A First Look at the Upcoming SISO Space Reference FOM

Björn Möller
Pitch Technologies
Repslagaregatan 25
58222 Linköping, Sweden
+46 13 4705503
bjorn.moller@pitch.se

Edwin Z. Crues
Simulation and Graphics Branch (ER7)
Software, Robotics, and Simulation Division (ER)
NASA Johnson Space Center
2101 NASA Road 1, Houston, TX
edwin.z.crues@nasa.gov

Dan Dexter
Simulation and Graphics Branch (ER7)
Software, Robotics, and Simulation Division (ER)
NASA Johnson Space Center
2101 NASA Road 1, Houston, TX
daniel.e.dexter@nasa.gov

Alfredo Garro
University of Calabria
Department of Informatics, Modeling, Electronics and Systems Engineering (DIMES)
University of Calabria
Via P. Bucci 41C, 87036 Rende (CS), Italy
alfredo.garro@dimes.unical.it

Anton Skuratovskiy
RusBITech
Varshavskoye shosse 26
117105 Moscow, Russia
+7 495 648 0640
a.skuratovskiy@rusbitech.ru

Alexander Vankov
RusBITech
Varshavskoye shosse 26
117105 Moscow, Russia
+7 495 648 0640
a.vankov@rusbitech.ru

Keywords: Space Simulation, HLA, FOM, Interoperability Standards

ABSTRACT: *Spaceflight is difficult, dangerous and expensive; human spaceflight even more so. In order to mitigate some of the danger and expense, professionals in the space domain have relied, and continue to rely, on computer simulation. Simulation is used at every level including concept, design, analysis, construction, testing, training and ultimately flight. As space systems have grown more complex, new simulation technologies have been developed, adopted and applied. Distributed simulation is one those technologies. Distributed simulation provides a base technology for segmenting these complex space systems into smaller, and usually simpler, component systems or subsystems. This segmentation also supports the separation of responsibilities between participating organizations. This segmentation is particularly useful for complex space systems like the International Space Station (ISS), which is composed of many elements from many nations along with visiting vehicles from many nations. This is likely to be the case for future human space exploration activities.*

Over the years, a number of distributed simulations have been built within the space domain. While many use the High Level Architecture (HLA) to provide the infrastructure for interoperability, HLA without a Federation Object Model (FOM) is insufficient by itself to insure interoperability. As a result, the Simulation Interoperability Standards Organization (SISO) is developing a Space Reference FOM. The Space Reference FOM Product Development Group is composed of members from several countries. They contribute experiences from projects within NASA, ESA and other organizations and represent government, academia and industry.

The initial version of the Space Reference FOM is focusing on time and space and will provide the following: (i) a flexible positioning system using reference frames for arbitrary bodies in space, (ii) a naming conventions for well-known reference frames, (iii) definitions of common time scales, (iv) federation agreements for common types of time management with focus on time stepped simulation, and (v) support for physical entities, such as space vehicles and astronauts. The Space Reference FOM is expected to make collaboration politically, contractually and technically easier. It is also expected to make collaboration easier to manage and extend.

1. Introduction

Spaceflight is difficult, dangerous and expensive; human spaceflight even more so. In order to mitigate some of the danger and expense, professionals in the space domain have relied, and continue to rely, on computer simulation. Simulation is used at every level including concept, design, analysis, construction, testing, training and ultimately flight. As space systems have grown more complex, new simulation technologies have been developed, adopted and applied.

Simulation is used in the space domain for a wide range of purposes. When developing new concepts and evaluating potential missions, simulation is a powerful tool to understand opportunities, limitations and challenges. When engineering systems and subsystems, it provides early insights into requirements and pros and cons of different designs. For training purposes, it makes it possible to train staff when real training opportunities are scarce, when real equipment is expensive or to train dangerous scenarios and emergency operations in new situations.

Distributed simulation makes it possible to build even more powerful simulations by making simulation models within one or more organizations work together. Different systems (such as propulsion, launch pad, command and control, etc.) supporting a launch may be simulated by different groups across an organization. Different organizations that produce different space vehicles may test docking processes before the real hardware is developed. Commonly used models and tools can be separated out and reused.

To maximize the benefits of interoperability, we want to minimize the integration effort each time a new system is reused. Open standards offer a more efficient way to combine and reuse systems and tools in new configurations. They also offer a neutral ground that is easier to accept to many participants than proprietary interfaces. Open standard can also capture best-practices and help communicate them to new developers.

1.1. NASA Experiences from HLA federations

Any spacecraft visiting the International Space Station (ISS) needs to be certified for ISS proximity operations prior to flight. Simulation is a significant part of the certification process. NASA has developed several distributes simulations to support visiting vehicles and resupply missions to the ISS. The Space Station Training Facility (SSTF) uses a distributed simulation of the ISS and the HII-A Transfer Vehicle (HTV) to supports flight controller and crew training. This distributed simulation has components at NASA's Johnson Space Center in Houston, Texas, USA and at the Japanese Aerospace Exploration Agencies (JAXA) Tsukuba Space Center in Tsukuba, Japan [3]. This simulation is now being updated and expanded as part of NASA's Training Systems for the 21st century (TS21). This simulation will support the existing HTV capabilities and new commercial systems that will be visiting the ISS. A significant challenge for TS21 is the lack of a FOM standard. Currently, NASA has to negotiate individually with each organization providing a distributed simulation model of their visiting vehicle. A Space Reference FOM standard would significantly help in the negotiation process.

Looking beyond low Earth orbit, NASA developed the Integrated Mission Simulation (IMSim) to support the Constellation Program. The IMSim was a distributed simulation of the principal space systems elements that would comprise the space systems to explore the Moon and Mars. This work is being further developed in the NASA Exploration Systems Simulations (NExSyS). NExSyS is developing general space systems simulation components to enable rapid assessment of mission and vehicle concepts. The NExSyS team supports NASA's Evolvable Mars Campaign (EMC), the Simulation Exploration Experience (SEE)(explained below), and the Space Reference FOM Product Development Group.

1.2. ESA and Russian Experiences from HLA federations

Early experiments on using HLA technology for space simulation purposes took place in 1997-1998 between the Russian Gagarin Cosmonaut Training Center near Moscow and the European Space Research and Technology Center of the European Space Agency (ESA/ESTEC, Noordwijk, The Netherlands). The scenario implied rendezvous and docking of a European unmanned transport spacecraft (called ATV - Automated Transfer Vehicle) to the Russian segment of the ISS [9]. On the basis of success of this project, the first prototype of a distributed cross-Atlantic space simulator was later implemented between the ESA/ESTEC and NASA JSC [1]. A similar demonstration was then (in 2003) deployed between the ESTEC and Tsukuba space centre in Japan. Also in the late 90's, European aerospace industry started an international R&D project called EDISON [7]. In that project, a consortium led by Aerospatiale built a more complex experimental space-oriented HLA federation consisting of about 10 federates where both man-in-the-loop and hardware-in-the-loop scenarios were tested. In all the above activities, the applications were given the highest priority. Accordingly, the FOM was, to a large extent, "application driven" and re-built nearly from scratch for every new project.

The concept was finally converted into an operational product by Astrium Space Transportation (now part of Airbus Space & Defence). HLA software (from Pitch Technologies) was used to integrate the ATV simulators in Toulouse and Bremen with a large complex of Russian simulation facilities in Korolev (near Moscow) to support training of several ground controller teams on ATV-ISS proximity operations [10]. This federation was in use for about 5 years, throughout four consecutive launch programs of the ATV spacecraft. However, the developed system was designed like a "common" (non-HLA) distributed application, and many important advantages of HLA were not involved. E.g., some FOM classes were introduced to be used like ordinary network connections containing data in the same format which can be used in TCP or UDP links.

1.3. Smackdown and SEE

The Simulation Exploration Experience (SEE), originally started in 2011 as the "SISO Smackdown", is an annual college-level modeling and simulation challenge. The Simulation Interoperability Standards Organization (SISO) and the Society for Modeling & Simulation International (SCS) are sponsors. The purpose is to bring practical M&S experiences to students. It attracts student teams from all over the world. Teams are invited to participate in a space scenario where they can contribute their own simulations. It is based on the HLA 1516-2010 standard [6], together with a prototypical space FOM. The scenario is a lunar mission where students choose their own sub-scenario, for example lunar mining operations, asteroid protection or establishing a moon base (see [4, 5]).

2. Developing a Standard

2.1. Why a standard for the Space domain?

Space simulations have some specific requirements that a Space Reference FOM needs to meet. It needs to be able to exchange data about the physical space environment such as planets and planetary bodies. It needs to be able to exchange data about facilities and processes in the proximity of different planets and planetary bodies, like the Earth, Moon, Mars or something more remote. It needs to correctly handle scenario time as well as the advancement of scenario time in relationship to wall-clock time. Simulations may include lengthy missions where running faster than real-time execution is required.

There already exists a FOM, developed by the defense and security community within SISO, called the Real-time Platform Reference FOM (RPR FOM) [2, 8]. It captures the information model of the older DIS standard, which, among other things, contains one space related object class. However, after some analysis it was concluded that the RPR FOM did not meet the requirement of many space related federations, for (at least) three reasons:

1. The RPR FOM makes the assumption that all positions shall be given using a geocentric coordinate system. This way of specifying positions is implicit and cannot be changed. In space simulation, different simulations need to specify positions in coordinate systems related to different bodies (Earth, Sun, Moon, Mars, etc). This makes it computationally inconvenient and in many cases even impossible to use the RPR FOM.

2. The RPR FOM uses a real-time, best effort approach to time management. HLA Time Management isn't used and a non-standard time-stamping approach is used. This makes it difficult or impossible to build federations that guarantee consistency and repeatability.

3. The RPR FOM offers an extensive set of classes tailored for warfare simulation but very little targeted at the Space community.

2.2. The SISO standardization process

SISO is an organization that develops open international standards for simulation. It is based in the US but has members all over the world with the majority of the members in North America and Europe. SISO develops several types of standards, including balloted standards, i.e. standard where every modification and the final result is balloted by members under a set of strict rules. The development processes for such standards is called the Balloted Products Development and Support Process (BPDSP). It contains two phases for the life cycle of a standard: the development phase and the support phase. This process can be seen as iterative, since the support phase includes initiating the development of a new version of the standard. The development phase, where the Space Reference FOM standard is at the moment, consists of the following steps:

Product Nomination: Formulating the activity and getting the activity approved.

Product Development: This is where the development of the technical product takes place.

Product Balloting: During this activity, SISO members are invited to give comments on the content of the draft standard and vote for or against approval. After this step a draft standard may need to be revised and re-balloted.

Product Approval: The culminating step by the SISO Standards Activity Committee and Executive Committee.

The product support phase then consists of distribution, interpretation and periodic review.

2.3. Setting up a team

The SISO BPDSP provides a clear process for developing a standard. The real challenge is to build a team of volunteers that share the same vision, has the required skill set as well as time and other resources to contribute to the standards development. A number of informal discussions and meetings were held at SISO workshops 2012-2014, in conjunction with SEE events in order to build such a team.

A product nomination for the Space Reference FOM was developed and submitted in January 2015 by nine SISO members. The Product Nomination was formally approved by the Standards Activity Committee in February 2015. The official kick-off meeting was held in September 2015. By summer 2016 more than 25 meetings (mostly teleconferences) have been held by the Drafting Group of the new standard.

The Product Development Group consists of people from government agencies, industry and academia. It contains members with expertise in at least three key areas:

Space simulation development, with extensive backgrounds from decades of development and integration of simulations in the space domain from North America, Europe and to some degree Asia.

Interoperability standards, including HLA and FOM development.

Standards development, including leading the standards development and drafting standards according to SISO, IEEE and other processes.

3. Overview of HLA

The Space Reference FOM builds upon the High-Level Architecture (HLA). This is an IEEE and SISO standard for distributed simulation. The main purpose of HLA is to achieve interoperability between simulations, but reuse and composability are also important aspects.

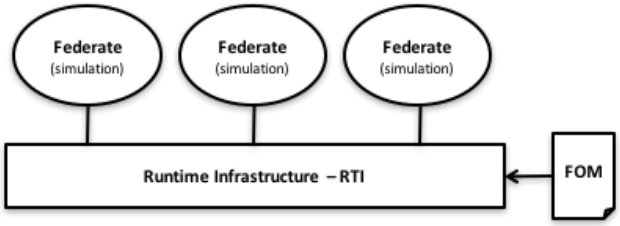

Figure 1: An HLA federation

Figure 1 shows the topology of an HLA federation, i.e. the topology when connecting systems using HLA. Each participating simulation is called a **federate**. All federates communicate using a **Run-Time Infrastructure** (RTI) that provides a number of services, like information exchange, synchronization and management.

HLA is not limited to any particular information domain or data models. Instead, data is exchanged according to a separate **Federation Object Model** (FOM), an XML document that describes what object classes, interactions and data types that are used. FOMs have been developed for various domain, like defence, medical and road transport. The federates, the RTI and the FOM together form a **Federation**. A simulation session is called a Federation Execution.

3.1. HLA Services

Some important services provided by the RTI include:

Information exchange using a publish-subscribe scheme. The RTI provides flexible information routing with loose coupling between federates.

Time Management for advancing the scenario time across a federation. This also guarantees that federates won't receive data with time stamps in the past, as seen from their current scenario time.

Synchronization points that enable federations to perform synchronized initialization and execution control.

Ownership Management that enables a federate to hand over the responsibility to simulate an entity to another federate.

Management services, using the Management Object Model, that for example gives management federates insight into which other federates that have currently joined.

Information filtering of large data sets (Data Distribution Management) for improved scalability in federations with large scenarios that have a large information flow.

3.2. Federation Agreements

When building a federation for a particular purpose, developers need to agree on a number of aspects. The most obvious aspect is what exact FOM to use. This specifies for which object classes and attributes that the federates shall exchange data. In many cases it is useful to specify the participating federates. It is also necessary to describe the execution flow, when to use which interactions, how to coordinate initialization and time advance, fault handling, and many other things

In many cases a standardized Reference FOM and federation agreement is used as a starting point. Project specific extensions are added as needed. The RPR FOM, mentioned above is the most commonly used Reference FOM.

4. Technical Content of the Space Reference FOM

The Space Reference FOM will consist of two parts:

The Space Reference FOM Federation Agreement. This document gives an overview and description of the FOM, and specifies rules for how it shall be used. This is a document intended for consumption by human readers.

The Space Reference FOM. This is a set of HLA FOM modules using XML format, that is intended for consumption by the HLA runtime infrastructure and other software tools.

The FOM modules provided are as follows and as shown in Figure 2.

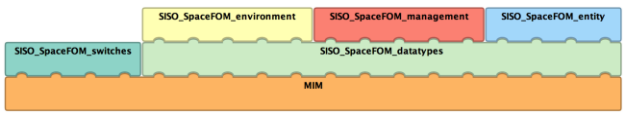

Figure 2: FOM Modules

Data types. A number of data types, based on the SI system ("metric") are defined here.

Environment. Defines concepts that are necessary to describe the physical space environment, mainly reference frames for bodies like the Sun, Earth and Moon.

Entity. Defines physical entities, like space vehicles.

Management. Describes objects, interactions and synchronization points that are used for managing a simulation execution.

Switches. Provides HLA Switches that are necessary for the execution.

MIM. This is a module that provides predefined HLA concepts, like standard data representations.

The FOM modules contain a relatively small but versatile set of Object Classes, as shown in Figure 3.

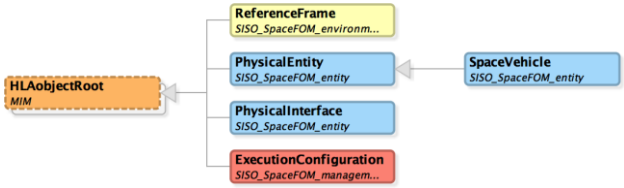

Figure 3: Object Classes

The classes are described in details below, according to the FOM modules where they are defined.

While the Space Reference FOM is not completed yet, the first version is expected to cover the areas below. The main focus can be seen as "time and space" since this forms the basis for any scenario in the space domain.

4.1. Federation Composition

A federation that complies with the Space Reference FOM standard is required to have federates that have certain predefined roles.

Master Role: A federate with this role is responsible for controlling the initialization of the federation as well as managing transitions between modes initialization, running, freeze and shutdown.

Pacing Role: A federate with this role is responsible for managing how the scenario time is advanced in relationship to wall-clock time.

Root Reference Frame Publisher Role: A federate with this role is responsible for registering the root of the reference frame tree. See next section for details.

In addition to the above roles, it is necessary to specify which federates are required versus optional during federation initialization, for a given federation execution.

4.2. Reference Frames

In order to describe the position and orientation of any physical entity, reference frames are used, for example a reference frame based on the inertial center of the moon. There may be several reference frames, where each reference frame has a translation (position) and orientation expressed with respect to a parent reference frame.

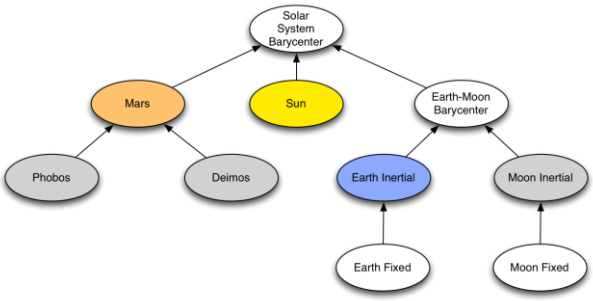

Figure 4: Reference Frame Tree

Reference frames thus form a directed acyclic graph. One example is shown in Figure 4. The root reference frame is an exception since it doesn't have a parent. Reference frames are explicitly described as HLA object instances.

This approach enables federates to simulate entities using the reference frame that is most computationally convenient to them at a particular time. It is still possible to convert the position of an entity in one reference frame into to a position in another reference frame, by traversing the tree. As an example, an entity in a lunar mission may

start by expressing its position in the Earth Inertial reference frame, then switch to the Earth-Moon Barycentric reference frame during the journey and, finally, land on the Moon using the Moon Inertial reference frame.

4.3. Well-known Reference Frames

Each federation execution is required to agree on the core set of reference frames to be used. To enable a higher degree of a priori interoperability, a set of standard reference frames are specified with focus on the solar system, including the Sun, Earth, Moon and Mars. Note that several of them are barycentric, i.e. based on the combined mass of several bodies, like EarthMoonBaryCentricInerital and SolarSystemBary-centricInertial.

There is also a naming scheme for specifying the name of a reference frame, which in turn is based on the International Astronomical Union Database, that contains a large number of bodies, like planets, dwarf and minor planets, satellites of planets, comets, stars, exoplanets, nebulaes, galaxies and other objects.

4.4. Initialization and Execution Control

The Space Reference FOM specifies an initialization process, as well as execution control for going into freeze mode and termination. The initialization process is particularly interesting. Each federate can be either an early joiner or a late joiner. The federate with the Master role manages the start-up of early joiners. The initialization process is shown in Figure 5.

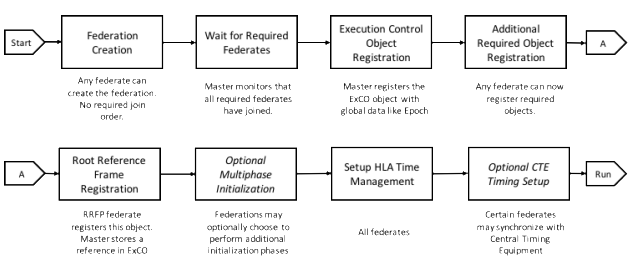

Figure 5: Initialization Process

It contains eight steps, out of which two are optional. The steps are:

1. Creation of the Federation Execution: Any federate can create the federation execution.

2. Waiting for Required Federates: This is monitored by the federate with the Master role.

3. Registration of the Execution Control Object (ExCO) Instance: This object instance contains key information about the execution state, the epoch (scenario starting time) and the root reference frame.

4. Registration of additional required object instances.

5. Registration/discovery of the root reference frame.

6. Optional multiphase initialization, as specified for each particular scenario.

7. Set-up of HLA Time Management. i.e. enabling time regulation and constrained mode of federates.

8. Optional set-up of timing for systems that have their timing based on Central Timing Equipment (CTE).

A federate that is not considered a required federate may also be a late joiner and join a federation that is already in the running mode.

4.5. Time Management

The Space Reference FOM mainly uses HLA Time Management to advance scenario time. The federate with the Pacing role is responsible for managing the advancement of scenario time. Note that the Execution Control Object specifies the Epoch of the federation, i.e. the starting time. This Epoch is specified in the Terrestrial Time (TT) scale and is expressed in Truncated Julian Date (TJD) format, referenced with respect to 1968-05-24 00:00:00 UTC. The HLA Logical Time and HLA service time stamps are interpreted as microseconds past the Epoch. All time managed federates advance their time with the Time Advance Request/Time Advance Grant HLA services.

Federates are required to use constant time steps. The standard defines three types of time steps, as illustrated in Figure 6.

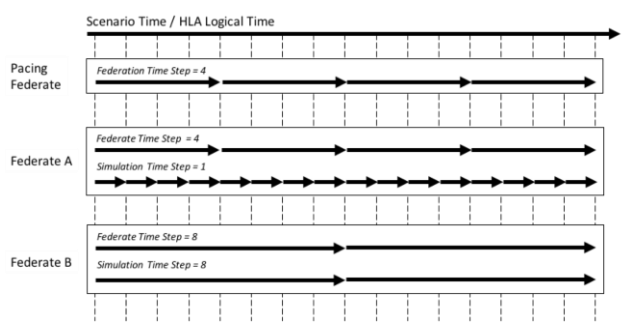

Figure 6: Different types of time steps

Simulation time step. This is the native time step by which a simulation model advances its state. This is sometimes known as the Dynamics rate.

Federate time step. This is the time step that a federate uses, when interacting in the federation. A federate may for example have a federate time step of 100 ms although

it contains a simulation model with simulation time step of 10 ms.

Federation time step. This is the federate time step of the pacing federate.

As can be understood from the figure, the federation, federate and simulation time step may be the same. In the case of Federate A, the simulation time step is smaller than the federate and federation time step. The federate time step is n times the simulation time step, where n needs to be an integer. In the case of Federate B, the simulation time step is larger than the federation time step. The federate time step is then equal to the simulation time step, which is n times the federation time step, where n needs to be an integer.

The Space Reference FOM also allows for a mix of HLA Time Management and federates using Central Timing Equipment. One of the challenges here is when the federation goes into freeze mode and the advancement of HLA logical time comes to a halt. The CTE (real time) time line now needs to be de-coupled from the HLA logical time. When resuming the advancement of HLA logical time the time lines are connected again.

4.6. Physical Entities

The Physical Entity and Space Vehicle class definition make it possible to exchange information about astronauts, man-made space vehicles and other entities. The position is described in relation to a parent reference frame. Physical entities also have properties like mass, inertia, velocity and acceleration. A particular class is the Physical Interface which can be used for representing docking ports, berthing interfaces, etc. It is expected that many Space Reference FOM applications will extend the classes of this module in order to exchange data about more specific types of equipment.

5. Discussion

The Space Reference FOM is developed based on tens of man-years of practical experience from simulation interoperability in the space domain. Simulations based on hundreds of man-years of development have been integrated. A large number practical experiences and a lot of engineering support has formed the input to the Space Reference FOM development.

In addition to this, the standard has been developed using a “test driven” approach. Early prototypes have been used in the SEE program. Sample federates, for example for simulating the environment (reference frames) have been developed. All design patterns in the full standard have been tested and verified using test federates as well as commercial HLA tools.

6. Conclusions

For the space simulation community, the Space Reference FOM is expected to make collaboration politically, contractually and technically easier. It is also expected to make collaboration easier to manage and extend.

For the simulation community in general, the Space Reference FOM also provides a number of reusable patterns for execution management and spatial positioning using a system of reference frames and execution synchronization.

For the HLA community in particular, the Space Reference FOM provides generally reusable designs for the use of the HLA Time Management services for building federations with time stepped federates, including late joiner.

For SISO, the Space Reference FOM is the first major step into the space simulation domain.

References

[1] L. Arguello, L. Dwedari, G. Lauderdale, A. Vankov, P. Chliaev: “ESA-NASA Distributed Simulation Experiment: First Results and Lessons Learned”, 2001 European Simulation Interoperability Workshop. Paper No. 01E-SIW-018, United Kingdom, 2001

[2] Björn Möller et al.: “RPR FOM 2.0: A Federation Object Model for Defense Simulations”, 2014 Fall Simulation Interoperability Workshop, (paper 14F-SIW-039), Orlando, FL, 2014.

[3] E. Z. Crues, M. Blum, V. Chung, J. Bowman: "The Distributed Space Exploration Simulation", Proceedings of the Spring 2007 Simulation Interoperability Workshop and Conference, Norfolk, Virginia, 25-30 March, 2007.

[4] A. Falcone, A. Garro, A. Anagnostou, N.R. Chaudhry, O.A. Salah, S.J.E. Taylor. Easing the development of HLA Federates: the HLA Development Kit and its exploitation in the SEE Project. Proceedings of the 19th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications (ACM/IEEE DS-RT), Chengdu, China, October 14-16, 2015.

[5] A. Falcone, A. Garro. The SEE HLA Starter Kit: enabling the rapid prototyping of HLA-based simulations for space exploration. Proceedings of the Symposium On Simulation for Planetary Space Exploration, part of the SCS SpringSim 2016 conference, Pasadena, CA (USA), April 03-06, 2016.

[6] D. Michel, D. Claude, F. Dubuc, P. Shlyaev, A. Vankov: “ATV System Validation Using a Distributed Simulation Facility”, 2000 Spring Simulation Interoperability Workshop. Paper No. 00S-SIW-009, Orlando, FL, 2000.

[7] 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.

[8] SISO: “Real-time Platform Reference Federation Object Model (RPR FOM) Version 2.0”, www.sisostds.org, September 2015.

[9] A. Vankov, P. Chliaev, A. Alyoshin, J. Miro: "Distributed Interactive Simulation of Rendezvous and Docking with International Space Station", 1997 Fall Simulation Interoperability Workshop. Paper 97F-SIW-005, Orlando, FL, 1997.

[10] A. Werkman, B. Sarter: "Integration of Spacecraft Training Simulators, the ATV and ISS *Rendez-vous* use case", 2011 European Simulation Interoperability Workshop. Paper No. 11E-SIW-004, The Netherlands, 2011.

Author Biographies

BJÖRN MÖLLER is the Vice President and co-founder of Pitch Technologies. He leads the development of Pitch’s products. 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 chair of the Space FOM Product Development group and the vice chair of the SISO HLA Evolved Product Development Group. He was recently the chair of the SISO RPR FOM Product Development Group.

EDWIN “ZACK” CRUES has over 25 years of professional experience in developing spacecraft simulation and simulation technologies. Zack is currently a member of the Simulation and Graphics branch at NASA’s Johnson Space Center in Houston, Texas where he leads the development of simulation technologies and the application of those technologies in the simulation of NASAs current and proposed crewed spacecraft. He has developed hundreds of models and simulations for NASA spacecraft including Shuttle, International Space Station (ISS), Orion, Altair, Morpheus and the Multi-Mission Space Exploration Vehicle. Zack’s recent research focus has been developing and applying distributed computation and distributed simulation technologies. This includes a large-scale distributed simulation of NASAs proposed human space exploration missions. Zack also has international experience in developing simulations of European Space Agency launch systems and Japanese Aerospace Exploration Agency spacecraft.

DAN DEXTER is an engineer in the Simulation & Graphics Branch in the Software, Robotics and Simulation Division of the Engineering Directorate at NASA’s Johnson Space Center in Houston, Texas. He has over 22 years of software and simulation development experience ranging from nonlinear signal and image processing, distributed supercomputing, and flight related software to national and international distributed simulations. He is the principal developer of the TrickHLA software package, a NASA developed middleware software package for using the HLA

distributed simulation standard with NASA standard M&S tools.

ALFREDO GARRO is an Associate Professor of Computer and Systems Engineering at the Department of Informatics, Modeling, Electronics and Systems Engineering (DIMES) of the University of Calabria (Italy). He is currently Visiting Professor (from January to October 2016) at NASA Johnson Space Center (JSC), working with the Software, Robotics, and Simulation Division (ER). His main research interests include: Modeling and Simulation, Systems and Software Engineering, Reliability Engineering. His list of publications contains about 100 papers published in international journals, books and proceedings of international and national conferences. He is vice chair of the SISO Space Reference FOM Product Development Group. He is the Technical Director of the "Italian Chapter" of INCOSE.

ANTON SKURATOVSKIY is a senior software engineer with RusBITech. After his 10-year service in the Air Force, he worked for D-3-Group and GTI6 companies since 1999 participating in research activities focused on using distributed simulation technologies in aerospace applications including support to ATV-ISS simulation and ground controller training. Currently at RusBITech he is working on both HLA and DDS middleware.

ALEXANDER VANKOV is a principal system analyst with RusBITech. Since 1980 he worked at the Gagarin Cosmonaut Training center of Russia for over 18 years and was involved in all major Russian manned aerospace programs. He then continued his research activities in distributed simulation at D-3-Group and GTI6. He participated in several large international projects in the area of distributed interactive simulation and operator training. His current research activities are focused on simulation and middleware technologies including management of HLA and DDS product lines development.