A Model-Based Framework for NASA Science Mission Formulation

Dr. Matthew L. Marcus[1], Henock A. Legesse[2], and Dr. David H. Richardson[3]
NASA Goddard Space Flight Center, Greenbelt, MD 20771 USA

This paper details an effort to implement NASA systems engineering standards and practices using a Digital System Model (DSM), which we also refer to as the system architecture model (SAM). Creating a SAM at the beginning of a design effort helps systems engineer identify errors, inconsistencies, and miscommunications as early as possible. Such issues can then be resolved before it becomes costly and requires significant rework to do so. A pre-formulation SAM also enables advanced trade studies at the earliest stage of concept development. This results in "win-wins" where architecture changes can reduce cost without decreasing effectiveness, or increase effectiveness without increasing cost. SAMs also streamline the creation of project documentation and facilitate superior dialogue between science, engineering, and management stakeholders. Common artifacts such as Master Equipment Lists (MELs), Science Traceability Matrices (STMs), and Mission Traceability Matrices (MTMs), can be automatically maintained through requirements and structural models in the SAM. Key performance parameters (and changes to them) can be tied to simulations in the SAM, allowing far more rapid and extensive trade space exploration. As will be shown in this work, these advantages have been realized for the first time in an actual NASA Goddard Space Flight Center (GSFC) pre-phase A and phase A concept study. Model-based design tools have been developed in a general, modular manner to maximize reuse on future programs. Focus is placed on structural and requirements architecture modeling. Data is ingested into the SAM from existing discipline model outputs (e.g. MS Excel spreadsheets) and used to update a SysML structural model. A mission requirements model is created within the SAM, containing mission specific as well as standard GSFC mission requirements. With the linked requirements model and structural model, automated compliance checking will be performed as mission parameters evolve without the need for discipline engineers to work directly with the SAM in SysML.

Nomenclature

DSM	=	Digital System Model
GSFC	=	Goddard Space Flight Center
MBSE	=	Model Based Systems Engineering
MD	=	MagicDraw
MEL	=	Master Equipment List
MTM	=	Mission Traceability Matrix
NGOs	=	Needs, Goals, and Objectives
PDL	=	Product Development Lead
PI	=	Principal Investigator
PSE	=	Project Systems Engineer
RMF	=	Requirements Modeling Framework
SAM	=	System Architecture Model
SE	=	Systems Engineer
STM	=	Science Traceability Matrix
V&V	=	Verification and Validation
WBS	=	Work Breakdown Structure

[1] AST, Flight Systems Design, Mission Systems Engineering
[2] AST, Aerospace Flight Systems, Instrument/Payload Systems Engineering
[3] Manager, Astrophysics Line of Business

I. Introduction/Motivation

The past decade has ushered in dramatic changes in spaceflight. Rapidly falling launch costs [1], miniaturization of flight instruments and electronics, new investment from terrestrial technology industry [2, 3, 4, 5], and growing government interest in fostering a commercial in-space industry [6, 7, 8, 9, 10, 11] have sparked a seismic shift in humanity's relation with spaceflight. These factors clash with the ongoing labor shortage, which has affected numerous industries globally, including the spaceflight industry. Existing workforces are struggling to meet a growing demand for space mission design. NASA has recognized the need to modernize standard engineering practice to address this situation. Focus has been placed on optimizing systems engineering processes, effectively managing concept formulations amidst growing program complexity**,** and enhancing the quality of concept development, all while increasing systems engineering throughput of new mission concepts. MBSE tools present key capabilities to realize these objectives. This study represents the beginning of GSFC's effort to develop a digital thread which follows new mission concepts through all phases of development, from initial pre-formulation through launch, operations, and eventual decommissioning.

In this study, model-based systems engineering (MBSE) tools and techniques are applied to the formulation phase (pre-phase A and phase A) of a competitive medium class science mission proposal. These are principal investigator (PI) led missions. They are usually managed by a major space flight/space research center such as Goddard, and may involve multiple additional partner organizations. Key interests at formulation are the science organization, staffed by research scientists and led by the PI, the engineering organization, staffed by mission and spacecraft discipline engineers and led by mission systems engineers and instrument systems engineers, and a management organization, which includes business, cost, scheduling and other programmatic specialists. Much of the work of a proposal occurs at the interfaces between these organizations, managed by systems engineers (SEs) through a mission architecture, as shown in Figure 1. Each interface is described by a design artifact, which must be formally included as part of any PI-led mission proposal [12].

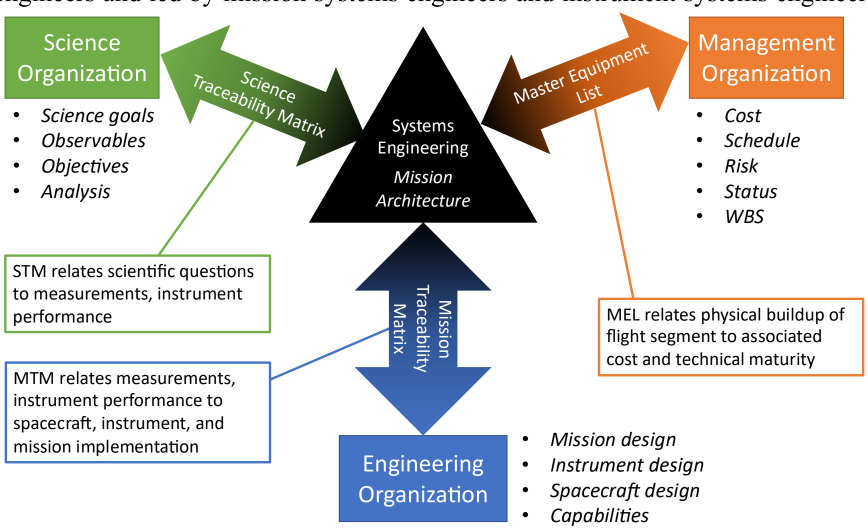

Figure 1. Interfaces between major mission proposal organizations

The interfaces to the science and engineering organizations are described by a Science Traceability Matrix (STM) and Mission Traceability Matrix (MTM), respectively. The STM starts with the major science questions a mission seeks to answer. Through several steps of successive requirements, it distills these questions into measurement objectives, observation requirements, and instrument requirements. A mission architecture is created to achieve the observation requirements set by the STM. The MTM continues this process, deriving spacecraft, and mission requirements to achieve the developed architecture.

The interface to the management organization is represented by a Master Equipment List (MEL). The MEL translates the planned implementation of the flight segment, represented by a detailed bill of materials, broken down by a functional architectural hierarchy (i.e. element/system/subsystem/component) into a form to aid in management considerations (schedule/cost/maturity level/risk). It contains details on the individual systems, subsystems, and components of a spacecraft following this hierarchical decomposition. For each line item within this hierarchy, the MEL lists details relevant to both the engineering and management organizations. Examples include mass, power, and Technology Readiness Level. The management organization uses the MEL to create a Work Breakdown Structure (WBS). Project and independent costing (within the management organization) utilize tools which consume the MEL as a basis for their cost estimate. It is through the MEL that the Product Design Lead (PDL) communicates design changes to the systems engineer. The MEL, like the STM and MTM, must be formally included as a part of any PI-led mission proposal [12].

Conventionally, there are two problems that plague project systems engineers (PSEs) when managing the MEL. First, the problem of branching: different project subteams produce updated MELs based on their design work. Often, propagating these MEL updates presents a bottleneck. When two subteams are interdependent, this leads to subteam designs based on outdated information. This situation leads to the second problem, the problem of consensus. The PSE needs to create a single agreed upon MEL based on the updates from each subteam. This exercise is tedious, error prone, and takes tremendous amount of time, which adds unnecessary burden to the PSE workload. The MBSE tools developed in this work can address both of these problems. The SAM can act as a single source of truth, streamlining the ingesting of MELs from subteams and automatically outputting an updated master MEL as each update is accepted.

The present work focuses on the earliest (pre-phase A) stages of mission concept development. A highly integrated mission model, which includes a detailed system architecture model, has been created as part of the proposal effort. It is designed to allow seamless extension of the digital thread into later stages of the program. For this work, the SAM includes a structural model of the mission, a connected requirements model, capturing requirements from the STM, MTM, GSFC standards and practices [13], and routines to ingest and export conventional design artifacts such as a MEL in spreadsheet form, facilitating updates to and from the structural model.

The scope of this work is limited to a demonstration of how MBSE tools and practices enhance communications between key mission stakeholders (the science, engineering, and management organizations). Focus will be placed on specific deliverables for each, namely, the STM, MTM, and MEL. A requirements modeling framework (RMF) to be applied to future GSFC missions is outlined. A routine for auto-updating a MEL tied to the digital twin of the flight segment, based on input from stakeholders and PDLs will then be described. An MBSE-derived STM and MTM will be developed based on this RMF. While this study focuses specifically on the STM, MTM, and MEL, work will be performed in a generalized, modular manner to maximize reuse and minimize adaption of models necessary for future missions. Though tools and techniques are developed for a competitive, PI-led mission proposal, they are widely applicable to directed missions as well. The model will also be developed in a way that facilitates extension of the digital thread in later phases of the program.

II. Methodology/Implementation

A. Requirements Modeling Framework

The most fundamental portion of this work is the development of a standardized SysML RMF which can be applied to future flight programs. Key requirements in the STM and MTM, and the relations between them, are greatly enhanced through the RMF. The RMF both captures requirements and links them to the mission model. The PSE can use diagrams connected to the RMF to ensure stakeholders' needs, goals, and objectives (NGOs) will be satisfied by the proposed system design. A requirement diagram for a given requirement serves multiple purposes; to act as a one stop shop for the system engineer to communicate top-level requirements, to ensure requirements are covered by the system design, and ensure consistency between different aspects of the system under development. Our process of creating a requirement diagram can be described in eight steps, with an example diagram shown in Figure 2a:

1. **Determine the text and type of each requirement within the system model**

   Within the requirements block is found the information that is typically specified for a requirement in spaceflight projects; this includes the primary text, which contains the formal requirement "shall" statement which must be met, as well as the type of requirement and any rationale that may elaborate on the primary text. Requirements are typed with extended requirement types (functional, performance, interface etc.) [14]. Use of extended requirement types facilitates built-in requirement validation capabilities in SysML.

2. **Use "satisfy" relationships to link each requirement to the correct model element**

   A satisfy relationship connects the requirement to the model element that satisfies it. For functional requirements, one would create a placeholder for the activity that satisfies a functional requirement and fill in the detail when the required information is developed. For performance requirements, the requirement would be satisfied by a value property or function of value properties within the model.

3. **When applicable, use “allocate” relationships to link a requirement to the party with ultimate responsibility for ensuring a requirement is met**

   Performing this step in pre-formulation ensures a responsible party is assigned as soon as possible.

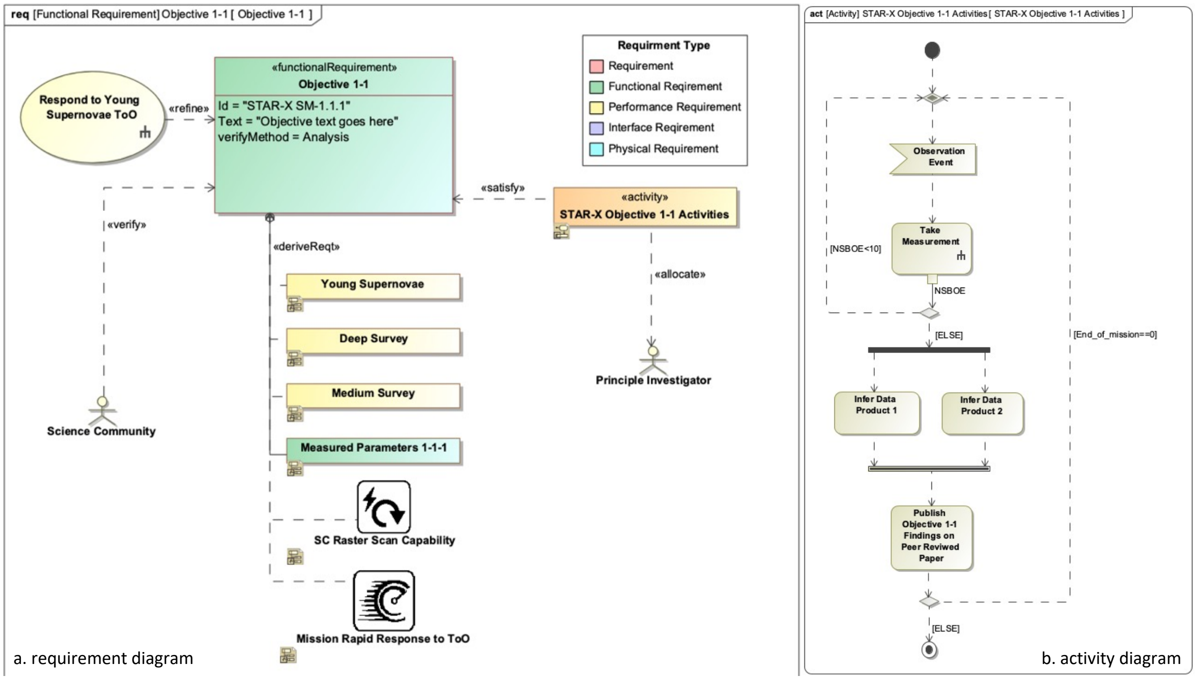

Figure 2. Example requirement diagram and resulting activity diagram

4. **When applicable, use the “refine” relationship to indicate the mode or configuration of the system in which the requirement is satisfied**

   To facilitate step 4, multiple use cases may be defined. We define a use case as a distinct design configuration of our system can exhibit. For example, operating in a nominal science mode, using instrument A to make observations (when a spacecraft potentially has multiple instruments onboard) with given instrument settings (filter, pointing performance, etc.).

5. **Specify the verification method (Analysis, Test, Demonstration, or Inspection) within the requirement block**

6. **Use the “verify” relationship to point to the evidence confirming that the requirement has been met (e.g. test report, analysis document, etc.)**

   The ‘verify’ relationship communicates two things; One, it establishes a relationship to the authority that has the final say as to whether the requirement is satisfied. Second, it establishes a link to the “paper trail” confirming that the requirement has been satisfied.

7. **Use “derive” relations to establish links between requirement levels**

   Derive relations point from child requirements (lower level requirement) to their parent requirements (higher level requirements). Requirement levels are defined based on the NASA Space Mission Architecture Framework [15]

8. **Use the "containment" relationship to indicate nested or "self-contained" sets of requirements**

   There are two cases where requirements can be nested; first, when a set of requirements directly descend from each other without branching into multiple requirements at a single level, and second, when multi-statement requirement must be broken into multiple single statement requirements

Within a given model, this eight step process is applied to the requirements at the level of focus of that model. For example, in a space mission model, the process would be applied to top-level mission requirements. It would not be applied to the science requirements, AO requirements, or NASA standards and practices from which the mission requirements derive, nor would it be applied to the instrument, spacecraft, and ground segment requirements which derive from the mission requirements.

B. Model Validation Using the Requirements Model

Compliance checking in a SAM can be achieved by creating a compliance rule in the form of a performance requirement. For instance, if the PSE wants to ensure that the mass property of an instrument does not exceed the absolute allowed value of 100 kg, they may create a performance requirement stating the constraint Figure 3 and using the satisfy relationship to link it to the block value property representing the instrument mass. Assuming the requirement glossary turned on the MD settings, when simulating the instrument block, value property violation of requirements will be highlighted red. In addition, by using embedded opaque behaviors containing custom python scripts, margin calculations on all performance requirements can be displayed in a table format for the PSE convenience, as shown in Figure 3.

Performance Requirement ...

Criteria — Scope (optional): Requirement Margins — Filter:

Margin: High Low Zero Negative N/A

#	Name	Text	Satisfied By	Margin
1	STAR-X MEL-1 XRT Mass	STAR-X XRT shall have MEV mass less than [redacted]	TotalMassWithContingency : Real = [redacted]	8.0%
2	STAR-X MEL-2 XRT Power	STAR-X XRT shall have MEV power less than [redacted]	TotalPowerWithContingency : Real = [redacted]	8.4%
3	STAR-X MEL-3 XEB Mass	STAR-X XEB shall have MEV mass less than [redacted]	TotalMassWithContingency : Real = [redacted]	8.7%
4	STAR-X MEL-4 XEB Power	STAR-X XEB shall have MEV power less than [redacted]	TotalPowerWithContingency : Real = [redacted]	20.7%
5	STAR-X MEL-5 FPS Mass	STAR-X FPS shall have MEV mass less than [redacted]	TotalMassWithContingency : Real = [redacted]	14.4%
6	STAR-X MEL-6 FPS Power	STAR-X FPS shall have MEV power less than [redacted]	TotalPowerWithContingency : Real = [redacted]	9.1%
7	STAR-X MEL-7 MS Mass	STAR-X MS shall have MEV mass less than [redacted]	TotalMassWithContingency : Real = [redacted]	6.0%
8	STAR-X MEL-8 MS Power	STAR-X MS shall have MEV power less than [redacted]	TotalPowerWithContingency : Real = [redacted]	5.8%
9	STAR-X MEL-9 TTS Mass	STAR-X TTS shall have MEV mass less than [redacted]	TotalMassWithContingency : Real = [redacted]	7.4%
10	STAR-X MEL-10 TTS Power	STAR-X TTS shall have MEV power less than [redacted]	TotalPowerWithContingency : Real = [redacted]	13.2%

Figure 3. Performance Requirement Compliance Checking Summary Table

C. Requirements Traceability

Following the process outlined in Section II.A above, preliminary science, instrument, and mission requirements are modeled. Several useful artifacts can be generated based on this model, including requirements trees, the STM, and MTM.

1. Requirements tree

Generating a requirements tree, which illustrates either the entire ancestry or progeny of a given requirement, is straightforward. These trees can be generated on demand for any modeled element. This capability enables rapid descope sensitivity analysis and other mission-level trade studies. It allows the PSE to immediately see which top level requirements would be affected by a design change that leaves the mission unable to meet a given lower level requirement. Alternatively, if a top-level requirement is under consideration for deletion, this analysis shows the impact. An example requirements tree is shown in Figure 4. The tree shows all of the higher level requirements from which a single instrument requirement (instrument on-axis angular resolution) descends, as well as the value property which satisfies that requirement (see step 2 of the RMF above). Failure of that value property to meet the performance specified in the instrument requirement implies a failure to meet the rest of the requirements in the tree.

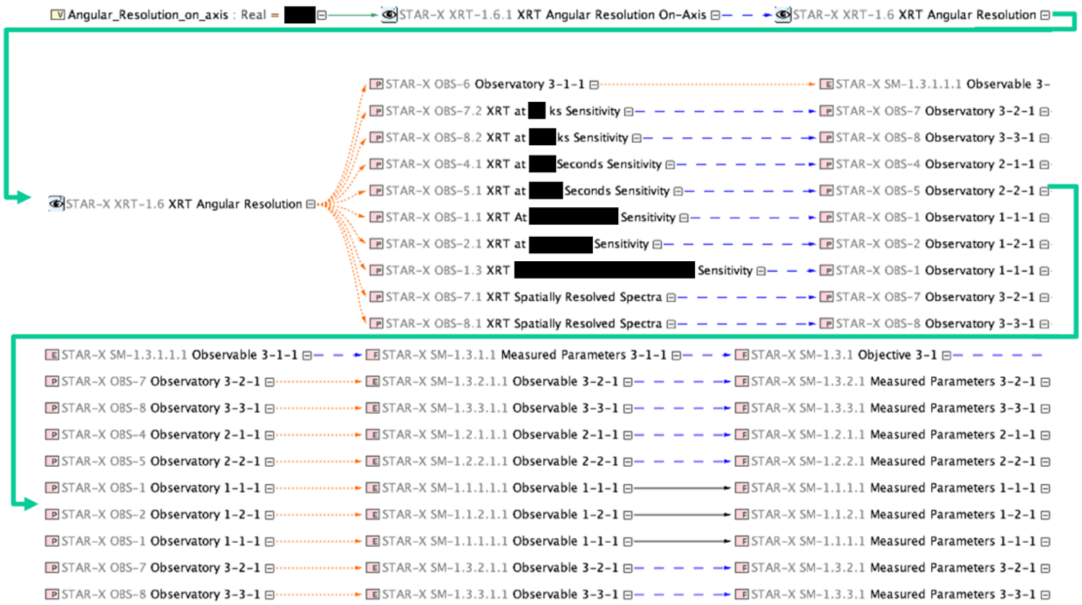

Figure 4. Example Requirements Tree

2. STM Traceability

The scope of this work originally included managing the STM within the SAM. Later insight into the ownership and responsibility for the STM, which exists outside of the SE team, has resulted in a modification of this scope. The STM is still tracked by the model, and requirements traceability within the model still largely derives from it. However, internal traceability of the STM now occurs outside the model. As its name suggests, the STM captures science goals, objectives, and characteristics of measurable and observable parameters. It links each objective to the driving requirements of the mission, instruments, and spacecraft. The SAM ingests the STM and can regenerate it in a generic table, but it does not and cannot serve as the single source of truth for the STM as it is the responsibility of the science team to manage the STM and communicate any changes to the PSE, so that the correct and current science requirements are propagated in the SAM. For the current work, the STM developed by the science team is replicated within the model, as shown in Figure 5.

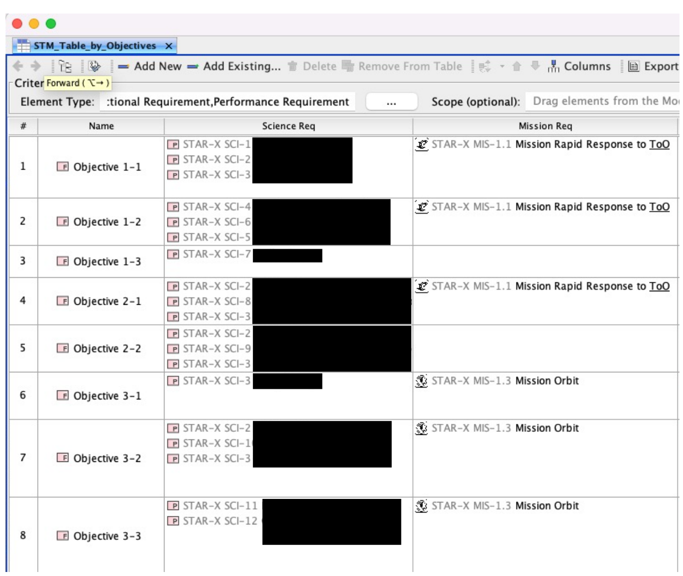

Figure 5. STAR-X STM, replicated in SysML

3. MTM Generation

Typical AOs for NASA mission specify an example MTM format, given as Table B2 in [12]. This MTM format may be used as is or modified to communicate mission, flight segment, and ground segment requirements to engineering stakeholders as these requirements are being developed during a pre-phase A concept study. As part of this study, a workflow was developed to output an MTM in a format closely resembling this example format. The data populating this MTM was directly pulled from the requirements model described in Section II.A. Future work will be performed to enhance the MTM format for maximum readability and information transfer.

A generic table, utilizing custom columns, requirement stereotypes, and a model element that can label other model elements was used to display the contents of the MTM. The MTM output by the model categorizes requirements into top-level mission requirements, mission design requirements, flight segment requirements, ground system requirements, and operational requirements, as shown in Figure 6. All other columns are derived form the first column, top-level mission requirements. The first column displays the top level mission requirements, which come from the STM, Announcement of Opportunity, and institutional standards and stakeholders. The second column displays mission design requirements, stereotyped in the model by "Mission Design". The third column displays flight segment requirements derived from the top-level mission requirements. The fourth column displays ground segment requirements derived from the mission requirements. The fifth column displays operations requirements, which are ground segment requirements specific to operational constraints, stereotyped by "Operational Requirements".

NoMagic™ MagicDraw (MD) Structured Expressions were used to create a series of relationship-based quires producing the custom columns mentioned above. Once the MTM table was constructed in MD, it was exported as an MS Excel spreadsheet, where an Excel macro was used to post-process and format the table to match the expected format of the MTM for dissemination to engineering stakeholders on the team and for inclusion in the mission proposal. This workflow can be followed to automatically generate an updated MTM whenever the mission requirements model is updated. A full example MTM, compared to the manually created version for the same mission, is presented and briefly discussed in the results section below.

Mission Traceability Matrix

Mission Req	Mission Design Req	Flight Segment Req	Ground System Req	Operations Req
STAR-X shall be capable of performing a fast slew of XXX deg in < XXX min.	STAR-X mission shall put observatory in a XXX orbit	STAR-X instruments shall be thermally isolated from bus and will provide their own radiators for heat dissipation	Ability to transmit transient alerts at XXX bps with a latency ≤ XXX seconds	Position accuracy (see left), at ≥ XXX Hz
STAR-X shall provide a pointing knowledge of less than XX arcsec (pitch/yaw, 3-sigma) and less than XX arcsec (roll, 3-sigma)	STAR-X orbit shall be selected to facilitate passive deorbit less than or equal to 25 years after end of mission.	Spacecraft shall maintain static and dynamic clearances to the respective instrument envelopes to accommodate apertures detailed in POD, including a XX° FOV.	[Requirement Redacted]	Ability to respond to ToO commands as specified at the mission level.
STAR-X shall transmit transient alerts to MOC less than XXX of on-board identification XX% of the time.	STAR-X Launch date shall be NLT December 2028.	STAR-X spacecraft shall support downlink of science data at least XXX from PL during nominal operations.	STAR-X Ground Segment shall have bit error rate less than XXX.	
STAR-X shall process real-time transient events on-board.	STAR-X shall be compatible with XXX LV outlined in Astro MIDEX 2021 program library.	STAR-X spacecraft shall have fast slew capability of XXX degrees in < XXX min.	Science data destination GSFC SOC	
STAR-X mission shall accommodate at least XXX bits/day telemetry rate.	STAR-X mission shall be designed with the expectation of at least XXX years of mission lifetime.	STAR-X shall provide orbit average power of XXX W for instruments for nominal operations	Transmit frequencies shall be S Band	
STAR-X shall have absolute timing accuracy of less than XXX from UTC		STAR-X Spacecraft shall have 3 axis stabilized, inertial pointing.	Spacecraft data destination LASP MOC	

Figure 6. Example MTM output by model

D. MEL Management

Figure 7 shows how information flows between the MEL and the models developed for this work. GSFC missions go through a concurrent design study at the mission and/or instrument level prior to mission formulation. The MEL is a critical output from these studies. Therefore, it is assumed that we start with step 1 in Figure 7 The initial MEL is in a human readable, presentation/proposal format that would be output from a concurrent study. It is similar to the format that would be included in a mission proposal. In Step 1, we utilize a Microsoft Excel macro to convert to the MEL to a machine friendly format (comma separate values (CSV)). Step 2 occurs in a standardized project starter

SysML model developed at GSFC in NoMagic™ MagicDraw (MD). The model runs an opaque action activity that ingests the CSV MEL and autogenerates a structural model (step 3). Once the structural model has been created or updated from the CSV MEL, an automated process developed for this work can output a standard human readable MEL from the model (step 4 of the MEL cycle).

III.Results

Results from this work fall into two broad categories. The first is the development of a procedure, framework, and software to ingest and create key design documentation expected by NASA standards and practices [12]. The second is a linking of the RMF to value properties in the structural model, enabling automated performance requirements checking as the design of the real system evolves. This performance checking immediately highlights requirements

Three specific documents, the STM, MTM, and MEL, represent the major interfaces between project SEs and other key stakeholders. The first two, along with NASA and GSFC standards and practices, set the top level mission requirements, from which all lower requirements are derived. As discussed above, the STM is not managed within the model, but does lead to many of the requirements within the model, and is therefore itself partially included in the model, and is therefore not further discussed in this section. Following the processes outlined in the methodology section above, the model is used to produce an MTM which can be updated as the mission design changes and evolves. An example MTM is given below in Figure 7. The autogenerated MTM output by the model requires minimal to no modification from the SE prior to being included in the proposal.

Use of the RMF described in this work also enables additional model functionality, eliminating busywork for the SE. Specifically, linking the RMF to properties in the structural model enables automatic performance requirements compliance checking as the design of the real system evolves. This compliance checking updates automatically as the model updates. From this functionality, an augmented compliance matrix was implemented within the model, an example of which is shown in Figure 3:

The full process for MEL management, as described in this abstract, is complete. The procedure outlined for requirements modeling is complete, and the STM has been incorporated into the mission model using this process. Additional work is planned to pull parameters and requirements currently in the model to produce an MTM in an expected format for inclusion in the mission proposal (Table B2 in the NASA standard AO template [12]).

IV.Conclusions

The model, in its current state, has already enabled additional analysis of a fidelity that would not have previously been possible. Examples include requirements diagrams from the RMF itself, as well as Monte Carlo analysis of mission operations performance. The latter has been used to assess concept of operations feasibility, while the former has caught holes in the science requirements. The requirement diagram in Figure 2a led to a conversation with the project science team, resulting in the PSE capturing the required detailed top-level activities **(**Figure 2b**).** The conversation also uncovered a critical missing science requirement.

Ultimately, the SAM under development is expected to produce three key benefits to SEs working on mission development efforts at GSFC. It automates tedious and repetitive tasks, including document generation and requirements compliance checking. It reduces errors and oversights in the design, and reduces design iteration time. As a result, the SE is free to dedicate greater effort to trade space exploration. The SE's work throughput is also increased, helping to mitigate one part of the labor shortage in the industry. Second, it forces the SE to develop a formal requirements model early in the design. Third, it enhances communication with key stakeholders through products like top level requirements diagrams connected to the model. Clearer visualizations of the state of the system design than previously possible are created. As a result, the SE is better able to identify disconnects in the design at the earliest stage of design, where change and rework effort are at a minimum. Together, these three benefits provide a deeper insight into the system design through all mission phases. The result, as stated in the NASA SE handbook [14] is a "win-win" situation, where cost can be reduced without decreasing effectiveness, or increase effectiveness without increasing cost.

Mission Traceability Matrix

Mission Req	Mission Design Req	Flight Segment Req	Ground System Req	Operations Req
STAR-X shall be capable of performing a fast slew of XXX deg in < XXX min.	STAR-X mission shall put observatory in a XXX orbit	STAR-X instruments shall be thermally isolated from bus and will provide their own radiators for heat dissipation	Ability to transmit transient alerts at XXX bps with a latency ≤ XXX seconds	Position accuracy (see left), at ≥ XXX Hz
STAR-X shall provide a pointing knowledge of less than XX arcsec (pitch/yaw, 3-sigma) and less than XX arcsec (roll, 3-sigma)	STAR-X orbit shall be selected to facilitate passive deorbit less than or equal to 25 years after end of mission.	Spacecraft shall maintain static and dynamic clearances to the respective instrument envelopes to accommodate apertures detailed in POD, including a XX° FOV.	[Requirement Redacted]	Ability to respond to ToO commands as specified at the mission level.
STAR-X shall transmit transient alerts to MOC less than XXX of on-board identification XX% of the time.	STAR-X Launch date shall be NLT December 2028.	STAR-X spacecraft shall support downlink of science data at least XXX from PL during nominal operations.	STAR-X Ground Segment shall have bit error rate less than XXX.	
STAR-X shall process real-time transient events on-board.	STAR-X shall be compatible with XXX LV outlined in Astro MIDEX 2021 program library.	STAR-X spacecraft shall have fast slew capability of XXX degrees in < XXX min.	Science data destination GSFC SOC	
STAR-X mission shall accommodate at least XXX bits/day telemetry rate.	STAR-X mission shall be designed with the expectation of at least XXX years of mission lifetime.	STAR-X shall provide orbit average power of XXX W for instruments for nominal operations	Transmit frequencies shall be S Band	
STAR-X shall have absolute timing accuracy of less than XXX from UTC		STAR-X Spacecraft shall have 3 axis stabilized, inertial pointing.	Spacecraft data destination LASP MOC	
STAR-X shall be have the capacity of collecting data greater than XXX Gbits/day of science mission data and relaying it to the SOC		The STAR-X observatory shall maintain nominal downlink capability in any orientation		
STAR-X shall provide structural, electrical, digital, and thermal support of XRT		X-Ray Telescopt (XRT) radiators on narrow end of telescope tube must have a largely unobstructed view to space		
STAR-X shall provide raster scan capability of slew + settle time less than XX sec for XX degree slews.		STAR-X Spacecraft shall have pointing control error <XX/<XX/<XX arcsec (3 sigma pitch/yaw/roll).		
STAR-X shall provide a pointing knowledge of less than XX arcsec (pitch/yaw, 3-sigma) and less than XX arcsec (roll, 3-sigma)		STAR-X spacecraft shall provide science mission data downlink at or above XXX bps during nominal operations		
STAR-X mission shall respond to target of opportunity (ToO) less than XXX minutes, XX% of the time.		STAR-X Observatory shall process real-time event and transmit alert to MOC within XXXs of on-board identification.		
STAR-X shall provide structural, electrical, digital, and thermal support of UVT		STAR-X spacecraft shall have absolute timing accuracy less than XXX seconds UTC.		
STAR-X mission shall have observing efficiency greater than XX percent.		The STAR-X flight segment shall downlink science mission data at or above XXX bits/day during nominal operations		
		The STAR-X flight segment shall be capable of downlinking all science mission data within three days of its collection		

Figure 7. Autogenerated MTM produced from SAM

References

[1] H. Jones, "The recent large reduction in space launch cost," in *48th Internaitonal Conference on Environmental Systems*, 2018.

[2] Q. H. a. C. Dougherty, "Google and Fidelity Put $1 Billion Into SpaceX," *The New York Times,* 2015.

[3] C. Henry, "Amazon planning 3,236-satellite constellation for internet connectivity," *SpaceNews,* 2019.

[4] J. Foust, "Amazon signs multibillion-dollar Project Kuiper launch contracts," *SpaceNews,* 2022.

[5] Berkshire Hathaway, "Impulse Space Propulsion Secures $20M in Seed Round Led by Founders Fund," *BusinessWire,* 2022.

[6] NASA Headquarters, *Request for Proposal (RFP), Solicitation No. 80HQTR18R0011R, Commercial Lunar Payload Services (CLPS),* 2018.

[7] NASA Headquarters, Human Exploration and Operations Mission Directorate, *Next Space Technologies for Exploration Partnerships-2 (NextSTEP-2) Appendix H - Human Landing System,* 2019.

[8] NASA Headquarters, Commercial LEO Development Office, *Commercial LEO Destinations,* 2021.

[9] NOAA, *NOAA GeoXO Program Formally Initiated,* 2021.

[10] Y. W. e. at., "NASA Leveraging Commercial Communication Ground Stations for Small Satellites," in *34th Annual Small Satellite Conference*, 2021.

[11] NASA, COMMERCIAL RESUPPLY SERVICES 2 – RFI NNJ14ZBG007L, 2014.

[12] NASA, "Standard PI-led Mission AO Template," 2018.

[13] NASA Goddard Space Flight Center, GSFC-STD-1000 - Rules for the Design, Development, and Operation of Flight Systems, Revision G, 2016.

[14] NASA, NASA SP-2016-6105 Rev2 - The NASA Systems Engineering Handbook, 2016.

[15] NASA, NASA-HDBK-1005 - NASA SPACE MISSION ARCHITECTURE FRAMEWORK (SMAF) HANDBOOK FOR UNCREWED SPACE MISSIONS, 2021.