CHAPTER 6: MULTI AGENT ENTERPRISE KNOWLEDGE MANAGEMENT SYSTEM FOR E-COMMERCE: ARCHITECTURAL DESIGN AND IMPLEMENTATION

The architectural design of MAEKMS for IT industry and healthcare services has already been discussed in the previous chapter. This chapter deals with the architectural design and implementation of MAEKMS for E-commerce in detail.

Many researchers have contributed towards designing of multi agent systems for different applications. The models vary with the applications for which they are designed. In our discussion in Chapter 2 i.e., MAS for e-commerce the need to appreciate the incorporation of knowledge management perspective into the existing MAS for e-commerce has already been highlighted. An algorithm using case based reasoning (CBR), knowledge beads and fuzzy c-means clustering (FCM) is proposed for e-commerce. Further, statistical techniques like Linear and Multiple regression models facilitate the analysis and interpretation of the sales data for better planning of business strategies by e-commerce sites. CBR was introduced briefly in chapter 2. The subsequent sections will elaborate on knowledge beads, Fuzzy c-means clustering (FCM) and then proceed towards the MAEKMS for e-commerce. The following sections will also compare the system with the Multi Agent System discussed in chapter 2 and the existing e-commerce systems proposed by different researchers.

6.1 Knowledge Beads
There has been a lot of research work done in the field of multi agent knowledge management systems. In particular, the main effort has been put on the ontology implementation and knowledge representation [191]. Automated negotiation is an important type of interaction in systems composed of autonomous agents [192 and 193]. Different agent’s knowledge models have been built up for automated negotiations [194, 195]. Amongst the most existing negotiation systems, the details on how to extract, integrate, and utilize the knowledge is still vague. To manage the business data in different forms and turn it into effective knowledge, there are certain steps that need to be followed:
1. Identify different forms of knowledge
2. Construct a methodology for manipulating the knowledge and
3. Integrate knowledge into agent negotiation process.

The first step is to formulate two main forms of knowledge, namely, general knowledge and meta-knowledge. The second step gives the necessary methods to manipulate a knowledge base and the knowledge about a knowledge base, i.e., the meta-knowledge. The ultimate resolution is the fusion of knowledge carried by knowledge bases into the negotiation process [196].

Knowledge bead (KB) provides an object-oriented way to specify the knowledge in agent negotiation for B2B (Business to Business) transactions [30]. In addition to the inheritance and hierarchy features of object-oriented modeling, each KB carries a criterion and a weight specifying the importance of the criterion. This makes it possible to represent the various forms of knowledge which includes the specification of products, negotiation strategies, user preferences, constraints and the desired final deals in an efficient way [31].

Knowledge bead has been defined in [30] as “an encapsulation of definition, behavior, and data. It can be a composite object or a simple, atomic part object”. Mostly each has their own methods and data. Definition is a static unique description. It can be a UPC (Universal Product Code) or a unique index implemented at the ontology databases for referencing this KB. Behavior is described by a set of methods and rules manipulating the KB’s and their attributes like KB formation, attributes alteration, duplication, pruning and linking to other KB. They are analogous to class methods or functions in object-oriented programming, and can be inherited from base classes. Data is the set of attributes contained in the knowledge beads. A weight is assigned to each attribute, which a relative priority is indicating how important this attribute is in the BOM (Bill of Material), which is delivered as RFQ (Request for Quote) to potential sellers [196].

Once the KB is defined based on ontology, knowledge taxonomy is defined. The goal of a corporate taxonomy is to provide a list of authorized terms in knowledge management and
information seeking [197], as well as the mapping between concepts to connect traders with the right knowledge at the right time and the right place. Knowledge beads make the communication and negotiation between the agents easy and simple. There is a difference between negotiation and communication. Communication is the exchange of messages between the agents. Negotiation is reaching an agreement. Both the parties place their deal and then finally they reach onto a conclusion. The negotiation problems can be tacked in an agent trading scenario by associating with it the information discovery and ontology issues. The negotiation can be made easy by incorporating certain structures, rules, protocols or conventions on the interaction between the agents [191]. To automate the agent’s negotiation process, two most important things that need to be done are [196]:

1. Formalization of the negotiation process

2. Incorporation of necessary negotiation knowledge and intelligence.

However, due to the fuzzy nature of agent negotiations, the re-use of the knowledge and knowledge interoperability becomes tough. Thus there is a need for an effective and sophisticated negotiation process which can do coalition formation, business assessment, criteria evaluation and most importantly knowledge management.

All the above discussed research papers [30, 31, 191 and 196] have used knowledge beads for B2B transaction and this fact also implied that there was a significant scope for research work available in the direction of using knowledge beads for B2C (Business to Consumer) transactions.

6.2 Fuzzy c-means Clustering

Clustering is used to group the data members into a small number of clusters, each of which captures a particular aspect of the query [142]. The more is the similarity between the data in the same group and more the difference between other groups, better is the clustering. Clustering is a general methodology and a remarkably rich conceptual and algorithmic framework for data analysis and interpretation [198]. It enables fast information retrieval. The data to be searched can be looked into its corresponding cluster instead of searching for
it in the entire database. Fuzzy c-means clustering methods allow the objects to belong to several clusters simultaneously, with different degrees of membership [153]. It eliminates the disadvantage of hard clustering in which an object can belong to one and only one cluster [199]. Determining the number of clusters in a data set is a frequent problem in data clustering, and is a distinct issue from the process of actually solving the clustering problem. The correct choice of ‘c’ is often ambiguous, with interpretations depending on the shape and scale of the distribution of points in a data set and the desired clustering resolution of the user. In the proposed system, the number of clusters is domain specific. The domain expert decides the number of clusters (‘c’) to be formulated”.

6.3 MAEKMS for E-commerce

With the advent of the internet, e-commerce has reached such great heights that its implementation has become all the more complex with the huge amount of information to be handled. Moreover, the e-commerce system should be such that it can direct the user to the desired products available at the best price and help them in buying the products in least time and energy. The authors have proposed an intelligent algorithm combined with case based reasoning, knowledge beads and fuzzy c-means clustering. The MAEKMS for e-commerce is an extension of the system described in chapter 2. In addition, Knowledge management is a revolution of enterprise management, and is an important means to improve the overall functioning of an enterprise [200, 201]. The properties of the multi agent e-commerce system are enhanced by incorporating the knowledge management perspective.

Nowadays, knowledge management is a prerequisite for a successful e-commerce system. To operate in e-commerce environment, an organization has to have good grasp of knowledge on its markets, customers, products and services, methods and processes, competitors, employee skills and its regulatory environment [202]. Such a knowledgeable system would be able to assist the user in buying the right thing from the right place. The proposed system takes care of all the above mentioned aspect and helps the user in buying a product. A simple elementary design of the system is depicted in the Figure 6.1.
A shopper practices multi-modal search. A user might also exercise other criteria apart from price in buying a product like customized specification [151]. The proposed system allows the user to specify multiple attributes for a product he is looking to buy. It also gives a common platform to the user, so that he doesn’t have to visit each and every site and look for the desired specifications.

Mostly consumers exhibit experimental behaviour, in which they don’t exactly have an intention of purchase. They just browse through the products available, look for the discounts, look for the past history of the products, and if everything suits him then only he purchases the item. Even an offline user trying to buy a product consults with his friends/relatives and then decides upon a shop from which he would like to buy it. The proposed system allows the user to look into the different past cases pertaining to sale of his desired product and then decide upon the site to buy from. A knowledgeable retrieval algorithm integrated with an effective clustering technique and case based reasoning is proposed to find the similar cases in the case base.

Enterprises are witnessing tremendous growth in business information yearly (more than 150% in some cases) [203]. As the information increases, the number of cases in the case base also increases. In such a scenario, search and information retrieval becomes all the more challenging. This research work uses Fuzzy c-means clustering for clustering the data as well as for analyzing the data.

Most of the existing multi-agent systems for e-commerce [9, 16 and 204] have an agent assigned for a product, which is not suitable in today’s scenario when there are huge numbers of products ready to be sold. The proposed system adopts the policy of having one
agent per e-commerce site. So, all the products being sold by the site are managed by that one single agent, allotted to it.

Simon's proposed negotiation system [30] embraces the full automation of online trading agents that are capable to cope with the rapid changes experienced by various business-to-business (B2B) commercial activities. This research work uses knowledge beads for B2C transactions. The case and the product details in their respective databases are stored in the form of knowledge beads.

Finally, the data is analysed using regressions. The regression analysis measures the degree and nature of the effect of one variable on another [205]. In simple linear regression there is one independent and one dependent variable. However, in multiple regression, which is a logical extension of linear regression, two or more independent variables are used to estimate the values of a dependent variable [206]. Polynomial regression is a form of linear regression in which the relationship between the independent variable $x$ and the dependent variable $y$ is modelled as an nth order polynomial [207]. By analysing the data, the e-commerce sites find out the most popular products amongst the online customers. The sites can also predict the price of the products, which they need to offer to the customers so that the sales are increased. Thus the sites can plan their business strategies accordingly.

6.4 Architecture of the Proposed System

An e-commerce system should be capable enough to handle large number of buyers, sellers and the huge number of products to be sold. A knowledgeable retrieval algorithm integrated with an effective clustering technique, case based reasoning and knowledge beads is proposed. The architecture of the system is depicted in the Figure 6.2.
The system consists of the following agents:

1. Login agent
2. RFQ agent
3. Search agent
4. Case agent
5. Match agent
6. Payment agent
7. Product agent
8. Register agent

6.4.1 Information Flow

The user logs in to the system through a GUI (Graphical User Interface). The login details are managed by the Login agent which keeps track of all the login ids and the passwords of the user in accordance with the Login data base. The user may be aware of the
 specifications/ attributes of the product which he wants to purchase or he may be unaware of all the details and need to surf before deciding upon a product and the site. In the first case he can directly go to the RFQ (Request for Quote) agent and fill in the details of his desired product. In the second case he will look into the product specifications as submitted by the selling sites. The product agent keeps track of all the products submitted by the registered sites. It maintains its own product base for all the products.

Once the details of the products are known, the RFQ entered by the user is compared with the stored cases in the case base in the CBR (Case Based Reasoning) module. CBR module displays the past experiences of other customers related to the product desired by the user. It performs its duties in collaboration with the case agent. Case agent is discussed in section 3.2. If no match is found then the user has to decide from the matches found by the match agent in the product database.

The details of the products to be sold by the sites are stored in the product database. The attributes of the products in the product database are the same as the ones present in the case in the case base. The working of the match agent and the search agent is exactly the same. The difference lies from which database the searching or the matching is to be done. It is described in detail in the section 6.8.

The payment agent is responsible of forwarding the request of the user to the selected site for the payment. It takes the user to the specific payment gateway of the site and the purchase is done.

The register agent is to register the sites for the system. There are certain criteria which needs to be fulfilled by the site like the site should be having COD facility or the delivery cities should be greater ‘n’ etc. If a site fulfills them then it can register for the system. Once it has successfully registered then it can register its products in the product database with the help of the product agent. The information flow is clearly depicted in the Figure 3.

107
6.4.2 RFQ Agent

Whenever a customer goes to buy a product, he negotiates with the seller. The topic of negotiation can be the price or it can be any other attribute of the product. The automation of this negotiation minimizes the human intervention and most of the work is done by the agents for the user. To automate the agent negotiation process, first of all the negotiation process should be formalized which incorporates negotiation knowledge and intelligence. The proposed negotiation process involves case based reasoning [137] as the process for negotiation. The negotiation knowledge and intelligence comes from knowledge beads. Case based reasoning (CBR) solves the purpose of knowledge sharing and knowledge reuse.

Knowledge bead is a combination of definition, behavior, data and weight [30]. A knowledge bead is a combination of many attribute beads. Each attribute beads consists of two fields. One is the attribute type and the other is the value of the attribute. After the user has successfully logged in, the user is asked to fill the Request For Quote (RFQ) [208] by the RFQ agent. In the RFQ there are certain negotiable and some non negotiable attributes. The non-negotiable have the maximum weight and the customer doesn’t want to negotiate onto that. A general RFQ is shown in the Figure 6.3.

![Figure6.3: RFQ using KB](image)

The addition of the weights should be 100%. The RFQs of different products is different because the attributes considered in different products are different. Once the RFQ is prepared, it is searched with the cases in the case base. The searching is in the order of the weights assigned to the attributes. A RFQ looks like as in the Figure 6.4.
The attributes $A_1$, $A_2$, $A_3$ and $A_4$ have values $V_1$, $V_2$, $V_3$ and $V_4$ and weights $W_1$, $W_2$, $W_3$ and $W_4$ respectively. The attribute with the maximum weight is the one which is non-negotiable and the most important. Once it has been matched with the case in the case base, then the rest of the attributes are searched for. If there are two attributes with the same weights, then the First Come First Serve (FCFS) is applied. The detailed algorithm is mentioned in section 6.4.6.

### 6.4.3 CBR Module

The CBR module of the system uses interpretive case based reasoning. It takes a situation or solution as input and its output is a classification of the situation, an argument supporting the classification or solution, and/or justifications supporting the argument or solution [138]. Since, in an E-commerce system there are no computational methods to evaluate a solution, interpretive CBR is most useful. A case stored in the case base has the format as shown in the Figure 6.5.

![Figure 6.5: A case in the case base](image)

For an E-commerce system, the “solution” tab as in the Figure 6.5, is the site from which the specific product satisfying all the mentioned attributes was purchased. A person looking for a similar product searches for such cases in the case base. The attributes mentioned in the
RFQ are matched with the attributes in the case in the order of their weights. The solution of a case is displayed to the user only if all the non-negotiable attributes have a match with the ones in the case.

6.4.4 Clustering- Case Base and Product Base

For retrieval of cases, the case base should be properly managed so that retrieval process is fast and efficient. For this, the cases are indexed using a clustering method. Clustering techniques are mostly unsupervised means to organize data into groups based on similarities among the individual data items. The proposed system uses fuzzy clustering. In an E-commerce system, there are many products which may lie in more than one product category. So, if clusters are being formed on the basis of product category, it's not possible for hard clustering to group the cases. Fuzzy clustering, however, allow the cases to belong to several clusters simultaneously with different degrees of membership. Fuzzy clustering algorithm is a kind of effective fuzzy space partition method [209].

Suppose there are N cases with n attributes then the n attributes are grouped into n-dimensional column vector

\[
    z_k = [z_{1k}, z_{2k}, \ldots, z_{nk}]^\top, z_k \in \mathbb{R}^n
\]  

(6.1)

The N cases are denoted by

\[
    Z = \{z_k | k = 1, 2, \ldots, N\}
\]

(6.2)

The n* N matrix is represented as

\[
    Z = \begin{pmatrix}
    z_{11} & \cdots & z_{1N} \\
    \vdots & \ddots & \vdots \\
    a_{n1} & \cdots & a_{nN}
    \end{pmatrix}
\]

(6.3)

The rows are the cases and the columns are the attributes. Z is the case matrix. The proposed system uses probabilistic partition. A case can belong to more than one cluster and the sum
of the memberships of the case in different clusters is not necessarily equal to one. In terms of membership (characteristic) functions, a partition can be conveniently represented by the partition matrix $U = \{\mu_{ik}\}_{c \times N}$. The $i$th row of this matrix contains values of the membership function $\mu_i$ of the $i$th subset $A_i$ of $Z$. The elements of $U$ must satisfy the following conditions:

$$\mu_{ik} \in [0, 1], 1 \leq i \leq c, 1 \leq k \leq N,$$

$$\exists i, \mu_{ik} > 0, \forall k,$$

$$0 < \sum_{k=1}^{N} \mu_{ik} < N, 1 \leq i \leq c.$$  \hspace{1cm} (6.4)

where $c$ is the number of clusters. The possibilistic partitioning space for $Z$ is

$$M_{pe} = \{U \in \mathbb{R}^{c \times N} | \mu_{ik} \in [0, 1], \forall i, k; \forall k, \exists i, \mu_{ik} > 0; 0 < \sum_{k=1}^{N} \mu_{ik} < N, \forall i\}$$  \hspace{1cm} (6.5)

where 'c' is the number of clusters. In the proposed system fuzzy c-means clustering is also used for data analysis. The E-commerce sites must be able to find out the items mostly bought or the ones which are not very common amongst online users. Fuzzy c-means (FCM) clustering groups the cases in the base and the product deals in the database such that the E-commerce sites can analyze the sales of particular items. The clusters can be customer based also, which tracks the purchases made by the customer in different months. This enables the sites to give discounts in those particular months. Since the complete dataset is available in the database, any type of cluster can be formed through FCM, which helps the E-commerce sites to analyze and interpret the data.

### 6.4.5 Product Agent

The sites willing to sell their products need to register with the system through the Register agent. There are certain criteria that a site needs to fulfill and then only it would be allowed to register. The criteria can be like, the site should have at least 'n' delivery cities, the site
should have Cash on Delivery (COD) facility etc. Once the registration is done successfully, the sites can put in the details of the products. An entry in the product base would like as shown in the Figure 6.6.

<table>
<thead>
<tr>
<th>Product_name</th>
<th>Attribute 1</th>
<th>Attribute 2</th>
<th>Attribute 3</th>
<th>Attribute 4</th>
<th>Attribute 5</th>
<th>Site_name</th>
</tr>
</thead>
</table>

Figure 6.6: Product details stored in the product base

This is similar to the case base; the only difference is that the “Solution” tab in the Figure 6.6 is replaced with “Site name” in the Figure 8. Hence the same fuzzy clustering technique can be used to search an entry in the product database with the RFQ.

If any change is to be made in the details of the product in the product base, then it has to be notified to the product agent. E.g. the price is discount by 20%, and then the new price should be updated in the database. Thus the user sees the latest information about the products.

6.4.6 Search Agent/Match Agent

The searching of buyer RFQ in the case base and in the product database should be such that it is fast and effective. Let $A$ represent the set of the attributed in the RFQ. $A$ is depicted as

$$A=\{A_1,A_2,A_3......A_n\}$$  \hspace{1cm} (6.6)

Where $n$ is the total number of attributes in the RFQ. The set of values of the attributes in a RFQ is represented by $V$ which is

$$V=\{V_1,V_2,V_3......V_n\}$$  \hspace{1cm} (6.7)

Where $V_i$ is the value of the $i$ th attribute. The set of weights assigned to the attributes is represented by $W$ and is

$$W=\{W_1,W_2,W_3......W_n\}$$  \hspace{1cm} (6.8)

Where $W_i$ is the weight assigned to the $i$ th attribute. So, the RFQ of a product is a collection of tuples of the form
\[ <A_i, V_i, W_i, N_i> \]  \hspace{1cm} (6.9)

For the \( i \) th attribute, where \( N_i \) is a Boolean value telling whether the attribute \( A_i \) is negotiable or not. E.g. the RFQ of the product Jeans would be like

\[ \text{RFQ}_{\text{Jeans}} = \{\{\text{Brand, Puma, 8, 1}\}, \{\text{Price, 2000, 9, 0}\}, \{\text{Color, Black, 7, 1}\}, \{\text{Size, Medium, 10, 0}\}\} \]

A case \( C_j \) in the case base is a collection of tuples of the form. The value of 0 for \( N_i \) means that’s it’s a non-negotiable attribute.

\[ <A_m', V_m'> \]  \hspace{1cm} (6.10)

Where \( C_j \) is the \( j \) th case in the \( i \) th cluster. \( A_m' \) is the \( m \) th attribute and \( V_m' \) is the value of the attribute \( A_m' \) of the \( j \) th case in the \( i \) th cluster.

Suppose there are \( 'n' \) attributes in the case ‘\( j \)’ then the case \( C_{ij} \) can be represented as

\[ C_{ij} = \{\{A_1, V_1'\}, \{A_2', V_2'\}, \{A_3', V_3'\}, \{A_n', V_n'\}, S\} \]  \hspace{1cm} (6.11)

Where \( S \) is the solution which is the site from where the product Jeans was purchased earlier.

\( C_{j} = \{\{\text{Brand, Puma}\}, \{\text{Price, 2800}\}, \{\text{Color, Red}\}, \{\text{Size, Medium}\}, \text{ebay}\} \)

It may happen that as the social trends in fashion changes, the attributes being considered in the product base has changed. Correspondingly the buyer RFQ for that product also changes. Now when the customer fills in the RFQ, then there are some extra attributes also which need to be compared with the cases in the case base. In such a case, if the extra attribute added is non-negotiable, then there won’t be any match with the cases in the case base and the searching is done in the product base. This can be represented mathematically as well. Suppose the number of attributes in the RFO is \( n \) and the attributes in the case base is \( m \), then the condition would be like

\[
\begin{align*}
\text{If } n \leq m \\
\text{search in the case base}
\end{align*}
\]

\[
\begin{align*}
\text{Else} \\
\text{search in the product base}
\end{align*}
\]
We have assumed that there will be at least one non-negotiable attribute in the RFQ. The algorithm of the entire search procedure is formulated below:

1. Find out the attribute with the highest weight in the RFQ and search for that attribute value pair in all the cases in the cluster.
2. If
   There is a match, store all the matched cases in ‘Matched’ and remove that attribute value pair from the RFQ
   Else
   Display to the user that the “No match”.
3. Repeat the steps 4 to 6 until all the attributes are matched
4. Get the attribute with the highest weight in the RFQ and search for that attribute value pair in all the cases in the ‘Matched’
5. If
   There is a match do nothing
   Else
   Store the case in ‘Matched1’
6. If
   There is a case/ cases in the ‘Matched’ then that’s the exact match and display to the user the cases in the ‘Matched’. The selection made by the user is informed to the payment agent.
   Else
   Repeat the steps a and b for all the non-negotiable attributes
   a. Compare the attributes in the RFQ with the negotiable field as ‘0’ with the attribute value pair in the ‘Unmatched’
   b. If
      There is match, do nothing
      Else
      Remove that case from the ‘Unmatched’
7. Display ‘Unmatched’ to the user in the order of the weight of the negotiable attributes. The selection made by the user is informed to the payment agent.

‘Matched’ ArrayList contains all the cases for which

\[ \forall i \ (A_i, V_i) = (A'_i, V'_i) \]  \hspace{1cm} (6.12)
where $n$ is the number of attributes in the RFQ.

The solutions $S$ in the cases are given as choices to the user, from which it can choose according to his desire. If there is an exact match then the ‘Unmatched’ ArrayList contain all the cases for which

$$\exists i \ (A_i, V_i) \neq (A_i', V_i')$$

(6.13)

where $A_i$ is the attribute whose value in the RFQ didn’t match with the value in the case.

If there is no exact match, then the ‘Unmatched’ contains all the cases for which

$$\forall i \ (A_i, V_i) = (A_i', V_i') \ ; \ \exists j \ (A_j, V_j) \neq (A_j', V_j')$$

(6.14)

where $i$ are the non negotiable attributes and $j$ are the negotiable attributes.

The same search algorithm can be used by the match agent to find out the match between the buyer RFQ and the product details mentioned in the product database.

Considering that the clustering is done, the complexity of the discussed algorithm turns out to be the order of $O(n)$, where the number of non negotiable attributes in the RFQ is less than or equal to three.

### 6.5 Implementation

The proposed system is implemneted in JADE using NetBeans IDE. JADE is probably the most widespread agent-oriented middleware in use today. It is a completely distributed middleware system with a flexible infrastructure allowing easy extension with add-on modules. The framework facilitates the development of complete agent-based applications by means of a run-time environment implementing the life-cycle support features required by agents, the core logic of agents themselves, and a rich suite of graphical tools [212].

The database used is Oracle 11g. First of all a seller agent, “SellerI” is created selling an apparel with the details as shown in the Figure 6.7
The buyer agent is created which is willing to buy a coat of price 5000. The non-negotiable attributes considered are price and size in Figure 6.8

Since the price asked by the buyer agent is more than the price being offered by the seller agent, the product “coat” is sold to the buyer agent as shown in the Figure 6.9
If the price would have been less than the price being offered by the seller agent, then even if all the other attributes are same, the product will not be sold as the price is a non-negotiable attribute with the highest priority. This scenario is shown in the Figure 6.10.
6.6 Comparison of the Proposed System with Multi Agent E-commerce System

The proposed system is compared with the Multi Agent E-commerce system described in the chapter 4:

1. The cases in the case base are stored in the form of ontology using ontology case based reasoning. In the current system, cases are stored in the form of knowledge beads.

2. Storing the cases in the form ontology requires lot of space. The product details are also stored in the form ontology taking huge amount of space. The category name,
sub-category name and the sub-sub category name are repeated again and again for one type of product belonging to different brands or having different prices.

3. In the proposed system the cases and the product details are stored in the form of ArrayList which takes very less space as compared to ontologies.

4. Clustering is used to cluster similar cases together which expedites the process of information retrieval.

5. A multi agent E-commerce system without KM can help the user in buying a product and answer the questions as Who, What, Where and How the products can be bought. But for an E-commerce site, it’s very important to find out “Why” a particular product was bought. MAEKMS for E-commerce addresses this problem.

6. The proposed knowledgeable multi agent system would be able to assist the user in buying the right thing from the right place.

7. Linear and multiple regression analysis technique enable the E-commerce sites to analyze and interpret the sales data, and then accordingly decide their business strategies.

8. The proposed system integrates CBR with knowledge management to make an agent behave just like a human being.

6.7 Comparison of the Proposed System with Existing E-commerce Systems

Current available e-commerce systems are based on centralized servers. Centralized servers are a weakness of E-commerce systems, since they introduce a single point of failure and a bottle neck. That is, if the server goes down for any reason, the whole system will go down.

The research paper [204] talks about collaborative MAS in which each agent can talk to its peers instead of having one centralized server. It provides flexible P2P communication.

The system presented by Chavez and Maes [9], helps users creating agents to negotiate the buying and selling of goods on their behalf, also allowing the specification of parameters to guide and constrain an agent’s overall behavior. However, these agents do not live
permanently but only during the completion of a certain transaction, thus not fully exploiting each actor’s profile. The proactiveness and semi-autonomy of their agents (e.g., towards new coming offers or requests) is also not fully implemented.

Karacapilidis and Moraitis [211] presented a web-based electronic commerce system in which customers and merchants delegate the related tasks to their personal software agents. Messages passed between these agents can fully encapsulate the associated parties’ points of view towards a market transaction.

Collins et al. [212] presented MAGNET, which is a mobile agent-based system that enables buyers to compare products from different sellers. Negotiation between agents is based on multi-attributes. Although MAgNET agents are able to negotiate over multi-attributes, only one buyer agent visits many sellers sequentially. This has a high impact on the system performance when the number of seller agents increases. Anthony and Jennings [213] proposed an agent framework to bid over multiple auctions. Although, they were able to address the problem of segregation, other issues were overlooked.

The comparison between the existing e-commerce systems and the proposed system can be represented by the Table 6.1. The rows are the properties of the system and the columns are the different systems proposed by different researchers.

<table>
<thead>
<tr>
<th>Properties</th>
<th>[10]</th>
<th>[147]</th>
<th>[193]</th>
<th>[207]</th>
<th>[211]</th>
<th>[212]</th>
<th>[213]</th>
<th>[214]</th>
<th>[215]</th>
<th>[216]</th>
<th>MAE KMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>KM</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>MAS</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Multi modal Search</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Multiple buyer / seller agent</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

None of the above discusses E-commerce models support the following features:

1. One seller agent per e-commerce site. All have one seller agent per each product being sold.
2. Combination of Case Based Reasoning and knowledge beads.
From the above discussions it can be concluded that the proposed system has many features and properties which doesn’t exist in current e-commerce systems. It also proves that the inclusion of knowledge management in a multi agent e-commerce system makes the system all the more intelligent and efficient. In the future, the authors wish to have effective case based reasoning technique in which the similar cases are not recorded. In the current implementation only the cases with the exact match are not being included in the case base. It may happen that the cases recorded in the case base, are outdated in terms that the product is not available anymore on the site, or the price of the product has changed. Such cases need to be removed from the case base. In the proposed system, multiple regressions is used to analyze the data with 3 attributes at a time, in the future, the authors plan to include more than three 3 attributes in the analysis at a time.

6.8 Summary
The MAEKMS for e-commerce has been designed and implemented using case based reasoning, knowledge beads and fuzzy c-means clustering. The proposed system is then compared with the e-commerce system proposed in the chapter 4 and the importance of knowledge management is highlighted. The proposed system is also compared with the existing e-commerce systems.