Published in
J.P.A. Almeida, U. Frank and T. Kühne, “Multi-Level Modelling (Dagstuhl Seminar 17492”, Schloss Dagstuhl--Leibniz-Zentrum für Informatik, Dagstuhl Reports 2018, Volume 7, Number 12, pp 24-27, DOI: 10.4230/DagRep.7.12.18,
http://drops.dagstuhl.de/opus/volltexte/2018/8675

Deutsches Zentrum für Luft- und Raumfahrt

Potential of Multi Level Modelling in Model Based Systems Engineering

A “National Research Lab” Perspective

Philipp M. Fischer, DLR – Software for Space Systems and interactive Visualization, philipp.fischer@dlr.de

Abstract

The lifecycle of a spacecraft follows a process of phases. There are important goals between these phases such as a preliminary design review (PDR). These goals have to be successfully passed by the whole project team. Usually they are connected to contractual conditions, e.g. after the PDR the design is usually settled and the spacecraft will be built. Such contracts consider agreements with industry and suppliers on part and equipment orders, etc. [1] [2] Changing the design after the PDR may require contractual changes. These changes tend to be expensive and have to be avoided. [3]

To avoid the aforementioned issues, model based systems engineering (MBSE) has been introduced in satellite design. It is focusing on data bases that provide a conceptual data model (CDM also known as meta model). The engineers start modeling the system within such data bases usually on a functional level. The system model is then used as central source of knowledge for further processes. [3] Such processes may cover on the fly analysis [4], configuration of simulators [5], new ways of modelling including interactive visualization [6] and emerging trends such as mixed reality. This notion of an MBSE approach has been implemented e.g. in DLR’s data base called Virtual Satellite. Until today it is successfully applied in early spacecraft design. [7] Nowadays, the data base is applied beyond the early phases. Therefore it has to deal with more detailed information and has to cope with yet unknown requirements of tomorrow. Accordingly the CDM has become more complex and offers extension mechanisms. [3]

The success of these data bases is partly founded in their configuration control capabilities. In fact, the engineers require not just one model of the spacecraft but several. These models are a master model, derived simulator models, or models for the actual satellites one and two. The different models are needed to reflect differences e.g. a different electrical harness for the simulator, or individual command ids for individual satellites. [3] [5] This is handled by the product structures of the data bases. They are standardized to a certain extend in the activities of European Ground Segment – Common Core (EGS-CC) as well as the European Cooperation for Space Standardization Technical Memorandum (ECSS-E-TM) 10-23. [8] [9] These structures are used for a hierarchical decomposition of the system. The first tree modelled, is usually a product tree. The engineers are using this tree for an initial description of the required parts such as a reaction wheel (RW) or magnetic-torquer (MTQ). They don’t yet model every instance of such a part. Instead, they are modelling them as an idea of a type. The second tree is the configuration tree where these types are instantiated into a virtual configuration of several RWs and MTQs. The final trees are the assembly trees, which are based on the configuration trees but representing how actual satellites are build. They reflect the assembly e.g. of an actual satellite number one and two. Since all trees, including their parts, are based on each other, defining the mass of the RW once in the product tree will update all its further instances in the configuration and assemblies as well. Override functionality allows changing the values when needed or to combine the information with so called realizations. These realizations represent the actual ordered parts that have been delivered. E.g. their calibrations are measured and the best fitting parts are now assigned to the assembly. The information such as a mass is modelled by so called engineering categories. They are based on what is known as type/object pattern. [10] [11] An engineering category defines a property such as a mass and assigning this category to an element in the product tree instantiates it. Now information can be stored in the instance of this property. [3]

Even though successful, there are some drawbacks, where Multi-Level Modeling promises some reasonable improvements. For example, an engineering category for storing geometric information of a part consists of a position, a size and a shape. These three properties make sense at configuration level, but not yet at product level. At product level or type level the position is simply not yet known. By today this is handled by either providing arbitrary values or by complex class inheritance hierarchies. Nevertheless such handling is a workaround rather than a proper solution to such problems.

Multi-Level Modelling and the idea of potencies and deep instantiation [12] in particular, seems to offer a solution. Assigning a potency of two for the position property creates awareness of this property already at product level. The actual value of that property can now be set starting from configuration level. Considering another example based on an engineering category for tele-commands, it consists of a purpose, e.g. RW turn on, an equipment identifier as well as a satellite identifier. With the concept of potency and deep

Published in
J.P.A. Almeida, U. Frank and T. Kühne, "Multi-Level Modelling (Dagstuhl Seminar 17492", Schloss Dagstuhl--Leibniz-Zentrum für Informatik, Dagstuhl Reports 2018, Volume 7, Number 12, pp 24-27, DOI: 10.4230/DagRep.7.12.18,
http://drops.dagstuhl.de/opus/volltexte/2018/8675

instantiation, engineers are aware of all three properties already at product level. The actual necessary information needs to be provided at the stages of configuration and assembly.

This approach works well with the accepted set of product structures. Nevertheless current work indicates that there might be further trees needed in future applications. Introducing a new integration tree in between configuration and assembly breaks the potency mechanism for the tele-command example. A decrease of that potency with every level of instantiation is not suitable. A fix to this issue could lead in the direction of context aware potencies.

At the moment, the MBSE data bases do not yet apply such Multi-Level Modelling. Nevertheless, it can be seen that certain directions of Multi-Level Modelling could improve the overall modeling activities. For sure this view is a highly practical driven adoption of the theories of Multi-Level Modelling and potentially breaks some of the clear cut semantics. Still it shows that these theories provide some answers to the problems of spacecraft related models of today. In general, the described idea of applying potencies and deep instantiation to the concept of engineering categories looks promising. In order to prove its applicability, further research is needed and some first prototypes are envisaged for evaluation.

References

[1] ECSS Secretariat, „ECSS-M-ST-10C Space project management - Project planning and implementation," ESA-ESTEC Requirements & Standards Division, 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] P. M. Fischer, D. Lüdtke, C. Lange, F.-C. Roshani, F. Dannemann und A. Gerndt, „Implementing model-based system engineering for the whole lifecycle of a spacecraft," *CEAS Space Journal,* 12 July 2017.

[4] M. Deshmukh, V. Schaus, P. Fischer, D. Quantius, V. Maiwald und A. Gerndt, „Decision Support Tool for Concurrent Engineering in Space Mission Design," in *Concurrent Engineering Approaches for Sustainable Product Development in a Multi-Disciplinary Environment: Proceedings of the 19th ISPE International Conference on Concurrent Engineering*, London, Great Britain, Springer London, 2013, pp. 497-508.

[5] P. M. Fischer, H. Eisenmann und 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.

[6] M. Deshmukh, R. Wolff, P. M. Fischer, M. Flatken und 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] P. M. Fischer, M. Deshmukh, V. Maiwald, D. Quantius, A. Martelo Gomez und A. Gerndt, „Conceptual Data Model - A Foundation for Successful Concurrent Engineering," *Concurrent Engineering: Research and Applications,* 2017.

[8] 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 und M. Poletti, „Objectives and Concepts of the European Ground Systems Common Core (EGS-CC)," in *Simulation & EGSE for Space Programmes*, Noordiwjk, Netherlands, 2012.

[9] ECSS Secretariat, „ECSS-E-TM-10-23A - Space engineering - Space system data repository," ESA-ESTEC Requirements & Standards Devision, Noordwijk, Netherlands, 2011.

[10] F. D. Lyardet, „The Dynamic Template Pattern," in *The 4th Pattern Languages of Programming Conference*, Monticello, Illinois, USA, 1997.

[11] R. Johnson und B. Woolf, „Type object," in *Pattern languages of program design 3*, Bosten, MA, USA, Addison-Wesley Longman Publishing Co., Inc., 1997, pp. 47-65.

[12] C. Atkinson und T. Kühne, „Reducing accidential complexity in domain models," *Software & Systems Modeling,* Bd. 7, Nr. 3, pp. 345-359, July 2008.