Piotr Szwed*, Wojciech Chmiel*, Stanisław Jędrusik*, Piotr Kadłuczka*

Business Processes in a Distributed Surveillance System Integrated Through Workflow

1. Introduction

SIMPOZ (*System of Intelligent Monitoring of Objects and Areas of Special Importance*) project aims at building a highly configurable system for surveillance of public areas and objects of a special importance based on the analysis of digital images. Development of dedicated software supporting automation of surveillance tasks is not a trivial issue, due to the fact that events or conditions, which the system should detect may occur in multiple unpredictable variants [9]. The goal of the intelligent monitoring system is to alert an operator about potentially suspicious situations according to assumed safety criteria. An important element of the system is software that exploits recent results of research on "image understanding" [13]. It is planned to build a unified hardware-software platform that will allow to process and send video signal and to generate, handle and document alarms. Such system requires an efficient integration of components and flexible management of information obtained from supervised areas (originating from various types of detectors: video, audio and thermovision). The approach adopted in the SIMPOZ architecture assumes that integration tasks will be performed by a workflow management system. Its role is to integrate heterogeneous devices (such as cameras, microphones, mobile phones, computers, network and signalling devices) with external systems (databases, existing access control systems) into a homogeneous safety management system. Main premises for using a workflow-type system is a distributed nature and potential diversity of monitoring features, a large number of alarm signals fed by various sensors (external events), the need to ensure a flexible event flow management and tailored to needs emergency response processes. It should be stressed that application of workflow system is a novel approach to security management, not having analogues the existing systems in the world.

Workflow processes in SIMPOZ system are defined in the XPDL (*XML Process Definition Language*) language [4] recommended by the Workflow Management Coalition (WfMC). The XPDL is considered worldwide as a standard in the field of workflow specification.

Due to the large number of elements occurring in surveillance systems (both software and hardware), their variety and complicated dependencies. it was found necessary to document system functions, processes and architecture using an easy in perception language supporting direct communication with stakeholders (clients). The language is used for developing abstract specifications, most preferably, allowing an inclusion of salient system aspects already at the business analysis stage. The ArchiMate language [1] seems to be the most suitable for this purpose. ArchiMate allows creating descriptions (views) at the business, application and technological layer levels using a rich set of expressive constructs.

The paper presents examples of business models developed with ArchiMate language. They reflect the specificity and characteristics of surveillance systems: reactive character, event driven control, a large number of asynchronous data flows and a frequent requirement of bidirectional communication between detection modules and a system operator.
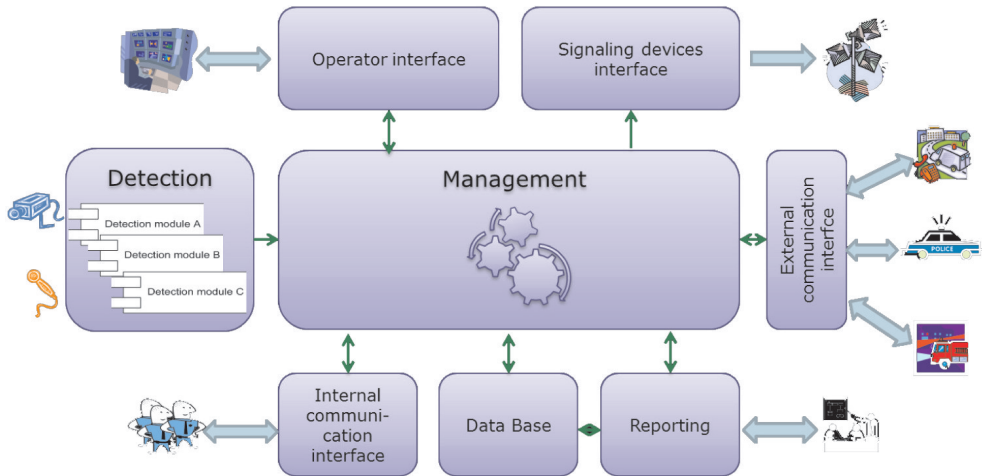

Fig. 1. The operational concept of the system

Figure 1 depicts the operational concept of the system and the links between its elements. In the center of the figure a management module is shown; it is responsible for communication and synchronization of all system components and supervision of currently running processes; these functionalities are provided by a workflow management system.

Detection modules perform surveillance tasks, i.e. they: detect potentially dangerous objects in the supervised areas, nonstandard (considered as dangerous) behavior and violen-

ce of restricted zones. They also extrapolate trajectories (speed and direction) of motion of observed objects, etc. Each of the detection modules have certain degree of freedom in communicating with the surveillance system using specialized interfaces designed in line with SOA (*Service Oriented Architecture*) approach. Information about an event detection triggers appropriate response procedures (specified in XPDL language) executed by a workflow engine.

Operator interface gives access to security related information, which comes from various types of detectors and running processes. In the case, when a suspicious situation is detected or an alarm generated, information about the event is passed to the operator. The operator can also communicate with intervention groups and other parties. In order to better assess the situation, he or she can access current and archive video streams.

Interface of signaling devices enable processes managed by workflow system to communicate with these devices and building automation systems (i.e. sirens, displays, emergency exit, illumination, floodgate, HVAC, etc.).

Interface of internal communication enables different intervention groups to communicate via various mobile devices. Its main functions are sending confirmation of the call reception, reporting current status of the intervention and communication with an operator and other actors.

External communication interfaces allow for sending automatic notification to emergency services (police, army, fire department) relevant to the detected danger. They also provide access to archival materials (video, intervention reports) to external systems.

The reporting interface provide access to the database of intervention reports and can be used to generate analytical reports on the basis of the history of events. It also provides entry points for processes that supervise reporting procedures (creation, modification, deletion).

The database stores recorded history of performed processes for later review. The adopted model assumes that types of events, time of their occurrence, decisions taken by executors and exchanged messages are recorded.

2. Review of surveillance systems

Advanced identification and recognition systems are developed mainly by foreign companies. The Bosch IVA (*Intelligent Video Analysis*) security system is a leader in this field [5]. Bosch IVA is a comprehensive solution designed for conducting intelligent video surveillance. Basic types of hardware used in the systems are IP cameras as well as VIP X1600 and VideoJet X servers. IVA supports all basic functions of intelligent surveillance. Moreover, it allows to define sophisticated surveillance scenarios. IVA also includes alarm transmission subsystem and centralized management. The software is running on PC class

computers. Received alert generates events that highlight a monitor, enlarge the image, automatically display an image preview of selected camera or run an appropriate procedure. Bosch VMS (*Video Management System*) provides complete surveillance, management of video signals and alarms handling. Alarms coming from IVA system are combined with general motion detection alarms. VMS system allows combining specific alarming conditions and ordering them according to their importance, resulting in possibly complex rules to manage emergency scenarios. An important system feature is, that alarms history is saved in a register, which facilitates finding the desired information in recorded video material. Bosch has developed its own system for searching events in the database called Forensic Search.

IndigoVision company [7] provides tools supporting integration and management of alarming systems. Its solutions allow centralized management of distributed monitoring systems through automated handling of alarms detected anywhere in supervised areas. The system allows for defining responsibilities for performing alarm responses and assigning them to registered users. The advantage of this solution is integration of alarms detected by many types of sensors. The system allows an integration of alarm zones (only one alarm is generated in a given area despite detection of events by many sensors), the localization of alarms, automatic forwarding and escalation of alarms, setting up emergency procedures integrated with illumination and automation and creating reports.

Another mature solution is VMS (*Video Management System*) system created by Mirasys Carbon company [10]. It is highly scalable and provides efficient analysis tools that allows an easy and intuitive handling of thousands of videocassette recorders and cameras. The system supports a centralized management of user profiles, constant monitoring of system status and generation of alarms associated with a hardware failure. An important feature of the system is the ability to define the procedures for handling registration and reporting of alarms. The analytical platform is open and can be integrated with any other video analysis software. Moreover, meta-data can be easily searched in databases containing historical data. Integration is based on an *open-gateway* solution. This allows combining already existing heterogeneous surveillance systems and integration with LDAP service.

In turn, Axis Company offers and promotes decentralized systems which perform essential calculations on cameras [3]. A key feature of this approach is free of charge, open standard to support network cameras (*Axis network cameras* – VAPIX). This allows not only to create own applications for intelligent video analysis, but also simplifies the development of a surveillance system, as the standard includes ready-to-use solutions. Camera Application Platform allows to develop client software, which then can be used with network cameras and video encoders from Axis Company.

System of Securiton company [11] is equipped with functions of location and geo-referenced positioning of objects and 3D technology. The company may deliver specialized

modules upon client request. IPS-Outdoor is a high quality video surveillance system for monitoring people and objects in outdoor areas. IPS-Indoor allows to track simultaneously up to 50 objects inside buildings. ISP-Subway, in turn, is dedicated for monitoring subways. It detects graffiti painting and loitering behavior. Apart from delivering locations of objects, it visualizes objects trajectories, also on large maps (geo-reference). This allows to easily track moving objects by different cameras in outdoor areas.

The Verint company [14] has developed Nextiva PSIM ™ Actionable Intelligence system that integrates a large number of detectors, introducing mechanisms, which simplify system management, increase the effectiveness of protection and optimize costs. The result of information processing is broadcast to users as well as other modules. The latter automatically identify dangerous situations. The system reduces time to initiate a response, provides tools for response management and effective cooperation with local security forces (police, fire department). Solutions provided by PSIM platform are scalable and are based on an open architecture. They include PSIM *scenario generator* that allows define scenarios launched in case of such events, as: explosion, flood, aggressive crowd behavior or gas leak. Execution of procedures is controlled through *control lists*. Visualization of the latter helps the staff to handle them. The system includes also built-in tools to supporting reporting. Other solutions are provided by companies such as Sanyo, GeoVision and Ganz.

Presented solutions have a long market history, they are technologically mature, however, they lack from fully process oriented approach. Hence, they do not allow for flexible modification of surveillance procedures without a deep system redesign. Available solutions are rarely based on the SOA paradigm, which recently was recognized as a leading approach in integration of heterogeneous and distributed systems.

3. Workflow systems and XPDL language

XPDL (*XML Process Definition Language*) is an XML based language for definition of workflow processes. Originally, it was designed as a format (standardized by *Workflow Management Coalition*) to store and exchange definitions of business processes between different modeling tools and/or workflow engines. However, because of its high expressiveness, XPDL is used in practice as a business process execution language.

The core element of XPDL language is activity. It represents a "working unit" that can be realized by a specified resource and/or external application. Activity can be executed automatically or manually. The undoubted advantage of the XPDL language, starting with the first version, is a flexible model of resources assigned to an activity. This allows for both static and dynamic allocation of resources. In addition, each activity can operate on data which are defined within a process or can refer to external resources.

The XPDL language distinguishes several types of activities. These are, inter alia: (i) *Route* activity which models decision, fork, join and synchronize nodes, (ii) *Subflow* activity which models subprocesses, (iii) *Block* activity which allows to treat a part of a process as a separate activity without creating a new subprocess, (iv) *Event* activity which models behaviors related to sending and receiving messages, (v) *No Implementation* activity which models behaviors not supported by a workflow system and (vi) *Task* activity which, generally, models collaboration with external applications/systems.

The specifications in XPDL language can be expanded by so-called *extended attributes* that can be assigned to almost any element. An extended attribute is defined as a key-value pair. Its semantics can be arbitrarily defined by software developers. In practice, extended attributes are used most often to store values of parameters specific to a given solution. However, they can be used to extend syntax and semantics of XPDL language without changing the language specification.

4. ArchiMate language

ArchiMate [1, 2] is a contemporary, open and independent language of enterprise architecture description. It includes three modeling layers: business, application and technology. The advantage of the ArchiMate is that it gives a comprehensive overview of the whole enterprise architecture. The business layer includes business processes and objects processed by them, functions, events, roles and services. The application layer includes components, interfaces, application services and data objects. Finally, the technology layer contains such elements as artifacts, nodes, software, devices, communication channels and networks. ArchiMate allows to present an architecture in the form of views which, depending on the needs, can include only items in one layer or can show vertical relations between layers (e.g.: the relationship between business activity and function of the component software).

ArchiMate language was built in opposition to UML [6], which can be seen as a collection of unrelated diagrams, and BPMN [4] which covers mainly behavioral aspect of enterprise architecture. The definition of a language has been accompanied by the assumption, that in order to build an expressive business model, it is necessary to use the relationships between completely different areas, starting from business motivation to business processes, services and infrastructure. ArchiMate goes beyond UML [8]: it defines a metamodel on the basis of which a user can create and illustrate the relationships between elements of different layers.

The subsequent sections present the views of the ArchiMate model limited to business layer. However, the decision of using this language in the project also includes documentation of lower layers: application and technology.

5. SIMPOZ system functions

The modeling approach applied here considers a function as a collection of processes. These processes can be flexibly combined through the events mechanism (e.g., after an operator approves an alarm, alarm identification process generates an event that initiates a response process; a response process triggers next event which is responsible for communication with external police services, etc.). The above mechanism of dynamic composition of processes is used for translation of descriptions of business processes to executable XPDL language.
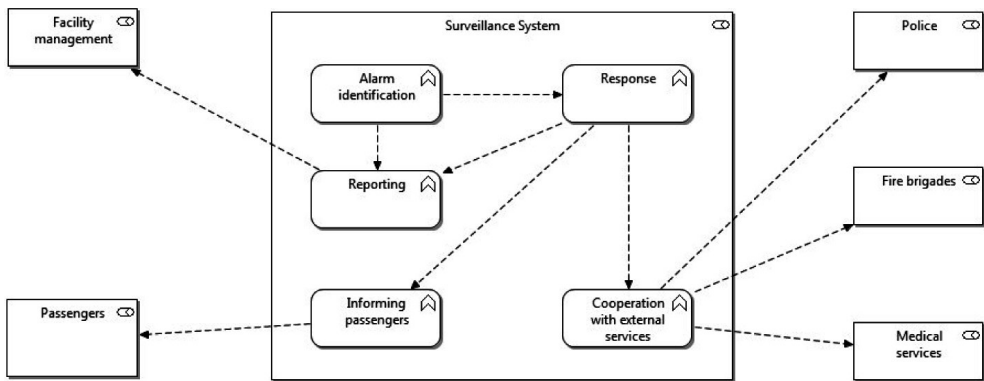

Fig. 2. SIMPOZ platform functions

Figure 2 presents the general overview of system functions. They include:
- *Alarm identification*: processes of events detection and classification, aiming at assessing that an alarm is founded or rejecting it.
- *Respons*e: processes in this group aim to coordinate the response to events.
- *Reporting*: notification of all alarms, creating reports and statistical summaries for subscribed parties.
- *Informing passengers*: broadcasting information about items left, managing evacuations and opening restricted areas to public use.
- *Cooperation with external services*: covers processes directing requests to police, medical services, fire department and processes providing access to digital evidence materials (video recordings).

Processes in functions are divided into three groups: *core processes* controlled by a workflow management system, *management processes* carried out using dedicated software and *support processes or services*.

The examples of processes in the *Response* function, divided into three groups, are summarized in Table 1.

Table 1

Groups of processes in the function *Response*

Core processes	R1. Takeover of an object violating a restricted zone R2. Response to identified bomb threat (detection of a suspicious stationary object in restricted area) R3. Response to dangerous behavior of a single person R4. Response to untypical/dangerous behavior of a group R5. Response to detection of moving object with colliding trajectory R6. Guarding tour
Management processes	RM1. Intervention group management (defining groups and their members) RM2. Defining schedules of tasks for groups RM3. Defining schedules of operators work (shifts) RM4. Defining schedules of guarding tours
Supporting processes	RS1. Storing reports and history of intervention in a repository RS1. Storing text and/or voice messages sent during intervention RS1. Storing recordings of operator's calls

6. Event-oriented model of the process

This section presents one of the processes in the *Response* function: *Takeover of an objects violating a restricted zone*. It was chosen as an example, because its flow includes communication patterns and shows more complex courses of interaction between two roles: *Operator* and *Intervention group*(s).

Specification of business processes within SIMPOZ project is prepared in two equivalent forms: narrative and graphical. Textual descriptions are made based on a template usually recommended for use cases. The template specifies goals, performers, a triggering event and pre- and post-conditions. In the case of the presented process, the goal is to "*coordinate the response to the event in order to take over the object violating a zone*". The process is triggered by an emergence of the event "*Alarm – zone violation*" generated in *Alarm identification* function. The postconditions are defined as "successful object takeover" ("failed object takeover") and require, that intervention history records are present in a repository.

The model of the *Takeover of an object violating a restricted zone* process is in fact a collection of subprocesses whose names (see Fig. 3) correspond to numbering in the scenario. The figures present only a partial specification. The considered process consists of nine subprocesses. Each of them starts and ends with an event. For example, the main

subprocess in Figure 3 starts after reception of external triggering event (*Alarm – zone violation*) and ends with one of the events: *R1 – Terminate* (internal event) or *To the police – take the detainee* (event passed to another function).
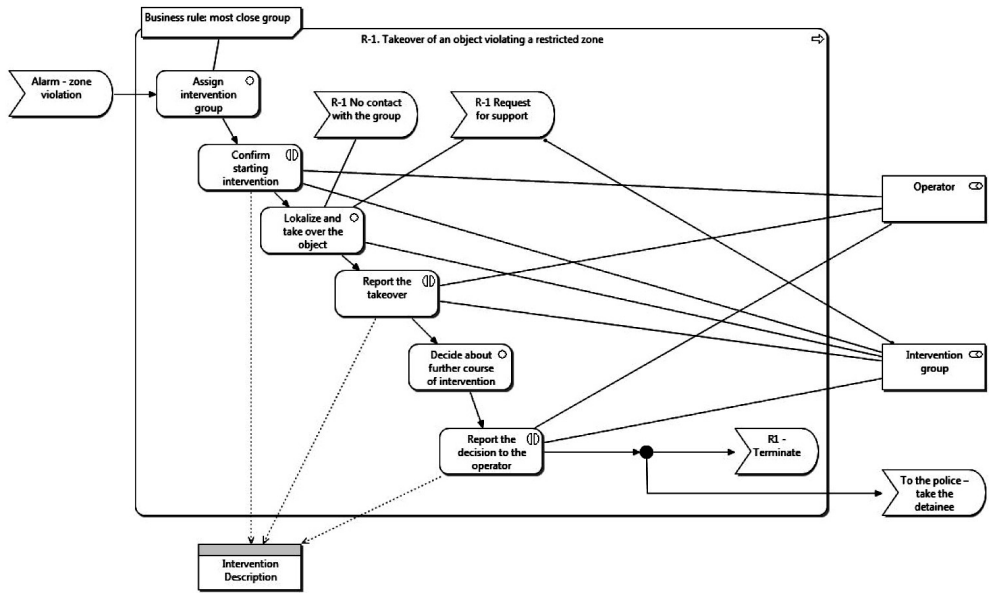

Fig. 3. Main flow of intervention process

In the adopted model (and the executive system), the *Terminate* event has a special semantics. It terminates all instances of the process and its child subprocesses and removes unhandled event from the system queues. The internal event *STOP* shown in Figure 4 terminates only a given subprocess.

The event *R1 – Request for support* can be generated and received during an occurrence of the activity *Localize and takeover the object*, what is depicted in Figure 4 by the appropriate association. As a result of its handling (Fig. 4), the request can be rejected, which ends the child process, the event requesting support can be generated or the main process can be recursively called through the generation of the event *Alarm – zone violation*.

Figure 5 shows the process of handling of the internal event *No contact with the group*. The *R1 – waiting time T1 elapsed* event is a typical construction for expressing time requirements. If the deadline is not met, the main process is called recursively.

The last of the sub-processes, shown in Figure 6, describes reactions to events that informs of movements of object and the intervention group, coming from tracking process in *Alarm identification* function (the presence of the intervention group in the supervised sectors changes the mode of operation of functions monitoring zone violation).
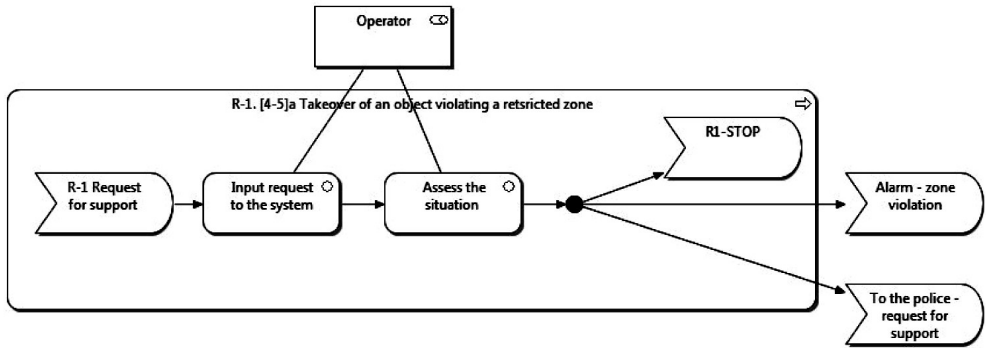
Fig. 4. Recursive call of the main process by sending the event *Alarm – area violation*
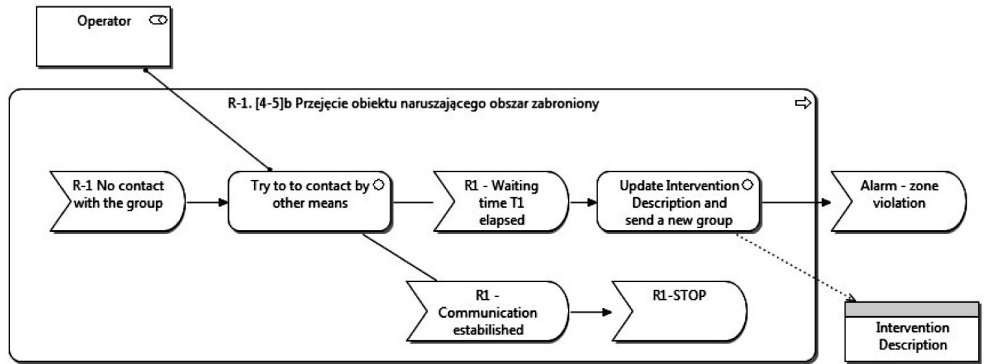

Fig. 5. Timeout detection (events generated as a result of not meeting deadline)
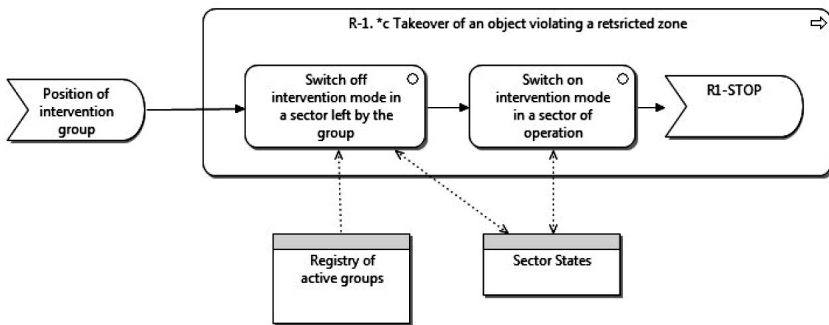

Fig. 6. Process controlling the changes of surveillance modes in the sectors
in which an intervention group is displacing

The remaining subprocesses include among others: handling of operator's decision (completion of the intervention, assignment of a support, calling external services) and re-

gistration of the course of the intervention. Several subprocesses are defined in order to react to special events such as disappearance of the object in the supervised area or remaining of the object in a restricted zone without possibility of taking it over (the process terminates when intervention is taken by police services).

It is worth noting, that all the events and decisions in the implemented system are registered. Registration of the decision is defined at the process specification level and registration (auditing) of events is a built-in function of the executive platform.

The presented examples demonstrate the complexity of the surveillance systems processes. On one hand, these processes presented in a decomposed form inspired by patterns from EPC [12] language seem relatively simple. On the other hand, it should be remarked that only selected subprocesses of a larger single process are presented. Their instances can be dynamically created by events generation, hence an integrated process resulting from their composition can be perceived as a very complex.

The ArchiMate language, in contrary to, e.g., BPMN, has no defined semantic. In spite of the smaller set of dedicated behavioral constructs, after adopting a certain convention, it can be used to express similar aspects of system behavior. Events *STOP*, *Terminate* or time events correspond in fact to BPMN events. Gathered experience indicates, that properly used constructs of the ArchiMate language allow obtaining high level of expressiveness. At the same time, the undoubted advantage of the language is the ability to combine elements of different architecture layers, business, application, and technical support, in one coherent model. This allows keeping consistency between the business and the application models and documenting implementation at the level of the technical infrastructure.

7. Conclusions

The main purpose of this paper is to present principles of modeling of business processes in distributed surveillance systems that were applied in the case of SIMPOZ system. The developed distributed surveillance platform is integrated through a workflow system which is an innovative element with respect to other solutions found on the market.

The experience acquired during the project realization justify the finding that the business modeling phase is essential for this class of systems and may strongly influence quality of project outcomes. It is important, however, to appropriately select tools, which in further stages will be used to describe various aspects of the analyzed system. Constructs offered by the ArchiMate language are sufficiently expressive to describe event orientation in workflow processes, assignment of performers to operations, manipulation of data, distinguishing types of components, documenting interfaces and system devices. These features proved particularly useful to define interfaces of modules and encoding processes using XPDL language.

The experience acquired from business analysis of SIMPOZ system may, in a long term perspective, result in development of tools supporting automatic translation of Archimate diagrams to XPDL language.

References

[1] Archimate 1.0 Specification, The Open Group, http://www.opengroup.org, 2009, pp. 122.

[2] Archi, Archimate Modelling Tool, http://archi.cetis.ac.uk/download.html, 2011.

[3] Axis (2012), Axis Comm. AB: http://www.axis.com/pl/corporate/contact.htm.

[4] Business Process Modeling Notation Specification. Version 1.2, January 2009, OMG Document 2009-01-03.

[5] Bosch (2012) *Intelligent Video Analysis* (IVA) www.boschsecurity.us.

[6] Fowler M., *UML Distilled*. 3rd ed. Addison-Wesley 2004.

[7] IndygoVision, www.indugovision.com.

[8] Malik N. (2009), *Will there be a battle between Archimate and the UML?*, http://blogs.msdn.com/b/nickmalik/archive/2009/04/17/will-there-be-a-battle-between-Archimate-and-theuml.aspx.

[9] Mikrut Z., Tadeusiewicz R., *Sieci neuronowe w przetwarzaniu i rozpoznawaniu obrazów*. [in:] Biocybernetyka i Inżynieria Biomedyczna 2000, t. 6, Sieci Neuronowe, AOW EXIT, Warszawa 2000, pp. 459–493.

[10] Mirasys Carbon VMS: http://www.mirasys.com/.

[11] Securiton AG, www.securiton.com.

[12] Scheer A.-W., ARIS – Business Process Modeling, Springer, 2000.

[13] Tadeusiewicz R., Ogiela M., Szczepaniak P., *Notes on a Linguistic Description as the Basis for Automatic Image Understanding*. Applied Mathematics and Computer Science 19(1), 2009, pp. 143–150.

[14] Verint Systems Inc., www.verint.com.