Adaptive User Interfaces for Ubiquitous Access To Agent-based Services

Joos-Hendrik Böse and Sebastian Feuerstack

Distributed Artificial Intelligence Laboratory
Faculty IV Electrical Engineering and Computer Science
Technical University of Berlin, Germany
EMail: {joos,sfeu}@dai-lab.de

Abstract This paper outlines an approach to provide adaptive user interfaces [5] for services provided by intelligent software agents in smart environments. After motivating the need for adaptive user interfaces, we propose how to achieve adaptivity at different levels by describing the basic ideas behind our implementation.

Motivation

The vision of Ubiquitous Computing proposes smart environments as a promising application domain for intelligent software agents, because they are designed to interact and cooperate in dynamic environments. We build smart environments by grouping intelligent agents together according to physical locations like rooms or buildings. These agent communities provide services for human users according to their location like a radio, stationary communication services or even a light switch. Based on the context[1] of a human user, only a subset of available services within a smart environment makes sense for the human user. We propose an approach to offer adaptive interaction between agents and human users in smart environments.

Approach

The process of adaption takes place at different levels of the interaction. These include the *functionality adaption* and the adaption to the *user's device*. The latter one is split into *protocol adaptation* and *content*[2] *adaptation*.

Within the following subsections we describe how we achieve the different levels of adaptation.

Functionality adaptation

By functionality adaptation we understand the selection of functionality that is useful for a human in his current context. This level of adaptation is implicitly considered by the agent architecture [6] that uses roles as a central concept for interaction between entities. A role assigns functionality to the entity adopting the role. We distinguish between primitive roles and complex roles, which aggregate functionality offered by several sevices.

Human users can take roles within the agent community with the help of a Mediative Agent (MA) that encapsulates a specific role. Different roles result in different UIs. For example a role for normal users will result in an other user interfaces (UIs) than an administrative role.

[1] By user context we refer to the location, identity, activity and the state of a user.

[2] By content we refer to semantical described objects, that are part of the service's communication and therefore need to get presented to the human user.

We propose a pragmatic solution, that enables the user to select his current role on his own, because the role of a human user may not be easily identified.

The process of choosing among a set of roles is an elementary different approach, than the FIPA proposes in it's Human-Agent Interaction [3] specification, because it conforms to a "pull"-based model. Whereas the FIPA proposes a "push"-based model, which allows an agent to proactively contact the human user. This results in an extensive user preference modeling, because the proactive agent has to learn about the current user's role each time it wants to interact with him.

We assume that both, a "push"-based model, that enables agents to proactively contact the human user, as well as a "pull"-based model, are needed that allow the user to provide the MA additional information regarding his current context, resp. the type of role he wants to interact as.

User device adaptation

The most desired effect of adaptation on the presentation level is *device independence* which means that UIs can be adapted to the capabilities of the user's device. To adapt a UI to a device we differ between the adaptation of the UI protocol (e.g. HTML, WML or VoiceXML) and the adaptation of the interaction content.

Protocol Adaptation A common approach is to use an abstract UI description that can be transformed to different presentation forms. We supply the MA with such an abstract UI description that is transformed into a UI according to general and specific transformation rules. The general transformation rules can be applied to any service and therefore ensure device independence. Because UI generated using general rules are often lack of usability, specific rules can be defined and assigned to a MA. Specific rules define how a service UI should be presented on a certain device.

The process of transforming the abstract UI description according to general and service specific rules is distributed among several roles (resp. agents). While the Multi-Access Agent (MAA) receives the request from the user's device and sense its capabilities, Transformation Agents take care of the UI generation.

The UI description elements are defined to support aggregations of services into one UI and to split up a UI into sub UIs that can be assigned to different devices.

Content Adaptation On the content level, objects taking part in the interaction, which are semantically described by the use of ontologies, should be presented. These objects have to be adapted according to the human user preferences. Beneath the semantical selection between a wide range of speech-only, text-only and various multimedia-based presentations, the capabilities of the device, that is used by the human user to participate in the interaction has to be taken into consideration.

We propose an agent called Media Provider (MP), that is able to store different types of media data, each linked to an instance of an ontology that specifies the content. In an interaction process between a human user and an agent, speechacts, that contain complex objects (instances of ontologies) get announced to a set of Media Providers in order to that they supply a suitable presentation of content. Constraints describe the current context of the user, his current preferences and the capabilities of the user's device and influence the selection-process in the case that various presentations of content have been found.

A Media Provider is able to resolve inheritance-based relationships and is therefore able to associate presentations regarding an ontology instance it does not directly know.

Conclusion

This work proposes an architecture that provides device independent access to adaptive user interfaces for agent-based services. Even through this architecture has been sucess-

fully implemented [1, 2], we noticed that the support for multi modal interfaces[4] and the aggregation of different UI must be considered in more details and still need further work.

References

[1] Joos-Hendrik Böse. Adaptive user interfaces for ubiquitous human-agent interactions. Master's thesis, Technical University of Berlin, 2002.

[2] Sebastian Feuerstack. An architecture for ubiquitous service access in distributed systems. Master's thesis, Technical University of Berlin, 2002.

[3] Foundation for Intelligent Physical Agents. FIPA 98 Part 8 Version 1.0: Human-Agent Interaction Specification, 1998.

[4] W3C Multimodal Interaction Working Group. Multimodal interaction activity. http://www.w3.org/2002/mmi/, 2002.

[5] Matthias Schneider-Hufschmidt Thomas Kühne Uwe Malinowski. State of the art in adaptive user interfaces. In *Adaptive User Interfaces*, Human Factors in Information Technology, pages 11–48, Amsterdam, 1993. North-Holland, Elsevier Science Publishers B.V.

[6] Ralf Sessler. Building agents for service provisioning out of components. In *AGENTS 01, Proceedings of the Fifth International Conference on Autonomous Agents*, 2001.