PERGAMON

Expert Systems with Applications 25 (2003) 589–602

Expert Systems with Applications

www.elsevier.com/locate/eswa

Intelligent policy recommendations on enterprise resource planning by the use of agent technology and data mining techniques

Andreas L. Symeonidis[a,b,*], Dionisis D. Kehagias[a,b], Pericles A. Mitkas[a,b]

[a]*Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Panepistimioupoli, 54124 Thessaloniki, Greece*
[b]*Laboratory of Intelligent Systems and Software Engineering, Informatics and Telematics Institute/CERTH, Thessaloniki, Greece*

Abstract

Enterprise Resource Planning systems tend to deploy Supply Chain Management and/or Customer Relationship Management techniques, in order to successfully fuse information to customers, suppliers, manufacturers and warehouses, and therefore minimize system-wide costs while satisfying service level requirements. Although efficient, these systems are neither versatile nor adaptive, since newly discovered customer trends cannot be easily integrated with existing knowledge. Advancing on the way the above mentioned techniques apply on ERP systems, we have developed a multi-agent system that introduces adaptive intelligence as a powerful add-on for ERP software customization. The system can be thought of as a recommendation engine, which takes advantage of knowledge gained through the use of data mining techniques, and incorporates it into the resulting company selling policy. The intelligent agents of the system can be periodically retrained as new information is added to the ERP. In this paper, we present the architecture and development details of the system, and demonstrate its application on a real test case.

Keywords: Supply Chain Management; Customer Relationship Management; Data mining; Multi-agent systems; Agent training

1. Introduction

In a typical supply chain, raw materials are procured, items are produced at one or more company sites, shipped to warehouses for intermediate storage, and then sent to retailers or customers. Consequently, effective Supply Chain Management (SCM) strategies are applied at various stages of the process, in order to reduce cost and improve service levels (Levi, Kaminsky, & Levi, 2000).

On the other hand, Customer Relationship Management (CRM) techniques are frequently applied to enable companies to master the basics of building customer focus, i.e. move from a product orientation to a customer orientation and define their market strategy from outside-in instead of inside-out. Customer orientation can be fostered through the integration of CRM across the entire customer experience chain, by leveraging technology to achieve real-time customer management (Rygielski, Wang, & Yen, 2002).

These two technologies have mainly been employed separately due to their increased complexity (Barbuceanu & Fox, 1994; Patterson, Grimm, & Corsi, 2003; Shen, Xue, & Norrie, 1998; Zeng & Sycara, 1999) and slight scope declination. Nevertheless, there have been some efforts to integrate them and exploit the advantages of such a coalition (Choy, Lee, & Lo, 2002; Choy, Lee, & Lo, 2003; Heikkila, 2002). These systems use the basic concepts of SCM and CRM and try to combine them, in an effort to produce a more sophisticated quality of services. Efficient may they be, the already developed systems are neither versatile nor adaptive, since newly discovered customer trends and changes in company policy cannot be easily incorporated into the system's backbone. In addition, the notions of synergy and collaboration, which are compulsory to such kinds of expert systems are not properly met, whereas their corresponding architecture is not always optimal.

Such systems facilitating Supply Chain and Customer Relation primitives (SC-CR systems) can be viewed as networks of collaborative, yet autonomous, units that

* Corresponding author. Address: Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Panepistimioupoli, Thessaloniki 54124, Greece. Tel.: +30-2310-99-6349; fax: +30-2310-99-6398.

E-mail addresses: asymeon@iti.gr; asymeon@ee.auth.gr (A.L. Symeonidis).

0957-4174/03/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/S0957-4174(03)00099-X

regulate, control and organize all distributed activities involved in procurement, manufacturing, order processing, order transaction and product distribution. Research literature on intelligent agent system architectures has proven that problems that are inherently distributed or require the synergy of a number of distributed elements for their solution can be efficiently implemented as a multi-agent system (MAS) (Jennings, Sycara, & Woolridge, 1998; Ferber, 1999). Thus, multi-agent technology constitutes a powerful technology for developing SC-CR systems.

In a MAS realizing a SC-CR system, all requirements collected by the end users are perceived as distinguished roles of separate agents, acting in close collaboration. All agents participating in MAS communicate with each other by exchanging messages, encoded in a specific Agent Communication Language (ACL). Each agent in the MAS is designated to manipulate the content of the incoming messages and take specific actions/decisions that conform to a particular reasoning mechanism designed by the agent programmer.

Another technology that has been widely used for solving CRM problems is data mining (DM). DM, which is defined as the extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) information or patterns from data in large databases (Fayyad, Piatetsky-Shapiro, & Smyth, 1996). DM has been recognized by many researchers as a key research topic in database systems and machine learning and considerable effort has been spent on the development of a large class of AI applications to exploit DM techniques. Market-basket analysis and customer segmentation (Amir, Feldman, & Kashi, 1997; Chen, Han, & Yu, 1996; Ganti, Gehrke, & Ramakrishman, 1999; Han & Kamber, 2001; Hong, Kuo, & Chi, 1999) are major CRM areas data mining has been applied on.

In this context, the presented work is focused on developing not only a DM framework for the identification of CRM and SCM patterns, but also on the design and implementation of a multi-agent system for exploiting the results of the DM procedure. More specifically, the developed multi-agent architecture combines multi-agent and data mining technologies in order to provide intelligent and adaptive policy recommendations, created on knowledge extracted through the use of data mining techniques. IPRA (Intelligent Policy Recommendation multi-Agent system) is an add-on to existing ERP systems, providing the ERP operator with customer/inventory/supplier useful recommendations, based Customer/Supplier Clustering and on Association Rule Extraction on item transactions. In particular, an agent that represents the current customer transaction collects all necessary information that concern the details of an order and contacts the appropriate IPRA agents, in order to finally get a set of recommendations by the system. These recommendations are tailored to each customer and his/her order, since clustering on selected customer data and transaction history is performed. In an analogous manner, supplier added value is discovered, whereas ordering habits are discovered and exploited through the system.

The main objective of IPRA is the optimization of the quality of services provided by the existing ERP, which provides a robust means for storing and manipulating a large amount of data on company transactions. The choice of developing IPRA as a MAS, provides the advantage of untroubled modification and extension of the system, according to altering company requirements. It should be mentioned that IPRA can extend any existing legacy database, containing customer and supplier data, and therefore increase the added value of the system.

It can be thought of as an intelligent system, since it increases its intelligence by embedding knowledge to the intelligent agents of the system. This knowledge is extracted by applying DM techniques on enterprise data in order to identify and exploit specific patterns among customers, suppliers and inventory items. Special care has been given to the agents that are designed to produce the recommendations of the system.

IPRA has been primarily tested on an existing ERP (over 25,000 of data records concerning transactions, over 8000 customers, over 500 suppliers) and the results seem quite promising. The IPRA approach seems that it can significantly increase the enterprise service level, in terms of delivery time, discount and correlated recommendations.

The rest of this paper is organized as follows: Section 2 introduces the implemented multi-agent system in detail and describes the functional characteristics of the different types of agents that comprise IPRA. In Section 3 the data mining methodology that is used in order to augment the agent decision quality, both in terms of intelligence and autonomy is presented. Finally, Section 4 illustrates the basic functional operations of IPRA and outlines the test case developed in the real enterprise environment, while Section 5 summarizes the work described and concludes this paper.

2. System architecture

2.1. IPRA use case description

The implemented MAS is illustrated in Fig. 1. Thin arrows represent messages exchanged between agents, while thick arrows correspond to data transfer from/to the MAS.

Upon receiving an order, an agent representing the customer collects all the necessary information, in order to provide the other IPRA agents with input. Collected data include the customer name and id, his/her geographic location, the list of items ordered and the corresponding quantity, as well as the customer's preferred payment terms, i.e. cash, by check, by credit card etc. Customer info are sent to an agent responsible for customer segmentation, which decides on the discount to be made to the particular customer.

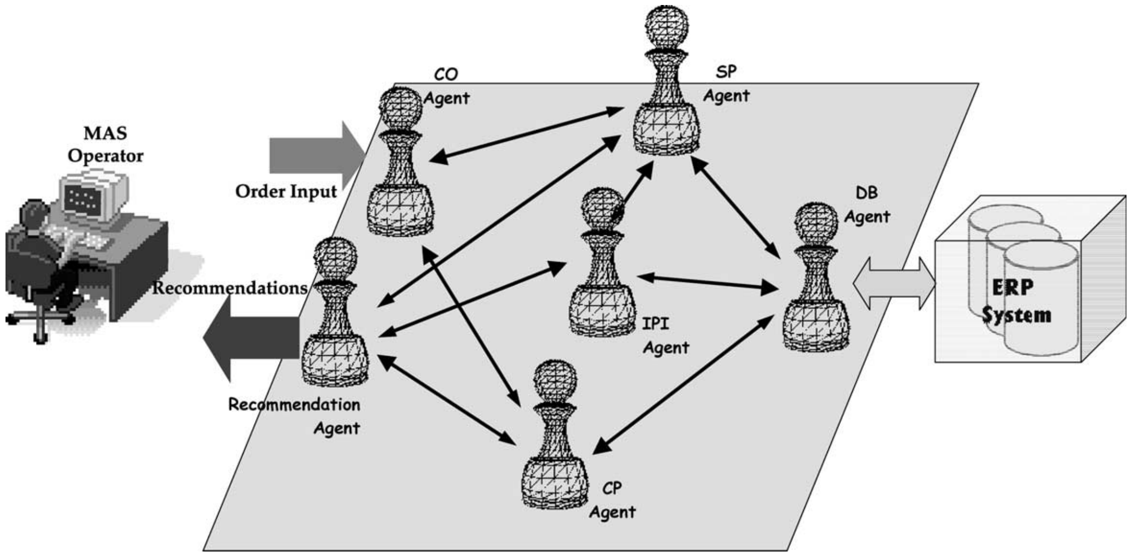

Fig. 1. The functional diagram of IPRA.

This agent also decides on customer priority, a metric that indicates customer 'quality', i.e. the added value of the customer's behaviour. Information on the list of ordered items are sent to an agent responsible for the company inventory, in order to provide a recommendation for additional items that could be bought within the same order invoice. Supplier info are sent to an agent responsible for supplier segmentation, which decides on Supplier credibility, therefore providing a reliable estimation of products due time. Finally, the decisions made on customer, supplier and inventory policy are reported to another agent, that provides the final recommendation on order discount, estimated time of order processing and delivery time, as well as a splitting policy, depending on customer priority, available product stock, and order turnover.

It should be denoted that the recommendations IPRA produces are extracted through the application of data mining techniques on historical data stored in the ERP.

More specifically, IPRA has six different types of agents:

1. The Database Agent (DBA)
2. The Customer Order Agent (COA)
3. The Customer Profile Agent (CPA)
4. The Inventory Pattern Identification Agent (IPIA)
5. The Supplier Profile Agent (SPA)
6. The Recommendation Agent (RA).

2.1.1. The Database Agent

DBA is responsible for connecting to the existing ERP and for fetching all the requested data. It exchanges messages with all the agents that need to query the ERP in order to fulfill their tasks. In addition, DBA calculates and provides some of the attributes used by the CPA and SPA data mining mechanisms. Due to the nature of DBA, connectivity to any existing data storing mechanism can be easily established and maintained. DBA provides customizability and versatility to IPRA.

2.1.2. The Customer Order Agent

COA handles incoming orders, which can enter the system asynchronously. It provides a Graphical User Interface (GUI) for representing the customer order and for collecting all the necessary information, namely customer id, ordered items and payment terms. The COA sends the appropriate information to the corresponding IPRA agents.

2.1.3. The Customer Profile Agent

CPA manipulates customer data, in order to identify the company CRM policy. CPA receives its input from COA and DBA. The activity diagram of CPA is illustrated in Fig. 2.

2.1.4. The Supplier Profile Agent

SPA manipulates supplier data, in order to identify the most suitable supplier to order from, therefore determining a policy regarding related suppliers. SPA's activity diagram is illustrated in Fig. 3.

2.1.5. The Inventory Pattern Identification Agent

IPIA manipulates data concerning the order itself and proposes additional items to be ordered, based on the market basket analysis paradigm. IPIA also receives its input from the COA and the DBA. IPIA's activity diagram is illustrated in Fig. 4.

2.1.6. The Recommendation Agent

The Recommendation agent (RA) takes into account the CPA, SPA and IPIA outputs and generates a set of recommendations in order to best satisfy the incoming

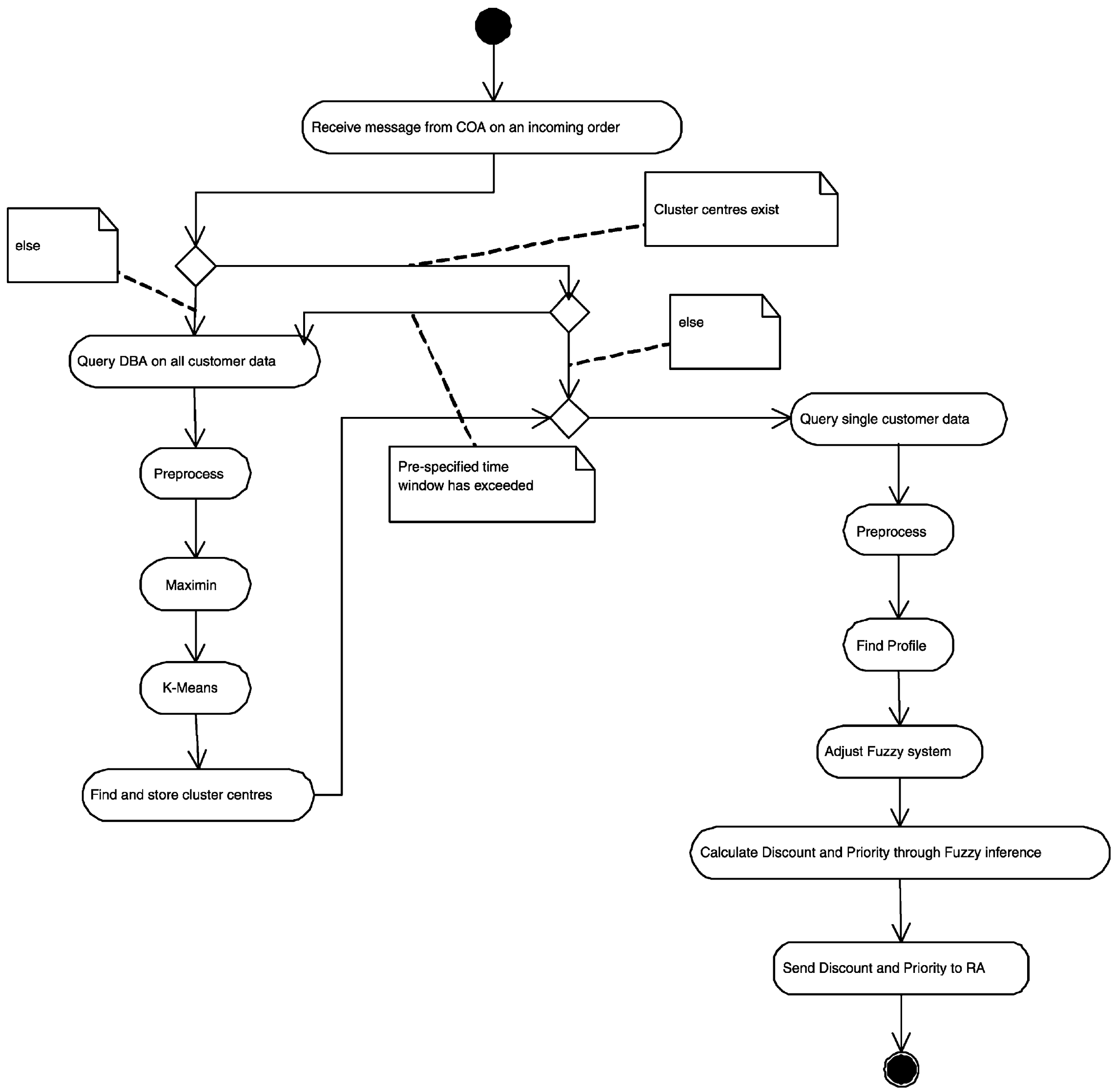

Fig. 2. The activity diagram of CPA, describing its possible actions.

order. More specifically, RA comes with a GUI, that provides recommendations on Customer priority, Order discount, Supplier credibility and Order splitting policy, as well as a suggestion for new, order-related items to buy. The activity diagram of RA is illustrated in Fig. 5.

2.2. Technologies adopted

IPRA has been developed with the use of Agent Academy[1] (AA) (Agent Academy Consortium, 2000; Mitkas, Symeonidis, Kechagias, & Athanasiadis, 2002; Symeonidis, Mitkas, & Kechagias, 2002) a platform for developing MAS architectures and for enhancing their functionality and intelligence through the use of DM techniques (Fig. 6). All the agents are developed over the Java Agent Development Framework (JADE) (Bellifemine, Poggi, & Rimassa, 2000), and Protege (Grosso, Eriksson, Fergerson, Gennari, Tu & Mus, 1999) which conforms to the FIPA specifications (FIPA, 2000), while the required ontologies have been developed through the Agent Factory module (AF) of AA. Data mining has been performed on ERP data through the Data Miner (DM) of AA, which expands the Waikato Environment for Knowledge Analysis (WEKA) tool (Witten & Frank, 1999).

The extracted knowledge structures are represented in Predictive Model Markup Language (PMML), a language that efficiently describes clustering, classification and association rule knowledge models (Data Mining Group, 2001). The resulting knowledge has been incorporated into the agents by the use of the Agent Training Module (ATM) of AA. All necessary data files (ERP data, agent behaviour data, knowledge structures, agent ontologies) are stored into

[1] The Agent Academy platform development has been partially supported by the European Commission through the IST initiative (IST project No 2000-31050).

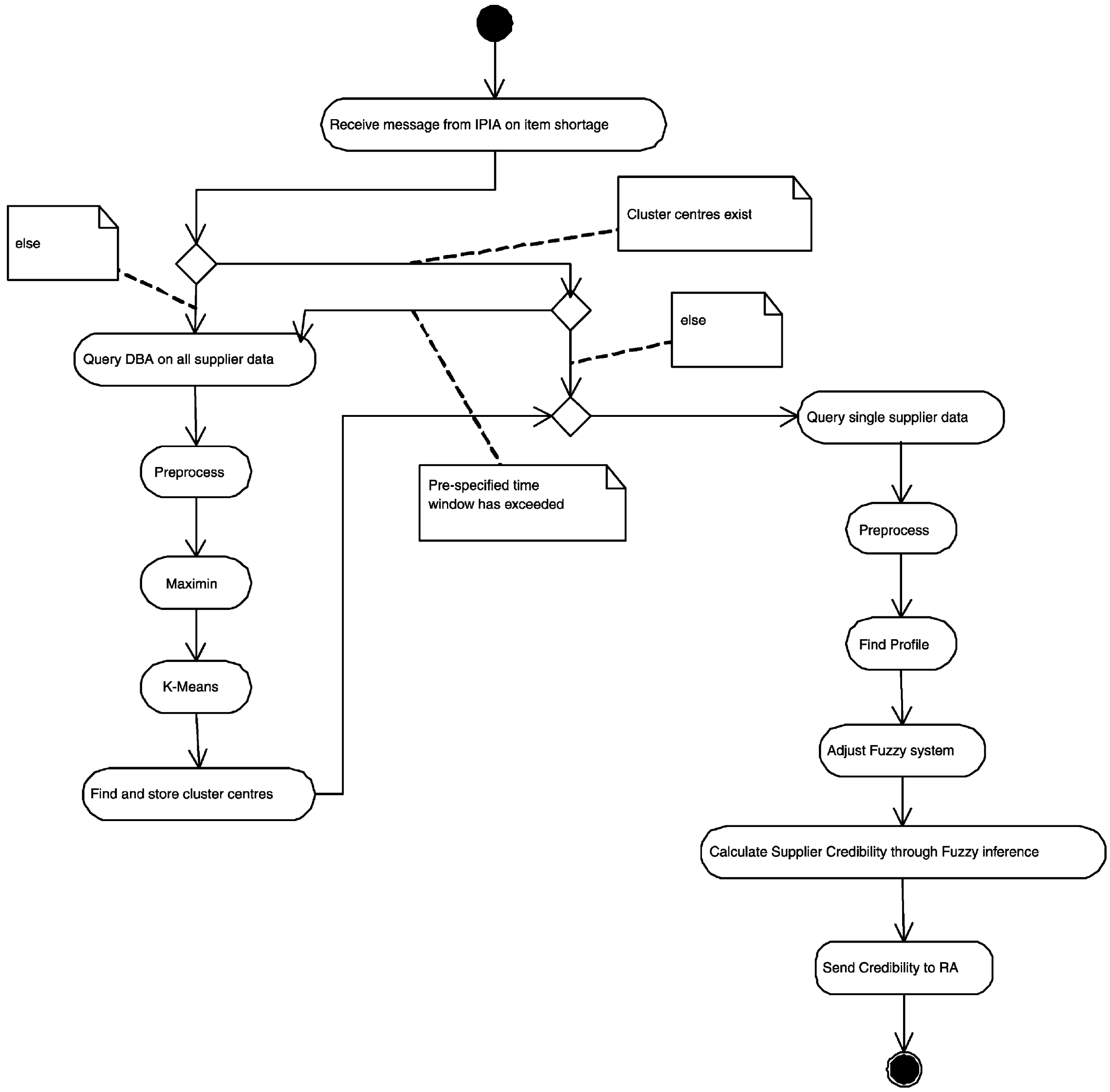

Fig. 3. The activity diagram of SPA.

AA's main database, the Agent Use Repository (AUR). Agents can be periodically recalled for retraining, since appropriate agent tracking tools have been incorporated into Agent Academy, in order to monitor agent activity after their deployment.

3. Augmenting agent intelligence by the use of data mining techniques

In order to augment agent intelligence, data mining is performed on three different levels. All the necessary data are acquired from the ERP system via the DBA. The agents that take decisions based on knowledge obtained through data mining techniques are the CPA, the SPA and the IPIA, whereas the RA employs an expert system, in order to specify the order splitting policy.

3.1. CPA intelligence

In order for the CPA to decide on customer value, clustering according to their financial record is performed. More specifically, CPA gets the customer id from the COA and checks his/her profile with respect to the already discovered (through data mining) cluster centres.

CPA performs clustering on the attributes illustrated in Table 1. Some of the dataset attributes are metrics calculated by the DBA, whereas others are raw data extracted from the ERP system.

In order for the CPA to decide on the number of customer clusters, we use the Maximin algorithm (Looney, 1997), which provides an optimal number of clusters, based on pattern recognition theory. The proposed number of clusters is then inputted into the K-Means clustering algorithm (Han & Kamber, 2001; MacQueen, 1967), in order to find customer clusters. The cluster centres are embedded into

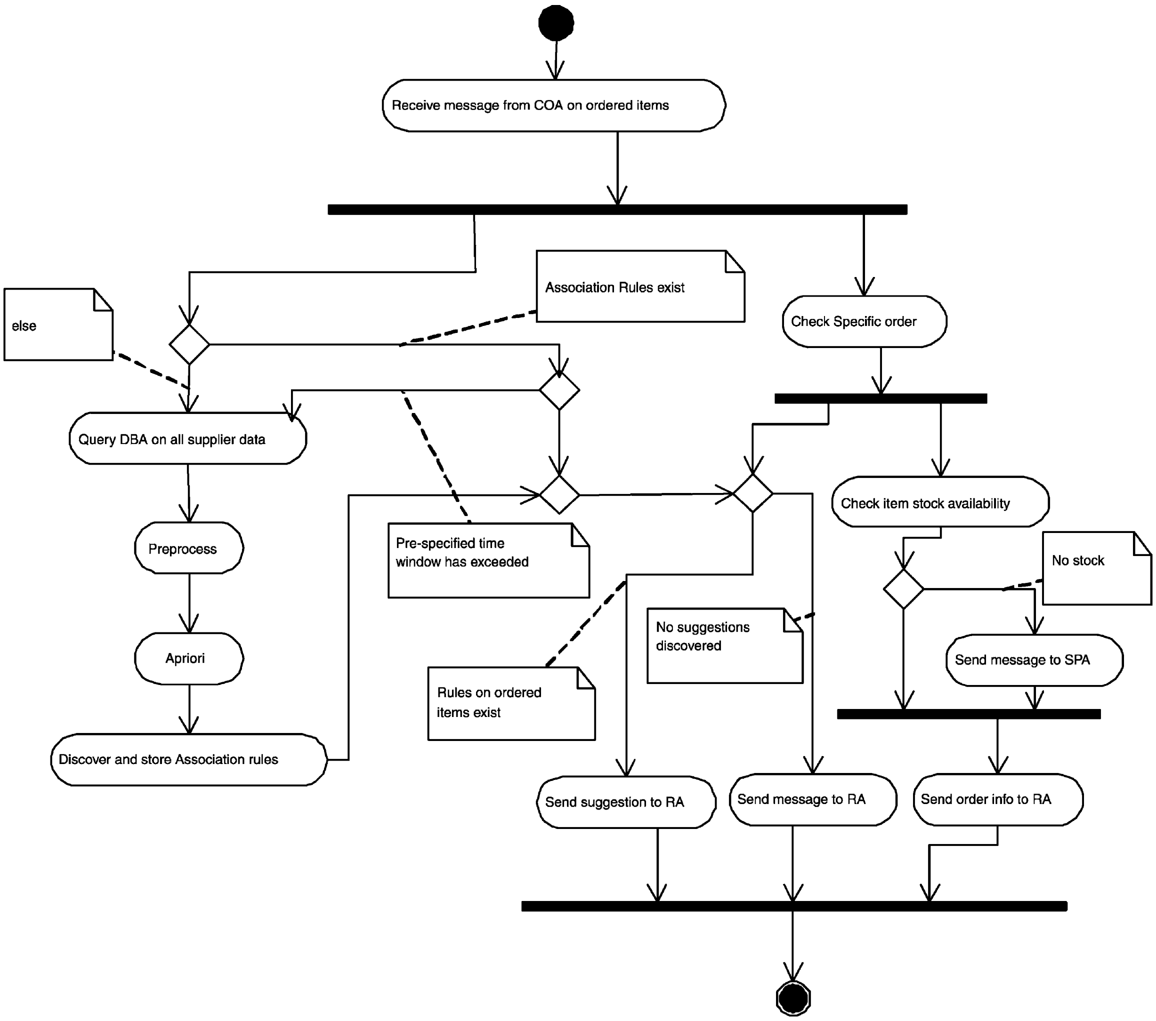

Fig. 4. The activity diagram of IPIA.

CPA and each customer is assigned to a cluster. The Customer Assigned Cluster (CAC), is a closeness-to-cluster-centre function, given by the following equation:

$$\mathrm{CAC} = \min_{i=1\ldots k}\left\{\sqrt{\sum_{j=1}^{n}(xc_i - c_{ji})}\right\},$$

where k is the number of customer clusters, n the number of attributes, xc_i is the ith attribute value of the customer vector $xc = (xc_1, xc_2, \ldots, xc_n)$, and c_{ji} the ith attribute value of the jth cluster centre vector $c_j = (c_{j1}, c_{j2}, \ldots, c_{jn})$.

In order to decide upon the recommended discount and customer priority, the CPA uses an adaptive fuzzy logic inference engine. Having incorporated domain knowledge into the system (Brafman & Tennenholtz, 1997; Delgado, Sanchez, Martin-Bautista, & Vila, 2001; Freitas, 1999; Kalles, 1994), the CPA dataset attributes that are used as input variables for the fuzzy engine, as well as the output variables are illustrated in Table 2. In case the company would benefit from the increase of a certain attribute, the 'Preferred Tendency' value is denoted as ↗, whereas a decrease of the attribute is denoted as ↘.

AOP, AOI and AOT are used as inputs only when:

$$\frac{\mathrm{STD(AM)}^2}{\mathrm{AM}^2} < \xi,$$

where AM stands for *Average Metrics* (AOP, AOI, APT) and ξ is a pre-specified threshold. This constraint is used in order to avoid polarization.

Since all inputs have 3 fuzzy values (*[LOW, MEDIUM, HIGH]*) the *Fuzzy Rules* (FR) produced are 3^v, where v is the number of inputs for the fuzzy engine. The *FRs* are of the type:

If $X1$ is $LX1(k)$ and $X2$ is $LX2(k)$ and…and Xn is $LXn(k)$
Then U is $LU(k)$, $K = 1, \ldots, m$

Triangular membership functions are adopted for all the inputs and outputs, whereas maximum defuzzification is used for crisping the FRs.

All inputs are given a *Corresponding Value (CV)*, ranging from −1 to 1, according to their company benefit criterion, illustrated in Table 2. The *Output Value* (OV) for

Fig. 5. The activity diagram of RA.

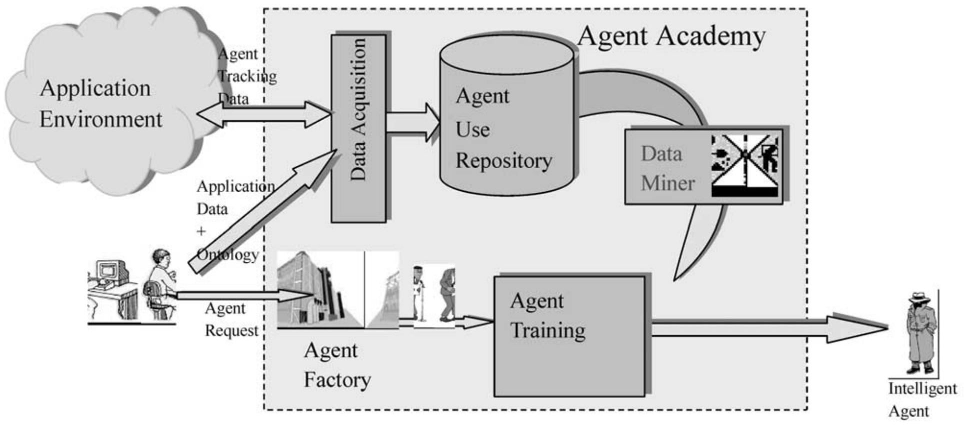

Fig. 6. The Agent Academy platform.

Table 1
Attributes of the CPA profiling dataset

Attribute	Description
Account balance (AB)	The amount of money a customer owes to the company
Credit Limit (CL)	The maximum account balance that the company allows to a particular customer
Turnover (TO)	The turnover a particular customer has brought to the company
Average Order Periodicity (AOP)	A metric that indicates how often the customer orders from the company
Standard deviation of Order Periodicity (STDOP)	A metric that indicates cyclic or asynchronous customer behaviour on orders
Average Order Income (AOI)	A metric that shows customer trends, as far as order profit for the company is concerned
Standard deviation of Order Income (STDOI)	A metric that indicates cyclic or asynchronous customer behaviour on order profit
Average Payment Terms (APT)	For the 8 different ways of payment, a number from 1 to 8 is assigned, and the average is calculated for each customer
Standard deviation of Payment Terms (STDPT)	The standard deviation for customer payment habits
Customer Geographic Location (CGL)	The geographic distance of the customer from the company

discount is then calculated for each *FR* as:

$$\mathrm{OV} = \sum_{i=1\ldots n} \mathrm{CV}$$

where n is the number of inputs.

Table 2
Fuzzy variable definition and Interestingness of CPA dataset attributes

Variable		Fuzzy tuple
Input	Preferred Tendency	
AB	↘	<AB, [Low, Medium, High], [$X1$, $X2$], Triangular>
CL	↗	<CL, [Low, Medium, High], [$X1$, $X2$], Triangular>
TO	↗	<TO, [Low, Medium, High], [$X1$, $X2$], Triangular>
AOP	↘	<AOP, [Low, Medium, High], [$X1$, $X2$], Triangular>
AOI	↗	<AOI, [Low, Medium, High], [$X1$, $X2$], Triangular>
APT	↘	<APT, [Low, Medium, High], [$X1$, $X2$], Triangular>
CGL	↘	<GCL, [Low, Medium, High], [$X1$, $X2$], Triangular>
Output	Value range	
Discount	Varies from 0 to 30%, using a step of 5%	<DSC, [Very_Low, Low, Mid_Low, Medium, Mid_High, High, Very_High], [0, 30], Triangular>
Priority	Varies from 0 to 3, using a step of 1	<PRT, [Low, Medium, High], [0, 3], Triangular>

Table 3
OV–FV lookup table

Output value	Fuzzy value
−6, −7	VERY_LOW
−4, −5	LOW
−2, −3	MID_LOW
−1, 0, 1	MEDIUM
2, 3	MID_HIGH
4, 5	HIGH
6, 7	VERY_HIGH

The OV–Fuzzy Values (FV) lookup table is illustrated in Table 3.

The final Discount and Priority values for each customer are sent to the RA in the form of an Agent Communication Language (ACL) message, encoded with respect to the FIPA-SL0 semantics definition (FIPA-SL Specifications, 2000).

3.2. SPA intelligence

The SPA's primary role is to cluster suppliers according to their supplying policy. More specifically, SPA receives information on the items ordered and decides on the appropriate supplier in case of insufficient stock. The IPIA generates the IDs for the suppliers who can provide the requested items and SPA assigns each possible supplier to a pre-determined cluster. Supplier clusters are formatted in a manner similar to that of customer.

The dataset that the SPA performs clustering on has the attributes listed in Table 4. Some of these data values are metrics calculated by the DBA and not raw data derived from the ERP system.

Table 4
Attributes of the SPA profiling dataset

Attribute	Description
Account balance (AB)	The amount of money the company owes to the supplier
Credit Limit (CL)	The maximum account balance that the company is allowed
Turnover (TO)	The turnover the company has brought to the supplier
Average Order Completion (AOC)	A metric that indicates the time slot in which the supplier carries out orders from the company
Standard deviation of Order Completion (STDOC)	A metric that indicates cyclic or asynchronous supplier behaviour on order completion
Average Payment Terms (APT)	For the 8 different ways of payment, a number from 1 to 8 is assigned, and the average is calculated for each customer
Standard deviation of Payment Terms (STDPT)	The standard deviation for customer payment habits
Supplier Geographic Location (SGL)	The geographic distance of the supplier from the company
Supplier Geographic Location (SGL)	The geographic distance of the supplier from the company

Table 5
Fuzzy variables and Preferred Tendency of SPA dataset attributes

Variable	
Input	Preferred Tendency
AB	↘
CL	↗
TO	↗
AOC	↘
APT	↘
CGL	↘
Output	Value range
Discount	Ranging from 0 to 1, using a step based on the number of supplier clusters

The Maximin and *K*-means algorithms are used in order to decide on the number of supplier clusters and the cluster centres, respectively. These cluster centres are then embedded into the SPA.

The recommended *Supplier Value (SV)* is the output of the second adaptive fuzzy logic inference engine of the proposed MAS. In an analogous to Table 2 manner, Table 5 illustrates the Preferred Tendency for each attribute and the possible input and output variables of the fuzzy engine.

The final recommendation on the most suitable supplier and his corresponding SV are sent to the RA in the form of a FIPA-SL0 message.

3.3. IPIA intelligence

The IPIA has a dual role in the system:

(i) It fetches information on price, stock and related suppliers of the ordered items, and
(ii) it provides recommendations on additional items to buy, based on association rule extraction techniques.

More specifically, the data that IPIA fetches are illustrated in Table 6.

In order to provide adaptive recommendations on ordering habits, IPIA incorporates knowledge extracted by the use of association rule extraction techniques. The Apriori algorithm is used (Amir et al., 1997; Ganti et al., 1999; Han & Kamber, 2001) and the rules extracted are embedded to the IPIA via the ATM of Agent Academy.

Special attention should be drawn to the fact that the transactions included into the dataset to be mined may span several different customer orders. The dataset comprises of transactions formatted by items ordered by a customer over a pre-specified period of time *Pt*. *Pt* depends on the nature of the company that uses the proposed MAS and the time frame–order correlation. In the case of a company selling toys, *Pt* could have the value of 2 weeks, since during holidays, sales increase substantially.

The recommendations of IPIA, as well as the information concerning stock availability and price, are sent to the RA, whereas the corresponding supplier ids are sent to the SPA.

3.4. The Recommendation Agent (RA)

RA is the final recommendation agent. It comes with a GUI, unlike the other intelligent agents of the system. Apart from providing the user with decisions (suggestions) made by the CPA, SPA and IPIA, RA can determine an efficient order fulfilling policy by the use of a JESS expert rule engine (Friedman-Hill, 2003). The engine, with the inputs shown in Table 7, can decide whether the order should be split, what an estimate delivery time would be and when should an order be placed to the suppliers.

The Re-order metric s is a no-fixed-order-cost value, applied for efficient inventory management (Levi et al., 2000). The reorder point is calculated as:

$$s = \mathrm{AIT} \times \mathrm{AOC} + z\sqrt{\mathrm{AOC} \times \mathrm{STDAIT}^2 + \mathrm{AIT}^2 \times \mathrm{STDAOC}^2},$$

where AIT × AOC represents average demand during lead time while $\sqrt{\mathrm{AOC} \times \mathrm{STDAIT}^2 + \mathrm{AIT}^2 \times \mathrm{STDAOC}^2}$ is the standard deviation of demand during lead time. Therefore, the amount of safety stock that has to be kept is:

$$z\sqrt{\mathrm{AOC} \times \mathrm{STDAIT}^2 + \mathrm{AIT}^2 \times \mathrm{STDAOC}^2}$$

where z is a constant chosen from statistical tables to ensure the satisfaction of a pre-specified value for the company Service Level. Table 8 illustrates the value of z in

Table 6
The attributes of IPIA

Variables
Stock Availability (SA)
Item price
Supplier ids
Average Item Turnover (AIT) for the last two years
Monthly Standard Deviation of AIT (STDAIT)

Table 7
Input variables of the RA JESS engine

Input variables
Customer Geographic Location (CGL)
Supplier Geographic Location (SGL)
Stock Availability (SA)
Ordered Quantity (OQ)
Lower Order Break-point (LOB)
Upper Order Break-point (UOB)
Re-order metric (s)

Table 8
Service level and corresponding z value

Service level	90%	91%	92%	93%	94%	95%	96%	97%	98%	99%	99.9%
Z	1.29	1.34	1.41	1.48	1.56	1.65	1.75	1.88	2.05	2.33	3.08

correlation with the desired Service Level: RA calculates and informs IPIA for re-ordering each time a need rises.

The splitting policy is initiated when the company stock availability cannot satisfy order needs. When a new order arrives, the quantity of ordered items and available stock are crosschecked. In case the requested quantities are available, the order is fulfilled immediately. Otherwise, the final supplying policy that the RA recommends, is set according to the the schema illustrated in Fig. 7.

The LOB and UOB thresholds depend the estimated customer value. Customers that enjoy better discount and have a higher priority, have a lower LOB and an upper UOB, according to the following equations:

$$\mathrm{LOB} = 0.5\exp[-(0.6\hat{p} + 0.316\hat{d})],$$

$$\mathrm{UOB} = 0.7\exp(0.189\hat{p} + 0.062\hat{d}),$$

where $\hat{p}$ is the priority normalized factor, $\hat{d}$ is the discount normalized factor, while the selected weighting factors satisfy minimal requirements on LOB and UOB range.

In case the available stock does not exceed LOB% of the ordered quantity, then the whole order in postponed until the company has been supplied with the ordered item. When the available stock falls into the [LOB − UOB]% of the ordered quantity, the order is split, all the available stock is sent to the customer, whereas the rest is ordered from the appropriate suppliers. Finally, in case the available stock exceeds UOB% of the ordered quantity, the order is immediately fulfilled and the remaining order percentage is passed over.

4. The test case

In order to study IPRA's performance, a test case was developed. The system was incorporated into the IT environment of a large retailer in the Greek market, hosting an ERP system with a sufficiently large data repository. IPRA was slightly customized to facilitate access to the existing Oracle™ database. Our system proved itself capable of managing over 25,000 transaction records, resulting in the extraction of truly 'smart' suggestions. The CPA and the SPA performed clustering of over 8000 customers and 500 suppliers, respectively. A detailed presentation of the execution phases and the final results can be found in subsequent sections.

4.1. Inserting a new order into the system

The first phase of the execution process is initiated upon a new order arrival. New requests are handled by the COA, which collects input from the customer and prepares all data required by the other agents (CPA, IPIA, SPA and RA), before they initialize and execute the recommendation-extraction mechanism. Data collection is realized through a GUI, illustrated in Fig. 8, which is comprised of two parts: one dedicated to the collection of order-specific information (Fig. 8a) and another for customer-specific information (Fig. 8b).

Each time the MAS operator enters the system (on behalf of a customer), he/she enumerates (or declares) explicitly the type and quantity of each item that must be included in the order. Next, the customer chooses a payment method and informs the system on whether the order must be delivered to his/her place. Finally, the order is completed by the provision of customer-specific details, which include customer's name, a unique customer identifier and customer's geographic location. At this point, the operator initializes the order-processing procedure and waits for the system's recommendations.

4.2. Formatting intelligence

Order processing by IPRA mainly involves the execution of clustering performed by the CPA, the SPA and the IPIA. For the specific company, CPA identified five major clusters representing an equal number of customer groups. The resulting customer clusters, as well as the discount and priority, calculated by the CPA Fuzzy Inference Engine for each cluster, are illustrated in Table 9.

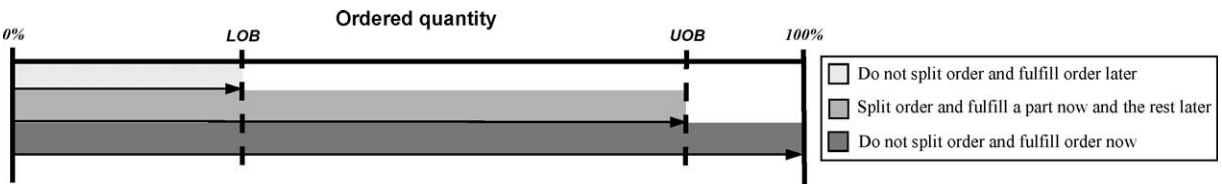

Fig. 7. RA order splitting policy.

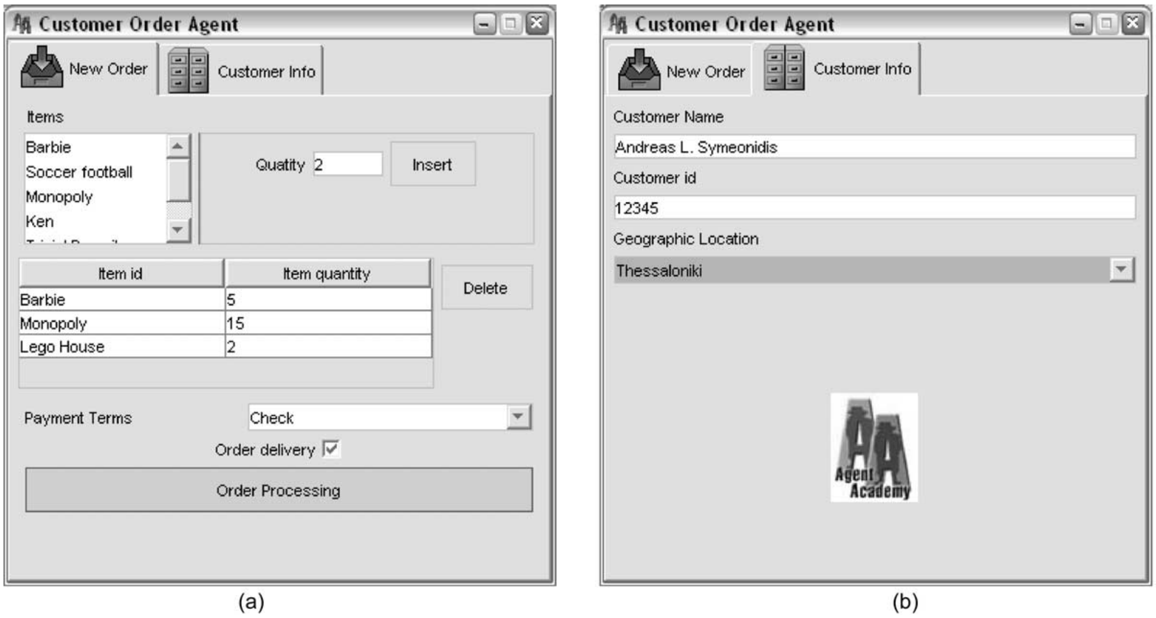

Fig. 8. The GUI of the Customer Order Agent with information on (a) ordered items and quantities and (b) customer details.

All attributes of the extracted clusters are expressed in a formal knowledge representation schema structured in a PMML document. Part of the resulting schema is illustrated in Fig. 9. This document represents the main part of the knowledge base embedded into the CPA through the Agent Training Module of Agent Academy.

Following a similar process to the one described above, the final knowledge structures for the SPA and the IPIA were generated.

4.3. The final recommendation

Fig. 10 illustrates the three parts of the final recommendation produced by IPRA for the order shown in Fig. 8. The *customer* tab recommendation (Fig. 10a) contains all information concerning the customer order, along with the suggested customer priority, discount and splitting of the order. Clicking on the *inventory* tab (Fig.10b), one can get information on item stock and supplementary items that IPRA recommends to be ordered. Information regarding possible suppliers (according to item stock and the re-order metric) for items that need to be ordered is shown on the third panel of the Recommendation Agent GUI, under the *suppliers* tab (Fig. 10c).

5. Summary and further work

This paper presented IPRA, a multi-agent system that manages customer services using data mining techniques. More specifically, the process of a new order, posted by a specific customer, was thoroughly described and focus was given on the developed methodologies, which were specified keeping in mind the effort for establishing efficient, quick and easy order processing. The presented MAS deploys techniques for creating ‘smart’ recommendations to its operator, who posts a request for a new order on behalf of a customer. The functional architecture of IRPA was described in detail, and all types of participating agents analyzed in terms of their functional specifications. Special focus was given on the explanation of our methodology for deploying intelligence into three of the development agent types (CPA, SPA and IPIA) by exploiting data mining techniques. The pattern recognition and fuzzy theory concepts, on which the developed methodology relied, as well as tools used for the implementation of IPRA were outlined. Theoretical aspects of our approach and details of the recommendation extraction mechanism deployed by our system, were discussed in detail. Finally, a test case was described in order to show the results of the application of IPRA into an enterprise environment with ERP data.

Preliminary results on IPRA efficiency have indicated that such an add-on could be easily incorporated into the philosophy of various ERP or ERP-like systems. Although

Table 9
The resulting customer clusters and their corresponding Discount and Priority values

Centre id	Population (%)	Discount (%)	Priority
0	0.002	20	High
1	10.150	10	Medium
2	46.600	15	Medium
3	22.240	10	Medium
4	20.830	5	Low

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE pmml_2_0.dtd (View Source for full doctype...)>
- <PMML>
  - <Header copyright="issel.ee.auth.gr" description="Clustering Model of ERP data">
      <Application name="Agent Academy Data Miner" version="0.3" />
    </Header>
  - <DataDictionary numberOfFields="9">
      <DataField name="AB" optype="continuous" />
      <DataField name="CL" optype="continuous" />
      <DataField name="TO" optype="continuous" />
      <DataField name="AOP" optype="continuous" />
      <DataField name="STDOP" optype="continuous" />
      <DataField name="AOI" optype="continuous" />
      <DataField name="STDOI" optype="continuous" />
      <DataField name="APT" optype="continuous" />
      <DataField name="STDPT" optype="continuous" />
    </DataDictionary>
  - <ClusteringModel modelName="ERP-
     org.agentacademy.modules.dataminer.filters.ReplaceMissingValuesFilter"
     modelClass="centerBased" numberOfClusters="5">
    + <MiningSchema>
      <ClusteringField field="AB" compareFunction="squaredEuclidean" />
      <ClusteringField field="CL" compareFunction="squaredEuclidean" />
      <ClusteringField field="TO" compareFunction="squaredEuclidean" />
      <ClusteringField field="AOP" compareFunction="squaredEuclidean" />
      <ClusteringField field="STDAOP" compareFunction="squaredEuclidean" />
      <ClusteringField field="AOI" compareFunction="squaredEuclidean" />
      <ClusteringField field="STDAOI" compareFunction="squaredEuclidean" />
      <ClusteringField field="APT" compareFunction="squaredEuclidean" />
      <ClusteringField field="STDAPT" compareFunction="squaredEuclidean" />
    - <Cluster name="0">
        <Array n="9">-1.825885405000001 17.3669590985 18.233535155 -0.5547026534999999
          -0.47161631449999997 7.9673528954999995 14.478504162 0.4015452435
          0.9260090569999999</Array>
      </Cluster>
    - <Cluster name="1">
        <Array n="9">0.065254309217391280 -0.2700838356000005 -0.12450477693913045 -
          0.6467945341130443 -0.7116394340086958 -0.352765316130435 -
          0.2181669595652179 -1.3250274658956527 -0.4930568987304352</Array>
      </Cluster>
    - <Cluster name="2">
        <Array n="9">0.037866447331439 0.042438334909091435 0.048315489856060635 -
          0.0805251161458331 -0.17864025739772726 0.16450308504166672
          0.09465143581818213 0.6366109786231056 0.006316590157197051</Array>
      </Cluster>
    - <Cluster name="3">
        <Array n="9">-0.06429684282539684 -0.04372558051587294 -0.12270015975793666 -
          0.6476984329999985 -0.713258177 -0.34270389580555505 -0.2148240383134917 -
          1.3290929909999991 -0.5051942009999977</Array>
      </Cluster>
    - <Cluster name="4">
        <Array n="9">-0.03238624846610169 -0.06382572756355923 -0.07092864160169489
          1.191644251190678 1.5120545920466106 0.10227567191101694
          0.0012368286016949195 0.6371860386101693 0.7577258649703401</Array>
      </Cluster>
    </ClusteringModel>
</PMML>
```


Fig. 9. The resulting PMML document for CPA.

the techniques employed are well-known, the innovation of IPRA is that it integrates Supply Chain Management with Customer Relationship Management primitives into one software tool, and therefore provides the ERP operator with the benefits of data mining technology and the versatility of a multi-agent platform.

Our experience from the application of IPRA in the enterprise environment with real ERP data, has shown that our methodology is highly dependent on the availability of existing data. As the number of available records increases, our system is directed to the production of even more specific recommendations. This aspect of IPRA indicates

(a)
(b)
(c)

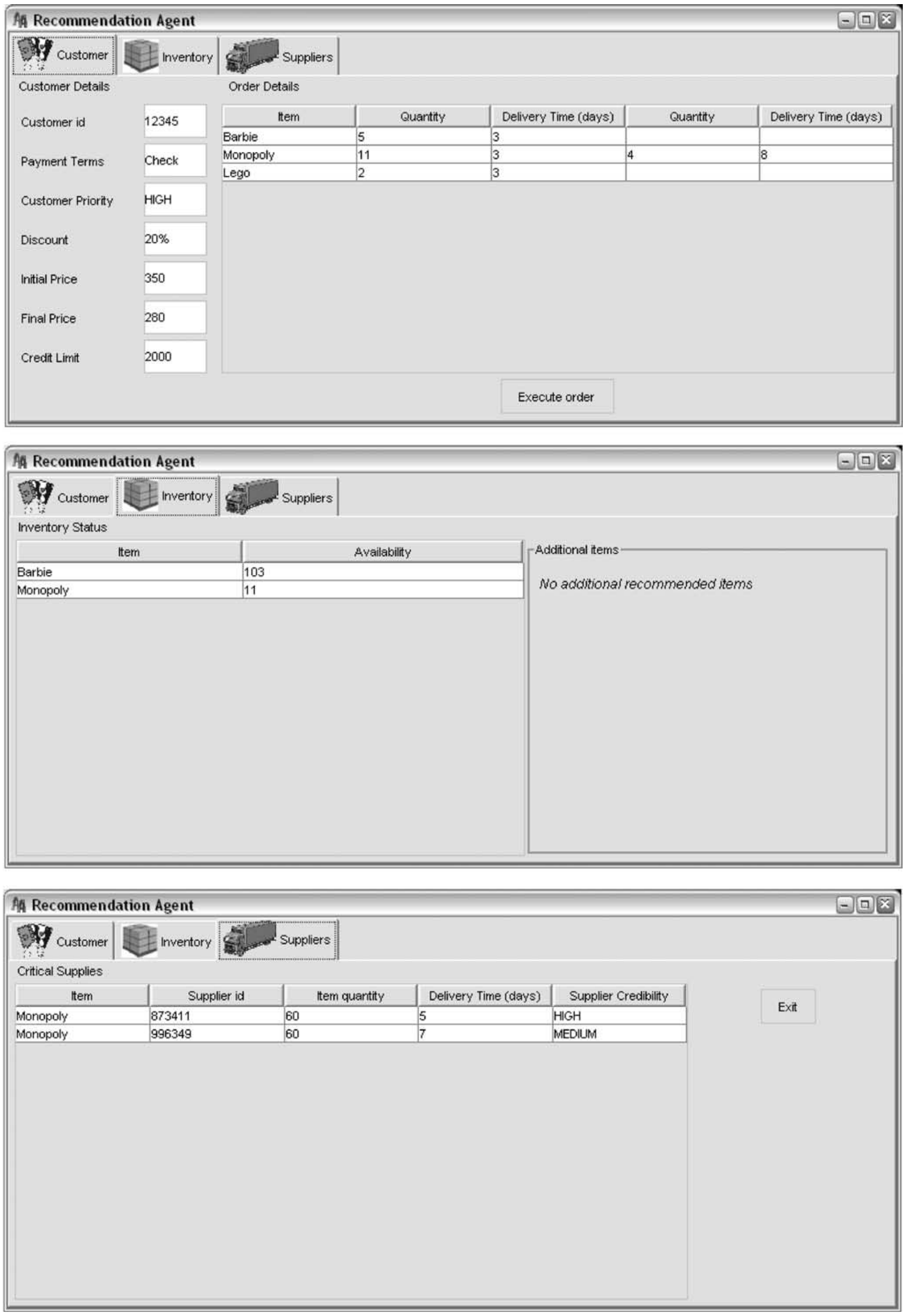

Fig. 10. The Customer (a), Inventory (b), and Supplier (c) tabs of the Recommendation Agent GUI.

an important property of our methodology; the ability to retrain agents and incorporate more complete knowledge bases into them. Considering the fact that ERP repositories increase with time, it becomes clear that IPRA provides an agent retraining facility, which is time-dependant. Another advantage of IPRA is that it has been developed with the use of Agent Academy, which allows easy redesign of the agent architecture, whereas it can recursively retrain the agents through the Data Miner.

Further work in the context of IPRA includes the development of an automated evaluation procedure for the data mining process, accompanied with tools for visualization. The presented multi-agent architecture can be expanded to fulfill the needs of a distributed network of existing ERP systems. This may be achieved mainly by introducing mobility characteristics to the existing agent types.

References

Agent Academy Consortium (2000). *The Agent Academy Project*. Available at: http://AgentAcademy.iti.gr

Amir, A., Feldman, R., & Kashi, R. (1997). A new and versatile method for association generation. *Information Systems*, *22*(6/7), 333–347.

Barbuceanu, M., & Fox, M. (1994). *The information agent: An infrastructure agent supporting collaborative enterprise architectures*. *Proceedings of the Third Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, IEEE Computer Society Press, pp. 112–117.

Bellifemine, F., Poggi, A., & Rimassa, G. (2000). Developing multi-agent systems with JADE. *In the Seventh International Workshop on Agent Theories, Architectures, and Languages (ATAL-2000), Boston, MA*, Available at: http://jade.cselt.it.

Brafman, R. I., & Tennenholtz, M. (1997). On the axiomatization of qualitative decision criteria. *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Ninth Innovative Applications of Artificial Intelligence Conference*, *1*, 76–81.

Chen, M. S., Han, J., & Yu, P. S. (1996). Data mining: An overview from a database perspective. *In IEEE Transactions on Knowledge and Data Engineering*, *8*(6), 866–883.

Choy, K. L., Lee, W. B., & Lo, V. (2002). Development of a case based intelligent customer–supplier relationship management system. *Expert Systems with Applications*, *23*(3), 281–297.

Choy, K. L., Lee, W. B., & Lo, V. (2003). Design of an intelligent supplier relationship management system: A hybrid case based neural network approach. *Expert Systems with Applications*, *24*(2), 225–237.

Data Mining Group (2001). *Predictive Model Markup Language Specifications (PMML)*, ver. 2.0. Available at: http://www.dmg.org

Delgado, M., Sanchez, D., Martin-Bautista, M. J., & Vila, M. A. (2001). Mining association rules with improved semantics in medical databases. *Artificial Intelligence in Medicine*, *21*(1–3).

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). Knowledge discovery and data mining: Towards a unifying framework. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Portland*, 82–88.

Ferber, J. (1999). *Multi-agent systems—An introduction to distributed artificial intelligence*. London: Addison-Wesley.

FIPA (2000). *Foundation for Intelligent Physical Agents Specifications*. Available at: http://www.fipa.org/

FIPA-SL Specifications (2000). *FIPA SL Content Language Specification*. Available at http://www.fipa.org/specs/fipa00008/SC00008I.html

Freitas, A. A. (1999). On rule interestingness measures. *Knowledge-Based Systems*, *12*(5–6), 309–315.

Friedman-Hill, E. J (2003). *Jess, The Expert System Shell for the Java Platform*, version 6.1. Sandia National Laboratories, CA. Available at: http://herzberg.ca.sandia.gov/jess/

Ganti, V., Gehrke, J., & Ramakrishman, R. (1999). Mining very large databases. *Computer Magazine*, *32*(8), 38–45.

Grosso, W. E., Eriksson, H., Fergerson, R. W., Gennari, J. H., Tu, S. W., & Musen, M. A. (1999). *Knowledge Modeling at the Millennium*. The Design and Evolution of Protege—2000. Available at: http://protege.stanford.edu

Han, J., & Kamber, M. (2001). *Data mining: Concepts and techniques*. Burnaby: Morgan Kaufmann.

Heikkila, J. (2002). From supply to demand chain management: Efficiency and customer satisfaction. *Journal of Operations Management*, *20*(6), 747–767.

Hong, T. P., Kuo, C. S., & Chi, S. C. (1999). Mining association rules from quantitative data. *Intelligent Data Analysis*, *3*(5), 363–376.

Jennings, N. R., Sycara, K., & Wooldridge, M. J. (1998). *A roadmap of agent research and development* (*1*). *Autonomous agents and multi-agent systems*, Boston: Kluwer Academic, pp. 7–38.

Kalles, D (1994). *Decision trees and domain knowledge in pattern recognition*. PhD Thesis, University of Manchester

Levi, S. D., Kaminsky, P., & Levi, S. E. (2000). *Designing and managing the supply chain*. Illinois: McGraw-Hill.

Looney, C. G. (1997). *Pattern recognition using neural networks: Theory and algorithms for engineers and scientists*. Oxford University Press.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley*, 281–297.

Mitkas, P., Symeonidis, A. L., Kechagias, D., & Athanasiadis, I. (2002). *An agent framework for dynamic agent retraining: Agent academy*. *In Challenges and achievements in e-business and e-work, Prague*, (pp. 757–764).

Patterson, K. A., Grimm, C. M., & Corsi, T. M. (2003). Adopting new technologies for supply chain management. *Transportation Research Part E: Logistics and Transportation Review*, *39*(2), 95–121.

Rygielski, C., Wang, J. C., & Yen, D. C. (2002). Data mining techniques for customer relationship management. *Technology in Society*, *24*(4), 483–502.

Shen, W., Xue, D., & Norrie, D. H. (1998). An agent-based manufacturing enterprise infrastructure for distributed integrated intelligent manufacturing systems. *Proceedings of the Third International Conference on the Practical Applications of Agents and Multi-agent Systems, London*, 533–548.

Symeonidis, A. L., Mitkas, P. A., & Kechagias, D. (2002). Mining patterns and rules for improving agent intelligence through an integrated multi-agent platform. *Sixth IASTED International Conference, Artificial Intelligence and Soft Computing ASC 2002, Banff, Alberta, Canada*.

Witten, I. H., & Frank, E. (1999). *Data mining: Practical machine learning tools and techniques with Java implementations*. New Zealand: Morgan Kaufmann.

Zeng, D., & Sycara, K. (1999). Agent-facilitated real-time flexible supply chain structuring. *Proceedings of Agents 99 Workshop on Agent Based Decision-Support for Managing the Internet-Enabled Supply-Chain, Washington*, 21–28.