Visualization of Multi-Level Data Quality Dimensions with QuaIIe

Sheny Illescas Martinez
Johannes Kepler University Linz
Linz, Austria
k1257276@students.jku.at

Lisa Ehrlinger
Johannes Kepler University Linz
Linz, Austria, and
Software Competence Center Hagenberg GmbH
Hagenberg, Austria
lisa.ehrlinger@jku.at

Wolfram Wöß
Johannes Kepler University Linz
Linz, Austria
wolfram.woess@jku.at

***Abstract*—Data quality assessment is a challenging but necessary task to ensure that business decisions that are derived from data can be trusted. A number of data quality metrics have been developed to measure dimensions like accuracy, completeness, and timeliness. The tool QuaIIe (developed as part of our previous research) facilitates the calculation of different data quality metrics on both, schema- and data-level, and for heterogeneous information systems. However, to gain meaningful results from the automatically calculated metrics, it is key that humans *understand* the results of such metrics. This understanding is specifically important when contextual information needs to be considered, which is not encoded in the data. In this paper, we present a visualization approach to enable human-centered data quality assessment across multiple dimensions and arbitrary complex data sources. The approach has been implemented as graphical user interface in QuaIIe.**

Keywords*—*Data visualization; Data quality, Data quality dimensions; Graphical user interface.

I. Introduction

In a world where data is consumed and created on a daily basis, it has become a challenging task to determine whether the consulted data is of acceptable quality, especially for enterprises. The absence of Data Quality (DQ) assessment can have a severe financial impact on organizations and enterprises. A study made by the Data Warehousing Institute has shown that DQ problems cost American enterprises over 600 billion dollars on a yearly basis [1]. Determining the quality of data has therefore become a pivotal activity in the business field.

Data quality can be calculated with the help of DQ metrics, which are functions to assess different DQ dimensions in a quantitative manner [2]. Every DQ dimension evaluates a specific feature on the data or schema [3]. Assessing the DQ of an entire information system is not trivial, since tasks like specifying the dimensions of interest, determining the methods for assessing the quality of the selected dimensions, analyzing the results, and cleaning problematic data need to be carried out. Yet, the last two activities require one key aspect, which is to *understand* the DQ measurement results.

To effectively communicate and interpret DQ measurement results, visual representations are crucial. DQ visualization systems support the nonlinear analytical process of the comprehension of the qualitative state of the underlying information [4]. It is essential to understand and to know the quality of the data, in order to define goals and to determine the required data cleansing activities [5].

Enterprises and organizations often store their data in Integrated Information Systems (IISs). IISs typically manage and process data from different and heterogeneous information sources. In prior research, we developed the DQ tool QuaIIe (Quality Assessment for Integrated Information Environments), which allows to calculate 15 different DQ metrics for automated data- and schema-quality assessment of IISs [5]. With *multi-level*, we refer to the different aggregation-levels of an IIS, where DQ measurement can be carried out: on attribute-, concept-, data-source-, and IIS-level. Real-world settings in enterprises require automated assessment of numerous and complex data sources, where the navigation and identification of quality issues through humans experts is not trivial. Thus, we contribute in this paper with a human-centered approach to facilitate DQ assessment across multiple data sources and DQ dimensions.

This paper is organized as follows: in Section II, we discuss existing DQ visualization tools and distinguish them from our work. Section III introduces the visualization approach, which was developed in this research and Section IV covers its implementation in QuaIIe. In Section V, we demonstrate the applicability of the approach and conclude with an outlook on future work in Section VI.

II. Related Work

In recent years, DQ has been gaining importance both in research and in industry. However, the exploration of visualization techniques in the context of DQ is sparse. In this section, we discuss ongoing research and DQ tools, which support the visualization of DQ-related aspects.

Kandel et al. [6] implemented Profiler, a tool that uses data mining methods and type inference to target DQ issues in relational Databases (DBs). Profiler provides visual assistance and automatically suggests visualizations for identifying problematic data. In contrast to QuaIIe, the default implementation of Profiler supports only relational tables and other types of data need to be integrated manually. Moreover, Profiler requires the user to choose between multiple suggested visualizations. Such a decision requires domain knowledge of the analyzed

ISBN: 978-1-61208-857-0

data, since the choice for one type of visualization affects the conclusions drawn about the data quality [6].

Bors et al. [7] present a visual approach for analyzing DQ aspects and integrate functions for customizing and creating DQ metrics. Automatically computed DQ metrics are used for determining the DQ of a given data set. Their tool MetricDoc allows to assess DQ on tabular data sets through interaction techniques and distinct views, which provide quality-related information on different aspects. Similar to Profiler, MetricDoc considers only tabular data by default.

Abedjan et al. [8] made a tutorial on metadata discovery, i.e., data profiling. The authors present a series of techniques for profiling activities and stress the importance of visualization when transmitting and interpreting data profiling results. They present a visual representation of functional dependencies with a sunburst diagram, but do not provide detailed explanation of this chart. The hierarchical structure of the diagram is used to express dependencies between sets of attributes and the color is used to represent these sets. In our work, we present a different use of the sunburst diagram, as we make explicit use of the hierarchical order to represent the aggregation levels of an information system. Additionally, we assign semantic meaning to the colors in the chart, to inform the user about the qualitative state of an entire IIS with a simple glance on the diagram.

Xie et al. [9] present two different techniques to visualize and transmit DQ information for multivariate data. In one approach, a given data set is extended with quality measures as new data dimension. In the second approach, DQ information is integrated into visual attributes of existing multivariate visualizations. Both approaches are evaluated with different visualization techniques, such as, parallel coordinates, scatterplots, matrices, and star glyphs [9]. Based on the experiments in [9], the authors conclude that the selection of the visual attributes is the main success factor for the visualization. They noted that hue has a stronger capacity of transmitting quality information in parallel coordinates when compared to other visual attributes (e.g., the size). A possible explanation for this effect is that hue has a high degree of preattentive processing and does not need extra space [9]. This argument supports our design decision of using color, i.e., hue for transmitting quality information.

Gratzl et al. [10] present an interactive visualization technique for rankings. Rankings are typically influenced by one or more attributes, which can have multiple dependencies to other attributes. The visual tool LineUp introduced in [10] aims to assist the user in analyzing and comparing multiple rankings, as well as, on how changes in the attribute combination can affect the end-ranking. The visualization uses mainly bar charts for representing different ranking-related information, such as the individual attribute categories. Attributes are mapped to a normalized value and the sum of these values reflects the obtained ranking. The user is able assign a weight to the different attributes and gets visual feedback when the weights are changed.

Furmanova et al. [11] introduce a scalable visualization technique for tabular data called Taggle. The tool is open-source and can be accessed via a web application [11]. It consists of two main components: the tabular panel and the data selection panel. Columns from the tabular panel can be altered by means of aggregation, filtering, sorting, etc. They can be further inspected in the detailed data selection panel, which provides a visual summary of the data in form of histograms for suitable data sets [11]. The authors provide different mechanisms for aggregating and grouping data.

Blumenschein et al. [12] present a visual approach for analyzing high-dimensional data called SMARTexplore. Their tool uses a table-based technique that supports the identification and examination of clusters, patterns, and correlation in high-dimensional data. Rows represent single records or a collection of records (i.e., record groups) and columns represent different dimensions [12]. Record groups are aggregated and visually encoded by color, making it possible to compare them across dimensions. The identification of patterns is facilitated by automatically sorting groups and dimensions.

III. Visualization Approach

Our work represents a visual extension of QuaIIe, a DQ tool, which was originally introduced in [5]. QuaIIe performs automated DQ assessment of multiple data sources within an IIS [5]. During the quality assessment process, various DQ measurements are computed on different IIS aggregation levels and are stored in a *DQ report*.

This DQ report contains quality information for several DQ dimensions at the attribute-, concept-, data-source-, and IIS-level. In short, a data source represents a single schema in an IIS and it can contain an arbitrary number of concepts or associations (i.e., schema elements). A concept is a representation of a real-world object, e.g., in case of a relational DB, it can be a table. An association is a relationship between two or more concepts. An attribute is a property of a concept or an association. The internal arrangement of the DQ report resembles the hierarchical structure of the components of the IISs under inspection. More information on the schema representation in QuaIIe is provided in [5] and [13].

The quality computations on the four levels (attribute-, concept-, data-source-, and IIS-level) can contain a wide range of information. Displaying all this information at once could overload the user and thus, hamper the extraction of important information. Therefore, we split the information into two views: an overview (see Section III-A), in which we decided to use the sunburst diagram, and a detailed view (see Section III-B), which allows to dig deeper on demand by clicking on particular IIS elements in the overview.

A. Overview: Sunburst Diagram and Scoring Function

The data investigated with QuaIIe has a hierarchical structure and the relationships between the data items build a tree network [5]. A common visual representation of trees is a graph (consisting of nodes and edges), since it displays the structure of the data. However, it misuses space, e.g., by displaying edges that provide little information [14].

After analyzing the state-of-art for hierarchical data visualization and taking into account their advantages and disadvantages, we decided to explore and evaluate the potential of the sunburst diagram for the overview. An essential aspect for this decision was that the sunburst diagram is a compact visualization (in contrast to graphs) and overcomes a key drawback of other representations such as the treemaps, since it preserves the structure of the tree hierarchy. A sunburst diagram indicates hierarchy through a series of rings, which are typically sliced based on the number of nodes within the hierarchy level [15]. The slices of the inner circles have a hierarchical relationship to the segments of the outer circles [16]. These relationships are formally translated as the parent-child relationship.

Figure 1 shows the overview on the Graphical User Interface (GUI), which was developed for QuaIIe. The sunburst diagram in sector (B) can be interpreted as follows: the white circle in the center represents an entire IIS, the slices of the first ring represent the different information sources of the IIS, e.g., DBs, ontologies, Comma-Separated Values (CSV) files, and the slices of the second ring represent the concepts (e.g., tables or classes) of the data sources. The sunburst diagram is ideally suited to represent large trees while preserving the hierarchical tree-structure [14]. This is an advantage over the traditional tree map, where the hierarchical structure of the tree cannot be easily detected due to its representation [14].

A common technique when employing the sunburst diagram is to use color (hue) for hierarchical grouping or for assigning categories to the schema elements. In the context of DQ, color can be used for transmitting information related to the qualitative state of the elements. To ease the learning effort for the user, we limit the number of colors and provide a color palette that indicates their meaning. To determine the color for each element in the IIS, a mechanism for assessing their quality rating is required, i.e., a *categorization function*. Due to the categorization function, a user can easily identify elements that require attention (i.e., have low quality) by inspecting the diagram.

With QuaIIe, multiple DQ dimensions can be analyzed simultaneously, which, in turn, can be assessed by one or more DQ metrics. We refer to the value computed by a metric as its *rating*. To visually transmit the quality information from the DQ report efficiently and effectively, a mechanism to summarize the data from the report is required. In our approach, we summarize DQ ratings by DQ dimension. Having a single (numerical) value per dimension allows to compute a quality rating for each element that contains DQ measurements. The quality rating is computed according to

$$rating_s = \frac{\sum_{i=1}^{n} dim_{si}.w_i}{n}, \qquad (1)$$

$$dim_{si} = \frac{\sum_{j=1}^{m} r_j}{m}, \qquad (2)$$

where $rating_s$ is the quality rating of an element s, w_i the weight of the dimension i, dim_{si} the dimension average of element s and dimension i, and r_j a rating computed with metric j. We contemplate the possibility to specify weights for the different dimensions and thus, to define the degree to which each DQ dimension contributes to the overall quality of the different elements. By default this value is set to one, assuming that all DQ dimensions are of equal importance.

After computing the quality rating, we can determine a category for it, that is, poor, fair, good, or excellent. The computed quality ratings are normalized values between zero and one, and thus, adhere to the DQ metric requirement (1) by Heinrich et al. [2]. This range is divided into four intervals, where each interval is assigned to a specific category. In combination with the quality rating, we determine the quality category of an element s as follows:

$$category_s = \begin{cases} poor, & \text{if } rating_s < 0.25 \\ fair, & \text{if } 0.25 \geq rating_s < 0.5 \\ good, & \text{if } 0.5 \geq rating_s < 0.75 \\ excellent, & \text{if } rating_s \geq 0.75 \end{cases}$$

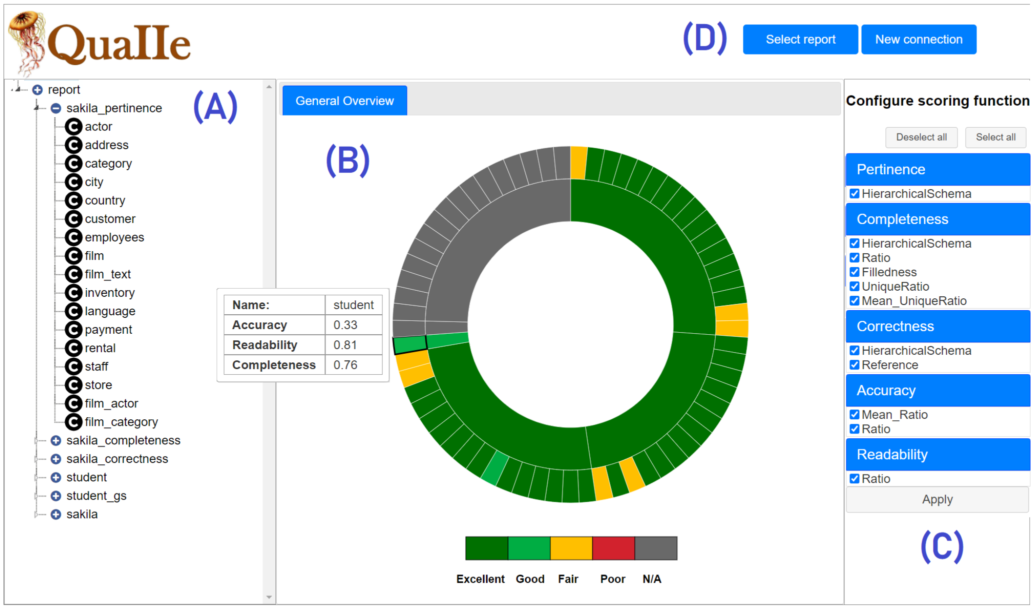
This function allows to determine a quality state for each element, which in turn allows to determine the overall quality state of IISs.

In addition, the sunburst diagram is enriched with tooltips that display extra information for each source, e.g., its name and computed quality category. Further, a filter (cf. sector (C) in Figure 1) allows to select DQ metrics and dimensions of interest. This feature allows to specifically analyze the impact of selected metrics and dimensions with respect to the calculated DQ category.

B. Detailed View

The purpose of the detailed view is to present the entire quality information from the DQ report in an organized and understandable way, including the information that was hidden in the overview. We assume that the data analyst is not interested in reviewing all DQ information at once, but rather has a goal in mind, e.g., an element with low quality detected through the sunburst diagram. This view can be activated by double-clicking on an element of the diagram or from the tree view (sector (A) in Figure 1). Figure 2 shows an excerpt of the detailed view of the concept `student`. It can be seen that the detailed view is organized in three sections.

The first section provides a summary of the quality dimensions of the selected element. Additional (textual) quality information provided by `QuaIIe` is displayed in the annotations table. The second section contains all DQ ratings measured by the different metrics for each dimension. Every metric is represented by a bar chart and the height of the bar represents the measured value. The third section displays all DQ information of the attributes, if available. Hence, the third section is only visible if the selected element contains attribute information. In Figure 2, we can notice that the attribute `id` has further quality information. The dimensions key attribute and completeness were measured by different metrics namely, "Pseudo_Boolean", "UniqueRatio", and "Filledness" (cf. [17]



Name:	student
Accuracy	0.33
Readability	0.81
Completeness	0.76

Pertinence
☑ HierarchicalSchema

Completeness
☑ HierarchicalSchema
☑ Ratio
☑ Filledness
☑ UniqueRatio
☑ Mean_UniqueRatio

Correctness
☑ HierarchicalSchema
☑ Reference

Accuracy
☑ Mean_Ratio
☑ Ratio

Readability
☑ Ratio

Fig. 1. Data quality visualization of an IIS

for details on the calculation). As can be seen in the chart, the dimension “Key_Attribute” was only evaluated by the metric “Pseudo_Boolean”, whereas the dimension “Completeness” was measured by the two remaining metrics. All three metrics computed a value of 1 for their respective dimension.

IV. User Interface Implementation for QuaIIe

The GUI presented in this paper was developed as a Java Server Pages (JSP) web application. Figure 1 displays the surface, which consists of four sections: (A) tree view of the IIS components, (B) sunburst diagram, (C) filter panel, and (D) resource loading and configuration.

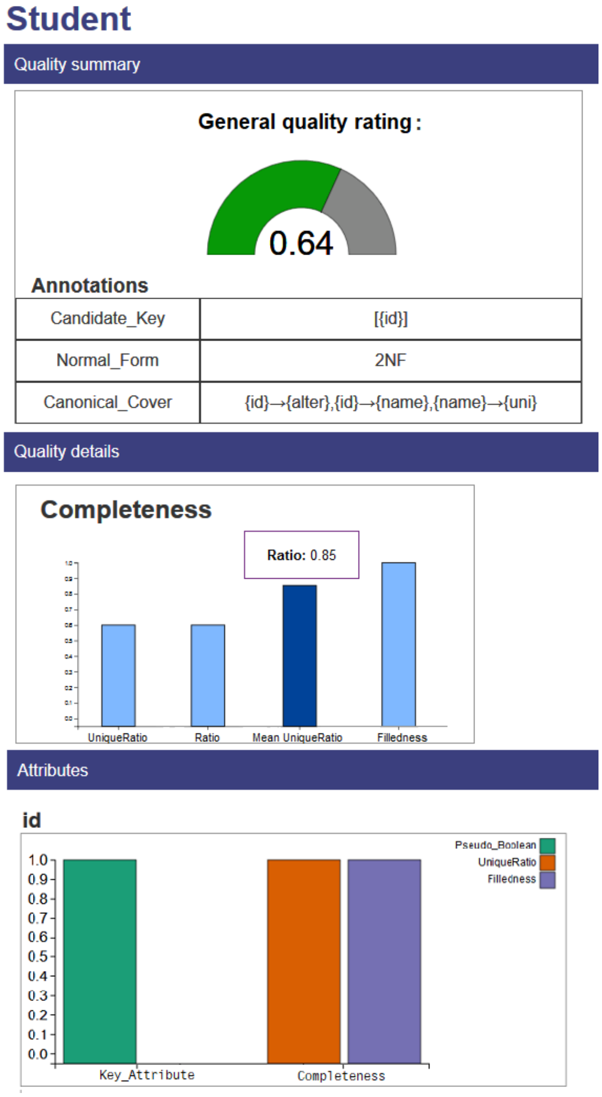
The sunburst diagram in the center provides on the one hand an overview on the entire IIS under inspection and allows on the other hand an easy identification of focal points. In Figure 1, we can quickly identify that there are seven elements with lower quality within our IIS, namely the elements with the yellow color. We are aware that using colors to distinguish graphical elements is not barrier-free, especially for users with red-green color blindness or complete color blindness. Although accessibility was no core requirement for this application, we plan to increase the customization by letting the user choose the respective colors and optionally use patterns to distinguish between the categories. These and other possibilities how graphical charts can be made accessible are discussed by Altmanninger and Wöß in [18].

The sunburst diagram was also extended with tooltips, which are displayed when the user hovers over the element of interest. In Figure 1, the tooltip for the data source `student` is highlighted, which contains a summary of its assigned quality dimensions. The tooltip indicates that the DQ dimensions accuracy, readability and completeness have a rating of 0.33, 0.81, and 0.76 respectively. Since the accuracy is notably lower than the readability and completeness, it subsequently reduces the total rating. This explains why the quality measurement for student falls within the quality range [0.5, 0.75[and is therefore rated as `good`.

The filter panel allows to configure the scoring function used to calculate the quality rating, and where the results are represented by the colors in the main chart. The user can select the DQ metrics and dimensions that should be considered when computing the quality rating of the elements. In this way, the user is able to analyze the impact that specific dimensions and metrics have on the quality state of the IIS. In addition, the filter panel also allows inspecting single DQ dimensions individually.

Resources are loaded/configured in the upper right section of the GUI (D). Existing DQ reports can be loaded and immediately visualized. Section (D) further allows to configure new data sources and provides a process for triggering new quality calculations. During the quantification process, special constraints are taken into account. For example, certain metrics require a *gold standard*, that is, a point of reference against which the available data source can be compared. If no such gold standard is provided, metrics that require one are hidden in the application.

To communicate with the back-end application (QuaIIe), several Java Servlets were created, where each serves a specific purpose, e.g., the configuration of a new MySQL data soure connection. The current version of the QuaIIe GUI supports the configuration of data sources and gold standards of the following types: ontologies, MySQL DBs, and CSV files. Configuration for all other data sources supported by QuaIIe (e.g., Cassandra wide-column stores) are planned for future work. Since QuaIIe creates an abstraction layer on all data sources, different types can be used in combination (cf. [5]). For example, a MySQL data source can have an ontology



Candidate_Key	[{id}]
Normal_Form	2NF
Canonical_Cover	{id}→{alter},{id}→{name},{name}→{uni}

Fig. 2. Detailed view of the QuaIIe GUI

as gold standard. After the configuration of the data sources, the DQ of all their elements can be measured. In this step, the seven quality dimensions supported by QuaIIe (accuracy, correctness, completeness, pertinence, minimality, readability, and normalization) can be analyzed. Each of these dimensions can be quantified using one or several metrics, which can be calculated on the schema- or data-level. When the user submits his/her selection of DQ dimensions and metrics, the DQ for the current data source is assessed. After the process is finished, the user can configure a new data source or can trigger the generation of a DQ report. When the DQ report is created, it is parsed into a JSON format suitable for different visualizations used in the application (sunburst diagram, bar charts, donut chart, etc.) and sent to the front-end application for its processing. All the visualizations of this application are created with the D3.js library [19].

V. Demonstration: DQ Assessment of Sakila DB

To demonstrate the capabilities of the visualization approach in the QuaIIe GUI, we present a case study, where we evaluated the DQ of the Sakila DB [20], which models a DVD rental store. For demonstration purposes, we also included a simple data set representing students information. Prior to DQ assessment, the connections to the data sources need to be established. For the demo, we created the following:

- `students_gs`: a CSV connection representing a file with data about students (name, address, academic title, matriculation number, etc.),
- `students`: a CSV connections where arbitrary attributes were altered/deleted from the original students data,
- `sakila_gs`: a MySQL connection, which represents the original Sakila DB and that is used as gold standard,
- `sakila_pertinence`, `sakila_complet- eness`, `sakila_correctness`: three ontology connections in Data Source Description (DSD) notation, which are representations of the original Sakila DB, but were modified to artificially downgrade the respective quality dimension.

The DSD notation is introduced in [13] and modification details of the three files are provided in [5]. For repeatability, the DSD files for all four data sources are published on the QuaIIe project website [21].

During the configuration process, a user can select an option to create a data source connection as the gold standard. When this option is selected, the connection is internally stored as a gold standard and can be reused for any other open connection.

To create an ontology or a CSV connection, the respective DSD file is uploaded and a name is assigned to the new connection. In the DQ form, the user selects the dimensions of interest and which metrics should be used for the quality assessment process. Each metric shows a list of all concepts. By selecting a concept, that metric is calculated for that concept at the concept level. In addition, the user can specify whether the calculation should be carried out at the schema level. In this use case for the Sakila ontologies, only one DQ dimension per connection is evaluated, e.g., relevance for `sakila_pertinence`. The process is repeated for all three files. In the case of the CSV connection, we analyze the dimensions accuracy, completeness, normality and readability in the schema and (whenever possible) also in the concept level. Details of the performed evaluation and DQ calculations are provided in [17].

Finally, the DQ report is created and its visualization is displayed immediately. Figure 1 shows the resulting sunburst diagram. When viewing the diagram, the gray color for `sakila_gs` and `students_gs` indicates that no DQ measurements have been performed for the gold standards. In addition, seven elements with lower quality (in yellow) stand out. By investigating these elements through the detailed view, the annotations indicate that all seven elements have been marked as *extra elements*. Extra elements are elements that exist in the investigated data source, but have no correspondence in the gold standard [5]. Thus, they impact the DQ dimensions pertinence (describing the prevalence of unnecessary elements) and correctness.

Next, we decided to specifically analyze the dimension completeness by specifying this in the filter. As expected, `sakila_completeness` has the lowest rating (compared to the other data sources) of 0.75. Nevertheless, the data source reaches the minimum for the rating `excellent` ($\geq$ 0.75). After further analysis and with the help of the chart, it is found that all concepts have a high rating for the dimensions completeness. Thus, another reason for the lower rating could be that complete concepts are missing from the data source. This assumption can be confirmed by comparing the number of concepts of `sakila_completeness` to its gold standard `sakila_gs` (e.g., with the help of the tree view). The comparison shows that the original database has 16 concepts, but `sakila_completeness` only has 14.

VI. Conclusion

In this paper, we presented an approach to visualize arbitrary complex IISs for exploring their DQ measurements across different DQ dimensions and metrics. The approach was implemented as a GUI for the DQ tool QuaIIe, originally introduced in [5]. The GUI supports a data analyst in selecting DQ dimensions and metrics of interest and in inspecting the qualitative state of several data sources. For ongoing and future work we plan the following:

- Averages can blur potential outliers, which might be important for users. Thus, the calculation of the quality rating will be extended with different aggregation functions, where the user can select the most appropriate one for a given use case.
- The user experience of the presented visualization will be evaluated to determine how easy and efficient the execution of DQ measurement tasks are perceived.
- The performance of the GUI should be evaluated with more real-world data.
- During the implementation of the GUI, the DQ tool QuaIIe has been extended with DQ monitoring capabilities. Thus, we additionally plan to extend the GUI to support the visualization of continuous DQ measurements over time.

Acknowledgment

The research reported in this paper has been funded by BMK, BMDW, and the Province of Upper Austria in the frame of the COMET Programme managed by FFG.

References

[1] W. W. Eckerson, “Data Quality and The Bottom Line Achieving Business Success through a Commitment to High Quality Data,” 2002. [Online]. Available: https://tdwi.org [Retrieved: 04, 2021]

[2] B. Heinrich, D. Hristova, M. Klier, A. Schiller, and M. Szubartowicz, “Requirements for Data Quality Metrics,” *Journal of Data and Information Quality*, vol. 9, no. 2, pp. 12:1–12:32, January 2018.

[3] R. Y. Wang and D. M. Strong, “Beyond Accuracy: What Data Quality Means to Data Consumers,” *Journal of Management Information Systems*, vol. 12, no. 4, pp. 5–33, March 1996.

[4] J. M. B. Josko and J. E. Ferreira, “Visualization Properties for Data Quality Visual Assessment: An Exploratory Case Study,” *Information Visualization*, vol. 16, no. 2, pp. 93–112, 2017. [Online]. Available: https://doi.org/10.1177/1473871616629516

[5] L. Ehrlinger, B. Werth, and W. Wöß, “QuaIIe: A Data Quality Assessment Tool for Integrated Information Systems,” in *Proceedings of the Tenth International Conference on Advances in Databases, Knowledge, and Data Applications (DBKDA 2018)*. Nice, France: International Academy, Research and Industry Association, 2018, pp. 21–31.

[6] S. Kandel, R. Parikh, A. Paepcke, J. Hellerstein, and J. Heer, “Profiler: Integrated Statistical Analysis and Visualization for Data Quality Assessment,” in *Advanced Visual Interfaces*, 2012. [Online]. Available: http://vis.stanford.edu/papers/profiler

[7] C. Bors, T. Gschwandtner, S. Kriglstein, S. Miksch, and M. Pohl, “Visual Interactive Creation, Customization, and Analysis of Data Quality Metrics,” *Journal of Data and Information Quality*, vol. 10, no. 1, pp. 3:1–3:26, May 2018. [Online]. Available: http://doi.acm.org/10.1145/3190578

[8] Z. Abedjan, L. Golab, and F. Naumann, “Data Profiling: A Tutorial,” May 2017, pp. 1747–1751.

[9] Z. Xie, S. Huang, M. O. Ward, and E. A. Rundensteiner, “Exploratory Visualization of Multivariate Data with Variable Quality,” in *2006 IEEE Symposium On Visual Analytics Science And Technology*, October 2006, pp. 183–190.

[10] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit, “LineUp: Visual Analysis of Multi-Attribute Rankings,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2277–2286, 2013.

[11] K. Furmanova, S. Gratzl, H. Stitz, T. Zichner, M. Jaresova, A. Lex, and M. Streit, “Taggle: Scalable Visualization of Tabular Data Through Aggregation,” *Information Visualization*, vol. 19, no. 2, pp. 114–136, 2019. [Online]. Available: https://taggle.caleydoapp.org [Retrieved: 04, 2021]

[12] M. Blumenschein, M. Behrisch, S. Schmid, S. Butscher, D. R. Wahl, K. Villinger, B. Renner, H. Reiterer, and D. A. Keim, “Smartexplore : Simplifying high-dimensional data analysis through a table-based visual analytics approach,” in *IEEE Conference on Visual Analytics Science and Technology (VAST) 2018*, 2018.

[13] L. Ehrlinger and W. Wöß, “Semi-Automatically Generated Hybrid Ontologies for Information Integration,” in *SEMANTiCS (Posters & Demos)*, ser. CEUR Workshop Proceedings, vol. 1481. Aachen: RWTH, November 2015, pp. 100–104.

[14] G. Wills, *Visualizing Hierarchical Data*. Boston, MA: Springer US, 2009, pp. 3425–3432. [Online]. Available: https://doi.org/10.1007/978-0-387-39940-9_1380

[15] M. Schermann, “A Reader on Data Visualization,” 2019. [Online]. Available: https://mschermann.github.io/data_viz_reader [Retrieved: 04, 2021]

[16] F. Aps, “Sunburst Diagram,” 2015. [Online]. Available: https://datavizproject.com/data-type/sunburst-diagram [Retrieved: 04, 2021]

[17] L. Ehrlinger, B. Werth, and W. Wöß, “Automated Continuous Data Quality Measurement with QuaIIe,” *International Journal on Advances in Software*, vol. 11, no. 3 & 4, pp. 400–417, December 2018.

[18] K. Altmanninger and W. Wöß, “Accessible Graphics in Web Applications: Dynamic Generation, Analysis and Verification,” in *International Conference ICCHP (Computers Helping People with Special Needs)*, ser. Lecture Notes in Computer Science, vol. 5105. Berlin Heidelberg: Springer-Verlag, 2008, pp. 378–385.

[19] M. Bostock, “D3 – Data-Driven Documents,” 2020. [Online]. Available: https://d3js.org [Retrieved: 04, 2021]

[20] Oracle, “Sakila Sample Database,” 2006. [Online]. Available: https://dev.mysql.com/doc/sakila/en [Retrieved: 04, 2021]

[21] L. Ehrlinger, “Automated and Continuous Data Quality Measurement,” 2016. [Online]. Available: http://dqm.faw.jku.at/ [Retrieved: 04, 2021]