DOMAIN SPECIFIC SIMULATION LANGUAGE FOR IT RISK ASSESSMENT

Artis Teilans,
Arnis Kleins,
Ojars Krasts
Exigen Services
29A Sergeja Eizensteina street
LV 1079, Riga, Latvia
E-mail: artis.teilans@exigenservices.com

Andrejs Romanovs,
Yuri Merkuryev,
Pjotrs Dorogovs
Department of Modelling and Simulation
Riga Technical University
1 Kalku street, LV-1658, Riga, Latvia
E-mail: andrejs.romanovs@rtu.lv

KEYWORDS
IT risk, risk assessment, domain specific language, UML, CORAS, modelling.

ABSTRACT

Information technology systems represent the backbone of a company's operational infrastructure. A company's top management typically ensures that computer software and hardware mechanisms are adequate, functional and in adherence with regulatory guidelines and industry practices. Nowadays, due to depressed economic and increased intensity of performed operations, business highly recognizes the influence of effective Information Technology risk management on profitability.
Design of Unified Modelling Language (UML) based Domain Specific language (DSL) described in this paper achieves synergy from in IT industry widely used UML modelling technique and the domain specific risk management extensions. As a novelty for UML modelling, especially for simulation purposes, the presented DSL is enriched by a set of stochastic attributes of modelled activities. Such stochastic attributes are usable for further implementation of discrete-event system simulators.

INTRODUCTION

Nowadays, business recognizes a great influence of effective risk management on profit abilities. Therefore risk management techniques have become an important part of the company's management instrument. There is no general agreement on the most suitable definition of risk for economists, decision makers, and IT theorists. As a result, different types of risks and, respectively, different risk management methods are considered in different areas. Four main general types of risk can be recognized in business,: strategic, market, credit and operational risks. In many companies, Information Technology (IT) related risk is considered to be a component of operational risk. However, Information Technology risk consists not only of breakdowns in computer software or hardware, or lack of expertise of the IT staff. IT risk also may relate to risk of loss resulting from theft of company's data or client information. IT risk also may be the risk of loss that originates from computer software malfunction, such as a manufacturer's software license expiration or glitches, and the ways it affects corporate activities. A risk assessment initiative for IT systems generally helps management understand areas in which significant losses may arise. IT risk assessment is carried out by identifying and evaluating assets, vulnerabilities and threats of using information technologies in business. An asset is anything that has value to the company – hardware, software, people, infrastructure, data, suppliers and partners, etc.
Taking into consideration the extreme complexity of IT risk assessment, we conclude that there is necessity to apply international frameworks of IT governance and risk management, such as Enterprise Risk Management Framework by Committee of Sponsoring Organizations of the Treadway Commission, Control Objectives for Information and related Technology, Code of Practice for Information Security Management, Information Technology Infrastructure Library, etc. (Klimov et al. 2008, Romanovs et al. 2008)
Within our research, an IT risk management domain specific language is developed. Nowadays, in the IT industry, majority of system specifications and procedure descriptions are made using the Unified Modelling Language (UML). UML is a graphical language and it consists from diagrams which are united in a model. The description of a system can be made from just a few diagrams in case of simple system or from hundreds of diagrams in case of a complex system. These diagrams are designed by system architects and system analysts.They are used in whole life cycle of a system. These models are frequently the main documentation for the system that is used for its operation and maintenance. That is why the authors have chosen UML as the base for designing the IT risk analysis DSL. UML uses general system organization terms such as Use Case, Activity, Action, State, Event etc. However, risk analysis professionals work with terms such as Threat, Vulnerability, Asset, Incident, Risk, Treatment etc. Therefore, to create an IT risk analysis tool, it was necessary to extend UML modelling approach with elements used by risk analysts. In fact there was an attempt to develop our own Risk analysis Domain specific modelling language, suitable for system developers and maintenance personnel and

Proceedings 25th European Conference on Modelling and Simulation ©ECMS Tadeusz Burczynski, Joanna Kolodziej Aleksander Byrski, Marco Carvalho (Editors)
ISBN: 978-0-9564944-2-9 / ISBN: 978-0-9564944-3-6 (CD)

for risk analysts as well. Design of modelling tools necessary for risk analysts was based on CORAS language which is well known in professional community (Lund et al., 2010). The CORAS language is a graphical modelling language for communication, documentation and analysis of security threat and risk scenarios in security risk analyses. This paper explains how the authors use CORAS Threat and Treatment diagrams, connecting them with UML Uses Case and Activity diagrams (Kleins et al., 2008). The result of this work provides means to unify both risk analysis model and IT system model.

COMMON IT RISK MANAGEMENT PROBLEMS

It is possible to indicate a set of IT risks management problems which are typical for Latvian business (Klimov et al., 2008). They are:

- customer service malfunction due to interruptions of continuous access to IT services;
- unsatisfied demand for qualified IT personnel;
- delayed modernization of information systems software and hardware;
- insufficient IT qualification of information system users;
- inadequate level of existing IT services quality monitoring;
- inadequate level of cooperation between IT specialists and other employees;
- inadequate assessment of financial losses resulting from failures or interruptions within information systems;
- absence of IT system development strategic plan, based on a general development plan of company;
- inadequately low IT security level;
- absence of strategy of IT system restoration after potential failures and interruptions.

Taking into consideration the extreme complexity of IT risk management within the framework of operational risk management system, it could be concluded that it is necessary to apply international standards and frameworks of IT governance, such as Information Technology Infrastructure Library, Control Objectives for Information and related Technology, Code of Practice for Information Security Management.

The proposed technique for IT risk assessment and management could be successfully used as a start point for development of the IT risks assessment support systems prototype, based on an IT risk management domain specification language with a metamodel that defines an abstract UML based language for graphical approach to identify, explain and document security threats and risk scenarios. The next chapter describes the Domain Specific Language (DSL) for IT risk analysis modelling and simulation. In the chapter presented tool will provide both IT process modelling and documentation as well as connection of these processes with identified risks.

DSL FOR IT RISK ANALYSIS

A Domain specific language (DSL) is language for programming, specification or modelling suitable for particular problem domain specialists to solve their specific technical tasks (Achim et al. 2007, Lenz and Wienands 2006). This chapter describes domain specific language for IT risk analysis designed by the authors. This language has organically emerged from unifying several methods and graphical languages which are used by developers and maintenance specialists from information systems domain, and also analysts responsible for risk analysis and risk mitigation activities for IT systems. The designed DSL is based on approach to Unified Modelling Language (UML) (Kleins et al. 2008), CORAS method (Lund et al. 2010) and Misuse Case Alignment Method (Sindre and Opdahl 2000; Matulevicius et al. 2008).

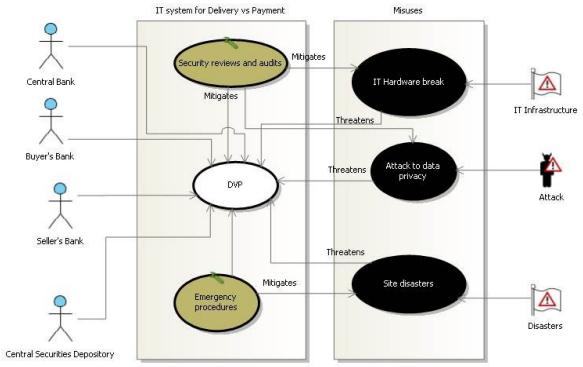

Figure 1: DVP IT System Use cases

Currently, using UML is one of the most commonly used approaches in IT system modelling. The authors' experience acquired while working in IT industry shows that UML modelling is used to some extent in all medium and large scale projects.

UML belongs to the group of graphical modelling languages. Initially UML was built for information systems modelling to facilitate the development and maintenance processes. Nowadays the usage of UML is broadened. This language is used for building business models, which exceed the initial task of modelling of information systems.

As regards system modelling, UML modelling is widely used at systems development or enhancement phases. UML modelling describes the structure and behaviour of the system. This language consists of graphical notations called diagrams and builds up an abstract model of a system. The UML standard is maintained by OMG (Object Management Group). In the beginning, UML was built for specification visualization and documentation of IT systems development. Nowadays usages of UML are not only limited to tasks of software engineering. UML is also used for business process

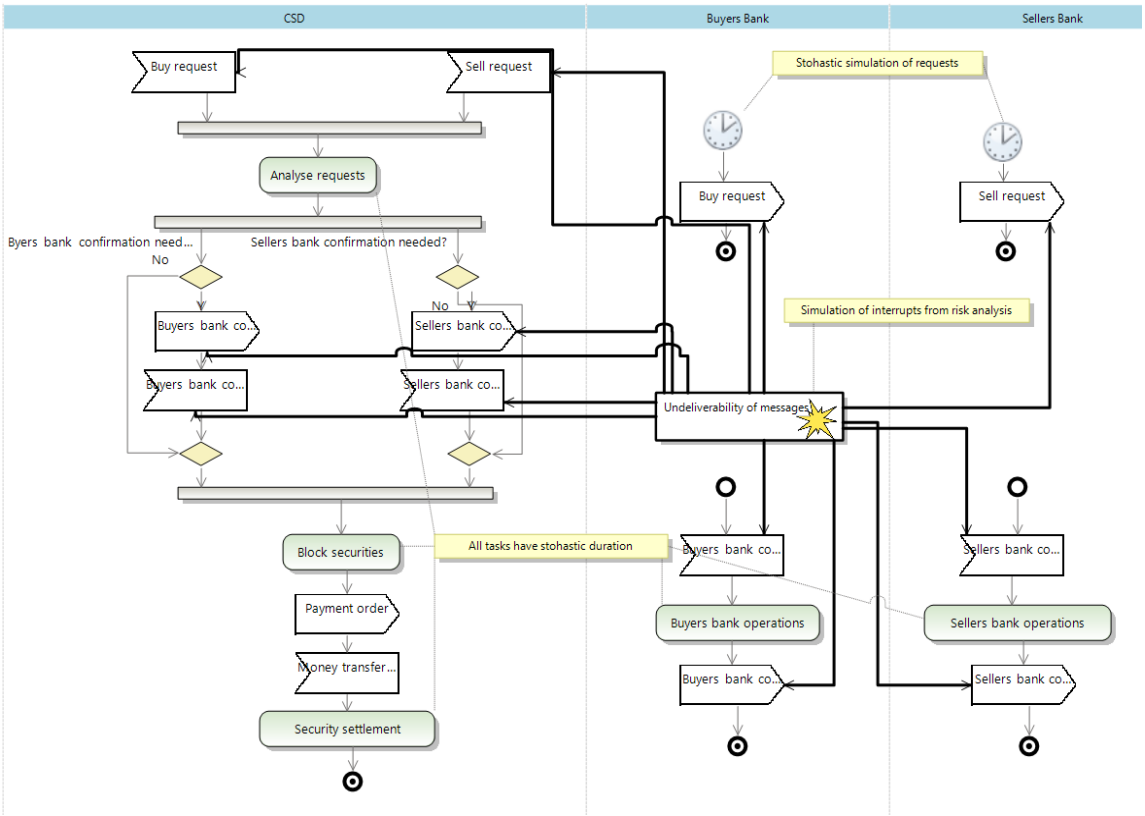

Figure 3: DVP Activity diagram

modelling and for the development of systems which are not pure information systems.

Modelling with UML promotes model-driven technologies, such as Model Driven Development (MDD), Model Driven Engineering (MDE) and Model Driven Architecture (MDA). Supplementing graphical notations with terms such as class, component, generalization, aggregation and behaviour, helps save

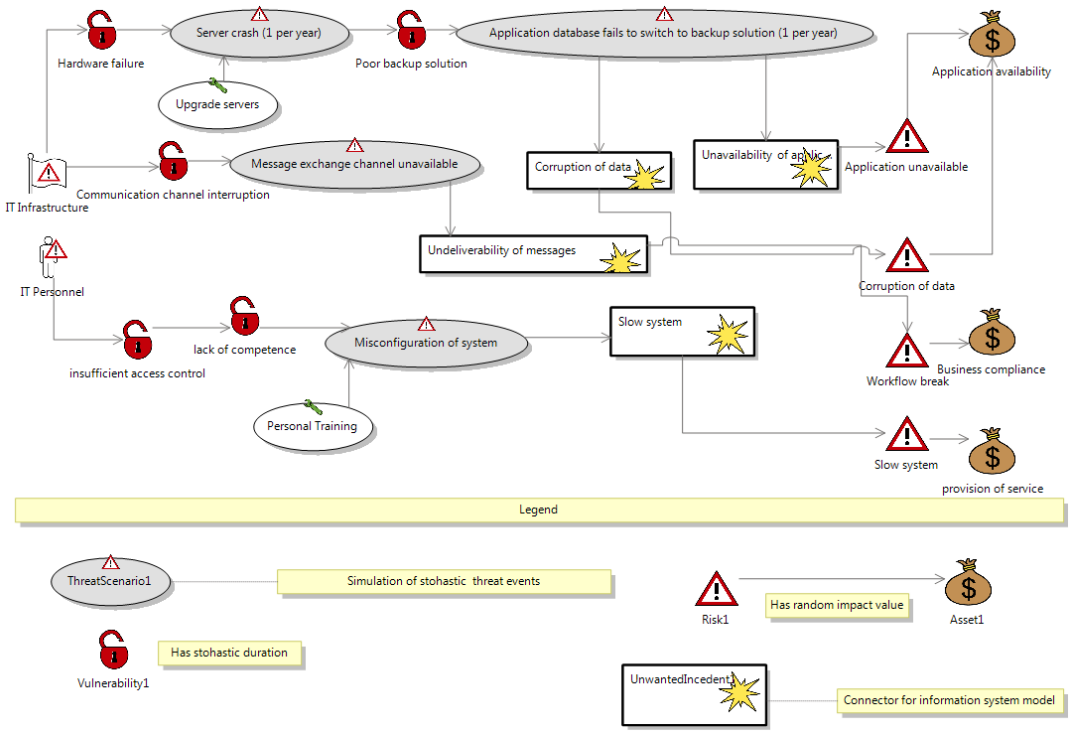

Figure 2: Treatment diagram for IT Hardware break

system designer's time for system architectural tasks and design.

A UML model consists of a set of diagrams. A diagram is a partial representation of the model. A system model could be divided into two parts. The first part is a functional model, which reflects functionality of a system from the system user's point of view. This kind of model is constructed using Use Case diagrams. The second part is the dynamical model that reflects internal behaviour of the system. A model of that kind is constructed using Activity, State, Sequence and Collaboration diagrams.

A system model to be created with UML language should not necessarily contain all diagrams. For example, when creating Information System vision model or requirement specification, it is enough for the system analyst to create Use Case and Activity diagrams. Use Case diagram answers a question – what a system does. Activity diagrams describe scenarios of every Use Case, i.e., Business processes. Therefore we prefer this work use only Use Case and Activity diagrams.

As mentioned above, IT industry use of UML is mostly directed to specification and documentation of a system. The authors as representatives of simulationist community would like to improve this situation and to add more dynamic to this static construction. Obviously simulation of the model can give to developer's possibilities to evaluate and forecast behaviour of a target system. The authors already addressed this issue in (Kleins et al. 2008). During development of the presented DSL for IT risk analysis, which is based on UML, one of the objectives was possibility of simulation of a model. Activity Diagram elements are complemented with stochastic attributes for simulation purposes (Table 1).

Table 1: Stohastic attributes of Activity diagram

UML element	**Stohastic attribute**
Task	Duration
Branch	Decision probabilities
Timer	Start Delay
	Number of Events in group
	Delay between groups
	Number of Groups

One more approach for the developed DSL is application of Misuse Case in a UML Use Case model. Misuse cases improve UML diagrams with a better support to analyse problems of IT risk management. The *Use Case* diagram is extended with graphically black *Use case*, called *Misuse Case* and black *Actor* called *Misuser*. *Misusers* are related with *Misuse Case*. *Misuse cases* are related to *Use Cases* with relation <threatens>. During risk analysis stage *Use Case* diagrams are extended with additional *Use Cases* for risk mitigation, which are connected with system *Use Case* with relation <include> and with *Misuse Case* with relation <mitigate> (see Figure 1).

Considering that the task to be solved by the authors was to provide a government institution responsible for IT risk evaluation with tools necessary for such tasks, the third technology used in this work is security risk modelling, analysis and documentation language CORAS. The initial CORAS approach was developed within the CORAS project funded by the European Commission that ran from 2001 until 2003. CORAS is both a language and a methodology for its application, described in the book (Lund at al. 2010). Although initially CORAS was designed for security risk analysis, its syntax and semantics allows applying this language to complete IT risk analysis scope. In the developed prototype only one CORAS language diagram - the

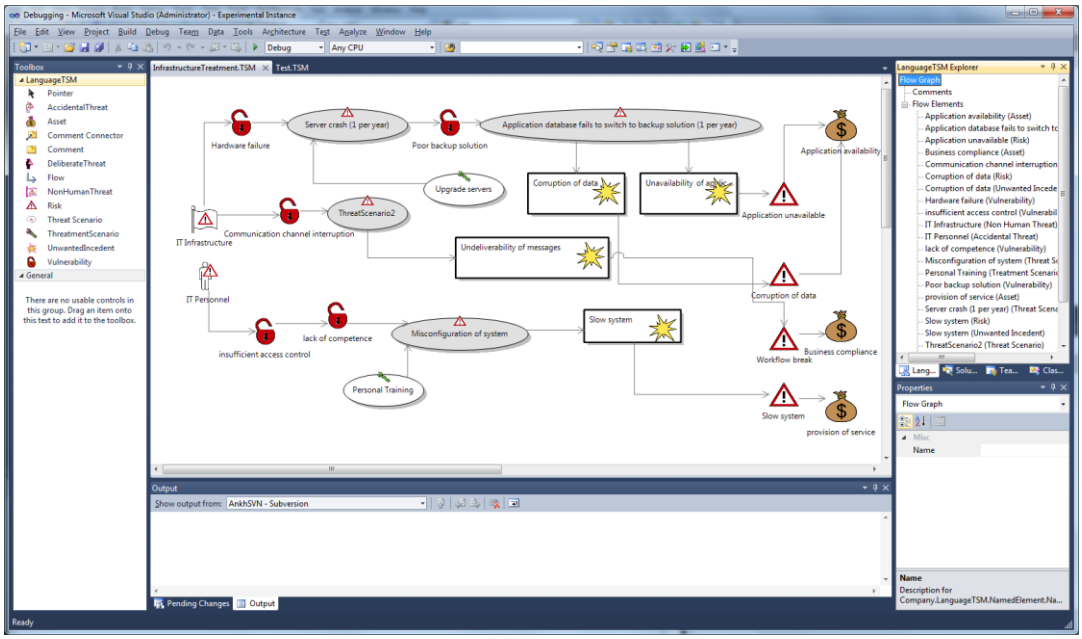

Figure 4: IT risk analysis tool

Treatment diagram - is used. Treatment diagram is CORAS method all-inclusive diagram, in which all main risk analysis entities – *Threat*, *Vulnerability*, *Risk*, *Asset*, *Threat Scenario*, *Unwanted Incident* and *Treatment Scenario* are included. In turn, by methodology developed by the authors, *Unwanted Incident* is common entity, which connects risk analysis Treatment diagram with UML Activity diagram used in IT system Activity diagram model (see Figure 2).
Additionally, we did similar enhancements to CORAS Treatment diagram as we did with UML activity diagram. For simulation purposes, Treatment diagram is complemented with stochastic attributes (Table 2).

Table 2: Stohastic attributes of Treatment diagram

Diagramm element	**Stohastic attribute**
Relation between Risk and Asset	Impact
Unwanted incident	Used as connector between risk and system models. Transfer events from treatment scenario to system model. Event raises a disability of selected activity of a system model.
	Duration of disability
TreatmentScenario	Start Delay
	Number of Threat events in group
	Delay between groups
	Number of Groups

Using the DSL described in the paper, a corresponding Activity diagram describing IT system functionality should be designed for each system Use case, a corresponding risk mitigation Activity diagram for each risk mitigation Use case should be designed, and Treatment diagram should be designed for each Misuse Case (see Figure 3).
Simulation will allow to perform simulation experiments on two models simultaneously - the risk analysis model and IT system model and gather more adequate risk estimation results.
For such IT risk analysis approach, a tool prototype which is based on Microsoft Visualization and Modelling SDK (VMSDK) is developed while designing DSL. This implemented modelling tool is functioning inside Microsoft Visual Studio Shell. It could be distributed either with Microsoft Visual Studio Shell, or as Microsoft Visual Studio Add-In (see Figure 4). Additionally this approach ensures ability of simulation program code generation, compilation and execution for any Microsoft .NET Framework supported language. Specially designed templates ar used for code generation purposes, and they consist of code snippets for simulation of diagram elements. The authors currently are working on this solution.

Another approach is code generation from DSL diagrams for some general purpose simulation package (for example ARENA).

CONCLUSIONS

The current situation within business indicates the necessity for more complicated and more effective IT risk management system development. In the presented paper the given approach allows to perform IT risk analysis which is based on the unified IT system model specification. In this way the one window approach is realised for both system developers and maintainers and for those responsible for the security policy of a system.
The presented DSL and modelling tool are still in the early stages. Further work will be performed to improve the Domain specific language. The second group of further activities will be devoted to implementation of an appropriate simulation engine. Model repository and tools for storing and processing simulation results will be developed for domain specific decision support.
This approach will be approved on state-wide IT systems and important financial sector IT systems.

REFERENCES

Artis Teilans, Arnis Kleins, Uldis Sukovskis, Yuri Merkuryev, Ivars Meirans. 2008. A meta-model based approach to UML modelling. Proceedings of EUROSIM/UKSIM 10th International Conference on Computer Modelling & Simulation. Emmanuel College Cambridge, UK, April 1-3.

Arnis Kleins, Yuri Merkuryev, Artis Teilans, Maxim Filonik. 2008. A meta-model based approach to UML modelling and simulation. Proceedings of the 7th International Conference on System Science and Simulation in Engineering. Venice, Italy, November 21-23.

Achim D. Brucker, Juergen Doser. 2007. Metamodel-based UML Notations for Domain-specific Languages. Workshops and Symposia at MoDELS, Nashville, TN, USA, September 30 - October 5, Reports and Revised Selected Papers. Lecture Notes in Computer Science. Volume 5002, 2008, DOI: 10.1007/978-3-540-69073-3

Gunther Lenz, Christoph Wienands. 2006. Practical software factories in .NET, Berkeley, CA : Apress ; New York : Distributed by Springer-Verlag.

Klimov R., Reznik A., Solovjova I., Slihte J. 2008. The Development of the IT Risk Management Concept. Scientific Proceedings of Riga Technical University, Vol.5, 131-139.

Romanovs A., Merkuryev Y., Klimov R., Solovjova I. 2008. A Technique for Operational IT Risk Management in Latvian Monetary and Financial Institutions. Proc. of 8th WSEAS International Conference on Applied Computer Science „Recent Advances on Applied Computer Science" 230-235.

Mass Soldal Lund, Bjørnar Solhaug, Ketil Stølen. 2010. Model-Driven Risk Analysis. The CORAS Approach. Springer.

G. Sindre and A. L. Opdahl. 2000. Eliciting Security Requirements by Misuse Cases. In Proceedings of the TOOLS Pacific.

Raimundas Matulevicius, Nicolas Mayer, Patrick Heymans. 2008. Alignment of Misuse Cases with Security Risk Management. Proceedings of the The Third International

Conference on Availability, Reliability and Security, ARES 2008, March 4-7, 2008, Technical University of Catalonia, Barcelona , Spain. IEEE Computer Society, 1397-1404.

AUTHOR BIOGRAPHIES

ARTIS TEILANS was born in Riga, Latvia and graduated the Riga Technical University, where he studied automatic and remote control and obtained his doctor degree in 1999. He is working in software industry at Exigen Services Latvia. In academic field he is a senior researcher at the Riga Technical university and an Associate professor at Rezekne Higher Education Institution as well. His professional interests include tecniques of system modeling and discrete-event simulation. His e-mail address is : artis.teilans@exigenservices.com.

ARNIS KLEINS was born in Riga, Latvia and went to the Riga Technical University, where he studied information technology. He obtained his master degree in 1996. He is working at Exigen Services in Riga. From 2010 he is a doctor student at the Riga Technical university. His professional interests include methology of system modeling and discrete-event simulation. His e-mail address is : arnis.kleins@exigenservices.com.

OJARS KRASTS graduated the Riga Technical University in 1986 as a computer hardware engineer. Since that, he has worked in software development field. He participated in development of modeling toolset GRADE for six years, and then worked in Latvian Central Depository for another six years, where settlement system risk analysis was one of his tasks. Current research work, based on that experience, is about domain specific modeling languages for settlement systems, model analysis and simulation. His e-mail address is: ojars.krasts@exigenservices.com.

ANDREJS ROMANOVS is an Associate professor of Riga Technical University, doctor of engineering sciences. His professional interests include modelling of management information systems, IT governance, logistics information technologies and electronic commerce, as well as education in these areas. He has 20 years practical experience in development of more than 50 data processing and management information systems in Latvia and abroad for state institutions and private business as IT project manager and system analyst. He is a member of IEEE and LSS; he participated in international scientific conferences and research projects; with scientific publications in the field of ICT. His e-mail address is: andrejs.romanovs@rtu.lv.

YURI MERKURYEV is a Professor and the Head of the Department of Modelling and Simulation at Riga Technical University in Riga, Latvia. His professional interests include methodology of discrete-event simulation, supply chain simulation and management, as well as education in the areas of simulation and logistics management. Prof. Merkuryev is a Corresponding Member of the Latvian Academy of Sciences, President of Latvian Simulation Society, Board Member of the Federation of European Simulation Societies (EUROSIM), SCS Senior Member and Director of the Latvian Center of the McLeod Institute of Simulation Sciences, Chartered IT Professional Fellow of the British Computer Society, and associate editor of *Simulation: Transactions of The Society for Modeling and Simulation International*. He has served as General Chair at the International Conference "European Conference on Modelling and Simulation", ECMS'2005 that has been held in Riga, Latvia, in June 2005. Prof. Merkuryev has promoted 7 PhD theses. He is an author of more than 300 scientific publications, including 6 books. He is a co-editor of the book “Simulation-Based Case Studies in Logistics: Education and Applied Research”, published by Springer in 2009. His e-mail address is merkur@itl.rtu.lv and his web-page can be found at http://www.itl.rtu.lv/mik/ymerk.html.

PJOTRS DOROGOVS - master of engineering sciences, Ph.D. student at Department of Modelling and Simulation, Riga Technical University. Graduated from RTU and earned the master of IT project management qualification (Mg.sc.ing., 2008). He focused his professional interests on the business modelling, Information Technology security and governance. Since 2006 the Head of National Schengen Information system unit of the Information centre of the Ministry of the interior of the Republic of Latvia. Member of IEEE, participated in international scientific conferences and research projects; with scientific publications in the field of ICT. His e-mail address is pjotrs.dorogovs@inbox.lv.