

HLA AND BEYOND: INTEROPERABILITY CHALLENGES

Judith DAHMANN, Marnie SALISBURY, Phil BARRY
Chris TURRELL and Paul BLEMBERG

1. Introduction

The High Level Architecture (HLA) is a runtime software architecture for distributed simulation, which can be used by developers to create simulation applications. The HLA was developed in an environment that required the maximum amount of flexibility along with a minimum number of constraints. Simulations run the gamut from human-in-the-loop individual training simulators to highly aggregated, discrete event simulations that are used to exercise entire battle staffs. Beyond the training environment, the research and acquisition communities each have their own requirements for simulation applications. To address these diverse requirements, the HLA is designed to be flexible enough to support this diverse set of simulation systems in their unique environments and sophisticated enough to embrace the next generation of simulations. This broad-based environment sets the stage for the development of the HLA specification following the initial baseline definition of the HLA in March 1995.

In essence, HLA provides a framework and a set of capabilities to support the design and execution of simulation applications (federations) composed of multiple simulations and interfaces with live systems (federates). It is the task of the federation developer to apply these capabilities to meet the needs of their application. A key to this is ensuring that the selected federates *interoperate* in a way that technically and substantively meets the needs of the application. Consequently, for the purposes of this discussion, we consider interoperability from two perspectives: *technical interoperability* and issues of coherency and “fair fight” within the simulation domain or *substantive interoperability*.

As a technical architecture, the HLA addresses technical interoperability issues and provides users with support in understanding how to apply the HLA technical

capabilities to create a working federation. Beyond the achievement of technical interoperability, there is a requirement to anticipate and address issues that impact on the ability of simulations to inter-operate in a logically meaningful manner. This paper identifies and discusses issues that affect interoperability in these two areas using the Federation Development and Execution Process (FEDEP) as a reference and guide.¹ The key interoperability issues are identified in a general way and are illustrated in the context of several battlespace applications. In the process, some activities that support HLA users to address interoperability challenges are outlined.

2. The Federation Development and Execution Process (FEDEP)

The Federation Development and Execution Process grew from the desires of the early HLA proto-federation development teams to organize and document the process for the creation and execution of an HLA federation. The FEDEP is a systems engineering process model that defines a generic framework for distributed simulation development. It is not unique to the HLA, but rather has been adapted to provide a common reference point for communication between federation development team members. The most common representation of the FEDEP consists of a graphical model, however, a supporting documented description is also available.

Recently the five step version of the FEDEP was expanded into six steps by segmenting the former Step 3, Design and Develop Federation, into two separate parts. This was undertaken in recognition of the complexities of these tasks, and the number of underlying events associated with each of them. In addition to providing a clear and concise introduction to federation development for HLA users, the FEDEP also offers a common foundation for defining functional overlays associated with the federation development process, such as, verification, validation and accreditation (VV&A), Security and Tools. The six steps in the process are:

- Step 1: Define Federation Objectives
- Step 2: Develop Federation Conceptual Model
- Step 3: Design Federation
- Step 4: Develop Federation
- Step 5: Integrate and Test Federation
- Step 6: Execute Federation and Prepare Results

Most recently, the FEDEP has been augmented with the addition of a checklist.² The idea behind the development of the checklists was to document many of the details that must be considered in the execution of the FEDEP and to group them logically within the activities to which they pertain. The checklists were developed based on the experiences gained through the development of several specific federation

applications. In general the items on the checklists are applicable to all federation efforts, but of course, they are tailorable as is the FEDEP itself.

The concept behind the FEDEP then is to provide a structured methodology to address the issues associated with the development and execution of an HLA federation. As such, it provides a useful framework for the discussion of simulation interoperability issues.

3. Technical Interoperability

Often when one discusses interoperability, the focus is on technical interoperability. Technical interoperability is achieved when the federation is integrated, and individual federates are passing data in accordance with the federation object model (FOM). This involves the use of common standards, compatible interfaces and coordinated data structures.

Technical interoperability is the capability of federates to physically connect and exchange data in accordance with the FOM. Elements include the following:

- Standards compatibility
- Hardware compatibility
- Time management coordination
- Coordinated use RTI Services
- Security issues (if applicable)

Appropriate use of *standards* is a key element of technical interoperability. Complying with the HLA standard, as an individual federate or as a federation,³ provides assurance that the specifications of the architecture have been correctly implemented, and hence basic capabilities for operation are met. It is a minimum essential capability for developing a federation. Other standards that contribute to technical interoperability are those for operating systems, languages, and network protocols.

Hardware compatibility is another basic essential technical component to federation development and execution. Physical interoperability, or the actual connection of one system to another, is one level of hardware interoperability while systems capabilities and capacities are another. For example, differences in the capabilities of network cards may cause performance problems within a federation that fall under the heading of interoperability. Consider the case of a 10BaseT network card in a 100BaseT LAN environment. These issues generally come to light during Step 5 of the FEDEP. The Federation Execution Planner's Workbook⁴ is one of the mechanisms available to assist federation managers in identifying and resolving these issues before they

become problems. Among other information, the workbook provides a place to capture information about the computing environment in which the federation is executed, and therefore provides insight into potential incompatibilities. Another hardware interoperability problem is the issue of data formats and byte-ordering, and the issue of big endian versus little endian. These issues are well documented and will not be discussed here, but need to be addressed in the federation development process during Step 4 of the FEDEP, Develop Federation.

The use of *time management* is a key element of technical interoperability in a federation. The mechanics of interoperability between time regulated and unregulated simulations must be considered and resolved. The HLA was designed from the outset with the capability to facilitate interactions among dissimilar simulations. The Time Management Services provided by the Runtime Infrastructure provide the means of solving the problem, but the correct use of these services is left to the federation development team. Failure of the federates to use the services correctly can result in a federation that stalls or never runs forward in time.

There are several *RTI Services* that are intended to support and apply to the entire federation, not just individual federates. Examples of these are the synchronization point services, save and restore services, and data distribution services. Uncoordinated or inaccurate use of these services within a federation can cause catastrophic failure of the federation execution. Use of these services must be addressed by the federation development team during FEDEP steps 3 and 4, Design Federation and Develop Federation. For example, will synchronization points be employed; if so where and how? If synchronization points are used, then some or all federates must be able to respond to them. If not, is a surrogate, such as the Federation Management Tool, available to respond on their behalf?

Security is the last issue to be addressed under the heading of technical interoperability. Data encoding and decoding, levels of security and methods for bridging secure and unsecured networks must all be addressed. Security considerations can be identified as early as Step 1 of the FEDEP and are addressed throughout the execution of the process. A security overlay to the FEDEP has been developed and is available to assist federation management teams with these issues.⁵

4. Substantive Interoperability

Resolving technical interoperability issues insures that the federation will run, but says nothing about the adequacy of the federation to accomplish its mission. Unlike technical interoperability where something works, or it does not, substantive interoperability is driven by the needs of the federation and has to be addressed by each federation in a federation specific way. Thus these issues are much less obvious,

and therefore more difficult to solve in a general way. The level of interoperability required depends in part on the desired outcome of the federation, the capabilities of the individual federates involved, and the degree to which consistency must be maintained to achieve the desired results. These considerations are different for almost every implementation, and therefore, the standardized methodology offered by the FEDEP becomes even more important as a means of focusing on the issues. The FEDEP provides a structured methodology for the development of the federation and along with the Checklists identifies the appropriate places in the process where these issues should be considered. The remainder of the paper will be devoted to the identification and discussion of coherency and “fair fight” issues that transcend technical interoperability.

4.1. Interoperability among Entities

When you form a federation, you are technically linking the federates, but you are logically linking the entities represented inside of each of the federates. To have a meaningful federation, the set of entities represented across the federation work together in a manner consistent with the needs of the federation application. In effect, in an HLA federation, you are creating an end-to-end model by reusing selected representations from the participating federates. The data in the FOM defines the exchanges among the federates, but the entity representations and their combined behaviors define the new simulation application supported by the federation.

The purpose of the federation is the key definer of this process. Defining the representational needs of a federation is the starting point for creating one that meets the needs of the user/sponsor. A good conceptual analysis during Step 2 of the FEDEP can not be overemphasized. During the Conceptual Modeling phase of the FEDEP the federation development team and sponsor develop the story line describing the sequence of activities, that, when executed in the federation, will achieve the federation objectives. This usually involves the creation of a scenario and a conceptual analysis. The scenario defines the activities and relationships that will exist among the entities, while the conceptual analysis captures a “real world” authoritative description of the entities and actions that must be represented in the federation. This analysis will influence the ultimate selection of federates as well as the development of the Federation Object Model (FOM). Selecting federates with the right representations for the purpose of the federation is key to substantive interoperability

While the specific areas which need to be addressed in any application will be defined by the needs of the federation, the type of issues to be considered include:

- *Entity level of representation:* Does the level of representation of entities included in the federation support the federation conceptual model, and as a group logically ‘fit’ together?

All entities do not necessarily need to be represented at the same level, but interactions between entities (e.g. individual weapon systems fired against composite targets) need to be appropriate for the needs of the federation.

- *Entity attribution:* Do the entities in the federation incorporate representation of the key attributes salient to the federation purposes in a sufficiently consistent manner to meet the federation needs?

Issues of completeness and consistency of representation across the federation are important primarily in the selection of federates. In Step 3 of the FEDEP, the federation manager selects federates and assigns responsibility to represent attributes of classes of objects to them. An important part of this process is the assessment of the ability of the participating federates to represent the critical characteristics of entities needed to meet the federation mission.

- *Entity behaviors:* Do the behaviors of the individual entities reflect the needs of the federation conceptual model and do they ‘work’ together logically to meet the needs of the federation? This includes the algorithms used in the simulations to compute the effects of one entity on another.

For example, for simulations to be considered interoperable in supporting an application involving battlespace outcomes, the effects of weapons must be consistent and realistic from both sides of a weapons engagement. Consider two simulations, A and B. Let simulation A represent ground forces in a federation and simulation B represents the air forces. Suppose that aircraft modeled in simulation B attack a target in simulation A with air to ground missiles. Suppose further that simulation A returns fire with surface to air missiles. Which simulation computes the attrition in each case? Is the calculation mathematically correct and from whose perspective? A key to interoperability is to determine if the weapons effects of each of the simulations on the objects owned by the other are realistic.

It is important to note that this assessment must include both the behaviors of entities that will be exposed to the federation (HLA objects) but also the behaviors of relevant entities that are wholly modeled inside an individual federate. For instance, consider the case of a federation designed to assess comparative capabilities of different sensors in a particular scenario that includes weather effects. In this case, it is important to ensure that the effects of weather are represented in the different federates, and these effects are consistently modeled across the set of sensors whose capabilities are under assessment.

- *Temporal resolution:* Each simulation computes changes in state of the entities it represents at some resolution of simulation time. While all entities in a federation application do not have to resolve time at the same rate, it is important that the overall end-to-end model has incremental state changes computed in a fashion commensurate with the needs of the application.

As an example, consider two simulations. The first is focused on ground combat, the second on air warfare. Suppose the aircraft in the aviation simulation attacks a ground target that defends itself with surface-to-air missiles (SAMs). Not only must the three dimensional locations of the aircraft, the SAM and the SAM site be consistent, but the time steps for the individual simulations also need to be representative of the activities being simulated. In the example, consider the situation where the attacking aircraft is flying at 480 knots which translates to eight miles per minute. If the simulation modeling the SAM systems searches for air targets at one-minute time steps, the attacking aircraft could directly overfly the SAM and possibly destroy it in between timesteps. In other words, at the first time step, the aircraft would be out of detection range of the SAM. At the second time step, the aircraft will have attacked the SAM and destroyed it without the SAM ever having the opportunity to acquire the aircraft. This would not accurately model the real world at a behavioral level.

- *Spatial resolution:* As with time, entity behaviors are represented with spatial ‘granularity’. Here too, the spatial resolution of entity behavior across a federation does not necessarily need to be common but needs to logically fit together in a way that supports the federation application.

The importance of spatial resolution can be illustrated with examples analogous to that cited for temporal resolution above. Consider the example of a federation designed to examine comparative effects of targeting technologies or technologies that improve the abilities of sensors to identify targets. In this federation, the spatial resolution of entities will be key to the results of the federation. If the representation of the terrain is too coarse, then entities may occupy locations at fixed intervals from each other and may appear in regular patterns. This may make target detection and identification unrealistically easy for the targeting technology. If the terrain representation used for the entities is more detailed than that used by the simulation of the targeting technology, then the movement of entities may be indiscernible and may prevent valid target detection or identification.

Considerations of this type are key in the selection of federates to meet the needs of a federation. “Choosing your partners carefully” is the most important part of designing a meaningful federation. Good federate selection depends on a solid federation conceptual model (FEDEP Step 2) that identifies the appropriate levels of resolution of key entities, the salient attributes and behaviors, the particular aspects of entities

and behaviors relevant, and the granularity of temporal and spatial resolution appropriate to accomplishing the federation purpose. Given a good statement of needs of the federation, the federation designer can evaluate the capabilities of candidate federates.

The success of the federation designer is also dependent on the ability to understand the capability of individual federates. Unless information on candidate federates is available, it will be difficult to assess their appropriateness to meet specific federation needs. The SOM for a federate is of limited use in this exercise, since by design it addresses only the data available for exchange in a federation. Here the interest is in the internal representations in a federate whether or not they are exposed at runtime. Given the lack of documentation on simulations and the ad hoc nature of this documentation, today this task typically involves person to person interaction between a federation manager and a person knowledgeable about the federate.

Finally, the federation manager needs to consider the composite suite of federation members. Each federate may meet the needs of a piece of the federation problem. However, the real question is whether the federates as a group work together to meet the needs of the federation as a whole. That is, whether the end-to-end model represented in the federation is appropriate to the federation needs.

4.2. Contextual Effects on Interoperability

In any simulation, the represented entities are situated in a context that affects their behavior as much as the interaction with other entities. When bringing together entities represented in different federates it is very important to ensure that the contexts of these entities are consistent in the areas of importance to the federation. In battlespace simulations, the most critical contextual effects are often environmental (natural and manmade). Depending on the nature of the simulation, there can be other effects whose impact is felt across the federation such as communications and the electromagnetic atmosphere.

In addition to the contextual factors themselves, the *effects* of these factors on entities in the battlespace scenarios must be sufficiently consistent to meet the needs of the federation. Common environmental representation is an important starting point, but beyond this, the effects of terrain, for instance, on line-of-sight calculations of simulated platforms represented in different federates can be crucial in creating a coherent federation.

In this section we will focus on the environment because of its overriding importance in most battlespace applications.

As discussed in ⁶, for multiple simulations to interoperate in the same domain they must have a sufficiently common, correlated view of the *environment* to support the needs of the application. The development of authoritative environmental representations is time consuming and costly, describing all of the relevant elements and environmental events and interactions that are expected to occur in an area of interest. Additionally, environmental representation involves description of the geometric and topological relationships between data objects. To create and sustain realism and accuracy, all federates should have a consistent perception of the environment.

Environmental representation consists of both the visual and non-visual aspects of the environment, and is composed of digital representations of the atmosphere, space, ocean, and terrain. If germane to the simulations, other aspects of the electromagnetic spectrum may also be represented such as infrared radiation and radar reflectivity. Additionally, objects on the battlefield such as trees, buildings, bridges, and highways, etc. may also need to be represented.

Previously, consistency in environmental representation was achieved by developing a mapping between the individual environmental representations of simulations that were required to interoperate. This mapping rectified disparate coordinate systems as well as resolving differences in fidelity and interpretation between the different environmental representations, which is critical to ensure accuracy of weapons and intelligence effects. In general, this type of point to point mapping was based upon proprietary database formats that required the development of customized software to convert the source data into a format that is useable by the simulation. The greater the number of different database formats involved the more, the greater the cost to achieve consistency, if even possible.

The task of ensuring a consistent environmental representation can be decomposed into several logical pieces. First, the environmental data must be drawn from a common authoritative source. Identifying a common authoritative source and generating a consistent environmental scenario eliminates a significant potential variability at the beginning of the process.

The next step is to establish access to the integrated data sets. There are numerous source databases. One approach to provide access to these data sets is to write software that will convert from native format(s) into the format of the simulation database generation system. Depending on the type and variability of the sources involved, this could be a costly and time-consuming process. The Synthetic Environment Data Representation and Interchange Specification (SEDRIS) project has developed a common interchange mechanism with supporting transmittal formats to provide access to all environmental representations through a standard interface.

The transmittal format fully supports a Data Representation Model (DRM). A Data Coding Standard (DCS) provides enumeration for the elements within the DRM.

The DRM and the DCS are logical constructs. The SEDRIS Transmittal Format (STF) is the physical construct that defines the order and physical formatting of the data. To use the STF one creates a SEDRIS transmittal by accessing the *Write Application Programmer Interface (API)* definition within the SEDRIS Interface Specification. The *Write API* takes data content from the native format of the data source as expressed in the DRM and creates a transmittal.

Extraction of environmental data from a transmittal is the function of the SEDRIS *Read API*. As defined in the Interface Specification, the *Read API* consists of a number of functions that pull data from a SEDRIS transmittal. To get the data into the native format of a particular simulation, converter software must be written to interface with the *Read API*. The converter software is simulation specific and must be maintained as native data representations evolve. Designing and implementing the converter software is one means for addressing the fidelity issues that must be resolved between the source data and simulation runtime requirements. Converter software can also use utilities in the SEDRIS *Read API* support libraries to perform requisite coordinate conversions and/or datum transformation.

The advantage of the SEDRIS approach is that a common interface to source data is provided along with the tools to use this interface. Each federate now only needs to write converter software to one standard. Content consistency is achieved through the use of a complete and unambiguous interchange mechanism. The cost of getting data into the simulation is reduced since only one set of conversion software is required to access a complete range of environmental and 3-D model data. Use of SEDRIS and its supporting capabilities aid the federation developer in creating common, or at least consistent, environmental representations across a federation as a context for entity representations.

5. An Example: Command, Control and Communications (C3) Interoperability

In this section we review several notional use cases of interoperability issues arising in applications typical of defense users, those which incorporate consideration of command and control. The representation of command, control, and communications within a federation can pose several interoperability challenges. In this section, we describe several challenges and solutions that have been applied in other federations. Although the problem space has been divided into categories in this section, the categories are quite inter-related.

5.1. Command Organizations and C2 Process

The first element to examine is the portrayal of command organizations and the command and control (C2) process within and outside the federation. Depending on the fidelity of the simulation and the intended uses for it, the representation of command organizations and the C2 process can vary widely. The features of the individual simulations must be well understood before the issues related to interoperability within the federation are examined.

The features of a simulation related to command organizations and the C2 process may be characterized by the answers to a few questions:

- Are the physical elements of command units (e.g., command posts) represented?
- Are the pre-planned actions of combat units directed or initiated by logic within the simulation or by intelligent human operators?
- Does the simulation model the cognitive decision-making functions of commanders or command units and to what extent?
- What impact does the collection and processing of situation awareness data have on the cognitive decision-making functions simulated? Are cognitive decisions made using perceived data about the battlefield situation, or ground truth data stored and calculated within the simulation?
- What battlefield activities, such as attrition or communications, affect the pre-planned or reactive actions of combat units?
- What battlefield activities, such as attrition or communications, affect the quality, completeness or timeliness of output of the cognitive decision-making functions?

Based on the answers to these questions for each simulation in the proposed federation, one can start to characterize the situation for the federation. The following list includes a set of examples:

- One or more federates has very detailed models of C2 command posts and C2 processes, while other federates have more simplistic models or none at all.
- One or more federates contain brittle models of cognitive decision-making, where the decision-making capability breaks down easily when confronted with a previously unknown situation, causing combat units to continue blindly and illogically forward or to stall and do nothing.
- One or more federates executes cognitive decision-making processes using ground truth data while other federates use perceived knowledge of the situation.
- One or more federates contain invulnerable models of command posts, where the C2 organizations, equipment, or processes cannot be influenced by battlefield outcomes.

The interoperability problems that arise from these situations can be described as “fair fight” issues. Here are some examples:

Potential Fair Fight Problems

- The actions and decisions of a unit or force whose cognitive decision-making is based on perceived data about the battlefield situation may be less effective or accurate than those of units whose cognitive decision-making is based on perfect knowledge of the battlefield situation.
- The results of specific offensive actions or campaigns against the command organizations and C2 processes of an opposing force will not be effective if the opposing force is modeled with no vulnerabilities.
- The actions and effectiveness of a unit or force with brittle model(s) of cognitive decision-making may be very adversely affected if it gets into a paralyzed state where it can't make a coherent decision and execute protective actions.

5.2. Command and Control Information Exchanges

Information and data are the resources that drive the C2 process. The Federation Object Model and the RTI services make the technical part of information exchange in a federation straightforward, but other challenges remain. The three key cognitive components of a coherent and successful information exchange are as follows:

1. Something meaningful to say
2. Someone to say it to
3. A recipient that can interpret what is said and react to it

Disparities between the representations of command organizations and the C2 process in different federates will definitely create interoperability problems when those simulated command organizations attempt to exchange information. These problems may be “fair fight” issues or technical incompatibilities that may cause failure during federation execution. Some examples are provided below:

Potential Fair Fight Problems:

- A federate with a detailed model of the C2 process may require information from other command organizations that cannot be provided. The absence of this information may result in less effective decisions and poor outcomes for the simulated units in the first federate.
- A federate with a detailed model of the C2 process may send sophisticated orders to command organizations in other federates assuming that the information will be received and acted upon correctly. If the receiving federates lack the

capability to receive and interpret the orders, the actions carried out by those units may be ineffective and detrimental to the simulated units or command organizations in the first federate.

Potential Technical Incompatibilities:

- A federate with a simplified model of the C2 process or no model of the C2 process will lack item #2 “someone to say it to” and item #3 “a recipient that can interpret and react”. Information that is sent from another federate to this federate will result in inappropriate action or no action.
- A federate with a simplified model of the C2 process may lack the ability to generate meaningful information (item #1) and pass that information to other federates.

5.3. Communications Effects

In real life many types of communications techniques, devices, and networks are used to move information around the battlefield. Building and maintaining a reliable communications infrastructure of sufficient bandwidth is a critical part of military operations. The degree to which the means of communication are modeled varies widely from one simulation to the next.

The features of a simulation related to its model of communications may be characterized by the answers to a few questions:

- Is the physical communications equipment represented? Can it be damaged and rendered ineffective by ordnance or electromagnetic means?
- Is the probability of successful information delivery over the communications infrastructure affected by the terrain, the weather, or the electromagnetic environment?
- Is the communications model flow-based or item-based? That is, does the communications model calculate the number of messages that flow from point-to-point in a given time period or does it provide a probability of delivery for each specific information exchange?
- Does the communications model account for resource contention on the communications infrastructure?

A set of federates with different approaches to communications modeling can lead to “fair fight” problems such as the following:

Potential Fair Fight Problems:

- A federate with a detailed model of communications equipment and networks may degrade the completeness and timeliness of information exchanged between simulated units or command organizations within the federate or information arriving from another federate. This degradation may lead to delayed decision-making or lower quality decisions within the federate.
- A federate with a detailed model of communications may also have a more detailed model of the impact of the terrain, the weather, and the combat actions of the opposing force on the communications infrastructure. In other words, the federate may show degraded combat effectiveness due to failures of the communications infrastructure. A federate that has a low fidelity model (or no model) of the communications infrastructure will not be affected by opposing force actions directed against it.

5.4. Possible Solutions

The solutions applied in existing examples fall into four basic categories:

1. Augment the federation of automated simulations with intelligent human operators.
2. Add functionality to the existing federates
3. Add one or more federates to bridge the coherency gaps between the existing federates or to provide a consistent modeling service to the entire federation.
4. Rescope federation expectations or factor out inconsistencies across the federation in interpretation of federation results

Augment with Intelligent Human Operators

One common way to bridge the gaps among inconsistent models of command organizations and the C2 process or to solve information exchange incompatibilities is to insert an intelligent human in the loop. The human receives and interprets the information generated by one federate and then creates a set of information that can be conveyed to one or more other federates. The intelligent human operator may perform any or all of these functions:

- *Translation.* Translate information from the set of terms generated by one federate to another set of terms that can be interpreted correctly by another federate.

- *Orders decomposition.* Interpret the orders generated by a command organization (simulated or live) at one level of the command hierarchy into a set of orders that can be executed by a lower level unit.
- *Aggregation.* Compute or abstract a higher-level view of a given set of data.
- *Degradation.* Intentionally changing the data generated by one federate before passing it on to other federates to introduce an appropriate level of error or insufficiency.
- *Augmentation.* Creating appropriate data that is not otherwise available in the federation, but is required by other federates.

Add Functionality to Federates

An obvious, but significant option is to extend the capabilities of a federate to make it match more closely the capabilities of the other federates. It may be quite difficult to add the necessary functionality to an existing federate. This approach could lead to a major redesign or rewrite of the software of the original federate.

Add Federates

New federates can be added to the federation to provide the capabilities that are missing or to provide a consistent model of a specific battlefield activity or effect. This approach has been used in both ways. One example is the use of artificial intelligence technologies to create models of specific command organizations that can interpret and react to orders and situational awareness data and execute the C2 process to produce “orders” or directives that can be interpreted by simulated units in another federate. The second way is the use of a federation-wide server to provide a specific algorithm or service to the federation to achieve consistent results. For example, a communications effects server has been used to consistently apply terrain or weather effects to information exchanges over communications devices within a federation.

Rescope Federation Purposes or Factor out Federation Inconsistencies

When adjustments such as those described here are not viable, there are two other options that the federation manager might consider. These are typically considered when resources (i.e., funding or time) don’t allow for changes in the federation, but the users still see value in the federation execution.

- Ignore the incompatibilities to the extent that they don’t cause the federation execution to fail, but factor out inconsistencies in interpretation of the federation results. Recognizing that simulations never provide a full representation of all desired capabilities, this approach may be workable in some circumstances.

- Eliminate modeling features of particular federates to get the federation down to a level (but lower fidelity) playing field. This may provide the consistencies needed, but limit other things that might have been of interest to the federation user. This approach requires a rescoping of the federation purposes to achieve some portion of the federation goals. Only, the federation manager can make tradeoffs such as these in conjunction with the federation user.

6. Summary

In summary, this paper has reviewed a range of technical and substantive interoperability issues to be addressed by a federation developer in the use of the HLA.

Technical interoperability challenges that must be addressed in order to configure and run a federation; these are basic to the use of HLA and are a necessary prerequisite to application of HLA to user problems. These include conformance with standards, hardware interoperability, coordination of time management coordinated use of RTI Services, and security.

While important, technical interoperability does not guarantee the creation of a meaningful federation application. Federation designers must address these issues of substantive interoperability in the design of the federation since the nature and degree of interoperability needed is driven by the purpose of the federation application.

The federation conceptual model, which lays out the representational requirements of the federation, is critical in this process. It provides the criteria for selection of federates based on the characteristics of their representations and their ability to meet the needs of the federation application. Assessing characteristics of the entities in each federate including their level of representation, their attributes and behavior, and their temporal and spatial representation, is the heart of this process. This assessment is done both in terms of the extent to which the entities individually address federation needs as well as the extent to which the set of entities fit together to meet the end-to-end needs of the federation application.

Beyond the entity characteristics, the context in which these entities operate, including the environment and other federation-wide effects, need to be assessed from the perspective of the federation needs. It is important to ensure that the environment and other contextual factors are consistently represented across the federation and the interactions between the entities and these contextual factors are sufficiently consistent to meet the needs of the application.

Successfully addressing these substantive interoperability issues is the responsibility of the federation manager and federation designer. Support in this process is

beginning to appear in several forms. The FEDEP and its associated checklist provide a starting point for this process. The Federation Execution Planner's Workbook provides aid in addressing technical interoperability issues. SEDRIS and its supporting capabilities can assist in addressing important environmental interoperability issues. In several cases (weather, for instance) the idea of federation servers is growing in an effort to provide consistent representation of dynamic federation-wide effects at runtime.

The efforts underway to develop processes for verification, validation and accreditation (VV&A) of federations are addressing many of the issues raised in this paper. In essence, building a federation that incorporates representations appropriate to the needs of the federation application is the heart of the VV&A problem. Further, work in the area of fidelity 7 is attempting to more precisely define some of the issues raised with respect to entity characteristics; both those needed by an application and those provided by a simulation.

Finally, as HLA become more widespread there greater attention is paid to how it is being used, rather than how to use it. This shift in emphasis highlights the importance of raising awareness of interoperability issues and developing techniques and solutions for addressing them.

References

1. HLA Federation Development and Execution Process (FEDEP) Model, Version 1.4 (June 1999).
2. HLA Federation Development and Execution Process (FEDEP) Checklists, Version 1.4 (Draft, May 1999).
3. HLA Compliance Checklist, Federate, Version 1.3 (May 1998).
4. Federation Execution Planner's Workbook, Version 1.3r3 (May 1999).
5. Security Overlay for the FEDEP, available at <http://hla.dmsomil/hla/federation/security/>.

6. U.S. Army Simulation, Training and Instrumentation Command. "SEDRIS and the Synthetic Environment Domain – Volume 1 of the SEDRIS Documentation Set" (March 1998).
7. Gross, David et al. "Report from the Fidelity Implementation Study Group." Paper 99S-SIW-167, presented at the Spring Simulation Interoperability Workshop (March 1999).

JUDITH S. DAHMANN, Ph.D. is the Chief Scientist for the U.S. Defense Modeling and Simulation Office. She has led the development of the High Level Architecture. Address: DMSO, Suite 500, 1901 N. Beauregard St., Alexandria, VA 22311-1772, tel.: (703) 998-0660. E-mail: jdahmann@msis.dmsomil.

MARNIE SALISBURY is a Principal Engineer at The MITRE Corporation. She has seven years of experience in battlefield simulation and interoperability, including 15 months as Technical Director for the DARPA STOW program. She currently leads the HLA cadre outreach team. Address: The MITRE Corporation, 1820 Dolley Madison Blvd., McLean, VA 22102, tel.: (703) 883-7064. E-mail: marnie@mitre.org.

PHILIP S. BARRY is the Deputy Division Chief for the Technology Applications Division at DMSO and a Lead Systems Engineer for the MITRE Corporation. Dr. Barry has a Ph.D. in Information Technology and a M.S. in Systems Engineering from George Mason University, and a B.S. in Aerospace Engineering from the University of Virginia. Address: The MITRE Corporation, 1820 Dolley Madison Blvd., McLean, VA 22102, tel.: (703) 883-7826. E-mail: pbarry@msis.dmsomil.

CHRIS TURRELL is a Senior Simulation Analyst with AB Technologies, Inc. of Alexandria, VA where he is the program lead for the Defense Modeling and Simulation Office HLA Tools and Services development efforts. He is actively involved in the DoD Joint Technical Architecture development process and participates in related IEEE and SISO standards development activities. Address: AB Technologies, Inc., 1901 N. Beauregard St., Suite 400, Alexandria, VA 22311-1772, tel.: (703) 824-3457. E-mail: cturrell@msis.dmsomil.

PAUL BLEMBERG is a senior analyst at AB Technologies. He is the liaison between the Modeling and Simulation Information Analysis Center (MSIAC) and the Technology Application Division of the Defense Modeling and Simulation Office (DMSO). He has a BS in Psychology from Northern Michigan University. Address: AB Technologies, Inc., 1901 N. Beauregard St., Suite 400, Alexandria, VA 22311-1772. E-mail: pbleMBER@msosa.dmsomil.