An OMG® CubeSat System Reference Model Publication



CubeSat System Reference Model

**OMG Document Number: space/21-08-01**

**Normative Machine Consumable File:**

CSRM Profile – XMI **space/21-08-04**

Copyright © 2021, International Council on Systems Engineering.

Copyright © 2021, Object Management Group, Inc

USE OF SPECIFICATION – TERMS, CONDITIONS & NOTICES

The material in this document details an Object Management Group specification in accordance with the terms, conditions and notices set forth below. This document does not represent a commitment to implement any portion of this specification in any company's products. The information contained in this document is subject to change without notice.

LICENSES

The companies listed above have granted to the Object Management Group, Inc. (OMG) a nonexclusive, royalty-free, paid up, worldwide license to copy and distribute this document and to modify this document and distribute copies of the modified version. Each of the copyright holders listed above has agreed that no person shall be deemed to have infringed the copyright in the included material of any such copyright holder by reason of having used the specification set forth herein or having conformed any computer software to the specification.

Subject to all of the terms and conditions below, the owners of the copyright in this specification hereby grant you a fully-paid up, non-exclusive, nontransferable, perpetual, worldwide license (without the right to sublicense), to use this specification to create and distribute software and special purpose specifications that are based upon this specification, and to use, copy, and distribute this specification as provided under the Copyright Act; provided that: (1) both the copyright notice identified above and this permission notice appear on any copies of this specification; (2) the use of the specifications is for informational purposes and will not be copied or posted on any network computer or broadcast in any media and will not be otherwise resold or transferred for commercial purposes; and (3) no modifications are made to this specification. This limited permission automatically terminates without notice if you breach any of these terms or conditions. Upon termination, you will destroy immediately any copies of the specifications in your possession or control.

PATENTS

The attention of adopters is directed to the possibility that compliance with or adoption of OMG specifications may require use of an invention covered by patent rights. OMG shall not be responsible for identifying patents for which a license may be required by any OMG specification, or for conducting legal inquiries into the legal validity or scope of those patents that are brought to its attention. OMG specifications are prospective and advisory only. Prospective users are responsible for protecting themselves against liability for infringement of patents.

IPR Mode

This specification is issued under the Non-Assert Mode based on the OMG IPR Policy. Each Obligated Party in a Non-Assert Mode Adoption or Revision Process must provide the Non-Assertion Covenant found in Appendix A of the OMG IPR Policy: https://www.omg.org/cgi-bin/doc.cgi?ipr

GENERAL USE RESTRICTIONS

Any unauthorized use of this specification may violate copyright laws, trademark laws, and communications regulations and statutes. This document contains information which is protected by copyright. All Rights Reserved. No part of this work covered by copyright herein may be reproduced or used in any form or by any means--graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems--without permission of the copyright owner.

DISCLAIMER OF WARRANTY

WHILE THIS PUBLICATION IS BELIEVED TO BE ACCURATE, IT IS PROVIDED "AS IS" AND MAY CONTAIN ERRORS OR MISPRINTS. THE OBJECT MANAGEMENT GROUP AND THE COMPANIES LISTED ABOVE MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS PUBLICATION, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF TITLE OR OWNERSHIP, IMPLIED WARRANTY OF MERCHANTABILITY OR WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE OR USE. IN NO EVENT SHALL THE OBJECT MANAGEMENT GROUP OR ANY OF THE COMPANIES LISTED ABOVE BE LIABLE FOR ERRORS CONTAINED HEREIN OR FOR DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL, RELIANCE OR COVER DAMAGES, INCLUDING LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY ANY USER OR ANY THIRD PARTY IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The entire risk as to the quality and performance of software developed using this specification is borne by you. This disclaimer of warranty constitutes an essential part of the license granted to you to use this specification.

RESTRICTED RIGHTS LEGEND

Use, duplication or disclosure by the U.S. Government is subject to the restrictions set forth in subparagraph (c) (1) (ii) of The Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013 or in subparagraph (c)(1) and (2) of the Commercial Computer Software - Restricted Rights clauses at 48 C.F.R. 52.227-19 or as specified in 48 C.F.R. 227-7202-2 of the DoD F.A.R. Supplement and its successors, or as specified in 48 C.F.R. 12.212 of the Federal Acquisition Regulations and its successors, as applicable. The specification copyright owners are as indicated above and may be contacted through the Object Management Group, 109 Highland Avenue, Needham, MA 02494, U.S.A.

TRADEMARKS

CORBA®, CORBA logos®, FIBO®, Financial Industry Business Ontology®, FINANCIAL INSTRUMENT GLOBAL IDENTIFIER®, IIOP®, IMM®, Model Driven Architecture®, MDA®, Object Management Group®, OMG®, OMG Logo®, SoaML®, SOAML®, SysML®, UAF®, Unified Modeling Language®, UML®, UML Cube Logo®, VSIPL®, and XMI® are registered trademarks of the Object Management Group, Inc.

For a complete list of trademarks, see: https://www.omg.org/legal/tm\_list.htm. All other products or company names mentioned are used for identification purposes only, and may be trademarks of their respective owners.

COMPLIANCE

The copyright holders listed above acknowledge that the Object Management Group (acting itself or through its designees) is and shall at all times be the sole entity that may authorize developers, suppliers and sellers of computer software to use certification marks, trademarks or other special designations to indicate compliance with these materials.

Software developed under the terms of this license may claim compliance or conformance with this specification if and only if the software compliance is of a nature fully matching the applicable compliance points as stated in the specification. Software developed only partially matching the applicable compliance points may claim only that the software was based on this specification, but may not claim compliance or conformance with this specification. In the event that testing suites are implemented or approved by Object Management Group, Inc., software developed using this specification may claim compliance or conformance with the specification only if the software satisfactorily completes the testing suites.

OMG's Issue Reporting Procedure

All OMG specifications are subject to continuous review and improvement. As part of this process we encourage readers to report any ambiguities, inconsistencies, or inaccuracies they may find by completing the Issue Reporting Form listed on the main web page https://www.omg.org, under Documents, Report a Bug/Issue.

Table of Contents

[CubeSat System Reference Model 1](#_Toc81174716)

[Preface 7](#_Toc81174717)

[0 Section 8](#_Toc81174718)

[0.1 Specification 8](#_Toc81174719)

[0.2 Guide to Material 8](#_Toc81174720)

[0.3 Proof of Concept 8](#_Toc81174721)

[0.4 RFP General Requirements Satisfaction 10](#_Toc81174722)

[0.5 RFP Specific Requirements Satisfaction 13](#_Toc81174723)

[0.6 Issues to be Discussed. Section 6.7 of CSRM RFP 25](#_Toc81174724)

[1 Scope 26](#_Toc81174725)

[2 Conformance 26](#_Toc81174726)

[3 References 26](#_Toc81174727)

[3.1 Normative References 26](#_Toc81174728)

[3.2 Non-normative References 26](#_Toc81174729)

[4 Terms and definitions 27](#_Toc81174730)

[5 Symbols and Abbreviations 28](#_Toc81174731)

[6 Additional Information 28](#_Toc81174732)

[6.1 Changes to Adopted OMG Specifications 28](#_Toc81174733)

[6.2 Acknowledgments 28](#_Toc81174734)

[7 CSRM Profile (Normative) 29](#_Toc81174735)

[7.1 Concerns and Requirements Profile 29](#_Toc81174736)

[7.1.1 StakeholderConcern 30](#_Toc81174737)

[7.1.2 MissionObjective 30](#_Toc81174738)

[7.1.3 MissionNeed 31](#_Toc81174739)

[7.1.4 MissionConstraint 31](#_Toc81174740)

[7.1.5 MissionRequirement 31](#_Toc81174741)

[7.1.6 CubeSatRequirement 31](#_Toc81174742)

[7.1.7 GroundSegmentRequirement 31](#_Toc81174743)

[7.1.8 SubsystemRequirement 32](#_Toc81174744)

[7.1.9 ComponentRequirement 32](#_Toc81174745)

[7.1.10 DeployerRequirement 32](#_Toc81174746)

[7.1.11 SpaceSegmentRequirement 32](#_Toc81174747)

[7.1.12 Group 32](#_Toc81174748)

[7.1.13 SpacecraftRequirement 32](#_Toc81174749)

[7.1.14 SegmentRequirement 33](#_Toc81174750)

[7.1.15 SatelliteRequirement 33](#_Toc81174751)

[7.2 Technical Measures Profile 33](#_Toc81174752)

[7.2.1 moeSpecification 34](#_Toc81174753)

[7.2.2 mopSpecification 35](#_Toc81174754)

[7.2.3 tpmSpecification 35](#_Toc81174755)

[7.2.4 mop 35](#_Toc81174756)

[7.2.5 tpm 35](#_Toc81174757)

[7.2.6 kppSpecification 35](#_Toc81174758)

[7.2.7 kpp 35](#_Toc81174759)

[7.2.8 MeasurementSpecification 36](#_Toc81174760)

[7.2.9 mopRequirement 36](#_Toc81174761)

[7.2.10 tpmRequirement 36](#_Toc81174762)

[7.2.11 moeRequirement 36](#_Toc81174763)

[7.2.12 kppRequirement 36](#_Toc81174764)

[7.3 Architecture Structures Profile 37](#_Toc81174765)

[7.3.1 Component 37](#_Toc81174766)

[7.3.2 Segment 37](#_Toc81174767)

[7.3.3 CubeSat 38](#_Toc81174768)

[7.3.4 Facility 38](#_Toc81174769)

[7.3.5 Equipment 38](#_Toc81174770)

[7.3.6 Stakeholder 38](#_Toc81174771)

[7.3.7 GroundSegment 38](#_Toc81174772)

[7.3.8 SpaceSegment 39](#_Toc81174773)

[7.3.9 Spacecraft 39](#_Toc81174774)

[7.3.10 Satellite 39](#_Toc81174775)

[7.3.11 CubeSatDeployer 39](#_Toc81174776)

[7.3.12 Mission 39](#_Toc81174777)

[7.3.13 ConcernOf 39](#_Toc81174778)

[7.4 Validation and Verification Profile 40](#_Toc81174779)

[7.4.1 ValidationActivity 40](#_Toc81174780)

[7.4.2 Validation 41](#_Toc81174781)

[7.4.3 VerificationActivity 41](#_Toc81174782)

[7.4.4 Verification 41](#_Toc81174783)

[7.5 Comments Profile 42](#_Toc81174784)

[7.5.1 Explanation 42](#_Toc81174785)

[7.5.2 HowTo 42](#_Toc81174786)

[7.6 SysML Extensions 42](#_Toc81174787)

[7.6.1 performanceRequirement 43](#_Toc81174788)

[7.6.2 moe 43](#_Toc81174789)

[7.6.3 extRequirement 44](#_Toc81174790)

[7.6.4 System 44](#_Toc81174791)

[7.6.5 System context 44](#_Toc81174792)

[7.6.6 Subsystem 44](#_Toc81174793)

[7.6.7 Domain 44](#_Toc81174794)

[7.6.8 VerificationMethodKind 44](#_Toc81174795)

[7.6.9 VerificationMethodKind 44](#_Toc81174796)

Figures

[Figure 1 CSRM Profiles 29](#_Toc81158949)

[Figure 2 Concerns and Requirements Profile 30](#_Toc81158950)

[Figure 3 Technical Measures Profile 34](#_Toc81158951)

[Figure 4 Architecture Structures Profile 37](#_Toc81158952)

[Figure 5 Validation and Verification Profile 40](#_Toc81158953)

[Figure 6 Comments Profile 42](#_Toc81158954)

[Figure 7 SysML Extensions 43](#_Toc81158955)

Preface

OMG

Founded in 1989, the Object Management Group, Inc. (OMG) is an open membership, not-for-profit computer industry standards consortium that produces and maintains computer industry specifications for interoperable, portable, and reusable enterprise applications in distributed, heterogeneous environments. Membership includes Information Technology vendors, end users, government agencies, and academia.

OMG member companies write, adopt, and maintain its specifications following a mature, open process. OMG’s specifications implement the Model Driven Architecture® (MDA®), maximizing ROI through a full-lifecycle approach to enterprise integration that covers multiple operating systems, programming languages, middleware and networking infrastructures, and software development environments. OMG’s specifications include: UML® (Unified Modeling Language™); CORBA® (Common Object Request Broker Architecture); CWM™ (Common Warehouse Metamodel™); and industry-specific standards for dozens of vertical markets.

More information on the OMG is available at https://www.omg.org/.

OMG Specifications

As noted, OMG specifications address middleware, modeling and vertical domain frameworks. All OMG Specifications are available from the OMG website at:

https://www.omg.org/spec

All of OMG’s formal specifications may be downloaded without charge from our website. (Products implementing OMG specifications are available from individual suppliers.) Copies of specifications, available in PostScript and PDF format, may be obtained from the Specifications Catalog cited above or by contacting the Object Management Group, Inc. at:

OMG Headquarters
9C Medway Road, PMB 274
Milford, MA 01757
USA
Tel: +1-781-444-0404
Fax: +1-781-444-0320
Email: *pubs@omg.org*

Certain OMG specifications are also available as ISO standards. Please consult https://www.iso.org

# Section

This Section 0 contains mission-specific material. See section 4.9.2 of the driving RFP

## Specification

Full name of this submission: CubeSat System Reference Model (CSRM)

Submitter: David Kaslow, Co-chair International Council of Systems Engineering (INCOSE) Space Systems Working Group (SSWG)

OMG Document Number: **space/21-08-01**

OMG RFP: space/18-09-04

Adoption of this specification does not require change to existing OMG specifications

## Guide to Material

An overview of the background, goals and technical content of this proposal is described in Chapter 1, “Scope” of this document.

## Proof of Concept

The CSRM Profile, via its use in a CubeSat systems reference model, has been evaluated by the following individuals, in the following environments and shown to be technically viable.

**Aerospace Corporation**

The model was reviewed with favorable results by an Aerospace Corp team lead by:

Ryan Noguchi. Director, Space Architecture Department

Robert Stevens. Director of Model Based Systems Engineering Office

Marilee Wheaton. Systems Engineering Fellow

**Raytheon Certified Architect Program**

John Bloomer, PhD. Engineering Fellow - Raytheon Space and Airborne Systems

John carried out a review of the CSRM with favorable results

**United States Naval Academy**

John Gregory of the United States Naval Academy, in collaboration with Dr. Jin Kang, Associate Professor of Aerospace Engineering and head of the USNA Small Satellite Program, is leading a project to create an MBSE representation of their PSAT1U CubeSat. The model will use the CSRM as a starting point. The model will define all of the important subsystems, components, and interfaces, but leave the definition of the payload and mission specific modules to mission-specific student teams. The PSAT1U model will be used in the class room for the year-long Capstone design course and for future CubeSat missions.

**Georgia Institute of Technology (Georgia Tech)**

Russell Peak, PhD. MBSE Branch Chief and Senior Researcher in the Aerospace Systems Design Lab

Vishnu Rajendran **-**Student

Patrick Miga **-**Student

The CSRM was evaluated by Russell Peak and students at Georgia Tech by seeing how well it represents a mission currently under development. A report was issued with recommendations and favourable results.

**Orbital Transports**

The CSRM was evaluated with favorable results by a team at Orbital Transports led by David Hurst. The model was applied to CubeSat mission development in the context of several CubeSat missions the company worked on.
The company is developing a comprehensive MBSE approach for delivering small satellite missions based in part on the CSRM.

**SE DSIG**

I presented the CSRM at the OMG September Technical Meeting, to the Systems Engineer Domain Special Interest Group (SE DSIG) chaired by Sandy Friedenthal. Sandy’s comment: “I think this serves as an excellent precedent for future reference architectures and reusable libraries.”

**University of Bristol**

Lucy Berthoud, PhD. Senior Teaching Fellow

Kathryn Pugh, Student

Santeri Hotakainen, Student

The CSRM was evaluated in three research papers with favourable results:

“Model-Based Systems Engineering for University of Bristol Satellite”

“Modelling the Memory Requirements of a CubeSat mission using a Model Based Systems Engineering (MBSE) Method”

“Developing an Executable Model of the UOBSAT Using Model-Based Systems Engineering Methods”

**NASA Langley Research Center (LaRC)**

Samantha Infeld, PhD

Deputy Lead, NASA MBSE Community of Practice

Model-Based Systems Engineering Infusion and Modernization Initiative (MIAMI)

NASA Langley Research Center Engineering Design Studio

Kurt Woodham

Systems Engineering and Engineering Methods Branch

LaRC has initiated project “Model-Based Systems Engineering for Rapid CubeSat Development.” This project includes leveraging the CSRM in the augmentation of their CubeSat rapid prototyping environment.

**University of Alaska Fairbanks**

Denise Thorsen, PhD

Electrical and Computer Engineering

College of Engineering and Mines

Professor Denise Thorsen is the principle investigator for the CubeSat science mission: “Communication Testbed for On-orbit Verification and Validation of Communication Protocols aimed at Maximizing Information Throughput”. This mission is part of Air Force Research Laboratory (AFRL) University Nanosat Program (UNP).

She has been successful applying the CSRM to the CubeSat project. She has generated requirements and design material in support of UNP reviews.

## RFP General Requirements Satisfaction

| Id | Para | Text | Documentation |
| --- | --- | --- | --- |
| RfpGeneralRqt\_1 | 5 | General Requirements on Proposals | Section title |
| RfpGeneralRqt\_2 | 5.1 | Requirements | Section title |
| RfpGeneralRqt\_3 | 5.1.1  | Use of modeling languagesSubmitters are encouraged to express models using OMG modeling languages ... Submissions containing models expressed using OMG modeling languages shall be accompanied by an OMG XMI [XMI]  | UML and SysML are used. |
| RfpGeneralRqt\_4 | 5.1.2  | PIMs & PSMsSection 6 of this RFP specifies whether PIM(s), PSM(s), or both are being solicited.  | A Platform Independent Model (PIM) model will be provided.This specification does not explicitly state that PIM is being sought. However, requirements 6.5.1.2 and 6.5.1.10 state that tool independent format is required.  |
| RfpGeneralRqt\_5 | 5.1.3 | Complete submissionsProposals shall be precise and functionally complete. Any relevant assumptions and context necessary to implement the specification shall be provided | The evaluations carried out as listed in section 0.2 Proof of Concept confirm the completeness of the CSRM space-ground specification as presented in section 7.  |
| RfpGeneralRqt\_6 | 5.1.4  | ReuseProposals shall reuse existing OMG and other standard specifications in preference to defining new models to specify similar functionality. | The CSRM Profile reuses SysML. The profile may be reusable within the satellite domain. The CSRM is a template that is unique within the CubeSat domain and there is no opportunity for reuse. |
| RfpGeneralRqt\_7 | 5.1.5  | Changes to existing specificationsEach proposal shall justify and fully specify any changes or extensions to existing OMG specifications necessitated by adopting that proposal. In general, OMG favors proposals that are upwards compatible with existing standards and that minimize changes and extensions to existing specifications | The CSRM Profile does not require changes or extensions to existing specifications. |
| RfpGeneralRqt\_8 | 5.1.6  | MinimalismProposals shall factor out functionality that could be used in different contexts and specify their models, interfaces, etc. separately. Such minimalism fosters re-use and avoids functional duplication. | The CSRM Profile has been separated out of the CSRM such that designs create from a CSRM template have commonality and that tools can take advantage of such commonality.  |
| RfpGeneralRqt\_9 | 5.1.7  | IndependenceProposals shall use or depend on other specifications only where it is actually necessary. While re-use of existing specifications to avoid duplication will be encouraged, proposals should avoid gratuitous use. | The CSRM Profile does not use or depend on other specifications other than the OMG UML, XMI and SysML specifications.  |
| RfpGeneralRqt\_10 | 5.1.8  | CompatibilityProposals shall be compatible with and usable with existing specifications from OMG and other standards bodies, as appropriate. Separate specifications offering distinct functionality should be usable together where it makes sense to do so. | CSRM profile extends SysML so it compatible with that specification. |
| RfpGeneralRqt\_11 | 5.1.9  | Implementation flexibilityProposals shall preserve maximum implementation flexibility. Implementation descriptions should not be included and proposals shall not constrain implementations any more than is necessary to promote interoperability. | An XMI file that is free of implementation specific elements is being provided.  |
| RfpGeneralRqt\_12 | 5.1.10  | EncapsulationProposals shall allow independent implementations that are substitutable and interoperable. An implementation should be replaceable by an alternative implementation without requiring changes to any client. | An XMI file will be provided. The user can implement as desired. |
| RfpGeneralRqt\_13 | 5.1.11 | SecurityIn order to demonstrate that the specification proposed in response to this RFP can be made secure in environments that require security, answers to the following questions shall be provided: ... | There is nothing in the CSRM Profile that requires or interferes with a secure environment. |
| RfpGeneralRqt\_14 | 5.1.12  | InternationalizationProposals shall specify the degree of internationalization support that they provide. The degrees of support are as follows: ... | There is nothing in the CSRM that is specific to an international region |
| RfpGeneralRqt\_15 | 5.2  | Evaluation criteriaAlthough the OMG adopts model-based specifications and not implementations of those specifications, the technical viability of implementations will be taken into account during the evaluation process. The following criteria will be used: | Agreed |
| RfpGeneralRqt\_16 | 5.2.1  | PerformancePotential implementation trade-offs for performance will be considered.  | The CSRM has no known performance issues. It is not an executable model. |
| RfpGeneralRqt\_17 | 5.2.2  | PortabilityThe ease of implementation on a variety of systems and software platforms will be considered. | An XMI file is provided along with supporting material |
| RfpGeneralRqt\_18 | 5.2.3 | SecurabilityThe answer to questions in section 5.1.11 shall be taken into consideration to ascertain that an implementation of the proposal is securable in an environment requiring security. | There is nothing in the CSRM that requires or interferes with a secure environment. |
| RfpGeneralRqt\_19 | 5.2.4 | Conformance: Inspectability and TestabilityThe adequacy of proposed specifications for the purposes of conformance inspection and testing will be considered. Specifications should provide sufficient constraints on interfaces and implementation characteristics to ensure that conformance can be unambiguously assessed through both manual inspection and automated testing. | Sections 6.1 Problem Statement and 6.2 Scope of Proposal Sought state that the CSRM Profile is intended as a foundation and guidance for the development of a CubeSat space and ground system. It is not intended to be used in assessing conformance of a CubeSat system architecture and design.  |
| RfpGeneralRqt\_20 | 5.2.5 | Standardized MetadataWhere proposals incorporate metadata specifications, OMG standard XMI metadata [XMI] representations should be provided. | The CSRM Profile is a standardization of the Metadata for the CSRM. |

## RFP Specific Requirements Satisfaction

| Id | RFP | Text | Satisfied By | Documentation |
| --- | --- | --- | --- | --- |
| RfpCsrmRqt\_1 |   | CubeSat System Reference Model (CSRM)Request For ProposalOMG Document: space/2018-09-04 |   | RFP title. Satisfied By not applicable. |
| RfpCsrmRqt\_2 | 6 | Specific Requirements on Proposals |   | Section title. Satisfied By not applicable |
| RfpCsrmRqt\_3 | 6.1 | Problem Statement A CubeSat is a low-cost standardized satellite. The basic CubeSat unit is 10x10x10 centimeters with a mass of about 1.3 kilograms. CubeSat units can be joined to form larger satellites. They are typically launched as secondary payloads or deployed from the International Space Station. CubeSats originated in 1999 in the university community as a means to enable the design, construction, and launch of satellites using primarily off-the-shelf components. Recently the worldwide community has adopted the CubeSat standard as a low-cost means of performing technology demonstration, scientific, commercial and government missions. The dramatic decrease in cost for delivering small payloads to orbit has caused an explosion in the number of CubeSats and the number of organizations developing CubeSats. A SysML compliant and tool-independent CubeSat template model that provides building blocks that can be specialized to support MBSE CubeSat design will lower the cost of development and increase the quality of CubeSat spacecraft and ground system design. |   | Contextual only. Satisfied By not applicable |
| RfpCsrmRqt\_4 | 6.1 continued | Problem Statement continued A CubeSat System Reference Model (CSRM) will provide the logical architecture elements of a CubeSat space and ground system. The CSRM logical elements 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. On the other hand, should the mission-specific team decide to adopt different logical architecture elements, the CSRM will be sufficiently flexible to accommodate these types of changes. The CSRM specifically will address the needs of the CubeSat community. However, because the basic architecture partitioning and functional allocation of most satellites are similar, the CSRM could be applied not only for CubeSat development but for satellite missions in general.  |   |   |
| RfpCsrmRqt\_5 | 6.1 continued | Problem Statement continuedThis RFP solicits proposals for the standardization of CubeSat development through the use of a SysML based CSRM that enables the following: • Decreased cost of development and increased quality of CubeSat spacecraft and ground system design utilizing proven design templates • Traceability between model design elements, requirements and regulatory constraints at all architectural levels • Logical Context for the design and building of a specific CubeSat  • Improved communications across all stakeholders though fit for purpose diagrams and textual views • Foundation for validation and verification of CubeSat design • Foundation for continuous evaluation of CubeSat design • Facilitate training for both MBSE and the CubeSat domain • Foundation and context for future integration with other external analysis tools  • Guidance and instruction for applying MBSE principles and methods consistently in a CubeSat domain  • Capture and management of institutional knowledge |   | Contextual only. Satisfied By not applicable |
| RfpCsrmRqt\_6 | 6.1 continued | Problem Statement continuedBoth start-up and mature satellite development organizations can benefit from leveraging an effective common model structure and framework to support increased production without jeopardizing successful deployment and operation. The framework should allow organizations to extend the knowledge base with component libraries from their own mission-unique subsystems and experience, enhancing internal reuse and transfer of institutional knowledge to successive programs.Many of these organizations are university programs that combine aerospace engineering instruction with fundamental research while developing, launching, and operating a spacecraft. With a planned turnover of most of the engineering staff within a short period of time, these educational and research organizations need a common engineering framework and knowledge base that stores the institutional knowledge acquired by previous space missions, so that incoming personnel can quickly contribute to the engineering tasks and mission success. |   | Contextual only. Satisfied By not applicable |
| RfpCsrmRqt\_7 | 6.2 | Scope of Proposals SoughtThis RFP solicits proposals for a CSRM based on the System Modeling Language [SysML] to facilitate the development of a mission specific CubeSat system. A CubeSat System Reference Model (CSRM), developed in SysML using a standard systems engineering language, can reduce the large learning curve for space mission systems design in the following areas: language (SysML), methods, tools, domain knowledge, regulations, physical constraints, costs, and schedules. The CSRM will be reusable and extensible to a variety of specific space missions and payloads. By reducing the learning curve to apply Model-Based Systems Engineering (MBSE), organizations will more easily avoid unstructured methods and data that impede reuse and analysis.Space mission-procuring organizations will also use the CSRM to capture mission requirements in a standard format that is readily accessible and understandable by the potential vendors/developers. Receiving a proposed system in a standard format would also allow easier evaluation of proposals from a technical and cost feasibility perspective. |   | Contextual only. Satisfied By not applicable |
| RfpCsrmRqt\_8 | 6.3 | Relationship to other OMG Specifications and Activities |   | Section title. Satisfied By not applicable |
| RfpCsrmRqt\_9 | 6.3.1 | Relationship to OMG specificationsThe CSRM shall conform to the OMG System Modelling Language (SysML®) [SYSML] specification, version 1.5, https://www.omg.org/spec/SysML.  |   | Agreed - by construction |
| RfpCsrmRqt\_10 | 6.3.2 | Relationship to other OMG Documents and work in progressThe Precise Semantics family of specifications published and in progress at the OMG could support greater in-model validation of system specifications and therefore are recommended for incorporation or accommodation in a proposed CSRM, but are not mandatory: |  | Not a requirement. See 6.7 below. |
| RfpCsrmRqt\_11 | 6.4 | Related non-OMG Activities, Documents and Standards The physical characteristics of a CubeSat are governed by California Polytechnic State University in: • CubeSat Design Specification, Revision 13, https://www.cubesat.org/resources/The National Aeronautics and Space Administration provides a basic concepts and processes guide including regulatory guidance for CubeSats in its CubeSat Launch Initiative program: • CubeSat 101, Revision Dated October 2017, https://www.nasa.gov/directorates/heo/home/CubeSats\_initiativeThe International Telecommunications Union provides a guide to filing procedures for small satellites on its web site. Filing is done through the national administration with authority over the CubeSat developer, rather than directly with the ITU: • ITU Filing Procedures for Small Satellites, https://www.itu.int/ |  |  Not a requirement.  |
| RfpCsrmRqt\_12 | 6.5 | Mandatory Requirements |   | Section title. Satisfied By not applicable |
| RfpCsrmRqt\_13 | 6.5.1 | General Requirements |   | Section title. Satisfied By not applicable |
| RfpCsrmRqt\_14 | 6.5.1.1  | Proposals shall specify a CubeSat System Reference Model (CSRM) in SysML. |   | The CSRM Profile is provided to support the CSRM. The CSRM is provided externally as non-normative reference as an example of a use of the profile. |
| RfpCsrmRqt\_15 | 6.5.1.2 | The CSRM shall be modeling tool independent. |  CSRM Profile.xmi | The XMI file for the CSRM Profile will satisfy this requirement |
| RfpCsrmRqt\_16 | 6.5.1.3 | The CSRM shall contain a glossary of terminology. |  | The glossary is not a part of this specification as it is intended to be changes and extended and thus not readily standardized.  |
| RfpCsrmRqt\_17 | 6.5.1.4 | The CSRM shall include model element organization overview to support navigation and model usage. | Examples in CSRM:1 - Stakeholder pkg5 - Architecture pkg4 - Requirements pkg2 - Technical Measures pkg3 - Use Cases pkg7 - Validation Verification pkg | Not Implemented The referenced CSRM organizes packages and table are ordered by a numerical prefix. The CSRM is only referenced by this specification, so this requirement no longer applies. |
| RfpCsrmRqt\_18 | 6.5.1.5  | The CSRM elements shall support requirements analysis, allocation, traceability, validation, and verification. | See: Validation and Verification Profile |  These concepts are supported by the profile |
| RfpCsrmRqt\_19 | 6.5.1.6  | The CSRM definition library shall contain and use the concepts as defined in the CSRM glossary.  | 9 - CubeSat Reference Information pkg |  Not Implemented |
| RfpCsrmRqt\_20 | 6.5.1.7  | The CSRM shall provide a description for every CSRM specific model element. | 9 - CubeSat Reference Information pkg | The CSRM Profile is fully documented. The CSRM which is not a part of the standard is partially documented and utilizes a glossary to aid in understanding where elements do not have detailed documentation. |
| RfpCsrmRqt\_21 | 6.5.1.8  | The CSRM shall allow for the modification and extension of model elements and responsibilities to support mission objectives. | Examples in CSRM:004 - CSRM Application sc pkg |  The CSRM Profile is intended to be extended. |
| RfpCsrmRqt\_22 | 6.5.1.9  | The CSRM shall provide basic instructions on how to use the CSRM for creation of the mission specific CubeSat |   | Not Implemented because the CSRM is not included as a part of the specification,. However the CSRM is publicly available and provided this functionality. The CSRM package diagrams have Note elements that are descriptive (contextual) and prescriptive (how to populate and apply, Need to determine how to present this information in tool agnostic manner.  |
| RfpCsrmRqt\_23 | 6.5.1.10 | The CSRM shall be provided in tool-independent formats that capture the content and visualization of the model in the form of: - A navigable document, preferably HTML - The associated XMI file  |   | CSRM Profile is published as XMI.  |
| RfpCsrmRqt\_24 | 6.5.2  | CSRM Logical Architecture Elements |   | Section title. Satisfied By not applicable |
| RfpCsrmRqt\_25 | 6.5.2.1 | The CSRM logical architecture shall consist of the following architectural components: CubeSat domain, CubeSat enterprise, space segment, ground segment.  | Examples in CSRM:L0\_Domain pkgL3\_Subsystems pkgL2\_Segment pkgL1\_Enterprise pkg5 - Architecture pkg |  Not Implemented. The referenced CSRM provides an example of the implementation of the requirement. |
| RfpCsrmRqt\_26 | 6.5.2.2  | The architectural components of the CSRM shall be extended to at least one additional hierarchical level to describe CubeSat subsystems and ground subsystems | Examples in CSRM:L4\_Components pkg |  Not Implemented. The referenced CSRM provides an example of the implementation of the requirement. |
| RfpCsrmRqt\_27 | 6.5.2.3  | The architectural levels of the CSRM shall be extensible by the template user to additional levels based on mission complexity. |  Not Applicable | Not Implemented in specification. The referenced non-normative CSRM provides an example implementation of the requirement. |
| RfpCsrmRqt\_28 | 6.5.2.4 | Each architectural level of the CSRM shall include requirements, behaviors, and structures supporting common system Use Cases. |  Not Applicable | Not Implemented. Package structure not defined in the profile. See the non-normative example of the CSRM. |
| RfpCsrmRqt\_29 | 6.5.2.5 | The CSRM model elements shall accommodate Ground Station services and Transport, Launch, and Deploy services being provided to the CubeSat enterprise. | Architecture Structures Profile | Supported by the Profile in the Architecture Structures Profile.  |
| RfpCsrmRqt\_30 | 6.5.3  | CubeSat Mission Stakeholder Relationships | Architecture Structures Profile and Concerns and Requirements Profile | Provided by the CSRM profile via Stakeholder, Stakeholder Concern and ConcernOf stereotype. |
| RfpCsrmRqt\_31 | 6.5.3.1  | The CSRM model elements shall support the identification of CubeSat mission stakeholders, their needs, and traceability. Provide utility for easy addition of stakeholders, and their needs. | Architecture Structures Profile and Concerns and Requirements Profile | Provided by the CSRM profile. |
| RfpCsrmRqt\_32 | 6.5.3.2 | The CSRM architecture shall include a library of regulatory stakeholders allowing the CSRM user to select the regulatory authorities relevant to the specific CubeSat mission.  | Not Applicable | Due to fact that regulatory stakeholders vary by application and local regulations that change over time, such a library cannot be the subject of a standard, so this requirement is not implemented. |
| RfpCsrmRqt\_33 | 6.5.3.3  | The CSRM shall be extendable to accommodate other national stakeholders and associated regulations and guidelines.  | CSRM Profile | CSRM is extensible. |
| RfpCsrmRqt\_34 | 6.5.3.4  | The CSRM shall provide well defined viewpoints and views for addressing stakeholder needs and objectives. |  Concerns and Requirements Profile | Profile supports these concepts with the elements described in the Concerns and Requirements Profile |
| RfpCsrmRqt\_35 | 6.5.4  | Mission Specific Model Validation and Performance Budgeting | Validation and Verification Profile | Supported by the Validation and Verification Profile. |
| RfpCsrmRqt\_36 | 6.5.4.1  | The CSRM model elements shall support the identification of technical measures that provide insight into the progress made in the definition and development of the CubeSat solution. These measures include, but are not limited to, Measures of Effectiveness (MOEs), Measures of Performance (MOPs), and Technical Performance Measures (TPMs). | Technical Measures Profile | Supported by the Technical Measures Profile |
| RfpCsrmRqt\_37 | 6.5.4.2 | The CSRM shall assist with the validation of compliance to the various regulatory agencies and their associated regulations and guidelines. |  Technical Measures Profile | Supported by the Technical Measures Profile |
| RfpCsrmRqt\_38 | 6.5.4.3 | The CSRM shall provide power, cost, and mass roll-up capabilities starting at the CubeSat component level. |  Not Applicable | Not Implemented.Support of roll-ups is both supported and often applied according to the needs/design and is inappropriate for standardization. |
| RfpCsrmRqt\_39 | 6.6 | Non-mandatory features - The CSRM proposal may provide the native CSRM model file as a convenience for CSRM maintenance - The CSRM proposal may provide a CubeSat ontology to establish a common terminology for the CubeSat community and enable model validation and extraction tooling.  |  Not Applicable | It has been determined that it is not appropriate nor beneficial to provide a CSRM ontology.  |
| RfpCsrmRqt\_40 | 6.7 | Issues to be DiscussedThe current and work-in-progress specifications for Precise Semantics in UML modelling are relevant to the validation of models developed using the CSRM. Proposer should discuss the methods for validation, traceability, and performance budget allocation/verification in the proposed CSRM. |  See section 0.4 and 1.1 | Not a requirement - no shall statement. However, see section 0.4 in the CSRM specification |
| RfpCsrmRqt\_41 | 6.8  | Evaluation Criteria - Proposals must include at least one sample CubeSat mission model built on proposed CSRM.  - Proposals will be evaluated on the degree of compliance with the mandatory requirements. |  Not Applicable | Because we are not delivering a reference model in this response and only a profile, providing an implementation is out of scope.  |
| RfpCsrmRqt\_42 | 6.9  | Other information unique to this RFPThe CSRM will be a SysML compliant and platform-independent template model that provides building blocks that can be specialized to support MBSE CubeSat design. Unlike typical language specifications such as UML or SysML, the CSRM is instead a model library specification.  | Not Applicable | For the many reasons listed in prior requirements, it is impossible to provide a model library. However, the profile supports the creation of compatible CubeSat libraries through the application of the profile. |
| RfpCsrmRqt\_43 | 6.1 | IPR ModeEvery OMG Member that makes any written Submission in response to this RFP shall provide the Non-Assertion Covenant found in Appendix A of the OMG IPR Policy [IPR]. |   | IPR is Non-Assert Mode |

## Issues to be Discussed. Section 6.7 of CSRM RFP

From section 6.7:

The current and work-in-progress specifications for Precise Semantics in UML modelling are relevant to the validation of models developed using the CSRM. Proposer should discuss the methods for validation, traceability, and performance budget allocation/verification in the proposed CSRM.

Proposer Comment:

The CSRM Profile is intended to be specification for enforcing compliance and thus interoperability of logical elements of a CubeSat system.

The CSRM Profile provides stereotypes used categorize the logical architecture of CubeSat space and ground system.

The CSRM Profile is a partial response to the original RFP. Due to the nature of the reference model as a template for a CubeSat System that varies depending on different design standards and context-specific content (government entities and local regulation), the submitter chose to provide only the profile. Different versions of the reference model is provided only as an example (first version focused on United States stakeholders and regulatory entities). The CSRM templates will be managed as an OMG Open Collaboration Project for the benefit of the community and vendors.

# Scope

The CubeSat System Reference Model Profile (CSRM Profile) defines the necessary stereotypes (extends SysML) for logical CubeSat space-ground architectures. The profile is intended to promote consistency and interoperability of logical CubeSat space/ground architectures models. The profile is intended to be used as-is or extended.

The CSRM Profile is composed of six sub-profiles

* **Architecture Structures Profile** - Extends the architectural structures of SysML to include common structure types (extensions of Block) of a CubeSat system.
* **Comments Profile** – Extends the Comment element to facilitate building of reference models with tool and instructional information.
* **Concerns and Requirements Profile** - Extends SysML with new requirement types to accommodate stakeholder needs, missions, and categorized requirements for the structural levels of specification of CubeSats and ground systems.
* **SysML Extensions** - Additional stereotypes similar to and slightly modified from the SysML non-normative extensions.
* **Technical Measures Profile** – Extensions of SysML to enhance specification and categorization of technical measures.
* **Validation and Verification Profile** – Extension and enhancement to SysML to facilitate validation (correctness) and verification (testing) of requirements.

An example use of the CSRM Profile, is provided as a non-normative example of a CubeSat System Reference Model template available at: https://github.com/ObjectManagementGroup/CSRM. The example represents a template and starting elements focused on the United States government and its regulatory environment. Although the example is for the United States, the CSRM profile is government agnostic. Our goal is to add additional examples of the CSRM template with other regulatory context in the future (like the European Union and its European Space Agency).

This release of the CSRM Profile contains the stereotypes and some constraints. It is expected that future versions will extend the profile with additional constraints and stereotypes to aid in consistency and coverage of CubeSat space/ground domains.

# Conformance

Conformant implementations must implement the CSRM Profile in its entirety. Ideally, the Profile should be usable "as-is" however implementers may extend the profile as needed.

**space/21-08-04** - CSRM Profile - XMI File (see the description in Section 7)

# References

## Normative References

The following normative documents contain provisions which, through reference in this text, constitute provisions of this specification. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply.

* Object Management Group, “Unified Modeling Language, version 2.5.1,” http://www.omg.org/spec/ UML/2.5.1, December 2017.
* Object Management Group, “Systems Modeling Language, version 1.6,” http://www.omg.org/spec/ SysML/1.6, November 2019.
* Object Management Group, “XML Metadata Interchange, version 2.5.1,” https://www.omg.org/spec/XMI/2.5.1, November 2019.

## Non-normative References

The following references were used to inform the contents of the CSRM Profile:

* CubeSat Systems Reference Model website: https://github.com/ObjectManagementGroup/CSRM. The referenced model uses a version of the CSRM Profile.
* INCOSE Systems Engineering Handbook, 4th ed., INCOSE-TP-2003-002-04 2015.
* NASA Systems Engineering Handbook, rev. 1, December 2007, NASA/SP-2007-6105 Rev1.
* S. Friedenthal, A. Moore, R. Steiner, A Practical Guide to SysML, 3nd ed., Elsevier, Waltham, MA, 2015.
* G. Roedler and C. Jones, Technical Measurement, ver. 1, INCOSE-TP-2003-020-01, December 2005.
* W. Larson, et. al., Applied Space Systems Engineering, (Space Technology Series), McGraw Hill, Boston, MA, 2009.
* 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.
* Systems Engineering Book of Knowledge https://www.sebokwiki.org/wiki/Guide\_to\_the\_Systems\_Engineering\_Body\_of\_Knowledge\_(SEBoK)
* ANSI/AIAA G-043A-2012, “Guide to the Preparation of Operational Concept Documents”
* “Operational Concepts and the Case for Use Cases: Unifying UML with Systems Engineering”, Ray Jorgensen, 2002.
* “Reference Model for Service Oriented Architecture 1.0, OASIS Standard, 12 October 2006.”
* “Reference Architecture Foundation for Service Oriented Architecture Version 1.0, 4 Dec. 2012.”
* “Systems and software engineering - Architecture description” ISO/IEC/IEEE 42010:2011.
* “Systems Engineering Fundamentals,” Defense Acquisition University Press, 2001.
* 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.
* D. Kaslow, B. Ayres, P. Cahill, 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.
* "Glossary of Defense Acquisition Acronyms and Terms", Defense Acquisition University.
* https://www.dau.mil/glossary/Pages/Default.aspx

# Terms and definitions

The following are acronyms, references and terms that are used in the CSRM.

For a detailed understanding of the CubeSat domain and related terms, see the non-normative references.

Table 2 Acronyms

| Term | Description |
| --- | --- |
| ConOps | Concept of Operations |
| CSRM | CubeSat System Reference Model |
| KPP | Key Performance Parameter |
| MOE | Measure of Effectiveness |
| MOP | Measure of Performance |
| RFP | Request for Proposal |
| SysML | Systems Modeling Language |
| TPM | Technical Performance Measure |
| UML | Unified Modeling Language |
| XMI | XML Metadata Interchange |

# Symbols and Abbreviations

The specification uses the same Symbology of UML.

# Additional Information

## Changes to Adopted OMG Specifications

No changes are required.

## Acknowledgments

The CubeSat System Reference Model is an International Council on Systems Engineering (INCOSE) Space Systems Working Group (SSWG) product.

Daniel Brookshier, Dassault Systemes.

# CSRM Profile (Normative)

The CSRM Profile contains the sub-profiles of the CubeSat System Reference Model (CSRM) Profile. The purpose of extensions is to add metadata, aid in dynamic reporting/analysis, validate a model's composition, and aid in grouping.



Figure 1 CSRM Profiles

The CSRM Profile diagram documents the sub-profiles contained in the CSRM Profile.

## Concerns and Requirements Profile

The Concerns and Requirements Profile contains extensions to add requirement types common to CubeSat Systems and a replacement of SysML's «Stakeholder» to improve that traceability of Stakeholders to concerns.

The intent of this design to extend the requirement types of SysML is to support a top down analysis from the stakeholders high-level requirements through each level of detail in the architecture.



Figure 2 Concerns and Requirements Profile

The Concerns and Requirements Profile contains extensions to better classify requirements for CubeSat systems.

### StakeholderConcern

Description

«StakeholderConcern» is an interest in a system relevant to one or more of its stakeholders. A stakeholder concern could be manifest in many forms, such as one or more stakeholder needs, goals, expectations, responsibilities, requirements, design constraints, assumptions, dependencies, quality attributes, architecture decisions, risks, or other issues.

A Stakeholder can have multiple «StakeholderConcern»s, and a single «StakeholderConcern» can be shared by numerous «Stakeholder»s. The «ConcernOf» dependency relates «Stakeholder» to corresponding «StakeholderConcern».

«StakeholderConcern» are high-level requirements that specific requirements are derived.

The 'Text' attribute of «AbstractRequirement» is used to document the stakeholder concern.

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### MissionObjective

Description

«MissionObjective» is one of a broad set of goals that must be achieved to successfully satisfy the stated mission need, such as the purpose to be achieved, product to be produced, or a service to be performed.

«MissionObjective» are derived from «MissionNeed» or «StakeholderConcern». A «MissionObjective» is usually not satisfied directly but one or more «MissionRequirement» are created (derived) to translate to requirements that are then satisfied by a part of the design.

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### MissionNeed

Description

«MissionNeed» is a concise description of a need or service that the system must provide. It should be solution-independent and only describe the problem the system is supposed to solve. The mission need is the main driver of the architecture.

«MissionNeed» is satisfied by one or more «Mission» or by one or more of its sub-elements. «MissionObjective» are derived from «MissionNeed». «MissionConstraint» can be used as a derived constraint of the «MissionNeed» that limits the scope of the «MissionNeed».

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### MissionConstraint

Description

«MissionConstraint» is a limitation placed on cost, schedule, or implementation techniques available to the system designer. It is typically fixed and not subject to trades, e.g., mission budget and schedule.

«MissionConstraint» can be satisfied by any element that is a subject to the constraint. Alternatively, «PerformanceRequirement» or specialization (like «moeRequirement») can be created and derived from the «MissionConstraint» and then the «PerformanceRequirement» is then satisfied by a corresponding value property.

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### MissionRequirement

Description

«MissionRequirement» is a statement of facts and assumptions that define expectations on the system's capabilities in terms of mission objectives, environment, constraints, and measures of effectiveness (MoE). «MissionRequirement» are satisfied by «Mission» or one or more of its sub-elements and or derived further into «SpaceSegmentRequirement» and «GroundSegmentRequirement». «MissionRequirement» are derived from «MissionObjective» or directly derived from mission needs/constraints or stakeholder concerns.

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### CubeSatRequirement

Description

«CubeSatRequirement» is a requirement specific to a «CubeSat» needed to design and operate the «CubeSat» or its parts. «CubeSatRequirement» is a kind of Requirement on the CubeSat system satisfied by the «CubeSat» or by one or more of its sub-elements.

Generalization:

* CSRM Profile::Concerns and Requirements Profile::SatelliteRequirement

### GroundSegmentRequirement

Description

«GroundSegmentRequirement» is a requirement specific to a «GroundSegment» needed to design and operate the «GroundSegment» or its parts. «GroundSegmentRequirement» is a requirement of satisfied by a «GroundSegment» or by one or more of its sub-elements.

Generalization:

* CSRM Profile::Concerns and Requirements Profile::SegmentRequirement

### SubsystemRequirement

Description

«SubsystemRequirement» is a requirement specific to a «Subsystem» needed to design and operate a «Subsystem» or its parts. «SubsystemRequirement» is a requirement satisfied by a «Subsystem» or by one or more of its sub-elements.

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### ComponentRequirement

Description

«ComponentRequirement» is a requirement specific to a «Component» needed to design and operate the «component» or its parts. «ComponentRequirement» is a requirement Satisfied by a «Component» or by one or more of its sub-elements.

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### DeployerRequirement

Description

«DeployerRequirement» is a requirement specific to a «CubeSatDeployer» needed to design and operate the «CubeSatDeployer» or its parts. «DeployerRequirement» is a requirement satisfied by a «CubeSatDeployer» or by one or more of its sub-elements.

Note that the Deployer in this case is referring to the Deployer mechanism, not an organization.

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### SpaceSegmentRequirement

Description

«SpaceSegmentRequirement» is a requirement specific to a «SpaceSegment» needed to design elements of the «SpaceSegment» and operate the «SpaceSegment» parts. «SpaceSegmentRequirement» is a requirement satisfied by a «SpaceSegment» or one or more composed-elements.

Generalization:

* CSRM Profile::Concerns and Requirements Profile::SegmentRequirement

### Group

Description

«Group» is used to group related requirements into a containment hierarchy. It facilitates hierarchical numbering of requirements by modeling tools. The intent of «Group» is used like a package that primarily contains elements of «AbstractRequirement» kind.

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### SpacecraftRequirement

Description

«SpacecraftRequirement» is a system requirement needed to design and operate a spacecraft and/or its subsystems. «SpacecraftRequirement»is a requirement satisfied by one or more kind of «Spacecraft».

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### SegmentRequirement

Description

«SegmentRequirement» is an abstract requirement that is satisfied by a «Segment» or its parts. Note that «SegmentRequirement» is a base type for «SpaceSegmentRequirement» and «GroundSegmentRequirement» and it is expected to only be used when creating a specific kind of «SegmentRequirement».

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### SatelliteRequirement

Description

«SatelliteRequirement» is a requirement of a satellite system needed to design and operate a satellite and/or its subsystems. «SatelliteRequirement» is satisfied by a «Satellite» or one or more of its sub-elements. «SatelliteRequirement» is derived from «SpaceSegmentRequirement» or directly derived from mission needs/objectives/constraints or stakeholder concerns.

Generalization:

* CSRM Profile::Concerns and Requirements Profile::SpacecraftRequirement

## Technical Measures Profile

The Technical Measures Profile adds extensions to the SysML profile for KPP, TPM, and MoE. The profile also adds «MeasurementSpecification» and its specializations which are used to define and describe measures.



Figure 3 Technical Measures Profile

The Technical Measures Profile diagram shows extensions of the SysML profile for identifying value property types KPP, MoP, TMP, and MoE with corresponding requirements and measurement specifications.

### moeSpecification

Description

The Measure of Effectiveness (MoE) Specification «moeSpecification» specifies attributes of a system that determine how well the system element is satisfying or expected to satisfy technical requirements. «moeSpecification» is a «Block» that specifies a technical measure, constraints, and measurement activities used to provide insight into the progress made in the definition and development of the technical solution, risks and issues. «moeSpecification» may refine «MissionNeed».

The «moeSpecification» captures Stakeholder descriptions of operational measures of success that are satisfied by «moe» properties.

Generalization:

* CSRM Profile::Technical Measures Profile::MeasurementSpecification

### mopSpecification

Description

The Measure of Performance (MoP) Specification «mopSpecification» specifies attributes of a system that determine how well the system element is satisfying or expected to satisfy technical requirements. «mopSpecification» is a «Block» that specifies a technical measure, constraints, and measurement activities used to provide insight into the progress made in the definition and development of the technical solution, risks and issues.

The «mopSpecification» captures Stakeholder descriptions of physical or functional attributes relating to system operation that are to be transformed into one or more MoPs.

«mopSpecification» may refine MoE or KPP Specifications.

Generalization:

* CSRM Profile::Technical Measures Profile::MeasurementSpecification

### tpmSpecification

Description

The Technical Performance Measure (TPM) Specification specifies attributes of a system that determine how well the system element is satisfying or expected to satisfy technical requirements. «tpmSpecification» is a «Block» that specifies a technical measure, constraints, and measurement activities used to provide insight into the progress made in the definition and development of the technical solution, risks and issues. «tpmSpecification» refine «tpmRequirement».

«tpmSpecification» may refine «mopSpecification» refine.

Generalization:

* CSRM Profile::Technical Measures Profile::MeasurementSpecification

### mop

Description

The «mop» stereotype represents a performance property of a of a system (or other element). The «mop» stereotype is applied to a «ValueProperty» element to mark the property as a Measure of performance (MoP). MoP is an engineering performance measure that provides a value necessary for meeting a Measure of Effectiveness (MoE). «mop» satisfy «mopRequirement».

### tpm

Description

The «tpm» stereotype represents a performance property of a of a system (or other element). Technical Performance Measure (TPM) of the attributes of a system element to determine how well the system element is satisfying or expected to satisfy, specified technical requirements. They are based on the driving requirements or technical parameters of high risk or significance - e.g., mass, power, or data rate. Actual versus planned progress of TPMs are tracked so the systems engineer or project manager can assess progress and the risk associated with each TPM.

### kppSpecification

Description

Key Performance Parameter (KPP) represents a performance property of a of a system (or other element). KPPs are a critical subset of Technical Measures representing the most critical capabilities and characteristics. The «kpp» stereotype is applied to a «ValueProperty» of a «Block». «kpp» Satisfy «kppRequirtement».

Generalization:

* CSRM Profile::Technical Measures Profile::MeasurementSpecification

### kpp

Description

KPP Specification represents the requirement of a Key Performance Parameter (KPP) of the system to be measured. KPPs are a critical subset of Technical Measures representing the most critical capabilities and characteristics.

The «kpp» stereotype applied to a value property of a Block.

### MeasurementSpecification

Description

Measurement Specification is a «Block» that specifies a technical measure, constraints, and measurement activities used to provide insight into the progress made in the definition and development of the technical solution, risks and issues.

The «MeasurementSpecification» is an abstract stereotype that is the base for defining specific measurement specifications. This is an extension of «Block» to allow the full capability of a «Block» to be used in the specification of a measurement.

A MeasurementSpecification are intended to be Validated by a ValidationActivity.

Generalization:

* SysML::Blocks::Block

### mopRequirement

Description

A Measure of Performance (MoP) Requirement («mopRequirement») is a requirement specific to a «mop» property that specifies performance criteria for the element. «mopRequirement» is a type of Performance Requirement («performanceRequirement») that is satisfied by a «mop» property of a «Block».

In a «refine», the «mopSpecification» is the supplier property and the client property is a «mopRequirement». In a «satisfy», the «mopRequirement» is the supplier property and the client property is a «mop».

Generalization:

* CSRM Profile::SysML Extensions::performanceRequirement

### tpmRequirement

Description

A Technical Performance Measure (TPM) Requirement («tpmRequirement») is a requirement specific to a «mop» property that specifies performance criteria for the element. «mopRequirement» is a type of Performance Requirement («performanceRequirement») that is satisfied by a «tpm» property of a «Block».

In a «refine», the «tpmSpecification» is the supplier property and the client property is a «tpmRequirement». In a «satisfy», the «tpmRequirement» is the supplier property, and the client property is a «tpm».

Generalization:

* CSRM Profile::SysML Extensions::performanceRequirement

### moeRequirement

Description

A Measure of Performance (MoE) Requirement («moeRequirement») is a requirement specific to a «moe» property that specifies performance criteria for the element. «moeRequirement» is a type of Performance Requirement («performanceRequirement») that is satisfied by a «moe» property of a «Block».

In a «refine», the «moeSpecification» is the supplier property, and the client property is a «moeRequirement». In a «satisfy», the «moeRequirement» is the supplier property, and the client property is a «moe».

Generalization:

* CSRM Profile::SysML Extensions::performanceRequirement

### kppRequirement

Description

A Key Performance (KPP) Requirement («kppRequirement») is a requirement specific to a «kpp» property that specifies performance criteria for the property. «kppRequirement» is a type of Performance Requirement («performanceRequirement») that is satisfied by a KPP property of a «Block».

In a «refine», the «kppSpecification» is the supplier property, and the client property is a «kppRequirement». In a «satisfy», the «kppRequirement» is the supplier property, and the client property is a «kpp».

Generalization:

* CSRM Profile::SysML Extensions::performanceRequirement

## Architecture Structures Profile

This profile extends the architectural structures of SysML to include common structure types (extensions of Block) of a CubeSat system.



Figure 4 Architecture Structures Profile

The Architecture Structures Profile diagram shows the structural extensions added to the SysML profile used to enhance SysML for CubeSat design.

### Component

Description

The «Component» is a type of «Block» that is a part of a subsystem element such as a hardware component, software, or procedures. «Component» satisfies «ComponentRequirement».

Generalization:

* SysML::Blocks::Block

### Segment

Description

A «Segment» is one of the parts into which something naturally separates or is divided. The «Segment» is abstract and intended to be implemented as a specific kind of segment like «GroundSegment» and «SpaceSegment».

Generalization:

* CSRM Profile::SysML Extensions::System context

### CubeSat

Description

The «CubeSat» is a type of «Satellite». A CubeSat follows the CubeSat form-factor established in 1999 by California Polytechnic State University and Stanford University.

«CubeSat» is a part of a «SpaceSegment» and satisfies one or more «CubeSatRequirement».

Generalization:

* CSRM Profile::Architecture Structures Profile::Satellite

### Facility

Description

A «Facility» is a place, amenity, or piece of equipment provided for a particular purpose.

A «Facility» may be a part of another «Facility». Generally «Facility» is part of a «GroundSegment».

Generalization:

* SysML::Blocks::Block

### Equipment

Description

«Equipment» is a type of «Block» used to represent tools, machines, or other items required for a particular job or activity. Generally «Equipment» are part of a «Facility».

Generalization:

* SysML::Blocks::Block

### Stakeholder

Description

A «Stakeholder» is a «Block» representing any entity (individual or organization) that has an interest in the system. Typical stakeholders include users, operators, decision-makers, parties to the agreement, regulatory bodies, developing agencies, support organizations, and society at large. Stakeholders can also represent an entity in opposition or threat to the system.

Stakeholder is a replacement of the SysML Stakeholder and should be used in its place.

Note: Because of the incompatibility between the SysML «Stakeholder» and the CSRM «Stakeholder», «ConcernOf» should be used to relate «ViewPoint» to «Stakeholder» property of the «ViewPoint».

Generalization:

* SysML::Blocks::Block

### GroundSegment

Description

The «GroundSegment» is a kind of «Segment» composed of kinds of block like «Facility», «System» and «Equipment» or other «GroundSegment». The composed parts of «GroundSegment» represent systems and facilities like the following:

* Ground stations, which provide communication interfaces with spacecraft (in the «SpaceSegment»)
* Mission operations, from which spacecraft are managed, including activities like mission planning and scheduling, command and control of satellites, control of the ground equipment, mission telemetry processing, and mission data processing and distribution
* Ground networks, which connect the other ground elements to one another
* Remote terminals, used by support personnel
* Spacecraft integration and test facilities
* Launch facilities

«GroundSegment» is a kind of «Segment» and usually satisfies one or more «GroundSegmentRequirement».

 Generalization:

* CSRM Profile::Architecture Structures Profile::Segment

### SpaceSegment

Description

The «SpaceSegment» is a «Segment» that is composed of types of «spacecraft». In addition the «SpaceSegment» may be also be composed of «blocks» representing orbits, the uplink/downlink and the space environment (radiation, atmospheric density, solar wind, etc.). The «SpaceSegment» is controlled by and communicates with the «GroundSegment». The «SpaceSegment» satisfies one or more «SpaceSegment» requirements.

Generalization:

* CSRM Profile::Architecture Structures Profile::Segment

### Spacecraft

Description

Spacecraft is a kind of System Block representing a spacecraft system—for example, satellite, rocket, rocket stage, interplanetary vehicle, or space station.

Generalization:

* CSRM Profile::SysML Extensions::System

### Satellite

Description

A «Satellite» is a kind of «Spacecraft» representing an orbital satellite system.

Generalization:

* CSRM Profile::Architecture Structures Profile::Spacecraft

### CubeSatDeployer

Description

«CubeSatDeployer» is a type of «System» used to deploy one or more «CubeSat».

Generalization:

* CSRM Profile::SysML Extensions::System

### Mission

Description

A «Mission» describes what the system will do and the purpose of doing it. The Mission provides the context for defining measures of effectiveness and for the development of the Concept of Operations.

A mission is accomplished by operational nodes completing one or more operational activities. An operational node can be an organization, individual(s), or system(s). Operational activities are actions that either transform one or more inputs into outputs or change the state of the system. A system provides capabilities through the execution of operational activities. There may be one or more missions composed of Space and Ground segments. Missions may also be composed of one or more missions (sub-missions).

 Generalization:

* SysML::Blocks::Block

### ConcernOf

Description

The «ConcernOf» stereotype is a dependency relationship used to relate an element of the model to a «Stakeholder». The client of the dependency can be any element and the supplier must be a type of «Stakeholder» that has a concern. The documentation of the dependency can be used to document the concern(s) the stakeholder has about the element.

Another client of «ConcernOf» is the «StakeholderConcern». «StakeholderConcern» is a specialized requirement that can be used to document a stakeholder concern for use in creating derived requirements from the concern.

## Validation and Verification Profile

The Validations and Verification Profile has extensions of SysML for validation and verification of requirements.



Figure 5 Validation and Verification Profile

The Validation and Verification profile diagram shows the extensions of the SysML profile.

The profile introduces «VerificationActivity», a process (Behavior) for verifying a requirement. The «Verification» relationship maps the «VerificationActivity» to a requirement or measurement specification that it verifies. The «Verify» has the verified property to indicate the status of a measurement specification or requirement's verification status as a result of performing the related «VerificationActivity».

The profile also introduces «ValidaionActivity», a process (Behavior) for validating a requirement is correct. The «Validation» relationship maps the «ValidationActivity» to a requirement or measurement specification that it validates. The «Validation» has the validated property to indicate the status of a measurement specification or requirement's validation status as a result of performing the related «VerificationActivity».

In addition, the «VerificationActivity» has the property verificationMethod such that the VerificationMethodKind of a requirement can be aligned with the «VerificationActivity».

### ValidationActivity

Description

The «ValidationActivity» stereotype is applied to a process for validating requirements and technical specifications for correctness, implementability, testability, and that the requirement meets the needs of stakeholders. It often involves acceptance and suitability with customers. The requirements should also be annotated with a «rationale» comment. The activity to validate one or more requirements is often an analysis task performed in concert with the stakeholders to ensure the following:

* The set of requirements is correct, complete, and consistent,
* A model can be created that satisfies the requirements, and
* A real-world solution can be built and tested to prove that it satisfies the requirements.

The state of the analysis is recorded by the validated property of «Validation».

The «ValidationActivity» has a metaclass of Behavior, which allows the modeler to choose the appropriate type of modeling element (Activity, Sequence, etc.) to document the process. Note that the documentation of the ValidationActivity can also be used to capture the process.

### Validation

Description

«Validation» is a relationship between a source element («AbstractRequirement» or technical specification) and a client element («ValidationActivity»).

The element has a boolean property, validated, which is used to document the status of the validation activity. The property has the following meaning:

* undefined(default): The element has not been validated.
* false: The supplier element has been validated and is considered invalid according to the «ValidationActivity».
* true: The supplier element has been validated and is deemed to be valid according to the «ValidationActivity».

Note that the requirement or technical specification is the driver (supplier) of the relationship. This means that if the supplier element changes, the state of validated property should be changed to unknown until the «ValidationActivity» is performed for the changed element. Changes to a «ValidationActivity» implies the need for re-running the «ValidationActivity» for each supplier but are based on the possibility of invalidating the validation state.

Generalization:

* SysML::Requirements::Trace

Operations:

* getValidates - The query getValidates() returns all the NamedElements that are suppliers ( "to" end of the concrete syntax ) of a «Validation» relationship whose client is the element input parameter, ref. This is a static query. Due to constraints, the getValidates() returns types of Abstract Requirement and Measurement Specification.

Parameters:

* in ref : NamedElement [1]
* return result : NamedElement [0..\*]

### VerificationActivity

Description

The «VerificationActivity» stereotype is intended to extend SysML to add a verificationMethod (from SysML) such that the Test Case or other behavior method can be annotated with the kind of verification. This improves the original SysML to allow a Requirement to have multiple verification methods by moving the method to the «VerificationActivity» rather than the Requirement.

«VerificationActivity» has the attribute verificationMethod which is a VerificationMethodKind. Users are encouraged to set the verificationMethod on «VerificationActivity» rather than on the Requirement as this allows a requirement to be verified by many verification method kinds.

### Verification

Description

«Verification» is a relationship between a type of «AbstractRequirement» (the client of the Abstraction relationship) and a «VerificationActivity» (the supplier of the Abstraction relationship). The relationship is to associate a prescribed Verification Activity that tests for the satisfaction of a requirement.

The «Verification» stereotype has a boolean property, verified, which is used to document the performance of the «VerificationActivity». The property has the following meaning:

* Unset(default): The supplier element has not been verified by the «VerificationActivity».
* false: The supplier element has failed the «VerificationActivity».
* true: The supplier element has passed the «VerificationActivity».

A change of the supplier or client elements should be followed by a change to the verified property to unknown. The verification property is also changed whenever the «VerificationActivity» is performed.

Note: The «Verification» should be used instead of the «verify» of SysML.

Generalization:

* SysML::Requirements::Verify

## Comments Profile

The Comments Profile adds comment types to aid in classifying comments when creating a reference model.



Figure 6 Comments Profile

The Comments diagram shows the CSRM extensions added to the SysML profile to add additional types of comment: «Explanations» and «HowTo».

### Explanation

Description

The «Explanation» stereotype is a type of «comment» used to contain explanatory text. This type of comment is used for documenting what model elements are to be created, why they are created or other tutorial information.

### HowTo

Description

The «HowTo» stereotype is a type of «comment» used to contain instructions on how to do a modeling tool task. For example, how elements of the model are created. These comments are usually tool-specific instructions.

## SysML Extensions

This package contains stereotypes that overcome the issues caused by the inability to use non-normative extensions from SysML. Note that documentation has been modified to be consistent to the CSRM profile.

The SysML Extensions profile contains the «moe» stereotype, extensions of «Block» («Domain», «Subsystem», «System Context», «System») and extension of «Requirement» («extRequirement») and two enumerations, VerificationMethodKind and RiskKind, that are used in the «extRequirement» to document verification method and risk respectively.



Figure 7 SysML Extensions

The SysML Extensions diagram documents the elements added that are used instead of the SysML non-normative extensions described by the SysML Specification.

### performanceRequirement

Description

Requirement that quantitatively measures the extent to which a system, or a system part, satisfies a required capability or condition.

Note: This stereotype should be used instead of the «performanceRequirement» in the SysML non-normative extension of SysML.

Generalization:

* CSRM Profile::SysML Extensions::extRequirement

### moe

Description

Measure of Effectiveness (MoE) represents a performance property of a system (or other element). The «moe» stereotype is applied to a «valueProperty» to mark the property as a MoE. The stereotype indicates an engineering performance measure. A «moe» property represents a parameter whose value is can be used to measure the effectiveness of a «Block». MoE are measures designed to correspond to the accomplishment of mission objectives and the achievement of desired results. MoE quantify the results to be obtained by a system.

Each «moe» should satisfy a kind of «moeRequirement».

Note: This stereotype should be used instead of the «moe» in the SysML non-normative extension of SysML.

### extRequirement

Description

The «extRequirement» is a mix-in stereotype that contains generally useful attributes for requirements.

Note: This stereotype should be used instead of the «extendedRequirement» in the SysML non-normative extension of SysML

Generalization:

* SysML::Requirements::Requirement

### System

Description

A System is an artificial artifact consisting of blocks that pursue a common goal that cannot be achieved by the system's individual elements. A block can be software, hardware, a person, or an arbitrary unit.

Note: This stereotype should be used instead of the «System» in the SysML non-normative extension of SysML.

Generalization:

* SysML::Blocks::Block

### System context

Description

A System context element is a virtual container that includes the entire system and its actors.

Note: This stereotype should be used instead of the «System context» in the SysML non-normative extension of SysML.

Generalization:

* SysML::Blocks::Block

### Subsystem

Description

A Subsystem is a - typically large - encapsulated block within a larger system.

Note: This stereotype should be used instead of the «Subsystem» in the SysML non-normative extension of SysML.

Generalization:

* SysML::Blocks::Block

### Domain

Description

A «Domain» block represents an entity, a concept, a location, or a person from the real-world domain.

Note: This stereotype should be used instead of the «Domain» in the SysML non-normative extension of SysML.

Generalization:

* SysML::Blocks::Block

### VerificationMethodKind

VerificationMethodKind is an enumeration consisting of the following enumeration literals:

* Analysis
* Demonstration
* Inspection
* Test

### RiskKind

RiskKind is an enumeration consisting of the following enumeration literals:

* High
* Medium
* Low