

# Development of an OMT Table Viewer/Editor Using the Matlab/Simulink for HLA-Based Distributed Simulation

Younghun You, Tan Lee, Wooshik Kim, and Sugjoon Yoon

**Abstract**—Currently, simulation has become an essential part of design, acquisition, manufacturing, and maintenance. Furthermore, as systems are becoming large, complex, and distributed, computational demand on simulation becomes large and distributed. Matlab is a widely-used, powerful tool and becomes a standard in the computational part of computationally intensive simulations. In distributed simulations, one of the most well-known and widely-used standard is HLA (High Level Architecture). HLA is a standard for distributed simulation that is defined in IEEE 1516 and SIS0. In distributed environments, data exchange between the heterogeneous entities is important. However, OMT (Object Model Template) table, which is assumed to follow the HLA standard, and Matlab/Simulink are not compatible with each other. In this paper, we develop a tool for creating, editing, importing, and exporting OMT tables on Matlab/Simulink. Since the format of xml is a standard for HLA, we also use xml for our standard format for importing and exporting for viewer / editor. We expect that the result here will be very helpful for the integration of HLA and Matlab/Simulink.

**Index Terms**—Matlab, OMT table, HLA, IEEE 1516.

## I. INTRODUCTION

With the development of computer technology, the importance of simulation in various fields has been rise. In addition, since the simulations are becoming complex and large, distributed simulation environment has been needed.

Distributed simulation function is emulator running on different computers which are independent, and they are performed while exchanging data with each other. In addition, the emulator can be a different type. Therefore, the most important thing of distributed simulation is data exchange. So, IEEE 1516 HLA (High Level Architecture) standard has been adopted for the interworking of distributed simulation between these other models.

Matlab is a widely-used, powerful tool and becomes a standard in the engineering, basic science, numerical, and simulation. Besides, it is the most popular tool, numerical analysis, modelling and visualization, which has the advantage of being able to perform various simulations.

Therefore, the future, Matlab and distributed simulations

are integration which is essential. So, in this paper, as part of integration of the Matlab and distributed simulation, we have developed the OMT (Object Model Template) table tool using the Matlab/Simulink for creating, editing, importing, and exporting data. Since, the data format is xml for HLA standard. We also use xml data format for importing and exporting for OMT table viewer / editor.

This paper is composed as follows. In Section II, we introduce the OMT Table and IEEE 1516 HLA standard. In Section III, we described several concepts for HLA standard. And, in Section IV, we will show the results of the simulation execution. Finally, in Section V, we conclude in this paper.

## II. IEEE 1516 HLA STANDARD AND OMT TABLE

### A. IEEE 1516 HLA Standard

IEEE 1516 HLA standard is focused on the reuse and interoperability of the simulation model in a simulation environment of HLA (High Level Architecture) structure. In addition, it specifies what to do data exchange in a distributed simulation environment [1].

This standard is either in a four-part, in the IEEE 1516.1 standard defines an interface between federates (simulation, system). Also, HLA interface specification rules are provided. Detailed definition of the interface shows in reference [2]. In the IEEE 1516.2 standard defines the syntax and data format for HLA object model. Also, this standard provides OMT (Object Model Template), is a necessary form for documentation standard describing the data used by each object model. Detailed rules reference [3]. IEEE 1516.3 standard defines FEDEP (Federation Development and Execution Process) and FEDEP concept is described. Detailed information reference [4]. Finally, IEEE 1516.4 standard defines the processes that should be followed to implement verification, validation for federations being developed using the HLA FEDEP. Detailed information reference [5].

### B. OMT Table

OMT Table is provided by IEEE 1516.2. The purpose of this table is to define data format and syntax for information in HLA object models, to include objects, parameters, interactions, and attributes. Because HLA has been developed to provide a common architecture for simulation, HLA object models are defined by common interface architecture. So, OMT table is provided. Also, HLA structure focuses on the reuse and interoperability of the simulation. Therefore, by applying a common interface via the OMT table, it makes possible to exchange data between the simulations that each made a different. This table is composed of various tables. For

Manuscript received December 15, 2014; revised November 19, 2015.

Younghun You, Tan Lee, and Wooshik Kim are with the Department of Information & Communication Engineering, Sejong University, Seoul 143-747, Korea (e-mail: ryh0824@naver.com, leetan7848@naver.com, wskim@sejong.ac.kr).

Sugjoon Yoon is with Department of Aerospace Engineering, Sejong University, Seoul 143-747, Korea (e-mail: sjyoon@sejong.ac.kr).

example, object model identification table, Object class structure table, Interaction class structure table, and etc. Figures below are examples of some tables.

Category	Information
Name	<name>
Type	<type>
Version	<version>
Modification Date	<date>
Security Classification	<security classification>
Release Restriction	[<release restriction>]
Purpose	[<purpose>]
Application Domain	[<application domain>]
Description	[<description>]
Use Limitation	[<limitation>]
Use History	[<history>]
Keyword	
Taxonomy	[<taxonomy>]
Keyword Value	[<keyword value>]
POC	
POC Type	<poc type>
POC Name	[<poc name>]
POC Organization	[<poc organization>]
POC Telephone	<poc telephone>
POC E-mail	<poc email>
Reference	
Type	[<ref type>]
Identification	[<identification>]
Other	<other>
Glyph	[<glyph>]
Type	<type>
Alt	[<alt>]
Height	[<height>]
Width	[<width>]

Fig. 1. Example for object model identification table.

HLAobjectRoot (<p/s>)	{<class><p/s>}	<class><p/s>	<class><p/s>	...	<class><p/s>
			<class><p/s>	...	<class><p/s>
			⋮	...	⋮
			<class><p/s>	...	<class><p/s>
			<class><p/s>	...	<class><p/s>
	{<class><p/s>}		<class><p/s>	...	<class><p/s>
			<class><p/s>	...	<class><p/s>
			⋮	...	⋮
			<class><p/s>	...	<class><p/s>
			<class><p/s>	...	<class><p/s>

Fig. 2. Example for object class structure table.

The Fig. 1 and Fig. 2 of detailed definitions, description and information refer to [3].

The information contained in these tables is transmitted to the data information between each simulation. So, the xml format is required for data. Therefore, the data is exchanged in xml format. Since, we developed OMT table Viewer/Editor using Matlab/Simulink. As an example, in Section IV, we show the result of simulation for xml format for data convert via Matlab/Simulink.

### III. SEVERAL CONCEPTS FOR HLA STANDARD

Described concepts are important factors that consist of the HLA standard.

#### A. Federation Rules

The HLA consists of a set of 10 HLA rules. The HLA rules are divided into two parts consisting of 5 rules for HLA federates and 5 rules for HLA federation. HLA federation rules as follows.

- Federations shall have a FOM, documented in accordance with the OMT.
  - All representation of objects in the FOM shall be in federates, not in the RTI.
  - During a federation execution, all exchange of FOM data among federates shall occur via the RTI.
  - During a federation execution, federates shall interact with the RTI in accordance with the HLA interface specification.
  - During a federation execution, an attribute of an instance of an object shall be owned by only one federate at any given time.
- HLA federates rules as follows.
- Federates shall have a SOM, documented in accordance with the OMT.
  - Federates shall be able to update and/or reflect any attributes of objects in their SOM, and send and/or receive SOM interactions externally, as specified in their SOM.
  - Federates shall be able to transfer and/or accept ownership of attributes dynamically during a federation execution, as specified in their SOM.
  - Federates shall be able to vary the conditions under which they provide updates of attributes of objects, as specified in their SOM.
  - Federates shall be able to manage local time in a way which will allow them to coordinate data exchange with other members of a federation.

#### B. Interface Specification

Interface Specification defines RTI (Runtime Infrastructure) services and interfaces and identifies “callback” functions each federate must provide.

RTI is software that provides common services to simulation systems. And implementation of the federate initiates services in accordance with the HLA Interface Specification. RTI services as follows.

- Support object declaration and management between federates.
- Assist with federation time management.
- Provide efficient communications to logical groups of federates.

Finally, OMT is important factor for configuration HLA standard, will be omitted because the description in the Section II.

RTI is described in section D determined with reference to the FOM for the interaction between federates and exchanges data.

The reason of this concept is described in order to explain that the data exchange is important to the HLA structure.

### IV. RESULT OF SIMULATION

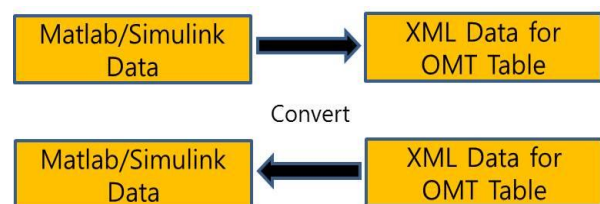


Fig. 3. Simple block diagram for the principle of OMT table viewer/editor.

The purpose of this study is a part of the integration for Matlab/Simulink and HLA-based distributed simulation. Also, we did this study to importing, exporting, editing and creating data format of the OMT table used in the HLA standard via Matlab/Simulink. Accordingly, this program shows that data from Matlab/Simulink converts xml data for OMT table used in HLA standard. This program was developed under the following environments.

- Windows 7 32-bit OS
- Matlab R2013b

And a simple block diagram in Fig. 3 shows that the principle of program.

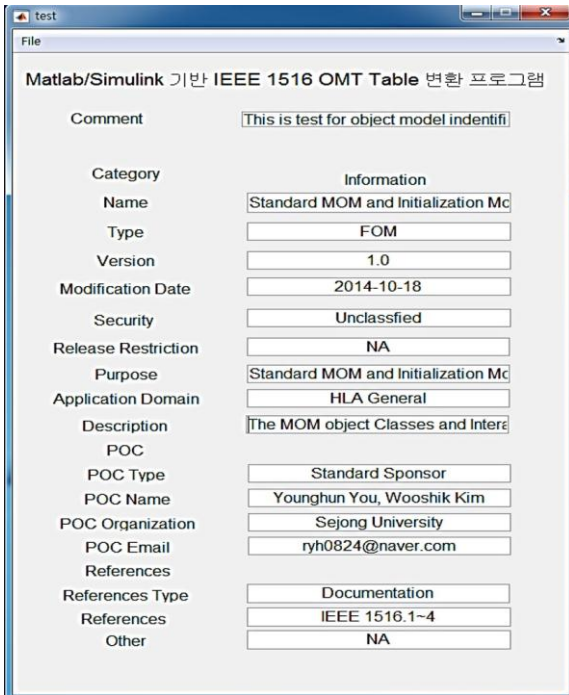


Fig. 4. The contents filled on the program for the simulation.

In order to determine whether the program is working properly, the following simulation was conducted. This simulation is the simulation for object model identification table corresponding to Fig. 1.

The following describes the functions and the process of simulation. First, a user fills the desired value in the program of object model identification table. Then, the user specifies the save path via the menu bar button at the top, and saves the file in xml format. If users want to erase the contents are filled by pressing the reset button on the menu bar. Also, if user wants to modify the contents of a file saved in xml format, using the open button on the menu bar loads the xml file.

So far, we have been talking about the simple operation of the OMT table Viewer/Editor program. The test results with the program as shown in the Fig. 5. Fig. 4 is the program's object model identification.

Fig. 5 shows xml file that the simulation result saved in the exported data. In addition, Fig. 6 shows xml file of HLAstandardMIM which is provided by IEEE 1516 standard.

The simulation result confirmed that syntax structure that we have created an xml file and xml file which is provided by IEEE 1516 that is equal.

## V. CONCLUSION

In this paper, we developed IEEE 1516 OMT Table Viewer/Editor program using Matlab/Simulink for a part of integration of Matlab/Simulink with HLA-based distributed simulation environment. So, we use the simulation of the developed program, the result are shown. As a result, it is possible to make the OMT Table in the XML file in the same syntax structure, is provided by HLA standard.

We expect that this result useful to integration of HLA and Matlab/Simulink. And it seems to be helpful a variety of study for the integration of Matlab/Simulink and the HLA standard.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--This is test for object model identification-->
- <objectModel xsi:schemaLocation="http://standards.ieee.org/IEEE1516-2010
http://standards.ieee.org/downloads/1516/1516.2-2010/IEEE1516-DIF-2010.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://standards.ieee.org/IEEE1516-2010">
- <modelIdentification>
  <name>Standard MOM and Initialization Module(MIM) for HLA IEEE 1516-2010</name>
  <type>FOM</type>
  <version>1.0</version>
  <modificationDate>2014-10-18</modificationDate>
  <securityClassification>Unclassified</securityClassification>
  <releaseRestriction>NA</releaseRestriction>
  <purpose>Standard MOM and Initiallization Module for HLA IEEE 1516-2010</purpose>
  <applicationDomain>HLA General</applicationDomain>
  <description>The MOM Object Classes and Interaction Classes of this object model may be
  extended</description>
- <poc>
  <pocType>Standard Sponsor</pocType>
  <pocOrg>Sejong University</pocOrg>
  <pocName>Younghun You, Wooshik Kim</pocName>
  <pocEmail>ryh0824@naver.com</pocEmail>
</poc>
- <references>
  <referencesType>Documentation</referencesType>
  <referencesId>IEEE 1516.1~4</referencesId>
</references>
  <other>NA</other>
</modelIdentification>
</objectModel>
```

Fig. 5. XML file from the simulation result.

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- The IEEE hereby grants a general, royalty-free license to copy, distribute, -->
<!-- display and make derivative works from this material, for all purposes, -->
<!-- provided that any use of the material contains the following -->
<!-- attribution: "Reprinted with permission from IEEE 1516.1(TM)-2010". -->
<!-- Should you require additional information, contact the Manager, Standards -->
<!-- Intellectual Property, IEEE Standards Association (stds-ipr@ieee.org). -->
<objectModel xsi:schemaLocation="http://standards.ieee.org/IEEE1516-2010 http://standards.ieee.org/downloads/1516/1516.2-2010/IEEE1516-DIF-2010.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://standards.ieee.org/IEEE1516-2010">
- <modelIdentification>
  <name>Standard MOM and Initialization Module (MIM) for HLA IEEE 1516-2010</name>
  <type>FOM</type>
  <version>1.0</version>
  <modificationDate>2010-08-16</modificationDate>
  <securityClassification>Unclassified</securityClassification>
  <purpose>Standard MOM and Initialization Module for HLA IEEE 1516-2010</purpose>
  <applicationDomain>HLA General</applicationDomain>
  <description>The MOM Object Classes and Interaction Classes of this object model may be extended.</description>
- <poc>
  <pocType>Standards Sponsor</pocType>
  <pocName>Simulation Interoperability Standards Organization</pocName>
  <pocOrg>SISO</pocOrg>
  <pocTelephone>+1 (407) 882-1348</pocTelephone>
  <pocEmail>info@sisostds.org</pocEmail>
</poc>
  <glyph type="GIF" alt="MOM and Initialization Module">
    R0lGODlhIAAgAMQAAAPeUHYFNdwAAAP3ozYRPD61oFmp5GJNYEfmz/xZs/vIi/ih0OWJG0InCLVtFfmpStSFJIEUBFIxCTaAGZRZETEEhBhAKAsZ2F2M7DHNFDgA
  </glyph>
</modelIdentification>

```

Fig. 6. HLAstandardMIM.xml file which is provided by IEEE 1516 standard.

#### ACKNOWLEDGMENT

This work was supported in part by ADD (Agency for Defense Development) via the Gwangju Institute of Science and Technology research center for specialized electronic warfare.

#### REFERENCES

- [1] *IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) — Framework and Rules*, IEEE Standard 1516, 2010.
- [2] *IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) — Federate Interface Specification*, IEEE Standard 1516.1, 2010.
- [3] *IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) — Object Model Template (OMT)*, IEEE Standard 1516.2, 2010.
- [4] *IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP)*, IEEE Standard 1516.3, 2003.
- [5] *IEEE Recommended Practice for Verification, Validation, and Accreditation of a Federation — An Overlay to the High Level Architecture Federation Development and Execution Process*, IEEE Standard 1516.4, 2007.



**Younghun You** was born on February 4, 1990. He received the B.S. degrees in information & communication engineering from Sejong University, Seoul, Korea, and now being enrolled in a master's course.

He has a military background from 2009 to 2011. And his current research interests are phase retrieval and electronic warfare modeling and simulation.



**Tan Lee** was born on September 11, 1989. He received the B.S. degrees in information & communication engineering from Sejong University, Seoul, Korea, and now being enrolled in a master's course.

He has a military background from 2010 to 2012. And his current research interests are phase retrieval and electronic warfare modeling and simulation.



**Wooshik Kim** received the B.S. and M.S degrees in electronics engineering from Seoul National University, Seoul, Korea, and Ph.D. degree in electronics engineering from Georgia I.T., U.S.

His current research interests are phase retrieval and electronic warfare modeling and simulation.

# **Electronics and Communication Engineering**

