Model-Based CubeSat Flight-Software Architecture using a Docs-as-Code approach

Introduction

Model-Based Systems Engineering (MBSE) is an approach to systems engineering whereby models, as opposed to documents, serve as the *authoritative source of truth* for conducting systems engineering activities, such as the design, specification, analysis, verification & validation of a sytem [1].

The NASA handbook on systems engineering can be applied to CubeSat Mission design, in effort to facilitate a top-down design methodology from mission concept to specification of subsystem components, including flight software architecture [2].

For the SeaLion Mission, a joint CubeSat mission between the Old Dominion University and Coast Guard Academy, the flight software team adopted an MBSE approach to flight software architecture, using a docs-as-code approach [3], whereby the same tools and methodologies for managing software are also used for configuration management of documentation. Applying both an MBSE and docs-as-code approach means that aspects of the mission architecture, such as stakeholder needs, user stories, and data structures pertaining to the CubeSat mission, are captured are within a model that is both human-readable and machine-queryable.

This article presents the modeling language, tools, and technical approach used to fascilitate the configuration management, design, specification, & implementation of the SeaLion mission architecture for the flight software.

Background

Information in Systems Engineering today is mostly captured informally, not authored based on a methodology, configuration managed in silo tools, adhocly and infrequently integrated, not easily traceable to its provenance, not properly configuration managed, not properly changed managed, and not effectively shared with stakeholders.

The goal of the SeaLion CubeSat flight software architecture was to capture the data structures and expected behaviors of the flight software, such that it can be unambiguiously understood enough to be implemented, as well as provide full traceability and rationale for architectural elements with minimal configuration management overhead.

The MBSE approach was of interest to SeaLion CubeSat flight software team, as to yield the benefits of reducing ambiguity that usually comes with using informal language for specifying aspects of a system, as well as minimizing duplication of content that tends to accumulate in a document-based systems engineering approach.

Furthermore, it was also of interest to adopt a docs-as-code approach, as to yield the benefits of utilizing the same tools used to configuration manage code, using version control tools, s.a., Git, for the configuration management of flight software architecture documentation, captured in a model-based approach.

Selection of a modeling language, modeling tool, and technical approach need be taken into consideration, in order to proper adopt an MBSE approach.

Other considerations include overhead incurred with training the team, as well as technical overhead with setting up and maintaining the modeling tools.

For the aforementioned reasons, the SeaLion CubeSat flight software team downselecting the Mach30 modeling language for specifying the SeaLion CubeSat flight software architecture.

Stakeholder Needs

Table 1. Stakeholder Needs

ID / Name	Statement
1.1: Primary Mission Objective A1	The SeaLion mission shall establish UHF communication link with Virginia ground station
1.2: Primary Mission Objective A2	The SeaLion mission shall establish S-Band communication link with MC3 ground station
1.3: Primary Mission Objective A3	The SeaLion mission shall successfully transmit “mission data” defined above to ground stations on the Earth.
1.4: Primary Mission Objective A4	The SeaLion mission shall adhere to CubeSat standards.[1]
1.5: Primary Mission Objective A5	The SeaLion mission shall validate the operation of the Impedance Probe (IP) as a primary payload in-orbit.
2.1: Secondary Mission Objective B1	The SeaLion mission shall provide a means to validate a Multi-spectral Sensor (Ms-S) in-orbit
2.2: Secondary Mission Objective B2	The SeaLion mission shall provide a means to validate a deployable composite structure (DeCS) in-orbit
3.1: Tertiary Mission Objective C1	The SeaLion mission shall qualify on-orbit the deployment and functioning of the newly developed UHF antenna system and its deployment.
3.2: Tertiary Mission Objective C2	The SeaLion mission shall qualify a CubeSat bus architecture for very-low Earth orbit (VLEO)

ID / Name	Statement
3.3: Tertiary Mission Objective C3	The SeaLion shall verify DeCS in-orbit behavior performance.

User Stories

Table 2. User Stories

ID / Name	Statement	Derived From
1: Ping Satellite	As a **Ground Station Operator** I want to **Ping satellite** so that I can **Establish communication link with satellite**.	Primary Mission Objective A1
2: View Satellite Beacon Data	As a **Ground Station Operator** I want to **view satellite beacon data (alternating between health & mission data), received via UHF** so that I can **verify that satellite is operating nominally**.	Primary Mission Objective A1 Primary Mission Objective A3 Primary Mission Objective A5 Secondary Mission Objective B1 Secondary Mission Objective B2 Tertiary Mission Objective C1 Tertiary Mission Objective C2 Tertiary Mission Objective C3
3: Send Request to Set Interrupt Timer	As a **Ground Station Operator** I want to **send a request to set count value at which interrupt timers (i.e., beacon, GPS ping, or orbit propagator) are triggered** so that I can **finetune parameters for attitude or orbit analysis or to conserve power**.	
4: Request Telemetry or EventLog Data	As a **Ground Station Operator** I want to **Request satellite telemetry or eventlog data** so that I can **verify/validate health status or mission data**.	

ID / Name	Statement	Derived From
4.1: Request Satellite Health Data	As a **Ground Station Operator** I want to **request satellite health data packet** so that I can **verify/validate AODS sensors & GPS data are within nominal parameters**.	Request Telemetry or EventLog Data
4.1.1: Request Satellite Health Data via S-Band Radio	As a **Ground Station Operator** I want to **request satellite health data packet via S-band radio** so that I can **verify/validate AODS sensors & GPS data are within nominal parameters**.	Request Telemetry or EventLog Data Primary Mission Objective A2
4.2: Request Satellite Mission Data	As a **Ground Station Operator** I want to **request satellite mission data** so that I can **validate in-orbit AODS and/or payload performance**.	Request Telemetry or EventLog Data Primary Mission Objective A1 Primary Mission Objective A3 Primary Mission Objective A5 Secondary Mission Objective B1 Secondary Mission Objective B2 Tertiary Mission Objective C1 Tertiary Mission Objective C2 Tertiary Mission Objective C3
5: Send Request to Set Mission Mode Duration	As a **Ground Station Operator** I want to **send a request to set mission mode duration** so that I can **manage time spent per mission mode**.	

References

[1] S. Friedenthal and C. Oster, *Architecting spacecraft with SysML*. .

[2] S. A. Asundi and N. G. Fitz-Coy, “CubeSat mission design based on a systems engineering approach,” in *2013 IEEE Aerospace Conference*, Mar. 2013, p. nil, doi: 10.1109/aero.2013.6496900.

[3] E. Holscher, “Docs as Code,” [Online]. Available: https://www.writethedocs.org/guide/docs-as-code/.

[1] CubeSat Design Specification Rev. 13 https://www.cubesat.org/s/cds_rev13_final2.pdf