

SCENARIO-BASED PLANNING AND MANAGEMENT OF SIMULATION GAME: A REVIEW

Jana Bikovska and Galina Merkurjeva
Riga Technical University
1 Kalku Street
Riga, Latvia
LV-1658
E-mail: jana@itl.rtu.lv, gm@itl.rtu.lv

KEYWORDS

Simulation game, scenario planning, scenario management.

ABSTRACT

This paper is devoted to scenario development methodology for planning and management of business simulation games. Nowadays the high efficiency of simulation games as a training method is obvious. Requirements to this kind of training are increasing. Modern computer technologies allow developing complex simulation games that are close to real life conditions. Generation of game scenarios is not a trivial problem; besides, management of such complex games requires some formal approach.

INTRODUCTION

Due to their nature, business simulation games can be considered as one of the most promising learning environments used in academic education as well as in personnel training in business companies. In this paper we are focusing on computer-based simulation games that are continuous and adopt a client-server architecture, in which users in remote locations (clients) interact in a common virtual environment maintained by a central game server. Such games can be considered as dynamic simulation games. They usually model complex systems, so it is not easy to model and manage them. As examples of such games can be mentioned the International Logistics Management Game introduced in (Bikovska et al., 2006) and The Distributor Game introduced in (van Houten et al., 2005). In general, a computer business simulation game may be interpreted as a sequential decision-making experience with reality, which is simulated and animated on the computer. Trainees can see the impact their decisions have upon the problem situation and future events, and can react to these effects and make new decisions (Merkurjeva G., 2000).

As human players are involved in the game by performing appropriate activities, it is important to develop and monitor the game environment in order to reach learning goal. Scenario usually creates the right state of the model at the right moment under the right conditions. A game manager decides about the

complexity level of designed scenario dependent on the specific goals of the course and background of the participants' knowledge.

Usually, the process of scenario development and subsequent control over it during the game session is carried out manually (Merkurjeva et al., 2004). But if the number of players is variable and rather large (for example 10-12 players), this leads to an increase in the game complexity and it becomes very difficult to manage it at a certain moment of time and to predict the effect of actions that are executed. Player can take unforeseen action which would possibly lead him to avoid fulfilling the goals of the exercise. And if the game is played for a long period (more than one day), management of it becomes infeasible. This is also stated in (Magerko and Laird, 2002; van Houten and Verbraeck, 2006). In the paper, existing methods for scenario development of complex systems are reviewed and scenario-based framework for simulation game planning and management is proposed.

SCENARIO DEFINITION

There are different views on the definition of scenario, depending on the context in which the term is being used. This term appears in a wide variety of fields, ranging from state administration, information systems, requirement engineering to human computer interaction. Interpretation of scenarios seems to depend on their usage and how they are generated. In general, a scenario can be defined as a set of possible sequences of future events and it could be interpreted by a structure (Luger, 2005):

$\langle Ec, Re, Pr, Ro, Sc \rangle$, where

- *Ec* presents *entry conditions* or descriptor of the world that must be true for the scenario to be called;
- *Re* defines *results* or facts that are true once the scenario has terminated;
- *Pr*, or *props*, reflects the "things" that support the content of the scenario;
- *Ro* introduce *roles* or the actions that the individual participants perform; and
- *scenes* *Sc* divide scenario into particular time periods.

According to (Kononov et al. 1999), a scenario usually describes the behaviour of the system, the process of

changing its parameters and identifying the conditions of the system behaviour, as well as depicting how the system's components interact with each other. As business simulation games are considered in this paper, it is useful to provide a business scenario definition, i.e. to give a complete description of a business problem, both in business and in architectural terms, which enables individual requirements to be viewed in relation to one another in the context of the overall problem (TOGAF, 2006). The synthesised scenario allows reflecting adequately the process of system behaviour, developing its organisational strategy and implementing reactions to changes in a real situation, generating strategic plans of action, providing qualitative analysis of consequences of actions, and also predicting loss, possible damage and undertaken risk.

This paper is focusing on business simulation games and as stated in (Dobson et al., 2004; and Van Houten and Verbraeck, 2006), in simulation games, scenarios usually describe the context of the game, the desired development of the game over time, and events that take place during game play to enhance the learning.

A business scenario in (TOGAF, 2006) describes: (1) a business process, application, or set of applications that can be enabled by the architecture; (2) the business technology environment; (3) the people and computing components (called "actors") who execute the scenario; (4) the desired outcome of proper execution.

From the above definitions it is clear that there are two types of scenarios: scenario of system behaviour and system management scenario. Such a difference exists because the management scenario is developed according to management objectives as the purpose of system behaviour scenario is to master its functioning without any influence from outside. In most investigated literature about simulation games two scenario types are mentioned; however, more attention is paid to system management scenario.

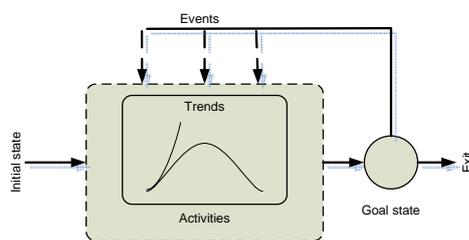


Figure 1: Generic Scenario Structure

The generic scenario structure is shown above (see Fig. 1); it indicates the main scenario elements: initial state of the system, system development trends, activities and events. According to it, a scenario development consists of:

- defining the main objective of a scenario;
- setting the parameters that define initial state of the game;

- defining trends of the desired development of the game over time;
- executing certain events if the desired state of the simulation game does not occur to achieve the desired state of the model.

On the one hand, activities can impact on the dynamics of the trends and cause or refuse other activities, but on the other hand events affect the trends and activities performed. Usually a game manager decides about the trends before the game run. During the game players make their decisions, i.e. perform actions, according to learning objectives and if it leads to unexpected results (for example to a bankruptcy), game manager can include some events in the game to motivate the players to achieve the goal. Figure 2 indicates the place of scenario in the game; it can be considered as scenario-based framework for simulation game planning and management.

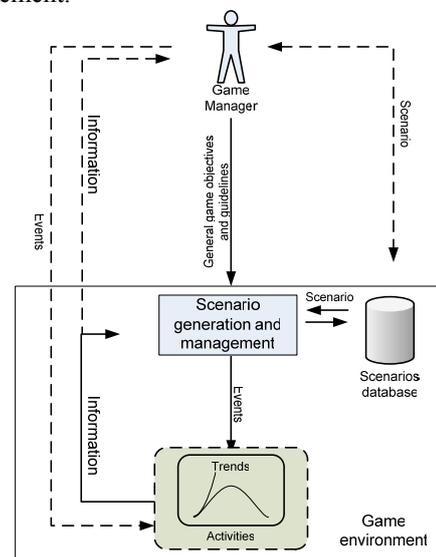


Figure 2: Proposed Scenario-based Framework for Simulation Game Planning and Management

The overall aims of scenario development can be defined (Kulba et al., 2004) as follows:

- to identify the key issues of investigated object development and generate on this basis qualitatively different variations of dynamics of its development;
- to analyse and estimate each of the variants obtained, study its structural features and possible consequences of its realization with the purpose of constructing certain plans for their realization.

For simulation games the second aim is primary as we need to see the consequences of possible participants' decisions and to manage them if the learning objective is not achieved.

Along with the *purpose view*, scenarios are classified according to the role they aim to play (Rolland et al., 1996):

- descriptive – used to explain process and understand its operations ;
- exploratory – used to explore and evaluate several different possible solutions for satisfying a system goal (our research is focused on scenarios of that kind);
- explanatory – used to provide detailed illustrations of particular situations and their rationale.

The following requirements could be recommended for business scenario description (TOGAF, 2006) which should be:

- specific, by defining what needs to be done in the business;
- measurable, through clear metrics for success;
- actionable, by a) clearly segmenting the problem, and b) providing the basis for determining elements and plans for the solution;
- realistic, in that the problem can be solved within the bounds of physical reality, time and cost constraints;
- time-bound, in that there is a clear statement of when the solution opportunity expires.

METHODS FOR SCENARIO FORMALISATION AND GENERATION

There are different scenario development methods (Kulba et al., 2004). Since such factor as uncertainty is present in scenarios, which can significantly affect their complexity, not all scenarios can be easily formalized; they could be classified as follows:

- formalized that include the methods of scenario generation based on automatic or computer-aided procedures;
- partly formalized, i.e. based on an automated procedure but adjusted by experts;
- non-formalized, i.e. based on expert opinion.

In our framework partly formalized scenario development methods will be used. The developer of the game can play the role of an expert.

As stated in (Kulba et al., 2004), methods of systems analysis and structural analysis can be used for scenario development as they allow decomposing the system, defining its *units*, *processes* and *structures*. Once these three components can be identified, a model of an object/system can be produced. This model is then run through a simulation that allows analyzing system behaviour, defining existing shortcomings in management, as well as clarifying new problems. Since any simulation game is a model of some system, we have to single out only those units, processes, and structures that are important for particular learning goals. As any scenario has a goal, the so-called goal approach can be used in considering methodology (Kulba et al., 2004). This approach supposes:

- defining a system of goals;

- developing a set of solvable tasks to achieve the goals;
- measuring certain results at the goals achievement stages.

A goal is the desired result of system functioning in a certain time period. Usually the goal is achieved by accomplishing certain tasks that have quantitative characteristics. A scenario goal has to be formalized; it can be qualitative or quantitative. The qualitative goal can be either achieved or not, however, the achievement of the quantitative goal can be easily measured by certain parameters.

Several scenario representation schemes can be distinguished: frames and semantic networks, (Luger, 2005), signed graphs (Kulba et al., 2004), and trees (Kindley, 2002).

Scenarios can be represented by *frames* as they are used for knowledge representation. Frame is a more advanced knowledge representation form than semantic networks as it supports object-oriented concept. A frame may be viewed as a static data structure used to represent well-defined stereotyped situations.

According to (Kulba et al., 2004), the most frequently used methods for describing system goals are signed graphs or trees. When such a tree is being constructed, heuristics methods and experts' estimates can be used. A *signed graph* is a graph in which each edge has a positive or negative sign. Formally, a signed graph is defined by a pair (G, σ) that consists of a graph $G = (V, E)$ and a sign mapping or signature σ from E to the sign group $\{+, -\}$.

A *tree* is a way of representing the hierarchical nature of a structure. In graph theory, a tree is defined as connected acyclic graph. As there is infinite number of system states in simulation game, it would be ineffective to use state space search tree (Nilson, 1980). We would propose to use a goal/sub-goal tree.

As a non-quantified scenario development method morphological analysis (Ericsson and Ritchey, 2002) can be mentioned. It is widely used for structuring and analysing technical organisational and social problem complexes. It is a method for exploring all possible solutions in a complex problem space.

As stated in (van Houten and Verbraeck, 2006), for a simulation game management scenario production systems and production rules can be effectively used. Along with other methods artificial intelligence can be applied as well (Dobson et al., 2004; Bikovska et al., 2006). In the gaming, agents can replace human players or work in the background as individual assistant, monitor trainee and evaluate his/her action online. As mentioned in (Dobson et al., 2004) consumer, whose behaviour in the traditional business game is either pre-

specified by the scenario or sometimes implemented by just over-simplified algorithm, can be modelled by using the intelligent agent technology. Agent can perform some tasks of a game manager concerning scenario development: defining initial conditions of the game according to composed story or developed case study, monitoring the game session, and making necessary changes in scenario. From the experience it is known that a human factor can significantly impact upon inconsistencies of designed scenario, i.e. a high probability of mistakes, thus intelligent agents would significantly increase the game scenario quality and software reliability. Production rules embedded in the game software are used to set up parameters of the economical situation according to defined policies (see Fig. 3).

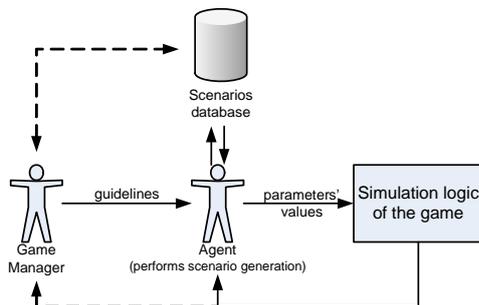


Figure 3: Agent Performs a Scenario Generation

Some other scenario representation schemes are used by different authors. For example, in (Luger, 2005) frame-like structures are used (see Fig. 4) and it refers to general scenario definition given above; in (Rolland, 1996) Use Case diagrams, but in (Uchitel et al., 2003) sequence diagrams are used.

<i>Scenario:</i>	Bicycle: production and trading	<i>Scene 1:</i> C make market analysis C make strategic decisions	} for each Market
<i>Properties:</i>	Product Markets Raw Materials Reports Overviews	<i>Scene 2:</i> C make tactical decisions C make operational decisions	
<i>Roles:</i>	Customers C=Companies Vendors	<i>Scene 3:</i> C analyse Reports C analyse Overviews	
<i>Initial Conditions:</i>	C have money C have assets Customers ask for products	IF results are not satisfactory THEN C adjust the business strategy AND C improve their tactics AND GOTO Scene 2	
<i>Results:</i> successfully	C operate	<i>Scene 4:</i> IF time is not over THEN C make decisions AND GOTO Scene 3	
satisfied	C go bankrupt Customers are not satisfied	<i>Scene 5:</i> End of the game	

Figure 4: Frame-like Structure for Scenario Representation

In business simulation games (see Fig.4) initial or entry conditions of the scenario may be interpreted as the assets and liabilities of the company to start with: like a

set of operational plants and their capacity, stocks of raw material and finished products, bank loans, etc. Usually, business simulations deal with production of goods or services and we assume to have some corresponding infrastructure that could be interpreted as props. Game participants perform different roles in the scenario such as managers of different departments at the company. At the end of the game, participants have to achieve a certain level of developments, or results. Dynamic games are divided into several periods.

As an example of the scenario planning and management a tree development example was created. The game situation can be described as follows: at a certain moment of the game customers demand has increased, the overall goal in this situation can be defined as satisfaction of customer demand with maximal profit for the company. As participants know the game rules, they can act (see Fig. 5).

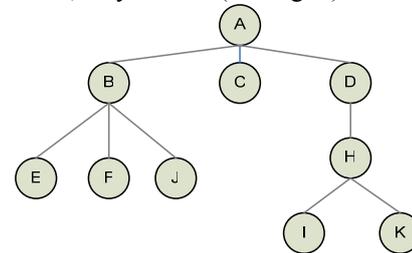


Figure 5: The scenario tree example

In the tree, the nodes mean the following:

- A – demand has increased and players has to satisfy it;
- B –invest money in machine capacity;
- C – work in overtime;
- D – work in two shifts;
- E – invest money in machine capacity and work in one shift;
- F – invest money in machine capacity and work in overtime;
- J – invest money in machine capacity and work in two shifts;
- H – work in two shifts;
- I – work in three shifts;
- K – continue to work in two shifts as in this case overtime is not allowed.

As can be seen from Figure 5, the goal can be achieved in different ways, but each company can calculate the most profitable one, i.e. choose the best strategy.

PROCEDURE OF SCENARIO DEVELOPMENT

In general, the process of a scenario development in a business simulation encompasses the following steps: (1) identifying the problem domain; (2) collecting or generating the data about chosen domain; (3) analysing the collected data in order to structure it; (4) building a “script” before the game starts, i.e., defining a specific set directions or instructions to be followed by users as well as defining a set of environmental parameters and an initial company’s state; (5) monitoring the run and

making necessary alterations in scenario during the game session.

In case of business scenario (TOGAF, 2006) development the above process also includes (see Fig. 6):

- identifying the human actors (participants) and their place in the business model;
- identifying computer actors (computing elements) and their place in the business model
- identifying roles, responsibilities and measures of success per actor.

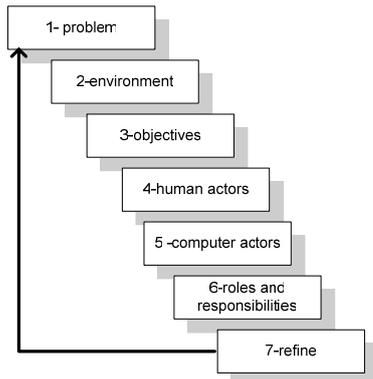


Figure 6. Steps to Create a Business Scenario

Specific procedure of scenario development is dependent on the structure that is used to formalize a scenario. For example, scenario development procedure by using computerized morphological analysis is proposed in (Ericsson and Ritchey, 2002) and a natural language-based scenario automatic generation procedure is given in (Prado Leite, 2000).

Since business simulation games can be considered as a model of complex social and economic system, scenarios of system behaviour could be used for their planning that are described by the following formal constructs (Kononov, 2001): identical system model, environmental model, the model of system behaviour, system state-measurement model, a set of rules for choosing the profile of object variation and a metaset of rules in order to describe system behaviour based on the above formal constructs. In this case the procedure of scenario development can be divided into several steps:

Step 1: Identify the initial state of the object or system $\mathbf{M}=(\mathbf{M}_s, \mathbf{M}_e, \mathbf{M}_b, t)$, where \mathbf{M}_s – internal system model, \mathbf{M}_e – environment model, \mathbf{M}_b – model of a system behaviour, t – time component.

Step 2: Define the system goals \mathbf{G} .

Step 3: Define the time step.

Step 4: Estimate the future states $s_i(t)$ of the system.

Step 5: Define the rules of choosing the future state $s(t_k)$.

Step 6: Choose the next state, i.e. updates to the system with regard to the current state of the models \mathbf{M} .

Step 7: Refine the initial state \mathbf{M} of the object or system.

Step 8: Evaluate the goal achievement degree.

Step 9: If the goal is achieved, terminate the procedure; otherwise go back to Step 7.

In terms of simulation games \mathbf{M}_s can be interpreted as internal game model with defined development trends \mathbf{M}_b (see Fig. 1), \mathbf{M}_e defines the game managers by choosing appropriate management events. System goals \mathbf{G} are defined by learning objectives but system states $s_i(t)$ depend on activities performed by the game participants. According to overall goal of the game and the rules of choosing the future state $s(t_k)$, participants make their decisions and evaluate their performance.

APPLICATIONS

In literature, various applications of different scenario formalisation and generation procedures could be found. In particular, in (van Houten et al., 2005) rule-based scenarios are imbedded in *The Distributor Game* which is the first of a series of management games developed for today's supply chain management challenges. An interactive scenario-based training simulator in (Magerko and Laird, 2002) presents a pedagogical tool that provides a way to dynamical modification of a scenario to ensure that the training goals are achieved.

Some authors' experiences of learning scenario development and their applications for ILMG game (*International Logistics Management Game*) management are described in (Grubbstrom et al., 2005).

In real life the procedure described in (Kononov, 2001) provides effective semiautomatic tool for scenario generation of social and economical systems behaviour planning. The applied areas are quite wide: a management to provide ecology security; researching sociological and economical system; scheduling and planning, etc. Business scenarios implemented within Architecture Vision are used to define relevant business requirements, and to build consensus with business management and other stakeholders (TOGAF, 2006). A natural language-based procedure in (Prado Leite, 2000) is used to capture the requirements and means for communication between stakeholders. Real life scenario structures and development procedures could also be adapted to simulation-based environments and games.

CONCLUSIONS

This paper gives an insight into scenario approach that we believe can significantly increase the effectiveness and quality of modern business simulation games. In our further research we suppose to formalise simulation game scenario development procedure as it has not been done till now. As a basic method we suppose to take trees as we consider them the most promising.

ACKNOWLEDGMENTS

This work has been partly supported by the European Social Fund within the National Programme "Support for carrying out doctoral study programs and post-doctoral research" project "Support for the development of doctoral studies at Riga Technical University".

The present research was partly supported by the ECLIPS project of the European Commission "Extended collaborative integrated life cycle supply chain planning system".

REFERENCES

- Bikovska, J., G.Merkuryeva, R.W. Grubbstrom, 2006. Enhancing Intelligence of Business Simulation Games. *20th European Conference on Modelling and Simulation*. Proceedings of the International Conference, Bonn, Sankt Augustin, Germany, p.641-646.
- Dobson, M., V. Kyrylov, T.Kyrylova, 2004. *Decision Training Using Agent-Based Business Strategy Game*. Proceedings of the Seventh IASTED International Conference "Computers and Advanced Technology in Education", August 16-18, Kauai, Hawaii, USA, p. 66-71.
- Ericsson T., Ritchey T., 2002. *Scenario Development Using Computerised Morphological Analysis*. URL: <http://www.swemorph.com/pdf/cornwallis3.pdf> (available online 20.02.2007).
- Grubbstrom R.W., G, Merkuryeva, J.Bikovska, J.Weber, 2005. Learning Arrangements and Simulation Scenarios. *19th European Conference on Modelling and Simulation*. Proceedings of the International Conference, June 1-4, Riga, Latvia, p. 715-720.
- Kindley R.W., 2002. "Scenario-Based E-Learning: A step Beyond Traditional E-Learning". URL: www.learningcircuits.org/2002/may2002/kindley.html (available online 20.02.2007).
- Kononov D.A., Kosyachenko S.A, Kulba V.V., 2001. *A Scenario Methodology as Connectivity from Strategy to Operation in Complex System*. Proceedings of the CONSA Conference at Linkoping University, Sweden, p. 57-70.
- Kononov D.A., Kulba V.V., Kovalevsky S.S., Kosjachenko S.A., 1999. *Development of scenario spaces and the analysis of dynamics of behaviour of social and economic system*. Preprint, Moscow. *In Russian*.
- Kulba V.V., Kononov D.A., Kosjachenko S.A., Shubin A.N., 2004. *Scenario Development Methods of Social and Economic Systems*. SINTEG, Moscow. *In Russian*
- Luger G.F., 2005. "Artificial Intelligence: Structures and Strategies for Complex Problem Solving". Addison Wesley.
- Magerko, B., J.Laird, 2002. Towards Building an Interactive, Scenario-based Training Simulator. *Proceedings of the Behavior and Representation and Computer Generated Forces Conference*.
- Merkuryeva G., 2000. "Computer Simulation in Industrial Management Games". In *Proceedings of IFAC Symposium*. Rio, Greece, July 12-14, p. 71-75.
- Merkuryeva, G., J. Bikovska, R.W. Grubbstrom, J.Weber, 2004. Development of Learning Scenarios for Network-Based Logistics Simulation Game. *Computer Science. Information Technology and Management Science. Scientific Proceedings of Riga Technical University*, Volume 20. RTU, Riga, p. 148-156.
- Nilson, N.J., 1980. Principles of Artificial Intelligence, Tioga.
- Prado Leite J.C., G. Hadad, J. Doom., G.Caplan, 2000. A Scenario Construction Process. *Requirements Eng 5:38-61*, Springer-Verlag London Limited.
- Rolland C., Achor Ben C., Cauvet C., Ralyte, etc. 1996. *A proposal for scenario classification framework*. URL: <http://cui.unige.ch/~ralyte/publications/REJournal.pdf> (available online 20.02.2007).
- TOGAF - *The Open Group Architecture Framework*, 2006. URL: http://www.opengroup.org/architecture/togaf8-doc/arch/p4/bus_scen/bus_scen.htm (available online 20.02.2007).
- Uchitel S., Kramer J. Magee J., 2003. *Modelling Undefined Behaviour in Scenario Synthesis*. 2nd International Workshop on Scenarios and State Machines: Models, Algorithms, and Tools. Portland, Oregon, USA. URL: <http://www.doc.ic.ac.uk/~su2/SCESM/papers/Uchitel.pdf> (available online 20.02.2007)
- van Houten, S.P.A., Verbraeck, A., 2006. "Controlling Simulation Games Through Rule-Based Scenarios". *Proceedings of the 2006 Winter Simulation Conference*. December 3-6, Portola Plaza Hotel, Monterey, CA, USA, p. 2261.
- van Houten, S.P.A., A. Verbraeck, S. Boyson and T. Corsi, 2005. Training for Today's Supply Chains: An Introduction to the Distributor Game. *Proceedings of the 2005 Winter Simulation Conference*, p. 2338.

AUTHOR BIOGRAPHIES

JANA BIKOVSKA, Mg.sc.ing, Ph.D. student, teaching assistant at the Department of Modelling and Simulation, Riga Technical University, Latvia. E-mail: jana@itl.rtu.lv

GALINA MERKURYEVA, Dr.tech.sc., Dr.sc.ing., Professor at the Department of Modelling and Simulation, Riga Technical University, Latvia. E-mail: gm@itl.rtu.lv