DESIGN AND SAFETY ASSESSMENT OF ON-BOARD SOFTWARE APPLICATIONS USING THE AMASS PLATFORM

Elena Alaña, Javier Herrero

GMV Aerospace and Defence (GMV), Spain, E-mail: ealana@gmv.com, jaherrero@gmv.com

Abstract: The complexity of the on-board software applications in Space Systems is continuously growing, together with the amount of data processed on-board. One of the reasons is the degree of autonomy. Current space missions are becoming more and more autonomous, implying that some computations (e.g., advanced and complex algorithms) and tasks previously performed on-ground are now executed on-board. Hence, the development lifecycle should be improved. The Space Agencies and the Space Industrial effort is focusing on advancing the design towards a model-based approach.

This paper presents the preliminary results of one of the Space Use Cases carried out in the frame of the AMASS (Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems) project, that assesses the design and safety processes of on-board software applications using the AMASS Reference Tool Architecture and its Tool Framework.

Keywords: Architecture-Driven, Contracts, Component-Based Design, Model-Based, Multi-Concern Assurance, Safety, Qualification.

1. AMASS PROJECT

1.1. Overview

AMASS (Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems) [1] is a H2020-ECSEL (Electronic Components and Systems for European Leadership) project that will create and consolidate the de-facto European-wide open tool platform, ecosystem, and self-sustainable community for assurance and certification of Cyber-Physical Systems (CPS) in the largest industrial vertical markets.

This three-year project was started in April 2016. The consortium is composed of 29 partners from eight countries and led by TECNALIA Research & Innovation. It includes the main stakeholders for CPS assurance and certification: Original Equipment Manufacturers (OEMs), system integrators, component suppliers, system assessors, certification authorities, tool vendors, research institutes, and universities.

The AMASS project aims:

- To lower certification/qualification costs.
- To demonstrate a potential reuse of assurance results.
- To increase the harmonization and interoperability of assurance and certification/qualification tool technologies.

1.2. AMASS Reference Tool Architecture and Tool Framework

The AMASS project is producing a Reference Tool Architecture, supported by the open AMASS Tool Platform, which can be applied to the aerospace, automotive, industrial automation, space, and railway domains.

The AMASS Reference Tool Architecture is being built on the results from previous successful EU projects such as OPENCOSS [2], SafeCer [3], CRYSTAL [4], CHESS [5], and SESAMO [6]. Namely, it extends the OPENCOSS and SafeCer conceptual, modelling and methodological frameworks for architectural-driven and multi-concern assurance.

This Reference Tool Architecture is sustained by concepts such as building blocks, collaborative tool environment, architecture-driven, multi-concern assurance, reuse, viewpoints, etc. The Reference Tool allows the designer to use basic functionalities (i.e., system component modelling, assurance case modelling, evidence management, compliance management) which are complemented with specific functionalities (i.e. system architecture-driven tools, multi-concern assurance, seamless interoperability, cross/intra-domain reuse tools). It also encompasses assurance and certification/qualification activities.

The AMASS Open Tool Platform is a concrete implementation of the Reference Tool Architecture. It is an open technological solution and represents an
implementation of the AMASS Reference Tool Architecture. The Eclipse open-source tools are used as tool infrastructure together with open standards for interoperability with external tools. Tab. 1 lists the selected technologies. They are categorized according to the 4 pillars of AMASS, that correspond to the Scientific and Technical Objectives (STOs):

- **Architecture-Driven Assurance (STO1):** It allows for explicit integration of assurance and certification activities with the CPS development activities, including specification and design.

- **Multi-Concern Assurance (STO2):** Tool-supported methodology for the development of assurance cases, co-assessment and contract-based assurance, which addresses multiple system characteristics.

- **Seamless Interoperability (STO3):** Open and generically applicable approach to ensure the interoperability between the tools used in the modelling, analysis, development of CPS, among other possible engineering activities.

- **Cross/Intra Domain Reuse (STO4):** Consistent assistance for intra-and-cross-domain and/or cross-concern reuse, based on a conceptual framework to specify and manage assurance and certification assets.

Table 1. Tools integrated in the AMASS Open Tool Framework

<table>
<thead>
<tr>
<th>STO</th>
<th>TOOLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture Driven Assurance</td>
<td>Papyrus Editor and its CHESS extension, including analysis tools (OCRA, nuXmv, xSAP) \ External Tool: SAVONA, Simulink, KM++ V&amp;V Manager</td>
</tr>
<tr>
<td>Multi-concern Assurance</td>
<td>OpenCert, EPF tool, BVR Tool, CHESS ML \ External Tools: ConcertoFLA, Safety Architect, Papyrus SSE</td>
</tr>
</tbody>
</table>

1.3. **AMASS Use Cases**

AMASS includes 11 industrial use cases, each one covering some areas of the project in the different domains (i.e., automotive, space, aerospace, railway, avionics). The demonstrators allows the verification of the AMASS approach and the identification of potential improvements. This paper describes one of the space demonstrators based on the Sentinel-3 mission.

2. **SPACE USE CASE: SENTINEL-3**

2.1. **Description**

The AMASS results are particularly relevant for the space domain. The space Stakeholders are focusing on designing critical on-board software applications towards a model-based approach. This approach shall allow Software and Safety Engineers:

1. To address dependability and safety aspects early in the development process.
2. The reuse of components from one mission to another.
3. Reduce time and costs needed for ensuring the dependability and safety aspects.

The aforementioned improvements can allow a better efficiency in the development of critical software applications.

GMV leads the Space Use Case named “Design and safety assessment of on-board software applications in
Thales Alenia Space, Intecs, Fondazione Bruno Kessler and Tecnalia participate in the demonstrator. The demonstrator is based on one of the instruments of the Sentinel-3 mission.

2.2. Sentinel-3 mission

The Sentinel-3 mission is part of the European Commission’s Copernicus programme, an environmental monitoring programme that tackles the effects of climate change and safeguard everyday lives. This satellite measures Earth’s oceans, land, ice and atmosphere to monitor and understand large-scale global dynamics. It also provides information in near-real time for ocean and weather forecasting.

This mission is based on two identical satellites orbiting in constellation. The first satellite (Sentinel-3A) was launched on 16 February 2016 capturing the first images two weeks after the launch [Fig. 1]. The second satellite (Sentinel-3B) joined on 25th April 2018, optimising coverage and data delivery for Copernicus.

Each satellite is composed of six payload instruments: SRAL (Synthetic Aperture Radar Altimeter), SLSTR (Sea and Land Surface Temperature Radiometer), GNSS (Global Navigation Satellite System), MWR (Microwave Radiometer), OLCI (Ocean and Land Colour Instrument) and DORIS (Doppler Orbitography and Radiopositioning Integrated by Satellite). Fig. 2 depicts these instruments together with the SMU (Satellite Management Unit), which represents the central intelligent core of the satellite at the same time as controls all the payload instruments (e.g., TC, TM, signals, etc.).

2.3. Sentinel-3 and AMASS

This Space Use Case covers some areas of AMASS process. In particular the Architecture-Driven Assurance, and some parts of the Multi-Concern Assurance.

The use case process and outputs are compared with the design and safety processes of the Sentinel-3 project to identify if AMASS outcomes improve:

- Components reuse from one mission to another.
- Re-qualification process when the execution platform (e.g., processor, communication buses, memory type) changes.
- The definition and verification of system safety requirements at model-level, being able to generate automatically dependability and safety evidences such as the Fault Tree Analysis (FTA).

Additionally, the compatibility with Space ECSS standards is verified. These standards shall be addressed to apply the proposed methodology in the space domain.
2.4. Implementation: Architecture-Driven Assurance

The Architecture-Driven Assurance process is completely covered. It relies on CHESS tool connected with tools, such as OCRA for contract-based analysis, nuXmv for model checking and xSAP for model-based safety analysis. The process is split into four activities:

1. Requirements specification (i.e. SysML requirements diagram) and formalization (i.e. formal properties and contracts).

Contracts specify the expected behaviour of a component by defining the assumptions that must be satisfied by the environment and the guarantees satisfied by the component in response [8].

This formalization at model-level allows traceability with design-entities, and verification at early development stages. Two analyses are conducted at this stage: requirements semantics analysis and contracts validation analysis.

2. Design of the System and Software architecture.

On the one hand, the static architecture is designed using SysML block definition diagrams, composite structure diagrams and class diagrams. This architecture is subsequently refined using SysML internal blocks diagrams.
On the other hand, nominal and faulty behaviour is modelled using UML state machines. They include the modes, the events that trigger the change of mode and the effects of these changes. SMV language is used to specify these transitions and effects.

Figure 5. OPSW Nominal modes

Models include non-functional properties such as contracts. CHESS provides new views that facilitate the visualization of this information: CHESS Contracts Decomposition view, CHESS Hierarchical Model view, and CHESS V&V Results view.

Figure 6. CHESS Hierarchical Model view

3. Conduction of safety analyses.

Once the architecture is available, several safety analyses can be conducted using the system and software model as input. They are used to demonstrate the safety of the system under development. In particular:

a. Model-checking. Verification of state machine properties.

b. Formal properties consistency. It allows the simulation of different scenarios configuring the default value of some properties. In case the validation fails, a counter example is shown.

c. Contracts analyses. It includes: i) contract-based verification, ii) contract refinement and iii) verification of strong/weak contracts. Strong contracts capture those behaviours that are expected to hold in all environments, whereas weak contracts are specific to certain environments.

Figure 7. Consistency check of formal properties

d. Safety analyses. It covers: i) model-based safety analysis (Fault Tree Analysis, FTA) and ii) contract-based safety analysis.

Fig. 9 illustrates a Fault Tree that shows the possible failures that can lead to a failure of the system implementation in satisfying the top contract.

The safety case collects all the relevant output evidences from the safety process using the AMASS Tool Framework. Evidences such as V&V and safety analyses, traceability matrices, documentation, etc.

Fig. 10 illustrates how the evidences are storage in the tool, whereas Fig. 11 depicts part of the documentation generated directly from the model. The documentation is produced in HTML format.

2.5. Implementation: Multi-Concern Assurance

Assurance represents the planned and systematic activities to get justified confidence that systems conform to its requirements for safety, security, reliability, availability, maintainability, standards and regulations. The various system characteristics that play a role in assurance are often called “dependability”. AMASS provides tool support for the development of assurance cases. The assurance case is a set of auditable claims, arguments, and evidence created to support the claim that a defined system/service will satisfy the particular requirements.

A simplified assurance case specification is being developed to check the relationship with the system component specification. Currently, two standards are being modelled: the ECSS-Q-ST-40C [9] and the ECSS-E-ST-40C [10]. These standards are being tailored for the use case needs creating an OpenCert assurance project.
During the next iteration of the case study, a complete assurance case will be produced.

### 2.6. Status

Both, the AMASS platform and the case study are being developed in an iterative way producing three deliveries along the project: the first release took place in mid-2017; the second one in early 2018 whereas the final version is expected to be delivered in 2019.

This paper is based on the second version of the space demonstrator. It was produced in March 2018 and presented in the 5th AMASS Plenary Meeting that was held in Vienna. This version will be refined and completed in 2019, providing:

- A complete assurance case.
- Assessment of the applicability of AMASS methodology to the space domain, analysing the impact of the AMASS framework to develop space applications.
- Comparison between the AMASS approach and the real OLCI development using a set of metrics: on the one hand, global AMASS metrics applicable to any case study; on the other hand, specific metrics for this demonstrator.

### 3. Conclusion

The dependability and safety processes are mostly carried out manually in the real development process, and conducted at the same time as the software was developed, with no tool support. That was the case of the OLCI ICM software. Therefore, space application can take advantage of AMASS methodology to conduct this analysis early in the development process relying on the AMASS Tool Framework to design the architecture and attached the safety/security information.

In AMASS, these processes are tightly coupled to the model-based design and the evidences are automatically generated from the model and available in the AMASS Tool Framework. They demonstrate the fulfilment of the project requirements. These evidences might derive new requirements or design constraints which can be introduced back in the model (i.e., iterative process).

The methodology can be applied to design critical systems, producing a model that stores the development process artefacts, including non-functional aspects (i.e., contracts) and allows the execution of safety analyses. This approach improves consistency (single source of information) and facilitates reuse (reusing not only building blocks but also evidences). This will lead to a reduction on time and costs at the same time as enforcing safety aspects. Safety analysis are directly conducted from the model. Then, changes in the architecture are directly reflected (if applicable) in the safety results.

### Acknowledgements

The research leading to this paper has received funding from AMASS project: National funding and H2020-ECSEL Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 692474. This Joint Undertaking receives support from the European Union’s Horizon 2020 research and innovation programme and from Spain, Czech Republic, Germany, Sweden, Austria, Italy, United Kingdom, France.

We also thank to the AMASS consortium, specially, Thales Alenia Space, Intecs, Fondazione Bruno Kessler and Tecnalia which participate in this space demonstrator.

### References

[1] AMASS Website: [https://www.amass-ecsel.eu/](https://www.amass-ecsel.eu/)
[7] Sentinel-3 Image gallery: [http://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Sentinel-3](http://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Sentinel-3)
