Proceedings of the 2007 Winter Simulation Conference
S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, eds.

DISTRIBUTED SIMULATION FOR INTEROPERABILITY TESTING ALONG THE SUPPLY CHAIN

Sanjay Jain

School of Business,
The George Washington University
2201 G Street NW, Suite 415
Washington, D.C. 20052, U.S.A.

Frank Riddick
Andreas Craens
Deogratias Kibira

Manufacturing Systems Integration Division,
National Institute of Standards and Technology
100 Bureau Drive, MS 8260
Gaithersburg, M.D. 20899, U.S.A.

ABSTRACT

The need for interoperability of information systems among supply chain partners has been recognized. A number of standards have been or are being developed to ensure interoperability of applications used along the supply chain. An associated need for interoperability testing has emerged. There is a need to evaluate compliance of applications to standards across specific platforms. The standards themselves need to be evaluated for a comprehensive coverage of the application scope (validation testing). This paper reports on a distributed simulation based approach for supply chain interoperability testing. Simulations are used to represent real life organizations to serve as sources and consumers of dynamic data. The data can be encapsulated per the standard under consideration and exchanged with other organizations directly or through selected applications for testing. Error free performance of the simulated systems over time will provide confidence in the interoperability of applications and in the standards themselves.

1 INTRODUCTION

The success of supply chains is highly dependent on seamless exchange of information among its partner organizations. The supply chain applications at any node in a supply chain should be able to exchange information with supply chain applications at other relevant nodes. Currently, most of the interfaces between the applications at different nodes along the supply chain are either proprietary or custom designed for specific implementations of selected applications. For example, the Enterprise Resource Planning (ERP) software at an automotive supplier may integrate with the same ERP software at the customer automotive company through its proprietary interfaces. Or, the supply chain planning software running at a component supplier may integrate with the ERP software from another vendor at the assembly operation through a custom designed interface. Such arrangements limit an organization's ability to quickly find new partners for forming supply chains to rapidly bring out new products.

The ability for quick integration of supply chain applications across different organizations is critical for competitiveness. Such ability can be provided through interoperability standards. This need has been recognized and consortiums established for creating interoperability standards.

An organization interested in using standards for supply chain interoperability is faced with a number of competing standards at various stages of development, and with a number of applications claiming to be compliant to some or all of the standards. There is no easy way available for an organization and its existing or potential supply chain partners to test the applicability of the standards to their operation environment. There is also no one methodology available for testing the compliance of the applications to the standards in their own operational environment. It takes a large effort to implement and integrate any supply chain application and hence, it is simply not practical to implement an application for testing its features and its interoperability. Examination of data files and fields allows syntactic verification of compliance based on the specific set of data used. There is no guarantee that the static data files can cover the large range of a different set of values that can occur during real operations.

The need for testing of standards and the compliance of applications to the standards over a range of real operation scenarios can be met through simulation. The applications can be interfaced with simulated (or virtual) manufacturing systems and tested with the streams of data transactions and messages generated during simulated operations. Researchers at the National Institute of Standards and Technology (NIST) are developing a virtual manufacturing environment for just such a purpose.

The Virtual Manufacturing Environment (VME) project at NIST is part of the Manufacturing Interoperability program (MIP). The goal of the MIP is to equip U.S. manufacturers with the technical guidance and testing support needed to interoperate in today's global, heterogeneous manufacturing world. The VME project is creating data-driven simulators across the manufacturing hierarchy extending from the supply chain network level to a process on the production floor (Kibira and McLean, 2007). The vision is to create the models with the fidelity suitable for testing applications at the different hierarchical levels. The first prototype implementation reported here is focused on the automotive industry, though the structure can be used for any supply chain with an assembly operation as the anchor.

This section introduced the need and the approach used to address it. The next section briefly reviews related work in this area. Section 3 defines the scope of this effort. Section 4 describes the simulation models used to represent the supply chain network and the automotive assembly plant. The conceptual view of the integration of the simulation models in a distributed manner is discussed in section 5. The implementation view including the modifications made to the models to enable the integration and the distributed simulation infrastructure are presented in section 6. The last section concludes the paper.

2 RELATED WORK

The Internet is being increasingly used to create links between information systems of organizations partnering in supply chains. A huge number of interconnections between multiple applications are being set up. These interconnections are being set up in a custom manner with large efforts. The need to develop interoperability standards has been recognized and is being responded to. It has been noted that the large number of interconnections and associated complexity threatens to overwhelm the capabilities of standards organizations and industry to develop the specifications (Ray, 2002). A number of organizations are engaged in developing interoperability standards for various applications.

There has been a significant amount of effort on interoperability among distributed simulations including the development of the high level architecture (HLA) (Kuhl et al 1999). HLA is a Department of Defense-developed set of rules, a method for information definition, and an interface specification that defines a way for simulations (called federates) to be combined into a distributed simulation (called a federation). HLA federates interact through a software component called the Run-Time Infrastructure, which implements the interface defined by HLA.

More recently, there is an effort on building a family of standards for interoperability of commercial off the shelf discrete event simulation packages under the auspices of the Simulation Interoperability Standards Organization (SISO) (Taylor et al 2006). The work reported in this paper does use an HLA-based approach for integration of the distributed simulation, but does not test such standards. The focus here is on testing of the interoperability standards for information exchange between supply chain partners.

Related efforts at the National Institute of Standards and Technology are investigating the use of semantic web technologies for enterprise integration (Anicic et al 2007) and a testbed for manufacturing business to business interoperability (NIST 2007a). The approach described in this paper should be able to detect semantic inconsistencies through analyzing the causes of poor and unexpected performances of simulated organizations and supply chains. Admittedly this would be a somewhat tedious process.

3 SCOPE

Interoperability testing involves multiple levels as below (NIST 2007b):

1. that the standards meet the business requirements they were intended to address (validation testing),
2. the standards conformance of key implementations – what was implemented agrees with the specification (conformance testing), and
3. that sets of business applications can successfully operate together (interoperability testing).

The simulation-based approach is aimed at testing the interoperability standards primarily for items (1) and (3) in the above list though it can be used for item (2) also. First, it can be used for validation testing of standards. The interactions between different nodes of the supply chain can be modeled using the data fields included in the applicable standards. The simulations can indicate if the data necessary for executing the basic supply chain applications is available for the range of situations created during the simulation. Second, the approach can be used to test interoperability of supply chain applications from software vendors. The simulation models can act as the real life operations and interface with the applications. Data and messages will be generated and provided to the applications using the standard form and format to test their capability to read and parse the information correctly, and to generate correct outputs that can be sent to other nodes represented either by same or different simulation models.

The work reported in this paper is focused on testing standards for interoperability of applications that support some aspect of a supply chain operation typically involving communications among partner organizations in a supply chain. The primary example of such standards is the Open Applications Group's Integration Specification (OAGIS). The version used here was one developed in collaboration with the Automotive Industry Action Group (AIAG) and made available for free downloads (OAGi, 2007). Other examples of such standards include ISA-95 (ISA, 2007) and Business to Manufacturing Markup Language (B2MML) (WBF, 2007).

The proposed virtual manufacturing environment will provide the application software vendors the capability to demonstrate the interoperability of their products. It will provide the manufacturing organizations the capability to test out the supply chain applications under consideration. Overall, it will help support the move to improved interoperability among organizations in supply chains and improved competitiveness of U.S. industry.

4 SIMULATION MODELS

This effort uses a simulation-based approach for interoperability testing. Simulations are used to represent real life manufacturing entities and hence avoid the problem of finding real life operations for interoperability testing. A number of simulation models integrated in a distributed simulation framework are proposed to allow modular development and maintenance.

This section includes description of the supply chain simulation at the network level and the manufacturing plant simulation.

4.1 Supply Chain Network Simulation

The purpose of the supply chain simulation is to provide a representation that generates dynamic information exchanges that would be created in a real life supply chain in order to test standards and interoperability of supply chain applications. The simulation executes a model of interactions and material and information flows through a defined supply chain network extending from suppliers to customers.

The scope of the supply chain model includes manufacturing facilities with multiple stages of suppliers on the input side and multiple stages of distribution network on the output side. Each of the supplier facilities is modeled at an abstract level based on the capacity of bottleneck and the lead-time through the facility. The flow of material is tracked at the supplier at three major stages: raw materials, work in process, and finished goods. Suppliers can send their outputs to multiple consuming facilities. For example, a tier II supplier can send its products to a tier I supplier and to the manufacturing facility directly.

The manufacturing facility itself is modeled in a bit more detail, with major sections (lines or departments) of the facility modeled with their individual bottlenecks. The flow of product is tracked through the stages of raw material (components), work in process within the major sections and in-between the sections, and finished goods. The production activity is modeled at shift level.

The distribution network can be modeled to include flow of product to distribution centers, retailers and customers either linearly through these stages or directly to any of them. Customer purchase activity through the retailers can be modeled by specifying appropriate distributions. The logistics is modeled at an abstract level with travel times defined in integer days based on a from-to matrix.

The current implementation of the model is based on a generic automotive supply chain data set with the final assembly plant at the center as the manufacturing facility, tier I and tier II automotive suppliers on the supply side, and distribution centers, car dealers and customers on the consumption end.

The model mimics the dynamics of the supply chain and associated interactions between the supply chain nodes. These interactions can be executed through transactional messages between the nodes consistent with a standard that may be under evaluation. With the current implementation of an automotive supply chain, the interaction messages use data fields consistent with those defined in OAGIS/AIAG Business Object Documents (BODs) for Inventory Visibility and Interoperability (IV&I) (OAGi, 2007). For example, the orders for vehicles from dealers to assembly plants are defined using XML fields defined in the ProcessPurchaseOrder BOD of the OAGIS standard. Similarly, the shipment notifications that are sent from the assembly plant to dealers use XML messages that are formed using the SyncShipmentSchedule BOD specification.

The supply chain simulation has been developed using ARENA. The simulation is data driven with all the supply chain parameters defined in variable structures. The data for the automotive supply chain has been defined in an Excel file and brought into ARENA to create the model. A graphical user interface (GUI) module has been developed using C## for flexibility in accessing and presenting information beyond that offered by ARENA. The GUI module communicates with the simulation model through an MS-Access database using the .NET framework.

The supply chain simulation model has been developed with a data-driven interface and basic representations of supply chain management policies. A supply chain can be described using data on various aspects including the structure of the supply chain, the demand characteristics, the capabilities of each node, and the policies for inventory management within each node. The simulator can read in the data and configure the model accordingly. The simulation can then dynamically execute the model to create instances of real life situations that may occur during operations of the supply chain.

The input data for the automotive supply chain has been developed as a neutral data set that is representative of the industry. It does not include any proprietary information and can be shared freely among researchers. The data includes a high level description of an automotive bill of material, the suppliers for the major components, the assembly plant, the dealers and the logistics network connecting all the nodes. A more detailed, external simulation model of a manufacturing plant, described in the next section, can replace the assembly plant node in the supply chain model.

4.2 Manufacturing Plant Simulation

The purpose of the manufacturing plant simulation model is to generate dynamic information representative of a real life assembly plant to enable testing of standards and applications for plant level information and decision-making systems. The simulation mimics the flow of product and associated information in a manufacturing plant. The interfaces to the model can be built using selected standards. Similar to the supply chain simulation, the current implementation uses interactions messages defined in the OAGIS standard.

The scope of the manufacturing plant simulation model includes all the major sections of the plant with the key workstations represented. In the current implementation of an automotive assembly plant, its three major sections, namely body shop, paint shop, and general assembly are modeled with a number of workstations for each connected by the appropriate material handling system. This level of detail is required for realistic representation of interaction messages such as those for Kanban consumptions that are included in the OAGIS standard. While a summarized model can be used to generate such messages, it may not gain confidence of the manufacturing personnel. Kibira and McLean (2007) provide a detailed description of the automotive assembly plant model. The shipment from the plant to distribution centers and/or car dealers is modeled in the supply chain simulation described above.

The manufacturing plant simulation model can also be developed using a data-driven approach in the future. The plan is to utilize an automotive assembly plant description defined using Core Manufacturing Simulation Data (CMSD) standard currently under development under the auspices of SISO (Leong et al 2006). Another standard that may be considered in future for modeling control systems is the ISA-95 (ISA, 2007).

The assembly plant model utilizes a more detailed bill of material than used for the supply chain simulation, definitions of stations on the line, and policies controlling the scheduling and flow of vehicles through the facility. In the future, the simulation will include a capability to replace line segments with an external detailed model briefly described next.

4.2.1 Line and Work Cell Simulations

The purpose of detailed simulations of parts of manufacturing plant is to generate dynamic control messages representative of real life manufacturing lines and work cells to enable development and testing of standards and applications at that level. The current line level implementation uses CMSD-based data files for reading in a description of a more detailed system. The paint process line of an automotive plant has been incorporated that uses Enterprise Dynamics software. The model is based on proprietary data and hence its details are not included. The experience will be useful for developing a publicly accessible model in the future.

5 CONCEPTUAL VIEW OF THE DISTRIBUTED SIMULATION PROTOTYPE

To move towards the goal of being able to test interoperability between supply chain applications, a prototype has been developed that integrates the simulations at two different levels of the supply chain. In the supply chain network simulation, the important activities of all of the partners (except the final assembly plant) of an automotive assembly chain supply are modeled. In this simulation, internal forecasts for production goals and specific production requests from retail car dealer partners are used to generate production orders for specific models of cars, purchase orders for the subcomponents needed to produce those cars, and the transportation request to move the sub-components and end products between the partners in the supply chain. In the final assembly plant simulation, orders for the production of specific quantities of specific models of cars direct which products are produced and in what quantities. Product mix and the initiation of production to fulfill an order are dynamically determined during the simulation run from external sources (the Supply Chain Network simulation). This is quite different from the typical simulation of production activities where stochastically defined random variables usually determine the product mix and the time for the start of production for an order. The final assembly plant simulation then carries out the activities necessary to produce the cars requested, monitoring and managing the sequence of models of cars produced to maintain efficient order fulfillment. As cars finish production, they are matched to orders requesting that model of car. When all of the cars necessary to fulfill an order have been matched to that order, the order is marked as completed and a shipment message it sent to the supply chain network simulation.

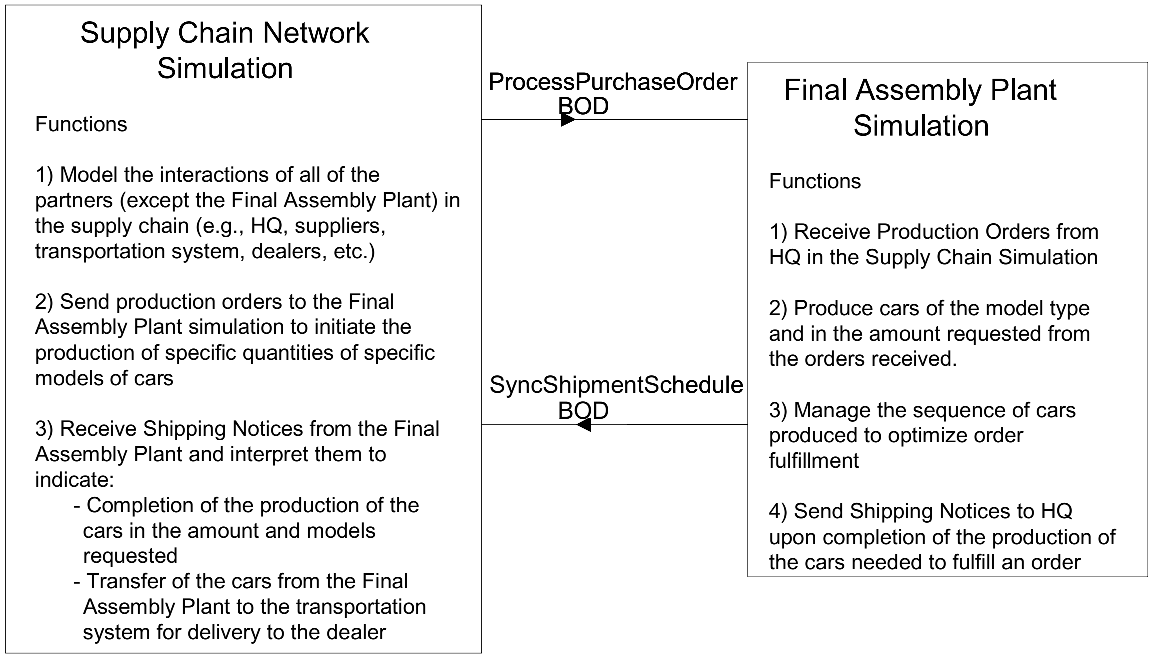

Figure 1: Conceptual view of distributed simulation prototype.

The format used for the production request that is sent from the supply chain simulation to the final assembly plant is the ProcessPurchaseOrder BOD. The format used for the shipping request that is sent from the final assembly plant to the supply chain simulation is the AcknowledgeShipment BOD. BODs are a part of the OAGIS specification, and are based on XML, and defined using an XML Schema. In the two BODs used in this prototype, all possible elements that can be defined for the BODS are not used, but both BODs contain information about the product, model, customer, customer location, and product amount. Figure 1 shows a high level functional view of the two simulations and how they interact.

The described conceptual set up provides the capability of a supply chain anchor organization interacting with the final manufacturing stage. Individual models similar to that of the assembly plant will need to be developed for modeling interactions between supply chain partners, such as the suppliers and the assembly plant. A supply chain model will interface with multiple distributed models representing selected suppliers and the assembly plant for such a scenario.

6 IMPLEMENTATION VIEW OF THE DISTRIBUTED SIMULATION PROTOTYPE

This section describes the enhancements made to each of the two models to allow interaction within a distributed simulation infrastructure. The infrastructure itself is described also.

6.1 Supply Chain Network Model Enhancements

Information involving requests for the production of finished cars are sent from the Supply Chain Network simulation to the Final Assembly Plant simulation in the form of OAGIS ProcessPurchaseOrder BODs. Microsoft Visual Basic for Applications (VBA) language was used to create model logic to access data stored in Arena global variables and to use that data to create BODS for transmittal. This approach was taken because Arena provides the ability to write model logic using the VBA language, and VBA code has the ability to access XML processing functionality commonly available on Windows systems. VBA was also used to create logic to extract data on finished car shipments from the SyncShipmentSchedule BODs received from the Final Assembly Plant simulation. This extracted data was then used to update global variables associated with simulated enterprise partners (such as the transportation network partner and dealer partners) in the Supply Chain Network simulation.

The Supply Chain Network simulation interacts with the Final Assembly Plant simulation by accessing functionality provided by the Distributed Simulation Manufacturing Adapter (DMS Adapter). The DMS Adapter is a NIST-developed component that provides a simplified means to create manufacturing-oriented distributed simulations that execute as a High Level Architecture federation (additional information about the DMS Adapter is provided later in this paper). VBA-based logic was added to the Supply Chain Network simulation to send (XML-based) messages, to receive messages, and to coordinate the advancement of simulated time with other federates using the facilities of the DMS Adapter.

6.2 Final Assembly Plant Model Enhancements

The Final Assembly Plant simulation was developed using Delmia's Quest simulation product. In this simulation, the frame and body assembly area, the paint shop, and the final assembly area for the production of cars is modeled. In each area, cycle times for each workstation, maximum capacity for each buffer/staging area, and cycle times for conveyors, robots, and other material handling equipment are modeled. The Final Assembly Plant simulation currently supports the manufacture of 12 different models of cars, but this can be easily extended.

Unlike typical models of assembly lines built in Quest, the initiation of production in the Final Assembly Plant model is based on the receipt of an OAGIS ProcessPurchaseOrder BOD from the Supply Chain Network simulation. The BOD contains information about the model and quantity of cars to produce, requested due date for order completion, and location and identification information about the customer to ship the completed cars to after they are produced. This information is extracted from the BOD using functionality from a specially written library that adds limited XML processing capabilities to Quest. The extracted information about each received order is maintained in a structure called the OrderList. As cars leave the last station, each finished car is associated with an order in the OrderList for that model of car. When the quantity of cars necessary to satisfy all of the requirement of an order have been assigned to that order, the information associated with that order is used to create a SyncShipmentSchedule BOD, and the BOD is sent to the Supply Chain Network simulation. Conceptually, this indicates that the cars have been transferred to the transportation partner, and are in transit to the customer that ordered them.

Another deviation from typical operation of Quest assembly line simulations is the use of a "production sequence" to determine the sequence of models that will progress through the assembly line. When a ProcessPurchaseOrder BOD is received from the Supply Chain Network simulation, a "sequence element" representing the requested car model is added to an internal structure called the production sequence for each car requested. When the cycle time for the line causes each production unit to move to the next station in the line, the next sequence element in the production sequence determines model of the next car to be produced, and causes the start of production of the sub components to complete that car model. The production sequence can be accessed by stations at the beginning of each of the three major areas of the plant, and the sequence can be altered to implement sequencing strategies to increase the operational efficiency of the assembly line.

The message processing, sequence management, and order list management logic are all written using Quest's Simulation Control language (SCL). In addition, most of the logic associated with Quest model elements (e.g., stations, conveyors, buffers, etc.) in the model have been modified to take advantage of the message processing, sequence management, and order list management logic.

The Final Assembly Plant simulation interacts with the Supply Chain Network simulation by accessing functionality provided by the Distributed Simulation Manufacturing Adapter (DMS Adapter). Quest SCL logic was added to the Final Assembly Plant simulation to send (XML-based) messages, to receive messages, and to coordinate the advancement of simulated time with other federates using the facilities of the DMS Adapter. Additional information about the DMS Adapter is provided next.

6.3 Distributed Simulation Infrastructure

The Supply Chain Network simulation and the Final Assembly Plant simulation work together as components in a distributed simulation. These simulations coordinate their startup, time advancement, and termination; and exchange data through messaging. This is accomplished by accessing the functionality provided by software based on the High Level Architecture (HLA) distributed simulation standard (Kuhl et al. 1999).

While HLA provides a flexible and robust methodology for developing distributed simulations, it is complex, requires special skills for effective use, and the interface it provides for (potential) federates to interact with does not integrate well with most existing COTS DES simulation packages. To address these concerns, NIST develop a software component called the Distributed Manufacturing Simulation Adapter (DMS Adapter). The DMS Adapter provides a simplified means to create manufacturing-oriented distributed simulations that execute as a High Level Architecture federation. It provides a simplified interface through which simulations can start distributed simulations, coordinate the advancement of time, and exchange data as XML messages. The interface to the DMS Adapter is implemented using Microsoft Component Object Model (COM) technology, which allows it to be accessed by many applications built using Microsoft-based technologies.

The Supply Chain Network simulation and the Final Assembly Plant simulation use the DMS Adapter to interact with and to create distributed simulations. Since the Supply Chain Network simulation is built using Arena, it can directly interact with the DMS Adapter using Arena's VBA interface. The Final Assembly Plant simulation is built using Quest, and Quest does not have the

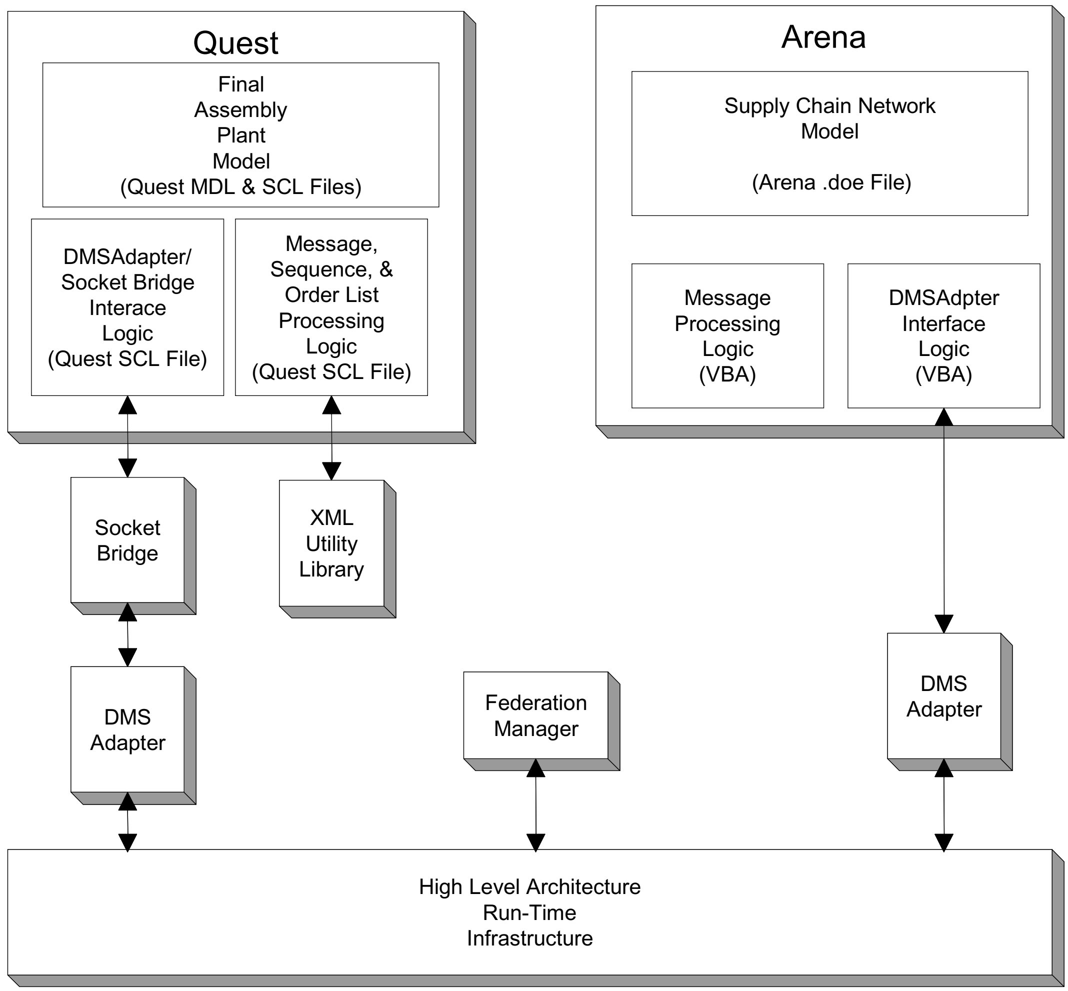

Figure 2: Set up for distributed simulation prototype

capability to directly interact with a COM object. Since Quest can interact with external applications through sockets, a Socket Bridge application was written to provide access to the DMS Adapter's interface. Each method in the DMS Adapter's interface was replicated as an SCL function, and each function is implemented to use the SocketBridge to send and receive information to/from the DMS Adapter. This allows all the other SCL code in the model to access the DMS Adapter's functionality just as if it could be directly accessed.

In addition to the Supply Chain Network simulation and the Final Assembly Plant simulation, a Federation Manager component is also present in the federation. Its purpose is to coordinate the start of the distributed simulation event by ensuring that both of the other simulations are initialized and ready to run before time in the federation is allowed to advance. It also provides capability to send termination messages to the other two simulations for controlled shutdown. The set up for distributed simulation is shown in figure 2.

7 CONCLUSION

A distributed-simulation-based approach was described for testing of interoperability standards for applications used among organizations partnering in a supply chain. The approach has been implemented using a prototype. The prototype was used for a small test using two messages exchanged between the supply chain simulation and the assembly plant simulation. The test was successful in the sense that the simulations executed smoothly with correct usage of the information shared. This was expected since a very basic set of fields was used. More importantly, the prototype and the tests have confirmed the feasibility of the approach.

Future work will focus on including a representative set of BODs from OAGIS standard that cover a full range of interaction messages for supply chain operations. Information system applications will be integrated to the simulation models, and the interactions will be executed between the applications to mimic the real life situation. Such a set up will help test the interoperability of the applications using interaction messages based on standards. Efforts will be made to develop a case study working with the industry. The set up will also be enhanced to test other standards such as the CMSD standard under development by SISO. Other directions under consideration include moving the DMS adapter implementation to .Net technology and a simulation integration platform independent of HLA.

Successful completion of the proposed work will facilitate the interoperability of manufacturing system applications through providing a way to test standards and applications. Use of standards such as CMSD can tremendously speed up the building of the simulations required for testing of interoperability standards and for simulation studies in general.

ACKNOWLEDGMENT

Work described in this report was sponsored by the National Institute of Standards and Technology (NIST) Manufacturing Interoperability Program, Gaithersburg, Maryland. The Manufacturing Interoperability Program supports NIST projects applying information technologies and standards-based approaches to manufacturing software integration problems. The work described was funded by the United States Government and is not subject to copyright.

DISCLAIMER

A number of software products are identified in context in this paper. This does not imply a recommendation or endorsement of the software products by the authors or NIST, nor does it imply that such software products are necessarily the best available for the purpose.

REFERENCES

Anicic, N., Marjanovic, Z., Ivezic, N., Jones, A., 2007. Semantic Enterprise Application Integration Standards. *International Journal of Manufacturing Technology and Management*, Vol. 10, No.2/3 pp. 205 – 226.

ISA, 2007. ISA-95 Manufacturing Enterprise Systems Standards and User Resources. Available on-line via: `<www.isa.org>`[Last accessed on April 16, 2007].

Kibira, D., and McLean, C.R., 2007. Generic Simulation of Automotive Assembly for Interoperability Testing. In *Proceedings of the 2007 Winter Simulation Conference*.

Kuhl, F., R. Weatherly, and J. Dahmann. 1999. *Creating Computer Simulations: An Introduction to the High LevelArchitecture*. Upper Saddle River, New Jersey: Prentice Hall.

Leong, S., Lee, Y. Tina , Riddick, F., 2006. A Core Manufacturing Simulation Data Information Model for Manufacturing Applications. In *Proceedings of the Systems Interoperability Standards Organization 2006 Fall Simulation Interoperability Workshop*. Paper 06F-SIW-028.

McLean, C.R., Riddick, F., and Lee, Y.T., 2005. An Architecture and Interfaces for Distributed Manufacturing Simulation. *Simulation: Transactions of the Society for Modeling and Simulation International.* Vol. 81, No. 1, pp. 15 - 32.

NIST, 2007a, B2B Interoperability Testbed, available online via `<http://www.mel.nist.gov/msid/b2btestbed>` [Last accessed on July 16, 2007].

NIST, 2007b, Manufacturing Interoperability, available online via `<http://www.mel.nist.gov/proj/mi.htm>` [Last accessed on April 16, 2007].

OAGi, 2007. Open Application Group's Integration Specification (OAGIS) 8.0 SP2 with AIAG Overlay 1.0a. Available on-line via: `<http://www.openapplications.org/downloads/oagisaiag/oagisaiag.htm>` [Last accessed on April 16, 2007].

Ray, S.R., 2002, Interoperability Standards in the Semantic Web, Journal of Computing and Information Science in Engineering, March 2002, Vol. 2, Issue 1, pp. 65-69.

Taylor, S.J.E., Wang, X., Turner, S.J., Low, M.Y.H., 2006. Integrating Heterogeneous Distributed COTS Discrete-Event Simulation Packages: An Emerging Standards-Based Approach. IEEE Transactions on Systems, Man & Cybernetics: Part A, Jan2006, Vol. 36 Issue 1, pp. 109-122.

WBF, 2007. Business to Manufacturing Markup Language. Available on-line via: `<www.wbf.org>`[Last accessed on April 16, 2007].

AUTHOR BIOGRAPHIES

SANJAY JAIN is an Assistant Professor in the Department of Decision Sciences, School of Business at the George Washington University (GWU), and works part-time at NIST under a research arrangement. Sanjay serves as an associate editor of the *International Journal of Simulation and Process Modeling* and also as a member of the editorial board of *International Journal of Industrial Engineering*. He is a senior member of the Institute of Industrial Engineers and a member of APICS - The Association for Operations Management. He received a Bachelors of Engineering from Indian Institute of Technology (IIT)-Roorkee, India, a Post Graduate Diploma

from National Institute of Industrial Engineering, Mumbai, India, and a Ph.D. in Engineering Science from Rensselaer Polytechnic Institute, Troy, New York. His email address is <jain@gwu.edu>.

FRANK RIDDICK is a staff member in the Manufacturing Simulation and Modeling Group in The National Institute of Standards and Technology (NIST) Manufacturing Systems Integration Division. He has participated in research and authored several papers relating to manufacturing simulation integration and product data modeling. He holds a Master's Degree in Mathematics from Purdue. His email address is <riddick@cme.nist.gov>.

ANDREAS CRAENS is a consultant with the Strategic IT Effectiveness (SITE) group within Accenture in the Netherlands. Earlier, he worked as a Guest Researcher with the Manufacturing Systems and Integration Division (MSID) at NIST where he was responsible for the development of the Generic Supply Chain Simulator (GSCS) module in the Virtual Manufacturing Environment (VME) Project. In 2006 André received his Master in Science degree in Business Administration (MSc BA) from the University of Groningen, Netherlands. He also has a bachelor's degree in Economics and Management from the same university and a Bachelor of Information and Communication Technology from the Hanze University, Netherlands. At NIST, he also served as the president of the NIST Guest Research Association (GRA). Email: <andre@craens.nl>

DEOGRATIAS KIBIRA is a Senior Lecturer in the Department of Mechanical Engineering at Makerere University in Uganda where he teaches Manufacturing and Quality systems. He has wide research experience in manufacturing simulation and production scheduling. He is currently a Guest Researcher at the National Institute of Standards and Technology where he is part of the research team involved in the development of a Virtual Manufacturing Enterprise for motor vehicle manufacturing. He has a first class honors degree in Mechanical Engineering from Makerere University and Masters and PhD degrees in Manufacturing Engineering from the University of New South Wales, Australia. His e-mail address is <kibira@cme.nist.gov>.