MAPSEC: Mobile-Agent Based Publish/Subscribe Platform for Electronic Commerce

Ozgur Koray Sahingoz[1] and Nadia Erdogan[2]

[1] Air Force Academy, Computer Engineering Department, Yesilyurt, Istanbul, TURKEY, o.sahingoz@hho.edu.tr

[2] Istanbul Technical University, Electrical-Electronics Faculty, Computer Engineering Department Ayazaga, 80626, Istanbul, TURKEY erdogan@cs.itu.edu.tr

Abstract. Electronic commerce technology offers the opportunity to integrate and optimize the global production and distribution on supply chain. Computers of various corporations, located throughout the world, communicate with each other to determine the availability of components, to place and confirm orders, and to negotiate delivery timescales. Software agents help to automate a variety of tasks including those involved in buying and selling products over the Internet. This paper presents MAPSEC, an e-commerce system based on mobile agents, that uses publish/subscribe protocol for registration and transaction processes. In this system, (as in a large-scale and dynamic environment) there can be any number of buyers and suppliers. Any supplier can connect, register or unregister to the system at any time, thus preserving the dynamic structure of the system.

1 Introduction

The number of businesses and individuals throughout the world who are discovering and exploring the Internet is growing dramatically. Through the past decades, we have seen an increasing rate of globalization of the economy and thereby of supply chains. Products are no longer produced and consumed within the same geographical area. Even the different parts of a product may, and often do, come from all over the world. This creates longer and more complex supply chains, and therefore it changes the requirements within supply chain management. The Internet is a cheap, open, distributed, and easy to use environment, which provides an easy way to set up shops and conduct commerce at any place [1].

Computer networks have the ability to rapidly distribute information to all concerned entities of an enterprise. Networks also present an infrastructure for coordination of planning and operational processes, not only within organizations, but also among them. Nowadays, people can shop for products and services online at the Internet and the World Wide Web have made electronic commerce possible. Web

A. Yazici and C. Şener (Eds.): ISCIS 2003, LNCS 2869, pp. 348–355, 2003.

sites that display products and take orders are becoming common for many types of businesses.

To enable the deployment of dynamic e-Commerce environments the European and the US Agentcities initiatives [2], combined with international initiatives in FIPA [3], aim to create a global and open information-exchange environment where dynamic services from geographically distributed organizations can be deployed, tested, interconnected and composed. Examples of such dynamic services are B2B (Business to Business) dynamic value chain creation, dynamic pricing through trading exchange, automatic discovery of business partners, advertising and marketing.

Accompanied by the growth of e-commerce, rapid responses to changes in demand and customer preference, and the ability to exploit new technologies, are becoming critical. As information on the Internet becomes more dynamic and heterogeneous, 'software agents' have been thought of as the new building blocks for a new Internet structure. In this paper, we propose a new architecture for e-commerce systems. We present a platform that uses the publish/subscribe mechanism for dynamic utilization of the system, and mobile agents as mediators between buyers and suppliers.

The rest of the paper is organized as follows. In the next section, we present some significant e-commerce systems. Section 3 introduces the computational model of MAPSEC. Our conclusions and directions for future work are presented in section 4.

2 Electronic Commerce Systems

An electronic commerce system covers any form of computerized buying and selling, both by customers and from company to company. These systems are distinguished by their implementation (or lack thereof) of the six stages of the consumer buying behavior model [4]. These stages are need identification, product brokering, merchant brokering, negotiation, purchase and delivery, and product service and evaluation. We describe several existing electronic commerce systems below.

AuctionBot [5,6] is a well-known experimental Internet auction server developed at the University of Michigan. Its users can create new auctions by choosing from a selection of auction types and specifying its parameters. Buyers and suppliers can then bid according to the auction's multilateral distributive negotiation protocols. The agents are dispatched to, and operate at, the single auction server; they do not move from supplier to supplier.

MIT Media Lab's Kasbah [7] is an online World Wide Web marketplace for buying and selling goods. A user creates a buyer agent, provides it with a set of criteria and dispatches it into the marketplace. The criteria include price, time constraints and quantity of merchandise desired. Users can select one of several price decay functions for goods they are attempting to sell. Supplier agents post their offers on a common blackboard and wait for interested buyer agents to establish contact. A buyer agent filters the available offers according to the user's criteria, and then proceeds to negotiate a deal. Both buyer and supplier agents operate within the single marketplace server and are dispatched by the buyer or supplier to that server; they do not move from server to server.

The Minnesota AGent Marketplace Architecture (MAGMA) [8] is a prototype for a virtual marketplace targeted towards items that can be transferred over the Internet (such as information). It consists of Java-based trader agents (buyer and supplier agents), an advertising server that provides a classified advertisement service, and a bank that provides financing support and payment options. Independent agents can register with a relay server that maintains unique identifiers for the agents and routes inter-agent messages.

MAGNET (Multi AGent NEgotiation Testbed) [9] is an experimental architecture developed at University of Minnesota to provide support for complex agent interactions, such as in automated multi-agent contracting, as well as other types of negotiation protocols. Agents in MAGNET negotiate and monitor the execution of contracts among multiple suppliers. A customer agent issues a Request for Quotes for resources or services it requires. In response, some supplier agents may offer to provide the requested resources or services, for specified prices, over specified periods. Once the customer agent receives bids, it evaluates them based on cost, risk, and time constraints, and selects the optimal set of bids that can satisfy its goals.

eNAs (e-Negotiation Agents) [10] and FeNAs (Fuzzy eNAs) [11] are prototypical intelligent trading agents developed at CSIRO to autonomously negotiate multiple terms of transactions in e-commerce trading. The agents can engage in integrative negotiations in the presence of limited common knowledge about other agents' preferences, constraints and objectives through an iterative exchange of multi-attribute offers and counter-offers. Fuzzy eNAs can also flexibly negotiate with fuzzy constraints and preferences. The F/eNAs environment can consist of many autonomous trading agents representing buyers and suppliers that can engage in concurrent bilateral negotiations according to a number of user-selected negotiation strategies.

3 MAPSEC

MAPSEC is a Mobile-Agent based Publish/Subscribe platform for Electronic Commerce systems. In many of e-commerce systems, a buyer (or the system) has a fixed number of suppliers, which are initialized with the system at start up. In case a new supplier is to be added, it has to be registered manually by supplying its address and the necessary parameters. In a large-scale and dynamic environment, any number of buyers and suppliers may be present at any time. In the dynamically changing electronic marketplace, we have to adopt out our electronic commerce system to this world. Therefore we propose a new architecture for e-commerce systems which uses the publish/subscribe mechanism for registration and transaction operations to increase efficiency and effectiveness of the procurement process, in terms of costs, quality, performance, and time for both buyers and suppliers. We use mobile agents as mediators to automate a variety of tasks including buying and selling of products over the Internet.

Our electronic commerce system (as most of the e-commerce systems) involves three actors. *Buyers* are looking to purchase services from suppliers. *Suppliers* or

sellers offer the services and *Dispatch Service* facilitates communication between buyers and suppliers.

MAPSEC has an extensible architecture, as depicted in Fig.1, which provides all the services, which are essential to agent-based commercial activities. This includes a communication infrastructure, a mechanism for storage, transfer of goods, banking and monetary transaction along with an economic mechanism for brokered buyer-supplier transactions.

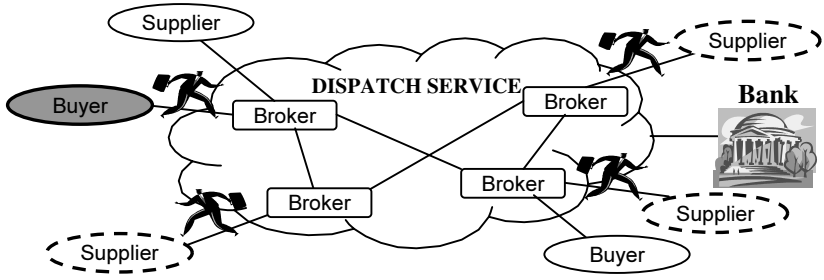

Fig. 1. The Architecture of MAPSEC

A user who wants to buy a product creates an agent, gives it some strategic directions, and sends it off into the logically centralized agent marketplace. MAPSEC mobile agents proactively visit suppliers and negotiate with them on behalf of their owners. Each agent's goal is to complete an acceptable deal, subject to a set of user-specified constraints such as a desired price, limit of an acceptable price, and a date by which to complete the transaction.

Banking: In order for an agent-based marketplace to become anything more than an experimental platform, it has to be able to communicate with existing banking and financial services. When an agent needs to make a payment, it sends a request to its bank to withdraw funds, and receive a secure wrapper, called a check. Before sending out a check, the bank verifies the existence of sufficient funds in the agent's account.

We will now describe the infrastructure details of MAPSEC subsystems (buyer, supplier and dispatch service), explain their functionalities and discuss the major design decisions.

3.1 Buyer Subsystem

An electronic transaction starts with a buyer agent. A user, whom the agent represents, specifies the name of an item(s) he wants to buy. For instance, attributes for describing a CD may be 'type', 'album', 'language', 'artist', etc.

To request a purchase order from the MAPSEC system, a buyer has to initialize a **buyer subsystem** on its machine. The Buyer Subsystem includes interfaces that are used to develop the Buyer Agent, necessary database connections and Buyer Interfaces as depicted in Fig.2.

As several transaction scenarios are possible, allowing users to generate mobile agents with different behavioral characteristics enhances the flexibility of the system. A human user interacts with the Buyer Agent via a *Buyer Interface* module. In the beginning of a transaction, the user supplies the necessary information. The buyer interface allows users to control and monitor the progress of transactions and to query past transactions from the *History DBMS*. Category of goods may include generic names such as 'car', 'CD', 'TV'. With this category information, a buyer agent prompts the user for search criteria to find the specific item to be purchased. Information required are name, maximum price, required quantity and required delivery date of the product to be acquired.

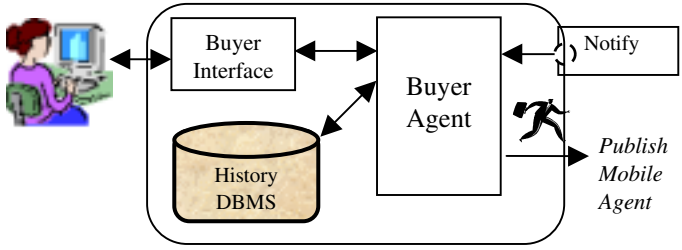

Fig. 2. Buyer Subsystem Architecture

A *Buyer Agent* is created with the basic capability to perform routine and simple tasks. Tasks that are more complicated may involve human instructions but once instructed, the agents cache them for future use. The function of a Buyer Agent is to search for product information and to perform goods or services acquisition. When a buyer agent receives a purchase request from a user, it creates a *Mobile Agent* to search for product information and to perform goods or services acquisition in MAPSEC. The Buyer Agent specifies the criteria for the acquisition of the product, and dispatches the mobile agent to the potential suppliers. The mobile agent visits each supplier site, searches the product catalogs according to the buyer's criteria, and returns to the buyer site with the best deal it finds and adds it to the History DBMS.

3.2 Supplier Subsystem

To subscribe to the system, a supplier has to initialize a **supplier subsystem** on its machine. The supplier subsystem includes interfaces that will be used for developing the Supplier Agent, necessary database connections and Supplier Interfaces as depicted in Fig.3.

The *supplier interface* module facilitates the interaction between a user and supplier agent. For instance, a user may inquire about past and current transactions from the agent. A user may also specify selling strategies through this module. When an agent is created for the first time, information regarding its name and URL is supplied through this interface to the web site maintenance component.

Supplier Agent processes purchase orders from buyer agents and decides how to execute transactions according to selling strategies specified by the user.

Since organizations differ in the products they sell, a supplier agent should be customized before it is placed online. To customize the supplier agent is to let it 'know' the goods it will sell. The *Products Database Management System* is designed for this purpose. It provides product information when requested by a supplier agent. The products database is not an operational database in the organization. As not all operational data is meant for public access, the products database only stores information that the organization has sanctioned for placement on web sites and for supplier agents. Goods will be added to the inventory either by manually adding a resource to the database or by purchasing goods from other agents.

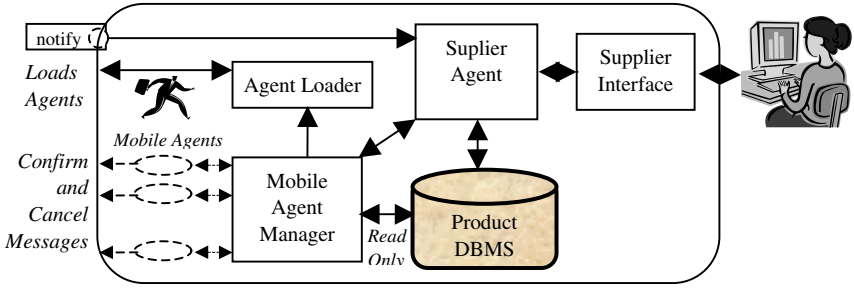

Fig. 3. Supplier Subsystem Architecture

Each *Mobile Agent* on the supplier side searches the supplier database for the product it is interest in. It determines whether the required quantity is already in the inventory and thus available to offer or not. If so, the supplier agent gives an immediate quotation to the buyer's mobile agent.

After a broker takes necessary decisions, there may be some requests to reserve a component, cancel a reservation or confirm a purchase. If a buyer cancels a reservation, the supplier agent may impose a reservation fee for the quantities of reservation reserved and bill to the buyer agent. Cancellation penalties discourage competitors from making malicious reservations that freeze stock. Because the buyer's mobile agents are fast, reservations are typically held for a short time and reservation fees are small.

3.3 Dispatch Service

The Dispatch Service plays an important role in cyberspace. A Dispatch Service is a logically centralized party, which mediates between buyers and suppliers in a marketplace. The main component of the Dispatch Service is the "broker". A Dispatch Service may consist of a single Broker or several brokers connected in a topology as shown in Fig.1. Brokers are useful when a marketplace has a number of buyers and suppliers, when the search cost is relatively high, or when trust services are necessary. The inner structure of a broker is given in Fig.4.

The client-server paradigm is not sufficient to solve satisfactorily all the problems that may arise during the life cycle of an e-commerce system. Often one needs to add a new supplier or a buyer to the system after the design-time. MAPSEC is basically an event-based system and a broker implements the publish/subscribe paradigm in which purchase events are published and made available to the supplier components of the system, through notifications.

Both suppliers and buyers have to know the address (URL) of the broker agent that they will connect, just like the URLs well-known sites, such as Yahoo!, Alta Vista or Excite. When a buyer wants to buy a product, it creates a mobile agent with the necessary information and sends this mobile agent by calling the *publish()* method of the broker. If a supplier wants to register (or unregister) to the broker, it calls broker's *subscribe()* (or *unsubscribe()*) method supplying the necessary parameters that include its products information, name, password and etc. *Knowledge Base* is a database, which keeps the information about the suppliers, buyers, brokers and products in the MAPSEC system

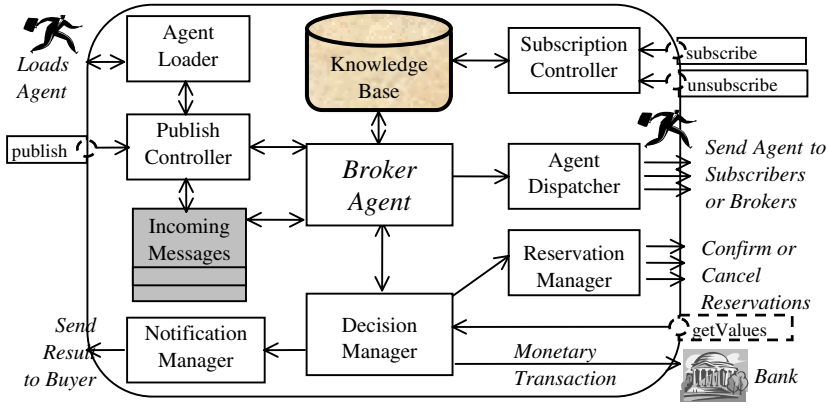

Fig. 4. Inner Structure of a Broker

Incoming agents are downloaded by the *Agent Loader* and then are dispatched to suppliers by the Agent Dispatcher when a request arrives from the Broker Agent. *Publish Controller* manages a queue of incoming messages. The Broker Agent processes these messages. *Broker Agent* is the main component of Broker System. It keeps controls all operations in the system. It evaluates an incoming message and, by using the Knowledge Base, selects a list of target suppliers and brokers to forward the mobile agent. The Broker Agent then sends copies of the mobile agent to the selected brokers and suppliers on the list, through the *Agent Dispatcher*. Suppliers and brokers return their results by calling the *getValue()* method of the Decision Manager. The *Decision Manager* compares the incoming results from the suppliers and selects the one that satisfies the constraints the best and sends a reply message to the brokers and suppliers through *Reservation Manager*. Decision Manager also sends its decision to the Buyer, to be added to its History Database, by means of *Notification Manager*.

4 Conclusions and Directions for Future Work

In this paper we introduced a general framework for a Mobile Agent based Publish/Subscribe platform for Electronic Commerce systems including mechanism for a communication infrastructure based on the publish/subscribe paradigm. The publish/subscribe protocol we have implemented allows for addition of new participants to the system dynamically, thus extending the adaptability of the system to new conditions. Another novel feature of the system is that it enables the utilization of mobile agents as mediators between buyers and suppliers. Currently a prototype of system is being implemented in Java. As future work, we have plans to change the API to conform to a standard Agent Communication Language (ACL) like KQML[12] or FIPA ACL[3].

References

1. Guanghao Yan, Wee-Keong Ng and Ee-Peng Lim. Toolkits for a Distributed, AgentBased Web Commerce System, In Proceedings of the International IFIP Working Conference on Trends in Distributed Systems for Electronic Commerce (TrEC '98), Hamburg, Germany, July 1998.
2. Agentcities Web, http://www.agentcities.org.
3. Foundation for Intelligent Physical Agents, http://www.fipa.org.
4. R. Guttman, A. Moukas and P. Maes, "Agent-mediated electronic commerce: A survey," Knowledge Engineering Review, vol. 13, no. 2 (June 1998) 147–159.
5. AuctionBot URL: http://auction.eecs.umich.edu.
6. P. R. Wurman, M. P. Wellman and W. E. Walsh, "The Michigan Internet AuctionBot: A configurable auction server for human and software agents," Proceedings of the Second International Conference on Autonomous Agents, Minneapolis, MN (May 1998), pp. 01–308.
7. A. Chavez and P. Maes, "Kasbah: An agent marketplace for buying and selling goods," Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, London, UK (April 1996), pp. 75–90.
8. M. Tsvetovatyy, B. Mobasher, M. Gini and Z. Wieckowski, "MAGMA: An agent-based virtual market for electronic commerce," Applied Artificial Intelligence, vol. 11, no. 6 (September 1997), pp. 501–523.
9. MAGNET: Mobile Agents for Networked Electronic Trading. http://alpha.ece.ucsb.edu/ ~pdg/magnet/
10. Kowalczyk R. and Bui V. "On Constraint-based Reasoning in e-Negotiation Agents". In F. Dignum and U. Cortes (Eds.) Agent Mediated Electronic Commerce III, LNAI (2000), Springer-Verlag, pp. 31–46.
11. Ryszard Kowalczyk, Van Anh Bui (2000). "On Fuzzy e-Negotiation Agents: Autonomous negotiation with incomplete and imprecise information". DEXA Workshop on e-Negotiation, UK, 2000
12. Finn, T., Labrou, Y. and Mayfield, J. "KQML as an agent communication language. In: Software Agents." Bradshaw, J. (Ed.) AAAI Press/MIT Press. ISBN 0-262-52234-9 (1997).