# Designing a Model-Based Systems Engineering (MBSE) Template for Satellites

Authors: David Walker, Dan Christensen, Brad Mercer, Gavin Payne

## A Overview

The goal of this project has been to digitally transform SDL systems engineering methods for scoping small satellite missions, developing system architectures, documenting the engineering solution, and verifying design solutions from being document-intensive (gathering non-transformable information in discrete, disjoint documents) to being model-based (employing cohesive, traceable, transformable graphical models).

We have produced a set of models that give satellite development programs (and any "model-based" program) a head start that enables the modeling effort to drive the systems engineering process instead of catching up with a document-driven design. We plan to use these models to develop, train, and encourage consistent modeling techniques across the Lab and lower the barrier of entry for creating focused, useful models.

## B What Makes an Ideal MBSE Template?

**1. A clean, lightweight starting point with easy access to common content for small satellite models:** The template itself has a minimal amount of suggested content that a systems engineer will need to find and refactor or delete in order to make all the content reflect the true nature of their system. Instead, all the common content can be easily found in a reference model and copied into the template to quickly populate an accurate model for a new application.

**2. Facilitates modeling everything that needs to be modeled and provides an intuitive place to put it:** The right set of custom profiles (extensions of a standard modeling language like SysML) have been defined that match SDL's systems engineering and modeling methodologies. All the modeling content that a systems engineer commonly has to create during the course of a project has an intuitive and clearly labeled package structure that it can be sorted into.

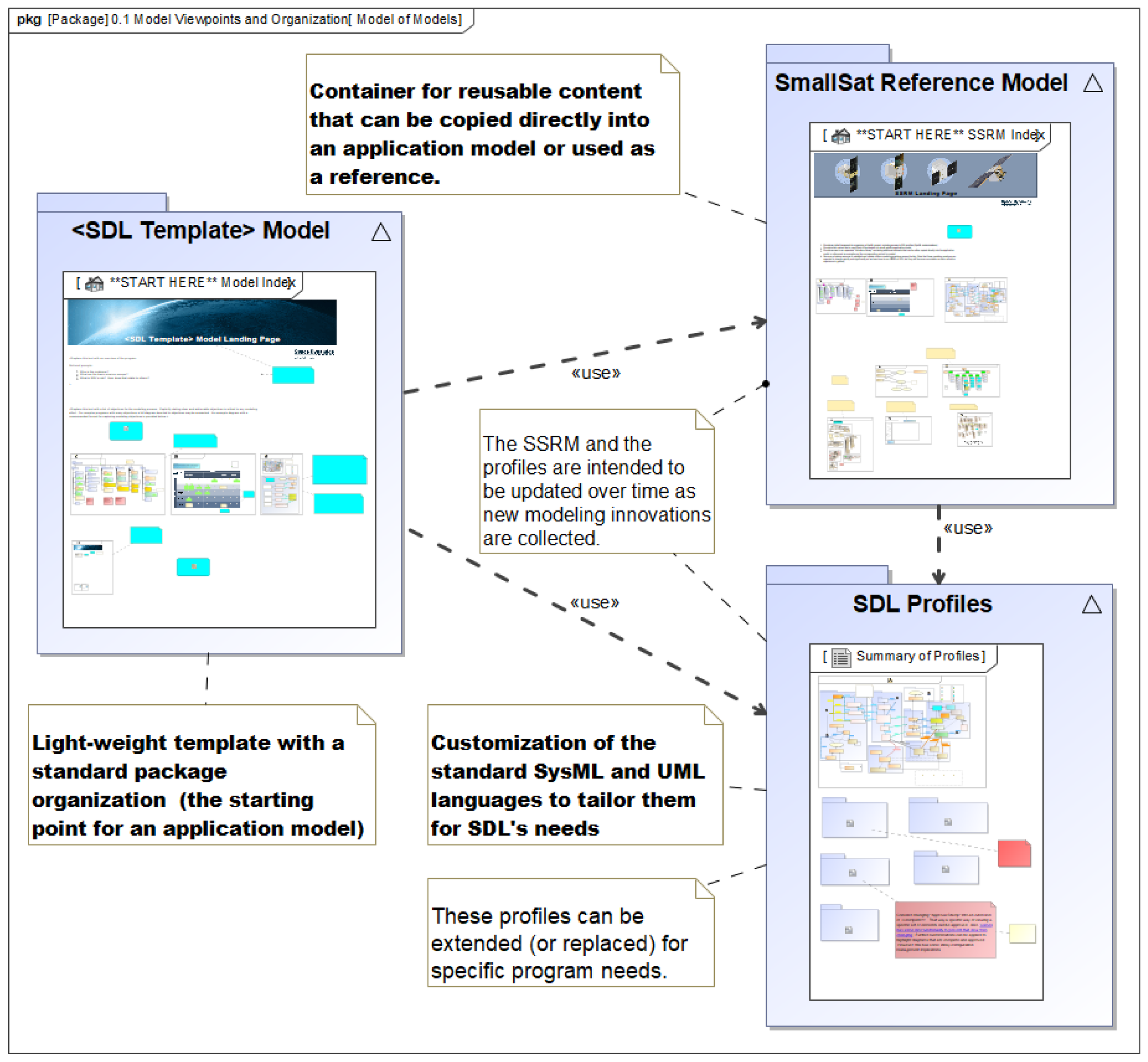
### B.1 What is "Common Content" for Small Satellite Models?

1. **Mission ("System of Systems") Structure:** Space, Ground, Launch Segments, and their user roles (operator, etc.)
2. **Subsystems in the Space, Ground, and Launch Segments**: Electrical Power, Comms, Command & Control, Propulsion, etc.
3. **General interfaces between segments and subsystems**
4. **Use Cases w/ functional breakdown:** Data Collection and Distro, Launch & Early Ops, Fault Detection & Recovery, etc.
5. **General Mission Stakeholders:** Customers, Regulatory Agencies, Launch Providers, etc.
6. **Measures of Effectiveness and analytical structures for computing them:** Mission Duration, Mission Cost, Data Resolution, Duty Cycle, Coverage, Response Time, etc.
7. **Ontologies for space domain and engineering terms**

## C References Used While Designing the Models

| Reference | How it was used |
|---|---|
| **CubeSat Reference Model**, Object Management Group, 2021 | A large portion of the technical content in the CSRM was incorporated into the SSRM |
| **Model Based Design with Systems Engineering Based on RFLP Using V6**, Kleiner, Kramer, 2013 CIRP Design Conf. | The Requirements → Functional → Logical → Physical (RFLP) approach was adopted in the definition of the modeling viewpoints |
| **Draft AFRL/AFIT model** (not publicly available for industry-wide use) | Leveraged Government perspective on mission requirements structure |
| **A Practical Guide to SysML 3rd Edition**, Friedenthal | Leveraged aspects of the Object-Oriented Systems Engineering Methodology (OOSEM) |
| **An MBSE Architectural Framework for Inter-Satellite Communication in a Multiorbit Disaggregated System**, Anyanhun, 2022 INCOSE Symposium | Lessons learned on alternatives to traditional spacecraft architecture for a multiorbit disaggregated system |
| **Magic Grid Book of Knowledge 2nd Edition**, Morkevičius | Influenced our methodology and arrangement of viewpoints |
| **Unified Architecture Framework**, Object Management Group | The Mission Viewpoint leverages the basic concept of UAF's Operational Viewpoint |

## D Introducing the Models




*Models Structure — The SDL Template is a ready-to-populate "empty" model with a standard layout for the package organization, content diagrams for easy navigation, and some pre-formatted tables to expose the content once it is populated. The common content for a satellite model is stored in the SmallSat Reference Model (SSRM) and can be transferred into the template for rapid population. The profiles project provides customized extensions to SysML that facilitate the structure of the content in the SSRM and the populated template model.*

## E Template Model Organization

| Viewpoints \ Aspects | Requirements | Behavior (ACT/STM/SD) | Structure (IBD & BDD) | Exchange Elements | Parameters | Unique Views |
|---|---|---|---|---|---|---|
| Metamodel | Model Stakeholder Analysis | 0.2 Systems Engineering Processes | <SSTM> Local Profile | <SSTM> Meta-Model | 0.4 Modeling Metrics | Model Organization; 0.3 Glossaries & Ontologies |
| Mission | 1.1 Stakeholders & Needs; 1.2 SSTM Mission Goals & Objectives | 1.3 Mission Behavior | 1.4 Mission Context | 1.5 Mission Exchange Elements | MOEs and KPPs | Mission Concept Review Presentation |
| Functional and Logical | L0 - System Requirements | 3.1 Functional Architecture | 3.2 Logical Architecture | 3.4 F&L Exchange Elements | 3.3 Functional & Logical Parametric Analysis | SRR Presentation |
| Physical | 2.0 Requirements | 4.1 Physical Sys Behavior | 4.2 System Structure | 4.3 Physical Exchange Elements | 4.4 Physical Sys Parametric Analysis | PDR Presentation; CDR Presentation |
| System Test | Testing/quality requirements | 5.1 V&V Workflows | 5.2 V&V Structure | 5.3 - V&V Exchange Elements | V&V Metrics | TRR Presentation; PSR Presentation |

*SDL's Architecture Framework — This grid organizes all the views that are needed by systems engineers as they scope, architect, document, and evaluate a system design. In general (but with many exceptions), progress in model development will go row by row from left to right.*

**Metamodel Viewpoint:** Auxiliary viewpoint for collecting information about the model itself and the modeling team.

**Mission Viewpoint:** System-of-systems viewpoint dedicated to the nature of the mission, independent from any material solutions. The analysis of the mission results in system-level requirements.

**Functional & Logical Viewpoint:** Contains a functional breakdown of the system and a synthesized logical structure to perform the functions. Results in a breakdown of the system requirements.

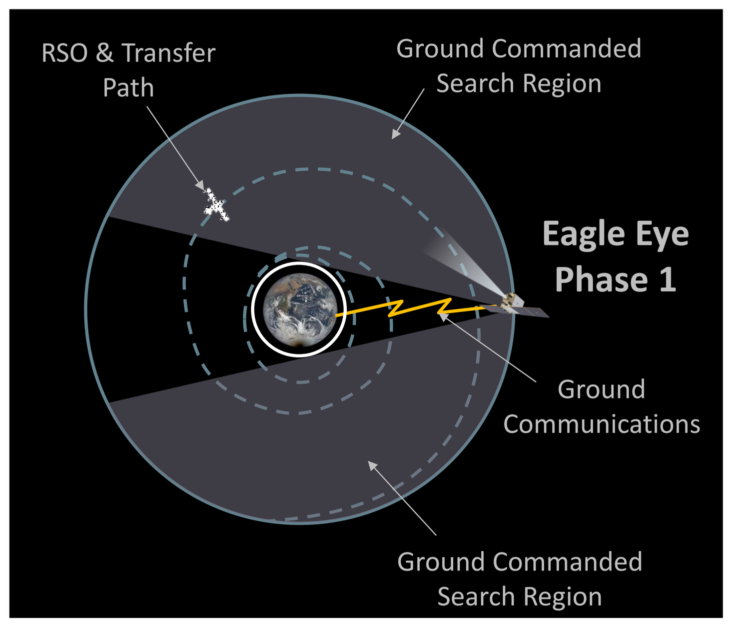
**Physical Viewpoint:** Documents hardware and software solutions and the necessary integration.

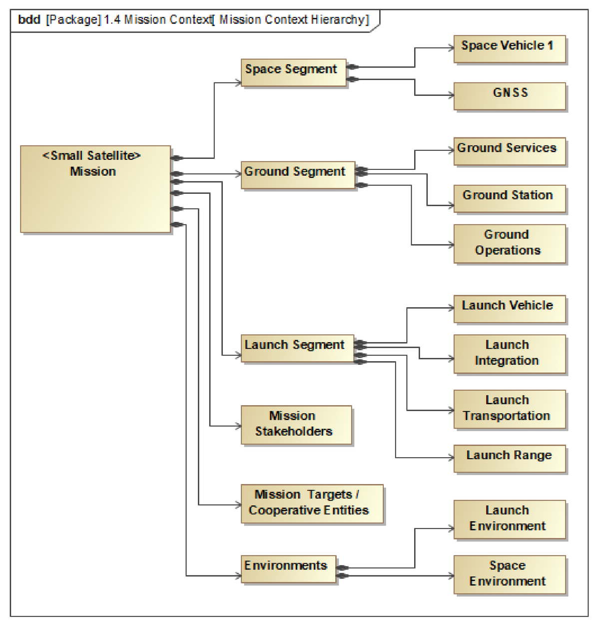
**System Test Viewpoint:** Plans for and summarizes results of system Verification and Validation.

Space Dynamics Laboratory
Utah State University

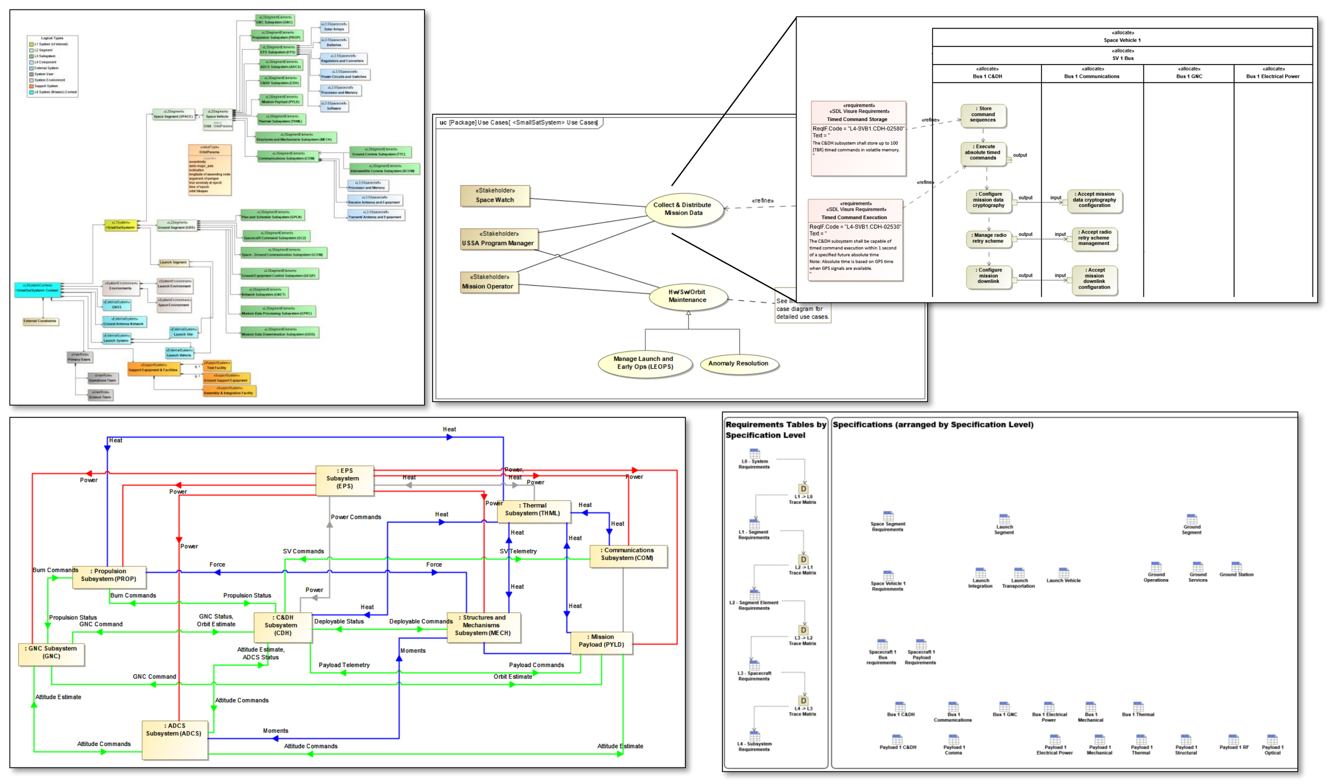
## F SmallSat Reference Model (SSRM)

The SSRM is a container for example and common content to be referred to and reused in the development of application models.



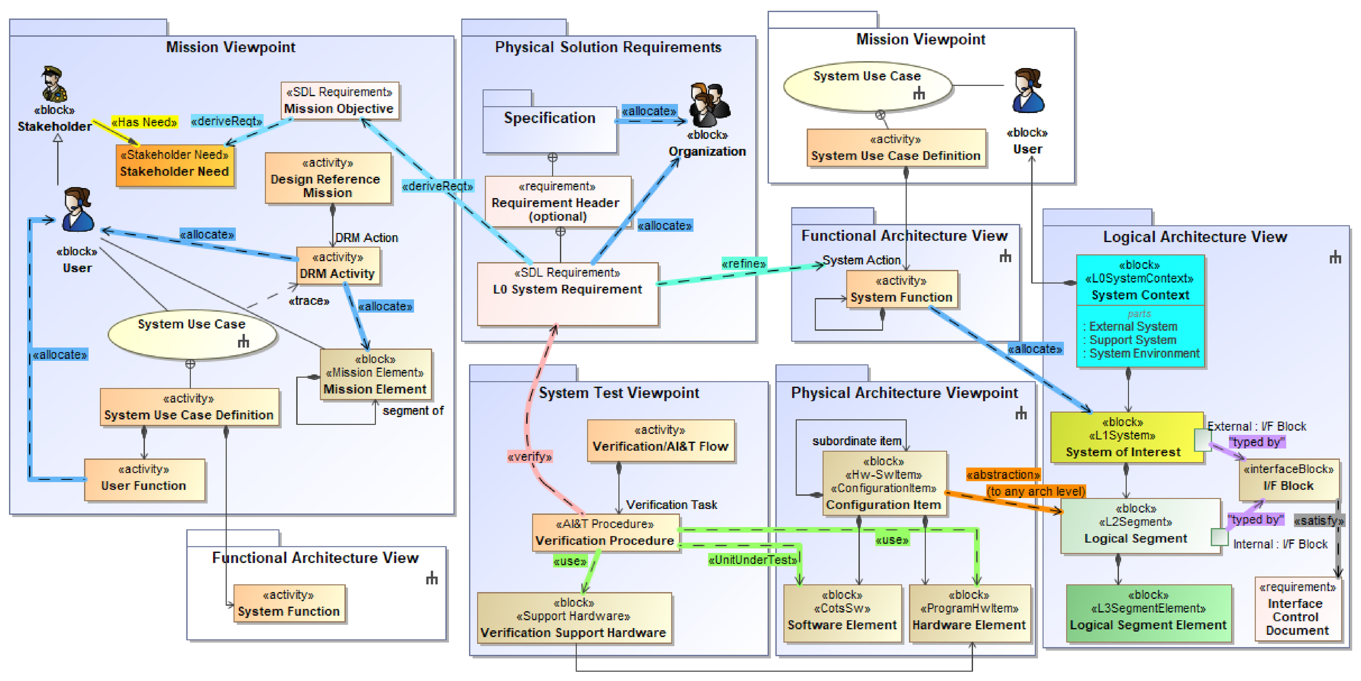




*Example Mission ConOps and Common Mission Content — Much of the content of the Mission Viewpoint of the model is highly dependent on the particular concept of operations (ConOps) for the system. For this reason, an example ConOps was developed along with example content for the SSRM (left). However, the general "system of systems" structure of the mission context (right) is common across many missions.*



*Sample Content — These diagrams illustrate the common content collected in the SSRM. In clockwise order from upper left, these represent (1) a logical hierarchical breakdown of a typical small satellite system, (2) typical use cases with (3) a multi-level functional decomposition to the sub-system level, (4) a standard set of requirement specifications, and (5) a block diagram showing typical interactions between satellite sub-systems.*

## G The Modeling Structure Enabled by the Profiles




*Structural Metamodel — This is a simplified structural metamodel (a model of the intended structure showing object types and relationship types) for the SSRM and template-application models. Each object represents what could be multiple objects of the same type in the application model. The packages correspond with viewpoints in the template (Section E).*

## H Next Steps

The template models are ready to be used on SDL's next model-based program. As repositories of common content and best modeling practices, the SSRM and the profiles will be continually expanded and refined as they are used and as feedback is received from users of the model. Plans are in place to produce training materials tailored to the organization and structure of the models.