Metadata of the chapter that will be visualized in SpringerLink

Book Title	Knowledge and Systems Sciences
Series Title	
Chapter Title	Mission Modeling for the Perseverance Rover Based on KARMA Language
Copyright Year	2025
Copyright HolderName	The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd.

Author	Family Name	**Liu**
	Particle	
	Given Name	**Z.**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Industrial and Systems Engineering Laboratory, Beijing Institute of Technology
	Address	Beijing, 100081, China
	Email	3120220369@bit.edu.cn
Author	Family Name	**Wang**
	Particle	
	Given Name	**G.**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Industrial and Systems Engineering Laboratory, Beijing Institute of Technology
	Address	Beijing, 100081, China
	Email	
Corresponding Author	Family Name	**Ma**
	Particle	
	Given Name	**J.**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Industrial and Systems Engineering Laboratory, Beijing Institute of Technology
	Address	Beijing, 100081, China
	Division	
	Organization	Yangtze Delta Region Academy of Beijing Institute of Technology
	Address	Jiaxing, 314019, Zhejiang, China
	Email	mjd2015@sina.cn

Author	Family Name	**Lu**
	Particle	
	Given Name	**J.**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Beihang University
	Address	Beijing, 100191, China
	Email	jinzhl@buaa.edu.cn
Author	Family Name	**Dong**
	Particle	
	Given Name	**M.**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Industrial and Systems Engineering Laboratory, Beijing Institute of Technology
	Address	Beijing, 100081, China
	Email	

Abstract	Over the last two decades, Model-Based System Engineering (MBSE) has been progressively introduced into the design of complex systems as an essential method, aiming to address the complexity management challenges in traditional document-based approaches. However, limited by a lack of diversity in project and constraints on knowledge, the application of MBSE faces several issues. These include insufficient hierarchical decomposition of missions, overly abstract meta-models, and inadequate adaptability of methods. When MBSE is applied to the rover, the complexity of the Mars environment and the multidisciplinary nature of the Mars rover's design increase the learning curve for researchers due to the high-level abstraction of the meta-model. This paper proposes a new multi-architecture modeling method based on KARMA language, particularly focusing on the mission and operation stages. Including how to start from client objectives to precisely decompose stakeholder needs and successfully build a conceptual model library, the method supports more accurate and adaptable missions analysis in complex systems. Finally, we use the Perseverance rover design as an example to validate the effectiveness of the proposed method.

Keywords (separated by '-')	MBSE - Mission modeling - Meta model - KARMA Language - Mars rover

Author Proof

Check for updates

Mission Modeling for the Perseverance Rover Based on KARMA Language

Z. Liu[1], G. Wang[1], J. Ma[1,2](✉), J. Lu[3], and M. Dong[1]

[1] Industrial and Systems Engineering Laboratory, Beijing Institute of Technology, Beijing 100081, China
3120220369@bit.edu.cn

[2] Yangtze Delta Region Academy of Beijing Institute of Technology, Jiaxing 314019, Zhejiang, China
mjd2015@sina.cn

[3] Beihang University, Beijing 100191, China
jinzhl@buaa.edu.cn

Abstract. Over the last two decades, Model-Based System Engineering (MBSE) has been progressively introduced into the design of complex systems as an essential method, aiming to address the complexity management challenges in traditional document-based approaches. However, limited by a lack of diversity in project and constraints on knowledge, the application of MBSE faces several issues. These include insufficient hierarchical decomposition of missions, overly abstract meta-models, and inadequate adaptability of methods. When MBSE is applied to the rover, the complexity of the Mars environment and the multidisciplinary nature of the Mars rover's design increase the learning curve for researchers due to the high-level abstraction of the meta-model. This paper proposes a new multi-architecture modeling method based on KARMA language, particularly focusing on the mission and operation stages. Including how to start from client objectives to precisely decompose stakeholder needs and successfully build a conceptual model library, the method supports more accurate and adaptable missions analysis in complex systems. Finally, we use the Perseverance rover design as an example to validate the effectiveness of the proposed method. AQ1

Keywords: MBSE · Mission modeling · Meta model · KARMA Language · Mars rover

1 Introduction

MBSE is an advanced theory of systems engineering, which has greatly facilitated the progress of SoS over past two decades. Including modeling method, modeling tool and modeling language as three pillars [1], MBSE significantly enhances design efficiency and management quality [2]. Methods have proven to be powerful ways for addressing complex architectural design challenges. Researchers continuously develop and refine methods to enhance the design, management, and efficiency of complex projects.

X. Tang et al. (Eds.): KSS 2024, CCIS 2269, pp. 1–15, 2025.
https://doi.org/10.1007/978-981-96-0178-3_14

However, existing modeling methods lacks a multi-perspective hierarchical analysis of mission, which often remains at a superficial lever without exploring the underlying business goals, user expectations, and environment factors. The relationship and dependencies are not fully revealed, such as cost overruns or design products not meeting user objectives. Moreover, the high abstract level modeling methods present a learning challenge for personnel from diverse backgrounds. Current high-abstraction meta-models complicate understanding and application. Additionally, existing methods lack of flexibility and adaptability limits their effective response to the unknown variables. As technology rapidly develops, method should also be flexible enough to adapt to new knowledge to support long-term innovation of projects.

The aerospace industry is the first to successfully apply MBSE. However, as we continue to use MBSE, we are finding some challenges in the field of Mars rover development. First, manufacturing a Mars rover is a high-tech process that involves many disciplines and complex technological innovations. This makes the modeling process highly specialized, leading to high learning costs and long learning cycles for non-experts. Second, the Martian environment is highly changeable and largely unknown, with variations in light, radiation, and other cosmic factors. Mars is also much farther from Earth than the moon, making rover development very difficult. Therefore, the design and modeling of the rover need to adapt to the changing Martian environment and must be flexible and robust.

To deal with these issues, this paper first customizes a meta-model library for Mars rover design, then proposes a multi-architecture modeling method based on KARMA language, supporting multi-perspective architectural design of mission and operation, and finally validates the proposed method with the Perseverance rover project.

The rest of the paper is organized as follows: Sect. 2 discusses related work in MBSE modeling methods. Section 3 proposes a multi-architecture mission modeling method for "mission analysis" and "operation analysis". Section 4 presents a modeling example based on the Perseverance rover and discusses it. Finally, Sect. 5 concludes the paper.

2 Related Work

In this chapter, we first summarize the current mainstream modeling methods in MBSE. Then some meta-models approach used in the methods are introduced. Research status of Mars rover design in the field of MBSE is also analyzed later. Finally, based on the related work, we summarize the objectives of this study.

2.1 Current MBSE Methods

Current mainstream modeling methods, such as OPM, STRATA, RFLP, and DoDAF, each have their strengths in mission or requirements analysis but can benefit from enhanced multi-perspective analysis. In OPM [3], requirements are typically modeled as states of objects and connected to components through labels such as "satisfies," "realizes," or "derives." While effective, this approach may benefit from clearer representation of relationships and multi-perspective analysis. Vitech's MBSE modeling methodology STRATA [4] breaks down requirements analysis into four system domains and focuses on the hierarchical decomposition of high-level requirements. However, incorporating a more comprehensive multi-perspective view could further enhance its utility. RFLP [5] (Requirement, Function, Logic, Physical) integrates mission and stakeholder needs at the requirement stage but could more fully reveal the interconnectivity and complexity of different perspectives. These observations suggest that there is room for improvement in multi-perspective analysis in complex system engineering projects.

2.2 Meta-models Used in Methods

Most mainstream MBSE modeling methods are designed for general domains, such as DoDAF and SysML. While powerful in many respects, these methods exhibit a high level of abstraction when applied to the Mars rover field, posing significant learning challenges for engineers. For example, DoDAF 2.0 is based on the UPDM [6] meta-model library, and MagicGrid refers to the SysML 1.0 [7] meta-model library. These general frameworks often do not fit well with the specific complexities and unique missions. Additionally, modeling tools like Capella, which supports the Arcadia method based on DSML [8] meta-models and has been successfully applied in specific domains like avionics and rail transportation, are not suitable for direct application in other field.

2.3 Mars Rover Design Using MBSE

In the field of Mars exploration, the flexibility and adaptability of MBSE modeling methods are crucial. Although methods like DoDAF and SysML demonstrate strong capabilities in multiple domains, they show certain limitations in the specific complex environment of Mars exploration. For instance, the NASA Perseverance rover project [9], despite widespread application of MBSE theory throughout the project's management process, the specialized nature of its modeling processes has resulted in high learning costs, particularly for non-specialists. This not only limits the widespread adoption of the methodology but also slows its practical application efficiency in Mars rover projects. Similarly, the ExoMars [2] mission collaboration between the European Space Agency (ESA) and Roscosmos shows that SysML-based modeling work failed to fully cover the system model, limiting comprehensive system analysis of complex missions. Additionally, Chinese scholars, such as Gao Jinyan [10], have achieved some success in modeling the maintenance and management systems of Mars rovers, but still lack a complete architectural design for the entire lifecycle of Mars rover projects, reflecting the existing methodologies' insufficiencies in supporting comprehensive system engineering. These examples illustrate that existing

MBSE modeling methods need to enhance their flexibility and adaptability to better address the specific challenges of Mars exploration, ensuring the effectiveness of design solutions and supporting the long-term success and innovation of projects.

2.4 Summary and Motivation

From the above literature review, we summarize the main motivations and contributions of this study: We provide a comprehensive MBSE method that includes multi-level analysis for Mars rover missions, addressing the gaps in existing methods. We make the MBSE method more accessible for engineers by reducing the abstraction levels of meta-models, thus lowering the learning difficulty. We enhance the flexibility and adaptability of MBSE methods to better handle the uncertainties and technological changes specific to the Martian environment.

This study is motivated by the need to overcome these limitations. We aim to develop a more accessible and flexible MBSE approach that better suits the complexities of Mars exploration, ultimately enhancing mission success and adaptability in an ever-evolving technological landscape.

3 A Mission Modeling Method Based on KARMA Language

The following is the framework of research in this article. From top to bottom, it is Meta-metamodel, Metamodel, System model, Physical world. In Meta-metamodel hierarchy, GOPPRR-E(Graph, Object, Property, Point, Relationship, Role and Extension) is a kind of meta modeling language. In Metamodel hierarchy, based on GOPPRR-E, we use modeling language KARMA to create the domain metamodel in order to express propriety concepts, such as mission task, operation task, mission phase, operation phase. In System model, we develope a set of viewpoints, divided into mission phase and operation phase, and based on the meta-model, according to each step of the method to build the model. In Physical world, MBSE models map to real-world objects, such as the Perseverance rover, the Zhu Rong rover (see Fig. 1).

3.1 GOPPRR-E Meta Modeling

Kelly [11] synthesized several widely used meta-meta models from existing metamodeling languages, such as ARIS, Ecore, GME, GOPPRR, and MS DSL Tools, among which GOPPRR stands out for its strong descriptive capabilities supporting more complex model concepts. In systems engineering practice, Ding [12] et al. extended the GOPPRR meta-meta model to GOPPRR-E, where 'E' stands for Extension. GOPPRR-E consists of Graph, Object, Point, Property, Relationship, Role, and Extension, each representing fundamental elements and extended derived concepts, detailed in paper [12]. This chapter utilizes the GOPPRR-E meta-modeling method, customized through the KARMA language, to develop a meta-model library specifically for the Mars rover domain.

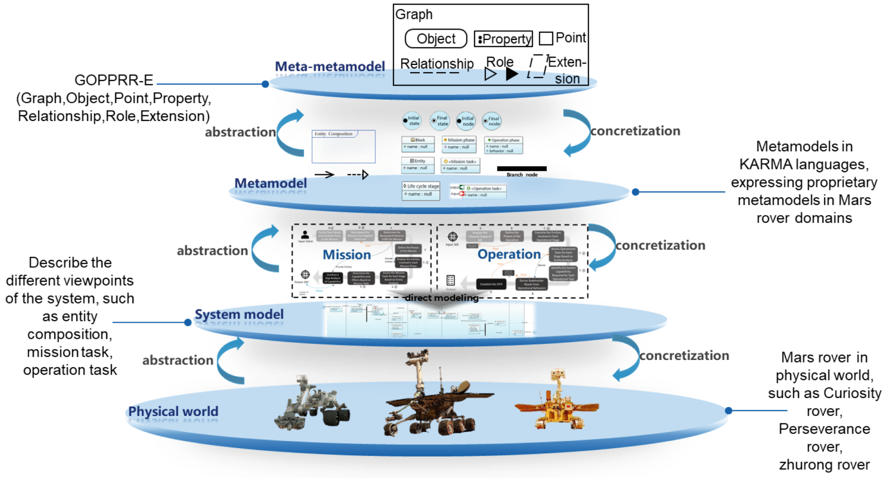

Fig. 1. Technical architecture

3.2 KARMA Language

Based on GOPPRR-E meta modeling, we use KARMA language to create metamodel in specific domain. KARMA [13], a multi-architecture unified modeling language for model-based systems engineering, supports architecture-driven design, code generation, dynamic behavior description, performance analysis, verification, and numerical analysis. It visualizes data as tables and Gantt charts while offering 2D and 3D modeling capabilities. KARMA's nature focus on "text-readable formalized language" to model requirements, functions, logic, and architectures. It enables to verify performance metrics and requirements via simulation and testing, enabling early formalization and validation of system views and reducing development costs and risks during conceptual design.

3.3 Mission Modeling Using KARMA

Metamodel in Specific Domain. Based on GOPPRR-E, We use KARMA language to create metamodel. Table 1 gives a summary of metamodel.

Methods. The method points out the modeling sequence, with each box in the method representing the viewpoints. Each viewpoint is instantiated with a metamodel or other, such as requirements forms(ReqIF), mTable and tables. The design modeling method based on KARMA language is divided into mission phase and operation phase.

Mission Phase. In the mission analysis phase, the objective is to define the problem to be solved, describing the business and problem space. This phase

Table 1. Metamodel in mission phase and operation phase.

Metamodel type	Metamodel example	Quantity
MetaGraph	Mission concept diagram,Task diagram, Life cycle diagram,Entity diagram, Operation concept diagram	5
MetaObject	Entity,Block,Mission phase,Mission task, Operation phase,Operation task,Life cycle stage	7
MetaProperty	Name,Behavior	2
MetaPoint	Input,Output	2
MetaRelationship	Directed aggregation,Object flow,Control flow, Sequential flow,Message flow	5
MetaRole	Beginning with no style,Beginning with dot diamond, Beginning with solid diamond,End with arrow	4

begins with "mission-client-objective" as inputs, where the mission is defined as a set of objectives to be achieved, stakeholders are entities related to but not directly involved in achieving the mission, and the client represents a specific category of stakeholders. The first step in the analysis is to define stakeholders, entities, and operational systems to determine the participants and relevant parties for subsequent analysis. Following this, the mission is decomposed to identify mission phases based on a time sequence, to understand the components of the mission thoroughly. This process includes defining transition events between mission phases, which are necessary conditions for moving from one phase to another, thereby building a conceptual framework for the mission. Further, entities, operational systems, and clients involved in each mission phase are analyzed to clarify their respective tasks, refining the mission phases into specific mission tasks and analyzing the capabilities and expected outcomes for each task. Finally, the operational system versions are determined, and their capability coverage assessed to provide a basis for defining the operation system in the next phase. Mission phase analysis is a system-level approach designed to move from an abstract mission to precisely identify the specific operational systems involved in the mission implementation (Fig. 2). AQ2

Operation Phase. The core purpose of the operation analysis phase is to explore in detail the interactions between external entities and the system of interest (SOI), defined by the mission phase as the operational system. This stage begins with a comprehensive consideration of the SOI's lifecycle activities, covering all aspects from design and development to application and eventual decommissioning. The SOI is then positioned within the entire mission task, analyzing its operational activities and the conditions and triggering events between these activities, thus constructing the operational concept of the SOI. The in-depth analysis during this phase associates operational activities with mission tasks to establish the specific contributions of the SOI to the mission. These operational activities are traced back to lifecycle stages of the SOI, providing guidance for implementation activities. Moreover, identifying the external entities involved

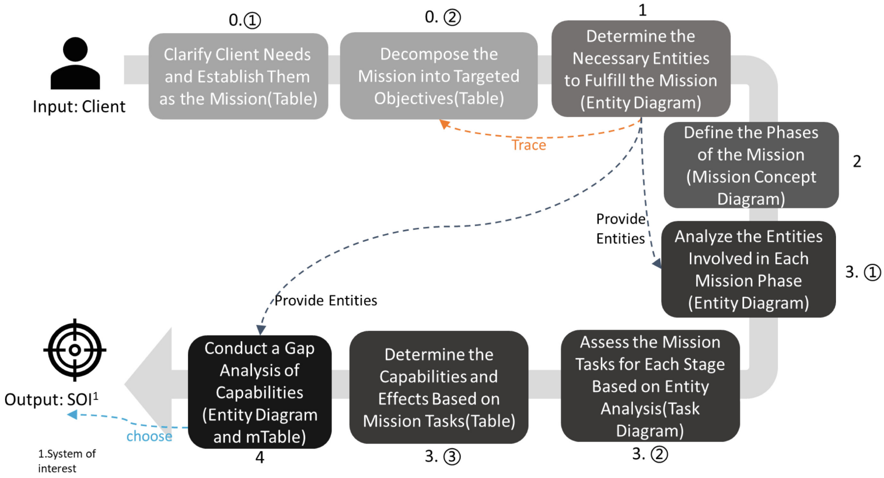

Fig. 2. Mission phase

in each operational stage provides clear definitions for the implements of operational tasks. Then, defining the operational tasks executed by the operational system and external entities in each operational phase lays the foundation for deriving stakeholder requirements. As operational tasks give rise to stakeholder requirements, each requirement corresponds to one or more operational tasks, ensuring that the requirements can be traced back to the capabilities and effects of the mission. Ultimately, creating Operational Task Realization (OTR) and tracing it back to related operational tasks ensures that an operational task may be associated with one or more OTRs, while ensuring that OTRs provide accurate information to the functional architecture phase (Fig. 3).

4 Case Study

4.1 Overview

The book "Perseverance and the Mars 2020 Mission" by Manfred [14] provides a detailed account of the Perseverance Rover's construction background, mission tasks, architectural design, landing site selection, and implementation process. Using the rich information from this book, we conduct a requirement design modeling of the Perseverance Rover. Initially, referencing the SysML meta-model library, a domain-specific meta-model library is constructed. Subsequently, based on the "mission and operation" requirement modeling methodology proposed in this paper, a model library focuses on the Perseverance Rover was developed. This model library not only includes the design, manufacturing, assembly, testing, and transportation processes of the Perseverance Rover but also encompasses the functional requirements of the Perseverance Rover in the Mars exploration mission.

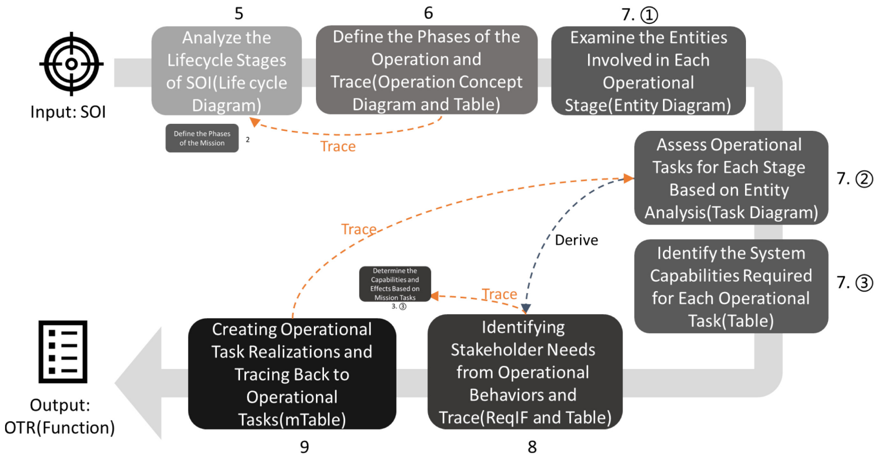

Fig. 3. Operation phase

4.2 Perseverance Rover's Mission Modeling

Step 0 (Client Input Mission). Using the Metagraph tool's table, customer needs for the "Exploring Mars" mission are analyzed based on government documentation. The mission decomposition targets four objectives: determining Mars' ability to support life, understanding the development and history of Mars' climate, comprehending the origins and evolution of the Martian geological system, and preparing for human exploration. These four objectives are specific breakdowns of the top-level mission "Exploring Mars," meaning that achieving all four objectives would signify the mission's completion (see Fig. 4(a)).

Step 1 (Determine the Necessary Entities to Fulfill the Mission). Considering the entities required to accomplish the "Exploring Mars" mission includes the "Perseverance Rover," "Atlas V-541 rocket," "Mars 2020 mission team," "JPL team," "Kennedy Space Center," "Mars Reconnaissance Orbiter," and "launch site operators" among others-totaling 22 entities. Steps 3.① and 7.①, based on Step 1, select entities that must not exceed the range defined(see Fig. 4(b)).

Step 2 (Define the Phases of the Mission). Initially, the mission "Exploring Mars" is analyzed to determine specific implementation stages, dividing it into six mission phases along with the initial and final states. The transitions between each phase, such as from the initial state to the early stages of the project being triggered by the project's commencement and from early preparation to the Atlas V-541 rocket launch triggered by the launch window, are defined(see Fig. 4(c)).

Step 3. ① (Analyze the Entities Involved in Each Mission Phase). This step involves analyzing the entities associated with the six mission phases using the cloning feature in the tool (where all cloned entities share all properties at the base level), selecting relevant entities from step 2. For instance, the mission phase 2 "Atlas V-541 Rocket Launch" involves entities such as Launch complex 41 at cape canaveral air force station, launch operators, and Atlas V-541 rocket(see Fig. 4(d)).

Step 3. ② (Assess the Mission Tasks for Each Stage Based on Entity Analysis). Based on the entities from Step 3.①, the mission tasks for phase 2 "Atlas V-541 Rocket Launch" are decomposed, starting with Launch complex 41 at cape canaveral air force station performing task T.2.1 Provide launch site for Atlas V-541 rocket, followed by the Launch site operators performing task T.2.2 Give launch order, and finally the Atlas V-541 rocket performing task T.2.3 Launch, concluding the task decomposition for mission phase 2(see Fig. 4(e)).

Step 3. ③ (Determine the Capabilities and Effects Based on Mission Tasks). All tasks of the Atlas V-541 rocket launch are summarized in a table, individually analyzing the capabilities and effects corresponding to each mission task, such as "T.2.1 Provide launch site for Atlas V-541 rocket" corresponds to the capability "launch Complex 41 provides the facilities," with the effect "launch -Complex 41 provides the well-established facilities" (see Fig. 4(f)).

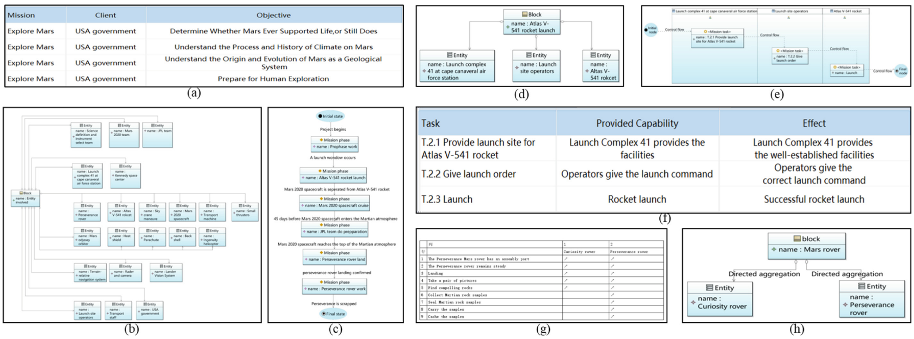

Fig. 4. Perseverance rover models (a) Mission, Client, Objectives (b) The relevant entities involved in the Mars mission (c) Mission concept (d) Atlas V-541 launch mission phase involves entities (e) Analyze mission task (f) Mission capability and capability effect (g) Identify potential operational systems (h) Curiosity versus Perseverance models

Step 4 (Conduct a Gap Analysis of Capabilities). Initially, based on the design history of the Mars rovers, the historical and latest versions of the rovers are analyzed, such as the latest version being the Perseverance Rover, with historical versions including Curiosity, Opportunity, and Spirit. Then, using the mTable relational mapping function provided by Metagraph, the mapping relationship between Curiosity and Perseverance in terms of mission tasks is established, demonstrating the capability gap between them, indicating Perseverance's full coverage of execution capabilities, based on which Perseverance is identified as the SOI for this analysis, being the focus and center of the analysis(see Fig. 4(g) and Fig. 4(h)).

Step 5 (Analyze the Lifecycle Stages of SOI). This step marks the beginning of the operation phase, starting with the Perseverance Rover identified in the mission phase as the System of Interest (SOI). The life cycle of the Perseverance Rover is decomposed, including stages such as the design, production, and manufacturing of the rover, assembly and testing, arrival at the launch site, successful landing on the Martian surface, and participation in Mars exploration missions. Transition conditions are added at each stage, specifying that the Perseverance Rover progresses to the next life cycle stage only when conditions are met(see Fig. 5(a)).

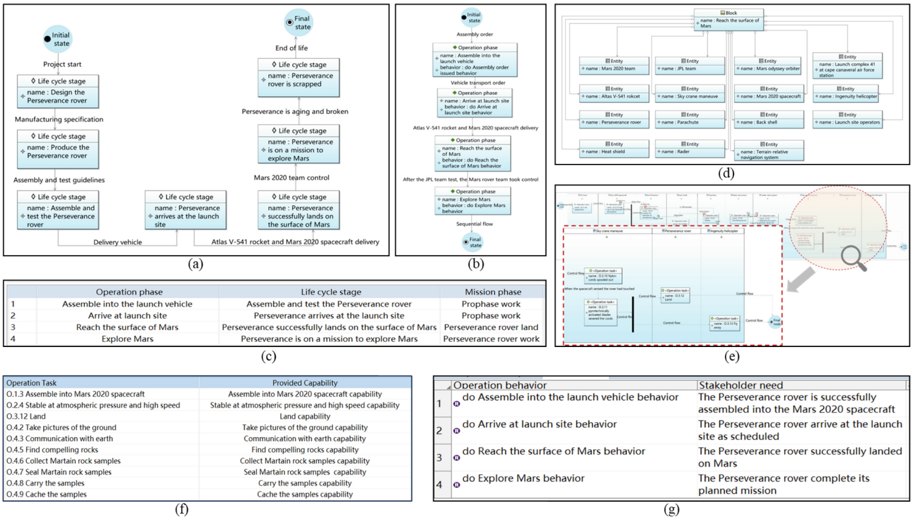

Fig. 5. Perseverance rover models (a) The perseverance lifecycle (b) The perseverance operation concept (c) Operation phase traceability to Lifecycle stage and mission phase (d) Operation phase 3 involved entities (e) Operation phase 3 operation tasks

Step 6 (Define the Phases of the Operation and Trace). This step involves defining the main operational phases of the Perseverance Rover in the context of exploring Mars, including assembling into the launch vehicle, arriving at launch site, reach the surface of Mars, and exploring Mars. The operational phases are traced back to both life cycle stages and mission phases, illustrating the relationship between the Perseverance Rover's operational and mission stages. Each operational phase includes phase names and operational behaviors, with operational behaviors providing a verb-oriented description for inputs into Step 8 (see Fig. 5(b) and Fig. 5(c)).

Step 7. ① (Examine the Entities Involved in Each Operational Stage). Taking operational phase 3 as an example, the entities involved include the Mars 2020 mission team, JPL team, Mars Odyssey Orbiter, parachutes, and the Ingenuity helicopter, among others(see Fig. 5(d)).

Step 7. ② (Assess Operational Tasks for Each Stage Based on Entity Analysis). Based on the entities involved in operational phase 3, operational tasks are analyzed. For instance, at the end of the task sequence, a sky crane performs "O.3.10 Nylon Cords spooled out," followed by the trigger event "When the spacecraft sensed the rover had touched," after which the sky crane performs the next task "O.3.11 Pyrotechnically activated blades severed at cords," and subsequently the Perseverance Rover and Ingenuity perform "O.3.12 Land" and "O.3.13 Fly away" respectively (see Fig. 5(e)).

Step 7. ③ (Identify the System Capabilities Required for Each Operational Task). As shown in Fig. 16, this step involves first identifying the operational tasks related to the Perseverance Rover, then analyzing the system capabilities required for these tasks(see Fig. 5(f)).

Step 8 (Identifying Stakeholder Needs from Operational Behaviors and Tracing Back to Mission Capabilities and Effects). From the operational behaviors identified in Step 6, stakeholder needs are derived, typically describing the specific execution effects of the operational behaviors. These stakeholder needs are then traced back to the mission capabilities and effects identified in Step 3.③, ensuring that the Perseverance Rover's capabilities satisfy the stakeholder needs, and verifying the completion of implementing stakeholder needs(see Fig. 5(g) and Fig. 6).

Step 9 (Creating Operational Task Realizations and Tracing Back to Operational Tasks). Operational Task Realization (OTR) describes the completion effects of operational tasks and also serves as the output of the operational phase, providing inputs to the architectural design of the Mars rover as a basis for fundamental functionalities. This step ensures that each operational task may be associated with one or more OTRs, simultaneously ensuring that OTRs provide accurate information to the functional architecture phase (see Fig. 7).

	Stakeholder need	1	2	3	4
Mission cpability effect		The Perseverance rover is successfully assembled into the Mars 2020 spacecraft	The Perseverance rover arrive at the launch site as scheduled	The Perseverance rover successfully landed on Mars	The Perseverance rover complete its planned mission
1	Perseverance was successfully assembled into the Mars 2020 spacecraft	↙			
2	The Perseverance rover remains steady		↙		
3	Successfully land			↙	
4	Take a pair of picture with the engineering imagers known as Hazard Camera				↙
5	Look for rocks that formed in, or were altered by				↙
6	Drill a core sample				↙
7	Break off the core sample from the rock and cap and hermetically seal it inside the tube				↙
8	Place each sealed tube in a storage rack onboard and transport it until the mission team decides to deposit it on the Martian surface				↙
9	Put the Martian samples in the same place on the Martian surface so that a future mission could potentially retrieve and return them all together				↙

Fig. 6. Stakeholder need trace to mission capability effect

	Operation task	1	2	3	4	5	6	7	8	9	10
OTR		O.1.3 Assemble into Mars 2020 spacecraft	O.2.4 Stable at atmospheric pressure and high speed	O.3.12 Land	O.4.2 Take pictures of the ground	O.4.3 Communication with earth	O.4.5 Fing compelling rocks	O.4.6 Collect Martian rock sample	O.4.7 Seal Martian rock samples	O.4.8 Carry the samples	O.4.9 Cache the samples
1	Assembly into Mars 2020 spacecraft successfully	↗									
2	Stable at atmospheric pressure and high speed		↗								
3	Land successfully			↗							
4	Take a pair of picture with the engineering imagers known as Hazard Camera				↗						
5	Communicate with earth successfully					↗					
6	Look for rocks that formed in, or were altered by						↗				
7	Drill a core sample							↗			
8	Break off the core sample from the rock and cap and hermetically seal it inside the tube								↗		
9	Place each sealed tube in a storage rack onboard and transport it until the mission team decides to deposit it on the Martian surface									↗	
10	Put the Martian samples in the same place on the Martian surface so that a future mission could potentially retrieve and return them all together										↗

Fig. 7. OTR trace back to operation task

4.3 Discussion

In the case study of the rover system's mission design, the Perseverance model is built using the mission and operation analysis methods proposed in this paper. Compared to previous methods, the proposed approach offers the following advantages:

1 The modeling method proposed in this paper divides requirement modeling into "mission" and "operation" phases, significantly enhancing the depth of mission analysis and the ability to decompose missions on multiple levels. By integrating auto-generated Sankey diagrams, this paper effectively maps the capability needs from mission phase analysis to the final stakeholder needs, demonstrating how the design of the Perseverance covers and meets all stakeholder needs. This approach not only exemplifies the multi-angle analysis of requirements but also highlights the deep exploration of multi-level need decomposition (see Fig. 8(a)).

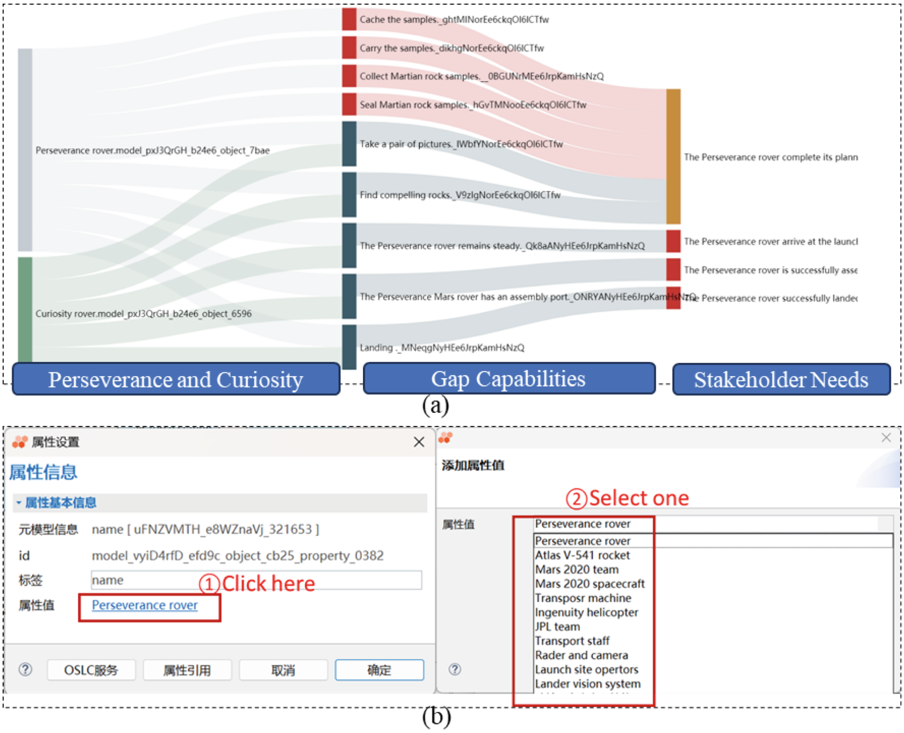

Fig. 8. (a) Stakeholder needs traceability (b)Meta model for entity

2 Facing the high abstraction and generality issues presented by traditional meta-models in Mars exploration projects, this paper adopts innovative steps to reduce these levels of abstraction using KARMA language. KARMA is a language that integrates text and graphics, providing a rigorous mapping between them for enhanced readability and intuitive visualization in meta-model design. Specifically, the meta-model states are detailed into "mission phases," "operational phases," and "life cycle stages," with invocation actions specified as "mission tasks" and "operational tasks." This detailing not only provides a clearer and more operable framework for projects but also makes the mission analysis and design processes more consistent with the actual operational modes of the projects. Moreover, by defining modules as "entities" and pre-recording attributes including names for these entities, such as the Perseverance Rover, Atlas V-541 rocket, and Mars 2020 spacecraft, the abstraction level of the model is further reduced (see Fig. 8(a)).

3 This paper introduces a more flexible modeling framework and dynamic adjustment mechanisms, significantly enhancing the adaptability of the modeling method. Firstly, by dividing the modeling process into multiple stages and defining clear goals and tasks for each stage, it allows for adjustments and optimizations during project implementation based on new information and conditions. Secondly, the use of modular design enables the replacement or updating of modeling components as needed, thereby increasing the model's flexibility and extensibility. Additionally, by specifying the components of the meta-model as entities and pre-recording their attributes, the model becomes more closely aligned with actual application scenarios, further enhancing the adaptability of the modeling method. These innovative measures address the deficiencies in flexibility and adaptability present in existing modeling methods and provide a modeling approach capable of effectively handling unknown environmental variables and technological advancements in Mars exploration projects. AQ3

5 Conclusion

In this paper, we propose a new mission modeling method tailored to the Mars rover system engineering field, utilizing a multi-level modeling approach based on the KARMA language. This method includes modeling both the mission and operation phases and involves the development of a domain-specific meta-model library customized for the Mars rover sector. The feasibility and logical soundness of the proposed method and meta-model are validated using the case study of the Perseverance Rover. This method not only ensures that our modeling framework is directly applicable to the unique missions of Mars rover projects but also demonstrates its effectiveness in handling the complexities associated with such ambitious engineering endeavors.

Acknowledgement. This study was funded by JCKY2021203A001 and D020101.

References

1. Delligatti, L.: SysML Distilled: A Brief Guide to the Systems Modeling Language. Pearson Education, London (2013)

2. Gregory, J.R.: A Model-Based Framework for Early-Stage Analysis of Spacecraft. Ph.D. thesis, University of Bristol (2022)
3. Di Maio, M., Weilkiens, T., Hussein, O., et al.: Evaluating MBSE methodologies using the FEMMP framework. In: 2021 IEEE International Symposium on Systems Engineering (ISSE), pp. 1–8. IEEE, Vienna (2021)
4. Booth, S.T.: Network centric architectures: are we up to the task? In: INCOSE International Symposium, vol. 15, no. 1, pp. 600–609 (2005)
5. Omoarebun, E.N., Cimtalay, S., Mavris, D.N.: Formalizing the decomposition process between elements in the RFLP framework using axiomatic design theory. In: AIAA AVIATION 2023 Forum, p. 3771. AIAA (2023)
6. Hause, M.: The unified profile for DoDAF/MODAF (UPDM) enabling systems of systems on many levels. In: 2010 IEEE International Systems Conference, pp. 426–431. IEEE (2010)
7. Kapos, G.-D., Dalakas, V., Tsadimas, A., et al.: Model-based System Engineering Using SysML: deriving executable simulation models with QVT. In: 2014 IEEE International Systems Conference Proceedings, pp. 531–538. IEEE (2014)
8. Bouba, K., Ait Wakrime, A., Ouhammou, Y., Benaini, R.: A transformation methodology for Capella to Event-B models with DSL verification. J. Comput. Lang. **77**, 101241 (2023)
9. Hill, T.: Small Steps, Giant Leaps: Episode 56, Model-based Systems Engineering. https://www.nasa.gov/podcasts/small-steps-giant-leaps/small-steps-giant-leaps-episode-56-model-based-systems-engineering/. Accessed 24 Mar 2021
10. Gao, J.Y., Wang, L.Y., Pan, Z.S., et al.: MBSE architecture modeling of mars maintenance and management devices. Syst. Eng. Electron. **45**(5), 1441–1450 (2023)
11. Kelly, S., Tolvanen, J.-P.: Domain-Specific Modeling: Enabling Full Code Generation. John Wiley and Sons, Hoboken (2008)
12. Ding, J., Reniers, M., Lu, J., et al.: Integration of modeling and verification for system model based on KARMA language. In: Proceedings of the 18th ACM SIGPLAN International Workshop on Domain-Specific Modeling, pp. 41–50. ACM (2021)
13. Lu, J., Wang, G., Ma, J., et al.: General modeling language to support model? Based systems engineering formalisms (Part 1). In: INCOSE International Symposium, vol. 30, no. 1, pp. 323–338 (2020)
14. Von Ehrenfried, M.: Dutch: Perseverance and the Mars 2020 Mission: Follow the Science to Jezero Crater. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-030-92118-7

Author Queries

Chapter 14

Query Refs.	Details Required	Author's response
AQ1	Please check and confirm if the authors Given and Family names have been correctly identified.	
AQ2	Please check and confirm if the inserted citations of Figs. 2 and 3 are correct. If not, please suggest an alternate citations.	
AQ3	As References [2] and [10] are the same, we have deleted the duplicate reference and renumbered accordingly. Please check and confirm.	