İSTANBUL KULTUR UNIVERSITY ★ INSTITUTE OF SCIENCE

AN ARCHITECTURAL MODEL FOR CONTENT MANAGEMENT IN E-COMMERCE APPLICATIONS USING INTELLIGENT AGENTS

Akhan AKBULUT

DEPARTMENT OF COMPUTER ENGINEERING

JULY 2008

Approval of the Institute of Research and Graduate Studies

Prof. Dr. Turgut UZEL
Director

I certify that thesis satisfies all the requirements as a thesis for the degree of Master of Computer Engineering.

Prof. Dr. Ümit KARAKAŞ
Head of the Department of
Computer Engineering

This is to certify that we read the thesis named “AN ARCHITECTURAL MODEL FOR CONTENT MANAGEMENT IN E-COMMERCE APPLICATIONS USING INTELLIGENT AGENTS” which has done by Mr. Akhan Akbulut, graduate student no: 0509050012, and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Computer Engineering.

Asst. Prof. Dr. Güray YILMAZ
Thesis Supervisor

Examining Committee Members:

Prof. Dr. Murat TAYLI ____________________________

Prof. Dr. Ümit KARAKAŞ ____________________________

Asst. Prof. Dr. Güray YILMAZ ____________________________

ACKNOWLEDGEMENTS

I would like to thank to my advisor Asst. Prof. Dr. Güray YILMAZ, for his contribution to my education, for his motivation and showing me the directions to follow in this study, and especially for giving his time and helping me constantly.

I am also grateful to Prof. Dr. Murat Taylı and Prof. Dr. Ümit Karakaş for their valuable criticisms and participating in my thesis jury.

I must specially thank to my family for their continuous support and trust in me. My dept to them is certainly the biggest of all.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

AGBEP	: Agent Based e-Commerce Platform
ACL	: Association for Computational Linguistics
AID	: Unique Agent Identifier
B2B	: Business to Business
B2C	: Business to Consumer
BCP	: Buyer Collective Purchasing
C2C	: Consumer to Consumer
CA	: Clustering Agent
DBMS	: Database Management Systems
DFS	: Distributed File System
FCM	: Fuzzy C-Means
FDA	: Feed Delivery Agent
FIPA	: The Foundation for Intelligent & Physical Agent
HA	: Hashing Agent
IA	: Inspector Agent
JADE	: Java Agent Development Platform
JAS	: Java Agent Services
KQML	: Knowledge Query Manipulation Language
MAS	: Multi-Agent System
MD5	: Message-Digest Algorithm 5
NFS	: Network File System
MTP	: Message Transport Protocol
NLB	: Network Load Balancing
O2A	: Object to Agent
OWL	: Web Ontology Language (W3C)
RDF	: Resource Description Framework
RMI	: Java Remote Method Invocation
SAA	: Site Administration Agent
SAN	: Storage Area Network
SHA	: Secure Hash Algorithm
SR	: Semantic Representation

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

This work addresses tuning content management and administration processes in e-commerce systems. The keyword “MALL” stands for the web sites which consist of many e-shops collected under a common roof for redounding their sale performances. The main problem for these MALLs is gathering product information from member shops and integrating them into the system efficiently as an autonomous job. My proposal is to design a platform called *AGBEP* which consists of multi-agents with different system roles. Intelligent agents work in many jobs respectively deciding the product’s category by observing the keywords in product specifications and explanations using clustering methods aided by hashing techniques.

Keywords - *Intelligent agents, e-commerce, e-malls, content management, clustering, hashing*

ÖZET

Bu çalışmanın amacı e-ticaret (e-commerce) uygulamalarının içerik toplama ve yönetimi konularına değinerek, bu alanlardaki problemlere bir çözüm önerisi sunmaktır. Ortak bir çatı altında birleşmiş pek çok e-mağaza'nın (e-shop) oluşturduğu e-Alışveriş Merkezi (e-Mall) sitelerde; üye mağazaların gönderdiği ürün bilgilerinin sisteme uygun bir şekilde dâhil edilmesi ve yönetimsel işlevlerin otonom bir yapı ile sunulması gerekmektedir. Bu mimari organizasyon için çoklu-ajan (multi-agent) destekli bir platform kullanılması öngörülmektedir. Çoklu-ajan platformunda farklı görevlerde çalışacak olan ajanlar sırasıyla; gönderilen içeriğin barındırdığı her bir ürün için açıklamalarında geçen kritik kelimeleri belirleyecek, anahtarlama (hashing) fonksiyonları ve kümeleme (clustering) teknikleri kullanarak sistem dâhilindeki en uygun ürün kategorisi altına yerleştirilmesi sağlanacaktır.

Anahtar Kelimeler – *Akıllı ajanlar, e-ticaret, e-mağaza, içerik yönetimi, anahtarlama, kümeleme*

1 INTRODUCTION

E-commerce and E-business are the major contributors to the current emergence of Digital Economy [1]. Today, e-commerce has some management difficulties that users are not aware of. Product management is not a problem for singular online stores, however, when the store number reaches hundreds for one site, much more complicated decision-making mechanisms must be used. Rather than some big MALLs, there is no suitable management infrastructure for the product structure. In this paper we propose a method that can be applied to any e-commerce site, before or at the time of installation, when it is determined that the data consistency of the product management should be increased.

At the big e-commerce and comparison sites like Kelkoo [2], Shopzilla [3] or Froogle [4]; the system structure is set upon a central management and consists of the services given by this central structure. In general, these services are supplied by web services that give answer, to the authentication, product management and reporting needs of the member stores. The most critical issue for these services is product management because it causes much more problems. Member stores update their products by uploading their catalogues in the systems required format. Since there is no standard format for products' feeds, XML is generally preferred because of its performance criterion. But as will be explained in chapter two, there are also various infrastructures available that use a tab-delimited text-based format, like Froogle.

Whatever format is used, the information being sent includes the product names, detailed explanations, prices, the product images and their categories. Member stores use this information in their structure and also control them by one hand in the system. For example, a member store can exhibit a laptop computer on its site under the heading name of “home electronics”. By integrating this product into the system with the information at the member store, the same product will be shown at many places with different heading names such as “computers”, “notebooks”, etc. For this reason, the data integrity will be failed in the system. Since neither product category of the member stores will be the same as the others, it will become harder to fit to the system. At this point, a smart decision making mechanism that will decide which category to include the product explanation must be used. We advise to give this duty to our proposed AGent Based E-commerce Platform (AGBEP) [5].

My approach is to use the intelligent agents to analyze the complex subject of the content management. It is important for working performance and prestige to minimize the mistakes that can occur and repeat themselves in product management.

This study is organized in eight chapters. In chapter two, the information about the basic definitions of e-commerce and e-mall technologies will be given. There exists the definition of the structure and how it works that AGBEP takes as a basis.

In chapter three, there exists the clustering and hashing methods' definition explanations and how they are used in order to solve the problem about the category determination.

Chapter four focus on the agent technologies, and its usability in e-commerce applications.

Chapter five gives us an opportunity to define the system by giving a detailed explanation about the AGBEP's general architecture and methodology. The whole technical information about the each element used in the system is presented in this chapter.

Chapter six is an implementation of AGBEP which is scened as a Digital Warehouse.

In chapter seven, we mention about the results of our work and illustrate an implementation project. We keep AGBEP technology as a basis and we interpreted the tests and their results of a sample application. We preferred a mall that contains e-shops which sale goods about photograph technology.

Last chapter conclude the thesis and describe some expansions that can be applied as further works.

2 BACKGROUND

2.1 Electronic Commerce

Electronic Commerce (e-Commerce) consists of the buying and selling of products or services over electronic systems such as the Internet and other computer networks. A wide variety of commerce is conducted in web, spurring and drawing on innovations in electronic funds transfer, supply chain management, Internet marketing, online transaction processing, electronic data interchange, automated inventory management systems, and automated data collection systems.

E-Commerce is about setting your business on the Internet, allowing visitors to access your website, and go through a virtual catalog of your products or services online. Types of e-Commerce can be studied in three parts.

Business to Consumer : While the term e-commerce refers to all online transactions, "business-to-consumer" stands for B2C and applies to any business or organization that sells its products or services to consumers over the Internet for their own use. B2C describes activities of commercial organizations serving the end consumer with products and/or services. It is usually applied exclusively to electronic commerce.

Business to Business: Business to business, shortly B2B is the exchange of services, information and/or products from one business to another, as opposed to between a business and a consumer

Consumer to Consumer: Consumer-to-consumer (or C2C) electronic commerce involves the electronically-facilitated transactions between consumers through some third party. Some examples of C2Cs are; gittigidiyor [6] and ebay [7].

2.2 Electronic Malls

An Internet mall (or e-Mall) is an online venue that hosts a community of individually-owned shops. The mall owner provides the online storefront and the tools for each shop owner to set up, manage and maintain their shop. Shop owners gather and manage their own inventory; they take the photographs then upload them, write the item descriptions, and decide on the price. Online shopping malls often provide a global search of their shops, and help with search engine placement.

A mall is a group of shops whose owners have agreed to work together to increase the number of visitors to their shop and hopefully their respective sales while malls can have some benefit for players with smaller shops or mini malls.

In mall there are several categories. Each member of a mall is assigned one or more category of item. Each shop in the mall displays a graphical banner which appears at the top of their shop and advertises the different categories of items available within the mall each category is linked to the shop specializing in that category so that clicking on the category brings the player to the relevant shop.

Mall drives the traffic to their members' shops in several ways but by far the most important is through the marketplace.

In Figure 2.1, we can analyze that an e-mall works in different layers for various type of user access.

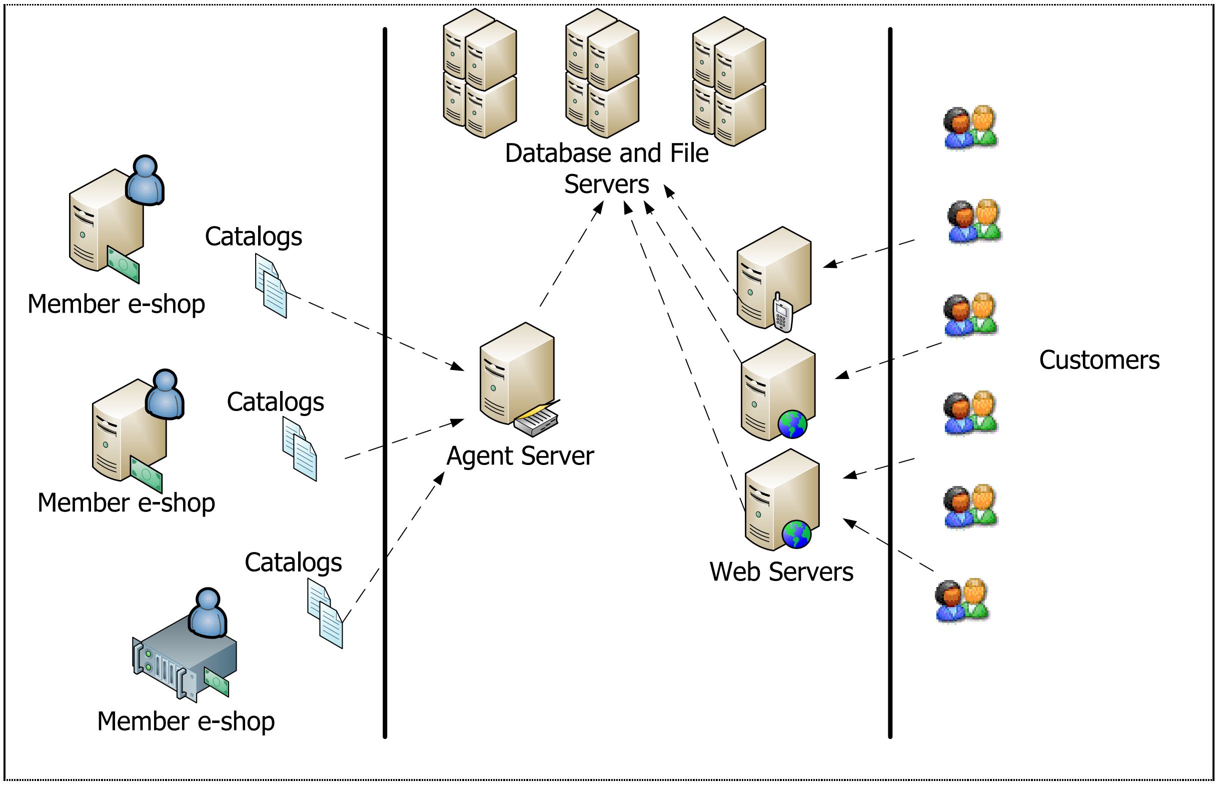


Figure 2.1: Overview of AGBEP as an e-Mall

Online malls offer several advantages to the seller. They usually get good results in web searches, and they offer attractive services such as hosting, newsletters and advertising.

The basic difference of an e-mall from a standard e-commerce page is its products in the catalog are much bigger. The biggest problem in e-mall is its products are provided from different suppliers and the catalogs are being continuously updated. In order to solve these problems, different companies offer different solutions. In generally, to attend editors in charge of each category is a solution where the technology is least used? This solution can be applied very less and we can't avoid it when it gets blocked because of human mistakes. To overcome these problems is to use non human-autonomous structures. The work explained in this research introduces the solutions for these problems by using agent architectures. We prefer to interpret and apply the whole processes by using the information inside the system with a non human used way.

2.3 Agent Technologies in e-Commerce Applications

When e-commerce site users are increases rapidly, manually controlling the accounts, products, orders etc. get difficult, so Agents step in to do some operations automatically and autonomously.

Agents are usually used searching [8, 9], selling [10, 11], negotiation [12], security [13] areas in e-business and e-commerce applications.

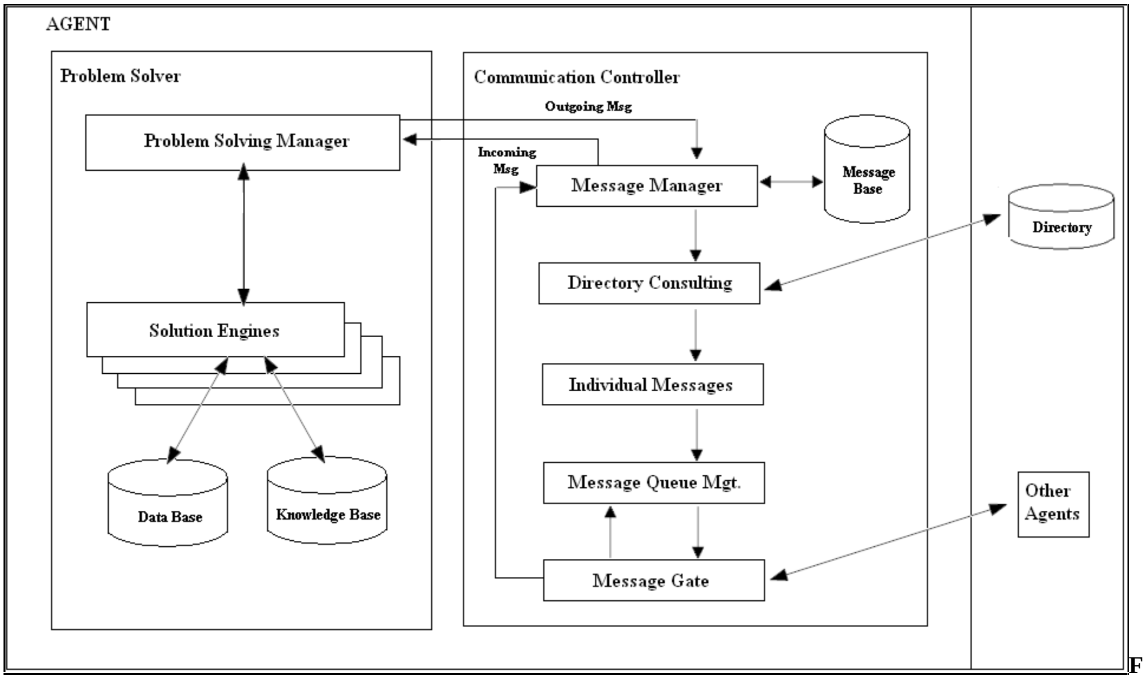
E-Commerce users want to advances product searching, so agents used to upgrades the searching algorithm.

One of the method is creates one search agent for each customer. In the scenario [8], when the customer search desired products, the customer agent search the web and contact with the merchant agents, and get information about product such as price, delivery cost, delivery time etc. Based on the replies by the merchant agents, the customer agent either reports its findings ranked according to the given preferences to its owner or it buys the goods directly from the merchant who made the best offer.

In searching there is more efficient agent is intelligent [9] UNIK-AGENT the new generation of electronic commerce which is based on intelligent agents is mentioned in Figure 2.2. Contract type affects the communication messages and solution methods of agents in electronic commerce environment. So new contract types are proposed in this article. So they reduce the costs of transactions in respect of traditional ways. The messages that are in agent based commerce contain three layers.

These layers are;

- Agent Communication Language Layer
- Electronic Commerce Layer
- Product Specification layer.



F

Figure 2.2 UNIK-AGENT prototype

The purpose of this idea is that to automate and makes easy transactions such as contracting, searching product and product selection by customer and vendor agents as intelligent agent. So to determine the contract type for communication is very important step for electronic commerce because of the sales and purchases.

UNIK AGENT model is model of communication controller and problem solver components. This agent idea can choose the most appropriate solution of a problem over a solution pool which includes many solutions. Namely it includes a solution engine inside of it. Problem solver mechanisms firstly control the process and then propose an available solution to the received messages. Communication controller ensures the message communication process .And it has five layers to process.

- Message Manager Layer
- Directory Consulting Layer
- Individual Message Layer
- Message Queue Management Layer
- Message Gate Layer

The main title of the E-commerce is selling so agents have a great role in this title, so advertisement is more important to sell products; the agents can easily adapted this situation.

A type of electronic commerce with agents is considered [10]. It is “consumer to business” electronic commerce.”Consumer-business” commerce type is rarely used by users in web because of the high transaction costs.

So the main purpose of using agents on this system is to combine needs and preferences of the buyers in a common decision, how the communication will be among the buyers in a group, and how the buyers interact with the sellers in web. So that multi-agent framework model was developed.

This called Buyer Collective Purchasing (BCP). So people can benefit from an easy collective purchasing system in a grouping behavior.

So, this agent system’s steps are buyer invitation, product description, combining needs…etc. In the agent system, the each agent has different roles. But they sometimes collaborate to process together. Some of them get the needs and preferences of the users; the others record the needs and purchasing, some of them make offerings to the candidates.

Also agents can work in mobile area [13]. This mobile agent based system is to support “business to consumer” electronic commerce type (e-commerce) and mobile commerce (m-commerce). First of all, customers or candidates must determine their wishes or specifications to an agent server by web browser. This server activates mobile agents for doing shopping tasks. There is a mathematical model to realize the consumer’s shopping wishes and decide what to do. And then the agents take these requirements or purchasing information to process and deliver the request result to the consumer again.

In fact m-commerce is the extension of the e-commerce in wireless network technologies. Consumers not only buy products by mobile commerce. M-commerce also offers location based services to the consumers. So that a mobile based agent system may be efficient and easy to do all these things. In this system, agents can communicate with each other by collaborating for processing especially; to buy products.

These agents also;

- Search some information. For example; to compare product prices on web to buy them.   Customers may want to search the product with lowest price.
- Mobile agents can make some routine purchases.

Mobile agents also use artificial intelligence for negotiations on internet. Negotiation part in E-commerce is usually doing manually but it’s too hard to handle so now agents are also using in this area. For negotiation [11], a large amount of research has been conducted to develop negotiation protocols and mechanisms for e-marketplaces, existing negotiation mechanisms are weak in dealing with complex and dynamic negotiation spaces often found in e-commerce. The new agents use a novel knowledge discovery method and a probabilistic negotiation decision making mechanism to become better the performance of negotiation agents. The agents use their recorded history files to improve negotiation performance. They use some algorithm to find best choice in negotiation.

If you use web the biggest problem is security this problem also take his place in e-Commerce platforms. So the most of the agents are programmed to secure the shopping. Some mobile agents are programmed for this [12]. These agents are being considered to have a secure platform in an open environment. The main purpose is to protect the mobile agents from open environment attacks to use the electronic commerce safely. So, protecting transactions are made in two applications;

- To protect the host from agent attacks
- To protect the agents from the host attacks which are malicious.

But in this article, mostly the second application is considered. First of all a security framework is created. This framework will protect the agents from the attacks. Namely this framework allows agents secure processing on hosts by restricting the access level of agents. So the agents cannot perform unsecure computations on not secure hosts. Agents and E-commerce grow up rapidly in semantic web [14] Semantic web technology is an extension of the World Wide Web .It includes semantics of information and services on internet. It enables to satisfy the people and machine on web.

The main aim of the agents in this presentation, how the agents that are related to the electronic commerce, will take part in the web and in the semantic web. And how can apply electronic commerce with agents into the semantic web technology.

Today, there is much more information on web. But suitable and meaningful information must be extracted easily, quickly and effectively. So in the next generation of web, an automatic system will be developed that will do everything on the web, instead of people by an agent.

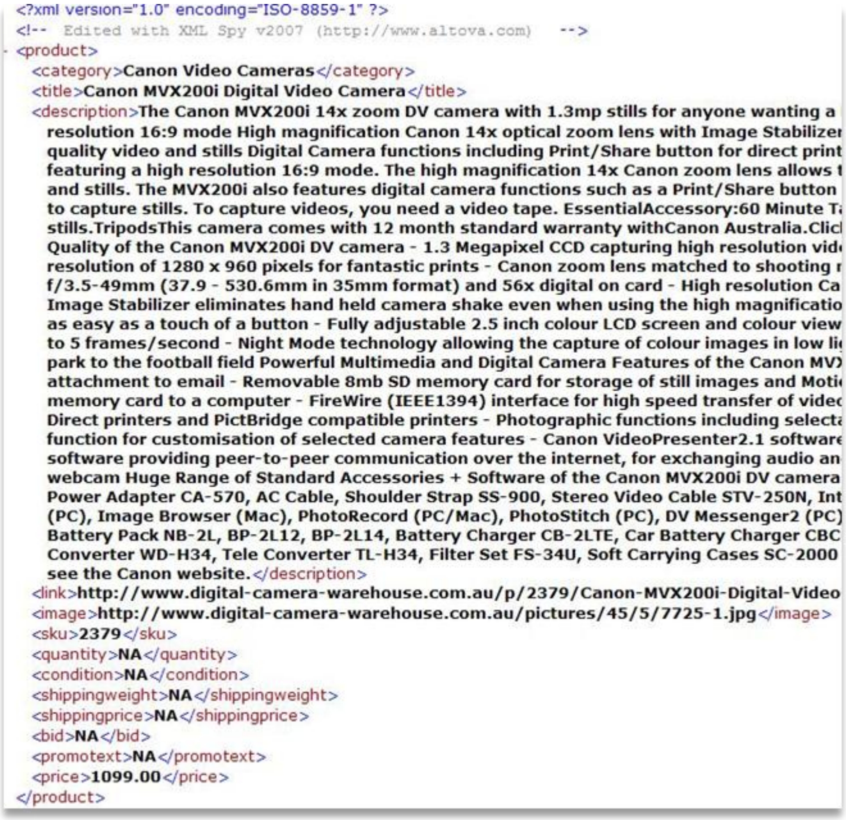
For example; reading, writing, searching information or making a new one...etc. in semantic web communication that are between user and company, interaction between agents is done automatically. Real transaction persons are not affected. So the productivity increased.

All in all, agent must be use in e-commerce, moreover must use in the entire web.

3 PRODUCT CATEGORY IDENTIFICATION PROBLEM

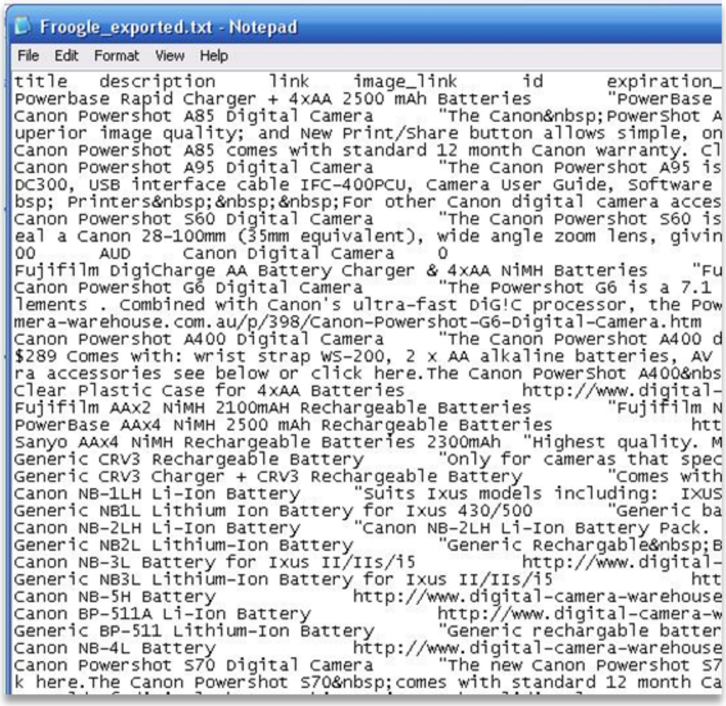
The systems embodying the information to itself that the member shops send, appears to be the biggest problem. Because of each member shops having the possibility of sending inaccurate information about the products with their categorization system, they need to be controlled with an autonomous and proactive intelligent mechanism.

The product portfolio of e-commerce MALL sites has comprised member stores. When transferring product information from member stores to MALL, various technologies have been used. The information is mostly transferred either by using web services or different remoting infrastructure in XML format (Figure 3.1.A) [15]. The primary goal of XML is to provide a marking text component and use such data for exchange among information sources [16]. Text based feeds (Figure 3.1.B) also have been used instead of XML.



```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!--  Edited with XML Spy v2007 (http://www.altova.com)   -->
- <product>
   <category>Canon Video Cameras</category>
   <title>Canon MVX200i Digital Video Camera</title>
   <description>The Canon MVX200i 14x zoom DV camera with 1.3mp stills for anyone wanting a
     resolution 16:9 mode High magnification Canon 14x optical zoom lens with Image Stabilizer
     quality video and stills Digital Camera functions including Print/Share button for direct print
     featuring a high resolution 16:9 mode. The high magnification 14x Canon zoom lens allows t
     and stills. The MVX200i also features digital camera functions such as a Print/Share button
     to capture stills. To capture videos, you need a video tape. EssentialAccessory:60 Minute Ta
     stills.TripodsThis camera comes with 12 month standard warranty withCanon Australia.Clicl
     Quality of the Canon MVX200i DV camera - 1.3 Megapixel CCD capturing high resolution vid
     resolution of 1280 x 960 pixels for fantastic prints - Canon zoom lens matched to shooting r
     f/3.5-49mm (37.9 - 530.6mm in 35mm format) and 56x digital on card - High resolution Ca
     Image Stabilizer eliminates hand held camera shake even when using the high magnificatio
     as easy as a touch of a button - Fully adjustable 2.5 inch colour LCD screen and colour view
     to 5 frames/second - Night Mode technology allowing the capture of colour images in low li
     park to the football field Powerful Multimedia and Digital Camera Features of the Canon MV
     attachment to email - Removable 8mb SD memory card for storage of still images and Moti
     memory card to a computer - FireWire (IEEE1394) interface for high speed transfer of videc
     Direct printers and PictBridge compatible printers - Photographic functions including selecta
     function for customisation of selected camera features - Canon VideoPresenter2.1 software
     software providing peer-to-peer communication over the internet, for exchanging audio an
     webcam Huge Range of Standard Accessories + Software of the Canon MVX200i DV camera
     Power Adapter CA-570, AC Cable, Shoulder Strap SS-900, Stereo Video Cable STV-250N, Int
     (PC), Image Browser (Mac), PhotoRecord (PC/Mac), PhotoStitch (PC), DV Messenger2 (PC)
     Battery Pack NB-2L, BP-2L12, BP-2L14, Battery Charger CB-2LTE, Car Battery Charger CBC
     Converter WD-H34, Tele Converter TL-H34, Filter Set FS-34U, Soft Carrying Cases SC-2000
     see the Canon website.</description>
   <link>http://www.digital-camera-warehouse.com.au/p/2379/Canon-MVX200i-Digital-Video
   <image>http://www.digital-camera-warehouse.com.au/pictures/45/5/7725-1.jpg</image>
   <sku>2379</sku>
   <quantity>NA</quantity>
   <condition>NA</condition>
   <shippingweight>NA</shippingweight>
   <shippingprice>NA</shippingprice>
   <bid>NA</bid>
   <promotext>NA</promotext>
   <price>1099.00</price>
</product>
```

A- XML-based product feed example for PriceRunner



```
title   description     link    image_link      id      expiration_
Powerbase Rapid Charger + 4xAA 2500 mAh Batteries       "PowerBase
Canon Powershot A85 Digital Camera      "The Canon&nbsp;PowerShot A
uperior image quality; and New Print/Share button allows simple, on
Canon Powershot A85 comes with standard 12 month Canon warranty. Cl
Canon Powershot A95 Digital Camera      "The Canon Powershot A95 is
DC300, USB interface cable IFC-400PCU, Camera User Guide, Software
bsp; Printers&nbsp;&nbsp;&nbsp;For other Canon digital camera acces
Canon Powershot S60 Digital Camera      "The Canon Powershot S60 is
eal a Canon 28-100mm (35mm equivalent), wide angle zoom lens, givin
00      AUD     Canon Digital Camera    0
Fujifilm DigiCharge AA Battery Charger & 4xAA NiMH Batteries    "Fu
Canon Powershot G6 Digital Camera       "The Powershot G6 is a 7.1
lements . Combined with Canon's ultra-fast DiG!C processor, the Pow
mera-warehouse.com.au/p/398/Canon-Powershot-G6-Digital-Camera.htm
Canon Powershot A400 Digital Camera     "The Canon Powershot A400 d
$289 Comes with: wrist strap WS-200, 2 x AA alkaline batteries, AV
ra accessories see below or click here.The Canon PowerShot A400&nbs
Clear Plastic Case for 4xAA Batteries           http://www.digital-
Fujifilm AAx2 NiMH 2100mAH Rechargeable Batteries       "Fujifilm N
PowerBase AAx4 NiMH 2500 mAh Rechargeable Batteries             htt
Sanyo AAx4 NiMH Rechargeable Batteries 2300mAh  "Highest quality. M
Generic CRV3 Rechargeable Battery       "Only for cameras that spec
Generic CRV3 Charger + CRV3 Rechargeable Battery        "Comes with
Canon NB-1LH Li-Ion Battery     "Suits Ixus models including:  IXUS
Generic NB1L Lithium Ion Battery for Ixus 430/500       "Generic ba
Canon NB-2LH Li-Ion Battery     "Canon NB-2LH Li-Ion Battery Pack.
Generic NB2L Lithium-Ion Battery        "Generic Rechargable&nbsp;B
Canon NB-3L Battery for Ixus II/IIs/i5          http://www.digital-
Generic NB3L Lithium-Ion Battery for Ixus II/IIs/i5             htt
Canon NB-5H Battery             http://www.digital-camera-warehouse
Canon BP-511A Li-Ion Battery            http://www.digital-camera-w
Generic BP-511 Lithium-Ion Battery      "Generic rechargable batter
Canon NB-4L Battery             http://www.digital-camera-warehouse
Canon Powershot S70 Digital Camera      "The new Canon Powershot S7
k here.The Canon Powershot S70&nbsp;comes with standard 12 month Ca
```

B- Tab delimited text based product feed example for Froogle

Figure 3.1 Technologies have been used to transfer product information from member stores to MALLs (A-B)

Whatever type of format has been chosen for the feed, the given information that it contains is exactly the same. It is important that the feed contains the information of each product in itself and that the MALL database is updated by the usage of this information.

There needs to be mechanism which will be able to find out the product names of these feeds that are sent in the system and match them. If there exists an undefined product in the system, than with the help of the products explanation information in the feed, we will be able to find out the product to which category to match. We must not trust to the category information that the e-shop sends the feed and the system must prefer the category proving method in order that there are no replication records and data integrity must be provided in the database.

To analyze product category identification problem, clustering and hashing methods are used. To identify the product's category, we must find out how close are the sent keywords to the keywords that are used in the products' explanations of the defined products by grouping them. The preferred method for the process of dividing into groups is clustering.

3.1 Clustering Methods

Clustering can be considered as the most important unsupervised learning method; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data.

A loose definition of clustering could be "the process of organizing objects into groups whose members are similar in some way".

A *cluster* is therefore a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters as shown in Figure 3.2.

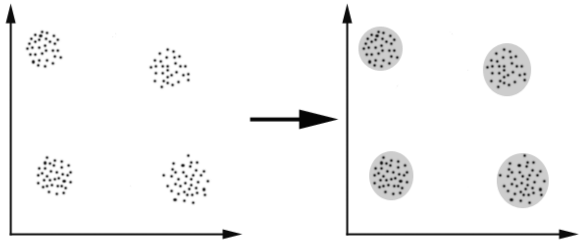


Figure 3.2 Clusters

In this case we easily identify the 4 clusters into which the data can be divided; the similarity criterion is *distance*: two or more objects belong to the same cluster if they are “close” according to a given distance (in this case geometrical distance). This is called *distance-based clustering*.

Another kind of clustering is *conceptual clustering*: two or more objects belong to the same cluster if this one defines a concept *common* to all that objects. In other words, objects are grouped according to their fit to descriptive concepts, not according to simple similarity measures.

3.1.1 Clustering Algorithms

Clustering algorithms may be classified as follows below:

- Exclusive Clustering
- Overlapping Clustering
- Hierarchical Clustering
- Probabilistic Clustering

In the first case; exclusive clustering, data are grouped in an exclusive way, so that if a certain datum belongs to a definite cluster then it could not be included in another cluster. A simple example of that is shown in the Figure 3.3 below, where the separation of points is achieved by a straight line on a bi-dimensional plane.

On the contrary the second type, the overlapping clustering, uses fuzzy sets to cluster data, so that each point may belong to two or more clusters with different degrees of membership. In this case, data will be associated to an appropriate membership value.

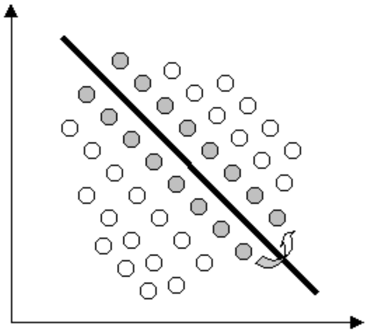


Figure 3.3 Overlapping Clusters

Instead, a hierarchical clustering algorithm is based on the union between the two nearest clusters. The beginning condition is realized by setting every datum as a cluster. After a few iterations it reaches the final clusters wanted.

Finally, the last kind of clustering uses a completely probabilistic approach.

Traditional Clustering Algorithms:

- K-means
- Fuzzy C-means
- Hierarchical clustering
- Mixture of Gaussians

Each of these algorithms belongs to one of the clustering types listed above. So that, K-means is an *exclusive clustering* algorithm, Fuzzy C-means is an *overlapping clustering* algorithm, Hierarchical clustering is obvious and lastly Mixture of Gaussian is a *probabilistic clustering* algorithm.

3.1.2 K-Means Clustering

K-means [17] is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result.

So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more.

The *K*-means algorithm assigns each point to the cluster whose center (also called centroid) is nearest. The center is the average of all the points in the cluster — that is, its coordinates are the arithmetic mean for each dimension separately over all the points in the cluster.

Finally, this algorithm aims at minimizing an *objective function*, in this case a squared error function. The objective function

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 ,$$

Where $\left\| x_i^{(j)} - c_j \right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre c_j, is an indicator of the distance of the *n* data points from their respective cluster centres.

The algorithm is composed of the following steps

- Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
- Assign each object to the group that has the closest centroid.
- When all objects have been assigned, recalculate the positions of the K centroids.

- Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centers. The k-means algorithm can be run multiple times to reduce this effect.

3.1.3 Fuzzy C-Means Clustering

Fuzzy c-means (FCM) is a method of clustering which allows one piece of data belongs to two or more clusters. This method [18,19] is frequently used in pattern recognition. It is based on minimization of the following objective function:

$$J_m = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}^{m} \left\|x_i - c_j\right\|^2 , \quad 1 \le m < \infty$$

where m is any real number greater than 1, u_{ij} is the degree of membership of x_i in the cluster j, x_i is the ith of d-dimensional measured data, c_j is the d-dimension center of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the center.

Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership u_{ij} and the cluster centers c_j by:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{\left\|x_i - c_j\right\|}{\left\|x_i - c_k\right\|}\right)^{\frac{2}{m-1}}} , \qquad c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

This iteration will stop when $\max_{ij}\left\{\left|u_{ij}^{(k+1)} - u_{ij}^{(k)}\right|\right\} < \varepsilon$, where ε is a termination criterion between 0 and 1, whereas k are the iteration steps. This procedure converges to a local minimum or a saddle point of J_m.

The algorithm is composed of the following steps:

- ✓ Initialize U=[u_{ij}] matrix, $U^{(0)}$
- ✓ At k-step: calculate the centers vectors $C^{(k)}$=[c_j] with $U^{(k)}$

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

- ✓ Update $U^{(k)}$, $U^{(k+1)}$

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$

- ✓ If || $U^{(k+1)}$ - $U^{(k)}$||< ε then STOP; otherwise return to step 2.

The preferred clustering methods inside AGBEP are Fuzzy C-Means and K-Means clustering algorithms. But we won’t be able to use the explanations that come from feed as it’s sent. Because of the words from the explanations are being alphanumeric, the clustering process becomes much more difficult. In order to prevent this difficulty and turn the chosen keywords to numeric values, the hashing methods are being used.

3.2 Hashing Methods

The hashing method is the changing the form of a string of characters into a usually shorter fixed-length value or key that represents the original string [20]. The hashing method is used to index and retrieve items in a database because by this way, we find the item much faster than using the shorter hashed key in order to find it using the original value. We also use the hashing method in many encryption algorithms.

3.2.1 Hash Function

The hashing algorithm is called the *hash function* [20] (the term has the origin from the idea that the resulting hash value can be thought of as a "mixed up" version of the represented value). Additionally, for faster data retrieval, hashing is also used to encrypt and decrypt digital signatures (in order to verify the message senders and receivers). The digital signature in the message is transformed with the hash function and then both the hashed value (known as a message-digest) and by this way the signatures are sent in separate transmissions to the receiver. The sender uses the same hash function, and by this way the receiver derives a message-digest from the signature and compares it with the message-digest it also received. They should be the same.

By hash function we index the original value or key and then use later each time the data associated with the value or key is to be retrieved. So, the hashing method is always a one-way operation. There's no need to "reverse engineering" the hash function by analyzing the hashed values. Actually, the ideal hash function cannot be obtained by such an analysis. The hash function also should not produce the same hash value from two different inputs. If it does, we call it as a *collision*. A hash function which offers an extremely low risk of collision is highly acceptable.

Some hash functions that are used as follows:

- ✓ *The division-remainder method*: In this method, the size of the number of items in the table is estimated. After, this number is used as a divisor into each original value or key to extract a quotient and a remainder. The remainder is the hashed value. (Because this method may produce a number of collisions, any search mechanism would have to be able to recognize a collision and offer an alternate search mechanism.)
- ✓ *Folding*: In this method we divide the original value (digits in this case) into several parts, adds the parts together, and then use the last four digits (or some other arbitrary number of digits that will work) as the hashed value or key.
- ✓ *Radix transformation*: If the value or key is digital, the number base (or radix) can be transformed resulting in a different sequence of digits. (For example, a decimal numbered key could be changed into a hexadecimal numbered key.) The high-order digits can be taken to fit a hash value of uniform length.
- ✓ *Digit rearrangement*: In this method, we take part of the original value or key such as digits in positions 3 through 6, and by reversing their order, then we use that sequence of digits as the hash value or key.

In terms of security, the hashing is a method of taking data, encrypting it, and creating unpredictable, irreversible output. There are many different types of hashing algorithms. MD2, MD4, MD5, SHA (Secure Hash Algorithms), SHA-1, SHA-256 and RIPEMD are examples of hashing algorithms.

3.2.2 MD Series

The message-digest algorithms called MD2, MD4, and MD5 are developed by Rivest [21]. They are the digital signature applications where a large message has to be "compressed" in a secure manner before being signed with the private key. The whole three algorithms take a message of arbitrary length and produce a 128-bit message digest. Since their structures are somewhat similar, the design of MD2 is quite different from that of MD4 and MD5. MD2 was optimized for 8-bit machines, whereas MD4 and MD5 were aimed at 32-bit machines.

3.2.3 MD5 Hashing Algorithm

MD5 (Message-Digest algorithm 5) is a well-known cryptographic hash function and has a hash value of 128-bit resulting. We observed that MD5 is widely used in security-related applications, and it is also used to check the integrity of files frequently.

We consider the MD5 value of file as a highly reliable fingerprint that can be used to verify the integrity of the file's contents. When the single bit value in the file is modified, the MD5 value for the file will completely change. The act of forging of a file in a way that causes MD5 to generate the same result as that for the original file is considered to be extremely difficult.

The set of MD5 checksums for critical system, application, and data files provides a compact way to store information for use during periodic integrity checks of those files.

Pseudo code of the MD5 algorithm follows as [21];

```
//Note: All variables are unsigned 32 bits and wrap modulo 2^32 when
calculating
var int[64] r, k

//r specifies the per-round shift amounts
r[ 0..15] := {7, 12, 17, 22,  7, 12, 17, 22,  7, 12, 17, 22,  7, 12,
17, 22}
r[16..31] := {5,  9, 14, 20,  5,  9, 14, 20,  5,  9, 14, 20,  5,  9,
14, 20}
r[32..47] := {4, 11, 16, 23,  4, 11, 16, 23,  4, 11, 16, 23,  4, 11,
16, 23}
r[48..63] := {6, 10, 15, 21,  6, 10, 15, 21,  6, 10, 15, 21,  6, 10,
15, 21}

//Use binary integer part of the sines of integers (Radians) as
constants:
for i from 0 to 63
    k[i] := floor(abs(sin(i + 1)) × (2 pow 32))

//Initialize variables:
var int h0 := 0x67452301
var int h1 := 0xEFCDAB89
var int h2 := 0x98BADCFE
var int h3 := 0x10325476

//Pre-processing:
append "1" bit to message
append "0" bits until message length in bits ≡ 448 (mod 512)
append bit (bit, not byte) length of unpadded message as 64-bit little-
endian integer to message

//Process the message in successive 512-bit chunks:
for each 512-bit chunk of message
    break chunk into sixteen 32-bit little-endian words w[i], 0 ≤ i ≤
15

    //Initialize hash value for this chunk:
    var int a := h0
    var int b := h1
    var int c := h2
    var int d := h3

    //Main loop:
    for i from 0 to 63
        if 0 ≤ i ≤ 15 then
```

```
            f := (b and c) or ((not b) and d)
            g := i
        else if 16 ≤ i ≤ 31
            f := (d and b) or ((not d) and c)
            g := (5×i + 1) mod 16
        else if 32 ≤ i ≤ 47
            f := b xor c xor d
            g := (3×i + 5) mod 16
        else if 48 ≤ i ≤ 63
            f := c xor (b or (not d))
            g := (7×i) mod 16

        temp := d
        d := c
        c := b
        b := b + leftrotate((a + f + k[i] + w[g]) , r[i])
        a := temp

    //Add this chunk's hash to result so far:
    h0 := h0 + a
    h1 := h1 + b
    h2 := h2 + c
    h3 := h3 + d

var int digest := h0 append h1 append h2 append h3 //(expressed as
little-endian)
```

3.2.4 SHA Hash Functions

We define the SHA hash functions [22] as the five cryptographic hash functions designed by the National Security Agency (NSA) and published by the NIST as a U.S. Federal Information Processing Standard. SHA is the opening of Secure Hash Algorithm. The hash algorithms calculate a fixed-length digital representation (known as a message digest) of an input data sequence (the message) of any length. They are called "secure" when (in the words of the standard), "it is computationally infeasible to:

1. Find a message that corresponds to a given message digest, or
2. Find two different messages that produce the same message digest.

Any change to a message wills, with a very high probability, result in a different message digests."

The five algorithms are denoted SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512. The latter four variants are sometimes collectively referred to as SHA-2. SHA-1 produces a message digest that is 160 bits long; the numbers in the other four algorithms' names denote the bit length of the digest they produce.

SHA-0 published in 1993 as the Secure Hash Standard, FIPS PUB 180 by National Institute of Standards and Technology.

SHA-1 published in 1995 in FIPS PUB 180-1.

SHA-256, SHA-384 and SHA-512 first published in 2001 as draft FIPS PUB 180-2 and released as official standard in 2002.

SHA-224 published in 2004 as change notice for FIPS PUB 180-2.

Pseudo code for the SHA-256 algorithm follows [23]. Note the great increase in mixing between bits of the w(16..63) words compared to SHA-1.

```
Note: All variables are unsigned 32 bits and wrap modulo $2^{32}$ when
calculating

Initialize variables
(first 32 bits of the fractional parts of the square roots of the first
8 primes 2..19):
h0 := 0x6a09e667
h1 := 0xbb67ae85
h2 := 0x3c6ef372
h3 := 0xa54ff53a
h4 := 0x510e527f
h5 := 0x9b05688c
h6 := 0x1f83d9ab
h7 := 0x5be0cd19

Initialize table of round constants
(first 32 bits of the fractional parts of the cube roots of the first
64 primes 2..311):
k[0..63] :=
   0x428a2f98, 0x71374491, 0xb5c0fbcf, 0xe9b5dba5, 0x3956c25b,
0x59f111f1, 0x923f82a4, 0xab1c5ed5,
   0xd807aa98, 0x12835b01, 0x243185be, 0x550c7dc3, 0x72be5d74,
0x80deb1fe, 0x9bdc06a7, 0xc19bf174,
   0xe49b69c1, 0xefbe4786, 0x0fc19dc6, 0x240ca1cc, 0x2de92c6f,
0x4a7484aa, 0x5cb0a9dc, 0x76f988da,
   0x983e5152, 0xa831c66d, 0xb00327c8, 0xbf597fc7, 0xc6e00bf3,
0xd5a79147, 0x06ca6351, 0x14292967,
   0x27b70a85, 0x2e1b2138, 0x4d2c6dfc, 0x53380d13, 0x650a7354,
0x766a0abb, 0x81c2c92e, 0x92722c85,
   0xa2bfe8a1, 0xa81a664b, 0xc24b8b70, 0xc76c51a3, 0xd192e819,
0xd6990624, 0xf40e3585, 0x106aa070,
   0x19a4c116, 0x1e376c08, 0x2748774c, 0x34b0bcb5, 0x391c0cb3,
0x4ed8aa4a, 0x5b9cca4f, 0x682e6ff3,
   0x748f82ee, 0x78a5636f, 0x84c87814, 0x8cc70208, 0x90befffa,
0xa4506ceb, 0xbef9a3f7, 0xc67178f2

Pre-processing:
append the bit '1' to the message
append k bits '0', where k is the minimum number >= 0 such that the
resulting message
    length (in bits) is congruent to 448 (mod 512)
append length of message (before pre-processing), in bits, as 64-bit
big-endian integer

Process the message in successive 512-bit chunks:
break message into 512-bit chunks
for each chunk
    break chunk into sixteen 32-bit big-endian words w[0..15]

    Extend the sixteen 32-bit words into sixty-four 32-bit words:
    for i from 16 to 63
        s0 := (w[i-15] rightrotate 7) xor (w[i-15] rightrotate 18) xor
(w[i-15] rightshift 3)
        s1 := (w[i-2] rightrotate 17) xor (w[i-2] rightrotate 19) xor
(w[i-2] rightshift 10)
        w[i] := w[i-16] + s0 + w[i-7] + s1

    Initialize hash value for this chunk:
    a := h0
    b := h1
    c := h2
    d := h3
    e := h4
    f := h5
    g := h6
    h := h7

    Main loop:
    for i from 0 to 63
```

```
        s0 := (a rightrotate 2) xor (a rightrotate 13) xor (a
rightrotate 22)
        maj := (a and b) xor (a and c) xor (b and c)
        t2 := s0 + maj
        s1 := (e rightrotate 6) xor (e rightrotate 11) xor (e
rightrotate 25)
        ch := (e and f) xor ((not e) and g)
        t1 := h + s1 + ch + k[i] + w[i]

        h := g
        g := f
        f := e
        e := d + t1
        d := c
        c := b
        b := a
        a := t1 + t2

    Add this chunk's hash to result so far:
    h0 := h0 + a
    h1 := h1 + b
    h2 := h2 + c
    h3 := h3 + d
    h4 := h4 + e
    h5 := h5 + f
    h6 := h6 + g
    h7 := h7 + h

Produce the final hash value (big-endian):
digest = hash = h0 append h1 append h2 append h3 append h4 append h5
append h6 append h7
```

4 AGENT TECHNOLOGIES

We may consider the agents as an example prototype that may improve on current methods for conceptualizing, designing and implementing software systems, and secondly they may be the solution to the legacy software integration problem.

4.1 Definition of an Agent

The agent is used in different technologies such as, in artificial intelligence [24, 25] , databases, operating systems and computer networks literature. Although there is no single definition of an agent [26], we agree that an agent is essentially a special software component that has autonomy that provides an interoperable interface to an arbitrary system and/or behaves like a human agent, working for some clients in pursuit of its own agenda. An agent system can be based on a solitary agent working within an environment and if necessary interacting with its users, and usually they consist of multiple agents. We call them multi-agent systems (MAS) and they can model complex systems and introduce the possibility of agents having common or conflicting goals. Also these agents can interact with each other both indirectly (by acting on the environment) or directly (via communication and negotiation). Agents can cooperate for mutual benefit or can compete to serve their own interests.

The agent is autonomous, because when it operates; the humans are not involved directly and it has control over its actions and internal state. Secondly the agent is social, because it cooperates with humans or other agents in order to achieve its tasks. Third of all the agent is reactive, because it perceives its environment and responds in a timely fashion to changes that occur in the environment. The last of all the agent is proactive, because it does not simply act in response to its environment but is able to exhibit goal-directed behavior by taking initiative.

Moreover, the agent can be mobile when it is necessary, because it has an ability to travel between different nodes in a computer network. It has a correctness that provides the certainty that it will not deliberately communicate false information. It is helpful, because it always tries to perform what is asked of it. It is logical, because it always acts in order to achieve its goals and never to prevent its goals being achieved, and it has an ability of learning, because it adapts itself to fit its environment and to the desires of its users.

4.2 Agent Architectures

The agent architectures are mechanisms that are fundamental and they underly the autonomous components that support effective behavior in real-world, dynamic and open environments (Figure 4.1). The agent-based computing efforts focused on the development of intelligent agent architectures, and several lasting styles of architecture are established in the early years. These differ from purely reactive (or behavioral) architectures that operate in a simple stimulus–response fashion, such as those based on the *subsumption architecture* of Brooks [27] , to more deliberative architectures that reason about their actions, such as those based on the belief desire intention (BDI) model [28]. Among these two, lie hybrid combinations of both, or layered architectures, which attempt to involve both reaction and deliberation in an effort to adopt the best of each approach. The whole agent architectures are divided into four main groups: logic based, reactive, BDI and layered architectures. *Logic-based* (symbolic) architectures draw their foundation.

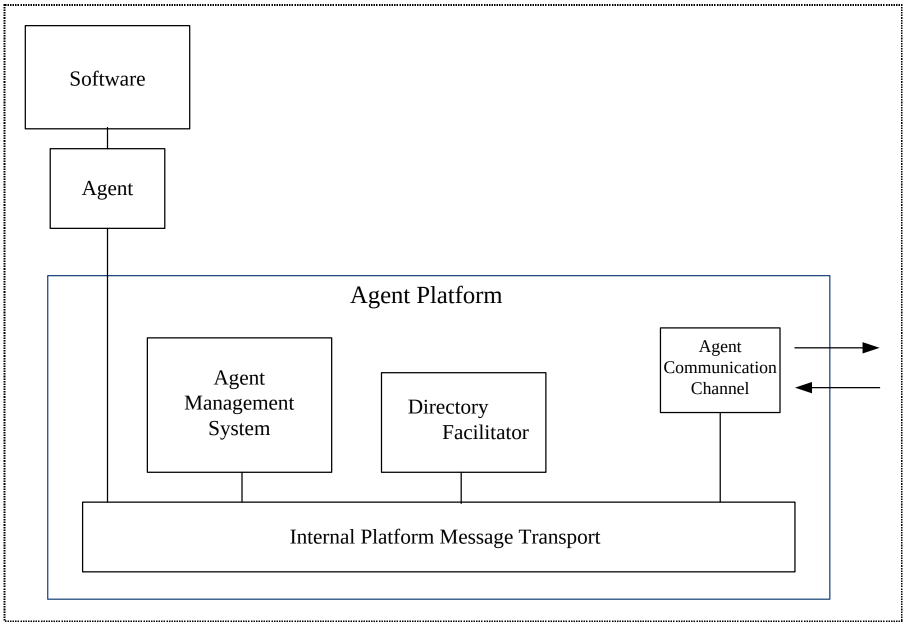


Figure 4.1 Model of an Agent Platform

4.3 Communication and Coordination

The multi-agent systems have a key component called communication. The agents need the ability to communicate with users, with system resources, and with each other if they need to cooperate, collaborate, and negotiate and so on. Generally, the agents interact with each other by using some special communication languages, which are called agent communication languages, that rely on speech act theory and that provide a separation between the communicative acts and the content language. In the history the first agent communication language with a broad uptake was KQML [29]. KQML was developed in the early 1990s and it was a part of the US government's ARPA Knowledge Sharing Effort. Its language and protocol was for exchanging information and knowledge that defines a number of performative verbs and allows message content to be represented in a first-order logic-like language called KIF [30].

Today's, the FIPA ACL is the most used and studied agent communication language and it incorporates many aspects of KQML. The first characteristics of FIPA ACL are the possibility of using different content languages and the management of conversations through predefined interaction protocols. The coordination is a process which the agents engage to help ensure that a community of individual agents acts in a coherent manner. The reasons for why multiple agents need to be coordinated are including: (1) the agents' goals may cause conflicts among agents' actions, (2) the agents' goals may be interdependent, (3) the agents may have different capabilities and different knowledge, and (4) the agents' goals may be more rapidly achieved if different agents work on each of them. Different approaches such as organizational structuring, contracting, multi-agent planning and negotiation handle the coordination among agents.

By the help of organizational structuring we get a framework for activity and interaction through the definition of roles, communication paths and authority relationships [31]. In order to ensure the coherent behavior and resolving conflicts, the easiest way seems to consist of providing the group with an agent which has a wider perspective of the system, by that exploiting an organizational or hierarchical structure. We can call it the simplest coordination technique and it allows a classic master/slave or client / server architecture for task and resource allocation between slave agents by a master agent. The information is collected from the agents in the group by master controller, and it creates plans and assigns tasks to individual agents in order to ensure global coherence. Despite this approach can be a little impractical in realistic applications because to create such a central controller is very difficult, and also the centralized control, as in the master/slave technique, is in opposite to the decentralized nature of multi-agent systems.

The contract net protocol [32] is an important coordination technique for task and resource allocation between agents and determining organizational structure. This technique focuses on a decentralized market structure where agents can take on two roles, a manager and contractor. This approaches' basic term is to coordinate such as if an agent cannot solve an assigned problem using local resources/expertise, it will separate the problem into simpler compounds and try to find out the other willing agents with the necessary resources/expertise to solve these sub-problems. The contracting mechanisms in the below, solves the problem of assigning the sub-problems:

(1) Agreement announcement by the manager agent,

(2) The submission of bids by contracting agents in response to the announcement, and

(3) The act of evaluating the submitted bids by the contractor, which leads to awarding a sub-problem contract to the contractor with the most appropriate, bids (Figure 4.2).

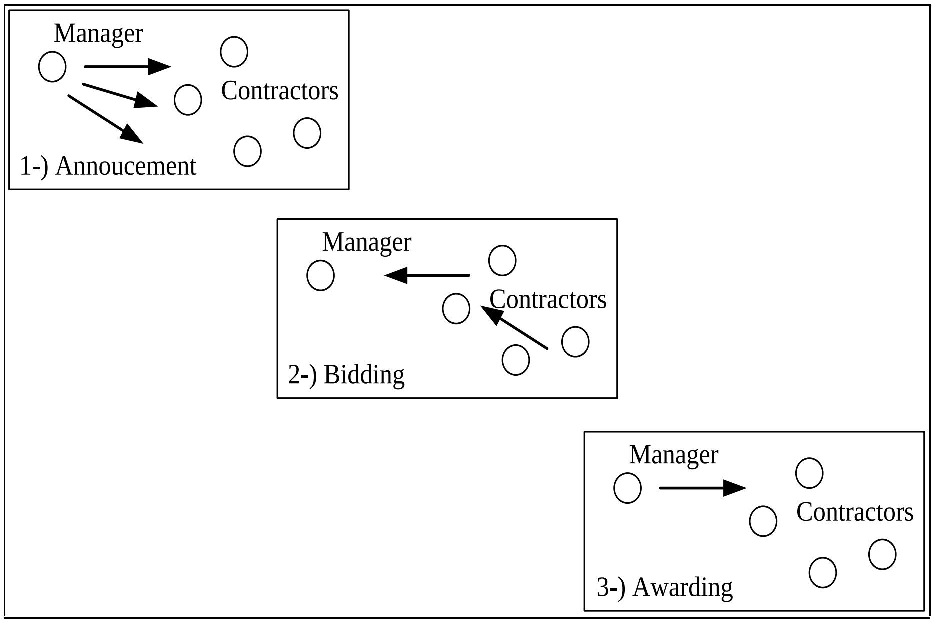


Figure 4.2 Phases of the Contract Net Protocol

We can also see the problem of coordinating agents as a planning problem. To prevent the inconsistent or conflicting actions and interactions, the agents can form a multi-agent plan that groups the items of the whole future actions and interactions that are required to reach their goals and interleave execution with additional planning and re-planning. The multi-agent planning is also being centralized or distributed. In approach of centralized *multi-agent planning*, there is a coordinating agent that is capable of receiving all partial or local plans from individual agents, and analyzing them to identify potential inconsistencies and conflicting interactions. After this, the coordinating agent modifies these partial plans and combines them into a multi-agent plan and by this, the conflicting interactions are eliminated. The distributed *multi-agent planning* has the idea of providing each agent with a model of other agents' plans. The agents share the information inside in order to build and update their individual plans and their models of other agents till all conflicts are solved.

The partial global planning unites the strengths of the organizational, planning, and contracting approaches by merging them with a single approach [33]. The ideal of this approach is to achieve the multi-agent planning benefits of detailed, situation-specific coordination while preventing extreme computation and communication costs.

This goal is achieved because the united organizational structures effectively prune the space of possible plans to keep the problem tractable. Moreover, the partial global planning views contracts as jointly held plans that specify future exchanges of tasks and results among agents. Because of this, in the approach of partial global planning, the coordination includes both sharing tasks and sharing results; both adhering to long-term organizational roles and reactively planning to achieve short-term objectives.

To coordinate the agents we rely on the Negotiation technique. Especially, the negotiation is a communication process of a group of agents to reach a mutually accepted agreement on some matter [34]. Depending on the behavior of the agents involved, the negotiation can be competitive or cooperative. *Competitive negotiation* is used when the agents have independent goals that interact with each other; they do not cooperate with each other or share information or willing to back down for the greater good. *Cooperative negotiation* is used when the agents have a common goal to achieve or a single task to execute. In that way, the multi-agent system has been generally designed to achieve a single global goal.

4.4 The Foundation for Intelligent, Physical Agents (FIPA)

Generally, the JADE is largely an implementation of the FIPA specifications [35] and it is highly dependent on the ideas generated through the specification process as expressed in the documents themselves. Since JADE has extended the FIPA model in several areas, the specifications do not provide complete coverage. But the fact of the core purpose of FIPA, JADE is being compliant remains in all aspects relating to interoperability.

The principles at the core of FIPA:

- The agent technologies gives us new terms of solving old and new problems;
- Some agent technologies have reached a considerable degree of maturity;
- Some agent technologies require standardization;
- The standardization of generic technologies has been shown to be possible and they can provide effective results by other standardization fora;

- We do not primarily concern on the standardization of the internal mechanics of agents, but rather we concern in the infrastructure and language required for open interoperation.

To point of time of the key achievements of FIPA are as follows:

- The set of standard specifications supporting inter-agent communication and key middleware services.
- An abstract architecture that provides an encompassing view across the entire FIPA2000 standards. This architecture underlies an incomplete reification as a Java Community Project known as the Java Agent Services (JAS) (JSR82).
- The well-specified and much-used agent communication language (FIPA-ACL), followed by a selection of content languages (e.g. FIPA-SL) and a set of key interaction protocols ranging from single message exchange to complex transactions.
- The several open source and commercial agent tool-kits with JADE and they are considered as the leading FIPA-compliant open source technology available today.
- The several projects outside FIPA such as the completed Agentcities project that created a global network of FIPA-compliant platforms and agent application services.
- An agent-specific extension of UML, known as AUML or Agent.

The Foundation for Intelligent Physical Agents (FIPA) [36] is an international non-profit association of companies and organizations that share the efforts to produce specifications of generic agent technologies. FIPA is regarded as not just as a technology for one application but as generic technologies for different application areas, and also not just as independent technologies but as a set of basic technologies that can be integrated by developers in order to make complex systems with a high degree of interoperability.

4.5 Preferred Framework: JADE

JADE (Java Agent DEvelopment Framework) [37] is a software framework to make easier the development of agent applications in compliance with the FIPA specifications for interoperable intelligent multi-agent systems. The goal of JADE is to simplify development while ensuring standard compliance through a comprehensive set of system services and agents.

The Java Agent Development Platform (JADE) is middleware designed to facilitate the development of multi-agent applications. Developed by Telecom Italia Labs in Italy, the software has been shared as open source since February 2000. JADE designed using java, providing interoperability between agents running on varied operating system, and can be used with any number of versions of java for both fixed and mobile devices. Because of this feature and its small footprint, JADE agents can run everywhere from powerful workstations to mobile phones.

JADE is written in Java language and is made by various Java packages, giving application programmers both ready-made pieces of functionality and abstract interfaces for custom, application dependent tasks. Java was the programming language of choice because of its many attractive features, particularly geared towards object-oriented programming in distributed heterogeneous environments; some of these features are Object Serialization, Reflection API and Remote Method Invocation (RMI).

Jade allows agents to cooperate and pass massages using FIPA-compliant message structures and simple set of API routines. In a JADE agent system, agents are able to register themselves and the services they can provide with a directory facilitator service, which then allows all agents to look up peers according to the services they provide. The directory facilitator also ensures the each agent is assigned a unique agent identifier (AID) that allows it to be located and identified as a massage recipient.

The message protocol utilized by JADE is the agent communication language message structure (Figure 4.3.)

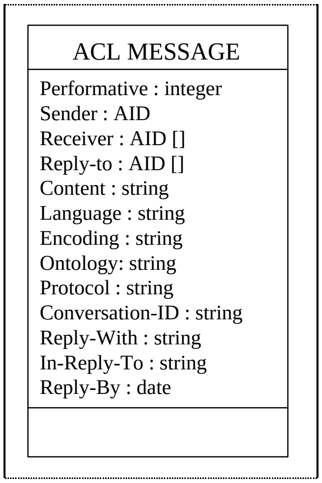


Figure 4.3 The ACL Message Structure

Of the fields that constitute the ACL message type, several are of use to sliding window scheduling agents:

- Sender : the agent from whom the message is being sent
- Receiver : the agent to whom the message is intended
- Content : the substance of the message

The ACL message structure provides the opportunity of for complex communication between agent, where agents negotiate with multiple peers using a variety of languages and message encoding techniques, and indicating their intensions with the inherent message performatives provided by ACL.

4.6 The Reasons to Select JADE as Agent Framework

Our reason for preferring Jade framework firstly is that because we can benefit from the experiences of previous agent based works. Also the other reasons are that the Jade framework is the best known agent framework and it is used with a high ratio in researches that are done related with this subject. We may prefer new generation agent frameworks so that the platform works with a higher performance and if we consider that especially the Microsoft based technologies (asp.NET, SQL Server 2005, NLB vs…) are used in the biggest part of the work than we may decide to use a net based framework.

5 AGENT BASED E-COMMERCE PLATFORM: AGBEP

As mentioned in chapter three, the duty of adding the information in the catalogues which the member shops send to the mall by using the agent technologies and to offer the best suitable infrastructure for an e-mall are the basic duties of AGBEP. The member shops and the customers roles inside the system needs are planned to give with the highest availability and the best performance. In order to give these services, servers are configured in different ways for different purposes.

As illustrated in Figure 5.1 the structure of AGBEP is directed by a front balancer at front end in order to resist overload. Visitors are transferred to an available web server where the load is not heavy. Web servers connect to the databases on the SQL 2005 servers at by the load balancer background of themselves in order to answer the needs of the users. Furthermore, there exists an Agent Server that makes possible the managerial structure and that keeps the agents above.

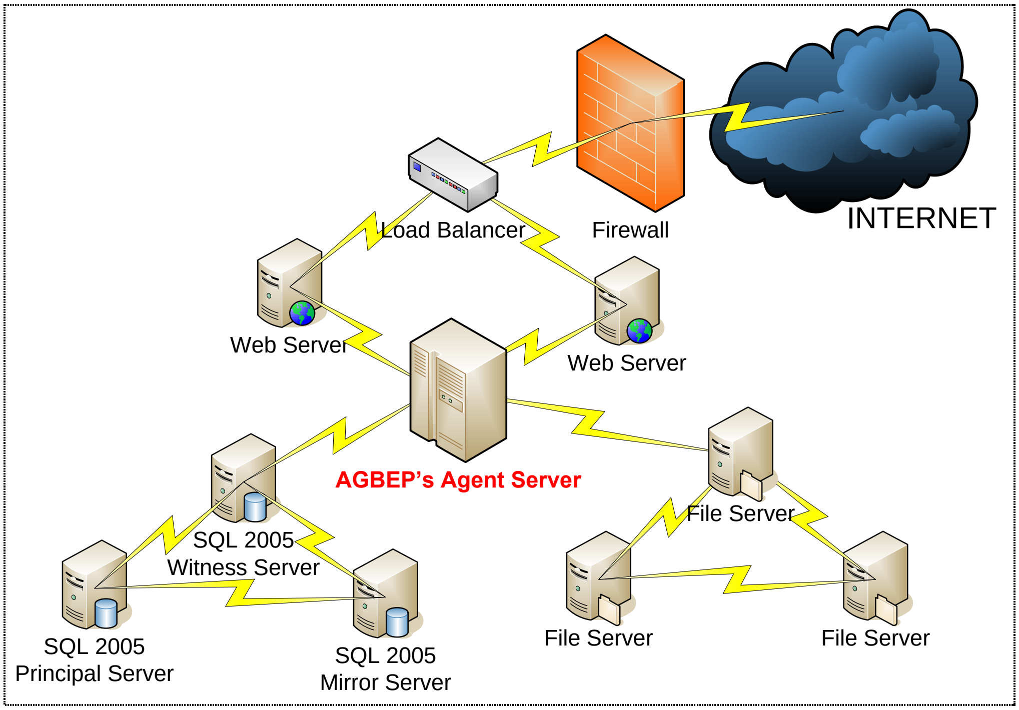


Figure 5.1 Overview of AGBEP

In order to analyze the physical structure detailed, we have to observe it in 4 different parts in according to duties and working styles. The first ones of these are the web servers that contain the AGBEP’s web interfaces and the server that makes the load balancing of these servers. The Internet Information Server (IIS) 6.0 service of the Windows Server 2003 is used in the hosting of the web pages. The Network Load Balancing which is one of the built-in services of Server 2003 is used for load balancing process also.

The second part in the system is a platform named Agent Server which makes possible the agents do their duties and work. The Agent Server where the Windows Server 2003 operating system is installed contains the JADE framework.

The parts beyond the physical structure are database servers and file servers. The database servers are the computers that use the operating system Windows Server 2003 and contain SQL Server 2005 Enterpriser Edition which Microsoft offers as the best suitable product for the Database Management Systems (DBMS).

The last part consists of the file servers that contain documents with contents such as feeds, product images, etc that the member shops send.

5.1 Intelligent Agents in AGBEP

The AGBEP’s Agent Server, which is located inside the physical structure, contains the Agents that are defined to work in different duties and the JADE framework which is an infrastructure [38] that makes these agents work. The agents that work on this server finishes the basic duties that have to be done in an autonomous way according to the duties’ content by sending messages and accessing to the servers among themselves.

The first entity of AGBEP is the *Site Administration Agent* (SAA) that is automatically appointed to the member stores by the system. SAA makes the managerial functions continue either by making a data transfer with the other agents or using the system's web services. However, the *Feed Delivery Agent* (FDA) includes the feeds to the system to be analyzed that keeps the product information that the member store sends. An *Inspector Agent* (IA) finds out the keywords by looking at the product descriptions at the feeds that are included to the system. A *Hashing Agent* (HA) then transfers the identified keywords through a hash function and registers to the system. Finally, *Clustering Agent* (CA) decides which product fits in which category within the system by performing a clustering study using these hashed values. Figure 5.2 illustrates the whole agent framework and its members.

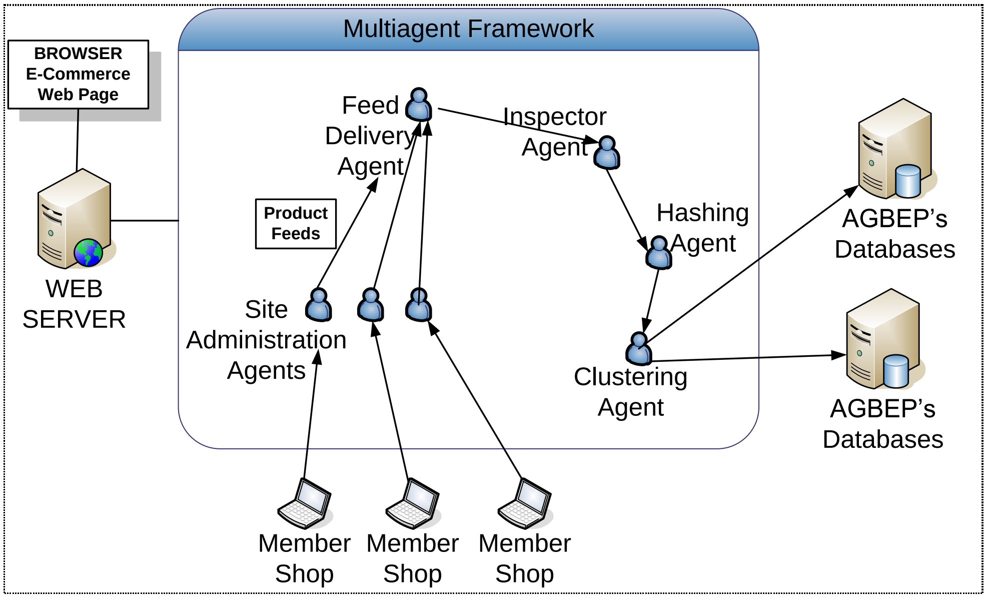


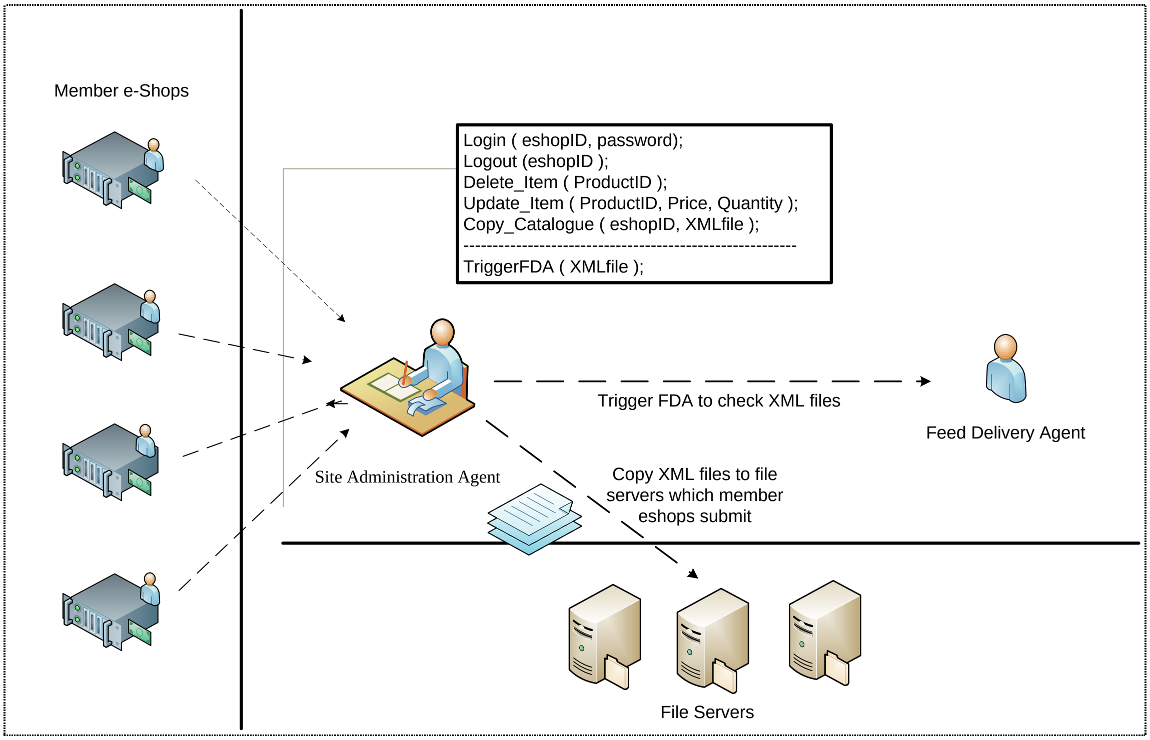
Figure 5.2 Agent Architecture in AGBEP

5.1.1 Site Administration Agent

As a member of the MALL structure, Site Administration Agent (SAA) is used to help the managerial functions of the store that is appointed. This is the starting point of the feed sending part of the managerial duties of the member store. It takes the feed that the store submits from the client computer to the Feed Delivery Agent and takes back a report of the operation as illustrated in Figure 5.3.

SSA gives us a safe entry by controlling the member shops' user name and password with mall records. It starts the logged e-shop users updating and deleting functions of their product catalog in the mall. It saves the XML feed over file server which the user sends that wants to load a new catalog and warns the Feed Delivery Agent to make the mistake and damage controls. By bringing the success or failure reports to the user which comes back from Feed Delivery Agent, it provides a safe logout from the system.

We may consider SAA as the starting point of the autonomous categorizing system of AGBEP which offers as a solution. Beyond it submits the most basic managerial duties, it's the most critical duty is to unite the XML document which the member shop sends with the member shop ID with the working time and by renaming it, it records them to the place which is allocated for the member shop over file servers. The information of the XML document which is recorded over file server is exceeded to the waiting status by transmitting to the Feed Delivery Agent with *TriggerFDA()* method. According to the information that comes from the control process that is made by FDA, it gives notice of whether the catalog that is sent to member shop is taken to action or whether there becomes a mistake.



```
Login ( eshopID, password);
Logout (eshopID );
Delete_Item ( ProductID );
Update_Item ( ProductID, Price, Quantity );
Copy_Catalogue ( eshopID, XMLfile );
--------------------------------------------------------
TriggerFDA ( XMLfile );
```

Member e-Shops
Site Administration Agent
Trigger FDA to check XML files
Feed Delivery Agent
Copy XML files to file servers which member eshops submit
File Servers

Figure 5.3 Site Administration Agent in Process

5.1.2 Feed Delivery Agent

Feed Delivery Agent (FDA) controls the feeds that are taken from SAA to the system. In the present system, the feeds are preferred in XML format. It is obliged to make the control of these files while they damage at the time of transfer until the mistake control is done at the XML tags.

After FDA checks the document whether it is reachable and accessible, it controls the opening and closing tags of the whole records in the XML file and it informs SSA.

First of all, FDA checks the XML document that is sent from the member shop is corrupted or not during copying. During this control it sets to work the *Check Accessibility()* method. This method gives notice of the result in a Boolean way whether it is successful or not by trying to access to all nods inside XML document. After its being understood that there is no problem in accessibility physically, it checks FDA, XML document by *CheckTags()* method whether it is in the format that AGBEP can use as a meaning. During this method, it uses the XSL document that is defined inside the system.

It takes the elements and attributes in the '*template.xsl*' document as a basis, it checks the whole nods in the sent Feed and it sends the result as a Boolean value. By controlling the results of both *Check Accessibility()* and *Check Tags()*, it decides whether the categorization process will be continued or not and it sets to work the *ReportToSAA()* and *TriggerIA()* methods. If the result is affirmative, it activates the next processes with *TriggerIA()* method. Whole process illustrated as in Figure 5.4.

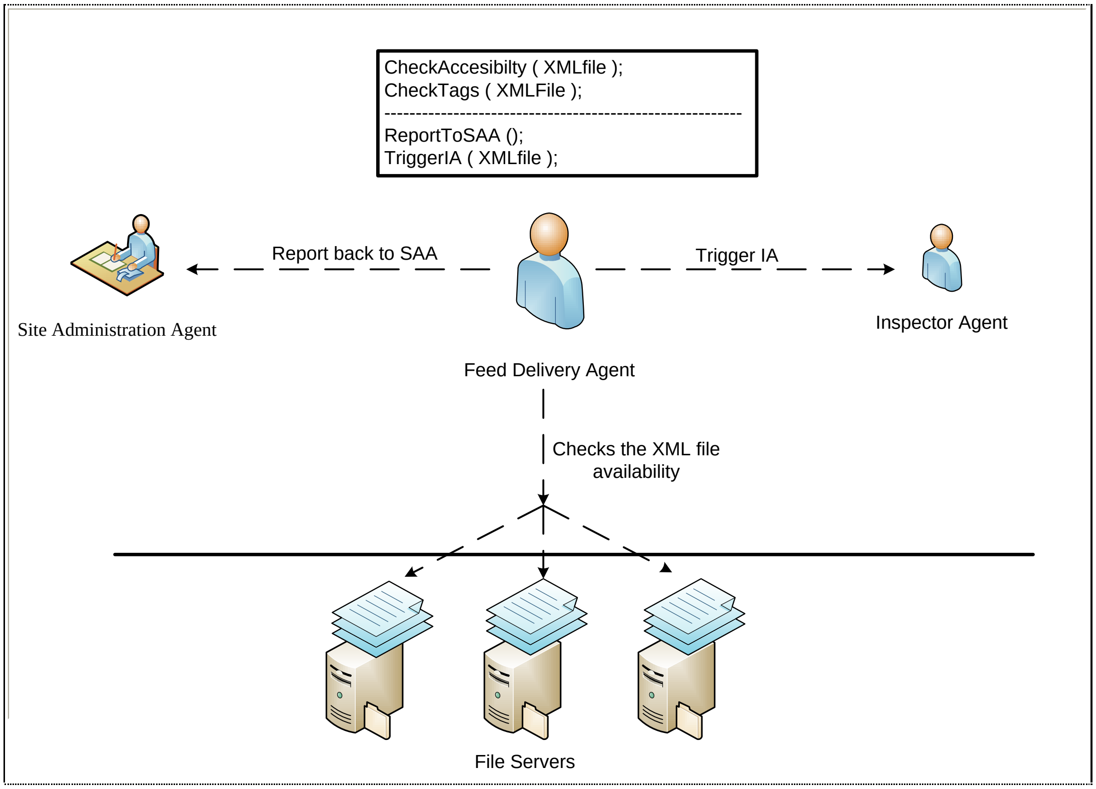


Figure 5.4 Feed Delivery Agent in Process

5.1.3 Inspector Agent

Inspector Agent **(**IA) chooses a certain quantity of keywords from each product description from the incoming feeds and, by grouping the products according to the IDs at the system, it delivers to the Hashing Agent. It is important that the IA work properly while the product category is identified. At the time of identifying the chosen keywords, what criteria are to be used and how many pieces are to be chosen from these keywords [39] are the most important subject factors in corroborating the categorical operation.

In the process of keyword determination, a random working strategy is being chosen. In the starting step of the operation (Figure 5.5), the whole words that exist in the product information are placed into an ArrayList as they make up separate records. After these records take form, a filtering operation is performed to purge the words that will not be keywords. During the filtration, the records of the words that frequently exist in most sentences such as “the, any, of, that, etc.” are erased from the ArrayList.

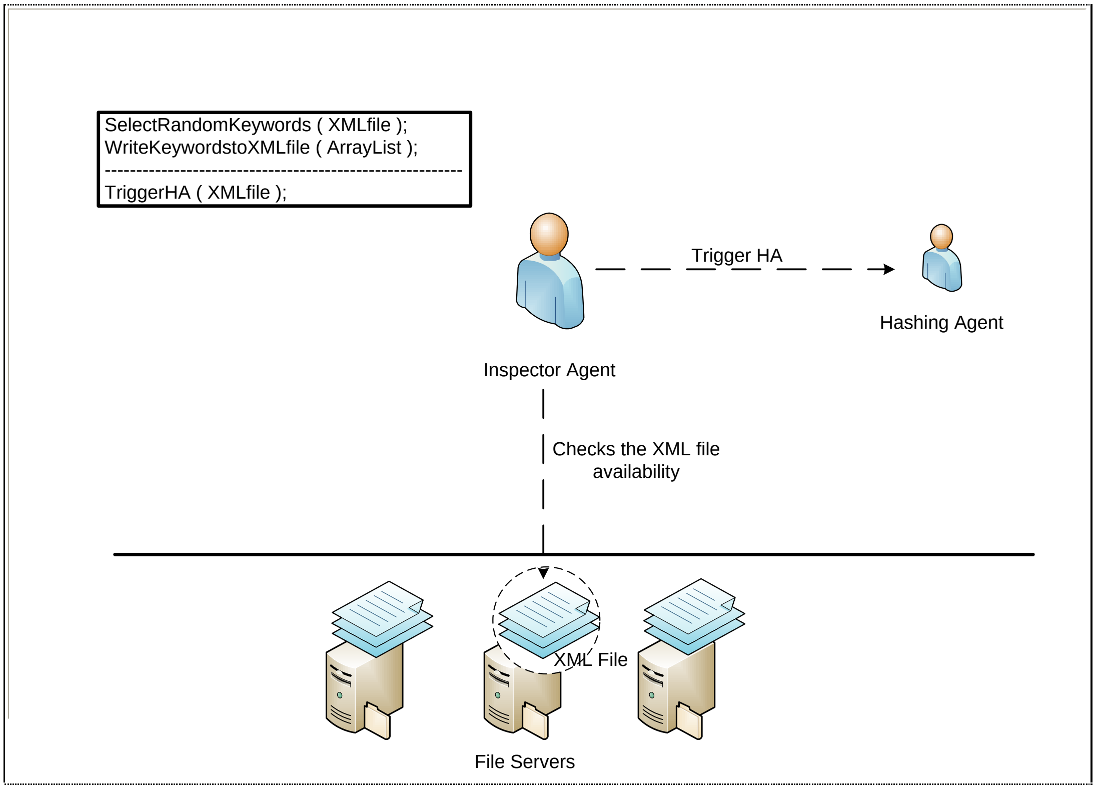

SelectRandomKeywords (XMLfile);
WriteKeywordstoXMLfile (ArrayList);
TriggerHA (XMLfile);
Trigger HA
Hashing Agent
Inspector Agent
Checks the XML file availability
XML File
File Servers

Figure 5.5 Inspector Agent in Process

With the records that remain, we start a selection process by giving priority to numerical statements. During the selection in order to increase the correctness ratio on deciding the product to which category to match, we must be careful in choosing the records at least the half of them to be chosen from non-numerical records. Because, if we choose the whole of keywords in records that have numerical value for a search to define 30 keywords, it will be nearly impossible for the product category to be decided. If half of the keywords numbers are not chosen from the numeric fields - in other words, if not enough numerical fields exist, then non-numerical fields will be preferred for selections.

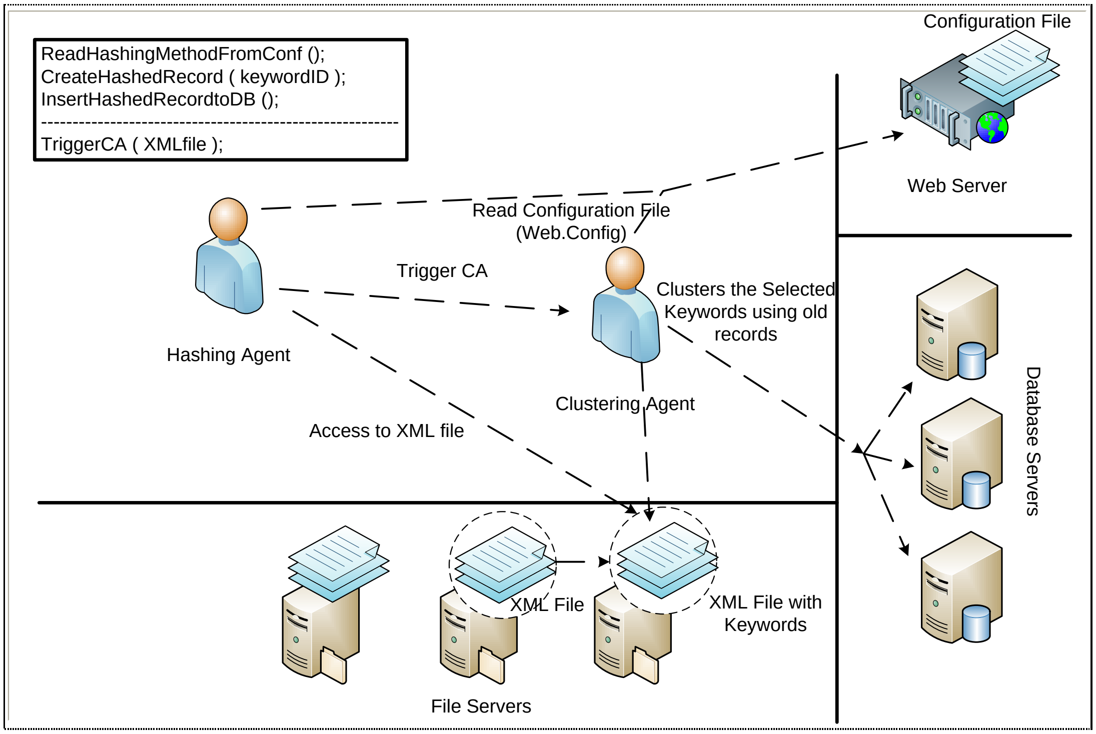
In selecting the non-numerical keywords, priority will be given to words that come after the measurement units. For this process, the index numbers such as "*cm, pixel, gallon, inch, Hz, rpm, etc.*" of the records' measurement units will be put into another ArrayList. The one-third (1/3) of the keywords chosen with a turning rotation (for example, 10 records for a search of 30 keywords) will be filled with words that come after these measurement units. The measurement for this process is to record the index number in ArrayList by increasing the index value by 1. As a result of this process we reach the words that have importance in deciding the category such as “55 gallon aquarium tank”. In determining the other keywords, the non-numerical records that remain are used, and without taking consideration of any priority, the selection is performed at random. Pseudo code of the keyword selection process follows as;

```
Open XML file
Read string from Description Tag
Repeat
        Copy each word into Description Arrray
Until all words passed
Close XML file
Set wNumber integer by counting words
Set numKeyword by wNumber / 2
For each word in Description Array
        for int i=0 to numKeyword
        if  isnumeric( DescriptionArray [ random( ) ] is true
                write into ArrKeyword
        end
        while i = wNumber
end
Create XML file
Copy keywords to XML file
```

The keywords identified at the end of the whole identification process are transferred into a XML file and are ready for the hashing process.

5.1.4 Hashing Agent

As shown in Figure 5.6, Hashing Agent (HA) takes the keywords determined by IA feeds from the hashing process and registers them in the system's database without the attending category. HA converts the keyword records into hashed records using *MD5, SHA1, SHA256, SHA384* and *SHA512* hashing algorithms [40] in order to match selected performance issues in the XML file. After the hashing process, the XML file now contains the hashed records to be clustered.



```
ReadHashingMethodFromConf ();
CreateHashedRecord ( keywordID );
InsertHashedRecordtoDB ();
-------------------------------------------------------
TriggerCA ( XMLfile );
```

Figure 5.6 Hashing Agent and Clustering Agent in Process

HA puts the keywords that are from the product feeds' product explanations into hash functions in order to use them in clustering process and saves them into mall database. Hashing Agent reads the hash function that it will use from the configuration file of the mall system and applies it. The mall administrators must determine the hashing algorithm that will be used according to the load and overpopulation of the system and must define it as a key to the systems configuration file. Otherwise, HA will use MD5 algorithm as default selection.

5.1.5 Clustering Agent

Clustering Agent (CA) controls the products not belonging to a category that are newly registered to the system and determines in which category to include them. By using the present reports at the system, it makes one point for each product at the multi dimensional space function. It determines that the places where these points come together are different categories. If the points are too near to each other and come together under a specific threshold, these points can be accepted as the same product and are accepted as wrong product descriptions in the member stores. Determination of the boundaries of the categories that are near each other is another important criterion. During the clustering process *Partition Co-efficiency*, *Classification Entropy*, *Partition Index* and *Separation Index* values must be evaluated correctly.

CA follows up a working method with two levels during the clustering processes using *K-Mean* [41] and *Fuzzy C-Means* [42] algorithms. First, it clusters the product using the keywords in the product information with 100 products that are randomly chosen from each category without taking into consideration the threshold value. It finds out to which category it belongs from among 30 categories extracted from eBay. After the product's category is found, second clustering process is then applied with all database records that belong to that category while keeping the threshold value at a very low level because of its sensitivity.

After the clustering process, it is being established to which sub-category (for example, there are sub-categories like Brass, Guitar, Electronic, Equipment, etc. under the main category of Musical Instruments) it belongs. When there are products at the limits against the threshold value, which is kept high during clustering process, they are added to the 'Everything Else' category.

5.2 Configuring Web Servers for Network Load Balancing

AGBEP's web applications typically perform such mission-critical tasks such as displaying products, organizing categories, financial transactions, etc. Businesses can lose millions of dollars when mission-critical applications aren't available, so it's important that these applications are available at all times. In a nutshell, these applications must be highly available and reliable.

An easy and effective way to achieve availability and reliability is to use redundant servers. If you have more than one server implementing your site, and then one of the server's crashes, the processing requests can be redirected to another server. This provides a highly available Web site.

To activate the Network Load Balancing (NLB) [43] feature we have to use the Network Load Balancing Manager (Figure 5.7) which is an administrative console of

Server 2003. NLB facilitates the process of creating a Web Server Farm. A Web Server farm is a redundant cluster of several Web servers serving a single (virtual) IP address.

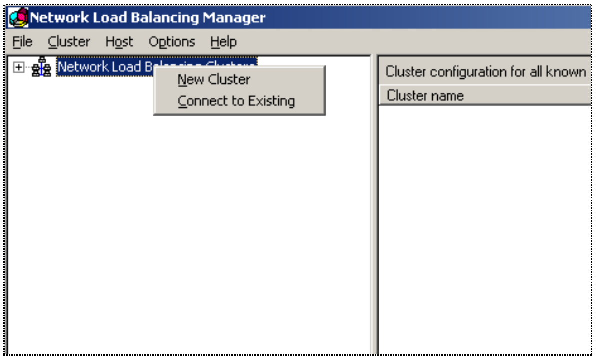
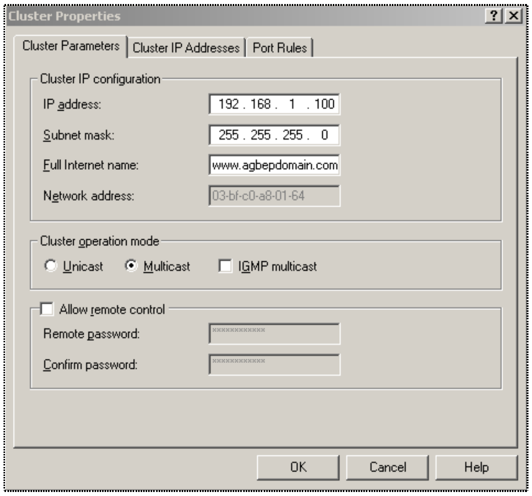

Network Load Balancing Manager
File
Cluster
Host
Options
Help
Network Load Balancing Clusters
New Cluster
Connect to Existing
Cluster configuration for all known
Cluster name

Figure 5.7 Network Load Balancing Manager

In order to create AGBEP's web server clusters, we must right click on the root node to add a new cluster whic is shown in Figure 5.8. Next we will configure the basic cluster configuration, which will consist of assigning the Cluster or virtual IP address.



Cluster IP configuration

IP address: 192 . 168 . 1 . 100

Subnet mask: 255 . 255 . 255 . 0

Full Internet name: www.agbepdomain.com

Network address: 03-bf-c0-a8-01-64

Cluster operation mode

○ Unicast ● Multicast ☐ IGMP multicast

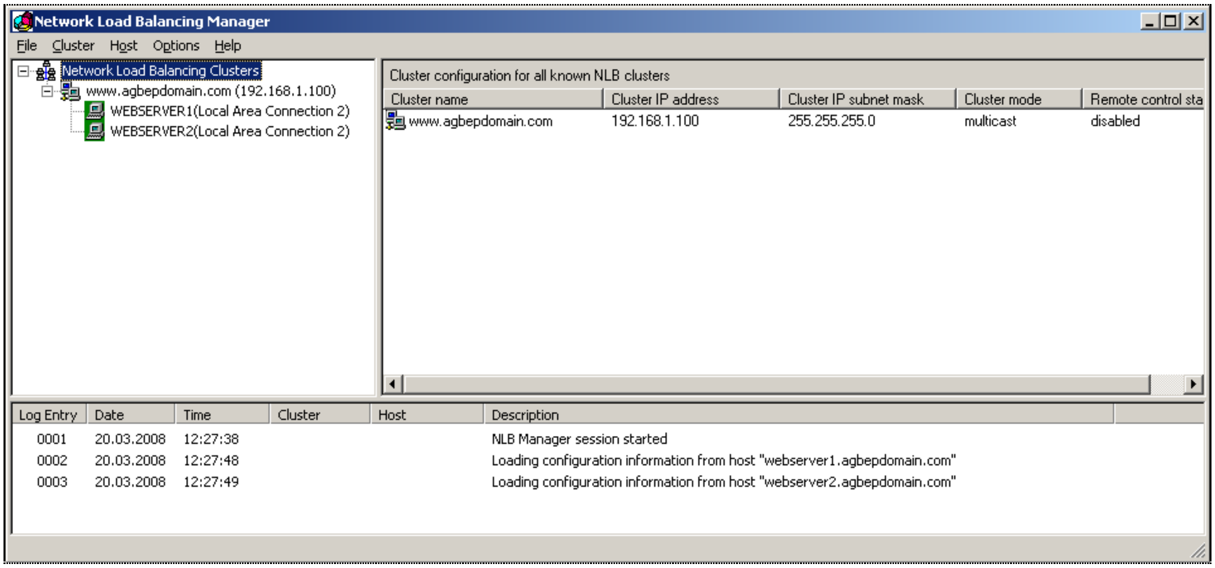
☐ Allow remote control

Remote password: ************

Confirm password: ************

Figure 5.8 Properties of a Cluster

After defining the two clusters to the system, the web farm now have two web servers running under a virtual IP to balance the client's request as illustrated in Figure 5.9.


Network Load Balancing Manager
File Cluster Host Options Help

- Network Load Balancing Clusters
  - www.agbepdomain.com (192.168.1.100)
    - WEBSERVER1(Local Area Connection 2)
    - WEBSERVER2(Local Area Connection 2)

Cluster configuration for all known NLB clusters

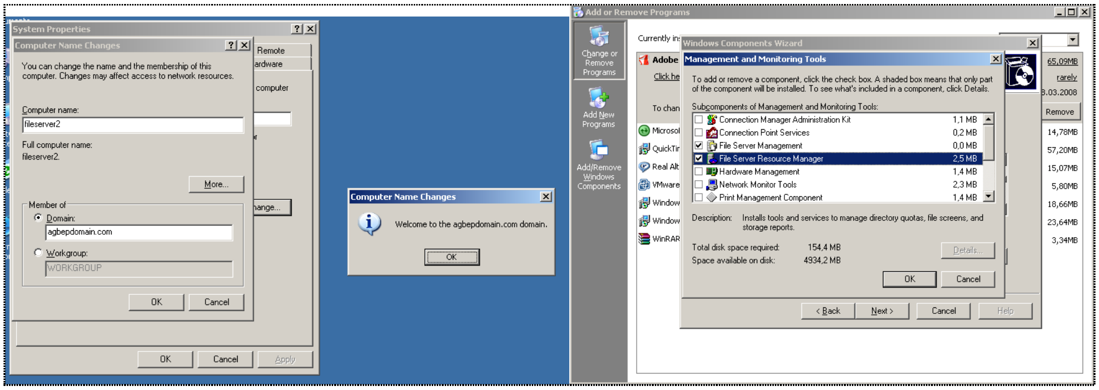
Cluster name	Cluster IP address	Cluster IP subnet mask	Cluster mode	Remote control sta
www.agbepdomain.com	192.168.1.100	255.255.255.0	multicast	disabled

```
Log Entry  Date        Time      Cluster  Host  Description
0001       20.03.2008  12:27:38                 NLB Manager session started
0002       20.03.2008  12:27:48                 Loading configuration information from host "webserver1.agbepdomain.com"
0003       20.03.2008  12:27:49                 Loading configuration information from host "webserver2.agbepdomain.com"
```

Figure 5.9 Network Load Balancing Manager with defined Clusters

5.3 File Servers Installation and Management

It is very important that all file servers have to be a member of AGBEP domain, if we want to control and limit the usage of the member shops as domain users. After including the servers into the agbepdomain.com we have to install the File Server Management and File Server Resource Manager services as shown in Figure 5.10.



Computer Name Changes

You can change the name and the membership of this computer. Changes may affect access to network resources.

Computer name: fileserver2

Full computer name: fileserver2.

Member of

● Domain: agbepdomain.com

○ Workgroup: WORKGROUP

OK Cancel

Computer Name Changes

Welcome to the agbepdomain.com domain.

OK

Windows Components Wizard

Management and Monitoring Tools

To add or remove a component, click the check box. A shaded box means that only part of the component will be installed. To see what's included in a component, click Details.

Subcomponents of Management and Monitoring Tools:

□	Connection Manager Administration Kit	1,1 MB
□	Connection Point Services	0,2 MB
☑	File Server Management	0,0 MB
☑	File Server Resource Manager	2,5 MB
□	Hardware Management	1,4 MB
□	Network Monitor Tools	2,3 MB
□	Print Management Component	1,4 MB

Description: Installs tools and services to manage directory quotas, file screens, and storage reports.

Total disk space required: 154,4 MB

Space available on disk: 4934,2 MB

OK Cancel

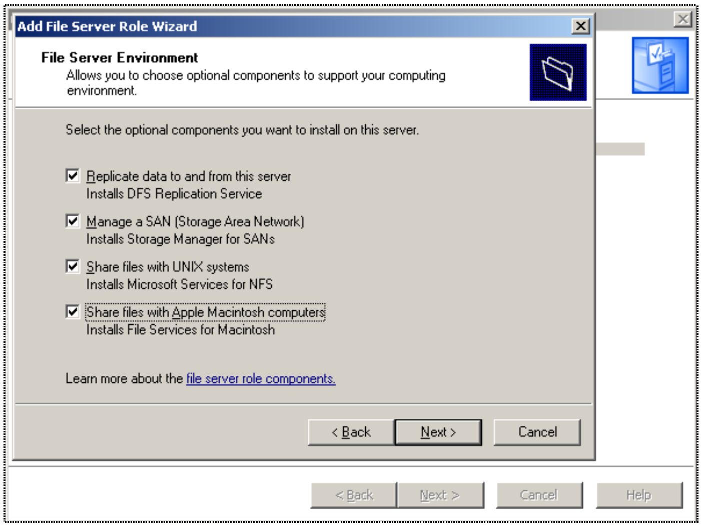
Figure 5.10 File Server Management and File Server Resource Manager Services

As displayed in Figure 5.11, we may choose to add some optional components such as;

Distributed File System (DFS) [43], is a set of client and server services that allow a large enterprise to organize many distributed file shares into a distributed file system.

In computing, a Storage Area Network (SAN) [43] is an architecture to attach remote computer storage devices (such as disk arrays, tape libraries and optical jukeboxes) to servers in such a way that, to the operating system, the devices appear as locally attached.

Network File System (NFS) [43] is a network file system protocol originally jointly developed by Sun Microsystems and IBM in 1984, allowing a user on a client computer to access files over a network as easily as if the network devices were attached to its local disks.



File Server Environment

Allows you to choose optional components to support your computing environment.

Select the optional components you want to install on this server.

☑ Replicate data to and from this server
Installs DFS Replication Service

☑ Manage a SAN (Storage Area Network)
Installs Storage Manager for SANs

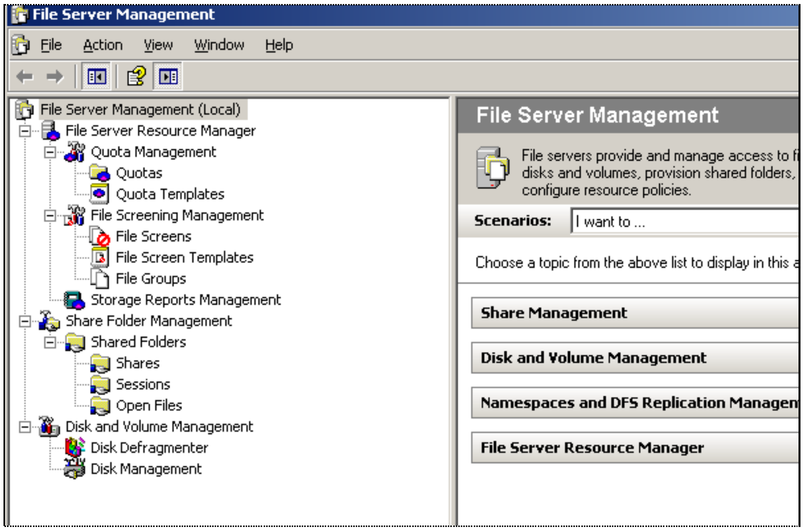
☑ Share files with UNIX systems
Installs Microsoft Services for NFS

☑ Share files with Apple Macintosh computers
Installs File Services for Macintosh

Learn more about the file server role components.

Figure 5.11 Optional Components

After setting up the system we can command the file servers using the File Server Management Console as illustrated in Figure 5.12. By using this console, we can create quotas for files which will be uploaded by member shops. Also system administrators can monitor the shared folders and can create scheduled reports of these.



File servers provide and manage access to fi disks and volumes, provision shared folders, configure resource policies.

Choose a topic from the above list to display in this a

Figure 5.12 File Server Management Console

5.4 SQL Server Database Mirroring Installation

At the problem of maintaining high availability, the web server’s numbers being enough doesn’t mean that AGBEP will work with a high performance. At time of duty of the web servers, the back front database servers operate in order to provide the necessary information. In order to function a large number of web servers, there need to function a large number of database servers as similar. The management of a large number of database servers is another problem that must be solved. Since AGBEP uses the SQL Server 2005 [43] as a database solution, the working methodology of the database servers is managed by Witness Server. In the following paragraphs, detailed information is mentioned about this structure.

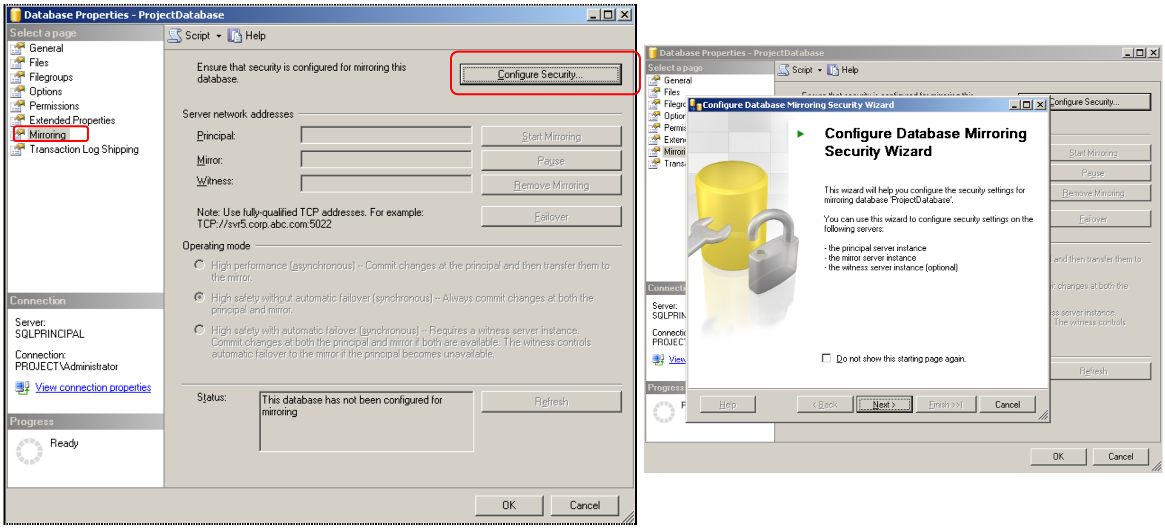
Physically there are two mirror database servers which are coordinated by scout server. The primary goal of database mirroring is to increase data availability and allow failover in case a server hosting the database becomes unavailable such as the result of hardware or network failure [44].

Maintaining synchronized copies of a database on two separate servers allows switching between them on an as needed basis, reversing roles of partners participating in the mirroring session, (the former principal becomes the mirror and vice versa). With database mirroring, this process can be automated, which requires the presence of another instance of SQL Server 2005, although this functionality is not available in the SQL Server 2005 Express edition, running on a separate server, referred to as witness. This server monitors operations of mirroring partners, triggering failover in case of a lack of heartbeat response from the principal and ensuring that at any given time there is only one principal within each mirroring session. The decision is coordinated between the witness and at least one other operational server. This protects against the "split brain" scenario, which could occur if direct connectivity between the principal and the mirror is lost. As long as the synchronization between the two is maintained, the mirror can take over processing client requests without noticeable delay (no more than a few seconds) and without any data loss (once the principal comes back on line, it automatically assumes the role of the mirror and catches up with changes on the new principal). Automatic failover is further supplemented by the new MDAC client software which is based on .NET provider, smart enough to redirect client applications transparently to the operational server.

When configuring database mirroring, one option is to use the High Availability mode. This option allows for synchronizing of transaction writes on both servers, as well as offers the ability of automated failover. When using the High Availability mode, you need to have three instances of SQL Server: the principal, mirror and the witness [44]. Here is a summary of what each component does.

- ✓ Principal – this is the instance that stores the active database.
- ✓ Mirror – this is the instance that receives transactions to keep the mirrored database in sync.
- ✓ Witness – this is the instance that communicates with the principal and mirror to determine if failover should occur.

To install mirroring of SQL server first we have to start the service in principal server (Figure 5.13).



Ensure that security is configured for mirroring this database.

Server network addresses

Principal:

Mirror:

Witness:

Note: Use fully-qualified TCP addresses. For example: TCP://svr5.corp.abc.com:5022

Operating mode

○ High performance (asynchronous) -- Commit changes at the principal and then transfer them to the mirror.

● High safety without automatic failover (synchronous) -- Always commit changes at both the principal and mirror.

○ High safety with automatic failover (synchronous) -- Requires a witness server instance. Commit changes at both the principal and mirror if both are available. The witness controls automatic failover to the mirror if the principal becomes unavailable.

Status: This database has not been configured for mirroring

Server: SQLPRINCIPAL

Connection: PROJECT\Administrator

Configure Database Mirroring Security Wizard

This wizard will help you configure the security settings for mirroring database 'ProjectDatabase'.

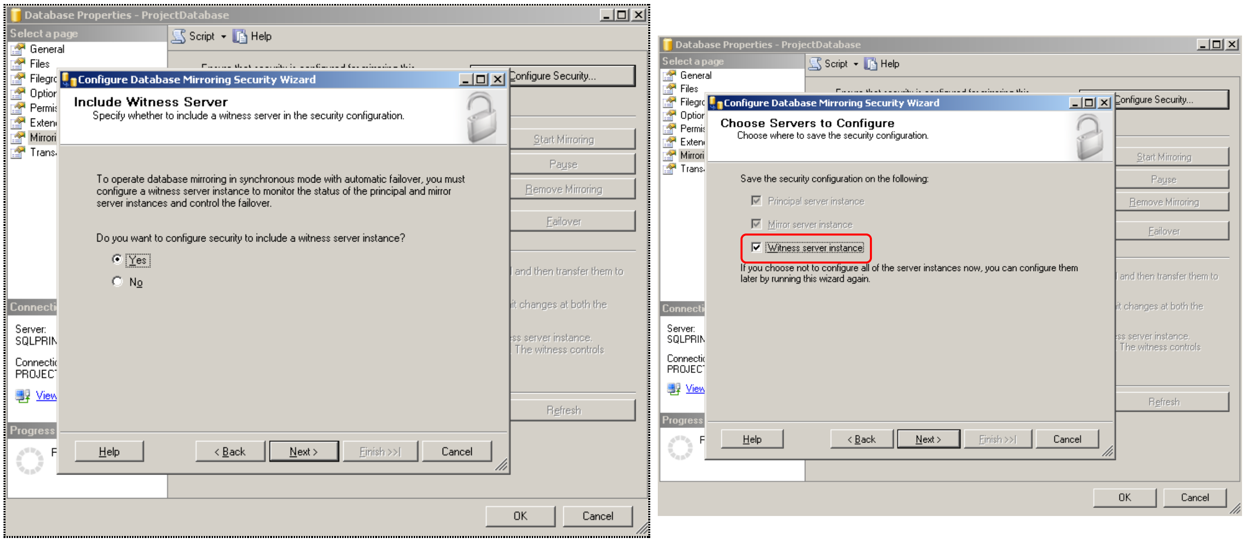
You can use this wizard to configure security settings on the following servers:

- the principal server instance
- the mirror server instance
- the witness server instance (optional)

□ Do not show this starting page again.

Figure 5.13 Installation of Principle Database Server

If we want a scout (witness server) to observe work efficiency on principal and mirror server, we have to enable its feature as shown in the Figure 5.14.



Include Witness Server

Specify whether to include a witness server in the security configuration.

To operate database mirroring in synchronous mode with automatic failover, you must configure a witness server instance to monitor the status of the principal and mirror server instances and control the failover.

Do you want to configure security to include a witness server instance?

● Yes

○ No

Choose Servers to Configure

Choose where to save the security configuration.

Save the security configuration on the following:

☑ Principal server instance

☑ Mirror server instance

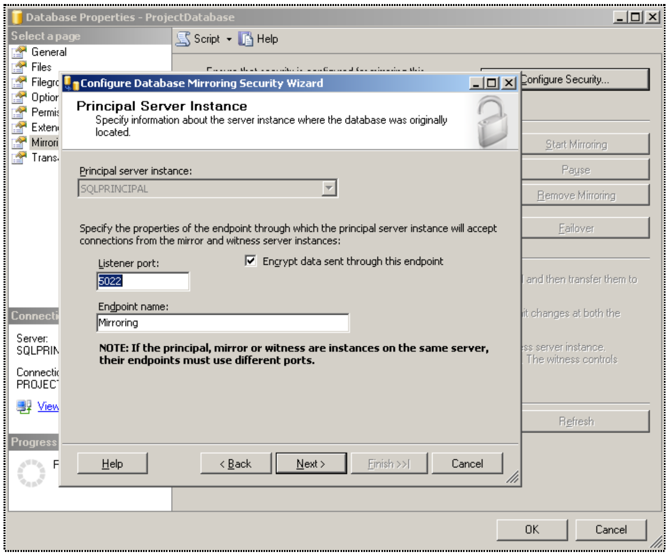
☑ Witness server instance

If you choose not to configure all of the server instances now, you can configure them later by running this wizard again.

Figure 5.14 Installation of Witness Server

The witness is a third instance of SQL Server 2005 that acts as an intermediary between the principal and the mirror in order to determine when to fail over. By having a third instance, it creates the ability to have a 2–1 vote that says one of my components is not available and therefore I am going to fail over. Because of the need to determine if the components are online or offline before an automatic failover, the witness server is only needed when you implement the High Availability mode and you want, or need, automatic failover. This instance doesn't do much more than communicate with the principal and the mirror to make sure they are still alive. No database activity is occurring on this instance, just communication between the three components.

Principal server and the mirror server communicate on a defined port in the configuration as illustrated in Figure 5.15.



Database Properties - ProjectDatabase

Configure Database Mirroring Security Wizard

Principal Server Instance

Specify information about the server instance where the database was originally located.

Principal server instance: SQLPRINCIPAL

Specify the properties of the endpoint through which the principal server instance will accept connections from the mirror and witness server instances:

Listener port: 5022

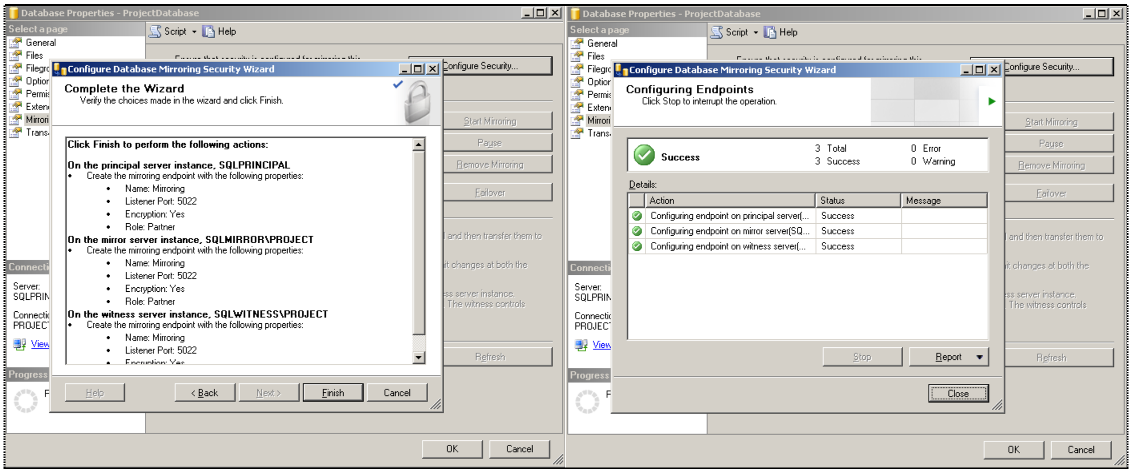
☑ Encrypt data sent through this endpoint

Endpoint name: Mirroring

NOTE: If the principal, mirror or witness are instances on the same server, their endpoints must use different ports.

Figure 5.15 Setting the Communication Port

After setting the “Listener Port” and the “Endpoint Name” parameters for all three SQL Servers, configuration will be finished (Figure 5.16).



Configure Database Mirroring Security Wizard

Complete the Wizard

Verify the choices made in the wizard and click Finish.

Click Finish to perform the following actions:

On the principal server instance, SQLPRINCIPAL
- Create the mirroring endpoint with the following properties:
  - Name: Mirroring
  - Listener Port: 5022
  - Encryption: Yes
  - Role: Partner

On the mirror server instance, SQLMIRROR\PROJECT
- Create the mirroring endpoint with the following properties:
  - Name: Mirroring
  - Listener Port: 5022
  - Encryption: Yes
  - Role: Partner

On the witness server instance, SQLWITNESS\PROJECT
- Create the mirroring endpoint with the following properties:
  - Name: Mirroring
  - Listener Port: 5022
  - Encryption: Yes

Configure Database Mirroring Security Wizard

Configuring Endpoints

Click Stop to interrupt the operation.

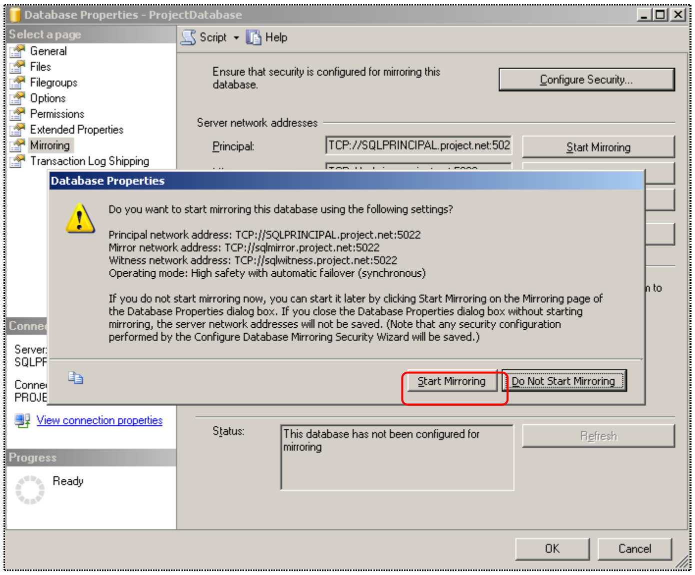
Success 3 Total 3 Success 0 Error 0 Warning

Details:

Action	Status	Message
Configuring endpoint on principal server(...	Success	
Configuring endpoint on mirror server(SQ...	Success	
Configuring endpoint on witness server(...	Success	

Figure 5.16 Confirmation of the parameters

At the end we have to install the certificates for authentication of servers before starting the mirroring service as shown on Figure 5.17.



Ensure that security is configured for mirroring this database.

Server network addresses

Principal: TCP://SQLPRINCIPAL.project.net:502

Database Properties

Do you want to start mirroring this database using the following settings?

Principal network address: TCP://SQLPRINCIPAL.project.net:5022
Mirror network address: TCP://sqlmirror.project.net:5022
Witness network address: TCP://sqlwitness.project.net:5022
Operating mode: High safety with automatic failover (synchronous)

If you do not start mirroring now, you can start it later by clicking Start Mirroring on the Mirroring page of the Database Properties dialog box. If you close the Database Properties dialog box without starting mirroring, the server network addresses will not be saved. (Note that any security configuration performed by the Configure Database Mirroring Security Wizard will be saved.)

Start Mirroring | Do Not Start Mirroring

Status: This database has not been configured for mirroring

Figure 5.17 Starting up the Mirroring Service

Principal SQL Server’s certificate can be demonstrated as;

```
create master key encryption by password = 'AGBEPp@ssw0rd';

create certificate Principal_Server_Cert with subject = Principal_Server certificate', start_date = '2008/11/01', expiry_date = '2030/11/01';

Create endpoint endpoint_mirroring state = started
as tcp(listener_port = 7024, listener_ip = all)
for database_mirroring (authentication = certificate Principal_Server_Cert, encryption = disabled, role = all);

Backup certificate Principal_Server_Cert to file = 'c:\ Principal_Server_Cert.cer';
GO
```

5.5 Composition of the Database

AGBEP has to contain a database with a suitable architecture in order to store the critical information such as products, users, categories, member shops, sales percentages and stock conditions. This database is used by giving accessed to logins that have access rights to different tables. The table indexes are updated with database maintenance duties which are scheduled in certain periods and to shorten the query durations of the stored procedures that are used in the system is aimed.

AGBEP’s database has three different login users and four different schemes. Using login users in database provides data security and consistency. First user is *admin* user. This user is the most powerful user in database. This user handle all transactions in all tables such as update, delete, and create. Briefly, admin has all permissions in database. The second one is *salesman* user. Salesman user has permissions only in customer tables and order tables. This login user only has edit operation. Last login user name is *customer*; this user can access customer tables and order tables.

Four different schemes [44] are created to divide the database in parts and increase security. These schemes are;

- Admin Scheme
- Customer Scheme
- Product Scheme
- eMallSetup Scheme

5.5.1 Admin Scheme

In admin scheme; there are tables, which are related to administrative operations such as adding new website role, adding new user, personalize users, log system events etc. in this scheme the most important tables are aspnet_roles and aspnet_users, because this tables contains all website users and users’ accessing roles such as admin role, user role etc. Figure 5.18 illustrates us the tables including Admin scheme.

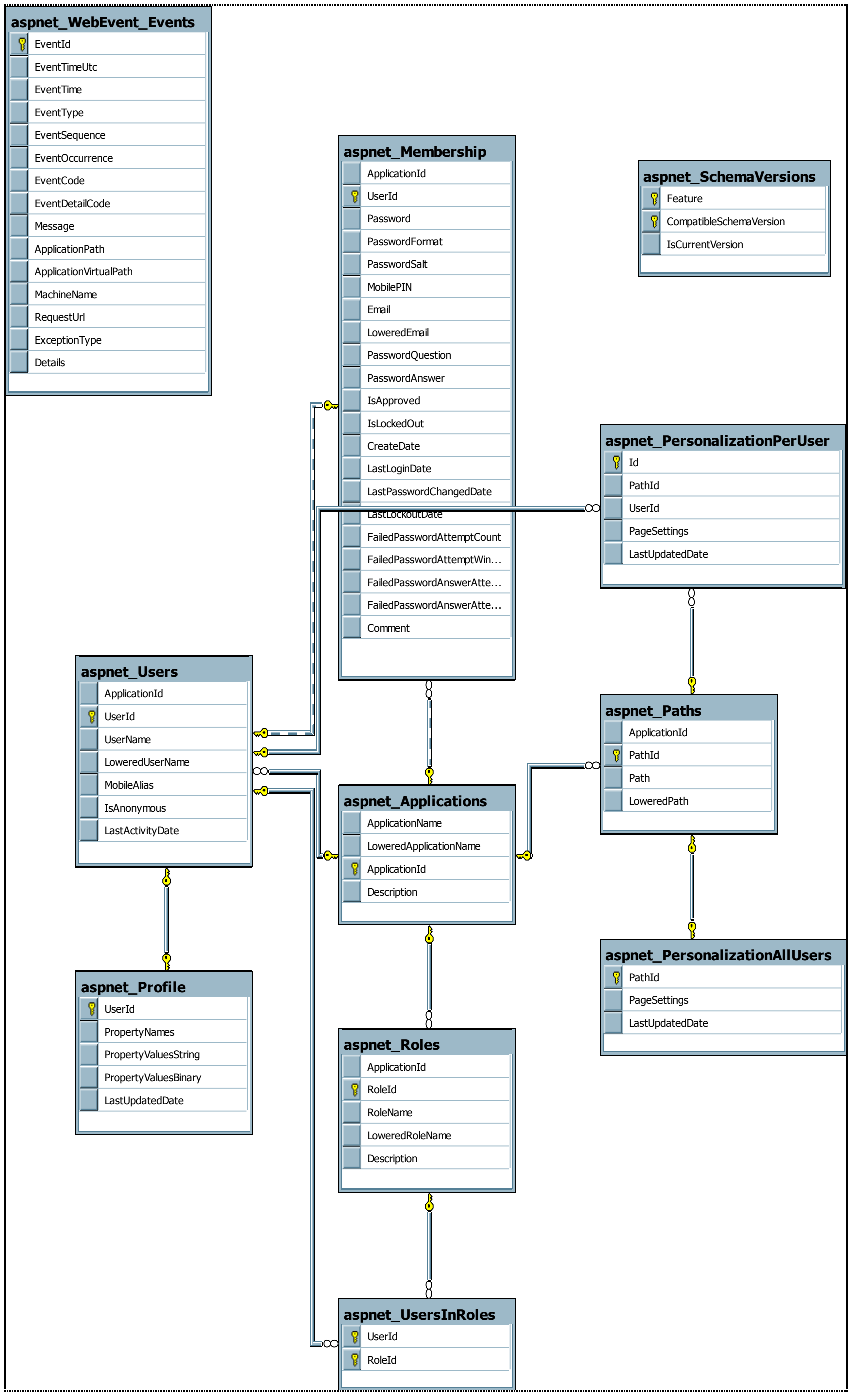


Figure 5.18 Tables of the Admin Scheme

5.5.2 Customer Scheme

Customer scheme composes of general customer information pool, currency part, order parts such as general orderings which made by the customers, order tracking information for each customer orders, ordered items by customers and other order system information, Customer Payment Logs for all of the customer paid and lastly coupon information as the shopping alternatives. Customer scheme constitutes the most important and risky parts of the agent management system. Because the most important and noteworthy records that are related to the customer, orders and payments are kept in this scheme as shown in Figure 5.19.

Like this; customer, paymentlog, orders, ordertracking, orderitem are the most considerable tables in that scheme. Customer table keeps the general customer personal and shipping information for the shopping on this site. The personal information is required for the verifying customer and shipping information is required for the submission of the products. Order table keeps the customer orderings which will be bought by site members and will be sending to the customers. Paymentlog table saves the sold products which were paid and sent to the customers. The other one is orderitem. Orderitem saves the many items which will be purchased and are being in the order list. The last worthwhile table is ordertracking table. Ordertracking table is used for finding easily the orders which were made by customers.And the one of the rest of the table is Coupon table. Coupon can be given to the customers for the shopping. Each coupon has an expiration quantity, date, type and also percentage. Currency table also is included for keeping the currency types and rates for the payment in the shopping.

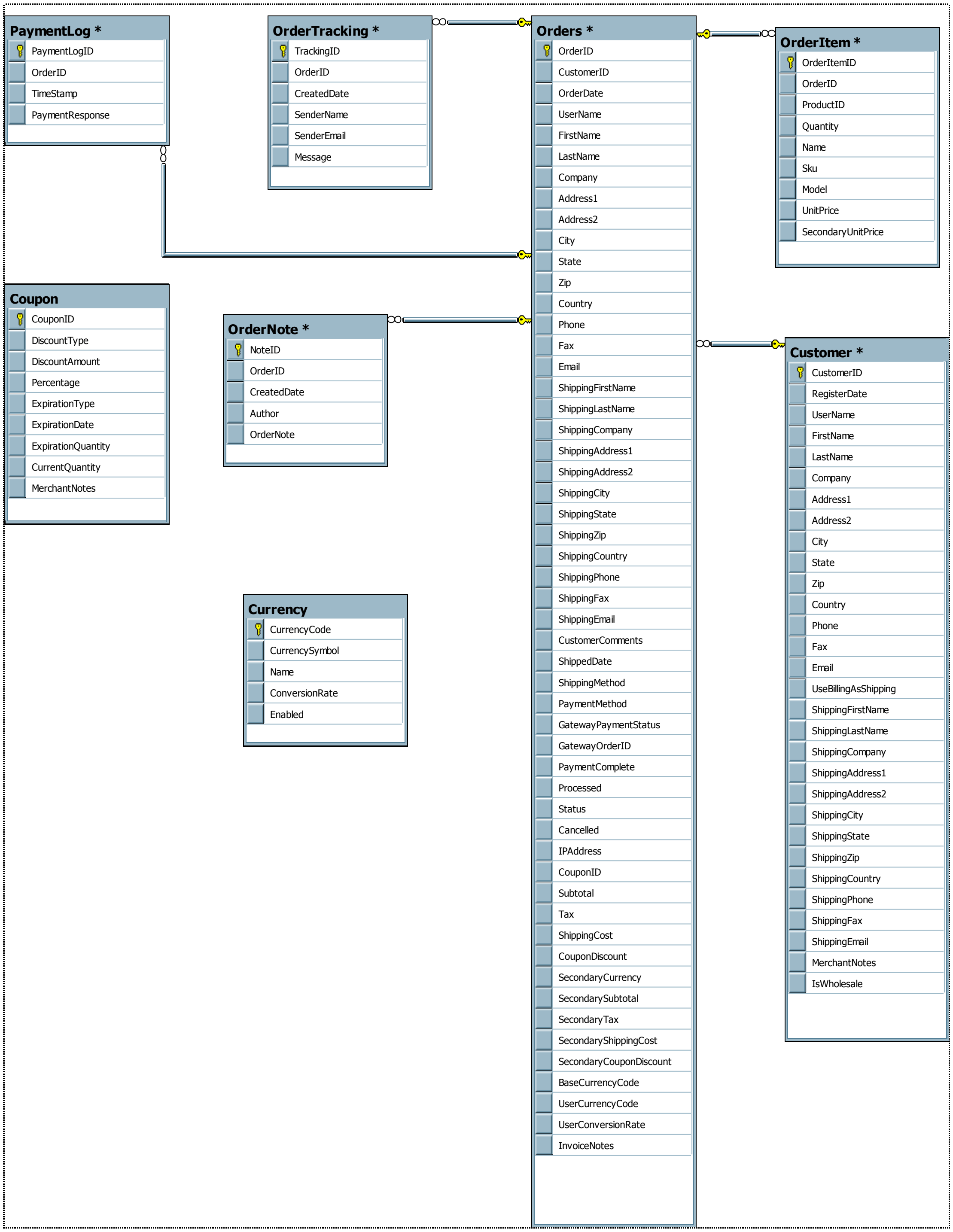


Figure 5.19 Tables of the Customer Scheme

5.5.3 Product Scheme

Product scheme composes of a few of system tables that are included Product tables and Category tables. Although, product scheme items are related to general gathering of the product information, they are related to category information too. In product information, generally information about the goods which are entered to the system and information about the product that will be purchased by the members of the website are told. So Product scheme ensures the configuration of the products and category area for the admin and ensures shopping configuration system for the customers.

The most important tables are product, category, product locale and category locale tables. The rest of them ensure the linking for the system. Product table consists of product information for the new entered goods or about the not sold product into the system. Category table designates the product in which category. The next one is product locale table that provides product information for the Multilanguage system e.g. long description, short description ...etc. in website. In the same way, category locale table includes category information for the Multilanguage system such as category description too. And the lastly, ProductCategory table includes only a relationship between product and category as many to many relationship in the system. Whole tables of this scheme are illustrated in Figure 5.20.

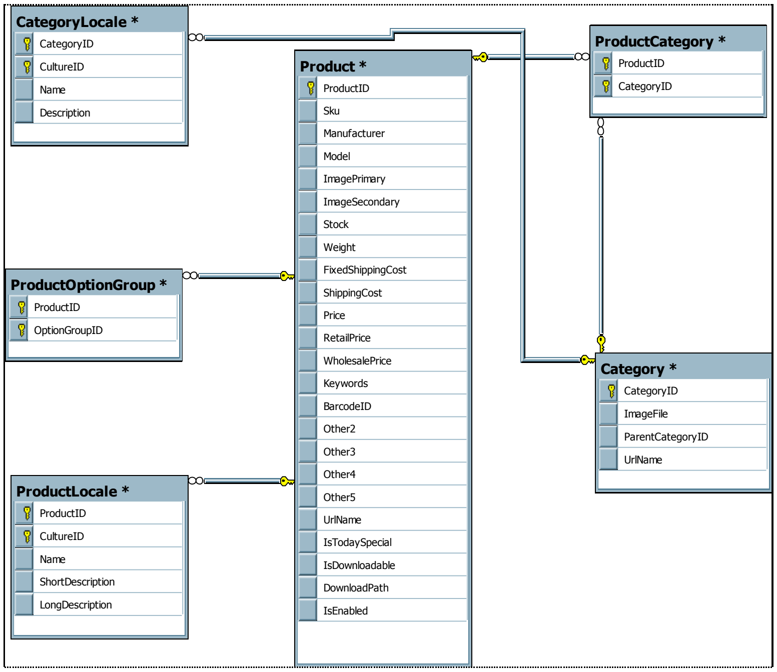


Figure 5.20 Tables of the Product Scheme

5.5.4 eMallSetup Scheme

As shown in Figure 5.21, the last scheme is called eMallSetup. In this scheme there are tables which contain website configuration data. Also this scheme contains, culture table this table stored different languages which are supported by website. There is also page table, this table contains website page URLs and names, with this table admin can easily change website page URLs manually.

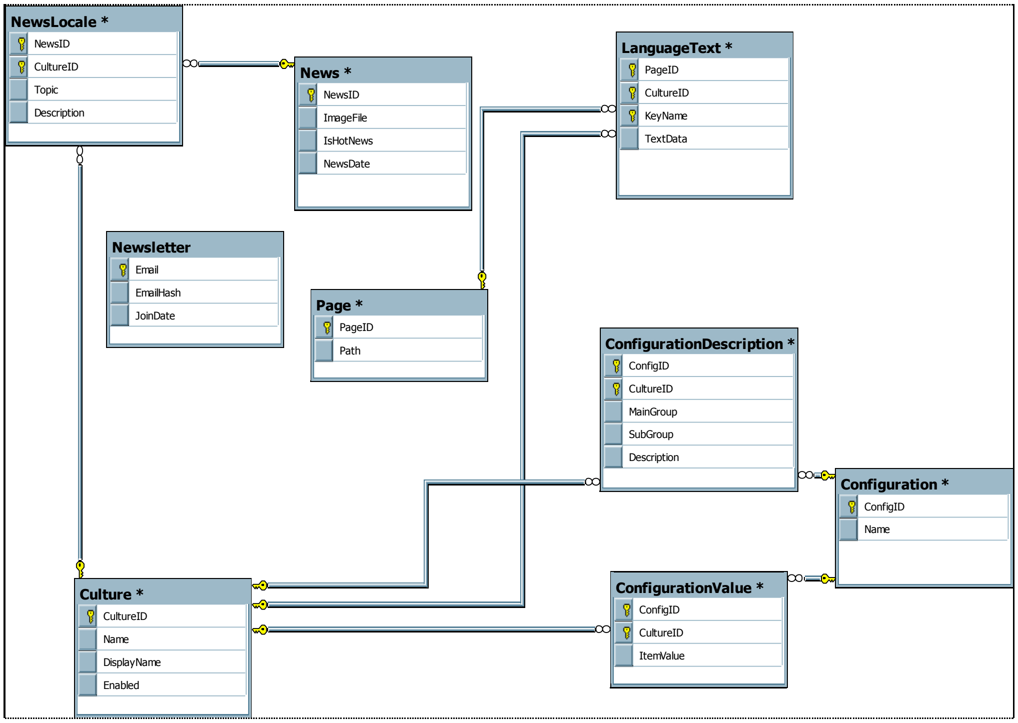
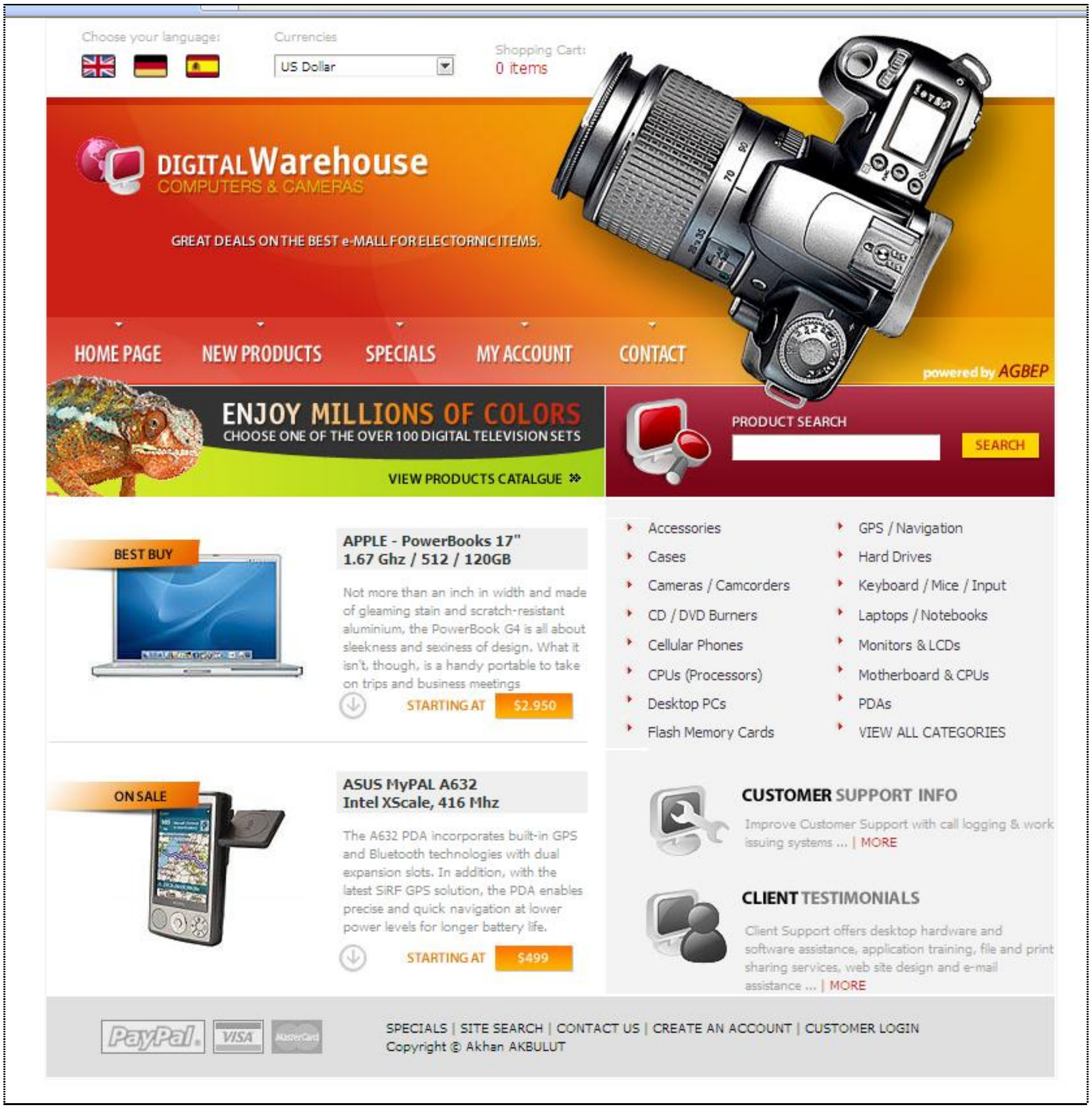


Figure 5.21 Tables of the eMallSetup Scheme

6 AN IMPLEMENTATION OF AGBEP: DIGITAL WAREHOUSE

The implementation of the AGBEP is scened as a Digital Warehouse which specialized on photography and its accessories. The web project where the AGBEP platform is integrated is designed as to give all services that an e-mall offers. Standard e-commerce processes such as showing of product lists, giving detailed product information and search and buying modules are accomplished.



APPLE - PowerBooks 17"
1.67 Ghz / 512 / 120GB

Not more than an inch in width and made of gleaming stain and scratch-resistant aluminium, the PowerBook G4 is all about sleekness and sexiness of design. What it isn't, though, is a handy portable to take on trips and business meetings

ASUS MyPAL A632
Intel XScale, 416 Mhz

The A632 PDA incorporates built-in GPS and Bluetooth technologies with dual expansion slots. In addition, with the latest SiRF GPS solution, the PDA enables precise and quick navigation at lower power levels for longer battery life.

CUSTOMER SUPPORT INFO

Improve Customer Support with call logging & work issuing systems ... | MORE

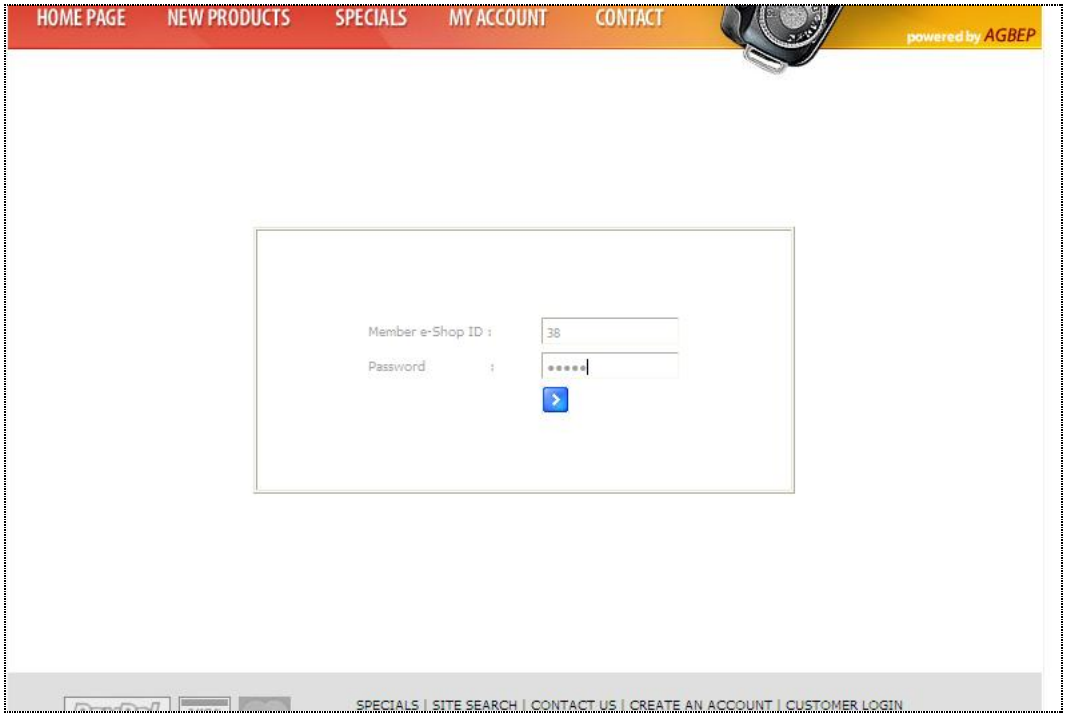
CLIENT TESTIMONIALS

Client Support offers desktop hardware and software assistance, application training, file and print sharing services, web site design and e-mail assistance ... | MORE

Copyright © Akhan AKBULUT

Figure 6.1 Digital Warehouse web site as The Implementation Subject

To access to AGBEP’s working parts, the member shop's manager gives his/her user name and password. We reach to login screen by choosing MY ACCOUNT on the main menu in Figure 6.1.


HOME PAGE
NEW PRODUCTS
SPECIALS
MY ACCOUNT
CONTACT
powered by AGBEP
SPECIALS | SITE SEARCH | CONTACT US | CREATE AN ACCOUNT | CUSTOMER LOGIN

Member e-Shop ID : 38
Password : •••••

Figure 6.2 Login Screen

If we logged on before or if there is an active session, than it is possible to reach feed uploading screen without facing the login screen in Figure 6.2. We access to the system by giving the member shops manager's user name and password -which are defined to the system- on login screen.

After the successful entrance to the system, the member shop copies the xml feed which contains the products' information that they want to add to the system, to the file server by using the interface in Figure 6.3 and starts to agents' working chain.

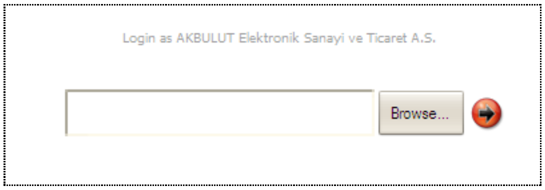
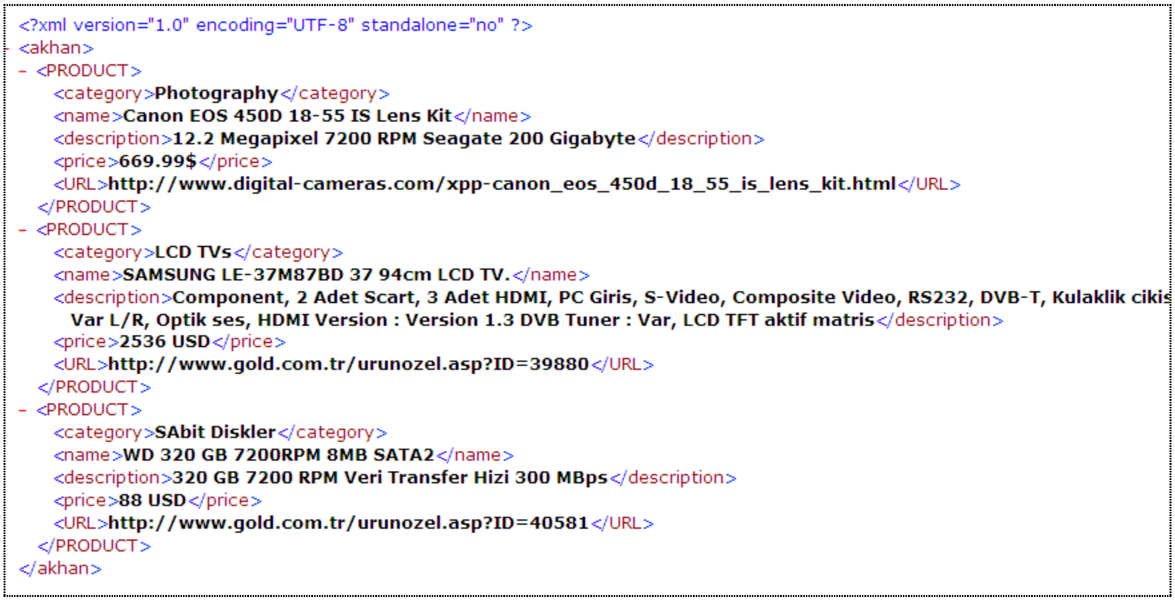

Login as AKBULUT Elektronik Sanayi ve Ticaret A.S.

Figure 6.3 XML Feed Upload Screen

An example of an xml feed is shown in Figure 6.4. The tag which is at the top level of feed contains the member shop's name information. Each product under the member shop differs from each other with -Product- tag. The Category, Name, Description, Price and URL information are stored by the same named tags of the products' which we want to add to the system under product tags.

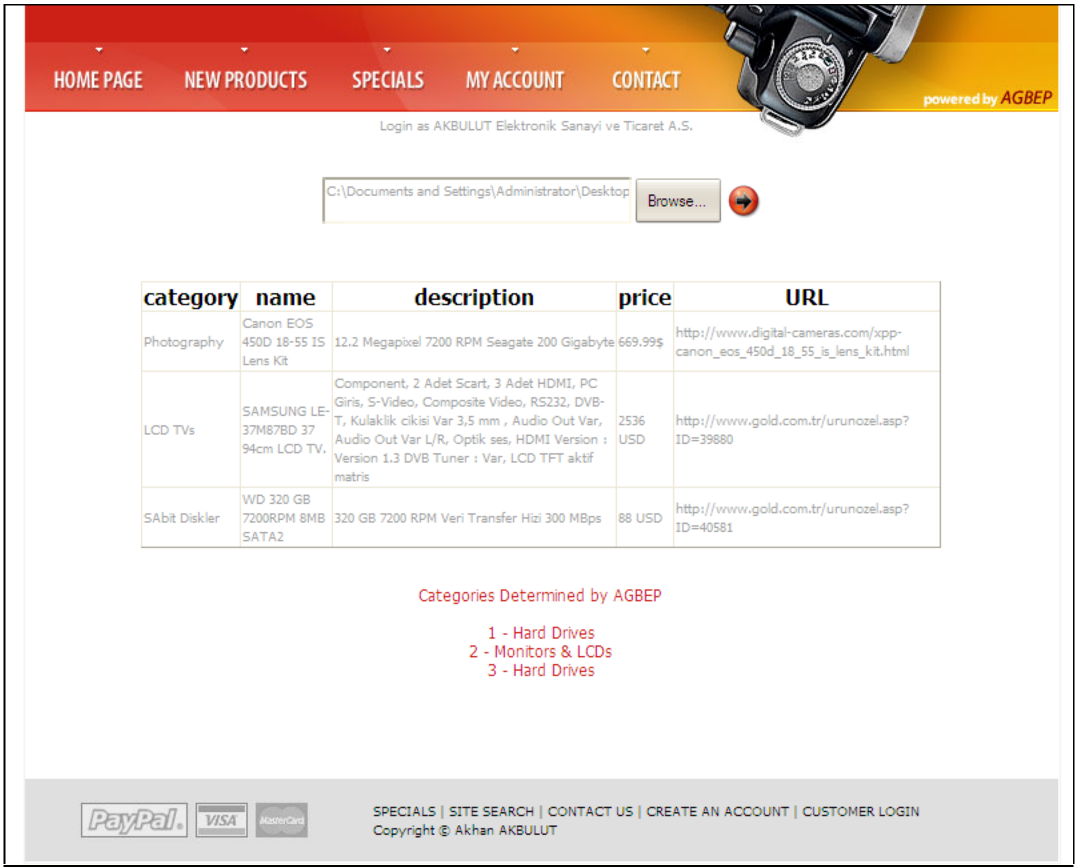


```
 <?xml version="1.0" encoding="UTF-8" standalone="no" ?>
- <akhan>
- <PRODUCT>
    <category>Photography</category>
    <name>Canon EOS 450D 18-55 IS Lens Kit</name>
    <description>12.2 Megapixel 7200 RPM Seagate 200 Gigabyte</description>
    <price>669.99$</price>
    <URL>http://www.digital-cameras.com/xpp-canon_eos_450d_18_55_is_lens_kit.html</URL>
  </PRODUCT>
- <PRODUCT>
    <category>LCD TVs</category>
    <name>SAMSUNG LE-37M87BD 37 94cm LCD TV.</name>
    <description>Component, 2 Adet Scart, 3 Adet HDMI, PC Giris, S-Video, Composite Video, RS232, DVB-T, Kulaklik cikis
      Var L/R, Optik ses, HDMI Version : Version 1.3 DVB Tuner : Var, LCD TFT aktif matris</description>
    <price>2536 USD</price>
    <URL>http://www.gold.com.tr/urunozel.asp?ID=39880</URL>
  </PRODUCT>
- <PRODUCT>
    <category>SAbit Diskler</category>
    <name>WD 320 GB 7200RPM 8MB SATA2</name>
    <description>320 GB 7200 RPM Veri Transfer Hizi 300 MBps</description>
    <price>88 USD</price>
    <URL>http://www.gold.com.tr/urunozel.asp?ID=40581</URL>
  </PRODUCT>
 </akhan>
```

Figure 6.4 Example XML Feed

The information in the Category tag is not taken into consideration. This product's best displayed category is determined with the analysis of keywords in the Description tag.

As shown in Figure 6.5, all the products in the feed are matched with the best fit category to e-mall by AGBEP.



Login as AKBULUT Elektronik Sanayi ve Ticaret A.S.

C:\Documents and Settings\Administrator\Desktop

category	name	description	price	URL
Photography	Canon EOS 450D 18-55 IS Lens Kit	12.2 Megapixel 7200 RPM Seagate 200 Gigabyte	669.99$	http://www.digital-cameras.com/xpp-canon_eos_450d_18_55_is_lens_kit.html
LCD TVs	SAMSUNG LE-37M87BD 37 94cm LCD TV.	Component, 2 Adet Scart, 3 Adet HDMI, PC Giris, S-Video, Composite Video, RS232, DVB-T, Kulaklik cikisi Var 3,5 mm , Audio Out Var, Audio Out Var L/R, Optik ses, HDMI Version : Version 1.3 DVB Tuner : Var, LCD TFT aktif matris	2536 USD	http://www.gold.com.tr/urunozel.asp?ID=39880
SAbit Diskler	WD 320 GB 7200RPM 8MB SATA2	320 GB 7200 RPM Veri Transfer Hizi 300 MBps	88 USD	http://www.gold.com.tr/urunozel.asp?ID=40581

Categories Determined by AGBEP

1 - Hard Drives
2 - Monitors & LCDs
3 - Hard Drives

SPECIALS | SITE SEARCH | CONTACT US | CREATE AN ACCOUNT | CUSTOMER LOGIN
Copyright © Akhan AKBULUT

Figure 6.5 Category Identification Screen

After this step, the member shop's products will be exhibited under the determined category in e-mall and will be ready to purchase.

7 PERFORMANCE EVALUATION

During the tests we used a feed with 700 products whose categories are known before they are added to the system during prototype testing. In the MALL database, nearly 20,000 products found under 30 main categories [7] are accepted as true, and we tested the AGBEP.

For testing the platform, we use 9 HP Compaq dc5700 micro tower computers which hardware specifications listed in Table 7.1.

Table 7.1: HP Compaq dc5700 Specifications

Processor	Intel Core2Duo 4300 @ 1.8Ghz
RAM	2GB
HDD	160GB 7200rpm
Operating System	Microsoft Server 2003 Enterprise with SP2
Network	Broadcom NetXtreme Gigabit Ethernet

7.1 Test with MD-5 and K-Means

As in the real-life scenario, after logging into the system as a member store, we uploaded our new catalogue. We wished our time performance to be high in our first test and limited the number of keywords that the Inspector Agent would choose to 20. Also, we identified the hashing method that the Hashing Agent would use as a MD5 algorithm. All the parameters and the results of the test displayed in Table 7.2. After the hashing process was completed, K-Means clustering process was applied by the system to 700 data with 20 parameters to an environment with 30,000 pieces comprised of 100 products randomly chosen, and first level results were achieved.

We observed that only 54 pieces from 700 products in the first level results were not forecasted to a specific category and were matched with the ‘Everything Else’ category. The category determination process took 6.2 seconds for 700 products in the first level. Figure 7.1 illustrates a small part of the first-level clustering process overview. Each point represents a product that is registered in the AGBEP’s database.

Table 7.2 Average results of the test with MD-5 and K-Means

Hashing Method	**Clustering Method**	**No # of Keywords**
MD5	K-Means	20
	Duration	**Accuracy**
1st Level Clustering	6.2 seconds	77.14 %
2nd Level Clustering	80.3 seconds	69 %

In the second-level clustering, we used the Fuzzy C-Means algorithm, again with 20 parameters by using the whole records in predicted categories for each of the 700 products. This process took 80.3 seconds with a success ratio of 69%.

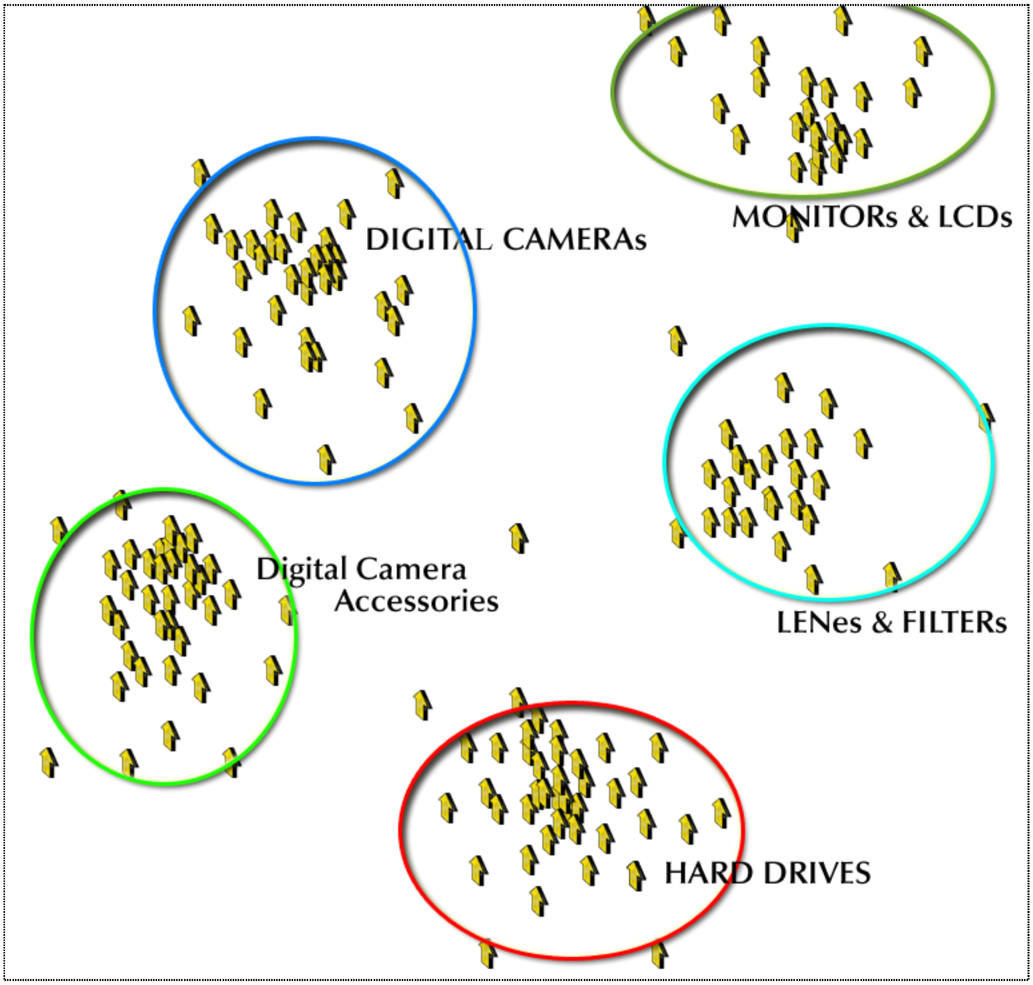


Figure 7.1 Products in clustering process

7.2 Test with SHA-1 and Fuzzy C-Means

In the second test (Table 7.3), we used the Fuzzy C-Means clustering algorithm as a parameter for the clustering agent and SHA1 parameter for the hashing agent. This clustering technique gave slightly better results than the first test. Results follow: complete accuracy was achieved on first-level clustering within 8.2 seconds. Second-level clustering was completed in 97.3 seconds with 87% accuracy.

Table 7.3 Average results of the test with SHA-1 and Fuzzy C-Means

Hashing Method	Clustering Method	No # of Keywords
Sha-1	Fuzzy C-Means	20
	Duration	**Accuracy**
1st Level Clustering	8.2 seconds	100 %
2nd Level Clustering	97.3 seconds	87 %

7.3 Test with SHA-512 and Fuzzy C-Means

We aimed for correctness rather than performance in the third test. So, keywords were chosen from their descriptions within the same 700 test products. We used the SHA512 hashing algorithm for these keywords and Fuzzy C-Means clustering for categorization as shown in Table 7.4.

In the first-level clustering, all of the 700 products were matched to the correct category in 310 seconds. When we used more keywords and a different hashing process, performance was directly affected.

When starting the second-level clustering, the matching of the totally true matched products with sub-categories took 1:40:35. According to the existing data, the correctness ratio was 96.1%.

The reason for the 3.9% mistake ratio is that when we cannot match the products with the categories, then we match them with the 'Everything Else' category. If we can make certain that the whole products categories exist in the system and increase the amount of products amount in the correct category then the success ratio will reach 100%.

Table 7.4 Average results of the test with SHA-512 and Fuzzy C-Means

Hashing Method	**Clustering Method**	**No # of Keywords**
Sha-512	Fuzzy C-Means	50
	Duration	**Accuracy**
1st Level Clustering	310 seconds	100 %
2nd Level Clustering	1.40:35 hour	96.1 %

The result of the tests showed us that, even if the clustering and the hashing methods are used, the ideal chosen keyword number is 52. Figure 7.2 shows us the correctness ratio that is getting by using different methods according to the chosen keyword numbers. The information we got so far shows us that to start the system successfully we need 50 and to get results that will not deceive us and that will not deplete the system resources we need 20 keywords to be find out.

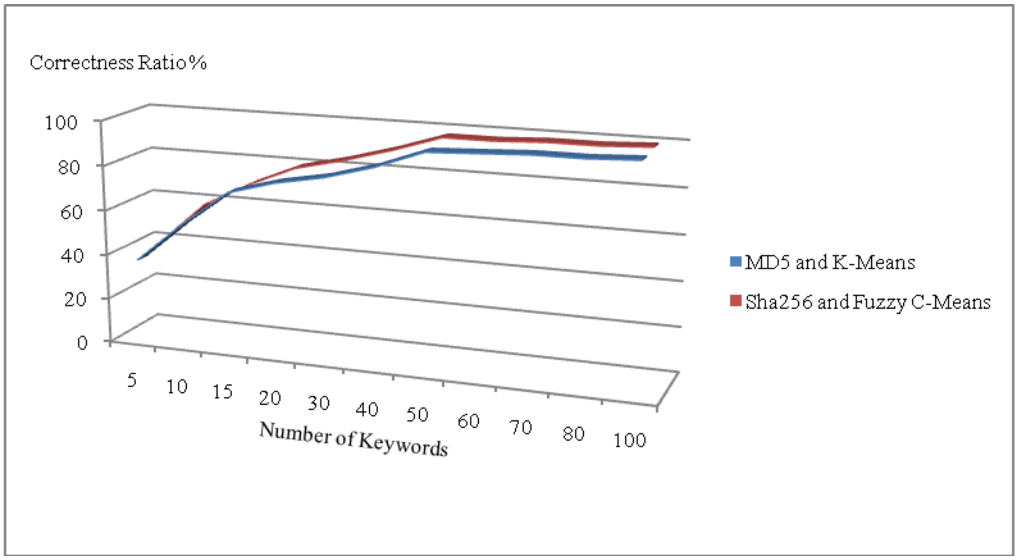


Figure 7.2 Tests of defining the ideal keyword selection as graph

Table 7.5 gathers all the test results together. These results designate that both the Fuzzy C-Means and K-Means clustering algorithms are successful to assemble products in groups and decide their categories. But, the C-Means algorithm performs better in the grouping process. And selecting the hashing method is related to one’s duration/correctness criteria. If the MD5 algorithm is selected as the hashing method, then more performance in process duration will be achieved but conversely correctness of category selection will be tightly failed. SHA algorithms demonstrate more performance, but the processes will take much time. Of all choices, the combination of the SHA512 as the hashing algorithm and the Fuzzy C-Means algorithm for clustering, gives the best performance.

Table 7.5 Performance tests and their results

	TEST 1	TEST 2	TEST 3
PARAMETERS			
No. of Keywords	20	20	50
Hashing Method	MD5	MD5	SHA512
Clustering Method	K-Means	Fuzzy C-Means	Fuzzy C-Means
RESULTS			
1st Level Clus. Time	6.2 sec.	8.2 sec.	310 sec.
1st Level Clus. %	77,14%	98,90%	100%
2nd Level Clus. Time	80.3 sec.	97.3 sec	1:40:35 hour.
2nd Level Clus. %	69%	87%	96.1%.

8 CONCLUSIONS

8.1 Overview

In this study, we developed an e-commerce platform using the existing intelligent agent platforms, hashing techniques and clustering methods. The categorization of the product information which we want to add to the platform, system is being carried out and is shown over e-mall in an autonomous way. As the product information is told detailed by member shops in chapter three, it is possible that it can be send in the requested format and its integration to the system is maintained by the agents that function over AGBEP’s agent server. In this process, after the keyword information of the products is determined, they are obtained to become suitable for clustering by hashing methods. After the second level clustering process that they are done by accepting the correctness of the categories of the products inside the system, the categories of the products inside the feed are determined and e-mall became ready for sales.

System performance has increased and data consistency has been achieved for AGBEP. During the development stage, we especially focused on the clustering process since it is the critical phase of the entire process. Also, the hashing process is another important phase for recovering product data before entering the clustering operation. By tuning these processes with intelligent agents, we originate a full performance autonomous job for e-commerce content management necessities.

While this autonomous structure works over the agent server in the system, some configuration is applied to web servers and database servers for the problem of maintaining high availability which is the most difficult problem that can occur. In order to balance the overload in the web servers and not to face any loss of performance, the NLB (Network Load Balancing) feature of the Windows 2003 Server are being used.

Suchlike, in order to balance the overload in database servers and not to face any loss of data, witness server feature that is the property of Microsoft SQL Server 2005 Enterprise Edition is being used.

8.2 Further Work

In order to advance AGBEP, we need to concentrate on some other subjects. Currently, products are matched to the categories and sub-categories defined in the system's existing infrastructure. Moving forward, a decision-making mechanism could be added that implements a dynamic category matching process for a category not yet defined in the system. In the meantime, we advise that, for a product that is far from the system's categories, a new category should be added to the system.

To gain more performance on the system, it is being advised that the load balancing process that is made in the web servers must be transferred to the physical load balancing devices. This solution is more expensive in terms of price but, works more efficient than the Windows Server 2003's software load balancing service.

As a second step, XQuery [45] language can be applied so that the Inspector Agent can analyze XML product feeds with much more higher performance. Extracting data from large XML documents causes difficulties on the entire system.

Finally, applying neural networks into system for decision support mechanisms could be much more effective for solving lack of intelligence in decentralized software systems, communication of software modules, and system learning problems [46, 47].

9 REFERENCES

[1] **Tang, O., Xie, F.,** 2006. A multi-agent system for e-commerce automation, *IEEE CCECE/CCGEI, Electrical and Computer Engineering, Canadian Conference,* Ottawa Page(s): 514-517

[2] **http://www.kelkoo.com/,** *Shopping and Price comparison European Leader*

[3] **http://www.shopzilla.com/,** *Shopzilla - Comparison shopping online*

[4] **http://www.froogle.com/,** *Google Product Search*

[5] **Akbulut A., Yılmaz G.,** 2007. An Architectural Model for Content Management in E-Commerce Applications, *Proceedings of IADIS e-Commerce 2007 Conference,* 7-9 Sept., Algarve - Portugal, Page(s): 260-264

[6] **http://www.gittigidiyor.com/,** *GittiGidiyor - Yeni ve ikinci el bilgisayar, araba, cep telefonu, giysi, elektronik ve daha fazlasý en düþük fiyatlarla burada!*

[7] **http://www.ebay.com/,** *eBay - New & used electronics, cars, apparel, collectibles, sporting goods &amp; more at low prices*

[8] **Hutter D., Mantel H., Schaefer I., Schairer A.**, 2007. Security of multi-agent systems: A case study on comparison shopping, *Journal of Applied Logic 5,* Page(s): 303–332

[9] **Lee J.K., Lee W.**, 1997, Intelligent Agent Based Contract Process in Electronic Commerce UNIK-AGENT Approach, *Proceedings of the Thirtieth Hawaii International Conference* on Page(s):230 - 241

[10] **Chen D., Jeng B., Lee W., Chuang C.,** 2008. An agent-based model for consumer-to-business electronic commerce, *Expert Systems with Applications 34* on Page(s): 469–481.

[11] **Lau R., Li Y., Song D., Kwok R.C.,** 2008. Knowledge discovery for adaptive negotiation agents in e-marketplaces. *Decision Support Systems,* (Unconditional Acceptance on 12-Dec-2007)

[12] **Marques P., Silva L., Silva J.,** 1999. Establishing a Secure Open-Environment for Using Mobile Agents in Electronic Commerce. *First International Symposium on*

Agent Systems and Applications Third International Symposium on Mobile Agents, Page(s) : 260-267

[13] **Chen H., Lam P., Chan H.C.B., Dillon T.S., Cao J., Lee R.S.T.,** 2007. Business-to-Consumer Mobile Agent-Based Internet Commerce System (MAGICS). *IEEE Transactions on Systems, Applications and Reviews*, Vol. 37, No. 6

[14] **Kang M., Park W., Koo J.,** 2003. Agent for electronic commerce on the semantic web. *Proceedings of the 71h Korea-Russia International Symposium*

[15] **http://www.pricerunner.co.uk/,** *PriceRunner UK - Compare Prices and Find Best Deals Online*

[16] **Pokorny, J.,** 2000. “XML Functionally”, *IEEE International Database Engineering and Applications Symposium*, Yokohoma, Japan, pages: 266-274

[17] **MacQueen J.B.**, 1967. Some Methods for classification and Analysis of Multivariate Observations, *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, University of California Press, 1:281-297

[18] **Dunn J.C.**, 1974. A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact, *Well Separated Clusters*, J. Cyber., 3, 32-57.

[19] **Bezdek J.C.**, 1981. Pattern Recognition with Fuzzy Objective Function Algorithms, *Plenum*, NY.

[20] **B. Schneier**, 1996. *Applied Cryptography*, John Wiley & Sons. ISBN 0-471-11709-9

[21] **Rivest R.**, 1992. The MD5 Message-Digest Algorithm, *MIT Laboratory for Computer Science and RSA Data Security, Inc.*

[22] **Secure Hash Standard,** 1993. *Federal Information Processing Standard Publication #180***,** US Department of Commerce, National Institute of Standards and Technology

[23] **Secure Hash Standard,** 2001. *Federal Information Processing Standard Publication #180-2***,** US Department of Commerce, National Institute of Standards and Technology

[24] **Russell S.J., Norvig P**., 2002. Artificial Intelligence: A Modern Approach, *Prentice Hall*; 2 edition, ISBN: 0137903952

[25] **Ferber J.,** 1999. Multi-agent Systems: Introduction to Distributed Artificial Intelligence, *Addison Wesley*

[26] **Wooldridge M., Jennings N.,** 1995. Intelligent Agents: Theory and Practice

[27] **Brooks R.A.**, 1991. Intelligence Without Reason, *Proceedings of the 12th International Joint Conference on Artificial Intelligence (*IJCAI-91)

[28] **Rao A., Georgeff M.,** 1995. BDI Agents: From Theory to Practice, *Proceedings of the First Intl. Conference on Multiagent Systems,* Technical Note 56

[29] **Finin T., Weber J., Wiederhold G., Genesereth M., Fritzson R., McKay D., McGuire J., Shapiro S., Beck C.,** 1993. *Specification of the KQML Agent-Communication Language,* The DARPA Knowledge Sharing Initiative External Interfaces Working Group

[30] **Genesereth M., Ketchpel** S., 1994. Software Agents, *Communications of the ACM,* Vol. 37

[31] **Durfee E.H.,** 1999. Distributed Continual Planning for Unmanned Ground Vehicle Teams, *AI Magazine,* Volume 20, Number 4

[32] **Smith R., Davis R., 1981**. Frameworks for Cooperation in Distributed Problem Solving, *Transactions On Systems,* Vol. Smc-11, No. 1

[33] **Durfee E., Lesser V., Daniel C.,** 1987. Coherent Cooperation Among Communicating Problem Solvers. IEEE *Transactions on Computers*, Volume C36, Number 11, Page(s): 1275-1291

[34] **Bussmann S., Müller J.**, 1992. A Negotiation Framework for Cooperating Agents. *In S.M.Deen (ed.), Proc. of the CKBS-SIG (CKBS'92), DAKE Centre, Univ. of Keele,* pp. 1-17.

[35] **FIPA TC Architecture Publications**, 2002. *FIPA Abstract Architecture Specification,* Document number : SC00001L

[36] **http://www.fipa.org/,** *Foundation for Intelligent Physical Agents*

[37] **http://jade.tilab.com** , Java Agent Development Framework

[38] **Bigus J., Bigus Jn.,** 2001. Constructing Intelligent Agents Using Java: Professional Developer's Guide Series, *John Wiley & Sons*; 2nd Edition

[39] **Lagus, K., Kaski, S.**, 1999. Keyword selection method for characterizing text document maps, *9th International Conference on Artificial Neural Networks-ICANN’99*, Edinburg, UK, pages: 371-376, vol.1 (Conf. Publ. No. 470)

[40] **Gutmann, P., Naccache, D., Palmer, C.C.**, 2005. When hashes collide [applied cryptography], *IEEE Security & Privacy Magazine*, Vol. 3, pages: 68- 71

[41] **Ordonez, C.**, 2006. Integrating K-means clustering with a relational DBMS using SQL, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 18, Issue: 2, on pages: 188 – 201,

[42] **Kaufman, L., Rousseeuw, P.**, 1990. Finding Groups in Data: an Introduction to Cluster Analysis, Publisher: *Wiley-Interscience*, ISBN: 0471878766

[43] **http://msdn.microsoft.com/,** *Microsoft Developer Network*

[44] **Shapiro J.**, 2006. *Microsoft SQL Server 2005: The Complete Reference,* Publisher: McGraw-Hill Osborne Media, ISBN-10: 0072261528

[45] **Funderburk, J.E.**, 2002. XML programming with SQL/XML and XQuery*, IBM Systems Journal, Dec*

[46] **Huhtinen, J.**, 2005. Utilization of neural network and agent technology combination for distributed intelligent applications and services, *Auditorium TS101*, Linnanmaa, on Nov. 4th

[47] **http://aosd.net/** , *Aspect-Oriented Software Development Community & Conference*