INTERNATIONAL DESIGN CONFERENCE – DESIGN 2024
https://doi.org/10.1017/pds.2024.43

Enhancing design representations of information and knowledge of complex and long-living assets by applying system modelling techniques

Fabian Niklas Laukotka ✉, Markus Christian Berschik and Dieter Krause

Hamburg University of Technology, Germany

✉ fabian.laukotka@tuhh.de

Abstract

Managing general domain knowledge and asset-specific information in the form of digital representations is especially important and challenging when focusing on long-living and complex assets. Implicit knowledge and existing structures need to be captured and digitalised, ideally without introducing unnecessary complexity through unfamiliar wording or new structures. To achieve this, a methodical approach that utilises ontologies as well as system modelling techniques and focuses on early-stage model instantiation is presented and applied to the cabin retrofit of aircraft.

Keywords: design representation, knowledge representations, system modelling, aircraft cabins, meta-model

1. Introduction

Over the past years, products have become increasingly more complex and interdisciplinary. Not least as part of the strive towards increased sustainability, longevity and maintainability, especially of big and costly products, became an additional factor. Therefore, evermore information needs to be managed across the many stakeholders and through the years. With long-living assets, there will be different teams that need to handle the information, even within the same stakeholder. Hence, a comprehensible structure of the information and its handling is required. Not only does information specific to a product or project need to be considered, but also the common knowledge of the company or whole industry needs to be managed. This is especially challenging for long-living assets as stakeholders are facing assets, that are already produced and come with structures and strategies that are already established and not created by the organisation itself (Moenck *et al.*, 2022b). Because the occurring meta-information, like relations and dependencies between information or fragments of information, are increasingly important in this situation as well, the handling of information needs to be able to handle that information as well. Making this, often abstract, knowledge digitally available and processable poses a challenge. In the field of Systems Engineering (SE) System Models are the means of choice for handling and defining abstract and meta information. These techniques can be transferred to topics aside from SE, however, a solid foundation of the modelling is required to enable the necessary consistency and comprehensibility across teams, projects and years. These underlying data structures for those representations are often very abstract and therefore usually not easy to establish directly. To summarise, the following question is formulated: How can the creation of virtual representations of complex and long-living assets, especially the occurring information and knowledge, be methodically supported and enhanced? While the answer to that question will incorporate many aspects, this publication will focus on the very first steps; the identification of the relevant aspects in the real world and a methodical approach to transfer those into the virtual world using abstractions and creating a framework. Other publications already describe applications and approaches that apply this methodology and utilise the created representations for applications in aviation's retrofit (Moenck *et al.*, 2022b; Laukotka and Krause, 2023).

2. Research background and fundamentals

Before a methodology to create those representations is presented, the underlying research background and relevant fundamentals are described in the following Section.

2.1. Managing knowledge and knowledge-based engineering

Companies do not only store and handle information related to current or ongoing tasks and projects but also strive to keep the general know-how within the company accessible to the engineers. The volatility of knowledge, due to changes in personnel or reorganisations becomes the central challenge for retaining the organisational knowledge base (Lehner, 2021). To support the recording, storing and general managing of knowledge, standards like the VDI 5610 (Verein Deutscher Ingenieure e.V., 2017) have evolved. In practice knowledge typically accumulates in the form of tutorials, short Lessons Learned and Best Practices documentation. In case the required infrastructure is available, this can for example easily be done using collaboratively created Wikis (Kiniti and Standing, 2013). Depending on the information to be described and in addition to textual descriptions, visual diagrams, mind maps or hierarchical views can help to document the knowledge.

The term ontology is widely used in the field of computer science. It is often applied in the areas of knowledge engineering or knowledge representation. As described in philosophy, an ontology can also be understood as a conceptualisation (Guarino, 1998). Ontologies allow for the transparent definition of the structure of knowledge within a subject and, thus, ease the challenge of communication between people, people and machines and even between machines (El-Haji, 2014). They define common terminology and also the relations between terms and elements. Thus, they can ensure that the same term is interpreted the same way and is cognitively connected to the same concept (Lehner, 2021). Committing to a common ontology means "to use the shared vocabulary coherently and consistently" (Gruber, 1995). A decisive factor in using ontologies is the so-called open-world assumption (Horrocks *et al.*, Vol. 2003). This assumption describes that everything not explicitly described is not known to the ontology and thus not explicitly forbidden. Thus, ontologies use underspecification as a specific feature of abstraction. This kind of description contrasts with the so-called closed-world assumption of system models and, thus, also meta-models. These models allow or forbid everything that lies outside what has been explicitly specified and, thus, define a strict system boundary (Aßmann *et al.*, 2006). Furthermore, the type of description between an ontology and a meta-model is different. An ontology, as described here, is descriptive, meaning that it describes the world for which it is created. On the other hand, a meta-model is prescriptive, in the sense that the model prescribes what the world the meta-model is describing looks like. Thus, an ontology is often located in the problem space whereas a meta-model is located in the solution space (Aßmann *et al.*, 2006).

2.2. System modelling, as originated from Model-Based Systems Engineering

The integration of more electronics and software into products made product development increasingly multidisciplinary, which required new approaches and methodologies. Going beyond simply the task of engineering systems, Systems Engineering (SE) provides these methodologies that are applied throughout different industries (INCOSE, 2007). The *International Council on Systems Engineering (INCOSE)* defines SE as an interdisciplinary approach that "means to enable the realisation of successful systems" and that "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 […]" (Walden *et al.*, 2015). While this new methodology supports these multidisciplinary products, with increasingly larger projects new challenges arose. The handling of the occurring data of the engineering and management processes got more complex, which was coped by moving from document-based to model-based documentation of (meta-) information, especially semantics or relations. This now-called Model-based Systems Engineering

https://doi.org/10.1017/pds.2024.43 Published online by Cambridge University Press

(MBSE) is described by the *INCOSE* as “the formalised 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 " (INCOSE, 2007). The information is now handled in centralised System Models usually in the form of graphical SysML Models (Williams and Boing, 2021; NoMagic Inc., 2011). These centralised System Models can include multiple views, focusing on different aspects of the system, enabling a clear and distinct structure while also allowing for a single source of truth for the model elements (Friedenthal *et al.*, 2015). Specialised authoring tools support the engineers during the creation of these models and even some simple analysis of modelled information and relations across the different views (NoMagic Inc., 2011).

Besides the origins of Systems Engineering, the fundamentals and best practices evolved with MBSE can be transferred to tasks facing similar challenges regarding the handling of data, which is why the System Modelling approaches can nowadays be found applied to many different applications and scenarios. While this is strictly speaking no System Engineering, thus, no, MBSE, the three pillars of modelling “modelling language, modelling tool and modelling method” as described by *Delligatti* (Delligatti, 2014) can be transferred. In that case and in a generalised way, these three pillars are supplemented by three more elements: the Modelling Context, the resulting System Model and the targeted Implementation and Application. These elements of System Modelling are visualised in Figure 1 and will be further described subsequently.

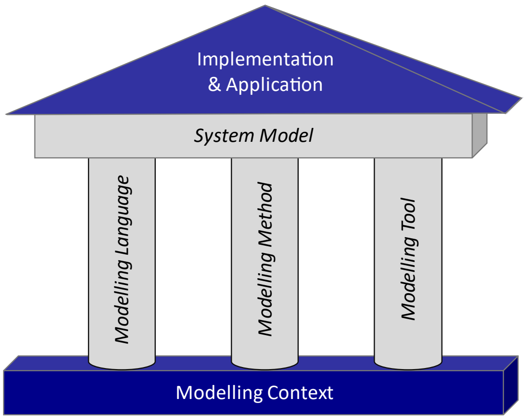

Figure 1. The system modelling temple: Delligatti’s three pillars of system modelling, the modelling context and the resulting system model as the joining elements to enable the implementation and application

The **Modelling Context** is the base for all other elements, as it provides the scenario with given boundary conditions, and, thus the demands that are to be satisfied.

The **Modelling Language** provides the definition of available modelling elements and thus, the possible syntax and semantics that are required for the modelling of non-geometric information and especially relations. *Delligatti* describes the language as “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” (Delligatti, 2014). Currently, SysML is one of the most used Modelling Languages (Berschik *et al.*, 2023).

The **Modelling Method** is like a road map or strategic plan. It is described as “a documented set of design tasks that a modelling team performs to create a system model” (Delligatti, 2014).

Modelling Tools are the authoring tools the System Models are created with. They assist the engineer with the incorporation of the rules for modelling and the visualisation given by the modelling language. They “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” (Delligatti, 2014).

The **System Model** as the core of the model-based approaches is the result of the modelling according to *Deligatti*’s three pillars as presented before. It stores the information using different views of the system of interest, hence, also is the basis for the application.

https://doi.org/10.1017/pds.2024.43 Published online by Cambridge University Press

The **Implementation and Application** depict the overall goal of the modelling effort. They provide a solution or service to face the demand that originated from the Modelling Context. This is made possible by the structured approach enabled by the four modelling elements, as described before.
This generalised structure needs to be tailored to the specific context and scenario it is meant to be applied. This includes a careful selection and definition of the Modelling Language, Method and Tools. It shall be noted, that while SysML depicts a great start for a modelling language, it also can be easily extended by defining custom stereotypes that then can be applied to blocks, relations etcetera. In system modelling, stereotypes are comparable to labels or classes, as known from object-oriented programming, and allow elements to be tagged, and, thus categorised or described by a standardised notation. This is then to be done by first defining the respective meta-models before the actual creation of the information-storing models. Similarly, there is a variety of Modelling Methods available. As these are tailored to a specific use case, it is often appropriate to define an own Modelling Method, that, however, might be based on available ones. If done and documented thoroughly, this allows for a systematic, consistent and structured documentation of information and knowledge: It also supports a coherent model creation even across engineers, teams and projects. That especially comes in handy, when targeting comprehensive systems and relations that need to grow and be handled by different engineers through the years.

3. Methodology

As described in the previous Section, there are many concepts and approaches in both areas, knowledge management and system modelling, that are promising. Especially, when considering complex and long-living objects, many advantages can be expected from the application of system modelling techniques. This requires a procedure to transfer the basic system structure from a real product to a complete digital System Model. Thus, a resilient and complete System Model based on a complete system understanding from a real use case needs to be established. In this Section, the proposed methodology is presented.

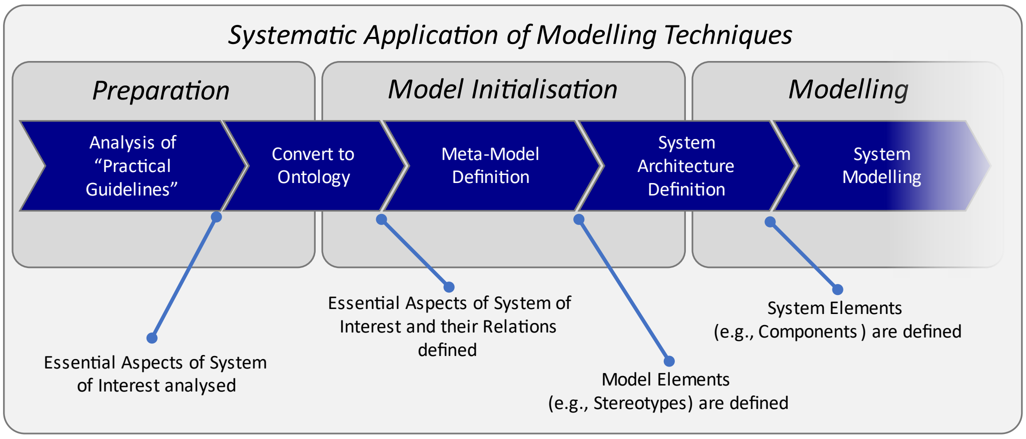

Figure 2. Approach for the establishment of the fundamentals for the systematic modelling of complex and long-living assets

The approach is divided into three phases as shown in Figure 2: the preparation phase, the model initialisation phase and the modelling phase.
In the preparation phase, the real overall system is analysed. For this purpose, the information to be modelled and the guidelines or rules used are collected. This forms the basis for the further steps and should therefore be close to real everyday life. Based on the analysis, an ontology of the information is created. In the ontology, the occurring essential aspects of the system of interest are summarised and related to each other on an abstract level. By using the ontology, the representation of the data connections, also called meta-data, should be presented intuitively.
With the help of the ontology, a meta-model can be defined in the model initiation phase. The information contained in the ontology is used as the basis for building the syntax of the modelling. Due to the initial description under an open-world assumption, based on the ontology, the contents can be discussed within the development team. They can be abstracted in the meta-model and standardised for

https://doi.org/10.1017/pds.2024.43 Published online by Cambridge University Press

further modelling afterwards. Finally, the overall system of related data can be built upon the meta-model. In the last step, the modelling can be started based on the established meta-model and a chosen modelling method. Hence, the modelling procedure does not change, the modelling is only based on a correspondingly more robust foundation. With the described procedure, different cases of application can be considered. Figure 3 shows a conceptual view of the application of the procedure.

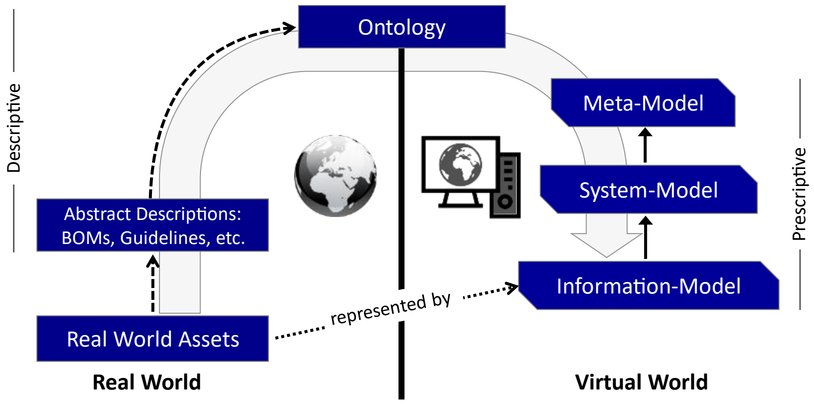

Figure 3. Generic representation of possible implementations using the described methodology

The basic aim of the approach is to enable the modeller to virtually capture the knowledge about a real-world asset and to map it in a useful way, even in the 3rd-party scenario, where the organisation is facing finite but established information. In this way, the information of the real-world asset can initially be described abstractly by descriptive models, for example with the help of a BOM or various development guidelines. If the available information is summarised in a superordinate ontology, this data can be discussed by the development team across the organisation. Based on the ontology, the step into the prescriptive description can be implemented in the Virtual World. For this purpose, the described information of the ontology must be transferred into the strictly defined syntax and semantics of the meta-model. With this transfer, further modelling is restricted, accordingly, it is important to iteratively adapt and verify the meta-model. By using the previously discussed ontology, this step can be greatly simplified, as the meta-model is thereby built on a consolidated basis. Subsequently, the System Model can be implemented based on the adaptation of the language by the meta-model. If the System Model is then filled with further instances, the corresponding information model can also be built. This completes the representation of the real-world asset in the information model. In the following, the procedure is shown based on a real use case.

4. Application using the example of cabin design for the cabin retrofit of aircraft

With typically around 400,000 parts (Stark, 2016) and a life span regularly up to 20 years (Niţă and Scholz, 2011) or even more, aircraft can be seen as one of the most complex products. Because of high safety standards and certification requirements that come with extensive documentation, the same can be said regarding the management of the respective information. One periodically occurring task where this becomes relevant is the so-called cabin retrofit. Every five to seven years, the cabin of an existing aircraft is partly equipped with new components, which requires thorough documentation of the aircraft's current state to allow for the new cabin to be installed into its body (Moenck *et al.*, 2022a). This is usually done by a third party who is facing the challenge of designing and installing a new cabin into an already-produced aircraft without having access to all information from its design and production. Instead, it heavily relies on common knowledge and the available fragmented documentation. Thus,. accessing the required information is currently mostly done manually and requires elaborate work by the engineers, as the information is fragmented across numerous documents and even the same space in the aircraft is described for each domain - *electric, water and waste*, etcetera - separately (Laukotka and Krause, 2023). With the information itself comes meta-information such as relations or dependencies that need to be considered as well. Despite the often very specific information

https://doi.org/10.1017/pds.2024.43 Published online by Cambridge University Press

regarding individual aircraft, there is comprehensive common knowledge within this field or domain. This, however, often comes only through learning and first of all experiences and is not digitally accessible (Laukotka and Krause, 2023). Conveying this knowledge into a digital usable format would allow for an enhanced representation of the information and enable the usage of digital services. Yet, such a digital base requires a solid foundation, which can be achieved by utilising system modelling techniques and the presented methodology.

While the retrofit poses a good application that is currently being pursued, the shown examples only depict a small fraction of the created models to allow for easier comprehensibility of the overall idea.

Before creating any models or even the ontology, there needs to be a solid understanding of the system of interest. Thus, the first step is the analysis of the system as it occurs in the real world. The goal is to identify structures that are already present and possibly in use every day. The models should reflect the real world and adapt to it. The possible users of the models should see them as a benefit, which includes the availability of already familiar and established structures.

Analysing the real world

Aviation excels with vast numbers of components, strict regulations and overall sophisticated products. To cope with the complex product and document structure and create a common referencing scheme the so-called *ATA-Chapters*, have become the de facto standard for commercial aircraft documentation (Mensen, 2013; FAA - Federal Aviation Administration, 2008). Originally established in 1965, today they are still widely used within documents, visualisations or even filenames. Contemporaneously with these complex structures, most of the commercial passenger aircraft flying nowadays are broadly similar, as there are resemblant standard elements in almost every aircraft. Thus, in 1997 Jackson visualised an abstracted generic system architecture, defining the essential segments, parts and systems in an *Aircraft System Architecture (ASA)* that also references the correlating main *ATA-Chapters* (Jackson, 1997). These two schemata, *ATA-Chapters* and the *ASA*, together depict an excellent starting point for the steps in the Preparation-Phase of the methodology.

Besides these general structurisations, there is more specific information that applies to most commercial aircraft and, thus, can be facilitated to enhance the virtual representation in the documentation of each of these aircraft. The structural setup of airframes, the aircraft's body, is usually composed of several *Frames* and *Stringers*. For each type of aircraft, for example, an Airbus A320, they are standardized. Because they are positioned at explicitly defined coordinates, referring to a nearby *Frame* also allows for a broad localisation within the aircraft. As the *Frames* are major structural mounting points for components, especially those of the cabin, these relations are occurring regularly. Regarding the components of the cabin, because of strict regulations and many dependencies that need to be considered, actually occurring variety of available types of components of an aircraft cabin is limited. Most of the elements of the cabin can be found in nearly every commercial passenger aircraft, with varying detailed implementations (Niţă, 2012; Kopisch and Günter, 1992). Besides others, these elements are for example the *Overhead Stowage Compartments (OHSC)* and *Seats*, or the so-called monuments including the *Galleys* (aircraft kitchens) and *Lavatories* (aircraft toilets). While their specific implementations may vary, the general setup of the elements including their required interfaces, to structures and infrastructure systems such as electronics or plumbing, are generally known.

Across all components and types of aircraft, this is a tremendous amount of knowledge, that is usually documented within standard documentation or known simply because of experiences. Making them digitally processable by transferring this implicit knowledge into models poses a real opportunity to improve the overall availability of knowledge.

Creating an ontology

With the analysis of the real world done, key points from the results are deduced, that are the basis for the ontology: References to frames, as part of the airframe, are regularly to be found when documenting the position of components of of the cabin within the aircraft. Additionally, many components and their respective documentations are referencing the corresponding *ATA-Chapters*. Both aspects are valuable to be included in the knowledge management. From the viewpoint of the organisation performing the retrofit, the aircraft can be divided into two segments; the airframe and the aircraft cabin. Both are

https://doi.org/10.1017/pds.2024.43 Published online by Cambridge University Press

generalisations consisting of individual elements; airframe elements or cabin elements respectively. An airframe section is a special kind of airframe element, as airframes are typically manufactured in defined sections that are handled individually. However, when combined with the whole airframe, they are not obvious anymore but are sometimes used for a broad locational reference, e.g., Section 19 refers to the aft segment of an Airbus A320. Likewise, there are cabin components, which are special cabin elements within the Cabin. The aforementioned *Seats*, *OHSC* and monuments like *Galleys* and *Lavatories* rank among these cabin elements. Each of these components consists of multiple distinct parts.
Both, airframe elements and cabin elements, can be described in more detail using properties, are regularly classified into the mentioned *ATA-Chapters* and may have connections. While there are different kinds of connections, physical connections such as being (structurally) *mounted to* another element are one of the most common. Visualising the described facets and their connection creates an ontology. Figure 4 shows the excerpt of the ontology including the described relations of the case study.

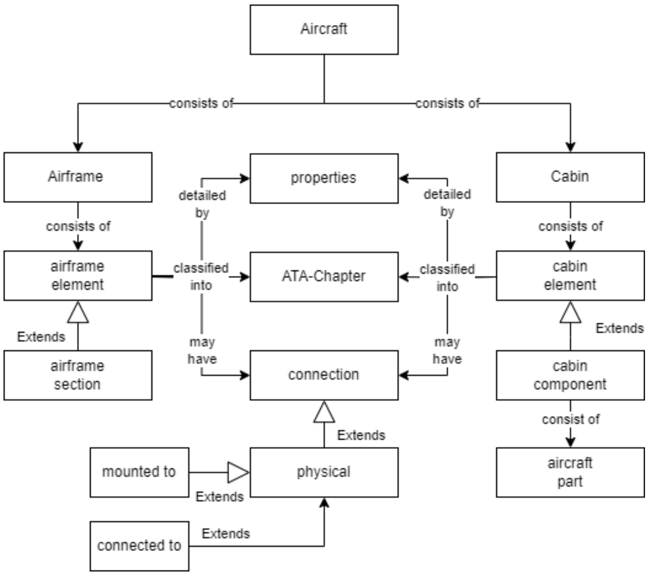

Figure 4. Excerpt of the ontology of the case study

The ontology itself can already help to communicate more clearly in the real world. Within the scope of this work, it is only the beginning as the next step is to transfer it into the virtual world, where it depicts the ground for the creation of the meta-model, thus, laying the foundation for the modelling.

Meta-model definition

To be able to create models based on the defined ontology, it needs to be transferred into the model space. Now, system modelling techniques are applied; Referring to the temple of modelling used the Modelling Tool is NoMagic's *Cameo System Modeller* and the Modelling Language is based on *SysML*, as hereinafter custom stereotypes extending the original set of *SysML* are defined and used. The context of the system modelling is the presented case study; modelling knowledge of aircraft for the cabin retrofit, where with the application an improved access to knowledge and information is targeted.
The first step is the creation of a meta-model, defining the elements, e.g. stereotypes, available for the model. It is based on the previously created ontology; in the centre of the meta-model, the elements and relations of the ontology are to be found (see Figure 5). These elements of the ontology are represented by stereotypes in the model, which can be applied to blocks and relations during the actual modelling. As the modelling language is based on *SysML*, some of these representing stereotypes are already defined for *SysML* or the *Magic Draw Customization for SysML*. It is good practice to relate to these stereotypes instead of creating new ones. Those ontology elements that cannot be related to already existing stereotypes are represented by custom stereotypes, which are clearly defined within a custom profile (see box *Aviation Framework* in Figure 5). With this meta-model created, the available stereotypes based on the ontology are defined, specifying the available model elements and relations functioning as a kind of framework.

https://doi.org/10.1017/pds.2024.43 Published online by Cambridge University Press

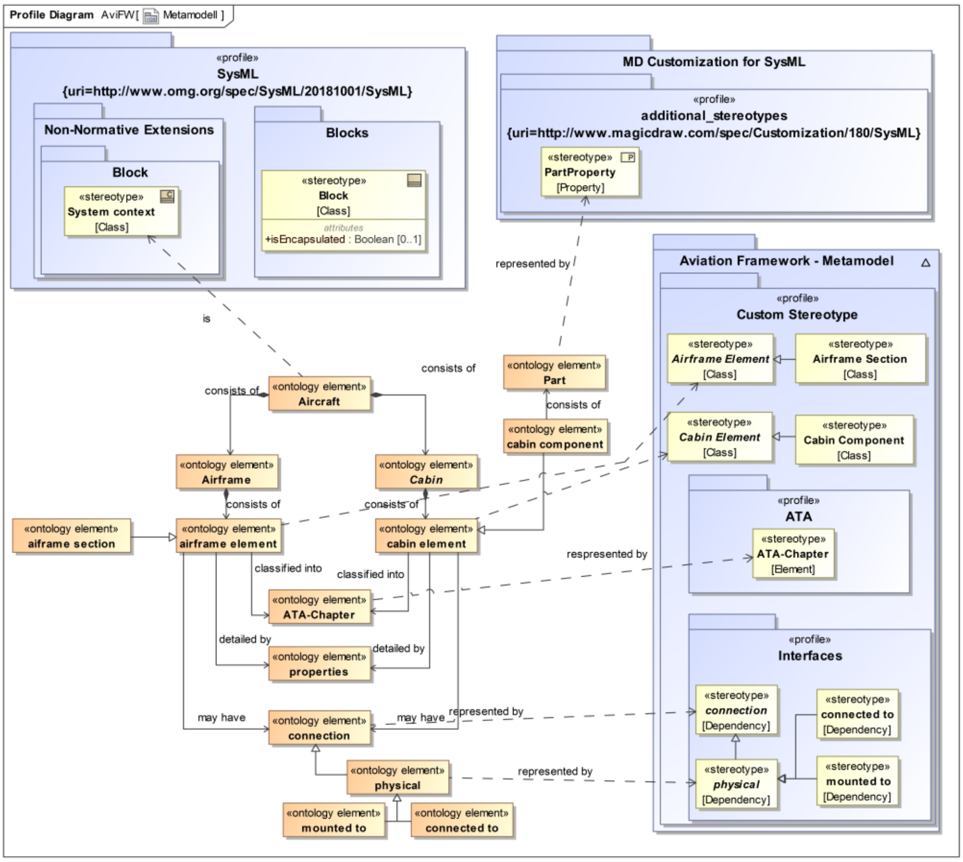

Figure 5. Meta-Model defining the available stereotypes, based on the ontology

System architecture definition

With the step of defining the System Architecture, the commonly done modelling is starting. The architecture of the system of interest, in this case, aircraft with a focus on their cabin, is modelled using block-definition diagrams. Using the stereotypes defined in the meta-model, the blocks and relations are labelled accordingly. With these diagrams, the available components and elements are defined on a generic, non-specific level. This is comparable to the definition of classes in the object-oriented approach to software engineering, where the available classes, their properties and maybe inheritances are defined prior to the actual instantiation. Together with the definitions of stereotypes in the meta-model, the building blocks for the specific instances are now fully defined.

System modelling

Finally, specific assets of the real world are modelled by creating instances of the elements defined in the system architecture. The relations between instantiated elements can be classified using the stereotypes defined in the meta-model. This modelling can be performed on very specific levels, defining the elements and relations of one specified aircraft, or on a more generic level, defining elements and relations that are applicable to a whole fleet of aircraft or even all aircraft of the same type. An excerpt of the former is shown in Figure 6, where the position of two *OHSCs* and the four *Frames* they are *mounted to* are modelled in the virtual world (right side), representing the actual installation in the real world (left). An example of the latter is the modelling of *Lavatories* or *Galleys*, including their systematic dependencies like the infrastructure systems they need to be *connected to*. While this may be obvious when focusing on one component of one type of aircraft, however, when including more components and types of aircraft, and considering their specifically defined systems and interconnections, having them digitally available in models poses a great benefit. A comprehensive modelling of generic information that applies to a range of instances will lead to easier access to information, that otherwise would have to be determined laboriously.

https://doi.org/10.1017/pds.2024.43 Published online by Cambridge University Press

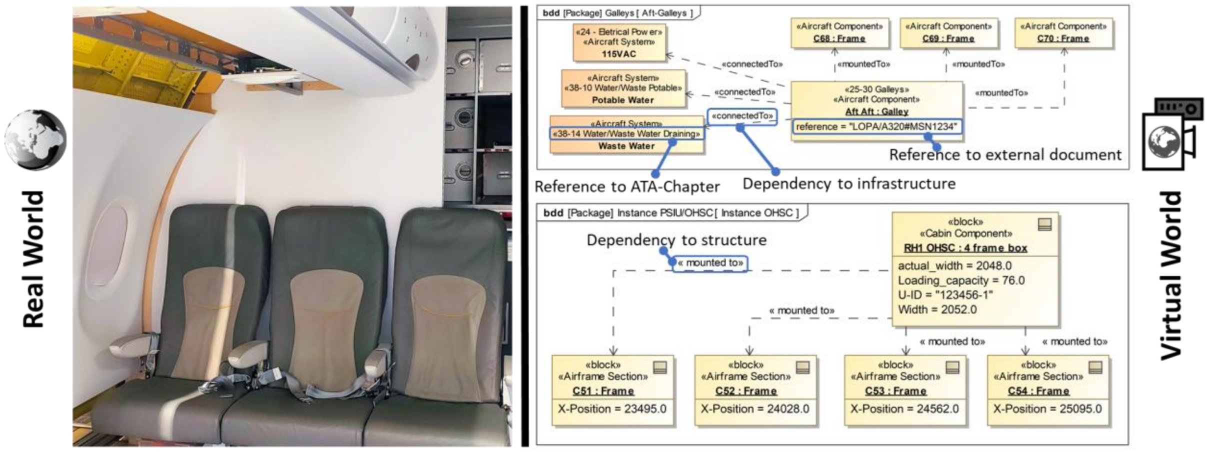

Figure 6. Excerpt of the information model (right) representing parts of the real asset (left)

When performing this modelling for multiple aircraft and including the common knowledge by modelling especially on the more generic levels, the reusability of models and access to the information is greatly enhanced by the clearly defined underlying structure. Additionally, references to the *ATA-Chapters* or external documents including more details about the respective information can be included using the properties of the elements. This further enhances the representations, as it allows the models to function as an assistance for the handling and connecting of already available and more specific data.

5. Conclusion and outlook

Digital access to information about assets can improve tasks along a range of applications during their lives. With complex products and longevity, a methodical approach to transfer this information, often existing in implicit forms, to virtual representation, is required. As long-living assets regularly face different stakeholders, who in consequence need to handle information and structures that are already established and not created by themselves, this transfer of knowledge and state from the real world to the virtual world can be challenging. This work presented an approach, that facilitates system modelling techniques to provide a basis for System Models, that are grounded in the engineer's everyday life. The methodically created foundation of the models allows the engineers to adhere to a common modelling standard and ensure compatibility. Utilising an ontology that originates in the real world and consecutive meta-models, allows for the creation of systematically structured information models by third parties and in later life phases. The modelling within the authoring tools becomes more defined that way, as the available elements and relations are clearly specified, and external tools can also rely on knowing what information and structure occur within exports. This also allows the data to be parsed using other tools, which is improved because of this defined structure. In currently ongoing research, this is used to great effect to allow access to the modelled information using custom user interfaces and also to perform extensive analyses of the modelled information (Laukotka and Krause, 2024). This all leads to great enhancements of the design representations of the available information and knowledge of complex and long-living assets such as aircraft. Currently, the methodology is applied to the case study in more detail. Besides the circumstances of the presented case study, the modular structure of product families results in similar challenges, like representing the induced complexity, that could similarly be faced using this approach. Ongoing and future work will also enable easier access to the modelled information from other processes by i.a. implementing custom user interfaces that provide engineers with a comfortable way to the stored information and knowledge. This integration greatly benefits from the clear definition and structure provided by the ontology and system modelling techniques.

Acknowledgements

Parts of this work are part of the research projects InDiCaT (Intelligent Digital Cabin Twin, contract number 20D1902C) and VerDiKa (Vernetzte Digitale Kabine, contract number 20D1923E), both funded by the Federal Ministry for Economic Affairs and Energy as part of the Federal Aeronautical Research Programme LuFo VI-1. The authors would like to thank all involved parties for their funding and support.

https://doi.org/10.1017/pds.2024.43 Published online by Cambridge University Press

References

Aßmann, U., Zschaler, S. and Wagner, G. (2006), “Ontologies, Meta-models, and the Model-Driven Paradigm”, in Calero, C. (Ed.), *Ontologies for software engineering and software technology: With 46 tables*, Springer Verlag, Berlin, Germany, pp. 249–273. https://dx.doi.org/10.1007/3-540-34518-3_9.

Berschik, M.C., Schumacher, T., Laukotka, F.N., Inkermann, D. and Krause, D. (2023), “MBSE within the engineering design community - an exploratory study”, paper presented at 24th International Conference on Engineering Design, 23rd-27th July 2023, Bordeaux, France. https://dx.doi.org/10.1017/pds.2023.260.

Delligatti, L. (2014), *SysML distilled: A brief guide to the systems modeling language*, Addison-Wesley, Upper Saddle River, NJ, United States.

El-Haji, M. (2014), *Ontologie-basierte Definition von Anforderungen an Validierungswerkzeuge in der Fahrzeugtechnik*, Dissertation, *Karlsruher Schriftenreihe Fahrzeugsystemtechnik*, Band 48, KIT Scientific Publishing, Karlsruhe, Germany. https://dx.doi.org/10.5445/KSP/1000052725.

FAA - Federal Aviation Administration (2008), “Joint Aircraft System/Component Code. Table and Definitions”, JASC, available at: https://av-info.faa.gov/sdrx/documents/JASC_Code.pdf (accessed 18 January 2023).

Friedenthal, S., Moore, A. and Steiner, R. (2015), *A Practical Guide to SysML: The System Modeling Language*, Third Edition, Elsevier/MK, Burlington, MA, United States. https://dx.doi.org/10.1016/C2013-0-14457-1.

Gruber, T.R. (1995), “Toward principles for the design of ontologies used for knowledge sharing?”, *International Journal of Human-Computer Studies*, Vol. 43 No. (5-6), pp. 907–928.

Guarino, N. (1998), “Formal Ontology and Information Systems”, *Proceedings of FOIS’98*, Vol. 1998, pp. 3–15.

Horrocks, I., Patel-Schneider, P.F. and van Harmelen, F. (Vol. 2003), “From SHIQ and RDF to OWL: the making of a Web Ontology Language”, *Journal of Web Semantics*, Vol. 1 No. (1), pp. 7–26.

INCOSE (2007), “INCOSE Systems Engineering Vision 2020”, available at: https://sdincose.org/wp-content/uploads/2011/12/SEVision2020_20071003_v2_03.pdf.

Jackson, S. (1997), “Systems Engineering for Commercial Aircraft”, *INCOSE International Symposium*, Vol. 7 No. (1), pp. 36–43.

Kiniti, S. and Standing, C. (2013), “Wikis as knowledge management systems: issues and challenges”, *Journal of Systems and Information Technology*, Vol. 15 No. (2), pp. 189–201.

Kopisch, M. and Günter, A. (1992), “Configuration of a passenger aircraft cabin based on conceptual hierarchy, constraints and flexible control”, *Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, Vol. 604, Springer Verlag, Berlin, Germany, pp. 421–430. https://dx.doi.org/10.1007/BFb0024994.

Laukotka, F.N. and Krause, D. (2023), “Supporting Digital Twins for the Retrofit in Aviation by a Model-Driven Data Handling”, *Systems*, Vol. 11 No. (3), p. 142.

Laukotka, F.N. and Krause, D. (2024), “Combining System Modelling and Graph Databases to Improve Access and Analysis in Data-Intensive Engineering Tasks. Paper Accepted/Unpublished”, in 2024 *IEEE International Systems Conference (SysCon), Montreal, QC, Canada*, IEEE.

Lehner, F. (2021), *Wissensmanagement: Grundlagen, Methoden und technische Unterstützung*, 7. überarbeitete und erweiterte Auflage, Hanser, München.

Mensen, H. (2013), *Handbuch der Luftfahrt*, 2. Auflage, Springer Verlag, Berlin, Germany.

Moenck, K.H.W., Laukotka, F.N., Deneke, C., Schüppstuhl, T., Krause, D. and Nagel, T.J. (2022a), “Towards an Intelligent Digital Cabin Twin to Support an Aircraft's Retrofit and Base Maintenance”, in *SAE Technical Paper Series, MAR. 15*, 2022, Warrendale, PA, United States. https://dx.doi.org/10.4271/2022-01-0046.

Moenck, K.H.W., Laukotka, F.N., Krause, D. and Schüppstuhl, T. (2022b), “Digital Twins of existing long-living assets: reverse instantiation of the mid-life twin”, in *Proceedings of the 33rd Symposium Design for X (DfX 2022), Hamburg, Germany*, The Design Society, p. 10. https://dx.doi.org/10.35199/dfx2022.20.

Niţă, M.F. (2012), *Contributions to aircraft preliminary design and optimization*, Dissertation, Verlag Dr. Hut, München, Germany.

Niţă, M.F. and Scholz, D. (2011), “Business opportunities in aircraft cabin conversion and refurbishing”, *Journal of Aerospace Operations (AOP)*, Vol. 1 No. (1-2), pp. 129–153.

NoMagic Inc. (2011), “Magic Draw - Model-based Systems Engineering”, (accessed 21 September 2022).

Stark, J. (2016), *Product Lifecycle Management: Volume 2 - The Devil is in the Details*, Springer International Publishing, Cham, Switzerland. https://dx.doi.org/10.1007/978-3-319-24436-5.

Verein Deutscher Ingenieure e.V. (2017), *Wissensmanagement im Ingenieurwesen - Wissensbasierte Konstruktion (KBE)*, Vol. 03.100.99 No. 5610 Blatt 2, Beuth Verlag GmbH, Berlin, Germany.

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 INCOSE-TP-2003-002-04*, 2015, 4th Edition, Wiley, New York, NY, United States.

Williams, M. and Boing (2021), “A Roadmap for MBSE Data Standards. Annual INCOSE International Workshop 2021”.