



Discussion on Design Method of Spacecraft Information Flow Based on SysML

Zheng Qi^{1,2}(✉), Xiuzhi An^{1,2}, Xiongwen He^{1,2}, Panpan Zhan^{1,2}, Yong Xu^{1,2}, Fang Ren^{1,2}, and Jiajin Li^{1,2}

¹ Beijing Institute of Spacecraft System Engineering, CAST, Beijing 100094, People's Republic of China
qzqz365@126.com

² Institute of Telecommunication Satellite, CAST, Beijing 100094, People's Republic of China

Abstract. Aiming at the low efficiency of the traditional design method of spacecraft information flow based on text mode, combing with the method of Model Based System Engineering, this paper presents the design idea of spacecraft information flow modeling. Modeling of spacecraft information flow will speed up the process of the design phase of the spacecraft information system and ensure the reliability and traceability of information flow through model. Model Based System Engineering provides a new idea for the design of spacecraft information flow.

Keywords: MBSE · Information flow · Model

1 Introduction

Because the traditional spacecraft development mode is based on document and the spacecraft development is a system engineering involving multi-disciplinary integration, and designers from different fields focus on different problems, which leads to the problem on understanding and repeated iterations and modifications in the process of products design.

Model based system engineering (MBSE) has many advantages, such as reusability, no ambiguity, easy to understand, easy to copy and spread, etc. With the application in JSC, OSC and JPL related projects and successful application in the system design of CubeSat and Firesat, MBSE has gradually replaced the traditional document based system engineering and played an more and more important role [1, 2].

System modeling language (SysML) comes from unified modeling language (UML), an important embodiment of MBSE, and unifies the modeling language used in the field of system engineering design. SysML uses graphical symbolic language to create the model of the engineering field, instead of the lengthy text expression. SysML can give guidance in requirements expression, analysis, design, verification, validation and other stages of system engineering. With the emergence of SysML, although there are reliable

and unified expression standards in the design phase and simulation phase of complex system represented by spacecraft, the research and expression standards of software design and simulation verification based on information flow are still in blank condition [3, 4].

How to apply the design method of SysML model to the design process of spacecraft information flow is the focus of this paper. This paper briefly analyzes the main process of spacecraft information flow design, the classification and application of SysML Meta model, and puts forward an idea of how to apply SysML Meta model to the design process of spacecraft information flow.

2 Spacecraft Information Flow Design Process

Information flow is a process of information transfer from source to sink with specified flow direction and format. The design of information flow includes the analysis of information requirements (to solve the problem of “what to transmit”), Information network design (to solve the problem of “where to transmit”), transmission protocol design (to solve the problem of “how to transmit”), information processing design (to solve the problem of “how to use”) and information verification (to solve the problem of “how to verify”), the relationship of parts above is shown in the Fig. 1.

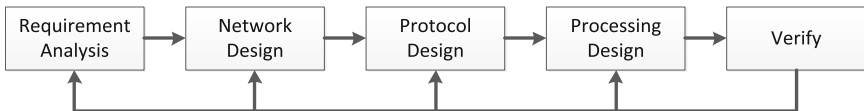


Fig. 1. Process of information flow design

(1) Requirement Analysis

In this stage, user requirements are used as input, information requirements are decomposed, and all functional and performance indicators required by the current information system are defined.

(2) Network Design

In this stage, it needs to know: 1) the source and destination of information in the current scenario; 2) the transmission medium between the source and the destination; 3) the complete information flow path from the source to the destination, and further determine the topology of the information transmission network.

(3) Protocol Design

In this stage, it is necessary to specify the transmission protocol used for information transmission, the internal implementation details of related protocols (Space Packet Protocol, etc.) and the message format related to the protocols.

(4) Processing Design

In this stage, it is necessary to make clear the process of information transmission through information network and after arriving at destination. The process depends on the purpose of the current information.

(5) Verify

In this stage, the test environment and test cases need to be defined. The test cases cover the complete design process of information flow: what information is sent from the source, the flow path of information in the network, whether the information received by the sink is complete and correct, and whether the processing results after the information is received are correct and reasonable. The results of information verification can provide confirmatory feedback for information requirement analysis, network design, protocol design and processing design.

The stage division of spacecraft information flow design process and the clarity of work content provide convenience for SysML modeling design and make the modeling objectives more clear.

3 SysML Modeling Method

3.1 MagicGrid Modeling Framework

The four pillars of MBSE model are requirements, behavior, structure and parameters. Starting from these four pillars, through the construction and iterative analysis of the MBSE system model, the MagicGrid system modeling framework realizes the synthesis of the design steps of the requirements→function→logic→Physics (corresponding to the four pillars of requirements, behavior, structure and parameters respectively) of the system engineering, and provides a standardized basis for the SysML modeling of information flow [5] (Fig. 2).

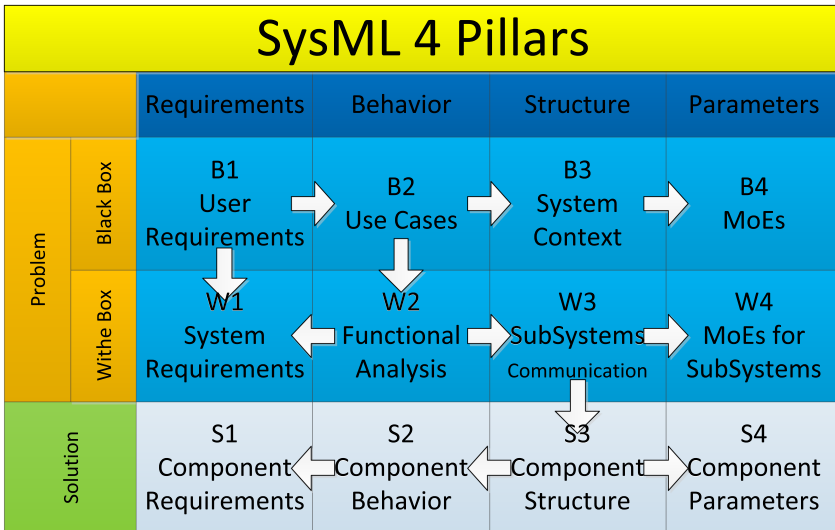


Fig. 2. Framework of MagicGrid system

3.2 SysML Model

SysML has nine kinds of diagrams (block definition diagram, internal block diagram, parameter diagram, package diagram, requirement diagram, use case diagram, activity diagram, sequence diagram and state machine diagram), which support three forms of system expression: requirement, structure and behavior. The requirement model emphasizes the traceability between requirements and the satisfaction of design to requirements; the structural model emphasizes the hierarchy of system and the interconnection between objects, including class and assembly; the behavioral model emphasizes the behavior of objects in the system, including their activity, interaction and state history. Besides, elements in diagrams are used to model as well.

The relationship is shown in the Fig. 3.

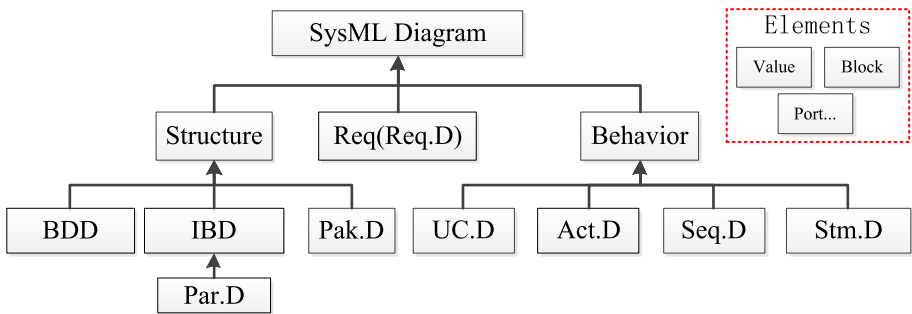


Fig. 3. SysML diagram

(1) Requirement Diagram (Req.D)

This diagram is used to represent text-based requirements and the relationship between requirements. The requirements table can clearly define each requirement of the system using the standard text description. A system requirement created by using a requirement diagram is a description of a system’s functional or non-functional purpose (performance, etc.). The detailed definition of these requirements requires case analysis based on specific system application scenarios. The traceability between requirements and from requirements to system structure and behavior can be modeled by using the requirements diagram. According to the traceability, impact analysis can be carried out on the downstream. When the requirements change, the system structure and behavior that need to be modified can be found easily.

(2) Block Definition Diagram (BDD)

Block definition diagram is used to define the composition relationship between blocks and blocks. It can also be used to refine instances of blocks, including their configuration and value types. The structure attributes of a block mainly include parts, references, and values. The value attribute is defined by the value type of the feasible region, its quantization type and unit, etc. Value attributes can be associated with parameter constraints. Block behavior attributes include operation and

reception, which describe the behavior triggered by the block's response to external trigger.

(3) Internal Block Diagram (IBD)

The internal block diagram of SysML is the product of constraining and extending the composition structure diagram of UML. Internal block diagrams are closely related to block definition diagrams, and various elements can be displayed to illustrate various aspects of the system structure. The abbreviation of internal block diagram is IBD. IBD is created to specify the internal structure of a single module. As a supplement to BDD, IBD can represent the connection between component attributes and reference attributes, the event, energy and data types between the connections, and the services provided and requested through the connection. IBD represents how components of a module are combined to create valid instances and how module instances are connected to external entities. IBD and BDD complement each other and provide module information.

(4) Parameter Diagram (Par.D)

Parameter diagram is a special internal block diagram, which is mainly used to describe the constraints on the system. These constraints are encapsulated in constraint block by means of mathematical expression. The variables involved in these constraints are described by constraint parameter of constraint block. Par.D can be applied to many kinds of scenarios that need to describe the system accurately, and it can be created at any stage of the system life cycle. In SysML, constraints are used to describe mathematical expressions. In order to support the reuse of constraints, constraint modules are used to encapsulate constraints. The constraint module has parameters that represent the variables involved in the constraint.

(5) Package Diagram (Pak.D)

In SysML, package is the basic unit of model organization. Package and its contents are displayed on package diagram, which is used to organize model structure.

(6) Use Case Diagram (UC.D)

A use case diagram is a view that is composed of actors, use cases, boundaries and their relationships to describe system functions. Use case diagram is a model diagram of system functions observed by external users. Use case diagram is the blueprint of the system. The use case diagram presents some participants, use cases and their relationships. It is mainly used to model the specific requirements scenarios of the system and subsystem, so as to analyze the system functional behaviors needed to implement the requirements.

(7) Activity Diagram (Act.D)

Activity diagram is the modeling of the action sequence that constitutes the dynamic behavior of the system. It emphasizes the input, output, sequence and condition of a single activity, and provides flexible links between activities to describe the complex behavior of the system. In SysML activities, there are two kinds of flows: object flow and control flow; object flow connects the output pins of activities with the input pins of other actions, so as to promote the flow of tokens between actions; control flow specifies the execution sequence of activities. In order to assign behavior to a structure, SysML defines an activity partition, which usually represents a module or a component, and is responsible for executing actions within the activity partition.

- (8) Sequence Diagram (Seq.D)
 Sequence diagram of SysML is an extension of sequence diagram in UML. Sequence diagram is the dynamic behavior view of the system, reflecting the behavior and event sequence over time. In software engineering, sequence diagrams provide three kinds of information: the order of behavior execution, the execution structure of behavior and the triggering of behavior by structure, so the software system behavior created by sequence diagrams can be automatically transformed into the definition of source code. Sequence diagrams are useful in the early stages of a system's life cycle to illustrate the possible interactions between the system of interest and its environment's performers.
- (8) State Machine Diagram (Stm.D)
 State machine diagram, like activity diagram and sequence diagram, is a kind of behavior diagram, which is used to describe the dynamic view of the system. Unlike these two graphs, it focuses on the event driven state changes of modules in the system. The description object of state machine diagram is usually block, which can be any level module in the system structure level, such as system itself, subsystem or individual component, so it can be created at any point in the system life cycle. Because of its precise and clear behavior description, state machine diagram is very suitable to describe the final result of the design as the input of the subsequent development process.

The focus of this chapter is to clarify the function and application method of SysML Model, so as to prepare the model for modeling in the process of information flow design using SysML Model.

4 SysML Model and Information Flow Process Mapping

As shown in Fig. 4, the SysML modeling of spacecraft information flow design process is essentially how SysML diagrams and elements are applied to the information flow design process to match the static information and dynamic process in the design process.

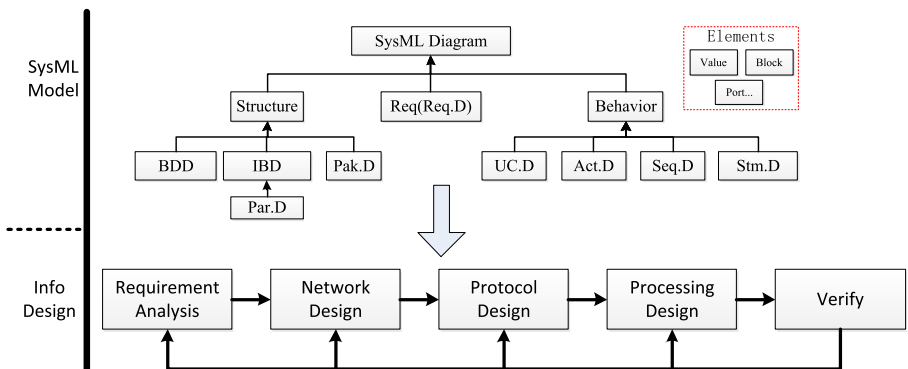


Fig. 4. Mapping between SysML model and information flow process

The Avoinics System is closely related to the information flow of spacecraft. The integrated electronic system involves the key factors related to the information flow, such as function, structure, data structure, transmission protocol and processing. Next, we will describe the process and method of SysML modeling from the perspective of integrated electronic system [6, 7].

4.1 Procedure Design

For the SysML modeling of spacecraft information flow, it is necessary to be familiar with the SysML modeling methodology, as well as the theoretical knowledge and engineering practice experience of electronic information system. Therefore, this paper adapts the modeling method according to the design characteristics of spacecraft information flow, and considers how to interface MBSE with spacecraft information flow design and simulation verification, such as the map of requirements, functions, behaviors and process, architecture, parameter constraints, etc. Based on the MagicGrid system modeling framework and the information flow design process, the information flow modeling process based on SysML model is designed as shown in the Fig. 5 [8].

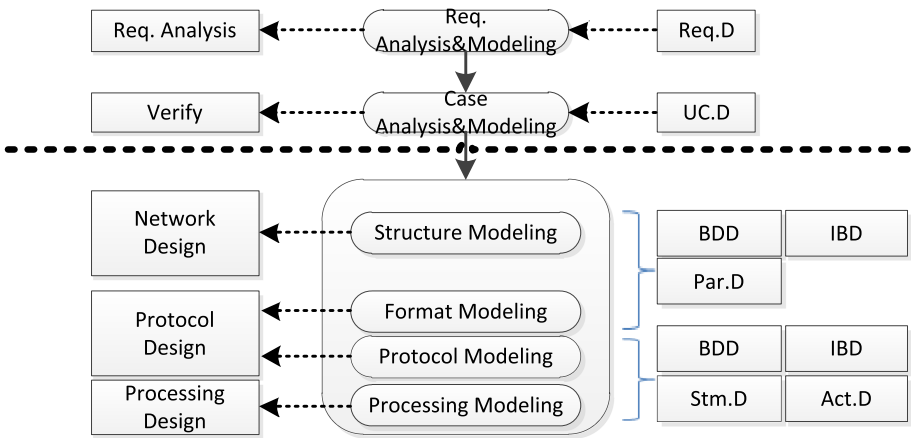


Fig. 5. Procedure design

First of all, requirement analysis and modeling is carried out based on user requirements; second, based on requirement model, functions to be verified are determined and use case model is established; third, based on use case model, function analysis is carried out to determine the structure model, data format model and transmission protocol model of the system and subsystem required to achieve this function; finally, through behavior modeling, structure is established The model establishes the information processing process model and completes the final process of information processing in destination.

To complete the modeling process, the following three corresponding relationships need to be clarified:

- (1) Correspondence between information flow modeling process and MagicGrid system modeling process
 - 1) Requirement analysis and modeling, use case analysis and modeling belong to the black box design stage of MagicGrid modeling process; 2) structure modeling, data format modeling, transmission protocol modeling and processing process modeling run through the gradual decomposition process of the whole satellite function and structure, and belong to the white box design stage and scheme design stage of MagicGrid modeling process.
- (2) Correspondence between information flow modeling process and SysML Meta model
 - 1) Use requirement diagram to analyze user requirements; 2) use case diagram to describe the Interaction scenario between user and system, and clarify the system functions that user needs to verify; 3) use block definition diagram and internal block diagram for structural modeling and data format modeling; 4) use activity diagram and state machine diagram for transmission protocol modeling and process modeling; 5) use parameters during structural modeling Figure for index constraints.
- (3) Correspondence between information flow modeling process and information flow design process

Requirement analysis & Modeling corresponding information requirement analysis; 2) use case analysis & Modeling corresponding information verification demonstration. The verification demonstration here can only be described in terms of logic and function, and does not have bottom-level code compilation and operation and code level simulation. Other modeling tools are required to participate in this stage; 3) structure modeling corresponding information network design; 4) data format modeling and transmission Transport Protocol Modeling corresponds to transport protocol design; 5) processing process modeling corresponds to information processing design.

4.2 Modeling Ideas

- (1) Requirements

The requirements of all stakeholders of information flow, which is all functions and performance that the system must meet, are reflected in the requirements diagram in the form of text.
- (2) Use Cases

For all functions in information flow design, use case diagrams are used to build corresponding use case models, namely application scenarios. Information input and system environment must be included in the models. The system environment contains the current use case and the functions to be verified. This model can be understood as test case design in software testing.
- (3) Structure

After completing the use case modeling, it needs to analyze the functions to be realized by the current use case, and the realization of the functions needs to be completed through the interaction between the functional entity and the entity.

- (1) Entity modeling needs to use block. Among them, the nesting of entities (satellite à equipment à module) is also modeled by block and shown with BDD and IBD.
- (2) In the IBD, the port and connector are added to the entity model to realize the data interaction between the entity block diagrams.

The entity model established by the BDD can be associated with the functions that need to be implemented or verified through the association matrix. Because entities may have multiple functions at the same time, the association matrix belongs to multi-to-multi association.

The parameter graph is used to constrain the parameters of the function module synchronously, so that the function must meet certain performance requirements on the premise of realization.

(4) Data Format

This part defines the format of information in different stages or different module structures in the process of transmission, and the visual representation of the model is block definition diagram.

The modeling of data format, like the modeling of the above entities, also has nesting, which is completed by block definition diagram.

(5) Transmission Protocol

This part defines which transmission protocols are used in different functional entities, and the information flow relationship between the protocols. The protocol itself is modeled by the block definition graph, the information processing within the protocol is modeled by the state machine graph, and the information flow between protocols is modeled by the internal block graph (input and output interfaces are added to the protocol model of the IBD).

(6) Process

This part defines the process of the information received by the sink. It uses activity diagram or state machine diagram to model, which belongs to the behavior modeling within the entity model.

5 Conclusion

Based on the analysis of the design process of Spacecraft information flow and the concept and structure characteristics of SysML Model, this paper proposes a SysML modeling design method of spacecraft information flow. This method provides a feasible solution to the problem of low efficiency in the requirement analysis and system design of spacecraft information flow based on text. At present, this method is only a prototype, still need more detailed and in-depth analysis and case verification, in order to better integrate the spacecraft information flow design process and SysML model, and improve the validity and reliability of spacecraft information flow design.

References

1. Hongtao, C., Yuchen, D., Jianhua, Y.: Basic principle of model-based system engineering. *Aerosp. China* (2016)
2. Shuhua, Z., Yue, C., Zheng, Z.: System design and simulation integration for complex mechatronic products based on SysML and Modelica, **30**(4), 728–738 (2018)
3. Lu, Z., Liu, X., Mao, Y.: Application practice of model-based system engineering method in satellite integrative system design, **27**(3), 7–16 (2018)
4. Kunsheng, W., Jianhua, Y., Hongtao, C., et al.: Research and practice of model—based systems engineering. *Aerosp. China* **11**, 52–57 (2012)
5. Yu, S., Li, M.: Model-based systems engineering and system modeling language. *Comput. Knowl. Technol.* **7**(31) (2011)
6. Han, F., Lin, Y., Fan, H.: Research and practice of model—based systems engineering in spacecraft development **23**(3), 119–124 (2014)
7. Yusheng, L., Yuqin, J., Shunting, G.: Model-driven modeling for system design of complex products: a survey. *China Mech. Eng.* **21**(6), 741–749 (2010)
8. Yifan, Z., Qun, L., Feng, Y., et al.: NASA. *NASA Systems Engineering Handbook*. Publishing House of Electronics Industry, Beijing (2012). (translated)