FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Empirical Evaluation of a Visualization Technique with Semantic Zoom

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Medizinische Informatik

eingereicht von

Stephan Hoffmann
Matrikelnummer 0325733

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung: Ao.Univ-Prof. Mag. Dr. Silvia Miksch
Mitwirkung: Univ.Ass. Dipl.-Ing. Dr. Wolfgang Aigner, Proj.Ass. Mag. Alexander Rind

Wien, 08.11.2011

(Unterschrift Verfasser) (Unterschrift Betreuung)

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Empirical Evaluation of a Visualization Technique with Semantic Zoom

MASTER'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Medical Informatics

by

Stephan Hoffmann
Registration Number 0325733

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Ao.Univ-Prof. Mag. Dr. Silvia Miksch
Assistance: Univ.Ass. Dipl.-Ing. Dr. Wolfgang Aigner, Proj.Ass. Mag. Alexander Rind

Vienna, 08.11.2011

(Signature of Author) (Signature of Advisor)

Erklärung zur Verfassung der Arbeit

Stephan Hoffmann
Stillfriedplatz 6, 1160 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum) (Unterschrift Verfasser)

Acknowledgements

First of all, I wish to thank my advisors Wolfgang Aigner, Alexander Rind, Silvia Miksch, and the the whole IEG team for their help and intense support throughout the thesis. In addition to a pleasant working atmosphere, they provided good advice and guidance whenever necessary, which I enjoyed very much and do not take for granted.

Furthermore, many thanks to all those who took the time and participated in the experiment; this work would not have been possible without their commitment.

Most importantly, I want to thank my family for their love and encouragement during all the years. I am really grateful that they gave me the opportunity to study at university and always supported me. In particular, I would like to thank my father Camille, Edmund, and Karl for doing a great job in extensively proofreading this thesis.

Finally, thanks to my girlfriend Heike for taking care of me and standing by me through these months and anybody else around me that supported me.

Abstract

This master thesis describes the evaluation of an interactive information visualization technique that is capable of displaying quantitative attributes (numeric values) of multivariate data over time and corresponding qualitative abstractions (interpretations) of the quantitative values (*SemTimeZoom*). The integration of interpretations and a-priori knowledge in the form of qualitative abstractions is especially useful in the medical domain. Vital parameters of patients can be analyzed using predefined domain knowledge and the resulting interpretations can be visualized together with raw numerical measurements.

The investigated visualization technique uses different visual representations of the data depending on the vertical display space of a single parameter and combines the quantitative and qualitative attributes of a parameter into one combined representation. The area-aware method to display different representations is called *semantic zooming*.

Although the developed visualization technique appears very promising, it has not yet been evaluated. Novel visualization techniques need to present measurable benefits to encourage more widespread adoption. To assess the effectiveness of this visualization technique, a comparative study was performed. The visualization technique that was used for the comparison is also capable of displaying raw quantitative values and qualitative abstractions but uses static and separate visual representations for quantitative and qualitative attributes of the data.

The comparative study was conducted by means of a controlled experiment that revealed faster completion times especially for more complex tasks involving comparison of quantitative values within specified qualitative categories in favor of the *SemTimeZoom* technique. All tasks that were used in the experiment involved the qualitative attributes of the data to evaluate the effectiveness for exploratory data analysis with qualitative abstractions.

It is generally acknowledged in the information visualization research field that it is necessary to evaluate visualization techniques, but the difficulties of conducting such evaluations still remain an issue. In the course of this study, evaluation functionality was integrated into the Java software prototypes that were used for the controlled experiment. A software library was built based on the evaluation functionality to facilitate future evaluation studies. Care has been taken to develop an easy-to-use, flexible and reusable software library that can be integrated into other prototypes that need to be evaluated. This thesis includes a detailed documentation of the structure and usage of the library.

Kurzfassung

Diese Diplomarbeit beschreibt die Evaluierung einer interaktiven Informationsvisualisierung (*SemTimeZoom*) von quantitativen Merkmalen zeitbezogener Daten mit mehreren Variablen und der zugehörigen qualitativen Abstraktionen (Interpretationen). Speziell im medizinischen Bereich ist es nützlich Interpretationen und a priori Kenntnisse über PatientInnendaten in die Darstellung der Daten zu integrieren. Die Analyse der Daten kann durch die Einbeziehung von Fachwissen und der daraus resultierenden Interpretationen der Daten unterstützt werden.

Diese Visualisierungstechnik verwendet verschiedene visuelle Repräsentationen für die Daten, abhängig vom vertikalen Platz, der für einen einzelnen Parameter zur Verfügung steht. Die quantitativen Daten und deren qualitativen Interpretationen werden gemeinsam in einer kombinierten Form dargestellt. Das Anpassen der Repräsentationen an die zur Verfügung stehende Darstellungsfläche wird *Semantic Zooming* genannt.

Obwohl diese Visualisierungstechnik sehr vielversprechend wirkt, wurde sie bis jetzt noch nicht evaluiert. Um eine weit verbreitete Benützung zu erreichen, müssen neue Visualisierungstechniken belegbare Vorteile präsentieren. Um die Effektivität von SemTimeZoom einschätzen zu können, wurde eine Vergleichsstudie mit einer anderen Visualisierungstechnik durchgeführt. Die Vergleichstechnik unterstützt ebenfalls die Darstellung von quantitativen Daten und zugehörige qualitative Abstraktionen, verwendet aber statische visuelle Repräsentationen und zeigt die quantitativen Werte getrennt von den qualitativen Abstraktionen.

Die Vergleichsstudie wurde mithilfe eines kontrollierten Experiments durchgeführt. Das Experiment zeigte, dass die Testpersonen speziell für komplexere Aufgaben, die den Vergleich von den quantitativen Werten innerhalb bestimmter qualitativer Levels beinhalten, mit der *SemTimeZoom* Technik deutlich weniger Zeit benötigten als mit der Vergleichstechnik.

Obwohl Wissenschafter die sich mit Informationsvisualisierung beschäftigen schon lange die Wichtigkeit von Evaluierungen ihrer Visualisierungstools erkannt haben, bleibt die Schwierigkeit der Durchführung solcher Studien ein wichtiges Thema. Im Zuge der Studie wurden die Software-Prototypen der Visualisierungstechniken um verschiedene Funktionalitäten für die Evaluierung erweitert. Um die zukünftige Durchführung von Evaluierungsstudien zu erleichtern wurde daraus eine wiederverwendbare und flexible Software-Bibliothek entwickelt, die in zu evaluierende Software-Prototypen integriert werden kann. Diese Software-Bibliothek und deren Verwendung wird in dieser Arbeit detailliert beschrieben.

Contents

CHAPTER 1

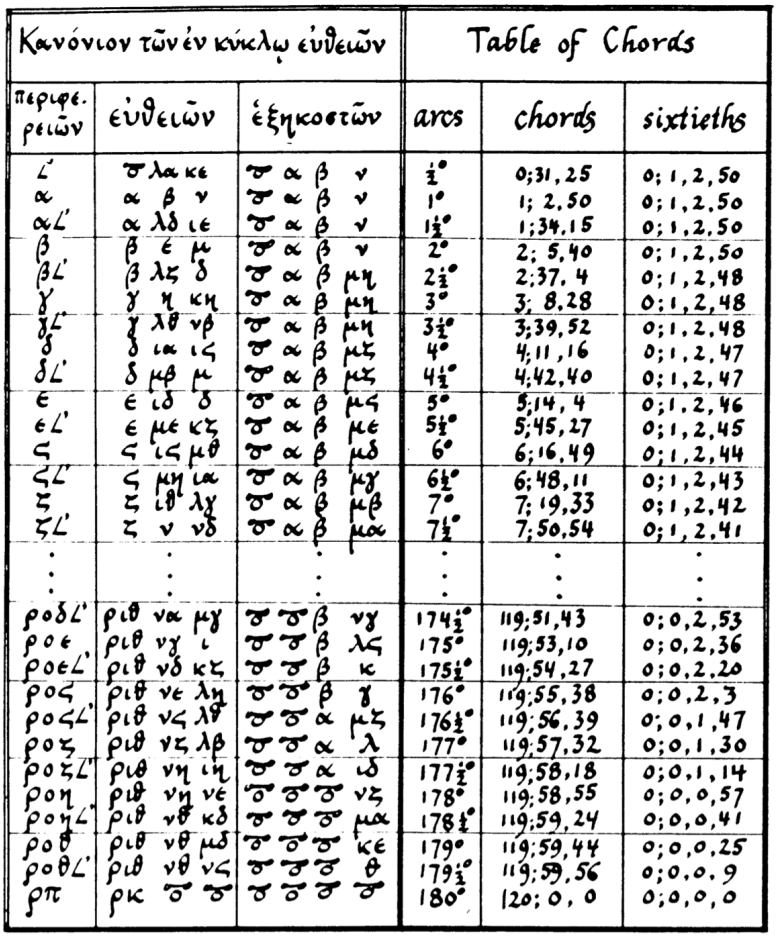
Introduction

"Visualization provides an interface between two powerful information processing systems—the human mind and the modern computer." [Gershon et al., 1998]

The term "I see!" stands for understanding something or having a sudden insight. This metaphor gives us a glimpse of the relationship between what we see and what we think. Using visual representations of information is very common to support the communication of knowledge and ideas, for example at schools or universities by the use of blackboards, overhead or powerpoint presentations.

The ability to think is extremely limited without external aids. Card et al. [1999] demonstrate the link between external perception and interior mental action by an example of mental arithmetic: a multiplication of a pair of two-digit numbers can be a difficult task without the use of pencil and paper. If the number of digits of the numbers increase, the task gets quickly impossible to do without the help of external aids or special techniques for mental multiplication. The problem is not the multiplication itself but to memorize the partial results in the multiplication process. The possibility to store partial results on paper relieves the human *working memory*. The visual working memory holds the visual objects of immediate attention, either external or mental images and is limited to a small number of objects or patterns [Ware, 2004]. The use of pencil and paper to write partial results in aligned columns converts an internal memory task into an external visual search task. This is one of the reasons why pattern recognition in data is enhanced by the use of traditional visual representations of quantitative data like line graphs, bar charts or scatter plots. Instead of reading the textual data representations and memorizing individual numeric values to find relevant patterns in the memorized objects, a visual representation that takes advantage of the human visual system simplifies the task by converting the memory tasks to visual search tasks.

Visual representations of data have a long history. A very early form that has been preserved is a table created in the 2nd century AD in Egypt to organize astronomical information as a tool



Κανόνιον τῶν ἐν κύκλῳ εὐθειῶν			Table of Chords		
περιφερειῶν	εὐθειῶν	ἑξηκοστῶν	arcs	chords	sixtieths
½	ϙ λα κε	ϙ α β ν	½°	0;31,25	0;1,2,50
α	α β ν	ϙ α β ν	1°	1;2,50	0;1,2,50
α½	α λδ ιε	ϙ α β ν	1½°	1;34,15	0;1,2,50
β	β ε μ	ϙ α β ν	2°	2;5,40	0;1,2,50
β½	β λξ δ	ϙ α β μη	2½°	2;37,4	0;1,2,48
γ	γ η κη	ϙ α β μη	3°	3;8,28	0;1,2,48
γ½	γ λθ νβ	ϙ α β μη	3½°	3;39,52	0;1,2,48
δ	δ ια ις	ϙ α β μξ	4°	4;11,16	0;1,2,47
δ½	δ μβ μ	ϙ α β μξ	4½°	4;42,40	0;1,2,47
ε	ε ιδ δ	ϙ α β μς	5°	5;14,4	0;1,2,46
ε½	ε με κζ	ϙ α β με	5½°	5;45,27	0;1,2,45
ς	ς ις μθ	ϙ α β μδ	6°	6;16,49	0;1,2,44
ς½	ς μη ια	ϙ α β μγ	6½°	6;48,11	0;1,2,43
ζ	ζ ιθ λχ	ϙ α β μβ	7°	7;19,33	0;1,2,42
ζ½	ζ ν νδ	ϙ α β μα	7½°	7;50,54	0;1,2,41
⋮	⋮	⋮	⋮	⋮	⋮
ροδ½	ριθ να μγ	ϙ ϙ β νγ	174½°	119;51,43	0;0,2,53
ροε	ριθ νγ ι	ϙ ϙ β λς	175°	119;53,10	0;0,2,36
ροε½	ριθ νδ κζ	ϙ ϙ β κ	175½°	119;54,27	0;0,2,20
ροϛ	ριθ νε λη	ϙ ϙ β γ	176°	119;55,38	0;0,2,3
ροϛ½	ριθ νς λθ	ϙ ϙ α μξ	176½°	119;56,39	0;0,1,47
ροζ	ριθ νζ λβ	ϙ ϙ α λ	177°	119;57,32	0;0,1,30
ροζ½	ριθ νη ιη	ϙ ϙ α ιδ	177½°	119;58,18	0;0,1,14
ροη	ριθ νη νε	ϙ ϙ ϙ νς	178°	119;58,55	0;0,0,57
ροη½	ριθ νθ κδ	ϙ ϙ ϙ μα	178½°	119;59,24	0;0,0,41
ροθ	ριθ νθ μδ	ϙ ϙ ϙ κε	179°	119;59,44	0;0,0,25
ροθ½	ριθ νθ νς	ϙ ϙ ϙ θ	179½°	119;59,56	0;0,0,9
ρπ	ρκ ϙ ϙ	ϙ ϙ ϙ ϙ	180°	120;0,0	0;0,0,0

Figure 1.1: A section from Ptolemy's table of chords [Maor, 2002].

for navigation (cf. Figure 1.1). Although a table is primarily a textual representation of data, it also uses some visual attributes like spatial position and vertical or horizontal lines to arrange the data into rows and columns.

The first line graphs go back to Robert Plot (1685), and to Christopher Wren (1750) who invented a mechanical device for automatically recording a temperature graph. The well-known graphical representation for the display of quantitative data such as pie charts, line graphs and bar charts which we use today are attributed to the social scientist William Playfair (1759–1823).

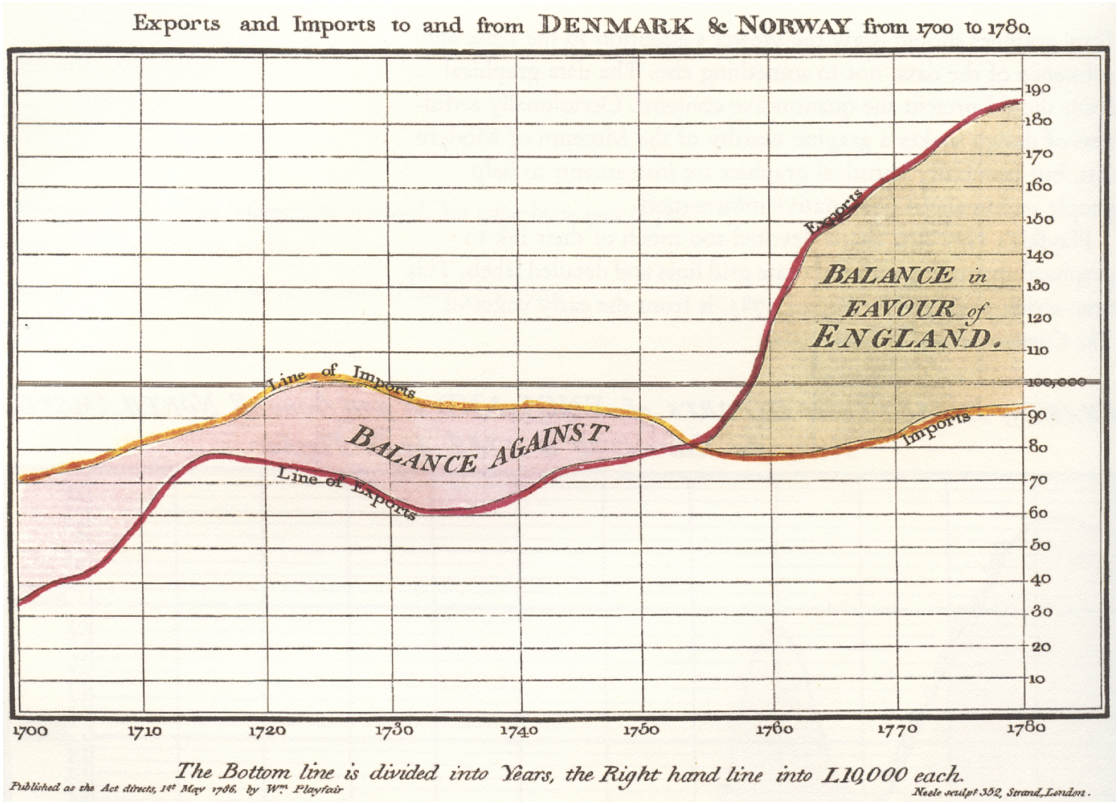


Figure 1.2: William Playfair's time series of exports and imports of Denmark and Norway [Playfair, 1786].

The use of modern electronic displays provides the possibility to manipulate the visual representations comparable to pencil and paper in the above example. With the introduction of affordable computers with graphic displays in 1980s emerged a new field of research called "information visualization". Software information visualizations are capable of displaying large datasets using various representations and give the user a wide variety of possibilities to explore the data interactively.

1.1 Motivation

Across different domains such as medicine, finance or the military, the data flow has surged; the increasing application of modern data collection technology makes a large number of multivariate data available. It is a major challenge for domain experts to exploit the deluges of data and identify crucial information. Sometimes vitally important decisions based on the collected data have to be made very quickly: for example, in the intensive care unit or at military combat operations. The analysts responsible for interpreting the swirl of data face a new problem: information overload.

Information visualization on electronic displays is an instrument to help the data analysts overcome the information overload problem and make such vast datasets intuitively comprehensible. The variety of possibilities to display the data makes it difficult for designers of interactive information visualization to decide how to represent the data appropriately and how to provide accurate interactivity.

The information visualization community has incorporated a lot of human perception research findings into guidelines and principles (e.g., [Ware, 2004; Wickens & Carswell, 1995]) to help designers find appropriate visual encodings and interactions for the data to be visualized.

However, it is important to estimate if these design decisions are applicable for the possible users. It is necessary to find out if the design of the information visualization is practical for the data to be displayed and the tasks the users want to fulfill.

1.2 Problem Statement

Bade et al. [2004] have developed some well thought-out interactive visualization techniques for multivariate time-oriented data that combines colored qualitative representations with more detailed quantitative representations. These techniques ease the recognition of critical periods or concrete fluctuations in the data even if the vertical display space of the visualization has a small height by the use of a *semantic zoom* technique. A subset of these visualization techniques has been implemented as a prototype called *SemTimeZoom.* A detailed introduction of *SemTimeZoom* will be presented in chapter 3.

It has become crucial for researchers to present actionable evidence of measurable benefits to encourage more widespread adoption of their techniques [Plaisant, 2004]. In other words, for the well-accepted adoption of novel visualization techniques, it is necessary to prove that the visualizations are fulfilling their proposed aims and meet the expectations and needs of users. Bade also mentions that it is necessary to perform user studies and evaluations to take the concept to application.

Performing a systematic evaluation of a visualization technique is not a trivial task; the researcher has to pick the right focus and questions, as in all empirical research. Having interesting questions, it is difficult to find the appropriate examination methodology.

Once the examination methodology has been found, the evaluation has to be planned with the intention of answering the research questions. During the execution of the evaluation, it is usually necessary for the researcher to observe and collect data. Once these data have been collected, they have to be analyzed. In a quantitative evaluation, it is usually statistical methods that are applied to find statistical evidence to reject or accept a hypothesis that was derived from the research question.

To encourage researchers in the information visualization domain to carry out an evaluation of their prototypes and tools, there is a need for a solid evaluation infrastructure. There is already awareness that evaluation is important and to stimulate effort on this issue, developers of prototypes need solutions for how to integrate evaluation functionality into their prototypes and how to collect and measure the data produced by the users participating in a evaluation study.

1.3 Research Objectives

The main objective of this thesis was to carry out a systematic evaluation on the *SemTimeZoom* technique, which will be introduced in detail in chapter 3. The evaluation was aimed to assess the effectiveness for lookup and comparison tasks and discuss possible improvements of the *SemTimeZoom* technique. The conclusions from the evaluation should also be generalizable and provide useful insights for future designers of visualization tools.

Another objective was to pave the way for future researchers in the information visualization domain to easily integrate evaluation functionality into their prototypes and tools. Consequently, the intention was to make sure that the developed evaluation functionality for the prototypes in this work is as reusable and flexible as possible and decoupled from the prototypes.

1.4 Research Questions

During the thesis, the following questions should be answered:

State of the art research

Which related visualization techniques for multivariate time-oriented quantitative data using qualitative abstractions and/or semantic zoom abilities are described in the scientific literature?

Have these techniques been evaluated and if so, what methodology was used and what were the results?

Evaluation

Is the *SemTimeZoom* technique effective for the identification and comparison of qualitative attributes of the data for multiple time-oriented variables?

Furthermore, is the *SemTimeZoom* technique well suited to find and compare quantitative values within specified qualitative levels?

How can the *SemTimeZoom* technique be improved to fulfill the intensions of the design?

Design patterns

Which evaluation functionality implemented in this and previous available evaluation prototypes can be reused and refined as design patterns for future researchers?

1.5 Methodological Approach

Essential for experimental research is the existence of one or more hypotheses that were derived from the research questions concerned with evaluation. The next step was to choose an appropriate evaluation methodology, based on literature research and depending on the current state of the prototype. The experiment had to be designed according to the methodology. An important factor for the design was the availability of a related visualization technique for multivariate quantitative data that is also capable of representing qualitative abstractions of the data to conduct a comparative study. Another possible approach would have been to measure the impact of variations of visualization modes and associated interactivity in the prototype itself, but a comparison with another, ideally already evaluated, visualization technique leads to a more informative study of the performance of the visualization technique. A study of the variations of the visualization modes can be used to explore the scalability of particular visualization modes (cf. [Lam et al., 2011]).

Based on the hypotheses and a methodology, meaningful data and corresponding tasks had to be found and the number of participants had to be determined. On this basis, the *SemTime-Zoom* prototype and a comparison visualization technique had to be extended with evaluation functionality to enable the users to execute a series of tasks according to the study design and measure the depending variables. To ensure that these methods are reusable, evaluation functionality from previous evaluation studies were examined and reviewed. Also literature research on generic design patterns have been done and findings were incorporated into these methods.

Once the experiment design was finished and the evaluation functionality was implemented, a pretest was conducted to test the experiment design, the system and the study instruments. The next challenge was to recruit enough test persons for the study, split them into test groups and finally carry out the evaluation experiment. The test persons were invited for experiment sessions where they got an introduction to the visualization tools they were about to use in the experiment. They also received training in the visualization techniques before the actual experiment. While an experiment was in progress, the quantitative data produced by a participant were recorded automatically with the implemented evaluation functionality. Also some qualitative data were collected prior to and following the experiment trials with the help of questionnaires.

Following the completion of the experiment, the quantitative results were analyzed with statistical methods and existing questionnaires were reviewed. The outcome of the analysis will be discussed as a conclusion at the end of the first part. Additionally, future applicability of the implemented evaluation functionality will be discussed at the end of the second part. The overall conclusion of this work will be presented at the end of this thesis.

Part I

Empirical Evaluation

CHAPTER 2

Background

This chapter gives an overview of the different types of data that can be visualized and also presents a brief introduction to qualitative abstractions and their advantages.

2.1 Data Types

Data represents results of measurements or observations of phenomena. The classical definition of data types or classes used by most statisticians was introduced by the American psychologist Stanley Smith Stevens. Stevens [1946] proposed the theory of levels of measurement, which was the result of the initial question: Is it possible to measure human sensation? Discussions and deliberations on this question led to the disagreement about what is meant by measurement. To find an agreement, he developed the theory of levels of measurement, introducing four distinct data scales, based on empirical and mathematical considerations (cf. Table 2.1) to distinguish between measured variables that have different properties.

Qualitative variables are non-numerical variables that fall into *categories* or *levels*. There are two types of qualitative variables: *nominal* variables have no ordering to their categories; the categories of an *ordinal* variable have a natural ordering, such as *cold - warm - hot*.

Quantitative variables are numerical measurements that are objectively measurable on an *interval* or *ratio* scale.

Scale	Basic Operations	Example
Nominal	Determination of equality	Colors: red, green, blue
Ordinal	Determination of greater or less	School grades: A, B, C, D, F
Interval	Determination of equality of intervals or differences (no natural zero)	Temperature: 10°C, 20°C
Ratio	Determination of equality of ratios	Speed: 40 km/h, 60 km/h

Table 2.1: Data scales by Stevens [1946]

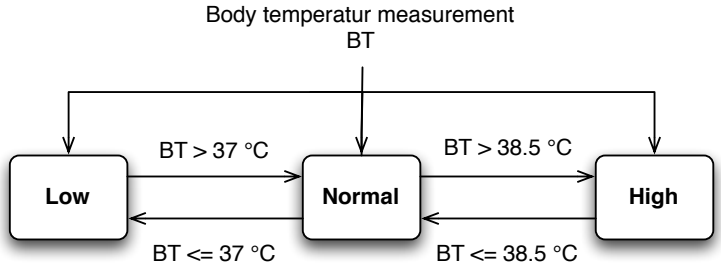


Figure 2.1: Example of a qualitative abstraction process of body temperature measurements as a state machine diagram

2.2 Qualitative Abstractions

"Qualitative data are sexy." [Miles & Huberman, 1994]

Modern data collection systems produce a huge amount of quantitative data across different practical domains such as medicine or finance. Especially in the medical domain, there is awareness that it is important to support decision-making in real-time medical environments like intensive care units (ICUs). It can be difficult for the clinicians to make accurate decisions, particularly when the decisions are based on multiple clinical parameters [Farrington, 2011]. The traditional monitoring of patients is a process where the vital parameters are measured with sensors and the raw quantitative values are shown on an electronic display. The typical representations used to display time-oriented quantitative data are line graphs, scatter plots or bar charts, etc. But these representations lack the possibility to display interpretations and meanings derived from a-priori or associated knowledge about the data to support the clinician in making quick decisions.

The transformation from quantitative data (values) to qualitative data (meanings and interpretations) is termed *qualitative abstraction* or *symbolization*. A simple example for the qualitative abstraction of measured body temperature can be seen in Figure 2.1.

Abstractions can either be performed on the non-temporal attributes of the data or incorporate the temporal attributes. An example for an abstraction using non-temporal attributes would be

that a body temperature of 40°C might be abstracted to "high". Two distinct "high" abstractions that hold on Monday and Wednesday might be abstracted into one "high" that holds during the interval from Monday to Wednesday. It is also possible to analyze the trend of temperatures values over time and abstract the trends into the categories *increasing*, *decreasing* and *steady*. If the temporal attributes are included in the abstraction, the process is also called *temporal data abstraction*.

The abstraction of raw data to meaningful information as higher level qualitative descriptions and displaying these abstractions on a patient monitor can support quick interpretation of patient data. The abstractions can also be used for recommending therapeutic actions as well as for assessing the effectiveness of these actions within a certain period [Miksch et al., 1996].

The overall process of using domain knowledge and data analysis to interpret the data is called *intelligent data analysis* (IDA). There has been intensive research activity in temporal feature extraction methods especially in the medical domain and an overview can be found in the survey by Stacey & McGregor [2007].

This work is focused on the evaluation of visualization systems that provide simultaneous depiction of quantitative values and their associated qualitative abstractions.

CHAPTER 3

Visualization Technique SemTimeZoom

3.1 Introduction

Bade et al. [2004] have developed several interactive visualization techniques, which enable the users to view a large number of time-oriented data at several levels of detail and abstraction, ranging from broad overview to the fine structure.

A major part of this work was focused on visualization techniques for qualitative abstractions and the associated quantitative time-oriented data. The idea was to use the available display space efficiently by using a *semantic zoom* [Bederson & Hollan, 1994] technique, i.e. adapting the visual representation of the data according to the available display space. A very well known example of semantic zooming is Google Maps[1], which uses different *visual information resolutions (VIRs)*, depending on the actual zoom level. For a zoom level containing an entire continent, only country borders, large water bodies and some country name labels are shown. Zooming in further, more information appears, like big cities and rivers. The next zoom level additionally shows traffic links, medium size cities, etc. (cf. Figure 3.1). This technique has proved its benefits through intensive worldwide adoption. Another commonly used technique is geometric zooming, where all objects are visible in every zoom level and only change their size when zooming.

To apply the idea of semantic zooming to time-oriented quantitative data, one way would be to add more information to a line plot in higher zoom levels, e.g. adding axis labels or highlighting individual data points, but the representation itself basically stays the same. Another opportunity arises if the data have known structures and it is possible to use categories higher up in the

[1] http://maps.google.com/

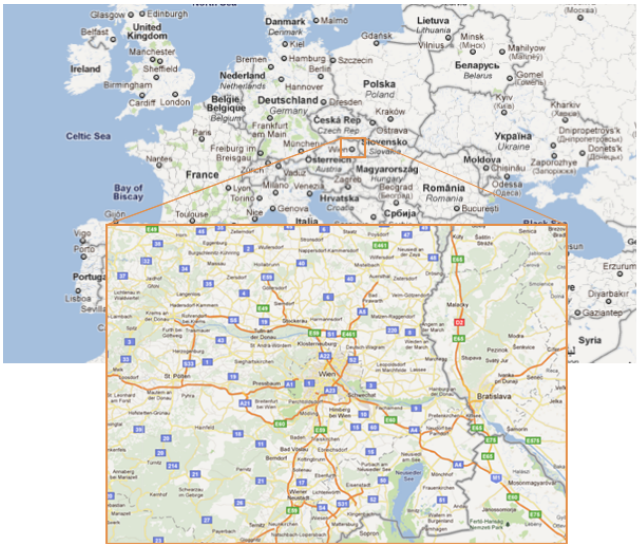


Figure 3.1: Google Maps as example for a semantic zoom technique. The upper map shows the entire continent and only country borders & labels, big cities and large water bodies are shown (low VIR). The lower map additionally shows traffic links, rivers, lakes, small cities etc. (high VIR) (c) 2011 Google

structure to create low *visual information resolutions* (cf. Figure 3.1).

It is not always possible to use categories, but, for example, in the medical domain it is quite common to interpret measured vital parameters and assign measured values to specific categories (qualitative abstractions). In the case of hyperthermia, body temperature is usually categorized using a threshold: a body temperature above 38.5°C (101.3°F) is assigned to fever and body temperature measurements below 38.5°C to normal temperature (see chapter 2 for more details on qualitative abstractions). The introduction of qualitative categories offers the possibility to create different *VIRs* for the visualized data at different zoom levels and also present the user interpreted a-priori knowledge, if available for the data.

3.2 Visual Encodings

In the following, a subset of selected visualization techniques developed by Bade et al. [2004] for different visual information resolutions using qualitative abstractions are presented, ordered by the required vertical display space.

Color-Coded Horizontal Bars

The lowest visual information resolution level only presents the qualitative abstractions (categories) of the underlying quantitative values as colored horizontal bars over a period of time, similar to *LifeLines* technique [Plaisant et al., 1996]. In the original *LifeLines* technique, the

Figure 3.2: Lowest VIR represented as color-coded horizontal bars for a fever curve [Bade et al., 2004]

Figure 3.3: Second VIR using color-coded horizontal bars with different heights to visualize the ordinal scale of the data [Bade et al., 2004]

colored bars are plotted over a period of time of an assigned action or event and the distinct colors of the bars are used to represent relationships between events or actions. In the case of the visualization technique investigated in this thesis, the colors are used to represent periods of different qualitative categories. As an example, critical fever values can be colored red, moderate fever yellow and normal temperature values green (cf. Figure 3.2). The use of intuitive, signalizing colors for the periods of qualitative abstractions allows the user to easily locate critical fever periods, despite a very small vertical display space required. The vertical display space could be reduced to a minimum of only one pixel, theoretically without losing information. It should be noted that appropriate and effective colors for the qualitative levels have to be found, depending on the underlying nature of the data and its abstractions, which is not always an easy task. Moreover, the used color scheme is also an important design choice, e.g. if the users of the visualization tool might have color vision deficiencies.

Height and Color-Coded Horizontal Bars

The visual representation for the next zoom level is a descendant of the color-coded horizontal bars and enhances the representation by using different heights for the bars. This visual encoding makes it possible to intuitively show the ordinal scale of the qualitative levels. Figure 3.3 shows the same data as Figure 3.2 using different heights for the color-coded bars according to the ordinal characteristics of the qualitative abstractions.

The introduction of different heights for the qualitative representations adds more information to the representation, but also the required vertical space increases, as a theoretical minimum of n pixels (where n is the number of different qualitative levels) is needed to display the same amount of data. Another potential benefit is the added redundancy to the representation, by using different colors and different heights for the qualitative abstractions.

Figure 3.4: Hybrid technique using color-coded regions below the line graph [Bade et al., 2004]

Hybrid Representation with Color-Coded Regions

The next step in the visual information resolution hierarchy combines the intuitive qualitative representations with a more detailed quantitative representation, called *hybrid techniques*. To display the quantitative values over time, a line graph connects the distinct data points chronologically. The first *hybrid* representation enhances the line graph with color-coded qualitative regions below the line that connects the points. An example for the known fever curve can be seen in Figure 3.4. It is possible to read off exact values and relations from the data, but also a-priori knowledge in the form of qualitative abstractions is included. The representation in Figure 3.4 is especially useful for a visualization of the data at small heights, since the colored regions aid the perception of the otherwise small variations in the data.

Hybrid Representation with Horizontal Level-Crossings

The second hybrid technique was influenced by the visualization approach of the *Graphical Summary of Patient Status* by Powsner & Tufte [1994]. This visualization uses small multiples to visualize a large number of patient data. The data is scaled to fit in five graphically same sized qualitative ranges (cf. Figure 3.5). Using scaled values simplifies the interpretation, but unfortunately, this visualization scales every qualitative range by a different amount, since the qualitative levels can have different value-ranges. The unequal scaling causes a distorted visual distance between data points and makes it complicated to read off exact values or estimate relations between distinct data points.

The visualization technique by Bade et al. [2004] overcomes this disadvantage and scales every qualitative range by the same amount. Again, the same color-coding is used as in the previous representations, but this time the y-axis is colored to visualize the qualitative attributes. Additionally, small colored lines mark the points in time where the values leave one qualitative level and enter another (cf. Figure 3.6).

3.3 Browsing the Data

The above presented visualization techniques can be connected to an interactive data browser (cf. Figure 3.7). Resizing the data panel vertically zooms through the different representations and more detail of the data is shown, as more vertical display space becomes available. As a result, the user can choose how much detail of the data is shown, depending on the task to be performed. The browser maintains the same colors for the qualitative abstractions in every representation and uses animations for the representation transitions to explain one representation by another.

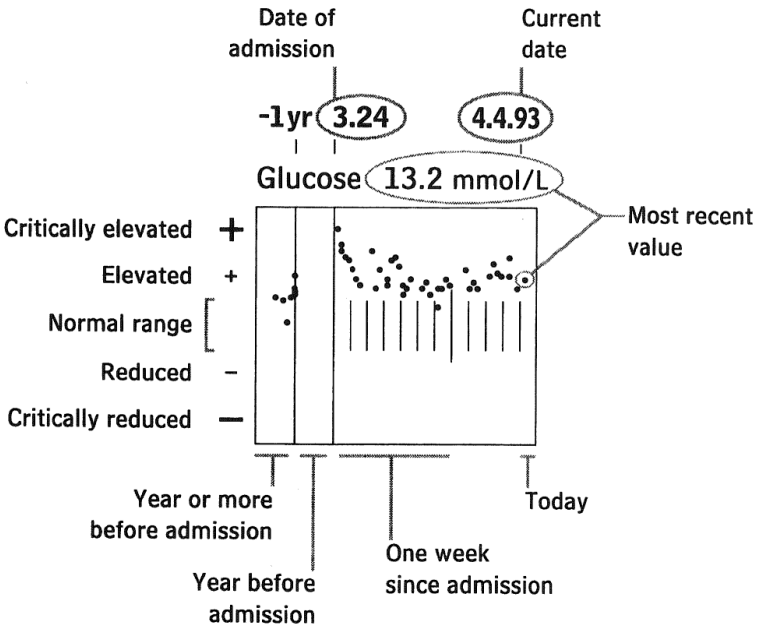


Figure 3.5: Graphical Summary of Patient Status [Powsner & Tufte, 1994]

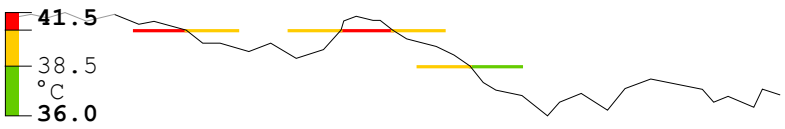


Figure 3.6: Second hybrid technique using colored y-axis and horizontal level-crossings to represent the qualitative attributes [Bade et al., 2004]

Figure 3.7 shows the changing visual representations for the same data, from a broad overview (low VIR) on the top to the fine structure at the bottom (high VIR).

3.4 Animations

Heer & Robertson [2007] investigated the effectiveness of animated transitions between common statistical data graphics. They found evidence that, with careful design, animated transitions can improve graphical perception of changes between statistical data graphics. They detected significantly improved perception of changes in data when using animations through experimental investigations. Also, the users significantly preferred animations for changes in the data to static transitions. With respect to the presented visualization technique, it is important that the users are aware of representation changes during a resizing interaction and Heer & Robertson [2007] suggest that the use of animation is appropriate to support the perception of changes in the data.

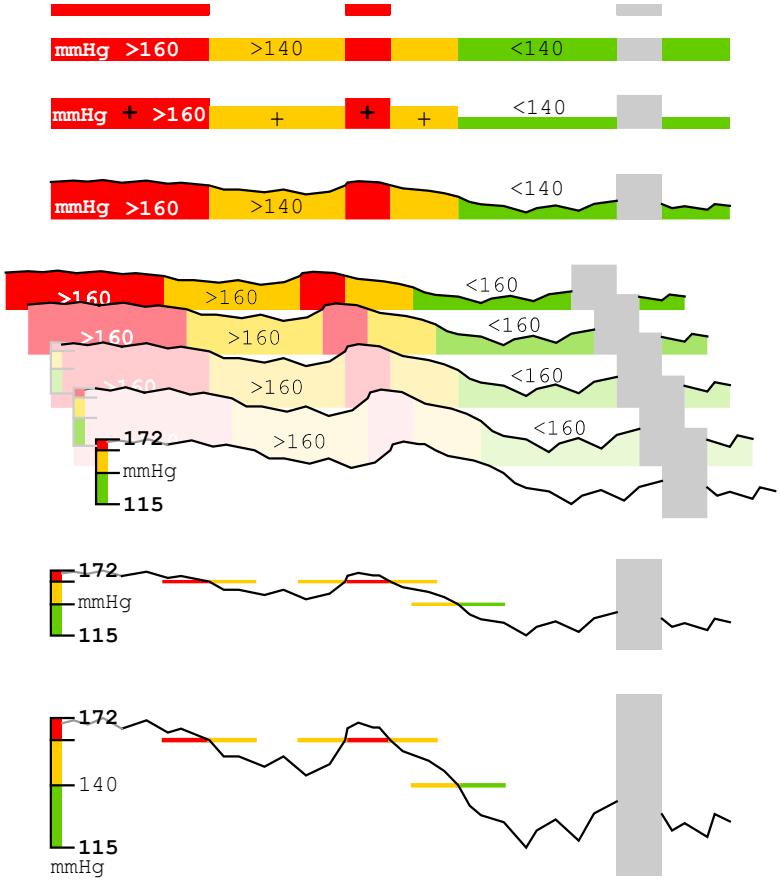


Figure 3.7: Visualization of the data in four different zoom levels. The used zoom level is depending on the vertical display space [Bade et al., 2004]

3.5 Prototype

The visualization techniques presented above were developed as part of the *Midgaard project* [Miksch, 2004]. Since only a subset of the techniques that were developed in the Midgaard project are investigated in this thesis, this subset will subsequently be called *SemTimeZoom*[2]. The prototype additionally includes a semantic zoom feature dependent on the time-period of the shown data using box-plots for high-frequency data. Since this technique was not part of the evaluation, this visualization mode was not presented in this work. More details on this topic are provided in [Bade et al., 2004].

The *SemTimeZoom* visualization technique was implemented as a prototype in the *Java* programming language using the *prefuse* visualization toolkit [Heer et al., 2005]. The documentation of the prototype and an executable file are available at [Hoffmann, 2010]. The SemTimeZoom im-

[2] Alternatively called *Gimlé*, the city that was built out of the ruins of Midgaard after Ragnarök

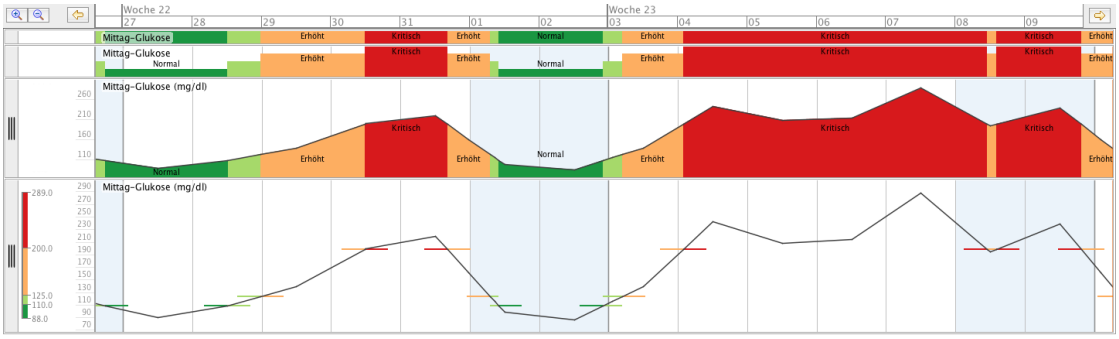


Figure 3.8: Screenshot of SemTimeZoom prototype showing blood glucose measurements including qualitative abstractions from broad overview to fine structure

plementation was later also refined and integrated into the VisuExplore Framework [Rind et al., 2011].

Figure 3.8 shows a screenshot of the interface of the *SemTimeZoom* prototype integrated into the VisuExplore Framework. Each data panel in Figure 3.8 uses one of the four introduced visual representations for the same time-oriented variable. Direct manipulation of the data panels with the mouse cursor can be used to resize the data panels vertically and thus switch between the different visual representations.

The data that were used in the screenshot in Figure 3.8 are blood glucose measurements of a diabetes patient record with the qualitative categories *normal*, *slightly elevated*, *elevated* and *critical*.

Colors

The colors for the qualitative abstractions in the example in Figure 3.8 have been found with the help of the online tool *ColorBrewer* [Brewer & Harrower, 2003], which was developed to help map designers choose effective and approved color schemes for thematic maps.

ColorBrewer offers three different types of color schemes: qualitative, sequential and diverging [Harrower & Brewer, 2003].

Qualitative color schemes primarily use differences in hue to create a color scheme that does not represent order, but differences in kind. Sequential color schemes comply with ordered scales from low-to-high values. They are suited to represent data that range from low-to-high values either on an ordinal scale or on a numerical scale (e.g. cold to warm). Diverging color schemes are sequential multi-hue schemes that emphasize break points of the data by changing the hue.

A diverging color scheme was chosen to emphasize an important break point in the data from the *slightly elevated* level to the *elevated* level. This breakpoint is emphasized by a hue change

Figure 3.9: ColorBrewer scheme used in the SemTimeZoom prototype for the qualitative levels of blood glucose data [Brewer & Harrower, 2003]

from the primary color green for the *normal* and *slightly elevated* levels to the primary color red for the *elevated* and *critical* levels. The hue change is visible in the middle of the color sequence in Figure 3.9, which shows the color scheme used in the prototype (cf. Figure 3.8) for the qualitative abstractions of the glucose data.
The chosen colors also correspond with the common convention that red = danger, red = stop, green = life, green = go. However, it is important to keep in mind that color conventions varies between cultures. For example, in China the color red symbolizes life and good fortune, and the color green symbolizes death.

Representation Transition Heights

Another important design decision for the prototype was the determination of the heights of a data panel for the representation transitions, e.g. the height for the representation change from the height-coded bar representation to the hybrid representation.

Color-coded horizontal bars

Though the lowest VIR representation (cf. Figure 3.2) can theoretically be reduced to a minimum of only one pixel, this is not recommended, because small color-coded objects are hard to distinguish. In general, the larger the color-coded area, the easier it is to distinguish [Ware, 2004]. The minimum size of the color-coded bars representation was determined following the recommendation of Ware that color-coded objects should have a size of half a degree of visual angle. If the size gets smaller, it is possible that the colors are confused even if they are different enough due to a phenomenon known as small-field color blindness.

To calculate the height for a visual angle, Equation 3.1 can be used, where Θ is the visual angle and *h* is the height of the viewed object and *d* is the distance to the object.

$$\Theta = 2arctan(\frac{h}{2d}) \tag{3.1}$$

For a screen size of approximately 36 x 25 cm with a resolution of 1280 x 800 pixel and the estimated object distance of 40 cm this results in a minimum height of about 11 pixels for the color-coded bars representation.

Color-coded and height-coded horizontal bars

The height-coded bar representation uses equal scaling of all qualitative levels (cf. Figure 3.3). Based on the previous calculated minimum height of a color-coded representation, the height for the transition from color-coded bars to the height-coded representation is calculated as n times the minimum size for color-coded bars, where n is the number of different qualitative levels. In the case of 4 qualitative levels, this would result in a minimum of 44 pixels for the height-coded bars.

However, I must add that this limit can be reduced because this representation adds redundancy to the visualization by height-coding the bars and thus the differentiation of the qualitative levels is enhanced. As a result, since Mackinlay [1986] ranked spatial position and color as the most effective graphical devices for communicating nominal data, I believe that half of the calculated minimum height is sufficient for this representation (22 pixels).

Hybrid Representations

Following the assumptions for color-coded objects (minimum of 0.5 visual angle) to find the appropriate height for the transition from qualitative representation to the hybrid representation in Figure 3.4, the unequal scaling of the qualitative levels in that representation has to be taken into account. Thus, the height for the qualitative level that takes up the least amount of vertical space has to be calculated. If the height of this level exceeds half a degree of visual angle in the hybrid representation, the transition is reasonable. Again, this limit was halved because of the additionally added height coding of the line chart. In the example presented in Figure 3.8, the qualitative level that needs the least vertical space is the slightly elevated blood glucose level and the resulting transition height was calculated as approximately 50 pixels.

I have not been able to find a clear answer for choosing an appropriate height for changing to the second hybrid representation (cf. Figure 3.6). It seems plausible that the filled line chart representation has its limits if the vertical display space gets too large. Ware [2004] suggests that if large areas of color-coding are used, the colors should be of low saturation but does not offer recommendations for maximum sizes of high saturated color-codes.

In my opinion, large areas of highly saturated colors could distract the users' attention from other relevant characteristics of the data, beside the categorization into qualitative levels (e.g. fluctuations or trends in the data, etc.). A possible solution is to let the users decide independently when to swap to the second hybrid representation by providing a button or hotkey to swap between representations. Currently, a fairly arbitrary transition height of 150 pixels is chosen.

With regards to the second hybrid representation, it might be appropriate to pick up the principle of *banking to 45°*, which was introduced by Cleveland et al. [1988]. Cleveland [1993] demonstrated how the choice of a line chart's aspect ratio (width/height) can impact graphical perception and showed that an average orientation of 45° maximizes the discriminability of line segments. Heer & Agrawala [2006] extended this technique for different aspect ratios using

spectral analysis, called *multi scale banking*. It could be an interesting approach to use an aspect ratio resulting from multi scale or 45° banking for the transition from the first hybrid representation to the second. It has to be noted that with this approach it is not possible to define an absolute height for the transition, since the aspect ratio is also depedent on the horizontal display space and the currently visible data.

Although the concept of this visualization technique appears very promising, it has not yet been evaluated. For the well-accepted adoption of novel visualization techniques, it is necessary to present actionable evidence of measurable benefits that will encourage more widespread adoption [Plaisant, 2004].

The next chapter will give an overview of related visualization techniques and also how they have been evaluated.

CHAPTER 4

Related Work

4.1 Introduction

This section presents some visualization techniques that are related to the STZ technique in terms of displaying qualitative abstractions and/or using several information resolution levels to represent the data. Additionally, the evaluations of these visualization techniques are introduced and discussed.

The first visualization technique (KNAVE) is part of a framework for knowledge-based interpretation of time-oriented clinical data and is capable of visualizing raw quantitative data and interval-based qualitative abstractions.

The second visualization is part of a system called LifeRAC that uses a semantic zoom technique with different visual representations for the data at varying display space. It is capable of displaying large collections of time-oriented variables as a matrix of charts. The system also uses color-coding for the lowest VIR representation, though the way how the abstractions for the colors are calculated differs from SemTimeZoom and KNAVE.

The last work in this chapter investigates how multiple information resolution interfaces perform for data that have only a single level of inherent structure, i.e., without the use of qualitative abstractions.

4.2 KNAVE – II

Introduction

KNAVE-II (Knowledge-based Navigation of Abstractions for Visualization and Explanation) is an application for interactive visualization, interpretation and exploration of time-oriented

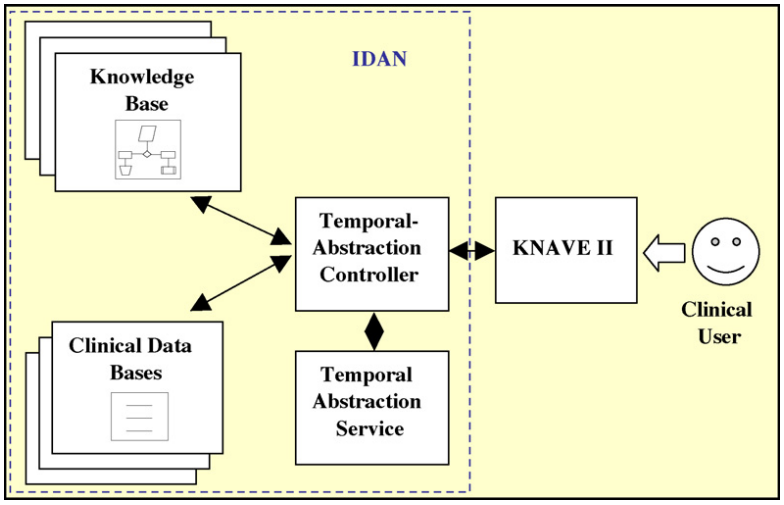


Figure 4.1: The combined architecture of KNAVE-II and IDAN for computing abstractions from time-oriented clinical data [Shahar et al., 2006]

clinical data [Shahar et al., 2006]. The application supports on-the-fly interpretation of time-oriented clinical data using a distributed knowledge-based temporal abstraction mediator for the computation of qualitative abstractions called IDAN [Boaz & Shahar, 2003]. An illustration of the architecture is presented in Figure 4.1. The IDAN system queries a domain-specific knowledge base and an appropriate data source module for a specific abstraction. The concepts obtained for the abstractions and the associated data from the data source are then sent to a temporal-abstraction service, which calculates the corresponding qualitative abstractions and hands it to the KNAVE application.

Interface and Visual Encodings

The main part of the KNAVE-II interface consists of the data-browsing panels, which either show raw quantitative values from the data source or the qualitative abstractions that are the result of the temporal abstraction process in the IDAN application applied on the quantitative data. A screenshot of the KNAVE-II user interface can be seen in Figure 4.2, which shows different data panels with raw quantitative data (third and fifth panel) as line charts and also abstracted qualitative abstractions (first, second and forth panel) represented as LifeLines [Plaisant et al., 1996] in different vertical positions. Raw quantitative data are visualized using line charts.

It is also possible to display statistics for the data on each data panel. Default statistics for the quantitative data are mean, maximum, minimum and standard deviation as can be seen in Figure 4.2 on the third and fifth data panel. The statistic used for qualitative abstractions is the distribution of the durations for the different qualitative levels of an abstraction (cf. Figure 4.2, first panel). One reason, among others, for the design choice of Shahar et al. [2006] to separate the representations of raw quantitative data and qualitative abstractions into different panels was to

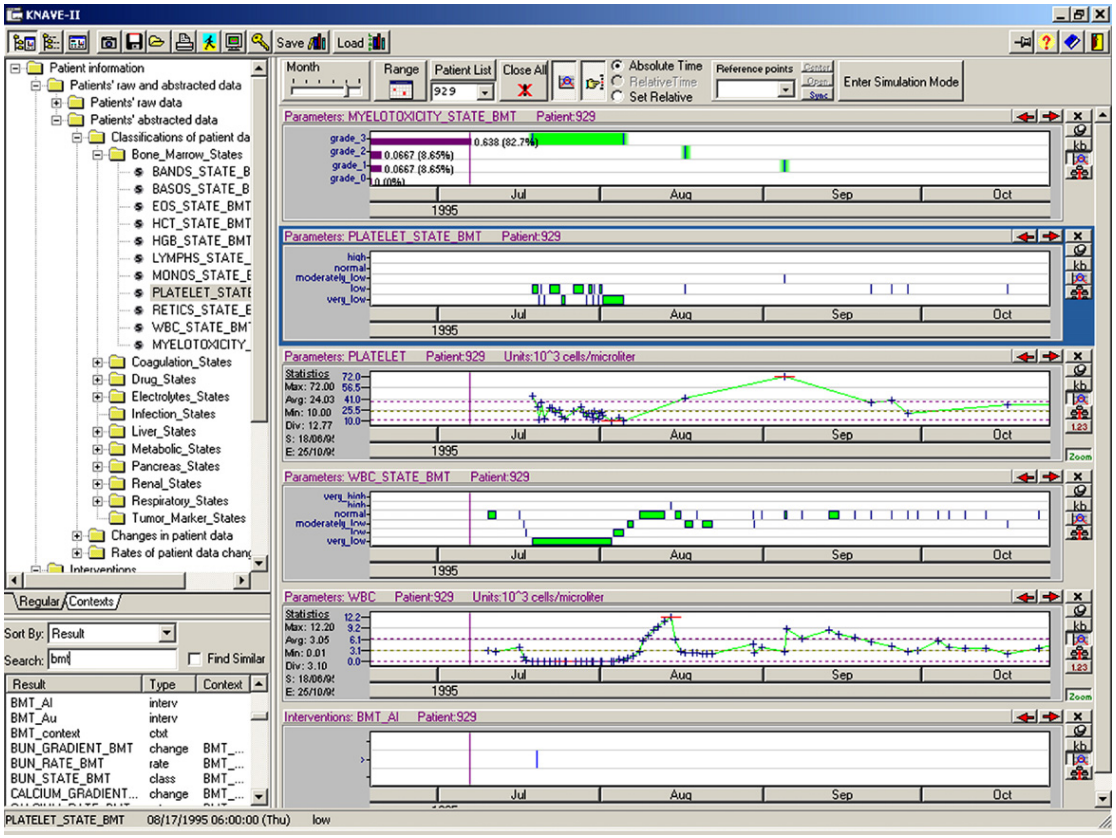


Figure 4.2: A screenshot of the user interface of the KNAVE-II application. On the right side of the window, different panels are shown, which either contain the quantitative values of a time-oriented variable, or the qualitative abstractions calculated from the quantitative values. The user can add a variable or a qualitative abstraction to the data panels by selecting a node in the tree of the ontology browser that is shown on the left side of the window. [Shahar et al., 2006]

enable separate computation of statistics for both, quantitative and abstracted data.

Interaction

KNAVE-II offers several possibilities for the exploration of time, which will be briefly introduced here. Two global time exploration techniques can be seen on top of the window in Figure 4.2: a slide control to zoom to a desired temporal granularity (e.g. week, month, year) and a calendar control allows to specify a start and end point for the displayed time span. Zooming and panning is applied to all data panels simultaneously, but it is also possible to desynchronize distinct data panels from global panning and zooming controls. As can be seen in Figure 4.2, each data panel contains a separate time scale below the visualized data. The user can click on each of the time granularities (e.g. July or 1995) to zoom to the pre-defined time span. If the

Figure 4.3: Explanation of the classification function for a qualitative abstraction [Shahar et al., 2006]

panel is desynchronized, the other panels are not influenced by this action. Also, a distinct time range can be selected with the mouse to zoom to specific contents in the panel.

Another interesting feature of the KNAVE-II application is the use of absolute and relative time. It is possible to select a significant event (e.g. start of medication), which serves as a reference (time zero) for all data panels and the time scale will change to display the time units starting from that event (e.g. hours, days) instead of the absolute, calendar-based time scale. This can be especially useful to compare data from multiple patients that received the same treatment on different dates.

To explore the concepts used for the qualitative abstractions, a semantic browser for each data panel can be activated to show from which components it is abstracted. For detailed information on the classification functions, a knowledge-based explanation for each abstraction is available, such as a table that maps the raw quantitative values to defined qualitative levels (cf. Figure 4.3).

Evaluation

Martins et al. [2008] have evaluated the KNAVE-II interface against an electronic spread sheet (ESS), which was the standard tool in the clinical environment where the experiment was conducted. The evaluation was performed in two consecutive parts, the first evaluation focused on the effectiveness and user satisfaction for answering a set of clinical tasks extracted from oncology protocols with KNAVE, ESS and a paper chart, which was produced by printing out the ESS. The second evaluation omitted the paper chart, since it performed consistently worse than the electronic counterpart, and focused on more complex tasks.

Complexity	Example task
Easy	Find the highest value for this patient's serum creatinine between 20 July 1995 and 31 December 1995
Moderate	During which period did this patient have a "very low" WBC count (defined in KB) post-BMT?
Hard	What are the dates of the last period of grade 3 myelotoxicity (defined in KB) post-BMT?
Hardest	Did this patient have a moderately high creatinine, moderately low hemoglobin and grade 2 liver toxicity (defined in KB) after BMT? On which dates?

Table 4.1: Examples of the 10 clinical tasks used in the first KNAVE-II evaluation [Martins et al., 2008]

First evaluation

The first evaluation was a randomized crossover study of eight participants using KNAVE-II, ESS and paper to perform 10 tasks of varying difficulty. The data were taken from a sample case of a patient who had a BMT (Bone-marrow transplantation) and was modified slightly for each round, creating three similar clinical cases. The experimenters used a stopwatch to record the completion time for each task and an oncology expert pre-determined the correctness of the answers. Also, user satisfaction was measured using a questionnaire. The tasks had 4 different levels of difficulties: 3 easy, 3 moderate, 3 hard and 1 hardest, examples can be seen in Table 4.1.

The participants received a 10-20 minute training session in the use of the KNAVE-II interface and had to answer two training tasks with each tool before advancing to the actual study.

The completion time data were analyzed using RM-ANOVA (Repeated measurements analysis of variance) and no effect was found on the order of the tool (sequence effect).

The test persons answered the hard and hardest tasks significantly faster with KNAVE-II compared to Paper ($p=0.00002$ and $p=0.008$) or the ESS ($p=0.007$ and $p=0.0006$). Easy tasks were significantly faster with the ESS compared to KNAVE-II ($p=0.02$) and moderate tasks were significantly faster with KNAVE-II compared to Paper ($p=0.004$).

The correctness was analyzed using paired t-tests and the result revealed a significantly higher correctness rate for hard queries for KNAVE-II compared to Paper ($p=0.04$). No significant effect was found between KNAVE-II and ESS for correctness, though KNAVE-II had on average higher correctness rates.

Based on a paired t-test, the usability scores were significant higher for KNAVE-II ($p=0.006$)

Complexity	Example task
Moderate	After the BMT, what was the longest period (give dates and number of days) that patient 911 had a "moderately low" WBC?
Hard	After the BMT, what was the longest period of grade 3 liver toxicity for patient 946? Give the number of days and dates
Hard	Did patient 813 have a very high alkaline phosphatase and a high LDH on the same date(s) after the BMT? If so, how many times and on which dates?
Hardest+1	For patients 813 and 946: after the BMT, did these patients have a pattern of "liver dysfunction"? If so, when was the last date? Which patient recovered in the shortest time after BMT?
Hardest+1	Did patients 813, 946 and 929 recover from their myelotoxicity (recovery is defined as myelotoxicity grade 0)? If so, how long after BMT? Give the date and number of days from BMT to recovery.
Hardest+2	Did patients 911, 929 and 946 develop simultaneous grade 3 myelotoxicity and grade 3 liver toxicity after their BMT? If yes, when?

Table 4.2: Examples of the 6 clinical tasks used in the second KNAVE-II evaluation [Martins et al., 2008]

compared to both, ESS and Paper.

Second evaluation

Five physicians took part in the second evaluation and they had to answer six tasks of increasing difficulty using KNAVE-II and ESS. This time the task difficulties ranged from moderate to hardest+2 (Table 4.2) and were considered by an oncology expert as more representative for clinical practice.

ANOVA was used again to test the completion times of the tasks, and all tasks were answered significantly faster with KNAVE-II than with the ESS, except for task 6 (Hardest+2), but four out of five test persons ran out of time using ESS on this task. No one ran out of time using KNAVE-II.

Also, task types hardest+1 and hardest+2 had a significantly higher correctness rate with KNAVE-II ($p<0.0001$ for both task types) compared to the ESS, as was revealed by a t-test. The usability score was also significantly higher with KNAVE-II ($p=0.011$) and a ranking of the user preference ranked KNAVE-II (80%) first.

Discussion

The results of the study are very promising regarding the usability of KNAVE-II for tasks in the clinical domain involving qualitative abstractions. Even though the test persons were quite experienced with Paper and the ESS for tasks in their daily working routine, most of the tasks were answered significantly faster using the KNAVE-II interface, which they had never used before. Especially complex tasks, which have been determined as representative for the oncology domain, had significantly faster completion times than ESS and also significantly higher correctness rates. Furthermore, an analysis of the post-experiment questionnaires revealed that KNAVE-II scored first in terms of user satisfaction.

The strength of the KNAVE-II evaluation is certainly that the test persons were domain experts for the clinical experiment setting and performed real-life tasks on actual patient records. Various comparable experiments (e.g. [Lam et al., 2007] and [Javed et al., 2010]) were performed with university students and tasks that are abstracted from real-life tasks.

Limitations

One limitation is the small number of test persons involved in both evaluations, since more test persons would have increased statistical power.

It should also be noted that both tools (paper and electronic spreadsheets), used as a comparison to KNAVE-II, did not include the calculated qualitative abstractions that were needed for most of the tasks. While both tools had included a list of tables containing the concept definitions that were needed for the calculation of an abstraction (comparable to Figure 4.3), the test persons had to calculate the abstractions for themselves. Even if these comparison tools represent typical or even better formats than the current standard for browsing clinical data, it is not surprising that the test persons performed significantly better with KNAVE-II for tasks involving such abstractions. This would also explain why easy tasks were significantly faster with ESS, because these tasks did not involve any qualitative abstractions.

In my opinion, it would have been a more fair evaluation of the visualization technique if ESS and paper had been included the calculated a-priori knowledge as qualitative levels listed next to the actual data values, though the evaluation was not only focused on the visualization of the data and the abstractions, but also on the overall usability of the KNAVE-II application with the IDAN architecture.

4.3 LiveRAC

Introduction

LiveRAC (Interactive Visual Exploration of System Management Time-Series Data) is a visualization system for analysis of large collections of network devices time-series data [P. McLach-

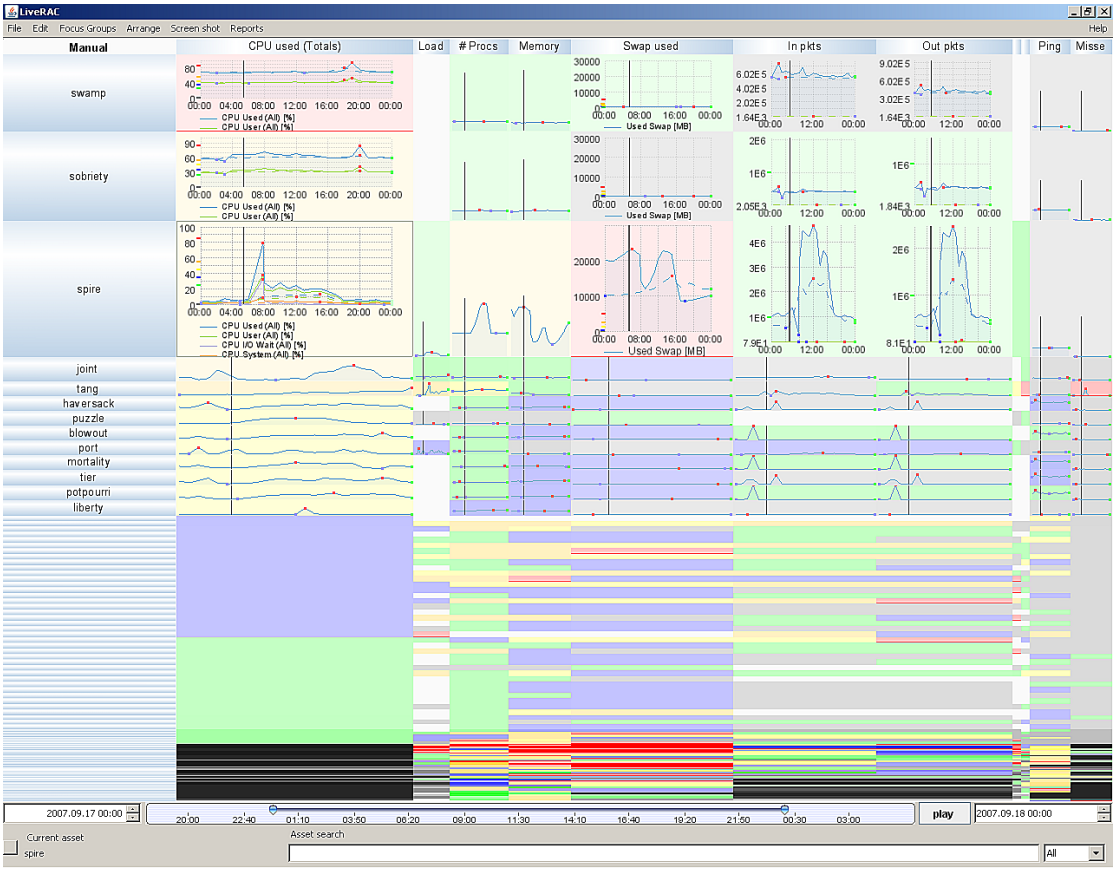


Figure 4.4: Screenshot of the LiveRAC system [P. McLachlan et al., 2008]

lan et al., 2008].

Similar to the SemTimeZoom technique, it uses a semantic zoom technique with different visual representations for the data at varying display space. In contrast to SemTimeZoom, LiveRAC takes a matrix visualization approach, presenting data in a grid of cells similar to a spreadsheet (cf. Figure 4.4). Rows represent network devices and columns present metrics or alarms of these devices. Because of the area-aware rendering technique, a matrix cell can contain a larger number of devices than pixels available for this cell by aggregating devices. The user can specify a focus in the visualization by enlarging regions of interest and thereby compressing the regions that are not in focus. The expanded regions reveal more detailed information about the represented data than the compressed ones. The users can also change the initial ordering of devices or parameters to their preferences.

Figure 4.5: Color scheme for LiveRAC [P. McLachlan et al., 2008]

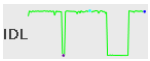


Figure 4.6: Sparkline representation used in LiveRAC [P. J. McLachlan, 2006]

Visual encodings

The visual encoding for the data in the lowest visual information resolution is a colored box. The used color scheme for the boxes is inherited from an AT&T internal application. These colors encode the alarm severity data according to common conventions: critical = red, major = orange, minor = yellow, warning = blue, normal = green, unknown = gray (cf. Figure 4.5).

To encode the information density of a box, different color saturations are used. The base saturation of a box is 25%. If the box is gets enlarged, the saturation of the box decreases to a minimum of 0.05%, following the recommendation of Ware [2004] to use colors with low saturation for large color-coded areas.

Aggregated cells containing more than one device use the color of worst alarm found in the aggregated data of the devices. The saturation of the alarm color depends on the number of found alarms in the devices and starts at a minimum of 25% up to 100% for an alarm count of 10.

The low VIR representation in LiveRAC is quite similar to the color-coded horizontal bars used in the SemTimeZoom technique, but differs in the way of how the qualitative abstractions (alarm severity data) of the data are represented. In LiveRAC, the entire investigated time period of the data is colored according to the worst alarm threshold that was exceeded or the average value in the time window. This is contrary to the SemTimeZoom technique, which only colors the time period where the quantitative data are actually within a qualitative level.

The representation for the next higher visual information resolution in LiveRAC is the sparkline representation introduced by Tufte [2006], which can be seen in Figure 4.6. A sparkline is a small graphic for visualization of trend information of quantitative data in a compact space. It does only include minimal axis and label information and can appear inside of a single line of text in its smallest form. It is quite common to mark the maximum, minimum and current value with small colored dots like in Figure 4.6.

The representation in the highest visual information resolution is shown in Figure 4.7, using line charts with detailed axis and label information. The size of the labels and the density of axis marks depend on the available display space for that cell.

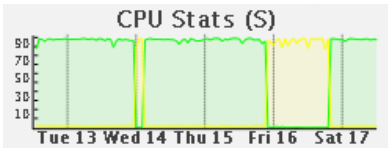


Figure 4.7: A full sized line chart used in LiveRAC [P. J. McLachlan, 2006]

Interactions

The primary interaction mechanism in LiveRAC is a rubber sheet navigation called *accordion drawing*, first introduced in TreeJuxtaposer [Munzner et al., 2003].

Accordion drawing uses a focus+context interaction metaphor where the user manipulates the display as though it was a rubber sheet tacked down at the borders [Sarkar et al., 1993]. If one region in the view gets expanded the rest of the view gets compressed. An illustration of this interaction metaphor can be seen in Figure 4.8. Accordion drawing extends this stretch and squish navigation by a technique called *Guaranteed Visibility* [Munzner et al., 2003]. This technique ensures that marked regions (critical zones) will remain visible regardless of the information density if they are getting compressed.

LiveRAC is the first system that combines accordion drawing with a semantic zoom technique. The size of cells or a group of cells can be manipulated interactively with the mouse by dragging the borders of a selected rectangle in the view. It is also possible to select multiple, non-continuous areas of interest. The representation of the data within the cells depends on the cell's size in the display.

The shown time period is the same for every cell in the matrix and can be changed with a slider or by entering start and end times in a textbox (cf. Figure 4.4 at the bottom).

Evaluation

P. McLachlan et al. [2008] employed an informal longitudinal study to better understand the strengths and weaknesses of LiveRAC. This evaluation was part of a user-centered staged design process in a production environment, which involved a series of prototypes and a varying participant pool in different design stages.

The design process started with the identification of the key requirements. After that, paper prototypes and proof-of-concept interactive software prototypes have been built to obtain feedback from the target users (Life Cycle Engineers). On the basis of the gathered feedback, a high-fidelity prototype has been implemented, running on the production database of the target organization. The high-fidelity prototype was refined and the resulting robust and deployable system

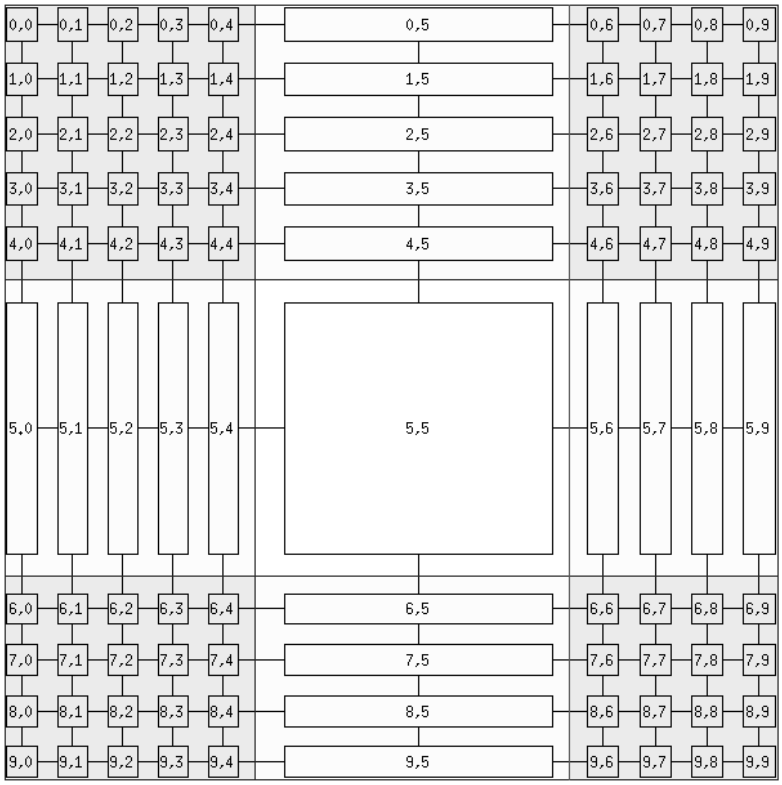


Figure 4.8: Rubber sheet navigation with orthogonal stretching [Sarkar et al., 1993]

was used to carry out the informal longitudinal evaluation over three months. P. McLachlan et al. [2008] performed many interviews and collected the audio recordings whenever possible. Also, hand written notes, audio, screen capture videos and log data of interactive sessions with LiveRAC have been collected and reviewed. The log data of 38 sessions have been collected from 13 test persons. To manage and analyze the diverging collected data, an internal wiki has been built.

The informal study revealed encouraging feedback from the target users and showed that the used visualization techniques stood well up in practice. Some interesting key findings are presented in the following.

- Viewing large numbers of charts side by side was critical in serendipitous pattern discovery: the analysis of the collected data revealed that visual, side-by-side comparison is important for the discovery of new, interesting patterns that the users did not intend to find.

- The rubber sheet navigation was not a barrier to adoption: although a laboratory study found a performance penalty for rubber sheet navigation [Nekrasovski et al., 2006], the

interaction mechanism did not present as much of a challenge to the participants as expected, despite its novelty and limited training.

- Visual, interactive sorting offers significant benefits: based on several screen capture video analysis, the experimenters found that data reordering was a key feature for the users of their deployed visualization system.

Discussion

Plaisant [2004] states that the advantages of longitudinal case studies is that they report on users in their natural environment doing real tasks, demonstrating feasibility and in-context usefulness. The disadvantages are that they are time consuming to conduct, and results may not be replicable and generalizable.

These characteristics can also be seen in the presented longitudinal study of LiveRAC. The experimenters report entire discovery processes of target users doing real life tasks in a production setting. Also, since the study is very user-centric, the user satisfaction is ranked on the first place of the evaluation and design process instead of user performance measured in time and error in a task based evaluation. Hence, the developed visualization tool is tailor-made for the target users, which had a major influence on the design of the tool.

The other side of the story is that the study was very time and resource consuming. It took the experimenters 3 months to observe 38 sessions of 13 test persons. These recorded sessions certainly do not cover every possible use case in the production environment. It would take even longer to cover a great percentage of all real-life tasks in that environment, since the experimenters are not in control of the scenarios and tasks the test persons perform with the tool. Although P. McLachlan et al. [2008] reported some interesting findings based on their observations, it is hard to tell if they apply so well to similar visualizations in another setting. Also, no evidence was reported that the target users are actually more productive (faster or less mistakes) with LiveRAC compared to their usually deployed tools.

4.4 Multiple Visual Information Resolution Interfaces

Lam et al. [2007] investigated the application of multiple visual information resolution (VIR) interfaces for single level data. Using multiple VIR for visualizations with limited display space for data with known structures is not an unusual approach. The structure of the data can be used to create a low VIR representation for an overview and a high VIR for details on demand for a region of interest. If the data to be visualized does not have known structure, it is not clear how to create a low VIR without omitting relevant features of the data.

Consequently, Lam et al. [2007] created two VIR representations with different visual encodings for quantitative time-series data without a known structure and investigated their performance for a set of selected tasks in a controlled experiment. Both VIR representations did not omit any features of the data.

Visual encodings and Interface

The visual encoding for the time dimension is the same for both VIR representations, but encoding of the actual data values is different. The low VIR representation uses only color to encode the data values as a strip of 6 pixels in height (cf. Figure 4.9).

The high VIR representation uses two different encodings for the data: color and spatial position for a line plot of 45 pixels in height (cf. Figure 4.10).

The color-coding is achieved by mapping the normalized data values to saturation and brightness in the HSB space (Hue, saturation and brightness).

These two visual representations have been used to create four different interfaces shown in Figure 4.11: (1) *LoVIR* (2) *HiVIR* (3) *Embedded* (4) *Separate*.

The *LoVIR* interface shows the time-oriented multivariate data only using the strips and the *HiVIR* interface shows only the plots. The *Embedded* and *Separate* interfaces both show strips and plots, but initially only strips. The user can left click in a strip in the *Embedded* interface to show the corresponding plot directly below the strip. The *Separate* interface adds the corresponding plot in the bottom of the view when the user clicks on strip. In both cases, the expanded plot and strip are marked with perimeter boxes.

All interfaces have a panel on the left, which shows the strip/plot number as text-strings for plots or as graphical bars for strips.

Interactions

The *Separate* interface is divided in two areas: the low VIR panel at the top and the high VIR panel at the bottom. The size of the panels is automatically resized if the user adds a plot to the bottom panel. The user can resize both panels by manipulating a divider between the two panels and thereby switching either to the *LoVIR* or *HiVIR* interface by dragging the divider all the way to the top or bottom.

Apart from adding or removing high VIR representations in the multiple VIR interfaces and the divider in the *Separate* interface, the interactions are consistent through all interfaces.

Figure 4.9: Low visual information resolution representation strip [Lam et al., 2007]

Figure 4.10: High visual information resolution representation plot [Lam et al., 2007]

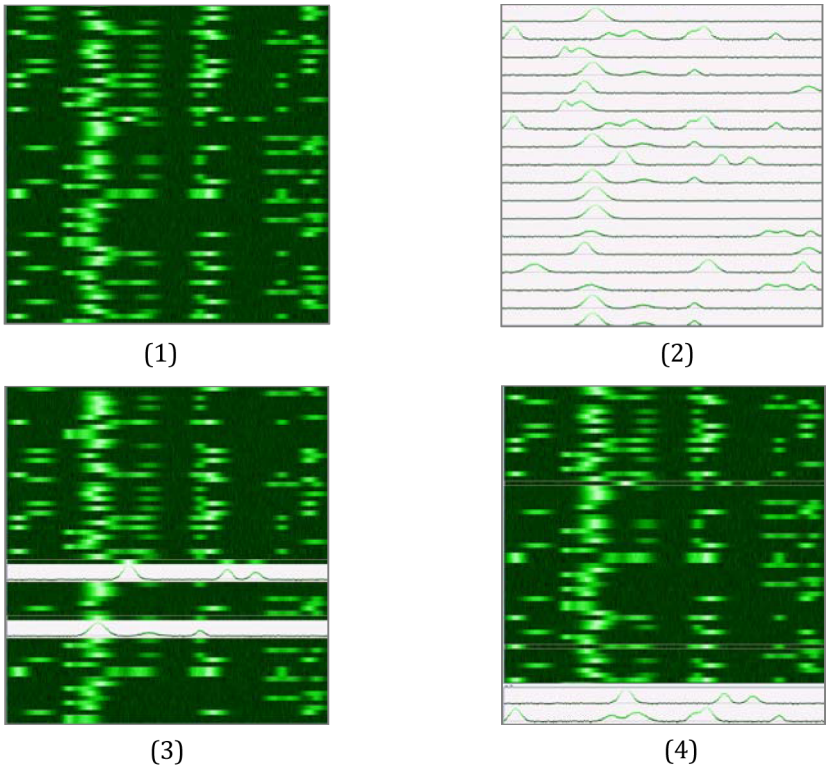


Figure 4.11: Single VIR interfaces: (1) LoVIR (2) HiVIR; Multiple VIR interfaces: (3) Embedded (4) Separate [Lam et al., 2007]

These interactions include scrolling with the help of a scrollbar, marking of strips or plots, keyboard short cuts to mark all elements or showing the high VIR representation for all strips in the multiple VIR interfaces, mouse over highlighting of strips/plots through a perimeter of one pixel and tooltips showing the actual data values and time.

Evaluation

The user performance evaluation of the four interfaces (*LoVIR, HiVIR, Embedded and Separate*) was executed using a within-subject experiment design with the interface and task being the two factors. 24 participants, most of them university students, had to master 4 different tasks using each interface. The experimenters recorded task completion times and error rates. Additionally, detailed observations of test person behaviors and strategies have been captured and noted. At the end of each experiment round, feedback of the test persons was collected.

The experiment scenario of monitoring and managing electric power in a control room was used to create concrete tasks of abstract task definitions. Based on pilot study results, two characteristics were identified that affected high and low VIR target identification: complexity and visual span. Complexity refers to the number of peaks in the data where complex targets have multiple

Task	Example instructions
Max	Which location has the highest power surge for the time period shown on the screen?
Most	Which location has the most number of power surges?
Shape	A fault happened at location <x> at 6:00, causing a similar power surge in another location afterwards. Which one?
Compare	Find the power profile that is the same as that of location <x>.

Table 4.3: Concrete task instructions of the four tasks (Max, Most, Shape, Compare) [Lam et al., 2007]

peaks. A local target refers to a limited horizontal display width of the target and dispersed when the target spans the entire display width. The four resulting tasks for the experiment using variations of these characteristics and the comparison aspect where:

- *Max*: simple, local, no comparison
- *Most*: complex, dispersed, no comparison
- *Shape*: complex, local, comparison
- *Compare*: simple, local, comparison

Table 4.3 shows examples of concrete instructions based on the abstract task definitions used in the experiment referring to the electric power control room scenario.

The data used in the experiment for the tasks were created synthetically to control the visual qualities of the data. In addition to the targets, distractors and background populations were also included to the data for each task to avoid visual pop out [Ware, 2004] of the targets.

Lam et al. [2007] wanted to investigate the impact of selective activation of high-VIR details on perceptual requirements established for single low-VIR views.

The first hypothesis was that the *LoVIR* interface would be the most effective for the *Max* task, insufficient but useable for the *Shape* task, and unusable for the *Most* task.

The second hypothesis was that the *Embedded* interface would better support the *Shape* task than the *HiVIR* and the *Separate* interface.

They also expected that the *Separate* interface would better support the *Compare* task than *HiVIR* and the *Embedded* interfaces.

Study design

The four interfaces were tested against each of the 4 tasks (cf. Table 4.3) with a different, but isomorphic dataset for each trial. The order of the interfaces was counterbalanced between the test persons, the task order was randomized, and the dataset order was fixed. Every test person had to perform each task in four interface sessions, with one training task following one actual task for each of the four tasks.

The experimenter observed the mouse actions, verbal comments, and non-verbal signals of each test person and wrote them down. Lam et al. [2007] also developed a coding scheme for the behavior. For the multiple VIR interfaces it was recorded if the test person used only the high VIR representations, only the low VIR resolutions or both. Also the answer confirmation method was respected in the behavior observation (visual or tooltip) and the visual search mode (*serial search* with the mouse or *visual spotting* simply by gazing at the screen).

Results

The task completion times were analyzed using two-factor ANOVA with interface and task as the factors. A main effect was found in interface ($p=0.001$), task ($p<0.0001$) and interface-task interaction ($p<0.0001$).

Post-hoc analysis revealed that the *LoVIR* interface task completion times were slower than *Embedded* or *Separate.* For interface-task interaction, *HiVIR/Max* tasks were 3.5 times slower than the rest, *LoVIR/Most* were almost 2 times slower and *LoVIR/Shape* 1.7 times slower.

The analysis of the overall subjective preference revealed that the test persons preferred the multiple VIR interfaces over the *LoVIR* interface, *Separate* over *HiVIR* and none preferred the *LoVIR* interface. The results of the subjective ratings of the four interfaces over the four tasks showed that *LoVIR* was preferred for the *Max* task, while HiVIR was preferred for the *Most* task. *HiVIR* and *Separate* were preferred for *Shape* and *Compare* tasks. Analysis of questions regarding the interfaces' ease of use also showed that *LoVIR* scored significantly poorer in all questions.

The results of the evaluation confirm the first hypothesis stating that the *LoVIR* interface would be effective for *Max* task. 22 out of 24 test persons could find the targets without using a *serial search*. Also, subjective preferences were in favor of the LoVIR interface for the *Max* task. The test persons used the plots in multiple VIR interfaces only for confirmation of the answer. The *Shape* task was not easy to handle for the test persons with the *LoVIR* interface. The majority of the test persons relied on *serial search* to locate the target and had to make intensive use of tooltips to confirm the answer. This is why the test persons made more errors, took longer and rated the *LoVIR* as less suitable for this task. Also the *Most* task was extremely difficult for the test persons using the *LoVIR* interface, even when using *serial searching* of the targets.

Relating to the second hypothesis, the results did not reveal that the *Embedded* interface enhanced complex target matching (*Shape* task). Half of the participants switched to the *HighVIR* interface to complete the task and no performance differences were found between the *Embedded*, *Separate* and *HiVIR* interface.

Also, the third hypothesis was not confirmed, as the *Compare* task was equally error prone and slow for all four interfaces. The test persons equally preferred the *Separate* and *HiVIR* interface for this task.

Discussion

The results suggest that a color-coded low-VIR interface for quantitative, unstructured data is only useable for simple and local targets. Also, selective display of high-VIR details did not show enhanced visual search times over using a single high-VIR interface. This may have been the case because the test persons did not have internalized the necessary confidence in the use of the multiple VIR interfaces.

The strength of this study was to perform a training before each task to ensure confidence with the task descriptions. Furthermore, the use of tasks abstracted from real-life tasks enabled the test persons who were novices in the experiment setting to perform the tasks without further knowledge of the domain.

The choice to use synthetic data for the study could have had an influence on the realism of the study scenario and the statistical robustness could have been increased if the tasks were repeated at least once. Also, the active observation of the test persons possibly introduced a bias, since a test person with a person watching over his shoulder may perform differently from a test person who is seated alone [Lazar et al., 2009]. This may particularly have exerted an influence on the multiple VIR interfaces, because the test persons seemed not to show proficiency in the use of these interfaces.

4.5 Discussion

This chapter provided an overview of visualization techniques that are related to the SemTimeZoom technique, which was introduced in chapter 3. The first visualization tool that was presented (KNAVE II) also uses interval-based qualitative abstractions to depict clinical data over time. In contrast to the SemTimeZoom technique, KNAVE II uses separate, static representations for the quantitative and qualitative attributes of the data. A comparison study revealed significantly faster completion times and also significantly better correction rates in favor of KNAVE II compared to current standard methods in clinical data analysis. Additionally, the KNAVE II scored first in terms of user satisfaction.

The second presented visualization tool is called LiveRAC, which uses a semantic zoom rendering technique to display the time-series data of large collections of network devices. Different

visual information resolution (VIR) levels are displayed, dependent on the available display space, but the qualitative abstractions are not interval based like in the SemTimeZoom technique. An informal longitudinal study revealed encouraging feedback from the target users and showed that the used interaction mechanism did not present much of a challenge to the users.

The last related work that was reviewed in this chapter is an evaluation of different visual information resolutions for single level data. Although no qualitative abstractions have been used, this evaluation study investigated the impact of selectable display of high visual information resolution details in overview displays to represent multivariate data over time. The study revealed no benefits of using selectable high-VIR details compared to a single high-VIR display. The single low-VIR display that used only color-coding for quantitative data performed significantly poorer in terms of user satisfaction.

With reference to the SemTimeZoom technique, the results of these evaluations reveal that the inclusion of interval-based qualitative abstractions into a visualization of time-oriented data has advantages compared to current standard methods to analyze data, especially in the medical domain. In addition, the use of a semantic zoom interaction technique for time-oriented did not present much of a challenge for the users. Interestingly, selectable display of details for multivariate, time-oriented data did not show advantages for raw quantitative data. The next chapter presents the design of the empirical evaluation of the SemTimeZoom technique.

CHAPTER 5

Comparative Study

5.1 Introduction

This chapter describes the design of the comparative study, which is the core piece of this work, in detail. The experiment compares the SemTimeZoom (STZ) technique against the visualization technique used in the KNAVE project [Shahar et al., 2006]. To accomplish this, I conducted a controlled experiment to test the effectiveness of these two techniques for lookup and comparison tasks [Andrienko & Andrienko, 2006] of qualitative abstractions of time-oriented data as well as for lookup and comparison tasks of quantitative values associated with qualitative abstractions.

I selected the visualization technique used in the KNAVE project as comparison to the STZ technique because it provides the possibility to display both quantitative data of a variable over time and the associated qualitatively abstracted data in the same manner as STZ. Additionally, it has already been evaluated in several studies (cf. [Martins et al., 2008] and [Klimov et al., 2010]), showing its benefits compared to currently used tools in clinical practice.

The main difference between these two visualization techniques is that KNAVE uses separate, static representation for qualitatively abstracted data and for raw quantitative data. The STZ visualization has the advantage of being more space-efficient than the visualization used in the KNAVE project and combines raw quantitative data with associated qualitative abstractions into one visual representation. The STZ technique was designed to adapt the visual representation to the available space and uses smooth animation between different modes to explain one representation by another. But since this technique requires the user to interact with the visualization, i.e. switch between different representation modes to achieve different kind of tasks, this is expected to influence the user-performance.

The goal of this study is to find out if and how the compact visual representations used by STZ and its interaction techniques affect the user-performance compared to a static visualization tech-

nique where qualitative abstractions are shown separated from the associated quantitative data.

The general data exploration techniques in both cases, i.e. interaction with particular data panels containing a diagram of a time-oriented variable, vertical resizing of a data panel and investigation of individual data-points and intervals of qualitative abstractions via tooltips, were the same to avoid interaction biases.

5.2 Hypotheses

My assumption is that the STZ technique is effective for lookup and comparison tasks of qualitative abstractions as well as for lookup and comparison tasks of quantitative values linked to qualitative abstractions when investigating a single or multiple time-oriented variables. To confirm this assumption, the STZ technique will be compared against the visualization used in the KNAVE project with the above named tasks to establish or reject the assumption.

The first hypothesis is focused on the performance of the visualization techniques when investigating only the qualitative abstracted data of a single or multiple time oriented-variables. The second hypothesis makes claims about the performance of the visualization techniques when, additionally to qualitative abstractions, also quantitative values within those qualitative ranges are investigated for single or multiple time oriented-variables.

The comparative visualization used in this experiment is based on the visualization in the KNAVE project and will hereinafter referred to as the KNAVE visualization, though it is not exactly the same visualization; the exploration technique and the representation of individual data points are equalized to the STZ technique, to avoid biases based on different interaction or the differing representation of single data points. Additionally, distinct colors for all data panels belonging to the same variable are used to ease differentiation of the variables. Figure 5.1 shows a screenshot of the KNAVE visualization technique used in this experiment.

The number of multiple variables in this experiment is limited by the maximum number of variables that can be reasonably displayed with the KNAVE visualization without the need to scroll. Therefore, I limited the number of the time-oriented variables in this experiment to four, also considering the study by Halford et al. [2005], which found that humans can only process up to four independent variables in bar graphs of statistical data accurately and efficiently.

Hypothesis 1: There is **no** difference between the SemTimeZoom technique and KNAVE in accuracy and time spent for tasks involving lookup and comparison of qualitatively abstracted data when investigating time-oriented variables.

Hypothesis 2: The SemTimeZoom technique performs **better** than KNAVE in accuracy and time spent for tasks involving lookup and comparison of quantitative data within specified qualitative abstractions when investigating time-oriented variables.

These hypotheses are based on my assumption that STZ has at least an equal performance compared to KNAVE for dealing with qualitative attributes of the data but should outperform KNAVE for tasks which involve identification of quantitative data which are linked to a specified qualitative level. This claim is based on the reduced vertical span between the representation of the qualitative and quantitative aspects of a variable and thereby reducing the eye travel distance to find quantitative values within a specified qualitative level, following the *proximity compatibility principle*. According to this principle, displays relevant to a common task or mental operation (mental proximity) should be rendered close together in perceptual space (close display proximity) [Wickens & Carswell, 1995]. Hence, comparison and lookup tasks for quantitative values in defined qualitative levels should be faster and less error-prone.

Also, it is assumed that similar colored *LifeLines* [Plaisant et al., 1996] with different vertical positions displaying the qualitative attributes of a time-oriented variable, as used in the KNAVE project, will not outperform a visualization that uses individual colors to mark distinct qualitative areas throughout different representation modes. This assumption is based on the fact that both, spatial position and color are preattentively processed [Ware, 2004] and therefore should have equal performance in terms of being visually identified even after very brief exposure. Also, Mackinlay [1986] ranked spatial position and color hue as the most effective graphical devices for communicating nominal data and color saturation or density is also ranked second behind spatial position for ordinal data. The STZ technique uses different color hues to communicate important breaks in the qualitative data and also different saturation for the ordinal ranking of the data. To increase the perception of the qualitative attributes, different heights for the color-coded bars are used if the vertical display is sufficient.

5.3 User Tasks

Task taxonomy

The tasks used in this evaluation were classified using the task taxonomy introduced by Andrienko & Andrienko [2006]. According to this taxonomy a task consists of two parts: the target, i.e. what information needs to be obtained, and the constraints, i.e. the conditions this information needs to fulfill. The taxonomy additionally takes into account the division of data components into referrers and characteristics. In this evaluation study the referrer value is time and the characteristics are the data values or qualitative levels.

Elementary tasks deal with individual elements of data, i.e. individual references and characteristics (e.g. “What is the value of variable x at the time a?”).

Synoptic tasks deal with the dataset as a whole and its subsets, considered in their entirety. The principal notion on this level is the notion of a behavior, i.e. a certain configuration of characteristics corresponding to a set of references (e.g. “Are the data values of variable x rising during the time period b?”).

Elementary tasks fall into three classes:

- *Lookup tasks:* Find values of data elements that correspond to given values of other data components.
- *Comparison tasks:* Determine the relation between characteristics or references. At least one of the data items must result from some lookup task.
- *Relation-seeking tasks:* Find references or groups of references such that specified relations exist between the corresponding characteristics.

Synoptic tasks also fall into three classes:

- *Pattern identification:* Find subsets of references such that the behavior over those subsets corresponds to a defined pattern.
- *Pattern comparison:* Find specific relations between behaviors of subsets of references.
- *Relation-seeking:* Find occurrences of specific relations between behaviors and determine the corresponding reference sets.

More details about the definitions of tasks and subtasks, behavior and patterns can be found in [Andrienko & Andrienko, 2006].

Task used in this experiment

Hypothesis 1 is based on the premise that the combination of quantitative data with associated qualitatively abstracted data in one representation will not result in lower performance for lookup and comparison of qualitatively abstracted data. Therefore, the first blocks of tasks represents tasks solely concerned with qualitative abstractions of the data (cf. Table 5.1).

Hypothesis 2 is based on the premise that the combined representation does result in improved performance for tasks involving quantitative data within defined qualitatively abstracted levels. Consequently, task block 2 represents tasks that involve raw quantitative data associated to qualitative abstractions (cf. Table 5.2).

To make it possible for the test persons to perform tasks repeatedly in a session the selected tasks are abstracted from real-life tasks a medical expert would perform on that data.

The atomic subtasks in Table 5.1 and Table 5.2 are listed in the second column. Note that every task involves at least one elementary lookup subtask concerning qualitative attributes of the data to ensure the inclusion of the qualitative abstractions in the tasks. A complete list of the tasks realized on the datasets used in this experiment can be found in Appendix F.

The first 3 tasks in each block are representative for the lookup tasks and the last 3 tasks in each block represent comparison tasks, as they include at least one comparison subtask. Synoptic pattern search tasks are classified as lookup tasks in the second block, since synoptic pattern

	Nr	Subtasks	Task description	Number of variables
Lookup tasks	1	EIL	How many intervals of <qualitative level a> occur in <variable x>?	Single
	2	EIL	Mark the first interval where both variables <x> and <y> are within <qualitative level a>.	Multiple
	3	EIL	Mark the first appearance of an interval of <qualitative level a> in <variable x>.	Single
Comparison tasks	4	EDL + EC	<Variable x>: Is the <first> qualitative level in <week> higher/lower/equal than the <third> qualitative level?	Single
	5	EIL + SBCA + SBCO	Which variable has the longest lasting interval of <qualitative level a>?	Multiple
	6	EIL + SBCA + SBCO	Which variable has the most occurrences of <qualitative level a>?	Multiple

Table 5.1: Conceptual tasks involving only qualitatively abstracted data. The second column states the subtask types referring to the task taxonomy by Andrienko & Andrienko [2006] using these abbreviations: EIL = Elementary inverse lookup, EDL = Elementary direct lookup, EC = Elementary comparison, SBCA = Synoptic behavior characterization, SBCO = Synoptic behavior comparison. The last column states the number of involved variables for the task.

search tasks correspond to lookup tasks on the synoptic level (cf. [Andrienko & Andrienko, 2006]). All task descriptions are related to the currently visible time span for the visualization, if not stated otherwise.

5.4 Apparatus

The experiment was conducted on the same laptop with the same computer mouse for all test persons. The laptop was an Apple Mac Book Pro 4,1 with Mac OS X 10.6.7 as operating system. The mouse was a standard symmetrical shaped Logitech optical mouse. The test application was maximized on a 15.4" LCD screen set to 1440x900 resolution. The test persons used both, mouse and keyboard during the experiment.

5.5 Procedure

Every test person was given a short introduction to the purpose of the experiment before every session. The test persons were asked to fill out a short questionnaire containing questions

	Nr	Subtasks	Task description	Number of variables
Lookup tasks	7	EIL + SPS	Which variable is <rising> when <variable x> enters <qualitative level a> the <first> time.	Multiple
	8	EIL + EDL	What value has the next measured data point of <variable x> when <variable y> enters in <qualitative level a> the first time in <week>?	Multiple
	9	EIL + EDL	How many measured values contains <variable x> <first> interval of <qualitative level a>.	Single
Comparison tasks	10	EIL + EDL + EC	<Variable x>: Which interval of <qualitative level> contains the largest number of measured values?	Single
	11	EIL + EDL + EC + EC	Which variable has the <highest/lowest> measured value in its <first> interval of <qualitative level y>?	Multiple
	12	EIL + EDL + EC	Find the <highest > measured value in <variable x>'s <first> interval of <qualitative level a>.	Single

Table 5.2: Conceptual tasks involving qualitatively abstracted & quantitative data. The second column states the subtask types referring to the task taxonomy by Andrienko & Andrienko [2006] using these abbreviations: EIL = Elementary inverse lookup, EDL = Elementary direct lookup, EC = Elementary comparison, SPS = Synoptic pattern search. The last column states the number of involved variables for the task.

about personal information and self-assessment to computer experience and graph reading skills (cf. Appendix C). The trials were blocked by visualization type deploying a within-subject design and every test person received a training session before each experiment block. A training session started with an introduction of the visualization technique and the corresponding interactions, demonstrated by the test supervisor. After the introduction, the participants were instructed to solve the training tasks and encouraged to ask any questions during the whole training session, before advancing to the actual trials.

The visualization tool was presented in full screen to avoid distraction and to offer enough space for the visualization itself along with task description and answering possibilities. Before a trial began, a pop-up message appeared with the task description, hiding the current visualization state. The participants were instructed to read the task instructions carefully and then press an "Ok" button. This initiated a trial, causing the visualization to reappear and the timer to start for the given task. The task description was still visible on the right side of the visualization window, along with the answering possibilities for a given task. The participants then had to

perform the task by either selecting an answer from a list, marking a distinct time interval with the mouse or enter a number in an answering field with the keyboard. The interaction techniques were equal for both visualization techniques and included tooltips for distinct data points as well as for qualitative intervals and also resizing of the data panels with the mouse. The tasks are finalized by pressing the "Next Task" button, at which point the timer stops and the trial ends. Then a pop-up message appeared again with the description of the next task.

The actual state of the visualization (i.e. vertical size of the facets) was not being reset during a block to avoid confusion for the test person. Since it is possible that the visualization is moved to an unusable state, the test person could reset the visualization state with the help of a reset button at any time. Figure 5.1 and Figure 5.2 show the evaluation window for both visualization techniques including a task description and the task answering possibilities.

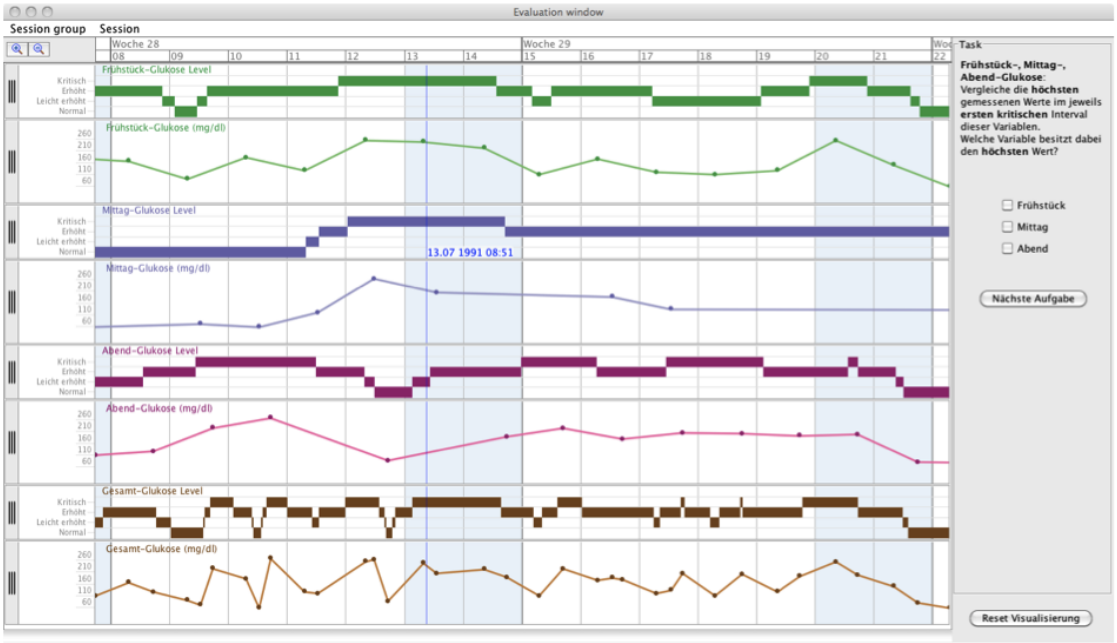


Figure 5.1: Screenshot of the evaluation window with the KNAVE visualization

For each trial task completion time and the given response from the test person was recorded to a comma-separated file. The following variables were stored for each task:

- Timestamp: the task start time
- Task number: the number of conceptual task (1-12)
- Task id: every concrete task has a unique number (realization of a conceptual task on a dataset)

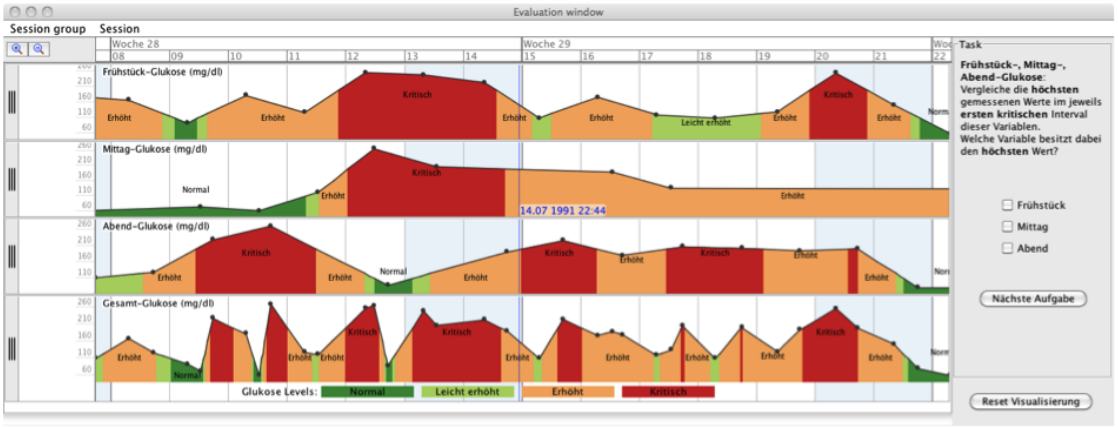


Figure 5.2: Screenshot of the evaluation window with the STZ visualization

- Visualization: the visualization used for a task (STZ or KNAVE)
- Dataset: the dataset used for a task (Dataset 1 or Dataset 2)
- Task completion time: in milliseconds
- Answering type: enter a numerical number (numerical), choose an answer from a list (list) or mark a time-interval (time)
- Task correctness:
  - For numerical tasks, a correct value with a tolerance for the correct answer were defined, if the answered value is outside of that tolerance, the question was assessed false
  - For a marked time-interval, a tolerance interval for both time points (start and end) was defined.
  - For a multiple choice task from a list, there is only one correct answer
- Given answer: the response from the participant
- Correct answer: the correct answer to verify the calculated task correctness

Every participant used one of the visualization techniques to master a set of 24 tasks with one dataset. Then the test person were offered the chance to take a break to stay alert and then continued to master another set of 24 tasks with the second visualization technique with another dataset. The test person also had the possibility to take a break after each task completion. After the test persons had finished both rounds of the experiment, they were asked to decide which of the visualization techniques they personally preferred over the other one. They were also asked

to provide some feedback about the two visualizations techniques and their interaction possibilities.

Every test person had to complete 24 tasks in one experiment block. The duration of completing a single task was expected to be between 30 seconds and one minute, resulting in approximately 45 minutes for the experiment itself. Adding 5 minutes for filling out the pre-experiment and post-experiment questionnaires and another 5 minutes for the training results in the total duration of about 1 hour for every test person. The procedure is outlined in Table 5.3. To verify these assumptions and to find flaws in the design, a pilot test has been carried out with one test person before recruiting the test persons for the actual experiment. The pilot test verified the experiment duration of one hour and did not reveal any flaws in the design. The average experiment duration throughout the study was between 45 minutes and 75 minutes. The variation of the experiment durations was mainly the result of different durations for the breaks between blocks or tasks.

Activity	**Time [min]**
Pre-experiment Questionnaire	5.0
Training Round One	5.0
Experiment Round One	22.5
Training Round Two	5.0
Experiment Round Two	22.5
Post-experiment Questionnaire	5.0
Total	**65.0**

Table 5.3: Overview of experiment procedure.

The collected data were checked for possible errors afterwards and preprocessed for further statistical analysis (cf. Appendix B). The goal was to find significant differences in task completion time and task correctness for a visualization technique with statistical hypothesis tests like the Student's t-test as suggested by Lazar et al. [2009], with visualization type as the factor. To detect or disqualify possible influence of other factors, t-tests were also were conducted using the following factors: dataset, experiment round and task number.

5.6 Participants

The potential participant pool consists of people with reasonable computer experience, and graph reading experience. I am defining reasonable computer experience as working with a computer more than 10 hours per week, since the tasks involve some interactions with the visualization tools (resizing facets, highlighting time-intervals and activating tooltips for data-points and qualitative intervals).

To ensure basic graph reading experience, the participants should either be students in their second-year or higher, or have to deal with graphical data representations frequently in their daily working routine (self reporting). It was planned to recruit around 20 participants, which is similar to comparable studies (cf. [Lam et al., 2007], [Javed et al., 2010] and [Ordóñez et al., 2010]).

Preconditions:

- Normal/corrected to normal vision (no color blindness)
- Reasonable computer experience (min. 10h/week)
- Graph reading experience (Line charts & bar charts)

5.7 Data

Every task type is defined for two datasets. The data come from the UCI Machine Learning Repository[1] (Diabetes dataset) and consists of blood glucose measurements for several patients. These data were selected because it is multivariate, temporal data and meaningful qualitative abstractions for blood glucose measurements exist. Also, the qualitative abstraction of these data should be easy to understand for non-experts in the medical domain. The datasets used in this study are subsets of these measurements from one patient over 4 weeks, and consists of 4 variables: overall blood glucose, pre-breakfast blood glucose, pre-lunch blood glucose and pre-supper blood glucose. The associated qualitative abstractions have been defined in agreement with a physician; the quantitative values can be grouped into four groups relating to hyperglycemia as listed in Table 5.4.

Qualitative abstraction	**Threshold values**
Normal	$<$ 110.0 mg/dl
Slightly elevated	$<$ 125.0 mg/dl
Elevated	$<$ 200.0 mg/dl
Critical	$\geq$ 200.0 mg/dl

Table 5.4: Threshold values for the qualitative abstractions of the quantitative data referring to hyperglycemia.

[1] http://archive.ics.uci.edu/ml/datasets/Diabetes, Retrieved 2011-09-14

5.8 Study design

In order to isolate the impact of individual differences of the participants, which are likely to appear due to the expected diversity of the qualified test candidates, and to increase the output of the test results, a within-subjects crossover design was selected. Following independent variables are included in this study:

- **Visualization technique (V)**: SemTimeZoom and KNAVE
- **Type of data (TD)**: Qualitative data and combined (quantitative values and qualitative abstractions)
- **Task number (T)**: 6 different tasks exist for each data type

The number of conditions in a factorial design is determined by the number and levels of the independent variables: V*TD*T = 2*2*6 = 24 different conditions. To increase robustness, every task is repeated, resulting in 48 different conditions for each test person.

Each participant had to perform every task with both visualization techniques. To minimize learning and fatigue effects, the order of the used visualization type was counterbalanced: one half of the participants first used the STZ technique to perform every task type and then used the KNAVE technique to perform every task type. The other half of participants first used KNAVE and then used STZ. To avoid learning effects due to the used dataset, two different datasets were used, each for one session with a visualization technique (cf. Appendix F). The order of the used dataset was counterbalanced within a visualization technique because of possible differences in difficulty of the datasets.

Table 5.5 illustrates the assignment order of visualization technique and dataset for each test person.

The order of the tasks was randomized, which yielded to an alternation of tasks involving qualitative and combined data. Also influences of certain sequences of tasks, which could be answered faster due to similar data in question, should be avoided by the randomization of task order.

Test persons	First round	Dataset	Second round	Dataset
1, 5, 9, 13, 17	STZ	Dataset 1	KNAVE	Dataset 2
2, 6, 10, 14, 18	KNAVE	Dataset 1	STZ	Dataset 2
3, 7, 11, 15, 19	STZ	Dataset 2	KNAVE	Dataset 1
4, 8, 12, 16, 20	KNAVE	Dataset 2	STZ	Dataset 1

Table 5.5: Assignment order of visualization technique and dataset for the test persons.

CHAPTER 6

Results

This chapter describes the detailed results of the evaluation study, which was presented in the preceding chapter.

6.1 Test Persons

20 test persons (12 male, 8 female) took part in the experiment. The test persons were all volunteers, not color blind and had normal or corrected to normal vision. The average age of the test-persons was 27 years and ranged between 22 and 30 years. Most of them were university students, with more than half from the Faculty of Informatics. Figure 6.1 gives an overview over the professions of the test persons.

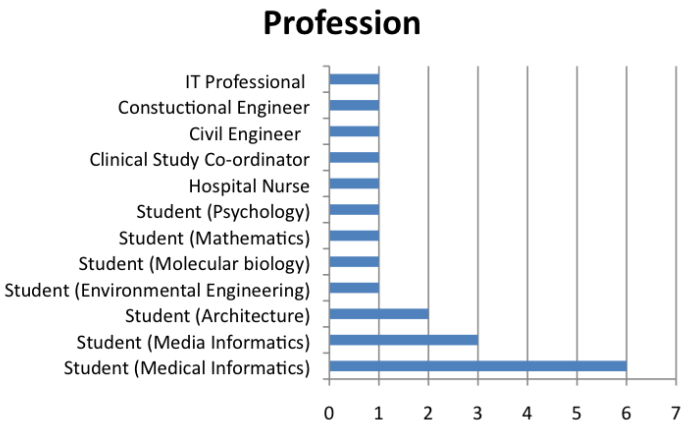


Figure 6.1: Profession distribution of the test persons

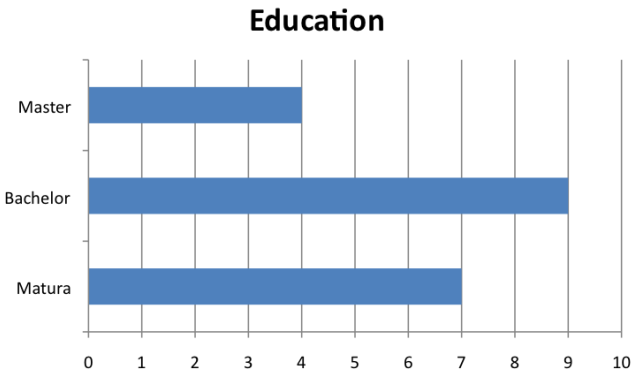


Figure 6.2: Education levels of the test persons

All test persons were at least in their second year of university or had reported to deal with graphical data representations frequently in their daily working routine.

Figure 6.2 shows the education levels of the test persons. The majority of the test persons had a bachelor's degree, four persons had a master's degree, and seven had a *Matura* (High-school leaving exam that must be passed in order to apply to a university or other institution of higher education).

Figures 6.3, 6.4 and 6.5 show the self-assessments of the test persons experience about line charts, bar charts, data analysis and computer experience.

Nobody did assess him or herself as inexperienced in any category and the experience level that was chosen most often for all categories was *good*, though computer experience had an equal distribution of *good* and *very good* and thus the category with the best self-assignment. This was probably the case because the majority of the test persons had an educational background in computer science or a related field.

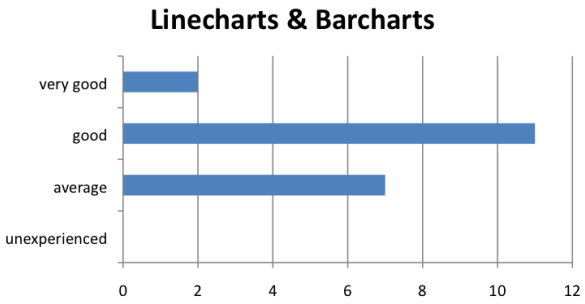


Figure 6.3: Experience levels with line and bar charts of the test persons

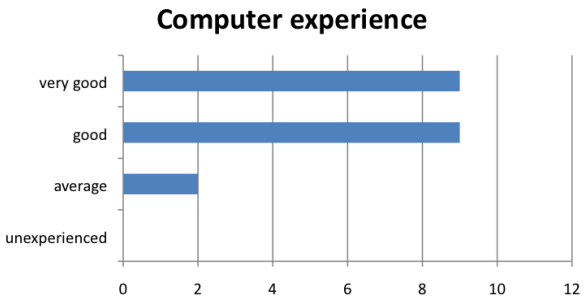


Figure 6.4: Experience levels with computers of the test persons

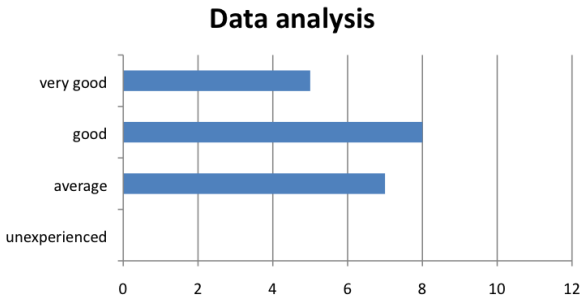


Figure 6.5: Experience levels with data analysis of the test persons

6.2 Data Analysis Approach and Results

After assembling the gathered data for further analysis (cf. Appendix B), the influence of the used dataset on timing was tested using a paired t-test. It was found that the time samples violated the normality assumptions of the t-test, so the logarithm of the times were used. This also makes sense in order to dampen the influence of overly long answering timings that would distort the results otherwise. The result of the t-test yielded no significant influence of the used dataset ($t(479) = 1.557$, $p = 0.12$, Cohen's $d = 0.071$). In Figure 6.6, the box plots of the logarithm of completion times for both datasets used in the experiment are shown. The correctness rate did not follow a normal distribution or log normal distribution, but a Mann-Whitney's U test (note: not paired because accumulated) also did not show a significant influence of the used dataset (The mean ranks of STZ and KNAVE were 23.8 and 25.2, respectively; $U = 271$, $Z = -0.37$, $p = 0.72$, $r = 0.053$). Hence, the following analysis will not take into account which dataset was used for the experiment trials.

Even though the order of the visualization types was counterbalanced to reduce possible learning effects or fatigue, the carryover effect seems unbalanced for visualization types (cf. Figure 6.7).

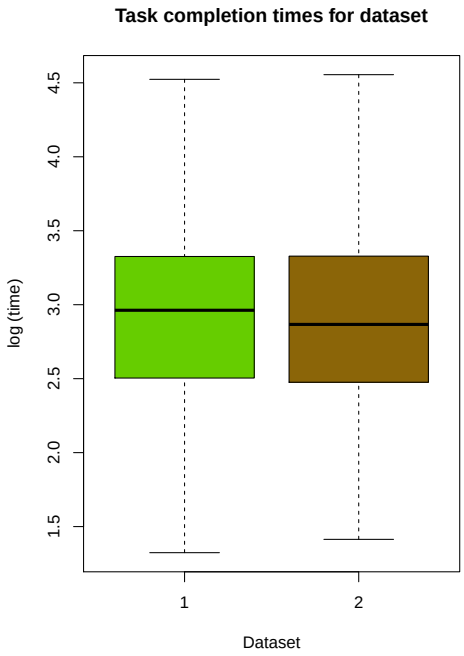


Figure 6.6: Box plots of logarithms of completion times separated by dataset.

On the one hand, the median of the completion time for STZ in the first round of the experiment was 17 seconds and in the second round 15.3 seconds resulting in an average improvement of 1.7 seconds. On the other hand, the median of the completion time for KNAVE in the first round was 24.9 seconds and in the second round 18.1 seconds with an average improvement of 6.8 seconds. Also, task completion times were considerably faster in the second round and therefore the completion times for each round needed to be compared separately, though the personal differences of the test persons will not be taken into account by this analysis.

Though the success rate is very high for each task with both visualization techniques, a Mann-Whitney's U test did show a significant influence of the experiment round (The mean ranks of STZ and KNAVE were 20.3 and 28.7, respectively; U = 186.5, Z = -2.2, $p<0.05$, r = 0.32). Therefore, success rate data were also analyzed separately for the first and second round.

Task completion times and error rates (1-success rate) have been aggregated for each task set according to Table 5.1 and Table 5.2 to test the hypotheses stated in Chapter 5. Completion times were summed up for each task set and error rates were calculated as ratio of errors to the overall number of tasks in a task set. Figure 6.8 and 6.9 show the completion time box-plots for each task set and visualization type in the first and second round.

Completion times for the task sets were tested for normal or log-normal distributions using the

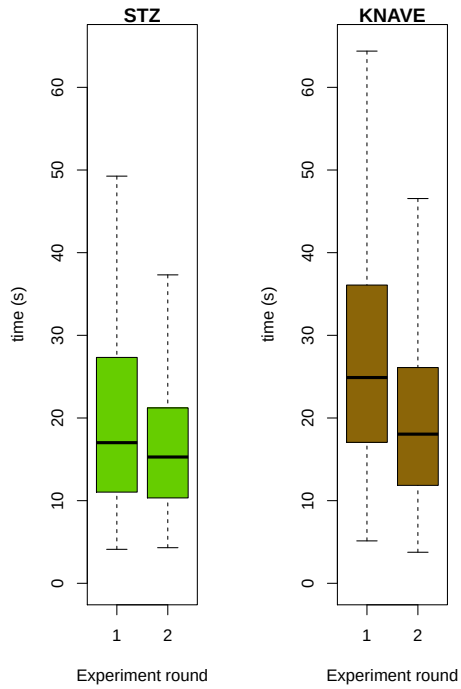


Figure 6.7: Box plots of the completion times separated by visualization type and experiment round.

Shapiro-Wilk test for every task set and visualization type. Task completion times tend to be right skewed [Sauro & Lewis, 2010]; presumably this is the reason that the completion times for all task sets follow a log-normal distribution. The logarithmized task set pairs of completion time also show equal variance for both visualization types in round 1 and 2, which was detected using an F-Test.

As a result, a t-test could be used to test significant differences of the logarithmized completion times for the task sets and thereby testing the hypotheses. Error rates have been quite low with both visualizations and do not follow a normal or log-normal distribution, therefore a non-parametric Mann-Whitney's U test was used to test the significance of error rates, since the error rate pairs for each task set did show equal variance for both visualizations. The R scripts that were used in this section to analyze the results can be found in Appendix D and E.

Hypothesis 1 – Qualitative Data

The first part of this analysis is focused on tasks involving only the qualitative abstractions of the data. In the case of this experiment, these tasks include questions regarding the temporal behavior, number of occurrences and ordinal characteristics of episodes of normal, slightly elevated, elevated and critical blood glucose measurements. Lookup tasks are analyzed separately from comparison tasks.

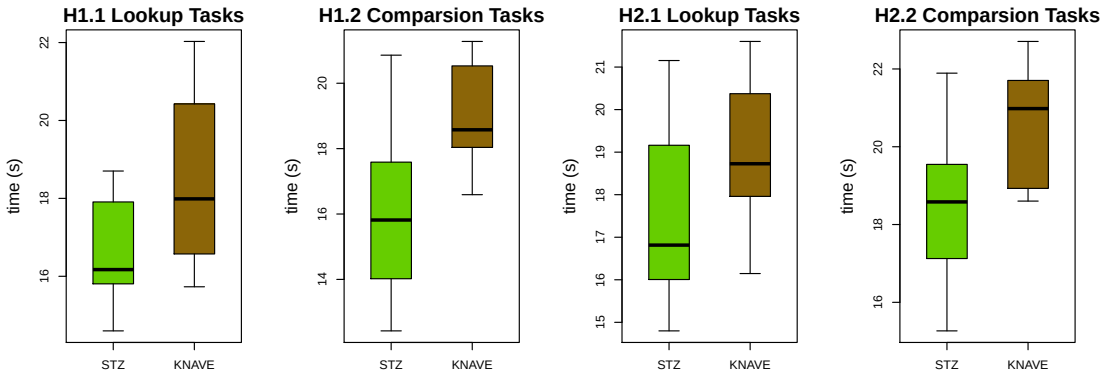


Figure 6.8: Box plots for completion time per task set in round 1.

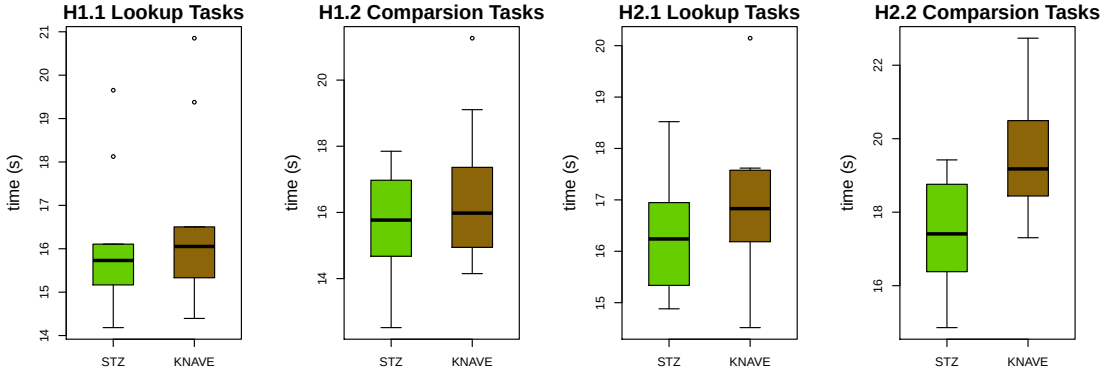


Figure 6.9: Box plots for completion time per task set in round 2.

Lookup Tasks

Table 6.1 reports mean and standard deviation of completion time along with the p-value of the related t-test for task set 1 for both rounds, whereat a green cell indicates that the p-value is found to be significant. Table 6.2 reports descriptive data on the relative number of mistakes (error rate) for task set 1 for both rounds. The table for error rates also includes the median because a Mann-Whitney's U test compares the medians in contrast to the t-test, which compares the means of two groups.

The maximum duration for the lookup task set is around 262 seconds (4.4 minutes) with average durations between 116 to 158 seconds in the first round and 99 to 109 seconds in the second round. Both test person groups performed faster in the second round, which can be attributed to

H1 Lookup Tasks	*Round 1*	*p-value: 0.02025*
STZ	mean: 116.3 sec.	std.dev.: 32.8 sec.
KNAVE	mean: 157.6 sec.	std.dev.: 60.6 sec.
H1 Lookup Tasks	*Round 2*	*p-value: 0.2857*
STZ	mean: 99.3 sec.	std.dev.: 30.0 sec
KNAVE	mean: 108.8 sec.	std.dev.: 39.8 sec

Table 6.1: Completion time of lookup tasks for qualitative abstractions.

H1 Lookup Tasks	*Round 1*		*p-value: 0.8646*
STZ	mean: 10.0%	median: 8.33%	std.dev.: 11.7%
KNAVE	mean: 8.33%	median: 8.33%	std.dev.: 8.8%
H1 Lookup Tasks	*Round 2*		*p-value: 0.5428*
STZ	mean: 1.67%	median: 0%	std.dev.: 5.3%
KNAVE	mean: 5.00%	median: 0%	std.dev.: 11.2%

Table 6.2: Error rates of lookup tasks for qualitative abstractions.

the learning effect. The test persons who used the KNAVE technique were on average slower than those who used the STZ technique. A one sided t-test showed a significant difference in completion time between the visualization types in round one ($t(15) = 2.2$, $p < 0.05$, Cohen's $d=1.00$) with STZ outperforming KNAVE. In the second round no significant difference between both visualization types ($t(17) = 0.6$, $p = 0.29$, Cohen's $d=0.26$) was found regarding completion time.

The error rates have an equal median for both visualization types in round one (8.3%) and two (0%); consequently no significant difference was found by a Mann-Whitney's U test between visualization types. Nevertheless, a learning effect is also evident in the error rates as the median is reduced from 8.3% to a 0% in the second round.

Comparison Tasks

Again, the average duration decreases per round for both techniques (cf. Table 6.3) for comparison tasks. The users of STZ were on average one minute faster to find answers than the KNAVE users in the first round and 16.5 seconds faster in the second round. In the first round, a one sided t-test revealed a significantly faster completion time for test persons using the STZ technique ($t(16) = 3.16$, $p < 0.01$, Cohen's $d=1.63$).

The test persons made on average 5% more errors with KNAVE in the first round but also 5% more errors with STZ in the second round. This could be grounded on the individual differences of the test persons, which had problems with this particular task set and made the same errors regardless of the visualization technique. Again, no significant difference was found on error

H1 Comparison Tasks	*Round 1*	*p-value: 0.003117*
STZ	mean: 107.5 sec.	std.dev.: 46.0 sec.
KNAVE	mean: 169.2 sec.	std.dev.: 53.0 sec.
H1 Comparison Tasks	*Round 2*	*p-value: 0.1697*
STZ	mean: 94.4 sec.	std.dev.: 24.0 sec
KNAVE	mean: 110.9 sec.	std.dev.: 44.4 sec

Table 6.3: Completion time of comparison tasks for qualitative abstractions.

H1 Comparison Tasks	*Round 1*		*p-value: 0.4653*
STZ	mean: 8.3%	median: 0%	std.dev.: 14.2%
KNAVE	mean: 13.3%	median: 8.3%	std.dev.: 17.2%
H1 Comparison Tasks	*Round 2*		*p-value: 0.5012*
STZ	mean: 10%	median: 0%	std.dev.: 14.1%
KNAVE	mean: 5%	median: 0%	std.dev.: 8.1%

Table 6.4: Error rates of comparison tasks for qualitative abstractions.

rates depending on the visualization technique in both rounds (cf. Table 6.4).
Hypothesis 1 expects that there is no difference in completion time and error rate for lookup and comparison task involving only qualitative data between STZ and KNAVE. This was confirmed for error rates, as there is no significant difference in both rounds and both task sets, nor had the mean and median of the error rates a trend in either direction. But it was observed that STZ performed significantly better than KNAVE in terms of completion time for both task sets in the first round and had on average better completion times in the second round, though no significance was found.

Hypothesis 2 – Qualitative & Quantitative Data

This part investigates the completion time and error rates for tasks involving quantitative data mapped to specified qualitative abstractions. Again, lookup tasks will be analyzed separately from comparison tasks.

Lookup Tasks

Table 6.5 shows descriptive data for the completion time of lookup tasks. In the first round, the test persons using KNAVE needed on average around 15% more time to master a lookup task than STZ users and 10% more time in the second round. The completion time was not found to be significantly faster for any visualization technique in the first round and second round.

Error rates do not have any significant differences; interestingly the mean of the errors rose in the second round compared to the first round with KNAVE. The medians of the error rates are

H2 Lookup Tasks	*Round 1*	*p-value: 0.06944*
STZ	mean: 138.3 sec.	std.dev.: 61.4 sec.
KNAVE	mean: 160.3 sec.	std.dev.: 46.1 sec.
H2 Lookup Tasks	*Round 2*	*p-value: 0.18965*
STZ	mean: 100.1 sec.	std.dev.: 18.8 sec
KNAVE	mean: 111.6 sec.	std.dev.: 36.8 sec

Table 6.5: Completion time of lookup tasks involving both; quantitative data and qualitative abstractions.

H2 Lookup Tasks	*Round 1*		*p-value: 0.3655*
STZ	mean: 6.7%	median: 0%	std.dev.: 11.7%
KNAVE	mean: 1.7%	median: 0%	std.dev.: 5.3%
H2 Lookup Tasks	*Round 2*		*p-value: 0.6231*
STZ	mean: 3.3%	median: 0%	std.dev.: 7%
KNAVE	mean: 6.7%	median: 0%	std.dev.: 11.7%

Table 6.6: Error rates of lookup tasks involving both; quantitative data and qualitative abstractions.

zero for both visualization types and rounds (cf. 6.6).

Comparison Tasks

Comparison tasks involving both, qualitative and quantitative data seem to be the most complex tasks, which is also reflected in the longest task completion times. The test persons were 40% to 45% faster with the STZ visualization than with KNAVE as can be seen in Table 6.7.

It is noticeable that the differences between the completion times in the first and second rounds stay relatively stable (68 seconds to 50 seconds) compared to previous tasks sets where duration differences were around three or four times higher in the first round than in the second. Completion time is also significantly faster with STZ in both rounds: $t(18) = 1.8$, $p < 0.05$, Cohen's $d=0.82$ (round 1) and $t(18) = 2.9$, $p < 0.01$ Cohen's $d=1.29$ (round 2).

Surprisingly, although these tasks are the most complex ones and had the longest completion times, the error rates do not stick out compared to the rates from the other task sets, neither with STZ nor with KNAVE (cf. Table 6.8). Once more, the error rates are lower in the second round but the median is constantly zero for both rounds and visualizations.
Hypothesis 2 proposes that the STZ visualization is more appropriate for tasks involving quantitative data within specified qualitative levels than the KNAVE visualization and should outperform the KNAVE visualization in terms of task completion time and error rate. This is confirmed

H2 Comparison Tasks	*Round 1*	*p-value: 0.009369*
STZ	mean: 154.2 sec.	std.dev.: 47.9 sec.
KNAVE	mean: 222.1 sec.	std.dev.: 51.4 sec.
H2 Comparison Tasks	*Round 2*	*p-value: 0.004515*
STZ	mean: 125.9 sec.	std.dev.: 34.0 sec
KNAVE	mean: 176.7 sec.	std.dev.: 47.2 sec

Table 6.7: Completion time of comparison tasks involving both; quantitative data and qualitative abstractions.

H2 Comparison Tasks	*Round 1*		*p-value: 0.9642*
STZ	mean: 8.3%	median: 0%	std.dev.: 14.2%
KNAVE	mean: 6.7%	median: 0%	std.dev.: 8.6%
H2 Comparison Tasks	*Round 2*		*p-value: 1.0*
STZ	mean: 1.7%	median: 0%	std.dev.: 5.3%
KNAVE	mean: 1.7%	median: 0%	std.dev.: 5.3%

Table 6.8: Error rates of comparison tasks involving both; quantitative data and qualitative abstractions.

regarding significantly shorter duration in both rounds for comparison tasks. Lookup task involving quantitative values did not have significant findings, though the durations were on average slower. The hypothesis was not confirmed regarding error rates, as no significant effect was found in both rounds for both task sets. Also, the error rates did not have a tendency to either visualization technique.

Results on Individual Task Level

The R scripts that were used in this section to analyze the results can be found in Appendix D.

Task Completion Times

An overview for the completion times separated by individual task number and experiment round can be seen in Figure 6.10 and Figure 6.11.

It is noticeable that in the first round of the experiment, every task has a faster mean completion time with STZ than with KNAVE, except for task number 4. Also in the second round, task number 4 has a longer mean duration with STZ. An explanation for the faster duration for task number 4 with KNAVE could be that this task is concerning the ordinal characteristics of the qualitative levels. The ordinal characteristics are not immediately visible in the STZ visualization and thus the completion of this task requires either user interaction with the data panel to change into the height coded qualitative mode or matching the colors of the qualitative level with

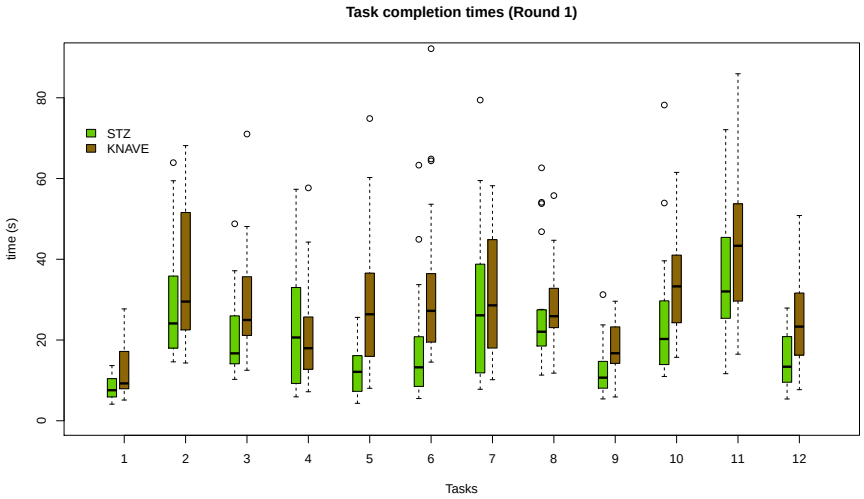


Figure 6.10: Box plots for completion times per task number in round 1.

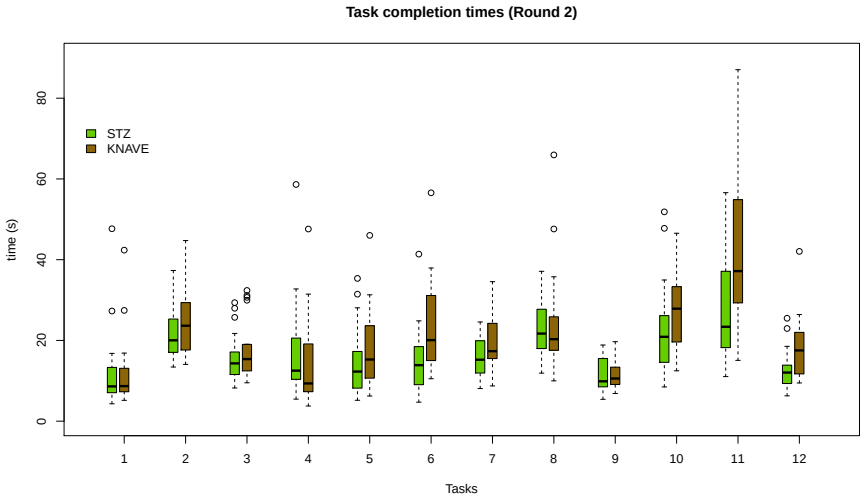


Figure 6.11: Box plots for completion times per task number in round 2.

the legend.

Every individual task has log-normal distributed completion times and equal variance between visualization types in each round.

In the first round, one-sided t-tests for every individual task revealed significant faster completion times with STZ for task numbers 1, 3, 5, 6, 9, 10 and 12. Analysis of the completion times in

	Task number	*p-value: (Round 1)*	*p-value: (Round 2)*
Qualitative Lookup	1	0.0042	0.3060
	2	0.0759	0.1752
	3	0.0054	0.1189
Qualitative Comparison	4	0.5299	0.8997
	5	<0.0001	0.1088
	6	0.0003	0.0023
Qualitative & Quantitative Lookup	7	0.1549	0.0138
	8	0.3023	0.5341
	9	0.0027	0.3365
Qualitative & Quantitative Comparison	10	0.0086	0.0311
	11	0.0952	0.0039
	12	0.0013	0.0030

Table 6.9: Results of one-sided t-tests on task completion on individual task level. A green cell indicates that the p-value was found to be significant in favor of STZ.

the second round showed significant faster completion times for task numbers 6, 7, 11 and 12 with STZ. An overview of p-values of the t-tests can be seen in Table 6.9, whereat a green cell indicates that the p-value is found to be significant. The only three tasks that are significantly faster in both rounds are task number 6, 10 and 12, noteworthy all three tasks include comparison sub tasks.

Task Success Rates

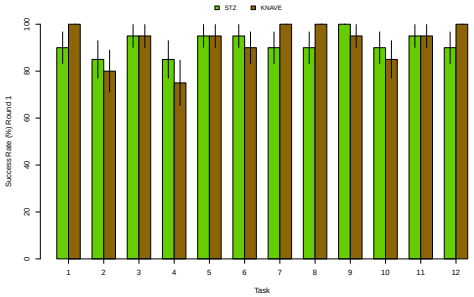
The success rate for each task number is shown for each visualization type per round in Figure 6.12(a) and 6.12(b).

Mann-Whitney's U tests were run to evaluate the difference between the success rate between the visualization techniques on individual task level separate for every round. The test did not reveal significant findings for any task in either round. Table 6.10 shows the p-values that resulted from the Mann-Whitney's U tests.

User Interactions

In addition to error rate and completion time, user interactions for each task have also been recorded. The recorded interaction log included activation of tooltips, marking of time intervals, resizing of data panels and concomitant with this, representation mode change in the STZ technique. The latter was intended to provide insight into which tasks need a representation mode change and if that has an impact on task completion times. But although the test persons where encouraged to use this feature in the training session and got a demonstration on how to use it,

(a) Round 1.



(b) Round 2.

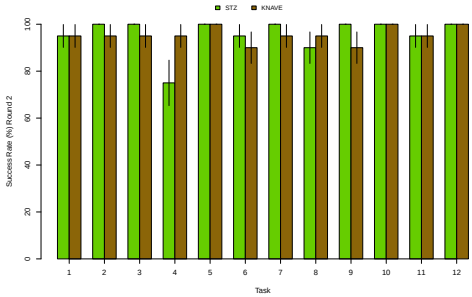


Figure 6.12: Success rate per task.

	Task number	*p-value: (Round 1)*	*p-value: (Round 2)*
Qualitative Lookup	1	0.487	1
	2	1	1
	3	1	1
Qualitative Comparison	4	0.695	0.182
	5	1	1
	6	1	1
Qualitative & Quantitative Lookup	7	0.487	1
	8	0.487	1
	9	1	0.487
Qualitative & Quantitative Comparison	10	1	1
	11	1	1
	12	0.487	1

Table 6.10: Results of Mann-Whitney's U tests on success rates on individual task level.

it was barely used in the experiment session. This was probably the case because all tasks could be mastered using the hybrid representation including colored qualitative regions below the line chart, which was the default mode at the start of the experiment. It was particularly not necessary to switch to the lifeline representation because there was plenty of space to display each variable in the hybrid mode anyway. Also, it is worth mentioning that two test persons activated the hybrid mode with level crossings for task number 9 but switched back to the default mode for subsequent tasks.

A Mann-Whitney's U test on the number of tooltips needed for each task was used between visualization types. The test showed that KNAVE users needed significantly less tooltips for task number 4, 8 and 7; STZ users needed significantly less tooltips for task number 6. Table 6.11 reports the average tooltip number per task for the two visualization techniques and overall,

Task number	*Overall*	*STZ*	*KNAVE*
11	8.09	8.08	8.11
8	3.41	4.11	2.71
6	3.20	1.79	4.61
12	3.13	3.05	3.21
4	3.09	4.08	2.11
7	2.87	3.42	2.32
5	2.39	2.13	2.66
10	2.36	2.34	2.37
2	1.96	2.05	1.87
1	1.63	1.50	1.76
9	1.62	1.95	1.29
3	1.13	1.18	1.08

Table 6.11: Average number of tooltips for each task sorted by frequency.

whereat the green cell indicates that the p-value is found to be significant.

The task number 11 has by far the most tooltip interactions throughout both techniques, which is not surprising because this task requires the user to compare multiple quantitative values of three variables. The significant difference of task number 4 between the visualization techniques may be attributed to the not immediately visible ordinal characteristics of the qualitative levels with STZ. It is also noticeable that comparison-tasks tend to have more tooltip interactions than lookup-tasks and also tasks involving quantitative values tend to have more tooltip interactions than tasks involving only qualitative attributes.

Feedback

After the test persons had finished both rounds of the experiment, they were asked to decide which of the visualization techniques they personally preferred over the other one. 19 out of 20 test persons preferred the SemTimeZoom visualization technique (cf. Appendix C). A Chi-square test revealed a significant difference for personal preference ($\chi 2 = 16.2$, $p < 0.001$). It may, however, be mentioned here that some test persons had problems in making a clear decision between the two techniques and finally chose STZ because it uses different colors for the qualitative levels. But it was also noticeable that some test persons, who completed the first round with KNAVE, already stated out loud during the STZ training session that they liked it much better than the first one because of the reduced clutter and better overview.

The test persons were also asked to provide some personal feedback and statements. On the one hand, statements regarding STZ can be boiled down to the prevailing view that the signalizing use of color for the qualitative levels and the aggregated visualization of quantitative and qualitative values in one diagram are helpful, particularly when comparing multiple variables. On

the other hand, some test persons rated KNAVE as more appropriate for counting of qualitative levels in one variable and regarded it as positive that the distinct colors for diagrams of the same variable eases the quick identification of a particular variable.

CHAPTER 7

Discussion and Outlook

The evaluation study described in this work compares two visualization techniques for displaying quantitative values and corresponding qualitative abstractions against each other by carrying out a controlled experiment. 20 test persons had to perform 12 different tasks (cf. Table 5.1 and 5.2) with each visualization technique in two rounds, whereas task completion time, error rate and user interactions have been recorded and analyzed with statistical methods.

The error rate was rather low throughout both visualization types and tasks. No significant difference could be found between visualization types for any task group defined in the hypothesis nor on an individual task level. This indicates that the test persons were equally careful, regardless of the visualization technique. I also believe that the error rates were rather low because of the basic nature of the tasks, which did not require the test persons to estimate values, and the answers could be found straightforwardly. I am attributing the reason for the mistakes that have still been made to carelessness or misinterpretations of task descriptions.

The dependent variable completion time revealed more interesting results with regards to visualization types. The analysis of the first round of the experiment has shown better results than initially expected as the STZ technique performed significantly better than the KNAVE technique for task groups defined for the first hypothesis. In other words, although the test persons performed these tasks significantly faster with STZ than with KNAVE, error rates were not worse than with KNAVE. In the second round, KNAVE did not perform better than STZ in terms of completion times. Consequently, the first hypothesis that proposed that STZ will at least perform equally with regards to completion time for tasks involving only qualitative attributes of the data was confirmed, although it was formulated cautiously and expected no significant faster completion times for the STZ technique.

The second hypothesis was only partly confirmed, as significantly faster completion times were found for comparison tasks of quantitative values mapped to qualitative abstractions for both rounds, but not for lookup tasks. Although there were no significant differences, it should be

added that the task completion time was on average faster in both rounds with the STZ technique (10-15%).

Additionally, the completion time for the visualization techniques have been analyzed on individual task level separately for each round. The results show that in the first round, 2 out of 3 comparison tasks involving multiple variables (Task 5 and 6), were completed significantly faster with STZ. In the second round, also 2 out of 3 comparison tasks involving multiple variables (Task 6 and 11) were completed significantly faster with STZ. A paired t-test disregarding the order of the experiment rounds did also reveal a significant shorter task completion time for all comparison tasks involving multiple variables (Task 5, 6 and 11) with STZ.

In summary, the results of the analysis of task completion time showed that the STZ visualization technique, despite using 40% less display space in the initial experiment setting, outperforms the KNAVE technique for comparison tasks involving quantitative values mapped to qualitative abstractions. Additional analysis on individual task level has revealed that comparison tasks involving multiple variables were also performed significantly faster with STZ. The KNAVE technique did not show a significantly faster effect on any individual task number nor on any task group relating to the hypothesis. The only task that was on average mastered faster with KNAVE than with STZ was task number 4. As already mentioned in Subsection 6.2, this task is the only one concerning the ordinal characteristics of the qualitative abstractions, which are not immediately visible in STZ. It is also suspected that the task description was misleading for some test persons, explaining the rather high error rate in the first round with both visualization techniques.

The test persons were also asked which visualization technique they preferred over the other one after they completed the experiment. The analysis of the personal preferences revealed a significant difference in favor of the SemTimeZoom technique.

The analysis of the interaction logs showed that the STZ visualization technique was more interaction-intensive than the KNAVE visualization technique, relating to the number of activated tooltips. This does not conflict with the idea of the STZ technique as an interactive visualization tool, although the test persons did hardly ever use the semantic zoom feature. The higher interaction activity in STZ is not reflected in increased completion times.

I believe that the combined visualization of the quantitative and qualitative aspects of a variable in one diagram excels especially for comparison tasks of quantitative values in defined qualitative levels due to reduced span between the different aspects for a variable. KNAVE requires the user's gaze to travel vertically between the diagrams that belong to the same variable to find the quantitative values that belong to a distinct qualitative area. This difficulty would probably increase, if the diagrams were not grouped together by variable like in the KNAVE experiment setting in this study. This belief is also supported by the *proximity compatibility principle*, which specifies that displays relevant to a common task or mental operation (mental proximity) should be rendered close together in perceptual space (close display proximity) [Wickens & Carswell,

1995] .

The second reason, in my opinion, for the better performance of STZ over KNAVE is the use of distinct signalizing colors for different qualitative levels because the features color hue and intensity are preattentively processed and "pop out" from their surroundings [Ware, 2004]. This advantage was also pointed out by several test persons after the experiment.

7.1 Limitations

The interaction logs revealed that the test persons hardly ever changed between the representation-modes with the STZ technique (i.e. resizing of data panels), which is in fact leading to a conversion of the experiment to a comparison study between the hybrid-representation with filled qualitative regions used in STZ with the KNAVE visualization. Nevertheless, this also showed that almost all the time, the occupied display space for the STZ visualization was 40% less than the space that was occupied by the KNAVE visualization, which was the initial experiment setting.

Another limitation of the study was the relatively low number of subjects used in the experiment. Though the study was initially planned as a within-subject experiment, the analysis showed that the differences between the first and second round of the experiment were unbalanced according to the learning effect for task completion times and error rate. Possibly the training sessions have been too short to understand the visualization techniques completely. Consequentially, the rounds where analyzed separately as a between-subject design for each round. Of course, this also reduced the size of the groups for each round to the half of the initial group size of 20. A larger number of test persons would have improved the statistical power of the results and maybe resulted in clearer results.

Furthermore, task number 4 showed an unusual behavior, both in completion time and error rate. The instructions for the test persons seem to have been confusing for some test persons and should have been explained more clearly. From the visualization design point of view, no labels for the LifeLines in the KNAVE visualization have been used, which maybe introduced some disadvantage for the KNAVE technique, although no labels are used in the original visualization technique of the KNAVE project.

Limitations of the STZ technique

Another limitation of this study is that the STZ technique does currently only support rather simple qualitative abstraction with threshold values for a single quantitative variable. The KNAVE framework includes a computational module, which processes information from a temporal-abstraction mediator to calculate specified qualitative abstractions from the data [Shahar et al., 2006]. Qualitative abstractions can be dependent of frequency, trends or distinct patterns of a quantitative variable [Catley et al., 2008]. An example for such a pattern would be a shift in heart rate to above 185bpm for a period of ten minutes or more. It is possible to use the STZ

visualization technique to display such abstractions, though those abstractions have to be calculated and managed by an external data abstraction module like that presented in the KNAVE project, RÉSUMÉ or ASGAARD (see [Shahar et al., 2006], [Shahar & Musen, 1993] and [Seyfang et al., 2001]). It would also be essential to show the user a detailed explanation of such abstractions, because they are not as self-explaining as the abstractions depending on values that are exceeding certain thresholds. A possible visualization of the explanation of qualitative abstractions is presented in [Klimov et al., 2010].

Despite this, qualitative abstractions can also depend on more than one quantitative or qualitative variable. An example would be the bone-marrow toxicity that was used in an evaluation of KNAVE [Shahar et al., 2006], which depends on 3 clinical parameters. To extend the STZ technique with multiple dependants, a shared display space technique for the hybrid representation of the dependants like *Multiple Line Graphs*, *Stacked Graphs* or *Braided Graphs* [Javed et al., 2010] would be reasonable.

One also has to keep in mind the cultural meaning of the colors and the limited number of preattentive distinguishable colors. According to Healey [1996] only between five and seven different colors can be identified rapidly and accurately.

7.2 Outlook

This section presents some suggestions for further research in the field of the STZ visualization technique.

The aim of this study was to compare the STZ technique with a commonly accepted visualization method. The results of this study suggest a benefit of STZ, as it performed equally or even better for all tasks in terms of completion time than the compared technique (KNAVE). To ensure a fair comparison, the numbers of the variables has been limited to four, which was the maximum number of variables that could be displayed with the KNAVE technique on the available display space. To assess the full ability of the STZ visualization technique, as it was designed to visualize a large number of time series data, it needs also to be evaluated with larger number of variables.

Also, this study did not provide much insight about the underlying interaction technique for the semantic zoom feature because the test persons had no necessity to use it. Despite this, the test persons were asked to provide their view on the used interaction technique of STZ after the experiment. Although most of them deemed it as practical, some of them suggested they would like to have the possibility to vertically resize a group of variables at the same time and thereby change their representation mode. A possible extension of the interaction technique could be the use of accordion drawing as presented in [Munzner et al., 2003] for the resizing of the data panels. Additionally, the use of buttons to resize the visualization of a variable vertically to the next representation mode in addition to adjusting it continuously with the mouse was considered as a nice enhancement. One test person also pointed out the possible benefit of a filtering function

for the qualitative levels in order to display only a subset of the available qualitative levels that are important to the user at a particular moment.

Lam et al. [2007] investigated the impact of selective display of high-VIR details on low VIR interfaces for single level data. They found that a multiple VIR interface did not enhance visual search over using a single high VIR interface. It would be highly interesting to know whether a similar experiment with multiple level data would have the same outcome. For example, the study could compare the STZ technique with semantic zooming (selective display of high VIR details) with the standalone hybrid representation (single high VIR interface) using large number of variables.

What might also require a follow up testing is a comparison of STZ with a variant of the KNAVE visualization additionally using color-coding for the qualitative attributes of the data, even though this would introduce a redundant visual coding (spatial position & color). It can be argued that the STZ also uses redundant encoding because of the height of the horizontal bars or the area below the curve.

Another aspect that has not been covered in this study is that the hybrid-representation with filled qualitative regions used in STZ emphasizes higher quantitative values because of the larger colored areas below the curve. It could be interesting to examine if this influences the identification of distinct qualitative levels. In parallel, it would be necessary to conduct experiments to find the optimal heights for the representation transitions in STZ, since currently some heights for the transitions are only fairly arbitrary chosen (cf. Section 3.5 on page 22).

CHAPTER 8

Conclusion

I compared two visualization techniques capable of displaying time-oriented quantitative data of multiple variables and corresponding qualitative abstractions of the quantitative values using two different task blocks. The first task block addressed only the qualitative attributes and the second block additionally addressed the quantitative attributes of the variables. Both task blocks were split up into lookup and comparison task sets.

One visualization technique used color-coding to display the qualitative attributes and spatial position coding for the quantitative attributes in a combined representation of both data attributes. The second visualization technique uses separate representations for quantitative and qualitative data using spatial position coding for both data attributes.

The analysis of the task completion times showed that the test persons were generally faster completing the tasks, using the visualization technique with a combined color and position-coded representation, compared to a separate visualization using spatial position to encode both attributes. Particularly for more complex tasks, involving comparison subtasks of quantitative values within specified qualitative levels, the difference of the completion times was found to be statistically significant. Also, the faster completion times did not affect the correctness of the tasks.

Despite the ranking of Mackinlay [1986], which implies that information encoded by spatial ordering is more accurately perceived than other encodings such as color, size, or orientation, I conclude that if a variable contains different attributes (e.g. quantitative and qualitative attributes), different visual encodings should be used to represent the different attributes. Color hue is very well suited for displaying nominal characteristics of the data. If it is necessary to additionally display the ordinal ranking of qualitative data, color intensity and brightness can be used to encode this ordinal ranking [Harrower & Brewer, 2003].

Using separate representations for different (e.g. qualitative and quantitative) attributes of the

data results in greater movement by the head and eyes, because the user has to look for potential targets in different places. Thus, combined displays following the *proximity compatibility principle* [Wickens & Carswell, 1995] and displaying all relevant attributes of a variable in one representation should be used for multilevel data, if possible. The evaluation presented in this work showed that a combined representation particularly excels for more complex tasks involving both lookup and comparison subtasks of qualitative and quantitative attributes in one or more variables.

Future studies of the usefulness of the semantic zoom feature of the *SemTimeZoom* technique to display a large number of time series data are necessary to encourage the promising results of this study.

Part II

Evaluation Library (EvalBench)

CHAPTER 9

Introduction

This part of the thesis describes the primal structure of an evaluation class library that evolved in the process of this work and is tailored to the needs of carrying out task-based controlled experiments in the HCI field that can be used for visualization tools or interaction techniques. At the beginning, the evaluation functionality used in a study of the *indexing method* for line graphs [Aigner et al., 2011] has been examined for reusability and was the starting point of this library. Care has been taken to pave the way for future experimenters to enable reuse and adoption of the library for their special needs.

The library was developed in the *Java* programming language but the structure described in the following could be used in any other object oriented programming language, too.

9.1 Motivation

Researchers in the field of information visualization have long identified the need to evaluate their visualization tools and prototypes to present measurable benefits to encourage more widespread adoption [Plaisant, 2004]. Yet, the difficulty of conducting these evaluations remains a common topic [Lam et al., 2011]. There is a need for a solid evaluation infrastructure to encourage information visualization researchers to carry out an evaluation of their tools and ideas. To stimulate the effort on this issue, the researchers need solutions how to integrate different methods for evaluation into their prototypes and how to collect and measure the data produced by the users participating a study.

9.2 Related Work

Mackay et al. [2007] did related work, but they were focused to provide a platform for designing controlled experiments in the HCI field in the first place. Even though they also developed a

run platform that runs experiments created with their *design* platform, the architecture of the *run* platform is presented only in broad outline and the source code or documentation was unfortunately not available from their project website[Appert, n.d.] for further investigation. The *run* platform consists of a separate experiment launcher that reads a script that was created with the *design* platform and controls the flow of the experiment with the help of a state machine. The script may contain references to *experiment components* that can be registered to the launcher. These components are Java objects that are loaded dynamically during an experiment. They also designed the TDE (Touchstone Development Environment) for the development of the mentioned components that have to implement defined interfaces. Additionally, the launcher collects data measures and outputs them to log files for analysis purposes later on. Though the idea of a separate experiment launcher with registered components is a somewhat different approach to facilitate evaluation than the one presented in this work, the thought of introducing a tool to design experiments that can be imported into the experiment environment could be picked up and assembled with the library presented in the following chapters in future work.

The following chapter describes the individual components of the library. After that, the overall structure of the library and the interaction between the individual components are presented. Moreover, the usage of the library will be demonstrated by a small example.

CHAPTER 10

Individual Components of the Library

Figure 10.1 tries to communicate the working principle of the library. The visualization tool that needs to be evaluated has to implement the EvaluationDelegate interface in order to react on the different states of the experiment (e.g. use a distinct dataset for the visualization for a certain session or task). The library is capable of importing tasks defined in an external file for an experiment session. During the execution of the experiment, the run-time attributes of the evaluation and the interaction log are stored separately for each experiment session. This chapter introduces the individual components that are used within this library.

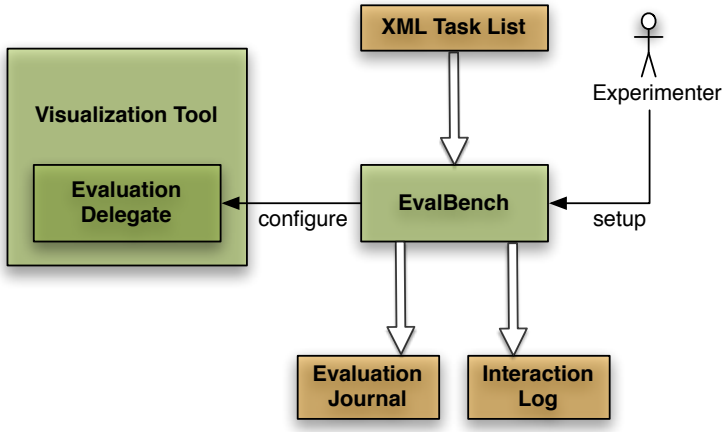


Figure 10.1: Simplified representation of the EvalBench working principle

10.1 Data Model

The first step towards a general library for task based evaluations was to create a data model that reflects the individual parts of a task based controlled experiment in the HCI field.

Task (Trial)

The finest granularity in the scope of a controlled experiment is a single task or trial a test person has to complete. A task contains the description for what needs to be done by the test person, the correct answer for the task, etc. *(design time attributes)*, but it also records the user performance of the task execution during an experiment *(run time attributes)*. Although, a task can take various forms, I tried to gather the general attributes, which should be consistent through every type of task. In the context of this library, a task type defines the type of answering possibility for the task (e.g. numerical, multiple-choice, etc.)

Design time attributes

These attributes have to be defined by the experimenter before the execution of the experiment.

- *Task id:* Unique identifier of a task.
- *Task category:* A short textual description of a task which can be used to identify the character of the task in logs and later analysis (E.g. "Lookup task").
- *Task description*: Specifies which action the user has to perform to get to the next step (E.g. "Find the highest value in the first critical elevated interval.").
- *Task instruction*: Additional instructions which can be shown optionally to help the test person to accomplish the task (E.g. "The red colored filled region below the chart indicates that these values are in a critical elevated range").
- *Task configuration(s):* It is necessary to define various configurations for a task, e.g. for which dataset the task has been defined or which visualization mode should be used for a certain task.
- *Correct answer(s):* This attribute defines the correct answer for the task to calculate the correctness of the task after the execution. This attribute varies between the task types and may consist of several attributes (e.g. correct value and tolerance for a numerical task)

Run time attributes

These attributes are set during the execution of the experiment.

- *Start date:* A timestamp representing the start event of a task, i.e. the date when the test person starts to work an a task.
- *End date:* A timestamp representing the event when a task was finished, i.e. the date when the test person has marked the task as finished.

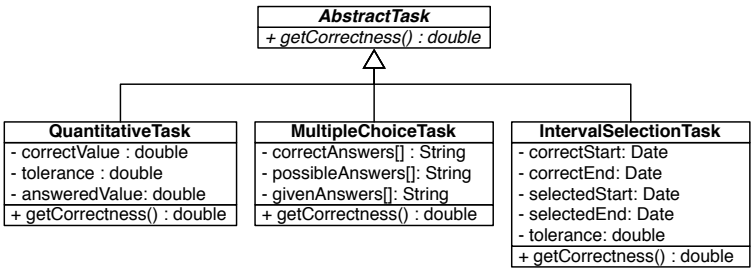


Figure 10.2: Simplified representation of the abstract task class and concrete subclasses implementing the abstract method `getCorrectness()`. Subclasses have to implement this method because the calculation of the correctness is specific for each task type

- *Given response(s):* The answer(s) that was given or selected by a test person for a task.
- *Task correctness:* This attribute represents the correctness of the given response (answer) for a task. The correctness is computed by comparing the defined correct answer with the given response by the test person.

Implementation

The abstract class Task provides the base class for every task type (relating to the task answering possibility). This class includes all attributes that all task types have in common *(Task id, category, description, instruction, start date, end date)*. Each task can have different configurations (e.g. the corresponding dataset) that can be assigned to a hash table as key/value pairs.

Because the definition of the correct answer and calculation of the correctness is specific for each task type, the task base class is only capable of recording the task completion time. The problem of the task correctness calculation is left to subclasses of the abstract class Task.

Predefined subclasses are quantitative tasks, multiple-choice tasks and interval-selection tasks. Quantitative tasks can be answered by specifying a number, multiple-choice tasks are answered by picking one or more predefined answers and interval-selection tasks are answered by selecting a distinct time-interval on a timescale.

Because the kind of answering possibilities vary between task types and hence another method to calculate the correctness of the given response from test persons has to be used, we ask the subclasses themselves to perform the calculation. The abstract class Task provides an abstract method, which has to be implemented by every subclass, regarding the calculation of correctness for the given answers. Figure 10.2 shows an example of concrete task classes, which implement the `getCorrectness()` method, other methods or attributes are omitted in the diagram for clarity. Every concrete subclass is responsible of providing a way to store the correct answers and the given response from the test persons to enable a calculation of the correctness. The

built-in subclasses are explained in detail to achieve a better understanding of the data model.

The subclass QuantitativeTask is initialized with a correct value and its corresponding unit (e.g. mg/dl) and tolerance value for the correctness calculation. The answered value must be set at the end of the task execution. The method `getCorrectness()` checks if the answered value is within the tolerance limits of the correct value and returns 1 if this is true, otherwise 0. If a more detailed calculation of the correctness is needed, one could return a value between 0 and 1 to represent the percentage of correctness.

MultipleChoiceTask is initialized with an array of strings containing all possible answers and an array of correct answers for this task. In addition, it has to be specified if the task allows only a single answer or multiple answers to be selected by setting the singleChoice flag. Again, the response of the test persons has to be set during task execution (i.e. an array of selected answers) and the correctness is calculated by comparing the array of the selected answers with the correct array. If the arrays contain the same elements, the method `getCorrectness()` returns 1, otherwise 0.

The last built-in task type is the IntervalSelectionTask, which is designed to compare a defined time interval with a time interval selected by a test person. It is initialized with the correct start date and end date of the interval asked for, additionally a tolerance in milliseconds can be specified. The method `getCorrectness()` compares the start and end date that was set during task execution specified by a test person with the correct dates. If both dates are within the tolerance limits of the correct dates, 1 is returned, otherwise 0.

Session (Block)

One step higher in the hierarchy of controlled experiments is a collection of tasks, usually grouped by a certain factor that is a subject of study (e.g. visualization technique or object size). EvalBench provides the class EvaluationSession for this purpose. It holds an array of tasks and manages the order of execution for the tasks; currently sequential or randomized order is supported. Again, an EvaluationSession can have several configurations, stored in a hash table as key/value pairs.

Session Group

Usually, a controlled experiment consists of several sessions for each test person. For example, if the experimenter chooses a *within-subjects* design, i.e. every test person is exposed to every experiment condition. Furthermore, the experimenter may want the test persons to perform a training session before advancing to the actual experiment session. For this reason, sessions can be aggregated to session groups for each test person. Currently, a session group is not capable of holding other instances of session group to structure an experiment into several session groups because it was not considered as necessary for standard experiments. Though, if an experimenter wants to run several session groups for a test person, this could be achieved by following the `Composite` pattern (cf. [Gamma et al., 1995]) to extend the data model in that way. Also,

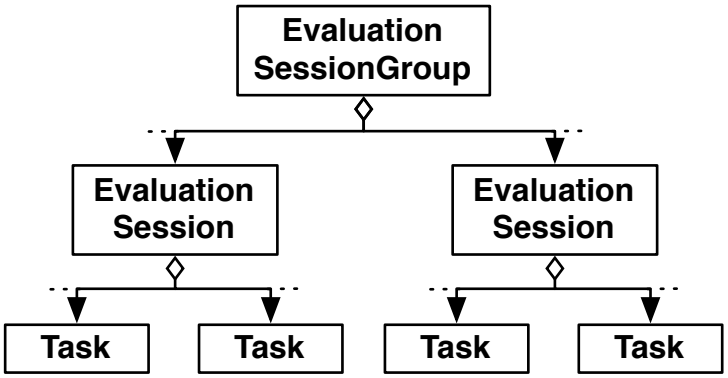


Figure 10.3: Underlaying data model of EvalBench

session groups currently do not use configurations.

Figure 10.3 illustrates the hierarchy in summary for the data model that forms the base of the library.

10.2 Data I/O

Task List Creator

To elude hard-wired task lists in the program code, a TaskListCreator is available that is responsible to load a list of tasks for a distinct session from the file system or a database. It is possible to use any file reader or database connector module to load the tasks in any desired format or creating the list manually in the code by implementing the TaskListCreator interface. But, since the built in task types use JAXB annotations to map to an XML schema, it is possible to utilize the Java Architecture for XML Binding (JAXB)[1] to unmarshal XML data into Task instances. Listing 1 shows an excerpt from a task list in XML format, composed of the predefined task types that can be used as input by the TaskListCreator implementation XMLTaskListCreator. The complete *Document Type Definition* for the XML files that can be understood by the XMLTaskListCreator can be seen in Listing 2. If additional Task types are needed, these subclasses only have to implement the necessary JAXB annotations to be used by the existing XMLTaskListCreator and the *Document Type Definition* has to be updated. These annotations also come handy if the experimenter wants to serialize the tasks in an XML file after completion for further analysis.

[1] http://jaxb.java.net/, Retrieved 2011-10-16

```
<tasks>
  <quantitative>
    <taskId>01</taskId>
    <taskCategory>01</taskCategory>
    <taskDescription>How often is pre-supper blood glucose ..
    </taskDescription>
    <taskInstruction/>
    <unit>times</unit>
    <isInteger>true</isInteger>
    <correctValue>2</correctValue>
    <tolerance>0.0</tolerance>
  </quantitative>
  <choice_selection>
    ...
    <possibleAnswers>
      <possibleAnswer>higher</possibleAnswer>
      <possibleAnswer>equal</possibleAnswer>
      <possibleAnswer>lower</possibleAnswer>
    </possibleAnswers>
    <correctAnswers>
      <correctAnswer>lower</correctAnswer>
    </correctAnswers>
    <singleChoice>true</singleChoice>
  </choice_selection>
</tasks>
```

Listing 1: Excerpt of a XML file that can be understood by the XMLTaskListCreator

```
<!ELEMENT tasks ( choice_selection | interval_selection |
   quantitative )* >
<!ELEMENT choice_selection ( taskId, taskCategory, taskDescription,
   taskInstruction, possibleAnswers, correctAnswers, singleChoice ) >
<!ELEMENT taskId ( #PCDATA ) >
<!ELEMENT taskCategory ( #PCDATA ) >
<!ELEMENT taskDescription ( #PCDATA ) >
<!ELEMENT taskInstruction ( #PCDATA ) >
<!ELEMENT possibleAnswers ( possibleAnswer+ ) >
<!ELEMENT correctAnswers ( correctAnswer ) >
<!ELEMENT singleChoice ( #PCDATA ) >
<!ELEMENT correctAnswer ( #PCDATA ) >
<!ELEMENT possibleAnswer ( #PCDATA ) >
<!ELEMENT interval_selection ( taskId, taskCategory, taskDescription,
    taskInstruction, intervalStart, intervalEnd, tolerance ) >
<!ELEMENT intervalEnd ( #PCDATA ) >
<!ELEMENT intervalStart ( #PCDATA ) >
<!ELEMENT tolerance ( #PCDATA ) >
<!ELEMENT quantitative ( taskId, taskCategory, taskDescription,
   taskInstruction, unit, isInteger, correctValue, tolerance ) >
<!ELEMENT correctValue ( #PCDATA ) >
<!ELEMENT isInteger ( #PCDATA ) >
<!ELEMENT unit ( #PCDATA ) >
```

Listing 2: Complete *Document Type Definition* that can be understood by the XMLTaskListCreator

Evaluation Journal

It is essential to record a protocol of the controlled experiment sessions for further analysis, usually with the help of statistical methods. A common practice in evaluations of user performance (cf. [Lam et al., 2011]) is to record task accuracy and task completion time. In some experiments (e.g. [Ordóñez et al., 2010] and [Biffl et al., 2005]), the experimenters did record task completion time manually, e.g. by asking the test persons to enter the time started and the time completed for each task. Apart from the fact that this is cumbersome for the test persons, it is also error-prone since the test person can forget to enter the start or completion date or enter the false dates out of unconcern. This would lead to skewed results in the analysis, especially for short task durations. Also, calculating the correctness for the tasks manually after finishing the experiment takes up a great deal of time and seems unnecessary since the experiments are done on computers anyway. For this reason, every EvaluationSession holds an instance of EvaluationJournal, which is in charge of saving all relevant data for every task after completion to a file.

The readily available implementation in the EvalBench library writes the data for each task within an experiment session to a comma-separated values (CSV) file that is ready to be imported into a statistics package such as R or SPSS (cf. Figure 10.4). It records all run-time attributes of the tasks like task completion time, correctness and given answer(s), along with

task	taskType	duration_ms	correctness	givenAnswer	correctAnswer	visType	dataset
13	7	12103	1.0	295.0	295.0	STZ	2
17	9	6257	1.0	1.0	1.0	STZ	2
4	2	17160	0.0	05:44:03-08:1	05:07:00-09:(	STZ	2
24	12	13019	1.0	Frühstück	Frühstück	STZ	2
11	6	5700	1.0	Gesamt	Gesamt	KNAVE	1
15	8	27872	1.0	162.0	162.0	KNAVE	1
18	9	12514	1.0	1.0	1.0	KNAVE	1
22	11	18752	1.0	Abend	Abend	KNAVE	1

Figure 10.4: Screenshot of an evaluation journal opened in Microsoft Excel.

some important design-time attributes like task id, task category, task description and correct answer(s) (cf. Section 10.1) to enable verification and analysis of the data according to these attributes. Additionally, every configuration that is defined in the configuration hash table of the Task will be recorded. To allow a consistent format throughout the CSV file, every Task has to have the same configuration keys within a session. If different configurations for the tasks are needed within a session, it is advised to use a proprietary EvaluationJournal implementation to save the task attributes in another format; the XML format would present itself as a good alternative due to the already implemented JAXB annotations in the built-in Task types (cf. Section 10.2). If necessary, the journal can be configured to take a screenshot of the current desktop after task completion.

Interaction Logging

EvalBench is capable to record a separate interaction log for each Session to enable interaction log analysis for each experiment session. The library uses the *Apache* log4j[2] logging utility to record user interactions. The logging can be configured using an XML file. This file specifies different Loggers, Appenders and Layouts. Loggers are logical names of logger instances that are known to the Java application and can be configured separately. Each logger can be configured as to what level of logging it should log (OFF, FATAL, ERROR, WARN, INFO, DEBUG & TRACE). The output of the logging can be configured by specifying Appenders like FileAppender, ConsoleAppender etc. The format of the log output can be configured by specifying Layouts for each Appender. The current implementation of the library logs user interactions with the root logger in the FATAL level to ensure visibility. Currently, the only Appender configured for the output is a FileAppender, which saves all interactions for an active Session to the same directory as the EvaluationJournal. This appender saves the log entries in the PatternLayout, which is an *one-line-at-a-time* format. Additional Loggers, Appenders and their format and can be set in the configuration file `evaluation.properties` that needs to be in the root directory of the visualization tool. Additional information on how to use the internal logger can be found in Chapter 11.

[2] http://logging.apache.org/log4j/index.html, Retrieved 2011-10-16

10.3 User Interface

Task Panel Factory

To accomplish a particular task, some kind of user interface needs to be provided for the test persons in order to display what they need to do and input options to specify the answer. Thus, EvalBench has a built-in TaskPanelFactory, generating an appropriate user interface panel for every subclass of Task. Basically, a panel consists of five elements: a task description field, an optional task instruction field, a task-answering field, a control field to display an error message for insufficient input and a button to mark the task as finished (cf. Figure 10.5).

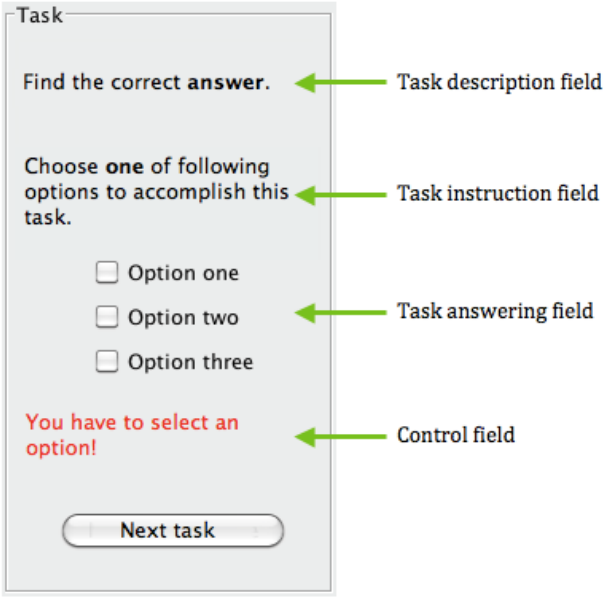


Figure 10.5: Example of a TaskPanel for a multiple choice task.

Assuming that subclasses of Task only differ in the form of how to set the answer, a task panel is the same for every type of task, except the answering field. The TaskPanelFactory is responsible of creating a new panel for every Task instance with a corresponding answering field. The standard TaskPanelFactory in EvalBench is the DefaultPanelFactory, which delivers a predefined panel for the task types QuantitiveTask and MultipleChoiceTask. If other types of tasks are needed for an experiment, the experimenter may want to extend the DefaultPanelFactory by overloading it, to deliver the corresponding panel for additional types of Tasks. For example, the DefaultPanelFactory does not support the creation of a TaskPanel for the task type IntervalSelectionTask, since the process of selecting a distinct time-interval in the user interface of a visualization tool is specific for each application and can therefore not be generalized for all applications.

Since the generic TaskPanel does not know how to set the answer for a given Task, the TaskPanelFactory uses the `Strategy` pattern (cf. [Gamma et al., 1995]) to solve this problem: the TaskPanel uses a distinct strategy for each task type. A strategy is responsible of providing a tailored answering field for a task, check if the test person did provide enough input to consider a task as answered and to record the answers that a test person has given for a task. For example, if the task accomplishment involves some user interaction, like clicking a mouse-button on an item in the visualization, the callbacks for the mouse could be hooked on the associated TaskPanelStrategy and used to set the answer for this task.

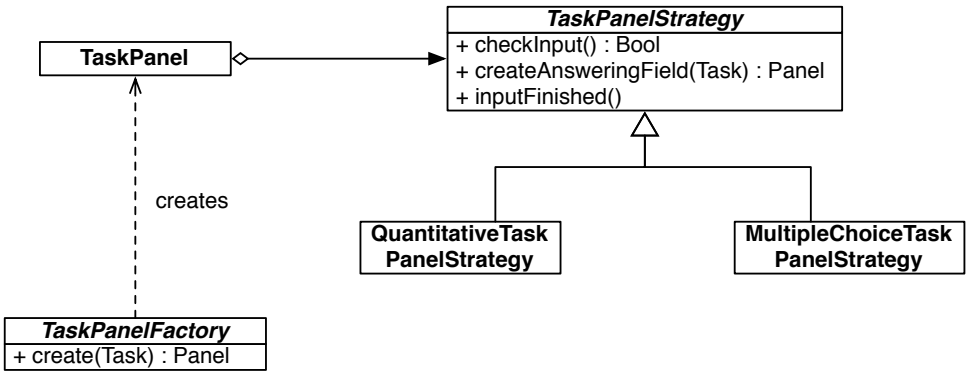


Figure 10.6: TaskPanel `Strategy` Pattern

To customize the standard layout or controls of a TaskPanel, a proprietary TaskPanelFactory could be used to create a special tailored panel or a subclass of TaskPanel instead of the predefined one. Note that the TaskPanelStrategy (or subclass) for each Task can still be used with the proprietary factory. Figure 10.6 illustrates how these components work together.

10.4 Summary

This chapter introduced the individual internal components of the library. To reflect the individual parts of a task based controlled experiment a data model was designed consisting of Tasks, Sessions and SessionGroups.

To elude hard-wired task lists in the program code, a TaskListCreator is available as part of the input-output system that is capable to import a list of Tasks for a Session from an external file. The EvaluationJournal is responsible to save the run-time attributes for the Tasks during a Session to an external file for further analysis and the internal logger is capable to log the user interactions for each Session separately.

The components TaskPanelFactory, TaskPanel and TaskPanelStrategy are responsible to provide an user interface for the currently available Task types. The design patterns used for the user interface generation should facilitate the extension of the TaskPanelFactory for new Task types.

The next chapter describes the overall structure of the library and how the individual components introduced in this chapter interact and work to together.

CHAPTER 11

Overall Library Structure

11.1 Evaluation Manager & Delegate

The central component of EvalBench is the EvaluationManager, which is implemented following the `Singleton` pattern to make it globally available. While it was initially designed rather to be responsible for interaction logging during an evaluation session, it displayed its benefits as the central managing component. It possesses one instance of EvaluationSessionGroup, TaskPanelFactory, TaskListCreator, Logger (see Chapter 10) and an EvaluationDelegate instance each, to provide a central access from anywhere in the program. Figure 11.1 illustrates the overall architecture of the library with all participating objects.

In principle, the EvaluationManger serves as a state machine (cf. Figure 11.2) to process an evaluation session group and changes its state according to events received from a TaskPanel on task completion or the HCI tool that needs to be evaluated by performing a controlled experiment, hereinafter referred to as the *evaluation client*.

The EvaluationManager holds an instance of the EvaluationDelegate, which, as the name already suggests, is used by the EvaluationManger to delegate certain events during an experiment to the client. These events include the start and end events of a task, session or session group to provide an opportunity for the *evaluation client* to react on the events. The interface definition can be seen in Listing 3 and needs to be implemented on the client side. For example, if an experiment is grouped by visualization type, the client has to prepare the required visualization for a certain session in the delegate method `prepareForSession` by looking at the session's configuration for visualization type. Another example for the necessity of the delegate events would be if each task were configured for another dataset, the client has to prepare the visualization for the dataset specified in the task's configuration in the `prepareForTask` delegate method.

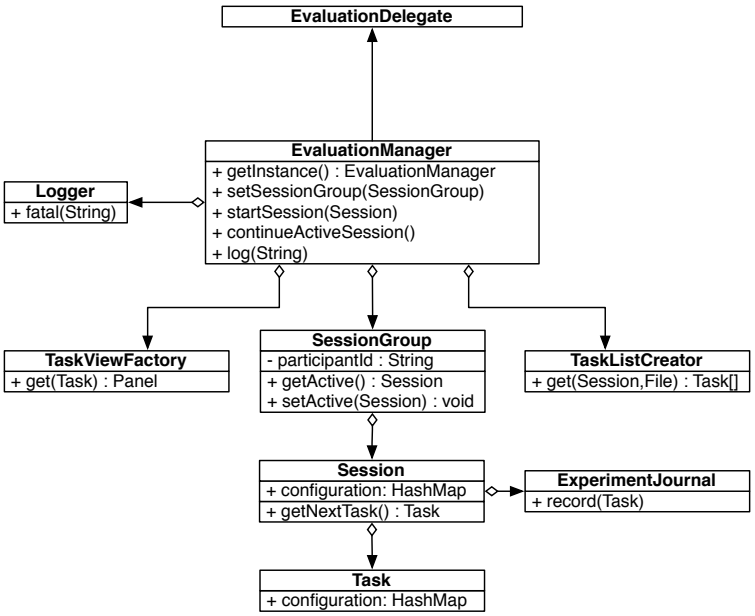


Figure 11.1: Overall architecture of EvalBench

As already mentioned above, the methods of the EvaluationDelegate interface have to be implemented on the client side, since EvalBench is independent from the object of study of the experiment and user interface that needs to be evaluated has to be prepared according to the actual state of the experiment.

The EvaluationManager controls the flow of the current evaluation session group: at the beginning of an experiment, the manager has to be set up with a session group via the method `setSessionGroup` from the client. Subsequently the manager calls the delegate method `prepareForSessionGroup`. The delegate has to decide which session in the actual session group has to be started; this could be managed through input of an experimenter or happen automatically in sequential order. To trigger the start of a session, the client has to call the method `startSession` with the session to be executed and a path to a file containing a list of tasks for the specified session as arguments. The manager utilizes its TaskListCreator instance to load the tasks for the current session and sets the given session to active in the current session group.

Also, the logger instance is configured to save the interaction log during that session to the same directory in the file system as the EvaluationJournal file for the current session. This ensures a separate interaction log file for each session. Eventually, the manager calls the delegate method `prepareForSession`.

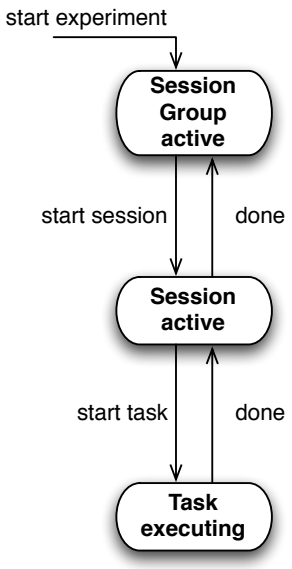


Figure 11.2: States of the EvaluationManager during an experiment

```
interface EvaluationDelegate {
  void prepareForSessionGroup(EvaluationSessionGroup aGroup);
  void sessionGroupDidFinish(EvaluationSessionGroup aGroup);
  void prepareForSession(EvaluationSession aSession);
  void sessionDidFinish(EvaluationSession aSession);
  void prepareForTask(Task aTask);
  void taskWasAnswered(Task aTask);
  void resetGUIForSession(EvaluationSession aSession);
}
```

Listing 3: Evaluation delegate interface that has to to be implemented on the client side

When the client has finished to prepare for the current session, the manager triggers the execution of the currently active session and fetches the next task from the session and tells the delegate to prepare the client for the next task by calling `prepareForTask`. If needed, the client can use the managers TaskPanelFactory to create an appropriate TaskPanel to provide an user interface for the upcoming task. When the `prepareForTask` call returns, the manager starts the execution of the task by setting the start date to the current time. The TaskPanel is responsible to tell the manager that the task was answered (e.g. the test person clicked on the "Next" button) by calling `continueActiveSession`. The manager sets the task end date, and marks this task as finished within the session, thereby causing the EvaluationJournal to record the task. Finally, the manager starts the next task for the active session; if no more tasks are available for the session, the delegate will be informed (`sessionDidFinish`) and the logger instance is configured to save the interactions to the initial directory.

The visualization tool has to notify the EvaluationManager if an interaction should be logged by calling the `log` method. A simple example for the usage of log entries separated for each evaluation session can be seen in Listing 4.

```
void myInteractionListener{
        EvaluationManager.getInstance().log("My interaction to
                be logged");
}
```

Listing 4: Using the EvaluationManager to log interactions for each evaluation session separately.

If the client does not need to use a TaskPanel created by the manager for the task execution (e.g. if the task instruction is to move the mouse cursor over a target field), the client is responsible to call the `continueActiveEvaluationSession` after task completion and has to set the correctness for the task (cf. Section 10.1).

The EvaluationDelegate interface could be extended with several methods; e.g. to gather information from the client. One possible extension would be to make an inquiry about the current visualization state that could be delivered in a key/value format and attached to the EvaluationJournal for every task or session.

11.2 How to use

The first step for an experimenter in the need of an evaluation of her or his visualization or interaction technique is to define distinct sessions, which a test person has to accomplish and group them into session groups. For example, the test persons may have to complete a set of tasks with *visualization technique A* and afterwards with *visualization technique B*. In addition, every test person receives some training for each technique before advancing to the actual experiment. This would result in four sessions: TrainingA, ExperimentA, TrainingB and ExperimentB. These four sessions are then grouped together as a session group, whereas the order of the sessions can be altered for each test person (e.g. if a within-subject design was employed). These sessions need to be configured according to their factors. In this example, every session would get a configuration key *visualization technique* and the value is either *A* or *B*, depending on the required technique. An example for this process is illustrated in Listing 5. Note that the delegate of the EvaluationManager needs to be set before setting the session group in order to react on the newly set SessionGroup.

```
EvaluationManager.getInstance().setDelegate(myEvaluationDelegate);

EvaluationSessionGroup sessionGroup =
        newEvaluationSessionGroup("participant1");

EvaluationSession trainingA = new EvaluationSession("TrainingA");
trainingA.getConfiguration().put("VisualizationType", "A");
EvaluationSession sessionA = new EvaluationSession("EvaluationA");
sessionA.getConfiguration().put("VisualizationType", "A");

EvaluationSession trainingB = new EvaluationSession("TrainingB");
trainingB.getConfiguration().put("VisualizationType", "B");
EvaluationSession sessionB = new EvaluationSession("EvaluationB");
sessionB.getConfiguration().put("VisualizationType", "B");

sessionGroup.addSession(trainingA);
sessionGroup.addSession(sessionA);
sessionGroup.addSession(trainingB);
sessionGroup.addSession(sessionB);

EvaluationManager.getInstance().setSessionGroup(sessionGroup);
```

Listing 5: Sample code illustrating the creation of a session group and setting up the evaluation manager with the group.

Furthermore, a list of tasks for the sessions need to be defined; the experimenter could use the built-in XMLTaskListBuilder (cf. Section 10.2) and define the tasks in an XML format as presented in Listing 1. If additional configurations for the tasks are necessary (e.g. the dataset on which the task is defined), the experimenter could extend Task and provide additional JAXB annotations to the class to enable definition of the dataset in the XML file or by adding the configuration manually in the program code.

The next step would be to implement the EvaluationDelegate interface (cf. Listing 3) to react on the different states of the experiment and prepare the visualizations for the upcoming session groups, sessions and the dataset for the upcoming tasks. The delegate method `prepareForSessionGroup` can be used to decide which session should be started and to trigger the execution. In the example in Listing 6, the delegate takes the first session in the group's session list (TrainingA) and tells the manager to start this session with a task list that is defined in the XML file `TasksTrainingA.xml`.

The delegate method `prepareForSession` in Listing 6 is used to load the needed visualization technique for the upcoming session.

To provide an interface to enable the test persons to answer a task, the EvaluationManager's TaskPanelFactory can be utilized to get a task answering panel that can be included anywhere in the presented user interface of the visualization tool (cf. method `prepareForEvaluationTask`

in Listing 6). Additionally, the visualized dataset is loaded for the current task and a modal dialog of the task description is shown before the task is executed.

```
void prepareForSessionGroup(EvaluationSessionGroup aGroup){
  // choose a session and trigger the execution
  EvaluationManager.getInstance().startEvaluationSession(
    sessionGroup.getSessionList().get(0), "TasksTrainingA.xml");
}

void prepareForSession(Session aSession) {
  // prepare the visualization for the upcoming session
  prepareMyVisualization(aSession.getConfiguration().
    get("VisualizationType");
}
void prepareForEvaluationTask(Task aTask) {
  // show a modal dialog with the task description
  showModalDialog(aTask.getDescription);
  // load the data to be visualized for this task
  loadData(aTask.getConfiguration.get("Dataset"));
  // get task panel and add it to the user interface
  setMyEvaluationPanel( EvaluationManager.getInstance().
    getPanel(aTask) );
}
```

Listing 6: Sample Code of an EvaluationDelegate implementation

CHAPTER 12

Discussion

In this part of the thesis, the design of a class library called EvalBench was presented. It provides an extendable architecture for experimenters to perform task-based controlled experiments to evaluate user performance of e.g. visualization tools or interaction techniques. The development was inspired by multiple software patterns summarized in the book by Gamma et al. [1995] to ensure flexibility and to pave the way for future experimenters to reuse and adopt the library for their special needs, though it remains to be seen if the library will prove its usefulness and flexibility in practice. Although it was designed on the basis of two different user studies, I could not cover every aspect of possible experiments. Therefore, the library currently constitutes a primal structure and will hopefully be adapted and further developed by future experimenters for additional applications in the HCI area. The next practical test of the library will presumably be carried out in near future with a task-based evaluation of the TimeRider visualization technique [Rind et al., 2011].

The *Touchstone* platform by Mackay et al. [2007] has related capabilities, but the major difference between these approaches is that the *run platform* of Touchstone determines the design and structure of the tool to be evaluated. This means that each tool that needs to be evaluated has to be integrated into the Touchstone *run platform* using a proprietary development environment (TDE) to fit into the evaluation system. In my opinion this can potentially create problems for researchers in need of an evaluation study of a distinct tool. Usually, novel visualization techniques are implemented either as proof-of-concept prototypes or as part of existing visualization frameworks without taking a possible evaluation into consideration in the first place. When an evaluation of the technique is imminent, the developer of the technique has to adapt the architecture and structure of the tool or possibly uncase the developed technique from a bigger visualization system. This takes a considerable amount of time and effort and also requires the understanding of the *Touchstone Development Environment*. EvalBench takes a different approach, because the library can be integrated into existing visualization solutions without the need for major changes in the architecture of the tool. Additionally, the set up of an experiment with EvalBench is relatively easy and does not require the developer to engage herself or himself

to much with the working principle of the library, as long as the default implementations of the components are sufficient.

The next sections present the limitations of the library and discuss possible further developments.

12.1 Limitations

The current implementation of the library makes it necessary to set up an experiment by defining the sessions, session groups and the sequence of their execution directly in the source code. This makes it necessary to change and recompile the source code if the experimenter wants to change the experiment design, which seems quite cumbersome. Additionally, this can be a problem if an experimenter who wants to change the experiment design for a visualization technique does not have Java programming abilities.

Another limitation is that this library does not support remote controlling of experiment sessions. It would be desirable to perform a number of evaluation sessions simultaneously with a group of test persons in order to save time. This approach would also ensure equal conditions for the test persons, since the instructions that test persons receive play a crucial role in an experiment and physical and social environmental factors may introduce systematic errors into the observed data [Lazar et al., 2009].

The answering possibilities for the tasks are currently limited to the specification of a single number, selection of one or more answers from a list or selection of a time interval.

The library currently does not support the gathering of data about the test persons. Usually a questionnaire has to be completed before the beginning of the experiment to collect demographic data of the test persons. After the experiment has been completed, feedback and user preference can be recorded for each test person. Currently these data is usually collected by writing them down on a sheet of paper. Since this is cumbersome it could be simplified by means of a computerized method.

12.2 Future Work

To overcome the limitation of this library that make it necessary for an experimenter to have programming experience in order to set up or change the experiment design, the library could be extended to import the experiment design using an external file, for example an experiment design defined in a markup language like XML, which has already been used to model the tasks for a session in an external file. This would make it possible for programming novices to set up and change the experiment without the need to change the source code of the visualization tool. It may also be envisaged that the external file could be generated with the help of a experiment design tool like the *design platform* of *Touchstone* [Mackay et al., 2007].

Another practical enhancement of the library would be to make remote controlling of the sessions available to manage multiple experiment sessions simultaneously. It would be conceivable to incorporate a module into the library that communicates over a network using a to be defined *TCP/IP Application layer protocol* to enable the triggering of experiment sessions. Additionally the journals and interaction logs of experiment sessions could be queried over this protocol to collect and save the recorded data of multiple sessions centrally. Subsequently an application has to be developed that finds the remote controlled applications in the network using the defined protocol and provides the management of those clients and their recorded data. The management application could also configure the remote clients with a certain experiment design (cf. above paragraph).

It is expected that future evaluation experiments will make it necessary to extend the task types in terms of answering possibilities. This is why the data structure of the library was built to facilitate the extension with new task types (cf. Section 10.1). Additionally the user interface for new task types should be easy to create by extending the TaskPanelFactory (cf. Section 10.3).

It might also be practical to extend the library with the possibility to present pre-experiment and post-experiment questionnaires e.g. to collect demographic data about the test persons or feedback about the experiment. One could create a Session with Tasks including questions about the test persons and add the Session to the start or end of a SessionGroup of an experiment. Probably new task types have to be created, e.g. to provide free text answers.

To make it possible for experimenters to perform task-based controlled experiments with visualization or user-interaction tools, the library can be ported to other object-oriented programming languages, since it does not use any Java specific patterns.

CHAPTER 13

Overall Conclusion

Modern data collection technology makes a large number of multivariate data available in various domains. It is important for the analysts to find crucial information in these vast datasets and information visualization on electronic displays is an instrument to support the analysis of deluges of data. However, it is necessary to evaluate if the deployed visualization techniques that are used to display the data are really suited for the analysts' goals. Novel visualization techniques have to present actionable evidence of benefits to encourage the adoption of these techniques.

This work investigated a visualization technique (*SemTimeZoom*) that is capable of displaying the quantitative (numerical values) and qualitative (interpretations) attributes of time-oriented, multivariate data. It uses a combined representation to display both data attributes by employing different visual encodings for the attributes. Spatial position is used to encode the quantitative attributes and color-coding is used to display the qualitative characteristics of the data. To make it possible to explore a large number of variables simultaneously, the visualization technique adapts the visual representation of the data according to the available vertical display space by using different visual information resolutions (*semantic zooming*).

The first research question of this thesis was concerned with related visualization techniques for multivariate time-oriented quantitative data using qualitative abstractions and/or semantic zoom abilities that are described in the scientific literature and how these techniques have been evaluated. The results of the literature research can be found in Chapter 4, which presents three related visualization techniques in detail. The only related visualization technique (*KNAVE-II*) that could be found also using interval-based qualitative abstractions for the visualization of time-oriented clinical data displays the quantitative and qualitative attributes separately and uses spatial position as visual encoding for both attributes. This technique has already been evaluated by means of a controlled experiment that compared KNAVE-II against electronic spreadsheets, which represents the current standard method to display clinical data. The analysis of the results of this experiment revealed significant differences for the dependent variables task completion

time, errors and user preference in favor of KNAVE-II.

To answer the research questions *"Is the SemTimeZoom technique effective for the identification and comparison of qualitative attributes of the data for multiple time-oriented variables?"* and furthermore, *"Is the SemTimeZoom technique well suited to find and compare quantitative values within specified qualitative levels?"*, a comparison study with the visualization technique used in the KNAVE-II application was conducted to examine differences in task completion time, correctness and user preference. The data that were used in the study were blood glucose measurements of a diabetes patient record with the qualitative categories *normal*, *slightly elevated*, *elevated* and *critical* according to the hyperglycemia condition.

The evaluation was planned as a within-subject experiment using two different blocks of tasks. The first block addressed the qualitative attributes of the data and the second block addressed the quantitative attributes of the data within specified qualitative levels. Each block was split up in three lookup and three comparison tasks according to the task taxonomy by Andrienko & Andrienko [2006]. Twenty test persons took part in the study. Their age ranged from 22 to 30 years and most of them were university students, with more than half from the Faculty of Informatics.

The analysis of the task completion times showed that the test persons were generally faster completing the tasks using the SemTimeZoom visualization technique (combined color and position-coded representation), compared to KNAVE-II (separate representations using spatial position to encode the qualitative and quantitative attributes). Particularly for more complex tasks, involving comparison subtasks of quantitative values within specified qualitative levels, the difference of the completion times was found to be statistically significant. The faster completion times did not affect the correctness of the tasks, since the correctness rates were equally high with both techniques. Additionally, significantly more test persons preferred SemTimeZoom rather than KNAVE-II.

In summary, the empirical evaluation described in this work showed that a combined visualization of quantitative and qualitative attributes using different visual encodings for both attributes performs at least equally than comparable visualization techniques and excels especially for more complex tasks. The combined visualization was also preferred over a separate visualization of the data attributes.

The research question *"How can the SemTimeZoom technique be improved to fulfill the intensions of the design?"* could not be answered clearly in this work, since the interaction logs of the controlled experiment revealed that hardly any test person used the semantic zoom feature of the SemTimeZoom technique. Additional follow-up studies are needed to refine the visualization modes and user-interactions used in the SemTimeZoom technique (cf. Section 7.2). However, the overall feedback of the test persons was quite positive regarding the interactions and visualization modes.

One research question that evolved in the process of this work was *"Which evaluation functionality implemented in this and previous available evaluation prototypes can be reused and refined as design patterns for future researchers?"* because it was not possible to find applicable software libraries or frameworks which support the integration of evaluation functionality into the visualization prototypes to be evaluated. To pave the way for future experimenters in the HCI field, a Java software library (*EvalBench*) was developed that focused on flexibility and reusability and should be easy to integrate into existing Java visualization applications. The design of this library and its individual components are presented and discussed in detail in the second part of this thesis along with suggestions on how to refine and extend the existing library. The EvalBench library constitutes the first steps towards an universal evaluation infrastructure to facilitate evaluation studies.

Bibliography

Aigner, W., Kainz, C., Ma, R., & Miksch, S. (2011). Bertin was right: An empirical evaluation of indexing to compare multivariate time-series data using line plots. *Computer Graphics Forum*, *30*(1), 215–228.

Andrienko, N., & Andrienko, G. (2006). *Exploratory analysis of spatial and temporal data: A systematic approach*. Secaucus, NJ, USA: Springer.

Appert, C. (n.d.). *TouchStone: Exploratory Design of Experiments.* Retrieved 2011-06-10, from `http://www.lri.fr/~appert/website/touchstone/touchstone.html`

Bade, R., Schlechtweg, S., & Miksch, S. (2004). Connecting time-oriented data and information to a coherent interactive visualization. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 105–112). New York, NY, USA: ACM.

Bederson, B. B., & Hollan, J. D. (1994). Pad++: a zooming graphical interface for exploring alternate interface physics. In *Proceedings of the 7th annual ACM symposium on user interface software and technology* (pp. 17–26). New York, NY, USA: ACM.

Biffl, S., Thurnher, B., Goluch, G., Winkler, D., Aigner, W., & Miksch, S. (2005). An empirical investigation on the visualization of temporal uncertainties in software engineering project planning. In J. Verner & G. Travassos (Eds.), *Proceedings of empirical software engineering 2005 (ISESE'05)* (pp. 437–446). Los Alamitos, CA, USA: IEEE Society Press.

Boaz, D., & Shahar, Y. (2003). Idan: A distributed temporal-abstraction mediator for medical databases. In *Proceedings of the 9th conference on artificial intelligence in medicine* (pp. 21–30). Protaras, Cyprus, Europe: Springer Verlag.

Brewer, C., & Harrower, M. (2003). *Colorbrewer: Color Advice for Maps.* Retrieved 2011-09-14, from `http://colorbrewer2.org/`

Card, S. K., Mackinlay, J. D., & Shneiderman, B. (Eds.). (1999). *Readings in information visualization: using vision to think*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Catley, C., Stratti, H., & McGregor, C. (2008). Multi-dimensional temporal abstraction and data mining of medical time series data: trends and challenges. In *Conference proceedings of the international conference of IEEE engineering in medicine and biology society* (pp. 4322–4325). Vancouver, BC: IEEE.

Cleveland, W. S. (1993). A model for studying display methods of statistical graphs. *Journal of Computational and Graphical Statistics*, *2*(4), 323–343.

Cleveland, W. S., McGill, M. E., & McGill, R. (1988). The shape parameter of a two-variable graph. *Journal of the American Statistical Association*, *83*(402), 289–300.

Farrington, J. (2011). Seven plus or minus two. *Performance Improvement Quarterly*, *23*(4), 113–116.

Gamma, E., Helm, R., Johnson, R. E., & Vlissides, J. (1995). *Design patterns: Elements of reusable object-oriented software*. Reading, MA, USA: Addison-Wesley.

Gershon, N., Eick, S. G., & Card, S. (1998, March). Information visualization. *interactions*, *5*, 9–15.

Halford, G. S., Baker, R., McCredden, J. E., & Bain, J. D. (2005, January 1). How Many Variables Can Humans Process? *Psychological Science*, *16*(1), 70–76.

Harrower, M., & Brewer, C. (2003). Colorbrewer.org: An online tool for selecting colour schemes for maps. *Cartographic Journal, The*, *40*(1), 27–37.

Healey, C. G. (1996). Choosing effective colours for data visualization. In *Proceedings of the 7th conference on visualization '96* (pp. 263–270). Los Alamitos, CA, USA: IEEE Computer Society Press.

Heer, J., & Agrawala, M. (2006, September). Multi-scale banking to 45 degrees. *IEEE Transactions on Visualization and Computer Graphics*, *12*, 701–708.

Heer, J., Card, S. K., & Landay, J. A. (2005). prefuse: a toolkit for interactive information visualization. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 421–430). New York, NY, USA: ACM.

Heer, J., & Robertson, G. (2007, November). Animated transitions in statistical data graphics. *IEEE Transactions on Visualization and Computer Graphics*, *13*, 1240–1247.

Hoffmann, S. (2010). *Semantic Zoom of Time-oriented Quantitative Data.* Retrieved 2011-09-8, from `http://ieg.ifs.tuwien.ac.at/projects/semzoom/index.html`

Javed, W., McDonnel, B., & Elmqvist, N. (2010, November). Graphical perception of multiple time series. *IEEE Transactions on Visualization and Computer Graphics*, *16*, 927–934.

Klimov, D., Shahar, Y., & Taieb-Maimon, M. (2010, May). Intelligent visualization and exploration of time-oriented data of multiple patients. *Artif. Intell. Med.*, *49*, 11–31.

Lam, H., Bertini, E., Isenberg, P., Plaisant, C., & Carpendale, S. (2011). *Seven guiding scenarios for information visualization evaluation* (Tech. Rep. No. 2011-992-04). Calgary, Alberta, Canada: University of Calgary, Department of Computer Science.

Lam, H., Munzner, T., & Kincaid, R. (2007, November). Overview use in multiple visual information resolution interfaces. *IEEE Transactions on Visualization and Computer Graphics*, *13*, 1278–1285.

Lazar, J., Feng, J., & Hochheiser, H. (2009). *Research methods in human-computer interaction*. Indianapolis, IN, USA: Wiley.

Mackay, W., Appert, C., Beaudouin-Lafon, M., Chapuis, O., Du, Y., Fekete, J.-D., et al. (2007). Touchstone: Exploratory design of experiments. In *Proceedings of the SIGCHI conerence on human factors in computing systems* (pp. 1425–1434). New York, NY, USA: ACM.

Mackinlay, J. (1986, April). Automating the design of graphical presentations of relational information. *ACM Trans. Graph.*, *5*, 110–141.

Maor, E. (2002). *Trigonometric delights*. Princeton University Press.

Martins, S. B., Shahar, Y., Goren-Bar, D., Galperin, M., Kaizer, H., Basso, L. V., et al. (2008, May). Evaluation of an architecture for intelligent query and exploration of time-oriented clinical data. *Artif. Intell. Med.*, *43*, 17–34.

McLachlan, P., Munzner, T., Koutsofios, E., & North, S. (2008). LiveRAC: interactive visual exploration of system management time-series data. In *Proceeding of the twenty-sixth annual SIGCHI conference on human factors in computing systems* (pp. 1483–1492). New York, NY, USA: ACM.

McLachlan, P. J. (2006). *LiveRAC : live reorderable accordion drawing*. Unpublished master's thesis, University Of British Columbia.

Miksch, S. (2004, June). *Project Midgaard.* Retrieved 2011-09-14, from `http://ieg.ifs.tuwien.ac.at/projects/midgaard.html`

Miksch, S., Horn, W., Popow, C., & Paky, F. (1996). Context-sensitive and expectation-guided temporal abstraction of high-frequency data. In Y. Iwasaki & A. Farquhar (Eds.), *Proceedings of the tenth international workshop for qualitative reasoning (QR-96)* (pp. 154–163). Menlo Park, CA, USA: AAAI Press.

Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook* (Vol. 2nd). Thousand Oaks, CA, USA: Sage.

Munzner, T., Guimbretière, F., Tasiran, S., Zhang, L., & Zhou, Y. (2003, July). Treejuxtaposer: scalable tree comparison using focus+context with guaranteed visibility. *ACM Trans. Graph.*, *22*, 453–462.

Nekrasovski, D., Bodnar, A., McGrenere, J., Guimbretière, F., & Munzner, T. (2006). An evaluation of pan & zoom and rubber sheet navigation with and without an overview. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 11–20). New York, NY, USA: ACM.

Ordóñez, P., desJardins, M., Lombardi, M., Lehmann, C. U., & Fackler, J. (2010). An animated multivariate visualization for physiological and clinical data in the icu. In *Proceedings of the 1st ACM international health informatics symposium* (pp. 771–779). New York, NY, USA: ACM.

Plaisant, C. (2004). The challenge of information visualization evaluation. In *Proceedings of the working conference on advanced visual interfaces* (pp. 109–116). New York, NY, USA: ACM.

Plaisant, C., Milash, B., Rose, A., Widoff, S., & Shneiderman, B. (1996). Lifelines: visualizing personal histories. In *Proceedings of the SIGCHI conference on human factors in computing systems: common ground* (pp. 221–227). New York, NY, USA: ACM.

Playfair, W. (1786). *The commercial and political atlas: Representing, by means of stained copper-plate charts, the progress of the commerce, revenues, expenditure and debts of england during the whole of the eighteenth century*.

Powsner, S., & Tufte, E. (1994). Graphical summary of patient status. *The Lancet*, *344*(8919), 386–389. (Originally published as Volume 2, Issue 8919)

Rind, A., Aigner, W., Miksch, S., Wiltner, S., Pohl, M., Drexler, F., et al. (2011). Visually exploring multivariate trends in patient cohorts using animated scatter plots. In *Proceedings of the 2011th international conference on ergonomics and health aspects of work with computers* (pp. 139–148). Berlin, Heidelberg, Germany: Springer-Verlag.

Rind, A., Aigner, W., Miksch, S., Wiltner, S., Pohl, M., Turic, T., et al. (2011). Visual exploration of time-oriented patient data for chronic diseases: Design study and evaluation. In A. Holzinger & K.-M. Simonic (Eds.), *Proceedings of USAB 2011: Information Quality in e-Health.* Heidelberg: Springer. (forthcoming)

Sarkar, M., Snibbe, S. S., Tversky, O. J., & Reiss, S. P. (1993). Stretching the rubber sheet: A metaphor for viewing large layouts on small screens. In *In UIST: Proceedings of the ACM symposium on user interface software and technology* (pp. 81–91). Providence, RI, USA: ACM Press.

Sauro, J., & Lewis, J. R. (2010). Average task times in usability tests: what to report? In *Proceedings of the 28th international conference on human factors in computing systems* (pp. 2347–2350). New York, NY, USA: ACM.

Seyfang, A., Miksch, S., Horn, W., Urschitz, M. S., Popow, C., & Poets, C. F. (2001). Using time-oriented data abstraction methods to optimize oxygen supply for neonates. In *Proceedings of the 8th conference on AI in medicine in europe: Artificial Intelligence Medicine* (pp. 217–226). London, UK, UK: Springer-Verlag.

Shahar, Y., Goren-Bar, D., Boaz, D., & Tahan, G. (2006, October). Distributed, intelligent, interactive visualization and exploration of time-oriented clinical data and their abstractions. *Artif. Intell. Med.*, *38*, 115–135.

Shahar, Y., & Musen, M. A. (1993). Resume: a temporal-abstraction system for patient monitoring. *Computers and Biomedical Research*, *26*(3), 25–273.

Stacey, M., & McGregor, C. (2007, January). Temporal abstraction in intelligent clinical data analysis: A survey. *Artif. Intell. Med.*, *39*, 1–24.

Stevens, S. S. (1946, June). On the theory of scales of measurement. *Science*, *103*(2684), 677-680.

Tufte, E. R. (2006). *Beautiful evidence*. Cheshire, CT, USA: Graphics Press.

Ware, C. (2004). *Information visualization - perception for design*. San Francisco, CA, USA: Morgan Kaufmann.

Wickens, C. D., & Carswell, C. M. (1995). The proximity compatibility principle: Its psychological foundation and relevance to display design. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *37*(3), 473–494.

List of Figures

List of Tables

List of listings

APPENDIX A

Detailed Information about the Test Persons

Test person	*Age*	*Gender*	*Job*	*Education*
1	29	m	Student (Medical Informatics)	Bachelor
2	24	f	Hospital Nurse	Matura
3	29	m	Student (Medical Informatics)	Bachelor
4	30	m	IT Professional	Master
5	30	m	Student (Environmental Engineering)	Bachelor
6	30	f	Constuctional Engineer	Master
7	29	m	Civil Engineer	Master
8	22	m	Student (Medical Informatics)	Matura
9	23	m	Student (Medical Informatics)	Matura
10	27	f	Student (Media Informatics)	Bachelor
11	28	f	Student (Media Informatics)	Bachelor
12	29	f	Student (Psychology)	Matura
13	29	f	Clinical Study Co-ordinator	Master
14	24	m	Student (Architecture)	Matura
15	24	f	Student (Molecular biology)	Bachelor
16	23	m	Student (Media Informatics)	Matura
17	28	m	Student (Medical Informatics)	Bachelor
18	27	m	Student (Mathematics in Computer Science)	Bachelor
19	24	f	Student (Architecture)	Matura
20	29	m	Student (Medical Informatics)	Bachelor

Table A.2: Personal information about the test persons

Test person	*Linecharts*	*Barcharts*	*Data analysis*	*Computer experience*
1	good	good	good	very good
2	average	average	good	average
3	good	good	average	average
4	average	average	average	very good
5	very good	very good	very good	very good
6	good	good	good	good
7	good	good	very good	good
8	average	good	average	good
9	good	good	good	good
10	good	good	good	very good
11	average	average	average	very good
12	good	good	good	good
13	average	average	very good	good
14	good	good	average	good
15	good	good	very good	good
16	average	average	average	very good
17	good	average	good	very good
18	very good	very good	very good	very good
19	average	average	average	good
20	good	good	good	very good

Table A.3: Self-assessments of the test persons

The questionnaire that was used to collect these data can be found in Appendix C.

APPENDIX B

Data collected during the Experiment

Test person	*Task number*	*Duration[s]*	*Correctness*	*Input type*	*Vis. type*	*Dataset*	*Round*
1	6	16.834	1	ChoiceSelectionTask	STZ	1	1
1	12	18.109	0	QuantitativeTask	STZ	1	1
1	2	23.152	1	IntervalSelectionTask	STZ	1	1
1	9	9.307	1	QuantitativeTask	STZ	1	1
1	10	12.865	1	IntervalSelectionTask	STZ	1	1
1	4	7.983	1	ChoiceSelectionTask	STZ	1	1
1	3	10.435	1	IntervalSelectionTask	STZ	1	1
1	11	62.397	0	ChoiceSelectionTask	STZ	1	1
1	9	15.236	1	QuantitativeTask	STZ	1	1
1	12	12.691	1	QuantitativeTask	STZ	1	1
1	5	7.638	1	ChoiceSelectionTask	STZ	1	1
1	5	12.646	1	ChoiceSelectionTask	STZ	1	1
1	7	31.504	1	ChoiceSelectionTask	STZ	1	1
1	8	27.157	1	QuantitativeTask	STZ	1	1
1	11	30.977	1	ChoiceSelectionTask	STZ	1	1
1	4	7.825	1	ChoiceSelectionTask	STZ	1	1
1	1	7.431	1	QuantitativeTask	STZ	1	1
1	7	10.63	1	ChoiceSelectionTask	STZ	1	1
1	6	12.084	1	ChoiceSelectionTask	STZ	1	1
1	1	7.447	1	QuantitativeTask	STZ	1	1
1	2	14.595	1	IntervalSelectionTask	STZ	1	1
1	10	53.933	1	IntervalSelectionTask	STZ	1	1
1	8	20.779	1	QuantitativeTask	STZ	1	1
1	3	11.215	1	IntervalSelectionTask	STZ	1	1
1	6	37.94	1	ChoiceSelectionTask	KNAVE	2	2

1	5	10.531	1	ChoiceSelectionTask	KNAVE	2	2
1	4	12.654	1	ChoiceSelectionTask	KNAVE	2	2
1	7	16.775	1	ChoiceSelectionTask	KNAVE	2	2
1	8	35.762	1	QuantitativeTask	KNAVE	2	2
1	3	18.946	1	IntervalSelectionTask	KNAVE	2	2
1	9	17.589	1	QuantitativeTask	KNAVE	2	2
1	1	6.176	1	QuantitativeTask	KNAVE	2	2
1	10	19.913	1	IntervalSelectionTask	KNAVE	2	2
1	11	15.065	1	ChoiceSelectionTask	KNAVE	2	2
1	7	11.878	1	ChoiceSelectionTask	KNAVE	2	2
1	6	12.719	1	ChoiceSelectionTask	KNAVE	2	2
1	3	15.949	1	IntervalSelectionTask	KNAVE	2	2
1	9	11.884	1	QuantitativeTask	KNAVE	2	2
1	2	15.094	1	IntervalSelectionTask	KNAVE	2	2
1	5	7.823	1	ChoiceSelectionTask	KNAVE	2	2
1	11	38.576	0	ChoiceSelectionTask	KNAVE	2	2
1	8	19.337	1	QuantitativeTask	KNAVE	2	2
1	2	14.371	1	IntervalSelectionTask	KNAVE	2	2
1	1	8.654	1	QuantitativeTask	KNAVE	2	2
1	12	10.106	1	QuantitativeTask	KNAVE	2	2
1	10	25.994	1	IntervalSelectionTask	KNAVE	2	2
1	12	10.779	1	QuantitativeTask	KNAVE	2	2
1	4	6.145	1	ChoiceSelectionTask	KNAVE	2	2
2	3	35.798	1	IntervalSelectionTask	KNAVE	1	1
2	2	67.544	0	IntervalSelectionTask	KNAVE	1	1
2	11	60.942	1	ChoiceSelectionTask	KNAVE	1	1
2	2	27.944	1	IntervalSelectionTask	KNAVE	1	1
2	9	26.95	1	QuantitativeTask	KNAVE	1	1
2	3	24.142	1	IntervalSelectionTask	KNAVE	1	1
2	7	41.044	1	ChoiceSelectionTask	KNAVE	1	1
2	1	16.953	1	QuantitativeTask	KNAVE	1	1
2	9	18.662	1	QuantitativeTask	KNAVE	1	1
2	10	25.678	1	IntervalSelectionTask	KNAVE	1	1
2	6	35.003	1	ChoiceSelectionTask	KNAVE	1	1
2	6	20.002	1	ChoiceSelectionTask	KNAVE	1	1
2	8	36.448	1	QuantitativeTask	KNAVE	1	1
2	11	50.211	1	ChoiceSelectionTask	KNAVE	1	1
2	5	38.258	1	ChoiceSelectionTask	KNAVE	1	1
2	7	50.784	1	ChoiceSelectionTask	KNAVE	1	1
2	4	44.259	1	ChoiceSelectionTask	KNAVE	1	1
2	12	30.241	1	QuantitativeTask	KNAVE	1	1
2	4	12.074	1	ChoiceSelectionTask	KNAVE	1	1
2	5	8.043	1	ChoiceSelectionTask	KNAVE	1	1

2	12	19.172	1	QuantitativeTask	KNAVE	1	1
2	10	37.243	1	IntervalSelectionTask	KNAVE	1	1
2	8	26.759	1	QuantitativeTask	KNAVE	1	1
2	1	8.534	1	QuantitativeTask	KNAVE	1	1
2	10	19.774	1	IntervalSelectionTask	STZ	2	2
2	10	14.718	1	IntervalSelectionTask	STZ	2	2
2	1	12.027	1	QuantitativeTask	STZ	2	2
2	4	58.631	1	ChoiceSelectionTask	STZ	2	2
2	11	14.734	1	ChoiceSelectionTask	STZ	2	2
2	5	16.641	1	ChoiceSelectionTask	STZ	2	2
2	4	10.401	1	ChoiceSelectionTask	STZ	2	2
2	6	7.93	1	ChoiceSelectionTask	STZ	2	2
2	12	9.592	1	QuantitativeTask	STZ	2	2
2	2	16.853	1	IntervalSelectionTask	STZ	2	2
2	8	19.759	1	QuantitativeTask	STZ	2	2
2	2	20.701	1	IntervalSelectionTask	STZ	2	2
2	6	13.903	1	ChoiceSelectionTask	STZ	2	2
2	9	9.522	1	QuantitativeTask	STZ	2	2
2	7	20.401	1	ChoiceSelectionTask	STZ	2	2
2	1	6.205	1	QuantitativeTask	STZ	2	2
2	8	15.776	1	QuantitativeTask	STZ	2	2
2	5	7.692	1	ChoiceSelectionTask	STZ	2	2
2	11	26.405	1	ChoiceSelectionTask	STZ	2	2
2	9	9.217	1	QuantitativeTask	STZ	2	2
2	12	11.967	1	QuantitativeTask	STZ	2	2
2	7	14.152	1	ChoiceSelectionTask	STZ	2	2
2	3	14.371	1	IntervalSelectionTask	STZ	2	2
2	3	10.324	1	IntervalSelectionTask	STZ	2	2
3	8	21.649	1	QuantitativeTask	STZ	2	1
3	10	27.291	1	IntervalSelectionTask	STZ	2	1
3	5	21.336	1	ChoiceSelectionTask	STZ	2	1
3	5	13.389	1	ChoiceSelectionTask	STZ	2	1
3	6	12.507	1	ChoiceSelectionTask	STZ	2	1
3	4	32.699	1	ChoiceSelectionTask	STZ	2	1
3	2	35.641	1	IntervalSelectionTask	STZ	2	1
3	3	21.1	1	IntervalSelectionTask	STZ	2	1
3	6	13.97	1	ChoiceSelectionTask	STZ	2	1
3	10	23.598	1	IntervalSelectionTask	STZ	2	1
3	4	28.743	1	ChoiceSelectionTask	STZ	2	1
3	1	11.845	1	QuantitativeTask	STZ	2	1
3	7	35.641	1	ChoiceSelectionTask	STZ	2	1
3	11	72.096	1	ChoiceSelectionTask	STZ	2	1
3	12	17.81	1	QuantitativeTask	STZ	2	1

3	1	9.697	1	QuantitativeTask	STZ	2	1
3	3	48.778	1	IntervalSelectionTask	STZ	2	1
3	8	26.931	1	QuantitativeTask	STZ	2	1
3	11	34.473	1	ChoiceSelectionTask	STZ	2	1
3	12	18.809	1	QuantitativeTask	STZ	2	1
3	9	17.669	1	QuantitativeTask	STZ	2	1
3	9	12.628	1	QuantitativeTask	STZ	2	1
3	7	45.252	1	ChoiceSelectionTask	STZ	2	1
3	2	31.445	1	IntervalSelectionTask	STZ	2	1
3	4	31.462	1	ChoiceSelectionTask	KNAVE	1	2
3	11	35.333	1	ChoiceSelectionTask	KNAVE	1	2
3	6	24.674	1	ChoiceSelectionTask	KNAVE	1	2
3	9	10.528	1	QuantitativeTask	KNAVE	1	2
3	7	16.518	1	ChoiceSelectionTask	KNAVE	1	2
3	6	28.579	0	ChoiceSelectionTask	KNAVE	1	2
3	11	59.849	1	ChoiceSelectionTask	KNAVE	1	2
3	12	20.004	1	QuantitativeTask	KNAVE	1	2
3	12	23.008	1	QuantitativeTask	KNAVE	1	2
3	3	30.668	1	IntervalSelectionTask	KNAVE	1	2
3	8	21.284	1	QuantitativeTask	KNAVE	1	2
3	5	23.817	1	ChoiceSelectionTask	KNAVE	1	2
3	3	32.374	1	IntervalSelectionTask	KNAVE	1	2
3	2	44.735	1	IntervalSelectionTask	KNAVE	1	2
3	7	23.103	1	ChoiceSelectionTask	KNAVE	1	2
3	8	26.1	1	QuantitativeTask	KNAVE	1	2
3	9	19.272	1	QuantitativeTask	KNAVE	1	2
3	5	23.442	1	ChoiceSelectionTask	KNAVE	1	2
3	10	25.196	1	IntervalSelectionTask	KNAVE	1	2
3	10	32.351	1	IntervalSelectionTask	KNAVE	1	2
3	4	16.026	1	ChoiceSelectionTask	KNAVE	1	2
3	2	27.678	1	IntervalSelectionTask	KNAVE	1	2
3	1	16.862	1	QuantitativeTask	KNAVE	1	2
3	1	12.556	1	QuantitativeTask	KNAVE	1	2
4	12	45.983	1	QuantitativeTask	KNAVE	2	1
4	3	34.832	1	IntervalSelectionTask	KNAVE	2	1
4	9	14.952	1	QuantitativeTask	KNAVE	2	1
4	10	36.04	1	IntervalSelectionTask	KNAVE	2	1
4	6	53.631	1	ChoiceSelectionTask	KNAVE	2	1
4	11	41.568	1	ChoiceSelectionTask	KNAVE	2	1
4	2	68.157	1	IntervalSelectionTask	KNAVE	2	1
4	11	85.948	1	ChoiceSelectionTask	KNAVE	2	1
4	6	37.87	1	ChoiceSelectionTask	KNAVE	2	1
4	9	13.302	1	QuantitativeTask	KNAVE	2	1

4	4	26.732	1	ChoiceSelectionTask	KNAVE	2	1
4	3	29.053	1	IntervalSelectionTask	KNAVE	2	1
4	12	16.882	1	QuantitativeTask	KNAVE	2	1
4	8	25.714	1	QuantitativeTask	KNAVE	2	1
4	5	32.836	1	ChoiceSelectionTask	KNAVE	2	1
4	2	31.098	1	IntervalSelectionTask	KNAVE	2	1
4	4	16.053	1	ChoiceSelectionTask	KNAVE	2	1
4	5	23.599	1	ChoiceSelectionTask	KNAVE	2	1
4	1	7.012	1	QuantitativeTask	KNAVE	2	1
4	7	39.779	1	ChoiceSelectionTask	KNAVE	2	1
4	1	8.968	1	QuantitativeTask	KNAVE	2	1
4	8	21.377	1	QuantitativeTask	KNAVE	2	1
4	7	24.608	1	ChoiceSelectionTask	KNAVE	2	1
4	10	17.306	1	IntervalSelectionTask	KNAVE	2	1
4	11	36.404	1	ChoiceSelectionTask	STZ	1	2
4	9	17.838	1	QuantitativeTask	STZ	1	2
4	10	17.079	1	IntervalSelectionTask	STZ	1	2
4	4	21.393	1	ChoiceSelectionTask	STZ	1	2
4	6	13.841	1	ChoiceSelectionTask	STZ	1	2
4	7	18.697	1	ChoiceSelectionTask	STZ	1	2
4	5	22.087	1	ChoiceSelectionTask	STZ	1	2
4	3	12.627	1	IntervalSelectionTask	STZ	1	2
4	11	37.207	1	ChoiceSelectionTask	STZ	1	2
4	7	20.386	1	ChoiceSelectionTask	STZ	1	2
4	12	9.865	1	QuantitativeTask	STZ	1	2
4	9	7.056	1	QuantitativeTask	STZ	1	2
4	10	24.126	1	IntervalSelectionTask	STZ	1	2
4	8	22.728	1	QuantitativeTask	STZ	1	2
4	1	8.752	1	QuantitativeTask	STZ	1	2
4	1	16.169	1	QuantitativeTask	STZ	1	2
4	6	16.785	1	ChoiceSelectionTask	STZ	1	2
4	8	21.011	1	QuantitativeTask	STZ	1	2
4	5	17.894	1	ChoiceSelectionTask	STZ	1	2
4	3	9.054	1	IntervalSelectionTask	STZ	1	2
4	2	19.349	1	IntervalSelectionTask	STZ	1	2
4	2	18.648	1	IntervalSelectionTask	STZ	1	2
4	4	11.977	1	ChoiceSelectionTask	STZ	1	2
4	12	25.488	1	QuantitativeTask	STZ	1	2
5	5	21.665	1	ChoiceSelectionTask	STZ	1	1
5	2	16.94	1	IntervalSelectionTask	STZ	1	1
5	8	27.592	1	QuantitativeTask	STZ	1	1
5	10	17.691	1	IntervalSelectionTask	STZ	1	1
5	4	44.037	1	ChoiceSelectionTask	STZ	1	1

5	5	17.138	1	ChoiceSelectionTask	STZ	1	1
5	3	26.648	1	IntervalSelectionTask	STZ	1	1
5	7	9.684	1	ChoiceSelectionTask	STZ	1	1
5	1	6.183	1	QuantitativeTask	STZ	1	1
5	6	7.125	1	ChoiceSelectionTask	STZ	1	1
5	4	18.327	1	ChoiceSelectionTask	STZ	1	1
5	11	27.452	1	ChoiceSelectionTask	STZ	1	1
5	3	10.261	1	IntervalSelectionTask	STZ	1	1
5	12	22.874	1	QuantitativeTask	STZ	1	1
5	12	13.269	1	QuantitativeTask	STZ	1	1
5	9	23.741	1	QuantitativeTask	STZ	1	1
5	1	11.317	1	QuantitativeTask	STZ	1	1
5	10	20.813	1	IntervalSelectionTask	STZ	1	1
5	7	9.622	1	ChoiceSelectionTask	STZ	1	1
5	11	46.25	1	ChoiceSelectionTask	STZ	1	1
5	8	16.355	1	QuantitativeTask	STZ	1	1
5	2	32.229	1	IntervalSelectionTask	STZ	1	1
5	9	8.955	1	QuantitativeTask	STZ	1	1
5	6	20.344	1	ChoiceSelectionTask	STZ	1	1
5	10	38.414	1	IntervalSelectionTask	KNAVE	2	2
5	4	14.611	1	ChoiceSelectionTask	KNAVE	2	2
5	1	5.32	1	QuantitativeTask	KNAVE	2	2
5	8	24.232	0	QuantitativeTask	KNAVE	2	2
5	7	16.695	1	ChoiceSelectionTask	KNAVE	2	2
5	3	15.038	1	IntervalSelectionTask	KNAVE	2	2
5	6	18.486	1	ChoiceSelectionTask	KNAVE	2	2
5	1	8.328	1	QuantitativeTask	KNAVE	2	2
5	6	15.576	1	ChoiceSelectionTask	KNAVE	2	2
5	8	17.64	1	QuantitativeTask	KNAVE	2	2
5	11	28.712	1	ChoiceSelectionTask	KNAVE	2	2
5	5	17.961	1	ChoiceSelectionTask	KNAVE	2	2
5	5	11.306	1	ChoiceSelectionTask	KNAVE	2	2
5	2	23.842	1	IntervalSelectionTask	KNAVE	2	2
5	11	44.06	1	ChoiceSelectionTask	KNAVE	2	2
5	4	5.701	1	ChoiceSelectionTask	KNAVE	2	2
5	3	12.225	1	IntervalSelectionTask	KNAVE	2	2
5	10	12.479	1	IntervalSelectionTask	KNAVE	2	2
5	9	19.692	1	QuantitativeTask	KNAVE	2	2
5	7	23.3	1	ChoiceSelectionTask	KNAVE	2	2
5	12	18.166	1	QuantitativeTask	KNAVE	2	2
5	9	10.33	1	QuantitativeTask	KNAVE	2	2
5	12	9.462	1	QuantitativeTask	KNAVE	2	2
5	2	23.446	1	IntervalSelectionTask	KNAVE	2	2

6	11	36.224	1	ChoiceSelectionTask	KNAVE	1	1
6	3	21.742	1	IntervalSelectionTask	KNAVE	1	1
6	10	15.714	1	IntervalSelectionTask	KNAVE	1	1
6	8	25.041	1	QuantitativeTask	KNAVE	1	1
6	2	24.67	1	IntervalSelectionTask	KNAVE	1	1
6	1	7.839	1	QuantitativeTask	KNAVE	1	1
6	3	12.512	1	IntervalSelectionTask	KNAVE	1	1
6	7	50.171	1	ChoiceSelectionTask	KNAVE	1	1
6	8	14.551	1	QuantitativeTask	KNAVE	1	1
6	5	28.225	1	ChoiceSelectionTask	KNAVE	1	1
6	12	30.959	1	QuantitativeTask	KNAVE	1	1
6	9	9.488	1	QuantitativeTask	KNAVE	1	1
6	1	9.017	1	QuantitativeTask	KNAVE	1	1
6	6	19.825	1	ChoiceSelectionTask	KNAVE	1	1
6	5	10.439	1	ChoiceSelectionTask	KNAVE	1	1
6	10	26.343	1	IntervalSelectionTask	KNAVE	1	1
6	11	16.494	0	ChoiceSelectionTask	KNAVE	1	1
6	2	14.312	1	IntervalSelectionTask	KNAVE	1	1
6	4	13.429	1	ChoiceSelectionTask	KNAVE	1	1
6	7	12.217	1	ChoiceSelectionTask	KNAVE	1	1
6	9	8.926	1	QuantitativeTask	KNAVE	1	1
6	12	15.671	1	QuantitativeTask	KNAVE	1	1
6	6	17.662	1	ChoiceSelectionTask	KNAVE	1	1
6	4	11.587	1	ChoiceSelectionTask	KNAVE	1	1
6	2	17.16	1	IntervalSelectionTask	STZ	2	2
6	12	12.103	1	QuantitativeTask	STZ	2	2
6	9	6.257	1	QuantitativeTask	STZ	2	2
6	10	11.993	1	IntervalSelectionTask	STZ	2	2
6	7	13.019	1	ChoiceSelectionTask	STZ	2	2
6	6	5.7	1	ChoiceSelectionTask	STZ	2	2
6	8	27.872	1	QuantitativeTask	STZ	2	2
6	9	12.514	1	QuantitativeTask	STZ	2	2
6	11	18.752	1	ChoiceSelectionTask	STZ	2	2
6	2	20.77	1	IntervalSelectionTask	STZ	2	2
6	11	14.112	1	ChoiceSelectionTask	STZ	2	2
6	1	6.637	1	QuantitativeTask	STZ	2	2
6	5	5.175	1	ChoiceSelectionTask	STZ	2	2
6	5	6.534	1	ChoiceSelectionTask	STZ	2	2
6	8	11.899	1	QuantitativeTask	STZ	2	2
6	12	8.704	1	QuantitativeTask	STZ	2	2
6	4	29.125	1	ChoiceSelectionTask	STZ	2	2
6	3	11.509	1	IntervalSelectionTask	STZ	2	2
6	7	8.913	1	ChoiceSelectionTask	STZ	2	2

6	1	6.447	1	QuantitativeTask	STZ	2	2
6	10	8.483	1	IntervalSelectionTask	STZ	2	2
6	6	4.708	1	ChoiceSelectionTask	STZ	2	2
6	3	8.224	1	IntervalSelectionTask	STZ	2	2
6	4	10.337	1	ChoiceSelectionTask	STZ	2	2
7	6	6.43	1	ChoiceSelectionTask	STZ	2	1
7	10	22.544	0	IntervalSelectionTask	STZ	2	1
7	3	37.153	1	IntervalSelectionTask	STZ	2	1
7	4	15.937	0	ChoiceSelectionTask	STZ	2	1
7	7	12.746	1	ChoiceSelectionTask	STZ	2	1
7	8	21.099	1	QuantitativeTask	STZ	2	1
7	2	20.456	1	IntervalSelectionTask	STZ	2	1
7	12	9.362	1	QuantitativeTask	STZ	2	1
7	9	14.685	1	QuantitativeTask	STZ	2	1
7	8	19.47	1	QuantitativeTask	STZ	2	1
7	9	7.031	1	QuantitativeTask	STZ	2	1
7	7	20.508	1	ChoiceSelectionTask	STZ	2	1
7	3	14.323	1	IntervalSelectionTask	STZ	2	1
7	5	9.521	1	ChoiceSelectionTask	STZ	2	1
7	5	6.472	1	ChoiceSelectionTask	STZ	2	1
7	6	5.501	1	ChoiceSelectionTask	STZ	2	1
7	10	19.705	0	IntervalSelectionTask	STZ	2	1
7	4	7.168	0	ChoiceSelectionTask	STZ	2	1
7	1	8.676	1	QuantitativeTask	STZ	2	1
7	11	25.064	1	ChoiceSelectionTask	STZ	2	1
7	11	22.945	1	ChoiceSelectionTask	STZ	2	1
7	2	18.32	1	IntervalSelectionTask	STZ	2	1
7	12	17.303	1	QuantitativeTask	STZ	2	1
7	1	5.171	1	QuantitativeTask	STZ	2	1
7	8	9.988	1	QuantitativeTask	KNAVE	1	2
7	11	67.344	1	ChoiceSelectionTask	KNAVE	1	2
7	6	56.566	1	ChoiceSelectionTask	KNAVE	1	2
7	7	26.105	1	ChoiceSelectionTask	KNAVE	1	2
7	8	13.485	1	QuantitativeTask	KNAVE	1	2
7	4	9.941	1	ChoiceSelectionTask	KNAVE	1	2
7	3	15.343	1	IntervalSelectionTask	KNAVE	1	2
7	10	30.685	1	IntervalSelectionTask	KNAVE	1	2
7	7	15.296	0	ChoiceSelectionTask	KNAVE	1	2
7	10	19.298	1	IntervalSelectionTask	KNAVE	1	2
7	5	31.297	1	ChoiceSelectionTask	KNAVE	1	2
7	1	13.639	1	QuantitativeTask	KNAVE	1	2
7	9	9.981	0	QuantitativeTask	KNAVE	1	2
7	12	12.249	1	QuantitativeTask	KNAVE	1	2

7	11	18.004	1	ChoiceSelectionTask	KNAVE	1	2
7	2	20.523	1	IntervalSelectionTask	KNAVE	1	2
7	3	10.79	1	IntervalSelectionTask	KNAVE	1	2
7	9	8.632	1	QuantitativeTask	KNAVE	1	2
7	1	8.722	1	QuantitativeTask	KNAVE	1	2
7	5	10.445	1	ChoiceSelectionTask	KNAVE	1	2
7	6	12.411	1	ChoiceSelectionTask	KNAVE	1	2
7	2	25.172	1	IntervalSelectionTask	KNAVE	1	2
7	4	3.757	1	ChoiceSelectionTask	KNAVE	1	2
7	12	11.596	1	QuantitativeTask	KNAVE	1	2
8	11	25.012	1	ChoiceSelectionTask	KNAVE	2	1
8	4	24.694	1	ChoiceSelectionTask	KNAVE	2	1
8	5	18.499	1	ChoiceSelectionTask	KNAVE	2	1
8	10	21.754	1	IntervalSelectionTask	KNAVE	2	1
8	8	24.773	1	QuantitativeTask	KNAVE	2	1
8	2	23.681	1	IntervalSelectionTask	KNAVE	2	1
8	1	9.919	1	QuantitativeTask	KNAVE	2	1
8	9	14.103	1	QuantitativeTask	KNAVE	2	1
8	1	5.901	1	QuantitativeTask	KNAVE	2	1
8	6	17.988	1	ChoiceSelectionTask	KNAVE	2	1
8	3	22.699	1	IntervalSelectionTask	KNAVE	2	1
8	8	27.813	1	QuantitativeTask	KNAVE	2	1
8	3	24.675	1	IntervalSelectionTask	KNAVE	2	1
8	7	15.732	1	ChoiceSelectionTask	KNAVE	2	1
8	10	23.391	1	IntervalSelectionTask	KNAVE	2	1
8	2	20.307	0	IntervalSelectionTask	KNAVE	2	1
8	6	19.124	1	ChoiceSelectionTask	KNAVE	2	1
8	4	11.791	0	ChoiceSelectionTask	KNAVE	2	1
8	12	11.914	1	QuantitativeTask	KNAVE	2	1
8	11	38.002	1	ChoiceSelectionTask	KNAVE	2	1
8	9	16.003	1	QuantitativeTask	KNAVE	2	1
8	7	25.796	1	ChoiceSelectionTask	KNAVE	2	1
8	5	12.336	1	ChoiceSelectionTask	KNAVE	2	1
8	12	23.162	1	QuantitativeTask	KNAVE	2	1
8	5	10.828	1	ChoiceSelectionTask	STZ	1	2
8	8	16.208	1	QuantitativeTask	STZ	1	2
8	6	24.848	0	ChoiceSelectionTask	STZ	1	2
8	7	13.524	1	ChoiceSelectionTask	STZ	1	2
8	3	14.746	1	IntervalSelectionTask	STZ	1	2
8	11	47.899	1	ChoiceSelectionTask	STZ	1	2
8	7	8.078	1	ChoiceSelectionTask	STZ	1	2
8	1	7.094	1	QuantitativeTask	STZ	1	2
8	10	24.351	1	IntervalSelectionTask	STZ	1	2

8	6	19.047	1	ChoiceSelectionTask	STZ	1	2
8	10	17.072	1	IntervalSelectionTask	STZ	1	2
8	3	14.264	1	IntervalSelectionTask	STZ	1	2
8	9	9.577	1	QuantitativeTask	STZ	1	2
8	8	21.783	1	QuantitativeTask	STZ	1	2
8	4	12.109	1	ChoiceSelectionTask	STZ	1	2
8	5	11.705	1	ChoiceSelectionTask	STZ	1	2
8	4	7.955	0	ChoiceSelectionTask	STZ	1	2
8	1	8.481	0	QuantitativeTask	STZ	1	2
8	2	16.567	1	IntervalSelectionTask	STZ	1	2
8	9	7.858	1	QuantitativeTask	STZ	1	2
8	2	21.838	1	IntervalSelectionTask	STZ	1	2
8	12	8.073	1	QuantitativeTask	STZ	1	2
8	11	27.494	1	ChoiceSelectionTask	STZ	1	2
8	12	12.213	1	QuantitativeTask	STZ	1	2
9	1	13.669	1	QuantitativeTask	STZ	1	1
9	6	33.722	1	ChoiceSelectionTask	STZ	1	1
9	1	11.168	0	QuantitativeTask	STZ	1	1
9	12	24.71	1	QuantitativeTask	STZ	1	1
9	9	31.243	1	QuantitativeTask	STZ	1	1
9	12	25.503	1	QuantitativeTask	STZ	1	1
9	5	21.029	0	ChoiceSelectionTask	STZ	1	1
9	8	54.11	0	QuantitativeTask	STZ	1	1
9	10	78.206	1	IntervalSelectionTask	STZ	1	1
9	9	14.118	1	QuantitativeTask	STZ	1	1
9	2	47.161	1	IntervalSelectionTask	STZ	1	1
9	8	53.789	1	QuantitativeTask	STZ	1	1
9	7	38.117	1	ChoiceSelectionTask	STZ	1	1
9	11	55.715	1	ChoiceSelectionTask	STZ	1	1
9	4	57.335	1	ChoiceSelectionTask	STZ	1	1
9	10	32.297	1	IntervalSelectionTask	STZ	1	1
9	3	25.331	1	IntervalSelectionTask	STZ	1	1
9	11	36.287	1	ChoiceSelectionTask	STZ	1	1
9	5	25.614	1	ChoiceSelectionTask	STZ	1	1
9	4	24.47	1	ChoiceSelectionTask	STZ	1	1
9	6	44.935	0	ChoiceSelectionTask	STZ	1	1
9	3	13.889	1	IntervalSelectionTask	STZ	1	1
9	2	36.058	1	IntervalSelectionTask	STZ	1	1
9	7	31.42	1	ChoiceSelectionTask	STZ	1	1
9	1	27.442	0	QuantitativeTask	KNAVE	2	2
9	12	42.05	1	QuantitativeTask	KNAVE	2	2
9	12	26.408	1	QuantitativeTask	KNAVE	2	2
9	5	46.016	1	ChoiceSelectionTask	KNAVE	2	2

9	7	33.367	1	ChoiceSelectionTask	KNAVE	2	2
9	7	34.552	1	ChoiceSelectionTask	KNAVE	2	2
9	8	65.952	1	QuantitativeTask	KNAVE	2	2
9	9	12.84	0	QuantitativeTask	KNAVE	2	2
9	4	24.405	1	ChoiceSelectionTask	KNAVE	2	2
9	2	39.24	1	IntervalSelectionTask	KNAVE	2	2
9	2	26.746	1	IntervalSelectionTask	KNAVE	2	2
9	11	60.049	1	ChoiceSelectionTask	KNAVE	2	2
9	4	47.594	1	ChoiceSelectionTask	KNAVE	2	2
9	8	47.605	1	QuantitativeTask	KNAVE	2	2
9	3	31.127	1	IntervalSelectionTask	KNAVE	2	2
9	3	29.959	1	IntervalSelectionTask	KNAVE	2	2
9	5	27.814	1	ChoiceSelectionTask	KNAVE	2	2
9	10	43.366	1	IntervalSelectionTask	KNAVE	2	2
9	11	87.061	1	ChoiceSelectionTask	KNAVE	2	2
9	10	29.727	1	IntervalSelectionTask	KNAVE	2	2
9	6	33.809	1	ChoiceSelectionTask	KNAVE	2	2
9	9	12.072	1	QuantitativeTask	KNAVE	2	2
9	6	34.255	1	ChoiceSelectionTask	KNAVE	2	2
9	1	42.363	1	QuantitativeTask	KNAVE	2	2
10	11	47.079	1	ChoiceSelectionTask	KNAVE	1	1
10	6	32.135	0	ChoiceSelectionTask	KNAVE	1	1
10	8	18.418	1	QuantitativeTask	KNAVE	1	1
10	1	27.711	1	QuantitativeTask	KNAVE	1	1
10	7	19.769	1	ChoiceSelectionTask	KNAVE	1	1
10	10	31.97	1	IntervalSelectionTask	KNAVE	1	1
10	3	27.365	1	IntervalSelectionTask	KNAVE	1	1
10	4	21.857	1	ChoiceSelectionTask	KNAVE	1	1
10	11	47.815	1	ChoiceSelectionTask	KNAVE	1	1
10	2	67.334	0	IntervalSelectionTask	KNAVE	1	1
10	12	50.833	1	QuantitativeTask	KNAVE	1	1
10	10	61.522	1	IntervalSelectionTask	KNAVE	1	1
10	6	92.163	1	ChoiceSelectionTask	KNAVE	1	1
10	4	19.876	1	ChoiceSelectionTask	KNAVE	1	1
10	5	55.215	1	ChoiceSelectionTask	KNAVE	1	1
10	3	71.021	1	IntervalSelectionTask	KNAVE	1	1
10	8	44.708	1	QuantitativeTask	KNAVE	1	1
10	7	58.211	1	ChoiceSelectionTask	KNAVE	1	1
10	1	21.342	1	QuantitativeTask	KNAVE	1	1
10	5	24.528	1	ChoiceSelectionTask	KNAVE	1	1
10	2	47.662	1	IntervalSelectionTask	KNAVE	1	1
10	9	16.768	1	QuantitativeTask	KNAVE	1	1
10	12	32.319	1	QuantitativeTask	KNAVE	1	1

10	9	29.607	1	QuantitativeTask	KNAVE	1	1
10	8	37.119	1	QuantitativeTask	STZ	2	2
10	5	9.543	1	ChoiceSelectionTask	STZ	2	2
10	4	32.751	0	ChoiceSelectionTask	STZ	2	2
10	5	12.865	1	ChoiceSelectionTask	STZ	2	2
10	2	16.935	1	IntervalSelectionTask	STZ	2	2
10	2	24.188	1	IntervalSelectionTask	STZ	2	2
10	9	15.064	1	QuantitativeTask	STZ	2	2
10	8	28.916	0	QuantitativeTask	STZ	2	2
10	6	10	1	ChoiceSelectionTask	STZ	2	2
10	12	15.828	1	QuantitativeTask	STZ	2	2
10	12	18.538	1	QuantitativeTask	STZ	2	2
10	6	10.368	1	ChoiceSelectionTask	STZ	2	2
10	11	40.788	1	ChoiceSelectionTask	STZ	2	2
10	9	18.193	1	QuantitativeTask	STZ	2	2
10	7	24.574	1	ChoiceSelectionTask	STZ	2	2
10	10	20.965	1	IntervalSelectionTask	STZ	2	2
10	1	8.011	1	QuantitativeTask	STZ	2	2
10	1	11	1	QuantitativeTask	STZ	2	2
10	7	15.299	1	ChoiceSelectionTask	STZ	2	2
10	10	20.818	1	IntervalSelectionTask	STZ	2	2
10	11	20.352	1	ChoiceSelectionTask	STZ	2	2
10	3	15.867	1	IntervalSelectionTask	STZ	2	2
10	4	31.964	1	ChoiceSelectionTask	STZ	2	2
10	3	17.266	1	IntervalSelectionTask	STZ	2	2
11	3	14.645	0	IntervalSelectionTask	STZ	2	1
11	5	8.535	1	ChoiceSelectionTask	STZ	2	1
11	3	16.274	1	IntervalSelectionTask	STZ	2	1
11	1	7.689	1	QuantitativeTask	STZ	2	1
11	6	8.309	1	ChoiceSelectionTask	STZ	2	1
11	11	25.686	1	ChoiceSelectionTask	STZ	2	1
11	5	12.282	1	ChoiceSelectionTask	STZ	2	1
11	4	33.309	1	ChoiceSelectionTask	STZ	2	1
11	7	39.46	1	ChoiceSelectionTask	STZ	2	1
11	10	11.475	1	IntervalSelectionTask	STZ	2	1
11	2	21.149	1	IntervalSelectionTask	STZ	2	1
11	8	17.547	1	QuantitativeTask	STZ	2	1
11	12	8.213	1	QuantitativeTask	STZ	2	1
11	9	5.414	1	QuantitativeTask	STZ	2	1
11	12	5.394	1	QuantitativeTask	STZ	2	1
11	4	7.564	1	ChoiceSelectionTask	STZ	2	1
11	1	4.111	1	QuantitativeTask	STZ	2	1
11	10	10.973	1	IntervalSelectionTask	STZ	2	1

11	7	10.972	1	ChoiceSelectionTask	STZ	2	1
11	2	17.67	1	IntervalSelectionTask	STZ	2	1
11	8	17.088	1	QuantitativeTask	STZ	2	1
11	11	29.759	1	ChoiceSelectionTask	STZ	2	1
11	6	11.752	1	ChoiceSelectionTask	STZ	2	1
11	9	8.962	1	QuantitativeTask	STZ	2	1
11	10	30.235	1	IntervalSelectionTask	KNAVE	1	2
11	9	9.742	1	QuantitativeTask	KNAVE	1	2
11	3	18.983	1	IntervalSelectionTask	KNAVE	1	2
11	1	6.582	1	QuantitativeTask	KNAVE	1	2
11	12	23.835	1	QuantitativeTask	KNAVE	1	2
11	8	18.21	1	QuantitativeTask	KNAVE	1	2
11	6	19.748	1	ChoiceSelectionTask	KNAVE	1	2
11	8	23.161	1	QuantitativeTask	KNAVE	1	2
11	1	9.783	1	QuantitativeTask	KNAVE	1	2
11	5	21.008	1	ChoiceSelectionTask	KNAVE	1	2
11	11	49.882	1	ChoiceSelectionTask	KNAVE	1	2
11	10	22.47	1	IntervalSelectionTask	KNAVE	1	2
11	2	18.09	1	IntervalSelectionTask	KNAVE	1	2
11	4	8.581	0	ChoiceSelectionTask	KNAVE	1	2
11	11	30.773	1	ChoiceSelectionTask	KNAVE	1	2
11	12	16.366	1	QuantitativeTask	KNAVE	1	2
11	6	24.346	1	ChoiceSelectionTask	KNAVE	1	2
11	7	25.513	1	ChoiceSelectionTask	KNAVE	1	2
11	2	21.4	1	IntervalSelectionTask	KNAVE	1	2
11	4	11.417	1	ChoiceSelectionTask	KNAVE	1	2
11	7	14.384	1	ChoiceSelectionTask	KNAVE	1	2
11	5	11.58	1	ChoiceSelectionTask	KNAVE	1	2
11	9	9.55	1	QuantitativeTask	KNAVE	1	2
11	3	14.236	1	IntervalSelectionTask	KNAVE	1	2
12	4	57.683	1	ChoiceSelectionTask	KNAVE	2	1
12	2	59.806	1	IntervalSelectionTask	KNAVE	2	1
12	7	34.86	1	ChoiceSelectionTask	KNAVE	2	1
12	5	34.9	1	ChoiceSelectionTask	KNAVE	2	1
12	2	51.128	0	IntervalSelectionTask	KNAVE	2	1
12	7	95.111	1	ChoiceSelectionTask	KNAVE	2	1
12	4	15.34	0	ChoiceSelectionTask	KNAVE	2	1
12	9	22.294	1	QuantitativeTask	KNAVE	2	1
12	12	34.148	1	QuantitativeTask	KNAVE	2	1
12	12	23.473	1	QuantitativeTask	KNAVE	2	1
12	6	32.78	0	ChoiceSelectionTask	KNAVE	2	1
12	1	18.867	1	QuantitativeTask	KNAVE	2	1
12	11	54.7	1	ChoiceSelectionTask	KNAVE	2	1

12	3	48.122	1	IntervalSelectionTask	KNAVE	2	1
12	1	17.409	1	QuantitativeTask	KNAVE	2	1
12	10	44.524	1	IntervalSelectionTask	KNAVE	2	1
12	5	29.346	0	ChoiceSelectionTask	KNAVE	2	1
12	10	43.294	1	IntervalSelectionTask	KNAVE	2	1
12	8	40.242	1	QuantitativeTask	KNAVE	2	1
12	9	24.222	1	QuantitativeTask	KNAVE	2	1
12	6	27.775	1	ChoiceSelectionTask	KNAVE	2	1
12	11	45.114	1	ChoiceSelectionTask	KNAVE	2	1
12	3	35.563	1	IntervalSelectionTask	KNAVE	2	1
12	8	33.424	1	QuantitativeTask	KNAVE	2	1
12	10	23.582	1	IntervalSelectionTask	STZ	1	2
12	9	10.165	1	QuantitativeTask	STZ	1	2
12	8	23.548	1	QuantitativeTask	STZ	1	2
12	1	47.665	1	QuantitativeTask	STZ	1	2
12	6	41.377	1	ChoiceSelectionTask	STZ	1	2
12	5	16.149	1	ChoiceSelectionTask	STZ	1	2
12	3	25.714	1	IntervalSelectionTask	STZ	1	2
12	5	15.026	1	ChoiceSelectionTask	STZ	1	2
12	11	40.518	1	ChoiceSelectionTask	STZ	1	2
12	2	32.242	1	IntervalSelectionTask	STZ	1	2
12	7	20.763	1	ChoiceSelectionTask	STZ	1	2
12	6	21.607	1	ChoiceSelectionTask	STZ	1	2
12	1	16.791	1	QuantitativeTask	STZ	1	2
12	2	23.774	1	IntervalSelectionTask	STZ	1	2
12	7	9.925	1	ChoiceSelectionTask	STZ	1	2
12	10	47.767	1	IntervalSelectionTask	STZ	1	2
12	4	17.2	1	ChoiceSelectionTask	STZ	1	2
12	3	21.733	1	IntervalSelectionTask	STZ	1	2
12	12	9.66	1	QuantitativeTask	STZ	1	2
12	8	26.31	1	QuantitativeTask	STZ	1	2
12	4	15.132	1	ChoiceSelectionTask	STZ	1	2
12	11	20.291	1	ChoiceSelectionTask	STZ	1	2
12	9	13.773	1	QuantitativeTask	STZ	1	2
12	12	22.959	1	QuantitativeTask	STZ	1	2
13	5	12.573	1	ChoiceSelectionTask	STZ	1	1
13	1	11.512	1	QuantitativeTask	STZ	1	1
13	7	37.946	1	ChoiceSelectionTask	STZ	1	1
13	9	5.905	1	QuantitativeTask	STZ	1	1
13	1	8.422	1	QuantitativeTask	STZ	1	1
13	11	29.704	1	ChoiceSelectionTask	STZ	1	1
13	6	63.318	1	ChoiceSelectionTask	STZ	1	1
13	10	39.612	1	IntervalSelectionTask	STZ	1	1

13	9	10.063	1	QuantitativeTask	STZ	1	1
13	6	14.119	1	ChoiceSelectionTask	STZ	1	1
13	4	22.919	1	ChoiceSelectionTask	STZ	1	1
13	12	9.722	1	QuantitativeTask	STZ	1	1
13	10	14.976	1	IntervalSelectionTask	STZ	1	1
13	2	25.938	0	IntervalSelectionTask	STZ	1	1
13	11	42.056	1	ChoiceSelectionTask	STZ	1	1
13	8	24.358	1	QuantitativeTask	STZ	1	1
13	5	9.938	1	ChoiceSelectionTask	STZ	1	1
13	8	20.431	1	QuantitativeTask	STZ	1	1
13	2	20.257	0	IntervalSelectionTask	STZ	1	1
13	12	13.496	1	QuantitativeTask	STZ	1	1
13	4	14.445	1	ChoiceSelectionTask	STZ	1	1
13	7	13.396	1	ChoiceSelectionTask	STZ	1	1
13	3	16.588	1	IntervalSelectionTask	STZ	1	1
13	3	17.107	1	IntervalSelectionTask	STZ	1	1
13	1	10.12	1	QuantitativeTask	KNAVE	2	2
13	11	33.514	1	ChoiceSelectionTask	KNAVE	2	2
13	3	18.795	1	IntervalSelectionTask	KNAVE	2	2
13	4	7.137	1	ChoiceSelectionTask	KNAVE	2	2
13	8	24.558	1	QuantitativeTask	KNAVE	2	2
13	4	6.672	1	ChoiceSelectionTask	KNAVE	2	2
13	3	15.412	1	IntervalSelectionTask	KNAVE	2	2
13	5	11.512	1	ChoiceSelectionTask	KNAVE	2	2
13	6	12.609	1	ChoiceSelectionTask	KNAVE	2	2
13	5	14.274	1	ChoiceSelectionTask	KNAVE	2	2
13	8	19.335	1	QuantitativeTask	KNAVE	2	2
13	7	17.865	1	ChoiceSelectionTask	KNAVE	2	2
13	1	6.819	1	QuantitativeTask	KNAVE	2	2
13	2	17.16	1	IntervalSelectionTask	KNAVE	2	2
13	6	15.262	1	ChoiceSelectionTask	KNAVE	2	2
13	7	16.823	1	ChoiceSelectionTask	KNAVE	2	2
13	2	33.76	1	IntervalSelectionTask	KNAVE	2	2
13	11	49.352	1	ChoiceSelectionTask	KNAVE	2	2
13	9	10.937	1	QuantitativeTask	KNAVE	2	2
13	10	24.53	1	IntervalSelectionTask	KNAVE	2	2
13	12	11.975	1	QuantitativeTask	KNAVE	2	2
13	12	11.814	1	QuantitativeTask	KNAVE	2	2
13	10	14.832	1	IntervalSelectionTask	KNAVE	2	2
13	9	6.853	1	QuantitativeTask	KNAVE	2	2
14	1	9.488	1	QuantitativeTask	KNAVE	1	1
14	12	19.816	1	QuantitativeTask	KNAVE	1	1
14	3	23.413	1	IntervalSelectionTask	KNAVE	1	1

14	4	29.411	1	ChoiceSelectionTask	KNAVE	1	1
14	5	28.819	1	ChoiceSelectionTask	KNAVE	1	1
14	2	25.559	1	IntervalSelectionTask	KNAVE	1	1
14	6	20.206	1	ChoiceSelectionTask	KNAVE	1	1
14	8	11.803	1	QuantitativeTask	KNAVE	1	1
14	2	31.606	1	IntervalSelectionTask	KNAVE	1	1
14	7	10.163	1	ChoiceSelectionTask	KNAVE	1	1
14	9	5.901	1	QuantitativeTask	KNAVE	1	1
14	5	13.42	1	ChoiceSelectionTask	KNAVE	1	1
14	9	14.333	1	QuantitativeTask	KNAVE	1	1
14	6	26.66	1	ChoiceSelectionTask	KNAVE	1	1
14	4	13.994	0	ChoiceSelectionTask	KNAVE	1	1
14	3	20.533	1	IntervalSelectionTask	KNAVE	1	1
14	11	31.252	1	ChoiceSelectionTask	KNAVE	1	1
14	12	23.854	1	QuantitativeTask	KNAVE	1	1
14	8	25.902	1	QuantitativeTask	KNAVE	1	1
14	7	39.144	1	ChoiceSelectionTask	KNAVE	1	1
14	11	21.483	1	ChoiceSelectionTask	KNAVE	1	1
14	10	15.838	1	IntervalSelectionTask	KNAVE	1	1
14	10	34.599	1	IntervalSelectionTask	KNAVE	1	1
14	1	8.46	1	QuantitativeTask	KNAVE	1	1
14	1	9.307	1	QuantitativeTask	STZ	2	2
14	5	9.519	1	ChoiceSelectionTask	STZ	2	2
14	3	15.261	1	IntervalSelectionTask	STZ	2	2
14	12	12.131	1	QuantitativeTask	STZ	2	2
14	5	8.656	1	ChoiceSelectionTask	STZ	2	2
14	2	29.249	1	IntervalSelectionTask	STZ	2	2
14	7	17.722	1	ChoiceSelectionTask	STZ	2	2
14	6	8.061	1	ChoiceSelectionTask	STZ	2	2
14	12	6.271	1	QuantitativeTask	STZ	2	2
14	11	20.092	1	ChoiceSelectionTask	STZ	2	2
14	9	16.327	1	QuantitativeTask	STZ	2	2
14	3	12.022	1	IntervalSelectionTask	STZ	2	2
14	9	9.448	1	QuantitativeTask	STZ	2	2
14	11	11.041	1	ChoiceSelectionTask	STZ	2	2
14	7	15.131	1	ChoiceSelectionTask	STZ	2	2
14	8	20.096	1	QuantitativeTask	STZ	2	2
14	1	4.309	1	QuantitativeTask	STZ	2	2
14	4	12.862	1	ChoiceSelectionTask	STZ	2	2
14	2	14.982	1	IntervalSelectionTask	STZ	2	2
14	6	12.651	1	ChoiceSelectionTask	STZ	2	2
14	10	22.856	1	IntervalSelectionTask	STZ	2	2
14	10	11.273	1	IntervalSelectionTask	STZ	2	2

14	4	8.485	0	ChoiceSelectionTask	STZ	2	2
14	8	16.209	1	QuantitativeTask	STZ	2	2
15	8	27.361	0	QuantitativeTask	STZ	2	1
15	9	14.741	1	QuantitativeTask	STZ	2	1
15	11	60.643	1	ChoiceSelectionTask	STZ	2	1
15	1	6.3	1	QuantitativeTask	STZ	2	1
15	12	11.287	0	QuantitativeTask	STZ	2	1
15	3	16.808	1	IntervalSelectionTask	STZ	2	1
15	4	43.644	1	ChoiceSelectionTask	STZ	2	1
15	5	11.957	1	ChoiceSelectionTask	STZ	2	1
15	8	13.473	1	QuantitativeTask	STZ	2	1
15	11	18.738	1	ChoiceSelectionTask	STZ	2	1
15	6	11.967	1	ChoiceSelectionTask	STZ	2	1
15	2	41.094	1	IntervalSelectionTask	STZ	2	1
15	2	25.068	1	IntervalSelectionTask	STZ	2	1
15	9	13.517	1	QuantitativeTask	STZ	2	1
15	3	12.829	1	IntervalSelectionTask	STZ	2	1
15	12	12.442	1	QuantitativeTask	STZ	2	1
15	10	11.089	1	IntervalSelectionTask	STZ	2	1
15	10	15.477	1	IntervalSelectionTask	STZ	2	1
15	6	6.862	1	ChoiceSelectionTask	STZ	2	1
15	7	49.258	1	ChoiceSelectionTask	STZ	2	1
15	7	19.992	1	ChoiceSelectionTask	STZ	2	1
15	1	5.23	1	QuantitativeTask	STZ	2	1
15	4	10.509	1	ChoiceSelectionTask	STZ	2	1
15	5	5.266	1	ChoiceSelectionTask	STZ	2	1
15	1	11.574	1	QuantitativeTask	KNAVE	1	2
15	12	18.406	1	QuantitativeTask	KNAVE	1	2
15	5	29.853	1	ChoiceSelectionTask	KNAVE	1	2
15	6	20.404	1	ChoiceSelectionTask	KNAVE	1	2
15	8	15.811	1	QuantitativeTask	KNAVE	1	2
15	1	8.103	1	QuantitativeTask	KNAVE	1	2
15	11	35.763	1	ChoiceSelectionTask	KNAVE	1	2
15	11	22.4	1	ChoiceSelectionTask	KNAVE	1	2
15	3	11.089	1	IntervalSelectionTask	KNAVE	1	2
15	12	21.596	1	QuantitativeTask	KNAVE	1	2
15	8	17.486	1	QuantitativeTask	KNAVE	1	2
15	7	20.818	1	ChoiceSelectionTask	KNAVE	1	2
15	4	8.445	1	ChoiceSelectionTask	KNAVE	1	2
15	9	13.915	1	QuantitativeTask	KNAVE	1	2
15	3	12.672	1	IntervalSelectionTask	KNAVE	1	2
15	10	46.549	1	IntervalSelectionTask	KNAVE	1	2
15	2	33.86	1	IntervalSelectionTask	KNAVE	1	2

15	10	18.095	1	IntervalSelectionTask	KNAVE	1	2
15	4	7.668	1	ChoiceSelectionTask	KNAVE	1	2
15	7	15.705	1	ChoiceSelectionTask	KNAVE	1	2
15	5	10.768	1	ChoiceSelectionTask	KNAVE	1	2
15	9	10.599	1	QuantitativeTask	KNAVE	1	2
15	2	19.341	1	IntervalSelectionTask	KNAVE	1	2
15	6	20.818	1	ChoiceSelectionTask	KNAVE	1	2
16	3	18.993	0	IntervalSelectionTask	KNAVE	2	1
16	12	15.583	1	QuantitativeTask	KNAVE	2	1
16	2	20.346	1	IntervalSelectionTask	KNAVE	2	1
16	4	27.969	0	ChoiceSelectionTask	KNAVE	2	1
16	8	32.205	1	QuantitativeTask	KNAVE	2	1
16	7	17.15	1	ChoiceSelectionTask	KNAVE	2	1
16	4	22.164	0	ChoiceSelectionTask	KNAVE	2	1
16	6	14.531	1	ChoiceSelectionTask	KNAVE	2	1
16	8	17.377	1	QuantitativeTask	KNAVE	2	1
16	9	27.247	1	QuantitativeTask	KNAVE	2	1
16	12	13.191	1	QuantitativeTask	KNAVE	2	1
16	5	12.069	1	ChoiceSelectionTask	KNAVE	2	1
16	10	38.843	0	IntervalSelectionTask	KNAVE	2	1
16	9	17.545	1	QuantitativeTask	KNAVE	2	1
16	5	60.25	1	ChoiceSelectionTask	KNAVE	2	1
16	3	14.975	1	IntervalSelectionTask	KNAVE	2	1
16	1	8.007	1	QuantitativeTask	KNAVE	2	1
16	11	28.028	1	ChoiceSelectionTask	KNAVE	2	1
16	11	23.635	1	ChoiceSelectionTask	KNAVE	2	1
16	6	18.1	1	ChoiceSelectionTask	KNAVE	2	1
16	1	11.4	1	QuantitativeTask	KNAVE	2	1
16	10	31.428	1	IntervalSelectionTask	KNAVE	2	1
16	7	17.571	1	ChoiceSelectionTask	KNAVE	2	1
16	2	16.751	1	IntervalSelectionTask	KNAVE	2	1
16	3	17.013	1	IntervalSelectionTask	STZ	1	2
16	3	12.919	1	IntervalSelectionTask	STZ	1	2
16	12	10.305	1	QuantitativeTask	STZ	1	2
16	5	28.079	1	ChoiceSelectionTask	STZ	1	2
16	10	34.969	1	IntervalSelectionTask	STZ	1	2
16	7	16.828	1	ChoiceSelectionTask	STZ	1	2
16	4	12.977	1	ChoiceSelectionTask	STZ	1	2
16	8	15.594	0	QuantitativeTask	STZ	1	2
16	6	14.468	1	ChoiceSelectionTask	STZ	1	2
16	12	13.633	1	QuantitativeTask	STZ	1	2
16	11	17.69	1	ChoiceSelectionTask	STZ	1	2
16	2	37.315	1	IntervalSelectionTask	STZ	1	2

16	1	7.437	1	QuantitativeTask	STZ	1	2
16	8	27.58	1	QuantitativeTask	STZ	1	2
16	11	14.906	0	ChoiceSelectionTask	STZ	1	2
16	9	9.149	1	QuantitativeTask	STZ	1	2
16	1	7.004	1	QuantitativeTask	STZ	1	2
16	2	18.447	1	IntervalSelectionTask	STZ	1	2
16	4	11.166	1	ChoiceSelectionTask	STZ	1	2
16	6	6.885	1	ChoiceSelectionTask	STZ	1	2
16	9	6.162	1	QuantitativeTask	STZ	1	2
16	10	14.415	1	IntervalSelectionTask	STZ	1	2
16	7	11.217	1	ChoiceSelectionTask	STZ	1	2
16	5	5.832	1	ChoiceSelectionTask	STZ	1	2
17	5	6.935	1	ChoiceSelectionTask	STZ	1	1
17	3	31.863	1	IntervalSelectionTask	STZ	1	1
17	1	5.666	0	QuantitativeTask	STZ	1	1
17	6	8.706	1	ChoiceSelectionTask	STZ	1	1
17	9	7.634	1	QuantitativeTask	STZ	1	1
17	11	44.605	1	ChoiceSelectionTask	STZ	1	1
17	4	12.684	1	ChoiceSelectionTask	STZ	1	1
17	9	8.495	1	QuantitativeTask	STZ	1	1
17	8	22.448	1	QuantitativeTask	STZ	1	1
17	6	30.641	1	ChoiceSelectionTask	STZ	1	1
17	7	20.808	1	ChoiceSelectionTask	STZ	1	1
17	7	7.799	1	ChoiceSelectionTask	STZ	1	1
17	3	36.622	1	IntervalSelectionTask	STZ	1	1
17	2	16.003	1	IntervalSelectionTask	STZ	1	1
17	2	16.483	1	IntervalSelectionTask	STZ	1	1
17	4	5.931	1	ChoiceSelectionTask	STZ	1	1
17	8	11.315	1	QuantitativeTask	STZ	1	1
17	5	5.173	1	ChoiceSelectionTask	STZ	1	1
17	10	24.834	1	IntervalSelectionTask	STZ	1	1
17	12	7.775	1	QuantitativeTask	STZ	1	1
17	11	11.66	1	ChoiceSelectionTask	STZ	1	1
17	12	8.84	1	QuantitativeTask	STZ	1	1
17	10	11.364	1	IntervalSelectionTask	STZ	1	1
17	1	6.11	1	QuantitativeTask	STZ	1	1
17	7	8.733	1	ChoiceSelectionTask	KNAVE	2	2
17	2	24.841	1	IntervalSelectionTask	KNAVE	2	2
17	4	7.458	1	ChoiceSelectionTask	KNAVE	2	2
17	7	15.224	1	ChoiceSelectionTask	KNAVE	2	2
17	11	29.876	1	ChoiceSelectionTask	KNAVE	2	2
17	10	43.12	1	IntervalSelectionTask	KNAVE	2	2
17	1	8.535	1	QuantitativeTask	KNAVE	2	2

17	2	14.062	1	IntervalSelectionTask	KNAVE	2	2
17	10	32.6	1	IntervalSelectionTask	KNAVE	2	2
17	3	12.195	1	IntervalSelectionTask	KNAVE	2	2
17	5	16.28	1	ChoiceSelectionTask	KNAVE	2	2
17	12	11.236	1	QuantitativeTask	KNAVE	2	2
17	6	16.433	1	ChoiceSelectionTask	KNAVE	2	2
17	4	8.749	1	ChoiceSelectionTask	KNAVE	2	2
17	12	16.887	1	QuantitativeTask	KNAVE	2	2
17	1	5.162	1	QuantitativeTask	KNAVE	2	2
17	9	8.229	1	QuantitativeTask	KNAVE	2	2
17	8	18.622	1	QuantitativeTask	KNAVE	2	2
17	5	7.618	1	ChoiceSelectionTask	KNAVE	2	2
17	11	21.265	1	ChoiceSelectionTask	KNAVE	2	2
17	6	10.509	1	ChoiceSelectionTask	KNAVE	2	2
17	8	11.613	1	QuantitativeTask	KNAVE	2	2
17	9	8.505	1	QuantitativeTask	KNAVE	2	2
17	3	9.506	1	IntervalSelectionTask	KNAVE	2	2
18	2	39.368	1	IntervalSelectionTask	KNAVE	1	1
18	3	37.72	1	IntervalSelectionTask	KNAVE	1	1
18	3	25.239	1	IntervalSelectionTask	KNAVE	1	1
18	1	16.607	1	QuantitativeTask	KNAVE	1	1
18	9	21.541	0	QuantitativeTask	KNAVE	1	1
18	1	22.899	1	QuantitativeTask	KNAVE	1	1
18	12	22.412	1	QuantitativeTask	KNAVE	1	1
18	8	32.053	1	QuantitativeTask	KNAVE	1	1
18	12	29.261	1	QuantitativeTask	KNAVE	1	1
18	5	51.926	1	ChoiceSelectionTask	KNAVE	1	1
18	11	70.926	1	ChoiceSelectionTask	KNAVE	1	1
18	10	43.226	0	IntervalSelectionTask	KNAVE	1	1
18	4	21.549	1	ChoiceSelectionTask	KNAVE	1	1
18	9	16.673	1	QuantitativeTask	KNAVE	1	1
18	5	74.862	1	ChoiceSelectionTask	KNAVE	1	1
18	11	52.746	1	ChoiceSelectionTask	KNAVE	1	1
18	2	52.05	1	IntervalSelectionTask	KNAVE	1	1
18	10	25.161	1	IntervalSelectionTask	KNAVE	1	1
18	8	25.884	1	QuantitativeTask	KNAVE	1	1
18	6	64.843	1	ChoiceSelectionTask	KNAVE	1	1
18	7	18.443	1	ChoiceSelectionTask	KNAVE	1	1
18	6	34.605	1	ChoiceSelectionTask	KNAVE	1	1
18	7	31.376	1	ChoiceSelectionTask	KNAVE	1	1
18	4	7.186	1	ChoiceSelectionTask	KNAVE	1	1
18	7	19.449	1	ChoiceSelectionTask	STZ	2	2
18	6	17.887	1	ChoiceSelectionTask	STZ	2	2

18	12	12.256	1	QuantitativeTask	STZ	2	2
18	2	31.662	1	IntervalSelectionTask	STZ	2	2
18	10	30.521	1	IntervalSelectionTask	STZ	2	2
18	3	29.357	1	IntervalSelectionTask	STZ	2	2
18	8	29.098	1	QuantitativeTask	STZ	2	2
18	8	20.19	1	QuantitativeTask	STZ	2	2
18	11	56.596	1	ChoiceSelectionTask	STZ	2	2
18	7	21.047	1	ChoiceSelectionTask	STZ	2	2
18	1	14.607	1	QuantitativeTask	STZ	2	2
18	11	32.614	1	ChoiceSelectionTask	STZ	2	2
18	5	35.364	1	ChoiceSelectionTask	STZ	2	2
18	9	11.975	1	QuantitativeTask	STZ	2	2
18	5	31.458	1	ChoiceSelectionTask	STZ	2	2
18	9	15.952	1	QuantitativeTask	STZ	2	2
18	10	27.931	1	IntervalSelectionTask	STZ	2	2
18	12	14.127	1	QuantitativeTask	STZ	2	2
18	3	27.966	1	IntervalSelectionTask	STZ	2	2
18	4	9.049	1	ChoiceSelectionTask	STZ	2	2
18	4	12.159	1	ChoiceSelectionTask	STZ	2	2
18	2	26.412	1	IntervalSelectionTask	STZ	2	2
18	1	7.418	1	QuantitativeTask	STZ	2	2
18	6	15.796	1	ChoiceSelectionTask	STZ	2	2
19	10	32.123	1	IntervalSelectionTask	STZ	2	1
19	12	26.17	1	QuantitativeTask	STZ	2	1
19	9	6.614	1	QuantitativeTask	STZ	2	1
19	12	27.916	1	QuantitativeTask	STZ	2	1
19	3	20.904	1	IntervalSelectionTask	STZ	2	1
19	2	63.933	1	IntervalSelectionTask	STZ	2	1
19	3	15.179	1	IntervalSelectionTask	STZ	2	1
19	5	15.122	1	ChoiceSelectionTask	STZ	2	1
19	4	40.835	1	ChoiceSelectionTask	STZ	2	1
19	7	79.438	0	ChoiceSelectionTask	STZ	2	1
19	4	23.723	0	ChoiceSelectionTask	STZ	2	1
19	6	21.291	1	ChoiceSelectionTask	STZ	2	1
19	7	59.499	0	ChoiceSelectionTask	STZ	2	1
19	8	62.65	1	QuantitativeTask	STZ	2	1
19	11	33.079	1	ChoiceSelectionTask	STZ	2	1
19	6	15.733	1	ChoiceSelectionTask	STZ	2	1
19	9	11.285	1	QuantitativeTask	STZ	2	1
19	1	8.735	1	QuantitativeTask	STZ	2	1
19	11	23.5	1	ChoiceSelectionTask	STZ	2	1
19	5	4.319	1	ChoiceSelectionTask	STZ	2	1
19	2	59.444	0	IntervalSelectionTask	STZ	2	1

19	10	16.689	1	IntervalSelectionTask	STZ	2	1
19	1	5.688	1	QuantitativeTask	STZ	2	1
19	8	46.833	1	QuantitativeTask	STZ	2	1
19	3	19.051	1	IntervalSelectionTask	KNAVE	1	2
19	8	25.617	1	QuantitativeTask	KNAVE	1	2
19	6	33.667	0	ChoiceSelectionTask	KNAVE	1	2
19	4	22.241	1	ChoiceSelectionTask	KNAVE	1	2
19	1	7.738	1	QuantitativeTask	KNAVE	1	2
19	7	25.165	1	ChoiceSelectionTask	KNAVE	1	2
19	12	20.8	1	QuantitativeTask	KNAVE	1	2
19	11	79.48	1	ChoiceSelectionTask	KNAVE	1	2
19	3	14.937	0	IntervalSelectionTask	KNAVE	1	2
19	12	22.404	1	QuantitativeTask	KNAVE	1	2
19	6	14.752	1	ChoiceSelectionTask	KNAVE	1	2
19	5	6.233	1	ChoiceSelectionTask	KNAVE	1	2
19	10	16.584	1	IntervalSelectionTask	KNAVE	1	2
19	1	14.404	1	QuantitativeTask	KNAVE	1	2
19	8	28.071	1	QuantitativeTask	KNAVE	1	2
19	11	40.848	1	ChoiceSelectionTask	KNAVE	1	2
19	9	15.163	1	QuantitativeTask	KNAVE	1	2
19	2	31.049	0	IntervalSelectionTask	KNAVE	1	2
19	4	23.869	1	ChoiceSelectionTask	KNAVE	1	2
19	2	14.938	1	IntervalSelectionTask	KNAVE	1	2
19	5	21.122	1	ChoiceSelectionTask	KNAVE	1	2
19	7	19.289	1	ChoiceSelectionTask	KNAVE	1	2
19	9	8.465	1	QuantitativeTask	KNAVE	1	2
19	10	34.008	1	IntervalSelectionTask	KNAVE	1	2
20	8	24.76	1	QuantitativeTask	KNAVE	2	1
20	8	55.773	1	QuantitativeTask	KNAVE	2	1
20	12	35.868	1	QuantitativeTask	KNAVE	2	1
20	6	64.385	1	ChoiceSelectionTask	KNAVE	2	1
20	6	25.085	1	ChoiceSelectionTask	KNAVE	2	1
20	10	37.746	1	IntervalSelectionTask	KNAVE	2	1
20	9	28.638	1	QuantitativeTask	KNAVE	2	1
20	7	48.661	1	ChoiceSelectionTask	KNAVE	2	1
20	7	25.487	1	ChoiceSelectionTask	KNAVE	2	1
20	2	21.317	1	IntervalSelectionTask	KNAVE	2	1
20	5	22.543	1	ChoiceSelectionTask	KNAVE	2	1
20	11	36.125	1	ChoiceSelectionTask	KNAVE	2	1
20	10	44.451	0	IntervalSelectionTask	KNAVE	2	1
20	9	14.383	1	QuantitativeTask	KNAVE	2	1
20	3	41.966	1	IntervalSelectionTask	KNAVE	2	1
20	1	5.132	1	QuantitativeTask	KNAVE	2	1

20	11	58.924	1	ChoiceSelectionTask	KNAVE	2	1
20	3	19.558	1	IntervalSelectionTask	KNAVE	2	1
20	4	7.28	1	ChoiceSelectionTask	KNAVE	2	1
20	1	7.815	1	QuantitativeTask	KNAVE	2	1
20	12	7.697	1	QuantitativeTask	KNAVE	2	1
20	5	18.809	1	ChoiceSelectionTask	KNAVE	2	1
20	4	13.676	1	ChoiceSelectionTask	KNAVE	2	1
20	2	25.773	1	IntervalSelectionTask	KNAVE	2	1
20	3	9.701	1	IntervalSelectionTask	STZ	1	2
20	1	27.288	1	QuantitativeTask	STZ	1	2
20	11	19.453	1	ChoiceSelectionTask	STZ	1	2
20	5	6.807	1	ChoiceSelectionTask	STZ	1	2
20	8	21.609	1	QuantitativeTask	STZ	1	2
20	4	5.46	0	ChoiceSelectionTask	STZ	1	2
20	9	5.416	1	QuantitativeTask	STZ	1	2
20	7	9.524	1	ChoiceSelectionTask	STZ	1	2
20	5	15.828	1	ChoiceSelectionTask	STZ	1	2
20	10	8.499	1	IntervalSelectionTask	STZ	1	2
20	1	11.28	1	QuantitativeTask	STZ	1	2
20	3	11.593	1	IntervalSelectionTask	STZ	1	2
20	9	18.862	1	QuantitativeTask	STZ	1	2
20	12	9.059	1	QuantitativeTask	STZ	1	2
20	7	12.63	1	ChoiceSelectionTask	STZ	1	2
20	4	19.764	0	ChoiceSelectionTask	STZ	1	2
20	6	11.869	1	ChoiceSelectionTask	STZ	1	2
20	12	8.52	1	QuantitativeTask	STZ	1	2
20	2	17.528	1	IntervalSelectionTask	STZ	1	2
20	10	51.849	1	IntervalSelectionTask	STZ	1	2
20	8	35.625	1	QuantitativeTask	STZ	1	2
20	2	13.412	1	IntervalSelectionTask	STZ	1	2
20	6	21.997	1	ChoiceSelectionTask	STZ	1	2
20	11	37.055	1	ChoiceSelectionTask	STZ	1	2

APPENDIX C

Post-Experiment Survey

Test person	*Preference*
1	STZ
2	STZ
3	STZ
4	STZ
5	STZ
6	STZ
7	STZ
8	STZ
9	STZ
10	STZ
11	STZ
12	STZ
13	STZ
14	STZ
15	STZ
16	STZ
17	STZ
18	STZ
19	KNAVE
20	STZ

Table C.2: Personal preference of the test persons regarding the visualization technique

The questionnaire that was used to collect these data can be seen in Figure C.1.

Fragebogen

TP-ID: _____

Alter: _____

Job: _____

Ausbildung: _____

Selbsteinschätzung(unerfahren, mittel, gut, sehr gut)

Liniendiagramme: _____

Balkendiagramme: _____

Datenanalyse: _____

Computer: _____

Persönliche Präferenz: _____ (STZ / KNAVE)

STZ- Animation/Interaktion sinnvoll? _____

Feedback/ persönliche Anmerkungen:

Figure C.1: Questionnaire that was used to collect the demographic and self assignment data of the test persons before the experiment and to collect the feedback of the experiments after completing the experiment.

APPENDIX D

R Scripts for Individual Task Analysis

The following R script tests dataset and experiment round for significant differences in task completion times. Individual task completion times are tested for normal, log-normal distribution and equal variance for each experiment round. At the end, two and one sided t-tests are performed to find significant difference between the tasks for the visualization types.

```
# Import data
aggregatedData <- read.table("raw_data.csv", header=TRUE, sep = ";")
aggregatedData <- aggregatedData[2:9]

# ================================================
# VISTYPE STZ --- Summary Statistics ---
# ================================================

TimingsSTZ <- aggregatedData[aggregatedData[6]=="STZ",1:8]

sumstats_STZ <- data.frame(Task=1, MIN=min(TimingsSTZ[TimingsSTZ[2]==1,3]),
        Q1=quantile(TimingsSTZ[TimingsSTZ[2]==1,3],0.25),
        MEDIAN=median(TimingsSTZ[TimingsSTZ[2]==1,3]),
        MEAN=mean(TimingsSTZ[TimingsSTZ[2]==1,3]),
        Q3=quantile(TimingsSTZ[TimingsSTZ[2]==1,3],0.75),
        MAX=max(TimingsSTZ[TimingsSTZ[2]==1,3]),
        SD=sd(TimingsSTZ[TimingsSTZ[2]==1,3]))
for (taskNr in 2:12) {
        sumstats_STZ <- rbind(sumstats_STZ, c(taskNr,
                min(TimingsSTZ[TimingsSTZ[2]==taskNr,3]),
                quantile(TimingsSTZ[TimingsSTZ[2]==taskNr,3],0.25),
                median(TimingsSTZ[TimingsSTZ[2]==taskNr,3]),
                mean(TimingsSTZ[TimingsSTZ[2]==taskNr,3]),
                quantile(TimingsSTZ[TimingsSTZ[2]==taskNr,3],0.75),
                max(TimingsSTZ[TimingsSTZ[2]==taskNr,3]),
                sd(TimingsSTZ[TimingsSTZ[2]==taskNr,3])))
}
write.csv(sumstats_STZ,file="sumstats_STZ.csv")

# ================================================
# VISTYPE KNAVE --- Summary Statistics ---
# ================================================
```

```
TimingsKNAVE <- aggregatedData[aggregatedData[6]=="KNAVE",1:8]

 sumstats_KNAVE <- data.frame(Task=1, MIN=min(TimingsKNAVE[TimingsKNAVE[2]==1,3]),
         Q1=quantile(TimingsKNAVE[TimingsKNAVE[2]==1,3],0.25),
        MEDIAN=median(TimingsKNAVE[TimingsKNAVE[2]==1,3]),
          MEAN=mean(TimingsKNAVE[TimingsKNAVE[2]==1,3]),
        Q3=quantile(TimingsKNAVE[TimingsKNAVE[2]==1,3],0.75),
        MAX=max(TimingsKNAVE[TimingsKNAVE[2]==1,3]),
        SD=sd(TimingsKNAVE[TimingsKNAVE[2]==1,3]))
 for (taskNr in 2:12) {
         sumstats_KNAVE <- rbind(sumstats_KNAVE, c(taskNr,
                 min(TimingsKNAVE[TimingsKNAVE[2]==taskNr,3]),
                quantile(TimingsKNAVE[TimingsKNAVE[2]==taskNr,3],0.25),
                median(TimingsKNAVE[TimingsKNAVE[2]==taskNr,3]),
                mean(TimingsKNAVE[TimingsKNAVE[2]==taskNr,3]),
                quantile(TimingsKNAVE[TimingsKNAVE[2]==taskNr,3],0.75),
                max(TimingsKNAVE[TimingsKNAVE[2]==taskNr,3]),
                sd(TimingsKNAVE[TimingsKNAVE[2]==taskNr,3])))
 }
 write.csv(sumstats_KNAVE,file="sumstats_KNAVE.csv")


# ================================================
# Two-way analysis of variance (ANOVA) - Interaction between dataset and vistype
# ================================================
aov.T = aov(duration_s ~as.factor(dataset)*as.factor(visType), aggregatedData)
summary(aov.T)
print(model.tables(aov.T,"means"),digits=3)

# ================================================
# Paired t-test between datasets
# ================================================

TimingsDataset1 <- aggregatedData[aggregatedData[7]==1,1:8]
TimingsDataset2 <- aggregatedData[aggregatedData[7]==2,1:8]
var.test(log(TimingsDataset2$duration_s), log(TimingsDataset1$duration_s))

shapiro.test(log(TimingsDataset1 $duration_s))
shapiro.test(log(TimingsDataset2 $duration_s))
t.test(log(subset(aggregatedData, dataset==1)$duration_s),
        log(subset(aggregatedData, dataset==2)$duration_s), paired=T)

# ================================================
# Two-way analysis of variance (ANOVA) - Interaction between vis-order and vistype
# ================================================
aov.T = aov(duration_s ~as.factor(order)*as.factor(visType), aggregatedData)
summary(aov.T)
print(model.tables(aov.T,"means"),digits=3)

# ================================================
# Two-way analysis of variance (ANOVA) - Interaction between inputtype and vistype
# ================================================
aov.T = aov(duration_s ~as.factor(inputType)*as.factor(visType), aggregatedData)
summary(aov.T)
print(model.tables(aov.T,"means"),digits=3)

# ================================================
# RM analysis of variance (ANOVA) - Influence of vistype
# ================================================

aov <- aov(duration_s ~ as.factor(visType) + Error(factor(participantId)/factor(visType)),
        aggregatedData)
```

```
summary(aov)

# ================================================
# Split up VisType and Order
# ================================================

TimingsSTZ_Order_1 <- aggregatedData[aggregatedData[6]=="STZ" & aggregatedData[8]=="1",1:8]
TimingsSTZ_Order_2 <- aggregatedData[aggregatedData[6]=="STZ" & aggregatedData[8]=="2",1:8]
TimingsKNAVE_Order_1 <- aggregatedData[aggregatedData[6]=="KNAVE" & aggregatedData[8]=="1",1:8]
TimingsKNAVE_Order_2 <- aggregatedData[aggregatedData[6]=="KNAVE" & aggregatedData[8]=="2",1:8]

# ================================================
# COMPARISON BOXPLOT - OVERALL
# ================================================
pdf("timingBoxplots-comparison_overall.pdf",width=11.7,height=7)
boxplot(duration_s ~ taskType, data = TimingsSTZ, at = 1:12 - 0.2, boxwex = 0.15, col = colors()[c(50)],
        names=c("","","","","","","","","","","",""), main="Task completion times (Overall)",
        xlab="Tasks", ylab="time (s)", axes = FALSE, ylim=c(0,90))
boxplot(duration_s ~ taskType, data = TimingsKNAVE, at = 1:12, boxwex = 0.15, col = colors()[c(79)],
        add = TRUE)

legend(0.2, 75, c("STZ", "KNAVE"),
       fill = colors()[c(50,79)], bty="n")
dev.off()

# ================================================
# COMPARISON BOXPLOT - DATASET
# ================================================


pdf("timingBoxplots-comparison_dataset.pdf",width=5,height=7)
boxplot(log(duration_s) ~ dataset, data = aggregatedData, col = colors()[c(50,79)],
        main="Task completion times for dataset", xlab="Dataset", ylab="log (time)" )
dev.off()

# ================================================
# COMPARISON BOXPLOT - ROUNDS
# ================================================

pdf("timingBoxplots-comparison_order.pdf",width=5,height=7)
par(mfrow=c(1,2),mar=c(5,5,1,2))
boxplot((duration_s) ~ order, data = TimingsSTZ, col = colors()[c(50,79)], main="STZ",
        xlab="Experiment round", ylab="time (s)", ylim=c(0,65) )
boxplot((duration_s) ~ order, data = TimingsKNAVE, col = colors()[c(50,79)], main="KNAVE",
        xlab="Experiment round", ylab="time (s)" , ylim=c(0,65) )
dev.off()


# ================================================
# COMPARISON BOXPLOT - ROUND 1
# ================================================
pdf("timingBoxplots-comparison_round_1.pdf",width=11.7,height=7)
boxplot(duration_s ~ taskType, data = TimingsSTZ_Order_1, at = 1:12 - 0.2, boxwex = 0.15,
        col = colors()[c(50)], names=c("","","","","","","","","","","",""),
        main="Task completion times (Round 1)", xlab="Tasks", ylab="time (s)", axes = FALSE,
        ylim=c(0,90))
boxplot(duration_s ~ taskType, data = TimingsKNAVE_Order_1, at = 1:12, boxwex = 0.15,
        col = colors()[c(79)], add = TRUE)

legend(0.2, 75, c("STZ", "KNAVE"),
       fill = colors()[c(50,79)], bty="n")
dev.off()
```

```
# ================================================
# COMPARISON BOXPLOT - ROUND 2
# ================================================
pdf("timingBoxplots-comparison_round_2.pdf",width=11.7,height=7)
boxplot(duration_s ~ taskType, data = TimingsSTZ_Order_2, at = 1:12 - 0.2, boxwex = 0.15,
        col = colors()[c(50)], names=c("","","","","","","","","","","",""),
        main="Task completion times (Round 2)", xlab="Tasks", ylab="time (s)", axes = FALSE,
        ylim=c(0,90))
boxplot(duration_s ~ taskType, data = TimingsKNAVE_Order_2, at = 1:12, boxwex = 0.15,
        col = colors()[c(79)], add = TRUE)

legend(0.2, 75, c("STZ", "KNAVE"),
       fill = colors()[c(50,79)], bty="n")
dev.off()

# ================================================
# NORMAL DISTRIBUTION - ROUND 1 STZ
# ================================================

 # --- Normal Distribution Tests - Round 1 --
 shapirowilk_STZ <- data.frame(Task=1,
        SW=shapiro.test(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]==1,3])$p.value)
 for (taskNr in 2:12) {
        sw <- shapiro.test(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]==taskNr,3])$p.value
        shapirowilk_STZ <- rbind(shapirowilk_STZ, c(taskNr,sw))
 }
 write.csv(shapirowilk_STZ,file="shapirowilk_STZ_round_1.csv")

  # --- Log-Normal Distribution Tests Round 1 --
 shapirowilk_STZ <- data.frame(Task=1,
        SW=shapiro.test(log(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]==1,3]))$p.value)
 for (taskNr in 2:12) {
        sw <- shapiro.test(log(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]==taskNr,3]))$p.value
        shapirowilk_STZ <- rbind(shapirowilk_STZ, c(taskNr,sw))
 }
 write.csv(shapirowilk_STZ,file="shapirowilk_STZ_log_round_1.csv")


# ================================================
# NORMAL DISTRIBUTION - ROUND 1 KNAVE
# ================================================

 # --- Normal Distribution Tests - Round 1 --
 shapirowilk_KNAVE <- data.frame(Task=1,
        SW=shapiro.test(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]==1,3])$p.value)
 for (taskNr in 2:12) {
        sw <- shapiro.test(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]==taskNr,3])$p.value
        shapirowilk_KNAVE <- rbind(shapirowilk_KNAVE, c(taskNr,sw))
 }
 write.csv(shapirowilk_KNAVE,file="shapirowilk_KNAVE_round_1.csv")

  # --- Log-Normal Distribution Tests Round 1 --
 shapirowilk_KNAVE <- data.frame(Task=1,
        SW=shapiro.test(log(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]==1,3]))$p.value)
 for (taskNr in 2:12) {
        sw <- shapiro.test(log(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]==taskNr,3]))$p.value
        shapirowilk_KNAVE <- rbind(shapirowilk_KNAVE, c(taskNr,sw))
 }
 write.csv(shapirowilk_KNAVE,file="shapirowilk_KNAVE_log_round_1.csv")

# ================================================
```

```
# NORMAL DISTRIBUTION - ROUND 2 STZ
# ================================================

 # --- Normal Distribution Tests - Round 2 --
 shapirowilk_STZ <- data.frame(Task=1,
         SW=shapiro.test(TimingsSTZ_Order_2[TimingsSTZ_Order_2[2]==1,3])$p.value)
 for (taskNr in 2:12) {
         sw <- shapiro.test(TimingsSTZ_Order_2[TimingsSTZ_Order_2[2]==taskNr,3])$p.value
         shapirowilk_STZ <- rbind(shapirowilk_STZ, c(taskNr,sw))
 }
 write.csv(shapirowilk_STZ,file="shapirowilk_STZ_round_2.csv")

  # --- Log-Normal Distribution Tests Round 2 --
 shapirowilk_STZ <- data.frame(Task=1,
         SW=shapiro.test(log(TimingsSTZ_Order_2[TimingsSTZ_Order_2[2]==1,3]))$p.value)
 for (taskNr in 2:12) {
         sw <- shapiro.test(log(TimingsSTZ_Order_2[TimingsSTZ_Order_2[2]==taskNr,3]))$p.value
         shapirowilk_STZ <- rbind(shapirowilk_STZ, c(taskNr,sw))
 }
 write.csv(shapirowilk_STZ,file="shapirowilk_STZ_log_round_2.csv")


# ================================================
# NORMAL DISTRIBUTION - ROUND 2 KNAVE
# ================================================

 # --- Normal Distribution Tests - Round 2 --
 shapirowilk_KNAVE <- data.frame(Task=1,
         SW=shapiro.test(TimingsKNAVE_Order_2[TimingsKNAVE_Order_1[2]==1,3])$p.value)
 for (taskNr in 2:12) {
         sw <- shapiro.test(TimingsKNAVE_Order_2[TimingsKNAVE_Order_2[2]==taskNr,3])$p.value
         shapirowilk_KNAVE <- rbind(shapirowilk_KNAVE, c(taskNr,sw))
 }
 write.csv(shapirowilk_KNAVE,file="shapirowilk_KNAVE_round_2.csv")

  # --- Log-Normal Distribution Tests Round 1 --
 shapirowilk_KNAVE <- data.frame(Task=1,
         SW=shapiro.test(log(TimingsKNAVE_Order_2[TimingsKNAVE_Order_2[2]==1,3]))$p.value)
 for (taskNr in 2:12) {
         sw <- shapiro.test(log(TimingsKNAVE_Order_2[TimingsKNAVE_Order_2[2]==taskNr,3]))$p.value
         shapirowilk_KNAVE <- rbind(shapirowilk_KNAVE, c(taskNr,sw))
 }
 write.csv(shapirowilk_KNAVE,file="shapirowilk_KNAVE_log_round_2.csv")

# ================================================
# TESTING FOR EQUAL VARIANCE (F-TEST) - ROUND 1
# ================================================

 equal_variance_round_1 <- data.frame(Task=1,
         SW=var.test(log(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]== "1", "duration_s"]),
        log(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]== "1", "duration_s"]))$p.value)
 for (taskNr in 2:12) {
         sw <-var.test(log(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]== taskNr, "duration_s"]),
                log(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]== taskNr, "duration_s"]))$p.value
         equal_variance_round_1 <- rbind(equal_variance_round_1, c(taskNr,sw))
 }
 write.csv(equal_variance_round_1,file="log_equal_variance_round_1.csv")

# ================================================
# TESTING FOR EQUAL VARIANCE (F-TEST) - ROUND 2
# ================================================
```

```
 equal_variance_round_2 <- data.frame(Task=1,
         SW=var.test(log(TimingsSTZ_Order_2[TimingsSTZ_Order_2[2]== "1", "duration_s"]),
        log(TimingsKNAVE_Order_2[TimingsKNAVE_Order_2[2]== "1", "duration_s"]))$p.value)
 for (taskNr in 2:12) {
         sw <-var.test(log(TimingsSTZ_Order_2[TimingsSTZ_Order_2[2]== taskNr, "duration_s"]),
                log(TimingsKNAVE_Order_2[TimingsKNAVE_Order_2[2]== taskNr, "duration_s"]))$p.value
         equal_variance_round_2 <- rbind(equal_variance_round_2, c(taskNr,sw))
 }
 write.csv(equal_variance_round_2,file="log_equal_variance_round_2.csv")


# ================================================
# T-TEST SINGLE TASKS - ROUND 1
# ================================================

t_test_round_1 <- data.frame(Task=1,SW=t.test(
        log(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]== "1", "duration_s"]),
        log(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]== "1", "duration_s"]))$p.value)
 for (taskNr in 2:12) {
         sw <-t.test(log(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]== taskNr, "duration_s"]),
                log(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]== taskNr, "duration_s"]))$p.value
         t_test_round_1 <- rbind(t_test_round_1, c(taskNr,sw))
 }
 write.csv(t_test_round_1,file="t_test_round_1_log.csv")

 # ================================================
# T-TEST SINGLE TASKS - ROUND 2
# ================================================

t_test_round_2 <- data.frame(Task=1,SW=t.test(
        log(TimingsSTZ_Order_2[TimingsSTZ_Order_1[2]== "1", "duration_s"]),
        log(TimingsKNAVE_Order_2[TimingsKNAVE_Order_1[2]== "1", "duration_s"]))$p.value)
 for (taskNr in 2:12) {
         sw <-t.test(log(TimingsSTZ_Order_2[TimingsSTZ_Order_2[2]== taskNr, "duration_s"]),
                log(TimingsKNAVE_Order_2[TimingsKNAVE_Order_2[2]== taskNr, "duration_s"]))$p.value
         t_test_round_2 <- rbind(t_test_round_2, c(taskNr,sw))
 }
 write.csv(t_test_round_2,file="t_test_round_2_log.csv")

 # ================================================
# T-TEST SINGLE TASKS - ROUND 1 - SINGLE SIDED
# ================================================

t_test_round_1 <- data.frame(Task=1,SW=t.test(
        log(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]== "1", "duration_s"]),
        log(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]== "1", "duration_s"]),
        alternative = "less")$p.value)
 for (taskNr in 2:12) {
         sw <-t.test(log(TimingsSTZ_Order_1[TimingsSTZ_Order_1[2]== taskNr, "duration_s"]),
                log(TimingsKNAVE_Order_1[TimingsKNAVE_Order_1[2]== taskNr, "duration_s"]),
                alternative = "less")$p.value
         t_test_round_1 <- rbind(t_test_round_1, c(taskNr,sw))
 }
 write.csv(t_test_round_1,file="t_test_round_1_log_ss.csv")

 # ================================================
# T-TEST SINGLE TASKS - ROUND 2 - SINGLE SIDED
# ================================================

t_test_round_2 <- data.frame(Task=1,SW=t.test(
        log(TimingsSTZ_Order_2[TimingsSTZ_Order_1[2]== "1", "duration_s"]),
        log(TimingsKNAVE_Order_2[TimingsKNAVE_Order_1[2]== "1", "duration_s"]),
```

```
        alternative = "less")$p.value)
 for (taskNr in 2:12) {
         sw <-t.test(log(TimingsSTZ_Order_2[TimingsSTZ_Order_2[2]== taskNr, "duration_s"]),
                log(TimingsKNAVE_Order_2[TimingsKNAVE_Order_2[2]== taskNr, "duration_s"]),
                alternative = "less")$p.value
         t_test_round_2 <- rbind(t_test_round_2, c(taskNr,sw))
 }
 write.csv(t_test_round_2,file="t_test_round_2_log_ss.csv")
```

The following script uses wilcox tests to find out the influence of the dataset and experiment round on errors. Following, the task error rates are aggregated for each task per interface and experiment round and wilcox tests are performed for each task to find out significant differences for visualization type on errors for each round.

```
# Import data
visData <- read.table("raw_data.csv", header=TRUE, sep = ";")
visData <- visData[2:9]
numberOfTasks <- 12
numberOfSubjects <- 20
numberOfDatasets <- 2

# ================================================
# Calculate error counts and error rates for tasks per interface
# ================================================
interfaceChars <- c("STZ","KNAVE")


ErrorsTasks <- data.frame(task=numeric(0), interface=character(0), correct=numeric(0),
        correctness_rate=numeric(0), sd=numeric(0), rate_sd=numeric(0))
for(taskNr in 1:numberOfTasks) {
        for(interfaceNr in 1:length(interfaceChars)) {
                successCount <- 0
                for(subjectNr in 1:numberOfSubjects) {

                        success <- visData[visData[1]==subjectNr &
                                visData[6]==interfaceChars[interfaceNr] & visData[2]==taskNr,4]

                        if (length(success)>1){
                                        success_sum <- success[1] + success[2]
                                        successCount <- successCount + success_sum

                                }

                }
                correctnessRate <- successCount/(numberOfSubjects*2)
                correctnessCountSD <- sqrt((numberOfSubjects*2)*correctnessRate*(1-correctnessRate))
                correctnessRateSD <- correctnessCountSD/(numberOfSubjects*2)

                ErrorsTasks <- rbind(ErrorsTasks, data.frame(task=taskNr,
                        interface=interfaceChars[interfaceNr], correct=successCount,
                        correctness_rate=correctnessRate, sd=correctnessCountSD,
                        rate_sd=correctnessRateSD))
        }
}


# ================================================
# wilcox on the influence of datasets on errors
# ================================================
interfaceChars <- c("STZ","KNAVE")
```

```
results <- data.frame(task=numeric(0), interface=character(0), dataset=numeric(0),
        success_count=numeric(0), success_rate=numeric(0))

for(taskNr in 1:12) {
        TaskError <- visData[visData[2]==taskNr, 1:8]
        error_rates <- numeric(0)
        for (datasetNr in 1:2) {
                TaskDatasetError <- TaskError[TaskError[7]==datasetNr, 1:8]
                for (interface in 1:length(interfaceChars)) {
                        TaskDatasetInterfaceError <-
                                TaskDatasetError[TaskDatasetError[6]==
                                interfaceChars[interface], 1:8]

                        count <- length(TaskDatasetInterfaceError$participantId)
                        correctnessSum <- sum(TaskDatasetInterfaceError$correctness)
                        correctness_rate <- correctnessSum/count
                        results <- rbind(results, data.frame(task=taskNr,
                                interface=interfaceChars[interface],
                                dataset=datasetNr,
                                success_count=correctnessSum,
                                success_rate=correctness_rate))
                }
        }
}

wilcox.test(subset(results, dataset == 1)$success_rate, subset(results, dataset == 2)$success_rate )

# effect size
library(coin)
g <- factor(c(rep("GroupA", length(subset(results, dataset == 1)$success_rate)),
        rep("GroupB", length(subset(results, dataset == 2)$success_rate))))
v <- c(subset(results, dataset == 1)$success_rate, subset(results, dataset == 2)$success_rate)
wilcox_test(v ~ g, distribution="exact")
r <- rank(v)
data <- data.frame(g, r)
lapply((split(data, data$g)), mean)
0.3654/sqrt(48)

# ================================================
# wilcox on the influence of order on errors
# ================================================
interfaceChars <- c("STZ","KNAVE")
results_order <- data.frame(task=numeric(0), interface=character(0), order=numeric(0),
        success_count=numeric(0), success_rate=numeric(0), rate_sd=numeric(0))

for(taskNr in 1:12) {
        TaskError <- visData[visData[2]==taskNr, 1:8]
        error_rates <- numeric(0)
        for (orderNr in 1:2) {
                TaskOrderError <- TaskError[TaskError[8]== orderNr, 1:8]
                for (interface in 1:length(interfaceChars)) {
                        TaskOrderInterfaceError <-
                                TaskOrderError[TaskOrderError[6]==interfaceChars[interface],
                                1:8]

                        count <- length(TaskOrderInterfaceError$participantId)
                        correctnessSum <- sum(TaskOrderInterfaceError$correctness)
                        correctness_rate <- correctnessSum/count
                        correctnessCountSD =
                                sqrt (count* correctness_rate * (1-correctness_rate))
                        correctnessRateSD = correctnessCountSD/count
```

```
                                results_order <- rbind(results_order, data.frame(task=taskNr,
                                        interface=interfaceChars[interface], order= orderNr,
                                        success_count=correctnessSum, success_rate=correctness_rate,
                                        rate_sd= correctnessRateSD))
                }
        }
}

wilcox.test(subset(results_order, order == 1)$success_rate, subset(results_order, order == 2)$success_rate)

# effect size
library(coin)
g <- factor(c(rep("GroupA", length(subset(results_order, order == 1)$success_rate)),
        rep("GroupB", length(subset(results_order, order == 2)$success_rate))))
v <- c(subset(results_order, order == 1)$success_rate, subset(results_order, order == 2)$success_rate)
wilcox_test(v ~ g, distribution="exact")
r <- rank(v)
data <- data.frame(g, r)
lapply((split(data, data$g)), mean)
2.1937/sqrt(48)

# ================================================
# Bar plot of success rates and standard deviation (Binomial distribution) Round 1
# ================================================
pdf("success-rates-tasks_barplot_round_1.pdf",width=11,height=7)

correctness_rates <- c(subset(results_order, order == 1)$success_rate)
correctness_rates <- correctness_rates * 100
correctness_rate_table <- matrix(correctness_rates, nrow=numberOfTasks, ncol=length(interfaceChars),
        byrow=TRUE, dimnames=list(1:numberOfTasks, interfaceChars))
correctnessRateSDs <- c(subset(results_order, order == 1)$rate_sd)
correctnessRateSDs <- correctnessRateSDs*100
correctnessRateSDs_table <- matrix(correctnessRateSDs, nrow=numberOfTasks, ncol=length(interfaceChars),
        byrow=TRUE)

par(xpd = NA)
xpos=barplot(t(correctness_rate_table), beside=TRUE,ylim=c(0,100), col=colors()[c(50,79)],
        xlab="Task", ylab="Success Rate (%) Round 1", axis.lty=1)
segments(xpos, correctness_rates-correctnessRateSDs, xpos, correctness_rates+correctnessRateSDs)
legend(14, 110.1, c("STZ", "KNAVE"), fill = colors()[c(50,79,91)], bty="n", cex=0.8, hor=TRUE)

dev.off()

# ================================================
# Bar plot of success rates and standard deviation (Binomial distribution) Round 2
# ================================================
pdf("success-rates-tasks_barplot_round_2.pdf",width=11,height=7)

correctness_rates <- c(subset(results_order, order == 2)$success_rate)
correctness_rates <- correctness_rates * 100
correctness_rate_table <- matrix(correctness_rates, nrow=numberOfTasks, ncol=length(interfaceChars),
        byrow=TRUE, dimnames=list(1:numberOfTasks, interfaceChars))
correctnessRateSDs <- c(subset(results_order, order == 2)$rate_sd)
correctnessRateSDs <- correctnessRateSDs*100
correctnessRateSDs_table <- matrix(correctnessRateSDs, nrow=numberOfTasks, ncol=length(interfaceChars),
        byrow=TRUE)

par(xpd = NA)
xpos=barplot(t(correctness_rate_table), beside=TRUE,ylim=c(0,100), col=colors()[c(50,79)], xlab="Task",
        ylab="Success Rate (%) Round 2", axis.lty=1)
segments(xpos, correctness_rates-correctnessRateSDs, xpos, correctness_rates+correctnessRateSDs)
```

```
legend(14, 110.1, c("STZ", "KNAVE"), fill = colors()[c(50,79,91)], bty="n", cex=0.8, hor=TRUE)

dev.off()

# ================================================
# Analysis on individual task level - Round 1
# ================================================

groupSTZ <- subset(visData, order == 1 & taskType == 1 & visType == "STZ")$correctness
groupKNAVE <- subset(visData, order == 1 & taskType == 1 & visType == "KNAVE")$correctness

g <- factor(c(rep("groupSTZ", length(groupSTZ)), rep("groupKNAVE", length(groupKNAVE))))
v <- c(groupSTZ, groupKNAVE)

wt <- wilcox_test(v ~ g, distribution="exact")

wilcox_individual <- data.frame(Task=1, pValue=wilcox.test(groupSTZ,groupKNAVE)$p.value)

wilcox_ties_individual <- data.frame(Task=1, pValue=pvalue(wt))

for(taskNr in 2:12) {
        groupSTZ <- subset(visData, order == 1 & taskType == taskNr & visType == "STZ")$correctness
        groupKNAVE <- subset(visData,
                order == 1 & taskType == taskNr & visType == "KNAVE")$correctness

        g <- factor(c(rep("groupSTZ", length(groupSTZ)), rep("groupKNAVE", length(groupKNAVE))))
        v <- c(groupSTZ, groupKNAVE)

        wt <- wilcox_test(v ~ g, distribution="exact")

        wilcox_individual <- rbind(wilcox_individual,
                c(taskNr,wilcox.test(groupSTZ,groupKNAVE)$p.value))

        wilcox_ties_individual <-  rbind(wilcox_ties_individual, c(taskNr,pvalue(wt)))
}

write.csv(wilcox_individual,file="error_individual_wilcox_round_1.csv")
write.csv(wilcox_ties_individual,file="error_individual_wilcox_ties_round_1.csv")

# ================================================
# Analysis on individual task level - Round 2
# ================================================

groupSTZ <- subset(visData, order == 2 & taskType == 1 & visType == "STZ")$correctness
groupKNAVE <- subset(visData, order == 2 & taskType == 1 & visType == "KNAVE")$correctness

g <- factor(c(rep("groupSTZ", length(groupSTZ)), rep("groupKNAVE", length(groupKNAVE))))
v <- c(groupSTZ, groupKNAVE)

wt <- wilcox_test(v ~ g, distribution="exact")

wilcox_individual <- data.frame(Task=1, pValue=wilcox.test(groupSTZ,groupKNAVE)$p.value)

wilcox_ties_individual <- data.frame(Task=1, pValue=pvalue(wt))

for(taskNr in 2:12) {
        groupSTZ <- subset(visData,
                order == 2 & taskType == taskNr & visType == "STZ")$correctness
        groupKNAVE <- subset(visData,
                order == 2 & taskType == taskNr & visType == "KNAVE")$correctness

        if (!identical(groupSTZ, groupKNAVE)){
```

```
        g <- factor(c(rep("groupSTZ", length(groupSTZ)), rep("groupKNAVE", length(groupKNAVE))))
        v <- c(groupSTZ, groupKNAVE)

        wt <- wilcox_test(v ~ g, distribution="exact")

        wilcox_individual <- rbind(wilcox_individual, c(taskNr,wilcox.test(groupSTZ,groupKNAVE)$p.value))

        wilcox_ties_individual <-  rbind(wilcox_ties_individual, c(taskNr,pvalue(wt)))
        }
}

write.csv(wilcox_individual,file="error_individual_wilcox_round_2.csv")
write.csv(wilcox_ties_individual,file="error_individual_wilcox_ties_round_2.csv")
```

APPENDIX E

R Scripts for Hypotheses Testing

The following script aggregates the task completion times by task sets defined in the hypotheses (cf. Tables 5.1 & 5.2), experiment round and visualization type. Normal and log-normal distribution tests are performed and also equal variance is tested. Based on these results, one sided t-tests are performed on these data to find out significant differences.

```
# ================================================
# HYPOTHESES - TASK COMPLETION TIME
# ================================================
# Import data

# Import data
aggregatedData <- read.table("raw_data.csv", header=TRUE, sep = ";")
aggregatedData <- aggregatedData[2:9]

numberOfSubjects <- 20
numberOfDatasets <- 2

# ================================================
# Round 1
# ================================================

# ================================================
# Calculate timings for hypotheses -
# ================================================


hypothesesNames <- c("H1.1","H1.2","H2.1", "H2.2")
taskNrH1 <- c(1,2,3)
taskNrH2 <- c(4,5,6)
taskNrH3 <- c(7,8,9)
taskNrH4 <- c(10,11,12)
taskNrHypotheses <- list(taskNrH1, taskNrH2, taskNrH3, taskNrH4)
interfacesH1 <- c("STZ","KNAVE")
interfacesH2 <- c("STZ","KNAVE")
interfacesH3 <- c("STZ","KNAVE")
interfacesH4 <- c("STZ","KNAVE")
interfacesHypotheses <- list(interfacesH1, interfacesH2, interfacesH3, interfacesH4)
```

```
TimingsHypotheses <- data.frame(subject=numeric(0), hypothesis=character(0),
        interface=character(0), time_s=numeric(0))
for (subjectNr in 1:numberOfSubjects) {
        for(hypothesesNr in 1:length(hypothesesNames)) {
                interfacesForHy <- interfacesHypotheses[[hypothesesNr]]
                numberOfTasksForHy <- length(taskNrHypotheses[[hypothesesNr]])
                tasksForHy <- taskNrHypotheses[[hypothesesNr]]
                for(interfaceNr in 1:length(interfacesForHy)) {
                        time_sum <- 0
                        for(i in 1:numberOfTasksForHy) {

                                time <- log(aggregatedData[aggregatedData[1]==subjectNr &
                                        aggregatedData[6]==interfacesForHy[interfaceNr] &
                                        aggregatedData[2]==tasksForHy[i] &
                                        aggregatedData[8]==1,3])
                                # uncomment to calculate non log times
                                #time <- aggregatedData[aggregatedData[1]==subjectNr &
                                        aggregatedData[6]==interfacesForHy[interfaceNr] &
                                        aggregatedData[2]==tasksForHy[i] &
                                        aggregatedData[8]==1,3]
                                if (length(time)>1){
                                        time_sum <- time_sum + time[1] + time[2]
                                }
                        }
                        if (time_sum>0) {
                                TimingsHypotheses <- rbind(TimingsHypotheses,
                                        data.frame(subject=subjectNr,
                                        hypothesis=hypothesesNames[hypothesesNr],
                                        interface=interfacesForHy[interfaceNr],
                                        time_s=time_sum))
                        }
                }
        }
}


# ====================================================
# Summary Statistics
# ====================================================
sumstats_hypotheses_timings <- data.frame(hypothesis=character(0), interface=character(0),
        MIN=numeric(0), Q1=numeric(0),
        MEDIAN=numeric(0), MEAN=numeric(0),
        Q3=numeric(0), MAX=numeric(0), SD=numeric(0))

for(hypothesesNr in 1:length(hypothesesNames)) {
        interfacesForHy <- interfacesHypotheses[[hypothesesNr]]
        for(interfaceNr in 1:length(interfacesForHy)) {
                times <- TimingsHypotheses[TimingsHypotheses[3]==interfacesForHy[interfaceNr] &
                        TimingsHypotheses[2]==hypothesesNames[hypothesesNr],4]
                sumstats_hypotheses_timings <- rbind(sumstats_hypotheses_timings,
                        data.frame(hypothesis=hypothesesNames[hypothesesNr],
                        interface=interfacesForHy[interfaceNr],
                        MIN=min(times), Q1=quantile(times,0.25),
                        MEDIAN=median(times), MEAN=mean(times),
                        Q3=quantile(times,0.75), MAX=max(times),
                        SD=sd(times)))
        }
}

write.csv(sumstats_hypotheses_timings,file="sumstats_hypotheses_timings_round_1.csv")
```

```
# ================================================
# Boxplot Task Completion Times
# ================================================
pdf("timingBoxplots-hypotheses-row_round_1.pdf",width=10,height=4)
par(mfrow=c(1,4),mar=c(7,5,2,2),xpd=NA)
for (hypothesisNr in 1:4) {
        hypothesisString <- hypothesesNames[hypothesisNr]
        hypothesisDescription <- ""
        if (hypothesisNr == 1 | hypothesisNr == 3)
                hypothesisDescription <- paste(hypothesisString,"Lookup Tasks",sep=" ")
        if (hypothesisNr == 2 | hypothesisNr == 4)
                hypothesisDescription <- paste(hypothesisString,"Comparsion Tasks",sep=" ")
        boxplot(time_s ~ interface, data = TimingsHypotheses,
                subset = hypothesis==hypothesisString, main= hypothesisDescription, ylab="time (s)",
                col = colors()[c(50,79,91)], boxwex=0.5, bty="n", cex.lab=1.4, cex.axis=1.0,
                cex.main=1.6, names = c("STZ", "KNAVE"), whisklty="solid")
}
dev.off()

# ================================================
# Test for normal distribution
# ================================================
shapirowilk_H_Round_1 <- data.frame(Hypothesis="Hx",Interface="X",SW=0.0)
interfaceChars <- c("STZ","KNAVE")
for (hypothesisNr in 1:4) {
        #hypothesisString <- paste("H",hypothesisNr,sep="")
        hypothesisString <- hypothesesNames[hypothesisNr]
        TimingsH <- TimingsHypotheses[TimingsHypotheses[2]==hypothesisString,1:4]
        for(interfaceNr in 1:2) {
                times <- TimingsH[TimingsH[3]==interfaceChars[interfaceNr],4]
                if(length(times)>0) {
                        sw <- shapiro.test(times)$p.value
                        shapirowilk_H_Round_1 <- rbind(shapirowilk_H_Round_1,
                                data.frame(Hypothesis=hypothesisString,
                                Interface=interfaceChars[interfaceNr],SW=sw))
                }
        }
}
write.csv(shapirowilk_H_Round_1,file="shapirowilk_hypotheses_round_1.csv")

# ================================================
# Testing for equal variance (F-Test)
# ================================================

equal_var <- data.frame(Hypothesis="Hx",SW=0.0)

# H1.1
sw <- var.test(TimingsHypotheses[TimingsHypotheses[2]=="H1.1" &
        TimingsHypotheses[3]=="STZ", "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H1.1" &
        TimingsHypotheses[3]=="KNAVE", "time_s"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H1.1",SW=sw))

# H1.2
sw <- var.test(TimingsHypotheses[TimingsHypotheses[2]=="H1.2" &
        TimingsHypotheses[3]=="STZ", "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H1.2" &
        TimingsHypotheses[3]=="KNAVE", "time_s"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H1.2",SW=sw))
```

```
# H2.1
sw <- var.test(TimingsHypotheses[TimingsHypotheses[2]=="H2.1" &
        TimingsHypotheses[3]=="STZ", "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H2.1" &
        TimingsHypotheses[3]=="KNAVE", "time_s"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H2.1",SW=sw))

# H2.2
sw <- var.test(TimingsHypotheses[TimingsHypotheses[2]=="H2.2" &
        TimingsHypotheses[3]=="STZ", "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H2.2" &
        TimingsHypotheses[3]=="KNAVE", "time_s"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H2.2",SW=sw))

write.csv(equal_var,file="hypothesis_equal_variance_round_1.csv")

# ================================================
# T-Test (not paired, one sided)
# ================================================

library(MBESS)

t_test <- data.frame(Hypothesis="Hx",SW=0.0, D=0.0)

# H1.1
sw <- t.test(TimingsHypotheses[TimingsHypotheses[2]=="H1.1" &
        TimingsHypotheses[3]=="STZ", "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H1.1" &
        TimingsHypotheses[3]=="KNAVE", "time_s"], paired = F, alternative="less")$p.value

# effect size
d <- abs(smd(TimingsHypotheses[TimingsHypotheses[2]=="H1.1" & TimingsHypotheses[3]=="STZ",
        "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H1.1" & TimingsHypotheses[3]=="KNAVE",
        "time_s"]))

t_test <- rbind(t_test, data.frame(Hypothesis="H1.1",SW=sw, D=d))

# H1.2
sw <- t.test(TimingsHypotheses[TimingsHypotheses[2]=="H1.2" &
        TimingsHypotheses[3]=="STZ", "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H1.2" &
        TimingsHypotheses[3]=="KNAVE", "time_s"], paired = F, alternative="less")$p.value

# effect size
d <- abs(smd(TimingsHypotheses[TimingsHypotheses[2]=="H1.1" & TimingsHypotheses[3]=="STZ",
        "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H1.2"
        & TimingsHypotheses[3]=="KNAVE",
        "time_s"]))


t_test <- rbind(t_test, data.frame(Hypothesis="H1.2",SW=sw, D=d))

# H2.1
sw <- t.test(TimingsHypotheses[TimingsHypotheses[2]=="H2.1" &
        TimingsHypotheses[3]=="STZ", "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H2.1" &
        TimingsHypotheses[3]=="KNAVE", "time_s"], paired = F, alternative="less")$p.value
```

```
# effect size
d <- abs(smd(TimingsHypotheses[TimingsHypotheses[2]=="H2.1" & TimingsHypotheses[3]=="STZ",
        "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H2.1"
        & TimingsHypotheses[3]=="KNAVE", "time_s"]))

t_test <- rbind(t_test, data.frame(Hypothesis="H2.1",SW=sw, D=d))

# H2.2
sw <- t.test(TimingsHypotheses[TimingsHypotheses[2]=="H2.2" &
        TimingsHypotheses[3]=="STZ", "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H2.2" &
        TimingsHypotheses[3]=="KNAVE", "time_s"], paired = F, alternative="less")$p.value

d <- abs(smd(TimingsHypotheses[TimingsHypotheses[2]=="H2.2"
        & TimingsHypotheses[3]=="STZ", "time_s"],
        TimingsHypotheses[TimingsHypotheses[2]=="H2.2"
        & TimingsHypotheses[3]=="KNAVE", "time_s"]))

t_test <- rbind(t_test, data.frame(Hypothesis="H2.2",SW=sw, D=d))

write.csv(t_test,file="hypothesis_t_test_round_1.csv")

# ================================================
# Round 2
# ================================================

# ================================================
# Calculate timings for hypotheses -
# ================================================


TimingsHypothesesRound2 <- data.frame(subject=numeric(0), hypothesis=character(0),
        interface=character(0), time_s=numeric(0))
for (subjectNr in 1:numberOfSubjects) {
        for(hypothesesNr in 1:length(hypothesesNames)) {
                interfacesForHy <- interfacesHypotheses[[hypothesesNr]]
                numberOfTasksForHy <- length(taskNrHypotheses[[hypothesesNr]])
                tasksForHy <- taskNrHypotheses[[hypothesesNr]]
                for(interfaceNr in 1:length(interfacesForHy)) {
                        time_sum <- 0
                        for(i in 1:numberOfTasksForHy) {

                                time <- log(aggregatedData[aggregatedData[1]==subjectNr &
                                        aggregatedData[6]==interfacesForHy[interfaceNr] &
                                        aggregatedData[2]==tasksForHy[i] & aggregatedData[8]==2,3])
                                # uncomment to calculate non log times
                                #time <- aggregatedData[aggregatedData[1]==subjectNr &
                                        aggregatedData[6]==interfacesForHy[interfaceNr] &
                                        aggregatedData[2]==tasksForHy[i]& aggregatedData[8] ==2,3]
                                if (length(time)>1){
                                        time_sum <- time_sum + time[1] + time[2]
                                }
                        }
                        if (time_sum>0) {
                                TimingsHypothesesRound2 <- rbind(TimingsHypothesesRound2,
                                        data.frame(subject=subjectNr,
                                        hypothesis=hypothesesNames[hypothesesNr],
                                        interface=interfacesForHy[interfaceNr], time_s=time_sum))
                        }
                }
        }
```

```
}

# change the order of the factor
TimingsHypothesesRound2$interface = factor(TimingsHypothesesRound2$interface,c("STZ","KNAVE"))


# ===================================================
# Summary Statistics
# ===================================================
sumstats_hypotheses_timings <- data.frame(hypothesis=character(0), interface=character(0),
        MIN=numeric(0), Q1=numeric(0), MEDIAN=numeric(0),
        MEAN=numeric(0), Q3=numeric(0), MAX=numeric(0), SD=numeric(0))

for(hypothesesNr in 1:length(hypothesesNames)) {
        interfacesForHy <- interfacesHypotheses[[hypothesesNr]]
        for(interfaceNr in 1:length(interfacesForHy)) {
                times <- TimingsHypothesesRound2[TimingsHypothesesRound2[3]==
                        interfacesForHy[interfaceNr] & TimingsHypothesesRound2[2]==
                        hypothesesNames[hypothesesNr],4]
                sumstats_hypotheses_timings <- rbind(sumstats_hypotheses_timings,
                        data.frame(hypothesis=hypothesesNames[hypothesesNr],
                        interface=interfacesForHy[interfaceNr],
                        MIN=min(times), Q1=quantile(times,0.25), MEDIAN=median(times),
                        MEAN=mean(times), Q3=quantile(times,0.75), MAX=max(times), SD=sd(times)))
        }
}

write.csv(sumstats_hypotheses_timings,file="sumstats_hypotheses_timings_round_2.csv")


# ===================================================
# Boxplot Task Completion Times
# ===================================================
pdf("timingBoxplots-hypotheses-row_round_2.pdf",width=10,height=4)
par(mfrow=c(1,4),mar=c(7,5,2,2),xpd=NA)
for (hypothesisNr in 1:4) {
        hypothesisString <- hypothesesNames[hypothesisNr]
        hypothesisDescription <- ""
        if (hypothesisNr == 1 | hypothesisNr == 3)
                hypothesisDescription <- paste(hypothesisString,"Lookup Tasks",sep=" ")
        if (hypothesisNr == 2 | hypothesisNr == 4)
                hypothesisDescription <- paste(hypothesisString,"Comparsion Tasks",sep=" ")
        boxplot(time_s ~ interface, data = TimingsHypothesesRound2,
                subset = hypothesis==hypothesisString, main=hypothesisDescription,
                ylab="time (s)", col = colors()[c(50,79,91)], boxwex=0.5, bty="n",
                cex.lab=1.4, cex.axis=1.0, cex.main=1.6, names = c("STZ", "KNAVE"),
                whisklty="solid")
        }

dev.off()

# ===================================================
# Test for normal distribution
# ===================================================
shapirowilk_H_Round_2 <- data.frame(Hypothesis="Hx",Interface="X",SW=0.0)
interfaceChars <- c("STZ","KNAVE")
for (hypothesisNr in 1:4) {

        hypothesisString <- hypothesesNames[hypothesisNr]
        TimingsH <-
                TimingsHypothesesRound2[TimingsHypothesesRound2[2]==hypothesisString,1:4]
        for(interfaceNr in 1:2) {
```

```
                times <- TimingsH[TimingsH[3]==interfaceChars[interfaceNr],4]
                if(length(times)>0) {
                        sw <- shapiro.test(times)$p.value
                        shapirowilk_H_Round_2 <- rbind(shapirowilk_H_Round_2, data.frame
                                (Hypothesis=hypothesisString,
                                Interface=interfaceChars[interfaceNr],SW=sw))
                }
        }
}
write.csv(shapirowilk_H_Round_2,file="shapirowilk_hypotheses_round_2.csv")

# ================================================
# Testing for equal variance (F-Test)
# ================================================

equal_var <- data.frame(Hypothesis="Hx",SW=0.0)

# H1.1
sw <- var.test(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.1" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.1" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H1.1",SW=sw))

# H1.2
sw <- var.test(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.2" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.2" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H1.2",SW=sw))

# H2.1
sw <- var.test(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.1" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.1" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H2.1",SW=sw))

# H2.2
sw <- var.test(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.2" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.2" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H2.2",SW=sw))

write.csv(equal_var,file="hypothesis_equal_variance_round_2.csv")

# ================================================
# T-Test (not paired, one sided)
# ================================================

t_test <- data.frame(Hypothesis="Hx",SW=0.0, D=0.0)

# H1.1
sw <- t.test(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.1" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.1" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"], paired = F, alternative="less")$p.value
```

```
# effect size
d <- abs(smd(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.1" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.1" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"]))


t_test <- rbind(t_test, data.frame(Hypothesis="H1.1",SW=sw, D=d))

# H1.2
sw <- t.test(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.2" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.2" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"], paired = F, alternative="less")$p.value

# effect size
d <- abs(smd(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.2" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H1.2" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"]))


t_test <- rbind(t_test, data.frame(Hypothesis="H1.2",SW=sw, D=d))

# H2.1
sw <- t.test(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.1" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.1" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"], paired = F, alternative="less")$p.value

# effect size
d <- abs(smd(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.1" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.1" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"]))


t_test <- rbind(t_test, data.frame(Hypothesis="H2.1",SW=sw, D=d))

# H2.2
sw <- t.test(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.2" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.2" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"], paired = F, alternative="less")$p.value

# effect size
d <- abs(smd(TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.2" &
        TimingsHypothesesRound2[3]=="STZ", "time_s"],
        TimingsHypothesesRound2[TimingsHypothesesRound2[2]=="H2.2" &
        TimingsHypothesesRound2[3]=="KNAVE", "time_s"]))


t_test <- rbind(t_test, data.frame(Hypothesis="H2.2",SW=sw, D=d))

write.csv(t_test,file="hypothesis_t_test_round_2.csv")
```

The following script tests the error rates of the task sets according to the hypotheses for significant differences for each round using Wilcoxon rank sum tests.

```
# ================================================
# HYPOTHESES - ERROR RATE
# ================================================
# Import data
visData <- read.table("raw_data.csv", header=TRUE, sep = ";")
visData <- visData[2:9]

numberOfSubjects <- 20
numberOfDatasets <- 2


# ================================================
# Calculate error counts and error rates for hypotheses
# Round 1
# ================================================

interfaceChars <- c("STZ","KNAVE")
hypothesesNames <- c("H1.1","H1.2","H2.1","H2.2")
taskNrH11 <- c(1,2,3)
taskNrH12 <- c(4,5,6)
taskNrH21 <- c(7,8,9)
taskNrH22 <- c(10,11,12)

taskNrHypotheses <- list(taskNrH11, taskNrH12, taskNrH21, taskNrH22)
interfacesH11 <- c("STZ","KNAVE")
interfacesH12 <- c("STZ","KNAVE")
interfacesH21 <- c("STZ","KNAVE")
interfacesH22 <- c("STZ","KNAVE")

interfacesHypotheses <- list(interfacesH11, interfacesH12, interfacesH21, interfacesH22)


ErrorsHypotheses_round_1 <- data.frame(subject=numeric(0),
        hypothesis=character(0), interface=character(0), errors=numeric(0),
        error_rate=numeric(0), success_rate=numeric(0) )
for (subjectNr in 1:numberOfSubjects) {
        for(hypothesesNr in 1:length(hypothesesNames)) {
                interfacesForHy <- interfacesHypotheses[[hypothesesNr]]
                numberOfTasksForHy <- length(taskNrHypotheses[[hypothesesNr]])
                tasksForHy <- taskNrHypotheses[[hypothesesNr]]
                for(interfaceNr in 1:length(interfacesForHy)) {
                        successCount <- 0
                        taskset <- numeric(0)
                        tasksuccess <- character(0)
                        for(i in 1:numberOfTasksForHy) {
                                success <- visData[visData[1]==subjectNr & visData[8] == 1 &
                                        visData[6]==interfacesForHy[interfaceNr] &
                                        visData[2]==tasksForHy[i],4]

                                if (length(success)>1){
                                                success_sum <- success[1] + success[2]
                                                successCount <- successCount + success_sum
                                                taskset <- c(taskset, tasksForHy[i])
                                                tasksuccess <- c(tasksuccess, success)

                                }
```

```
                        }

                        if (length(taskset)>0){
                                errorCount <- (numberOfTasksForHy*2) - successCount
                                errorRate <- errorCount/(numberOfTasksForHy*2)
                                ErrorsHypotheses_round_1 <- rbind(ErrorsHypotheses_round_1,
                                        data.frame(subject=subjectNr,
                                        hypothesis=hypothesesNames[hypothesesNr],
                                        interface=interfacesForHy[interfaceNr],
                                        errors=errorCount, error_rate=errorRate,
                                        success_rate = (1-errorRate) ))
                        }

                }
        }
}

# ================================================
# Summary Statistics
# ================================================
sumstats_hypotheses_errors_round_1 <- data.frame(hypothesis=character(0),
        interface=character(0), MIN=numeric(0), Q1=numeric(0), MEDIAN=numeric(0),
        MEAN=numeric(0), Q3=numeric(0), MAX=numeric(0), SD=numeric(0))

for(hypothesesNr in 1:length(hypothesesNames)) {
        interfacesForHy <- interfacesHypotheses[[hypothesesNr]]
        for(interfaceNr in 1:length(interfacesForHy)) {
                error_rates <- ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[3]==
                        interfacesForHy[interfaceNr] &
                        ErrorsHypotheses_round_1[2]==hypothesesNames[hypothesesNr],5]
                sumstats_hypotheses_errors_round_1 <- rbind(sumstats_hypotheses_errors_round_1,
                        data.frame(hypothesis=hypothesesNames[hypothesesNr],
                        interface=interfacesForHy[interfaceNr], MIN=min(error_rates),
                        Q1=quantile(error_rates,0.25), MEDIAN=median(error_rates),
                        MEAN=mean(error_rates), Q3=quantile(error_rates,0.75),
                        MAX=max(error_rates), SD=sd(error_rates)))
        }
}

write.csv(sumstats_hypotheses_errors_round_1,file="sumstats_hypotheses_errors_round_1.csv")

# ================================================
# Boxplot Error Rates - ALL in single plot
# ================================================
boxplot(error_rate ~ interface * hypothesis, data = ErrorsHypotheses_round_1,
        main="Error rates of hypotheses", xlab="Tasks", ylab="Error rate",
        col = colors()[c(50,79)], boxwex = 0.5)


# ================================================
# Test for normal distribution
# ================================================
shapirowilk_H_round_1 <- data.frame(Hypothesis=character(0),Interface=character(0),
        SW=numeric(0))
interfaceChars <- c("STZ","KNAVE")
for (hypothesisNr in 1:4) {
        hypothesisString <- hypothesesNames[hypothesisNr]
        ErrorsH <- ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]==hypothesisString,1:6]
        for(interfaceNr in 1:2) {
                success_rates <- ErrorsH[ErrorsH[3]==interfaceChars[interfaceNr],6]
                if(length(success_rates)>0) {
                        sw <- shapiro.test(success_rates)$p.value
```

```
                        shapirowilk_H_round_1 <- rbind(shapirowilk_H_round_1,
                        data.frame(Hypothesis=hypothesisString,
                        Interface=interfaceChars[interfaceNr],SW=sw))
                }
        }
}
write.csv(shapirowilk_H_round_1,file="shapirowilk_hypotheses_errors_round_1.csv")

# ================================================
# Test for log-normal distribution
# ================================================
shapirowilk_H_round_1_log <- data.frame(Hypothesis=character(0),Interface=character(0),SW=numeric(0))
interfaceChars <- c("STZ","KNAVE")
for (hypothesisNr in 1:4) {
        hypothesisString <- hypothesesNames[hypothesisNr]
        ErrorsH <- ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]==hypothesisString,1:6]
        for(interfaceNr in 1:2) {
                success_rates <- ErrorsH[ErrorsH[3]==interfaceChars[interfaceNr], 6]
                if(length(success_rates)>0) {
                        sw <- shapiro.test(log(success_rates))$p.value
                                shapirowilk_H_round_1_log <- rbind(shapirowilk_H_round_1_log,
                                data.frame(Hypothesis=hypothesisString,
                                Interface=interfaceChars[interfaceNr], SW=sw))
                }
        }
}
write.csv(shapirowilk_H_round_1_log,file="shapirowilk_hypotheses_log_errors_round_1.csv")

# ================================================
# Test for equal variances (F test)
# ================================================

equal_var <- data.frame(Hypothesis="Hx",SW=0.0)

# H1.1
sw <- var.test(ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H1.1" &
                ErrorsHypotheses_round_1[3]=="STZ", "error_rate"],
                ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H1.1" &
                ErrorsHypotheses_round_1[3]=="KNAVE", "error_rate"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H1.1",SW=sw))

# H1.2
sw <- var.test(ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H1.2" &
        ErrorsHypotheses_round_1[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H1.2" &
        ErrorsHypotheses_round_1[3]=="KNAVE", "error_rate"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H1.2",SW=sw))

# H2.1
sw <- var.test(ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H2.1" &
        ErrorsHypotheses_round_1[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H2.1" &
        ErrorsHypotheses_round_1[3]=="KNAVE", "error_rate"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H2.1",SW=sw))

# H2.2
sw <- var.test(ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H2.2" &
        ErrorsHypotheses_round_1[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H2.2" &
```

```
        ErrorsHypotheses_round_1[3]=="KNAVE", "error_rate"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H2.2",SW=sw))

write.csv(equal_var,file="error_hypothesis_equal_variance_round_1.csv")

# --> h2.1 not equal variance

# ================================================
# Wilcoxon rank sum test - Test hypotheses
# ================================================

 wilcox.test(ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H1.1" &
         ErrorsHypotheses_round_1[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H1.1" &
        ErrorsHypotheses_round_1[3]=="KNAVE", "error_rate"])

 wilcox.test(ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H1.2" &
         ErrorsHypotheses_round_1[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H1.2" &
        ErrorsHypotheses_round_1[3]=="KNAVE", "error_rate"])

 wilcox.test(ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H2.1" &
         ErrorsHypotheses_round_1[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H2.1" &
        ErrorsHypotheses_round_1[3]=="KNAVE", "error_rate"])

 wilcox.test(ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H2.2" &
         ErrorsHypotheses_round_1[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_1[ErrorsHypotheses_round_1[2]=="H2.2" &
        ErrorsHypotheses_round_1[3]=="KNAVE", "error_rate"])


# ================================================
# Calculate error counts and error rates for hypotheses
# Round 2
# ================================================

ErrorsHypotheses_round_2 <- data.frame(subject=numeric(0),
        hypothesis=character(0), interface=character(0), errors=numeric(0), error_rate=numeric(0))
for (subjectNr in 1:numberOfSubjects) {
        for(hypothesesNr in 1:length(hypothesesNames)) {
                interfacesForHy <- interfacesHypotheses[[hypothesesNr]]
                numberOfTasksForHy <- length(taskNrHypotheses[[hypothesesNr]])
                tasksForHy <- taskNrHypotheses[[hypothesesNr]]
                for(interfaceNr in 1:length(interfacesForHy)) {
                        successCount <- 0
                        taskset <- numeric(0)
                        tasksuccess <- character(0)
                        for(i in 1:numberOfTasksForHy) {
                                success <- visData[visData[1]==subjectNr & visData[8] == 2 &
                                        visData[6]==interfacesForHy[interfaceNr] &
                                        visData[2]==tasksForHy[i],4]

                                if (length(success)>1){
                                                success_sum <- success[1] + success[2]
                                                successCount <- successCount + success_sum
                                                taskset <- c(taskset, tasksForHy[i])
                                                tasksuccess <- c(tasksuccess, success)

                                }
```

```
                        }

                        if (length(taskset)>0){
                                errorCount <- (numberOfTasksForHy*2) - successCount
                                errorRate <- errorCount/(numberOfTasksForHy*2)
                                ErrorsHypotheses_round_2 <- rbind(ErrorsHypotheses_round_2,
                                        data.frame(subject=subjectNr,
                                        hypothesis=hypothesesNames[hypothesesNr],
                                        interface=interfacesForHy[interfaceNr],
                                        errors=errorCount, error_rate=errorRate))
                        }
                }
        }
}

# ====================================================
# Summary Statistics
# ====================================================
sumstats_hypotheses_errors_round_2 <- data.frame(hypothesis=character(0), interface=character(0),
        MIN=numeric(0), Q1=numeric(0), MEDIAN=numeric(0), MEAN=numeric(0), Q3=numeric(0),
        MAX=numeric(0), SD=numeric(0))

for(hypothesesNr in 1:length(hypothesesNames)) {
        interfacesForHy <- interfacesHypotheses[[hypothesesNr]]
        for(interfaceNr in 1:length(interfacesForHy)) {
                error_rates <- ErrorsHypotheses_round_2[ErrorsHypotheses_round_2[3]==
                        interfacesForHy[interfaceNr] &
                        ErrorsHypotheses_round_2[2]==hypothesesNames[hypothesesNr],5]
                sumstats_hypotheses_errors_round_2 <- rbind(sumstats_hypotheses_errors_round_2,
                        data.frame(hypothesis=hypothesesNames[hypothesesNr],
                        interface=interfacesForHy[interfaceNr], MIN=min(error_rates),
                        Q1=quantile(error_rates,0.25), MEDIAN=median(error_rates),
                        MEAN=mean(error_rates), Q3=quantile(error_rates,0.75),
                        MAX=max(error_rates), SD=sd(error_rates)))
        }
}

write.csv(sumstats_hypotheses_errors_round_2,file="sumstats_hypotheses_errors_round_2.csv")

# ====================================================
# Boxplot Error Rates - ALL in single plot
# ====================================================
boxplot(error_rate ~ interface * hypothesis, data = ErrorsHypotheses_round_2,
        main="Error rates of hypotheses", xlab="Tasks", ylab="Error rate",
        col = colors()[c(50,79)], boxwex = 0.5)

# ====================================================
# Test for equal variances (F test)
# ====================================================

equal_var <- data.frame(Hypothesis="Hx",SW=0.0)

# H1.1
sw <- var.test(ErrorsHypotheses_round_2[ErrorsHypotheses_round_2[2]=="H1.1" &
        ErrorsHypotheses_round_2[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_2[ErrorsHypotheses_round_2[2]=="H1.1" &
        ErrorsHypotheses_round_2[3]=="KNAVE", "error_rate"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H1.1",SW=sw))

# H1.2
```

```
sw <- var.test(ErrorsHypotheses_round_2[ErrorsHypotheses_round_2[2]=="H1.2" &
        ErrorsHypotheses_round_2[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_2[ErrorsHypotheses_round_2[2]=="H1.2" &
        ErrorsHypotheses_round_2[3]=="KNAVE", "error_rate"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H1.2",SW=sw))

# H2.1
sw <- var.test(ErrorsHypotheses_round_2[ErrorsHypotheses_round_2[2]=="H2.1" &
        ErrorsHypotheses_round_2[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_2[ErrorsHypotheses_round_2[2]=="H2.1" &
        ErrorsHypotheses_round_2[3]=="KNAVE", "error_rate"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H2.1",SW=sw))

# H2.2
sw <- var.test(ErrorsHypotheses_round_2[ErrorsHypotheses_round_2[2]=="H2.2" &
        ErrorsHypotheses_round_2[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_2[ErrorsHypotheses_round_2[2]=="H2.2" &
        ErrorsHypotheses_round_2[3]=="KNAVE", "error_rate"])$p.value

equal_var <- rbind(equal_var, data.frame(Hypothesis="H2.2",SW=sw))

write.csv(equal_var,file="error_hypothesis_equal_variance_round_2.csv")

# ================================================
# Wilcoxon rank sum test - Test hypotheses
# ================================================

 wilcox.test(ErrorsHypotheses_round_2[ErrorsHypotheses_round_1[2]=="H1.1" &
         ErrorsHypotheses_round_2[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_2[ErrorsHypotheses_round_1[2]=="H1.1" &
        ErrorsHypotheses_round_2[3]=="KNAVE", "error_rate"])

 wilcox.test(ErrorsHypotheses_round_2[ErrorsHypotheses_round_1[2]=="H1.2" &
         ErrorsHypotheses_round_2[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_2[ErrorsHypotheses_round_1[2]=="H1.2" &
        ErrorsHypotheses_round_2[3]=="KNAVE", "error_rate"])

 wilcox.test(ErrorsHypotheses_round_2[ErrorsHypotheses_round_1[2]=="H2.1" &
         ErrorsHypotheses_round_2[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_2[ErrorsHypotheses_round_1[2]=="H2.1" &
        ErrorsHypotheses_round_2[3]=="KNAVE", "error_rate"])

 wilcox.test(ErrorsHypotheses_round_2[ErrorsHypotheses_round_1[2]=="H2.2" &
         ErrorsHypotheses_round_2[3]=="STZ", "error_rate"],
        ErrorsHypotheses_round_2[ErrorsHypotheses_round_1[2]=="H2.2" &
        ErrorsHypotheses_round_2[3]=="KNAVE", "error_rate"])
```

APPENDIX F

User Tasks

F.1 Training Tasks

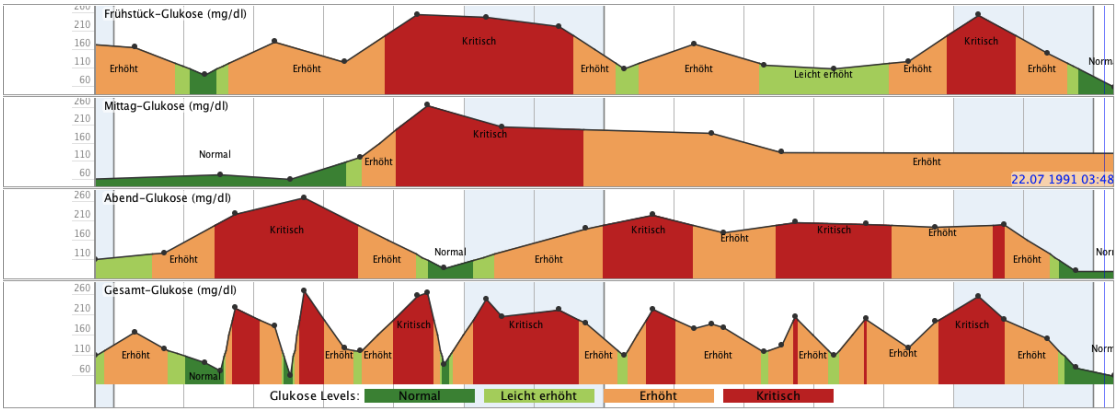


Figure F.1: SemTimeZoom visualization of the training dataset

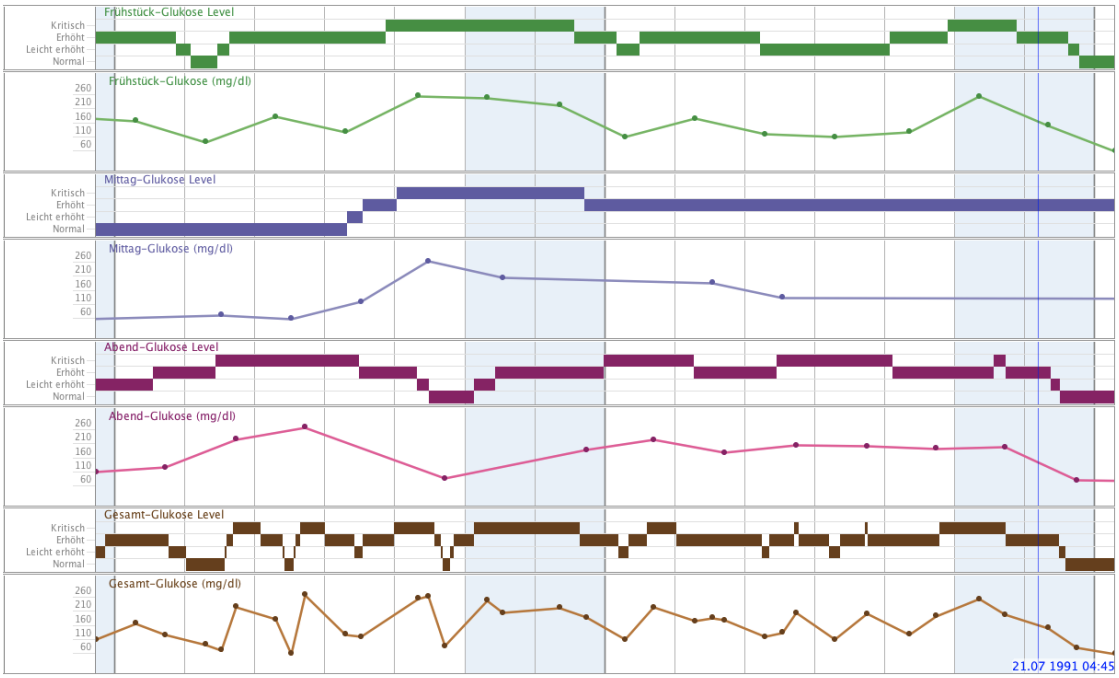


Figure F.2: KNAVE visualization of the training dataset

Task	*Task description*
02	Markiere den ersten Intervall in dem sich sowohl Frühstücks-Glukose als auch Mittags-Glukose im erhöhten Level befinden. (Wo überlappen sich diese beiden Intervalle?)
11	Frühstück-, Mittag-, Abend-Glukose: Vergleiche die höchsten gemessenen Werte im jeweils ersten kritischen Interval dieser Variablen. Welche Variable besitzt dabei den höchsten Wert?
08	Welchen Wert hat der nächste gemessene Datenpunkt von Abend-Glukose, nachdem Gesamt-Glukose den normalen Level das erste mal verlässt? Schreibe den Wert in das Eingabefeld.

Table F.1: Concrete training tasks

Training tasks formulated in XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
Trainings task list
Created by Stephan Hoffmann on 2011-05-11.
-->
<taskList>
        <tasks>
                <interval_selection>
                        <taskId>TrainingTask1</taskId>
                        <taskType>02</taskType>
                        <taskDescription>Markiere den ersten Intervall in dem sich sowohl
                        Fruehstuecks-Glukose als auch Mittags-Glukose im erhoehten Level
                        befinden. (Wo ueberlappen sich diese beiden Intervalle?)
                        </taskDescription>
                        <intervalStart>1991-05-29T19:37:00</intervalStart>
                        <intervalEnd>1991-05-30T11:48:00</intervalEnd>
                        <tolerance>14400000</tolerance>
                        <!-- 4 hours -->
                </interval_selection>
                <taskId>TrainingTask2</taskId>
                <taskType>11</taskType>
                <choice_selection>
                        <taskDescription>Fruehstueck-, Mittag-, Abend-Glukose: Vergleiche
                        die hoechsten gemessenen Werte im jeweils ersten kritischen
                         Interval dieser Variablen. Welche Variable besitzt dabei den
                         hoechsten Wert?</taskDescription>
                        <taskInstruction/>
                        <possibleAnswers>
                                <possibleAnswer>Fruehstueck</possibleAnswer>
                                <possibleAnswer>Mittag</possibleAnswer>
                                <possibleAnswer>Abend</possibleAnswer>
                        </possibleAnswers>
                        <correctAnswers>
                                <correctAnswer>Abend</correctAnswer>
                        </correctAnswers>
                        <singleChoice>true</singleChoice>
                </choice_selection>
                <quantitative>
                        <taskId>TrainingTask3</taskId>
                        <taskType>8</taskType>
                        <taskDescription>Welchen Wert hat der naechste gemessene
                        Datenpunkt von Abend-Glukose, nachdem Gesamt-Glukose
                        den normalen Level das erste mal verlaesst? Schreibe den
                        Wert in das Eingabefeld.</taskDescription>
                        <taskInstruction/>
                        <unit>mg/dl</unit>
                        <isInteger>false</isInteger>
                        <correctValue>229.0</correctValue>
                        <tolerance>0.1</tolerance>
                </quantitative>
        </tasks>
</taskList>
```

F.2 Tasks Dataset 1

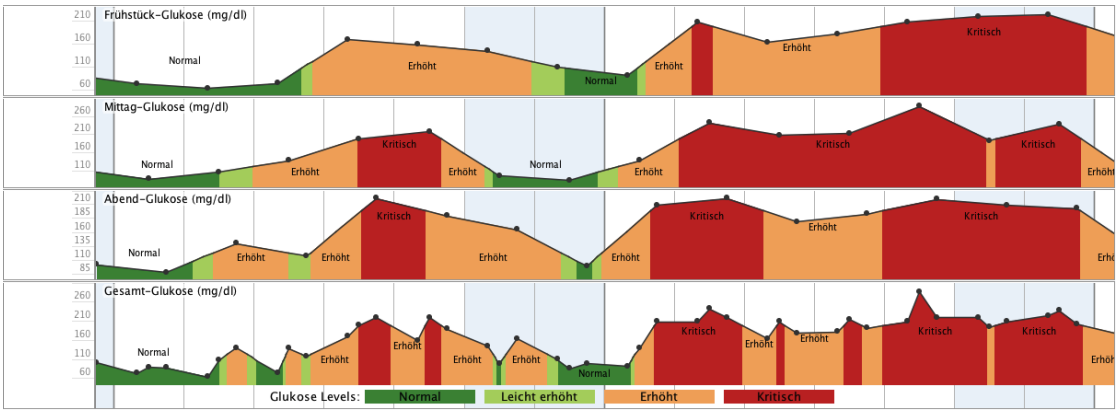


Figure F.3: SemTimeZoom visualization of the first dataset

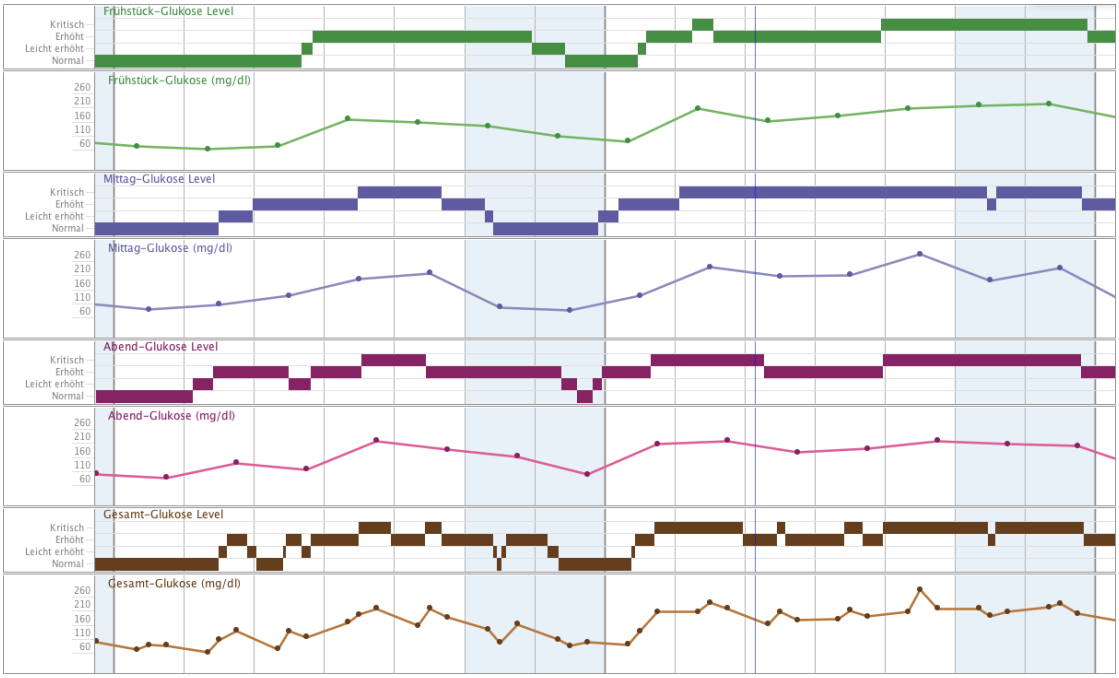


Figure F.4: KNAVE visualization of the the first dataset

Task	*Task description*
01	Wie oft ist Abend-Glukose in einem normalen Level?
01	Wie oft ist Frühstück-Glukose in einem erhöhten Level?
02	Markiere das erste Intervall in dem sich sowohl Frühstücks-Glukose als auch Mittags-Glukose im erhöhten Level befinden. (Wo überlappen sich diese beiden Intervalle?)
02	Markiere das zweite Intervall in dem sich sowohl Mittag-Glukose als auch Abend-Glukose im kritischen Level befinden. (Wo überlappen sich diese beiden Intervalle?)
03	Frühstück-Glukose: Markiere das erste Intervall eines erhöhten Levels.
03	Mittag-Glukose: Markiere das erste Intervall eines kritischen Levels.

Table F.2: Concrete lookup tasks for dataset one addressing the qualitative attributes of the data. Each task is repeated once with slightly different targets.

Task	*Task description*
04	Frühstück-Glukose: Ist der erste Glukose-Level höher/niedriger/gleich als der dritte Glukose Level?
04	Gesamt-Glukose: Ist der erste Glukose-Level höher/niedriger/gleich als der dritte Glukose Level?
05	Welche Glukose Variable hat den am längsten dauernden erhöhten Glukose Level?
05	Welche Glukose Variable hat den am längsten dauernden kritischen Glukose Level?
06	Welche Glukose Variable hat das häufigste Auftreten von kritischen Glukose Levels? Die Dauer der Intervalle spielt dabei keine Rolle.
06	Welche Glukose Variable hat am wenigsten Auftritte von erhöhten Glukose Levels? Die Dauer der Intervalle spielt dabei keine Rolle.

Table F.3: Concrete comparison tasks for dataset one addressing the qualitative attributes of the data. Each task is repeated once with slightly different targets.

Task	*Task description*
07	Welche Glukose-Variable hat gerade einen Anstieg wenn Frühstück-Glukose den kritischen Level das erste Mal verlässt? Vergleiche dabei nur die unmittelbar davor und danach liegenden gemessenen Datenpunkte.
07	Welche Glukose-Variable hat gerade einen Anstieg wenn Gesamt-Glukose den kritischen Level das erste Mal verlässt? Vergleiche dabei nur die unmittelbar davor und danach liegenden gemessenen Datenpunkte.
08	Welchen Wert hat der nächste gemessene Datenpunkt von Abend-Glukose nachdem Gesamt-Glukose den normalen Level das erste mal verlässt? Schreibe den Wert in das Eingabefeld.
08	Welchen Wert hat der nächste gemessene Datenpunkt von Mittag-Glukose nachdem Frühstück-Glukose das erste mal in den erhöhten Level eintritt? Schreibe den Wert in das Eingabefeld.
09	Mittag-Glukose: Wieviele Messpunkte beinhaltet das zweite kritische Intervall.
09	Frühstück-Glukose: Wieviele Messpunkte beinhaltet das erste normale Intervall.

Table F.4: Concrete tasks for dataset one addressing the quantitative attributes of the data in defined qualitative levels . Each task is repeated once with slightly different targets.

Task	*Task description*
10	Frühstück-Glukose: Welches kritische Intervall beinhaltet am meisten Messpunkte? Markiere das gesamte kritische Intervall.
10	Abend-Glukose: Welches leicht erhöhte Intervall beinhaltet am meisten Messpunkte? Markiere das gesamte leicht erhöhte Intervall.
11	Frühstück-, Mittag-, Abend-Glukose: Vergleiche die höchsten gemessenen Werte im jeweils ersten kritischen Interval dieser Variablen. Welche Variable besitzt dabei den höchsten Wert?
11	Frühstück-, Mittag-, Abend-Glukose: Vergleiche die niedrigsten gemessenen Werte im jeweils ersten erhöhten Interval dieser Variablen. Welche Variable besitzt dabei den niedrigsten Wert?
12	Frühstück-Glukose: Finde den höchsten gemessenen Wert im letzen kritischen Glukose Level und schreibe den Wert in das Eingabefeld.
12	Mittag-Glukose: Finde den niedrigsten gemessenen Wert im zweiten normalen Glukose Level und schreibe den Wert in das Eingabefeld.

Table F.5: Concrete comparison tasks for dataset one addressing the quantitative attributes of the data in defined qualitative levels . Each task is repeated once with slightly different targets.

F.3 Tasks Dataset 1 formulated in XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
Task list Dataset 1
Created by Stephan Hoffmann on 2011-05-11.
-->
<taskList>
        <tasks>
                <!-- 1 -->
                <quantitative>
                        <taskId>01</taskId>
                        <taskType>01</taskType>
                        <taskDescription>Wie oft ist Abend-Glukose in einem
                        normalen Level?</taskDescription>
                        <taskInstruction/>
                        <unit>Mal</unit>
                        <isInteger>true</isInteger>
                        <correctValue>2</correctValue>
                        <tolerance>0.0</tolerance>
                </quantitative>
                <quantitative>
                        <taskId>02</taskId>
                        <taskType>01</taskType>
                        <taskDescription>Wie oft ist Fruehstueck-Glukose in einem
                        erhoehten Level?</taskDescription>
                        <taskInstruction/>
                        <unit>Mal</unit>
                        <isInteger>true</isInteger>
                        <correctValue>4</correctValue>
                        <tolerance>0.0</tolerance>
                </quantitative>
                <!-- 2 -->
                <interval_selection>
                        <taskId>03</taskId>
                        <taskType>02</taskType>
                        <taskDescription>Markiere das erste Intervall in dem sich sowohl
                        Fruehstuecks-Glukose als auch Mittags-Glukose
                        im erhoehten Level befinden.
                        (Wo ueberlappen sich diese beiden Intervalle?)
                        </taskDescription>
                        <intervalStart>1991-05-29T19:37:00</intervalStart>
                        <intervalEnd>1991-05-30T11:48:00</intervalEnd>
                        <tolerance>14400000</tolerance>
                        <!-- 4 hours -->
                </interval_selection>
                <interval_selection>
                        <taskId>04</taskId>
                        <taskType>02</taskType>
                        <taskDescription>Markiere das zweite Intervall in dem sich sowohl
                        Mittag-Glukose als auch Abend-Glukose im
                        kritischen Level befinden.
                        (Wo ueberlappen sich diese beiden Intervalle?)
                        </taskDescription>
                        <intervalStart>1991-06-04T01:36:00</intervalStart>
                        <intervalEnd>1991-06-05T06:39:00</intervalEnd>
                        <tolerance>14400000</tolerance>
                        <!-- 4 hours -->
                </interval_selection>
                <!-- 3 -->
                <interval_selection>
```

```
        <taskId>05</taskId>
        <taskType>03</taskType>
        <taskDescription>Fruehstueck-Glukose: Markiere das
        erste Intervall eines erhoehten Levels.
        </taskDescription>
        <intervalStart>1991-05-29T19:56:00</intervalStart>
        <intervalEnd>1991-06-01T23:03:00</intervalEnd>
        <tolerance>14400000</tolerance>
        <!-- 4 hours -->
</interval_selection>
<interval_selection>
        <taskId>06</taskId>
        <taskType>03</taskType>
        <taskDescription>Mittag-Glukose: Markiere das erste
         Intervall eines kritischen Levels.
         </taskDescription>
        <intervalStart>1991-05-30T11:48:00</intervalStart>
        <intervalEnd>1991-05-31T15:52:00</intervalEnd>
        <tolerance>14400000</tolerance>
        <!-- 4 hours -->
</interval_selection>
<!-- 4 -->
<choice_selection>
        <taskId>07</taskId>
        <taskType>04</taskType>
        <taskDescription>Fruehstueck-Glukose: Ist der
        erste Glukose-Level
        hoeher/niedriger/gleich als der dritte Glukose Level?
        </taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>hoeher</possibleAnswer>
                <possibleAnswer>gleich</possibleAnswer>
                <possibleAnswer>niedriger</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>niedriger</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<choice_selection>
        <taskId>08</taskId>
        <taskType>04</taskType>
        <taskDescription>Gesamt-Glukose: Ist der erste Glukose-Level
        hoeher/niedriger/gleich als der dritte Glukose Level?
        </taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>hoeher</possibleAnswer>
                <possibleAnswer>gleich</possibleAnswer>
                <possibleAnswer>niedriger</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>niedriger</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<!-- 5 -->
<choice_selection>
        <taskId>09</taskId>
        <taskType>05</taskType>
        <taskDescription>Welche Glukose Variable hat den am
```

```
        laengsten dauernden erhoehten Glukose Level?
        </taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Gesamt</possibleAnswer>
                <possibleAnswer>Fruehstueck</possibleAnswer>
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>Fruehstueck</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<choice_selection>
        <taskId>10</taskId>
        <taskType>05</taskType>
        <taskDescription>Welche Glukose Variable hat den am laengsten
        dauernden kritischen Glukose Level?</taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Gesamt</possibleAnswer>
                <possibleAnswer>Fruehstueck</possibleAnswer>
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>Mittag</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<choice_selection>
        <taskId>11</taskId>
        <taskType>06</taskType>
        <taskDescription>Welche Glukose Variable hat das haeufigste
        Auftreten von kritischen Glukose Levels? Die Dauer der
        Intervalle spielt dabei keine Rolle.</taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Gesamt</possibleAnswer>
                <possibleAnswer>Fruehstueck</possibleAnswer>
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>Gesamt</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<choice_selection>
        <taskId>12</taskId>
        <taskType>06</taskType>
        <taskDescription>Welche Glukose Variable hat am wenigsten Auftritte
        von erhoehten Glukose Levels? Die Dauer der Intervalle spielt dabei
        keine Rolle.</taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Gesamt</possibleAnswer>
                <possibleAnswer>Fruehstueck</possibleAnswer>
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
```

```
        <correctAnswers>
                <correctAnswer>Fruehstueck</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<quantitative>
        <taskId>13</taskId>
        <taskType>12</taskType>
        <taskDescription>Fruehstueck-Glukose: Finde den hoechsten gemessenen
         Wert im letzen kritischen Glukose Level und schreibe den Wert in das
         Eingabefeld.</taskDescription>
        <taskInstruction/>
        <unit>mg/dl</unit>
        <isInteger>false</isInteger>
        <correctValue>226.0</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<quantitative>
        <taskId>14</taskId>
        <taskType>12</taskType>
        <taskDescription>Mittag-Glukose: Finde den niedrigsten gemessenen Wert
        im zweiten normalen Glukose Level und schreibe den Wert in das
        Eingabefeld.</taskDescription>
        <taskInstruction/>
        <unit>mg/dl</unit>
        <isInteger>false</isInteger>
        <correctValue>88.0</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<quantitative>
        <taskId>15</taskId>
        <taskType>08</taskType>
        <taskDescription>Welchen Wert hat der naechste gemessene Datenpunkt
        von Abend-Glukose nachdem Gesamt-Glukose den normalen Level das
        erste mal verlaesst? Schreibe den Wert in das Eingabefeld.
        </taskDescription>
        <taskInstruction/>
        <unit>mg/dl</unit>
        <isInteger>false</isInteger>
        <correctValue>142.0</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<quantitative>
        <taskId>16</taskId>
        <taskType>08</taskType>
        <taskDescription>Welchen Wert hat der naechste gemessene Datenpunkt
        von Mittag-Glukose nachdem Fruehstueck-Glukose das erste mal in den
        erhoehten Level eintritt? Schreibe den Wert in das Eingabefeld.
        </taskDescription>
        <taskInstruction/>
        <unit>mg/dl</unit>
        <isInteger>false</isInteger>
        <correctValue>201.0</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<quantitative>
        <taskId>17</taskId>
        <taskType>09</taskType>
        <taskDescription>Mittag-Glukose: Wieviele Messpunkte beinhaltet das
        zweite kritische Intervall.</taskDescription>
        <taskInstruction/>
        <unit>Messpunkte</unit>
```

```
        <isInteger>true</isInteger>
        <correctValue>4</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<quantitative>
        <taskId>18</taskId>
        <taskType>09</taskType>
        <taskDescription>Fruehstueck-Glukose: Wieviele Messpunkte beinhaltet das
        erste normale Intervall.</taskDescription>
        <taskInstruction/>
        <unit>Messpunkte</unit>
        <isInteger>true</isInteger>
        <correctValue>3</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<interval_selection>
        <taskId>19</taskId>
        <taskType>10</taskType>
        <taskDescription>Fruehstueck-Glukose: Welches kritische Intervall beinhaltet
        am meisten Messpunkte? Markiere das gesamte kritische Intervall.
        </taskDescription>
        <intervalStart>1991-06-06T22:44:00</intervalStart>
        <intervalEnd>1991-06-09T21:11:00</intervalEnd>
        <tolerance>14400000</tolerance>
        <!-- 4 hours -->
</interval_selection>
<interval_selection>
        <taskId>20</taskId>
        <taskType>10</taskType>
        <taskDescription>Abend-Glukose: Welches leicht erhoehte Intervall beinhaltet
        am meisten Messpunkte? Markiere das gesamte leicht erhoehte Intervall.
        </taskDescription>
        <intervalStart>1991-05-29T11:11:00</intervalStart>
        <intervalEnd>1991-05-29T19:00:00</intervalEnd>
        <tolerance>14400000</tolerance>
        <!-- 4 hours -->
</interval_selection>
<choice_selection>
        <taskId>21</taskId>
        <taskType>11</taskType>
        <taskDescription>Fruehstueck-, Mittag-, Abend-Glukose: Vergleiche die
        hoechsten gemessenen Werte im jeweils ersten kritischen Interval
        dieser Variablen.Welche Variable besitzt dabei den hoechsten Wert?
        </taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Fruehstueck</possibleAnswer>
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>Abend</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<choice_selection>
        <taskId>22</taskId>
        <taskType>11</taskType>
        <taskDescription>Fruehstueck-, Mittag-, Abend-Glukose: Vergleiche die
        niedrigsten gemessenen Werte im jeweils ersten erhoehten Interval
        dieser Variablen. Welche Variable besitzt dabei den niedrigsten Wert?
        </taskDescription>
```

```
                        <taskInstruction/>
                        <possibleAnswers>
                                <possibleAnswer>Fruehstueck</possibleAnswer>
                                <possibleAnswer>Mittag</possibleAnswer>
                                <possibleAnswer>Abend</possibleAnswer>
                        </possibleAnswers>
                        <correctAnswers>
                                <correctAnswer>Mittag</correctAnswer>
                        </correctAnswers>
                        <singleChoice>true</singleChoice>
                </choice_selection>
                <choice_selection>
                        <taskId>23</taskId>
                        <taskType>07</taskType>
                        <taskDescription>Welche Glukose-Variable hat gerade einen Anstieg
                        wenn Gesamt-Glukose den kritischen Level das erste Mal verlaesst?
                        Vergleiche dabei nur die unmittelbar davor und danach liegenden
                        gemessenen Datenpunkte</taskDescription>
                        <taskInstruction/>
                        <possibleAnswers>
                                <possibleAnswer>Fruehstueck</possibleAnswer>
                                <possibleAnswer>Mittag</possibleAnswer>
                                <possibleAnswer>Abend</possibleAnswer>
                        </possibleAnswers>
                        <correctAnswers>
                                <correctAnswer>Mittag</correctAnswer>
                        </correctAnswers>
                        <singleChoice>false</singleChoice>
                </choice_selection>
                <choice_selection>
                        <taskId>24</taskId>
                        <taskType>07</taskType>
                        <taskDescription>Welche Glukose-Variable hat gerade einen Anstieg
                        wenn Fruehstueck-Glukose den kritischen Level das erste Mal verlaesst?
                        Vergleiche dabei nur die unmittelbar davor und danach liegenden
                        gemessenen Datenpunkte</taskDescription>
                        <taskInstruction/>
                        <possibleAnswers>
                                <possibleAnswer>Gesamt</possibleAnswer>
                                <possibleAnswer>Mittag</possibleAnswer>
                                <possibleAnswer>Abend</possibleAnswer>
                        </possibleAnswers>
                        <correctAnswers>
                                <correctAnswer>Abend</correctAnswer>
                        </correctAnswers>
                        <singleChoice>false</singleChoice>
                </choice_selection>
        </tasks>
</taskList>
```

F.4 Tasks Dataset 2

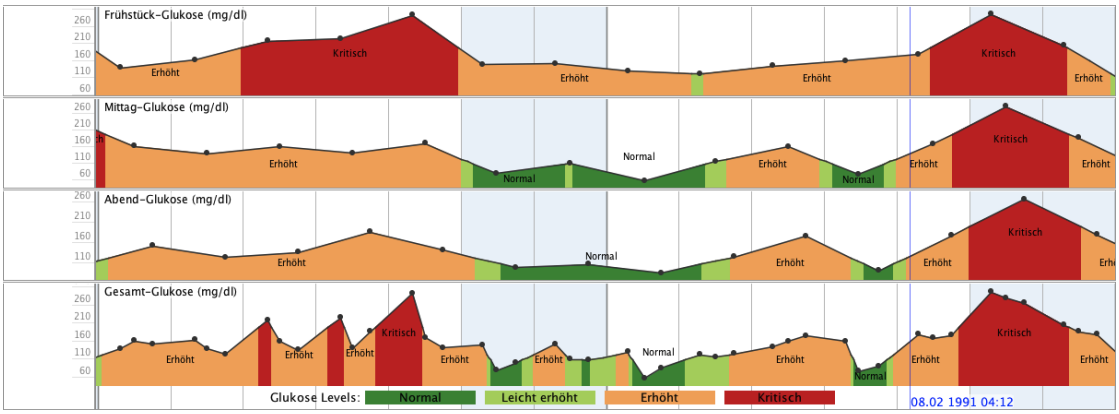


Figure F.5: SemTimeZoom visualization of the second dataset

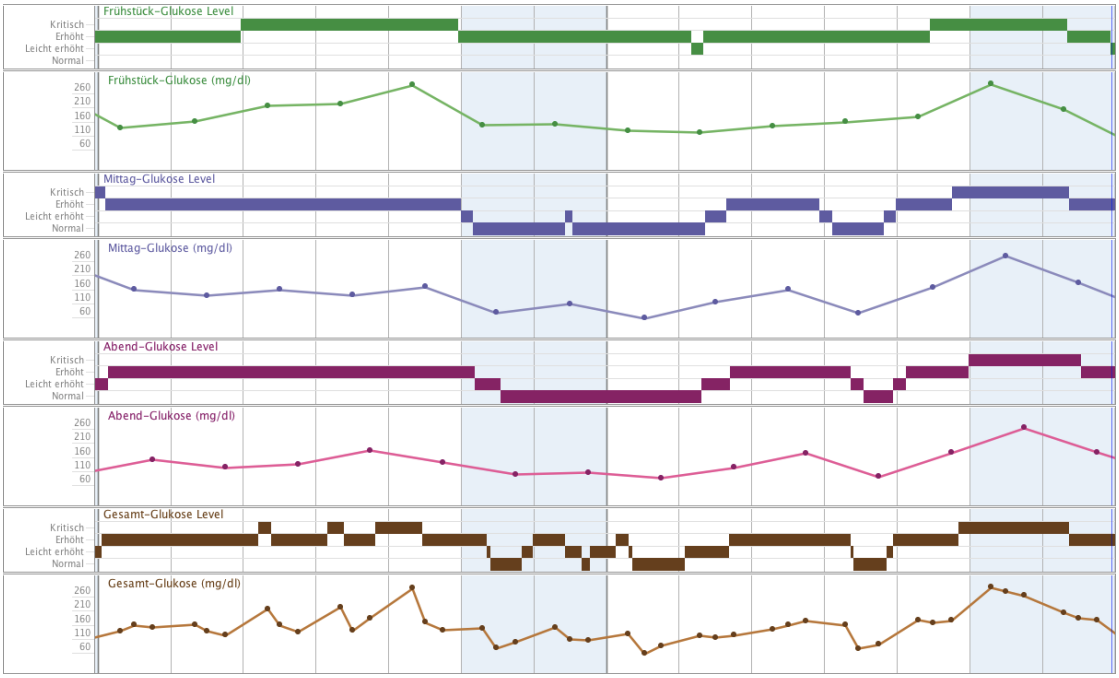


Figure F.6: KNAVE visualization of the the second dataset

Task	*Task description*
01	Wie oft ist Frühstück-Glukose in einem kritischen Level?
01	Wie oft ist Abend-Glukose in einem normalen Level?
02	Markiere das erste Intervall in dem sich sowohl Gesamt-Glukose als auch Mittags-Glukose im normalen Level befinden. (Wo überlappen sich diese beiden Intervalle?)
02	Markiere das erste Intervall in dem sich sowohl Gesamt-Glukose als auch Frühstück-Glukose im kritischen Level befinden. (Wo überlappen sich diese beiden Intervalle?)
03	Abend-Glukose: Markiere das zweite Intervall eines erhöhten Levels.
03	Mittag-Glukose: Markiere das erste Intervall eines normalen Levels.

Table F.6: Concrete lookup tasks for dataset two addressing the qualitative attributes of the data. Each task is repeated once with slightly different targets.

Task	*Task description*
04	Frühstück-Glukose: Ist der erste Glukose-Level höher/niedriger/gleich als der dritte Glukose Level?
04	Mittag-Glukose: Ist der erste Glukose-Level höher/niedriger/gleich als der dritte Glukose Level?
05	Welche Glukose Variable hat den am längsten dauernden erhöhten Glukose Level?
05	Welche Glukose Variable hat den am längsten dauernden kritischen Glukose Level?
06	Welche Glukose Variable hat das häufigste Auftreten von kritischen Glukose Levels?
06	Welche Glukose Variable hat am wenigsten Auftritte von leicht erhöhten Glukose Levels?

Table F.7: Concrete comparison tasks for dataset two addressing the qualitative attributes of the data. Each task is repeated once with slightly different targets.

Task	*Task description*
07	Welche Glukose-Variable hat gerade einen Abfall wenn Gesamt-Glukose das dritte Mal in den kritischen Level eintritt? Vergleiche dabei nur die unmittelbar davor und danach liegenden gemessenen Datenpunkte.
07	Welche Glukose-Variable hat gerade einen Anstieg wenn Mittag-Glukose das zweite Mal den erhöhten Level verlässt?Vergleiche dabei nur die unmittelbar davor und danach liegenden gemessenen Datenpunkte.
08	Welchen Wert hat der nächste gemessene Datenpunkt von Mittag-Glukose nachdem Gesamt-Glukose den kritischen Level das erste mal verlässt? Schreibe den Wert in das Eingabefeld.
08	Welchen Wert hat der nächste gemessene Datenpunkt von Abend-Glukose nachdem Frühstück-Glukose den erhöhten Level das erste mal verlässt? Schreibe den Wert in das Eingabefeld.
09	Mittag-Glukose: Wieviele Messpunkte beinhaltet das erste normale Intervall.
09	Frühstück-Glukose: Wieviele Messpunkte beinhaltet das erste leicht erhöhte Intervall.

Table F.8: Concrete tasks for dataset two addressing the quantitative attributes of the data in defined qualitative levels . Each task is repeated once with slightly different targets.

Task	*Task description*
10	Frühstück-Glukose: Welches kritische Intervall beinhaltet am meisten Messpunkte? Markiere das gesamte kritische Intervall.
10	Gesamt-Glukose: Welches kritische Intervall beinhaltet am meisten Messpunkte? Markiere das gesamte kritische Intervall.
11	Frühstück-, Mittag-, Abend-Glukose: Vergleiche die höchsten gemessenen Werte im jeweils letzten kritischen Interval dieser Variablen. Welche Variable besitzt dabei den höchsten Wert?
11	Frühstück-, Mittag-, Abend-Glukose: Vergleiche die niedrigsten gemessenen Werte im jeweils ersten erhöhten Interval dieser Variablen. Welche Variable besitzt dabei den kleinsten Wert?
12	Frühstück-Glukose: Finde den höchsten gemessenen Wert im ersten kritischen Glukose Level und schreibe den Wert in das Eingabefeld.
12	Gesamt-Glukose: Finde den niedrigsten gemessenen Wert im ersten normalen Glukose Level und schreibe den Wert in das Eingabefeld.

Table F.9: Concrete comparison tasks for dataset two addressing the quantitative attributes of the data in defined qualitative levels . Each task is repeated once with slightly different targets.

F.5 Tasks Dataset 2 formulated in XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
Task list Dataset 2
Created by Stephan Hoffmann on 2011-05-31.
-->
<taskList>
        <tasks>
                <!-- 1 -->
                <quantitative>
                        <taskId>01</taskId>
                        <taskType>01</taskType>
                        <taskDescription>Wie oft ist Fruehstueck-Glukose in einem
                        kritischen Level?</taskDescription>
                        <taskInstruction/>
                        <unit>Mal</unit>
                        <isInteger>true</isInteger>
                        <correctValue>2</correctValue>
                        <tolerance>0.0</tolerance>
                </quantitative>
                <quantitative>
                        <taskId>02</taskId>
                        <taskType>01</taskType>
                        <taskDescription>Wie oft ist Abend-Glukose in einem normalen
                        Level?</taskDescription>
                        <taskInstruction/>
                        <unit>Mal</unit>
                        <isInteger>true</isInteger>
                        <correctValue>2</correctValue>
                        <tolerance>0.0</tolerance>
                </quantitative>
                <!-- 2 -->
                <interval_selection>
                        <taskId>03</taskId>
                        <taskType>02</taskType>
                        <taskDescription>Markiere das erste Intervall in dem sich sowohl
                        Gesamt-Glukose als auch Mittags-Glukose im normalen Level
                        befinden. (Wo ueberlappen sich diese beiden Intervalle?)
                        </taskDescription>
                        <intervalStart>1991-02-02T09:41:00</intervalStart>
                        <intervalEnd>1991-02-02T19:44:00</intervalEnd>
                        <tolerance>14400000</tolerance>
                        <!-- 4 hours -->
                </interval_selection>
                <interval_selection>
                        <taskId>04</taskId>
                        <taskType>02</taskType>
                        <taskDescription>Markiere das erste Intervall in dem sich sowohl
                        Gesamt-Glukose als auch Fruehstueck-Glukose im kritischen
                        Level befinden. (Wo ueberlappen sich diese beiden Intervalle?)
                        </taskDescription>
                        <intervalStart>1991-01-30T05:07:00</intervalStart>
                        <intervalEnd>1991-01-30T09:04:00</intervalEnd>
                        <tolerance>14400000</tolerance>
                        <!-- 4 hours -->
                </interval_selection>
                <!-- 3 -->
                <interval_selection>
                        <taskId>05</taskId>
                        <taskType>03</taskType>
```

```
        <taskDescription>Abend-Glukose: Markiere das zweite Intervall
        eines erhoehten Levels.</taskDescription>
        <intervalStart>1991-02-05T16:59:00</intervalStart>
        <intervalEnd>1991-02-07T08:46:00</intervalEnd>
        <tolerance>14400000</tolerance>
        <!-- 4 hours -->
</interval_selection>
<interval_selection>
        <taskId>06</taskId>
        <taskType>03</taskType>
        <taskDescription>Mittag-Glukose: Markiere das erste Intervall
        eines normalen Levels.</taskDescription>
        <intervalStart>1991-02-02T03:36:00</intervalStart>
        <intervalEnd>1991-02-03T09:59:00</intervalEnd>
        <tolerance>14400000</tolerance>
        <!-- 4 hours -->
</interval_selection>
<!-- 4 -->
<choice_selection>
        <taskId>07</taskId>
        <taskType>04</taskType>
        <taskDescription>Fruehstueck-Glukose: Ist der erste Glukose-Level
        hoeher/niedriger/gleich als der dritte Glukose Level?
        </taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>hoeher</possibleAnswer>
                <possibleAnswer>gleich</possibleAnswer>
                <possibleAnswer>niedriger</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>gleich</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<choice_selection>
        <taskId>08</taskId>
        <taskType>04</taskType>
        <taskDescription>Mittag-Glukose: Ist der erste Glukose-Level
        hoeher/niedriger/gleich als der dritte Glukose Level?
        </taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>hoeher</possibleAnswer>
                <possibleAnswer>gleich</possibleAnswer>
                <possibleAnswer>niedriger</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>hoeher</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<!-- 5 -->
<choice_selection>
        <taskId>09</taskId>
        <taskType>05</taskType>
        <taskDescription>Welche Glukose Variable hat den am laengsten
        dauernden kritischen Glukose Level?</taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Gesamt</possibleAnswer>
                <possibleAnswer>Fruehstueck</possibleAnswer>
```

```
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>Fruehstueck</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<choice_selection>
        <taskId>10</taskId>
        <taskType>05</taskType>
        <taskDescription>Welche Glukose Variable hat den am
        laengsten dauernden normalen Glukose Level?</taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Gesamt</possibleAnswer>
                <possibleAnswer>Fruehstueck</possibleAnswer>
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>Abend</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<!-- 6   -->
<choice_selection>
        <taskId>11</taskId>
        <taskType>06</taskType>
        <taskDescription>Welche Glukose Variable hat das haeufigste
        Auftreten von kritischen Glukose Levels?</taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Gesamt</possibleAnswer>
                <possibleAnswer>Fruehstueck</possibleAnswer>
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>Gesamt</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<choice_selection>
        <taskId>12</taskId>
        <taskType>06</taskType>
        <taskDescription>Welche Glukose Variable hat am wenigsten
        Auftritte von leicht erhoehten Glukose Levels?</taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Gesamt</possibleAnswer>
                <possibleAnswer>Fruehstueck</possibleAnswer>
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>Fruehstueck</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<quantitative>
        <taskId>13</taskId>
```

```
        <taskType>12</taskType>
        <taskDescription>Fruehstueck-Glukose: Finde den hoechsten
        gemessenen Wert im ersten kritischen Glukose Level und
        schreibe den Wert in das Eingabefeld.</taskDescription>
        <taskInstruction/>
        <unit>mg/dl</unit>
        <isInteger>false</isInteger>
        <correctValue>295.0</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<quantitative>
        <taskId>14</taskId>
        <taskType>12</taskType>
        <taskDescription>Gesamt-Glukose: Finde den niedrigsten
        gemessenen Wert im ersten normalen Glukose Level und
        schreibe den Wert in das Eingabefeld.</taskDescription>
        <taskInstruction/>
        <unit>mg/dl</unit>
        <isInteger>false</isInteger>
        <correctValue>81.0</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<quantitative>
        <taskId>15</taskId>
        <taskType>08</taskType>
        <taskDescription>Welchen Wert hat der naechste gemessene
        Datenpunkt von Mittag-Glukose nachdem Gesamt-Glukose
        den kritischen Level das erste mal verlaesst? Schreibe den
        Wert in das Eingabefeld.</taskDescription>
        <taskInstruction/>
        <unit>mg/dl</unit>
        <isInteger>false</isInteger>
        <correctValue>162.0</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<quantitative>
        <taskId>16</taskId>
        <taskType>08</taskType>
        <taskDescription>Welchen Wert hat der naechste gemessene
        Datenpunkt von Abend-Glukose nachdem Fruehstueck-Glukose
        den erhoehten Level das erste mal verlaesst?
        Schreibe den Wert in das Eingabefeld.</taskDescription>
        <taskInstruction/>
        <unit>mg/dl</unit>
        <isInteger>false</isInteger>
        <correctValue>138.0</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<!-- 9  -->
<quantitative>
        <taskId>17</taskId>
        <taskType>09</taskType>
        <taskDescription>Mittag-Glukose: Wieviele Messpunkte
        beinhaltet das erste normale Intervall.</taskDescription>
        <taskInstruction/>
        <unit>Messpunkte</unit>
        <isInteger>true</isInteger>
        <correctValue>1</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<quantitative>
        <taskId>18</taskId>
```

```
        <taskType>09</taskType>
        <taskDescription>Fruehstueck-Glukose: Wieviele Messpunkte
        beinhaltet das erste leicht erhoehte Intervall.
        </taskDescription>
        <taskInstruction/>
        <unit>Messpunkte</unit>
        <isInteger>true</isInteger>
        <correctValue>1</correctValue>
        <tolerance>0.1</tolerance>
</quantitative>
<!-- 10         -->
<interval_selection>
        <taskId>19</taskId>
        <taskType>10</taskType>
        <taskDescription>Fruehstueck-Glukose: Welches kritische
        Intervall beinhaltet am meisten Messpunkte?Markiere das
        gesamte kritische Intervall.</taskDescription>
        <intervalStart>1991-01-29T23:20:00</intervalStart>
        <intervalEnd>1991-02-01T22:46:00</intervalEnd>
        <tolerance>14400000</tolerance>
        <!-- 4 hours -->
</interval_selection>
<interval_selection>
        <taskId>20</taskId>
        <taskType>10</taskType>
        <taskDescription>Gesamt-Glukose: Welches kritische Intervall
        beinhaltet am meisten Messpunkte?Markiere das gesamte
        kritische Intervall.</taskDescription>
        <intervalStart>1991-02-08T20:20:00</intervalStart>
        <intervalEnd>1991-02-10T08:28:00</intervalEnd>
        <tolerance>14400000</tolerance>
        <!-- 4 hours -->
</interval_selection>
<choice_selection>
        <taskId>21</taskId>
        <taskType>11</taskType>
        <taskDescription>Fruehstueck-, Mittag-, Abend-Glukose:
        Vergleiche die hoechsten gemessenen Werte im
        jeweils letzten kritischen Interval dieser Variablen.
        Welche Variable besitzt dabei den hoechsten Wert?
        </taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Fruehstueck</possibleAnswer>
                <possibleAnswer>Mittag</possibleAnswer>
                <possibleAnswer>Abend</possibleAnswer>
        </possibleAnswers>
        <correctAnswers>
                <correctAnswer>Fruehstueck</correctAnswer>
        </correctAnswers>
        <singleChoice>true</singleChoice>
</choice_selection>
<choice_selection>
        <taskId>22</taskId>
        <taskType>11</taskType>
        <taskDescription>Fruehstueck-, Mittag-, Abend-Glukose:
        Vergleiche  die niedrigsten gemessenen Werte im jeweils
        ersten erhoehten Interval dieser Variablen. Welche Variable
        besitzt dabei den kleinsten Wert?</taskDescription>
        <taskInstruction/>
        <possibleAnswers>
                <possibleAnswer>Fruehstueck</possibleAnswer>
```

```
                        <possibleAnswer>Mittag</possibleAnswer>
                        <possibleAnswer>Abend</possibleAnswer>
                </possibleAnswers>
                <correctAnswers>
                        <correctAnswer>Abend</correctAnswer>
                </correctAnswers>
                <singleChoice>true</singleChoice>
        </choice_selection>
        <!-- 07         -->
        <choice_selection>
                <taskId>23</taskId>
                <taskType>07</taskType>
                <taskDescription>Welche Glukose-Variable hat gerade einen
                Abfall wenn Gesamt-Glukose das dritte Mal in den kritischen
                Level eintritt? Vergleiche dabei nur die unmittelbar davor
                und danach liegenden gemessenen Datenpunkte.
                </taskDescription>
                <taskInstruction/>
                <possibleAnswers>
                        <possibleAnswer>Fruehstueck</possibleAnswer>
                        <possibleAnswer>Mittag</possibleAnswer>
                        <possibleAnswer>Abend</possibleAnswer>
                </possibleAnswers>
                <correctAnswers>
                        <correctAnswer>Abend</correctAnswer>
                </correctAnswers>
                <singleChoice>false</singleChoice>
        </choice_selection>
        <choice_selection>
                <taskId>24</taskId>
                <taskType>07</taskType>
                <taskDescription>Welche Glukose-Variable hat gerade einen
                Anstieg wenn Mittag-Glukose das zweite Mal den erhoehten
                Level verlaesst?Vergleiche dabei nur die unmittelbar davor
                und danach liegenden gemessenen Datenpunkte.
                </taskDescription>
                <taskInstruction/>
                <possibleAnswers>
                        <possibleAnswer>Gesamt</possibleAnswer>
                        <possibleAnswer>Fruehstueck</possibleAnswer>
                        <possibleAnswer>Abend</possibleAnswer>
                </possibleAnswers>
                <correctAnswers>
                        <correctAnswer>Fruehstueck</correctAnswer>
                </correctAnswers>
                <singleChoice>false</singleChoice>
        </choice_selection>
    </tasks>
</taskList>
```