Modeling to Mars: a NASA Model Based Systems Engineering Pathfinder Effort

Nipa Phojanamongkolkij, Ph.D.[1], Kristopher A. Lee[2], Scott T. Miller[3], Kenneth A. Vorndran[4], Karl R. Vaden[5], Eric P. Ross, PE[6], Bobby C. Powell[7], Robert W. Moses, Ph.D.[8]

National Aeronautics and Space Administration, USA

The NASA Engineering Safety Center (NESC) Systems Engineering (SE) Technical Discipline Team (TDT) initiated the Model Based Systems Engineering (MBSE) Pathfinder effort in FY16. The goals and objectives of the MBSE Pathfinder include developing and advancing MBSE capability across NASA, applying MBSE to real NASA issues, and capturing issues and opportunities surrounding MBSE. The Pathfinder effort consisted of four teams, with each team addressing a particular focus area. This paper focuses on Pathfinder team 1 with the focus area of architectures and mission campaigns. These efforts covered the timeframe of February 2016 through September 2016. The team was comprised of eight team members from seven NASA Centers (Glenn Research Center, Langley Research Center, Ames Research Center, Goddard Space Flight Center IV&V Facility, Johnson Space Center, Marshall Space Flight Center, and Stennis Space Center). Collectively, the team had varying levels of knowledge, skills and expertise in systems engineering and MBSE. The team applied their existing and newly acquired system modeling knowledge and expertise to develop modeling products for a campaign (Program) of crew and cargo missions (Projects) to establish a human presence on Mars utilizing In-Situ Resource Utilization (ISRU). Pathfinder team 1 developed a subset of modeling products that are required for a Program System Requirement Review (SRR)/System Design Review (SDR) and Project Mission Concept Review (MCR)/SRR as defined in NASA Procedural Requirements. Additionally, Team 1 was able to perform and demonstrate some trades and constraint analyses. At the end of these efforts, over twenty lessons learned and recommended next steps have been identified.

Nomenclature

ARC	=	Ames Research Center
GRC	=	Glenn Research Center
GSFC	=	Goddard Space Flight Center
ISRU	=	In-Situ Resource Utilization
JPL	=	Jet Propulsion Laboratory
JSC	=	Johnson Space Center
LaRC	=	Langley Research Center
MBSE	=	Model-Based Systems Engineering
MCR	=	Mission Concept Review
MSFC	=	Marshal Space Flight Center
NESC	=	NASA Engineering Safety Center
NPR	=	NASA Procedural Requirements

---

[1] NASA Langley Research Center, Hampton, VA.

[2] NASA Johnson Space Center, Houston, TX.

[3] NASA Ames Research Center, Moffett Field, CA.

[4] NASA Goddard Space Flight Center IV&V Facility, Fairmont, WV.

[5] NASA Glenn Research Center, Cleveland, OH.

[6] NASA Stennis Space Center, Stennis Space Center, MS.

[7] NASA Marshall Space Flight Center, Huntsville, AL.

[8] NASA Langley Research Center, Hampton, VA.

American Institute of Aeronautics and Astronautics

SDR = System Definition Review
SE = Systems Engineering
SRR = System Requirement Review
SSC = Stennis Space Center
SysML = Systems Modeling Language
TDT = Technical Discipline Team

I. Introduction

THE NESC SE TDT initiated the MBSE Pathfinder effort in FY16. The goals and objectives of the MBSE Pathfinder include developing and advancing MBSE capability across NASA, applying MBSE to real NASA issues, and capturing issues and opportunities surrounding MBSE. The Pathfinder effort consisted of four teams, with each team addressing a particular focus area using Systems Modeling Language (SysML). This paper focuses on Pathfinder team 1 with the focus area of architectures and mission campaigns. These efforts covered the timeframe of February 2016 through September 2016. The team was comprised of eight team members from seven NASA Centers (GRC, LaRC, ARC, GSFC IV&V Facility, JSC, MSFC, and SSC). Collectively, the team had varying levels of knowledge, skills and expertise in systems engineering and MBSE.

Figure 1 shows goals and objectives for Pathfinder team 1. There were four goals to meet the Pathfinder's requirements. Traced back to these goals, seven objectives were defined. The team applied their existing and newly acquired system modeling knowledge and expertise to work on all of these goals and objectives. The team develops modeling products for a campaign (Program) of crew and cargo missions (Projects) to establish a human presence on Mars utilizing ISRU. The campaign was based on and described in Ref. 1. At the campaign level, Ref. 1 studies three campaigns (Disposable ISRU, Light ISRU, and Extensive ISRU). Within any of these campaigns, there are multiple missions (e.g., initial cargo, recurring cargo #1, recurring cargo #2, crew, etc.) Within any of these missions, there are multiple launches. Within any launches, there are multiple payload elements. The campaign can be viewed as a NASA program consisting of multiple projects (missions). To prioritize the modeling efforts, Pathfinder team 1 chose to model the Extensive ISRU campaign option and the first two cargo missions to place ISRU on the Mars surface.
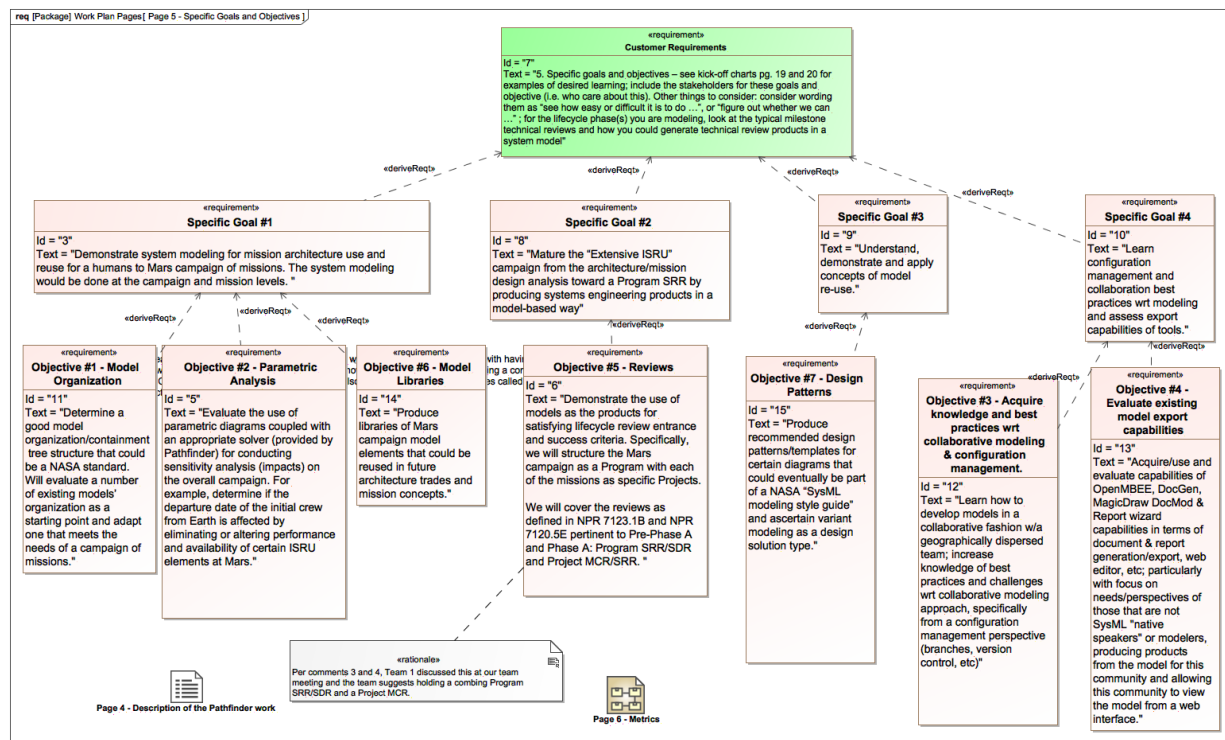

Figure 1. Pathfinder Goals and Objectives.

American Institute of Aeronautics and Astronautics

Organization of the paper is as follow. Section II describes accomplishments by goals. Lessons learned are given in Section III. Section IV provides conclusion and recommended next steps.

II. Accomplishments

Accomplishments are summarized in the next sub-sections by Pathfinder's goals and objectives.

A. Goal 1

Goal 1 is to demonstrate system modeling for mission architecture use and reuse for a human to Mars campaign of missions. The system modeling would be done at the campaign and mission levels. There are three objectives to fulfill this goal and are given in the next sub-sections.

1. Objective 1: Model Organization

Model organization is an important part of the approach to handling the system complexity and the modeling task. The high-level structure that paralleled the team's dual focus areas is depicted in Figure 2. The high-level structure allows the team to partition work related to the Pathfinder project management (referred to as *MBSE Pathfinder* in Figure 2) in a separate and parallel space from the Mars Campaign modeling work (*Mars Campaign*). *MBSE Pathfinder* package contains project work for this Pathfinder effort such as final report document generation modeling, lessons learned, presentation using model walk-through, etc.

Within the *Mars Campaign* package (the left-side of Figure 2), the *02. Mars Campaign* package contains elements that pertain to multiple missions to accomplish a campaign as well as elements that serve as patterns for individual missions. The *03. Mars Missions* package is where the team modeled individual missions for the overall campaign. Additionally, the *Glossary* package was created to provide definitions and explanations about the model elements to help the team understand the concepts included in the model. And finally, the *Model Libraries* package was created to organize information about the model which could be reused in multiple places. Figure 3 presents an overview of Mars campaign organization, with the same structure for both campaign and missions sub-packages.
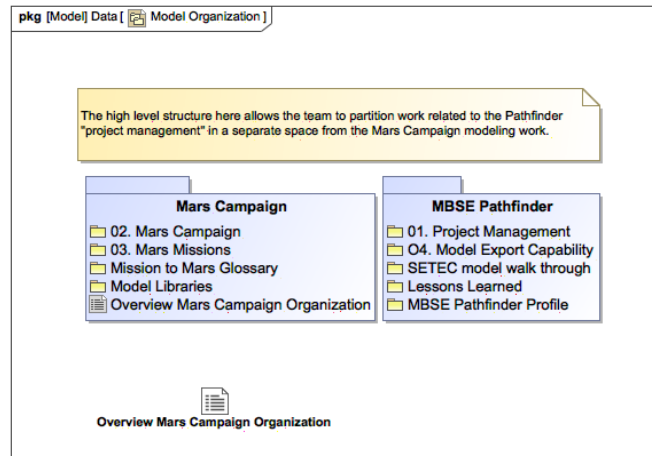

Figure 2. High-level model organization for Team 1.

Figure 3. Overview of Mars Campaign Organization.

2. Objective 2: Parametric Analysis

As discussed in Section I, the team chose to model the Extensive ISRU campaign option and the first two cargo missions (i.e., initial cargo and recurring cargo #1 missions.) Based on Ref. 1, the first two missions consist of four launches each. Specifically, the first four launches (Launches 01-04) are part of the initial cargo mission. Similarly, Launches 05-08 are part of the recurring cargo #1 mission. Figure 4 shows the modeling of the generic campaign as well as the customization of the generic campaign for the extensive ISRU campaign.

Figure 4. Customized Mars Campaign from the generic campaign's design pattern for parametric analysis.

Parametric analysis for the Extensive ISRU campaign was performed using Phoenix ModelCenter (MC) plug-in. MC enables a seamless integration between SysML model and Matlab. SysML provides a structure definition for all modeling elements (e.g., campaign, mission, etc.) and element's properties (e.g., mass, power, etc.) necessary for parametric study; while Matlab provides a computation engine as well as a capability to perform complex engineering analysis via a call to user-defined functions.

4

American Institute of Aeronautics and Astronautics

For this effort, there are two nested parametric analyses. The inner analysis computes mass roll-up of each launch based on its payload's content. Launch's masses are then used along with other campaign's properties in the outer parametric analysis for a call to perform discrete event simulation (DES) function (implemented in Matlab) for the campaign. After the simulation is terminated, its outputs will be written back to the associated instances of the SysML elements. As part of the Matlab DES function, several time series of cumulative masses and arrival and departure counts at different orbits are also captured. Figures 5 and 6 present sample screen shots for the parametric analysis. There are several possibilities for sensitivity analyses to be performed. The model can be used, as examples, (1) to evaluate alternative campaign designs (e.g., launch date, launch vehicle, etc.) throughout project life cycle, (2) to determine the quantitative values of requirements during the earlier project life cycle, or (3) to identify driver(s) of system's performance requirements (e.g., propulsion performance is critical to the project success, etc.) as part of resource planning and control process. With limited time for the Pathfinder effort, the team successfully explored a design trade space to determine how many months the first launch (from Earth surface) for this campaign can be delayed without delaying the first arrival on Mars surface.
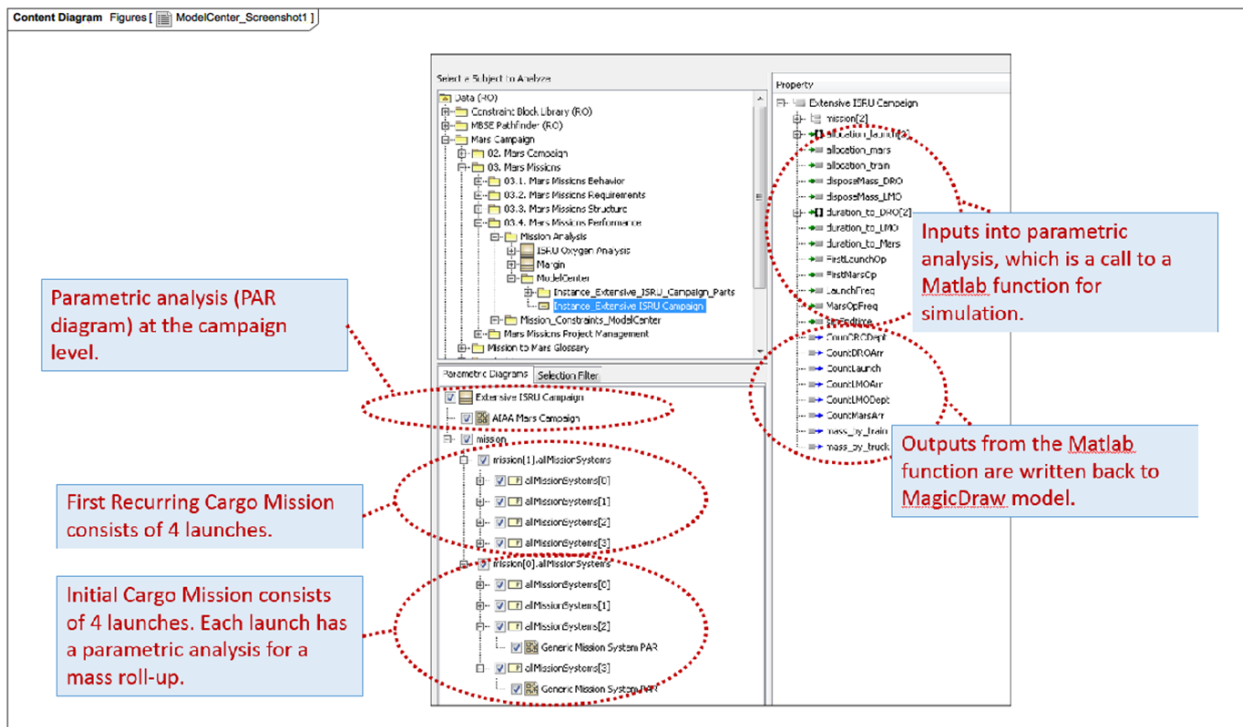
Figure 5. Parametric analysis for the Extensive ISRU campaign with two cargo missions.
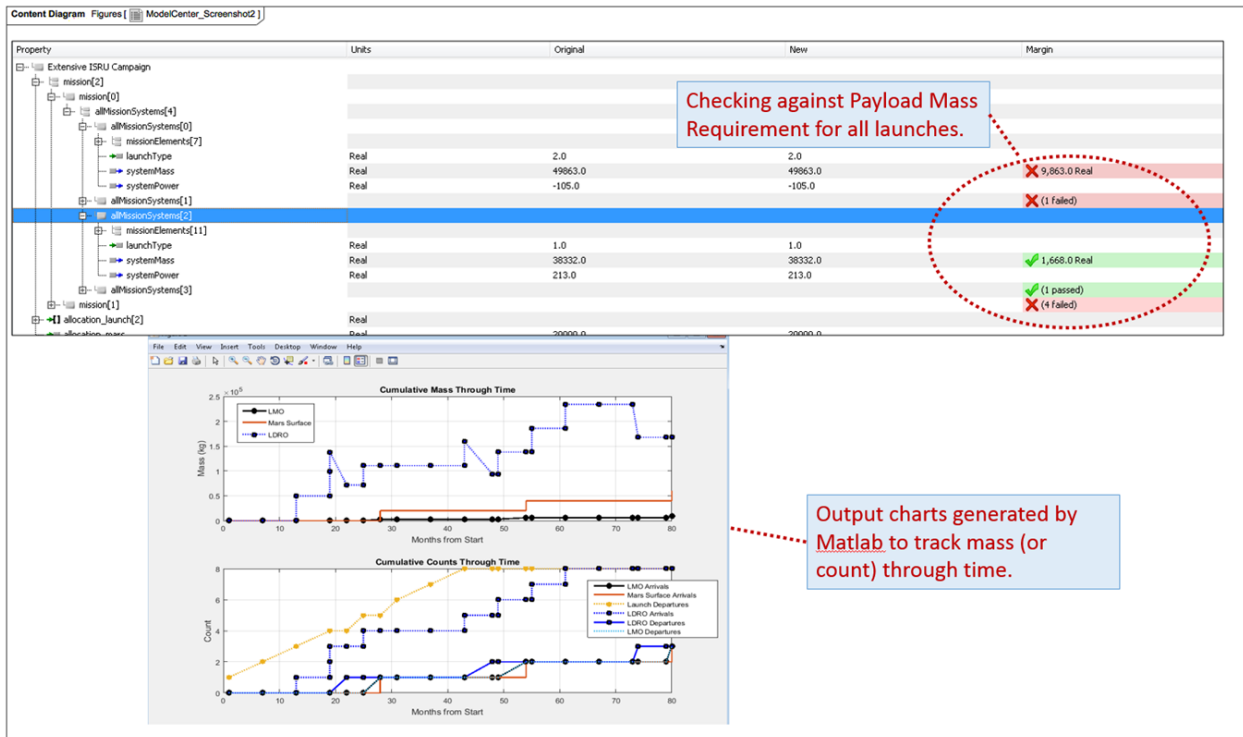
Figure 6. Output of parametric analysis for the Extensive ISRU campaign with two cargo missions.

3. Objective 6: Model Libraries

SysML Model Libraries are packages designated to contain reusable elements that are intended to be shared with other Models or Modelers. Model Libraries should contain elements common to a specific domain. For this effort, two logical groupings of SysML elements with commonalities were identified and placed into *Model Libraries* package (as shown in Figure 2.) Two libraries are for ISRU resource types and for Value types. The Resource library (Figure 7) contains a hierarchical structuring of resources, their locations and their associated technology areas. The Value type library (Figure 8) contains Value types, Units, and Quantity Kinds necessary for parametric analysis. In addition to these libraries, the team used several user-defined profiles. Similar intent usage to libraries, profile is used to extend the SysML language and allow the disciplined engineer or modeler the ability to use "terms of the trade". The reader is refered to Ref. 2 for the full description of library and profile. The following profiles were created and used for this effort, and can be reused for other projects beyond the Pathfinder.

• A Pathfinder team profile to capture and ensure consistency and completeness for team member information, which include name, role, NASA center, and contact information.

• A Lessons Learned profile to capture the full content of Lessons Learned including owner, related Pathfinder objective, driving event, lesson description, recommendation, and category (e.g., modeling, model governance, etc.)

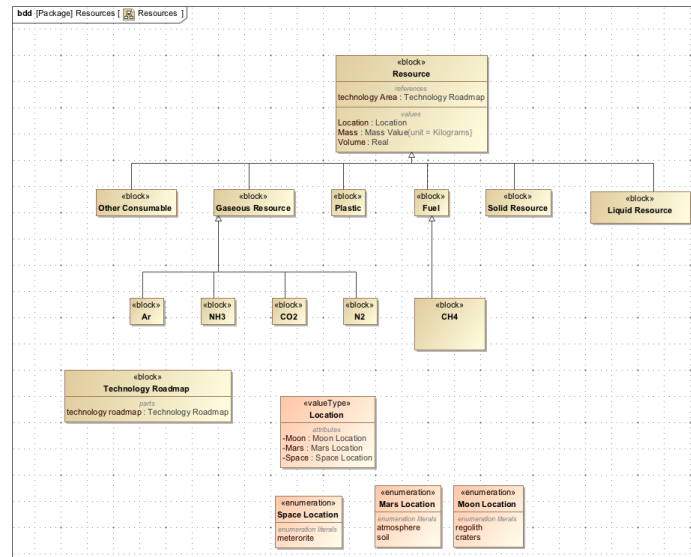• A Campaign and NPR profile to enable the use of Agency specific language in the model.

American Institute of Aeronautics and Astronautics

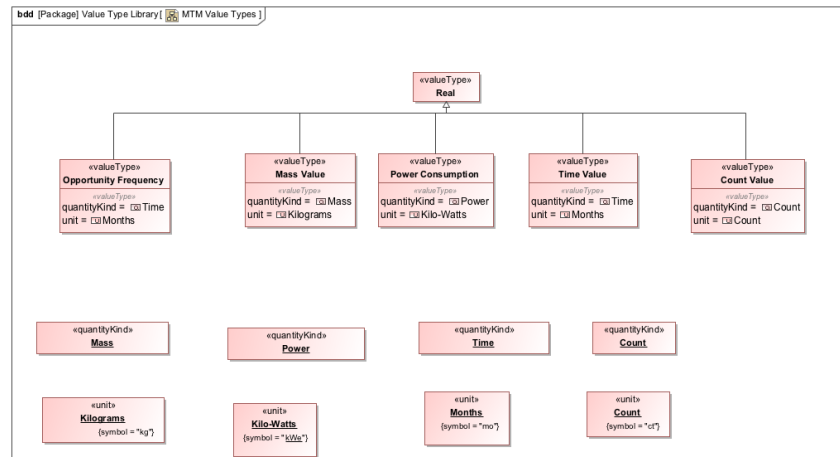**Figure 7. Resource type library for ISRU.**

Figure 8. Value type library necessary for parametric analysis.

B. Goal 2

Goal 2 is to mature the "Extensive ISRU" campaign from the architecture/mission design analysis toward a Program SRR by producing systems engineering products in a model-based way. There is one objective to fulfill this goal and is given in the next sub-section.

4. Objective 5: Reviews

The team's area of focus was a Mars campaign (Program level) and cargo missions to transfer 20 tons of ISRU cargo to the Mars surface (Mission level). For this objective, the team was to develop models that represent lifecycle review products that can satisfy NASA lifecycle review entrance and success criteria as defined in NPR 7120.5[9] and 7123.1[10] for the following reviews:

- Program (Campaign) level: System Requirements Review and System Definition Review
- Project (Mission) level: Mission Concept Review and System Definition Review

Table 1 summarizes review products that were produced from the model broken down by program and project levels. Some samples review products from the model are provided in Figures 9 to 12.

---

[9] NASA Space Flight Program and Project Management Requirements
[10] NASA Systems Engineering Processes and Requirements

Table 1. Review products generated in the model.

Campaign / Program SRR/SDR	Mission / Project MCR/SRR
Stakeholder Needs, Goals, and Objectives	Mission Goals and Objectives
Program Requirements	Concept of Operations (CONOPS, Figure 9)
Program Interfaces	Mission Requirements
Requirement Traceability and Allocation Data (Figure 10)	Mission Architecture
Program Architecture	Model Tree/Structure (Figures 2, 3, 11, and 12)
Technical Performance Measures (TPMs)	
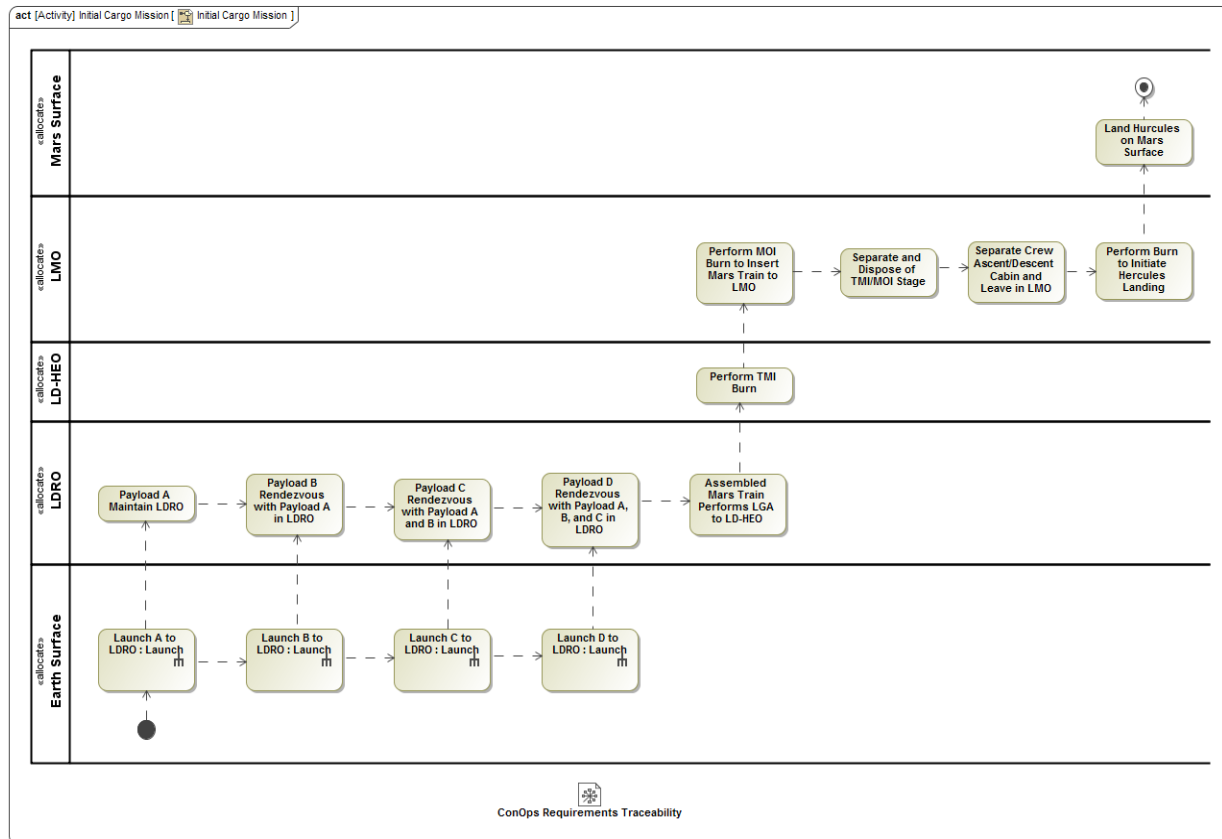
Figure 9. Initial Cargo Mission CONcept of OPerations (CONOPs).

American Institute of Aeronautics and Astronautics
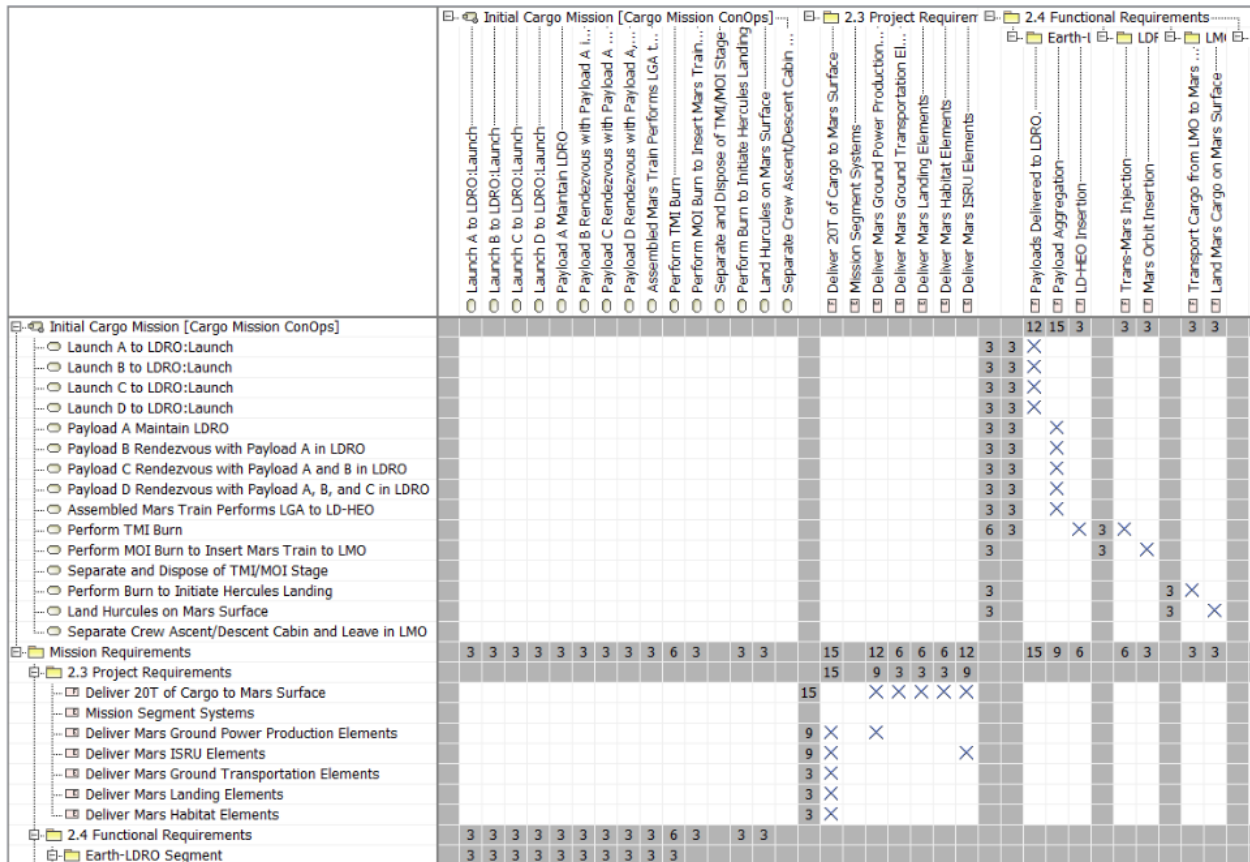
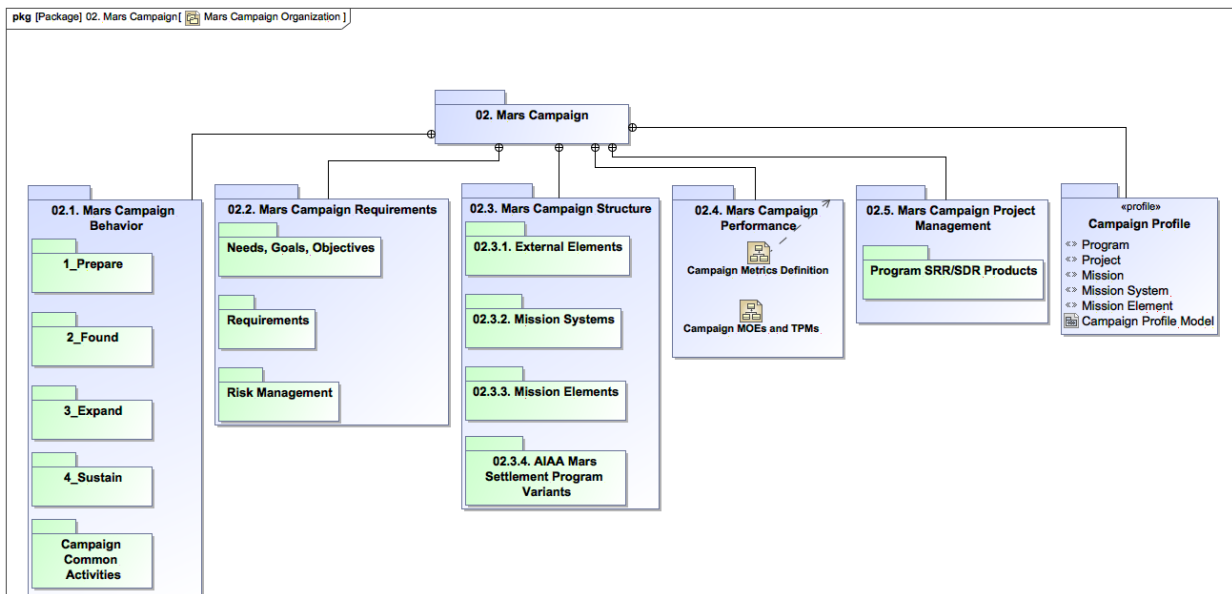**Figure 10. CONOPs to Requirement Traceability Matrix.**

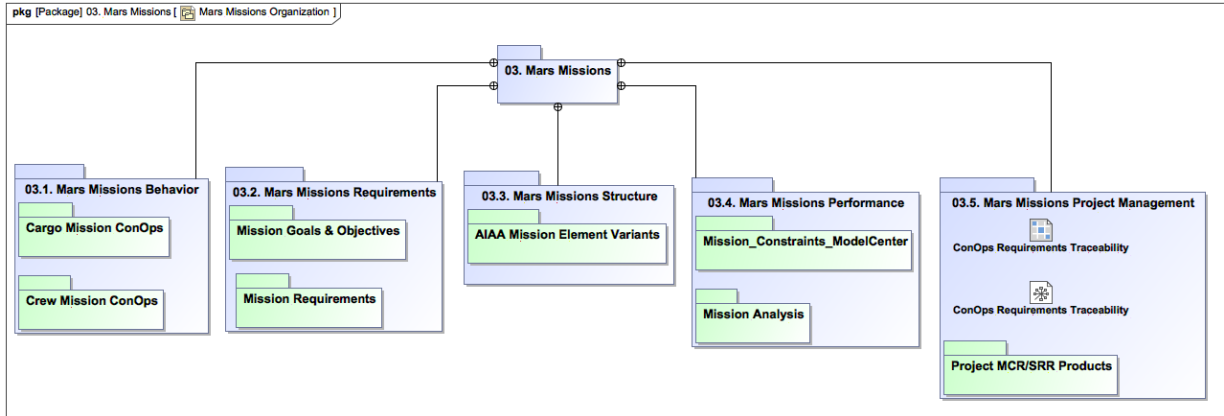**Figure 11. Mars Campaign Organization.**

Figure 12. Mars Mission Organization.

C. Goal 3

Goal 3 is to understand, demonstrate, and apply concepts of model re-use. There is one objective to fulfill this goal and is given in the next sub-section.

5. *Objective 7: Design Patterns*

SysML itself does not prescribe a particular modeling approach or pattern to follow for any given model or model diagram. Therefore, there are as many different ways to represent a given system as there are modelers. In order for multiple modelers to create a self-consistent and valid model, certain common approaches and guidelines are necessary across the different modeling products. One approach is to standardize modeling products or the views of modeling products through the use of design pattern concept. Design patterns are the modeling templates or views developed and used by an organization to help standardize the way certain modeling products are produced and interpreted. Through the Pathfinder efforts, the team produced some recommended design patterns for future NASA missions/projects to consider applying to their models. These patterns are in the area of Behavior, Structure, Requirements, and the relationships among them. Samples are shown in Figures 13 and 14 for Structure and the relationship patterns.
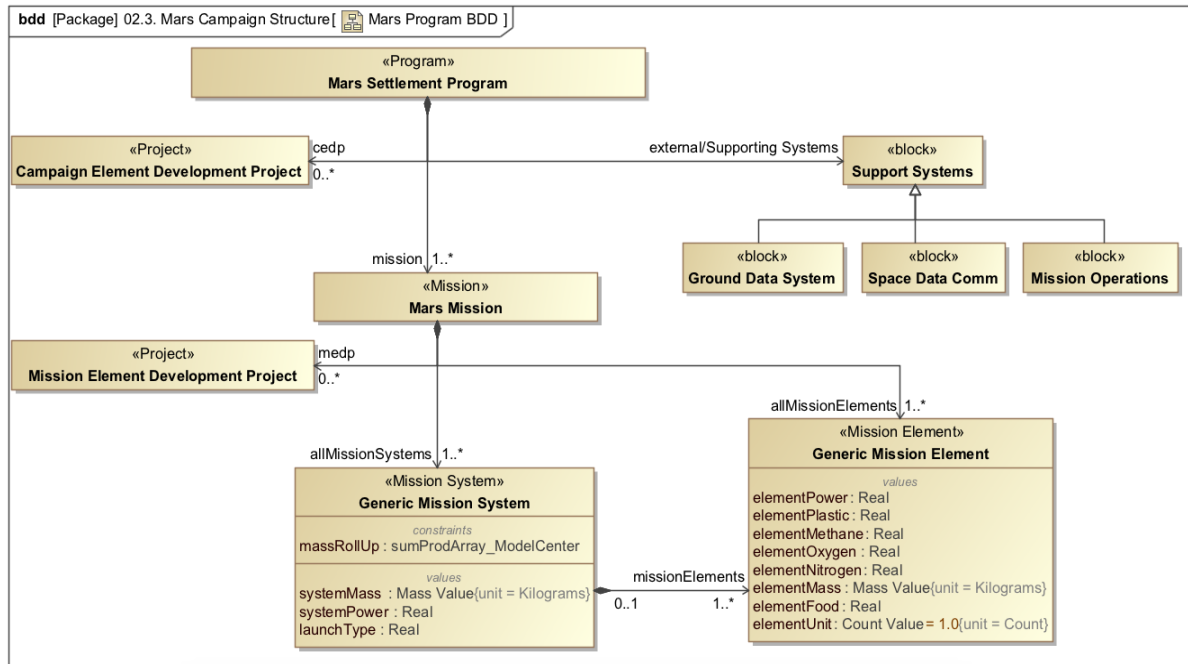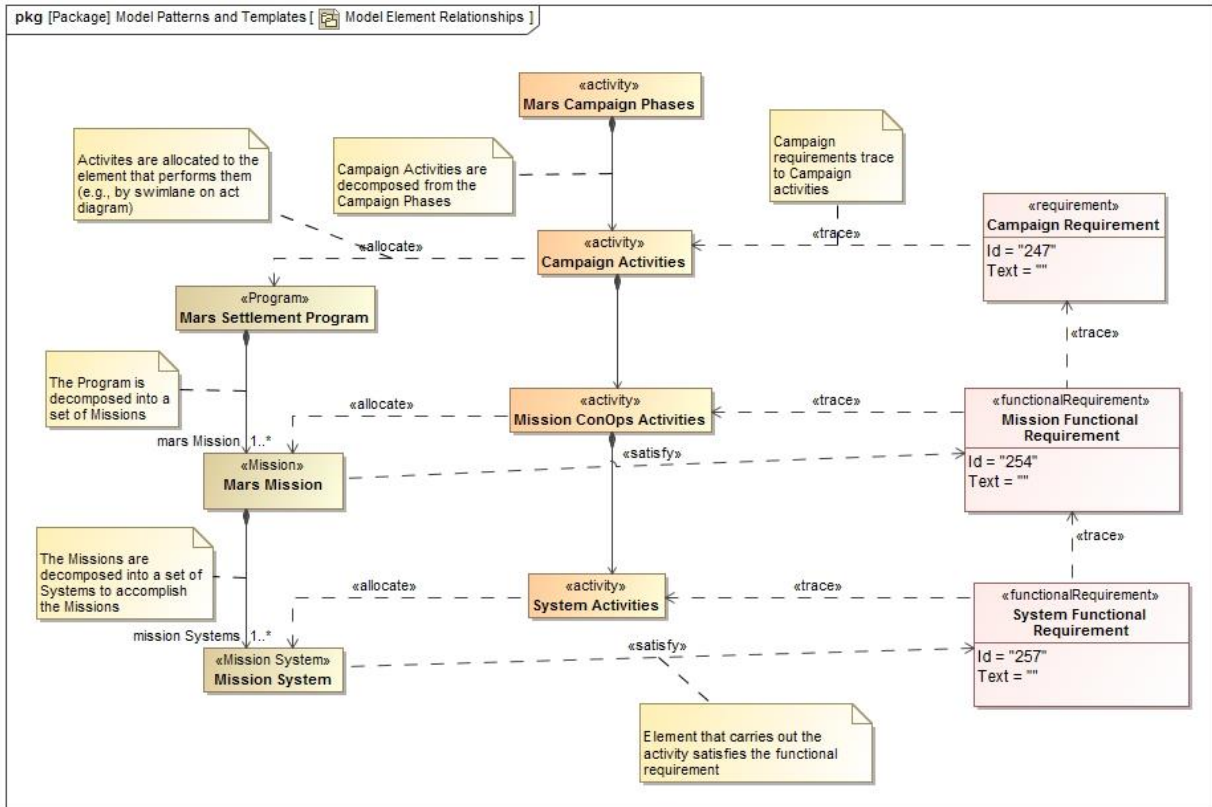

Figure 13. Design Pattern for Mars Campaign Structure.

10

American Institute of Aeronautics and Astronautics

Figure 14. Design Pattern for Model Element Relationships.

D. Goal 4

Goal 4 is to learn configuration management and collaboration best practices with respect to modeling and assess export capabilities of tools. There are two objectives to fulfill this goal and are given in the next sub-sections.

6. *Objective 3: Collaborative Modeling and Configuration Management*

One of the key benefits of MBSE is the self-consistency that accrues when multiple aspects of a system are captured in a single model. For NASA-scale projects this almost always requires a group of modelers to work collaboratively on different parts of the model, so the tools and workflow to allow updates to a common model are critical for effective collaboration.

MagicDraw offers model repository called Teamwork Server for version control and updates for SysML models. It uses concepts familiar to other software version control systems for "checking out" and "committing" changes.

By default, a user accesses a model in "read-only" mode. The user then "locks" a part of the model, making it "editable" by the user and "read-only" for all other users. The lock feature works at the model element level, so different users can have different parts of the model "locked" for edit simultaneously (but two users cannot both have the same model elements locked at the same time).

Once a user is satisfied with the edits, they can then be "committed" back to the model, creating an incremental version of the system model on the server. A complete version history is maintained on the server, to allow for reference or reversion to an older version.

The primary difficulty with this arrangement is if people do not follow good "lock/unlock" etiquette, leaving parts of the model that others need to edit "locked" when they should not be. This is mostly obviated by promoting a best practice in collaborative modeling to use "unlock all" command before ending a modeling session.

MagicDraw's Teamwork Server also supports configuration management capability through its Project Merge plugin. The plugin offers the software versioning idea of "branching" and "merging" to facilitate configuration management. The user creates a branch of the project that is saved as a unique version on Teamwork server based on the "production" or "trunk" model. This unique version can then itself be versioned to iterate on that branch of the model. In the example below (Figure 15), a branch called "Spike 03" was created based on the trunk version 155,

and was committed as Spike 03 version 0. A subsequent version 1 was committed for this branch. Meanwhile, development of the trunk model continues, incrementing to versions 156, 157, etc.

While the branching function is a standard feature of Teamwork Server, merging requires Project Merge, which allows the modelers to merge a branch back into the trunk and reconcile changes and conflicts. Reconciling differences in models has a degree of difficulty greater than reconciling differences in software code, in that it's not a simple "diff" or comparison between lines of code in different versions, but also a difference in the diagrams built from the model elements. Project Merge addresses this by providing both detailed diagnostic and graphical views of model changes.

The initial finding in evaluating Project Merge is that it requires all modelers to have the model in an unlocked state for all the branch and trunk models. By the nature of having so many modelers working in parallel, this essentially means that a stop-work condition needs to be put on the model while the merging takes place. It is recommended that Project Merge is probably most useful when one or two modelers are proposing significant changes to the trunk model, and those changes should be carefully considered by the team prior to incorporation. Standard "lock/unlock" should be used for day-to-day model changes.

Content Diagram Figures [History Log]		
160	rcpowell	6/16/16 4:06 AM
159	rcpowell	6/15/16 1:16 PM
158	kalee1	6/14/16 2:02 PM
157	nphojana	6/14/16 1:21 PM
156	gvorndra	6/14/16 9:37 AM
155 (Spike 03)	stmille1	6/13/16 11:02 AM
Spike 03	stmille1	6/16/16 9:57 AM
1	stmille1	6/16/16 9:57 AM
0	stmille1	6/13/16 11:11 AM
154	kalee1	6/13/16 10:52 AM
153	kalee1	6/13/16 10:29 AM
152	kalee1	6/13/16 7:59 AM
151	gvorndra	6/10/16 1:08 PM
150	gvorndra	6/10/16 1:06 PM

**Figure 15. Screenshot of Model History Log.**

The team also experimented, within the main project model, with Sandbox packages. Each team member was given a personal sandbox package where personal work and discovery could occur. The motivation for this sandbox setup is that the team consented that each member needed the freedom to edit, learn to model, and change the model without making changes to the current state of the model and without causing conflicts to other team members. When a team member wanted to explore and prototype with the model, the necessary model elements were copied into the member's sandbox package, and the work was performed in that user's sandbox. If a member's work was deemed by the team to be helpful for the model, that member's work was copied back into the main model packages. The team later faced model validation errors, generated by the active validation tool of MagicDraw, for duplicated elements (with the same unique element IDs) between the main model and the Sandbox. As a result, the Sandbox was eventually removed from the main model.

7. Objective 4: Model Export Capability
The team has successfully evaluated existing model export capabilities. As part of the evaluation, the team uses this export capability to generate the final report. There are two approaches to generate a document directly from the model in MagicDraw at no additional cost. One is MagicDraw Report Wizard and the other is MagicDraw Document Modeling Plug-in.   The Report Wizard comes with generic templates in various formats (e.g., Microsoft

American Institute of Aeronautics and Astronautics

Word, Microsoft PowerPoint, html, Microsoft Excel, etc.) In order to tailor the templates to suit the team's need, team members would need to learn the Velocity Template Language (VTL) and Open API scripting within the limited timeframe. With this challenge in mind, the team chose to evaluate and use the Document Modeling plugin instead. This approach doesn't require team members to learn scripting language, but does require learning to model using viewpoints and views concept.

The document model for the team's final report is shown in Figure 16. It is important to note that this figure is intentionally presented at the bird-eye view for the reader to see the overall laborious setup of the chosen approach. It is not an intent to be readable. Although the Document Modeling plugin has shorter learning time, besides from its labor intensive it is non-trivial to get the document into the desired format with this approach. The ability to fine tune the document style to a professional look requires an external document post-processing tool. It is recommended that the other approach of VTL scripting should be explored in order to compare and contrast with the current approach.

American Institute of Aeronautics and Astronautics
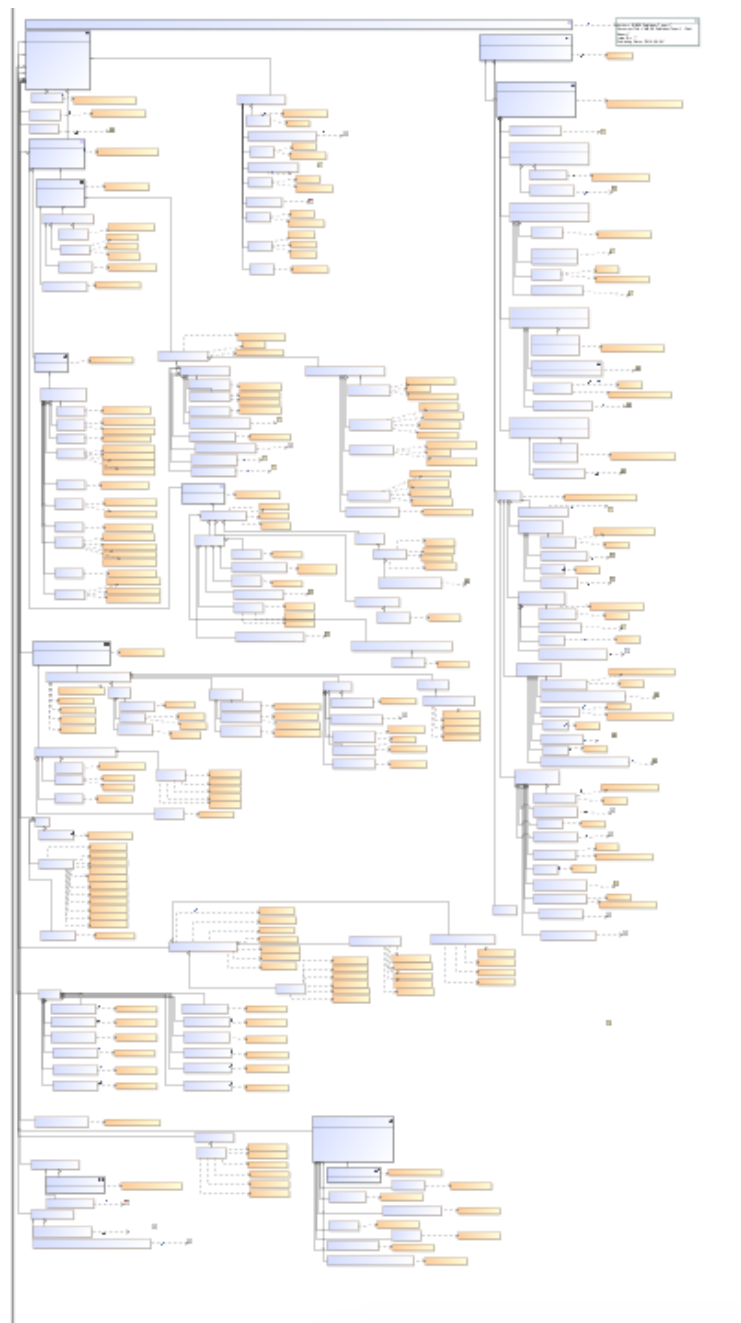
Figure 16. Document Model using the plugin.

III. Lessons Learned

Pathfinder team 1 identified over twenty lessons learned during these efforts. Categories of lessons learned include modeling, model governance, team operations and the MagicDraw SysML tool. Three key lessons are:

- Parametric Analysis - Basic analysis (e.g., simple mass roll-up) is possible within MagicDraw and its plugins available during the time of the effort. Performing time-based simulation analysis (or other disciplined analysis) required the team to use external tools to integrate advanced analysis with MagicDraw. This was not already part of the standard MagicDraw/MBSE Pathfinder toolset.
- Collaboration – The team used separate "sandbox" packages in the main model to experiment. However, this ended up creating validation errors due to multiple copies of model elements. It is recommended to keep Sandboxes as separate models and have team members incorporate changes into the main team model.
- Terminology – The team spent a lot of time rehashing terminology and encountered a lot of frustration and rework as a result of each team member leveraging their own experience(s) and the entire team not being on same page regarding terminology, definitions and hierarchy.

IV. Conclusion and Recommended Next Steps

Pathfinder team 1 was successful in delivering results on goals and associated objectives. Accomplishments by goals are summarized in Table 2.

Table 2. Accomplishments by goals.

Goal	Goal Text	Accomplishments
1	Demonstrate system modeling for mission architecture use and reuse for a human to Mars campaign of missions. The system modeling would be done at the campaign and mission levels.	• The team demonstrated system modeling by developing numerous models for a Mars Campaign and Missions to take ISRU cargo to the Mars surface. • Some of the developed models can be reused by other teams and are in a model library, as part of final deliverables. • The team acquired knowledge of, as well as developed some lessons learned and observations regarding model organization for a (Program) Campaign of Missions. • The team performed sensitivity analyses at the Campaign level for two factors (mass limits for a particular launch, slips/delays in launch dates).
2	Mature the Extensive ISRU campaign from the architecture/mission design analysis toward a Program SRR by producing systems engineering products in a model based way.	• The team developed 11 model based systems engineering products for the Extensive ISRU Mars Campaign. • The model based systems engineering products generated represent a subset of lifecycle review products for a Program SRR, Program SDR, Mission level MCR

Goal	Goal Text	Accomplishments
		and Mission level SRR.
3	Understand, demonstrate, and apply concepts of model reuse.	• The team acquired knowledge of and, to some extent, applied model reuse as we developed our model based systems engineering products at the Campaign and Mission levels.
		• The team demonstrated model reuse by developing a model library that contains several reusable modeling assets.
4	Learn configuration management and collaboration best practices with respect to modeling and assess export capabilities and tools.	• The team acquired knowledge of, as well as developed numerous lessons learned and observations regarding configuration management and collaborative modeling.
		• The team assessed the Document Modeling plugin and utilized it in the generation of the final report.

After seven months supporting the Pathfinder efforts, and leveraging the lessons learned, wisdom, and experience gained from participation, Pathfinder team 1 members were well positioned to identify recommended next steps related to the focus area (architecture and mission campaigns), the Pathfinder effort in general, and MBSE within the Agency. The recommended next steps include: continue to mature and advance the demonstrated parametric capability associated with performing trade analysis for a Mars Campaign; creation of an Agency wide collaboration space for MBSE; development of common and reusable assets; further refinement of model governance principles; establishing a MBSE "boot camp"; and ascertaining the concepts of model Verification and Validation.

Acknowledgments

References

1. Arney, D., Jones, C., Klovstad, J., Komar, D., Earle, K., Moses, R., and Shyface, H., "Sustaining Human Presence on Mars Using ISRU and a Reusable Lander," *AIAA SPACE 2015*, AIAA 2015-4479.
2. Friedenthal, S., Moore A., and Steiner, R., *A Practical Guide to SysML, The Systems Modeling Language*, 3rd Edition, Elsevier, 2015, Chapter 15.