Embarking to end-to-end system modelling for Galileo Second Generation

Authors:

Catherine Morlet, Alberto Gonzalez Fernandez, Riccardo Dellago, Steven Bouchired, Gustavo Lopez Risueno, Miguel Manteiga Bautista (ESA/ESTEC)

Carmela Ruta, Riccardo Dall'Ora (Thales Alenia Space Italy, Rome)

Thomas Bey (Airbus Defence and Space, Munich)

Mounir Chattou (Thales SIX GTS France)

Objectives : O-1, O-3, O-5 and T-1

Extended Abstract:

This paper presents the Model-Based System Engineering (MBSE) approach followed for Galileo Second Generation. We discuss the choices made and methodology applied in Phase B1 and B2 of the system design activity during the last 4 years, the generation of the design documentation from the model and the future evolutions expected on the MBSE approach. The paper also discusses our (upcoming) use cases for expanding usage of MBSE in Galileo and future areas of work.

1. Galileo Second Generation (G2G) in context

The Galileo program is Europe's initiative for a state-of-the-art global satellite navigation system; it started on the 1990s with the first experimental satellite launched in 2005 and the system is providing operations since 2016 with currently 28 satellites in orbit (from the In-Orbit Validation phase and the first Generation of satellites). Studies for Galileo Second Generation (G2) as an evolution of the First Generation (G1) started about 10 years ago while the G1 IOV phase was finishing and deployment of ground and space infrastructure for G1 was on-going.

While the first Generation of Galileo is in service since 2016, the system evolution began through a set of initial studies in 2013 and the Galileo Second Generation Phase B is reaching completion. At the same time, procurement of the first batch of second generation satellites is on-going and several ground segment Phase B contracts are running. Investigation about using a modelling tool started during the Phase B1 of the system activity. Galileo is a very complex system, consisting of a MEO constellation, a worldwide distributed core ground segment, many interfaces with dedicated Galileo service facilities and external providers or stakeholders, a significant security component, and the delivery of the mission relies on a closed loop where the data broadcasted are used to generate the next set of data for broadcast. Based on those considerations, it appeared that moving towards a modern model-based methodology for Galileo evolution was a must to ensure the consistency, the completeness of the design especially for functional aspects, interfaces between segments and interactions with system operator. System modelling throughout the development and exploitation phase will be a valid support for keeping under control all system versions to be operated.

The design authority for G2G is ESA as per Financial Framework Partnership Agreement (FFPA) with the European Commission. ESA is also the System Prime for the System in Development. ESA is

supported by and implements contracts with industry for all activities (system and segments). Our system engineering activities comprise all design aspects covering the generation and delivery of the navigation products for all services, the contribution to services such as the search and rescue, the system management and control of the ground and space segments, the maintenance of the Galileo constellation (including launch and any orbital manoeuvre), cyber-security monitoring, etc. The design team consists of engineers from different industry organisations and ESA who are located in different physical locations. Thus, a server-based implementation with remote access and collaborative capabilities to develop concurrently the same model was identified as a must.

2. Development organisation and achievements

An MBSE approach had been experimented in G1 until 2009 but had been abandoned as too complex to maintain. The model maintainability complexity was due to a trend to over-specify by going deep inside the procured component black boxes and the inadequate tooling available at that time (homemade set of scripts around a data-base).

Later on, still in support to G1, Enterprise Architect (EA) has been used in to support activities related to accreditation where specific questions are raised with respect to the security of the system before approving its operational usage. However, the development made was more bottom-up trying to represent with the model the reality of the implemented system (thus more a kind of physical model).

For G2G, we started with the idea of developing a model in 2018, thus tools on the market were less advanced than they are today. Among the proposals received, Team for Capella (T4C) was judged as the most promising at the time to fulfil our expectations and deploy a collaborative infrastructure, most suitable to the top-down systems engineering decomposition work process, with more automated features to ensure design consistency, more intuitive and simpler to use. The fact that a significant amount of information on the tool and Capella standalone installation was free of charge as well as opportunities to learn about the tool as team of developers were also key assets.

Our infrastructure installation allows for remote access to a virtual machine in a server dedicated to the MBSE; this allowed us to continue the development without interruption during all the Covid-19 constraints in the various countries where the developers were.

The core development team received a set of joint training to acquire an equivalent knowledge on tool capabilities and development methodology. The first attempts for model development were weekly reviewed by the team in joint meetings where modifications of the approach were agreed. It was in particular decided to start the development from the system analysis and skip the operational analysis available in the tool, since this high-level analysis is given to the team as an input. We also got a verification of our model development by the T4C trainer once we had finalised a first complete system analysis and before transiting to the logical analysis as per Arcadia methodology. This initial verification focused on the level of detail, consistency of the design achieved and readiness of the system analysis before doing the transition to the logical level analysis. The logical analysis development continued in view of the System Requirements Review (SRR) but turned out to be too detailed, stepping sometimes into the suppliers solution space. It was also decided to extend the system perimeter to include a number of service provision facilities, thus our model had to be reconsidered in full since more interfaces and functions were anyway to be included.

It was therefore decided to rebuild completely the model after the SRR, still focusing on the system analysis and the logical analysis provided by the environment. This new phase of model development

is now concluded and sets the ground of the design proposed at the Preliminary Design Review (PDR). The system analysis has been simplified compared to our initial model, a consistent development in terms of views/diagrams for each system function has been achieved, a colouring pattern has been considered for each component of the design (segment or service facility) in the logical analysis, and the description of system and logical functions as well as all functional scenarios is now included inside the model. The MBSE approach has allowed to verify the coherency and completeness of the system design, as well to facilitate the collaboration and communication with relevant stakeholders.

The model developed in our remotely-reachable collaborative environment concerns the unclassified development of the design. Galileo has also a significant security contribution which comes with different levels of classification and need-to-know. We identified the need for n additional models developments. Each model relies on elements present in the unclassified model (Uncla) or in a less "secure" part of the model. For illustration purposes, let's call these models SEC1 to SECn. Two strategies were identified for the development. The first approach builds the models in sequence: SEC1 starts from Uncla, SEC2 starts from SEC1 and so on until SECn that starts from SEC(n-1). In this approach, the completion of the previous security level (all logical functions and functional exchanges or exchange items created) is required to initiate the next level and this was not compatible with the development time until PDR. Therefore, we moved to a second approach where each classified model is built in parallel in different branches in the repository, all starting from the unclassified model of reference. Anyhow, some modifications have to be included in the unclassified model of reference, followed by a resynchronization of all SECi models to the latest unclassified version. This will be the PDR reference G2G system model. The n+1 branches of the model will be kept for future developments and follow-on of the next phases.

It is to be noted that each SECi model is constructed in a similar way as the unclassified and the same views have also been developed. This provides us with confidence on the details and consistency level achieved to follow the segment developments, to analyse deviations or non-compliances, or to perform impact analyses in case of further mission evolutions.

Our current G2G model is composed of about 270 system and logical functions, 300 scenarios that may be further detailed to reflect specific usage or behaviours of the system

3. Generation of documentation

The formal reviews still rely on documentation: the design definition and justification files are key documents for system reviews and, although the system design is documented inside the modelling tool in the form of functional descriptions, functional interactions, segments interactions and operational interactions, it has been necessary to extract the information from the model for project reviews.

While in Phase B1 we used diagrams and tables we exported from the model to update the documents, we decided to investigate the possibility to automatically generate the documents from the model, with a few introduction and conclusion chapters written outside the tool. The plugin M2Doc was considered for this exercise. This turned out to be a difficult exercise at start but, once coding principles were understood, it gave us the possibility to generate the documents very easily, changing our mind in terms of order of figures and text, filtering which elements were to be present in the document, etc. The documentation exercise also helped us figuring out discrepancies or errors in the model itself that we could not sometimes determine through the validation rules inside the tool.

Typical features exported in a consistent manner for all areas of the design are:

- Description of system and logical functions;
- Functional decomposition diagrams;
- Diagrams and tables providing exchanges of a given functions with other system functions or external actors; description of the exchanges are also considered and interface documents to which they belong;
- Functional scenarios and their detailed description step by step

4. G2G (upcoming) use cases

The use cases for using MBSE developed as we were getting more familiar with the tools and techniques. Our core list is as follows:

- Use case 1 (starting point): establish and maintain the end-to-end system view as a supporting tool for ESA being system design authority for Galileo Second Generation;
- Use case 2: improve the interactions between system and segment contracts (and suppliers) beyond the traditional requirements baseline;
- Use case 3: perform impact assessments related to changes requested at mission level or stemming from segments feedback;
- Use case 4: establish and maintain several system configurations along the time line of development, deployment and operations (as specified, as built, as operated)
- Use case 5: establish and maintain several system versions to be specified for the successive system releases;
- Use Case 6: maintain traceability between model(s) and requirements baseline at all stages of the design for all versions.

While the System Phase B is entering its last stage with the upcoming PDR, Use Case 1 has been primarily tackled with the engineering team, leading to a first end-to-end system model view, and the other use cases are under investigation or initial implementation. Our initial findings for future development and usage of MBSE for our Galileo Second Generation are as follows.

Finding 1: Need for different models and modelling tools?

Our core development end-to-end is based is built with Capella deployed and used in a collaborative manner by the system design team with several industry partners including notably Thales and Airbus. However, the time and the complexity to build this end-to-end system view left us with the need to progress in parallel with some focused development under another tool, Innoslate, for fast assessment of specific technical topics. To keep one source of truth of the system design, one tool would sound the natural answer, but reality of programme developments show us that multiple tools may be more suitable to perform fast track analyses. The fast track is not necessarily fully representative of the end-to-end system but allow to get a good-enough feedback for the decision-making process. In addition, we started to use the Reqtify tool to bridge and allow dynamic traceability assessment between the model in Capella and the DOORS database gathering most of our requirements baseline.

Finding 2: The need for several configurations of the same system of interest to follow development and deployment with the segments

The model is targeted to be used as an engineering tool to iterate with the segment suppliers to become a common source of truth on the implemented system. The exchange of models is to be

structured so that it can be used on both sides in an efficient way. Besides the contractual aspects, the best would be to have an organisation that allows to not impose a tool (i.e. a segment could run with a different tool than the system) but imposing the details and levels of exchanges of the data. Read-only models may be avoided to preserve efficiency of use of the models (otherwise as good as a visio or a document). Along the development process and the iterations between the system and segments, the model will naturally evolves from an "as specified" to an "as designed" and finally an "as implemented" model. Maintaining a traceability to the applicable requirements is desirable to justify the compliance, however the amount of requirements in a project like Galileo makes this a challenge to maintain.

Finding 3: The need for configuration control.

As Agility is being enforced in Galileo (becoming a standard way of developing software in industry), the model(s) configuration control is made even more challenging and the tools have to support system engineers in identifying evolutions from one increment to the next. We need a solid version management for the models, structuring, organising and documenting the changes implemented, being able to have the documented design and access to any version of interest in a simple way. The problem to tackle is basically the following: each system/segment or element version/release has its own set of requirements documents/database to which it complies. Ultimately we deliver one system release after the other integrating all segments themselves integrating all of their elements, but their development time could overlap, and modifications made in version N under development may become also applicable to version N+1 under development at the same time and maybe with another industry. From a global perspective we need to have the possibility to show in a visual viewpoint (as opposed to text or list of requirements or compliance to requirements):

- What is unchanged
- What is new
- What is modified

Today's tools are offering some capabilities but this is still limited with respect to the wide variety of situations we have to face and some "manual" steps are still expected until and end-to-end configuration control can be considered.

5. Future work

Since we do have now a complete end-to-end model including all its security components, we are initiating the expansion of the work in several directions:

- Creation of full gradual end-to-end system versions keeping our full Galileo second Generation model as reference;
- Implementing a dynamic traceability of the functional design with the functional requirements at system and segment/facility level; this is foreseen using the Reqtify tool that allows linking Capella and DOORS ;
- Implementing the traceability between the ICD/IRD and the MBSE elements
- Implementing the link between the MBSE model and the various performance models.
- Investigating ways to iterate in the future at model-level with the development contracts for each component that will also use a MBSE methodology.