2012

How is M&S Interoperability Different From Other Interoperability Domains?

Andreas Tolk
Old Dominion University, atolk@odu.edu

Saikou Y. Diallo
Old Dominion University, sdiallo@odu.edu

Jose J. Padilla
Old Dominion University, jpadilla@odu.edu

Charles D. Turnitsa

Original Publication Citation

Tolk, A., Diallo, S. Y., Padilla, J. J., & Turnitsa, C. D. (2012). How is M&S interoperability different from other interoperability domains? *M&S Journal, 7*(3), 5-14.

How is M&S Interoperability different from other Interoperability Domains?

AUTHORS

Dr. Andreas Tolk
Engineering Management
& Systems Engineering
242B Kaufman Hall
Old Dominion University
Norfolk, VA 23529, USA
atolk@odu.edu

Dr. Saikou Y. Diallo, Dr. Jose J. Padilla
Virginia Modeling Analysis
& Simulation Center
Old Dominion University
1030 University Blvd
Suffolk, VA 23435, USA
sdiallo@odu.edu; jpadilla@odu.edu

Dr. Charles D. Turnitsa
TSYS School of Computer Science
Columbus State University
Columbus, GA 31907, USA
cturnitsa@gmail.com

Keywords

M&S interoperability standards

Abstract

During every standard workshop or event, the examples of working interoperability solutions are used to motivate for 'plug and play' standards for M&S as well, like standardized batteries for electronics, or the use of XML to exchange data between heterogeneous systems. While these are successful applications of standards, they are off the mark regarding M&S interoperability. The challenge of M&S is that the product that needs to be made interoperable is not the service or the system alone, but the model behind it as well. The paper shows that the alignment of conceptualizations is the real problem that is not yet dealt with in current interoperability standards.

1 Introduction

TO ANSWER THE QUESTIONS OF HOW AND WHY MODELING AND SIMULATION (M&S) INTEROPERABILITY ARE DIFFERENT FROM OTHER INTEROPERABILITY DOMAINS, WE HAVE TO GAIN A BETTER UNDERSTANDING OF WHAT MAKES M&S SPECIAL FIRST. IN OTHER WORDS, WE NEED TO UNDERSTAND THE EPISTEMOLOGY OF M&S AND ANSWER THE QUESTION IF AND HOW IT IS DIFFERENT FROM OTHER RELATED INTEROPERABILITY DOMAINS. TO ANSWER THIS QUESTION, WE FURTHERMORE LIMIT OUR DISCOURSE AND FOCUS ON M&S SUPPORTING COMPUTER SIMULATIONS AND INFORMATION TECHNOLOGY (IT) INTEROPERABILITY DOMAINS.

2 Current Interoperability Standards

One of the most often used examples for solved interoperability challenges are batteries. There is hardly a workshop on interoperability in which it is not used: based on the standard that defines measurements like size, electronic data, voltage, and ampere, the same battery can power a radio, flashlight, night vision goggles, or the proverbial toy bunny. Another example closer to software is the use of the Extensible Markup Language (XML) to exchange data between heterogeneous systems. The XML standard uses basic standard foundations, so that many heterogeneous systems can support them easily (like being fully Unicode compliant), but is extensible to support complex information exchange needs.

The final examples of working interoperability solutions are web services and cloud computing. Although different in their implementation, the underlying conceptual ideas are comparable: a service is well defined by its interface (input and output parameters) and, if necessary, by additional constraints, such as timing, synchronization points, and more. The semantic markup for services OWL-S [1] defines three categories needed to describe services (as shown in figure 1):

- With the *ServiceProfile*, the service presents "what the service does." As specified in OWL-S, [1] this includes the description of what is accomplished by the service, limitations on service applicability and quality of service, and requirements that the service requester must satisfy to use the service successfully.
- Within the *ServiceGrounding* definition, the service supports different ways "how to access it." In this part, communication protocols, message formats, and other service-specific details such as port numbers are specified.
- Finally, a service is described by a *ServiceModel* that defines "how the service works." This description fulfills the tasks of detailing the semantic content of requests, the conditions under which particular outcomes will occur, and, where necessary, the step by step processes leading to those outcomes.

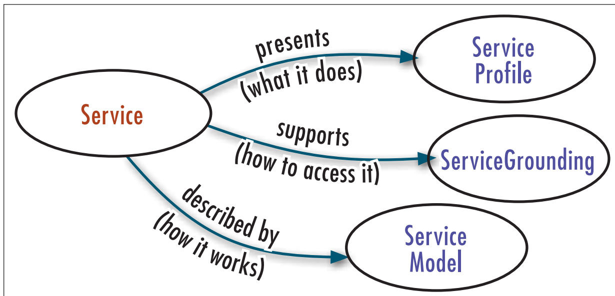

Figure 1: OWL-S

The authors showed in "Ontology Driven Interoperability – M&S Applications," [2] that OWL-S is one of the most advanced available standards supporting interoperability for M&S applications. These findings were based on research conducted in support of the Extensible Modeling and Simulation Framework (XMSF) initiative that evaluated the applicability of web-based standards to drive interoperability for M&S [3, 4]. All these standards are applied successfully, including in the M&S domain.

In addition, we have M&S specific solutions that successfully have been standardized via SISO and IEEE, namely the Distributed Interactive Simulation (DIS) protocol [5], standardized in IEEE1278, and the Modeling and Simulation High Level Architecture (HLA) [6], standardized in IEEE1516. Despite significant success stories, M&S interoperability standards seem to have "hit the wall." In recent years, no break-through has been accomplished. Instead, we look at gradual improvements, but the promised "plug and play" functionality, as suggested by the battery example, is still a dream. What is this wall? In the next section, we will have a look at where we are and how we got there, and this may help to better understand where the current challenge lies.

3 A Brief Historical Overview

In order to better understand the current view on M&S interoperability standards it is necessary to review the history of distributed simulation.

The use of simulators and simulations in the armed forces has a long history, including the use of strategic games, life exercises, and board games. However, with the advance of computers, a new era of computer simulation and simulators began. The birth of simulation standards can be seen with the creation of the Simulator Network SIMNET, which was a project of the Defense Advanced Research Project Agency (DARPA). Developed between 1980 and 1990 in collaboration with DARPA and the U.S. Army, SIMNET showed how to combine individual tank simulators of the Combined Arms Tactical Training System (CATT) to enable tank crews to operate side by side in a common synthetic battle space. The individual simulators represented weapon systems on this common virtual battlefield that had a well defined set of actions and interactions: tanks could move, observe, shoot at each other, exchange radio communication, etc. Individual activities led to status changes that were communicated via status reports. Interactions were communicated via messages.

If two tanks engaged in a duel, the order of activities and the data to be exchanged between these entities were well defined. The shooter decided to engage the victim. He moved his weapon system, and potentially platform components like a turret and a cannon into the best direction, always updating his status, so that other simulators could update their visualization showing that the tank/turret/cannon is moving. He shot at the victim. This data was sent to everyone as well. All observing systems could visualize the shooting (smoke, flash, etc.). The victim also received information on the ammunition shot at him such as velocity, angle, etc. The victim computed the result of this engagement – like catastrophic kill, movement kill, firepower kill, etc. – and communicated the result. All observers, including the shooter, updated their visualization of the victim (like being on fire, smoking, or no effect beside the impact explosion). Based on his assessment of the effect, the shooter could reengage, or continue with a new task. The tasks of who is doing what based on what data was well understood by those simulators embedded as individual independent entities in the common battle space.

As the set of information exchange specifications could be well defined, this resulted in the idea to standardize these messages, which led to the IEEE1278 Distributed Interactive Simulation (DIS) standard: the Protocol Data Units (PDUs) captured syntactically and semantically all possible actions and interactions based on the idea that individual simulators represent individual weapon platforms. Only later, instead of individual platforms also groups and aggregates (like platoons or companies) were accepted as receivers and producers of such PDUs, but these groups were understood as individual entities in the battle space as well. DIS is still successfully used and supported by a large user community.

In parallel to the simulator community that serviced the tactical level training needs, higher commands started to use computer assisted exercises (CAX) to support their command post exercises as well. Ever since Baron von Reisswitz introduced the "Kriegsspiel" during his tenure as war counselor in Prussia in 1811, [see figure 2] combat models were used to train command post officers. These exercise support games had well defined units with well defined interactions, all ruled by very detailed tables enumerating in detail the effects of each possible interaction.

Figure 2: Kriegsspiel (War Game)

The computer based successors also required a distributed capability, in particular to support higher command training of distributed facilities. As the earlier war games, these computer simulations represented aggregates on the operational level, like battalions and brigades. Again, they were interpreted as individual entities on the battlefield. MITRE developed the Aggregate Level Simulation Protocol (ALSP) to exchange information between these simulation systems.

However, unlike the simulator solution, in ALSP several units were represented in each system. When these systems were connected, the protocol ensured that each simulated aggregate had exactly one simulation system that was responsible for updates. In all other simulation systems, the respective aggregate was "ghosted," which means that a simulation object was instantiated in the simulation system, but it was tagged to be controlled by another system and was only used to make decisions for the aggregates controlled by the system, e.g., where to place surveillance radars in the surveillance simulation systems based on the distribution of tanks in the combat simulation system.

As the diversity of aggregates were higher than that of platforms and in addition differed from exercise to exercise, ALSP did not standardize the messages to be exchanged. Instead, ALSP standardized the syntax to be used, but allowed to specify the semantics (meaning of information exchange) in special formats that today would be described as metadata allowing the interpretation of the exchanged data. While during the time of "das Kriegsspiel" the possible units were limited to a set of categories supported by all armies, such as infantry, cavalry, artillery, scouts, etc.), ALSP provided a frame to communicate the participating entities (or better aggregates), possible interactions, and the effects of such interactions.

The High Level Architecture (HLA) was developed to replace both approaches – DIS and ALSP – with a new and merging approach. Originally developed within the U.S. DoD, the final version HLA 1.3 NG was handed to IEEE for international standardization, resulting in the IEEE 1516-2000 and was only recently updated to the HLA evolved standard IEEE 1516-2010. Significantly influenced by recent new methods developed in computer science in general and software engineering in particular, a very flexible protocol was developed providing more flexibility and configurability than both of its predecessors.

The HLA interoperability standard was focused to maximize the flexibility for all kinds of M&S application domains and supported M&S paradigms. The information exchange requirements of a federation are captured in the Federation Object Model (FOM). This model defines all persistent objects and their attributes and transient objects and their parameters that can be exchanged between participating simulations. While persistent objects have to be created and then are updated (and the responsibility can be switched between the participating simulation systems during runtime), transient objects are like messages created in case of need and only used once.

Six service groups are provided as a result of generalizing the synchronization challenges ensuring that all the required information needed is delivered at the right time to the right simulation system. The purpose of *Federation Management* is to determine the federation. Federates join and leave the federation using the functions defined in this group. The purpose of *Declaration Management* is to identify which federate can publish and/or subscribe to which information exchange elements. This defines the type of information that can be shared. The purpose of *Object Management* is managing the instances of shareable objects that actually are shared in the federation. Sending, receiving and updating belong to this group. The purpose of *Data Distribution Management* is to ensure the efficiency of information exchange. By adding additional filters this group ensures that only data of interest are broadcasted. The purpose of *Time Management* is the synchronization of federates. The purpose of *Ownership Management* is to enable the transfer of responsibility for instances or attributes between federates.

HLA significantly increased the flexibility of simulation federation definitions. Instead of being limited to predefined information exchange groups, the developer can specify the objects and interactions and can even support different time model philosophies. It neither assumes the level of resolution nor does HLA assume the partition of the battle space into tactical unit or the phasing of a supported operation. HLA supports component level simulation, platform level simulation, and all levels of aggregation

4 What makes M&S Special?

The last section showed the development of M&S interoperability standards with an increase in flexibility and support of different M&S paradigms. *However, the mental model behind all these developments remained the idea of one shared virtual battle space that was populated by individual independent aggregates and/or platforms that interact with each other.*

These individuals, or group of individuals, were well defined by their own actions and interactions with each other, which could be represented by boundaries around the individuals – or a group thereof – being the boundaries of the simulation system that was responsible for their simulation and the specification of data that could be exchanged via these interfaces. The individual becomes a black box that can represent a simulated system or a live system, as long as the interface specifications are fulfilled. They build a perfect participation of the battle space and what goes on within it.

However, with the introduction of the flexibility provided by HLA, we opened Pandora's Box. While DIS enforced the one shared battle space view by defining syntax and semantics of the PDUs, and while ALSP ensured with the ghost concept that simulated entities are only available once (and merely reflected in other simulation), HLA said farewell to this paradigm.

The interoperability view of HLA is indeed that the same objects are represented in two simulations, and that these objects are represented as the persistent objects in the FOM. If an attribute changes in one of the representing systems, the attribute change is communicated via updates. Nonetheless, we have as many instances of the same object as we have implementing simulations.

As every participating simulation has been developed for a special purpose, it is unlikely that the representations are going to be identical. Actually, it is very likely that the scope will be different, which means that attributes needed to describe the object in one context are meaningless and therefore not even modeled in another context. A simulation system written to support combat operations will use a different model to represent a main battle tank than a simulation system written to support logistics. A radio modeled for support of communications of dismounted infantry will look different than one modeled to be evaluated in the light of electronic warfare.

As all models are simplifications and abstractions of a perception of reality in order to support a certain task, they have to be different. And as simulations are implementations of models, the implemented objects will look different as well:

- Simplification takes things away. Even if we start with a common definition of a real object, we will chop off different aspects of this real object in the process of simplification. Therefore, we end up with different scopes.
- Abstraction in general leads to models with different structures and resolutions. Again, even when starting with identical observations, the detail represented in two models is likely to be different. Even worse, if aggregation is part of the abstraction process, the resulting aggregates may look very different, resulting in different structures.

To show the challenges deriving from abstraction, we already introduced the example of 'number world' and 'letter world' in "Federated Ontologies Supporting a Merged Worldview for Distributed Systems," [7]: a system exposes the six observables *a1, a2, a3, b1, b2,* and *b3*. In letter world, the three observables *a1, a2,* and *a3* are abstracted into attributes of *A,* and *b1, b2,* and *b3* are abstracted into attributes of *B*. In number world, the abstraction of *a1* and *b1* results in One, *a2* and *b2* in *Two*, and *a3* and *b3* in *Three*. Both are plausible models, but they are quite different. While on this lowest level the common attributes are still derivable, supporting the information exchange between the abstractions, what if the resolution for the model is changed and only *A, B, One, Two, a*nd *Three* remain in the models?

Even when starting from an agreed description of reality that comprises all possible attributes that a participating simulation may be interested in, the process of simplification and abstraction is going to produce very different modeling results. Furthermore, not everything going on in the real world referent is observable, even when perfect sensors are assumed. Then it depends on additional assumptions how to model these "hidden" attributes, and as no reference for them can exist by definition, different models may easily result from observing the same system with the same sensors.

It becomes worse when we take the aspect of perception into account. In this paper, perception is the physical-cognitive process of observing reality and building a conceptualization of the observation.

- The physical aspect defines what attributes of an object are observable with the sensoric system of the observer, or more general, the information about the object that can be obtained in the process of perception (this can include gaining insight from literature, discussions with colleagues, etc.).
- The cognitive aspect is shaped by the education and the knowledge of the observer. In order to conceptualize the observation the observer needs to have an internal model he can map this observation to. A physician will see more in an x-ray than a layman. An educated mechanic sees more in an engine than a novice. The subject matter expert has more internal models to explain an observation in his field than others do.

Physical and cognitive perception will therefore shape the model and resulting simulation significantly, even more than simplification and abstraction does, as perception results in a different starting point: We no longer can assume that everybody starts from a common reality, we all have individual perceptions thereof! This common conceptual starting point, however, is the necessary requirement and builds the conceptual foundation for developing a common information exchange specification between simulation systems.

As long as we are starting to support a common theory, like we did in the successful example of a common battle space following the laws of Newtonian physics, we can always track our models and resulting simulations back to the common ground defined by this theory. We can observe with more accuracy, we can model with higher resolution, and we can add "missing" attributes (those that are described in the theory, but not used in individual models). Actually, following the philosophy of science, ***a simulation system is an***

executable hypothesis or – once proven to be valid – an executable theory![1]

Mathematically, a simulation system is a production system representing the theory: starting with the initialization data, we apply production rules encoded as functions, procedures, and methods. Every state that is simulated is a valid state represented by the theory encoded as the simulation. This is equivalent to assigning TRUE and FALSE values to such states: if a certain state can be produced (and we can even add the constraint of 'within a given time') it is true, otherwise it is false. The M&S interoperability challenge comprises the task to ensure the logical equivalence of all representations of an object in the federation.

Again, we can start with assuming that we start from the common ground of a common and accepted description of reality in the form of an object model that can serve as the *Übermodell* from which all simulation representations can be derived by pruning and aggregating. We show in [8] how to apply model-based data engineering to construct the model from the information exchange needs, but this algorithm and similar ones only work if we can assure that all models started from the same common ground. And even then, strange effects can be observed.

To better address the challenges, a formal approach to simulation interoperability [9] has been developed and applied. Without going into the mathematical details, this approach showed significant shortcomings of our current M&S interoperability approaches. From the data modeling theory, we know two categories of dependencies of two objects *A* and *B*:

- *A* is existential dependent on *B* if *A* can only exist if, and only if, *B* exist.
- A is transformational dependent on *B* if *A* needs to be changed if *B* is changed.

None of our current standards support this kind of dependency. We can have a perfect FOM communication in every aspect of *A* and *B*, but we cannot communicate the dependencies. If now two simulation systems implement *A* and *B* identical despite the dependency, we can end up with two versions of truth in the same federation, if we delete or change *B*: in the simulation system that implements the dependency, the deletion or change of *B* implies the deletion or change of *A* as well; but that is not the case in the system that does not implement the dependency. While *A* continues to exist in one federate, it ceases to exist in another, and all under valid current standard conditions.

So far, all of the examples can be understood as examples that someone made a mistake: an important detail was not implemented, a model was over-simplified, an important relation was overlooked, etc. In addition, our focus has been on physical-technical models. As these models have a common referent, this 'real world' can always be used to find out if a model is sufficient or 'realistic.' The assumption here is, however, that truth exists on its own, it is independent of the observer, and reality is separated from the individual who observes it. The traditional scientific method is rooted in this world view called positivism. There exists one world and one truth, and it is possible to find this truth by observation and experimentation. This world view worked well for Newtonian physics and the physical-technical models that model it based on this common ground of a common theory.

However, the M&S community is currently starting to look into better approaches to support human, social, cultural, and behavioral modeling. Davis summarizes his research in as follows: *"Fortunately, the social science literature has a great deal to offer. However, the literature is fragmented along boundaries between academic disciplines, between basic and applied research, and between qualitative and quantitative research. ... Realistically, the research base is not mature enough to support a coherent expression of the body of knowledge. The uncertainties and disagreements are profound, on both subject-area facts and even the nature of evidence and the appropriateness of different methodologies. Those hoping to find a nicely compiled body of knowledge that can be used to write computer models will be disappointed. Further, they will often find that there are multiple competing "theories." And, even if a particular "theory" is chosen, it will be found upon inspection to involve numerous variants and uncertainties."*[10] These findings are supported by other researchers as well.

[1]Using the scientific method, a hypothesis becomes a theory only after it has been repeatedly tested and confirmed via real world data using experiments. This is in contrast to the every day use of the term "theory," where it is often understood as a collection of ideas that are not yet proven. In both cases, however, internal consistency is mandatory. In the rest of this paper, we will assume that our models are indeed grounded in theory that has been proven to be relevant and is backed by empiric evidence to avoid having to discriminate explicitly between hypothesis and theory. Whenever this is not the case, it only amplifies the implications of misuse of current practice.

To make things worse for the M&S engineer, we no longer deal with positivism in this domain, but with interpretivism. Interpretivism holds the belief that truth is a construct of the observer. Reality is relative and cannot be separated from the individual who observes it. The majority of social and human sciences subscribe to interpretivism. That means that we have to take the aspect of perception into account when evaluating if two simulation systems can operate together.

If two simulation systems implement competing theories, they can never become interoperable, as the underlying mathematical production systems produce different versions of truth. This does not make one of them wrong or the other solution better. It is a fact of life and no interoperability standard can solve this challenge: we simply do not know, and in some cases even cannot know what is needed to solve the conflict between competing theories. The challenge of the M&S engineer and of supporting M&S interoperability standards is to ensure that no competing theories (and following competing simulation systems) are federated to produce a common federation model.

In summary, our challenges lay often on the modeling side. It is understood that while modeling targets the conceptualization, simulation challenges mainly focus on implementation, in other words, modeling resides on the abstraction level, whereas simulation resides on the implementation level. Our interoperability problems are derived from the abstraction level, but our standards only focus the implementation level.

5 Implications

One of the first things to do about these challenges is to raise the awareness regarding them [11]. It would be naïve to apply standards that were developed for physical-technical models based on a common theory representing the positivistic worldview to integrate socio-psychological models derived from competing theories representing interpretivism and expect valid results. As pointed out in "Towards Methodological Approaches to meet the Challenges of Human, Social, Cultural, and Behavioral (HSCB) Modeling," [10], the best way ahead may be to live with contradicting models. It is highly unlikely that we will be able to address all problems with one common approach based on a common theory resulting in a consistent federation. It is much more likely that the multi-simulation approach based on multi-resolution, multi-stage, and multi-models envisioned by Yilmaz et al. [12] needs to be exploited to support the analysis of these multi-facetted challenges we are faced with as a community.

Generally, it is necessary to focus more on the abstraction level (the modeling) when building federations than on the implementation side. Our approaches to M&S interoperability have been shaped by software engineering and computer engineering principles that are necessary, but not sufficient. The alignment of conceptual constraints is not supported enough by the current approaches and standards. As we are connecting simulated things we need transparency of what we are simulating, as the real world referent use in other interoperability domains has been replaced in the modeling phase by its representing conceptualization in the M&S interoperability domain.

It is worth mentioning that it is possible to apply competing methods in one federation if they are coupled via a common theory. For example, two agents implementing competing theories can be coupled by purely exchanging their actions in the physical world. The underlying conceptual model, however, is well aware that one agent implements one theory, the other agent implements another theory. If we know the agents run into oscillating states or produce inconsistent results, this is part of the underlying common conceptual model that allows for this to happen, as both theories are contained in their agents.

Another aspect is the applicability of current methods for validation and verification to human, social, cultural, and behavioral modeling. As pointed out in the paper, there are many competing hypotheses, and the dearth of real-world data as well as the epistemological nature of simulation forcing us into interpretivism. However, as in interpretivism truth is subjective to the observer and not objective for the observation, validation becomes relative as well. As a consequence, socio-psychological hypotheses may remain in general objectively untestable and cannot graduate into general common theories. This challenge increases with the complexity of proposed solutions and the number of participating hypotheses, resulting in uncertainties and risks adverse to successful application of federated approaches.

The fundamental difference between M&S systems and other software systems is that M&S adds the level of conceptualization to what needs to be aligned. While other software systems connect with the real thing or support the real thing, in M&S systems the "conceptualization is the real thing" that is simulated: the model is the reality of the simulation. If we use technical means to make two simulations interoperable on the implementation level that are based on competing theories, we merge things together that do not belong together, and instead of creating a solution, the result is a conceptual chimera … or worse. However, it is well known that conceptual problems cannot be solved with technical solutions. More work is needed to make sure that the next generation of M&S interoperability standards contributes towards a solution of this category of challenges we are just becoming aware of.

Summary

After all this explanation we still did not have the answer to the question posted in the title of this paper: *How is M&S Interoperability different from other Interoperability Domains?* The answer is simple: M&S ***interoperability requires interoperability of the simulations*** – that is provided by the software engineering standards we focused on so far, including mediation of data representations, conversion of different unit of measures, mappings between different styles of enumeration, etc. – as well as ***composability of the models*** [13]. We have to ensure transparency of our conceptualizations, as they represent the real world references for the simulation. While other interoperability domains connect real things and can refer to the same real world referents, M&S interoperability connects conceptualizations, and we have to understand what the participating systems concepts look like in order to operate together. The same real world referent can have different conceptualization in different models.

The Levels of Conceptual Interoperability Model (LCIM) [14] addresses these issues for some time. Only interoperability domains that are model-driven have the second challenge.

- The battery is plugged into the system and either connects to the socket or does not. As long as power is left it is provided. The battery does not need a model of what it is powering.
- A web service that connects the fill out order for books with the inventory list of Amazon doesn't need a common model: it connects the real list with the real database. Integratability and Interoperability is all it has to be concerned about. The ordered book is either there, or it is not.
- If two simulation systems exchange data, they need to support common concepts of a model. As such, there is a conceptual overlap of the models implemented by the simulation systems. Within this overlapping area, the six interrogatives Who, What, Where, When, Why, and How need to be consistent.

In other words, for the simulation systems, the implemented model is reality. In order to couple two simulation systems, there needs to be an overlap; otherwise both systems have nothing in common to exchange data about. This overlap must be consistent, which means that the results of computations regarding the research questions must be identical. If this is not the case, we end up with two versions of truth in the federation. This problem of model-based reality is unique to M&S. Consequently, the application of software engineering standards cannot solve this problem. Therefore, a new generation of M&S standards needs to support the alignment of models to support and ensure not only interoperability, but also composability, in a form that allows the automation of such processes wherever possible.

This new generation of M&S standards must ensure the transparency of models, not only the mediation of simulations. While standards for real components can focus exclusively on the exchange of data, model-based components must ensure that the same concepts are represented consistently in all participating components. This problem does not occur outside of the model-based world. If the same real world referent is modeled or changed inconsistently in model-based components, this introduces inconsistencies on the conceptual level that are not necessarily observable. While in real components the real world reference exist only once, in model-based components the concept of this one real component can exist independently in every component.

Even more importantly may become the recognition that simulations are implemented theories, as it is the case when human behavior is modeled and implemented. As long as the simulation systems to be federated support consistent theories, the upcoming interoperability challenges can be resolved. In new application domains, such as the emerging domain of HSCB, many conflicting theories exist. This is a conceptual block that cannot be solved by M&S interoperability standards. Federating such models into one common federation must lead to inconsistencies and meaningless results! Instead, alternative uses of alternative theories need to be supported by new approaches like the proposed Multisimulations [12].

This requires a domain of new standard efforts: the efficient and effective support of exploratory analysis under uncertainty and disagreement, and supporting development of strategies that are flexible, adaptive, and robust, as requested by Davis in [10]. SISO should address these challenges in respective efforts.

Although current standards are not sufficient, they are necessary and are building a strong foundation new approaches can extend. The authors made first recommendations in "Conceptual Modeling for Composition of Model-based Complex Systems" [8] and "Using a Formal Approach to Simulation Interoperability to Specify Languages for Ambassador Agents," [9], extending the work presented in [12]. It is now time to focus on building better tools to support the work of the M&S engineer sufficiently well to help avoid mistakes and guide him/her to better solutions in support of the customer not only in the military domain.

Acknowledgement

Significant parts of the underlying research were funded by the US Joint Forces Command Training Directorate and the US Modeling and Simulation Coordination Office.

The authors furthermore thank Hans U. Mair, Johns Hopkins University/Applied Physics Laboratory, for his critical review and constructive remarks regarding theories, validity, and hypotheses.

The paper was originally published as paper 11S-SIW-008 during the 2011 Spring Simulation Interoperability Workshop in Boston, MA, April 4-8, 2011. It has been awarded with a SIWzie and is on the Recommended Reading List of the workshop

References

[1] David Martin, Mark Burstein, Jerry Hobbs, Ora Lassila, Sheila McIlraith, Srini Narayanan, Massimo Paolucci, Bijan Parsia, Terry Payne, Evren Sirin, Naveen Srinivasan, and Katia Sycara (2004). "OWL-S: Semantic Markup for Web Services," *W3C Member Submission 22 November 2004,* last accessed January 2011 at http://www.w3.org/Submission/OWL-S

[2] Andreas Tolk, Saikou Diallo, and Charles Turnitsa (2006). "Ontology Driven Interoperability – M&S Applications," *Whitepaper in support of the I/ITSEC Tutorial 2548,* VMASC Report, Old Dominion University, Suffolk, VA

[3] Don Brutzman, Michael Zyda, J. Mark Pullen, and Katherine L. Morse (2002). "Extensible Modeling and Simulation Framework (XMSF) - Challenges for Web-Based Modeling and Simulation," *Workshop Report*, Naval Postgraduate School, Monterey, CA, 22 October 2002

[4] Curtis Blais, Don Brutzman, David Drake, Dennis Moen, Katherine Morse, Mark Pullen, and Andreas Tolk (2005). "Extensible Modeling and Simulation Framework (XMSF) 2004 Project Summary Report," *Project Report NPS-MV-05-002*, Naval Postgraduate School, Monterey, CA, 28 February 2005

[5] *Institute of Electrical and Electronics Engineers. IEEE 1278 Standard for Distributed Interactive Simulation,* IEEE publication, Washington, DC.

[6] *Institute of Electrical and Electronics Engineers. IEEE 1516 Standard for Modeling and Simulation* High Level Architecture, IEEE publication, Washington, DC.

[7] Charles D. Turnitsa, and Andreas Tolk (2007). "Federated Ontologies Supporting a Merged Worldview for Distributed Systems," A*ssociation for Advancements in Artificial Intelligence (AAAI) Fall Symposium, Technical Report FS-07-06*, AAAI Press, Menlo Park, CA, pp. 116-119

[8] Andreas Tolk, Saikou Y. Diallo, Robert D. King, Charles D. Turnitsa, and Jose Padilla (2010). "Conceptual Modeling for Composition of Model-based Complex Systems" *in* Stewart Robinson, Roger Brooks, Kathy Kotiadis, and Durk-Jouke van der Zee (Eds.) *Conceptual Modeling for Discrete-Event Simulation,* CRC Press, pp. 355-381

References (continued)

[9] Andreas Tolk, and Saikou Diallo (2010). "Using a Formal Approach to Simulation Interoperability to Specify Languages for Ambassador Agents," *Proceedings of the 2010 Winter Simulation Conference,* B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan (Eds.), IEEE CS Press, pp. 359-370

[10] Andreas Tolk, Paul K. Davis, Wim Huiskamp, Gary L. Klein, Harald Schaub, and James A. Wall (2010). "Towards Methodological Approaches to meet the Challenges of Human, Social, Cultural, and Behavioral (HSCB) Modeling," *Proceedings of the 2010 Winter Simulation Conference,* B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan (Eds.), IEEE CS Press, pp. 912-924

[11] Andreas Tolk (2010). "M&S Body of Knowledge: Progress Report and Look Ahead," *SCS M&S Magazine*, Vol. 1, No. 4, October

[12] Levent Yilmaz, Tuncer Ören, Alvin Lim, and Simon Bowen (2007). "Requirements and Design Principles for Multi-simulation with Multiresolution, Multistage Multimodels." *Proceedings of the 2007 Winter Simulation Conference,* S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, (Eds.), IEEE CS Press, pp. 823-832

[13] Andreas Tolk (2006). "What comes after the Semantic Web, PADS Implications for the Dynamic Web," *Proceedings of the 20th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation (PADS 2006),* Singapore, May 2006, pp. 55-62

[14] Andreas Tolk, Charles Turnitsa, and Saikou Diallo (2008). "Implied Ontological Representation within the Levels of Conceptual Interoperability Model," *Journal of Intelligent Decision Technologies (IDT)*, 2(1):3-19

Authors' Biographies

Andreas Tolk

Andreas Tolk is Associate Professor for Engineering Management and Systems Engineering at Old Dominion University. He is senior member of IEEE and SCS. His email is atolk@odu.edu.

Saikou Diallo

Saikou Diallo is Assistant Research Professor at the Virginia Modeling Analysis and Simulation Center of Old Dominion University. His email is sdiallo@odu.edu.

Jose Padilla

Jose Padilla is Assistant Research Professor at the Virginia Modeling Analysis and Simulation Center of Old Dominion University His email is jpadilla@odu.edu.

Chuck Turnitsa

Chuck Turnitsa is faculty member at the TSYS School for Computer Science at Columbus State University. His email is cturnitsa@gmail.com.