TimeCleanser: A Visual Analytics Approach for Data Cleansing of Time-Oriented Data

Theresia Gschwandtner
Wolfgang Aigner
Silvia Miksch
Institute of Software Technology & Interactive Systems (ISIS)
Vienna University of Technology, Austria
{lastname}@cvast.tuwien.ac.at

Johannes Gärtner
XIMES GmbH
Vienna, Austria
http://www.ximes.com/en/
gaertner@ximes.com

Simone Kriglstein
Margit Pohl, Nik Suchy
Institute for Design & Assessment of Technology (IGW)
Vienna University of Technology, Austria
{lastname}@cvast.tuwien.ac.at

ABSTRACT

Poor data quality leads to unreliable results of any kind of data processing and has profound economic impact. Although there are tools to help users with the task of data cleansing, support for dealing with the specifics of time-oriented data is rather poor. However, the time dimension has very specific characteristics which introduce quality problems, that are different from other kinds of data. We present TimeCleanser, an interactive Visual Analytics system to support the task of data cleansing of time-oriented data. In order to help the user to deal with these special characteristics and quality problems, TimeCleanser combines semi-automatic quality checks, visualizations, and directly editable data tables. The evaluation of the TimeCleanser system within a focus group (two target users, one developer, and two Human Computer Interaction experts) shows that (a) our proposed method is suited to detect hidden quality problems of time-oriented data and (b) that it facilitates the complex task of data cleansing.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces And Presentation**]: User Interfaces—*User-centered design*; I.3.6 [**Computer Graphics**]: Methodology and Techniques—*Graphics data structures and data types*

General Terms

Design Study

Keywords

data cleansing, time-oriented data, Visual Analytics, data quality

i-KNOW '14, September 16 - 19 2014, Graz, Austria

http://dx.doi.org/10.1145/2637748.2638423

1. INTRODUCTION

Results of any kind of data analysis can only be as good as the quality of the data at hand. Data quality control can be divided into (1) *data profiling*, (2) *data wrangling*, and (3) *data cleansing*. *Data profiling* is the identification and communication of data quality problems. *Data wrangling* is the transformation of data into a structure that is suited for subsequent processing, such as splitting of variables, removing unnecessary items (e.g., summary rows), but also merging of data from different sources. The actual task of *data cleansing* – also called data cleaning, or data scrubbing – is the process of correcting dirty data. The majority of types of data quality problems require intervention by a human domain expert to be cleansed [9], which makes this topic an ideal match for Visual Analytics (VA). Data quality control is important in both science and industry and often requires considerable effort and has profound economic impact.

The special characteristics of time-oriented data are hardly considered in existing quality control approaches. However, time is an important data dimension with distinct characteristics which introduce specific problems (e.g., implausible amount of instances at a given time of day), and thus, requires special consideration.

In the design study presented in this paper we describe TimeCleanser, a VA approach for the task of data cleansing with a special focus on time-oriented data (see Section 3). First we give an outline of related work in Section 2 and we outline our design requirements in Section 3.2. We implemented TimeCleanser, a novel approach for data cleansing of time-oriented data based on a combination of well-known VA techniques (i.e., automated quality checks, visualizations, and interactive data tables). We evaluated our design in two formal studies; in this paper we focus on the second study, a focus group evaluation involving target users as well as Human Computer Interaction (HCI) experts (see Section 4). We document insights found by the target users (see Section 4.1), as well as lessons learned and derived design principles (see Section 5). Finally, we discuss and sum up the main results of our work (see Section 6).

2. RELATED WORK

On top of different taxonomies of data quality problems [15, 9, 11, 13, 1, 5], a catalog of data quality problems focusing specifically on time-induced problems was built [4]. Such

collections of data quality problems form a basis for *data profiling*. Profiler [7] is an analysis tool which uses data mining methods to automatically identify quality problems in tabular data sets and automatically composes coordinated visualizations for communicating these issues. The Data Profiler from Talend [21] assesses the quality of a data set by using predefined indicators, such as simple statistics of the data set, counting empty cells or duplicates, or by user defined indicators. Results (e.g., statistics) are presented mainly by bar charts and quality problems are highlighted in the corresponding data tables. However, the provided functionality for editing the data tables, such as merging two columns, is limited to only a few.

Potter's Wheel [16] offers a spreadsheet-like interface for interactively specifying data transformations. It supports transforms such as add, crop, copy, fold, and merge. Wrangler [6] not only supports the direct manipulation of data tables but also supports the user with the automatic inference of relevant transforms, such as suggesting to fill empty cells by copying values from above. It uses knowledge about semantic data types (e.g., geographical location) to judge validity of data entries and to improve suggestions for transformations and conversions.

AJAX [3] provides a framework to model the logic of a data cleansing program, by means of expressive and declarative language specifications. It supports mapping, matching, clustering and merging transformations. OpenRefine [17] (formerly GoogleRefine) is designed to apply transformations over many cells in bulk and extending the data with more data from other sources. It supports transformations such as search and replace, remove rows, as well as transpose columns into rows. Moreover, it supports user-defined checks such as 'find entries in the customer's name column with over 50 characters'. Another nice feature is the use of similarity metrics to detect approximate duplicates.

We took a close look at Wrangler [6] and AJAX [3] and couldn't find any time-specific operations. Other approaches partly consider the special characteristics of quality problems introduced by time-oriented data. For instance, most tools support checks for correct data and time formats [21, 16, 17]. In addition, Profiler [7] provides means for univariate outlier detection in time-series data as well as a time-series chart, that supports grouping the data by time spans (e.g., days, months, years), and OpenRefine [17] provides an interactive chart to filter for specific temporal ranges. However, characteristics such as length of intervals, exact timing, gaps and overlaps within time records, plausible temporal ranges, or plausible data values for given temporal ranges demand special consideration (see [4] for a comprehensive list of time-induced data quality problems). There is one approach with a special focus on the visual-interactive preprocessing of time series data [2]. It is a system for the interactive design and control of a time series preprocessing pipeline. It features data reduction, data normalization, data segmentation, descriptors, and similarity measures. This approach is focused on composing time series preprocessing pipelines, and thus, data cleansing operations such as removing missing values, merging identical time stamps, and getting rid of noise and outliers by smoothing the curve (i.e., calculating a moving average) are applied at the whole time series at once. In contrast to that, we propose not only means for applying corrections to the whole time series at once, but also means for scrutinizing the time-oriented data set to identify and correct single suspicious entries as well as more complex semantic problems (e.g., implausible weekly profiles).

Syntax Checks	Correct column names
	Each row contains the same number of columns
	Correct table structure
	Correct date/number/text format
	Empty cells *(i.e., columns can/must not contain empty cells and column that can be all empty)*
	Valid entries *(from a user-defined list of valid entries)*
	Text-delimiters
	White-space *(white-space before/after/within entries)*
	Duplicates
Time Checks	**Valid overall temporal range**
	Durations/interval length *(i.e., strictly defined length or plausibility of not strictly defined length)*
	Missing time point or interval *(i.e., no gaps/some gaps allowed; obligatory time gaps)*
	Entries for different IDs cover same temporal range *(e.g., entries of department A and B both cover March 2012)*
Time-Oriented Value Checks	**Valid minimum and maximum values within a given temporal range**
	Values which do not change for too long *(i.e., any value/a specific value should not outlast a user-defined duration)*
	Dependencies between columns *(e.g., if column 'Substance' contains entry 'saline solution', column 'Unit' must contain 'ml')*
	Dependencies over multiple rows *(e.g., for each identifier there should be three rows with predefined entries in column 'Unit')*
	Valid timing of values *(e.g., minutes divisible by five)*
	Valid value sequences
	Valid intervals between subsequent values
Multiple Data Sets	**Cover same temporal range**
	Contain same set of identifiers
	Have same table structure
	Have same data formats
	Contain intervals of equal length
	Contain time stamps of same precision
Visualizations	**Overview of values over time** (see Figure 4, top)
	Difference plot of subsequent data values (see Figure 4, center)
	Interval length as bars over time (see Figure 4, bottom)
	Heatmap of interval lengths and data values (see Figure 3)
	Difference between numeric data value and interval length

Table 1: TimeCleanser: quality checks and visualizations (items in bold are quality problems specifically induced by time-oriented data).

3. TIMECLEANSER

Due to a lack of approaches that consider the various data quality problems induced by time-oriented data, we have designed TimeCleanser (see Figure 1), a VA approach for detecting and correcting data quality problems with a special focus on time-oriented data. We derived our requirements from [4], a taxonomy of time-oriented data quality problems. Following a user-centered design process and effective models of design studies ([12, 19]), we built and gathered

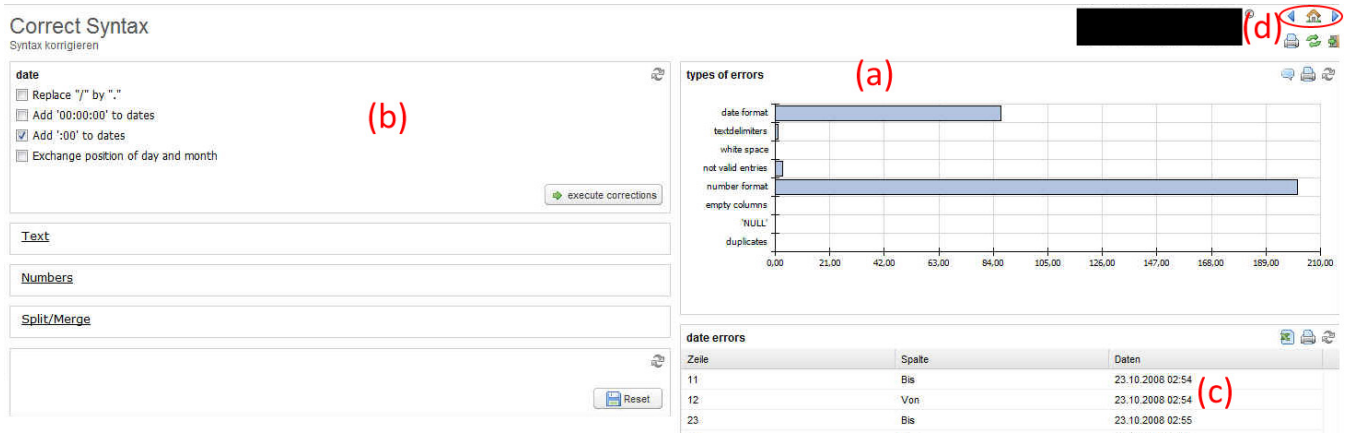

Figure 1: TimeCleanser: a screenshot of automatic means for correcting syntax errors. (a) Bar chart of different types of identified syntax errors. (b) Facet to select corrections for invalid text entries (facets for corrections of date entries, number entries, and other kinds of corrections can be found above and below the text facet). (c) Erroneous text entries in tabular form for manual corrections of the raw values. (d) Wizard-like navigation through the sequential cleansing process.

feedback on a series of prototypes. We worked at our partner company site for over three years to get familiar with the data analysts' tasks and workflows. During this time, we refined our list of features with every feedback cycle. We conducted two formal evaluation sessions: one was concerned with the intuitive alignment and composition of dependent and independent user interface panels, where a group of five participants had to solve a number of tasks (concerning the arrangements of these panels) and results were drawn from observation and interviews. However, in this paper we focus on the second evaluation, which was concerned with the usefulness of TimeCleanser (see Section 4).

3.1 Development Team and Collaborators

We tightly collaborated with a software company in the field of shift scheduling. On the one hand, this company is concerned with consulting in workforce management: They get data from companies, analyze the existing processes, forecast for the next planning period, and determine the optimal demand. On the other hand, the company is concerned with queue management, i.e., real-time measurement of queue length and queue waiting time. The CEO of this company was involved in the TimeCleanser project right from the start, leading the problem analysis with his business experience and giving frequent informal feedback during design and implementation of the prototype. Two data analysts from this company were involved by attending feedback sessions and evaluating the prototype within a focus group (see Section 4). Moreover, the software development team of the company (a group of seven developers) was involved in the design and implementation by attending regular presentations and feedback sessions. Besides the cooperation with the company, a team of nine VA experts guided the design and implementation of the prototype and a team of three HCI experts led two formal evaluation sessions.

3.2 Problem Analysis and Design Rationales

The taxonomy [4] of time-induced quality problems comprises a list of 39 different problems, ranging from simple syntactical problems such as 'wrong date format' or 'times outside raster' to semantic problems such as 'implausible weekly profile'. We broke down these quality problems to specific tests on problems specifically induced by time-oriented data and augmented them with general tests that also need to be tackled (e.g., syntactical issues, semantic dependencies between table entries). This list of data quality tests (see Table 1) served as a reference of design requirements for TimeCleanser. Many checks and data formats can be easily specified (e.g., the intended structure of the table and tests regarding the syntax of the data). Thus, we support the user by providing predefined tests to detect these issues. Figure 2 shows a small subset of these predefined tests. However, there are also numerous checks which cannot be defined in this precise and formal way. For instance, a data set containing activities of ambulances may contain a driving time of multiple days which would be obviously wrong, but it is not clear how to set numerical borders to define which driving times are valid and which are not. There are many other checks that cannot easily be defined by concrete numbers, such as 'plausibility of changes of subsequent values', 'nearly identical entries', 'plausibility of sales on different weekdays', etc. To detect these kinds of quality issues we provide a number of visualizations representing different aspects of the data, helping the user to spot these implausible values and outliers (see Table 1, Figure 3, and Figure 4). To facilitate the task of data cleansing, we provide a wizard-like guide through the different cleansing steps (see the navigation symbols in Figure 1 (d)). Figure 1 shows one of many cleansing steps, i.e., automatic means for correcting syntax errors of date formats. In addition, we provide links to jump back and forth between important steps. To ensure the quality and usefulness of the design of TimeCleanser, we conducted a focus group evaluation as described in the following section.

4. EVALUATION

In order to develop an in-depth understanding of the environment and work practice, we worked at the company site for over three years and constantly refined the design and

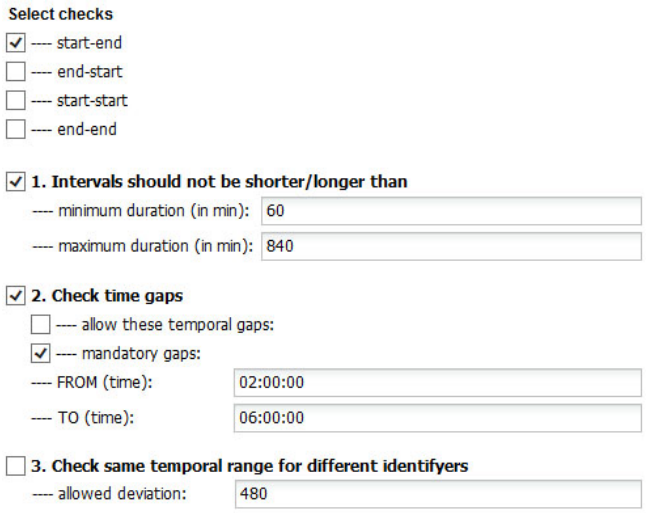

Figure 2: Interface for the definition of temporal constraints: select start and end of the interval of interest and define temporal constraints for these intervals: 1. minimum and maximum duration, 2. test if there are missing intervals, including a possibility to allow missing intervals within a certain time span or define that there must not be any intervals within certain times, and 3. test if entries for each ID cover the same overall time span. Different combinations of these definitions cover a broad range of time-oriented quality checks.

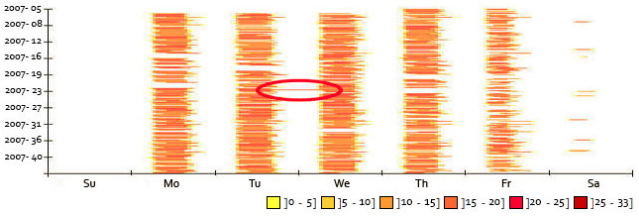

Figure 3: Working hours visualized in a heatmap (weekdays are plotted on the x-axis, days are plotted on the y-axis, and the color ranging from yellow to red indicates the amount of working employees). Besides the general structure, we see a few days without working hours (maybe holidays), and sporadic shifts on Saturdays. The visual representation of the data allows for an immediate identification of an outlier working shift during the night (red circle) which may indicate an error.

Figure 4: Quality issues with the data values (km/h values) of a GPS data set (containing data of two cars, represented by different colors). In the first chart we see the raw data values laid out along a time axis with suspicious, long lasting high values in the middle. The second chart is a difference plot revealing very high velocity changes at the beginning and end of long lasting high values from the first chart – this substantiates the initial suspicion that this is an error. The third picture shows the length of the time intervals for which these values are recorded. We can see that these constant high velocity values are given for unusual long intervals (in contrast to several short intervals with the same km/h values) – another indication that these selected entries differ from the majority of data entries in this data set.

functionality of TimeCleanser. Moreover, we conducted a focus group to learn more about target users' opinions in order to identify further directions and to get feedback regarding user satisfaction with the TimeCleanser prototype. The focus group was guided in order to get answers to the following research questions: (1) Does the prototype help the target users (i.e., the analysts) in their working steps to perform data cleansing tasks? (2) Is an integration of visualizations methods useful? (3) What are the advantages and disadvantages in comparison with the data cleansing methods they have used so far? (4) For which tasks are visualization methods, common data cleansing analysis methods, and a combination of both suitable? (5) Which interaction methods for the visualizations are useful to support users' working steps to perform data cleansing tasks?

Data Sets. We decided to use two time-oriented data sets from two different domains to show that the cleansing methods are not limited to one specific domain and we used one data set for each evaluator. The first one was a data set of

our partner company, about working hours of an organization. It contained daily working hours of 33 employees which constituted about 5700 rows. The data was anonymized and preprocessed (i.e., getting rid of additional columns). Since there was no single real-life data set containing a big enough variety of quality problems, we manually introduced typical time-induced quality issues from [4] into the given real-life data set (i.e., we manually constructed a suitable test data set as, for instance, described in [18]). The second data set was about GPS data. More precisely, the set contained velocity values of two cars together with intervals in form of start and end time (i.e., date plus time). The data covered a period of two days, mostly given in five second intervals (about 4200 rows). This data set was taken from Zheng's GPS data sets [23], preprocessed (i.e., computing km/h values from GPS coordinates), and again, quality issues were introduced manually.

Sample. The focus group consisted of five participants and one skilled moderator who was familiar with the domain. We chose the participants in such a way that we had a mixture of different roles: two analysts (A1 and A2) of the company, who are the target users of TimeCleanser, one developer (in case of technical questions and for the discussion about possible further features of the prototype), two HCI experts (HCI1 and HCI2), and one moderator. This allowed us to obtain feedback and ideas for further development of the prototype from different perspectives.

Procedure. The focus group was conducted in a meeting room for a duration of about two hours. Audio and video were digitally recorded during the session. The focus group session began with an introduction of TimeCleanser. After this introduction the two scenarios (GPS data and working hours) were presented. For both scenarios the following tasks were considered: (1) remove syntax errors, (2) check interval lengths, (3) check plausibility of velocity values (GPS data set only), and (4) check validity of working hours as well as of weekly profiles (working hours data set only). Each analyst (A1 and A2) had to solve one of these scenarios within 30 minutes. While working with TimeCleanser they were observed by the HCI experts. After they finished the scenarios, topics such as a) their opinions about TimeCleanser, b) how they solved the scenarios, c) what they liked and what they disliked, d) the lack of important features, and e) the usefulness of VA and interaction techniques for data cleansing tasks, were discussed.

4.1 Results

Based on our research questions we analyzed our observations and discussion during evaluation. We identified six topics that were discussed within the focus group:

Traditional methods: Usually, the analysts utilize their in-house software solution [TIS] [22] in combination with MS Excel [10] to cleanse their data sets. However, it is much more cumbersome than doing the same data cleansing using TimeCleanser and much of the work has to be done manually. Especially semantic errors are hard to find with these traditional methods. The analysts stated that to avoid this time-consuming task they process their data sets without previous quality checks and to try to fix selected issues only when they run into troubles.

Workflow: TimeCleanser is designed to meet the workflow of the analysts. In the course of the discussion we made sure that our proposed sequence of cleansing steps (i.e., syntax first, then semantics, then time and data values; see Section 5) is compatible with the usual habits of the analysts. Moreover, the HCI experts could observe that TimeCleanser largely corresponded with the analysts' requirements regarding their working steps.

Advantages of TimeCleanser: TimeCleanser offers convenient features which enable the user to find errors in the data much faster than with traditional methods. Moreover, it enables the user to quickly get an overview of the data and its structure which fosters insights into the data beyond the task of identifying data quality issues.

Attitudes towards visualizations: The analysts stated that the visualizations present a good overview and patterns in the data can be found more easily. Various characteristics of the data as, for instance, outliers and relations can be identified easily using visualizations. In case of the working hours scenario outliers were detected by the provided heatmap visualization (see Figure 3). In case of the GPS data scenario visualizations were used as well (see Figure 4). All three visualizations in Figure 4 indicate that there is something wrong with the data. Be it that the GPS device did not send data for some time or that there are other reasons, the analyst needs to be aware of this quality issue, and it is up to her/him to confer with the customer and make decisions how to deal with it for further analysis.

> *'In case there are any recognizable patterns, i.e., systematic outliers, you are more likely to detect them with a visualization than in the data set.' (A2)* [14]

Intertwinedness of analytical and visual methods: Both analysts found the combination of analytical and visual methods beneficial. They wanted to get an overview of the data through the visualization and more detailed information through the exact values in combination with the information about errors provided by TimeCleanser. Automating the detection of easily definable quality issues (e.g., syntax errors, the violation of hard constraints, etc.) allowed the analysts to focus on more difficult quality problems, hidden in the data. Visualizations helped them to quickly spot suspicious entries, whereupon the analysts investigated these entries in more detail by filtering the data set and refining analytical methods.

> *'I think that the combination [of visualizations and analytical checks] is very useful. From the visualization you can quickly recognize the different kinds of problems.' (A1)* [14]

Negative points and possible new features: The analysts identified some drawbacks of TimeCleanser and made useful suggestions of possible new features. One of the analysts (working on the GPS scenario) was quite happy about existing interactive functionality (especially zooming), while the other analyst argued that in his scenario interactive features were not necessary. The HCI experts suggested that more interactive features would be necessary to enable the analysts to really explore the data. For instance, linking and brushing or overview and detail techniques were stated as useful. Besides this issue, which has to be addressed in future research, we list the most interesting suggestions for additional features:

- Synchronized zooming for multiple visualizations
- Linking and brushing between visualizations and data tables
- Statistics about string lengths to support the detection of outliers
- Use of wildcards and regular expressions for filter functionality
- A one-page statistical summary of the data set (e.g., minimum, maximum, average, distribution)

We believe that this evaluation session led to important insights and we will thoroughly consider the findings and improve TimeCleanser accordingly. In the following section we outline the implications for the design of a data cleansing system, which we could draw from these insights.

5. IMPLICATIONS

The whole process of design, implementation, and evaluation of TimeCleanser led us to formulate the following design principles. Quotations to back up these principles are taken from the focus group evaluation (as described in the previous section).

Design Principle 1: Data cleansing is a sequential task with loops. During our design phase we found out that following a certain sequence of cleansing steps is indicated. In particular, a given sequence of cleansing steps with back loops and branches. For the specific case of data cleansing we need to extend Shneiderman's famous Visual Information Seeking Mantra *'overview first, zoom and filter, then details-on-demand'* [20, p.337] to *'correct syntax first, assign semantic roles, overview, zoom and filter, then analysis and details-on-demand'*. This adds important preprocessing steps that are necessary when you cannot rely on the quality of your data. Our proposed sequence of data cleansing steps also fits Keim's Visual Analytics mantra well: *'analyse first – show the important – zoom, filter and analyse further – details on demand'* [8, p.82]. The application of automatic and user-defined syntax checks is consistent with 'analyse first', showing the results and highlighting entries with identified quality problems is consistent with 'show the important', zoom and filter (as described below), and the investigation of the data set to detect less obvious problems (as described in 'Then data analysis and details-on-demand' below) is consistent with 'analyse further – details on demand'. For the task of data cleansing of time-oriented data, however, we again need to add 'correct syntax first', 'assign semantic roles', and 'overview' to the beginning of Keim's Visual Analytics mantra.

1. Correct syntax first: A prerequisite of any data analysis is the correct syntactical format of the data entries. For instance, if a date entry is not in correct date format, the temporal information of the data set (e.g., the sequence of values, information about durations, etc.) cannot be processed, visualized, or analyzed. Thus, as a first step, TimeCleanser automatically derives the syntactic structure of the data set which then can be edited by the user.

2. Assign semantic roles: In order to process time-oriented data, the semantic roles of different data columns need to be defined. In particular, the user needs to declare which column contains 'from' values, which column contains 'to' values (or timestamps, or durations, respectively), and which column contains the associated data values that are to be analyzed. Moreover, we advise to define a 'differentiator' column. This column contains IDs which differentiate groups of data which need to be considered for specific time-oriented quality checks (e.g., overlapping time intervals within one department may not be valid and, thus, should be detected as errors, while overlaps may be valid for different departments). These semantic roles are assigned semi-automatically with the help of user input, since they may be ambiguous.

3. Overview: The previous step (i.e., correcting the syntax of the data set and the definition of different semantic roles) allows for the correct visualization of the important data. We provide an overview visualization where the values of the currently selected 'data value' column are laid out along a time axis. Besides getting a better understanding for the data at hand, suspicious entries, like extreme outliers, can already be spotted.

4. Zoom and filter: On the one hand, we provide means to filter for specific temporal ranges. On the other hand, we provide filters to investigate only specific data values. We observed two ways of using the filter: (1) The analyst knows right from the start that she/he wants to analyze only specific entries: she/he may be interested in only a specific temporal range (e.g., sales data from the last month). (2) Filters are applied and removed multiple times during data analysis: she/he wants to filter for specific departments or she/he wants to analyze specific temporal ranges or values in more detail.

5. Then data analysis and details-on-demand: To support the data quality analysis step, we provide different automatic as well as user-defined checks in combination with visualizations. The different steps necessary for this kind of data analysis have an inherent order:

a **Time Values**: Depending on the data set, the temporal values may need to meet certain constraints. For instance, all intervals need to have the same length, no missing intervals or times, some time gaps may be obligatory (e.g., no working hours during the night), times need to be within a plausible temporal range (e.g., the last 10 years), intervals should not overlap, etc.

b **Data Values**: When looking at the validity of the actual data values it is important that the time values are already corrected or modified to meet the given constraints. For instance, if sales data is given for some days in an hourly manner and for other days they are given for the whole day at once, numbers are hard to compare and the time intervals have to be normalized before analysts can investigate the validity of these values.

c **Value Sequences**: In certain cases the sequence of data values and their exact timing are important. For instance, we were processing a data set about workflows at counter desks where the employee had to press different buttons for 'being ready to receive a customer', 'being busy', 'start' and 'end' of service, etc., which had to follow a given sequence. Moreover, data generated by a server had to be checked for correct timing (e.g., every five minutes, only whole minutes,

etc.). Checking the validity of such sequences includes both correct time values and correct data values.

d **Multiple Data Sets**: Quality problems that may occur between multiple data sets can be tackled only after resolving the issues of each single data set. Issues of multiple data sets include different table structures, references between tables (e.g., two documents cover the same department IDs), heterogeneity of scales (e.g., different interval length), etc.

The sequence of cleansing steps above is – of course – only a rough outline of the actual procedure of data cleansing. It is very likely that the analyst has to jump back and forth between important steps or omit some steps. However, following a certain pipeline (syntax first, then time entries, and finally data values) seems mandatory.

Design Principle 2: Complex quality problems are best spotted with visualizations. Some quality problems can be well detected by formal rules but when it comes to judge the validity of data values which do not violate any hard constraints, it is difficult to define such quality checks in a computer-executable way. For instance, you can estimate a definite upper limit of velocity values, but surrounding values are also important to judge the plausibility of one entry. Defining such exact and possibly complex rules is often not practical or even impossible in a completely formal and automated manner and chances are high to miss some quality problems when not seeing the whole picture.

> *'You get a picture of the data set, not only of erroneous entries, but also of how the data looks like and how it should look like.' (A1)* [14]

Design Principle 3: Visualizations and raw data tables are complementary. When analysts identify suspicious data with the help of visualizations, they tend to jump back and forth between the visualization and the corresponding data tables to see the raw values as well as the surrounding raw values. This also serves to double-check the findings. Thus, it is important to put visualizations and raw data tables side by side, or at least make them easily accessible.

Design Principle 4: Algorithmic means are suited to identify precisely definable errors. On the other hand, there are quality checks that are easily specified in a formal way, for instance, intervals should not be longer than 10 hours (supposing some working time regulations). In this case an easy formal specification in combination with automatic quality checks and error corrections are preferred.

> *'The means for automatic corrections are very useful and allow for an immediate correction of typical errors.' (A1)* [14]

Design Principle 5: Original data needs to be preserved. During the whole data cleansing process, it is crucial that original data is preserved. In particular, even if analysts decide that a certain data value is not valid, they cannot be completely certain. Thus, they need to be able to correct and process the data right away, but they also need to confer with customers about the original data and their decisions at a later date, and – in case – undo these changes quickly.

> *'You need to be very cautious when changing data coming from customers. Changing data within the visualization would only make sense if you can still retrace these changes. [...] In a way that changes are accepted conditionally.' (A2)* [14]

6. DISCUSSION AND CONCLUSION

Originally, we were confronted with the problem of effectively supporting the data cleansing process of time-oriented data including not only syntactic and quantifiable problems but also supporting the user to judge the plausibility of diverse aspects of the data. We derived the design requirements of TimeCleanser from a taxonomy of 39 time-induced data quality problems [4] and tested the prototype with data from different domains: data of workforce management (including data sets of more than 10 million records, as well as data from different sources), sensor data of queue management (i.e., the analysis of complex sequences and correct timing of data values), as well as GPS data. TimeCleanser provides means (i.e., automatic, user-definable, and visual means) for identifying and correcting suspicious entries, with a special focus on problems that may be introduced by time-oriented data (scales, precisions, homogeneity of intervals, filter for specific temporal ranges, times when gaps are not allowed vs. times when gaps are obligatory, plausible sequence of values, and many more). The combination of automated and visual methods helps to speed up the detection and elimination of easily definable problems, allowing the user to concentrate on quality issues where human judgment is needed.

Some lessons learned (and assumptions confirmed) from design, implementation, and evaluation are:

- In cases which are more easily defined in terms of numbers, analytical methods are preferred (e.g., sensor data must appear exactly every 5 seconds)
- In other cases visualizations are superior to analytical methods (e.g., for the detection of implausible working shifts)
- The two analysts appreciated the use of visualizations as an interactive analysis tool
- Efficient connection of visualizations to raw data and a side by side display of both is important

According to the findings of the evaluation, we believe that TimeCleanser is an important step towards dealing with quality problems introduced by time-oriented data. In addition, our findings yield the following contributions utilizing a VA approach:

- Systematic list of data quality checks (see Table 1)
- Sequence of cleansing steps (i.e., syntax, semantic roles, time values, data values, value sequences, matching multiple data sets)
- Implications for the design of a data cleansing support (with special focus on time-oriented data)
- Prototypical design and implementation in very close collaboration with end-users
- Results of the evaluation demonstrate the need of visualizations for specific cleansing tasks

The evaluation with two data analysts (the intended target users), two HCI experts, one developer, and one moderator can only yield tentative conclusions. Nevertheless, some of the results seem to be quite clear. The analysts appreciated the TimeCleanser prototype. They mentioned that it had more functionalities, was faster, and enabled them to analyze the data and errors in detail. The analysts also clearly stated that the combination of analytical methods with visualizations, a VA approach, was highly beneficial although they traditionally did not use visualizations for data cleansing. On the other hand, interactions seem to be a more controversial issue depending on the tasks, which certainly demands further research.

7. ACKNOWLEDGMENTS

We wish to thank Werner Marschitz, Thomas Tippl, and the developer team of XIMES for their cooperation, helpful suggestions, time, and thoroughly investigation of the various stages of the TimeCleanser system. Moreover, the research leading to these results has received funding from the Centre for Visual Analytics Science and Technology CVAST, funded by the Austrian Federal Ministry of Science, Research, and Economy in the exceptional Laura Bassi Centres of Excellence initiative (#822746).

8. REFERENCES

[1] J. Barateiro and H. Galhardas. A survey of data quality tools. *Datenbankspektrum*, 14:15–21, August 2005.

[2] J. Bernard, T. Ruppert, O. Goroll, T. May, and J. Kohlhammer. Visual-Interactive preprocessing of time series data. In *Proc. of SIGRAD 2012: Interactive Visual Analysis of Data*, pages 39–48, November 2012.

[3] H. Galhardas, D. Florescu, D. Shasha, and E. Simon. AJAX: An extensible data cleaning tool. *SIGMOD Record*, 29(2):590–596, June 2000.

[4] T. Gschwandtner, J. Gärtner, W. Aigner, and S. Miksch. A taxonomy of dirty time-oriented data. In G. Quirchmayr, J. Basl, I. You, L. Xu, and E. Weippl, editors, *Multidisciplinary Research and Practice for Information Systems*, LNCS 7465, pages 58–72. Springer, Berlin/Heidelberg, Germany, 2012.

[5] R. P. Jagadeesh Chandra Bose, R. S. Mans, and W. M. P. van der Aalst. Wanna improve process mining results? It's high time we consider data quality issues seriously. In *Proc. of the IEEE Symposium on Computational Intelligence and Data Mining (CIDM 2013)*, pages 127–134, April 2013.

[6] S. Kandel, A. Paepcke, J. Hellerstein, and J. Heer. Wrangler: Interactive visual specification of data transformation scripts. In *Proc. of the ACM Conference Human Factors in Computing Systems (CHI 2011)*, pages 3363–3372, May 2011.

[7] S. Kandel, R. Parikh, A. Paepcke, J. Hellerstein, and J. Heer. Profiler: Integrated statistical analysis and visualization for data quality assessment. In *Proc. of the International Working Conference on Advanced Visual Interfaces (AVI'12)*, pages 547–554, May 2012.

[8] D. A. Keim, F. Mansmann, J. Schneidewind, J. Thomas, and H. Ziegler. Visual analytics: Scope and challenges. In S. J. Simoff, M. H. Böhlen, and A. Mazeika, editors, *Visual Data Mining: Theory, Techniques and Tools for Visual Analytics*, LNCS 4404, pages 76–90. Springer, Berlin/Heidelberg, Germany, 2008.

[9] W. Kim, B.-J. Choi, E.-K. Hong, S.-K. Kim, and D. Lee. A taxonomy of dirty data. *Data Mining and Knowledge Discovery*, 7(1):81–99, January 2003.

[10] Microsoft. Excel. office.microsoft.com/en-us/excel/ (accessed: 2014-04-17).

[11] H. Müller and J.-C. Freytag. HUB-IB-164. Problems, methods, and challenges in comprehensive data cleansing. Technical report, Humboldt University Berlin, 2003.

[12] T. Munzner. A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928, November 2009.

[13] P. Oliveira, F. Rodrigues, and P. Henriques. A formal definition of data quality problems. In *Proc. of the International Conference on Information Quality (MIT IQ Conference)*, November 2005.

[14] Original German quotes of the focus group session. Attached to the submission as supplemental material. ieg.ifs.tuwien.ac.at/~gschwandtner/material/quotes.pdf (accessed: 2014-04-17).

[15] E. Rahm and H.-H. Do. Data cleaning: Problems and current approaches. *IEEE Bulletin of the Technical Committee on Data Engineering*, 23(4):3–13, March 2000.

[16] V. Raman and J. M. Hellerstein. Potter's wheel: An interactive data cleaning system. In *Proc. of the 27th International Conference on Very Large Data Bases*, pages 381–390, September 2001.

[17] Random Developers. OpenRefine. http://openrefine.org/ (accessed: 2014-04-17).

[18] J. Scholtz, M. A. Whiting, C. Plaisant, and G. Grinstein. A reflection on seven years of the VAST challenge. In *Proc. of the 2012 BELIV Workshop: Beyond Time and Errors - Novel Evaluation Methods for Visualization*, pages 13:1–13:8, October 2012.

[19] M. Sedlmair, M. Meyer, and T. Munzner. Design study methodology: Reflections from the trenches and the stacks. *IEEE Trans. Visualization and Computer Graphics*, 18(12):2431–2440, October 2012.

[20] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proc. of the 1996 IEEE Symposium on Visual Languages*, pages 336–343, September 1996.

[21] Talend. Profiler. http://www.talend.com/ (accessed: 2014-04-17).

[22] XIMES GmbH. Time Intelligence Solutions [TIS]. www.ximes.com/en/software/products/tis/ (accessed: 2014-04-17).

[23] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma. Mining interesting locations and travel sequences from GPS trajectories. In *Proc. of the International Conference on World Wild Web (WWW 2009)*, pages 791–800, April 2009.