
RelateGateways: A User Interface for Spontaneous Mobile Interaction with Pervasive Services.

Dominique Guinard

Computing Department
InfoLab21, Lancaster University
Lancaster LA1 4WA, UK
dguinard@gmipsoft.com

Sara Streng

Institute for Computer Science
Ludwig-Maximilians University
Amalienstrasse 17
80333 Munich, Germany
sara.streng@gmail.com

Hans Gellersen

Computing Department
InfoLab21, Lancaster University
Lancaster LA1 4WA, UK
hwg@comp.lancs.ac.uk

Abstract

RelateGateways is a user interface and system that supports end-user discovery of and interaction with services dynamically encountered in their immediate environment. The interface is designed for mobile devices and indicates available services through small widgets arranged at the periphery of the screen based on a compass metaphor.

Keywords

Mobile interaction, spatial user interface, ubiquitous computing, relative positioning, pervasive services.

ACM Classification Keywords

H5.2. Interfaces and Presentation: User Interfaces - Interaction.

Introduction

An expectation at the core of ubiquitous computing is that mobile users will be able to make spontaneous use of physically manifest services in their environment. Ad hoc networking technologies already facilitate, in principle, the spontaneous connection of a user's personal device to encountered services. However it remains difficult for users to understand what services

are available in their environment, how networked-discovered services relate to physically present devices, and how the actual interaction can be initiated. Related work has, for example, advocated user interfaces that put users' in the driver seat for associating services [1], used location to associate services with devices [2], and proposed spatial representations in the user interface for interaction across devices [3].

In this paper we introduce a user interface concept that, like our previous work discussed in [3], assumes a sensing technology that provides relative positioning information such that the spatial relationships between a mobile user's device and devices encountered in their environment are established and tracked in real-time. The *RelateGateways* interface uses the positioning information to dynamically and contextually display the services the user can interact with in their immediate environment. The devices providing these services appear as gateways in the interface, taking the form of small widgets at the periphery of the user's screen as they move through an environment. The gateways are spatially arranged around the screen based on a compass metaphor. They serve as "access point" for association and interaction with services, and are integrated with a mobile desktop interface to support direct manipulations such as "drag-and-drop" of documents to a service.

The RelateGateways Concept

For a first exploration of our interface concept we have set up a very basic scenario with three devices (a printer, a display and a keyboard) at different locations in a larger office room, and used an OQO handheld PC running Windows XP as mobile user device (Fig. 1). Each of the three devices provides a number of

services. For instance, with the keyboard (1) users can type text that is redirected to the handheld PC, the display (2) supports presentation of transmitted documents, and the printer (3) offers standard print services.

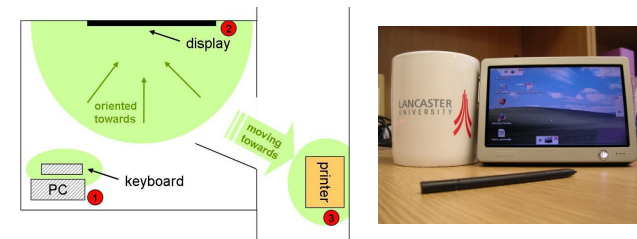


Figure 1. Left: location of the services. Right: the OQO running the RelateGateways application.

The gateways are interactive areas, effectively widgets, on the edges of the screen, indicating where a device (providing services) is located. The gateway's position depends on the direction of the device. For example if the user is standing in front of a printer, he will see a printer gateway on the top of his screen. If the printer is on the left, the gateway will appear on the left. Thereby, the RelateGateway interface has the functionality of a compass.

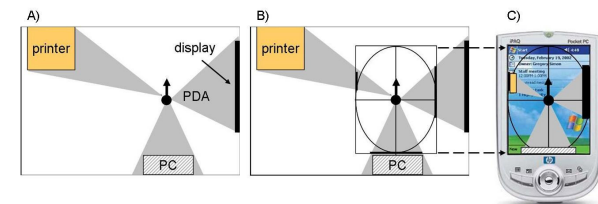


Figure 2. The application as a compass.

This is illustrated in more detail in Figure 2. In the left picture a user with their PDA is indicated by a black spot. The arrow represents the user's orientation. The directions to the devices are mapped onto the PDA's screen as shown in Figure 2. As the user moves around the gateways' positions are updated.

The RelateGateway system provides two different modes of operation. In the scanning mode the system discovers all the devices (and their corresponding services) within visibility range (i.e. within line-of-sight, from a sensing perspective). The goal of this first mode is to replicate, in the user interface, what the user actually sees. Alternatively, in the conditional mode the gateways appear automatically when certain spatial conditions are met. The purpose of this latter mode is to indicate that the service exists and might be useful in the current context. Imagine a user sitting in front of the PC writing text on his PDA. A small keyboard gateway will appear on the screen, advertising use of the keyboard instead of the more limited keypad provided by the PDA itself. In the conditional mode, the spatial conditions which cause the gateways to appear differ depending on the service. For example, in our initial prototype the keyboard gateway is only shown if the PDA is within 50 cm of the keyboard since the user needs the keyboard and the PDA to be within reach.

Each gateway can be used in two different ways. First, it is a target area for drag-and-drop. For example a file can be printed by dragging it onto the printer gateway and a slideshow can be started by dragging a slide set onto the display gateway. Secondly, it can be used as a button. If the user clicks on it an options menu will be opened. This is an important functionality since drag-and-drop always triggers a default action without

allowing for any specific settings. By dragging a file on the printer gateway the whole file will be printed with the default options. However, if the user wants to print only some pages or change the properties, this is possible by invoking the service menu by 'clicking' on the gateway.

Implementation of the User Interface

As shown in Figure 3 (left) the gateway interface is designed to be a seamless extension of any existing desktop-style interface. The concept is to integrate the gateway in an unobtrusive manner at the perimeter of the screen but closely integrated such that direct interactions with the desktop or with other applications are facilitated. To achieve this, the user interface was implemented as a portable Java application using Swing GUI components. In order to be able to move the gateways around the screen they are implemented as small windows whose position can be controlled.

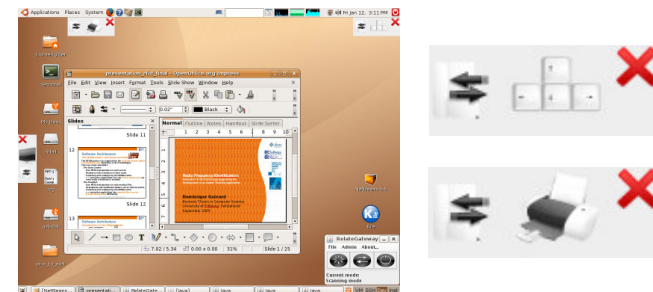


Figure 3. Left: User interface running on a Linux laptop
Right: Detailed view of two gateways.

Figure 3 (right) shows that each gateway is composed of two distinct zones. The first is used for drag and drop invocation of services whereas the second is used to for

click invocations. On the click zone an icon representing the device the gateway interfaces is provided. In addition to the gateways, a toolbar is integrated in the interface, to permit users to start or stop the RelateGateway application, and to toggle between the modes of operation (scanning/conditional).

The application can operate with any suitable accurate positioning system but is being developed in tandem with a specific sensing technology for direct sensing of the spatial arrangement between co-located devices [3]. However, we have integrated a Wizard of Oz interface in order to be able to run studies on UI aspects before the sensing technology is ready to use for this application. The wizard only simulates the user position and orientation, as a service architecture enabling remote interactions with the devices has been developed for the prototype.

Preliminary Study

The interface prototype has been tried with five users in a pre-test carried out for preparation of a larger study. The test involved basic tasks, such as printing of a document that required users to move around the test environment (cf. Fig. 1) to physically discover and interact with services.

The tests indicated that the principal idea of an interface that presents services in a dynamic position- and orientation-dependent manner is well received. The dynamic spatial arrangement of service gateways led users to comment on the interface as being, for instance, "really intelligent". The users perceived the mobile device as being an "enhanced universal remote control" for the devices surrounding them. They really

enjoyed this aspect and referred to this, for example, as "having real power over the devices" or "a SciFi movie-like application".

The tests also brought a range of specific usability issues. For example, the visual distinction between the drop and the click zones was consistently found to be confusing. The users also suggested a simpler gateway representation with a single icon (representing the device) for every type of interaction. Some users also discussed the size of the gateways. We designed them to be small and unobtrusive which was unobtrusive was fine with the users as long as they did not interact with the gateways. However it was suggested that the system should somehow predict their intent to use a particular gateway and automatically enlarge the gateway to ease the interaction.

References

- [1] Newman, M. W., Izadi, S., Edwards, W. K., Sedivy, J. Z., and Smith, T. F. 2002. User interfaces when and where they are needed: an infrastructure for recombinant computing. In *Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology* (Paris, France, October 27 - 30, 2002). UIST '02. ACM Press, New York, NY, 171-180.
- [2] Addlesee, M., Curwen, R., Hodges, S., Newman, J., Steggle, P., Ward, A., and Hopper, A. 2001. Implementing a Sentient Computing System. *Computer* 34, 8 (Aug. 2001), 50-56.
- [3] Kortuem, G., Kray, C., and Gellersen, H. 2005. Sensing and visualizing spatial relations of mobile devices. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology* (Seattle, WA, USA, October 23 - 26, 2005). UIST '05. ACM Press, New York, NY, 93-102.