

MULTI-AGENT SYSTEM FOR BUSINESS APPLICATIONS

Martin TOMÁŠEK

Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics,
Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic, tel. +421 55 602 31 78, e-mail: martin.tomasek@tuke.sk

ABSTRACT

In this paper we describe a multi-agent system which implements distributed computational system. The software architecture identifies distributed locations as abstractions of distributed computational system and various software agents as abstractions of different autonomous functional components. The interactions among agents within a system are the most critical for the application. Most common and one of the elementary types of e-commerce applications is consumer buying behavior. We propose a model of a general consumer buying behaviour application using general multi-agent system to examine its expressiveness. By identifying specific roles of software agents and designing new specific interaction patterns for them we implement the commercial model.

Keywords: multi-agent system, e-commerce, agent technology

1. INTRODUCTION

Internet is computational environment that is highly distributed, heterogeneous, extremely dynamic, and comprising a large number of autonomous nodes. Any information system operating in such an environment must handle several emerging problems. The most traditional architecture on the Internet is still the clientserver model, but it is limited and restrictive. Some nodes will want to take both roles (client and server) depending on which node they are interacting with. There are several forms of heterogeneity in this environment, e.g. different platforms, different data formats, the capabilities of different information services, and the implementation technologies. The information system must handle these problems to be able to work effectively.

The e-commerce is one of the most significant applications participating in the Internet environment. It has ambition to encompass modern technologies as event simulation, applied knowledge-based reasoning, advanced information retrieval, speech processing, etc. which are already prepared for this environment. However, there is a lack of tools and techniques for constructing intelligent clients and servers or for building agent-based software in general.

Agent technology [1] is one of the technologies that can address each of the problems mentioned above. When we describe these agents as intelligent, we refer to their ability to: communicate with each other using an expressive communication language, work together cooperatively to accomplish complex goals, act on their own initiative, and use local information and knowledge to manage resources and handle requests from other agents. Agents have attracted a lot of attention in artificial intelligence as well as distributed systems circles.

2. ROLES OF AGENTS IN BUSINESS APPLICATION

It is useful to explore the roles of agents as mediators in electronic commerce in the context of a common framework. The model we present stems from consumer buying behavior research and comprises the actions and decisions involved in buying and using goods and services.

Although the research covers many areas, it is important to recognize its limitations up-front. For example, the research focuses primarily on retail markets (although most concepts pertain to business-to-business and consumer-to-consumer markets as well). Even within retail, not all shopping behaviors are captured (e.g., impulse purchasing). Also, electronic commerce covers a broad range of issues, some of which are beyond the scope of a consumer buying behavior model (e.g., back office management, supply chain management and other merchant issues).

There are several descriptive theories and models that attempt to capture buying behavior e.g., the Nicosia model, the Howard-Sheth model, the Engel-Blackwell model, the Bettman information-processing model, and the Andreasen model. Although different, these models all share a similar list of six fundamental stages of the buying process. These six stages also elucidate where agent technologies apply to the shopping experience and allow to more formally categorizing existing agent-mediated electronic commerce systems [2]:

- Identification. Buyer can be stimulated through product information.
- Product brokering. This comprises the retrieval of information to help determine what to buy.
- Merchant brokering. This comprises the previous results with merchant-specific information to help determine who to buy from.
- Negotiation. This is about how to settle on the terms of transaction.
- Purchase and delivery. In some cases, the available payment or delivery options can influence product and merchant brokering.
- Product service and evaluation. The post-purchase stage involves product service, customer service, and an evaluation of the satisfaction of the buying process.

3. MULTI-AGENT SYSTEM

Based on the results of our previous research [3] the abstraction of the software architecture of the distributed system follows principles of multi-agent systems. We do not use the traditional design of the distributed systems, where an environment is strictly homogeneous and

implemented by standards as SOAP or REST. Our design follows principles of mobile code systems [4]. The environment is heterogeneous and can differ in each place of the distributed systems. Components of the system are presented as autonomous processes – software agents. Such a system then can be easily nested in the heterogeneous environment as Internet and fulfill the distributed tasks there (e.g. information retrieval, ecommerce tasks, etc.).

3.1. Architecture

Now there is a short overview of our previous results made on the field of multi-agent architecture design [5]. The multi-agent architecture shown in Figure 1 comprises both static and mobile agents. Static agents provide resource and facilities to mobile agents, and mobile agents move between places taking advantage of these resources to fulfill their goals. A place is a logical boundary used to delimit nodes, agents and resources into manageable and distinct entities. The agents are one of these types:

- *Host agent.* The host agent coordinates the activities that occur within a place. It offers a number of services to resource agents, user interface agents and mobile agents within a given place.
- *Resource agent.* The resource agent provides the access to information resources for mobile agents. It understands how to access the resource understands permission structures associated with the resource and offers services, by which mobile agents interrogate, update and manage resource.
- *User interface agent.* The user interface agent provides an interface between a user and a mobile agent and collaborates with the user in the architecture.
- *Mobile agent.* The mobile agent migrates between places according journey plan and performs user's tasks there. They may cooperate and they can provide groups of mobile agents. Then agents within a group can cooperatively perform defined tasks.

Each agent, whether mobile or static, is identified by a global unique agent identifier. An identifier is generated by the system at agent creation time and provides location transparency. Not only agents, but also places and groups of agents have unique global identifier too.

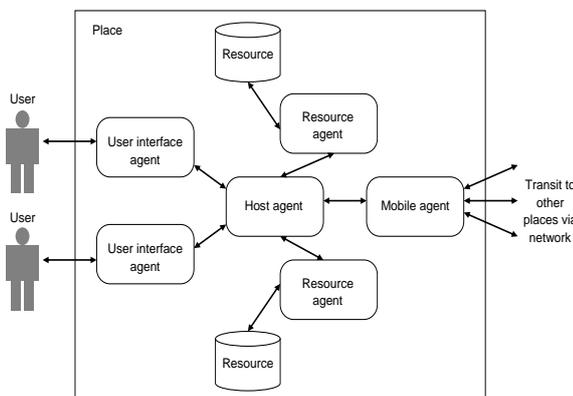


Fig. 1 Architecture of the multi-agent system.

3.2. Communication

Our previous research on agent-based distributed systems was also focused on general interaction techniques and communication schemes [6]. The short description of our results is given here. Considering interaction of mobile agents within the architecture, we have to distinguish among following types of communication:

- *Mobile agent to static agent interaction.* Since the static agents are the representatives of services in the agent world, the style of interaction is typically client-server. Consequently, static agents are requested by issuing requests, results are reported by responses.
- *Mobile agent to mobile agent interaction.* This type of communication significantly differs from the previous one. The role of the communication partners is peer-to-peer rather than client-server. The communication patterns that occur in this type of interaction might not be limited to request-response only.
- *Anonymous mobile agent interaction.* There are situations, where sender does not know the identities of the agents that are interested in the sent message. Anonymous type of communication is supported by group of communication protocols. Senders send out event message anonymously, and receivers explicitly register for those events they are interested.

There are two types of communication schemes designed within the multi-agent system among mobile agents: session-oriented communication scheme and anonymous communication scheme.

Session-oriented communication

A session defines a communication relationship between a pair of agents. Agents that want to communicate with each other must establish a session before the actual communication can be started. After session setup, the agents can interact by message passing or RPC. Sessions have the following characteristics:

Two agents participating in a session are not required to reside at the same place.

In order to preserve the autonomy of agents, each session peer must explicitly agree to participate in the session.

While an agent is involved in a session, it is not supposed to move to another place, to avoid the need for message forwarding, etc.

Anonymous communication

This scheme is used for cooperation of mobile agents in a group to perform a user task. For anonymous communication we model an application as a sequence of reactions to events that in turn generate new events. Events may be defined by user, application or system. An event-based view maps very closely onto real life, and any programming primitives that support event-based concepts tend to be more flexible in modeling a given problem.

4. DESIGN OF BUSINESS APPLICATION

The goal of this paper is to apply the abstraction of our multi-agent software architecture to the customer buying

behavior model to examine the expressiveness of our approach in the e-commerce environment.

The system is designed as a set of distributed sales places. Each sales place includes various agents supporting sales management, brokering, product retrieval and transaction based on general multi-agent architecture.

4.1. Agent Roles

Our multi-agent architecture defines four general roles of agents that should be mapped in to the main roles of the customer buying behavior model (Figure 2):

- *Product and shop identification.* The need of product identification is represented by the Resource agent. It makes the common interface between product data and the multi-agent environment and publishes the relevant product information for the buyer.
- *Product and merchant brokering.* Product brokering and merchant brokering is made by Mobile agent with specific product information retrieval task.
- *Negotiation and transaction.* Negotiation and final *transaction* is provided by Mobile agent with specific ordering and buying task.

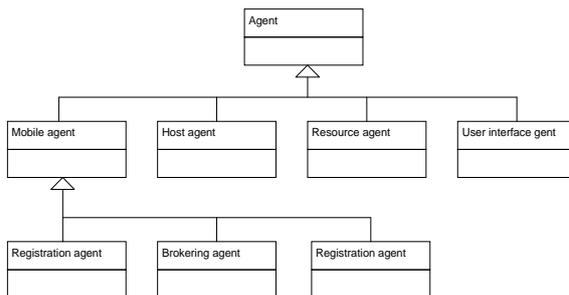


Fig. 2 Agent roles in the application.

On the other hand there are other important tasks within whole application that must be designed to implement complete e-commerce application helping customer to buy products from Internet shops:

- *Sign on and registration.* Each customer has to be registered within each sales place. The actual sales data are then used for later product brokering and merchant-specific information. The Mobile agent with specific registration and sign on tasks is designed.
- *Web-based user interface.* User interface agent is implementing web-based user interface that allows managing whole sales process from the user point of view.
- *Sales place management.* Host agent is designed to manage the coordination and cooperation of each agent within one sales place. The general management of whole e-commerce application is not defined and it is left on the autonomy of each sales place and the autonomy of each agent.

4.2. Agent Interactions

In this section we describe the design of agent interactions based on the identified roles (Figure 3). Each

agent implements its own interface with specific interaction patterns.

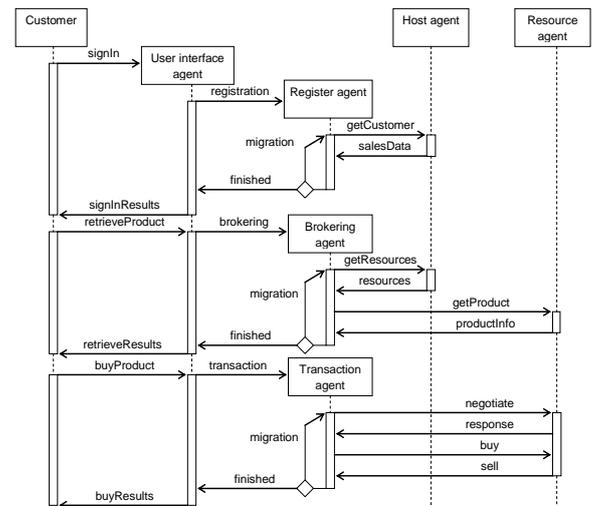


Fig. 3 Interactions in the application.

Registration agent

When a customer is signed in to the system the Registration agent is created and migrates through all sales places to retrieve registered sales data about the customer from Host agent of each place. When it is a new user, customer is registered in the signed sales place as a new one. The sales data are then used during brokering process.

Brokering agent

At first it gets the requirements for product information retrieval from the user. Then it migrates according journey plan and reads the information about products by contacting Resource agents on each sales place. When the agent returns back to the home sales place it provides results to the customer.

Transaction agent

This agent is created after brokering process is finished. The new journey plan is created from the brokering results and the agent negotiates the best price and orders (buys) the products from the sales place by interacting with Resource agents. When the transactions are completed the agent returns back and it provides results to the customer.

User interface agent

This agent is situated in each sales place and allows customer to interact with other agents providing web-based user interface:

- Enter registration information and send Registration agent.
- Enter brokering conditions and send Brokering agent.
- Select product retrieval results and send Transaction agent.
- Get results from all sent agents.

Host agent

This agent is situated in each sales place and the main purpose is to manage the interactions among agents. The agent keeps the register about customer sales data and provides them for Registration agent. The register about resources is provided to Brokering agent for identification of each Resource agent of the product within a sales place. In general Host agent allows the execution and grants access rights to each agent entering the sales place.

Resource agent

Resource agent provides relevant information about product to Brokering agent. In addition it negotiates with Transaction agent and sells the product. Host agent of the sales place, which helps providing general product database of the place, registers the Resource agents within one sales place.

5. CONCLUSIONS

Usage of autonomous software agents is one of the efficient ways to develop large heterogeneous distributed information systems for Internet environment. There are many technologies that can participate in this development. Our goal is to present the expressiveness of the multi-agent solution in this area and to test it on the most common Internet application as e-commerce.

We used our previous research to designed architecture of the distributed system based on the paradigms of agent technology and mobile code [3, 5, 6]. It is not specific distributed system and it can handle any distributed tasks within the network environment. To handle mobile agent interactions we defined two types of communication schemes: session-oriented interaction and anonymous interaction.

We defined a customer buying behavior model of the e-commerce where software agents help broker, decide, negotiate and buy the products to customer. The description of software architecture is provided in this paper and it was implemented using Java technology, where the electronic bookstore was created [7, 8]. We can see that the roles of software agents of our multi-agent architecture naturally follow the needs of the presented e-commerce application and the application design is straightforward.

Our future development is focused on more complex interaction patterns between software agents. The implementation of higher-level communication can provide better autonomy of agents in brokering and negotiation processes as well as in general communication and coordination needs.

REFERENCES

- [1] JENNINGS, N. R. – WOOLDRIDGE, M. J.: Applications of Intelligent Agents, In: Jennings, N. R., Wooldridge, M. J. (eds.): Agent Technology: Foundations, Applications, and Markets, Springer,

Berlin, Heidelberg, New York, 1998, pp. 3 – 28

- [2] MAES, P. – GUTTMAN, R. H. – MOUKAS, A. G.: Agents that Buy and Sell: Transforming Commerce as We Know It. Communications of the ACM, Vol. 42, Nr. 3, 1999, pp. 81 – 84
- [3] TOMÁŠEK, M.: Distributed System Based on Technology of Mobile Agents. Acta Electrotechnica et Informatica, Vol. 1, No. 1, pp. 55 – 60, ISSN 1335-8243, Košice, 2001
- [4] FUGGETTA, A. – PICCO, G. P. – VIGNA, G.: Understanding Code Mobility, Software Engineering, 24(5), May 1998, pp. 342 – 361
- [5] TOMÁŠEK, M.: Designing mobile agent architecture, In: Proceedings of the EMES'99 International Conference, (Z. Maghiar Ed.), Oradea, Romania, 27 - 30 May 1999, pp. 194 – 200
- [6] TOMÁŠEK, M.: Concepts for Mobile Agents Interaction, In: Proceedings of Scientific Conference with International Participation Computer Engineering and Informatics (CEI'99), Herľany, Slovakia, 14 - 15 October 1999, pp. 259 - 264.
- [7] HORVÁTH, M.: Architecture of Agent Oriented system. Thesis project, Technical University of Košice, Slovakia, 2003 (in Slovak)
- [8] HORKAVÝ, M.: Application of E-Shop in the Agent Oriented System. Thesis project, Technical University of Košice, Slovakia, 2003 (in Slovak)

Received July 30, 2015 , accepted September 7, 2015

BIOGRAPHIES

Martin Tomášek received the master degree in computer science in 1998 and PhD degree in software and information systems in 2005 both at the Faculty of Electrical Engineering and Informatics of the Technical University of Košice, Slovakia. Currently he is an associate professor at the Department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice, Slovakia. His research interests include multi-agent systems, distributed systems, component-based systems, and concurrency theory.