Proceedings of the 2006 Winter Simulation Conference
L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, eds.

ASSESSMENT OF THE NIST SHOP DATA MODEL AS A NEUTRAL FILE FORMAT

Greg Harward

ProModel Corporation
556 East Technology Ave
Orem, UT 84097

Charles Harrell

Manufacturing Engineering and Technology
Brigham Young University
Provo, UT 84602

ABSTRACT

This paper evaluates the shop data model (SDM) being developed by the National Institute of Standards and Technology (NIST) in terms of its viability as a neutral file format (NFF) for the discrete-event simulation (DES) of manufacturing systems. ProModel simulation software served as the test case for this evaluation. Observations are also provided regarding the challenges that simulation vendors might encounter when implementing the proposed NIST SDM. This paper shows that the NIST SDM doesn't pose any limitations which would prevent it from syntactically representing a manufacturing simulation model, however, it is not without certain challenges and difficulties. While only 28% of the ProModel data elements are currently supported by the SDM, future enhancements to the SDM should allow the information model to serve as a foundation upon which a common information model and NFF for the DES industry could be built.

1 INTRODUCTION

In the simulation industry there are at least a dozen software companies that market discrete-event simulation software to manufacturing companies for purposes of systems planning and process improvement. While each simulation product provides a basic common core of modeling capabilities, each also includes certain unique modeling constructs and stores model data in its own proprietary data format. The model file type for each product also varies (binary, ASCII, .XLS, XML, etc.). This makes exchanging data to and from simulation models extremely challenging. At the 2003 Winter Simulation Conference it was stated, "While the art of simulation continues to flourish, the continuity and the consistency of the simulation data usage may have not kept pace" (McLean 2003). "Hence, it takes real effort when exchanging simulation models during scenario iterations among project participants" (Lu 2003).

One proposed solution to the problem of exchanging data between dissimilar simulation model files is to develop a neutral file format (NFF). We have already seen the widespread use of NFFs in the CAD industry. Though not without its problems, a CAD NFF makes CAD data exchange practical and sometimes (not always) painless. Similar benefits (and challenges) can be anticipated in the deployment of NFFs in the simulation industry. "The development of neutral, vendor-independent data formats for storing simulation models could greatly improve the accessibility of simulation technology to industry by enabling the development of reusable models" (McLean 2002). This type of solution would help "make simulation technology more affordable and accessible to a wide range of potential industrial users" (Lee 2003).

As a step toward the creation of an NFF, "NIST has been developing an information model in an extensible markup language (XML) exchange-file format that facilitates the exchange of information between manufacturing simulation applications and other manufacturing applications and/or data sources" (McLean 2002). This information model is designed to conceptually outline model information in an XML schema or XML Schema Definition (XSD). Utilizing this XSD, the contents of an information model could be exported to a portable NFF using extensible markup language (XML) syntax. The purpose of this paper is to assess the ability of the NIST XSD information model, also referred to as the NIST SDM (Shop Data Model), to conceptually represent simulation model information. This study was performed using the latest available draft of the specification dated February 24, 2003. (McLean 2003). This evaluation also includes a test of the inherent capability of the XML syntax to represent simulation model information syntactically.

Four specific questions were addressed in the study:

1. Can the data elements found in a typical DES product be represented using XML syntax?
2. What is involved in mapping these elements to the NIST XSD?
3. What percentage of these elements can be successfully represented in the NIST XSD?

1-4244-0501-7/06/$20.00 ©2006 IEEE

4. What challenges might simulation vendors encounter when working with the proposed NIST XSD?

2 NEUTRAL FILE FORMATS IN SIMULATION

In recent years several efforts have been made to define neutral file formats for use in DES. Some of these have been industry-wide while others have been company-wide. Some have been actually implemented while others were only proposed. Some have been commercially driven while others have been academic or government promoted. Here we look at a few of these developments.

2.1 Simulation Data eXchange (SDX)

The Simulation Data eXchange (SDX) file format (not to be confused with XSD) is an NFF that has been successfully used in order to improve data exchange issues between dissimilar simulation models. Originally developed by Shreekanth Moorthy of Engineering Animation, Inc, the SDX file format was intended to be open source in nature and enable the transfer of DES model information via an NFF. Currently, only about four simulation companies support SDX for import functionality and only FactoryCAD, a product now sold by UGS-Tecnomatix, supports the creation of the SDX file format for the purpose of exporting.

While originally intended to serve as an NFF for the DES industry, usage of the SDX file format has primarily been limited to the transfer of routing information and environment layout information such as equipment, machines, and conveyors with their associated breakdown, cycle time, and scrap rate information from FactoryCAD software to one of the handful of SDX supporting DES software packages. With this narrow focus, the SDX file format has been used as an efficient means of providing initial simulation data extracted from factory layout information. On a larger scale, the acceptance of the SDX file format as a viable NFF in the DES industry has been limited. This is due to the inability of the SDX file format to transfer all of the information required by DES software packages, the necessity of creating custom software translators before it can be used, and limited support for the file format from industry vendors (Kim 2003).

The SDX file format has contributed to the formation of an NFF for the DES industry by supplying specifications for representing layout elements. Although it currently supports an XML format, it still falls short as a comprehensive NFF solution for the DES industry. For this reason more extensive NFF solutions are now being explored.

2.2 Open SML an Open Ssource NFF for DES

The LINUX (Linux 2005) computer operating system has thrived under the open source development paradigm. Open source software development allows for any member of the open source community to contribute to the body of computer source code while being governed by a set of organization rules and a governing central body. The net result of this type of development is a product which is royalty free because it is the culmination of free member contributions. It is also flexible as it can be changed by any contributing member. The concept of open source software development has been applied to the problem of developing an NFF for the DES industry (Wiedemann 2002).

During the 2001 Winter Simulation Conference, OpenSML-project was presented. This project focuses on an open source development approach in order to create a Simulation Modeling Language (SML) as a method to supply common core simulation functionality for the DES industry. Originally developed as a Java-based programming language, the program is set up to leverage the open source development process in order to develop a standard DES language that is readable, modular, and extensible (Kilgore 2001).

2.3 Simulation Reference Markup Language (SRML)

Simulation Reference Markup Language (SRML) is an XML- based markup language developed at Boeing for representing simulation models. The language is used in combination with the Simulation Reference Simulator (SR Simulator), also developed at Boeing, to run SRML simulation models. SRML and SR Simulator have been developed with the goal of leveraging existing internet technologies in order to overcome existing simulation model obstacles. With this goal in mind, SR Simulator was developed as a plug-in that can be used in any internet enabled application or other supporting application to permit the execution of simulations in many different host applications. Using SR Simulator as a plug-in to a popular internet browser, such as Microsoft Internet Explorer, provides the added benefit of instantly inheriting the additional capability provided by embedded internet communication protocols. This approach enables the running of simulation models on the same computer or across a network in a distributed fashion.

The creation of simulation models using SRML is patterned after the process of web page authoring. SRML syntax includes the use of concepts such as embedded logic defined in javascript, vbscript, or another simulator-supported language, item classes which support hierarchical containment relationships, item quantities for describing large nested numerical items, links for use in referencing external items, and external files to describe simulated item behavior in a separate stand alone file (Reichenthal 2002).

The development of SRML is an example of how XML is being used to represent simulation model information in an XML file format. Wisely, this implementation is developed to leverage existing proven internet technologies in order to benefit the simulation industry.

2.4 The NIST Shop Data Model (SDM)

Within NIST there is a department dedicated to addressing the information interface needs of the U.S. manufacturing community. The System Integration of Manufacturing (SIMA) group works to:

1. Develop information exchange and interface protocols to address manufacturing integration problems.
2. Establish test mechanisms for validating protocols and implementations.
3. Transfer information technology solutions to manufacturing enterprises.

The goal of SIMA is to create a collection of manufacturing specifications which are then reviewed by industry until they mature into an authoritative specification. Each of these specifications is referred to as Initial Manufacturing Exchange Specifications (IMES). One such IMES that SIMA has developed is called the Shop Data Model (SDM) and Interface Specification. This specification is presented in two formats which are Unified Modeling Language (UML) and Extensible Markup Language (XML). The goals for the SDM are:

1. The model will be used to support the integration of a manufacturing execution system, a production scheduling system, and a prototype machine shop simulator.
2. The model will be promoted as a standard data interface for manufacturing simulators (Lee 2003).

The NIST SDM is an XSD information model consisting of fifteen major categories with four major supporting structures. As shown in Figure 1, the categories include organizations, calendars, resources, skill definitions, setup definitions, operation definitions, maintenance definitions, layout, parts, bills of materials, inventory, procurements, process plans, work and schedules (McLean 2003). The supporting structures include time sheets, probability distributions, references and units of measure.

Figure 2 shows a conceptual diagram of the NIST XSD and interface specification.

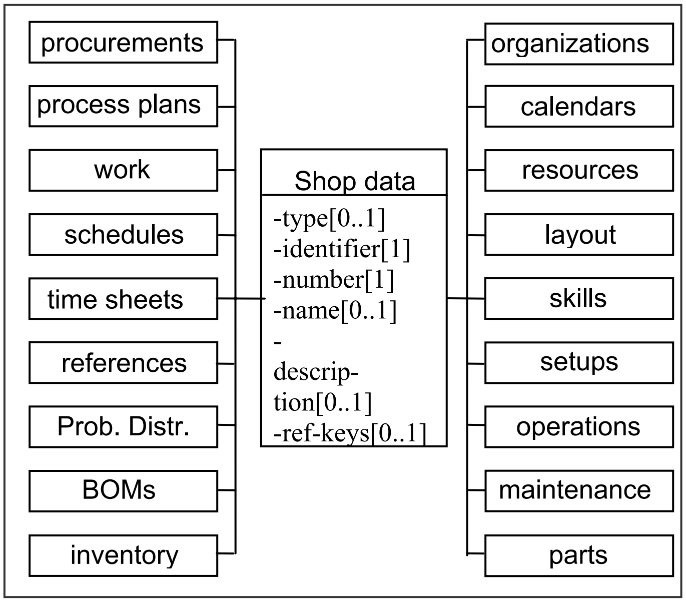

Figure 1: Top Level Diagram of the NIST XSD and Interface Specification (Lee 2003)

3 EVALUATION PROCEDURE

To evaluate the suitability of the NIST SDM as an NFF for the DES industry, a file translator was manually created and used to test which elements of a sample vendor's information model are supported by the NIST SDM. As it turned out, the process itself of manually creating a file translator revealed much about the ability that the NIST SDM has in supporting model elements from a simulation vendor's information model.

Assessing the compatibility of commercial simulation software file formats with the NIST SDM format would be relatively straightforward if, on the commercial side, model data schema were readily available. Unfortunately, few, if any, simulation software vendors have published complete data models for their software. Consequently, this investigation necessitated the construction of a vendor's data model as well as comparing the data model to the NIST SDM. ProModel Corporation, a leading software vendor in the DES industry since 1988, was chosen as the test case for this evaluation. Currently, ProModel software utilizes a binary file format for storing simulation model information because it is compact and efficient. Though a direct conversion from the binary file to the NIST SDM could have been developed, to provide better documentation and make the process more understandable, a ProModel XSD was created first.

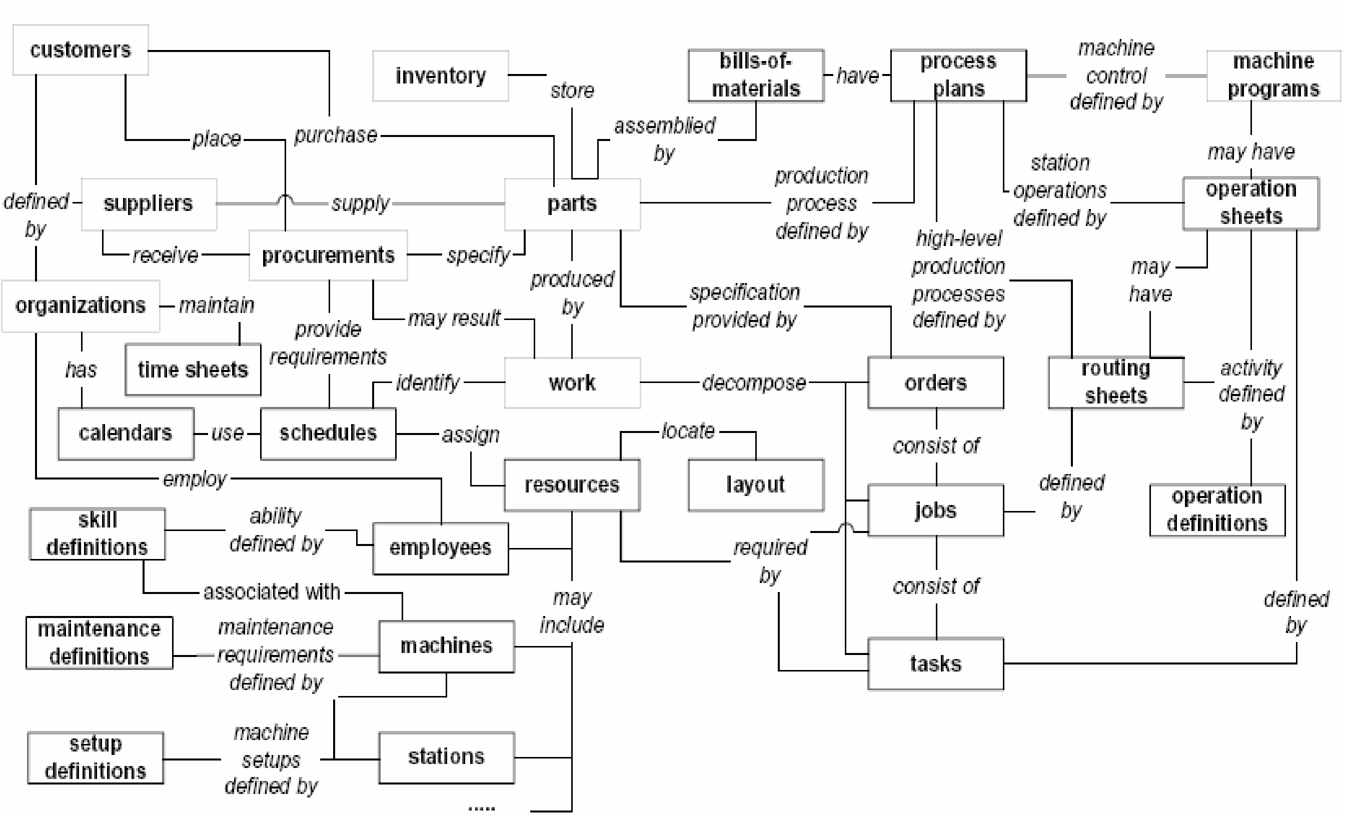

Figure 2: Conceptual Diagram of the NIST XSD and Interface Specification (Lee 2003)

Once an XSD representation of the ProModel information model was manually created, it was classified into essential and extraneous elements in order to narrow down the group of available elements to only those that belong in a common information model. This classification was necessary as some ProModel elements are unique and therefore do not belong in an NFF. Next, a mapping comparison between the essential ProModel XSD elements and the NIST XSD information model elements was performed in order to evaluate compatibility. The results of this comparison provide at least one data point for evaluating the viability of the NIST SDM as an NFF for the DES industry.

After the mapping process was completed, each of the elements was organized into one of three categories according to its ability to support the concept and content of the original ProModel simulation model. The three categories utilized for this classification were Supported, Unsupported but Essential, and Unsupported but Unessential.

4 RESULTS AND ANALYSIS

Figure 3 displays the results of the mapping process with a column that describes the support for each element. When not supported, elements are categorized into one of four incompatibility classifications. The four unsupported element categories are: Basic Data Elements, Decision Logic, Basic Software Programming Elements, and Layout Related Elements. These categories are identified with the support classifications "No-Basic", "No-Logic", "No-Program", and "No-Layout" respectively.

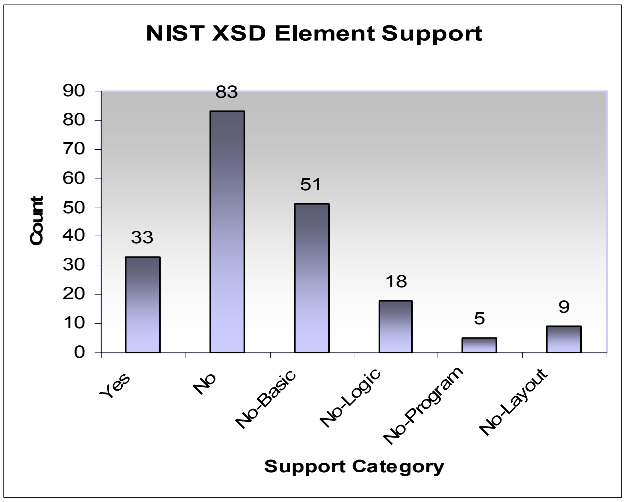

Figure 3: NIST XSD Support for ProModel Data Elements

In attempting to map the ProModel XSD elements which are classified as vital and meet the criteria as valid NFF elements, 28.5% of the elements currently have valid representation contained in the current NIST XSD information model. This means that 71.5% of the elements do not currently map to the NIST XSD.

A total of eighty-three ProModel elements make up the 71.5% of unmapped elements. A summary of each of these categories follows.

Fifty-two of the elements that are unsupported by the NIST XDS information model fall into the category of Base Data Elements. Support for elements in this category is possible and can be accomplished with the addition of elements which contain basic XML data types.

Sixteen of the ProModel elements were not supported due to the lack of support for decision logic. According to NIST engineer Frank Riddick, NIST plans to support decision logic in the next release of the SDM specification. The exact format for supporting such logic remains to be determined; however, representation for this logic can be achieved utilizing several different approaches which include the use of vbscript, javascript, or other machine readable languages embedded directly into XML tags as described in the referenced article *Reintroducing Web-Based Simulation* (Reichenthal 2002). Additionally, a simplified approach could be used to supply model logic in the form of strings that would then be parsed and compiled by the host application after being read. A third option is for the XML document to reference external files which contain precompiled code. This option would save the time associated with parsing and recompiling machine readable code while at the same time simplifying the XML syntax.

Five of the unsupported elements are what have previously been referred to as Basic Software Programming Elements. These programming elements are used to provide general functionality which is not related to any particular modeling paradigm and serve to open up possibilities for additional flexibility for simulation applications. While adding decision logic, it would make sense to also add the five Basic Programming Elements to support items in this category as they are commonly used in cooperation with decision logic. The five Basic Software Programming Elements consist of Variables, Attributes, Arrays, Macros, and Subroutines.

Eight of the unsupported elements are related to the unfinished state of the Layout portion of the SDM specification. According to Frank Riddick of NIST, the Layout portion was under development when the decision was made internally to change the approach and align the work more closely with the paradigm used in the SDX file format. In order to support the ProModel methodology, the concept of path network nodes or some equivalent structure must be supported in the Layout portion of the SDM specification.

5 CONCLUSIONS

As a result of this evaluation, the following conclusions can be drawn:

1. The process of creating the ProModel XSD and mapping model elements from the ProModel paradigm over to the NIST SDM paradigm demonstrates that XML did not pose any syntax related limitations which would invalidate the ability of XML to represent a common information model or associated XML NFF candidate.
2. The results of performing the process of mapping ProModel model elements to NIST XSD elements are that less than a third of the ProModel elements are currently supported by the NIST SDM specification. Therefore, utilizing ProModel as the test case, the NIST XSD does not yet support all of the required elements to effectively serve as an NFF.
3. The most challenging elements to map to the NIST SDM are elements which are specific (and often idiosyncratic) to a particular simulation product (e.g., levels of preemption) and special logic statements related to a particular product's functionality, such as capturing multiple alternative resources with different priorities and usage times.

This study shows that full implementation of the NIST XSD (or any NFF for that matter) is not trivial for either NIST or for simulation vendors. There will likely be partial implementations initially that will convert a percentage of elements and then require some manual touch-up work to complete the model conversion.

This evaluation of the NIST XSD is based on one simulation vendor's data elements. Other simulation products may be more or less challenging depending on the accessibility of their data models and their congruency with the NIST SMD. The results of this evaluation, along with the description of the steps required to convert an existing information model over to the information model presented by NIST, provide at least a reference point for other DES simulation vendors with similar data requirements. This reference point can be used to make an informed decision regarding the effort required to support the proposed NIST NFF specification.

REFERENCES

Harrell, C., Bateman, R. E., Gogg, T. J., & Mott, J. R. A. (1997). *System Improvement Using Simulation.* ProModel Corporation. (p.1-20).

Harward, G. (2005) *Suitability of the NIST Shop Data Model as a Neutral File Format for Simulation.* Thesis Brigham Young University, 2005. Available at: http://contentdm.lib.byu.edu/ETD/image/etd899.pdf.

Kilgore, R. A. (2001). Open Source Simulation Modeling Language (SML). In *Proceedings of the 2001 Winter Simulation Conference*, ed. B.A. Peters, J.S. Smith, D.J. Medeiros, and M.W. Rohrer.

Kim, K. (2003) *The Simulation Data Exchange (SDX) File Format: A Useful Neutral File format (NFF) For Discrete-Event Simulation.* Thesis Brigham Young University, 2003.

Lee, T., McLean, C., & Shao, G. (2003). A Neutral Information Model for Simulating Machine Shop Operations. In *Proceedings of the 2003 Winter Simulation Conference*. ed. S. Chick, P. J. Sanchez, D. Ferrin, and D.J. Morrice.

Linux (n.d.). available via <www.linux.org> [accessed March 10, 2006].

Lu, R., & Qiao, G. (2003). NIST XML Simulation Interface Specification at Boeing: A Case Study, In *Proceedings of the 2003 Winter Simulation Conference*, ed. S. Chick, P. J. Sanchez, D. Ferrin, and D. J. Morrice.

McLean, C., & Leong, S. (2002). A Framework for Standard Modular Simulation. In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yucesan, C. –H. Chen, J.L. Snowdon, and J. M. Charnes.

McLean, C., Lee, Y. T., Shao, G., Riddick, F., & Leong, S. (2003). Shop Data Model and Interface Specification. Draft NISTIR. Gaithersburg, Maryland: National Institute of Standards and Technology.

McLean, C., Leong, S., Harrell, C., Zimmerman, P., & Lu, R. (2003). Simulation Standards: Current Status, Needs, and Future Directions, In *Proceedings of the 2003 Winter Simulation Conference*, ed. S. Chick, P.J. Sanchez, D. Ferrin, and D. J. Morrice.

ProModel Corporation. (2004). *ProModel User Guide* Utah.

Reichenthal, S. (2002). Re-Introducing Web-Based Simulation, In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yucesan, C. –H. Chen, J.L. Snowdon, and J. M. Charnes.

Wiedemann, T. (2002). Next Generation Simulation Environments Founded On Open Source Software And XML-Based Standard Interfaces, In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yucesan, C. –H. Chen, J.L. Snowdon, and J. M. Charnes.

AUTHOR BIOGRAPHIES

GREG HARWARD is a simulation consultant for ProModel Corporation where he previously held positions in including software testing, development, support, and training. He received an M.S. degree with a Business minor in Manufacturing Engineering Technology from Brigham Young University in 2005. He resides in Austin, Texas with his wife and two children. His e-mail address is <GregH@promodel.com>.

CHARLES HARRELL is an associate professor of Manufacturing at Brigham Young University and founder of PROMODEL Corporation. He received an M.S. in Industrial Engineering from the University of Utah, and Ph.D. in Manufacturing Engineering from the Technical University of Denmark. He was an engineer at Ford Motor Company and Eaton Corporation and has thirty years of experience in simulation co-authoring several books on simulation. He serves on the board of directors of PROMODEL Corp. He is a Senior member of IIE and SME. His e-mail address is <harrellc@byu.edu>.