Intelligent Agents and Multi-Agent Systems

AGENT ORIENTATION AS A TOOLBOX FOR ORGANIZATIONAL MODELING AND PERFORMANCE IMPROVEMENT

Jacek Jakieła, Bartosz Pomianek

Abstract: *The main goal of the paper is to convince that agent orientation may be considered as a powerful paradigm for organization modeling and the reference architecture for Management Information Systems, that if properly applied, would lead to firm's overall performance improvement. The scenario of proving these theses consists of three steps. Firstly the basic concepts of agency have been defined. Then the agent and multi-agent system are presented as playing the roles of very natural and intuitive modeling constructs and complexity management tools. Finally the benefits of agent oriented software application are described from the perspective of gaining competitive advantage by improving intra- and interorganizational efficiency due to basic characteristics of agents such as constant monitoring of the environment (where they are situated), reactivity, pro-activeness and social ability.*

Keywords*: Software Agent, Intelligent Agent, Agent Oriented Approach, Organization Modeling, Multi-agent System, Complexity Management, Business Performance Improvement, Supply Chain, Demand Chain, Management Information Management System*

ACM Classification Keywords*: I. Computing Methodologies; I.2 Artificial IOntelligence I.2.11 Distributed Artificial Intelligence; Multiagent systems*

Introduction

Business organizations today are becoming increasingly complex systems. In order to manage this emerging complexity proper toolbox is needed. The paper presents two aspects of agent oriented toolbox. The first one considers support for organization modeling which is currently crucial to improve business processes performance as well as ICT development effort. As North et al. suggest, nowadays organizations are facing several problems which are mainly related to fragmented consumer markets, more interwoven industrial supply chains, sophisticated transportation systems and growing interdependency of infrastructures [North, 2003]. Better understanding of how these would affect specific organization requires the business model that may be analyzed from different angles and on several levels of abstraction. The model development process is conducted with the use of modeling language that must have modeling constructs which enable to fully express the characteristics of business problems. As there is shown in the paper, agent orientation is a new way to capture and analyze the structure and behavior of enterprises. Multi-agent system as a modeling metaphor can be considered very natural and intuitive paradigm for building business models.

The second aspect of the toolbox regards the agent based software support for process and knowledge oriented business organizations. As paper implies, performance improvement efforts should take into consideration business computing paradigm shift – from "direct manipulation metaphor" to "do not navigate – delegate!" way of using software systems. In such scenario, the societies of software agents work autonomously on behalf of business actors, who delegated to them sophisticated tasks, helping organization to effectively achieve its goals.

Basic concepts of agent orientation

Before advantages of agent oriented organization modeling and performance improvement will be presented, it seems advisable to explain the essence of agency.

The agent

Over the last two decades the concept of an intelligent agent has become really popular. A number of researchers dealing with artificial intelligence focused on agency. Consequently numerous definitions of an agent have been coined. Two of them have been mentioned below.

Michael Wooldridge and Nicholas R. Jennings describe an agent as: "a hardware or (more usually) software-based computer system that enjoys the following properties:

- autonomy: agents operate without the direct intervention of humans or others, and have some kind of control over their actions and internal state;
- social ability: agents interact with other agents (and possibly humans) via some kind of agent-communication language;
- reactivity: agents perceive their environment, (which may be the physical world, a user via a graphical user interface, a collection of other agents, the INTERNET, or perhaps all of these combined), and respond in a timely fashion to changes that occur in it;
- pro-activeness: agents do not simply act in response to their environment, they are able to exhibit goal-directed behavior by taking the initiative [Woolbridge, 1995].

Another definition has been proposed by S. Franklin and A. Graesser in their paper attempting to distinguish software agents from regular computer programs [Franklin, 1996]: An autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future.

Basing on this definition few vital attributes of an agent can be noticed:

- an agent exists in a certain environment and thus it ceases to be an agent when extracted from such environment,
- an agent senses its environment, acts on this environment and its actions can affect what an agent will sense in the future,
- an agent operates over time and acts whenever it "feels" it's necessary; unlike regular program which terminates once its mission is accomplished,
- an agent operates autonomously pursuing its own goals.

All these basic characteristics constitute conceptual framework that will be used later when trying to show how agent oriented applications may lead to organization performance improvement.

Multi-agent systems

A Multi-Agent System may be defined as a set (society) of decentralized software components (where every component exhibits the properties of an agent, mentioned in the previous section), that are carrying out tasks collaboratively (often in parallel manner) in order to achieve a goal of the whole society. Later in the paper this definition has been disaggregated and all the properties are used to show why the multi-agent system can be considered as a very intuitive organization modeling metaphor.

As presented definition reveals, software agents have the ability to collaborate with each other which enables the creation of multi-agent systems. Collaboration is defined as a process in which society coordinate its actions in order to achieve common goals. Software agents are able to collaborate with one another as well as human agents.

The corner-stones of inter-agent collaboration are: communication and knowledge sharing. Communication is basically an exchange of information among agents (agents can send messages to each other, observe each other's state and behavior, however, communication takes place on the so called knowledge level). To enable knowledge sharing agents must have common goals and decompose the process of achieving these goals into sequence of actions providing that every agent is capable of performing task assigned to it.

Inter-agent collaboration requires also a communication language. Currently the most popular agent communication languages are: Knowledge Query Manipulation Language (KQML) developed in early 90's and FIPA-ACL developed by Foundation for Intelligent Physical Agents. Both rely on speech act theory and define a set of performatives, their meaning and protocol for perfomatives exchange.

To enable agents to understand each other they must not only speak the same language, but also have a common ontology. According to Gruber "…An ontology is an explicit specification of a conceptualization… A body of formally represented knowledge is based on conceptualization: the objects, concepts, and other entities that are assumed to exist in some area of interest and the relationships that hold among them" [Gruber, 1993].

Within an agent community agents have different knowledge, capabilities and responsibilities. Each one can have access to different resources and can perceive certain matters differently. Developing a successful multi-agent system demands effective coordination of autonomous agents thus creating a synergy enabling this system to solve problems in dynamic environment despite imperfect data and information. To achieve it one must answer few critical questions [Farhoodi, 1993]:

- What are appropriate agent architectures for different classes of problem-solving?
- How does an agent acquire its knowledge and how should it be represented?
- How does a complex task get decomposed and allocated to different agents?
- How should agents cooperate and communicate with each other?
- Can an intelligent agent be trusted?

Agent orientation as a modeling paradigm

When considering agent orientation as a modeling paradigm it is essential to answer two fundamental questions. The first one is why agent and multi-agent system characteristics make agents so natural and intuitive organizational modeling constructs? The second asks why agent orientation is optimal choice for complexity management? The following sections answer them one by one.

Agent as a modeling metaphor

One of the fundamental assumptions for this article is that organization modeling process bases on multi-agent system metaphor, that leads to perceiving and understanding of organization in the way typical for multi-agent system software engineering, but also takes under consideration business aspects along with basic organization characteristics.

The use of a metaphor during the process of organizational analysis and understanding is of great significance. Morgan says that metaphor is frequently understood as a way to make an argument more appealing, however its meaning is much more important. The use of a metaphor is a consequence of a way one thinks and perceives which penetrate our understanding of the world that surrounds us [Morgan, 2005].

We use metaphor when we try to understand some portion of reality using another its portion. Thus we formulate a theorem that "A is B" or that "A is similar to B". Concerning this article an assumption has been made that in qualitative terms the characteristics of modern organizations are really close to the characteristics of multi-agent systems both in structure and in behavior.

The appearance of obvious similarities between multi-agent systems and business organizations is particularly visible in the case of companies which adopted such business strategies as decentralization of operations and process orientation, that led to the creation of specific organizational forms [Drucker, 2000].

The table 1 shows observed similarities. In the first column are basic structural and behavioral characteristics of multi-agent systems, and in the second characteristics of modern organizations extracted from key titles which deal with organization design and process orientation.

Table 1. Similarities between structural and behavioral characteristics of modern organizations and multi-agent systems [Jakieła, 2006].

Multi-agent system	Business organization
Multi-agent system is a set of **decentralized** software components.	Centralized model and functional decomposition reflect precisely decomposition of workload into smaller tasks, that are distributed among particular company departments and are accomplished in sequences [Peppard,1997]. Modern organizations operate according to decentralized business process patterns which are accomplished by distributed organizational actors. Decentralization causes, that these processes are moved from companies headquarters to local offices. The operational model of modern companies is a highly decentralized.
Multi-agent system is a set of **autonomous** software components.	Decentralization requires in turn autonomy delegation, that has drastically changed the role of organizational actors, because “controlled positions” have been replaced by positions which give full competence [Drucker, 2000]. This trend is really similar to that which takes place in software engineering due to control encapsulation into distributed components, which therefore poses operational autonomy. According to Champy and Hammer people working within processes that are being reengineered must be equipped with delegations. As members of a process team they are both allowed and obliged to make decisions. [Hammer, 1996]. In case of process orientation it is impossible to avoid situation when organizational actors, who perform process oriented jobs, are fully autonomous entities.
Multi-agent system is a set of **goal-oriented** software components.	In modern organizations functional departments have been replaced by process teams. A set of organizational actors, who cooperate in order to achieve particular goals of certain process is a natural form of modern firm’s organization [Hammer, 1999].
Multi-agent system is a set of software components, which may carry out tasks in **parallel manner**.	In the company organized around processes, subsequent work-stages are performed in natural order. Instead of artificial operations order, natural operation order is used. Processes’ de-linearization allows task performance acceleration due to two factors. Firstly, lots of tasks are performed in the same time. Secondly, shrinking of time between initial and final stages of processes causes that the necessity for serious changes, which can undermine or make incoherent work done so far is less probable [Hammer, 1996].

Agent orientation and complexity management

Thank to the use of an agent as a modeling construct designer/analyst can cope better with problem domain and design process complexity.

Complexity management problem is a really vital issue, because each human is in this field inherently limited. When analyst/designer sees the problem domain, he is trying to solve, for the first time, he sees vast variety of components, which interact with one another in many various ways. Trying to structuralize the model during the analysis she is forced to take under consideration plenty of factors. Unfortunately, according to research conducted by cognitive scientists, each human being can work effectively with 7±2 portions of information at the same time [Miller, 1956]. Simon claims that the velocity of information processing by the human, which totals approximately 5 sec for the perception of each new portion of information, is an additional restriction [Simon, 1996].

Complexity management is also a serious problem in case of organization modeling. Each organization is undoubtedly a complex system. According to Carley "Organizations are heterogeneous, complex, dynamic nonlinear adaptive and evolving systems. Organizational action results from interactions among adaptive systems (both human and artificial), emergent structuration in response to non-linear processes, and detailed interactions among hundreds of factors." [Carley, 1999].

Booch [Booch, 2007] relying on Simon's work [Simon, 1996] has defined the following set of basic characteristics of complex systems:

Complexity frequently takes a form of hierarchy, where the system is composed of sub-systems connected with each other, which have their sub-systems, which in turn have their sub-systems and so on until the elementary level is reached. This hierarchy does not mean the superior-subordinate relation. Thank to the fact, that complex systems are nearly decomposable we can fully understand them, describe or even perceive. Simon claims that it is highly probable that in reality only the systems that have a hierarchical structure can be understood [Simon, 1996]. Looking at process oriented organizations from this perspective, it is possible to distinguish such levels of hierarchy as organization actors level, business process level, singular organization level and specific configuration of few organizations in a form of extended enterprise or virtual organization (See Fig. 1.)

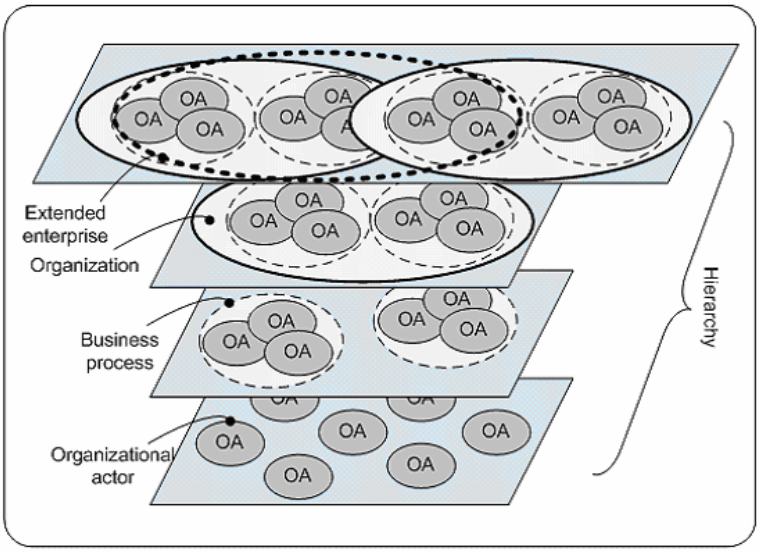

Figure 1. Business organization as a complex system

The choice which components of a system should be treated as elementary is arbitrary and depends on the system observer's decision.

It is possible to identify interactions taking place between sub-systems as well as inside sub-systems between their components, however interactions of the second type have one row higher frequency and are more predictable. The interaction frequency will differ depending on the level of hierarchy. For example, within a business organization more interactions will take place between employees working on the same process than between teams of employees working on different processes. The differences in interaction frequency within and between sub-systems allow decomposition and lead to the clear division between domains of analysis. In case of social systems, and undoubtedly every organization can be seen as such system, nearly decomposable character is clearly visible, therefore it is possible to exploit advantages of the decomposition method.

Complex systems are mostly sets of similar elements composed in various combinations. In other words there are certain common templates created on the basis of reuse of similar elementary components or more complex structures in the form of sub-systems.

Systems organized hierarchically tend to evolve over time, and hierarchical systems evolve faster than non-hierarchical ones. Simon claims that complex systems will evolve out of simple systems, if certain intermediary forms exist [Simon, 1996].

Taking under consideration basic characteristics of complex systems as well as agent approach described above we can start our argumentation, purpose of which is to show advantages of agent approach in the context of complexity management in the organization as well as information system modeling process.

As the first argument it can be noticed, that agent oriented decomposition of a problem domain is an effective way to division of the problem space, while modeling organizations and information systems. It can be concluded from a number of factors.

Firstly, hierarchical structure of complex systems causes, that modularization of organization components in terms of goals, that are to be achieved is a really intuitive solution. As Jennings and Wooldridge claim hierarchical organization of complex systems causes that at each level of the hierarchy the purpose of the cooperation between sub-systems is achieving a functionally higher level. Whereas within sub-systems components, which these sub-systems are composed of cooperate in order to achieve total functionality of a sub-system. As a consequence, decomposition oriented on goals that are to be reached is very natural division [Jennings, 2000]. Applying this schema to an organization the situation emerges where organization actors cooperate in order to achieve goals of the process, in turn processes are realized in order to achieve the goal of the organization, and organizations combine their inherent competences in order to achieve goals of the extended enterprise or virtual organization. It is worth to remember that goal orientation is one of the main characteristics of an agent and thus agent concept can be used without any additional effort.

Another vital issue is presentation of such characteristic of a modern organization as decentralization in the area of information processing and control. In this case agent oriented decomposition seems to be an optimal solution due to such characteristics of an agent as thread of control encapsulation in the form of autonomy property. The distributed organizational components may be thus modeled with autonomous agents as a basic modeling constructs.

Agent oriented approach allows also to solve problems connected with the design of interactions taking place between system components. It is a serious issue due to the dynamics of interactions between organization components. It is really frequent, that organization components enter an interaction in difficult to predict time and for unknown at the stage of design reasons. As a consequence it's really challenging to predetermine parameters of such interactions. The solution to this problem is existence of system components with characteristics thank to which they can make decisions concerning the type and range of interaction not before the system is running.

Another argument for an agent oriented approach is that it allows to eliminate semantic gap between agent abstraction used during the information system design phase and structures used during organization modeling. It is directly connected with similarities which appear between structural and behavioral characteristics of a multi-agent system and organization (table 1). Continuing this thread it is advisable to point out the following conveniences:

Mutual interdependencies present among organization actors and organization sub-systems can be naturally mapped into the system architecture in terms of high-level social interactions which take place among agents.

In most organizations dependencies of this type are present in the form of really complex network of dynamically changing relations. Agent based approach includes mechanisms which allow to describe such relations. For example, interaction protocols such as Contact Net Protocol can be used in order to dynamically create business process teams, which can be in case of such need activated and after reaching particular goals deactivated. What is more, there are off-the-shelf structures, which can be used during the community modeling, what is really useful when modeling organization actors and sub-systems [Jennings, 2000].

The process of organization modeling and system design frequently requires to perceive modeled object from the perspective of various abstraction levels, treating set of elements as atomic modeling structure. The idea of an agent is flexible enough to be used in an elementary component role on any detail level depending on the analyst's needs. For example, an agent could be organization actor, process or whole organization and components treated as elementary interact only in an integrated form omitting details concerning intra-interactions.

Organization modeling and system design with agent oriented approach leads to the structure, which has numerous stable intermediary forms, what is really important concerning complexity management. Among others it means that system components in the form of agents can be created rather independently and in case of such a need added to the system providing a smooth functionality growth.

Besides advantages of agent oriented approach mentioned above, additional benefits can be pointed out, which are connected with agent oriented approach during information systems development. They are analyzed in details in monograph [Cetnar, 1999]. As its author claims, agent oriented approach should lead to improvement and enrichment following characteristics of information systems:

- *Flexibility*. System can be rather easily adopted to the performance of new tasks in changing environment. The adaptation process is much more flexible with the use of an agent approach.
- *Reliability*. Particularly in case of heavily developed systems, which can cause un-stability problems understood as permanent or temporary loss of the ability to perform tasks of the system.
- Ability to combine functions of various systems flexibly, as well as ability to create new properties such as self-organization or self-adaptation.

What is more, application of agent orientation may lead to the simplification, improved reliability, and as an effect decrease in costs of analysis, design and implementation of decentralized information systems.

Agent oriented vs. object oriented modeling

It can be seen that agents are really similar to objects and as a consequence agent based modeling is similar to object oriented approach. However agents have some vital advantages over regular objects which can bring modeling of contemporary enterprises one step higher.

Agents are intelligent, have the ability to learn. They can use regular objects in pursuit of their own goals. Their behavior may vary depending on the circumstances and environment due to their experience. They pursue their goals actively cooperating with each other and influencing one another. As Farhoodi suggests "…business objects make a major contribution to modeling information in the enterprise. Intelligent business agents extend this capability to provide the breakthrough in modeling knowledge in the enterprise" [Farhoodi, 1993].

Object oriented notation is really well suited for software engineering, but can be difficult to understand by business people. As a consequence models created with the use of OO techniques are difficult to validate for them. Whereas models created with agent oriented approach are much more comprehendible, because the same perspective and vocabulary is used during analysis, design and construction phases in software development cycle. The example of how agent orientation may significantly reduce the semantic gap between organization model and management information system architecture has been presented in [Jakieła, 2006]] in the form of AROMA methodology.

To emphasize once again the potential of agent oriented approach let's quote Farhoodi and Fingar: "…Intelligent agents can facilitate the incorporation of reasoning capabilities within the application logic (e.g. encapsulation of business rules within agents or modeled organizations). They permit the inclusion of learning and self improvement capabilities at both infrastructure (adaptive routing) and application (adaptive user interfaces) levels.

Intelligent user interfaces (supporting task centered user interfaces and intelligent assistance to end-users) can be a boon to productivity in a network-centric world" [Farhoodi, 1993].

Agent as a tool for performance improvement

When evaluating the ways agents can be used for performance improvement it is advisable to look at the organization from the perspective of value chain model (see Fig. 2.). Porter introduced the value chain concept as a systematic way of examining all activities a firm performs and how applications of ICT interact to provide competitive advantage. A firm gains competitive advantage by performing these strategically important activities in more efficient way then competitors [McCormack, 2003]. It is possible when properly chosen ICT are applied to the processes which have the highest information density and/or are used as activities coordination mechanism.

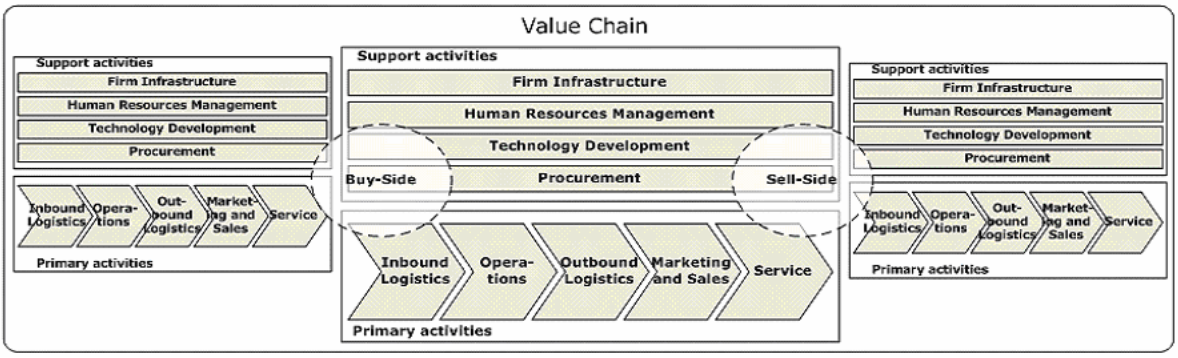

Figure 2. The value chain model

Besides the basic activities that organization performs individually, it is also important to think about links that connect it with its business partners. These "market interfaces" are known as buy-side and sell-side of firms operations. The analysis that follows, showing advantages of using agent oriented software in the process of optimizing inter- and intra-organizational effectiveness, has been divided into two parts: Agents in Supply Chain Management and Agents in Demand Chain Management. In order to reveal the superiority of agent technology over the traditional solutions, the references have been made to most important agent's characteristics (See section entitled The Agent).

Agents in Supply Chain Management

Contemporary enterprises integrate sell-side and buy-side activities into sophisticated Supply Chain Management (SCM) systems. Figure 3 depicts the generic structure of typical supply chain.

As in [Singh, 2005] was shown, agents may work as Management Information Systems, enabling integration of information flows across multiple electronic marketplaces and thus support performance improvement of the critical SCM activities. In this case the delegation of tedious tasks to agent society enables the availability of market information to all business partners as well as effective and efficient coordination of the supply chain interactions. The coordination effort is mainly related to dynamic supply and demand planning but the aggregated information flowing through e-marketplaces is extensively used also for such tasks as real-time planning, buyer-supplier selection and transaction facilitation.

It is easy to observe that agent orientation of such solution provides significant advantages to all participants over traditionally automated (systems not using agent-oriented approach) supply chains of competitors. The argumentation may be based on the analysis of basic characteristics of agents in this specific context.

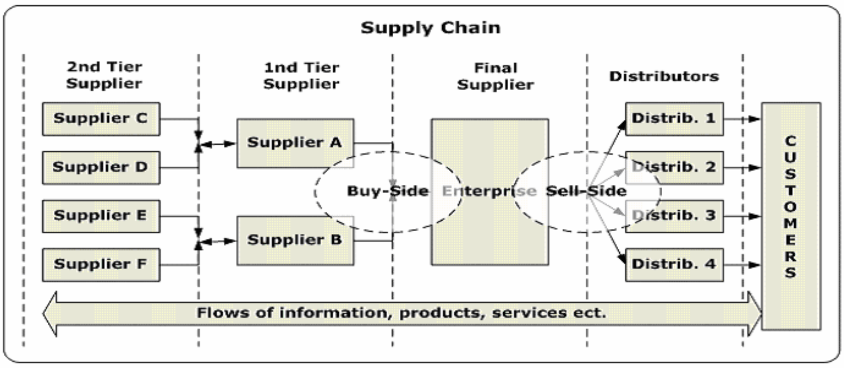

Figure 3. The generic model of Supply Chain

The SCM system architecture described in [Singh, 2005] has been developed with the use of agent oriented conceptualization of problem domain. As has been already shown in the paper, agents can be very natural and intuitive modeling constructs. Mentioned architecture consists of basic agents for buyers and suppliers as well as several auxiliary agents (discovery agents, transaction agents, monitoring agents). Agent oriented decomposition significantly reduced the semantic gap between the high-level description of the business requirements and system architecture.

The agents are situated in dynamically changing environment consisted of e-marketplaces related to business actors roles in the whole supply chain. Because agents are inextricably linked with their environment – what is an essence of agency – this is quite natural mapping between business context and the information architecture enabling incorporating all important flows into the system. Agents are monitoring role specific data such as demand requirements, matching suppliers' properties, transaction details and level of satisfaction using their sensors and affecting the flows through effectors.

Every agent operates autonomously and knows its role and behavior related to it. It is working on behalf of the user, and what is important, it performs goal oriented tasks, without direct intervention of human agent.

Roles are connected with goals, which agents are trying to proactively achieve. For example transaction agent's goal is optimal transaction facilitation, buyers and suppliers agents' goal is to maximize the level of satisfaction related to executed transactions, discovery agent is responsible for matching buyers and suppliers, maximizing the value of utility and monitoring agents fulfill the goal of marketplace synchronization.

Finally, the social abilities of agents enable the cross e-marketplace information transparency. Communication among agents plays the role of coordination mechanism and allows for dynamic and transparent planning of demand and supply requirements through real-time information integration across agent-oriented supply chain.

Agents in Demand Chain Management

On the other value chain side enterprises develop Demand Chain Management systems. The role of Management Information Systems again can be taken over by agents. Such agents are mainly used by companies to enhance the performance of Customer Relation Management Systems enabling constant assistance throughout the whole buying decision making process. According to Turban [Turban, 2008] this process includes such stages as need identification, information search, product or merchant brokering, purchase terms negotiation, payment and delivery facilitation and even post-purchase support (See Fig. 4.). On every stage customer has a goal and tasks needed for its achievement may be smoothly mapped into agent functionality.

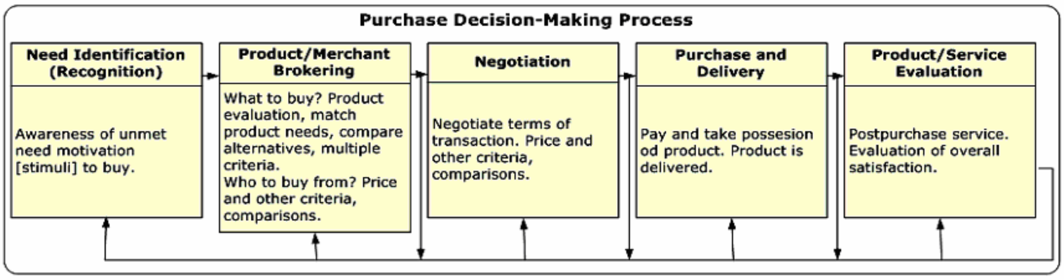

Figure 4. Buying Decision Making Process

Agents in such scenario allow unobstructed information exchange between vast numbers of actors (sellers and buyers) participating in numerous e-marketplaces, and are of invaluable help to buyers who are forced to deal with the problem of information overload thus greatly improve companies relations with their customers.

It is obvious that the application of agents in such context can bring vital advantages for all the participants (buyers and sellers) in terms of performance improvement. Let's refer again to the basic characteristic of agents. The fact that agents are highly decentralized entities enables smooth decomposition of a problem domain thus creating numerous agents responsible for various tasks related to all the stages in decision making process. For example we can observe agents that support need identification, reduce information overload, provide customers with comparison shopping engine, facilitate payments and offer after-sales support.

Agents are proactive what enables them to aggressive solution searching, active monitoring of vast amounts of information dispersed among numerous sources and as a consequence bringing the edge over traditional passive systems, which are much less effective concerning present rapidly changing environment. An agent can monitor a given information source without being dependent on the system from which it originates. Agents can wait for certain kinds of information to become available. It is often important that the life spans of monitoring agents exceed or be independent of the computing processes that created them [Maes, 1999]. There are agent implementations that are using comparable techniques to recommend complex products based on multi-attribute utility theory and assist customers in the merchant-brokering and negotiation stages. Some agent oriented solutions use automated word-of-mouth recommendation mechanism called "collaborative filtering." Whenever someone would like to buy something the system first compares a shopper's product ratings with those of other shoppers. After identifying the shopper's "nearest neighbors," or users with similar taste, the system recommends the products the neighbors rated highly but which the shopper may not yet have rated [Maes, 1999]. Proactive agents can also acting on their own initiative negotiate terms of transactions, organize product/service delivery as well as evaluate consumers satisfaction.

Agents are autonomous what allows them to work effectively on behalf of the customers, taking initiative, identifying the need of their "bosses", independently searching for the best product as well as merchant choice, making decisions related to their goals and form valid contracts. In automated negotiation for instance, agents find and prepare contracts on behalf of the real-world parties they represent. This automation saves negotiation time, and agents are often better in finding deals in combinatorially and strategically complex settings. Agents also make it possible to provide customers with dynamic pricing models. The main benefit is that the burden to determine *a priori* the price of a good is pushed into the marketplace. As a result the limited resources are allocated to those who value them most. Using dynamic pricing models in the real world may be to costly to customers (geographical collocation, time spent for offers monitoring etc.). This is where agent technologies come in. Customer can delegate the task by creating the auction, specify auction parameters (reservation price, clearing times and method for resolving tie bids) and let agent to do the rest for her. Autonomous agents can also participate during purchase and delivery stages as well as during post-purchase stage taking care of post-purchase support.

Each agent is goal oriented and each goal depends on the role of a particular agent. For example need recognition agents “keep an eye” on information sources and basing on consumer preferences inform her when the specific product is available, other agents analyzing vast quantities of information select best products and vendors, transaction agents negotiate purchase terms, delivery support agents facilitate purchase and delivery tasks, call center agents solve consumer problems and in case it is necessary connect them with appropriate human assistant. As can be seen the whole decision making process may be significantly supported by pro-active agents’ services, increasing consumer satisfaction level and creating switching costs.

What is more, agents can form societies thus enable smooth flow of information, strengthen cooperation between various actors and through the exchange of experience improve consumer trust. Agent societies sharing information can significantly enhance consumers need recognition as well as product and merchant selection. Agents organized in societies can not only negotiate terms of transaction and arrange product delivery, but also exchange information concerning post-purchase satisfaction thus helping each other making better choices in the future and greatly improving overall consumer’s loyalty.

Conclusion

Organization modeling is still a niche, but it is developing rapidly. A progression from proprietary models and tools to new standards can be seen. The growing complexity of contemporary firms requires the tools enabling us to better understand distributed and knowledge oriented business processes. It is highly likely that in ten years, business modeling will be the mainstream. Using business models, will become the natural and ordinary way for interdisciplinary teams to communicate, much as software modeling is mainstream today for software engineers.

More and more sophisticated business architectures require also flexible ICT infrastructure that will properly coordinate their operations. The paper shows that agent-orientation is a very promising solution. As Maes et al. predict “…in the near future the agent oriented solutions will explore new types of transactions in the form of dynamic relationships among previously unknown parties. At the speed of bits, agents will strategically form and reform coalitions to bid on contracts and leverage economies of scale” [Maes, 1999].

As a summary the following conclusions have been drawn:

1. Taking into consideration that characteristics of decentralized and process oriented organizations are semantically very close to properties of multi-agent systems, using agent oriented conceptualization seems to be very natural and intuitive modeling metaphor. This direct mapping eliminates semantic gap between business requirements and management information systems architecture.
2. Agent orientation can be considered as a toolbox for complexity management. It has been shown by analyzing the relationships between the classical definition of complex system provided by Simon and structures and mechanisms offered by agent paradigm. The paper shows why agent approach is especially well suited for dealing with inherently complex systems such as contemporary business organizations.
3. The application of agent–oriented software may lead to performance improvement of firms due to software agents’ properties. What can be easily observed, agent technology started to transform the way companies conduct business but the real prominent changes will occur as agent technologies mature. The predictions say that the next wave of agent solutions will be able to better manage ambiguous content, personalized preferences, complex goals, changing environments, and disconnected parties. However it is important to remember that the full adoption of agent technologies will occur after standards are widely accepted and used. Unfortunately it may take some time as in case of object oriented technologies where it took over three decades.

Bibliography

[Booch, 2007] G. Booch, R. A. Maksimchuk, M. W. Engel, B. J. Young. Object-Oriented Analysis and Design with Applications. 3rd Edition. Addison-Wesley, 2007

[Carley, 1999] K. M. Carley, L. Gasser. “Computational organization theory”, in Weiss G. “Multiagent systems. A modern approach to distributed artificial intelligence”, MIT Press, 1999.

[Cetnarowicz, 1999] K. Cetnarowicz. “Problemy projektowania i realizacji systemów wieloagentowych”, Uczelniane wydawnictwa naukowo-dydaktyczne AGH, Kraków, 1999.

[Drucker, 2000] P. F. Drucker. „Zarządzanie w XXI wieku”, Muza S.A., W-wa 2000.

[Farhoodi, 1997] F. Farhoodi, P. Fingar. Competing for the Future with Intelligent Agents; DOC Magazine; 1997.

[Franklin, 1996] S. Franklin, A. Greasser. Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents; University of Memphis; 1996.

[Gruber, 1993] T. A. Gruber. A Translation Approach to Portable Ontology Specifications; 1993.

[Hammer, 1996] M. Hammer, J. Champy. Reengineering w przedsiębiorstwie, Neumann Management Institute, Warszawa, 1996.

[Hammer, 1999] M. Hammer. Reinżynieria i jej następstwa, Wydawnictwo Naukowe PWN, Warszawa, 1999.

[Jakieła, 2006] J. Jakieła. AROMA – Agentowo zorientowana metodologia modelowania organizacji; WAEiI Politechnika Śląska, Gliwice, 2006.

[Jennin00] N. R. Jennings. Wooldridge “Agent-oriented software engineering”, Proceedings of the 9th European Workshop on Modeling Autonomous Agents in a Multi-Agent World : Multi-Agent System Engineering , 2000.

[Maes, 1999] P. Maes, R. H. Guttman, A. G. Moukas. “Agents that buy and sell. Communication of the ACM, Volume 42 , Issue 3, 1999

[McCormack, 2003] K. McCormack, W. C. Johnson. Supply chain networks and business process orientation. Advanced Strategies and Best Practices, CRC Press LLC, 2003.

[Miller, 1956] Miller, G. March. “The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information”, Psychological Review vol. 63(2), p. 86, 1956.

[Morgan, 2006] G. Morgan. Images of organizations, Sage Publications, 2006.

[North, 2003] M. North, C. Macal, P. Campbell. Oh Behave! Agents-Based Behavioral Representations in Problem Solving Environments, LNCS, Springer Verlag, 2003.

[Peppard, 1997] J. Peppard, P. Rowland. „Reengineering”, Gebethner i S-ka, Warszawa, 1997.

[Singh, 2005] R. Singh, A. F. Salam, L. Iyer, Agents in e-supply chains, Communication of the ACM, vol 48, No. 6, 2005.

[Simon, 1996] H. Simon. ”The Sciences of Artificial”, MIT Press, 1996.

[Turban, 2008] D. King, P. Marshall, J. K. Lee, E. Turban, J. McKay. Electronic Commerce 2008, 4/E, Prentice Hall, 2008

[Wooldridge, 1995] M. Wooldridge, N. R. Jennings. Agent Theories, Architectures, and Languages: A Survey; Berlin: Springer-Verlag; 1995.

Authors' Information

Jacek Jakieła, Ph.D., Eng. *– Department of Computer Science FMEA RUT; W. Pola 2, 35-959 Rzeszow, Poland; e-mail: jjakiela@prz.edu.pl*

Bartosz Pomianek, M.Phil. *– Department of Mathematical Economics and E-business, University of Information Technology and Management, Sucharskiego 2, Rzeszow, Poland ; e-mail: bpomianek@wsiz.rzeszow.pl*