

## Integrated Business Agents for Processing Free Text Information\*

I-HENG MENG, SHIN-MU TSENG\*\* AND WEI-PANG YANG

*Department of Computer and Information Science*

*National Chiao Tung University*

*Hsinchu, 300 Taiwan*

*E-mail: ihmeng@iii.org.tw*

*E-mail: wpyang@cis.nctu.edu.tw*

\*\**Department of Computer Science and Information Engineering*

*National Cheng Kung University*

*Tainan, 701 Taiwan*

*E-mail: tsengsm@mail.ncku.edu.tw*

The Internet functions as an increasingly important channel for retailing commerce and business transactions. Although nearly five billion web sites exist on the World Wide Web, an integrated mediator, called a business agent, is sought to negotiate between suppliers and buyers. Such an intelligent business broker is thus necessary to support efficient and effective sharing of information. This work proposes an entirely new business agent architecture comprised of five elements, namely Business Spy Agent, Supply and Demand Analysis, Supply and Demand Classification, Matching and Negotiation, and Hidden Business Mining. The proposed architecture encompasses four trading processes, including supply and demand identification, product brokering, merchant brokering and negotiation. The architecture introduces the notion of ontology to increase the precision of classification and more the effectiveness of negotiation. Experiments involving the proposed business agent show a significant successful pairing rate of 82% among suppliers and buyers. Simulation results indicate that the options, including Negotiable, Designated Brand and Split Demand, affect the match ratio differently from various perspectives.

**Keywords:** business agents, ontology, negotiation, classification, supply and demand

### 1. INTRODUCTION

The Internet has become a very successful platform for exchanging messages owing to its ease of use. Consequently, an increasing amount of commercial trading is being undertaken on the Internet. As the amount of available information has increased, users and companies have wasted much time in searching and browsing various web sites to obtain the required product information. As information on products is becoming easier to access, human beings are becoming used to following the procedures of the conventional buying process, increasing the transaction costs. A buyer is still responsible for

---

Received September 3, 2002; accepted March 17, 2003.

Communicated by David H. C. Du.

\* A preliminary version of this paper appeared in the 2002 International Conference on Computational Science (ICCS).

collecting and interpreting information on products, making decisions about products, negotiating prices with the supplier, and handling purchases and payments. A supplier frequently posts product information on web sites to promote sales. Under these conditions, users apply product categories and navigating interfaces to search for products for matching purposes among suppliers and buyers, but potential opportunities in electronic commerce on the WWW remain largely unrealized. Thus, an integrated mediator for connecting suppliers and buyers to conclude agreements is lacking. An intelligent business brokering agent between suppliers and buyers is required to share information, bid for purchases, and pay for products efficiently and effectively.

The demand side or supply side publishes most of the merchandise information on the trading web sites manually. This product information generally is presented in free text format. Buyers seek product items in the classified structure and then browse the related information to assess its cost effectiveness. When numerous web pages are returned, user time is inevitably wasted owing to the need to deal with numerous unsuitable product links. The key problem is that manual browsing is not an ideal method for buyers to identify the products that best fit their demands. Moreover, the increasing quantity of product information leads to the loss of potential match opportunities owing to a lack of designed agents with negotiation and matching abilities. The second problem is that most product information is written as text. The trading messages, including buyer and seller personal data, product information and transaction information, must be parsed from the non-structured information and transformed into a template format for use in subsequent negotiations. The third problem is that ontology is essential for agents but a useful ontology generally requires considerable user effort. Recent research has focused on how to automatically and precisely establish the ontology. The ontology provides a vocabulary for representing and communicating knowledge about the domain. Research on Ontology construction uses the Natural Language Process (NLP), machine learning, data mining and information retrieval to archive the goal about automatically ontology construction.

Innovations in integrated design depend on two main prerequisites. First, both ontology and agents must be considered to retrieve, filter and classify product information and then negotiate on behalf of users. The ontology-based agents classify products based on the computer domain ontology, and are considered essential to effective pairing. Second, the intentions and behavior of buyers and sellers must be understood. Free text information is parsed based on named entity recognition algorithms. Crucial trading information can be extracted from the subject model, including people (buyers or sellers), events (buying or selling), dealing time, delivery locations, trading entities and pricing policies.

Contributions of this study include the following. An ontology-based business agent is designed and an integrated architecture is constructed to maximize number of matches. Additional design features also are considered, including allowance for product split, an integrated trading factor called the cost model that influences trading behaviors such as delivery location, price, features, and so on. The cost model selects the best price offer from several qualified suppliers based on the integrated trading factor mentioned above.

This study presents a novel business agent architecture, in which the Business Spy Agent (BSA) automatically retrieves supply and demand information from e-commerce web sites. The Supply and Demand Analysis Mechanism (SDAM) uses Natural Language Processing (NLP) technologies to extract products and trading information. In this

design, domain ontology is used to classify the supply and demand, and divide them into many subsets to increase the probability of matching. Under this classification and matching mechanism, buyers can identify alternative merchandise resembling the original ready-to-buy merchandise using an ontology structure. Furthermore, a model of benefits is presented to handle the benefits of pairing suppliers and buyers. A “divide and conquer” algorithm is also presented to pair all sellers with buyers. During negotiation, the proposed architecture uses NLP and templates to produce negotiation texts that are mailed through e-mail. Finally, the Hidden Business Mining Mechanism (HBMM) is applied.

The Internet poses a peculiar set of trust issues that must be emphasized to minimize risk. Customers submit transactions information via the Web only when they are confident that their personal information, such as credit card numbers and financial data, is secure. The study proposes a novel business agent architecture based on an effective matching algorithm. The research efforts of this study mainly focus on the areas, such as matching algorithm, business agent and its designed architecture. Presently, a secure e-commerce infrastructure is an essential issue for the proposed architecture, while this study do not pay much attention on security issues that have been studied with a great achievement in the current stage. So existent solutions are considered to integrate into a total solution for providing a secure and trust e-commerce infrastructure in this study.

Most existing electronic marketplace adopt the trust mechanism as their underlying security protocol. The trust mechanism is a loosely confident protocol. From this protocol, a mechanism to regulate the buyers and seller on the marketplace is designed to prevent malicious behaviors such as put faked information on the web. The algorithm maintains a trust score calculated based on the success transactions and trading behaviors across web sites. If a user generates faked product information on the web that will lead to decrease his trust score owing to the bad credits. The trust model of this study meets the current requirements of buyers and sellers for making transactions on the e-commerce web site. Three commercial web sites, including eBay.com, Coolbid.com and bbs.ntu.edu.tw, establish their trading protocol mainly based on the trust mechanism.

A strict security model is not necessary for this work because cash transactions are mostly conducted based on a commercialized mechanism. PKI cryptography and digital signature technology, applied via Secure Sockets Layer (SSL) digital certificates, provide the authentication, data integrity, and privacy necessary for e-commerce. Internet payment gateway systems provide online merchants with the ability to efficiently and securely accept and process a variety of online payments from customers. Eighty-five percent of Web users surveyed reported that a lack of security made them uncomfortable sending credit card numbers over the Internet.

The rest of the paper is organized as follows. Section 3 introduces the proposed integrated business agents and its architecture. Section 4 discusses the experiment that yields a satisfactory rate of success and a simulation that shows the impacts of these options as mentioned. Finally, section 5 draws conclusions.

## 2. RELATED WORK

Until now, most business agents support services like posting, bidding, asking and others in the Internet marketplace [1]. At these service sites, a supplier can use the

“Posting System” supported by the service site, to publish product information. Users can retrieve product information using retrieval mechanisms provided by the systems. Some sites provide electronic price quotations, allowing users to receive a quote or obtain bid and ask prices. XML technology is used to exchange information and make buying more efficient.

Such sites as <http://jango.excite.com>, <http://www.firefly.com>, <http://www.onsale.com>, <http://www.personalogic.com>, <http://ecommerce.media.mit.edu/tete-a-tete>, <http://auction.eecs.umich.edu>, <http://bf.cstar.ac.com/bfare> and <http://www.ebary.com/aw> are well-known e-commerce web sites. Several descriptive theories and models have been developed to capture buying behavior, including the Andreassen model, the Bettman information-processing model and the Nicosia model. They all share six fundamental stages [2] in the buying process, including Supply and Demand Identification, Product Brokering, Merchant Brokering, Negotiation, Purchase and Delivery, Production Service and Evaluation.

To some extent, agent technology can help to automate or support buying and selling at e-commerce web sites. Purchase and Delivery as well as Production and Evaluation are two stages that are not suitable for implementation by business agents.

PersonaLogic (<http://www.personalogic.com>) filters out unwanted products by applying some specified constraints and returns a list of products. The system mainly focuses on the product-brokering stage. Like PersonaLogic, Firefly (<http://www.firefly.com>), which uses collaborative filtering, helps consumers to find appropriate products. Unlike PersonaLogic, Tête-à-Tête (<http://ecommerce.media.mit.edu/tete-a-tete>) uses techniques based on multi-attribute utility theory, to recommend complex products. The system also helps buyers and sellers at the merchant brokering and negotiation stages. On some sites, data mining techniques are used to discover patterns in customer purchasing behavior to help customers to locate other products that meet their needs. BargainFinder (<http://bf.cstar.ac.com/bf>) from Accenture is the first business agent to compare prices of online products. This proof-of-concept system looks up nine merchant Web sites to find and compare their prices. BargainFinder encounters the difficulty of merchant-blocking, meaning that some Web sites block all price requests issued from BargainFinder. Jango (<http://jingo.excite.com>) overcomes this shortcoming of BargainFinder by having product requests arise from each user’s web browser rather than a central site. This proactive agent facilitates the comparison of prices from a number of online product catalogs. Both BargainFinder and Jango emphasize the use of shopping agents to automate merchant-brokering process.

An auction is a typical negotiation process, but it is too complex or frustrating for average consumers in the non-digital world. In the digital world, OnSale (<http://www.onsale.com>) and eBay’s Auction Web (<http://www.ebay.com/aw>) are two popular web sites that sell products or goods using auction protocols. They all require customers to maintain their own negotiation strategies over a period. AuctionBot (<http://auction.eecs.umich.edu>) is a general-purpose auction server at the University of Michigan, which manages and enforces bidding according to the auction’s protocols and parameters. Kasbah (<http://kasbah.media.mit.edu>) of MIT Media Lab is an online, multiagent consumer-to-consumer transaction system that employs three negotiation strategies - anxious, cool-headed and frugal.

Most of the aforementioned agent systems do not handle Demand Identification. However, Tête-à-Tête handles Product Brokering, Merchant Brokering and Negotiation [3]. In this work, business agents are made increasingly intelligent to identify the real needs of users, using the huge amount of product information on the WWW. Such agents can help users in Supply and Demand Identification, Product Brokering, Merchant Brokering and Negotiation. Business agents help users to identify the products they need, search for available products, compare prices, negotiate and purchase. Users must handle the other processes, payment and product delivery. Business agents can help suppliers to find hidden business. This paper proposes a new architecture that uses NLP (Natural Language Processing), knowledge retrieval and ontology to build business agents to pair supply and demand. The completely new architecture includes five parts, namely Business Spy Agent, Supply and Demand Analysis, Supply and Demand Classification, Matching and Negotiation and Hidden Business Mining. An experiment shows a considerable (82%) success in pairing. A simulation reveals that options including negotiation, designated brand and split demand differently affect the match ratio.

Generally, three types of matching exist, namely exact, plug-in and relaxed matching [4]. Exact matching is the most accurate and restrictive of these forms of matching. Meanwhile, plug-in matching is less accurate than exact matching, and involves the two sides negotiating based on descriptions and constraints. Finally, relaxed matching has a weak semantic and is the least accurate but most useful form of matching. This investigation conducts experiments using exacting matching and relaxed matching to compare these two techniques in terms of their affects on matching ratio. Exact matching stresses that both descriptions must be equivalent, either literally equal or equal by item names. However, a relaxed match has a significantly weaker semantic reading than an exact match. A relaxed match does not distinguish according to whether or not two descriptions match semantically. Instead, a relaxed match decides how close the two descriptions are by returning a numerical distance value. The two descriptions match if this distance value is below a threshold value. From the structural perspective, the criteria for relaxed match closely resemble the match design developed here, which applies ontology on information classification and pairing to obtain a similarity comparison. This study adopts the generalization concept of class hierarchy in ontology to cluster different concepts into a common parent concept that represents the distance value related to the similarity. Section 4 illustrates and compares the simulation results for both the exact and relaxed matches. Exact match stands for matching without applying ontology. Meanwhile, relaxed match represents the use of ontology for matching.

### 3. BUSINESS AGENT

The business agent applies five mechanisms as shown in Fig. 1, namely Business Spy Agent (BSA), Supply and Demand Analysis Mechanism (SDAM), Supply and Demand Classification Mechanism (SDCM), Matching and Negotiation Mechanism (MNM) and Hidden Business Mining Mechanism (HBMM). BSA can automatically capture the supply and demand information from electronic market web sites [5-9]. The supply and demand information can be analyzed using the following mechanism to construct a supply and demand list. SDAM uses NLP to analyze the information aggregated by BSA

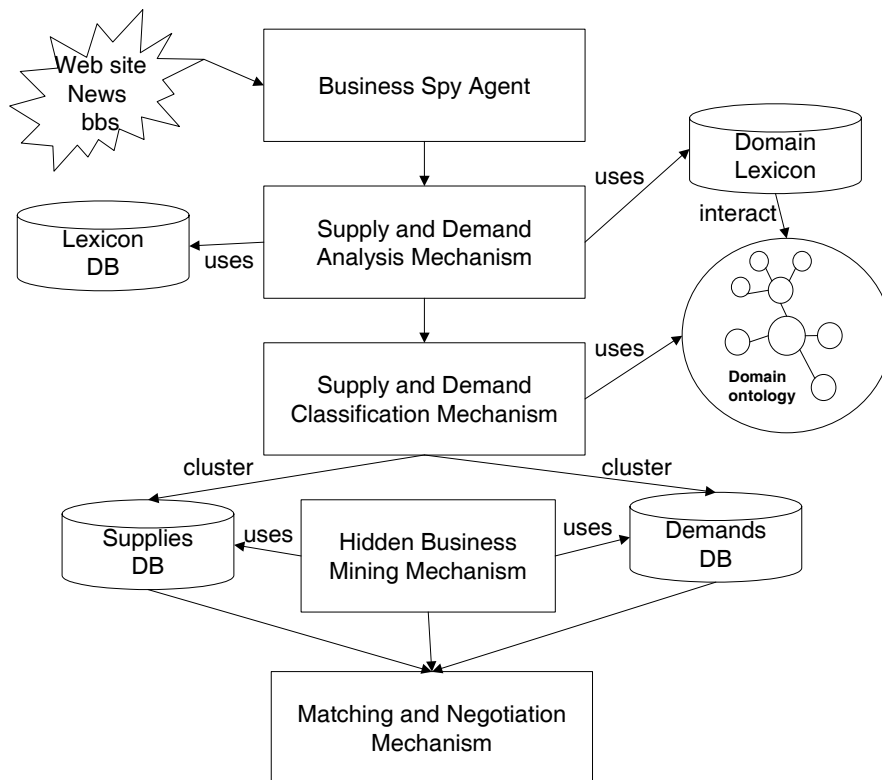


Fig. 1. System architecture.

and construct supply and demand lists. SDCM uses the ontology-based classification algorithm to classify information analyzed by the preceding mechanism. The classification algorithm is based on domain-ontology [10]. MNM uses a benefit model and a pairing algorithm to match up supply and demand. HBMM uses a data mining algorithm to mine trading history for hidden business opportunities.

### 3.1 Business Spy Agent

Trading on the Internet involves several activities. Conventionally, messages from the supply side (the supplier) or the demand side (the buyer) are posted. The poster then waits for a response or posts more messages, and performs searches to find matches. Trading is time-consuming. In the architecture presented here, the BSA automatically obtains supply and demand information from the Internet. A knowledge base stores the URLs (such as <http://www.bid.com.tw>), BBS (e.g., <http://bbs.ntu.edu.tw>) and news of nationally well known electronic markets sites. Such sites include much supply and demand information that can be captured by the Business Spy Agent presented here.

### 3.2 Supply and Demand Analysis

The SDAM analyzes and filters any redundant information captured by BSA, following which useful supply and demand information is formatted. The information captured by BSA is in two different text formats. One is in free text format and the other in semi-structured. The information formatted as free text is obtained from the Bulletin Board System (BBS) or the news. NLP technology is required to handle the non-structured information. For example, the supply and demand information in the Bulletin Board System (BBS) is presented in brief and free-style sentences, as depicted in Fig. 2.

Sell Pentium 586 computer Intel Celeron 850A Processor Asus main-board 6178 Hard-disk IBM 40GB Main memory 128MB Asus 52X CD-ROM Price \$6500 e-mail: u3261143@cc.ncu.edu.tw
---

(a)

Author: RETER.bbs@bbs.nsysu.edu.tw (Try To Remember), Board: market Subject: buy PCMCIA network card from: Formosa bbs (Fri Sep 28 10:39:58 2001) Forward Site: cis_nctu!news.cis.nctu!netnews.csie.nctu!news.civil.ncku!news.nsysu!Fo from: bbs.nsysu.edu.tw wanted PCMCIA network card Plz reply with prices and product information
---

(b)

Fig. 2. (a) Supply information, (b) Demand information.

The SDAM uses the NLP technology to analyze free text information and present useful “supply information” and “demand information”. The analysis includes the three following steps.

- (1) Determining supply and demand. This step separates the supply from the demand by using a corpus of trading concepts, such as “sell,” “buy,” “wanted,” “ask for,” and others.
- (2) Determining product information. After the supply and demand information is obtained, product information is determined. This paper addresses processing on computer hardware peripherals. The domain ontology of computer hardware peripherals is constructed, and then ontology and NLP technology are applied to extract product information. Domain ontology and product information can be used later by SDCM in pairing.
- (3) Determining transaction information. The transaction information includes delivery time, location, contact information, and other related information.

In other cases, semi-structured information, obtained from electronic market sites [11], is given specific tags (tables or lists) to specify whether it is supply or demand information. For example, Bid (<http://www.bid.com>) presents semi-structured supply information as in Fig. 3 (a).

CPU	Intel Celeron 1G
Main board	ASUS POLO
Memory	128MB PC-133
CD-ROM	24X CD-ROM
Hard Disk	20.4GB/5400 rpm ATA-100
Soft Disk	1.44MB
VGA	128-bit 32MB 3D (4X)
Sound card	AC97
Intenet card	Realtek RTL8100 10/100M

(a)

```

<table>
<tr><td>CPU</td><td >Intel Celeron 1G</td></tr>
<tr><td>Main board</td><td>ASUS POLO </td></tr>
<tr><td>memory</td><td>128MB PC-133</td></tr>
<tr><td>CD-ROM</td><td>24X CD-ROM</td></tr>
<tr><td>Hard Disk</td><td>20.4GB/5400 rpm ATA-100</td></tr>
.
.
.</table>

```

(b)

Fig. 3. (a) Supply information presented in table from electronic market site Bid, (b) The HTML tag for (a).

Structured supply and demand information can be easily retrieved. Different structures require different ways to convert information. For example, information presented in a table, for example as shown in Fig. 3 (b), can be retrieved by parsing the HTML tag `<table>` and `</table>`. All supply information can be retrieved with html tags `<tr>`, `</tr>`, `<td>` and `</td>`. Various electronic market sites use different HTML tags to present the information. Consequently, each electronic market site uses its specific method of retrieval to convert specific structures.

Presently, the supply and demand information obtained from agents surfing the web merely contains free text without script codes, such as JavaScript and ASP and so on. The free text contains sufficient information for parties on both sides. The integrated business agent operates primarily using this information format in the current design. This study does not consider information extracted from dynamic homepage. Generally,



these script codes contain syntactic rules and semantic information regarding web pages. This information significantly helps designers of the agents to understand the web pages. This study leaves dynamic web page processing for future research.

### 3.3 Supply and Demand Classification

After the supply and demand information is identified by SDAM, it is classified by SDCM to accelerate pairing of supply and demand information. Supply information is matched with demand information in the same class to complete the supply and demand pairing. This work adopts an ontology-based classification algorithm to classify supply and demand information [12]. Each ontological node is considered to be a class, and supply and demand information are mapped onto ontological nodes, as depicted in Fig. 4.

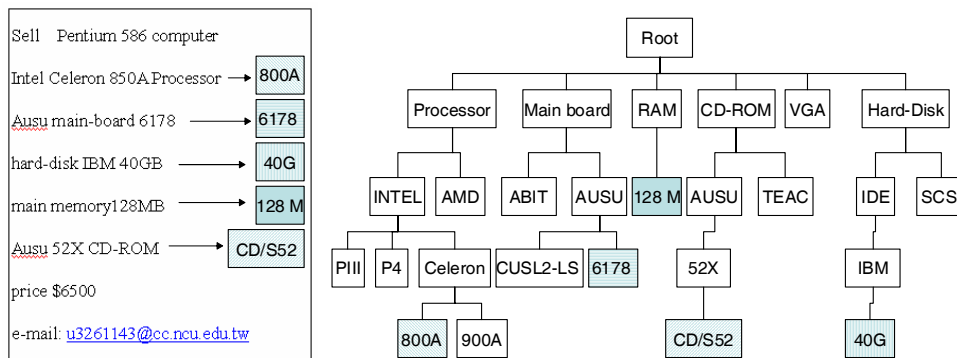


Fig. 4. Supply and demand information mapped onto ontological nodes.

Another merit of the classification algorithm is that it can divide supply and demand information into its corresponding sub-supply and sub-demand parts. For example, the demand for computers can be divided into several sub-demands that including that for processors, main boards, hard disks, main memory and other parts. The sub-demand information can be matched to sub-supply information and pairing can be completed.

The domain ontology for computer peripherals is built manually in the current stage. The categories of the second level are classified based on the items of the root class in the first level. Meanwhile, the nodes in second level themselves are also other classes in the ontology tree. The ontology is constructed based on the above rules accordingly. The lowest level of the tree comprises the instances of the above-mentioned classes in upper levels. The other ontology for various products can be created according to the above notions and procedures. Automatic ontology construction is always a research topic, but the accuracy of the results constructed by automatic algorithms requires improvement.

### 3.4 Matching and Negotiating

#### Pairing supply and demand

After classification, supply is paired with demand. The three types of supply and demand pairings are described and a Divide and Conquer Pairing Algorithm is proposed.

Type 1: Only one supplier can satisfy the demand (one-to-one pairing), as shown in Fig. 5. This is the simplest type of pairing. Negotiation follows.

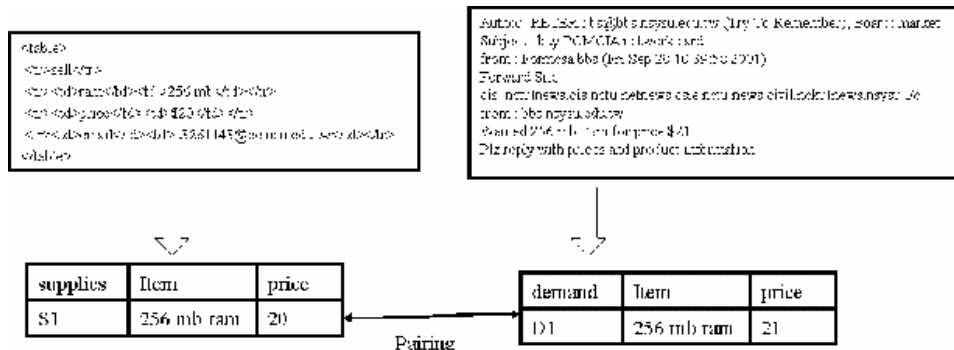


Fig. 5. One-to-one pairing.

Type 2: Many suppliers can satisfy the demand (many to one pairing) as shown in Fig. 6. The best supplier is paired with the demand.

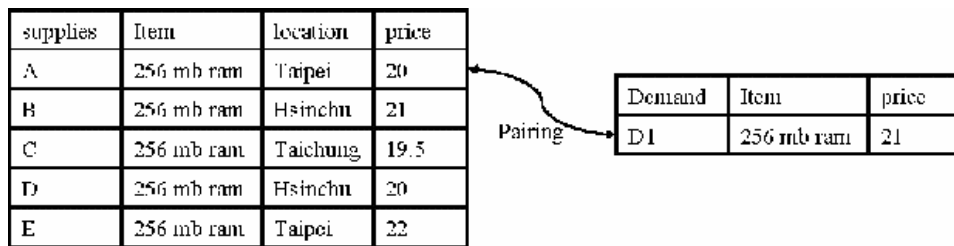


Fig. 6. Many-to-one pairing.

The benefit model selects the best supplier based on the price. The lower the total sub-demand cost, the higher the revenue for the supplier. The benefit model is as follows.

$$B_i = \begin{cases} \text{Min}(\sum_{j \in \text{sub-demand of } i} B_{ij}, \text{price of supply pair of } D_i + C_i), & \text{if } D_i \text{ can be decomposed into sub - demands } D_{ij} \\ \text{price of supply pair of } D_i + C_i, & \text{if } D_i \text{ can't be decomposed into sub - demands} \end{cases} \quad (1)$$

where  $B_i$  is the benefit of demand  $D_i$  and  $B_{ij}$  is the benefit of the sub-demand  $D_{ij}$  of demand  $D_i$ .  $C_i$  is the variable cost, including that of freight, packaging and postage. The variable cost is recorded by referring real cases. For example, Table 1 presents the costs of freight.

**Table 1. The variable costs of freight.**

Source	Destination	Freight (\$)
Taipei	Taipei	3
Hsinchu	Hsinchu	4
Taichung	Taichung	5
Taipei	Hsinchu	7
Taipei	Taichung	10
Hsinchu	Taichung	7

Type 3: Different suppliers together satisfy the one demand of one consumer, as shown in Fig. 7. The demand can be decomposed into many sub-demands. Each sub-demand may be paired to one supplier. A Divide and Conquer pairing algorithm is used to achieve the goal, and is described in Fig. 8.

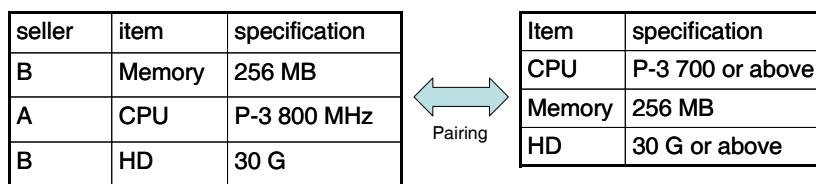


Fig. 7. Different suppliers together meet the demand.

```

Function Pair ( $D_i$ )
  For demand  $D_i$  do begin
    If  $D_i$  can be divided into sub-demands according to domain ontology then
      Divide  $D_i$  into sub-demands  $\{D_{i1}, D_{i2}, \dots, D_{ij}\}$ 
      Set Benefit  $B_i = 0$ 
      For each sub-demand  $D_{ij}$  do begin
        Set Benefit  $B_{ij} = \text{Pair}(D_{ij})$ 
        If  $D_{ij} = 0$  then
          Set Benefit  $B_i = 0$ 
          Exit For
        Else
           $B_i = B_i + B_{ij}$ 
        End If
      End For
    End If
    If  $D_i$  can be paired with supply without dividing  $D_i$  then
      Set Benefit  $C_i =$  then Benefit of pairing among  $D_i$  and supply
    Else
      Set Benefit  $C_i = 0$ 
    End If
    If  $C_i <> 0$  and  $B_i <> 0$  then
      Return  $\text{Min}(C_i, B_i)$ 
    Else
      Return  $\text{Max}(C_i, B_i)$ 
    End If
  End For
End Function
    
```

Fig. 8. Divide and Conquer algorithm for supply and demand pairing.

In this Divide and Conquer algorithm, if the demand  $D_i$  can be divided into sub-demands according to the domain ontology, then  $D_i$  is divided into sub-demands  $\{D_{i1}, D_{i2}, \dots, D_{ij}\}$ . For each sub-demand,  $D_{ij}$ , the algorithm will call the  $\text{Pair}(D_{ij})$  function to obtain the benefit  $B_{ij}$  of the pairing candidate with  $D_{ij}$ . If the demand  $D_i$  can be paired with a supplier without dividing it into sub-demands, then the benefit  $C_i$  of pairing  $D_i$  is calculated. Finally,  $C_i$  and  $\sum B_{ij}$  are compared to satisfy the benefit model (1) described in Fig. 8. The Divide and Conquer algorithm handles the pairing efficiently and effectively.

### Negotiation

After a candidate pairing of supply and demand is generated, negotiation begins. The architecture presented here includes a hiding negotiation mechanism and uses auto-mailing technology. Fig. 9 shows a flowchart of the negotiation process.

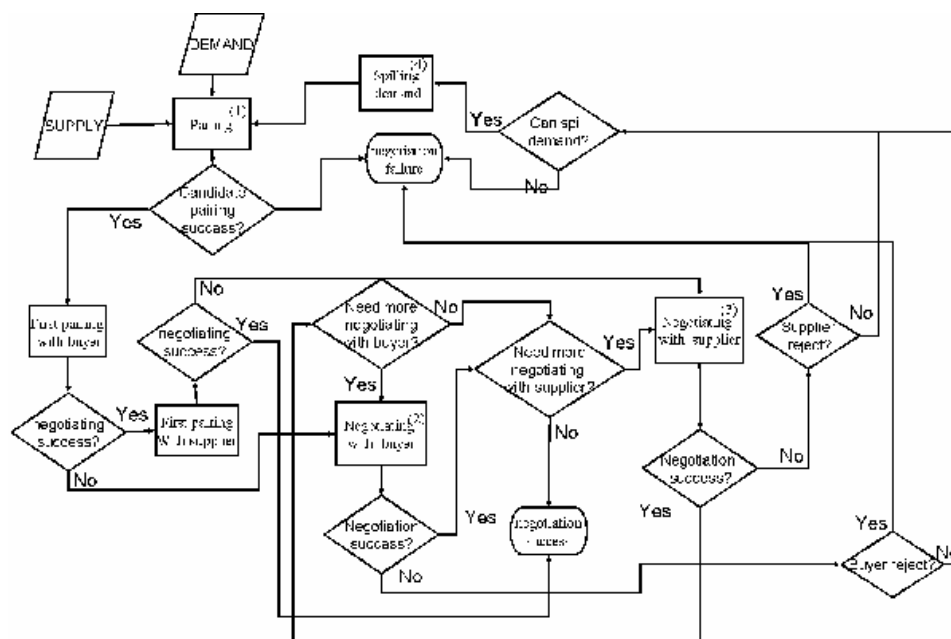


Fig. 9. Business negotiation flowchart.

Before negotiation, the supply is paired to the demand <sup>(1)</sup>. The Divide and Conquer pairing algorithm performs pairing, as shown in Fig. 8. The first pairing of buyer and supplier is then archived by a negotiation mechanism and auto-mailing technologies. If the first pairing with a buyer fails, negotiation with another buyer <sup>(2)</sup> is initiated. Similarly, the negotiation with a supplier <sup>(3)</sup> begins when the first pairing with the supplier fails. Supply and demand information available on the Internet (WWW, BBS, News etc.) includes statistical data (on products, prices, times and places). The buying and selling negotiating templates are established in advance. The negotiation mail is developed using

a NLP technique and a suitable template mail is selected. Next, the mechanism sends negotiation mail to a supplier and a buyer to perform successful negotiation. Fig. 10 shows the buying negotiation template.

I want to buy \_\_\_\_\_

Your selling price is \_\_\_\_\_

Could I bid the product for \_\_\_\_\_

Could we finish the deal on \_\_\_\_\_ at \_\_\_\_\_

(a)

I want to buy Pentium-2 desktop computer

Your selling price is \$50

Could I bid the product for \$48

Could we finish the deal on Ku-Huw market at 19/12/2000 pm. 6

(b)

Fig. 10. (a) Buying negotiation template, (b) Buying negotiation text.

In Fig. 10 (b), the product information including selling price, trading place, trading time are determined by SDAM as described in section 3.3. Negotiation with the buyer may not be successful because of issues of price, location, or time. Negotiation with the supplier is performed using the same mechanism and technology as that with the buyer. Occasionally, the supplier on an electronic market site does not negotiate because the price is fixed or some other criteria are imposed. In these cases, the demand is extended according to domain ontology<sup>(4)</sup>. The demand is extended to the ontological sibling node where the demand is adjusted and the appropriate negotiation proceeds. For example, a demand for the “Intel Celeron 850A” CPU can be changed to that for a “Intel Celeron 900A” according to the ontology in Fig. 4.

### 3.5 Hidden Business Mining (HBM) Mechanism

In the preceding section, negotiation occurs when various demands exist. Sometimes, one side may have to wait for a long time to buy or sell, or may never be able to buy or sell. This architecture uses the HBM (Hidden Business Mining) mechanism. The HBM mechanism uses data mining technology to find a hidden business and do the promotion [13]. The algorithm that applies the rules of association is used accordingly [14-16]. HBM mines the association rules using a history transaction log. If a rule, “buy printer => buy printer paper” exists, then the system retrieves users who had bought printers but had not purchased printer paper, and automatically promotes the printer paper to the targeted users.

## 4. RESULTS AND DISCUSSION

This section presents the results of one experiment and two simulations. The experiment conducted here focuses on the matching algorithm proposed to demonstrate the feasibility in the real world. Meanwhile, one simulation assesses the impacts of using ontology and matching algorithm. Another simulation presents the effects of match ratio after using different options. This study describes the simulation model and analyzes some performance results of business agents in the following. Additionally, the match ratio of various designs also is presented.

### 4.1 Experimental Background

This study established a prototype system for assessing and verifying the proposed integrated agent architecture. This system is a Win32 console program without a URL link that executes following the designed steps. A crawler agent first gathers the web pages from three electronic market sites. Next, a parser uses the keyword approach to parse the free text contained in the web pages into structural information that fits the supply and demand templates. In other cases, semi-structured information, such as HTML files, is converted into structural information by processing the HTML tags. The classifier program refers to the domain ontology and then categorizes the parsed supply and demand information. The prototype then pairs supply and demand based on the proposed benefit model from the classified databases and negotiates suppliers and buyers for matching the deals.

Two famous electronic market web sites and one Bulletin Board System (BBS) are selected. Table 2 lists the supplies and demands in these electronic markets. The www.bid.com.tw site supports two formats for supply information - table format and free text format. The www.coolbid.com site supports only free text format for supply information. These two sites provide supply information without permitting demand information. The bbs.ntu.edu.tw site includes the "salecomp" board that presents a very large amount of supply and demand information in free text format. A business agent is built here to search for and collect supply and demand information from these three web sites; the agent negotiates with both sides and finally finds a pairing.

**Table 2. The supply and demand information.**

Electronic market	http://www.bid.com.tw	http://www.coolbid.com	bbs.ntu.edu.tw
Supply count	150	100	200
Total supplies	450		
Demand count	X	X	400
Total demands	400		

### 4.2 Experimental Results and Analysis

Table 3 presents the experimental results. In the candidate pairing generation stage, 356 of 400 demands are successfully matched to candidate suppliers. Almost 89% de-

**Table 3. Experimental results.**

	Success count	Failure count	Percentage of successes
Candidate pairing generation	356	44	89%
First negotiation with supplier and buyer	153	203	43%
More negotiation with buyer	96	57	63%
More negotiation with supplier	63	33	65%
Split demand	5	28	15%
Total	328	72	82%

mands are paired. In the first negotiation between supplier and buyer, 153 demands are negotiated successfully while 203 demands must be negotiated further. During this further negotiation, only 96 demands (63%) are negotiated successfully and 57 demands must be further negotiated. Most of the failures in negotiations are due to price. The buyer typically asks for a reduced price. In only 65% of cases are negotiations successful. Most of the failures occurred when using [www.bid.com](http://www.bid.com) and [www.coolbid.com](http://www.coolbid.com) for fixed prices for products. Negotiations with these two supply sites always fail. When the demand is split, 15% of cases are successful because the buyer does not change his or her original demands. The system has an overall success rate of 82%.

#### 4.3 Simulation and Analysis

##### Impacts of using ontology and matching algorithm

This study establishes a supply and demand environment to assess the impacts of match ratio for different settings, including the adoption of ontology, the split option, and matching types such as exact and relaxed matching. The execution flow of the simulation model follows the flowchart in Fig. 9. This study creates product information on 450 products for sale by random number generator. A random five digit number indicates the composite detail of the product being sold. Each of the five digits represents the composite class identifiers that are compared with the descriptions of buyers. For example, the composite classes can be product brands, CPU model, motherboard manufacturer, RAM size, hard disk capacity, and so on. Besides, the dominant effect on the match comparison of the classes varies decreasingly from the rightmost to the leftmost digit. This behavior indicates most buyers make decisions based on only two or three decision factors. Specifically, buyers may decide to buy if the brand and CPU model fit their requirements, or alternatively they may simply be guided by the product brand. This study used the number of comparison classes, varying between one and five, as the experimental parameter. From the above description, this study generates random numbers to represent buyers. Furthermore, this study simulates the classification behavior involved in using domain ontology by establishing rules for converting each digit into another class identi-

fier, ranging from 0 to 2, as defined in the ontology structure. For instance, the agent can transform P4 into Intel, where Intel belongs to the class known as Processor.

The negotiations of business agents proceed in a multi-stage fashion. Initially, the agent generates potential pairings between sellers and buyers. The agent then acts on behalf of the user to negotiate with both sides to achieve a deal. If the negotiation fails, the agent follows the algorithm mentioned in section 3 and negotiates with the buyer or seller individually. This study employs a random number as the basis for judging which side (buyer or seller) to negotiate with first. Additionally, the merchandise can be assumed to be split into several components to maximize the potential business benefits. Following the successful split, the negotiation process repeats the above loop.

Trading behavior related to decision-making is extremely complicated in the real world, including cost issues, personal preferences, delivery locations and so on. This study elaborates the simulation to simplify the agents designed to deal with the details of matching, for example exact match and relaxed match. This work does not consider all three types of pairings that appear during the initial stage, including one-to-one pairing, many-to-one pairing, and split demands. This study only considers and assesses two types of pairing, namely one-to-one pairing and split demands. Many-to-one pairing involves not only matching criteria but also cost and other issues. This simulation conducted here thus concentrates on the matching behavior of buyers and sellers, and price and cost and left to be negotiated by the two sides.

Table 4 lists the simulation parameters and corresponding descriptions. *mr* then denotes the matching ratio calculated based on total demand. The number of demands, listed in Table 4, varies with simulated needs. Meanwhile, the split option is always set to a TRUE value. The number of dominant classes for comparison ranges from 1 to 5. To achieve an exact match, this work sets *nc* to 1, 2, and 3. Meanwhile, to achieve a relaxed match, this work sets *nc* to 1, 3, and 5. A total of six cases thus are assessed. For each of these cases, a match type and *nc* value are selected for all the demand values in each of these cases.

**Table 4. List of parameters and descriptions for match evaluation.**

Parameter	Description	Value
<i>ns</i>	Number of supplies	Create 450 product information for selling
<i>nd</i>	Number of demands	50, 100, 200, 300, and 400
<i>np</i>	Number of pairings	(Blank)
<i>nm</i>	Number of match	(Blank)
<i>mr</i>	Match ratio based on the total demands	$nm/nd$
<i>nc</i>	Number of dominated classes for comparison	1, 2, 3, 4, and 5
<i>sd</i>	Split demand	Yes

Figs. 11 and 12 illustrate the effects of using ontology on different *nc* settings, the setting values are 1 and 3 individually. This investigation notes that the improvement in the application of ontology on matching algorithms increases with the number of comparison classes. Additionally, relaxed match performance, indicated by the dotted line in



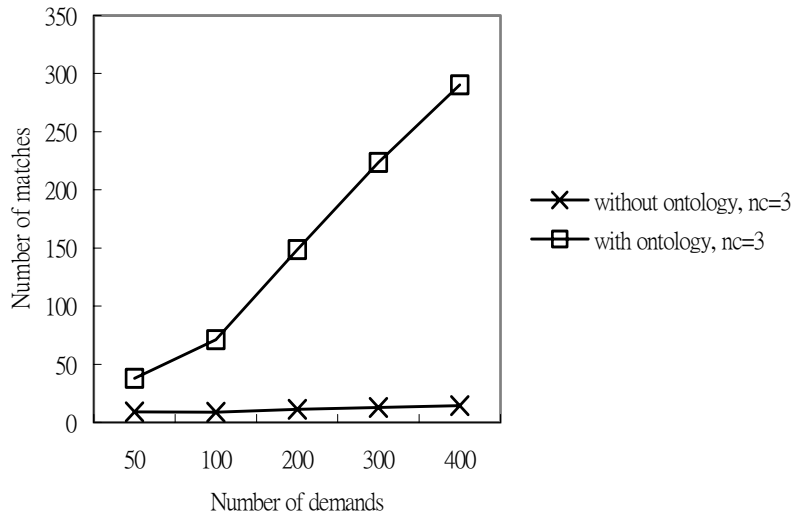


Fig. 11. The effects of using ontology for nc = 3.

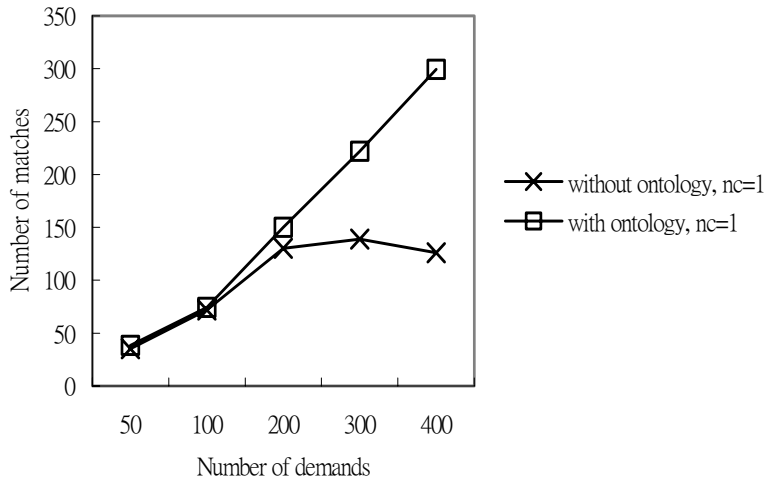


Fig. 12. The effects of using ontology for nc = 1.

both figures, display similar results. This phenomenon indicates that the design using ontology maintains a constant performance regardless of *nc* value. From Fig. 11, the number of matches does not increase as the number of demands grows from 200 to 400.

Figs. 13 and 14 illustrate the effects of match ratio after using the match algorithm. The dotted line denotes the match ratio without applying the match algorithm, while the solid line indicates the improvement achieved after using the match algorithm. For the exact match illustrated in Fig. 13, the two match ratio values converge closely as the number of demands increases. Meanwhile, for the relaxed match shown in Fig. 14, the difference between the two match ratios improves steadily as the number of demands increases.

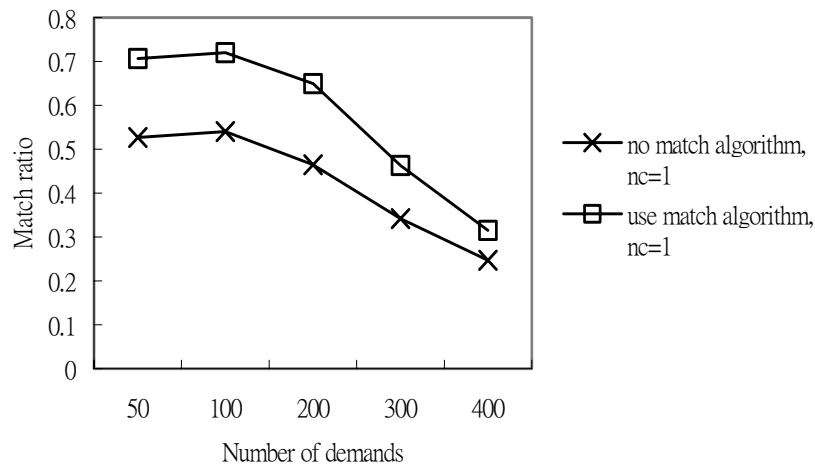


Fig. 13. The effects of using matching algorithm for  $nc = 1$ .

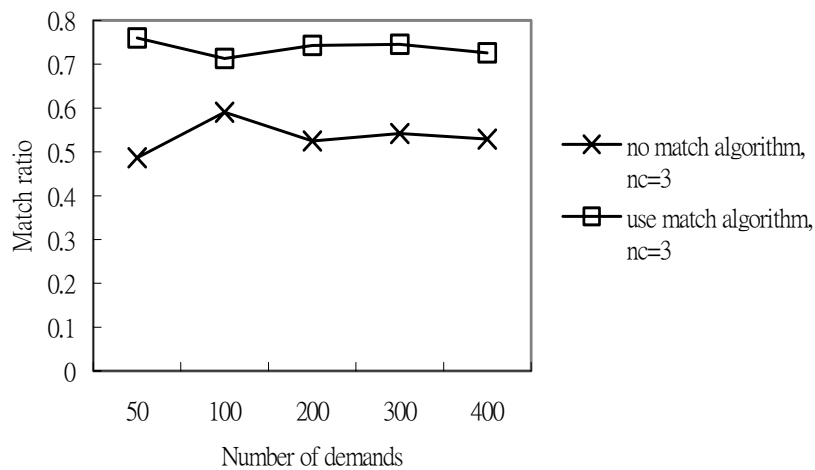


Fig. 14. The effect of using matching algorithm for  $nc = 3$ .

Fig. 15 displays the overall match ratio for the six cases. These cases involving ontology always maintain a satisfied match ratio regardless of number of demands. Regarding cases without ontology, the match ratio reduces rapidly with increasing demand number.

#### Effects of match ratio for different options

Complex situations that occur in the real world are simulated. Table 5 lists the parameters used in the simulation. The options include five possible cases, as listed in Table 6. In Table 5, the first parameter refers to the number of different supplied product classes. The second parameter represents the number of products of each type available; the products may be of a different brand. The number of demanded products ranges from

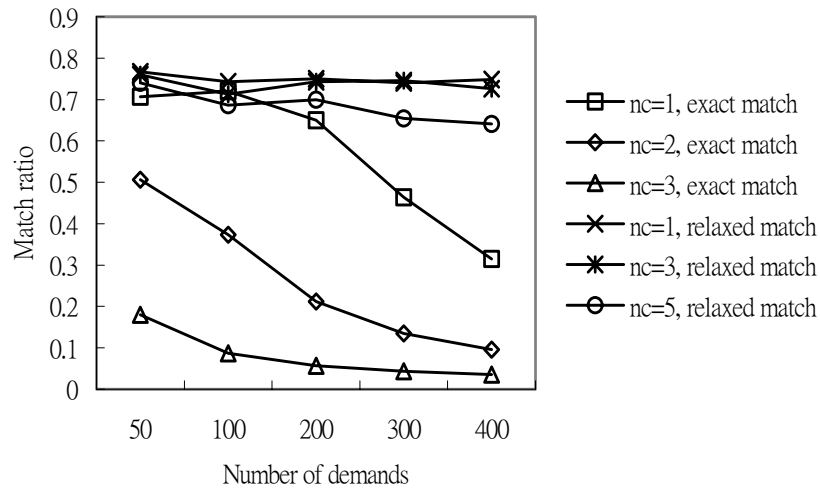


Fig. 15. The overall match ratio for the six cases.

Table 5. Parameters in the simulation.

Description	Value
Number of product types	50
Number of products for each type	5
Total number of products to supply	250
Number of products for demand	50, 100, 150, 200, 300, 400, 500

Table 6. Options for the simulation.

	Negotiable	Designated Brand	Split Demand
Case 1	Y	Y	Y
Case 2	Y	N	Y
Case 3	Y	N	N
Case 4	N	Y	Y
Case 5	N	N	Y

50 to 500 to determine the effects of the demand on the match ratio. In Table 6, five different cases constitute a complicated trading situation. The ‘negotiable’ parameter indicates whether the price is negotiable, ‘designated brand’ states whether the buyer will purchase a particular brand of products. Negotiation addresses the price. ‘Split demand’ refers to the option of dividing the demand to make the deal more possible.

Fig. 16 presents the simulation results in terms of match ratio for different options. The x-axis represents the number of demanded products and the y-axis indicates the match ratio between supply and demand. Some interesting phenomena, listed below, are observed in the different test cases. They are

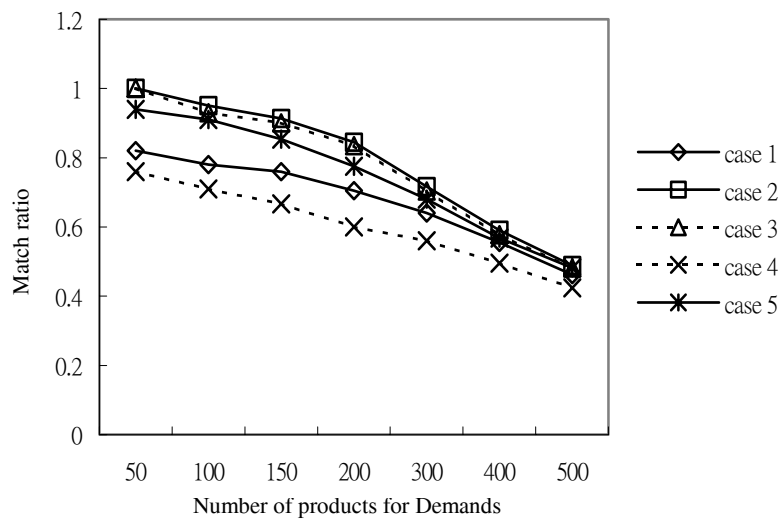


Fig. 16. Results of the simulation.

- Case 1 vs. case 2 and case 4 vs. case 5 show that the ‘designated brand’ always affects the match ratio. Not specifying a specific brand of product increases the match ratio.
- Case 2 vs. case 3 shows that ‘split demand’ has a negligible effect when the brand is not specified.
- In case 1 vs. case 4 and case 2 vs. case 5, the ‘negotiable’ option has a moderate effect.

According to the simulation results, allowing negotiation between supplier and buyer greatly increases the chance of making a deal.

## 5. CONCLUSIONS

This work proposes an integrated business agent architecture that supports four buying processes, including Identifying Supply and Demand, Brokering Products, Merchant Brokering, and Negotiating. The BSA automatically extracts supply and demand information from e-commerce web sites. SDAM uses NLP technologies to extract product and transaction information. Domain ontology is used to classify supply and demand information and divide it into subsets. A benefit model is also presented to pair sub-supplies with sub-demands, and a divide and conquer algorithm is presented to handle the pairing of all supply and demand. During negotiation, the architecture uses NLP and templates to produce a negotiation text that is mailed via the mailing mechanism. Finally, HBMM uses data mining technology to find hidden business. Experimental results show that by using the proposed architecture the business agent performed very well.

The simulation results show the feasibility of the proposed architecture for business agents. For experimental purposes, we construct agents to surf product information only on three commercial web sites, including eBay.com, Coolbid.com, and bbs.ntu.edu.tw. Presently, the study focuses on verifying the effective agent designs, such as supply and

demand information analysis, information classification using ontology, matching, and negotiation. In the future, the business agent architecture will be introduced into software vendors which concentrate in e-commerce areas to deploy the improved match mechanism on real systems for increasing the benefits of trading markets.

### ACKNOWLEDGMENT

This research was supported by the Software Technology for Advanced Network Application project of Institute for Information Industry and sponsored by MOEA, R.O.C.

### REFERENCES

1. A. Moukas, "Amalthaea: information discovery and filtering using a multi-agent evolving ecosystem," *International Journal of Applied Artificial Intelligence*, Vol. 11, 1997, pp. 437-457.
2. P. Maes, R. Guttman, and A. Moukas, "Agents that buy and sell," *Communications of the ACM*, Vol. 42, 1999, pp. 81-91.
3. R. Guttman and P. Maes, "Agent-mediated integrative negotiation for retail electronic commerce," in *Proceedings of the Workshop on Agent-Mediated Electronic Trading (AMET '98)*, 1998, pp. 70-90.
4. K. Sycara, S. Widoff, M. Klusch, and J. Lu, "LARKS: dynamic matchmaking among heterogeneous software agents in cyberspace," *Journal of Autonomous Agents and Multi-Agent Systems*, Vol. 5, 2002, pp. 173-203.
5. A. Chavez, D. Dreilinger, R. Guttman, and P. Maes, "A real-life experiment in creating an agent marketplace," in *Proceedings of the 2nd International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM '97)*, 1997, pp. 160-179.
6. R. Doorenbos, O. Etzioni, and D. Weld, "A scalable comparison-shopping agent for the World-Wide Web," in *Proceedings of the 1st International Conference on Autonomous Agents (AGENTS '97)*, ACM Press, 1997, pp. 39-48.
7. S. Kuke, L. Spector, D. Rager, and J. Hendler, "Ontology-based web agents," in *Proceedings of 1st International Conference on Autonomous Agents (AA '97)*, 1997, pp. 59-66.
8. H. Liebermann, "Autonomous interface agents," in *Proceedings of ACM Conference on Computers and Human Interaction (CHI '97)*, 1997, pp. 67-74.
9. A. Moukas, R. Guttman, and P. Maes, "Agent-mediated electronic commerce: an MIT media laboratory perspective," *International Journal of Electronic Commerce*, Vol. 4, 2000, pp. 5-22.
10. S. Bergamaschi, S. Castano, S. Vermercati, S. Montanari, and M. Vincini, "An intelligent approach to information integration," *International Conference on Formal Ontology in Information Systems (FOIS '98)*, 1998, pp. 253-268.
11. J. Fuernkranz, T. Mitchell, and E. Riloff, "A case study in using linguistic phrases for text categorization on the WWW," in *Proceedings of AAAI/ICML Workshop Learning for Text Categorization*, 1998, pp. 5-12.

12. A. Maedche and S. Staab, "Ontology learning for the semantic Web," *IEEE Intelligent Systems*, Vol. 16, 2001, pp. 72-79.
13. G. Adomavicius and A. Tuzhilin, "Using data mining methods to build customer profiles," *IEEE Computer*, Vol. 34, 2001, pp. 74-82.
14. R. Srikant, Q. Vu, and R. Agrawal, "Mining association rules with item constraints," in *Proceedings of 3rd International Conference on Knowledge Discovery and Data Mining (KDD '97)*, 1997, pp. 67-73.
15. S. W. Changchien and T. C. Lu, "Mining association rules procedure to support on-line recommendation by customers and products fragmentation," *Expert Systems with Applications*, Vol. 20, 2001, pp. 325-335.
16. I. H. Meng, W. P. Yang, W. C. Chen, and L. P. Chang, "Data mining mechanisms in knowledge management system," in *Proceedings of 4th International Conference on Enterprise Information Systems (ICEIS '02)*, 2002, pp. 399-404.



**I-Heng Meng (蒙以亨)** was born on 1963 in Hsinchu, Taiwan, R.O.C. He received the B.S. and M.S. degree in Applied Mathematics from National Chung Hsing University in 1986 and 1988. He received the Ph.D. degree from Institute of Computer and Information Science at National Chiao Tung University in 2003. His research is focused on database theory, object-oriented database, knowledge management, and semantic processing. He is currently a senior manager in Advanced E-Commerce Institute (ACI), Institute for Information Industry (III).



**Shin-Mu Tseng (曾新穆)** received his B.S. and M.S. from Department of Computer Science and Information Engineering at National Chiao Tung University, Taiwan, R.O.C., in 1988 and 1990, respectively. He received the Ph.D. degree from Institute of Computer and Information Science at National Chiao Tung University in 1997. During January 1998 and July 1999, he was an invited postdoctoral research fellow in Computer Science Division of University of California, Berkeley, U.S.A. Since August 1999, he has been on the faculty of Department of Computer Science and Information Engineering at National Cheng Kung University, Taiwan. Dr. Tseng has a wide variety of research specialties covering data mining, Internet technology, bioinformatics and real-time systems. He has published numerous research papers in referred journals and international conferences. He is a member of the Association for Computing Machinery, IEEE and honorary member of Phi Tau Phi Society. He has served as program committee for a number of international conferences including ACM SIGKDD Workshop on Data Mining in Bioinformatics (BioKDD), 2003, Pacific-Asia Symposium on Knowledge Discovery and Data Mining (PAKDD), 2002, International Conference on Real-Time Technology and Applications (RTAS), 2001, etc.



**Wei-Pang Yang (楊維邦)** was born on May 17, 1950 in Hualien, Taiwan, R.O.C. He received the B.S. degree in Mathematics from National Taiwan Normal University in 1974, and the M.S. and Ph.D. degrees from the National Chiao Tung University in 1979 and 1984, respectively, both in Computer Engineering. Since August 1979, he has been on the faculty of the Department of Computer Science and Information Engineering at National Chiao Tung University, Hsinchu, Taiwan. In the academic year 1985-1986, he was awarded the National Postdoctoral Research Fellowship and was a visiting scholar at Harvard University. From 1986 to 1987, he was the Director of the Computer Center of National Chiao Tung University. In August 1988, he joined the Department of Computer and Information Science at National Chiao Tung University, and acted as the Head of the Department for one year. Then he went to IBM Almaden Research Center in San Jose, California for another one year as visiting scientist. From 1990 to 1992, he was the Head of the Department of Computer and Information Science again. His research interests include database theory, database security, object-oriented database, image database, and Chinese database retrieval systems. Dr. Yang is a senior member of IEEE, and a member of ACM. He was the winner of the 1988, and 1992 AceR Long Term Award for Outstanding M.S. Thesis Supervision, 1993 AceR Long Term Award for Outstanding Ph.D. Dissertation Supervision, and the winner of 1990 Outstanding Paper Award of the Computer Society of the Republic of China. He also obtained the Outstanding Research Award of National Science Council of the Republic of China.