Overview and Migration Advice for the New RPR FOM 3

Björn Möller
Pitch Technologies
Repslagaregatan 25
S-582 45 Linköping, Sweden
bjorn.moller@pitch.se

Aaron Dubois
MAK Technologies
10 Fawcett Street
Cambridge, MA, USA
adubois@mak.com

René Verhage
CAE
Steinfurt 11
52222 Stolberg, Germany
rene.verhage@cae.com

Keywords:
HLA, DIS, Distributed simulation, Defense simulation, RPR FOM

ABSTRACT: *The RPR FOM (Real-time Platform Reference FOM), produced by SISO, is the most widely used HLA Federation Object Model for defense and security simulation. The new version 3 is planned to go to balloting during the first half of 2023. This paper provides an overview of the proposed new and updated features, as well as some advice for users wishing to migrate to the new version.*

Many updates originate from DIS version 7. One major update is the support for additional IFF (Identification, Friend or Foe) modes, where Mode 5 and Mode S are now supported, including interactive mode. Another major update is the support for additional appearance and capabilities for platforms. Directed energy weapons are now supported, with stateful directed energy fire. A new FOM module for Information Operations, such as electronic warfare (EW), computer network operations (CNO), psychological operations (PSYOP), military deception (MILDEC), and operations security (OPSEC), has also been added.

A new time representation is proposed, using a 64-bit integer indicating microseconds since January 1, 1970. This is in line with the proposed time representation of DIS 8, and enables the use of HLA Time Management services. A new appendix with proposed solutions for initialization of shared data across a federation is also included.

The RPR FOM benefits from the efforts of the SISO-REF-010 group, where enumerations are continuously being extended. These are included in a continuously updated version of the Enumerations FOM module.

A dedicated section of the RPR FOM 3 GRIM standard provides information for migration of federates and federations. Some updates, in particular of data types, will impact all users. The IFF module has been restructured and requires all federates using IFF to be updated. Lists of updated classes and attributes and parameters will be provided. In addition to this, the Product Development Group plans to provide additional migration tools, comparing RPR FOM version 2 and 3, separately, to avoid overloading the standard.

We believe that the RPR FOM 3 will be a valuable update for the distributed simulation community, providing new and refreshed capabilities.

1. Introduction

The Real-time Platform Reference FOM (RPR FOM) [1] is the most widely used Federation Object Model (FOM) for distributed simulation using the High Level Architecture (HLA) [2]. It is an open international standard through the Simulation Interoperability Standards Organization (SISO). It builds on the heritage of the SIMNET [3] project, the first known major distributed simulation, developed for the US Advanced Research Project Agency (ARPA), and its successor, Distributed Interactive Simulation (DIS) [4], standardized through IEEE Std 1278™. The current version of RPR FOM is version 2, which roughly corresponds to DIS version 6. Version 3, that matches DIS 7 is now under development.

The purpose of this paper is to follow up on the previous paper on RPR FOM 3 [5] which gives some background on the RPR FOM. It also covers several major updates in detail, in particular Appearance and Capabilities, Identification Friend or Foe (IFF), Information Operations, Fire and detonation and Directed energy fire. This paper gives a broader overview of updates but in less depth. It also includes updates finalized after the previous paper was written, as well as migration advice.

1.1 Technical overview the RPR FOM

The RPR FOM standard consists of two parts

- The RPR FOM 3 FOM modules developed in HLA 1516-2010 OMT XML format but also available in HLA 1516-2000 and HLA 1.3 format.
- The Guidance, Rationale, and Interoperability Modalities (GRIM) document which is a PDF specifying how to use the RPR FOM.

For the RPR FOM the following FOM modules are provided:

- Foundation, providing a few generic data types
- Enumerations, with enumerated values, shared with DIS and provided by the SISO-REF-010 working group [6]
- Base, providing general properties for object classes, for example identifier, type, and spatial data
- Physical, providing object classes for physical entities like ground vehicles
- Aggregate, providing an object class for aggregate entities
- Synthetic Environment
- Minefield
- Communication, such as radio and the base class for datalinks
- Distributed Emission Regeneration
- Underwater Acoustics
- Warfare, such as fire and detonation
- Logistics
- Simulation Management
- Switches, specifying HLA switch settings
- Information Operations. This FOM module is new in RPR FOM 3 and supports electronic warfare (EW), computer network operations (CNO), psychological operations (PSYOP), military deception (MILDEC), and operations security (OPSEC).

1.2 Finalizing the RPR FOM 3

The RPR FOM 3 effort started with a Product Nomination in 2018. A Drafting Group (DG) performs the technical work and produces Product Change Requests (PCRs). The Product Development Group (PDG) votes on PCRs and drafts are built. A of January 2023, 54 PCRs have been approved by the PDG and incorporated in a draft. The most recent drafts of the FOM and GRIM are version 6.1. One more PDG meeting is expected before going to balloting. Remaining PCRs to complete and incorporate are the Migration appendix and some minor adjustments. The PDG plans to create the final draft in the first quarter of 2023 and go to balloting in the second quarter.

2. What's New in RPR FOM 3

Several major updates have already been covered in the earlier paper. This section provides an overview of all updates. It elaborates to some degree on major updates not covered in the earlier paper. Not to be underrated are the quality improvements, clarifications and cleanups that have been done, which are also described below.

2.1 Previously described updates

In 2020, a paper titled "An update on RPR FOM 3" [5] was published which gave an overview of some of the functionality to expect in RPR FOM 3.0. A summary of the major items discussed in that paper is included here, however the previous paper should be read for full details.

- The Enumerations module has been regenerated from the SISO-REF-010 XML source file. This has resulted in some datatype and enumerator name changes, as well as a few cases where compatibility with RPR FOM 2.0 has been broken in favor of aligning with the data size defined in SISO-REF-010.
- RPR FOM 3.0 has been updated to include attributes for all the Entity Appearance and Entity Capabilities records defined in SISO-REF-010. Furthermore, in some cases the names have changed to better align with the names in SISO-REF-010. Appearance and Capabilities attributes are now defined in the leaf nodes to which they apply, and not in the PhysicalEntity and Platform base classes, as they were in RPR FOM 2.0.
- DIS v7 added additional Identification Friend or Foe (IFF) capabilities including support for military Mode 5, civilian Mode S, and interactive mode, which allows for a higher fidelity representation of interrogator and transponder interactions. RPR FOM 3.0 adds support for all these capabilities. Additionally, the class hierarchy has been restructured to minimize the cases where an attribute was included in a system to which it was not applicable, while also trying to minimize attribute duplication.
- The RPR FOM 3.0 PDG discussed several approaches to how the DIS v7 Attribute PDU might be supported. This PDU introduced a capability to generically extend existing PDU types or communicate additional information not associated with any specific PDU type. To accomplish this, the Attribute PDU includes one or more Standard Variable (SV) records. While some SV records are explicitly defined, DIS allows for the possibility of defining additional records as required by the simulation. RPR FOM 3.0 has added support for all explicitly defined SV records by creating attributes or parameters for them in the associated FOM classes. In the 2020 paper noted above, there was also a plan to allow for generic extensibility through an opaque attribute and parameter defined in new RPRobjectRoot and RPRinteractionRoot base classes. All other classes in the FOM would have been derived from these classes, giving them all access to this attribute or parameter. While there were several benefits to this approach, as outlined in the paper, ultimately the RPR FOM 3.0 PDG decided not to add such generic extensibility, since, as a reference FOM, RPR already supports adding new attributes via FOM extensions. PCR-RPR-016 was therefore updated to propose adding language to the GRIM to describe how attributes included in Attribute PDUs in DIS v7 might be added as FOM extensions to the RPR FOM.

- An Information Operations (IO) module has been added to RPR FOM 3.0, which defines the new interaction classes IOaction and IOreport. These interactions will support the IO capabilities added to DIS v7 using the IO Action and IO Report PDUs. This capability will support electronic warfare (EW), computer network operations (CNO), psychological operations (PSYOP), military deception (MILDEC), and operations security (OPSEC).
- The RPR FOM 2.0 WeaponFire and MunitionDetonation classes were not sufficient to support the range of fire and detonation capabilities supported by the DIS v7 Fire and Detonation PDUs. These PDUs also explicitly support expendable fire, non-munition explosions, and the burst of expendables. To fully support these capabilities, RPR FOM 3.0 has added ExpendableFire, ExpendableBurst, and Explosion interaction classes in addition to the existing WeaponFire and MunitionDetonation. These classes will be derived from new Fire and Detonation base classes, as appropriate.
- A new DirectedEnergyFire object class has been added to the Warfare module to correspond with the Directed Energy PDU added in DIS v7. An object class was created, rather than an interaction class, because in DIS this PDU has a heartbeat.
- The Transfer Ownership section of the GRIM has been rewritten to align with the improved and clarified procedure described in DIS v7. Additionally, the TransferControl interaction class has been renamed TransferOwnership to match the name now used in DIS. All records defined in DIS to support the transportation of entity internal state will also be defined as RPR FOM datatypes.
- The RPR FOM PDG also decided to add support for some existing DIS v6 functionality that was not included in RPR FOM 2.0. This includes adding CommentR and EventReportR interactions to support the Comment-R and Event Report-R PDU capabilities in DIS, as well as adding support for the "Other Parameters" that are included in the Entity State PDU's Dead Reckoning Parameters. However, the PDG decided not to support the Intercom Signal and Intercom Control PDUs, as the existing support for the Simple Intercom method was deemed sufficient.

2.2 Additional major updates – 64-bit timestamps and initialization.

The timestamp used in RPR FOM 2 (and earlier) is a challenge for users who need to use HLA Time Management. It is based on DIS 5/6/7 timestamps and uses a 32-bit integer where 31 bits are used to specify "ticks" past the hour, where a tick is approximately 1.667 microseconds. One bit is used to indicate whether the value is absolute (synchronized with a reference clock) or relative (unsynchronized). The timestamps used for HLA Time Management are required to be strictly increasing.

For DIS 8 an improved timestamp, using 64 bits and indicating microseconds since 00:00 Jan 1, 1970 UTC, is proposed in the DIS 8 PCR-257. RPR FOM 3 uses this timestamp format. It will still be exchanged in the HLA User Supplied Tag of HLA service calls for real-time execution, a requirement based on limitations of HLA 1.3. When using HLA Time Management, the timestamp shall also be sent as the HLA Logical Time in HLA service calls.

Initialization of shared state during startup is important in many federations. The initial scenario data, such as participating platforms, their identity, type, side, and position and other state, may be loaded from local files, databases, or other means. Once available in each federate they need to be shared. A standardized pattern for initializing the shared state of a federation is also included. It is optional for federations to us it.

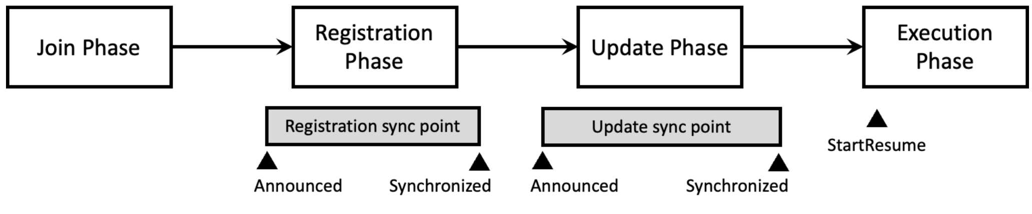

Figure 1: Phases of RPR FOM initialization

The pattern specifies four phases for the federation execution:

1. In the Join phase, federates that wish to participate in the initialization join and perform publish and subscribe.
2. In the Registration phase, federates register object instances and discover object instances from other federates.
3. In the Update phase, federates send initial attribute value updates.
4. In the Execution phase, a StartResume interaction is sent to inform federates about scenario time and to start the execution.

A dedicated federate, called the Execution Master coordinates the initialization. Other federates may be active or passive participants. An active participant would typically be a federate that publishes and subscribes to shared information. A passive participant would typically be a subscriber like a data logger or a visualizer.

The Registration and Update phases are managed using HLA synchronization points. To optimize the initialization process and minimize the data exchange for large scenarios, the HLA Auto Update switch is turned off during the initialization, but it can be switched on during the Execution phase to support late joiners.

Note that both the time representation and the initialization pattern to some degree aligns the RPR FOM with the recently released SISO Space Reference FOM.

2.3 Quality Improvements

A number of PCRs have been generated since the earlier paper to clean up discrepancies between the RPR FOM and DIS.

- The Synthetic Environment has been updated with new appearance attributes and new object and interaction classes.
- The RadioTypeStruct has been updated to include seven fields rather than six.
- The RadioTransmitter object class was updated to support the radio modulation parameters available in DIS v6 and DIS v7.
- Several changes have been made to the Minefield module to cleanup inconsistencies with DIS.
- The datatype used for the SampleRate field in the EncodedAudioRadioSignal interaction was corrected.
- A SupplementalData attribute was added to both the RadarBeam and JammerBeam object classes.
- Some simple datatypes were updated to avoid opaque data, correct integer size, correct signed/unsigned integers, and use a consistent naming scheme.
- The RadioTransmiter attribute AntennaPatternData is now using the correct datatype.
- The Distributed Emission Regeneration object classes were updated to reflect changes in DIS v7.
- The Underwater Acoustics object classes required some minor updates to reflect changes in DIS v7.
- The RTIobjectId datatype was updated to be consistent with HLA IEEE 1516 object name encoding.
- The ObjectClass and AttributeName parameters in SIMAN object classes were updated to use a datatype of HLAunicodeString.
- The RPRlengthlessArray array encoding was removed in favor of using the standard HLAvariableArray encoding.
- Padding fields were removed from datatypes where they were not needed in HLA IEEE 1516 versions of the FOM.
- The ambiguous encoding of the VariableDatumStruct datatype was cleaned up.
- Support was added for unattached RadioTransmitter and RadioReceiver objects.
- The GRIM was cleaned up to match the symbolic names used in DIS v7.
- A warning note was added to all datatypes in the Enumerations module that have a datatype size that does not match the size defined in SISO-REF-010.
- The semantics and field names of the Spatial attributes and datatypes were updated to avoid confusion.
- The naming and semantics of many datatypes, attributes, and parameters were updated for consistency.
- The EntityTypeStruct datatype was updated to better support both entities and aggregates.

3. Migrating to RPR FOM 3

During the finalization of version 2.0 of the RPR FOM, from 2012 to 2015, the primary goal was to remain buffer compatible with the so-called “draft 17”, which had been in frequent use as a de-facto standard. Hence at the time a couple of improvements, discussed in the drafting group as well as proposed in ballot comments, were not implemented. As per the product nomination, the main purpose of RPR FOM version 3 is to support interoperability with DIS version 7. In addition, the drafting group agreed to use this opportunity to resolve the pending issues identified in the past. This resulted in the (draft) RPR FOM 3 not being backward compatible with the current version of the standard. Therefore, also other PCRs proposing modifications to existing capabilities, resulting in the need to adjust federate implementations, have been approved.

The consequence is that even in the case that the added capabilities, introduced in DIS version 7, are not or not yet relevant to a particular federate, a migration is still required when upgrading from RPR FOM 2 to RPR FOM 3. The next subsection discusses three RPR FOM migration approaches. In addition, it reflects on migrating from DIS to RPR FOM. This is followed by an overview of the sources with information that may be useful during the migration.

3.1 Migration approaches

Arguably the simplest approach to developing a federate complying to the RPR FOM standard is by creating classes and data structures equivalent to the ones defined in the RPR FOM, and using these directly for creating simulation objects and events. When using a code generation tool this may even be a “no-brainer”. Hence a migration from RPR FOM 2.0 to 3.0, or even supporting both versions of the standard while others are transitioning, may be quick and cheap.
This approach works well for federates that require no or only basic simulation models to be implemented, e.g. data loggers or 2D or 3D visualization of entities. Existing logic developed for RPR FOM 2.0 can be copied to the new RPR

FOM 3.0 code base. The base capabilities of the RPR object model, such as entities having an identifier, a type, and spatial properties, are still the same. The adaptation to the changed spatial data structures is straightforward, just processing the new timestamp format requires a small modification.

However, when the federate includes more simulation logic, the more attributes and parameters of the RPR FOM classes are processed, the higher the probability that reuse of the RPR FOM 2.0 code requires significant migration effort. In such cases it may be a better approach to initially keep the RPR FOM 2.0 classes as they are and apply the adapter design pattern to integrate into a RPR FOM 3.0 based federation. This well-known design pattern fits quite well to the upgrade of an interface that is not (binary) compatible to its previous version, but for which in general existing capabilities are retained. Federates may even be built on a software architecture that prescribes to use adapters in order to easily support multiple standards and their versions.

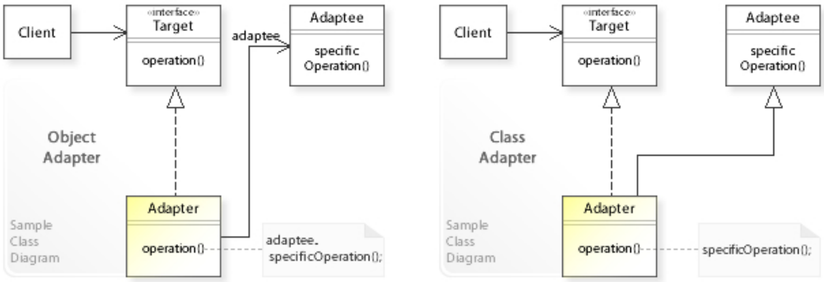

Figure 2: A sample UML class diagram for the adapter design pattern [7]

The adapter class or classes to be developed map the attributes and parameters from the RPR FOM 3.0 classes to the corresponding attributes and parameters of the existing implementation, translating any changes to the datatypes as necessary. As a result, the existing RPR FOM 2.0 federate implementation can be reused as-is, while it appears within the federation as a RPR FOM 3.0 federate. During a transition period in which both versions of the standard need to be supported, the adapter(s) can be extended with only the new RPR FOM 3.0 functionality. When RPR FOM 2.0 compliance is no longer required, the code can be successively moved to the adapter(s). Applying the object adapter pattern (as opposed to the class adapter pattern) has the benefit that a complete clean-up of any classes from previous versions of the standard can be performed in a stepwise approach.

In case not just a single federate but a set of closely cooperating federates are to be migrated, and in particular when it is not possible to update the implementation in a common framework layer, it may be preferable to plan for a stepwise migration starting with a bridge federate. Conceptually a bridge is also an application of the adapter design pattern, but on a (sub-)federation level. The bridge wraps a set of RPR FOM 2.0 federates on one side, joined in their own federation, providing a RPR FOM 3.0 interface on their behalf towards the larger federation on the other side.

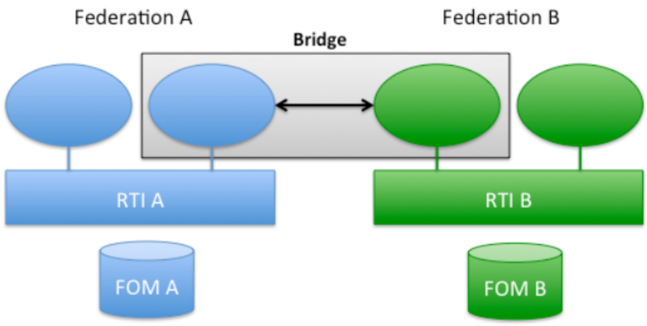

Figure 3: Typical HLA-to-HLA Bridge

Initially the development of the bridge is very similar to the adapter(s) at federate level, mapping attributes and parameters and translating between data structures. The actual migration of the federates, however, follows a slightly different path as any change in the direction of RPR FOM 3.0 also requires a corresponding stepwise migration of the internal FOM from RPR FOM 2.0 to 3.0. Although obviously the RPR FOM 3.0 draft versions have not been created with such a migration path in mind, they may be useful in a gradual update to the final version 3.0 of the standard. An even more fine-grained migration of the federates can be performed by reviewing the individual PCRs. Note, however, when cherry-picking or ordering the PCRs based on requirement priorities, that there may be some dependencies between the PCRs.

The above approaches also apply when migrating from DIS to RPR FOM. For relatively small and simple simulation applications, the risk of a one-shot migration may be acceptable and saves the costs of developing intermediate software. With increasing size and simulation logic a stepwise approach is preferable to keep the risks of regression in functionality under control. When the full extent of the functionality of the simulation application needs to be quickly made available in a RPR FOM 3.0 federation, using a DIS/RPR FOM gateway is the preferred solution.

When migration from DIS to RPR FOM is the goal, it may be preferable to develop a gateway within the own organization to build up knowledge of and experience with HLA and the RPR FOM. This may still include the use of third-party tools, such as code generators, that are also applicable when the DIS simulation application itself is transformed into an HLA federate in subsequent phases. If both the DIS and the RPR FOM standards are to be supported, then it may be more cost-effective to buy an off-the-shelf gateway solution. It may be expected that some of the well-known distributed simulation infrastructure vendors will have their gateway products updated to RPR FOM 3.0 at or soon after the publication of the standard.

Note that, even when the purpose of the RPR FOM is to ease the migration from or bridging between DIS and HLA, there are differences between the standards that restrict some of the capabilities when developing a heterogeneous DIS/RPR FOM distributed simulation. The RPR FOM extends the DIS capabilities with respect to relative positioning of entities and a more direct simulation management (SIMAN) at attribute level. In addition, with the HLA property of ownership at object attribute level, a form of shared entity ownership is possible within the RPR FOM federation that cannot be translated to the DIS exercise.
From DIS to HLA/RPR FOM, there is usually little to no reduction of the capabilities. Though, the Intercom Signal and Intercom Control PDUs remain to be unsupported by the RPR FOM and alternative solutions are in place for the IsGroupOf and IsPartOf PDUs that may require a more detailed investigation whether all required functionalities can be supported across the gateway. With respect to the new capabilities in DIS version 7 and RPR FOM version 3.0, it will be important to confirm that only the SV records defined in the DIS standard itself are being used. In case additional, custom SV records are in use within the DIS exercise, the RPR FOM will need to be extended when exchanging this data with the HLA federates is required. The RPR FOM 3.0 GRIM contains a dedicated section on extensibility (introduced in the GRIM 3.0 draft 6.1), reflecting on SV records and the Attribute PDU and providing design guidelines for such cases.

Migration can be simplified and de-risked if middleware that implements the RPR FOM 3 is available, or FOM flexible code generators that can generate code from RPR FOM 3. Similarly, software for data logging and inspection that supports RPR FOM 3 is highly useful for verifying the data exchange.

3.2 Information resources

The RPR FOM 3.0 GRIM will include a dedicated appendix intended to support the migration from RPR FOM 2.0. Whereas the appendix in the RPR FOM 2.0 GRIM is limited to simple listings of names of classes added and attributes/parameters added and removed, the new GRIM will include an indication of the type of change at the attribute/parameter level.

These changes may be the RPR FOM 2.0 attribute or parameter being renamed, moved to another class, or deleted, a change to its datatype resulting in a different encoding, a change to the update type or update condition, a minor change such as an update to the semantics, or no change at all. The appendix is structured similar to chapter 7 in the GRIM, i.e., including a table per class, grouped in a section per module. These tables support the migration by providing the perspective from RPR FOM 2.0 to 3.0. If applicable, also attributes/parameters added in RPR FOM 3.0 are added to the tables to complete the overview of the changes.

Details to the changes can be found in a separate spreadsheet. This spreadsheet is already available in the Digital Library reflecting the latest RPR FOM 3.0 draft and will continue to be maintained until the final publication of the standard. The migration appendix in the GRIM will also refer to it as a resource to find more details, e.g. on the minor changes. Although obviously the RPR FOM modules remain the authoritative sources, this spreadsheet may in particular be helpful due to the reordering of attributes, parameters, and datatypes which make it difficult to use the typical file comparison tools. In addition to detailing the class and attribute/parameter changes listed in the appendix, it also contains a dedicated tab listing all datatypes per module and their changes.

The perspective from this spreadsheet, however, is reversed. It primarily lists all the RPR FOM 3.0 classes, attributes, parameters, and datatypes. By providing both perspectives, also developers writing e.g. an adapter for a RPR FOM 3.0-based federation to internal RPR FOM 2.0-based structures can easily progress through the listings. As the Comments columns in the spreadsheet include the RPR FOM 2.0 name when it changed and also information is included on elements no longer being present, the spreadsheet application’s find tool will assist in obtaining the details when developing from the RPR FOM 2.0 perspective.

If needed, or just interested, the next level of details to the RPR FOM modifications are the Problem/Change Requests (PCRs) that formed the basis for every change to the RPR FOM 2.0. These are publicly available within the RPR FOM

PDG folder in the Digital Library. In particular for the more complex changes (e.g. entity appearance and capabilities, IFF) the analysis section of the PCR can provide useful background information. But also for new DIS version 7 capabilities that did not result in major changes (e.g. the Attribute PDU) the corresponding PCR provides the rationale for the choices made. In general, taking the PCRs as a starting point may be beneficial for gateway developers that need to upgrade their implementation to include the new DIS 7 / RPR FOM 3.0 capabilities. The section "Impact Analysis" of each PCR may support quickly estimating the effort for upgrading RPR FOM 2.0 federates.

Further information resources, in varying degree of detail, are the RPR FOM 3.0 papers (this one, [5], and [8]), the RPR FOM 3.0 Drafting Group meeting minutes (also stored in the Digital Library), and the RPR FOM PDG Discussion Board. Also the RPR FOM 3.0 in the various draft versions may be useful as each module contains use history elements referring to the PCRs that have been incorporated. Note, however, that the use history will be cleaned upon the final publication of the RPR FOM (the draft versions will remain available in the Digital Library).
The most important resources for understanding the RPR FOM remain the Standard for Guidance, Rationale, and Interoperability Modalities (GRIM) for the RPR FOM (SISO-STD-001) and the DIS version 7 standard (IEEE Std 1278.1™-2012). To prevent too much duplication of information, the former heavily refers to the latter. So even developers only dealing with HLA federations are strongly advised to have the DIS standard readily available.

Last but not least, the DIS / RPR FOM Product Support Group is the place-to-go for questions about these standards. A post on the reflector will reach more than 230 subscribers, people that have contributed to the standards as well as users, with experience ranging from a couple of years to over three decades. Also note that not only questions may be raised in the discussion group. Users are also encouraged to share their experience, hence providing support to other users and contributing to improvement of simulation interoperability in general and the standards in particular.

4. Discussion

The effort to finalize RPR FOM 2.0 focused on taking the existing RPR FOM 2 draft 17, which was widely supported within the community, and going the final mile to cleanup remaining issues and release a fully standardized version of the FOM and the GRIM. With RPR FOM 3.0, the drafting group decided to primarily focus on adding support for the new capabilities added in DIS 7, while also considering ideas that were considered out of scope for RPR FOM 2.0, such as trying to limit the use of RPR specific data representations and encodings.

4.1 RPR FOM opportunities with HLA 4

The RPR FOM 3.0 development work has been concurrent with the development of the new HLA 4 standard. Some of the features being added to HLA 4 offer interesting opportunities for future RPR FOM versions. The ability to extend existing FOM classes with new attributes via additional FOM modules provides exciting new capabilities for RPR based federations to extend the FOM without the need to edit the standard modules or add many new subclasses. Similarly, the new HLAextendableVariantRecord encoding for variant records allows the extension of existing variant records with new alternatives, without the need for existing federate encodings to be updated if they do not use the new alternatives. Finally, HLA 4 is adding unsigned integer data representations, which would allow RPR FOM to discontinue use of the RPR specific unsigned integer representation.

4.2 Road ahead

With the completion of RPR FOM 3, it is a good time to begin thinking about the future of the standard. There are multiple paths that could be considered for future versions. The most obvious need will be support for the new features being added to DIS 8. Any new additions to the DIS object model will need to be included in RPR FOM 4. There will also be a need for a RPR FOM version that supports HLA 4. Adapting the existing FOM to HLA 4 will not require much work, but the new features of HLA 4 may give RPR the chance to make additional improvements not previously possible. The biggest decision, however, is whether RPR FOM 4 should consider adding concepts to the object model outside of those just defined in DIS. Many FOMs based on RPR, such as the NATO Education and Training Network (NETN) FOM, have added a multitude of new classes to the object model that would be useful to the wider RPR user base. The drafters of the next version could consider adding capabilities beyond the scope of DIS. Alternatively, these ideas could be fed into the DIS object model, and thus eventually into RPR as well.

5. Summary and Conclusions

While migration to RPR FOM 3 from RPR FOM 2 will involve some work for existing federations, several new features, previously, some only available in DIS, may provide the impetus for exercises to make the effort. Features such as IFF modes 5 and S, IFF interactive mode, Information Operations, and Directed Energy Fire will be widely available in a standardized way for the first time. More appearances and capabilities for platforms have been added. Federations can now perform initialization of shared data in a standardized way. The support for analysis and Monte Carlo simulation has been improved since HLA Time Management can now be used. Additionally, RPR FOM 3 makes many minor improvements and necessary cleanup that was out of scope in the finalization of RPR FOM 2. The new GRIM also offers more informative sections to aid federation designers in developing their own capabilities, such as procedures for federation initialization and ownership transfer.

New federation efforts that start afresh may want to use the RPR FOM 3 from the beginning and immediately benefit from the full set of features and improved quality. Existing federations will need to weigh the benefits of the new standard vs the cost of migration. Tools for the new version will gradually become available. Tool vendors are already prototyping versions of their tools for RPR FOM3. NATO modeling and simulation groups plan to build the upcoming NATO NETN FOM version 4 on RPR FOM 3 with preliminary release 2024, which is also expected to drive a wider adoption.

6. References

[1] SISO: “Real-time Platform Reference Federation Object Model 2.0”, SISO-STD-001, SISO, www.sisostds.org

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

[3] D.C. Miller, J.A. Thorpe: “SIMNET: the advent of simulator networking”, Proceedings of the IEEE (Volume: 83, Issue: 8, August 1995)

[4] IEEE: “IEEE Std 1278.1™-2012 - Standard for Distributed Interactive Simulation - Application protocols”, www.ieee-org, August 2012

[5] Björn Möller, Aaron Dubois, René Verhage: “An update on RPR FOM 3”, Proceedings of 2020 Simulation Innovation Workshop, 20-SIW-028, Simulation Interoperability Standards Organization, February 2020

[6] SISO: “Reference for Enumerations for Simulation Interoperability”, SISO-REF-010-2022, www.sisostds.org, 17 April 2022

[7] Vanderjoe, W3sDesign, Wikimedia Commons, 1 October 2016

[8] Björn Möller, Aaron Dubois, Patrice Le Leydour, Graham Shanks, René Verhage, Fredrik Antelius: “Towards RPR FOM 3: Revisiting the Datatypes”, Proceedings of 2015 Fall Simulation Interoperability Workshop, 15-SIW-039, Simulation Interoperability Standards Organization, February 2015

Author Biographies

BJÖRN MÖLLER is the President and co-founder of Pitch Technologies. 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 a M.Sc. in Computer Science and Technology after studies at Linköping University, Sweden, and Imperial College, London. He is currently serving as the chairman of the SISO RPR FOM Product Development group, chairman of the SISO Space FOM Product Development Group. and the vice chairman of the SISO HLA Product Development Group.

AARON DUBOIS is the Director of Simulation Products at MAK Technologies. He has worked as a software engineer in modeling and simulation since joining MAK in 2007. In that time he has worked on a number of products, including the MAK RTI, VR-Link, and VR-Forces. He has been an editor of the GRIM since joining the RPR FOM 2.0 drafting group in Fall SIW 2012 and is currently serving as the vice chairman of the RPR FOM Product Development group. Aaron Dubois holds a B.A. in Computer Science from Boston University.

RENÉ VERHAGE is a system software architect and synthetic environment product expert at CAE’s office in Germany. He has been involved in many projects related to distributed simulation and interoperability and working with DIS and HLA since 1999. From 2011, he contributed to the finalization of the RPR FOM 2.0, acting as one of the FOM editors in the Drafting Group. He is currently actively involved in SISO’s RPR FOM 3.0 Drafting Group and the TADIL TALES PDG & PSG. René Verhage holds a B.Eng. in Aeronautical Engineering and a B.Sc. in Computer Science.