USER PROFILING:
A MOBILE NETWORKS PERSPECTIVE

Shashank Agarwal
Department of Computer Science and Engineering,
University of Texas at Arlington
Email: agarwal@cse.uta.edu

Abstract:

In present scenario, when increasing number of users are using mobile devices to search for information from the Internet, it has become quite necessary to personalize and filter the vast amount of information available. User Profiles are efficiently used to cache only the relevant information on the user's mobile device. Furthermore, User Profiles can also be used to improve QoS, by including the Mobility Profiles. In this paper, we discuss User Profiling successfully implemented for efficient caching and improving QoS.

Keywords
User Profiles, Information Filtering, Data Caching, Mobile Network, Mobility Profiles.

Introduction

An increasing number of clients use mobile devices for diverse applications such as stock quotes, news headlines, travel information, weather reports, and email. As wireless connectivity becomes more pervasive, the number of mobile devices, applications, and services is likely to increase correspondingly. Currently most wide-area wireless networks have limited bandwidth (on the order of 10s or 100s of kilobits). As the number of clients and applications increase, it is necessary to better utilize the available bandwidth and improve QoS on low-bandwidth wireless networks.

Efficient web browsing on mobile devices presents a unique challenge. These machines are different from other classes of client computers since they have relatively low bandwidth connections and they are battery-powered and therefore limited by their energy consumption. Moreover, the mobile nature of such devices combined with the economic limitations of size and costs make it impractical to keep them continually connected to the infrastructure network. "Caching" seems to be the best solution to overcome the problems of disconnection. The term "Data Recharging" [1] has been defined as updating the cache of the user's mobile device with the components of user's interest and with least user intervention. For data Recharging, however User Profiles must specify not only the user's data interests, but also the priorities and preferences the user has regarding those items.

Moreover, the handover procedure is one of the most critical mechanisms in wireless networks. If a handover procedure is necessary due to e.g., a bad link quality due to high path loss or interference, but no capacity is available in the next cell, the communication link is often interrupted if quality decreases further. A number of mechanisms have been proposed to reduce these losses. One possible solution is the reservation of network resources in the next cell in advance. These methods assume that the mobile networks are able to make forecasts for prediction of the user's movement.

I. *User Profiles in Data Caching*

1. *Characteristics of User Profiles*

Profiles for data recharging must contain two types of information. First, the profile must describe the types of data that are of interest to the user. This description must be *declarative* in nature so that it can encompass newly created data in addition to existing data. The description must also be flexible enough to express predicates over different types of data and media. Second, because of bandwidth, device-local storage, and recharging time limitations, only a bounded amount of information can be sent to a device during data recharging. Thus, the profile must also express the user's preferences in terms of priorities among data items, desired resolutions of multi-resolution items, consistency requirements, and other properties. A key challenge for data recharging, therefore, is the development of a suitable language and processing strategy for these highly expressive user profiles.

2. *Creating User Profiles*

Some methods have been proposed, which can be used to create, update and modify user profiles [2].

2.1. **User Created Profile -** In this approach the user specifies his/her areas of interest by a list of (possibly weighted) terms. This is the simplest and most natural approach, and the specified terms are used to guide the filtering process.

2.2. **System Created Profile by Automatic Indexing** - A set of data items which have already been judged by the user as relevant, are analyzed by software, in order to identify the most frequent and meaningful terms in the text. Those terms, weighted according to the frequency of their appearance, constitute the user profile.

2.3. **System plus User-Created Profile -** This is a combination of the above two approaches. First, an initial profile is created automatically (by automatic indexing). Then, the user reviews the proposed profile and updates it (by adding or deleting terms, and changing their weights).

2.4. **System-Created Profile based on Learning by Artificial Neural-Network (ANN) -** Based on a sample set of data items that have already been judged relevant by the user, an ANN may be trained. The inputs of the ANN are the meaningful terms, and the outputs are the relevance judgments of the users. An algorithm can calculate a Causal Index that gives the relative magnitude (and sign) of the influence of each input on each output. After training, the ANN may serve as the user profile for future filtering [8,9].

2.5. **User-Profile Inherited from a User-Stereotype** - This method assumes that the Information Filtering System (IF system) has pre-defined user-stereotypes. A user-stereotype is represented as a content-based profile, i.e. a weighted-vector of terms that represents a set of (virtual) users who have common information usage and filtering behavior. A user-stereotype is also represented by a set of demographic and social attributes that are common to those users. A new user is attached to a predefined stereotype to which he/she is most close with respect to the demographic and social attributes. The user inherits from his stereotype its content-based profile [10].

2.6.**Rule-based Filtering -** All previous methods deal with the creation of a content-based profile. Contrarily, a rule-based profile consists of a set of filtering rules. Questioning the user on his/her information usage and filtering behavior can generate such rules. An alternative method for creating a rule-based profile for a user is to inherit filtering rules from user-stereotypes, similar to the inheritance of a content-based profile. As before, this method assumes that the IF system has pre-defined user-stereotypes, but in this case a user-stereotype is represented by a set of filtering-rules that are common to the users who belong to the stereotype. As in [DIRECT 2, INDIRECT 17], user-stereotype is also represented by a set of demographic and social attributes that are common to those users. A new user is attached to a predefined stereotype to which he/she is most close with respect to those attributes, and inherits from that stereotype filtering rules.

3. *Architecture Overview* [1]

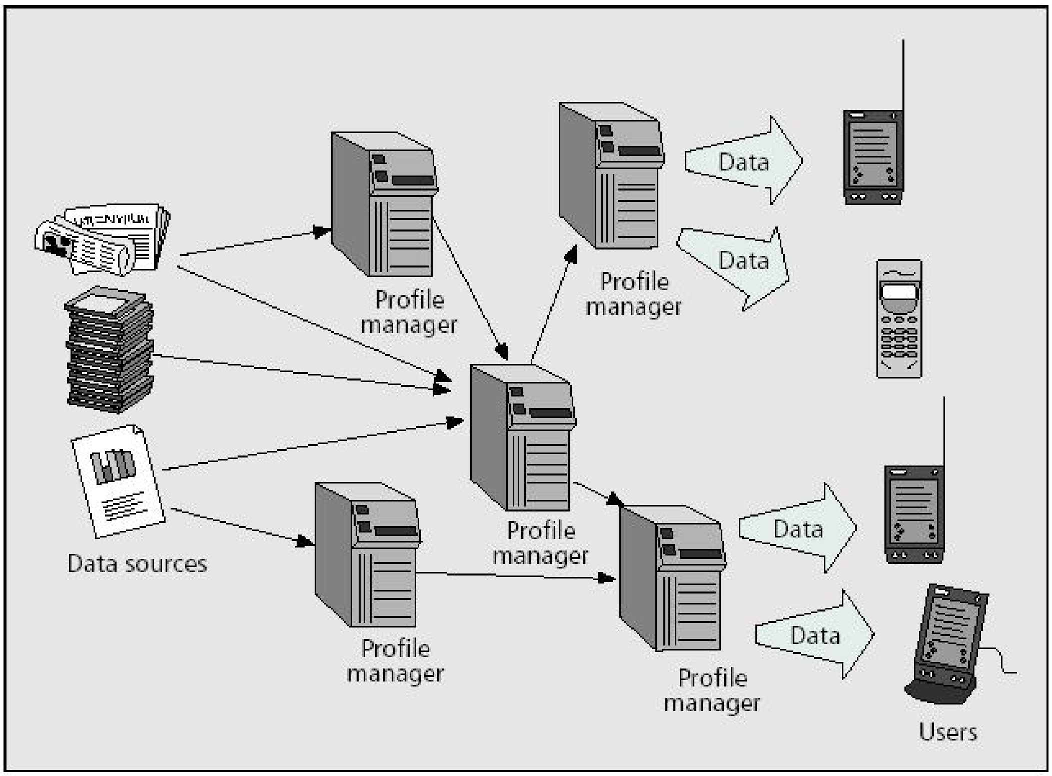

Figure 1. Architecture [1]

The data recharging service is implemented as a distributed system that consists of a network of *profile managers* located throughout the Internet. Profile managers are responsible for retrieving needed data from data sources and packaging it for delivery to specific devices based on their recharging profiles.

There are three different functions of a profile manager:

- **Collecting data** - Profile managers can request and, if necessary, reformat the data, interacting directly with the underlying data sources.
- **Managing Profiles** - Profile managers are responsible for storing and managing profile information for individual users, and for delivering the data to the user devices during recharging sessions.

- **Collecting user patterns** - Profile managers aggregate profiles from many devices and data from many sources in order to more efficiently distribute data through the network.

Profile managers can play one or more of these roles concurrently. In a sense, profile managers are application-level routers that move data through the network based on the detailed information available in the profiles. The profile information flows from the devices and edge nodes back through the infrastructure, being aggregated as it goes. The data collector nodes, to obtain relevant data from the data sources, ultimately use this aggregated profile information. The collected data is then spread through the network in the opposite direction, being distributed to the edge nodes from which it will ultimately be delivered to the devices.

4. *Expressing User Profiles*

Let us consider an example of a User Profile [5].

```
PROFILE Academic
DOMAIN
S = SELECT School
FROM www.acm.org/JobAnnouncements.db
WHERE Area = “Databases”
P = SELECT Title, Author, Affil, Text
FROM www.vldb.org/Papers.db
WHERE Topic = “Query Optimization”
UTILITY
U (S) = 1;
U (P [#S [S.School = Affil] > 0]) = 10
END
```

Figure 2. Academic Profile [5]

Figure 2 shows one possible way to express a profile, which we have explained through Academic Profile. The profile in this figure has two sections: the DOMAIN section specifies the *profile domain*, and the UTILITY section specifies the *utility function.* Let us consider issues regarding user profiles, using the profile example of Fig. 2.

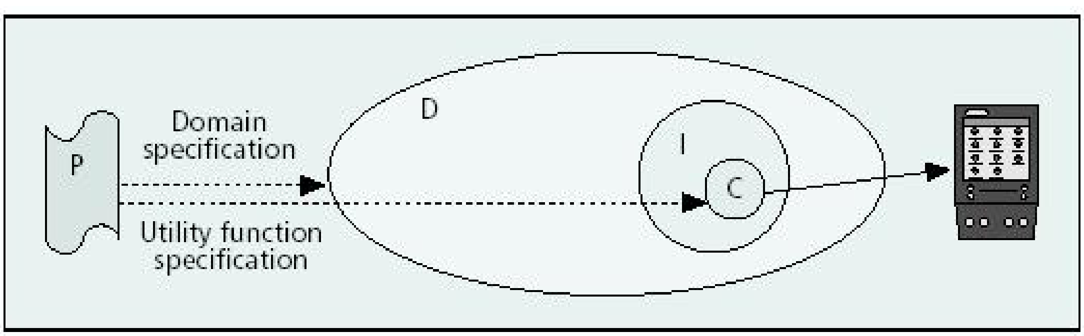

Figure 3. A Profile working in a data Recharging System [1]

Let me first illustrate how profile specifications determine the behavior of the profile manager. Figure 3 shows a profile, **P**, in the context of a data recharging system. **D**, **I**, and **C** denote progressively smaller subsets of data objects which are determined by **P** and the profile manager that processes **P**. **D** is the *profile domain*: the set of all data objects that are specified to be of interest within **P**'s domain specification. **I** is the *profile instantiation*, consisting of the data objects in **D** the profile manager is able to locate. **C** is the chosen *data charge*: the subset of objects in **I** that the profile manager chooses to deliver to the mobile device on the basis of **P**'s utility function specification. For the Academic profile described in this section, **D** consists of two kinds of data objects: School Objects and Publication objects, and **C** is determined from the School and Publication objects the profile manager is able to find (**I**), the utility function specification that establishes the relative worth of these data objects, and the resource restrictions (e.g., bandwidth, space available on charged device) evident at the time when charging occurs.

II. An example User Profiling Network "OCEAN" [7]

1. Objective

While the range of network device has been spreading from PCs, PDAs to mobile phones and home appliances, their objects stored in them are still left isolated, and this isolation causes inconveniences. To resolve those inconveniences, it becomes important to keep liaison (association) of the objects in those information devices. Furthermore, the variety of communication methods has also been increasing.

Therefore, in order to provide liaison of objects, the OCEAN system propagates *proper contents* at *proper timing* by *proper method* according to *object profile*, *device profile*, and *user profile*.

The OCEAN system supposes that the growth of network infrastructure and improvement of network connectivity will allow information devices to be clients of network storage. A server stores and manages the masters of objects, and network devices store proper replicas of master objects according to the character of the device. In other words, a small mobile device with minimal memory will store objects of strong interest to the user of the device, and a device like a note PC with larger memory will store more wide range of objects. The policy of memory management will also differ among devices according to the respective characters. For example, any object about phone number list must not be expired in a device with telephone function.

Hence, the OCEAN system will not only make objects liaison according to the profiles, but also provide the device side with basic functions such as memory management and user notification.

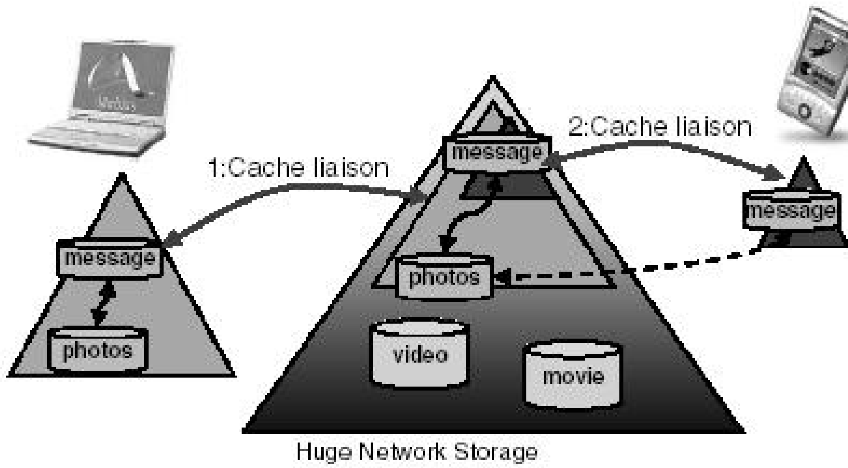

Figure 4: Object Liaison in OCEAN [7]

Figure 4 shows typical object propagation. In this example, an object comprising messages and photographs is recently created in a PC, then the PC propagates the object to the server, and then the server propagates the object to a proper PDA. In this process, only the message element and link information for elements of large photographs of the object are propagated, since the PDA has limited memory size and poor communication method charged by quantity. Therefore, the propagation is carried out within the cost matching to the importance of the object.

Thus, the objects in those devices are automatically associated. Hence the device notifies the user about updates of shared objects when other user updates them, the novel communication scheme of objects sharing becomes possible.

2. OCEAN Architecture

A prototype of OCEAN framework is currently under development. This section provides an overview of planned system and describes some major underlying technologies.

3. Overview

Considering the purpose of this environment, this architecture should be commonly applicable to every kind of network devices. Therefore, it is desirable that the process carried out on device side should be as light as possible. This proposing system utilizes a server client model i.e. the server takes a heavy part of processes so that each client can take minimum tasks. The server keeps liaison of the devices' objects by managing following three kinds of profiles. First is object profile, such as owner, type, importance, and so on. Second is device profile, such as owner, device type, and performance of connectivity and so on. Third is user profile, such as Nickname, age, preference, his or her belonging Group-IDs, and so on.

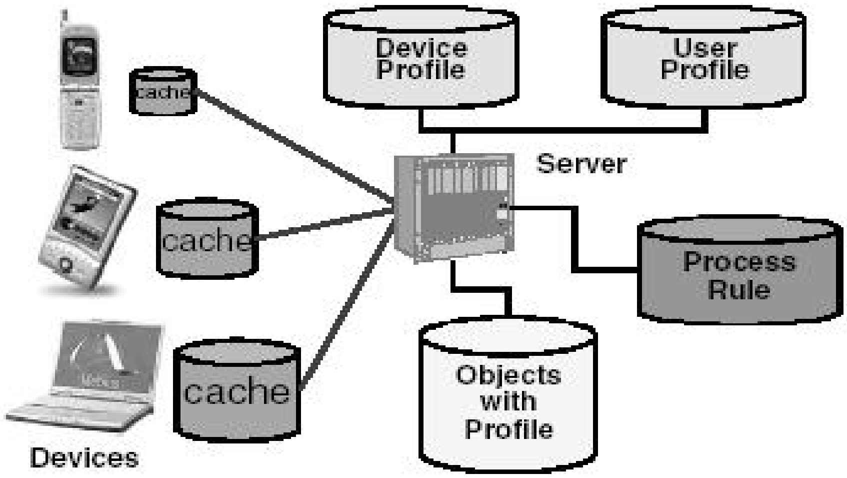

Figure 5. Basic Structure [7]

Figure 5 shows basic structure of the proposed architecture.

Objects are managed in a structured format, and generally consist of some elements, each of which is minimal unit of objects such as a photo image of an album object for example. The common structure of major types of objects and the description of objects' profiles are specified in the OCEAN system.

When an object is updated in a device, the necessary elements of the object are propagated to sever and then, the server transmits its modification to some proper devices with proper elements, according to the profiles above. Then the device, which received the modification via the server, will notify the user about the modification when another user makes a change in the object.

The modification is sometimes transferred by appropriately selected subset of the object. That is, the subset contains only the elements, which can be treated by the target device and can be transferred within the cost matching to the necessity of the object. Furthermore, some elements may be simplified or transformed according to the capacity of the targeted device. For example, an image of high resolution and deep depth might be simplified to a low resolution and shallow depth image when transferred to some devices with poor representation ability such as mobile phone.

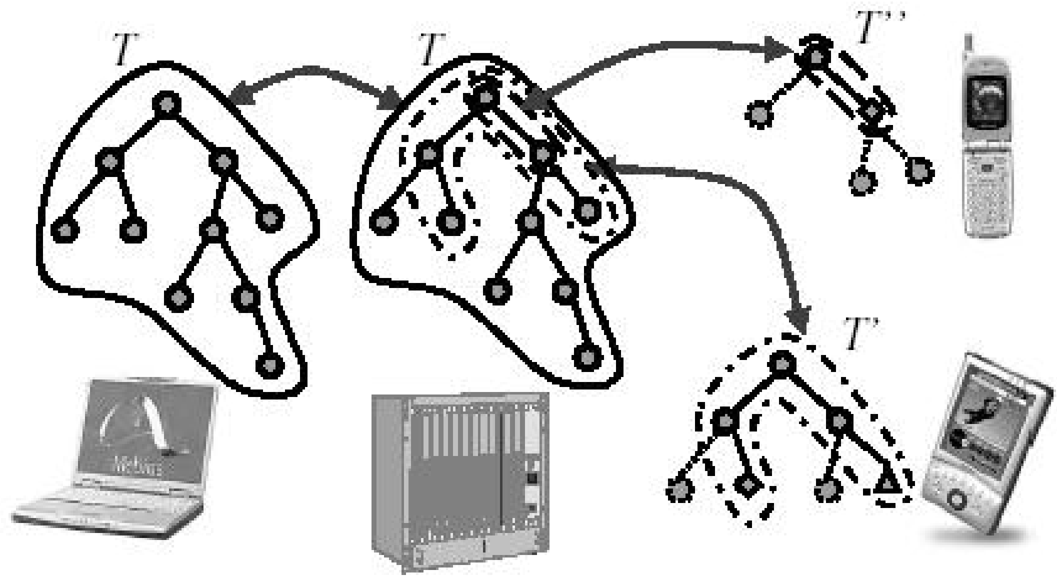

Figure 6. Propagation of structured Object [7]

Figure 6 illustrates the propagation of an object in the OCEAN system. As shown in the figure, while a new object *T* created in a PC is propagated via the server to a mobile phone and a PDA, the propagated objects become subsets of original object *T*. Those nodes whose shapes are not round in the figure depict the simplified or transformed elements as describe above. Although this figure shows the propagation of entire unit of an object, necessary elements are similarly propagated, when a part of an object is modified.

In the OCEAN system, *update messages* carry the information of propagation.

Major processes in this architecture are described below.

- Server

1: Reception of object update

The server updates the master object upon receiving *update messages* from a device. Then, the server specifies appropriate devices according to the profiles, and sends each device proper *update messages*. If there are any inaccessible devices in those devices, the server queues each message.

2: Management of queued messages to devices

The server periodically tries to send queued *update messages* while the queue is not empty.

3: Response to request from device.

The server returns appropriate objects when requested from a device, after authorization and verification.

- Device

1: Update of object by user operation

When a device updates its object according to user operation, the device sends an *update message* to modify the master object of the server. If the connection to the server is not available, the device queues the *update message*.

2: Update of object by message from server

When a device receives an *update message* from the server, it updates its object according to the message and notifies the user of the modification as and when required.

3: Object request from server

The device requests an object to the server when it is not stored in the device.

4: Memory management

The device manages the objects in its memory according to the profiles of objects and situation. When some objects are expired, the device sends an expiration message to the server. If the connection to the server is not available, the device queues the *update message*.

5: Management of queued messages to server

The client periodically tries to send queued message to the server while the queue is not empty.

4. Profiles

The OCEAN system specifies attributes of device profile, and user profile, as well as object profile, to construct mentioned architecture.

Table 1 shows major attributes of these profiles.

Profile	**Attribute**
Object	Object ID, Owner, Type, Importance, Updated-Time, Holding-Device-ID, Notification, Purpose
Device	Device ID, Owner, Type, Memory Capacity, Representation, Connection-capacity, Current-connectivity, Activity, Purpose
User	User ID, Group IDs, Nickname, Name, Address, Age

Table 1: Profiles and Attributes [7]

The significance of some major attributes is described below.

Type attribute describes the category to which the object or the device belongs. Owner attribute describe the owner, and it can be a group ID.

Importance attribute of objects is manually set by the user according to the semantic significance of the object. The object with high Importance attribute propagates fast and hard to expire in each device.

Holding-Device-ID attribute is an attribute for the server to comprehend whether each object is stored in each device or not. Some modification need only propagation to the devices, which are holding the modified object at that point.

Notification attribute of object indicates whether the user have already recognized the last update of the object or not, and should be respectively managed with each user. The propagation of update of un-notified object precedes that of notified update.

Activity attribute of device is device activity. It indicates whether the user is using the device or not. An urgent message will propagate the active device prior to inactive devices.

Connection-capacity attribute describes communication capacity of the connection, which is available at that time, and Current-connectivity attribute indicates whether or not the communication between the client and the server is available.

Group IDs of user is a list of group to which the user belongs. When group object is modified, the target devices can be selected by this attribute with owner attribute of object and device.

Making effective use of those profiles, the OCEAN system keeps liaison of objects stored in each device.

III. User Profiling in Location Management and QoS

Now, there is another interesting aspect of User Profiles in Cellular Networks [6]. Modifications in User Profiles leads us to User Mobility Profiles of customers, which are created and maintained based on prior use at the Mobile Switching Centers (MSC). Network resources like cellular channels and bandwidth are preferentially allocated to priority customers. At the same time, regular users of the network are treated fairly. Simulation results demonstrate the resulting improved QoS like reduced handoff failures to priority users while ensuring good QoS to other customers.

In each cellular area, a database containing the mobility profiles of frequent users of the cellular network is created and maintained. Data contained in these databases provide information about the routes customers frequently use as they traverse the network. This predicts an impending handover to the next and subsequent calls. Thus, channels can be made available at the predicted time of arrival in the new cell(s).

Users often have an ordered list of favored paths; they use alternate routes when the most frequently used route is unavailable or inefficient. Thus road repairs, weather conditions, accidents or any of the causes that result in congestion of the roadways force the mobile user to resort to alternate routes. The statistical data, if collected over a reasonable length of time, can reflect the likelihood of such choices. An important use of the mobility patterns of customers for the service provider is the provisioning of different grades of service based on the frequency of usage of network. Thus, the resources of the network can be allocated to such users in a preferential manner over those who are infrequent or casual users.

IV. Conclusion

We have discussed ‘selective data caching’ in lieu of ‘Data Recharging’. As mobile devices become smaller and more pervasive, consumers of computing technology will be able to perform their tasks using many different platforms throughout their working day. We see the ability to supply these devices with the data they need in a manner as simple as the way they recharge their battery power as being crucial to realizing the full potential of device-based computing.

The OCEAN system discussed in this paper, seems to overcome the inconsistency problems seen in today's information society to provide unique solution to maintain liaison of objects among variety of network devices, and enables utterly new style of communication through objects sharing. The main feature of this system is giving liaison of objects stored in devices through propagating proper contents at proper timing by proper method according to object profile, device profile, and user profile.

I believe, that the field of user profiling technology as discussed in numerous papers, I came across, has a lot of research potential which can incorporate the areas beyond cache management into network management in general.

References

[1] **"Expressing User Profiles for Data Recharging"** Mitch Cherniack, Michael J. Franklin and Stan Zdonik, IEEE Personal Communications: Special Issue on Pervasive Computing, *August 2001.*

[2] **"Generation of user profiles for information filtering — research agenda (poster session)"**, Tsvi Kuflik and Peretz Shoval. Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, p.313-315, July 24-28, 2000, Athens, Greece

[3] **"Profile-Driven Cache Management"**, Mitch Cherniack, Eduardo F. Galvez, Michael J. Franklin and Stan Zdonik, International Conference on Data Engineering (ICDE), 2003, Bangalore, India.

[4] **"Supporting diverse mobile applications with client profiles"**, Laura Bright, Samrat Bhattacharjee, Louiqa Raschid. Proceedings of the 5th ACM international workshop on Wireless mobile multimedia, p.88-95, 2002, Atlanta, Georgia, USA

[5] "**Profile-Driven Data Management**", A Presentation by Mitch Cherniack, Department of Computer Science, Brandeis University

[6] "**User profile-driven resource allocation for cellular networks**", A. N. Rudrapatna, D. P. Agrawal, P. Agrawal, C. Giardina, Conference on Personal Wireless Communications, 1997 IEEE International, p: 178 – 183, 17-19 Dec. 1997, Mumbai, India.

[7] "**OCEAN: Object Communication Environment for Arbitrary Network**", K. Nakagawa, M. Kawakita, K. Sato, M. Minakuchi, O. Tsumori, K. Hanada, T. Chiba, I. Shirakawa, International Conference on Distributed Computing Systems Workshops, 2002. Proceedings. 22nd, 2002 p: 162 –166

[8] **"Modeling Users through an Expert System and a Neural Network**", Moghrabi, C. and Eid, M. S. Computers and Industrial Engineering, 35, (3-4) 583-586, 1998.

[9] **"A comparison of classifiers and document representations for the routing problem**" Schutze, H., Hull, D. A., and Pedersert, J. O. In Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 229-237, 1995.

[10] "**Stereotypes in Information Filtering Systems**", Shapira, B., Shoval, P. and Hanani, U. Information Processing & Management, 33 (3), 273-287, 1997.

[11] "**Efficient Profile Matching for Large Scale Webcasting**", Q. Lu, M. Eichstaedt, and D. Ford, 7th International WWW Conf., Brisbane, Australia, Apr. 1998

[12] "**Personalized Information Delivery: An Analysis of Information Filtering Methods**", P. W. Foltz and S. T. Dumais, Communications ACM, vol. 35, no. 12, Dec. 1992, pp. 51–60.

[13] "**Profile-Driven Cache Management**", Mitch Cherniack, Eduardo F. Galvez, Michael J. Franklin and Stan Zdonik, International Conference on Data Engineering (ICDE), 2003, Bangalore, India. To appear.

[14] "**A framework for expressing and combining preferences**", R. Agrawal and E. L. Wimmers, *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, May 16-18, 2000, Dallas, Texas, USA*, volume 29, pages 297–306. ACM, 2000.

[15] "**Data caching issues in an information retrieval system**", R. Alonso, D. Barbar´a, and H. Garcia-Molina. *TODS*, 15(3): 359– 384, 1990.

[16] "**Information filtering and information retrieval: Two sides of the same coin?**", N. J. Belkin and W. B. Croft, *Communications of the ACM, December 1992*, pages 29–38 1992.

[17] http://oceanstore.cs.berkeley.edu/

[18] "**OceanStore: An architecture for Global-Scale Persistent Storage**", W. K. Edwards et al., Proceedings of the Ninth International Conference on Architectural Support for Programming Languages and Operating Systems, November, 2000.

[19] **"Estimating position and velocity of mobiles in a cellular radio network**", Hellebrandt, M.; Mathar, R.; Scheibenbogen, M., IEEE Transactions on Vehicular Technology, Volume: 46 Issue: 1, Feb. 1997, Page(s): 65 –71

[20] **"Of packets and people: a user-centered approach to quality of service",** Bouch, A.; Sasse, M.A.; DeMeer, H. Eighth International Workshop on Quality of Service, 2000. IWQOS. 2000, 2000 Page(s): 189 –197

[21] **A mobile network architecture with personalized instant information access** Sahinoglu, Z.; Matsubara, F.; Peker, K.A.; Cukier, J. International Conference on Consumer Electronics, 2002. ICCE. 2002 Digest of Technical Papers, 2002, Page(s): 34 –35.

[22] **"A regular path recognition method and prediction of user movements in wireless networks",** Erbas, F.; Steuer, J.; Eggesieker, D.; Kyamakya, K.; Jobinann, K. Vehicular Technology Conference, 2001. VTC 2001 Fall. IEEE VTS 54th , Volume: 4 , 2001 Page(s): 2672 -2676 vol.4