

Exploring HLA Process Model Gateway Components

David Payne
The MITRE Corporation
7515 Colshire Drive
McLean, Virginia 22102
703-983-6366
dpayne@mitre.org

Keywords:

Multi Agency, Enterprise Analysis, Process Models, Interoperability, HLA

ABSTRACT: *As part of a MITRE Center for Enterprise Modernization (CEM) independent research effort, a team of MITRE Corporation researchers is exploring the feasibility of building HLA gateway components for use in several commercial process modeling environments. The concept is to capture the information elements that are exchanged when work objects flow through a process model, and then to devise an HLA federation object model, tentatively called COTS FOM, that is the superset of information elements needed by the various commercial environments to accept an incoming work object generated elsewhere in the federation. If successful this will greatly expand the utility of existing process models in situations where individual processes interact as part of a larger enterprise or multi-enterprise consortium. A secondary output of this effort will be a set of prototype gateway components built in two or more COTS process model environments that will interface with the native COTS components within a process model and with the HLA COTS FOM within the "COTS Fed" federation. Then the RTI would publish events whenever a work object enters an outbound gateway component, as well as subscribe to events that trigger generation of native work objects from incoming gateway components. An expected limitation is that model interoperability will be somewhat limited, in comparison to a combat or command and control model where one event is of interest to many federates and causes many federate level reactions. However, COTS process model simulations typically model incoming work requests or raw materials feeding a single federate work flow process that then outputs finished products or documents to the next process in line. It is believed that the constraints of the component based COTS modeling environments will accommodate the limited HLA interoperability envisioned by the study team.*

1. Introduction

As part of a MITRE Center for Enterprise Modernization (CEM) independent research effort, a team of MITRE Corporation researchers is exploring the feasibility of building HLA gateway components for use in several commercial process modeling environments. The concept is to capture the information elements that are exchanged when work objects flow through a process model, and then to devise an HLA federation object model, tentatively called "COTS FOM", that will define a federation (COTS Fed) which enables work objects generated in one commercial process simulation environment in the federation to move to another commercial process simulation environment in the federation. If successful this will greatly expand the utility of existing process models in situations where individual processes interact as part of a larger enterprise or multi-enterprise consortium.

2. The Process Model Interoperability Problem

A number of commercial off the shelf (COTS) simulation environments are available in the open market. Most of these now support, directly or through a so-called layered product, rapid graphical modeling, using a set of simulation components. This approach to simulation modeling is very popular in industry since it supports extremely rapid and flexible model generation, and allows analysts with little or no formal computer programming skills to model complex processes. These packages are used to model manufacturing processes, supply chains, transportation networks and transaction flows. They are commonly used in business process design and reengineering. Virtually all of the commercial environments support robust backward compatibility, so a large collection of models is often available when starting a new cycle of process analysis and improvement.

However, especially in today’s world of mergers, acquisitions, and multi-partner trading consortiums, as well as in government multi-agency collaborations, the available collection of process models are often rendered in different COTS environments. Thus the technology inhibits the attractive solution of building larger models of the combined processes by chaining existing models of component processes. Several efforts are underway to standardize the interfaces or data structures of COTS process simulation environments. Our approach is to explore how HLA can help.

3. An HLA Solution?

Our specific application is modeling and improving government multi-agency collaborative processes. We note that these processes typically involve a transaction or a string of transactions that flow mostly sequentially from one agency partner to another, eventually resulting in an end product or an end state. Even when work flows back and forth between partners, the cross flow tends to be discrete. We feel we can leverage this trait to use a simplified HLA federation design which will support generic interoperability between COTS environments, rather than model-specific interoperability, which would require a new federation design for each specific multi-agency model instantiation.

4. COTS Fed

Our HLA federation design will of course be expressed in a Federation Object Model (FOM). Our working name for this FOM is “COTS FOM”, which we will use to instantiate the “COTS Fed” federation.

4.1. Overview

COTS Fed will depend on two generic object models. COTS FOM is the superset of information elements needed by the various commercial environments to accept an incoming work object generated elsewhere in the federation. COTS Portal is a family of simulation component object templates that will support design of COTS environment specific work object import and export components, called portals. Once portal components are generated in a specific COTS environment, they should be

reusable, and act as extensions to the native COTS library of model building components. We considered leveraging the concept of Base Object Models (BOM), but currently feel that the state of definition of a SISO standard BOM is not mature enough to use as a base for our portals, and the direction of BOM standards may not directly apply to the type of reusable object template that we need to define for the portal objects. We will continue to monitor the evolution of BOM and will align our work with the BOM standard if it becomes apparent that such a standard describes the generic components we call portals.

4.2. COTS FOM

Since the initial scope of COTS Fed is to simply facilitate transfer of work objects between component based process models, we will initially define only one class in COTS FOM—the COTS Transfer Object class. COTS Transfer Object class will be publishable and subscribable within COTS Fed. The only interaction initially defined will be an acknowledgement that a COTS Transfer Object was received by the destination federate. This will allow the sending federate to then delete that specific instance of COTS Transfer Object, having served its purpose. The “CTO Received” interaction should be invoked by the receiving federate after the COTS Transfer Object has triggered generation of an equivalent HLA Transfer Object within the receiving native environment.

The COTS Transfer Object will have the attributes shown in Table 1, below.

Object	Attribute
COTS Transfer Object	Name
	Type
	Source Model
	Source Portal
	Destination Model
	Destination Portal
	Payload

Table 1. COTS Fed Object Attributes

Basically, the key innovation of our approach is to use a standard, stable, transfer object structure incorporating a “payload” attribute, which we will initially use to hold a list of lists, corresponding to the model specific attributes and corresponding attribute values we want to

transfer among COTS Fed federates. The other attributes are believed to be a minimal set of metadata required to inform the receiving model of the nature of the incoming object. The Name attribute may be “None” or may be a name for an instance of a work object that is significant to a specific set of COTS Fed federate models. The Type attribute is meant to be a class name defined in the receiving model’s internal object hierarchy. This Type is assumed to typically be a recognized type of object or artifact in the domain of the specific set of federates within a specific instantiation of COTS Fed. For example, it might be a specific Federal form number, that the federation participants have agreed is a valid transfer format in the domain modeled by their specific federation. The Source and Destination Model attributes simply support the limited scope of our approach. All federates in a pure COTS Fed federation will be subscribed to COTS Transfer Object, so all will be aware of all object transfers. Hence federates other than the specified receiver will be aware of the transfer between sender and receiver, and could, if desired, use the COTS Transfer Object to generate a local copy of the transferred object. Obviously, in a very active federation, with many federates, we will want to take advantage of the addressing and routing capabilities of HLA, and we will consider how to extend the COTS FOM definition to support this, but only after our initial experiments in COTS environment to environment level generic object exchange. Note also that COTS Fed could be pulled into a larger federation of models, which are not all COTS based process models, and in that case not all federates may choose to subscribe to COTS Transfer Object. The Source and Destination Portal attributes are optional fields that further define where work objects transfer between federate models. They will default to “none”, and are not needed at all in cases where federates have single points of work entry and exit. In those cases where a federate model has several input and/or output portals, the portal attributes will ensure that work objects flow through the models properly, and will support detailed work flow tracking when a federate is designed to monitor or track overall work flow.

4.3. HLA Transfer of Ownership

Given the work ongoing in improving formal HLA transfer of ownership, we intend to side step the issue initially by not formally

transferring objects between federates. Instead we initially intend to use the RTI to route COTS Transfer Objects to COTS Fed federates, and then use the CTO Received interaction to let the destination federate let the federation know that the COTS Transfer Object has been received. Then the owning sending federate can delete that instance of COTS Transfer Object. Other federates may internally track the location of work objects in the federated process model, and further track which federate currently controls each object. However this will be fully unique to any given COTS Fed instantiation.

4.4. COTS Environment HLA Gateways

The two required interfaces between HLA and component-based COTS process simulation environments are a COTS environment specific HLA gateway and native COTS environment portal components. The HLA gateway is implemented in a programming language and of course conforms to the requirements of the HLA Standard. More specifically, gateways typically conform to a specific release of the HLA RTI. Some COTS process simulation vendors have implemented HLA gateways and provide licensed users the gateway code, with or without charge. However, the use of HLA has not significantly penetrated the COTS process simulation market and most COTS environments do not have a vendor developed HLA gateway. And at least one vendor has pulled back their HLA gateway from a supported product to limited, “on request” release of developmental software. Hence, one major task facing us is to write an HLA gateway for several of our target COTS environments that do not supply such a gateway.

4.5. COTS Environment HLA Portal Components

The second required extension of the COTS environment is the addition of native simulation component objects designed to work as portals from the models to the HLA gateway. All of the modern COTS process simulation environments we have seen use the modeling paradigm of common simulation task components linked in some way to other simulation task components. This supports an easy to use graphical model building environment. So the natural approach to an HLA portal is a specialized simulation task

component that plugs into, or connects to, the other simulation task components and appears to send work objects running through the connected components to the RTI, and pulls work objects into the model from the RTI. Figure 1, below, provides a conceptual implementation of the COTS Fed portal components.

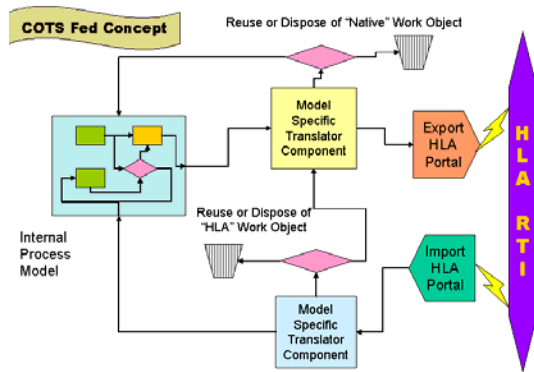


Figure 1. COTS Fed Concept Diagram

Our design for COTS FOM facilitates a generic, two component design for these portal objects. The translation component accepts all valid work objects in the native environment and produces a native form of the COTS Transfer Object, preferably locally defined as an “HLA Transfer Object”. The primary requirement for the translation component is to build a valid “payload” attribute for the HLA Transfer Object and, through that, the COTS Transfer Object. The second component is the portal itself, which receives incoming HLA Transfer Objects and programmatically interfaces with the COTS environment HLA gateway to publish a corresponding COTS Transfer Object on the RTI. At the receiving end we have the same arrangement in reverse. The other COTS environment HLA gateways detect the published COTS Transfer Object, via subscription. The import portal components programmatically interface with the HLA gateway, becoming aware of a new COTS Transfer Object. Then, depending on local implementation, the import portal component filters out unwanted COTS Transfer objects (that is, ignores those with other “destination” attributes), generates native HLA Transfer work objects corresponding to the COTS Transfer Object, or performs some more sophisticated locally defined filtering. The newly generated native HLA Transfer Objects are then sent to a translation component which has a locally defined routine to unpack the “payload”

attribute and generate a corresponding native work object, which is then released into the local work flow. The local model provides the work flow logic and routes work out of portals to the federation, and returns work from the federation, in a totally native fashion. This approach simply extends the graphical modeling paradigm that users of these COTS environments have mastered to rapidly produce sophisticated process simulations.

The two different components in the portal assembly (the translator and the portal itself) are designed to minimize the risk that a misguided modeler will redefine the behavior of the HLA portal component, and thus break the generic HLA capability provided for their COTS environment. The suggested design focuses local component extensions and modifications on the local translator component, which requires no specific HLA expertise. The only connection to COTS Fed within the translator component is the requirement to conform to the COTS Transfer Object class definition, expressed in the native object format of the COTS tool. Most COTS simulation environment users are at least moderately familiar with, and competent with, native object definition and use.

5. Using COTS Fed

Our design for COTS Fed is aimed at producing a generic, component based, HLA linking mechanism. The COTS FOM, the HLA gateways, the portal components and the translation components should only need to be defined once, and then are available for use with many federations without modification. However, bringing up a COTS Fed federation will take some HLA expertise. The start up sequence will be to start the RTI, then start the HLA gateways, and finally to start the component models. Each federation will have to work out synchronization issues to ensure that all federate models are running on the same simulation clock. This should be an easier task than usual in COTS Fed, though, since the work flow basis of process simulations enforces temporal dependencies, provided that the federate models are correctly built to reflect real world precedence and concurrence constraints. Again, this is something that the COTS process simulation user community is well aware of, and the COTS Fed model distribution approach simply builds on these modeling skills.

6. Current Status of COTS Fed

This paper is currently the primary functional description of COTS Fed. We plan to do more detailed design work over the summer of 2003 and begin implementing COTS Fed in Fall 2003. We are currently planning to develop in parallel the COTS FOM and portal components for two COTS environments, ReThink and Extend. We will also have to implement an HLA gateway for Extend. (Gensym, the developer of ReThink, has provided us an HLA gateway package for G2, the underlying environment for ReThink.) Later, perhaps in early 2004, we may be able to also develop portal components and an HLA gateway for Bonapart (sic). Finally, we have established a beginning dialog with the HLA - COTS Simulation Package Integration Forum (SIW-SG-HLA-CSPIF) virtual forum, who have been working on similar approaches to linking COTS models, primarily through the European Simulation Interoperability Workshop (Euro SIW) series.

HLA-CSPIF has focused on linking three different COTS process simulation environments than we are focusing on, so if we can successfully coordinate our efforts we may be able to provide the community an extensive set of interoperable COTS process simulation environments in a relatively short period of time. This should also speed definition of and acceptance of a new SISO standard, formally detailing our combined approach.

7. References

- [1] Bonapart information: PIKOS GmbH Site: <http://www.pikos.net/english/> .
- [2] Extend information: Imagine That Inc. Web Site: <http://www.imaginethtatinc.com/> .
- [3] Payne, Hoffman, and Zheng. “Using Interoperable Process Models in a Multi-Agency Planning Toolkit for Enterprise and C4ISR Architecture Analysis”, (03F-SIW-014), Fall 2003 Simulation Interoperability Workshop (SIW), September 2003, Orlando, FL.
<http://www.sisostds.org/conference/index.cfm?conf=03fall> .
- [4] ReThink information: Gensym Corporation Web site: <http://www.gensym.com> .
- [5] Taylor, Simon. “High Level Architecture – COTS Simulation Package Interoperability Forum (HLA-CSPIF)”, (non-paper presentation 03E-SIW-129.ppt), European Simulation Interoperability Workshop 2003, June 2003, Stockholm Sweden.
<http://www.sisostds.org/conference/index.cfm?conf=03fall> .