



Multi-aspect Ontology for Semantic Interoperability in PLM: Analysis of Possible Notations

Alexander Smirnov¹ and Nikolay Shilov²(✉)

¹ ITMO University, St. Petersburg, Russia
smir@iias.spb.su

² SPIIRAS, St. Petersburg, Russia
nick@iias.spb.su

Abstract. Product lifecycle management covers a number of stages that deal with different tasks and apply different methods but require intensive information exchange to be efficient. Information support of these stages has to address this problem. However, successful implementation of information support systems requires solving the problem of interoperability of heterogeneous information related to different PLM stages. The paper investigates the problem of developing a single ontology for PLM support taking into account differences between terminologies (multi-aspect ontology) used at various stages of the PLM cycle. Different options of designing ontologies covering multiple domains have been considered. Three of them: (i) ontology localization/multilingual ontologies (ii) granular ontologies, and (iii) ontologies with temporal logics are considered in details and analysed. The analysis is based on the case study of PLM support at the automation equipment producer Festo AG & Co KG.

Keywords: Information management · Interoperability · Multi-aspect Ontology

1 Introduction

Product lifecycle management covers a number of stages that deal with different tasks and apply different methods but require intensive information exchange to be efficient. Information support of these stages has to address this problem. However, successful implementation of information support systems requires solving the problem of interoperability of heterogeneous information related to different PLM stages.

The paper develops the earlier presented work [1, 2] based on a long-time collaboration with automation equipment producer Festo AG & Co KG. This is a worldwide provider of automation technology for factory and process automation with a wide assortments of products (more than 40 000 products of approximately 700 types, with various configuration possibilities) ranging from simple products to complex systems (Fig. 1).

During a number of years of collaboration, an eco-system of software tools aimed at support of various PLM stages within the company has been developed as shown in Fig. 2. Below, the elements of the figure are described in detail.

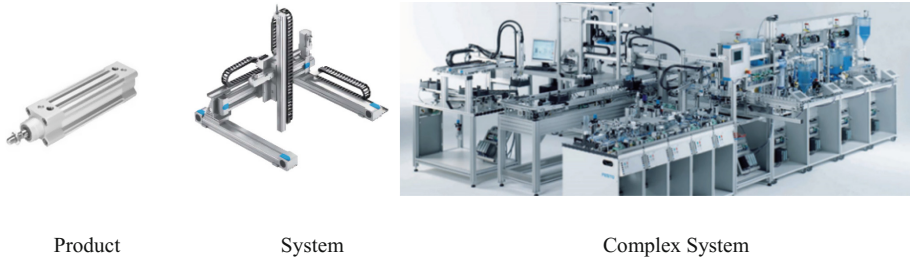


Fig. 1. Assortment range: from simple products to complex systems.

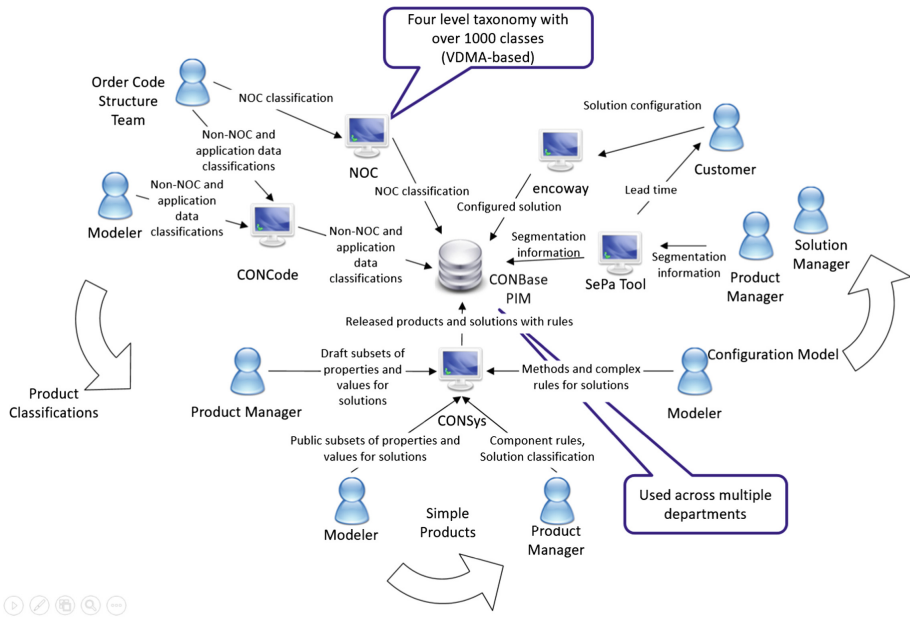


Fig. 2. Information and knowledge management systems developed by the moment.

One of the first projects of the considered company related to this problem was launched in 2010 [3]. It was aimed at modification of work and information flows related to configuration of product combinations. The business process reorganization started with setting up a product ontology in a semi-automatic way originally aimed at product codification (order code scheme) by the NOC tool [3]. The resulting ontology (described in detail in [4]) consisted of more than 1000 classes organized into a four level taxonomy, based on the VDMA (Verband Deutscher Maschinen- und Anlagenbau/Mechanical Engineering Industry Association) classification [5].

The same taxonomy was used in the company's PDM (Product Data Management) and ERP (Enterprise Resource Planning) systems. The ontology structure enabled separation of various types of entities (e.g., physical products and software services)

what made it possible to easily deal with it even though it was rather large. Different specialists could work with different parts of the ontology without the need to overview and manage the whole ontology at once (CONBase Product Information Management). Overall, application of the common ontologies in this particular project has proved itself as a convenient and reliable way of product and system knowledge organisation.

However, extension of the support of different PLM stages has caused appearance of extensions of the central ontology aimed at particular processes (e.g., CONCode to support products, which have descriptions incompatible with the ontology). Complex product modelling and design system (CONSys tool) together with product segmentation policy definition tool (SePa) also introduced additional information to be managed. E.g., application data (an auxiliary component used for introduction of some additional characteristics and requirements to the product, for example, operating temperatures, certification, electrical connection, etc.) had to be added for marketing purposes and combined with other features through defined rules. Development of the product configuration tool (CONFig) was aimed at testing the possibility to configure systems based on the rules stored in the CONBase. The tool supported the configuration process in terms used within the company.

Although some significant results have been achieved in the area of complex product and system information management and configuration, still a lot has to be done to support the whole lifecycle of this type of products. One of the key tasks is to provide a coherent way of integration of different extensions into the common ontology.

Having a number of ontologies is not an efficient way due to the necessity to continuously translate information and knowledge between them. However, plain integration is not possible either. For example, the customers are used to operate different terminology, which does not correspond one-to-one to that used within the company. Besides, customers from different industries also operate different terms.

The paper is aimed at investigating the problem of developing a common ontology for PLM supporting differences between terminologies (multi-aspect ontology) used at various stages of the PLM. Different options of designing ontologies covering multiple domains have been considered. Three of them: (i) ontology localization/multilingual ontologies, (ii) granular ontologies, and (iii) ontologies with temporal logics are considered in details and analysed.

2 Related Work

Ontologies are a mean to represent knowledge about a problem domain in a machine-readable way. They enable obtaining, exchanging and processing information and knowledge based on their semantics rather than just syntax. Ontology is a formal conceptualisation of a particular domain of interest shared among heterogeneous applications [6, 7]. Usually, it consists of concepts existing in the problem domain, relationships between them and axioms. Ontologies are a well-proven tool to solve the interoperability problem, but the problem of applying ontologies to PLM is due to different terminologies used at PLM stages even within one company [8, 9]. E.g., in [10] a model-driven interoperability framework is presented as a technical support of

co-evolution strategy of products and manufacturing systems. The authors address connecting possible product modules managed in the PLM tool to all possible production capabilities managed on the Manufacturing Process Management tool through establishing “connector framework” to match different ontologies.

The whole scale of the problem was identified in [11] and presented in Fig. 3.

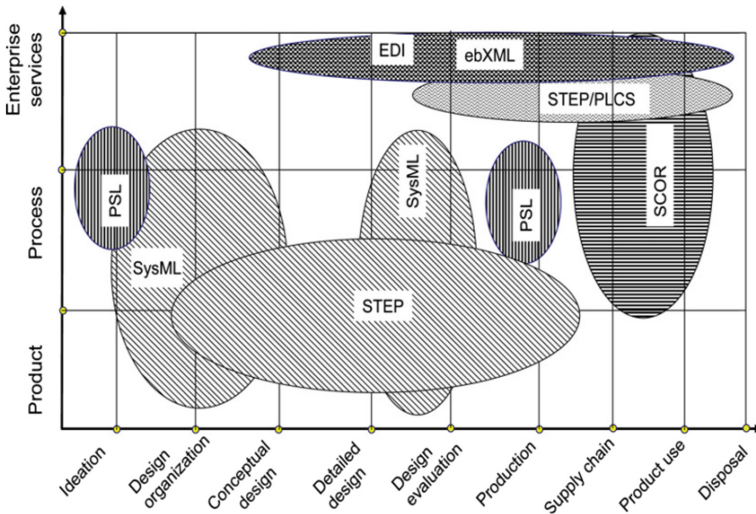


Fig. 3. Usage of different standards at various PLM stages [11].

In [12] a solution is proposed based on semantically annotated multi-faceted ontology for product family modelling to automatically suggest semantically-related annotations based on the design and manufacturing repository on the example of laptop computers. A fragment of the solution is shown in Fig. 4.

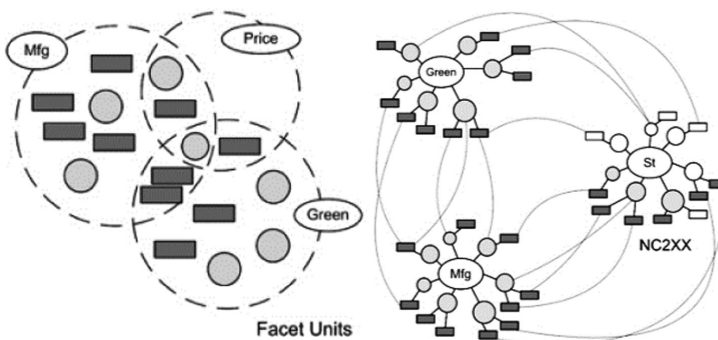


Fig. 4. Building a multi-faceted ontology for PLM (adapted from [12]).

Thus, ontology matching is one of the techniques to solve the problem. However, automatic ontology matching is not very reliable and manual ontology matching requires significant time and efforts. There are efforts aimed at enriching ontologies with additional information (e.g., extension of DAML+OIL for description of configuration problems [13], introducing semantic annotations [14], etc.), however, they still cannot solve the problem of integrating heterogeneous information and knowledge described in different terminology. For example, it is common understanding that domain specific models (e.g., configuration models) can be derived by inheriting or subclassing the ontologies within the general model. Thus, SWRL is a rule-based language for description of constraints is based on OWL and the resulting ontology is an extension of OWL ontology. Then, actual configuration system is implemented using JESS what requires mapping of OWL-based configuration knowledge and SWRL-based constraints into Jess facts and Jess, respectively [15].

As it can be seen there are many efforts aimed at integration of heterogeneous knowledge into a single complex ontology. After an extensive study of the domain, we have identified three main and most promising possibilities, which are discussed below.

3 Possible Notations for Multi-aspect PLM Ontology

In this section we consider three different notations that can be used for implementation of the multi-aspect PLM ontology. The aspects are assumed as different PLM stages, however, in some cases the aspects can be more narrow and assume PLM processes.

3.1 Multilingual Ontology

Multilingual ontologies are aimed at solving terminological issues arising from usage of different languages. Among the terminological issues the following can be selected [16]:

1. Existence of an exact equivalent. This is the easiest case when two terms have completely the same meaning. In real life (when talking of regular languages such as English or German) this is a rear situation, however in a company most of terminology would be the case. For example, “product” would mean the same both during the design stage and the production stage, or in the considered company, “feature” during the design stage means the same as “characteristic” during the production stage.
2. Existence of several context-dependent equivalents. This case assumes that one can choose the right translation (the right equivalent) based on the situation. An example could be the term “modular product” that can stand for both product consisting of several modules or product with some variable characteristics. Treating such ambiguity for a person could be relatively simple but for a machine it can cause significant difficulties.
3. Existence of a conceptualization mismatch. This is an important issue for regular languages, standing for a lack of semantic equivalent for a given term. In case of

PLM this is a much less common issue since the lack of a certain term at a PLM stage usually means that it is just not used (not needed) at this stage.

Usually, such ontologies are built as an ontology with language specific fragments with relationships between terms and it might be a straightforward enough solution for multi-aspect domains. This really helps to overcome the terminological issues, as well as to solve the problem of heterogeneity of information and knowledge between different lifecycle stages.

However, a multilingual ontology is formulated in a single formalism and collecting together for example, configuration knowledge with procurement knowledge would not be possible without losing some semantics. As a result this approach could not solve the problem formulated. On the other hand, multilingual support could be really useful for global companies that have employees speaking different languages, which is actual for the company considered but out of the scope of the presented here research.

3.2 Granular Ontology for PLM

Granular ontologies are based on the integration of ontology-based knowledge representation with the concept of granular computing. Granular computing is based around the notion of granule that links together similar regarding to a chosen criteria objects or entities (“drawn together by indistinguishability, similarity, proximity or functionality” [17]). The granules can also be linked together into bigger granules forming multiple levels of granularity.

From the knowledge representation point of view, a granule can be considered as a chunk of knowledge made about a certain object, set of objects or sub-domain [18]. When speaking about PLM, higher-level granules can combine knowledge related to a certain PLM stage, and lower-level granules can be related to particular PLM processes (Fig. 5). A level is a collection of granules of similar nature. The hierarchy of granules then would form a hierarchy of PLM processes.

Granular ontologies seem to be a suitable solution to support various PLM stages: they enable splitting the domain in smaller areas with consistent terminology and formalisms. The possibility to form a hierarchy (generalisation) is also beneficial due to the possibility to define generic concepts and relationships at higher levels.

However, PLM stages usually overlap in terms of used information and knowledge (Fig. 6). This means that there exist multiple processes that assume collaboration and usage of the same information and knowledge. Pure granular ontologies cannot solve the problem of terms having different meaning at different PLM stages or different company departments. There are multiple efforts in the area of rough granular computing [19–21], however, they are not directly related to ontology design. As a result, additional research in this area is required. Another possibility is to extend a granular ontology with a concept that would enable certain “roughness” of it, and the following section proposes such a possibility.

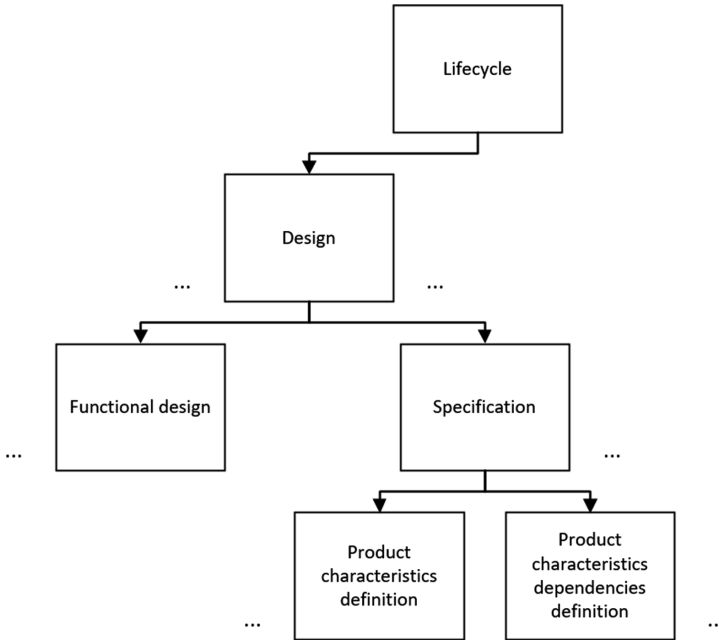


Fig. 5. Example of PLM ontology granules from whole lifecycle down to particular processes.

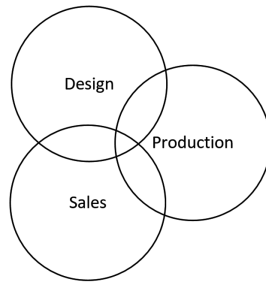


Fig. 6. Example of overlapping PLM stages.

3.3 Temporal Logics-Based Ontology for PLM

The authors of [22] propose to address the problem of terms having different meaning at different PLM stages or different company departments through usage of temporal logics. The idea of using temporal logics in describing PLM originates from the fact that most of product related information and knowledge is used in the product lifecycle only during some stages.

PLM is basically always associated with time. The most often met PLM schemes are “time arrow” (when the PLM stages follow one after another) and “time wheel” (when the time arrow of PLM stages is connected into one or several circles). Time is

also considered as one of the key resources in PLM, when, for example one speaks of decreasing lead time or increasing the product usage period.

The approach presented in [22] is based on the fuzzy extension of temporal logics to enable links and overlapping between different stages of the lifecycle. The metaphor used in the approach is based on the idea of representing lifecycle stages as time intervals with fuzzy duration.

The ontology (ONT_{LC}) is described by the following formula:

$$ONT_{LC} = \langle C_{LC}, R_{LC}, O_{LC}, T_{LC} \rangle, \text{ where}$$

C_{LC} is the set of concepts related to lifecycle (all the concepts of the ontology used at all stages of the lifecycle),

R_{LC} is the set of relations between the concepts (including structural relationships between products and their characteristics),

O_{LC} is the set of operations over concepts and/or relations,

T_{LC} is the set of temporal characteristics for lifecycle.

Since the ontology is aimed at separation of concepts between lifecycle stages, the lifecycle systemic kernel is represented as the following triple:

$$ONT_S = \langle S, R_S, O_S \rangle, \text{ where}$$

S is the set of lifecycle stages,

R_S is the set of relations between the stages,

O_S is the set of operations used on the stages.

As it was mentioned the lifecycle stages are considered as time intervals $s = [t, t^+]$, with starting and ending time points t and t^+ respectively. However, in order to indicate the overlapping of stages, the intervals are considered to be fuzzy.

Though the usage of granular ontology with temporal logic as a notation for multi-aspect PLM representation looks complex, it can solve the heterogeneity problem arising from different mental models at different product lifecycle stages. Besides, the “complexity” of this approach makes it possible to include different representations related to different processes of the lifecycle preserving the expressiveness of the representations and languages used unlike multilingual ontologies. As a result, it was concluded define that in the considered case the semantic interoperability support in PLM based on a multi-aspect ontology should be implemented via the notation of granular ontology with temporal logic.

4 Conclusion and Future Work

The paper represents an analysis of the possible notations for building a multi-aspect ontology supporting semantic interoperability in PLM.

Building a multi-aspect segmented ontology basically consisting of a number of ontologies (sub-ontologies) can be based on using unchanged source ontologies and the overall structure of such an ontology would be simple and easy to process. However, this would lead to the necessity of continuous translation of information and knowledge between different representations and standards, which is not an easy task. The dynamic structure of the terminology would make this issue even more complex for solving. As a result, this solution was not accepted.

Multilingual ontologies can solve the problem of heterogeneity of information and knowledge but lack the possibility to support multiple problem-specific formalisms. This solution was not accepted wither but it was noted that multilingual support could be useful for global companies like the considered one, which has employees speaking different languages.

It was identified that in the considered case the semantic interoperability support in PLM should be based on granular multi-aspect ontology extended with temporal logics elements. The pilot efforts related to building smaller ontologies with the purpose of methodology validation proved its viability and potential efficiency. The future research is aimed at building a larger-scale ontology including real company data.

Acknowledgements. The paper is due to collaboration between SPIIRAS and Festo AG & Co KG, State Research # 0073-2018-0002, projects funded by grants ## 18-07-01201, 18-07-01272 of the Russian Foundation for Basic Research. The work has been partially financially supported by Government of Russian Federation, Grant 074-U01.

References

1. Smirnov, A., Kashevnik, A., Teslya, N., et al.: Knowledge management for complex product development framework and implementation. *IFIP Adv. Inf. Commun. Technol.* **409**, 110–119 (2013)
2. Smirnov, A., Shilov, N., Oroszi, A., Sinko, M., Krebs, T.: Changing information management in product-service system PLM: customer-oriented strategy. In: Ríos, J., Bernard, A., Bouras, A., Foufou, S. (eds.) *PLM 2017. IAICT*, vol. 517, pp. 701–709. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-72905-3_62
3. Oroszi, A., Jung, T., Smirnov, A., et al.: Ontology-driven codification for discrete and modular products. *Int. J. Prod. Dev.* **8**, 162–177 (2009). <https://doi.org/10.1504/IJPD.2009.024186>
4. Smirnov, A., Shilov, N.: Ontology matching in collaborative recommendation system for PLM. *Int. J. Prod. Lifecycle Manag.* **6**, 322–338 (2013). <https://doi.org/10.1504/IJPLM.2013.063210>
5. VDMA. German Engineering Federation (2018)
6. Gruber, T.R.: A translation approach to portable ontology specifications. *Knowl. Acquis.* **5**, 199–220 (1993). <https://doi.org/10.1006/knac.1993.1008>
7. Staab, S., Studer, R.: *Handbook on Ontologies*. Springer, Heidelberg (2009). <https://doi.org/10.1007/978-3-540-92673-3>
8. Asmae, A., Souhail, S., El Moukhtar, Z., Hussein, B.: Using ontologies for the integration of information systems dedicated to product (CFAO, PLM...) and those of systems monitoring (ERP, MES...). In: 2017 International Colloquium on Logistics and Supply Chain Management (LOGISTIQUA), pp. 59–64. IEEE (2017)

9. Palmer, C., Urwin, E.N., Young, R.I.M., Marilungo, E.: A reference ontology approach to support global product-service production. *Int. J. Prod. Lifecycle Manag.* **10**, 86 (2017). <https://doi.org/10.1504/IJPLM.2017.083003>
10. Lafleur, M., Terkaj, W., Belkadi, F., Urgo, M., Bernard, A., Colledani, M.: An onto-based interoperability framework for the connection of PLM and production capability tools. In: Harik, R., Rivest, L., Bernard, A., Eynard, B., Bouras, A. (eds.) *PLM 2016. IAICT*, vol. 492, pp. 134–145. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-54660-5_13
11. Rachuri, S., Subrahmanian, E., Bouras, A., et al.: Information sharing and exchange in the context of product lifecycle management: role of standards. *Comput. Des.* **40**, 789–800 (2008). <https://doi.org/10.1016/j.cad.2007.06.012>
12. Lim, S.C.J., Liu, Y., Lee, W.B.: A methodology for building a semantically annotated multi-faceted ontology for product family modelling. *Adv. Eng. Inform.* **25**, 147–161 (2011). <https://doi.org/10.1016/j.aei.2010.07.005>
13. Felfernig, A., Friedrich, G., Jannach, D., et al.: Configuration knowledge representations for semantic web applications. *Artif. Intell. Eng. Des. Anal. Manuf.* **17**, 31–50 (2003). <https://doi.org/10.1017/S0890060403171041>
14. Liao, Y., Lezoche, M., Panetto, H., Boudjlida, N.: Semantic annotations for semantic interoperability in a product lifecycle management context. *Int. J. Prod. Res.* **54**, 5534–5553 (2016). <https://doi.org/10.1080/00207543.2016.1165875>
15. Yang, D., Miao, R., Wu, H., Zhou, Y.: Product configuration knowledge modeling using ontology web language. *Expert Syst. Appl.* **36**, 4399–4411 (2009). <https://doi.org/10.1016/j.eswa.2008.05.026>
16. Espinoza, M., Montiel-Ponsoda, E., Gómez-Pérez, A.: Ontology localization. In: *Proceedings of the Fifth International Conference on Knowledge Capture - K-CAP 2009*, pp. 33–40. ACM Press, New York (2009)
17. Zadeh, L.A.: Is there a need for fuzzy logic? *Inf. Sci. (Ny)* **178**, 2751–2779 (2008). <https://doi.org/10.1016/j.ins.2008.02.012>
18. Calegari, S., Ciucci, D.: Granular computing applied to ontologies. *Int. J. Approx. Reason.* **51**, 391–409 (2010). <https://doi.org/10.1016/j.ijar.2009.11.006>
19. Jankowski, A., Skowron, A.: Toward rough-granular computing. In: An, A., Stefanowski, J., Ramanna, S., Butz, C.J., Pedrycz, W., Wang, G. (eds.) *RSFDGrC 2007. LNCS (LNAI)*, vol. 4482, pp. 1–12. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72530-5_1
20. Inuiguchi, M., Hirano, S., Tsumoto, S.: *Rough Set Theory and Granular Computing*. Springer, Heidelberg (2003). <https://doi.org/10.1007/978-3-540-36473-3>
21. Polkowski, L., Skowron, A.: Rough mereological calculi of granules: a rough set approach to computation. *Comput. Intell.* **17**, 472–492 (2001). <https://doi.org/10.1111/0824-7935.00159>
22. Tarassov, V., Fedotova, A., Stark, R., Karabekov, B.: Granular meta-ontology and extended Allen’s logic: some theoretical background and application to intelligent product lifecycle management systems Valery. In: Schwab, I., van Moergestel, L., Gonçalves, G. (eds.) *INTELLI 2015: The Fourth International Conference on Intelligent Systems and Applications*, St. Julians, Malta, pp. 86–93 (2015)