A Ten-Step Design Method for Simulation Games in Logistics Management

Michele Fumarola

Jan-Paul van Staalduinen

Alexander Verbraeck

System Engineering Group,
Faculty of Technology,
Policy and Management,
Delft University of Technology,
Jaffalaan 5, 2628 BX Delft,
The Netherlands

Simulation games have often been found useful as a method of inquiry to gain insight in complex system behavior and as aids for design, engineering simulation and visualization, and education. Designing simulation games are the result of creative thinking and planning, but often not the result of a rigorously applied design method. Design methods can be used to structure the creative process. The specific types of games we chose for studying design methods are simulation games focused on information-intensive domains, of which logistics management is an example. Our new design method takes into account the information intensiveness of the domain. The design method incorporates enterprise information management, simulation model design, and instructional design. The design method we propose uses ten steps in designing a simulation game: the first five for making a conceptual design and the final five for using the conceptual design as a basis for the simulation game. Iterative cycles are added to improve intermediate results. This paper discusses the design method and presents two different case studies. The first case study helped in developing the design method, while the second case study served for assessment and improvement. [DOI: 10.1115/1.3617440]

Keywords: design method, game design, gaming simulation, engineering simulation and visualization, instructional design

Introduction

Computersupported management games have a long going history, starting in the early 1960s with the AMA top management decisions simulation [1]. From there on, many other management games for educational purposes have been developed. Computer supported management games are based on underlying models of a decision situation that the management game addresses. These decision situations often have time constraints in real life, requiring a stepwise, time-driven approach to the game. As time is a prominent aspect of these games, computer simulations can be the starting point for management games, making them simulation games. These simulation games have been found useful as a method of inquiry to gain insight in complex system behavior and as aids for training and education [2–5]. Simulation games, as defined by Galvao et al. [6], are a specific kind of simulation that can be used for educational and training purposes.

Logistics management helps organizations to plan, implement, and control the flow and storage of goods and related information between two points. Logistics management revolves around decision-making in complex ill-structured problems, where multiple actors are involved (as defined by the Council of Supply Chain Management Professionals). Logistics management is an information-intensive domain. It often requires a form of knowledge management: a way of structuring, sharing, and using the knowledge of its particular domain in a specific situation.

Simulation games in logistics management are a subset of simulation games, whose main purpose is gaining insight into the broad field of logistics. The creation of such a simulation game is the result of creative thinking and planning, but often not the result of a rigorously applied design method [7]. But design methods for specific types of games, in this case simulation games focused on information-intensive domains, can be used to structure this creative process.

In this paper, we present a ten-step design method for simulation games focused on information-intensive domains, domains in which organizations have to deal with vast quantities of relevant business information. This design method was based on the work by Van Houten [8] and by using the Knowledge Representation Requirements Model by Turnitsa and Tolk [9] as a theoretical framework to ground the design method in. In this paper, we also present one case study used for developing the design method and one case study used for testing the design method.

Background

Following Lane [10], we acknowledge two different definitions of both management simulations and games. We use these definitions to construct a theoretical basis for the design method, we present later on in this paper. Lane [10] distinguishes two definitions: the *intervention* and the *verisimilitude* definition.

The intervention definition originates from system dynamics and defines a (simulation) model as "simply a collection of information and relationships in rules, algebra, and logic made visible to an observer." According to the intervention definition, a game is "a collection of information and relationships in rules, algebra, and logic made visible to an observer and presenting data to and requiring information from an observer/participant." The keywords here are "requiring information." What distinguishes a game from a simulation is the fact that observers/participants ("players") actively provide information to a game during play. They provide input to the game, and the game responds, or reacts to this input.

To have a complete set of definitions, we also need a definition for simulation in general. This definition is provided by Shannon [11]: "Simulation is the process of designing a model of a real system and conducting experiments with this model for the purpose of either understanding the behavior of the system or of evaluating various strategies (within the limits imposed by a criterion or set of criteria) for the operation of the system."

The verisimilitude definition defines a simulation as a "specified sequence of verisimilitudinous activities designed to convey lessons to the participants on the properties of a real-world

Contributed by the CAD/Solid Modeling Committee for publication in the Journal of Computing and Information Science in Engineering. Manuscript received November 27, 2009; final manuscript received December 13, 2010; published online December 21, 2011. Assoc. Editor: Andras Kemeny.

situation" and defines a game as a "specified sequence of activities designed to convey benefits to the players." Here, the difference between both definitions is smaller. In a situation where the supposed benefit of a game is "learning," both definitions can even be considered the same, as the benefits in that case equate "lessons."

Again in Lane [10], the author argues the intervention definition is in total harmony with computer simulation modelling found in operations research. Simulations for logistics management can, therefore, be considered to fit in the intervention definition as well. The intervention definition has a strong focus on knowledge: a simulation makes knowledge visible, while a game makes knowledge intractable. From an educational perspective, simulation games are about the transfer knowledge.

The simulation games on which we focus are the ones defined by the intervention definition. These simulation games are widespread in literature and are often based on realistic data and simulation models with valid behavior.

Game Design. Game design is the process of coordinating the evolution of the design of a game [12]. A distinction has to be made between the game design itself (the artifact) and the game design process; the process through which the design is created. The game design process is guided and structured through the game design method.

Game design is not purely an art, because it is not primarily a means of aesthetic expression. Nor is game design an act of pure engineering. It is not bound by rigorous standards or formal methods. The goal of a game is to entertain through play, and designing a game requires both creativity and careful planning. [...] Designing game is a craft like cinematography or costume design [7].

Authors on game design only give an abstract overview of the game design process, stressing that it is at best an ill-structured process that depends on the experience and intuition of the game designer [7,12–16]. Adams and Rollings [7] discern three stages in the development process (see also Fig. 1) as follows:

1. The concept stage, which you perform first and whose results do not change.
2. The elaboration stage, in which you add most of the design details and refine your decisions through prototyping and testing.
3. The tuning stage, at which point no new features may be added, but you can make small adjustments to polish the game.

The game design, ideally, is created during the initial stages of the development process and is only slightly adjusted during later stages. A game design starts with a tight concept, which grows during the process. Bateman and Boon [12] emphasize the purpose of tight design: to use the minimum quantity of elements required in a game design to support the desired gameplay. They use the term elasticity to describe the freedom to discard components. *Contractile elasticity* means having the freedom to discard components, and *expansile elasticity* is the freedom to add new components. When both these properties exist in unison, game designers have total freedom to redesign the game during the development process.

Most experts on game design agree that there is not one single dominant design method for games [7,12,14,16]. Most current books on game design consist of high-level overviews of the design process combined with abstract representations of elements that should be included in a game. Furthermore, although academic interest in games has risen quickly over the past decade, the games industry has never shown a similar interest in academic work [17]. Therefore, scientific validation of formal game design methods remains limited at best.

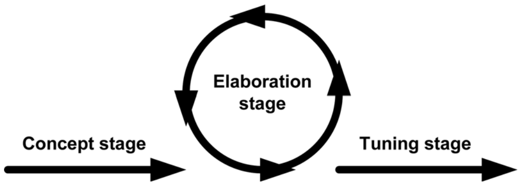

Fig. 1 Three stages of the design process [7]

Based on the work by Van Houten [8], and the broadness and abstractness of available game design methods, we perceived the need for more structure in designing simulation games focused on information-intensive domains. A more structured approach is required, because of the increased need for carefulness, while designing games that are information-intensive.

Exploratory Case Study: The Global Supply Chain Game

To create our design method for simulation games focused on information-intensive domains, we initially followed the research of Van Houten [8], where a distributed supply chain game for MBA education was developed and tested. This game was developed to provide a realistic setting in which students and managers could experience fundamental concepts and issues in supply chain management. They would be able to experiment with different strategies, in different supply chains, with different products, and different issues. The game's aim is to challenge players to integrate and apply their knowledge and skills related to supply chain management.

In his research, Van Houten used a nine-step game design method to create the so-called "global supply chain game," although in the research project the development of the underlying game engine and building and testing of support tools were the main focal point of the project. For the development of the game, Van Houten started (1) with the formulation of the context of the game, looking at the goal of the game and the requirements for training, all expressed with respect to the real-world system and problem. He followed this by defining practical boundaries for the game, i.e., by looking at the practical setting of the game and defining the mode of play.

After that, he (2) conceptualized the real-world system, e.g., using objects and relations. In the conceptualization, it is important to set the system boundary for the game, and make choices what to include and what not.

In Van Houten's next step, he (3) selected the components to be included in the business game. By including a consistent set of components, a specific case with a specific teaching purpose can be modeled.

Specification of the business game (4) was the subsequent step for the global supply chain game. Actors, relationships and scripts are worked out in more detail in this step. The conceptual components of the game are enriched with realistic data. For Van Houten's game, data came from real systems, company documents, papers, experts, and websites. Examples were realistic data about product prices, cost of transportation per kilogram and kilometer, with discounts for larger orders and larger distances, and data about realistic locations for factories, warehouses, and markets. After defining what types of data he needed, the raw data were retrieved for the game.

Van Houten [8] focused on construction of the engine and support services, considering the game design as a software engineering project using the spiral cycle of Boehm [18]. The data of the game were (5) specified and (6) structured using a database. Additional, static data are (7) added using a content management system[1] and a player manual, which can be used to prepare for the game and during game play. The game uses (8) the D-SOL simulation engine [19] to model the dynamic aspects, the scenario, and the events for the players, as well as the state changes of the

[1] http://www.w3.org/1999/xlink">www.gscg.org

internal components as a reaction on the player's decisions. The final step of Van Houten [8] was the (9) verification, validation, testing, and preparation of the game for real play.

Van Houten uses the following nine design steps:

1. formulating of the context of the game
2. conceptualizing the real-world system
3. selecting the components to be included in the game
4. specifying the game structure
5. specifying the game data
6. structuring the game data
7. adding a data management system
8. implementing a simulation engine
9. verifying, validating, and testing the game

Our ten-step design method was based on the work by Van Houten, and by grounding the design method in the Knowledge Representation Requirements Model (KRRM) by Turnitsa and Tolk [9], as introduced in the section Toward a Ten-Step Design Method for Simulation Games.

Toward a Ten-Step Design Method for Simulation Games

In our research, we use the "KRRM" for representing knowledge. Turnitsa and Tolk [9] introduced the KRRM, which presents an increased layering of requirements from data to information, knowledge, awareness, and finally understanding. This model has been presented to achieve interoperability between simulation models with different worldviews. We will, however, show how this model is also useful in the design of simulation games.

The KRRM starts from raw data as the lowest level of knowledge. As soon as semantics are attached to raw data, information is created. These semantics tell us what the data actually are. As soon as information is put into context, knowledge is achieved. Turnitsa and Tolk [9] explain this as using ontological entities, which are objects "that can be anything in the system that can be addressed." Adding dynamics to knowledge gives us awareness followed by understanding once we can anticipate intentionality, which helps us reach understanding as we try to anticipate the next state of the system. Dynamics is mainly the addition of changes over time.

The intervention definition of simulation and game aligns well with the KRRM. A simulation can be a computer model that steps through time, resulting in the awareness-layer of KRRM. A game, on the other hand, requires information from the participant, possibly generated by the anticipation of intentionality of the system. If the game is effective, the player should, in time, be able to understand the system. The KRRM is graphically presented in Fig. 2.

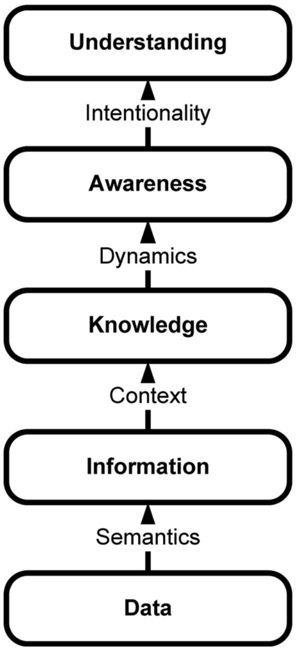

Fig. 2 The KRRM: the rectangles contain the different levels of knowledge, while the arrows represent the requirements [9]

A comparison between the nine steps used by Van Houten [8] and the KRRM lead us to a "top-down, bottom-up" approach for our design method. The design starts out with the conceptual design and works from a broad overview of the decision situation in the game to a low-level design of the usage of raw data. From there the game is developed in layered steps, starting with the lowest level (data carrier) to the game itself; the outer shell or user interface. We next describe our proposed design method.

A Ten-Step Design Method for Simulation Games. The design method we propose uses ten steps in designing a simulation game; five for the conceptual design and five for game development, as seen in Fig. 3. The first five steps for the conceptual design are as follows:

1. *Decision situation*. Simulation games in logistics management revolve around players making decisions about the different situations that occur, while managing logistics. Thus, the first step is to design the general type of decision situation for logistics management that will form the basis for the entire simulation game. A well-described decision situation provides both context and purpose to the simulation game design, and ultimately the simulation game. In this step, the designers can also develop first ideas about the game to be developed: what is the goal of the game, what should the duration of the game be, and how many users will play the game at the same time? Will they play individually or in teams?
2. *Underlying model*. Once the decision situation that forms the basis of the simulation game is designed, the underlying model can be constructed. The underlying model describes the basic rules, dynamics and behavior of the logistics system that will be managed in the game. In many cases, choices will have to be made what rules and relationships will have to be included in the game that has to be constructed.
3. *Concrete case*. To make the simulation game more tangible and recognizable for players, the game should be anchored in a concrete case that reflects a possible situation within the domain of logistics management. Designing the concrete case means selecting the type of industry, scope of the decision situation, and the actors and players involved in the game. This is a specification of the decision situation; multiple concrete cases can be derived from a decision situation, which could lead to different games. For the game to transfer the main teaching points of the first step, the selection of the concrete case is critical. A case might have to be simplified or abstracted to convey the teaching goal in a better way.
4. *Structured data*. To handle large amounts of raw data and to provide overview, a data structure needs to be developed. The aim of the data structure should be to categorize and sort raw data into manageable chunks of information. This phase can be started by searching the enterprise resource planning, knowledge management system, and supply chain management systems of the organization whose decision situation forms the basis of the simulation game. Structured organizational data are collected, which are then transformed into raw data for the simulation game.

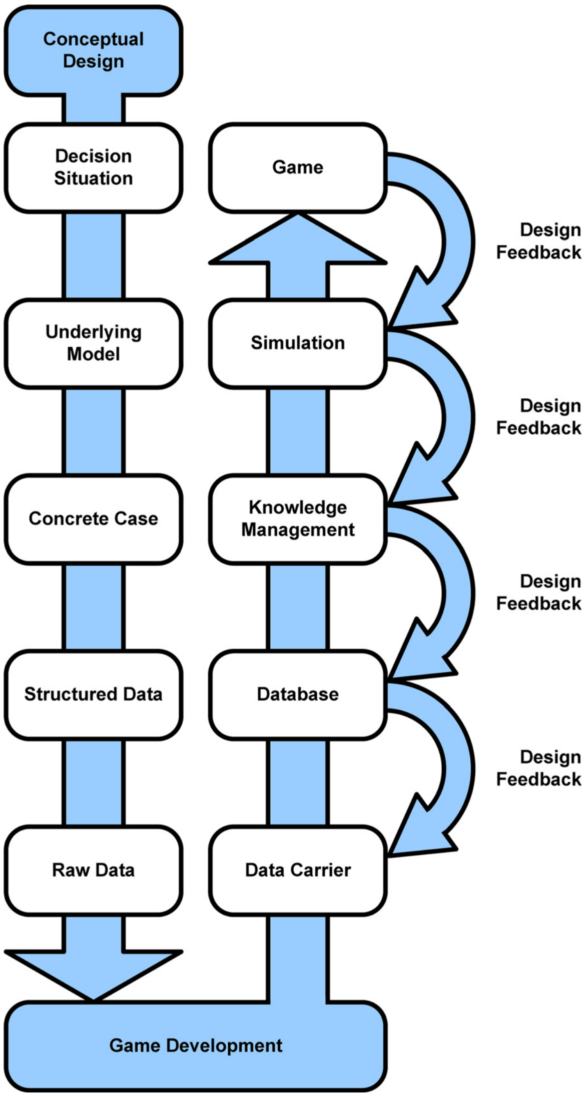

Fig. 3 The design method

Sometimes the dataset has to be reduced for the gaming purpose; sometimes the dataset has to be enriched with exceptions to make the game play satisfy the teaching needs. All in all, a valid and realistic data set has to be defined that supports the teaching goal of the game.

5. *Raw data*. After the concrete case has been written, sources of raw data need to be defined and acquired. This raw data provides the variables in the game (and thus the dynamism), ultimately powering the simulation that drives the game. When working with real-life concrete cases, actual organizational data from the field of logistics management can fill this need.

   With the development of the data structure, the final step in the conceptual design phase has been taken, and the simulation game design can then be developed into a real game. This necessitates the following steps:
6. *Data carrier*. In order to present data in the simulation game, a data carrier needs to be developed. This data carrier will often be computer-based, or supported, although this is not a requirement. In simpler games, file cards could be used, although in practice information-intensive decision situations tend to veer toward computer-based games.
7. *Database*. The database stores, structures, and manages the data that are used within the simulation game. This usually involves designing the data model; e.g., the entity relationship model.
8. *Knowledge management*. In this step, designers enable game users to have ready access to data from the simulation game, and to have it presented in context, knowledge management is necessary. With knowledge management relevant information, such as documents, pictures, films, etcetera, can be connected to objects in the game.
9. *Simulation*. The next step is to develop the simulation model that is capable of presenting the behavior of the logistics systems, based on the data available. The simulation model gives insight into the behavior of the real-life or hypothetical logistics system and forms the basis for the simulation game.
10. *Game*. The final step of the development phase is to create the simulation game.

In current work, where new game instances are being developed using the global supply chain game engine, a data driven approach is taken to develop these new business games and steps taken by the developers of new games appear to match with the ten-step approach outlined in this paper.

Case Study: Designing Automated Container Terminals The design phase is crucial in the development of an automated container terminal. If not taken correctly, the decisions made during the design phase can have serious repercussions for the future productivity of the terminal under development. The main decisions involve various kinds of equipment used on the terminal, the layout of the stacks, etcetera. Although these decisions are often based on results of sound simulation models, this simulation based design process is often unstructured, which could lead to a less than optimal result. To tackle this problem, a game can be designed that gives players the opportunity to explore the solution space of different alternatives for a certain problem. This concept has been presented as the “multiple worlds” concept [20].

The design process presented in this paper is perfectly applicable to this game as it resides in the same domain. The game is based on a simulation, which represents a valid model of the system at hand. There is also a large amount of information required in addition to the results of the simulation. Decision makers need to know how the container terminal is going to look like, based on their design decisions. They need the specifications of the equipment to choose from the various possibilities and they require other “soft” data, which does not result from a simulation (e.g., labor regulations and environmental restrictions).

The conceptual phase of this project focused on collecting information about the decision situation. The project was done in cooperation with a large container terminal operator actively involved in the design of new container terminals. Based on existing literature on container terminal simulation, a conceptual model has been developed. Following the design method, cases have been selected so that the step toward an actual implementation can be taken by collecting additional information.

The first five phases of the design method, focusing on the conceptual design resulted in the following artifacts:

1. A description of a simulation game that helps decision makers gain insight into the design process and product of automated container terminals.
2. A conceptual model consisting of (1) an actor analysis on the actors involved in a container terminal design process, (2) process maps on the operations carried out on automated container terminals, and (3) the decisions that need to be made during the design process.
3. Scenarios based on current projects that are currently being considered for development by the terminal operating company.

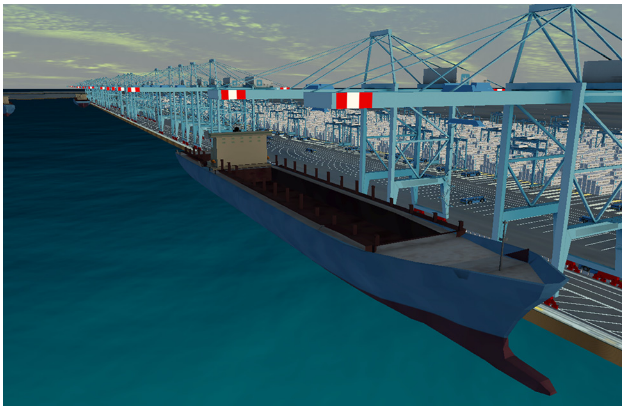

Fig. 4 The 3D virtual environment to support the design process

4. An ontology for container terminals has been constructed. This ontology provides the necessary means to structure the design product.
5. Raw data collected in the form of historical data (i.e., productivity of existing container terminals) and equipment specifications.

The development phase consists of developing different software components (this is the data carrier identified in phase 6): a knowledge visualization tool, and a simulation model for automated container terminals. The knowledge visualization tool presents the user's possibility to *visualize* and *structure* new container terminals. The visualization is based on available CAD drawings provided by terminal designers, from which required information is extracted to XML files (the database from phase 7) to construct the 3D environment. Structuring is done by attaching information to objects in the virtual container terminal. The type of information that can be added is varied: documents, pictures, movies, etc. This enables decision makers to quickly have access to the information specific to the context they are focusing on: the specific object in the specific container terminal. The knowledge visualization tool fulfills the needs identified in phase 8. The simulation model (phase 9) is developed as a library of components, which allows a flexible way of building models for different container terminal designs. This is achieved by using Discrete Event System Specification [21] as the modeling formalism and system entity structure, which is a specification of structural and specialization relations for a model family.

The tool and model are then combined to have the dynamics and results of the simulation models visualized in the knowledge visualization tool. Hereafter, the environment can be embedded in a gaming context. In the game, different teams of designers have to compete against each other to achieve the best performing container terminal in terms of TEU/year (TEU stands for twenty foot equivalent, which is the size of a small container). This matches the last phase of the proposed design method.

Although extensive evaluations are still needed for the overall gaming simulation, preliminary evaluations have been carried out with the visualization component. This evaluation, which is extensively discussed in Fumarola and Versteegt [22], has been performed to assess the 3D virtual environment to design new container terminals, which is shown in Fig. 4. The evaluation has been carried out both with students and professionals. Standardized usability tests, the IsoMetrics questionnaire based on ISO 9241 part 10 usability standards [23], have been used to assess the usability. Qualitative data have been collected by conducting semistructured interviews with container terminal experts. This data have been used to assess the contents of the design environment. The evaluation produced positive results and provided insights to improve the environment. One of the major insights was gained from the interviews with experts who discussed the design process. It was concluded that the environment could be used best during the conceptual phase, where different actors could come together to define the prominent features of the design. The environment would help the different actors assess different alternatives and make informed decisions.

Reflections and Conclusions

Designing simulation games is often a process of creative thinking and not the result of a rigorous design method. Design methods bring structure to the creative process in order to achieve the desired result. Existing design methods for simulation games are often too broad or vague. By focusing on specific simulation games, we introduced a design method that offers a more strict method that can be followed to design simulation games for logistics management.

Starting from an existing knowledge representation model, we developed a design method that goes through various products in the development cycle. The design method has been studied based on two case studies. The first case study helped us to develop the design method. Herein, we recognized nine distinct phases and deduced a tenth phase. After mapping these phases onto an existing model to represent knowledge, we developed a ten-step design method. Though a second case study, we assessed and improved this design method.

The design method mainly focuses on computer based logistics management games. We expect, however, that the design method could be applied in other information-intensive domains as well as to noncomputer supported simulation games. However, to assess this, specific case studies should be performed.

References

[1] Cohen, K. J., and Rhenman, E., 1961, "The Role of Management Games in Education and Research," Manage. Sci., **7**(2), pp. 131–166.

[2] Mayer, I., and Veeneman, W., *Games in a World of Infrastructures: Simulation-Games for Research, Learning and Intervention* (Eburon, Delft, 2002).

[3] Bekebrede, G., and Mayer, I., 2006, "Build Your Seaport in a Game and Learn about Complex Systems," J. Des. Res., **5**(2), pp. 273–298.

[4] Mayer, I., van Bueren, E., Bots, P., van der Voort, H., and Seijdel, R., 2006, "Collaborative Decision-Making for Sustainable Urban Renewal Projects: A Simulation-Gaming Approach," Environ. Plann. B, **31**, pp. 403–423.

[5] Bots, P. W. G., and Daalen, C. E. v., 2007, "Functional Design of Games to Support NRM Policy Development," Simul. Gaming, **38**(4), pp. 512–532.

[6] Galvao, J. R., Martins, P. G., Gomes, M. R., 2000, "Modelling Reality With Simulation Games for a Cooperative Learning," *2000 Winter Simulation Conference* J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, eds., IEEE, New Jersey.

[7] Adams, E., and Rollings, A., *Game Design and Development: Fundamentals of Game Design* (Pearson Education, Inc., Upper Saddle River, NJ, 2007).

[8] Van Houten, S. P. A., 2007, *A Suite for Developing and Using Business Games: Supporting Supply Chain Business Games in a Distributed, Context*, Ph. D. dissertation, Delft University of Technology.

[9] Turnitsa, C., and Tolk, A., 2008, "Knowledge Representation and the Dimensions of a Multi-Model Relationship," *2008 Winter Simulation Conference*, S. J. Mason, R. R. Hill, L. Moench *et al.* eds., IEEE, New Jersey.

[10] Lane, D. C., 1995, "On a Resurgence of Management Simulations and Games," J. Oper. Res. Soc., **46**(5), pp. 604–625.

[11] Shannon, R. E., *Systems Simulation: The Art and Science* (Englewood Cliffs, Prentice-Hall, NJ, 1975).

[12] Bateman, C., and Boon, R., *21st Century Game Design* (Charles River Media, Inc., Hingham, MA, 2006).

[13] Crawford, C., *The Art of Computer Game Design* (Osborne/McGraw-Hill, Berkely, CA, 1984).

[14] Schuytema, P., *Game Design: A Practical Approach* (Charles River Media, Boston, MA, 2007).

[15] Simons, I., *Inside Game Design* (Laurence King, London, 2007).

[16] Schell, J., *The Art of Game Design: A Book of Lenses* (Morgan Kaufmann, Burlington, MA, 2008).

[17] Hopson, J., 2006, "We're Not Listening: An Open Letter to Academic Game Researchers," Retrieved December **8**, 2008, from http://gamasutra.com/features/20061110/hopson_01.shtml

[18] Boehm, B., 1988, "A Spiral Model of Software Development and Enhancement," IEEE Comput., **21**(5), pp. 61–72.

[19] Jacobs, P., *The DSOL Simulation Suite: Enabling Multi-Formalism Simulation in a Distributed Context* (Delft University of Technology, Delft, The Netherlands, 2005).

[20] Fumarola, M., and Verbraeck, A., 2008, "Multiple Worlds in Simulation Games for Spatial Decision Making: Concept and Architecture," *2008 Winter Simulation Conference*, S. J. Mason, R. R. Hill, L. Moench, and O. Rose, eds., IEEE, New Jersey.

[21] Zeigler, B. P., Praehofer, H., and Kim, T. G., *Theory of Modeling and Simulation* (Academic, Wiley Interscience, New York, 2000).

[22] Fumarola, M., and Versteegt, C.,2011, "Supporting Automated Container Terminal Design Processes With 3D Virtual Environments," *Handbook of Research on Practices and Outcomes in Virtual Worlds and Environment*, H. H. Yang and S. Yuen, eds., IGI Global, Hershey, PA.

[23] Gediga, G., Hamborg, K.-C., and Düntsch, I., *et al*. 1999, "The IsoMetrics Usability Inventory: An Operationalisation of ISO 9241-10," Behav. Inf. Technol., **18**(3), pp. 151–164.