Managing Presentations in an Intelligent Environment

Christoph Stahl, Michael Schmitz, Antonio Krüger, Jörg Baus
Saarland University, Stuhlsatzenhausweg 36.1
66123 Saarbrücken, Germany.
{baus, krueger, schmitz, stahl}@cs.uni-sb.de

ABSTRACT

Intelligent environments enable users to receive information from a variety of sources, i.e. from a range of displays embedded in those environments. From a services perspective delivering presentations to users in such an environment is not a trivial task. While designing a service it is, for example, not clear at all which displays will be present in the specific presentation situation and which of those displays might be locked by other services. It is further unclear if other users are able to see the presentation, which could cause problems for the presentation of private information in a public space. In this paper we propose a solution to this problem by introducing the concept of a presentation service that provides an abstraction of the available displays. The service is able to detect conflicts that arise when several users and services try to access the same display space and provide strategies to solve these conflicts by distributing presentations in space and time. The service also notifies the user by a alarm signal on a personal device each time a presentation is shown on a public display in order to disambiguate content between multiple users.

Keywords

Smart Environments, Public Displays, Shared use of multiple services by multiple users using multiple devices

INTRODUCTION

The project REAL is concerned with the main question: How can a system assist its user in solving different tasks in an intelligent environment? We have developed two applications, which proactively provide a user with shopping assistance and navigational aid in response to their actions within the environment, minimizing the need for a traditional GUI, but the user can also use their PDA to formulate multimodal requests using speech and gesture combined. System output, such as directions and product information, is presented to the user in a flexible fashion on suitable public displays nearby to the user, based on the requirements of the content and spatial knowledge about the positions of the displays and the user. In such a scenario of multiple users, applications, and displays, conflicting presentation requests are likely to arise and need to be resolved.

In the following, we briefly describe the architecture of our intelligent environment before we explain the presentation service in detail.

THE SUPIE ARCHITECTURE

In order to investigate intelligent user interfaces based on implicit interaction and multiple devices, we have set up the *Saarland University Pervasive Instrumented Environment* (SUPIE). Its architecture has been designed for the seamless integration of the shopping assistant *ShopAssist* [5] and the pedestrian navigation system *Personal Navigator* [4]. It is organized in four hierarchical layers, which provide in bottom-up order: blackboard communication (based on the *EventHeap* [3] tuplespace), positioning and presentation services, knowledge representation and the applications. The presentation service will be explained in more detail in the next section.

Knowledge Layer

The knowledge layer models some parts of the real world like an office, a shop, a museum or an airport. It represents persons, things and locations as well as times and events. The ubiquitous world model *UbisWorld*[1] describes the state of the world in sentences made of a subject, predicate and object. A hierarchical symbolic location model represents places like cities, buildings and rooms, and serves as a spatial index to the situational context. In order to generate localized presentations and navigational aid, the positions of the user, the landmarks and the displays have to be known. Therefore the symbolic model is supplemented by a geometric location model, which contains coordinates of the building structure, landmarks, beacons and displays, and even their viewing angles and distances, if necessary.

Application Layer

Currently three applications employ the presentation manager in order to present information to the user on public displays. The *shopping assistant* provides product information and personalized advertisements to the user. As the user interacts with real products on the shelf, their actions are recognized by a RFID reader and in response, the assistant proactively serves product information to the user on a display mounted at the shopping cart. A wall-mounted display allows the user to browse through the vendor’s product website, which opens automatically.

The *navigation application* runs on a PDA and picks up beacon signals, which are send to the positioning service

[1] www.u2m.org

and result in a location identifier. The handheld provides a visualization of the location on a graphical map and offers navigational aid by arrows and speech synthesis. It additionally utilizes the presentation service in order to present directions to the user on nearby public displays.

Another application welcomes the user as they enter the shop by a steerable projection of a virtual character.

More applications, such as the posting service *PlasmaPoster* [1] or the messaging service *IM Here* [2], could also easily benefit from the presentation service and run simultaneously in the environment.

MANAGING PRESENTATIONS ON MULTIPLE DISPLAYS

In a public space with various displays we assume that a number of applications are running simultaneously and concurrently attempting to access display resources. Therefore we favour World Wide Web technology, such as HTML and Flash, for presentations that still allow simple form-based interaction, instead of running custom applications on the public displays. Whereas canonical conflict resolution strategies could be first come, first served or priority based assignments of display resources, we focus on rule-based planning: Presentation strategies are modelled as a set of rules that are applied to a list of available displays and queued presentation requests. These rules generate plans at runtime, which define where and when presentations are shown. Applications post their presentation requests on the blackboard, which include the following mandatory and optional(*) arguments:

Source	URL of the content
Destination	Display *or* location *or* user
Type	Image, text, video or mixed
Expiration Deadline	e.g. in 30 minutes from now
(Minimum Display Time)*	e.g. 60 seconds
(Minimum Display Size)*	Small, medium, large
(Minimum Resolution)*	e.g. 800x600
(Audio Required)*	Yes, No
(Browser Input Required)*	Yes, No

Based on these requests, the presentation service plans the presentation schedule for all displays in a continuous loop:

1. Generate a list of feasible displays based on their properties and spatial proximity to the users' location.
2. Sort the list according to: idle state, minimum requirements (e.g. size), general quality and release time.
3. Resolve conflicts by queuing requests (division by time) and splitting displays (division by screen space).

This set of rules provides coherent presentations in public spaces and resolves conflicts by dividing display resources in time and space: Presentations are scheduled according to their deadline requirements and are delayed until resources are available (time). Screen space is shared if an appropriate device is available such that presentations are rendered simultaneously on the same screen in different windows (space).

Privacy issues require additional rules: Each user can specify contents as private within the user model, for instance all navigational aid. The presentation service would now simply remove displays in the first step which can be seen by other users.

Conflicts that arise from multiple users interacting concurrently can be handled by the same strategies. However from the users' perspective, it is crucial to be aware of presentations intended for them and to avoid confusion caused by similar presentations for other users. Therefore the presentation service notifies the user via a personal device by an alarm signal (e.g. mobile phone vibration), that is synchronized with the appearance of the presentation on a public display. If no such notification device is available, the presentation service can automatically tag the content with a personal graphical icon or sound that is stored in the user model.

IMPLEMENTATION

We have implemented Internet Explorer-based presentation clients for Windows CE and 2000, running on various public displays, including PocketPCs as interactive office doorplates, a tablet PC connected to a shopping cart, and wall-mounted panel PCs and plasma displays.

The presentation service currently resolves conflicts by considering the deadlines combined with priorities. It matches the display positions with the user's current range of perception, and presentations are queued until displays become available or multiple browser windows are opened (division in time and space). A rule-based planner is currently under development in Prolog.

REFERENCES

1. Churchill, E., Nelson, L., Denoue, L., Helfman, J. and Murphy, P. Sharing Multimedia Content with Interactive Displays: A Case Study. In *Proceedings of Designing Interactive Systems (DIS2004).* ACM Press, 2004.
2. Huang, E., Russel, D. and Sue, A. IM Here: Public Instant Messaging on Large, Shared Displays for Workgroup Interactions. In *Proceedings of CHI 2004*, pages 279-286. ACM Press, 2004.
3. Johanson, B. and Fox, A. The event heap: A coordination infrastructure for interactive workspaces. In Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications, page 83. IEEE Computer Society, 2002.
4. Krüger, A., Butz, A., Müller, C., Stahl, C., Wasinger, R., Steinberg, K. and Dirschl, A. The Connected User Interface: REalizing a Personal Situated Navigation Service. In Proceedings of the International Conference on Intelligent User Interfaces (IUI 2004), pages 161-168. ACM Press, 2004.
5. Wasinger, R., Schneider, M., Baus, J. and Krüger, A. Multimodal Interaction with an Instrumented Shelf. In Artificial Intelligence in Mobile Systems (AIMS) 2004, number ISSN 0944-7822, pages 36-43, 2004