Making User Interface Adaptation in Multi-Device Environments Understandable to End Users

Fabio Paternò, Christian Sisti, Giuseppe Zichitella
CNR-ISTI, HIIS Laboratory
Via Moruzzi 1
56124 Pisa, Italy
fabio.paterno@isti.cnr.it

Abstract
In this paper we discuss issues and solutions for making understandable context-dependent adaptive user interfaces. The discussion considers an approach based on the use of declarative user interface languages and oriented to Web applications.

Keywords
Adaptive User Interfaces, Model-base user Interface Languages, User Control

ACM Classification Keywords
H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms
Design

Introduction
Emerging ubiquitous environments call for supporting access through a variety of interactive devices. Web applications are widely used and a number of approaches and techniques have been proposed to support their adaptation to mobile devices (e.g. [1]). Little attention has been paid so far on identifying more general solutions to adapt Web applications for a variety of devices, including vocal access. We have been working for various years on solutions for adaptive user interfaces, which aim to support any Web application independent of the type of authoring environment used for its development. We have focused on adaptation to various possible interactive devices, but the approach is extensible to consider other contextual factors, such as user preferences, environmental aspects (such as position, noise, ..) and social aspects.

An example of related work is DANTE [3], which provides semantic transcoding for visually-disabled users. For this purpose it uses ontology-based annotations that are manually created and included in the Web page. Such annotations are then used for automatically transforming the original pages to obtain pages more easily accessible for visual-impaired users. In our case, we want to provide a tool that does not require any manual intervention on the application code and is able to automatically analyse the content of a Web page and transform it into an application for the target device. For this purpose we consider intermediate model-based representations [2], which aim to represent the logical structure of the user interface and enable relevant reasoning to derive a more suitable structure for the target device. Given this type of approach, we want to investigate novel solutions that allow end user to customize the effects of the adaptation transformations.

In the position paper we discuss in particular adaptation to mobile devices and for vocal browsing and indicate preliminary ways to customize them ([5][6]). In both cases we provide designers with the possibility of customizing the adaptation rules, in order to obtain results that better suit their preferences.

Architecture of the Solution

We exploit the model-based framework MARIA [4] for performing a more semantically-oriented transformation. The framework provides abstract (independent of the interaction modality) and concrete (dependent of the interaction modality but independent of the implementation languages) languages. Such languages share the same structure but with different levels of refinements. A user interface is composed of a number of presentations. In each presentation there are groupings that identify the main logical parts and contain the interface elements, which are called interactors. Examples of interactors are navigators (allow moving from one presentation to another), activators (allow triggering functionality)…

Our solution is based on an adaptation server, which provides a number of functionalities:

- *Reverse engineering*, it automatically parses the content of the Web page and the associated style sheets and scripts, and builds a corresponding concrete logical description;
- *Orchestrator,* it receives contextual information and depending on it decides which adaptation transformation to apply*;*
- *Adaptation*, the graphical concrete description provided by the reverse module is transformed into a concrete description adapted for the target device, there are various adapters for each target platform;
- *Generation*, an implementation of the concrete description for the target device is generated.

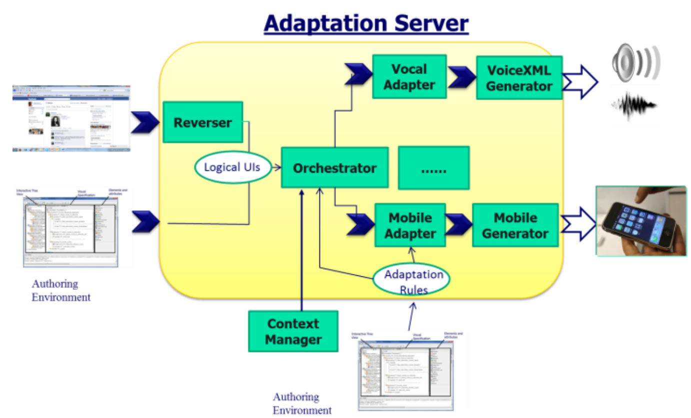

figure 1. The architecture of our approach.

Adaptation to Mobile Devices

In the transformation for adaptation to mobile devices various aspects are considered. The starting point is the concrete logical description in MARIA of the page currently accessed. First, if the user has expressed some preferences regarding attributes for the target mobile device (e.g. in terms of font sizes), they are applied to the available content. Then, there is a calculation of the cost of such content in terms of screen size requested for its presentation. The result is then compared to what can actually be presented in the current device. If the cost is not sustainable by it then there is a phase in which the user interface elements are reduced without changing the structure of the user interface. This means that the images are resized maintaining the height/width ratio, long texts are reduced, interactors are replaced with other semantically equivalent but occupying less screen space (e.g. a radio-button can be replaced by a pull-down menu). The cost of the resulting presentation is calculated and compared again with the sustainable cost of the current mobile device. If it still exceeds it then page splitting is performed. In this case the algorithm calculates the cost of each groups of elements in the presentation in order to identify groups of elements that can be removed from the current presentation and associated with a newly generated mobile presentation. This is done recursively until presentations sustainable by the current mobile device are obtained. The basic idea is that in this way groups of elements that have an autonomous identity are identified and then they are allocated to a new mobile presentation, still providing a meaningful set of information. Two options are possible in this page splitting transformation: one is to allocate the most expensive group of elements to the new mobile presentation. In the other case the algorithm first calculates the groups whose removal would make the current presentation sustainable by the target device, and then selects the one that has the lowest cost. The rationale for this second option is that it allows obtaining a sustainable presentation but by removing the least amount of information possible, thus preserving the original design as much as possible.

Customizing Adaptation to Mobile Devices

We also provide the possibility to customize such adaptation rules by the end users through a form-based user interface, which allows them to indicate whether they want to apply specific rules and to provide parameters that can affect the final result. For example, when the sustainable cost of the current device is calculated it is possible to change it by indicating how much scrolling would be acceptable in the navigation. Thus, the overall acceptable cost will be calculated by considering the screen size of the current device and such scrolling coefficients. To clarify how it works we can take an example Web page (Wikipedia, Figure 2).

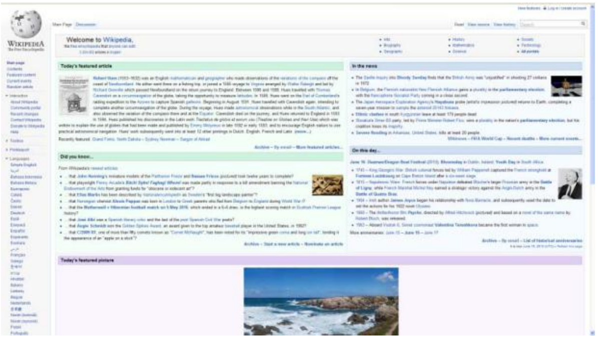

figure 2. An example page.

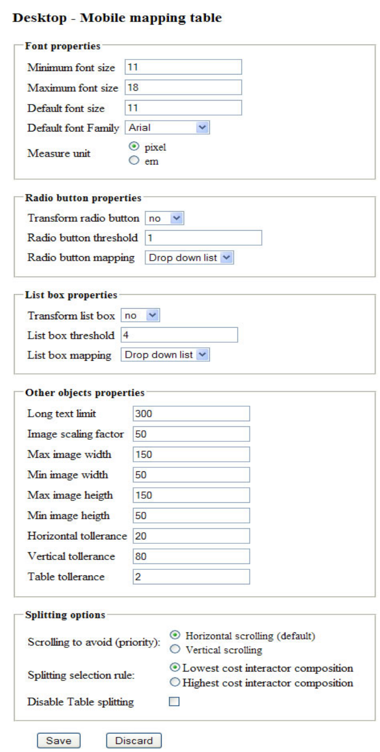

figure 3. Customization tool for desktop-mobile adaptation.

Figure 3 shows the user interface of the tool for customization. It groups various types of properties. Some concern general attributes (e.g. font attributes), others refer to specific interactor types (e.g. radio buttons, listboxes). Then, we have the attributes that can have an impact on whether or not to perform page splitting and to what extent. Figure 4 shows the result of adaptation on the example page with the parameters shown in Figure 3.

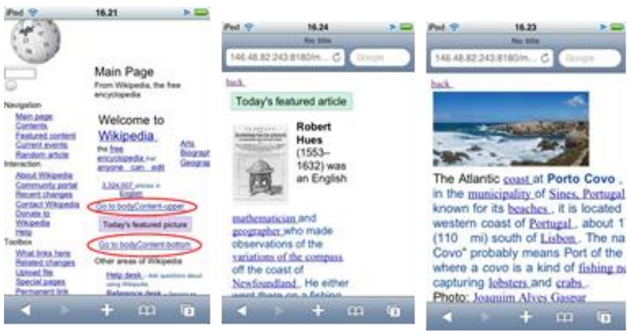

figure 4. Result of desktop-mobile adaptation.

As you can see the result consists of three mobile pages, in the first one a couple of links (circled in red) to access the other two have automatically been included. However, if the customization parameters are changed, for example increasing the vertical tolerance and indicating that the scrolling to avoid is vertical then we can obtain the results shown in Figure 5, in which only two mobile presentations are generated.

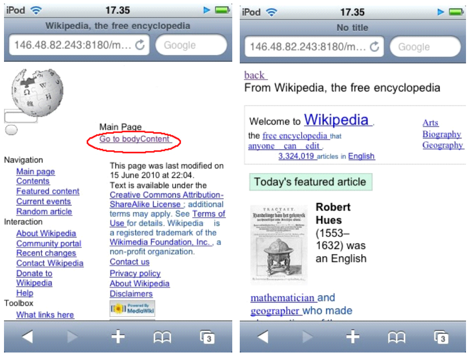

figure 5. Another result of desktop-mobile adaptation

Adaptation to Vocal Devices

Little attention has been paid so far on how to adapt Web applications for vocal access. This seems useful and important given the recent improvements in vocal technologies (see for example iPhone SIRI and Google voice), which can allow users to access their applications when the visual channel is busy (e.g. when driving) or when they are visually-impaired. The adaptation transformation for vocal devices is composed of three main phases (graphical description processing, main vocal menu generation, graphical-to-vocal mapping).

Graphical Description Processing

This phase performs the content optimization by removing from the graphical logical description the elements that badly reflect into vocal interfaces (e.g., images without alternative text); then it performs the structure optimization by recognizing the page components and removing the unnecessary one (e.g., grouping elements used for only formatting purposes); lastly, it performs the calculate cost computation, whose results are used in the menu generator phase.

Main Vocal Menu Generation

The next step is to build up a menu structure allowing the users to navigate inside the associated content. One of the main problems is how to find keywords (or short sentences) that summarize the content of the destination link. For this purpose, when the developers of the original Web pages indicated the ID value, we suppose that that value is meaningful and it is used for the vocal selection, otherwise the tool performs an analysis of the grouping content and uses the string of the first description/text with the highest text role. The text roles indicated in the graphical concrete description are sorted from the most important to the less one in this way. If the string is a sentence and not a simple word we use the first three words in the sentence.

Graphical-to-Vocal Mapping

This is a transformation that takes the elements of the graphical concrete description and maps them into corresponding elements of the vocal concrete description that have similar effects. Each graphical presentation is often transformed into more than one vocal presentation because, while in a graphical environment a presentation is associated with all the elements perceivable at a given time, in a vocal environment it refers to a logically complete dialogue. Each such vocal presentation will then be mapped onto a VoiceXML document. Thus, we create nesting among the VoiceXML documents that contain the vocal forms. In this way we solve the issue that the VoiceXML forms cannot be nested as it happens with containers (e.g. div, tables, …) in graphical pages.

Customizing Adaptation to Vocal Devices

The adaptation process is really complex and the results depend on a number of factors, such as the structure of the Web pages in input and their conformance to the accessibility guidelines. In order to obtain better results we have designed a customization tool, which allows the user to customize the adaptation results.

The adaptation process can be driven setting a number of parameters. Such parameters can influence different states of the transformation process.

To adjust the *pre-conversion* step it is possible to indicate whether text normalization should be carried out and the voice browser character set.

It is also possible to indicate the maximum number of elements that a vocal menu should have. Each main construct can be associated with a weight, which has an impact on how to derive the structure of the corresponding vocal interface.

Finally, to customize the desktop-to-vocal transformation we have parameters that set how the final vocal interface will perform the multiple choice. There are two solutions: *Yes/No Questions,* for every possible choice the platform will ask a Yes/No confirmation to the user; *Grammar Based*: the user can select more than one possible choice with one single sentence (listing the choices in sequence). There is also a parameter to decide if each vocal dialogue should terminate with a short sound. The voice speed and gender can be chosen as well.
Figure 6 shows our tool that allows such customization. The left panel contains some modifiable parameters and their default's values.

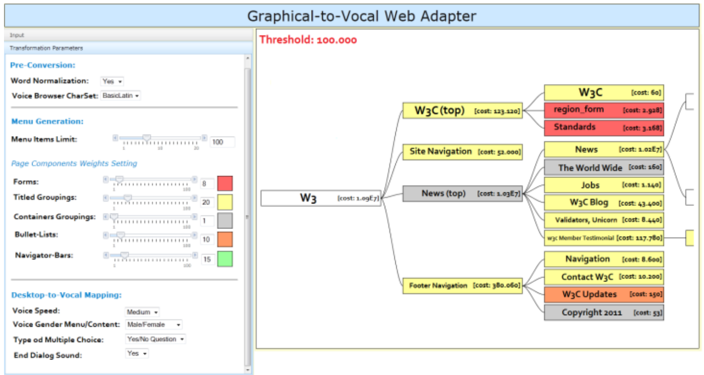

Figure 6. Customization of the desktop-to-vocal adapter.

The right panel shows the structure and the menu items of the generated vocal page. In this way the designer can decide whether to download the final vocal interface or change the transformation parameters in order to obtain a different structure.

Conclusions and Future Work

We have presented an approach to adaptation in multi-device environments. We have introduced how it is implemented for a couple of possible target platforms (mobile and vocal devices), and shown the current level of end user customization. We are aware that some users can have difficulties in understanding the customization parameters and their effects.

Future work will be dedicated to making more controllable the adaptation rules for end users through more effective representations of the customization parameters.

Acknowledgements

We thank the EU ICT SERENOA Project for the support of this research work.

References

[1] Buyukkokten, O., Kaljuvee, O., Molina, H., Paepcke, A., and Winograd, T. 2002. Efficient Web browsing on handheld devices using page and form summarization. ACM Trans. Inform. Syst. 20, 1, 82-115.

[2] Fonseca J.M.C. (ed.), W3C Model-Based UI XG Final Report, May 2010, available at http://www.w3.org/2005/Incubator/model-based-ui/XGR-mbui-20100504/

[3] Yesilada, Y., Stevens, R., Harper, S. and Goble, C. Evaluating DANTE: Semantic transcoding for visually

disabled users. ACM Transactions on Human-Computer Interaction 14, 3, (2007), Article 14.

[4] Paternò F., Santoro C., Spano L.D.: MARIA: A universal, declarative, multiple abstraction-level language for service-oriented applications in ubiquitous environments. ACM Trans. Comput.-Hum. Interact. 16(4): (2009).

[5] Paternò F., Sisti C., Model-Based Customizable Adaptation of Web Applications for Vocal Browsing, ACM SIGDOC, pp.83-90, ACM Press, Pisa, October 2011.

[6] Paternò F., Zichitella G., Desktop-to-Mobile Web Adaptation through Parametric Bidimensional Semantic Redesign, Proceedings HCSE 2010, 3rd Conference on Human-Centred Software Engineering, pp.79-84, LNCS 6409 Springer, Reykjavik, October 2010.