RESEARCH PAPERS OF WROCŁAW UNIVERSITY OF ECONOMICS No. 147
Advanced Information Technologies for Management – AITM 2010 2010

Stanisław Stanek, Mariusz Żytniewski
University of Economics in Katowice, Poland
e-mail: {stanek; zyto}@ue.katowice.pl

PROCESS APPROACH IN MULTI-AGENT SYSTEMS

Abstract: One of the challenges that developers have to face when building multi-agent systems involves the design of a system architecture that would support seamless coordination of the activities of multiple agents. Despite a rich variety of technologies, design methodologies and internal agent architectures, the goal- and task-orientation of the components remains a common feature of all multi-agent systems. Concerted actions performed by specific constituents can be, whatever architecture is adopted, regarded as a business process whose actors are people, information systems and software agents. The aim of the paper is to demonstrate the possibilities of applying the business approach in developing multi-agent systems, with a focus on employing the WADE system that represents an extension of the JADE platform.

Keywords: multi-agent systems, process approach, workflow, JADE, WADE.

1. Introduction

As Jeston and Nelis [2008] observe, "some of the recent literature in the process world has suggested that business processes are so important that the organization structure should be turned upside down to be a process-centric organization, rather than functionally based. It is argued that changing from the traditional functional, hierarchical orientation to a process-centric orientation will mean that our organizations will function with greater efficiency and effectiveness, to the benefit of management, staff, customers and all other stakeholders".

Process-centred approach assumes a horizontal perspective of organizational structures, where groups of structural elements are coordinated by managers of specific business processes who are responsible for the execution and supervision of these processes. Organizational design, nevertheless, still appears as a major challenge, given the fact that organizations operate in a highly complex environment where theoretical studies, placing an increasing emphasis on the need to strike a balance between processes and functions, differentiate a number of dimensions rather than a single important aspect to be monitored in change processes.

The operations of a process-oriented organization can be supported through the use of information technology tools which allow the design, implementation, monitoring

and simulation of business processes [Dayal et al. 2009]. Agent technologies may, in this context, constitute an element of the organizational information system where software agents support selected tasks within the framework of a specific business process. On the other hand, in the case of multi-agent systems, the process-centred approach comes to be applied in defining the system architecture.

This paper is dedicated to a range of issues concerning the process-centric approach and software agent technologies. It seeks to draw attention to the applicability of the process-oriented approach in designing multi-agent systems, particularly to the potential inherent in employing the WADE system which represents an extension of the JADE platform [Bellifemine et al. 2007]. The first part of the paper recapitulates the ongoing evolution of methodologies for deploying the process-oriented approach in organizations. Further, the article presents a selection of multi-agent system architectures and introduces some ideas for applying the process-focused approach to their development. In the final section, the WADE solution and the WOLF graphical editor are highlighted as sample applications of the process-centred approach.

2. Process-oriented approach in management science

The dynamics of changes in an organization's environment and the increasing competitiveness of the global market compel business companies to use their resources efficiently and effectively. Business management theory offers the process-centric approach founded on a systemic perspective of the organizational structure.

Within management science, the process-centred approach originates in relates to a shift in the way we perceive the organizational structure, i.e., a move away from a vertical, linear, functional perspective toward a horizontal, process-centric one. An organization is thus considered as a collection of clearly defined, repetitive and interleaving processes. The approach aims at enhancing the competitiveness of an enterprise in a rapidly changing environment by reducing its operating costs, improving the quality of its products and/or services, and at streamlining its operations by clearly isolating the stages within each process, monitoring these, and by eliminating any potential bottlenecks. The process-oriented approach supports effectiveness, largely helping an organization achieve its objectives. It has been noted that currently the third wave in the evolution of the process-oriented approach has begun [Bitkowska 2009].

The ISO 9000:2000 standard interprets a process as an assembly of correlated and interlinked activities by which inputs are transformed into outputs, while the process approach is described as systemic identification and management of processes applied within an organization, particularly the management of interactions between such processes.

A business process is characterized by:

– a clearly identified, measurable goal around which the process is centred,
– defined inputs and outputs,

– repetitiveness,
– measurable outcomes,
– the possibility of transcending the boundaries of specific organizational units/ divisions.

From the viewpoint of an organization's operations, it is necessary to define the owner of each process, who will be responsible to supervise it at each stage and make sure it produces the desired outcomes. The owner must understand and embrace the entire process and have the requisite competences to determine the responsibilities of respective organizational units involved in the process, which is critical to its effectiveness. Software solutions available in the market can now deliver a lot more than a graphical representation of a business process – they offer the capability of generating user interfaces which can support execution and simulation of processes. A sample solution of this kind is TIBCO Business Studio. An example of a business process simulation is shown in Figure 1.

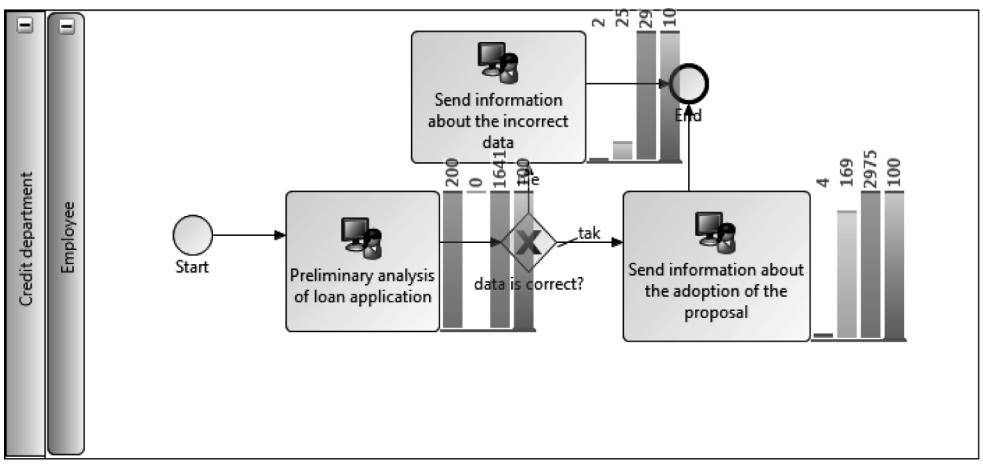

Figure 1. A sample process simulation using the BPMN notation

The recent advances in agent technologies and the continued development of support tools for systems integration make it possible to view software agents as a viable alternative in providing support to process-centred organizations.

3. Multi-agent systems

When it comes to multi-agent systems, a critical step that precedes the emergence of any specific solution is architecture design and modelling. As Luck et al. [2004] point out, it is indispensable to our ability to anticipate and explain the behaviors of the whole system as well as of specific agents. A large part of the system architecture

problem relates to a high level of abstraction [Lind 2000] needed to describe the structure of a multi-agent system.

Table 1 delivers a classification of multi-agent system architectures proposed by Dignum et al. [2001] and based on a distinction of market, network and hierarchical ones.

Table 1. The characteristics of multi-agent system architectures

Item	Market	Network	Hierarchy
Systemic goal	Exchange	Collaboration	Production
Agent autonomy	High	Medium	Low
Agent goals	Individual	Individual and global	Global
Relation forms	Negotiation	Negotiable within society norms and rules	Fixed; imposed by designer
Communication capabilities of agents	Interaction based on standards; communication concerns exchange only	Interaction and exchange procedures and can be negotiated	Specified on design
Interface to outside world	Open for agents (on identification)	Admittance procedures	Closed for agents, open for data

Source: based on [Dignum et al. 2001].

The first type distinguished in the above classification is thus the so-called market architecture. Under this architectural model, the system includes seller (supplier) agents alongside buyer (customer) units that are uniquely implemented within each specific solution. Agents are self-interested and each pursues its own goals only, communicating and negotiating with others in doing so. Such systems typically observe fixed, clearly defined communication standards and are open to any agents that would like to participate in the matchmaking and exchange process. A high level of agent autonomy is characteristic of multi-agent systems using this type of architecture and stems from the relationships among seller and buyer agents that, by definition, have contradicting or disparate goals. Openness, another conspicuous feature of the market architecture, implies a number of problems resulting from heterogeneity and incompatibility. The application of the process-centred approach within such architectures may hinge upon the adopted scheme for handling the matchmaking process – an agent performing the role of seller or buyer can, on its integration into a multi-agent system, handle all of the process as long as that process has been, through a relevant procedure, defined and aligned as a business process.

The second class in the typology based on [Dignum et al. 2001] is network architecture, where the following relationships are found:

- **Alliance** – this being the relation among agents, particular units cooperate over a period of time, sacrificing their individual goals for the attainment of a shared goal.
- **Coalition** – is a relation similar to alliance, except that agents pursue common goals without sacrificing their individual ones.
- **Team** – a relation that assumes the presence of a leader agent who is responsible for the appointment of agents to perform tasks whenever a new goal arises. In addition, there will be a control process which imposes restrictions on the autonomy of subordinate agents.

The network type of system architecture implies a number of problems that do not occur in market architectures. A major challenge relates to the notion of global/common goal, which makes it necessary to equip agents with mechanisms enabling them to not only investigate the environment but also to analyze other agents' goals in the context of their own. Consensus seeking is particularly difficult in network architectures relying on alliance, since agents operating in these must be able to ascertain when an activity should be halted or abandoned altogether for the sake of the system's primary goal. In such cases, the choice of a particular line of action may be determined through a prior analysis of available business processes and the selection of an agent that meets relevant criteria regarding the goal function.

The third class of architecture under the Dignum et al. [2001] typology is that of hierarchy. Hierarchical systems are characterized by full cooperation among agents. Such solutions are effective in reflecting the existing internal relationships in organizations and may be successfully used to control production processes. Owing to their staticity, hierarchy systems are closed, integral entities in which agents follow fixed, strictly defined goals. According to Dignum et al. [2001], three types of agents can be distinguished within the hierarchical model. These are as follows:

- **controllers** – which are responsible for control and monitoring of the overall system performance as well as of other system components; they represent the upper echelon in data processing mechanisms;
- **interface agents** – which will ensure interactive communication between the system and the user; and
- **worker agents** – which perform specific tasks within the system and should therefore be incorporated as the third type.

The architectures described above, determining the functional characteristics of agents, can be supported by well-designed business processes that guide the activities of agents through their goals. A business process can be treated as a mode of cooperation between agents or a source of data they require to be able to perform actions. This approach chiefly evolves toward its linkage with the Business Process Management concept where a multi-agent system deployed in a process-focused environment will become either a part or the controlling element of a larger process.

4. Software agents and their role in the development of the process approach

In terms of technology, it is network (Web) services or, in a broader context, service-oriented architectures (SOA) which contribute the most to the development of the process approach. There are a number of advantages to using SOA: it supports access to and distribution of network services, integration by defining communication standards, transparency, and interoperability.

The designates of the term may comprise [Rishi et al. 2008]:

- **SO Enterprise**: A Service-Oriented Enterprise (SOE) represents a new approach to defining software architecture models and business infrastructures. It permits an organization to view its activities from the perspective of a customer, supplier or collaborator. As an added value, the approach results in reduced operating costs and easier, flexible adaptation to changes in the environment.
- **SO Architecture:** A Service-Oriented Architecture constitutes, to a large extent, a set of services which cooperate within a business process. Communication among them may involve exchanging complex data types defined under the XML standard.
- **SO Computing:** Service-Oriented Computing delivers support solutions for the development of new architectures focused on autonomy and heterogeneity.

A single Web service, each coming with a set of protocols through which it can be published and located, represents the central element of the SOA concept. All such services are thus carried out to closely knit standards. The SOA is a body of frameworks, practices and conceptions prescribing the ways in which Web services should be created and used.

Solutions employing Web services will often find applications in technologies tailored to support business processes, as they provide for easier integration of services within and across different processes. Furthermore, they can facilitate the integration of software agents and multi-agent systems within business processes.

The advantage of using software agents rather than standard Web services is that the former have a proactive character and can utilize network services as data sources in performing their goals. Combining software agents with Web services alters the passive nature of the latter by making it capable of triggering activity within the system.

Poggi et al. [2007] argue that a typical application of SOA with agent technologies includes such actors as service providers, business process managers, and users, each of them performing roles defined within the multi-agent system. Service providers will usually tackle registering available network services in a predefined service catalogue, while business process managers are, within this kind of architecture, responsible for managing the services. This is particularly the case of service orchestration, where all Web services are managed by a central component, whereas

under service choreography each service has fragmentary knowledge on subsequent actions. The distinction is illustrated in Figure 2.

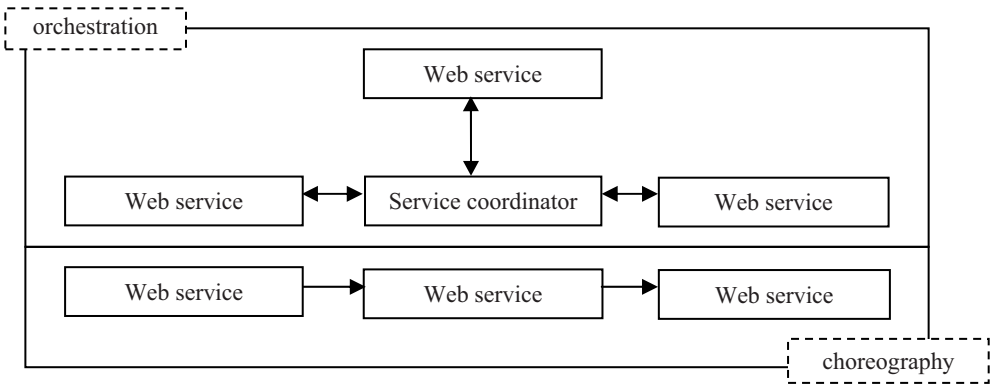

Figure 2. The distinction between orchestration and choreography

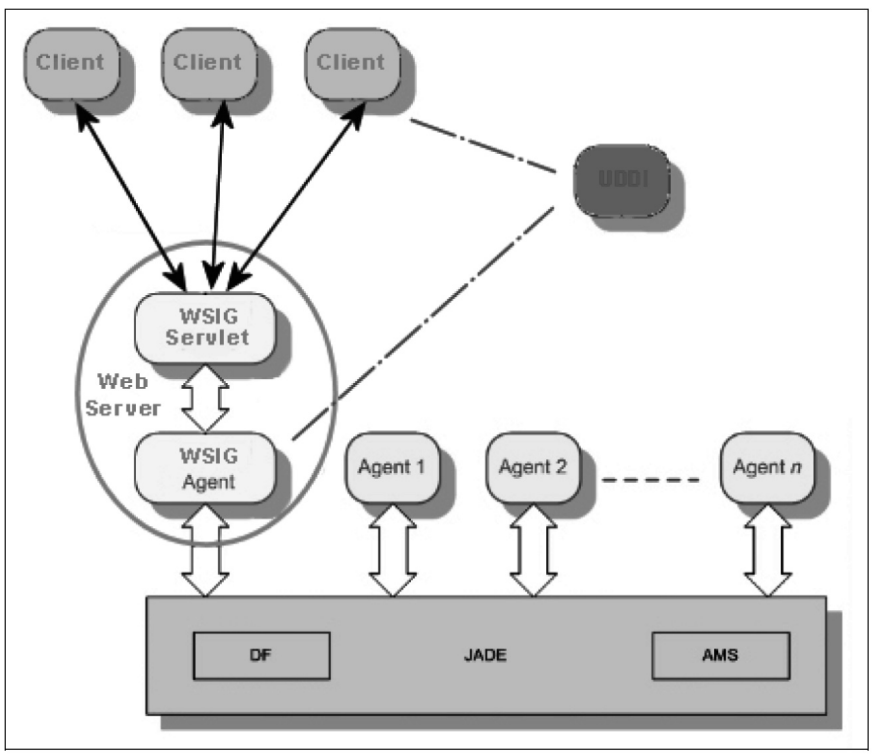

Figure 3. A sample application of Web services to extend the functionality of a multi-agent platform

Source: http://jade.tilab.com/doc/tutorials/WSIG_Guide.pdf.

Current solutions designed to support business processes in organizations have centrally managed architectures that make it far from easy and obvious to adapt the existing processes to changes in the environment. Deploying multi-agent systems into such solutions may substantially increase their adaptability [Bastin et al. 2005], especially if it is accompanied by the incorporation of semantics which follow the semantic trend of evolution envisaged for the Internet. The application of software agents can, under these circumstances, lead to automating the discovery and use of new Web services which are complementary to those currently in use. This, in turn, can make it possible to access the same resources and services which are used by the original business process, yet at a lower cost.

In the context of the process approach, the evolution of agent technologies follows either of two strands: on the one hand, software agents may be part of the business process, which entails the use of an interface to support the inclusion of a multi-agent system into the business process. An example of this approach, i.e. extending a multi-agent system architecture through the availability of Web services, is WSIG for the JADE platform [Greenwood 2005]. This idea is depicted in Figure 3.

Incorporating this sort of a solution into a multi-agent platform provides a possibility to quickly convert ACL messages to the WSDL format and backwards, which largely facilitates the integration of a multi-agent system with Web services. The definition of standardized interfaces through which the multi-agent system communicates with its environment makes it easier to orchestrate such solutions with existing business processes.

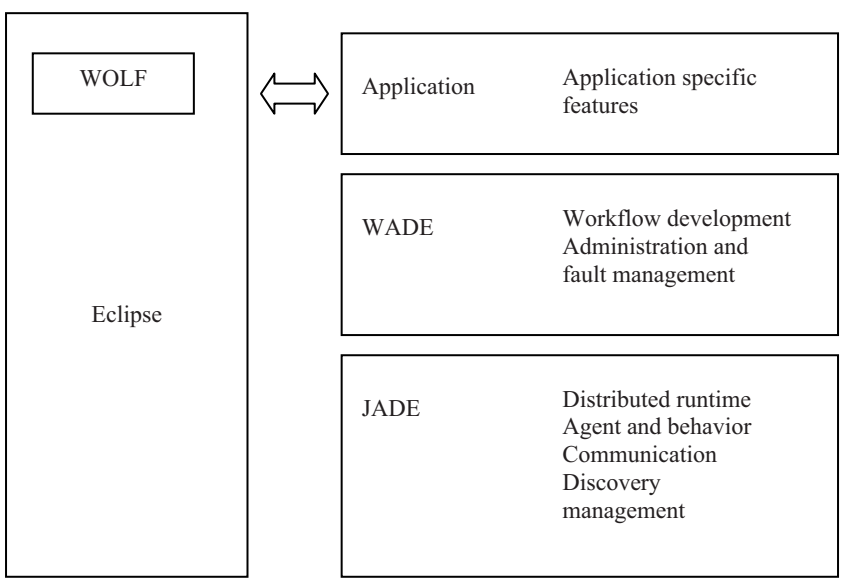

Figure 4. WADE platform architecture

Source: Caire et al. [2008].

The other strand of the development is to mould the multi-agent system architecture toward the process approach, as embodied by e.g. the WADE (Workflows and Agent Development Environment) project [Caire et al. 2008], which strives to support the design of multi-agent system architectures by supplying the necessary functionality to define specific agents' workflows and manage the whole process. The WADE platform includes, e.g., WOLF – a plug-in application for the Eclipse platform, which supports the design and development of multi-agent systems (Figure 4).

Applying the process-centric approach to define the functional characteristics of a multi-agent platform implies that there will be support for specifying the tasks to be performed by particular actors in the process, as well as the relationships between them and the criteria which trigger specific actions. It should also guarantee adequate supervision throughout the business process.

A central problem of multi-agent system design is, regardless of system architecture, the diversity of design methodologies and agent architectures. Where the process approach is applied, emphasis is placed on the structure of the process itself, which is supposed to yield to analysis not only by programmers but also by business analysts and domain experts who can thus easily examine it and recommend modifications. Graphical representations of the process enhance its understanding by both the actors, programmers or analysts. Breaking the process down into clearly defined tasks facilitates code generation and makes it possible to use a transaction type of relationship throughout the process as well as at its specific stages, improving security in case of execution errors. The process-centred approach also helps ensure proper monitoring of the agent system performance by enforcing a detailed specification of all possible states of the system along with corrective measures to be taken in case of failure.

The current version of WOLF (2.6) allows defining such diagram elements as:

1. **Code** – to define events executed as JAVA code.
2. **Tool** – to launch any software required.
3. **WebService** – to use Web services within a business process.
4. **Route** – to be used as an additional object; it does not support data processing but offers graphical representation of additional tasks.
5. **Subflow** – to execute another business process.
6. **If** – to define logical gateways.

An object file so defined is not as expressive as one using the BPMN notation, yet it allows designing agent-supported business processes. A sample process definition is shown in Figure 5.

Along with the diagram, some of the JAVA code is generated to facilitate subsequent execution of the underlying business process (see Figure 6).

The WADE currently does not support any of the BPM notations or, consequently, any of the related technology solutions. For its operation, it relies on the JADE platform and the WorkflowEngineAgent library which provides an extension of

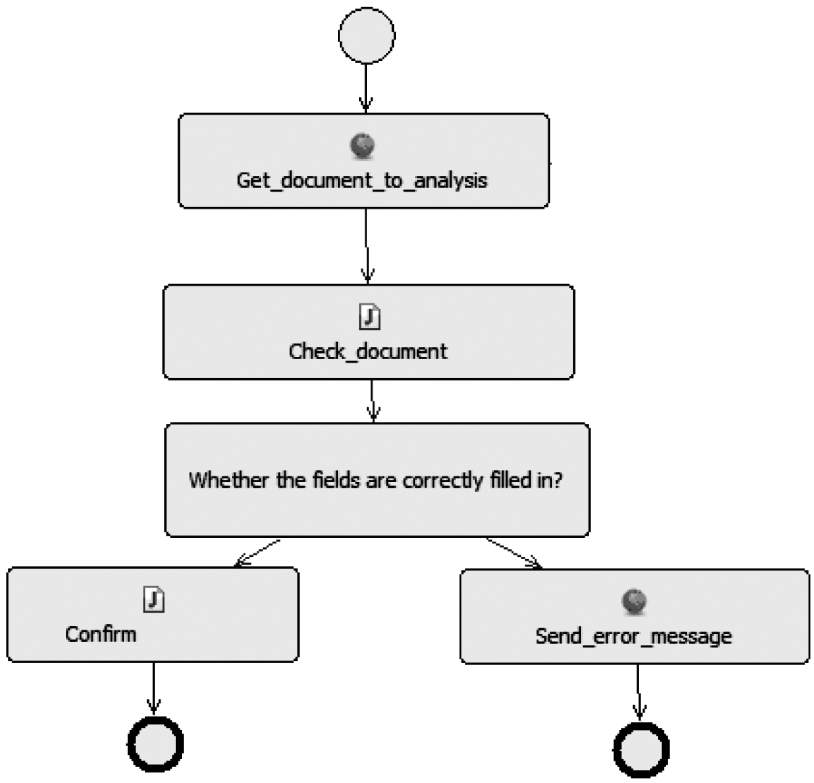

Figure 5. Sample WOLF notation for a business process

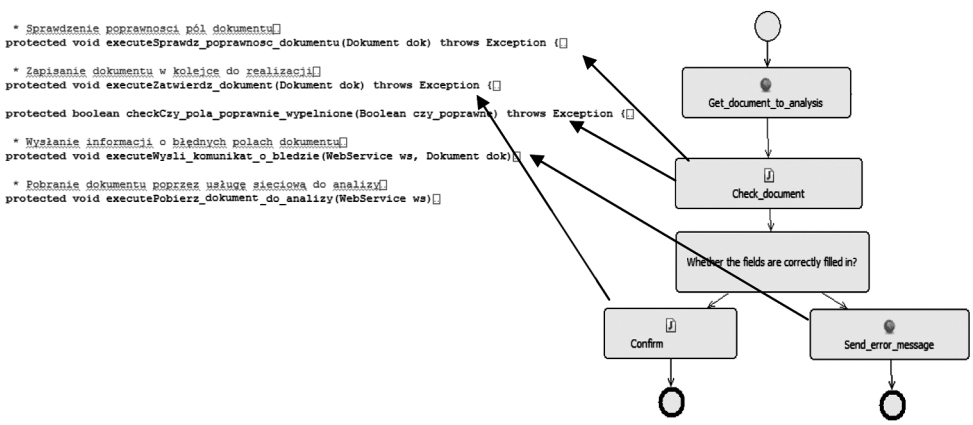

Figure 6. Code generation for a business process using WOLF technology

the Agent class. As a result, the JAVA language is used to define the structure of the business process. Graphical representation of the model is based on the XPDL language.

5. Summary

By way of summary, it can be pointed out that the application of the process-centric approach in developing a multi-agent system entails a broader outlook on the system architecture in the context of its environment. A process-oriented multi-agent system must be highly deterministic. Concepts discussed in the paper demonstrate that the process-centric approach can be used in developing multi-agent systems irrespective of the multi-agent platform and the design methodology applied. A multi-agent system can be treated as an element of a business process, which either becomes an actor within the process or, owing to some internal logic, controls and oversees the process. This approach leads to the creation of a proactive network service that can vitally extend the functionality of a business process. The application of agent technologies and relevant ontology concepts can also support the cataloguing and propagation of network services.

On the other hand, the process approach can help define the multi-agent system architecture by enforcing a logic of agent collaboration. If this is the case, specific agents become actors within a business process.

References

Bastin T., Savarimuthu R., Purvis M., Purvis M., Cranefield S. (2005), Agent-based integration of Web services with workflow management systems, [in:] *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2005*, portal.acm.org.

Bellifemine F., Caire G., Greenwood D. (2007), *Developing Multi-Agent Systems with JADE*, John Wiley & Sons, New York.

Bitkowska A. (2009), *Zarządzanie procesami biznesowymi w przedsiębiorstwie*, Vizja Press & IT, Warszawa.

Caire G., Gotta D., Banzi M. (2008), WADE: A software platform to develop mission critical applications exploiting agents and workflows, [in:] *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2008*, portal.acm.org.

Dayal U., Eder J., Koeheler J., Reijers H. (2009), *Business Process Management*, Springer-Verlag, Berlin–Heidelberg.

Dignum V., Weigand H., Xu L. (2001), Agent societies: Towards framework-based design, [in:] *Proceedings of the 2nd Workshop on Agent-Oriented Software Engineering, Autonomous Agents, Montreal, May 2001*. A reviewed version of this paper appears in: *Agent-Oriented Software Engineering II*, Eds. M. Wooldridge, G. Weiss, P. Ciancarini, Lecture Notes in Computer Science, Vol. 2222, Springer-Verlag, 2002.

Greenwood D. (2005), JADE Web Service Integration Gateway (WSIG), [in:] *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2005*, portal.acm.org.

Jeston J., Nelis J. (2008), *Management by Process: A Roadmap to Sustainable Business Process Management*, Elsevier, London.

Lind J. (2000), Issues in agent-oriented software engineering, [in:] *Agent-Oriented Software Engineering*, Ed. P. Ciancarini, First International Workshop, AOSE, Limerick, Ireland, pp. 45-58.

Luck M., Ashri R., d'Inverno M. (2004), *Agent-Based Software Development (Agent-Oriented Systems)*, Artech House Publishers, Boston–London.

Poggi A., Tomaiuolo M., Turci P. (2007), An Agent-based Service Oriented Architecture, [in:] Proceedings of the Eighth Workshop from Objects to Agents (WOA'07), *Dagli Oggetti agli Agenti. 8th AI*IA/TABOO Joint Workshop "From Objects to Agents": Agents and Industry: Technological Applications of Software Agents, 24-25 September 2007, Genova, Italy*, http://www.informatik.uni-trier.de/~ley/db/conf.

Rishi O.P., Sharma A., Bhatnagar A., Gupta A. (2008), Service oriented architecture for business dynamics: An agent-based approach, [in:] *6th International Conference on E-Governance ICEG, 2008*, http://www.iceg.net/2008/.

PODEJŚCIE PROCESOWE DO SYSTEMÓW WIELOAGENTOWYCH

Streszczenie: W przypadku budowy systemów wieloagentowych jednym z problemów, z którym muszą się zmierzyć jego twórcy, jest odpowiednie zaplanowanie architektury systemu, tak by uwzględnić potrzebę koordynowania działań poszczególnych agentów. Mimo różnorodności w sferze metodyk projektowych, technologii, architektur wewnętrznych agentów, elementami łączącymi dowolny system wieloagentowy są celowość oraz zadaniowość działania poszczególnych jednostek. Skoordynowane działania realizowane przez poszczególne jednostki, niezależnie od przyjętej architektury, mogą być rozpatrywane jako proces biznesowy, którego aktorami są ludzie, systemy informatyczne oraz agenci programowi. Celem niniejszej pracy jest prezentacja możliwości zastosowania podejścia procesowego w budowie systemów wieloagentowych, w szczególności z wykorzystaniem systemu WADE będącego rozszerzeniem platformy JADE.