18

INTER-ENTERPRISE PLANNING OF MANUFACTURING SYSTEMS APPLYING SIMULATION WITH IPR PROTECTION

Kai Mertins and Markus Rabe
Fraunhofer IPK Berlin, Germany
e-mail: markus.rabe@ipk.fraunhofer.de

Abstract: Discrete Event Simulation is a well-proved method to analyse the dynamic behaviour of manufacturing systems. However, simulation application is still poor for external supply chains or virtual enterprises, encompassing several legal entities. Most conventional simulation systems provide no means to protect intellectual property rights (IPR), nor methods to support cross-enterprise teamwork. This paper describes a solution to keep enterprise models private, but still provide their functionality for cross-enterprise evaluation purposes. Applying the new modelling system, the inter-enterprise business process is specified by the user, including a specification of the objects exchanged between the local models. The required environment for a distributed simulation is generated automatically. The mechanisms have been tested with a large supply chain model.

Key words: Distributed Simulation, HLA, Supply Chain, Manufacturing Engineering, IPR Protection

1. INTRODUCTION

Simulation is a well-proved method to analyse the dynamic behaviour of manufacturing systems. Today, simulation systems for manufacturing are basically closed systems, which always run the complete scenario within one simulation execution machine. This brings several disadvantages, especially for the efficiency of the team work, the reusability of simulation models and the protection of the intellectual property rights (IPR). IPR are a specific issue if the system under consideration is an external supply chain or a virtual enterprise, and includes several legal entities.

A consortium of companies and research institutions from the European Union, Japan and USA have faced this problem, and joined for the common IMS project " Modelling and Simulation Environments for Design, Planning and Operation of Globally Distributed Enterprises " (MISSION) [1]. Methods have been developed to efficiently structure the simulation model into several separate federates, which may run on different computers within different simulation execution machines, even from different software vendors.

The European Module of the MISSION project has put specific effort to introduce methods to keep the interfaces flexible, and at the same time to support common enterprise models shared by several enterprises. Specification methods and tools provide the necessary environment to model the federation on the interface level and to generate the required interfaces.

2. SIMULATION AND IPR

Simulation models contain detailed strategies of the company. Engineers might not wish to make these known to others, especially not outside of the enterprise. Therefore, mechanisms for knowledge protection are required.

Knowledge is distributed, as local engineers know the local rules and environment best. Therefore, best modelling results are expected with decentralised models, which are built and maintained locally. For these decentralised (partial) models, there is no IPR problem, as none of these models has to leave the originating enterprise. However, in order to achieve results about the relationships within the supply chain, these partial models need to be combined.

Within classical simulation systems, all parts of a large simulation model have to be physically joined, before the simulation can be run [2]. Therefore, the detailed description of the enterprises' business processes has to be handed out to an external company, in the shape of the simulation model. There might be a chance to define a neutral third party, e.g. a simulation service provider, for this purpose. Specific IPR protection regulations between this service provider and the co-operating enterprises might help to secure the business process information.

However, enterprises might wish to make sure, and not provide their internal processes to any other firm. Separate distributed models are adequate to fulfil this requirement. They are built and maintained locally, and joined for evaluation purposes, only. The interface description, which is necessary to evaluate the complete supply chain, can be used for purposes of process engineering, evaluation by simulation and specification of the supply chain control system.

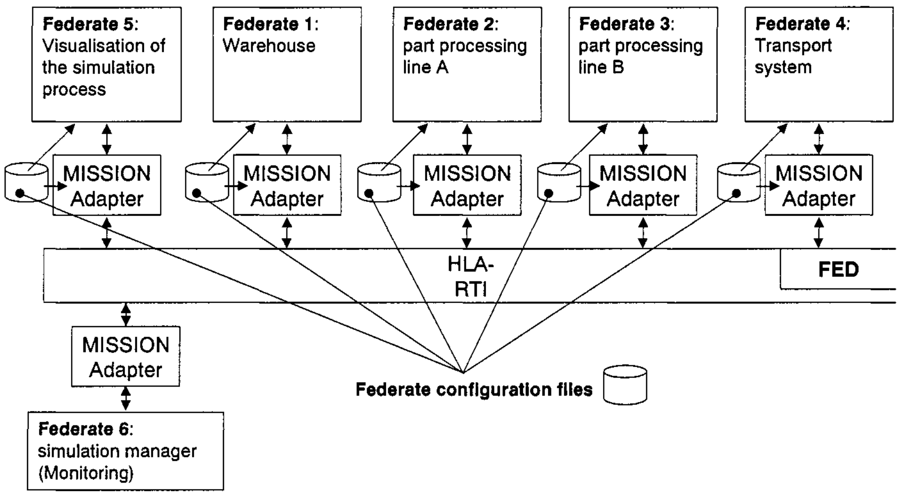

Figure 1. Execution of a Simulation Scenario

3. MODELLING TECHNOLOGY

There is no super-model describing the supply chain as one large simulation model. In contrary, separate models of the chain elements are communicating in a way, which is rather similar to their communication in reality. As a consequence, the interface descriptions generated by the MISSION process description can be directly used as specification of the supply chain interfaces.

3.1 Connection Platform

The High Level Architecture (HLA) [3] was selected as the base architecture. The HLA satisfied many requirements for distributed simulations [4], e.g. time synchronisation, communication between independent simulation models etc. [5]. Woerner [6] states, that HLA requires exchange of detailed model information. However, experiments have demonstrated that the exchange of interface information is sufficient, and all model details may be hidden when applying HLA.

In fig. 1 the runtime environment of the model is shown. The separate simulation models are called "federates" and may be enhanced by other federates for purposes like monitoring or visualisation. The run-time infrastructure (RTI), which provides the basic mechanisms, requires a specific set of Federation Execution Data (FED). The interface between RTI and federates is provided by adapters. These adapters encompass a generic part (bridge to RTI) and a second part bridging to the simulation systems, which has to be implemented once for each additional simulation system.

3.2 Federation Specification

Before an evaluation can be started, the manufacturing system under consideration has to be planned and modelled. This is done within the Manufacturing System Engineering (MSE) process, and the result is the process model of the Manufacturing System (MS) [7].

A central supporting structure is the template library [8]. It enables a flexible definition of classes, attributes and objects. The library content is not fixed. The user can add classes and attributes at any time.

The specification task is done in two steps. First, the process itself is modelled. This process describes the steps for production and logistics, but it does not define the systems implementing these steps. Then, the template library is searched for suitable production and logistics systems, and those are added to the model, connecting from the bottom to the respective process steps. Parameters are set for the single templates. For example, if a specific warehouse is selected as part of the chain, it should be specified how much space for pallets should be reserved there.

Exchange Objects, which are described for each template, specify the classes used for the communication with the distributed simulation environment.

3.3 Adapters

A new software package includes all the necessary interfacing components as well as special sets of building blocks for specific commercial simulation systems (Taylor and ARENA available, will be extended). The major task of these new elements is to replace the sources and sinks of the original models by connectors to the outside world. Using these building blocks, engineers can model the interface of the simulation model just in the same way they are used to model the manufacturing system itself.

A major problem of the HLA detected when developing the adapters is the ownership mechanism. HLA requires that exactly one federate "owns" an attribute at any point of time, which avoids conflicts and ensures that there is always exactly one valid instantiation. However, when applying to manufacturing system simulation, a federate releases ownership of an exchange object, when it has fulfilled all necessary tasks with this object. Sometimes there is a time gap before an other simulation system is ready to take this ownership. In these cases the exchange object can be completely lost [9]. For the adapters described, workarounds have been implemented, which collect dangling ownership and provide it later to a federate requiring it. A more sophisticated approach is under development.

3.4 Configuration Files

The HLA-RTI expects a Federate Execution Data (FED) file. It contains the class names of common objects as well as the names of their attributes. This file can be easily generated from the federation specification. The FED information is sufficient to run the RTI, but not adequate to control the interfaces to the RTI. HLA proposes to implement specific interfaces based on the specification. This means, that most changes in the classes will induce changes in the adapter implementations.

The approach described in the previous section, however, requires that the interface between the commercial simulation system and the HLA-RTI is somehow "self-configuring". This is done by separating all model-specific information from the interface code, and storing it in a separate XML file, the *Federate Configuration File* (Table 1). All information which is contained in this file has already been specified by the engineer within the template library or the process model before, as described above. Therefore, it was possible to automate the generation of these files.

The Federate Configuration Files (FCF) are generated using the process model (including the parameters of the single templates), the Template Library and information about the input and output segments, specified with building blocks (fig. 2). Further details about the FCF, the interface and the underlying mechanisms are reported by Rabe and Jäkel; the technical implementation is given by Rabe and Gurtubai [7, 10].

Table 1. Federation Execution Definition File (FED) vs. Federate Configuration Files

Content of the Federation execution definition (FED) file used by the HLA-RTI	Content of the federate configuration files (FCF) used for simulation interface configuration
• necessary class names of exchanged object classes • attribute names	• class names of exchanged objects • attribute descriptions of the classes including: + names, + type information + default values • initial values of attributes or parameters • parameter settings (e.g. max vehicle speed) • Information about sequences (process flow) • Information about output and input relations

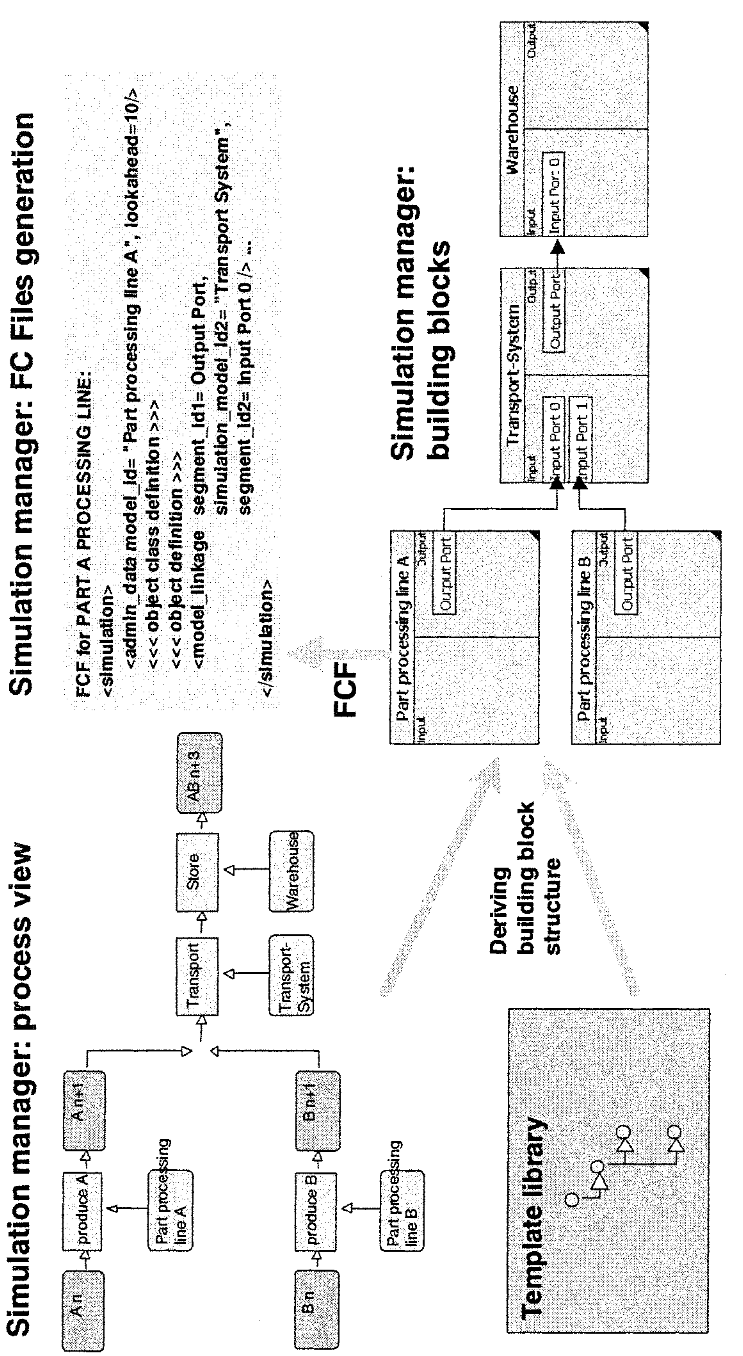

Figure 2. Generation of the Federate Configuration Files (FCF)

4. SUPPLY CHAIN APPLICATION

The MISSION methods and software have been applied within a large supply chain scenario. Specific focus has been set on the realistic representation of the order flow. Research conducted at the U.S. National Institute of Standards and Technology (NIST) in the framework of the IMS MISSION project [11] has been the used for the Exchange Objects representing this order flow.

5. BENEFITS ACHIEVED AND CONTINUED WORK

The approach described has proved to work with large applications, including real-world simulation models. Modelling is efficient, and can be performed without significant additional skills compared to an engineer applying simulation today. The major advantages achieved are:

- IPR protection, as models can be run on any server with specific access rights. Only the template has to be published.
- Reusability, even when changing simulation tools, as the single federates can be implemented by any system with a suitable interface.
- No need to agree on common simulation tools within the supply chain members, for the same reason.
- Clear and structured teamwork on large simulation models, as the single parts can be developed, tested and maintained separately.
- For large models, performance improvement is achieved by distributing computer power together with the models.

The mechanisms have been tested with a large supply chain model, which now serves as an adaptable system to configure and test supply chains very fast. However, the specification has been done for any kind of manufacturing system, and tests have been performed with a common Japanese-European example during the development work. Therefore, the mechanisms can be used for any large manufacturing simulation model, in order to improve the reliability and efficiency of the work.

Any component of this model may at any time be replaced by an application-specific detailed model. Furthermore, the interface information can be amended without the need to change any interface software. This has been proved with one application from the automotive supplier company Bosch (Germany).

The components described are ready to use. An additional approach is ongoing to apply the MISSION technology for training purposes, in cooperation with the University of Limerick (Ireland). An advanced adapter has been designed and implemented work in co-operation with the University of Bari (Italy).

6. CONCLUSION

The European module of the MISSION project has provided a modelling technology, which sets up the structure of a distributed simulation model and automatically generates all the necessary interfaces. Specific template libraries support the distributed generation and maintenance of the simulation models, and support flexibility as well as some standardisation. . As only the interfaces are published, IPR protection is given with respect to the internals of the model.

REFERENCES

1. Rabe, M.; Jäkel, F.-W. (2000). Simulation for Globally Distributed Enterprises, *12th European Simulation Symposium* (ESS), Hamburg, September 2000, pp 322-327.
2. Rabe, M.; Jäkel, F.-W. (2002). Decomposition of Large Simulation Systems for Supply Chains and Manufacturing Systems, *9th European Concurrent Engineering Conference* (ECEC), Modena, pp. 96-102.
3. IEEE (2000). IEEE P1516.1/D5. Draft Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Federate Interface Specification. IEEE.
4. Kuhl, F.;Weatherly, R.; Dahmann, J. (1999). *Creating computer simulation systems:* an introduction to the high level architecture. Upper Saddle River (USA, NJ): Prentice-Hall.
5. Schulze, Th.; Straßburger, St.; Klein, U. (1999). Migration of HLA into Civil Domains: Solutions and Prototypes for Transport Applications, *Simulation* 73 (1999) 5, pp. 296-303.
6. Woerner, J.; Laengle, T.; Woern, H. (2002). Corporate planning and Simulation of plant production facilities in the virtual world, *18th International Conference on CAD/CAM, Robotics and Factories of the Future*, Porto (Portugal) 2002. Porto: INESC Porto, 2002, pp. 109-116.
7. Rabe, M.; Jäkel, F.-W. (2001a). Non Military use of HLA within Distributed Manufacturing Scenarios, *Simulation und Visualisierung 2001*, Magdeburg, pp. 141-150.
8. Mertins, K.; Rabe, M.; Jäkel, F.-W. (2000). Neutral template libraries for efficient distributed simulation within a manufacturing system engineering platform, *Winter Simulation Conference*, Orlando (USA), pp. 1549-1557.
9. MISSION Consortium (2001). Public Deliverable 21 Input to standardisation working groups. Internet publication www.ims-mission.de.
10. Rabe, M.; Garcia de Gurtubai, G.; Jäkel, F.-W. (2001b). Modelling and Simulation for Globally Distributed Enterprises, *Proceedings of the EUROSIM 2001 conference*, Delft (CD ROM Publication).
11. Lee, Y.T.; Umeda, S. (2001). Information Model of the Supply Chain Simulation, *Proceedings of the EUROSIM 2001 conference*, Delft (CD ROM Publication).