

---

# The interface development for machine shop simulation

---

**Yan Luo\* and Y. Tina Lee**

Manufacturing Engineering Laboratory  
National Institute of Standards and Technology  
Gaithersburg, MD 20899-8260, USA  
Email: luoyan@cme.nist.gov  
Email: leet@cme.nist.gov  
\* Corresponding author

**Abstract:** Modelling and simulation technology is recognized for facilitating training, reducing production cost, improving product quality, and shortening development time. However, this technology remains largely underutilized by industry today. This is because custom simulator development is complex and costly, and custom translators are needed to run commercial simulation software. Information models and standard interfaces could help address these problems. A machine shop information model, developed at the National Institute of Standards and Technology (NIST), provides data interfaces for integrating machine shop software applications with simulation. The interfaces include *organizations, calendars, work, resources, schedules, parts, process plans, and layout* within a machine shop environment. The model is represented by Extensible Markup Language and Unified Modelling Language. This paper briefly presents the machine shop information model, introduces a data transfer mechanism between the machine shop database and XML document, and discusses data translators and data import/export.

**Keywords:** modeling; interface; simulation.

**Biographical notes:** Yan Luo is a contractor at NIST. His research interests include CAD/CAM, IT, software and manufacturing engineering. He is an Associate Editor of the International Journal of Agile Manufacturing. He has more than ten years of experience in software and manufacturing engineering. He has published more than 40 papers in academic journals and scientific conferences.

Y. Tina Lee is a computer scientist in the Manufacturing Simulation and Modeling Group at NIST. She joined NIST in 1986. Her major responsibility in recent years has been to develop information models to support various manufacturing application areas. Previously she worked at Contel Federal Systems and Sperry Corporation. She received her BS in Mathematics from Providence College and MS in Applied Science from the College of William and Mary.

---

# 1 Introduction

Extensible Markup Language (XML) is a simple, flexible text format derived from the Standard Generalized Markup Language (SGML, ISO 8879). An information model is a representation of concepts, relationships, constraints, rules, and operations to specify data semantics for a chosen domain of discourse (Lee, 1999). A machine shop information model has been developed at the National Institute of Standards and Technology (NIST) (McLean et al., 2005; Lee et al., 2003) and is described in both a graphical form in Unified Modeling Language (UML) and a textual form in XML. The objective of the information modeling effort is to develop a standardized, computer-interpretable representation that allows the exchange of information in a machine shop environment. The model supports data requirements for the entire manufacturing life cycle, enables data exchange between simulation and other manufacturing software for machine shops, and supports testing and evaluation of machine shops' manufacturing software.

A database is a collection of related information. It provides a structured method for storing and querying data. Most existing databases are relational databases. A database management system (DBMS), such as Microsoft Access or Oracle, provides software tools for users to organize data in a flexible manner. A database supporting the machine shop information model has been developed (Luo and Lee, 2003; Luo, 2000). The database, described in this paper, has been developed using Microsoft Access, which has the capability to import and export data with various formats and data access interfaces, such as Data Access Objects (DAO), Open Database Connectivity (ODBC), and Dynamic Data Exchange (DDE).

## 2 Machine shop information model

This section briefly discusses the machine shop information model. The primary objective is to develop a structure for exchanging shop data between various manufacturing software applications, including simulation. The idea is to use the same data structures for managing actual production operations and for simulating the machine shop.

### 2.1 Information model description

The model contains twenty major manufacturing data elements. The name along with a brief description of each major element is described as follows.

- *Organizations* is used to maintain the organizational structure, contacts, and address information for the manufacturing organization and its customers and suppliers.
- *Calendars* identifies the shift schedules that are in effect for a period of time, breaks, and holidays.
- *Resources* describes all the resources that may be assigned to tasks in the shop. The resource types available in the machine shop environment include: stations and machines, cranes, employees, and tool and fixture catalog items.
- *Skill-definitions* lists the skills that an employee may possess and the levels of proficiency associated with those skills.
- *Setup-definitions* typically specifies tool or fixture setups on a machine. Tool setups are typically the tools that are required in the tool magazine. Fixture setups are work holding devices mounted on the machine. Setups may also apply to cranes or stations.
- *Operation-definitions* defines the operations that may be performed at a particular station or group of stations in the shop.
- *Maintenance-definitions* defines preventive or corrective maintenance to be done on machines or other maintained resources.
- *Layout* defines the location of reference points within the shop, area boundaries, paths, resources, and part objects.
- *Parts* provides elements for part specifications, group technology codes, customers, suppliers, as well as links to bills of materials, process plans, drawings, part models, and other references.
- *Bills-of-materials* cross-references the parts and quantities required in a hierarchical bill-of-materials structure.

- *Inventory* identifies the instances and locations for parts, materials, tools, and fixtures inventory.
- *Procurements* identifies the external purchases that have been created to satisfy the part inventory and manufacturing requirements.
- *Process-plans* specifies a set of process plans that are associated with production and support activities for a particular part or parts. A process plan has routing sheets and operation sheets that correspond to the job and task level in the work hierarchy.
- *Work* is used to specify a collection of a hierarchy of production orders, jobs, and tasks. It is also used to specify a collection of internal support orders for maintenance activities, inventory picking, and tool preparation.
- *Schedules* lists planned assignment or mapping of work to resources and resources to work.
- *Revisions* specifies information about a set of revisions of the subjects. Information included in the element is each revision's identification, description, date, creators, etc.
- *Time-sheets* provides a list of individual time sheet elements. A time sheet is used to log the hours that an employee has worked, the time an employee has taken off from work, and accrual of leave hours.
- *Probability-distributions* specifies distributions that are used to vary processing times, breakdown and repair times, and availability of resources, etc.
- *References* describes the information about reference materials that support or further define the data elements contained within the shop data structure.
- *Units-of-measurement* describes various measurement units used in the file, for example, the measurement may be for distance, speed, mass, time duration, or currency.

## 2.2 Schema description

An XML schema describes the structure of an XML document. The purpose of an XML Schema is to define the legal building blocks of an XML document. It defines elements, attributes, the elements' child elements, the order and number of child elements, data types, default and fixed values for elements and attributes, etc. The schema for the machine shop information model is being developed using the XML Schema Language (XML Schema Definition (XSD)). The schema defined for the machine shop information model is presented in Figure 1.

**Figure 1** Schema for the machine shop model

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:include schemaLocation="common.xsd"/>
  <xs:include schemaLocation="organizations.xsd"/>
  <xs:include schemaLocation="calendars.xsd"/>
  <xs:include schemaLocation="resources.xsd"/>
  <xs:include schemaLocation="setup-definitions.xsd"/>
  <xs:include schemaLocation="skill-definitions.xsd"/>
  ...
  <xs:element name="shop-data" type="shop-data"/>
  <xs:complexType name="shop-data">
    <xs:annotation>
      <xs:documentation>Machine shop data
structure</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element minOccurs="0" ref="units-of-measurement"/>
      <xs:element minOccurs="0" ref="organizations"/>
      ...
    </xs:sequence>
    <xs:attributeGroup ref="id.attrs"/>
  </xs:complexType>
  <xs:element name="units-of-measurement" type="units-of-
measurement"/>
  <xs:element name="organizations" type="organization-directory"/>
  <xs:element name="calendars" type="calendars"/>
  <xs:element name="resources" type="resources"/>
  ...
</xs:schema>
```

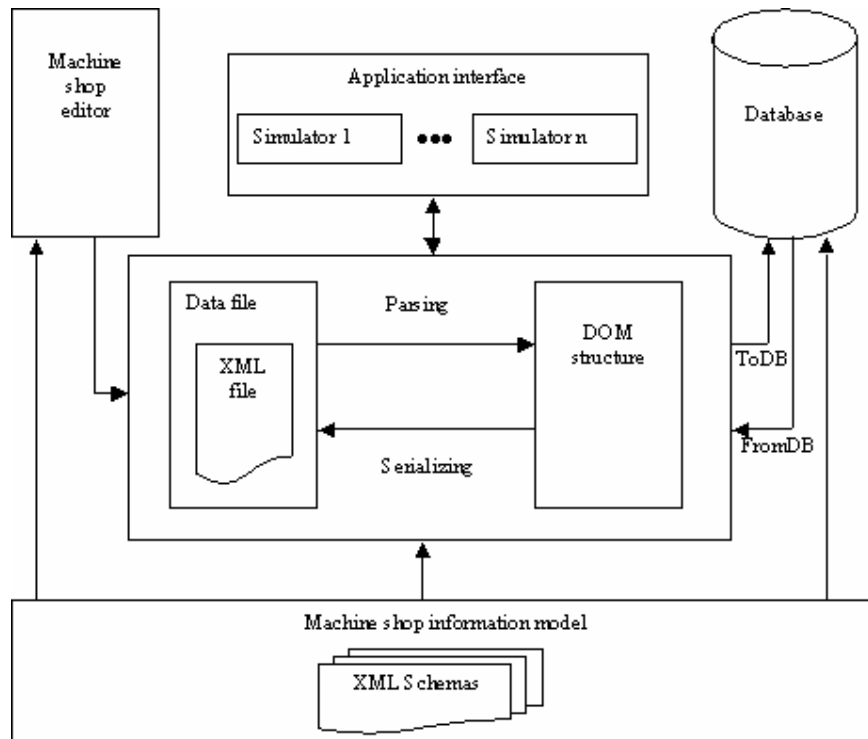
### 3 Data exchange

Information consistency is required for effective data exchange between different data formats. The machine shop information model describes data requirements for the machine shop environment.

#### 3.1 Architecture description

NIST proposed an architecture for a generic data-driven machine shop simulator, and has been constructing a prototype simulator based on the architecture using commercial off-the-shelf software (McLean et al., 2002). The architecture for the generic machine shop simulation is divided into the following elements: neutral shop data file, XML data processor, system supervisor and reporting, machine shop emulator, discrete event simulator, and user interface system. Figure 2 shows a framework for machine shop simulation. The machine shop information model is a key factor in effectively and efficiently integrating the generic machine shop simulator. The information model/XML schema is used as a neutral data format for representing and exchanging machine shop data. A data exchange prototype is developed to integrate and share information of different representation formats. The development is based on widely accepted standards: interface for database tables query, DOM (Document Object Model) for XML document operation, and XPath (XML Path Language) for nodes query.

**Figure 2** The framework for machine shop simulation



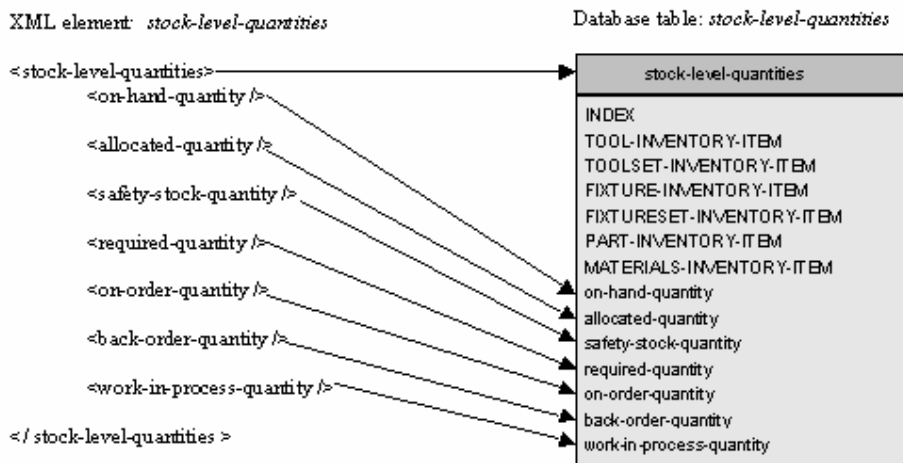
Simulators provide the support for manufacturing activities such as design, process, manufacturing, assembly, and production management. The XML parsers and DOM structure serializers are integrated in the program. Machine shop editor, ToDB, and FromDB provide interfaces of data import/export based on the machine shop information model.

### 3.2 Data mapping mechanism

The objectives of the database implementation effort include: demonstrating the feasibility of the information model, developing a pilot database system and then migrating to a database management system, and supporting the integration of manufacturing applications and simulations used in machine shops.

This section shows the mapping mechanism from an XML element to a relational table via a sample. A database model is designed to map onto the machine shop information model. The database contains a set of relational tables presented in a tree shape structure. Tables comprise the fundamental blocks of a database. A table is a grouping of selected data organized into fields (columns) and records (rows) on a datasheet. A field identifies a data type for a set of value in a table while a record stores a set of values defined by fields. Elements in the information model are used to describe fundamental features or common features of machine shop data. They are represented as tables in the database model. Figure 3 is a sample mapping between an XML element and a table. *Stock-level-quantities* specifies the quantities of tools, fixtures, materials, or parts for various purposes and is represented by the following data elements: *allocated-quantity*, *back-order-quantity*, *on-hand-quantity*, *on-order-quantity*, *required-quantity*, *safety-stock-quantity*, and *work-in-progress-quantity*. *TOOL-INVENTORY-ITEM*, *TOOLSET-INVENTORY-ITEM*, *FIXTURE-INVENTORY-ITEM*, *FIXTURESET-INVENTORY-ITEM*, *PART-INVENTORY-ITEM* and *MATERIALS-INVENTORY-ITEM* are parent-elements of *stock-level-quantities*. *INDEX* is used for system purpose.

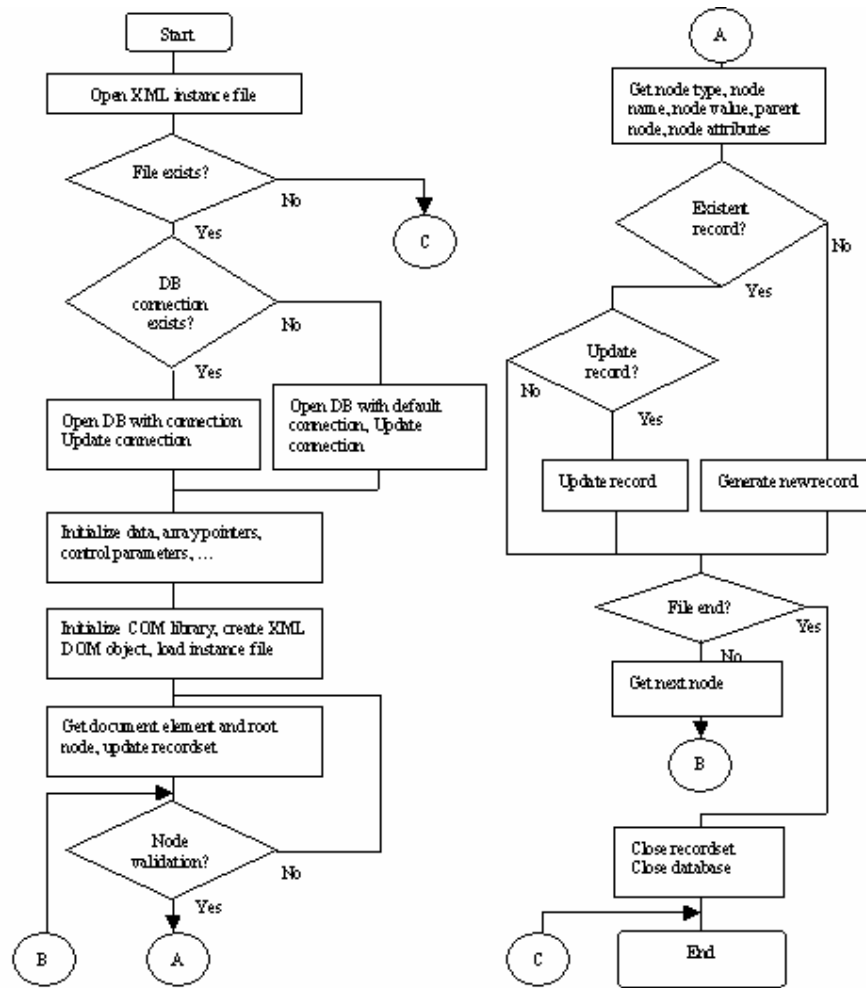
**Figure 3** Element mapping



### 3.3 Importing XML data to database

A prototype implementation has been developed to import XML data into the database. Four function modules, including loading XML documents, validating data files based on the schema appointed, initiating and generating DOM structures, connecting and closing the database, and generating and updating database records, have been developed to transfer data from XML documents to a relational database. The implementation is based on an object-oriented method. A set of classes is designed to complete the file preparation, data generation, database updating and so on. A control class is defined to set up connection for database, open data file, judge conditions, initiate data, initiate DOM structure, enquiry nodes, get node's parameters, generate database records, and update records, as shown in Figure 4.

**Figure 4** Importing XML data to machine shop database



## 4 Application of information model

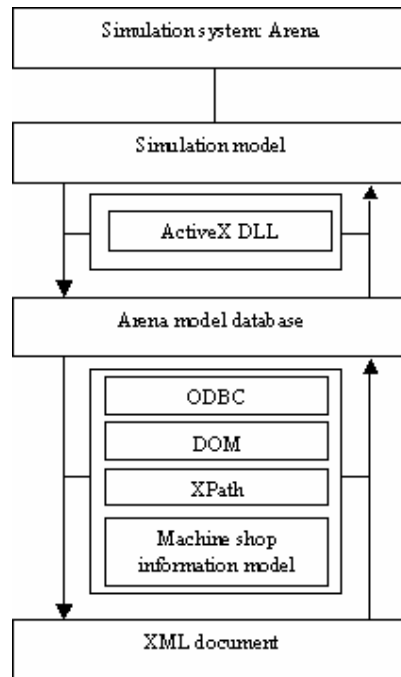
The simulation application, based on the machine shop information model, is introduced in this section.

### 4.1 Application for Arena

Rockwell Automation's Arena is used for various environments from customer service to manufacturing, it can create a basic model, refine model, simulate model, and analyze simulation results. As the Arena data format of simulation model (.DOE) is proprietary, a set of data process utilities has been developed to support data transfer between the Arena model database and XML data document. Figure 5 presents an architecture for an Arena application. The utilities developed using ActiveX Dynamic Link Libraries (DLL), DOM, XPath, and ODBC database engine contain four main modules:

- generating Arena model database from XML document: this module opens an XML data document, which is previously validated by the machine shop schemas, and loads the XML document data into an Arena model database.
- generating simulation model: this module reads an Arena model database, and dynamically generates a simulation model in Arena.
- generating Arena model database from simulation model: this module exports simulation model data to an Arena model database.
- generating XML document: this module generates an XML data document from an Arena model database.

**Figure 5** The application architecture for Arena

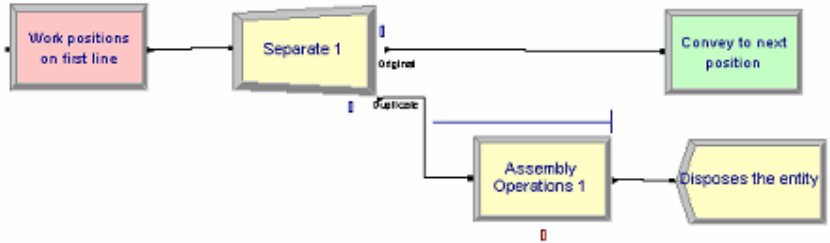


To demonstrate these capabilities, we selected the truck assembly line model from the Arena sample models library. The new truck chassis enter the line at a constant rate. They are conveyed down the line from one work position where five processes are performed: *arrival*, *assembly-line-1*, *turn*, *assembly-line-2*, and *exit*. Each process has two to fourteen activities. The *arrival* process executes every 9.5 minutes. The *assembly-line-1* process performs operations related to axes, air tanks, etc. The *turn* process performs turning operations at two turning stations. The *assembly-line-2* performs operations related to bumper, radiator, engine, etc. The *exit* process completes assembly activities by exiting from conveyor, delivery to truck store, and collecting statistics data. An XML instance document, which contains *parts*, *work*, *resources*, *operation-definitions*, *schedules*, *units-of-measurement*, and *process-plans*, is manually generated to support this truck assembly line operation. The XML document is then loaded into an Arena model database using the custom-built Arena utilities mentioned previously. As a result, an Arena simulation model is generated dynamically from the model database using internal ActiveX dynamic link libraries. Figures 6-10 present screen captures of each process from the described Arena simulation model.

**Figure 6** The arrival process of a truck assembly



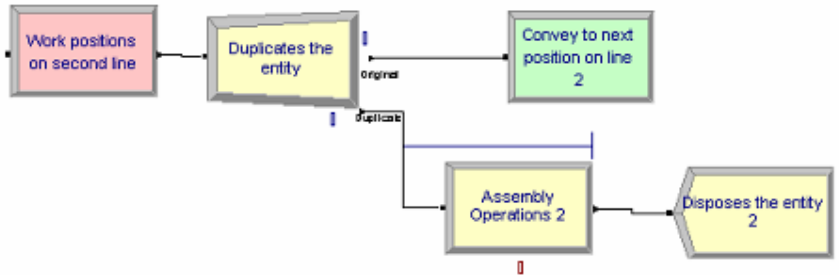
**Figure 7** The assembly-line-1 process of a truck assembly



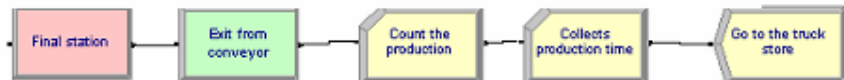
**Figure 8** The turn process of a truck assembly



**Figure 9** The assembly-line-2 process of a truck assembly



**Figure 10** The exit process of a truck assembly

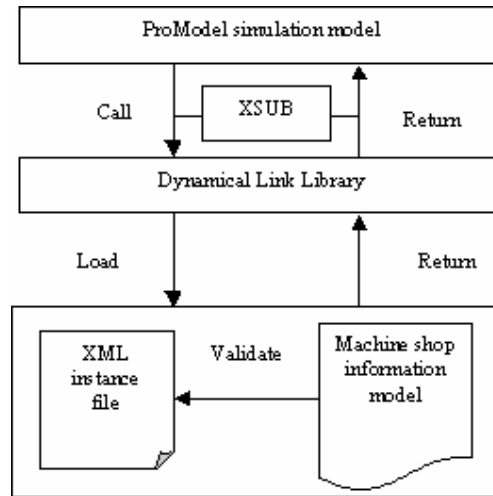




## 4.2 Application for ProModel

ProModel is a simulation and animation tool designed to model manufacturing systems of all types quickly and accurately (Harrell and Price, 2003). ProModel communicates with the external data source by XSUB. As illustrated in Figure 11, the Dynamic Link Library (DLL) executes external programs to load XML documents; query, analyze, and return values to XSUB. XSUB will transfer the value to related cells for the simulation modules to use.

**Figure 11** Machine shop information model used in ProModel



## 5 Conclusion

New technologies and strategies have been developed to reduce production time and improve product quality (Luo, 2004; Luo, 2003; Luo, 2002). Simulation technology is recognized in industry (Lee et al., 2003; McLean et al., 2002). However, simulation software supports different formats. The machine shop information model provides neutral data interfaces for a machine shop environment. A data exchange mechanism has been designed to support data exchange. Translators are developed to support data exchange between the XML document and machine shop database. The future work contains enhancing the information model and database model, studying the data sharing and integration mechanism for distributed simulation system, and developing a prototype application for the model.

## Disclaimer

*Commercial software products are identified in this paper. These products were used for demonstration purposes only. No approval or endorsement of any commercial product by the National Institute of Standards and Technology is intended or implied. The work described was funded by the United States Government and is not subject to copyright.*

## References

- Lee, Y. T. (1999) 'Information modeling: from design to implementation', Proceedings of the Second World Manufacturing Congress, eds. S. Nahavandi and M. Saadat, Canada/Switzerland, International Computer Science Conventions, New Orleans, LA, pp.315-321.
- Lee, Y. T. and Luo, Y. (2003) *A Database Design for the Machine Shop Information Model*, NISTIR 7077, National Institute of Standards and Technology, Gaithersburg, MD.

- Luo, Y. and Lee, Y. T. (2003) 'A database application for manufacturing simulation system integration', Proceedings of the IASTED International Conference: Applied Modeling and Simulation, Marina Del Rey, CA.
- Luo, Y. (2000) 'Injection molding product application activities models', International Journal of Advance Manufacturing Technology, Vol. 16, No. 4, pp.285-288.
- Lee, Y. T.; McLean, C. and Shao, G. (2003) 'A neutral information model for simulating machine shop operations', Proceedings of the Winter Simulation Conference, eds. S. Chick, P.J. Sanchez, D. Ferrin, and D. J. Morrice, Piscataway, NJ, Institute of Electrical and Electronics Engineers, pp.1296-1304.
- Luo, Y. (2004) 'Parametric tool wear estimation solution of HSC appropriate machining', International Journal of Advanced Manufacturing Technology, Vol. 23, No. 7-8, pp.546 – 552.
- Lee, Y. T. ; Luo, Y. and McLean, C. (2003) 'An interface data model supporting manufacturing simulation', Proceedings of the International Industrial Simulation Conference.
- Luo, Y. (2003) 'Chip formation analysis for high speed cutting', International Journal of Advanced Manufacturing Systems, Vol. 2 , No. 2, pp. 247-254.
- McLean, C.; Jones, A.; Lee, Y. T. and Riddick, F. (2002) 'An architecture for a generic data-driven machine shop simulator', Proceedings of the Winter Simulation Conference, San Diego, CA.
- Luo, Y. (2002) 'Manufacturing features based solution for high speed cutting', International Journal of Production Engineering and Computers, Vol. 4, No. 5, pp.25-30.
- Rockwell Software Inc. (2000) *The Arena Example Models Library*, Version 5.00, Thousand Oaks, CA.
- Harrell, C. R. and Price, R. N. (2003) 'Simulation modeling using ProModel technology', Proceedings of the 2003 Winter Simulation Conference, New Orleans, LA.
- Luo, Y. and Lee, Y. T. (2005) 'Application of machine shop data model in manufacturing simulation', Proceedings of the 2005 International Conference on Modeling, Simulation and Visualization Methods (MSV 05), Las Vegas, NV.
- Lee, Y. T. and Luo, Y. (2005) 'Data exchange for machine shop simulation', Proceedings of the 2005 Winter Simulation Conference (WSC '05), Orlando, FL.
- Lee, Y. T.; Luo, Y. and Shao, G. (2005) 'Prototype implementation based on the machine shop information model', Proceedings of Advances Production Management System: Modeling and Implementing the Integrated Enterprise, Rockville, MD.
- Deitel, H. M.; Deitel, P. J.; Listfield, J. A.; Nieto, T. R.; Yaeger, C. H., and Zlatkina, M. (2005) *C# For Experienced Programmers*, Deitel Developer Series, Prentice Hall, Upper Saddle River, NJ.
- McLean, C.; Lee, Y. T.; Shao, G.G. and Riddick, F. (2005) *Shop Data Model and Interface Specification*, NISTIR 7198, Gaithersburg, MD.

## Notes

- 1 <http://www.w3c.org>.
- 2 <http://www.oracle.com/ip/dep/otn/database/oracle9i/>.
- 3 [http://searchdatabase.techtarget.com/sDefinition/0,,sid13\\_gci213877,00.html](http://searchdatabase.techtarget.com/sDefinition/0,,sid13_gci213877,00.html) .
- 4 [http://searchvb.techtarget.com/sDefinition/0,,sid8\\_gci214133,00.html](http://searchvb.techtarget.com/sDefinition/0,,sid8_gci214133,00.html) .
- 5 <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winui/WinUI/WindowsUserInterface/DataExchange/DynamicDataExchange/AboutDynamicDataExchange.asp> .
- 6 <http://www.omg.org/uml/>.
- 7 <http://www.w3.org/XML/>.
- 8 <http://www.software.rockwell.com/index.cfm>.
- 9 <http://www.promodel.com/support/>.