Model Based Engineering Hub – A firm foundation for a new generation of MBSE exchange

3rd Model Based Space Systems and Software Engineering workshop
- MBSE 2022 -

22.-24. November 2022

Tobias Hoppe(1), Todor Stoitsev(2), Colin Borrett(1)

(1)Airbus Defence and Space GmbH
Claude-Dornier-Straße, D-88090 Immenstaad, Germany
Email:{tobias.hoppe, colin.borrett}@airbus.com

(2)SpaceCube GmbH
Berliner Allee 47, D-64295 Darmstadt, Germany
Email: t.stoitsev@space-cube.de

[O-4, O-5] The Model-Based Engineering Hub (MBEH) is an Airbus led study to develop a prototype MBSE SaaS application for enabling easy information and knowledge exchange between stakeholders along the space system engineering lifecycle, including between disciplines and Customer/Supplier boundaries. Representing an evolution of the software originally developed in the ESEP study, the MBEH aims at enabling engineering data exchange in line with the semantics of the Space Systems Ontology (SSO), with a focus on extending the Operations and RAMS Universe of Discourse, and a set of demonstration scenarios designed to clearly showcase MBSE in action.

Introduction

The benefits from MBSE are well known and the Large System Integrators (LSIs) have long researched and industrialized MBSE based applications. These solutions have demonstrated their benefits in the S/C development, verification and operations; however it is impossible for the LSIs to impose their own tools or environments to suppliers, subcontractors or customers: They also need to interact with other entities with their own processes, data format, etc. ESA's Model Based System Engineering Factory Roadmap [1] proposes a strategy for achieving a harmonization and the rationalization of the use of model based system engineering within the space system engineering community with the MBEH as core of the MBSE Factory.

An envisaged answer to this knowledge exchange challenge is seen in an ontology developed using the Object Role Modelling (ORM) [2] approach. Implementing a prototype of a fully interconnected environment supporting an ORM Conceptual Data Model is hence of vital interest to extend the existing MBSE environment consequently towards a fully digitalized and industrialized S/C design, development, verification and operational solution. Another challenge however is seen with respect to maintaining compatibility with the already industrialized solutions. A clear focus thereby needs to be maintained on the long-term maintenance of the MBEH infrastructure and interoperability with a variety of Domain-Specific Tools (DSTs) using open data formats.

In the frame of the MBEH study [3] the analysis of data exchanges had a detailed focus on the operations as well as Reliability, Availability, Maintainability and Safety (RAMS) domains for supporting semantic engineering data exchanges based on conceptual models designed in ORM which will be feed back into the SSO [4]. The focus will be on supporting real-life use cases, stemming from concrete data exchange scenarios. While we are open to consider a more generic architecture in some areas, having real-life references is considered of paramount importance for the MBEH adoption to focus on the most relevant parts first.

Operations Use Cases

The operations engineering domain is widely ramified with many interfaces to other engineering domains as well as to suppliers and customers. For the development of an operations-specific universe of discourse of the SSO it was essential to identify all of these exchanges and describe the exchanged data – especially regarding the semantics of the data being exchanged.

Monitoring and control (M&C) data is received by a prime from several suppliers on equipment level. After aggregation on system level, M&C data is shared internally with several engineering domains and used e.g. for checkout system configuration during validation testing. In addition, it is shared with the customer e.g. with ESOC to facilitate ground monitoring when the spacecraft has been launched. The exchanges with customers also cover On-Board Control Procedures (OBCPs), Flight Operation Procedures (FOPs), Rules and Constraints, Verification Test Reports, Instrument Data Return Models, Damage Avoidance and Safety Sheet, and further related data.

RAMS Use Cases

Analysing system safety and dependability is a fundamental step that is performed in parallel to system design and verification of functional correctness. That is why, the analysed RAMS use cases are focused on Failure Modes and Effects Analysis (FMEA) on several levels, feared events list, and reliability prediction. Hereby, the exchanged data between engineering domains and RAMS as well as the RAMS-specific tooling analysis was a key factor to identify the exchanged data aspects and their semantics. This was the baseline to develop a RAMS universe of discourse for the SSO which is progressing while validating parts of it with the scenario presented in the following section.

Selected Scenarios

In the frame of the MBEH study specific use cases from operations and RAMS have been selected to demonstrate the MBEH in action.

The demonstration case for the OPS-ENG exchange will focus on a FOP exchange scenario between a prime and ESOC. This will be realized in a two steps approach to showcase the stepwise integration of the MBEH into an existing tool landscape.

Figure 1 presents the basic demonstration case where TM/TC data in MIB-format is available outside the MBEH, i.e. created by an authoring tool not connected yet with the MBEH. The TM/TC data is imported into MOIS to create the FOPs and they are transformed into a MBEH compatible format and uploaded into the MBEH, i.e. they are available in a format defined by the SSO. The FOPs and TM/TC data is than imported into OPEN-M by taking the FOPs from the MBEH.

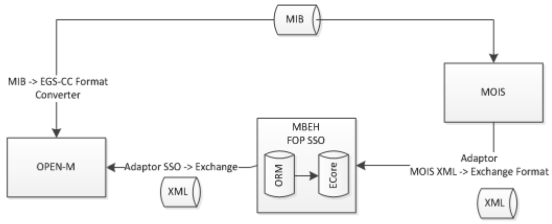

Figure 1 Operations demo case with TM/TC data from outside the MBEH

The second step of the demonstration scenario realization covers the integration of a TM/TC data import into the MBEH as illustrated in Figure 2.

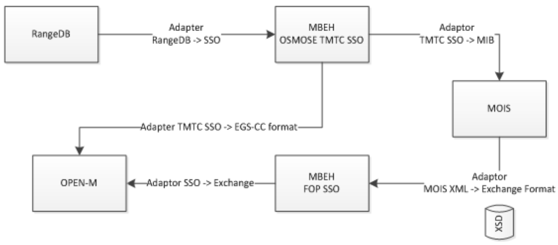

Figure 2 Extended Operations demo case with TM/TC data shared via the MBEH

The RAMS demonstration use case focuses on the FMEA generation based on specific workflows supported through dedicated Micro Frontend Components (MFCs) where necessary. The FMEA generation requires an import of a product structure into the MBEH as well as required failure catalogues that are shared between projects. The augmentation of the product tree with failure catalogue data is realized by workflows that run directly on the MBEH infrastructure. In case of inconsistencies between failure catalogues and product tree the user is informed and can correct the product tree data by either uploading an updated version or performing modifications directly in the MBEH web frontend using build-in editors.

MBEH Architecture Overview

The MBEH architecture is mainly driven by the analyzed exchange scenarios and the concepts, lesson-learnt, and findings identified in related work – especially in the frame of the ESEP [5] and GSEF [6] activities. As such the MBEH architecture is based on more than 10 years of experiences in building and applying MBSE frameworks using a state-of-the-art technology stack.

The generic architecture can manage multiple Eclipse Modeling Framework (EMF) Ecore [7] compatible data model versions and associated models and versions thereof in parallel. EMF Ecore is generally seen as a reference implementation of the Essential Meta Object Facility (EMOF) [8] – the highest abstraction level metamodel in the metamodels devised by the Object Management Group (OMG) Model Driven Architecture (MDA). Standard modelling languages such as the Unified Modeling Language (UML) and languages built on top such as the Systems Modeling Language (SysML) are all based on EMOF. Many of the associated MBSE frameworks and tools use EMF either directly (e.g. Capella, GSEF, OPEN/MODE, VSEE, ESA-AF exploitation environment, OCDT for the integration with ESA-AF and VSEE) or an EMOF or UML based representation with a different data storage implementation such as relational database or proprietary file formats.

Figure 3 outlines the relations between the DST data models and the MBEH internal data model. Each DST is connected either directly using a generic REST end point that matches the SSO semantics or it can be integrated via a dedicated interface that can be plugged into the existing architecture. This is then supported on frontend level through an associated MFC. For engineering data that is provided in CSV format an importer including frontend components is available in the MBEH. The current features support the definition of the required mapping to the SSO concepts reflected in the internal MBEH data structure.

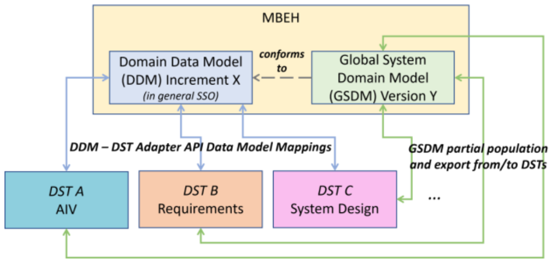

Figure 3 Composition of domain data into Global Domain Model Population in MBEH

Apart from data exchange the MBEH also supports the systems engineering processes by providing:

- configuration control for data being uploaded
- data sharing and reuse by splitting data into projects and data sets as needed – even to fulfil the need to know principle
- consistency check execution and reporting for SSO compatibility as well as user defined data constraints
- workflow integration to reflect business processes in the MBEH with tasks being implemented as MFCs to handle stakeholder specific contexts
- data querying interface to perform data and meta-data analyses based on custom dashboards
- collaboration features to foster discussions and exchanges about critical design elements without the need of an additional tool

Challenges and Roadmap

One general challenge is the "openness" of DSTs, i.e. the kind of external interfaces they provide. Different integration strategies need to be considered depending on this. For example, OSLC does not seem to be supported very well, and if at all, through additional, commercial modules that need to be procured separately from the main product. In conjunction with having several different DSTs to be supported along the life-cycle the maintainability of all these DST-specific interfaces – especially of outdated once required for long-term maintainability – will become a major challenge.

A further challenge is the adjustment on upcoming ways of working with the MBEH. After roll-out and introduction into stakeholders processes the MBEH will evolve further driven by the natural way of improving things.

The current roadmap foresees implementation of a first version of the MBEH until the end of 2022 with implementation of selected scenarios and finalization of prototype implementation until mid-2023.

Conclusion

The implementation of commonly shared conceptual models from the SSO within the MBEH builds the foundation for semantically strong model-based data and knowledge exchange between all stakeholders involved in space systems engineering. It paves the way for digital continuity with pure digital end-to-end workflows covering the whole space system engineering life-cycle from early design, over development, and operations until decommissioning.

References

[1] ESA-TECS-TN-011986 Requirements for a Model Based System Engineering Factory
[2] Technics Publications 2015 Object-Role Modelling Fundamentals – A Practical Guide to Data Modelling with ORM
[3] ESA-TRP-TECSYE-SOW-014909 Issue 2.1 Model-Based Engineering Hub: Statement of Work
[4] ESA-TECSWM-SOW-015604 Space System Ontology Development, Statement of Work
[5] ESEP.SSDD End-to-End System Engineering Portal (ESEP): Software Specification and Design Document
[6] ESA-GRST-STU-DD-0001 Ground Segment Engineering Framework - Software Specification and Design Document
[7] STEINBERG, Dave, et al. EMF: eclipse modeling framework. Pearson Education.
[8] formal/2019-10-01 OMG Meta Object Facility (MOF) Core Specification