



**HAL**  
open science

## Using High Level Architecture in the SEE Project for industrial context

Simon Gorecki, Gregory Zacharewicz, Nicolas Perry

► **To cite this version:**

Simon Gorecki, Gregory Zacharewicz, Nicolas Perry. Using High Level Architecture in the SEE Project for industrial context. Service Orientation in Holonic and Multi-Agent Manufacturing, 2018, 10.1007/978-3-319-73751-5\_21 . hal-01809557

**HAL Id: hal-01809557**

**<https://hal.science/hal-01809557>**

Submitted on 6 Jun 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Using High Level Architecture in the SEE Project for industrial context

Simon Gorecki<sup>1</sup>, Gregory Zacharewicz<sup>1</sup>, Nicolas Perry<sup>2</sup>

<sup>1</sup>IMS UMR CNRS 5218, University of Bordeaux, France  
{gregory.zacharewicz, simon.gorecki}@u-bordeaux.fr

<sup>2</sup>Arts et métiers, ParisTech, ENSAM of Bordeaux, France  
nicolas.perry@ensam.eu

**Abstract.** Nowadays, systems are becoming more and more complex. Therefore, Modeling & Simulation (M&S) are also growing in complexity. That's why it is important to form, train and graduate students and faculty. The National Aeronautics and Space Administration (NASA) host an international event of M&S oriented in, solving interoperability problems, Distributed Simulation (DS), and student cooperation. The goal is to use a DS standard: High Level Architecture (HLA) to simulate a moon base. Each team build a module of it and have to communicate with each other. This context will be propitious for student to learn HLA programming, in order to reuse it in an industrial manufacturing context. The use of HLA is interesting in order to couple already existent heterogeneous works.

**Keywords.** High Level Architecture, Mobile Factory

## 1. Introduction

Nowadays modeling & Simulation (M&S) becomes increasingly important. They allow to represent any complex system, to study their functioning and their interactions with the environment. The simulation of a system will allow us to virtually design our subject to anticipate and avoid problems. With the growing technologies, the systems complexity is increasing and they became more and more complicate to simulate them. This makes appear distributed simulation (DS): one simulation divided multiple sub functions that are executed on separated computers. That solution allows to solve complexity problems, but it also raise interoperability problematics. There is a growing need to educate, train, and certify students, researchers, and faculty for M&S, DS and interoperability problematics [1][2]. Zack

Crues, a National Aeronautics and Space Administration (NASA) engineer has suggested to the Simulation Interoperability Standards Organization (SISO) Space Community Forum, launch an inter-university event which would learn to students about the importance of simulation interoperability and simulation. The SEE Project.

This project will serve as a springboard to study HLA mechanism in order to be reused in industrial domain. A French company has launched an innovative project for setting up a solar power plant. This project deals with different domains including risks. University of Bordeaux supports research in M&S and specifically in DS (Distributed Simulation). Most of these research works have created specific domain simulators. Each of these autonomous simulations is capable of representing a fragment of the global project. One of the last phases of this project is to assemble all these simulations to obtain a global simulation of the problem. [3] However, all these simulations use different technologies and manipulate heterogeneous data, which complicates the assembly. To solve these problems, we will use the HLA standard learned in the SEE project.

## ***2. Distributed Simulation & High Level Architecture***

In the computer simulation domain, distributed simulations are one of the most useful and powerful application. Indeed, they consist of several components (often associated with one or more functions) that can be processed by different processes. All these components are part of a single execution which can be relocated to a different computer / server, hence the term "distributed". This concept of functions relocation makes possible the loads distribute on different machines and thus increase the efficiency of a program.

One of the advantages of distributed simulation is to solve some interoperability problems. Interoperability is the ability of interactions between systems. This problematic appears when several systems highly dissimilar (by their internal structure, exchanged data format, or semantic data) must communicate. The interoperability problematic must be considered if interactions are at data level, service level or process level [4]. Those problematics involves at least two entities which try to communicate. Consequently, establishing interoperability means to relate two systems together and remove incompatibilities. Incompatibility is the fundamental concept of interoperability. The concept 'incompatibility' has a broad sense and is not only limited to 'technical' aspect as usually considered in software engineering, but also 'information' and 'organisation' [4].

Indeed, in distributed simulations the components are modular, they can have a heterogeneous architecture and exchange different structured messages, this allows to solve interoperability problems.

In our application case, this notion of distributed simulation will be used with the High Level Architecture standard (HLA) [5]. It is a specification of software

architecture. It defines a framework that allows to create global execution, consisting of distributed simulations. This framework defines how to create a "global" simulation consisting of several distributed simulations that can communicate with one another. It was originally created by the Office of Defense Modeling and Simulation (DMSO) of US Department of Defense (DoD) to facilitate the assembly of stand-alone simulations with a different architecture. The original goal was the reuse and interoperability of military applications, simulations and sensor. This standard is designed to resolve interoperability and reusability issues between software components.

HLA, is an architecture that enables several simulation systems to work together. This is called interoperability, a term that covers more than just sending and receiving data. The systems need to work together in such a way that they can achieve an overarching goal by exchanging services. Interoperability refers to the ability of interactions between systems. [6] It extends beyond the boundaries of any single system, and involves at least two entities. Consequently, establishing interoperability means to relate two systems together and remove incompatibilities.

Another interesting aspect of this technology is synchronization. It enables to dynamically manage interoperability issues. Simulations exchange messages, it must be ensured that messages are sent at the right time, in the right order, and that they do not violate causal constraints. To do this, various systems for synchronization of processes and time management are proposed by HLA.

According to the HLA standard, each simulation participating to the application is called "federate". A federate interacts with other federates there are forming a group named HLA federation. All of these entities can communicate with each other through a Run-Time Infrastructure (RTI). It is the RTI that will manage the federation, authorize federates to communicate or not, and provide various services such as time management, file or data exchange, etc.

### **3. Agent based approach**

In simulation, agent based simulation is one of the most popular approach for simulating systems. It is based on studying actions and interactions between autonomous components (agents). High Level Architecture (HLA) is based on Distributed Simulation, which is not dedicated to agent model. However, there are already in the past some approaches to use HLA in order to developed agent based model [7].

#### **4. Run-Time Infrastructure (RTI)**

A federation is composed of a set of federates and a Run-Time Infrastructure [2]. This RTI provide to federate all functionalities that are described by the specification. Federates can only interact through the RTI. By using HLA and throw the RTI, all federate are stand-alone simulations. That allow them to keep their own architecture which can be totally different to the others federates. This principle enables to solve interoperability problematics. By keeping they own architecture, and using a pivot-language, federates can be plug-in and plug-out to a federation that enable reusability. Federates can “Publish” to inform RTI and other federates about an intention to send information. They also can “Subscribe” to reflect some information created and updated by other federates. This is the basics communication mechanic. The data flow exchanged between all the federates are represented in the same form of classical object-oriented programming. There are two kind of object who are exchanged in HLA standard: Object Class and Interaction Class.

Object Class are time persistent during the simulation. They have attributes that can be updated. For example, in a simulation, an Object Class would be a car and his attribute would be a position, a speed, a name. Interaction are not persistent over time and can have parameters. For example, “Start car”, “Stop car”, “Accident” would be Interaction example.

This two kind of object are described by a XML file named Federation Object Model (FOM) attached to a federate. That’s an important point: the FOM describe all the information that will be exchanged between federates. That the only think who will be shared between simulations. This has an impact on safety of this technology. The federate are totally autonomous, the only exchange between us is described by the XML file.

Each federate are single application which can be executed separately. Originally, HLA was created for reusability, all the component can be executed separately, or with others through a federation. Overall, HAL and the RTI provide information transport, federates time synchronization, does not violate causal rules.

#### **5. SEE Project**

The Simulation Exploration Experience (SEE) is a project initiated by Zack Crues, a National Aeronautics and Space Administration (NASA) engineer. He suggested that the Simulation Interoperability Standards Organization (SISO) Space Community Forum launch an inter-university event which would learn to students about the importance of simulation interoperability by doing it and, become more employable and job-ready than the average college graduate. Called ‘Smack-down’[1], this project is supported by industry, academia and government. The first event las taken place in 2011 and since, it is renewed every year. In 2017, about 20 teams around the world has participated to the project like Massachusetts Institute

of Technology (MIT), Florida Institute of Technology (FIT), Carolina State Colleges (interns at Johnson Space Center), University of Bordeaux (France), University of Genoa (Italy), University of Calabria (Italy), and many others.

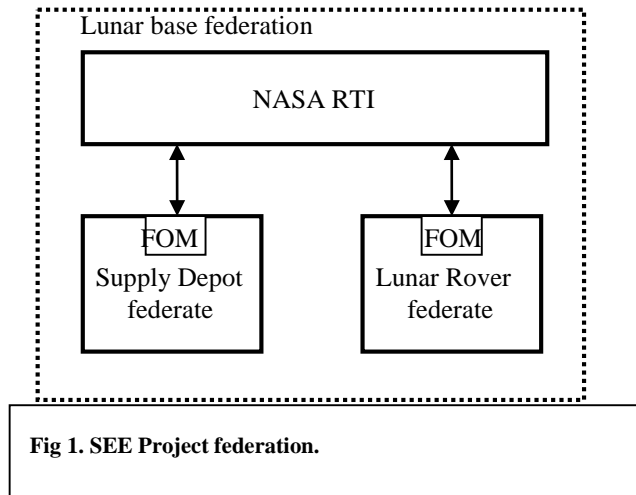
The SEE Project meet industries, students, teachers and professional associations in a modeling and simulation (M&S) challenge. The goal is to run simulation of a virtual moon base where each team use the last release of HLA (1516e), and industry software, in order to create his own module. By using the HLA standard, the NASA provide a RTI and several federates that simulate a lunar environment. The RTI and all the federates are forming a lunar base federation. Each team has worked for month with the technical team, tutorials, and also together throw a forum in order to create or upgrade his own federate.

MÄK and Pitch Technologies provide to student evolved RTI software and his associated tutorials [8]. NASA provide also serval software like “HLA Starter kit” which simplify the HLA development [2]. It allows team to be more focused on data exchange between their federate by reducing the development complexity. NASA provide also Distributed Observer Network (DON3) that use a video game graphical engine in order to simulate the scenario in a 3D environment. This one is a special federate which permit to associate a 3D object model to each federate. By sharing throw FOM x, y and z coordinate, that allow to move 3D models in a virtual space. As the event is international, it is necessary to use remote connection to associate federates with RTI and VPN also provided by NASA.

Using all this different tools, the 2017 SEE Project enabled student from Liverpool to create and simulate an Asteroid harvest fleet allowing to mining resources. This team also improved his lunar orbit asteroid. University of Munich build a base which houses observatories and can serve as a means of exploration for the needs of lunar resources. The University of Calabria and Genoa created the Moon City Center (MCC) with interconnected modules provided by Orange Technical College, FACES, Sophia, Liverpool University as astronaut habitat, a standalone residence designed for long term comfort, a greenhouse to supply fresh vegetables for astronauts, a spacecraft Launchpad ready to launch a rocket to asteroid belt. And finally, the Florida Institute of Technology (FIT) created an exploration rover linked to the Bordeaux’s Supply Depot. Both of them can communicate in two ways described in next section.

## 6. Application

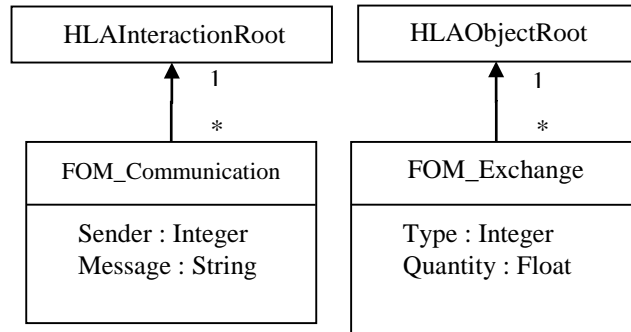
As Bordeaux team, we worked on the Supply Depot Federate and particular on the communication with the FIT’s rover as described in the Fig. 1.



Our objective was to communicate by using two types of Object that used for communicate with HLA: Object Class and Interaction Class. For that, we had to find two types of exchange: one that needs a persistent class that will be used with an Object Class, and another with an ephemeral notion, used with an Interaction Class.

The ephemeral interaction (mechanism who will use Interaction Class) will be the radio communication between the two federates. Each message will instantiate an Interaction Class at his reception. After message reception, the federate will save and treat it. The class will destroy itself. For the persistent interaction (mechanism who will use Object Class) will be the power exchange from the Supply Depot to the rover. At the federate initialization, this class will be created and when a power exchange will be needed the two federates will use this class for energy exchange. In our process algorithm, for trigger a power exchange (when the rover is low battery), the rover has to be near the supply depot and send to it an energy request message.

The Federation Object Model as the role of describe the two kind of object that we just describe. This specification is necessary for ensuring communication in HLA environment.



**Table 2.** Lunar Supply Depot Federation Object Model (FOM)

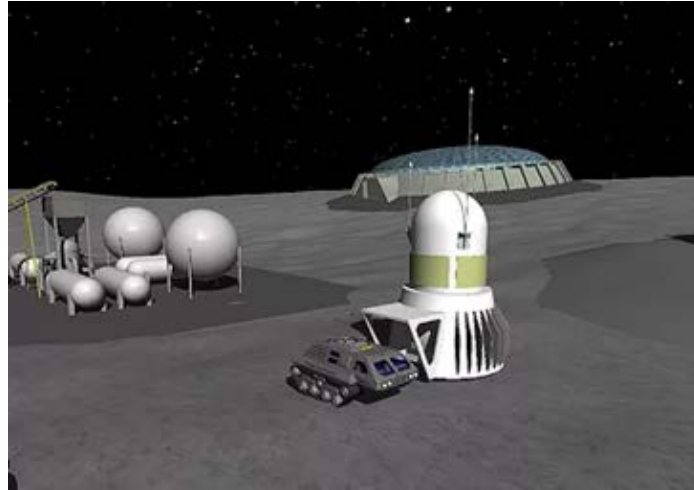
We can see on the above UML Diagram (Fig 2.) the lunar supply depot Federation Object Model representation. As an object-oriented programming language, the "communication" class (left) inherit from the Interaction Class (HLAInteractionRoot), and the "power exchange" class (right) inherit from the Object Class (HLAObjectRoot). The parameters and attributes are specified below the class name. The diagram represents the description of the pivot format that enable HLA interoperability. Indeed, in our case, we are trying to establishing communication between two simulations (supply depot and rover) which are totally different. They haven't got the same structure but, by using the FOM technology, they will be able to communicate throw HLA.

From a technical point of view, federates are developed in java on Eclipse IDE. HLA concept are implemented throw several frameworks. The HLA library provided by Pitch allow to implement HLA functionalities. And the NASA HLA Starter kit is an overlay to simplify the HLA development.

For this project, the Bordeaux team has to build a 3D object model in order to be represented by the graphical engine (Distributed Observer Network). For this work, we used the Industrial design software SolidWorks with an additional plugin to export it in geometry definition file format (.obj).

We can see on the picture (Fig 3) a simulation screenshot that takes places in a past event with a supply depot having interaction with a rover. The visual render was made with DON3, a graphical engine developed by NASA.





**Table 3. SEE Project Simulation screenshot**

We can see on the above schema, the UML communication diagram sequences representing communication between the two federates, and the RTI. Supply Depot and Rover are designed for communication. Bordeaux team and FIT team had work together for establishing the scenario below.

In our case, the communication by messages allow to exchange text messages. Those messages aren't persistent over the time, so they will be set as Interaction class. One the message received, the object will be destroyed. His main function during the simulation is to test if federates are operational. Contrary to the other communication mean, transfer energy between the depot and the rover will be persistent over the simulation. It will be defined as Object class.

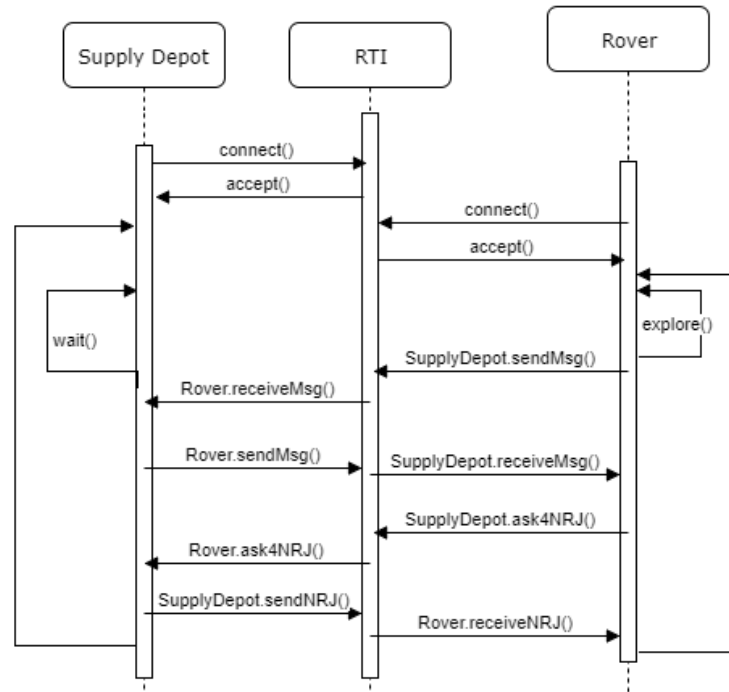


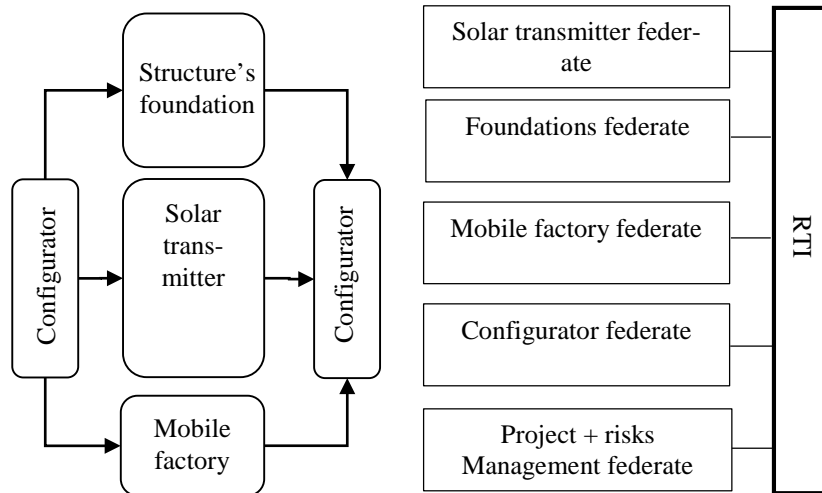
Fig. 4. UML Communication diagram sequence

## 7. Industrial application

Once the SEE project completed, we will be able to apply HLA knowledges on an industrial context. This context is an franco-morocan company who want to design, develop and control a solar power plant. This project consists in installing solar panels fields in several countries in order to provide electricity in world areas which are not powered so far. In terms of implementation and difficulties caused by the project complexity, this innovation gave rise to several simulations that solve several problematics:

- Simulation for the designing of solar transmitter supporting structure
- Simulation for defining the structure foundations of the solar panels field, depending on the ground structure.
- Study and dimensioning of the mobile factory dimensions, cost, etc, according to the demand
- Study of project management method integrating risks. Calculating risks probabilities into project management

However, those simulations were made by several persons, not in the same time, and with different languages. So, they aren't compatible with each other. That's why we will use the HLA in order to solve interoperability issues and assemble them in order to create a global simulation composed of each one of those simulations.



**Table 5. Global simulation as BPMN and HLA view**

We can see on the left diagram above, a representation of all simulations process workflow needed. The three main simulations take and store data on a configurator which act as database and GUI. The risks management project isn't considered in this workflow because it act separately. On the right diagram above, we can see what could be the project architecture by using HLA technology. In that configuration, each simulation could be a federate connected to the RTI. The configurator federate acting as data center of the federation, and control center in order to drive scenarios.

## 8. Conclusion

The works developed in the frame of 2017 SEE Project has been a very instructive adventure at many levels for students. It has permitted to develop distributed simulation components. Also on social aspect, it has allowed to expose oneself to problems of interoperability through the simulation of lunar modules. It is also a very good exercise to understand and master the simulations distributed through the HLA specification. It also makes it possible to use widely used technologies such as Virtual Private Network (VPN) in order to communicate between each federated and the RTI hosted on NASA servers, or industrial design software like SolidWorks.

Here the SEE project allowed us to study and understand the HLA mechanisms in order to apply it in an industrial context. The state of the art revealed that distributed simulation and the HLA standard can give an interesting answer to couple these heterogeneous works. The future step will consist in defining, according to HLA Federation Development Process (FEDEP), the behavior of each federate to assemble each simulation together.

**Acknowledgments.** From a general point of view, this project brings much experience at the human level. Indeed, it has allowed us to work with major universities around the world. For that we would like to thank you NASA's engineers for their support and the developers of the SEE Project providing all participants with a memorable, interactive, problem-solving experience which can contribute importantly to the workforce of the future.

## References

1. Elfrey P. R., Zacharewicz G., Ni M., (2011). SMACKDOWN: adventures in simulation standards and interoperability. Winter Simulation Conference, 3963-3967.
2. Falcone A., Garro A., Anagnostou A., Chaudhry NR., Salah O., Taylor SJE, (2015). Easing the Development of HLA Federates: The HLA Development Kit and Its Exploitation in the SEE Project. Distributed Simulation and Real Time Applications (DS-RT), 50-57.
3. Gorecki S., Zacharewicz G., Perry N., (2017). Using High Level Architecture to combine simulations in company context mobile factory. International Multidisciplinary Modeling & Simulation Multiconference.
4. Zacharewicz G., Labarthe O., Chen D., Vallespir B., (2009). Multi Agent/HLA Enterprise Interoperability (Short-Lived Ontology Based). The International Workshop on Modelling & Applied Simulation, Puerto de la Cruz, Spain. 187-196.
5. IEEE Computer Society, (2010). IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Federate Interface Specification. IEEE Computer Society, USA. Available from standards.ieee.org [accessed 1 June 2017]
6. Zacharewicz G., Chen D., Vallespir B., (2009). Short-Lived Ontology Approach for Agent/HLA Federated Enterprise Interoperability. Interoperability for Enterprise Software and Applications, 329-335.
7. Lees M., Logan B., (2007). Distributed Simulation of Agent-Based Systems with HLA. ACM Transactions on mOdeling and computer Simulation TOMACS, 881-888.
8. Möller B., Eriksson S., Wihlborg A., (2012). Developing an HLA Tutorial: Philosophy and Best Practices Proceedings of 2012 Fall Simulation Interoperability Workshop Simulation Interoperability Standards Organization, 12F-SIW-047.