Formalization of semantic annotation for systems interoperability in a PLM environment

Yongxin Liao, Mario Lezoche, Eduardo Rocha Loures, Hervé Panetto, Nacer Boudjlida

Formalization of Semantic Annotation for Systems Interoperability in a PLM environment

Yongxin Liao[1, 2], Mario Lezoche[1, 2], Eduardo Loures[1, 5], Hervé Panetto[1, 2] and Nacer Boudjlida[3, 4]

[1]Université de Lorraine, CRAN, UMR 7039, Boulevard des Aiguillettes B.P.70239, 54506 Vandoeuvre-lès-Nancy, France
[2]CNRS, CRAN, UMR 7039, France
[3]Université de Lorraine, LORIA, UMR 7503, Boulevard des Aiguillettes B.P. 70239, 54506 Vandoeuvre-lès-Nancy, France
[4]CNRS, INRIA, LORIA, UMR 7503, France
[5]Industrial and Systems Engineering, Pontifical Catholic University of Parana. Imaculada Conceicao 1155, Curitiba, Brazil
{Yongxin.Liao, Mario.Lezoche, Herve.Panetto}@univ-lorraine.fr, eduardo.loures@pucpr.br, Nacer.Boudjlida@loria.fr

Abstract. Nowadays, the need for systems collaboration across enterprises and through different domains has become more and more ubiquitous. Due to the lack of standardized models or architecture, as well as semantic mismatching and inconsistencies, research works on information and model exchange, transformation, discovery and reuse are carried out in recent years. One of the main challenges in these researches is to overcome the semantic gap between enterprise applications along any product lifecycle, involving many distributed and heterogeneous enterprise applications. We propose, in this paper, an approach for semantically annotating different knowledge views (business process models, business rules, conceptual models, and etc.) in the Product Lifecycle Management (PLM) environment. These formal semantic annotations will make explicit the tacit knowledge generally engraved in application models and act as bridges to support all actors in along the product lifecycle. A case study based on a specific manufacturing process will be presented for demonstrating how our semantic annotations can be applied in a Business to Manufacturing (B2M) interoperability context.

Keywords: Ontology, Semantic Annotation, Systems Interoperability, Business Process, PLM, BPMN

1 Introduction

The opening of enterprise information systems towards integrated access has been the main motivation for the interest around systems interoperability. In order to achieve the main objective of the enterprise, the business domain and the manufacturing domain need to exchange information and to synchronise their knowledge concerning the related product. Over the last ten years complex engineered products have discovered the benefits of PLM solutions and are adopting efficient PLM software in increasing numbers [1]. Contemporary PLM systems typically use workflow technology to provide support for process management. From many common business processes in the manufacturing industry in areas such as accounting, engineering design, product release, process planning, and production control, emerges the problem on versioning policies [2]. PLM represents an all-encompassing vision for managing all data relating to the design, production, support and ultimate disposal of manufactured goods. PLM can be thought of as both a repository for all information that affects a product, and a communication medium between product stakeholders: principally marketing, engineering, manufacturing and field service. The PLM system is the first place where all product information from marketing and design comes together, and where it leaves in a form suitable for production and support. In the same philosophy the product centric vision developed by [3] theorizes an omnipresence of the product related knowledge in the product itself.

Panetto et al. [4] postulate that an ontological model of a product may be considered as a facilitator for interoperating all applications software that share information during the physical product lifecycle. Their approach concerns the formalization of all technical data and concepts contributing to the definition of a Product Ontology, named ONTO-PDM, embedded into the product itself and making it interoperable with applications, thus minimizing loss of semantics. The ONTO-PDM acts as a common core model for enterprise applications interoperability in manufacturing process environment. Chen et al. [5] proposes an ontology-based framework for sharing and integrating product lifecycle knowledge. The authors present a mechanism that integrates ontology-based product lifecycle knowledge distributed among different cooperative enterprises allowing all knowledge actors to share product lifecycle knowledge. Wang et al. [2] stress out those current methods of process modeling which lack adequate specification of terminology used in supply chain process models. In a complete supply chain process, this leads to inconsistency and semantic conflicts between the interchanging of various process models. They propose combining BPMN ontology with SCOR ontology, deriving the so-called scorBPMN ontology, which specifies the semantics in supply chain processes.

The enterprise models are mainly related to some views and artefacts such as processes, behaviors, activities, data, resources, material and information flows, infrastructure and architecture. These models must contain the necessary and sufficient semantics in order to be intelligible and then enabling the global Enterprise Interoperability [6]. Semantic Annotations are generally used in heterogeneous domains and help to bridge the different knowledge representations [7]. There are several methods for modelling semantic annotations that vary in their referenced ontology (languages, tools and design), models and corresponding applications, as presented in our previous paper [8]. For example, the semantic business process model defines in details the business process flows, modelling the information, resource policy, business rules and other element encompassed in a workflow [9]. In [10], Author presents a complete overview on business process semantic annotations and divides the existing proposals into two groups: (i) adding semantics to specify the dynamic behaviour exhibited by a business process; (ii) adding semantics to specify the meaning of the entities of a process. In our approach, a semantic annotation is formally represented by the so-called semantic annotation structure model (SASM). The additional knowledge provided by the SASM makes bridges between different models to support the different actors in PLM environment.

This paper is organized as follows. Section 2 presents an updated version of the formal definition of semantic annotation, SASM and semantic annotation framework. In section 3, a real case study is presented in order to demonstrate the applicability of our framework. Section 4 concludes the paper and presents the future research.

2 Formalization of semantic annotation

2.1 Formal Definition of Semantic Annotation

A semantic annotation can be considered as a formal model which describes the relationship between the original information source and an ontology [9]. We proposed a formal definition: semantic annotation SA is a tuple $(\mathcal{M}, R)$ that is composed by two parts: the structural part, a set of mappings $\mathcal{M}$, between a set of elements of knowledge $\mathcal{E}$ and a powerset of ontology $\mathcal{P}(O_l)$; and the representational part, a set of meta-model references R [8].

$$SA = \{\mathcal{M}(\mathcal{E}, \mathcal{P}(O_l)), R\}$$

Where:

$\mathcal{E} = \{e_1, e_2, \dots, e_n\}$, $\mathcal{E}$ is composed by a set of element e_i from different knowledge views, which represents the knowledge that needs to be annotated.

$O = \{o_1, o_2, \dots, o_r\}$, O is composed by a set of ontology o_k, which represents the specific knowledge in a formal way. An *Ontology* $o_k \epsilon \mathcal{O}$ is a 4-tuple (C_k, is_a, RN_k, σ_k), where C_k is a set of *concepts*, *is_a* is a partial order relation on C_k, RN_k is a set of relation names, and σ_k: $RN_k \rightarrow (C^+)$ is a function which defines each relation name with its arity [11].

$\mathcal{P}(O_l) = \{p_1, p_2, \dots, p_m\}$, $\mathcal{P}(O_l)$ is composed by a set of powerset of ontologies p_j, which brings meaning to annotated element of knowledge.

$p_j = \{\prod_{k=0}^{z} o_k \mid o_k \in \prod_{l=1}^{s} O_l\}$, p_j is composed by different concepts that referenced from one or more ontologies.

$\mathcal{M} = \{m_x\langle e_i, p_j\rangle \mid e_i \epsilon \mathcal{E} \times p_j \epsilon \mathcal{P}(O_l)\}$, $\mathcal{M}$ is composed by a set of mapping m_x, which describes the semantic relationship between e_i and p_j.

- $m_\sim (e_i, p_j)$: Equivalence relationship, which states that e_i is semantically equivalent to p_j;
- $m_\supset (e_i, p_j)$: Subsumes relationship, which states that e_i subsumes the semantics of p_j;
- $m_\subset (e_i, p_j)$: Subsumed by relationship, which states that e_i is subsumed by the semantics of p_j;

- $m_{\cap}(e_i, p_j)$: Intersection relationship, which states that e_i intersects with the semantics of p_j

R = $\{r_1, r_2, \ldots, r_t\}$, R is composed by a set of meta-model representation r_y, which represents the meta-model specification for the element of knowledge.

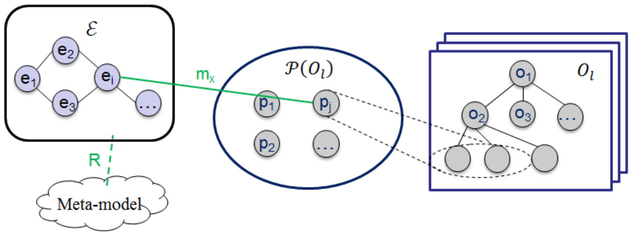

Fig. 1. Formal Definition of Semantic Annotation.

The constituent parts of a formal semantic annotation are illustrated in Figure 1. On the left side, it indicates that an element of knowledge e_i in $\mathcal{E}$ is annotated with the powerset of ontology p_j in $\mathcal{P}(O_l)$ with their semantic relationship m_x. On the right side, it shows that the powerset p_j is composed by one or more concepts from one or more ontology O_l. The relationship R describes the referenced meta-model that expresses the annotated knowledge in the language represented.

2.2 Semantic Annotation Structure Model (SASM)

One of the well-known studies in semantic annotation area is proposed by SAWSDL Working Group[1]. They developed SAWSDL (Semantic Annotation for Web Services Definition Language) [12], which provides two kinds of extension attributes as follow: (i) modelReference, which is used for identifying the reference from a WSDL (Web Services Definition Language) or a XML Schema component to a semantic concept; (ii) liftingSchemaMapping and loweringSchemaMapping, which are used for describing the mappings between semantic data and WSDL type definitions in XML [13]. This approach cannot be easily used in a PLM environment when the business processes are represented with the help of formal or semiformal notations. In spite of some constraints that SAWSDL imposes [12], as well the need to represent with more details the procedural knowledge, we focus our study on discovering an appropriate SASM for our annotation.

In general, the common components of a SASM are elements of knowledge, powerstes of ontology and the semantic relations that relate them with a reference to the language meta-model. The meta-model of SASM is described in Figure 2. One element of knowledge has zero or more semantic annotations. One semantic annotation is composed by one powerset of ontology and defined by one annotation type. An element of knowledge corresponds to one or more elements of meta-model.

1 SAWSDL Working Group. http://www.w3.org/2002/ws/sawsdl/#Introduction

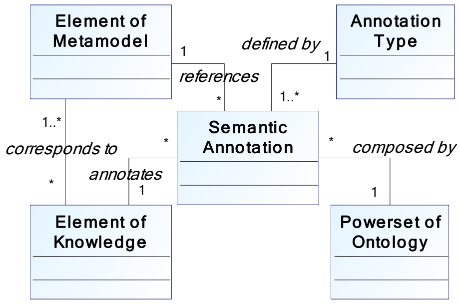

Fig. 2. Semantic Annotation Structure Meta-model.

2.3 Semantic Annotation Framework

In a PLM environment, enterprises are using different kinds of engineering systems to manage their products. Applications in these engineering systems create many corresponding knowledge views for their product information flows. But because of the different specifications, the information among products is represented in many styles. When collaborative actors in or between enterprises need to cooperate, the tacit knowledge that hides behind these knowledge views must be made explicit. Figure 3 illustrates our semantic annotation framework in a PLM environment. There are four main modules: different knowledge views (KV), knowledge cloud (KC), set of Meta-models (MM) and the formal semantic annotation (SA).

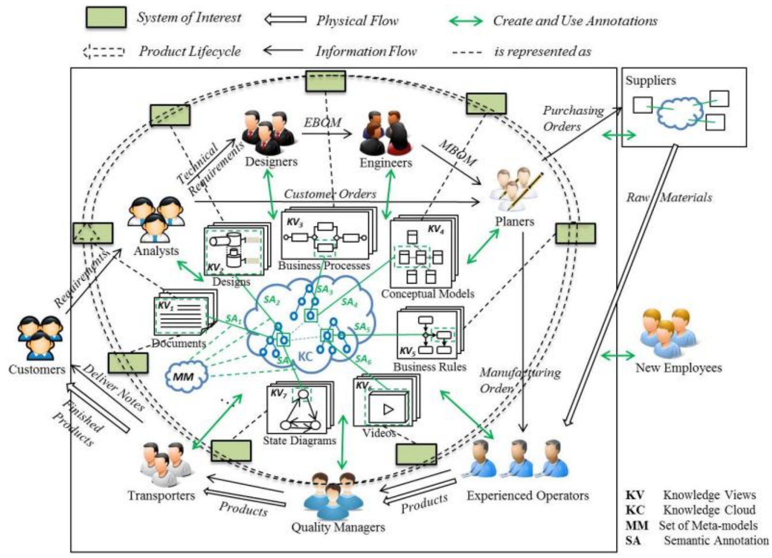

Fig. 3. Semantic Annotation Framework in PLM environment

System of interest is represented into many different KVs along the product lifecycle. For example, business process models graphically depict their internal business procedures [14], conceptual models express the concepts and their mutual relationships [15], state machine diagrams represent the dynamic behaviour of an entity based on its response to events [16], Computer-Aided Design (CAD) models create a prototype of a product [17], and so on.

KC, in this context, is considered as a formal shared definition of concepts and relationships used for describing a domain of knowledge. In the same philosophy of Big Data [18], the ontology concepts that are combined in this KC are integrated from different sources and different levels. KC is composed by several interrelated ontologies and some common relevant domain concepts. They are structured as a loosely connected graph in three different abstract levels: general product ontology, top level, is the common share understanding of product definition and evolution; domain product ontology, middle level, capture the main structure concepts of product; instance product ontology, base level, instantiated the product information in the domain ontology. The links between them are done through similar concepts, for the moment this mapping is done manually. There are many approaches that are focused on representing the product knowledge: ONTO-PDM [4], Edinburgh enterprise ontology [19], PRONTO [20], OntoSTEP [21] and etc.

The set of MM refers to knowledge representation, which is used to explain the semantics of the different elements of knowledge. This set includes, among many others, UML 2.0 meta-model for specifying, constructing, and documenting the artefacts of systems [22], BPMN 2.0 meta-model for business process modelling [14], PNML meta-model (Petri Net Markup Language) for developing an XML-based interchange format Petri nets [23].

The formal semantic annotation $SA = \{\mathcal{M}(\mathcal{E}, \mathcal{P}(O_l)), R\}$ is presented in section 2.1 and 2.2. It makes explicit the hidden semantics embedded in all KVs with their meta-model specifications. The annotations are created and used by all participants (analysts, designers, engineers, planners, operators, quality managers, transporters, new employees, etc.) according to the corresponding confidential and privacy strategies.

Through this semantic annotation framework, collaborative actors annotate their own KVs with the concepts defined in the KC and refer them to the corresponding MMs. This activity can help the process of co-designing, sharing, exchanging, versioning and aligning knowledge throughout the product lifecycle: (i) All the associated knowledge for each annotated element can be located via this framework; (ii) Different knowledge views can be underhanded by their meta-model specifications; (iii) Tacit knowledge that is engraved in the different knowledge views can be made explicit. It will contribute to all processes along the product lifecycle.

Some approaches are similar to our method, but there are some differences that have to be pointed out. An ontology-based framework for integrating product lifecycle knowledge is proposed in [5], which stresses more on the reorganisation of the internal knowledge to fit the external needs. Our method, through the extensive use of semantic annotations, focuses more on carrying out the different views and much less on changing the knowledge expressed by ontologies. In this way the interoperability between the different systems is preserved using the local expressed and formalized knowledge. Li, C. et al present a standardised ontological annotation approach [24], OntoCAD, which is used to support multiple engineering viewpoints in CAD Systems. It is an interesting way to formalise the design step and it mostly focuses on the CAD part. Our domain of interest is all knowledge views along the product lifecycle.

3 Case Study

In order to explain how this semantic annotation proposition can be applied, in this section, we focus on the semantic annotation of the manufacturing processes in a product lifecycle. This case study is based on the cooperative production between two production sites: AIPL (Atelier Inter-Établissements de Productique Lorrain, France) and DIMeG (Dipartimento di Innovazione Meccanica e Gestionale, Italy).

Product models are designed at DIMeG with ProEngineer CAD system, which generates product technical and geometrical information into an EBOM (Engineering Bill of Material). However, the EBOM information represents the product structure from the designer point of view, which may not include every data needed by ERP (Enterprise Resource Planning) system and MES (Manufacturing Execution System) to support production [25]. For this reason, when AIPL received EBOM from DIMeG, they need to create a BOP (Bill of Process) according to EBOM. As can be seen from Figure 4, the manufacturing processes are planned as follow:

- Bar cutting process, cut 3 meter aluminium bar into 1 meter.
- Base turning process, chip a bar into the several design bases.
- Disc cutting process, cut galvanized plate and magnetic plate and into discs.
- Part sticking process, stick galvanized or magnetic discs with different bases.
- Product assembling process, use parts to assemble products.

After combining BOP with EBOM (also with other associated information, such as machine capability, stocks, company schedules and etc.), MBOM (Manufacturing Bill of Material) is generated.

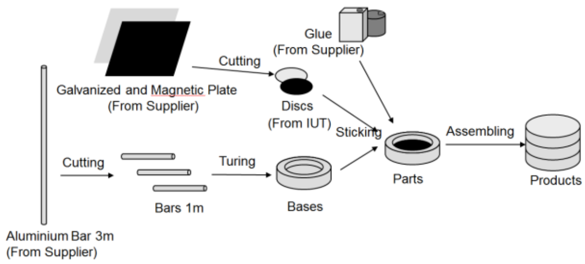

Fig. 4. Manufacturing Processes of AIPL Products

Sage X3 ERP system and Flexnet MES are used in AIPL. This site is in charge of purchasing row materials, outsourcing components and manufacturing products.

- In the purchasing part, based on the stock states and resources, ERP will generate a set of purchasing orders. The row materials (aluminium bar 3m, galvanized plate, magnetic plate and glue) will be purchased from different suppliers.

- In the outsourcing part, because the lack of disc cutting equipment, AIPL can't perform the disc cutting process. Galvanized plate and magnetic plate need to be delivered to IUT (Institute Universitaire de Technologie Nancy-Brabois).
- In the manufacturing part, the ERP system sends work order suggestions to the MES, which proposes the production schedules, and the MES performs the production and updates the stocks information for the ERP system.

At the end, all the qualified products are packaged in boxes and dispatched from AIPL to DIMeG, which will be delivered to the customers. The business process models in Figure 5 represent the product lifecycle of the AIPL products.

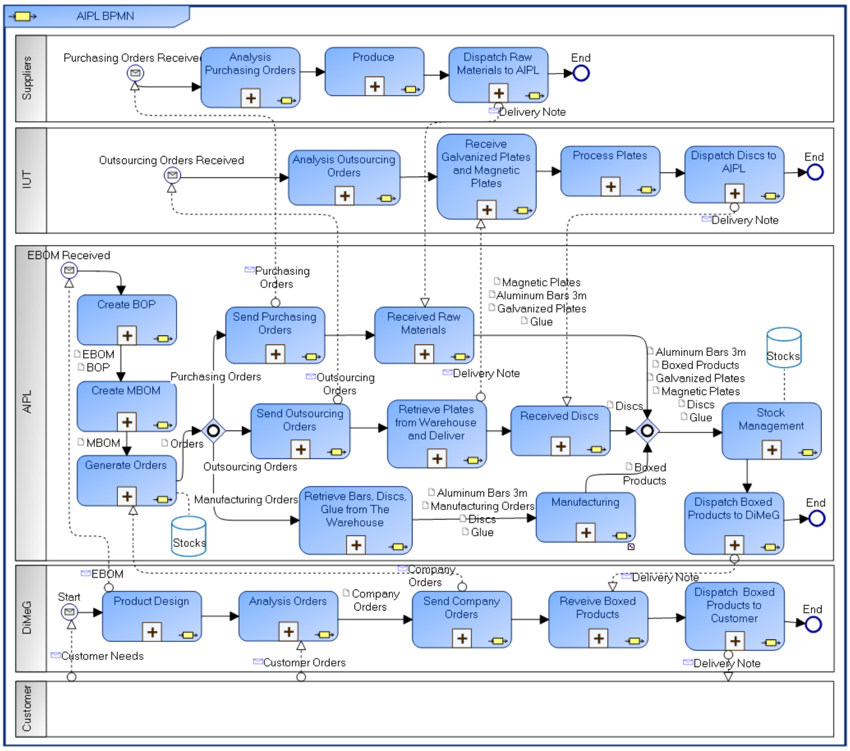

Fig. 5. Product Lifecycle of the AIPL Products

Because of the page limit, we will only present the part of the semantic annotations in this section that will be related to the sticking process of parts and assembling process of Prod5. Figure 6 illustrates these two processes:

- On the left side of the figure, it shows that glue A (AN 302-50) is used to stick galvanized disc with base (P10 and P88), glue B (HR-496) is used to stick magnetic disc with base (P11 and P60).
- On the right side of the figure, it shows that product Prod5 is composed by one P10, one P11, one P88 and one P60.

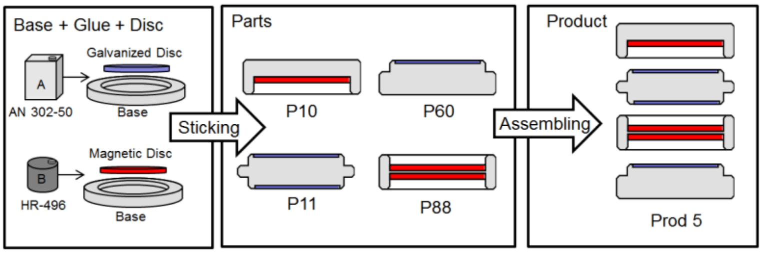

Fig. 6. Sticking and Assembling Processes of Prod 5

There is some tactic knowledge hidden in the experienced operator's and in the process designer's mind, for example:

- Under the conditioning temperature of around 20 ℃, the full cure time of glue A is 6 hours, and glue B is 24 hours. The sticking parts will be not stable if they are assembled before the full cure time. In this case, the stock of parts will became available only if they achieve the full cure time.
- In order to improve the production rate, Prod 5 can be assembled in different ways, which is dependent on the stock level of four parts. There exist three possible configurations named g1, g2 and g3: (i) g1 is the minimum stocks of p60 and p88; (ii) g2 is the minimum stocks of p11 and p88; (iii) g3 is the minimum stocks of p10 and p11. The associated business rule is:
  - If (g1 >= g2) and (g1 >= g3) then perform start assembly with P60,88;
  - If (g2 >= g3) and (g2 > g1) then perform start assembly with P88,11;
  - If (g3 > g2) and (g3 > g1) then perform start assembly with P11,10.

Figure 7 illustrates the semantic annotation example for above processes. In this figure, there are three types of knowledge views: business process model (Petri net and BPMN 2.0 instances), business rule and conceptual model.

In BPMN 2.0 instances, "Sticking" is annotated as $AIPL\_Manufacture\_001 = \{m_{\sim}\langle Sticking, P001\rangle, "SubProcess, BPMN\ 2.0"\}$, in which, the powerset P001 is labelled by the ontology concept "*Sticking: ProductSegmentType*". "Assemble Prod5" is annotated as $AIPL\_Manufacture\_002 = \{m_{\sim}\langle Assemble\ Prod5, P002\rangle, "SubProcess, BPMN\ 2.0"\}$, in which, the powerset P002 is composed by two ontology concepts, "*Prod5:ProductDefinitionType*" and "*Prod5_Assembly:ProductSegmentType*".

In business rule, Rule 001 is annotated as $Business\ Rule\_001 = \{m_{\sim}\langle Sticking\ Rule, P001\rangle, "Natural\ Language"\}$. We can easily find the connection between "Sticking" and "Rule 001". This business rule is in the form of natural language, which explicit the tactic knowledge behind the "Sticking" process. According to this associated information, new operators in the sticking process can avoid making mistakes.

In Petri net, "Prod5 assemble states" is annotated as $Petri\ net\_001 = \{m_{\sim}\langle Prod5\ assemble\ states, P002\rangle, "PNML"\}$. We can find the connection between "Prod5 assemble states" and "Assemble Prod5". The meta-model of this Petri net is based on the PNML. The Petri net fully explains the state changes and business rule during in the whole Prod5 assemble process.

In conceptual model, “Operation” is annotated as $Conceptual\,Model\_001 = \{m_{\cap}\langle Operation, P003\rangle, \text{"}Class, UML\ 2.0\text{"}\}$. “Article” is annotated as $Conceptual\,Model\_002\ = \{m_{\cap}\langle Article, P004\rangle, \text{"}Class, UML\ 2.0\text{"}\}$. The powerset P003 is labelled by the ontology concept “ProductSegmentType” and P004 is labelled by the ontology concept “ProductDefinitionType”. The relations between conceptual model and process model can be inferred by using semantic annotation.

The ontology used to support the semantic annotation in this case study is Onto-PDM. It is designed to formalize all technical data and concepts during the product lifecycle into product ontology [4]. It is a suitable approach to explain the semantics between those concepts.

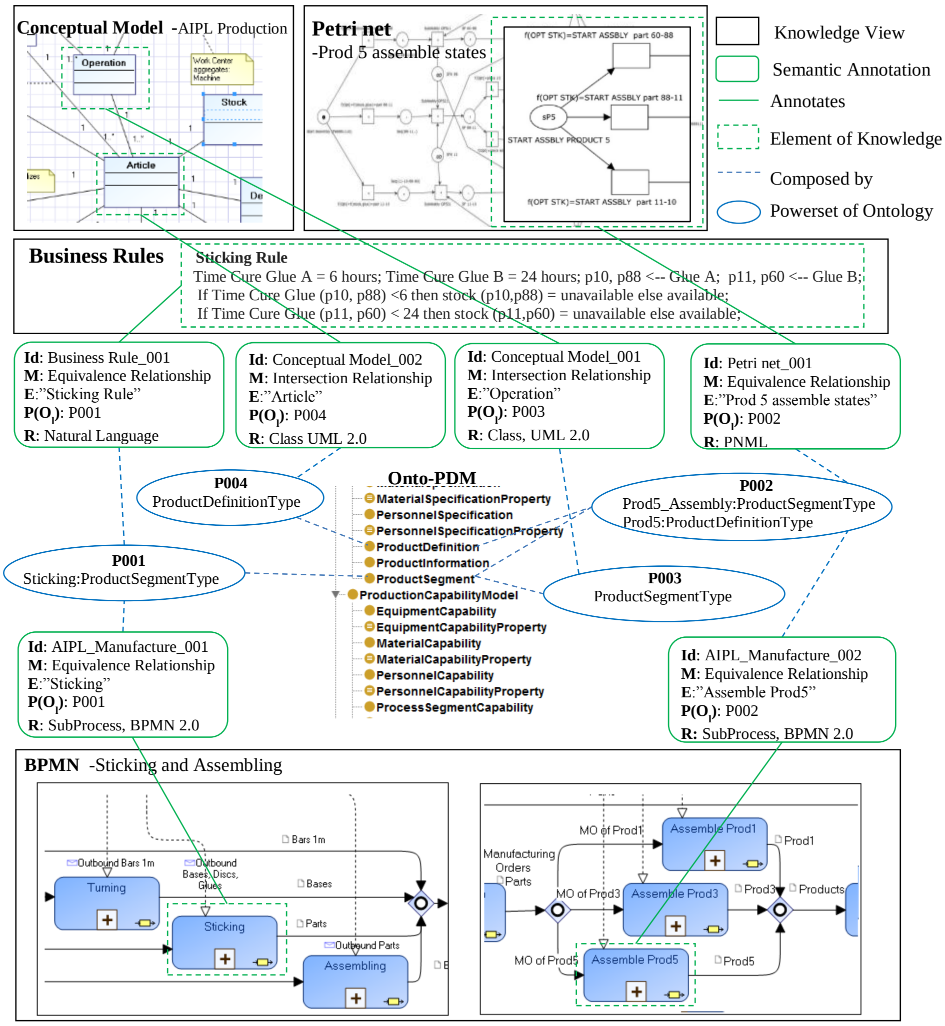

Fig. 7. Semantic Annotation Examples

This example shows the possible use of semantic annotation to create formal connection between different knowledge representations: behavioural knowledge (BPMN diagram, Petri net, business rule) and structural knowledge (conceptual model).

4 Conclusions

This paper provides a semantic annotation approach, which focused on the interoperability problem, to overcome the semantic gap between enterprise applications along the product lifecycle. We first introduced the system interoperability issues in PLM environment. Then we illustrated an updated semantic annotation definition and structure model. We proposed a semantic annotation framework that can help collaborative actors to overcome the semantic gaps. Finally, a case study based on AIPL and DIMeG product lifecycle is presented for demonstrating how our semantic annotations can be applied in a Business to Manufacturing interoperability context.

Future research will be focused on the following aspects: deeply analyse the interoperability requirements among the enterprise applications during the product lifecycle; use semantic annotation to help the evaluation of semantic gap between collaborative systems; explore the way to make heterogeneous enterprise systems and application interoperate, to help enterprises that use different process model notations to exchange process models and to operate on them; investigate the confidential and privacy strategies of the knowledge sharing in and between enterprises.

References

1. Marchetta, M. G., Mayer, F., Forradellas, R. Q.: A reference framework following a proactive-product approach for Product Lifecycle Management. Computers in Industry 62, pp.672–683 (2011)
2. Wang, X., Li, N., Cai, H., Xu, B.: An Ontological Approach for Semantic Annotation of Supply Chain Process Models. In : OTM 2010: On the Move to Meaningful Internet Systems, Hersonissos, Crete, Greece, pp.540-554 (2010)
3. Morel, G., Panetto, H., Zaremba, M., Mayer, F.: Manufacturing Enterprise Control and Management System Engineering: Paradigms and Open Issues. In : IFAC Annual Reviews in Control, 27 (2), pp. 199-209 (2003)
4. Panetto, H., Dassisti, M., Tursi, A.: ONTO-PDM: Product-driven ONTOlogy for Product Data Management interoperability within manufacturing process environment. Advanced Engineering Informatics, Vol.26(2), April (2012)
5. Chen, Y., Chen, Y., Chu, H.: Development of a mechanism for ontology-based product lifecycle knowledge integration. Expert Systems with Applications: An International Journal, 36 (2), pp. 2759-2779 (2009)
6. Yahia, E., Lezoche, M., Aubry, A., Panetto, H.: Semantics enactment for interoperability assessment in Enterprise Information Systems. Annual Reviews in Control 36(1), April (2012)
7. Boudjlida, N., Panetto, H.: Enterprise Semantic Modelling for Interoperability. In: Proceedings of the 12th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2007, Patras, Greece, pp.25-28 (2007)

8. Liao, Y., Lezoche, M., Panetto, H., Boudjlida, N.: Semantic Annotation Model Definition for Systems Interoperability. In : OTM 2011 Workshops, 6th International Workshop on Enterprise Integration, Interoperability and Networking, Hersonissos, Crete, Greece, Springer-Verlag, LNCS 7046, pp. 61-70, Lecture Notes in Computer Science.(2011)
9. Lin, Y.: Semantic Annotation for Process Models: Facilitating Process Knowledge Management via Semantic Interoperability. PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway. (2008)
10. Di Francescomarino, C.: Semantic Annotation of Business Process Models. PhD thesis, International Doctorate School in Information and Communication Technologies, University of Trento, Trento, Italy (2011)
11. 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 (2001)
12. Kopecký, J., Vitvar, T., Bournez, C., Farrell, J.: SAWSDL: Semantic Annotations for WSDL and XML Schema. IEEE Internet Computing , Vol.11(6), pp.60-67 (2007)
13. Verma, K., Sheth, A.: Using SAWSDL for Semantic Service Interoperability. Tutorial at Semantic Technology Conference, San Jose, CA, USA (2007)
14. Business Process Model and Notation (BPMN) Version 2.0. In: OMG Standard. Available at: http://www.omg.org/spec/BPMN/2.0/
15. Winkler, M., Staníček, Z.: Conceptual modelling using the HIT method. Practical tutorial, Masaryk university Faculty of Informatics (2011)
16. Ambler, S. W.: The Elements of UML 2.0 Style., Cambridge University Press, New York (2005)
17. Patel, M., Ball, A., Ding, L.: Curation and Preservation of CAD Engineering Models in Product Lifecycle Management. In : Proceedings of the 14th International Conference on Virtual Systems and Multimedia, Limassol, Cyprus, pp.59-66 (2008)
18. Jacobs, A.: The pathologies of big data, Communications of the ACM 52(8), August (2009)
19. Uschold, M., King, M., Moralee. S., Zorgios. Y.: The Enterprise Ontology. The Knowledge Engineering Review 13, 31-89 (1998)
20. Vegetti, M., Leone, H., Henning, G.: PRONTO: An ontology for comprehensive and consistent representation of product information. Engineering Applications of Artificial Intelligence, 1305-1327 (2011)
21. Krima, S., Barbau, R., Fiorentini, R., Sudarsan, R., Foufou, S., Sriram, R.D. : OntoSTEP: OWL-DL ontology for STEP. In: Proceedings of 6th International Conference on Product Lifecycle Management, Bath, (2009)
22. Unified Modeling Language (UML) Version 2.0. In: OMG Standard. Available at: http://www.omg.org/spec/UML/2.0/
23. Hillah, L. M., Kindler, E., Kordon, F., Petrucci, L., Trèves, N.: Petri Net Newsletter. In : 10th International workshop on Practical Use of Colored Petri Nets and the CPN Tools , pp.9-28 (2009)
24. Li, C., McMahon, C., Newnes, L.: Progress with OntoCAD: A Standardised Ontological Annotation Approach to CAD Systems. In: International Conference on Product Lifecycle Management (PLM11), Eindhoven, Netherlands, (2011)
25. Tursi, A.: Ontology-Based Approach for Product-Driven Interoperability of Enterprise Production Systems. Co-mentorship PhD Thesis, University Henri Poincaré (France) and Politecnico di Bari (Italy), (2009)