On semi-automated matching and integration of database schemas

Ünal Karakaş, Ö.

Publication date
2010

Chapter 7

Thesis conclusions and future work

7.1 Summary of General Approach

The importance of developing a supporting infrastructure for data sharing has been understood clearly during the last years, with the increasing need for collaboration among organizations in a wide variety of domains, from manufacturing and service industry to scientific virtual laboratory and disaster management. In order to facilitate and enable collaboration among distributed, heterogeneous, and autonomous organizations, one of the first requirements that needs to be met is enabling access to certain data that is to be shared among the stakeholder organizations. However, before any sharing of data could possibly occur, many existing syntactic, semantic, and structural heterogeneities among the stakeholder database schemas need to be resolved. Manual resolution of schema heterogeneities is very time consuming, cumbersome, and error prone. This becomes more challenging when scaling up is required in large networks. Namely, without automated ways of removing such heterogeneities between separate database schemas of participants, data interoperability and therefore effective collaboration goals cannot be met.

Consequently, provision of automated schema matching and integration tools is an active area of research with numerous technical challenges. One of the biggest challenges in this area is the automatic **resolution of database schema heterogeneity**, without which provision of integrated data access and sharing among autonomous, heterogeneous, and distributed databases will remain difficult to achieve.

In this thesis, we propose a supervised automated approach to solve both the problem of schema matching and the schema integration assisting the users with removal of heterogeneities among source database schemas and to integrate them effectively. We also provide an implementation of this approach in the form of a software system, which we call the Semi-Automatic Schema Matching and INTegration (SASMINT).

As the **first** step towards the provision of a supporting infrastructure for data sharing in collaborative networks of organizations, we have performed an analysis of different types of information sharing heterogeneities. Furthermore, we have identified the varieties of heterogeneities that represent the most important obstacles to Schema Matching and Schema Integration tasks.

As the **second** step, we have analyzed related research on Schema Matching and Schema Integration approaches. Based on this survey and the identified open issues, we have devised

our approach to deal with many challenges, and have proposed a new approach for schema matching and schema integration in relational databases. In order to not reinvent the wheel, we have combined a number of well-known algorithms suggested in related research tackling some of the challenges of matching terms from different domains. We have generated an innovative mechanism and format to represent the results generated by the schema matching and schema integration processes. Furthermore, we have proposed an approach for automatically generating an integrated schema using the results of schema matching through the design and development of a number of heuristic rules. Finally, we have defined a derivation language to formally specify and store how the integrated schema is derived from its input donor and recipient schemas.

As the **third** step, we have implemented our approach to semi-automatic schema matching and schema integration both as a proof of concept and in order to verify and validate it. The main components of the SASMINT system architecture, which are implemented in this thesis comprise:

a) *Sampler Component*, which helps users with automatic identification of appropriate weight for each algorithm used for linguistic matching.
b) *Graph Representation Component*, which is responsible for representing schemas in the DAG format.
c) *GUI Component*, which enables users to interact with the system to configure needed parameters and to modify and accept the results generated by schema matching and schema integration processes.
d) *Schema Matching Component*, which matches the recipient and the donor schemas using a combination of linguistic and structure matching techniques.
e) *Schema Integration Component*, which both integrates the donor and recipient schemas using the set of pre-defined rules and generates the formal specification of integrated schema results using a derivation language.

As the **fourth** step, we have finally evaluated our approach for Schema Matching in comparison with the most closely related approach and system, the COMA++'s approach, through experimenting with six pairs of schemas consisting of a variety of heterogeneities. Furthermore, we have also evaluated our schema integration approach. We could not compare this part against other schema integration approaches, since there was no other system similar to SASMINT that uses its results from schema matching for the purpose of semi-automatic schema integration. Our experiment results for schema matching and schema integration have shown that SASMINT provides good quality results and higher than its closely related competitor.

7.2 Reflections on the Research Questions

> **RQ1.** Which effective approaches and architectures can enable data sharing through interlinking and/or integrating heterogeneous databases of distributed nodes?

Before establishing a solution for a problem, it is important to understand all concepts and terminology related to it. In order to meet this requirement, in Chapter 2, we provided definitions of a variety of terms in the database management research domain concerning approaches, architectures, and systems for interlinking and/or integrating heterogeneous data provided by distributed nodes, in order to enable data sharing among them. Since in the research literature quite often the same term was used to mean different things and different terms were used to refer to the same concept, we thought that it was crucial to differentiate

among these definitions and clarify the terminology used in the research work explained in this thesis. For this purpose, in Chapter 2, we described a number of concepts related to distributed information management and we provided our classification for multidatabases, based on schema coupling. We then defined schema matching and schema integration and specified how they relate to distributed information management.

> **RQ2.** What is a representative taxonomy for addressing database schema heterogeneities, and in turn applicable to formalization of schema matching and schema integration challenges?

Heterogeneity is the biggest obstacle to schema matching and schema integration. In Chapter 3, we presented different types of heterogeneities that exist among information systems. Information systems heterogeneity ranges from the heterogeneity of information and its definition and classification, to the systems heterogeneity. Considering the aim of schema matching and schema integration processes, schema conflicts are the ones that need to be tackled. We categorize schema conflicts as structural and linguistic and the linguistic conflicts further as syntactic and semantic. Especially semantic and structural conflicts are difficult to automatically resolve. The more a schema matching and integration approach can automatically resolve such conflicts, the higher the value of the approach, as less user input is required.

> **RQ3.** What are effective mechanisms for semi-automatic schema matching and schema integration, and how should the user be involved in the process?

In Chapter 4, we addressed a number of efforts and systems related to providing access to heterogeneous databases. We examined them in four main groups: 1) database integration and interoperability approaches, 2) schema matching approaches, 3) schema integration approaches, and 4) ontology matching and merging approaches. Database integration and interoperability approaches typically do not consider any automation in schema matching. Schema matching approaches, on the other hand, are either limited in the solutions that they provide or utilize a few match algorithms resolving only specific heterogeneities. They still require a lot of manual input. Furthermore, most of these schema matching approaches do not provide any GUI for helping users to modify match results and do not address using their results for schema integration. While very much related, the schema matching is seen as a separate problem than schema integration, and using the match results for semi-automatic schema integration is not taken into account. As for the schema integration approaches, their provided solutions are not generic enough. They generally assume that correspondences among schemas are already a given input. We proposed the SASMINT approach to overcome the limitations of previous approaches. It supports both semi-automatic schema matching and schema integration. SASMINT addresses and handles all types of conflicts addressed in Section 3.3. However, a fully automatic resolution is not possible for some types of semantic and structural conflicts, as described earlier and thus user input might be required in some cases. SASMINT uses a combination of linguistic and structure matching metrics and algorithms in order to resolve different types of conflicts addressed in Section 3.3. A novel way of identifying appropriate weights for each metric and algorithm is also proposed. It also represent that once formally specified, the results of schema matching can be exploited for semi-automatic schema integration. By means of a GUI, users can easily modify and store the match and integration results. SASMINT defines an XML-based derivation language as the storage format for the results of matching and integration. In order to verify and validate the SASMINT approach, we have implemented it. In Chapter 5, we provided details of the development architecture of SASMINT together with a number of screenshots of this system.

> **RQ4.** How can we assess and validate the effectiveness of the proposed semi-automatic approaches for schema matching and schema integration?

In order to validate the proposed approach, its evaluation needs to be done against the leading competitors and/or the well defined generic measures. In Chapter 6, we explained how we evaluated the approach of SASMINT. In order to measure the quality of schema matching, a number of well-known measures are addressed, namely, the precision, recall, f-measure, and overall. Then, SASMINT is compared applying these measures, against a leading competitor. For the schema integration, completeness and minimality measures are introduced and applied to identify how well the input schemas are combined by the integrated schema approach and whether the integrated schema is optimal or if it contains redundant elements or keys. After identifying the set of test schemas, quality of schema matching approach of SASMINT was compared to that of COMA++. It was shown that SASMINT was at least as good as if not better than one of the leading state of the art schema matching systems. Evaluation of schema integration experiment of SASMINT also generated very promising results. Furthermore, we also performed some tests to evaluate the Sampler component. In these tests, we identified that Sampler helped to improve the quality of the match results.

7.3 Future Work

There are several areas of research that can continue and further extend certain aspects and features of the work presented in this thesis:

- *Support for XML Schema*

Considering the current extensive use of relational databases, the implementation of the proposed SASMINT system that is provided in this thesis supports matching and integration of relational schemas only. However, both the system and the data architecture of SASMINT have been designed to also be able to support matching and integration of other frequently used data model representations, such as the XML Schema. For supporting matching of XML Schemas, the adapter framework that is now in place needs to be extended with XML Schema import features for incorporating XML Schema support features. In order to integrate XML Schemas, new integration rules need to be defined into SASMINT.

- *Support for Ontology*

Similar to what is stated above for XML Schema support, SASMINT could be extended with the support for Ontologies. This extension would require more work compared to XML Schema support, since the semantics of Ontologies could entail incorporation of technologies/tools like inference/reasoning engines.

- *Using Machine Learning Techniques for the Sampler Component*

At present Sampler applies an approach based on f-measure to identify the best applicable weights for each linguistic matching algorithm in relation to the considered specific schemas. However, it could be further extended to also utilize machine learning techniques for this purpose. Machine learning algorithms can examine large amounts of data and make intelligent decisions based on these data. Therefore, by learning from the true and false positive matches, machine learning algorithms might identify more appropriate weights.

- *Creating a benchmark for schema matching and integration*

In our evaluation studies, one challenge was to find/design objective (relational) schemas that we could use to measure the functional performance of our schema matching and integration system. The test schemas used by other schema matching and evaluation research were designed for a specific need in mind and consisted of only certain types of schema heterogeneities. As a future work, the creation of a benchmark would be valuable, in order to generate more generic schemas to serve as the base for comparable evaluation between systems.

- *Fragmented Matching and Integration*

Current focus of SASMINT is to address and resolve different types of heterogeneities, when two schemas are compared, and not addressing very large schemas. However, future work could consider matching and integrating very large schemas. This would require enabling the fragmented matching and integration in order to make it easier to compare and integrate big schemas, and in turn require the identification of most appropriate fragments for this purpose.