SAMOA - A Semi-automated Ontology Alignment Method for Systems Integration in Safety-critical Environments

Thomas Moser, Kathrin Schimper, Richard Mordinyi and Amin Anjomshoaa
Institute of Software Technology and Interactive Systems, Vienna University of Technology
{thomas.moser, kathrin.schimper, richard.mordinyi, amin.anjomshoaa}@tuwien.ac.at

Abstract

The integration of heterogeneous data sources with even heterogeneous semantic meanings poses a challenge for data and system integrators. Ontology Alignment (OA) tries to identify similarities between heterogeneous ontologies and to automatically create suitable mappings for transformation. However, the usage of standard OA approach for safety-critical domains needs further investigation.

In this paper, we describe a semi-automated ontology alignment approach (SAMOA) well-suitable for integration scenarios of safety-critical applications. The major contribution of our approach is the modeling differentiation between individual system knowledge and generic domain-specific knowledge.

We evaluate our approach by providing a typical use case example from the Air Traffic Management (ATM) domain. In addition we analyze to what extent the SAMOA approach can be supported by state-of-the-art OA approaches.

1. Introduction

Complex information systems (CIS), like systems for the air traffic management (ATM) or production automation domain, usually consist of a high number of heterogeneous subsystems. Each of these subsystems usually has its own data types or data structures. Many of today's information systems were developed independently for targeted business needs, but when the business needs change, these systems need to be integrated into other parts of the organization or entirely into other organizations [8]. The integration of such systems poses a number of challenges for data and system integrators. The high number of systems and data structures results in the need for time consuming and often error-prone human contributions.

A solution approach is to query multiple data sources at once. While the users of these systems still see a single schema (whether relational or XML), queries are translated on the fly to appropriate queries over the individual heterogeneous data sources, and results are combined appropriately from partial results obtained from the sources [8]. In any data sharing architecture, reconciling semantic heterogeneity is the key. No matter whether the query is issued on the fly or data is loaded into a warehouse, the semantic differences between data sources need to be reconciled. Typically, these differences are reconciled by semantic mappings. These are expressions that specify how to translate data from one data source into another in a way that preserves the semantics of the data, or alternatively, reformulate a query posed on one source into a query on another source [16].

Ontology Alignment (OA) is an automated process, which tries to identify similarities between two or more heterogeneous ontologies based on a set of metrics, like string similarity or structural similarity measurements. For the identified similarities, mappings are created in order to overcome the semantic gaps between the heterogeneous ontologies. This works well for big taxonomies where the failure rate regarding wrong mappings is not the crucial factor, however the usage of these OA approaches for safety-critical domains needs further investigations. [9]

In this paper we describe a semi-automated ontology alignment approach (*SAMOA*) well-suitable for integration scenarios of safety-critical applications. This ontology alignment approach is part of an iterative system engineering process, called *ISN* – Information Sharing Network, described in Biffl et al. [1, 2]. Main features of *ISN* are the differentiation between individual system knowledge and generic domain-specific knowledge, which allows concurrent modeling, and the modeling support for integration restrictions. The major benefits of the *ISN* approach are the externalization of implicitly known knowledge in a machine-readable and machine-understandable way,

978-0-7695-3575-3/09 $25.00 © 2009 IEEE
DOI 10.1109/CISIS.2009.138

allowing the automation of time-consuming and error-prone tasks today primarily conducted manually by humans.

In a use case example from the ATM domain we show a typical application of our approach by providing the automated derivation of solution candidates for system integration and automatically generated transformation instructions for message exchange between the integrated systems.

The remainder of this paper is structured as follows: section 2 summarizes related work on ontology alignment, section 3 defines the research issues, section 4 pictures the use case, and section 5 describes the *SAMOA* approach, while section 6 discusses the results of the *SAMOA* approach. Finally, section 7 concludes the paper and identifies further research.

2. Related Work

This section summarizes related work on ontology alignment, presents a generic ontology alignment approach and concludes by shortly describing typical ontology alignment approaches and tools.

2.1. Ontology Alignment

In general ontology alignment is used to connect partially different ontologies of one certain domain and overcomes therewith the heterogeneity problem. Ontology alignment tries to find a corresponding entity in one ontology for an entity in another ontology, with the same or at least a similar meaning. To align two or more ontologies, connections between the entities (concepts, relations or instances) of the ontologies need to be discovered. These entities can be equal, similar or different. Alignments between ontologies can be detected by using information sources like a common reference ontology (upper ontologies, background knowledge), lexical information, ontology structure, user input, external resources (WordNet, synonym databases, dictionaries) or prior matches [11].

2.2. A Generic Ontology Alignment Approach

Based on the fact that there has been defined a general mapping process [4, 5], which is said to subsume all the other approaches, it will be described first. Initially two ontologies are needed as input. In the first step, called feature engineering, features will be selected, which describe a specific entity (concept, attribute, and relation). The following features of ontologies are used to detect alignments [12]: concept names and descriptions, class hierarchy (relationships), property definitions (domains, ranges, restrictions), instances of classes, and class descriptions.

After that it is possible to restrict the search space by choosing the entities for a comparison. For the next step, similarity computation for strings, objects and sets of objects, as well as analysis of dissimilarity and so on are applied. Then the similarity values for a candidate pair of entities have to be aggregated to get one single value. These values will be used for mapping the entities of the ontologies. There are several possibilities like thresholds, relaxation labeling or combining structural and similarity criteria. After these steps it is possible to iterate over the whole process, for a better using of the structure of ontologies, because similarities of related entities are able to influence similarities of other entities.

2.3. Ontology Alignment Approaches & Tools

Generally, there are two different types of tools for working with ontologies, ontology development tools and ontology alignment, mapping or merging tools. One common development tool is Protégé[1]. Protégé is a java-based free, open source ontology editor and knowledge-base framework, where ontologies can be modeled via the Protégé-Frames or the Protégé-OWL editors. There are many plug-ins available, which range from visualization to mapping tools. Today there are many approaches for ontology mapping or merging. There are console- and web-based tools as well as tools with graphical user interface. They reach from completely manual to fully automatic processes. In the majority of cases ontology mapping is done manually, although this is a very time and effort consuming work. Hence there are more and more semi-automatic ontology mapping approaches, which try to support users by making suggestions or providing visualizations.

In the following three different approaches for ontology mapping are introduced shortly. Each approach applies a different method. FOAM, Framework for Ontology Alignment and Mapping [4, 6], is based on NOM, Naïve Ontology Alignment, and is a fully and semi-automatically framework for aligning two or more ontologies. FOAM is based on the general alignment process and applies heuristic measures, more precisely a wide range of similarity functions, to compute similarities of labels, structure and instances. PROMPT [7, 13, 15] is a semi-automated mapping and merging tool available as a plug-in for Protégé. It works with simple lexical-distance measures, to detect similar labels. It is also designed for other algorithms to be easily plugged in,

[1] http://protege.stanford.edu/

like WordNet, the FOAM algorithm and so on. It has an additional function called Anchor-PROMPT [14], which analyses the structure of the graph to find even more alignments. GLUE [3] is a semi-automated machine learning approach, which originates from the research area of schema mapping. It needs a large number of instances for learning and it is not possible to align relations and instances directly.

3. Research Issues

Recent projects with industry partners from safety-critical domains raised concerns about the challenges of data and systems integration in modern technology-driven environments. From an integration point of view, a major goal was to improve the capability of assuring validity of an integration solution while facilitating team work and tool support.

From this general goal, we focus in this paper on the following research issues regarding the *SAMOA* ontology alignment process:

1. Safety-Critical Ontology Alignment: Investigate to what extent the mainly manual ontology alignment tasks of the *SAMOA* approach could be supported by other more automated ontology alignment approaches without violating the requirements regarding safety-criticalness.

2. Risks of applying state-of-the-art ontology alignment approaches: Investigate the risks of using standard OA approaches within the *SAMOA* approach. Analyze the requirements resulting from the use case which have to be fulfilled by the investigated OA approaches.

4. Use Case Description

Air Traffic Management (ATM) is a business based on providing timely and correct data analyses from a web of heterogeneous legacy applications. With the need to dramatically improve the flexibility to provide new ways of systems integration while keeping the usual high level of safety this domain seems very well suited to prototype the proposed approach.

The prototype case study shown in Figure 1 presents a set of business applications and the ISN network consisting of nodes connected by edges. There are two types of nodes: red nodes handle highly secure connections only, while green nodes do not provide specific security mechanisms. An edge refers to a network connection with specific characteristics, e.g. bandwidth, security level, reliability. Business applications, listed on the left and on the right hand sides, are loosely coupled, i.e., they do not know anything of each other apart from data providing and consuming contracts. Each application is connected to at least one network node.

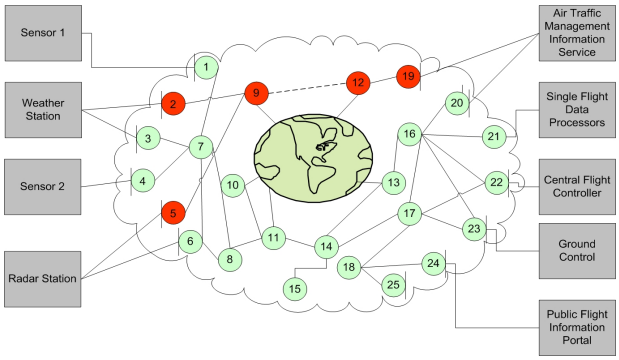

Figure 1: ATM example use case

In our simplified example a business application is a sink service that requires a specific type of data to work properly; or a source service that produces data needed by sink services. The business system Air Traffic Management Information Service (ATMIS) has to provide information services about flights to business partners via a Public Flight Information Portal (PFIP). ATMIS needs to collect and refine information from at least two other systems: the Central Flight Controller (CFC) and the Single Flight Data Processors (SFDPs).

As input to integration process each data provider, in our case CFC and SFDPs, defines the data content and format it can provide and the quality of service, e.g., the frequency of incoming data such as radar signals; each data consumer, in our case ATMIS, similarly defines his needs for data content, format and quality of service, and may additionally require conditions such as data coming from a defined geographical area and within a defined time window.

The described scenario makes one particular demand towards ontology alignment technologies. Since it is a safety critical domain, clear responsibilities and actions are significant. Therefore, the alignment procedure itself is not allowed to make decisions on its own, and each performed action has to be deterministic (e.g. reproducible and valid). It may suggest solutions but at the end the human response needs to be able to supervise the process and make the final decision.

5. SAMOA Ontology Alignment Approach

In *ISN*, a three step ontology architecture is used. Generally, the customer ontology extends the domain ontology and the domain ontology extends the Information Sharing Network (ISN) ontology. Figure 2 shows an overview of the ontology architecture used in *ISN*.

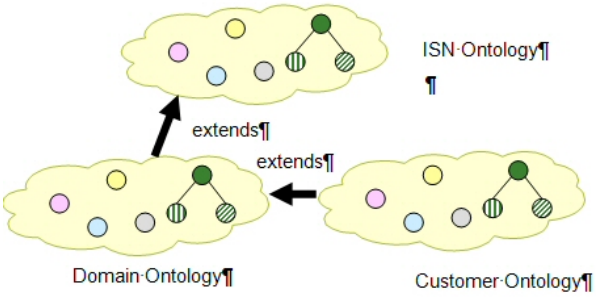

Figure 2: Ontology Architecture

The ISN ontology contains basic concepts like infrastructure concepts, service and message concepts, policy and contract, and transformation and conversion. The domain ontology contains concepts of the domain-specific knowledge and instances for the concepts described in the ISN ontology. The customer ontology defines the legacy applications, services, provided and consumed messages by adding instances to the concepts of the domain or the ISN ontology. In the customer ontology the semantic context of the message segments and the format are described. If a concept exists in more than one customer ontology, it is possible to include it in the domain ontology.

The ISN process for designing the ontologies starts with the description of the ISN Ontology, which is the top level ontology, done by the ISN owner. Then the Domain Ontology has to be defined by the domain expert. After that the network administrator describes the infrastructure. While the domain expert describes the global policies, the customer ontology will be defined by the expert team of each consortium member organization supported by the domain expert. Furthermore the provider and consumer services and the customer policies will be described. The next step is to match each message segment, which has been specified in the customer ontology, to a corresponding domain concept, which has been specified in the domain ontology. This is done by the expert team of each legacy application to be integrated and the domain expert. Since this matching is performed in a completely manual way, it can become quite time-consuming and error-prone, especially in the case of larger domain ontologies. Additionally, manual matching decisions are hard to reproduce and therefore may mean a risk for a safety-critical application. Afterwards the message concepts are defined. Then consistency checks of the ontologies are accomplished and a domain expert review will be performed to check the meaningful modeling of the customer ontologies, the aligning of the segments to the domain concepts and the consistency of the ontologies.

In *ISN*, the alignment process is done fully manual. After the definition of a customer ontology, the message segments have to be aligned to the domain concepts by the customer expert team and the domain expert by hand. Each message segment will be aligned to one domain concept. This alignment provides the identification of semantically equal data and therefore supports the integration of applications in an existing domain. A message can consist of several message segments, which contain one value of an application. In the design time the alignment is needed for the identification of collaboration candidates and for the creation of Transformation Maps (T-Maps). During the run time, the alignment is used for the actual transformation accomplished by T-Maps. A T-Map is needed to transform message segments between two applications, because the provider service may label or format a message segment in another way than the consumer service does. Due to the alignment it is known that both are semantically equal, because they are aligned to the same domain concept. Figure 3 shows a simple example: the message of the provider contains information about an “aeroplane” while the message of the consumer contains information about an “aircraft”, but since both are mapped to the same concept defined in the domain ontology, namely to the domain concept “airplane”, a transformation between provider and consumer is possible.

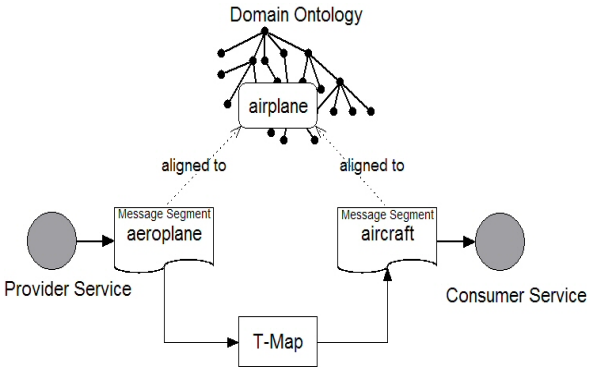

Figure 3: Overview of the alignment process

A T-Map is defined in XML syntax and consists of at least one input and output message segment. The segments contain a unique ID and instructions, how the input segment is transformed to an output segment.

There are a number of possible transformations like changing the name of a segment, converting the format using converters, merging or splitting a set of input segments or querying external services for transformation.

6. Results and Discussion

ISN is a typical example for the need of ontology alignment and mapping. There are systems from different and similar domains, which need to work together, but do have a different language and format for equal or similar things. But to gain a good and efficient collaboration and to guarantee a high level of safety, which is very important in a safety-critical environment like the air traffic management domain, a common understanding is essential. As mentioned before, in the *ISN* project the alignment itself is done manually by the domain expert or an expert team by human intuition. This is a very time and effort consuming task. Following a possibility is presented, which shows how the alignment can be semi-automated. The alignment should not be done fully automatic, because of the safety aspect of the whole system. But it should be possible that suggestions are made, which could be accepted or declined by the user. This can be implemented as an additional feature and should not replace the manual process. Hence next to the suggestion handling the user must be able to do alignments by hand.

The advantages of using such a semi-automated approach are various [10]. First of all it is possible to speed up the discovering of alignments. It takes a lot of time to search an ontology, consisting of over thousand entities, for an appropriate alignment candidate by hand. If suggestions are generated automatically to be accepted or declined by the user, time and effort can be reduced significantly. Furthermore, the generated suggestions are repeatable. It is possible to rerun all executed steps of the alignment process accurately. In addition, consistency failures in the domain ontology can be detected by receiving similar or equal domain concepts as alignment suggestions. On the other hand it has to be considered that an alignment tool will not always suggest the optimal alignment candidates and sometimes also generates false or no suggestions. The quality of the alignment suggestions depends on the implemented algorithm. Therefore it is necessary to use an appropriate algorithm or a combination of algorithms to gain the best possible result.

In *ISN*, the ontologies are developed in Protégé. Hence PROMPT, which is available as plug-in for Protégé, would be appropriate for aligning theses ontologies. As mentioned before, PROMPT only makes suggestions, which can be accepted or declined by the user. The user is also able to do mappings manually, without using a suggestion. This is suitable for *ISN*, because nothing is done without manual affirmation, which is very important regarding the safety-criticalness of the overall application domain. Next to an algorithm which searches for lexical equality and Anchor-PROMPT, which uses the graph structure of ontologies to identify alignments, it is also possible to integrate a set of different algorithms within PROMPT.

These algorithms are implemented as a plug-in for PROMPT, like the FOAM algorithm or an algorithm that uses WordNet. The FOAM algorithm works with a range of similarity functions to find alignments. One function, for example, is based on Levenshtein's edit distance, which computes the distance between two strings. This distance is the number of needed operations to transform one string into the other. WordNet is a large semantic database of nouns, verbs, adjectives and adverbs in English language. They are grouped to so-called synsets, which are defined as sets of synonyms. These synsets are connected by conceptual-semantic and lexical relations. The length of the path between two strings can be interpreted as a measure of similarity. Needless to say, that WordNet is only helpful when it includes vocabulary of the given domain.

7. Conclusion

In this paper we introduced the semi-automated OA approach *SAMOA* which is part of the iterative systems engineering process called *ISN*. We described the ontology architecture of the *ISN* process and gave an overview of the process steps of the *SAMOA* process. By describing an exemplary use case from the Air Traffic Management domain, we derived the requirements for OA in safety-critical environments. Based on a literature survey regarding state-of-the-art OA approaches we examined the questions whether the *SAMOA* approach can benefit from these state-of-the-art OA approaches, by answering the following two research questions.

Safety-Critical Ontology Alignment: Since the alignment should not be performed fully automated because of the safety-criticalness of the application domains, we propose a semi-automated approach that provides suggestions to the user which can be accepted or declined. The advantages of using such a semi-automated approach are a signifant reduction of time and effort needed for the mapping, the reproducibility of the given suggestions (and mappings), and the

detection of consistency failures in the domain ontology.

Risks of applying state-of-the-art ontology alignment approaches: While presenting a set of advantages, the adaptation of the *ISN* OA approach to the *SAMOA* OA approach also bears some risks. The quality of the alignment suggestions heavily and primarily depends on the implemented OA method and may not always suggest the optimal alignment candidate or sometimes – even worse – result in no or false suggestions. Therefore it is necessary to use an appropriate algorithm or a combination of algorithms to gain the best possible result.

Future work will include the practical evaluation of the combination of state-of-the-art OA approaches and the *SAMOA* approach, as well as the implementation of our approach into other safety-critical domains in order to assure cross-domain usage.

Acknowledgments

The authors would like to acknowledge all project members of the SWIS/FISN project performed from 2006-2008 at Vienna University of Technology together with Frequentis AG.

References

[1] S. Biffl, R. Mordinyi, T. Moser, and D. Wahyudin, "Ontology-supported quality assurance for component-based systems configuration," in Proc. 6th International Workshop on Software Quality, 2008, pp. 59-64.

[2] S. Biffl, R. Mordinyi, and A. Schatten, "A Model-Driven Architecture Approach Using Explicit Stakeholder Quality Requirement Models for Building Dependable Information Systems," in Proc. 5th International Workshop on Software Quality, 2007, pp. 1-6.

[3] A. Doan, J. Madhavan, P. Domingos, and A. Halevy, "Ontology matching: A machine learning approach," in Handbook on Ontologies in Information Systems: Springer, 2004, pp. 397-416.

[4] M. Ehrig and SpringerLink, *Ontology Alignment: Bridging the Semantic Gap*: Springer, 2007.

[5] M. Ehrig and S. Staab, "Efficiency of Ontology Mapping Approaches," in Proc. International Workshop on Semantic Intelligent Middleware for the Web and the Grid, 2004, pp. 51-66.

[6] M. Ehrig and Y. Sure, "FOAM—framework for ontology alignment and mapping: results of the ontology alignment initiative," in Proc. Workshop on Integrating Ontologies, 2005, pp. 72-76.

[7] S.M. Falconer, N.F. Noy, and M.A. Storey, "Towards understanding the needs of cognitive support for ontology mapping," in Proc. Ontology Matching Workshop, 2006, pp. 25-36.

[8] A. Halevy, "Why your data won't mix," *Queue*, vol. 3, (no. 8), pp. 50-58, 2005.

[9] K. Kotis and M. Lanzenberger, "Ontology Matching: Current Status, Dilemmas and Future Challenges," in Proc. Complex, Intelligent and Software Intensive Systems, 2008. CISIS 2008. International Conference on, 2008, pp. 924-927.

[10] K. Kotis, G.A. Vouros, and K. Stergiou, "Towards automatic merging of domain ontologies: The HCONE-merge approach," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 4, (no. 1), pp. 60-79, 2006.

[11] N. Noy and H. Stuckenschmidt, "Ontology Alignment: An Annotated Bibliography," in Proc. Semantic Interoperability and Integration, 2005, pp. 48-56.

[12] N.F. Noy, "Semantic integration: a survey of ontology-based approaches," *ACM SIGMOD Record*, vol. 33, (no. 4), pp. 65-70, 2004.

[13] N.F. Noy and M. Musen, "PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment," in Proc. National Conference of Artificial Intelligence, 2000, pp. 450-455.

[14] N.F. Noy and M.A. Musen, "Anchor-PROMPT: Using Non-Local Context for Semantic Matching," in Proc. Workshop on Ontologies and Information Sharing at the International Joint Conference on Artificial Intelligence (IJCAI), 2001, pp. 63-70.

[15] N.F. Noy and M.A. Musen, "The PROMPT suite: interactive tools for ontology merging and mapping," *International Journal of Human-Computer Studies*, vol. 59, (no. 6), pp. 983-1024, 2003.

[16] H. Wache, T. Vögele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann, and S. Hübner, "Ontology-based integration of information-a survey of existing approaches," in Proc. Workshop on Ontologies and Information Sharing at the International Joint Conference on Artificial Intelligence (IJCAI), 2001, pp. 108-117.