

Using agent and UML technologies in modelling organizations: the case of a vehicle theft

Raul Savimaa

Department of Computer Control, Tallinn University of Technology, Ehitajate tee 5, 19086 Tallinn, Estonia; raul.savimaa@dcc.ttu.ee

Received 4 July 2003, in revised form 14 December 2004

Abstract. The paper considers modelling the emergent behaviour of human multifunctional organizations. A combined modelling methodology is introduced that unites UML, the Q-model and multi-agent approach. UML is used for modelling the processes, the Q-model for analysing timing characteristics and multi-agent model for simulating interactions between actors. Although this approach needs further refinement, it can already be used for modelling the time-dependent emergent behaviour in organizations. As an example, the case of the theft of a vehicle has been considered.

Key words: modelling of organizations, multi-agent systems, simulation, emergent behaviour, UML.

1. INTRODUCTION

Reliability of work processes and ability to offer service on the guaranteed quality level express maturity and trustworthiness of a human-based organization. There is an ongoing need to review and modify existing business processes in organizations in order to ensure efficiency and to cope with dynamics of the social environment. Objective of the present paper is to find a suitable approach for modelling the behaviour and designing modifications of processes in a human-based organization. The organization should operate effectively in a dynamic environment where everyday response to emerging situations must be made in minutes and seconds and, in addition, pre-defined work routines should be modified from time to time according to long-term changes in the environment, still keeping in mind the existing normative basis. In a large organization a modification of some of the processes for improving the ways to achieve the goals, or reconsideration of the goals themselves is usually going on. These

modifications are often made intuitively. Thus the results depend on subjective capabilities of the leaders. A tool that could analyse and simulate the results of planned modifications would help in planning and re-designing organizational processes.

Need for such modelling exists, for example, in the law enforcement domain. In the case of a stolen vehicle, the police has a set of pre-specified work processes, including forms for information exchange and updating databases. The desired result – to find the stolen vehicle quickly – can be achieved by efficient activities of all the related authorities. Therefore modelling should analyse not only actions of the police but of the overall system: what are the related possible activities of the owner, the thief, other authorities, and what is the priority of the current task for the police relative to other tasks. The final target of such analysis is to give suggestions for improving information exchange and related activities in the organization in order to ensure the best performance.

As a solution to the given problem, this paper describes a novel methodology for integrated modelling of the work (business) processes and behaviour of actors (humans) in an organization.

The paper is structured as follows. In Section 2 it will be shown that the problem cannot be solved using conventional methods and grounds for elaborating a suitable methodology is introduced. The methodology itself is described in Section 3. In Section 4, as an example, the case of a stolen vehicle is solved using the introduced methodology. Conclusions are drawn in Section 5.

2. APPROACHES TO MODELLING ORGANIZATIONS

2.1. Basic requirements

Planned behaviour of an organization stems from the goals, general tasks, and strategic plans. It is adjusted to changing requirements and emerging situations. Inside the organization several aspects, such as the goal functions for different units, bonuses, specific features of the organizational structure and others influence the behaviour of single actors (employees). Often choice of an activity depends not only on objective criteria about the situation or task priorities, but also on the subjective criteria of the actors (like motivation, intentions and personal preferences). The resulting behaviour is therefore an integral reflection of all these factors and its formation can not be completely described in detail. For that reason it is not sufficient to analyse only pre-defined business processes but models of the actors' behaviour must be added into the model of the organization.

Existing analysis and optimization tools for organizations like Gantt and PERT charts, SWOT and PEST analysis, Balanced Scorecard [¹], Best Value [²], enterprise modelling (EM) methods [³] and other conventional process modelling methods (whose aim in general is only to design an information system for supporting pre-defined processes), cannot solve this problem since the emphasis

is on the specification of a single set of work processes. The dynamics of work processes and their possible modification in time are usually not considered and not supported. Actors and their subjective capabilities and dynamically changing attitudes are not considered either. The EM approach (e.g. EKD [3]) is the most promising and can be used also for planning global modifications in an organization. Although successfully implemented by modification of large organizations in power supply industry, EKD has two major weaknesses from the point of view of our problem. First, it does not support modelling of subjective capabilities of human actors; second, the project work is often very voluminous and models do not consider detailed characteristics of low-level processes. Besides, no actual computerized support tool is available yet.

A suitable methodology should integrate modelling an organization from two viewpoints: from process perspective and from actors' perspective, keeping in mind that actors are humans with very low predictability of behaviour but who in general are motivated to follow existing rules and norms in the organization. Behaviour of a human actor cannot be modelled as a single algorithmic process, since the actual behaviour is emerging and depends on the concrete situation. Therefore primitive components for the modelling of such a behaviour can be a set of loosely coupled, interacting and repeatedly activated algorithms [4] that may give acceptably similar result [5]. In this case the agent-oriented approach [6] is the most useful one for modelling the behaviour of actors in an organization.

2.2. Specification of organizations using UML

The Unified Modelling Language [7] is a graphical language, accepted by the Object Management Group [8], suitable for specifying and documenting the artefacts of a software-intensive system [7]. It is widely used in industry, in academic circles and major software companies, mainly for modelling information systems. UML describes different parts of the model in one language. In some cases (e.g. by formal verification or modelling very specific systems) specific languages are used (e.g. the Q-model [9]). Such a possibility is included in UML specifications. For instance, UML Profile for Performance, Schedulability and Time [10] and the conversion from UML to another language and back can be implemented semi-automatically, if such a tool has been elaborated.

In the present approach, UML is considered as the best suitable modelling notation due to its characteristics, availability and popularity. UML satisfies the requirements of this approach: it permits the modelling of different aspects of the system, it is logical and visual, therefore the diagrams can be also manually interpreted. It has also several software tools that support sophisticated modelling or illustration of diagrams. Since UML can be widely used already for modelling information systems and related processes, it is recommended to use the same tool also for modelling the behaviour in order to support development of the corresponding information system.

2.3. Organization as a multi-agent system

Socially intelligent agents [^{11,12}] are used to investigate and model human-style intelligence in a social group (team) in order to design useful intelligent agents [¹³]. Using agents is one of the most suitable approaches for representing group communications and interactions. Agent technologies are widely used in different applications (e.g. in office, banking, commerce, military, social services [¹⁴⁻¹⁶]) including management of complex commercial and industrial processes [¹⁷].

The term “agent” in this paper means a simulated entity that represents a human or artificial actor (a physical or organizational entity) in a modelled system and that can be implemented in the model of the organization as a description or an intelligent piece of software (similarly to the general interpretation of agents [^{6,15}]). Some of the agents can be later used for designing software agents in the information system of the organization.

Agents used for modelling human organizations must be socially proactive: they must possess the ability to communicate with other agents [⁶] and to decide what actions in the current situation are to be taken to maximize progress towards its goals.

Compatibility of goals, sufficiency of resources, abilities of actors in relation to tasks and trustworthiness determine the type of interaction and relationships between actors [^{6,18}]. Designers may use different types of relationships by modelling in order to achieve different behaviour of the organization. To be able to respond to dynamic changes in the environment and to adjust its behaviour to achieve a goal, an agent has to understand the time concept used in that environment [¹⁹]. Perhaps cognitive agents with teleonomic behaviour [⁶] are the most suitable ones for the modelling of humans. Cognitive agents can be designed in software as more complex agents that process a lot of information in order to find the most rational way of behaviour. If models are simplified, other types of agents can be used. In the present paper reactive agents with reflexive (e.g. agent “Owner”), teleonomic (e.g. agent “Thief”) or combined (e.g. agent “Patrolling Officer”) behaviour are used for the illustration.

UML can be used for describing an organization as a multi-agent system (MAS), although certain difficulties may arise by that. UML currently does not fully support the design of MAS; for example, UML does not provide the means of capturing all agent-related modelling aspects like autonomy and pro-activity. Also existing multi-agent approaches are not offering a unified solution for describing agents and their environment in UML. Nevertheless, UML is gaining more and more popularity in agent modelling [²⁰] and an extension of it, called Agent UML (AUML) [²¹], has been suggested. Since AUML is still under elaboration, the existing version of standard UML that is supported by modelling tools (currently version 1.4 with possible manual implementation of some principles from version 1.5) is used for describing MAS.

3. MODELLING METHODOLOGY

The suggested model development cycle consists of four stages. In the first stage the modelling task is formulated and the organization described (often only in natural language). The modelling task consists of prioritized sub-tasks and measurable criteria for later evaluation of the effectiveness of the modelling. The description of the organization should contain the aims, tasks, structure, goal functions, relations to the environment and activity boundaries (both static, as laws or external requirements, and dynamic, as internal normative acts).

3.1. Specification of work processes

The second stage completes the model of the organization. It concentrates on the key processes and related actors. This restriction simplifies the model and reduces the amount of work needed for modelling. Process control approach is used for more traditional modelling of described work (business) processes in the organization. Specification of processes begins with the description of aims, tasks and essential processes actor by actor. A special form of a table of work processes is used to simplify the analysis and its later description in other notations. Each process together with its characteristics is described by a row in the table. Both planned and real work processes of units and employees are described (stored in different sub-models, if necessary).

Use case diagrams in UML are used to describe interactions of the actors and activities. Activity diagrams are used for modelling activities and their sequences. Interaction diagrams can be used for more detailed specification of interactions. Use case and activity diagrams together with the table of work processes should be well coordinated during the whole modelling process, in this way supporting coherence of the description.

The Q-model [9] is used for more precise modelling of timing characteristics of inter-process interaction. The Q-model to a certain extent also supports hierarchical decomposition of the processes. Modelling in the Q-model notation starts from elementary processes that are presented in the table of work processes. Since all real actions may be successful or unsuccessful, then for more realistic imitation it is also important to model “wrong” but realistic outcomes. In the Q-model this can be implemented using selector processes. The Q-model demands very detailed specification of process characteristics, in this way forcing the designer to plan and analyse all related issues. As a result, that often shows gaps in processes and interrelations between processes.

There is no suitable tool available for conversion of use cases, activity diagrams, and interaction diagrams given in the UML notation into the Q-model notation. Therefore this work at the present stage is to be performed manually. The Q-model analysis in this research is implemented and stored in Limits CASE Tool [22] projects.

3.2. Representation and simulation of proactive components of an organization

The second stage in the methodology continues with modelling the organization from the viewpoint of the actor. Organizational behaviour is illustrated using agent simulation. Behaviour of selected key actors, mainly related to achieving a goal (e.g. choosing from multiple tasks, information capturing, processing and exchange) and interactions between actors is described as the behaviour of collaborating agents in MAS. Consideration of an organization as MAS is justified since there are many acting entities with their own objectives and capabilities inside an organization.

In general, all related key actors with their pre-determined (ideal) roles in the organization should be described as agents and also descriptions of artificial agents (robots, software agents, etc) may be added. In reality it is too voluminous and for each particular simulation only a suitable set of agents and their characteristics can be chosen (e.g. for demonstration of information exchange or analysing cooperative activities). Each agent has methods that can be considered as work processes for the actor that the agent represents. Actually, implemented agents in the simulation may be simpler than actors described in the model and the simulation may cover only a part of the model (for example, some specific interactions or information exchange). Internal description of an agent (e.g. its methods) may be presented in UML and analysed in the Q-model [23]. Use case and activity diagrams are used for specifying the dynamic behaviour of agents.

Although the actual behaviour depends on different personal characteristics like motives, intentions, interests (e.g. career interests), ethical and moral norms (versus the goals of the organization), or sympathies and antipathies between employees, it is very difficult to model those characteristics in relation to work processes. Therefore in the present methodology only issues of cooperation, competition of different tasks with similar priority level and choosing of activities, if personal priorities do not correspond to official priorities, are considered. Such personal characteristics are easy to include into the methods and behaviours of the agents as additional internal modules for delays or calculating weights for selected activities.

Currently (as used so far) an agent satisfies the FIPA [24] standards and recommendations. Interaction diagrams are used to model the application aspects of inter-agent communication protocols. Agent-modelling software tool JADE [25] has been selected as a starting platform for the development and simulation of agents. Communication (message exchange) between agents in JADE is FIPA compliant. Cooperation, any kind of control and storing data in the log file results from inter-agent communication. JADE is not the only available environment for developing agents for the implementing simulation model. Other possible environments are, for example, AgentTool [26] and Zeus [27].

3.3. Implementation of modifications

As a result of the second stage of the methodology, the model of an organization consists of four components. The first component comprises description of the organization (in natural language), table of work processes, use case and activity diagrams in UML and process model in the Q-model notation and, if used, also interaction diagrams, an agent class model and agent simulation. Other components of the model are model files in UML, in the Q-model notation, and in agent simulation environment that are kept separately for technical reasons since no unified environment exists so far.

The model of the organization describes it at the moment of modelling. It should be validated and can then be used for analysis and simulations during the third stage of the methodology in order to elaborate suggestions for the modification of the organization. Process of (successful) modelling and simulations should be fixed. The best solution for the improvement of the organization is usually composed of different scenarios [28]. Well-specified modification plans, with a visualized model that estimates possible behaviour, are a basis for actual modification of the organization.

The fourth stage of the methodology is implementation of organizational modifications, monitoring and the evaluation of results.

The model should be updated regularly. This stage may be most time-consuming since so far feasible supporting technology is not developed yet. Therefore also keeping components of the model coherent and updated is a job of the designer.

4. EXAMPLE: MODELLING A VEHICLE THEFT

This section presents a simplified example about implementation of the methodology. The example considers information exchange in police in case of a vehicle theft. The methodology is applied as described in Section 3. Initially the modelling task is set up and the organization is described. Work processes are described in the form of a table using UML use case and activity diagrams. Agent class diagrams are described for three posts in the police: local and central command-control centres and the police patrolling officer. Agent simulation with 12 agents is used.

4.1. Modelling

When a vehicle is stolen, one of the four typical scenarios is usually followed:

- there may be an attempt to re-register the vehicle in the vehicle registration centre,
- the vehicle may be taken abroad,
- the vehicle is dismantled for spare parts,

– the vehicle is used for driving inside the country for some time, after that the thief chooses behaviour 1–3 or abandons the vehicle.

When the owner notices that the vehicle is stolen, he (she) informs the police personally or by phone. The police (the command and control centre in a local police authority) performs all necessary activities for checking and registering the information and informs all other police units, vehicle registration centre and border guard. To prevent illegal actions with the stolen vehicle, it is important that all related authorities have this information as soon as possible.

The modelling task is to evaluate how operative the current information exchange is from receiving the initial information about a car theft until forwarding the data to databases in police, vehicle registration centre and border guard. Is there unnecessary duplication of work? What suggestions can be given for the modification of the related processes?

4.2. Organizational framework and related processes

Three actors in the police play crucial role in information exchange in the given case: the duty officer in the police prefecture (command and control centre of the local police authority, LC), a duty officer in the central command and control centre of the police (CC) and a patrolling officer (PO) in the police prefecture.

The LC is the 24-hour acting centre for organizing the operative police work in a police prefecture. It has a wide range of tasks that are quite well defined in the regulations. Available resources have to be divided for solving multiple tasks concurrently. The main activities of the LC, considered in the current example, are reception, registration and processing information about stolen vehicles, controlling the patrols and answering to the patrol queries. An example of tasks of a duty officer is presented in Table 1. In the first column of Table 1 the structured identity number of the process is given. For example, “J2.4.A” means that this is a modified version (“A”) of the fourth process (“4”) in task 2 of actor LC (“J”).

The tasks of CC in the case of a vehicle theft are to receive messages, check the data in the vehicle register, enter the information into the database of stolen vehicles and forward it to the vehicle registration centre (VRC) and to the border guard. An operator (VRO) in the VRC and an official in the border guard insert the stolen vehicle in their information systems according to the fax received from the police. The VRO and the border guard officer (BGO) at a border point use their databases for confirming re-registration or border crossing and also for detaining the stolen vehicle. The most important work processes of LC and CC are illustrated in the combined use case model in Fig. 1.

Police patrol officers, among other duties, check vehicles and send queries to LC. LC checks the data in different databases and sends the report to PO who then decides to release the vehicle or detain the vehicle and persons.

Table 1. A selection of tasks of a duty officer

ID	Activity	Purpose	Input description	Description of the activity	Duration, min	Output description
II	Receiving calls and controlling of patrols					
J2.2.A	Receiving the call about the vehicle theft	Obstruction of violation	Call	Answering the call, getting information, making decision	2–3	Information to be used
J2.4.A	Registering the information about the vehicle theft	Registration of a possible violence	Call, info, decision	Registering in information system POLIS as initial information, checking the data in vehicle register	2	Correctly entered information
J2.6 +J2.7	Forwarding the call to the patrol	Solving the problem	Decision, choosing the patrol	Calling the patrol, giving task, determination of priority	1	Patrol starts solving the problem
J2.9.A +J5.3.	Re-registering the information about an event	Registering the violence	Confirmation about the statement	The initial record is re-classified to “event” and information is sent (“send” button pressed) to CC	1	Correctly entered information

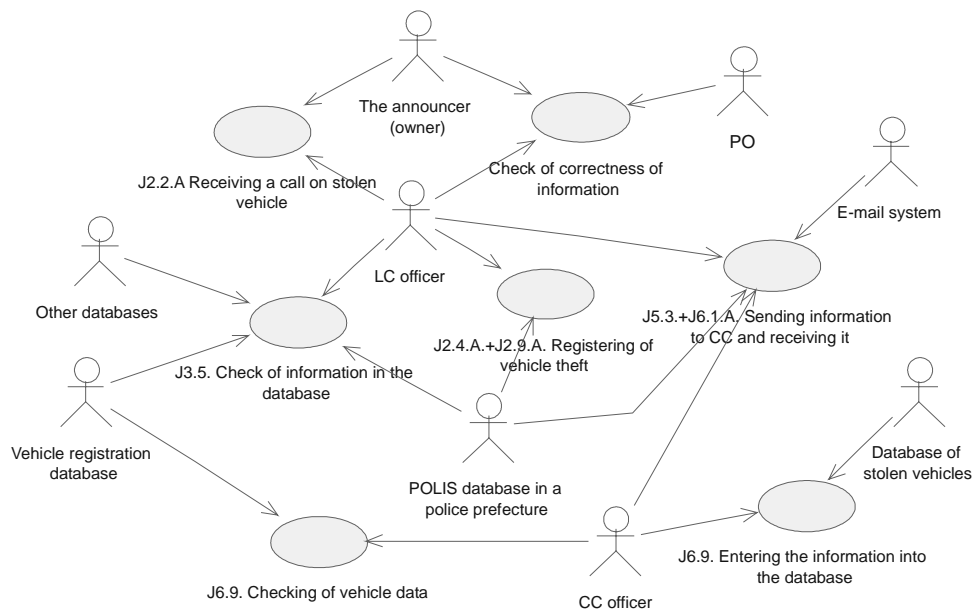


Fig. 1. The use case model of event registration and information exchange.

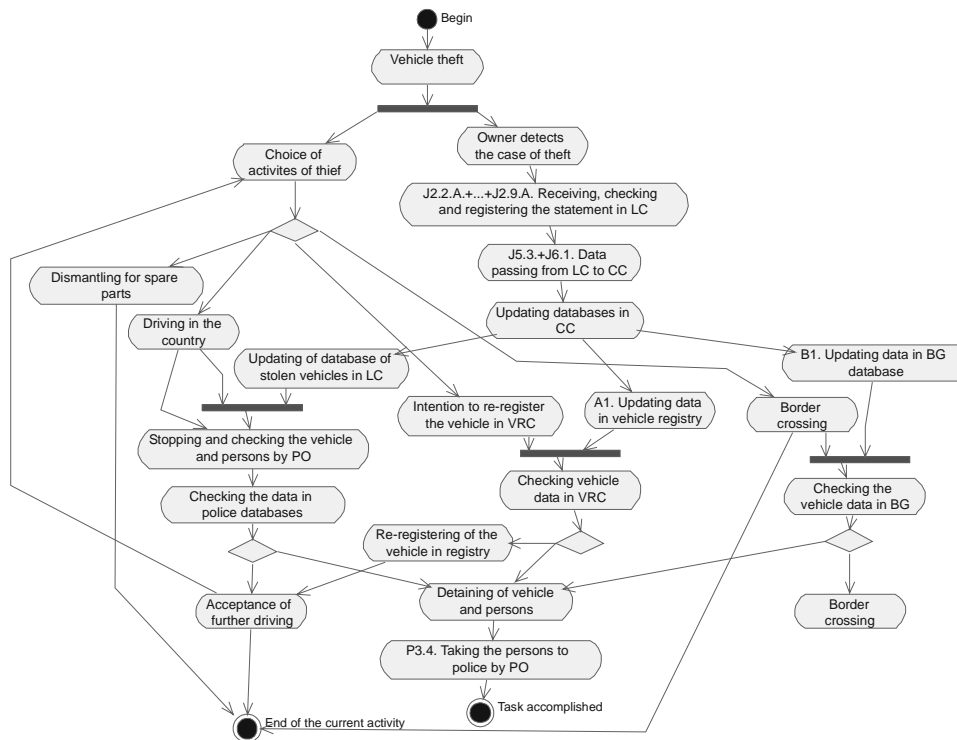


Fig. 2. The activity diagram of the vehicle theft case.

Figure 2 illustrates the concurrence of activities related to the vehicle theft and information exchange. Besides police-related processes also additional activities like the stealing of the vehicle, alternative actions of the thief, and actions related to the vehicle (driving, re-registration, border crossing, dismantling for spare parts) are added. The consequences depend on whether the information is delivered in time (the stolen vehicle is detained) or is delayed (the stolen vehicle may drive further).

Process model in the Q-model notation, resulting from conversion of the use cases and activity diagrams given in UML, shows that multiple time-consuming processes, related to the checking of the vehicle data and information exchange, precede the process of entering the information into the database of stolen vehicles. Unnecessary duplication exists by updating the vehicle register and the border guard database.

4.3. Emerging behaviour of actors

The overall behaviour of the system emerges from the activities of the actors. The actual result (police finds the vehicle or not) depends on the actions of all related actors (owner, thief, police, etc.) and therefore cannot be predicted. At the

same time it is still possible to analyse and illustrate different activities that lead to one or another result.

In the current example a multi-agent simulation in JADE environment with twelve agents was constructed. The actors were LC, CC, PO, BGO, VRO, the owner, the thief, three database engines (agents poldb1, vrdb1 and bgdb1), the vehicle and the logger agent. Interactions between agents were implemented as sending and receiving corresponding messages. The functioning of the multi-agent system in general corresponds to the activity diagram in Fig. 2. The agent “owner1” checks periodically whether the vehicle is present. If the agent “owner1” receives no proper message from the agent “vehicle1”, it sends a message “inform” about the car theft to the agent “lc1”. The agent “patr1” periodically asks data from the agent “vehicle1” and received information is then sent to agent “lc1” for performing checks. Other agents are mainly implemented as ping-and-wait-behaviour-agents that wait for specific messages and then perform actions and send answers depending on the message received. All communication between agents was stored in a log file by the agent “logger”. An example of a log file is given here:

```
[15.03.03 12:08:22] thief1:->vehicle1:INFORM. The vehicle
is now stolen.
[15.03.03 12:24:30] owner1:->vehicle1:QUERY-IF. Checking
the vehicle.
[15.03.03 12:24:32] vehicle1:->owner1:FAILURE. The
vehicle is stolen.
[15.03.03 12:31:48] owner1:->lc1:INFORM. Vehicle theft:
123ABC, owner: First Owner.
[15.03.03 12:41:11] lc1:->cc1:INFORM. Vehicle theft:
123ABC, owner: First Owner.
[15.03.03 12:51:51] cc1:->poldb1:INFORM. Vehicle theft:
123ABC, owner: First Owner.
[15.03.03 12:52:17] cc1:->vrdb1:INFORM. Fax: Vehicle
theft: 123ABC, owner: First Owner.
[15.03.03 12:52:42] cc1:->bgdb1:INFORM. Fax: Vehicle
theft: 123ABC, owner: First Owner.
[15.03.03 15:12:02] thief1:->bgo1:REQUEST. Request for
border crossing: vehicle 123ABC, owner: First Owner.
[15.03.03 15:12:14] bgo1:->bgdb1:REQUEST. Check data:
123ABC, owner: First Owner.
[15.03.03 15:12:17] bgdb1:->bgo1:INFORM. The vehicle IS
STOLEN.
[15.03.03 15:12:20] bgo1:->thief1:REQUEST. You and the
vehicle are detained.
```

Since hard real time is not important in this case, the whole simulation is performed on a single processor (an ordinary office computer). In the current example, each agent acts in a rather trivial way according to its perception ability (e.g. as a reactive agent). The log file was used for analysing system performance and for illustrating modifications.

4.4. Analysis

In the current example the analysis of the system behaviour and suggestions for its modifications are based on use case and activity diagrams, the Q-model diagram and multi-agent simulation log files. Some duplication of processes and grounds for possible delays were detected during analysis (e.g. overall workload of the LC does not guarantee that vehicle theft is a prioritized task to be served as soon as possible; also the LC must be sure that the information is correct before transmitting it to CC; CC re-checks the information again in the vehicle register to be sure in the correctness of the information).

UML and the Q-model diagrams and multi-agent simulations were modified. For example, instead of the use case model given in Fig. 1, the system should be re-structured according to the use case model represented in Fig. 3. The modified use case model will also lead to simpler activity diagram as compared with the one in Fig. 2.

As a result, the goals that were set up in the initial analysis and the modelling task were generally fulfilled. The modelling clarified existing stages in information exchange and the efficiency of the information exchange methods. Some possibilities for improvement were found. The information system, information exchange, and some actions of the police (e.g. involvement of CC) were later modified to avoid duplication and to reduce the execution time of the planned work.

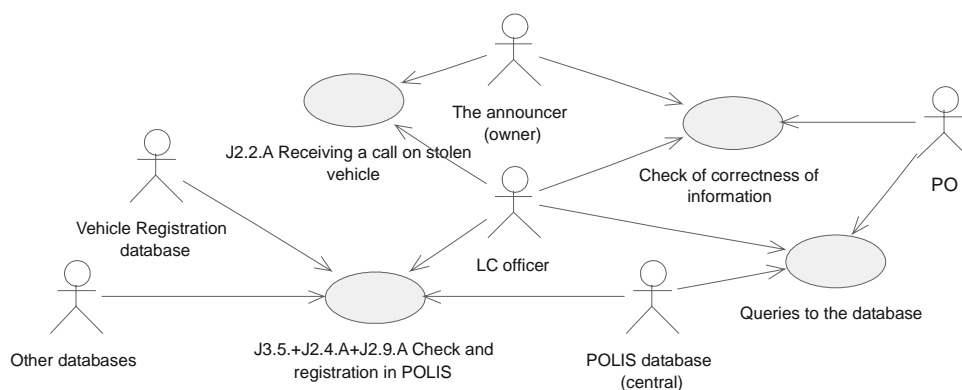


Fig. 3. The modified use case diagram of event registration and information exchange.

5. CONCLUSIONS

The experiments and obtained experience indicate that the introduced modelling methodology is in principle implementable. It actually helps to visualize the processes in the system and detect possible collisions (multiple requests for the same resources or different simultaneous tasks of an actor). The modelling methodology satisfies the requirements, formulated in Section 2: the UML and the Q-model permit specification of the process in detail including timing criteria. Agent-oriented approach supports visualization of interactions and leads to the actor-centred approach to the behaviour of an organization.

At the same time, the experiments demonstrate that the modelling methodology requires time for describing the processes. The model may become complicated and its processing time-consuming. Most sophisticated is to compose the timing model in the Q-model notation. Design of a corresponding multi-agent simulation depends very much on the information about the domain and on what aspect of the system performance should be modelled. It should also be noted that so far only very simple agent models have been used.

The proposed modelling methodology should be developed further in three aspects. First, it could be useful to create a software that dynamically filters the log file according to the wishes of the user, in this way simplifying the analysis. Second, the overall methodology and especially encapsulation of the dynamic behaviour of an organization during modifications should be described in greater detail. Also the use of an agent model for describing different characteristics of actors should be specified in greater detail. For supporting application of the methodology, the ultimate task is to elaborate a computerized tool that supports coordinated updating of different components of the model.

ACKNOWLEDGEMENT

This research was partially financed by the Estonian Science Foundation (grant No. 4860) and the Estonian Ministry of Science and Education (Research project No. 0142509s03).

REFERENCES

1. Kaplan, R. S. and Norton, D. P. *The Balanced Scorecard: From Strategy to Action*. Harward Business School Pr., Harward, 1996.
2. Boyne, G. External regulation and best value in local government. *Public Money & Manag.*, 2000, **20**, 7–12.
3. Bubenko, J. A. jr., Persson, A. and Stirna, J. *User Guide of the Knowledge Management Approach Using Enterprise Knowledge Patterns*. Deliverable D3, IST Programme Project HyperKnowledge–Hypermedia and Pattern Based Knowledge Management for Smart Organizations, project No. IST-2000-28401, Appendix B: EKD User Guide. Department of Computer and System Sciences, Royal Institute of Technology, Stockholm, 2001.

4. Motus, L. and Meriste, M. Towards self-organising time-sensitive control systems software. In *Proc. IFAC Conference on New Technologies in Computer Control*. Hong Kong, 2001, 236–241.
5. Wegner, P. and Eberbach, E. New models of computation. *Comput. J.*, 2004, **47**, 4–9.
6. Ferber, J. *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. Addison-Wesley, Harlow, London, 1999.
7. Booch, G., Rumbaugh, J. and Jacobson, I. *The Unified Modelling Language. User Guide*. Addison Wesley, Longman, Reading, Massachusetts, 1999.
8. Object Management Group. <http://www.omg.org>.
9. Motus, L. and Rodd, M. G. *Timing Analysis of Real-Time Software*. Elsevier, Pergamon, Oxford, 1994.
10. Selic, B. and Motus, L. Using models in real-time software design. *IEEE Control Syst. Mag.*, 1998, **12**, 677–699.
11. Edmonds, B. Modeling socially intelligent agents. *Appl. Artif. Intell.*, 1998, **12**, 677–699.
12. Dautenhahn, K. The art of designing socially intelligent agents: science, fiction, and the human loop. *Appl. Artif. Intell.*, 1998, **12**, 573–617.
13. Boudriga, N. and Obaidat, M. S. Intelligent agents on the WEB: a review. *IEEE Comput. Sci. Eng.*, 2004, **6**, 35–42.
14. Wooldridge, M. J. *An Introduction to Multi-Agent Systems*. J. Wiley, Chichester, West Sussex, England, 2002.
15. Prietula, M. J., Carley, K. M. and Gasser, L. *Simulating Organizations: Computational Models for Institutions and Groups*. AAAI Pr./MIT Pr., Massachusetts, 1998.
16. Wagner, T. A. *An Application Science for Multi-Agent Systems*. Kluwer, New York, 2004.
17. Jennings, N. R. Using intelligent agents to manage business processes. In *IEE Colloquium on Intelligent Agents and Their Applications*. IEE Digest No. 1996/101, 5/1–5/3.
18. Carley, K. M. and Prietula, M. J. Webbots, trust, and organizational science. In *Simulating Organizations: Computational Models of Institutions and Groups* (Prietula, M., Carley, K. and Gasser, L., eds.). AAAI Pr./MIT Pr., Menlo Park, California, 1998, 3–22.
19. Barber, K. S., Goel, A., Han, D., Kim, J., Liu, T. H., Martin, C. E. and McKay, R. Problem-solving frameworks for sensible agents in an electronic market. *Lecture Notes in Comput. Sci.*, 2004, **1611**, 470–479.
20. Bauer, B. and Müller, J. P. Using UML in the context of agent-oriented software engineering: state of the art. *Lecture Notes in Comput. Sci.*, 2002, **2935**, 1–24.
21. Bauer, B., Mueller, J. P. and Odell, J. An extension of UML by protocols of multiagent interaction. In *Proc. Fourth International Conference on Multi-Agent Systems*. Boston, Massachusetts, 2000, 207–214.
22. *Limits-PC User Manual, version 1.1*. Institute of Automation, Tallinn Technical University, Tallinn, 1998.
23. Savimaa, R. On modelling emerging behaviour of multifunctional non-profit organizations. In *Information Systems Development, Advances in Methodologies, Components, and Managements* (Kirikova, M. et al., eds.). Kluwer/Plenum, New York, 2002, 203–214.
24. Federation of Intelligent Physical Agents. <http://www.fipa.org>.
25. Bellifemine, F., Poggi, A. and Rimassa, G. JADE – a FIPA-compliant agent framework. In *Proc of 4th International Conference and Exhibition on the Practical Application of Intelligent Agents and Multi-Agents*. London, 1999, 97–108.
26. DeLoach, S. A. and Wood, M. Developing multiagent systems with agentTool. *Lecture Notes in Comput. Sci.*, 2001, **1986**, 46–60.
27. Nwana, H., Ndumu, D., Lee, L. and Collis, J. ZEUS: a tool-kit for building distributed multi-agent systems. *Appl. Artif. Intell. J.*, 1999, **13**, 129–186.
28. Savimaa, R. A methodology for modelling of modifications in multifunctional human organizations. In *Proc. 8th World Multi-conference on Systemics, Cybernetics and Informatics*. Orlando, Florida, 2004, **X**, 265–269.

Agenttehnoloogiate ja UML-i kasutamine organisatsiooni modelleerimiseks sõidukivarguse analüüsi näitel

Raul Savimaa

Artiklis on käsitletud ilmneva käitumise modelleerimise võimalusi multifunktsionaalsetes organisatsioonides. On esitatud kombineeritud meetodika organisatsiooni mudeli koostamiseks ja selle kasutamiseks tööprotsesside täiustamisel. Esitatud meetodika kasutab protsesside modelleerimiseks UML-i kasutusjuhte (*use case*) ja tegevusdiagramme (*activity diagrams*), Q-mudelit protsesside ajaliste parameetrite ja omavahelise järgnevuse täpsemaks analüüsiks ning multiagentlähenumist töötajate omavaheliste interaktsioonide kujutamiseks. Kuna organisatsiooni tegelik funktsioneerimine oleneb töötajate käitumisest, mida lisaks planeeritud tööprotsessidele kujundavad ka subjektiivsed tegurid (isiklikud eesmärgid, eelistused, koostöövalmidus ja suutlikkus), on realistlikuma tulemuse saamiseks oluline võimaluste piires modelleerida ka võtmeisikute võimalikku käitumist. Seetõttu vaadeldakse organisatsiooni multiagentsüsteemina. Pakutud meetodika rakendamise tulemusena valmib organisatsiooni mudel, mis võimaldab analüüsida protsesside koostoimet ja interaktsioonide tulemusi. Mudeli edasiarendus võimaldab testida organisatsiooni võimalikku käitumist protsesside muutmisel.

Kirjeldatud lähenemist on artiklis illustreeritud näitega infoedastuse protsessidest sõidukivarguse korral. Organisatsiooni käitumise eeltoodud aspektid ilmnevad üldjuhul kõige eredamalt dünaamilise, kindlate ajakriteeriumidega organisatsiooni (antud juhul politsei) meeskonnatöös: organisatsioonil on üheaegselt täita mitu erinevat ajakriitilist ülesannet, on olemas üldised ettemääratud standardprotseduurid situatsioonide lahendamiseks ning organisatsiooni tegelik käitumine sõltub piiratud ressursside oskuslikust kasutamisest ja sellest, kuidas otsustaja probleemi hindab.