



**HAL**  
open science

# An Approach to Semantic Interoperability for Product Development Through Automatic Requirement Extraction and Semantic Reconciliation

Matheus Beltrame Canciglieri, Athon Moura Leite, Anderson Luis Szejka,  
Osiris Canciglieri Junior

## ► To cite this version:

Matheus Beltrame Canciglieri, Athon Moura Leite, Anderson Luis Szejka, Osiris Canciglieri Junior. An Approach to Semantic Interoperability for Product Development Through Automatic Requirement Extraction and Semantic Reconciliation. 16th IFIP International Conference on Product Lifecycle Management (PLM), Jul 2019, Moscow, Russia. pp.126-135, 10.1007/978-3-030-42250-9\_12. hal-04444749

**HAL Id: hal-04444749**

**<https://inria.hal.science/hal-04444749>**

Submitted on 7 Feb 2024

**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.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

# An Approach to Semantic Interoperability for Product Development through Automatic Requirement Extraction and Semantic Reconciliation

Matheus Beltrame Canciglieri<sup>1</sup>, Athon Francisco Curi Staben de Moura Leite<sup>1</sup>,  
Anderson Luis Szejka<sup>1[0000-0001-8977-1351]</sup> and Osiris Canciglieri Junior<sup>1[0000-0002-8503-9275]</sup>

<sup>1</sup> Industrial and Systems Engineering Graduate Program, Pontifical Catholic University of Parana, Curitiba, Brazil  
{athon.leite, matheus.canciglieri, anderson.szejka, osiris.canciglieri}@pucpr.br

**Abstract.** The strong competitiveness challenges manufacturing industry to rationalize different ways of bringing new products to the market in the shortest time with competitive prices while ensuring higher quality. Industries need to effectively share product requirements, which comes from various sources and are heterogeneous in nature, during the Product Development Process (PDP) to stay competitive. However, problems with misinterpretation of such requirements have been identified during its sharing due to semantic interoperability obstacles related to the process of automatically gathering requirements and their translation and reuse. This research proposes an approach to automatically gather product requirements, extracting its knowledge and translate it for further use and reuse along PDP. The research structure consisted of firstly studying the current issues of the topic, secondly by exploring an approach, to be validated in an experimental case. Current issues point out to gaps related to the process of semantic reconciliation and knowledge extraction perspectives, giving it a multi-dimensional panorama where semantic issues are intertwined among different perspectives. The later solution presents an approach which considers raw information being processed into product features, refining knowledge during PDP and making it reusable in different cycles. The approach brings a new view on practical methods for automatically collect product requirements, extract its knowledge and translate it into product features, based on knowledge extraction methods and by using semantic reconciliation as means for translating product's requirements. Further research will focus on expanding the approach and including more features to increasingly complex cases, to explore the full potential of the approach.

**Keywords:** Product Development Process, Multiple Domains, Knowledge Extraction, Product Design and Manufacturing, Semantic Reconciliation, Semantic Interoperability.

## **1 Introduction**

The current panorama of product development is defined by sets of complex requirements, higher standards of quality, changes in value perception and products that better fulfill customer's needs. In this context, product development relates to current trends on smart factories, integrated manufacturing systems and Industry 4.0, which in the product lifecycle scenario accompanies the use of technologies and methods to improve communication and support the decision-making process.

The market needs related to quick development cycles demands parallelism of activities and knowledge from different areas, using information technologies as a basis for communication across development. An Integrated Product Development Process (IPDP), in this context, considers multiple domains and various sources of knowledge as a means of gathering requirements to new or improved products. These requirements must have a good definition and must be shared without losing their meaning, in order to avoid incoherency, misinterpretation and other negative impacts during development, such as delays and increased costs [1].

Current praxis is still not able to cope with all semantic issues within an integrated product development process, as requirements must be clear, consistent, measurable, stand-alone, testable, unambiguous, unique and verifiable [2]. These issues are mostly caused by different taxonomies, divergent views from professionals of different areas and limitations in the translation of knowledge in product development phases. Consequently, the misinterpretation of market requirements and development mistakes are a result of heterogeneous information from different domains of knowledge [2].

To address such problems, the concept of Semantic Interoperability aims to ensure the effective sharing of information in a collaborative environment with multiple domains of knowledge. Semantic Interoperability is currently being applied in the context of product design and manufacturing, to reduce mistakes and heterogeneity in information [3]. There are still, though, problems with its implementation, regarding the methods to automatic requirement extraction and translation from different sources and organize it in a standardized way [3][4]. In this context, this research aims to develop an approach to gather, organize and translate product requirements in a standardized manner, reducing heterogeneity of interpretation during product development in multiple domain environments.

## **2 Conceptual Background**

### **2.1 Product Development Process**

The Product Development Process approaches different domains of knowledge, involving specialists from different areas with multiple viewpoints. The transdisciplinary nature of PDP activities represents a holistic approach, which meets the requirements and constraints (laws, specification, etc.), respecting technical limitations and enterprise's goals [1]. Authors [5][6][7] cite development environments as integrated, communicative, collaborative and interoperable as, according to [7], the

better the product information is defined, the lesser the misinterpretation, mistakes, and semantic barriers occurrence.

Most of the more systematic approaches for PDP consider three main phases: Pre-Development, Development and Post Development. Current models normally emphasize the systematization process of the Product Development Process. Originally, PDP was defined as a linear flow, with later activities starting after the conclusion of the previous ones [1]. Later approaches review this structure by inserting parallelism along the development cycle [5][6][7]. This structure allows a better design of information and the understanding of how to formalize it, in order to reduce the misinterpretation. There are not, though, enough methods that ensure the seamless information interoperation along development cycles.

The representation of knowledge in PDP has a direct impact on the capability of information semantically interoperate [2]. This is due to the degree of standardization in structured information. The Product Development Process has two most significant types of model to represent its knowledge: the product model [2][8]; and the manufacturing model [2][9].

Current research on this field points out the need for ensuring interoperability between product and manufacturing models since there occur misinterpretation issues when the product development approaches multiple domains of knowledge [4]. The researches found on [2][9][10][11] presented a tendency to use ontological approaches to gather and formalize knowledge in product and/or manufacturing models.

## 2.2 Ontology-Driven Semantic Interoperability

The use of ontologies has increased for the development of shared representations. Recent research, as shown in [2][9][11] proved that the ability for sharing semantics across these representations is supported by ontological formalisms. Currently, ontology is recognized as a key technology to deal with semantic issues related to interoperation [12]. Its computational structure has been developed to provide a machine-processable semantics of varied knowledge sources [13].

In spite of its contributions to semantic wholeness, it is worth mention that even when using ontology-based methods to assure sharing semantics, semantic heterogeneity is still unavoidable and, for this reason, methods for ontology mapping are in development to reconcile semantics between ontologies that need to interoperate [14].

In the field of application, ontologies can be categorized into three levels of abstraction, depending on the deepness in which the ontology aims to represent a knowledge [3]:

- **Foundation Ontology** - suited for general concepts and relationship, can be used in different domains;
- **Reference Ontology** - Domain-specific ontology, reusable for different activities in the same domains;
- **Application Level Ontology** - represents specific knowledge, dedicated to a task.

### 2.3 Semantic Reconciliation Methods

Ontology integration is the process of finding common aspects between different ontologies, creating a new ontology with its common aspects. [15] discussed the distinction among integration approaches, as follows:

- **Ontology Inclusion** - simple inclusion of an ontology in the other's structure;
- **Ontology Mapping** - processing of similarities and relations from different sources through equivalent relations and combining it at one new ontology;
- **Ontology Merging** - processing a vast number of data sources through a mediator and querying process to achieve a newly combined ontology

Even though ontology mapping/matching has been key to solve semantic heterogeneity problems, there is a present of methods that rely on lexical similarity matching, which is not optimal from a semantic interoperability viewpoint. This is because in multiple domain environments similar terms are used across different groups to refer to diverse concepts. Ergo, it is only through the semantics associated with these terms that existing differences can be identified, highlighting the need to capture semantics in the first place [3]

On the other hand, ontological formalisms like the Web Ontology Language (OWL) support built-ins for ontology mapping, however, these built-ins have limitations when mapping the semantic content of manufacturing ontologies and their associated knowledge bases [3].

Moreover, there are requirements to aid the construction of mapping/matching techniques which can be formally interpreted and are focused at identifying potential solutions for semantic mismatches. Therefore, enabling the reconciliation process at several levels, including the instance level, of ontology-based models. These requirements aid the reconciliation process to have better accuracy, automation and reduces the time that takes to resolve cross-model correspondences [16].

## 3 Research Development for an Approach to Semantic Interoperability for Product Development

### 3.1 Problem Statement

Modern Product Development Process (PDP) has required simultaneous collaborations of multiple groups, producing and exchanging information from multi-perspectives within and across institutional boundaries. However, semantic obstacles have been identified during this process, affecting the process of product development. In this context, [ ] identified in their research three main issues: (i) Cross Domain issue; (ii) Cross PDP Phases Issues; and (iii) Cross Requirement Representation and Translation Issues.

In *Cross Domain issue* to standardize the structure of information in different domains, representation models were developed, such as UML (Unified Modelling Language), Model-Driven Engineering (MDE), Domain Specific Language (DSL), and

others [17][18][19]. Despite presenting an intent to formalize and represent knowledge from various domains in a standardized manner, those models do not present in an accurate way the dynamic nature of market requirements and knowledge from different product development and manufacturing phases [19].

Recently, few models consider the consistency of requirements and performance in dynamic environments. In [20], the authors explain a design framework for cyber-physical systems, based on the design rationale, connecting different design parameters and requirements from varied sources. In [3], the authors explore the Manufacturing domain and requirement's gathering from different domains. Both researches have combined different models in order to achieve verifiable and validated information within a dynamic requirements context, but still, there is a significant presence of specialist's decisions during the process of translating requirements. This might result in semantic issues, as there are subjective factors and different methods related to each domain.

To *Cross PDP phases issue*, the communication in PDP relies heavily on the semantic interpretation of each agent [21]. Across different phases, the different set of information may cause misinterpretation due to variable meanings for a term. This occurs due to the different knowledge background of developers and their experience in different stages of product development [7][19][21]. Corroborating that, [19] states that the knowledge required for a stage of the development process may have different implications in later activities, being changeable due to the dynamic nature of product development requirements.

Currently, as demonstrated in [3], literature proposes a formalization through semantic annotations for semantically interoperable applications, regarding different views in PLM. There are no annotations, however, that accurately represent dynamic requirements in mentioned research. In [22], the author proposes a solution to cross-PDP issues based on a model-driven ontology, which was limited to two domains and presented no continuity for further expansion and integration for more domains.

In *Cross Requirement Representation and Translation Issues*, requirements represent the main inputs in an ontology-driven semantically interoperable system. This knowledge must be 'semantically whole', to avoid further interoperation issues, by using formalized structures, well-defined statements and axioms [4]. In most cases, however, the abstraction of poorly defined statements ends up generating divergent interpretation, which might result in issues related to the uniqueness, the comprehensiveness and, most important, the traceability of information [2].

Correct, consistent and traceable information can prevent further inconsistencies in product development and manufacture [23]. In [23], a framework for semantic interoperability corroborates this view. Despite that, the provided framework do not ensure the requirements' traceability and optimization of knowledge gathering and its structuring process. In [24], the author presented an interoperable model considering multiple domains to ensure traceability through validation and verification methods. This model, however, was limited only to the early phases of a system. Current research, as shown in [2], points to standardized forms of extracting knowledge and information and ensure its traceability through verification and validation, but still with no automatic requirement extraction.

### 3.2 Main research questions

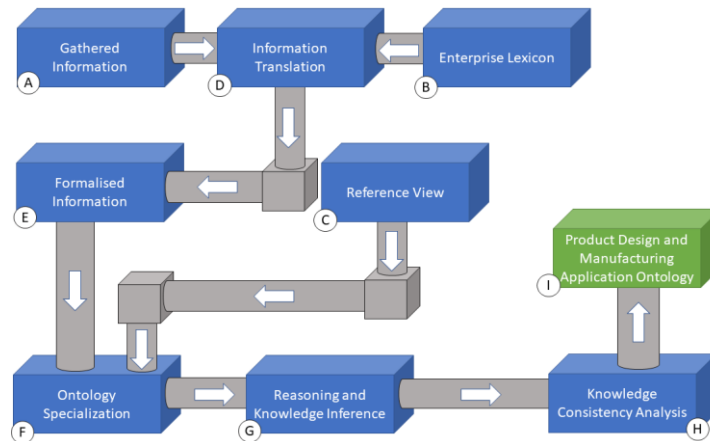
Globalization, collaboration, and cooperation have contributed to the emergence of a knowledge sharing culture in open and large environments. However, communication between project teams is often hindered by the lack of clarity in the terms used. This represents that the context in which the information is exchanged affects the overall meaning and interpretation of the shared implicit and explicit knowledge. Therefore, two problems may appear: i) *the same term is being applied to different concepts (semantic problem)* and ii) *different terms are being applied to the same concept (syntax problem)* [25]. These problems are known as semantic heterogeneity

One approach to solve these problems is the creation of an ontology in which a specific domain's concepts and terms are defined and can be understood through the different teams across that domain. Nevertheless, this presents a different issue: *How can the knowledge be shared effectively through multiple domain systems without the problems caused by semantic heterogeneity?*

### 3.3 The approach to Semantic Interoperability for Product Development through Automatic Requirement Extraction and Semantic Reconciliation

In order to answer this problem, an approach is proposed in Fig. 1 that aims to apply semantic interoperability concepts and tools to develop an interoperable environment that can efficiently represent and translate knowledge between different stages of the product development process and analyze the consistency of the created knowledge, reducing the misunderstandings and mistakes caused by semantic heterogeneity.

The proposed approach showed in Fig. 1 is divided into 9 stages. The first one (Detail A of Fig. 1) consists in gathering knowledge from the consumer, using tools as the Quality Function Deployment (QFD) or through an enterprise's Customer Relationship Management software, alongside Computer Aided Design/ Computer Aided Manufacturing (CAD/CAM) and production planning and control systems.



**Fig. 1.** Proposed Semantic Interoperable System for Product Development Process and Manufacturing.



This information is gathered from different sources and, therefore, it is heterogeneous, as the information is presented differently by the different tools. Therefore, before this information can be used in the decision process, it must be translated and for this process to happen, a lexicon of the enterprise's terms is needed (Detail B of Fig. 1). This lexicon contains all the enterprise concepts, their associated context, and explicit meaning and is formalized in an OWL format.

Alongside the enterprise's lexicon, the proposed approach has a reference view (Detail C of Fig. 1), which gathers information from within the enterprise alongside the real world and represents it in a high-level abstraction of different domains through ontological models. These models are considered the knowledge core of the proposed approach.

The gathered information is translated into useful information (Detail D of Fig. 1) to the product development decision making process through an ontology alignment where the gathered information is compared to the lexicon in order to create a formalized information structure that can be used by the enterprise without mistakes. This information is structured in an OWL format to allow the proposed approach to further refine it and to share it across all stages of the product development process (Detail E of Fig. 1).

With the necessary knowledge gathered and formalized in an ontology, the proposed approach uses an ontology alignment to create the application ontology for this stage (Detail F of Fig. 1). The ontology alignment first "cuts" the useful information from the different ontologies, the gathered information ontology and the reference view ontologies, and then "merges" them into an application ontology.

After the alignment, the knowledge translation is applied in order to transform the design information into manufacturing information (Detail G of Fig. 1). This translation, as in the previous step occurs through a similarity analysis made by the inference engine in three levels; i) critical concept similarities, that is, the critical requirements gathered from the external sources are compared to the critical knowledge presented in the ontology; ii) relationship analysis, the relationship between concepts in the external sources are compared to the relationship presented in the reference view ontologies and any similar relations are selected and iii) concept relationship, the concepts of the external sources are analyzed regarding its similarity to the reference view ontologies.

A consistency verification in the design and manufacturing application ontology (Detail H of Fig. 1) in order to verify the information consistency, if there is an inconsistency in the application ontology, it must be adjusted and verified which information from the external sources or the reference view was responsible for this error and corrected. After the adjustment process, the consistency analysis must be made again in order to avoid ambiguous information. If there was no inconsistency the ontology is shared with the next stage of the IPDP.

The resulting ontology from this stage shares the product design and manufacturing planning to the fabrication process, where the product will be mass produced. This planning is shared through the hierarchical structure of the design and manufacturing planning application ontology (Detail I of Fig. 1).

### 3.4 Experimental Case

In order to validate the proposed approach, a conceptual experimental case is proposed. For this experimental case, the product chosen Uninterrupted Power Supply (UPS) with the potency of 20 kVA as it was being reformulated, through the customer requirements gathering, by the enterprise in which this research was conducted. This product was developed without using the proposed approach. It took 20 months to be developed, using 2808 hours of work and cost to the enterprise US\$ 32.840,00.

From these 20 months, the project was paused through 6 months due to reviews the project had to undergo in order to comply with customer needs. This pause cost to the enterprise US\$ 13.432,00.

For the experimental case, initially, the proposed approach focused on gathering the requirements from clients through an automated QFD tool. This tool gathers the information from users and automatically determines most relevant requirements, crossing with the organization's technical requirements. The product of that is an XML file, which will undergo through a semantic reconciliation process with an ontology that contains the Enterprise Lexicon, to correctly designate the organization's terminologies to customer's requirements from the QFD tool. This process consisted of adapting the generated XML from the QFD into a formal taxonomy (new ontology) by a computational algorithm, which identifies similar terms in the Enterprise Lexicon ontology and the customer's requirements.

The formalized knowledge gathered is presented as an ontology and it is further refined by comparing them to the design reference ontology and manufacturing reference ontology. This comparison is made through an ontology alignment process and will work as a guide to the product design and the manufacturing planning process. In the experiment, the technical requirements and manufacturing restrictions are translated into design requirements which reduced the problems in the production line and more aligned with the consumer needs.

After the Alignment the design information is translated into manufacturing planning information to assist the production planning and control. In the experiment, the design information gathered from the CAD systems (Altium for the circuit boards and SolidWorks for the mechanical parts) is translated into manufacturing information, such as tool path, process order, etc. Alongside, a consistency analysis is made through the inference engine in order to establish if the manufacturing information is coherent with the enterprise's production facilities.

With the consistency analysis performed and with no inconsistency found the resulting ontology is used to control the production and usage of the product, with the aim of creating a prioritization for further improvements on the product.

The reformulation of the UPS using the proposed approach reduced the development time from 20 months to 12 months and the cost of the project was 30% as there was no need to pause the development to adjust the product to the customer needs.

## 4 Conclusion

As this application was of a conceptual nature, this research could not completely explore the full potential of the proposed approach. However, even with this limitation, the approach presented significant benefits in the integration of the product development process and manufacturing, as it reduced the time and cost of the development of a product and improved the overall quality as there was no need to redesign the product afterward. Further research will focus on expanding the approach and including more features to increasingly complex cases, to explore the full potential of the approach.

## Acknowledgments

The authors would like to thank Pontifical Catholic University of Parana (PUCPR) and NHS Sistemas Eléctricos for the financial support to the development of this research.

## References

1. Rozenfeld, H., Forcellini, F. A., Amaral, D. C., De Toledo, J. C., Silva, S. L., Alliprandini, D. H., & Scalice, R. K. *Product Development Management: a reference for process improvement*. Sao Paulo: Saraiva (2006).
2. Szejka, A.L., Júnior, O.C., Loures, E.R., Panetto, H. and Aubry, A. Proposal of a model-driven ontology for product development process interoperability and information sharing. *IFIP International Conference on Product Lifecycle Management*, 158-168 (2016).
3. Chungoora, N., Young, R. I., Gunendran, G., Palmer, C., Usman, Z., Anjum, N. A., Cutting-Decelle, A. F., Harding, J. A. and Case, K. 'A model-driven ontology approach for manufacturing system interoperability and knowledge sharing', *Computers in Industry*, 64(4), 392–401 (2013).
4. Szejka, A.L., Aubry, A., Panetto, H., Canciglieri, O.J. and Loures, E.R. Towards a conceptual framework for requirements interoperability in complex systems engineering. In *OTM Confederated International Conferences*, 229-240 (2014),.
5. Andreasen, M.M. and Hein, L. *Integrated product development*. Bedford, UK: IFS Publications Ltd (1987).
6. Sosa, M.E., Eppinger, S.D. and Rowles, C.M. The misalignment of product architecture and organizational structure in complex product development. *Management Science*, 50(12), 1674-1689 (2004).
7. Pereira, J.A. and Canciglieri, O. J., Product Development Model Oriented for the R&D Projects of the Brazilian Electricity Sector. *Applied Mechanics and Materials*. 518, 366-373 (2014).
8. Canciglieri, O.J. and Young, R.I.M. Information mapping across injection moulding design and manufacture domains. *International Journal of Production Research*, 48(15), 4437-4462 (2010).
9. Chungoora, N., Canciglieri, O. and Young, R. I. M. Towards expressive ontology-based approaches to manufacturing knowledge representation and sharing. *International Journal of Computer Integrated Manufacturing*, 23(12), 1059–1070 (2010).

10. Panetto, H., Dassisti, M. and Tursi, A. ONTO-PDM: Product-driven Ontology for Product Data Management interoperability within manufacturing process environment. *Advanced Engineering Informatics*, 26(2), 334-348 (2012).
11. Imran, M. and Young, B. The application of common logic based formal ontologies to assembly knowledge sharing. *Journal of Intelligent Manufacturing*, 26(1), 139-158 (2013).
12. Palmer, C., Urwin, E.N., Pinazo-Sánchez, J.M., Cid, F.S., Rodríguez, E.P., Pajkovska-Goceva, S. and Young, R.I.M. Reference ontologies to support the development of global production network systems. *Computers in Industry*, 77, 48-60 (2016).
13. Fensel, D. Triple-space computing: Semantic Web Services based on persistent publication of information. In *Intelligence in Communication Systems*, 43-53 (2004).
14. Fahad, M., Moalla, N., Bouras, A., Qadir, M.A. and Farukh, M., (2010). Disjoint-knowledge analysis and preservation in ontology merging process. In *Software Engineering Advances (ICSEA)*. 422-428.
15. Malucelli A. Ontology-based Services for Agents Interoperability. 266p. *Thesis (Ph.D.)* - University of Porto, Porto, (2006).
16. Kumar S. and Harding J. Ontology mapping using description logic and bridging axioms. *Computers in Industry* 64(1): 19-28 (2013).
17. Canciglieri, O.J. and Young, R.I.M. Information sharing in multiviewpoint injection moulding design and manufacturing. *International Journal of Production Research*, 41(7), 1565-1586 (2003).
18. Haveman, S.P. and Bonnema, G.M., Requirements for high level models supporting design space exploration in model-based systems engineering. *Procedia computer science*, 16, 293-302 (2013).
19. Nattermann, R. and Anderl, R. Approach for a Data-Management-System and a Proceeding-Model for the Development of Adaptronic Systems. In *ASME 2010 International Mechanical Engineering Congress and Exposition*, 379-387 (2010).
20. Moneva, H., Hamberg, R. and Punter, T. A Design Framework for Model-based Development of Complex Systems. In *32nd IEEE Real-Time Systems Symposium 2nd Analytical Virtual Integration of Cyber-Physical Systems Workshop*, Vienna (2011).
21. Stechert, C. and Franke, H.J. Managing requirements as the core of multi-disciplinary product development. *CIRP Journal of Manufacturing Science and Technology*, 1(3), 153-158 (2009).
22. Liao, Y., Lezoche, M., Panetto, H., Boudjlida, N. and Loures, E. R. Semantic annotation for knowledge explicitation in a product lifecycle management context: A survey. *Computers in Industry*, 71, 24-34 (2015).
23. Pohl, K. The three dimensions of requirements engineering: a framework and its applications. *Information systems*, 19(3), 243-258 (1994).
24. Cleland-Huang, J., Chang, C.K., Sethi, G., Javvaji, K., Hu, H. and Xia, J. Automating speculative queries through event-based requirements traceability. In *Requirements Engineering, 2002*, 289-296 (2002).
25. Lin, H-K, Harding, J., Shahbaz, M. Manufacturing system engineering ontology for semantic interoperability across extended project teams. *International Journal of Production Research*, 42(24), 5099-5118 (2004).