TERESA: A Transformation-based Environment for Designing and Developing Multi-Device Interfaces

Silvia Berti, Francesco Correani, Giulio Mori, Fabio Paternò, Carmen Santoro
I.S.T.I.-C.N.R.
Via G.Moruzzi 1
56124, Pisa - Italy
+39 050 3153066
{s.berti, f.correani, g.mori, f.paterno, c.santoro}@isti.cnr.it

Abstract
The ever-increasing availability of new types of devices raises a number of issues for user interface designers and interactive software developers. We have designed and developed a model-based authoring environment (TERESA), which provides support when designing and developing interfaces accessible through various device types in Web-based environment.

Categories & Subject Descriptors: H.5 INFORMATION INTERFACES AND PRESENTATION – I2.2 Automatic Programming: Program Transformation.

General Terms: Design, Human Factors.

Keywords: Model-based Design, User Interface Transformations, Heterogeneous Devices

INTRODUCTION

With the advent of the wireless Internet and the rapidly expanding market of smart devices, designing interactive applications supporting multiple platforms has become a difficult issue. The main problem is that many assumptions that have been held up to now about classical stationary desktop systems are being challenged when moving towards nomadic applications, which are applications that can be accessed through multiple devices from different locations. Consequently, one fundamental issue is how to support software designers and developers in building such applications: in particular, there is a need for novel methods and tools able to support development of interactive software systems able to adapt to different targets while preserving usability.

Model-based approaches [2] could represent a feasible solution for addressing such issues: the basic idea is to identify useful abstractions highlighting the main aspects that should be considered when designing effective interactive applications. Our approach extends previous work in the model-based design area in order to support development of nomadic applications. In particular, we have designed and developed the TERESA (Transformation Environment for inteRactivE Systems representAtions) tool providing general solutions that can be tailored to specific cases. This tool supports transformations in a top-down manner, providing the possibility of obtaining interfaces for different types of devices from logical descriptions. It differs from other approaches such as UIML [1], which mainly consider low-level models.

CHI 2004, April 24–29, 2004, Vienna, Austria.
ACM 1-58113-703-6/04/0004.

THE METHOD

Our method for model-based design is composed of a number of steps that allows designers to start with an overall envisioned task model of a nomadic application and then derive concrete and effective user interfaces for multiple devices:

- *High-level task modelling of a multi-context application*. In this phase designers develop a single model that addresses the possible contexts of use and the various roles involved and also a domain model aiming to identify all the objects that have to be manipulated to perform tasks and the relationships among such objects. Such models are specified using the ConcurTaskTrees notation [2], which also allows designers to indicate the platforms suitable to support each task.
- *Developing the system task model for the different platforms considered*. Here designers have to filter the task model according to the target platform and, if necessary, further refine the task model, depending on the specific device considered, thus, obtaining the system task model for the platform considered.
- *From system task model to abstract user interface*. Here the goal is to obtain an abstract description of the user interface composed of a set of presentations that are identified through an analysis of the task relationships. Each presentation is structured by means of interactors composed of various operators.
- *User interface generation*. In this phase we have the generation of the user interface. This phase is completely platform-dependent and has to consider the specific properties of the target device.

THE TOOL

TERESA is intended to provide a complete semi-automatic environment supporting a number of transformations useful for designers to build and analyse their design at different abstraction levels and consequently generate the user interface for various types of platforms. A number of main requirements have driven the design and development of TERESA:

- *Mixed initiative*; we want a tool able to support different level of automations ranging from completely automatic solutions to highly interactive solutions where designers can tailor or even radically change the solutions proposed by the tool.
- *Model-based*, the variety of platforms increasingly available can be better handled through some abstractions that allow designers to have a logical view of the activities to support.
- *XML-based*, each abstraction level considered can be described through a XML-based language.
- *Top-down*, this approach is an example of forward engineering. So, designers first have to create more logical descriptions, and then move on to more concrete representations until the final interface.
- *Different entry-points*, our approach aims to be comprehensive and to support various possibilities, including also when different set of tasks can be performed on different platforms. However, there can be cases where only a part of it needs to be supported and, for example, designers want to start with a logical interface description and not with a task model.
- *Web-oriented*, we decided that Web applications should be our first target. However, the approach can be easily extended to other environments (such as Java applications, Microsoft environments, …) by just modifying only the last transformation (from concrete interface to final interface).

The TERESA tool offers a number of transformations and provide designers with an integrated environment for generating XHTML interfaces for desktop, mobile phones and VoiceXML user interfaces. With the TERESA tool, at each abstraction level the designer is in the position of modifying the representations while the tool keeps maintaining forward and backward the relationships with the other levels. For example, it maintains links between abstract interaction objects and the corresponding tasks in the task model so that designers can immediately identify their relations. This results in a great advantage for designers in maintaining a unique overall picture of the system, with an increased consistence among the user interfaces generated for the different devices and consequent improved usability for end-users.

Once the elements of the abstract user interface have been identified, every interactor has to be mapped into interaction techniques supported by the particular device configuration considered (characterised by the modalities supported, the screen size, …), and also the abstract operators have to be appropriately implemented by highlighting their logical meaning: a typical example is the set of techniques for conveying grouping relationships in visual interfaces by using presentation patterns like proximity, similarity and continuity. However, different techniques for grouping elements are used in case of vocal interfaces, such as using a specific sound to delimit a set of elements.

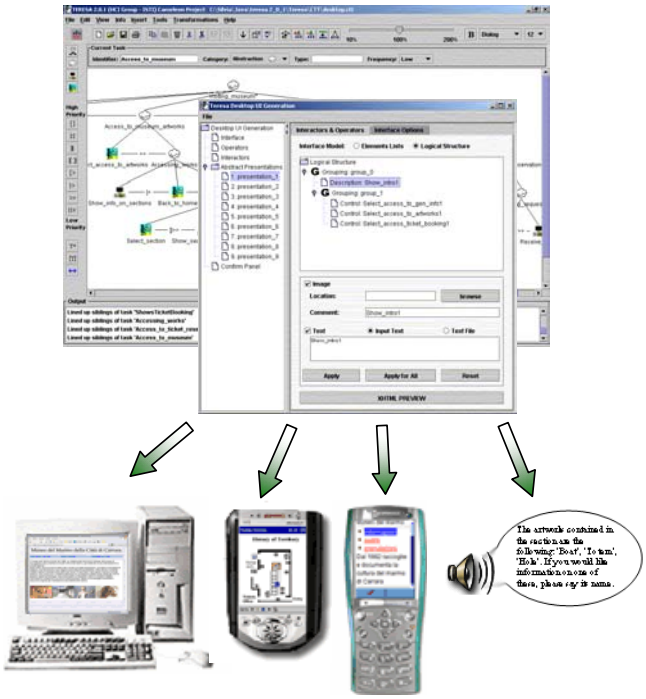

Figure 1: The TERESA tool.

CONCLUSIONS

The environment presented supports many features to allow designers to design and develop multi-platform user interfaces through a number of transformations that can be performed either automatically or through interactions with the designer. It helps designers clarify design issues and support analysis and evaluation of design options. It can be freely downloaded at http://giove.cnuce.cnr.it/teresa.html.

ACKNOWLEDGMENTS

We gratefully acknowledge support from the CAMELEON project (http://giove.cnuce.cnr.it/cameleon.html).

REFERENCES

1. Abrams, M., Phanouriou, C., Batongbacal, A., Williams, S., Shuster, J. *UIML: An Appliance-Independent XML User Interface Language*, Proceedings of the 8th WWW conference, 1999.
2. Paternò, F., Model-Based Design and Evaluation of Interactive Application. Springer Verlag, ISBN 1-85233-155-0, 1999.