

Designing Multi-device User Interfaces: How to Adapt to the Changing Device

Fabio Paternò

ISTI-CNR, Via Moruzi. 1,
56124 Pisa, Italy
Fabio.Paterno@isti.cnr.it

Abstract. Nowadays, everyday life is becoming a multi-platform environment where people are surrounded by different types of devices through which they can connect to networks in different ways. Most of them are mobile personal devices carried by users moving freely about different environments populated by various other devices. Such environments raise many issues for designers and developers, such as the possibility of obtaining user interfaces able to adapt to the interaction resources of the available devices. The main learning objective is to gain knowledge and skills in methods and tools for the design of multi-device interfaces that can support designers and developers to address a number of issues raised by ubiquitous computing.

Keywords: Multi-device interfaces, Model-based design, Ubiquitous Environments

1 Tutorial Content

The tutorial starts with an introduction to multi-device interfaces, their importance and the issues that they raise in order to address adaptation to the changing interaction resources. Particular attention is dedicated to explaining how the device choice has an influence on the possible tasks to accomplish and how the structure of such tasks can vary in terms of possible secondary tasks, tasks' temporal relations, and content requires depending on the device.

Next, the tutorial provides an overview concerning results that can be obtained through model-based approaches when multi-device interfaces, even using different modalities, are considered, and will link up the discussion to projects currently underway. Indeed, as Myers, Hudson, and Pausch [1] indicated, while discussing the future of user interface tools, the wide platform variability encourages a return to the study of some techniques for device-independent user interface specification, so that developers can describe the input and output needs of their applications, so that vendors can describe the input and output capabilities of their devices, and so that users can specify their preferences. Then, the system might choose appropriate interaction techniques taking all of these into account. The basic idea is that instead of having separate applications for each device that exchange only basic data, there is some abstract description and then an environment that is able to suggest a design for a specific device that adapts to its features and possible contexts of use. Thus, a key

aspect is to be able to have different views on interactive systems, each view associated with a different abstraction level. With the support of tools, XML-based languages and transformations, it is possible to move from one level to another and convert a description for one interaction platform to another for a different one.

Then, the tutorial shows how these concepts can be incorporated in authoring environments for multi-device interfaces and discuss how to address a variety of platforms with different modalities (such as graphical and vocal interfaces, digital TV, tilt-based interaction, ...). Examples of such tools will be discussed, along with a demo of one of them (Multimodal TERESA [2]), which is publicly available and developed by the instructor's group.

The second unit of the tutorial is dedicated to run-time support for multi-device environments. Issues and solutions for automatic transformation from desktop interfaces to different platforms will be first discussed, showing how presentation, navigation and content can be transformed and the usability issues to address in this process. The tutorial provides examples of results that can be obtained by tools provided by main software companies such as Google, Nokia, Microsoft, Opera along with research results from various groups (including the instructor's group).

Then, the tutorial moves on to discuss how mobile users can be supported in multi-device environments. To this end, distributed and migratory interfaces are introduced. Migratory interfaces are interfaces that can transfer among different devices, and thus allow the users to continue their tasks. This definition highlights important concepts: task performance continuity, device adaptation and interface usability. The diversity in features of the devices involved in migration, such as different screen size, interaction capabilities, processing and power supply, can make a user interface developed for a desktop unsuitable for a PDA and vice versa. For example, an interface layout designed for a desktop platform does not fit in the smaller screen of a PDA, or a graphic interface running on a desktop system must be transformed to a voice interface when the application migrates to a car. Thus, an interface cannot migrate as is from one device to another (except in case of homogenous devices), and needs intelligent engines in order to adapt it to the different features of the target platform taking into account usability principles. Task performance continuity means that when migration occurs users do not have to restart the application on the new device, but they can continue their task from the same point where they left off, without having to re-enter the same data and go through the same long series of interactions to get to the presentation they were accessing on the previous device. Lastly, a research agenda for the field is introduced and discussed.

References

1. Myers, B., Hudson, S., Pausch, R.: Past, Present, Future of User Interface Tools. *Transactions on Computer-Human Interaction*, ACM 7(1), 3–28 (2000)
2. Mori, G., Paternò, F., Santoro, C.: CTTE: Support for Developing and Analysing Task Models for Interactive System Design. In: *IEEE Transactions on Software Engineering*, vol. 28(8), pp. 797–813. IEEE Press, Los Alamitos (2002)