Semantic Composability and XMSF

Mikel D. Petty, Ph.D.
Virginia Modeling, Analysis and Simulation Center
Old Dominion University
7000 College Drive, Suffolk VA 23435
mpetty@vmasc.odu.edu 757-686-6210

1. Abstract

Composability is the capability to select and assemble simulation components in various combinations into simulation systems. The defining characteristic of composability is the ability to combine and recombine components. There are both *syntactic* and *semantic* forms of composability; they deal respectively with technical aspects of enabling components to work together and with whether their combined computation is meaningful. Composability is a central requirement for XMSF. Interoperability is necessary but not sufficient for composability. The envisioned XMSF infrastructure is oriented towards interoperability and syntactic composability and does not directly address semantic composability. Guaranteeing or enforcing semantic composability may not be within the scope of XMSF. However, certain features of XMSF, together with a formal theory of composability, could support semantic composability.

2. Composability and interoperability

Composability is the capability to select and assemble simulation components in various combinations into simulation systems to satisfy specific user requirements.[1] The defining characteristic of composability is that different simulation systems can be composed in a variety of ways, each suited to some distinct purpose, and the different possible compositions will be usefully valid.[2] Composability is more than just the ability to put simulations together from parts; it is the ability to combine and recombine, to configure and reconfigure, sets of parts from those available into different simulation systems to meet different needs.

Composability exists in two forms, *syntactic* and *semantic* [1] [2]. Syntactic composability is the actual implementation of composability, and requires that the composable components be constructed so that their implementation details, such as parameter passing mechanisms, external data accesses, and timing assumptions are compatible for the different configurations that might be composed. Semantic composability goes beyond the question of whether the components can work together; it is the more difficult question of whether the models that make up the composed simulation system can be meaningfully composed. Are the data representations compatible among the composed models? Is the output produced by one model and input to the next within the latter's domain of validity? Are the assumptions the models make about the reality being simulated consistent? Although there has been some work on syntactic composability there has been much less on semantic composability, though the need for such research has been recognized [3] [4].

Related to composability is *interoperability*, the ability of different simulations or simulation components, connected in a distributed system, to collaboratively simulate a common scenario.

---

[1] The term *composability* is used with multiple meanings or levels. The difference is the question what is being composed and what is formed by the composition [7]. Here we will focus on composability of software *components*, as defined for XMSF [6].

[2] If the compositions aren't valid, then by definition they aren't composable.

Their interoperation may be mediated by an interoperability protocol, such as DIS, ALSP, or HLA. Like composability, interoperability exists in two forms, *technical* and *substantive*. Technical interoperability is the compatible, correct use of the interoperability protocol. Substantive interoperability is the exchange of information that is mutually consistent with the interoperating simulations' model semantics [5].[3] These two forms of interoperability are closely analogous to the syntactic and semantic forms of composability. Interoperability is necessary but not sufficient to provide composability. Components that are not interoperable can not be composed, so interoperability is necessary for composability.[4] However, interoperability is not sufficient to provide composability, i.e., components may be interoperable but not composable. Components that are interoperable in one specific configuration and cannot be combined and recombined in other ways (without significant effort) are not composable.

3. The importance of composability in XMSF

Composability is a central feature of XMSF: *"XMSF must support multiple levels of model and component composability…"* [6]. However, most of the envisioned XMSF infrastructure is oriented towards interoperability and syntactic composability, i.e., the question of allowing the models to work together. The XMSF infrastructure does not directly address semantic composability, the question of whether the composed components can meaningfully work together to simulate something. This has been recognized: *"… the semantics of composability is outside the scope of XMSF itself."* [6]. A goal of XMSF is that a simulation user faced with a requirement will have the ability to easily find and select relevant components from among those available, enable their interoperability with the XMSF infrastructure capabilities, and distribute their run-time computation via the web. Meeting this goal requires syntactic composability, of course, but it also requires semantic composability. Without both, XMSF would essentially be another interoperability protocol, similar to HLA or ALSP. It would be web-capable, certainly a useful feature, but it would retain the characteristic that building simulation systems could require extensive integration effort to reconcile components' semantics. Indeed, such integration could be even more difficult in the context of XMSF because of the increased potential for composition of independently developed and remotely executing components.

4. Support for semantic composability in XMSF

Semantic composability appears essential for XMSF, but *guaranteeing* or *enforcing* semantic composability may not properly be within the scope of XMSF. Guaranteeing or enforcing semantic composability seems to require control or access into the internal computations of the components. Such an intimate relationship between the XMSF protocols and the components' semantics would be undesirable at several levels and negate some of the advantages of XMSF.

XMSF could, however, provide capabilities that usefully *support* semantic composability while stopping short of guaranteeing or enforcing it. We[5] have recently embarked upon a research project aimed at developing a complete formal theory of (semantic) composability, starting from the existing theories of computability and mathematical logic.[6] It is premature to say how that

---

[3] HLA compliance primarily establishes technical, not substantive, interoperability.

[4] Also, federates that are composable are necessarily interoperable.

[5] The project is at Old Dominion University's Virginia Modeling, Analysis, and Simulation Center.

[6] The idea for a theory of composability was suggested to us by some earlier work that examined certain aspects of composability from a formal point of view [8].

work will end, but it is a not-too-unreasonable extrapolation to claim that it will be possible to represent those aspects of a component's semantics pertinent to its composability in a formal, unambiguous form. Assuming that to be true, than if each XMSF component has associated with it meta-data defining its semantics in a common formal way, XMSF algorithms could use that formal semantic meta-data to support semantic composability in at least two ways:

1. *Component search and selection.* Suppose a simulation user would like to locate a component that provides a needed capability, i.e., has certain semantics. If the requirements can be expressed formally, perhaps with the help of a component search tool, the semantic meta-data of the available components could be searched to find a component (or components) that meets the requirements.[7] The comparison of the requirement with the components' semantics would be done formally, so ideally the user would be certain, if the requirements were properly specified, that selected component(s) would meet it.
2. *Composability checking.* Once a set of components has been selected, the formal semantic meta-data could be logically processed to determine if those components were semantically composable. As with the previous point, the comparison would be done formally, so the results could be relied upon.

5. References

[1] D. R. Pratt, L. C. Ragusa, and S. von der Lippe, "Composability as an Architecture Driver", *Proceedings of the 1999 Interservice/Industry Training, Simulation and Education Conference*, Orlando FL, November 29 1999-December 2 1999.

[2] A. Z. Ceranowicz, "Composability Wrapup", *Electronic mail*, June 10 2002.

[3] P. Davis, P. A. Fishwick, C. M. Overstreet, and C. D. Pegden, "Model composability as a research investment: Responses to the featured paper", *Proceedings of the 2000 Winter Simulation Conference*, Orlando FL, December 10-13 2000, pp.1585-1591.

[4] S. Kasputis and H. C. Ng, "Composable simulations", *Proceedings of the 2000 Winter Simulation Conference*, Orlando FL, December 10-13 2000, pp. 1577-1584.

[5] J. S. Dahmann, "High Level Architecture Interoperability Challenges", *Presentation at the NATO Modeling & Simulation Conference*, Norfolk VA, October 25-29 1999.

[6] D. Brutzman, M. Zyda, M. Pullen, and K. L. Morse, "Extensible Modeling and Simulation Framework (XMSF), Opportunities for Web-Based Modeling and Simulation, Technical Opportunities Workshop Whitepaper", *Unpublished manuscript*, Naval Postgraduate School, 2002.

[7] M. D. Petty, "A Composability Lexicon", *Unpublished manuscript*, Virginia Modeling, Analysis and Simulation Center, 2002.

[8] E. H. Page and J. M. Opper, "Observations on the Complexity of Composable Simulation", *Proceedings of the 1999 Winter Simulation Conference*, Phoenix AZ, December 5-8 1999, pp. 553-560.

6. About the author

Mikel D. Petty is Chief Scientist of the Virginia Modeling, Analysis & Simulation Center at Old Dominion University. He received a Ph.D. in Computer Science from the University of Central Florida in 1997. Dr. Petty has worked in modeling and simulation research and development since 1990 in the areas of simulation interoperability, computer generated forces, multi-resolution simulation, and applications of theory to simulation.

---

[7] The process of selecting a set of components to meet a given set of requirements is surprisingly complex [8].