A New Simulation Infrastructure Supporting the Next Generation of Simulation-Based Testing and Training Applications

Dr. Andreas Tolk
The MITRE Corporation
903 Enterprise Pkwy #200, Hampton, VA 23666
UNITED STATES

atolk@mitre.org

Omar Valverde & Joe Clapis
The MITRE Corporation
202 Burlington Road, Bedford, MA 0130-1420
UNITED STATES

ovalverde@mitre.org jclapis@mitre.org

ABSTRACT

The next generation of simulation systems contains new requirements for simulation infrastructures significantly different from current solutions. Current simulation interoperability standards, such as IEEE 1278 Distributed Interactive Simulation (DIS) and IEEE 1516 High Level Architecture (HLA), are updated on a regular basis by standardisation bodies. Nonetheless, their conceptual frameworks and technical structures are rooted in the last decades' technology and often reach their capability limits when applied to support the most recent system engineering challenges. NATO continues to support and apply these standards, despite them being no longer sufficient to support training, education, and testing of next generation defence systems. We present concepts needed to implement a novel simulation infrastructure, centred around an Information Exchange Services Matrix (IESM) and accompanied by supporting services. This allows the application of new technologies and accommodates the evolution of these technologies in support of composable solutions that provide both "fair fight" training and analysis. These general concepts are applicable to the development of a Joint Simulation Environment (JSE) for the next generation of simulation systems. This paper describes the concepts behind the IESM and their supporting services.

1.0 INTRODUCTION

The concept of modelling and simulation (M&S) has been successfully applied in the context of developing new solutions, testing related components and systems, and training future users. To support the reusability of existing solutions for such applications, several interoperability standards were developed and successfully applied. Mainly driven by the defence simulation community, the IEEE 1278 Standard family defining the Distributed Interactive Simulation (DIS) concepts and processes and the IEEE 1516 Standard family defining the High-Level Architecture (HLA) are among the most notable solutions; the latter is even captured in the military standard STANAG 4603. The NATO M&S community relies heavily on standards and agreements, as national contributions should be easily integrated into a multinational solution. To test conformity with standards, NATO only recently provided a new HLA certification process [1]. Overall, the HLA standard has been a success for coalition interoperability improvements, with standards being continuously improved and re-evaluated by users and the standardisation bodies.

The DIS concept originates from SIMNET [2], which was developed in the late eighties. The ideas and concepts underpinning the HLA also date back to the late eighties and early nineties [3], although some newer concepts have been integrated into the latest version, such as better support for web-based—and recently even cloud-based—solutions, together with greater flexibility regarding the information exchange specifications.

Since its introduction, the HLA has been successfully applied in many domains outside of the military, such as in medical environments, transportation, and other domains, such as those discussed in [4]. However, recent systems engineering challenges—such as development of systems to become part of the Internet-of-Things (IoT) [5], initiatives to support the Industry 4.0 efforts, developing and testing of cyber-physical systems (CPS) [6], or next generation defence systems [7]—often reach the limits of the principles and concepts of distributed simulation. Common challenges observed in these new systems engineering domains are as follows:

- *Significant increase in data utilised by the systems:* IoT and Industry 4.0 systems are networked systems by design, providing the foundation for smart houses, smart cities, smart production facilities, for example. As such, their information exchange requirements are far greater that those of traditional solutions. Similarly, the next generation of defence systems are not only simple effectors on the battlefield, but also provide, disseminate, and utilise information for the distributed battle management process. Likewise, cyber-physical systems are data-intensive as they provide decentralised decision and management structures. Current simulation interoperability standards were not designed for the increase in data required by these systems.
- *Multi-modality of the systems:* Traditional systems usually receive their input via a small set of well-defined interfaces, such as one or two well-defined sensors onboard the system. Current systems, such as CPS, have a multitude of sensors and use a high variety of modalities to communicate with humans, or exchange information with other systems. The same information must be communicated via a large variety of channels, each with its own information exchange formatting requirements. Current simulation interoperability standards focus on one information exchange medium supporting a common structure (such as the protocol data units of DIS or the objects defined in the federation object model of HLA). Multi-modality has not been sufficiently considered.

In addition to these changes in the systems themselves, the nature of operations in which they are applied have also become more complex. Since the current simulation interoperability standards were introduced more than a decade ago, systems engineering has developed new methods and tools to cope with system of systems challenges, emergence in complex environments, and has recently started to study complex systems engineering more intensively, as discussed in more detail in [8].

Although the simulation community has expressed considerable interest in leveraging distributed systems and infrastructures—such as cloud-based services in support of planning, conducting and evaluating complex operations in complex environments (as documented in [9]), and including more extensive use of IoT or Industry 4.0 related concepts—these trends have not yet become part of new standardised simulation infrastructure recommendations. It is suggested that every solution that does not address these new complexity issues will fall short of providing adequate support to its user—including industry, government, and military.

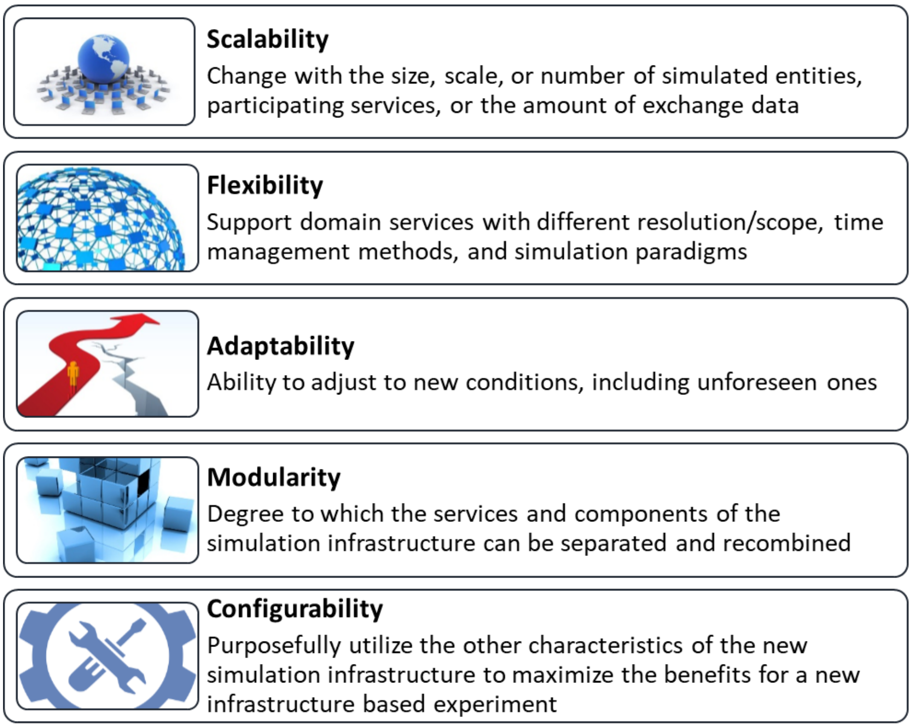
In summary, the evolution of systems' technology as well as the way it is utilised has led to an unforeseeable increase in data exchange requirements, multi-modality, and an increase in distributed operations in more complex environments. In the defence community, additional requirements contribute to the growing number of challenges, such as the need for multi-domain operations across service borders and—in the context of NATO—across national forces as well. A new simulation infrastructure is required that will address these challenges and provide actionable solutions. This new simulation infrastructure will need to take advantage of the same technology advancements that provide interoperability and composability. This will pave the way to incorporate newer M&S technologies and continue to reuse legacy solutions in a new context. At the same time, these new developments in systems' technology not necessarily render existing solutions obsolete. Previously successful concepts that successfully provided solutions for simulation infrastructure challenges need to be integrated and utilised, such as time management services. Therefore, the new infrastructure shall allow the reuse of calibrated existing solutions side by side with the integration of technically improved solutions.

2.0 SIMULATION INTEROPERABILITY SERVICES

It goes beyond the scope of this paper to address all relevant contributions of past and current approaches to provide a standardised set of services and processes to support simulation interoperability. The interested reader is directed to [10] for additional information and references to the details of IEEE 1278 and IEEE 1516, as well as the Aggregate Level Simulation Protocol (ALSP), the Test- and Training Enabling Architecture (TENA), the Extensible M&S Framework (XMSF), or the use of the Semantic Web.

2.1 Tenets for the Simulation Infrastructure

The NATO M&S Group Study MSG-136 (2014-2017) documents the recent technical development in cloud computing technology and service-oriented architecture. This study shows how these developments and architecture offer opportunities to utilise M&S capabilities more effectively to satisfy NATO critical needs [11]. This is of interest for new infrastructure solutions; particularly how they should not become a 'monolithic bottleneck' at the centre of otherwise scalable and composable service solutions. Instead, new solutions should expose key characteristics, as presented in Figure 1.



Scalability
Change with the size, scale, or number of simulated entities, participating services, or the amount of exchange data

Flexibility
Support domain services with different resolution/scope, time management methods, and simulation paradigms

Adaptability
Ability to adjust to new conditions, including unforeseen ones

Modularity
Degree to which the services and components of the simulation infrastructure can be separated and recombined

Configurability
Purposefully utilize the other characteristics of the new simulation infrastructure to maximize the benefits for a new infrastructure based experiment

Figure 1: Architecture Characteristics for the Future Simulation Infrastructure

The envisioned concepts should therefore be a set of composable services fulfil the requirements for scalable, flexible, adaptable, modular, and configurable solutions. The infrastructure must ensure that

1. All information needed by the simulation services is provided
2. Only the information needed will be provided
3. The information is provided in a secure and timely manner

Furthermore, a short analysis of current approaches shows conceptual ideas that should be reused to support and leverage recently developed technologies. Our proposed approach to implement the concepts described in this paper takes advantage of current approaches within the new infrastructure, further evolving them, adapting them, or replacing them with new methods using most recent research insights in both the conceptual and technical domains. Our analysis led us to the conclusion that the services provided by the Runtime Infrastructure (RTI) of the HLA are particularly helpful in guiding conceptual development. The RTI (standardised within the IEEE 1516) provides functions in support of six service categories, depicted in Figure 2.

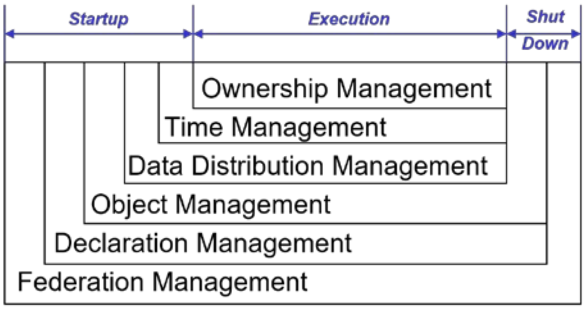


Figure 2: Services provided by the HLA Runtime Infrastructure

- Federation Management determines the federation. Federates join and leave the federation using the functions defined in this group.
- Declaration Management identifies which federate can publish and/or subscribe to which information exchange elements. This defines the type of information that can be shared.
- Object Management manages the instances of shareable objects that are shared in the federation. Sending, receiving, and updating belong in this group.
- Data Distribution Management ensures the efficiency of information exchange. Adding additional filters ensures that only data of interest are broadcast.
- Time Management synchronises the execution of the participation federates.
- Ownership Management enables the transfer of responsibility for instances or attributes between federates.

The RTI introduced innovative concepts to the community. While other interoperability solutions—such as DIS, ALSP, and TENA—all use standardised information exchange objects, the RTI utilises the Object Model Template to allow the declaration and sharing of every piece of information, as agreed between the participating federates. This allows the use of the same infrastructure for very different application domains beyond defence applications. The service groups provided the functionality to utilise this information exchange; however, the price for this flexibility was the need to agree on the semantics of the exchanged objects in additional administrative processes.

2.2 The Information Exchange Services Matrix

When the US Air Force decided to build the Joint Simulation Environment (JSE) as their next generation enterprise training and test capability, the engineers faced the same challenges described in the previous section: how to provide a simulation infrastructure that is fast, secure, and ready for data-intensive multi-modality systems? In partnership with the US Air Force, MITRE is developing a new simulation infrastructure enabling experimentation, rapid prototyping, and concept exploration capability to support testing, training, and acquisition decisions. Conceptual studies have resulted in the recommendation to build an Information Exchange Services Matrix (IESM) forming the core of this new simulation integration framework.

The IESM is designed as a composition of core services that builds an information broker for data-intensive multi-modality systems and their simulations, simulators, and synthetic environments needed for testing, training, and analysis. The derived requirements include the following: the ability to interface with live systems; to provide key simulation services for time management, communications, and data transformation across multiple scenario domains in a variety of atmospheric conditions; to allow simulation configuration and orchestration (without the need to involve a simulation specialist) facilitating data collection and analysis; and to enable multi-level security (MLS) solutions to support localised and distributed simulation operations with partners and allies.

To implement such a scalable, flexible, adaptable, modular, and configurable simulation infrastructure, MITRE identified the use of composable services as best practice. The justification for this decision, considering these characteristics, can be summarised as follows:

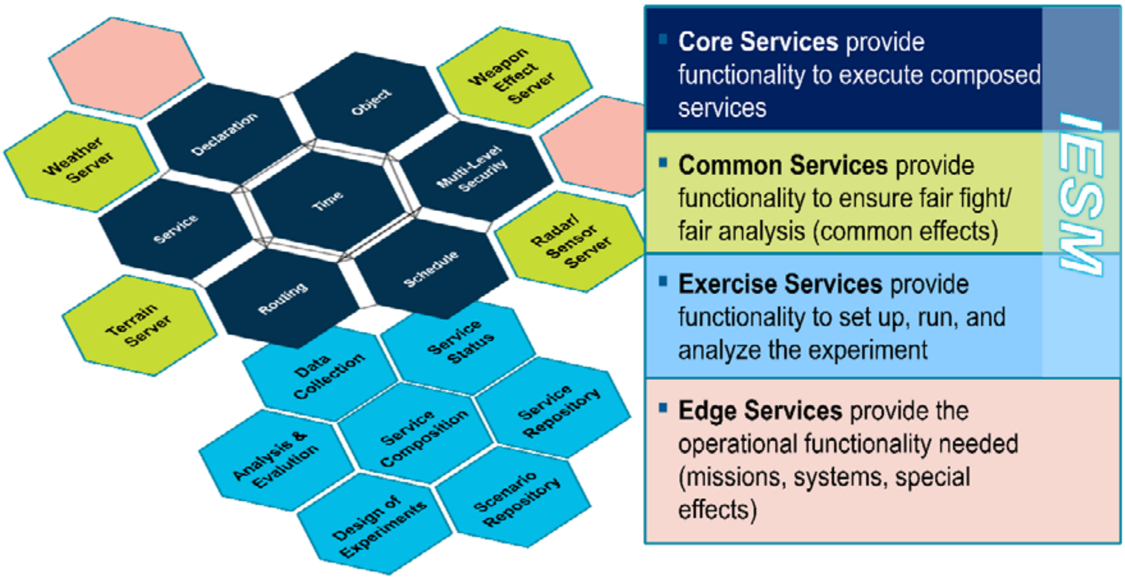
- Depending on the size, scale, or number of simulated entities and participating services, or the amount of data to be exchanged, the number of services providing the needed functionality can vary. Composable services allow engineers to add or remove services as needed.
- Different service instantiations can provide similar functionality to services that have different resolution/scope, time management needs, and supported simulation paradigms.
- Should new or unforeseen conditions occur, only those services affected by those conditions need to be updated, allowing rapid and efficient adaptation of the solution to the new environment, tasks, missions, or threats.
- Services are modular by design, allowing them to be separated and recombined by engineers, based on the principles of loose coupling in the technical domain and composability in the conceptual domain.
- All properties captured so far are leading to a fully configurable solution that provides all needed functionality in an efficient way.

In the following subsections, we introduce the categories of services belonging to the IESM. These are core services providing the information broker functionality, common services that enable fair fight within the composition of services, and exercise services that provide help in setting up, running, and evaluating the simulation event. The existing simulation systems are interpreted as services as well. As they are interconnected via the IESM, they are referenced as edge services. They provide the functionality needed for training, testing, and analysis by military users.

Figure 3 illustrates these four categories of services. Based on our research, we consider the set of core services to be stable, meaning we do not foresee additional core services to be required. However, if future research reveals the need for additional core services not thus far discovered, these can also be integrated into the IESM. The common services and exercise services depicted are only a subset of possible services of these categories, and additional services can be added as needed. When conducting a training exercise, the support needed will differ from running an operational test, so a different set of exercise services can be configured and composed. Furthermore, analysts may have preferred tools they like to use, so that the integration of such tools as an exercise service should also be supported. Finally, the edge services in the figure are only exemplary placeholders for the wide variety of simulation systems that can provide the requisite functionality for any given use case.

2.2.1 Core Services

The initial design focused on the definition of a set of core services that implements the information broker functionality needed to execute a composition. The influence of the concepts provided by the HLA RTI are clearly visible. However, these original groups were modified and extended to address the challenges more effectively. The following seven service categories were identified as core services for the IESM:



- **Core Services** provide functionality to execute composed services
- **Common Services** provide functionality to ensure fair fight/ fair analysis (common effects)
- **Exercise Services** provide functionality to set up, run, and analyze the experiment
- **Edge Services** provide the operational functionality needed (missions, systems, special effects)

Figure 3: Information Exchange Services Matrix and Supporting Services

- *Service management*: These services provide information on which simulation services are accessible and online (i.e., currently participating in an ongoing exercise). If necessary, the IESM can launch more than one service manager to provide other IESM services with status information on services that could potentially participate, scaling to the number of participating edge services.
- *Declaration management*: These services provide information on which components will produce and which will consume information objects to be exchanged, and which operationally relevant services the IESM provides to the simulated operation. They work closely with service managers and allow them to initialise publish-subscribe, push-pull, and alternative information exchange methods, based on availability and interest.
- *Object management*: In contrast to declaration management services, these services supply information on simulated objects that are currently shared, including which edge service initiated the object and which service is currently in charge of such distributed information objects. These services also provide transient storage for information objects that are used frequently and should be provided immediately to a requesting service if the operation is time critical. They also perform data mediation services. Several object managers may be needed to scale the IESM to the desired number of simulated objects.
- *Time management*: These services synchronise the execution of all participating simulation services, including services that operate faster and slower than real time. The details go beyond the scope of this paper. Instead, we refer to other publications, such as [12].
- *Routing management*: These services enable the transfer of tailored data between participating services via configurable communication channels. The services ensure that only required information is submitted, and only information that is not already available in the transient storage is requested from the producing simulation service. Routing managers can scale and adapt to the number of required routes and constraints for such routes. The concept of multiple parallel but

synchronised routes is new to the simulation community, and has not been addressed in any of the current simulation interoperability standards.

- *Schedule management*: When operating in high-performance environments, it may become necessary to adapt the order in which simulation and IESM services are executed and to produce required information objects to ensure real-time availability. Schedule managers address this challenge. In addition to knowing the technical state of all participation simulation services, these services may also require mission awareness. The topic of variable scheduling has also not been addressed in current simulation interoperability standards.
- *Multilevel security management*: When simulation services and audiences with different security classification/clearance levels participate in the same simulation experiments, the accessibility and visibility of data and services depend on user role and clearance/classification level of the user and/or service. By obfuscating and filtering/masking information objects and service performance accordingly, these services provide cyber security against illegal access or modification of the configurations and data.

Conceptually, all these services collaborate with each other to deliver precisely the required information to each endpoint, without any extraneous traffic, in a secure and timely manner. In addition, if a simulation or experiment does not require certain functionality, the service providing it will not be loaded. The new infrastructure will only provide the required functionality at the specific point of need, such as geographically distributed simulation centres all supporting a common exercise. Higher functionality, such as data mediation facilitating the information exchange between participating simulation systems (that may have different views on the information being exchanged) can be provided as an interplay of these services. The technical solution applied for our proof of feasibility is documented in section 3.1 of this paper.

2.2.2 Common Services

The second category of services belonging to the IESM is common services, which are based on a concept well-known to the NATO M&S Group. Common services provide functionality required by several participating simulation systems, such as common weather, terrain, or weapon effects, for example. A recently presented NATO example for providing such common services is presented in [13].

The reason for common services is to ensure fair fights between simulated entities in different participating simulation systems, by ensuring the consistent computational representation of effects [14]. Instead of using the various implementations as provided by the original simulation systems, a common implementation is used instead. Otherwise, the result of a fight may differ, depending on which simulation system computes the effect. If such calculations result in systemic preferential treatment for any one side, the fight between the entities would be considered unfair and not suitable for training, and even less so for analysis. Common services mitigate this imbalance by replacing some functionality provided by the individual simulation services with a common conceptualisation and implementation of that functionality. Therefore, the use of common services will likely require potentially significant code changes in the participating simulation systems, especially for legacy systems with a monolithic code structure. Modular systems may be easier to adapt.

In the context of NATO, it should also be pointed out that common services should not cover up national doctrinal differences and variations. If a mission’s outcome depends upon the applied techniques, tactics, and procedures, these differences must be considered while planning training, analysis, and testing events. The intent is to eliminate systemic unfair fight combinations resulting from artificialities within the composition, not to de-emphasise any real-world variations.

2.2.3 Exercise Services

The third category are the exercise services. These are 'custodian' services needed to set up, run, and evaluate the simulation event. Although not needed for the execution itself, this set provides important practicality for military users. The exercise services facilitate and simplify the administration of this new infrastructure by providing intuitive interfaces and abstracting the complex technical details.

Exercise services must support exercise planning, preparation, execution, and assessment. They provide the information required by the exercise control group, as well as by simulation and network engineers. The current simulated scenario (and the various perceptions thereof by exercise or test participants) is as important as the current workload in the computer systems' execution of the simulation and the bandwidth availability of the underlying network. All data of possible use during the execution shall also be accessed and stored for after-action reviews, in an operational sense (i.e. providing better training and tests), as well as in the technical sense (i.e. providing technically stable and efficient solutions).

Therefore, exercise services should provide a framework that allows the integration of all services needed to support the simulation experiment, including the use of operational data where desirable and possible within classification constraints. The military training and test communities already apply comparable solutions providing such functionality [15], but often not in the scalable, flexible, adaptable, modular, and configurable way envisioned for the next generation of the simulation infrastructure.

2.3 Applying the Concept of Composable Services

The concept of composable services is also applied to the core, common, and exercise services of the IESM to maximise reuse of applicable solutions already provided. This includes using operational planning tools to plan and initialise simulation experiments. The planner responsible for a training or testing event should not have to be concerned with the simulation infrastructure, but should be able to design the mission underlying the training or test with the tools they are used to working with.

The concepts of composable services should also allow the reuse of parts of current infrastructure solutions—such as services providing functionality needed for time management—that can be based on RTI implementations, or to reuse particularly successful after-action-review tools. Software engineering provides numerous solutions to make successful solutions web-accessible as services, and several research projects in support of the idea of M&S as a service have been conducted by NATO [9].

Therefore, concepts presented in this paper are more of an evolutionary (rather than a revolutionary) new approach and allow the transition from current implementations towards the new simulation infrastructure needed to support the new systems engineering challenges, such as those captured in [7].

3.0 ADDITIONAL CHALLENGES

An implementation of the concepts of composable services must overcome some additional challenges that we will address in this section. Some are technical, such as the selection of an appropriate method and technology to couple the services; others are conceptual, such as providing a common conceptual understanding of the mission to be supported in the training or testing exercise, allowing for the conceptual alignment of providing services, or the avoidance of epistemological or hermeneutical mistakes in the assessment of the exercise or training event.

The following section provides a selection of challenges for consideration, which is neither complete nor exclusive.

3.1 Service Coupling

As discussed in the earlier sections, one of the driving requirements for the JSE is to utilise distributed, modular, composable architectures that leverage many purpose-built components simultaneously, rather than one monolithic simulation engine that drives the entire system. It has already been noted that such architectures pose significant challenges concerning scalability and integration. As a response to such challenges, MITRE has developed the *LightBind* Framework as an example of how to loosely couple services efficiently. *LightBind* is designed to let developers quickly build applications that need to integrate with many other heterogenous hardware or software components, both local and remote. It serves as an 'Application Programming Interface (API) Aggregation Engine', which wraps the functionality of those components with a series of lightweight 'drivers' and brings them all into a single environment. This allows for real-time metrics collection, event notification, and functionality invocation without any modification of the original components. It includes a platform and language-agnostic set of interfaces and libraries; support is currently included for both Windows and Linux systems, with software development kits (SDKs) for C++, .NET (C#), and Java components.

LightBind utilises three distinct entities in its framework: drivers, a server, and the agent, as visualised by the simple example presented in Figure 4:

- *LightBind Drivers* are the fundamental atoms of the framework. They are wrapper programs that convert the existing API of disparate software and hardware resources into a common standard that can be accessed by other components. They allow the integration of applications or resources via their native API as well as non-executable binaries, such as dynamic link libraries (DLL) or Java archive (JAR) files. The driver interface specification allows developers to expose their components' functionality as regularly measurable metrics, discrete notable events, and remotely invokable functions. Drivers can be simple, stateless translators or deep, complex programs depending on the use case. As they are only tied to each software or hardware component, they are meant to be reusable across organisations for different experiments or simulations.
- The *LightBind Server* is the central control and monitoring application, providing a single endpoint for the administration and analysis of the entire environment, down to the raw infrastructure level. The server is not one common application that is run in every environment; rather, it is purpose-built by each organisation and tailored to their specific needs. Its nature is dependent upon the organisation's use case. Thus, it can be a web application, a desktop application, a mobile application, or even a rich and dynamic web service with a complete authentication, logging, and database backbone. The *LightBind* Software Development Kit (SDK) provides everything developers need to quickly and efficiently implement their own server programs in the form they require.
- The *LightBind Agent* is a background service that is installed on each machine involved in the activity. It serves as the gateway for that system into the *LightBind* network. The agent manages the locally-installed drivers and exposes them to the server. The server can connect to many agents across many machines, explore what drivers are installed, and use the agents to instantiate any drivers required by the users during experimentation and operations.

The *LightBind* approach can be used to couple services that do not need any time management, and that agree on the meaning and use of information exchanged between the systems. As such, it can be used by itself to couple simulation systems without any additional infrastructure, if the real-time paradigm is followed and a standardised information exchange model is used (such as the protocol data units (PDU) defined in IEEE 1278, or an agreed federation object model (FOM) defined in IEEE 1516).

In practice, engineers within the MITRE team were able to use this approach to successfully couple DIS capable real-time simulations and simulators in only a few days.

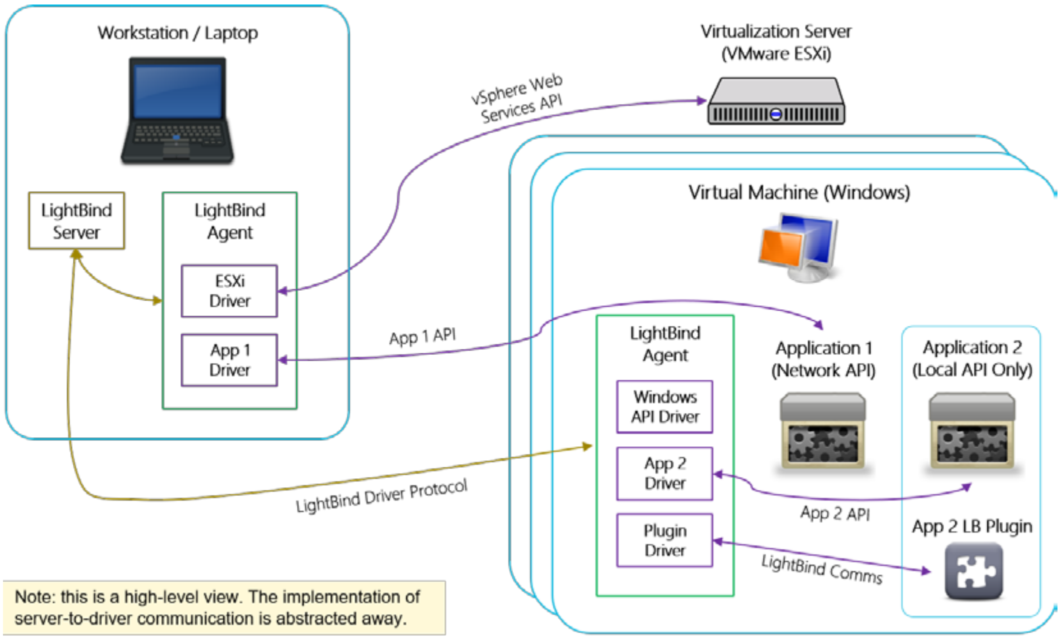


Figure 4: ***LightBind*** **Framework Components**

Its true potential is revealed when *LightBind* is used as a meta-framework, in which each IESM service and the edge services are connected via their respective drivers. As these services are loosely coupled, exchanging services is easy, as the only requirement is the registration and deregistration of the drivers from the agent. In addition, the approach does not assume a given number of services. If many simulated objects need to be exchanged, the amount of IESM service required to support the object management can be easily scaled. If the composition is distributed locally, IESM services can be provided locally and synchronised with each other, thus addressing (for example) the time challenges identified in [16].

The *LightBind* approach is not mandatory but is presented as an example of a promising implementation approach. The concepts presented in this paper can be implemented by various approaches, as long as they provide the required exchange of information between the services. *LightBind* is an example of new technology applied in the current prototypes and provided the desired technical maturity and flexibility.

3.2 Conceptual Alignment

The principles and concepts discussed thus far have focused on developing and providing a simulation infrastructure mainly concerned with *interoperability*, which we define as *the ability to exchange data and make use of these data* in the receiving system or service. As discussed in [17], interoperability is necessary but not sufficient to support fair fights for training and analyses. This requires *composability*, which we define as *the consistent representation of truth* in all participating systems, which can be enabled by conceptual alignment of services that are used to simulate the diverse aspects of a mission. The use of common services, as defined in section 2.2.2 of this paper, is the ultimate solution. However, as this may require significant changes for edge services, the following solution can be applied as an interim solution. The central idea of conceptual alignment is to use the military planning process for a mission needed to provide the contexts for the training or test event as the reference model for what the different edge services and applicable common service must provide.

- In the conceptual planning phase, the military planners define the mission, including metrics for success. Planning can be conducted with operational mission planning tools as well as with alternative planning tools, such as discussed in [15]. The mission can also be captured in operational standards, such as System Modelling Language (SysML) constructs, as used in the NATO Architecture Framework (NAF) or the DoD Architecture Framework (DoDAF).

- In the composition phase, the edge services and common services that can provide needed simulation capabilities to represent systems and activities required to conduct the mission (including opposing forces) are identified. Applying user-defined optimisation rules, the services that are the best composition to simulate the full mission are then selected. The mission information is enriched by data about the providing service, package information including interfaces and other details for the providing services, and more. The mission data enriched by the technical information becomes the blueprint for the next phase. As the blueprint is derived from the mission, the operational requirements for training and testing are fulfilled.
- In the operations phase, services are composed to implement the blueprint. IESM and edge services are adapted and scaled to fulfil the technical requirements for training and testing. By orchestrating the composed services, the event is conducted, and the results captured and evaluated. The assessment provides feedback to improve operational ideas as well as technical modifications for future events.

Figure 5 presents this Concept of Operations to ensure conceptual alignment of services by ensuring that the mapping from mission to blueprint to implementation by a service is unambiguous and well documented.

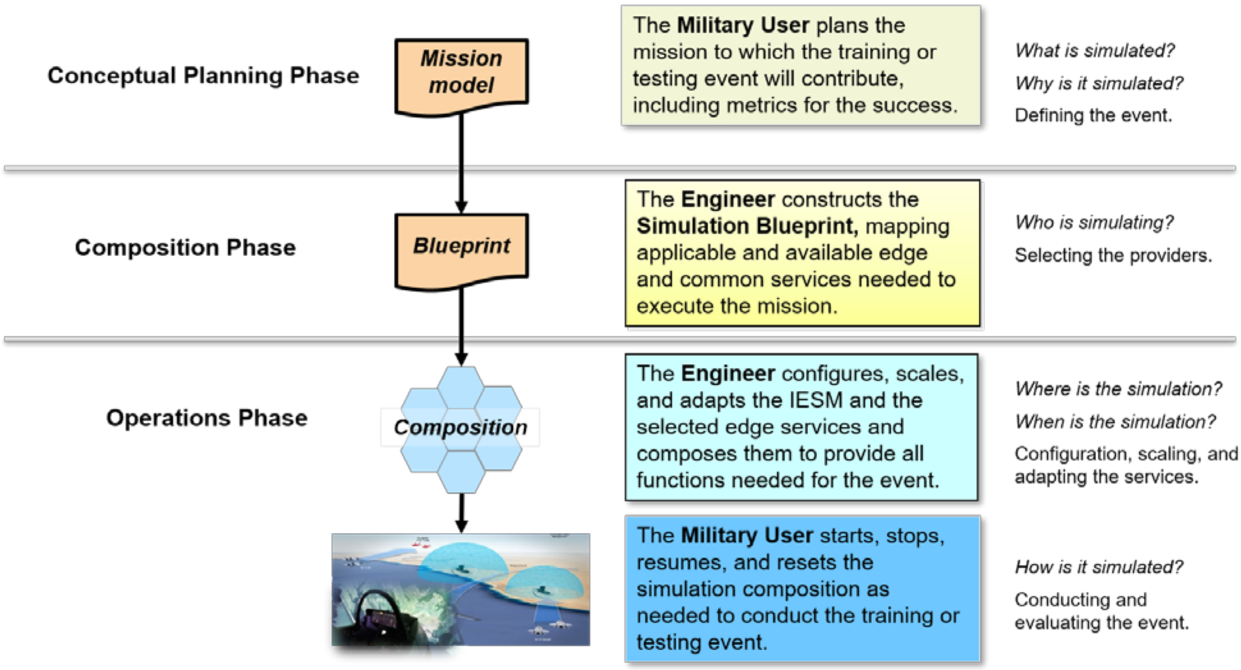


Figure 5: Concept of Operations

In practice, services supporting these different phases can be composed to modular and configurable supporting solutions, such as scenario generation tools, including operational solutions already used in the operational context, in the conceptual planning phase, or network configuration tools and workload monitoring tools used for the operations phase.

3.3 Epistemological and Hermeneutical Challenges

When assessing simulation results, two general risks must be avoided. These can be encapsulated in two key questions, namely (1) ‘Do we simulate everything important for the training or test event?’ and (2) ‘Do we base our assessment exclusively on unbiased, simulation-based data?’

The *epistemological* challenge centres on the limits of an experiment that stem from the conceptual model of the mission: Does the mission model capture all relevant concepts, properties, relations, and processes? Does the mission model capture them at the correct abstraction level? Because the conceptual mission model describes the ‘reality’ to be captured in the simulation (and used as the blueprint for conceptual alignment, as shown in the previous subsection), only concepts included in the model can be evaluated. If important aspects are overlooked, the simulation systems cannot take them into consideration.

The *hermeneutical* challenge centres on the interpretation of the results themselves. When interpreting the results of a simulation experiment, engineers must be guided by the conceptual mission model underlying the experiment. It is human nature to interpret results in the light of one’s favoured worldview, often resulting in unconscious bias. Interpretations should only use assumptions and constraints that are captured within the conceptual model, because all other explanations extend beyond the simulation experiment and consider concepts outside the model’s scope.

Epistemological and hermeneutical challenges further emphasise the role of the mission/blueprint artefact used for conceptual alignment, not only for the technical implementation, but also for the assessment and documentation of results. Having a clear and unambiguous representation of the conceptual ideas of the mission that supports the testing and training event is necessary for success. If this is achieved using formal methods that clearly capture all desired entities, properties, relations, actions, and effects, it is then sufficient for the agreement at the conceptual level.

4.0 DISCUSSION AND SUMMARY

Systems engineering developments in IoT and CPS domains result in systems that utilise significantly more data than current approaches, and expose multi-modality. Current simulation interoperability standards (such as IEEE 1278 and IEEE 1516) potentially fall short in supporting the resulting information exchange requirements for simulation systems representing such new systems. The next generation weapon systems of NATO are not an exception.

As IEEE 1516 does not standardise the implementation of an RTI, but rather its interfaces and the required functionality, there have been various efforts to web-enable the RTI as described in [18]. However, as the service groups were too tightly coupled, separating and individually configuring them while remaining compliant with the standard proved to be challenging. While time management [12] was the state-of-the-art, the data-rich and multimodal systems presented unforeseen challenges that ultimately resulted in the providing technology being perceived as insufficient for these new systems engineering environments by practitioners. Thus, instead of using standardised solutions to foster reuse and collaboration, engineers focused on highly efficient point solutions driven by the requirements of individual programs, which carries the risk of resulting in the same fragmentation and isolated-island solutions the community has sought to avoid with the introduction of standards.

As a future option, this paper proposes a scalable, flexible, adaptable, modular, and configurable simulation infrastructure based on common, core, and exercise services, comprised in the IESM as a central piece of a possible JSE. The services can be loosely coupled to maximise flexibility and scalability, as demonstrated using the *LightBind* framework. While this provides a solid technical solution representing the state-of-the-art, the unambiguous and formal representation of the simulation purpose is necessary and sufficient to ensure the challenging conceptual alignment of independently developed solutions. Representing the simulation purpose (such in military scenarios) in form of a formal and machine-readable description of the mission makes all design decisions traceable and avoids misinterpretations in the evaluation and assessment process. It also conforms with the IEEE 1730 Recommended Practice for Distributed Simulation Engineering and Execution Process (DSEEP).

This proposal is not the end of the road. The research is driven by the dichotomy between the need for highly efficient support of exercises and the support of reuse through standardised approaches. Highly efficient solutions do not include any unnecessary ‘dead weight’, (for example, mediating data into or out of the information exchange format, providing multi-stage translation from standardised interfaces to the really needed interfaces, or providing information required by the standard not really needed for the actual connection). As a result, they are often so specialised that they do not qualify for reuse in any other context, and they require qualified experts to maintain them. Standardised solutions are always a compromise between efficiency and reusability.

The concepts presented in this paper relate to the use of composable, configurable, adaptive services for the IESM as well as the JSE itself. This provides a modular and easily scalable solution, as services can be loaded as often and where they are needed—and only loaded if needed—reducing ‘dead weight’ significantly. However, as model-based systems engineering (MBSE) methods increase, it will become more likely that future standards will be applied at the modelling level, and MBSE methods will derive highly efficient solutions from this meta-alignment of underlying concepts. Although research has already been conducted in this domain, the resulting technologies are not yet mature enough to be applied on the scale required to support JSE or related efforts. Nonetheless, the community should continue to observe and apply these ideas in experimental set-ups, as the auto generated production of highly efficient solutions (based on common standardised meta-models for the aligning of data, harmonising of processes, and orchestration of the composition) offers a promising future solution.

Although the examples here are from the defence simulation domain, the proposed new simulation infrastructure is generally applicable across all domains for simulation systems. Using loosely coupled services (composed on the basis of unambiguous and formal representation of the simulation purpose) combines the openness of IEEE 1516 with the semantic and pragmatic rigour of solutions that prescribe the use of a common object model as the centrepiece of their agreements (such as the object bus used in TENA, or the use of clearly standardised IEEE 1278 PDUs). At the same time, valuable lessons learned can be applied, and solutions that can be transformed to services (for example by using the *LightBind* approach) can be reused. As such, the proposed solution should be of interest for future collaborations within NATO as well.

Acknowledgements and Remarks

The research leading to this paper has been supported by the US Air Force. We would like to thank Tim Menke for his contributions and discussions. The MITRE team was essential in conducting the research, and we would like to thank Jen Hebert, Mike Lewis, David Prochnow, Patrick Getz, Lesley Piper, Tarana Chowdhury, Katie Stinebruner, Larry Sun, and Bob Connell.

This paper has been approved for public release, distribution unlimited, by The MITRE Corporation Case No. 19-1089. The views, opinions, and/or findings contained in this paper are those of The MITRE Corporation and should not be construed as an official government position, policy, or decision, unless designated by other documentation.

REFERENCES

[1] Behner H, and Löfstrand B. “The New HLA Certification Process in NATO,” Proceedings of NATO MSG Symposium, *NATO Report STO-MP-MSG-149*, Paper 19, Lisbon, Portugal, 2017

[2] Calvin J, Dickens A, Gaines B, Metzger P, Miller D, and Owen D. “The SIMNET Virtual World

Architecture." *Proceedings of the Virtual Reality Annual International Symposium*. IEEE, Hoboken, New Jersey, pp 450–455, 1993

[3] Kuhl F, Weatherly R, and Dahmann J. *Creating Computer Simulation Systems: An Introduction to the High Level Architecture.* Prentice Hall, Upper Saddle River, New Jersey, 1999

[4] Mittal S, Durak U, and Ören T. *Guide to Simulation-Based Disciplines: Advancing Our Computational Future.* Springer International Publishing, Cham, Switzerland, 2017

[5] Khan R., Khan SU, Zaheer R., and Khan S. "Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges." *Proceedings of Frontiers of Information Technology*. IEEE, Hoboken, New Jersey, pp 257-260, 2012

[6] Baheti R., and Gill H. "Cyber-Physical Systems." *Journal on the Impact of Control Technology* 12(1):161-166, 2011

[7] Hill RR, Tolk A, Hodson DD, and Millar JR. "Open Challenges in Building Combat Simulation Systems to Support Test, Analysis and Training." *Proceedings of the Winter Simulation Conference*. IEEE, Piscataway, New Jersey, pp. 3730-3741, 2018

[8] Mittal S, Diallo SY, and Tolk A. *Emergent Behavior in Complex Systems Engineering: A Modeling and Simulation Approach*. John Wiley and Sons, Hoboken, New Jersey, 2018

[9] Cayirci E. "Modeling and Simulation as a Cloud Service: A Survey." *Proceedings of the Winter Simulation Conference*. IEEE, Piscataway, New Jersey, pp. 389–400, 2013

[10] Tolk, A. (Ed.). *Engineering Principles of Combat Modeling and Distributed Simulation.* John Wiley and Sons, Hoboken, New Jersey, 2012

[11] Hannay JE, and van der Berg T. "The NATO MSG-136 Reference Architecture for M&S as a Service." Proceedings of NATO MSG Symposium, *NATO Report STO-MP-MSG-149*, Paper 3, Lisbon, Portugal, 2017

[12] Fujimoto RM. "Time Management in the High-Level Architecture." *Simulation* 71(6): 388-400, 1998

[13] Stüber R. "SEDRIS on the Test Bench – The Future of Exchanging Environmental Data to become Part of M&S as a Service." Proceedings of NATO MSG Symposium, *NATO Report STO-MP-MSG-149*, Paper 16, Lisbon, Portugal, 2017

[14] Siegfried R, Lüthi J, Herrman G, and Hahn M. "How to ensure Fair Fight in LVC Simulations: Architectural and Procedural Approaches." Proceedings of NATO MSG Symposium, *NATO Report RTO-MP-MSG-087*, Paper 16, Bern, Switzerland, 2011

[15] Boukhtouta A, Bedrouni A, Berger J, Bouak F, and Guitouni A. "A survey of military planning systems." *Proceedings of the 9th International Command and Control Research and Technology Symposium*, Command and Control Research Program Press, Washington, DC, 2004

[16] Millar JR, Hodson DD, and Seymour R. "Deriving LVC State Synchronization Parameters from Interaction Requirements." *Proceedings of the IEEE/ACM 20th International Symposium on Distributed Simulation and Real Time Applications*, IEEE, Piscataway, New Jersey pp. 85–91, 2016

[17] Tolk A. "Interoperability and Composability: A Journey through Mathematics, Computer Science, and Epistemology." Proceedings of NATO MSG Symposium, NATO Report STO-MP-MSG-149, Invited Paper, Lisbon, Portugal, 2017

[18] Morse KL, Drake DL, and Brunton RP. "Web enabling HLA compliant simulations to support network centric applications." *Technical Report,* Science Applications International Corporation, San Diego, California, 2004