

## **HLA OBJECT MODEL TEMPLATE BASED FOM-AGILE CODE GENERATION**

### **SUMMARY**

The interoperability of simulations systems has incredibly become important after their wide usage including mostly military purposes, time-consuming and expensive systems, incrementing the effectiveness of education systems and performance analysis. The idea of reusability of the components that were previously designed for similar systems or the components that can serve within multiple system, made it necessary to have a common understanding in developing such systems.

Primary works in this area were started by DARPA under the name SIMNET, for the purpose of real-time weapon system development in late 1980s. From the beginning of 1990s, DARPA again led the research about making Distributed Interactive Simulations possible. These research yielded a NATO standard in 1995 by the STANAG 4482 agreement. Meanwhile, DIS is defined as 1278 Standard by IEEE. In 1996, High Level Architecture, firstly-named as DIS++ because of being defined as the next step of DIS, was defined in HLA 1.3 and eventually standardized in HLA1516 by IEEE.

All this effort for the name of developing common architecture for simulation systems, also caused a new requirement to define the data that is used in communication with respect to some other standard. Especially, after HLA was agreed as a standard, the definition of these data structures was also standardized with IEEE 1516.2-2000 Standard for Modeling and Simulation High Level Architecture - Object Model Template (OMT) by IEEE. Starting after that, object model templates are also defined for this purpose, led by SISO.

This work is intended to develop an auto-generated data communication layer from object model templates. The layer is supposed to be compatible with existing standards and can commonly be used by distributed simulation components. This layer is integrated with two additional layers within the developed framework, one for linking the application with the simulation and the other for developing the domain code onto it. The framework also designed as FOM-agile to keep the developed applications minimally affected from the changes in object model. The applications over the framework are also tested with existing applications from the point of view of compatibility, usage and performance.