RESEARCH ARTICLE

Graph-based information management for retrofitting long-living assets

Fabian N. Laukotka[1] , Keno Moenck[2] , Thorsten Schüppstuhl[2] and Dieter Krause[1]

[1]Institute of Product Development and Mechanical Engineering Design, Hamburg University of Technology, Hamburg, Germany
[2]Institute of Aircraft Production Technology, Hamburg University of Technology, Hamburg, Germany
Corresponding author: Fabian N. Laukotka; Email: fabian.laukotka@tuhh.de

Abstract

Many documents are produced over the years of managing assets, particularly those with long lifespans. However, during this time, the assets may deviate from their original as-designed or as-built state. This presents a significant challenge for tasks that occur in later life phases but require precise knowledge of the asset, such as retrofit, where the assets are equipped with new components. For a third party who is neither the original manufacturer nor the operator, obtaining a comprehensive understanding of the asset can be a tedious process, as this requires going through all available but often fragmented information and documents. While common knowledge regarding the domain or general type of asset can be helpful, it is often based on the experiences of engineers and is, therefore, only implicitly available. This article presents a graph-based information management system that complements traditional PLM systems and helps connect fragments by utilizing generic information about assets. To achieve this, techniques from systems engineering and data science are used. The overarching management platform also includes geometric analyses and operations that can be performed with geometric and product information extracted from STEP files. While the management itself is first described generically, it is also later applied to cabin retrofit in aviation. A mockup of an Airbus A320 is utilized as the case study to demonstrate further how the platform can provide benefits for retrofitting such long-living assets.

Impact Statement

The retrofitting of long-living assets presents a unique set of challenges, primarily related to the management of information about the specific assets. Over the years, the condition of the assets has changed, and while many changes may be documented, this is usually done in numerous different and fragmented documents. An overarching graph-based information management system can facilitate common knowledge and known relationships to provide engineers with an access system that eases the otherwise tedious work of collecting and assembling the required information. Exploiting further potentials of graph databases extends the created platform with functions that allow geometric-oriented services such as mountability assessments. This article presents a graph-based information management and applies it to retrofitting aircraft cabins.

1. Introduction and motivation

Not least because of rising sustainability reasons, but also with regard to increasingly high values and complex structures on the technological and system level as well as resulting cost, increasing an

asset's lifespan is of exceeding interest. Typically, due to their long lifespan, assets are handled by multiple and changing stakeholders such as the original designer and manufacturer, the owner, and also additional parties such as maintenance or retrofit organizations. As a result, there are numerous data-intrinsic challenges, ranging from specific aspects such as the required and occurring levels of detail or model fidelity to stakeholder-related ones such as accessibility or confidentiality (Stark, 2016; Moenck et al., 2022a). The extension of the usage phase of the asset is associated with increased activities to maintain its condition and usability, which require access to asset-specific information, e.g., for planning and engineering efforts (Moenck et al., 2022b). The concept of product lifecycle management (PLM) is a conceptual and technical approach that targets the handling of information along the lifecycle of a product. A product data management (PDM) system can manage and provide product data in a (semi-)structured way, including semantically enriched product data, the Bill of Materials (BOM), or geometric representations (Stark, 2020). However, current PDM systems, often historically grown software monoliths, are usually file-based and only partially allow for additional layers of semantics (Moenck et al., 2023). In addition, within their closed, proprietary structure, the extension of, e.g., data models to support new requirements, tools, or queries is inhibited.

As outlined in previous work (Moenck et al., 2022a; Laukotka and Krause, 2023), this handling of information is especially challenging for the process of retrofitting assets in their later life phases. Much of the required information is available in some form, such as manuals, maintenance documentation, or general knowledge, but at the same time, it is usually fragmented across many systems, documents, and files. This fragmentation of information often leads to interoperability issues. Sometimes, the required information is not even modeled and cannot be integrated, linked, or managed in modern PDM software systems. Hence, the connection of these fragments and the handling of metadata and relations becomes a crucial aspect of information management. Nevertheless, this consolidation of the individual fragments is a challenging endeavor because the respective information is distributed across various silos, can be of different types, and are subject to differing chronological states: When retrofitting complex systems, stakeholders typically need different digital representations of the asset, such as an "as-designed" state or an "as-built" state. The data describing these representations can be modeled product data, geometric models, manuals, bills of materials, system-level models, or other representations of knowledge. Software for managing and serving each of these different types of data is usually available because it was developed with each type in mind. Examples are CAD programs or tools for working with system models. However, without digital continuity between the fragments, it is not possible to query across the multiple layers that occur. For example, determining which parts are within a certain distance of a specific component or querying all manuals related to an assembly often requires custom plug-ins in current PDM software, if possible at all.

One approach to face that is to base all data on the same generic data model and data management system, allowing a single query language to access the multiple semantic layers and allow queries to work on a single data sink. Common generic data models are relational and graph-based models. In relational systems, the referencing of information across a data frame, for example a set of entities, necessitates the execution of computationally and memory-intensive common operations. In contrast, in graph database systems, the interconnectivity of the entities is of equal importance to the entities themselves. Consequently, the conceptual representation of densely connected information is more straightforward in a graph than in, for instance, tabular representations. While the approach to handling information in that form is not new, many approaches and applications do this from the perspective of the designer or manufacturer and with the concomitant view of the occurring entities and relations. The special circumstances that come with the perspective of a third-party organization that is neither the designer, manufacturer nor the owner of the asset but is planning and performing its retrofit differs from that and needs to be considered. Because of this, in this article, we propose an approach to a graph-based, superordinated information management tailored for the retrofit of long-living assets that helps to bring the different available fragments, silos, and states together. It integrates modeled product and geometric data as well as metainformation and system models into an overarching data model. The system comprises

a central platform incorporating a variety of data handlers, each specializing in a specific data type, silo, or source, and a specifically tailored user interface that serves as the primary point of access for user interaction. In addition, data generated during product development, such as CAD models, and during product maintenance activities, such as 3D scans, can be managed within it. While CAD data usually provides an as-designed state, 3D scans represent the geometric as-built state of the object and its environment, e.g., used in retrofitting buildings (Sanhudo et al., 2018), ships (Rata and Secobeanu, 2019; Njaastad et al., 2022), and aircraft (Moenck et al., 2022a; Moenck and Schüppstuhl, 2024). The objective of the proposed solution is the creation of a system that provides engineers responsible for planning and implementing a retrofit with a unified access point to the necessary available information and serves the data from the different sources. Besides this assisted access to information, tools that directly utilize this information and assist with planning tasks, such as interference detection, are also incorporated.

This work is structured as follows: First, we present the fundamentals for the subsequent parts in Section 2. Subsequently, we introduce the characteristics of retrofitting an aircraft as the primary example of this approach in Section 3. This helps to contextualize the remaining work. Furthermore, we explain the process behind this research (Section 4) as well as the resulting proposed concept (Section 5) and a case study (Section 6). Finally, we discuss the results (Section 7) and conclude as well as present an outlook in Section 8.

2. Fundamentals

2.1. Data, information, and knowledge

Although the terms are often used synonymously, a clear differentiation and definition of the terminology are necessary when describing the handling of "data,", "information," and "knowledge." This sub-section will elaborate on these terms, their meanings, and the relationships between them.

As described by Geisler, *data* represents the entirety of facts stored within a file or database (Geisler, 2014). Data are immaterial entity that cannot be consumed (North, 2021) and typically serve as the foundation for communication, interpretation, or further processing (DIN - Deutsches Institut für Normung, 1995). It is typically stored in the form of formalized structures (Meyer, 1999).

Information is derived from the processing or interpretation of data within a particular context (Geisler, 2014) and for a specified purpose (Meyer, 1996). Descriptions of the characteristics, properties, and structures of specific assets or circumstances are, in essence, information (DIN - Deutsches Institut für Normung, 1995).

Knowledge can be conceived as the totality of facts available to individuals or groups of people, with the objective of facilitating the identification, analysis, and resolution of problems (Irlinger, 1999; Probst, 1993). Knowledge is necessary to transform data into information and produce new information from existing information.

Knowledge can be written down, documented, preserved, and communicated in different forms, from textual descriptions to visual formats such as structure or process diagrams, flow charts, mind maps, and hierarchical views. Similarly, ontologies enable the transparent documentation of the structure of knowledge within a subject area while also facilitating communication between people, people and machines, and even between machines (El-Haji, 2014). Gruber characterizes the commitment to ontologies as an agreement to utilize a shared vocabulary in a coherent and consistent manner (Gruber, 1995). Ontologies define a terminology, thereby ensuring that the same term is interpreted in the same way and is cognitively connected to the same concept (Lehner, 2021). In addition to defining the terminology itself, they also define and describe the relations between terms and elements. Committing "to a common ontology guarantees consistency, even if the representation does not claim completeness" (Gruber, 1995). Because of this essential role, there are approaches for creating comprehensive ontologies like the Basic Formal Ontology (Arp et al., 2015) and standards like the OWL - Web Ontology Language (Bechhofer, 2018).

2.2. *Current data handling of products*

Different information is stored in different formats, such as different file formats and their respective modeling approaches, which require individual handling methods. In today's engineering practice, geometric information plays the most important role. However, with the current advances in systems engineering and its move towards modeled documentation of information, namely model-based systems engineering, system models that define semantic information in particular are playing an increasingly important role. At the same time, the ever-increasing volume of information and the equally increasing need to process this mass of information are met by applying techniques from the field of data sciences, such as storing information in the form of specialized databases, e.g., graph databases. This section provides insights into related and current techniques for handling product data.

2.2.1. *PDM in the concept of product lifecycle management*

With the move to digital product development and the emergence of digital files, the storage and handling of product information had to adapt. Initially, files were stored primarily on local hard drives, but the sharing requirements of collaborative projects led to files being stored on network drives and made available to a team of engineers. More sophisticated solutions than network file shares are PDM or other PLM systems and their respective approaches to managing product-specific data throughout the development or even the entire lifecycle (Stark, 2020). Commercial software companies typically use the term PLM system to describe a set of subproducts used to manage a product's data and information. We understand the term PLM as all the activities that manage the information or data of a product, and not only the system that stores the specifics of the mentioned activities. Thus, we use the term PDM only for the management of data and information; consequently, a PDM system is the specific software stack. In accordance with (Stark, 2020), other commonly used applications in the scope of PLM are, e.g., idea management, CAD, CAM, or complaint management tools. Therefore, the concept and activities of PLM are carried out by specialized applications that can work independently or link different phases of the lifecycle to a unified data and management structure, which may include a PDM system.

Typically, PDM systems target the management of product-related information in the form of engineering drawings, BOMs, and CAD models (Sendler, 2009). PDM systems focus primarily on processes and files related to product development, while the goal of PLM systems is to extend this scope and manage the entire product lifecycle, ideally starting from the first idea and ending with recycling (Stark, 2020). The basis of the information structure is usually the part that is the smallest unit in the BOM of the (future) product. For modular products, there is a generalized product structure, the so-called 150-percent BOM, to which individual variants and their BOMs with varying or optional positions can be mapped (Stjepandić et al., 2015). Although the goal is to provide information across domains, the stored product models usually represent a specialized and domain-specific view of the product (Muggeo and Pfenning, 2015). In these systems, interdisciplinary data handling, connecting, and presenting the stored information on an abstract level is usually missing (Muggeo and Pfenning, 2015).

Managing the logic, functions, and requirements of the product at a detailed level would require a complementary addition to current PLM systems (Weilkiens and Sendler, 2013). The same can be said for other types of relationships or metainformation, such as dependencies between entities in the systems or linkages, such as dependencies between separate documents. Managing the detailed geometry of the product is also not part of common PLM systems, as they focus on files that act as containers of information, and the geometry itself is handled by the respective authoring tools (Nzetchou et al., 2019).

Although PDM and PLM systems are widely used, they typically focus on product development or lifecycle from the product creator's perspective. Ríos et al. (2015) describe that "current PLM systems treat the product as a type of product and not as a [unique digital instance] that could have a one-to-one link with its corresponding physical one." Thus, they do not follow each individual produced entity along its life but manage or define the overarching idea of how the products should be treated during their life. The current trend of concepts that focus on individual instances, then often called Digital Twins (DT), is described in the following section.

2.2.2. Models and DTs

Models can come in many forms and applications and in a variety of domains. Although often defined similarly, the term 'model' itself is understood in different ways. In engineering and product development, it is often associated with virtual representations of geometric information or CAD and simulation models of (future) products. These models are often stored in the PDM/PLM systems mentioned above. In addition to these geometric representations, there are models of systems or semantics of products that focus on aspects other than the geometry or assembly of products (Muggeo and Pfenning, 2015). Although these models are most often created from the perspective of product creation, which represents an ideal view of the product "as planned" or "as designed," there are approaches that aim to connect these models to the actual produced product entities in the real world. These concepts that connect the real world with the digital space are often referred to as "Digital Twin" (s. Figure 1). One of the first detailed descriptions of this concept was published by Grieves et al. in the early 2000s (Grieves, 2019). Since then, research and industrial interest in DTs has multiplied in recent years, and the concept has been described for a number of applications targeting different domains. In the context of industrial, e.g., prevalent mechanical or electrical products, approaches focus on sensory or state-describing information and less on geometric or product-describing information (Laukotka and Krause, 2023).

In the context of long-living assets, such as plants or buildings, approaches often focus on twinning an already existing product to perform or further digitally supported activities, such as retrofitting as described for example by (Esteves et al., 2023). For nonstationary long-lived assets, such as aircraft (see Section 3), the implementation of a DT is not trivial, especially for a third party, as the asset may move around the world and face a variety of stakeholders. Existing approaches, e.g., Singh et al. (2021) and Moenck et al. (2024), address this application from the perspective of manufacturing, production, or maintaining the asset, which is different from the retrofit perspective considered here.

2.2.3. 3D scanning in as-is state generation

In the realm of long-living assets, digital replication, also known as twinning, is often necessary when the current "as-built" state of the asset is not digitally available. For example, in the context of Building
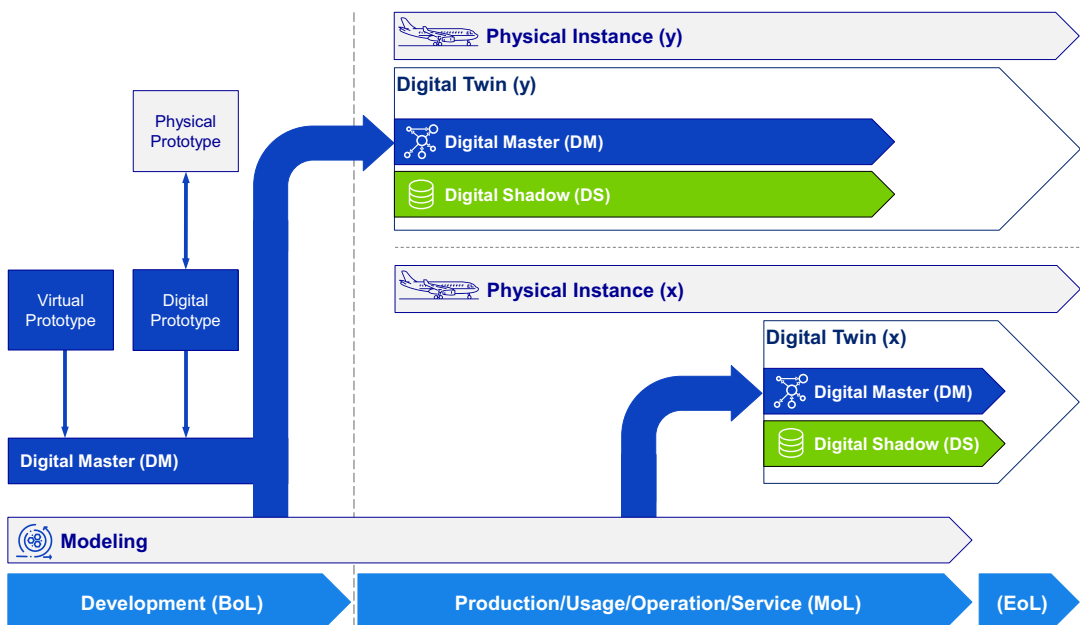

Figure 1. *The lifecycle of DT instantiated beginning with the start of the usage and later Mid-of-Life phase, based on (Moenck et al., 2022b).*

Information Modeling (BIM) (Jiang et al., 2021; Stojanovic, 2021), industrial plant retrofits (Sommer et al., 2019, 2021; Stjepandić et al., 2022), or offsite assembly (Rausch et al., 2021), there is a plethora of methodologies based primarily on geometric representations. If available, "as-designed" 3D models are sufficient for these activities, but even then, there is usually a discrepancy between the planned, built, and (after years) latest state. In most scenarios, a combination of as-designed 3D models and newly acquired data is required (Sommer et al., 2019; Stojanovic, 2021). Typically, different sensor modalities are used in the digitization process, with terrestrial laser scanners (TLSs) being the most common. Newer systems not only generate laser data but are also equipped with an RGB camera, which facilitates the addition of color information to the resulting point clouds. TLSs provide an accuracy of less than or a few tens of millimeters, which is sufficient for most design and engineering tasks. In cases where greater accuracy is required, tools such as a *theodolite* can be used for more precise measurements, or alternatively, local measurements can be augmented with systems of higher accuracy but typically smaller measurement volumes.

Once the raw data is collected, it can be used in a manual process with minimal postprocessing, such as noise reduction or segmentation of planar surfaces, such as walls. Full panoptic segmentation, where each measured 3D point is mapped to an object instance, is still a topic of current research—approaches originate from the field of computer vision. Here, learning-based methods show significantly superior generalization capabilities but require training data that includes both the object and the sensor domain. Publicly available datasets for scenes with everyday objects and common sensor systems cover a variety of use cases. However, object recognition or segmentation in the absence of training data remains a research frontier and is the focus of one-shot and zero-shot learning approaches. In practical applications, 3D perception often leads to the extraction of the most critical objects of interest using existing technologies, while the rest is handled by "click workers." However, off-the-shelf software provides a set of tools to facilitate manual processes in the remodeling of a scene (Son et al., 2015). Furthermore, integrating these elements into the models and formats of a specific CAD system is predominantly a manual process. 3D scans and their potential additional semantic information fall into the category of geometric models, which represent a component of the as-built state of the physical asset.

2.2.4. Models for product data

"As-designed" data comes in the form of models, sometimes created during the design phase using CAD. They serve as virtual representations of products, detailing geometric information, and are typically integral to both design and subsequent stages, such as simulation. The utility of CAD models extends beyond mere physical representation; they are essential for simulating and understanding a product's behavior under various conditions. This rich representation helps designers and engineers anticipate potential problems and effectively iterate designs. CAD models in PLM provide a comprehensive framework for managing the lifecycle of a product from conception to disposal, particularly its geometric representation. In addition to proprietary software stack-specific models from the few major CAx companies, there are numerous neutral and open standardized exchange formats and modeling standards for product data, such as IGES, JT, or STEP. The vendor- and company-independent exchange of "as-designed" data through neutral formats such as the Standard for the Exchange of Product Model Data (STEP) aims to facilitate seamless data exchange across different software and platforms (Kemmerer, 1999). STEP, an ISO 10303 standard, supports the interoperability of product data, including geometric, material, and processing information. The AP 242 (International Organization for Standardisation, 2022) (third edition published in 2022) is an application protocol for managing product data, covering not only typical PDM requirements such as BOM management but also model-based product development, long-term archiving, supply chain integration, product manufacturing information (PMI), and many others. For the remainder of this work, we will use STEP AP242 as a modeling standard for product data.

2.3. System modeling, as originated from model-based systems engineering

The current rise in the application of system modeling techniques has its roots in systems engineering and its aspect of moving from document-centric to model-centric documentation. This, then called Model-

Based Systems Engineering (MBSE), supports the tasks of the systems engineering methodology as described by INCOSE (Walden et al., 2015) with "the formalized application of modeling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later lifecycle phases" (INCOSE, 2007). The approach of handling information in centralized system models, often in the form of *SysML* models (Williams and Boing, 2021; NoMagic Inc., 2011), is usually supported by specialized authoring tools that provide a graphical user interface for model creation and allow for easy analysis of the modeled elements and relationships across different views within the model (NoMagic Inc., 2011). These different views and diagrams can focus on different aspects of the same system. Because of the single-source-of-truth management, often included in the authoring tools, the information of identical elements that occur multiple times across the different views and diagrams are automatically synchronized. "When you modify an element on a diagram within a modeling tool, you are actually modifying the element itself in the underlying model" (Delligatti, 2014). This systematic process greatly eases the definition of information about comprehensive assets, as information can be distinctly modeled in different diagrams, allowing each of them to stay overseeable and comprehensible.

This approach can be applied to a range of applications other than systems engineering, although in this case, it is not strictly speaking to be called MBSE (Berschik et al., 2023). However, the fundamentals of system modeling, as also presented by Delligatti (Delligatti, 2014), can be generalized, adapted, and finally tailored to the specific application. The resulting models are then based on a clear definition and methodology, which also allows for coherent model creation even among engineers and for projects that need to grow over the years (Laukotka et al., 2024). This allows engineers to systematically create the information in digital form using the system models, while the authoring systems provide a visual user interface and also manage the elements, allowing for a single source of truth even across different views within the model. Thus, this approach is practical for engineers without programming skills or other, perhaps more sophisticated, ways of defining information digitally. Although the creation of information is greatly facilitated by the authoring tools, as of now and with the current revision of the *SysML* modeling language, interaction with the modeled information from outside the authoring tool and by other programs or processes is very limited. To access the information, the authoring tools must either be actively opened and used, or selected information from the model must be exported in tabular form to a neutral file format, e.g., CSV, which can then be used by external tools (Laukotka and Krause, 2024).

As mentioned above, systems modeling is nowadays used for a variety of applications, even outside the origins of typical systems engineering tasks. In terms of relevant application to aviation, it can already be found focusing on different aspects: (Fuchs et al., 2022) use system models to support the conceptualization of aircraft cabins in the early stages of cabin design, (Hanna et al., 2021) support the handling of information during the design of modular lightweight components for aircraft cabins, and (Singh et al., 2021) base the model layer of their DT information management framework for aircraft manufacturing on system models. Each of these works focuses on their individual application and circumstances, but mostly from the perspective of the designer or manufacturer. The unique circumstances of a retrofit (also see Section 3) are rarely addressed.

2.4. Graph databases

Although the tools and methodologies mentioned above originated in systems engineering and MBSE and, thus, are tailored to the engineering tasks occurring there, the challenge of managing an increasing amount of information is present in many domains. Today, applications can store almost all the information and data that are generated and accumulated during any activity, which can be stored and later reused in intelligent applications. Data and processing resources are relatively cheap. Tables are one of the most intuitive ways to collect and present information (Meier and Kaufmann, 2019), so using relational data structures is common for various applications. Although they are a good solution for handling structured data (Meier and Kaufmann, 2019), managing relationships between entities in relational databases is a complex and resource-intensive task (Fasel and Meier, 2016) compared to other approaches.

Graph databases, based on the mathematical foundation of graph theory (Edlich et al., 2011), consisting of nodes, representing entities and edges, representing the relationship between them (Kastens, 2008). In addition, edges and nodes can be further described by adding attributes and labels (Meier and Kaufmann, 2019). This structure makes them particularly well suited for defining structures as relationships or dependencies. Furthermore, graph databases support a variety of graph theory algorithms, such as search, graph cutting, or clustering methods. Existing Database Management Systems (DBMS) have various user interfaces and application programming interfaces (APIs) to interact with the data and perform manipulation operations. However, the user interfaces are typically not domain-specific, which means they are not directly suited for, e.g., managing system models from an end-user perspective.

3. Aircraft: a complex and long-living asset

Aviation is characterized by a number of specialties: With a typical number of parts of around 400,000 (Stark, 2022) and a lifespan of up to 20 years (Niţă and Scholz, 2011) or even more, aircraft can be described as one of the most complex products built today. Similarly, they encounter a variety of different stakeholders throughout their lives. High regulatory requirements regarding their certification (de Florio, 2011) and ongoing rigorous maintenance measures ensure high safety standards (Hall et al., 2013).

Aircraft are designed as product families to simplify complexity. For example, the A320 family includes the A318, A319, A320, and A321, which differ primarily in length. Airbus also has the A330, A340, A350, and A380 families. These families share most features, simplifying pilot training, maintenance, and spare parts management. Variance within these families is similar to other product families, although on a smaller scale compared to mass products such as cars. Each variant, with identical designs, is produced multiple times and delivered to airlines.

Regular maintenance, based not only on flight hours but also on idle time, ensures the airworthiness of the aircraft. In addition, airlines or aircraft operators regularly request modifications to the consumer product cabin, which we summarize as cabin retrofit, where the cabin or parts of it are replaced with new components (Deneke et al., 2019). This allows airlines to adapt to changing customer needs or (Altfeld, 2010) and maintain the condition of the daily used cabin, e.g., by replacing worn components (Niţă and Scholz, 2011). However, planning and executing maintenance and especially modification tasks requires a solid knowledge of the actual state of the aircraft. Despite regulatory requirements for documentation, the resulting documents and files are typically created separately for each task performed and for each stakeholder or domain within the aircraft (Altfeld, 2010). The information may be available, but it is spread across multiple documents, revisions, and systems, resulting in a fragmentation of information that makes it difficult to derive a comprehensive view of the current state of the aircraft.

With each change to the aircraft due to maintenance, the life of the asset is extended, but the asset also becomes more and more divergent from the original "as-planned" or "as-built" state, as well as from the other instances that were once created identically. In addition to the variance of the product family, there is now deviance due to ongoing adjustments and modifications. During these modifications, appropriate documentation of the operations performed and changes made to the product must also be created. These required revisions to the aircraft documentation only increase the fragmentation of information described above. Manually merging this fragmented information, especially from third parties not involved in manufacturing but in retrofit planning, requires significant effort. Figure 2 visualizes this variance and deviance of long-living assets and the associated fragmentation of documentation.

3.1. Taxonomies and documentations

Based on the architecture of commercial aircraft, Jackson (1997) presented a generic aircraft system architecture. Such a generic system architecture was possible because there are standard elements in most
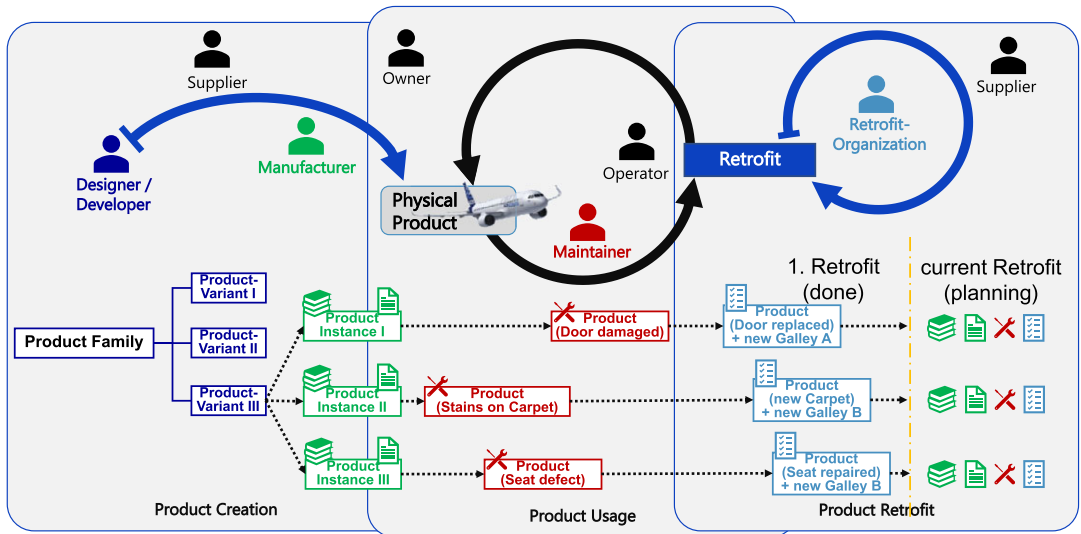
Figure 2. *The occurring variance and deviance in long-living assets requires handling of semantic information, based on (Berschik et al., 2024).*

commercial aircraft, which may appear in different detailed realizations for specific aircraft. In 1965, the Air Transport Association (ATA) established the ATA Chapters, a naming and referencing scheme to support the classification and organization of information (FAA - Federal Aviation Administration, 2008) and somewhat face the challenge of fragmentation described above. Today, these ATA Chapters are used throughout the industry and have become the standard for commercial aircraft documentation (Mensen, 2013). Typically, the systems and components related to these ATA Chapters are developed more or less independently (Altfeld, 2010). This includes design, documentation, and often even installation, which are performed by separate teams, resulting in the aforementioned fragmentation of information. The aircraft and cabin dependencies and requirements are typically documented in many separate documents, but they are usually labeled in the ATA Chapter taxonomy, which allows for easy attribution to specific systems.

To name a few, even if they focus on the same area of the aircraft, there are separate documents to describe the water/wastewater system (*ATA 38*), electrical power (*ATA 24*), and air conditioning (*ATA 21*). Although these documents generally provide guidance on the general or exact location within the aircraft, they usually do not refer to documents from other domains. Some of these documents are generic for the aircraft type, e.g., the *A320 Family Maintenance Manual (AMM)* (AIRBUS S.A.S., 2016), the *Illustrated Parts Catalog (IPC)* (AIRBUS S.A.S., 2002b), or the *A320 Ground Operations Manual* (AIRBUS S.A.S., 2002b). Others are specific to the aircraft itself, such as the *Layout of Passenger Accommodations (LOPA)* (AIRBUS S.A.S., 2003), which visualizes the cabin layout of that particular aircraft (instance). After initial production, detailed documents such as *Wiring Diagram Manuals (WDM)* may be available from the manufacturer as part of *Aircraft Schematic Manuals (ASM)* (AIRBUS S.A.S., 2002a) in their first revision, but with each operation, maintenance, or modification, changes to the aircraft may occur.

Although generic documents such as AMM, GOM, and IPC may not be affected, the same cannot be said for more specific documents such as LOPAs and WDMs. These documents must be revised accordingly, rendering their original versions obsolete (Niţă and Scholz, 2009). Therefore, existing documents cannot be modified, but new revisions must be created. This allows time-referenced documentation of the changes and status of the aircraft and facilitates certification because the changes made are clearly documented. However, it hinders the easy creation of a comprehensive representation because it also increases the fragmentation of information across even more documents. The representation of

information is tailored to the individual certification of a change, rather than to a later available, comprehensive, linked as-is state. As a result, third-party stakeholders who are not the original designer or manufacturer typically must manually research and access these relationships and documents. Due to standardization and the fact that aircraft are quite similar, some information can be classified as generic and applicable to all aircraft of the same type, while others are specific to a single aircraft and differ even between once identically built assets. While this can help classify information and improve its reusability, it is rarely implemented in an automated or digitally supported way today.

3.2. Cabin design and retrofit

The cabin of a commercial aircraft is the point of interaction between passengers and the transportation service, including the aircraft. Therefore, it is the main differentiator for airlines and is specifically tailored to their strategies (Giesecke, 2005; Hall et al., 2013). However, this tailoring has to be done within strict limits, because on the one hand, the space is very limited and clearly defined by the aircraft structure of the aircraft type, and on the other hand strict regulatory requirements have to be met (de Florio, 2011). Another result is that the available elements within an aircraft cabin are more or less standardized across different airlines and aircraft types. Most cabin elements are present in almost all commercial passenger aircraft (Kopisch and Günter, 1992; Niță, 2012), varying only in their specific detailed implementation. Typical elements are *Seats*, *Overhead Stowage Compartment (OHSC)*, *Partitions*, or *Class Dividers*, and the larger so-called monuments such as *Galleys* (aircraft kitchens) and *Lavatories* (aircraft toilets). In order to appeal to different types of passengers and at the same time provide them with a comparable travel experience, largely independent of the exact aircraft, different classes have evolved within this standard (Kopisch and Günter, 1992; Mensen, 2013).

Because most of these components must be connected to aircraft infrastructure systems such as plumbing or electrical circuits, and because there are distinct structural mounting points, especially for the larger components, there are many dependencies to consider when designing these cabins. The frames, vertical structural members on the fuselage, are one of the key attachment points for many of these components. Because they are standardized for all Airbus A320 aircraft and are located at specific coordinates within the aircraft, they can also serve as a means of broad localization. Knowing which frame a component is mounted to can be used to determine the coordinates of that frame, and thus the location of the component in the aircraft. Airframe-specific interior panels are also largely standardized. Furthermore, regulations define the required number of monuments based on the number of passengers, given by the number of seats, and the possible positions of these monuments depend on the structural and infrastructure connections provided by the aircraft type. All this leads to a situation where "many disciplines relate to one another and many rules have to be accounted for, in order to optimize all the parameters involved in this task" (Niță and Scholz, 2009). To support this process, there are knowledge-based software solutions such as *Pacelab Cabin* (PACE Aerospace Engineering and Information Technology GmbH, 2023) or approaches that also use virtual reality systems (Fuchs et al., 2022).

Most of these solutions and approaches focus on early cabin concepts and designs or target the initial cabin layout of the aircraft during production. However, retrofit cabin design faces additional challenges because the cabin is being developed for an existing airframe that has undergone numerous maintenance and modification activities. The retrofit of such aircraft cabins can vary in scope. Some retrofits may only replace the seats with some additional cleaning and maintenance, while others may replace all major components with new ones (see, e.g., Figure 2). The history of the aircraft and its previous maintenance and retrofits lead to a state of the aircraft that does not reflect the original state known during the production and installation of the original cabin (Laukotka and Krause, 2023). Thus, this additional layer of dependencies that occurs when developing a new cabin for an existing aircraft requires additional approaches that also facilitate the actual state of the specific aircraft. This makes aircraft a prime example of said long-living assets and the challenges of handling their specific information.

3.3. Multimodel handling of data and related work

With the increasing amount of information and data that needs to be handled today, and the additional move to include semantic and relational information in data handling, new approaches have emerged. These combine different sources and methods or approaches to multimodel data handling. While some information is stored in a simple database and others in file storage, additional information is now made available using new approaches and new methods. In addition to the application presented in this paper, this can provide benefits for a number of other applications and domains. A brief excerpt of related work and their approaches to data and information handling is presented below. The use of system models, e.g., in the form of *SysML* models or graph databases, were identified as approaches. In the engineering design community, the *SysML* modeling language can be found in many applications, methodologies, and implementations (Berschik et al., 2023). Likewise, more and more industries are implementing system modeling techniques as part of research projects or in smaller projects for testing purposes.

Nonrelational data handling has been a driving trend since the concept of NoSQL emerged. structured query language (SQL) has grown to a standard query language for two-dimensional relational databases, which were predominant in the early stages of databases. NoSQL name databases which do not follow a relational-based approach, e.g., including concepts like document-oriented, graph-based, key-value, multivalue, object, or data stream databases. Multimodel databases are able to combine multiple modeling aspects, while the boundaries are fluid, meanwhile. The management of handling data in the context of DT in a multimodel manner is already proven and, e.g., a methodical way of modeling is described in (Koch et al., 2022).

The approaches of utilizing system models can also be found applied to other applications within aviation: Fuchs et al. (2022) describe an approach to support engineering during the initial design of an aircraft cabin, focusing on the early phases of the cabin and aircraft life. Hanna et al. (2021) use system models and *SysML* to handle interdependencies between information on the conflicting tasks of modularization and weight reduction of aircraft components. Berschik et al. (2022) have shown that a holistic aircraft cabin metamodel based on *SysML* can be used to link development data in different phases of the product lifecycle. For this purpose, not only the product development data were modeled but also the overall assembly planning data were linked. Aspects such as the service and communication platform of a cabin and its cybersecurity were also integrated. In previous work, the approach of using system models to handle information relevant to the retrofit of the aircraft cabin was introduced (Laukotka et al., 2021). Furthermore, to improve access to the modeled information, the general concept of combining these system models with graph databases was recently presented (Laukotka and Krause, 2024).

4. Research method

The background to this research stems from previous research collaborations with aviation industry partners where a need for improved information and knowledge management was identified. The typical research process was then followed, including a more detailed analysis of the current situation, an evaluation of existing approaches, the creation of a customized concept, and, finally, its application to a case study (see Figure 3). Because of its origins in an already specific application area, this research tends to stay close to that application and the given scenario. However, as some aspects may be applicable to other contexts, they are described in a neutral way before being applied to the given scenario. A specific exploration of their applicability to other scenarios would, however, go beyond the scope of this work. This section describes the overall process of the research conducted and its specific particularities before the findings and results are described in the following sections.

During a research project focusing on improving the retrofit by providing the engineers with more detailed information, e.g., by utilizing 3D scanning technology, the demand to improve the data handling

Figure 3. *The key steps of the process resulting in the presented approach.*

altogether became apparent. Subsequent interviews with experts in the field of aircraft cabin retrofit and process monitoring were used to analyze the current state of information acquisition, handling, and access for new cabin design and retrofit in more detail. During that, the many occurring individual silos of information and the resulting fragmentation were identified as a major challenge. Workshops were held with these experts to identify potential improvements and define requirements. An example of a nonfunctional requirement identified is the need to consider intellectual property requirements within the research, leading to a focus on solutions that can be deployed on-premise. A literature review was conducted to identify potential approaches and methods that address similar challenges and are already used in other applications and domains. The identified approaches or relevant aspects were documented and evaluated concerning their potential adaptation to aircraft cabin retrofits. Based on this, a customized approach was created to handle the information occurring and required in this application. The respective concept is presented throughout Section 5. Finally, it is validated using a representative case study based on real aircraft components and information, as well as workshops with industry experts, which is described in Section 6.

While the general idea (Moenck et al., 2022a) and some aspects of the results (Moenck et al., 2022b; Laukotka and Krause, 2023) of this approach have already been presented; this article focuses on the important contribution of the use of graph databases as a key component of the resulting platform. Thus, this article can be seen as the overarching concept that connects all the different dots and provides a traversing representation of the application and scenario, with the graph database and its surrounding platform created as the essential tool. In addition, this article includes the application of the entire platform to the case study, which has not been done to this extent before. Various aspects of the first five steps, such as the identification of the need, the essentials resulting from the analysis of the current state, or the relevant approaches resulting from the literature review, have already been presented in the previous sections. The following sections describe the developed approach, and the case study used to test and validate the concept.

5. Concept

Long-living assets, especially more complex products or systems such as aircraft, require a wide variety of information during the processes that occur regularly throughout their life. Likewise, during these processes, new knowledge and information of various types is produced, often specific to the particular asset being managed. As a result, the information accumulated about a particular asset may come from different sources, take different forms, and require unique forms and handling practices. A neutral perspective is required when focusing on applications where a variety of different stakeholders are involved, and many aftermarket processes are performed by third parties rather than the product designer or manufacturer. Because of this diversity, information is typically not stored on a single monolithic platform. Instead, the use of individual storage solutions for different types of data and information is a different approach that requires an overarching management system. Rather than relying on proprietary PLM/PDM systems that may fully, partially, or with little implementation effort meet the above requirements, we propose a complementary superordinate management system that brings the fragments together and can also access the information within proprietary PLM/PDM systems or other sources as needed. Here, an asset's longevity is a challenge and an opportunity, as data availability may be limited, even for the manufacturer, but the aftermarket parties have accumulated a tremendous amount of knowledge over the years that can be leveraged. To exploit this knowledge, it must first be digitally accessible. The goal is not to create yet another place to store the same information but to provide a superordinate layer that makes sense of the fragments, provides easier access to the underlying information and documents as well as to provide digital assistance or services that build on that now digitally available information.

5.1. Platform

This section describes the platform as part of the concept in more detail, broken down into its core elements, before applying it to a case study in Section 6. The realization of the individual platform components with current technology selection, e.g., as in the case of choosing a specific database, is part
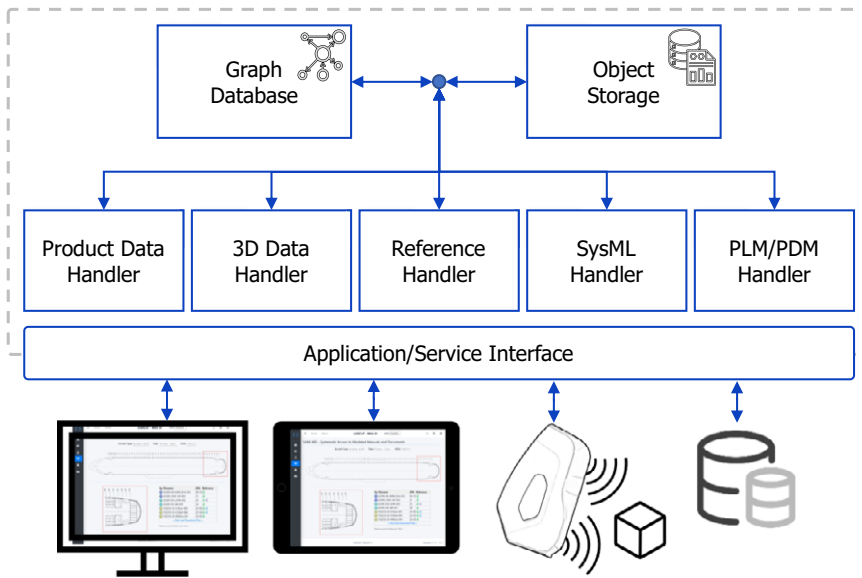
Figure 4. *The core elements of the data handling platform, based on (Moenck et al., 2022a).*

Table 1. *The given entries name examples, which we refer to in this work*

Geometric		Semantic Layer	
As-designed	As-Is	As-designed	As-Is
3D models (BREPs)	3D-Scans	Product Data	Product descriptions
…	…	…	…
Descriptive		**Document-/Paper-based**	
As-designed	*As-Is*	*As-designed*	*As-Is*
Product structures	Annotations	Manuals	Layout Sketches (LOPA)
…	Warnings	Technical drawings	Maintenance Job-Cards
	Maintenance hints	Interface specifications	…
	…	…	

of our prototypical implementation, given the available state-of-the-art technology. Figure 4 visualizes these core elements of the proposed superordinate data handling and information management platform, with the graph database as one major element to combine the different data sources and types.

5.1.1. Overview on data and information

In the previous sections, we have already outlined different data and information from sources that we want to support and incorporate. We start by differentiating between data from the early life phases that represent some kind of "as-planned" or "as-designed" data, while data from the later life phases differ and describe the product in its current state.

Based on the various data types to be supported, the following constraints were formulated for the data and information management system:

1. No generic data model is capable of handling all of the types in Table 1. As we have already outlined, compared to a relational-based model, a **graph-based model** allows for more flexibility,

   adaptive changes, and efficiency during the modeling of the various individual data and information.
2. The indexing of data in a database is directly related to the overall performance of queries. As an example, trying to query 3D data in a relational database in which rows are simply uniquely indexed by an *id* is rather inefficient since all data have to be processed to derive the specific results. Instead, if the data are, e.g., ordered and indexed in a spatial tree, only parts of the data have to be processed. As outlined in Table 1, most of the data and information have spatial characteristics, which is why we formulate the constraint that the platform must support **spatial indexing and support queries**.

5.1.2. Data sources

The data that needs to be handled during the planning and execution of retrofits can originate from a variety of sources and may be in different forms. Typically, data sources for generic descriptions of the asset are its standard handbooks or manuals and common knowledge of the respective domain. Nevertheless, this information must be digitally processable in order to facilitate its efficient utilization. More specific data sources include documentation of past modifications of the asset or freshly acquired information such as measurements and photographs. These sources may take the form of drawings or schematics in PDF format, texts or images, or 3D CAD data in the form of parts, assemblies, or even individual 3D scans of a specific asset. As previously discussed, this data can be sourced from a range of systems, with simple file systems or sophisticated PLM systems being just two typical examples.

5.1.3. Data sinks

During the selection of appropriate data sinks, specifically DBMS (Database Management Systems), up to the date of this publication, we discovered that no Multi-Model Database Management System (MMDBMS) directly supports storing, handling, and provisioning of all the diverse data types as formulated in Table 1, especially supporting spatial indexing. Therefore, we had to build a solution similar to a Multi-Model Multi-Database Management System (MMMDBMS), i.e., a combination of different DBMS. The choice for the graph-based DMS was *Neo4j*, since it supports both graph-modeled data and spatial indexing and querying, while *Min.IO* was chosen for Binary Large Object (BLOB) storage. All unstructured data is simply stored as BLOBs and referenced in the graph. We developed a web-based higher-level management system that feeds and independently reads from the two data sinks while providing the necessary Create, Read, Update, and Delete (CRUD) operations through a RESTful API. As shown in Figure 4, we then implemented five different functional components: *Product Data Handler*, *3D Data Handler*, *Referenced Data Handler*, *SysML Handler*, and *PLM/PDM Handler*, which handle individual data and information. Except for the *PLM/PDM Handler*, the individual characteristics of these functional components will be outlined in the following sections. A connection to an industry partner's PLM system (*Windchill*) was introduced in the course of this research. The use of a custom API enables linking to projects and specific CAD or PDF files stored in the PLM system from within the platform. Further details of this integration cannot be presented due to various proprietary and security restrictions.

5.1.4. Ontology for semantic information

Independent of the individual data sources and sinks while focusing on the handling of semantic information, a simplified ontology is created specifically for the concept and platform. This is particularly important for semantic information, as it needs to accommodate greater flexibility in terminology and interpretation. Focusing on the perspective of aircraft design in general (Ast et al., 2013) presented an approach and their experiences of creating such an ontology. Similarly, Franzén et al. (2022) describe an approach for an ontology-assisted aircraft concept generation and (Arista et al., 2023) an ontology-based engineering system to support aircraft manufacturing system design. The three related works, which already address the relevant domain of aviation, offer distinctive insights into their three distinct applications: general design, concept generation, and the manufacturing process. Each of these works

offers valuable insights and perspectives into the structures and terminology that occur within their respective field. However, due to the differing perspectives from which they are derived, they lack crucial aspects that are relevant to the cabin retrofit and cannot be readily applied to this scenario. Thus, and in order to establish such a systematic foundation for the distinct scenario and circumstances of the cabin retrofit, it is important to first analyze the context of the real-world scenario and asset in question in order to identify its essential information as well as the specific established structures and characteristics. The results are then abstracted to create a specifically tailored ontology (see Section 2.1). This will then be used as a basis for defining semantic data. Furthermore, it allows the different elements of the concept that process the same data to work together while maintaining their individual development and creation. In handling the semantic relationships that occur, the basic structure of the system models, as well as the corresponding graph database, is defined. Further details, as well as the combination of system models and graph database, are described in this section and in the case study in Section 6.

5.1.5. Product data handling

Typically, all PDM software has its own closed-loop modeling. Sharing data between solutions requires a neutral exchange format, e.g., STEP, as we have already outlined in the previous sections. A variety of approaches demonstrate that the use of STEP-modeled data for handling product data can go beyond only exchanging files. As an example, OntoSTEP translates the semantics defined in APs and STEP-modeled data into ontologies and knowledge graphs, respectively (Barbau et al., 2012; Kwon et al., 2022). OntoSTEP is used by, e.g., (Kwon et al., 2020) to integrate as-designed and as-inspected data across lifecycle phases. The ontology conversion facilitates activities like knowledge extraction and inference. However, the ontology, as can be derived from an AP, defining typical mechanical part and assembly concepts, is not of further interest in the scope of the targeted use cases since it does not contain specific domain knowledge. To support the storage of product data, in this work, we align with our previous work, where we developed a system to manage STEP-modeled data in a graph database (Moenck et al., 2023). The system, called GraphSTEP, provisions graph-modeled STEP data, and we implemented functional components to allow concurrent access to these representations. GraphSTEP itself is Application Protocol (AP)-independent. However, in the current implementation, we support the manipulation of assemblies, i.e., changing part positions and adding/removing components, according to AP 242. The connection of the STEP entities to the higher-level management graph follows a simple edge connection to the assembly root element.

5.1.6. 3D data handling

The task of the *3D Data Handler* is to import, export, and provide 3D scans and all other geometric data. This includes enabling spatial query functions for subsequent services. *Neo4j* allows spatial queries in the form of a distance query function based on points. In combination with 3D data, e.g., to query all objects within a certain radius, the objects modeled as points must be replaced. We achieve this by replacing surface meshes with density sampling and point clouds with voxel downsampling. This reduces the amount of data while enabling spatial query functions. The resolution of the downsampling determines the possible resolution of the queries.

5.1.7. SysML handling

To digitally define the relevant occurring semantic information of the asset, system models based on the modeling language *SysML* are employed. Their creation is greatly assisted by the authoring tools *Cameo Systems Modeler*, which provides a reasonable way to create the models, including the intrinsic single-source-of-truth management visually. As described in Section 2.3, this allows for the systematic digital definition of even comprehensive systems. While this requires knowledge of system modeling, this is done visually using the authoring tools and does not require any further programming knowledge. Currently, system models are increasingly being used in many engineering tasks, including those in the aerospace industry. The creation and use of these models are becoming more widespread, although not

ubiquitous. However, it provides a reasonable way to make the semantic information digitally available and processable.

To further assist the creation and improve the reusability of models and information, the systematic modeling approach used here (Laukotka and Krause, 2023) bases the models on a systematically created and well-defined meta-model, which in turn is derived from the ontology (see Section 5.1.4). Such a systematic approach to modeling is essential for large projects to enable consistent modeling across teams, projects, and years, as described above. The meta-model provides the basic structure of the system models. The aim is not to redefine all available information but to create an abstract representation of the specific asset. It is defined as which components and systems are generically available and can be instantiated for a specific given asset. Details of the individual entities are then made available by referencing information and files available in other parts of the platform from the abstract representation. The focus is, therefore, on structure, similar to a product structure, and on relationships as components that are "connected to" a system or "mounted to" another component. Likewise, references to given structures, such as naming conventions and the aforementioned links to more detailed information, are explicitly part of the system models. Because of the approach's underlying system models, the level of detail in which this is done can vary but is generally kept as simple as possible. The aim is to provide a digital representation that helps to navigate the available information, both semantically and geometrically, and ultimately leads to the specific files containing all the required information at the required level of detail. Complex products and large datasets, such as those found in long-living assets, go beyond the standard capabilities of BOM-based PLM systems and are, therefore, explicitly added to the platform as a complementary element.

Although the creation of system models using specific authoring tools and their graphical user interfaces is convenient, the ability for digital services to directly access the modeled information is currently a challenge. In the future, it may be possible to improve direct access to modeled information using the new *SysML v2* (Bajaj et al., 2022). However, this is currently only possible in prototype implementations and is not yet included in typical authoring systems. Therefore, the *SysML* handling transfers the relevant information from the visually created system models to the graph databases, allowing better access to the information and linking the modeled semantic information to the other information handled by the platform. To achieve this, the export function of the authoring tool is used. Although it is limited to tabular information that can only contain data that can be aggregated within the authoring tool, it also allows a clear selection of the required data by creating specific diagrams such as generic tables of elements (as blocks and instances) and generic tables of relationships or dependency matrices. The meta-model and ontology, created as the systematic base that defines the framework for the system modeling, result in additional elements and relationships. For the targeted processing, only the instances of the elements and their relationships that have been created as part of the definition of the semantic information are necessary, while the elements of the meta-model and ontology are not necessarily required. The datasets to be exported can be easily assembled in the authoring tool using the generic tables and respective filtering. The resulting tables of element and relation instances are exported as standardized CSV files and then injected into the graph database (Laukotka and Krause, 2024) using this *SysML* handler. First, the instances of the elements are parsed from the CSV files, added to the graph database as nodes, and stored in an internal list together with their respective element IDs. Then, the instances of the occurring relations are parsed from the CSV files, the start and end points are linked to the respective elements, and the resulting relation is added to the graph database, including all available attributes.

5.1.8. Handling of referenced objects
The *Reference Handler* spatially links various types of data by creating a reference to the object. The object is stored as a BLOB in *Min.IO*. The spatial linkage is called Point-of-Interest (POI) and, in addition to its position in space, also has an orientation. This is also used, e.g., to model the camera orientation when saving images.

5.2. Process for data and information mining

For this data-centric approach to information handling, it is important to consider not only the technological aspects of its implementation but also the methodical process of acquiring, storing, and accessing the information. A widely used and accepted process model is CRISP-DM (Huber et al., 2019) as presented by (Shearer, 2000). Since the steps generically described by them are similar to the required steps for the application of this approach, which we have already elaborated in previous works (Moenck et al., 2022b), CRISP-DM provides the methodological basis for the application and use of the presented platform, which is described in this section.

5.2.1. Acquiring relevant information

A good starting point for the acquisition of information about long-living assets is the general knowledge of the specific domains as well as standards or regulations that provide systematic structures. A detailed description of how these real-world structures can be systematically used as a basis for virtual descriptions and representations has recently been described in detail (Laukotka et al., 2024); the core idea is to reduce and abstract the identified semantics and structures of the assets to an ontology that describes and also defines the occurring essential elements and their relationships. The syntax of the digital descriptions and models, especially of the semantic information, is then based on the structure provided by the ontology. This ensures a clear structure and clarity of meaning, thus improving interoperability and compatibility. Although this is only the basis for new digital descriptions, the other goal of this step is to capture and filter the relevant information for the specific task at hand. For long-living assets, the number of documents and the amount of information can be extensive, as described and visualized in Figure 2. These assets, on the other hand, usually have some commonalities that, in combination with the systematically defined digital base described above, allow for some reusability of information; while some information may be applicable only to some specific instances of the assets, others are more generic and most likely identical for a group or even all assets. Since a holistic representation of the asset is not required every time, and the task of creating such a holistic representation from scratch on the very first iteration is tedious and likewise unnecessary, there are three important steps to gathering the relevant information. The first is to identify what information is essential and necessary to complete the task at hand by analyzing what information is already available because of previous tasks because it is common knowledge, or because the relevant documents are directly accessible. Second, the available information is collected or extracted from the available documents, and the missing information is recorded. When extracting information from documents, it is crucial to consider the level of detail. More extracted information provides direct access to more data, but it also increases the effort required for extraction and maintenance in the future. Alternatively, only essential information can be extracted, and references can be made to the original documents. At this point, these references are not directly integrated into the dataset but need to be considered. Close to the concept of DTs, the acquisition of this "as-is" information can be done in cooperation with its original manufacturer, based on possibly available maintenance records or using modern techniques such as 3D scans or smart image processing (Moenck et al., 2022b). The third step is the aggregation of the collected and recorded information into a dataset that can be further processed. The general idea of this process has already been presented in more detail, although focusing mainly on the modeling aspect and not describing the integration into the platform as presented here (Laukotka and Krause, 2023).

5.2.2. Storing the data

Data storage generally follows the multimodel data handling approach described above. If available and feasible, some information is stored in a PLM/PDM system, while other information may come from and be stored in system models, e.g., using *SysML*. Additional relationships between entities are stored in the graph to manage the information. Also, during data processing, objects and relations may be stored in the backbone object repository or graph database.

5.2.3. Linking

As described in the previous paragraphs, not all information can and will be made available directly, but some will reside in specific documents, and linking to those source documents can make it available indirectly. In this way, it is possible to use existing relationships and generic knowledge and also to link to specific information from dedicated documents. These links or references also allow different types of data to be combined, as well as different sources of information, to be combined without unnecessarily creating new copies of the same information, just stored in a different way. One reference might point to a detailed 3D scan stored on a file server, while another might use a *uuid* to point to a specific record within a PLM system, and yet another might simply contain a specific identification of a standard document that can provide additional information if needed. Since the proposed platform is intended to act as an additional layer of information management that combines these different sources and types of information, this linking is an essential part of the platform and the process.

5.2.4. Analyzing

With data collected in a centralized and managed way, different approaches to analyzing the information are appropriate. Information provided by system models or already stored in the graph database can be analyzed using graph algorithms. Schummer and Hyba (2022) have already presented several use cases of graph data analysis, albeit with a different focus on testing and anomaly resolution and using the example of the MOVE-II space mission. They also define a modeling schema and strategy that facilitates the transfer of system models to labeled property graphs, starting from the perspective and requirements of these graphs and the targeted analyses.

In the approach presented here, the analysis is not the focus of the work but an additional aspect that arises when using graph databases. For example, it is possible to identify crucial elements or clusters within the semantic dataset. Also, comparing different instances of a product, existing "as-designed" CAD or geometric "as-built" 3D scans could be done using change detection methods, as already demonstrated in (Moenck and Schüppstuhl, 2024). These changes can highlight areas of interest so that engineers are aware of which components are different or where the change needs to be modified from previous ones.

5.2.5. Accessing

With the help of the previous steps, the information is now stored in a structured, digitized form and, thus, can be made digitally accessible. Using the application or service interface (see Figure 4), e.g., a custom user interface (UI) can filter and access relevant information in a convenient and application-specific way. An example of such a UI and the principle of accessing information and documents through the platform is visualized in Figure 5; the user can define an area of interest regarding a specific asset, and based on the dependencies, relationships, directly stored information, and contained references to external documents, a corresponding dataset can be aggregated and presented to the user in the interface. This integration into modern web-based user interfaces is greatly enhanced and facilitated by the graph-based information management described above. Likewise, the platform's interface also allows other external tools or applications to control access to the dataset.

With this approach, the digitized general relations and semantics assist in the identification of the relevant systems or components of the asset, depending on the provided user input. The linking of abstract descriptions to specific documents and information allows previously fragmented information to be more easily aggregated and provided to users, thereby reducing the manual effort previously required to identify relevant entities and their respective documentation.

6. Case study

The presented approach is applied to the task of retrofitting an aircraft cabin using a mock-up of an Airbus A320 as a real-world asset. In this section, the case study and the mock-up itself are introduced before the
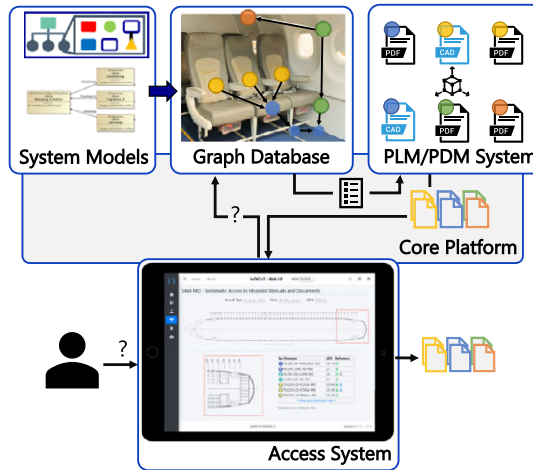
Figure 5. *The principle of access to information and documents through the platform, based on (Laukotka and Krause, 2024).*

relevant information is presented. Relevant links are identified, and then the access and use of the presented approach are described.

6.1. Introduction to the case study

As described above, aircraft are among the most complex and long-living assets. During their regular retrofits, cabin modifications are introduced, usually by a specialized third party, which faces challenges in acquiring and handling the necessary information to plan and design the new cabin with sufficient accuracy. The methodology presented to address these challenges is applied to a scenario that includes a mock-up of portions of the rear section of an Airbus A320 aircraft, which contains many of the major components of the cabin, thus providing a clear and representative case study to prove and demonstrate the approach. There are two rows of *Economy Seats*, on each side of the aisle and each with an *OHSC*. A *Partition* separates this part of the cabin from the rear, where a so-called *Skypax Gallatory*, a combination of a *Galley* and a *Lavatory*, is mounted next to the implied rear *Exit Doors*. Figure 6 shows photographs of a part of the mock-up (top) as well as the schematic of the corresponding rear section of the A320, including the actual cabin components as well as the aircraft *Frames FR59* to *FR70*, with the rear *Exit Doors* placed between the *Frames FR66* to *FR68*.

Two main examples will be used throughout this case study: the assisted acquisition of the required information and documentation, and the assessment of the mountability of specific components. In the following sections, these examples will be briefly introduced and then explored from different perspectives, starting with the acquisition of available information through modeling, transfer, and finally providing engineers with access to documentation and digital support.

6.1.1. Identifying affected components and relevant documents

The planning and execution of the cabin retrofit requires thorough documentation of the current state of the aircraft (Laukotka and Krause, 2023). Identifying the components and systems that will be affected by the retrofit, and then filtering, selecting, and reviewing the related documents is a complex task that is usually performed manually today because there is usually no easy-to-use and accessible virtual representation of the required information (Moenck et al., 2022a).
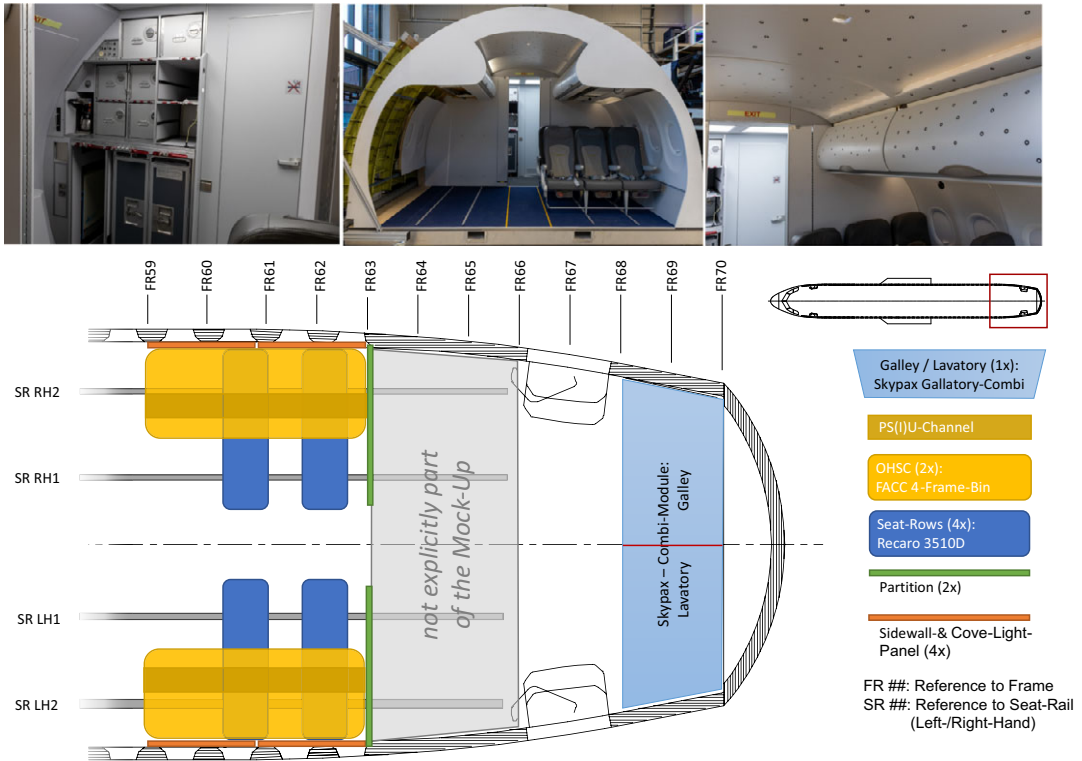
Figure 6. *Photos and schematic of the mock-up, based on (Laukotka and Krause, 2024).*

6.1.2. Assessing mountability

A typical engineering task in cabin retrofitting is, e.g., the assessment of the mountability of a component into the existing system. As an example, we outline a use case in which a *Power Delivery Unit (PDU)* must be fitted for each *Passenger Service Unit (PSU)* into the *Passenger Service Unit Channel (PSUC)* to allow *USB-Power Delivery (USB-PD)* connectors in the PSUs (cf. Figure 7). Therefore, besides checking for necessary interfaces, the engineers must estimate if the new component fits the installation space and if the mountability is given by means that a workman can mount or retrofit the specific component. Although regulatory-induced, the exact planning later must be based on documents from the original manufacturer, in this case, the aircraft manufacturer, Airbus, and the manufacturer of the *OHSC*, Diehl, the mountability assessment given the, e.g., 3D scans is a first step towards elaborating a suitable position.

6.2. Occurring semantic information

As described in the fundamentals, long-living assets, especially aircraft, rarely have a comprehensive and holistic virtual representation, but they are described by numerous distinct documents and different points of view. In addition to this information directly describing aspects of the aircraft, there usually is metainformation and comprehensive common knowledge that is not digitally accessible today. This includes relationships and dependencies between systems and components, but also documents or processes. The occurring information can be classified by their specificity: Some identified and defined information is applicable to all assets of the same type, while other only applies to a single individual asset —identifiable by its Manufacturer Serial Number (MSN). This allows to reuse of information that was already digitally defined without the need to create each virtual representation of a similar asset from scratch. To enable an enhanced virtual representation of the aircraft, including common knowledge as well as generic and specific information, a digital foundation is required. This can be achieved by using
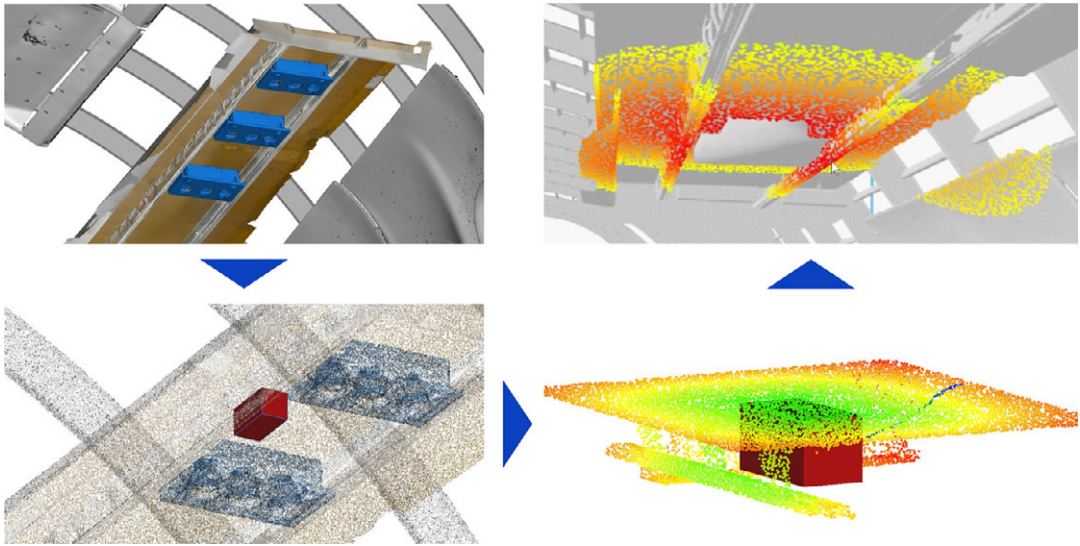
Figure 7. *Geometric mountability assessment of a power delivery unit in the passenger service unit channel.*

system modeling techniques (Laukotka et al., 2024), as was described in Section 2.3 and Section 5.1.7. In the mock-up situation, the electrical system (*ATA 24*) plays an important role, as the *PDU* and an accompanying *Power Conversion Unit (PCU)* must be connected to the electrical system of the aircraft, or in this case, the mock-up. The appropriate information is gathered, modeled, and transferred to the database.

The following sections describe this process, both on a generic level and using the example of this scenario. It should be noted that the overall process is meant to be iterative, with increasingly more information already available on the platform and less information needed to be acquired and implemented into the platform. Especially generic knowledge and information regarding a general type of aircraft rather than a specific distinct aircraft will accumulate faster and can be reused. For the scope of the mock-up scenario, the generic structure of the aircraft's airframe and the generic relations of the base components are already available on the platform. Yet, details regarding the *PDU*, *PCU*, and the general electric system of an Airbus A320 need to be handled by the process in this iteration.

6.2.1. An ontology for semantic information of the cabin retrofit

As described during the presentation of the concept, especially the digital definition of the semantic information requires a solid foundation, which is realized by creating a specifically tailored ontology. While the overall processing and digitization of semantic information is designed to be iterative and done for each retrofit, this creation of the ontology is done just once for this specific scenario of the cabin retrofit in aviation. Afterward, it will mostly stay the same for all upcoming iterations. The resulting ontology is highly specific to the application of the scenario presented here. It might even change when just shifting the scope from performing the cabin retrofit from the perspective of a third-party organization to the perspective of the original manufacturer. However, it should be seen as one tool as part of the overall concept and not the core of this contribution.

As per the concept, the standard practices and structures found in the context of the real world's assets are the base for the ontology. For this use case, there predominantly is the general taxonomy as described by the ATA Chapters and Aircraft System Architecture (ASA) (see Section 3.1) as well as the overall split of the aircraft into the existing airframe and the cabin. This ontology describes the occurring essential aspects of the system of interest as well as their relations on an abstract level. It clearly defines how an aircraft consists of different elements, with individual properties, systematic relations to each other, and

references to specific files. Because of this abstraction from the results of the previous step, the description is less detailed, and no distinct element of an aircraft is explicitly named. The definition simply states that there are elements within an aircraft and that there is a distinction between the airframe and the cabin. Similarly, no specific ATA Chapter is named, but the general possibility to reference from elements to ATA Chapter is defined. This abstract and general decomposition of the aircraft into its components is analogous to the decomposition of aircraft in related works. (see Section 5.1.4). However, the strict differentiation into airframe (as already produced) and cabin (as being newly designed and retrofitted), as well as the explicit integration of the ATA Chapters, constitute a core approach of this concept and are unique to this perspective. Within the aircraft cabin, the ontology defines a partition between cabin components as found in the cabin configuration guide, such as Galleys, Lavatories, etc., and infrastructure systems that provide these components with the required power or plumbing. The abstract description likewise considers the identified specificity of information. Since the semantic information that will be defined in the system models based on this ontology is intended to provide an abstract representation rather than a detailed definition, this abstract differentiation allows the main cabin elements to be defined, as well as their relevant dependencies on other components and systems. For this purpose, possible relationships between elements are defined in the ontology, e.g. the relationship "connected to," which describes the relationship to an infrastructure system, or "mounted on," which defines a structural connection between elements. In addition to the described references to the respective ATA Chapters, links to documents and files containing the relevant aspects in more detail are explicitly integrated.

In conclusion, the ontology specifically tailored to this application…

– targets the creation of abstract representations of aircraft for the cabin retrofit.
– defines the split of aircraft into airframe and cabin elements.
– defines the differentiation between cabin components and infrastructure systems.
– defines possible relations between elements.
– includes references from elements to their respective ATA Chapters.
– considers the specificity of elements and information.
– considers references from elements to external documents and files, which include more detailed information.

Thus, it offers a clear definition of the general structure and available elements at a high level. Additionally, it outlines the terminology in a precise manner, allowing the independent development and maintenance of the different elements of the concept that handle this semantic information.

6.2.2. From data sources to stored information

An approach to systematically model this information in system models has already been presented (Laukotka and Krause, 2023). In addition, ways to make it digitally available and to provide a way to create the information as well as to transfer it to graph databases have recently been described (Laukotka and Krause, 2024). Essentially, the process starts with retrofit planning, where the exact parameters and boundary conditions of the retrofit to be performed are defined. Based on this, the data collection phase begins, during which the resulting required documents are identified. From these documents, information is collected, and, if necessary, new information is recorded at the specific asset, e.g., by taking measurements and photographs or by determining part or serial numbers of installed components. Currently, this process is done manually and involves the manual collection of the many fragmented documents and information described in earlier sections. The collected documents and information are analyzed for relationships, which are also documented for further use in later phases. Finally, the fragments are combined into a dataset that contains all relevant information for the planned retrofit of the specific aircraft. In the next phase, this meta information is made digitally available using system modeling techniques. Relationships to existing documents and the respective ATA Chapters (see previous

sections) are included as parameters in the system models (Laukotka and Krause, 2023). As described above, the process is iterative, and the benefit of the presented approach increases with each retrofit and passed iteration. In the last phase of this process, the modeled information is transferred to a graph database where it can be accessed and processed more easily (Laukotka and Krause, 2024). Figure 8 visualizes these described phases from acquisition to stored semantic information.

In the case of the mock-up, the electrical system of an aircraft was first modeled on an abstract level by defining the available main systems, namely the *28 V DC* and *115 V AC* systems and the respective buses that supply power to other systems. The information has been extracted from the standard documentation of the electrical system architecture of an Airbus A320 and represents the basis for further detailed models, as well as possibly other modeled documentation referencing the electrical system that will be implemented later. Since the platform already provides the fundamental definition of the airframe of the aircraft or mock-up, including the existing *Frames* and their positions, this iteration focuses on the generic electrical system and mock-up specific information. This includes the installed *Seats*, *OHSCs*, and of course, more specific electrical system components. These elements are related to the connected systems and other components or structures to which they are attached. They also reference the existing documents that contain more details, such as the datasheet of the *PCU* to the more detailed documentation of the electrical system. Finally, by transferring them to the schematic database, the information becomes accessible to other systems. Figure 9 shows an example of the original schematics for the electrical system (left), an excerpt of the resulting system model (center), and an excerpt of the respective part of the graph database (right). The structure of the generic base model visibly consists of an abstract description of the entities the system consists of, as well as their essential relations.

This approach is uniquely aligned with the already established processes and structures that occur during today's retrofit process. It is integrated into the current steps of retrofit planning and execution and is based on a common system, including ATA Chapter relationships and the inherent differentiation between airframe and cabin, as found in cabin retrofits. This tailoring to the specific circumstances of aircraft cabin retrofits takes the concept beyond the usual standard approaches.
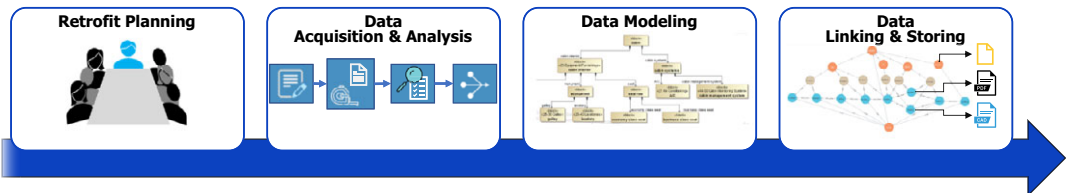

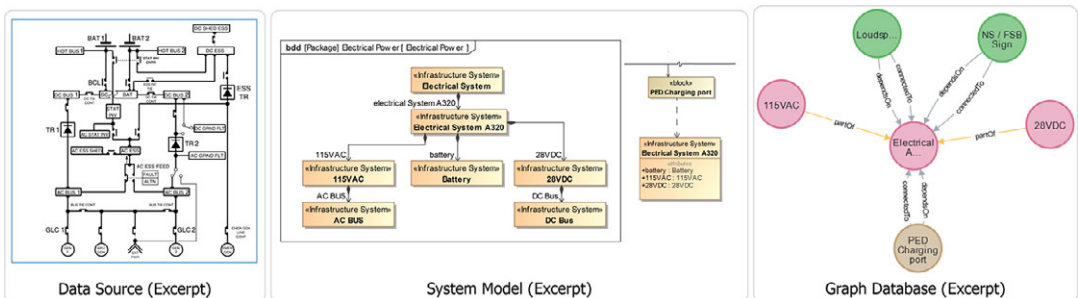**Figure 8.** *From acquisition to stored semantic information, based on (Laukotka and Krause, 2023).*

Figure 9. *Acquiring, modeling, and storing information specific to the scenario of the mock-up.*

6.2.3. *Utilizing references and providing access*

With the semantic information available in the graph database, and relations to specific documents or ATA Chapters added where appropriate, the foundation is laid for use within the design of the new cabin. The platform only becomes a useful tool if the metainformation layer is complementary and provides links to the specific actual information or documents in which the information is documented. This general idea of combining information and metainformation to create a better virtual representation of the aircraft was presented in (Laukotka et al., 2021). Since then, the concept has been further developed and implemented using the case study described. Within this implementation, not only can references be made to ATA Chapters or document IDs, but there is also a direct connection to the existing PDM system. This allows the access system to provide relevant documents, such as PDF files, directly from its own interface without having to redirect the user to another interface. Therefore, filtering and identification of relevant documents and their direct access greatly facilitates access to information and documents required for the retrofit, especially for the planning and design of the new cabin.

In the case of the presented scenario and the mock-up, the access system can help to identify the affected systems during the installation of the new components, namely the electrical system, and provide access to the available information and documents. The necessary relationships are available in the graphical database, as well as references to existing documentation that provides more details about the affected systems. The complementary geometric mountability assessment can also help with the structural side of the installation, as described in a subsequent section. The described approach of complementary information and metainformation handling using common access and management of information to the virtual representation is visualized in Figure 10.

Currently, different variations of the access management UI are being developed to optimize the selection and concomitant filtering of relevant information for different scenarios and tasks that occur regularly during the planning and design of the new cabin but also maintenance measures. With this approach, data handling goes beyond what typical PLM/PDM systems alone usually provide. Information and documents are related across products and independent of individual BOM structures, allowing the data set to be queried at different levels of specificity, going beyond current solutions that are mostly product structure oriented.

6.2.4. *Further analyses of stored semantics*

Since the data is already provided in the form of a graph database, albeit primarily for ease of access and combination with the geometric information, there is potential for further analysis of the dataset. Using the underlying graph theory and its possibilities, different analyses could be performed to get further insights into the dataset, as was already shown for other datasets by (Schummer and Hyba, 2022) and described in Section 5.2.4. For this application, such analyses could assist in filtering and selecting relevant elements and relations. While the groundwork for this is already done in the form of the used graph database and the custom UIs, this functionality is not currently the focus of this article but could complement the implementation in future work.

6.3. *Occurring geometric information*

In the presented use case, geometric information is used to describe the "as-is" and "as-designed" states. The semantic layer of the product data is extracted from STEP files, CAD data, and parsed using *GraphSTEP*, while the surface structure of the CAD models is fed through the 3D data handler into the graph database. As already described, 3D scans are also fed into the graph database as 3D points in a downsampled version (s. Figure 11). All initial input data are additionally stored in the object storage.

Individual 3D objects will be represented by an *Object* property, while all subsampled points get a *Point* property. Directed edges in the graph then establish connections between the root level *Object* element and the individual points. The *Object* itself links to a *Scene* element, which represents the root level of the specific aircraft.
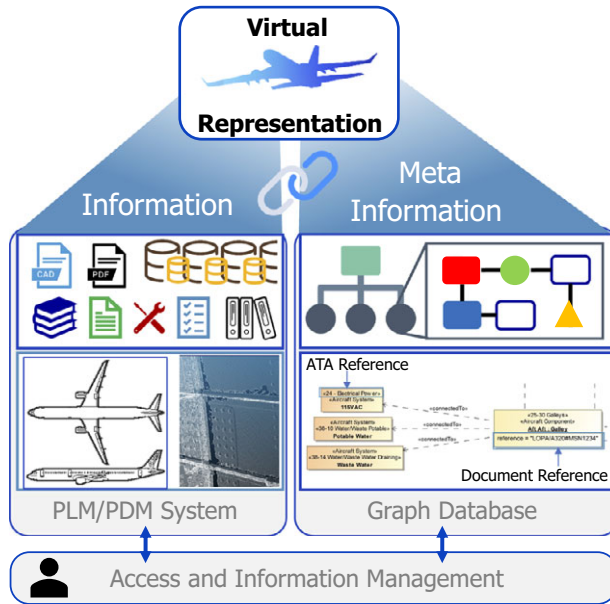
Figure 10. *Referencing between semantic metainformation and information from specific datasets, adapted and updated from (Laukotka et al., 2021).*
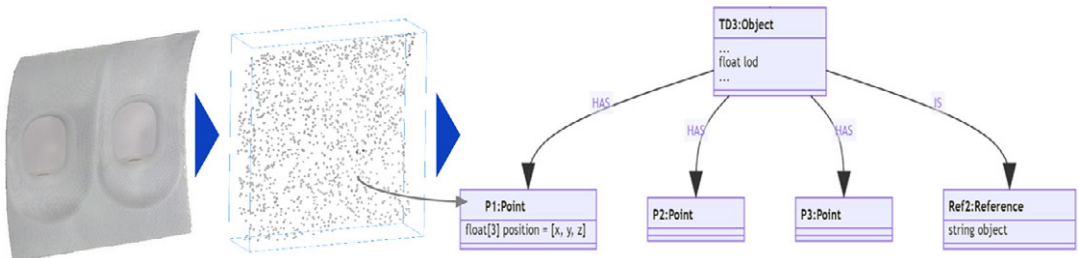

Figure 11. *Subsampling of a 3D object (left) to reduce the number of points, which form nodes in the graph (middle); the underlying data structure in the graph (right).*

6.4. Mountability assessment

Given the geometric information, a service in the application layer of the platform determines the distances to all neighboring points based primarily on *Neo4j*'s spatial querying capabilities. Based on this, the results can be graphically not only visualized in the 3D data (cf. Figure 7) but also summarized for each object on a point-wise level. The engineer can move the object to be mounted through the geometric digital twin and test the various positions to assess the mountability.

7. Discussion

The complexity of information management for aircraft and other long-living assets is the motivation for the presented approach, and it is a challenge due to the amount of information involved. The concept faces this challenge by not assuming to have a complete dataset of all available information accumulating from the beginning but instead let this dataset grow and extend over time. This is done by designing the overall approach as an iterative process that can be performed with each cabin retrofit. The compatibility of information and models across these iterations is greatly assisted by employing a systematic modeling approach and utilizing a

custom-created ontology. That way, especially the semantic information, that are more relevant across iterations because they are more generic than specific CAD geometries, can be extended over time. This clear distinction between generic and specific information ensures and supports the reusability of the information. While this aspect is not described in detail in this work, it is described in more detail in previous publications that focus on the modeling aspects like (Laukotka and Krause, 2023) or (Laukotka et al., 2024).

The presented platform for information management considers a range of aspects, from the identification and acquisition of relevant information to its digital storage and, in particular, the innovative possibilities of accessing it using a custom-tailored web interface. The injection of data into the platform is accomplished in consideration of the standards established by the engineering community, such as the STEP or system models, wherever feasible. The web interface enables users to access and interact with the information without the necessity for highly specific knowledge, for example, regarding the Cypher queries that are used internally for interacting with the graph database. While system models created using the commonly used modeling language *SysML* necessitate the respective knowledge, they are currently relatively common within the aerospace industry. The creation of such models is made easier by the application of the respective graphical authoring systems. It is therefore concluded that the overall concept, as presented in Section 5, can be utilized by engineers without the necessity for extensive training. Rather, it can be employed on the basis of the common knowledge held within the industry, with the provision of a brief introductory session.

While aircraft are the main target of the approach presented, and thus the details and background information have been described in more detail, similar challenges can be found in other domains. When retrofitting ships with newer cabins or newer engines, there is often uncertainty about the current state and the need to perform extensive manual work to extract the necessary details from existing documents and maintenance records. Borczyk and Singh (2019) describe digital ship design tools that help capture and manage ship-specific information for retrofit and conversion projects. Another area with similar challenges is the building sector, where many buildings currently need to be upgraded with more energy-efficient heating or insulation, for example. Wang and Cho (2015) describe how, for existing buildings, it is challenging to obtain the current, or sometimes even "as-built," information needed for energy modeling. They go on to describe how the process of collecting this information is time-consuming and prone to errors. Since these other domains face similar challenges, albeit under quite different circumstances, evaluating whether the presented approach is directly applicable to these scenarios is not easy and requires appropriate research. However, the general approach is based on universal tools and concepts that can theoretically be adapted and tailored to different applications. Therefore, some of the key aspects have been described in more detail to give the community access to a wider range of applications.

8. Summary and outlook

The management of information about assets throughout their lifecycle is a challenge. This is especially true for more complex and long-living assets that accumulate a variety of information and documents that need to be managed and accessed for tasks such as retrofits. However, this information is often fragmented across multiple documents or must be captured from the asset. This work presents an information management approach based on graph databases and partial system modeling techniques to facilitate the retrieval of critical information. Generic knowledge and asset-specific information are defined in system models, transferred to graph databases, and combined with additional acquired geometric information. References to existing documents are integrated, allowing links from the defined and directly accessible information to additional documents that can provide more details. Because of the different elements of the platform and the utilization of references, these documents can also be stored at different locations, like PDM systems or other data sinks. This also allows all data to be based on the same generic data model and data management system, hence allowing a single query language to access the multiple semantic layers and allow queries to work on a single data sink. A customized user interface allows different interactions with the information, such as specific selection and filtering of relevant information, as well as geometric analysis, such as mountability assessments. After describing

the background and various aspects in detail, the management was implemented with a focus on the application for retrofitting in aviation. A mock-up of the rear section of an Airbus A320 was used as a case study to illustrate and further describe the various aspects of information management.

Looking forward, this information management can form the basis for various enhancements and extensions that then further improve how information gets handled for retrofitting in aviation. Currently, only essential information is directly available, and then a reference is made to the specific documentation for further details; with the rise and improvement of AI and information extraction from documents, more information could be made directly available by including a parser that automatically makes the information available from within those referenced documents. In terms of current work, as part of the ongoing research project, more details of the mock-up and the A320 in general are currently being integrated into the platform. In addition, the information access system is being enhanced. This will allow for additional and more refined queries to the database, as well as improved visualization of the results.

References

AIRBUS S.A.S (2002a) A319/a320/a321 Aircraft Schematic Manual (ASM).

AIRBUS S.A.S (2002b) A319/a320/a321 Illustrated Parts Catalog (IPC).

AIRBUS S.A.S (2003) A320/a321 Ground Operations Manual (GOM).

AIRBUS S.A.S (2016) A320 Family Aircraft Maintenance Manual (AMM): 06 - Dimensions and Areas.

Altfeld H-H (2010) *Commercial Aircraft Projects: Managing the Development of Highly Complex Products*. Farnham, Surrey: Ashgate ISBN 9780754677536.

Arista R, **Zheng X**, **Lu J and Mas F** (2023) An ontology-based engineering system to support aircraft manufacturing system design. *Journal of Manufacturing Systems 68*, 270–288. ISSN 02786125. doi: 10.1016/j.jmsy.2023.02.012.

Arp R, **Smith B and Spear AD** (2015) *Building Ontologies with Basic Formal Ontology*. The MIT Press. ISBN 9780262329583. doi: 10.7551/mitpress/9780262527811.001.0001.

Ast M, **Glas M and Roehm T** (2013) Creating an Ontology for Aircraft Design an Experience Report about Development Process and the Resulting Ontology. https://www.dglr.de/publikationen/2014/301356.pdf.

Bajaj M, **Friedenthal S and Seidewitz E** (2022) Systems modeling language (sysml v2) support for digital engineering. *INSIGHT 25*, 19–24. ISSN 2156-485X. doi: 10.1002/inst.12367.

Barbau, Krima S, **Rachuri S**, **Narayanan A**, **Fiorentini X**, **Foufou S and Sriram RD** (2012) Ontostep: Enriching product model data using ontologies. *Computer-Aided Design 44*(6), 575–590. ISSN 00104485. doi: 10.1016/j.cad.2012.01.008.

Bechhofer S (2018) Owl: Web ontology language. In Liu L and Özsu MT (eds.), *Encyclopedia of Database Systems*. New York, NY: Springer, pp. 2640–2641. ISBN 978–1–4614-8266-6. doi: 10.1007/978-1-4614-8265-9_1073.

Berschik MC, **Schumacher T**, **Laukotka FN**, **Inkermann D and Krause D** (2023) Mbse within the engineering design community - an exploratory study. In *Proceedings of the 24th International Conference on Engineering Design (ICED 2023)*, Vol. *3*, pp. 2595–2604. doi: 10.1017/pds.2023.260.

Berschik MC, **Zuefle M**, **Laukotka FN and Krause D** (2024) Product Family Engineering along the Life Cycle – Research Aspects to cope with Variability in Advanced Systems, *Design Science*, Volume *10* , e27. doi: 10.1017/dsj.2024.21.

Berschik MC, **Blecken M**, **Kumawat H**, **Rath J-E**, **Krause D**, **God R and Schüppstuhl T** (2022) (eds.) *A holistic aircraft cabin metamodel as an approach towards an interconnected digitised cabin lifecycle.* In *The International Council of the Aeronautical Sciences*. doi: 10.15480/882.4757.

Borczyk A and Singh MP (2019) Applications & advantages of employing digital ship design tools for newbuilding, retrofit and conversion projects In *International Conference on Computer Applications in Shipbuilding (ICCAS 2019)*, Rotterdam, The Netherlands.

de Florio F (2011) *Airworthiness: An Introduction to Aircraft Certification and Operations*, 2nd Edn. United Kingdom: Elsevier/BH, Oxford. ISBN 9780080968025.

Delligatti L (2014) *SysML Distilled: A Brief Guide to the Systems Modeling Language*. Upper Saddle River, NJ: Addison-Wesley. ISBN 978-0321927866.

Deneke C, **Oltmann J**, **Krause D and Schüppstuhl T** (2019) Technology innovations for faster aircraft cabin conversion. In *7th international Workshop on Aircraft System Technology (AST 19)*. Hamburg, Germany: Hamburg University of Technology.

DIN - Deutsches Institut für Normung (1995) *Informationsverarbeitung - Begriffe - Teil 1: Allgemeine Begriffe*. DIN

Edlich S, **Friedland A**, **Hampe J and Brauer B** (2011) *NoSQL: Einstieg in die Welt nichtrelationaler Web 2.0 Datenbanken*, 2nd Edn. Germany: Hanser Verlag, München. 2. aktualisierte und erw. auflage edition. ISBN 978-3-446-42753-2.

El-Haji M (2014) *Ontologie-basierte Definition von Anforderungen an Validierungswerkzeuge in der Fahrzeugtechnik: Dissertation*, volume Band 48 of *Karlsruher Schriftenreihe Fahrzeugsystemtechnik*. Karlsruhe, Germany: KIT Scientific Publishing. ISBN 978-3-7315-0496-2. doi: 10.5445/KSP/1000052725.

Esteves LA, **Benavides AM**, **Barbieri G**, **Olaya C and Mantilla CA** (2023) Value-based smart retrofitting of maintenance in the hydropower plants of celsia. *IFAC-PapersOnLine 56*(2), 6555–6560. ISSN 24058963. doi: 10.1016/j.ifacol.2023.10.306.

FAA - Federal Aviation Administration (2008) Joint Aircraft System/Component Code: Table and Definitions: JASC. https://avinfo.faa.gov/sdrx/documents/JASC_Code.pdf.

Fasel D and Meier A (2016) *Big Data*. Wiesbaden, Germany: Springer Fachmedien. ISBN 978-3-658-11588-3. doi: 10.1007/978-3-658-11589-0.

Franzén LK, **Munjulury RC and Krus P** (2022) Ontology-assisted aircraft concept generation. In *33rd Congress of the International Council of the Aeronautical Sciences* (ICAS 2022).

Fuchs M, **Ghanjaoui Y**, **Abulawi Jutta**, **Biedermann J and Nagel B** (2022) Enhancement of the virtual design platform for modeling a functional system architecture of complex cabin systems. *CEAS Aeronautical Journal 13*(4), 1101–1117. ISSN 1869–5582. doi: 10.1007/s13272-022-00608-z.

Geisler F (2014) *Datenbanken: Grundlagen und Design*. Heidelberg, Germany: MITP 5. überarbeitete und erweiterte auflage edition. ISBN 9783826697074. https://learning.oreilly.com/library/view/-/9783826697074/?ar.

Giesecke R (2005) *Airbus Competence Training (ACT): Develop New Cabin & Cargo*. Airbus Hambur.

Grieves MW (2019) Virtually intelligent product systems: Digital and physical twins. In Flumerfelt S, Schwartz KG, Mavris D and Briceno S (eds.), *Complex Systems Engineering: Theory and Practice*, Reston, VA: American Institute of Aeronautics and Astronautics, Inc, Vol. *411*, pp. 175–200. ISBN 978-1-62410-564-7. doi: 10.2514/5.9781624105654.0175.0200.

Gruber TR (1995) Toward principles for the design of ontologies used for knowledge sharing? *International Journal of Human-Computer Studies43*(5–6), 907–928. ISSN 10715819. doi: 10.1006/ijhc.1995.1081.

Hall A, **Mayer T**, **Wuggetzer I and Childs PRN** (2013) Future aircraft cabins and design thinking: optimisation vs. win-win scenarios. *Propulsion and Power Research 2*(2), 85–95. ISSN 2212540X. doi: 10.1016/j.jppr.2013.04.001.

Hanna M, **Schwenke J**, **Schwede L-N**, **Laukotka F and Krause D** (2021) Model-based application of the methodical process for modular lightweight design of aircraft cabins. *Procedia CIRP100* 637–642. ISSN 22128271. doi: 10.1016/j.procir.2021.05.136.

Huber S, **Wiemer H**, **Schneider D and Ihlenfeldt S** (2019) Dmme: Data mining methodology for engineering applications – a holistic extension to the crisp-dm model. *Procedia CIRP 79*, 403–408. ISSN 22128271. doi: 10.1016/j.procir.2019.02.106.

INCOSE (2007) Incose Systems Engineering Vision 2020. https://sdincose.org/wp-content/uploads/2011/12/SEVision2020_20071003_v2_03.pdf.

International Organization for Standardisation (2022) Industrial Automation Systems and INTEGRATION - Product Data Representation and Exchange - Part 242: Application Protocol: Managed Model-Based 3d Engineering. https://www.iso.org/standard/84667.html.

Irlinger R (1999) Methoden und Werkzeuge zur nachvollziehbaren Dokumentation in der Produktentwicklung. In *Konstruktionstechnik München Band* 31. Aachen: Shaker. Zugl.: München, Techn. Univ., Diss., 1998, ISBN 978-3-8265-4714-0.

Jackson S (1997) Systems engineering for commercial aircraft. *INCOSE International Symposium 7*(1), 36–43. ISSN 23345837. doi: 10.1002/j.2334-5837.1997.tb02151.x.

Jiang F, **Ma L**, **Broyd T and Chen K** (2021) Digital twin and its implementations in the civil engineering sector. *Automation in Construction 130*, 103838. ISSN 09265805. doi: 10.1016/j.autcon.2021.103838.

Kastens U (2008) *Modellierung: Grundlagen und formale Methoden*. München, Germany: Hanser Verlag. 2. überarbeitete und erweiterte auflage edition. ISBN 3446415378. doi: 10.3139/9783446417212.

Kemmerer SJ (1999) *STEP: The Grand Experience, Special Publication (NIST SP)*, National Institute of Standards and Technology, Gaithersburg, MD, [online], https://doi.org/10.6028/NIST.SP.939. https://www.nist.gov/publications/step-grand-experience.

Koch J, **Lotzing G**, **Gomse M and Schüppstuhl T** (2022) Application of multi-model databases in digital twins using the example of a quality assurance process. In Andersen A-L, Andersen R, Brunoe TD, Larsen MSS, Nielsen K, Napoleone A and Kjeldgaard

S (eds.), *Towards Sustainable Customization: Bridging Smart Products and Manufacturing Systems*, Lecture Notes in Mechanical Engineering. Cham: Springer International Publishing, pp. 364–371. ISBN 978-3-030-90699-3. doi: 10.1007/978-3-030-90700-6_41.

Kopisch M and Günter A (1992) Configuration of a passenger aircraft cabin based on conceptual hierarchy, constraints and flexible control. In Belli F and Radermacher FJ (eds.), *Industrial and Engineering Applications of Artificial Intelligence and Expert Systems. Lecture Notes in Computer Science*, vol *604* Berlin, Germany: Springer Verlag pp 421–430. ISBN 3–540-55601-X. doi: 10.1007/BFb0024994.

Kwon S, **Monnier LV**, **Barbau R and Bernstein WZ** (2020) Enriching standards-based digital thread by fusing as-designed and as-inspected data using knowledge graphs. *Advanced Engineering Informatics 46*, 101102. ISSN 14740346. doi: 10.1016/j.aei.2020.101102.

Kwon S, **Monnier LV**, **Barbau R and Bernstein WZ** (2022) A new implementation of ontostep: Flexible generation of ontology and knowledge graphs of express -driven data. *Journal of Computing and Information Science in Engineering 22*(2). ISSN 1530–9827. doi: 10.1115/1.4053079.

Laukotka FN and Krause D (2023) Supporting digital twins for the retrofit in aviation by a model-driven data handling. *Systems 11* (3), 142, doi: 10.3390/systems11030142.

Laukotka FN and Krause D (2024) Combining system modelling and graph databases to improve access and analysis in data-intensive engineering tasks. In *2024 IEEE International Systems Conference (SysCon)*. doi: 10.1109/SysCon61195.2024.10553605.

Laukotka FN, **Hanna M and Krause D**. (2021) Digital twins of product families in aviation based on an mbse-assisted approach. *Procedia CIRP 100*, 684–689. ISSN 22128271. doi: 10.1016/j.procir.2021.05.144.

Laukotka FN, **Berschik MC and Krause D** (2024) Enhancing design representations of information and knowledge of complex and long-living assets by applying system modelling techniques, international design conference 2024. In *Proceedings of the Design Society: DESIGN Conference, ISSN 2732-527X*. doi: 10.1017/pds.2024.43.

Lehner F (2021) Wissensmanagement: Grundlagen, Methoden und Technische Unterstützung. Hanser München, vol. *7*. überarbeitete und erweiterte auflage edition. ISBN 978-3-446-46780-4.

Meier A and Kaufmann M (2019) *SQL & NoSQL Databases*, 1st Edn. Wiesbaden, Germany: Springer Fachmedien. ISBN 978-3-658-24548-1. doi: 10.1007/978-3-658-24549-8.

Mensen H (2013) *Handbuch der Luftfahrt*, 2nd Edn. Berlin, Germany: Springer Verlag. ISBN 978-3-642-34401-5. doi: 10.1007/978-3-642-34402-2.

Meyer J-A (1996) *Visualisierung im Management: Zugl.: Berlin, Techn. Univ., Habil.-Schr., 1996.* DUV. Dt. Univ.-Verl., Wiesbaden. ISBN 3824403234.

Meyer J-A (1999) *Visualisierung von Informationen: Verhaltenswissenschaftliche Grundregeln für das Management*. Wiesbaden: Gabler-Lehrbuch. Gabler. ISBN 978-3-409-11413-4.

Moenck K and Schüppstuhl T (2024) Geometric digital twins of long-living assets: uncertainty-aware 3d images from measurement and cad data. *Procedia CIRP 126*, 975–980. ISSN 2212–8271. doi: 10.1016/j.procir.2024.08.369.

Moenck K, **Laukotka FN**, **Deneke C**, **Schüppstuhl T**, **Krause D and Nagel TJ** (2022a) Towards an intelligent digital cabin twin to support an aircraft's retrofit and base maintenance (sae paper 2022-01-0046). In *SAE Technical Paper Series MAR. 15, 2022* Warrendale, PA. doi: 10.4271/2022-01-0046.

Moenck K, **Laukotka FN**, **Krause D and Schüppstuhl T** (2022b) Digital twins of existing long-living assets: Reverse instantiation of the mid-life twin. In *DS 119: Proceedings of the 33rd Symposium Design for X (DFX2022)*. The Design Society, pp 10. doi: 10.35199/dfx2022.20.

Moenck K, **Pustelnik A**, **Koch J and Schüppstuhl T** (2023) Graphstep: Concurrent, cloud-based, lifecycle-oriented editing and updating of product data. *Procedia CIRP 120*, 678–683. ISSN 22128271. doi: 10.1016/j.procir.2023.09.058.

Moenck K, **Rath J-E**, **Koch J**, **Wendt A**, **Kalscheuer F**, **Schüppstuhl T and Schoepflin D** (2024) Digital twins in aircraft production and mro: Challenges and opportunities. *CEAS Aeronautical Journal*. ISSN 1869–5582. doi: 10.1007/s13272-024-00740-y.

Muggeo C and Pfenning M (2015) Die rolle von mbse und plm im industrial internet. In Schulze S-O and Tschirmer C (eds.), *Tag Des Systems Engineering*, München, Germany: Hanser Verlag, pp. 279–287. ISBN 978-3-446-44729-5. doi: 10.3139/9783446447288.028.

Niță MF and Scholz D (2009) The process chain to a certified cabin design and conversion. *Deutscher Luft- und Raumfahrtkongress 2009.* Aachen, Germany: DLRK.

Niță MF and Scholz D. Business opportunities in aircraft cabin conversion and refurbishing. *Journal of Aerospace Operations 1*(1–2), 129–153, 2011. ISSN 2211002X. doi: 10.3233/AOP-2011-0008.

Niță MF (2012) *Contributions to Aircraft Preliminary Design and Optimization: Dissertation*. Germany: Verlag Dr. Hut, München. ISBN 978-3-8439-1163-4.

Njaastad EB, **Steen S and Egeland O** (2022) Identification of the geometric design parameters of propeller blades from 3d scanning. *Journal of Marine Science and Technology 27*(2), 887–906. ISSN 0948–4280. doi: 10.1007/s00773-022-00878-6.

NoMagic Inc (2011) *Magic draw - Model-Based Systems Engineering*. Online-Brochure,

North K (2021) *Wissensorientierte Unternehmensführung*. Wiesbaden: Springer Fachmedien Wiesbaden. ISBN 978-3-658-32770-5. doi: 10.1007/978-3-658-32771-2.

Nzetchou S, **Durupt A**, **Remy S and Eynard B** (2019) Review of cad visualization standards in PLM. In Fortin C, Rivest L, Bernard A and Bouras A (eds.), *Product Lifecycle Management in the Digital Twin Era*, vol. *565*, Cham: Springer International Publishing and Imprint Springer, pp. 34–43. ISBN 978-3-030-42249-3. doi: 10.1007/978-3-030-42250-9_4.

PACE Aerospace Engineering and Information Technology GmbH (2023) Pacelab ace Mycabin: The Industry Standard in Aircraft Cabin Interior Configuration. https://pace.txtgroup.com/products/product-configuration/mycabin/.

Probst GJB (ed.) (1993) *Vernetztes Denken: Ganzheitliches Führen in der Praxis*, 2nd Edn. Wiesbaden: Gabler. ISBN 9783409233576.

Rata V and Secobeanu S. Aspects of using 3-d laser scanning technology in ship retrofit projects. *Annals of "Dunarea de Jos" University of Galati 42*, 157–162, 2019. doi: 10.35219/AnnUGalShipBuilding.2019.42.21.

Rausch C, **Lu R**, **Talebi S and Haas C** (2021. Deploying 3d scanning based geometric digital twins during fabrication and assembly in offsite manufacturing. *International Journal of Construction Management* 1–14. ISSN 1562–3599. doi: 10.1080/15623599.2021.1896942.

Ríos J, **Hernández JC**, **Oliva M and Mas F** (2015) Product avatar as digital counterpart of a physical individual product: Literature review and implications in an aircraft. In Curran R (ed.), *Transdisciplinary Lifecycle Analysis of Systems, Advances in Transdisciplinary Engineering Ser* Burke: IOS Press Incorporated, pp. 657–666. ISBN 978-1-61499-544-9. doi: 10.3233/978-1-61499-544-9-657.

Sanhudo L, **Nuno MMR**, **Martins JP**, **Ricardo MSFA**, **Eva Barreira MLS and Cardoso V** (2018) Building information modeling for energy retrofitting – a review. *Renewable and Sustainable Energy Reviews 89*, 249–260. ISSN 13640321. doi: 10.1016/j.rser.2018.03.064.

Schummer F and Hyba M (2022) An approach for system analysis with model-based systems engineering and graph data engineering. *Data-Centric Engineering 3*. doi: 10.1017/dce.2022.33.

Sendler U (2009) *Das PLM-Kompendium*. Berlin, Germany: Springer Verlag. ISBN 978-3-540-87897-1. doi: 10.1007/978-3-540-87898-8.

Shearer C (2000) The crisp-dm model: The new blueprint for data mining. *Journal of Data Warehousing 4*(5), 13–22.

Singh S, **Shehab E**, **Higgins N**, **Fowler K**, **Erkoyuncu JA and Gadd P** (2021) Towards information management framework for digital twin in aircraft manufacturing. *Procedia CIRP 96*, 163–168. ISSN 22128271. doi: 10.1016/j.procir.2021.01.070.

Sommer M, **Stjepandić J**, **Stobrawa S and von Soden M** (2019) Automatic generation of digital twin based on scanning and object recognition. In Hiekata K, Moser B, Inoue M, Stjepandić J and Wognum N (eds.), *Transdisciplinary Engineering for Complex Socio-Technical Systems*, *Advances in Transdisciplinary Engineering.* Amsterdam: IOS Press. ISBN 9781643680200. doi: 10.3233/ATDE190174.

Sommer M, **Stjepandić J and Stobrawa S** (2021) Incremental update of a digital twin of a production system by using scan and object recognition. In Newnes L, Lattanzio S, Moser BR, Stjepandić J and Wognum N, (eds.), *Transdisciplinary Engineering for Resilience: Responding to System Disruptions*, Advances in Transdisciplinary Engineering. IOS Press. ISBN 9781643682082. doi: 10.3233/ATDE210086.

Son H, **Kim C and Turkan Y** (2015) Scan-to-bim - an overview of the current state of the art and a look ahead. In Malaska M and Heikkilä R (eds.), *Proceedings of the 32nd International Symposium on Automation and Robotics in Construction and Mining (ISARC 2015)*. doi: 10.22260/ISARC2015/0050.

Stark J (2016) *Product Lifecycle Management: Volume 2 The Devil is in the Details*. Cham, Switzerland: Springer International Publishing, ISBN 978-3-319-24434-1. doi: 10.1007/978-3-319-24436-5.

Stark J (2020) *Product Lifecycle Management (Volume 1): 21st Century Paradigm for Product Realisation.* Decision Engineering Ser, 4th Edn. Cham: Springer International Publishing AG. ISBN 978-3-030-28863-1. doi: 10.1007/978-3-030-28864-8.

Stark J (2022) *Product Lifecycle Management Volume 1*. Cham: Springer International Publishing. ISBN 978-3-030-98577-6. doi: 10.1007/978-3-030-98578-3.

Stjepandić J, **Wognum N and Wim JCV** (2015) *Concurrent Engineering in the 21st Century*. Cham, Switzerland: Springer International Publishing. ISBN 978-3-319-13775-9. doi: 10.1007/978-3-319-13776-6.

Stjepandić J, **Sommer M and Denkena B** (2022) *DigiTwin: An Approach for Production Process Optimization in a Built Environment*. Cham: Springer International Publishing, ISBN 978-3-030-77538-4. doi: 10.1007/978-3-030-77539-1.

Stojanovic V (2021) *Digital Twins for Indoor Built Environments*. PhD thesis, Universität Potsdam

Walden DD, **Roedler GJ**, **Forsberg K**, **Hamelin RD and Shortell TM** (2015) *Systems engineering handbook: A Guide for System Life Cycle Processes and Activities: INCOSE-TP-2003-002-04, 2015*. 4th Edn. New York, NY: Wiley. ISBN 978-1-118-99941-7. http://gbv.eblib.com/patron/FullRecord.aspx?p=4040424.

Wang C and Cho YK (2015) Application of as-built data in building retrofit decision making process. *Procedia Engineering 118*, 902–908. ISSN 18777058. doi: 10.1016/j.proeng.2015.08.529.

Weilkiens T and Sendler U (2013) Modellbasierte Systementwicklung. Was Sie Schon Immer über mbse, PLM und Industrie 4.0 Wissen Sollten.

Williams M (2021) A Roadmap for MBSE data Standards: Annual Incose International Workshop 2021, Boing.