

# Chapter 12

## Simulation-Based Enterprise Management

### Model Driven from Business Process to Simulation

Gregory Zacharewicz, Amir Pirayesh-Neghab, Marco Seregni,  
Yves Ducq and Guy Doumeingts

**Abstract** Industrial enterprises are gradually integrating Modeling & Simulation (M&S) approaches to support their management processes and to keep themselves competitive in the market by handling and connecting more efficiently their key information. On the one hand, several modeling solutions exist, with different views or abstraction levels, which are not always compatible; on the other hand, the usage of simulation for enterprise management should be aligned with the nature of decision-making. This hinders the choice of an adapted M&S solution. To facilitate the resolution of this issue, this chapter mainly proposes to apply Model Driven Service Engineering Architecture (MDSEA), which guides the usage of M&S for enterprise management at business/technical levels or with static/dynamic points of view. In its first part, the chapter focuses on different state-of-the-art elements (e.g., Enterprise Modeling, Discrete Event Simulation, etc.) which support the development of a *simulation-aided decision making cycle* for enterprise management. Simulation models involved in this cycle can be gradually created from transformation of high-level or static models. An example of such transformation is described in the second part of this chapter. The objective is to move from BPMN 2.0 (Business Process Model and Notation) to DEVS (Discrete Event Specification) which is a simulation-ready language. The second part ends by presenting a use-case and the implemented open-source software, called Service Lifecycle Management Tool Box (SLMToolBox). The chapter is concluded by discussing the propositions and the perspectives, particularly simulation of decision models for enterprise management.

---

G. Zacharewicz (✉) · Y. Ducq · G. Doumeingts  
Laboratory IMS CNRS Production Engineering Team, Talence Cedex, France  
e-mail: gregory.zacharewicz@ims-bordeaux.fr

A. Pirayesh-Neghab · G. Doumeingts  
INTEROP-VLab, Brussels, Belgium

M. Seregni  
Department of Management, Economics and Industrial Engineering,  
Politecnico Di Milano, Milan, Italy

**Keywords** Simulation · Enterprise management · M&S · Enterprise modeling · Business process · MDE · MDI · MDA · MDSEA · GRAI · Hierarchical decomposition · Decision aid · DEVS · BPMN · EA\* · SLMTool box · Model transformation · ATL · Interoperability · System

## 12.1 Introduction

To remain competitive, a company must differentiate itself from other competitors based on higher value propositions (products or services) or performant business processes. Since improving the product's performance can reach some limits, one open solution is to improve the enterprise system, redefine its processes, and share more information (considered as additional services) with customers and suppliers; the so-called servitization. Indeed, this is the role of enterprise management "*to design, control and improve the business processes in order to adapt to the changing business environment to cope with innovations, mutations of customers' expectations and increasing competition*" (Burlton 2001; Sienou et al. 2007). Therefore, management requires a proper understanding of the enterprise in order to rapidly and precisely evaluate its performance.

Facing the above needs, enterprise system complexity, particularly in the manufacturing context, has been mentioned as a real challenge since a long time (Wiendahl and Scholtissek 1994). ElMaraghy et al. have reviewed this subject, its factors and sources in the design process, products, manufacturing, and business (ElMaraghy et al. 2012). The complexity can be due to the following:

- the variety of the components' nature; human, social, technical, economic, organizational, etc.
- the numerous interactions of these components; internally, between enterprise resources and externally with the environment, customers, suppliers, and competitors.
- automation and new data exchange technologies (ElMaraghy et al. 2012) and developments in the global market.
- the excessive data and information.

Complexity is usually followed by the introduction of uncertainty and risk to the enterprise systems through creation of several internal and external factors (e.g., demand, market share, supply rate ...) which affect the performances and the final output of the system.

In such an environment, Enterprise Modeling techniques can be applied as preliminary support for management by simplifying the representation of the processes. Nevertheless, models provide solely a static abstraction of enterprise system. To go further, simulation can be necessary for providing assessments of the system performance regarding dynamicity and behavior, for both existing system and the one to be developed (Pirayesh-Neghab et al. 2011; Bruzzone et al. 2000). Therefore, joint modeling and simulation (M&S) is a necessity for enterprise management.

Considering the plethora of existing M&S solutions, one current approach consists of proposing platforms or toolboxes gathering a set of ad hoc solutions. This might help identifying solutions for a specific and clearly defined problem. Nevertheless, modeling expert is faced with issues such as lack of conceptual or technical interoperability when it is required to transpose models and simulations when moving from solution to another one. Also the model contents and simulation results are still difficult to exploit with another level of abstraction. It is mainly due to lack of methodological connection between M&S solutions. For example, (Zacharewicz et al. 2016) has stated that it remains difficult to pave the way from conceptual models to executable models (simulation).

The problem of selecting the adapted M&S is even more radical considering the different levels of decision-making (i.e., strategic, tactical, and operational) in enterprise management (Doumeings et al. 1998). Thus, in order to support enterprise management, it is important to study the way to model and decompose decisions, to connect the decisions to adapted simulations, and to aggregate the simulation results according to the decision level. Another difficulty for selecting the adapted M&S simulation is the complexity of decision-making process due to multiplicity of information, which information (e.g., performance indicator) should be taken into account for simulation; which simulation result can support the decision-making; and finally, how the results can be aggregated according to the decision-making level.

Therefore, the usage of M&S solutions should be, on the one hand, followed by a structural architecture and a methodology covering different points of view (e.g., business or technical, static, or dynamic) and, on the other hand, capable of considering and relating to the different layers of enterprise management and decision-making (e.g., from machine control to factory management in a manufacturing system). In order to select the appropriate M&S solution, a model-driven architecture can be adopted. These architectures are mainly proposed in the frame of Model Driven Engineering. For instance, Model Driven Service Engineering Architecture (MDSEA) (MSEE Book 2014), supporting the resource management to improve the enterprise performance, is discussed in this chapter. In this architecture, simulation is considered as an aid for the decision cycle. Several types of simulation can be executed such as product, information workflow, or process simulation. In MDSEA, the informational process simulation is started from higher level process models which gradually integrate technical aspects to become “simulation-ready”.

After this introductory section, an overview of research literature is addressed while briefly presenting enterprise modeling, simulation, and model-driven concepts and methods. Then a global architecture situating simulation as decision aid is presented. Within this architecture, an example of process simulation focused on model transformation from BPMN 2.0 to DEVS is also introduced including the illustration on the use-case. Finally, the perspectives of this work are proposed, particularly for the transformation of decision processes to simulation models.

## 12.2 Model-Driven Enterprise Management

This section presents the concept of Model Driven Enterprise Management which is founded on the methods of MDE and Enterprise Modeling.

### 12.2.1 MDE

MDE (Schmidt 2006) is a system engineering approach that uses the capabilities of conceptual representations of a system independently of computer technologies. MDE adopts models and languages in order to describe both the problem posed (need) and its solution. Then it goes smoothly to concrete solution.

As a structured method in the frame of MDE, MDA (Model Driven Architecture) can be mentioned. This method, defined and adopted by the OMG (Object Management Group) in 2001, then updated in 2003 (MDA 2003), is designed to promote the use of models and their transformations to consider and implement different systems. It is a four-level architecture guiding the passage from generic to specific models of a software product.

Based on MDA and in the frame of Task Group 2 (TG2) of INTEROP-NoE, the approach “Model Driven Interoperability” (MDI) considers interoperability problems from enterprise models level instead of only at the technical or coding levels (Bourey et al. 2007). The main goal of MDI, based on modeling, is to allow a complete follow, through model transformation, from expressing interoperability requirements (determination of barriers) to coding of a solution. This approach provides a greater flexibility, thanks to the automation of these transformations.

### 12.2.2 MDSEA

MDSEA is an engineering architecture which has been recently proposed in the servitization context based on MDI (Ducq et al. 2014). This architecture was developed in the frame of MSEE project (MSEE book 2014).

The main goal of MDSEA is to model enterprise system and support the development of its major components in three domains: Information Technology (IT), Human/Organization, and Physical Means (i.e., machine or any physical tool) (see Fig. 12.1). One of the originalities of the approach lays on these passages:

- from BSM (Business Service Model) level, as the high-level abstraction adapted to the business point of view, till TIM (Technology Independent Model) level, as the technical point of view regardless of the technology choice.
- from TIM level to TSM (Technology Specific Model) level, as the detailed technical level resulting in the final development of components.

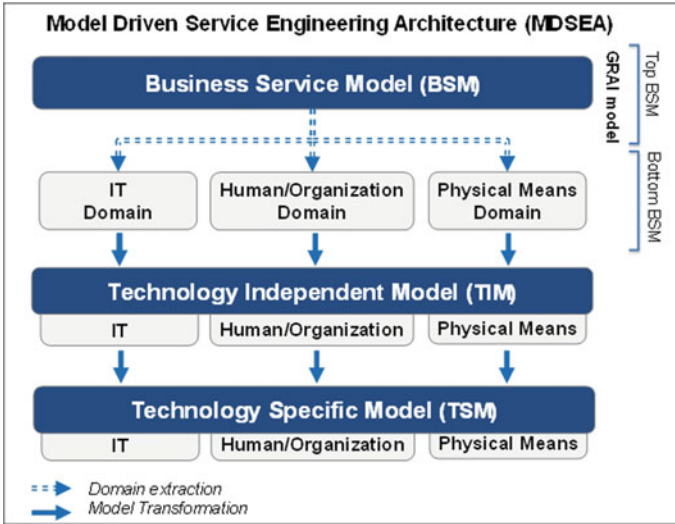


Fig. 12.1 Model Driven Service Engineering Architecture (MDSEA)

The above passages can be supported by Model Transformation which is based on a sequence of Top-Down and Bottom-Up iterations; the more we progress toward the development of the solution, the more complementary and detailed information is necessary.

MDSEA starts from an integrated and high-level modeling at BSM level (Top BSM) which consists of several integrated enterprise models elaborated based on GRAI (Graph with Results and Activities Interrelated) methodology and its formalisms. From BSM models, it is then necessary to extract the elements which will allow describing each of the three domains in order to create three main categories of resources (bottom BSM) (see “domain extraction” in Fig. 12.1). It is worth mentioning that even though these domains are focused on specific type of resource, they are not completely independent and might be overlapping. For each domain, the model at TIM level is created by a transformation of the BSM bottom using Model Transformation.

For instance, models can be elaborated with Extended Actigram Star (EA\*), which can be mentioned as a high-level business process modeling languages used at top BSM level. Concepts related to IT domain, which are necessary for the development of IT resources, are first extracted from EA\* models. Then, they are transformed to corresponding concepts of BPMN model, at TIM level (see (a) in Fig. 12.1).

### 12.2.3 Enterprise Modeling

Enterprise Modeling (EM) allows the representation of enterprise with concepts which aim at describing the strategy, the processes, the functionalities, the organization, the decisional structure, the evolution in time, the relationships with the environment (e.g., with customers and suppliers), etc. EM, through elaboration of enterprise models, supports the understanding of an enterprise system with the objective of analyzing and improving its performance. These methods can be applied not only on industrial enterprises, but also in services and public administrations such as hospitals and teaching institutes.

In order to design the enterprise model, ad hoc conceptualization or abstraction methods could be applied. Reference or standard methods could also support this task by providing a common view among different industrial branches, clarifying the current trends, key dimensions, and layers of the system.

According to a survey on the tools for system modeling, by extension enterprise modeling (Kettinger et al. 1997), there are lots of modeling languages and tools available which are capable of different aspects of a system. Kettinger et al. listed over 100 tools.

In manufacturing context, in order to be able to improve the competitiveness in the 70s, United States Department of Defense (DoD) proposed to use Enterprise Modeling Techniques (EMT) for describing a manufacturing system according to its various aspects (i.e., activities, processes, information, and simulation) (Savage 1996); the pioneer was ICAM project from which IDEF (Integration DEFINition: IDEF0 ... IDEFx) modeling method was born (Doumeingts and Ducq 2001). Since this time, several concepts, methods, and tools have been developed such as CIMOSA (Computer Integrated Manufacturing Open System Architecture), GRAI model, GIM (GRAI Integrated Modeling), EIM (Enterprise Integration Modeling), ARIS (Architecture of Integrated Information Systems), etc. (Doumeingts and Ducq 2001; Chen et al. 2008).

Another approach for representation of enterprise structure is Enterprise Architecture which is more ICT-oriented. Such architectures can be developed to provide an abstract view of the ICT structure. A system architecture is defined as “*a conceptual model of a system together with models derived from it that represent (1) different viewpoints defined as views on top of the conceptual model, (2) facets or concerns of the system in dependence on the scope and abstraction level of various stakeholders, (3) restrictions for the deployment of the system and description of the quality warranties of the system, and (4) embedding into other systems*” (Jaakkola and Thalheim 2011). A survey of the viewpoints from the most popular enterprise architectures (e.g., Zachman, Sagace Matrix, DoDaf 1.5, Pera, MDA, AFIS, and Pahl & Beitz) has been performed in Benkamouna et al. (2014).

In GRAI method (Doumeingts 1984; Chen and Doumeingts 1996), a system or particularly an enterprise system is decomposed, into three subsystems according to the System Theory (Le Moigne 1977), Management Decision (Simon 1969), and

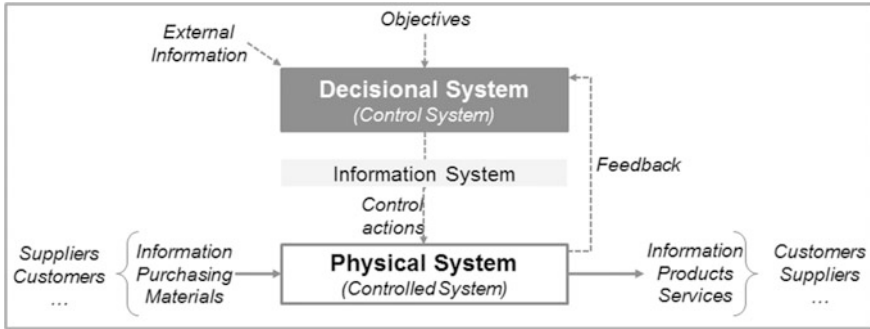


Fig. 12.2 GRAI model for integrated BSM model (top BSM) of MDSEA

Theory of Hierarchical Multilevel Systems (Mesarovic et al. 1970). A brief description of these subsystems is given below and shown in Fig. 12.2:

- The controlled subsystem (physical subsystem) transforms the inputs (materials and information) into outputs (new information, products or services) to be mainly delivered to the customers.
- The control subsystem (decisional subsystem) manages the physical subsystem based on the objectives of the global system (e.g., enterprise system) and the feedback information in order to delivers actions or adjustments.
- The information subsystem includes information from the physical subsystem and from the customers, suppliers, and other stakeholders (external environment).

### 12.2.3.1 Decisional Modeling

Regarding the representation of decisional subsystem, hierarchical decomposition and aggregation of information is a real necessity and challenge, particularly in manufacturing as complex enterprise system (see Fig. 12.3). This decomposition describes the different decision-making level (i.e., strategic, tactical, and operational). Furthermore, the decomposition highlights the required information to be provided by decision aids (e.g., simulation).

For a decision-maker at operational level of enterprise system, the scope and time span of decision-making is small (e.g., daily scheduling of a machine) and usually detailed information (e.g., machine capacity or machining time) are required which correspond to the reality. However, a decision-maker in charge of tactical or strategic levels is in charge of decisions (e.g., annual production planning) which covers numerous information with larger scope and time span (e.g., annual material requirements).

Another reason for a proper decisional modeling is cognitive limitation; the quantity of information that a decision-maker is able to process in a unit of time should be limited (Doumeingts et al. 1998).

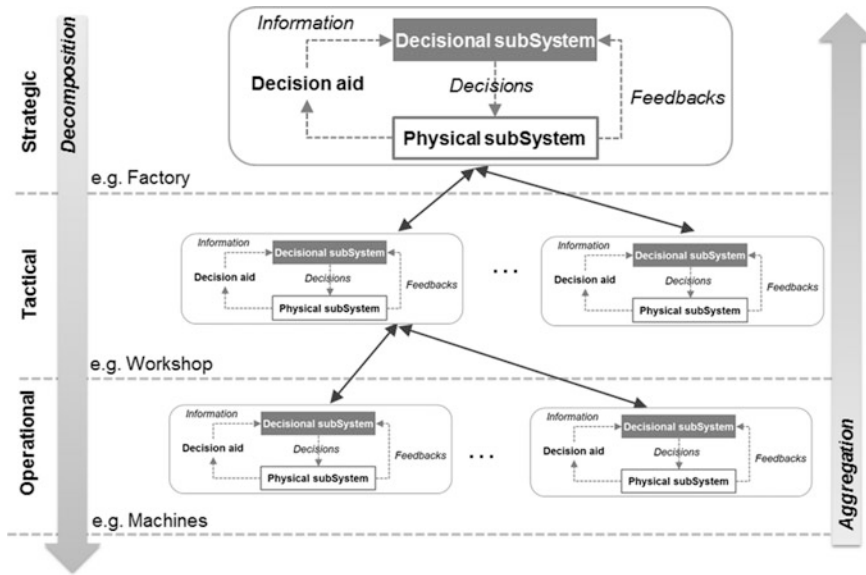


Fig. 12.3 Decomposition/aggregation of decisional subsystem

In GRAI model, the decisional modeling is performed using the GRAI Grid which takes into account the enterprise layers according to the decision complexity. Such layers are based on the concept of horizon, the timespan of interest for the decision, and period, the timespan for (re)-evaluating a decision to find deviations from the expected pattern. *“The GRAI Grid does not aim at the detailed modeling of information processes [for decision making], but puts into a prominent position the identification of those points where decisions are made in order to manage a system”* (Doumeingts et al. 1998). The characteristic of the GRAI Grid is based on Decision Management (Simon 1969) and Theory of Hierarchical Multilevel Systems (Mesarovic et al. 1970). It should be mentioned that the GRAI Grid is coupled with GRAI Nets which provide a more detailed perspective about activities forming the decision-making process and the relationships among them.

Considering the dichotomy of decisional/physical subsystems in an enterprise (see Fig. 12.3), each decisional level is in charge of controlling a specific part of the physical subsystem. Therefore, the decomposition/aggregation of decisional modeling should be aligned with the physical subsystem modeling (e.g., process models). Indeed, the model of the physical subsystem is different at each level of decision using aggregated information where criteria of aggregation must be properly determined.



### 12.2.3.2 Business Process Modeling

As an example of methods for representing the physical and information subsystems (see GARI Model in Sect. 12.2.3), Business Process Modeling (BPM) can be mentioned (Cardoso et al. 2012). BPM results in a representation of an organization's business processes to be analyzed and improved (Weske 2007).

The world of business processes has changed dramatically over the past few years. Processes can be coordinated from behind, within and over organizations' natural boundaries. A business process now spans multiple participants and coordination can be complex. Business process models can help business actors to handle the problems of heterogeneity, complexity, and flexibility in layered operational Enterprise Architectures and across the enterprise knowledge spaces of network life cycles. Business process is a structured, measured set of activities designed to produce a specific output (product or service) for a particular customer or market (MSEE D15.2 2012). A process is thus a specific ordering of work activities across time and space, with a beginning and an end, and clearly defined inputs and outputs: a structure for action.

One of the most commonly used approaches for BPM is Object-Oriented (O-O). This approach is based on a set of object classes which substitute the behavior of real process components (Anglani et al. 2002). Through O-O modeling the real process components are categorized into three different types of flows: materials, information, and decisions (Chen and Lu 1997). Chen and Lu presented an O-O oriented methodology using the Petri nets, the ERD (Entity Relationship Diagram), and the IDEF0 (Integrated DEFinition language for functional modeling) (Chen and Lu 1997).

Another commonly used languages is called BPMN (Business Process Model Notation) which is a standard notation for BPM. BPMN is supported by the Object Management Group (OMG 2003). Its objective is to provide a framework to describe a process in a way that is common to all users, irrespectively of the tool used. The tool is of course supposed to support the standard that is currently BPMN 2.0.

In MDSEA architecture (see Fig. 12.1), BPMN 2.0 is applied at TIM level for the IT domain. However, Business Process modeling is ideally started from a higher abstraction level (at BSM top level) using Extended Actigram Star (EA\*). The main objective is to provide a common and simple modeling notation to the business user through an accessible syntax (MSEE D15.2 2012). Rather than EA\*, other languages such as Archimate Business Level modeling can be also applied for this purpose. However, considering the focus of MDSEA on the management, development, and interoperability of different categories of resources, we believe that EA\* might be a better choice since on the one hand, it distinguished the category of the resources and on the other hand, it clarifies the contents of flows (e.g., information, physical, sequential) in the process model.

In comparison to BPMN, EA\* might reduce more the gap between the ideation, from user point of view, and the design of business process. EA\* models can be transformed to BPMN model (MSEE D15.2 2012). In this research work, we

directly apply BPMN models which are enriched after obtaining their skeleton from EA\* models. The drawback of BPMs is due to the fact that it provides only a suitable static view which is missing the temporal dimension to express output performance such as an expected cost or a desired duration. In detail, the impact of correct or incorrect behavior of complex models over time is not clearly visible using static view. This issue can be solved by running a business process simulation for analyzing and understanding the business process model according to its dynamic.

BPMN is frequently associated to Business Process Execution Language (BPEL) (Thatte et al. 2003), that is a programming language for running business processes. Nevertheless, BPEL is intended to execution rather than simulation and it is not associated to clear execution semantics. The notions of states, dynamics remains open to the interpretation of the modelers potentially subjective so difficult to reuse and compare. Formal modeling theories can overcome this limitation by extending the credibility of the simulation models. It provides a sound ground for comparing the results and opening interoperability with different simulation platforms.

#### ***12.2.4 Simulation for Enterprise Management***

Simulation solutions are designed according to different enterprise needs such as tracking performance indicators or providing information about the real behavior of products, services or processes, in a didactical and pedagogical way to support decision-making.

Manufacturers, for instance, can shorten time needed to develop new products/services and to (re)-engineer business process-related activities. At the same time, simulation techniques are expected support them in gaining new customers, lowering costs, improve business processes, increase customer satisfaction, and getting access to know-how. Thus, simulation techniques are now orienting toward customized applications in the form of Software-as-a-Service (SaaS), based on visual modeling tools to interactively create a model of the reality of interest for the specific user.

Simulation, and by extension M&S, as a service can be expected as a pre-validation on a perspective scenario of enterprise management. It is run on models of future enterprise management, and this is done by anticipation using probable and credible set of data. The simulation requires significant amount of data and resources to run the process and to consider the behavior of the process activities. Cloud-based approaches and High-Performance Computing (HPC) give more and more support to enhance these simulations. Nevertheless, the collaboration between HPC cloud infrastructure providers and Engineering Manufacturing users is often difficult (confidentiality, interoperability issues) and requires long times and huge efforts. So the remaining question is how to mobilize companies and

particularly SMEs and midcaps to benefit from simulation digitization and HPC facilities to improve their competitiveness?

Several European projects (e.g., CloudSME, Fortissimo2, etc.) have been devoted to cope with the aforementioned challenge. For example, the project CloudSME ([www.cloudsme.eu](http://www.cloudsme.eu)) supports SMEs to utilize. SMEs require the development of models involving their activities. Therefore, CloudSME proposes models that are simulated numerically by using either continuous or discrete event simulation techniques. The modeling and simulation processes take advantage of visual modeling tools to interactively create a model of the reality of interest. Fortissimo and Fortissimo2 ([www.fortissimo-project.eu/](http://www.fortissimo-project.eu/)) are collaborative projects that enable European SMEs to be more competitive globally through the use of simulation services running on HPC cloud infrastructures. The Fortissimo project (through its Marketplace) provides a number of approaches required by SMEs to find solutions to their challenges, which are mainly on-demand access to advanced simulation and modeling resources, and access to state-of-the-art HPC facilities, leading to a reduced computation time.

SMEs involved with the solutions provided by the two presented projects have benefited of a reduction of the design costs, thanks to the use of a set of simulation software ported to HPC system available through the cloud. Especially it appeared that for the use of a federation of heterogeneous simulations a workflow or process model was required to cope with the different solutions and services. Nevertheless, the modeling of this workflow and its transformation to simulation model is not simple and even explicit enough to be given directly to the usage of SME. Discrete event simulation can be a solution to model formally, and then orchestrate such a workflow of M&S in the aim to drive anticipation of enterprise management scenarios.

#### 12.2.4.1 Discrete Event Simulation

Discrete Event Simulation (DES), which is a frequently used method in process M&S, significantly facilitates the enterprise management process definition and validation (Pirayesh et al. 2011). Semini et al. (2006) performed a literature survey on use of discrete event simulation in real-world manufacturing & logistics decision-making in 2006. According to this survey there are several reasons why a simulation study can support manufacturing and logistics decision-making:

- provides better understanding of the real system and its behavior.
- reveals previously hidden relationships.
- performs a systematic analysis of the situation.
- facilitates communication and provide a basis for discussions.
- allows the decision-maker to test the influence of different alternative scenarios without having to make changes in the real system.

Another review on simulation in manufacturing and business was presented by Jahangirian et al. in 2010. In this survey simulation is recognized as the second most widely used technique in the field of operation management and it has been applied in areas such as manufacturing, services, defense, healthcare, and public services. In real-world applications there are factors to be considered in selecting a proper simulation technique. Jahangirian et al. suggest that in case of dealing with different layers of decision-making within a system, a better understanding will be needed of the relationship between the different layers of organizations and of the way to connect simulation tools that relate to each layer in order to deal with the system as a whole (Jahangirian et al. 2010).

Simulation has been a widely used tool for manufacturing system design and analysis. It has proven to be an extremely useful analysis tool, and many hundreds of articles, papers, books, and conferences have focused directly on the topic. Smith presented a classification of a subset of these publications and the researches and applications that underlie these publications (Smith 2003).

As an example, in the nuclear industry, Monte Carlo simulation is proper for the study of system availability/reliability and component importance. Monte Carlo simulation involves no complex mathematical analysis and is preferred to deterministic methods which are difficult to solve specially in case of large and complex systems (Wu 2008).

Zeigler has proposed since 1976 the Discrete EVent Specification (DEVS) (Zeigler et al. 2000) as an integrated formalism which enhances the model designing efficiency with unambiguous specification formalism and provides a methodology for execution process by means of an executable semantics. We have chosen DEVS as the simulation language for the reasons enounced previous in order to remove ambiguity and unify the M&S concepts.

#### 12.2.4.2 Simulation Tools

There are numerous software products in the simulation field. Semini et al. identified the papers using DES software tool in a literature survey. For instance they reported in the use of several relevant application papers from the last decade of Winter Simulation Conference proceedings. Arena and Automod/Autosched were used most frequently, followed by Quest, ProModel, Sigma, and Extend (Semini et al. 2006). Also simulation tools are developed in the academic and/or open source context (NetLogo, MS4ME, VLE, etc.).

DEVS supporting tools deserve a particular attention since this language has been selected for targeting simulation within this chapter. The DEVS group standardization maintains on its website the updated list of most used DEVS tools known by the DEVS community (Wainer 2013). In Hamri and Zacharewicz (2012), the authors have given a brief description and comparison of popular tools.

ADEVS was the first DEVS tool developed. DEVSJAVA is a Java framework in which the kernel simulator is ADEVS. CD++ Builder is a DEVS modeling and simulation environment that integrates interesting features and facilities for the user.

Other DEVS tools are dedicated to specific areas. VLE is a C++ M&S framework that integrates heterogeneous models from different scientific fields. This integration is based on the agent paradigm. In addition, JDEVS is the Java implementation of a DEVS formal framework. It supports multi-modeling paradigms based on DEVS. It ensures the interoperability among the reused components. Also SIMSTUDIO can be considered, and it is focused on a simplified DEVS editor for DEVS non-Expert. The authors also investigate LSIS\_DME that is focused on a graphical interface and code source generation in order to complete the model by complex Java functions.

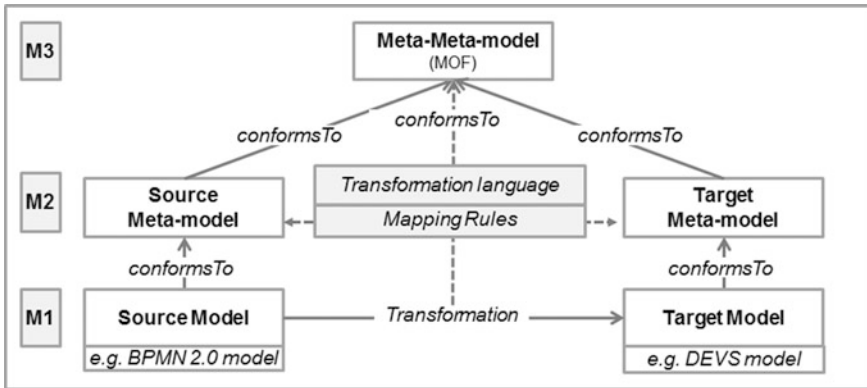
At the end each DEVS editor is covering interesting aspects that complete basic DEVS facilities or propose different model views. Nevertheless, we found it difficult to import by the tool non DEVS models other than hard coded matching, i.e., the customization is limited. We suggest that the feeding by other model can be facilitated if following a Model-Driven approach, e.g., MDA. One core concept of MDA is the Meta Model that is required for model matching, an example of which has been proposed by Garredu et al. (2012).

### 12.2.5 Model Transformation

Considering the diversity of actors in product engineering, several heterogeneous standards or modeling languages, with different purposes, are applied. Therefore, treatments on models should be considered in order to achieve interoperability in model exchanges (Pirayesh et al. 2015). For this purpose, Model Transformation, defined as the process of converting a [Product or Process] Data model to another model of the latter (Miller and Mukerji 2003), can be mentioned. It is indeed considered as a common interoperability solution in MDE and is classified as a federative approach. A taxonomic classification of the various existing approaches for Model Transformation is proposed in Czarnecki and Helsen (2003). The authors also classify existing approaches of transformation as follows: direct-manipulation, relational, graph-transformation-based, structure-driven, and hybrid approaches.

Figure 12.4 shows the Model Transformation architecture, in the frame of Meta Object Facility (MOF) of OMG. In this architecture,

- A source model is transformed into a target model. These models, in M1 level, are in accordance (conformsTo) with their own meta-models of the M2 level, and meta-models are consistent with a single meta-meta-model (M3 level).
- Transformation is based on rules (mapping or projection). Indeed, the transformation is based on semantic and syntactic relations between models, which are developed by domain experts of these models.
- The transformation also requires a transformation language that implements the transformation rules. This language is itself conforming to level M3 (meta-meta-model) in the transformation architecture.



**Fig. 12.4** Model transformation architecture and its key elements

As it is mentioned above, one of the key elements of Model Transformation is Data Abstraction. In abstraction, concepts and conceptual relationships are created as conceptual models (Lezoche et al. 2011). After abstraction and formalization, the mapping can be started as an operation that defines the transformation rules between a pair of representations [meta-models] (Shvaiko 2005). Alignment is the main part of mapping and the basic task of a mapping operator. A more complex mapping process is described in Bouquet et al. (2005).

Once the mapping is defined, a language is required for realizing the transformation. As an example of such language, ATLAS Transformation Language (ATL 2013) and MISTRAL (Kurtev and van den Berg 2005) are widely used in the context of MDE. ATL is a model transformation language specified as both a meta-model and a textual concrete syntax. It complies with MOF and provides a way to generate the target model from the source model for developers in MDE. ATL provides developers with a mean to specify the way to produce a number of target models from a set of source models.

Currently, there are several toolkits (e.g., Topcased), used as integrated plugins in the Eclipse platform, which support this language. These toolkits also allow the implementation of the transformation of XML documents as well as MOF or Ecore meta-models (Lu 2012).

In MDSEA architecture, according to the distinction of static and dynamic process modeling (Cardoso et al. 2012), the focus will be on a complementary step at TIM level. This step concerns transformation of static business process models (e.g., BPMN 2.0 model (OMG 2011), at TIM level, to a simulation model) able to analyze the behavior of the system (see Fig. 12.4). Based on the feedbacks provided by the simulation, the high-level process models (EA\* model) can be modified. This cycle will continue till obtaining the most performant configuration of the system and its processes.

### 12.2.6 Simulation as Decision Aid for Enterprise Management

Considering the structure of enterprise system described in Sect. 12.2.3, its decisional subsystem receives information about the physical subsystem at various levels of management. Such information can be collected from the real operational processes, from the process models with different abstraction levels (the so-called digital twins), or from the simulation of the models. In such case, a simulation-aided decision cycle is formed (see Fig. 12.5). Here, we emphasize on the importance of a hierarchical structure, covering, and connecting different enterprise levels, for the use of simulation tool in a decision aid approach. In enterprise systems, it is not always possible to simulate the processes at the operational level due to the amount of information and the run time.

In the proposed approach, the first step (see (1) in Fig. 12.5) of the decision-making cycle is started by the decomposition of the decisions and the information [e.g., simulation needs and Performance Indicators (PIs)] supporting those decisions. This step can be performed using decisional modeling methods such as GRAI Grid (see Sect. 12.2.3). Then, the simulation solution should be selected according to the required information (see (2) in Fig. 12.5). For instance, in a manufacturing system, the decision at strategic level can be about the choice of suppliers. Therefore, in case of lacking historical data, the simulation solution is intended to provide an overall estimation of the consumed materials in a period of time.

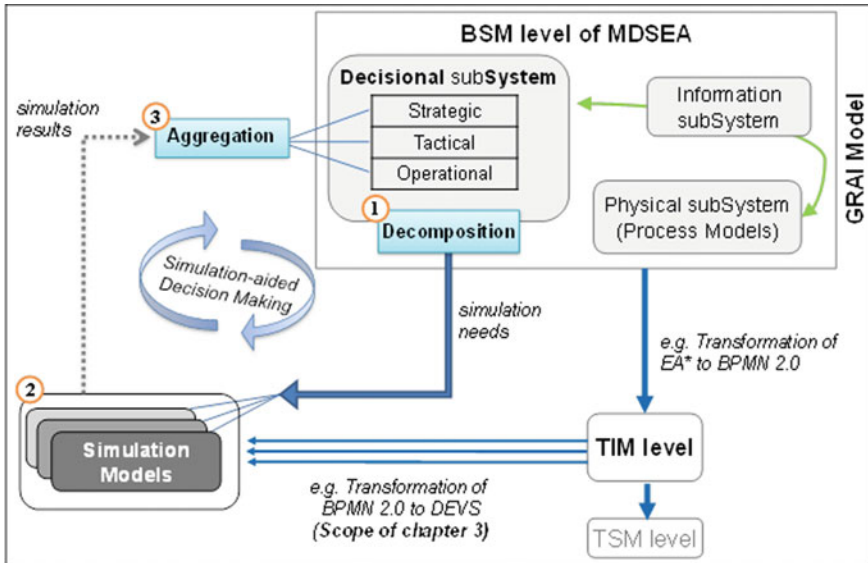


Fig. 12.5 Simulation-based decision aid for enterprise management in MDSEA

After the simulation, it is usually required to aggregate the results according to the same criteria of decomposition or enterprise layers (see (3) in Fig. 12.5). In the above example, the information about raw material consumption should be classified and aggregated based on annual consumption, category of product, overall cost of quality, etc.

The simulation models enabling the “*simulation-aided decision making cycle*” can be the result of transforming physical subsystem models (e.g., process models). The modeling work can be guided by Enterprise Modeling techniques.

As explained previously in Sect. 12.2.2, such modeling is often started at a high-level abstraction adapted to business point of view (BSM level of MDSEA). For instance, according to the decisions and objectives at strategic level of GRAI Grid a, process modeling can be performed using EA\* language. Then, models can be transformed with two main purposes;

- enrichment of technical capacities: this is considered as a vertical transformation, from one level to another, in MDSEA architecture. EA\* to BPMN is an example of transformation from BSM level to TIM level.
- provision of dynamic capacities: this is considered as a horizontal transformation, at the same level, in MDSEA. As an example of the latter, BPMN 2.0 to DEVS transformation can be mentioned which occurs at TIM level (from TIM-static to TIM-dynamic). This example is discussed in the following section of this chapter.

## 12.3 From Business Process to Simulation

Developing a high-level process model (business process model) before the development of the simulation model helps the recognition of the operation and also it is a time and cost saving act (Nethe and Stahlmann 1999). At design (or build time), it exists many process modeling languages. Yet, there are several reasons to choose BPMN among different formalisms. First, it is standardized by OMG and widespread in the industrial domain. Then, it can be generated from higher level languages such as EA\*. Finally, it is associated to a set of execution languages. Moreover, it is important to adopt a federative formalism that can group the concepts of simulation and DES to be shared between different authors where the difference is the notations. For this purpose, DEVS language is selected which embraces a very large scope of domain.

This part presents the main principles of model transformation from Business Process to Simulation, based on the example of BPMN model to DEVS model, including the transformation architecture, DEVS meta-model, the mapping of concepts, and the implementation using a transformation language.



### ***12.3.1 Background***

In the context of BPMN to DEVS transformation, authors in Cetinkaya et al. (2012) and (Mittal and Risco Martin 2012) presented a Model Driven Development (MDD) framework for modeling and simulation (MDD4MS). In the frame of this framework they defined a model-to-model transformation from BPMN as a conceptual modeling language to DEVS as a simulation model specification. BPMN and DEVS meta-models were presented. In addition, a set of transformation rules were defined in order to transform BPMN models into DEVS models. According to these rules, some BPMN concepts (Pool, Lane, SubProcess) were mapped to DEVS coupled component, while Task, Event (Start, End, and Intermediate), and Gateway were mapped to DEVS atomic component.

Comparing the BPMN meta-model defined with the latest version of BPMN 2.0 meta-model (OMG 2011) we can conclude that several concepts are missing and thus were not transformed into their corresponding DEVS concept. Authors did not mention the different types of BPMN Tasks (User Task, Manual Task, Service Task...) and BPMN Intermediate Events (Message, Signal...) that can be mapped differently when transformed into DEVS concepts. The difference would be in the number of states forming each DEVS Atomic Model. Based on these remarks, the work presented in this chapter takes into consideration these points in an attempt to benefit from previous work and propose new mapping and transformation rules.

### ***12.3.2 Transformation Concepts***

The meta-model approach (OMG 2003) is one of the most used transformation techniques (Fig. 12.3). It has been adapted by Bazoun et al. (2013) to the context of model transformation from BPMN 2.0 model to DEVS model. Three different levels are identified: model, meta-model, and meta-meta-model. The BPMN model is the source model to be transformed, while the DEVS model is the target model resulting from the ATL transformation. BPMN and DEVS models conform to the BPMN 2.0 and DEVS meta-models, respectively. In addition both meta-models conform to a meta-meta-model named Ecore (McNeill 2010) meta-model (developed using an Ecore-based modeling framework). A mapping is implemented by ATL between the concepts of BPMN 2.0 and DEVS.

### ***12.3.3 Meta-Models***

Source and target meta-models should be well identified to proceed with the transformation (see Fig. 12.3). BPMN 2.0 meta-model specified in OMG (2011) is the source meta-model. There is no endorsed meta-model for the target DEVS



To complete this approach, different types of task have been proposed (Receive task, Send Task, User Task, Service Task, and Manual Task); all of these tasks are mapped to “DEVS Atomic Model” concept but with different local behaviors. This is also applied to intermediate events (Receiving and Sending Messages). (Zacharewicz et al. 2008) has defined different task models. A basic task is an activity where a work is performed by a resource. For a more accurate matching between BPMN model and DEVS model it has been proposed in Bazoun et al. (2014) and then (D’Ambrogio and Zacharewicz 2016) to distinguish the “Reception Task” from the “Basic Task”.

Also we clearly distinguish between tokens and messages. The structure of tokens and messages is a multi-value event as described in G-DEVS (Zacharewicz et al. 2008) that is implemented by one object with several variables. Each variable is representing one data. The notion of Event is used to represent something that “happens” during the execution of the process. It represents a step in the process and its meaning differs from DEVS event. These events affect the flow of the process. There are three types of events, based on when they affect the flow: Start Event, Intermediate Event, and End Event. In this paper we will present an example of an Intermediate Event; Intermediate Reception Event. Some information of the token will be updated by the workflow according to actions defined in the task, current values of the token, and message received. At the end, the token reflects the path taken, the duration, etc. All the data are tracked in order to compute some performance indicators. This chapter will not detail each concept, but only the most relevant are described in the following.

Table 12.1 presents a non-detailed mapping between BPMN and DEVS. It shows new concepts (\*) added regarding the previous approaches in the literature introduced in Sect. 12.2.1.

This conceptual mapping has been implemented into transformation rules using ATL transformation language. Each atomic component is generated from the BPMN model than the generated components are assembled in the coupled model.

## ***12.3.5 Tooling and Implementation***

### **12.3.5.1 Transformation Language**

An ATL M2 M (eclipse) component has been developed in the Eclipse modeling Project (EMP). The ATL Integrated Environment (IDE) provides a number of standard development tools (syntax highlighting, debugger, etc.) that aims to ease development of ATL transformations. The ATL project includes also a library of ATL transformations. ATL M2 M is also used for compliance reason with SLMToolBox (presented in the next section) both developed under Eclipse. The more exhaustive transformation rules and specifications have been introduced in a technical paper (Bazoun et al. 2014) presenting the mapping details.

**Table 12.1** BPMN elements to DEVS components

BPMN	DEVS
Pool	DEVS Coupled Model
Lane	DEVS Coupled Model
Sub process	DEVS Coupled Model
Flow Message Flow* Sequence Flow*	DEVS Atomic Model
Task Basic Task Send Task* Receive Task*	DEVS Atomic Model
Event Start* {Message, Timer, Conditional} Intermediate* {Message, Signal, Conditional} End* {Message, Timer, Conditional}	DEVS Atomic Model
Gateway Exclusive Gateway Inclusive Gateway* Parallel Gateway	DEVS Atomic Model

### 12.3.5.2 SLMToolBox

SLMToolBox (Boye et al. 2014) is a software tool developed in the frame of MSEE project. The SLMToolBox will be used by enterprises willing to develop a new service or improve an existing one, within a single enterprise or a virtual manufacturing enterprise. The tool will be used at the stage of “requirement” and “design” of the service engineering process. The SLMToolBox is regarded to be an integration of several scientific concepts related to services into one tool. These concepts can be summarized by MDSEA methodology, services’ modeling, engineering, simulation, monitoring, and control.

The simulation feature is based on model transformation from BPMN to DEVS models. Source BPMN model is extracted from the BPMN graphical editor (integrated in SLMToolBox), a transformation engine is implemented based on ATL, and the output of this engine is DEVS model. A new developed version of (Zacharewicz et al. 2008) will be integrated in the SLMToolBox for graphical visualization and simulation of DEVS models.

### 12.3.6 Use-Case

One use-case model from the MSEE European project has been reused to serve in this research as a case study. The process consists in the creation of a cloth patron adapted and fitted to each client by tailoring, thanks to customer data.

In MSEE project, the modeling is starting from BSM level with an Extended Actigram model. Then the next step is going down to the BPMN model at TIM level. At this level before the creation of service from the model it could be valuable to simulate its behavior in order to correct potential errors of conception that can be detected through dynamical aspects not seen by reading a static model. The next part of the section will focus on the transformation to the simulation model.

One extract from the BPMN model is detailed in top of Fig. 12.6. Two pools representing client and manufacturer are described in the use-case. In particular, the sequence and the messages exchanged with the client are considered. The distinctive contribution of this research work permits first to differentiate the type of BPMN event. For instance the model shows an intermediate “Message Event”. In addition, the task 1 is emitting a message to another blind pool (with basic a reception and triggering behavior). We consider this possibility as expressing representatively BPMN 2.0 collaboration model.

At DEVS level, the LSIS\_DME editor (Zacharewicz et al. 2008) was tentatively selected to perform tests on the DEVS models obtained from BPMN matching before moving to final development stage, to the DEVS engine of the SLMToolBox. One interest for the tool comes from the fact that it enables the creation, storage library, modification, and composition of XML-based models that can be feed in our case by the ATL transformation from ATL BPMN models. Also, the editor allows editing visually a model with geometric shapes representing the different elements of a DEVS atomic or coupled DEVS model.

Mapping realized the DEVS Coupled Model based on the library developed from BPMN components (Table 12.1) and integrated in the LSIS\_DME DEVS models library of BPMN diagram. The DEVS coupled model presented is the transformation results of the selected extract from BPMN model of use-case (see Fig. 12.7). Each atomic DEVS component is selected from the library and instantiated according to data values coming from the BPMN description. Then the models are coupled to represent the BPMN chain of tasks and it take into account resources represented by lanes. In this example we differentiate between a fully described lane and another non-detailed lane (blind lane).

Then Fig. 12.7 has been run to present an extract of the simulation results provided by the tool. In this simulation it was confirmed that the token variables declared in the initial state of each “start event” atomic model can be followed in term of evolution of their attributes values accordingly to activities actions of the process and regarding time. The new values depend on the operation of the task and message received. The main idea resulting from the first simulations performed is the proof of feasibility in terms of definition and monitoring of quality indicators, the capacity to measure the impact of input factors and parameters. The goal is to provide simulation feedbacks to parameters tuning to reach as close as possible the services desired results.

At the moment, results are not handled to be displayed graphically nor interpreted by BPMN. The simulation has been set up to follow performance indicators on tokens that circulate through the different components of the process. Tokens gather information on service development and its delivery; they can be considered

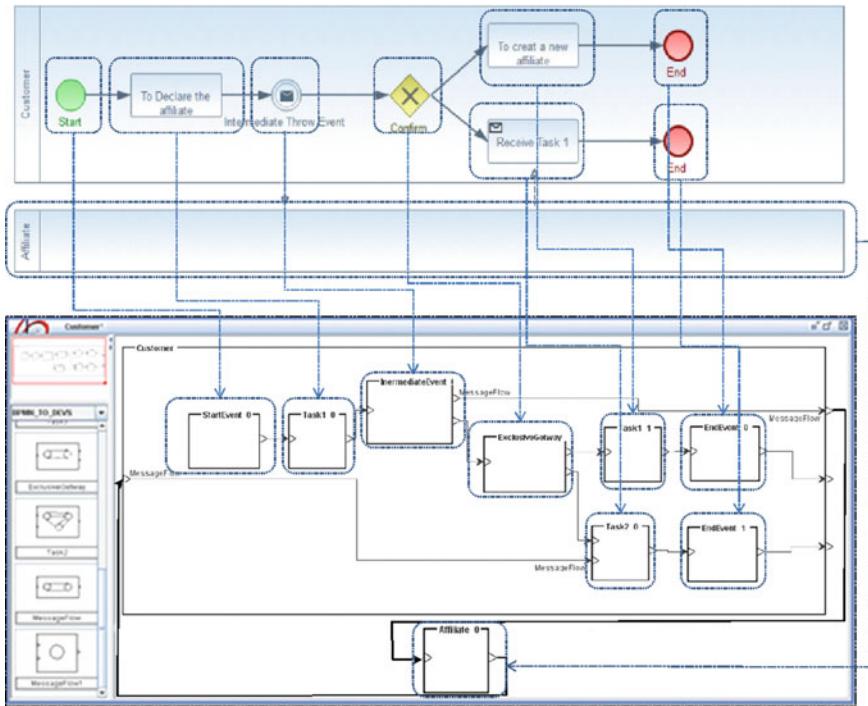


Fig. 12.7 Equivalent DEVS model example in LSIS DME

as the memory of service development. For instance, the time to complete the service delivery can be traced during the simulation. The number of resources called to achieve the service delivery process and the cost of material and human resources can be computed using the simulation. Another point is to analyze failure in the service delivery. Some service building can lead to bottlenecks. Several scenarios can be proposed and run to evaluate the best one before the next implantation step: the architecture implementation.

Nonetheless, these results are of paramount importance for decision modeling, since they provide a broader set of information (e.g., historical events, PIs and What-If scenarios) to a decision-maker.

### 12.4 Discussion and Perspective

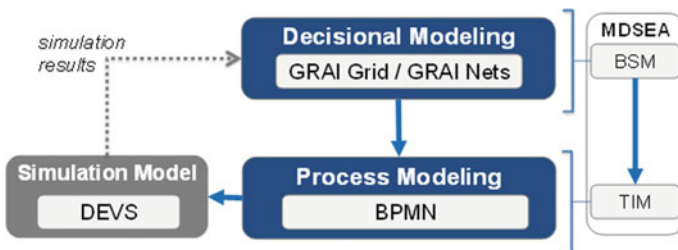
Within this work, the correlation between simulation and decision-making as one of the main applications of simulation for enterprise management was discussed. Several fundamental elements, required for developing simulation-based decision-making cycle, were also presented such as GRAI decisional model.

GRAI easiness to position a decision at a specific level as well as its capability to provide a global view of the decisional perspective represent the major advantages of such an approach (Carrie and Macintosh 1997). Thanks to these characteristics, it has been applied for a large scope of different purposes (Noran 2012), such as service monitoring (Taisch et al. 2014) or analysis of business model of enterprise network (Álvares-Ribeiro et al. 2004), performance evaluation (Ducq and Vallespir 2005), and information and manufacturing system alignment (Goepf-Thiebaut and Kiefer 2008).

Some critics have been raised on this method, particularly due to the lack of dynamic modeling of information systems which does not allow modeler to show the effects of delays in decision (Carrie and Macintosh 1997). Álvares-Ribeiro et al. (2004) integrated GRAI Grid together with Zachman framework, particularly with its “Business Model” dimension. However, this conceptual integration appears limited in coping with the static nature of GRAI modeling. A more promising approach is to interconnect GRAI Grid and GRAI Nets models with simulation models. Some authors have worked toward this direction, attempting to interconnect GRAI Grid with simulation model, such as (Al-Ahmari and Ridgway 1999). Only DGRAI, an evolution of GRAI for decision systems design and monitoring, combined simulation also with GRAI Nets (Poler et al. 2002).

Inspired from this work, this chapter proposes a “*simulation-aided decision making cycle*” as an approach for coupling decisional modeling with simulation in the frame of MDSEA architecture. Within this cycle, two model transformations are to be clearly defined (see Fig. 12.8): (1) from GRAI Grid/Nets to BPMN 2.0 and (2) from BPMN 2.0 to DEVS model.

Regarding the first transformation (from GRAI Grid/Nets to BPMN 2.0), few examples of translation GRAI Grid formalism into business process modeling languages can be found in the literature. A mapping of GRAI languages for semantic translation into ULM Activity and Class diagram has been proposed (Seguer et al. 2010). There are not works aiming at translating this decisional formalism into nor BPMN neither other modeling formalism and this transformation has been addressed only conceptually in this work. A concept mapping and transformation rules are required to implement the “*simulation-aided decision making cycle*” and thus support decision-making under different perspectives. For



**Fig. 12.8** Transformation path from decision process modeling to simulation

instance, during the running of the decisional system, simulation will help also in identifying critical activities and bottlenecks on which targeting interventions (e.g., additional supporting resources, changes in the triggers of a decisions, etc.).

GRAI Nets formalism can be also further extended with new concepts coming from decision-making theories. For instance, concepts such as decisional roles can be included based on Organizational Buying Behavior (OBB), a widespread theory used for complex decision involving groups or individuals. In practical terms, this might be done introducing a new modeling component representing a human resource, specified by an attribute “role”, which can assume values like “users”, “proposer”, “influencer”, “decider”, and “gatekeeper”. Rules and constraints in the association with the other modeling elements of GRAI Net would be then defined according to the value of this attribute

For the second model transformation (from BPMN 2.0 to DEVS) concepts, meta-models and a concrete implementation have been proposed and implemented. It remains to visualize the DEVS models resulting from the transformation to be later displayed in a DEVS Graphical editor completely integrated in the SLMTToolBox. The DEVS meta-model will be completed independently from any simulator’s architecture to take into account multi-value state variables. In addition, new features such as export format will be developed. Storage will be improved. Authors claim that the durability of this work relies on the adoption of the open platform. In addition, BPMN models (subject of simulation) will be animated for better understanding of the process. Thanks to the visualization of DEVS models, users will be capable of tuning more precisely performance indicators’ values (time, costs, and combined indicators) needed for simulation. The simulation results offer sufficient information needed for business process analysis, but the problem frequently faced is the lack of temporal data from enterprises because of the domain no long experience.

## 12.5 Conclusion

This chapter highlighted the interest of M&S for Enterprise Management. It is intended to guide enterprise managers in the choice of appropriated M&S solutions. The chapter has also stated that models and simulations have to be closely linked; vertically, from business to technical level, and horizontally, from static to dynamic views. In these links interoperability should be ensured to preserve the information when changing or coupling different paradigms or tools. For this purpose, the interest of using architectures, such as MDSEA, based on model-driven approaches was presented. Then the chapter has focused on decisional and business process modeling and simulation in the frame of this architecture. The benefits of simulation for enterprise management and recent trends were also discussed with a focus on *simulation-aided decision making cycle*. As an example of creating simulation models in this cycle from transformation of static models, the chapter presents a transformation of BPMN models into DEVS models. It described the



transformation architecture, mappings, and an implementation in an open-source tool (SLMToolBox). The development of the presented work is being followed with a special focus on the usage of the M&S results as decision aid. Eventually, the simulation of decision models is under discussion as an open perspective.

### Review Questions

1. What do MDSEA, M&S, EA\*, DES, BPMN, and DEVS stand for?
2. How MDSEA architecture can support resource management and development?
3. Name five advantages of simulation for enterprise management.
4. Why modeling and simulation are bundled?
5. What is simulation-aided decision-making cycle? In this cycle, what is the benefit of hierarchical decomposition of decision and aggregation of simulation results?
6. Why BPMN 2.0 models should be transformed to DEVS models before simulation? How this transformation is performed?
7. What can be the perspective of simulation as decision aid for enterprise management?

**Acknowledgements** This work has been partially supported by the FP7 Research Project MSEE “Manufacturing Service Ecosystem” ID 284860. <http://www.msee-ip.eu/>

### References

- Al-Ahmari, A. M. A., & Ridgway, K. (1999). An integrated modeling method to support manufacturing systems analysis and design. *Computers in Industry*, 38(3), pp. 225–238. [https://doi.org/10.1016/S0166-3615\(98\)00094-3](https://doi.org/10.1016/S0166-3615(98)00094-3)
- Álvares-Ribeiro, N., Martins, ÁA., & Pinto Ferreira, J. J. (2004). Introducing Time Horizons to Enterprise Networking Architecture. In *Virtual Enterprises and Collaborative Networks* (pp. 63–70). Boston: Springer, Boston, MA. [https://doi.org/10.1007/1-4020-8139-1\\_7](https://doi.org/10.1007/1-4020-8139-1_7)
- Anglani A., Grieco A., Pacella M., Tolio T. (2002), Object-oriented modeling and simulation of flexible manufacturing systems: a rule-based procedure, *Simulation Modeling Practice and Theory*, Vol. 10, No. 3–4, pp. 209–234.
- ATL (2013), ATL/User Guide—The ATL Language <http://wiki.eclipse.org/ATL/> (accessed November 2013).
- Bazoun, H., Zacharewicz, G., Ducq, Y., Boye, H. (2013), Transformation of Extended Actigram Star to BPMN 2.0 and Simulation Model in the frame of Model Driven Service Engineering Architecture. TMS.
- Bazoun, H., Zacharewicz, G., Ducq, Y., Boye, H. (2014), SLMToolBox: An implementation of MDSEA for Servitisation and Enterprise Interoperability. In I-ESA, 7th international conference.
- Benkamouna, N., ElMaraghy, W., Huyet, A.-L. & Kouiss, K., (2014). Architecture Framework for Manufacturing System Design. s.l., s.n., pp. 88–93.
- Bouquet, P. et al. (2005), D2.2.1. Specification of a common framework for characterizing alignment, KWEB EU-IST-2004-507482 Project, v2.0., 2005.

- Bourey, J.P., Grangel Seguer, R. and Doumeings G., Berre, A.J. (2007) 'Report on Model Driven Interoperability', Deliverable DTG2.3, INTEROP Network of Excellence, pp. 91, <http://interop-vlab.eu/deliv/tg2-model-driven-interoperability/>, Avril.
- Boyé, H., Bazoun, H., Belkhelladi, K. (2014), SLMToolBox: A Tool Set For Service Engineering, Paper accepted in MODELSWARD 2014 2nd international conf on Model-Driven Engineering and Software Development.
- Bruzzo A.G., Mosca R., Revetria R., Rapallo S., (2000), Risk analysis in harbor environments using Simulation, *Safety Science*, Vol. 35, pp. 75–86.
- Burlton, R.T. (2001), *Business Process Management: Profiting From Process*. Sams publishing, Indianapolis.
- Carrie, A. S., & Macintosh, R. (1997). An assessment of GRAI Grids and their use in the Strathclyde Integration Method. *Production Planning & Control*, 8(2), pp. 106–113.
- Cardoso, J., Pedrinaci, C., Leidig, T., Rupino, P., De Leenheer P. (2012), Open semantic service net-works. The International Symposium on Service Science (ISSS).
- Çetinkaya, D., Verbraeck, A., Seck, M. D. (2012), Model Transformation from BPMN to DEVS in the MDD4MS Framework”, *TMS-DEVS*, pp. 304–309.
- Chen, D., Doumeings, G. & Vernadat, F., Architectures for enterprise integration and interoperability: *Computers in Industry*, p. (59), pp. 647–659. 2008.
- Chen, D. and Doumeings, G., (1996), The GRAI-GIM reference model, architecture and methodology, in “Architectures for Enterprise Integration”, Bernus P., Nemes L., J. Williams T. (Ed), *IFIP Advances in Information and Communication Technology 1996*.
- Chen K. Y., Lu S. S., (1997), A petri-net and entity-relationship diagram based object-oriented design method for manufacturing systems control”, *International Journal of Computer Integrated Manufacturing*, Vol. 10, Nos. 1–4, pp. 17–28.
- Czarniecki K. and Helsen S., (2003), Classification of Model Transformation Approaches, in *OOPSLA'03 Workshop on Generative Techniques in the Context of Model-Driven Architecture*, 2003.
- D'Ambrogio, A. and Zacharewicz, G. 2016. Resource-based modeling and simulation of business processes. In *Proceedings of the Summer Computer Simulation Conference (SCSC '16)*. Society for Computer Simulation International, San Diego, CA, USA, Article 63, 8 pages.
- Doumeings, G. (1984), *La méthode GRAI* [Ph.D. thesis], Bordeaux, France, University of Bordeaux 1.
- Doumeings, G., Vallespir, B., Chen, D. (1998), “Decisional modeling GRAI Grid” in *International handbook on information systems*, Bernus, P., Mertins, K. & Schmidt, G. ed., Berlin, Springer.
- Doumeings, G. & Ducq, Y., (2001). Enterprise Modeling techniques to improve efficiency of enterprises” in. *International Journal of Production Planning and Control*, 12(2), pp. 146–163.
- Ducq, Y., & Vallespir, B. (2005), Performance evaluation using decisional modeling. In *4th international conference on decision* (pp. 15–30).
- Ducq, Y. et al. (2014), “Generic Methodology for Service Engineering based on Service Modeling and Model Transformation State of the art in model driven approaches and model transformation” in “MSEE book: Manufacturing Service Ecosystem: Achievements of the European 7th Framework Programme”, FoF-ICT Project MSEE: Manufacturing Service Ecosystem (Grant No. 284860), Aachen: Verlag-Mainz: Bremer Schriften zur integrierten Produkt- und Prozessentwicklung, pp. 41–49.
- ElMaraghy, W., ElMaraghy, H., Tomiyama, T. & Monostori, L., (2012). Complexity in engineering design and manufacturing. *CIRP Annals— Manufacturing Technology*, Volume 61, p. 793–814.
- Garredu, S., Vittori, E., Santucci, J-F., Bisgambiglia, P-A. (2012), A Meta-Model for DEVS Designed following Model Driven Engineering specifications, *SIMULTECH*, page 152–157. SciTePress.
- Goepf-Thiebaud, V., & Kiefer, F. (2008). Multi-screen view and GRAI GridS to model decisional process of manufacturing IS alignment. *IFAC Proceedings Volumes (IFAC-Papers Online)*, 17 (1 PART 1).

- Hamri, M. and Zacharewicz, G. (2012), Automatic generation of object-oriented code from DEVS graphical specifications. In WSC'12. Article 409.
- Jaakkola, H. & Thalheim, B., (2011). Architecture-driven modeling methodologies. s.l., IOS Press, p. 98.
- Jahangirian M., Eldabi T., Naseer A., Stergioulas L. K, Young T., (2010), Simulation in manufacturing and business: A review, *European Journal of Operational Research*, Vol. 203, pp. 1–13.
- McNeill K. (2010), "How to extend the Eclipse Ecore metamodel." <http://www.ibm.com/developerworks/library/os-eclipse-emfmetamodel/index.html>
- Kettinger W. J., Teng J. T. C., Guha S.,(1997), Business process change: a study of methodologies, techniques, and tools, *MIS Quarterly*, 1997, Vol. 21, No. 1, pp. 55–80.
- Kurtev I., van den Berg K., (2005), MISTRAL: A language for model transformations in the MOF meta-modeling architecture, *Lecture Notes in Computer Science 3599*, pp. 139–158.
- Le Moigne, J. -L. (1977), *La théorie du système général. Théorie de la modélisation*, University Press of France, Paris.
- Lezoche, M., Panetto, H. & Aubry, A., (2011), Conceptualisation approach for cooperative information systems interoperability, in 13th International Conference on Enterprise, Information Systems, ICEIS 2011, pp. 101–110.
- Lu Y. (2012), Approach for Information Systems Semantic Interoperability in Supply Chain Environment, Ph.D. Thesis, Zhejiang University, Hangzhou, Zhejiang Province, China.
- Mesarovic, M. D., Masko, D. and Takahara, Y., (1970), *Theory of Hierarchical Multilevel Systems*, New York and London, Academic Press.
- Miller J. and Mukerji, J. (June, 2003). "MDA Guide Version 1.0.1".
- Mittal, S., and Risco Martin, J. L. (2012). *Netcentric System of Systems Engineering with DEVS Unified Process*. 610–613. CRC Press.
- MSEE book. (2014). "Manufacturing Service Ecosystem, Achievements of the European 7th Framework Programme FoF-ICT Project", Stefan Wiesner, Guglielmina C., Gusmeroli S., Doumeings G. (Ed), Aachen: Mainz: Bremer Schriften zur integrierten Produkt- und Prozessentwicklung.
- MSEE D15.2, (2012) deliverable "D15.2 D15.1 Methodologies and Tools for SLM components development M9 issue".
- Nethe A., Stahlmann H. D. (1999). "Survey of a general theory of process modeling", *Proceedings of the International Conference on Process Modeling*, Cottbus, Germany, pp. 2–16.
- Noran, O. (2012). Building a support framework for enterprise integration. *Computers in Industry*, 64. <https://doi.org/10.1016/j.compind.2012.09.006>
- OMG, "Business Process Model and Notation (BPMN) version 2.0" doc num: formal/2011-01-03.
- OMG, "MDA Guide Version 1.0." document number: omg/2003-05-01.
- Pirayesh-Neghab, A., Siadat A., Tavakkoli-Moghadam R., Jolai, F. (Sep. 2011). An integrated approach for risk-assessment analysis in a manufacturing process using FMEA and DES , ICQR 2011 IEEE International Conference.
- Pirayesh-Neghab, A., Etienne, A., Kleiner, M., Roucoules, L.(Sep. 2015). "Performance evaluation of collaboration in the design process: Using interoperability measurement", *Computers in Industry*. Vol. 72, pp. 14–26.
- Poler, R., Lario, F. C., & Doumeings, G. (2002). Dynamic modeling of Decision Systems (DMDS). *Computers in Industry*, 49, 175–193.
- Savage, Charles M. (1996). *Fifth Generation Management : Co-creating Through Virtual Enterprising, Dynamic Teaming, and Knowledge Networking* Butterworth-Heinemann, p. 184.
- Semini M., Fauske H., Strandhagen J. O. (2006). "Applications of discrete-event simulation to support manufacturing logistics decision-making: a survey", *Proceedings of the 38th conference on Winter simulation*, pp. 1946–1953,

- Simon, H. A. (1969). *The Science of the Artificial*. MIT Press, Cambridge, Mass, 1st ed. [3rd ed. in 1996, MIT Press].
- Smith J. S. (2003). "Survey on the Use of Simulation for Manufacturing System Design and Operation", *Journal of Manufacturing Systems*, Vol. 22, No. 2.
- Seguer, R. G., Bigand, M., & Bourey, J.-P. (2010). Transformation of Decisional Models into UML: Application to GRAI Grids. *International Journal of Computer Integrated Manufacturing*, (september), 655–672.
- Schmidt, D.C. (February 2006). "Model-Driven Engineering" (PDF). *IEEE Computer*. 39 (2).
- Shvaiko P.(2005). A survey of schema-based matching approaches, *Journal on Data Semantics IV* 146–171.
- Sienu A., Lamine E., Karduck A., Pingaud H. (2007) Conceptual Model of Risk: Towards a Risk Modeling Language. In: Weske M., Hacid MS., Godart C. (eds) *Web Information Systems Engineering—WISE 2007 Workshops*. WISE 2007. *Lecture Notes in Computer Science*, vol 4832. Springer, Berlin, Heidelberg.
- Taisch, M., Heydari, M., Carosi, A., & Zanetti, C. (2014). Service performance monitoring and control Toolset. *Procedia CIRP*, 16, 62–67.
- Thatte, S., Andrews, T., Curbera, F., Dholakia, H., Golland, Y., Klein, J., ... & Trickovic, I. (2003). Business process execution language for web services version 1.1. Microsoft and others, OASIS Standard BPELv11-May052003.
- Wainer, DEVS TOOLS, hosted by G. Wainer at Carlton University, November 2013, <http://www.sce.carleton.ca/faculty/wainer/standard/tools.htm>
- Weske, M., (2007). "Business Process Management: Concepts, Languages, Architectures". New York, Springer-Verlag: p. 368.
- Wiendahl, H. & Scholtissek, P. (1994). Management and Control of Complexity in Manufacturing. *CIRP Annals—Manufacturing Technology*, 43(2), p. 533–540.
- Wu Y-F. (2008). "Correlated sampling techniques used in Monte Carlo simulation for risk assessment", *International Journal of Pressure Vessels and Piping*, Vol. 85, pp. 662–669.
- Zacharewicz G.; Frydman C.; Giambiasi N. (2008). "G-DEVS/HLA Environment for Distributed Simulations of Workflows", *Simulation*, 84(5), pp. 197–213.
- Zacharewicz, G., Diallo, S., Ducq, Y., Agostinho, C., Jardim-Goncalves, R., Bazoun, H., ... & Doumeingts, G. (2016). Model-based approaches for interoperability of next generation enterprise information systems: state of the art and future challenges. *Information Systems and e-Business Management*, 1–28.
- Zeigler, B. P., Praehofer, H. and Kim, T. G. (2000). "Theory of modeling and Simulation", NY.

## Author Biographies

**Gregory Zacharewicz** is Associate Professor at University of Bordeaux (IUT MP) with both competences in enterprise engineering and computer sciences. His research interests include Discrete Event Modeling (e.g., DEVS), Distributed Simulation, HLA, and Workflow. He has been involved in FP7 European projects (e.g., MSEE and FITMAN). He is recently focused on Enterprise Modeling and Semantic Interoperability.

**Amir Pirayesh-Neghab** is a research engineer in INTEROP-VLab, Belgium (operational office: Talence, France). He obtained his Ph.D. in industrial and mechanical engineering from Arts et Métiers (ENSAM), ParisTech, France in 2014. His research interests include Interoperability Evaluation, Performance Measurement, Enterprise Modeling and Process Simulation. He has been involved in a national project, called NOSCIFEL, and several FP7 and H2020 European projects such as TELLM, FITMAN, PSYMBIOSYS, and C2NET.

**Marco Seregni** is currently a Ph.D. candidate at the Department of Management, Economics and Industrial Engineering (DIG) of Politecnico di Milano (POLIMI) where he has been graduated in 2013. He has been participating in two EU funded projects (CTC-Close to Customer, Psymbiosys) and an Italian regional project. His main research interests include requirements engineering in Product Service Systems, supply chain management for distributed manufacturing systems, and decisional modeling for reshoring strategy.

**Yves Ducq** is Full Professor at University of Bordeaux from September 2008. He is working on Performance Measurement, Enterprise Modeling and Production Management since fifteen years and has published more than 20 papers in books and international journals and more than 70 papers in international conferences. He is responsible of the research group dedicated to production engineering sciences (GRAI research Group) at IMS laboratory.

**Guy Doumeingts** is Emeritus Professor at the University of Bordeaux, where he was appointed Full Professor and directed the GRAI/LAP (Laboratory of Automation and Productics). He chaired the Technical Committee on Computer Application in Technology at IFIP, and was member of the Editorial Board of several journals. In 2012 he was elected General Manager of INTEROP-VLab (International Virtual Laboratory for Enterprise Interoperability), a network of scientists and organizations to develop and promote Enterprise interoperability.