Proceedings of the 2015 Winter Simulation Conference
L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.

MODEL-ARCHITECTURE ORIENTED COMBAT SYSTEM EFFECTIVENESS SIMULATION

Yonglin Lei,
Ning Zhu, Jian Yao, Zhi Zhu
Simulation Engineering Institute,
109 Yanwachi Rd, National University of Defense Technology, Changsha, Hunan, CHINA

Hessam S. Sarjoughian
Arizona Center for Integrative Modeling & Simulation,
699 S. Mill Avenue, Arizona State University,
Tempe, AZ, 85281, USA

ABSTRACT

Combat system effectiveness simulation (CESS) is a special type of complex system simulation. Three non-functional requirements (NFRs), i.e. model composability, domain-specific modeling, and model evolvability are gaining higher priority from CESS users when evaluating different modeling methodologies for CESS. Traditional CESS modeling methodologies are either **domain-neutral** (lack of domain characteristics consideration and limited support for model composability) or **domain-oriented** (lack of openness and evolvability) and fall short of the three NFRs. Inspired by the concept of **architecture** in systems engineering and software engineering fields, we extend it into a concept of **model architecture** for complex simulation systems, and propose a **model-architecture oriented** modeling methodology in which model architecture plays a central role in achieving the three NFRs. Various model-driven engineering (MDE) approaches and technologies, including SMP, UML, DSM, and so forth, are applied where possible in representing the CESS model architecture and its components' behaviors from physical and cognitive domain aspects.

1 INTRODUCTION

Combat system effectiveness simulation (CESS) is a special type of complex system simulation. Generally, there are two kinds of requirements for simulation modeling. One is functional requirement (FR), i.e. what functionalities should be provided by the simulation models. Typical concerns include what elements and relations should be taken into consideration; what measures should be evaluated based on the simulation outputs; what precision should be supported in representing certain variables; etc. The other is non-functional requirement (NFR), i.e., how well the simulation models are structured and represented. There are many possible criterions for judging "how well". Any validated simulation model must have met the NRs, but not necessarily NFRs. Usually, NFRs are the main focus where one modeling methodology is preferred over others. In CESS field, three such kinds of criterions or NFRs are found significant and prioritized by the users and experienced modelers, i.e. model composability, domain-specific modeling, and model evolvability.

Traditional CESS modeling methodologies can be roughly divided into two categories. One is domain-neutral and application-specific by using generic M&S technologies and providing a powerful infrastructure and a model library with many domain components inside. Examples include simulation protocol standards, like HLA and SOA; model specification standards, like BOM and SMP2; unified/universal modeling formalisms, like DEVS and UML. The second category can be called domain-oriented by providing a CESS-oriented simulation system, within which different simulation applications can be composed from built-in components and configured with application-specific parameter values. Prominent examples include EADSIM, FLAMES, OneSAF, NSS, etc. The limitations of domain-neutral methodologies lie in lack of domain characteristics consideration and limited support for model composability; whereas domain-oriented methodologies usually fall short of openness and evolvability.

978-1-4673-9743-8/15/$31.00 ©2015 IEEE

Inspired by the concept of architecture in systems and software engineering fields, we extend it into a concept of model architecture for complex simulation systems, and propose a model-architecture oriented modeling methodology in which model architecture plays a central role in achieving the three NFRs. Various model-driven engineering (MDE) approaches and technologies, including SMP, UML, DSM, EMF, GMF, etc., are applied where possible in representing the CESS model architecture and its components' behaviors from physical and cognitive domain aspects.

2 FROM ARCHITECTURE TO MODEL ARCHITECTURE

Architecture is the "*Fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution*." A complex CESS is a system. Its architecture would include elements like a simulator, simulation services, and simulation models; and relationships between simulator and simulation models, and those among different simulation models as schematically shown in the left part of Figure 1. Since the simulator, simulation services, and relationships between simulator and simulation models are largely determined by the M&S technology chosen, the remaining complexity will mainly lie in simulation models and their relationships. Following the definition of architecture of a system, we define model architecture of a simulation system as follows:

Model architecture *of a simulation system is the fundamental concepts or properties of the simulation system in its execution environment embodied in its model components, their relationships, and principles of building or evolving these models*.

In a sense, model architecture is the kernel of a CESS-like complex simulation. Model architecture is also a key to resolve the NFRs emerged from CESS and other complex simulations. To achieve this, the model architecture is further divided into a domain model architecture (DMA) and many possible application model architectures (AMAs). Each AMA will reuse the DMA by either customizing or extending.

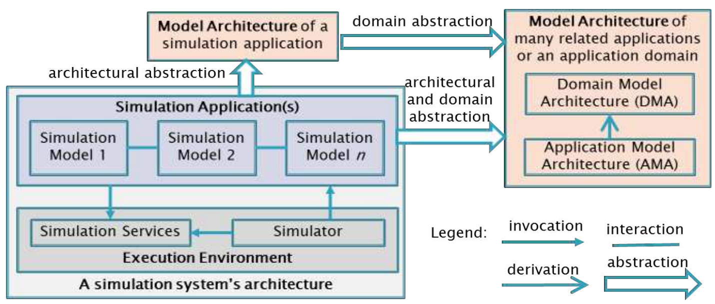

Figure 1 from simulation's architecture to its model architecture

3 MODEL ARCHITECTURE-ORIENTED CESS MODELING METHODOLOGY

From the viewpoint of model architecture, traditional CESS modeling methodologies can be structured as in Figure 2 (a, b) with four logic layers generic to simulation modeling, including experiment, simulation application, simulation application environment (SAE), and simulation development environment (SDE) layer. Model-architecture oriented methodology (see the c part of figure 2) basically follows ideas behind domain-oriented methodology and incorporates powerful M&S technologies found in domain-neutral methodology and make three steps further: 1) Building complex simulations around model architecture. 2) Making domain abstraction in model architecture. 3) Applying MDE to model architecture and its components' behaviors.

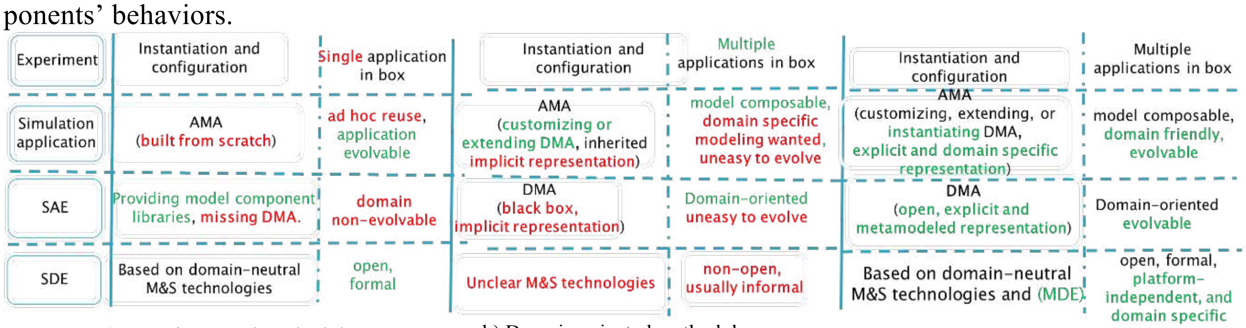

Figure 2 Traditional CESS modeling methodologies from the viewpoint of model architecture