RPR FOM 2.0: A Federation Object Model for Defense Simulations

Björn Möller, Pitch Technologies, Sweden
Aaron Dubois, VT MÄK, USA
Patrice Le Leydour, Thales, France
René Verhage, CAE, Germany

bjorn.moller@pitch.se
adubois@mak.com
patrice.leleydour@thalesgroup.com
rene.verhage@cae.de

Keywords: RPR FOM, Defense, DIS, HLA, FOM Modules, Platform simulation

ABSTRACT: *The Real-Time Platform Reference Federation Object Model (RPR FOM) is the most widely used FOM for defense simulations. The main focus is on real-time simulation of war fighting scenarios including platforms, humans and weapons, however many related interactions such as radio, logistics and synthetic environments are also included.*

Version 1.0 of the RPR FOM was published in 1999; following this release the development of version 2.0 started. After having published developed several drafts, one of which was known as version 2 draft 17 and was widely used, the development slowed down. But in 2011 the work was revived.

The earlier draft standard has now been overhauled and modernized. This includes fixing many issues, providing FOM versions in the newer, modular FOM format and revising the documentation, known as the GRIM.

This paper describes the major updates of the new standard as well as experiences from drafting and standardizing this new FOM version.

We believe that the finalized RPR FOM will be of great value to governments, end users and vendors. Technically it establishes a good baseline for interoperability in defense and security training. Policy-wise it provides a vendor-neutral standard, developed in consensus by government, industry and academia. Long-term it provides a solid platform for the development of future FOM versions and for inclusion in national and international training architectures.

1. Introduction

The purpose of the High-Level Architecture (HLA) [1] is to achieve interoperability between simulations. One of the key components is the Federation Object Model (FOM), which describes the data that is exchanged between federates (simulations) during execution. The FOM is sometimes called "the language of the federation". Commonly used components of a FOM are Object Classes with Attributes, Interaction Classes with Parameters and Datatype definitions. A FOM for defense applications may for example contain classes like Aircraft, Fire and Detonation, whereas a FOM for space applications may contain classes like Planet, Satellite and Docking Request. The use of FOMs makes the HLA standard very flexible. Both standardized and custom FOMs can be easily developed.

1.1 Reference FOMs

It is common to use standardized FOMs, commonly known as Reference FOMs, and then extend them to meet the requirements of a particular project or program. Reference FOMs save time and money since efforts from earlier projects can be reused. In many projects only minor extensions need to be developed. They also reduce risk, since they have been tested, adjusted and proven to work well in previous projects.

Another important advantage is that it is considerably easier to reuse simulations supporting reference FOMs in new combinations, since they are already able to exchange data in a compatible way. This also enables a market for COTS and GOTS simulations that are interoperable.

1.2 The RPR FOM

This paper describes version 2.0 of the Real-Time Platform Reference FOM (RPR FOM) [2], which is the most commonly used FOM for defense simulations. The RPR FOM supports many common defense simulation features like:

- Platforms, such as aircrafts, ground vehicles, surface vehicles (like ships), submersible vehicles, amphibious vehicles and space crafts.
- Humans and other life forms
- Aggregates (for example a platoon) at a basic level. Note that the main focus is on platforms.
- Warfare, such as fire and detonation
- Radio communications
- Logistics
- Underwater acoustics
- Synthetic environments and mine fields
- Emissions like laser, radar and jammers
- Simulation management, like starting, stopping and pausing

As the name implies, it is focused on real-time speed simulation, as opposed to simulations that run faster or slower than real-time or as-fast-as-possible.

2. Background, History and Adoption

With the creation of the HLA 1.3 standard, there was a desire to provide a reference FOM that facilitated the transition of DIS [2] implementations to HLA. The goal was to provide an intelligent translation of DIS without trying to enhance or improve what was already present in the DIS protocol. The initial proposal was started shortly after the introduction of the HLA standard in 1996, with the SISO Standards Nomination submitted in June, 1997. The Standards Nomination was approved in June, 1998, and the kickoff of the RPR FOM Standards Development Group (SDG) occurred at SIW in the fall of 1998.

Development of RPR FOM 1.0 continued through 1998 and 1999. RPR FOM 1.0 was designed to translate the IEEE 1278.1-1995 version of the DIS standard. The final RPR FOM 1.0 standard was approved in 1999 and resulted in two documents: SISO-STD-001-1999 and SISO-STD-001.1-1999. SISO-STD-001-1999, “Guidance, Rationale, and Interoperability Modalities for RPR FOM” (GRIM), provided usage rules, definitions, and descriptions and to accompany the RPR FOM. SISO-STD-001.1-1999, “Real-time Platform Reference Federation Object Model” (RPR FOM), consisted of a document outlining the contents of the FOM as well as the machine-readable Federation Execution Data (FED) file.

After completion of RPR FOM 1.0, work immediately began on RPR FOM 2.0 to represent the IEEE 1278.1a-1998 standard. Between 1999 and 2003, seventeen draft revisions of the standard were produced. However, development then stalled and the standard was never finalized.

2.1 Restarting the RPR FOM 2.0 effort

For many years the RPR FOM 2.0 only existed as unfinished drafts. The most commonly used draft was draft 17. It contained some issues, since it was both incorrect and incomplete. The main FOM format was the HLA 1.3 OMT. An incompliant version for HLA 1516-2000 was available, forcing developers to develop their own versions for HLA 1516-2000 and HLA 1516-2010. The lack of a finalized open standard also caused some other issues since requirements during government procurement sometimes cannot be based on draft standards, only on finalized standards.

In April 2011 during the SISO Spring Simulation Interoperability Workshop (SIW) an informal meeting about restarting the RPR FOM effort was held. A follow-up meeting was held at the Fall SIW in Orlando This effort coincided very well with the NATO Modeling and Simulation Groups (MSG) 68 and later 106, who were actively working with improving the RPR FOM as the basis for the work with making existing training resources with NATO and PfP countries interoperable.

A Product Nomination (PN) of the new effort was developed. The purpose of this first effort was to produce a correct and complete RPR FOM with the same scope as the RPR FOM 2 draft 17. One key criterion was to maintain buffer compatibility with draft 17, to minimize the need for modification of existing simulations. Another criterion was to support all three HLA versions (1.3, 1516-2000, 1516-2010) in parallel. The NATO MSG 68/106 group contributed their modularized version of the RPR FOM and stated that their intent was to align with the final standard as the basis for their work.

Most of the technical work was completed by the summer 2013. The balloting was carried out early 2014. The balloting was successful with the vast majority voting for Accept or Accept with comments. Nobody voted Decline, but a few people abstained for various reasons (change of jobs, lack of time, etc). At the SISO Fall SIW in September 2014 the comments are being resolved. No additional balloting is planned. It is expected that the standard will be sent for approval to the SAC during fall 2014.

2.2 Adoption and Maturity

The RPR FOM is today probably the most commonly used FOM in US, NATO and European defense federations, both nationally and in international federations. Here are some examples with focus on international federations.

MSG-068 NATO Education and Training Network Task Group, formed in 2007, initiated the development of a distributed and networked education and training capability. MSG-068 furthermore initiated the creation of a new reference FOM called NETN FOM, which used the RPR FOM version 2.0 draft 17 as a key input. Using the FOM modularity feature of IEEE 1516-2010 ("HLA Evolved"), MSG-068 has recommended new NETN FOM modules being built on top of the RPR FOM.

As the new NETN FOM modules only used a subset of the object and interaction classes of the RPR FOM, MSG-068 did recommend modularizing the RPR FOM in accordance with HLA Evolved FOM modularity principles. This work has been further conducted as part of MSG-106 Enhanced CAX Architecture, Design and Methodology Task Group, formed in 2011.

Since the beginning, the NETN FOM has been used as a key input to various NATO and PfP exercises. Some examples are the Viking 11 and 14 projects – a series of combined civil-military crisis response operations exercises, carried out under a joint Swedish and US initiative – and the VULCAIN project – a series of CAX demonstrators in multi-level training capacity, carried out under the initiative of the French Ministry of Defense.

In the United States, many platform level federations have used the RPR FOM as a starting point on which to base their own FOMs. This enables some basic interoperability while allowing federation developers the freedom to add new classes and attributes to meet the needs of their federations. This is in no way a comprehensive list, but these are a few examples of the US federations that use the RPR FOM as their basis:

- Air Force Modeling and Simulation Training Toolkit (AFMSTT), an Air Force simulation for training senior commanders and battle staffs
- Joint Land Component Constructive Training Capability (JLCCTC) Entity Resolution Federation (ERF), which provides training for Army Commanders and their staffs
- Joint Live Virtual Constructive (JLVC), developed by the Joint National Training Capability (JNTC) to allow the Army, Air Force, Navy, and Marines to participate in joint exercises
- Joint Multi-Resolution Model (JMRM), a multi-resolution federation consisting of both the Joint Theater Level Simulation (JTLS) and Joint Conflict and Tactical Simulation (JCATS) models, with the origins of the entity-level model based in the RPR FOM
- Marine Air Ground Task Force (MAGTF), supporting the Deployable Virtual Training Environment (DVTE) Marines training simulation
- Navy Aviation Simulation Master Plan (NASMP), developed by the Naval Air Systems Command (NAVAIR) for naval aviation training
- Navy Training Federation (NTF), a component of the Navy Continuous Training Environment (NCTE) managed by the Navy Warfare Development Command (NWDC)

2.3 RPR FOM and COTS/GOTS

Due to its widespread adoption throughout the world, there is a great deal of software that supports the RPR FOM, including both Commercial Off-The-Shelf (COTS) and Government Off-The-Shelf (GOTS) products. In addition, the RPR FOM is known to be used with turn-key full-flight simulators, for internal synthetic environment simulation as well as for enabling wide area network (WAN) connectivity. The diverse array of products supporting the RPR FOM include Computer Generated Forces (CGF) applications, communications and sensor simulations, visualizers, gateways, toolkits and code generators. Listed here are a number of the software tools commonly in use. This is by no means a comprehensive list, but simply a small number of examples to represent the diversity of options available to RPR FOM based federations.

2.3.1 CGFs

CGFs model and simulate the entities that fill a simulated environment. There are many CGFs available to choose from, both as COTS and GOTS products. Some common COTS CGFs include Presagis STAGE, Ternion Flames, VT MÄK VR-Forces, and VT MÄK DI-Guy Scenario. In addition, there are a number of US GOTS products available, including Joint Conflict and Tactical Simulation (JCATS), Joint Semi-Automated Forces (JSAF), and One Semi-Automated Forces (OneSAF).

2.3.2 Network, Communications, and Sensors

Another common need within RPR FOM based federations is modeling of communications and sensors. There are several COTS products designed to perform such modeling which feature RPR FOM support, including AGI Systems Tool Kit for modeling aircraft, satellites, and ground vehicles and their sensors; ASTi Telestra HLA radio simulation; Calytrix Comm Net Radio (CNR) for simulated radio and intercom; and Scalable Network Technologies QualNet communications simulation platform.

2.3.3 Visualizers

Visualization applications can render a simulation in 2D or 3D space. These applications include Image Generators (IGs) and Stealth observers. Some of the many COTS simulation visualization software tools available are CAE Medallion, MetaVR Virtual Reality Scene Generator (VRSG), Presagis Vega Prime, URS X-IG, and VT MÄK VR-Vantage.

2.3.4 Games

Computer game based training software is increasingly used in defense applications. Some products that support the RPR FOM are Bohemia Interactives VBS 2/3 and Havok Vision.

2.3.5 Gateways

Gateways allow HLA federations to communicate with simulations using other standards such as DIS or TENA, or with other HLA federations using different FOMs. The need to bridge DIS exercises with RPR FOM federations, in particular, has led to a large number of both COTS and GOTS gateways, but many support multiple different architectures or FOMs. Available COTS gateways include Calytrix LVC Game, Pitch DIS Adapter, and VT MÄK VR-Exchange. Some of the available GOTS gateways are TENA Gateway Builder and Alion Joint Simulation Bus (JBUS).

2.3.6 Toolkits and Code Generators

Some COTS providers have developed toolkits and code generators to aid developers to quickly produce federates using the RPR FOM. Pitch Developer Studio and VT MÄK VR-Link are two such tools.

3. Structure of the RPR FOM

This section provides an introduction to the structure of the RPR FOM. A complete description is available in the GRIM.

3.1 Class Hierarchy

As object-orientation became the dominant methodology for programming in the 1990's, also the HLA allowed for the information to be exchanged to be specified in a class hierarchy. All the information in the DIS PDUs, from the 1995 version of the standard for RPR FOM 1.0 and the additional PDUs added in 1998 for RPR FOM 2.0, were analyzed for their commonality. This resulted in two hierarchies, one for (persistent) object classes (see Appendix A, Figure A1 and A2) and one for (transient) interaction classes (see Appendix A, Figure A3 and A4).

Beyond discussion, the most important object class is the BaseEntity; every entity within the simulation is represented as a kind of BaseEntity, i.e. its properties are published as an object of a class derived from this base class. There may be additional levels of specialization between this base class and the actual class representing the entity object. For example an Aircraft is a Platform, which is a PhysicalEntity, which is a BaseEntity. Through this form of generalization and specialization all common properties have been pushed as attributes in a parent class, whereas characteristics that only apply to a subset of entities are specified in a derived class.

Already DIS showed some form of the object-oriented concept composition through the use of separate PDUs to represent emission and communication. This has been represented in the RPR FOM hierarchy by the EmbeddedSystem and EmitterBeam object class structures. An object of one of the leaves from EmitterBeam points to an EmitterSystem object it originates from, and EmbeddedSystem derived objects point to the BaseEntity object it is mounted on.

Additional information exchange capabilities introduced in DIS 1998 were added to the hierarchy, in some cases as specialization of a base class already existing in RPR FOM 1.0, in other cases like the environment objects in a new class hierarchy, whereas other PDUs are represented with a single object class.

The interaction class structure is a more shallow hierarchy. In RPR FOM 1.0 only the RadioSignal, representing the actual, transient communication, has been derived into four subclasses. The addition of the Simulation Management with Reliability PDUs in the DIS version 1998 added the corresponding derived classes to the interaction class hierarchy.

In addition to the class hierarchy, also another grouping of the classes is possible. Similar to this structuring in the DIS standard, earlier versions of the RPR FOM GRIM showed this in particular for the interactions by organizing the classes into sections representing distinct functionality. With the introduction of FOM modules in the HLA 2010 standard, this also became possible for the RPR FOM itself. In section 4.1.3 of this paper an overview is provided of the different modules and the functionality domains they represent.

3.2 The FOM and the GRIM

The existence of two standards related to the RPR FOM, the RPR FOM itself and the GRIM (Guidance, Rationale, and Interoperability Modalities) document, presents another difference to the DIS standard. The DIS standard consists of a document, and a document only, providing both the technical specification of the interface as well as additional information necessary to implement the interface and realize interoperability (relaying enumeration details to SISO-REF-010 [5]). The RPR FOM however is the technical specification of the interface, to be used by the RTI implementation. And although the FOM contains semantics and even more free text information in notes (not used by the RTI), it clearly has its limitations in explaining how the interface is supposed to be used. In addition some requirements cannot be captured (other than in free text) within an HLA FOM. Therefore the GRIM is not just a reference document, but the standard describing and prescribing the intended usage of the RPR FOM. And, reflecting the origin of this reference for real-time platform-based distributed simulation on the HLA, the GRIM frequently refers back to the DIS standards. Hence a thorough and complete understanding of the RPR FOM can best be obtained by also studying the IEEE 1278.1 standard.

4. Major Changes after Draft 17

This section provides details on the major updates that have been performed during the finalization of the RPR FOM 2.0.

4.1 FOM Changes

The following major changes were made in the RPR FOM.

4.1.1 Earlier Draft 17 Comments

Respecting the earlier work of course also includes taking into account the comments that had been issued during the ballot phase of the draft 17. Some of the comments now had to be rejected in favor of maintaining buffer compatibility. Others were finally officially being addressed, such as the request to provide the FOM in IEEE 1516 OMT format. Given that today the latest version of this standard is the 1516-2010, the RPR FOM modules in this OMT format are now the normative version of the FOM.

One of the other 2003 ballot comments reflected on the incomplete semantics in the FOM. Therefore a lot of effort has been invested by the drafting group members to review the semantics, including those of the Datatypes, and adding notes for additional information. However, current users of the draft 17 are advised to verify their understanding and implementation of the RPR FOM with the latest semantics and information in the GRIM. In order to assist this activity, a comparison spreadsheet may be obtained from the SISO RPR FOM reflector.

4.1.2 Bugs

Apart from many minor changes, such as names of Datatypes, the comparison spreadsheet will also reveal that in a few cases there may be a break in the buffer compatibility. For although maintaining buffer compatibility has been a high priority objective for finalizing the RPR FOM 2.0, resolving errors identified in the earlier drafts was of course even more important.

DG members identified that there appeared to be no way to associate an ActiveSonarBeam with an ActiveSonar object. To resolve this, similar to the EmitterBeam referencing the EmitterSystem, the attribute ActiveSonarIdentifier has been added to the class ActiveSonarBeam.

Another clear mistake found was the Datatypes used for the MinefieldIdentifier and RequestingEntity-Identifier attributes of the MinefieldData, Minefield-Query, and MinefieldResponseNACK classes. Meant to reference other objects, these have been changed from unsigned long to an RTIobjectId.

Less profound was the reduction of capability, compared to DIS, in the publication of the appearance of silent entities in an aggregate. In the draft 17 the EntityAppearance field of the SilentEntityStruct had been specified with cardinality 1 (as a consequence, the field NumberOfAppearanceRecords could only contain 1). The datatype of the EntityAppearance has now been changed to an array. Backward buffer compatibility could be maintained by using a lengthless array, so that a SilentEntityStruct sent by a draft 17 implementation can still be processed by a federate implementing the final RPR FOM 2.0 standard, and this latter federate may respect the limitations of the older implementation by still publishing 1 appearance record.

A change that included more discussion within the DG, and even resulted in a dedicated paragraph in the GRIM on its usage, is the BaseEntity attribute IsPartOf. At the time of the RPR FOM development up to draft 17 only HLA 1.3 was taken into account. Since this version of the HLA standard does not include any rules on the alignment of data structure elements to their Octet Boundary Value (OBV), the RPR FOM in HLA 1.3 format includes explicit padding fields for this alignment. However, in the IsPartOf datatype these two fields were mistakenly specified with cardinality 1+. Depending on the length of the HostRTIObjectIdentifier padding may or may not be necessary. In the latter case, and a federate implementing strictly the 1+ cardinality, the (unneeded) padding will result in buffer incompatibility with the current specification of the RPR FOM.

Furthermore it must be noted that in the update of the enumerations according to the latest SISO-REF-010, some enumerators have been removed or their semantics changed. In this case it was deemed that interoperability would be better served by the alignment to the current situation than keeping the outdated, and in some cases erroneous, enumerator values.

4.1.3 Modularizing the FOM

Considering the result achieved as per MSG-106 Task Group, the PDG decided on adopting the FOM modularity principles for the definition of RPR FOM 2.0. The PDG did further improve RPR FOM modularization with the creation of new FOM modules. The end result of this activity including module dependency is shown in Figure 1.

The definition and content of upper modules from the Base module inherits from the work conducted within MSG-106 Task Group. These FOM modules are domain-related modules. The PDG further decided gathering almost all enumerated Datatypes used in the domain-related modules together in a separate FOM module called Enumerations. At the same time, the PDG created the Foundation module, which includes definitions for the basic Datatypes the upper modules from the Enumerations module refer to.

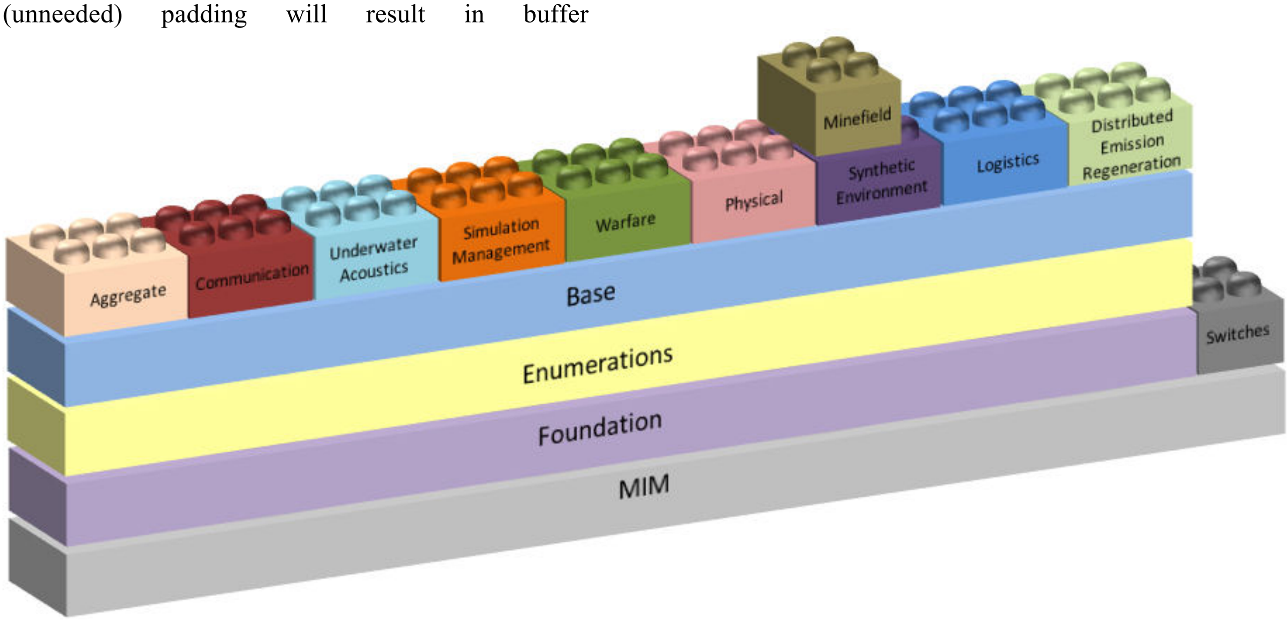

Figure 1: RPR FOM 2.0 modularization

Last, the PDG created the Switches module as a separate FOM module depending only on the MIM (MOM and Initialization Module for HLA IEEE 1516-2010).

Hence, definition and content of RPR FOM 2.0 modules is as follows:

Switches module: it provides default values for the settings required by the HLA standard to be part of a complete FOM; these switches regulate the behavior of some of the optional actions the RTI can perform on behalf of the federates;

Foundation module: it provides definitions for the basic (common) Datatypes that are used in the upper modules; it remains independent of the intended RPR FOM purpose;

Enumerations module: it provides enumerator values for many of the enumerated Datatypes that are defined in SISO-REF-010 reference document; it depends on the Foundation module;

Base module: it provides a common basis for developing RPR based FOM modules; it provides definitions for common Datatypes and the BaseEntity and EmbeddedSystem object classes; it depends on both Enumerations and Foundation modules;

Aggregate module: it provides object class definitions for representing aggregates of entities; it depends on the Base module;

Communication module: it provides object and interaction class definitions for simulating radio communications; it depends on the Base module;

Underwater Acoustics (UA) module: it provides object and interaction class definitions for representing acoustic signatures that can be detected by passive sonar sensors; it depends on the Base module;

Simulation Management (SIMAN) module: it provides interaction class definitions for managing exercise execution; it depends on the Base module;

Warfare module: it provides interaction class definitions for representing weapons, expendables and any type of explosion; it depends on the Base module;

Physical module: it provides object class definitions for representing physical entities including aircrafts, ground vehicles, ships, life forms, ammunition, etc.; it depends on the Base module;

Synthetic Environment (SE) module: it provides object and interaction class definitions for simulating the environment both under the form of (point, linear, areal) objects and processes; it depends on the Base module;

Minefield module: it provides object and interaction class definitions for representing a minefield either as a collection of mines or as individual mines; it depends on the SE module;

Logistics module: it provides interaction class definitions for simulating repair and resupply logistics services; it depends on the Base module; and

Distributed Emission Regeneration (DER) module: it provides object class definitions for representing laser, active electromagnetic emissions and acoustic emissions including active countermeasures; it depends on the Base module.

4.1.4 Enumerations in a replaceable module

The PDG decided on handling enumerations in a separate and replaceable FOM module with the perspective that this would help FOM developers aligning the enumerator values with further releases of this document. The initial Enumerations module was aligned with the 2011 edition dated 19/04/2011 of SISO-REF-010 reference document. It is expected that this module will be updated during the comment resolution, to match the most recent version of SISO-REF-010.

Discussing the perspective with the Enumerations Working Group (EWG) in charge of the governance of this document ended with a proposal from the EWG to generate the Enumerations module upon each new release of SISO-REF-010 reference document

Handling enumerations in a separate FOM module would in addition further enable easy replacement of this module with another module that may provide alternative enumerator values for the enumerated Datatypes defined in SISO-REF-010 reference document.

4.1.5 Extensibility Example: Link 16

Another example of the flexible use of the RPR FOM, of its purpose as a basis or reference FOM to be built upon, is the addition of the Link 16 BOM [4]. At the time of publication of that standard, in 2006, HLA was not yet ‘evolved’ to allow a modular structure; hence its definition as a Base Object Model (without conceptual model). Today this can easily be implemented by defining a dedicated Link 16 FOM module. This module then contains all interaction classes and Datatypes for their parameters, building upon (dependency reference) the Communications module.

However, this example also shows that it is not always possible to use the RPR FOM standard in an unmodified form, even when using modules. Although the variant record datatype in principle is very well suited for this, according to the current HLA standard it is not possible to redefine a datatype in another module. Thus the alternative added to the SpreadSpectrumStruct (now renamed to SpreadSpectrumVariantStruct) will have to be defined in a modified Communications module. Note that the discriminant datatype is already completely defined in the RPR FOM Enumerations module since it has been generated from the latest SISO-REF-010.

4.1.6 RPR FOM conventions - Datatypes

Datatypes are key elements for the definition of object and interaction classes. The work done on cleaning Datatypes' identification, semantics, and naming demonstrates the wish of the PDG to improve the quality of the RPR FOM 2.0.

The PDG has worked out some naming conventions with the goal of enabling users to understand the class parameter and attribute definitions without the need to look up the exact data type definition. These also include some general rules such as: the "_" (underscore) character is not allowed anymore.

As an illustration, new simple Datatypes shall follow the template "**<property><unit><type><size>**", where:

<property> indicates the meaning of the data;

<unit> represents the unit of the property;

<type> indicates the basic data representation's type; and

<size> indicates the number of bits in the basic data representation.

For example, TemperatureDegreeCelsiusFloat32 data type defines the temperature of an object in degree Celsius; its basic data representation's type is a 32 bit floating point number.

Similarly, new array Datatypes shall follow the template "**<datatype>[encoding]Array[cardinality]**", where:

<datatype> indicates the data type of an element in the array;

[encoding] indicates how the array is encoded, this information is omitted when the encoding is of a predefined type; and

[cardinality] indicates the size of the array, this information is omitted for dynamic arrays that may be of any size and is set to "1Plus" for dynamic arrays that must have at least one element.

For example, WorldLocationStructLengthlessArray data type defines a dynamic array of elements, each represented by the WorldLocationStruct data type; it is encoded as RPRlengthlessArray.

4.1.7 RPR FOM Conventions - Notes

In connection with the Datatypes naming conventions, the PDG has worked out some naming conventions for the notes, again to improve the quality of the RPR FOM 2.0. All unreferenced or duplicate notes have been accordingly removed from the RPR FOM modules.

Thus, new notes shall follow the template "**RPRnote<moduleName><number>**", where:

<moduleName> represents the name / acronym of the FOM module; and

<number> indicates the note number (two digits).

4.1.8 RPR FOM Conventions - FOM formats

Version 2.0 of the RPR FOM standard consists of both the GRIM (delivered as a PDF file) and a collection of FOM modules (delivered as a collection of XML files) according to the principles of HLA IEEE 1516-2010. The delivery package also includes versions of the RPR FOM compatible with former versions of the HLA standard i.e.:

- OMT and FED files matching HLA (DoD) 1.3;
- XML file matching HLA IEEE 1516-2000; and
- A monolithic XML file matching HLA IEEE 1516-2010 for those federations that do not implement the FOM modularity principles.

4.2 GRIM Changes

The following major changes were made in the GRIM.

4.2.1 Support for HLA 1.3, IEEE 1516-2000, and IEEE 1516-2010

Since the publication of RPR FOM 1.0 in 1999, the IEEE 1516-2000 and 1516-2010 HLA standards have both been published. Previous drafts of the GRIM focused only on HLA 1.3, sometimes resulting in information that was incorrect for the newer HLA standards. The GRIM has been updated to account for the differences between the three HLA standards where necessary.

The section on the Implementation of Transfer Control was particularly specific to the HLA 1.3 standard. This section was rewritten and reorganized, removing the details of transfer control that are specific to each HLA standard. To implement transfer control with the RPR FOM, users must refer to the HLA standard they are using in conjunction with the GRIM.

4.2.2 Review, Correction, and Clarification

The GRIM was fully reviewed by the PDG. A number of errors were found, including inconsistencies with IEEE 1278.1, the RPR FOM, and other sections in the GRIM. These errors and inconsistencies were rectified. In addition, a number of areas required further detail or clarification. The descriptions of many object classes, attributes, interaction classes, and parameters were improved.

4.2.3 Reorganization

Many sections of the GRIM were reorganized. This was done primarily to collocate related ideas. The most dramatic reorganization was to group the class descriptions by FOM module rather than general functional areas. In many cases the FOM modules corresponded well with the previous functional areas, but not all. This structure allows readers to more easily locate classes within both the FOM modules and GRIM, while maintaining a logical functional grouping for those not using FOM modules.

5. Conclusions and Road Ahead

We believe that the finalized RPR FOM 2.0 will be of great value to governments, end users and vendors. Technically it establishes a good baseline for interoperability in defense and security training. Policy-wise it provides a vendor-neutral standard, developed in consensus by government, industry and academia. Long-term it provides a solid platform for the development of future FOM versions and for inclusion in national and international training architectures.

5.1 Road ahead

When the RPR FOM PDG was restarted it was decided that the first step would be to finalize an RPR FOM version with the same scope as draft 17. This would include bringing it up to date, solving all known issues, developing a complete and correct set of data type descriptions, support for all three HLA versions and modularization. The result forms the foundation for future work. It is now time to look at the road ahead for the RPR FOM.

A mechanism for continued synchronization with the enumerations provided in SISO-REF-010 has now been established. As long as no major restructuring takes place, it will be a smooth process to replace the initial enumerations FOM module with future versions of these enumerations.

On a technical level, some of the data representation in the RPR FOM needs to be modernized and made more native to HLA, in particular the handling of arrays.

The Link 16 BOM is increasingly used, more or less as a FOM module, requiring minor modifications of the RPR FOM 2.0. It would be a minimal effort to include it in a future RPR FOM in a proper way.

The RPR FOM of today is mainly platform oriented. It needs to be analyzed whether a more extensive support for aggregate level simulation should be added as part of the RPR FOM or possibly as a separate effort. Support for multi-resolution modeling, where platforms and higher-level echelon units co-exists and interacts is also related to this.

The NATO MSGs perform extensive development and experimentation based on, among other things, the RPR FOM. A continued liaison with these groups could be a valuable source of proven solutions that meet the requirements of NATO joint and combined training.

Several newer SISO standards that are related to defense simulations have matured since the RPR FOM was initiated. Some of these are MSDL, for scenarios and CBML for plans, orders, requests and reports. There are probably some steps that need to be taken to make sure that the RPR FOM can be effectively used together with these standards.

Simulation-to-C2 modules are often used together with the RPR FOM. This should also be discussed, possibly in relation to the ongoing CBML/MSDL efforts.

Many other extensions have been proposed for future versions of RPR FOM. These include terrain data modules, extended scenario handling, extended meteorology, multimedia (audio and video) and more. These should be analyzed.

From a process perspective it should be discussed whether FOM modules can be updated separately or if the PDG should always release a complete set of updated modules.

The most recent version of DIS (IEEE 1278.1-2012) improves existing PDUs and adds support for information operations, directed energy fire and entity damage status, which are good candidates for inclusion in future versions of the RPR FOM.

The most important question is the long-term relationship between the RPR FOM and DIS. Three possible options are:

1. The content of the RPR FOM will always be a one-to-one mapping to the DIS information model. In this case it doesn't make sense to have separate PDGs, instead a joint PDG should be established.
2. The RPR FOM PDG should focus on the requirements of the RPR FOM community and stop synchronizing with the DIS standard.
3. The RPR FOM PDG should focus on the requirements of the RPR FOM community while maintaining a reasonable compatibility with the DIS standard, for example though a well-defined subset.

The authors recommend that the next development effort for the RPR FOM starts with an analysis and open discussion about the priorities and scope of the effort, before any implementation takes place.

References

[1] IEEE: "IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)", IEEE Std 1516-2010, IEEE Std 1516.1-2010, and IEEE Std 1516.2-2010, www.ieee.org, August 2010.

[2] SISO: “Real-time Platform Reference Federation Object Model (RPR FOM) Version 2.0D17”, www.sisostds.org, September/October 2003.

[3] IEEE: “IEEE standard for, Distributed Interactive Simulation – Application Protocols”, IEEE Std 1278.1-1995 and IEEE Std 1278.1a-1998, www.ieee.org, September 1995 and March 1998.

[4] SISO: “Standard for: Link 16 Simulations”, SISO-STD-002-2006, www.sisostds.org, May 2006.

[5] SISO: “Reference for: Enumerations for Simulation Interoperability”, SISO-REF-010-00v20-0, www.sisostds.org, November 2013.

Author Biographies

BJÖRN MÖLLER is the Vice President and co-founder of Pitch Technologies. He leads the development of Pitch's products. He serves on several standardization groups and has a wide international contact network in simulation interoperability. He has more than twenty-five 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 RPR FOM Product Development Group.

AARON DUBOIS is a principal software engineer at VT MÄK and is currently working on VR-Forces, MÄK's Computer Generated Forces application. Prior to that, he was the lead engineer for MÄK's interoperability products, including the MÄK RTI, VR-Link, VR-Exchange, and the MÄK Data Logger. He has been an editor of the GRIM since joining the latest RPR FOM drafting group in Fall 2012.

PATRICE LE LEYDOUR is a system architect in the field of M&S for support to bids and projects at TTS (Thales Training & Simulation). His main interests are simulation interoperability using HLA (High Level Architecture), and C2-Simulation interoperability. During Fall SIW 2012 he volunteered for the RPR FOM drafting group as FOM editor.

RENÉ VERHAGE is a software architect at CAE's office in Germany, with a focus on synthetic environment and simulator networking. Since his introduction to DIS and HLA in 1999, the topic distributed simulation and interoperability returned on his desk in the execution of many projects. In the 2011/2012 timeframe, when Björn Möller started an attempt to revive the RPR FOM PDG, he realized that also his experience with the RPR FOM had to be shared with the community. Hence he volunteered for an active contribution as one of the FOM editors in the Drafting Group.

Appendix A: Object and Interaction Classes

This appendix provides the RPR FOM 2.0 Object and Interaction class hierarchies.

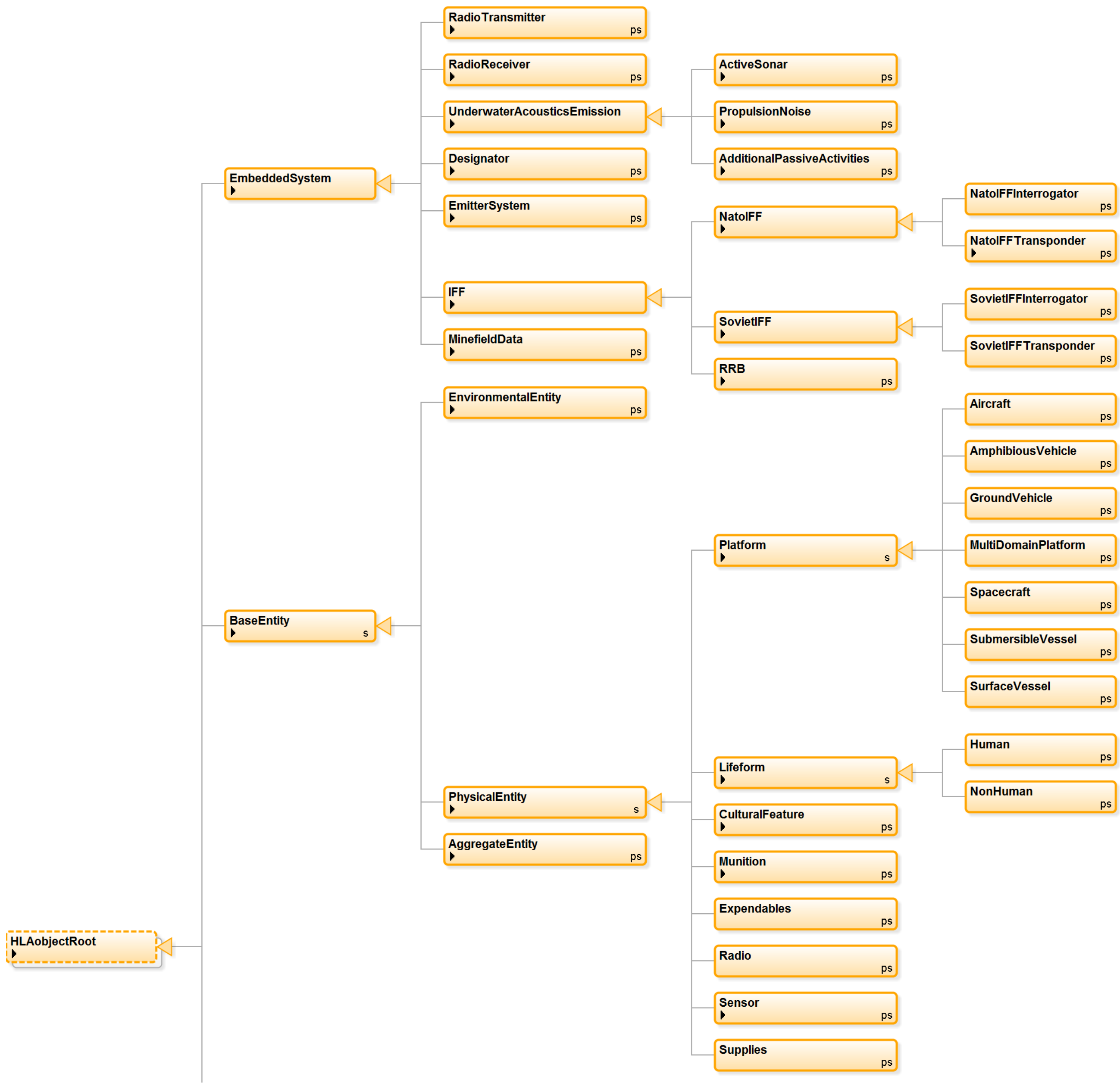

Figure A1: RPR FOM Object class structure, part 1

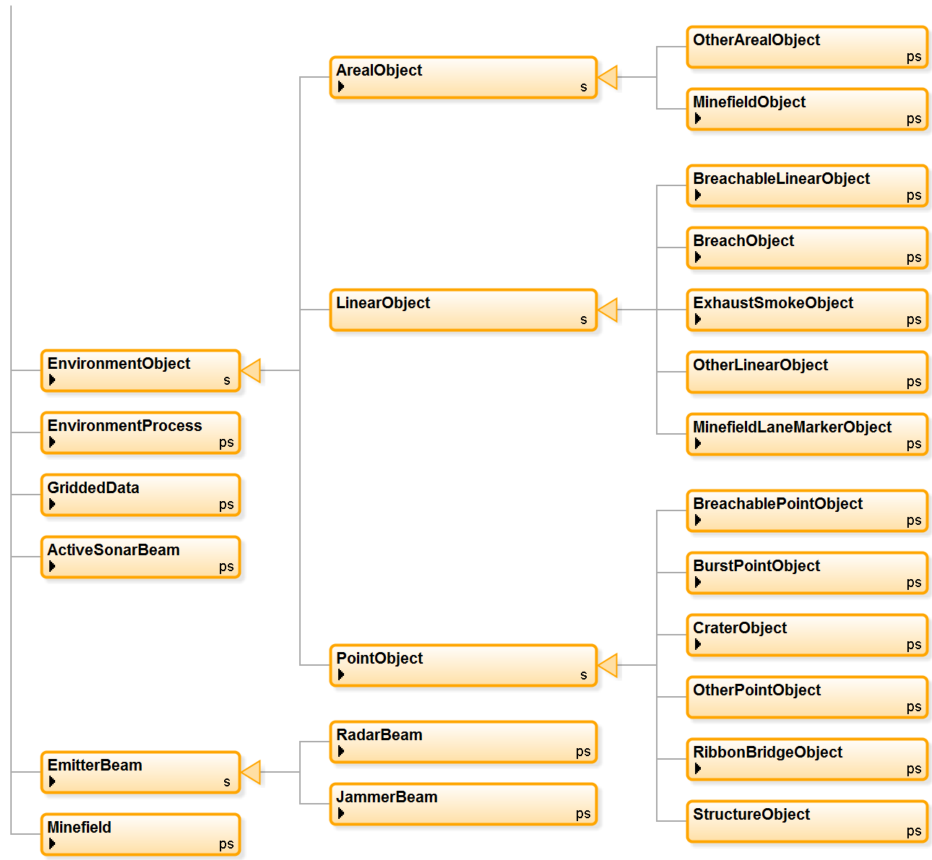

Figure A2: RPR FOM Object class structure, part 2

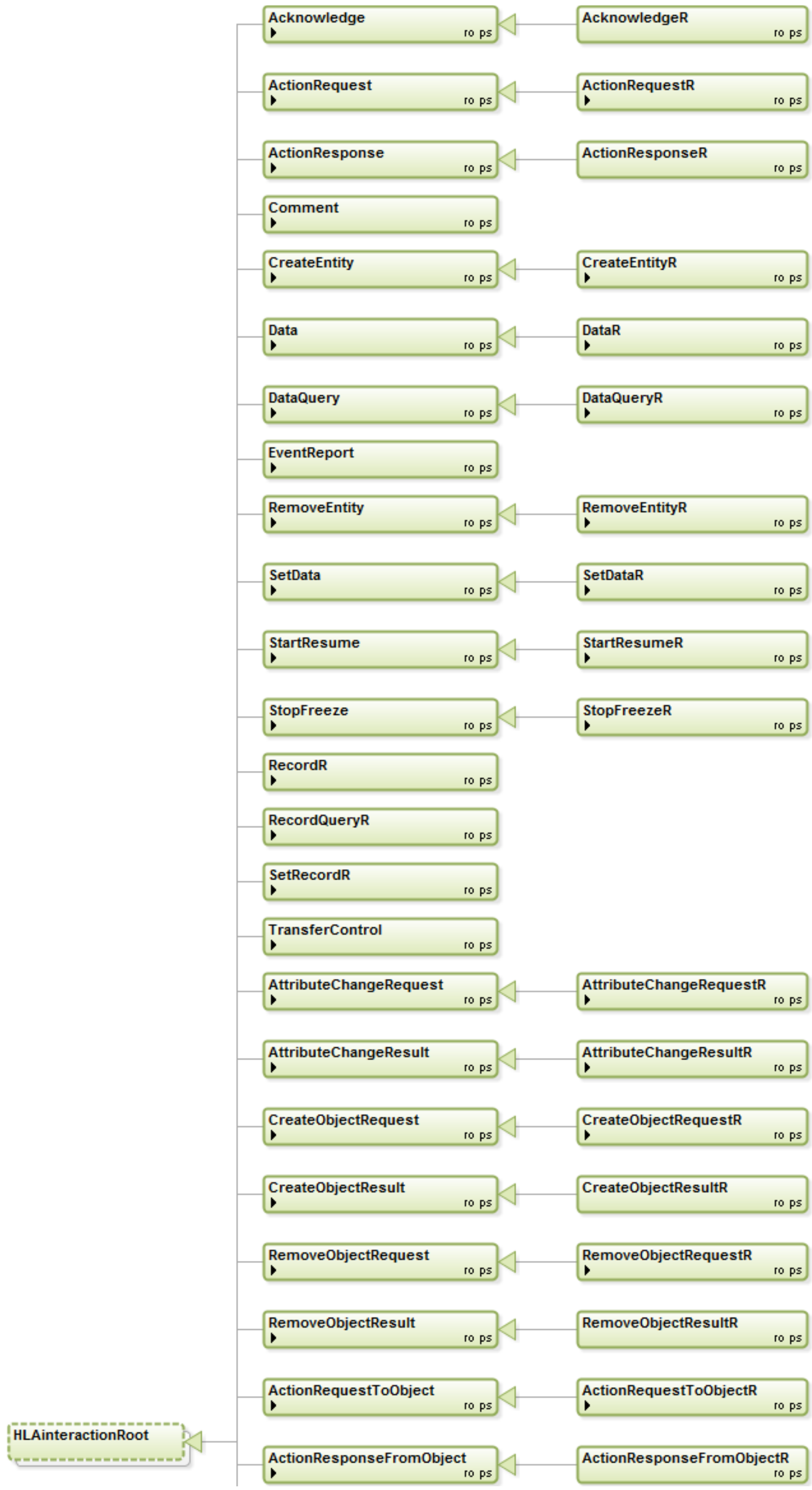

Figure A3: RPR FOM Interaction class structure, part 1

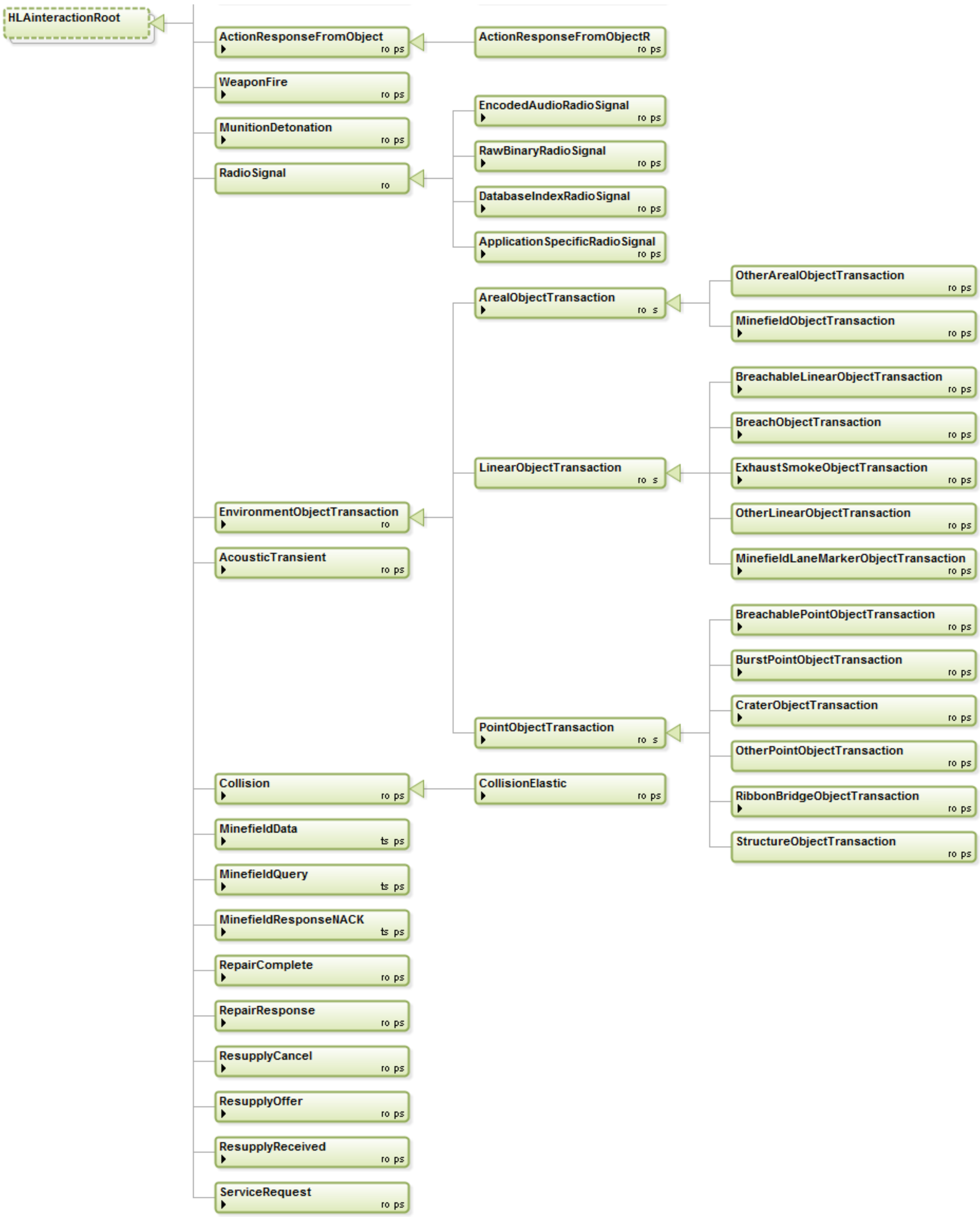

Figure A4: RPR FOM Interaction class structure, part 2