SSC19-P2-16

Developing a CubeSat System MBSE Reference Model – Interim Status #5

David Kaslow
DEKaslow Consulting
Newtown Square, PA
610 405 6685
david.kaslow@gmail.com

Philip T. Cahill
Consultant
Bryn Mawr, PA
610 787 0283
navyred@msn.com

Rodger Frank
Consultant
Annapolis, MD
814 360 6413
zugzwang6@gmail.com

ABSTRACT

Model-Based Systems Engineering (MBSE) is the formalized application of modeling in support of systems engineering. The International Council on Systems Engineering (INCOSE) Space Systems Working Group (SSWG) has been investigating the applicability of MBSE for designing CubeSats including developing a CubeSat System Reference Model (CSRM). Our application of MBSE is enabled by the graphical modeling language Systems Modeling Language (SysML).

The CSRM is intended for use at the university level for designing and building a mission-specific CubeSat. Additionally, the CSRM will be submitted to the Object Management Group (OMG) as a candidate OMG Specification.

This paper provides an overview of the CSRM including the requirements hierarchy, architecture hierarchy, and the incorporation of stakeholders, technical measures, and use cases. There are two modeling effort. One is the development of the CSRM with its logical architecture. The other is a mission team using the CSRM as a basis for its mission-specific logical and physical architectures. This paper addresses 1) the validation of the CSRM, 2) the application of the CSRM by a mission team, 3) and the validation and verification of the Mission-specific CubeSat Model.

INTRODUCTION

The International Council on Systems Engineering (INCOSE) Space Systems Working Group (SSWG) began investigating the applicability of MBSE for designing CubeSats in 2011.

CubeSat

CubeSat, a type of nanosatellite, is a low-cost standardized satellite with its origin in the CubeSat Project which was established in 1999 by California Polytechnic State University (Cal Poly), San Luis Obispo and Stanford University's Space and Systems Development Laboratory (SSDL).

The basic CubeSat unit is a 10x10x10 centimeter cube with a mass of about 1.3 kilograms, and this cubic unit is referred to as 1U. Over the years, this basic form factor has been modified to include larger form factors such as 3U, 6U and 12U CubeSats. They are typically launched as secondary payloads or deployed from the International Space Station.

MBSE

Model-based systems engineering (MBSE) is the formalized application of modeling in support of all design activities (e.g. system requirements, design, analysis, verification, and validation activities) beginning in the conceptual design phase and continuing throughout development and later life cycle phases including operations, maintenance, and retirement. References [1] and [2] provide additional information about MBSE including the practice of MBSE using SysML.

In MBSE, requirements and design are captured in the model rather than the traditional document centric series of independent engineering artifacts. Our application of MBSE is demonstrated using Systems Modeling Language (SysML) and a graphical modeling tool.

SysML

Systems Modeling Language (SysML) is a graphical modeling language for modeling complex systems including hardware, software, information, personnel, procedures, and facilities[2].

Utilizing the semantics of SysML, packages are used to organize model elements. A block is the fundamental SysML modeling element. A block can define 1) a type of logical or conceptual entity; 2) a physical entity, a hardware, software, or data component; 3) a person, a facility, an entity that flows through the system; or 4) an entity in the natural environment.

SysML has modeling elements for: requirements, structures, behaviors, and parametrics. Structural elements and their relationships are presented in block definition diagrams and internal block diagrams. Behaviors describe how a block deals with inputs and outputs and changes to its internal state - what the system must do to meet requirements. Behaviors are presented in activity, state machine, sequence, and use case diagrams. Parametrics are the mathematical formulations that describe non-functional requirements or emergent properties of the system and may also be needed by system simulators.

CSRM

The CubeSat System Reference Model (CSRM) is a product of the INCOSE SSWG and is a proposed Object Management Group (OMG) normative specification.

The name has been changed from CubeSat Reference Model (CRM) to CubeSat System Reference Model (CSRM) to reflect that this model encompasses the space and ground systems not just the CubeSat itself.

The CSRM provides the logical architecture of a CubeSat space and ground system. The CSRM logical components are intended to be reused as a starting point for a mission-specific CubeSat logical architecture, followed by the development of physical architecture during CubeSat development. The mission-specific team is free to adopt a different logical architecture and modify the CSRM to accommodate this change.

The CSRM adheres to MBSE, SysML, and primary references including the INCOSE Systems Engineering Handbook[1], A Practical Guide to SysML[2], NASA Systems Engineering Handbook[3], and Space Mission Engineering: The New SMAD[4].

The CSRM is a repository for systems engineering artifacts. However, it is not pre-populated with specific stakeholders, technical measures, use cases, and requirements. That is the job of the mission-specific CubeSat team. Development of a mission-specific CubeSat utilizing the CSRM establishes a mechanism to share and reuse components with other design activities.

It is important to note that the structure and content of the CSRM is intended to provide a rich "sand box" of engineering objects and relationships from which a necessary and sufficient, mission-purpose model can be constructed. In particular, the levels of abstraction in the CSRM for requirements, constraints and logical design are presented as suggestions and must be tailored to the mission purpose. Suggestions for pruning or expanding the CSRM "sand box" are included in this paper.

Figure 1 shows the CSRM Landing and Navigation page. The following five sections and subsections address the five over-arching CSRM elements below. Refer also to section III of Reference [5]. Figures 2-7 are as in Reference [5].

- Requirements
- Architecture
- Stakeholders
- Technical measures
- Use cases

Package diagrams have been created that establish relationships between these elements and provide for viewing and populating these elements.

Additionally, presented in this paper are:

- The inclusion of CubeSat Deployer Systems stakeholder
- Architecture and population of the CSRM elements
- Application of the CSRM - providing the CSRM to a mission-specific team and that team creating their Mission-specific Cubesat Model (MSM)
- Validation and verification of the MSM
- Validation of the CSRM

REQUIREMENTS AND ARCHITECTURE

The CSRM has established a requirements and architecture hierarchy as shown in Figures 2 and 3. The architecture hierarchy starts with the Cubesat Domain and the CubeSat Mission Enterprise as shown in Figures 4 and 5 and drills down to the Space and Ground segments and subsystems as shown in Figures 6 and 7. The hierarchy provides for the subsystem components as shown in Figures 2 and 3.

CUBESAT SYSTEM REFERENCE INFORMATION.

The CubeSat System Reference Information package, Figure 8, is the source of terminology definition and references. The CSRM underlines any terminology with a definition provided in the CubeSat System Reference Information. Hovering over the terminology will reveal the definition. The terminology in this paper is from this package however references are not included in this paper.

STAKEHOLDER

A stakeholder is any entity that has an interest in the system. As shown in Figure 9, stakeholders have various interests in the CSRM: Some are interested in the models themselves and others are interested in the missions that can be realized from the mission-specific instantiations of the model, and some have interests in both. Representative stakeholders are included in the model.

There are two modeling efforts. One is the SSWG developing a CSRM with its logical architecture. The other is a team eventually taking the CSRM as a basis for its mission-specific logical and physical architectures.

Model development stakeholders are concerned with the proper development, management, and distribution of the CSRM. Stakeholders include SSWG, INCOSE, and OMG.

CubeSat projects are pursued internationally, but the licenses and regulations that cover their activities are administered at the national level. The timelines and procedures for requesting and receiving approval must be well understood and part of the model. U.S. regulatory stakeholders include:

- Federal Communications Commission
- NASA Orbital Debris Program Office
- NOAA Commercial Remote Sensing Regulatory Office

The design and development of a mission-specific CubeSat must satisfy the requirements for interfacing with a CubeSat Deployer such as the Cal Poly CubeSat Design Specification.

Stakeholder Concerns

Stakeholder Concern - Interest in a system relevant to one or more of its stakeholders.

Mission Need - A concise description of a need or service that the system must provide.

Mission Objective - A broad set of goals that must be achieved in order to successfully satisfy the stated mission need.

Mission Constraint - A limitation placed on cost, schedule, or implementation techniques.

Mission Requirement - Statements of fact and assumptions that define the expectations of the system in terms of mission objectives, environment, constraints, and measures of effectiveness.

A stakeholder concern can be manifest in many forms, such as, goals, expectations, responsibilities, requirements, design constraints, assumptions, dependencies, quality attributes, architecture decisions, risks or other issues pertaining to the system.

There are typically a number of stakeholders each with a number of concerns. As shown in Figure 10, the needs, objectives, constraints, and requirements result from review, assessment, and integration of the varied concerns.

The CSRM has model elements for stakeholders, concerns, mission needs, mission objectives, mission constraints and mission requirements. These model elements and their relationships are mission and engineering methodology specific. The mission and methodology enables starting at different points in the process such as:

- Starting with stakeholder concerns
- Starting with mission objectives and mission constraints
- Using a simpler approach of starting with just mission requirements

This terminology is consistent with a number of well-established and accepted references. The user of the CSRM should establish terminology and then model elements as needed for their stakeholders and mission.

Technical Measures

Technical Measures provide a stakeholder insight into the definition and development of a technical solution. Technical Measures are typically non-functional requirements. Verification activities provide data to the technical measurement process that is used to assess how well the technical measure is either projected to meet, or is meeting, its stated value. Refer to Reference [6].

Measure of Effectiveness. An operational measure of success that closely relates to the achievement of a mission objective being evaluated, in the intended operational environment under a specified set of conditions.

Measure of Performance. A measure that characterize the physical or functional attributes relating to the system operation; i.e., it provides insight into the performance of the specific system.

Technical Performance Measure. A measure of the attributes of a system element to determine how well the system element is satisfying, or expected to satisfy, specified technical requirements.

Technical Measure Specifications capture descriptions of technical measures in textual form. Stakeholders will likely describe their technical measures as text. They will communicate and negotiate these descriptions with engineers who will transform them into measures that can be tracked and assessed.

As with the constraints and requirements hierarchies, the CSRM Technical Measure hierarchy should be truncated as needed to supply the specificity needed for the particular mission application – if the Technical Measures can be adequately expressed in two layers of a decomposition, stop there.

Technical Measure Methods are Word documents or Excel spreadsheets that contain the methods to calculate Technical Measures. These methods can be used as is or incorporated into parametric diagrams. The documents or spreadsheets are part of the model and can be edited within the model.

Use Case

A Use Case describes the functionality of a system in terms of how it is used to achieve the goals of its various users. The users of the system are described by actors, which may represent external systems or humans who interact with the system. Refer to Reference [7].

A Concept of Operations (ConOps) describes how the system will fulfill the stakeholder needs and objectives. What the system will do and the rational. An Operations Concept (OpsCon) describes how the system will be used from the operator’s perspective. How the system will do what it is intended to do when operated as designated.

The definitions of ConOps and OpsCon tend to be inconsistent across engineering organization and methodologies. Use Cases are the ConOps and OpsCon. The CSRM will not use the ConOps and OpsCon terminology.

Use Case Descriptions are text-based descriptions that provide additional information to support the use case definition. A use case description may include: pre-conditions, primary flow, alternate flow, and post-conditions.

CSRM ELEMENTS AND POPULATION

Figure 11 is an overview of the five over-arching CSRM elements. Also shown are the element properties that establish the relationships between the elements.

Figure 12 is an overview of the CSRM population package. Each package contains the associated CSRM elements and tables residing in the containment tree. The tables are used to add, delete, and modify the individual elements and to establish relationships between the elements. For example, see Figure 13.

CSRM APPLICATION

Figures 14, 15, and 16 illustrate 1) maintaining and providing the CSRM to a mission-specific team and 2) the mission specific team creating their mission specific model.

The Foundational CSRM is a fully capable CSRM implemented in a graphical modeling tool. It is the CSRM as provided to and evaluated by CubeSat teams. The Web Published CSRM allows a user to explore and evaluate the CSRM without the need to acquire a graphical modeling tool. The CSRM XMI File is the OMG normative specification. The Foundational CSRM is an implementation of the OMG normative specification

Mission-specific CubeSat Model

The Mission-specific CubeSat Model (MSM) is based on the CSRM. The MSM captures the design and development of the mission-specific CubeSat:

CSRM → MSM → CubeSat

The mission team downloads the CSRM from OMG to establish the initial instantiation of their MSM**.**

The mission team identifies the systems engineering methodology to be followed revising the MSM organization, elements, relationships, and diagrams as required:

- Confirmation of the above five over-arching elements as necessary and sufficient: stakeholders, technical measures, use cases, requirements, and architecture. If not, modify accordingly.
- Identification of all stakeholders and stakeholder terminology such as concerns, needs, objectives, and constraints, as they relate to establishing mission requirements.
- Resolution of the relationships between use cases, technical measures, and space and ground segment and subsystems requirements. Refer to the Figure 10.
- Selection of the model artifacts that are needed such as for stakeholder buy-off, internal and external design reviews, and design specification such as component interfaces and power, weigh, and performance properties.

The engineering methodology should address the extent that the technical measures, use cases, and requirements adequately define the design space. There is a trade-off between not enough detail and unnecessary detail. This has been referred to as requirements truncation or termination. That is, how do you know when enough is enough relative to requirements development and specification. The MSM supports defining requirements development all the way down to the subsystem component level or truncating requirement development based on defining use cases and verification strategies.

The MSM is maintained throughout all phases of development and operations as the single and central design baseline.

Transition: MSM Logical Architecture → MSM Physical Architecture

The MSM architecture includes packages for CubeSat subsystems and subsystem components. Initially these packages contain the logical subsystems and subsystem components from the CSRM. Logical components are representative of the types of components needed for a subsystem. For example: processor, memory, antenna, solar array, battery, and GPS receiver.

MSM systems engineering methodology that defines stakeholders, technical measures, use cases, requirements, and architecture will result in the translation of the logical architecture into the physical architecture as follows:

- Add, delete, and modify the subsystems
- Replace the representative subsystem logical components with mission-specific logical components
- Populate component properties such as power, weight, cost, and performance – which can be rolled-up from components to subsystems to the CubeSat

There are packages for adding, deleting, and modifying subsystems and subsystem components. Refer to Figure 12.

MSM and Project Management

The MSM can be the repository for project management artifacts such as identification of activities, assignments, and schedules.

MSM VALIDATION AND VERIFICATION

This section addresses the validation and verification of the MSM. This is when a team eventually uses the CSRM as a basis for its mission-specific logical and physical architectures.

MSM Verification

MSM Verification confirms, by providing objective evidence, that the system and all its elements perform their intended functions and satisfy the requirements allocated to them. That is, the system has been built right.

The Verification Activity element describes the process for verifying an element, e.g. a requirement. For example, the process could include verification plans, procedures, execution, results, and status. The process can be captured in a 1) Verification Activity element properties, or 2) Verification Activity Diagram, or 3) Verification Activity Word document.

The verify relationship is a relationship between a requirement and a test case or other model element that is used to verify the requirement is satisfied. The relationship is shown with a dashed line pointing from the test case to the requirement or other model element.

The Verification Activity element has a verifies property. The Requirement element has a verified by property.

Figure 17 is a conceptual representation of a verify relationship between Verification Activity and a Requirement

MSM Validation

MSM Validation confirms, by providing objective evidence, that the system, as-built (or as it will be built), satisfies the stakeholders' needs, objectives, and technical measures. That is, the right system has been (or will be) built.

The Validation Activity element describes the process for validating an element, e.g. a Technical Measure or a Mission Objective. For example, the process could include validation plans, procedures, execution, results, and status. The validation process can be captured in a 1) Validation Activity element properties, or 2) Validation Activity Diagram, or 3) Validation Activity Word document.

The Validation Activity element has a validates property. The Technical Measure or Mission Objective element has a validated by property.

Figure 18 is a conceptual representation of a validate relationship between Validation Activity and a Technical Measure or a Mission Objective. Note that the validations of Technical Measure and Mission Objective trace back to the Stakeholder.

Figures 17 and 18 are illustrations of the types of relationships that can be established between Validation Activities, Verification Activities, Stakeholders, Technical Measures, and Requirements. These relationships will also exist for the lower tier Technical Measures and Requirements. Conceptually, validation and verification relationships exist between elements, not between tables.

Refer also to Figure 11 for validation and verification properties of the Requirement, Technical Specification, Validation Activity, and Verification Activity elements.

The Validation Activity and Verification Activity elements have Active Hyperlink properties that can be used to link to V&V Activity Diagrams or V&V Word documents within the CSRM.

CSRM VALIDATION

This section addresses the validation of the CSRM by the SSWG.

The CSRM validation is based on:

- Satisfaction of requirements in the OMG CSRM Request for Proposal (RFP) as shown in Figure 19
- Evaluation in accordance with “A Practical Guide to SysML”
- Evaluations based on application by mission-specific CubeSat development teams

The intent is for the CSRM to be self-contained relative to the validation approach and results.

RFP

The following sections from the RFP have been parsed and populated into CSRM packages and tables:

4 – Instructions for Submission

5 – General Requirements on Proposal

6 – Specific Requirements on Proposals

Satisfaction of the requirements will be recorded in the tables with links to CSRM elements and packages. The tables can be exported for external review.

A Practical Guide to SysML

Sections 2.2.3 Model Evaluation and 2.2.4 Establishing Model Evaluation Criteria provide a set of model validation questions. A set of model validation requirements have been derived and mapped to the questions. The validation questions and requirements have been populated into CSRM packages and tables. Satisfaction of the validation requirements will be recorded in the tables.

Evaluation by Trusted Partners

The validation requirements take the form of “The CSRM shall provide the capability for the Mission CubeSat Development Team to ….”. Initially satisfaction of these validation requirements is based on the judgment of the CSRM development team. However, the satisfaction of these validation requirements is best evaluated by several Mission-specific CubeSat teams as noted in Figure 20.

Additionally, a CubeSat team could apply the Practical Guide model valuation questions to their mission-specific CubeSat model.

NEXT STEPS

The CSRM is sufficiently mature for validation to be accomplished followed by submission to OMG for candidate normative specification.

The Next Steps section of Reference [5] mentioned adding parametrics for power, weight, and cost. They have not been added since they are more appropriate for a physical model.

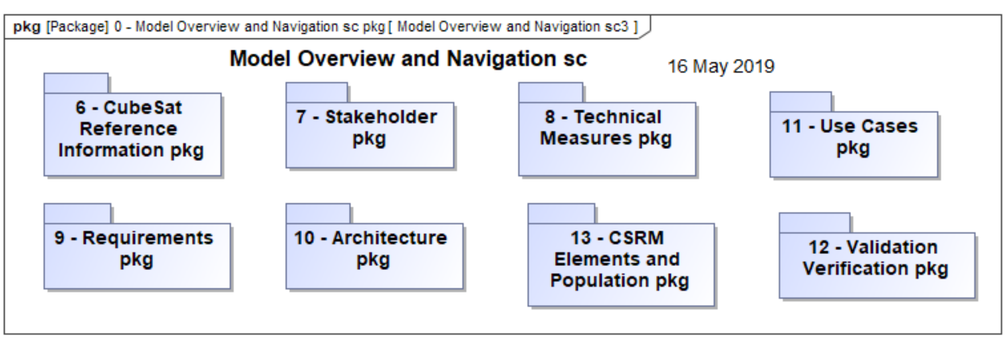

Figure 1. CSRM Landing and Navigation Page

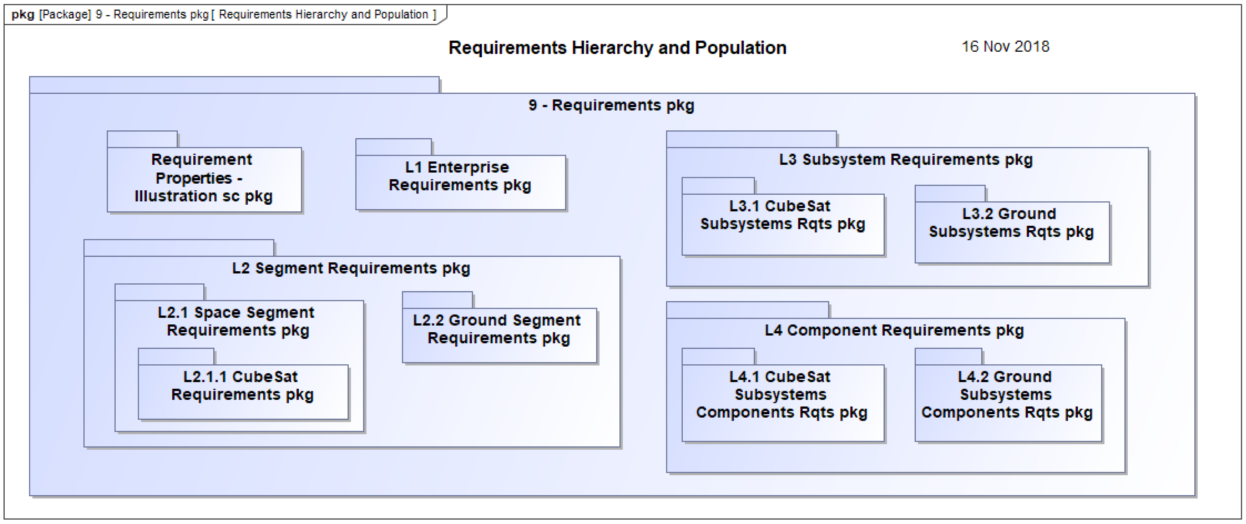

Figure 2. Requirements Hierarchy

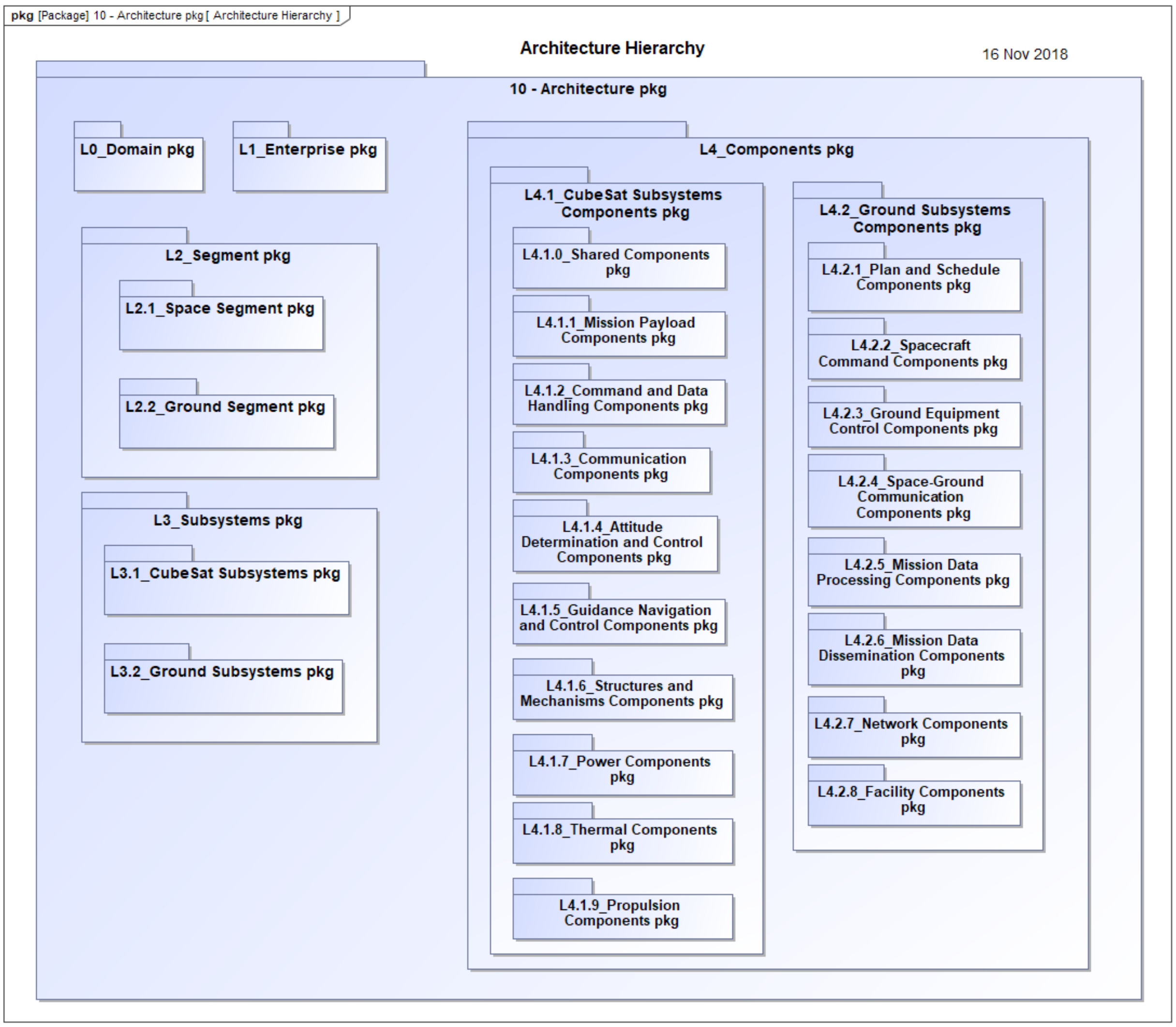

Figure 3. Architecture Hierarchy

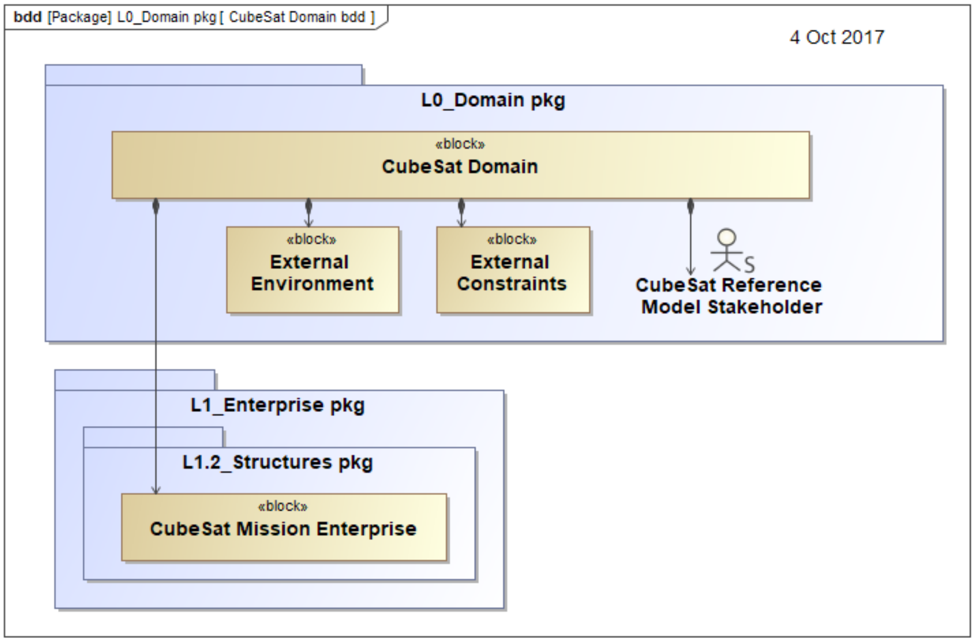

Figure 4. CubeSat Domain

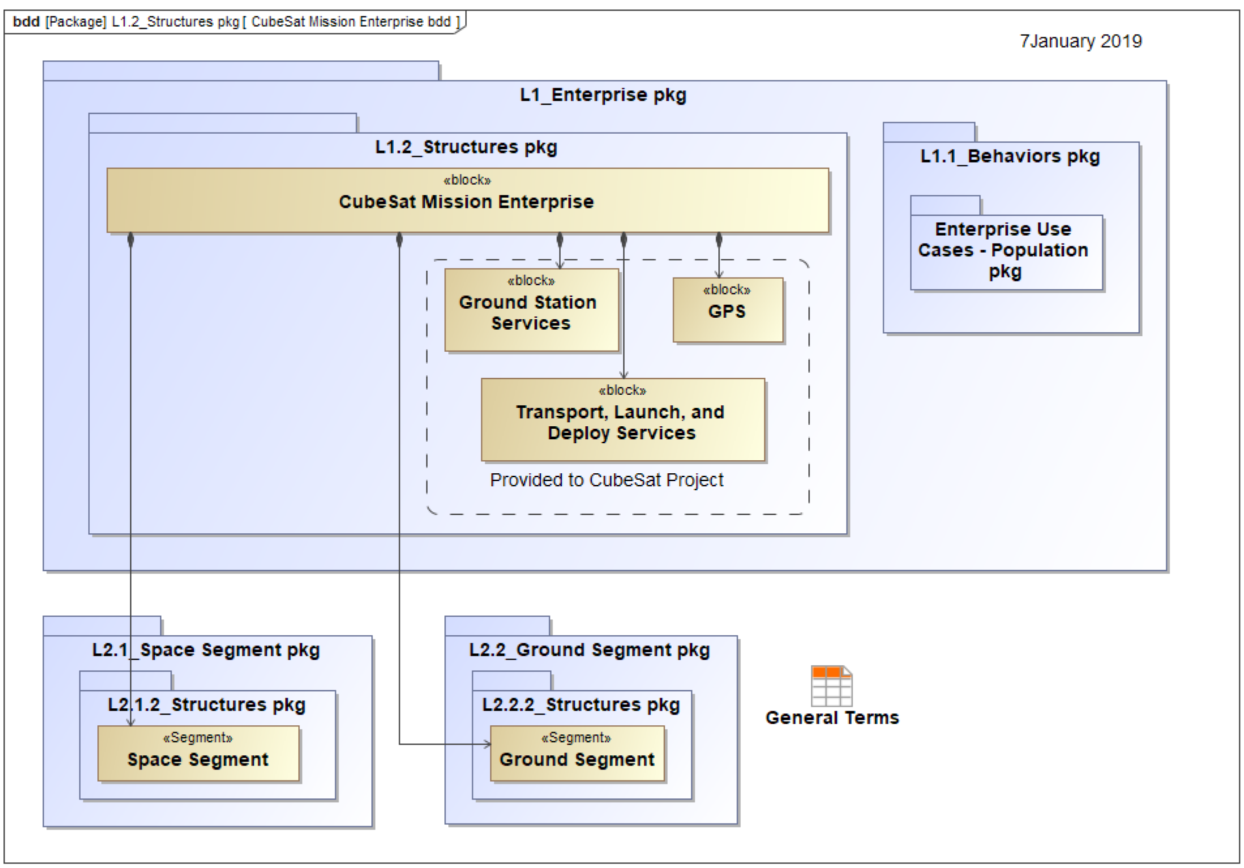

Figure 5. CubeSat Mission Enterprise

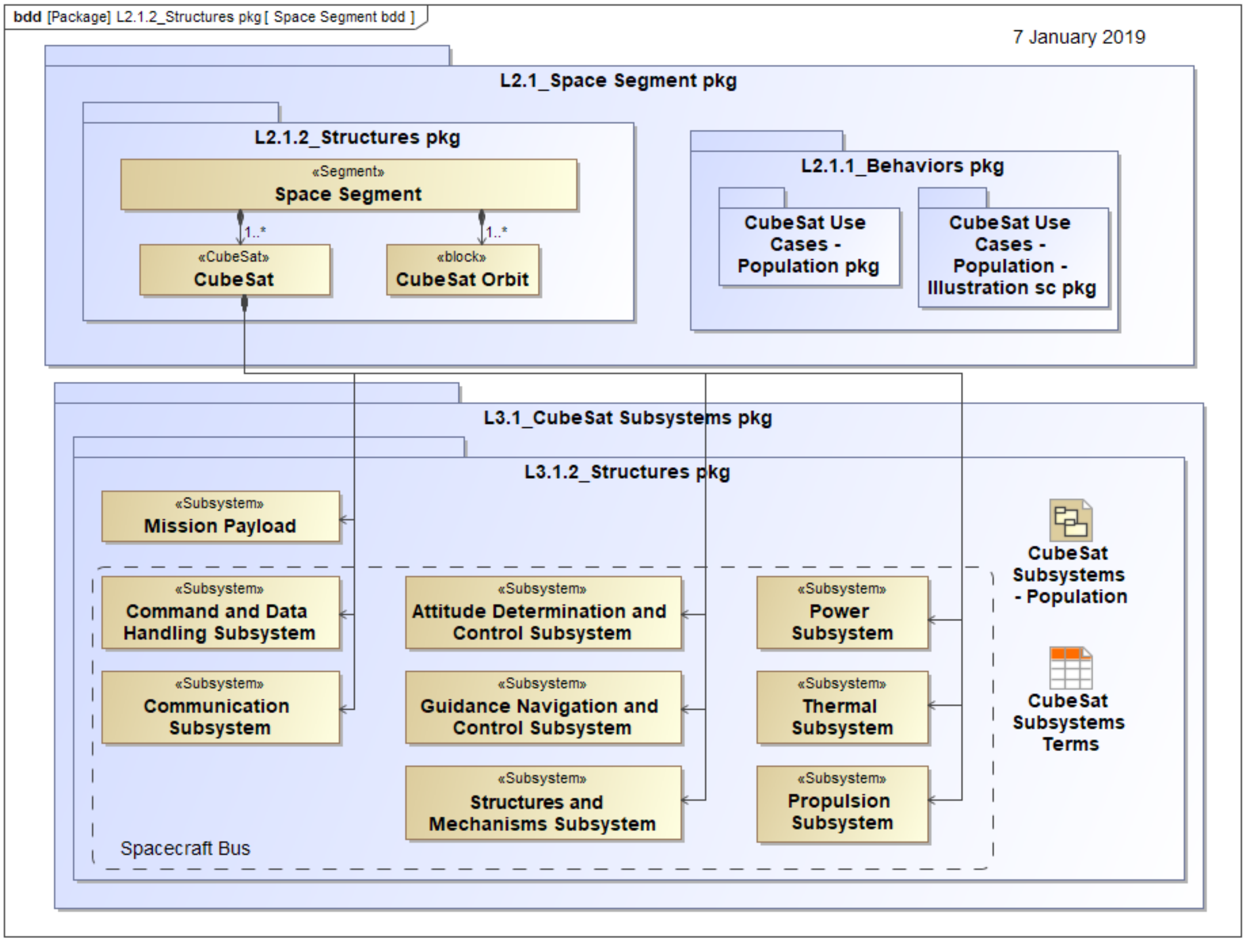

Figure 6. Space Segment

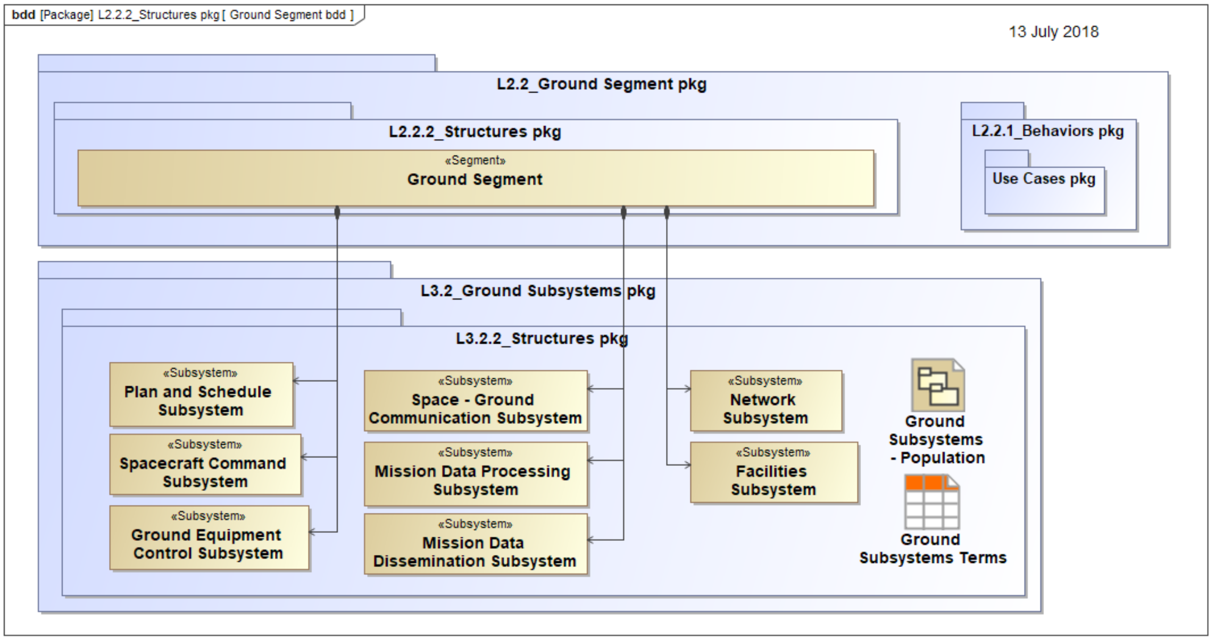

Figure 7. Ground Segment

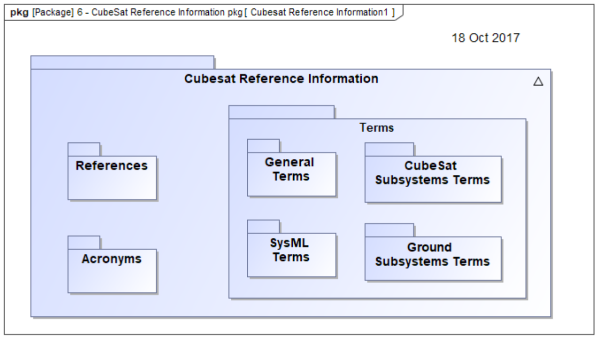

Figure 8. CubeSat System Reference Information

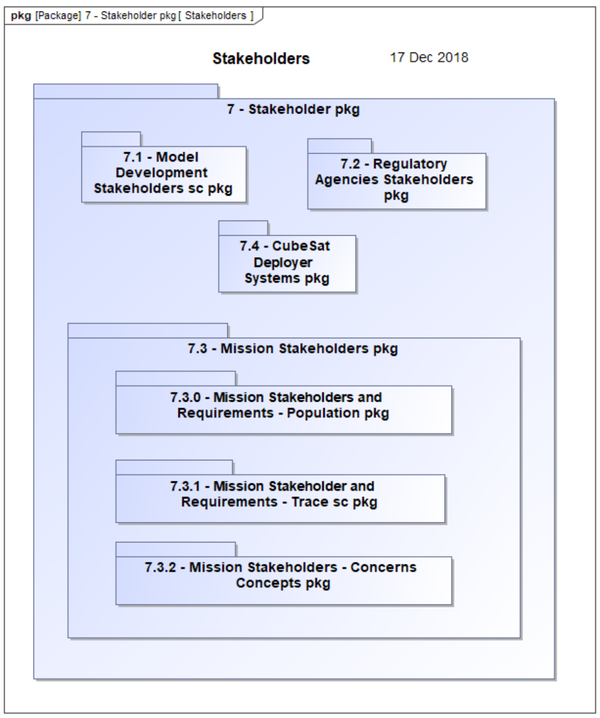

Figure 9. Stakeholders

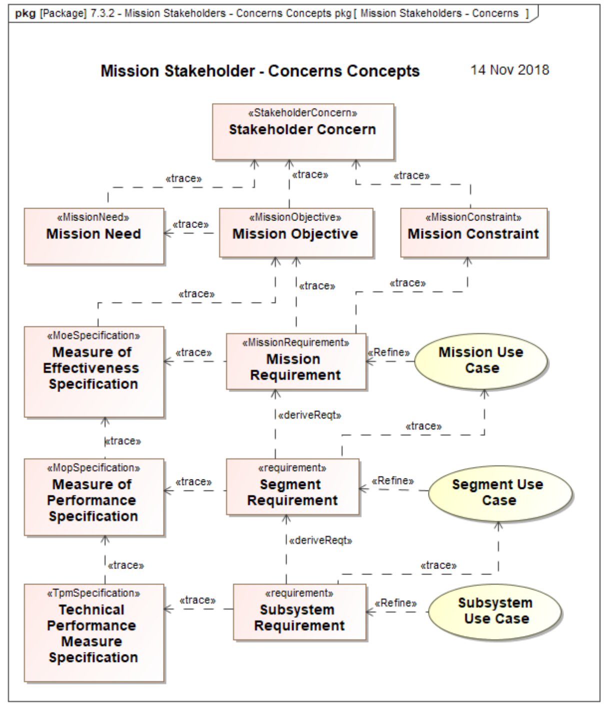

Figure 10. Mission Stakeholder Concerns

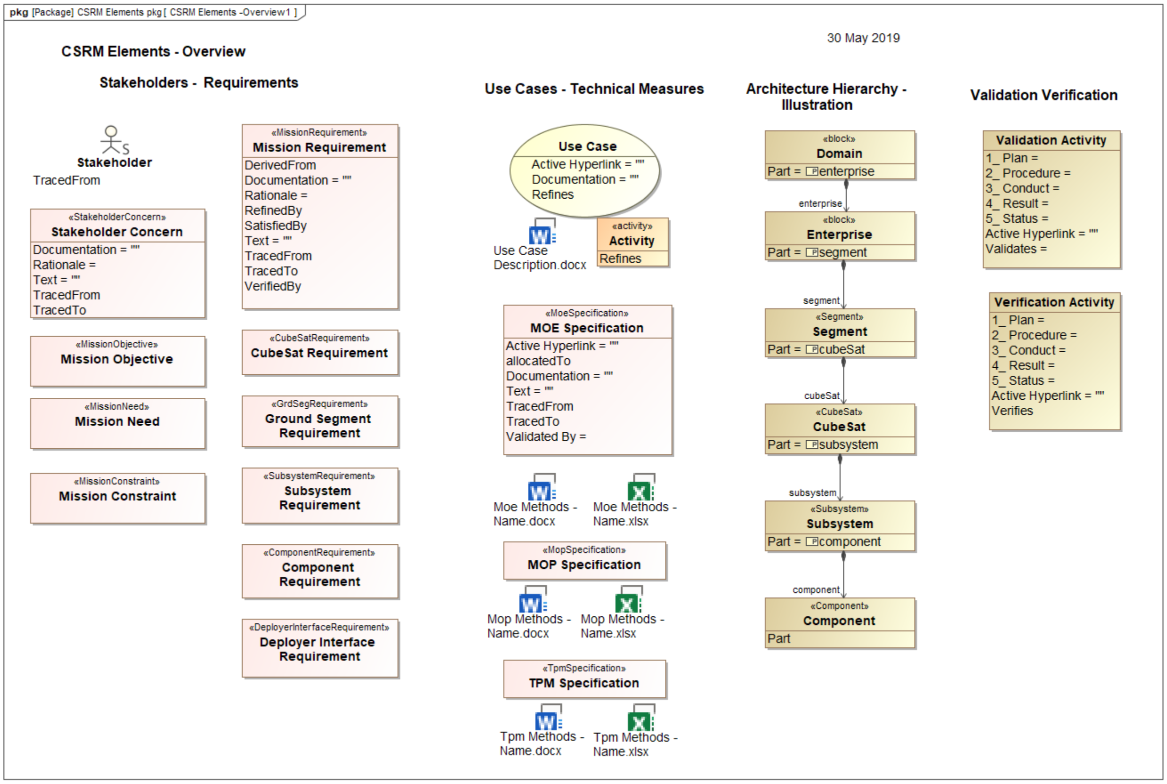

Figure 11. Overview of the Five Overarching CSRM Elements

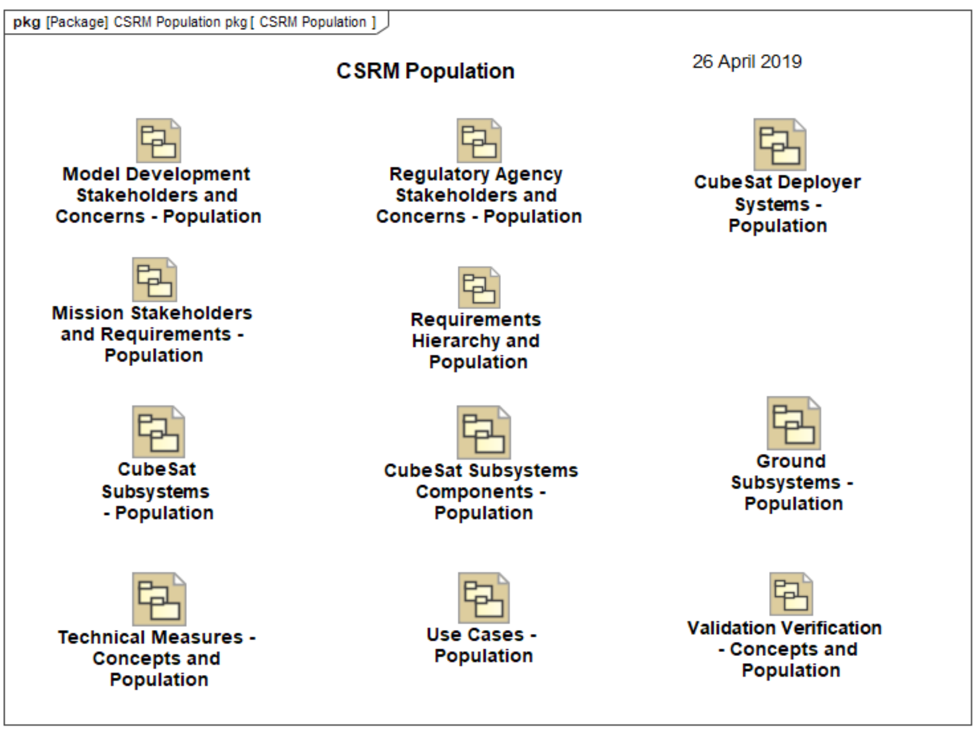

Figure 12. CSRM Population Package

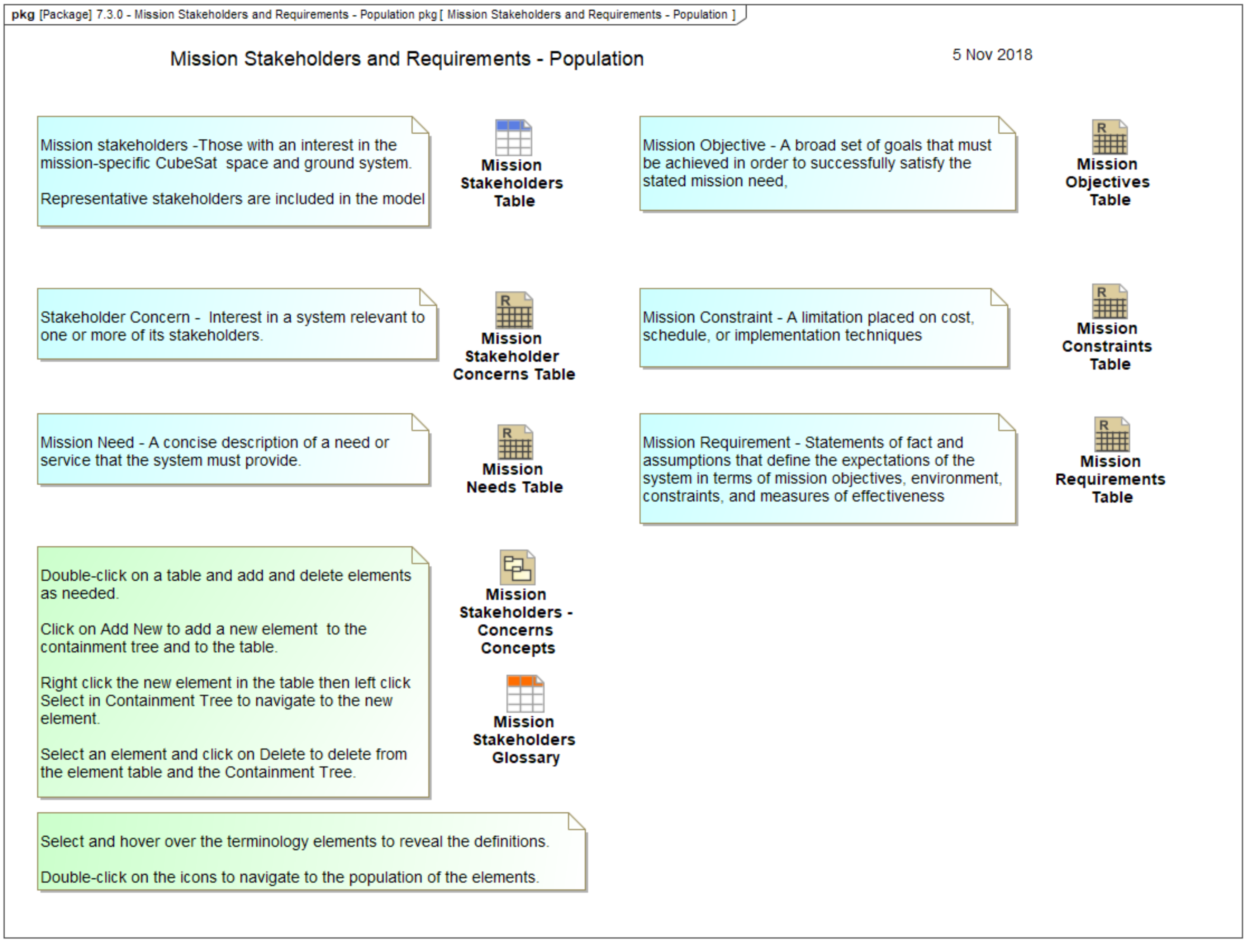

Figure 13. Mission Stakeholders and Requirements - Population

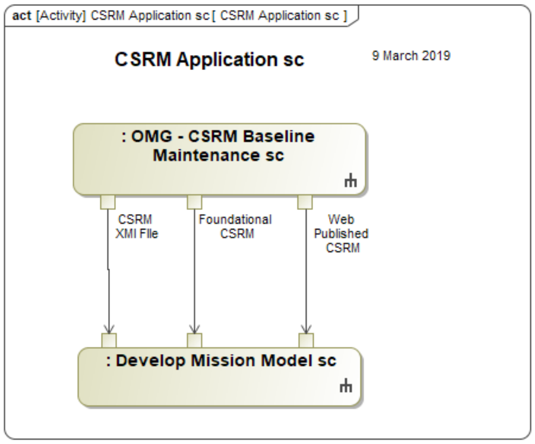

Figure 14. CSRM Application

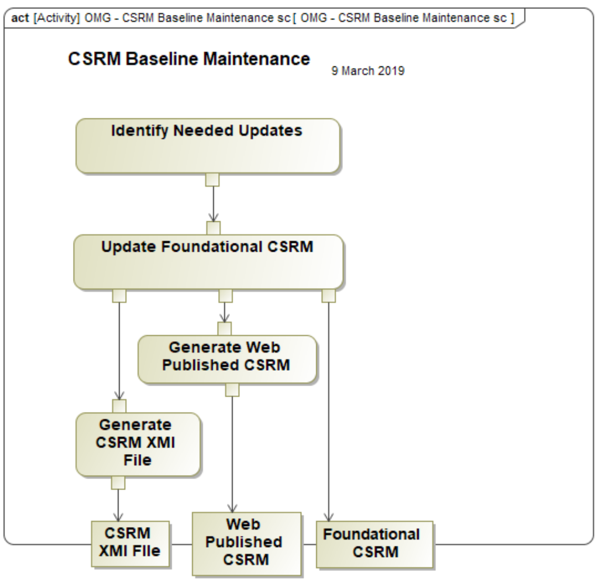

Figure 15. CSRM Maintenance

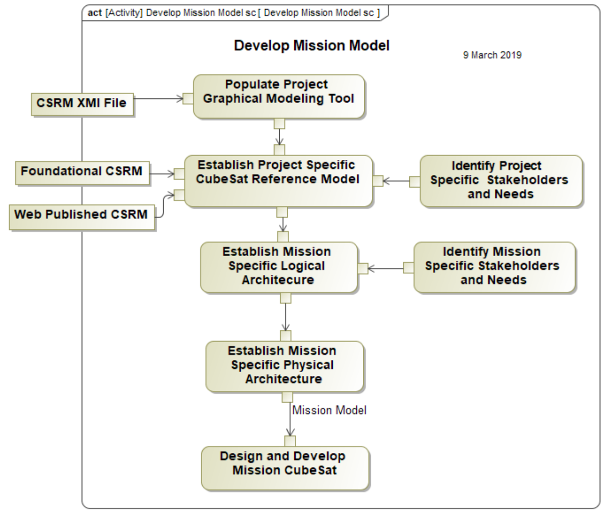

Figure 16. Developing Mission Model

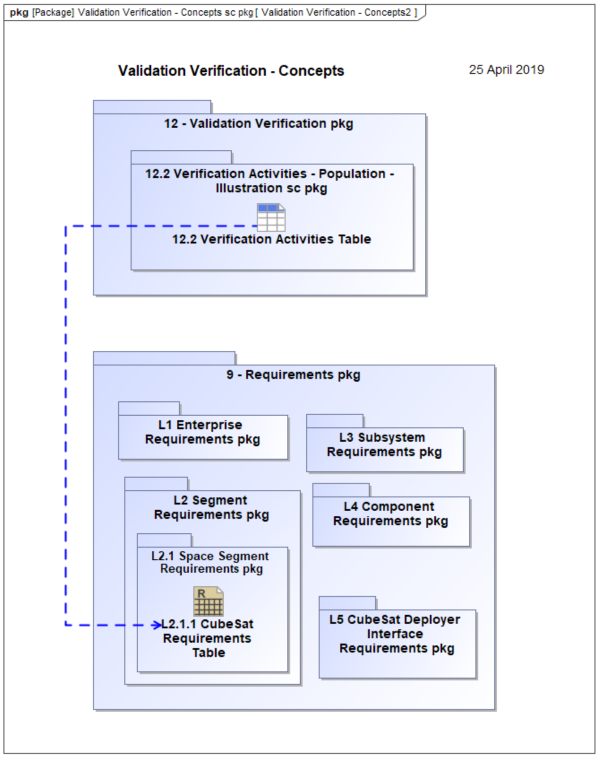

Figure 17. MSM Verification – Conceptual Representation.

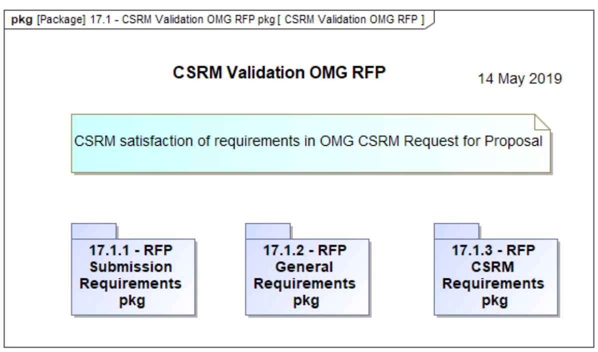

Figure 19. CSRM Validation

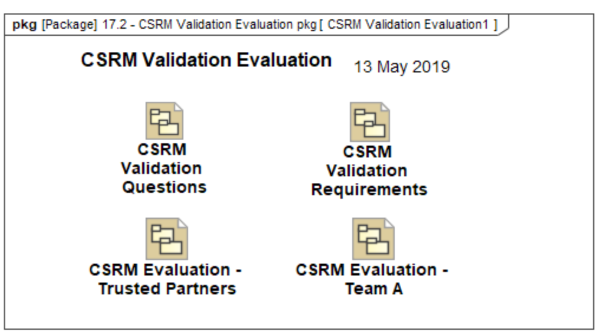

Figure 20. CSRM Validation Evaluation

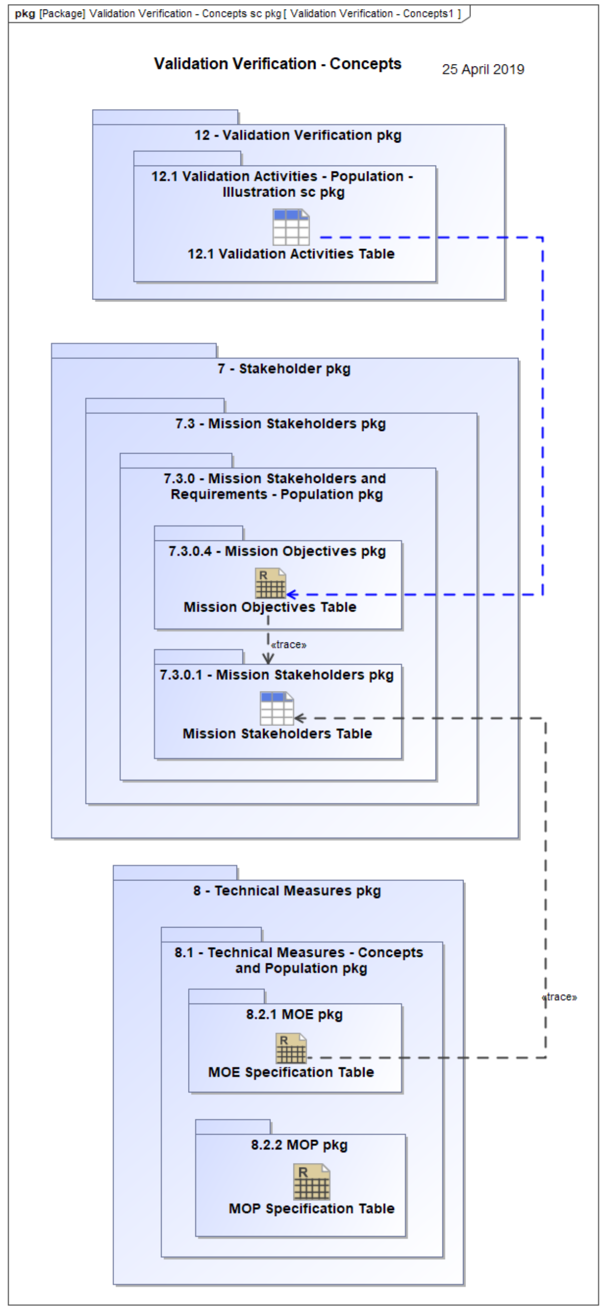

Figure 18. MSM Validation – Conceptual Representation

References

1 *INCOSE Systems Engineering Handbook*, 4th ed., INCOSE-TP-2003-002-04 2015.

2 S. Friedenthal, A. Moore, and R. Steiner, *A Practical Guide to SysML*, 3nd ed., Elsevier, Waltham, MA, 2015.

3 *NASA Systems Engineering Handbook*, rev. 1, December 2007, NASA/SP-2007-6105 Rev1.

4 J.R. Wertz, D. Everett, and J. Puschell, Eds., *Space Mission Engineering: The New SMAD*, (Space Technology. Library, Volume 28), Hawthorne, CA, Microcosm Press, 2011.

5 D. Kaslow, B. Ayres, P. Cahill, L. Hart, Croney, L Hart, A. Levi. "Developing a CubeSat Model-Based Systems Engineering (MBSE) Reference Model – Interim Status #4," *Proceedings of AIAA Space Forum.* Orlando, FL. 2018.

6 D. Kaslow, B. Ayres, P. Cahill, and L. Hart, "A Model-Based Systems Engineering Approach for Technical Measurement with Application to a CubeSat," *Proceedings of IEEE Aerospace Conference*, Big Sky, MT. 2018.

7 D. Kaslow, B. Ayres, P. Cahill, L. Hart, and R. Yntema, "A Model-Based Systems Engineering (MBSE) Approach for Defining the Behaviors of CubeSats," *Proceedings of IEEE Aerospace Conference*, Big Sky, MT. 2017.