

ARTStudio; tool for multi-target UI design

David Thevenin

National Institute of Informatics
Hitotsubashi 2-1-2-1913 Chiyoda-ku, Tokyo,
101-8430, Japon
thevenin@nii.ac.jp

ABSTRACT

In this poster, I present ARTStudio, a tool for multi-target UI design based on conceptual framework that helps structuring the development process. It proposes a set of visual editors for describing specifications and modifying generated models. It generates UI for Palm and Work-Station.

KEYWORDS: Design tool, adaptation, Multi-target design, plasticity, Model-based.

INTRODUCTION

The field of user interface technology is expanding rapidly. With the advent of digital cell phones, personal organizers, and wall-size augmented surfaces, users expect to interact with the same interactive system both in the small and in the large using multiple modalities. In addition, wireless connectivity offers new opportunities for using interactive systems in different environments. The result is the apparition of multi-target User Interface; an UI which can be executed in different environments, by different users, on different platforms. Meanwhile, current software techniques primarily address the development of mono-target user interfaces. As a consequence, multi-target user interfaces are built in an ad-hoc way, resulting in high development and maintenance costs.

GOAL

To propose software for multi-target UI design, I was naturally interested by adaptation mechanisms. There are different ways to address this problem, such as context toolkit, abstract languages (like UIML [1]), ... But an important limitation of these implementational approaches is the non-integration of information (meta-information) coming from high-level specifications. However, this information, extracted from models of tasks or data for example, will drive the adaptation of UI by changing the navigation, splitting or bringing together windows, etc.[5]. For this, we have to propose tools covering all design levels for encapsulating the good information for adaptation.

So, for covering all design levels, I reconsider the Model-based (MB) approach, which rely on high-level specifica-

tions, and provide the foundations for design tools and code generation. As a result, they alleviate the cost of code production while improving code quality.

But these technologies have relatively failed in the past. The main reasons are the difficulty of specifying, the lack of tools and languages flexibility, and the failure of only automatic generation. Therefore, my goal is to propose a tool implementing a reference framework based on MB approach, and which proposes graphic tools for visual specification editing. It must help the designer and not impose a solution. So ARTStudio is a semi-automatic multi-target UI generator.

FRAMEWORK

As shown in figure 1, The Multi-target Framework recommends developers and designers to specify the following classes of description:

- The Concepts Model captures the domain concepts and their relations. Domain concepts denote the entities that users manipulate in their tasks.
- The Task Model describes how users reach their goals with the system.
- The Platform Model, the Environment Model and the User Model, define the context of use, that is, the set of targets (platform, environment, user) that the user interface is intended to cover. For the moment I only use a platform model. The Interactors Model, which describes the interactors available to render domain concepts and functions, denotes a particular class of software resources of the target platforms.

These descriptions are referenced along the development process. As shown in figure 1, the process is a combination of vertical reification and horizontal translation. A vertical reification is applied for a particular target while translation is used to express bridges between the descriptions for different targets. The framework is completely described in [2,5].

THE TOOL

In its current implementation, ARTStudio (ART for Adaptation through Reification and Translation) supports the four-step reification process of the Multi-target Reference Framework, as well as human intervention. Multi-targeting is limited to Java-enabled single screen elementary platforms (JFC, Waba). ARTStudio is implemented in Java and uses CLIPS, a rule-based language, for generating Abstract and Concrete UI's. All of the models used and produced by ARTStudio are saved as XML files.

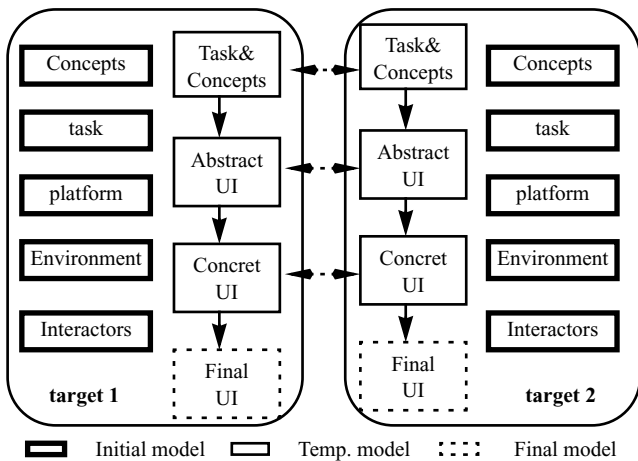


Figure 1: Multitarget reference framework for the development of multitarget user interfaces.

As shown in figure 2, the development of a multi-platform UI forms a project. That includes the Concepts-and-Tasks model, the Concrete UI and the Final UI. When clicking on the "Contexte" thumbnail of figure 2, the developer has access to the platform and interactors models. The "règles" thumbnail gives access to the generation rules used by ARTStudio during the reification process. These rules, which address presentation issues, can be adapted by the designer to the case at hand. In the current implementation, rules are not editable.

Domain concepts are modelled as UML objects using ArgoUML. In addition to the standard UML specification, a concept description includes the specification by extension of its domain of values. The type and the domain of values of a concept are useful information for identifying the candidate interactors in the Concrete UI.

The figure 2 shows the different models editors in ARStudio:

- The Task editor uses CCTE [3] as a formalism for describing the tasks and the links with the functional core. Functional core's objects are described with UML in the concepts model. ARTStudio cannot edit this model but only offers a view on all accessible concepts.
- The Abstract UI editor is only a view on the model. The designer cannot yet modify it.
- At last, the Concrete UI editor is a kind of graphic presentation editor, that offers to the designer the ability to modify the generated UI. This figure shows the editor for Palm and for Java/Swing based UI.

CONCLUSION AND FUTURE WORK

I present here a tool for multi-target UI generation. This tool is based on a framework, which structures the design process. It offers a set of graphic tools for specifying the different levels of design.

Actually, ARTStudio implements a sub-part of the general framework and it supports only a generation for PalmOS/Waba and for JFC. In addition, the rules for generating

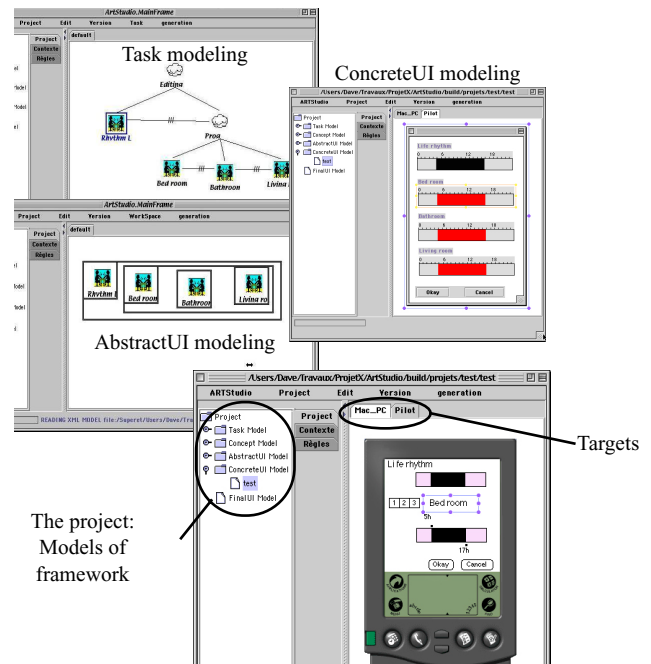


Figure 2: ARTStudio: Tool for design multitarget UI. A multi-platform UI is developed within a project. The frame of the left end-side gives access to the sets of models. Pictures show ARTStudio modelers for task, Abstract UI and Concrete UI.

Abstract and Concrete user interfaces are simple. Thus in the future, I plan to work on these points. In particular, works were done about how to describe the dialog from the structure of tasks and data.

REFERENCES

1. Abrams, M. & Phanariou, C. *UIML: An XML Language for Building Device-Independent User Interfaces*. XML '99, Dec. 1999, Philadelphia. <http://www.xml.com/pub/a/1999/12/xml99/wrapup.html>.
2. Calvary, G., Coutaz, J. & Thevenin, D. *A Unifying Reference Framework for the Development of Plastic User Interfaces*. Proc. of the 2001 Engineering of Human-Computer Interaction Conference (EHCI'2001), Lecture Notes in Computer Science, Springer-Verlag.
3. Paternò, F., Mancini, C. & Meniconi, S., *ConcurTask-Trees: a diagrammatic notation for specifying task models*. Proc. of Human-Computer Interaction 1997 (INTERACT'97), Chapman&Hall, pp. 362-369.
4. Szekely, P. *Retrospective and challenges for Model-Based Interface Development*. Proc. of Computer-Aided Design of User Interfaces (CADUI'96), 1996, pp. xxi-xliv.
5. Thevenin, D. *La plasticité des Interfaces Homme Machine : une approche, un outil*. Computer Science Ph.D Thesis, Joseph Fourier University - Grenoble I, 2001, 212 p.