# Towards an Evolution Model for Supporting Plasticity of User Interfaces

*Alexandre Demeure, Gaëlle Calvary, Nicolas Barralon*

CLIPS-IMAG
BP 53, 38041 Grenoble Cedex 9, France
{Alexandre.Demeure, Gaelle.Calvary, Nicolas Barralon}@imag.fr

## Abstract

The increasing proliferation of computational devices has introduced the need for applications to run on multiple platforms in different physical environments. Providing a user interface specially crafted for each context of use is extremely costly and may result in inconsistent behavior. As a result, user interfaces must now be capable of adapting to multiple sources of variation. Plasticity refers to the ability of a user interface to mould itself to a range of contexts of use, where a context of use is defined as a triple <user, platform, environment>. This poster focuses on the *Evolution Model* that specifies the reaction to perform in case of changes of context of use. It first presents the problem space of the reaction then proposes a model for specifying the adaptation.

## 1 Introduction

Mobility coupled with the development of a wide variety of access devices has engendered new requirements for HCI such as the ability of user interfaces to adapt to different contexts of use. By context of use, we mean the triple <user, platform, environment>. The platform denotes the set of interaction and computational resources that are available for supporting the current activities. The environment corresponds to the physical space where the interaction takes place. By analogy with materials that expand and contract under natural constraints without breaking, plasticity refers to the ability of a user interface to withstand variations of context of use while preserving usability (Thevenin, 2001). This poster deals with plasticity. It focuses on the Evolution Model that specifies the reaction to perform at run-time when the context of use changes. For instance, because a PC is turned on, migrate the drawing area of a paint editor from the PDA to the PC. In this *If condition then reaction* rule (Calvary, 2001), section 2 focuses on the *reaction* part. It presents the problem space of the reaction. Section 3 proposes a model for the specification of the *condition* and *reaction* parts.

## 2 The problem space of the reaction

From a user perspective, three kinds of reactions may be distinguished:

- An action on the context of use: for instance, asking the user for speaking English, changing the screen resolution or switching the light on;
- A recasting of the user interface (UI): for instance, replacing a set of radio buttons with a combo box or splitting a panel into two windows;

- A redistribution of the UI across the available platforms: for instance, migrating the drawing area from a PDA to a PC (partial migration) or fully migrating the drawing editor from the PDA to the PC.

From a system perspective, these reactions may require:

- A function call: for instance, for changing the screen resolution or adapting the UI;
- A change of software components: for instance, switching from a UI to another one. In this case, the UI component is not preserved. It is replaced with another one;
- A change of executable code, i.e. switching from an interactive system to another one.

Any reaction may be performed by the user and/or the system. If the user is involved in the evolution then a UI is required for the evolution model itself.

# 3 A model

The evolution model is modelled as a set of rules *If condition Then Proposing (reaction, attributes)* where:

- The *condition* expresses what in the context of use has changed and who of the user and/or the system has triggered the variation;
- The *reaction* evolves in the problem space presented in section 2. It may consist in an action on the context of use, a recasting or a redistribution of the UI. A description of the reaction is provided (both in natural language and conceptual graphs). The actor in charge of the execution (the user and/or the system) is specified. The *reaction* is prescribed with a set of attributes;
- The *attributes* specify whether the reaction is a luxury or a necessity, to which extent the rule should be applied (undesirable to mandatory), and who of the designer and end-user has specified the rule.

Explicit links to the interactive system may be necessary for the evaluation of the condition and the execution of the reaction. Typically, for sensing the context of use and specifying the function to be called. The links identify the appropriate functions in the interactive system.

## Conclusion and perspectives

This poster has briefly focused on the evolution model that had been so far ignored in the literature. With ubiquitous computing this model is helpful for specifying reactions to variations of context of use. Based on conceptual graphs, we are now modeling both the descriptions of reactions and the capitalization of pre-computed software components. These components may be stored at different levels of abstraction ranging from a task-based description to executable UIs.

## References

Calvary, G., Coutaz, J., Thevenin, D. Supporting Context Changes for Plastic User Interfaces : a Process and a Mechanism, in “People and Computers XV – Interaction without Frontiers”, Joint Proceedings of AFIHM-BCS Conference on Human-Computer Interaction IHM-HCI’2001 (Lille, 10-14 September 2001), A. Blandford, J. Vanderdonckt, and Ph. Gray (eds.), Vol. I, Springer-Verlag, London, 2001, pp. 349-363

Thevenin, D. Adaptation en Interaction Homme-Machine: le cas de la Plasticité, PhD thesis, Grenoble, 2001, pp 234