Simulation Interoperability on Earth and in Space: RPR FOM vs SpaceFOM

SIM-2023-Presentation-01

Björn Möller, Pitch Technologies, Sweden
bjorn.moller@pitch.se

Abstract

- Distributed simulation is a key technology, both for Earth based scenarios such as defense training, as well as for space exploration. There are also SISO standards to match: the Realtime Platform Reference FOM for defense and security and the SpaceFOM for space simulation.
- But why do we have two standards, what are the differences and are there any similarities?
- This presentation looks at the history, the use cases, the requirements, and the content of both standards and makes a comparison.
- It also looks at opportunities for making federations using the two standards work together.
- Finally, it discusses benefits of, and obstacles for long-term convergence.

Agenda

- Introduction
- RPR FOM background
- SpaceFOM background
- Emerging requirements, synergies and use cases
- Some key technical differences
- Bridging RPR FOM and SpaceFOM
- Convergence/Alignment
- Summary
- Further Reading

Generations of Standards – An HLA Perspective

Gen. 0

SIMNET

First step.
Early "de facto" standard.

Gen. 1

DIS

Open Standard.
Data model plus protocol.
SIMNET heritage

Gen. 2

RPR FOM

Object model based on the domain independent HLA standard.
Reuse of DIS data model and solutions.

Gen. 3

SpaceFOM

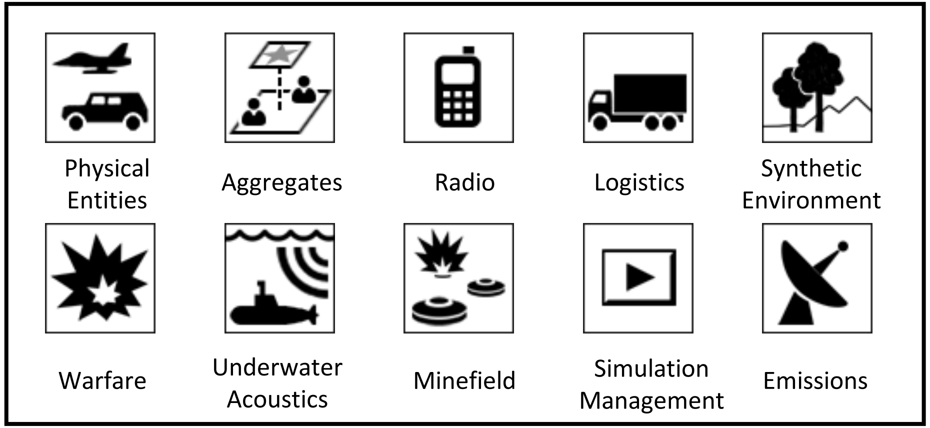
HLA only.
Less legacy.
Less ad-hoc solutions.
Extensive use of HLA features

RPR FOM

- Standardized as Real-time Platform Reference Federation Object Model 2.0”, SISO-STD-001.
- The most widely used FOM for defense and security federations, in particular platform level operations but also C2 with aggregates.
- Models physical entities (platforms, humans, etc), aggregates, weapons, radar, infrared sensors, laser designators, underwater acoustics, jamming, radio communications, collisions, logistics, synthetic environment, information operations, simulation management, start/stop.
- RPR FOM 2 matches DIS 6.
- RPR FOM 3 will match DIS 7.

Other FOM Modules Extend the RPR FOM

RPR FOM 2 Modules



NATO NETN FOM Modules

METOC	Communi-cation Networks	Entity Tasking & Reporting	CBRN
Organization	AIS	Multi Resolution Modeling	Transfer of Modeling Responsibilty

- **There are very active NATO Modeling and Simulation Groups that build modules that add capabilities to the RPR FOM.**
- **SISO provides additional standards that add Link 16 and Link 11 support.**

SpaceFOM

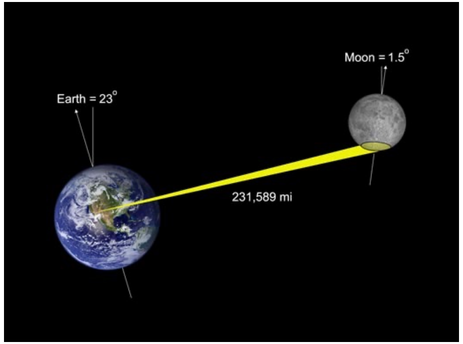
- Standardized as SISO “Space Reference Federation Object Model (SpaceFOM)”, SISO-STD-018-00-2020.
- Builds on experiences from several early NASA federations including the International Space Station Training Facility.
- Highly accurate positioning across a vast space, with exact timing required. Being 1 second off means being 7 kms off at ISS.
- Use cases: NASA Artemis “Return to the Moon”, European Space Agency Harwell Robotics lab.
- Version 1 focuses on two of the grand challenges: ”where and when”, i.e. time-space coordinates, reference frames, time handling and execution control.

Emerging Requirements, Synergies and Use Cases

- **Traditionally, users of the RPR FOM are in the defense and security community and users of the SpaceFOM are in the space community.**
- **These communities often have differing interoperability use cases.**
- **There are however several use cases that bridge these communities.**
  - The space community can reuse existing RPR FOM models for launch, atmospheric flight, and operations in Low Earth Orbit (LEO), which is an altitude of about one-third of Earth's radius, corresponding to an orbit period of less than 128 minutes.
  - The defense community can make use of the SpaceFOM to support higher fidelity modeling of satellites used for communications, navigation and reconnaissance in geocentric orbit, and operations across the Earth-Moon system (”cis-lunar space”).

Low Earth Orbit (LEO)

Earth-Moon system



RPR FOM vs SpaceFOM

Key Technical Differences

Object Classes 1(2)

RPR FOM

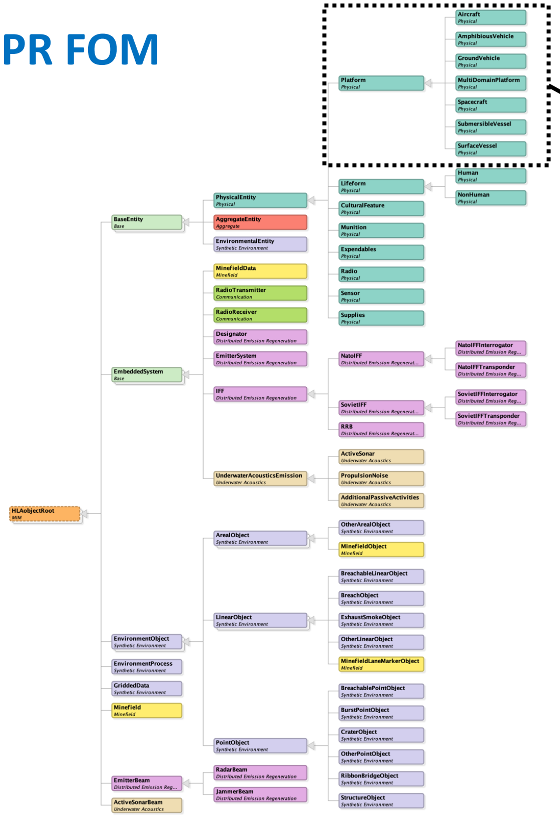
- Many specialized object classes.
- Rich set of attributes.
- Object classes for entities are subclasses of the PhysicalEntity class.
- Entity parts are often specified using subclasses of EmbeddedSystem.
- Entity Types are specified based on SISO-REF-010 Enumerations.

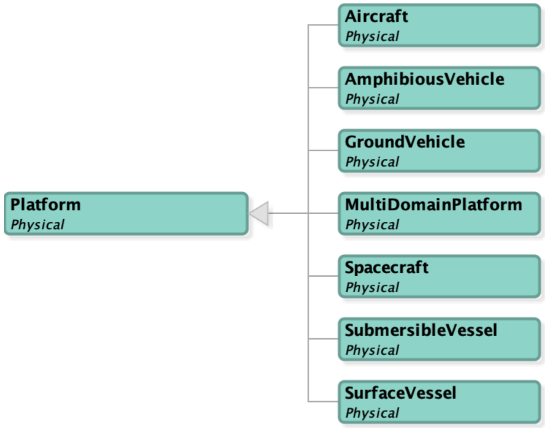
Space FOM

- A few, very generic object classes.
- Only a few, generic attributes.
- Object classes for entities are subclasses of the PhysicalEntity class.
- Entity parts are often specified using subclasses of PhysicalInterface.
- Entity Types are specified using strings, specified in the federation agreement.

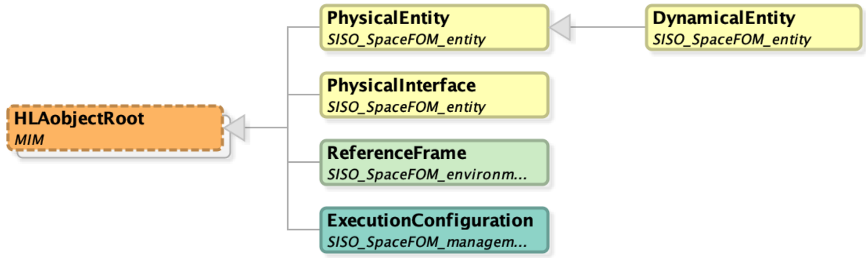
Object Classes 2(2)

RPR FOM

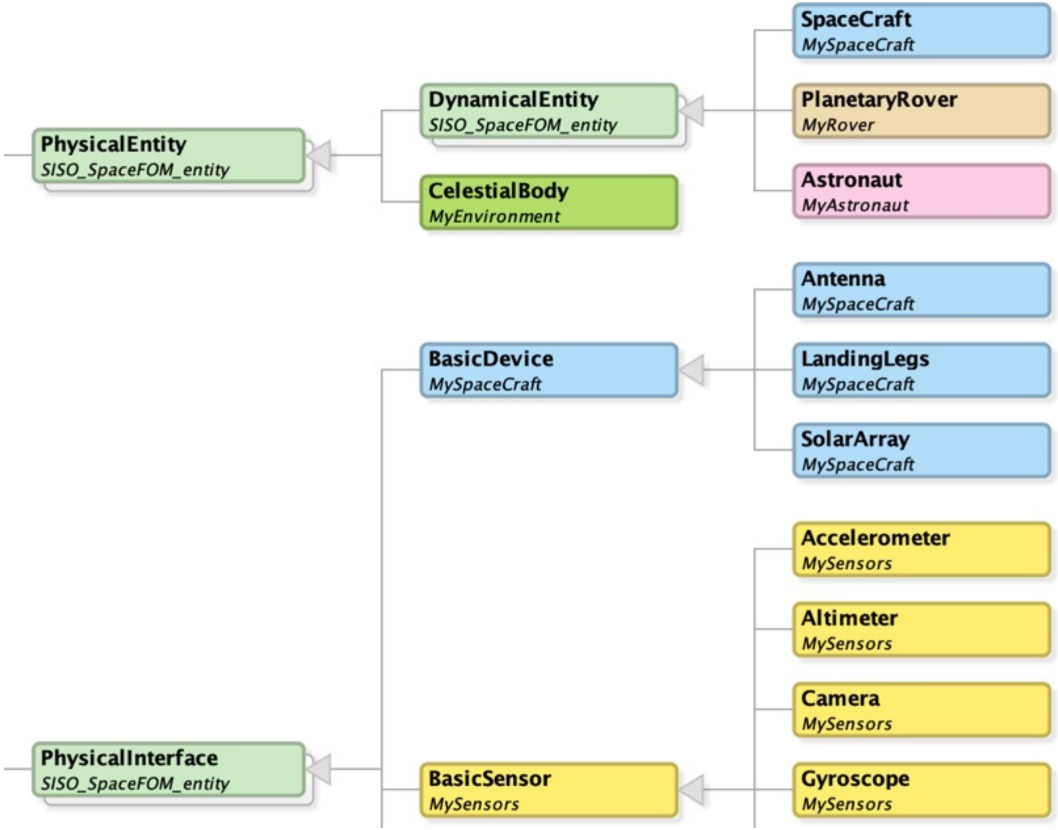




SpaceFOM



Sample Extended SpaceFOM



Spatial Representation 1(3)

RPR FOM

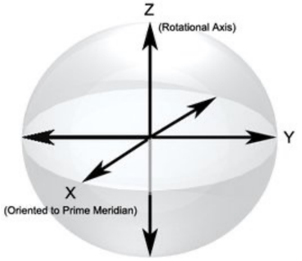
- Key data type: Spatial.
- Right-handed geocentric Cartesian coordinate system, where the origin is the centroid of WGS 84.
- Body coordinate systems are also supported.
- Orientation is described using Euler angles.
- Velocity and acceleration vectors can also be provided for example for dead reckoning.

SpaceFOM

- Key data type: TimeSpaceCoordinate
- A flexible set of reference frames can be used.
- A reference frame is an object instance.
- A standardized set of reference frames, related to the Solar system exists, but each federation can add more reference frames as needed.
- Orientation is specified using quaternions.

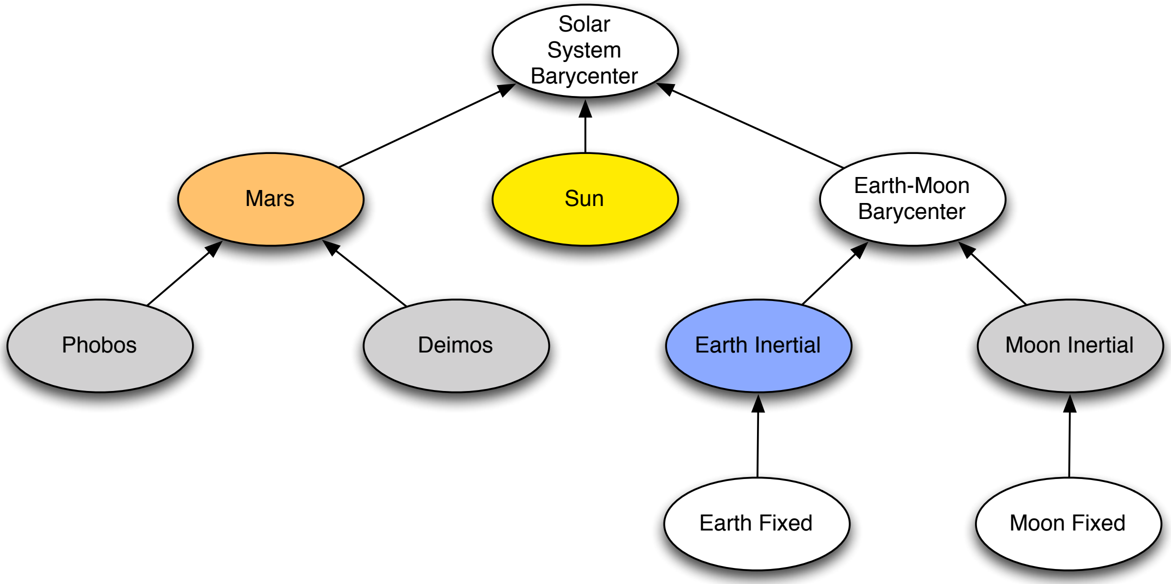
Spatial Representation 2(3)

RPR FOM Reference Frame



Earth
Geocentric
WGS84

Sample SpaceFOM Reference Frame Tree



Spatial Representation 3(3)

RPR FOM

- **Sample spatial representation: SpatialFPW struct, for object with constant velocity linear motion in world coordinates**
- **SpatialFPW fields**
  - WorldLocation (x, y, z)
  - IsFrozen (boolean)
  - Orientation (Psi, Theta, Phi)
  - VelocityVector (Xvelocity, Yvelocity, Zvelocity)

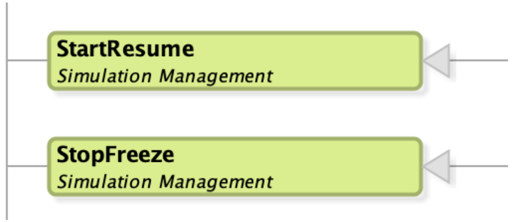
SpaceFOM

- **Each Entity provides Parent Reference Frame (String) and a TimeSpaceCoordinate**
- **Fields**
  - translational_state
    - *position (3-vector)*
    - *velocity (3-vector)*
  - rotational_state
    - *attitude_quaternion (scalar + vector)*
    - *angular_velocity (3-vector)*
  - time (simulated physical time expressed as Terrestrial Time)

Execution Control

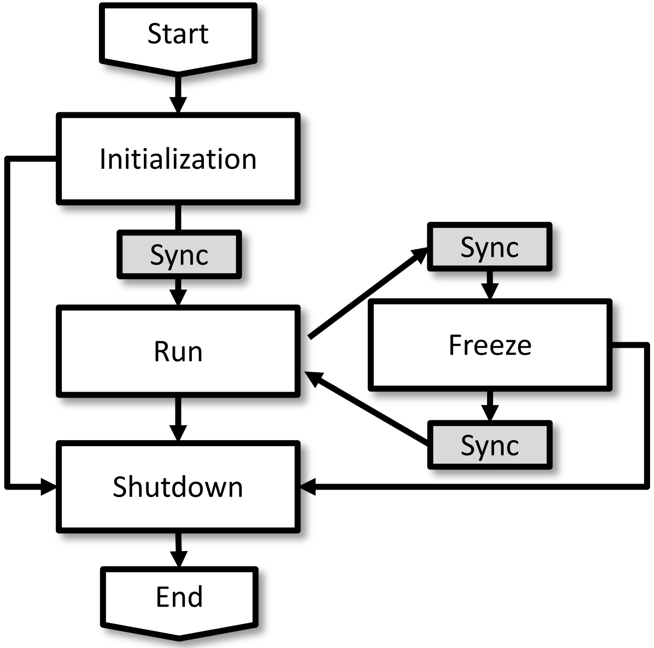
RPR FOM

- Start/Resume and Stop/Freeze interactions are provided in the Simulation Management module.
- These are building blocks that can be used as needed in a federation.



SpaceFOM

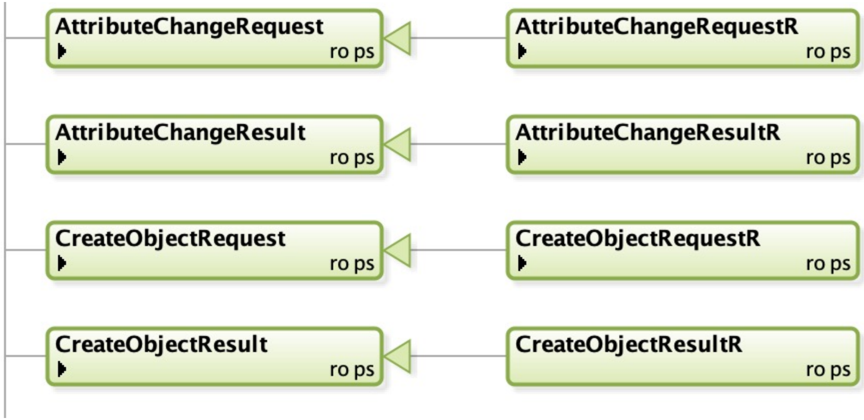
- Well defined execution flow.



Initialization

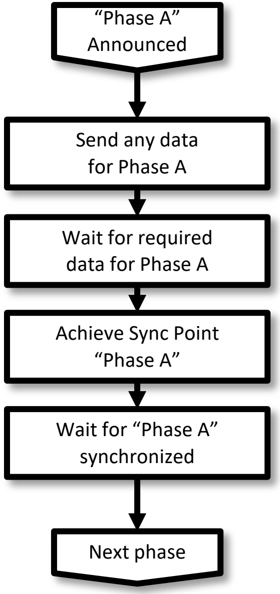
RPR FOM

- No dedicated initialization phase.
- Some interactions for initialization provided in Simulation Management module.



SpaceFOM

- Configurable set of multi-phase initialization phases.



Simulating Over Time

RPR FOM

- Mainly real-time execution with host time synchronized using NTP.
- Adapted versions are sometimes used for Monte Carlo simulation with HLA Time Management.

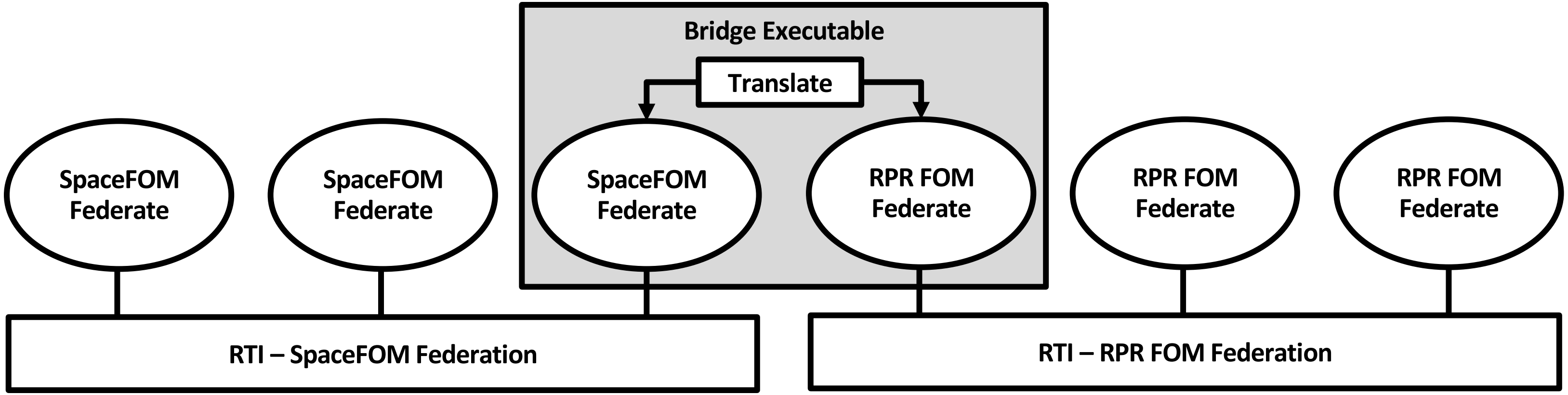
SpaceFOM

- No Pacing (as-fast-as-possible).
- Pacing (regulated by a clock).
  - Scaled Pacing
  - Real-time Pacing
    - *Strict/Conservative Real-time (no frame overruns*
    - *Elastic Real-time (catch-up on overruns) with Limited Overruns*
    - *Elastic Real-time (catch-up on overruns) with Unlimited Overruns*
- Includes hard real-time.
- As-fast-as-possible is useful for Monte Carlo simulation.
- Real-time is useful for “man in the loop” or “hardware in the loop”.

RPR FOM vs SpaceFOM

Bridging RPR FOM and SpaceFOM

Bridging RPR FOM and SpaceFOM



- A bridge is typically <u>one</u> executable that can join <u>two</u> federations.
- A bridge translates data and coordinates management and synchronization services.

Some Solutions for a Bridge

- **Object Classes for entities can be mapped between corresponding RPR FOM/SpaceFOM classes, depending on type enumerator and Entity Type string.**
- **A bridge could translate position and orientation between an Earth centric rotational reference frame, like EarthMJ2000Eq, in the SpaceFOM federation and geocentric WGS 84 positions in the RPR FOM based federation.**
  - Another option is to publish the WGS 84-based reference frame in the SpaceFOM federation.
- **RPR FOM federates can be Late Joiners and only support the Run phase.**
- **The biggest challenge is time management, including the multitude of modes supported by the SpaceFOM. No complete solution known (at least to the author).**

Other Approaches

- Time management, initialization, and execution control features from the SpaceFOM could be added to the RPR FOM federation.
- The requirement for Time Management could be relaxed in the SpaceFOM federation. This would unfortunately result in a loss of many desired capabilities, like hard real-time and as-fast-as-possible execution, causality, and repeatability.
- A bridge could publish information from the SpaceFOM federation into the RPR FOM federation, but not the other way around.
- Finally, another solution is to not use a bridge and instead migrate existing RPR FOM federates to be SpaceFOM compliant, and only run a SpaceFOM federation. This would retain many key capabilities of the SpaceFOM.

RPR FOM vs SpaceFOM

Convergence/Alignment

Benefits of Convergence

- **Meet the emerging requirements from users such as the Space Forces in several nations**
- **Added capabilities for civilian space simulation**
- **Interoperability and reuse of models and systems between the defense and space community.**
  - Low Earth Orbit
  - Cis-lunar
- **Knowledge exchange and reuse**
- **Build a better long-term standards landscape**

Convergence Challenges

- **Technical content, structure and effort**
- **Legacy and investments**
  - Systems, tools, knowledge, workforce.
- **Dependencies**
  - RPR FOM is intended to mirror DIS. Limited wiggle room.
- **Communities**
  - RPR FOM and DIS communities are well aligned.
  - The Space FOM community focuses on their own user requirements.
- **Cultural**
  - "Not invented here"
- **More**

Early Technical Steps Towards Convergence

- **Two new features have been added to the RPR FOM 3 to better use HLA features and for some alignment with SpaceFOM.**
  - These are optional to use.
- **Time Representation compatible with HLA Time Management.**
  - 64 bit integer, strictly increasing, representing microseconds since epoch.
  - Key advantage is support for as-fast-as-possible execution.
  - DIS/RPR FOM 2 timestamp with “ticks past the hour” still available.
- **Initialization with Phases.**
  - Based on synchronized steps: join – create objects – intialize objects – run.
  - Key advantage is safe and scalable intialization across the federation.

Summary

- **There are several use cases where the defense and space communities would benefit from federating simulations.**
- **One approach is to create a bridge between RPR FOM and SpaceFOM federations.**
  - The biggest challenge lies in initialization, execution control and time management.
  - SpaceFOM and RPR FOM federates can still interoperate given that some limitations, or updates to existing federates, are acceptable.
- **RPR FOM 3 will add a few steps towards convergence.**
- **Some degree of long-term convergence between neighbouring standards is desirable.**

Further Reading

- **Standards**
  - SISO: “Real-time Platform Reference Federation Object Model 2.0”, SISO-STD-001, SISO, www.sisostds.org
  - IEEE: “IEEE 1278.1-2012 - Standard for Distributed Interactive Simulation - Application protocols”, www.ieee.org
  - SISO: “Reference for Enumerations for Simulation Interoperability”, SISO-REF-010-2021, SISO, www.sisostds.org
  - SISO: “Space Reference Federation Object Model (SpaceFOM)”, SISO-STD-018-00-2020, SISO, www.sisostds.org

- **Papers**
  - Björn Möller, Aaron Dubois, Patrice Le Leydour, René Verhage: “RPR FOM 2.0: A Federation Object Model for Defense Simulations”, Proceedings of 2014 Fall Simulation Interoperability Workshop, 14F-SIW-039, Simulation Interoperability Standards Organization, September 2014.
  - Björn Möller, Aaron Dubois, René Verhage: “An Update on RPR FOM 3”, Proceedings of 2020 Winter Simulation Innovation Workshop, 2020-SIW-028, Simulation Interoperability Standards Organization, February 2020
  - Björn Möller, Edwin Z. Crues, Daniel E. Dexter, Alberto Falcone, Alfredo Garro: “A First Look at the Upcoming SISO Space Reference FOM”, Proceedings of 2017 Simulation Innovation Workshop paper 2016-SIW-017
  - Edwin Z. Crues, Daniel E. Dexter, Alberto Falcone, Alfredo Garro, Björn Möller, “SpaceFOM: An Interoperability Standard for Space Systems Simulations”, Proceedings of the 2022 IEEE Aerospace Conference
  - Edwin Z. Crues, Daniel E. Dexter, Alberto Falcone, Alfredo Garro, Björn Möller : “Enabling Simulation Interoperability between international Standards in the Space Domain”, Proceedings of the IEEE/ACM 26th International Symposium on Distributed Simulation and Real Time Applications

SISO

Simulation Interoperability Standards Organization

"Simulation Interoperability & Reuse through Standards"

QUESTIONS