Modular and Hierarchical User Profile Instantiation for E-Commerce Applications
Using Mobile Software Agents

Oshadi Alahakoon
Dr Seng Loke
Prof Arkady Zaslavsky

School of Computer Science and Software Engineering
Monash University
e-mail: {Oshadi, swloke, a.Zaslavsky}@csse.monash.edu.au

Abstract
eHermes is a Business to Consumer (B2C) e-commerce application, for supporting users in on-line shopping. In this paper, we describe the construction of user profiles, within the User Modeling component of e-Hermes. The focus of our work is to provide personalized e-services by use of user models (profiles). The user profiles are modular in structure, which consists of a fixed core component and an application domain dependable changeable component. The changeable component enables the customisation of user profiles to better represent user needs targeted towards specific application domains. Mobile and stationary agents are used in the processes of information gathering and profile construction.

Keywords
E-commerce, user models, personalization

INTRODUCTION
Most of the currently available e-commerce applications fall in to the category of B2C (Business to Consumer) and B2B (business to Business) (He and Jennings, 2003). B2B refers to business transactions between companies while B2C refers to on line retailing transactions between individual consumers and companies via company web pages. eHermes, which is currently being developed, is a web based multi-agent system, which falls in the B2C category. It has a flexible and extensible open architecture, which can adapt to changing environments. eHermes helps users with their information needs such as financial services and online shopping for goods and services.

In electronic commerce, consumers do their shopping without physically visiting the shopping malls. A personal agent, which is a piece of software, carries out the work on behalf of the consumer. To assist these agents to identify user needs and act accordingly, it is important to have an accurate model of the individual user. Once this user model or the user profile is built, it is possible to make inferences about users preferences to fulfil his/her needs. Personalized services in e-commerce are important to both service-users and vendors. From the user’s viewpoint, such services greatly reduce time spent on searching in various websites. From the vendor’s viewpoint, it increases the sales by approaching the people who really wants to buy the product, and also by converting the “browsers” in to buyers.

In this paper we present the ongoing work of the personalization component of the eHermes system. The advantages of this personalization component are two fold. It would enhance the ability of chermes to unobtrusively obtain user information. In addition the modular structure of the constructed user profiles will provide personalized services in multiple domains. Currently work is being carried out to build an agent based user profile constructor. This personalizing component unobtrusively constructs models of its users, which are modular in structure. To illustrate unobtrusive construction of modular user profiles, we use purchasing of food items, as an example scenario. We discuss the importance and advantages of modular user profiles, with regard to storage and reusability in multiple domains.

The rest of the paper is organised as follows. Related Work section presents related research work in the area of user profiling for e-commerce, with a discussion and comparison with our work. In Unobtrusive Information Gathering section describes the methods of information gathering from multiple sources. In Domain Hierarchies section we describe, the construction of domain hierarchies and in Importance of Domain Hierarchies and Profile Structure section we discuss the importance of modular profile structure. System Overview section presents the overall architecture of our system and how the techniques described in other sections can be combined together to implement the model. The last section provides the conclusions and future work.
RELATED WORK

A number of systems, which attempt to provide improved customer services with user profiling techniques, have been described in literature. The main aim of these systems is to build a profile of the user to be used in providing the user with products that satisfy his/her needs. In all such systems the profiles are built using content-based methods, collaborative methods or a hybrid of these two methods.

In content-based methods predictions are done about the future behavior, needs or priorities of an individual, based on his/her present behavior, needs or priorities. The system monitors individual user behavior and use that information to predict his/her future behavior towards new events, depending on the past experience.

In collaborative predictions, an individual is mapped in to an existing user group and then predict his/her behavior, needs or priorities based on the other group members. First, depending on their demographics, needs or behavior, users are classified in to groups. Then new users are mapped in to the groups of people with similar interests or characteristics. The new user inherits all the common features of the group. Hence the possible needs of the current user could be guessed.

A common feature in many of these systems is the use of stereotypes as an initial information source for building user profiles. The notion of stereotypes first appeared in Rich’s GRUNDY system (Rich, 1989). In GRUNDY, individuals are mapped to similarly behaving groups, depending on their attributes (triggers). The mapped individuals then inherit all the other attributes of the group. The stereotypes are initially generated before starting the system and change over time influenced by the user interactions. Stereotypes can be constructed manually or by using machine learning clustering methods. Manual stereotypes are built either using domain knowledge about people or by analyzing the existing information such as survey data about people. Machine learning techniques (artificial Intelligence methods) such as neural networks can be use to identify clusters (groups) of people in existing data sets.

The system described below (SETA) use some knowledge about its users and hence has some similarity to our proposed model. This system is described in (Ardissono et al., 1999, Ardissono et al., 2001) and does user modeling and flexible hypermedia techniques to tailor the suggestion of goods and descriptions of the store catalogue to individual users. It does parallel execution of identifying the user preferences and dynamically generating the hypertext pages of the store catalog. It uses collaborative methods to map a new user to an existing stereotype, based on his/her socio-demographic characteristics. A mapped user inherits all the predicted behavior of that stereotype. The stereotypes used in this system have to be predefined. Although the user profiles are updated according to user behavior, stereotypes remain unchanged. Such fixed stereotypes becomes a major limitation in dynamic environments, where available items may keep changing. This leads in to imprecise user models.

The system described in (F. Abbattista et al., 2002), presents a personalizing component that uses supervised machine learning to induce a classifier able to discriminate between interesting and uninteresting books for the user. Its aim is to improve customer supplier relationship in future e-commerce. User information is obtained by explicit user inputs (registration data or questions and answers), using stereotypes and depending on the user history (e.g. purchases made, number of visits etc). Based on the initial interaction, this system uses collaborative methods to assign new users to pre-existing models. These models simply consist of book categories preferred by a user. The profile is changed over time using content-based methods over user behaviour. The profiles used are more application specific and hence unusable for online purchase of different types of items.

The project described in (Tewari et al., 2000) is the design and implementation schema of a subset of the Impulse research project at the MIT Media Lab. This addresses the location specific resource brokering. (Location based support for recommending restaurants in the physical proximity of a mobile human user, for a context where a person can’t wait for long delivery latencies). The profiles are populated first by initial user inputs. The rest of the information is filled out using (collaborative methods) stereotypes. User can override the values if required. The final output is the best matching restaurant. This system is more intrusive and application specific as well. The number of stereotypes is limited and static.

There are several other popular and much published e-commerce systems described in literature. These systems such as PersonaLogic, Kasbah, Firefly (Gutman et al., 1998, Gutman and Maes, 1999) provide online purchasing services based on user profiles. They use more intrusive methods of building profiles. PersonaLogic and Kasbah use product attribute constraints to filter out consumer needs from a large number of products. Users have to explicitly specify their likes and dislikes towards product attributes. Firefly uses collaborative methods to purchase books and music. A new user inherits the preferences of an existing user based on his ratings for an item (like minded ratings closer to user ratings).
Our system proposed in this paper, attempts to overcome the major limitations highlighted in the above systems. Mainly the application dependency is addressed with modular user profiles. The use of multiple information sources and both content based and collaborative prediction methods result in a less intrusive user profile building system.

**UNOBTRUSIVE INFORMATION GATHERING**

In any service-oriented system, direct user inputs provide the most precise and first hand information about the user. Although ideal in terms of information gathering, systems expecting high user inputs may discourage or even offend some users. Therefore it is very important to extract as much information as possible in an unobtrusive manner. Our approach to this problem is to obtain the minimal possible initial information from the user, and acquire more information from reliable information sources depending on the existing information. Attempts to achieve such unobtrusiveness will result in a drop in precision. Therefore we need to come to a compromise between precision and unobtrusiveness. Methods of unobtrusive information gathering are described below.

Once sufficient information satisfying the user request has been gathered, the system maps him/her to a closest matching stereotype. The stereotypes are built from application domain knowledge, and organised into specialization hierarchies. As such, the more information available will result in a better and more accurate mapping in to a stereotype at a lower (more specialized and better matching) level of the hierarchy. In the proposed system, we let the user decide on the level (or amount) of information to be divulged, such that the system can then take appropriate action to ‘fill up’ any other missing information. The filling up of the information will be carried out using heterogeneous set of multiple data sources, which is discussed in this paper. All the information gathered from these diverse sources is used to build a highly personalised user profile. For an existing user, the current profile will be updated with any new information or changed circumstances.

![Figure 1: Use of heterogeneous information sources](image)

As shown in figure 4, we use five techniques in information gathering. First the user registration details are used as basic information for the profile. If the user already had a profile populated for a different request that information is obtained.

**User inputs, Unobtrusive information gathering and Precision of the user models**

Users of online shopping will prefer a system, which is less obtrusive. That is a system asking fewer questions from the users to understand user needs. Most of the available user modeling systems requires various user inputs as the basis of drawing inferences about its users. Users can provide information about their needs, preferences, etc. by interacting with the system in different ways:

- Conversation with the system (dialog systems) (Rich, 1989) – Users are asked questions and depending on their answers they are mapped in to stereotypes. One user can inherit from more than one stereotype & the model of the user is a collection of stereotypes.
- Explicit input or selecting appropriate values from a set of given values (Morris, 2000, Tewari et al., 2000, Middleton and D.C. De Roure, 2001)– in these systems the users are asked to fill a form with their demographic information & their preferences.
- Relevance ranking (Balabanovic and Shoham, 1995, Shearman and Lieberman, 2001). – Users are asked to rank the findings of the system. Then the system adapts the user model according to user rankings.
As described above it is possible to obtain the most accurate and reliable first hand information from the user about him/herself. But users prefer to spend less time with providing their information. So to make precise user models it is required to employ alternative user information sources. These alternative information sources will have varying levels of accuracy. As a measure of maintaining the accuracy of the profiles, we assign priorities to information sources. This also helps in conflict situations where values are obtained from two different sources, for the same attribute.

The assigned priorities are as follows. (Highest to lowest)

- Direct user input
- Information already in the profile
- Information gathered from the web and external databases
- User behavior (past performance and users feed back on results)
- Stereotypes

The most important attributes get values from user’s direct input, which is the highest priority information source. Those are the most critical attributes towards the accuracy of the user profile; hence made compulsory to provide by the user at registration.

**DOMAIN HIERARCHIES**

In cyber shopping, humans and products do not meet each other. The purchaser is faced with the problem of finding items to satisfy his needs, out of a massive number of products, which are online. As such, a cyber shopping system should contain a way of mapping user interests to appropriate products. A ‘one way’ mapping of user needs to products by user preferences is not expected to satisfy this requirement. Instead these systems require the knowledge about user (customer) properties and preferences towards the products as well as knowledge regarding what product characteristics will map to user properties (Ardissono et al., 1999).

To implement such a ‘two way mapping’, we develop profiles of users, which are structured in a way that facilitate such mapping. We will use the definitions used for domain-objects, domain-attributes and user-attributes in (Kay 1994) to describe this techniques. These definitions are as follows.

- **Domain-objects (DO)** - These are the actual objects in the domain. (e.g., if the domain is grocery shopping, then the domain objects are actual groceries)
- **Domain-attributes (DA)** - These are the attributes describing the objects. (e.g. if the domain is grocery shopping, then attributes of groceries, if the domain object is Butter then its attributes can be price, weight, cholesterol content, etc.)
- **User-attributes (UA)** - These are the properties of the user. This is independent of the domain. (Example, user’s age, education, etc.)

To achieve above described mapping, we need the user profiles and domain objects in a suitable form.

i) The user profiles should contain user needs in the form of his/her interest in domain attributes.
ii) The pre-processed domain objects should be in databases with their descriptive information.

There are different types of goods available online for purchase. User’s needs, priorities and behavior vary for each of these different purchases. As an example a person who is interested in buying quality garments may not behave the same when buying furniture. In such situation, for different goods, the same user attribute holds different values. The most obvious answer is to maintain different user models for purchasing of different kinds of products. But descriptive information about users do not change while above described user attributes describing domain attributes changes with the different purchases. Therefore the user attributes describing domain attributes can be considered as application dependent. To model users accurately we may need to know user attributes that are application specific as well as some which are more general. When we look at the shopping available online, it is possible to arrange different shopping domains hierarchically (Figure 2).

The classification shown in Figure 2 is as follows. Shopping could be either short-term purchases or long-term purchases. Short-term purchases are mostly day-to-day life needs, which are comparatively less expensive than long-term purchases. Items higher up the hierarchy are more general while other shopping domains need special user characteristics. For an example, generally we may need to know the age of a person to do shopping for him. But to know whether a user is a vegetarian or not will be useful only in food purchasing. Therefore we can arrange domain dependent user attributes according to the above hierarchy such that, depending on the user request we can obtain the user attributes that need to fulfill his/her needs.

Alahakoon, Loke, Zaslavsky (Paper #289)

14th Australasian Conference on Information Systems
Page 4
26-28 November 2003, Perth, Western Australia
User Profiles

As discussed above a profile of a user has to contain two types of attributes that properly describe the user.

i) Attributes that are characteristics of the user.

ii) User’s likes and dislikes towards item attributes he/she is expecting to purchase.

We call former attributes as Classifications and latter as Classification based predictions. The two terms are described below.

i) Classifications

These are user’s demographic information and special behaviors such as smoker or vegetarian. Classification details are considered as domain (items purchasing) independent. But certain classification details may have an effect on his/her purchases. For example, whether a user is a vegetarian or not will be useful only in food purchasing. But still it is better not to leave any attributes aside as there may be relations between some classification details & user-buying patterns, which are not obvious.

ii) Classification based predictions

Prediction details are either domain dependent or independent. Domain dependent predictions are the descriptions of the items, predicted based on user’s characteristics that might affect his purchases. For an example, if the user is interested in purchasing weekly groceries and if user classifications say he/she is a vegetarian, then the prediction is not to include meat items in the shopping list (Meat – N, no meat). Domain independent predictions are made based on classification details. For example if the title of a person is “Mr.” then the gender is predicted as Male. All these predictions are done either based on user past behavior or available classification details, using stereotypes.

Figure 3 describes the contents of a profile.

Classification details are kept as domain independent. They are obtained from user inputs, web search and using stereotypes. Some of the attributes never change as name, date-of-birth or gender; these are called static attributes. Attributes that can change over time such as address, occupation or income are called default attributes. As an existing user makes requests, to purchase items from different shopping domains, system updates the default user classifications (if changed) and re-generates appropriate classification predictions. As classification details are domain independent, such information is kept common for different domain dependent information.

Figure 2: Hierarchically arranged application domains

Figure 3: Contents of a User Profile
Pre-processed Domain Objects

In our grocery-food purchasing scenario, we maintain databases of food items for different supermarkets. For each supermarket the data is organized hierarchically. We maintain a complete description of food items with their applicable attributes such as name, expiry date, price, brand name, food-state, sugar content etc. For an example, “need of fat free items” is a user attribute describing a domain attribute. When system search for matching food items for that profile, the domain objects with matching attribute descriptions (e.g. object: Cheese, Attribute: Fat level – 0) get selected. Preparation of data according to above described manner, facilitate finding of user needs by matching the predicted user needs and domain object descriptions.

IMPORTANCE OF DOMAIN HIERACHIES AND AND PROFILE STRUCTURE

Once the domains are hierarchically arranged, it leads to modular structured user profiles, which can be hierarchically linked. In figure 4, we illustrate the incremental growth of user profiles with continuous usage of the system. We have considered user A’s shopping behaviour across several domains. In Figure 4.1 the user has started shopping in domain 1a. Once the profile is build for that domain, domain 1 which is up in the hierarchy inherits approximated predictions. Such predictions are based on predictions in domain 1a. As the user’s shopping continue across domains, his/her profile grows incrementally. The classification details part of the profile remains common for all the domains and predictive parts are different from one another, to be used in corresponding applications.

By maintaining these modular profiles we have the following advantages.

- Personalization arises concerns about privacy issues. People using web services sometimes hesitate to give away critical personal information such as birth date or credit card details. In our system, the modularity of user profiles support privacy, as it is possible to store potions of the profile in different locations. Users are allowed to request their privacy preferences by allowing them to store parts of the profile either with the system or with themselves.
- The classification details part of the profile is built only once and reusable as the user continue to shop across different domains. If user prefers, it is possible to store his classification details in his computer.
- Each module of the profile describes user related to a certain domain. This gives a better representation of the user with regard to that particular domain

SYSTEM OVERVIEW

To demonstrate the above discussed methodologies we use a food-recommendation scenario as an example. The system builds user’s profile based on the registration details he/she input and other information collected by
using diverse information sources. Finally the system generates an item list selected according to his/her profile. The prototype system demonstrating the above example is still under construction, and in its preliminary state.

In our example scenario, we achieve unobtrusiveness by employing different user detail gathering techniques other than user inputs. The information sources used are online White Pages and a medical database with user health information. The user profiles are structured modularly and user classification details and user classification based predictions are saved separately to facilitate the use of classification details for different applications.

Domain data repositories need to be there in advance with pre-processed information, which is in the form of ready to map in to user needs. For implementation purposes, we have product data of different supermarkets pre-processed and stored in different databases. Once the profile is ready, it is mapped to the products in the supermarket, which is located close to the user address.

Figure 5  Shows an abstract view of the overall architecture. System interacts with the user through the user interface. Profile population and updating is carried out using information from external data sources and stereotypes. Finally depending on the user request the domain objects are selected according to his/her profile, from the corresponding application domain. Once the system selected item list is presented to the user, user’s feedback (adding or removing items) is recorded and used in profile updating.

Within the above processes there are mobile and stationary agents involved in to perform various tasks. Mainly these agents do the interacting with the users and information gathering for profile population. From the point of registration to the construction of the user profile, the steps involved are as follows.

i) Before using the system for the first time, users have to register for the system by filling a registration form. Although this is designed for collecting various detailed user information, to be less intrusive, only few of these fields are compulsory to filling in.

ii) If the minimum is filled in, the user interaction agent passes the remaining attributes to be filled out to the Web Search agent. In our implementation as a temporary measure we use White Pages and a Medical database that contain some user information (Assume we have the right to use it) to gather additional personal information.

iii) The User Interaction agent analyse the request and inform the Domain Information Agent, about user’s interesting shopping domain (e.g. food items, clothing car, etc). Then this agent visits the websites to find either more information about that domain or user involvements in the domain. By communicating with agents who monitor these web sites, Domain Information Agent brings back any information that can be added to the user profile.

After above ways of gathering & filling the user profile are performed, the classification information is sufficient to do the predictions towards user needs in the selected shopping domain. There are predefined domain dependent object attributes that need to be evaluated based on above classification details in the profiles. For food items, the important domain attributes are of two types: Quantitative and Qualitative. Quantitative needs are number of items the user needs and qualitative needs are item description based. For example, quantitative attributes are number of food items need in fresh state (fresh-food), in half-done state (half-done-food), and raw items (cooked-food). The values for this change depending on the amount of time a user can spend on food preparation. A person doing a part time job is likely to spend more time on food preparation
rather than one with a full time job. If number of working hours is less, time available for food preparation is more. Such user needs more cooked and half-done food than raw food. For both finding the times available and then to decide on quantities of different states of food items, stereotypes are used. An example of a complete user profile built for grocery purchasing is shown in Figure 6. User profiles mainly have two parts. First part contains user classification information and the second part contains classification based predictive information. Each of these parts has attribute, value pairs in columns. In the Predictions, the quantitative attributes get values L-Low, M-Medium or H-High. The qualitative attributes get Y-Yes or N-No.

<table>
<thead>
<tr>
<th>Classification part</th>
<th>Predictive part</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static</strong></td>
<td><strong>Quantitative</strong></td>
</tr>
<tr>
<td>User-Id</td>
<td>No_of_Snack_Meals: L</td>
</tr>
<tr>
<td>Name</td>
<td>Fresh_food: M</td>
</tr>
<tr>
<td>Date of Birth</td>
<td>Half_done_food: M</td>
</tr>
<tr>
<td>Age</td>
<td>Cooked_food: L</td>
</tr>
<tr>
<td>Gender</td>
<td>Cholesterol_free_food: L</td>
</tr>
<tr>
<td>Cultural background</td>
<td>Sugar_free_food: H</td>
</tr>
<tr>
<td>Australian</td>
<td></td>
</tr>
<tr>
<td><strong>Default</strong></td>
<td><strong>Qualitative</strong></td>
</tr>
<tr>
<td>Address</td>
<td>Egg: Y</td>
</tr>
<tr>
<td>Suburb</td>
<td>Meat: N</td>
</tr>
<tr>
<td>Marital status</td>
<td>Fish: N</td>
</tr>
<tr>
<td>Occupation</td>
<td></td>
</tr>
<tr>
<td>Monthly Income</td>
<td></td>
</tr>
<tr>
<td>Weight</td>
<td></td>
</tr>
<tr>
<td>Height</td>
<td></td>
</tr>
<tr>
<td>Allergies</td>
<td></td>
</tr>
<tr>
<td>Habits</td>
<td></td>
</tr>
<tr>
<td>Special preference</td>
<td></td>
</tr>
<tr>
<td>Car</td>
<td></td>
</tr>
<tr>
<td>Special health conditions</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Figure 6: A complete user profile build for food item purchasing*

For any other request of the user (e.g. to buy household cleaning items or to buy personal care items) it is possible to use the same classification information of that user. Only the predictions need to be re-generated. The predictive part of user profiles organized according to the domain hierarchy (for simplicity, we use only a sub-hierarchy) of Figure 2 is shown in Figure 7.
As shown in figure 7, the prediction details of an individual can be linked hierarchically. Managing profile data in this manner, helps re-use of profile components (we reuse the classification part), and also helps profile maintenance – updating the food items prediction part updates the prediction part for grocery buying.

When the profile is all complete and ready, the user preferences are mapped with the items in the domain object database. Finally, the system generates the selected list of food items. Users are allowed to edit this system generated Item list. Users can either add new items that they prefer to have, but not already in the list, or delete any item in the list, that they don’t need. This system-user interaction is recorded in the past history repository to update the user profile.

CONCLUSION

The work presented in this paper is part of the ongoing research and development of eHermes project. The motivation of this work stems from the vision of creating adaptive e-commerce applications, which provide personalised, simplified and secure services to customers. In this paper our main focus is on modular structured hierarchical user profiles and the gathering of information in an unobtrusive manner. The modularity of user profiles we construct facilitates (a) Reusability of profile components (saves time), (b) Use of different and desired locations of storage for potions of the profiles (important for privacy issues and security), (c) Each module better describing users with regard to purchasing items (a better representation of the user).

Among the currently available e-commerce systems using user profiles we can see the following as main differences from our on-going work:

- Use of hierarchically arranged modular user profiles; enable users to re-use potions of the same profile for different types of purchases. This results in reducing the time spent on user information gathering and profile construction.

- Less intrusive manner of initial user information gathering by using different information sources in addition to user inputs (and different information gathering strategies).

- To improve the dynamic adaptation of the system we propose new stereotype construction and updating existing stereotypes by periodically monitoring the changes in user profiles and product databases. Segmenting user information by identifying clusters can be used to build stereotypes of
users. Our system not only requires stereotypes, but hierarchies of stereotypes organized by level of specialization. The unsupervised neural network model GSOM (Alahakoon et al., 2000), is well suited for this purpose.

As a possible answer to privacy issues, users are given the option of storing the profile either in users PC or in the system profile repository, according to his/her desire. As privacy requirements dictate we will continue to explore these issues. We are currently implementing a prototype of our system using .NET platform and C#. We would also like to enhance our user-modeling component as to include personalization to software agents by profiling the agents requiring eHermes services. This will allow the systems dealing with agents to learn about agent’s characteristics in advance, before any interaction. Finally we aim to evaluate the effect of the unobtrusiveness and the effect of modularly structured profiles. The system can be tested quantitatively to prove the ability of building effective user profiles with a lesser number of direct user questions (70-80% less questions), than an intrusive system. The importance of modularly structured profiles, will demonstrated by using such profiles to purchase items from different application domains.

REFERENCES


COPYRIGHT

Oshadi Alahakoon, Seng Loke, Arkady Zaslavsky © 2003. The authors assign to ACIS and educational and non-profit institutions a non-exclusive license to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive license to ACIS to publish this document in full in the conference papers and proceedings. Those documents may be published on the World Wide Web, CD-Rom, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors(Shearin and Lieberman, 2001).