ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/228620538

Practical Experiences from HLA 1.3 to HLA IEEE 1516 Interoperability

Conference Paper · June 2004

CITATIONS
4

READS
2,756

2 authors, including:

Björn Möller
Pitch Technologies

52 PUBLICATIONS **271** CITATIONS

SEE PROFILE

All content following this page was uploaded by Björn Möller on 06 June 2014.

The user has requested enhancement of the downloaded file.

Practical Experiences from HLA 1.3 to HLA IEEE 1516 Interoperability

Björn Möller
Lennart Olsson
Pitch AB
Nygatan 35
SE-582 19 Linköping, Sweden
+46 13 13 45 45

bjorn.moller@pitch.se, lennart.olsson@pitch.se

Keywords:
HLA 1.3, HLA IEEE 1516, RTI, Interoperability, Life-cycles, Legacy systems

ABSTRACT: *As new versions of the HLA standard, like IEEE 1516, becomes available, many programs still have the requirement to build on investments made in simulation systems based on older versions of the standard. For some existing federates the source code may no longer be available or the cost of migrating and re-certifying a federate may be too high. A middleware that allows federates built for the HLA 1.3 standard to run in a HLA IEEE 1516 federation is one possible solution for these cases. This paper describes the important differences that such middleware need to bridge, both for the RTI interface specification as well as for the mapping of the HLA 1.3 FOM to the 1516 FOM. Possible architectures have been evaluated and a solution based on a HLA 1516 RTI has been implemented. An existing set of HLA 1.3 federates from the Swedish defense have been used to test it. The application is the Swedish Army "FbSim" tactical trainer, which includes computer-generated forces and a detailed terrain model. Some practical experiences from the tests are given together with potential HLA IEEE 1516 migration strategies.*

1. Interoperability and Life Cycles

This paper describes the "1516 Adapter for 1.3 Federates" that allows HLA 1.3 and HLA 1516 federates to interoperate. It describes the functionality, implementation and some practical experiences from using it. Before describing the solution we need to analyse why and when such a solution may be needed. The answer mainly lies in the life cycle of simulation systems and their funding.

1.1 Life cycles of simulations

Simulation systems have different life cycles. Three common examples are:

A simulation system may be developed on behalf of a specific component of an organization for training or analysis within that component. As the needs of the organization changes the simulation is modified accordingly. The scope of the real world to be simulated may gradually expand. At some point funding may decrease or be diverted to the development of other simulations meeting similar needs, possibly with other approaches. The simulation may still be used for a short or long time with limited funding for maintenance before it reaches its end-of-life. During its life the simulation may be used as a federate in different federations with varying requirements. The need for the simulation to become a part of a larger simulation may come from the need of the components to train together with other components or to evaluate the components capabilities in joint efforts with other components.

Another federate may be a standard tool that is developed to meet general requirements from several federations. As the tool matures some concepts may be modified and features may be added. Changing requirements from users and the availability of competing tools may make the tool less attractive at some point in time. If the demand from the market decreases the resources for maintenance will decrease and the tool may finally be discontinued but nevertheless still in use by some users.

There are also situations where federates are developed mainly for use in one specific federation and are not reused after this.

1.2 Life cycles of federations

Federations also have varying life cycles. Some federations may be developed for a shorter time of use, for example a proof of concept demonstration. Other federations may live for decades, for example mission trainers used in the everyday operations of a component of an organization.

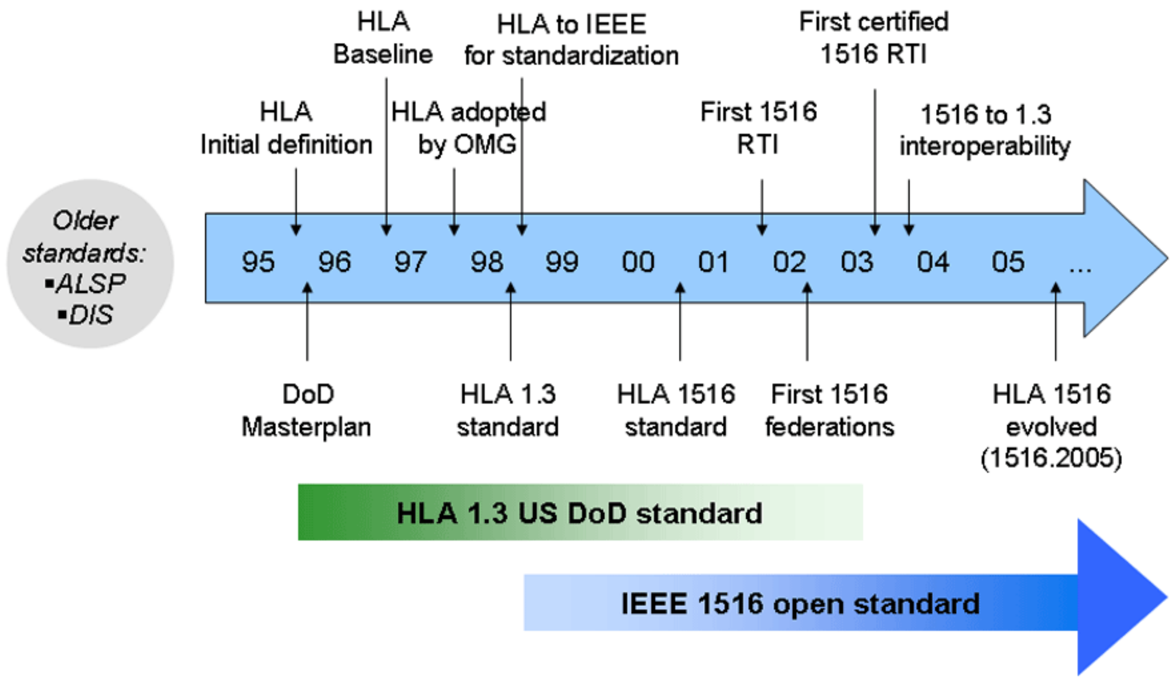

Figure 1: The development of HLA from 1.3 to 1516

1.3 Life cycles of interoperability standards

Interoperability standards may be improved during their lifetime and possibly replaced by new standards. The HLA standard for example was initiated in 1995 to supersede DIS and ALSP. The first complete version of the standard, HLA 1.3 [1], was released in 1998, an open international version, IEEE 1516 [2] was released in 2000. The next improvement, HLA 1516-2005/6 is expected to be released in 2005 or 2006.

1.4 Conclusions about life cycles

Federations need to be composed of existing and new federates, possibly COTS or in-house applications. One approach would be to always require federates to support the latest version of an interoperability standard. As can be understood from the above this may be a simple choice or pose major challenges depending on where in the life cycle different systems are and what funding is available.

2. Rationale For and Against HLA 1.3 to HLA 1516 Interoperability

Many federates have been adapted to the older HLA 1.3 standard. Many newer projects now choose the current HLA IEEE 1516 version. This may justify the use of an interoperability solution like the “1516 Adapter for HLA 1.3 Federates”. Using such a solution has some pros and cons.

2.1 Advantages with 1.3 to 1516 interoperability

The main reasons for using this kind of interoperability solution are:

It may not be possible to justify the costs for migrating an HLA 1.3 federate that is soon to be retired to HLA 1516.

It may also be extremely costly to migrate a federate from HLA 1.3 to HLA 1516. The technical migration effort for most federates can be expected to be in the range two to six man-weeks, assuming that the developer is well familiar with the federate and HLA 1.3. The main cost however is sometimes in quality assurance, certification and VV&A, which may be significantly more expensive.

In some cases migration to HLA 1516 may not be possible at all since the federation developers have no access to the source code, development competence or environment. This may be the case with COTS tools.

Another case is when a federation wishes to migrate to HLA 1516. By using an interoperability solution the federations can start using an HLA 1516 RTI immediately with federates connecting through the interoperability solution. As funding becomes available federates may migrate to HLA 1516, one by one to finally make up a native HLA 1516 federation.

2.2 Disadvantages with 1.3 to 1516 interoperability

There may however be good reasons for not using this kind of interoperability solution:

Properties of the HLA 1.3 federates will impose some limitations on the 1516 federates as described later in this paper. A mixed federation may thus be unable to benefit form all of the improvements that has been made in the 1516 standard.

A full 1516 migration of HLA 1.3 federates may yield a cleaner solution which may be beneficial in the long run.

2.3 A suggested 1516 migration strategy

The following strategy for migration is suggested:

New federates	Develop for HLA 1516
Existing federates that will be maintained and developed	Migrate to HLA 1516
Legacy federates that will not be further developed	Use "1516 Adapter" for HLA 1.3 federates

Figure 2: Suggested migration strategy

New federates are recommended to use HLA 1516.

Federates that are expected to be maintained and developed for a reasonable time are recommended to be migrated to HLA 1516.

Legacy federates that will not be further developed or where migration is not possible at all are recommended to use a 1.3 to 1516 interoperability solution such as the 1516 adapter.

3. Differences Between HLA 1. 3 and HLA 1516

This section gives an overview of the technical differences between the HLA 1.3 and the HLA 1516 standard that an interoperability solution will need to handle.

3.1 For the Object Model Template

A more complete documentation of object model is possible. This speeds up integration and is expected to reduce cost and risk. It will also introduce fewer conflicts in projects.

Standardized data types as well as standardized encoding of data has been introduced. This includes padding/alignment, variable arrays, cardinality, etc. On the lowest level it is based on BasicData, a description of data types when stored in memory and transported between hosts, for example "32 bit Big-Endian Integer".

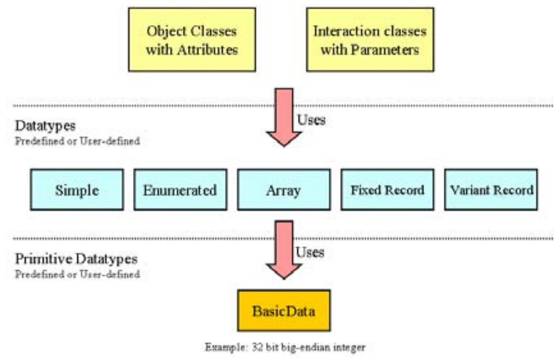

Figure 3: HLA 1516 OMT Components

The object model now uses industry standard XML format. It also uses industry standard Unicode instead of ASCII.

The FOM is used directly by the RTI. The separate FED file has been removed.

3.2 For the Interface Specification

The 1516 standard solves a number of problems found during practical use of HLA 1.3. It is also more strictly defined.

The behavior of many functions is more logical. One example is subscriptions to attributes. Consider a federate that subscribes to the attributes objectClass.A and objectClass.B. At a later point in time it subscribes to objectClass.C. In HLA 1.3 the result is that the earlier subscription is lost, which in many cases will confuse a federate developer. In HLA 1516 the subscriptions are added. To remove earlier subscriptions an explicit unsubscribe call is used.

Some additions that allow for better performance and scalability have been added, for example for simulations with many objects.

Filtering of information (with DDM) is also easier and more flexible

3.3 For the rules

No changes have been made to the rules.

4. Concept of Operation for the "1516 Adapter"

The following properties were identified as desired in the interoperability solution:

No changes should be required to a federate to use it. This means that the standard library and class named should be used. The full HLA 1.3 API should be available in these libraries.

The full HLA 1.3 functionality and semantics should be implemented.

The impact on performance should be minimal.

4.2 Interoperability using a 1516 RTI

It was also decided that the solution should be based on an HLA 1516 RTI for two major reasons:

Technical: Some services, like DDM, have greater flexibility in 1516 than in 1.3. An adapter based on a 1516 RTI can offer full 1516 DDM functionality to 1516 federates but subtract some flexibility from the interface available to the 1.3 federates. If the solution were to be based on a 1.3 RTI it would be very hard or impossible to emulate the flexibility of the 1516 DDM using a 1.3 RTI.

While DDM is crucial to some federations, many federates today do not use DDM at all. However, the use of simulation moves towards more and more distributed execution. This will provide greater value at lower cost to projects and their sponsors. For interoperability solutions to work well in these environments bandwidth reduction techniques are necessary. In the case of HLA this means using Declaration Management and DDM services.

Strategic: To encourage developers to use HLA 1516 and take advantage of the new standard and the improved functionality. Many projects that are starting around the world prefer to start afresh with HLA 1516 and the latest development environments. The resulting federates may be in use for an extended period. Because of this it can be expected that HLA 1516 RTIs will be available for a longer time and support new environments and technologies than HLA 1.3 RTIs. This is similar to the support for older and newer versions of an operating system.

4.3 Implementation of the 1516 adapter

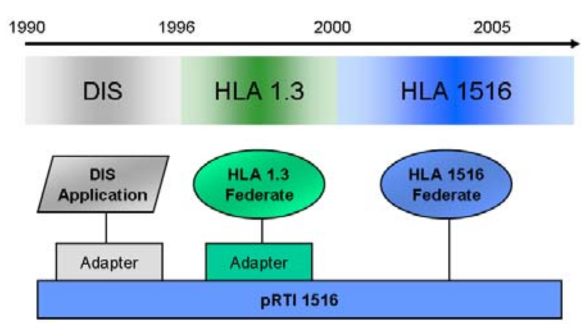

Figure 4: Adapters for legacy standards

The “1516 Adapter for 1.3 Federates” [3] has been implemented on top of a certified 1516 RTI: pRTI 1516 [4]. Together with the “DIS to 1516 Adapter” [5] it provides a way to achieve backward compatibility with older standards while allowing the user to take advantage of the latest and most powerful standard.

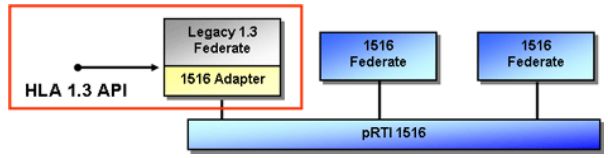

Figure 5: Providing HLA 1.3 APIs through an adapter

It provides the HLA 1.3 C++ API compatible with RTI-NG v6, the C++ API of pRTI 1.3 v2.x and a Java API compatible with pRTI 1.3 v2.x. It provides two-way interoperability between any combination of HLA 1516 and HLA 1.3 federates. The full 1.3 API is supported with the exception of two MOM interactions that exist in the HLA 1.3 specification but have no correspondence in the HLA 1516 standard.

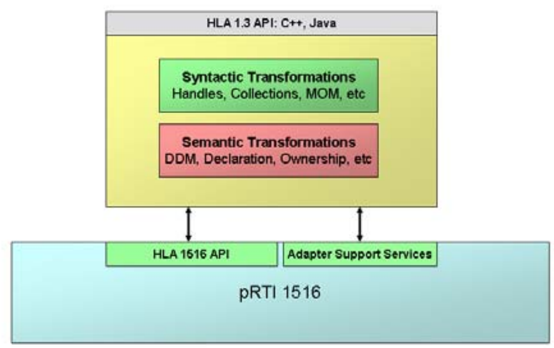

Figure 6: Transformations in the adapter

The adapter handles two types of conversions:

Syntactic transformation. This includes the conversion of different handle types and different types of collections in the API. It also handles conversion of MOM data.

Semantic transformation. This includes necessary transformations for Declaration Management, Ownership Management, DDM and more.

The adapter makes calls to the 1516 API of the RTI. A small set of additional services called the “adapter support services” has also been introduced in the RTI.

5. Considerations for a Mixed Federation

In order to build a federation that mixes HLA 1.3 and 1516 federates the end user still has to consider some issues:

The data to be exchanged must be buffer compatible [6]. As the use of the adapter implies that HLA 1.3 federates are not modified this means that 1516 federates may need to be adapted to the existing data layouts. One of the major improvements in HLA 1516 is the standardized encoding/decoding of data, which may then not be used. The older HLA 1.3 interpretation of data as a “byte array” will remain.

This constraint will only apply to parts of the FOM, namely the ones that an HLA 1.3 federate subscribes to or publishes. Note that in some cases, like the RPR FOM [7], buffer compatible FOMs are developed in parallel.

A corresponding HLA 1.3 FED file and a HLA 1516 FDD (XML) file must be used. This is usually a minor issue since a 1516 OMT tool, such as Visual OMT 1516, can generate the FED file automatically from the 1516 FOM.

Time representations that are compatible throughout the federation must be provided. This is because the RTI will have to do time calculations based on information from both HLA 1516 and HLA 1.3 federates. Compatible standard time types has been developed for the 1516 adapter but federations using specialized time representations may need to implement both HLA 1.3 and HLA 1516 time representations.

There is one situation where 1.3 DDM usage will not be compatible with 1516 DDM. HLA 1516 requires the name of a dimension to be unique. HLA 1.3 does not. If the same dimension name is used for different purposes in the HLA 1.3 federate it will not be possible to achieve HLA 1.3 to HLA 1516 interoperability when it comes to DDM.

Developers may also need to note that the name of the corresponding MOM interactions and services will be different in HLA 1.3 and HLA 1516 federates.

6. Practical Experiences

The 1516 adapter was tested at two occasions in the Swedish defense at FMV in February and March 2004. It was used to connect two federates of the same type: FbSim and PC Dart, both originally developed for HLA 1.3.

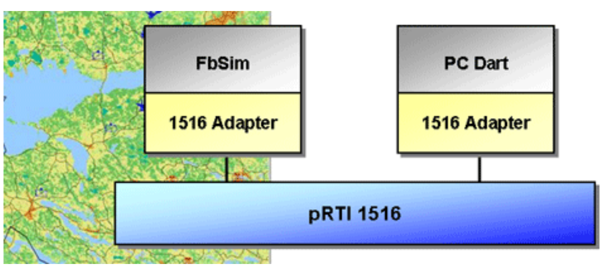

Figure 7: FbSim/PC Dart Federation

FbSim [8] is a general system for building force-level simulations. It has been developed since 1989 for the Swedish defense by several Swedish defense industries and the FOI (Swedish Defense Research Institute).

PC Dart is a system that handled short formatted text messages. It can use multiple communication modes such as Radio, LAN, etc. It is used for the majority of the command and control traffic within the Swedish Army.

These HLA 1.3 federates were connected through the 1516 adapter using the certified pRTI 1516.

6.1 Test results

In this federation the following service groups were used: Federation, Declaration, Object and Time Management. All of these services were used successfully. Other service groups are not used by these federates.

No modifications were necessary to the HLA 1.3 federates.

No performance problems were encountered. These federates do not, however, require extreme performance from the RTI.

It was noted that great care was required during the configuration of the computers executing the federates. These computers had both the pRTI 1516 and the 1516 adapter installed. They also had a native HLA 1.3 RTI installed (pRTI 1.3). Since this meant that they now had two different HLA 1.3 libraries and one HLA 1516 library installed there was some initial confusion about the configuration before the correct RTI was invoked from the federate. This has led to improvements in the installation procedures.

7. Conclusions

It has been shown that it is possible to create an adapter that allows HLA 1.3 and HLA 1516 federates interoperate using a 1516 RTI. The full API is available to the respective federates.

The adapter has been used successfully to make Swedish Army federates interoperate without performance problems.

Using an adapter to avoid 1516 migration makes sense for HLA 1.3 federates that are soon to be retired.

The drawback is that HLA 1.3 federates are likely to impose some limitations on the HLA 1516 federates. The biggest limitation is that 1516 federates may not be able to take advantage of the HLA 1516 standardized data encoding.

Full 1516 migration of all federates will yield a cleaner solution.

It is recommended that the decision whether to use the 1516 adapter is based mainly on an analysis of the expected life-length and need for further development of each federate in a federation.

8. References

[1] : "High Level Architecture Version 1.3", DMSO , www.dmso.mil, April 1998

[2] "IEEE 1516, High Level Architecture (HLA)", IEEE, www.ieee.org, March 2001.

[3] “1516 Adapter Users Guide”, Pitch AB www.pitch.se/1516adapter, June 2004

[4] Mikael Karlsson, Lennart Olsson, "pRTI 1516 - Rationale and Design," Simulation Interoperability Workshop, 01F-SIW-038, September 2001.

[5] “DIS to 1516 Adapter Users Guide”, Pitch AB, www.pitch.se, July 2004

[6] Mark Rybka et al: “Developing 1516.2 Compatible Object Models in 1.3 OMT”, Simulation Interoperability Workshop, 03F-SIW-78, September 2003

[7] “RPR FOM SISO-STD-001.1-1999”, SISO www.sisostds.org, 1999

[8] “FbSim User Manual”, Sjöland och Thyselius Datakonsulter AB, Stockholm, Sweden, Sept. 2001 (in Swedish).

Author Biographies

BJÖRN MÖLLER is the vice president and co-founder of Pitch, the leading supplier of tools for HLA 1516 and HLA 1.3. 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 MSc in Computer Science and Technology after studies at Linköping University, Sweden and Imperial College, London.

LENNART OLSSON is the chief architect of the 1516 adapter at Pitch. He has more than twenty years of experience from developing, integrating and fielding defense applications. He has been involved in the development of several HLA tools including pRTI. Lennart Olsson holds an MSc in Computer Science and technology from Linköping University, Sweden.

View publication stats