



HAL
open science

Modelling of Engineering and Manufacturing Data Structures Through a DSM-Based Approach

Yana Brovar, Yaroslav Menshenin, Dominik Knoll, Clement Fortin

► **To cite this version:**

Yana Brovar, Yaroslav Menshenin, Dominik Knoll, Clement Fortin. Modelling of Engineering and Manufacturing Data Structures Through a DSM-Based Approach. 18th IFIP International Conference on Product Lifecycle Management (PLM), Jul 2021, Curitiba, Brazil. pp.96-109, 10.1007/978-3-030-94399-8_8. hal-04195219

HAL Id: hal-04195219

<https://inria.hal.science/hal-04195219v1>

Submitted on 4 Sep 2023

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.

Modelling of Engineering and Manufacturing Data Structures through a DSM-based Approach

Yana Brovar¹[0000-0001-9923-9772], Yaroslav Menshenin¹[0000-0002-1547-9857], Dominik Knoll²[0000-0001-8593-4731], and Clement Fortin¹[0000-0002-9761-3107]

¹ Skolkovo Institute of Science and Technology, Bolshoy Boulevard 30, bld. 1, Moscow, Russia 121205

² Babeş-Bolyai University, Str. Mihail Kogălniceanu, nr. 1, 400084, Cluj-Napoca, România
yana.brovar@skoltech.ru

Abstract. In order to achieve effective Digital Engineering solutions, it is necessary to solve the problem of integrating Product Lifecycle Management (PLM) and Model-Based Systems Engineering (MBSE) domains. Indeed, design processes are supported by a variety of ontologies representing each of the domains. Since none of the ontologies is sufficient on its own to describe the complete lifecycle, we propose an ontological engineering approach to model Engineering and Manufacturing data structures. In this paper, we address the fundamental issue of semantic integration of ontologies that can support Digital Engineering solutions. To achieve this goal, we define various types of interfaces between systems ontologies such as the Object-Process Methodology (OPM), the Core Product Model (CPM), and the Manufacturing Process Management (MPM). Revealing the connections between these ontologies has the potential to provide eventually more efficient management of all information throughout the entire product lifecycle. To accomplish this, we propose to use the fundamental OPM structural and procedural links that can be used to explicitly represent all causal interfaces within MBSE and PLM domains using the Design Structure Matrix (DSM)-based approach.

Keywords: product design, ontology, system relationships, DSM, MBSE.

1 Introduction

Currently, the product lifecycle has no continuous digital thread. A particular discontinuity appears between the conceptual design phase and the subsequent stages of the design process. Integration should enable design teams to have a digital data thread and processes through the product development process - from the Model-Based Systems Engineering (MBSE) to the Product Lifecycle Management (PLM), including the manufacturing process definition and other downstream applications. In [1], the authors emphasized the complexity associated with "... the need to work not only with different terminologies, but also with different formalisms ..." when building a multidimensional PLM ontology to ensure the interaction of PLM processes. A similar problem occurs when integrating PLM data with earlier and later design stages.

Successful data exchange to support the development of complex products can be realized by integrating the ontologies that underlie the applications used by teams during the design process. In this paper, we analyze the importance of defining the types of relationships between ontologies' entities.

In our work, we use the Design Structure Matrix (DSM) [2] to map the core entities from conceptual to detailed design and to manufacturing stage. In doing so, we identify overlaps and gaps in the digital engineering information exchange and define the types of interconnections between entities of ontologies representing each design phase. DSM is recognized as an effective tool for representing interfaces within complex systems as well as for encoding relationships types.

As a set of relationships types, we propose to use fundamental structural and procedural links [3] of the Object-Process Methodology (OPM). Their comprehensiveness allows to effectively describe any complex system, including ontologies.

The paper is structured as follows: in section 2, we discuss MBSE, PLM and manufacturing data modeling based on three models – Object-Process Methodology (OPM) (sub-section 2.1), Core Product Model (CPM) (sub-section 2.2), and Manufacturing Process Management (MPM) (sub-section 2.3). In section 3, we discuss ontology engineering highlighting important constituents towards engineering and manufacturing data structures integration, such as: relationships and interfaces (sub-section 3.1) and DSM (sub-section 3.2). In section 4 we discuss the OPM set of interfaces, describing structural and procedural links. After this, in section 5, we demonstrate the proposed approach could be applied to the identification of types of relationships between chosen ontologies. We discuss the results of our work in section 6 and conclude in section 7.

2 System Ontologies

The term "ontology" is borrowed from philosophy, where it denotes a systematic description of Existence. One of the first definitions of ontology in the context of knowledge sharing was given by Neches et al. [4], which states that the ontology “defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary”. Thus, this definition indicates that the ontology includes not only terms, but also the knowledge that can be obtained from it. Gruber [5] defined the notion of ontology as an “explicit specification of a conceptualization”, implying that ontology is a description of entities that exist within some domain of interest and the relationships among these entities.

In this subsection, we overview the following ontology models that are studied throughout a paper: OPM [3] (sub-section 2.1), CPM [6] and Open Assembly Model (OAM) [7] (sub-section 2.2), and MPM [8] (sub-section 2.3).

2.1 Object-Process Methodology (OPM)

Conceptual modeling languages provide a powerful means to model the design environment and manage the information across the system/product lifecycle. Such languages are exploited in different domains and sectors. For example, BPMN [9] is primarily used in business environment to model business processes; SysML [10] is widely used in aerospace industry; IDEF0 was found useful across industries to define process and data flows [11]. The modeling languages are used during the development and the evaluation of the alternative solution concepts. OPM [3] has been proposed as a conceptual modeling language with just three core entities - object, process, state; and their relationships - structural and procedural links [3]. This language has been extensively used for a variety of cases and mapped against other conceptual modeling approaches and languages [12], [13]. In our work we use OPM, since it comprehensively allows describing any complex system, including ontologies.

2.2 Engineering System/Core Product Model

To better illustrate the proposed approach to ontology engineering, we use the CPM model developed by NIST as representative ontology at the detailed design stage. It has system-like elements embedded in a solid modelling ontology, which can also represent a standalone CAD system. The main entity in the core model is an artifact that represents a distinct element of a product. In turn, the artifact entity has three main entities representing the main characteristics which include the form, the corresponding function and the behavior of the product. The function entity describes what the artifact is supposed to do based on the engineering requirements and the stakeholders' needs. In turn, the form entity represents the design solution for function implementation in terms of geometry and material. The behavior entity describes how the form of the artifact implements its function [6].

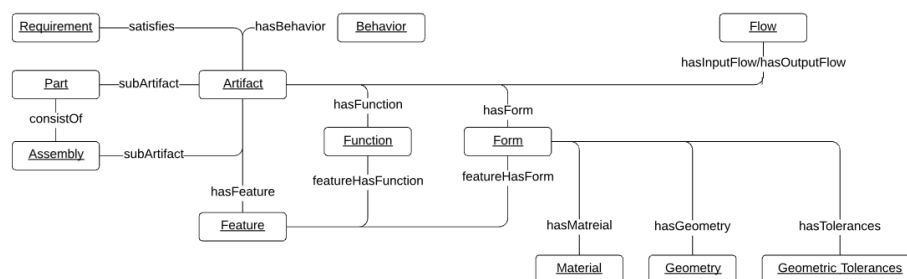


Fig. 1. Engineering System/CPM data structure.

To properly capture information about assembly and system-level tolerances, the CPM framework was extended with the Open Assembly Model [7]. The OAM emphasizes information requirements for the part features and assembly relationships. By doing this, the basic CPM data structure is complemented by basic entities that are critical

to product definition at the detailed design stage such as: “Part”, “Assembly” and “Geometry Tolerance”. “Assembly” entity incorporates information about assembly relationships and component composition. In turn, “Geometric Tolerance” is a critical issue in the product design which characterizes assembly analysis and controls the variability of linear dimensions.

Thus, in the paper, we consider the Engineering System/CPM (ES/CPM) ontology based on extended CPM. The basic data structure of the ES/CPM shown in Figure 1 is based on the CPM data structure proposed by Fenves [6], and is supplemented by entities from OAM, relevant for the detailed design phase. It shows all the entities that can be represented in the ontology (rounded rectangle), as well as the connections that demonstrate all the different kinds of relationships (aggregation, association, etc.) between CPM entities and complemented ones.

2.3 Engineering and Manufacturing Data Structures

The Bill of Material (BOM) represents a data structure aggregating and presenting product data by disclosing hierarchical and logical dependencies among assemblies, parts and all relevant attached objects. The BOM is a product structure representing particular function within the product lifecycle. It organizes the product information in such a way that facilitates access to product data considering the appropriate field of expertise and the lifecycle maturity.

Two types of BOMs are frequently encountered:

- The As-designed or Engineering BOM (eBOM) is oriented towards a functional decomposition of the product, and thus discloses its systems arrangement with interface control objects. The product, sub-assembly and component design data are accessible through this type of structure. The Engineering BOM is supported by a data structure that essentially represent the physical definition of the product where space is the primary representation.

- The As-planned or Manufacturing BOM (mBOM) refers to a data structure reflecting how the product is to be assembled and manufactured. The manufacturing resources include manufacturing materials, manufacturing dispersions, machine-tools, tools and facilities. Process plans are central in this type of data structure, which is largely dominated by time, where manufacturing states play an important role. Process plans can be represented by various means such as a list of operations, the Manufacturing BOM for large assemblies or product states defining the various assembly stations and manufacturing steps.

There are two approaches to consider the various BOMs in a PLM system: the first one is based on the Master Model data structure where the various BOMs are considered as different views and a second one based on the complementary data structure principles which approaches the various BOMs has independent but interrelated data structures, supporting the various stages of the product lifecycle. Their basic principle is thus one of unity in diversity rather than a monolithic data structure.

This first approach sees the Engineering and Manufacturing data as part of a unique and centralized data structure which ensures coherency and what is called the single source of truth. This unique and monolithic data structure cannot easily accommodate

the various functions that need to be carried-out throughout the product lifecycle and much of the excessive complexity of current PLM systems comes from this approach.

The second approach is supported by complementary data structures and is more flexible, allowing each product/system development function to represent its data by its own data structure. The coherency and single source of truth coming from the product definition is ensured through sophisticated relationships.

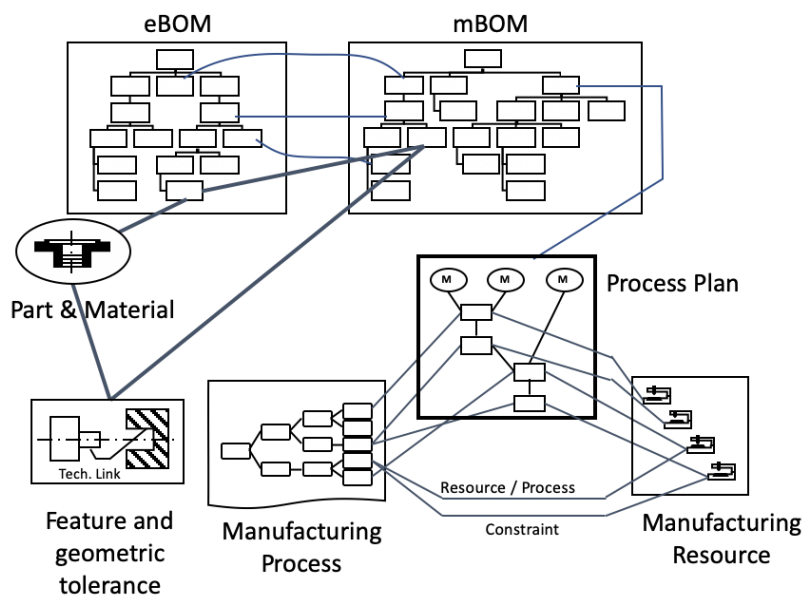


Fig. 2. Complementary Engineering and Manufacturing data structures.

2.4 Manufacturing Process Management (MPM)

MPM has been developed based on the principle of unity in diversity between the various applications along the product development chain [13].

The MPM approach has been shown in large industrial implementations to enable reliable data integrity, ensure configuration management and support a collaborative process between the engineering and manufacturing teams. MPM can bridge the worlds of engineering and production while supporting the manufacturing process definition [13]. Its ontology is based on complementary information relationships linking design and manufacturing product definitions, through the eBOM (as-designed product structure) managed by the PDM system, and the mBOM (as-planned product structure) respectively, managed by the MPM system; it effectively contributes to the coherence between the digital and physical twins, the latter being generated by the manufacturing processes including configuration management. The inclusion of the MPM can play a central role in concurrent engineering, providing synchronization of design and manufacturing processes definitions through a digital collaborative environment [7]. In concrete terms, this represents the transition from the 3D product definition defined by the

engineering team to the complete manufacturing process definition, then to production planning, and integration with the Enterprise Resource Planning (ERP) and other downstream applications.

3 Ontology engineering

As illustrated in the previous section, different ontologies are used at different lifecycle phases. Identifying the connections among these ontologies has the potential to allow for more effective management of all information throughout the product lifecycle. None of the ontologies is sufficient on its own to describe the product in its entirety, but they should complement each other and together allow a more holistic approach.

Ontology engineering in the context of product development aims to support knowledge management by providing semantic interoperability between ontologies implemented by different teams at each phase of product development. Mizoguchi et al. [15] defined the purpose of ontology engineering as: “to provide a basis for building models of all things in which computer science is interested”. Thus, ontology engineering includes the development of methods, approaches and tools for creating and supporting ontologies.

There exists another type of ontology engineering definition based on the process followed to build the ontology that emphasizes the relationship between ontologies and knowledge bases. Bernaras et al. [16] define ontology as a means for explicitly describing the conceptualization of the knowledge represented in a knowledge base. This definition was proposed in the framework of the KACTUS project, where the authors presented the method for constructing an ontology in the domain of electrical networks by ‘extracting’ an ontology from a knowledge base.

Many methodologies have been proposed for ontology development. This distributed nature of ontology building has led to the need to evaluate and integrate ontologies spanning overlapping areas. To create a coherent and consistent data exchange process, several integration processes have been studied: mapping, which is mainly related to the representation of correspondences between ontologies; merging, which is associated with creation of a single new ontology based on several initial ontologies, and alignment, which is associated with the automatic detection of correspondences between ontologies. While we propose an approach from the complementarity perspective to realize supportive communications between ontologies by defining relations between entities of ontologies that are representative for each design phase.

3.1 Relationships/Interfaces

An ontology consists of definitions of terms, as well as definitions of how they are interrelated [17]. Product interface management is an essential element of engineering and manufacturing data structures integration allowing to create a digital twin of a complex system. In this paper, we consider ontologies as complex systems themselves and model their entities as systems components. System interfaces contain essential information about the system and its decomposed parts - both at the subsystem level and at the processes / functions level. Fosse and Delp [18] emphasized that describing and

specifying interfaces in general is a challenging task. For this reason the interfaces are defined in a variety of ways in the design literature. The NASA System Engineering Handbook [19] defines interfaces as “any boundary between one area and another”. There are other points of view on the definition of interfaces, such as a logical or physical relationship [20] or a linkage [21].

3.2 DSM

DSM as a tool is used in universities, business, industry to support managing system complexity and relationships between the system elements [2]. The simplicity and utility of DSM are united in the ability to quantify the system architecture; and its applicability to a wide variety of architectural, organizational, and multi-domain problems. Over time, DSM has proven to be effective as a tool for managing relationships in a complex system or product [2]. It is not by coincidence that the DSM as a method itself has been extended to a variety of directions, such as Multi-Domain Matrix (MDM) [22] application to Building Information Modelling (BIM) [23]; and dynamic modeling of socio-technical systems [24].

These premises make the DSM/MDM approach suitable to the goal of our research: mapping the core entities from conceptual to detailed design and to manufacturing stage.

4 Types of Relationships

The OPM methodology [3], [25] is a powerful approach for effectively representing the interfaces of complex systems. Therefore, the set of interfaces proposed in [3] can be used to explicitly represent all casual interfaces within the system, which can then be translated into a DSM representation [26]. OPM consists of 4 structural interfaces (specialization, exhibition, decomposition, and instantiation) that connect entities with the same persistence to express the time-invariant relationship between them. An aggregation-participation link is an interface between a whole and its parts. Exhibition-characterization link represents an interface between features and characterized entities to represent properties, features or other attributes. Generalization-specialization link is an interface between specializations and the generic part. Classification-instantiation link represents an interface between an abstraction and its instances with the same structure, features and behavior.

Another type of interface is procedural links, which connect processes to objects to express dynamic aspects of the system. A procedural link could be of one of two main types: transforming, enabling, or control link. Transforming link connect object that the processes transform or its state with this particular process, and consist of consumption, result and effect links. A consumption link is a unidirectional interface originating from a consumable object and leading to a consuming process. A result link is a unidirectional interface from a process to a produced object. An effect link is a bi-directional interface identifying process that changes the state of an object.

Enabling links could be decomposed into two groups: agent links and instrument links. We do not specialize agent links as they refer to human or a group of human. In turn, an instrument link is a unidirectional interface that identifies an object that enables the occurrence of a process to which it is linked. Each procedural link may be qualified as state-specified procedural link. A state-specified procedural link is a detailed version of its counterpart to a procedural link in the sense that, instead of connecting a process with an object, it connects a process with a specific state of that object.

5 Integrated view in DSM

To build an integrated view of ontologies we adopted the following approach for defining the types of relationships between ontologies' entities. We start by extracting key entities associated with the OPM, the ES/CPM, and the MPM ontologies. Following the entities' descriptions in the corresponding literature, we list the selected entities in the DSM matrix presented in Fig. 3. For each type of the OPM link set, we assign a designator (see Table 1). The cells at the intersection of different rows and columns indicate the type of relationship between the corresponding entities. DSM should be read from 'column to row'.

It is also important to note that we have mapped the relationships between entities of each ontology, shown in grey zones in Fig. 3. To identify the types of relationships within ES/CPM entities, we use the ES/CPM data structure shown in Fig. 1 to define existing relationships between entities of the ES/CPM ontology. Regarding the classification of relationships, we use the OPM set of links. We consider the persistence of each entity from the OPM perspective, analyzing whether the entity describes only an object (stateful one or stateless one) or a process, or both at the same time.

In the ES/CPM ontology Function, Behavior and Flow entities have the persistence of the Process, as demonstrated in the DSM by the generalization-specialization (GS) link. All of these entities are connected with the State entity through unidirectional effect links, since they all change Artifact's particular situation during its lifetime. Artifact, Assembly and Part (Eng.) entities have the persistence of Object that describes a physical entity in a product, as shown in the DSM by a GS link. All these entities may have a link with a State entity, since at some point of time they can be affected, i.e., their states can change. As there is no special link in OPM notation to demonstrate the relationship between Object and State, these stateful objects can be converted into several stateless objects that will specify the original stateful one, thus they will be linked by a GS link.

For example, Assembly, Part and Feature are specializations of Artifact entities (assemblies and parts in CPM notation are subartifacts), which in OPM ontology corresponds to an aggregation-participation (AP) link, because Assembly consists of Parts (Eng.). A fairly large number of the relationships between entities in the ES/CPM are of the exhibition-characterization link, since the main focus of ES/CPM ontology is on describing the key characteristics of Artifact information stored in PLM systems. Thus, Form and Requirement describe and characterize an Artifact (or assembly or part depending on specialization). In turn, entity Form can be decomposed into Geometry,

Material and Tolerances attributes, that means that these entities are linked by a GS link.

Table 1. Designators for OPM links.

Link type	Designation
aggregation-participation	AP
exhibition-characterization	EC
generalization-specialization	GS
classification-instantiation	CI
instrument	I
effect	E
result	R
consumption	C
state-specified effect	SE
state-specified result	SR
state-specified consumption	SC

As for the procedural links that connect entities with opposite persistence, they are present between Artifact and Function, since Artifact is an instrument for implementing a Function. These entities are connected through an instrument (I) link. As Behavior describes how Artifact implements its function, these two entities are connected using consumption (C) link, while Artifact and Flow with an E link. Since Function affects product's Tolerance, so the relationship between them is also described by E link.

For the MPM ontology, we also start by defining the persistence of each entity from the OPM perspective. In MPM ontology, Part, Machine Tool, Tools and Facility entities have the persistence of the stateful Object that describes a physical object. This is demonstrated in the DSM by a GS link to both Object and State entity similarly with CPM. The Process Plan has the persistence of a Process. In turn, Manufacturing BOM has the persistence of both a stateful object and a process, as it contains all the Parts (Man.) and Assemblies required to build the product structured based on the assembling process through an aggregation-participation (AP) link. Thus, this entity could be specialized into Object, State and Process with a GS link. As MBOM is one of the approaches to describe the plan of the assembly process, the link between MBOM and Process Plan is GS. This entity also affects the Facilities implemented during assembling process, so an E link connects these two entities.

	OPM			Engineering System/CPM													MPM										
	(1) Object	(2) Process	(3) State	(4) Artifact	(5) Feature	(6) Product Function	(7) Form	(8) Product Behavior	(9) Geometry	(10) Material	(11) Geometric Tolerance	(12) Requirement	(13) Flow in product	(14) Assembly	(15) Part (Eng.)	(16) Manufacturing BOM	(17) Part (Man.)	(18) Process Plan	(19) Machine Tool	(20) Manufacturing Processes	(21) Manufacturing Features	(22) Manufacturing Material	(23) Tools	(24) Manufacturing Dispersions	(25) Facility		
OPM				GS										GS	GS	GS	GS	GS							GS		
(2) Process						GS		GS								GS	GS	GS	GS	GS							
(3) State					SE		SE							SE	GS	GS	GS,SE	GS	SE						GS		
(4) Artifact	GS	GS		AP	EC	R					EC	E		GS	GS	GS	GS	GS		GS							
(5) Feature			AP		EC					EC					AP	AP	AP			GS							
(6) Product Function		GS		I	I																						
(7) Form				EC	EC				GS	GS	GS							EC	E			EC	EC	EC			
(8) Product Behavior		GS																									
(9) Geometry								GS									EC				EC		EC	EC			
(10) Material								GS									EC			E	EC	AP	EC	EC			
(11) Geometric Tolerance					EC	E	GS										EC						EC	EC			
(12) Requirement				EC														E									
(13) Flow in product			GS																								
(14) Assembly		GS	GS	GS										AP	AP									EC			
(15) Part (Eng.)		GS	GS	GS	AP									AP	AP	AP	GS										
(16) Manufacturing BOM		GS	GS	GS	GS	AP								AP	AP	AP	GS										
(17) Part (Man.)		GS	GS	GS	AP		EC	EC	EC	EC				GS	AP	AP	AP			AP	EC		EC	EC			
(18) Process Plan		GS														GS	AP			I	AP	E	I	I			
(19) Machine Tool		GS		GS																				EC	AP	EC	AP
(20) Manufacturing Process		GS								I,E								AP	I		E	I	I				
(21) Manufacturing Feature				GS	EC	EC	EC										AP					EC	EC	EC	EC		
(22) Manufacturing Material													E	EC			EC			EC	EC	EC	EC	EC	EC		
(23) Tool		GS		GS			EC	EC	EC	EC									AP		EC	EC	EC	EC	EC		
(24) Manufacturing dispersion					E	EC	E	EC									EC			EC	E	EC	EC	EC	EC		
(25) Facility	GS	GS												EC	E			AP					EC				

Fig. 3. A DSM-based representation of an integrated view of ontologies.

Using the same logic as for ES/CPM, Part (Man.) and Feature relationship can be described by an AP link according to the OPM set of links. In turn, links between Part (M) with Manufacturing material and Manufacturing dispersion strictly characterize the entity Part (Man.) (through EC link), although in the ES/CPM ontology, similar characteristics are a specialization of the entity Form.

The Process Plan entity, which can be aggregated from Manufacturing Process entities (AP link), is implemented by instruments such as Machine Tool, Manufacturing materials and Tools (I link) and directed to Manufacturing Features (E link).

Manufacturing Material and Manufacturing dispersions characterize Machine Tools (EC link), while Machine Tool components are Tools and Facility, so they are linked by AP. Manufacturing process depends on Manufacturing Features (unidirectional effect link from Manufacturing Features entity to Manufacturing Process entity), influence Manufacturing dispersion (unidirectional link that is directed from Manufacturing Process entity) and uses Manufacturing materials and Tools as instruments, which is represented by an I link.

Manufacturing Feature can be characterized by Manufacturing Materials, Tools and Manufacturing dispersion, so they are linked by EC. In turn, Manufacturing Material can be characterized by Tools, and Tools can be characterized by Manufacturing Discretions and Facility, so the type is EC.

Moreover, we analyzed a similar mapping between ES/CPM and MPM, which represents the data flow between engineering and manufacturing data structures. Artifact entity from the detailed design stage is represented in MBOM, Part, Machine Tool and Tools entities from the manufacturing process definition stage, so the relationship between Artifact and all entities listed above is generalization-specialization. Artifact's feature is specialized into Manufacturing Feature in the MPM ontology through a GS link and characterizes MBOM and Part (Man.) entities through EC link. Form and Geometry entities that describe Part (Man.), Manufacturing Features, Tools and are characterized (C link) by Manufacturing dispersions entity, that is relevant at manufacturing phase. Therefore, Form and Geometry entities are connected with all of the entities listed above through a bidirectional EC link. More precisely, the direction from Form and Geometry to Manufacturing dispersions is opposite to the relationship of Form and Geometry entities with other entities from the MPM ontology. And also Form entity influences Process Plan through E link. The same situation with Product Behavior entity, that affects Manufacturing dispersions with an E link.

Material that is linked to Part (Man.) with a C link, Manufacturing Features and Tools are represented in the essence of Manufacturing Material entity through a GS link and has an I link to Manufacturing process. Geometric tolerances are linked to Part (Man.), Tools and Manufacturing dispersion through a C link.

6 Results and Discussion

As described above, we proposed an approach to support the ontology engineering by defining ontologies' relationships to ensure efficient and concise information flow through the product development process. In our approach, we use fundamental structural and procedural OPM links to ensure comprehension and consistency in knowledge management between modeling ontologies. Procedural links, which are classified into enabling, transformation and event links, connecting processes to the involved objects. To identify fundamental structural rlinks, the OPM ontology provides aggregation, generalization, exhibition and classification links.

Our analysis is based on the defined set of entities derived from modeling ontologies used in the subsequent lifecycle phases of conceptual design, engineering, and manufacturing, respectively being OPM, ES/CPM and MPM. The result is represented in the DSM matrix in Fig. 3. This DSM shows how the various OPM, ES/CPM and MPM entities relate to each other. The proposed DSM methodology demonstrated that it is a powerful tool for simplifying the analysis of the integration of the ontologies and encode their relationships types. The relationships that support the flow of information from the conceptual stage to the physical reality are represented below the diagonal, and the relationships representing the feedback from the physical reality to the digital definition are represented above the diagonal of the DSM matrix.

It is important to note that the DSM is not symmetrical as some types of the links are unidirectional, such as result and instrument links. With regard to structural links, there is no classification-instantiation link, since it will arise only with implementation

of the approach to the exact design case. It is easy to see that the majority of the entities relationships within the ES/CPM are exhibition-characterization or generalization-specialization links, since the main focus of ES/CPM ontology is on describing the key characteristics of Artifact's information stored in PLM systems. In turn, MBOM is the only entity that represents objects and processes simultaneously, having both structural and procedural links, therefore this entity represents the link between product as designed by engineering and as-planned.

7 Conclusion

The DSM approach is widely used in industrial cases, and there exist multiple algorithms for analysis and reorganization of matrix elements as presented by Eppinger and Browning [2]. We foresee this method as a way forward towards using integrated ontologies as concrete implementation tools to increase the clarity of digital systems and define mapping rules and significantly reduce their inconsistency. As this method demonstrates how the ontology of each step of product development is structured with the respect to the domain, it lacks proper attention to interfaces between ontologies to better understand the fundamental principles and current flaws of the digital chain integration.

The proposed approach could support ontology modeling procedures such as represented by Noy and McGuinness [27], which provides an elegant seven-step guidance to ontology development. Ontology development is a dynamic process, starting with an initial rough ontology, which is later revised, refined and filled in with the details. In our case, we consider already existing ontologies and in accordance with ontology development methodology, enumeration of important terms in the ontology was done by defining the representative set of entities for each ontology.

The proposed DSM supports the following steps in defining classes and class hierarchy by defining the relationships of the individual entities that allow ontology visualization. This visualization could represent ontology structure, support evaluating ontologies consistency and complementarity, highlighting redundant entities or missing interconnections.

References

1. Smirnov, A., Shilov, N., Parfenov, N.: Building a multi-aspect ontology for semantic interoperability in PLM. In IFIP 16th International Conference on Product Lifecycle (2019).
2. Eppinger, S.D. and Browning, T.R.: Design structure matrix methods and applications. MIT press, Cambridge (2012).
3. Dori, D.: Object-Process Methodology: A Holistic System Paradigm, Springer, Berlin, <http://dx.doi.org/10.1007/978-3-642-56209-9> (2002).
4. Neches R, Fikes RE, Finin T, Gruber TR, Senator T, Swartout WR.: Enabling technology for knowledge sharing. AI Magazine 12(3):36–56 (1991).
5. Gruber, T. R.: A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition, 5, (2):199-220 (1993).

6. Fenves, S.J.: A Core Product Model For Representing Design Information. National Institute of Standards and Technology, NISTIR 6736, Gaithersburg, MD 20899, USA (2001).
7. Sudarsan, R., Fenves, S.J., Sriram, R.D. and Wang, F.: A product information modeling framework for product lifecycle management, *Computer-Aided Design*, Vol. 37, No. 13, pp.1399–1411 (2005).
8. Fortin, C. and Huet, G.: Manufacturing Process Management: iterative synchronization of engineering data with manufacturing realities, *International Journal of Product Development*, 4(3-4), pp. 280-295 (2007).
9. Object Management Group: Business Process Model and Notation <https://www.omg.org/spec/BPMN/2.0/> (2010) (accessed April 28, 2021).
10. Friedenthal S, Moore A, Steiner R.: A practical guide to SysML: the systems modeling language, Morgan Kaufmann (2014).
11. Lightsey, B.: Systems engineering fundamentals. Defense Acquisition University Press (2001).
12. Shaked, A. and Reich, Y.: Using Domain-Specific Models to Facilitate Model-Based Systems-Engineering: Development Process Design Modeling with OPM and PROVE. *Applied Sciences*, 11(4), p.1532 (2021).
13. Dori, D.: Model-based systems engineering with OPM and SysML. New York: Springer (2016).
14. Huet, G., Fortin, C., McSorley, G., & Toche, B.: The management of manufacturing processes using complementary information structures, *Proceedings of the 18th International Conference on Engineering Design (ICED 2011)*, Vol. 6, pp. 186-192 (2011).
15. Mizoguchi R, Vanwelkenhuysen J, Ikeda M.: Task Ontology for reuse of problem solving knowledge. In: Mars N (ed) *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing (KBKS'95)*. University of Twente, Enschede, The Netherlands. IOS Press, Amsterdam, The Netherlands, pp 46–57 (1995).
16. Bernaras A, Laresgoiti I, Corera J.: Building and reusing ontologies for electrical network applications. In: Wahlster W (ed) *European Conference on Artificial Intelligence (ECAI'96)*. Budapest, Hungary. John Wiley and Sons, Chichester, United Kingdom, pp 298–302 (1996).
17. Uschold M, Jasper R.: A Framework for Understanding and Classifying Ontology Applications. In: Benjamins VR (ed) *IJCAI'99 Workshop on Ontology and Problem Solving Methods: Lessons Learned and Future Trends*. Stockholm, Sweden. CEUR Workshop Proceedings 18:11.1–11.12. Amsterdam, The Netherlands (1999) (<http://CEUR-WS.org/Vol-18/>).
18. Fosse, E., Delp, C.L.: Systems engineering interfaces: A model based approach, *2013 IEEE Aerosp. Conf.* 1–8 (2013), doi:10.1109/AERO.2013.6497322
19. Hirshorn, S. R., Voss L. D., Bromley L.K. *NASA Systems Engineering Handbook*, Technical Report, NASA/SP-2016-6105 Rev 2. HQ-E-DAA-TN38707 (2017).
20. Rahmani, K., Thomson, V.: Ontology based interface design and control methodology for collaborative product development, *Comput. Des.* 44, 432–444 (2012) doi:10.1016/j.cad.2011.12.002
21. Mikkola, J.H. Modularity and interface management of product architectures”, *International Conference on Management of Engineering and Technology, PICMET'01*. Portland, pp. 599–609 (2001).
22. Lindemann, U., Maurer, M. and Braun, T.: *Structural complexity management: an approach for the field of product design*. Springer Science & Business Media (2008)
23. Hicketier, G., Tommelein, I.D., Hofmann, M., Lostuvali, B. and Gehbauer, F.: MDM as a Tool to Improve BIM Development Processes. In *Proceedings of the 13th International DSM Conference*, pp. 349-362 (2011)

24. Hollauer, C., Wilberg, J. and Omer, M.: A matrix-based framework to support dynamic Modeling of sociotechnical systems. In Proceedings of the 17th International DSM Conference, pp. 19-28, Fort Worth, Texas, USA (2015).
25. ISO 19450 (2015) 'Automation systems and integration - Object-Process Methodology'.
26. Crawley, E. and Colson, J.: The projection relationship between object process models (OPM) and design system matrices (DSM). In DSM 2007: Proceedings of the 9th International DSM Conference, Munich, Germany (2007).
27. Noy, N.F., McGuinness, D.L.: Ontology development 101: A guide to creating your first ontology. Knowledge Systems Laboratory Tech report KSL-01-05. Stanford University, Palo Alto (2001).