DLR-IB-DW-JE-2021-55

The Current State of Research and Technology of Digitalization in the Space Industry

Sarah Böning

Deutsches Zentrum
für Luft- und Raumfahrt

The Current State of Research and Technology of Digitalization in the Space Industry

Sarah Böning

March 24, 2021

Contents

Common Acronyms

AIT assembly, integration, and testing

AM additive manufacturing

AP application protocol

API application programming interface

CAD computer-aided design

CD concurrent design

CDM conceptual data model

CE concurrent engineering

CEF concurrent engineering facility

COTS commercially off the shelf

CPS cyber-physical system

DLR Deutsches Zentrum für Luft- und Raumfahrt

DT digital twin

EDS electronic data sheet

ESA European Space Agency

IoT Internet of Things

IT information technology

JSON JavaScript Object Notation

MBSE model-based systems engineering

NLP natural language processing

NASA National Aeronautics and Space Agency

PDF portable document format

PT physical twin

REST Representational State Transfer

SE software engineering

UML unified modeling language

1 Introduction & Space 4.0

The space industry is currently undergoing a structural change. One factor is the rising number of new actors in space. For example, private space flight companies, such as SpaceX and Blue Origin, are now driving innovation. It is shifting the industry from very few, expensive mission launches by a small number of space-faring nations to many companies and universities around the world frequently launching their own satellites and satellite constellations. For instance, SpaceX is currently in the process of launching 7,500 satellites that will form a mega-constellation to offer high-speed internet around the world [4]. In order to launch so many satellites, the manufacturing process and its related factors have to adapt. Therefore, many rely on small, inexpensive satellites with commercial-off-the-shelf (COTS) components, such as CubeSats [85].
Other driving factors include digitalization and corresponding digital and technical innovations, including advancements in artificial intelligence, big data, additive manufacturing (AM) (for example, 3D-printing), and intelligent factories [43, 109].
This new paradigm is often referred to as New Space and Space 4.0, following the term Industry 4.0 that describes the current fourth industrial revolution of manufacturing started by digitalization [26].

The advantages of implementing Industry 4.0 principles into the manufacturing of the space industry to fit the New Space concept have already been investigated in literature. Nardon [85] analyzed the impact of digital technologies on the European space industry in their policy paper. They found that the space industry in the United States is ahead in adapting to Space 4.0, as there reside the most space start-ups and big digital companies (Amazon, Apple, Facebook, Google, Microsoft). These do not only provide the technology that is used in space-related processes, for example, operating systems, but also started funding their own space projects. The founder of Amazon, Jeff Bezos, established the aerospace company Blue Origin. Furthermore, the author noted that software is becoming more important to the space industry. A new business model is to offer a customized software solution with cheaper, mass-produced hardware: launchers and satellites. The focus is shifting towards provision of complete services to clients rather than customized hardware to system designers of space vehicles.

Gaudenzi et al. [43] investigated the potential of Industry 4.0 in the context of manufacturing. They identified application scenarios of Industry 4.0 and Internet of Things (IoT) to the manufacturing, assembly, integration, and testing cycle of satellite production. The focus was on cyber-physical systems (CPS) in intelligent factories that get information from a sensor network for distinct processes. The provided data can be analyzed with the help of big data analytics and machine learning to enable automatic task scheduling, performance monitoring, and better production. The authors concluded that factories that include CPS can be beneficial for the satellite manufacturing.

In their master's thesis, Klemme [70] identified important technologies from Industry

4.0 applicable to the space industry using the example of the German satellite manufacturer OHB. They found that each aspect of a manufacturing process has different needs that can be addressed by different Industry 4.0 technologies. For example, endoscopes and augmented reality can be helpful for satellite integration. Satellite integration is the process of assembling and connecting the sub-systems together into the satellite unit. An obstacle for further digitalization is the lack of paperless processes. The author found that there is a need for company-wide, coherent IT-systems that digitize multiple processes related to satellite manufacturing.

Braun and Braun [14] also concluded that new technology such as appliances from the Internet of Things can be advantageous for satellite manufacturing and operation. However, they pointed out several security issues that arise from implementing IoT technology. IP-connected devices could potentially be vulnerable to malicious attacks. Security flaws in IoT devices on earth can get patched easily with updates. This is not the case for satellites in operation, as their debugging ports can require physical access. The authors concluded that these issues need to be addressed before IoT technology should be implemented in satellites.

As mentioned above, the most important drive of Space 4.0 are small, inexpensive satellites. The whole life-cycle of a spacecraft, including all satellite types, from design to disposal can be divided into several phases. The European Space Agency (ESA) defines seven phases in the standard ECSS-M-ST-10C [27]: 0, A, B, C, D, E, F. In the first phase, 0, the feasibility of the mission is determined. Phase A is the initial design phase, in which the feasibility of the planned spacecraft needs to be assessed. Phases B and C aim to refine this first design by defining the spacecraft with all its components in detail. The fifth phase, D, contains the assembly, integration, and testing (AIT) of the spacecraft and the corresponding ground segment. Launch and operation are phase E. Finally, the spacecraft is disposed in phase F. Each of these phases has their unique set of challenges and opportunities regarding digitalization, which will be discussed in the following.

In this report, we highlight the current state of research and technology of the digitalization in the space industry and its related benefits to the whole life-cycle of space vehicles. In the following, we focus on ten areas we find have the biggest potential for further digitalization: concurrent engineering, model-based systems engineering, digital twins, ontologies, artificial intelligence, big data analytics, electronic data sheets, augmented and virtual reality, robotics and manufacturing, and standardization.

2 Concurrent Engineering

The first steps in the life-cycle of a future spacecraft are planning and design. For the initial phases 0 and A (following the ESA standard), it has become common practice to utilize concurrent engineering (CE). Concurrent engineering is the "simultaneous design

of a product and all its related processes in a manufacturing system" [66, p. 4] by (physically or virtually) co-located experts for each design discipline to be considered. The goal is to identify problems that would otherwise emerge in later engineering phases at intersections of the different disciplines as early as possible. Therefore, the number of design and conception iterations for the product in later engineering phases is decreased, which reduces costs over the course of the project [36]. It is also associated with decreased design and development time and higher customer satisfaction [80].
Concurrent engineering is not a new concept, but the advancements in computer systems and software over the last decades helped improve and digitize the CE procedure. Most improvements result from the experts being able to simultaneously use the same software tools [63]. In this section, we outline the current state of research for concurrent engineering in the space sector; a non-exhaustive list of concurrent engineering facilities can be found in Appendix A.1, and a list of CE software in Appendix A.2.

In recent years, the focus of research regarding concurrent engineering for spacecrafts has been on improving the design of existing CE spaces and software but also adapting them to more phases of the spacecraft life-cycle. The Concurrent Design Laboratory was developed as a part of the ESA_Lab@TU Darmstadt [53] with focus on design studies for ground segments and operations of space missions. Until now, CE studies focused mainly on spacecrafts and often excluded more complex systems, such as ground stations and operations. In the past, smaller components of a ground station were designed; for example, a mobile Telemetry, Tracking, and Command ground segment has been planned with concurrent engineering [78]. At the moment, there are no standards defined regarding the collaboration of different design experts and CE procedures for ground stations and operations to the best of our knowledge. Ground-station design is more interdisciplinary, as non-space related aspects, such as IT infrastructure, have to be considered. Thus, Hoffmann et al. [53] adapted the CE procedure including models and software tools for the concurrent design of ground stations to be used in the Concurrent Design Laboratory. It was opened in late 2019.

In a similar approach, Tomasicchio et al. [106] described their architecture of a Concurrent & Collaborative Design Facility (C2DF) as a new form of concurrent design facilities (CDF) covering complete space missions, including modeling single spacecrafts and constellations as well as ground segments and operations. The described CDF has been created for the Italian spaceflight service company Telespazio. The common CE approach was supplemented with collaborative methods, such as augmented reality to visualize system models. Together with virtual machines providing the same software for all participants, this allowed for (remote) collaboration. According to the authors, it is planned to cover studies for phases 0 - B in the facility, with a possible later extension to phases C–D, when suitable methods have been defined for complete missions.

CE facilities are successful at improving the designs of spacecrafts. However, little research has been done yet on how to improve the design process with new technology.

Therefore, Chavy-Macdonald and Kneib [20] proposed the concept of CD^2F, a concurrent design and data facility. This CE lab includes the idea of a design observatory in which different simple designs can be tested by experts of the CE process. This allows for ethnographic studies on design processes and, together with sensors such as microphones, enables the collection of empirical data about collaborative design activities. The data was analyzed after a completed study to gain more insight about the process. However, the authors noted that the goal is to process the data in real-time to offer support to the designers by provision of an indicator of the quality of different design aspects. The proposed CD^2F concept has not been realized yet.

Other facilities, for example the Concurrent Engineering Facility (CEF) in Bremen, Germany by the German Aerospace Center (DLR), have been used to conduct studies for several years already. Here, the focus is more on gathering insight and lessons learned from finished studies to further improve the procedure at the facility and study design [80]. From 2007 to 2017, over 60 CE studies were conducted at the CEF at DLR Bremen. In [80], Martelo et al. presented the statistics of the finished studies. The biggest benefits included decreased development time and engineering changes as well as fewer defects and an increase in overall quality of the designed product. In most cases, studies had new participants that never took part in a CE study before. Thus, emphasis should be on a guided procedure and easy-to-understand software tools. The authors found that it is also important to have experienced team leader that help prevent potential miscommunication between experts of different fields.

Similarly, Knoll et al. [71] reviewed the common practices in different concurrent engineering centers for space products. Two of the biggest ongoing challenges were identified as team formation and special training of team leaders. Further improvements in all five cornerstones of CE could be expected: conceptual design, team psychology, conceptual models, software tools, and procedure and facility [71]. Many studies had at least one person working remote, but current solutions, such as conference calls, had their drawbacks. Often mentioned problems were disconnections and unstable internet connections of the people working remotely. Complete in-person meetings are still perceived as more productive. One possible future solution was identified as meetings set in augmented or virtual reality environments (see section 9). However, the issue of unstable internet connections persists. Another point for improvement was new software with real-time synchronization between several users instead of the common version-controlled approach. Furthermore, the current modeling techniques with formal languages were perceived as not adequate by the engineers in the early stages of development by participants in a survey. The authors concluded that more research is needed to find more appropriate modeling for the conceptual design phase of CE studies.

The growing number of CubeSats missions being designed in CE studies revealed room for improvements. Jahnke and Martelo [64] proposed possible adaptations to the CE process for CubeSats. Since there are many COTS parts available for CubeSats, the role of some experts shifts from sub-system design to searching the most suitable component. Here, one experienced designer could take the task of searching for several sub-systems. This reduced the number of participants. In addition, using COTS parts reduced the number of design iterations and the product design underwent fewer changes. In addition, for many CubeSat parts, data sheets were available but needed to be searched for. Often no 3D model were available. These are needed for visualizing the designed product. This amplifies the need for an extensive database of COTS components and subsystems for later reuse. For the CE study itself, the authors concluded that a model-based approach could be more suited.

CE studies are not exclusive to spacecrafts and their mission. They can also be applied to design single parts by manufacturers. To generate parts with multiple functionalities and modern manufacturing techniques, the common CE process needs to be adapted. Samoil et al. [98] proposed a CE methodology for spacecraft parts. They found that in this particular case, more preliminary studies at the beginning followed by more detailed design sessions until the review of the design product are needed. It is planned to test the proposed procedure at OHB System.

As mentioned above, concurrent engineering is mostly used in the initial phases of the spacecraft life-cycle. However, work in the later phases could benefit from a similar collaborative work, as concluded by Jahnke et al. [65]. According to the, the biggest starting point is the utilization of model-based systems engineering (see section 3). Here, the data model describing the system and its state is designated to be used during the whole life-cycle. The authors state that concurrent work with a shared data model are not used outside CE studies at the moment. However, especially in later stages of phase B design work, a central model could be helpful in a concurrent setting, according to the authors. Therefore, a software tool that is also suitable for activities of other phases and supports the sharing of such a model should be considered to extend CE to phases B-D.

In a similar approach to make the results of CE studies more accessible in later development phases, Fortin et al. [38] defined a data model based on the SAPPhiRE (State-Action-Parts-physical Phenomenon-Inputs-oRgan-physical Effect) model of causality [19]. It is used to help product designers across all disciplines explore the design space and find new concepts to apply to their problem [19]. The extended model by Fortin et al. for space application is supposed to improve the understanding of a spacecraft's structure and behavior between different experts and stakeholders. In order to apply this model for spacecrafts, its concepts were mapped to the data model of the CE software CEDESK [72]. This allowed for a potential reuse of models and data in future studies.

Another aspect to improve the CE studies is to tailor software tools to the facilities. ESA [35] has developed their own software tools to be used in their concurrent design facilities: OCDT and ConCORDE as server-client software. The server side is realized as the Open Concurrent Design Tool (OCDT), which utilizes the data model defined in ECSS-E-TM-10-25A [28]. It offers a REST-API to synchronize and exchange data model files during a CE study via a database. This has to be implemented in a client. ESA's client is realized as ConCORDE, a Microsoft Excel plugin, which can used by all experts during the study.

The CE software Concurrent Engineering Data Exchange Skoltech (CEDESK) was developed by Knoll and Golkar [72] for the CE facility of the Skolkovo Institute of Science and Technology. It is used for conceptual design studies of complete space exploration missions and satellite constellations. The focus is on a collaborative approach of several experts to consider the structure of a system and the interaction of individual components. The tool is primarily used in the conceptual design phase and focuses on synchronization of a system model between workstations similar to OCDT. In addition, role assignments were implemented to allow for collaboration [72].

Ehresmann et al. [24] proposed the use of evolutionary algorithms for the design of affordable satellites for constellations. The Evolutionary System Design Converger (ESDC) automatically optimized the design of a spacecraft with real-world performance data from the individual parts within a given set of constraints. With the growing number of COTS components, it can happen that a systems engineer misses an optimal configuration. The evolutionary algorithm starts with a random configuration, which it mutates until an optimal solution is found that satisfies all constraints. The authors note that ESDC is planned to be integrated into a satellite constellation design tool.

One aspect that is often not considered during CE studies is the environmental impact of a spacecraft. Since this is becoming a more pressuring issue, a life-cycle-assessment (LCA) tool that can be integrated into existing CE software was proposed by Wilson and Vasile [111]. Its backbone was the Strathclyde Space Systems Database (SSSD), which can be used to calculate the environmental impact of a space system and its related processes [110], such as materials and production. The analysis is based on the ISO standards 14040 [59] and 14044 [60] for life cycle assessments. According to the authors, this can help alter the optimal design criteria towards more environmentally friendly processes when already employed in the early design phases.

In summary it can be said that concurrent engineering is a tried and tested method for the design of satellites for particular space missions. Its benefits have been reported in literature [80]. However, it is rarely used for other systems of missions, such as ground stations, or different types of space vehicles. Additionally, the CE methods had to be adapted for CubeSat-based missions [64]. The studies are highly individual and operators of facilities adapted to their own tailored set-ups. To this end, many individual CE software tools have been developed, often to suit the needs of one facility [36, 72]. These customized procedures require special training of staff and participants and cooperation between procedures and software is not given. Moreover, the CE procedure in itself is rather closed. Efforts have been made to extend CE procedures to later life-cycle stages and to make the generated data easily accessible outside the CE study [38, 65]. An aspect that is often omitted in CE set-ups but is becoming more important of late is the environmental impact of the designed system. Here, CE studies should also include evaluation methods for sustainability.

3 Model-Based Systems Engineering

Another force that actuates further improvements in the CE process is model-based systems engineering (MBSE). For this report, we follow the definitions of [40] and [94] that MBSE "can be described as the formalized application of modeling principles, methods, languages, and tools" [94, p. 105] "to support system requirements [engineering], design, analysis, verification and validation" [40, p. 5] over the whole life-cycle of a system from design to disposal. MBSE can be understood as the digital advancement of traditional, document-based systems engineering.
While concurrent engineering is often applied only in phases 0 and A, MBSE covers the whole life-cycle. However, both techniques share common principles, and more and more advancements of MBSE have been incorporated into CE studies [63]. In this section, we describe current trends in MBSE for the digitalization in the space sector. A list of MBSE tools and software can be found in Appendix B.1.

In current research, the focus is on model representation, suitable software tools, and improvements to the MBSE process. An ongoing issue is to improve the digitalization of documents. While MBSE is characterized by digital representations and processes, communication between involved parties is mainly still document-based. For example, reports, data sheets, and contracts are still exchanged as PDF documents making the integration of data and information into system models cumbersome [34]. Here, standardized communication methods are needed across the whole industry, which is a challenging and ongoing task [34]. Thus, Ferguson et al. [34] adapted the MBSE process at BAE Systems to implement document representation in their MBSE software as a transitional solution. The system model was adapted to additionally contain a representation for documents and needed attributes to later generate documents again. These documents are based on the state of the model and can be exchanged in response to other stakeholders. First, documents, such as requirements, got imported into a database. Their information was extracted and imported into the model. Information from the documents could then verified with the model in different life-cycle phases. Together with other data from the model, documents such as reports could be then generated by the MBSE software based on a predefined, reusable template [34].

In a similar approach to reduce the number of documents exchanged between actors, the Common Information Platform (CIP) was developed by RHEA Group [11]. It enabled the exchange of models rather than documents between different stakeholders and made them compatible to different tools via an API. This was achieved by combining different models, for example, requirements and physical architecture, into a single model. This also allowed the model to be verified and tracked through the whole life-cycle [11]. CIP is still in development.

While the engineers designing systems are familiar with modeling languages and MBSE tools, the ones responsible for the requirements verification, inspection, and testing are not [77]. Thus, Latserus et al. [77] proposed a methodology to link modeled requirements with their respective verification procedure supporting engineers without modeling experience. This allowed for the requirements to be traced through the life-cycle as well as testing and verification documents to be generated automatically [77].

Models of spacecrafts are subjected to many changes during a design phase. However, since there is no interconnection between models, changes from one model cannot automatically be applied to the others if necessary [102]. Thus, Stevens [102] proposed to implement the concept of a Digital Thread into MBSE for space. A digital thread is a framework that helps to intertwine separate features of a system by exchanging information. For MBSE, these separate parts are often individual models from different life-cycle stages. The proposed framework focused on the conceptual design phase of a spacecraft. Here, changes in the mission requirements have implications on the design of sub-systems. To implement the digital thread, Stevens used a shared modeling language. This resulted in a model of the complete system with requirements and constraints that could be verified automatically. This model relied on the models of the sub-systems. The author concluded that it would be possible to extend the digital thread framework to later life-cycle phases connecting the models with data from a real system or digital twins (see Section 4).

DLR implemented MBSE in their CE procedures with Virtual Satellite, a modeling software tool with a built-in conceptual data model (CDM) [36]. A CDM is used in the conceptual design phase and is a language to describe a system and the relations of its components. The CDM for Virtual Satellite by Fischer et al. [36] was designed to support engineers during phase A spacecraft design studies at DLR's CEF. It is not based on an ECSS data model standard but is built to fit the CEF procedure at DLR and to be extensible and customizable to other systems or phases [36].

The extension of Virtual Satellite to more spacecraft life-cycle phases is outlined in [37] by Fischer et al. The system models were stored in a central database to allow reuse in either a later phase or a new study. The architecture of the underlying CDM was based on a generic systems engineering language, so it could be adapted and extended for each phase and specifications of different systems. Thus, there were suitable models for each phase which could exchange data by being based on the same database and modeling language. In addition, the authors note that information specific to each phase also needs to be stored in the database. Engineers should then be able to access the parts of the models relevant to their field.

Conceptual data models can be more meaningful if they also include semantics [54]. Hoppe et al. [54] introduced the use of ontologies (see Section 5) to add semantics to CDMs. Ontologies facilitate automatic reasoning for inference of knowledge. The authors presented the Digital Space Systems Engineering approach to add knowledge to space CDMs. The basis was a transformation of the ontology model to Ecore, the Eclipse Modeling Framework (EMF) data model, and back. This allowed the incorporation and representation of knowledge in MBSE spacecraft models. The mechanism could be used to further improve and add functionality to the MBSE procedures according to the authors.

In a further development of semantics for data models, Hoppe et al. [56] described the Semantic Engineering Modeling Framework (SEMF). It aimed at improving MBSE by the introduction of reasoning to evaluate the validity of models and discovery of new information by inferring knowledge from the model. The authors concluded that this reduced possible inconsistencies. When stored in a database together with inference logic and access for engineers, ontologies also enable the later reuse of the models and an easier adaptation to different systems than other modeling languages (typical MBSE modeling languages are UML and SysML) according to Hoppe et al.

With the increased popularity of small satellites, more so-called mega-constellations are designed. However, since this is rather a new field, most modeling software tools are not suited well enough for constellations [69]. Thus, Kharlan [69] proposed a tool for the early mission analysis that was customized to model and analyze constellations of small satellites for telecommunication purposes. The tool features one main, logical model of the overall system accessible by all engineers and a method to store information about subsystems. More information can also be imported from external sources if real data is available for parts from manufacturers or past missions. The tool is planned to be integrated into concurrent environments for constellation design studies.

Model-based systems engineering can be utilized for more aspects of a space mission than just spacecrafts. LaSorda et al. [76] introduced an MBSE approach for architecture selection of ground-based systems, such as a mission control center, for satellites. The main concern for these larger segments is uncertainty of different design options. The goal was to model these as well as possible to enable an automatic and understandable evaluation of different design aspects and their respective uncertainty. The authors could then trace the concerns of stakeholders along the process, which lead to more informed decisions on all architecture alternatives. They envision to implement this process in the early stages of system acquisition.

Uncertainty also plays a role in the design of spacecrafts and their components. Often, problems in the design of the complete system are only seen later when combining the different sub-system models. The outcomes of the decisions made during the initial design are not completely foreseeable and therefore have a high degree of uncertainty[21]. To take uncertainty into account during the design process, Chodas et al. [21] proposed the Model-based Adaptive Design under Uncertainty (MADU) framework. It adapts the complete system's model when the design is changed. If the decision has implications on other parts of the system, MADU can be used to automatically adapt the concerning parts using conflict learning. Conflict learning is used to avoid picking solutions from unsatisfiable regions of the search space [21]. This method also takes uncertainty into account by solving constraint satisfaction problems. MADU builds the constraints with information from the system model. For further improvement, the authors want to incorporate previous design data for optimization later.

As seen in the examples above, the focus of research for MBSE in the space domain often focuses on the design stage of spacecrafts, namely satellites. However, it is designed for the whole life-cycle of a product. One important area is the interconnection and linkage of the different design and system models along the production phase, so that different actors can benefit without extensive modeling knowledge [77, 102]. As with concurrent engineering, many companies and research facilities have developed their own MBSE software tools tailored to their individual needs and procedures. However, one prominent obstacle is the ongoing use of documents, such as PDFs, that are difficult to integrate automatically [11, 34]. Interoperability and standardization of electronic documents and models across different platforms could be beneficial here. Regarding the underlying CDMs of the MBSE software, the use of semantics and ontologies seem to be promising [56].

4 Digital Twins

A rather new development that is often closely related to MBSE is the Digital Twin (DT). There are various definitions of the digital twin in the literature. We use the term digital twin in accordance with the definition presented at INCOSE 2020 [12] in this report: A digital twin is the virtual representation of a system or product and utilizes digital information available to it (such as product specifications) and sensor data provided by the physical twin (PT). It is used to run simulations and make predictions for adaptations on and better the understanding of said product or system over its whole life-cycle. In this section, we illustrate the use of digital twins in the space industry.

Garnier et al. [42] identified three common patterns of digital twins and their respective use cases for space. The first pattern is called "Twins acting concurrently" [42, p. 2]. Here, the digital and physical counterparts share the input and output data and can interact with each other. This pattern has two use cases: avatars and complementary activities. In the avatar case, the DT acts as a middleman between the user and the physical counterpart. An example in the space domain is a collaborative robot. In the case of complementary activities, one twin can support or command the other one. In the space domain, this can be used to predict failure and anticipate maintenance.

The second pattern is "DT to provide feedback" [42, p. 3], where both entities collaborate in a form of feedback loop. The actions and output of the PT get analyzed by the digital counterpart, which then can correct the PT's behavior if necessary. This pattern has two variations: the closed and open feedback loop. In the former, there is no interception, and in the latter, a human or other system can be interposed. In the space domain, the open loop pattern is used to support decision-making of critical functions, and the closed loop is considered for in-operation adjustments,for example, power regulation.
The final pattern is the "DT to feedforward directives" [42, p. 4] pattern. It is similar to the second pattern. Here, the digital twin analyses the effect of its directives together with observations of the behavior from the PT. Here again, the loop can be open or closed to other entities. Space-related use cases are ground-based planning of spacecraft operations and virtual operators of unmanned space vehicles.

Fault detection in spacecrafts is often simply threshold-based [113]. The recent advancements in neural networks and artificial intelligence (AI) offer the possibility for better and more profound fault analysis. However, since the computing power of spacecrafts is limited, Yue et al. [113] proposed a framework to analyze the sensor data from rockets with a digital twin for ground station-based fault detection. The ground station received the sensor data, which was fed directly into the DT model. It was then analyzed together with other available data (including flight path and geometric data) with a combination of algorithms, including AI-powered fault prediction, in real-time. A 3D model was updated to see the rocket's current state. When a possible fault occurred, a simulation with the DT was run to verify several recovery strategies. All data was stored in a database for later reuse; for example, they were used as input to improve the DT's AI algorithms for better prediction and detection in the future. At the moment, the described fault detection is constrained to rockets, but, according to the authors, could be generalized to include more fault scenarios; for example, power and electrical of different (un-)manned space vehicles.

In a similar approach but applied on a smaller scale, Burov and Burova [15] developed a digital twin of a composite over-wrapped pressure vessel (COPV). COPVs can hold high-pressure fluids and are commonly used in propulsion engines of spacecrafts. They are lighter than metal containers but more prone to material failures [81]. To prevent these faults and predict when they occur, the DT is used for stress state and failure analysis via simulations. The goal is to run simulations with the DT for prototyping, testing, and inspection rather than performing the actual procedures on the physical counterpart, as the vessel is often difficult to access [15].

The manufacturing of spacecraft parts can also benefit from digital twin technology. Meng et al. [82] used digital twins to simulate and plan the behavior of intelligent assembly robots for large spacecraft components. The simulation received real data from the robots, so the DT could then planned the assembly process avoiding collisions. This was achieved by a probabilistic road map motion planning algorithm performed on a 3D virtual model. The robots and the assembly platform had sensors installed to gather data. This were then sent to the DT which updated the 3D model. The digital twin simulated possible movements with the model and sent the best configuration to its respective PT [82].

Weber Martins and Anderl [108] also proposed the use of digital twins for robotic assembly. However, they envisioned the assembly of satellites for mega-constellation in space with the so-called Space Factory 4.0. They applied concepts of Industry 4.0 to an in-orbit assembly procedure. Digital twins are supposed to represent the current state of the satellites at all times during assembly, integration, and testing. They assist their respective PTs by analyzing their telemetry data as well as reacting to disruptions. The DTs should be able to send commands to their physical counterparts [108].

In summary, digital twin is a new but promising method for the space industry. At the moment, their use is often limited to one certain problem, such as fault detection [113]. However, digital twins can be utilized during the whole life-cycle, so there is room for more complex use cases. Future applications include the use as an interface between human operator and a partly autonomous robotic assembly platform in orbit.

5 Ontologies

As mentioned in a previous chapter, ontologies can improve the MBSE procedure for spacecrafts. In philosophy, ontology is the the study of being. In computer and information science, an ontology is the formal, explicit representation of concepts of one or multiple domains. For the common understanding and sharing of knowledge, an ontology represents the concepts as relationships between objects. These are defined in a representational vocabulary. Ontologies model existing things and their relations so that new knowledge can be inferred [46, 49, 100]. In this section, we highlight the current research regarding ontologies in the space sector. A list of general and space-related ontologies can be found in Appendix C.

As mentioned in Section 3, ontologies can be used to add semantics to conceptual data models. Hoppe et al. [54] added knowledge to MBSE CDMs by transforming ontologies to the Eclipse Modelling Framework data model, Ecore. In a further development, the authors [57] evaluated the use of ontologies as the single modeling tool without the mapping to a different language. They found that ontologies are well suited to enrich MBSE as so-called knowledge-based systems engineering. Other modeling languages (UML, SysML) would not expressive enough to include semantics in contrast to ontologies defined with the Web Ontology Language (OWL). OWL also allows for adjustment and extension of the data model during run time, which is infeasible with other languages [57]. Furthermore, engineers can profit from reasoners, which infer knowledge from ontologies with rules. The rules can be formulated with the help of the engineers' past experiences and previous designs from an MBSE database. The authors found that inferred knowledge helped to easily access otherwise unknown or implicit facts and find inconsistencies in the overall design. Since the transformation can only be as expressive as the target language, the authors conclude that a transformation from ontologies to other data model languages should not be in the focus of research any more. Thus, they proposed the use of ontologies for knowledge-based systems engineering in the space domain; an approach they named semantic space systems engineering [57, p. 3].

In an effort to advance semantic space systems engineering, Hoppe et al. [55] extended their semantic CDM framework to more concepts with OWL profiles. The goal was to connect the models' components to different stages of the system life-cycle. Thus, the view on the model is changed based on the current design stage (for example, initial design, detailed design). This also implies that certain aspects can be abstracted in some stages, forcing the engineer to change the stage of the model to adjust the design of a certain part. With the help of an ontology reasoner, switching the stage leads to checking for consistency and correctness of the changes providing instant feedback. Hoppe et al. found that this can prevent over- or under-engineering at the preliminary design stages. The authors call this approach guided systems engineering as a part of knowledge-based SE. They conclude that the stage-based abstraction and automatic inference can increase the data quality of the system model and reduce the time to complete a design.

Similarly, Hennig et al. [52] developed ontologies to improve the MBSE approach for space system design. In accordance with [54], they found that ontologies are the best way to integrate semantics into conceptual data models, and that system engineers can benefit from the automatic reasoning for better design quality. They transformed a CDM to OWL and added formalized knowledge from available documents of previous missions. This was compiled into a knowledge base. A knowledge base holds structured data (knowledge) from an ontology and a corresponding inference engine. They designed two ontologies: an MBSE ontology representing a CDM, and one describing a satellite system. The MBSE ontology is based on the data model specified in ECSS-E-TM-10-23A [31]. Both proposed ontologies were published: [50] [51]. It was found that while MBSE can greatly benefit from ontologies, it cannot be completely replaced by them because a knowledge base is not capable of replacing an MBSE systems database as certain functionality. Thus, Hennig et al. see ontologies as a valuable enhancement to MBSE and its system database. This contradicts the findings from [54] in parts.

Berquand and Riccardi [10] proposed a method to automatically convert system data models to knowledge graphs. Knowledge graphs contain connected instances of an ontology. They also allow for inference of knowledge. All models compliant to ECSS-E-TM-10-25A [28] can be migrated, according to the authors. The 10-25 UML model was first transformed to fit the knowledge graph scheme. To populate the graph, the system models were exported as a JSON (JavaScript Object Notation) file, which was then mapped to the graph scheme and added. Additionally, the authors trained a document-based embedding to calculate the similarity of mission requirements. Together with a reasoner, this embedding helps to gain more knowledge of the graph's contents. This approach is aimed at assisting system designers of spacecrafts in early design stages.

As a first step to build an extensive space mission design ontology, Berquand et al. [7] proposed an approach to semi-automatically generate an ontology based on unstructured data. Their data corpus included different types of documents related to space systems engineering: feasibility reports, space mission design books, and related Wikipedia entries. From these documents, frequent entities and synonyms were extracted with the help of a natural language processing (NLP) pipeline. Similar entries were merged to identify different concepts. Before they could be added to the ontology, they needed to be reviewed by human experts. This approach is related to the ongoing development of the Digital Engineering Assistant (DEA) at the University of Strathclyde. DEA is planned to be an intelligent virtual assistant to engineers in the conceptual design phase of a concurrent engineering study. The ontology is supposed to bridge the gap between the intelligent system and the knowledge of engineers [84].

Arista et al. [1] proposed the use of ontologies for the design and planning of assembly lines in the aerospace industry. The ontologies were part of a framework to model products, processes, and resources related to the assembly. Its aim was to support the design and management of an assembly line in the conceptual phase of a spacecraft, as the assembly needs to be considered as early as possible to avoid later problems and shortcomings. In future research, the actual ontologies have to be realized.

With the help of ontologies, Zhao et al. [115] developed an automatic conceptual design tool for spacecrafts. They built a knowledge base of the design of complete spacecrafts. As a basis, an ontology describes the whole design process for a spacecraft and the relationship between single activities. The design is represented in a conceptual map. New knowledge is added if the quality is evaluated as high enough. A reasoner was used to infer production rules from domain knowledge. For each sub-system of a spacecraft, a knowledge model can be built, which is then used to automatically generate a valid concept design. This procedure was demonstrated by designing a power sub-system with an ontology. In the evaluation, the authors found that the rendered design was valid and that this automated procedure can save time and work of engineers. In future work, this can be extended to combine all designed sub-systems into one automatically designed spacecraft.

All in all, ontologies are powerful tools for modeling concepts and are often used as conceptual data models for MBSE. However, there is no clear consensus if they only enrich or replace MBSE methods [52, 54]. Other important application scenarios include the design of spacecrafts. Here, the provided inference of knowledge and rules defined for reasoners allow for an automatic design procedure. However, this has only been applied on a sub-system level [115]. An important issue is the lack of ontologies for the space domain [7]. The generation of meaningful ontologies is a tedious procedure that requires verification by experts. Efforts have been made to automatically compile an space mission design ontology, however this also needed expert reviews [7].

6 Artificial Intelligence

One of the most prominent driving factors of digitalization is artificial intelligence. The fields of artificial intelligence and machine learning use learning algorithms to automate different tasks. The more data becomes available, the better AI algorithms can train and improve. Typical AI tasks include object recognition, information extraction, classification, prediction, and optimization [44]. In this section, we highlight some of the trends regarding applications and digitalization in the space industry. In Appendix D, a list of related AI tools is given.

In current research, a variety of application areas for AI in the space domain are found. For example, optimization of spacecraft designs is an area that can benefit from advancements in machine learning. Krijen and Guo [73] evaluated the use of reinforcement learning (RL) to generate CubeSat designs out of a catalog of COTS parts automatically. Reinforcement learning is often selected for optimization tasks where the output cannot be foreseen, as the problem is rather complex [73]. This is the case for the design of a complete satellite: with the increasing amount of suitable components becoming available, the design space is also growing. Thus, it is becoming infeasible for a systems engineer to find the optimal configuration for the given requirements. The optimization is based on the formulated requirements of the satellite and its sub-systems. In a case study, it was found that RL is promising for the automated design of a CubeSat for an earth observation mission. However, the chosen approach had some limitations, as the design tool was unable to generalize over previously unseen data (requirements, components). The authors therefore recommended a different approach and to extend the database of COTS components to further improve the design tool.

In a similar approach, Norheim [86] developed a component selection algorithm for the conceptual design of satellites with mixed-integer nonlinear programming. They also applied their optimization algorithm on CubeSat designs of earth observation missions with all available COTS components. They generalized the requirements, constraints, and properties of the individual parts as a nonlinear problem. This was then given to a solver. They found that their approach can find an optimal solution within about 100 million possible configurations in a few minutes, in a worst-case scenario. This outperformed any manual component selection by far. Thus, it could be beneficial for engineers, who want to test several options in the early design stages in a short period of time [86].

Engineers in the conceptual design phase can benefit from access to knowledge from previous missions. To make this information reusable, Berquand et al. [6] automatically classified space mission requirements with topic modeling. Topic modeling is an unsupervised machine learning method to learn and extract topics from texts. They trained a Latent Dirichlet Allocation (LDA) model on Wikipedia entries related to space mission design. LDA is a probability-based topic modeling method. It is based on the assumption that similar texts have a similar distribution of topics. In the evaluation, the authors found that the performance in a classification task depended on the queried category, which indicates a bias in the training data. The lack of a curated, free data corpus on space mission design was identified as one of the main hindrances to further improve the LDA model. This tool is also part of the Digital Engineering Assistant developed at the University at Strathclyde [6].

Tipaldi et al. [105] proposed a framework to utilize AI for the analysis of spacecraft flight data. The goal was to augment current health monitoring and diagnostics tools during operation. The framework includes three main tasks: root cause analysis of anomalies, behavior prediction of the spacecraft, and simulation models. However, using domain knowledge about spacecrafts for the root cause analysis is an ongoing challenging task [105]. The authors concluded that while their framework is applicable to PROBA (Project for On-Board Autonomy) satellites in a case study, more work is needed for the root cause analysis and simulation models to achieve better results.

With new optimization in hardware and frameworks, the execution of AI algorithms could be no longer restricted to stationary, powerful computers. Manning et al. [79] proposed to run software with the machine learning framework Tensorflow Lite on hardware used for SmallSats. They queried a convolutional neural network for image classification on existing flight hardware. Their findings indicate that machine learning can be used directly on board a spacecraft in future missions. The application scenarios include processing of acquired data before its sent to ground stations, optimizing communication, and mechanisms for autonomous operation.

Guariniello et al. [47] proposed the use of AI algorithms for the automatic extraction of space mission-related data to support the decision making in early conceptual modeling of spacecrafts. The tool first mined relevant data and documents, from which it extracted information. The information was stored in a space architecture database. The algorithm was then trained to find useful input for analysis tools. In a case study, their approach was evaluated on a design of a habitat module. The tool was used to gather all relevant data for an analysis in an analytic work bench tool. To further improve the trained model and expand their features, the authors proposed to extend the architecture database with more data verified by experts.

Pate [89] envisioned a complete concurrent engineering platform that is based on AI algorithms. The goal was to support engineers in a concurrent design study of a complex system with an intelligent tool. The service should run in the cloud and implement concepts from MBSE and IoT supported digital twins. The cloud-based approach was chosen to support multiple users from different locations at once. The IoT technology is supposed to enrich a possible digital twin of a system with better visualization and understanding. For the concurrent engineering process, the platform should support the design conceptualization, feasibility studies, and generation of an optimal system design with AI-based tools. In a case study, a re-entry vehicle was designed to evaluate the workflow of the platform. The approach is not implemented yet.

The idea of an intelligent assistant for concurrent engineering studies, the Design Engineering Assistant (DEA), at the University of Strathclyde was first proposed by Murdaca et al. [84] and its architecture has been refined since: [8] [9]. DEA is based on several machine learning techniques to support engineers in the design phase of spacecrafts. It has two intended use cases: a knowledge base that can be queried by an engineer to access knowledge and an active AI-based assistant that monitors the current design process and automatically prompts engineers with discovered inconsistencies. The queries will be answered in natural language text by the system for a better understanding. Additionally, it is emphasized that AI should not be used to completely replace the engineer, but to support them in an unobtrusive manner. The DEA architecture is divided into two parts, smart-squid and smart-dog [9] [84]. Smart-dog holds the knowledge graph and its inference engine as well as related NLP functionality to add knowledge, while smart-squid includes the front-end and UI with which the engineers and experts interact. The system should also able to learn during run time. After a CE study has concluded, its final report should be added to the knowledge base to further improve its mission design knowledge. Human experts should also be allowed to provide feedback to the system in a feedback loop. As of now, the overall system of DEA is still being planned, but smaller parts have been published: [6, 7, 10].

The idea of an intelligent assistant for the design phase of spacecrafts was also adopted at Cornell University. They developed the concept of Daphne, an intelligent assistant for the design of earth observation satellites. Daphne was first introduced in [3] and later refined in [58] [107]. The main goal is to reduce the mental load of engineers exploring the design space of distributed space missions for earth observation. Here, the design of several satellites in a constellation produce a large, high-dimensional design space, and it can become overwhelming to explore all possibilities. Thus, Daphne is designed to support engineers in the conceptual design phase. The back-end is called Daphne Brain, which sends the users' requests to different roles. Roles encapsulate different functionality of Daphne. For example, the critic role analyses a given design and suggests improvements, whereas the historian rule returns information from previous missions [58]. Queries can be sent in natural language and are then classified with a convolutional neural network before they are passed to the corresponding role. In a user study, the authors found that Daphne can improve the design quality. Engineers preferred an intelligent system as a team mate that gave feedback from time to time rather than a stand-alone tool. Daphne currently has a web user interface. The source code for Daphne has been published and a demo is available online (see Appendix D.1).

In summary, artificial intelligence can be applied to a variety of procedures in the space industry. Again, major applications are the automatic component selection and optimization of satellite designs [73, 86]. Moreover, the possibility of running AI algorithms directly on board a satellite is being considered [105]. However, the most promising use case in the long run is the concept of an intelligent assistant. Two assistants are being developed at the moment: DEA and Daphne [8] [58]. However, both are restrained on helping engineers during the conceptual design phase of satellites. In the future, such intelligent agents could become more universal and be applied to more phases of missions.

7 (Big) Data Analytics

The term Big Data describes "large-volume, complex, growing data sets with multiple, autonomous sources" [112, p. 97]. These data sets can be data from a variety of IoT sensors or content published on the internet, for example. To gain knowledge from these huge amounts of data, new and effective processing methods are needed, so-called Big Data analytics. Many sectors can benefit by gaining new insights and knowledge from these huge data sets that would otherwise be hard to discover. We describe the current use of Big Data in the space sector related to digitalization in this section.

In the space industry, Big Data is often related to big data sets of satellite data, for example, operational and experimental sensor data. In their thesis, Keppel and Bretagnolle [68] reviewed the possibilities for Big Data to improve the operation management of satellites. Operation management is the "planning, controlling[,] and directing of activities within a business to satisfy customer needs"[68, p. 5]. They found that Big Data has been extensively researched for benefits on earth; for example, earth observation for defense, climate change, and agriculture. However, there is a lack of research for Big Data in space applications. Possible scenarios include health monitoring, failure prediction as well as space debris localization and collision evasion. All these applications can help to improve the longevity of satellites and increase profit of companies offering services based on satellite data.

Qiao et al. [93] used data analysis in the form of principal component analysis and clustering for the trade-space analysis of CubeSat designs. The trade-space comprises all viable combinations of design choices, that is part selections. As mentioned before, this space can become very large and difficult to find the optimal solution for an engineer. The authors chose principal component analysis (PCA) to reduce the dimensionality of the individual objects in the trade-space. Then, k-medoids clustering was applied to show similar design choices. The result were visualized in a scatter plot and the user could select their group of interest. The visualization was then reduced to only the selected points and further details were provided to support the decision between all options. In a case study, it was found that spacecraft designers found this tool helpful, as they were less overwhelmed during decision-making on components.

In order to analyze the data gathered during testing of a spacecraft, Sun et al. [103] developed a Big Data processing framework. The raw experiment data was stored in a Hadoop cluster. Hadoop is a framework for distributed data storage and processing. The data was then processed in parallel. Analysis results, statistics, and other information was is exported to a database. Users coukd then access these via a web service. For further improvement, the authors suggested that more nodes can be added to the cluster to reduce the processing time and handle bigger data sets.

A Big Data processing framework for experiment data from spacecraft testing was also proposed by Zhang et al. [114]. However, they envisioned the test data to act as a baseline for health monitoring and failure prediction. First, the data was preprocessed and cleaned before storage. During the analysis, intermediate results were saved to reduce the execution time. Afterwards, the results got processed and were visualized in real-time in a user interface. The system calculated the system health and fault prognostics. In case of a detected anomaly, a notification was sent. The authors plan to connect the framework to other analysis tools in the future.

Schwenk and Herschmann [99] developed a framework for on-board data analysis for satellites. The framework should also be able to analyze the gathered data during operation and offer a system to query the data in near real-time. At the moment, it can take days until all raw data of a satellite is available on the ground, as the data cannot be sent consistently to ground stations [99] . Thus, they proposed the on-board data analysis and real-time information system (ODARIS) framework. It should be able to analyze data on board. The results should be accessible on request within minutes, and the system should instantly send notifications in case an alarm event was detected. The ODARIS system is still in planning and has not been launched yet.

In summary it can be said that huge amounts of data accumulate over most phases of the space vehicle life-cycle. Thus, methods have been developed to efficiently analyze them [103, 114]. While Big Data is used successfully for benefits on earth in many areas, such as agriculture, climate research, and defense, less is known about its benefits in space and for mission optimization [68].

8 Machine-Readable Documents & Electronic Data Sheets

Despite the growing expansion of digitalization in the space industry, there are still several areas that are very document-based; for example, contracts, reports, and data sheets of components are mostly exchanged as documents between the involved parties [34, 90]. Unstructured documents, such as PDF files, are not machine-understandable, and the relevant data needs to be extracted and made available in the corresponding software tools. This can be done by hand, which can be very error-prone, or by automatic extraction, which requires a standardized representation of the documents and certain knowledge about the vocabulary of the extracted data. For example, parameters might be called by synonyms (mass, weight). Thus, a digital standard for data transmission and processing is needed, that is electronic data sheets (EDS). In this section, we describe the current state of research regarding the digitalization of documents and development of electronic data sheets. A list of EDS-related tools can be found in Appendix E.

To enable a digital and direct exchange of product data from manufacturer to client, Peters et al. [90] proposed a prototype to supply the MBSE tool Virtual Satellite with product information of CubeSat COTS components. This is often provided as PDF documents, or bullet points and continuous text on websites in a non-machine-readable form. In addition, there is no standard vocabulary to describe the information and often the manufacturers use different terms than the data model of MBSE tools (e.g., *mass* & *weight*), or the same term describes a different parameter. Their prototype mined data sheets from CubeSat parts manufacturers. The product information was stored in a database that was connected to Virtual Satellite via a plug-in. The entries were mapped to the corresponding entities of the Virtual Satellite data model. The authors concluded that a standardized format to exchange product data is needed and that it should also describe the semantics of the entities for easier mapping between terms.

Satsearch.co is a website that lets users search for suitable satellite components and related services. In [74], they described their efforts to digitize PDF data sheets and transform them into electronic data sheets. The information was gathered in a knowledge base and is accessible via an API. They used JSON as the format for their EDS to provide machine-readable information. The transformation has been done previously by hand but was planned to be automated with additional software. The data from satsearch's EDS is currently available as extensions for valispace and CDP4 to automatically integrate information about satellite parts into the design [74].

Murdaca et al. [83] used a knowledge-based approach to automatically extract relevant information from satellite part data sheets. Relevant information is considered to be a parameter with the value and a possible measurement unit (e.g., mass 1kg). They designed a domain ontology to capture all important concepts and knowledge about satellite parts from the data sheets. This was used for ontology population to extract the parameters. The approach also identified and extracted synonyms and patterns. In a case study, they extracted information from data sheets of reaction wheels provided by *Satsearch*. As the baseline, the corresponding EDS from the *Satsearch* API were chosen. Eight attributes were selected that should be extracted. *Satsearch* also built an ontology from their data, which was combined with additional concepts to act as a terminological ontology. This ontology was used to compare the extracted data with the data provided in the EDS. In the evaluation, it was found that a formal ontology is the most important part of the process, as the data is unstructured and not standardized regarding format and vocabulary. A more accurate extraction is only possible when the documents become more standardized and an extensive ontology is used. Other external factors that complicated the correct extraction included conversion errors and typographical mistakes in the data sheets. The knowledge-based information extraction procedure is part of the smart-dog component of DEA.

In a similar approach, Opasjumruskit et al. [87] also used ontologies to extract information from satellite part data sheets. They developed a tool called ConTrOn (continuously trained ontology) that extracts information and extends the underlying ontology from the documents. This ontology was initially developed by domain experts together with the authors. In the first step of the pipeline, all texts got extracted from the data sheets. These were then analyzed to identify topics. A synset from WordNet was chosen to represent the concept of the topic. Next, the ontology got extended with knowledge from Wikidata of the derived concepts. In the final step, the parameter-value-unit tuples were extracted with the help of the ontology by highlighting them in the PDF file.
The user interface for ConTrOn, DSAT (data sheets annotation tool), was described in [88]. It enabled engineers to review and export the information found by ConTrOn. Found key-value-unit-tuples were highlighted in the PDF. Users could then correct errors and add new data that was missed. The data was then exported for further use and was automatically added to a database.

ESA [92] is working on the standardization of electronic data sheets with Spacecraft Onboard Interface Services (SOIS) EDS and Space Avionics Open Interface Architecture (SAVOIR) EDS. SOIS EDS was only intended for data handling and spacecraft communication. SAVOIR EDS is aimed at standardizing the knowledge about multiple domains (communications, electrical, thermal, and mechanical) related to spacecrafts to be adapted across the aerospace industry. The SOIS EDS standard is defined and published in CCSDS 876.0 – "Spacecraft Onboard Interface Services – XML Specification for Electronic Data Sheets". ESA is planning to release or rework further standardization documents: CCSDS 867.1 – "Specification for Dictionary of Terms for Electronic Data Sheets for Onboard Components" and 870.1 – "Electronic Data Sheets and Common Dictionary of Terms – Overview and Rationale".

In an effort to digitize the document exchange between supplier and customer, Garcia et al. [41] proposed a tool to access digital engineering data. In a pre-study, they determined which type of documents are candidates for digitalization. They found that not every document is suited to be digitized and transformed into models in the near future, for example, design descriptions and justifications. Based on the results, they built a prototype to access digital engineering data (such as requirements, documentation, models). The data was stored in a graph database, which is based on a domain ontology. It used NLP to suggest similar, but not linked, requirements and PDF documentation augmentation to link directly to a requirement mentioned in a document. The authors plan to develop the prototype further in future projects at Thales Alenia

Space.

In summary, the persistent issue of non-machine-understandable documents is a major obstacle for the digitalization of the space industry. Here, the most prominent effort is the standardization of electronic data sheets [92]. However, the complete adaptation across the industry will be a lengthy process. Many transitional solutions have been proposed in the mean time. Within this field, one trend is the automatic extraction of technical data of satellite components to aid the design process [74, 83, 87, 90].

9 Augmented & Virtual Reality

An important driving factor from Industry 4.0 that is also adapted to the space sector in different life-cycle phases is augmented & virtual reality (AR, VR). In augmented reality, additional virtual information is laid over the environment on AR glasses (AR head-mounted displays) or the display of a mobile device. In virtual reality, an immersive 3D virtual world can be explored with special VR headsets; the real environment is not visible any more. In this section, we describe the current research regarding the application of AR and VR in the space industry.

In the space domain, AR and VR are often used for new visualization of space vehicle designs and analysis of different procedures. Cipriano et al. [22] surveyed different possible applications and advantages for AR in the early stages of CE studies. They identified four possibilities. The first concerns the orbits of the to be designed satellites. AR could be used to better visualize the planned altitude and orbit as well as related simulations and analysis results. The second application is an AR interface for software tools to show the 3D system models from the corresponding computer-aided design (CAD) software. It could also be possible to assemble the model in AR. The third application scenario extends the previous one, as the authors envision the use of AR for MBSE. AR could be used to visualize the progress of the design, link and map requirements, and edit collaboratively the models in the AR environment. The final use case is a tool to help manage CE and CD sessions. Possible features include visualization of team member domains, links between experts, and demonstration to replace classical slide-based presentations. The authors plan to test a prototype of one of the four scenarios in the near future.

In [5], Baranowski et al. proposed the use of AR head-mounted displays and a 3D user interface for spacecraft design and verification. The goal was to provide a tool with which the design can be visualized and manipulated in 3D. In the evaluation, the authors built an AR prototype integration for Virtual Satellite with Microsoft HoloLens. They compared the task performance of users with the 3D HoloLens interface against the standard UI of Virtual Satellite. Baranowski et al. found that an AR interface can be a good extension of the desktop-based visualization, but more intuitive manipulation metaphors than hand gestures are needed.

Similarly, Bahnmüller et al. [2] developed a controller-based AR tool for the early design phases of space vehicles. The controller was used to translate and rotate objects in six degrees of freedom. It has a button to signal hold or release of an object. The controller's orientation was tracked via infrared relative to a head-mounted AR display. In a user study, the authors evaluated the task performance of users with the controller-based input versus hand gestures. It was found that the controller method had more precise object placement and that it took less time to complete the design of a satellite.

A VR-based framework for aerospace design called AeroVR was proposed by Tadeja et al. [104]. It should help engineers get an overview of the system design and compare the geometry and performance parameters. To get a meaningful 3D visualization, the design space had to be reduced in dimensionality. This was achieved with subspace-based dimension reduction. The AeroVR environment could visualize the part that was currently designed within the system in 3D and a 3D scatter plot of the part's performance parameters. The authors evaluated AeroVR in a user study with the goal to design a compressor blade for engines. Participants interacted with the visualizations with an Xbox game controller. The interface was verified for usability and expressiveness. In the near future, the prototype is planned to be extended to a complete 3D turbo-machine model and include a knowledge-base of domain expert knowledge to help with the design procedure.

Zimmermann et al. [116] conceptualized the use of AR and VR for change impact analysis to improve the design review process. They used heat maps to visualize the impact of the changes during analysis. The graphics can be viewed and edited by several users simultaneously. To make the review understandable to different stakeholders, a filtering mechanism was developed to present consistent information for each review step. The authors concluded that the AR and VR could improve the fast decision making and exploration of design alternatives. Overall, less design iterations would be needed.

In [39], Freitas and Sousa described the application of AR to satellite AIT procedures for the project AR2Telecom at Lusospace. They defined workflows, to which additional information and 3D models were linked. These were than displayed on a head-mounted AR display. It was found that AIT engineers were more informed and more aware of the relevant data and actions. Overall, the AR technology increased the productivity by guiding through the procedures and providing additional information. The authors estimated that the costs for AIT could be reduced by 50%.

In conclusion, virtual and augmented reality technologies are flagship innovations of Industry 4.0 and can also be used within the Space 4.0 paradigm. Many possible applications have already been identified, predominantly for better visualization of designs and analysis results [2, 39, 116]. However, many of these ideas are still elementary and not completely implemented yet. Preliminary studies have been performed as a proof of concept [2, 116]. Both AR and VR are known to cause several discomforts, also known as cybersickness [101]. This has yet to be discussed for the applications in the space domain.

10 Robotics & Manufacturing

The driving factors of Industry 4.0 include improved manufacturing processes, such as additive manufacturing, and intelligent factories. These advancements can also be applied to the space industry for the manufacturing of space vehicles and their components [43]. In this chapter, we highlight the current trends of robotics and manufacturing related to digitalization in the space industry.

Additive manufacturing techniques promise to reduce cost and weight of a manufactured product by combining different components into one with multiple functionalities, so-called part consolidation and function coupling [13]. Spacecraft parts are often highly individual and manufactured with product modularization: one component only has one function [13]. To find a trade-off between these two opposites, Borgue et al. [13] designed a methodology to weigh between integrality by part consolidation and modularity for space vehicle parts for additive manufacturing. As input, it took information about available parts, such as data sheets and CAD models, and output a modular design architecture optimized for AM. The calculations took into account adaptability, interface costs, and weight reduction. In a case study, the authors applied their methodology to redesign a satellite antenna. They found that the new design has a higher value and that design decisions were perceived as precise and informed by domain experts.

The advancements of Space 4.0 facilitate a pivotal new sector: in-obit and in-space manufacturing and assembly. Driven by the visionary goals of newer scientific missions with larger telescopes and lunar and Martian habitats, it becomes vital to build larger constructs in space to reduce the costs and weight of launchers. Fuel is directly related to the weight of the cargo [97]. Roa et al. [96] reviewed technologies for in-orbit assembly by robots. The largest remaining challenges are the absence of gravity and its implications on the assembly procedure and platform robots are attached to. In addition, in an analysis of possible application scenarios, the authors identified needed functionalities of future assembly in space. The assembly platform should be modular to support a variety of operations on a range of different structures. The robot should be capable of executing several operations autonomously to rely less on remote human operators on earth. Furthermore, there is a need for a standard interface for docking and berthing as well as data and power connections. While additive manufacturing can be beneficial in this area, the authors emphasized that for the near future, robotic assembly of structures will still be relevant. However, there are still on-going challenges, such as the effects of zero gravity, that need to be further researched.

To launch satellites more frequently, automatic integration and assembly can also be performed in orbit. This was the goal of the project Space Factory 4.0 by Weber Martins et al. [109]. It implemented concepts from Industry 4.0 for the New Space sector. The goal was to develop procedures and technologies for an in-orbit platform with robotic assembly of satellites for more rapid production and testing. To achieve a higher production frequency, the designs of the satellites have to be highly modular and be assembled by a robot. Digital Twins should be used to facilitate testing and automatic documentation. Since reliability of the autonomous assembly platform could be an issue, a second approach was developed within the project: a bilateral controller for the assembly robot with a human-machine interface. The developed concepts could be applied to an in-orbit space factory in the future.

The exploration of exoplanets require large, space-bound telescopes. In order to assemble them in space, Roa et al. [95] proposed the use of an autonomous robotic platform, PULSAR (Prototype of an Ultra Large Structure Assembly Robot). PULSAR is related to the Horizon 2020 program by the European Commission. The goal was to provide an experimental verification of the concept, that can then be further developed into an autonomous in-space assembly platform for large structures and telescopes. PULSAR was planned to have three different demonstration prototypes. The first is a mobile robotic manipulator that should perform autonomous assembly and optical verification thereof. The second is an underwater platform to test the assembly process in low gravity. The final prototype is be a simulation that comprises a full mission. Final demonstrations have yet to be performed.

The examples described above show that new trends of Industry 4.0 are also implemented in the space industry for manufacturing. Additive manufacturing and intelligent, autonomous assembly robots and complete factories are in the forefront [109]. One emerging trend is the concept of autonomous in-orbit assembly platforms. It is envisioned to build complete satellites, telescopes, and later even larger structures in space. First concepts have been demonstrated, but the launch and operation of such platforms is rather in the distant future [109].

11 Standardization

Together with new technology and methods, the need for standardization emerges. In this chapter, we focus on standardization efforts in the space domain outside of industry committees. A non-exhaustive list of space industry-related standards and industry consortia can be found in Appendix F.

The rising complexity of CubeSat systems increases the design cycle time. Expertise about the interconnection of different components is needed, shifting the focus from the payload design. This can diminish the experience of students in educational projects [45]. Thus, Gregory et al. [45] defined a standardized MBSE template for PSAT1U CubeSats. The template is based on the CubeSat System Reference Model by INCOSE

[67] (see Appendix F.2). A baseline model describes the CubeSat completely except the payload and mission information. In this way, mission-specific models can be easily designed without any work on subsystems. To further support student engineers, changes to the system are propagated automatically through the model. A library of system components is available to promote the understanding of the design space. While this template was designed for educational purposes, the authors concluded that it could be beneficial for experts in different engineering disciplines without knowledge in systems engineering to design CubeSats more easily.

Lanza et al. [75] proposed extensions to the ISO 10303 [61] AP209 standard in an approach to unify several data types that accumulate during AIT procedures. ISO 10303 is also called the Standard for the Exchange of Product model data (STEP) standard and AP209 focuses on simulation and CAD data. During AIT, huge amounts of data arise from many different platforms and tools and in a variety of formats. To pool the data in one platform for more efficient AIT, the authors want to map specifications and convert all data into one standard format based on ISO 10303. Thus, they extended the norm to fit all AIT data. The converting functionality and the complete platform are planned to be realized in the near future.

Little research has been done yet on the environmental impacts of space missions. Harris and Landis [48] reviewed current life cycle assessment standards and described their application in the space industry. The use of LCA for space applications is very limited. Analysis should not only include the impacts on earth but also in space. Here, the main concern is debris. In a case study, LCA was applied to a potential space elevator design. The results indicated that it is a sustainable and economically viable possibility for orbital transport and that LCA can be successfully applied in the space domain. The authors concluded that sustainability engineering and assessment should be more prioritized in the space industry.

In summary, it can be stated that the emergence of new technologies, such as CubeSats, is always accompanied by need for industry-wide standardization. Efforts have been made to develop an MBSE template for CubeSats and to facilitate uniform data exchange [45, 75]. Additionally, standards are needed for life cycle assessment, as the environmental impact has not been considered much yet for space missions [48].

Acknowledgments

I thank my colleagues Diana Peters, Laura Thiele, Tobias Andreas Köhler, Kobkaew Opasjumruskit, and Philipp Matthias Schäfer for their input and feedback during the writing of this report.

A Concurrent Engineering

A.1 Concurrent Engineering Facilities

A non-exhaustive list of concurrent engineering facilities, shortened and reorganized as a table from [80]:

Organization	**Facility name**	**Location**
Airbus Defence & Space	Space Code Space Code	Friedrichshafen, Germany Toulouse, France
CAST Shenzhou Institute	CDF	Beijing, China
DLR	CEF	Bremen, Germany
ESA	CDF	Noordwijk, The Netherlands
JAXA	Emergence Studio	Tsukuba, Japan
Massachusetts Institute of Technology	DE-ICE	Cambridge, US
NASA, i.a.	COMPASS PDC HEDS-DIE EDS	Brook Park, US Pasadena, US Houston, US Hampton, US
Skoltech	CEDL	Moscow, Russia
Thales-Alenia Space	CDF ISDEC COSE	Cannes, France Rome, Italy Torino, Italy
University of Strathclyde	CDF	Glasgow, UK

A.2 Concurrent Engineering Tools & Software

A list of available free and commercial CE software:

	Virtual Satellite CEF	CEDESK	IRAS DCEP	OCDT	CDP4
Developer	DLR	Skoltech	University of Stuttgart	ESA	RHEA Group
Status	available, open-source	available, open-source	in development	available, ESA community	available, commercial
Link	[a]	[b]	[c]	[d]	[e]
ECSS compatible	✓	×	?	✓	✓
Role assignments	✓	✓	?	role-based permissions	✓
Life-cycle phase focus	conceptual design	conceptual design	conceptual design	conceptual design	conceptual design
Modeling focus	behavior & geometry	behavior	?	behavior	behavior
Version control	✓	✓	?	✓	✓
3rd party tools	×	limited	?	✓	✓
MBSE compatible	✓	✓	?	✓	✓

---

[a] `https://github.com/virtualsatellite/VirtualSatellite4-CEF`, accessed 17.02.2021

[b] `https://github.com/cedesk/data-exchange`, accessed 17.02.2021

[c] `https://www.dlr.de/bt/en/desktopdefault.aspx/tabid-12817/22396_read-51559/#/gallery/33638`, accessed 17.02.2021

[d] `https://ocdt.esa.int/`, accessed 17.02.2021

[e] `https://products.rheagroup.com/cdp4`, accessed 17.02.2021

B Model-Based Systems Engineering

B.1 MBSE Tools & Software related to the Space Industry

A list of available free and commercial MBSE software by actors in the space industry:

	Virtual Satellite	VSD	RangeDB	valispace
Developer	DLR	ESA	Airbus	valispace
Status	available, open-source	available, ESA community	not available	commercial, charged
Link	[a]	[b]	[25]	[c]
ECSS data structure	×	✓	✓extended	×
Lifecycle Phase Focus	conceptual design	whole lifecycle	whole lifecycle	whole lifecycle
Modeling Focus	behavior & geometry	system level design	spacecraft reference database	documents, engineering data
CE support	✓	×	×	✓

---

[a] `https://github.com/virtualsatellite/VirtualSatellite4-Core`, accessed 17.02.2021

[b] `https://www.vsd-project.org/`, accessed 17.02.2021

[c] `https://www.valispace.com/`, accessed 17.02.2021

B.2 General MBSE Tools & Software

An overview of more general MBSE software that could also be used in the space domain:

	Developer	Status	Link	Focus	Note
3DExperience	Daussault Systèmes	commercial, charged	[a]	Platform solution	offers marketplace
T-Flex	Top Systems	commercial, charged	[b]	whole suit, e.g. Flex CAD avail.	
TeamWork Cloud	NoMagic	commercial, charged	[c]	repository for collaborative development	storage for Cameo &MagicDraw models
Capella	Eclipse	free, open source	[d]	MBSE tool	Add-ons available
CAMEO Systems Modeler	NoMagic	commercial, charged	[e]	MBSE platform	
Papyrus	Eclipse	free, open source	[f]	Modeling Environment	JVM required
Enterprise Architect	SPARX Systems	commercial, charged	[g]	System architecture	Behavioral code editor

---

[a] `https://www.3ds.com/de/3dexperience`, accessed 17.02.2021

[b] `https://tflex.com/`, accessed 17.02.2021

[c] `https://www.nomagic.com/products/teamwork-cloud`, accessed 17.02.2021

[d] `https://www.eclipse.org/capella/`, accessed 17.02.2021

[e] `https://www.nomagic.com/products/cameo-systems-modeler`, accessed 17.02.2021

[f] `https://www.eclipse.org/papyrus/`, accessed 17.02.2021

[g] `https://sparxsystems.com/products/ea/15/index.html`, accessed 17.02.2021

C Ontologies

C.1 General Ontologies

This is a list of websites that provide either their own ontology or list others:

- W3C
  `https://www.w3.org/wiki/Lists_of_ontologies`
  A list of general ontologies and links to other lists.

- Protege
  `https://protegewiki.stanford.edu/wiki/Protege_Ontology_Library`
  A list of ontologies in the OWL format.

- DBpedia Ontology
  `https://wiki.dbpedia.org/services-resources/ontology`
  This ontology contains all information that appears in info-boxes on Wikipedia. It has around 4,233,000 instances.

C.2 Space Industry-Related Ontologies

A list of space industry-related ontologies:

- IMCE Ontologies for Model-based Systems Engineering
  `https://github.com/JPL-IMCE/gov.nasa.jpl.imce.ontologies.public`
  This repository contains the Integrated Model-Centric Engineering (IMCE) ontologies for MBSE by NASA' Jet Propulsion Lab.

- Model-Based Space Engineering Ontology 1.1 [51]
  `https://zenodo.org/record/57955`
  This ontology describes the design of a space system for MBSE. It can be used to specify the system's architecture, requirements, design, and functional verification.

- The Orbital Space Domain Knowledge Modelling Project - Ontologies for Astronautics by Robert J. Rovetto
  `https://rrovetto.github.io/Orbital-Space-Ontology-Project/`
  This is a collection of ontologies under development by Robert Rovetto related to the astronautical domain.

- ConTrOn Spacecraft parts ontology [23]
  `https://zenodo.org/record/3862854`
  This repository holds the spacecraft pats ontology that was developed for ConTrOn [87].

- MagSat Dataset [50]
  `https://zenodo.org/record/50671`
  This data set contains the design of a spacecraft that was specified with the Model-Based Space Engineering Ontology.

C.3 Ontology-Related Tools

A short list of ontology tools:

- Space Lexicon Generator
  `https://github.com/strath-ace/smart-nlp/tree/master/SpaceLexiconGenerator`
  Here, the code described in [7] is stored. It can be used to semi-automatically generate a domain-specific lexicon for ontologies from texts.
- Migration of Engineering Models to Knowledge Graphs
  `https://github.com/strath-ace/smart-nlp/tree/master/EngineeringModelsMigration`
  This repository holds the code used in [10]. It contains a pipeline that converts engineering models compliant to ECSS-E-TM-10-25A into a knowledge graph.
- Protégé
  `https://protege.stanford.edu/`
  Protégé ia a free, open-source ontology editor.

D Artificial Intelligence

D.1 AI Tools

A short list of intelligent tools within the space domain:

- Topic Modeling for Space Mission Requirement Categorisation
  `https://github.com/strath-ace/smart-nlp/tree/master/TopicModeling`
  Code for [6] that identifies and extracts topics from documents.
- Resources of the intelligent assistant Daphne [3]:
  - Online Demo
    `https://www.selva-research.com/daphne/`
  - Code
    `https://github.com/seakers/daphne_brain`
- Evolutionary System Design Converger
  `https://github.com/aerospaceresearch/ESDC`
  Evolutionary algorithm to optimize the design of a satellite as described in [24].

E Machine-Readable Documents & Electronic Data Sheets

E.1 Tools for EDS

Here, useful tools for electronic data sheets are listed:

- satsearch
  `https://satsearch.co`
  Satsearch is a website that digitizes data sheets of satellite parts into electronic data sheets and connects potential buyers with suppliers.

- satsearch API
  `https://api.satsearch.co/`
  The API to get details of a product available on satsearch.co.

- Data Sheets Annotation Tool
  `https://gitlab.com/kobkaew/dsat-client`
  The Implementation of the DSAT tool as described in [88].

- CCSDS EDS reference tooling by ESA
  `https://essr.esa.int/project/ccsds-eds-reference-tooling`
  This is a tool published by ESA to work with electronic data sheet as defined in their CCSDS standards.

F Standardization

F.1 Consortia & Organizations

A list of (space-related) standardization organizations and consortia:

- **ISO**
  `https://www.iso.org`
  The International Standardization Organization (ISO) develops, publishes, and sells access to standards and norms.

- **CCSDS**
  `https://public.ccsds.org`
  The Consultative Committee for Space Data Systems (CCSDS) is an international forum of space agencies. They develop and publish standard for space data and information systems.

- **ECSS**
  `https://ecss.nl`
  The European Cooperation for Space Standardization (ECSS) was established by ESA in an effort to standardize European spaceflight. The ECSS publishes standards, handbooks, and a space glossary.

- **SAVOIR**
  `https://savoir.estec.esa.int`
  The Space Avionics Open Interface Architecture (SAVOIR) is an initiative by ESA to standardize spacecraft aviation systems. SAVOIR works together with other standardization organizations, such as ECSS and CCSDS.
- **INCOSE Space Systems**
  `https://www.incose.org`
  INCOSE Space Systems is a working group of the International Council on Systems Engineering (INCOSE). INCOSE develops principles and guidelines of systems engineering. The Space Systems groups aims to expand the advantages of systems engineering to the space sector.
- **MB4SE**
  `https://mb4se.esa.int/MB4SE_Home.html`
  Model Based 4(for) System Engineering is an initiative by ESA to promote MBSE and its benefits in the European space sector.
- **OSMoSE**
  `https://mb4se.esa.int/OSMOSE_Main.html`
  The Overall Semantic Modelling for System Engineering (OSMoSE) is an initiative by ESA in order to improve and standardize the knowledge and information exchange during the systems engineering process of spacecrafts. OSMoSE develops the Space Systems Ontology, which covers the system aspects over the whole life-cycle of a spacecraft.

F.2 Standards

A (non-exhaustive) list of space industry-related standards:

- **CCSDS**
  - CCSDS-311.0-M-1 [16] – Reference Architecture for Space Data Systems, recommended practice
    It can be used for the description of data system architectures and high-level designs within the space domain.
  - CCSDS 876.0-B-1 [18] –XML Specification for Electronic Data Sheets, recommended standard
    This standard defines XML specification for SOIS electronic data sheets for onboard devices. The XML file is available on the CCSDS SANA registry[1]
  - CCSDS 876.1-R-2 [17] – Specification for Dictionary of Terms for Electronic Data sheets, draft recommended practice
    This describes the dictionary of terms for EDS of SOIS-compliant services.

---

[1] `https://sanaregistry.org/r/sois`, accessed 06.01.2021

- **ECSS**
  - ECSS-E-ST-10C [32] Space engineering - System engineering general requirements
    The document describes all key elements of project planning and implementation to identify top-level requirements and products.
  - ECSS-E-ST-40-07 [33] Space engineering – Simulation modelling platform standard
    This is a standard based on ECSS-E-ST-40 for the engineering of simulation software.
  - ECSS-E-TM-10-20A [29] Space engineering - product data exchange
    This defines the methods and protocols for the exchange of machine-interpretable product data of space projects.
  - ECSS-E-TM-10-21 [30] System modelling and simulation
    This document describes how to use system simulation to support system engineering tasks.
  - ECSS-E-TM-10-23 [31] Space Systems Data Repository
    This specifies the semantics of the data needed during engineering processes as specified in ECSS standards. Thus, it enables the concept of a space system data repository to gather all engineering data produced during the life-cycle of a space system.
  - ECSS-E-TM-10-25A [28] Engineering design model data exchange – CDF
    The standard offers recommendations for model-based data exchange for the early design phases.
  - ECSS-M-ST-10C Rev.1 [27] Space project management - Project planning and implementation
    This document describes the system engineering implementation requirements for space systems and space product development. It also defines the life-cycle stages of a space product.
- **ISO**
  - ISO 10303 [91]
    Standard for the exchange of product model data (STEP), which is organized in several application protocols. For example AP203 [62]: Configuration Controlled Design and AP 209 [61]: Managed model-based 3D engineering are used in the aerospace sector.
- **INCOSE Space Systems**
  - CubeSat System Reference Model (CSRM) [67]
    Reference model for CubeSats for the design, verification, and validation of designs based on MBSE principles.

References

[1] Rebeca Arista, Fernando Mas, Manuel Oliva, and Domingo Morales-Palma. Applied Ontologies for Assembly System Design and Management within the Aerospace Industry. In *Proceedings of the Joint Ontology Workshops 2019*, volume 2518, 2019.

[2] Anna Bahnmüller, Azeem Syed Muhammad, Georgia Albuquerque, and Andreas Gerndt. Evaluation of Interaction Techniques for Early Phase Satellite Design in Immersive AR. In *2020 IEEE Aerospace Conference*. IEEE, August 2020.

[3] Hyunseung Bang, Antoni Virós Martin, Arnau Prat, and Daniel Selva. Daphne: An intelligent assistant for architecting earth observing satellite systems. In *2018 AIAA Information Systems-AIAA Infotech@ Aerospace*, page 1366. American Institute of Aeronautics and Astronautics, 2018.

[4] Tanja Banner. "Starlink"-Satelliten von SpaceX: Das steckt hinter dem Projekt von Elon Musk. `https://www.fr.de/wissen/starlink-satelliten-spacex-elon-musk-projekt-internet-weltall-falcon-9-13699873.html`, 2020. Accessed 08.01.2021.

[5] Artur Baranowski, Sebastian Utzig, Philipp M. Fischer, Andreas Gerndt, and Jens Herder. 3D spacecraft configuration using immersive AR technology. In *Workshop Proceedings/Tagungsband: Virtuelle und Erweiterte Realität–15. Workshop der GI-Fachgruppe VR/AR*, pages 71–82. Shaker Verlag, 2018.

[6] Audrey Berquand, Iain McDonald, Annalisa Riccardi, and Yashar Moshfeghi. The Automatic Categorisation of Space Mission Requirements for the Design Engineering Assistant. In *70th International Astronautical Congress*, 2019.

[7] Audrey Berquand, Yashar Moshfeghi, and Annalisa Riccardi. Space mission design ontology: extraction of domain-specific entities and concepts similarity analysis. In *AIAA Scitech 2020 Forum*. American Institute of Aeronautics and Astronautics, January 2020.

[8] Audrey Berquand, Francesco Murdaca, Annalisa Riccardi, Tiago Soares, Sam Generé, Norbert Brauer, and Kartik Kumar. Artificial intelligence for the early design phases of space missions. In *2019 IEEE Aerospace Conference*, pages 1–20. IEEE, 2019.

[9] Audrey Berquand, Francesco Murdaca, Annalisa Riccardi, Tiago Soares, Sam Gerené, Norbert Brauer, and Kartik Kumar. Towards an artificial intelligence based design engineering assistant for the early design of space missions. In *69th International Astronautical Congress*, 2018.

[10] Audrey Berquand and Annalisa Riccardi. From engineering models to knowledge graph: delivering new insights into models. In *9th International Systems & Concurrent Engineering for Space Applications Conference (SECESA 2020)*, 2020.

[11] M. Bieze. MARVL-Model-based requirements-verification lifecycle. *Software Systems Division (TEC-SW) and Data Systems Division (TEC-ED) Final Presentation Day*, 2018.

[12] Graham Bleakley. The role of Simulation and AI in the implementation of a Digital Twin. *INCOSE Workshop*, 2020.

[13] Olivia Borgue, Massimo Panarotto, and Ola Isaksson. Modular product design for additive manufacturing of satellite components: maximising product value using genetic algorithms. *Concurrent Engineering*, 27(4):331–346, 2019.

[14] B. Braun and M. Braun. Space and the Internet of Things. *2020 IEEE Aerospace Conference*, pages 1–9, 2020.

[15] A. E. Burov and O. G. Burova. Development of digital twin for composite pressure vessel. In *Journal of Physics: Conference Series*, volume 1441, page 012133. IOP Publishing, 2020.

[16] CCSDS. Reference architecture for space data systems (CCSDS 311.0-M-1). techreport CCSDS 311.0-M-1, CCSDS, 2008.

[17] CCSDS. Spacecraft onboard interface services – Specification for dictionary of terms for electronic datasheets (CCSDS 876.1-R-2). techreport CCSDS 876.1-R-2, CCSDS, 2016.

[18] CCSDS. Spacecraft onboard interface services – XML specification for electronic datasheets (CCSDS 876.0-B-1). techreport CCSDS 876.0-B-1, CCSDS, 2019.

[19] Amaresh Chakrabarti, Prabir Sarkar, B. Leelavathamma, and B. S. Nataraju. A functional representation for aiding biomimetic and artificial inspiration of new ideas. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 19(2):113–132, 2005.

[20] Marc-Andre Chavy-Maconald and Jean-Paul Kneib. Concurrent Design & Data Facility (CD2F): Conceptualizing a Next-Generation CDF. *SECESA*, 2020.

[21] Mark Chodas, Rebecca Masterson, and Olivier de Weck. Addressing Deep Uncertainty in Space System Development through Model-based Adaptive Design. In *2020 IEEE Aerospace Conference*, pages 1–17. IEEE, 2020.

[22] Ana Cipriano and Robin Biesbroek. A Survey of Augmented Reality for Use in the Concurrent Design Facility. *SECESA*, 2018.

[23] ConTrOn. ConTrOn - Spacecraft parts ontology. 10.5281/zenodo.3862854, May 2020.

[24] Manfred Ehresmann, Martin Fugmann, Jonathan Skalden, Daniel Galla, Christoph Montag, Georg Herdrich, Stefanos Fasoulas, Sabine Klinkner, and Tina Stäbler. IRAS: Progress in Development of the Digital Concurrent Engineering Platform, Software Tools and Innovative Technologies. In *69th International Astronautical Congress*, 2019.

[25] Harald Eisenmann, Claude Cazenave, and Thierry Noblet. RangeDB the product to meet the challenges of nowadays system database. In *9th ESA Workshop on Simulation for European Space Programmes*, 2015.

[26] European Space Agency (ESA). What is space 4.0? `http://www.esa.int/About_Us/Ministerial_Council_2016/What_is_space_4.0`, 2017. Accessed: 11.12.2020.

[27] ESA-ESTEC. Space project management - Project planning and implementation. Technical Report ECSS-M-ST-10C, ECSS, 2009.

[28] ESA-ESTEC. Space Engineering - Engineering design model data exchange (ECSS-E-TM-10-25A). Technical Report ECSS-E-TM-E-10-25A, ECSS, 2010.

[29] ESA-ESTEC. Space Engineering - Product data exchange (ECSS-E-TM-10-20A). Technical Report ECSS-E-TM-E-10-20A, ECSS, 2010.

[30] ESA-ESTEC. System modelling and simulation (ECSS-E-TM-10-21A). Technical Report ECSS-E-TM-10-21A, ECSS, 2010.

[31] ESA-ESTEC. Space system data repository (ECSS-E-TM-10-23A). Technical Report ECSS-E-TM-10-23A, ECSS, 2011.

[32] ESA-ESTEC. System engineering general requirements (ECSS-E-ST-10C). Technical Report ECSS-E-ST-10C, ECSS, 2017.

[33] ESA-ESTEC. Simulation modelling platform (ECSS-E-ST-40-07). Technical Report ECSS-E-ST-40-07, ECSS, 2020.

[34] Richard Ferguson, Joseph Marshall, and Maria Guzman. Adapting Progressive MBSE Development to Document Based Contracts. In *2020 IEEE Aerospace Conference*, pages 1–11. IEEE, 2020.

[35] Ivo Ferreira, Hans Peter de Koning, Sam Gerene, Andrew Pickering, Johan Vennekens, and Frederike Beyer. Open Concurrent Design Tool: ESA Community Open Source Ready to Go! 6th International Workshop on System & Concurrent Engineering for Space Applications (SECESA), October 2014.

[36] Philipp Martin Fischer, Meenakshi Deshmukh, Volker Maiwald, Dominik Quantius, Antonio Martelo Gomez, and Andreas Gerndt. Conceptual data model: A foundation for successful concurrent engineering. *Concurrent Engineering*, 26(1):55–76, November 2017.

[37] Philipp Martin Fischer, Daniel Lüdtke, Caroline Lange, Frank-Cyrus Roshani, Frank Dannemann, and Andreas Gerndt. Implementing Model-Based System Engineering for the Whole Lifecycle of a Spacecraft. *CEAS Space Journal*, 9(3):351–365, July 2017.

[38] Clément Fortin, Grant McSorley, Dominik Knoll, Alessandro Golkar, and Ralina Tsykunova. Study of Data Structures and Tools for the Concurrent Conceptual Design of Complex Space Systems. In *Product Lifecycle Management and the Industry of the Future*, pages 601–611. Springer International Publishing, 2017.

[39] Frederico Freitas and Fernando Sousa. AR4Telecom - Showcasing AR applied to AIT procedures. *AR/VR For European Space Programs@ESA-ESTEC*, 2019.

[40] Sanford Friedenthal, Regina Griego, and Mark Sampson. INCOSE model based systems engineering (MBSE) initiative. In *INCOSE 2007 symposium*, volume 11, 2007.

[41] Gèrald Garcia, Raymond Sougmagne, Règis De Ferluc, and Jèrèmy Roumegue. Digital Transformation of Engineering Exchanges between a Customer and a Supplier. *SECESA*, 2020.

[42] Jean-Luc Garnier, Helene Bachatene, and Pierre Nowodzienski. Architecture Patterns for Digital Twins in Space Applications. *SECESA*, 2020.

[43] Paolo Gaudenzi, Luca Lampani, Marco Eugeni, Franscesco Costantino, Alberto Boschetto, Alberto Marchietti Spaccamela, Massimo Meccella, Leonardo Querzoni, Ralf Usinger, Marco Manuel Ivagnes, Tatiana Quercia, Manuel Alprandi, and Andrei Stancu. Smart manufacturing in the framework of space industry. An industry 4.0 approach to large scale production of satellite constellations. *International Astronautical Congress*, 2020.

[44] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning.* MIT Press, 2016. `http://www.deeplearningbook.org`, accessed 17.02.2021.

[45] John M. Gregory, Jin S. Kang, Robert Bruninga, Ronald M. Sega, John M. Borky, David Kaslow, and George H. Downey. Applying a Model-Based Approach to Develop a Standardized Template for CubeSat-class Satellites. In *2020 IEEE Aerospace Conference*, pages 1–11. IEEE, 2020.

[46] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220, 1993.

[47] Cesare Guariniello, Thomas B. Marsh, Ron Porter, Christopher Crumbly, and Daniel A. DeLaurentis. Artificial Intelligence Agents to Support Data Mining for SoS Modeling of Space Systems Design. In *2020 IEEE Aerospace Conference*, pages 1–11. IEEE, 2020.

[48] Tyler M. Harris and Amy E. Landis. Life Cycle Assessment: A Tool to Help Design Environmentally Sustainable Space Technologies. In *2020 IEEE Aerospace Conference*, pages 1–11. IEEE, 2020.

[49] C. Hennig. *From Data Modeling to Knowledge Engineering in Space System Design.* phdthesis, KIT, 2017.

[50] Christian Hennig and Harald Eisenmann. MagSat Dataset. 10.5281/zenodo.50671, April 2016.

[51] Christian Hennig and Harald Eisenmann. Model-Based Space Engineering Ontology 1.1. 10.5281/zenodo.57955, July 2016.

[52] Christian Hennig, Alexander Viehl, Benedikt Kämpgen, and Harald Eisenmann. Ontology-Based Design of Space Systems. In *Lecture Notes in Computer Science*, pages 308–324. Springer International Publishing, 2016.

[53] J. Hoffmann, R. Bertrand, and S. Ueltzhöffer. Concurrent Engineering for Ground Segments and Operations Conceptual Design. *Deutscher Luft- und Raumfahrtkongress 2019*, 2020.

[54] Tobias Hoppe, Harald Eisenmann, Alexander Viehl, and Oliver Bringmann. Digital Space Systems Engineering through Semantic Data Models. In *2017 IEEE International Conference on Software Architecture (ICSA)*, pages 93–96. IEEE, IEEE, April 2017.

[55] Tobias Hoppe, Harald Eisenmann, Alexander Viehl, and Oliver Bringmann. Guided systems engineering by profiled ontologies. In *2017 IEEE International Systems Engineering Symposium (ISSE)*, pages 1–6. IEEE, 2017.

[56] Tobias Hoppe, Harald Eisenmann, Alexander Viehl, and Oliver Bringmann. SEMF – The Semantic Engineering Modeling Framework - Bringing Semantics into the Eclipse Modeling Framework for Space Systems Engineering. In *Proceedings of the 5th International Conference on Model-Driven Engineering and Software Development.* SCITEPRESS - Science and Technology Publications, 2017.

[57] Tobias Hoppe, Harald Eisenmann, Alexander Viehl, and Oliver Bringmann. Shifting from data handling to knowledge engineering in aerospace industry. In *2017 IEEE International Systems Engineering Symposium (ISSE)*. IEEE, October 2017.

[58] A. V. i. Martin and D. Selva. Daphne: A Virtual Assistant for Designing Earth Observation Distributed Spacecraft Missions. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:30–48, 2020.

[59] ISO. Environmental management – Life cycle assessment – Principles and framework. Technical Report ISO 14040, International Organization for Standardization, 2006.

[60] ISO. Environmental management – Life cycle assessment – Requirements and guidelines. Technical Report ISO 14044, International Organization for Standardization, 2006.

[61] ISO. Industrial automation systems and integration – Product data representation and exchange – Part 209: Application Protocol: Multidisciplinary analysis and design. Technical Report ISO 10303:209, International Organization for Standardization, 2014.

[62] ISO. Industrial automation systems and integration – Product data representation and exchange – Part 242: Application protocol: Managed model-based 3D engineering. Technical Report ISO 10303:242, International Organization for Standardization, 2014.

[63] Curtis Iwata, Samantha Infeld, Jennifer M. Bracken, Melissa McGuire, Christina McQuirck, Aron Kisdi, Jonathan Murphy, Bjorn Cole, and Pezhman Zarifian. Model-based systems engineering in concurrent engineering centers. In *AIAA SPACE 2015 Conference and Exposition*, pages 4437–4437, 2015.

[64] Stephan S. Jahnke and Antonio Martelo. Special characteristics of concurrent engineering studies dealing with cubesatmissions and their impact on the CE process. In *7th International Systems & Concurrent Engineering for Space Applications Conference (SECESA)*, 2016.

[65] Stephan Siegfried Jahnke, Antonio Martelo Gomez, Philipp M. Fischer, and Caroline Lange. Concurrent Engineering in later project phases: current methods and future demands. In *Proceedings of the International Astronautical Congress, IAC*, 2018.

[66] Hyeon H. Jo, Hamid R. Parsaei, and William G. Sullivan. *Principles of concurrent engineering*, pages 3–23. Springer US, Boston, MA, 1993.

[67] David Kaslow and Azad M. Madni. Validation and Verification of MBSE-Compliant CubeSat Reference Model. In Azad M. Madni, Barry Boehm, Roger G. Ghanem, Daniel Erwin, and Marilee J. Wheaton, editors, *Disciplinary Convergence in Systems Engineering Research*, pages 381–393, Cham, 2018. Springer International Publishing.

[68] David Keppel. *An Empirical Exploration of the Potential Benefits of Applying Big Data Analytics for Improved Space to Space Operations Management in the Satellite Industry.* PhD thesis, Toulouse Business School, 2018.

[69] Alexander Kharlan. A Mission Analysis Tool for Aiding Feasibility Studies in Small Satellite Mission Projects. *International Astronautical Congress*, 2020.

[70] Franziska Klemme. Potenziale der ”Industrie 4.0” für die Raumfahrtindustrie am Beispiel von OHB: Welche Technologien werden die Zukunft der Satellitenproduktion verändern? Master’s thesis, Universität Bremen, 2018.

[71] Dominik Knoll, Clement Fortin, and Alessandro Golkar. Review of Concurrent Engineering Design practice in the space sector: state of the art and future perspectives. In *2018 IEEE International Systems Engineering Symposium (ISSE)*. IEEE, October 2018.

[72] Dominik Knoll and Alessandro Golkar. A coordination method for concurrent design and a collaboration tool for parametric system models. *Concurrent Engineering*, 26(1):5–21, September 2017.

[73] B. Krijnen and J. Guo. Feasibility Study on the Use of Reinforcement Learning for Automated CUBESAT Concept Generation. *SECESA*, 2020.

[74] K. Kumar, A. Vaccarella, N. P. Nagendra, Gerené, and L. Lindlad. Integrated Mission Design Using Satsearch. *SECESA*, 2018.

[75] Remi Lanza, Jochen Haenisch, Kjell Bensston, Terje Rolvag, and Matteo Appolloni. DEFINE - Multidisciplinary 3D Digital Models for AIT environments. *SECESA*, 2020.

[76] Maj Michael LaSorda, John Borky, and Ron Sega. Model-Based Systems Architecting with Decision Quantification for Cybersecurity, Cost, and Performance. In *2020 IEEE Aerospace Conference*, pages 1–13. IEEE, 2020.

[77] Karolina Latserus, Petr Mukhachev, Anton Ivanov, and Angelina Prokopeva. Model-based approach for requirements management connection with verification procedures applied to space systems. *International Astronautical Congress*, 2020.

[78] Geilson Loureiro, Alison de Oliveira Moraes, Luciana Sêda Cardoso, Sergio Fugivara, and Sonia Mara de Souza. System concurrent engineering of a mobile TT&C ground station for an unmanned aerial vehicle. *Product Management and Development*, 8, June 2010.

[79] Jacob Manning, David Langerman, Barath Ramesh, Evan Gretok, Christopher Wilson, Alan George, James MacKinnon, and Gary Crum. Machine-learning space applications on smallsat platforms with tensorflow. *Small Satellite Conference 2018*, 2018.

[80] Antonio Martelo, Stephan S. Jahnke, Andy Braukhane, Dominik Quantius, Volker Maiwald, and Oliver Romberg. Statistics and Evaluation of 60+ Concurrent Engineering Studies at DLR. *68th International Astronautical Congress (IAC)*, 2017.

[81] Pat B. McLaughlan and Lorie R. Grimes-Ledesma. Composite Overwrapped Pressure Vessels, A Primer. `https://ntrs.nasa.gov/api/citations/20110008406/downloads/20110008406.pdf`, 2011. Accessed: 22.01.2021.

[82] Shaohua Meng, Shaolin Tang, Yahong Zhu, and Changyu Chen. Digital Twin-Driven Control Method for Robotic Automatic Assembly System. *IOP Conference Series: Materials Science and Engineering*, 493:012128, March 2019.

[83] F. Murdaca, A. Berquand, K. Kumar, A. Riccardi, T. Soares, S. Gerené, and N. Brauer. Knowledge-based information extraction from datasheets of space parts. In *8th International Systems & Concurrent Engineering for Space Applications Conference*, September 2018.

[84] F. Murdaca, A. Berquand, A. Riccardi, T. Soares, S. Gerené, N. Brauer, and K. Kumar. Artificial intelligence for early design of space missions in support of concurrent engineering sessions. In *8th International Systems & Concurrent Engineering for Space Applications Conference*, 2018.

[85] Laurence Nardon. New Space: The Impact of the Digital Revolution on Space Actors and Policies in Europe. Notes de l'Ifri, Ifri. `https://www.ifri.org/sites/default/files/atoms/files/nardon_new_space_digital_revolution_2017.pdf`, 2017. Accessed: 07.01.2021.

[86] Johannes Norheim. Satellite Component Selection with Mixed Integer Nonlinear Programming. In *2020 IEEE Aerospace Conference*, pages 1–9. IEEE, 2020.

[87] Kobkaew Opasjumruskit, Diana Peters, and Sirko Schindler. Contron: Continuously trained ontology based on technical data sheets and wikidata. *arXiv preprint arXiv:1906.06752*, 2019.

[88] Kobkaew Opasjumruskit, Diana Peters, and Sirko Schindler. DSAT: Ontology-based Information Extraction on Technical Data Sheets. In *ISWC 2020*, November 2020.

[89] Sweety Pate. AI-Powered & Cloud Based Concurrent Systems Engineering Platform. *SECESA*, 2020.

[90] Diana Peters, Philipp M. Fischer, Philipp M. Schäfer, Kobkaew Opasjumruskit, and Andreas Gerndt. Digital Availability of Product Information for Collaborative Engineering of Spacecraft. In *International Conference on Cooperative Design, Visualization and Engineering*, pages 74–83. Springer, 2019.

[91] Michael J. Pratt. Introduction to ISO 10303—the STEP Standard for Product Data Exchange. *Journal of Computing and Information Science in Engineering*, 1(1):102, 2001.

[92] Marek Prochazka and David Perillo. SAVOIR Electronic Data Sheet Definition: Overview and Status. 14th ESA Workshop on Avionics, Data, Control and Software Systems (ADCSS), October 2020.

[93] Li Qiao, Mahmoud Efatmaneshnik, and Michael Ryan. A combinatorial approach to tradespace exploration of complex systems: A cubesat case study. In *INCOSE International Symposium*, volume 27, pages 763–779. Wiley Online Library, 2017.

[94] A. L. Ramos, J. V. Ferreira, and J. Barcelo. Model-Based Systems Engineering: An Emerging Approach for Modern Systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(1):101–111, 2012.

[95] Máximo A. Roa, Korbinian Nottensteiner, Gerhard Grunwald, Pablo Lopez Negro, Aurelian Cuffolo, Sabrine Andiappane, Mathieu Rognant, Antoine Verhaeghe, and Vincent Bissonnette. In-Space Robotic Assembly of Large Telescopes. In *15th Symposium on Advanced Space Technologies in Robotics and Automation, ASTRA*, pages 27–28, 2019.

[96] Màximo A. Roa, Korbinian Nottensteiner, Armin Wedler, and Gerhard Grunwald. Robotic Technologies for In-Space Assembly Operations. In *14th Symposium on Advanced Space Technologies in Robotics and Automation*, June 2017.

[97] Enea Sacco and Seung Ki Moon. Additive manufacturing for space: status and promises. *The International Journal of Advanced Manufacturing Technology*, 105(10):4123–4146, 2019.

[98] M. Samoil, M. Mulser, S. Jahnke, M. Rohrbeck, P. Vermeer, A. Wyen, R. Palicka, J. Slavik, S. Gerené, M. Fijneman, I. Roma, and L. Pambaguian. Concurrent Engineering Method to Implement Advanced Manufacturing Technologies for Spacecraft Design. *SECESA*, 2020.

[99] Kurt Schwenk and Daniel Herschmann. On-board data analysis and real-time information system. *Deutscher Luft- und Raumfahrtkongress*, 2020.

[100] Steffen Staab and Rudi Studer. *Handbook on Ontologies.* Springer Publishing Company, Incorporated, second edition, 2009.

[101] Jan-Philipp Stauffert, Florian Niebling, and Marc Erich Latoschik. Latency and Cybersickness: Impact, Causes, and Measures. A Review. *Frontiers in Virtual Reality*, 1:31, 2020.

[102] Robert Stevens. Weaving a Digital Thread into Concept Design. In *2020 IEEE Aerospace Conference*, pages 1–7. IEEE, 2020.

[103] Bo Sun, Lei Zhang, and Yongheng Chen. Design of big data processing system for spacecraft testing experiment. In *2017 7th IEEE international symposium on microwave, antenna, propagation, and EMC technologies (MAPE)*, pages 164–167. IEEE, 2017.

[104] S. K. Tadeja, Pranay Seshadri, and Per Ola Kristensson. AeroVR: An immersive visualisation system for aerospace design and digital twinning in virtual reality. *The Aeronautical Journal*, 124(1280):1615–1635, 2020.

[105] Massimo Tipaldi, Lorenzo Feruglio, Pierre Denis, and Gianni D'Angelo. On applying AI-driven flight data analysis for operational spacecraft model-based diagnostics. *Annual Reviews in Control*, 2020.

[106] G. Tomasicchio, A. Ceccarelli, A. De Matteis, and L. Spazzacampagna. A Concurrent and Collaborative Design Facility for End-to-End Space Mission Modelling and Design: An Integrated Approach from Mission Requirements to Satellite Operations and Service Delivery Engineering. *SECESA*, 2020.

[107] Antoni Viros Martin and Daniel Selva. From design assistants to design peers: Turning daphne into an AI companion for mission designers. In *AIAA Scitech 2019 Forum*, page 0402, 2019.

[108] Thiago Weber Martins and Reiner Anderl. Digital Twins for Space Factory 4.0. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume Volume 1: 39th Computers and Information in Engineering Conference, August 2019.

[109] Thiago Weber Martins, Aaron Pereira, Thomas Hulin, Oliver Ruf, Stefan Kugler, Alessandro Giordano, Ribin Balachandrand, Fabian Benedikt, John Lewis, Reiner Anderl, et al. Space Factory 4.0-New processes for the robotic assembly of modular satellites on an in-orbit platform based on ”Industrie 4.0” approach. In *Proceedings of the International Astronautical Congress, IAC*, 2018.

[110] Andrew Ross Wilson. *Advanced Methods of Life Cycle Assessment for Space Systems.* PhD thesis, University of Strathclyde, Glasgow, 2019.

[111] Andrew Ross Wilson and Massimiliano Vasile. Integrating life cycle assessment of space systems into the concurrent design process. *68th International Astronautical Congress (IAC)*, 2017.

[112] X. Wu, X. Zhu, G. Wu, and W. Ding. Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering*, 26(1):97–107, 2014.

[113] Mengyun Yue, Chen Huang, Suming Zhang, and Hao Xu. Design and verification of a rocket's mirroring system based on digital twin technology. *71th International Astronautical Congress (IAC)*, 2020.

[114] Xiao Zhang, Panfeng Wu, and Chao Tan. A big data framework for spacecraft prognostics and health monitoring. In *2017 Prognostics and System Health Management Conference (PHM-Harbin)*, pages 1–7. IEEE, 2017.

[115] Hanmo Zhao, Zhaowei Sun, and Hongzhu Zhang. Application of knowledge engineering in spacecraft overall design. In *Journal of Physics: Conference Series*, volume 1510, page 012015, 2020.

[116] Thomas Zimmermann, Marvin Schmidt, Kathrin Konkol, and Rainer Stark. Change Impact Anlysis with Sensitivity Filtered AR/VR for Design Reviews. *SECESA*, 2020.