Solving Common Interoperability Challenges with HLA-to-HLA Bridging

Björn Möller, Pitch Technologies, Sweden
Fredrik Antelius, Pitch Technologies, Sweden
Martin Johansson, Pitch Technologies, Sweden
Linus Lindholm, Pitch Technologies, Sweden
Per-Philip Sollin, Pitch Technologies, Sweden

bjorn.moller@pitch.se
fredrik.antelius@pitch.se
martin.johansson@pitch.se
linus.lindholm@pitch.se
per-philip.sollin@pitch.se

Keywords:
HLA, architecture, bridging, scalability, robustness, security

ABSTRACT: *Recently there has been a lot of focus on gateways between different simulation standards, like HLA, DIS and TENA, for example as part of the LVC-AR work and several SISO working groups. Bridging between different HLA federations has received little attention. Still, this is a highly useful technique for solving many interoperability issues. It is currently in use in several large federations.*

We argue that there are two categories of HLA-to-HLA bridging: quick fixes and long-term architectures.

A sample quick fix is when an HLA federation with a large number of objects is partitioned so that one of the partitions includes a subset of the objects. This prevents federates with limited capacity from being overloaded or crash.

Another case is when federates that use a slightly different FOM or use a slightly different data format are joined to a separate federation, that connect to the main federation using an HLA-to-HLA bridge. In some federations different federates may also use different RTI implementations. In these cases the long-term solution is to correctly adapt all of the federates to the same federation agreement and FOM or to use more HLA services, like DDM. This may not always be possible, for example because of time constraints, which makes HLA-to-HLA bridging the only viable solution.

One long-term architectural use case is when a federation is partitioned into two federations with different security levels. The HLA-to-HLA bridge filters and modifies data before sending it from the higher to the lower security domain. Another case is hierarchical federations, where a number of sub-federations connect to a super-federation. This is useful where the FOM in the super-federation is on a more aggregated level than in the sub-federations. It is also useful for fault-tolerance in highly distributed federations where local sub-federations can keep running even when the super-federation is unavailable.

This paper presents and discusses a number of use cases, including experiences from real life federations and implementations of HLA-to-HLA bridges.

1. Introduction

This paper describes how HLA-to-HLA bridging can be used for solving common interoperability problems. It also gives a number of practical use cases and describes some experiences.

1.1 Bridging different M&S standards

When different architectures are used in the same simulation environment we sometimes need to build bridges between them to make them interoperate. In the modeling and simulation communities, in particular within aerospace and defense, the two most commonly used standards are the High-Level Architecture (HLA) [1] and Distributed Interactive Simulation (DIS) [2]. There are also related standards like Data Distribution Service for Real-time Systems (DDS) [3] and Web Services [4] that are commonly bridged to HLA and DIS. Another common framework in the US test an training ranges is the Test and Training Enabling Architecture (TENA) [5], which is sometimes bridged to HLA or DIS. Test ranges in other countries often use proprietary Time-Space-Position-Information (informally known as "TSPI") protocols. For command and control training it is common to feed simulated data through bridges and gateways to proprietary Command and Control (C2) protocols. Voice and video are increasingly important components of modern training, which means that streaming protocols are also bridged with M&S standards.

The Live Virtual Constructive Architecture Roadmap (LVCAR) [6] covers many aspects of bridging between different defense architectures, at least from a US Department of Defense perspective. This work has since continued, for example with the LVCAR-I Gateways and Bridges [7].

1.2 Why HLA-to-HLA bridging?

Obviously a bridge makes it easy to combine and reuse systems that implement different standards. Assuming that we have implemented HLA in all of our systems, do we still need bridges? This paper argues that connecting different architectures are not the only reason to use bridges. Some other reasons are:

- Partitioning one federation into several federations, for example for robustness, scalability or security
- Overcoming minor differences in how federates implement the federation agreement
- Handling issues with certain federates

This paper starts with a short technical analysis and then presents a number of practical use cases.

HLA-to-HLA bridging is also somewhat related to the work of the SISO RTI Interoperability Study Group [8] back in 1999. The experiences presented below matches well the analysis made in this study, in particular the Homogeneous vs Heterogeneous FOM/RTI analysis and the Model-Service-Communication interoperability model.

2. A Technical Look at Bridging

This section provides some technical background and describes the bridging software used in the use cases.

2.1 Technical analysis of bridging in general

Interoperability architectures usually differ in several aspects, as shown in Figure 1.

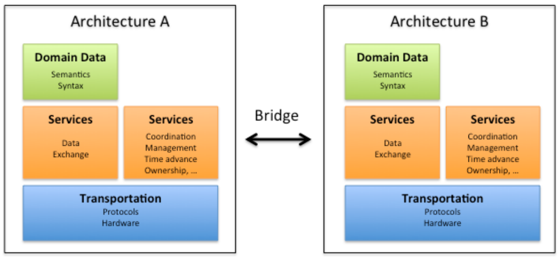

Figure 1: Bridging Aspects

From the bottom and up, the main aspects are:

1. **Transportation**, for example TCP or UDP protocols, protocols based upon this such as http, shared local or reflective memory, etc.
2. **Services** such as the different mechanisms for exchanging data, coordination and management of federations, transfer of modeling responsibility, heart-beating, time advance/synchronization, etc.
3. **Domain data** model for the information that is exchanged. Two important aspects are the syntax (technical representation) and the semantics of the data. This model builds upon the services for data exchange, as shown in Figure 1.

Differences in technical transportation may be the easiest difference to bridge. Differences in general services may be almost impossible to overcome, in particular if some service is unavailable in one standard, like bridging HLA Time Management to a DIS exercise. Differences in information models may require extensive knowledge of the domain for proper translation.

In the case of DIS and HLA with the Real-time Platform Reference FOM (RPR FOM) [9], the data model and the services used are almost identical. This makes it very straightforward to develop a bridge.

2.2 Technical analysis of HLA-to-HLA bridging

The architecture of an HLA-to-HLA bridge is shown in figure 2. The bridging application acts as a federate in each federation. It publishes and subscribes data in both federations according to the Federation Object Model (FOM) in each federation. It may modify or drop data as it is transferred between federations.

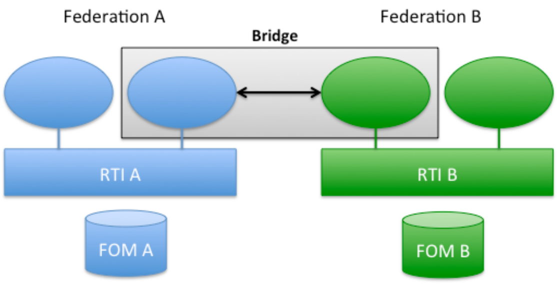

Figure 2: Typical HLA-to-HLA Bridge

Many commonly used HLA services, like Declaration Management, Object Management and DDM, are easy to bridge. Other services, like MOM, Save/Restore and Synchronization Points are more difficult to bridge, whereas Time Management and Ownership Management may not be possible to fully bridge, although many common use cases can be supported.

2.3 Overview of Pitch Extender

In this section we will look at the software used in the use cases in section 3 and 4. Pitch Extender, shown in figure 2, is a COTS product that provides HLA-to-HLA bridging. The product refers to the two federations that it bridges between as "Federation A" and "Federation B".

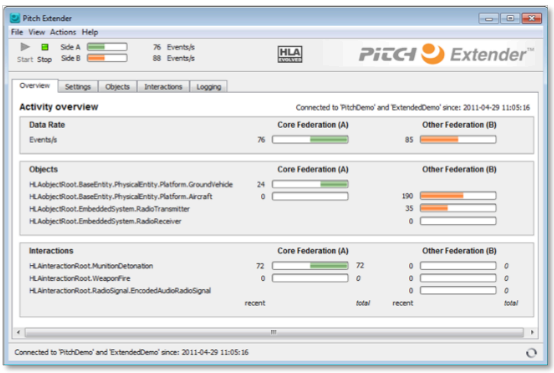

Figure 3: GUI of Pitch Extender

It is RTI independent and FOM independent. Out of the box it supports all versions of HLA and most leading RTI implementations from several vendors (Raytheon, VT MÄK, Pitch). The choice of RTIs may be different between the two federations. It is also able to load FOMs for the two federations in different HLA formats. In many cases, but not all, the same FOM is used in both federations. The user can graphically select which classes (with attributes) and interactions (with parameters) that are to be transmitted to the other federation. The RTI and FOM configuration is done with a graphical user interface before the bridge is joined to the federations. There is also a graphical monitoring interface that enables the user to visualize the information flow for each type of information during bridging. Known objects and recently bridged interactions can be listed. Figure 3 shows the graphical user interface for a typical RPR FOM bridging case.

By default the software does not modify the data in any way. This approach minimizes the processing in order to support large exercises. Depending on update size and CPU power it can typically handle tens of thousands of updates per second.

A user can then add plug-ins that process data in more or less any way that will be supported by the FOMs on each side. The most common plug-ins perform operations like filtering information and minor modifications to the data. One sample plug-in that comes with the product provides terrain-box filtering for RPR FOM entities.

By default, Pitch Extender tries to minimize the federation state that is stored for each federation. Many plug-ins need to store the state of certain object instance attribute, which means that it may become more stateful for advanced applications. Note that it is possible to enable caching of the most recent value for each instance attribute, which is useful for some applications.

3. Use Cases – Quick Fixes

This section describes a number of practical use cases for HLA-to-HLA bridging, based on real-life experience. They can be described as "quick fixes". In these cases there are better long-term solutions. Unfortunately restraints like time, budget or source code availability sometimes makes a quick fix the best short-term solution. Several of the examples are from the Viking exercises [10], a series of US-Swedish lead command and control exercises. These involve thousands of people from more than 30 countries in the training audience; military, police and civilian. The scenario includes Planning and conducting a UN mandated Chapter VII Peace Operation/Crisis Response Operation.

3.1 Reduce federate load by filtering data

In this use case, shown in figure 4, a federate, or a set of federates, cannot handle large amounts of data in a federation. As an example, in some RPR FOM based federations there may be a huge number of platforms and humans that, for example, visualization federates have a hard time handling.

The HLA-to-HLA bridge is used as follows: The same RTI and FOM are often used on both sides. A plug-in for the bridge filters out all entities that are of little interest to the constrained federates. A typical approach is to filter out all entities outside of a selected terrain box.

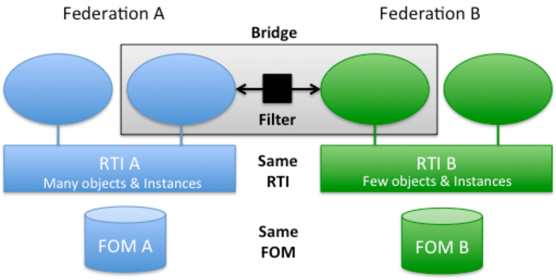

Figure 4: Reducing load by filtering

A practical example is from the Viking 11 exercise where a certain system was not able to handle all platforms simultaneously. A terrain box filter was loaded into the HLA-to-HLA gateway and coordinates were configured, i.e. creating the federation "hot spots" which allowed the system to handle those areas well.

A better long-term solution would be to implement a DDM scheme, in the federation, which would enable each system to express its explicit interest. Another solution would be for the system to be developed to better handle platforms or humans outside of the locally loaded terrain box.

3.2 Reuse federates without changing RTI

In this use case, shown in figure 5, the installed RTI cannot be changed for one or more federates. One example is where a new, large federation needs to include several previously accredited simulators. These federates typically use an older RTI or an older HLA version. The federates may not be updated or modified. The new federation typically uses a newer HLA and RTI version. The FOM is the same or a superset of the FOM used with the older federates.

The HLA-to-HLA bridge is used as follows: The bridge connects to the older RTI on one side and the newer RTI on the other side. Only the required subset of the FOM is selected for bridging. In this case no plug-in filtering is necessary. Note that there may be some technical challenges when using the same API (i.e. HLA version and language) of different RTIs, because of identical library names.

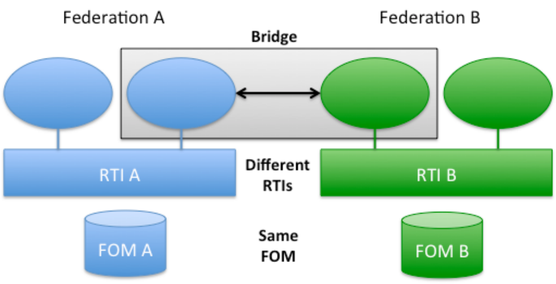

Figure 5: Reusing federates without changing RTI

One practical example is the Viking 11 exercise that included federates running legacy versions of HLA.

A better long-term solution would be to upgrade the older federates to support the new HLA version and install the newer RTI. For many applications there may also be a benefit from adding support for additional classes of the new, extended FOM.

3.3 Handle unreliable simulators

In this use case, shown in figure 6, some federates tend to crash frequently. When the federate returns the RTI will inform it of existing object instances. The returning federate will then request updates for a huge number of attributes, creating a load on other federates as well as the network. The disappearing federate may also create additional load since its object instances will be removed from the federation when it crashes and then registered again when it returns.

The HLA-to-HLA bridge is used as follows: The same RTI and the same FOM are used on both sides. The bridge caches the most recent value for all attributes, which can then be provided to rejoining federates, upon request. This means that a rejoining federate can update its state without imposing any load on the main federation. This use case is often combined with use case 3.1, reduce federate load.

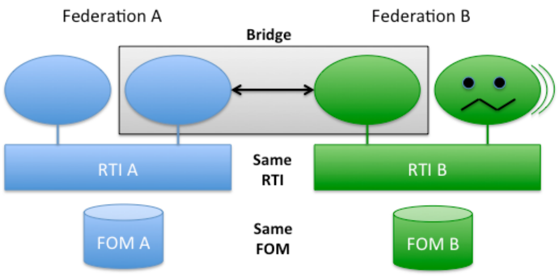

Figure 6: Handle unreliable simulators

Practical examples include some recent south European exercises where attributes were cached, and Viking 11 where some federates were prone to crashing and would otherwise have brought the federation to a halt while rejoining.

A better long-term solution would obviously be to increase the stability of these simulators.

3.4 Fix minor FOM differences

In this use case, shown in figure 7, some of the federates need to exchange data with slightly different data encoding, although the overall semantics is the same across all federates. Some examples are when certain systems have restrictions on the format of the RPR FOM marking attribute or when an enumeration describing the type of platform needs to be modified.

The HLA-to-HLA bridge is used as follows: the same RTI is used on both sides. The FOMs are slightly different. A plug-in is used to convert the attribute values uses different data encodings.

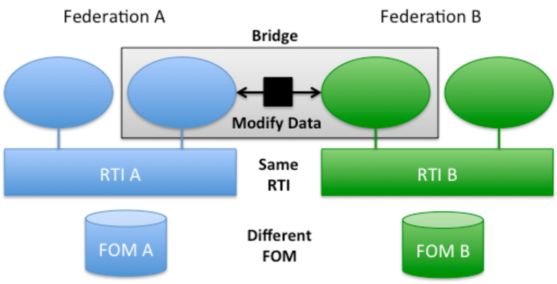

Figure 7: Fix minor FOM differences

One example is Viking 14 where a federate published objects as the wrong object type. Another example is Viking 11 where a plug-in corrected a spelling issue in object publication.

Note that this solution is only useful for relatively small modifications of the FOM since differences in FOMs usually include several attributes, additional semantics and even different sequences of interactions. In many real life cases it is also required to change the behavior of a simulator, which cannot be solved using this approach.

A better long-term solution is to adapt all federates to the same FOM and federation agreement.

4. Use Cases - Long Term Architectures

This section describes use cases that can be considered long-term architectures. In these cases the HLA-to-HLA bridge is not a quick fix but rather a desirable solution. In many of these cases the segmentation or partitioning of a federation is the most important aspect of the bridge.

4.1 Segment federation into different security levels

In this use case, shown in figure 8, a federation is partitioned into two different domains with different security classifications, such as Restricted and Secret.

The HLA-to-HLA bridge is used as follows: A set of plug-ins implements the policy that restricts the transmission of data from higher to lower levels. Data may also be modified and timing may be changed. In some cases dummy values need to be inserted as required to make all federates work properly.

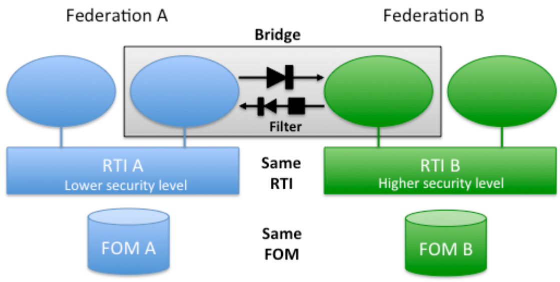

Figure 8: Segment federation into different security levels

Like any system handling classified data, security accreditation is required.

This approach was used in the French Air Force experimental federation AXED [12]. This is not a deployed security solution but rather a demonstrator testing an architecture that might be suitable to address some technical aspects of information security issues pertaining to distributed simulation.

4.2 Increase robustness by partitioning huge federations

In this use case, shown in figure 9, a huge federation, possibly distributed between different sites, needs increased robustness. It is thus partitioned into several federations, typically with each site, or each group of related federates, in one federation.

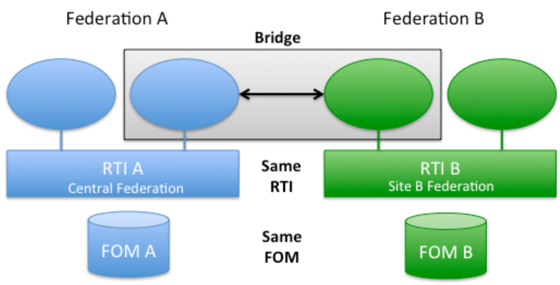

Figure 9: Increased robustness by partitioning

The HLA-to-HLA bridge is used as follows: The same RTI and FOM are used. One or more bridges are deployed between federations. In some cases this may be done in a “star” topology with a central federation, as shown in figure 10.

This approach has been used in several practical examples, like Viking 11 and Viking 14 to increase robustness, and also in the French AXED to allow for independence between sites.

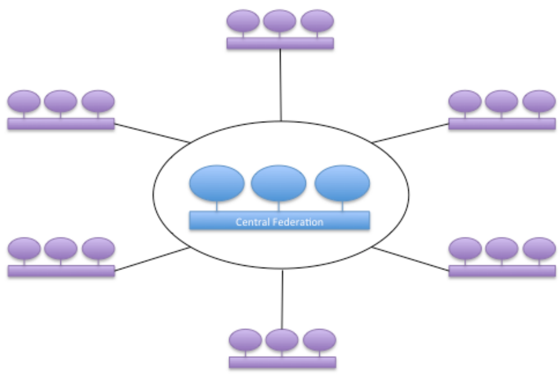

Figure 10: Central Federation in star topology

Note that there are obvious similarities with case 3.3, although this can be considered a long-term architecture for increased robustness, as opposed to the case when you need to isolate a federate that you know is unstable.

4.3 Reduce WAN link load by filtering data

In this use case, shown in figure 11, one or more federates need to run in a location with a limited WAN data link. These federates do not necessarily need to receive all of the data in the federation.

The HLA-to-HLA bridge is used as follows: The same RTI and FOM are used. A plug-in is used to remove the unneeded data.

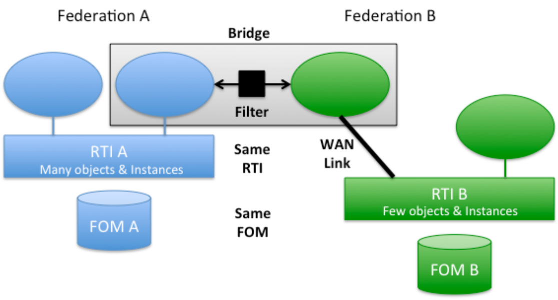

Figure 11: Reduce WAN link load

This approach has been used in several practical examples like Viking 14, where three remote sites running national Brigade level (and down) simulators connected into the main federation in Sweden. In another European federation, three sites ran a distributed setup where the main site had two simulators and the two remote sites had national Brigade level simulators connect remotely.

This use case works well if the filtering criteria are reasonably static. The filtering will become very complex if the filtering criteria are highly dynamic.

4.4 Tap exercise data to experimental federation

In this use case, shown in figure 12, an exercise or test is running while at the same time, a group of users wants to examine and experiment with the exercise data in real time. It is necessary to ensure that data can be tapped from the federation without any risk that experimental data leaks into the main federation.

The HLA-to-HLA bridge is used as follows: The same RTI and FOM are used. In the configuration the FOM data from the exercise is selected for publication in the experimental federation, but not the other way around.

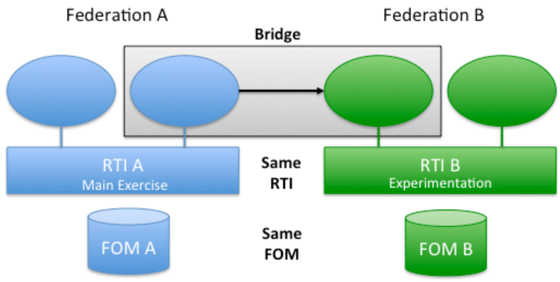

Figure 12: Tap exercise data

Practical examples include Viking 08 where data was tapped for experimentations and Viking 14 where data was tapped for recording.

There may be more reasons for using this approach, for example to separate a data logging sub-federation. It is related to the case described in section 4.2 (increased robustness).

5. Discussion

HLA-to-HLA bridges have been developed and deployed within programs for a long time. Still, they have not been discussed a lot in the SISO community. Some of the discussion that has indeed taken place focuses on hierarchical federations, which are only touched upon in this paper in use case 4.2.

5.1 Mixed-mode federations

It is possible to use HLA-to-HLA bridging between different versions of HLA, for example if one federate implements HLA 1.3 and another federate implements HLA 1516-2010. A considerably better approach for this case is to use one of the RTIs on the market that supports a mix of several HLA versions at runtime, sometimes known as mixed-mode federations [11]. In this case one federate can call the HLA 1.3 API and another federate the HLA 1516-2010 API and still operate, using more or less all of the services in the HLA standard, reducing the number of software components and getting higher performance.

5.2 Bridging and performance

Bridges will break the peer-to-peer or broadcast communications approach that many federations use. A large amount of data may need to be exchanged between the two sides of a bridge. There is a risk that the bridge will become a bottleneck in the simulation. When implementing a bridge between federations the expected data flow should be calculated and the performance of the bridging software should be investigates. This is also one reason for implementing Pitch Extender as a bridge that only handles HLA-to-HLA bridging with a minimum of processing of the bridged data. Experience shows that in most defense federations the number of updates will create higher load on the bridge than the entity count.

5.3 Using a central hub

It is possible to argue that all federates should connect directly to one central bridge (hub) that makes all required data transformations. In this way you would avoid costly modifications of systems. This argument misses the fact that an evolving degree of interoperability usually includes support for new types of information and services in a federation. The major modification doesn't lie in the format of the exchanged data, but in the intrinsic support for new data, models and services within each simulation. At the end of the day, the only viable long-term strategy for interoperability is to converge towards the same architecture (federation agreement) for the same type of problem.

6. Conclusion

This paper has shown how HLA-to-HLA bridging can be a powerful solution for common interoperability approaches. A number of real life examples have also been given.

Short-term, this approach has made it possible to quickly integrate a number of legacy systems for large exercises. The most important long-term promises in this approach lie in scalability, security and robustness.

HLA-to-HLA bridging should be further discussed within SISO, for example from the following perspectives:

- It should be included in the Gateways and Bridges studies
- Performance, architectural and HLA services aspects should be covered in the infrastructure forum
- In case the Security in Simulation Study Group restarts, or a related group is started, the potential security benefits of HLA-to-HLA bridging should be further analyzed as a potential solution.

References

[1] IEEE: "IEEE 1516-2010, High Level Architecture (HLA)", www.ieee.org, August 2010.

[2] IEEE: "IEEE 1278, Distributed Interactive Simulation (DIS)", www.ieee.org.

[3] OMG: Data Distribution Services version 1.2, http://www.omg.org/spec/DDS/

[4] W3C: "Web Services Description (WSDL) 1.1", W3C, URL: www.w3.org/TR/wsdl

[5] TENA, Test and Training Enabling Architecture, https://www.tena-sda.org/

[6] A Henninger, et. al. "Live Virtual Constructive Architecture Roadmap (LVCAR) Final Report", US DoD, September 2008.

[7] R.R. Lutz: "Live-Virtual-Constructive Architecture Roadmap Implementation: Common Gateways And Bridges Execution Plan", NSAD-R-2010-049, The Johns Hopkins University/APL June 2010

[8] SISO: "RTI Interoperability Study Group Final Report", SISO-REF-003-1999, www.sisotds.org

[9] SISO: "Real-time Platform Reference Federation Object Model 2.0 ", SISO-STD-001 SISO, draft 17, www.sisostds.org

[10] Viking 14 home page, http://www.ltc.mil.se/viking14/default.aspx, retrieved 5-Aug 2014

[11] Andy Bowers et al. "FOM Modularity and Mixed-Mode Federation Design and Development: Challenges and Observations" Proceedings of 2011 Fall Simulation Interoperability Workshop, 11F-SIW-029, Simulation Interoperability Standards Organizations, September 2011

[12] Lionel Khimeche, Jean-Louis Gougeat: "AXED – Towards a Distributed Mission Operations capability in the French Air Force", Proceedings of 2014 Fall Simulation Interoperability Workshop, 14F-SIW-006, Simulation Interoperability Standards Organizations, September 2014

Author Biographies

BJÖRN MÖLLER is the Vice President and co-founder of Pitch Technologies. He leads the strategic development of Pitch HLA products. He serves on several HLA standards and working groups and has a wide international contact network in simulation interoperability. He has twenty years of experience in high-tech R&D companies, with an international profile in areas such as modeling and simulation, artificial intelligence and Web-based collaboration. Björn Möller holds an M.Sc. in Computer Science and Technology after studies at Linköping University, Sweden, and Imperial College, London. He is currently serving as the vice chairman of the SISO HLA Evolved Product Support Group and the chairman of the SISO Real-time Platform Reference FOM PDG.

FREDRIK ANTELIUS is a Senior Software Architect at Pitch and is a major contributor to several commercial HLA products, including Pitch Developer Studio, Pitch Recorder, Pitch Commander and Pitch Visual OMT. He holds an M.Sc. in Computer Science and Technology from Linköping University, Sweden.

MARTIN JOHANSSON is a Developer at Pitch with extensive experience from product development, interoperability services and training.

LINUS LINDHOLM is a Developer at Pitch with extensive experience from product development, interoperability services and training.

PER-PHILIP SOLLIN is an Enterprise Integration Architect at Pitch and mainly responsible for creating large federations and exercises. He has worked with Integration since 2008 and has participated in planning and executing among others the Viking Exercises since 2008 and onwards as well as the SEESIM 12 exercise. He has also worked in software development as well as served on several NATO MSG:s. He studied M.Sc. in Information Technology at Chalmers University of Technology in Gothenburg, Sweden.