P. C. (1959). Topological Aspects of Apparent Visual Motion. *Psychologische Forschung*, 26:1ff.
- [Stangier et al., 1999] Stangier, U., Heidenreich, T., Berardi, A., Golbs, U., and Hoyer, J. (1999). Die Erfassung sozialer Phobie durch die Social Interaction Anxiety Scale (SIAS) und die Social Phobia Scale (SPS). *Zeitschrift für Klinische Psychologie*, 28(1):28 – 36.
- [StatSoft, 2003] StatSoft, I. (2003). Multidimensional scaling. URL, <http://www.statsoft.com/textbook/stmulasca.html>. viewed at: July 7, 2006.
- [Tognazzini, 1992] Tognazzini, B. (1992). *TOG on Interface*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- [Tominski et al., 2003a] Tominski, C., Abello, J., and Schumann, H. (2003a). Interactive Poster: Axes-Based Visualizations for Time Series Data. In *Poster Compendium of the IEEE Symposium on Information Visualization (InfoVis '03)*, pages 68–69. IEEE.
- [Tominski et al., 2003b] Tominski, C., Schulze-Wollgast, P., and Schumann, H. (2003b). Visualisierung zeitlicher Verläufe auf geografischen Karten. In *Proceedings GeoVis'2003*, pages 47–54, Hannover, Germany.
- [Tory and Möller, 2004] Tory, M. and Möller, T. (2004). Human Factors in Visualization Research. *Visualization and Computer Graphics, IEEE Transactions on*, 10(1):72–84.
- [Tufté, 1983] Tufté, E. R. (1983). *The Visual Display of Quantitative Information*. Graphics Press, Cheshire.
- [Tufté, 1990] Tufté, E. R. (1990). *Envisioning Information*. Graphics Press, Cheshire.
- [Tukey, 1977] Tukey, J. W. (1977). *Exploratory Data Analysis*. Addison-Wesley.
- [Ullmann, 1979] Ullmann, S. (1979). *The interpretation of visual motion*. Cambridge, MA: MIT Press.

- [Vallis, 1998] Vallis, T. M. (1998). *When the Going Gets Tough: Cognitive Therapy for the Severely Disturbed*. Wiley.
- [Vernon, 1974] Vernon, M. D. (1974). *Wahrnehmung und Erfahrung*. Klotz, Eschborn.
- [Vincent et al., 1996] Vincent, J., Moreno, R., Takala, J., Willatss, S., Mendonca, A. D., Bruining, H., Reinhart, C., Suter, P., and Thijs, L. (1996). The SOFA (Sepsis-related Organ Failure Assessment) Score to Describe Organ Dysfunction/Failure. *Intensive Care Med*, 22:707–710.
- [von Zerssen, 1975] von Zerssen, D. (1975). *Befindlichkeitsskala (Bf-S)*. Hogrefe, Göttingen.
- [Waldeck, 2005] Waldeck, C. (2005). Liquid 2D Scatter Space for File System Browsing. *Information Visualisation, 2005. Proceedings. Ninth International Conference on*, pages 451–456.
- [Ward, 1994] Ward, M. O. (1994). XmdvTool: Integrating Multiple Methods for Visualizing Multivariate Data. In *Proceedings of the conference on Visualization (Vis '94)*, pages 326–333. IEEE Computer Society Press.
- [Ward, 2002] Ward, M. O. (2002). A Taxonomy of Glyph Placement Strategies for Multidimensional Data Visualization. *Information Visualization*, 1(3/4):194–210.
- [Ware, 2000] Ware, C. (2000). *Information Visualization*. Morgan Kaufmann Publishers, San Francisco.
- [Wegman, 1990] Wegman, E. J. (1990). Hyperdimensional Data Analysis Using Parallel Coordinates. *Journal of the American Statistical Association*, 85(411):664 – 675.
- [Welsch and Liebmann, 2003] Welsch, N. and Liebmann, C. C. (2003). *Farben, Natur Technik Kunst*. Spektrum Akademischer Verlag.
- [Wertheimer, 1912] Wertheimer, M. (1912). Experimentelle Studien /"uber das Sehen von Bewegung. *Zeitschrift f/"ur Psychologie und Physiologie der Sinnesorgane*, 61:161–265.
- [Wessels, 1998] Wessels, B. (1998). *Development of Support: Diffusion or Demographic Replacement?* Oxford Scholarship Online Monographs.
- [Whittal and Zaretsky, 1996] Whittal, M. L. and Zaretsky, A. (1996). *Cognitive-Behavioral Strategies for the Treatment of Eating Disorders*. The Guilford Press, New York.
- [Willim, 1989] Willim, B. (1989). *Leitfaden der Computer Graphik*. Drei-R-Verlag, Berlin.
- [Wilson, 1999] Wilson, G. T. (1999). Cognitive Behavior Therapy for Eating Disorders: Progress and Problems. *Behaviour Research and Therapy*, 37:79 – 95.
- [Wilson and Fairburn, 1993] Wilson, G. T. and Fairburn, C. G. (1993). Cognitive Treatments for Eating Disorders. Special Section: Recent Developments in Cognitive and Constructivist Psychotherapies. *J Consult Clin Psychol*, 61:261 – 269.
- [Wyeld, 2005] Wyeld, T. G. (2005). 3D Information Visualisation: an Historical Perspective. In *Information Visualisation, 2005. Proceedings. Ninth International Conference on*. IEEE.
- [Yang et al., 2005] Yang, C.-C., Chiang, C.-C., Hung, Y.-P., and Lee, G. C. (2005). Visualization for high-dimensional data: VisHD. In *Proceedings of the Ninth International Conference on Information Visualisation, 2005*, pages 692 – 696.
- [Yi et al., 2005] Yi, J. S., Ponder, R. M., Stasko, J., and Jacko, J. A. (2005). Dust & Magnet: Interactive Visualization of Everyday Data. *Proceedings of the Eleventh Annual IEEE Symposium on Information Visualization (InfoVis 2005)*.

Curriculum Vitae

Adress

Klaus Alexander Hinum
Dieselgasse 5a/13
1100 Vienna, Austria
hinum@ifs.tuwien.ac.at

Date of Birth

December 6, 1977

Education

July 2004 - December 2006
Ph.D. studies in Information Science at the Vienna University of Technology

October 1996 - March 2004
M.S. studies in Computer Science at the Vienna University of Technology

Job Experience

July 2004 - December 2006
Project assistant (in2vis project) at the Institute of Software Technology & Interactive Systems

Publications

Please see <http://www.ifs.tuwien.ac.at/hinum/>