Intelligent Information Management in Mobile Electronic Commerce

Vagan Terziyan

Information Technology Research Institute, University of Jyvaskyla, P.O.Box 35, FIN-40351 Jyvaskyla, Finland, e-mail: vagan@cs.jyu.fi

Oleksandra Vitko

Department of Artificial Intelligence, Kharkov State Technical University of Radioelectronics, Lenin Avenue 14, 61166 Kharkov, Ukraine, e-mail: alexandra_vitko@yahoo.com

Abstract

A mobile phone today is rapidly evolving into Personal Trusted Device with the ability to handle a wide variety of new services and applications including mobile electronic commerce. Managing and sailing the Web content to mobile users based on their preferences and location are becoming now a leading trend in mobile commerce. M-commerce services should be able to integrate different types of information with geographical data and manage different type of profiles related to preferences, security, privacy, transaction management, etc. The multilevel profiling framework discussed in this paper should be able to allow business operations based on different levels of profiles to be able to reach consensus between a buyer and a seller about general and specific features of their relationships. We consider the following main levels of profiles from the top level to basic level: international and local laws, standards (transaction management, payment, security, privacy, and so on); metaprofiles (Meta Document Type Definitions); profiles - Document Type Definitions (for security, for privacy, for preferences, for transaction management, for negotiations, for orders and invoices, for shipping schedules, for contracting and billing, and so on), E-Speak Service Framework Specification, Common Business Library patterns; interpreted profiles (concrete XML documents, which describe different features of reached consensus). Some hints to measure distances (evaluate similarity) between different profiles are also given.

1. Introduction

1.1. Mobile electronic commerce

The recent cellular phones have a lot of extra possibilities compared to the older ones. This way the telephone companies can offer a lot of so called value-added services. Percentage of users for this kind of service is expected to become more due to the fact that with new technologies new (and better) services can be offered. In the future several new services are expected. The ability of paying by phone is a service in which a lot of companies are interested. Since the wireless network will be more data oriented Internet will become a serious possibility as well. Because it is small, secure, personal, familiar and carried at all times, the mobile phone is rapidly evolving into much more than a wireless telephone. It is transforming into Personal Trusted Device with the ability to handle a wide variety of new services and applications, such as banking, payments, ticketing, and secure access-based operations [26]. Advances in wireless network technology and the continuously increasing number of users of hand held terminals make the latter a possible channel for offering personalized services to mobile users and give space to the rapid development of Mobile Electronic Commerce (m-commerce). M-commerce operates partially in a different environment than Internet E-Commerce due to the special characteristics of mobile terminals and wireless networks and the context, situations and circumstances that people use their hand-held terminals. Ericsson, Motorolla and Nokia, as key facilitators of the mobile Internet and the mobile information society, together have established the MeT initiative [26], the first major cooperative effort to provide common open-core technology for the m-commerce market. MeT is targeted to define a common framework for m-commerce applications by extending existing industry standards.

1.2. Managing and sailing the Web content

In [2] intelligent integration techniques for heterogeneous data sources and knowledge sources were considered, and the way of their implementation to problems in electronic commerce, knowledge management and information retrieval. A prerequisite for successfully integrating various information sources is a standardized and an access device friendly description of their semantics.

In [9] the Ontology Inference Layer (OIL) that is being proposed as a description language for ontology interchange (i.e. static information) and also UPML language has been developed for describing reasoning components. Integrating these two description types is concluded as a necessary step toward a knowledge web.

The next-generation Web i.e. "Semantic Web", enables intelligent services such as information brokers, search agents, and information filters, which offers greater functionality than the current services. In [7] the role of ontologies is studied in the architecture of the Semantic Web. It was shown that the use of XML as a tool for semantic interoperability would be ineffective in the long run. Further layer is needed [14] on top of the Web's current layers, which can be a general method for encoding ontology languages into RDF/RDF Schema.

Future of agent-mediated electronic business is discussed in [13]. It was concluded that agent-to-agent e-commerce transactions might soon dominate the global economy. Mobile commerce supported by personalized agents is not too far away from realization. Such optimistic views to the use of agents in electronic commerce one can find in many papers, for example in [19] where mobile agents are considered as able to participate auction's trading and contracting while their user is disconnected.

2. Important Features of Location Based Services

2.1. General requirements

Combining positional mechanisms with information about location of various objects can develop very powerful and flexible personal information services [21]. Suppose there is some geographical area that contains a certain number of objects (points of interests [10]). Each point of interest is assumed to have its virtual representation or, rather, a source of relevant to it information. A user of this information is expected to be mobile. The aim of the location-aware service is providing a user with information about the objects taking into account spatial relationships between him and the objects. One of the main input parameters is user's location. It is obvious that system should have information about all objects with their spatial location and links to their information sources. If system has this information, it is able to find the near objects. Note that for mobile objects system has to update periodically location information via location service or request it directly from the objects. First could be done automatically if we have an access to location service. In the second case, the user can input his location by himself such as a street address or a name of a region. After that, the service is able to provide a geographical description of his surroundings. These data act an auxiliary role of a navigator or a guide in order to connect real objects with their virtual representations. The next client's function is providing customers with information from sources associated with surrounding objects. The user can either directly receive information from the chosen object or charge the server to find needed information. In the last case, the server performs all work, analyses information sources according to user's directives and sends to him results of his query.

2.2. The use of XML

We assume information sources, i.e. points of interests, are XML-enabled, they can provide information in the XML format. This is implied that the system can define semantic meaning of the information and use it for analysis. The system should not be restricted by any particular XML-format. The use of one format is impossible because of variety of object types that can be used in the service. An information provider can use any XML format for its data. He must provide information about logical structure of this format and semantic meaning of tags, otherwise the system will not be able to extract the needed information. We make an integration of new content providers into our service as painless as possible. We should perform transformation rules for every information provider. Then a standard XSLT processor can be used for extracting data from XML format [30].

3. Transaction management

Transactions in mobile environment are used to encapsulate operations and provide Atomicity, Consistency, Isolation and Durability. For mobile e-commerce it means the following as described in [22]: *Atomicity* - transaction is *atomic* if either all operations necessary for preserving e-commerce atomicity are executed or all executed operations will become compensated. With money atomic protocols, funds are transferred from one party to another without the possibility of the money remaining in the middle of the money transfer infrastructure or alteration in value. Goods-atomic protocols are such that a good is received if and only if the money is transferred. Certified delivery protocols allow both a merchant and a customer to prove exactly which goods were delivered. *Consistency* - transactions must preserve consistency at various levels. For instance, a customer should not be allowed to draw funds from an account if this would result into a negative balance. *Isolation* - various steps of a transaction do not interfere with steps of other transactions. *Durability* - once a transaction completes its execution, its results become permanent even in the presence of failures. *Anonymity* - customer's desire not to disclose his identity.

3.1. Basic requirements for the agent-based transaction management

Intelligent Agents can be used in combination with WAP as an extension. Intelligent Agent Technology provides solutions for taking actions out of the user's hand. In a mobile environment this could mean decreasing the network traffic.

Some basic requirements for the agent-based transaction management model can be formulated as follows:

- The more equally transaction management duties are distributed among all intelligent agents within the mobile communication environment (MCE) the better management of the whole transactional process can be achieved.
- The distribution of intelligent functions on management of transactions for every pair "mobile client - server" should be done in accordance with functionality of an appropriate mobile device. More responsible transactions should be committed to more intelligent devices.
- All possible transactions should be distributed between the intelligent agents and each agent becomes an autonomous running machine for his transactions.
- Every intelligent agent should manage transactions, which effect to certain part of MCE that can be observed by this agent. Agent makes analysis of observed part of MCE, selects appropriate transaction to run, and changes the observed part of MCE according to transactional outcome.

Observed parts of MCE for different agents can overlap. It means that different agents can run transactions that pretend to change the same part of MCE. To manage this process there is a need of agents who can observe active transactions of another agents and resolve conflicts. Such observation is similar to one mentioned above if we assume that transactions are also part of MCE. Appropriate agents are also similar to previous ones because they observe active transactions (i.e. appropriate part of MCE), select appropriate meta-transactions to run and make appropriate changes within the observed transactions (make some transactions passive and some active).

If transactional part of MCE is very complex itself then it might result to a need of more agents who can run meta-meta-transactions that makes changes in meta-transactions. This process of responsibility distributions in theory can result to several levels of agents for management of transactions in very complex MCE.

3.2. Typical transactions in location based mobile services

In Figure 1 the typical simple cycle of transactions is shown, which serves the request of mobile user to get some information based on his location.

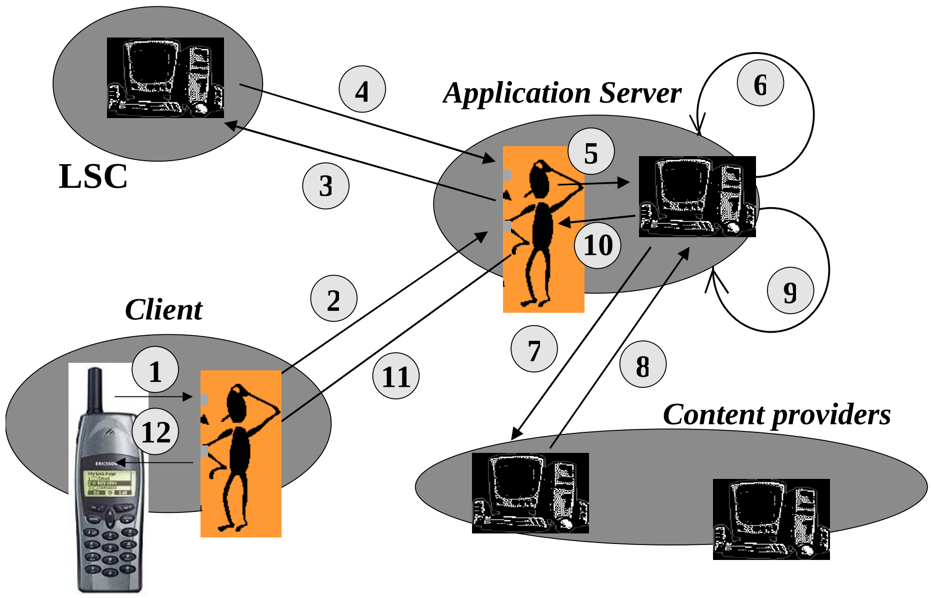

Figure 1. Simple cycle of transactions for location based application.

Transactions in Figure 1 can be principally described as follows:

1. A user starts up appropriate client software agent from the mobile device, authorizes him for the appropriate type of service and specifies request to the agent;
2. The client agent creates connection with appropriate server agent and forwards user's request in appropriate form to him adding information about requested format for output report;
3. The server agent connects with Location Service Center with the request to locate the client's position;
4. Location Service Center returns to the server agent the coordinates of the client;
5. The server agent connects with the application server and submits users request completed also with discovered info about user preferences and location;
6. The application server analyses request, classifies it according to its knowledge about content providers in the context of user's preferences and location, and finally selects appropriate contend provider to access for requested data;
7. The application server connects with selected content provider…;
8. … and picks up requested data;
9. The application server processes downloaded data to necessary format ...;
10. … and delivers the final report to the server agent;
11. The server agent delivers the requested report to client agent;
12. The client agent displays the report at the mobile device monitor.

3.3. Logical model of transactions

3.3.1. Constants

There are three sets of constants in the model:

$\mathrm{A}:\{A_1, A_2, ..., A_n\}$ - the names (id-s) of objects from the MCE;
$\mathrm{L}:\{L_1, L_2, ..., L_m\}$ - the names (id-s) of properties or relations between objects from the MCE;
$\mathrm{R}:\{R_1, R_2, ..., R_p\}$ - the names (id-s) of transactions, which change relations or properties within the MCE.

Example of set A: A_1 *- Bill Smith;* A_2 *- Nokia 9110-0503611426;* A_3 *- M#007;* A_4 *- M-Commerce-2000;* A_5 *- Kiesland;* A_6 *- UJ-Jytko,* A_7 *- 31524/2309;* A_8 *- Curriculum of a Mobile Computing Master program in the nearest university;* A_9 *- Mobile devices: 8ECTS, …, Wireless Internet: 4ECTS;* A_{10} *- S#011.*

Example of set L (*properties):* L_1 - to be a customer; L_2 - to be a mobile device (client), L_3 - to be an agent, L_4 - to be an application server, L_5 - to be a Location Service Center, L_6 - to be a content provider, L_7 - to be a location (coordinates), L_8 - to be a request, L_9 - to be a content.

Example of set L (*relations):* L_{10} - to wait, L_{11} - to be located within, L_{12} - to know.

Example of set R: R_1 - to send; R_2 - to receive, R_3 - to process, R_4 - to confirm, R_5 - to get confirmation.

3.3.2. Relations and Properties

P - basic relational predicate defines an *atomic statement* (denoted by logical variable P_i) of the model:

$$P_i = P(\mathrm{A}', L_k, \mathrm{A}'') = \begin{cases} \textit{true}, & \text{if } \mathrm{A}'\cap\mathrm{A}''=\varnothing \text{ and there is a relation } L_k \in \mathrm{L} \\ & \text{between each object } A_i \in \mathrm{A}'\subset \mathrm{A} \text{ and each } A_j \in \mathrm{A}''\subset \mathrm{A}; \\ \textit{true}, & \text{if } \mathrm{A}'\equiv \mathrm{A}'' \text{and each object } A_i \in \mathrm{A}'\subset \mathrm{A} \text{ has a property named by } L_k \in \mathrm{L}; \\ \textit{false}, & \text{otherwise.} \end{cases} \tag{1}$$

If an atomic statement is effected by operation of logical negation we will call it as negative. The negation (logical "not") operation will be denoted by " ¯ " above an appropriate argument.

Interpreted atomic statement is atomic statement of a type $P(A_i, L_k, A_j)$, in which all variables are exactly defined.

Examples of positive and negative atomic statements:

$P_1 = P(\mathrm{A}^1, L_3, \mathrm{A}^1)$ - "An agent"; $P_2 = \overline{P}(\mathrm{A}, L_{11}, A_4)$ - "Nobody is located within M-Commerce-2000".

Examples of interpreted atomic statements:

$P_3 = P(A_1, L_1, A_1)$ - "Bill Smith is a customer"; $P_4 = P(A_3, L_{11}, A_4)$ - "*M#007* is located within *Nokia 9110-0503611426*".

(Interpreted) Statement of the model is a logical function, which consists of one or more (interpreted) atomic statements connected with Boolean operations:

- Negation;
- Disjunction (logical "or"), which will be denoted by " $\vee$ " between appropriate arguments;
- Conjunction (logical "and"), which will be denoted by "•" between appropriate arguments.

Examples of statements:

$P_5 = P(\mathrm{A}^1, L_3, \mathrm{A}^1) \cdot (P(\mathrm{A}^1, L_{11}, \mathrm{A}^2) \cdot P(\mathrm{A}^2, L_4, \mathrm{A}^2) \vee P(\mathrm{A}^1, L_{11}, \mathrm{A}^3) \cdot P(\mathrm{A}^3, L_2, \mathrm{A}^3))$ - "An agent is located within an application server or within a mobile device";
$P_6 = \overline{P}(\mathrm{A}, L_{12}, \mathrm{A}^4) \cdot P(\mathrm{A}^4, L_7, \mathrm{A}^4) \cdot P(\mathrm{A}^5, L_{11}, \mathrm{A}^4) \cdot P(\mathrm{A}^5, L_2, \mathrm{A}^5)$ - "Nobody knows location of a client".

Example of an interpreted statement:

$P_7 = P(A_{10}, L_{11}, A_4) \cdot P(A_{10}, L_3, A_{10}) \cdot P(A_4, L_4, A_4) \cdot \overline{P}(A_{10}, L_{12}, A_7) \cdot P(A_7, L_7, A_7) \cdot P(A_2, L_{11}, A_7) \cdot P(A_2, L_2, A_2)$
- "Agent *S#011* located within application server *M-Commerce-2000* does not know that coordinates of client *Nokia 9110-0503611426* are *31524/2309*".

An *environmental statement* $P_i^E(t)$ is a conjunction of interpreted atomic (positive or negative) statements, which describes all relations and properties within the *i-th* observed part of MCE at current moment of time *t*.

An *MCE statement* $P^E(t)$ is a conjunction of interpreted atomic (positive or negative) statements, which describes all relations and properties within the whole MCE at current moment of time *t*:

$$P^E(t) = \prod_i P_i^E(t). \tag{2}$$

3.3.3. Transactions

T - basic transactional predicate defines an *atomic transactional statement* (denoted by logical variable T_q) of the model:

$$T_q = T(P_i, R_k, P_j) = \begin{cases} \textit{true}, & \text{if at the beginning of the transaction named } R_k \ (R_k \in \mathrm{R}) \text{ the statement } P_i \text{ should be } \textit{true} \\ & \text{and just after finishing the transaction } R_k \text{ the statement } P_j \text{ becomes } \textit{true}; \\ \textit{false}, & \text{otherwise,} \end{cases} \tag{3}$$

where $P_i = P_i^{'} \cdot P_i^{''}$, P_j is atomic (positive or negative) statement and $P_j = \overline{P}_i^{'}$.

Interpreted atomic transactional statement is an atomic transactional statement of a type $T(P_i, R_k, P_j)$, in which P_i and P_j are interpreted statements.

Example of atomic transactional statement: $T_1 = T(P_8, R_1, P_9)$, where:
$P_8 = P(\mathrm{A}^1, L_{12}, \mathrm{A}^2) \cdot P(\mathrm{A}^2, L_9, \mathrm{A}^2) \cdot P(\mathrm{A}^1, L_3, \mathrm{A}^1) \cdot P(\mathrm{A}^3, L_3, \mathrm{A}^3) \cdot \overline{P}(\mathrm{A}^3, L_{12}, \mathrm{A}^2)$ and

$P_9 = P(\mathrm{A}^3, L_{12}, \mathrm{A}^2)$ - "If an agent knows some content and another agent does not know it, then it is possible to run transaction "to send" after which also the second agent will know the content".

(Interpreted) transactional statement of the model is a logical function, which consists of one or more (interpreted) atomic transactional statements connected with Boolean operations: negation, disjunction and conjunction.

An *environmental transactional statement* $T_i^E(t)$ is a conjunction of interpreted atomic (positive or negative) transactional statements, which describes all currently valid transactions within the *i-th* observed part of MCE at the current moment of time *t*.

An *MCE transactional statement* $P^E(t)$ is a conjunction of interpreted atomic (positive or negative) transactional statements, which describes all currently valid transactions within the whole MCE at the current moment of time *t*:

$$T^E(t) = \prod_i T_i^E(t) \,. \tag{4}$$

3.3.4. Running rules for transactions

If $T_q = T(P_i, R_k, P_j)$ is an atomic transactional statement and $P_r^E(t) = P_r^{'} \cdot \overline{P}_j$ is an appropriate *environmental statement,* then the running rule for transaction R_k is as follows: $(T_q \cdot P_i \cdot P_r^E = true) \xrightarrow{R_k} P_r^E = P_r^{'} \cdot P_j$.

4. Profiling

Businesses assign profiles to customers that describe their needs, requirements, and interests [4]. These profiles are then used to target the audience of particular products. When one visits an e-commerce web site, he provides them with some profile information when registers. Each time while a user making a purchase, his profile may be updated with the specific content that has been purchased. Some adaptive sites even have the ability to track users clicks through its content to determine user's interests. All this information is combined to create users personal profile. The profile is then used to target certain products for a user. XML allows marking up the data in such a way that the context of that data is described in the markup language itself. Additionally, attributes allow us to provide meta-data regarding that context. The target audience for the information is a type of meta-data and can be captured in attributes on each element. By profiling the information according to its target audience, we can match the information profile to the audience profile in order to deliver the information that best meets the needs, requirements, and interests of customers.

The architecture discussed below should be able to allow operations based on different levels of profiles to be able to reach consensus between a buyer and a seller about general and specific features of their relationships.

4.1. Profiles of different level

We consider the following main levels of profiles from the top level to basic level (Figure 2):

- International and local laws, standards (transaction management, payment, security, privacy, and so on);
- Metaprofiles (Meta DTDs - Meta Document Type Definitions);
- Profiles (for security, for privacy, for preferences, for transaction management, for negotiations, for orders and invoices, for shipping schedules, for contracting and billing, and so on) - DTDs (Document Type Definitions), E-Speak Service Framework Specification [25], Common Business Library patterns [1, 6], etc.;
- Interpreted profiles (concrete XML documents, which describe different features of reached consensus).

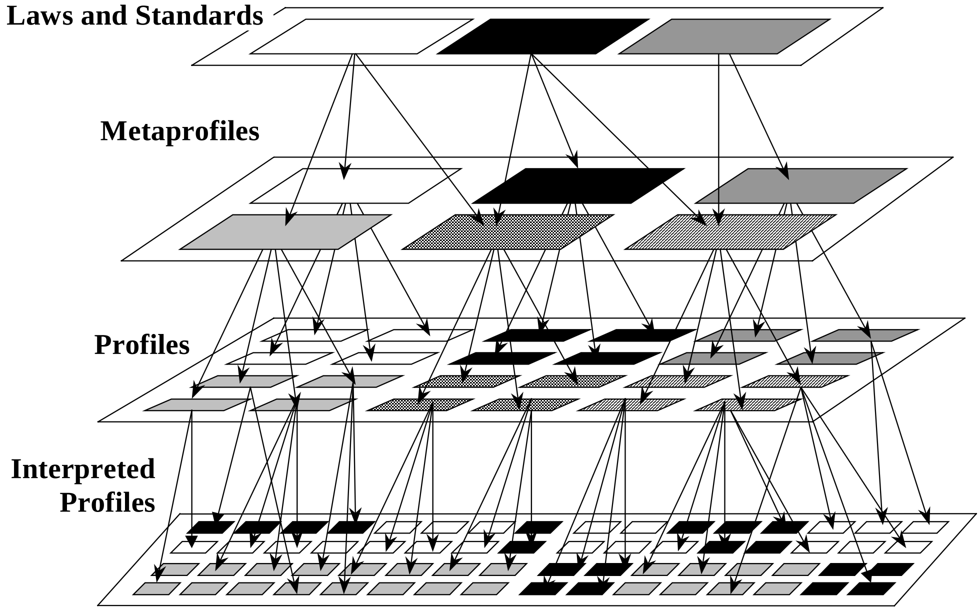

Figure 2. Multilevel structure of profiles

The advantage of using XML documents as profiles (interfaces in distributed business environment) instead of CORBA (Common Object Request Broker Architecture) objects is that they are easier to develop and use for a wider range of users, and that they represent naturally the documents, whether paper or EDI (Electronic Data Interchange) [1].

Document Type Definition (DTD) is some kind of frame by specifying slots of it one can get some piece of concrete knowledge in XML form [29]. A meta-DTD [3] is the set of declarations, which defines architecture of DTDs. The Common Business Library (CBL) [1, 6] is being developed by Veo Systems, Inc. as a set of building blocks with common semantics and syntax to ensure interoperability among XML applications. CBL models can be also considered as profiles of certain level, which fit well the general architecture of m-commerce.

4.2. Modeling profile structure by semantic metanetwork

Semantic Metanetwork [23, 24] is considered formally as the set of semantic networks, which are put on each other in such a way that links of every previous semantic network are in the same time nodes of the next network (see Figure 3).

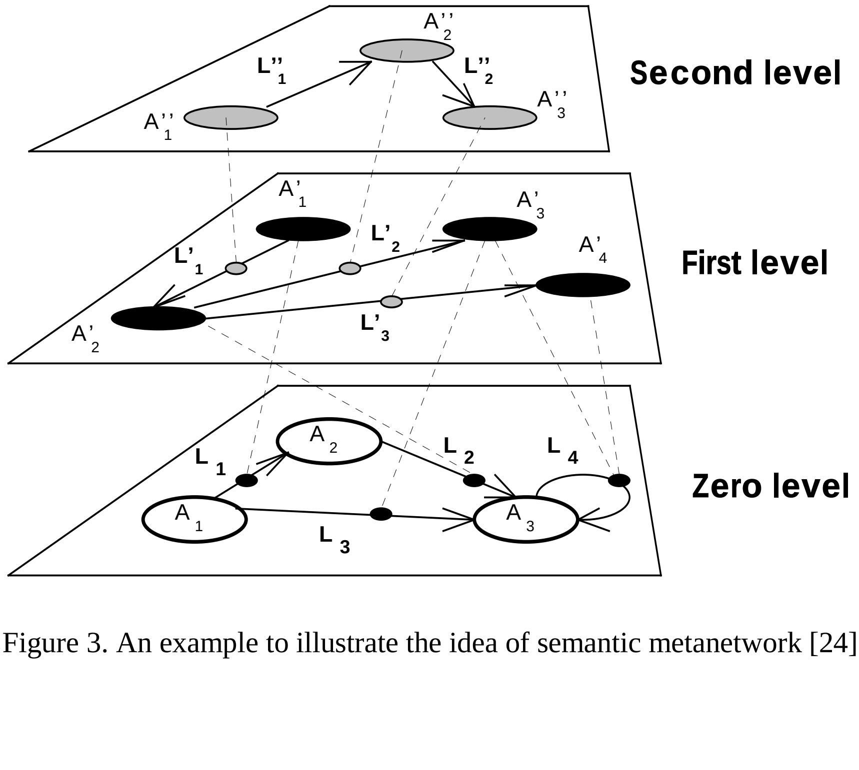

Figure 3. An example to illustrate the idea of semantic metanetwork [24]

In a Semantic Metanetwork every higher level controls semantic structure of the lower level. Simple controlling rules might be, for example, in what contexts a certain link of a semantic structure can exist and in what context it should be deleted from the semantic structure [24]. Thus such multilevel network can be used in an adaptive control system which structure is automatically changed following changes in a context of the environment. Multilevel representation of a context allows *reasoning with contexts* [24] towards solution of the following problems:

- to derive knowledge interpreted using all known levels of its context;
- to derive unknown knowledge when interpretation of it in some context and the context itself are known;
- to derive unknown knowledge about a context when it is known how the knowledge is interpreted in this context;
- to transform knowledge from one context to another;
- to derive trends within any problem considering it in several contexts, from the most fuzzy context to the most concrete one, and to use such trends to derive more precise solutions for the problem.

The above abilities of a Semantic Metanetwork make it very suitable to model multilevel structure of profiles in m-commerce.

4.3. Modeling profiles-scenarios by metapetrinets

High level Petri nets are a suitable formal method to design communication protocols because of their ability to express concurrency, non-determinism and system concepts at different levels of abstraction [12]. Colored Petri nets are used already to model and generate the possible primitive sequences of the wireless request/response transactional service [12]. In [11] wireless transaction protocol was considered as part of WAP architecture and was analyzed by colored Petri nets.

The idea of semantic metanetwork has been developed for facilitating flexible modeling and ***control of complicated dynamic processes*** using metapetrinets [23, 20]. *A **metapetrinet*** is able not only to change the marking of a petrinet but also to reconfigure dynamically its structure. Each level of the new structure is an ordinary petrinet of some traditional type. A basic level petrinet simulates the process of some application. The second level, i.e. the metapetrinet, is used to simulate and help controlling the configuration change at the basic level. There is conformity between the places of the second level structure and places or transitions of the basic level structure. One possible control rule is such that a certain place or transition is removed from the present configuration of the basic level if the corresponding place at the metalevel becomes empty. If at least one token appears to an empty metalevel place, then the originally defined corresponding basic level place or transition immediately is created back to the configuration (see Figure 4).

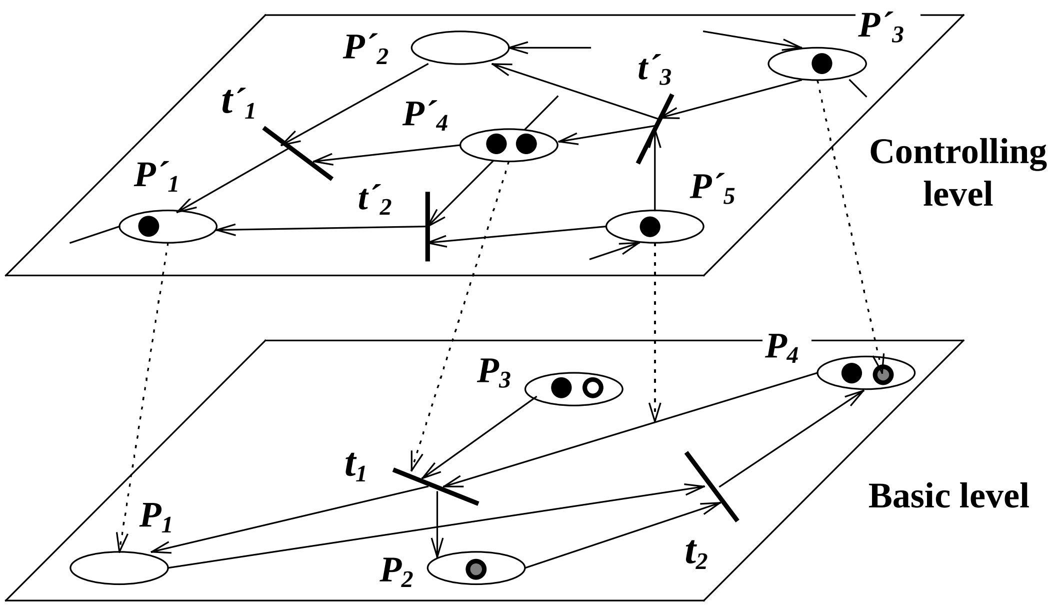

Figure 4. An example to illustrate the idea of metapetrinets [20]

In principle, the controlling level could effect all of the four attributes of a petrinet, i.e. places, transitions, links and tokens. The possible ways of a basic level control are presented in Table 1.

basic level petrinet attributes				
	<place>	<transition>	<link>	<token>
controlling effect	1) removing a place; 2) restoring a place; 3) changing a place's capacity; 4) changing a place's marking	1) removing a transition; 2) restoring a transition; 3) changing time settings; 4) changing the fire rule	1) removing a link; 2) restoring a link; 3) changing a link's direction; 4) changing a link's capacity	1) removing a token; 2) restoring a token; 3) changing a token's color; 4) changing a token's place

Table 1. Classes of controlling interactions between the metapetrinet levels [20]

In [20] it were also considered multilevel metapetrinets where every upper level may change the configuration of the previous lower level. In such a case even a few levels with a simple structure are able to simulate quite complicated control processes at the basic level. It was shown that metapetrinets offer more compact facilities to the models of complex scenarios than single level petrinets. The above abilities of a metapetrinet make it very suitable to model multilevel structure of scenarios (transaction management profiles in location context) in m-commerce.

5. Distance Evaluation within Profiles

One of the important functions of the proposed service is to calculate distances between profiles, e.g. to find out the nearest m-commerce business offering for the given mobile customer's request. Offering and request are two different database objects with similar structure of attributes and their values. However attributes in appropriate profiles are usually heterogeneous (nominal, numerical, logical, interval, etc.). This means that appropriate distance function should be able to measure distance between objects of heterogeneous nature.

There are many approaches to define distance between any two instances based on their numerical or semantic closeness. For example the semantic closeness between terms is a measure of how closely terms are related in the classification schema [27]. Distance metric used by Rada *et al.* [17] represents the conceptual distance between concepts by using only the path length with no consideration of node or link characteristics. Distance is measured as the length of the path representing the traversal from the first classification term to the second one. Rocha [18] has suggested a method to calculate continuously varying conceptual distances between nodes in an entailment mesh, on the basis of the number of linked nodes they share. Instance-based learning techniques typically handle continuous and linear input values well, but often do not handle nominal input attributes appropriately. A probabilistic metric of the PEBLS classification algorithm [5] can be used to compute the similarity between two instances. The distance d_i between two values v_1 and v_2 of a certain attribute is:

$$d(v_1, v_2) = \sum_{i=1}^{k} \left(\frac{C_{1i}}{C_i} - \frac{C_{2i}}{C_2} \right)^2, \tag{5}$$

where C_1 and C_2 are the numbers of instances in the training set with values v_1 and v_2, C_{1i} and C_{2i} are the numbers of instances of the *i*-th class with values v_1 and v_2, and k is the number of classes. The value difference metric was designed by Wilson and Martinez [28] to find reasonable distance values between nominal attribute values, but it largely ignores continuous attributes, requiring discretization to map continuous values into nominal values. As it was mentioned in the Wilson and Martinez [28] review there are many learning systems that depend on a good distance function to be successful. A variety of distance functions are discussed in this review and available for such uses, including the Minkowsky, Mahalanobis, Camberra, Chebychev, Quadratic, Correlation, and Chi-square distance metrics; the Context-Similarity measure; the Contrast Model; hyperrectangle distance functions and others [28].

Our approach to improve distance evaluation quality is to integrate multiple techniques or distance functions. The integration method should be able to estimate and select the most appropriate distance function from the ensemble or combine them efficiently. In [15] a method was presented for dynamic integration of multiple data mining techniques. This method is a variation of the stacked generalization method with an assumption that each of the component techniques is best inside certain subarea of the whole domain area. This method includes two phases: a learning phase and an application phase. During the learning phase a performance matrix of each component technique is derived using the instances of the training set. Each matrix thus contains information concerning the "competence area" of the corresponding component technique. These matrices are used during the application phase to predict the performance of each component technique for each new input instance. The proposed integration method was evaluated on datasets from the UCI machine learning repository. The same idea can be used also with the ensemble of distance functions. It uses a distance function to determine how close a new input vector is to each stored instance, and uses the nearest instance or instances to predict the performance of each component technique inside that local area. That is why the quality of this integration method depends heavily upon the used distance function. In [16] an experimental study was conducted with different distance functions such as (1) Euclidean distance function, (2) Heterogeneous Euclidean-Overlap Metric, (3) Heterogeneous Value Difference Metric, and others. The promising dependencies were shown between the quality of the dynamic integration, the distance function, and some parameters of the dataset [15].

Assume that we have two interpreted profiles with the same set of attributes, which have numerical or nominal values. Assume that first profile is taken from " business requests" database and the second one from the "business offerings" database. The distance between these two interpreted profiles is as follows:

$$E(X,Y)=\sqrt{\sum_{\forall i, x_i \in X, y_i \in Y} \omega_i \cdot d(x_i, y_i)^2}\,, \qquad (6)$$

where ***X*** and ***Y*** are two vectors of the values of the attributes of two profiles. Component distance $d(x_i, y_i)$ for every attribute is normalized by the range of the previously known values of the attribute so that it is mostly in the range [0,1], and weighted by weights ω_i according to the importance of the attribute. ω_i is the probability that a client, whom a request (interpreted profile) belongs to, will not be satisfied by an offering (interpreted profile of the same structure) if i-th attribute of these profiles will not be taken into account.

The Heterogeneous Euclidean-Overlap Metric is used for both nominal and numerical features. This function defines the component distance between two values of an attribute as:

$$d(x_i, y_i)=\begin{cases} \text{if } i-\text{th attribute is nominal - } \begin{cases} 0, \text{if } x_i = y_i \\ 1, \text{otherwise} \end{cases} \\ \text{else}: \dfrac{|x_i - y_i|}{range_i} \end{cases}, \qquad (7)$$

where $range_i$ is the range of the attribute *i*.

Interpolated Value Difference Metric (3) defines following component distance between two values of an attribute:

$$d(x_i, y_i)=\begin{cases} \sqrt{\sum_{j=1}^{k} \left|P(j|i \in [x_i, x_i+\Delta]) - P(j|i \in [y_i, y_i+\Delta])\right|^2}, \text{if } i \text{ numerical (continuous) attribute} \\ \sqrt{\sum_{c_i=1}^{k} \left|P(j|i = x_i) - P(j|i = y_i)\right|^2}, \text{otherwise,} \end{cases} \qquad (8)$$

where *k* is the number of classes of profiles; $P(j|i = x_i)$ is the conditional probability that a profile belongs to class *j* if its attribute *i* has the value x_i, $P(j|i \in [x_i + \Delta])$ is the interpolated conditional probability that a profile belongs to class *j* if its discretized attribute *i* has the value x_i, and Δ is the discretization step.

6. Conclusions

In this paper we study some information management applications in mobile electronic commerce. Paper is based into assumption that the future growth of m-commerce information services (more specifically - in location-based information content sailing) nowadays depends on the ability to integrate different types of information with geographical data and manage different type of profiles related to preferences, security, privacy, transaction management, etc.

A shared understanding of business domain that can be agreed among different players in m-commerce is necessary to provide reasonable service. The multilevel profiling framework discussed in this paper should be able to manage ontologies, which are necessary to standardize views to commercial operations and help users to reach consensus between any buyer and any seller about general and specific features of their business relationships. Web content interpreted via several contexts (filtered through different levels of profiles) becomes really useful for certain user in certain location and can be handled by small mobile devices. Also scenarios for business transaction management can be negotiated based on some profiles (like meta-scenarios) and then used for handling business transactions.

According to the "semantic web" trend, "the emphasis is on enriching the web's data markup languages with knowledge representation features, to permit inference over the content of web pages (prominent initiatives include DAML, OIL, and RDF). Goals include the production of: internet-scale inference mechanisms, knowledge markup languages, and active information-seeking services" [8]. Members of the e-business and the semantic web communities are starting cooperative effort to identify strengths, weaknesses, opportunities, and threats in the interaction between the two areas. One of the key questions is: where will be the biggest e-business pay-offs from using intelligent web technology [8]? Our answer is - commerce with mobile devices.

Acknowledgement

The economic support of the Multimeetmobile project by the National Technology Agency of Finland (TEKES), Nokia, Hewlett Packard Finland, and Yomi Vision under contract 40599/99 is highly appreciated.

References

1. T. Allen, Common Business Library (CBL), May 1999, available in: http://www.infoloom.com/gcaconfs/WEB/granada99/all.HTM#N5.
2. G. Antoniou, F. Maruyama, R. Masuoka, H. Kitajima, Issues in Intelligent Information Integration, Proc. *Internet, Multimedia Systems and Applications*, IMSA-1999, Nassau, The Bahamas, pp. 345-349.
3. Architectural Form Definition Requirements, International Organization for Standardization, 1997, available in: http://www.ornl.gov/sgml/wg8/docs/n1920/html/clause-A.3.2.html.
4. T. Brown, The Use of Profiling in XML Documents: Using Profiling for Personalized Information Delivery, *Document Management Solutions*, White paper, 1 January 2000, available in: http://www.arbortext.com/Think_Tank/XML_Resources/profiling_wp/index.html.
5. S. Cost, S. Salzberg, A Weighted Nearest Neighbor Algorithm for Learning with Symbolic Features, *Machine Learning*, Vol. 10, No. 1, 1993, pp. 57-78.
6. R. Cover, Common Business Library (CBL), July 28, 1999, available in: http://www.oasis-open.org/cover/cbl.html.
7. S. Decker, F. van Harmelen, J. Broekstra, M. Erdmann, D. Fensel, I. Horrocks, M. Klein, S. Melnik, The Semantic Web - on the Roles of XML and RDF, *IEEE Internet Computing*, September - October 2000.
8. E-Business & the Intelligent Web, available in: http://www.csd.abdn.ac.uk/ebiweb/.
9. D. Fensel, M. Cruzeby, F. van Harmelen, I. Horrocks, OIL and UPML: a Unifying Framework for the Knowledge Web, *ECAI'00 Workshop Proceedings*, Berlin, August 2000.
10. A. Garmash, A geographical XML-based format for the mobile environment, *Proc. of HICSS-34 conference*, Hawaii, USA, January, 2001.
11. S. Gordon, J. Billington, Analysing the WAP Class 2 Wireless Transaction Protocol using Coloured Petri Nets. In M. Nielson and D. Simpson (Ed.): *21st International Conference on Application and Theory of Petri Nets (PN2000)*, Aarhus, Denmark, 26-30 June 2000, Lecture Notes in Computer Science 1825, Springer-Verlag, 2000, pp. 207-226.

12. S. Gordon and J. Billington. Modelling the WAP Transaction Service using Coloured Petri Nets. In H. V. Leong, W. -C. Lee, B. Li and L. Yin (Ed.): *First International Conference on Mobile Data Access (MDA'99)*, Hong Kong, China, 16-17 December 1999, Lecture Notes in Computer Science 1748, Springer-Verlag, 1999, pp. 109-118.
13. M. Klusch, M., Agent-Mediated Trading: Intelligent Agents and E-Business, In A.L.G. Hayzelden, R. Bourne (eds.): *Agent Technology applied to Networked Systems*, John Wiley & Sons, 2000.
14. S. Melnik, S. Decker, A Layered Approach to Information Modeling and Interoperability on the Web, September 2000, available in: http://www-db.stanford.edu/~melnik/pub/sw00/.
15. S. Puuronen, V. Terziyan, A. Tsymbal, A Dynamic Integration Algorithm for an Ensemble of Classifiers, In: Zbigniew W. Ras, Andrzej Skowron (Eds.), *Foundations of Intelligent Systems*, Lecture Notes in Artificial Intelligence, V. 1609, Springer-Verlag, June 1999, pp. 592-600.
16. S. Puuronen, A. Tsymbal, V. Terziyan, Distance Functions in Dynamic Integration of Data Mining Techniques. In: B.V. Dasarathy (ed.), *Data Mining and Knowledge Discovery: Theory, Tools and Technology II*, Proceedings of SPIE, Vol.4057, The Society of Photo-Optical Instrumentation Engineers, USA, 2000, pp.22-32.
17. R. Rada, H. Mili, E. Bicknell, M. Blettner, Development and Application of a Metric on Semantic Nets, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 1, 1989, pp. 17-30.
18. L. M. Rocha, Fuzzification of Conversation Theory, *Principia Cybernetica Conference*, Free University of Brussels, Brussels, June 1991.
19. Sandholm, T. 1999. eMediator: A Next Generation Electronic Commerce Server. At the National Conference on Artificial Intelligence (AAAI), Intelligent Systems Demonstration Program, Orlando, FL, July. AAAI proceedings pp. 923-924.
20. Savolainen V., Terziyan V., Metapetrinets for Controlling Complex and Dynamic Processes, *International Journal of Information and Management Sciences*, V. 10, No. 1, March 1999, pp.13-32.
21. G. Swedberg, Ericsson's Mobile Location Solution, *Ericsson Review*, 1999.
22. J. Tang, J. Veijalainen, On E-Commerce Transaction Protocols that Support Atomicity Based Dispute Handling with Untrustworthy Players, *3rd Intl. Conference on Telecommunications and Electronic Commerce*, Dallas, TX, USA, November 16-19, 2000.
23. Terziyan V., Multilevel Models for Knowledge Bases Control and Their Applications to Automated Information Systems, *Doctor of Technical Sciences Degree Thesis*, Kharkov State Technical University of Radioelectronics, 1993.
24. Terziyan V., Puuronen S., Reasoning with Multilevel Contexts in Semantic Metanetworks, In: P. Bonzon, M. Cavalcanti, R. Nossun (Eds.), *Formal Aspects in Context*, Kluwer Academic Publ., 2000, pp. 107-126.
25. What is E-Speak? Product Information, 2001, Hewlett Packard Company, available in: http://www.e-speak.hp. com/product/overview.shtm.
26. The MeT Initiative - Enabling Mobile E-Commerce, *Met Overview White Paper*, 2 October, 2000, available in: http://www.mobiletransaction.org/pdf/MeT_White_Paper.pdf .
27. D. Tudhope, C. Tailor, Navigation via Similarity: Automatic Linking Based on Semantic Closeness, *Information Processing and Management,* Vol. 33, No. 2, 1997, pp. 233-242.
28. D. R. Wilson, T. R. Martinez, Improved Heterogeneous Distance Functions, *Journal of Artificial Intelligence Research*, Vol. 6, 1997, pp. 1-34.
29. XML Translator Generator, AlphaWorks Technologies 21 May 1999, http://www.alphaworks.ibm.com/tech/xmltranslatorgenerator.
30. XSL Transformations (XSLT) Version 1.0, available in: http://www.w3.org/TR/xslt.html.