Journal of Universal Computer Science, vol. 16, no. 15 (2010), 2019-2037
submitted: 31/1/10, accepted: 28/7/10, appeared: 1/7/10 © J.UCS

Supporting Mobile Users in Selecting Target Devices

Giuseppe Ghiani, Fabio Paternò
(ISTI-CNR, Pisa, Italy
{giuseppe.ghiani, fabio.paterno}@isti.cnr.it)

Abstract: The availability of applications able to exploit multi-device environments is steadily increasing. Rather than using all devices in the same way, users tend to assign different roles to devices due to the capabilities needed, such as computational power and screen size. Researchers and developers have started to introduce various techniques and tools to support managing applications across multiple devices. In this context, target device selection is often problematic, especially in unfamiliar environments. We present a novel technique for supporting device selection by providing dynamic graphical representations of user's orientation and position in relation to the available target devices in the current environment. We report on its design, implementation and discuss two possible location-aware representations of the user and the target devices.

Keywords: Context-aware interactive systems, multi-device environments, mobile devices, user location and orientation, device selection
Categories: H.5.m

1 Introduction

Our life is becoming a multi-device experience. The availability of interactive services as well as the possibility of accessing them through a wide variety of devices is steadily increasing.

Dearman and Pierce [Dearman, 08], in a recent study about why and how people use multiple devices, found out that users employ a variety of techniques for accessing information across devices, even if each device has specific features that determine its limitations and make it suitable to perform some tasks. Indeed, even if the computational resources of mobile devices (e.g.: PDAs) are growing, specific capabilities are recommended for performing some tasks. For example, watching a long video or looking at high resolution pictures are typical examples of functions requiring certain features, such as a suitably large screen. Thus, users do not use all devices in the same way and tend to assign different roles to devices both by choice and by necessity.

There has been an increasing number of features and services aiming to exploit the multi-device technological availability of modern contexts of use. Pick-and-drop [Rekimoto, 97] was the first interaction technique to provide support across multiple devices, in particular to interactively exchange data across various devices. Kozuch and Satyanarayanan [Kozuch, 02] put forward a solution for application migration based on the encapsulation of all volatile execution state of a virtual machine (which was limited only to migration of applications among desktop or laptop systems). Luyten and Coninx [Luyten, 05] presented a system for supporting distribution of the

user interface over a federation or group of devices. They mainly considered distribution of graphical user interfaces across desktop and traditional mobile systems. Support for the migration of Web applications across various types of devices was proposed [Bandelloni, 05] exploiting logical descriptions and automatic user interface generation. More generally, researchers and developers have started to introduce various techniques and tools to support managing information and activities across many devices.

In this context one important problem is how to support mobile users in selecting the target device in which to continue their interactive session when multiple devices are available. There is potentially no limit to the type and number of target devices that may be deployed. Thus, enabling the user to easily discover and select the best candidate is often problematic. This is particularly true when a user has never visited the environment before and does not know what devices are available, where they are located, and their availability for cross-device activities.

Another aspect that can be useful in this respect is to enhance smart environments with the possibility of automatically detecting and exploiting events related to user interaction with the current context, such as changes in orientation or proximity of the user with respect to a certain device. Managing such events can also be useful for the automatic triggering of specific functions (e.g., launching a particular service on a target device as soon as it is in front of the user).

In this paper we describe an environment to help the mobile user discover the devices available in the visited environment and facilitate selection of the target one in order to enable inter-device operations. We have designed and tested a graphical component that exploits positioning support, based on active RFID (Radio Frequency IDentification) technology, and an orientation capability enabled by a wearable electronic compass. Users are provided with graphical feedback about their location with respect to the available target devices by means of iconic representation. We describe the design of the solution, and discuss two representations of the interactive embedded component. We are also interested in better understanding how the combination of the two above-cited relatively well-known techniques (RFID and compass) may support the mobile user in realizing where the available target devices are located in an interactive map. We also discuss the benefits of the proposed solution, as well as its limitations and possible further improvements.

In the paper, after discussion of related work, we present the design and implementation of our Device Selection environment and an example application for migratory interfaces. Two possible representations of the available devices are discussed, and we report on an associated user test. Lastly, we provide some conclusions and indications for future work.

2 Related Work

In our investigation we study the possible use and representation of user location information with respect to the available devices in a smart environment to provide support for selecting a target device. Many applications could benefit from this technique. For example, [Ciavarella, 2004] presented one of the first indoor mobile guides for museums. This type of application could also exploit the use of other interaction platforms, e.g. large public screens, and thus would need a tool such as the

one presented in this paper. Such a tool would allow users to get graphical representations of the dynamically discovered devices depending on their position, and to select one to continue receiving additional information regarding the artworks encountered. The use of ontologies to support location-based services is explored in [Tsetsos, 06]; we focus on a different issue: how to discover and represent dynamic set of devices that can be selectable by the user.

An interesting approach to the user positioning support was proposed in InfoRadar [Blom, 04]: the authors enhanced a location-based messaging system with a radar-view of the environment locations with associated virtual content. The radar strategy for presentation consists of showing the user in the centre and the environment items around her according to the distance and orientation. Evaluation revealed that some users found the radar-like presentation quite disorientating due to the continuous relocation of the virtual items on the display. As in InfoRadar, we also exploit a localization technology and support for detecting user direction in real time. However, while InfoRadar provides only a radar-like view of the environment (i.e.: user-centred), we further investigate the positioning display strategies to improve their usability. The study illustrated in [Hermann, 04] also deals with the use of maps for guiding users through an environment. The authors present a mobile guide for fair exhibitions and conclude that the egocentric modality for presenting the map (that we call user-centred mode) provides the best support for navigation. A user test was conducted involving random fair visitors but, as the authors state, no electronic compass was used for automatically updating the map orientation. We are instead interested in evaluating a full automated system which provides a feedback in real time by means of an electronic compass as well.

Research work on how to represent off-screen objects has recently stimulated a good deal of interest. Halo [Baudisch, 03] conveys direction and distance, but is susceptible to clutter resulting from overlapping halos. Burigat, Chittaro and Gabrielli [Burigat, 06] compared Halo to scaled and stretched arrows that encoded distance as size and length of arrows, respectively. They reproduced Baudisch and Rosenholtz's finding that Halo improved performance when precise distance was required, and also found that scaled and stretched arrows were faster and more accurate than Halo in an off-screen target ordering task. Another technique is Wedge [Baudisch, 08], which represents each off-screen location using an acute isosceles triangle: the tip coincides with the off-screen locations, and the two corners are located onscreen. Wedge showed significant accuracy advantages over the Halo. However, in general such techniques address slightly different types of issues and applications. They tend to provide support for applications representing information such as geographical maps with associated points of interest (such as hotels, restaurants, …). Our system aims to provide support for representing smaller areas, such as rooms populated with various types of interactive devices (PCs, laptops, printers, …). Thus, in our case it is important to provide a complete view of what is available (deployed and active) and its features rather than to have indications that there is something of potential interest somewhere off the screen.

Seager and Fraser [Seager, 07] describe a study on the usability of automatic, manual and physical map rotation compared with static north-up map for pedestrian navigation supported by handheld device. The users involved in their tests preferred

the physical map rotation (i.e.: rotating the handheld to align the represented map with the real world).

An evaluation on north up map and heading-up map on automotive navigation was presented in [Daimon, 93]. The authors did not find any particular difference, but reported that the rotation angle for the heading-up display must be carefully chosen in order to not affect user's performance. In [Hermann, 03], where egocentric maps on mobile devices are considered, the authors indicate that the users preferred the track-up (or egocentric) view. The authors, after testing the automatically rotated, the physically rotated and the north-up map, conclude that the north-up is the less usable modality since it determines the highest number of navigation mistakes. In [Kray, 03], the use of three-dimensional maps together with textual and spoken instructions is investigated. The authors conclude that different contexts are characterized by different cognitive load: route instructions may be presented according to the context in order to reach a target point. However, while [Daimon, 93], [Hermann, 03] and [Kray, 03] focus on outdoor support for car or pedestrian navigation, our study refers to indoors and involves different issues that also arise from the different technology used for location and orientation detection. Indeed, our map representation aims to support a different task (selecting a target device), which requires users to create a mapping between the device representations in the map and the actual devices in the surrounding physical environment.

Previous work [Nacenta, 05] compared multi-display reaching techniques. Within their design framework our tool would be considered as coupling virtual and physical space. The nature of our destination space is discrete, and in terms of range we consider a room. However, that paper addresses a different topic since it considers a user in a fixed position moving objects between a laptop and an interactive table. Thus, our solution is novel with respect to those discussed in that paper because they consider a specific hardware configuration and users who are stationary.

In [Fischer, 08] the authors consider the RELATE system and the interaction with co-located devices that a user encounters in their surrounding environment. They present an experimental comparison of two graphical interfaces (spatial list and iconic map) and one non-graphical interface (alphabetic list) for the discovery and selection of nearby devices. The results highlight that users prefer the iconic map to the spatial and alphabetic list. The authors observe that, due to the limited number of devices used during the evaluation, the scalability of the solution cannot be assessed. They also state that, when the number of devices per room increases, additional visualization techniques are necessary to avoid compromising usability. In contrast to the RELATE paradigm, which aims to support spontaneous meetings among various people, we focus more on mobile users in multi-device environments who want to select one of such devices for some reason, and thus are interested in obtaining a representation of the environment and the associated devices rather than in identifying the relative position of the available devices. Indeed, we assume that the visited smart environment is already configured (i.e. an area map with the position of the stationary devices is available). Since we count on a ready to use map, we do not have to determine the relative position of the target devices, but only that of the user. Our approach simplifies the localization infrastructure and prevents us to equip each target device with specialized equipment (a sensor dongle or RELATE-like hardware). In

addition, RELATE hardware is limited to line-of-sight between devices and has been optimized for co-planar devices.

Indeed, instead of RELATE dongles we have adopted off-the-shelf technology, RFID (Radio Frequency Identification) active tags, for detecting when the user is in proximity of certain points and a wearable electronic compass for gathering her current direction. In our Device Selection support, dynamic information about the target devices is gathered through a discovery protocol, which is merged with the users' location, in order to provide them with representations showing the state of the devices as well (e.g., active/inactive, …). More detail on the device discovery protocol is provided in the next section.

3 Design and Main Features

The initial requirement for the Device Selection support was to implement a solution for representing the set of target devices available to the mobile user in the visited smart environment. The display should not be limited to a list of devices, but suggest in some informative way the location of the devices as well. A map view representation seemed a reasonable option for this purpose. We are aware that a map-based user interface typically introduces some bias, due to the simplification and potential small inaccuracies with which it reflects the real environment. However, a map can provide more information than a simple device list, a solution often used to indicate the available devices. Indeed, a map view can show the set of devices by providing their type (using different icons for different types) and, at the same time, by giving indications about the colocation of the device(s) and the user. Our map representations are consistent with indications proposed in previous studies on indoor navigation: [Puikkonen, 09] reported on a study carried out in a shopping mall, which stimulated some recommendations for user interface design supporting indoor navigation, such as heavy pruning of details and simplification of the graphical layout. In the simplified maps there are icons for each device that, when selected, trigger the display of various pieces of associated information (e.g. device type, operating system, screen resolution, name, address, picture of the actual device, …).
We soon realized that the map of a visited smart environment could be mainly presented in two ways: with the map fixed and the user icon moving around and rotating, or with the user fixed and the map rotated/translated according to user's movements and direction. In particular, the following display strategies have been adopted.

3.1 Map-centred approach

In this modality, often referred as north-up or exocentric, the map of the environment has fixed position and orientation. The user icon is translated and rotated according to her movements. For example, if the user changes location/direction, then the position and orientation of her icon on the map are updated accordingly. See Figure 1.

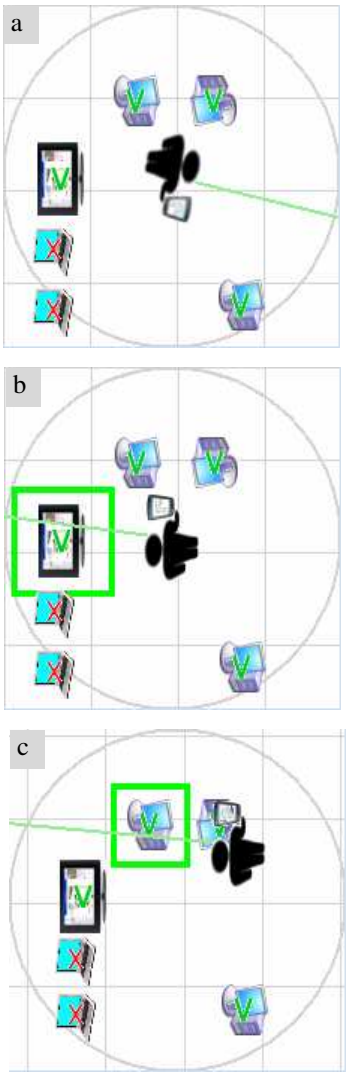

Figure 1: Examples of visualization in map-centred mode: the user in the center of the room (a); the user in the center of the room looking at a large screen device (b); the user after moving towards a desktop device and looking another desktop device (c)

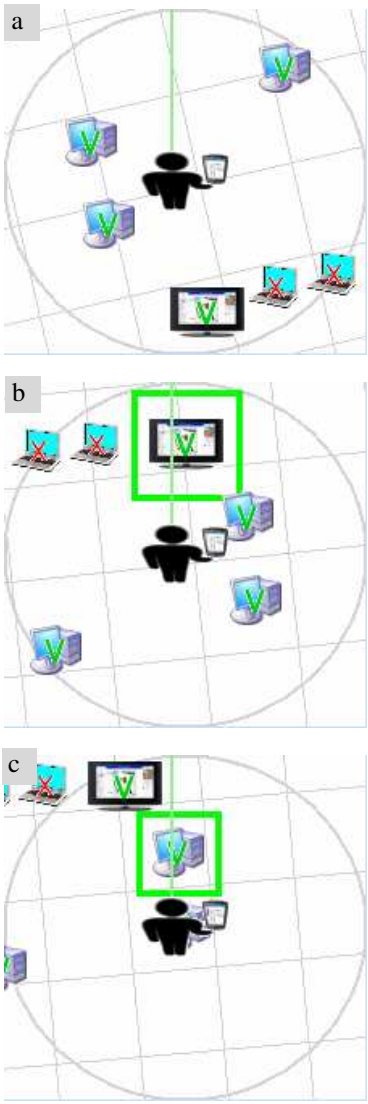

Figure 2: Examples of visualization in user-centred mode: same situation as Figure 1

3.2 User-centred approach

This strategy, often referred as heading-up or egocentric, consists of centring the user icon on the container and never changing its position or orientation, see Figure 2. As the user moves and/or changes direction, the map is translated and rotated around her. For example, if the user turns left 30°, then the map is rotated 30° right. When the

user is approaching a new tagged device, the device position becomes the new rotation centre of the map.
The first version of the tool was evaluated through a first formative usability test to investigate which visualization modality (map or user-centred) was preferred by the users. From the test, that involved 11 users, a slight preference towards the map-centred mode emerged.
Taking into account the observations and suggestions reported during the preliminary evaluation, we enhanced the Device Selection Map component with some functionalities before setting up a further user test. One of the new functions was the icon orientation, which shows the devices' icon oriented reflecting the orientation of the real device. Another function was the zoom in/out over the visited point. The grid of the map was also introduced for giving an indication about the current zoom factor. This feature can be switched off by the user.

4 Example Application: Support for Migratory User Interfaces

A possible application for the Device Selection environment is the support of user interface migration. This means that user interfaces can dynamically move from one device to another to follow the users and allow them to continue their activities. Thus, when users change device they can immediately continue from the point they left off and found all the information they entered in the previous device.

On the source user device side, migration is supported by a migration client (such as Device Selection Map), which is a tiny application running in background able to trigger the user interface migration from the current device to another (e.g. from the mobile to a stationary desktop system) and to select the target device. Examples of software architectures supporting migration are presented in [Paternò, 09] and [Grolaux, 04]. Many applications can benefit from this type of infrastructure, usually applications that require long sessions, such as interactive games. For example, a user playing with her PDA outside, after reaching her house might want to migrate the game to a specific desktop system. However, such solutions for supporting migration either are not able to dynamically discover and represent the available devices [Grolaux, 04] or provide limited solutions for this purpose [Paternò, 09].

We have integrated the Device Selection tool with a migration infrastructure in such a way that when the mobile user enters a tagged area (e.g., a certain room or public environment), selects a candidate target device and triggers the migration. The tool highlights the selected target device, and then the application can be resumed on the selected target device by the migration infrastructure. Figure 3 shows the user interface of a simple client with list-based presentation of the discovered device names (left) (as used in [Paternò, 09]), and an enhanced client exploiting the Device Selection Map (right). Note that both the list and the map refer to the same situation, but in different ways. The list-based presentation shows only the 4 devices detected as active. In addition, the textual list of the names of the available devices is often unusable because users do not know the name/device associations. The Device Selection Map provides a more intuitive and informative representation of the deployed devices, their types, active state (the icon is annotated with a green "V" or a red "X" indicating whether the device is active or not), and their location with respect to the user. Six icons are shown, but only 4 devices are active: the three desktops and

the large screen, which are checked with a green "V" (the two laptops are instead annotated with a red "X").

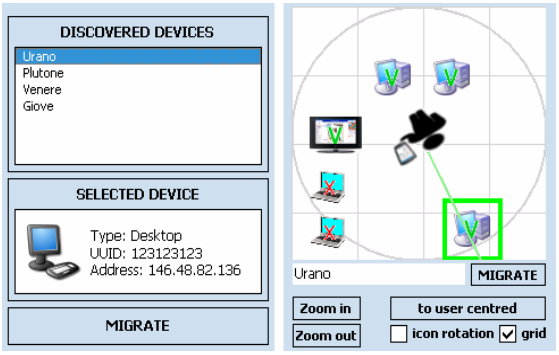

Figure 3: The user interfaces of two migration clients: the list-based (left) and the Device Selection Map-enhanced (right)

5 Device Selection Tool Architecture

The architecture of the support for device selection presented in this paper is summarized in Figure 4.

5.1 Enabling Hardware

We configured a localization solution based on RFID technology and electronic compass for detecting user direction and position with respect to the devices deployed on the environment.

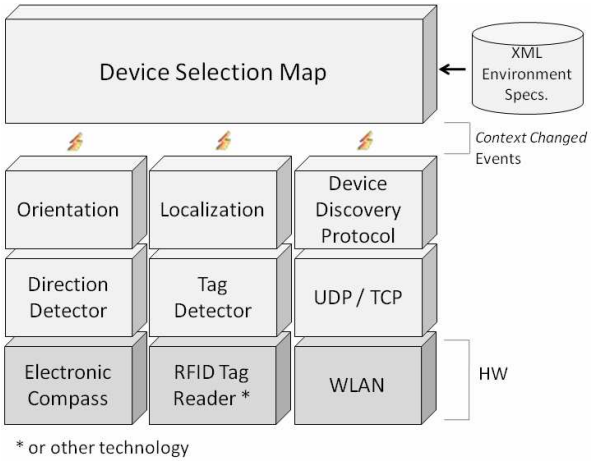

Figure 4: The architecture of the Device Selection environment

The enabling hardware is indicated by the lowest layer of the diagram in Figure 4. The electronic compass is a wearable device that detects user direction with respect to North. We have developed the compass peripheral device in our laboratory to satisfy the requirements of our application. Since wearability and compatibility with PDAs were needed, the electronic compass has been made battery-operated and Bluetooth-enabled. The current electronic compass is the result of many experimental trials that regarded both user comfort and performance. In its current version it is wearable on a belt and thus it detects the direction of the user's body.

Figure 5 shows the detector device that we have developed and used for our experiments (it embeds both a compass sensor and distance sensor).

Figure 5: The prototype of our module that detects both user orientation and obstacle distance

We believe that in the near future many PDAs equipped with support for direction sensing may be available on the market. Thus, we trust that the complexity of the hardware may reduce in the future, leading to a simpler and lighter solution.

The RFID tag reader is plugged on the PDA through a Compact Flash (CF) card. The tag reader has an antenna of 6 centimetres and in this way it is able to detect a RFID tag up to 5 meters.

5.2 Software Architecture and Graphical Interface

The first layer above the HW consists of the framework for easily accessing the enabling technologies (i.e. to integrate them into any application) and is independent of the application. Direction detector and Tag detector are software modules that interface to the electronic compass (to get user direction) and to the RFID reader (to compute the nearest visible tag), respectively. The device Discovery Protocol is a set of functions for dynamically detecting the active target devices using UDP (User Datagram Protocol). The protocol, executed by every device that runs the device discovery client, basically announces the presence of the local device by "hello" messages and collects the announcements of the remote devices. The "hello" messages contain a description of the device: name, type, internal and external IP address, screen size and operating system. Such information is stored in the description file of each device. Additional parameters could also be added. However, care must be taken while coping with dynamic information. For example, the expected time to perform a migration may not be indicated, because it is strictly

related to the structure of the application to be migrated (e.g.: a Web page), that is not known before the adaptation module has performed its computations.

In practice, every device joins the discovery multicast group, listens continuously to incoming UDP "hello" messages from other devices and periodically multicasts its own "hello". When a device is about to be deactivated, the discovery protocol multicasts a "bye" message. The list of "discovered" devices there are only those devices that have announced themselves recently (the time-to-live interval is settable in the configuration file), in order to avoid inconsistencies. This is because when a device is abruptly switched off without correctly multicasting a "bye" message, (or when the "bye" UDP message is lost), that device is kept in the discovery list of the other devices. One characteristic of the UDP multicasting is that it is implicitly limited to the local network (the gateways typically block multicasting to avoid flooding). This is well-suited to the concept of user interface migration, which involves devices within a well defined setting.

The third layer from the bottom belongs to the proposed tool, such a layer triggers events when relevant changes occur in the context. The events are then caught by the Device Selection Map, which updates the graphics. For example, a Direction Changed event is raised by the Orientation module only when the user changes direction by a value above the threshold, to lower graphical flickering. The Tag detector triggers a Location Changed event only when a tag signal is detected with a certain strength (i.e. when the user is in close proximity to it). The XML Environment Specifications define the set of areas and devices deployed in. An area may represent a room or a part of it and is basically defined by size, orientation (with respect to North) and set of items. The main attributes of an item are: type, position within the area, orientation with respect to the area and (if the object is tagged) id of the tag (e.g., RFID). Note that an area item could be a target device as well as an element of the environment that may help the user to orientate herself (e.g., a coffee machine).

A graphical editor for the smart environment allows the creation and update of the area maps: devices can be instantiated and deployed on the map in order to reflect their real position within the room (see Figure 6).

The editor is a desktop application and produces an XML specification of the environment layout. The XML descriptions can be manually copied onto the devices or downloaded automatically from the server by the device selection client when it is started.

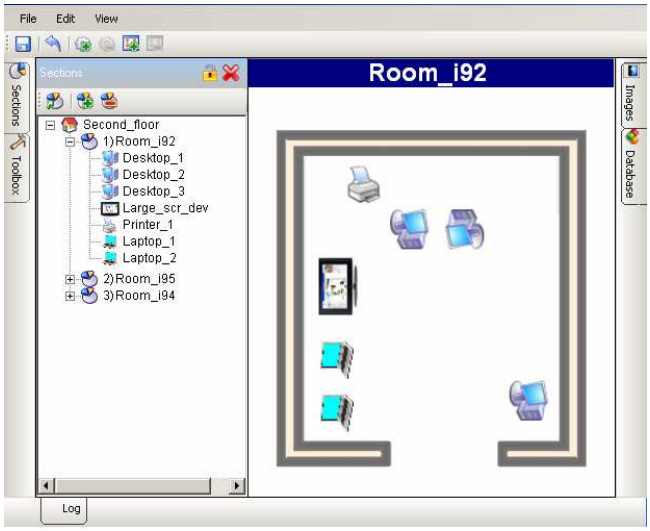

Figure 6: Overview of the environment editor main window

The set of devices deployed in the visited area is compared at run-time with the list of active devices in order to define the ones that are currently available for selection. Thus, the representation of all the deployed devices reflects the real current state, which users see represented on the PDA, facilitating their orientation. The icon style (annotated with a green "V" or a red "X") indicates whether each device is active or not.

Location/orientation detection and devices display modes

Locating the user and sensing her direction is fundamental to inform her about the devices deployed around. So far we have relied on pre-defined maps of the environment, that can be stored on the mobile device or downloaded on-the-fly from a gateway of the smart environment. The position is detected when the user approaches a RFID tag associated to a device, while direction is continuously sensed by the compass and updated in real time on the map.

Localization is event-based: when the Tag detector determines that the user has approached a new tagged point it raises a Location Changed event with the tag id of the new location. Even if only RFID tag detection is enabled so far, the Tag detector component can host additional specific threads for exploiting different localization technologies.

We use a compass to detect user orientation in our mobile support. Mapping the current user direction on the graphical representation (i.e.: updating the area map) is simple since the orientation of the visited area is defined at configuration time: the area orientation is fixed. This aspect is important for enabling the right orientation of the map in user-centred mode and the right orientation of the user icon in map-centred mode. The environment map should reflect the actual layout with respect to the current location and direction in order to facilitate user's orientation.

Device discovery

The location/direction aids facilitate associating the virtual items to the corresponding ones in the real world (and vice versa) through a graphic overview of the currently visited area. This is done by matching user current position/direction with the static data of the target devices. Dynamic data about the devices are sensed through a device discovery protocol, similar to the UPnP one, executed by the application that supports the device selection capability. The device discovery indicates which devices are actually active among those represented in the static map.

Management of context-related events

The events that the Device Selection Map is able to manage are: location changed, faced device(s) changed and discovered devices list changed.

The Device Selection Map, as already said, rather than as a standalone component, has been designed to ease device selection in a smart environment by being integrated into a wider application. This can be the case of a mobile client application for accessing functionalities deployed on stationary devices. The orientation capabilities of the Device Selection Map can be exploited even outside of the component to facilitate the identification of the right target device. As an example, we implemented a technique by which the user orientation with respect to a particular external device is detected by the component and notified to the user by showing some alert on the display of that device as well. A big font sized message such as "you are pointing at me", shown on the external device, can be useful for the user to be sure about the correspondence between the virtual item and the real device (e.g., a desktop PC) and to avoid ambiguity. This is especially true when one or more devices in the room are very close to each other. In particular, we have implemented a client that hosts the Device Selection Map and exploits web services deployed on the target stationary devices to provide feedback upon verification of relevant events, such as graphical alerts on the display of the watched device. For example, when the user moves towards a device, the mobile client application detects the location changed event or the new device(s) watched event of the Device Selection Map and calls the web service of the selected device, which shows a text such as "you have selected me" or "you are watching me" in a popup window.

6 Evaluation

A user test has been performed in order to evaluate the Device Selection Map component enhanced with icon rotation and zoom capabilities.

6.1 Participants

The 14 participants were all recruited among the personnel of our institute and were between 24 and 42 year old. (M=31, SD=5.6). Six of them had previously used orientation supports (such as car navigation systems). They also rated on a scale from 1 to 5 (the lowest and highest value, respectively) their experience on the use of PCs (M=4.6, SD=0.5) and PDAs (M=3.2, SD=1.3).

All the participants, at the time of the test, were working in a informatics Institute. We believe that such a sample represents well the typical users of our

support, who would be people with a good knowledge of technology and the various interactive devices available.

6.2 The Test Tasks and Environment

The test took between 10 and 15 minutes per participant and was performed in two rooms of our laboratories. Each user was introduced to the capabilities of the system and instructed about the tasks to perform. In the first room the user could freely move around, taking some familiarity with the position/direction feedback and trying the various options (zoom, icon rotation, switch to user-centred mode). At this stage, the user was also encouraged to ask any question regarding how the system works.

In the second room, which was about 4 meters wide and 8 meters deep, 7 devices were deployed: two laptops, three desktops, one digital TV with large screen and one printer. Just four devices were RFID tagged, the system was tuned to detect the tag from up to one meter and the participants were informed about that. Figure 7 shows an overview of the room. The relevant devices are annotated as: 1-2 (laptops), 3 (digital TV with large screen), 4 (printer), 5-6-7 (desktops).

Figure 7:.Overview of the room where the user test took place. The relevant devices are annotated with 1 to 7 numbers.

Each participant had to carry out specific device selections by pointing exactly the requested device with the body and clicking the button “migrate”. The device to select was indicated through a descriptive sentence (e.g., “the desktop near the window” or “the laptop next to the door”) by the test supervisor. For feedback purposes, the description of the device currently “watched” by the user was automatically displayed by the application on a label in the GUI. Users had to select six devices: three in one type of representation and three in the other. For avoiding biases due to the learning effect, half users started with the map-centred mode and the others started with the user-centred mode.

During the test session, each user was free to use any additional capability of the Device Selection Map, such as zooming, icons rotation, grid.

6.3 The Test Results

The time elapsed between the supervisor's request and the right selection was logged: in map-centred mode, it varied between 2 and 44 seconds (M=11.9, SD=9.9); in user-centred mode, it varied between 2 and 150 seconds (M=38.8, SD=32.3). As a consequence of the high diversity of the time needed by the users for selecting the target devices in user-centred mode, the Standard Deviation is also high. A Wilcoxon test (N=14, Z=-3.297, p=0.001) on the time logging revealed "high significance".

After the test sessions each user answered to a questionnaire providing some informal observations/suggestions and an overall rating on the solution. The two presentation versions were rated on a scale from 1 to 5 (the most negative and positive value, respectively). On average, the user-centred approach was rated 3.3 (SD=0.9) and the map-centred 3.8 (SD=1.2). Eleven users preferred the map-centred and 3 the user-centred. The results are also summarized in Table 1.

	Map-centred	User-centred
Mean time for selection / Standard Deviation (seconds)	11.9 / 9.9	38.8 / 32.3
Preference by users	11	3
Average rating / Standard deviation	3.8 / 1.2	3.3 / 0.9
Difficulties encountered	None 10	None 4
	Some 4	Some 9
	Many 0	Many 1

Table 1: Summary of the test result.

Most users reported that the map-centred approach was more intuitive and that was easier to find the target devices due to the static representation of the environment that provides an overview faster to analyse at different times. Someone also found that it is more realistic because it shows the user icon approaching the devices, differently from the user-centred in which the map moves while the user does not. Among those who preferred the map-centred, a user said that one of its benefits is the better fluidity of the refresh when the direction changes.

The participants who preferred the user-centred strategy found intuitive to see the devices repositioned on the component by reflecting the current vision of the environment (i.e.: the desktop in front of the user is drawn on the upper area of the component) making it possible to observe just one portion of the component to see, for example, what is in front of him/her.

Even if all the participants received the same instructions before starting the trial, we found that some of them, after making little moves (e.g., a half meter), looked at the PDA waiting for the upload of their position on the Device Selection Map. They simply had forgotten that the position update occurs when particular points (i.e. a RFID tagged point) is reached. Some users tried to point the devices by rotating the PDA towards them. This misunderstanding was cause of troubles because the system detected the direction of user's body and not that of the PDA (since the compass is attached to the user's belt).

In general, the comments of the users were positive. Indeed, the question asking whether the approach was useful to support user orientation and ease device selection received an average evaluation of 3.8 (SD=0.8). Another question was about whether the user encountered particular problems in using the support (possible answers: none/some/many): for the map-centred 4 declared that they found some difficulties and the other 10 none, while for the user-centred; one found many difficulties, nine some, and four none. Also this result shows that the map-centred was found easier to use.

Regarding the use of the icons rotation mode, 8 users out of 14 judged it positively. The others told that rotating the icons in two dimension is not realistic, and thus it does not provide any useful indication to the non-experts.

6.4 Test Results Discussion

The preference towards the map-centred modality, that clearly emerges from the reported results, is related to the requirements, the task supported, and the capabilities and enabling technologies of our Device Selection tool. In general, the indoor localization is still an open problem, especially when coping with very short-range navigation. Thus, we believe that a possible way to improve the usability of our solution, alternatively to upgrade the hardware, is trying to optimize its performance. Indeed, more than one participant agreed on the need to improve the precision of the localization, but they also agreed on the usefulness of some feedback on the change of position. An example could be an alerting sound followed by a short animation showing the user icon moving towards her current position. It is interesting to note that several users suggested adopting some of the paradigms used by well known navigation applications (e.g. Google Maps), such as the indication of the current zoom status by means of a numeric value or a slidebar. Other indications suggested mapping the main elements of the rooms, such as doors and windows, besides the target devices.

As stated in [Seager, 07], we assume that the difficulty of recognizing a map that rotates without looking it could be an important cause of issues when coping with wide maps (i.e.: with representations more complex than simple turn-based instructions). Decreasing the flickering could be useful to increase the usability of the user-centred modality, even if it was not the main concern. This can be obtained by using more vectorial graphics instead of oriented bitmap/icons since a lower number of pixels would be repainted for each direction/position change.

A complementary feature could be the animated re-orientation of the map on request: the map would rotate only when the user asks for it. Since the application is able to detect orientation changes in real time, it may be also able to compute the animation in background, in order to play it when needed. An interesting advantage of this strategy is the efficiency, especially for devices with limited capabilities, because the elaboration can be performed between the change of direction and the request of map re-orientation.

The outcomes of our study might not be put in strict relation with those regarding the modalities of automotive or pedestrian navigation (i.e.: heading-up/north-up) because of the differences both in the relative precision of the localization systems and in the features of the tasks, as well as in the context of use. Before performing the user test session, we actually expected a preference towards the user-centred mode,

since it is the same used by in-vehicle and pedestrian navigation systems. During our test it was clear that most users preferred the map-centred even if most of them were familiar with generic navigation systems. A reasonable motivation can lie in the different structure between the road navigation and our indoor environments. Indeed, outdoor environments tend to be more structured than indoor ones because they contain elements that are easy to be mentally matched to the map objects. Thus, intuitively, a road along which the user is walking/driving, especially if it contains an intersection, is easier to match with its map representation than a desktop device of a room with its icon in the Device Selection Map (and this is particularly true when the environment is populated by many devices).

Our maps are instead defined by a set of items in the environment and if they often change position in the map it can be difficult then to associate them with their correspondences in the real environment.

The authors of [Hermann, 03] discuss about the importance of environment structure in aligning the map with the real environment (especially in manual alignment). Users' performance and the preference towards the user-centred modality in the Device Selection tool could increase with more structured maps.

7 Conclusions and Future Work

In this study we have described the design and implementation of an interactive tool for mobile devices aimed at facilitating target device selection in smart environments, which can be useful for example in supporting migratory user interfaces. We have also reported on its evaluation. The study conducted has highlighted a significant preference of the tested users for one specific representation modality: the map-centred display. We have explained that this preference is motivated by the type of task they have to accomplish (mapping device icons in the map with real devices in the room) and the indoor technology.

Considering the appreciation the users expressed for the display indicating the orientation of real devices on their icons, we are planning further work on this aspect. In particular, we plan to manage more context specific events related to the orientation of the devices as well (e.g., to trigger a "device pointed" event only when the user is facing towards the front side of the device, and not whatever side as happens now). We are also investigating the integration of information related to the state of the selectable target devices, such as time-related features (e.g., the expected time before a target will be available).

References

[Bandelloni, 05] Bandelloni, R., Mori, G., Paternò, F.: Dynamic Generation of Migratory Interfaces, Proceedings Mobile HCI 2005, ACM Press, pp.83-90, Salzburg, September 2005.

[Baudisch, 08] Baudisch, P., Irani, P., Gustafson, S., Gutwin, C.: Wedge: Clutter-Free Visualization of Off-Screen Locations. Proc. CHI 2008, 787-796.

[Baudisch, 03] Baudisch, P. and Rosenholtz, R.: Halo: A technique for visualizing off-screen locations. Proc. CHI 2003, 481–488.

[Blom, 04] Blom, J., Mäntylä, M., Oulasvirta A., Rantanen, M., Tiitta, S.: InfoRadar: group and public messaging in the mobile context. Proc. NordiCHI 2004, ACM Press (2004), 131-140.

[Burigat, 06] Burigat S., Chittaro L., Gabrielli S.: Visualizing locations of off-screen objects on mobile devices: A comparative evaluation of three approaches, Proc. MobileHCI 2006, 239-246.

[Ciavarella, 2004] Ciavarella, C. and Paternò, F. 2004. The design of a handheld, location-aware guide for indoor environments. Personal Ubiquitous Comput. 8, 2 (May. 2004), 82-91. DOI= http://dx.doi.org/10.1007/s00779-004-0265-z

[Daimon, 93] Daimon, T., Kawashima, H. and Mashimo, S.: Driver's Characteristics for Map Information Representation (North up Map/Heading up map) in Navigation Displays. Proc. VNIS '93, IEEE (1993) pp. 410-413.

[Dearman, 08] Dearman, D., Pierce, J.: "It's on my Other Computer", Computing with Multiple Devices, Proceedings CHI'08, Florence, ACM Press, pp.767-776.

[Fischer, 08] Fischer, C., Gellersen, H., Gostner, R., Guinard, D., Kortuem, G., Kray, C., Rukzio, E., Strengin, S.: Supporting Device Discovery and Spontaneous Interaction with Spatial References, special issue on Mobile Spatial Interaction in Personal and Ubiquitous Computing, 2008. (DOI 10.1007/s00779-008-0206-3).

[Grolaux, 04] Grolaux D., Van Roy P., Vanderdonckt J. Migratable User Interfaces: Beyond Migratory Interfaces, In the First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous'04), August 2004, pp. 422-430.

[Hermann, 03] Hermann, F., Bieber, G., Duesterhoeft, A.: Egocentric maps on mobile devices. In Proc. 4th International Workshop on Mobile Computing, IMC 2003, pages 32- 37. IRB Verlag, 2003.

[Hermann, 04] Hermann, F., Schmidt-Belz, B.: User Validation of a Nomadic Exhibition Guide, Proceedings Mobile HCI 2004, Springer LNCS, Vol. 3160, pp.86-97.

[Kray, 03] Kray, C., Laakso, K., Elting, C., Coors, V.: Presenting route instructions on mobile devices. Proceedings of IUI 2003, pp. 117–124. ACM Press, 2003.

[Kozuch, 02] Kozuch, M., Satyanarayanan, M.: Internet Suspend/Resume, Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'02) IEEE Press, 2002.

[Luyten, 05] Luyten, K., Coninx, K.: Distributed User Interface Elements to support Smart Interaction Spaces. Proceedings of IEEE Symposium on Multimedia, Irvine, California, 12-14 December 2005, pp. 277-286.

[Nacenta, 05] Nacenta, M., Aliakseyeu, D., Subramanian, S., Gutwin, C. A, Comparison of Techniques for Multi-Display Reaching. Proceedings CHI 2005, ACM Press, pp. 371-380.

[Paternò, 09] Paternò, F., Santoro, C., Scorcia, A.: Ambient Intelligence for Supporting Task Continuity across Multiple Devices and Implementation Languages, the Computer Journal, the British Computer Society, 2009.

[Puikkonen, 09] Puikkonen, A., Sarjanoja, A., Haveri, M., Huhtala, J., and Häkkilä, J. 2009. Towards designing better maps for indoor navigation: experiences from a case study. In Proceedings of the 8th international Conference on Mobile and Ubiquitous Multimedia (Cambridge, United Kingdom, November 22 - 25, 2009). MUM '09. ACM, New York, NY, 1-4. DOI= http://doi.acm.org/10.1145/1658550.1658566

[Rekimoto, 97] Rekimoto J.: Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environments, Proceedings of UIST'97, pp. 31-39, 1997.

[Seager, 07] Seager, W. and Fraser, D.S.: Comparing Physical, Automatic and Manual Map Rotation for Pedestrian Navigation, Proceedings CHI'07, San José, ACM Press (2007), 767-776.

[Tsetsos, 06] Tsetsos, V., Anagnostopoulos, C., Kikiras, P., and Hadjiefthymiades, S. 2006. Semantically enriched navigation for indoor environments. Int. J. Web Grid Serv. 2, 4 (Dec. 2006), 453-478. DOI= http://dx.doi.org/10.1504/IJWGS.2006.011714