Process Integration and Design Optimization Ontologies for Next Generation Engineering

Massimo D'Auria and Roberto D'Ippolito

Noesis Solutions N.V., Gaston Geenslaan 11, B4 – B3001 Leuven, Belgium
{massimo.dauria,roberto.dippolito}@noesissolutions.com

Abstract. The recent years have seen a significant increase in ontology research in the field of engineering due to the needs of domain knowledge capturing, reusing and sharing between software agents and/or engineers.

The increasingly available enhancements of computing resources have considerably increased the role of Computer-aided engineering applications in the product development process by saving time and costs. In this context a key role is played by the Process Integration and Design Optimization tools that facilitate and automate the integration and interoperability of all different Enterprise Applications involved in the simulation of engineering tasks. However, all these tools inherit a severe limitation about the semantic meaning of the tools they automate.

This paper proposes a platform independent Process Integration and Design Optimization (PIDO) ontology that aims at providing an extensive mapping of the PIDO domain, with a special attention to its adoption in software applications. A comparison with existing similar ontologies and an explanation of the reasons that lay behind the classes and relations introduced is provided. Finally, a real application case is illustrated along with the related main benefits.

Keywords: ontologies, optimization, process integration.

1 Introduction

In the recent years a lot of research has been carried out on ontology definition and development due to the growing interests and needs in sharing and reusing the knowledge between different artificial intelligent systems. A clear definition of what is an ontology has been proposed by A. Gruber [1, 2] where he says: "*An ontology is a description (like a formal specification of a program) of the concepts and relationships that can formally exist for an agent or a community of agents.*"

In the context of computer science the ontologies are the means by which the knowledge of a specific domain can be formalized in a set of concepts and relationships between couples of concepts. The ontologies in this sense provide a model representation of the domain knowledge that they aim to describe and support the reasoning about the concepts and relations defined for the domain considered. As it is stated in [4] the ontologies enable the machines to discover, combine and compare knowledge between different databases. Noy and Mc Guinnes in [6] identified the main advantages of building an ontology that can be summarized as follows:

Y.T. Demey and H. Panetto (Eds.): OTM 2013 Workshops, LNCS 8186, pp. 228–237, 2013.

- Share common understanding of the structure of information among people or software agents
- Enable reuse of domain knowledge
- Make domain assumptions explicit
- Separate domain knowledge from the operational knowledge
- Analyze domain knowledge.

Hence by developing and using ontologies in computer science knowledge and information can be more easily found, shared and combined by users and software agents. Navigli et al. [3] declared that ontologies, i.e. semantic structures encoding concepts, are the backbone of the Semantic Web [4]. Similarly the PIDO ontology aims to build the foundations for a true semantic workflow for engineering that can enable a workflow engine to dynamically create simulation workflows by autonomously discovering and selecting annotated available services without requiring human intervention.

1.1 PIDO Challenges and Benefits

Thanks to the great performance enhancements of available computing resources, the applications of Computer-aided engineering and virtual prototyping are becoming more and more relevant in the product development process (PDP). In many markets (automotive, aerospace, consumer electronics, etc.) computer-based simulations are used to study and analyze the behaviour of complex physical systems. Many tools support engineers in performing different design engineering tasks such as computer-aided design (CAD), finite element analysis (FEA\FEM), multibody analysis and multidisciplinary design optimization [7, 8, 9]. This trend is indeed justified by the considerable reduction of time and costs that is achieved with virtual prototyping. Currently available CAE tools indeed have reached a so relevant maturity such that within innovative companies most of the design verification tests during product development process are now done via computer simulations on virtual prototypes rather than building and testing expensive physical prototypes. Stress, vibration, thermal, are just few examples of analysis that companies perform routinely. However, analysis answers only part of the Product Design question: "*How does the system I have designed behave under the given working conditions?*" Simulation often lags behind the actual design process. Furthermore, design requirements are shifting and becoming more demanding and interrelated between each other. There is a multitude of Design Performance characteristics spanning many different engineering disciplines. Some of these characteristics are conflicting and must be optimized concurrently. Solving such complex problems lies outside the scope of what the current CAE technology can provide. In this context a key role is played by the Process Integration and Design Optimization tools (PIDO) that facilitate and automate the integration and interoperability of all different enterprise CAE applications involved in the simulation engineering tasks by means of the creation of Simulation Workflows. In this way design engineer can focus more on the design aspects of the problem they are addressing rather than wasting time in repetitive, tedious and error-prone tasks like solving interoperability issues with several enterprise applications, platforms, services or implementing best practices explicitly. PIDO tools provide a number of key enabling technologies where the design space can be automatically explored and visualized, gaining the critical insights into the dynamics of the problem. These tools provide also sets of built-in optimization algorithms that enable the engineers to easily find the optimum solution for the design problem addressed.

1.2 Motivations for a PIDO Ontology

However, one severe limitation that both CAE and PIDO tools share is related to the 'meaning' of what they simulate or automate. CAE and PIDO tools do not provide yet means for the design engineer to explicitly express more knowledge about what is simulated or automated, i.e. if a specific design task is a fluidodynamic or a static analysis or what disciplines are involved in a specific problem. This knowledge is currently implicitly captured but not exploited for the benefits of solving the design problems. As such, the natural evolution of these tools is in implementing an ontological layer that allows capturing, managing and reusing the design knowledge already captured and of the new knowledge that can be explicitly expressed by the design engineers. This knowledge, once captured, can be shared between people and software agents of different enterprises and in the context of virtual prototyping.

In this paper a PIDO ontology is proposed in order to leverage the benefits coming from the use of ontologies for the benefit of product design engineering. By developing and using this ontology a first step is taken towards the concepts of a semantic simulation workflow. The PIDO ontology has been developed within the iProd FP7 EC joint research project [5] where the advantages of having a semantic description of this domain in an ontology are fully exploited.

In the following sections 2 and 3 of this paper firstly an overview of the existing typologies of workflows and workflow management systems is provided, then candidate ontologies for simulation workflow and optimization process that have been investigated are presented. In section 4 the PIDO ontology developed is described in terms of main concepts and relationships introduced in order to capture the domain. Section 5 presents one of the case studies addressed in iProd where this ontology has been used to store and retrieve from the knowledge base the necessary information to execute the simulation workflow related to a typical virtual test performed in the automotive PDP. Section 6 in the end summarizes the conclusions of the results achieved by the research work reported in this paper.

2 Workflow Domain

The concept of workflow from the most general point of view can be defined as a sequence of subsequent work activities where each of them follows the previous one without delay or gap in order to perform a specified target job [19]. Workflow can be seen as an abstraction of or a view on any real work. Workflow management systems allow the users to define, modify, control and share workflows, i.e. the activities associated to a business process, simulation process or generic process [19]. Workflow management systems automate redundant tasks allowing a better, cheaper and faster management of the processes. Most of the workflow systems are useful in order to integrate other different systems used by an organization (document management systems, databases, production applications etc.) or software tools (as in case of simulation workflows).

Nowadays there exist plenty of workflow management systems that serve different purposes, provided with various features and based on several workflow languages. An instance of a workflow may involve a series of human tasks rather than tasks that can be executed by tools, machines or software codes. These tasks often need to be repetitively performed.

When considering the context of engineering disciplines, mainly two different workflow categories can be identified: Business Process Workflows and Simulation or Computational Workflows. Within the context of Computational workflows the most important features adopted to evaluate the soundness of a workflow system are: existence of a neutral representation of the computational workflow; distributed computation capability; possibility to implement a set of control flow patterns like those described in [20]. Especially for Simulation Workflows, a big effort has been put in place in order to enable the interoperability and reusability of automated computational design workflows between different tools and enterprises. In this view, the decoupling (or at least loose coupling) of the logic of the computational workflows from their implementation has become fundamental. Hence it is important to have a neutral description of the computational workflows that, in most cases, is achieved by adopting a commonly standardized XML document representation of the workflow like partially done in the EU FP7 project Crescendo [10, 11, 12], where an XML computer readable representation has been drafted in order to allow the interoperability between different computational workflows that can be executed on different platforms and exchanged across different organizations. With this approach no semantic annotation are involved and a thorough conversion of the standardized XML representation to the specific workflow management system is always required and it may lead to mismatching or incompatibility problems between the several platforms and data objects. As a consequence, not all the systems support all the features or patterns and if the conversion is not done consistently by keeping into account all the interoperating systems then a workflow composed in a specific tool may be translated in a completely different one when used in another environment. However, this first tentative tries to respond to the need to formalize knowledge in a neutral format and to share this knowledge across different applications.

This severe issue can be overcome by introducing an ontology neutral standard description of a simulation workflow as the PIDO ontology proposed in this paper. The ontologies allow the definition of the simulation workflow from a higher level, providing a more general representation of the domain that can be easily extended and specified to support completely the different simulation workflow management systems available. The ontological standard representation of a simulation workflow enables the federation of the existing tools rather than their integration (where conversions between the different representations may lead to inconsistencies of the data).

In this context cross-organization simulation collaboration, plug-and-play interoperability between heterogeneous and independent-developed simulation workflow assets and execution time reduction are the major benefits that such approach based on the semantic description provided by the ontology and proposed in this paper can deliver.

3 Simulation Workflow and Optimization Ontologies

In this section a brief literature overview of existing ontologies that aim to capture the knowledge of process integration and design optimization domain is provided. An ontology that covers the simulation workflow and design optimization domains in an exhaustive way does not exist yet. Few attempts have been made in this direction but all have limitations to the scope of the specific application for which they have been developed. These limitations are justified by the fact that the ontologies should not include all the possible information about the domain, but only all those that are relevant and necessary for the specific application for which the ontology itself has been developed. The Workflow Ontology [15, 16] for instance has been created with the objective to capture both sequential and state-based workflows [14]. The ontology captures the different aspects of collaboration workflows and therefore contains concepts more related to a business workflow rather than a simulation workflow. Furthermore the coverage and flexibility of this ontology have been evaluated taking into account only two different collaborative workflows described in the literature like DILIGENT and BiomedGT and so from our point of view is also not general enough to cover any type of collaborative workflow. The ontology for simulation optimization (SoPT) has also been investigated and it includes concepts from both conventional optimization/mathematical programming and simulation optimization. SoPT aims at describing simulation optimization methods and help to detect the correct tool for each specific case and to facilitate component reuse, especially in systems where simulators and optimizers are loosely-coupled. The top-level abstract classes of SoPT are Optimization Component, Optimization Problem and Optimization Method. The relationships among them are shown in Fig. 1. As it is clear from the figure this ontology doesn't contain any concept or relationship related to simulation workflows. Nevertheless, the basic ideas and concepts were taken into account into our PIDO ontology in order to describe the optimization domain. ONTOP, or the Ontology for Optimization, was developed at the University of Massachusetts Amherst with the object to facilitate Engineering Design Optimization (EDO), allowing the instantiation of multiple design optimization models under a single optimization type as well as the creation of multiple model revisions using a single method. The preliminary work with ontologies began with the development of a Finite Element Model (FEM) knowledge-capturing tool, ON-TEAM. ON-TEAM, or the Ontology for Engineering Analysis Models, provides engineers with an ontological framework designed to capture engineering analysis model knowledge [17, 18]. This prototype knowledge framework was founded on the "concept that engineering analysis models are knowledge-based abstractions of physical systems, and therefore knowledge sharing is the key to exchanging, adapting, and interoperating Engineering Analysis Models, or EAMs, within or across organizations,". ONTOP was developed as a knowledge framework tool to incorporate standardized optimization terminology, formal method definitions, and higher-level EDO knowledge. ONTOP's structure affords engineers the ability to approach design optimization problems within an established optimization knowledge base, providing a means to quickly identify feasible optimization techniques for a given design optimization problem.

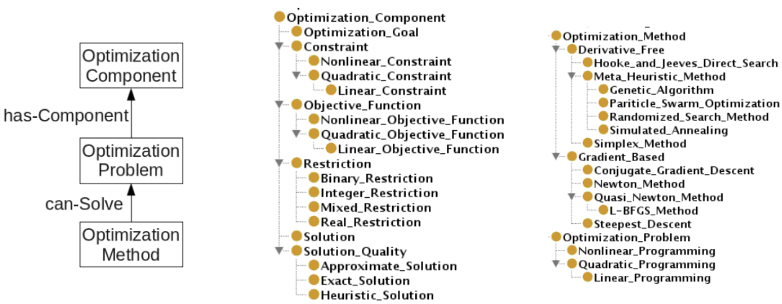

Fig. 1. Main classes of SoPT ontology

4 PIDO Ontology

In this section the Process Integration and Design Optimization ontology is introduced. This ontology aims at formalizing and capturing all the knowledge related to simulation process and design optimization domains. The goal of this ontology is therefore to gather all the necessary information needed to perform a generic simulation task, e.g. virtual test or simulation design optimization. It also provides a formalization of a generic design improvement cycle so that the iterative nature of the design process can be effectively captured. This ontology therefore includes the definition of typical design optimization problems along with a description of the methods and algorithms exploited to solve the considered iterative cycles.

Fig. 2 and Fig. 3 show the fundamental concepts and relationships contained in the PIDO ontology. In most of simulation workflow tools there is the possibility to link to a simulation project one or more simulation workflows representing various simulation tasks. A generic simulation workflow is composed by a sequence of items that are connected each other in order to create the workflow. Depending on the workflow management system adopted, there can be several different items that are put in a user defined sequences so to allow formalization of a specific simulation task. On top of the captured simulation process numerous types of methods can be applied. Also the set of methods available is dependent from the workflow management system used, but some basic methods like the one for performing an optimization process or a design of experiment (DOE) [13] are supported by most of existing systems and have therefore been included in this ontology. For the sake of simplicity and readability only few methods are here reported, but other already exists or can be very easily added as subclass of WorkflowMethod class. An optimization method, for example, is a possible specialization of the more general concept WorkflowMethod, it has constraints and objectives and uses an optimization algorithm in order to solve an optimization problem. (See [5] for more details on the ontology)

However, due to the complexity of the process examined, an iteration of the simulation workflow is computationally expensive and may take hours or even days for the evaluation. In this case surrogate models that mimic the behaviour of the process defined by the simulation workflow turn to be very helpful to find a solution to the optimization problem and are therefore massively used in most real application cases. Surrogate models are built on the base of datasets that are provided by workflow method and obtained by running a number of times the simulation workflow.

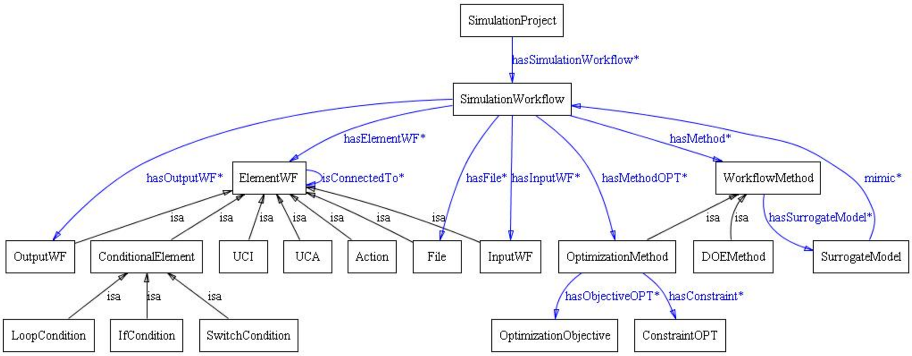

Fig. 2. Main concepts and relationships that formalize a generic simulation workflow in PIDO ontology

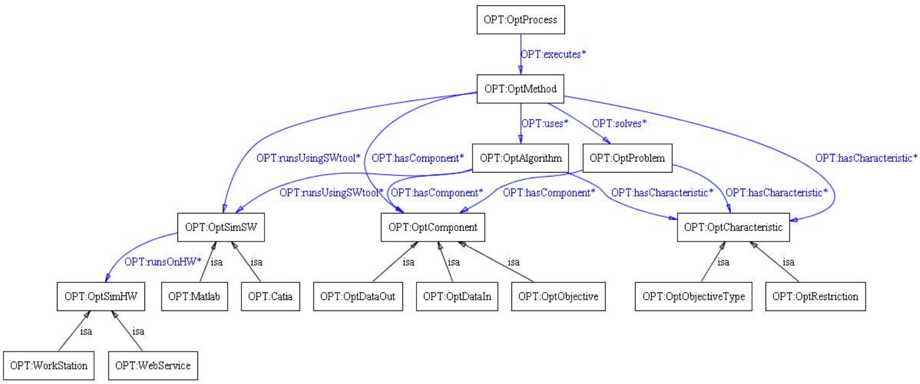

Fig. 3. Main concepts and relationships that formalize the design optimization process in PIDO ontology

The concepts and relationships represented in Fig. 3 are focused on the formalization of a generic design optimization process trying to cover all the entities that occur in an optimization task. As it is shown in the figure an optimization problem is solved by an optimization method by applying one or more optimization algorithms in order to provide the solution. All three entities have some characteristics (that categorize them) and are accompanied by several components (data, objectives, etc.) in order to be performed. The above mentioned workflow problems are implemented by executing some processes that call specific software (SW) tools running on supporting hardware (HW). There is no limit to expanding the problem, method and algorithm entities contained in the ontology, as there are hundreds variations of them. Equally expandable are the SW & HW subclasses that currently focus on the Simulation Workflow applications and are dependent on the product under design/development.

5 Case Study

The PIDO ontology can be used to support the storage and retrieval of the necessary information to execute virtual tests and product design optimizations. These tasks represent the typical simulation processes performed in the PDP of automotive, aerospace and home appliance industries. In this section, an automotive case study is presented where the PIDO ontology have been used to support the execution of a virtual test for a new car door designed by Pininfarina. The virtual design process of a new car door in Pininfarina involves four main simulation tasks: CAD model creation; meshing of the created CAD model; finite element analysis preparation and virtual test execution. These simulation tasks are performed by different tools and are integrated in a seamless way in a simulation workflow that allows automating the execution of this process. The entire simulation process is illustrated in Fig. 4. The objective of the virtual design process here presented is to design a new car door of the NIDO vehicle designed by Pininfarina. In this specific process different versions of hinges have been designed and need to be tested on the virtual prototype of the door. Once the full process is set up the virtual test execution is automated and results delivered for each hinge version. In this case the virtual test consists of verifying the door sag in order to guarantee that the door doesn't fall when opened.

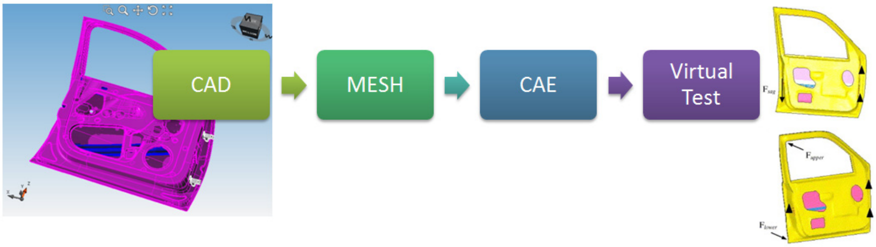

Fig. 4. Virtual design process of a new car door in Pininfarina

This virtual test is captured by the simulation workflow shown in Fig. 5 (created with Noesis Optimus software tool) that is composed of the following elements: input parameters "Config" and "Design Input"; output parameter MaxDisplacement; input file "pininfarina_use_case.dat"; output file "pininfarina_use_case.f06" and an action "NASTRAN" that executes the FEA tool NASTRAN for the stress analysis. All the elements present in this simulation workflow are covered by the concepts present in the above mentioned PIDO ontology. The proposed example has been implemented in iProd in order to validate the PIDO ontology and to proof that this ontology can be easily used in a real application context.

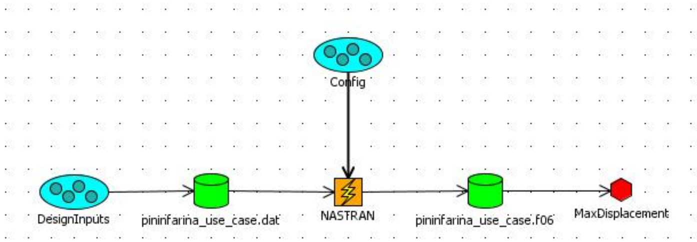

Fig. 5. Simulation workflow that performs the virtual door sag test in Pininfarina case study

The main benefits achieved with this current approach relate to the capability to store and re-use knowledge related to the simulation workflow, to the optimization method used for the solution of the specific problem and of the different elements that lead to a design solution (e.g. constraints, boundary conditions, etc). In this way the design engineer can actually query the knowledge and re-use it for new or similar simulation workflows as well as gather knowledge about the best optimization approaches that have been used in the past for similar problems. This helps the efficient reuse of knowledge by reducing manual steps, by reducing the number of possible mistakes and by avoiding solving two times the same problem. As such, the PIDO ontology constitutes an effective foundation to build semantic workflows where simulation and design improvement knowledge can be captured, re-used and operationalized.

6 Conclusions and Future Works

This paper presented the Process Integration and Design Optimization ontologies developed within the EC 7th Framework joint research project iProd. Firstly a brief introduction on what is an ontology, why develop an ontology and the importance of the role of the PIDO tools in the context of the CAE and virtual prototyping is given. Next an overview of the existing typologies of workflows and workflows management systems is provided. Different existing ontologies for simulation workflow and optimization process have been described highlighting the respective limitations. A PIDO ontology is then introduces and explained in terms of fundamental concepts and objectives. Finally an application case study in the automotive industry implemented in iProd has been used to validate the developed ontology and to proof its usability in a real application context. The proposed ontology is the basis for the future development of a semantic workflow that dynamically creates simulation workflows on the base of the discovered and selected available computing resources without the necessity of the human intervention.

Acknowledgement. The research leading to these results has been supported by the iProd collaborative research project, which has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement number 257657. The authors are solely responsible for its content. It does not represent the opinion of the European Community and the Community is not responsible for any use that might be made of the information contained therein.

References

1. Gruber, T.R.: A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition 5(2), 199–220 (1993)
2. Gruber, T.: Toward Principles for the Design of Ontologies Used for Knowledge Sharing. International Journal Human-Computer Studies 43(5-6) (1995)
3. Navigli, R., Velardi, P.: Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites. Computational Linguistics (MIT Press) 30(2) (2004)
4. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American Magazine (May 17, 2001)
5. http://www.iprod-project.eu/
6. Noy, N.F., McGuinness, D.L.: Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880 (March 2001)
7. Laplante, P.A.: Comprehensive dictionary of electrical engineering, 2nd edn. CRC Press (2005)
8. Meguid, S.A.: Integrated computer-aided design of mechanical systems. Springer (1987)
9. Matthews, C.: Aeronautical engineer's data book, 2nd edn. Butterworth-Heinemann (2005)
10. http://www.crescendo-fp7.eu/
11. Wenzel, H., Gondhalekar, A., Balachandran, L., Guenov, M., Nunez, M.: Automated generation of Isight-Models through a neutral workflow description. In: 2011 SIMULIA Customer Conference, Barcelona, May 17-19 (2011)
12. Gondhalekar, A.C., Guenov, M.D., Wenzel, H., Balachandran, L.K., Nunez, M.: Neutral Description and Exchange of Design Computational Workflows. In: 18th International Conference on Engineering Design (ICED 2011), Technical University of Denmark, August 15-18 (2011)
13. http://en.wikipedia.org/wiki/Design_of_experiments
14. http://protegewiki.stanford.edu/wiki/Workflow_ontology
15. Sebastian, A., Noy, N.F., Tudorache, T., Musen, M.A.: 16th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2008), Catania, Italy. Springer (published in 2008)
16. Sebastian, A., Tudorache, T., Noy, N.F., Musen, M.A.: 4th International Workshop on Semantic Web Enabled Software Engineering (SWESE) at ISWC 2008, Karlsruhe, Germany (published in 2008)
17. Witherell, P., Krishnamurty, S., Grosse, I.R.: Ontologies for Supporting Engineering Design Optimization, J. Comput. Inf. Sci. Eng. 7 (2007)
18. http://www.scribd.com/doc/59675031/19/Review-of-ONTOP
19. http://en.wikipedia.org/wiki/Workflow
20. Russell, N., ter Hofstede, A.H.M., van der Aalst, W.M.P., Mulyar, N.: Workflow Control-Flow Patterns: A Revised View. BPM Center Report BPM-06-22, BPMcenter.org (2006)