Intelligent Support for Nomadic Web Applications Personalization

José Macías and Fabio Paternò

ISTI-CNR
Via G.Moruzzi 1
56124 Pisa, Italy
{Jose.Macias, Fabio.Paterno}@isti.cnr.it

Abstract: The increasing availability of different types of interaction devices raises the need of applications able to adapt to the changing interaction resources. Even systems able to generate interfaces able to adapt taking into account usability criteria may not meet all the specific user preferences. In this paper, we present an approach able to allow users to specify examples of desired changes through direct manipulation of the user interface, and then analyse such changes in order to derive general preferences. The analysis of the user changes is done at a logical level that can be useful to better identify semantic aspects in the users' request. Our approach is mostly addressed to support customization of Web nomadic applications with minimal assumptions about user programming skills.

1 Introduction

The explicit customization of software applications is likely to be considered a cumbersome task that most of non-computer-skilled end-users cannot afford. The implicit complexity of programming and specification languages takes users apart from carrying out software customization [MPC06]. Customisation of interactive Web applications still requires considerable skill in programming and Web technology. Some preliminary studies indicate that these limitations in user Web development activities are not due to lack of interest but rather to the difficulties inherent in interactive web development [RRP06]. Providing users with real facilities to modify existing applications is not yet as widespread as one would expect. However, some researchers have devoted considerable effort to address such issues. This involves End User Development (EUD) research [LPW06], where the main concern is to ease how users can modify and create software artefacts. Programming by Demonstration [L01] is one of the most relevant efforts in EUD for obtaining a real trade-off between ease of use and expressiveness. Programming by Demonstration has the potential to allow users to customize their applications. Rather than writing a program in an abstract programming language to automate a task, users demonstrate how to perform the task. In our case, we want to allow users to modify the existing interface of a nomadic application and that the system be able to learn and generalize design patterns that can be applied in new applications that can be accessed through different platforms. We mean for nomadic application an application that can be accessed through different types of devices.

In order to address the problem of customizing nomadic applications by end-users, an intelligent approach is here proposed, which is able to analyse the different changes that the users make to a nomadic user interface by means of a standard authoring tool. The user provides the system with an example of want s/he wants to modify and the system analyses the modifications at the server side for a given user and platform. Additionally, our approach supports an intelligent mechanism in order to infer and generalize design patterns from the modifications performed. Such design patterns will be used for customizing automatically the application for the given user and platform.

2 Our Approach

Our approach comprises a EUD environment intended to provide the user with an easy mechanism to freely customize user interfaces for a given platform, supported by a nomadic application. Figure 1 describes the main architecture of our approach. The scenario consists of a final user navigating through a Web application on a specific platform. The application server generates Web interfaces for different platforms, so the user can access the application using a desktop computer, laptop, mobile, PDA and vocal interface. Reverse engineering techniques are applied in order to get high-level information about the user's modifications performed through an authoring environment. In particular, our approach supports the following steps (see Figure 1):

1) The nomadic application Web server generates platform-dependent user interfaces.

2) An end-user navigates through the information and, at some point, s/he decides to modify something by using a standard web authoring tool, such as Macromedia Dreamweaver, which supports modifications by direct manipulation of the interface elements. In order to ease the modification process, the changes are performed on a desktop platform, which has larger screens, even if the modified interface is for a different platform (mobile or vocal).

3) Once the user has finished the changes, s/he sends back the modified pages to the server, by using a special Web application in which s/he first needs to login.

4) The server receives the modified Web page and then starts the inference process to identify user preferences. First, the server transforms the modified page into a logical description stored into an XML file, using the reverse mechanism developed by the HIIS Laboratory [BMP05]. The logical description is specified in terms of implementation language-independent elements. In addition, the system compares the logical description corresponding to the modified page, with the logical description of the originally generated page. In the comparison process, the system also generates high-level information in order to indicate whether the desired modifications imply changes only at the syntactical level or also the semantic abstract description is involved. At the end of the process, the system builds an End-User Profile taking into account all the information inferred.

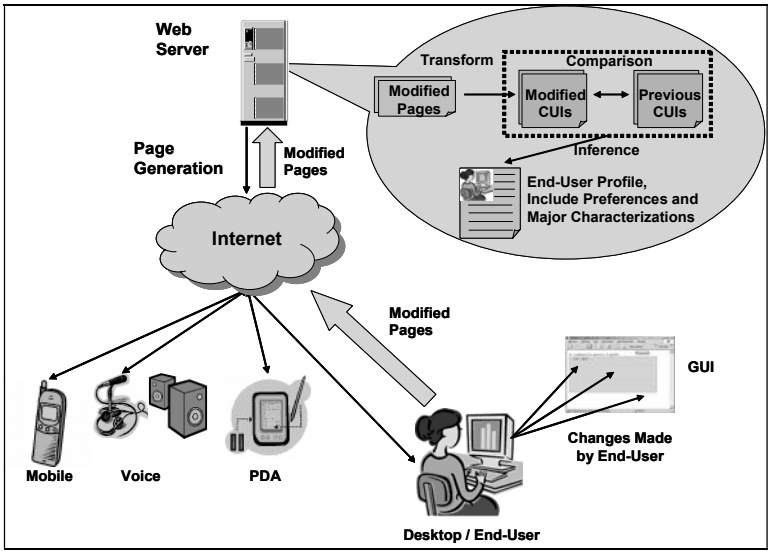

Figure 1. End-user framework for inferring user's preferences in customizing nomadic presentations

5) Such End-User Profile will be used in order to generate again the Web interface, taking into account preferences and personal customization. The system stores an End-User Profile for each user and platform, controlling which aspects of the generated interface could be relevant for each one.

The information included in the profile is updated every time the user sends back a modified Web page to the server. The most relevant information stored by the End-User Profile is the set of Interface Rules. Such rules are inferred from the logical descriptions comparison and aim to reflect the knowledge acquired from the user's changes. We consider two levels of logical user interface descriptions: the concrete level is an implementation-independent but platform-dependent description, while the abstract one is also a platform-independent language. The rules are used for driving the generation of the Web pages after the changes, customizing the Web presentation and navigation depending on the inferred preferences. The rules are based on knowledge acquisition algorithms and targeted at obtaining information regarding user's intents in order to characterize some preferences for customization purposes. Such information can be modelled by means of both a knowledge base and a set of rules to be applied when new information about user's modifications is identified. When the modifications just involve the concrete level then a syntactical change is detected, which usually means that some attributes of an interface elements have been changed. When they imply changes at the abstract level then it means that some semantic modification has been performed because the logical structure has varied somehow because of interactors deleted or added, or changes in the type of interaction supported by one interactor or changes in the way to compose the available interactors.

Conclusions

We have presented an overview of an approach intended to customize nomadic Web applications. Our approach aims to integrate Model-Based User Interface design [P99] and End-User Development techniques [MC04], in order to support the customization of nomadic applications accessible through different types of platforms [BPS06]. This solution allows the end-user to modify elements of the user interface by using a direct manipulation tool in order to indicate examples of desired interfaces. The changes are uploaded to a server where they are analysed at the level of the corresponding logical descriptions in order to better characterise such changes and identify possible general preferences to apply when the use accesses again the nomadic application.

We have carried out a first user test. The outcomes obtained provided interesting indications about user's preferences by detecting user's changes to a Web application. The changes observed can be classified as either syntactical or semantical changes, in the case that they change the logical structure of the application itself, described in terms of logical interactors and ways of composing them. Furthermore, the information inferred helps the environment to identify interaction patterns that can be applied in order to define future customisation. During the test, user questionnaires have also been applied to receive feedback regarding the perceived system's ease-of-use, with positive results.

References

[BMP05] Bandelloni, R., Mori, G. and Paternò, F. Dynamic Generation of Web Migratory Interfaces. Proceedings MobileHCI'2005. September 19-22. Salzburg, ACM Press, pp. 83-90. 2005.

[BPS06] Berti, S., Paternò, F. and Santoro, C.: Natural Development of Nomadic Interfaces Based on Conceptual Descriptions. Lieberman, H., Paternò, F., and Wulf, V. (eds): End-User Development. Human Computer Interaction Series. Springer Verlag, pp 143-160. 2006.

[L01] Lieberman, H. (ed.): Your Wish is my Command. Programming by Example. Morgan Kaufmann Publishers. 2001.

[LPW06] Lieberman, H., Paternò, F., and Wulf, V. (eds): End-User Development. Human Computer Interaction Series. Springer Verlag, 2006.

[MC04] Macías, J.A., and Castells, P.: An EUD Approach for Making MBUI Practical. Proceedings First International Workshop on Making model-based user interface design practical. Vol. 103. Online http://CEUR-WS.org/Vol-103. 2004.

[MPC06] Macías, J.A., Puerta, A. and Castells, P.: Model-Based User Interface Reengineering. HCI Related Papers of Interacción 2004. Jesús Lorés y Raquel Navarro (eds.). Springer-Verlag, pp 155-162. 2006.

[P99] Paternò, F.: Model-Based Design and Evaluation of Interactive Applications. Springer Verlag, 1999.

[RRP06] Rode, J., Rosson, M.B. and Pérez, M.A. : End-User Development of Web Applications. Lieberman, H., Paternò, F., and Wulf, V. (eds): End-User Development. Human Computer Interaction Series. Springer Verlag, pp. 161-182. 2006.