Deutscher Luft- und Raumfahrtkongress 2016
DocumentID: 420085

IMPLEMENTING MODEL BASED SYSTEM ENGINEERING FOR THE WHOLE LIFECYCLE OF A SPACECRAFT

P. M. Fischer[1)], D. Lüdtke[1)], C. Lange[2)], F.-C. Roshani[3)], F. Dannemann[2)] and A.Gerndt[1)],
[1)] DLR – Software for Space Systems and Interactive Visualization,
Lilienthalplatz 7, 38108 Braunschweig, Germany
[2)] DLR – Institute for Space Systems,
Robert-Hooke-Str. 7, 28359 Bremen, Germany
[3)] DLR – Space Operations and Astronaut Training,
Münchener Straße 20, 82234 Weßling, Germany

Abstract

Design information of a spacecraft is collected over all phases in the lifecycle of a project. A lot of this information is exchanged between different engineering tasks and business processes. In some lifecycle phases Model Based System Engineering (MBSE) has introduced system models and databases that help to organize such information and to keep it consistent for everyone. Nevertheless, none of the existing databases approached the whole lifecycle yet. Virtual Satellite is the MBSE database developed at DLR. It has been used for quite some time in Phase A studies and is currently extended for implementing it in the whole lifecycle of spacecraft projects. Since it is unforeseeable which future use-cases such a database actually needs to support in all these different projects, the underlying data model has to provide tailoring and extension mechanisms to its Conceptual Data Model (CDM). These mechanisms as they are implemented in Virtual Satellite enable extending the CDM along the project without corrupting already stored information. As an upcoming major use-case, Virtual Satellite will be implemented as MBSE tool in the S2TEP project. This project provides a new satellite bus for internal research and several different payload missions in the future. Virtual Satellite will be used to manage configuration control problems associated with such a multi mission platform. The S2TEP project will also start using it for collecting the first design information from concurrent engineering studies, then making use of the extension mechanisms of the CDM to introduce further information artefacts such as functional electrical architecture, thus linking more and more processes into an integrated MBSE approach.

1. INTRODUCTION TO MODEL BASED SYSTEM ENGINEERING IN THE LIFECYCLE OF A MULTI MISSION SPACECRAFT PROJECT

Designing, constructing and operating spacecraft is organized and structured in various different phases. They are known as project lifecycle phases and described by respective standards of the European Cooperation for Space Standardization (ECSS) and the European Space Agency (ESA) as well as the National Aeronautics and Space Administration (NASA). This described lifecycle starts with a Phase 0 aka. Pre Phase A and is followed by A, B, C, D, E and ends with Phase F. Each Phase describes a certain set of engineering tasks that have to be fulfilled as well as goals in terms of defined deliverables and design reviews. In the early phases, a spacecraft project starts with feasibility studies followed by detailed design work, which is addressed in Phases A and B. This is followed by development as well as assembly, integration and test (AIT) activities in Phase C and D. Finally, the spacecraft is operated in Phase E and disposed later in Phase F [1] [2].

Currently, the German Aerospace Center (DLR) is developing a Small Satellite Technology Experiment Platform (S2TEP). Goal of this project is to develop a small multi mission platform and to provide it to various customers to integrate their payloads to it as well as using it for future research on the platform bus itself. Within this project, DLR occupies the unique position of planning, building and operating the satellite bus. Later, either DLR or a customer will plan, build and operate a mission consisting of both the S2TEP satellite bus and the payload together. This multi-mission scenario raises several challenges in terms of system engineering and configuration control. For instance, the interface between the bus and platform needs to be as generic as possible. It has to provide a common set of functionality that is customizable and flexible enough for all various future missions. Additionally, it needs to be considered that the interface evolves over time the same way as the S2TEP satellite bus eventually evolves [3].

To deal with these challenges, it is intended to implement Model Based System Engineering (MBSE) into the S2TEP project. The software Virtual Satellite will serve as system engineering database and central source of knowledge for several engineering tasks and processes linking to this information hub. In the beginning, the database will be applied to support system engineering in some critical and already identified fields. Later in the project, new use-cases, such as dealing with harness management, will be evaluated, implemented into it and integrated to their associated engineering processes. Introducing MBSE step-by-step into the whole lifecycle of a project is new compared to existing approaches and requires new ways of describing the data model of a systems engineering database in order to technically enable this approach.

This paper elaborates on the vision of implementing Virtual Satellite as MBSE tool in the whole lifecycle of the S2TEP project. Section 2 gives an overview of related work in the field of MBSE and spacecraft engineering in Europe. Additionally it will provide some details to some important modelling aspects. Section 3 clarifies on the meaning of MBSE and defines it in the context of a multi-mission platform such as S2TEP. Section 4 discusses the vision of MBSE over the whole lifecycle of a spacecraft and the corresponding conceptual and technical needs, whereas Section 5 provides information of how MBSE and Virtual Satellite will actually be implemented in the S2TEP project. The paper finally concludes and outlines some future activities in Section 6.

2. RELATED WORK IN THE EUROPEAN LANDSCAPE OF MBSE FOR SPACE SYSTEMS

MBSE is not a new topic within the European space community. Several MBSE related projects provide solutions to support the development in various parts of the lifecycle of a spacecraft. All of the following presented approaches provide software containing databases that allow creating a system model of the developed spacecraft. All of them also allow sharing that model with all stakeholders. One can reuse the information stored in the model either for verification or in various places during development by interfacing it to other processes. To ensure semantic correctness of the stored information, these databases provide Conceptual Data Models (CDM), in computer science often referred to as meta-models. These CDM provide the common language to all stakeholders in a spacecraft project to describe the system.

Two software frameworks target the early phases of a spacecraft: the Open Concurrent Design Tool (OCDT) [4] and Virtual Satellite. OCDT can be seen as a tool that implements the CDM described by the ECSS Technical Memorandum E-TM-10-25 [5]. Accordingly, it tries to create a common understanding of Phase A studies within the European space community. Based on the E-TM-10-25, it will help to ease the exchange of information across agencies, companies and research institutions. Until today, it does not fully address the transition of gathered information towards Phase B.

Virtual Satellite has been developed and is used in parallel. Rather than focusing on information exchange across big entities, its major focus is on research, in particular on exchanging design information along the project lifecycle. Under constant development, it allows adjusting to new processes within DLR’s Concurrent Engineering Facility (CEF) or implementing new methods of interacting with or understanding captured data. Among others, these processes and methods cover the integration of virtual reality [6] as well as formal methods as a continuous verification tool [7].

Both tools, OCDT and Virtual Satellite, are easy to use and integrate smoothly into the process of the CEF, which is used for early spacecraft design. This is reflected in the data model as well. Accordingly, both underlying CDM provide capabilities to capture the product structure as well as key performance indicators of such concurrent engineering studies such as masses of equipment within the system model.

Looking to Phase B, C and D, ESA was running the Virtual Spacecraft Design (VSD) project to develop a method and framework to improve exchange and organization of information. This has been captured in the Technical Memorandum E-TM-10-23 [8] [9]. With emphasis to exchange information during these phases, one major aspect is to support AIT. It means that information already used for the design is also important for activities such as simulator configuration. This requires information such as the functional electrical architecture as well as operational behavior of the spacecraft and its single components. For example, Interfaces or state machines can technically express such aspects. The modelled information is organized in product structures as well, but additionally, VSD delivers configuration control mechanisms that allow reusing information and restructuring it for specific applications. It allows defining single components such as a reaction wheel of the spacecraft that can be placed into different spacecraft configurations such as one for an engineering model or a configuration for a simulator bench. Information of the reaction wheel that has been specified on product level is reused and inherited by the spacecraft configurations. This approach centralizes common information such as the mass of the reaction wheel to one central point in the database. Changing this information can be automatically propagated into the various different configurations.

Within the project European Ground Segment Common Core (EGS-CC), a new ground segment infrastructure is defined and developed. This work includes the definition of a CDM to improve information exchange and configuration of ground segments, in particular between AIT and operations [10] [11]. In contrast to VSD, this project focuses on bridging the phases from AIT to operations. Accordingly, its major field of application is modelling and exchange of telemetry and telecommand (TMTC) including functional electrical architecture and incorporated standards such as the Packet Utilization Standard (PUS).

RangeDB is one of the industrial implementations of a database and is developed at Airbus Defence and Space. RangeDB is based on the outcomes of E-TM-10-23 and the underlying CDM is developed closely to the CDM of EGS-CC to ease the data exchange to Central Checkout Systems (CCS) for example. Accordingly, it is well suited for the Spacecraft References Data Base (SRDB) use-case of handling TMTC but it is clearly not limited to it. It is developed with a focus on project-specific tailoring to change the CDM when needed. The database can be efficiently integrated into the spacecraft development processes since it provides several interfaces to other databases and tools.

Even though all databases implement individual CDM, most of them are complemented by incorporating standards such as the Quantities, Units, Dimensions and Values (QUDV) [12]. These standards are an important asset to normalize the interpretation of information. The QUDV standard in particular allows assigning specific units and quantity kinds to values and derives their relationship to others [13]. By this, misinterpretation can be avoided if meters to feet are automatically converted using the modelled information of their relationship. Finally, all these databases also provide capabilities for version and configuration control, either on a technical level by version control systems or on the conceptual level

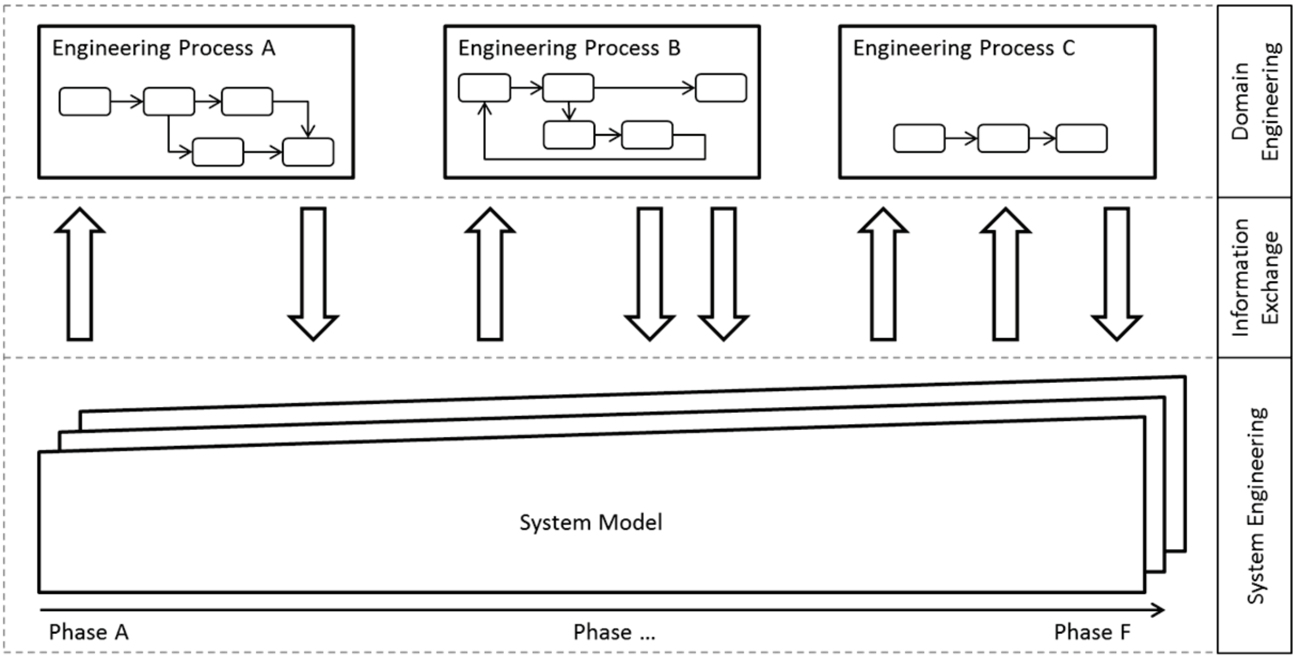

FIGURE 1 The instantiations of the Conceptual Data Model along the lifecycle phases of a spacecraft. The amount of system models relating to each other increases over time on multi-mission platforms. Various discipline-specific engineering processes are fed by and contribute back to the system model.

in the CDM by providing information inheritance mechanisms [8].

By providing such central databases in a project, it is possible to connect the various different business and engineering processes to it. This requires the development of adequate input/output (IO) mechanisms as, for example, in industrial databases. Once they are implemented, process stakeholders can read information from the system model, process it and write back results, which can be used again by other processes. Rather than just storing information for review, such as in classical document centric approaches, the information is now accessible, consistent and can be reused and refined. It also enables direct verification feedback to the design process, by analyzing the information in the database on the fly [14]. The IO capabilities have been shown in various ways such as exporting spacecraft configuration information into a mechanical computer aided design (CAD) process during concurrent engineering studies [15]. Others showed capabilities of feeding back mechanical CAD information back into the system model [16], thus closing the loop and providing the information to further computer aided engineering (CAE) steps. Some further work highlighted that it is possible to connect a simulator configuration process with its own individual database as well. In that case, the system database was used to actually update and construct parts of the simulator configuration database, while the simulator database was enriched with simulation-specific information [17].

3. DEFINITION OF MBSE IN THE CONTEXT OF A MULTI MISSION PLATFORM

Even though MBSE is very often part of today's space projects, it is rarely defined and brought into context of the project. Nevertheless, it is important to set the correct scope to make all stakeholders in the project understand what is actually meant by introducing MBSE. In general, experts involved in designing, engineering and operating the spacecraft have their specific and approved ways of working. They usually follow their well-established business and engineering processes, which require specific tools and data. For instance, considering software development activities for the onboard computer (OBC), the engineers have to follow a specific process to ensure quality and correctness of their deliverables. They have to start looking at Interface Control Documents (ICD), which describe how the software is connected to other components of the spacecraft. Following these ICDs, they have to implement the software according to the specifications for the operational behavior that may include information about TMTC used between the ground segment and the spacecraft. Once the software is built, it has to be tested and verified following standards and test specifications before it can be successfully integrated into the spacecraft.

This overall process shows a flow and dependencies of information starting from the ICD's over additional artifacts, such as TMTC, towards the final software. However, the TMTC information which has been used for developing onboard software is maybe used in other places as well, as for example stated by industrial database use-cases using the information to setup the CCS. In order to ensure that the TMTC used for implementing the software is the same TMTC as used for the configuration of the CCS, it should be handled by a consistent and central storage. This consistent storage is called the system model.

Figure 1 illustrates the aforementioned information dependencies together with the use of a system model. Three levels are depicted showing the system engineering, domain engineering and the information exchange level in between. In line with the on-board software example, the various different engineering processes provide their own individual tools but they are connected with each other. As seen, they transfer data from task to task but additionally, they either require

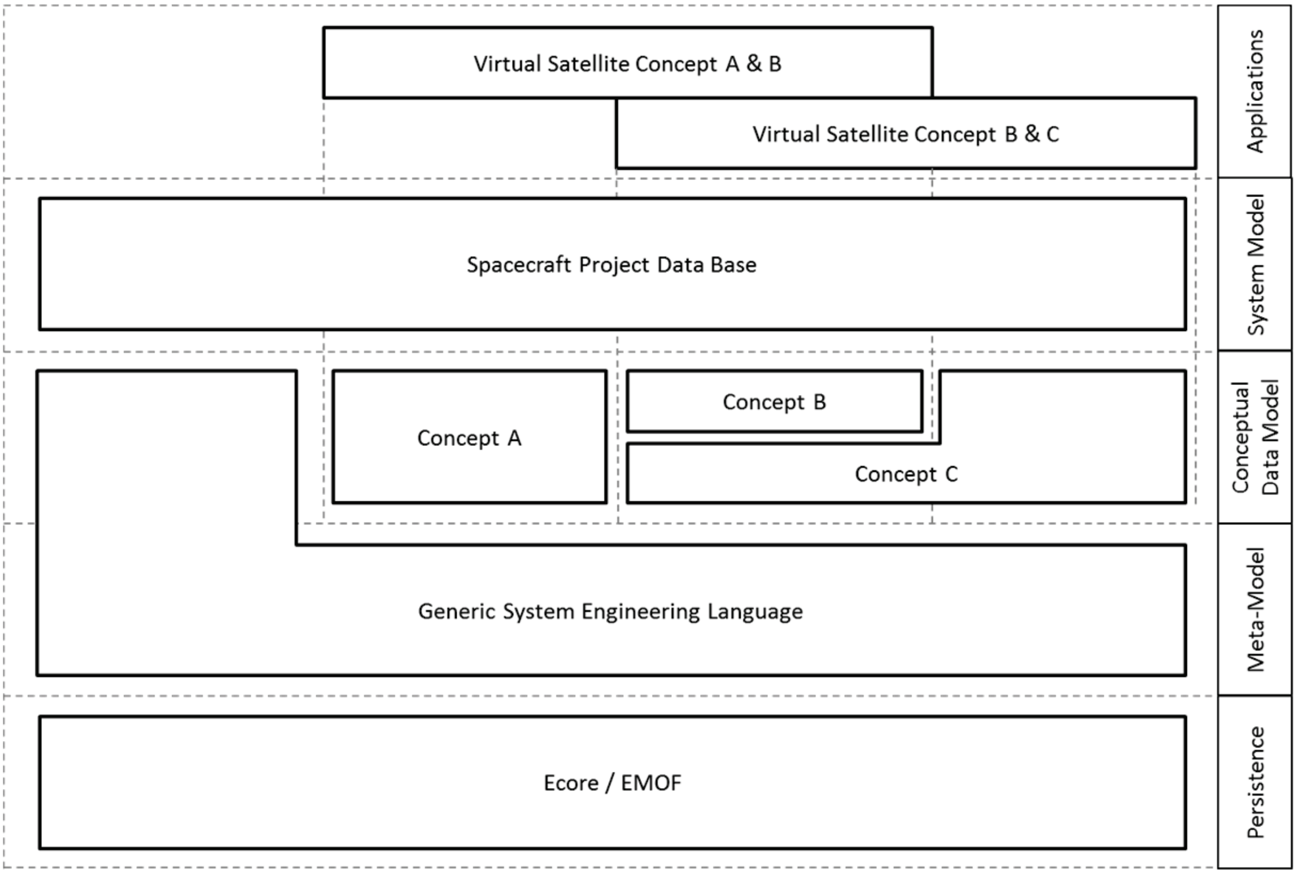

FIGURE 2 Layers of the Virtual Satellite data model using Ecore as persistency for the generic system engineering language which allows defining new concepts for the CDM that can finally be instantiated in the actual project databases. The common project database is accessed by variations of Virtual Satellite with different concepts packed to the CDM.

information provided by other processes or provide information to other processes themselves. The system model acts as a central source of truth for such system-relevant information. The various different engineering processes can connect to the system model by using specific IO methods allowing the processes to store or retrieve relevant information. With the increasing amount of engineering processes, the detail of information in the system model is increasing. Figure 1 also highlights that a lot of information is shared across the disciplines using the system model.

Back to the aforementioned software example, it is understandable that the TMTC that has once been used to develop the on-board software should be reused. As already described, such a reuse may target simulator configurations in AIT or support the configuration of the ground-segment for the operational phase. Together with the nature of multi-mission platforms, there are actually two dimensions to be considered in terms of reuse. The first dimension is that in the course of a single mission being built, the TMTC of the OBC has to match exactly the one used for the ground-segment. The second dimension is that in the frame of the satellite bus for a multi-mission scenario, the TMTC of a second mission is just a variation of the first one. This means that the TMTC for the bus will be identical, besides some mission-specific settings such as identifiers, but the TMTC for the payload can still change completely. Nevertheless, the TMTC needs to be consistent for the specific mission.

The discipline of systems engineering identifies these special and critical sets of information, which have a direct impact to various engineering disciplines. They are important artifacts and need to be identified to build corresponding databases with their underlying CDM. These CDM together with their implementations as software allow structuring the information from the engineering tasks and bring it into contextual reference to each other.

4. THE VISION OF EXTENSIBLE MBSE IN THE WHOLE LIFECYCLE OF A SPACECRAFT

All current databases and MBSE tools are focusing on specific tasks in the lifecycle of a spacecraft. This is due to the specific business and engineering needs by their respective stakeholders or contractual boundaries such as a Preliminary Design Review (PDR). For instance, most stakeholders of Phase A designs hand over their requirements for the actual spacecraft to the industry. Even though understandable, it leads to undesired situations like model-to-text-to-model transformations when transferring information, e.g., from Phase A to B. In contrast to the already mentioned databases, Virtual Satellite is targeting for a holistic approach to support the whole lifecycle of a spacecraft. It is a declared goal to bridge the transition over phases and to maintain the database from the very first idea to the end-of-life of the spacecraft.

Virtual Satellite is already the standard tool and data model for concurrent engineering studies within DLR's

CEF. As a next step, the database has to be enabled for Phases B and beyond. Therefore, specific use-cases have to be selected and implemented into the database. Since it is a nearly impossible task to anticipate the important use-cases for the whole period of a spacecraft mission, which can be several, sometimes even around 20 years, in advance, new ways of defining the data model are required. They have to allow for extensions and tailoring of the database when the specific needs actually arises. Virtual Satellite has enabled these new mechanisms to tailor the CDM when needed. As an example, Virtual Satellite can be used to capture the product structure and the functional electrical architecture quite early in the project. Later, the CDM can easily be extended for state machines and other features to capture the spacecraft's operational concept. Such an extension is called a *concept*. Even though most databases can be extended to some degrees, the way Virtual Satellite deals with it allows to also providing software packages by *concern*, which means tailored to specific engineering groups of the same project. As a further example, it is now possible to provide a Virtual Satellite software to the engineer responsible for modelling equipment ports, and it is possible to provide a second Virtual Satellite software to the engineer that connects these ports by modelling interfaces. The first software package does not understand interfaces in full depth since they are not relevant to the engineer responsible for modelling the ports by following specifications or by input from suppliers. Still, the software is capable to deal with data conflicts, e.g., in case that an engineer removes a port from a component that has already been connected by an interface by some other engineer.

Figure 2 helps to explain the previously described tailoring and extension mechanisms of the CDM. The figure shows that three concepts (A, B and C) are being packed to different variations of the Virtual Satellite but still acting on the same project database. Looking at the depicted levels of the database, the most abstract one is based on Ecore [18], which is an Eclipse-specific implementation of OMG's Essential Meta Object Facility (EMOF) [19]. That EMOF provides a standardized and interchangeable meta-modelling language and builds a foundation for other modelling languages such as the Unified Modelling Language (UML). All information stored in the database will be persisted using XML Metadata Interchange (XMI) [20]. A generic systems engineering language is defined on top of that Ecore layer. This language provides a common subset of meta-modelling capabilities needed for systems engineering and for describing future use-cases of the CDM. As an example, the language provides capabilities to describe structural features as well as data containers. These structural elements and data containers are then used and instantiated by the concepts using the language. The concepts together with the generic system engineering language construct the complete CDM. This CDM is finally used in the various different variations of Virtual Satellite. These software packages make use of the concepts in the CDM to instantiate the common system model.

Back to the example of interfaces and ports: the generic system engineering language does not know these features at all. In order to make them available in Virtual Satellite a concept would have to describe a structural element such as a component where ports can be assigned to same as describing the ports themselves. As indicated by concept B and C, they can also make use of already existing ones such as B. representing the concept for ports and C the one for interfaces which is finally referencing the ports of concept B.

As already mentioned, it is possible to specify lifecycle-specific versions of the software as well, such as a tailored variation for the initial design where the underlying database will be reused and extended using later Phase B specific variations of Virtual Satellite. Together with the capabilities to create different software packages acting on the same database, the CDM can be tailored systematic along the lifecycle of a project. This concludes that the database does not need to be static and completely defined once the project starts, but is extensible whenever needed. New concepts can be added during the course of the project and connected to each other to inherit from already existing ones. Rather than trying to anticipate all future engineering challenges, the database can evolve in an iterative approach. New IO interfaces to the engineering processes can be introduced together with a concept and therefore with new variations of the software. This ensures that IO capabilities match the actual concepts and resulting CDM as they are defined. It also helps to simplify access to the databases since engineers have to deal with their IO functionalities only.

All these aspects allow implementing the database as MBSE tool into new projects, still not knowing all specific needs. Introducing MBSE in such an iterative way seems promising to reduce reaction times in the later processes. In particular, implementing MBSE in the later phases is also considered to then give direct feedback from operations to the early development of following new missions. All this will reduce change requests after design reviews since such MBSE databases allow early verification. As first application of this Virtual Satellite concept-based MBSE approach, we will incrementally implement the vision in the S2TEP project.

5. IMPLEMENTATION OF VIRTUAL SATELLITE AND MBSE IN THE S2TEP PROJECT

The outlined vision of Virtual Satellite as an MBSE tool over the whole lifecycle is embedded in the S2TEP project. The demand of a suitable MBSE database is high, due to the nature of S2TEP's multi-mission architecture. The overall characteristic of such a project is that the satellite bus is currently developed to the requirements of a reference payload. In the future this payload will be replaced by various different payloads, which in consequence lead to various different missions. It is planned to evolve the satellite bus and replace certain commercial-off-the-shelf (COTS) components with in-house developments over time. This will enable research in respect to the satellite bus, but raises a complex configuration control problem as well. The database is required to not just adequately manage information, such as interfaces across all bus components the same way as the interfaces to the payload, but it has to also support this use-case over all variations and combinations of the actual missions.

Figure 3 elaborates on the described configuration control issues concerning the system model of the S2TEP satellite bus and the resulting missions. On the upper part of the diagram, the system model contains some initial information of the satellite bus. This system model is used for the first mission A. It is then extended by the first

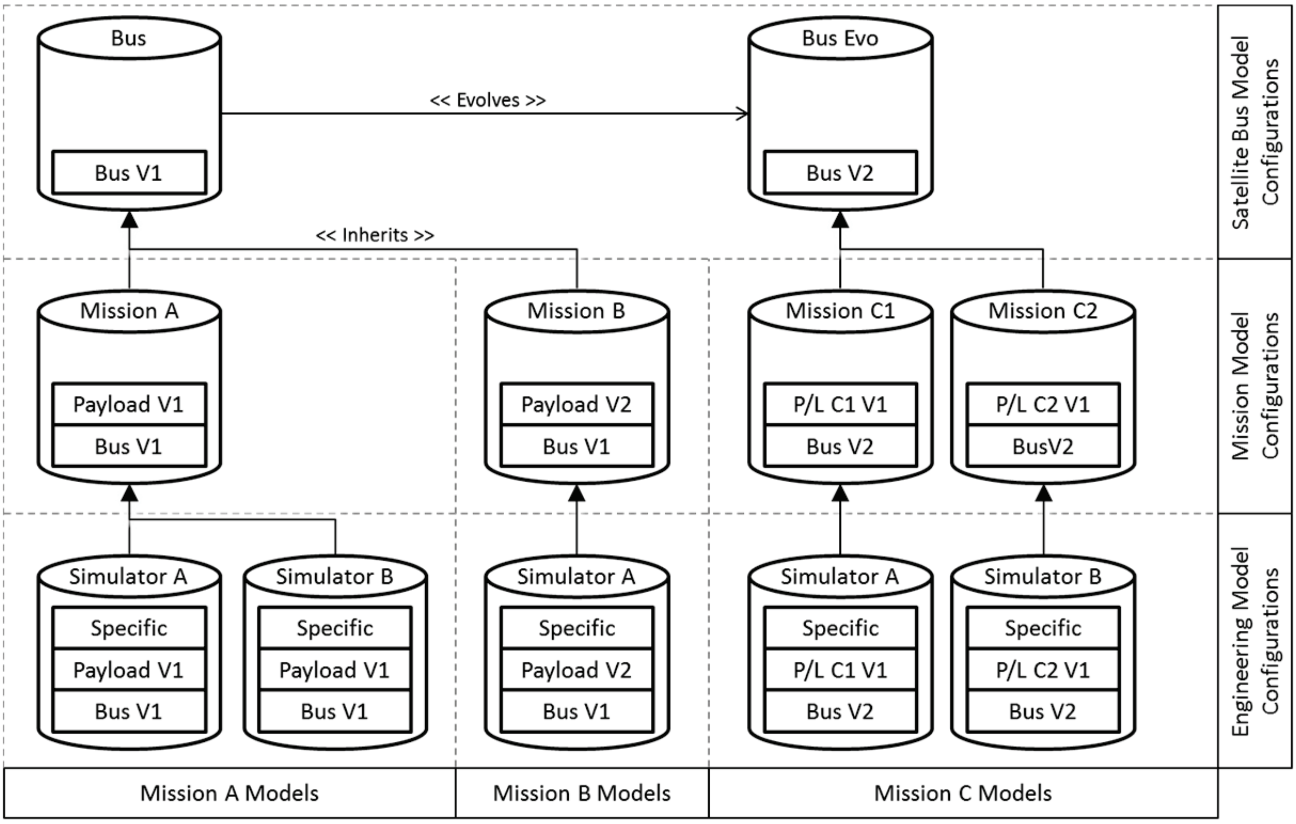

FIGURE 3 Configuration control scenarios as expected in S2TEP with various configurations, missions and application-specific models on a yet evolving satellite platform.

version of the payload, visible in the middle layer. Based on this mission-specific model, two engineering models are derived, depicted in the lowest layer. They are needed to reflect specific adaptions for specific tasks. E.g., simulator A is running Hardware-in-the-Loop (HIL) simulations with just some dedicated equipment whereas simulator B is running the environment as software simulation with the real satellite in the loop.

Such specific tasks may imply different requirements on components such as the harness or breakout boxes. Similar considerations apply for the case that an evolution of the payload is about to be flown. The content of the system model regarding the bus may stay the same, but slight variations will definitely apply to the payload and maybe small changes to bus components. All these aspects apply for mission C as well, where now the satellite bus has evolved. Mission C actually depicts a fictive tandem mission with two satellites carrying a combined payload. Within such a scenario, TMTC has to be specific such as the satellites must be commanded individually even though both satellite buses are basically the same. These configuration control mechanisms are not an exclusive feature of Virtual Satellite but are strongly required by the S2TEP project. In the context of MBSE in the whole lifecycle of S2TEP, further configurations need to be considered when going back to Phase A studies defining new payloads after already successful precursor missions have been launched.

Besides these configuration control mechanisms, which are integrated to Virtual Satellite, some specific use-cases are depicted from the S2TEP project. It is intended to implement them along the way and to reflect reasonable and in terms of MBSE representative applications within the various phases of the project. Later in the S2TEP project, it is intended to extend the MBSE database with further functionality and interfaces to the processes where applicable.

Figure 4 shows some of the currently selected use-cases for the S2TEP project. They start with the usage of Virtual Satellite for the early CEF studies. The data gathered are the first artefacts in the database. Currently it is assumed that the Phase A product structures can directly outline the initial Phase B product structures. Key indicators such as masses, dimensions and power consumptions may be used for automated design checks, e.g., the user could be notified if a payload-equipment is violating the reference-payload envelope.

For Phase B, it is intended to enrich the system model with interface information. Even though starting on the level of the functional electrical architecture, it is already considered to investigate interfaces of thermal and mechanical nature as well. Nevertheless, the interface information of the functional electrical architecture stored in the database will be reused during the Phases C and D for Interface Control Document (ICD) generation. Additionally, it will be investigated in which respect source code for the onboard computer (OBC) might be generated. It is also intended to investigate the integration of TMTC to the database for AIT and operations. First discussions showed the importance of not to duplicating databases that are already used in the German Space Operation Center (GSOC). However, it will be inevitable to combine them where applicable to make use of correct TMTC during AIT and to provide individual TMTC specifics of the various missions directly into the control center.

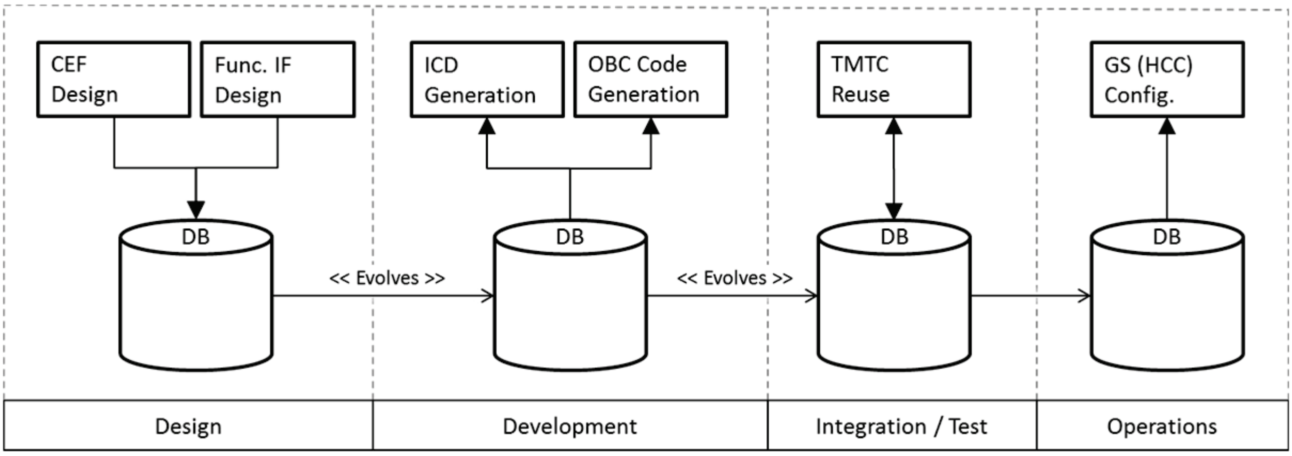

FIGURE 4 Overview to the MBSE implementation in the S2TEP project. Beginning from the CEF studies, Virtual Satellite will be used to accompany the development through the phases until the spacecraft is build and ready to operate.

With the introduction of the Hosted Control Center (HCC) at GSOC, customers will have the opportunity to directly access and control their mission. A quick instantiation of such a HCC with standardized TMTC for the bus will enhance feedback from operations into the early design when integrating the customer's payload.

By today, Virtual Satellite is ready to provide the here described generic system engineering language to define new concepts for S2TEP as well as the intrinsic configuration control capabilities. The first Phase A studies of S2TEP have already been performed using Virtual Satellite within the CEF. Even though these first analysis have been done using the old platform, current ongoing work to integrate the CEF use-case as a concept for the evolved Virtual Satellite platform will enable reuse of that data. The concept for dealing with the functional electrical architecture has been defined as well and is shipped to the project members together with corresponding IO functionality. Once this software version is well integrated into the development processes, the next concepts will be integrated.

This iterative approach provides inside to various different aspects of MBSE. Rather than introducing MBSE as a whole and force everyone to change the way they work, it will combine the existing processes where necessary in a non-intrusive way. Today, this is already reflected in several discussions identifying the need of advanced configuration management for the harness design. The idea is to integrate a new harness concept by reusing the already existing functional electrical architecture. Exactly these discussions present the future use-cases that have to be addressed with the following S2TEP missions.

6. SUMMARY AND OUTLOOK

Supporting the development of a spacecraft over the whole lifecycle needs the support of MBSE. The basic idea shows that it is necessary to gather system relevant information into a common database. Such a database helps to keep the data consistent. Additionally, respective IO functionality is needed to communicate between the database as well as discipline-specific engineering and business processes.

Different from other MBSE databases in the European space community, Virtual Satellite tries to address the whole lifecycle of a spacecraft. The DLR is in a unique position of planning, designing, constructing and finally operating space missions. Therefore, use-cases for Virtual Satellite can directly be derived from these projects. Since it is almost impossible to foresee all relevant use-cases for such a database, Virtual Satellite introduces a new data model based on a generic systems engineering language. This new language enables defining certain concepts consisting of data model elements and corresponding IO capabilities. These concepts can be bundled into various different Virtual Satellite variations, all with their specific CDM consisting of these concepts but still operating on the same system model. Finally, this functionality allows implementing Virtual Satellite iteratively and adjusting the MBSE process to the needs as they arrive rather than trying to address the perfect solution straight away.

S2TEP is one of the first projects where Virtual Satellite is introduced as MBSE tool for the whole lifecycle of a spacecraft. S2TEP is actually not just one spacecraft but a whole family of missions based on the S2TEP satellite bus. The configuration control problems that arise with such a multi-mission setup are inevitably requiring adequate tool support like MBSE. Within the S2TEP project, Virtual Satellite is already used to gather the first system model during the concurrent engineering study based on the old platform, but there is already some ongoing work to integrate this information as a concept for the evolved platform. Aspects of later phases such as the functional electrical architecture are already defined in a concept and shipped to the S2TEP project. On top of that, it is planned to investigate the impact of modeled TMTC in AIT and reusing it in operations to improve design feedback from GSOC back into the early design.

From current discussions, it becomes already apparent that there are many more use-cases to be addressed in the future. Some will be identified when the S2TEP satellite bus is evolving, some others, such as harness management, is already encountered together with complex configuration control problems. MBSE promises answers in these areas and will drive the research for Virtual Satellite in the upcoming years.

7. REFERENCES

[1] ECSS Secretariat ESA Estec, "ECSS-M-ST-10C Space project management - Project planning and implementation," European Space Agency (ESA), Noordwijk / Netherlands, 2009.

[2] NASA, "Systems Engineering Handbook (NASA/SP-2007-6105 Rev 1)," National Aeronautics and Space Administration, Washington, D.C. / USA, 2007.

[3] F. Dannemann and M. Jetzschmann, "Technology-Driven Design of a Scalable Small Satellite Platform," in *Small Satellites, Systems & Services (4S) Symposium*, Valetta, Malta, 2016.

[4] European Space Agency ESA, "OCDT Community Portal," [Online]. Available: https://ocdt.esa.int/. [Accessed 20 04 2016].

[5] ECSS Secretariat ESA Estec, "ECSS-E-TM-10-25A - Space Engineering - Engineering design model data exchange (CDF)," Noordwijk/Netherlands, 2011.

[6] M. Deshmukh, R. Wolff, P. M. Fischer, M. Flatken and A. Gerndt, "Interactive 3D Visualization to Support Concurrent Engineering in the Early Space Mission Design Phase," in *5th CEAS Air and Space Conference*, Delft, Netherlands, 2015.

[7] V. Schaus, M. Tiede, P. M. Fischer, D. Lüdtke and A. Gerndt, "A Continuous Verification Process in Concurrent Engineering," in *American Institute of Aeronautics and Astronautics*, San Diego, CA, 2013.

[8] ECSS Secretariat ESA Estec, "ECSS-E-TM-10-23A - Space engineering - Space System Data Repository," Noordwijk/Netherlands, 2011.

[9] European Space Agency ESA, "Virtual Spacecraft Design," [Online]. Available: http://www.vsd-project.org/. [Accessed 20 4 2016].

[10] European Space Agency ESA, "EGS-CC - European Ground Segment Common Core," 2013. [Online]. Available: http://www.egscc.esa.int/. [Accessed 24 02 2016].

[11] M. Pecchioli, A. Walsh, J. M. Carranza, R. Blommestijn, M.-C. Charmeau, M. Geyer, C. Stangl, P. Parmentier, H. Eisenmann, J. Rueting, P. Athmann, W. Bothmer, I. Krakowski, P.-Y. Schmerber, F. Chatte, P. Chiroli and M. Poletti, "Objectives and Concepts of the European Ground Systems Common Core (EGS-CC)," in *Simulation & EGSE for Space Programmes*, Noordiwjk / Netherlands, 2012.

[12] OMG, "SysML 1.2 RTF - ANNEX C.5 - Quantities, Units, Dimensions, Values (QUDV)," 2009.

[13] V. Schaus, P. M. Fischer and A. Gerndt, "Taking Advantage of the Model: Application of the Quantity, Units, Dimension, and Values Standard in Concurrent Spacecraft Engineering," in *Proceedings of 23nd Annual International Symposium of the International Council of Systems Engineering*, Philadelphia, USA, 2013.

[14] M. Deshmukh, V. Schaus, P. Fischer, D. Quantius, V. Maiwald and A. Gerndt, "Decision Support Tool for Concurrent Engineering in Space Mission Design," in *ISPE International Conference on Concurrent Engineering*, Trier, Germany, 2012.

[15] V. Schaus, J. Müller, P. M. Fischer, R. Wolff and A. Gerndt, "Collaborative Satellite Configuration Supporting CATIA Export," in *SECESA*, Lisbon, Portugal, 2012.

[16] ScopeSet - The Tools Experts, "Mapping and importing Catia Properties," [Online]. Available: http://www.vsd-project.org/download/ssdevideos/CatiaImport.swf. [Accessed 06 04 2016].

[17] P. M. Fischer, H. Eisenmann and J. Fuchs, "Functional Verification by Simulation based on Preliminary System Design Data," in *6th International Conference on Systems & Concurrent Engineering for Space Applications (SECESA)*, Stuttgart / Germany, 2014.

[18] D. Steinberg, F. Budinsky, M. Paternostro and E. Merks, EMF Eclipse Modeling Framework, Boston, USA : Pearson Education, Inc, 2009.

[19] OMG, "Meta Object Facility (MOF) Core Specification - Version 2.5," 2015.

[20] OMG, "XML Metadata Interchange (XMI) Specification - Version2.5.1," 2015.