

Problem-Oriented Integration of SMP-models in the Simulation Modeling Infrastructure

Ludmila Nozhenkova, Olga Isaeva, Evgeniy Gruzenko and Aleksey Markov

Institute of Computational Modelling of the Siberian Branch of the Russian Academy of Sciences, Akademgorodok 50/44, 660036, Krasnoyarsk, Russia

Abstract—We suggest an original method of problem-oriented integration of SMP-models allowing to unite different manufacturer's models in one complex. The method consists in adding to the SMP-models structural-parametric and functional description creating a semantic level of presentation of a modeled object. Implementation of this method allows to solve the tasks of compatibility and interoperability of the SMP-models in the problem-oriented infrastructure of simulation modeling. The infrastructure is designed for the work of spacecraft onboard equipment designers.

Keywords—simulation model portability; spacecraft; onboard equipment; command and measuring system

I. INTRODUCTION

In the modern high-tech world we can clearly see the specialization of production entailing the necessity to create new approaches and software tools for different stages of the products' lifecycle. The European Cooperation for Space Standardization has developed the standards that recommend application of the computer modeling technologies for different tasks of space system design and production. As a rule, such complex technical systems include hardware of different manufacturers, and for its successful implementation it is economically efficient to conduct their primary study and integration on simulation models, and to use the built models during further operation. Universal approaches to organization of the modeling systems in order to provide union of different models in complex solutions are regulated by the Simulation Model Portability standard, or SMP2 in the current edition [2], recommended by the European Cooperation for Space Standardization. The SMP standard opens wide possibilities for building and control of the models, determines the development rules for model interaction interfaces for their compatibility and transferability in heterogeneous modeling environments.

Software environments realizing the SMP standard are the Simulation Infrastructures. The most famous are SimTG infrastructure used in Astrium Satellites [3], SimSAT of the European Space Agency [4], SWARMSIM mission control center simulator[5]. Some manufacturers create their own software solutions on the basis of the SMP standard [6]. Although there are uniform principles provided by the standard, every manufacturer is basing on his own understanding of the model realization, the ways of its creation and the internal logics of the simulation modeling. We are developing an original simulation infrastructure applying and adapting

universal principles of the standard, in order to analyze spacecraft onboard equipment's function. [7, 8]. The infrastructure is basing on the problem-oriented approach and is designed to support construction of onboard equipment. The infrastructure includes visual tools for information-and-graphic modeling allowing to build models in terms of the subject area with their further automated translation to the software elements working in accordance with the SMP standard. The logics of simulation models' function is set with the help of knowledge base. Introduction of an additional intellectual superstructure allows to save and distribute the experience of the unique spacecraft onboard equipment designers. Extension of the modeling principles provided by the SMP standard, caused the necessity to solve the task of the problem-oriented integration and interoperability of the models built in accordance with the standard, adding them with new graphical and intellectual elements, determining additional data semantics and including updated solutions in special working environment of the onboard equipment designer. Our original method of model integration allows to combine different models in one complex.

II. PROBLEM-ORIENTED MODEL DESCRIPTION

Creation of a model's problem-oriented description includes the stages of the modeling mission and goal analysis, forming of a semantic layer on its basis, determining of the structures of the subsystems constituting the model, and their functional dependences, building of graphical schemes. As a rule, UML is used as graphical modeling language in the infrastructures [9], and it's the base for integration of the business-logics in simulation models. UML-notation turned out to be not enough for building a problem-oriented model, we had to provide the designer with additional semantic constructions, typify the links between the information objects and bring models' description close to the terms of the subject area. We have developed a unified presentation of the modeled object that allows to use one and the same approaches and software tools to models creation, their integration and testing.

Structural-parametric description of an onboard equipment model is presented in form of $S = \langle B, C, D \rangle$, where B – model blocks describing properties or functions of different physical devices, C – typified information dependences describing commutation connections between the blocks, D – data structures.

Model's semantic level is created on the basis of the terms characterizing modeling's tasks and description of the

structural-parametric elements, i.e. the presentation that, on one hand, is typical for the modeled subject area, and on the other hand, is interpreted in graphical and software presentations of the simulation model.

The model blocks B_i from B can be represented as $B_i = \langle N_i, K_i, I_i, P_i \rangle$, where N_i – name of i -block; K_i – type of block, I_i – commutation interfaces; P_i – a set of parameters.

Interface $I_i = \{I_i^1, \dots, I_i^n\}$, where n – the number of input and output points B_i . Every interface I_i^n of the array I_i has characteristics: type of interface $Tp(I_i^n) \subseteq \{RS-232, RS-422, SpaceWire, impulse, relay, power\}$; direction of transmission $Rt(I_i^n) \subseteq \{Receiver, Transmitter\}$ – determines incoming or outgoing interface; status indication $Onf(I_i^n) \subseteq \{On, Off\}$ – shows whether it is possible to transmit data at a current moment.

Typification of informational dependences between the blocks is performed on the basis of the given commutation interfaces' characteristics. In course of modeling, typification allows to apply uniform rules to presentation and processing of the data transmitted through one-type interfaces.

Let's define the connection C_{ij} of the pair B_i and B_j as $C_{ij} = \langle I_i^n, I_j^m, \tau_{ij} \rangle$, where I_i^n – B_i interface, I_j^m – B_j interface, τ_{ij} – the signal travel time between interfaces. Only one-type interfaces can be connected: $Tp(I_i^n) = Tp(I_j^m)$.

The D of the model's data structures contains standard data types and special constructions describing telemetry and telecommand packages [10, 11], that are used to perform informational interaction of the onboard equipment. Introduction of the additional data structures typical for the modeled task, allows to provide the infrastructure with additional mechanisms for analysis and visualization both of the initial data and the results of modeling.

In order to create structural-parametric description of an onboard equipment model S in the infrastructure, the authors have developed the information and graphic modeling tools [12]. The tools allow to build models, define configurations of blocks and their connections.

III. AN EXAMPLE OF GRAPHICAL BUILDING OF A MODEL

Let's explain the method of information and graphic modeling with the help of an example. The problem-oriented infrastructure includes language constructions and sets of the terms of the subject area of the research.

For example, there are two devices completing interaction, i.e. $B = \{Device\ 1, Device\ 2\}$. Let's define their function goals and tasks. Let's suppose that one of the devices generates and transmits data to the other, and the other one processes data and returns it to the first one.

Let's define the types of the transmitted data and the data exchange channels. Let it be the model of transmitting of analogue signals and digital data packages. In total, there are five input/output points for each of the model's elements. For analogue signal transmission, each device has an appropriate interface for data reception and transmission, digital packages will be sent from the first device through interface RS-232 and

they will be received through RS-422. The array of commutation interfaces of the first device is $I_1 = \{I_1^1, I_1^2, I_1^3, I_1^4, I_1^5\}$. $Tp(I_1^1) = high-frequency$, $Tp(I_1^2) = RS-232$, $Tp(I_1^3) = high-frequency$, $Tp(I_1^4) = RS-422$, $Tp(I_1^5) = impulse$. The array of commutation interfaces of the second device is $I_2 = \{I_2^1, I_2^2, I_2^3, I_2^4, I_2^5\}$. $Tp(I_2^1) = high-frequency$, $Tp(I_2^2) = RS-232$, $Tp(I_2^3) = high-frequency$, $Tp(I_2^4) = RS-422$, $Tp(I_2^5) = relay$.

An example of graphical presentation created with the use of the author's software, is shown in Fig.1.

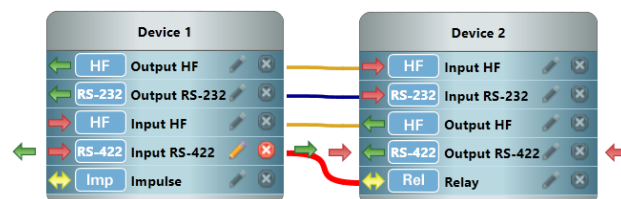


FIGURE 1. AN EXAMPLE OF GRAPHIC PRESENTATION

Each of the devices in the infrastructure is set as a separate model block. Visual software tools allow to determine the necessary commutation interfaces, specify their types and data transmission directions. Interface typification helps during building of information dependencies and determination of the received data processing algorithms. The modeling environment shows the hints: recommended interfaces (arrows), acceptable (the line in color of the legend indicating the type of interface) and erroneous (red line).

For the model building convenience, we have developed configurations of the onboard equipment's base elements. Our way of visual creation of a model using graphical elements is an alternative to using UML. In our opinion, its significant advantage is simplicity and clarity of usage for setting different configurations of the onboard systems.

IV. DEFINING OF THE FUNCTIONS OF MODEL

The functions of the model are determined on the basis of its structural-parametric description. The simulation infrastructure includes three ways of creating the functions of a model: in form of condition-action rules [13], in form of C++ programs, that appear during importing of SMP models, and in the form of virtual instruments (VI), realized in a specialized environment Labview [14].

Model's functional description is set in the form $F = \{R, CPP, VI\}$, where R – the rules from the knowledge base, CPP – software libraries, VI – virtual instruments.

Thus, a simulation model M is presented in the form: $M = \langle S, F, T \rangle$, where S – structural-parametric description, F – functional description, T – the array of the control elements of the modeling time and the transitions between the model events: model time, system time, timers, execution delays, etc.

The simplest way is to create the functional description of a model in form of the rules $R = \{A \rightarrow D\}$. R is a knowledge base consisting of condition-action rules. Each rule is a construction $A \rightarrow D$, where A – condition, D – action. Condition A determines at what state of the model's elements the given rule can be applied. The right part of the rule is D – action. It

determines the next step in task solution, or the way of changing the model's state. The conditions and actions are the expressions on variables set in structural-parametric description of the model, or the functions performing changes of a model's state.

In the example above the simulation model can be described with the rules determining data reception and transformation. For instance, Figure 2 shows the rule in accordance with which a receiving device gets data through communication interface RS-232 and sends a two byte of the received package through interface RS-232 back to the device, from which it has received the data.

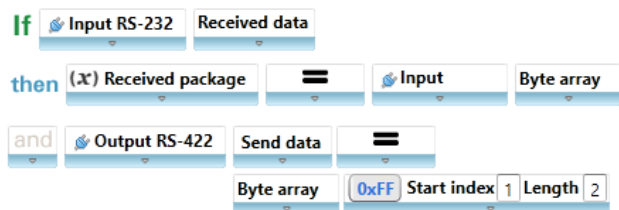


FIGURE II. AN EXAMPLE OF A RULE SETTING

Creating of the function s of a model in terms of the subject area allows a qualified spacecraft onboard systems designer to easily build and then modify a simulation model, without using programming skills.

The additional methods of modeling can be integrated in the simulation model in form of connectable libraries realized with C++ or in Labview.

V. STRUCTURAL-AND-FUNCTIONAL METHOD OF MODEL INTEROPERABILITY

Structural-parametric and functional description has allowed to solve the tasks of compatibility and interoperability of the models in the problem-oriented simulation infrastructure. The infrastructure provides control of the models' function on the basis of the events (event-based design), appearing during modeling, or provided by the scenarios. A class of the events appearing during interaction of the models is formed in the infrastructure on the basis of different models' descriptions, their entry points and commutation interfaces. The work methods declared in the functional description allow to create the events produced by the results of the modeling.

Summarizing all of the above, we have the following groups of the events: appearing when input parameters are changed, appearing during data transmission, related to the model function call, related to the modeling results' obtainment, initiated by timers and time services. Fig. 3 shows the principle of interaction of simulation models in simulation infrastructure through the events mentioned above. The scheme shows that the integrating infrastructure is the central component that performs subscriptions of the models to the events, supports message exchange, logging and state model monitoring.

The infrastructure sets input data, calls the methods changing the models' state, consolidates parameters and the results of the modeling. It changes timers, controls the model's

time. Besides, it realizes the logical inference that are called for the models simulating the devices' function logics.

In the process of simulation tests the state of the models is changed in accordance with the modeling scenario or the user's actions.

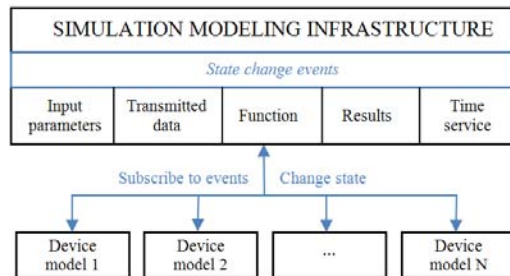


FIGURE III. SIMULATION MODELS' INTERACTION PRINCIPLE IN THE INFRASTRUCTURE

Although there are direct connections of the model's elements (as shown in Fig.2), they are realized through the infrastructure's events.

Simulation model's creation on the basis of structural-parametric and functional description provides unification of different implementations of simulation models, their consolidation and interoperability. Introduction of the types of connections of the model's elements allows to apply uniform methods of data processing and visualization, enter errors and transmission interference. Such approach allows to conduct modeling of emergency situations, communication line breakages and channels' duplication analysis.

Our methods allow to adapt the universal approaches of the SMP standard to the specifics of the problem-oriented modeling environment.

VI. CONCLUSION

Investigation of the SMP-model integration methods complies with the modern tendencies in the area of simulation modeling. It provides development of the problem-oriented software tools able to unite models of different manufacturers in one technological complex in the space instrument-making industry.

Our formalization of the simulation model and structural-functional approach to interoperability provides problem-oriented integration of the models, modeling process support, organization and planning of simulation experiments.

Introduction of a semantic layer in the simulation infrastructure in order to include the subject area's language constructs and sets of terms, makes the infrastructure easy to work with for the designers, thus helping to accumulate and use the knowledge in the area of spacecraft onboard equipment design.

ACKNOWLEDGMENT

The reported study was funded by RFBR and Government of Krasnoyarsk Territory according to the research project № 16-41-242042.

REFERENCES

- [1] Space engineering. System engineering general requirements (ECSS-E-ST-10C), European space agency (ESA): ESTEC, 100 p., 2009.
- [2] Simulation modelling platform. ECSS E-40-07 // ESA Requirements and Standards Division ESTEC, The Netherlands, 2011, p. 49.
- [3] C. Cazenave and W. Arrouy, "Implementing SMP2 Standard within SimTG Simulation Infrastructure," SESP (ESA) : Simulation and EGSE for Space Programmes, 2012.
- [4] J. Eggleston, H. Boyer, D. van der Zee, A. Pidgeon, N. de Nisio, F. Burro, and N. Lindman, "Simsat 3.0: Esoc's New Simulation Infrastructure," 6th International Symposium on Reducing the Costs of Spacecraft Ground Systems and Operations (RCSGSO)
- [5] Peter Fritzen, Daniele Segneri and Max Pignède, "SWARMSIM – The first fully SMP2 based Simulator for ESOC," 11th Int. WS on Simulation & EGSE facilities for Space Programmes SESP, 2010.
- [6] Cheol-Hea Koo, Hoon-Hee Lee, Sung-Tae Moon, Sang-Hyuck Han and Gwang-Hyeok Ju, "Development of simulation infrastructure compatible with ESA SMP for validation of flight software and verification of mission operation" SESP (ESA): Simulation and EGSE for Space Programmes, 2012.
- [7] Nozhenkova L.F., Isaeva O.S., Gruzenko E.A., Koldyrev A.Yu, Markov A.A., Belorusov A.I., Vogorovskiy R.V. Unified description of the onboard equipment model on the basis of the «Simulation Model Portability» standard // Advances in Intelligent Systems Research (ISSN 1951-6851), Vol. 133, 2016, pp. 481-484, doi:10.2991/aiie-16.2016.111.
- [8] Nozhenkova L.F., Isaeva O.S., Gruzenko E.A., Koldyrev A.Yu. Integration technology of the onboard equipment simulation models in simulation modeling infrastructure // Proceedings of the 2016 International Conference on Electrical Engineering and Automation (ICEEA2016), 2016, pp.618-622, doi:10.12783/dtetr/iceea2016/6728.
- [9] M. Pignède, J. Morales, P.Fritzen and J.Lewis, "Swarm Constellation Simulator," SpaceOps 2010 Conference Delivering on the Dream Hosted by NASA Marshall Space Flight Center and Organized by AIAA, 2010, p. 2323.
- [10] Packet Telemetry Standard (ESA PSS-04-106) Issue 1: European space agency, 73 p., 1988.
- [11] Packet Telecommand Standard (ESA PSS-04-107) Issue 2: European space agency, 166 p., 1992.
- [12] Nozhenkova L., Isaeva O., Gruzenko E. Computer Simulation of Spacecraft Onboard Equipment // ACSR-Advances in Computer Science Research, Vol.18, 2015, pp.943-945, DOI: 10.2991/cisia-15.2015.255.
- [13] Russell and J. Stuart, Artificial intelligence: a modern approach, Prentice-Hall, Inc. A Simon & Schuster Company Englewood Cliffs: New Jersey, p. 932 , 1995.
- [14] LabVIEW function and VI reference manual. National Instruments Corporation, Austin, Texas, 1998.