Smart User Assistance Based on Dynamic Model Composition*

René Zilz and Peter Forbrig

University of Rostock, Department of Computer Science
Albert-Einstein-Str. 21, 18059 Rostock, Germany
{rene.zilz,peter.forbrig}@uni-rostock.de

Abstract. Ubiquitous computing environments are complex systems since they support a variety of different input modalities such as voice, touch and keyboard/mouse input. Especially inexperienced users (e.g., elderly or novel) might be confused with the abundance of interaction opportunities. This paper presents an approach to specify smart user assistance for mobile devices. The mobile application shows performable tasks, explains modalities and commands, and provides interactive exercises in order to get the user familiar with the system interaction. The complexity of the interactive exercises and the selection of most suitable modalities are adapted based on user characteristics. The advantage is that users become acquainted with the use of different modalities step by step.

Keywords: Human Computer Interaction, Automatic System Adaptation, Ubiquitous Computing, User Assistance, Task Models.

1 Introduction

In [9], Weiser introduces the paradigm of ubiquitous computing. It describes a world where information is available at any time and any place. Current ubiquitous computing environments are mostly complex systems that are composed of different devices and are connected in heterogeneous networks. The devices are embedded in the surrounding environment and the user should not be aware of them. Frequently, multimodal interfaces are used in order to enable user interaction with different modalities (e.g., voice, touch and keyboard/mouse input). However, while it is already difficult to specify traditional high quality interfaces, the problem in multimodal user interface design is becoming more critical. Especially inexperienced users (e.g., elderly or novel) might be confused with multimodal interfaces since they might have several usability problems. For instance, the user needs to know which tasks are performable and which modalities are available in a certain situation. Additionally, modalities can be combined and each modality can have several commands.

Due to this complexity, we set out to assist the user in order to simplify the handling process of the system.

* Supported by a grant of the German National Research Foundation (DFG), Graduate School 1424, Multimodal Smart Appliance Ensembles for Mobile Applications (MuSAMA).

A. Marcus (Ed.): Design, User Experience, and Usability, Pt I, HCII 2011, LNCS 6769, pp. 706–714, 2011.
© Springer-Verlag Berlin Heidelberg 2011

The proposed assistance can be deployed in lightweight devices (e.g., PDAs or mobile phones) and tries to get inexperienced user familiar with the system interaction. The assistance is able to list performable tasks, to describe interaction modalities and commands and to provide interactive exercises that are adapted according to the level of user experience. Furthermore, for each task the modalities that fit to the user's preference best are selected.

The rest of the paper is structured as follows: In Section 2 we give an overview of related work. Section 3 is concerned with the acquisition of a user profile that includes measurement of user characteristics (experience and preference). Section 4 discusses the idea and specification of smart user assistance for mobile devices. Finally, a summary and future work is presented in the last section.

2 Related Work

In recent years, the generation of assistance in interactive systems has been well recognized. Publications in this area present a vast amount of application fields. Wandke [8] lists amongst other things online help systems in human-computer interfaces, automatic functions in real-time human-machine systems, assistive technology for handicapped people, smart homes, and personal digital assistants. In the following, we aim at presenting some of these systems and compare them with our approach.

Thimbleby and Addison present a system called HyperDoc in [7]. HyperDoc is primarily used for the design and analysis of user interfaces and their manuals. The interactive system as well as its corresponding documentation is specified as finite state machine. Thereby, each state has an associated natural language description and each transition has an associated button name. Due to this specification the system is able to provide answers for user questions. For instance, in order to answer the question "How do I?" the best route from the current state to the required state is determined. By the use of weighted transitions adaptive help can be given. For that reason, weights are adapted based on the current user behavior or previous training with the system. However, finite state machines do not seem to be suitable to model interactive systems. One problem is the large number of states in such systems. Subsequently, simple transition networks are not able to represent the hierarchical structure of an interactive system [3].

There are several assistive systems for handicapped people that apply media adaptation techniques in order to present relevant information with other output modalities. The systems differ essentially by the technique of media adaptation. While a lot of systems are using fixed rules, there are some systems that consider such a process as optimization problem. In the following, the AVANTI [5] system that uses a rule-based media adaptation approach and the CUMAPH [2] system that uses an optimization based approach [1] are presented.

AVANTI [5] is able to display interactive views of adaptive multimedia Web documents. The authors distinguish between adaptability and adaptivity. Adaptability denotes adaptations that take place during the initiation of an interaction (e.g., preferences and abilities), whereby adaptivity denotes adaptations that occur at runtime (e.g., changing user characteristics, situations or familiarity with specific tasks).

AVANTI provides both techniques that allow the system to tailor the user interface according to abilities, skills, requirements and preferences of the user. AVANTI is also capable to support handicapped people (e.g., usage of voice input for visually impaired users). Besides adaptive web documents, traditional web documents can be displayed as well.

Habieb-Mammar et al. propose their system CUMPAPH (Cognitive User Modeling for Adaptive Presentation of Hyper-documents) in [2,6]. CUMAPH is able to adapt hyper documents that are specified as XML-documents. XML-documents consist of several blocks and each block has several elements that can be presented with different media. By computing a compatibility matrix that includes all possible combinations of elements, the system is able to select the most suitable combination of interactive elements. Therefore, the authors have two metrics: the first metric is used to compute the best combination that fit to a cognitive user's profile (generated by a sequence of interactive exercises); the second one favors a combination of media. The combination for which the sum of both metrics is the greatest one is selected.

The problem of media-adaptation is quite similar to ours. However, we focus more on input modality assistance and less on output modality assistance and in contrast to both systems; our main objective is a stepwise familiarization with the abundance of interaction opportunities. Thus, we are not tailoring the system interface itself. Instead, assistance consists of explaining modalities and corresponding commands, and providing adaptable interaction exercises to get the user familiar with the system interaction. The interactive exercises are specified in terms of predefined task models and are substituted according to the provided user information. As a consequence, inexperienced users can become acquainted with the system interaction step by step.

3 User Profile Acquisition

User profiles are employed to store personal characteristics of different users. Based on those user profiles, we are able to adapt the assistance. At the moment, we are measuring two characteristics: *user preference* and *user experience*.

Preference is used in order to determine which modalities have a higher priority since a task can be performed with several input modalities. A stammering user for example, might want to assign voice input a lower priority than people with normal speech. We measure user preference in the following way: if the user handles the mobile application for the first time he/she has to create a profile and adjusts scales with different input modalities. The value ranges of the scales are from zero to ten, whereby one means the lowest priority and ten the highest one. The value zero is used to disable a modality.

Since we need an indicator for determining the complexity of assistance, *user experience* must be regarded as an important characteristic. For example, novel users need assistance for one modality (e.g., write a text with voice input) and basic commands (e.g., start text input or end text input). However, advanced users and experts need assistance for several modalities (e.g., write a text with voice and touch input) and complex commands (e.g., change text properties).

The measurement of user experience takes place as follows: our system stores all modalities and commands in user profiles that are applied by users in order to perform a task. Moreover, if the user is working with the system for the first time, he/she can select well-known modalities and commands. Based on these data, we assign each user into one of the groups shown in Table 1.

Table 1. Categories of Users and their Experience

User Category	Experience
Novel	User knows neither modalities nor commands to perform the task
Experienced	User knows one modality and corresponding basic commands to perform the task
Advanced	User knows one modality and all corresponding commands to perform the task
Expert	User knows all modalities and corresponding commands to perform the task

For example, the task *Write Text* can be performed with voice, touch and keyboard/mouse input. The most suitable combination according to the preference could be: voice input in order to write a first draft and correct possible errors with keyboard/mouse input. Basic voice commands (e.g., start input or end input) are used to begin and start the text input. As soon as the user is familiar with this concept, we adapt the assistance with respect to higher complexity (e.g., several modalities or complex commands).

4 Specification of Smart User Assistance

4.1 Overview

Section 3 presented the acquisition of a user profile. This section describes a method to specify a smart mobile assistance system in ubiquitous computing environments. The main goal of our assistance is a stepwise familiarization of users with several input modalities. Therefore, we are using interactive exercises that are specified in terms of predefined task models and are substituted according to the user profile. Fig.1 depicts a Concur Task Tree (CTT) [4] model that shows how our mobile assistance is given. CTT is a graphical notation to model user activities and distinguishes between different kinds of tasks (e.g., abstract, user, interaction and application). Subsequently, tasks are hierarchical decomposed and linked with each other by temporal operators (e.g., enabling, concurrent etc.) in order to specify the relationship between them. In Fig.1, the abstract task "Select Performable Task" displays all performable tasks according to the context. After that, the user selects the task for which assistance is desired.

The application task "Select Assistance" chooses the most appropriate interaction modality and corresponding interactive exercise to perform the task. Based on this selection the interactive exercise is composed and added dynamically.

Thus, the process of giving assistance can be divided into three parts: determination of performable tasks, selection of most appropriate input modalities and dynamical model composition. Each of these parts is described in the following subsections.

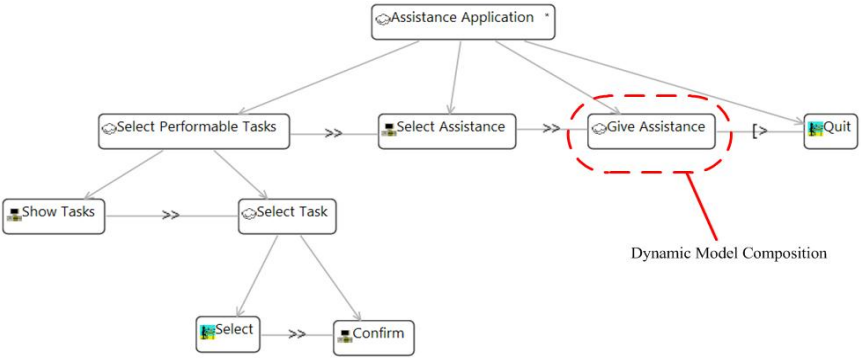

Fig. 1. Specification of Assistance

4.2 Determine Performable Tasks

The process of giving assistance starts with the determination of performable tasks. To accomplish this, we are modeling the ubiquitous computing system with the collaborative task modeling language (CTML) [10]. CTML is a task-based specification language for ubiquitous computing environments. In CTML, user roles are expressed as task-models and the behavior of each role can be expressed by an associated collaborative task expression [10]. By analyzing of these expressions, we are able to determine all performable tasks for one user.

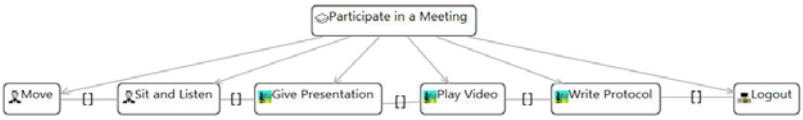

Fig. 2. User Role Participate in a Meeting

The task model in Fig. 2 shows a possible user role for a meeting in a smart meeting room. Possible task constraints (preconditions) could be: the user has to be in the presentation zone to perform the task "Give Presentation"; A DVD has to be inserted into the player to perform the task "Play Video".

The user has to have left the room to perform the task "Logout". Thus, if the user is in the presentation zone, the performable tasks are: "Give Presentation" and "Write Protocol".

If a DVD is in the DVD-Player and the user is in the presentation zone, the performable tasks are: “Give Presentation”, “Write Protocol” and “Play Video”. After determination of all performable tasks, the user chooses the task for which assistance is desired.

4.3 Selection of Input Modalities

After determination of the task, it is necessary to select which modalities should be used to perform the task. There are different solutions to reach this goal. One solution is that the user selects the desired modalities himself/herself. However, this approach is not applicable in each case since some users do not know which input modalities are most suitable. A better solution would be to let the developer decide which modalities should be used to perform a task. Nevertheless, predefined modalities might be unacceptable for handicapped people. Speech-impaired people, for example, are not able to interact via voice input. Consequently, user preferences have to be considered as described in Section 3 and to deselect a modality according to the user profile. For example, if the user deselects speech in the user profile (priority zero); all modality-combinations that include speech-input are deselected. After that, the most suitable modality chosen by the developer is selected unless the user has specified a preference for one of the modalities.

4.4 Dynamic Model Composition

Since our assistance should dynamically adapt interactive exercises according to the user experience, we need for each user category an exercise of different complexity. Table 2 gives an overview what kind of interaction exercise is given for each user category.

Table 2. User Categories and the corresponding Complexity

User Category	Experience
Novel	Interactive exercise for selected modality and corresponding basic commands
Experienced	Interactive exercise for all commands of the selected modality
Advanced	Interactive exercise for new modality and corresponding commands
Expert	Interactive exercise using different modalities

The interactive exercises are specified in terms of predefined task models and substitute the task “Give Assistance”, as illustrated in Fig. 1.
For example, the user asks for assistance for the task “Write Protocol” (Fig. 2).

Available modalities in order to perform the task are the following ones:

- Speech input and correction of possible text errors with mouse/keyboard input
- Stylus input
- Keyboard/mouse input

Additionally, there is the constraint that the modalities speech and stylus input cannot be used concurrently. Let us assume that the stylus input is preferred by the user. It is also assumed that it is a novel user. For this case, a possible interaction exercise is depicted in Fig. 3. At first, the user gets an explanation of the interactive exercise. This includes a general explanation how to use the stylus and basic commands (e.g., open and close a text input). After that, the user has to perform an interactive exercise. This means, that he has to pick up a stylus and to write a text. For writing of a text the user has to open a text input writes something and closes the input.

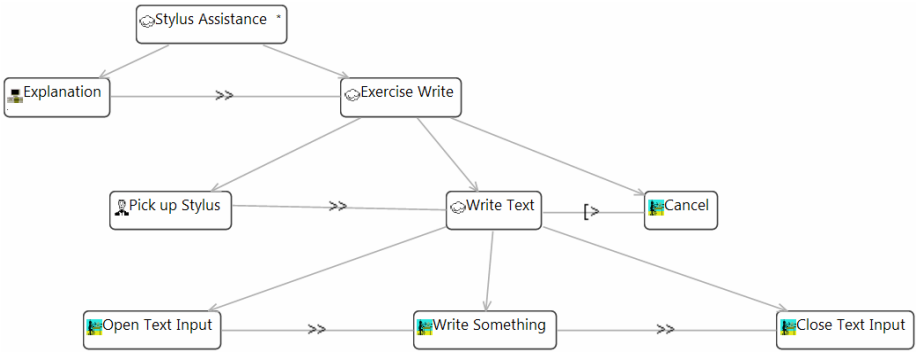

Fig. 3. Specification of Assistance for Novel Users

As soon as the user was able to fulfill the exercise and is familiar with this interaction concept, he is regarded as experienced. Experienced users are able to perform tasks with one modality (the most suitable according to their user profile) and they know all corresponding commands. Fig. 4 shows how further commands are assisted (e.g., change text properties) and how the interactive exercise is adapted.

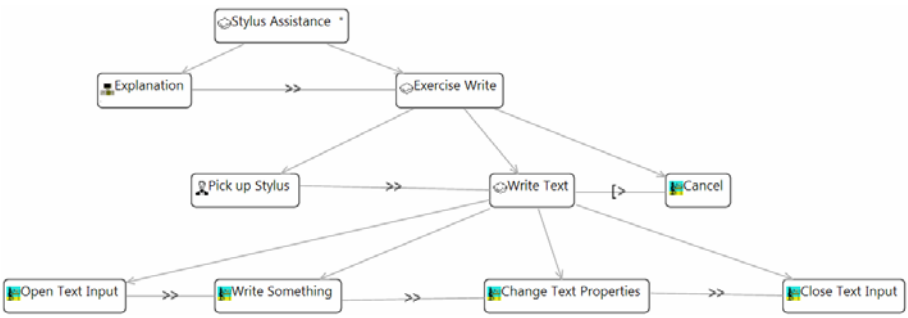

Fig. 4. Specification of Assistance for Experienced Users

This adaptation process is continued with advanced users and experts as well. Advanced users perform the task with other modalities than stylus (e.g., speech input and mouse/keyboard input).

Finally, experts get assistance while using different modalities to perform the task. Fig. 5 and Fig. 6 show the assistance for advanced users and experts.

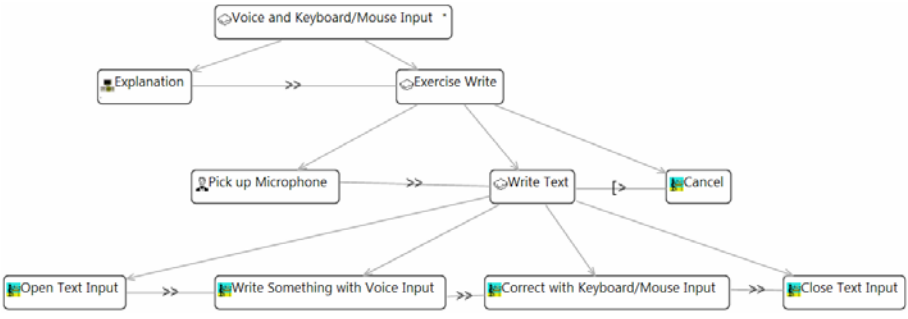

Fig. 5. Specification of Assistance for Advanced Users

To become familiar with voice input that can be corrected with keyboard and mouse there will be an explanation how this can be reached. After the explanation there is again a task a user has to perform. After fulfilling the task the user is expected to have learned the corresponding interaction with the modalities in focus. If there is a problem the exercise can be interrupted and the assistance can start again. This can be seen by looking at the root task that is iterative ("*").

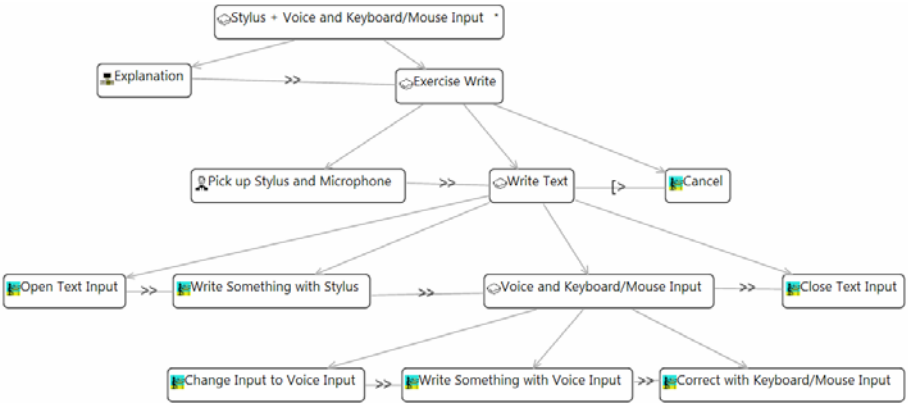

Fig. 6. Specification of Assistance for Experts

Again after an explanation an exercise has to be performed. For this model it is important to note that the hierarchy for the three tasks "Change Input to Voice Input", "Write Something with Voice Input" and "Correct with Keyboard/Mouse Input" was introduce for graphical reasons only. Six tasks at the same level made the model unreadable.

5 Conclusion

The goal of the approach described in this paper is a specification of smart user assistance for mobile devices in ubiquitous computing environments. The assistance provides support in using different input modalities especially for inexperienced user. At first, our assistance determines performable tasks and corresponding input modalities. The modalities are favored according to the given user preference. Thereafter, an interactive exercise is explained and finally performed by the user. The interactive exercises are specified in terms of predefined task models and are substituted according to the provided user experience. As soon as the experience is increasing, the task models are substituted by more complex interactive exercises. The proposed system specification is of particular interest for inexperienced users (e.g., elderly or novel) since they become acquainted with the system step by step.

In the future, we want to evaluate our system based on user studies. We assume that the results support our thesis that changing the underlying models during run time allows a personalized support that is appreciated by users. We assume that interaction with multimodal systems becomes easier with this approach and thus an improvement in the system's usability and accessibility can be reached.

References

1. Bunt, A., Carenini, G., Conati, C.: Adaptive content presentation for the web. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 409–432. Springer, Heidelberg (2007)
2. Habieb-Mammar, H., Tarpin-Bernard, F.: CUMAPH: Cognitive User Modeling for Adaptive Presentation of Hyper-documents. An Experimental Study. In: Adaptive Hypermedia and Adaptive Web-Based Systems, pp. 718–736. Springer, Heidelberg (2004)
3. Kieras, D., Polson, P.G.: An approach to the formal analysis of user complexity. International Journal of Man-Machine Studies 22(4), 365–394 (1985)
4. Paterno, F.: Model-based design of interactive applications. Intelligence 11(4), 26–38 (2000)
5. Stephanidis, C., Paramythis, A., Sfyrakis, M., Stergiou, A., Maou, N., Leventis, A., Paparoulis, G., Karagiannidis, C.: Adaptable and adaptive user interfaces for disabled users in the AVANTI project. In: Campolargo, M., Mullery, A. (eds.) IS&N 1998. LNCS, vol. 1430, pp. 153–166. Springer, Heidelberg (1998)
6. Tarpin-Bernard, F., Habieb-Mammar, H.: Modeling elementary cognitive abilities for adaptive hypermedia presentation. User Modeling and User-Adapted Interaction 15(5), 459–495 (2005)
7. Thimbleby, H., Addison, M.: Intelligent adaptive assistance and its automatic generation. Interacting with Computers 8(1), 51–68 (1996)
8. Wandke, H.: Assistance in human–machine interaction: a conceptual framework and a proposal for a taxonomy. Theoretical Issues in Ergonomics Science 6(2), 129–155 (2005)
9. Weiser, M.: The computer for the 21st century. Scientific American, pp. 94–104 (1991)
10. Wurdel, M., Sinnig, D., Forbrig, P.: CTML: Domain and Task Modeling for Collaborative Environments. Journal of Universal Computer Science 14(19), 3188–3201 (2008)