Semantic Interconnection of Distributed Numerical Simulations Via SOA

Dipl.-Ing. D. Schilberg, Dipl.-Ing. A. Gramatke, Prof. Dr.-Ing. K. Henning

***Abstract*— The focus of this work is to promote the semantic interconnection of distributed numerical simulations by using a service oriented architecture. The product planning phase usually requires a huge range of various simulations. To effect increasing productivity on the one hand and reducing complexity of simulation chains on the other hand it is essential to examine differing semantics and the possibility of connecting the various interoperability layers. Service oriented architecture provides the technology for loose connection of distributed numerical simulations at data and semantic interoperability.**

***Index Terms*— Semantics, Service Oriented Architecture, SOA, Semantic Interconnection.**

I. Introduction

By reason of low costs, production in low-wage countries has become popular in the last few years. To slow down the trend of outsourcing production to low-wage countries, new production concepts for high-wage countries have to be created. Therefore today's production industry in high-wage countries is confronted with two dichotomies: value orientation vs. planning orientation as well as economies of scale vs. economies of scope.

Developing new concepts means to overcome the polylemma of production, shown in Figure 1, which summarizes the two dilemmas mentioned above. Future-proof production systems have to accomplish the apparent incompatibility of the two dichotomies. To improve the competitiveness of production in high-wage countries compared to production in low-cost countries, it is not sufficient to achieve a better position within one of the dichotomies; it is necessary to resolve the polylemma of production [1]. The research questions pursued within the Cluster of Excellence "Integrative Production Technology for High-Wage Countries" aims at dissolving this polylemma.

Manuscript received July 21, 2008. The depicted research has been funded by the German Research Foundation DFG as part of the Cluster of Excellence "Integrative Production Technology for High-Wage Countries".

D. Schilberg is with the Center for Learning and Knowledge Management and Department of Information Management in Mechanical Engineering of RWTH Aachen University, Aachen, Germany (phone: 0049 241 8091136; fax: 0049 241 8091122; e-mail: schilberg@zlw-ima.rwth-aachen.de).

A. Gramatke is with the Center for Learning and Knowledge Management and Department of Information Management in Mechanical Engineering of RWTH Aachen University, Aachen, Germany (e-mail: gramatke@zlw-ima.rwth-aachen.de).

Prof. Dr.-Ing. K. Henning is with the Center for Learning and Knowledge Management and Department of Information Management in Mechanical Engineering of RWTH Aachen University, Aachen, Germany (e-mail: henning@zlw-ima.rwth-aachen.de).

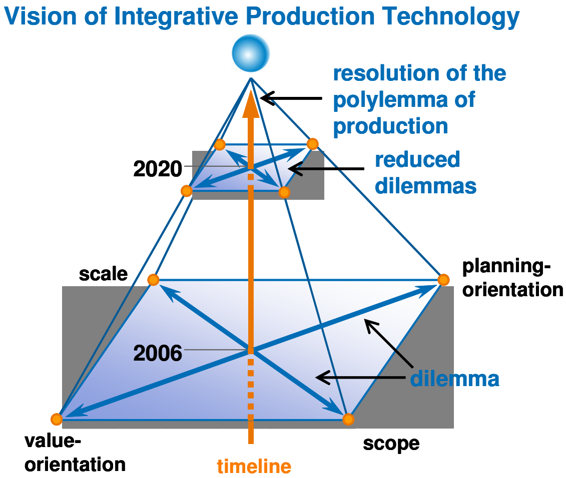

Fig. 1. Polylemma of production

This research cluster unites twenty institutes collaborating for this purpose. Professional competencies of the research partners are domain specific for certain aspects in production processes like moulding, grinding, welding, etc. One of the project`s objectives is to provide a toolbox merging the partners` competencies. This toolbox will support enhancing virtual production systems, making them an essential asset on the global market.

Certain aspects of the production process are well explored and corresponding models describing the processes have been developed accurately. Still missing today are realistic simulations of the whole production chain which have not been realized yet. The possibility of a semantic interconnection between the applied simulations is the research objective. Therefore an application is needed to provide the ability to adapt existing simulations models without redefining or extending them.

The combination of existing simulations covering and addressing specific aspects in process chains suggests creating a new and augmented comprehension of process chains as a whole. Using adequately simulated input parameters, which reflect the production history, to feed the next simulation in the chain, will most probably produce better results for that specific simulation than using standard assumptions or pre-computed values to parameterize the model. While the overhead for modelling and planning will be increased by simulating entirely interlinked processes, the expected results will be more accurate. Hence criterion for judging this highly planning oriented approach is the better value of the benefits in terms of insight, understanding, efficient technical processes, lower production costs or higher product quality without ignoring the costs of creating simulated process chains [2].

II. Related Work

Over the last decade the research area of semantic mapping has made a great leap forward. It is the aim of many projects to increase the efficiency of program chains by interconnecting applications. There is a huge range of varying approaches to achieve the actual interconnection. Approaches can be assigned to two conceptual streams in semantic mapping: one is the ontology-based and the other is the model-based semantic mapping. Both approaches start addressing two conceptual simulation models where semantically related concepts have to be identified. There is a main difference from comparing the model-driven approach and the ontology-based approach. The model-driven approach aims at finding the semantic mapping directly starting from the two models, say simulation A and simulation B, deriving information from the mapping between the data of the simulations A and B. The ontology-based approach however is working indirectly, by means of a reference ontology [3].

To introduce this, two conceptual streams project examples will be presented briefly. Semaphore is a model-based attempt. It is a syntactic and semantic mapping tool. The basic idea behind this tool is to configure mappings between the present information formats by defining the mappings on platform-independent models of the information formats. After defining mappings between a source model and a target model, transformation code is generated to be used on the instances of the source model. The transformation code will convert an instance of the source model into an instance of the target model according to the mappings performed on both of the models [4].

An example ontology-based attempt is the project SOIRA. The main objective of the SOIRA architecture is to improve the capabilities of semantic mapping and the translation between ontology mediated semantic view and local data sources. Fulfilling this aim will force a flexible information extraction, information integration and knowledge acquisition from heterogeneous, distributed, autonomous information sources [5].

Solutions evolved within these and similar projects and approaches are giving a new direction to research of mapping tools. Possibilities of semantic interconnection will be augmented. But a major issue when interconnecting simulations are the simulations themselves as they need a decent way to exchange their data. The described approaches are not broad enough to supply a framework to integrate the different kinds of applications to simulate a whole production process.

A method for a semantic interconnection of distributed numerical simulations is presented in the next section.

III. Semantic Iterconnection of Distributed Numerical Simulations

A. *Classification of distributed numerical simulations*

A classification of distributed numerical simulations is needed, because each simulation is developed for solving a specific problem. According to the problem the simulation uses specific initial values and dedicated algorithms.

To clarify the different classes of simulations Figure 2 shows three classes of simulations using unified modeling language (UML). All of these can be used in a process chain of a toothed wheel. Every simulation uses different attributes and methods. Therefore it is not possible from within one simulation to use attributes or methods of another simulation even if these two simulations are interconnected. Furthermore the different simulations do not operate within the same dimensions and number systems and not all output data is needed for the following simulations. The recast-simulation mentioned in Figure 2 uses a finite element grid to describe the geometry used; all relevant parameters are assigned to the knots of the finite element grid. The microstructure simulation examines in addition to that a field of phases. The generated results of both simulations can not be coupled on the data level, because each of the two programmes is not able to interpret the results for the data of the other simulation.

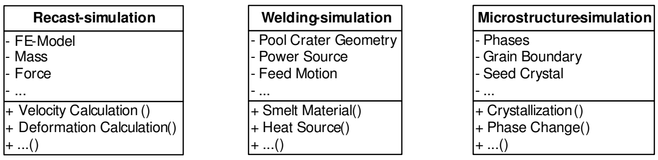

Fig. 2. Representation of simulation classes using UML

Most of the simulations provide the input and output data in proprietary formats that are interpreted by specific pre- and postprocessors. Interfaces for inter program communication are not implemented. Because of these circumstances it is necessary to establish an elaborated framework for simulations` interconnection [6]. The framework has to connect each simulation to exchange data and the context of the data (semantics). The interconnection based upon exchanging data can be provided by the implementation of converters and sequencers. They are converting the output files from one simulation to the input files of the next simulation. The implementation and integration of these programs is a trivial problem [7]. The main challenge of interconnecting simulations and the focus of this work, is interconnection on the level of semantics. Semantic interconnection is needed to assure that the chosen simulations, their order and the converted data are reasonable [8].

Within the following section the Service Oriented Architecture (SOA) is chosen for building up the framework.

B. *Service Oriented Architecture*

Service Oriented Architecture is used to describe how distributed services can be reached by middleware. In the considered case the distributed services are distributed numerical simulations. The middleware is defined as computer software which interconnects software components or applications. The software, consisting of a set of enabling services, allows multiple processes running on one or more computers to interact across a network [9]. SOA is mainly used to deploy business processes, but it is appropriate to connect every kind of application at the data layer.

Dealing with application interconnection a SOA has to handle complexity. That is why generalizations which are giving a better understanding of the production process have to be prepared. Contrary to abstraction activities, generalization ignores many details and produces "general" ideas. Therefore the collection of types and the analysis of commonalities have to be performed.

ISBN: 978-988-98671-0-2

The semantic heterogeneity and divergence of the simulations hinder the process of generalization. Because of representing commonalities of two entities in semantically various ways the identification of differences is difficult. Ontological analysis sets a foundation for generalization by defining the properties of the entities. Ontological analysis for application integration and SOA encourage generalization processes. What should be solved by application interconnection via SOA is the problem of achieving semantic interoperability between different systems. Therefore predefined ontologies will be adapted for the use in a required process chain. For the mappings between the simulations, ontologies are needed.

Figure 3 presents the interconnection of four simulations to one simulation chain using the Business Process Modeling Notation (BPMN) [10]. The chain can easily alternate passing every decision node. The input data will be allocated through a sequencer and the output data will be processed via a parser for visualization. It could also be processed by another sequencer of an additional simulation. All applications that are included in the simulated process chains are structured by a framework that is based on the requirements and design patterns of a SOA [8].

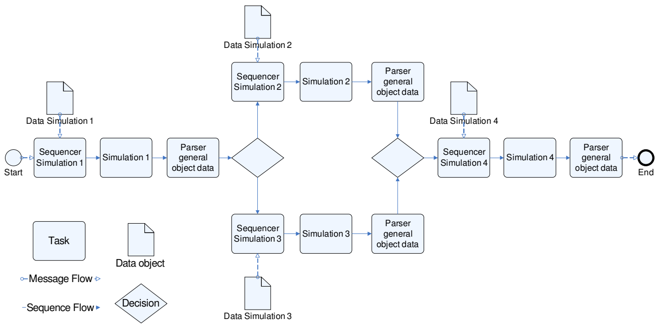

Fig. 3. Flow chart of the simulation interconnection

SOA is mainly used to establish clear syntactical rules and definitions to address and bind services. By using established SOA design patterns the effort for syntactic interconnections is minimized [11]. In order to realize a semantic interconnection of simulations the sequencer and the parser have to be supplemented by additional functions. Sequencer and parser are used as interfaces for the inter simulation communication. The following section describes an approach for semantic interconnection that can be implemented via a SOA.

C. *Semantic interconnection*

Semantic interconnection is defined in terms of feasible mappings between the interoperability layers of systems. In case of semantic interconnection of distributed numerical simulations four layers of interoperability have to be considered [12]. The different layers are the data layer, object layer, component layer and application layer. At each layer the quota of syntactic and semantic interoperability differs in comparison to the overall interoperability. Figure 4 shows the mentioned interoperability quotas qualitatively. At the data layer there is no semantic interoperability necessary. All information (data) is exchangeable by using a defined syntax. The object layer is mainly based on the syntax used. The quota achieved by communication based on semantics is one-fifth of the overall communication. The simulation models are represented in the object layer. The component layer comprises the modules and module interfaces of an application. The distribution between syntax and semantics is still dominated by syntax but semantics gets more important for the communication. At the application layer communication is just reasonable by considering the semantics, because the semantic provides the context. To draw a conclusion from Figure 4 for the use of a semantic mapper it is necessary to focus on the component and application layers. Because of the low percentages of semantic interoperability at the data and object layer semantic interconnection is inadequate. The described classes of distributed numerical simulations do not allow any separation of components and applications. For the semantic interconnection only the application layer and therefore especially the semantic interoperability is of interest.

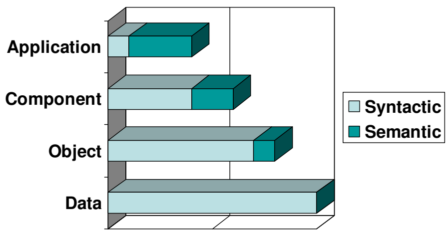

Fig. 4. Interoperability Layer according to Obrst [12]

A semantic mapping tool is needed to interconnect different distributed numerical simulations and their functions. The semantic mapper will be augmented by developing a library of matrices correlating with the ontology. The matrices have to consist of simulation data. The knowledge of experts who developed or use the distributed simulations is represented in the ontology. Every matrix reflects one simulation. This collection of matrices includes all necessary input and output data according to any simulation. The information about feasible orders of simulations will be generated by using the ontology.

Figure 5 shows a rough schematic diagram of the interoperability system. Before using the semantic mapper a production process that will be simulated has to be selected. After selection a chain of simulations that fits to the real process will be arranged by the semantic mapper.

The task of the semantic mapper consists of the objectives. The semantic mapper browses through the matrices, collects points of interoperability and correlates these points with the requirements of the process chain by using the ontology. Thereafter all simulations essential for the process chain have to be identified by the mapper. The necessary input data needed by all used simulations has to be provided, either through a database or a human machine interface (HMI).

Without a detailed transformation of the domain specialist’s knowledge the proper use of the semantic mapper is not reasonable. The quality of the semantic mapper is directly depending on the information that is included in the ontology.

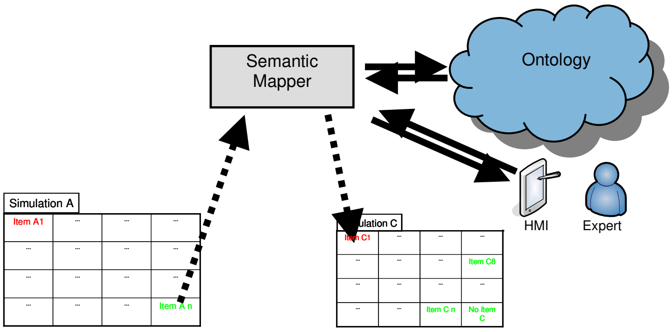

Fig. 5. Interoperability of the semantic mapper

The following section describes in which test case the present semantic mapper including the knowledge of the domain specialist will be put into practise.

IV. Case Study

One project objective is the production of a toothed wheel for a gearbox. The toothed wheel will be formed by a recast process. After that the final shape is given by a metal-cutting process. The definitive material properties are given by a heat treatment. The concluding process step is to weld on a ring.

The manufacturing of the described toothed wheel with a welded ring is simulated on two different levels of granularity. At the macro level, structural forming, heat treatment and welding simulations are needed. Also needed are structural simulations for micro structures at the micro level. In addition, there is a transfer level, which is needed to step up the micro structural data to the macro level (Figure 6).

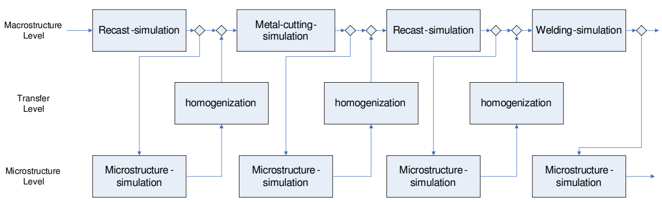

Fig. 6. Test case toothed wheel with welded ring

Every used simulation is a service of SOA. The user has to select a specific process chain. The simulations that are necessary for the simulation chain of the whole process will be chosen by the semantic mapper. The semantic mapper proves which of the essential data is provided by which simulation and which of the data has to be added by a specialist through the HMI. After the identification of the essential data, each value that is transferred from one simulation to another is prepared by the semantic mapping to fit the requirements of the next simulation.

Since the interconnection of simulations to a process chain by a SOA on data level and semantic level will open up new vistas. It could become possible to examine the macro structure level, separately or in combination with the micro structure level, without affecting the planning effort for the manufacturing process [5].

V. Conclusion

The need for semantic interconnection of distributed numerical simulations derives from the need of reducing the planning effort of production processes and therefore the requirement of flexible simulation chains. Investigating in this research means increasing the insight and understanding of the scientific and the technical processes, the product quality or the market advantages as well as reducing production costs. This paper has introduced a concept of semantic interconnection of distributed numerical simulations. It also presents how it can be realized using a semantic mapper. The approach shown of semantic interconnection at the application layer has the advantage that the intellectual process of producing the knowledgebase of the ontology lays in the scope of responsibility of the domain specialist. Close cooperation between semantic mapper developers and domain experts during the implementation process affects the loss of semantic information. It can be reduced to a minimum.

To increase the understanding of how the semantic interconnection of distributed numerical simulations can be successfully implemented is the main research goal. The development of tools and techniques to support the interoperability of simulations along the process chain supports the achievement of the main goal.

References

[1] G. Schuh, F. Klocke, C. Brecher, R. Schmidt (Ed.), *Excellence in Production*. Aachen: Apprimus Verlag, 2007.

[2] P. Cerfontaine, T. Beer, T. Kuhlen, C. Bischof, *Towards a Flexible and Distributed Simulation Platform*. In: ICCSA Perugia, 2008.

[3] K. Arnarsdóttir, A.-J. Berre, A. Hahn,M. Missikoff, F. Taglino, *Semantic mapping: ontology-based vs. model-based approach Alternative or complementary approaches?* In: Proceedings of the Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability. Luxembourg: 2006.

[4] A. Limyr, T. Neple, A. J. Berre, B. Elvesæter, *Semaphore – A Model-Based Semantic Mappig Framework*. In: BPM 2006 Workshops. Berlin: Springer-Verlag, 2006, LNCS 4103, pp. 275-284.

[5] L. Zhang, J.-G. Gu, *Ontology Based Semantic Mapping Architecture*. In: Proceedings of the Fourth International Conference on Machine Learning and Cybernetics. Guangzhou, 2005.

[6] X. Chen, W. Cai, S. J. Turner, Y. Wang, *SOAr-DSGrid: Service-Oriented Architecture for Distributed Simulation on the Grid*. In: Proceedings of the 20th Workshop on Principles of Advanced and Distributed Simulation (PADS'06). Singapore, 2006.

[7] W. T. Tsai, Q. Huang, X. Sun, Y. Chen, *Dynamic Collaboration Simulation in Service-Oriented Computing Paradigm*. In: Proceedings of the 40th Annual Simulation Symposium (ANSS'07). Norfolk, 2007.

[8] D. Schilberg, A. Gramatke, K. Henning, *Koppelung von heterogenen numerischen Simulationen durch eine Service-orientierte Architektur*. In: Automation 2008. Baden Baden, 2008.

[9] T. Erl, SOA *Principles of Service Design*. Upper Saddle River, NJ: Prentice Hall (2007)

[10] Business Process Modeling Notation Specification, OMG Final Adopted Specification, 2006.

[11] I. Melzer, *Service-orientierte Architekturen mit Web Services*. Heidelberg: Spektrum Akademischer Verlag, 2007.

[12] M. Daconta, M., Obrst, L., Smith, K.: *The Semantic Web: The Future of XML, Web Services, and Knowledge Management*. 2003.