A Manufacturing B2B Interoperability Testbed

Nenad IVEZIC[1], Boonserm KULVATUNYOU[1] , Albert JONES[1]
[1]*National Institute of Standards and Technologies, 100 Bureau Drive, Gaithersburg, MD, 20817, USA*
Tel: +001 301 975-3536, Fax: + 001 301 975 4482, Email: {nivezic|serm|ajones@nist.gov}

Abstract: This paper describes the NIST Manufacturing Business-to-Business Interoperability Testbed developed at the Manufacturing Systems Integration Division of the National Institute of Standards and Technology. The testbed is geared to advance state of practice and art in interoperable information systems for the extended manufacturing enterprise. We discuss lessons learned while developing a Web-based, distributed architecture in support of piloting and testing interoperability trials in collaboration with manufacturing organisations and software vendors. We make a case that the combination of industry-focused testing activities and bottom-up testing research and development efforts within this testbed offer unique benefits to all stakeholders in advancing interoperable systems for the manufacturing enterprise.

1. Introduction

The NIST Manufacturing Business-to-Business (B2B) Interoperability Testbed is a collection of activities performed at the Manufacturing Systems Integration Division (MSID) of the National Institute of Standards and Technology (NIST) to advance state of practice and art of interoperable information systems in the extended manufacturing enterprise.

The testbed allows research, development, experimentation, demonstration, and testing to closely co-exist within a common environment. The testbed architecture is adopted to encourage both bottom-up 'technology push' in the context of realistic enterprise integration scenarios and top-down business 'needs pull' in the context of new technology development.

This paper describes the testbed architecture, testing technologies, and lessons learned through developing a Web-based, distributed architecture in support of monitoring and testing interoperability trials and working with manufacturing organisations and software vendors to design and run interoperability pilots. We conclude the paper by identifying a series of ongoing research and development areas to enhance the testbed capabilities.

2. Testbed Architecture

The distributed testbed architecture comprises participating nodes of two logical types: the test/monitor type and middleware/application type. A test/monitor node is a single logical node that, however, may consist of multiple distributed functions running on multiple nodes. The middleware/application nodes are distributed among participating organisations (i.e., both software vendors and users). The top of Figure 1 shows these node types and the supported testbed interoperability stack.

To enable interoperable behaviour of these nodes, standards at different levels of the interoperability stack are adopted. The testbed has focused on three layers of the architecture: messaging, business processes, and business content. The standards that are being used are ebXML Messaging, Business Process Specification Schema (BPSS), Collaborative Partner Protocol and Agreement (CPP/A) and the OAG Business Object Document (BOD) content standards [1,2,3,4]. However, the testbed architecture only requires that the HTTP protocol [14] be used in order to support basic functions of monitoring and testing interoperability.

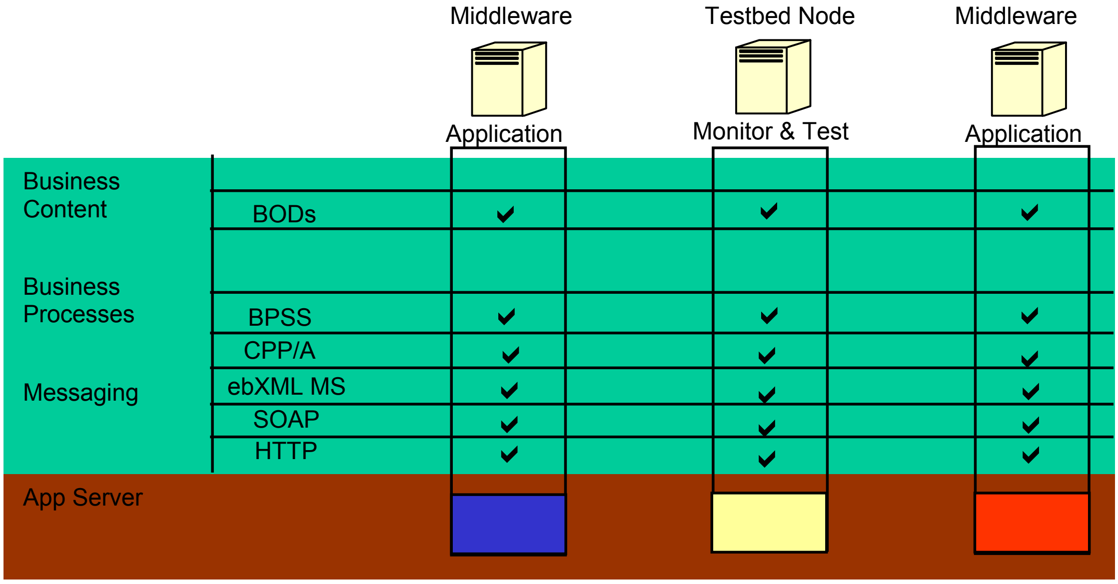

Figure 1: The testbed node types and supported interoperability stack

3. Testbed Tools

With the emergence of a number of standards to define message content, business process choreography, and messaging protocols as well as new software products supporting these standards, there is a need to provide monitoring and conformance checking tools to assist in demonstrations and testing of these standards and products. A number of monitoring and testing tools have been constructed and adopted to support testbed functions:

- The Reflector is a testing tool that supports both disconnected and connected testing scenarios while allowing for the transactions to be routed to the specified end points, reflected to the originator, and stored in a permanent transaction log [5].
- The Business Process Monitor enables monitoring and conformance checking for choreographed transactions between business partners. The monitor currently supports ebXML BPSS and CPA standards [2,3]. The tool provides a Web-based graphical user interface to monitor in real time the business interactions based on the ebXML BPSS specification. The monitoring tool takes the ebXML BPSS and CPA instances as input and produces a graphical presentation of the collaboration as an output. The monitoring tool checks whether each message has the right sender and receiver and that they come in the right order. Further, each transaction may have a time constraint associated for its execution. Should the constraints be exceeded, the monitoring tool raises a flag that the collaboration has failed
- The Collaborative Content Checking tool enables specification and execution of content constraints that define valid syntax, structure, or semantics of the business messages. This facility allows standard developers, users, and implementers to precisely specify, extend, and test for conformance with, semantics of a common data dictionary (lexicon). The content tool allows a user to create his or her own profile and specify test cases using Schematron [6] expressions and using a Web-based interface. The test cases

are then stored in a customer's repository. Then another user (that presumably is engaged in a collaborative process) may select and execute the test case associated with a selected customer by posting an XML document.

- The Graphical Semantic Constraint Construction tool supports the Collaborative Content Checking facility. Manual encoding of syntactically valid content constraints is hard and tedious. This tool assists the user by offering an intuitive interface to construct the constraint specifications. The tool, which is motivated by natural language processing approaches, uses a set of classifications based on relatively low-level constraint semantics (such as cardinality, uniqueness, etc.) to guide the user [7].
- The Virtual Trading Partner aims to provide a reference implementation for a trading partner based on the ebXML BPSS specification. The user can utilise this tool to interact with the candidate system in a stepwise manner through a series of collaboration states. The tool generates finite state automata from a BPSS instance and uses it for internal consumption.

4. Testbed Interactions and Usage

Our interactions with manufacturing organisations and software vendors to implement validation testing services and interoperability pilots are illustrated by three case studies:

1. An interoperability pilot for an automotive retail consortium;
2. An inventory visibility tools integration pilot for an automotive supply chain; and
3. A message content validation for an enterprise integration effort.

4.1 An Interoperability Pilot for An Automotive Retail Consortium

STAR/XML is an effort within the Standards for Technology in Automotive Retail (STAR) consortium to define standard XML messages for dealer-to-OEM (Original Equipment Manufacturer) business transactions [8]. The STAR/XML initiative uses Electronic Business using eXtensible Markup Language (ebXML) Business Process Specification Schema (BPSS) specifications to represent scenarios of collaboration between OEMs and dealers/retailers.

To support the STAR/XML information exchanges, software vendors have included support for BPSS in their products. Both the users (i.e., OEMs and retailers) and the vendors desire to assess the functionality and interoperability aspects of these BPSS implementations. Users are asking, “Did we define the collaborations properly?” Vendors are concerned, “Do our implementations work when using real integration scenarios?” Both users and vendors pose the question, “Will the different vendor products interoperate when using the standards?”

The testbed offered an environment to pilot an interoperability testing effort with STAR/XML users and software vendors to start addressing the above questions. The goal of the STAR/XML BPSS interoperability pilot was to explore the following scenario:

1. Identify a real business collaboration between OEMs and retailers;
2. Analyse and model the collaboration using a modelling tool;
3. Define the XML documents and BPSS schema required by the collaboration;
4. Execute the collaboration using products from different vendors; and
5. Assess that the participating vendor products can interoperate.

We were able to take parts of the ordering process in the auto dealer environment and define the business scenario using OAGIS BODs and ebXML BPSS to specify the characteristics of the collaboration. The business process was modelled using a modelling tool, which also generated the specified BPSS schema [9]. The BPSS schema were loaded into execution engines from two distinct vendor products to validate that the schema, as generated by a third party modelling tool, is recognised by the B2B servers [10,11]. We were able to demonstrate that each of the BP engines properly executes the intent of the BPSS schema, within the scenario and that the BP engines interoperate for the given scenario.

The testbed pilot was able to make evident a number of interoperability issues, provide a venue to resolve those issues, and demonstrate that the participating software products interoperate for the given scenario. Without the testbed infrastructure actively supporting the pilot, the participants identified significant additional cost incurred by the pilot group.

The testbed pilot was successfully completed and provided insight into the BPSS modelling and execution process as needed by the STAR/XML process. In part, as a consequence of the pilot, the STAR/XML project team adapted the BPSS approach to support their specific needs while managing the complexity of the BPSS methodology. Figure 2 illustrates the STAR/XML interoperability pilot topology.

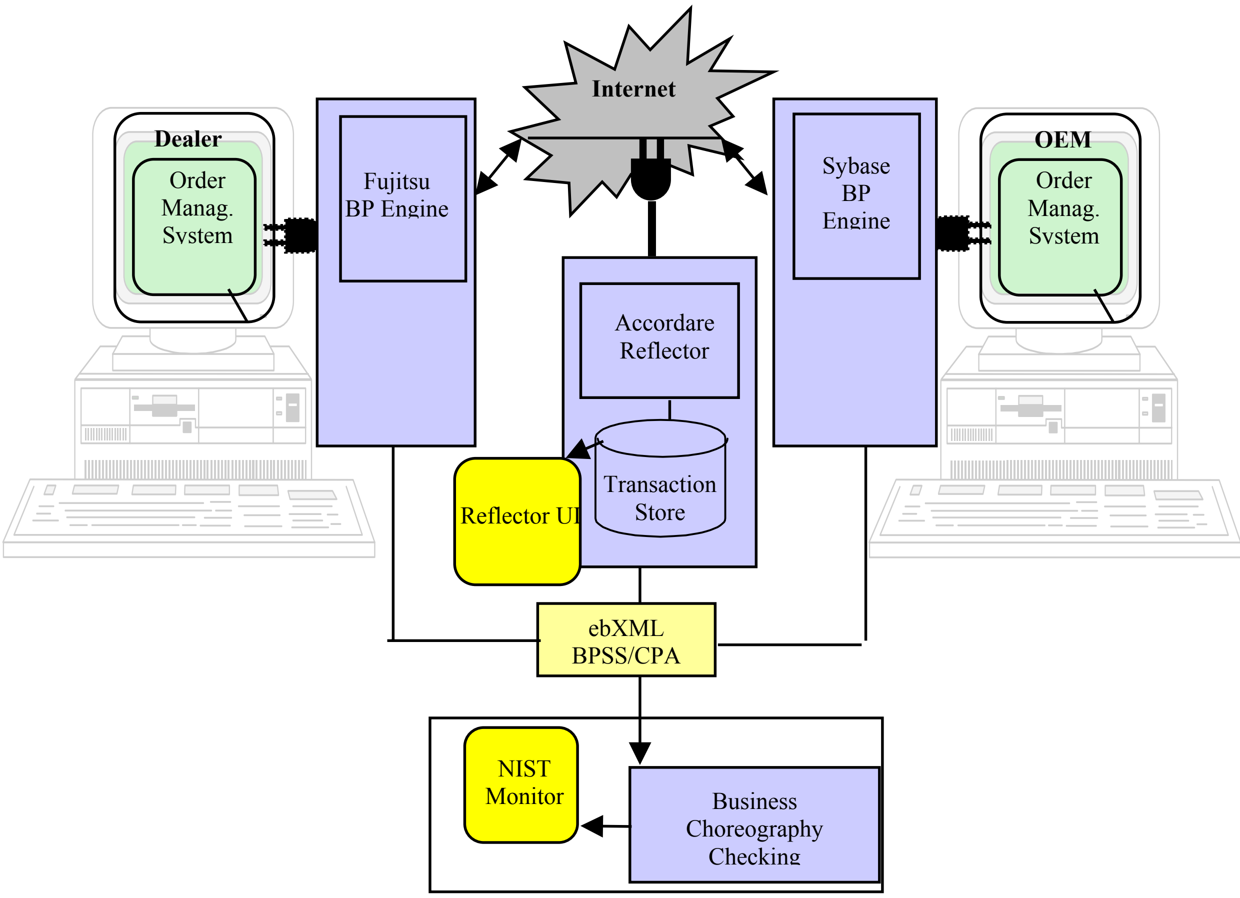

Figure 2: The STAR/XML interoperability pilot topology

4.2 – An Inventory Visibility Tools Interoperability Proof of Concept

The Automotive Industry Action Group (AIAG) is a North American automotive consortium that started the Inventory Visibility and Interoperability (IV&I) project to foster development of data exchange standards and B2B infrastructure enabling interoperable inventory visibility (IV) tools [12]. Currently, the automotive customer companies require suppliers to monitor customer inventory status and provide parts using IV tools that use proprietary data standards. Consequently, a supplier must use multiple IV tools to communicate with multiple customers. Once the IV data exchange standards are in place, the suppliers will use only one tool of their choosing to communicate with multiple IV tools on the customer side and, in this way, significantly cut the costs of training for, management, and operation of, the IV tools.

IV&I uses OAG BODs as their message content definition standard and is investigating alternative options for messaging and business process specification standards. One of the first steps in developing an interoperable data exchange standard is development of BOD schema and mappings from the alternative IV tool interfaces onto the BOD schema elements. IV software vendors participate in drafting the initial BOD designs. The automotive companies and vendors desire to validate correctness of these BOD designs in increasingly complex IV scenarios.

The goal for the IV&I Proof of Concept (POC) is to support and demonstrate the following scenario:

1. Identify the current and to-be IV business process for the initial MinMax scenario, analyse, and model the process using a modelling tool;
2. Define the BOD and collaboration schemas required by the business process to be executed during the data exchange; and
3. Show that two or more IV applications can collaborate with each other using the BOD and collaboration schemas.

We have completed the initial versions of the IV business processes, elements of the BOD schema, and preliminary mappings between two IV product interfaces and BOD elements. Additionally, we outlined collaboration schemas in support of data exchange that address two principal protocol alternatives: customer data push and supplier data pull.

These preliminary artefacts are provided for inspection and feedback to the automotive community with stake in the IV process. The comments received from the community will be factored into the versions of the artefact that will be used in the initial POC demonstration.

The POC demonstration will take place in two stages. In the first stage, the IV vendors will use the testbed to validate that their products can map onto the devised BOD schema and, consequently, interoperate with each other for the MinMax scenario. For example, the BOD elements such as Ship From, Ship To, Maximum Quantity, Quantity Inspection, and Quantity Blocked will be mapped on the IV product interface specifications by respective vendors to support specific functional requirements posed by customers and suppliers. In other words, the mapping of these elements on the product interfaces will necessarily be viewed in the context of some 'normative' functional scenarios to assure that tools capture the intended meaning of data. The testbed will provide access to the functional scenario definitions and validation mechanisms to assess the mappings of IV tools onto the BOD instances in support of the scenarios.

In the second stage of the POC demonstration, the vendors will select messaging and business collaboration execution protocols. Using these selected protocols, the vendors will exchange the messages using selected scenarios and exchanging data using the defined BOD schema instances using the agreed messaging protocols. The testbed will provide, again, the monitoring and validation mechanisms to assess the message exchange among the IV tools in the context of the functional scenarios.

4.3 – A Message Content Validation Pilot for Enterprise Integration

The United States Air Force (USAF) is conducting a large enterprise integration effort that uses OAG BODs to specify content format and semantics for messages exchanged among the parts suppliers and USAF enterprise portal. USAF has established a development process whereby independent BOD design groups create BOD schemas to enable data exchange for different functional areas such as cataloguing, auditing, and others. A BOD design quality assurance team was established to assess and ensure design quality of the BOD schemas. To effectively provide design quality assessment function, the testbed will provide validation testing of BODs:

- Schema validation using a W3C Schema Recommendation-compliant validating parser [15];
- Schema instance checking using the validating parser; and
- Manual and automated syntax and semantic checking using Schematron-based Content Checker.

The approach that was proposed includes the following stages:

- Uploading and schema design validation. In this stage, the schema is uploaded to the testbed Web-based interface with role-based access management and is run against a number of validation tests. The schemas will be run using a validating parser for XML grammatical conformance. Next, the schemas may be run through a quality checker to test for compliance with XML Schema semantics. Then, the schemas will be run through a schema design quality check. Based on these tests, the BOD design group will be provided with feedback on the schema quality. The development groups will use the feedback, for example, to normalise parts of the content model, which overlaps existing schemas. Once the provided schema passes the grammar, quality, and semantic checks, the resulting schemas will be stored on the web site and marked as "approved for pre-implementation testing".
- Pre-implementation testing. In this stage, the developer group should test the schemas approved from the design stage for additional requirements conformance. This step includes ensuring that the schemas can achieve the intended integration objectives and mapping(s) from supplier system(s) with no problem. A schema, which may go through some modification, are re-submitted to be uploaded to the Web based interface again and ensures that the schemas do not break the validation in the design stage. The process may be iterated between this step and the first step until approval. The approved schemas in this stage will be stored on the web site and marked as "approved for implementation testing". Approved schemas will be used as a canonical/reference model for later schema submission. During this stage, structural and semantic constraints will be identified. A tool is in development to support creation of these constraints. The constraints are represented in Schematron. These constraints are also submitted to be stored onto the web site
- Implementation testing stage. In this stage, the schemas are used within an integration implementation test run whereby BOD instances generated from participating applications are sent on the wire to the Web site for testing. Testing in this stage may extend to include testing of behaviour under exceptions. Schemas approved in this stage are marked "approved for implementation". The schemas can be download by the supplier user groups for actual production integration projects.

Differently from the previous two cases, the testbed will provide a repository of test cases, schema designs, and conformance requirements (i.e., test cases) to be accessed by BOD design groups as well as the design quality assurance group.

5. Ongoing Work

The ongoing testbed development activities can be seen as an effort to formalise lessons learned within a formal, rigorous testing framework and to evaluate new technologies within this framework on realistic manufacturing interoperability scenarios. There are four main areas of ongoing work:

- Infrastructure area comprises activities to assure that systems can communicate in a secure and reliable manner. Availability of multiple algorithms for security, authentication, and confidentiality for multiple transport protocols impact interoperability at the infrastructure levels. Presently, the testbed is collaborating with the ebXML IIC Technical Committee to pilot an ebXML Messaging Service test engine based on the test framework developed within this technical committee [13].

- Business document and content semantics area seeks to provide facilities to the standards developers, standards customers, and implementers to precisely specify, extend, and test for conformance with, semantics from the common data dictionary (lexicons). As content standards have been increasingly built with flexibility to support users in various industry sectors, formal semantics and structure requirements have been placed into separate layers of specifications and some are delayed until the standard implementation. In addition, most popular schema languages do not provide sufficient expressiveness to support accurate and precise semantic expression. These flexibility and expressiveness issues drive development in this area. We are currently developing additional tools in this to help users to define semantic constraints and schema quality assertions and to manage business document standard development and implementation lifecycle.
- Business process specification area involves activities to help represent, and test for, alignment of business states throughout collaboration; specify message choreography including validation of signals and actions; specify success, failure, exception, and timeout and retry conditions of business trading; and capture legal/security requirements. Currently, the NIST testbed team works with a Korean testbed team to enable an ebXML Business Process Specification Schema (BPSS) testing functionality. The teams are extracting test requirements for the ebXML BPSS and will use the test requirements to extend the IIC test framework for the BPSS testing in collaboration with the ebXML IIC Technical Committee in collaboration with the ebXML IIC Technical Committee.
- End-to-end integration area involves activities to ensure that business systems can achieve the goal of business integration and, consequently, it focuses on the business semantics of information exchanged in the collaboration.

6. Conclusions

The NIST Manufacturing B2B Interoperability testbed is a unique initiative to drive the advancement of interoperability testing and the state of the art in B2B integration for manufacturing enterprises. The testbed provides a neutral environment where industry partners meet, identify problems, and find solutions in collaboration with government and academia. On the other hand, the testbed provides a venue to perform research with the industry partners and academia and to address key industry issues by evaluating state-of-the-art technologies developed in laboratories.

During the initial phase of testbed development, we have worked with industry partners and gathered lessons learned that are reported within this paper. We are in the process of formalising these lessons learned into interoperability testing requirements to be implemented within a formal, rigorous testing framework. Initial tools developed to support monitoring and testing will continue to be developed within this testing framework. The combination of top-down, industry focused testing activities and bottom-up testing framework and tool development offers benefits to all stakeholders in the process of advancing state-of-the-practice and art in interoperable systems for the manufacturing enterprise.

7. Acknowledgments

The authors would like to acknowledge help of Tony Blazej, Hyunbo Cho, David Connelly, Jacques Durand, Simon Frechette, Deborah Hamill, Bill Harrelson, Mike Kass, Monica Martin, Hima Mukkamala, Steve Ray, Tim Sakach, Pat Snack and other individuals participating in, and supporting, the testbed developments.

8. Disclaimer

Certain commercial software products are identified in this paper. These products were used only for demonstration purposes. This use does not imply approval or endorsement by NIST, nor does it imply that these products are necessarily the best available for the purpose.

References

[1] OASIS ebXML Messaging Services Specification Technical Committee Web Site, accessed March 2003. *EbXML Messaging Service Specification version 2.0*. Available online via <http://www.oasis-open.org/committees/documents.php?wg_abbrev=ebxml-msg>.

[2] UN/CEFACT Web Site, accessed August 2002. *EbXML Business Process Specification Schema version 1.01*. Available online at < http://www.ebxml.org/specs/ebBPSS.pdf>.

[3] OASIS ebXML Collaboration Protocol Profile and Agreeement Technical Committee Web Site, accessed April 2003. Available online at < http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ebxml-cppa>.

[4] Open Application Groups Web Site, accessed March 2002. *Open Application Group Integration Specification version 8.0*. Available online via <http://www.openapplications.org/downloads>.

[5] Accordare Web Site, accessed March 2003. *The Reflector*. Available online at <http://www.accordare.com>.

[6] Schematron Web Site, accessed March 2003. The Schematron. Available online at < http://www.ascc.net/xml/resource/schematron/schematron.html>.

[7] Winograd, T. (1972). *Understanding Natural Language*. New York: Academic Press.

[8] STAR (Standards for Technology in Automotive Retail). *Making the Case for IT Standards in Retail Automotive*. STAR publication, 2003.

[9] Mega International Web Site, accessed August 2002. Business Process Modeling. Available online at <http://www.mega.com/us/product/megaprocess/>.

[10] Fujitsu Web Site, accessed August 2002. *Interstage Application Server*. Available online at <http://www.fsw.fujitsu.com/INTERSTAGE/index.html>.

[11] Sybase Web Site, accessed August 2002. *Enterprise Application Server*. Available online at <http://www.sybase.com/easerver>.

[12] Automotive Industry Action Group (AIAG) Web Site, accessed April 2003. Available online at < http://www.aiag.org/>.

[13] OASIS ebXML Implementation, Interoperability, and Conformance Technical Committee Web Site, accessed March 2003. *EbXML IIC Test Framework version 1.0.* Available online at < http://www.oasis-open.org/committees/documents.php?wg_abbrev=ebxml-iic>.

[14] W3C Web Site, accessed March 2003. *HTTP – Hypertext Transfer Protocol*. Available online at < http://www.w3.org/Protocols/>.

[15] W3C Web Site, accessed March 2003. *XML Schema*. Available online at < http://www.w3.org/XML/Schema/>.