Using Ontology-Based Reference Models in Digital Production Engineering Integration

Peter Szulman[+], Mark Hefke[+], Adrian Trifu[+], Martin Soto[*], Danilo Assmann[*], Joerg Doerr[*], Michael Eisenbarth[*]

[*] *Fraunhofer IESE, Sauerwiesen 6, 67661 Kaiserslautern, Germany*
{danilo.assman, michael.eisenbarth, joerg.doerr, martin.soto}@iese.fraunhofer.de

[+] *FZI Forschungszentrum Informatik,Haid-und-Neu-Str 10-14, 76131 Karlsruhe, Germany*
{hefke, szulman, trifu}@fzi.de

Abstract: In the digital planning process of a manufacturing plant several partners like OEM, prime contractor and further service providers are participated usually. Since the partners have partially overlapping views (electricity, mechanical structure, plant controlling) on the same plant to be produced, they have to exchange data during their collaboration. However the syntactical, structural and even semantical differences of output and expected data of partners makes a data integration necessary. Whilst dealing with syntactical and structural differences is rather easy, overcoming semantical differences has several obstacles. In this paper we introduce our ontology based reference model approach to solve this problem.

Keywords: Automobile industry, control engineering, industrial production systems, integration, data models, planning, validation

1. INTRODUCTION

In the planning process for a manufacturing plant a variety of departments (both engineering and organisational) are invariably involved. Whilst the mechanical concept planning is generally carried out by the OEM (original equipment manufacturer) itself all other tasks like electrical planning, PLC programming or development of a plant controlling system are usually accomplished by a prime contractor who delegates some of these tasks to further service providers (see Figure 1.).

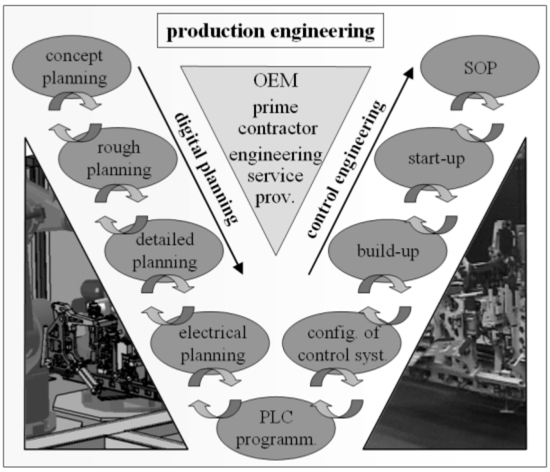

Figure 1: Key aspects of production engineering

Splitting up the production engineering process into independent tasks and delegating them to different domain experts ensures the success of the project. However collaboration of multiple partners has several obstacles:

- Shorter development cycles and increase of variants in a product family makes a full digital planning process necessary. Concepts, concrete process sequences and necessary resources can be defined and verified digitally before the real construction process is initiated. This also means that not only single steps of the planning process but also the exchange of planning data between collaborating partners have to be supported electronically. Further details on this problem are described in (Schmidgall, 2005).
- The second challenge of nowadays is, that data generated by one partner do not necessary correspond to the expected input format of an other partner. These differences are raised on several levels. On the syntactical level partners might use different data encoding formats like UTF8 vs. UTF16. On the structural level one partner might output the 3D construction data using AutoCAD's DXF file format while an other partner expects these data in a scalable vector graphics (SVG) format. The varying domain knowledge and overlapping views (see Figure 2.) on the same planning process of the partners lead to a semantical gap during collaboration. Whereas dealing with differences on the syntactical and structural level is rather an easy task using already existing file converters, overcoming the semantical heterogeneity constitutes an important obstacle.

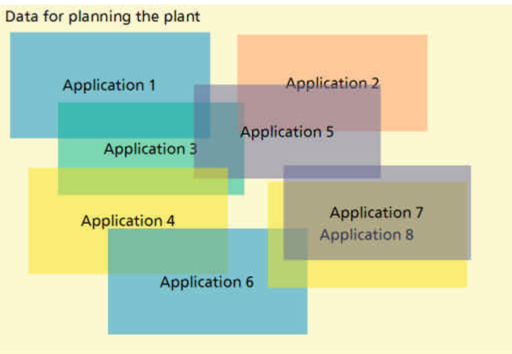

Figure 2: Data overlaps between applications

Thus the aim is to integrate the diverse IT systems of collaborating partners in a single environment. There are two ways to solve this problem. One alternative is to develop a single application (for example Delmia[1]. or Tecnomatix[2] for the automotive domain) that covers all relevant aspects of the planning process or to come up with an approach such as ours that allows us to create a single virtual environment, where already existing tools of the partners can be easily integrated into.

To achieve this aim, we introduce the so called reference model (see Figure 3), that enables to easily integrate partners based on ontologies. The key idea is to represent the semantical meaning of that part of data in each partners data schema that has to be exchanged, using component ontologies. In the second step we introduce semantic bridges to overcome semantical differences between different data schemas. In addition the reference model provides the overall infrastructure and functionality necessary for user interactions, e.g retrieval queries, or for the data exchange and the general ontology management.

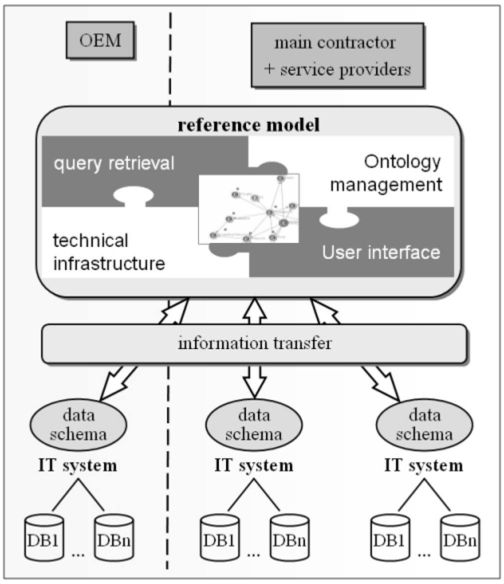

Figure 3: Reference model for inter-organizational information transformation

2. BACKGROUND

2.1 Ontologies

Before the production engineering process starts, each partner has to represent the semantic meaning and interrelationship of internal processes, resources, as well as of data to be exchanged. This can be done by (semi-)automatically extracting information about data structure and content from (un-)structured information sources as well as by representing this extracted information using a common machine-processable ontology representation language. Common languages to represent ontologies are RDF(S)[3]or OWL[4]. In the computer science ontologies are formal models of a domain, that support the

---

[1] http://www.delmia.de/
[2] http://www.tecnomatix.com/
[3] http://www.w3.org/RDF/
[4] http://www.w3.org/TR/owl-features

communication between human and/or machines and so the exchange and sharing of a knowledge base. On a sociocultural level ontologies demand a common understanding of concepts and their relationships. These aspects can be covered by the following definition of an ontology (Gruber, 1993): "*An ontology is an explicit specification of a shared conceptualization*". The following short definition describes ontologies as they are used in our approach. Here ontologies consist of both schema and instance data.

$$O := < C, H_C, R_C, H_R, I, R_I, A >$$

An ontology O is a tuple, consisting of the following: The concepts C of the schema are arranged in a subsumption hierarchy H_C. Between single concepts exist relations (properties) R_C. Relations can also be arranged in a hierarchy H_R. Instances I of a specific concept are interconnected by property instances R_I. Additionally one can define axioms A which can be used to infer knowledge from already existing ones.

2.2 Approaches to ontology mapping

In order to overcome semantic differences and to finally achieve a common integration ontology for exchanging data between all involved partners of the production engineering process, entities of a source ontology have to be transformed into entities of a target ontology by the use of so called "mappings". Much research has been done in the area of (semantic) information integration. Existing information integration systems and approaches (e.g., TSIMMIS (Hammerer, 1995), Information Manifold (Levy, 1986), Infomaster[5], MOMIS[6], Xyleme[7]) are "centralized" systems of mediation between users and distributed data sources, that exploit mappings between single mediated schema and schemas of data sources. Those mappings are typically modeled as views (over the mediated schema in the local-as-view approach, or over the sources schemas in the global-as-view approach) which are expressed using languages having a formal semantics. For scaling up to a real planning project, the "centralized" approach of mediation is probably not flexible enough, and distributed systems of mediation are more appropriate. Furthermore, existing mapping approaches can mainly be distinguished along the following three categories:

- Different representation forms of mappings by the use of a "mapping ontology" or individual mappings between relations and between objects.
- A variety of different identification mechanisms of mappings using "lexical" similarity measurement approaches (string comparison by also considering synonyms etc.), semantic similarity measures (taxonomic similarity (entities that have a common super-concept), relational similarity (entities are similar due to "common" relations to other entities) or by even considering background knowledge).
- Different approaches how mappings are later used for data exchange.

However, none of the proposed solutions has really encompassed the overall mapping process especially considering the evolution and consensus building of semantic bridges. Semantic bridges establish the correspondence between component ontologies. The role of a semantic bridge is to encapsulate all necessary information at conceptual level in a way that each instance represented according to the source ontology is translated into the instance described according to the target ontology. Apart from the semantic correspondence, additional information is needed to further specify the transformation to be performed, e.g. translations of measures like different geometrical properties (e.g., radius and diameter). The nature of semantic bridges may be understood by considering different dimensions, each describing one particular aspect of a semantic bridge:

- Entity dimension: reflects the type of ontology entities being bridged. Entities are either concepts (modeling classes of objects from the real word), relations (modeling relationships between objects), attributes (modeling simple properties of objects).
- Constraint dimension: respects the necessity to constraint the transformation based on source instances values. Constraints act as conditions attached to transformations. The transformation is executed if the condition holds, and vice-versa.
- Cardinality dimension: reflects the number of ontology entities being mapped. This dimension determines the number of ontology entities at both sides of the mapping, ranging from 1:1 to m:n.

3. THE REFERENCE MODEL LANDSCAPE

As suggested in previous sections, the purpose of the reference model is to facilitate the exchange of data between the individual partners that cooperate in the digital planning process. As a natural consequence, the focus of the reference model lies on the "intersection regions" between partners (see Figure 3.). These "regions" can be defined as the subset of all produced information that is of interest to at least two of the cooperating partners.
In the course of the planning process, several types of models are used. Conceptually, they reside on three distinct levels of abstraction (see Figure 4.): the instance level, the concept level and the concept bridging level.

[5] http://infomaster.stanford.edu/infomaster-info.html
[6] http://sparc20.ing.unimo.it/Momis/
[7] http://www.xyleme.com/

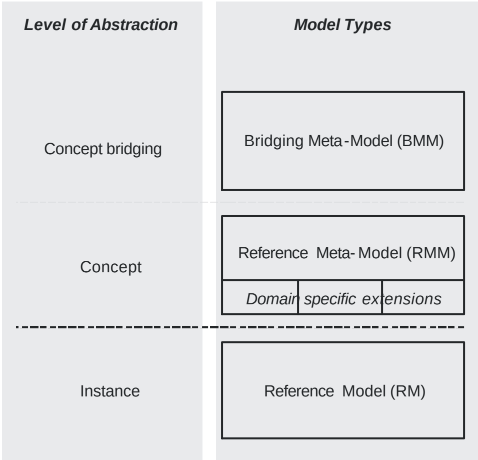

Figure 4: The three levels of abstraction and their corresponding model types

Each level contains models that instantiate concepts defined in the level above, and define concepts that are instantiated in the level below. In the following, we describe each of the levels and provide small examples.

3.1 The concept bridging level

As previously mentioned, the main obstacle to seamless data exchange between cooperating partners is the lack of syntactic, structural and semantic compatibility between the way their tools represent and store information of common interest. As will be shown later, a formal description of each partner-specific concept world in the form of an ontology makes their harmonization possible. This is accomplished with the help of semantic bridges, which are also represented in a separate bridging ontology. The bridging meta-model (BMM) is nothing else than the definition of the concept of semantic bridges, and other associated concepts. The approach that we follow is based on MAFRA, a MApping FRAmework for distributed ontologies (Maedche, 2002).

The MAFRA meta-model is presented in Figure 5. Depending on the entities they map, it defines three types of semantic bridges: concept bridges, attribute bridges and relation bridges, where “concept”, “attribute” and “relation” are the fundamental building blocks of ontologies. Several semantic bridges can be composed into composite bridges, forming a hierarchical structure. A bridge can be augmented with a rule, which can be either a condition or a transformation. Conditions specify restrictions that are checked before the bridge is crossed, while transformations provide support for eventual transformations may be necessary in the mapping. Transformations rely on external services to actually carry out the transformation process on the data. In an instance of a BMM, instances of bridges map instances of concepts, attributes and relations to one another, as will be shown below.

3.2 The concept level

The concept level is the richest and most interesting abstraction level in our approach. At this level, the harmonization between the domain concepts used by the cooperating partners is defined and the handing over of data is modelled through the Reference Meta-Model (RMM). The harmonization between the concept worlds used by different partners is realized as follows. First, each cooperating partner must define the syntax, structure and semantics (Modale-SOTA, 2004) of the data in his

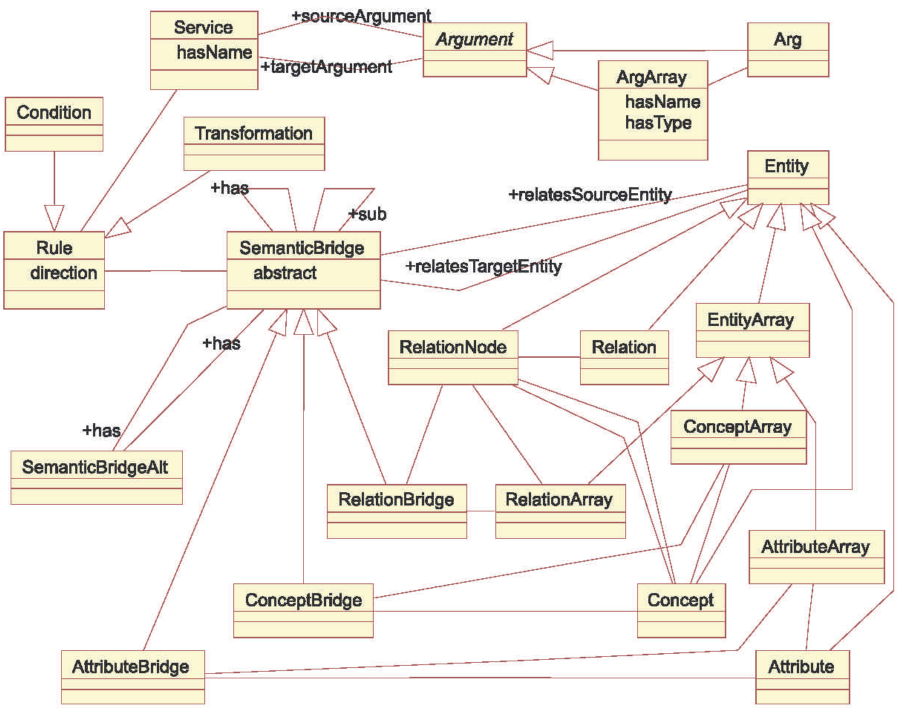

Figure 5: A Bridging Meta-Model (BMM)

own intersection region, in the form of an ontology (i.e. by instantiating "concepts", "attributes" and "relations" from BMM). Following that, the two ontologies are integrated through inclusion (Maedche, 2003) in a so called integration ontology, described by the Reference Meta-Model (RMM). It is conceivable that domain-specific extensions to the RMM could be formalized. Such standard meta-models could speed up the construction of the models, and therefore save time at the beginning of the planning process.

At the same time, a number of bridges are instantiated to describe the necessary mappings and transformations between the concepts of the two ontologies. The bridges are specified in a separate ontology, called the semantic bridging ontology (SBO).

As an example, let us consider two partners (Partner A und B) that need to exchange geometrical data concerning a mechanical component that partner A designs. Partner A uses Tool A to design the component, and partner B uses Tool B to visualize the design. The tools use different file formats (syntax, structure) and semantics for representing and storing geometrical data and schemas: Tool A represents 2D segments as pairs of 2D Points, while Tool B represents them as vectors. The first step is to capture the two schemas in two ontologies. The two relevant ontology fragments relating to our example are shown in Figure 6. In the first ontology, a segment is characterized by an ID, a start and an end point, which in turn has an ID and two coordinates. In ontology B, points are defined similarly, but there is no concept called segment. Instead we have a vector, which contains ID, modulus and an angle alpha.

In order to enable the two tools to interoperate, we need to map concepts, attributes and relations from ontology A to concepts, attributes and relations in ontology B.

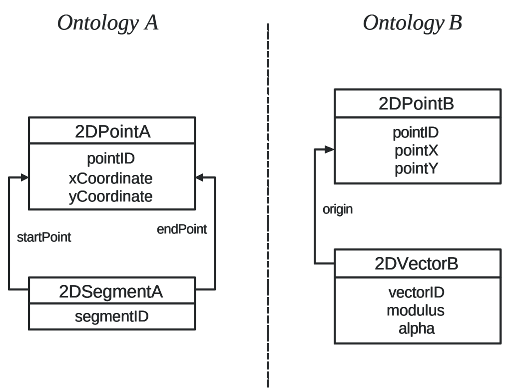

Figure 6: Two incompatible concept worlds

An obvious concept bridge can be instantiated between "2DPointA" and "2DPointB". Their corresponding attributes can also be mapped without difficulty ("pointID" to "pointID", "xCoordinate" to "pointX" and "yCoordinate" to "pointY"). Using a relation bridge, relation "startPoint" can also be mapped to the relation "origin" without problem. In order to transform segments to vectors, we define two attribute bridges between "startPoint" and "endPoint" (belonging to 2DSegmentA) and modulus and alpha (belonging to 2DVectorB). These bridges are "2 to 1" bridges, each augmented with an appropriate transformation. For example, the transformation rule associated to the bridge for "modulus" will use a service called "SrvPitagoras" which will have to be invoked to compute the vector's modulus, based on the coordinates of its two extremities. The two ontology fragments and the semantic bridges connecting them are shown in Figure 7.

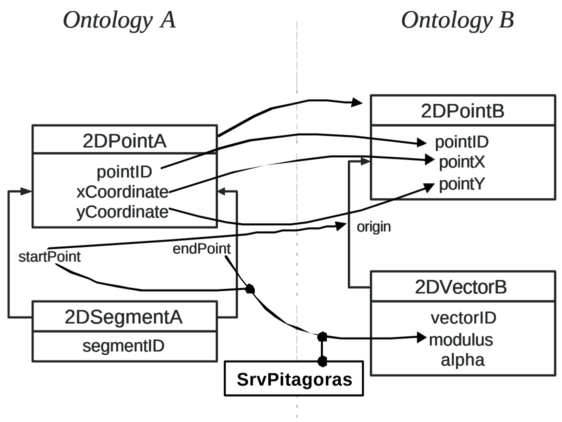

Figure 7: Ontology fragments connected by semantic bridges

3.2 The instance level

On the instance level, we find an instantiated reference model, mapping instances of actual data to another ones. As new data are being produced by partners and the need to exchange it arises, the RM continually increases with instances of relevant concepts (for example, segments, points and vectors). Using the RM and based on the bridges defined in the RMM, data can be automatically transformed from one representation to another, according to the need of the partners.

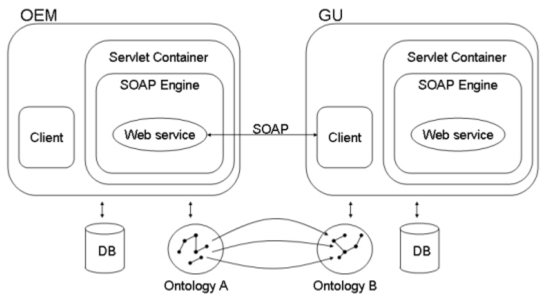

Figure 8: Architecture of the prototype

5. SUMMARY AND FUTURE WORK

One of the main obstacles in the way of realizing a seamless integration between cooperating partners in the field of digital production engineering consists of the syntactic, structural and semantic heterogeneity of their internally used tools. The paper introduces an ontology-based data integration approach, that provides a solid formal foundation, and has the potential to overcome the obstacle. As proof of concept we have also provided a web-service based prototypic implementation of our approach (the architecture of the prototype is shown in Figure 8).

However, many questions still remain open, requiring more research effort to be invested. The main directions that concern us in the short to medium term revolve around the following:

- a methodology that allows an efficient (even automatic) construction of the necessary models and semantic bridges;
- standardized, domain-specific extensions to the RMM, which would allow a very short start-up time for projects;
- issues concerning the integration with each partner's internal processes, in order to achieve minimal disturbances in their existing workflows.

ACKNOWLEDGEMENT

The development of some parts of the ideas mentioned in this article are supported through the publicly funded BMBF project (Deutsches Bundes-ministerium für Bildung und Forschung) MODALE (Modellbasiertes Anlagenengineering, kundenorien-tierte Dienstleistungen für Anlagensteuerung und –kontrolle). More information about this project is available on the MODALE[8] website.

REFERENCES

T. Berner-Lee, J. Hendler, O. Lassila. (2001). The Semantic Web. *Scientific American, May 2001*

T. R. Gruber (1993). Toward Principles for the Design of Ontologies Used fot Knowledge Sharing. *International Workshop on Formal Ontology, Padoba, Italy.*

J. Hammer, H. Garcia-Molina, K. Ireland, Y. Papakonstantinou, J. Ullman, and J. Widom (1993). Information Translation, Mediation, and Mosaic-Based Browsing in the TSIMMIS System . In *Exhibits Program of the Proceedings of the ACM SIGMOD Internation Conference on Management of Data, page 483, San Jose, California.*

A. Maedche, Boris Motik, Nuno Silva, and Raphael Volz (2003). MAFRA – A Mapping FRAmework for Distributed Ontologies. *Proceedings of the 13th European Conference on Knowledge Engineering and Knowledge Management EKAW, Madrid, Spain*.

A. Maedche and B. Motik and L. Stojanovic and R. Studer and R. Volz (2003). An Infrastructure for Searching, Reusing and Evolving Distributed Ontologies. *The Twelth International World Wide Web Conference*.

Alon Y. Levy, Anand Rajaman, and Joann J. Ordille (1986). Querying Heterogneous Information Sources Using Source Descriptions. In *Procededings of the 1st ACM OOPSLA Conference, Portland, Oregon*.

G. Schmidgall, J. Kiefer, T. Bär (2005). Objectives of Integrated Digital Production Engineering in the Automotive Industry. Submitted to the *16th IFAC World Congress, Prague, Czech Republic*

P. Szulman, A. Trifu (2004). MODALE, State of The Art on Information Integration. http://www.modale.de/Modale/cms/Arbeitspakete/AP2/

[8] http://www.modale.de/