CONTINUOUS MODELING IN AIRCRAFT TOP-LEVEL DEMONSTRATION

Zang Jing, Wang Qian, Zhu JiaQiang
Aviation Industry Development Research Center of China,
Aviation General Demonstration Laboratory

Keywords: *system engineering, modeling and simulation, aircraft demonstration*

Abstract

The requirement of modeling and simulation for SoS design and virtual verification and validation in aircraft top-level demonstration is introduced. To fill in gaps between architecture models and simulation models, a continuous modeling method is proposed. There are three steps which are manual modeling, direct drive, and smooth transformation to realize the continuous behavior modeling. For physical modeling, the aircraft conceptual scheme(physical) can be transformed to the simulation model for cross-domain integration through the interface definition. A sample case is provided to illustrate the low-level behavior model transformation.

1 Introduction

As a kind of typical complex system, to satisfies the need of the system-of-systems(SoS) operation, aircraft development requires the top-level demonstration as the initial stage[1]. The core of top-level demonstration is the modeling of SoS for operational concept analysis. In Model-Based System Engineering(MBSE)[2-4], Model-Centric Engineering(MCE)[5], Digital Twin/Digital Thread[6-8] for Intelligent Manufacturing, and other system engineering theories, the continuous virtual verification and validation is necessary. Therefore, the model continuum is also necessary in aircraft top-level demonstration which is the upper left part of V-model(systems development lifecycle) of the systems engineering process.

2 Modeling and simulation in Aircraft Top-level Demonstration

Shown as the red circle in Figure 1, the top-level demonstration stage is located in the upper left corner of the V-model in system engineering. The process of the top-level demonstration can also be represented as a small "V" diagram including scenario analysis, SoS design, virtual verification and validation.

The purpose of SoS design in aircraft top-level demonstration is proposing the aircraft stakeholder requirements including functional requirements and performance requirements through operational concept analysis. Such as architecture for functional requirements analysis, mission simulation and effectiveness evaluation are common methods used for performance requirements analysis in aircraft top-level demonstration. For the top-level demonstration process, the architecture modeling method is used in SoS design and the mission simulation is used in virtual verification and validation for the SoS architecture.

In this article, it is worthwhile to note that the mission simulation levels include campaign, mission, and engagement to achieve SoS simulations. Therefore, shown as the red rectangle in Figure 1, the modeling and simulation(M&S) levels in aircraft top-level demonstration correspond to the top three layers of the traditional M&S pyramid[9].

The traditional architecture models are concept models described by formal language such as SysML(Systems Modeling Language). There are some modeling tools can support logical verification by state machine after generating codes from concept models. However, the concept models still can not be directly executed as behavior models in the mission simulation system. Without effective model transforming, the behavior models should be mapped from architecture models manually. Consequently, there will be not only longer iteration cycle of verification and validation, but also the misunderstanding risks due to the non-model data transferring. On the other hand, in SoS architecture design, the detailed behavior describing is not always existed and neither necessary, but more details may be indispensable in mission simulation according to simulation granularity. Therefore, to fill in gaps between architecture models and simulation models, a continuous modeling method is urgently needed in aircraft top-level demonstration.

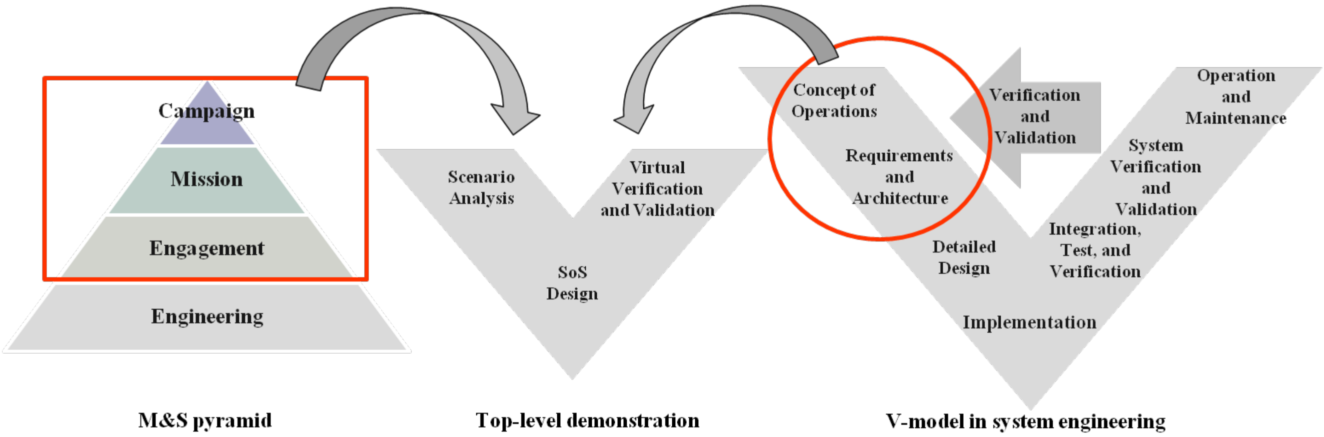

Fig. 1. Modeling and simulation in aircraft top-level demonstration

3 Continuous Modeling Method

The models in the mission simulation can be roughly divided into physical models and behavior models. Behavior can include the reactions and interactions of simulation entities to environmental conditions or other simulation entities[9]. Shown in Figure 2, to achieve the continuous modeling from architecture model to simulation model, the meta-model which define the range of models should be defined firstly[2][10-12]. Then the architecture models can be transformed to simulation models with the same granularity. In addition, using existing simulation resources, the continuous physical modeling can be achieved by cross-domain model integration. However, the essential physical models of developing aircraft should be completed through some concept design tools[13] after system architecture modeling and can be mapped into mission simulation system in model integration.

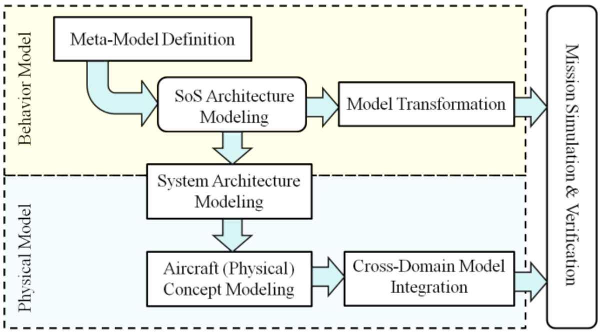

Fig. 2. Continuous modeling process

3.1 Behavior Modeling

As a part of MBSE process, the top-level demonstration focus on the SoS architecture modeling which describes the relationship among the systems and obviously includes the concept of the system's behaviors. Thus, the continuous behavior modeling is to achieve the transformation from concept model to simulation model. As the core of continuous modeling in aircraft top-level demonstration, the behavior model transformation can be realized by three steps(Fig. 3).

- The first step is manual modeling which has quite a lot of shortages mentioned above. This method can be regarded as the discontinuous modeling.
- In the second step, regarded as the low-level continuous modeling, the entities in the simulation system are directly driven by the architecture models instead of the behavior models in the same range. This method is used for achieving the continuity at the tool level through the interface definition and will be shown in the following sample case.
- As the high-level continuous modeling, in the final step, the smooth transformation is achieved through the meta-model definition. Consequently, both the architecture model and behavior model will be instantiation of the meta-model which is based on aircraft operational ontology[4][14]. Through the use of artificial intelligence technology, automatic/semi-automatic transformation also can be realized in this step. However, there may be some difficulties caused by the undisclosed meta-models(actual existed) of the behavior models from most simulation systems.

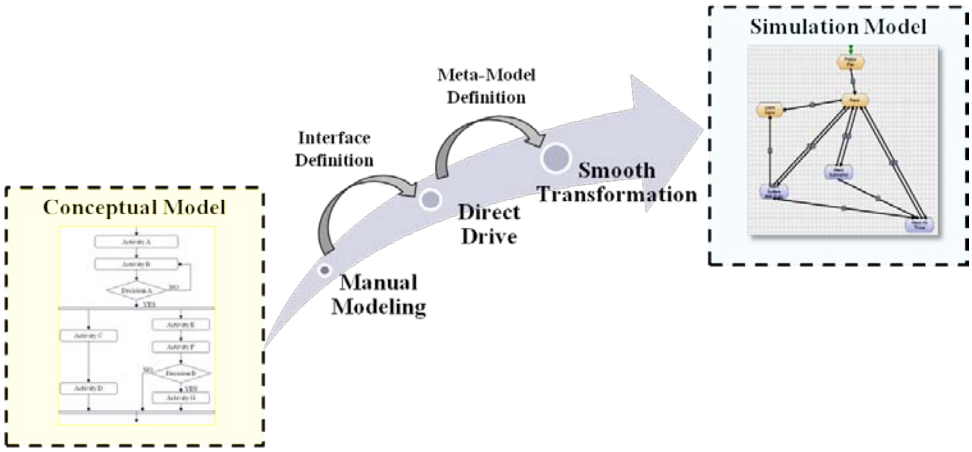

Fig. 3 Behavior model transformation

3.2 Physical Modeling

The mission simulation in the stage of top-level demonstration should use numerous physical models, most of which are existing proven reliable models. However, as the object of demonstration, the aircraft's simulation model(physical) should maintain continuity with the model from top-level demonstration. Shown in Figure 2, after the system architecture modeling based on the SoS architecture model from top-level demonstration, the aircraft conceptual scheme(physical) will be generated and can be transformed to the simulation model for cross-domain integration through the interface definition. It is important to note that this work may not be completed in top-level demonstration. But the achievement will be used in the mission simulation for top-level demonstration.

4 Sample Case

This sample case is provided to illustrate the continuous modeling by behavior model transformation. First, the scenario analysis based on ConOps(concept of operations) and SoS architecture modeling in DoDAF(Department of Defense Architecture Framework) are completed traditionally. Then parts of the behavior models in mission simulation system are replaced by corresponding state machine models from systems viewpoint (SV) including SV-10b systems state transition description and SV-10c systems event-trace description through DDS(Data Distribution Service) flexible bus[15][16]. Therefore, the architecture models can drive the simulation entities directly instead of once more behavior modeling in simulation system(Fig. 4). For example, if the behavior of the regular reporting from the aircraft to the command center is modeled in SoS architecture modeling, the state machine of the regular reporting can drive the corresponding aircraft relate to this state machine to report to the command center. Meanwhile, the behavior model with the same function in the simulation system is unnecessary any more. Moreover, the changes of the entities' state's in mission simulation will also feed back and drive the logical simulation of the architecture models. It is important to note that the behavior models outside the SoS architecture can take advantage of the legacy resources of the different mission simulation systems. As low-level continuous modeling, the sample case achieves the model transformation between the architecture design tool and the simulation system with the limited development work.

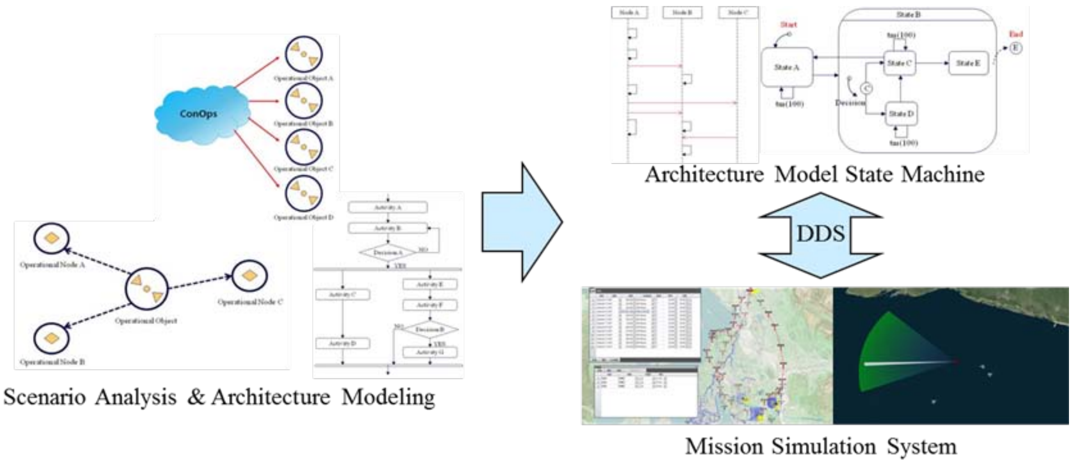

Fig. 4. Model transforming from architecture to mission simulation

5 Conclusion

The proposed continuous modeling method can satisfy the requirement of the continuous virtual verification and validation in aircraft top-level demonstration. For behavior modeling, there are three steps which are manual modeling, direct drive, and smooth transformation to realize the continuous modeling. For physical modeling, the aircraft conceptual scheme(physical) can be transformed to the simulation model for cross-domain integration through the interface definition. However, the in-depth research on the meta-model across the architecture models and the behavior models will continue to achieve the smooth model transformation.

References

[1] Li Q, Yan J, Zhu J Q, Huang T and Zang J. State of art and development trends of top-level demonstration technology for aviation weapon equipment. *Acta Aeronautica et Astronautica Sinica*, Vol. 37, No. 1, pp 1-16, 2016.

[2] MacCalman A, Lesinski G and Goerger S. Integrating external simulations within the model-based systems engineering approach using statistical metamodels. *Procedia Computer Science*, Vol 95, pp 436-441, 2016.

[3] Acheson P, Dagli C and Kilicay-Ergin N. Model based systems engineering for system of systems using agent-based modeling. *Procedia Computer Science*, Vol 16, pp 11-19, 2013.

[4] Ruijven L C. Ontology and model-based systems engineering. *Procedia Computer Science*, Vol 8, pp 194-200, 2012.

[5] Blackburn M, Bone M. Transforming systems engineering through model-centric engineering technical report SERC-2016-TR-109, 2016.

[6] West T D and Blackburn M. Is Digital Thread/Digital Twin affordable? A systemic assessment of the cost of DoD's latest Manhattan project. *Procedia Computer Science*, Vol 114, pp 47-56, 2017.

[7] Uhlemann T H, Schock C, Lehmann C, Freiberger S and Steinhilper R. The Digital Twin: demonstrating the potential of real time data acquisition in production systems. *Procedia Manufacturing*, Vol 9, pp 113-120, 2017.
[8] Helu M, Hedberg T and Feeney AB. Reference architecture to integrate heterogeneous manufacturing systems for the digital thread. *CIRP Journal of Manufacturing Science and Technology*, Vol 19, pp 191-195, 2017.
[9] Loper M L. *Modeling and simulation in the systems engineering life cycle*. Springer, 2015.
[10] Zomer L, Moustaid E and Meijer S. A meta-model for including social behavior and data into smart city management simulations. *2015 Winter Simulation Conference (WSC)*, pp 1705-1716, 2015.
[11] Xu J, Ye D and Ma Y. The design of a domain-specified visualization modeling language for warfare simulation based on meta-model. *2013 IEEE 4th International Conference on Software Engineering and Service Science*, pp 179-183, 2013.
[12] Vo D, Drogoul A and Zucker J. An Operational Meta-Model for Handling Multiple Scales in Agent-Based Simulations. *2012 IEEE RIVF International Conference on Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF)*, pp 1-6, 2012.
[13] Luo M Q, Liu H and Wu Z. Prototype system research on open conceptual aircraft design. *Acta Aeronautica et Astrnautica Sinica*, Vol. 29, No. 4, pp 954-959, 2008.
[14] Ruijven L C. Ontology for systems engineering. *Procedia Computer Science*, Vol 16, pp 383-392, 2013.
[15] Kim D, Oh H and Hwang S W. A DDS-based distributed simulation for anti-air missile systems. *2016 6th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH)*, pp 1-7, 2016.
[16] Park Y and Min D. Distributed traffic simulation using DDS-communication based HLA for V2X. *2015 Seventh International Conference on Ubiquitous and Future Networks*, pp 450-455, 2015.

4 Copyright Issues

The copyright statement is included in the template and must appear in your final pdf document in the position, style and font size shown below. If you do not include this in your paper, ICAS is not allowed and will not publish it.

5 Archiving

The ICAS 2018 proceedings will receive an ISBN number and will be cataloged, and archived by the German National Library.

8 Contact Author Email Address

mailto: zangjing2006@163.com

Copyright Statement