Compaction of Spacecraft Operational Models with Metamodeling Domain Knowledge

Kazunori Someya[1,2], Toshiaki Aoki[2] and Naoki Ishihama[3]
[1]Space Technology Directorate I, Japan Aerospace Exploration Agency, Tsukuba, Ibaraki, Japan
[2]School of Information Science, Japan Advanced Institute of Science and Technology, Nomi, Ishikawa, Japan
[3]Research and Development Directorate, Japan Aerospace Exploration Agency, Tsukuba, Ibaraki, Japan

Keywords: Spacecraft Operation, Metamodel, Compaction, Review, Model-Based Systems Engineering.

Abstract: Due to the difficulty of performing repairs during flight, spacecraft is operated according to operational scenarios tested before launch. Operational models, a type of SysML activity diagram, can be used to depict these scenarios. Although it aids in communication between engineers and stakeholders, the activity diagram can rapidly grow rather large. Due to the extensive operational model, it is therefore challenging to review the activity diagram, which could result in serious issues. Therefore, to make operational models easier to evaluate, they should be made concise. This study offers a metamodel that offers stereotypes that can succinctly characterize spacecraft operational scenarios. First, a mind map was used to depict the domain knowledge of spacecraft operations. Second, stereotype metamodel was created by extracting common knowledge from the mind map. Utilizing stereotypes, operational models' size can be decreased; however, crucial review-related data could be lost. Therefore, shrinking stereotypes assures that crucial information would not be lost. Several trials were conducted and showed that the number of elements of operational models with stereotypes, as well as their size, reduced by almost half, compared with the original ones, allowing for a simplified review process and boosting trust in the accuracy of the operational scenarios.

1 INTRODUCTION

Spacecraft systems require high dependability as repairs during operation in space are challenging to undertake after a rocket launch. All operations are carried out only using operational scenarios tested in advance using system behaviors (Japan Aerospace Exploration Agency [JAXA], 2013; JAXA, 2016). Failure to validate a particular situation could result in unpredictable spacecraft behaviors and, in the worst case, loss of the spacecraft. Operational scenarios should be broken down into specific processes with system design. These designs leverage the model-based systems engineering (MBSE) technique and are represented using SysML (Friedenthal et al, 2014; OMG, 2015) to clearly show traces (Friendenthal and Oster, 2017; Poupart et al, 2012). Using the MBSE, engineers and stakeholders can quickly come to a common understanding through visualization. Additionally, an operational scenario is considered a flowchart represented as an activity diagram using operational models.

An extremely large activity diagram is challenging to review. We ran into this issue when we examined the spacecraft operational scenarios of JAXA produced by the spacecraft manufacturer. A document of the operational design, which included operational scenarios illustrated with activity diagrams, contained approximately 1,000 A4-size pages and hence, was difficult to review. Operational scenarios for spacecraft include large volumes of data that are difficult to remove because they cover everything from high-level concepts to specific instructions. This research focuses on the visualization of pertinent data required to achieve compaction. Particularly, a short compact diagram can make a review easier. Additionally, operational scenarios for a review are often printed on A4-size papers. A broad operational scenario is printed on multiple pages. However, this long description makes reviewing challenging, and failure to conduct

Someya, K., Aoki, T. and Ishihama, N.
Compaction of Spacecraft Operational Models with Metamodeling Domain Knowledge.
DOI: 10.5220/0011839200003464
In *Proceedings of the 18th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2023)*, pages 102-113
ISBN: 978-989-758-647-7; ISSN: 2184-4895

a comprehensive review could compromise reliability.

In this research, we provide operational metamodels that consist of stereotypes and tagged values to express operational scenarios as small diagrams. Particularly, the metamodels were created focusing on the shades of information that are important behaviors, but not significant behaviors in the assessment of operational scenarios. We believed that a particular characteristic of frequent or consistent activities was a small value to depict each time. Additionally, a distinctive feature is that a stereotype in the metamodels defined stereotype's name and a behavior shown in a flowchart. Compaction involves creating a series of common or repeated actions as stereotypes and tagged values and expressing them as one piece. However, some information may be lost because of stereotype compaction. As a result, vital portions of a review should be kept as necessary information, despite the removal of uninteresting information. In order to ensure that the important information is not deleted, we made a validation model which represents information used in the review and confirmed that, for each element of the validation model, there exists an element which corresponds to it in the metamodels.

2 RELATED WORKS

The MBSE technique was introduced by the National Aeronautics and Space Administration (NASA), the European Space Agency, and a Japanese manufacturer of satellites for the spacecraft design; at the moment, the modeling of spacecraft systems is a top priority (Fosse et al, 2015; Feo-Arenis et al, 2015; Tamakoshi et al, 2019). This method is based on systems engineering methodology (Haskin et al, 2011) and is also applied to the creation of spacecraft systems (Friendenthal and Oster, 2017). SysML, as an extension of the UML modeling language (OMG, 2015), is commonly used to create activity diagrams. Modeling a spacecraft system using the MBSE method at a space agency intends to create a system model (Poupart et al, 2012; Fosse et al, 2015). Operational scenarios are modeled based on a concept of operation (ConOps) or high-level activity for system requirement analysis (Herzig et al, 2018; Kaslow et all, 2017). ConOps contains scenarios for all important operational circumstances (NASA, 2016), but does not grow significantly, as no in-depth operational technique is defined. As a result, this research is distinguished by emphasizing on operation models, with a recognition of the challenge presented by a huge model, and by attempting to contribute to spacecraft system development through compression.

UML or SysML stereotypes are prepared in advance as UML profiles and formally published from Object Management Group (OMG) to execute domain-specific modeling. For instance, metamodels or profiles exist for spacecraft operation by SOLM (OMG, 2012) and embedded system by MATRE (OMG, 2011). These UML profiles aimed to increase the amount that can be specified specifically for domains (Selic, 2013). The distinctive point of our method is that stereotypes will be employed for compaction, which is the reverse of extension. Additionally, stereotypes with certain behavior could help shrink the size of activity diagrams through experimentation.

Abstraction is one sort of compaction to condense diagram sizes. An ontology-based compaction technique exists against the backdrop of ever-more complicated models (Guizzardi et al, 2019; Figueiredo et al, 2018). This research does not focus on the rise in huge models as a result of ontology complexity. Additionally, an ontology is challenging to develop. However, an operation must be modeled by outlining the specific design processes. Clarifying the big picture is imperative, regardless of the operation. Our method involves arranging operational domain information using a mind map, for example, clarifying ontology.

Another compression technique is combining identical or related parts into a single element. A technique that mixes similar action components from different scenarios is possible, represented by an activity diagram into a single scenario (Beckman et al, 2017). This technique is effective for reducing the number of scenarios. However, in our operational scenarios, the same elements also emerge in a single scenario, and not only in many scenarios. This strategy makes a scenario review tough due to a generated new loop flow, making the execution order difficult to grasp. Consequently, this does not meet our goal of enhancing scenario reviews. Additionally, no information is lost because this method incorporates equivalent information transformation. Our compaction method does not guarantee equal information transformation. Because of this, our strategy incorporates metamodel validation to protect crucial data after compaction.

3 METHODOOLOGY

We suggest a technique for operational model compaction using stereotypes and tagged values described as metamodels, thereby concealing irrelevant information during operation reviews. Specifically, behavior, event and interface data included in spacecraft operation are clarified using a mind map to visualize domain knowledge before the metamodels are created. Additionally, we ensure that essential information is not hidden during the review.

People involved in spacecraft development share common understanding on the operation and system behavior. Common practices are known information, such as low-value actions in the evaluation. Such actions could increase the size operation procedure diagram. Regarding derivational development, the difference between current systems is more significant than typical practices in a review. The differing points show a high risk of failure. We intend to comprehend the essence of a given behavior, rather than detailed common behaviors. Consequently, typical behaviors, defined by stereotypes and tagged values, are characterized as a metamodel. Additionally, in compaction preparation, we provide a layer to govern the granularity of an operational scenario as a metamodel. Stereotypes outline a series of behaviors in the same layer. The stereotype efficiency drops when many layer behaviors are included in single activity diagram. We also anticipate that the size of one diagram can be reduced by dividing it into layers.

The primary contributions of this study are the size reduction of activity diagrams using the metamodels, concealment of unimportant information, and necessary retention of information, to concentrate on the core of operation behavior. Furthermore, our approach includes a validation procedure for determining whether the metamodels preserve the required data for a review. A compacted operational model helps reduce the number of elements and size of an activity diagram compared to the original activity diagram, thereby enhancing and simplifying a review. As a result, the goal is to keep each scenario to one page so that it can be easily reviewed.

Figure 1 shows an overview of the compaction approach. Using mind maps, this approach starts by arranging and displaying domain information on spacecraft. Although these maps are not exactly specified, they are easy and useful for sharing and comprehending the design and functionality of the entire spacecraft. The operational layer metamodel is developed according to the spacecraft architecture based on system design domain knowledge. This metamodel is referred to as the layer of the operational scenario. It seeks to offer direction and backing for the granularity of the description based on each layer. The operational stereotype metamodel is constructed from using behaviors based on domain knowledge on spacecraft operation. Specifically, a common behavior is defined by selecting a mind map. These two metamodels are collectively referred to as operational metamodels, and a condensed activity diagram is constructed using these metamodels. The SysML requirement diagram is the validation model, and not a metamodel. Certain information might be lost due to compaction. Consequently, a validation procedure for assessing the operational metamodels from the review perspective is needed. The review

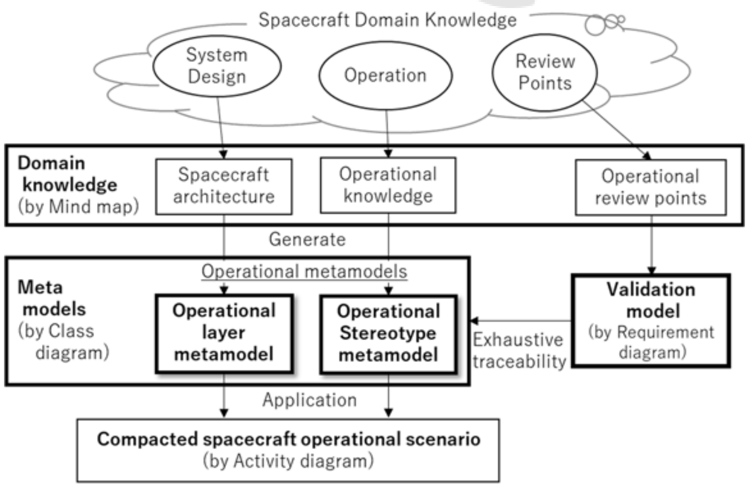


Figure 1: Overview of the relations between operational metamodels.

perspective is taken from review points acquired from engineers and stakeholders using mind maps. These review points are used to abstract and design common objects, which form the review perspective.

4 OPERATIONAL METAMODELS

We suggest creating a compacted activity diagram using the operational metamodels that have two metamodels: the operational layer and operational stereotype. Both metamodels exhibit an activity diagram at their base. This section discusses the operational layer metamodel, operational stereotype metamodel, and validation of both metamodels.

4.1 Operational Layer Metamodel

The operational layer metamodel is intended to govern and guide granularity for applying the stereotype effectively and describes the overall structure of the operational scenario. This metamodel shows a modest direct impact on making the diagram smaller and is used for compaction reprocessing. If the granularity is uneven and the descriptions are mixed among the activities of many layers, replacing one action with a stereotype that matches the series of actions is challenging. This is because a stereotype outlines a string of behaviors in the same layer.

We propose an operational layer metamodel that defines the relations between operational situations that relate to the spacecraft development layer, swim lanes, and actions in an activity diagram using a class diagram (Fig. 2). The layer is described in terms of a general spacecraft system as integrated systems (i.e., as a SoS), system, subsystem, component, hardware, and software. Operational scenarios are designed in synchronization with the system development layer. This metamodel regulates the granularity scenario by referring to the system development layer. A development layer extends up to one layer below the development goal layer. Therefore, swim lanes (i.e., activity partition) of an activity diagram are assigned in one layer beneath the system layer. By fixing actors in place, interactions in a scenario are set according to swim lanes, making inserting the actions of other layers challenging. Consequently, the granularity of operational situations is guided by specifying actors allocated in swim lanes. In this manner, granularity can be guided by arranging actors and actions that match to the layers for each scenario. The metamodel at the top is the layered activity diagram with four layers: a SoS, system, subsystem, and component. Each activity diagram depicts the link of traces between layers in accordance with the system development process. Actors in one-level layer and one layer lower in a swim lane appear in each activity diagram. Additionally, the actions of each actor are placed in the corresponding swim lane. The multiplicity of action elements is set to a number greater than zero.

4.2 Operational Stereotype Metamodel

The operational stereotype metamodel design has two steps to organize the spacecraft operational domain knowledge into categories to find a common behavior and related information, and to define stereotypes and tagged values based on frequent behaviors.

4.2.1 Domain Knowledge of a Mind Map

We compile data kinds and events from the spacecraft operation domain knowledge and visualize them using a mind map (Fig. 3). This study

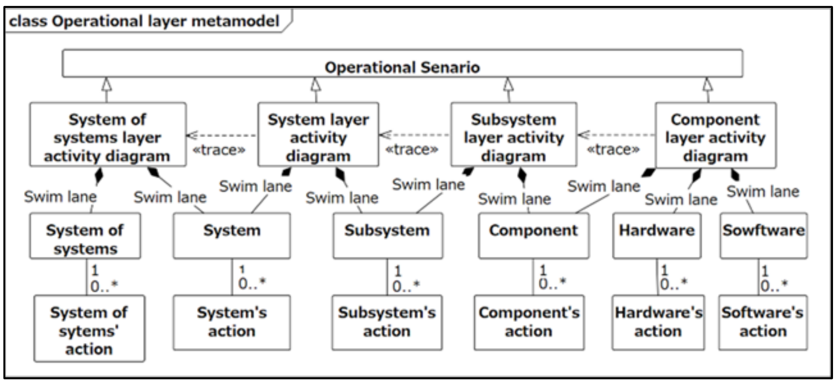


Figure 2: Operational layer metamodel based on the spacecraft structure.

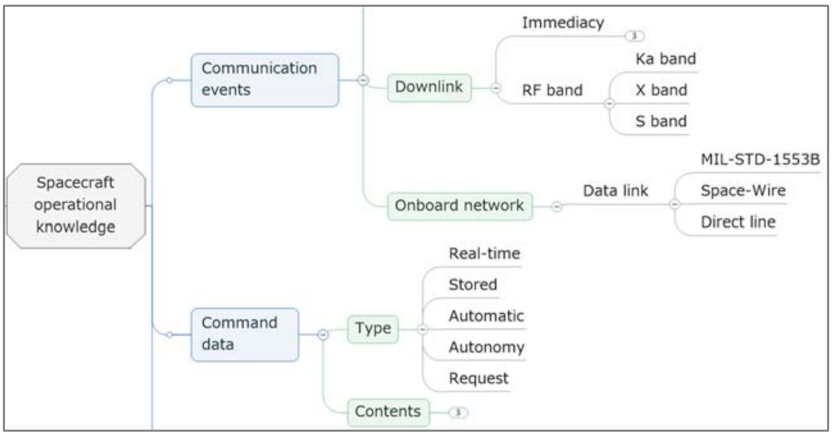


Figure 3: Part of spacecraft operational domain knowledge as a mind map.

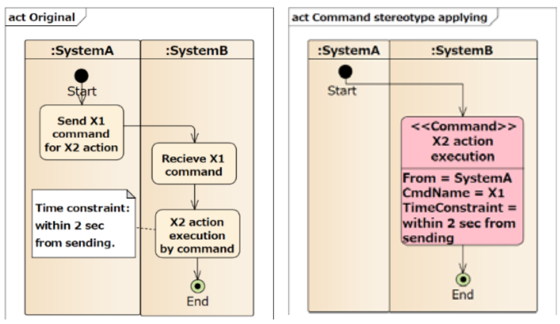
sought to first visualize a standard document for spacecraft operation and then to compile common knowledge through interviews with engineers and stakeholders, as well as from other documents. The standard documents are applicable documents for spacecraft development and describes common actions and data pertaining to all spacecraft activities. These documents are very helpful as they are founded on accepted common knowledge. This idea is intriguing since typical conduct quickly shows up on the mind map. Thus, concealed information on the typical document-based mind map will be spacecraft-specific behaviors or parameters, which should be defined as tagged values and present the condensed activity diagram. A mind map has 58 components.

4.2.2 Stereotype and Tagged value

The conditions of spacecraft common behaviors frequently occur in operational settings, and data do not change between input and output within a sequence. Behaviors that emerge frequently are likely common behaviors, with a strong compactification effect. No data change suggests that this sequence lacks data processing activity for providing unimportant information. For instance, the execution behavior by a command comprises systems A and B sending a command and receiving it, respectively, and then executing it (left side of Fig. 4). Execution by command includes the three actions, namely, transmission, reception, and execution, which are command-related behaviors. As a result, these three behaviors are together classified as one stereotype of “Command” (right side of Fig. 4). The new action that applies the stereotype is set in the swim lanes of command execution because the idea of this sequence is that the execution action is more significant than data transmission and reception activities. However, information regarding the transmission source and command name in this sequence is obscured due to compaction. To provide information, tagged values are made to hold the data.

We suggest an operational stereotype metamodel to shrink the size of operational scenario drawings by excluding irrelevant details (Fig. 5). The stereotype chosen from the mind map by a function of spacecraft operation is that a typical behavior is dependent on the data kind. For instance, the data type of command data shown in Fig. 3 exhibits the typical behavior (Fig. 4). We take advantage of the unique feature of spacecraft system operation that once the data type is determined, the behavior is also uniquely determined.

This metamodel is called the Spacecraft Operation UML Profile. The elements of a stereotype on the metamodel display a hierarchical structure and an inheritance relationship, as indicated by generalization notation. A behavior defined by each stereotype is categorized as parents or children, if the behavior of each element reveals



(a) Original behavior (b) Command stereotype application

Figure 4: Definition of common behavior as flowchart for command stereotype.

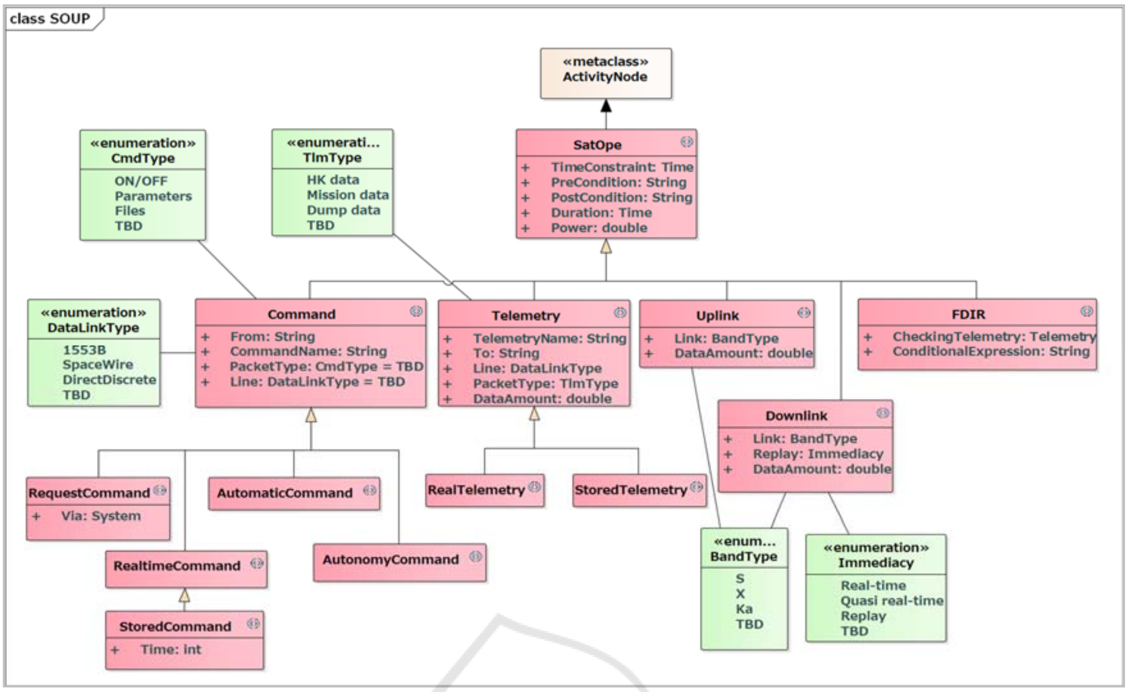


Figure 5: Operational stereotype metamodel.

an inheritance relationship. A stereotype at the top (e.g., SatOpe stereotype presented in Fig. 5) creates elements that are frequently used in each activity (e.g., a time constraint, precondition, postcondition, execution time, and power consumption). Other stereotypes describe parent–child interactions by various common characteristics. For example, a command stereotype is designed from a parent class from other types of command stereotypes. In other words, the command stereotype has a common behavior with other command stereotypes of the child class, which is a receive command from other systems and execution. Particularly, the stored command stereotype adds a timer of a tagged value. A parent class of a real-time command stereotype by a stored command stereotype exhibits the same behavior with the command stereotype. However, a behavior of stored command is different from them, and this data type has a timer for command executing. If the difference in common behaviors exists between parent and child stereotypes, a tagged valued is provided.

Enumeration components are intended for quickly choosing tagged values and controlling the granularity of a scenario. The type of granularity word being used can be understood by preparing the values beforehand. The granularity of a scenario is controlled by a word defined by enumeration elements. For example, the DataLinkType element is an enumeration element that describes the type of internal communication for the spacecraft. This element uses the line tag, designated as the command stereotype's type of DataLinkType. This scenario focuses on the data exchange within the spacecraft, and not the connection between the spacecraft and a ground system. Particularly, this becomes additional knowledge for comprehending the operational context. This data is utilized for the link tag on the command stereotype. Enumeration of DataLinkType is designed by an internal network of data links from the mind map shown in Fig. 3. A data link is always used when transmitting commands, and thus choosing which data link to use is required. Tagged values can help in choosing the right kind of a data link. If this information was a note element, an omission could have occurred. If a crucial tagged value is blank, we advise writing “to be determined” (TBD) to denote an incomplete design. Further, mentioned elements include TBD parameters.

4.3 Validation of Metamodeling

We demonstrate the ability of operational stereotype metamodels to retain the relevant information for a review following compaction. Unnecessary information is concealed via stereotype application. Information may be lost when drawing a compaction activity diagram since some actions are concealed. However, tagged values give hidden information through compaction, which restores required information. Because of this, validation is done to see if tagged values provide the necessary data needed for a review. This necessary information depends on the review perspective of engineers and stakeholders who act as operational scenario reviewers. The tagged values of the operational metamodels are checked to ascertain whether all requirements are satisfied.

The requirement diagram of SysML is used for verification. The review perspective is extracted from review points gathered from engineers and stakeholders and then defined as requirement elements. Validation is performed to check whether the elements of metamodels (e.g., tagged values corresponding to requirements) exist. If a requirement cannot be traced, the requirement is deemed not satisfied because of insufficient necessary information. However, metamodels can be improved if any inadequacy is discovered. They are validated only when all requirements are met.

Figure 6 depicts the validation model used to determine if the components of a metamodel adhere to all requirements from review points. It also displays the proportion of all elements, four components of the review viewpoint, and three

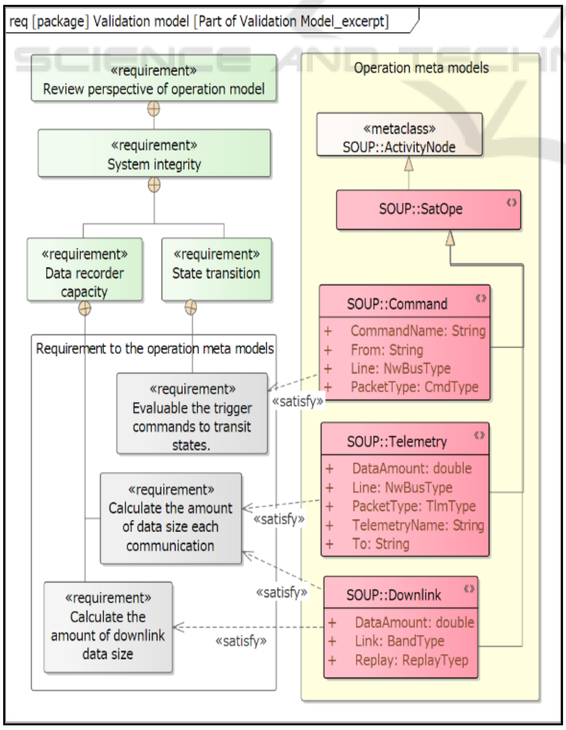


Figure 6: Part of the validation model for the traceability of the operational metamodel.

elements of the requirements for the operational stereotype metamodel. The review perspective is divided into a nesting structure (upper-left side of Fig. 6). The metamodels' requirements are examined and connected as nesting shown in the lower-left corner of the figure. The elements of the metamodels on the right side correspond to the requirements and are connected by a satisfaction connection for satisfying them. For instance, the element of "calculate the amount of data for each communication" necessitates data size between each communication that needs information regarding the amount of data. This is satisfied using the DataAmount tag, which is created using the telemetry and downlink stereotypes to represent the quantity of data.

5 EXPERIMENT AND RESULTS

In the first experiment, we verified the operational stereotype metamodel and confirmed its effectiveness. The real reviewer for the JAXA operation scenario was given from the review perspective. The suggested metamodel was assessed in terms of this review perspective, and no deficiencies were discovered. Subsequently, the proposed compaction method was compared with a traditional approach employed in actual development, and its effectiveness was evaluated. Final tests were carried out by employing the operational metamodels in various contexts to validate the compaction impact. We targeted three actual operational scenarios of the JAXA earth observation satellite. Our goal is to assess the effect of the suggested strategy on ease of reviews. Activity diagrams were made in each layer based on the operation layer metamodel before applying the stereotypes to these diagrams using the operational stereotype metamodel. We also assessed the compaction impact before and after the application of the stereotypes to determine whether any reduction occurred in the number of elements and drawing size (whether it could be fully included on an A4 sheet).

5.1 Validation Operational Stereotype Metamodel

The operational stereotype metamodel was assessed to determine whether relevant information is kept after compaction. We used the requirement diagram to extract requirements for the metamodel from the review perspective. We checked whether elements that correspond to each requirement exist in the metamodel. If it did not exist, this implies that the data needed for the review is lost. The metamodel was improved by adding new stereotypes or tagged values. Table 1 shows the tracing findings among the review perspective, requirements, and components of the metamodel.

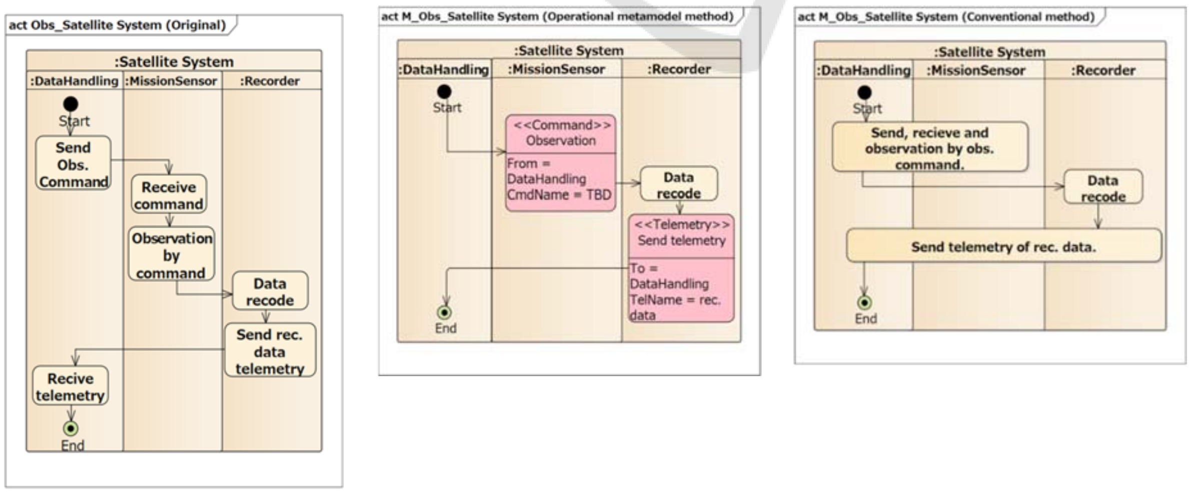
A total of 13 elements are derived from the review viewpoint using requirements for the operation metamodel. For instance, the review perspective of the data recorder capacity was divided into the amount of data size and the link type (i.e., link speed). In particular, 12 of the 13 requirements were satisfied by the elements of the operation metamodel, and the remaining 1 requirement showed no comparable element. Data that corresponds to the execution time of each action was not found. Therefore, by adding a new tagged value so that this information may be kept, the metamodel was enhanced to ensure that the information is saved even after compaction. Because each action time was connected to all stereotypes, we added it as a "Duration" tagged value to SatOpe. The stereotype seen in Fig. 5 was the enhanced version of the metamodel, with the addition of "Duration."

Table 1: Result of the validation of the operational stereotype metamodel.

Elements	Number of elements
Review perspective	10
Requirement for the operational metamodel	13
Satisfy the requirement	12
Found a lack of element in the operation metamodel (i.e., not satisfy the requirement)	**1**

5.2 Comparison Method

We assessed and compared the proposed strategy with a traditional approach (Fig. 7). In this instance, the mission sensor subsystem performs observations using the commands sent from the data handling subsystem. Observation data were captured by the recorder subsystem and were sent to the data handling subsystem. The command and telemetry stereotypes were applied to the original activity diagram in Fig. 7(a) and used to produce the compaction activity diagram. Several acts were swapped out for one action with the stereotypes in Fig. 7(b). The number of pieces was decreased from eight to five, and the length of the drawing size was trimmed. The information obscured by compaction was written down as the associated tagged values. For instance, the execution sequence of observation by a command applies the command stereotype and substitutes it with the action for observation on the swim lane of the mission sensor. The "From" tag on the command stereotype is configured for data handling and instead conceals the sending action. This command stereotype exhibits the "CmdName" tag for the command name as on the display. We discovered that the "CmdName" tag will be empty, and no information regarding the command name is present on the original activity diagram. This tag sets TBD from the enumerated list and denotes an incomplete scenario.



(a) Original activity diagram.

(b) Compaction activity diagram obtained using the operational metamodel.

(c) Compaction activity diagram obtained using the conventional method.

Figure 7: Comparison of the compaction methods in a simple scenario.

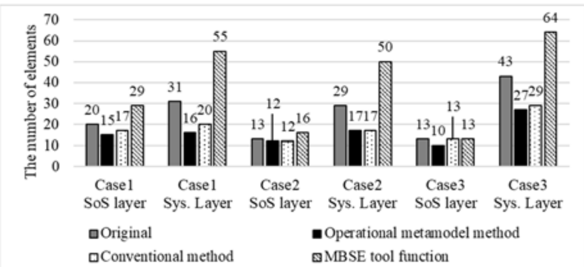
Figure 7(c) displays the outcome of the traditional procedure. The traditional approach combines related activities into one action (i.e., related acts were replaced by one action) without using the stereotypes. This approach is extremely straightforward and is used in practical advancements and operational scenario applications. It describes the contents of several actions by combining or abstracting them into a single action. The substituted action was expressed horizontally for the associated swim lane. Compared to that in Fig. 7(b), the range to be replaced by one action and the total number of elements are the same. However, no other points could achieve a one-to-one correspondence with actions and participants on the swim lane. In the review, a single action should exhibit a relation to one actor, which was also defined by the operational layer metamodel. Here, the mission sensor was not involved in the original action of transmitting telemetry. However, it is connected to the conventional approach used (Fig. 7(c)). The traditional approach worked well when there were nearby swim lanes. However, it could not be expressed well in long swim lanes.

5.3 Actual Operational Scenarios

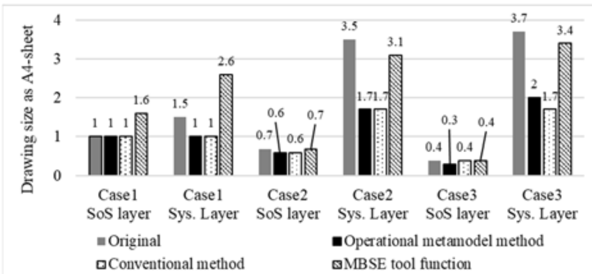
Characteristic scenarios from the actual earth observation satellite of JAXA were chosen as test cases to assess the compaction effect of the three approaches evaluated in the previous section. The three chosen test cases included two scenarios, and six scenarios were targeted at the SoS and system layers. As a result, the same actions were the focus for the application of each compaction method.

Test case 1 exhibits activity diagrams of the observation operation as a typical satellite mission scenario. This study evaluated the compaction effect in a standard scenario. If this scenario has a compression effect, it can be considered effective in other JAXA satellite scenarios. Six action elements are common behaviors in the SoS layer to be the compaction targets. Meanwhile, 18 action elements are common behaviors in the system layer as compaction targets. Test case 2 includes a critical, high-risk operational scenario based on an unstable initial satellite operation after separation from the rocket. Critical operations demanded careful review due to the high risk of spacecraft loss. The applicability of high-risk operations was evaluated. Five action elements are common behaviors in the SoS layer to be the compaction targets. In the system layer, 19 action elements are common behaviors to be the compaction targets. Test case 3 includes a satellite orbit control scenario with the largest number of actions and has a long flow. Longer scenarios were difficult to review. Further, how far a long scenario could be shortened was evaluated. In the SoS layer, one action element is common behaviors to be the compaction target. In the system layer, 23 action elements are common behaviors to be the compaction targets.

Figure 8 presents the compaction effect of the three methods using actual scenarios. The bars indicate the original scenario before compression, proposed method using the operational metamodel, conventional method used in the actual field, and method using the functions of the MBSE tool from the left. The MBSE tool makes use of the call behavior action element defined in SysML, which combines actions into a single element and expresses them through the call function (Aleksandraviciene and Morkevicius, 2018; Sparx Systems central Europe, 2022). Figure 8(a) and (b) present the total number of elements in each scenario and the drawing diagram size, respectively. The operational metamodel method had fewer elements throughout, and the diagram size was the same or smaller compared with those of other methods. Despite a small difference between the number of elements for the conventional and proposed methods, the



(a) Compaction effect using the number of elements.



(b) Compaction effect using the drawing size as an A4 sheet.

Figure 8: Result of compaction using three methods by actual scenarios.

proposed method had the advantage in that it could incorporate the information regarding memo elements included in the scenario in one action element as a tagged value. Diagram sizes were also similar. However, the system layer of case 3 in Fig. 8(b) was smaller than that of the proposed method. This was because the memo element was included in one element, and the action element became large. The conventional method, wherein the memo and action elements are separate, had more flexibility in terms of the element arrangements, and the diagram size was smaller.

The use of the MBSE tool function resulted in more elements and larger diagram size than the original one because multiple actions were combined into one, called behavior action. This action must be redefined in detail in another diagram. The call behavior action was used between layers as an abstraction of the scenario of one layer below, but making it compact in the same layer was difficult. The proposed method has the advantage in that the detailed behavior is not drawn each time because the targeted behavior is a common behavior and is only defined once.

5.4 Compaction Result

The method with the proposed metamodels had the best compression effect from the result presented in Section 5.3. In this section, we analyze the compression effect for the proposed method using the test cases presented in Section 5.3. Table 2 lists the ratios of compaction by the number of elements and drawing size. We found that the effect of compaction in the system layer was larger than that in the SoS layer. The number of elements and diagram size in the system layer was cut in half, and the compaction rate remained nearly constant. Meanwhile, the effects were smaller in the SoS layer than those in the system layer. Although the number of elements was reduced in case 1 of the SoS layer, the diagram size was not reduced. This is because the actions using stereotypes resulted in a larger drawing size due to the tagged values added for display.

Table 2: Effect of compaction.

Layer	Case 1	Case 2	Case 3	**Avg.**
Number of elements				
SoS layer	75%	92%	77%	**81%**
System layer	52%	59%	63%	**58%**
Drawing size as A4 sheet				
SoS layer	100%	86%	75%	**87%**
System layer	67%	49%	54%	**56%**

As we were able to quantitatively demonstrate that the system is compact, we should confirm the ease of review. Although the ease of review is difficult to quantitatively evaluate, we knew that making the diagram smaller from the eye tracking experiment in the study of Lubke et al. (2021) would improve the ease of review. Therefore, we evaluated the effectiveness of this method based on the opinions of key persons for the introduction of our proposed method. The three selected stakeholders were a person with experience in creating operational scenarios, a person in charge of reviewing operational scenarios, and a manager from JAXA. They are key people in deploying this methodology in other spacecraft systems. Although the number of interviewees (i.e., three persons only) may seem small, a strong impetus from a top-down approach by key persons is effective for the penetration of new model-based methods (e.g., the proposed method and MBSE). Therefore, evaluation results should be obtained from key stakeholders.

These three persons were asked to check the compact before and after and were interviewed based on three points: whether it is easy to review, whether their review perspectives are drawn using the diagram, and whether they have any concerns about the compact. The reviewers stated that the activity diagram became easier to view and evaluate because it became smaller and simpler and only focused on important operational behaviors. For the retention of the necessary information, no answer was received because any information for review was missing. One interesting reviewer responded that the time constraints about a scenario they could not review were because time information was not displayed, and the time-tagged value was indicated as a TBD. Time information was not hidden due to compaction but was excluded in the original scenario. By applying the operational metamodel, the diagram size in this study was reduced while still showing the information necessary for the review, TBD. These tagged values are related to the reviewer's viewpoint, as described under item C in Section IV. Finding missing information depends on the skill of individual reviewers. Many reviewers can use a tagged value, guided by TBD, to find missing information.

6 DISCUSSIONS

Particular benefits have been revealed through the experiments. First, the proposed compaction method that uses the metamodel with stereotypes is superior to the conventional method and a call behavior action used based on the MBSE tool function. In the conventional method, when actions that span distant swim lanes were combined into one, the relationship between actions and actors was broken to a one-to-one relationship. Here, we achieved compaction without breaking the relationship of the original scenario, even if it was combined into one action by defining the behavior with stereotypes. Additionally, by introducing a mechanism to validate the stereotypes and tagged values based on the review perspective, clarifying the information that should not be hidden during compaction was possible by tracing the requirement diagram. In this study, the reviewers were interested in the operational scenario and found that they did not want time information to be hidden during compaction. However, they realized that the metamodel did not support this. Therefore, we improved the operational stereotype metamodel based on the validation result. Particularly, a compressed activity diagram that contains the necessary information for a review can be created.

Second, the operational scenarios as activity diagrams were transformed into compact drawings by applying the operational metamodel. Specifically, the application of stereotypes with commonly recognized behavior in an activity diagram reduced the number of elements and activity diagram size compared to the original diagrams (Table 2). The flow of the scenario is much easier to understand by printing the diagrams on one A4 sheet. An activity diagram of two or more pages can be reduced to one page by compaction. The compaction effect is particularly high in the system layer scenario (Fig. 8). Twice as many applied stereotypes in the system layer are found compared to in the SoS layer. Particularly, the command stereotype entailing the known sequence of the three steps of transmission, reception, and execution can be merged into one element with the stereotype, reducing the number of elements to one-third. In this experiment, the command stereotype was applied many times; even general operational scenarios include this command operation. Therefore, we believe that high compaction can be expected even when our proposed method is used in other scenarios of JAXA's satellite operation design document mentioned in Section 1. Additionally, a similar effect can be expected in other spacecraft operational scenarios.

We demonstrated an example that the review of an operational scenario becomes easy when the scenario is fitted in one A4 sheet. However, the complete image becomes challenging to see if it is displayed in a huge sheet. Therefore, we believed that the size of a scenario diagram should be smaller to help ease the reviews. Some of the interviewed reviewers claimed that the suggested procedure eased the review, and we think that this is empirically true. However, simplifying reviews is not limited to reducing the number of elements and realizing an A4-size document. As a result, we will eventually gather quantitative data on simple reviews.

7 CONCLUSIONS

We proposed operational metamodels that comprise layers and stereotypes defined to reduce the size of activity diagrams using domain knowledge. We also developed a mechanism to validate these metamodels and confirmed that relevant information can be retained by fine-tuning after compaction. To solve the problem of large-scale activity diagrams used in presenting operational scenarios during reviews, this study achieved a compact presentation by hiding unimportant information using operational metamodels. Consequently, an actor in an activity diagram is defined according to the layer defined by the operation layer metamodel, the number of elements in an activity diagram could be reduced, and some elements were consolidated into one element with a stereotype. Thus, the diagram becomes compacted and is reduced down to the target, that is, a one-page document. Therefore, we demonstrated that using the activity diagrams with our stereotypes could make the review of operational scenarios easier.

In subsequent work, we will continue to assess whether a review may be made better. The three engineers in this study responded that the condensed operational model enabled an easier review. However, our study only included a few cases, and the interview results are qualitative. We think that more quantitative analysis and experiments are required. For instance, we may collect some mistakes related to the review perspective and incorporate them in the compacted operational and original models. We could also assess the number of errors originated from which model in the experiment. If relevant information is kept in the condensed operational model, more faults are anticipated to be discovered because activity diagrams became easier to review using our suggested strategy.

REFERENCES

Aleksandraviciene, A. and Morkevicius, A. (2018). *MagicGrid Book of Knowledge*, No Magic, Inc.

Beckman, M., Michalke, V. N., Vogelsang, A. and Schlutter, A. (2017). *Removal of Redundant Elements within UML Activity Diagrams.* Proceedings ACM/IEEE of 20th International Conference on Model Driven Engineering Languages and Systems, Austin, Texas, USA, 334-343.

Feo-Arenis, S. and Teraillon, J. (2017). *The SAVOIR Road Map of Model-Based Avionics*. ESA Workshop on Avionics Data, Control and Software Systems (ADCSS2017), Noordwijk, Netherlands.

Figueiredo, G., Duchardt, A., Hedblom, M. M. and Guizzardi, G. (2018). *Breaking into Pieces: An Ontological Approach to Conceptual Model Complexity Management*, Proceeding of IEEE 12th International Conference on Research Challenges in Information Science (RCIS 2018), Nantes, France.

Fosse, E., Devereaux, A., Harmon, C. and Lefland, M. (2015). *Inheriting Curiosity: Leveraging MBSE to Build Mars 2020.* Proceedings of AIAA SPACE 2015 Conference and Exposition, California, USA.

Friedenthal, S., Morre, A. and Steiner, R. (2014). *A Practical Guide to SysML The Systems Modeling Language (3rd. Ed.).* Morgan Kaufmann.

Friendenthal, S. and Oster, C. (2017). *Architecting Spacecraft with SysML A Model-Based Systems Engineering Approach.* Createspace Independent Pub.

Guizzardi, G., Figueiredo, G., Hedblom, M. and Poels, G. (2019). *Ontology-Base Model Abstraction.* Proceeding of IEEE Thirteen International Conference on Research Challenges in Information Science, Brussels, Belgium.

Haskins, C., Forsberg, K., Kureger, M., Walden, D. and Hamelin, R. (2011). *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities.* International Council on Systems Engineering (INCOSE), v3.2.2.

Herzig, S. J, Velez, D., Nairouz, B., Weatherspoon, B., Tikidjian, R., Randolph, T. M. and Muirhead, B. (2018). *A Model-based Approach to Developing the Concept of Operations for Potential Mars Sample Return*. Proceedings of AIAA Space and Astronautics Forum and Exposition, Florida, USA.

Japan Aerospace Exploration Agency (JAXA). (2013). *Spacecraft Design Standard*, JERG-2-000A.

Japan Aerospace Exploration Agency (JAXA). (2016). *Software Development Standard for Spacecraft*, JERG-2-610A.

Kaslow, D., Ayres, B., Cahill, P. T., Hart, L. and Yntema, R. (2017). *A Model-Based Systems Engineering (MBSE) Approach for Defining the Behaviors of CubeSats.* Proceedings of 2017 IEEE Aerospace Conference, Montana, USA, 1-14.

Lubke, D., Ahrens, M. and Schneider, K. (2021). *Influence of Diagram Layout and Scrolling on Understandability of BPMN Processes: An Eye Tracking Experiment with BPMN Diagrams.* Information Technology and Management, (2021) 22:99-121, Springer, pp. 99-131.

NASA Office of the Chief Engineer. (2016). *NASA Systems Engineering Handbook*, NASA SP-2016-6105 Rev. 2.

Object Management Group (OMG). (2011). *A UML Profile for MARTE: Modeling and Analysis of Real-Time and Embedded Systems.* ver 1.1.

Object Management Group (OMG). (2012). *Satellite Operations Language Metamodel (SOLM).* ver1.0.

Object Management Group (OMG). (2015). *OMG Unified Modeling Language (OMG UML),* v2.5.

Object Management Group (OMG). (2015). *OMG System Modeling Language (OMG SysML)*, v1.4.

Poupart, E., Charmeau, M. and Cortier, A. (2012). *Modeling Space System to Provide Global Coherency from Design to Operation Phases.* Proceedings of Space Ops 2012 Conference, Sweden.

Selic, B., Gerard, S. (2013). *Modeling and Analysis of Real-Time and Embedded Systems with UML and MARTE*. Morgan Kaufmann.

Sparx Systems Central Europe. (2022). *Activity Diagram*, https://www.sparxsystems.eu/resources/project-development-withuml-and-ea/activity-diagram/ (accessed November 14, 2021)

Tamakoshi, D., Hirayama, Y., Kubota, H. and Inoue, T. (2019). *Application of Model-Based Systems Engineering to Satellite System Development.* Proceedings of 32nd International Symposium on Space Technology and Science, Fukui, Japan, 2019-t-2007.