Multi-agent technology and ontologies to support personalization in B2C E-Commerce

Multi-Agent Technology and Ontologies to Support Personalization in B2C E-Commerce

D. Rosaci and G.M.L. Sarné

{DIES, DICEAM}, Università "Mediterranea" di Reggio Calabria, Via Graziella Loc. Feo di Vito, 89060 Reggio Calabria (Italy)
E-mail: {domenico.rosaci,sarne}@unirc.it

Abstract

In this paper we present an XML-based multi-agent system, called *Multi Agent System for Traders* (MAST), that supports several Business-to-Customer e-Commerce activities, including advertisements and payments. MAST helps both customers and merchants in performing their tasks by using a personalized approach. Moreover, e-payments in MAST are implemented under the availability of financial institutions. This avoids to exchange sensible customers' information and reinforces the confidence between customers and merchants. A complete prototype of MAST has been implemented under the JADE framework, and it has been exploited for realizing some experiments, in order to evaluate its performances.

Keywords: e-Commerce, Business-to-Customer, Electronic payment, CBB

NOTICE

This is a post-print version of the paper. The final version is available at http://dx.doi.org/10.1016/j.elerap.2013.07.003

1. Introduction

Nowadays, e-Commerce (EC) plays a pivotal role in the Web, involving different aspects (i.e., technological, economic, legal, etc.) depending on the characteristics of the EC transactions (Palopoli et al., 2006).

In particular, EC transactions between a merchant and a customer are commonly denoted as Business-to-Customer (B2C) processes, that can be compared to the retail trade of traditional commerce. More specifically, B2C market involves a large number of merchants interested in offering products by using a convenient media and customers that desire to purchase those products. In this context, customers and merchants can exploit different opportunities (Zwass, 2003) as: (*i*) absence of time and space boundaries; (*ii*) simplicity, efficiency and comfortability of sales and purchases; (*i*) availability of low costs and several sale terms. However, a significant customer-merchant distrust still persists in EC, mostly due to the absence of personal contacts and to a low acceptance of the e-payment methods for security reasons.

A B2C transaction is a complex decision-making process consisting of different activities such as searching for a product, selecting a merchant, negotiating the best price and so on, that have to be carried out by both customers and merchants. In this context, a relevant attention has been devoted to identify the customer's behaviour and the complementary merchant's behaviour. Several studies have been proposed in the literature in order to model the different phases composing a B2C process. Some of them are derived by traditional retail commerce as the Nicosia (Nicosia, 1966) or the Engel and Blackwell (Engel et al., 1995) models, while others have been specifically designed for the Web as the Nissen's Commerce model (Nissen, 1997), the E-commerce Value Chain model (Feldman S., 1999) or related to Simon's decision making process, usually used in Decision Support Systems (Miles and Howes, 2000).

A widely adopted behavioural model is the Consumer Buying Behaviour (CBB) (Guttman et al., 1998) that is also exploited in this paper. The CBB is structured into six different phases, each one relative to a well defined activity, as briefly described below; (*i*) *Need Identification*, where a user identifies his/her needs; (*ii*) *Product Brokering*, in which the user searches for products that satisfy his needs; (*iii*) *Merchant Brokering*, dealing with the identification of a merchant selling the chosen goods or services; (*iv*) *Negotiation*, to fix the transaction terms (i.e. price, quantity, etc.); (*v*) *Purchase and Delivery*, where the customer finalizes the purchase choosing both payment and delivery modality; (*vi*) *Service and Evaluation*, that consists of the customer's evaluation of his satisfaction level about the performed purchase.

In this context, the multi-agent technology (Costina et al., 2011; Hector, 2005; Hubner et al., 2009; Maes, 1994; Nwana, 1996; Perini, 2007) appears as a promising solution for designing tools capable of supporting virtual community of users. It allows users to interact with the environment and carry out delegated tasks in simple, intelligent and independent manner in order to realize some kind of collaborative space (Nocera et al., 2011; Rosaci et al., 2012; Tsvetovatyy and Gini, 1996; Ye et al., 2001). Software agents have been fruitfully applied also in EC (He et al., 2003; Lax and Sarné, 2006; Liu and Ye, 2001; Rosaci and Sarnè, 2012a; Ursino et al., 2004) in order to design systems characterized by high levels of automation. Currently, only few agent systems cover more than one phase of a B2C process, while the most part of them provide only a rough and non-integrated support for a fixed typology of B2C activities. However, in developing such agent systems it is crucial that customers and merchants can be fully supported along all the tasks of a B2C process with a high automation level by attending them step by step, in a safe, reliable, and personalized way.

To this purpose, there is the need to obtain, maintain and update information about both customers' interests and preferences and merchant's trading data using suitable profiles (De Meo et al., 2007; Rosaci and Sarnè, 2012b) and in such activity the results obtained by using software agents appears more effective than those obtained by other approaches. The customers' profiles can be realized either on the merchant or the customer side; each of the two possibility implies a different representation of the interests and preferences. Indeed, in the first case only the activities performed by the customer on that merchant's EC site are representable; differently, using the second alternative, it is possible to obtain a complete representation of the whole customer's B2C history. Furthermore, an initial user's profile should be generated to solve the cold start issue (usually by exploiting direct elicitation methods). Finally, the presence of a network in the payment phase introduces some critical issues, absent in traditional payment systems, that requires the development of specific e-payment schemas in order to offer a trusted environment.

1.1. Contribution

To provide a solution for the aforementioned issues, the most important contribution of our research is that of proposing a mechanism to weight the importance of the different CBB activities from the customer's perspective. However, it is important to highlight that this contribution is not limited to only defining weights and coefficients of interests but, more important, we introduce a new method to allow a customer agent to assist its own user by using the aforementioned weights, obtaining a better effectiveness with respect to other approaches proposed in the past. Indeed, in our approach, the coefficients of interest of a customer for a product category, a product or a merchant are computed taking into account the weights above, and thus weighting the importance that the customer assigns to the different phases. The past approaches proposed in the literature computed similar coefficients of interest without discriminating the different CBB phases. In other words, if a customer shows an interest for a product in the "Need Identification phase", for those approaches this fact is considered equivalent to showing interest for that product in the "Merchant Brokering"phase. But for some customer, from the viewpoint of the personal interest, the act of showing interest for a marketing campaign about a product category could be considered less important than the decision, for instance, of searching for a suitable merchant in order to actually purchase that product. This observation leaded us to compute the coefficients of interests using different weights for the different CBB phases. The goal of our proposal is that of assisting the customers that use our approach in a more effective way than the classical approaches. In the next Section, we will describe some widely used measures to evaluate the system effectiveness, and the results of some experiments that we present in Section 7 clearly show that our approach outperforms other approaches proposed in the literature in terms of user's satisfaction.

As a second contribution, we propose a B2C framework based on the above approach, called *Multi-Agent System for Traders* (MAST). MAST is, to the best of our knowledge, the first proposal of a multi-agent system capable of assisting both customers and sellers of a B2C community in all the phases involved in B2C activities. In other words, a customer using MAST will be assisted by this tool in (i) determining the most important needs; (ii) finding the most appropriate products to satisfy those needs; (iii) selecting the most suitable merchants for purchasing the desired products; (iv) defining the details of the transaction with the merchant; (v) operating the payment. Moreover, MAST assists also the human merchant in the activities above, automatically sending to customers appropriate offers, responding to customers' requests, etc. Any multi-agent system has been proposed in the past to assist customers and merchants in such a way.

MAST is composed of a set of personal XML-based agents, associated with customers and merchants, and an agency that manages the whole system. In particular, in MAST each merchant and each customer is provided by a software agent, managing a personal profile automatically built on the customer's or on the merchant's side, able to take into account the competencies of the involved parties accordingly to all the performed B2C activities. The underlying CBB model provides MAST with a useful starting point and guideline to identify and suitably weight the different events composing a B2C process. We point out that the choice of the CBB model among all the other possible models existing in the literature is due to the fact the CBB model is so general to be considered as a generalization of all the other models, and it is the most widely applied model for B2C in the recent related work.

The MAST framework presents the following important features: (*i*) software agents adopt the eXtensible Markup Language "XML" (www.w3.org) to manage agent profiles and messages in a light and easy manner, to represent categories of interests and their instances belonging to various catalogues and to realize agent communications in ACML language (Grosof and Labrou, 2000) for guaranteeing portability and other benefits; (*ii*) an *Ontology* model (De Meo et al., 2012; Grosof and Labrou, 2000; Kumar, 2011) , used as a common language for all the agents, allows to give a unique representation of products and categories belonging to various catalogues; (*iii*) an e-payment protocol, called AIPP (Agent Internet Payment Protocol) (Garruzzo et al., 2006), based on existing financial institutions, fully compliant with the standard FAST (Financial Agent Secure Transaction, 2000) framework, it is used together with single-use account identifiers (Shamir, 2002) in order to perform safe and trusted payments; (*iv*) a "yellow page" service is available for all the agents.

1.2. Plan of the paper

The paper is organized as follows: A brief overview of our approach for supporting B2C activities is presented in Section 2. The MAST framework is described in details in Section 3. In Section 4 the AIPP protocol is briefly illustrated and in Section 5 the adopted functionalities for customer and merchant support are described. Section 6 deals with some Related Work. In Section 7, some experiments performed using a MAST prototype are discussed and finally, in Section 8, some conclusions are drawn.

2. Overview of the Approach

In this section, we briefly describe our approach for supporting B2C activities of customers and merchants by using a multi-agent system. The characteristics of autonomy and proactivity of software agents make them good candidates to act as personal assistants of the B2C actors, and similarly to several approaches presented in the past, we have conceived that each customer and each merchant is associated with a personal agent. However, differently from past approaches, we have decided to associate with each actor also an agent representing his financial institution, in order to exploit it for assisting the actor in the payment stage. Indeed, our purpose is that of giving assistance to customers and merchants in all the stages of a B2C process. Moreover, if such an assistance consists only in supporting the communication between the actors in Negotiation and in Purchase and Delivery stages, for the first three stages (Need Identification, Product Brokering, Merchant Brokering) the support consists in generating suggestions for the actors, helping them to take correct decisions.

As many other similar proposals, our approach is based on the construction of individual profiles of customers and merchants, where a profile is a collection of information representing the behaviour of the associated customer or merchant in performing B2C activities. This information is retrieved by examining the actions performed by customers and merchant in the different B2C stages as, for instance, selecting a product to purchase or selecting a merchant to begin a transaction. The core of such a profile consists of three coefficients, denoted by CW, PW and AW, that are present both in a customer and in a merchant profile and that represent the interest for a product category, for a product instance and for an agent, respectively. In other words, when a customer visits an EC site associated with the framework, as a main activity both the agents of the customer and of the seller start to monitor the customer's behaviour in the site, during the different stages of the B2C process. Moreover, each way a specific product category is involved, or a specific product is selected, or a specific agent is contacted from the customer or from the merchant, the associated CW, PW and AW values, belonging to the profiles of the customer and merchant agents, will be updated. In particular, for a customer agent, CW represents the global interest of its user in a product category, while PW denotes the interest for a specific product instance belonging to a product category and AW the interest about a merchant. Similarly, for a merchant agent, CW (resp., PW, AW) represents the interests of all its customers in a product category (resp., product instance) on the interest of a merchant in a given customer.

The coefficients CW, PW and AW are exploited in the first three stages of the B2C process, in such a way: (*i*) CW is used in the Need Identification stage by each customer agent to filter unwanted offers coming from merchants, and thus focusing the assistance activity only on those categories having the highest CW values; CW is also used, already in the Need Identification stage, by the merchant agent, to select the categories that are preferred by the customers of the associated merchant; (*ii*) PW is used in the Product Brokering stage by the customer agent to select the most interesting product to search for, and by the merchant agent to offer the products that have the highest possibility to result as interesting for the customers; (*iii*) AW is used in the Merchant Brokering stage by the customer agent to select the most suitable merchants to be contacted for purchasing a given product, and by the merchant agent to determine the most suitable customers to propose a given product.

Although the use of interest coefficients associated to assistant agents is not a new idea, however in our approach we propose a novel methodology to compute such coefficients. While most of the past proposals computed similar coefficients by observing the global behaviour of customers and merchants, without differentiating the importance of the various stages, our idea is that of allowing customers and merchants to give a different weight to each different stage when computing the coefficients. For instance, if a customer accepts to evaluate an offer dealing with a product category c and coming from a merchant in the Need Identification stage, its customer agent can update the coefficient CW related to c assigning a given weight to this update, while if the customer subsequently decides to actually purchase a product of the category c in the Product Brokering stage, the customer agent can assign a different weight to this action, that could be judged more representative of the interest that the customer shows for the category c.

Our supposition is that the suggestions generated by the customer agents using the above approach should appear as more satisfactory for customers with respect to those that would be performed if the coefficients CW, PW and AW were computed without weighting the actions in the different stages.

To evaluate the effectiveness of the suggestions generated by a customer agent, we can ask a set of customers to explicitly assign a score to those suggestions. More in particular, we assume that in each stage of the B2C process, each customer is provided with an ordered list of recommendations R, where the elements of R are product categories (resp. product, merchants) in the Need Identification (resp. Product Brokering, Merchant Brokering) stage. The elements of R are ordered based on the interest coefficient, i.e CW (resp. PW, AW) for the Need Identification (resp. Product Brokering, Merchant Brokering). More in particular, to provide the user with a intuitively understandable relevance measure, each element i of R is associated with an integer rate belonging to the set $[1, 2, 3, 4, 5]$. Such a rate of i, denoted by p_i, represents a prediction of the user's interest about the element i, and it is computed based on the interest rate ir_i of i, as follows: $p_i = 1$ if $0 < ir_i < 0.2$, $p_i = 2$ if $0.2 \leq ir_i < 0.4$, $p_i = 3$ if $0.4 \leq ir_i < 0.6$, $p_i = 4$ if $0.6 \leq ir_i < 0.8$, $p_i = 5$ if $0.8 \leq ir_i \leq 1$, where the interest coefficient is equal to CW (resp. PW, AW) for the Need Identification (resp. Product Brokering, Merchant Brokering).

In words, we have chosen to use the classical classification of the rates in the interval 1-5, and therefore we have partitioned in five classes the interval $[0, 1]$ to which the interest rate ir_i belongs.

The user, in an Evaluation phase is required to provide his rating of each recommended element i, denoted by r_i, as in the experiments that we have described in Section 7.

In the literature, three main categories of metrics have been proposed for evaluating the accuracy of a prediction algorithm, namely accuracy metrics, classification accuracy metrics, and rank accuracy metrics. (see Herlocker et al. (2004)). Predictive accuracy metrics measure how close the recommender systems predicted ratings are to the true user ratings. Predictive accuracy metrics are particularly important for evaluating tasks in which the predicting rating will be displayed to the user such as in the case of our experiment. To measure statistical accuracy it is usually proposed the mean absolute error (MAE) metric, defined as the average absolute difference between predicted ratings and actual ratings. Formally:

$$MAE_{=}\frac{\sum_{i=1}^{N}|p_i - r_i|}{N} \tag{1}$$

where N is the total numbers of recommendations generated for all the users involved in the evaluation test.

Classification accuracy metrics measure the frequency with which a recommender system makes correct or incorrect decisions about whether an item is good. We use Receiver Operating Characteristic (ROC) sensitivity to measure classification accuracy. The ROC model attempts to measure the extent to which an information filtering system can successfully distinguish between signal (relevance) and noise. The ROC curve represents a plot of recall (percentage of good recommendations returned), versus fallout (percentage of bad recommendations returned). We consider a recommendation good if the user gave it a rating of 4 or above, otherwise we consider the recommendation bad. We refer to this ROC sensitivity with threshold 4 as ROC-4. ROC sensitivity ranges from 0 to 1, where 1 is ideal and 0.5 is random. Since comparing multiple systems using ROC curves is tedious and subjective, we provide as a single summary performance number the area underneath a ROC curve, also known as Swets A measure, that can be used as a single metric of the systems ability to discriminate between good and bad recommendations. Moreover, to complete our analysis, we compute, besides MAE and ROC curve, also the Customer ROC (CROC) curve, another metric introduced in (Schein et al., 2005), and we use as synthetic evaluation parameter the area under the CROC curve.

Rank accuracy metrics measure the ability of a recommendation algorithm to yield a recommended ordering of items that matches how the user would have ordered the same items. It is important to point out that ranking metrics do not attempt to measure the ability of an algorithm to accurately predict the rating for a single item, therefore in our case, where we display to the user predicted rating values, it is important to additionally evaluate the system using a predictive accuracy metric as described above. We use the Normalized Distance-based Performance Measure (NDPM) as rank accuracy metric, that is computed as follows:

$$NDPM = \frac{2 \cdot C^{-} + C^{u}}{2 \cdot C^{i}} \tag{2}$$

where C^{-} is the number of contradictory preference relations between the system ranking and the user ranking. A contradictory preference relation happens when the system says that item 1 will be preferred to item 2, and the user ranking says the opposite. C^{u} is the number of compatible preference relations, where the user rates item 1 higher than item 2, but the system ranking has item 1 and item 2 at equal preference levels. C^{i} is the total number of preferred relationships in the users ranking (i.e. pairs of items rated by the user for which one is rated higher than the other). NPDM is a value ranging in [0.0..1.0], where 0.0 means best recommendations and 1.0 means worst recommendations. In our experiments, we have computed the average of NPDM on all the users.

In Section 7 we will use the metrics above to evaluate the effectiveness of our approach with respect to other approaches proposed in the past that do not give different weights to the different stages of the B2C process, showing the advantages introduced by our approach.

3. The MAST Framework

The MAST framework, represented in Fig. 1, has been conceived to completely support the B2C actors. In MAST each customer C (resp. merchant M) is associated with a personal agent c (resp. m) and with the agent of his financial institution (FI). Each agent has to be logged into the MAST Agency (Ag). All the personal agents support B2C activities managing (in terms of insertion, deletion and updating) their respective *Knowledge* profiles. In this section, a representation of the knowledge of personal agents and agency will be briefly described, while the B2C support activities in the CBB stage are exposed in Sect. 5. From hereafter, the terms product and service will be used in an interchangeable manner.

3.1. The MAST Knowledge Representation

To deal with products belonging to different categories, all the agents share a common *Ontology* ($\mathcal{O}$) representing each category of interest which products belong to. More in detail, $\mathcal{O}$ contains a set of *product categories*, each one represented by a pair $(code, d)$, where *code* is a code identifying the product category and d (*description*) is its textual description. The *Ontology* is implemented as an XML-Schema (www.w3.org/XML/Schema) defining a product category as an *element* and each its instances (i.e., product) as an *element instance*. In the current version, the ontology exploits the North America Industry Classification "NAICS" (www.census.gov/eos/www/naics)

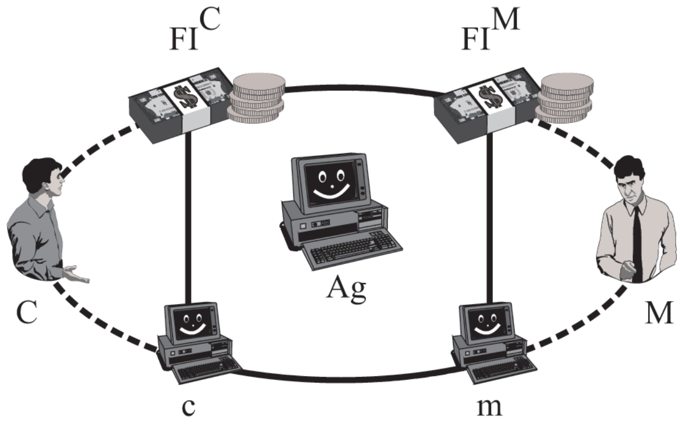

Figure 1: The MAST architecture

coding, a public classification of business area used in North America in the six digit format. Obviously other ontologies of this kind might as well be adopted instead of the NAICS coding.

3.2. The MAST Personal Agents

From hereafter, U will denote the generic user (a customer or a merchant) and a will represent his personal agent. Each MAST personal agent a manages its Agent Knowledge (AK) profile, represented in Fig. 2, and uses the information in the described structures to realize its goals, as explained in the following, excluding the CBB support which is presented in Sect. 5. More in detail:

- the *Working Data* (WD) collects the user's data (i.e., Name, City, etc.), the data of the financial institution (FI) associated with the user (i.e., the agent identifier with its address and the user's account number at FI), the agent password, the system parameters *Memory* (M), *Weight CBB Set* (K), *Need Identification Threshold* (NIT), *Product Brokering*

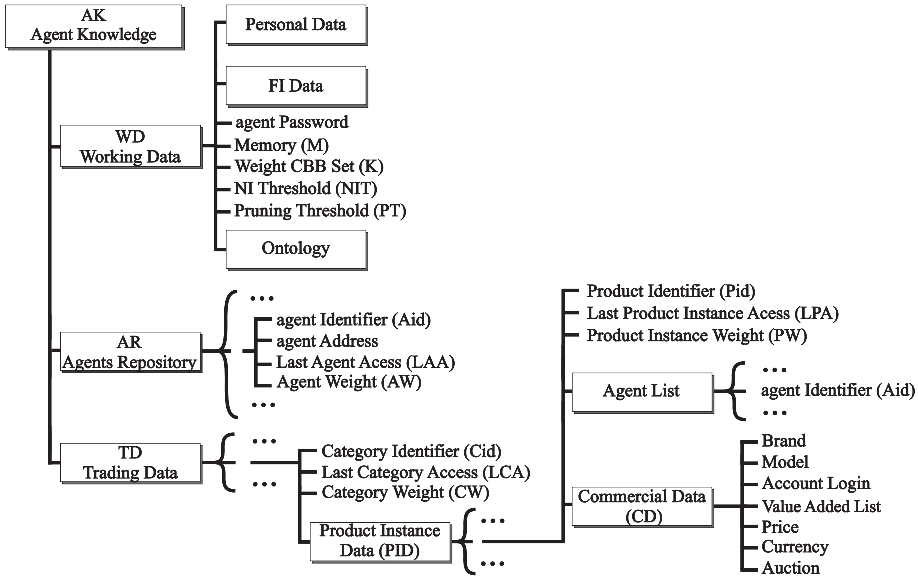

Figure 2: The Agent Knowledge (AK)

Threshold (PBT), *Merchant Brokering Threshold* (MBT) and *Pruning Threshold* (PT) that will be described in the following and, finally, the current ontology $\mathcal{O}$.

- in the *Agents Repository* (AR) each element is associated with a personal (customer's, merchant's or FI's) agent contacted by a during its activity) and stores the agent's identifier and address, the $LastAgentAccess$ (LAA) date and the $AgentWeight$ (AW) value, ranging in $[0;1]$, that measures the interest of a in that agent.

- the *Trading Data* (TD) is a set of data that an agent a obtains monitoring the CBB activities occurring in the MAST environment. For a customer its c agent collects the data of the products which the customer is interested in; elsewhere, the agent m, associated with a merchant M, collects the data of the CBB activities carried out on the site by the agents of the customers for the products offered on its site.

More in detail, each element represents a product category of interest for a customer (resp., offered by a merchant) consisting of: (i) a *Category Identifier* that is a category code belonging to the ontology $\mathcal{O}$; (ii) the *Last Category Access* (LCA) date, (iii) the *Category Weight* (CW) value, ranging in $[0; 1]$, that measures the interest in that product category; (v) the *Product Instance Data* (PID) storing some data for each product of interest for U.

In its turn, the *Product Instance Data* section stores for each product instance: an identifier; the *Last Product Instance Access* (LPA) date, the *Product Instance Weight* (PW) value, ranging in $[0; 1]$, that measures the interest in that product category; the *Agent List* (AL), a list that, for a customer agent, stores the identifiers of all the merchant agents offering that product instance (resp., for a merchant agent stores the identifiers of all the customer agents interested in that product instance); the *Commercial Data* (CD) (acquired by a customer's agent by exploiting the messages exchanged with the merchant's agents, that instead refers to its catalogue) consisting of some information useful to describe the product instance. In the current version the *Commercial Data* section includes information about *Brand*, *Model*, *Value Added List* (a list of possible benefits as coupons, gifts, etc.), *Price*, *Currency* and the *Auction* flag (to identify an auction process).

In the behaviour of the agent a, we identify the following two main steps (note that the trading support provided by MAST will be described in the next sections):

- ***setup steps***: some simple procedures are activated: (*i*) when a is first activated, its user sets in the agent profile both his data and parameters and FI's data and parameters; (*ii*) the user and the agency provide the agent affiliation to the framework;

- ***operative steps***: a customer or a merchant agent is automatically activated when a Web session starts and the EC site is on-line, and deactivated when a Web session ends or the EC site is off-line or for an explicit user's choice. In detail, an agent: (*i*) periodically sends its address to the agency in order to update the *Yellow Page* data structure managed therein; (*ii*) updates its profile after each access to an EC site, if the involved product category (i.e., product instances, agent) is absent in the agent profile it is added and the associated last acces date and weight is computed, otherwise if the product category (i.e., product instances, agent) is present in the agent profile such parameters are updated; (*iii*) filters the merchant offers based on the value of the NIT parameter (that specifies the number of categories for which the customer desires to consider merchants' offers and that are those having the highest CW values in the agent profile); (*iv*) dynamically and autonomously generates one-time account number (see below); (*v*) updates its data and parameters in its profile AK; (*vi*) periodically prunes its AK from any data evaluated as negligible on the basis of the CW and *Pruning Threshold* values.

In other words, when a customer visits an EC site associated with the framework, as a first activity both the agents of the customer and merchant update the respective AR lists. Then the two agents start to monitor the customer's behaviour through the site visit and to support the different activities identified by the behavioural model. Moreover, with respect to a specific product category and its instances, the associated CW, PW and AW values, belonging to the profiles of the customer and merchant agents, will be updated after each performed CBB activity. In particular, for a customer agent, CW represents the global interest of its user in a product category, while PW denotes the interest for a specific product instance belonging to a product category and AW the interest about a merchant. Similarly, for a merchant agent, CW (resp., PW) represents the interests of all its customers (resp., a given customer) in a product category (resp., product instance). While AW is the interest of a merchant for a specific customer. As in the Need Identification Stage the parameter NIT is used to propose to the customer only those product category having a CW coefficient greater than NIT, also in the Product Brokering (resp. Merchant Brokering) stage the parameter PBT (resp. MBT) is used to suggest to the customer only those products (resp. merchants) having a value greater than PBT (resp. MBT). Next, we show how CW (resp., PW, AW) is computed:

$$CW_{new} = \begin{cases} (1-M)\cdot\frac{CW_{old}}{1+ln(1+LCA-current\_date)} + M\cdot K_s, & \text{if } s\in[1,\cdots,5] \\ CW_{old} + M\cdot\phi & \text{if } s=6 \end{cases} \tag{3}$$

in which CW is computed on the basis of the whole customer's access history and s identifies the specific CBB activity performed. For each $s \in [1, \cdots, 5]$, CW is computed by giving more relevance to the more recent accesses having a high value of the parameter M, where M is a real value ranging in $]0;1[$, tuning the "memory" effect in CW, set based on the results of some tests appositely performed to this aim. Moreover, the natural log function seems a reasonable way to represent how CW decreases when the date of the last access becomes older, as remarked in (Brown and Lewandowsky, 2010), giving a reasonable priority to new items (Recuenco and Bueno, 2009).

The current access is weighted taking into account the performed CBB activity by using the parameter K_s that belongs to the *Weight CBB Set*. This is a set of five weights (one for each codified CBB activity) that are arbitrarily set by the user, in the range $[0;1]$, in order to weight the relevance that he assigns to a specific CBB activity. Note that the sum of the five K_s parameters has to be equal to 1 (i.e, $\sum_{s=1}^{5} K_s = 1$). Moreover, based on the temporal distance of the last access to the product category, expressed in day, the current CW value is decreased.

When the customer is unsatisfied about a purchase, he can carry out the last CBB activity of *Service and Evaluation* (i.e., $s = 6$) by setting the parameter ϕ in the range $[-1, 0[$, where $\phi = -1$ means the minimum degree of satisfaction. The parameter ϕ is weighted by means of M and added to CW in order to reset the contribution of that product instance in CW. Note that if $0 < \phi < -1$ and the updated CW assumes a negative value, ϕ will be set to 0, but if $\phi = -1$ also the CW weight will be set to -1 in order to mark that product category for avoiding in the future new purchases of

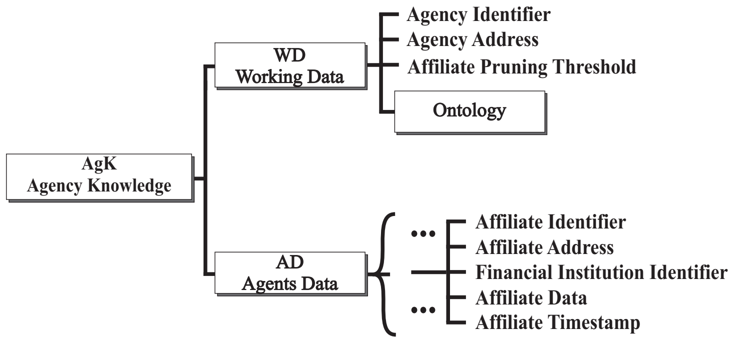

Figure 3: The Agency Knowledge (AgK)

product instances belonging to it. The value of the other weights PW and AW are computed similarly to CW.

3.3. The MAST agency

The MAST agency (*i*) manages (in terms of insertions, deletions and updating) its *Agency Knowledge* (AgK) profile, (*ii*) describes agents and FIs affiliations and deletions and (*iii*) provides them with some basic services.

The AgK (Fig. 3) consists of: (*i*) the *Working Data* (WD) including the agency identifier and its address, the *Affiliate Pruning Threshold* exploited to deallocate long-time inactive agents and the current *Ontology* $\mathcal{O}$; (*ii*) the *Affiliate List* (AfL), where for each affiliated to the platform are stored the agent identifier and address, the *FI Identifier* (set only when the affiliate is a customer or a merchant agent), the *Affiliate Data*, *Password* and a *Timestamp* of the last address update.

The main activities of the agency can be basically described as a two-steps process:

- ***managing steps***: the agency automatically carries out the following operations: (i) When the agency receives an *Affiliation Request* it registers the new member in its (AfD) list and replies with an agent identifier, an initial password and the current ontology. At this point the agent (resp., the FI) is logged in MAST; (ii) Following a data change notification in the affiliates' data, such as an address, the agency updates AfL; (ii) The agency deletes a member in response to a specific user's request or autonomously after an inactivity time greater than the *Affiliate Pruning Threshold* (to limit the potential growth of inactive members), and then informs the agent community about the agent deletion.

- ***service managing step***: the agency provides a yellow page service that, after receiving an agent or FI identifier, returns its last known address.

4. The Agent Internet Payment Protocol (AIPP)

Payment schemes can be evaluated using subjective criteria, as customer acceptance or trust (O'Mahony et al., 2001; Pasquet et al., 2008), and/or using objective criteria, by parameters like transaction cost, security, privacy, etc. Moreover, the presence of a network in a payment scheme introduces new issues, absent in traditional scenarios (Abrazhevich et al., 2009; Kuhne, 2012; Pasquet et al., 2008) , where: (i) identities of the transaction actors need to be authenticated and validated; (ii) payments and their effects guaranteed; (iii) operations, frauds and legal risks minimized. In addition, an extended use of standard protocols, existing products and services, payer anonymity, purchases confidentiality and low costs are desirable.

Currently, the most used e-payment system is the credit card, but in this case a credit card number should be provided to the merchant; this could be risky because the card number is provided over Internet and/or stored in the merchant site (Benson, 2009; Laleh and Abdollahi, 2009). Conversely, payment means as the electronic cash systems cannot be used due to law and crime prevention regulation/legislation (Merlonghi, 2010). Recently, centralized account schemes are quickly increasing in popularity for their aptitude to integrate usual financial instruments in a secure Internet transaction context.

This payment type, also proposed by well known financial institutions, includes general purpose or EC specific applications and can be realized completely either in secure software or in secure hardware.

In MAST, it is proposed the adoption of an electronic payment protocol, called AIPP (Agent Internet Payment Protocol) (Garruzzo et al., 2006), reinforced by means of the use of single-use account identifiers (Shamir, 2002). AIPP is complying with the simple and versatile Financial Agents Secure Transaction (FAST) “pre-negotiation” scheme (Financial Agent Secure Transaction, 2000) promoted by the Financial Service Technology Consortium “FSTC” (www.fstc.org), with other four different payment schemes for different scenarios but without to specify any detailed protocol. All the FAST payment schema are based on financial institutions managing user’s accounts and exploiting agent technologies to take advantage from existing infrastructures.

The main benefits introduced by AIPP are: (i) the customers and merchants without common authentication mechanisms (AIPP is not an authentication model) are reciprocally authenticated by their financial institutions when they log in their on-line accounts with the usual procedures, commonly with login and password over an SSL connection (wp.netscape.com/eng/ssl3); (*ii*) the payments occur directly via financial institutions to guarantee effective funds availability, fund transfer and connected effects, but also promote credit-push; (*iii*) the interoperability among accounts located in different financial institutions is easy to realize (as between two banks) choosing among different transfer modalities usually available. On the contrary, it is hard to implement when accounts are located into competitor payment systems; (*iv*) the payments are carried out by agents that replace customers and merchants in most uninteresting and/or complex tasks; (*v*) any hardware component is required. (*vi*) a significant low amount of information needs to be exchanged without any explicit encryption level. The risks in AIPP can be further minimized by transferring funds over interbanking networks, assigning to each message a time to live and a unique identifier, managing as much sensible information as possible off-line, etc. The problems of security communication among financial institutions, as those related to defense against viruses or hacker attacks, was beyond the AIPP and MAST project objectives.

Moreover, AIPP adopts only asynchronous agent communications without multiple Internet connections (other parties connected to the infrastructure, such as Internet providers, are considered as external risk factors). In this way, it is proposed as a simple potentially well acceptable Internet financial transaction method able to satisfy all issues of an e-payment scheme that have been previously described.

5. Support to CBB Activities

In this section, we illustrate the support provided by MAST to customers and merchants for all their activities involving in a B2C process represented accordingly to the CBB model. In MAST, to realize this task, several parties (i.e., merchant, customer with their respective agents and FIs) reciprocally interact with each other. Typical interaction between agents involves a customer (C) with his agent (c) and financial institution (FI^C), a merchant (M) with his agent (m) and financial institution (FI^M). Note that in this paper the financial institution typologies are limited to banks, card issuers or relevant financial organizations; further, it is assumed that payers and payees can manage their on-line accounts.

To assure multi-agent coordination and interoperability, MAST adopts the eXtensible Markup Language "XML" (www.w3.org) to overcome several heterogeneity problems (platforms, languages, applications and communication modalities) and to transfer business information in a consistent way (He et al., 2003). However, like in other domains, specific agreements have to be established on the semantics of XML tags in order to achieve interoperability. To this aim, in MAST a simple and yet compact XML specification has been designed with the only aim to transfer, in a consistent and efficient way, the needed business information according to the CBB model.

In MAST, to avoid possible attacks, single-use account identifiers (preserving also financial privacy), a nonce (i.e., an agent sender marker) and a Time To Live (TTL), used as message deadline for each agent communication, are adopted. Moreover, to promote trust among customers and merchants, the AIPP protocol allows the FIs to be third parties in a financial transaction, still guaranteeing user's privacy.

When two MAST agents interact between them some information are reciprocally exchanged. In the following, the messages notation and their data contents, used in MAST to transfer in a consistent and efficient way the business information, will be illustrated before of describing the MAST protocol. Note that the subscripts identify the message sender and receiver, while *data* is an XML document, whose content is context sensitive (see Table 1). More in detail:

- $INF_{x,y}(data)$: it requires/provides commercial information for a product;
- $REQ\_INV_{c,m}(data)$: it requires an invoice for a product offered by M;
- $INV_{m,c}(data)$: it contains the invoice required with $REQ\_INV_{c,m}(data)$;
- $PO_{c,m}(data)$: it is the purchase order with respect to $INV_{m,c}(data)$;
- $PE_{x,y}(data)$ (resp.,$PA_{x,y}(data)$): it notifies that the payment has been performed (resp., aborted) with respect to $PO_{c,m}(data)$;
- $MTO_{c,FI^C}(data)$: it is an irrevocable money transfer order with respect to $INV_{m,c}(data)$;
- $A\_MTO_{c,FI^C}(data)$ (resp., $R\_MTO_{c,FI^C}(data)$): it notifies the MTO acceptance (resp., rejection) with respect to $MTO_{c,FI^C}(data)$;
- $ACT\_COD_{x,y}(data)$: it contains the MTO activation code with respect to $INV_{m,c}(data)$;

- $NEW\_AI_{x,y}(data)$: it contains a new single-use account identifier to be employed in the next purchase or sell;
- $EVAL_{c,m}(data)$: it is an optional evaluation of a purchase.

A *data* XML document is structured in three sections including:

1. *H (Header)* that is composed by: the agent identifiers of *Sender* (S) and *Receiver* (R); the *CBB Stage* (s); a *Nonce* (nc); a *Time To Live* (TTL); a *Product Invoice Identifier* (PII).
2. *P (Products)* that encodes: a *Category Identifier Cid*, a *Product Identifier Pid* and all the associated data stored in the corresponding *Commercial Data* (CD) previously described in Sect. 3.2; the *Delivery Descriptor* (DD) corresponding to the avalaible delivery modality; the *Commercial Unit Required* (Cu); the *Final Price* (Fp).
3. *F (Financial)* is constituted by: the *Financial Institution Identifier* (FII); the *Financial Institution Address* (FIA); the *Financial Institution Single-Use Account identifier* (Ac); the *User Address* ($Address$).

The actions performed by agents in MAST to support customers and merchants in their B2C activities during all CBB stages are described below in detail for each CBB stage and represented in Fig. 4.

Need Identification Support. ($s = 1$) In the first CBB stage, customers identify their needs and merchants advertise their products to theit potential customers. In detail: (i) when a merchant M wants to offer a product to some potential customers, his agent m selects from its profile those agent that could be potentially interested in that product and sends $INF_{m,c}$ to them. m can also exploit the *Yellow Page* service provided by the *Agency*; (ii) after a c agent has received the merchants's offers, it will provide to present such offer to its customer only if it is fully compatible with his interests and preferences accordingly to the customer's parameter $NIT \in AK$.

Product Brokering Support. ($s = 2$) This stage occurs when a customer has identified a need and looks for a suitable product to satisfy it. In detail: (i) a customer can ask information on the desired product typology to one or more merchants by means of $INF_{c,m}$; (ii) all the merchants that have in their catalogue a product that matches with the customer's request, reply with a new $INF_{m,c}$ message with all the details of the product and commercial information.

Merchant Brokering Support. ($s = 3$) The actions performed in this stage are related to the identification of a suitable merchant to purchase a product, namely: (i) if the customer has a sufficient knowledge of the product details, a c's message $INF_{c,m}$ is sent to one or more merchants; (ii) if the merchant sells a product that matches the customer's request, the merchant replies with a message reporting a complete description of the product; in such a way the agent c can select the best product offer. Note that if in the previous stage a customer has received a sufficient number of $INF_{m,c}$ messages it is possible to choose a merchant without carrying out this present stage explicitly.

Negotiation Support. ($s = 4$) In this stage a customer and a merchant define the purchase details. They realize suitable strategies in a multi-round bid-offer message session. This stage is closed when either an agreement is

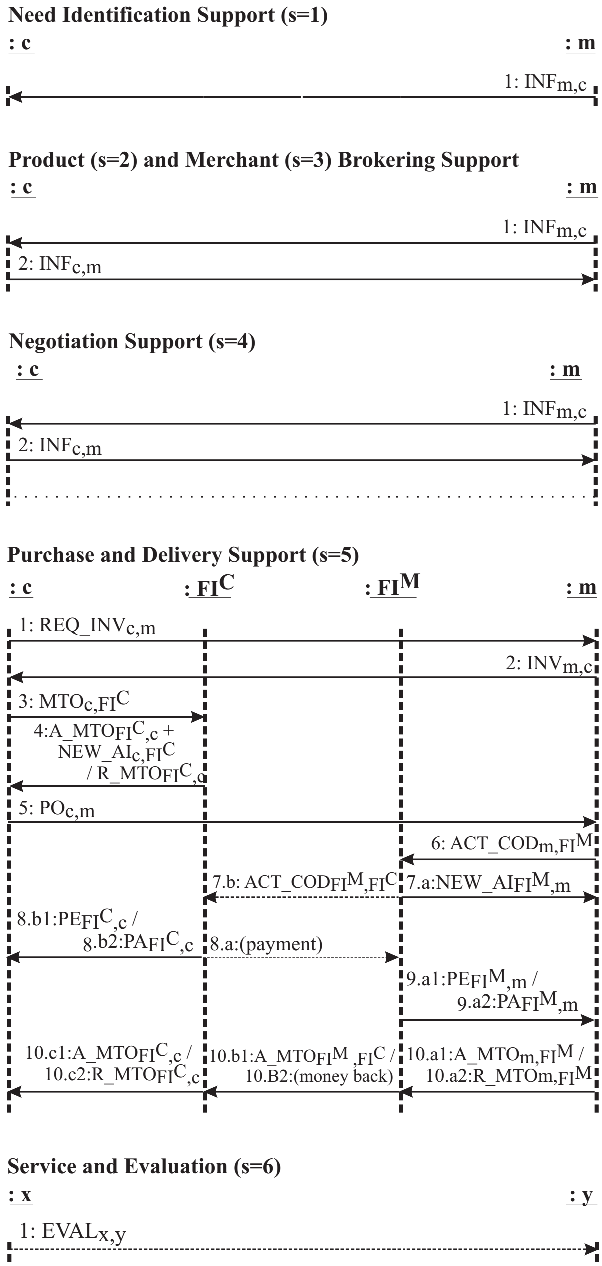

Figure 4: UML of the MAST support activities for the different CBB stages (s)

reached or the timeout TTL of the last message is reached.

Purchase and Delivery Support. ($s = 5$) The customer purchases, pays and chooses a delivery modality for a product offered by a merchant in this stage. The payment occurs by exploiting the AIPP protocol where: payer and payee identities are authenticated by the respective financial institutions during their on-line accounts accesses (usually with login and password over a *SSL Internet session*); payments occur directly among the financial institutions; single-use account identifiers are adopted; no heavy protocol is needed; no sensible financial and commercial information exchange happen for assuring privacy; financial institutions are third parties in the transaction to guarantee customers and merchants. More in detail, the actions performed in this stage are: (*i*) when a customer C wants to purchase a product offered by a merchant M, his agent c sends the message $REQ\_INV_{c,m}$ to the M's agent m; (*ii*) m replies to c with $INV_{m,c}$ (a pro-forma invoice); (*iii*) c logs into FI^C and performs a *Money Transfer Order* (MTO_{c,FI^C}) to the FI^M payee; (*iv*) FI^C accepts/rejects the MTO based on the presence of sufficient C's funds and then notifies to c its choice with a $A\_MTO_{FI^C,c}$ associated with a new single-use account identifier (*AcT*) for the next purchase or with a $R\_MTO_{FI^C,c}$ message; (*v*) c sends a $PO_{c,m}$ to m for confirming the purchase order; (*vi*) m logs into FI^M and sends the required payment activation code $H(INV_{m,c})$ to FI^M; (*vii*) FI^M provides M with a new single-use account identifier (*AcT*) for the next sell and sends to FI^C the payment code $H(INV_{m,c})$; (*viii*) if the activation code is the same as that provided by c, then FI^C effects the payment via FI^M and informs c about the state of success ($PE_{FI^C,c}$) or failure ($PA_{FII^C,c}$) of the MTO process; (*ix*) if the payment has been performed by FI^C, then FI^M informs m with a $PE_{FI^M,m}$ message otherwise, after the TTL of the $ACT\_COD_{FI^M,FI^C}$ message, FI^M informs m with a $PA_{FI^M,m}$ of the sell failure; (x) finally, m could however accept the payment informing FI^M (and consequently FI^C) or refuse it aborting the sale and returning back the money to FI^C by means of his FI^M. Finally, FI^C will inform c whether the product has been purchased or not.

Service and Evaluation. $(s = 6)$ It is an optional feedback provided by an agent to express his satisfaction about the purchase of a product, the counterpart or both. Two kinds of actions can be carried out to update the agent profile: (i) setting the parameter ϕ in the range $[-1, 0[$, where $\phi = -1$ means the maximum degree of unsitasfaction, in order to suitably update the CW, PW and AW interest weights. In particular, when ϕ is set to -1, the AW associated with the involved agent is set to -1 to mark it for avoiding future interactions with such an agent (i.e., merchant or customer); (ii) the unsatisfied agent could choose to inform its counterpart by using $EVAL_{x,y}$.

6. Related Work

The various aspects related to B2C commerce have been dealt with by using software agents in a large number of models and architectures proposed in a number of works. For such a reason, an overall contextualization of this paper within these backgrounds would require too much space and would be beyond our aims. Therefore, in this section, the examined approaches are those that, to the best of our knowledge, come closest to the material presented in this paper. The state-of-the-art has been investigated in a considerable number of surveys and the interested reader might refer to (He et al., 2003; He and Leung, 2003; Hubner et al., 2009; Maes, 1994; Palopoli et al., 2006; Perini, 2007; Sierra and Dignum, 2001; Ye et al., 2001) for a more complete overview. At the end of the section, differences and similarities among MAST and those discussed systems will be pointed out.

The most part of Multi-Agent Systems (MASs) supporting B2C activities within a CBB context are solely focused on the brokering and the negotiation stages. Only a restricted number of MASs cover a whole B2C process as codified into the CBB model. Note that many MASs where the CBB is not explicitly addressed to, their functionalities can, most often, easily brought back to it. Moreover, only an even more restricted number of MASs explicitly adopt an existent payment scheme, whereas the largest part of them ignore this issue and limit themselves to record that a payment has occurred. Finally, altough there exists a large variety of protocols and communication languages adopted by MASs in B2C, these will not be specifically addressed here.

MAS and CBB are tightly related since this behavioural model has been formalized in 1998 by Guttman et al. (Guttman et al., 1998) to provide their mediator agent of guidelines to assist users in the most suitable way. MAGMA (Tsvetovatyy and Gini, 1996) is an earlier marketplace antecedent to CBB definition but its activities can be easily described within it. MAGMA realizes an architecture for a partially automatized marketplace that supports message-based communication among agents (all agents communicate with each other through socket connections), allows different automated and human-controlled transactions, supports competitive and cooperative alliances. Different agents are delegated to perform advertising, negotiations (based on a Vickrey mechanism) and payments (a virtual bank provides financial services where users' accounts are managed by their respective agents).

WEBS (Web-based Electronic Brokering System) (Lau et al., 2000) consists in a set of brokering agents, each one specialized in providing a particular category of products and services (e.g., security, trading, books, software, etc.). In its turn, each brokering agent is associated with more sub-agents for dealing with various CBB tasks. In particular, for each customer the Need Identification and the Product Brokering tasks are handled by a *profiling sub-agent* that manages an internal profile to take into account the user preferences showed in the past with respect to a product domain. Sub-agents extract customer's behavioural rules by using probabilistic-logic formulas.

CASBA (Vetter and Pitsch, 2001), resulting from a CEE ESPRIT project, implements an Internet CBB and agent-based marketplace supporting all its stages in a flexible market mechanism with various auction types, dynamics negotiations and payments compatible with some existing payment schemas. CASBA exploits Java, JavaScript, CORBA and XML technologies, while the advertising is e-mail based. XML eases matching the data structures of the CASBA ontology with those of the client databases, and supports the negotiation in the associated CBB stage. In (Liu and Hwang, 2004) the authors propose a framework for heterogeneous agents able to handle various commerce protocols for the different commercial phases. An event-driven approach it is used to support users' agents with their processes, arranged in three main phases (each one structured in several sub-phases) and including existing payment schemas. Heterogeneity is studied in (Rosaci, 2005), covering all the CBB stages, with an ontology approach for assisting buyers and sellers in an unified manner that includes also the buyer coalition formation. A standard ontology helps heterogeneous agents to work in an e-marketplace built ad-hoc in order to represent both concepts involved in consumers and buyers interests as well as their behaviours in performing B2C activities. Agents assist users in translating their interests and preferences in ontologies having the standard format. The main limitation of this framework is its inability to automatically extract behavioural rules by monitoring users in performing their B2C activities. Therefore, these rules have to be explicitly specified by the agent owners. Furthermore, the homonyms and synonymies problems are not fully taken care of and there is not any specific payment mechanism.

In a context of mobile networks, Podobnik and Lovrek explore in (Podbnik and Lovrek, 2008) the opportunities provided by multi-agent systems and CBB model for promoting services. To support customers along all the CBB activities, the authors propose an agent architecture based on three agent typologies, namely, Consumer, Broker and Provider. In brief, the Consumer agent takes into account customer's preferences, suitably collected in a profile. The Broker agent interacts with the Consumer agent in order to execute the "Service" (i.e., "Product") and the "Provider" (i.e., "Merchant") brokering stages. The Provider agent supports the other two agents in all the interactions with the network services. Payments are supposed to be realized by exploiting the opportunities provided by the communication provider.

PumaMart (Wang et al., 2004) is an agent-based B2C marketplace that uses (*i*) a 2-phase fuzzy evaluation model with a parallel dispatch model and (*ii*) an auction-like "one-to-multiple" negotiation model. Following a process flow formed out of six stages (quite like the CBB model), a customer is monitored and supported in the B2C activities. In particular, a Java-enabled browser collects customer's information requests and preferences about a product instance. Such information are exploited by several agents types to search and filter amongst e-shops (including also commercial credit and security ranking criterions), to collect and evaluate offers (also based on price, warrantee services and delivery time), to negotiate and to pay. To provide a customer with fast response, some operations can be performed in parallel by using several agents (e.g. when a large number of e-shops has to be visited). Note that not all the aspects related to security, commercial credit and payment management have been solved. Another framework is proposed in (Liang and Huang, 2000) to coordinate EC activities, by organizing agents in three layers (market, transaction, activity). Market Maker agents (belonging to the market layer) coordinate tasks. Users'agents can exploit six different trading modalities along a B2C process arranged following the Simon's Decision Process model (Miles and Howes, 2000). Agent communications adopt KQML and payments are made by a dedicate agent (belonging to the activity layer), but no specific details thereof are provided.

The work presented in (Al-Shrouf et al., 2011) proposes an agent framework to provide a virtual shop for consumers and buyers. The online purchase of items through delegating requirements by using agent coordination and collaboration in a distributed computing environment. An agent controller provides robustness and scalability to the e-market place. Furthermore, multiple sellers can be registered on the platform, whereas buyers satisfy their requirements by using a mobile purchasing agent, which translates their requirements to the e-market place. In addition, the framework is customized to satisfy e-business transactions for buyers and sellers. Finally, a distribute open multi-tiered agent systems supporting the first CBB phases of B2C processes by means of recommendations has been recently presented in (Palopoli et al., 2012). This system, called DAREC, is characterized by a high computational efficiency and introduces significant advantages in terms of openess, privacy, security and allows new personalized terms to be introduced into the domain ontology. Moreover, we can observe that it is relatively easy to add in DAREC the support to the CBB activities actually do not covered in order to obtain a complete CBB support.

The main similarities between all the cited systems and MAST are that they (*i*) exploit the agent technology and (*ii*) store information about user's interests and preferences in an internal profile. Regarding the most relevant differences, we point out that each of the discussed systems adopts a different technique in the construction, managing and updating the customers' profiles. In this scenario, only MAST considers the relevance of the different activities in the construction of the users' profiles as illustrated below. In MAST, the activities are modelled by following the CBB model and weighting them according to how much they actually influence the product purchase. This peculiarity, in our opinion, allows to construct more precise profiles of the users' interests and preferences than the other mentioned systems. MAST exploits such profiles in all the performed support activities (i.e., in the recommendations generation and in the support of the B2C processes).

Moreover, we observe that (*i*) MAST and a restricted number of systems are XML-based; (*ii*) the payment issue is handled or considered in all of the cited systems but MAST, CASBA and (Liu and Hwang, 2004) support native commercial payment schema (in MAGMA a virtual bank acts as a mediator among different financial institutions and in (Podbnik and Lovrek, 2008) the communication provider effects payments as an added networking service, while in the other systems no specific details are provided); (*iii*) The systems above are not fully automatized and do not adopt an homogeneous approach to provide a B2C support to the user as MAST.

7. System Prototype and Experiments

We have implemented a prototype of the MAST framework under the JADE (jade.tilab.com) platform, to evaluate the advantages for customers and merchants by simulating CBB processes in a small B2C scenario. Furthermore, to perform such experiments some XML EC sites have been appositely realized.

A first campaign of experiments is devoted to measure the effectiveness of MAST when providing suggestions to the customers in the first three stages of the B2C process, namely Need Identification, Product Brokering and Merchant Brokering. In these experiments, the effectiveness of the recommendations generated by MAST has been compared with the systems EX-XAMAS (De Meo et al., 2007) and X-COMPASS (Garruzzo et al., 2002).

To realize this experiment, we have monitored the activities of 43 real customers while they performed sequences of partial and complete CBB processes on a set of 11 EC sites that we have built for this experiment. The sites represented merchants that offered products belonging to 14 different categories, for a total number of 2722 different products. Each product was

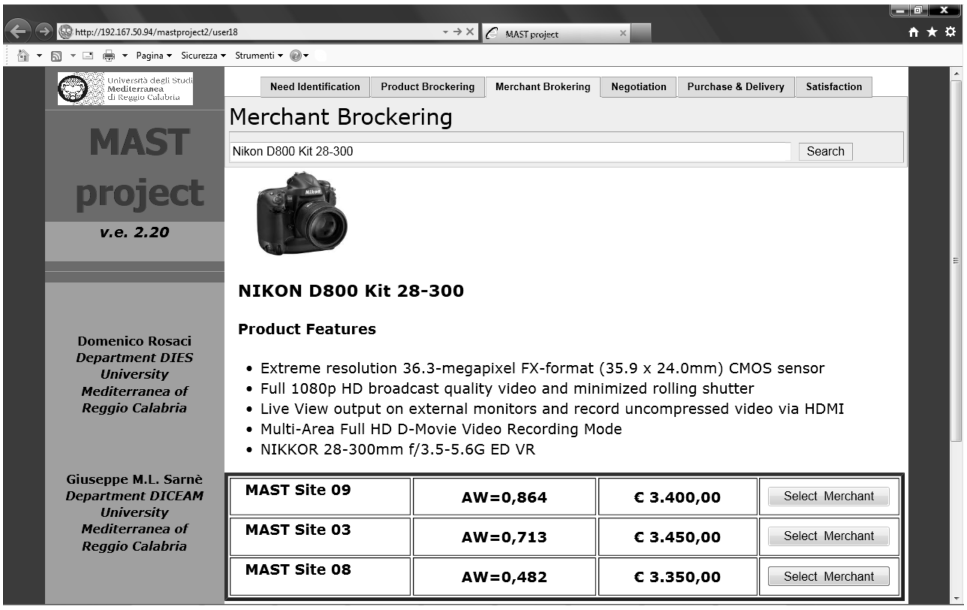

Figure 5: The MAST user interface interacting with the EC sites.

associated with a price, a payment method and a delivery time. Figure 5 shows the general aspect of the MAST user interface interacting with the EC sites. Each EC site was controlled by a software agent, capable of accepting incoming messages by customers and managing transactions. We have not implemented the Negotiation stage, assuming for simplicity that the products offered in the sites are proposed at fixed prices.

We asked the real customers to perform a period of *learning activity*, operating a given number of B2C actions (Need Identification, Product Brokering, Merchant Brokering) necessary to build the profiles of customer and merchant agents. Each customer performed about 200 different B2C actions in average.

Then, we have asked the customer to perform a set of other B2C actions, supported by MAST, X-COMPASS and EC-XAMAS that gave them suggestions in the first three stages of the B2C process. MAST computes the suggestions based on the coefficient CW (resp. PW, AW) for the Need Identification (resp. Product Brokering, Merchant Brokering), suggesting those product categories (resp. products, merchants) having a CW (resp. PW, AW) greater than the threshold NIT (resp. PBT, MBT). We have used a value $NIT = PBT = MBT = 0.5$ in our experiments, that appeared as a suitable threshold in the practical situations we have experimentally observed.

7.0.1. Results

Table 2 reports the average values of the measures MAE, Swet's A, area under the CROC curve and NPDM.

We can see that MAST presents an average MAE smaller than the other systems, and that MAST outperforms EC-XAMAS, that is the second best performer, with a 18 percent of advantage. The good quality of the MAST recommendations is confirmed by the analysis of the Swet's A measure related to the ROC-4 curve, where the advantage of using our recommender with respect to the second-best performer recommender (EC-XAMAS) is 14 percent. Analogous considerations can be done considering the area under the CROC curve, with an advantage of MAST with respect to X-Compass equal to 18 percent. The combined analysis of MAE and ROC shows that MAST performs significantly better than the other two systems both in predicting the rates of the users and in providing recommendations judged as good by the users. Finally, the analysis of the NPDM measure shows that also in this case, MAST is the system that presents the best performance, with an advantage of about 17 percent with respect to EC-XAMAS (which is the second best performer).

8. Conclusion

This paper describes MAST, an XML-based multi-agent system, fully implemented and tested, to support customers and merchants in an integrated and personalized way, taking into account their interests based on the behaviours shown during their B2C activities, represented as in CBB model.

MAST agents build, update and exploit users' profiles able to provide high quality representations of customers' orientations by suitably weighting the different activities performed in B2C processes arranged as in CBB model. More in detail, MAST: (*i*) considers the relevance of the different CBB activities to build high quality user profiles; (*ii*) exploits the advantages provided by XML in the formalization, representation and communication issues; (*iii*) adopts a secure centralized payment scheme based on existing Financial Institutions and one-time account numbers.

An experimental campaign has been carried out using a complete JADE-based prototypal MAST implementation focused on a small B2C scenario. The results have confirmed our expectations and the benefits of the proposed platform.

As for ongoing research, a development of MAST is planned by the introduction of different behavioural models taking in account emerging behaviours in the B2C area, such as formation of coalitions or the EC-site visiting. In fact, thanks to the richness of the customers' profiles MAST could easily integrate a recommender system in order to support users with personalized suggestions and/or user-adapted site presentation (De Meo et al., 2007; Rosaci and Sarnè, 2012b, 2013c).

References

Abrazhevich D., Markopoulos P., Rauterberg M., 2009. Designing Internet-Based Payment Systems: Guidelines and Empirical Basis. Human-Computer Interaction 24 (4), 408–443.

Al-Shrouf F. and Turani A. and Al-Shqeerat K., 2011. Software Agents for E-Commerce Data Workflow Management. In: Software Engineering and Computer Systems. Vol. 180 of CCIS. Springer, pp. 96–106.

Benson E.J., 2009. Analysis on Credit Card Fraud Detection Methods. In: Proc. of the Int. Conf. on Communication and Electrical Technology (IC-CCET 2011). IEEE Press, pp. 152–156.

Brown G.D.A., Lewandowsky S., 2010. Forgetting in Memory Models: Arguments Against Trace Decay and Consolidation Failure. Forgetting Psychology Press, pp. 49-75.

Costina B., Zoranb B., Hans-Dieterc B., Mirjanab I., 2011. Software Agents: Languages, Tools, Platforms. Computer Science and Information Systems 11 (8), 255–298.

De Meo P., Rosaci D., Sarne G.M.L., Terracina G., Ursino D., 2007. EC-XAMAS: Supporting E-Commerce Activities by an XML-based Adaptive Multi-Agent System. Applied Artificial Intelligence 21 (6), 529–562.

De Meo P., Quattrone G., Rosaci D., Ursino D., 2012. Bilateral Semantic Negotiation: A Decentralised Approach to Ontology Enrichment in Open Multi-agent Systems. International J. of Data Mining, Modelling and Management 4 (1), 1–38.

Engel J.F., Blackwell R.D., Miniard P.W., 1995. Consumer Behaviour. Int. ed. The Dryden Press, London, UK.

Feldman S., 1999. The Objects of the E-Commerce, Keynote speech at ACM 1999 . http://www.ibm.com/iac/oopsla99-sifkeynote.pdf.

Financial Services Technology Consortium (FSTC), 2000. Financial Agent Secure Transaction (FAST), Phase One Final Report (White Paper).

Garruzzo S., Modafferi S., Rosaci D. and Ursino D., 2002. X-Compass: An XML Agent for Supporting User Navigation on the Web. In: Flexible Query Answering Systems. Vol. 2522 of Lecture Notes in Computer Science. Springer, pp. 197–211.

Garruzzo S., Sarné G.M.L., Palopoli L., 2006. AIPP. A FAST-complied e-Payment Protocol. In: Proc. of the IADIS International Conference - Applied Computing (IADIS '06). IADIS Press, pp. 406–410.

Grosof B.N., Labrou Y., 2000. An Approach to Using XML and a Rule-Based Content Language with an Agent Communication Language. In: Issues in Agent Communication. Vol. 1916 of LNCS. Springer, pp. 96–117.

Guttman R.H., Moukas A., Maes P., 1998. Agents as Mediators in Electronic Commerce. Electronic Markets 8 (1).

He M., Jennings N.R., Leung H., 2003. On Agent-Mediated Electronic Commerce. IEEE Trans. Knowl. Data Eng. 15 (4), 985–1003.

He M., Leung H., 2003. Agents in Commerce: State of the Art. Knowl. and Information Sys. (4), 957–982.

Hector A., 2005. A New Classification Scheme for Software Agents. In: Proc. 3rd Int. Conf. Information Tech. and Appl.. Vol. 1. IEEE, pp. 191–196.

Herlocker J.L., Konstan J.A., Terveen L.G. and Riedl J.T., 2004. Evaluating Collaborative Filtering Recommender Systems. ACM Trans. Inf. Syst. *22,* 1, 5–53.

Hubner J.F., Bordini R.H., Picard G., 2009. Current Issues in Multi-Agent Systems Development. In: Engineering Societies in the Agents World VII. Vol. 5442 of LNCS. Springer, pp. 238–242.

Kuhne R., 2012. Charging and Billing in Modern Communications Networks A Comprehensive Survey of the State of the Art and Future Requirements. Communications Surveys & Tutorials 14 (1), 170–192.

Kumar M., 2011. Roles and Ontology for Agent Systems. Global Journal of Computer Science and Technology 11 (23), 39–46.

Laleh N., Abdollahi A.M., 2009. A Taxonomy of Frauds and Fraud Detection Techniques. In: Information Systems, Technology and Management. Vol. 31 of CCIS. Springer, pp. 256–267.

Lau R., Hofstede A., Bruza P., 2000. Adaptive Profiling Agents for Electronic Commerce. In: Proc. of the 4th CollECTeR Conf. on Electronic Commerce. Breckenridge, Colorado.

Lax G., Sarné G.M.L., 2006. CellTrust: a Reputation Model for C2C Commerce. Electronic Commerce Research 8 (4), 193–216.

Liang T.P., Huang J.S., 2000. A Framework for Applying Intelligent Agents to Support Electronic Trading. Decision Support Sysstems 28 (4), 305–317.

Liu D.R., Hwang T.F., 2004. An Agent-based Approach to Flexible Commerce in Intermediary-Centric Electronic Markets. J. Network and Computer Applications 27 (1), 33–48.

Liu J., Ye Y., 2001. Introduction to E-Commerce Agents: Marketplace Solutions, Security Issues, and Supply and Demand. In: E-Commerce Agents, Marketplace Solutions, Security Issues, and Supply and Demand. Vol. 2033 of LNCS. Springer, pp. 1–6.

Maes P., 194. Agents that Reduce Work and Information Overload. Commun. ACM 37 (7), 30–40.

Merlonghi G., 2010. Fighting Financial Crime in the Age of Electronic Money: Opportunities and Limitations. Journal of Money Laundering Control 13 (3), 202–214.

Miles G.E., Howes A., 2000. A Framwork to Understanding Human Factors in Web-base Electronic Commerce. Int. J. of Human-Computer Studies 52, 131–163.

Nicosia F., 1966. Consumer Decision Processes: Marketing and Advertising Implications. Prentice Hall, New York, NY, USA.

Nissen M.E., 1997. The Commerce Model for Electronic Redesign. J. of Internet Purchasing, 1 (2), http://www.arraydev.com/commerce/JIP/9702–01.htm.

Nocera A., De Meo P., Rosaci D., Ursino D., 2011. Recommendation of Reliable Users, Social Networks and High-quality Resources in a Social Internetworking System. AI Communications 24 (1), 31–50.

Nwana H.S., 1996. Software Agents: An Overview. Know. Eng. Review 11 (3), 11–40.

O'Mahony D., Pierce M., Tewari H., 2001. Electronic Payment Systems for E-Commerce, 2nd Ed. Artech House, Norwood, MA USA.

Palopoli L., Rosaci D., Ursino D., 2006. Agents' Roles in B2C e-Commerce. AI Communications 19 (2), 95–126.

Palopoli L., Rosaci D., Sarné, G.M.L., 2012. A Multi-tiered Recommender System Architecture for Supporting e-Commerce. In: Intelligent Distributed Computing VI. Vol. 446 of Studies in Computational Intelligence. Springer, pp. 71–81.

Pasquet M., Vernois S., Aubry W., Cuozzo F., 2008. Electronic Payment. In: Electronic payments, Encyclopedia of Information Science and Technology, 2nd ed., chap. 212. IDEA, p. 13411348.

Perini A., 2007. Agent-Oriented Software Engineering. John Wiley & sons, pp. 1–39.

Podbnik V., Lovrek I., 2008. Multi-agent System for Automation of B2C process in the Future Internet. In: Proc. of the IEEE Conf. on Computer Comm. Workshops (INFOCOM 2008). IEEE, pp. 1–4.

Recuenco J.G., Bueno D., 2009. Balanced Recommenders: A Hybrid Approach to Improve and Extend the Functionality of Traditional Recommenders. In: Proc. of Int. Work. on Adaptation and Personalization for Web. Vol. 2, pp. 88–97.

Rosaci D., 2005. Exploiting Agent Ontologies in B2C Virtual Marketplaces. J. UCS 11 (6), 1011–1039.

Rosaci D., Sarnè G.M.L., 2006. MASHA: A Multi-Agent System Handling User and Device Adaptivity of Web Sites. User Modeling User-Adaptivity Interaction 16 (5), 435–462.

Rosaci D., Sarnè G.M.L., Garruzzo S., 2012. Integrating Trust Measures in Multiagent Systems. International Journal of Intelligent Systems 27 (1), 1–15.

Rosaci D., Sarnè G.M.L., 2012a. A Multi-agent Recommender System for Supporting Device Adaptivity in e-Commerce. J. Intelligent Information Systems 38 (2), 393–418.

Rosaci D., Sarnè G.M.L., 2012b. A Multi-Agent Recommender System for Supporting Device Adaptivity in e-Commerce. Journal of Intelligent Information System 38 (2), 393–418.

Rosaci D. and Sarnè G.M.L., 2013. Recommending Multimedia Web Services in a Multi-Device Environment. Information Systems 38 (2), 198–211.

Schein A.I., Popescul A., Ungar L.H. and Pennock D.M., 2005. CROC: A New Evaluation Criterion for Recommender Systems. Electronic Commerce Research *5,* 1, 51–74.

Shamir A., 2002. SecureClick: A Web Payment System with Disposable Credit Card Numbers. In: Proc. of the 5th Int. Conf. on Financial Cryptography (FC 2001), Proc.. Vol. 2339 of LNCS. Springer, pp. 223–233.

Sierra C., Dignum F., 2001. Agent-Mediated Electronic Commerce: Scientific and Technological Roadmap. In: Agent Mediated Electr. Com., The Europ. AgentLink Perspective. Vol. 1991 of LNCS. Springer, pp. 1–18.

Tsvetovatyy M.B., Gini M., 1996. Toward a Virtual Marketplace: Architectures and Strategies. In: Proc. 1st Int. Conf. on the Practical Application of Intelligent Agents and Multi-Agent Technology, pp. 597–613.

Ursino D., Rosaci D., Sarne G.M.L., Terracina G., 2004. An Agent-based Approach for Managing e-Commerce Activities. International Journal of Intelligent Systems 19 (5), 385–416.

Vetter M., Pitsch S., 2001. Towards a Flexible Trading Process over the Internet. In: Agent Mediated Electronic Commerce, The Europ. AgentLink Perspective. Vol. 1991 of LNCS. Springer, pp. 148–162.

Wang Y., Tan K.L., Ren J., 2004. PumaMart: a Parallel and Autonomous Agents based Internet Marketplace. Electronic Commerce Research and Applications 3, 294–310.

Ye Y., Liu J., Moukas A., 2001. Agents in Electronic Commerce. Electronic Com. Res. 1 (1/2), 9–14.

Zwass V., 2003. Electronic Commerce and Organizational Innovation: Aspects and Opportunities. Int. J. Electron. Com. 7 (3), 7–37.

Table 1: Message Specification

Message	**Message Content**
$INF_{x,y}$	$\mathbf{H}(S^x, R^x, s, nc^x, TTL^x), \mathbf{P}(Cid, Pid, CD, DD, Cu, Fp)$
$REQ\_INV_{c,m}$	$\mathbf{H}(S^c, R^c, s, nc^c, TTL^c), \mathbf{P}(Cid, Pid^M, CD, DI^M, Cu, Fp)$
$INV_{m,c}$	$\mathbf{H}(S^m, R^m, s, nc^m, TTL^m, PII^m), \mathbf{G}(Cid, Pid^M, CD,$ $DI^M, DD, Cu, Fp), \mathbf{F}(FII^M, FIA^M, AcT^M)$
$PO_{c,m}$	$\mathbf{H}(S^c, R^c, s, nc^c, TTL^c, PII^m),$ $\mathbf{F}(FII^C, FIA^C, AcT^C, Address^C)$
$PE_{x,y}$	$\mathbf{H}(S^x, R^x, s, nc^x, TTL^x, PII^m)$
$PA_{x,y}$	$\mathbf{H}(S^x, R^x, s, nc^x, TTL^x, PII^m)$
MTO_{c,FI^C}	$\mathbf{H}(S^c, R^c, s, nc^c, TTL^c, PII^m),$ $\mathbf{F}(FII^M, FIA^M, AcT^M, H(INV_{m,c}))$
$A\_MTO_{FI^C,c}$	$\mathbf{H}(S^{FI^C}, R^{FI^C}, s, nc^{FI^C}, TTL^{FI^C}, PII^m)$
$R\_MTO_{FI^C,c}$	$\mathbf{H}(S^{FI^C}, R^{FI^C}, s, nc^{FI^C}, TTL^{FI^C}, PII^m)$
$ACT\_COD_{x,y}$	$\mathbf{H}(S^x, R^x, s, nc^x, TTL^x, PII^m), \mathbf{F}(H(INV_{m,c}))$
$NEW\_AI_{x,y}$	$\mathbf{H}(S^x, R^x, s, nc^x, TTL^x), \mathbf{F}(AcT^y)$
$EVAL_{x,y}$	$\mathbf{H}(S^x, R^x, s, nc^x, TTL^x, PII^y)$

.

In the first three CBB stages the messages can be addressed to c agents chosen among those stored in an agent profile AK employing the Ag's broadcasting messages service

Table 2: Performances of MAST, EC-XAMAS and X-Compass in terms of effectiveness

	MAE	Swet's A	CROC	NPDM
MAST	0.97	0.78	0.73	0.24
EC-XAMAS	1.19	0.68	0.60	0.29
X-Compass	1.21	0.61	0.62	0.31