

Proposal of a model-driven ontology for product development process interoperability and information sharing

Anderson Szejka, Osiris Junior Canciglieri, Eduardo Loures, Hervé Panetto, Alexis Aubry

► To cite this version:

Anderson Szejka, Osiris Junior Canciglieri, Eduardo Loures, Hervé Panetto, Alexis Aubry. Proposal of a model-driven ontology for product development process interoperability and information sharing. 13th IFIP International Conference on Product Lifecycle Management (PLM), Jul 2016, Columbia, SC, United States. pp.158-168, 10.1007/978-3-319-54660-5_15 . hal-01417620

HAL Id: hal-01417620

<https://hal.archives-ouvertes.fr/hal-01417620>

Submitted on 15 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Proposal of a model-driven ontology for product development process interoperability and information sharing

Anderson Luis Szejka^{1,2,3}, Osiris Canciglieri Júnior¹, Eduardo Rocha Loures¹,
Hervé Panetto^{2,3}, Alexis Aubry^{2,3}

¹Production and Systems Engineering Graduate Program, Pontifical Catholic University of Paraná, Curitiba, Brazil
{anderson.szejka, osiris.canciglieri, eduardo.loures}@pucpr.br

²Centre de Recherche en Automatique de Nancy, University of Lorraine, Vandoeuvre-lès-Nancy Cedex, France
{herve.panetto, alexis.aubry}@univ-lorraine.fr
³CNRS / CRAN UMR 7039, France

Abstract. The semantic interoperability of information has become increasingly important in Product Development Process (PDP) to support different phases during the product development. This article presents a proposal of an Interoperable Product Design and Manufacturing System (IPDMS) concept based on a set of engineering domain ontologies and semantic mapping approaches. The concept explores the potentials of semantic well-defined core-foundations in a Semantic Web ontology language. The formal core-foundations can be specialised to perform application view in Product or Manufacturing Model. The application view is used to support the information sharing between product design and manufacturing and verify the accordance with product requirements. A preliminary experimental has been realised, using a test case to share information from the design to manufacturing of an injection moulding plastic. As results, it was identified potential benefits and limitations. The main contributions are: semantic interoperability during information sharing in PDP and analysis of inconsistencies in PDP.

Keywords: Integrated Product Development Process, Transdisciplinary Engineering, Semantic Interoperability, Formal Model, Model-Driven Ontology.

1 Introduction

The complexity of Product Development Process (PDP) has increased over the years to meet the customer's needs. This process requires continuous information sharing with different groups within institutional boundaries as well as across multiple organisations. Thus, the current PDP's approaches of information exchange have been based on the product master models, as detailed in [1] and [2]. These approaches have a strict information structure, but mistakes and misinterpretation issues have been identified

across different PDP phases [3],[4]. This occurs because PDP has a set of transdisciplinary activities and multiple viewpoints that generate semantic obstacles without interoperability. The lack of interoperability is expensive for several globally distributed companies as discussed by [5].

Several resourceful efforts have been fostered to integrate solutions, following product master models through the definition of common information models. This is the way that international standards have been providing basis for product information exchange, e.g. STEP PLCS (ISO 10303). Related works such as [6], [7], [8], [9] indicate that there is a tendency to explore the use of Semantic Web ontology languages, like the Web Ontology Language (OWL), to model the knowledge of product models. However, a significant problem is to work with multiple domains since it is necessary to find effective and technical methods for semantically map information across related domains during the PDP. It is also evident that current works do not entirely address the rules to establish an analysis of product requirements, defined by the customer alongside the product design and manufacturing. Product requirements define the constraints and characteristics of a product and its information must be effectively shared across different phases of PDP without losing any meaning [10].

Based on this context, this paper proposes a model-driven ontology concept to formally structure a product design and manufacturing interoperability system and the establishment of the main rules for the information and knowledge sharing to achieve the semantic interoperability in the PDP. Thus, the main contributions of this research are highlighted as: (i) improvement of the product design and manufacturing information and knowledge sharing; (ii) interoperability between multiple domains; (iii) continuous analysis of product constraints across the PDP.

2 Technological Background

PDP is a set of multidisciplinary activities structured to transform market opportunities, customers' needs and technological constraints in products [2]. It is composed by six different phases: (i) planning; (ii) design; (iii) manufacturing; (iv) production; (v) maintenance and (vi) retirement. The process of product development is complex because information from multiple knowledge fields are simultaneously shared and exchanged through heterogeneous groups that may jointly function within institutional boundaries as well as across multiple organisations [11]. Although, PDP has a holistic view of product, providing information support to different phases of product development, misinterpretation and mistakes has been identified in design and manufacturing phases [12]. Furthermore, according to Rozenfeld et al. [2], the activities of design and manufacturing are most costly (85% of product final cost). Therefore, in a modern PDP, design and manufacturing information and knowledge handling must be efficiently shared across different phases of PDP.

This is a typical semantic interoperability problem for which the meaning associated to the captured information and knowledge must be effectively shared across systems without any loss of the meaning and intent of the information and knowledge during the exchange process [13]. The most common way to support information sharing has been to explore integrated solutions through defining common information

models [6], [14]. In this way, improved information sharing is being investigated through constructing formal ontologies (lightweight and heavyweight ontologies) [15] in different domains of application such as engineering, medicine, dentistry, business, etc.

Ontology is originally defined by Gruber in 1993 [16], as an “explicit specification of conceptualization”. Another important definition is provided by ISO 18629:2005 [17] that ontology is “a lexicon of specialised terminology along with some specification of the meaning of the terms in the lexicon”, where lexicon is the vocabulary of a language. Ontologies can be classified according to their degree of expressiveness. Simple ontologies, for example, involving only taxonomy of concepts and basic relations, are referred to as lightweight ontologies. Ontologies with concepts and relations as a lightweight ontology enriched with axioms in the form of constraints are classified as heavyweight ontology. According to [18], axioms are used to clarify the intended meaning of the terms gathered on the ontology. However, ontologies are usually limited to the purpose of its application and it has limited reusability outside the scope of its application. Thus, ontology integration is an important task to achieve different levels of integration. Ontology integration is the process of finding commonalities between two different ontologies O and O’ and deriving in a new ontology O” [18]. Based on this, three approaches for combining heterogeneous ontologies can be distinguished: (i) Ontology Inclusion; (ii) Ontology Mapping; and (iii) Ontology Merging.

Although, ontologies create semantic formalisms, a expressive problem is how to work with multiple ontologies of multiple domains to provide an effective mapping information across them [19]. Ontology mapping has been a key direction to tackle semantic heterogeneity issues across ontologies, intending to promote semantic interoperability. Several categories of ontology mapping methods have been suggested [20], [21], but there is a common consensus over the types of methods that can be applied. These types are: (i) Merging; (ii) Transformation; (iii) Alignment, and (iv) Articulation. Related works as [20], [21] present significant results on using matching techniques that use the semantics of logic-based systems, which employ upper ontologies. Therefore, this work uses the structure of product development process, ontologies and ontologies mapping ontologies to extract and enrich information to support the information sharing across PDP design and manufacturing phases in a transdisciplinary environment and in accordance to the customer’s needs.

3 IPDMS Concept

As discussed in the section 2, PDP is a set of multidisciplinary actions that gathers considerations of multiple domains and different developers team with multiple viewpoints across different phases of product development. These actions may jointly function within institutional boundaries as well as across multiple organisations [11]. Thus, although PDP has different approaches to systematize the information structure, it has been identified problems of information misinterpretation and mistakes caused by non-semantic interoperability [12].

Semantic interoperability is achieved when the meaning associated to the information and knowledge captured in computational form can be effectively exchanged across systems without losing any meaning and intent of the information and knowledge during the exchange process [13]. Additionally, interoperability is defined by IEEE [22] as the capacity of two or more systems to exchange information and to use the information that has been shared. Therefore, it is necessary to develop model-driven to semantic interoperability that provides balance and integrates multiple domain work and developers team with distinct viewpoints to allow the information exchange in a computational form across different phases of PDP.

In this context, Fig. 1 depicts the Interoperable Product Design and Manufacturing System (IPDMS) concept to support the information exchange in a computational form across three different phases of PDP.

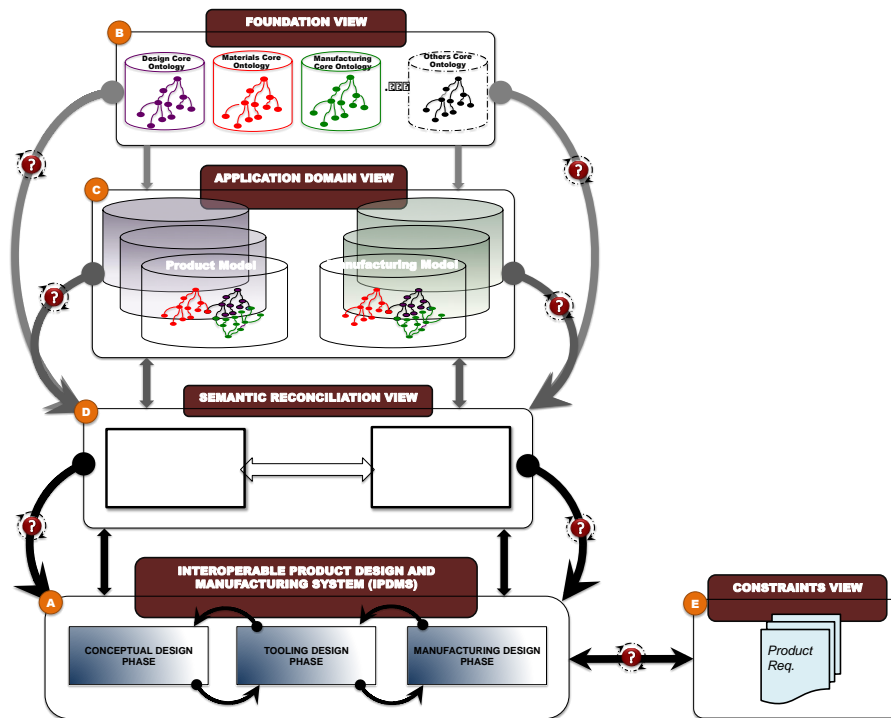


Fig. 1. Architecture of the Interoperable Product Design and Manufacturing System (IPDMS) Concept.

The concept uses a semantically well-defined core of concepts and constraints to instantiate the information in the application view and link them to with the semantic rules. Thus, the architecture of IPDMS is composed by four views: (i) Foundation View; (ii) Application View; (iii) Semantic Reconciliation View; and (iv) Constraints View. The three phases of PDP are exposed on Detail A of Fig. 1: (i) Conceptual Design Phase; (ii) Tooling Design Phase; and (iii) Manufacturing Design Phase. IPDMS works

based on formal models and semantic reconciliation through the semantic mapping concepts. The implication of each part of the IPDMS concept is next explained.

3.1 Foundation View and Application Domain View

The Foundation View (FV), detail B of Fig. 1, gathers and expressively represents concepts of different domains in a formal model at a relatively high level of abstraction. FV has Product Design Core, Tolerances Core, Materials Core, Manufacturing Core, etc. A Common Logic Based formalism is used to govern the way that the concepts can be formalised at the computational level.

In this way, we are using the ontologies approach to structure the FV, but we are not building ontologies. Thus, two distinct procedures to formalize the concepts based on ontological approaches have been used: (i) standards and information model in UML structure are formalized in a lightweight representation; and (ii) representations already published in ontology libraries, such as [DAML, OWL: Library, JOWL, DBpedia] are analysed and integrated to the FV. For both procedures, it is used Web Ontology Language (OWL) [23] with axioms rules in Semantic Web Rule Language (SWRL) [24].

The cores in FV can be individually specialized in the Application Domain View (ADV) to meet the needs of specific Product Design and Manufacture domains. Thus, the ADV was structured in Product Model and Manufacturing Model, as illustrated in Detail C of Fig. 1. A product model may be defined as an information model that stores information related to a specific product. The same occurs with the manufacturing model that has specific information related to the method to perform the product. According to [13], product and manufacturing models have a key role on PDP because they hold and share product information that are generated, used and maintained over the process of design, manufacturing, production, maintenance and disposal. Fig. 2 illustrates an example of three specialisations from the FV to the ADV to create the semantic links between concepts and a specific product.

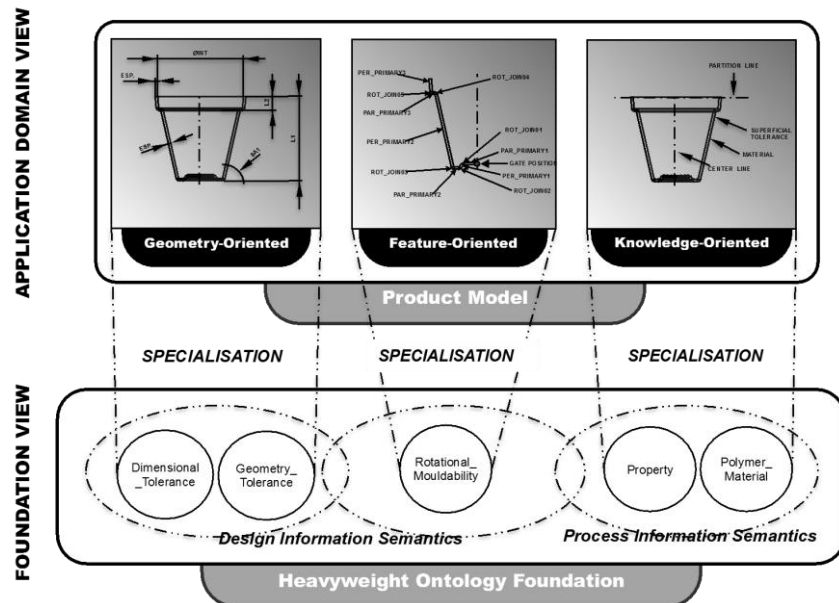


Fig. 2. Link between Foundation View and Application Domain View.

3.2 Semantic Reconciliation View

Several transdisciplinary information are shared across different phases of PDP and need to be integrated to other models in the ADP to verify possible inconsistencies. In the event that these models need to interoperate with the intention of sharing knowledge, domain semantics need to be reconciled. Semantic Reconciliation View (SRV), detail D of Fig. 1, covers relevant applied ontology-based techniques enabling the reconciliation of domain semantics.

These techniques work segments of known ontology matching methods such as: (i) the computation of contexts for domain ontologies [25]; (ii) ontology mapping [20], [21]; and (iii) semantic alignment [26]. Based on this, Fig. 3 illustrates the basic concepts involved in the mapping of domain models at the SRV. The process of semantic reconciliation can be performed between pairs of models at a time, as can be encountered in almost all-current ontology mapping frameworks and methodologies [20]. Context Adjustment involves a first stage of contexts adjustments (namespaces in this case) of two domain models which are to be reconciled. Following this stage is a simple ontology intersection process, where both models are intact loaded on to a single specialise ontology. The last procedure in the SRV is the semantic alignment, where semantic mapping concepts are loaded into the intersected models.

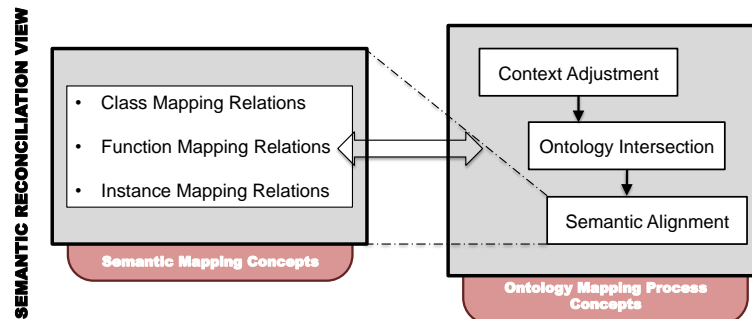


Fig. 3. Stages of Semantic Reconciliation View.

3.3 Constraints View

The Constraints View (CV), detail E of Fig. 1, gathers the restrictions for developing the product based on the Product Requirements defined by the customer's needs and technological specifications. Requirement is a statement from the stakeholder's needs to define a product; system or process characteristics and it must be unambiguous, clear, unique, consistent, stand-alone and verifiable [10]. Additionally, according to ISO/IEC 15288 [27], a requirement must be complete, coherent, unique, feasible, traceable and verifiable. Each requirement matches a single part of the future product, system or process and is grouped in an appropriate combination of textual statement views. Based on that, product requirements are used to create constraints rules to govern the way to develop the product and verify the consistency of the product design and manufacturing. Fig. 4 presents the main concepts involved to add these requirements in the IPDMS.

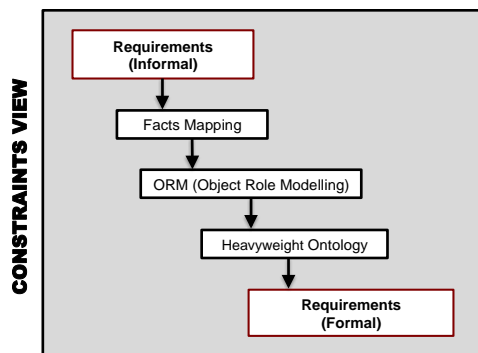


Fig. 4. Phases of Constraints View.

Firstly, the main facts are manually mapped in the requirement statement and modelling in a Conceptual Graphical approach as discussed by [4]. For this modelling is used the ORM (Object Role Modelling) that is a fully communication oriented information modelling method rooted in NIAM (Natural language Information

Analysis Method) [28]. According to [29], ORM has been used in ontology engineering to model domain ontologies. Thus, the product requirements in ORM are translated to heavyweight ontologies with axiomatic rules that are submitted to the semantic reconciliation view and matched in application domain view.

4 Discussion of preliminary results

An experimental system has been designed and performed to validate the Interoperable Product Design and Manufacturing System (IPDMS) model-driven approach and a rotational product was chosen to explore it. The product is a polystyrene cup with thermal conservation properties, as illustrated at the left side of Fig. 5. The figure also presents a simple demonstration of the IPDMS concept, supporting the information sharing from product concept design phase to product tooling design phase.

This test case explores the material and dimension information sharing to reduce the problems with product shrinkage, which is inherent in the injection moulding process. Shrinkage occurs because the density of polymer varies from the processing temperature to the ambient temperature. According to [30], during the injection moulding, the variation in shrinkage creates internal stress. If the internal stress are high enough to overcome the structural integrity of the product, the product are going to warp upon ejection from the mould or crack with external load during the extraction. Therefore, during the conception design and tooling design phases is necessary to share and transform information of the material properties and dimensions. This is a semantic obstacle because the meaning information found in the product concept must be effectively shared to product tooling design.

The IPDMS experimental demonstration is illustrates in Fig. 5. The product dimensions and geometry are extracted from the product modelled in CAD (Computer Aided Design) and structured in a XML file as illustrated on detail A of Fig.6. The XML file has the context alignment in the Semantic Reconciliation View that identifies the main concepts and search in the foundation view the core ontologies, detail B of fig. 5, related to the information extracted from the CAD. In this test case, the ontologies specialized in the product model are material core ontology and design core ontology and the ontologies specialized in the manufacturing model are material core ontology and manufacturing core ontology. The core ontologies identified in the foundation view are specialized in the Product Model and in the Manufacturing Model of the Application View, as shown on detail C of fig. 5 and instantiated with the specific information extracted from the CAD model.

Additionally Requirements, Specifications and Technological Constraints are added to the models through the constraint view, detail D of Fig. 5, such as: Material Name, Maximum Temperature, Liquid Type, Injection Process Type, etc. These new information are semantically reconciled and intersections occur between the constraint ontology and specialized ontologies. Thus, new semantic rules are established between the both ontologies through the SWRL (Semantic Web Rule Language) and shrinkage properties are embedded in the models. Moreover, concept mapping are established between the Product Model Ontology and Manufacturing Ontology Model as well as Semantic Rules in SWRL.

According to the set of formal information and rules defined in the specialized ontologies in product and manufacturing model, a new xml file is created and is responsible for the product tooling design, as show on detail E of Fig. 5. Part of the product tooling design can be automatically performed, respecting the rules of the shrinkage process, injection process and product dimensions and tolerance. Furthermore, constraints can be added to the model, as illustrated on Detail F of Fig. 5, such as: Machine type, Injection Pressure, Cooling Time, Melt Temperature, Mould Temperature, Holding Pressure. This new XML file is converted to a CAD file and the new 3D model with the tooling design is built.

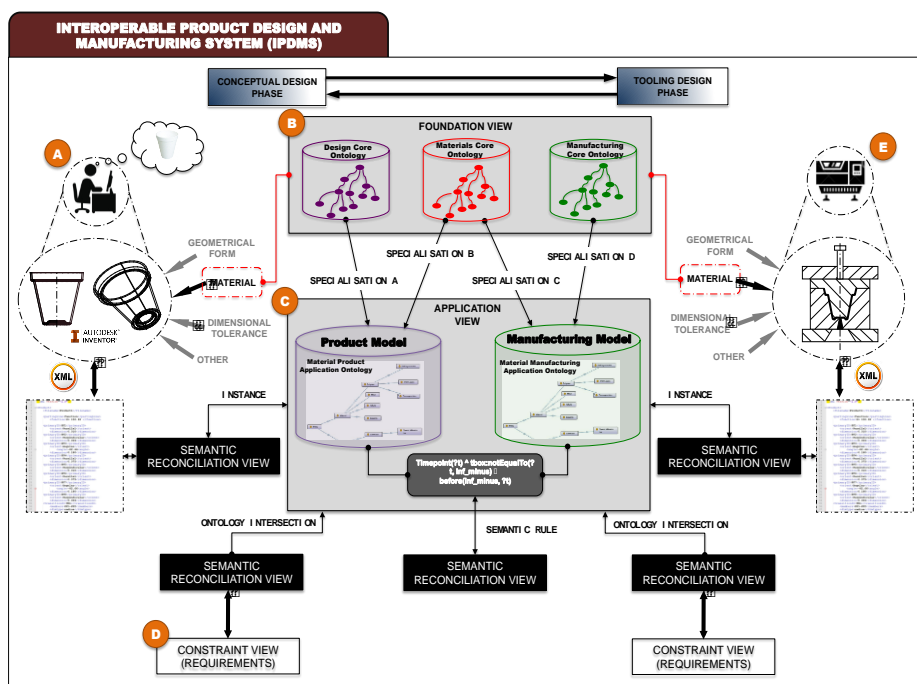


Fig. 5. IPDMS experimental demonstration system.

This preliminary test case shows the performance of the IPDMS concept. The preliminary results show the potential of the semantic interoperability to reduce the semantic obstacles. In the test case, shrinkage issues were solved by a structured information exchange from the product design to manufacturing.

5 Conclusion

This paper presented the development of an Interoperable Product Design and Manufacturing System (IPDMS) concept that are able to exchange information from multiple domains across different phases of PDP in a semantic interoperability manner.

The IPDMS supports the product developers, providing structured and formal information as well as transforming automatically information based on the knowledge added to the system.

The IPDMS concept uses a semantically well-defined core concepts and constraints, modelling in ontologies, to instantiate the information in the application view and constraint by semantic rules. The concept architecture has four different views: (i) Foundation View, (ii) Application View, (iii) Reconciliation View and Constraint View. The Foundation View has core concepts relating to PDP, such as: material core, design core, manufacturing core, dimensional tolerance, geometry tolerance, etc. The concept core is modelled in OWL language with semantic rules in SWRL. The core concepts are specialized in Application View in Product Model and Manufacturing Model and enriched with the detailed information of the product that is being developed. More information can be added to this model by the constraints view and all information are semantically reconciled and interrelated through the semantic reconciliation view.

This concept was evaluated in a preliminary test case. The product is a polystyrene cup with thermal conservation properties. In the evaluation process the concept potential to support the information sharing between the conceptual designs and tooling design to avoid the shrinkage issues was analysed. The 3D model information was extracted and submitted to the IPDMS concept. More information was added through the constraints view. In the IPDMS, instances, context alignment, ontologies intersection and mapping concepts were established to create a product and manufacturing ontology model specialized with semantic rules to design the product tooling. The results demonstrate the potential of this concept to reduce the semantic obstacles during the information sharing across the PDP. To extend this research work it is necessary to explore more variables simultaneously and more phases of the PDP in order to analyse the potential of this concept in working and sharing information from multiple domains across different phases of PDP.

Acknowledgments. The authors would like to thank the Coordination of Improvement of Higher Education Personal – CAPES, under Science without Borders Program (Special Visiting Researcher) - Project 178/2012 and Pontifical Catholic University of Parana (PUCPR) for the financial support.

References

1. Pahl, G.; Beitz, W.: Engineering design: a systematic approach. Springer, Darmstadt, 617p (1996).
2. Rozenfeld, H., Forcellini F.A., Amaral D.C., Toledo J.C., Silva, S.L., Alliprandini, D. H., Scalice, R.K.: Gestão de desenvolvimento de produtos: uma referência para a melhoria do processo. Saraiva: São Paulo: Saraiva, 560p (2006).
3. Szejka A.L., Aubry A., Panetto H., Canciglieri Jr. O., Loures E.R.: Towards a Conceptual Framework for Requirements Interoperability in Complex Systems Engineering. In: Meersman R., Panetto, H., Mishra A., Valencia-García, R., Soares, A.L, Ciuciu I., Ferri F., Weichhart G., Moser T., Bezzi, M., Chan, H. (eds.) OTM 2014. LNCS, vol.8842, pp. 229--240. Springer, Heidelberg (2014).

4. Szejka A.L., Canciglieri Jr. O., Loures E.R., Panetto H., Aubry A.: Requirements Interoperability Method to Support Integrated Product Development. In: 45th Computers and Industrial Engineering, vol. 147, pp. 1--8, Metz (2015).
5. Brunnermeier, S.B., Martin, S.A.: Interoperability costs in us automotive supply chain. *Supply Chain Management: An International Journal*. 7(2), 71--82 (2002).
6. Canciglieri Jr., O., Young, R.I.M.: Information mapping across injection moulding design and manufacture domains. *Int. J. Prod. Res.* 48(15), 4437--4462 (2010).
7. Chen Y.-J.: Knowledge integration and sharing for collaborative molding product design and process development. *Comput. Ind.* 61(7), 659--675 (2010).
8. Ouertani, M.Z., Baïna, S., Gzara, L. and Morel, G.: Traceability and management of dispersed product knowledge during design and manufacturing. *Comput. Aided Design*. 43(5), 546--562 (2011).
9. Liao, Y., Lezoche, M., Panetto, H., Boudjlida, N., Loures, E.R.: Semantic Annotation for Knowledge Explicitation in a Product Lifecycle Management Context: a Survey. *Comput. Ind.* 71, 24--34 (2015).
10. BKCASE Editorial Board: The Guide to the Systems Engineering Body of Knowledge (SEBoK), v. 1.6. R.D. Adcock (EIC). Hoboken, NJ: The Trustees of the Stevens Institute of Technology, www.sebokwiki.org (visited on 28/03/2016).
11. Hameed, A., Preece, A., Sleeman, D.: Ontology reconciliation. In: Staab, S., Studer, R. (eds.) *Handbook on Ontologies*. pp. 231--250. Springer, Heidelberg (2004).
12. Penciu, D., Durupt, A., Belkadi, F., Eynard, B., Rowson, H.: Towards a PLM Interoperability for a Collaborative Design Support System. *Procedia CIRP*. 25, 369--376 (2014).
13. Chungoora, N., Young, R.I.M., Gunendran, G., Palmer, C., Usman, Z., Anjum, N.A., Cutting-Decelle, A.-F., Harding, J. A., Case, K.: A model-driven ontology approach for manufacturing system interoperability and knowledge sharing. *Comput. Ind.* 64(4), 392--401 (2013).
14. Yang, A., et al.: A multi-agent system to facilitate component-based process modeling and design. *Comput. Chem. Eng.* 32(10), 2290--2305 (2008).
15. Noy, N.F. and Rubin, D.L., 2007. Translating the foundational model of anatomy into OWL. *Web Semantics*. 6(2), 133-136 (2008).
16. Gruber, T.R.: A translation approach to portable ontology specification. *Knowledge Acquisition*. 5(2), 199-220 (1993).
17. ISO (International Organization for Standardization). ISO 18629:2005: Industrial automation systems and integration - Process Specification Language (PSL). Geneva, CH. (2005).
18. Gómez-Pérez, A., Fernández-López, M., Corcho, O.: *Ontological engineering: with examples from the areas of knowledge management, e-commerce and the semantic web*. Springer-Verlag: London (2004).
19. Nagahanumaiyah, K.S., Ravi, B.: Computer aided rapid tooling process selection and manufacturability evaluation for injection mold development. *Comput. Ind.* 59(2/3), 262--276 (2008).
20. Kalfoglou, Y., Schorlemmer, M. Ontology mapping: the state of the art. *The knowledge engineering review*. 18(1), 1-31 (2003).
21. Ehrig, M., Sure, Y.: (2004). Ontology mapping – An integrated approach. In: *The Semantic Web: Research and Applications*. Bussler C.J., Davies, J., Fensel, D., Studer, R. (eds.) ESWS2004. LNCS, vol. 3053, pp. 76--91. Springer, Heidelberg (2004).
22. IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. Institute of Electrical and Electronics Engineers (1990)
23. Web Ontology Language (OWL). <http://www.w3.org/TR/owl-features/> (visited on 02/04/2016).
24. SWRL - A Semantic Web Rule Language Combining OWL and RuleML. <http://www.w3.org/Submission/SWRL/> (visited on 02/04/2016).

25. Stumme, G., Maedche, A.: Ontology merging for federated ontologies on the semantic web. In: Proceedings of the International Workshop for Foundations of Models for Information Integration. pp. 413-418, Viterbo (2001).
26. Euzenat, J. Shvaiko P.: Ontology Matching. Springer-Verlag: London (2007).
27. ISO (International Organization for Standardization) / IEC (International Electrotechnical Commission). ISO/IEC 15288:2008: Systems and software engineering – System life cycle processes. Geneva, CH. (2008).
28. Halpin, T. Object-Role Modeling (ORM/NIAM). In: International Handbooks on Information Systems Springer, pp 81-103 (2006).
29. Pan, W., Liu, D.: Mapping Object Role Modeling into Common Logic Interchange Format. In: 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), 2, pp 104–109. (2010).
30. Mohan, M., Ansari, M.N.M., Shanks, R.A.: Review on the Effects of Process Parameters on Strength, Shrinkage, and Warpage of Injection Molding Plastic Component. Polymer-Plastics Technology and Engineering, (just-accepted) (2006).