

Towards a Framework for Distributed User Modelling for Ubiquitous Computing

Marcus Specht, Andreas Lorenz, and Andreas Zimmermann

Fraunhofer Institute for Applied Information Technology
Schloss Birlinghoven
53754 Sankt Augustin, Germany
{Marcus.Specht, Andreas.Lorenz, Andreas.Zimmermann}@fit.fraunhofer.de

Abstract. This paper introduces a framework for distributed user modelling in ubiquitous computing. Regarding the requirements in ubiquitous computing, communication details are not fixed per se or even unknown during the system specification phase. Hiding the technology within specialized components allows for unique migration of information between the components and releases system developers from considering specific properties and protocols. Supporting efficient development of applications, the distributed components are able to react both to their environment and to messages received from neighbouring components.

Introduction

In the vision of ubiquitous computing the technology becomes invisible to the user and will be embedded in the objects of our daily life. The user will be surrounded with capabilities for accessing information and services everywhere with many different information devices. These information devices have a direct contact to each other in order to offer common services. The user has *one personal information space* independent of devices and the system manages the information spaces of its users. To have access to this personal information space, distributed components on heterogenous platforms need to exchange information with one another. On the one hand users will use personal devices they carry with them (like mobile phones), on the other hand they will access stationary access points connected to the network. Future applications therefore have to support the information exchange between those “user terminals” and their available sensor networks like location tracking facilities built into a mobile phone and stationary sensors hooked up to the network like environmental sensors. Furthermore for each single user, her actual task and current situation an application has to select the most appropriate device for interaction.

In this sense for future application development for personalized and contextualized applications we expect centralized design-approaches to be confronted with a variety of clients running on heterogeneous devices with different properties, such as personal or wearable devices of the users, embedded technology in displays or printers, and everyday-objects like keys or coffee-machines. Following classical centralized approaches of adaptive information selection and presentation [1], passing

all data from clients to servers for analyzing and centralized decision on adaptation means will cause high network traffic and computational requirements on servers. Furthermore, central user-modelling servers [2] holding all information from registered users will not be applicable for two other reasons: On the one hand, every device of the mobile users will not have permanent access to the server, and on the other hand, the local system will not constantly need all information about the user to make a local decision like contextualization to a current environmental change. Each local component needs to use local information and must be able to integrate in an integrated *representation of knowledge* about the user. In our vision, components for user model acquisition and user model application are equally distributed with the user model itself.

Distributed User Modelling

The provision of personalized information services becomes a complex task in open and distributed environments of mobile users. Mobile applications, in particular in ubiquitous environments, rely on a network of sensors placed within the physical environment and watching indicators for changing situations. On the other hand, the actuators are specialized software-components that process the delivered data or display information snippets on a particular device. The aim of *distributed systems* is to spread the application logic among different parts hosted on different physical devices. For the mobile user, the devices will continuously connect to local networks and therefore will have access to all information available in this network. A centralized solution fails because of its inability to cope with a high degree of change, which requires that the solution is both robust to disruption and self-configurable. For *distributed user modelling* approaches this implies that monolithic user modelling is replaced by distributed user model fragments [3, 6]. The information about the user, i.e. the current state of the user-model, will be merged with all information that can be requested from components reachable in the current context.

In recent research in smart *sensor-networks*, sensors build ad-hoc *self-organizing networks* and deliver requested information on demand. In difference to such sensor-networks, distributed modelling components actually receive pre-processed data from virtual components instead of direct measuring the physical environment. Providing a framework will enable all applications on the devices to check into the network and to make use of the available user-related information. Although this information will just be a cut-out of all potentially known knowledge, it will reflect the current environment by including all relevant information at the current location. In the next section we will illustrate our approach for supporting the information-exchange between ad-hoc networked components.

Framework for Distributed User Modelling

Although there exist approaches for applying mobile agents considered to migrate between devices and always stay with the user, we will focus on supporting migration of information in a network of distributed components. By introducing brokering

components on the local host we create unique communication interfaces for all local components. From our previous work [5, 8] we propose to have components for sensing, modelling, controlling and actuating distributed on different devices. The basic underlying cooperation-approach between those components is *cooperation by information-exchange*. For knowledge-exchange and command-delivery, the components share local message-boards on their hosting devices. The message-boards are managed by specific information-brokers: Locally, the broker provides access to a message-board whereas the information exchange between devices is based on message-sending between the brokers. An important difference to existing solutions is the level of integration of user information as we focus on an information integration of personal and contextual information about a user in the current context and not a general integration which is also critical from a privacy point of view.

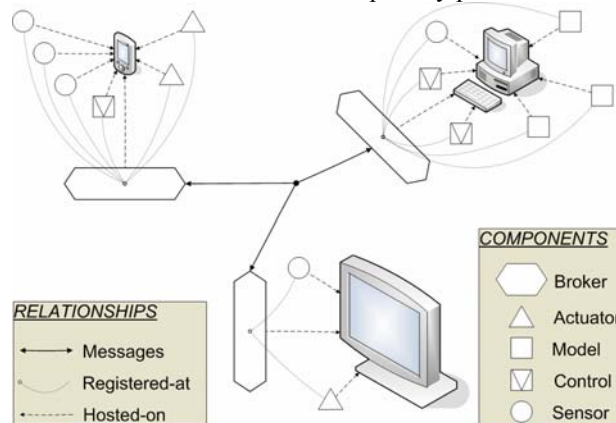


Fig. 1 Distributed User Modelling Components

As illustrated in Fig. 1, different components host physically on different devices. Each component registers at the local board based on defined check-in/check-out mechanisms, announcing what information they provide and what information they request. Potentially, each component can provide any number of attributes and in turn listen to any number of incoming events. The broker manages the list of registered components on the device and cross-links their names in two maps:

1. *Map of local information providers:* The name of the component is mapped to the attributes it provides.
2. *Map of local information listeners:* The attributes are mapped to the names of components who have registered as listeners.

Mapping the components by their names introduces the requirement of unique attributes, whereas it is not required to have one-to-one relationships between attribute names and information providers. Currently, the order of check-in messages defines the active component: The last registered information provider for an attribute will be mapped to be its information provider, overwriting the former one. For sharing information between applications, we will need to install any kind of name-service, and a common user model exchange language (such as *UserML* [4]) supporting the communication between different user adaptive systems.

The Information Flow

When any component hosted on a device sent a message to the local broker, it will be forwarded to all other brokers. For example, the broker at the PDA in Fig. 2 forwards the message to the broker at the desktop via WLAN as well as to the smart-board broker via Bluetooth. The receiving brokers will forward the messages to local receivers, if any registered component was interested in this data.

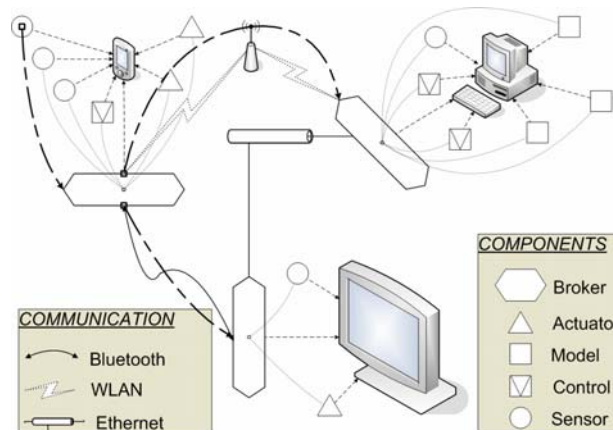


Fig. 2 The Information Flow

Communication among the components usually isn't just a random exchange of messages. Two standard negotiation protocols manage the message flow between the components:

- *Question-Reply*: At check-in, each component has announced what information it delivers. In question-reply protocol, the local component requests missing information from the broker hosting on the local device. The broker broadcasts the request towards all surrounding brokers reachable in the current technical context. Each receiver checks the map of local information providers for a registered component. If such a component was found, the broker forwards the request locally. After receiving the answer, the broker returns the message to the requesting broker, which in turns replies to the question of the local component.
- *Subscribe-Inform*: At check-in, each component has registered as a listener to all information it needs to be informed. If one of the components fires an event, the event-message is sent to its local broker, who forwards the message to all surrounding brokers. Each receiver checks the map of local information listeners and forwards the event-message to each registered component.

Conclusions and Future Work

In this paper we introduced a framework for future user-adaptive application development in ubiquitous computing. We have illustrated our approach to implement a communication-platform hosting distributed components. In this approach, the

knowledge about the user, i.e. the current state of the user-model, will be assembled from many entities reachable in the current context. The devices will continuously connect to local networks and therefore will have access to all information available in this network. Providing a framework will enable all applications on the devices to check into the network and make use of the available user-related information. Usually, the cut-out of potentially available knowledge will reflect the current environment by including all relevant information at the current location automatically. In the current state of work, we specified the platform, defined and implemented messages for check-in/-out, information-request and response, registration of listeners and event-firing, and exceptions. In realisation of our approach, we will implement cooperating agents as active components hosting on the devices and using the communication-framework [7]. As illustrated in this paper, instead of a one-to-one relationship between the user and an agent, sets of agents will be implemented for distributed user-modelling, user-model acquisition and user-model-application.

References

- [1] P. Brusilovsky. Methods and techniques of adaptive hypermedia. *User Modelling and User-Adapted Interaction*, 6(2-3):87-129, 1996.
- [2] J. Fink and A. Kobsa. A review and analysis of commercial user modelling servers for personalization on the world wide web. *User Modelling and User-Adapted Interaction*, 10(2-3):209-249, 2000.
- [3] V. Chepegin, L. Aroyo, P. De Bra, and D. Heckmann. User Modelling for Modular Adaptive Hypermedia. Workshop on Semantic Web for E-Learning, Eindhoven, The Netherlands, pp. 366-371, 2004.
- [4] D. Heckmann and A. Krüger. A User Modelling Markup Language (UserML) for Ubiquitous Computing. In P. Brusilovsky, A. Corbett, and F. de Rosis (eds), *Proceedings of the 9th international conference on user modelling*, pp. 403-407, Johnstown, USA, Springer-Verlag, 2003.
- [5] A. Zimmermann and A. Lorenz. Augmentation of Content with Context Meta-Data. Workshop on Managing Context Information in Mobile and Pervasive Environments, Ayia Napa, Cyprus, 2005
- [6] J. Vassileva. Distributed User Modelling for Universal Information Access. In Stephanidis C. (ed.), *Universal Access in Human - Computer Interaction*, Lawrence Erlbaum: Mahwah, N.J., 122-126, 2001.
- [7] A. Lorenz. Agent-Specification for Distributed User Modelling for Ubiquitous Computing. Workshop on Decentralized, Agent Based and Social Approaches to User Modelling, Edinburgh, 2005, to appear
- [8] A. Zimmermann and A. Lorenz. Creating Audio-Augmented Environments. *Journal of Pervasive Computing and Communication*, 1(1):31-42, 2005.