A Taxonomy of Dirty Time-Oriented Data

Theresia Gschwandtner, Johannes Gärtner, Wolfgang Aigner, Silvia Miksch

A Taxonomy of Dirty Time-Oriented Data

Theresia Gschwandtner[1], Johannes Gärtner[2], Wolfgang Aigner[1], and Silvia Miksch[1]

[1] Institute of Software Technology and Interactive Systems (ISIS)
Vienna University of Technology
Favoritenstrasse 9-11/188, A-1040 Vienna, Austria
{gschwandtner,aigner,miksch}@cvast.tuwien.ac.at
http://www.cvast.tuwien.ac.at/
[2] XIMES GmbH
Hollandstraße 12/12, A-1020 Vienna, Austria
gaertner@ximes.com
http://www.ximes.com/en/

Abstract. Data quality is a vital topic for business analytics in order to gain accurate insight and make correct decisions in many data-intensive industries. Albeit systematic approaches to categorize, detect, and avoid data quality problems exist, the special characteristics of time-oriented data are hardly considered. However, time is an important data dimension with distinct characteristics which affords special consideration in the context of dirty data. Building upon existing taxonomies of general data quality problems, we address 'dirty' time-oriented data, i.e., time-oriented data with potential quality problems. In particular, we investigated empirically derived problems that emerge with different types of time-oriented data (e.g., time points, time intervals) and provide various examples of quality problems of time-oriented data. By providing categorized information related to existing taxonomies, we establish a basis for further research in the field of dirty time-oriented data, and for the formulation of essential quality checks when preprocessing time-oriented data.

Keywords: dirty data, time-oriented data, data cleansing, data quality, taxonomy

1 Introduction

Dirty data leads to wrong results and misleading statistics [1]. This is why data cleansing – also called data cleaning, data scrubbing, or data wrangling – is a prerequisite of any data processing task. Roughly speaking, data cleansing is the process of detecting and correcting dirty data (e.g., duplicate data, missing data, inconsistent data, and simply erroneous data including data that do not violate any constraints but still are wrong or unusable) [2]. Dirty data include errors and inconsistencies in individual data sources as well as errors and inconsistencies

when integrating multiple sources. Data quality problems may stem from different sources such as federated database systems, web-based information systems, or simply from erroneous data entry [1].

The process of data cleansing, as described in [1], involves several steps:

- Data analysis
- Definition of transformation workflow and mapping rules
- Verification of the transformation workflow and the transformation definitions
- Transformation
- Replacement of the dirty data with the cleaned data in the original sources

Others describe the different steps as data auditing, workflow specification, workflow execution, and post-processing/control [3]. In any case, it is mandatory to analyze the given data before any actual cleansing can be performed. To this end, a classification of dirty data is of great value, serving as a reference to identify the errors and inconsistencies at hand. There are several different general approaches to create a taxonomy of dirty data, such as [1–5].

Other interesting studies on data quality include Sadiq et al. [6] who present a list of themes and keywords derived from papers of the last 20 years of data quality research, Madnick and Wang [7] who give an overview of different topics and methods of quality research projects, and Neely and Cook [8] who combine principles of product and service quality with key elements (i.e., management responsibilities, operation and assurance cost, research and development, production, distribution, personnel management, and legal function) of data quality across the life cycle of the data. However, none of these approaches systematically builds a taxonomy of data quality problems.

When dealing with the detection of errors in time-oriented data there are special aspects to be considered. Time and time-oriented data have distinct characteristics that make it worthwhile to treat it as a separate data type [9–11]. Examples for such characteristics are: Time-oriented data can be given either for a time point or a time interval. While intervals can easily be modeled by two time points, they add complexity if the relationship of such intervals are considered. For example, Allen [12] describes 13 different qualitative time-oriented relationships of intervals. Also, intervals of validity may be relevant for domain experts but might not be explicitly specified in the data. When dealing with time, we commonly interpret it with a calendar and its time units are essential for reasoning about time. However, these calendars have complex structures. For instance, in the Gregorian calendar the duration of a month varies between 28 and 31 days and weeks do not align with months and years. Furthermore, available data may be measured at different levels of temporal precision. Given this complex structure of time, additional errors are possible and correspondingly a specific taxonomy is helpful in addressing these issues.

To start with, we give an outline and summarization of taxonomies of general data quality problems in Section 2. In Section 3 we take a closer look at the different types time-oriented data that demand special consideration. We introduce

some terms on different types of time-oriented data in Section 3.1 before we continue with a detailed description of our main contribution–the categorization of dirty time-oriented data in Section 3.2. In Section 4 we provide a short outlook on further work we have planned to carry out in this area, and we sum up the main results of our work in Section 5.

2 Related Work

In preparing our taxonomy of dirty time-oriented data and data quality problems we start with a review of some general taxonomies. More specifically, we look at the general partitions used in this research (e.g., single-source problems versus multi-source problems), but are especially interested in the 'leafs of the taxonomies', i.e. those types of possible errors that are specific enough to be covered by a specific test (e.g., duplicates, missing values, contradictory values). The leafs mentioned in those general taxonomies are summarized in an overview table (see Tab. 1).

Rahm and Do [1] provide a classification of the problems to be addressed by data cleansing. They distinguish between single-source and multi-source problems as well as between schema- and instance-related problems (see Fig. 1). Multi-source problems occur when there are multiple data sources that have to be integrated such as different data representations, overlapping or contradicting data. Schema-related data problems are quality problems that can be prevented by appropriate integrity constraints or an improved schema design, while instance-related problems cannot be prevented at the schema level (e.g., misspellings).

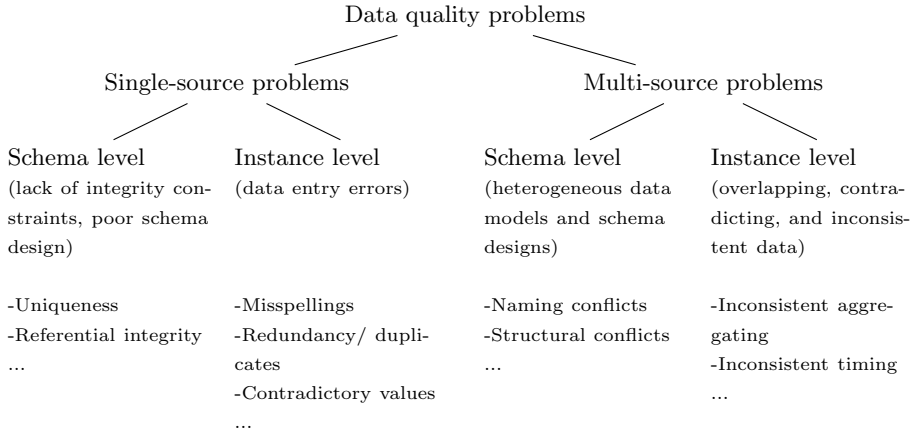

Fig. 1. Classification of data quality problems by Rahm and Do [1].

Later, Kim et al. [2] published a comprehensive classification of dirty data. They aimed at providing a framework for understanding how dirty data arise and which aspects have to be considered when cleansing the data to be able to provide reliable input data for further processing steps. To this end, they present a taxonomy consisting of 33 primitive dirty data types. However, in practice dirty data may be a combination of more than one type of dirty data. Kim et al. start with a root node with only two child nodes – missing data and not-missing data – and continue to further refine these categories adopting the standard 'successive hierarchical refinement' approach (see Fig. 2). Thus, they keep the fan-out factor at each non-leaf node small in order to make intuitively obvious that all meaningful child-nodes are listed. Furthermore, they distinguish wrong data in terms of whether they could have been prevented by techniques supported in today's relational database systems (i.e., automatic enforcement of integrity constraints). When Kim et al. talk about their category of 'outdated temporal data' they refer to the time instant or time interval during which a data is valid (e.g., an employee's occupation may no longer be valid when the employee gets promoted).

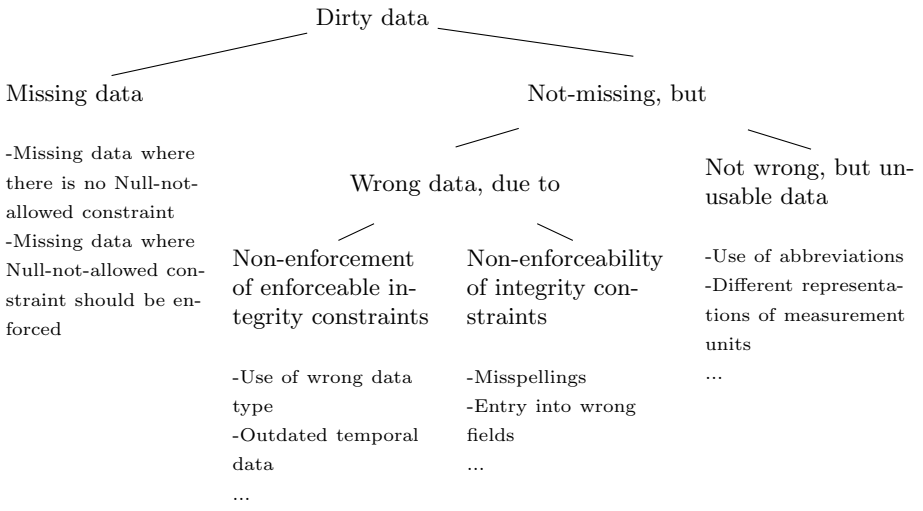

Fig. 2. Classification of dirty data by Kim et al. [2].

Müller and Freytag describe a rougher classification of data anomalies [3]. They start with the differentiation of syntactical anomalies, semantical anomalies, and coverage anomalies (missing values). Syntactical anomalies include lexical errors, domain format errors, and irregularities concerning the non-uniform use of values (e.g., the use of different currencies). Semantic anomalies include integrity constraint violations, contradictions (e.g., a discrepancy between age and date of birth), duplicated entries, and invalid tuples. In this context, invalid

tuples do not represent valid entities from the mini-world but still do not violate any integrity constraints. Coverage anomalies can be divided into missing values and missing tuples (see Fig. 3).

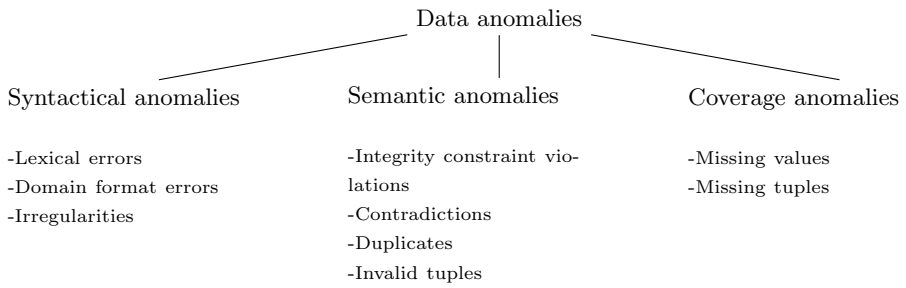

Fig. 3. Classification of data anomalies by Müller and Freytag [3].

Oliveira et al. organize their taxonomy of dirty data by the granularity levels of occurrences [4]. They act on the assumption that data is stored in multiple data sources each of which is composed of several relations with relationships among them. Moreover, a relation contains several tuples and each tuple is composed of a number of attributes. Consequently, they distinguish problems at the level of attributes/tuples (e.g., missing values, misspellings, existence of synonyms in multiple tuples), problems at the level of a single relation (e.g., duplicate tuples, violation of business domain constraints), problems at the level of multiple relations (e.g., referential integrity violations, heterogeneity of syntaxes, incorrect references), and problems at the level of multiple data sources (e.g., heterogeneity of syntaxes, existence of synonyms/homonyms, duplicate tuples) (see Fig. 4).

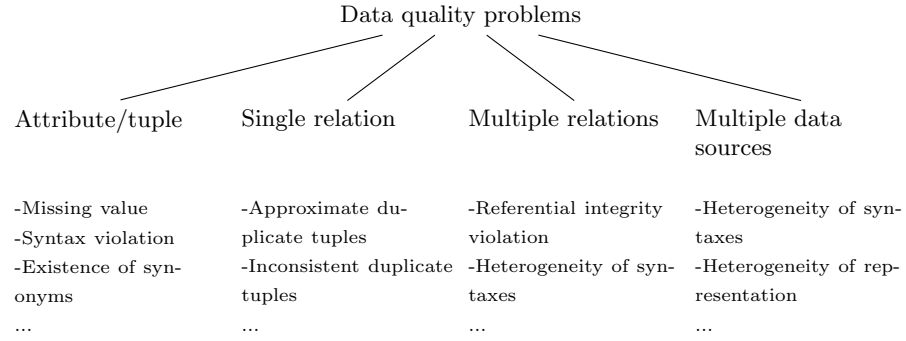

Fig. 4. Classification of data quality problems by Oliveira et al. [4].

Barateiro and Galhardas published a paper [5] about data quality tools including a classification of dirty data which contains problems that are very similar to those of Kim et al. [2]. The clustering of these problems, however, differs from the clustering in [2]. Instead it shows some similarities to the clustering of Rahm and Do [1]: They divide data quality problems into schema level problems (i.e., problems that can be avoided by existing relational database management systems (RDBMS) or an improved schema design) and instance level problems (i.e., problems that cannot be avoided by a better schema definition because they are concerned with the data content). Moreover, schema level data problems are divided into problems that can be avoided by RDBMS and those that cannot. Instance level data problems are further grouped into problems concerning single data records and problems concerning multiple data records (see Fig. 5).

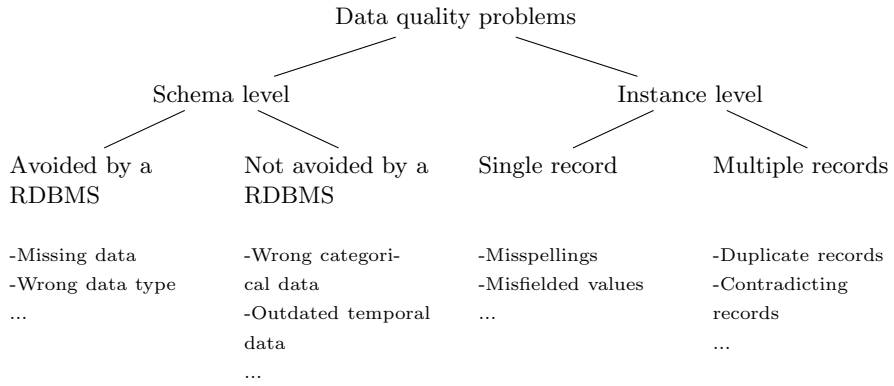

Fig. 5. Classification of data quality problems by Barateiro and Galhardas [5].

These approaches construct and sub-divide their taxonomies of dirty data quite differently. However, when it comes to the actual leaf problems of dirty data, they arrive at very similar findings (see Tab. 1). We omitted the category 'Integrity guaranteed through transaction management' from Kim et al. [2] which contains the problems 'Lost update', 'Dirty read', 'Unrepeatable read', and 'Lost transaction', since we do not consider these kinds of technical problems in the context of this paper. Moreover, we did not include the distinction between schema level problems and instance level problems because we wanted to investigate data quality problems on a more general level and not limit our research to the database-domain. In the following we introduce some definitions and explain our derived taxonomy of dirty time-oriented data using examples to ease understanding.

	Rahm & Do, 2000 [1]	Kim et al., 2003 [2]	Müller & Freytag, 2003 [3]	Oliveira et al., 2005 [4]	Barateiro et al. 2005 [5]
Single source					
Missing data	●	●	●	●	●
Missing value	●	●	●	●	●
Missing tuple	○	○	●	○	○
Semi-empty tuple	○	○	○	●	○
Dummy entry (e.g., -999)					●
Syntax violation / wrong data type	●	●	●	●	●
Duplicates	●	●	●	●	●
Inconsistent duplicates / Contradicting records	●	●	●	●	●
Approximate duplicates	●	○	●	●	●
Unique value violation	●	●		●	●
Incorrect values	●	●	●	●	●
Misspellings	●	●	●	●	●
Domain violation (outside domain range)	●	●	●	●	●
Violation of functional dependency (e.g., age-birth)	●	●	●	●	●
Circularity in a self-relationship	○	○	○	●	○
Incorrect derived values (error in computing data)	○	●	○	○	○
Unexpected low/high values			●		
Misfielded values	●	●	●		●
Invalid substring / Embedded values	●	●		●	●
Ambiguous data; imprecise, cryptic values, abbreviations	●	●		●	●
Outdated temporal data		●			●
Inconsistent spatial data (e.g., incomplete shape)		●			●
Multiple sources					
References	●	●		●	●
Referential integrity violation / dangling data	●	●		●	●
Incorrect references	●			●	
Heterogeneity of representations	●	●	●	●	●
Naming conflicts	●	●	●	●	●
Synonyms	●	●	●	●	●
Homonyms	●	●		●	●
Heterogeneity of syntaxes	●	●		●	●
Different word orderings	●	●		●	●
Uses of special characters	○	●		○	○
Heterogeneity of semantics	●	●	●	●	●
Heterogeneity of measure units (EUR vs. $)	●	●	●	●	●
Heterogeneity of aggregation/abstraction	●	●	●		●
Information refers to different points in time	●	●			●
Heterogeneity of encoding formats (ASCII, EBCDIC, etc.)		●			●

Table 1. Comparison of taxonomies of general data quality problems. (●...included in taxonomy; ○...further refinement, included in parent problem)

3 Dirty Time-Oriented Data

When extending the taxonomies of dirty data to dirty time-oriented data, we focus our research on types of dirty time-oriented data that are distinct to general errors listed in the overview of existing taxonomies above. I.e., we try to add leafs that help to think about possible errors, to build tests to detect these errors, and possibly to correct them.

One of the authors is CEO of a time intelligence solution provider and has extensive business experiences in dealing with real life problems of dirty time-oriented data. In his projects, numerous time-oriented datasets provided by customers are used to support addressing questions of work organization (e.g., working time, staffing levels, service levels) with software solutions specifically developed for these purposes [13, 14]. A typical project may consist of 5 to 20 different types of data files, some of them in more or less structured Excel [15] formats and others exported from databases. Some of these data files may be very small (e.g., a list of active employees), others may be mid-size (e.g., working times of 1000's of employees over many years), and sometimes they are rather large (> 10mio records). Overall more than 50 such projects were pursued in the course of the last years and problems with the quality of data were always a substantial and painful part of the overall projects.

Before we actually present the taxonomy of quality problems, we introduce some terms on different types of time-oriented data. The categorization originates from the observation that checking the data for given problems turns out to be different for these distinct types of time-oriented data.

3.1 Definitions: Types of Time-Oriented Data

An *interval* is a portion of time that can be represented by two points in time that denote the beginning and end of the interval. Alternatively, intervals can be modeled as start time (i.e., a point in time) in combination with its duration (i.e., a given number of seconds, minutes, hours, etc.), or as duration in combination with end time [9]. For instance, 08:00–09:00; 08:17–17:13; 8:17+50'.

A *raster* can be defined as a fragmentation of time without gaps consisting of raster intervals (usually with same lengths). For example, a 30' raster interval that is typically aligned with coarser time units: 00:00–00:30; 00:30–01:00; ...

A *raster interval* is a unit of time that constitutes a raster: 'hour', 'day', 'week', or 30'. In exceptional cases raster intervals may also be of uneven length, such as for the temporal unit 'month'.

Moreover, raster intervals may have *attributes* attached such as 'weekday', 'holiday', 'opening hour', 'working hour', 'school day', or 'Christmas season'. Consequently, there are attributes that can be calculated (e.g., the attribute 'weekday') and attributes that require further information (e.g., the attribute 'holiday').

A given *rastered data set*, however, may contain gaps between the raster intervals, for instance sales data with gaps on weekends and holidays.

Overall we propose to distinguish the following types of time as they may cover errors in different ways:

1. Non-rastered points in time
2. Non-rastered intervals (i.e., start+end, start+duration, or duration+end):
   (a) Start/End of non-rastered intervals (non-rastered points in time)
   (b) Duration of non-rastered intervals
3. Rastered points in time
4. Rastered intervals (i.e., start+end, start+duration, or duration+end):
   (a) Start/End of rastered intervals (rastered points in time)
   (b) Duration of rastered intervals

For instance, rastered time-oriented data may have distinct errors. On the one hand, the raster itself may be violated (e.g., a data set rastered on an hourly basis which contains an interval of minutes). On the other hand, the attributes of rastered intervals may indicate incorrect data values (e.g., sales values outside opening hours), or the values within the rastered intervals may violate some constraint such as 'each rastered interval must contain a value greater than 0 for a given data attribute'. In addition, a further type of data has to be considered when dealing with quality problems of time-oriented data, namely time-dependent values such as 'sales per day'.

3.2 Categorization of Time-Oriented Data Problems

From a methodological perspective, we applied an iterative mixed-initiative approach combining a bottom-up grounded theory procedure [16] with a top-down theory-centric view. On the one hand, our work gathered, modeled, and coded iteratively a number of time-oriented data quality problems that appeared in our real-life data analysis projects. These projects led to a large collection of examples of time-oriented data quality problems in diverse industry sectors and diverse kinds of data. On the other hand, we analyzed, compared, and merged the existing taxonomies discussed above that aim to model dirty data aspects (see Sec. 2 and Tab. 2–4).

In the course of integrating the time-oriented data quality problems with the categorizations of general data quality problems, we re-arranged, refined, extended, and omitted some categories according to our needs and practical experiences. We kept the concept of categorizing data quality problems into problems that occur when the data set stems from a single source and those that occur when two or more data sets need to be merged from multiple sources. Single source problems may of course occur in multiple source data sets too but the provided list of multiple source problems focuses on problems that specifically emerge when dealing with data sets from multiple sources (as mentioned by Rahm and Do [1]). Moreover, we excluded some categories of quality problems which do not relate to any time-oriented aspect such as 'inconsistent spatial data'.

We categorized the considered data types into non-rastered and rastered data. Each category contains the temporal units 'point in time' and 'interval' –

		Description *Example*	non-rastered: Point in time	non-rastered: Start/End of interval	non-rastered: Duration	rastered: Point in time	rastered: Start/End of interval	rastered: Duration	Time-dependent values
Single source									
Missing data	Missing value	Missing time/interval and/or missing value *(Date: NULL, items-sold: 20)*	●	●	●	●	●	●	●
		Dummy entry *(Date: 1970-01-01); (duration: -999)*	●	●	●	●	●	●	
	Missing tuple	Missing time/interval + values *(The whole tuple is missing)*				●	●	●	●
Duplicates	Unique value violation	Same time/interval (exact same time/interval though time/interval is defined as unique value) *(Holidays: 2012-04-09; 2012-04-09)*	●	●	●	●	●	●	
	Exact duplicates	Same time/interval and same values *(Date: 2012-03-29, items-sold: 20 is in table twice*	●	●	●	●	●	●	●
	Inconsistent duplicates	Same real entity with different times/intervals or values *(patient: A, admission: 2012-03-29 8:00) vs. (patient: A, admission: 2012-03-29 8:30)*	●	●	●	●	●	●	●
		Same real entity of time/interval (values) with different granularities (rounding) *(Time: 11:00 vs. 11:03); (Weight: 34,67 vs 35)*	●	●	●				●
Implausible values	Implausible range	Very early date / time in the future *(Date: 1899-03-22); (date: 2099-03-22); (date: 1999-03-22, duration: 100y)*	●	●	●	●	●	●	
	Unexpected low/high values	Deviations from daily/weekly... profile or implausible values *(Average sales on Monday: 50) vs. (this Monday: 500)*							●
		Changes of subsequent values implausible *(Last month: 4000 income) vs. (this month: 80000 income)*							●
		Too long/short intervals between start–start/end–end *Below one second at the cash desk*		●	●				
		Too long/short intervals between start–end/end–start *Off-time between two shifts less than 8h*		●	●				
		Too long/short overall timespan (first to last entry) *Continuous working for more than 12 hours*	●	●	●	●	●	●	
		Same value for too many succeeding records *17 customers in every intervall of the day*							●
Outdated	Outdated temporal data	Only old versions available *Sales values from last year*	●	●	●	●	●	●	
		New version replaced by old version *Project plan tasks overwritten by prior version*	●	●	●	●	●	●	

Table 2. Time-oriented data quality problems within a **single source** (●...has to be checked for this data type).

		Description *Example*	non-rastered: Point in time	non-rastered: Start/End of interval	non-rastered: Duration	rastered: Point in time	rastered: Start/End of interval	rastered: Duration	Time-dependent values
Single source (continued)									
Wrong data	Wrong data type	No time/interval *Date: AAA; duration: **	●	●	●	●	●	●	
	Wrong data format	Wrong date/time/datetime/ duration format *(Date: YYYY-MM-DD) vs. (date: YY-MM-DD); (duration: 7.7h) vs. (duration: 7h42')*	●	●	●	●	●	●	
		Times outside raster (e.g., for denoting end of day) *1-hour-raster but time is 23:59:00 for the end of the last interval*				●	●	●	
	Misfielded values	Time in datefield, date in time field/duration field *(Time in datefield: 14-03, date in timefield: 12:03:08)*	●	●	●	●	●	●	
		Values attached to the wrong/adjacent time/interval *GPS data shows sprints followed by slow runs although the velocity was constant*	●	●	●	●	●	●	●
	Embedded values	Date+time in date field, timezone in time field/duration field *(Time: 22:30) vs. (time: 22:30 CET)*	●	●	●	●	●	●	
	Coded wrongly or not conform to real entity	Wrong time zone *UTC data in stead of local time*	●	●		●	●		
		Valid time/interval but not conform to the real entity *(Admission: 2012-03-04) vs. (real admission: 2012-03-05)*	●	●	●	●	●	●	
	Domain violation (outside domain range)	Outliers in % of concurrent values (attention with small values) for a given point in time/interval *On average (median) 30 customers in a shop in a given hour – in a 10' interval within that hour, a value of 200 is present*							●
		Uneven or overlapping intervals *Turnover data for 8:00–9:00, 9:00–11:00, 11:00–12:00*					●	●	
		Minimum/Maximum violation for given time/interval/type of day *Sales at night even though no employees were present*							●
		Sum of sub-intervals impossible *Seeing the doctor + working hours longer than regular working hours*		●	●		●	●	
		Start, end, or duration do not form a valid interval *(End $\leq$ start); (duration $\leq$ 0)*			●				
		Circularity in a self-relationship *Interval A $\subset$ interval B, interval B $\subset$ interval A, A $\neq$ B*		●	●		●	●	
	Incorrect derived values	Error in computing duration *Error computing sum of employees present within two intervals: (interval: 8:00–8:30, employees: 3), (interval: 8:30–9:00, employees: 3) $\rightarrow$ (interval: 8:00–9:00, employees: 6); no proper dealing with summer time-change; computing the number of work hours per day without deducting the breaks*	●	●	●	●	●	●	●
Ambiguous data	Abbreviations or imprecise/unusual coding	Ambiguous time/interval/duration due to short format *(Date: 06-03-05) vs. (date: 06-05-03); 5' interval encoded as '9:00': (interval: 8:55–9:00) vs. (interval: 9:00–09:05); average handling time per given interval: 3' – not clear: (average of completed interactions) vs. (average of started interactions) within this interval*	●	●	●	●	●	●	
		Extra symbols for time properties *+ or * or 28:00 for next day*	●	●	●	●	●	●	

Table 3. Time-oriented data quality problems within a **single source** (continued) (●...has to be checked for this data type).

		Description *Example*	non-rastered: Point in time	non-rastered: Start/End of interval	non-rastered: Duration	rastered: Point in time	rastered: Start/End of interval	rastered: Duration	Time-dependent values
Multiple Sources									
Heterog. syntaxes	Different data formats/ synonyms	Different date/duration formats *(Date: YYYY-MM-DD) vs. (date: DD-MM-YYYY); (Date: 03-05 (March 5)) vs. (date: 03-05 (May 3))*	●	●	●	●	●	●	
	Different table structure	Time separated from date vs. date+time or start+duration in one column *(Table A: start-date, start-time) vs. (table B: start-timestamp)*	●	●	●	●	●	●	
Heterog. semantics	Heterogeneity of scales (measure units / aggregation)	Different granularities; different interval length *(Table A: whole hours only) vs. (table B: minutes)*	●	●	●	●	●	●	
	Information refer to different times/intervals	Different times/intervals *(Table A: current sales as of yesterday) vs. (table B: sales as of last week)*	●	●	●	●	●	●	
References	Referential integrity violation/ dangling data	No reference to a given time/interval in another source *(Table A: sales per day), (table B: sales assistants per day), problem: table B does not contain a valid reference to a given day from table A or table A does not contain any referencing time*	●	●	●	●	●	●	
	Incorrect reference	Reference exists in other sources but not conform to real entity *Sales of one day (table A) are assigned to certain sales assistants (from table B) because they reference the same day, however, in reality a different crew was working on that day*	●	●	●	●	●	●	

Table 4. Time-oriented data quality problems between **multiple sources** (●...has to be checked for this data type).

the latter being defined by either two points in time (i.e., start and end of the interval), by its start (i.e., one point in time) and its duration, or its end and its duration (as defined in Sec. 3.1). Besides the temporal units, we especially consider time-dependent values (e.g., all events at a given point in time, all events within a given interval). With respect to these categories we outline which data quality problems arise for which data type (indicated by bullets in Tab. 2–4). The first two columns of the tables reflect the general categories derived from existing taxonomies. The third column gives descriptions and examples of specific time-dependent quality problems for each category.

In the course of investigating data quality problems from real-life projects, we realized that the kinds of problems that are subject of this paper (i.e., wrong, duplicated, missing, inconsistent data, etc.) are not the only ones that need to be identified and resolved. Tasks, like checking the credibility of data entries that cannot easily be categorized as 'wrong', or transforming the data table into a specific format that is suitable for further processing steps are strongly linked to the process of data cleansing and need special consideration. Also, a relevant number of problems occur as a consequence of cleansing/transforming the data set, thus such dirtiness might be created by the process itself.

4 Further Work

The generated taxonomy serves as important basis for further planned initiatives to support time-oriented data quality issues. Specifically, we plan to develop a prototype that

1. checks time-oriented data for these kinds of quality problems,
2. generates a report about the problems found,
3. visualizes the 'dirtiness' of the data set and its progress,
4. provides tools for data cleansing:
   - means to specify automatic transformations, and
   - Information Visualization [17] methods for interactive manipulation of the whole dataset as well as of selected entries.
5. supports the management of various versions and corrections/partial updates of the dataset.

The majority of types of dirty data require intervention by a domain expert to be cleansed [2]. Thus, a combination of means for transforming the whole dataset at once with means for interactively investigating the data problems and manipulate single table entries or groups of table entries seems to be a promising solution. Since the sight is the sense with the highest bandwidth we believe that visualization is a good way to communicate a compact overview of the 'dirtiness' of the data as well as to point the user to those cases of data quality problems where manual interaction is needed. Moreover, we plan to realize an interactive Information Visualization [17] prototype that allows for direct manipulation of the data set. This would not only facilitate the task of cleaning the data but it would also provide immediate visual feedback to user actions.

Another important issue of data cleansing is the transformation of the given data table into a table structure that is suited for subsequent processing steps, such as splitting/merging of columns, removing additional rows (e.g., summary rows and comments), or the aggregation of temporal tuples into rastered intervals. A couple of software tools exist to aid this transformation [13, 18–20]. However, further research is needed on which kinds of transformations should be supported and how to support them most efficiently as well as how to organize the management of the various versions and updates.

5 Conclusion

A catalog of general data quality problems which integrates different taxonomies draws a comprehensive picture of problems that have to be considered when dealing with data quality in general. It serves as a reference when formulating integrity constraints or data quality checks.

In this paper we have investigated different approaches of categorizing data quality problems. We have examined a number of relevant taxonomies of dirty data and carved out their similarities and differences. Furthermore, we have focused on the data quality problems that occur when dealing with time-oriented data, in particular. We have derived a number of time-oriented data quality problems from our experience in numerous projects in different industry sectors and we merged the results of the literature review of existing taxonomies with our practical knowledge in dealing with time-oriented data.

Specifically, we presented an integrated and consistent view of general data quality problems and taxonomies. Thus, we provided a useful catalog of data quality problems that need to be considered in general data cleansing tasks. In particular, we provide categorized information about quality problems of time-oriented data. Thus, we established an information basis necessary for further research on the field of dirty time-oriented data, and for the formulation of essential quality checks when preprocessing time-oriented data.

The dimension of time implicates special characteristics which cause specific data quality problems. Thus, a catalog of data quality problems focusing specifically on time-induced problems yields benefits. In spite of its length, we do not claim our categorization of time-oriented data problems to be complete. However, we provide a collection of numerous problems from real life projects which constitutes an important basis for further research. Moreover, we integrated this collection with existing taxonomies of general data quality problems to provide a categorized and unified work of reference. This reference comprises several important aspects that need to be considered when dealing with the quality of time-oriented data.

Acknowledgments. 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 Economy, Family and Youth in the exceptional Laura Bassi Centres of Excellence initiative).

References

1. Rahm, E., Do, H.H.: Data Cleaning: Problems and Current Approaches. IEEE Techn. Bulletin on Data Engineering 31 (2000)
2. Kim, W., Choi, B.-J., Hong, E.-K., Kim, S.-K., Lee, D.: A Taxonomy of Dirty Data. Data Mining and Knowledge Discovery 7, 81–99 (2003)
3. Müller, H., Freytag, J.-C.: Problems, Methods, and Challenges in Comprehensive Data Cleansing. Technical report HUB-IB-164, Humboldt University Berlin (2003)
4. Oliveira, P., Rodrigues, F., Henriques, P.: A Formal Definition of Data Quality Problems. In: 2005 International Conference on Information Quality (MIT IQ Conference) (2005)
5. Barateiro, J., Galhardas, H.: A Survey of Data Quality Tools. Datenbankspektrum 14, 15–21 (2005)
6. Sadiq, S., Yeganeh, N., Indulska, M.: 20 years of data quality research: Themes, trends and synergies. In: 22nd Australasian Database Conference (ADC 2011), pp. 1–10. Australian Computer Society, Sydney, NSW, Australia (2011)
7. Madnick, S., Wang, R., Lee, Y., Zhu, H.: Overview and Framework for Data and Information Quality Research. Journal of Data and Information Quality (JDIQ), 1(1), 1–22 (2009)
8. Neely, M., Cook, J.: A Framework for Classification of the Data and Information Quality Literature and Preliminart Results (1996-2007). In: 14th Americas Conference on Information Systems 2008 (AMICS 2008), pp. 1–14 (2008)
9. Aigner, W., Miksch, S., Schumann, H., Tominski, C.: Visualization of Time-Oriented Data. Springer, London (2011)
10. Andrienko, N., Andrienko, G.: Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach. Springer, Berlin, Germany (2006)
11. Shneiderman, B.: The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In: IEEE Symposium on Visual Languages, pp. 336–343. IEEE Computer Society Press (1996)
12. Allen, J.: Towards a general model of action and time. Artificial Intelligence 23(2), 123–154 (1984)
13. XIMES GmbH: Time Intelligence Solutions [TIS], `http://www.ximes.com/en/software/products/tis` (accessed: 2012-03-30)
14. XIMES GmbH: Qmetrix, `http://www.ximes.com/en/ximes/qmetrix/background.php` (accessed: 2012-03-30)
15. Microsoft: Excel, `http://office.microsoft.com/en-us/excel/` (accessed: 2012-03-30)
16. Corbin, J., Strauss, A.: Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory, 3rd edn. Sage Publications, Los Angeles (2008)
17. Card, S., Mackinlay, J., Shneiderman, B.: Readings in Information Visualization: Using Vision to Think. Morgan Kaufmann, San Francisco (1999)
18. Raman, V., Hellerstein, J.: Potter's Wheel: An Interactive Data Cleaning System. In: 27th International Conference on Very Large Data Bases (VLDB 2001), pp. 381–390. Morgan Kaufmann Publishers (2001)
19. Kandel, S., Paepcke, A., Hellerstein, J., Heer, J.: Wrangler: Interactive Visual Specification of Data Transformation Scripts. In: ACM Human Factors in Computing Systems (CHI 2011), pp. 3363–3372. ACM, New York (2011)
20. Huynh, D., Mazzocchi, S.: Google Refine, `http://code.google.com/p/google-refine` (accessed: 2012-03-30)