PART V - Conclusion

11 Synopsis and Future Work

The research goal of this thesis was the investigation of the usefulness of ontologies for data quality management. In this thesis project, we created an ontology, called the Data Quality Management vocabulary (DQM vocabulary), to collect and store data requirements in a structured and linkable format. Moreover, we configured a wiki, called data requirements wiki, which contains standard forms to capture data requirements and to store them based on the elements of our ontology, the DQM vocabulary. Because of the storage of data requirements in the DQM vocabulary schema, we were able to create a reporting tool, called the Semantic Data Quality manager, that automatically processes the captured data requirements and creates data quality monitoring and assessment reports without any additional manual intervention. In the following, we review our initial research questions, provide answers, and highlight the findings and results of this thesis. Moreover, we draw a final conclusion on the usefulness of ontologies and provide starting points for future work.

11.1 Research Summary

In section 2.1, we have subdivided the initial research goal into five research questions, which served as the roadmap for this thesis. In the following, we provide a short summary of the answers to the research questions:

RQ1: What kind of data quality problems exist?

We have argued that, in order to develop solutions to improve data quality, the nature of data quality problems has to be understood. Therefore, we have developed a typology of data quality problems for relational systems (see section 3.3) and for the Semantic Web (see section 5.2). The derived typologies are based on an analysis of literature related to data quality problems in relational databases and the Semantic Web.

C. Fürber, *Data Quality Management with Semantic Technologies,*
DOI 10.1007/978-3-658-12225-6_11, © Springer Fachmedien Wiesbaden 2016

RQ2: Which activities have to be performed during data quality management?

Since we have aimed to develop an artifact that facilitates data quality management, we had to identify typical activities that are performed during data quality management. Consequently, we analyzed the two most popular data quality management methodologies, namely Total Data Quality Management (TDQM, Wang, 1998) and Total Information Quality Management (TIQM, English, 1999), for commonalities as part of section 3.5. Based on the commonalities, we defined a new data quality management process in section 8.2 that is fitted to SDQM, the major artifact of this thesis.

RQ3: Which knowledge has to be represented to support data quality management?

In section 3.6, we argued that data requirements represent knowledge about the characteristics of high-quality data. Assuming that data quality problems are the result of requirement violations, we derived ten generic data requirement types from the typology of data quality problems. We thereby focused on quality problems of relational data. The generic data requirement types represent the core knowledge concepts that have to be represented to support data quality management.

RQ4: How can we represent knowledge relevant for data quality management to reduce manual work?

Based on the generic requirement types, we developed an ontology, called the DQM vocabulary, that supports the representation of knowledge for data quality management activities, such as data requirements definition, data quality monitoring, and data quality assessment (see section 0 and (Fürber & Hepp, 2011b)). The development procedure followed the ontology development methodology as provided in (Uschold & Gruninger, 1996). The DQM vocabulary consists of classes and properties that can be used to represent data requirements in a machine-readable format. Due to this design, we reduced manual input by automating the generation of data quality monitoring and assessment reports based on the representation of data requirements knowledge via the DQM vocabulary.

RQ5: How can we utilize knowledge for data quality management represented within ontological structures?

In chapter 7, we have developed the SDQM framework, a data quality management framework that is based on other programming frameworks and artifacts primarily from

the Semantic Web community. SDQM processes quality-relevant knowledge represented in the DQM vocabulary to derive data quality monitoring and assessment reports. Knowledge processing within the SDQM framework is based on generic SPARQL queries which provide the basis for the derived reports. Since the SPARQL queries only use elements from the DQM vocabulary, they are of generic use for any domain, as long as the data requirements are formulated based on the DQM vocabulary. SDQM's data requirements wiki can be used to capture data requirements from business experts via standardized forms. Thus, users of SDQM do not need to possess programming skills to evaluate the quality of data. Furthermore, we have shown in section 9.4 that the represented knowledge can also be used to automatically identify inconsistent or duplicate data requirements. Finally, we provided an installation and application procedure for SDQM in chapter 8 of this thesis so that our research project is reproducible.

11.2 Contributions

The contributions of this thesis can be separated into (1) practical and (2) theoretical contributions. On the practical side, we developed a new artifact, called SDQM, which solves real-world problems in the area of data quality management and integrates state of the art technology of the Web.

SDQM consists of three major artifacts that have been developed in the course of this thesis, namely (1) an ontology for representing knowledge that is relevant for data quality management, (2) a wiki for capturing and maintaining data requirements, and (3) a reporting frontend to create data quality monitoring and assessment reports. *SDQM's data requirements wiki* can be used to capture quality-relevant knowledge from business experts via standardized forms. Thus, users of SDQM do not need to possess programming skills to evaluate the quality of data. The captured data requirements are automatically represented in RDF based on the DQM vocabulary. Therefore, SDQM's reporting frontend, called *the Semantic Data Quality Manager (SDQMgr),* can automatically process the captured knowledge to derive data quality monitoring and assessment reports without any additional programming. As evaluated in section 0, this is a major distinction from conventional data quality tools such as Talend OS for Data Quality, since they usually represent data requirements as part of

programming code. Due to its integration with standard wiki software, SDQM is especially suited for large organizations with distributed knowledge. The reduced complexity of maintaining data requirements logic may mitigate the effort for data quality management. To the best of our knowledge, SDQM is the first data quality management framework that uses standard wiki software to capture, manage, and utilize data requirements for automated data quality monitoring and assessment. Moreover, SDQM facilitates the automated identification of inconsistent and duplicate requirements with standard SPARQL queries, since the captured data requirements are represented in RDF format. At present, we do not know of any data quality management software that has a similar feature.

Moreover, this thesis provided several theoretical contributions for data quality research as listed below:

(1) A typology of data quality problems in relational systems and the Semantic Web (sections 3.3 and 5.2).
(2) A requirement-centric methodology for data quality management (section 8.2).
(3) Ten generic data requirement types (section 3.6.1).
(4) A survey of related work (chapter 10).

These theoretical contributions of this thesis may be useful for future research and applications in the area of data quality management.

11.3 Conclusion and Future Work

In this thesis, we have shown a way how ontologies can be employed for data requirements management, data quality monitoring, and data quality assessment for information systems and Semantic Web data. The evaluation results documented in chapter 9 indicate that the developed approach is also usable in real-world settings. Furthermore, we have collected first evidence that Web and Semantic Web technologies can facilitate the management of data quality in several ways, namely

- Semantic wikis facilitate the generation of data requirements by non-programmers, since they offer standardized forms for knowledge capturing.
- Representation of data requirements within ontological structures facilitates the automated derivation of requirement violations and data quality scores.

- Representation of data requirements within ontological structures facilitates the automated identification of duplicate and inconsistent data requirements.

However, we also discovered some limitations. Compared to conventional data quality architectures, such as Talend OS for Data Quality with a MySQL database, SDQM still has a significant performance gap. Moreover, SDQM does not yet provide features for data profiling and may not be able to represent complex functional dependencies in RDF. Additionally, we discovered that the use of SDQM for open environments, such as the Semantic Web, has some limitations. For example, Semantic Web scenarios contain a large diversity of world views which may sometimes collide. Therefore, it may not be possible or even suitable to solely seek for consistent data requirements (cf. Madnick & Zhu, 2006, p. 460f.). In consequence, the perceived characteristics of high quality data may be diverse and contradictory. Thus, data quality improvement directed to a single, harmonized quality perception is most likely not applicable for the Semantic Web. However, the results of this thesis provide multiple possibilities for future work in several areas which are explained in the following.

Semantic Web settings: Currently, SDQM is focused on closed environments based on relational information systems. Future work could address the extension of SDQM to cover specific data quality problems of the Semantic Web as specified in section 5.2. Moreover, SDQM could be deployed to the World Wide Web to collect data requirements from the Web community about public Semantic Web data sources, such as DBpedia or Geonames. Based on the captured knowledge, agreement and disagreement about data requirements could be identified and further investigated.

Technological optimization: Currently, SDQM was mainly used in single source scenarios. Future work could address the investigation of SDQM's ability to cover multi-source scenarios, e.g. in which properties with identical intensions are stored in disparate data sources. Moreover, SDQM's duplicate checking algorithms require further performance optimizations as explained in section 9.2. Additionally, SDQMgr's reports could be extended by charts to visualize data quality scores. Finally, SDQM could be extended by data profiling features to identify data requirements via data analysis.

Economic impact: SDQM may save manual effort due to the provision of standardized forms for capturing data requirements and standardized data quality reports. However, solid evidence is still missing that really proves a higher efficiency

and lower costs compared to conventional data quality management tools. Future studies could also address the potential of SDQM to reduce costs of information exchange among different parties within a supply chain. For example, SDQM could be used to express and publish data requirements of customers within supply chains in an audit-proof way. Then the delivered data of the supplier could be verified according to these explicitly specified data requirements with SDQM. As a potential outcome, ambiguity and misunderstandings during information exchange may be reduced and the result of the verification against the customer's data requirements could be part of contracts and, therefore, used as an incentive to improve the quality of the information exchange within the supply chain. SDQM could be applied in a study related to such a scenario to investigate its potential to reduce costs for information exchange within the supply chain.