MBSE-PLM Integration: Initiatives and Future Outlook

Detlef Gerhard[1], Sophia Salas Cordero[2], Rob Vingerhoeds[2], Brendan P. Sullivan[3](✉), Monica Rossi[3], Yana Brovar[4], Yaroslav Menshenin[4], Clement Fortin[4], and Benoit Eynard[5]

[1] Ruhr-University Bochum, Universitaetsstr. 150, 44801 Bochum, Germany
[2] ISAE-SUPAERO, Université de Toulouse, Toulouse, France
sophia.salas-cordero@isae-supaero.fr
[3] Politecnico di Milano, Via Lambruschini 4B, 20156 Milan, Italy
Brendan.Sullivan@polimi.it
[4] Skolkovo Institute of Science and Technology, Bolshoy Boulevard 30, Bld. 1, Moscow 121205, Russia
[5] Université de Technologie de Compiègne, Rue du Dr Schweitzer, 60203 Compiègne Cedex, France

Abstract. The development of Cyber-Physical-Human Systems which are pervasive today as proposed by the Industry 4.0 vision, requires an efficient integration of the systems definition which is modelled within the Model Based System Engineering (MBSE) applications with the models representing the detailed design and analysis of these products, which are generally embedded in Product Life Cycle Management (PLM) systems. In this paper, we are presenting an overview of some of the important initiatives on this topic across the IFIP 5.1 community, and also projecting a future outlook based on promising new approaches to this emerging problem of MBSE and PLM integration.

Keywords: Systems Engineering · Model based systems engineering · Product life cycle management · Digital engineering · Cyber-physical-human systems

1 Introduction

Systems Engineering (SE) serves the purpose of managing systems complexity and reducing the uncertainty associated with the design process. SE appeared in the mid-twentieth century, when the system's complexity reached extreme levels in the aerospace industry. Since then, SE found its presence and utility not only in very large corporations and defence acquisition programs, but also in automotive, healthcare, energy, and other sectors of the economy. SE itself, as a discipline, has been extended towards different fields of knowledge. However, the increasing complexity of systems has created challenges in the life cycle management of projects, integration of diagrams, retention of knowledge and test planning.

The original version of this chapter was revised: The last name of the author has been corrected as "Salas Cordero". The correction to this chapter is available at https://doi.org/10.1007/978-3-031-25182-5_67

F. Noël et al. (Eds.): PLM 2022, IFIP AICT 667, pp. 165–175, 2023.
https://doi.org/10.1007/978-3-031-25182-5_17

Outlined by IFIP5.1, there is a need for new approaches that "support systems that allow the information and data associated with products to be developed and sustained through the product life-cycle" [1]. This shift in development processes focus utilizes a more complex series of interfaces and necessitates a shift from previous dominant document-based approaches for the facilitation of communication, management of development risks, quality, process, and business productivity, as well as knowledge transfer. In response to this transition this paper presents a series of tools, methods and approaches that are enabled through an integrated model-based approach for the development and management of artifacts and reference models.

Potential advantages offered by MBSE and PLM integration, can include enhanced communications, reduced development risks, improved quality, increased productivity, and enhanced knowledge transfer, can be further scaled up, by increasing the level of automation throughout the system life cycle [2, 3]. In this way, the ability to leverage artifacts for decision making can be extended to larger parts of the development effort. However, both disciplines – MBSE and PLM – have grown to some extent independently, whereas we hypothesize that through better integration more value to our understanding and practice of product/system development is possible.

The paper is structured as follows. Section 2 reviews the trends related to digital engineering. To this purpose, fundamental concepts such as digital twin (DT), the V-Model [8] life cycle diagram and data continuity are reviewed. Section 3 then details the overview of existing initiatives and approaches in support of a MBSE-PLM integration. Section 4 presents a discussion, conclusions, and outlook.

2 Background

MBSE and PLM integration is a topic that keeps on gaining momentum. In the past years, there has been work on how to integrate MBSE and PLM throughout parametric models as in [4], also how to automate trade studies using MBSE and PLM [5], how MBSE and PLM industrial integration is a need for mission-critical systems [6], and incorporation of DT technology into MBSE [7].

2.1 Model Based Systems Engineering

The International Council on Systems Engineering (INCOSE) defines Model Based Systems Engineering (MBSE) as "the formalized application of modelling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle" [9]. The focus on developing, managing, and controlling system models offers the potential to enhance product quality, enhance reuse of the system artifacts, and improve communications throughout all parts of the business and development team. Through the use of descriptive and analytical models that can be applied throughout the life cycle of a system, MBSE can reduce the time and cost to design, integrate and test a system, while simultaneously reducing risk and improving quality.

Through the Systems Modelling Language (SysML) it is possible to model numerous critical aspects required in the SE domain (structure, requirements, behaviour, parametric). SysML, which was developed as an extension to the Unified Modelling Language

(UML), is considered a standard modelling notation adopted in the context of MBSE and utilized in this paper [10, 11]. Additional MBSE solutions and languages such as Capella and the Object Process Methodology (OPM) can be applied/supplemented for SysML, and in the case of OPM even integrated with SysML.

2.2 Digital Engineering

INCOSE coined the term Digital Engineering [12] to characterize MBSE when complemented with simulation technologies. Digital Engineering supports an integrated model-based approach through the utilization of digital methodologies, tools, processes, and digital artifacts. Grieves and Vickers [13] define Digital Engineering as methods and tools to support the design of well-structured DTs, which are embodiments of both the systemic perspective and the product view. The main popularity of the DT paradigm came with the breakthrough of the development of big data analytics, simulation technologies, and Internet of Things. The DT is a set of virtual information constructs that fully describes all of the information required to define, describe, and produce a potential or actual physical manufactured product, including requirements, a fully annotated 3D model with geometric dimensioning and tolerancing (GD&T), material specifications, Product Manufacturing Information (PMI), etc. Rooted in SE, Digital Engineering [14] uses MBSE [15], to model the essential system characteristics including system requirements, structure, functions, and behaviour.

2.3 Product Life Cycle Management

Product Life Cycle Management (PLM) can be considered as a business strategy that focuses on the management of data, information, knowledge, and experience essential to creating and sustaining a product-centric knowledge environment throughout all passes of a system/product (beginning, middle and end of life) [16]. As such an integrated PLM environment enables collaboration between informed decision makers by combining and integrating various stakeholder perspectives of a product throughout its lifecycle. This collaborative information environment can be strengthened through the inclusion product data management (PDM) which ensures that the right information is received at the right time, in the right place through temporal product data evolution control (versioning, revision, bill of materials, etc.).

PLM vendors have gradually added MBSE functionalities into their solutions over the last decade, beginning with the linking of the system design and detailed design phases, however the integration requires substantial improvements to make it efficient. This is in part due to the still early development maturity of MBSE functionalities combined with the specificity of mechanical design functions. One should remember that PLM systems were developed to replace 2D drafting, which has been well standardized more than a century ago and is still used to legally define products in most industries. As an example, explicit time and state representations which effectively link space and time properties are absent in PLM systems but are fundamental for system, software and electronic design and thus are critical MBSE functionalities. Many approaches bring together multi-domain product development, including product requirements engineering, product architecture and system modelling, system simulation management, program planning, systems engineering, risk management and change management.

In addition to industrial PLM solutions, open-source frameworks such as TASTE, DocDokuPLM and Open Services for Lifecycle Collaboration (OSLC) (Sect. 3.2) are capable of integrating many aspects of MBSE and PLM. As an open-source tool-chain for embedded software development TASTE[1] was developed by the European Space Agency "to bring true, formal models-based Engineering into the way we develop space SW" [17], which supports modelling, model analysis, code generation & deployment, debugging & testing, and execution platforms [18]; DocDokuPLM[2], includes document management, product structure management, product configuration to manage alternatives, bill of materials (BOMs), process management change management, and a platform for data visualization and documents (Word, PDF, CAD…).

2.4 MBSE/PLM Integration

According to the basis and objectives of PLM, it can be seen that spanning from the first ideas, feasibility studies, through the actual development, the operation use and in the ultimately to the retirement of the product, different engineering skills come into play. Depending on the system/product type there can be mechanical engineering, electrical engineering, software engineering, etc. involved. Each of these engineering disciplines are "addressed using specific authoring tools, for example electrical/electronic CAD tools [19]. Building upon the expected advantages of MBSE development for the development of mechatronic products, the earlier phases should be able to rely on good MBSE tools, and then a seamless connection to the specific engineering authoring tools.

Ideally, a complete tool chain should cover all necessary engineering disciplines (e.g., mechanical, software, electronics and electrical), with seamless data continuity between tools, to guarantee upstream as well as downstream functionality. Which means that eventual decisions made in a specific authoring tool are not only cascaded down to the next tool, but also back up to the MBSE tool chain to enable to study specific impacts on other parts of the design.

Despite the initiatives mentioned throughout this paper, a number of obstacles hamper MBSE/PLM integration. Firstly, with the exception of tools commercialized by a single vendor, real data continuity is not guaranteed. This poses problems, as in industry, typically, different authoring tools, as well as data management tools are selected separately, and are partly based on historical observations. Moving from one authoring tool to another one, while maintaining a history on what is already in production, is very costly and time-consuming. Secondly, developing proficiency with a new tool represents is an important investment. It should be mentioned that to connect tools between them may require major adaptations to the tools. Additionally, extended enterprise leads to the need to connect tools between different companies. Such issues, if not addressed, can lead to the necessity to re-enter information that was already entered in other tools,

[1] https://gitrepos.estec.esa.int/taste/taste-setup.
[2] https://github.com/docdoku/docdoku-plm.

with the associated risks such as loss of coherency, introducing errors, etc. that could potentially lead to incoherent design decisions.

As a result of the above-mentioned issues, the complete value and potential of MBSE is partially diminished. In the light of absence of data continuity and harmonization, the use of MBSE often remains limited diagram drawing, leaving designers to transfer the necessary information to other tools.

3 Initiatives and Approaches for Life Cycle Collaboration

There are initiatives in a number of countries covering various approaches to support life cycle model collaboration and MBSE/PLM integration. We present an overview of important projects and initiatives which aim to foster collaboration in the field and increase model-based management, control, and analysis.

INCOSE has a number of initiatives advancing SE theory and practice, one of them is the working group "Digital Engineering Information Exchange" (DEIX WG). Its focus is on the digital artifacts (DA). DA is a digital form of information content that a digital engineering ecosystem produces and consumes by generally following the SE life cycle's process areas as defined in ISO 15288 [20]. DA provide "data for alternative views to visualize, communicate, and deliver data, information, and knowledge to stakeholders", in order to make informed and evidence-based decisions.

3.1 Business Process Modelling Notation

Through the efforts of the Object Management Group (OMG) the Business Process Modelling and Notation (BPMN) has been introduced to support the formalization of the Business Process Model (BPM) layer that offers a common unified visual language capable of defining the interactions amongst and between processes and organizations that make systems at the context and/or operational scenario analysis level [21]. Collaboration between BPMN during the concept of operations (CONOPS) phase provides businesses with the comprehensive capability of understanding development and business procedures through a graphical notation and give organizations the ability to communicate these procedures in a standard manner. Furthermore, life cycle model collaboration can enhance the performance, collaboration and business transactions between people and business entities necessary for a product-centric knowledge environment.

Integration of MBSE and BPMN is capable of increasing life cycle capabilities through increased abstraction and automation, ensuring that relevant artifacts, functions, and elements align with stakeholder needs [10, 11]. The technical system model (MBSE) and organizational model (BPM), enable improved complexity management, communication, and process controllability throughout the entire [22]. The automation layer through this abstraction is able to reflect on different BPMN models (Processes/Orchestration, Choreographies, and Collaborations) that connect distinct processes within the organization, facilitating the conversion and coordination of workflows. Through this graphical notation BPMN is capable of providing a comprehensive view to the system or business models in relation to one another.

3.2 Open Services for Lifecycle Collaboration

Through open formal modelling languages and standards, such as Modelica [23], Functional Mock-Up Interface (FMI) [24], Structure and System Parametrization (SSP) [25], etc., a growing interest in open-source tools that allow collaboration and shared access to information appeared. Modelica, as well as Acumen [26] or Bloqqi [27] are widely used, open-source and mature industrial language. The Modelica Association complements its solutions with machine learning frameworks [28] and continuously improve support for the FMI, enabling models exchange between tools.

Another initiative in line with the purpose of INCOSE towards tools integration is OSLC [29]. OSLC consists of specifications designed for different integration scenarios to make vendor-independent heterogeneous tool integration easier. The standard is built on web standards for communication and ontology definitions. The OSLC specifications consist of an OSLC Core specification, and a set of Domain specifications. The OSLC Core specification defines basic concepts and rules for integration methodologies and ensures consistency among different domain specifications. In turn, OSLC domain specification focuses on specific life cycle topics such as requirements management, change management, configuration management, architecture management, etc. [30]. It has also been recently proposed to improve OSLC through the CONEXUS tool, using category theory as a basis for database integration [31].

3.3 System Architecture Definition

Every system or product has an architecture which defines fundamental relationships between and within its elements during the high-level design phase. The system architecture is defined iteratively, in a series of decision-making activities [32] in which stakeholders, system architects, project managers and designers play a crucial and integrative role. Focusing on the upper-left side of the V-Model, the architecture includes feasibility study/concept exploration, concept of operations, system requirements, and high-level design. Nowadays, the system architecture definition tends to be realized through model-based approaches.

System architecture is "the embodiment of concept, the allocation of physical/informatic function to the elements of form, and the definition of relationships among the elements and with the surrounding context" [33]. Concept is a critical entity, as it rationalizes the architecture and maps function to form. Additionally, the presence of relationships amongst elements, and interfaces between elements, can be established according to subsystems, or departments and units within the organization.

3.4 Detailed Domain Specific Digital Engineering

Detailed engineering within different domains (bottom of V-model) requires the consistent use of standards throughout different digitalization fields and levels, to ensure the exchange of information in cross-company based cooperation. Customers and suppliers require digital representations (models) of components and systems, for geometry and product structure. JT (ISO/DIS 14306:2017) and STEP AP242 (ISO 10303–242:2020) are the most advanced standards in this context. As companies move from traditional paper-based workflows to MBSE, STEP AP 242, is the most commonly used file format for CAD interoperability with downstream departments, suppliers, subcontractors, and customers. STEP AP 242 includes PMI such as GD&T, BOM and other meta information, as well as references for part or assembly design and measurement. It comprises all the features of AP203 (ISO 10303–203:2005) and AP214 (ISO 10303–214:2010), and additionally contains features such as semantic 3D PMI, 3D shape quality and 3D design with parametric/geometric constraints. Therefore, it is particularly suited for the exchange of product and assembly structure data with external references to geometry files (regardless of file format). Based on merging of standards from PLM (different STEP Application protocols) and SE (ISO/IEC 15288, ISO/IEC/IEEE 12207, ISO 10303 - STEP AP233 (ISO 10303–233:2012) and new AP243 (ISO 10303–243:2021), an added value proposal for DT standardization is under development, also associated with the Product Lifecycle Management ATLAS Program[3].

3.5 Verification and Validation

The development of increasingly complex products and their functions requires the use of MBSE methods and advanced simulation and testing capabilities for a variety of applications. Simulation-based decision making and release of complex systems in collaborative development scenarios between partners is gaining significant importance in industry. The current trend shows that the utilization of simulations exceeds physical tests for verification and validation purposes due to the cost and general flexibility of the solution. The quality of the simulation results and their traceability throughout the entire development process is essential.

While multiple options exist, OpenModelica[4] was identified as one solution capable of integrated large-scale modelling, simulation, optimization, model-based analysis [34, 35]. In order to achieve interoperability among different behaviour modeling tools, the FMI standard defines the Functional Mock-Up Units (FMU) as a container and an interface to exchange dynamic models or to perform co-simulation or scheduled execution of simulation models. When exchanged, FMU metadata and FMU structural information are exchanged as well. The ProSTEP iViP Simulation data management (Sim PDM) recommendation [36] provides integration guidelines of simulation data in PLM environments and defines communication processes between simulation data management systems (SDM system) and CAE systems. Additionally, ProSTEP iViP "Smart Systems Engineering Behaviour Model Exchange" [37] describes a comprehensive and representative spectrum of the exchange of behaviour models particularly for the use case of joint SE development network between the contracting entity and the supplier. As a complementary development, the Simulation Model Meta Data (SMMD) specification, under development, defines a data file format as a consignment note to FMUs to integrate FMU exchange processes into PLM [38]. The Automated Functional Data FDX recommendation describes a data model and format for the standardized and traceable exchange of functional data and relevant, associated metadata.

---

[3] ATLAS is related also to the new ISO 23247-2021 or IEC 62832:2020 for the specific smart manufacturing field https://plmatlas.com/.

[4] https://github.com/OpenModelica.

4 Discussion, Conclusion and Outlook

The importance of the traceability and reproducibility of models and simulations, as a major task in the development process of increasingly complex products, requires the integration of MBSE and PLM as seamless as possible; since the importance of decisions derived from modelling and simulation is also growing.

The basis for a system architecture definition is set by requirements. Requirements information can be exchanged through the transmission of Requirements Exchange Format (ReqIF)-compliant XML documents at a systems level for general architecture definition and requirements management [39]. Collaboration between partners is enhanced by the benefits of applying such methods across organizational boundaries.

As discussed in Sect. 3.4, core PLM issues with respect to geometry, shape and part definition and digital thread are supported by AP242. It covers the entire development process from the detailed design stage until the end of production development. This well aligned standard integrates with existing solutions for life cycle support, with minimal migration disruption. It can additionally replace paper-based processes with 3D master models since it contains information normally found in technical drawings, according to a semantic PMI. Product data and non-geometric metadata are represented by AP242, as well as simulation. It enables the description of kinematic structures (e.g., connections, articulations, pairings, movements). Links from the kinematic structure to CAD parts can be established by external references.

While the benefits of BPMN and MBSE integration have strong practical value for PLM, it should be acknowledged that there remains at present a comparatively large modelling effort required for MBSE and BPMN. The integrated knowledge and process information facilitated through this integration will allow for a broader and more comprehensive view of all operations/activities, facilitating enhanced decision making. The expected improved integration of diagrams through intelligent operations will be fundamental to changing how businesses go about their entire engineering processes [40].

INCOSE defines some avenues for the future of SE: it is a model-based environment, with an increasing emphasis on AI powered by "large data sets and expert's domain knowledge" [39]. In this way, there will be a further development of systems engineering methods and tools based on established science and mathematics.

In MBSE, data are quantitative and qualitative variables that characterize an artifact (e.g., product or service) to be designed in terms of its specific requirement, constraint, functionality, behaviour, structure, etc. Data acquisition and processing is applicable to many activities in product development, such as interpretation of customer opinions, market information, customer needs and competitive benchmarking, identification of dependencies. Data in the context of data driven engineering is characterized by high volume and variety, which leads to the requirement for special tools and procedures. Data science provides approaches, algorithms, and technologies to gain knowledge, understanding and intelligence based on data. The product life cycle can be looped back, products already produced and sold provide data that can be used to design the next generation of products. The concept of DT, encompasses information, models, and data from all phases of the life cycle, is an important enabler for Data-Driven Engineering.

The next steps for integrating MBSE and PLM in a common digital backbone will certainly be involving DT and its various applications. According to [3] and [4] key enabling technologies are needed to be associated and merged to provide a real and effective solution to support DT. In the classical architecture of DT based on Virtual models, Physical entities, and Services, MBSE and PLM clearly allow the consistent storage, management, and integrity of DT data. Then, one of the biggest issues under process by academics and industrialists is the connection and integration with physical assets and service operations. Some developments with Internet of Things, Cyber-Physical Systems [11], and Servitization are promising but still require scaling before it will be possible to transform research concepts into viable enterprise and business solutions.

References

1. IFIP5.1 - International Federation for Information Processing. Accessed 19 May 2022. http://www.ifip-wg51.org/topics.php
2. Friedenthal, S.: Future directions for product lifecycle management (PLM) and model-based systems engineering (MBSE). Whitepaper (2020). Accessed 18 May 2022
3. Schuler, R., Kaufmann, U., et al.: 10 theses about MBSE and PLM - challenges and benefits of model based engineering (MBE). PLM4MBSE working group position paper (2015)
4. Katzwinkel, T., Löwer, M.: MBSE-integrated parametric working surfaces as part of a PLM design approach. In: Proceedings of the Design Society: International Conference on Engineering Design (2019)
5. Chadzynski, P.Z., et al.: Enhancing automated trade studies using MBSE SysML and PLM. In: INCOSE International Symposium, vol. 28, pp. 1626 1635 (2018)
6. Pavalkis, S.: Towards industrial integration of MBSE into PLM for mission-critical systems. In: INCOSE International Symposium (2016)
7. Madni, A.M., Madni, C.C., Lucero, S.D.: Leveraging digital twin technology in model-based systems engineering. Systems, **7**(1), 7 (2019). https://doi.org/10.3390/systems7010007
8. Fairley, D., et al.: System life cycle process models: V, systems engineering book of knowledge (SEBoK) (2014). https://www.sebokwiki.org/wiki/System_Life_Cycle_Process_Models:_V Accessed 15 Mar. 2022)
9. INCOSE Technical Operations. Systems Engineering Vision 2020. International Council on Systems Engineering, Seattle, WA, INCOSE-TP-2004–004–02 (2007)
10. Cwikła, G., Gwiazda, A., Banas, W., Monica, Z., Foit, K.: Analysis of the possibility of SysML and BPMN application in formal data acquisition system description. In: IOP Conference Series Materials Science and Engineering, vol. 227 (2017)
11. Lu, J., Ma, J., Zheng, X., et al.: Design ontology supporting model-based systems engineering formalisms. IEEE Syst. J. **16**, 5465–5476 (2021)
12. Giachetti, R., et al.: Digital engineering, systems engineering book of knowledge (2018). https://www.sebokwiki.org/wiki/Digital_Engineering. Accessed 15 March 2022
13. Grieves, M., et al.: Digital twin: mitigating unpredictable, undesirable emergent behaviour in complex systems. In: Kahlen, F.J., Flumerfelt, S., Alves, A. (eds.) Transdisciplinary Perspectives on Complex Systems, pp. 85–113. Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-38756-7_4
14. Tsui, R., Davis, D. Sahlin, J.: Digital engineering models of complex systems using model-based systems engineering (MBSE) from enterprise architecture (EA) to systems of systems (SoS) architectures & systems development life cycle (SDLC). In: INCOSE International Symposium, vol. 28, no. 1, pp. 760–776 (2018)

15. Peterson, T.A.: Systems engineering: transforming digital transformation. In: INCOSE International Symposium, vol. 29, no. 1, pp. 434–447. https://onlinelibrary.wiley.com/doi/abs/10.1002/j.2334-5837.2019.00613.x (2019)
16. Terzi, S., Bouras, A., Dutta, D., Kiritsis, D.: Product lifecycle management - From its history to its new role. Int. J. Prod. Lifecycle Manag. **4**, 360 (2010)
17. ESA / ESTEC / TEC-SWT 2020: TASTE (2005). https://taste.tools/. Accessed 07 Aug 2021
18. Perrotin, M.: TASTE: an open-source tool-chain for embedded software development. https://taste.tools/. Accessed 07 Aug 2021
19. Vingerhoeds, R.A., et al.: Educational challenges for cyber-physical systems modelling. In: 12th International Conference on Modelling, Optimization and Simulation (2018)
20. INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, 4th edn. Wiley, INCOSE (2015)
21. OMG: Systems Modeling Language,Ver. 1.6, Object Management Group (2018). https://www.omg.org/spec/SysML/
22. Konrad, C., et al.: Enabling complexity management through merging business process modeling with MBSE. Procedia CIRP, **84**, 451–456 (2019)
23. Modelica Association. Modelica-A Unified Object-Oriented Language for Physical Systems Modeling-Language Specification Version 3.4. https://www.modelica.org/documents/ModelicaSpec34.pdf. Accessed 20 May 2022
24. MODELISAR. Functional Mock-Up Interface, Version 2.0. Interface Specification (2017). https://fmi-standard.org/downloads/. Accessed 20 May 2022
25. Modelica Association. System Structure and Parameterization, Version 1.0. https://ssp-standard.org. Accessed 20 May 2022
26. Zeng, Y., et al.: Modeling electromechanical aspects of cyber-physical systems. J. Softw. Eng. Robot. **7**, 100–119 (2016)
27. Fors, N., et al.: Modular feature-based block diagram programming. In: Proceedings of the 2016 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software, Onward! Amsterdam, The Netherlands, 2–4 November 2016, pp. 57–73 (2016)
28. Tensorflow.org. An end-to-end open source machine learning platform (2019). https:// www.tensorflow.org/. Accessed 05 Apr 2022
29. OCSL: OSLC core specification version 2.0. Open Services for Lifecycle Collaboration (2010)
30. El-khoury, J.: An Introduction to OSLC and Linked Data, Lecture. Stockholm, Sweden (2015). https://open-services.net/resources/video-20151118-jad-ecs/ Accessed Mar 2022
31. CONEXUS, Category Theory for Semantic Data Interoperability: Exchange and Consolidate Data for Agile PLM at Scale (2021)
32. Menshenin Y., et al.: Model-based system architecting and decision-making. In: Handbook of Model-Based Systems Engineering Madni, A.M., Augustine, N., Sievers M. (eds.), pp. 1–42. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-27486-3_17-1
33. Crawley, E., Cameron, B., Selva, D.: System Architecture: Strategy and Product Development for Complex Systems. Prentice Hall Press, Boston (2015)
34. Otter, M.: Modelica Overview, Modelica Association, Presentation (2018). https://modelica.org/education/educational-material/lecture-material/english/ModelicaOverview.pdf/view.html. Accessed March 2022
35. Fritzson, P., et al.: The OpenModelica integrated environment for modeling, simulation, and model-based development. Model. Ident. Control **41**, 241–295 (2020)
36. Prostep ivip/VDA: Simulation Data Management - Integration of Simulation and Computation in a PDM Environment (SimPDM), Version 2.0 (2008)
37. Prostep ivip; SmartSE Recommendation V 2 - Smart Systems Engineering - Simulation Model Exchange, prostep ivip Association. PSI I l. Darmstadt, Germany (2018)

38. OMG 2016 Requirements Interchange Format. https://www.omg.org/spec/ReqIF
39. Geissen, M.: Traceability of simulation tasks – building blocks for simulation-based decision-making. ProductData J. 20–23. (2020) ISSN 1436–0403
40. McDermott, T.A., Blackburn, M.R., Beling, P.A.: Artificial intelligence and future of systems engineering. In: Lawless, W.F., Mittu, R., Sofge, D.A., Shortell, T., McDermott, T.A. (eds.) Systems Engineering and Artificial Intelligence, pp. 47–59. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-77283-3_3