

Towards Model-Based System Engineering for Simulation-Based Design in Product Data Management Systems

Thomas Vosgien, Thomas Nguyen Van, Marija Jankovic, Benoît Eynard, Jean-Claude Bocquet

▶ **To cite this version:**

Thomas Vosgien, Thomas Nguyen Van, Marija Jankovic, Benoît Eynard, Jean-Claude Bocquet. Towards Model-Based System Engineering for Simulation-Based Design in Product Data Management Systems. 9th International Conference on Product Lifecycle Management (PLM), Jul 2012, Montreal, QC, Canada. pp.612-622, 10.1007/978-3-642-35758-9_55 . hal-01526133

HAL Id: hal-01526133
https://inria.hal.science/hal-01526133

Submitted on 22 May 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Towards Model-Based System Engineering for Simulation-Based Design in Product Data Management Systems

Thomas Vosgien[1, 2], Thomas Nguyen Van[2], Marija Jankovic[1],
Benoît Eynard[3], Jean-Claude Bocquet[1]

[1] Ecole Centrale Paris - Industrial Engineering Laboratory, France
{thomas.vosgien,marija.jankovic,jean-claude.bocquet}@ecp.fr
[2] Snecma Villaroche – Safran Group, France
thomas.nguyenvan@snecma.fr
[3] Université de Technologie de Compiègne - Mechanical System Engineering Unit, France
benoit.eynard@utc.fr

Abstract. System integration and simulation are essential phases in design verification and optimization of system capabilities. Currently, different approaches in Systems Engineering (SE) are not entirely taking into account integration and simulation constraints thus complicating the process and enhancing its running time. The target of this paper is to propose a framework bridging the gap between Model Based System Engineering (MBSE) and current Product Lifecycle Management (PLM) and Simulation Lifecycle Management (SLM) functionalities. In this paper we address these gaps between MBSE design concepts and the current PLM capabilities and propose an innovative approach to manage architecture design for simulation. The proposed framework supports the definition of product architectures so as to organize and facilitate simulation and the specification of different types of interfaces enabling the organisation of several product behaviour simulations. An example of this framework based on an industrial case study for the structural analysis of an aero-engine is described and discussed addressing the limits and future developments.

Keywords: Model Based System Engineering, Engineering Data Management, Product Lifecycle Management, Product Architecture Modelling, Design for Simulation

1 Introduction

In the context of large-scale partnerships, developing complex systems (such as aeronautics products) is a collaborative and distributed work involving several domains/disciplines, teams, processes, design environments, tools and modelling formalisms. In that context, engineering data has to be processed and managed in the most consistent way so as to be used by all the partners and through the different ac-

tivities [1, 2]. Due to the increasing complexity of aeronautical products, the System Engineering approach, offering multi-domain, multi-actors and multi-level system characterisation, can significantly contribute to the subsystem consistency insurance within the integration phase. The main objective of the integration phase is to validate global system behaviour through carefully planned and chosen numerical simulations. Depending on the discipline and the type of analysis performed, these numerical simulations require defining specific product architecture models in order to create appropriate simulation models. A major issue for the integrator is to manage these models in order to identify the relevant data set to be used for the simulation and to organize this data set into a new adapted product structure and "engineering environment". Furthermore, integrating product components in complex system design is iterative and often produces large scale intermediate data with heterogeneous formats and complex relationships. The efficient organisation and management of engineering data are therefore a bottleneck of product design performance [3]. Finally, to ensure the continuity of information between working teams, data exchange efficiency, interoperability between systems and the control through an integrated reference framework for product development are required [4]. Model-Based Systems Engineering (MBSE) appears as being the most appropriate approach proposing a consistent, coherent, interoperable, and evolving model of a system throughout its lifecycle [1]. The objective of this paper is precisely to address this issue in the engineering design process proposing an innovative approach to the definition and organisation of product architecture that will ease and organize numerical simulations. We propose to illustrate this approach introducing an industrial case study in aero-engine development. Finally, based on the proposed case, we will discuss the current limitations and remaining technical challenges.

2 Industrial challenges and improvements foreseen

The multi-disciplinary nature of system engineering projects results in large quantities of design data, managed in several tools corresponding to each domain. The ability to describe a system from multiple viewpoints such as different disciplinary domains, life-cycle phases, or levels of detail, fidelity and abstraction is required [5].

Designing, simulating and integrating a complex system also requires reinforcing design-simulation links. The Digital Mock-Up (DMU) is a virtual 3D representation of the product built from an integrated and modular collection of 3D CAD models [6] permitting collaboration and contextual design [7]. Only a few and recent researches highlight the DMU potential for being the backbone of design-simulation loops and to be adapted for domain-specific engineering needs and especially for simulation needs [6, 8]. Nevertheless, DMU is often used to prepare the structural analysis of a whole assembly or to generate a fluid domain for thermal and CFD calculations. Therefore adapting DMUs represents a very time-consuming and tedious effort due to the following gaps that currently impact such usage of the DMU:

- DMU has often inconsistencies regarding the functional description of assemblies resulting from geometric inconsistencies [6].

- Organisation of models in a DMU is defined with respect to the reference frame of the assembly. Hence, there is no geometric constraint between the components and the relative position of components may be subjected to errors. Concerning large systems it is difficult for the integrator to update a set of constraints when there is a modification on components that impacts several geometric constraints [8]. As a result, the interaction areas between components are not captured in most DMUs, while this information is required to represent intrinsically system interfaces.
- There is a need to manage model's complexity reduction essential for numerical simulations, since numerous details present in CAD models drive to prohibitively long and expensive computations [8].

This paper addresses the first two gaps to make the DMU the most adapted input for simulation activities. Therefore we consider the DMU and its usage extensions as the major input for simulation-based design process. The high level objective of this study is to reduce the lead time setting-up and simulating the integrated product with a better integration between design and analysis activities, data and tools. Regarding the industrial issues mentioned above, achieving this objective involves dealing with the following technical challenges:

- Implement in PDM system a **dedicated integrator environment** based on System Engineering concepts that enables to:
  - **Ease the preparation and organisation of numerical simulations** by reducing the time required for finding, acquiring, re-structuring and retrieving required data,
  - Manage **consistently multi-level and multi-domain system representations** and specifications of design artefacts,
  - Provide innovative, coordinated and flexible **engineering methods to exchange specification of components' interactions**.
- Improvement of **PDM information model** for:
  - Supporting interfaces specifications coordination **enabling a functional, structural and behavioural definition**,
  - Ensuring **associativity and traceability between CAD and CAE data**.

3 Technical orientations: why MBSE to support System Integration activities?

MBSE approaches address different system modelling methodologies and frameworks in order to support complex system design [9, 10, 11]. Nowadays, design, integration and verification/validation activities are performed through the use of CAD and CAE tools. In an MBSE approach these tools and related models should populate, interrogate and exploit the system model in order to identify, structure, retrieve, share, disseminate and visualize product engineering data. In Alemani et al [12], authors introduce Model-Based Definition design as a new way of managing engineering and business processes using 3D models as complete sources of information for design activities. The target of this approach is to store most of the

data related to the product in the 3D CAD model, instead of being scattered in multiple forms throughout the PLM database.

Some of the ongoing researches in MBSE address the issue of representing and integrating design models with diverse analysis/simulation models [13, 14, 15] in order to optimise the design-simulation loops that occur in Product Development Process (PDP). Peak et al. [13] introduce the parametric SysML to demonstrate the way to integrate engineering analysis models with system architecture/design models. The Composable Objects (COB) representation is based on object and constraint graph concepts allowing capturing diverse multi-fidelity models and their fine-grained relations. The multi-representation architecture (MRA) [15] is a design/analysis integration methodology based on knowledge patterns that naturally exist in engineering analysis processes and on explicit design-analysis associativity. Peak et al [13] have transformed the MRA patterns and representations into COBs that can be implemented in SysML. Within a MRA context applied on a flap linkages part, authors have demonstrated the usage of parametric SysML and COBs at component level, linking the behavioural parameters of a FEA mechanical model to the related CAD model parameters. Remaining challenges for current MBSE approaches are to manage this CAD-CAE integration at assembly level and establish relations and related constraints at different levels of system decomposition as well as to ensure the continuity of information between multi-domain working teams.

The architecture of a product is defined as the scheme by which the decomposed elements are arranged in chunks (building blocks or modules) in order to organize engineering and integration process [16, 17, 18]. This modularisation introduces challenges in integration phase for interface management and for identifying impacts between sub-systems on the behavioural level. In [19, 20] the author proposes to address the complexity of models integration by using an interface information model with a three-layered architecture, that is, a design layer, a generic behaviour layer, and an application layer [19]. The definition of the interface is predominant in simulation because it describes the overall behaviour of the system to be studied. We define an interface as a real or virtual area of interaction between two system elements. This interaction permits to link two elements to ensure functional, structural and behavioural continuity. According to Sellgren [20], the interactions between design sub-models take place at interfaces, where an interface is a pair of mating faces. Interfaces have characteristic properties that cannot be directly derived from the related mating features. Sellgren highlights the need to rely on a modular model architecture that enables configuration of systems models from a stored library of sub-models and interface models. He proposed a model-based and feature-based interface information model as extension and improvement of PDM data models. Therefore, previous critical literature review confirms the need to work on MBSE enrichment in order to address design/simulation integration. In view to that, enhancement of design models with adequate interface definition is of importance. In this work we propose to enrich PDM information models with the required interfaces data in order to better support integration and simulation activities.

4 Application of MBSE concepts to PLM: Enhancing system architecture definition integrating simulation constraints

Interfaces and associated information are created and known from the creation of components' geometry. The interacting parts are then identified and specified for the assembly. This definition, usually necessary in the digital mock-up phase, is missing and therefore we propose to define necessary information related to the interfaces:

- The **function**: defining the role of the interface within the system
- Application **domain** (mechanical, thermal, etc.),
- Type of **connection** (surface contact, fitting, pivot connection, etc.),
- **Associated technological components** (bolted flange, bearing, etc.),
- The **design intent**: justifying the choice of the type of interface,
- Identification of the **parts and geometric elements that interact,**
- **Assembly constraints** at these interactions (contact, coincidences, plot, etc.),
- **The representation of the interface** if it is a physical interface (CAD model of the bolt-nut assembly).

The target of this proposal is as well to integrate in this definition features supporting the definition of system behaviour:

- The **behavioural modelling parameters** (e.g. interface mesh specifications expressed through the use of CAE mating features),
- The **finite element (FE) representation** of the interface specific to the analysis domain that are derived from the interface design properties (function, technology, design intent) - e.g. a rigid beam node-to-node connection for a bolted flange, combination of a spring and rigid body element for a bearing, etc.,
- The **behavioural parameters of the connection** (e.g. the stiffness/rigidity of a bolted flange),
- The **behaviour at the interface** (degrees of freedom, boundary conditions, etc.).

We propose to couple the interface information model proposed by Sellgren [20] with the use of a system modelling language such as SysML. In SysML, the system architecture is made of sub-models connected by ports that represent energy, data, material or signal flow. We propose to keep this usage but extend it to use ports and connectors to specify these interactions by providing a functional, geometrical, technological and behavioural interface definition enabling to exploit the features present in CAD and CAE models within PDM systems (see **Fig. 1**). The goal is to capture the design intent and support the designer/integrator in designing (typing, positioning and dimensioning the interface) and specifying the interface properties from their creation. The geometrical specification is based on CAD and CAE models and related features. We propose to use these port objects to specify the interaction area between CAD models of system parts by associating these ports to the mating CAD features. A way forward goes clearly through the association and integration of CAE features for specifying the way of assembling simulation models. It is advocated to do it by using:

- **System/Components Blocks**: represent the system constituents. They permit to access to the corresponding design and simulation models and related data sets.
- **Interface blocks**: contain the **functional and technological description** and allow multi-level representation of the interface. They can be linked to a stored library of interface models.
- **Ports:** specify the **area of interaction**. They consist of a **geometrical specification** using features extracted from CAD models (e.g. surface in interaction) as well for interface mesh specification using features from CAE
- **Connectors:** capture the **design interaction properties** and specify the related **behaviour features** of the interface (linked to the technology used). These features permit to specify the behavioural representation and the way to model the connection between FEMs in pre-post tools.

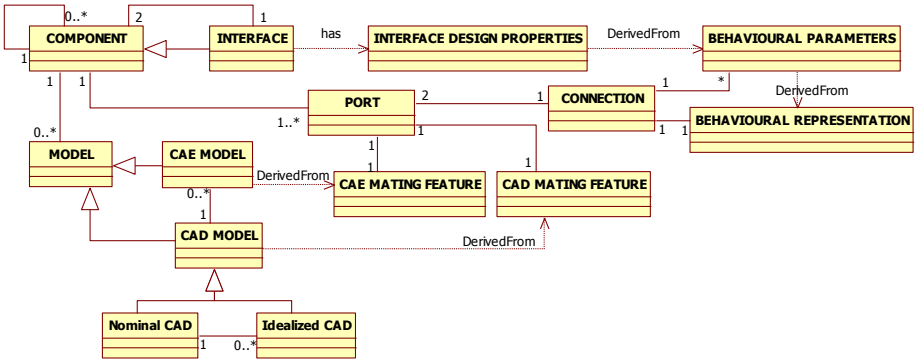


Fig. 1. Proposition of simplified model-based and feature-based interface data model

The proposed enhanced system architecture definition organizes and supports the integration phase. **Fig. 2** illustrates the concepts giving the equivalent meaning of an aero-engine's fan case 3D assembly with the proposed system modelling formalism. This example also illustrates the links between the system architecture objects and related numerical engineering data.

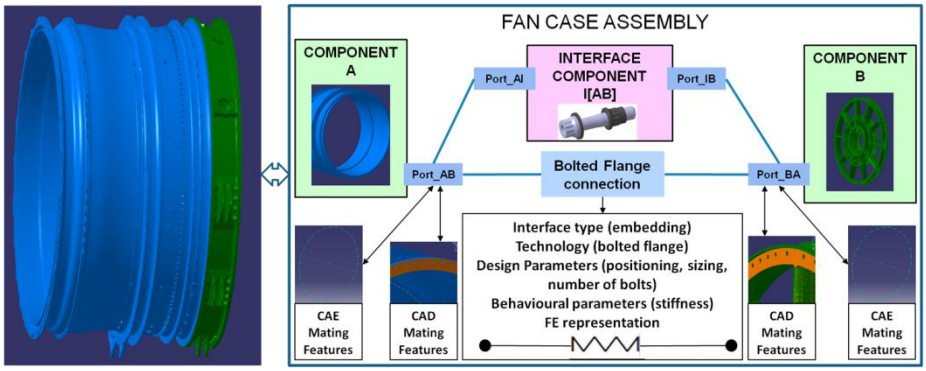


Fig. 2. Links between SE objects and engineering data – Example with a FAN Case assembly

The use of such a framework within PDM system should provide the capability for a multi-level, multi-physics and multi-domain system characterisation. Such an approach can have several benefits for the integration process since it allows to:

- Ensure the consistency with the others representations of product definition (tree structure, 3D physical view, etc.),
- Have **interactions and consistency between the different specific-domain views** of the product,
- **Capture the assembly constraints/links from CAD and CAE models** and related features,
- **Support interface definition and specification** that include multiple levels of hierarchy via ports and ports delegations optimizing the integration/assembly activities,
- **Integrate** or reference **consistently design information** from the product definition (e.g. DMU and Bill of materials) with the associated **simulation data** in order to **ensure** a **complete traceability** of the design/simulation information chain and **ease the management of change impact in PDP**.

5 Illustrated case study of MBSE concepts integrated in a PLM

In order to discuss and test previous proposition we have chosen the Integrated Power Plant System (IPPS). In the case of an assembly under the wing the IPPS consists of the main elements that are the turbojet, the nacelle, and its interface with the aircraft: the pylon (see **Fig. 3**).

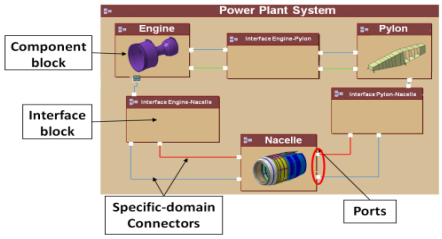


Fig. 3. IPPS system diagram

The design scenario concerns integrating the engine, nacelle and pylon FEM to obtain the mechanical integrated finite element model (IFEM) of the IPPS and performing a structural analysis and feeding back downstream design offices and partners with results (see **Fig. 4**). The main target is to reduce the lead time for setting-up and simulating the integrated product ensuring partner collaboration through:

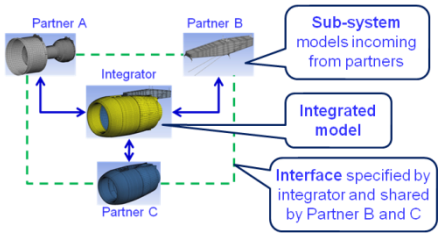


Fig. 4. Case-study scenario overview

- the exchange of appropriate engineering data set between multi-level and multi-domain working teams and partners,
- a complete traceable design-analysis information chain,
- the use of more efficient modelling and simulation capabilities

- the use of automated quality check procedures,
- the use of digital collaborative platform based on a standardized meta-data model enabling to interoperate with various commercial PDM/PLM systems and CAD/CAE authoring applications.

Fig. 5 shows the system modelling framework as seen by the mechanical engine integrator for the engine-pylon assembly. It is important to notice that the mechanical IPPS integrator cannot see the detail of the mechanical engine system description. He can only visualize the engine block and its interfaces with the pylon and the nacelle. He only has access to the 3D CAD models of the engine and nacelle (but they are only BRep models for confidentiality and intellectual property reasons) to specify the interfaces by extracting features (publications in CATIA V6) from these models and create implementation links between these feature and the ports (see **Fig. 6**).

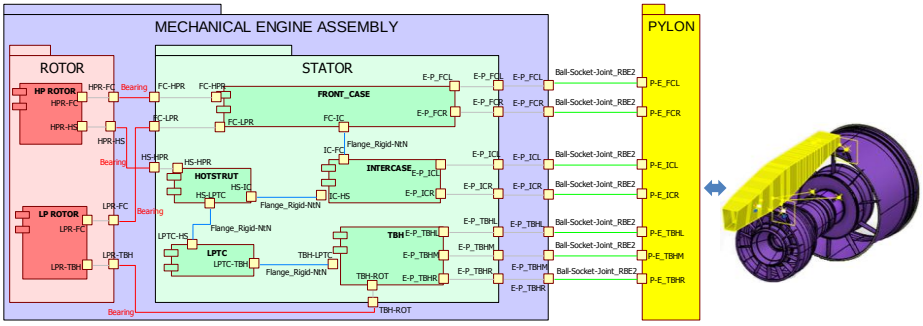


Fig. 5. System Diagram of the Engine-Pylon Mechanical Assembly as seen by the mechanical engine integrator

At engine level (**Fig. 5**), the mechanical engine integrator has a specific mechanical view of the system with a rotor-stator decomposition which is derived from the referential DMU usually modularly decomposed in functional modules (Module 1 for the FAN and Booster assembly, Module 2 for the compressor and the high pressure turbine, Module 3 for the low pressure turbine). The CAE product structure and related data are specific to a discipline and the type of performed simulation.

The mechanical system view is a simplified view of the aero-engine and some of the engine's constituents are missing (e.g. the rotor elements). On this system diagram (Fig. 5) and in the frame of the case-study scenario, the ports specify the CAD mating features that interact for each interface. The interface mesh specifications are also carried by the ports but defined from a library of interface models accessible while defining the connectors' features. This interface mesh specification is generally a mesh file and can be whether an imposed meshed surface or a list of mating nodes (with coordinates and numbering rules). The connectors carry the design interface properties that are derived for simulation in appropriate applicative interface FE representation (e.g. a rigid beam node to node connection for a bolted flange). These FE interface modelling rules are specific to the domain and studies performed and are usually defined by design/simulation method engineers.

This case study is part of a use case scenario carried out under the European FP7 project called CRESCENDO (Collaborative & Robust Engineering using Simulation Capability Enabling Next Design Optimisation). In this use case we are implementing the concepts mentioned above within two heterogeneous environments: CATIA/SIMULIA V6 and Teamcenter 9 (see Fig. 6 below to see the implementation in CATIA/SIMULIA V6).

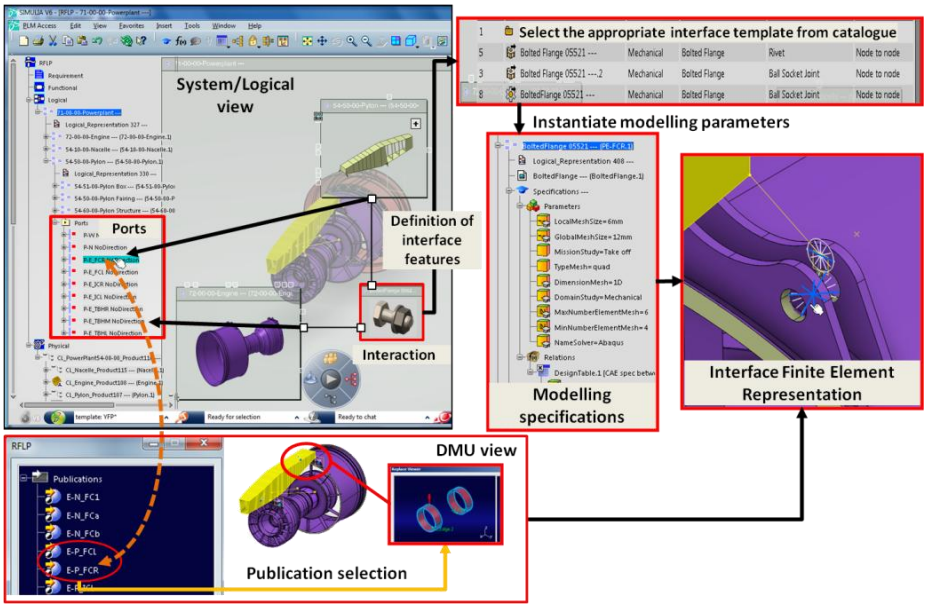


Fig. 6. Implementation of the proposed integrator dedicated environment in CATIA V6 RFLP structure – Specification of a mechanical interface

Regarding these preliminary results, we can list a number of technical limitations in current industrial PLM solutions:

- The links between system objects and the related engineering data present in other product representations are still not automated. It is therefore a tedious work for the user to define manually all the links for a large system as an aero-engine,
- There is still a crucial need to implement standardized data exchange formats in authoring tools and to define aeronautics standards for managing interfaces data to exploit and exchange the CAD/CAE models and related features within heterogeneous PLM environments. Two levels of interoperability are required:
  - The ability to exploit the information available in CAD and CAE models in the PLM/SLM environments,
  - The ability to exchange in a standardized way (model-based) the interface specifications between two heterogeneous PLM/SLM environments.

6 Conclusion

This paper uses a MBSE approach to introduce a system engineering dimension in current PLM/SLM systems. The proposed approach, using an extended SysML formalism and based on an innovative model-based and feature-based interface data model, provides a robust interface modelling capability to designers and integrators. The proposed system framework also allows to have a multi-level (hierarchical), multi-physics and multi-domain system and interfaces characterisation. This approach and these new digital capabilities aim at improving the lead time for setting-up and simulating an integrated product. The approach consists mainly by enriching the engineering knowledge contained in CAD models and enabling to adapt DMU structure and content in order to reduce time and rework in design/analysis modelling and data treatment (acquiring, structuring, analyzing, verification, etc.) activities. We have also illustrated the proposed concepts using an industrial case study: the creation of a mechanical IFEM of an aero-engine; based on a current development performed in collaboration with Dassault Systèmes and Siemens PLM Software. Some of possible future developments are:

- Improving the automation of the CAD and CAE features extraction and the automation to create the links between CAX models' features and system objects.
- Better integration between CAD and CAE data within the implemented solution for traceability purpose.
- Enhance collaboration between partners and interdisciplinary co-designers by enabling interoperability between PLM/SLM systems and various CAD/CAE authoring applications.
- Definition of an aeronautical standard and neutral conceptual meta-model for consistent interface data exchange within aeronautical partnerships.
- Application and demonstrations of the proposed approach and concepts in other disciplines (e.g. Aerothermal analysis and management of fluid interfaces)
- Coupling interdependent system views permitting multi-physics coupling (e.g. thermo-mechanical analysis and thermo-mechanical interfaces management).

Acknowledgements. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 234344 (www.crescendo-fp7.eu/).

References

1. Bajaj, M., Zwemer, D., Peak, R., Phung, A., Scott, A., Wilson, M. (2011) 'Satellites to Supply Chains, Energy to Finance — SLIM for Model-Based Systems Engineering - Part 1: Motivation and Concept of SLIM'. INCOSE International Symposium, 20-23 June 2011, Denver CO, U.S.A.
2. Nguyen Van, T., Féru, T., Guellec, P., Yannou, B. (2006) 'Engineering Data Management for extended enterprise - Context of the European VIVACE Project'. International Conference on Product Lifecycle Management – PLM'06 Bangalore, India.

3. Feng, G., Cui, D., Wangb, C., Yu, J. (2009) 'Integrated data management in complex product collaborative design'. Computers in Industry 60: 48–63.
4. Song, H., Roucoules, L., Eynard, B., Lafon, P. (2006) 'Interoperability between cooperative design modeller and CAD system: software integration versus data exchange'. International Journal for Manufacturing Science & Production 7(2): 139-149.
5. Shah, A., Schaefer, D., Paredis, C. (2009) 'Enabling Multi-View Modeling With SysML Profiles and Model Transformations'. International Conference on Product Lifecycle Management, Bath, England.
6. Foucault, G., Shahwan, A., Léon, J.C., Fine, L. (2011) 'What is the content of a DMU? Analysis and proposal of improvements'. 12ème Colloque National AIP PRIMECA 2011, Le Mont Dore, France.
7. Eynard, B., Yan, X.T. (2008) 'Collaborative product development'. Concurrent Engineering: research and application, 16(1).
8. Shahwan, A., Foucault, G., Leon, J.C., Fine, L. (2011) 'Towards Automated Identication of Functional Designations of Components Based on Geometric Analysis of a DMU'. 12èmes Journées du Groupe de Travail en Modélisation Géométrique (GTMG 2011).
9. Estefan, J.A., (2007) 'Survey of Model-based systems engineering (MBSE) methodologies'. INCOSE MBSE Focus Group.
10. National Defence Industrial Association (NDIA). (2011) 'Final Report of the Model Based Engineering (MBE) Subcommittee'. NDIA Systems Engineering Division M&S Committee.
11. Fridenthal, S., Moore, A., STEINER, R. (2009) 'A pratical guide to SysML'. Burlington: Elsevier.
12. Alemanni, M., Destefanis, F., Vezzetti, E. (2011) 'Model-based definition design in the product lifecycle management scenario'. The International Journal of Advanced Manufacturing Technology, 52:1–14.
13. Peak, R., Burkhart, R., Friedenthal, S., Wilson, M., Bajaj, M., Kim, I. (2007) 'Simulation-Based Design Using SysML—Part 1: A Parametrics Primer'. INCOSE International Symposium, San Diego, USA.
14. Bajaj, M., Zwemer, D., Peak, R., Phung, A., Scott, A.G., Wilson, M. (2011) 'SLIM: collaborative model-based systems engineering workspace for next-generation complex systems'. IEEE Aerospace Conference 2011, Atlanta, GA, USA.
15. Peak, R., Fulton, R., Nishigaki, I., Okamoto, N. (1998) 'Integrating Engineering Design and Analysis Using a Multi- Representation Approach'. Engineering with Computers, Volume 14 No.2, 93-114.
16. Pimmler, U., Eppinger, S. (1994) 'Integration analysis of product decompositions. ASME Design Theory and Methodology Conference Minneapolis, MN, USA.
17. Yassine, A., Braha, D. (2003) 'Complex Concurrent Engineering and the Design Structure Matrix Method'. Concurrent Engineering 2003 11: 165.
18. Sosa, M.E., Eppinger, S.D., Rowles, C.M. (2003) 'Misalignment of Product Architecture and Organizational Structure'. Management Science 50(12), pp. 1674–1689.
19. Sellgren, U. (2006) 'Interface modeling – a modular approach to identify and assess unintended product behavior'. NAFEMS Seminar: Prediction and Modelling of Failure Using FEA. Copenhagen / Roskilde, Denmark.
20. Sellgren, U. (2009) 'The journey towards PLM managed and interface-driven design'. 2nd Nordic Conference on Product Lifecycle Management - NordPLM´09, Göteborg, Sweden.