

35th CIRP Design 2025

# MBfX: Proposing Semantics for systematically applying System Modelling Techniques across different Applications and Domains

Fabian N. Laukotka\*, Markus C. Berschik, Dieter Krause

*Hamburg University of Technology, Institute of Product Development and Mechanical Engineering Design, Denickestrasse 17, 20173 Hamburg, Germany*

\* Corresponding author. Tel.: +49-40-42878-2325. E-mail address: [fabian.laukotka@tuhh.de](mailto:fabian.laukotka@tuhh.de)

## Abstract

System modelling techniques have become increasingly popular in recent years. While many of the approaches to using system models originated in systems engineering, hence Model-Based Systems Engineering, they are now being applied to various applications and domains, even outside the field of systems engineering. However, the term MBSE is still widely used; thus, a diverse understanding has evolved. This work proposes semantics to clearly describe research and approaches when systematically applying these system modelling techniques across different applications and domains of product or service design, to support transparency and comprehensibility.

© 2025 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>)

Peer-review under responsibility of the scientific committee of the 35th CIRP Design 2025

*Keywords:* MBSE; MBfX; System Modelling; Semantics

## 1. Introduction

Due to ongoing megatrends such as digitalization and globalization, the demands on systems sold are increasing. It is not only the increasing number of interconnections between different interacting system parts but also their development that is leading to ever more significant challenges for the processes involved. Today's systems are often more complex, but their development is becoming increasingly comprehensive due to the interaction of different development domains [1]. The resulting challenges can be addressed using model-based approaches. These approaches build on a holistically linked system model as the basis for development, thus providing a basis for developing, analyzing and documenting the product throughout its entire life cycle [2]. The resulting advantages are demonstrated in the literature in various applications. However, it is often unclear how the existing modelling foundations were used or which subject area the various approaches and models consider [3]. For this reason, semantics will be presented below, making it possible to systematically classify model-based approaches in different applications and domains and, thus, make them comparable.

### 1.1. Background

The model-based approaches described above, which emerged primarily from systems engineering, have brought new methods and techniques. The resulting thought patterns and principles, which revolve around a fundamental system model, offer several advantages in everyday development. Because of that, the modelling of systems introduced with Model-Based Systems Engineering (MBSE) is now used in many other applications in different domains within the design community beyond the former focus on SE. This leads to different understandings and sometimes to different terminology [3]. The new definition of SysML v2 is not expected to reduce this fuzziness, but rather lead to an even more significant expansion and diversification of terminology [4].

### 1.2. Research Goal

From this standpoint, the terminology and application areas' fuzziness are major impediments to broader use in the research

community and the industry. Therefore, this paper focuses on the following research question:

RQ: How can the application of system modelling techniques across applications and domains be semantically described to explicitly include applications outside of SE / MBSE?

## 2. Fundamentals

The following section presents fundamentals, from introducing SE and MBSE to modelling aspects. Finally, an excursion to the Design for X approaches is given.

### 2.1. SE and MBSE as the origin of today's System Modelling

Creating and developing products and systems has become increasingly multidisciplinary, requiring new concepts, approaches, and methodologies. Systems Engineering (SE) goes beyond the task of simply engineering systems; it provides a specific set of methodologies for engineering or management tasks and is applied in a variety of industries [5]. Per the definition of the International Council on Systems Engineering (INCOSE), SE “focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem [...]” [2] and, thus, “means to enable the realization of successful systems”. To support the engineers and further assist in handling information, which requires increasingly more effort with larger products or systems, the handling of the occurring (meta-) information, especially semantics or relations, moved from a document-based to model-based documentation. The INCOSE describes this so-called Model-based Systems Engineering 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 phases” [5]. Within this approach, the information is defined, documented and handled in centralized system models, often created using a graphical notation, e.g., with SysML models [6, 7].

This system modelling approach has spread and evolved beyond the original scope of tasks clearly associated with SE. Nowadays, it is also applied to other tasks that face similar challenges regarding data handling [3]. As described in the introduction, this shift to a broader scope likewise came with challenges regarding understanding and terminology.

### 2.2. Aspects of Systems Modelling

Strictly speaking, applying the model-based approach and system modelling techniques to non-SE activities or tasks is no SE and, thus, no MBSE. However, the significant aspects of system modelling can be transferred, as is done on numerous occasions today. Regarding system modelling and the model-based approach Delligatti famously describes three pillars of

modelling - “modelling language, modelling tool and modelling method” [8] (cf. Fig. 1).

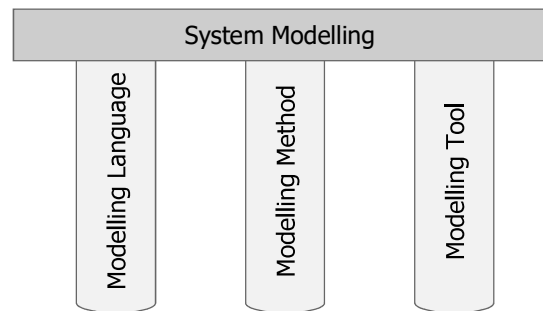


Fig. 1: The "Three Pillars of System Modelling" as described by Delligatti

These pillars are described in more detail in the following three subsections.

#### 2.2.1. Modelling Language

According to Delligatti, the modelling language is “a semiformal language that defines the kinds of elements you’re allowed to put into your model, the allowable relationships between them, and—in the case of a graphical modelling language—the set of notations you can use to display the elements and relationships on diagrams” [8]. Thus, it defines the available modelling elements and syntax and semantics for the modelling. While there are different modelling languages like UML [9] or CONSENS [10], the literature study [3] has shown that as of 2023, SysML (versions up to 1.6 [11]) is the one most often used. The upcoming SysML v2, brings updates and changes to the modelling approach [12], with the addition of a textual notation being one of the major changes [13].

#### 2.2.2. Modelling Method

The second pillar Delligatti describes is the modelling method, a road map consisting of “a documented set of design tasks that a modelling team performs to create a system model” [8]. This modelling method is usually specific and tailored to the task at hand. Delligatti encourages adhering to modelling methods to prevent a wide model variance [8] and supports modelling understanding and compatibility. As a result, various modelling methods have evolved, focusing on multiple applications. Two typical examples of such modelling methods are Weilkien’s System Modelling (SYSMOD) [9] and the INCOSE’s Object-Oriented Systems Engineering Method (OOSEM) [14].

#### 2.2.3. Modelling Tool

Delligatti’s third explicitly named pillar is the modelling tool, or the authoring tool, in which the system model is created. These modelling tools “serve as views of the underlying model. When you modify an element on a diagram within a modelling tool, you’re actually modifying the element itself in the underlying model” [8]. This allows for an immediate synchronization of all views and diagrams within the system model, thus enabling a single source of truth for the information. Some tools include additional features, such as simulation and formal verification of created diagrams to check

against inconsistencies or errors or to generate additional system views [15]. While there are different tools like Enterprise Architect (by Sparx Systems), Rhapsody (by IBM Rational), or Papyrus (free, creator Atos Origin), the authoring tool Cameo Systems Modeler (‘Cameo’, formerly by NoMagic, now ‘CATIA Magic’ by Dassault Systèmes) is one of the most used tools in research for creating system models [3].

### 2.3. Design for X

In product development, various approaches deal with specific aspects of design, production or other product-specific areas. These approaches build on the basic understanding of product development and extend it further. Design for Excellence (DfX) was established to enable a uniform description already in the name. The X can be used as a placeholder and an abbreviation to describe the various sub-areas covered, such as Design for Manufacturing (DfM) or Design for Assembly (DfA) [16]. The term is also used for Design for Variety (DfV) [1].

### 3. Presentation of Proposed Semantics of MBfX

As described above, system modelling and related techniques are nowadays applied to SE tasks as well as various tasks and implementations outside classical systems engineering. However, the essence of the model-based approach remains similar. For this reason, a superordinate classification of approaches as model-based approaches for X, i.e. MBfX, is appropriate and comparable to the convention of Design for X described above. X is deliberately used to open up the field of implementations, in this case, without directly classifying everything that uses system models as MBSE. According to this convention, MBSE can be seen as subordinate to MBfX, with many other possible contexts available. Examples are the already existing approaches of MBPLE (Model-Based Product Line Engineering) [17] or MBPFE (Model-Based Product Family Engineering [18], see also Section 4.2). Likewise, if the term MBSE is used appropriately, the specific approach at hand describes the very realization of MBfX, from which the systems modelling techniques originate.

To improve transparency across communities about specific individual research and applications, and with this openness to contexts and applications other than SE in mind, Delligatti's three pillars can include three more aspects: the resulting system model, the modelling context, and the targeted modelling application. These three additional aspects are described below.

- Resulting System Model

While not explicitly named by Delligatti, the system model itself can be seen as the combining result of the three pillars. It is the central storage of information using different views of the system of interest.

- Modelling Context

While the context of implementations within the scope of SE is usually straightforward, the modelling context needs to be clearly defined when applying system modelling techniques outside that scope. It provides the demand and defines the boundary conditions of the specific scenario. Thus, it can be seen as the base for all other aspects.

- Modelling Application

The modelling application provides the solution or service that faces the demand identified in the modelling context. It is the overall goal of the modelling effort, enabled by the structured approach utilizing the other described aspects.

While not as important when applying to original SE tasks, especially when focusing on non-SE applications, these additional aspects join Delligatti's three pillars and comprehensively describe the specific system modelling implementation. This step of including aspects whose manifestation was obviously predefined within the scope of SE, and therefore did not need to be explicitly considered brings semantics to a more generic level. This allows different application domains to be described while respecting the pre-existing terminology of the original scope. By standardizing the terminology used to describe the aspects of each approach, different implementations can be easily described in a similar way that is easy to understand and compare. This increases the comprehensibility of these often abstract approaches and their descriptions.

Picking up the visualization of the pillars, as shown in Fig. 1, the context depicts the foundation for the three pillars, the system model is the combining result of the three original pillars, and the modelling implementation and application serve as the overall goal at the very top of the other aspects. Together, the pillars can visually be extended to create a generalized structure in the form of the *Visual Modelling Schematic for Comprehensive System Modelling Descriptions (Modelling Temple)* as shown in Fig. 2.

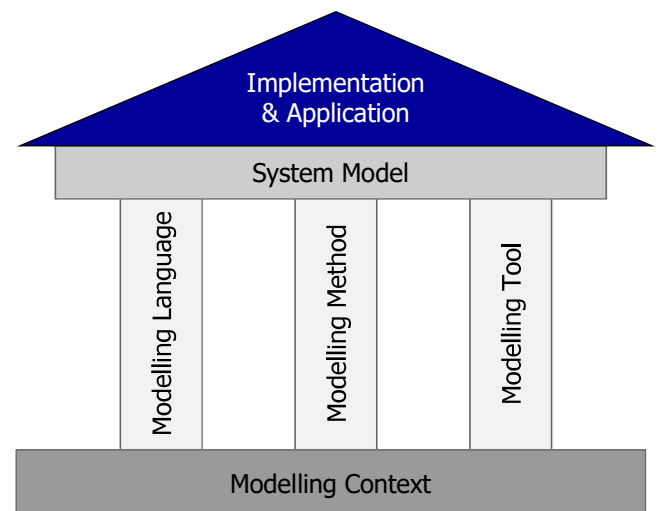


Fig. 2: Visual Modelling Schematic for Comprehensive System Modelling Descriptions (Modelling Temple), based on [19]

When applying this template to a specific context, this generalized structure must be adapted to suit the particular context and scenario in which it will be deployed. This includes a detailed process of selecting and defining the modelling language, method and tools. It should be noted that while SysML provides a solid foundation for a modelling language, it also offers the flexibility to extend its capabilities by defining custom stereotypes, which can then be applied to blocks, relations, and other elements. Similarly, there is a variety of modelling methods available. As these are tailored to a specific use case, it is often appropriate to create and define a modelling method, which might be based on already available ones.

If done and documented thoroughly, this allows for systematic, consistent and structured documentation of information and knowledge and supports a coherent model creation across engineers, teams and projects. This is especially beneficial when targeting comprehensive systems and relations that need to be handled by different engineers over time [19].

#### 4. Application

Over the past years, this template of the modelling schematic has already been internally applied to different implementations. During that process and when applied to a specific scenario and application, the schematic's template is filled with the respective information, thus being extended into the third dimension. Three representative examples of the application of this Modelling Schematic to different system modelling approaches according to the proposed convention of MBFX are described in this section.

##### 4.1. Typical Systems Engineering Tasks

Traditional tasks of SE can be described using the Modelling Schematic template. An example is the functional modelling of a system. Here, the functions of a system are modelled and connected to its physical realization. This can be done using the modelling methods SYSMOD [9] or OOSEM [20]. An exemplary resulting populated modelling temple is shown on the left in Fig. 3.

##### 4.2. Modelling Modular Product Families

In [18] and [21] essential aspects of a modelling approach to establish a model-based Modular Product Family are presented.

The product family is organized into product lines [17] and describes these product lines in more detail under specific aspects. In particular, the internal variance of the various components is modelled. This allows the variety of the product family to be better analyzed and evaluated. In the approach, the physical variety of different components and the variety in behaviour is modelled. Together with its linkage to the customer-relevant properties and the underlying requirements, the representation of the product family is completed. The approach references existing approaches in SE and extends it but is not provided in the INCOSE manual [2] nor the ISO 26580 [17].

Applying the individual realizations of the aspects of system modelling (cf. Chapter 2.2) to the proposed Modelling Schematic (cf. Chapter 3) results in the populated temple as visualized in the centre of Fig. 3.

##### 4.3. Model-Based Data Handling for Aviation's Retrofit

In [19] and [22] the utilization of a model-based approach to support the handling of data and information for the cabin retrofit of aircraft is described. System modelling techniques are used to digitally define occurring semantic relations of elements within an aircraft cabin, within the occurring documentation, and between the elements and their respective documentation.

This application lies outside of SE; no direct product development process is targeted here. However, the system modelling techniques are applied to allow for a digital representation and definition of the occurring instances and relations. Benefits such as hierarchical modelling, single source of truth management, and digital processability are leveraged and ultimately allow for the implementation of an innovative access tool. Applying the individual realizations of the aspects of system modelling (cf. Chapter 2.2) to the proposed modelling schematic (cf. Chapter 3) results in the populated temple as visualized in the right part of Fig. 3.

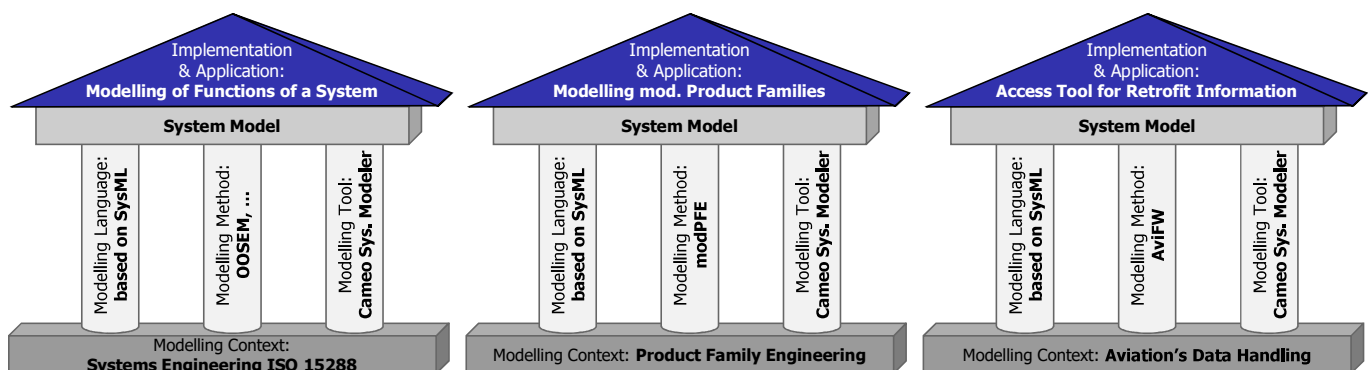


Fig. 3 Example Modelling Templates for the application to the context of: Systems Engineering (the 'original' MBSE) (left) Product Family Engineering (centre) and a model-based support for Aviation's Data Handling (right)

## 5. Summary, discussion, and outlook

In this paper, a semantic description is presented based on the three already-known pillars of systems modelling, as described by Delligatti. Additional aspects have been added to provide a comprehensive description of applications outside the typical systems engineering tasks, as found in various research publications today. Thus, the research question formulated in Section 1.2 was answered by this proposed semantics of MBfX, similar to the principal idea of DfX. This concept, notably the Modelling Schematic, has been presented in some form during previous presentations at scientific conferences or meetings with industry experts. The audience has received it well, and it has been identified as an easily feasible and good way to improve communication across domains by referring to unified aspects. Some of these example applications of semantics have been presented in Section 4, demonstrating in different ways the generic approach that can also be used to describe the specific implementations of systems modelling techniques in different domains, including the original MBSE.

The visual style of the *Modelling Schematic for Comprehensive System Modelling Descriptions* is intended to be an easily understandable format. However, it is not required to be used in precisely this form. The underlying aim is to make the community and possible model creators aware of the current ambiguous wording as well as present a solution that allows clearer descriptions. This is the foundation for more profound discussions in the future and when presenting research and industrial applications.

## Acknowledgements

Parts of this work are part of the research projects InDiCaT (contract number 20D1902C), VerDiKa (contract number 20D1923E) and DISKUS (contract number 20K2202), funded by the Federal Ministry for Economic Affairs and Climate Action as part of the Federal Aeronautical Research Programs LuFo VI-1 and LuFo VI-3. The authors would like to thank the involved parties for their funding and support.

## References

- [1] Krause D, Gebhardt N. *Methodische Entwicklung modularer Produktfamilien*. Springer Berlin. Berlin; 2018. DOI: 10.1007/978-3-662-53040-5
- [2] Walden DD, Roedler GJ, Forsberg K, Hamelin RD, Shortell TM. *Systems Engineering Handbook - A guide for system life cycle processes and activities* INCOSE-TP-2003-002-04. New York; 2015.
- [3] Berschik MC, Schumacher T, Laukotka FN, Inkermann D, Krause D. MBSE within the engineering design community - an exploratory study. 24th International Conference on Engineering Design (ICED 24). Bordeaux; 2023. DOI: 10.1017/pds.2023.26
- [4] Williams M, Boing. *A Roadmap for MBSE Data Standards - Annual INCOSE International Workshop 2021*; 2021.
- [5] INCOSE. *INCOSE Systems Engineering Vision 2020*; 2007. available at: [https://sdincose.org/wp-content/uploads/2011/12/SEVision2020\\_20071003\\_v2\\_03.pdf](https://sdincose.org/wp-content/uploads/2011/12/SEVision2020_20071003_v2_03.pdf).
- [6] Williams M, Boing. *MBSE Data Standards - Annual INCOSE International Workshop 2019*; 2019. available at: [https://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:smswg:smswg\\_19:incose\\_timlm\\_mwilliams\\_iw2019.pdf](https://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:smswg:smswg_19:incose_timlm_mwilliams_iw2019.pdf)
- [7] NoMagic Inc. *Magic Draw - Model-based Systems Engineering*. Online-Brochure; 2011.
- [8] Delligatti L. *SysML distilled: A brief guide to the systems modeling language*. Addison-Wesley. New York; 2014.
- [9] Weikiens T. *Systems Engineering with SysML / UML*. Morgan Kaufmann OMG Press; 2008.
- [10] Gausemeier J, Dumitrescu R, Tschirner C, Stille KS. *Modellbasierte Konzipierung eines hybriden Energiespeichersystems für ein autonomes Schienenfahrzeug*. In: *Tag des Systems Engineering 2011*. Hanser; 2011. pp. 45–56.
- [11] Object Management Group. *OMG Systems Modeling Language (OMG SysML™) Version 1.6*; 2019. available at: <https://www.omg.org/spec/SysML/1.6>
- [12] Object Management Group. *SysML V2: The next-generation Systems Modeling Language*; 2019. available at: <http://www.omg.org/SysML-2.htm>
- [13] Bajaj M, Friedenthal S, Seidewitz E. *Systems Modeling Language (SysML v2) Support for Digital Engineering*. INSIGHT. Vol. 25; 2022. pp. 19–24. DOI: 10.1002/inst.12367
- [14] Friedenthal S, Moore A. *A Practical Guide to SysML: The System Modeling Language*. 3rd ed. Elsevier. Burlington; 2015. DOI: 10.1016/C2013-0-14457-1
- [15] de Saqui-Sannes P, Vingerhoeds RA, Garion C, Thirioux X. *A Taxonomy of MBSE Approaches by Languages, Tools and Methods*. IEEE Access. Vol. 10; 2022. pp. 120936–120950. DOI: 10.1109/ACCESS.2022.3222387
- [16] Bopp R. *Design for X-Methoden*. In: *Forschungs- und Entwicklungsmanagement. Technologiemanagement — Wettbewerbsfähige Technologieentwicklung und Arbeitsgestaltung*. Vieweg+Teubner Verlag. Wiesbaden; 1997. pp. 195–203. DOI: 10.1007/978-3-663-05946-2\_10
- [17] ISO/EC/IEEE (Ed.). *International Standard ISO/EC/IEEE 26580, Software and systems engineering - Methods and tools for the feature-based approach to software and systems product line engineering*. 35.080 Software; 2021.
- [18] Berschik MC, Züfle M, Laukotka FN, Krause D. *Product Family Engineering along the Life Cycle – Research Aspects to cope with Variability in Advanced Systems*. DesignScience - Special Issue: *Advanced Systems Engineering in Research and Education*; 2024. DOI: 10.1017/dsj.2024.21
- [19] Laukotka FN, Berschik MC, Krause D. *Enhancing Design representations of information and knowledge of complex and long-living assets by applying system modelling techniques*. International DESIGN Conference 2024. Proceedings of the Design Society. Vol 4; 2024. pp. 403–412. DOI: 10.1017/pds.2024.43
- [20] Friedenthal S, Izumi L, Meilich A. *Object Oriented Systems Engineering*. In: *Process Integration for 2000 and Beyond: Systems Engineering and Software Symposium*. New Orleans; 1998.
- [21] Berschik MC, Laukotka FN, Züfle M, Krause D. *Supporting Product Family Representations by methodically utilising Meta-Models*. International DESIGN Conference 2024. Proceedings of the Design Society. Vol 4; 2024. pp. 2513–2522. DOI: 10.1017/pds.2024.254
- [22] Laukotka FN, Krause D. *Supporting Digital Twins for the Retrofit in Aviation by a Model-Driven Data Handling*. Systems 2023. Vol 11,142; 2023. DOI: 10.3390/systems11030142