다중에이전트 기반의 지능형 전자상거래 시스템

이 은 석† · 이 진 구††

요 약

인터넷상에서의 전자상거래의 중요성과 복잡성이 증가함에 따라 상거래과정을 통해 소비자와 공급자 양쪽을 다 효율적으로 지원할 수 있는 지능형 소프트웨어인 에이전트에 대한 관심과 요구가 급속도로 늘어나고 있다. 이러한 에이전트를 기반으로 한 지능형 전자상거래를 실현하기 위해서는 상품정보의 표준적 온토로지의 확립, 메시지교환 및 교섭을 위한 프로트콜의 정의와 중계기술 등의 기반기술을 바탕으로 한 다중에이전트기반의 전자상거래시스템이 필요하다. 본 논문에서는 에이전트 기반의 전자상거래를 위한 개방형 기반구조의 하나로 지능형 전자상거래시스템, ICOMA를 제안한다. 구체적으로, 6종류의 에이전트기반의 전자상거래시스템의 구조와 에이전트간 상호교섭 및 상호통신을 위한 프로토콜, 개인적응형 상품검색 및 필터링 기능 등을 제안, 설계, 구현하였고 실험을 통해 그 유효성을 확인하였다.

Multiagent-based Intelligent Electronic Commerce System

Eunseok Lee† · Jingoo Lee††

ABSTRACT

With the increasing importance and complexity of EC (Electronic Commerce) across the Internet, the need and expectation for intelligent software agents to support both consumers and suppliers through the whole process of EC are growing rapidly. To realize the intelligent EC, a multiagent based EC system, which includes foundational technologies such as the establishment of standard product ontology, the definition of message and negotiation protocol and brokering, is required. In this paper, we propose an intelligent EC system, named ICOMA (Intelligent electronic COmmerce system based on Multi-Agent) as an open infrastructure for multiagent-based EC. Concretely, we have proposed, designed and implemented an architecture of multiagent-based EC system including 6-types of agents, message protocol for inter-agent negotiation, personalized product retrieval and filtering. We have confirmed the effectiveness of the system through experiments.

키워드 : 다중에이전트(Multiagent), 전자상거래(Electronic Commerce), 온토로지(Ontology), 개인화(Personalization), 가상정보시스템(Virtual information system)

1. Introduction

With the explosive proliferation of Internet, Electronic Commerce (EC) has occupied the attention of the whole company (seller) and customer (buyer) as either infrastructure or business model for all kind of business transaction. In actual fact the EC brings a lot of benefits to both buyer and seller in an economic aspect [1]. In spite of those, however, the EC has some critical problems such as the complexity in finding available information about products and stores on the large-scaled, distributed and heterogeneous world-wide on line markets, and the complexity of transaction and so on.

Agent-based approach has been expected as a good solution to cope with the above ones [2]. The agent is able to have special roles in the range of user requirement acquisition, information retrieval, filtering, negotiation, transaction, and up to the delivery. But there is no agent-based system to support the whole process fully. We have classified the conventional EC systems as follows : 1) nonagent-based systems like the greater part of current cyber shopping malls, 2) partially agent-based systems. Some systems like Bargain Finder [HREF 1], BargainBot [HREF 2], ShopBot [HREF 3], and Jango [HREF 4] are mainly for support information retrieval. Others like Kasbah [3] and MAGMA [4] provide a negotiation model. To realize multiagent-based autonomous EC system, we have to consider some important issues like ontology, message and negotiation protocol, and brokering.

† 종신회원 : 성균관대학교 전기전자컴퓨터공학부 교수
†† 준 회 원 : 삼성전자 무선개발실 연구원
논문접수 : 2000년 10월 31일, 심사완료 : 2000년 12월 29일

In this paper, we propose a completely agent-based EC system, named ICOMA (Intelligent electronic Commerce system based on Multi-Agent) to cope with the decentralized, dynamic, and diverse EC environment. We have designed and implemented ICOMA for last three years, and confirmed the effectiveness. In this paper, we particularly focus on the architecture of ICOMA and the product search and filtering, and negotiation mechanism for space constraints.

In Section 2, the architecture of ICOMA is suggested. Section 3 reviews the basic product search problem, and describes how our search policy operates. In Section 4, we give suppliers a solution to construct a business-enabled shop on the WWW. Section 5 proposes our filtering strategy restricted on structured information. In Section 6 we review the definition of an automated negotiation in EC, and present our electronic negotiation strategies. Section 7 presents the results of our experiments. This paper concludes with a discussion of future work and conclusions.

2. ICOMA

ICOMA has been designed with the following design concepts.

- *A Virtual Information System*

To make market transaction in EC, user should first find counterparts who can satisfy his demand and then search product of his interests, which is supplied by each counterpart as much as possible for the comparison-shopping. Most users search items with general search engines but they can only point to a lot of shopping malls. So user must get the ability to perform this procedure by continuous efforts. It is a hard work to an unskilled user because of the exponential growth and chaotic evolutionary structure of Internet. To solve this problem, ICOMA offers a service of autonomous product search from heterogeneous electronic markets on behalf of user. In other word, ICOMA plays the role of virtual information system.

- *Personalization*

After product search from many electronic markets, the filtering process is needed. By this process, user chooses the product that he really wants. But this process is very tiresome work. ICOMA offers a personalized service to filter the products by introducing hybrid-learning mechanism, which consists of memory-based learning and reinforcement one.

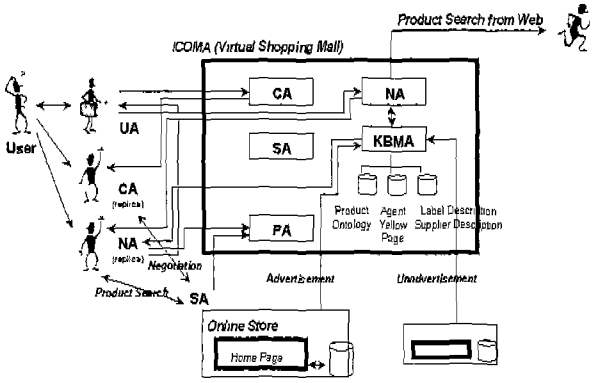
- *Delegation*

As a final state in EC, user must negotiate with his corresponding counterpart manually. ICOMA assists its user by performing negotiation. In other word, user's intelligent agents precede the negotiation not by direct contact between users.

2.1 Architecture of ICOMA

ICOMA is a multi-agent system in which six semi-autonomous agents interact or work together to perform an user's goal. ICOMA is mainly composed of the following agents : User Agent (UA), Customer Agent (CA), Supplier Agent (SA), Navigation Agent (NA), Knowledge Base Management Agent (KBMA), and Payment Agent (PA).

The complete architecture is depicted in (Figure 1).



(Figure 1) Architecture of ICOMA

2.1.1 *User Agent (UA)*

UA is a personalized learning interface agent that provides active assistance to the user and becomes smarter and more adaptive to its user. The UA receives its user's request for selling or finding product then sends the request to a replica of the other functional agents. After receiving the results from the other functional agents, the UA refines the results into the user adaptive form with the user information acquired by monitoring user's behaviors and learning (and/or training) from the user-after receiving the list of products as the results from NA (Navigation Agent : Product Search Agent), the UA ranks and recommends each product.

To serve as a user's intelligent assistant and acquire its competence, the UA employs a reinforcement learning and memory-based learning mechanism.

2.2.2 *Customer Agent (CA)*

The CA acquires user's request that which product he wants. To do this role, the CA should decide which product

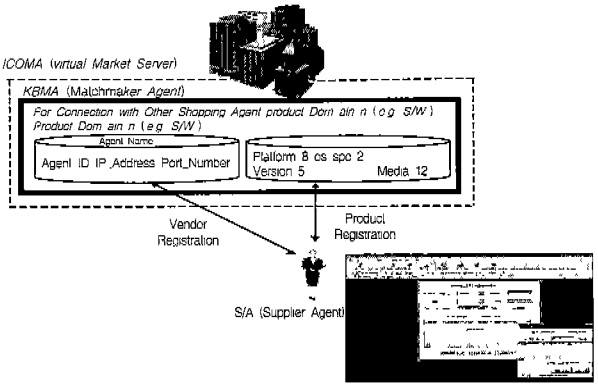
domain user wants and then receive user's interest in a product specification form. To get the user's interest, the CA creates the dynamic user interface based on product ontology (a knowledge base of which contents are structured product information) that is managed by the KBMA. The CA finds SAs by querying the matchmaker agent for the list of the SAs that may satisfy user's needs to buy items of a specific domain.

The CA finds the product from SAs by sending request message that contains product specification user described. Then the CA integrates the results and sends them to the UA. After receiving user's allowance to buy the product and user's strategies for negotiation, the CA starts negotiation with SA which sells the item user mentioned on behalf of the user.

2.2.3 *Supplier Agent (SA)*

For individual sellers, the SA acquires user's request that which products he wants to sell and then advertises/unadvertises himself to the matchmaker agent (KBMA). To assist suppliers who want to create dynamic, commerce-enabled business market on the WWW more efficiently and economically, SA plays a role of merchant server. The SA creates HTML-based pages and DB according to the inputs, and a supplier can easily change his pages and the change is reflected automatically to the DB (see (Figure 2)).

Whenever a SA receives a request message for buying from a CA, the SA responds by sending a message describing which contract type is applied for this product. And then the SA starts negotiation with the CA on behalf of the user. SA provides a solution to ontology problem. SA connects with KBMA, then refers and adds a new attribute on the current product ontology. And when receiving a message saying that other supplier updated product ontology, the SA notifies the change to its user.



(Figure 2) User Interface of SA as a Merchant Server

2.2.4 *Navigation Agent (NA)*

As more and more products become available on the Internet, users become more and more desperate for tools that will help them search the products of interest to them.

As a user's intelligent delegate, NA plays a role of commerce-specific search engine. The NA finds corresponding counterpart who can satisfy user's demand, and retrieves the products of user's interest. But the heterogeneous presence of information and exponential growth of the Internet complicate the problem. To solve this problem, there is some talk on standard for describing product, but it is impossible to develop a standard language that is expressive enough to cover the diversity of all products.

In this paper, we propose NA as an alternative way. The NA can find products from ordinary online shopping stores that provides searchable form page. In other words, the NA operates as an adaptable and trainable intelligent CGI agent that can be adapted to all product domain with user's training. The NA gets user's demand in a product specification form and matches each input to the corresponding form then construct the query and sends to the CGI server simultaneously. After receiving results from multiple online store, NA returns the output to the user.

2.2.5 *Knowledge Base Management Agent (KBMA)*

The KBMA manages the knowledge base - including Product Ontology, Agent Yellow Page, Label/Supplier Description :

- Product Ontology is a structured information on product descriptions and is classified by product domain in a tree structure. Product ontology is items agent to update and describe in a cooperative way. And the customer can describe products or services of their interest.
- Agent Yellow Page is a knowledge base for matchmaker agent name and agent capability-agent name should be enough to enable direct communication between agents. And agent name is expressed by the composition of Agent_ID, IP_Address and Port_Number. Agent Yellow Page is classified by product domain in a tree structure, so agent capability is inferred by traversing the tree.
- Label Description and Supplier Description are the inference results of the NA about product retrieval method of on-line markets. Label Description is structured as the same way with the product ontology (See <Table 1>).

NA creates supplier description after visiting, querying,

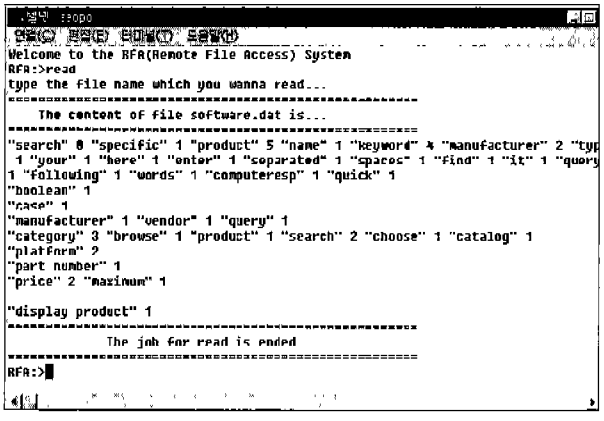
and analyzing the result. The content of supplier description is composed of the followings : URL that points the form page for quick finding, query description that describes the query format, *method attribute* of a supplier's online store (GET/POST), *product information format* showing the format with which product information is described, *failure page* which is presented when there is no such item (See <Table 2>).

2.2.6 *Payment Agent (PA)*

PA provides an payment service for market transaction between customer and supplier. It is mostly affected by international or domestic standardization. The representative payment systems are JEPI proposed by CommerceNet and CAFE project developing secure and confidential electronic payment systems for the European Community. The key is to develop a high security payment system that meets the international standard payment protocol. In this paper, PA is just for the connection with other payment system.

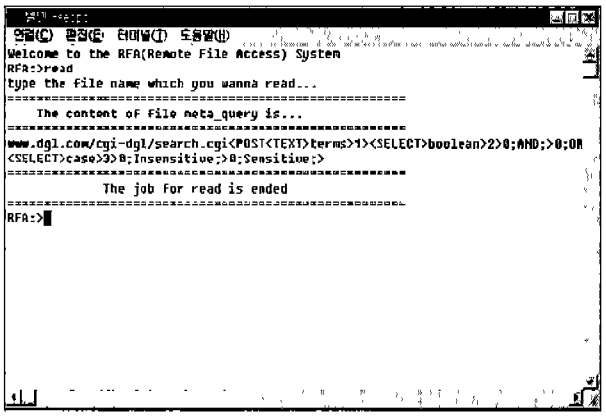
<Table 1> An example of LD

Attributes	Regular Expressions (User Training)	Label Description (Atomic Strings)
Product Name	search for the following words, search, product, search for, search by product, search by, product name, keyword search	search, product, keyword
Brand	manufacturer, brand, publisher	manufacturer, brand, publisher
Price	maximum price, price, from	price. from
	to	to
Platform	platform, OS for	platform, OS for
Category	category	category



<Table 2> An Example of Supplier Description

Meta Query	www.isn.com/cgi-bin/getNode&node = 110418&result_count = 0&next_state = results&equal_id.manuf = 2&equal_id.name = 3<POST<RADIO>fecfifofum>>1 ; Product Name ; product > 0 ; Manufacturer ; manufacturer
Failure Page	www.isn.com.fp
Product Information Description Format	<tr><td><a><td><a><td><a><td>



3. Product Search

To realize the full potential of EC on the Internet, the ability to accomplish an effective search for specific product items is crucial. As more and more products become available on the Internet, users become more and more desperate for tools that help them locate the products of interest, and the need for software agent technology has been emphasized. This kind of software agent is called as shopping agent.

These kinds of systems such as BargainFinder, BargainBot and WebShopper, however, have a number of constraints. First, the agent can retrieve from the same online markets that are hand-coded at the early development phase. Second, the agent can handle only a few restricted pre-assigned items. Third, there is no profit to suppliers who are inferior in price because the stores are evaluated on price alone.

The next research version of the above systems operates as an intelligent CGI agent. It infers the vendor's searchable form page with domain knowledge and transforms user's requests into each search page adaptable form, then integrates and returns output to the user. It also has open questions. First, to create knowledge base that is used in the inference phase is very hard, so it can retrieve only pre-assigned items and can not support the user who wants to employ this system to new product domain. Second, it relies on a very strong bias, which ought to be weakened somewhat. In particular, it assumes that each product description resides on a single line, and that all descriptions on a single page have the same format.

To solve the above problems, ICOMA suggest NA. The NA is an adaptable and trainable commerce - specific search engine with a web robot which automatically make the web based online markets ready for search by building indexes. Users can extend NA's ability by informing of a new online store or applying to a new product domain - NA has a basic knowledge on product independent attributes in advance. With user's training, NA finds online markets on the Internet and induces query for each online market, then extends the knowledge base (LD : Label Description, SD : Supplier Description) by itself. One user's training is shared with the other users because NA is not an application but a replica downloaded with the knowledge base from market server, named ICOMA. The NA can automatically extend their knowledge base (supplier description) by itself.

3.1 *User Training*

NA gets user training for extension of label description (one per product domain, e.g. software, printer etc.) : regular expressions (form label describing an attribute, usually different from each supplier), example input for each form, URL of the online market that user want NA to manage.

The structure of label description is like as following.

$$\textbf{\textit{Label Description}} = \{\, i \mid \textbf{\textit{Attribute}}\ i = \{\, j \mid \textbf{\textit{reg\_exp}}\ j \,\} \,\}$$

3.2 *User Training Verification*

It is possible that a badly intended user or a novice user may mis-train the NA. To avoid this kind of mis-training, NA verifies the user training after being trained by its user. At first, NA verifies the user training by inspecting the form page and regular expressions which user input. And then, NA computes the confidence by drawing a parallel between regular expressions and domain specific thesaurus (thesaurus is a collection of label description). If the confidence is less than a threshold, NA warns a user against the possibility of mis-training.

3.3 *Induce Retrieving Page*

Most online markets have more than one form page. NA selects one among them, which contains more forms for domain specific attributes than any other form pages.

3.4 *Failure Page Learning*

NA learns the failure page that provided differently by each supplier through submitting each form with meaningless value.

3.5 *Query Generation*

NA generates query by matching each regular expression against LD. The matching algorithm is depicted (Figure 6). If a certain searchable page contains an untrained, NA regards that there is no user input against the untrained form.

$$Label\_Match(RE,\ LD) = \sum_{i=1}^{\#RE} \max_{j=1}^{\#LD}(compare\ (RE_i, LD_j) \times \frac{FN_j}{F\_Sun})$$

$$(FN: Frequency\ Number,\ F\_Sun: Sum\ of\ FN)$$

$$compare(RE_i, LD_j)\ is$$

$$if\ (LD,\ contains\ RE_i)\ \{$$

$$if(|RE_i| \geq |LD_j|)\ |LD_j|/|RE_i|$$

$$else\ |RE_i|/|LD_j|\}$$

$$else \qquad 0$$

(Figure 3) Label Matching Algorithm

3.6 *Result Analysis*

After receiving result, NA compares the result with the failure page. If the result is not the failure page, NA considers the result is successful. NA queries one more differently from the former (if an user submitted his request with four submissions, the latter contains two submissions).

Then, compares the two result pages form the top and bottom. NA regards that the same information is not a product information (it's just for other use, advertising, for example). After removing the overlapping parts, NA induces the format of product information with which the product is described- the format is the set of html tags that are used to describe each item in the web-based online market. Most suppliers describe their products in the same format for a sense of identity.

3.7 *Supplier Description Generation*

After the above procedures, NA generates SD for each supplier. Once SD is generated, A can construct query directly without learning phase for generating SD.

Supplier Description(SD)
= { *URL of searchable page for product search, Query Description, Method Attribute, Product Information Format, Failure Page* }

3.8 *Label Description Management*

NA refines newly-added regular expressions into a set of

atomic words. The label description is a pair of atomic word and frequency number. Pseudo code for label description management is depicted in (Figure 4) and an example of the LD is shown in <Table 2>.

```
Given:
        User Training (UT)
        - attribute and regular_expression set for domain K
        UT_K = {i | <Attribute_i, RE_i >}
        .Past UTs for the same domain (Label Description: LD)
        - LD_k = {j | <Attribute_i, RE_j1, RE_j2, . . . RE_jl>} .

Determine Atomic Strings per Attribute_i
- in each RE_i, stop words {and,in,for,the,of,&,space,-,/,',;}
are removed
- tokenize RE_i and RE_jl
  each RE_jln has frequency number
  RE_i = {RE_i1, RE_i2, . . . RE_im}
  RE_jl = {<RE_jl1,FN_jl1>, <RE_jl2,FN_jl2>, . . . <RE_jln, FN_jln>}
- if RE_jl is not empty
  for each RE_jln {
    for each RE_im {
      if match (RE_jln, RE_im)
        { add(temp)
        increment corresponding frequency number }
      else continue }}
    if temp is not null
     replace RE_jl with temp
```

(Figure 4) Pseudo code for label description management

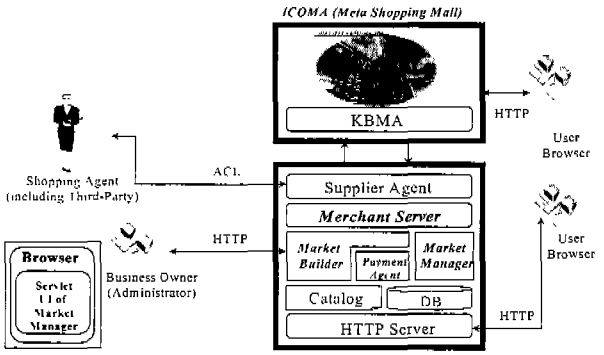
3.9 *Automatic Extension of Supplier Description*

NA visits new online markets and induces retrieving method then generates SD by itself. NA finds the new online market by user training and asking to search engines which provide a directory service (e.g. Yahoo, Excite, InfoSeek, Lycos, WebCrawler).

4. Enabling Business with ICOMA on the Internet

The Internet can no longer be ignored as a serious business tool, and the World Wide Web (WWW) is the vehicle that has brought the potential of the Internet to the masses. With its interactive look, a hospitable and compelling environment for novices and low cost, the WWW is a wideopen marketplace. Electronic commerce provides businesses a growing, dynamic channel for efficient delivery of goods and services to customers. Many companies have had tremendous success on the Web. The number of business professionals setting up shop and frequenting the Internet has increased dramatically over the last year. But to construct a businessenabled electronic market has been inhibited by high cost and complexity.

To solve this problem, ICOMA gives suppliers a solution to construct a businessenabled shop on the WWW. SA is a kind of software tool enabling suppliers to create commerce related web content more easily, and allow the web content to be both businessenabled and dynamically generated in a clear, searchable fashion on the basis of database (the commerce related web content is stored and managed in the relational database). The database contains product information as well as user information. With this user information, suppliers can do a personalized marketing to more potential customer by email. And because all updates to the database are immediately reflected on the electronic catalog, suppliers can manage their electronic market more easily. The whole architecture of merchant server is depicted in (Figure 5).



(Figure 5) Architecture of Merchant Server

5. Product Filtering

Product filtering is to extract the products of user's interest from searched items and to let the user concentrate on the important product information. Product filtering is a personalized service and UA do this role. Since users' interests are different in many ways, UA has to adapt to the behavior of its user. To serve as a user's intelligent assistant and acquire its competence, UA employs a reinforcement learning and memorybased learning mechanism. UA keeps a memory of everything the user does, stored as proposed item and feedback pairs (An item is described with attributes and values. The feedback is one of the following affirmative/negative values : 1, 0.5, 0, 0.5, 1). This set of pairs is called as user history. UA maintains a set of priority weightings for each attribute. A set of priority weightings is used to induce user's preference. Given a new proposed items from NA, UA can then use memorybased learning mechanism to extract an items which user really wants. UA scores each item then recommends the high scored items. The score is computed as a weighted sum of the scores between the values in each attribute. The score is computed as :

$$\sum_i ((P_{oi} * W_i) * Opinion)$$

where,

- P_{oi} (profile of attribute i) is the percentage of items which has affirmative user feedback and the same value for attribute i with the offered item.
- W_i is a priority weighting of attribute i.
- Opinion is the user feedback against an item that is stored as user history.

After recommendation of some items, UA gets user's feedback for each item. User feedback is used to adjust the priority weightings the UA keeps. Through the reinforcement learning, UA becomes smarter and more adaptive to its user with usage, and acquires its competence.

The weights are computed as follows :

$W_{ai} = W_{ai} + *feedback*opinion(in\ case\ of\ a_{hi} = a_{oi})$

$W_{ai} = W_{ai} - *feedback*opinion(in\ case\ of\ a_{hi} \neq a_{oi})$

$$\alpha_N = \frac{1}{N}$$

where,

- W_{ai} is a priority weighting of attribute i.
- feedback is an user's feedback for proposed item
- opinion is the user feedback against an item which is stored as user history.
- a_{hi} is a value of attribute i of the item stored as user history
- a_{oi} is a value of attribute i of an offered item

6. Automated Negotiation

An automated negotiations take place when negotiation functions is performed by intelligent software agent programmed with a negotiation strategy by its user. ICOMA adopts an auction as a negotiation protocol and applies the following four main types of auctions the English auction, the Dutch auction, the Firstprice, sealedbid auction, the Vickrey auction. In ICOMA, the negotiation can be proceeded manually or automatically. Users should initially program their software agents with negotiating strategy for automated negotiation. The software agents would electronically negotiate with each other within the user's predefined strategy. The truth is that the entire subject of auction strategy is very complex with numerous factors coming into play. In ICOMA, the negotiation will be along with the dimension of price alone. The strategy and information varies from auction type :

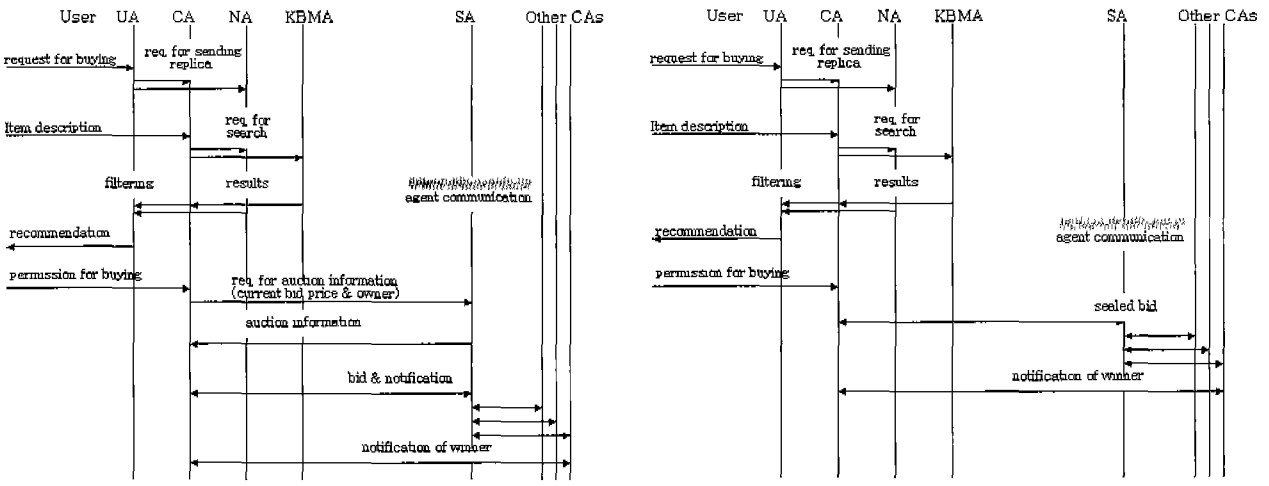
- English Auction

In a privatevalue English auction, a customer's best strategy is to bid a small amount more than the previous high bid until he reaches his price limit and then stop. But the bidding is influenced by customer's motivation. If a customer wants the item enthusiastically, he will raise his bids much more than the previous high bid so that they are more likely to win. If a customer wants to pay the lowest possible price, he will bid a small amount more than the previous high bid. The bids are determined as :

$$Bid = PHB + \frac{MPL - PHB}{RR}$$

where, PHB is the previous high bid and MPL is the maximum price limit and raising rate RR (percentage).

From supplier's perspective, supplier can influence results by opening an auction with the start price.



(Figure 6) Message & Data Flow in Auction Protocol

• Dutch Auction

A supplier's strategy is influence by supplier's motivation. If a supplier wants to make as much profit as possible, he will lower the price slowly. If a supplier sells items that must get to market quickly as possible (e.g. flowers), he will lower the price quickly. The offers are determined as :

$$Offer = PHO + \frac{PHO - mPL}{LR}$$

where, PHO is the previous high offer and mPL is the minimum price limit and lowering rate LR (percentage). From customer's perspective, customer must decide the maximum price limit he will pay and programs his agent with that variable. Customer agent will stop the auction based upon the user's maximum price limit.

• Firstprice, sealed bid Auction & Vickrey Auction

Firstprice, sealed bid auction and Vickrey auction are a kind of sealed bid auction. The best strategy in a sealed bid auction is to bid his true price without regard for the actions of others. A supplier opens an auction at a time constraint and a customer submits his sealed bid within the time limit.

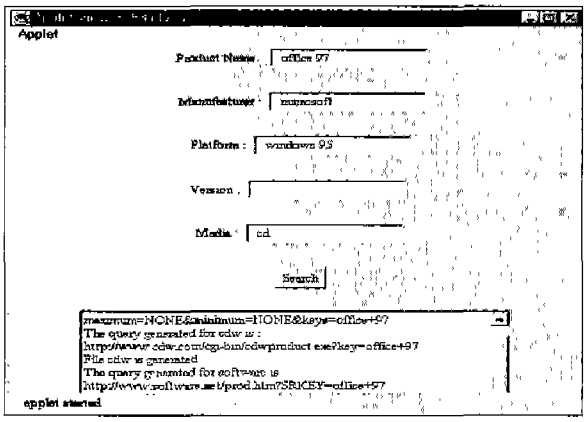
The above four types of auction can classified into the two in view of data and message flow outcry auction :

English & Dutch auction, and sealed bid auction : Firstprice, sealed & Vickrey auction. The data and Message flow of the two classified auctions is depicted in (Figure 6).

7. Experiment & Evaluation

7.1 Product Search

In these experiments, we attempted to measure the usefulness of NA. The usefulness is divided into user support aspect and system learning ratio. We consider the following three evaluation items responsible for the NA's user support aspect utility. First is flexibility. NA can analyze the form in each supplier's pages and generate queries automatically for each supplier after a user's training. NA is flexible system because it can be adapted to any product domain and any supplier by users' trainings. Second is user's information overload. NA reduces a user's burden for commerce by retrieving and synthesizing pure information to user interesting. Third is a comparison-shopping. User can compare many products provided by multiple suppliers broadly, so NA provides users with a chance for efficient electronic shopping. The experimental results are shown in (Figure 7) and <Table 3>.

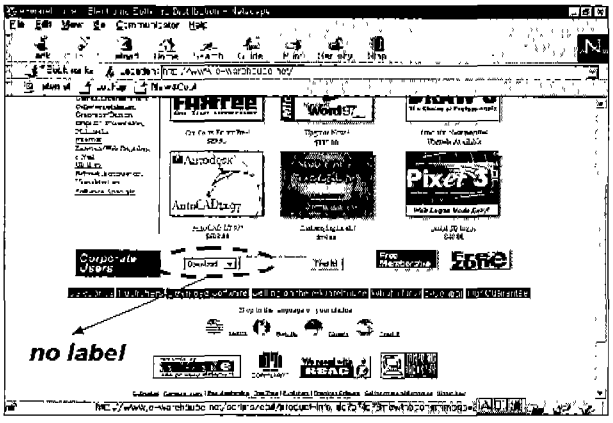


(Figure 7) The Queries Generated by NA

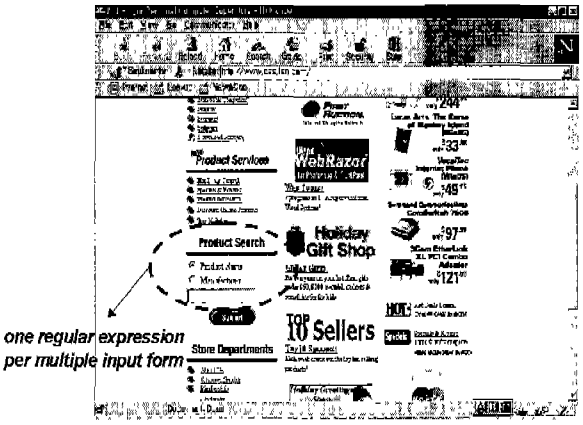
<Table 3> The Result of Experiment

Group	Product Name	Price	Elapsed Time
Group 1	Microsoft Office 97 Bookshelf Microsoft Office 97 Standard Netscape Navigator Gold 3.0 Netscape Navigator V. 3.0	$537.42 $449.16 $59.63 $38.76	11 : 51
Group 2	same as group 1	Not found $465 $79 $49	92 : 30

From the result, we have confirmed that to find online markets that provide the products of user interest, is very hard and inefficient by using current search engines. On the other hand, we have verified that the NA users can complete the task much faster and more comfortable than the others, and find the products with lower price. And we experiment the NA's learning ability by training in the software domain. Among the total 19 online shop, NA can generate 15 metadata. The reasons of the form page analyzing failure is like a follows : first, there is no label which describes each input form(see (Figure 8)) second, one label describes multiple input forms(see (Figure 9)).

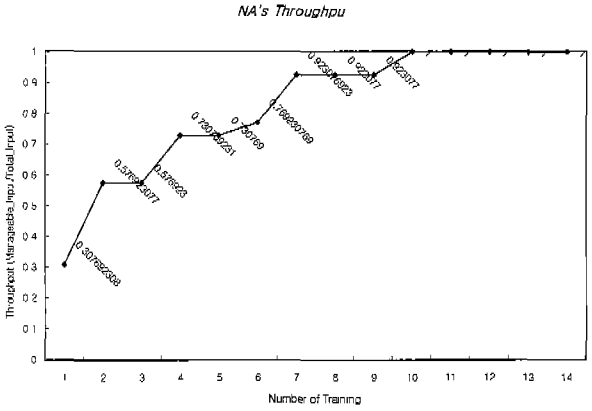


(Figure 8) An example of no label



(Figure 9) An example of one label describes per multiple input forms

The result is depicted at (Figure 10). The y axis describes the throughput which is calculated by dividing total number of input forms into the number of manageable input forms. And the x axis describes the number of online shops trained.



(Figure 10) The result of Experiment

From the result, we can confirm that at the software domain, which has about 9 attributes, can be manageable after 10 times training.

8. Conclusion

In this paper, we have proposed an intelligent EC system based on multiagent, named ICOMA, as an infrastructure for multiagentbased EC. Although we focused on product search and filtering that is necessary for commerce and serving user client as a proxy in both case of buying and selling toward an autonomous deal, the experimental results show that ICOMA can reduce user's information overload for commerce by helping locate information of user interest and by applying sophisticated filters to data.

The more experiments are planned to evaluate the effectiveness of ICOMA quantitatively from now.

References

[1] E. S. Lee, "Agent-based Electronic Commerce Tutorial," International Conference on Electronic Commerce ICEC'98, April, 1998.

[2] E. S. Lee, "Application of Agent Technologies to EC," The Magazine of the IEEK, Vol.26, No.1, pp.61-70 Jan. 1999.

[3] A. Chavez, P. Maes, "Kasbah : An Agent Marketplace for Buying and Selling Goods," *PAAM96*, pp.75-90, 1996.

[4] B. M. Tsvetovatyy, M. Gini, "MAGMA : An Agent-based Virtual Market for EC," to appear in Journal of Applied AI, 1997.

[5] Adam Farquhar, Richard Fikes and James Rice, "The Ontolingua Server : a Tool for Collaborative Ontology Construction," Knowledge Systems Laboratory, KSL9626, 1996.

[6] H. Takeda, K. Iwata, M. Takai, A. Sawada and T. Nishida, "An Ontology-based Cooperative Environment for Real-world Agents," Proceedings of the International Conference on MultiAgent System (ICMAS-96), pp.353-360, 1996.

[7] T. Finin, R. Fritzson, D. Mckay, and R. McEntire, "KQML as an agent communication language," Proceedings of the third International Conference on Information and Knowledge Management (CIKM-94), 1994.

[8] Maes, P., "Agents that Reduce Work and Information Overload," Communication of ACM, Vol.37, No.7, pp.31-40, 1994.

[9] Stanfill, C. and Waltz, D., "Toward Memory-based Reasoning," Communication of ACM, Vol.29, No.12, pp.1213-1228, 1986.

[10] Maes, P. and Kozierok, R., "Learning Interface Agents," Proceedings of the AAAI'93 Conference, 1993.

[11] Robert, B. Doorenbos, Oren, Etzioni, and Daniel, S. Weld, "A Scalable Comparison-Shopping Agent for the World-wide Web," Proceedings of the First International Conference on Autonomous Agent, 1997.

[12] Finin, T. et al., "DRAFT Specification of the KQML Agent-Communication Language," The DARPA Knowledge Sharing Initiative External Interface Working Group, 1993.

[13] Keith Decker, Mike Williamson, and Katia Sycara. "Matchmaking and Brokering," Proceedings of the Second International Conference on MultiAgent Systems (ICMAS-96), pp.432, 1996.

[14] Carrie Beam and Arie Segev, "Automated Negotiations : A Survey of the State of the Art," CITM Working Paper 97-

WP-1020, University of California at Berkeley, 1997.

[15] Joseph Williams, "Bots and Other Internet Beasties," Sams Net, pp.257-263, 1996.

[16] Genesereth, M. R. and Ketchpel, S. P., "Software Agents," Communication of ACM, Vol.37, No.7, pp.48-53, 1994.

[17] J-P. Boly, et al, "The ESPRIT Project CAFE High Security Digital Payment Systems," Computer Security-ESORICS '94, SpringerVerlag, pp.217-230. 1994.

[18] M. Perkowitz, R. B. Doorenbos, O. Etzioni, D. S. Weld, "Learning to Understand Information on the Internet : An ExampleBased Approach," Journal of Intelligent Information Systems, to be appeared.

<Hypertext References>

[HREF1] http : //bf.cstar.ac.com/bf/

[HREF2] http : //www.ece.curtin.edu.au/~saounb/

[HREF3] http : //www.dimos.de/dnns/original/

[HREF4] http : //www.jango.com

이 은 석

e-mail : eslee@seopo.skku.ac.kr

1985년 성균관대학교 전자공학과(공학사)

1988년 일본 Tohoku (동북)대학교 정보공학과(공학석사)

1991년 일본 Tohoku (동북)대학교 정보공학과(공학박사)

1992년~1993년 일본 미쯔비씨 정보전자연구소 특별연구원

1994년 일본 Tohoku(동북)대학교 전기통신연구소 Assistant Prof.

1995년~현재 성균관대학교 전기전자컴퓨터공학부 부교수

관심분야 : 소프트웨어공학, 전자상거래, 에이전트지향 지능형 시스템, HCI 등임

이 진 구

e-mail : jglee@khgw.info.samsung.co.kr

1996년 성균관대학교 정보공학과(공학사)

1998년 성균관대학교 정보공학과(공학석사)

1998년~현재 삼성전자 무선개발실 연구원

관심분야 : 무선이동통신, 에이전트지향 지능형 시스템, 전자상거래