# Integration of MBSE and PLM: complexity and uncertainty

Yaroslav Menshenin, Carolina Moreno, Yana Brovar, Clement Fortin

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

# Integration of MBSE and PLM: complexity and uncertainty

Yaroslav Menshenin*, Carolina Moreno,
Yana Brovar and Clement Fortin

Skolkovo Institute of Science and Technology, Space Center
121205, Moscow, Russian Federation
Bolshoy Boulevard 30, bld. 1
Email: y.menshenin@skoltech.ru
Email: carolina.moreno@skolkovotech.ru
Email: yana.brovar@skoltech.ru
Email: c.fortin@skoltech.ru
* Corresponding Author

**Abstract:** Currently, MBSE and PLM methods and solutions are not well aligned with each other resulting in excessive complexity and uncertainty. Their full-scale integration would facilitate the development of complex systems. Better data flow from conceptual design to detailed design and feedback from later stages of product development are needed. In this paper, we analyse the MBSE and PLM integration from a system of systems (SoS) perspective and apply some of the methods used in systems engineering to better understand their nature, quantify their epistemic uncertainty and propose possible solutions to reduce their complexity and uncertainty. To achieve these goals, we study systems ontologies, such as the object-process methodology (OPM), the core product model (CPM), and manufacturing process management (MPM), which represent essential elements of a digital engineering solution. We also propose a measure of complexity to better quantify the structure of the interfaces through the design structure matrix (DSM)-based approach.

**Keywords:** complexity; product design; model-based systems engineering; MBSE; product lifecycle management; PLM; ontology; uncertainty, product lifecycle management.

**Reference** to this paper should be made as follows: Menshenin, Y., Moreno, C., Brovar, Y. and Fortin, C. (2021) 'Integration of MBSE and PLM: complexity and uncertainty', *Int. J. Product Lifecycle Management*, Vol. 13, No. 1, pp.66–88.

**Biographical notes:**

Yaroslav Menshenin is a Research Scientist at the Space Center, Skolkovo Institute of Science and Technology – Skoltech (Moscow, Russia). He holds a PhD (2020) from Skoltech and a Specialist Degree (MSc equivalent) (2012) from the National University of Science and Technology "MISIS" (Moscow, Russia). He is also a graduate of the Singularity University located at NASA Ames Research Center (California, USA). Dr. Menshenin is a Member of INCOSE, AIAA, IFIP WG 5.1, and the DESIGN Society. He was also a visiting doctoral candidate at the System Architecture Group, MIT (2016-2017).

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

Carolina Moreno is a doctoral candidate in Engineering Systems at Skoltech, where she also earned a master's degree in Information Science and Space Systems. In 2017, Carolina held a visiting scholar position at the Harvard-MIT Division of Health Sciences and Technology. She has contributed to research funded by Airbus, working on methods for estimating future performance of technologies under development and on evaluating to what degree roadmap information can be used for budgeting decisions. Currently, she conducts research on methods and tools to facilitate the early assessment of SoS in conceptual design for new technology insertion.

Yana Brovar is a Third year PhD student in the Space Center at Skolkovo Institute of Science and Technology (Skoltech), a private graduate research institute in Moscow, Russia. She received a Specialist's degree (equivalent to a Master's Degree) (2018) in Power Engineering from the Bauman Moscow State Technical University (Moscow, Russia).

Clement Fortin is Professor and Associate Provost for International Affairs at Skoltech. He has extensively worked in the field of Product Lifecycle Management (PLM), particularly in computer aided geometric tolerancing, integration of design and manufacturing, assembly process planning, product development methodologies and high-performance machining and more recently in Model Based Systems Engineering (MBSE). He obtained a Master's degree in Aerospace Engineering at the University of Toronto, Institute for Aerospace Studies, and a PhD in Mechanical Engineering from Queen's University in Canada. Dr. Fortin is a Fellow of the Canadian Academy of Engineering. In June 2015, he was awarded the status of Professor Emeritus for his exceptional contributions to Polytechnique Montréal.

## 1 Introduction

Many years of research on the implementation of major Digital Transformation projects have shown that less than 30% of them are successful (De la Boutetière *et al.*, 2018). This low success rate is largely due to the great complexity of the information systems involved. In the manufacturing sector, current industrial practice recognises that PLM systems have reached a level of extreme complexity that make their implementation and future evolution extremely difficult (Chevalier, 2019). Moreover, as internet and cloud-based information technologies are more and more integrated into all types of products, there is a need to develop them from a Cyber-Physical Human (CPH) systems perspective as indicated in the Industry 4.0 initiative, which requires the integration of a Systems Engineering approach supported by Model-Based Systems Engineering (MBSE) with Product Lifecycle Management (PLM) methodologies and tools. The resultant Integrated System, which is called a Digital Engineering platform by the International Council for Systems Engineering (INCOSE) (Giachetti *et al.*, 2018) is still in its early development, and proper methodologies must be developed to decrease its complexity and thus its uncertainty.

To properly understand the main reason for the uncertainty in the Digital Engineering domain, which is deeply embedded into the core of current PLM systems, one must look at the genesis of these systems and their evolution. PLM systems have evolved from CAD

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

systems that have themselves been developed to replace paper drawings which come from the 1st Industrial Revolution in the middle of the 19th century (Chandrasegaran *et al.*, 2013). The product definition supported by 2D drawings has been greatly standardised for more than a century and is still the legal support for the vast majority of modern mechanical and aerospace products. These 2D drawings have been well proven to properly define the mechanical product itself, including its Product Manufacturing Information (PMI), and coupled with Product Development Management (PDM) tools for configuration management, have served us so well for their original purpose.

PLM systems are thus today very powerful modelling and analysis tools for very complex mechanical systems of all types where the spatial representation plays a vital role. As presented by Mukhachev and Fortin (2020), drawings and CAD solid models carry essentially no definition of time and are thus stateless objects which are not able to properly represent CPH systems developed today for Industry 4.0 environments (Michels, 2018). Therefore, they are inappropriate because they cannot properly represent systems, including their software and electronics components and subsystems, which are very much time-dependent and rely on proper time management capabilities for their representations. Therefore, they have a fundamental epistemic uncertainty for systems representation which increases their complexity unnecessarily. This is why we propose to build new Digital Engineering systems for Industry 4.0 based on a fundamental ontology which is perfectly aligned with the nature of CPH systems.

MBSE systems are much more recent and were developed to offer digital support for document-based Systems Engineering (SE) processes developed in the second part of the 20th century in space and aviation industries. The SE approach was fully implemented for the Apollo program a little more than 50 years ago and has since been further developed for a number of fields supported by its Vee model (NASA Systems Engineering Handbook, 2016; INCOSE Systems Engineering Handbook, 2016). MBSE has been developed to model the essential elements of systems including system behaviour and structure, functions and requirements. The most common approach is based on SysML (Friedenthal *et al.*, 2014), which has been derived from UML and therefore has a very strong information system component. It is currently widely used in some industrial sectors but does not offer a well-structured ontology linking its nine types of diagrams. However, SysML is capable of representing very complex CPH systems and has been implemented in several software environments. There is currently ongoing work to further develop this modelling language under the SysML 2.0 initiative to align it closer with the Digital Engineering vision. Another approach for MBSE is the one proposed by Dori with OPM (Dori, 2002) which is based at its genesis on CPH systems representations and offers a formal ontology and language to support any System of Systems (SoS) development. The SoS concept is extensively used in SE and perfectly corresponds to the MBSE, PLM, Industry 4.0 and Digital Engineering environments.

There is a huge demand for such work, as Digital Engineering aims to supports the definition of well-structured Digital Twins, which are embodiments of both the systemic view and the product view, including its manufacturing and its maintenance, as defined by Grieves and Vickers (2017). Thus, the Digital Twin is a set of virtual information constructs that fully describes all the information necessary to define, describe and produce a potential or actual physical manufactured product, including requirements, fully annotated 3D model with geometric dimensioning and tolerancing (GD&T), material specifications, manufacturing processes, etc. To support the flow of virtual data throughout the product development process, more work remains to be done with regards to the ontological integration of MBSE and PLM: from a systemic representation to a detailed product definition (Menshenin *et al.*, 2020) as well as to manufacturing, operations and maintenance with appropriate feedback to design phases.

Thus, the integration of PLM (Stark, 2016) with MBSE (Walden *et al.*, 2015) is still in its infancy, and much remains to be solved to reach efficient Digital Engineering methodologies and tools which will result from the integration of these two domains.

## 2 Methodological Approach

As mentioned above, in this research work, we propose to look at some of the fundamental reasons which have led to this extreme level of complexity and therefore, to the significant uncertainty present in our current PLM, MBSE and Digital Engineering environments. We examine the hypothesis that this is due to some fundamental epistemic uncertainty, and propose three possible approaches and analysis tools to overcome these difficulties in the implementation of Digital Engineering solutions for the development of complex CPH systems (Sowe *et al.*, 2016).

The first approach proposes the adoption of the Object-Process Methodology (OPM) fundamental ontology to model all types of CPH systems; its simplicity and comprehensiveness can contribute to reducing complexity and clearly define objects, processes and systems states, thus reducing uncertainty.

The second approach proposes the Design Structure Matrix (DSM) tool as a powerful approach to model and integrate the various components of a Digital Engineering platform represented by its various ontologies. We thus propose to use ontologies to integrate the various information systems in a complex digital thread and represent their various relationships within the DSM in order to visualise and analyse an Integrated System of any complexity. Our goal here is not to propose a unique ontology of Digital Engineering but to define a methodology able to simplify the integration of various systems of any nature and size.

Our third approach and contribution is to propose an uncertainty dashboard to help implementation teams to quantify and compare various Digital Engineering projects. The dashboard proposes a number of variables to compare projects and quantifies the complexity of any system by analysing the eigenvalues of the DSM matrix representation as proposed by Sinha and de Weck (2013).

These three approaches make complex integrated systems components and their relationships visible. By quantifying complexity, risks and other important variables, we propose to significantly reduce the epistemic uncertainty of complex PLM, MBSE and Digital Engineering implementation projects.

To achieve these, we are analysing the uncertainty in a product/system development digital environment, which is really a SoS and particularly the MBSE/PLM integrated environment. Therefore, we apply an SE approach and tools to the integration of MBSE and PLM systems themselves.

To illustrate our general approach, we present a representative ontology of Digital Engineering built on the OPM formal ontology of Object-Process and State entities which can properly represent any CPH system. This representative ontology of Digital Engineering also uses the CPM-based ontology, which includes behaviour and function entities represented in the CPM model developed by NIST, including geometric tolerances and assembly entities represented in the Open Assembly Model (OAM) (Sudarsan, 2005). Thirdly, we also propose to use an MPM ontology which imbeds the manufacturing process definition. Our aim is to present an integrated ontology that demonstrates how our proposed

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

approach could reduce complexity and thus uncertainty of future PLM and Digital Engineering solutions.

# 3 Literature Review

## *3.1 Systems ontologies*

In this subsection, we overview the ontology models that are studied throughout the paper. We have chosen these ontologies as representative ones to demonstrate the meta-models for MBSE and PLM systems. In principle, we argue that any other meta-model can be used following the same principles as discussed in our paper. Thus, the chosen ontologies are representative rather than those that can only be used within the proposed approach.

### *3.1.1 Object-Process Methodology (OPM)*

OPM (Dori, 2002) is based on a solid fundamental knowledge of systems and is now standardised in ISO 19450 (ISO 19450, 2015). Overall, as proposed by Dori, there are three core entities of a system in OPM methodology: objects, processes, and related states. An entity "state" links an entity "object" (space representation) and an entity "process" (time representation). Thus, time is explicit in OPM models, and "objects" can be defined as stateful objects, as described by Dori (2002) and Crawley *et al.* (2015), meaning that they carry both space and time in their essence. OPM also has four structural relationships which connect objects to express static, long-term relationships between them. These relationships are specialisation, exhibition, decomposition, and instantiation. Also, OPM has two procedural relationships which connect processes to objects to express these transformations - transformation link and enabling link.

The OPM ontology is very powerful, as it allows representing the CPH systems simply and elegantly. OPM has been developed to represent specifically CPH systems, as it enables modelling and textual representations of fundamental constructs expressed as objects, processes, and states. OPM provides both graphical (Object-Process Diagram) and linguistic (Object-Process Language) representations that allow documenting and modelling the core information about Digital Twins for complex CPH systems at the conceptual design stage. The idea behind OPM is that the combination of these entities and relationships allows a systems designer to effectively represent a complex system of any nature, its function and behaviour particularly at any stage of the system/product lifecycle (Dori *et al.*, 2019).

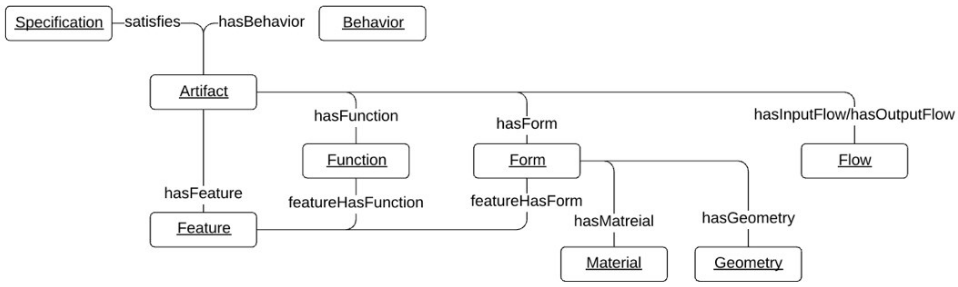
### *3.1.2 Core Product Model (CPM)*

CPM is an abstract model with generic semantics, allowing to describe key characteristics of PLM information, the development of which was driven by the need of next-generation product development systems to manage voluminous and heterogeneous data flows. The main entity in the core model is the artifact that represents a distinct element in a product. In turn, the artifact entity has three main entities representing main characteristics: form, corresponding function and product behaviour. The function entity describes what the artifact is supposed to do based on engineering requirements and stakeholders' needs. In turn, the form entity represents the design solution for function implementation in terms of geometry and material. The behaviour entity describes how the form of an artifact implements its function (Fenves, 2001).

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

The basic data structure of CPM presented in Figure 1 shows all the entities that could be represented in ontology (rounded rectangle) as well as the connections that demonstrate the existence of any kind of relationship (aggregation, association, etc.) between these entities.

**Figure 1** CPM basic data structure




To properly capture assembly and system-level tolerance information and to define kinematic constraints, the CPM framework was extended with the Open Assembly Model (OAM) (Sudarsan, 2005), highlighting information requirements for part features and assembly relationships. The data structure used is part of the Standard for Exchange of Product model data (STEP), based on ISO 10303 (ISO 10303, 1994). The ISO 10303 standard defines a system-independent format for computer-interpretable representation of product data and for its exchange between different CAD systems or between CAD and downstream application systems. By doing this, the basic CPM data structure is complemented by such basic entities as: “kinematic pair”, “assembly”, “assembly feature” and “tolerance”. The “Assembly” entity, decomposed into “parts” and “subassemblies”, incorporates information about assembly relationships and component composition. In turn, “geometric tolerancing” is a critical issue in the design of CPH systems which characterises assembly analysis and controls the variability of linear dimensions. This extension is critical to the product definition at the detailed design stage.

CPM and its several extensions (including OAM) form the NIST Information modelling framework to specify the product design information and knowledge to facilitate semantic interoperability of this product information between CAD/CAE/CAM systems and to capture the evolution of product families. Based on that framework, several attempts were made to complement CPM and create a more applicable engineering system. For example, such ontological approaches as OntoSTEP (Barbau *et al.*, 2012) and ONTO-PDM (Panetto *et al.*, 2012) were developed.

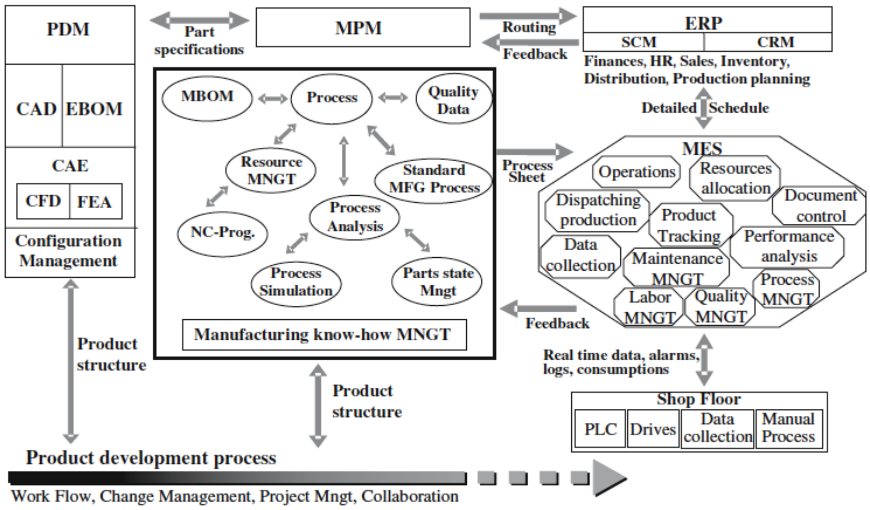
OntoSTEP is a model for translating the STEP schema and its instances into the Ontology Web Language (OWL) for easy integration of geometry representation via STEP with beyond geometry representation via a semantic model to incorporate non-geometrical concepts of function, behaviour and requirements based on CPM/OAM (Fiorentini *et al.*, 2007). The main challenge with OntoSTEP models is that they represent not only semantic information related to the features of the product but also structural data that does not bring any semantics and is used for future implementation. In turn, ONTO-PDM is an approach to facilitate system interoperability in manufacturing environments. Its formalisation is based on the integration and reuse of knowledge embedded in existing standards for product technical data (ISO 10303, 1994) and ERP/MES data (IEC 62264, 2003).

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

*3.1.3 Manufacturing Process Management (MPM)*

MPM has been developed to bridge the worlds of engineering and production by focusing on the manufacturing process definition (Huet *et al.*, 2011). 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 provides 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 synchronisation of design and manufacturing processes definitions through a digital collaborative environment (Fortin and Huet, 2007). In concrete terms, this represents the transition from the 3D geometry to the complete manufacturing process definition, then to production planning, and integration with the Enterprise Resource Planning (ERP), the Manufacturing Resource Planning (MRP) and Manufacturing Execution System (MES), as shown in Figure 2. MPM has been extensively implemented in large industrial organisations globally and is, therefore, a well-proven technology.

**Figure 2** MPM integration within PDM, ERP and MES environments




*Source*: Gagné and Fortin (2007)

*3.2 Design Structure Matrix (DSM)*

To analyse the ontologies of MBSE to PLM data integration, we have used the DSM, which is widely addressed in the SE community. DSM was developed by Steward (1981) and over time, proved its effectiveness as a tool to manage interconnections within a complex system or product (Browning, 2001; Menshenin and Crawley, 2018).

Eppinger and Browning have demonstrated the utility of DSM-based methods not only for product architectures but also for project organisations architectures, and for process architectures (Eppinger and Browning, 2012). Rizzuti and Luigi De Napoli provided a perspective to integrate the DSM and Axiomatic Design (Suh, 1990) in product design (Rizzuti and Luigi De Napoli, 2014). Danilovic and Browning conducted a comparison of the DSM approach and the cross-domain Domain Mapping Matrices (DMM) approach showing their complementary nature and mutual advantages (Danilovic and Browning, 2007; Danilovic and Browning, 2004). DSM has also been extended to the Multiple-Domain Matrix (MDM) (Maurer, 2007; Lindemann *et al.*, 2008). Therefore, DSM can potentially be used for a variety of industries and a variety of applications within these industries. Thus, the DSM-based method could be used as an appropriate tool to determine the relationships between the various systems elements. DSMs have been used to represent and analyse very complex industrial systems (Eppinger and Browning, 2012). DSM has such capabilities that make it a universal approach to not only analyse the architecture, but also the data integration between different models.

A DSM is also effectively an Adjacency Matrix which can be used to calculate interesting system properties (Eppinger and Browning, 2012). As an example, this approach has been used to calculate the structural complexity of a system, as proposed by Sinha and de Weck (2013).

In a DSM, the upper diagonal elements represent the upstream information, which is the feedback from the physical twin to the Digital Twin within the SoS. The elements below the diagonal represent the downstream information flow going from the conceptual design stage to the final product delivery.

### 3.3 *Project Dashboards*

Uncertainty influences all stages of system development, from planning to development to control. A widely used tool for monitoring and tracking this uncertainty is a project management dashboard, which is essentially a data dashboard that provides an overview of project status and displays metrics and insights specific to a particular project.

Dashboards help stakeholders identify correspondence patterns and anomalies and monitor and analyse critical project processes. Such visual tools facilitate routine tasks and simplify the representation of complex data by displaying information quickly, clearly and efficiently.

Several approaches to address uncertainty and complexity using dashboards have been proposed. Loch *et al.* (2000) suggest a framework for evaluating the uncertainty of a project by classifying it into complexity, variation, risk, ambiguity and chaos. Shenhar (2001) proposed a two-dimensional model based on the notion that there are different hierarchies within a product or system. Projects can be classified into four levels of technological uncertainty at the time of project initiation and three levels of complexity depending on the scope of the system. Shenhar and Dvir (2007a) built on this work and proposed the NTCP model; a set of four dimensions to evaluate the novelty, complexity, pace and technological innovation of a project. The diamond-shaped diagram accommodates these four metrics along two axes, and each parameter is evaluated based on expert knowledge. In their work, implementation experts compare the various variables to previous and well-known projects which serve to provide references for the new systems or projects. However, the framework does not specify clear-cut criteria or quantitative measures that might help classify and manage projects. Moreno and Fortin (2020) addressed this gap and proposed an approach to facilitate the early assessment of SoS in conceptual design for new technology insertion by embedding quantitative and qualitative variables into a dashboard. The quantitative variables of complexity and technology integration risks are based on the DSM analysis of complex systems of systems. The qualitative variables are evaluated using historical data from previously implemented similar scenarios. When no reliable data is available, the authors turn to experts to elicit their knowledge on the particular design scenario based on the information available at the moment of the study.

Vasnier *et al.* (2020) suggest using dashboards to establish optimised strategies and identify risk environments for Small and Medium Enterprises (SMEs). The authors identified key features of such visualisation and explored how dashboard design can positively affect users' perception of usefulness and ease of use.

### *3.3.1 Complexity and Uncertainty*

Analyses of MBSE and PLM Integration have shown that one of the fundamental problems associated with integrating MBSE and PLM occurs due to the fundamental essence of systems. This issue is interwoven with the complexity arising from the need to explicitly represent time and space to completely define the system form and behaviour throughout the product/system lifecycle (Menshenin *et al.*, 2020).

The topic of complexity is widely addressed in the literature and is approached from different viewpoints. For example, Suh proposed the theory of complexity based on the semantic theory of information (Suh, 2005). He defines complexity as "a measure of uncertainty in achieving the specified functional requirements". The idea is that complexity is associated with the functional requirement and the information content: the greater the information needed to achieve the functional requirement, the greater the information content (of the functional requirements), leading to greater complexity.

Complex systems have "many interrelated, interconnected, or interwoven elements and interfaces" (Crawley, 2007). Crawley also distinguishes between essential, perceived, and actual complexity. Uncertainty can be measured through the notion of complexity, as these two definitions have been deeply studied in the literature - for Systems of Systems (Crawley *et al.*, 2015; Sinha and de Weck, 2013); for Structural Complexity and its Implications to Design of Cyber-Physical Systems (Sinha, 2014); and for Product Life Cycle Oriented Representation of Uncertainty (Sprenger and Anderl, 2012). Like many other authors, Suh sees a high degree of interconnectivity between the terms "complexity" and "uncertainty" (Suh, 2005).

Uncertainty has also been defined by Wynn *et al.* (2011) as the lack of definition, lack of knowledge and lack of trust in knowledge. This is a very general definition which sheds light on uncertainty in the design process. Uncertainty can be divided into epistemic and aleatory uncertainties. Epistemic uncertainty relates to the lack of knowledge we may have about the system, whether it is modelled or real (Thunnissen, 2003). This definition suggests that the fundamental problem is incomplete and conflicting information of some characteristic of the system. Epistemic uncertainty can thus be decreased by a better definition of the entities and their relationships within each subsystem and between the various subsystems, leading us to better define the interfaces between them.

Complexity is a significant contributor to epistemic uncertainty in Digital Engineering, and it needs to be quantified to properly mitigate it and improve product and complex system development. Complexity is hierarchical in nature, as each system has its own unique set of subsystems (Simon, 1998). Complexity arises from the interactions between these subsystems and systems with interdependent behaviours that interact with one another (Miller and Page, 2008).

Several authors have developed approaches to evaluate the complexity of engineering systems. The most straightforward approach counts the number of components in a system (Bralia, 1986). Braha and Maimon argue that design complexity is a function of its representation, which includes facts, causal relations, and models; and functional design complexity as a function of the probability of successfully achieving functional requirements and constraints (Braha and Maimon, 1998). In his work, Suh proposed an entropic measure of function and knowledge, describing real complexity as a measure of the uncertainty in meeting the requirements, and imaginary complexity as the uncertainty that arises due to a lack of knowledge of the design (Suh, 1998). El-Haik and Yang evaluate component complexity based on the valuation of information such as design parameters and their correlations and derive mathematical relationships (El-Haik and Yang, 1999). Ameri proposed to evaluate design complexity based on the number of independent/dependent variables and their relations (Ameri *et al.*, 2008). Summers and Shah evaluate the feasibility of a given design by calculating the probability of producing a design by considering the size of the design space (Summers and Shah, 2010). Pahl and Beitz calculate the number of interactions in a system (Pahl and Beitz, 2013).

Of all the proposed methods, we have found that the method developed by Sinha and de Weck (Sinha and de Weck, 2013) is the most applicable to evaluate the structural complexity of a Digital Engineering System. The method relates the architecture of a physical system, its components and their interactions, as presented in Equation 1 below. The structural complexity of a system *C* consists of the complexity of individual components alone *C1*, the complexity of each interaction *C2* and the effect of the internal arrangement of these interfaces, which defines the relationships within the system. This is also known as topological complexity *C3*.

$$C = C_1 + C_2\, C_3 = \sum_{i=1}^{n} \alpha_i \; + \left[\sum_{i=1}^{n} \sum_{j=1}^{n} \beta_{ij}\, A_{ij}\right] \gamma E(A) \qquad (1)$$

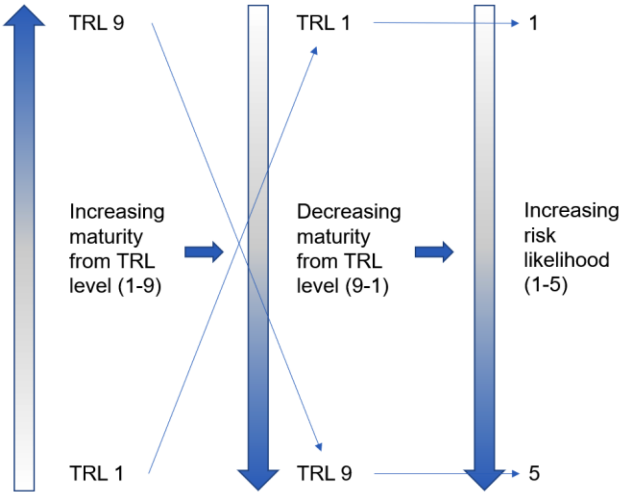
The authors analyse the architecture of any given system, where $\alpha_i$, which makes up *C1,* represents the individual complexities of the components; the interface complexities $\beta_{ij}$ that depend on the complexities of each pair-wise interfacing components and a factor assigned to the type of interface, which are part of *C2*; and the topological complexity *C3* that corresponds to the energy of the graph or energy of the matrix *E(A)* and a normalisation factor $\gamma$. The graph energy represents the impact of the internal connectivity between the components of the system. It is defined as the sum of absolute values of the eigenvalues of the Adjacency Matrix $A_{ij}$, which represents the complexity of the relationships between the components of the system and is usually a binary matrix. The normalisation factor used to scale the topological complexity is based on known cases and is calculated by the authors as 1 over the number of components in the particular system.

### *3.3.2 Technology Integration Risk*

Technology Integration Risk refers to the potential for integrating technology into a system such that all elements, processes and activities satisfy its intended purpose (technical, functional, performance). Several authors have worked on measuring technology integration of engineering systems. The first efforts focused on assessing the effects of new technologies with respect to existing baselines using Pareto Frontiers (de Weck *et al.*, 2003).

Smaling and de Weck searched for the optimal solution using a component-based design structure matrix (DSM) and fuzzy Pareto frontier for rating optimal solution (Smaling and de Weck, 2007). Suh measures the potential market impact of the technology based on customer value, expressed through utility curves for system technical performance measures (Suh *et al.*, 2010). Recent efforts have seen the use of Technology Readiness Levels (TRLs) to quantify the likelihood of a technological risk of the system components (Garg *et al.*, 2017). Hernandez and Fortin (2018) evaluate the use of N-squared matrices to represent physical or functional interfaces between components of the system, which provide a measure of impact coupled with an inverted [1,5] TRL scale to obtain a likelihood score, as shown in Figure 3.

**Figure 3** TRL to likelihood inverted scale




*Source*: Hernandez and Fortin (2018)

The equation to calculate the Technology integration risk is shown below in Equation 2. The values *L* and *U* correspond to the lower and upper limits in the *A* and B scales. *X* is a certain value in a scale.

$$X_B = \frac{(L_B - U_B)X_A + U_B L_A - L_B U_A}{(L_A - U_A)} \tag{2}$$

The total integration risk score is calculated as the sum of the individual scores of all the components for each system. This simple method can be used to quantify the integration risk based on the well-known TRL scale; for the Digital Engineering domain, a new technology could be a new AI module for a PLM system or an advanced IoT module to track operational data from the field.

## 4 Implementation and results

We consider MBSE and PLM ontologies as complex systems themselves and model the corresponding data structures, implying that their entities are system components.

We propose a DSM-based methodology for the analysis of Digital Engineering to support Digital Twin platforms and demonstrate the integration capabilities from the point of view of their respective ontologies. The following levels of integration are chosen as representative systems to illustrate the proposed approach: (1) OPM, which allows a systems designer to effectively represent a complex system of any nature, its function and behaviour particularly at the conceptual design level; (2) Engineering System (ES/CPM), which is a CPM-based ontology supplemented by entities necessary for a proper product definition, such as geometric tolerances and assemblies; and (3) MPM, which represents

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

the transition from the product/system definition to the complete manufacturing process definition, which can be further integrated with Production Planning, Enterprise Resource Planning (ERP) and Manufacturing Execution System (MES) on the shop floor.

The decomposition of the ontologies presented above into logical components or entities and the identified integration interfaces are mapped using a DSM. The word "systems" refers to the ontologies previously presented in section 3 and the word "components" to the critical entities associated with them. We use simplified representations of the systems ontologies, but the approach can be used for Digital Engineering systems of any complexity and comprising thousands of entities.

The DSM shown in Figure 4 reflects the interfaces among representative sets of ontologies' entities. The matrix layout is based on the following reasoning. The names of entities are placed down the side of the matrix as row headings and across the top as column headings in the same order. Interfaces among all components are represented in a binary mode; however, to further represent mapping rules, a numerical DSM could also be used, meaning the specification of the relationships is also possible. When there is a choice of mapping, such specification could make it more concrete, i.e., the type of relationships that exist or the weight of a specific relationship.

**Figure 4** Integrated System - OPM to ES/CPM to MPM

| | | OPM | | | Engineering System / CPM | | | | | | | | | | | MPM | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Object | Process | State | Artifact | Feature | Product function | Form | Product behavior | Geometry | Material (E) | Geometry tolerance | Requirement | Flow in product | Assembly | Part (E) | Manufacturing BOM | Part (M) | Process plan | Machine tool | Manufacturing processes | Manufacturing features | Material (M) | Tools | Manufacturing dispersions | Facility | Technology Readiness Level (TRL) | Number of Interfaces |
| OPM | Object | | 1 | 1 | 1 | | | | | | | | | | 1 | 1 | 1 | 1 | | 1 | | | | 1 | | 1 | 8 | 20 |
| | Process | 1 | | 1 | | | 1 | | 1 | | | | | | | | 1 | | 1 | | | | | | | | | 12 |
| | State | 1 | 1 | | 1 | | 1 | | 1 | | | | | | 1 | 1 | 1 | 1 | 1 | | | | | | | 1 | | 22 |
| Engineering System / CPM | Artifact | 1 | | 1 | | 1 | 1 | 1 | 1 | | | | 1 | 1 | 1 | 1 | 1 | 1 | | 1 | | | | 1 | | | 3 | 28 |
| | Feature | | | | 1 | | 1 | 1 | | | | 1 | | | 1 | 1 | 1 | 1 | | | | 1 | | | | | | 18 |
| | Product function | | 1 | 1 | 1 | 1 | | | | | | 1 | | | | | | | | | | | | | 1 | | | 12 |
| | Form | | | | 1 | 1 | | | | 1 | 1 | 1 | | | | | | 1 | 1 | | | 1 | | 1 | 1 | | | 20 |
| | Product behavior | | 1 | 1 | 1 | | | | | | | | | | | | | | | | | | | | 1 | | | 8 |
| | Geometry | | | | | | | 1 | | | | | | | | | | 1 | | | | 1 | | 1 | 1 | | | 10 |
| | Material (E) | | | | | | | 1 | | | | | | | | | | 1 | | | 1 | 1 | 1 | 1 | | | | 12 |
| | Geometry tolerance | | | | | 1 | 1 | 1 | | | | | | | | | | 1 | | | | | | 1 | 1 | | | 12 |
| | Requirement | | | | 1 | | | | | | | | | | | | | | 1 | | | | | | | | | 4 |
| | Flow in product | | | | 1 | | | | | | | | | | | | | | | | | | 1 | | | | | 4 |
| | Assembly | 1 | | 1 | 1 | 1 | | | | | | | | | | 1 | 1 | | | | | | | | | 1 | | 14 |
| | Part (E) | 1 | | 1 | 1 | 1 | | | | | | | | | 1 | | | 1 | | | | | 1 | | | | | 14 |
| MPM | Manufacturing BOM | 1 | 1 | 1 | 1 | 1 | | | | | | | | | 1 | | | 1 | 1 | | | | | | | 1 | 6 | 18 |
| | Part (M) | 1 | | 1 | 1 | 1 | | 1 | | 1 | 1 | 1 | | | | 1 | 1 | | 1 | | | 1 | 1 | | 1 | | | 28 |
| | Process plan | | 1 | 1 | | | | 1 | | | | | 1 | | | | 1 | 1 | | 1 | 1 | 1 | 1 | 1 | | | | 22 |
| | Machine tool | 1 | | | 1 | | | | | | | | | | | | | | 1 | | 1 | | 1 | 1 | 1 | 1 | | 16 |
| | Manufacturing processes | | | | | | | | | | 1 | | | | | | | | 1 | 1 | | 1 | 1 | 1 | 1 | | | 14 |
| | Manufacturing features | | | | | 1 | | 1 | | 1 | 1 | | | | | | | 1 | 1 | | 1 | | 1 | 1 | 1 | | | 20 |
| | Material (M) | | | | | | | | | | 1 | | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 | | 1 | | | | 18 |
| | Tools | 1 | | | 1 | | | 1 | | 1 | 1 | 1 | | | | | | | 1 | 1 | 1 | 1 | 1 | | 1 | 1 | | 26 |
| | Manufacturing dispersions | | | | | | 1 | 1 | 1 | 1 | | 1 | | | | | | 1 | | 1 | 1 | 1 | | 1 | | | | 20 |
| | Facility | 1 | | 1 | | | | | | | | | | | 1 | | 1 | | | 1 | | | | 1 | | | | 12 |

Mapping within each ontology is performed based on the approach presented below using the example of the CPM ontology. To map entities within CPM, we analyse the interconnections between its key entities using the basic data structure in Figure 1. If there exists a link from one entity i to another entity *j*, then the values of matrix elements *ij* (intersection of column *i* and row *j*) and *ij* are unity. Otherwise, the values of the matrix elements are left empty. For instance, "artifact" and "geometry" entities do not have a direct link, as shown in Figure 1; therefore, the matrix elements "artifact-geometry" and "geometry-artifact" are empty, at the same time the geometry entity is connected to the form entity and therefore the values of matrix elements "form-geometry" and "geometry-form" are unity, as shown in Figure 5.

**Figure 5** DSM representation of CPM data structure

| | | CPM | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Specification | Behaviour | Artifact | Feature | Function | Flow | Form | Geometry | Material |
| CPM | Specification | | | 1 | | | | | | |
| | Behaviour | | | 1 | | | | | | |
| | Artifact | 1 | 1 | | 1 | 1 | 1 | 1 | | |
| | Feature | | | 1 | | 1 | | 1 | | |
| | Function | | | 1 | 1 | | | | | |
| | Flow | | | 1 | | | | | | |
| | Form | | | 1 | 1 | | | | 1 | 1 |
| | Geometry | | | | | | | 1 | | |
| | Material | | | | | | | 1 | | |

The resulting Integrated System is presented in Figure 4 and includes OPM, ES/CPM and MPM as representative systems of a new Digital Engineering system. To map all the entities between different ontologies, we check whether the information contained within each entity is transferred to previous and later stages of the design process, and to which entities of other ontologies are directly related. As a way to represent a Digital Twin that translates digital components coming from upstream into physical realities downstream and vice-versa, we consider bi-directional data exchanges. This DSM should be read "from column to row" as shown for the CPM example above. In the case of an even partial transfer of the information contained in the selected entity into an entity of another ontology, the value of the matrix element is a value of unity, and in the absence, the matrix element is empty. Thus, matrix elements with the values of unity below the grey areas in Figure 4 indicate the presence of a downstream flow of data related to the considered entity from conceptual stage to physical reality. Similarly, matrix elements with the values of unity above grey areas demonstrate the upstream data flow, going from the physical realities to the digital components. For example, the entity "product function" from the ES/CPM ontology, could be transferred upstream onto the OPM ontology by being partially represented by the entity "process" and partially by the entity "state". The entity "product function" in ES/CPM, therefore, corresponds to a combined "process" and a "state" to be fully defined in the OPM ontology. In the DSM, it is presented by a value of unity in the corresponding matrix elements. Furthermore, the downstream flow of the "product

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

function" entity is represented by a value of unity in the corresponding cell with the "manufacturing dispersions" entity. The rest of the cells remain empty, which means there is no link between the "product function" with other entities of the MPM ontology.

Once we have built a synthetic view of the Digital Engineering system through the DSM matrix, we propose to use this representation to analyse the Integrated System with the purpose to help the implementation team of such complex systems and thus reduce the failure rate of such projects.

We thus present a method to assess uncertainty in the integration of MBSE and PLM as an analogy with a technology insertion scenario in complex SoS implementations. The integration is addressed from an ontological perspective with selected systems representing the relationships within and between the Digital Engineering systems.

Since there is no absolute value of uncertainty that exists yet, the implementation team must take a known existing system as a Reference System. This Reference System is defined from a previous project of known complexity, which can benchmark the development of the new Integrated System. Choosing the right reference is of particular importance for mapping variable elements and tracking and controlling change throughout the product lifecycle phases. To illustrate the proposed Digital Engineering methodology, we use the CPM model developed by NIST as a Reference System because it has system-like elements embedded based on a solid modelling ontology, which can also represent a standalone CAD system. An implementation team could also use a well-known system of its own from its past experience.

Our approach places the problem of MBSE and PLM integration into the context with other technology management processes with important variables such as Structural complexity assessment, Technology integration risks, Leap Potential in the market and Pace of the project as defined in section 3.3. These four dimensions are the foundation for building a visual representation of the overall MBSE and PLM integration in an uncertainty dashboard, plotted using the results obtained from a reference "AS-IS" and a desired Digital Engineering system "TO-BE".

The metrics employed to develop the dashboard were briefly introduced in the literature review in section 3.3. Details on the use of each metric and the modifications implemented are discussed below.

### *4.1 Structural complexity*

The metric selected for this particular dimension is the graph energy-based structural complexity proposed by Sinha and de Weck (2013), whose method and equation were reviewed in section 3.3.1 and are based on substantial industrial use-cases. The results of the complexity analysis based on equation 1 are presented in Table 1.

To clarify the terminology between previous work and our current work, we consider the reference and integrated systems as SoS, the analysed ontologies as systems, and the corresponding entities as components. Based on this, we have adjusted the name of $C1$ in the equation from complexity of individual components to complexity of individual systems.

To calculate the structural complexity, we first look at the complexity of the individual systems for term $C1$. In the original metric, the individual complexities of the systems $\alpha i$ are measured on a scale of [0,5] and computed from NASA's TRL definitions. As a surrogate scale, we used the inverted [1,5] TRL introduced in section 3.3.2. We first assign TRLs to each of the systems according to the NASA TRL scale. An example of this is presented in Figure 4 on the right hand side of the DSM matrix, for the Integrated System.

The TRLs are evaluated based on expert knowledge and CPM is assigned a TRL of 3, since it has not been fully implemented in industrial applications yet; this is equivalent to a level 4 in the inverted TRL [1,5] scale, which would correspond to a high risk in the industrial project implementation.

Similarly, the constitutive systems of the Integrated System are assigned TRLs with OPM evaluated at a TRL of 8, which corresponds to level 2 in the inverted TRL scale; MPM evaluated at a TRL of 6, which corresponds to level 3 in the inverted TRL scale; and as specified above, a TRL of 3 for the CPM-based ES.

We then look at the complexity of the interfaces for term *C2*, which in this study, correspond to data integration, therefore are categorised as informational. For this particular design scenario, the complexity of the individual interfaces is assumed to be 1 for both, the Reference and Integrated Systems. These values can be adjusted and assigned to each of the components depending on the level of information available. The number of interfaces is the sum of all the non-zero interfaces inside the DSM.

Finally, the internal arrangement of these interfaces corresponds to the energy of the DSM and a normalisation factor for term *C3*. The normalisation factor is calculated as 1/9 (number of entities or components) for the Reference System. A similar process was done to calculate the complexity of the Integrated System.

A simple comparison of the complexities is made by calculating the relative change with respect to the reference DSM, as shown in Table 1. The estimated structural complexity increases with the integration of the system. We see an increase by a factor of 16 for the total structural complexity. The results show the propagation of change and the increased complexity due to the integration of the new capabilities. This result provides a useful indicator for the implementation team.

**Table 1** Structural complexity comparison

| | CPM | OPM - ES - MPM |
|---|---|---|
| Number of systems | 1 | 3 |
| Number of entities | 9 | 25 |
| Number of interfaces | 20 | 202 |
| Graph energy | 10 | 52 |
| Component complexity | 4 | 9 |
| Interface complexity | 20 | 202 |
| Topological complexity | 1 | 2 |
| Total structural complexity | 25 | 428 |
| Relative change | - | 16 |

*4.2 Technology Integration Risk*

The metric selected for this particular analysis is the Technology Readiness Levels (TRL)-based likelihood and impact analysis proposed by Hernandez and Fortin (2018), which provides a straightforward way to analyse the interfaces related to the integration risks. The method and equation were reviewed in section 3.3.2.

We replaced the N-squared matrices used by the authors with the DSMs previously developed, to account for bi-directional data exchange. To calculate the integration risk, we use the previously assigned TRLs from NASA's definitions, as discussed in 4.1.

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

The number of interfaces is first calculated for each of the entities and summarised to obtain the number of interfaces of each system. The total integration risk score is calculated as the sum of the individual scores of all the systems in each SoS. As seen in Table 2, the estimated integration risk score increases by a factor of 6. The observed scores represent the increased integration risk between the Reference and Integrated Systems.

**Table 2** Technology integration risks comparison

| | Likelihood | | Impact | | Final risk score |
|---|---|---|---|---|---|
| | TRL | Likelihood score [1,5] | N interfaces | Impact score [1,5] | |
| CPM | 3 | 4 | 40 | 23 | 93 |
| OPM | 8 | 2 | 54 | 31 | 47 |
| ES | 3 | 4 | 156 | 90 | 358 |
| MPM | 6 | 3 | 194 | 111 | 278 |

| | CPM | OPM-ES-MPM |
|---|---|---|
| Total risk score | 93 | 683 |
| Relative change | - | 6 |

*4.3 Leap Potential and Pace*

The qualitative dimensions of the Uncertainty Dashboard are based on the definitions proposed by Shenhar and Dvir (2007) in their NTCP approach to compare and manage projects, previously reviewed in section 3.3.

We adjusted the name of the Novelty metric to Leap Potential, which is a concept better understood when referring to a competitive industrial environment but preserved the qualitative assessment scales, as presented in Table 3.

**Table 3** Leap Potential and Pace dimensions

| ***Leap Potential—how new is the product to the market*** | ***Pace—system urgency and available timeframe*** |
|---|---|
| -*Derivative*: Improvement of an existing product<br>-*Platform*: A new generation of existing product line<br>-*Breakthrough*: A new-to-the-world product | -*Regular:* Delays not critical<br>-*Fast-competitive:* Time to market is important for the business<br>-*Time-critical:* Completion time is crucial for success-window of opportunity<br>-*Blitz:* Crisis project- immediate solution is necessary |

*Source*: adapted from Shenhar and Dvir (2007)

The Leap Potential and Pace are assessed by Digital Engineering/PLM systems implementation experts for major systems, such as MBSE and PLM systems to characterise the particular implementation project. For our illustrative example, we chose the level of Derivative for the Leap Potential as presented in Table 1 above, when CPM is considered standalone; in a possible scenario as a Regular development from a Pace perspective in the past.

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

The illustrative Integrated System was evaluated as a Platform project for Leap Potential to represent its extensive refinement of current MBSE and PLM systems once fully implemented. We also chose a project Pace as being Time-critical as current needs for such systems are rather urgent.

### *4.4 Uncertainty Dashboard*

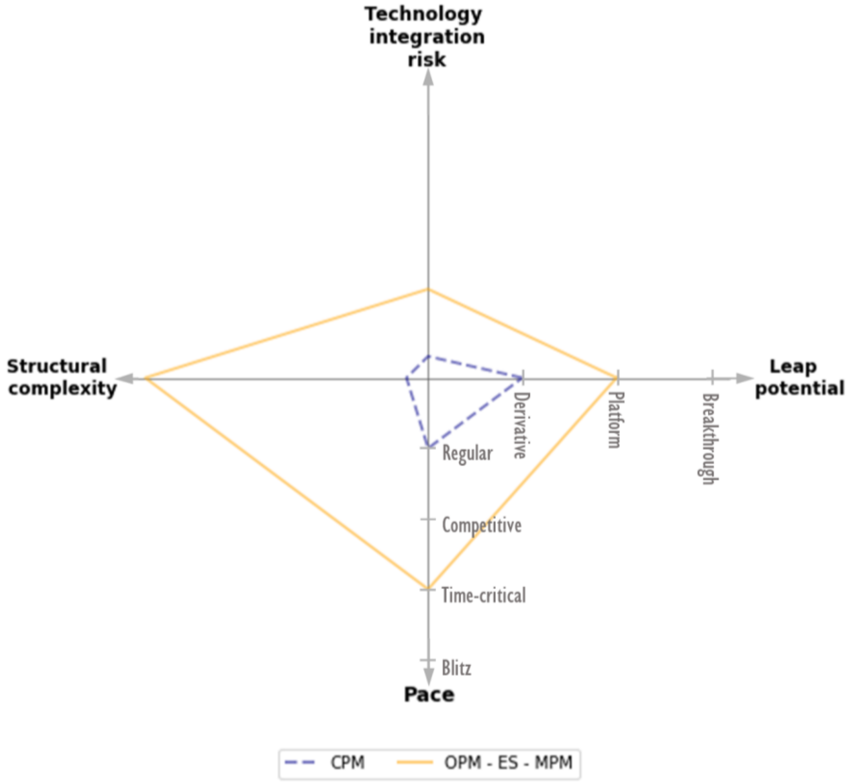
The dashboard shown in Figure 6 is plotted using the results obtained from the reference "AS-IS" and the desired Digital Engineering system "TO-BE". It provides a good visual representation of the MBSE and PLM integration scenario's overall uncertainty as a combination of these four parameters. The results of the quantitative metrics of the Integrated System are normalised to those of the Reference System and plotted accordingly.

The methodology developed and its corresponding dashboard are foreseen as tools that can help an implementation team assess the complexity and technological risks while also taking into account the project pace and the system potential in a competitive market. A concurrent conceptual team could use this dashboard in an interactive mode while the various subsystems or even a complete Digital Engineering System is being defined. The selection of the Reference System is essential as it provides the team with a known benchmark.

The area within the graph on the left-hand side of the diamond can be used to quantify the overall uncertainty of the project as a combination of structural complexity, technology integration risks and project pace. On the other hand, the complete area within the diamond is proportional to the potential benefits of the system since complexity and risks are not fundamentally negative. When managed properly, the structural complexity can be considered as a barrier to entry for competitors, the technology integration risks as a technological advantage and the pace as early market entry.

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

**Figure 6** MBSE and PLM integration Uncertainty dashboard




## 5 Discussion

As described above, three new approaches and tools to support the definition and implementation of Digital Engineering systems have been presented. These approaches and tools are proposed to reduce epistemic uncertainty and complexity in these types of complex systems that aim to support Digital Twin platforms.

We estimate that part of the epistemic uncertainty in current PLM systems comes from the current emphasis of ontologies on basic language constructs such as UML, which are correct in their definitions but are too far from the reality of CPH systems. The result of such approaches based on UML generates excessive complexity and epistemic uncertainty in current MBSE and PLM systems. As a first approach to reduce uncertainty in PLM and Digital Engineering systems, we propose to use OPM, which is a fundamental ontology supporting the core definitions of systems and is particularly well suited to represent CPH systems. In the particular example presented, through the defined set of entities, which are as object, process, and state, the integrated systems based on OPM, ES/CPM and MPM can be easily analysed, as demonstrated in Figure 4. In the DSM, it is easy to see how the various entities of ES/CPM and MPM relate to the OPM entities. One can easily see that the ES/CPM ontology has correspondences with "Process" and "State" entities of OPM, which are absolutely required for proper system behaviour representations. Such an approach is fundamentally universal, leaving room for usage of any other meta-model. For example, we foresee the integration of a SysML 2.0 ontology with OPM as a potential future system level definition. The clarity of the OPM model allows the integrated model to be solidly grounded on a system fundamental reality and the DSM representation of the integrated model to be easily understood.

The proposed DSM methodology, which represents our second proposed approach, demonstrated that it is a powerful method to simplify the analysis of the integration of the various systems and visualise their relationships. As mentioned above, the relationships supporting the flow of information from the conceptual stage to the physical reality are represented below the diagonal. The relationships representing the feedback from the physical reality to the digital definition are represented all above the diagonal of the DSM matrix. The Digital Twin definition with its corresponding entities and relationships can thus be fully represented within the DSM based on the various individual ontologies and their relationships.

The DSM approach for the integration of the various ontologies forming a Digital Engineering system also allows the integration of various types of systems that are complementary and do not necessarily have to be defined from a unique data structure, such as an Engineering Master Model. As an example, the "part" relationships in the MPM system do not have to correspond exactly to those of the ES/CPM system, as they reflect the manufacturing process and plant information which will need to be further integrated with the ERP system for production planning, manufacturing and operations/maintenance system. The definition of the various relationships between the ontologies within the DSM, allows for the inclusion of mapping rules by using numerical values that could correspond to a selected set of relationships. The DSM approach has been used extensively for industrial cases and there exists multiple algorithms to analyse and reorganise the elements of the matrix as presented by Eppinger and Browning (2012). We foresee this method as a way forward to use integrated ontologies as concrete implementation tools to improve the clarity of digital systems and thus reduce significantly their complexity and epistemic uncertainty.

For our third approach, at the level of the implementation of the integrated system, knowledge of the overall system architecture is absolutely critical to be able to quantify and track uncertainty in Digital Engineering systems. There may be systems that are more complex than others, and their respective development team should be able to quantify and track such uncertainty in order to be successful.

Understanding the uncertainty of the design also gives us an idea, whether the design as such is comprehensible for humans. Quantifying uncertainty would give an idea of whether the problems are inherent in the design or somewhere else (i.e. the level of expertise or experience of the developing team).

The DSMs of the systems represented in Figure 4 and Figure 5, show fundamental differences in the density and connectivity of the interfaces, which are attributed to the ontologies in the Integrated System. The individual connections increased in number, thus reflecting in a higher degree of interconnection. The increased graph energy and the topological complexity of the Integrated System presented in Table 2, indicate that the system is more complex and distributed than its Reference System by a factor of 16, which provides an interesting measure for the implementation team and to quantitatively compare various implementations.

The DSMs were created from an abstracted view of the elements to provide the ability to assess the complexity as well as the connectivity across PLM and MBSE systems. To illustrate the proposed Digital Engineering methodology, we use the CPM model as a Reference System. However, we foresee a more complex STEP-based Reference System for more industrial implementations, since nowadays almost every major CAD/CAM system contains a STEP module for managing technical product data. However, the number of entities and their interconnections contained in the STEP AP's is much greater than in CPM and therefore for greater visibility of the proposed approach CPM is a more appropriate ontology. However, a balance is needed in having sufficient detail to perform the required analysis, without making the DSM generation more complicated. The DSM reflects the system level decomposition. The connectivity of the elements is important to map value delivery upstream. A 2-level decomposition may be good enough for comparing engineering systems with similar goals.

PLM systems become increasingly complex and thus uncertain due to growing demands to model and analyse CPH systems within an Industry 4.0 perspective. Systems are hard to design and maintain, and using a systems perspective into their definition can decrease their complexity while improving their usefulness.

The application of the methodology demonstrated the impact of uncertainty propagation in structural complexity estimation. This is manifested as we go from the "AS-IS" to the "TO-BE" systems.

## 6 Conclusion

We have demonstrated the integration path of MBSE and PLM methods and solutions. In particular, we quantified epistemic uncertainty in the integration of these methods. To do this, we applied Systems Engineering approaches and tools, namely, OPM and DSM, to the integration of MBSE and PLM systems themselves. We consider ontologies such as OPM, CPM, and MPM as systems and their entities as systems components.

Throughout our paper the DSM representation has been applied to assess system ontologies and support the implementation of any MBSE, PLM or Digital Engineering approach through an uncertainty dashboard. In summary, our work proposes an elegant yet powerful basic methodology which can reduce the complexity and thus the uncertainty of very complex Digital Engineering environments and improve significantly their chances of successful implementation. We argue that this approach can be used for Digital Engineering systems of any complexity regardless of the number of systems and entities composing the integrated system.

## References

Ameri, F., Summers, J., Mocko, G. and Porter, M. (2008) 'Engineering design complexity: An investigation of methods and measures', *Research in Engineering Design*, 19(2–3), pp. 161–179.

Barbau, R., Krima, S., Rachuri, S., Narayanan, A., Fiorentini, X., Foufou, S. and Sriram, R.D. (2012) 'OntoSTEP: Enriching product model data using ontologies', *Computer-Aided Design*, 44(6), pp. 575-590.

Braha, D. and Maimon, O. (1998) 'The measurement of a design structural and functional complexity', *IEEE Transactions on Systems Man and Cybernetics Part A-Systems and Humans*, 28(4), pp. 527–535.

Bralia, J. (1986) Handbook of product design for manufacturing: A practical guide to low-cost production, New York: McGraw-Hill Book Company.

Browning, T.R. (2001) 'Applying the Design Structure Matrix to System Decomposition and Integration Problems: a Review and New Directions', *IEEE Transactions on Engineering management*, vol, 48, no. 3, pp. 292-306.

Chandrasegaran, S.K., Ramani, K., Sriram, R.D., Horváth, I., Bernard, A., Harik, R.F. and

Gao, W. (2013) 'The evolution, challenges, and future of knowledge representation in product design systems', *Computer-Aided Design*, 45(2), pp. 204-228.

Chevalier, T. (2019) 'Are PLM software soon to be dinosaurs?', *Keynote Speech at the IFIP 18th International PLM conference*, Moscow, Russia. Available Online: https://www.plm-conference.org/wp-content/uploads/2021/02/PLM2019-keynote-Thierry-Chevalier.pdf

Crawley, E., Cameron, B. and Selva, D. (2015) *System architecture: strategy and product development for complex systems*, Prentice Hall Press, USA.

Crawley, E. (2007) https://ocw.mit.edu/courses/engineering-systems-division/esd-34-system-architecture-january-iap-2007/lecture-notes/lec6.pdf (accessed July 30, 2020)

Danilovic, M. and Browning, T.R. (2007) 'Managing complex product development projects with design structure matrices and domain mapping matrices', *International Journal of Project Management*, 25(3), pp. 300-314.

Danilovic, M. and Browning, T. (2004) 'A formal approach for Domain Mapping Matrices (DMM) to complement Design Structure Matrices (DSM)', *Proceedings of the 6th Design Structure Matrix (DSM) International Workshop*, Cambridge, UK.

De la Boutetière, H., Montagner, A. and Reich, A. (2018) 'Unlocking success in digital transformations', *McKinsey & Company*, October, Available Online: https://www.mckinsey.com/business-functions/organization/our-insights/unlocking-success-in-digital-transformations# (accessed July 30, 2020)

de Weck, O., Chang, D., Suzuki, R. and Morikawa, E. (2003) 'Quantitative assessment of technology infusion in communications satellite constellations', *21st Int Communication Satellite System Conference*, Yokohama, Japan.

Dori, D., Kohen, H., Jbara, A., Wengrowicz, N., Lavi, R., Soskin, N.L., Bernstein, K. and Shani, U. (2019) 'OPCloud: An OPM Integrated Conceptual-Executable Modeling Environment for Industry 4.0', *Systems Engineering in the Fourth Industrial Revolution*, Wiley: Hoboken, NJ, USA, pp. 243-271.

Dori, D. (2002) *Object-Process Methodology: A Holistic System Paradigm*, Springer, Berlin, http://dx.doi.org/10.1007/978-3-642-56209-9

El-Haik, B. and Yang, K. (1999) 'The Components of Complexity in Engineering Design," *IIE Transactions*, 31(10), pp. 925–934.

Eppinger, S.D. and Browning, T.R. (2012) *Design structure matrix methods and applications*, MIT press, Cambridge, UK.

Fenves, S.J. (2001) 'A Core Product Model for Representing Design Information', *National Institute of Standards and Technology*, NISTIR 6736, Gaithersburg, MD 20899, USA.

Fiorentini, X., Gambino, I., Liang, V., Rachuri, S., Mahesh, M. and Bock, C. (2007) 'An ontology for assembly representation', National Institute of Standards and Technology, NISTIR 7436. Gaithersburg. MD 20899, USA.

Fortin, C. and Huet, G. (2007) 'Manufacturing Process Management: iterative synchronisation of engineering data with manufacturing realities', *International Journal of Product Development*, 4(3-4), pp. 280-295.

Friedenthal, S., Moore, A., Steiner, R. (2014) *A practical guide to SysML: the systems modeling language*, Morgan Kaufmann, San Francisco, CA, United States.

Gagné, S. and Fortin, C. (2007) 'Application of the CMII model to an integrated engineering and manufacturing development environment', *International Journal on Interactive Design and Manufacturing*, 1(1), pp. 5-13.

Garg, T., Eppinger, S., Joglekar, N. and Olechowski, A. (2017) 'Using TRLs and system architecture to estimate technology integration risk', *In 21st International Conference Engineering Design*, 3, pp. 301–310.

Giachetti *et al.* (2018), Digital Engineering, Systems Engineering Book of Knowledge (SEBoK), https://www.sebokwiki.org/wiki/Digital_Engineering (accessed July 30, 2020)

Grieves M. and Vickers J. (2017) 'Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems', In: Kahlen FJ., Flumerfelt S., Alves A. (eds) Transdisciplinary Perspectives on Complex Systems, Springer, Cham.

Hernandez, A. and Fortin, C. (2018) 'Technological and complexity risk analysis for set based design evaluation', *IEEE International Systems Engineering Symposium ISSE*, pp. 1-5.

Huet, G., Fortin, C., McSorley, G. and Toche, B. (2011) '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.

IEC 62264 (2003) 'Enterprise-control system integration'.

INCOSE (2016) Systems Engineering Handbook.

ISO 19450 (2015) 'Automation systems and integration - Object-Process Methodology'.

ISO 10303 (1994) 'Industrial Automation Systems and Integration - Product Data Representation and Exchange'.

Lindemann, U., Maurer, M. and Braun, T. (2008) *Structural complexity management: an approach for the field of product design*, Springer Science & Business Media, New York.

Loch, C.H., Pich, M.T. and De Meyer, A. (2000) 'Project uncertainty and management styles', Insead R&D, Fontainebleau, France.

Maurer, M. S. (2007) 'Structural awareness in complex product design', *PhD thesis*, Technische Universität München, Munich, Germany.

Menshenin, Y., Knoll, D., Brovar, Y. and Fortin C. (2020) 'Analysis of MBSE/PLM Integration: From Conceptual Design to Detailed Design', *Proceedings of the IFIP 17th International Conference on Product Lifecycle.*

Menshenin, Y. and Crawley, E. (2018) 'DSM-Based Methods to Represent Specialization Relationships in a Concept Framework', *Proceedings of the 20th International DSM Conference*, pp. 151-157.

Michels, J.S. (2018) 'Industrial Connectivity and Industrial Analytics, Core Components of the Factory of the Future', *The Internet of Things*, pp. 247-270, Springer Vieweg, Berlin, Heidelberg.

Miller, J.H. and Page, S.E. (2008) *Complex adaptive systems: an introduction to computational models of social life*, Princeton University Press, Princeton, NJ..

Moreno, C. and Fortin, C. (2020). 'Early Assessment Dashboard of Complex Systems for New Technology Insertion', *IEEE International Systems Engineering Symposium ISSE.*

Mukhachev, P. and Fortin C. (2020) 'A review of behavior and state representation methods in modern PLM systems', *Proceedings of the IFIP 17th International Conference on Product Lifecycle*.

Hirshorn, S. R., Voss L. D., Bromley L.K. (2017) NASA Systems Engineering Handbook, Technical Report, NASA/SP-2016-6105 Rev 2. HQ-E-DAA-TN38707.

Pahl, G. and Beitz, W. (2013) Engineering design: A systematic approach, *Springer Science & Business Media.*

Panetto, H., Dassisti, M. and Tursi, A. (2012) 'ONTO-PDM: Product-driven ONTOlogy for Product Data Management interoperability within manufacturing process environment', *Advanced Engineering Informatics*, 26(2), pp.334-348.

Accepted Version of the paper:
https://doi.org/10.1504/IJPLM.2021.10037270

Rizzuti, S. and De Napoli, L. (2014) 'The integration of DSM and Axiomatic Design in product design as part of a MDM process', *Proceeding of the 16th International DSM conference: Risk and Change management in complex systems*, pp. 35-42.

Shenhar, A. (2001) 'One size does not fit all projects: exploring classical contingency domains', *Journal of Management Science*, 47(3), pp. 394-414.

Shenhar, A. and Dvir, D. (2007) 'How projects differ and what to do about it', *Wiley Guide to Project Program and Portfolio Management*, Hoboken, NJ, Wiley and Sons, pp. 177-198.

Simon, H. (1998) 'The sciences of the artificial', MIT Press, Cambridge.

Sinha, K. (2014) 'Structural complexity and its implications for design of cyber-physical systems', *PhD Thesis*, MIT.

Sinha, K. and de Weck, O. (2013) 'A network-based structural complexity metric for engineered complex systems', *IEEE International Systems Conference SysCon*, pp. 426–430, Orlando, Florida.

Smaling, R. and de Weck, O. (2007) 'Assessing risks and opportunities of technology infusion in system design', *Systems Engineering,* 10(1), pp. 1–25.

Sowe, S.K., Simmon, E., Zettsu, K., de Vaulx, F. and Bojanova, I. (2016) 'Cyber-physical-human systems: Putting people in the loop', *IT professional*, 18(1), pp. 10-13.

Sprenger, A. and Anderl, R. (2012) 'Product life cycle oriented representation of uncertainty', *Proceedings of the IFIP 12th International Conference on Product Lifecycle*, pp. 277-286.

Stark, J. (2016) Product lifecycle management, *Product Lifecycle Management*, Volume 2, pp. 1-35, Springer, Cham.

Steward, D.V. (1981) 'The design structure system: A method for managing the design of complex systems', *IEEE transactions on Engineering Management*, vol. EM-28, no. 3, pp. 71-74.

Sudarsan, R., Fenves, S.J., Sriram, R.D. and Wang, F. (2005) 'A product information modeling framework for product lifecycle management', *Computer-aided design*, 37(13), pp. 1399-1411.

Suh, E.S., Furst, M.R., Mihalyov, K.J. and de Weck, O. (2010) 'Technology infusion for complex systems: A framework and case Study', *Systems Engineering*, 13(2), pp. 186–203.

Suh, N.P. (2005) *Complexity: theory and applications*, Oxford University Press, Inc., 198 Madison Avenue, New York, New York 10016.

Suh, N.P. (1990) *The principles of design*, Oxford University Press, Oxward.

Suh, N.P. (1998) 'Axiomatic design theory for systems', *Research in Engineering Design-Theory Applications and Concurrent Engineering*, 10(4), pp. 189–209.

Summers, J.D. and Shah, J.J. (2010) 'Mechanical engineering design complexity metrics: size, coupling, and solvability', *Journal of Mechanical Design*, 132(2), pp. 1–11.

Thunnissen, D. (2003) 'Uncertainty classification for the design and development of complex systems'*, 3rd Annual Predictive Methods Conference*, Newport Beach, CA.

Vasnier, J.M., Maranzana, N., Messaadia, M. and Aoussat, A. (2020) 'Preliminary design and evaluation of strategic dashboards through the technology acceptance model', *International Design Conference–DESIGN 2020.*

Walden, D.D., Roedler, G.J., Forsberg, K., Hamelin, R.D. and Shortell, T.M. (2015) *Systems engineering handbook: A guide for system life cycle processes and activities*, John Wiley & Sons, Inc., Hoboken, New Jarsey

Wynn, D.C., Grebici, K. and Clarkson, P.J. (2011) 'Modelling the evolution of uncertainty levels during design', *International Journal on Interactive Design and Manufacturing*, 5(3), pp. 187.