Migratory MultiModal Interfaces in MultiDevice Environments

Silvia Berti, Fabio Paternò
ISTI-CNR
Via G.Moruzzi 1
56126 Pisa, Italy
+39 050 3153066
{silvia.berti, fabio.paterno}@isti.cnr.it

ABSTRACT

This paper describes an environment able to support migratory multimodal interfaces in multidevice environments. We introduce the software architecture and the device-independent languages used by our tool, which provides services enabling users to freely move about, change device and continue the current task from the point where they left off in the previous device. Our environment currently supports interaction with applications through graphical and vocal modalities, either separately or together. Such applications are implemented in Web-based languages. We discuss how the features of the device at hand, desktop or mobile, are considered when generating the multimodal user interface.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: Prototyping, Evaluation / Methodology; D.2.2 [**Design Tools and Techniques**]: User Interfaces

General Terms

Design, Human Factors, Languages.

Keywords

Migratory Interfaces, Multi-Device Environments, Model-based Design, MultiModal User Interfaces, Ubiquitous Systems.

1. INTRODUCTION

In recent years there has been an increasing interest in ubiquitous environments, where users can interact through a variety of devices while freely moving through different contexts. Multimodal user interfaces represent an important aspect in order to allow users to fully exploit such possibilities. However, the design of such multimodal user interfaces should take into account the features of the interaction resources of the devices currently available.

Migratory interfaces are interfaces that allow users to freely change device and still be able to continue the task from the point they left off in the previous one. In order to obtain a satisfying interaction three aspects are important: the user interface should adapt to the features of the new device, it should apply design criteria for obtaining usable results, and continuity at the task performance level should be guaranteed. The combined use of migration and multimodal interfaces is fundamental to obtain environments able to support natural interaction, where users can freely move about and still continue their activities in an effective and efficient manner. These features can be important in many types of applications: financial, auctions on line, games, and whenever users have to change place, but they still would like to carry on their tasks without having to start from scratch. While in the area of distributed systems there has been work on session persistence, such work has not addressed issues related to how the user interface can adapt to device's change, in particular when multimodal environments are considered.

In the following, we present a solution supporting migration through various types of platforms supporting various modalities. We use the concept of platform to group devices that share the same interaction resources (for example, the graphical desktop, the multimodal mobile device, the vocal device). The modalities supported are a key element in characterizing the interaction supported by one device. Our prototype supports interaction with applications through graphical and vocal modalities, either separately or together and it can be extended to support other modalities. Our current prototype supports generation for multimodal interfaces implemented in Web-oriented languages. We decided to start by considering this type of environment because the Web is the most common user interface, and recently new standards have emerged in order to extend user interaction through the Web to different modalities and not limit it to the graphical one alone. However, since the approach is based on the use of intermediate conceptual, device-independent languages, it can be easily extended to other implementation environments.

In the paper, we first discuss related work and then describe the logical descriptions used to manage the complexity inherent in the interaction resources variability. The next section is devoted to how multimodal user interfaces are generated depending on the available platforms. Then, we describe the migration support provided in our environment and an example application. Lastly, we provide some concluding remarks along with indications for future work.

2. RELATED WORK

Developing multimodal user interfaces is still difficult and tools that support the designer to develop efficient and usable multimodal interfaces are still lacking. One possible solution is the Multimodal Toolkit for WebSphere Studio, which allows designers to develop X+V (XHTML+ VoiceXML: an emerging W3C standard for multimodal user interfaces) applications. This tool supports useful functionality that provides for reuse of some dialog components. Unfortunately, the designer still has to deal with a plethora of low-level details of the X+V language. In order to address such issues, some approaches [4] have been proposed to provide authoring environments for component-based development of multimodal user interfaces, which mainly provide graphical direct manipulation environments representing the component-based structure. In our work we follow a different approach, a model-based approach based on logical descriptions, supported by an environment able to transform them into implementations of multimodal user interfaces taking into account the features of the device at hand. Obrenovic et al. [12] have investigated the use of conceptual models expressed in UML in order to then derive graphical, form-based interfaces for desktop or mobile devices or vocal devices. UML is a software engineering standard mainly developed for designing the internal software of application functionalities. Thus, it seems unsuitable to capture the specific characteristics of user interfaces and their software. For this purpose, other models (such as task and user interface models) seem more suitable. While some other proposals (such as UIML [1], UsiXML [9] and XIML [15]) consider such HCI oriented models, they have mainly focused on development of user interfaces for graphical devices, with screens of different sizes without providing tool support for multimodal user interfaces.

A general reference framework for understanding the issues in distributed and migratory interfaces is presented in [5]. In this paper we present a tool supporting migration through multimodal interfaces in multidevice environments. One approach to the generation of user interfaces starting with logical descriptions is the Personal Universal Controller [11] but this is limited to generating simple interfaces for domestic appliances and it does not support user interface migration. Aura [8] also provides migratory support across different devices, but that is achieved by selecting among off-the-shelf applications the one that is most adequate to satisfy the user task with the available resources. In contrast, our approach extends applications in such a way to adapt their user interfaces to the availability of interface resources. In [14] the authors analyse the most typical structure of Web pages in order to identify the basic tasks that are supported and propose a taxonomy of tasks for presenting Web-based information in an effective and usable manner. They concentrate on information seeking characterised by four basic tasks: Situate, Navigate, Query and Details on demand. However, their analysis is limited to vocal interaction. Another approach [7] aims to adapt the interface content to different devices using annotated documents that define that define the appropriate modality for each object in each device, but there is no automatic support to produce the annotated documents.

A first version of our migration support was presented in a previous work [3], but that was able to support only transmodal migration (from graphical to vocal modality). Here we present a deeply revised solution able to support various types of multi-modal user interfaces (combining graphical and vocal modality) in different types of devices (both desktop and mobile).

3. USER INTERFACE LOGICAL DESCRIPTION

For several years one research area in HCI has focused on model-based design [16] [13]. The basic idea is to have some conceptual descriptions that should be easy to develop and modify, and then an environment able to render the corresponding interface. However, this approach has had limited success until an increasing variety of interactive devices became available. Given such a variety, allowing designers to work with logical descriptions has also had the result of limiting the effort required for developing the multiple versions. The corresponding environments should be able to support design criteria that adapt according to the features of the devices considered. Different abstraction levels can be considered in this process. In particular, various studies [16][2] have agreed on three main conceptual levels: the task level, where the activities that the users intend to perform are considered; the abstract level, which is a modality-independent description of the user interface; and lastly the concrete level, which provides a logical description that is more refined and thus modality-dependent. Therefore, for example at the task level we can say that the user wants to select a painting, at the abstract level we indicate that a selection interaction object is required, but we do not specify anything regarding the modality, so selection could be performed by gestural pointing or graphical selection or vocal input. At the concrete level we have to identify a modality, say the vocal one, and then we indicate which interaction techniques is used for this purpose (for example selection from an enumeration).

Such logical descriptions are represented through languages which are device-independent. Thus, they allow developers to designdevelop and implement their multidevice solution focusing on the main aspects without having to learn a plethora of low-level details in many implementation languages.

We have developed an environment [10] able to support various transformations through such abstraction levels, with each of them defined through XML-based representations. The activities that users intend to perform are represented in ConcurTaskTrees [13]: it is a hierarchical description where high level tasks are decomposed into more refined ones. A set of temporal operators among such activities can be specified (sequential, concurrency, disabling and so on) in order to obtain descriptions of flexible and realistic activities, during which it can happen to be interrupted or to perform concurrently different activities. For each task a number of attributes can be specified, indicating their frequency, performance allocation, objects manipulated to accomplish them.

is structured into a number of presentations. A presentation identifies a set of interaction techniques that are enabled at a given time. The presentation is structured into interactors (logical descriptions of interaction techniques) and composition operators that indicate how to put together such interactors. It is possible to distinguish between interactors supporting user interaction (interaction_interactor) and those that present results of application processing (only_output_interactor). While at the abstract level such interactors and their compositions are identified in terms of their semantics in a modality independent manner, at the concrete level their description and the values of the attributes depend on the available modality.

In the abstract description we indicate the logical interactions (such as selection, editing, control) and how to put them together through composition operators within each presentation. In the definition of the composition operators we aimed to capture the main communication goals of designers when they structure the user interface. Indeed, such operators indicate various composition policies: grouping (set of elements logically related), hierarchy (different levels of importance), relation (one element that has a relation with some other elements), ordering.

At a concrete level such logical descriptions are refined in a modality-dependent, but still implementation language-independent manner. So, for example, the selection in case of a graphical modality can be performed through a list or a radio-button whereas in the vocal modality it can be obtained either through a short enumeration or through a list of messages. Likewise, the concrete descriptions of the composition operators vary according to the available modality and interaction resources. For example, grouping can be implemented in the graphical channel through one or multiple attributes (fieldset, colour, position...) whereas in the vocal channel it is possible to group elements by inserting a sound or a pause at the beginning and the end of the grouped elements.

4. MULTIMODAL DESIGN PLATFORM-DEPENDENT

In the design and development process it is important to consider that there are tasks that may be meaningful only when using some specific platform or modality. For example, watching a long movie makes sense only in a multimedia desktop system whereas accessing information from a car in order to get directions to avoid a traffic jam can be done through a mobile device and if this task should be performed while driving when eyes and hands are busy then it can be better supported through a vocal interface. The available modality may also have an impact on how to accomplish a task, for example a vocal interface or a graphical mobile phone interface can require sequentially performing tasks that can be performed concurrently in a desktop graphical interface.

In the case of both vocal and graphical support the multimodality can be exploited in different manners [5]. The modalities can be used alternatively to perform the same interaction. They can be used synergistically within one basic interaction (for example, providing input vocally and showing the result of the interaction graphically) or within a complex interaction (for example, filling in a form partially vocally and partially graphically). Some level of redundancy can be supported as well, for example when feedback of a vocal interaction is provided both graphically and vocally.

We have decomposed each interaction interactors in three different states:

- Prompt: represents the interface output indicating that it is ready to receive an input.
- Input: represents how the user can actually provide the input.
- Feedback: represents the feedback of the system after the user input.

We have designed an environment, which is able to take into account the differences among the available platforms. Indeed, it is different to design a multimodal interface for a desktop and one for a PDA for several reasons, for example because the graphical resources available are different.

The tool can provide solutions according to predefined design criteria that can be still modified by developers. In the case of composition operators, they are implemented graphically in desktop interfaces because the graphical modality is dominant whereas in mobile devices they are supported both graphically and vocally. For example, grouping on a multimodal desktop interface can be implemented using the graphical channel through one or multiple attributes (fieldset, colour, position...), whereas on a multimodal PDA interface it can be implemented utilizing both graphical attributes and the vocal channel, for example inserting a sound or a pause at the beginning and the end of the grouped elements.

As regards the interactors, the reasoning is similar. In case of multimodal interfaces for a desktop platform we have decided to employ for only-output interactors mainly the graphical representation because the interface has a lot of space available while in the case of PDA we have decided to use either modality depending on the case. For example, if the description text is too long then it is better to use the vocal channel in order to avoid many scrolling and, if it is possible, to add an image to support the description.

In the case of the interaction interactors we have to consider the three possible states: for multimodal desktop interface the prompt and the feedback are provided only in graphical modality (assignment) and the input is provided either in graphical or vocal modality (equivalence), whereas in multimodal PDA interfaces the prompt and the feedback are provided in both modalities (redundancy) and the input is provided by either modality (equivalence).

In the multimodal PDA interface the designer can define some vocal commands (such us “next”, “back”, “last”…) to change the focus of the concrete objects in the user interface in order to make simpler the interaction with the PDA, because it can be tedious scrolling by means the stylus.

The environment supports the definition of some default parameters (general settings and default choices for the implementation of the composition operators). The settings can be applied to either the entire multimodal application or to specific presentations. For example, in both multimodal platforms (desktop and mobile) the welcome vocal message can be defined in order to allow users to understand that the interface is multimodal and that they can interact using both vocal and graphical modality.

In addition, in multimodal interfaces for both desktop and PDA if there are two or more interaction interactors in a presentation it is possible to define one global vocal command in order to perform all the interactions at once. This feature is very useful to perform the interaction faster especially when interacting with the PDA where writing by stylus can be complicated and take a long time. For example, if users have to book a room in hotel they have to insert the number of room, the type of room (double or single), the date of arrival and departure. In a graphical interface there is one interactor for each basic input, with the support of this feature the user can provide all data through one vocal command, such us “I would like two single rooms from 21 June to 24 June”.

Lastly, in both multimodal platforms in order to speed-up the specification of the interactors, it is possible to indicate how to implement the interactors (possible choices are: using a specific modality, using both modality in a redundant manner, or in a complementary manner) at the level of the composition operator. For example, the designer can decide to implement all aspects (prompt, input, and feedback) of the interactos involved in a grouping redundantly in both graphical and vocal modalities.

5. MIGRATORY INTERFACES

In order to support migratory interfaces we have developed an infrastructure, which supports the mapping and transformation rules in descriptions external to the application. Thus, we avoid the limitations of other approaches in which they are hard coded in the infrastructure in an ad-hoc manner. Our migratory infrastructure exploits the logical descriptions of such interfaces [3]. Such logical descriptions can be available either because they have been used to generate the user interface following a model-based approach or because a reverse engineering tool has been applied. Such tools are able to derive at runtime a logical description starting with the implementation code. In our group we have implemented a tool able to take the code of Web pages and automatically build the corresponding logical descriptions at the three considered levels (concrete, abstract, task) in an incremental manner.

In our migration environment, first of all the devices that want to be involved in any migration have to register to the migration service. In this way, the migration server has a list of available devices and additional information regarding their features (platform, availability, whether it is suitable for shared use, …). The migration requests can be issued by either the user or the system (for example, when it detects that the battery is low).

In order to allow users to send migration requests a client application with its user interface should be loaded and running. When migration is requested the environment is able to collect the state of the source user interface resulting from the previous user interactions (such as text entered, elements selected and so on).

Then, a version of the user interface adapted to the features of the target device is activated. Such version is transformed in such a way to keep the state resulting from the previous interactions and adapt it to the features of the target interface as well. The identification of which part of the target interface to activate is based on the analysis of the corresponding logical descriptions. The tool selects the part that supports the tasks most similar to those supported by the source user interface when migration occurred.

In the migration process the modality supported by the target device can affect the process. So, for example, in case of migration to a vocal device then in the target interface activated first the input entered in the previous device is summarized so that users can more easily recall it and then the interface starts to ask for input without requiring again that already provided in order to avoid boring the user.

When dealing with different modalities we have to consider several issues. Graphical interfaces do not translate immediately

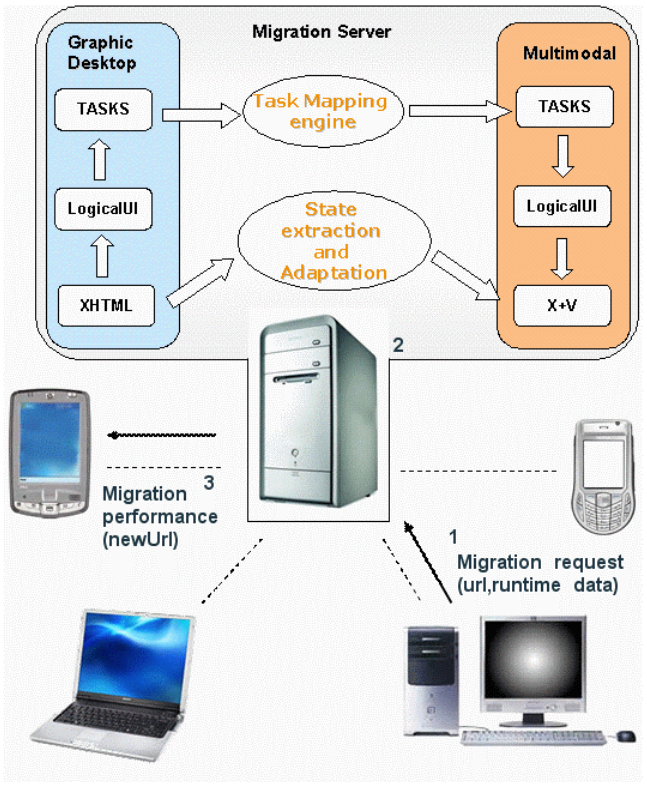


Figure 1: The migration process supported by our environment.

into speech interfaces. In fact, graphical interfaces do not always reflect the vocabulary that people use when talking to one another, and having the system read exactly what is displayed on the screen is rarely effective. Likewise, users find it hard to listen to exactly what is printed on the display. For example, in an e-mail application an inbox usually consists of a sort of table with a list of headers containing information, such as the sender, subject, date, time and size. In graphical interfaces, users can quickly scan the list and find a subject of interest or identify a message from a particular person. In vocal interfaces reading all the information aloud exactly as displayed takes a long time, and some information such as size, time and date, which users can easily ignore in graphical presentation, is given too much prominence. Thus, an effective speech interface would require an alternative organization structure, which groups vocal information into categories, such us subject and sender.

Figure 1 shows the migration process supported by our environment: when a request of migration is generated (in a desktop platform in the figure) then it is sent to the migration server, which has to identify the target device, extract the state resulting from the previous interaction and adapt such state to the user interface for the target device (a multimodal mobile platform in the figure). In addition, using the logical descriptions it identifies the part of the interface associated with the target device supporting the set of tasks most similar to those supported by the source interface when migration was requested. This part of the user interface is then activated with the associated state, already transformed for the interaction objects actually supported by the target device. In the case of multimodal migration, platforms support different modalities and adopt different concrete objects, so the migration server cannot perform a simple one-to-one mapping between tasks supported by the graphic platform and those supported by the multimodal one. In fact, some tasks are not supported or must be implemented with different concrete objects by the mobile device, and the runtime data have to be adapted to such new concrete objects.

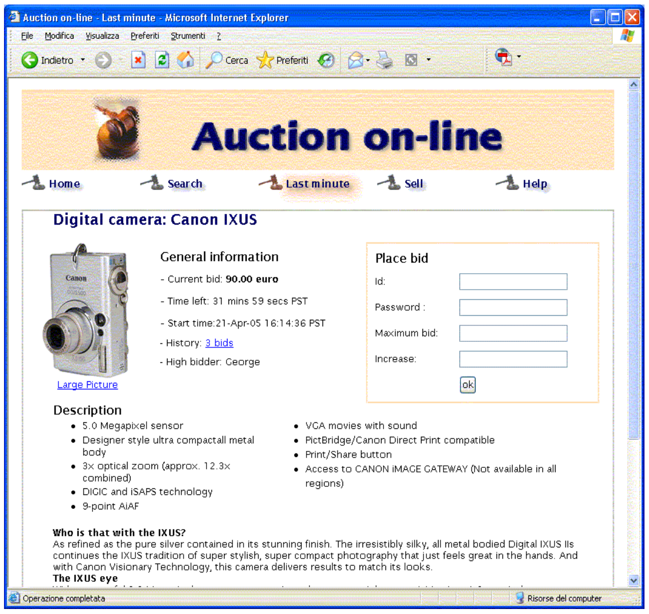
Since this architecture is based on logical descriptions independent of the specific interaction devices and corresponding implementation languages, it has a wide spectrum of application. So, it can be applied to any device and modality as long as there are transformations able to take the logical descriptions and generate the implementation in an implementation language supported by the device at hand. Our environment already supports generation in XHTML, XHTML Mobile Profile, VoiceXML, and SVG. In this paper we also introduce the new transformation for multimodal interfaces in X+V that can be supported by both desktop and mobile devices. However, the design of the corresponding user interface can vary to better account for the platforms' features, and thereby better exploit the possibilities opened up by the multimodality. In the event support for a new implementation language is required there is no need to modify either the entire architecture or the transformations for the logical user interfaces. We need only to add a new transformation from the low-level logical description (the concrete one) to the new implementation language. This usually takes a limited amount of time and effort.

6. EXAMPLE APPLICATION

In this section we present one scenario and the corresponding application in order to explain the features of the migration service and the multimodal interfaces obtained.

On Wednesday at 5 p.m. while George is finishing up at work, he receives an e-mail from an on-line auction service to which he is subscribed. The mail reminds him of the auctions that are expiring in an hour. He notices a digital camera and then he opens the main page of the auction service and accesses the "Last minute" page, where he can select the object of interest. In this page, George can read general information about the state of the auction, the object technical specifications and make a bid (see Figure 2). After reading the technical characteristics, he decides to participate in the auction. In the meantime, he must leave if he is to catch his train home and therefore George decides to migrate the interface to his PDA.

During the migration the interface becomes multimodal in order to better support the main tasks and not overload the visual channels. Thus, George can listen to/see all the information and make bids using both the vocal and graphical modalities.


Auction on-line - Last minute - Microsoft Internet Explorer
File Modifica Visualizza Preferiti Strumenti ?
Indietro
Cerca
Preferiti
Home
Search
Last minute
Sell
Help
Operazione completata
Risorse del computer

Auction on-line

Digital camera: Canon IXUS

Large Picture

General information

- Current bid: **90.00 euro**
- Time left: 31 mins 59 secs PST
- Start time:21-Apr-05 16:14:36 PST
- History: 3 bids
- High bidder: George

Place bid

Id:
Password :
Maximum bid:
Increase:
ok

Description

- 5.0 Megapixel sensor
- Designer style ultra compactall metal body
- 3x optical zoom (approx. 12.3x combined)
- DIGIC and iSAPS technology
- 9-point AiAF
- VGA movies with sound
- PictBridge/Canon Direct Print compatible
- Print/Share button
- Access to CANON iMAGE GATEWAY (Not available in all regions)

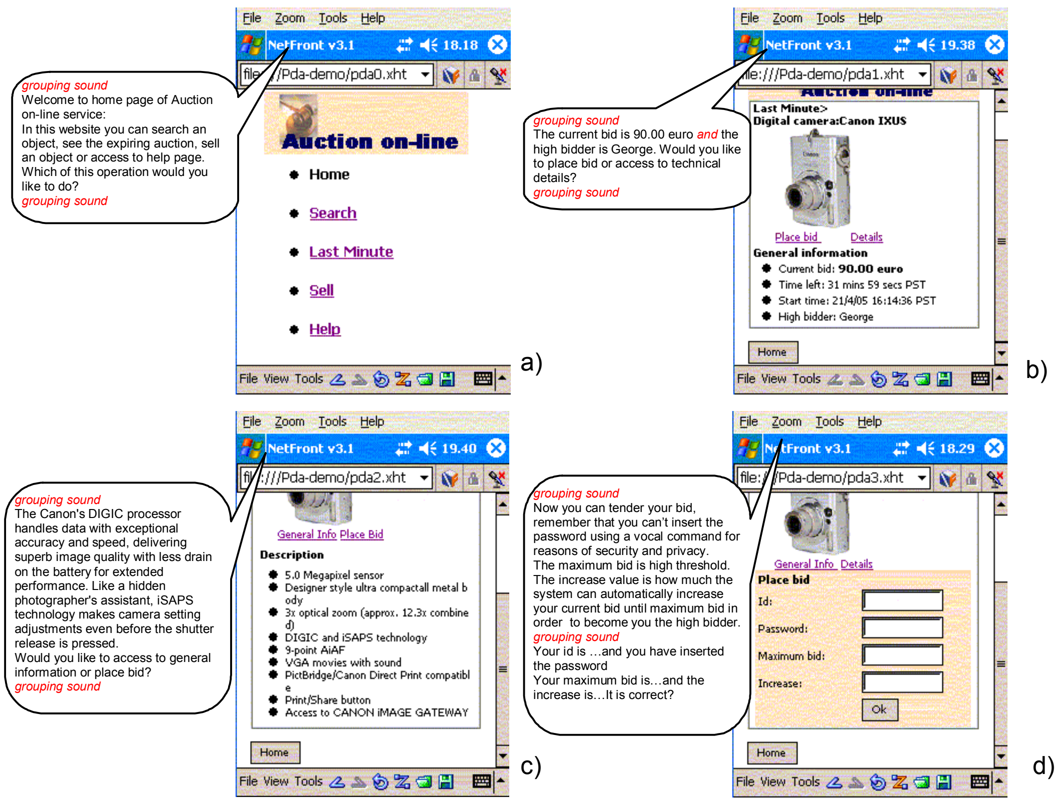
Who is that with the IXUS?
As refined as the pure silver contained in its stunning finish. The irresistibly silky, all metal bodied Digital IXUS IIs continues the IXUS tradition of super stylish, super compact photography that just feels great in the hands. And with Canon Visionary Technology, this camera delivers results to match its looks.
The IXUS eye

Figure 2: The desktop interface in the example considered.

Now, let us analyse the details of the two versions of the auction on-line service application, one for the desktop that supports only the graphical modality and one for the multimodal PDA that supports a combined vocal and graphical interface.

Figure 2 shows the desktop interface where users, in only one presentation, can see general auction information, make a bid, see details of objects and navigate through the auction on-line service Web site. Instead, the corresponding PDA interface is composed of four presentations: in the first presentation users can access the main pages of the Web site using the vocal or graphical modality, (in our case it is the "Last minute" page); in the second presentation users can see and listen to the general auction information; in the third, users can see and listen to the technical characteristics of the object; and lastly in the fourth presentation users can make bids using both vocal and graphical modalities (see Figure 3).

Another interesting difference is the use of the vocal modality in order to highlight some concrete objects on the interface, such as repeating the current high bid and bidder (in Figure 3.b) or the use of the vocal modality in order to provide some additional information, such as in Figure 3.c or in Figure 3.d. In the first case, we adopted a redundancy technique in order to draw attention to particular objects, whereas in the second case we adopted a complementary technique, because the limited capabilities of the PDA screen do not allow for presenting all the useful information though the graphical channel.



grouping sound
Welcome to home page of Auction on-line service:
In this website you can search an object, see the expiring auction, sell an object or access to help page.
Which of this operation would you like to do?
grouping sound

Auction on-line
- Home
- Search
- Last Minute
- Sell
- Help

grouping sound
The current bid is 90.00 euro *and* the high bidder is George. Would you like to place bid or access to technical details?
grouping sound

Last Minute>
Digital camera:Canon IXUS
Place bid Details
General information
- Current bid: **90.00 euro**
- Time left: 31 mins 59 secs PST
- Start time: 21/4/05 16:14:36 PST
- High bidder: George

Home

grouping sound
The Canon's DIGIC processor handles data with exceptional accuracy and speed, delivering superb image quality with less drain on the battery for extended performance. Like a hidden photographer's assistant, iSAPS technology makes camera setting adjustments even before the shutter release is pressed.
Would you like to access to general information or place bid?
grouping sound

General Info Place Bid
Description
- 5.0 Megapixel sensor
- Designer style ultra compactall metal b ody
- 3x optical zoom (approx. 12.3x combine d)
- DIGIC and iSAPS technology
- 9-point AiAF
- VGA movies with sound
- PictBridge/Canon Direct Print compatibl e
- Print/Share button
- Access to CANON iMAGE GATEWAY

Home

grouping sound
Now you can tender your bid, remember that you can't insert the password using a vocal command for reasons of security and privacy.
The maximum bid is high threshold.
The increase value is how much the system can automatically increase your current bid until maximum bid in order to become you the high bidder.
grouping sound
Your id is …and you have inserted the password
Your maximum bid is…and the increase is…It is correct?

General Info Details
Place bid
Id:
Password:
Maximum bid:
Increase:
Ok

Home

Figure 3 The Multimodal PDA Interface

It is also important to notice the different techniques used to implement the composition operators. For example, in the graphical desktop interface, the grouping operator is obtained through an unordered list or fieldset, whereas the multimodal PDA interface uses both sound to delimit the grouped elements and graphical techniques, such as an unordered list or fieldset. In addition, in the PDA version all concrete objects are aligned in columns, while in the desktop version many objects are aligned in rows, such as the menu to navigate the Web site, which is in columns in the PDA interface (see Figure 3.a), whereas it was in rows in the Desktop interface (see Figure 2). In the PDA interface version, users can provide any command by means of vocal or graphical modality, and the system provides graphical and vocal feedback of interaction.

Regarding the migration service, let us consider when the user has filled in the first two fields and then realises that it is getting late, so he decides to continue the auction with the multimodal application on the PDA (see Figure 4).

The first step is performed by the migration service in order to identify the multimodal presentation most similar to the source graphical presentation. During this operation the migration server compares the task set supported by the source presentation (graphical platform) with the task sets associated with the target presentations (multimodal mobile platform) and identifies the most similar abstract presentation of the multimodal abstract interface. In our case, the migration server identifies the presentation (fourth presentation) on the PDA that contains the object with which the user performed the last interaction (the password field) and continues on from this point, while maintaining the runtime data previously inserted via the graphical interface: identifier and password in the example (see Figure 4). The left side of Figure 4 shows the desktop graphical interface where the user has entered his id and password; while the right contains the corresponding multimodal interface for the PDA, with the fourth presentation highlighted, which is the presentation identified by the migration server for immediate activation. In the fourth presentation, the vocal interactions corresponding to those already performed on the graphical interface are dimmed to indicate that they are inactive and not prompted again in the multimodal PDA interface because their performance would be redundant.

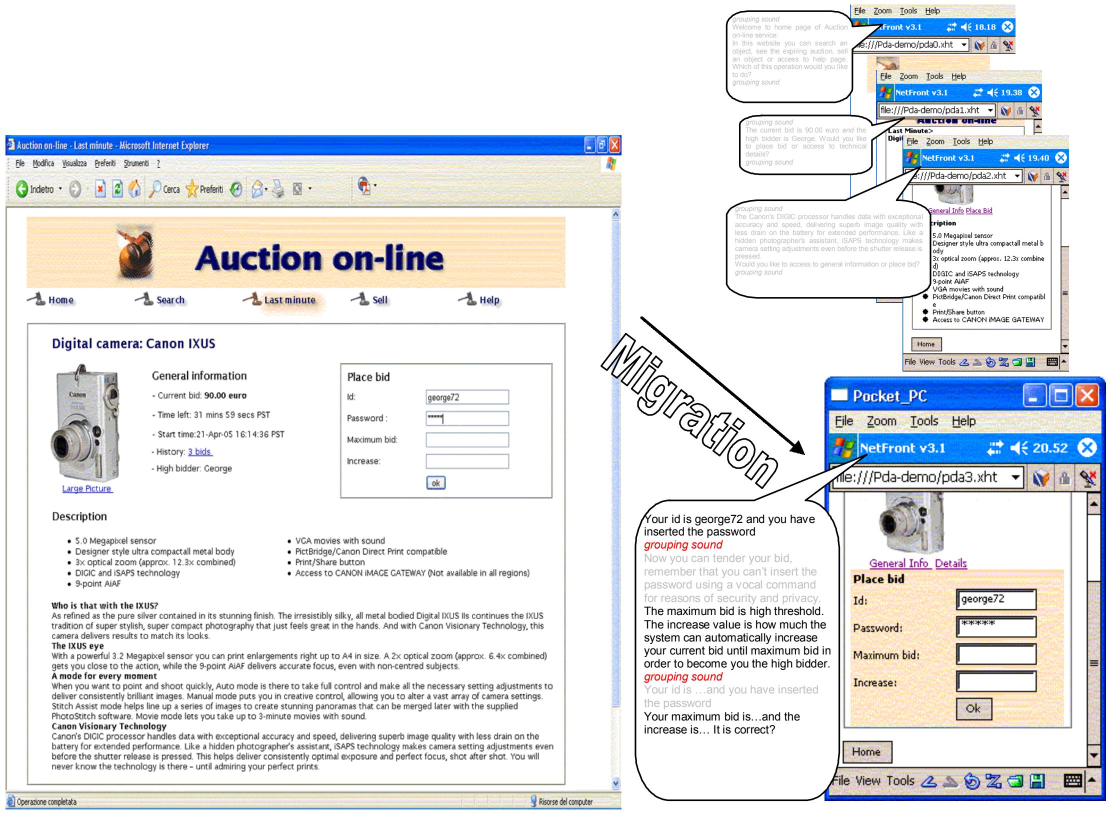
After the migration, the user can complete the bidding while moving about, choosing from many possibilities: only one vocal command, such as "increase my bid up to a maximum of 200 Euros in increments of 5 Euros" or multiple vocal commands or the graphical interface or both modalities in a complementary manner.

7. CONCLUSIONS AND FUTURE WORK

We have presented a solution to support multimodal user interfaces in multidevice environments. It is based on the use of conceptual descriptions, represented in XML-based, device-independent languages and a migration server. The system is able to support the necessary processing to obtain user interfaces that adapt to the features of the target device, while maintaining the state resulting from the user interactions in the previous one. One important aspect of our approach is that even when multimodality (such as combined use of graphical and vocal interaction) is supported, it is provided differently depending on the features of the device at hand.

We have already begun some empirical evaluation of our migration support. The first tests focused on the change of modality, in particular migrating from a graphical desktop to a vocal device and we decided to test if the users get disoriented during the migration from graphical to vocal interface and how they react at the possibilities provided by the migration service. While further empirical work will certainly be needed to investigate the usability of migratory interfaces, some useful suggestions have been obtained through this study. The interface of the migratory client can be improved with more direct selection of application and target devices. More importantly, users were not disoriented by the interface migration and appreciated the initial vocal feedback (summarising previous input through the source device) provided by the vocal device and, when given the chance to remove it, chose to keep it. The users barely noticed the difference in terms of number of tasks supported by the various devices. This means that the interfaces were able to adapt to the interaction resources available without creating sudden disruptions in users' expectations.

Future work will be dedicated to extending our environment in order to support additional modalities, such as gestural interaction.



Digital camera: Canon IXUS

General information

- Current bid: **90.00 euro**
- Time left: 31 mins 59 secs PST
- Start time:21-Apr-05 16:14:36 PST
- History: 3 bids
- High bidder: George

Place bid

Id:	george72
Password :	*****
Maximum bid:	
Increase:	

Description

- 5.0 Megapixel sensor
- Designer style ultra compactall metal body
- 3x optical zoom (approx. 12.3x combined)
- DIGIC and iSAPS technology
- 9-point AiAF
- VGA movies with sound
- PictBridge/Canon Direct Print compatible
- Print/Share button
- Access to CANON iMAGE GATEWAY (Not available in all regions)

Who is that with the IXUS?
As refined as the pure silver contained in its stunning finish. The irresistibly silky, all metal bodied Digital IXUS IIs continues the IXUS tradition of super stylish, super compact photography that just feels great in the hands. And with Canon Visionary Technology, this camera delivers results to match its looks.
The IXUS eye
With a powerful 3.2 Megapixel sensor you can print enlargements right up to A4 in size. A 2x optical zoom (approx. 6.4x combined) gets you close to the action, while the 9-point AiAF delivers accurate focus, even with non-centred subjects.
A mode for every moment
When you want to point and shoot quickly, Auto mode is there to take full control and make all the necessary setting adjustments to deliver consistently brilliant images. Manual mode puts you in creative control, allowing you to alter a vast array of camera settings. Stitch Assist mode helps line up a series of images to create stunning panoramas that can be merged later with the supplied PhotoStitch software. Movie mode lets you take up to 3-minute movies with sound.
Canon Visionary Technology
Canon's DIGIC processor handles data with exceptional accuracy and speed, delivering superb image quality with less drain on the battery for extended performance. Like a hidden photographer's assistant, iSAPS technology makes camera setting adjustments even before the shutter release is pressed. This helps deliver consistently optimal exposure and perfect focus, shot after shot. You will never know the technology is there - until admiring your perfect prints.

grouping sound
Welcome to home page of Auction on-line service:
In this website you can search an object, see the expiring auction, sell an object or access to help page. Which of this operation would you like to do?
grouping sound

grouping sound
The current bid is 90.00 euro *and* the high bidder is George. Would you like to place bid or access to technical details?
grouping sound

grouping sound
The Canon's DIGIC processor handles data with exceptional accuracy and speed, delivering superb image quality with less drain on the battery for extended performance. Like a hidden photographer's assistant, iSAPS technology makes camera setting adjustments even before the shutter release is pressed.
Would you like to access to general information or place bid?
grouping sound

Your id is george72 and you have inserted the password
grouping sound
Now you can tender your bid, remember that you can't insert the password using a vocal command for reasons of security and privacy.
The maximum bid is high threshold. The increase value is how much the system can automatically increase your current bid until maximum bid in order to become you the high bidder.
grouping sound
Your id is …and you have inserted the password
Your maximum bid is…and the increase is… It is correct?

Figure 4: The Migration in the example.

8. REFERENCES

[1] Abrams, M., Phanouriou, C., Batongbacal, A., Williams, S., Shuster, J., 1994. UIML: An Appliance-Independent XML User Interface Language, Proceedings of the 8th WWW conference.

[2] Balme, L. Demeure, A., Barralon, N., Coutaz, J., Calvary, G. CAMELEON-RT: a Software Architecture Reference Model for Distributed, Migratable, and Plastic User Interfaces. In Proceedings the Second European Symposium on Ambient Intelligence (EUSAI '04), LNCS 3295, Markopoulos et al. Springer-Verlag, Berlin Heidelberg, 2004, 291-302.

[3] Bandelloni, R., Berti, S., Paternò F., "Mixed-Initiative, Trans-Modal Interface Migration", Proceedings Mobile HCI 2004, Glasgow, September 2004, Lecture Notes Computer Science 3160, pp.216-227, Sprinter Verlag..

[4] Bouchet J., Nigay L., ICARE: A Component-Based Approach for the Design and Development of Multimodal Interfaces, Adjunct Proceedings CHI 2004, pp.1325-1328.

[5] Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L., Vanderdonckt, J. A Unifying Reference Framework for Multi-Target User Interfaces. Interacting with Computers. Vol. 15, No. 3, June 2003, pp. 289-308.

[6] Coutaz J., Nigay L., Salber D.,.Blandford A, May J., Young R., 1995. Four Easy Pieces for Assessing the Usability of

Multmodal Interaction: the CARE properties. Proceedings INTERACT 1995, pp.115-120

[7] Channarukul S., McRoy S. W.,. Ali S. S. MULTIFACE: Multimodal Content Adaptations for Heterogeneous Devices.

[8] Garlan, D., Siewiorek, D., Smailagic, A., Steenkiste, P. Project Aura: Toward Distraction-Free Pervasive Computing. IEEE Pervasive Computing, Vol 21, No 2 (April-June 2002), 22-31.

[9] Limbourg, Q., Vanderdonckt, J., UsiXML: A User Interface Description Language Supporting Multiple Levels of Independence, in Matera, M., Comai, S. (Eds.), «Engineering Advanced Web Applications», Rinton Press, Paramus, 2004,

[10] Bandelloni, R., Paternò, F. Migratory User Interfaces Able to Adapt to Various Interaction Platforms. In International Journal of Human Computer Studies 60, pp. 621-639, Elsevier, 2004.

[11] Nichols, J. Myers B. A., Higgins M., Hughes J., Harris T. K., Rosenfeld R., Pignol M., 2002. “Generating remote control interfaces for complex appliances”. Proceedings ACM UIST’02, pp.161-170.

[12] Obrenovic, Z., Starcevic D., Selic B., A Model-Driven Approach to Content Repurposing, IEEE Mutimedia, Januray March 2004, pp.62-71.

[13] Paternò, F., “Model-Based *Design and Evaluation of Interactive Application”*. Springer Verlag, ISBN 1-85233-155-0, 1999.

[14] Pérez-Quiñoes, M, A., Capra, R, G., Shao, Z.. The Ears Have It: A Task by Information Structure Taxonomy for Voice Access to Web Pages. In Proceedings of INTERACT 2003, IOS Press.

[15] Puerta, A., Eisenstein, XIML: A Common Representation for Interaction Data, Proceedings ACM IUI’01, pp.214-215.

[16] Szekely, P., 1996. Retrospective and Challenges for Model-Based Interface Development. 2nd International Workshop on Computer-Aided Design of User Interfaces, Namur, Namur University Press.