

SIMULATION DATA ARCHITECTURE FOR SUSTAINABLE DEVELOPMENT

Adrien Boulonne
Björn Johansson
Anders Skoogh

Mark Aufenanger

Product and Production Development
Chalmers University of Technology
Gothenburg, 412 96, SWEDEN

Heinz Nixdorf Institut
Universität Paderborn
Paderborn, 33102, GERMANY

ABSTRACT

Reducing costs, improving quality, shortening the time-to-market, and at the same time act and think sustainable are major challenges for manufacturing industries. To strive towards these objectives, discrete event simulation (DES) has proven to be an effective tool for production system decision support.

Large companies continuously log raw data, and are therefore able to collect large quantities of resource event information. However, usually it is difficult to reuse data for future DES projects. Thus, the aim of this paper is to describe how to facilitate data sharing between data sources and DES models.

A test implementation of a simulation data architecture has been realized. A data processing tool, a database and an interface were created, which provide reusable resource event data to pave the way for sustainable resource information in DES projects. The entirety data exchange is provided by standard XML documents following the latest Core Manufacturing Simulation Data recommendations.

1 INTRODUCTION

Nowadays, virtual production tools gain increasing popularity in industry because they facilitate work parallelization and concurrent engineering. This helps to reduce times to market for new products and the tools also enable development of sustainable and efficient production systems on a continuous basis.

One example of a virtual production tool is Discrete Event Simulation (DES), an engineering application that is used for developing, analyzing and improving material flows in production. DES is applied within system design topics, e.g. for resource requirement specification, as well as in the area of system improvement, e.g. bottleneck analysis and buffer optimization. However, despite a well documented potential, the use of DES is rare in industry (McLean and Leong 2001) and one of the main reasons is argued to be the extensive time-consumption for keeping simulation models up-to-date (Skoogh and Johansson, 2010).

A possible solution to this problem is reuse of data, either from previous simulation studies or from other engineering applications and data sources. In a DES project for example, operation times can be extracted from process preparations performed earlier in the Product Lifecycle Management (PLM) chain. However, to succeed with the reuse of data within a PLM system, companies are more or less forced to select engineering applications provided by one single vendor. This is not an ideal situation since organizations select specific tools based on their engineers' education and experience, and also based on the tool's appropriateness for specific tasks. This gap between data systems and industry needs results in interoperability problems costing industry \$1Billion/year to U.S auto suppliers and \$3.9Billion/year in electronics (Gallaher et al. 2004).

The aim of this paper is to provide a database structure, based on neutral formats, which enables storage of DES data in digital factory building blocks. The building blocks include necessary data for sustain-

nability simulations, e.g. operation times, breakdown patterns and energy consumption. Storing data in neutral building blocks enables efficient reuse of data between simulation models and facilitates communication with external data sources such as maintenance systems. This is demonstrated by automatically extracting and processing breakdown data from running production. The results are supplied to an Arena simulation model using CMSD structured XML files.

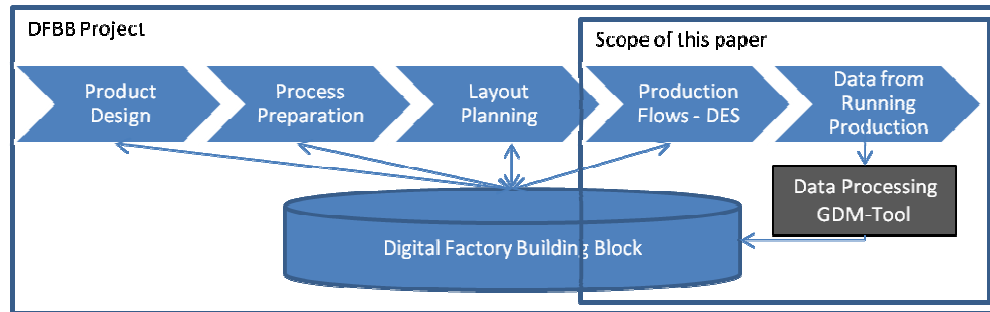


Figure 1: Framework of Digital Factory Building Blocks (DFBB) Project

This study is part of a research project called “Digital Factory Building Blocks” (DFBB 2010) funded by the Swedish government through the research program FFI (Strategic Vehicle Research and Innovation). The aim of the DFBB project is to extend the building blocks presented in this paper to also include data necessary for other engineering applications. As a result, the building blocks will improve interoperability and allow engineering applications, independent from major PLM systems, to communicate without expensive customized interfaces for data exchange.

Another major challenge in DES projects is to integrate sustainability concerns. Recently, SIMTER project (Lind et al. 2009) works on combining human’s adaptability with, speed and performance of the machine was done. Environmental impacts are also taking part of the SIMTER project. The simulation data architecture developed in this paper provides means to integrate ergonomic and environmental information, which can be use in DES projects.

2 LITERATURE REVIEW

A number of approaches could be chosen to develop data management architectures. Previous work describes many different ways to store, exchange and normalize exchange from the data origin to any manufacturing application. In the following sections we describe some previous solutions to realize this communication.

Production systems generates large amount of information, e.g. about cycle time and downtime. Furthermore, they are often also able to communicate with big databases which can store process times and downtimes. One of the main challenges for this paper is dealing with how to process and reuse this information in DES, Enterprise Resource Planning (ERP), Master Production Schedule (MPS), or Manufacturing Execution Systems (MES). In this paper, we identify four main areas to address for efficient bridging industrial data architectures from data origin to the target: (1) How to store data? (2) How to exchange data? (3) What is the target? and (4) How to write data?

2.1 Data Architecture: Input Data Methodologies

This section presents a frame of reference for simulation data architecture and standardized data exchange. Moreover, it identifies the present research and main gaps that this paper addresses.

2.1.1 Data Storage

Robertson and Perera (2002) define four different data management architectures for DES modeling:

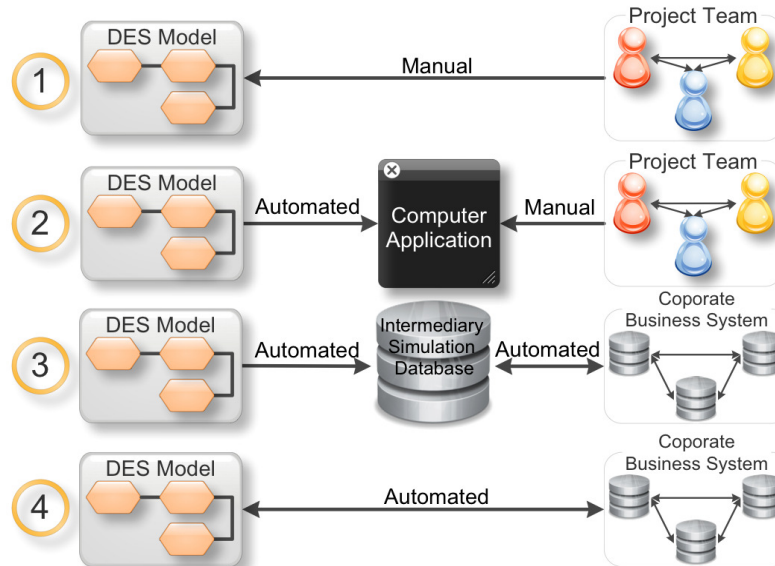


Figure 2: Possible data collection methods for model building (Robertson and Perera 2002)

Figure 2 shows two different intermediaries from the origin (Project Team or Corporate Business System) to the target (DES model): computer application (e.g. spreadsheet) and intermediary simulation database (e.g. MS Access or MySQL). This paper is focused on the third method: Corporate Business System (CBS) for data origin storage, and a database for the intermediary storage.

2.1.2 Share Information: Data Format

Since 1996, XML has proven to be an effective method of transporting data from one location to another (W3C 2008). But why use XML? Data structure is the main advantage of XML. Its syntax allows storing each piece of data in user defined markup tags. The markup tags are arranged in a hierarchy that represents the ordering of the data in a parent/child relationship. XML is a neutral format that simplifies data sharing, data transport, platform changes, and makes data more available.

2.2 Information Management and Standardization

Data exchange standardization is adopting in many areas like process engineering, process control engineering, oil and gas industry, and in manufacturing automation engineering. Some norms are described below.

Automation Markup Language (AutomationML) is a XML based neutral data format for storing and exchanging plant engineering information. The aim is to interconnect the heterogeneous tool landscape of engineering tools in mechanical plant engineering, electrical design, HMI development, PLC, and robot control fields. AutomationML works as the glue between all the factory planning tools. It implements an object oriented paradigm; real factory components are described as data objects. They encapsulate different aspects of a real object. Typical factory components are; geometry, kinematic behavior, classification within the factory topology, and the relations between the objects. There are already many industrial standards for many of those aspects. Within AutomationML these standards are used and, if necessary, a standard enhancement is tried to be developed. Basically AutomationML supports the following concepts: Factory topology – CAEX, geometry – COLLADA, kinematic – COLLADA, behavior description – PLCOpen-XML, and references from CAEX to external data (Drath 2010).

ISA-95, International Standard for the Integration is a standardization of enterprise and control systems, known internationally as IEC/ISO 62264, provides UML models and technology that can be used to determine which information has to be exchanged between system for sales, finance, logistics and sys-

tems for production, quality and maintenance. As a general rule, ISA-95 describes all exchange from ERP to MES systems using World Batch Forum (WBF) XML schemas standards. Business To Manufacturing Markup Language (B2MML) is an XML implementation of ISA-95 family standards. B2MML consists of a set of XML schemas written using the World Wide Web Consortium's XML Schema language (The International Standard for Integration 2010).

National Institute of Standards and Technology (NIST) researchers have been working on a standards development effort, Core Manufacturing Simulation Data (CMSD) under the guidelines, policies and procedure of the Simulation Interoperability Standards Organization (SISO). The work behind this paper relies on the CMSD Information Model. Due to a lack of interoperability between manufacturing applications and simulation systems, the CMSD effort was launched. The CMSD Information Model defines a data specification for efficient exchange of manufacturing data in a simulation environment (Leong, Lee and Riddick 2006).

CMSD Goals are: (1) to foster the development and use of simulations in manufacturing applications, (2) to facilitate data exchange between simulation model and other manufacturing software applications and (3) to increase manufacturing software application interoperability.

CMSD Specification is based on two different mainstays: (1) class and packages diagrams using UML and (2) XML schemas. The UML diagram and the XML schemas are intended to be identical representations of the same CMSD conceptual model (Riddick and Lee 2008). The UML representation provides a means to describe interrelationships between the essential entities in the manufacturing field. The XML file serves as a support for store and exchange information about those relationships. The CMSDIM use six main UML packages: (1) CMSD package, (2) Support package, (3) Resource Information package, (4) Production Planning package, (5) Production Operations package, (6) Part Information package. CMSDIM makes sixteen manufacturing information categories available, to only quote some of the categories: Organization, Resource, Skill, Part, Inventory, Process, Schedule, etc. All of them are described in Riddick and Lee (2008) as well as in Leong, Lee and Riddick (2006).

2.3 Test Implementation of CMSD Specification

Since the first CMSD Information Model publication some test implementations have been realized. One of the major studies was in a paint shop at Volvo Cars (Johansson et al. 2007). CMSD XML document was used to link a Microsoft Excel user interface and an Enterprise Dynamics simulation model. The results have showed importance of a detailed modeling approach to specify the information model and data structure. Moreover, findings did also point out that it is better to consider a CMSD approach for defining manufacturing information directly in the early phases of simulation model development. Through CMSD XML document, cycle time, downtime, and many other resource properties can be exchanged following a standardized data structure. In the same way, a pilot implementation of the CMSD, based on a real-world production operation (Lee et al. 2007), has showed that the CMSD Information Model could be a viable interface standard between simulation systems and manufacturing applications. More recently, a project has demonstrated interoperability of manufacturing applications in a simulated environment for generic job shop via CMSDIM (Mazhari and Son 2009). This project gave rise to several XML schema, XML documents, and converters/translators for different manufacturing applications using the Visual Basic for Applications (VBA) language.

2.4 Input Data Management for DES

The data collection process is the most crucial stage in the model building process according to Robertson and Perera (2002). This step is also one of the most time-consuming in simulation model realization. In project using DES to analyze and improve material flows in production, input data management consumes on average 31% of the total project time (Skoogh 2009). The Input data management process starts with the data collection from the real world and ends with entering the information into the DES model.

A time-consumption analysis on 15 DES projects identified two main parameter classes which required more than 70% of the total time in the input data phase. Process time, with 42%, and downtime, with 32%, are the most time-consuming of the entire input data phase (Skoogh and Johansson 2007). This concludes that cycle time and downtime data are the main targets for shortening the time consumption.

An automated input data management software solution, called the GDM-Tool (Generic Data Management-Tool), was developed to reduce time-consumption and improve input data management. The GDM-Tool output provides standard XML files (CMSD) which can be used as input for DES or for any other manufacturing application. A first evaluation of the GDM-Tool showed that the time consumed on input data management can be reduced by 78% with maintained data quality (Skoogh and Johansson 2007).

These results pave the way of new projects for developing software solutions, which will help to reduce time-consumption.

2.5 Identification of Gap and Research Contribution

Previous research projects in the area of competence based input data management provide data format standardization (CSMD Information Model), data management methodologies, weaknesses identification for DES project development, and automated software solution for reducing time-consumption in DES development (GDM-Tool). The CMSD Information Model is the new reference for structuring simulation data in DES projects. These previous results take part of the data management toolbox with the same goal: improving interoperability and data management between manufacturing application and DES models.

The aims of this paper is to provide a simulation data architecture integrating, CMSD document, the GDM-Tool, and a generic relational database which is able to store and reuse input simulation information. A user interface (UI) system provides means to generate any factory configuration for any activity. The relational database (Microsoft Access) with its generic UI can generate standard XML files (CMSD Specification) from the GDM-Tool to DES models or any other manufacturing application. This paper groups and reuses research results in order to develop an implementation of a generic data architecture for DES projects.

3 INPUT DATA MANAGEMENT: ARCHITECTURE AND IMPLEMENTATION

This section presents the research contribution of this paper and the main results on simulation data architecture and input data management.

3.1 Outline of Architecture

The presented work provides a solution where each individual element plays a part in a dynamic architecture. The CMSD XML document is designed to be the architectural binder, enabling interoperability and proper understanding between each element. Thus, building and parsing XML files and keeping to strict rules is the cornerstone of successful data exchange. Moreover, using a standardized interface for communication provides the possibility to separately develop each element without affecting the complete architecture. Figure 3 shows the data management architecture in greater detail. Four main parts can be identified:

- **The RIM Database:** (Resource Information Management) composed of a MS Access relational database for data storage and a user interface for data selection and generation of XML files. The RIM database can be populated using the GDM-Tool (see below) XML files. However, it is also possible to manually populate the database through dialogue boxes. See Figure 3 and section 3.2.
- **The GDM-Tool:** (Generic Data Management) for extraction of data, conversion of data to information, and data presentation. For further information see section 3.3.

- **CMSD Document (XML):** Neutral interface that links the RIM database, the GDM-Tool and a DES model by providing an efficient and standardized structure for data exchange. See section 3.4.
- **Plug-in DES model:** the last element in the chain is an Arena DES model, which understands XML files through VBA blocks and parsing algorithms. See also section 3.5.

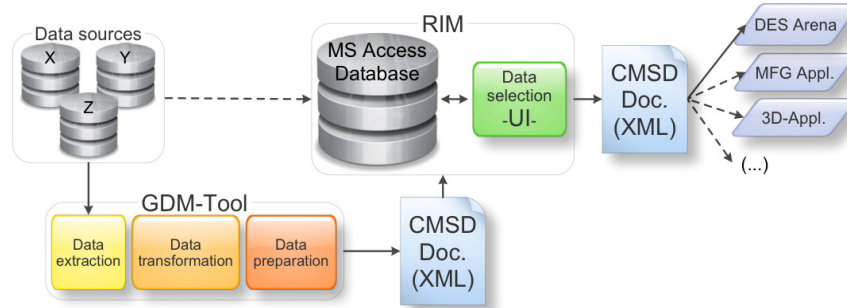


Figure 3: Outline of the architecture

3.2 Data Storage: The RIM Database

The RIM solution consists of two parts: a relational database and a user interface. Both parts are described in the section below.

3.2.1 Design of the Relational Database

In this architecture, a database provides means to create generic blocks for resource description. A Microsoft Access relational database is applied to three main objectives: (1) store information about resources and their properties, jobs, and products, (2) reuse information for new DES projects, and (3) structure information in a way that enables generation of standardized XML files describing factory resources. Figure 4 shows the entity-relationship diagram with all defined tables, which is the final step before database implementation. This database is designed to facilitate resources version management, which provides a reusable predefined combination of properties.

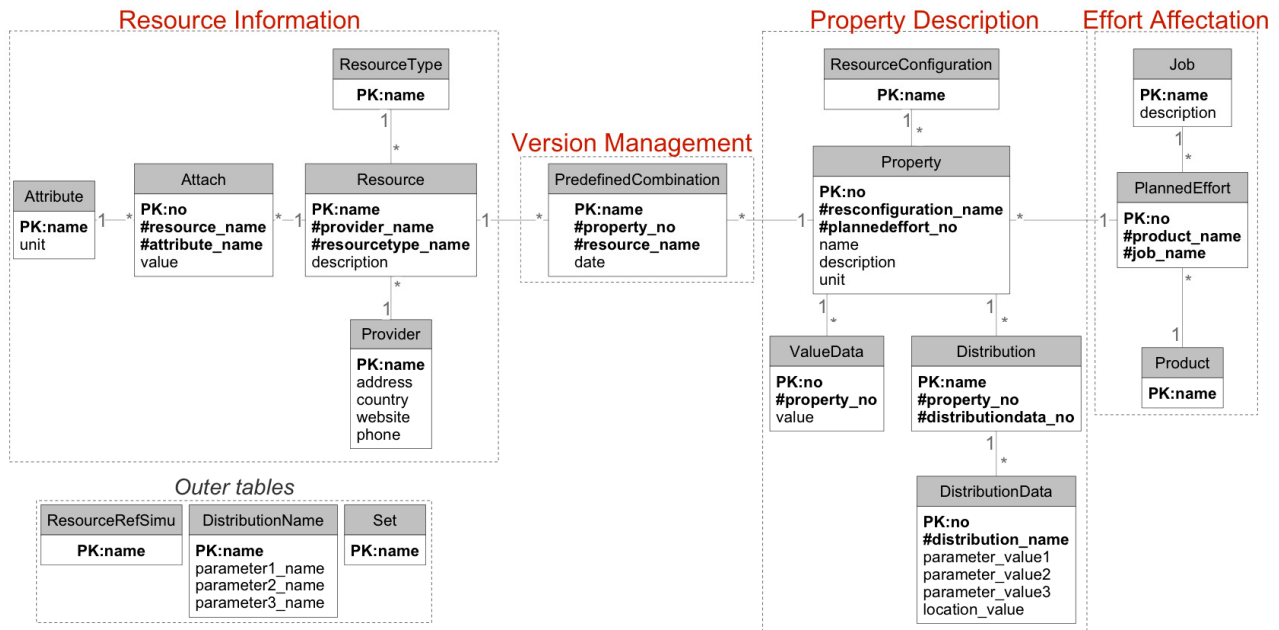


Figure 4: Enhanced Entity-Relationship diagram

3.2.2 RIM User Interface

As a link to the relational database, the user interface provides means for generating standard XML files according to the CMSD Specification. The RIM database accumulates experience of previous work by storing information (cycle time, downtime and other technical attributes). All this information can be used and reused for new factory configurations. Furthermore, the RIM solution can generate any XML file describing any production line via an interface that consists of four main parts (Figure 5):

PART A: Resource Selection provides means to associate a resource reference to a resource name. The resource reference is a unique identifier which identifies a resource in the DES model or in any other manufacturing application.

PART B: Resource Information. In this part, the user has access to all information concerning a resource, regardless of the *Resource Configuration* choices. Information like picture, provider information, technical attributes, energy consumption information, and a CAD drawing gives an overview of the selected resource.

PART C: Resource Configuration gives full access to design a resource. Each resource reference is associated to a product-job couple which it links to a set of properties (cycle times, MTBF, and MTTR). Therefore, a resource configuration is a combination of information. From database information, it is possible to select, create, and modify information in order to define a resource reference. In fact, there are three possibilities to design a resource: (1) manual data entry, (2) reuse existing data from the database, and (3) import historical production data processed by the GDM-Tool.

PART D: Factory Composition & XML Generation. A new line is added to the factory list each time a resource configuration, associated to a product-job couple, is saved. The user can add resources without specific logic because factory configuration is just a list used for generating XML files (logic will be implemented in the DES software or in other manufacturing applications). Both parsing algorithms and operations for generation of XML files have been written in the Visual Basic for Application (VBA) language. Chilkat ActiveX Component is used to facilitate working with XML files.

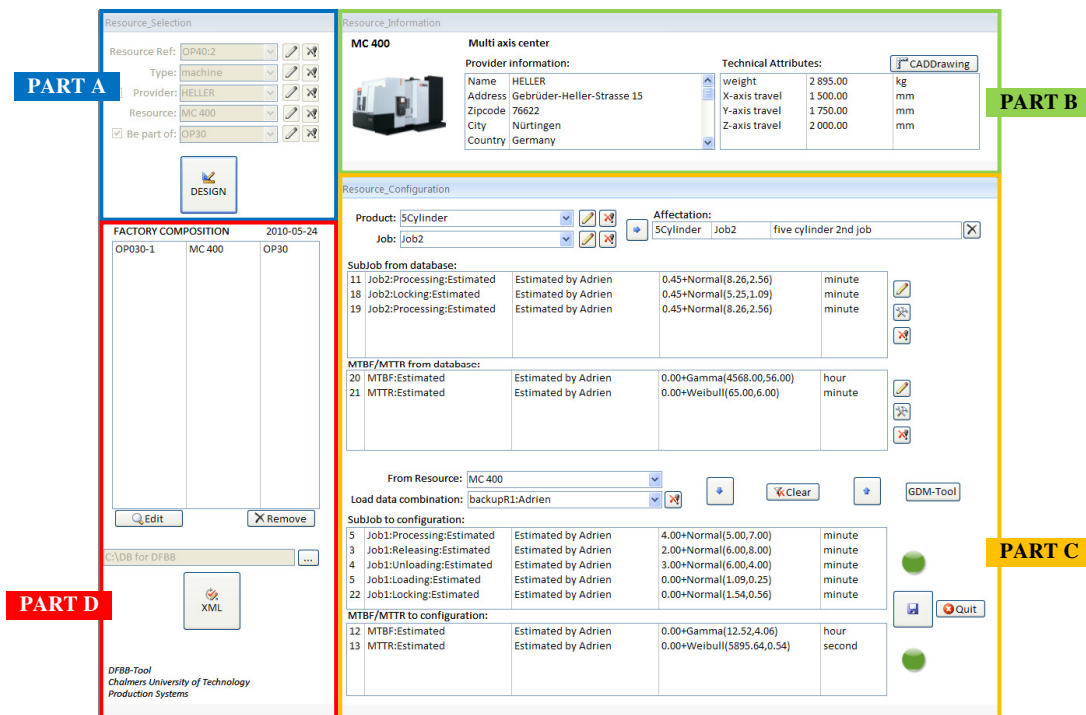


Figure 5: RIM User Interface

3.3 Input Data Management: The GDM-Tool

The GDM-Tool is a computer application that demonstrates the concept of automated input data management. It is a MS Windows desktop application created in C# (Visual Studio), originally developed for a case study in automotive industry (Skoogh, 2009). The tool automates all steps in the input data management process, except for the collection of raw data; see Figure 6. This includes extracting data from original sources, cleaning the data from irrelevant or erroneous samples, performing required calculations, and condensing data to statistical distributions. Finally, the results are exported to a neutral format for further use in engineering applications such as DES software. One key feature is that the application is able to extract data from several sources, such as automated data logging systems, ERP systems and customized spreadsheets. This makes it independent from major corporate business systems, which usually do not contain all required data for dynamic simulations. In this paper, the GDM-Tool is used to extract and process operational data from the shop-floor in order to populate the building blocks of the RIM database.

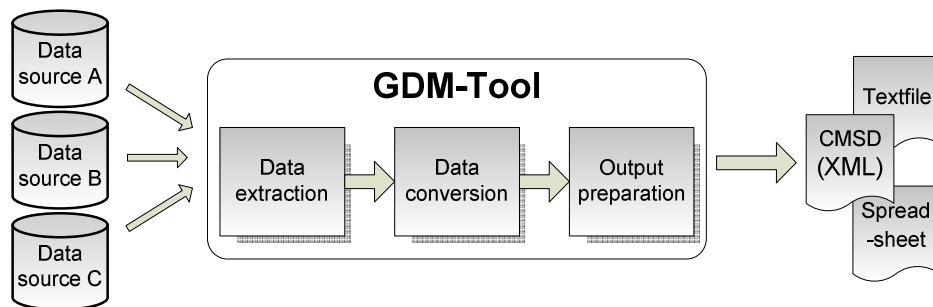


Figure 6: Outline of the GDM-Tool functionality (Skoogh 2009)

3.4 Data Exchange: CMSD Document

According to the CMSD Information Model, Figure 7 presents the XML tree view with four main parts: (1) job description, (2) sub job description, (3) resource group description, and (4) resource description. This CMSD XML document is generated using SQL requests through the MS Access relational database.

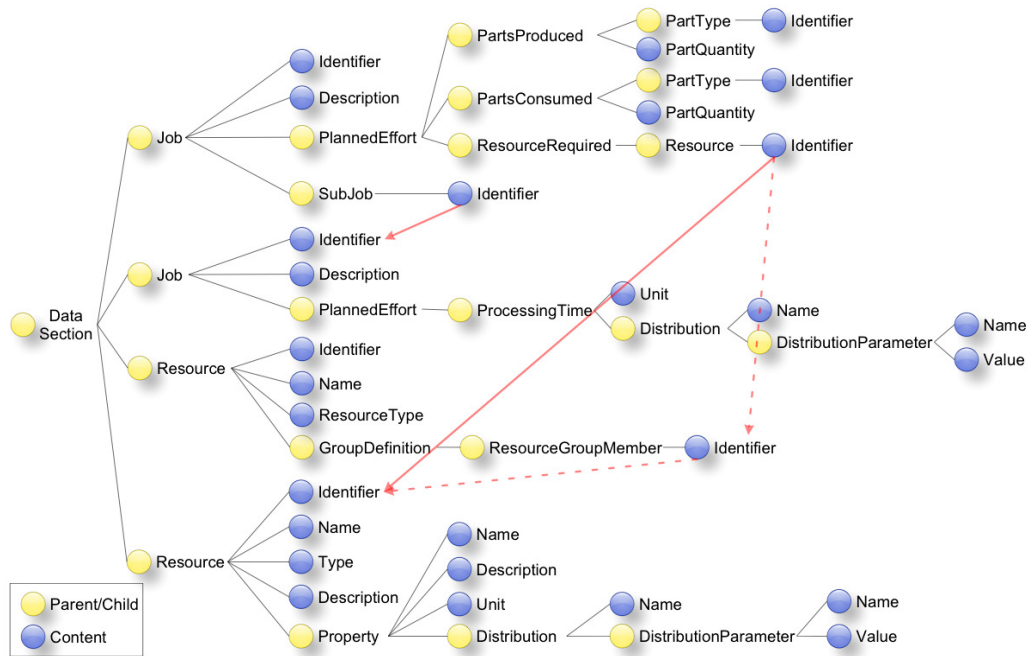


Figure 7: CMSD Document – Tree View

3.5 Engineering Application: ARENA Simulation Model

The purpose of generating the CMSD XML document is to populate a DES model with information. In this case, the software ARENA provided by Rockwell Automation was chosen to test the data exchange between CMSD and simulation models. A production line model at a company in the automotive industry was built to enable this test procedure. The Arena DES model includes cycle times divided in five steps (loading, locking, processing, realizing, and unloading) as well as downtime patterns for machines, transporters, and conveyors. The previously generated XML file stores input information necessary to run the ARENA model. The main challenge is to parse this XML file using SIMAN and VBA commands through ARENA VBA blocks in order to find and allocate the correct values from the XML to the corresponding variables in the ARENA model.

4 SUSTAINABLE DEVELOPMNET: ARCHITECTURE IMPACTS

The developed architecture was enhanced by the integration of sustainability data. Therefore, the developed environment supports storing of sustainability indicators – environmental and ergonomic – in the DES resource description. Designing reusable resource production blocks by the use of a resource information repository is a very efficient way to adapt existing and develop new manufacturing lines in a sustainable way.

4.1 XML Document: Integrating Sustainability Indicators

In addition to the possibility of storing and generating XML documents, including cycle time and downtime information, the RIM solution was designed to integrate sustainability indicators: energy consumption and emission data for environmental sustainability, and ergonomic data for social sustainability. Figure 8 illustrates an example of a XML document, including energy consumption data for different resource’s status: busy, idle, and down. In this way, the RIM solution takes completely part on “green performance” improvement and Life Cycle Assessment methodology (Heilala et al. 2008). A more efficient way of modeling becomes possible.

```
<Property>
  <Name>EnergyConsumption:Busy</Name>
  <Description>Measured on real production system</Description>
  <Unit>watt</Unit>
  <Value>50000</Value>
</Property>
<Property>
  <Name>EnergyConsumption:Idle</Name>
  <Description>Measured on real production system</Description>
  <Unit>watt</Unit>
  <Value>1000</Value>
</Property>
<Property>
  <Name>EnergyConsumption:Down</Name>
  <Description>Measured on real production system</Description>
  <Unit>watt</Unit>
  <Value>10</Value>
</Property>
```

Figure 8: A XML sample document including energy consumption

4.2 Discrete Event Simulation: Sustainable Development

Interoperability between different manufacturing applications is considered as a key feature to allow efficient and sustainable product and production development (DFBB 2010). Therefore, the main goal of the DFBB project is to create and define a library of resource production models that can be reused within many different analysis and applications. The RIM database provides the possibility to store and reuse

predefined resource configurations for new DES projects. The use of the building block concept is a cornerstone within the RIM solution design. Therewith, it becomes possible to provide and maintain a standard data set. The availability of standard resource descriptions – as the highest priority – leads to reusable and sustainable DES model development. When doing so, many resources like manpower, energy, and time can be saved during the model creation process.

5 DISCUSSIONS

This paper describes an example of the possibilities to build a data architecture in order to manage DES input data information. The authors developed a software solution, called the RIM solution, which complements the GDM-Tool functionalities. The GDM-Tool automates two of the most time-consuming input data management stages (Skoogh and Johansson 2007): extraction of raw data, and transformation raw data into information. The RIM database provides a resource information repository through a MS Access relational database and information selection functionalities with a user interface. Therefore, the main results of the RIM solution development are: reducing time-consumption by enabling reuse of resource information, generation of standard (CMSD) XML files with resource information to design factory configurations, and pave the way for introducing sustainability indicators such as environmental (e.g. CO₂ and Energy consumption) and ergonomics in DES projects. The default exchange format in this study follows the CMSD Information Model, which facilitates the communication between GDM-Tool/RIM/ARENA simulation models.

However, it is important to highlight that there are previous contributions explaining detailed methods to input data management facilities. For instance, Lee and Luo (2005) describes a data exchange for machine shop simulation using XML files and machine shop database. Moreover, Johansson et al. (2007) presents a test implementation of the CMSD specification, which includes an Excel interface, a CMSD XML document (for supporting information), and an Enterprise Dynamics simulation model. Thus, the work made on this paper is placed in the continuity of the previous publications. The Development of a software solution using database and user interface to store and facilitate resource data reuse is the most novel part of this paper. In addition, building standardized XML files following CMSD specification to describe jobs and required resources in a real case study is in line with NIST standardization development work to provide examples of a CMSD implementations.

6 CONCLUSIONS

The purpose of this study is to present a simulation data architecture, which enables to manage input data for DES models. This paper proposes a full chain between raw data and DES model or any other manufacturing applications, using a CMSD XML document including resources cycle times, downtimes, sustainability data (energy consumption, CO₂ ergonomics, etc.), and other resource properties. The data architecture described gives good perspectives to include sustainable preoccupations in DES models. Likewise, the RIM database makes available reusable resource descriptions improving DES model development.

The development of the CMSD effort is frequently updated in order to extend the area of simulation data standardization. This paper takes part of CMSD research in order to diffuse simulation data standardization into general use in industrial fields of DES model building.

For future work, we aim to validate the development in additional industrial cases. In order to evaluate the time-consumption using the architecture described in this paper and compare the results to traditional methodologies for data management in DES modeling. Moreover, future projects could extend resource data information to add e.g. scheduling, planning description, and tool management. Also, it would be interesting to focus the research on other data format standardizations (AutomationML, ISA-95, etc.). Finally, this paper shows a “home-made” relational database (the RIM database). In order to extend simulation data standardization, it will be worthwhile to develop a more standardized database, which enables standardized mapping vs. CMSD XML documents.

ACKNOWLEDGMENTS

The funding for this research is granted by VINNOVA (Swedish Government Agency for Innovation Systems) and ProViking. The authors would also like to thank Frank Riddick (NIST, USA) for his extensive and valuable contributions on design and development of standard CMSD Document.

REFERENCES

- Drath, R. (ed.). 2010. *Datenaustausch in der Anlagenplanung mit AutomationML – Integration von CAEX, PLCopen XML und COLLADA*. Springer. Berlin.
- Gallagher, M. P., A. C. O'Connor, J. John L. Dettbarn, and L. T. Gilday. 2004. Cost analysis of inadequate interoperability in the U.S. capital facilities industry. National Institute of Standards and Technology (NIST).
- McLean, C. and S. Leong. 2001. The expanding role of simulation in future manufacturing. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, M. W. Rohrer, 1478-1486. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Heilala, J., S. Vatanen, H. Tonteri, J. Montonen, S. Lind, B. Johansson, and J. Stahre. 2008. Simulation-Based Sustainable manufacturing System Design. In *Proceedings of the 2008 Winter Simulation Conference*, eds. S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, J. W. Fowler, 1922-1930. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Johansson, M., S. Leong, Y. T. Lee, F. Riddick, G. Shao, B. Johansson, A. Skoogh, and P. Klingstam. 2007. A Test Implementation of the Core Manufacturing Simulation Data Specification. In *Proceedings of the 2007 Winter Simulation Conference*, eds. S. G. Henderson, B. Biller, M.-H Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, 1673-1681. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Lee, Y. T., S. Leong, F. Riddick, M. Johansson, and B. Johansson. 2007. A Pilot Implementation of the Core Manufacturing Simulation Data Information Model. In: *Proceedings of the Simulation Interoperability Standard Organization 2007 Fall Simulation Interoperability Workshop*.
- Lee, Y. T., and L. Yan. 2005. Data Exchange for Machine Shop Simulation. In *Proceedings of the 2005 Winter Simulation Conference*, eds. M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Leong, S., Y. T. Lee, and F. Riddick. 2006. A Core Manufacturing Simulation Data Information Model for Manufacturing Applications. In: *Proceedings of the Systems Interoperability Standards Organization Fall Simulation Interoperability Workshop*.
- Lind, S., Johansson, B., Stahre, S., Berlin, C., Fath, A., Heilala, J., Helin, K., Kiviranta, S., Krassi, B., Montonen, J., Tonteri, H., Vatanen, S. and Viitaniemi, J, 2009. SIMTER – A Joint Simulation Tool for Production Development. VTT Working Paper 125. 49 p. Espoo.
- Luo, Y., and Y. T. Lee. 2005. Application of Machine Shop Data Model in Manufacturing Simulation. In: *MSV*.
- Mazhari, E. and Y.-J. Son. 2009. Inter-Operability of Manufacturing Applications in a Simulated Environment via CMSD Information Model. Systems and Industrial Engineering, The University of Arizona. Available via <http://www.sie.arizona.edu/faculty/son/NIST3.html> [accessed May 21, 2010].
- Microsoft Office Access 2007. Available via <http://www.office.microsoft.com/> [accessed May 21, 2010].
- National Institute of Standards and Technology: NIST. Available via <http://www.nist.gov/> [accessed May 21, 2010].
- Riddick, F., and Y. T. Lee. 2008. Representing Layout Information in the CMSD Specification. In *Proceedings of the 2008 Winter Simulation Conference*, eds. S. J. Mason, R. R. Hill, L. Mönch, O. Rose,

- T. Jefferson, J. W. Fowler, 463- 470. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Robertson, N., and T. Perera. 2002. Automated data collection for Simulation?. In: *Simulation Practice and Theory*, Volume 9 (2002), pp. 349-364.
- SISO – Simulation Interoperability Standards Organization, CMSD Product Development Group. 2010. Standard for: Core Manufacturing Simulation Data – UML Model, April. 30, 2010. Available via <http://www.sisostds.org/index.php?tg=fileman&idx=list&id=49&gr=Y&path=Specifications> [accessed May 21, 2010].
- Skoogh, A. 2009. *Methods for Input Data Management – Reducing the Time-Consumption in Discrete Event Simulation*. Department of Product and Production Development. Division of Production Systems. Chalmers University of Technology, Göteborg, Sweden.
- Skoogh, A., and B. Johansson. 2010. Mapping of time-consumption during input data management activities. *Simulation News Europe*. 19:2.
- Skoogh, A., and B. Johansson. 2007. Time-consumption analysis of input data activities in discrete event simulation projects. In: *Proceedings of the 2007 Swedish Production Symposium*. Gothenburg, Sweden.
- Strategic Vehicle Research and Innovation - Digital Factory Building Blocks. Chalmers University of Technology. Department of Product and Production Development. Available via <http://www.chalmers.se/ppd/SV/forskning/forskning-inom8036/pagaende-projekt/ffi-8211-digital-factory> [accessed May 21, 2010].
- The International Standard for Integration: ISA-95. Available via <http://www.isa-95.com/> [accessed May 21, 2010].
- World Wide Web Consortium: W3C 2008. Extensible Markup Language: XML 1.0 (Fifth Edition). Available via <http://www.w3c.org/> [accessed May 21, 2010].

AUTHOR BIOGRAPHIES

ADRIEN BOULONNE is an exchange master student in at the Department of Product and Production Development, Chalmers University of Technology, Sweden. He is mainly interested in discrete event simulation, production systems, inventory management, and supply chain. His email address is Adrien.Boulonne@ifma.fr.

BJÖRN JOHANSSON is an Assistant Professor at the Department of Product and Production Development, Chalmers University of Technology, Sweden. He serves as Production Modeling Corporation director for the Swedish office in Gothenburg. He is also a guest researcher at National Institute of Standards and Technology in Gaithersburg, Maryland, USA. He was a WSC proceedings co-editor in 2009. His research interest is in the area of discrete event simulation applied for manufacturing industries, including environmental effects modeling, modular modeling methodologies, software development, user interfaces, and input data architectures. His email address is Bjorn.Johansson@chalmers.se.

ANDERS SKOOGH is PhD student in the field of Discrete Event Simulation at the Department of Product and Production Development, Chalmers University of Technology, Sweden. In 2005 he obtained his M.Sc. degree in Automation and Mechatronics from the same university. Anders has industrial experience of Discrete Event Simulation from his former employment as logistics developer at Volvo Car Corporation. His email address is Anders.Skoogh@chalmers.se.

MARK AUFENANGER studied Business Computing, at the University of Paderborn. Since 2005, he is a research assistant at the group of Prof. Dangelmaier, Business Computing, esp. CIM. He is mainly interested in simulation of logistic systems, artificial intelligence and knowledge based systems. His email address is Mark.Aufenanger@hni.uni-paderborn.de.