

Design of Adaptive User Interfaces for Electronic Patient Records

Angel R. Puerta

Section on Medical Informatics

Stanford University

251 Campus Drive -MSOB x215

Stanford, CA 94305-5479 USA

+1 650 723 5294

puerta@smi.stanford.edu

<http://www.smi.stanford.edu/projects/mecano>

INTRODUCTION

One of the important challenges in designing user interfaces for electronic patient records is to account for adaptation. These interfaces must typically adapt to the needs of different types of physicians, specialists, and patient situations. At the Section on Medical Informatics at Stanford University, my group is investigating methodologies for the effective, user-centered design of adaptive user interfaces for electronic patient records. Our approach is based on a technology called model-based interface development [1] that we have been developing for a number of years.

The model-based approach to user-interface design includes the following premises:

- All relevant aspects of a user-interface design are captured in a declarative interface model. This model includes abstract knowledge (e.g., user tasks, domain, and user type models) and concrete knowledge (such as presentation and dialog characteristics).
- A model-based interface development environment supports the design of an interface from the definition of user tasks to the layout of widgets and specification of behavior.
- The knowledge needed for runtime adaptation in the user interface is captured at design time by the interface model. Thus, there is no need for additional runtime processes to support adaptation.

We have built a model-based interface development environment, called MOBI-D (Model-Based Interface Designer) [1], that supports our model-based paradigm. We are using this environment to experiment with adaptation issues in electronic patient record interfaces.

THE MOBI-D INTERFACE DEVELOPMENT ENVIRONMENT

The Model-Based Interface Designer environment [1] supports end users and interface developers in designing

and implementing user interfaces under a user-centered development cycle. MOBI-D integrates a number of design-time and runtime tools, including:

- A user-task elicitation tool to obtain user-task models directly from domain experts.
- A set of interactive model editors. Each category of interface elements (user task, domain, presentation, dialog, and user type) is handled via a model editor with specific functionality pertinent to that category.
- A task-interface model mapping tool that acts as an interface design assistant. It allows developers to make global and specific design choices for presentation and dialog.
- A task-based interface builder. Similar to the familiar palette-and-canvas builders but where operations are dictated by a user-task model.

USING MOBI-D FOR ELECTRONIC PATIENT RECORD INTERFACES

We are utilizing MOBI-D in the electronic patient record area for the following:

- Elicitation of user-task models for various physician types (e.g., internists, oncologists, and infectious diseases)
- Building domain and user type models for electronic patient records.
- Construction of interface models for user interface prototypes that adapt to physician types and patient characteristics (as dictated by the user-task, user type, and domain models).

ACKNOWLEDGEMENTS

The work on MOBI-D is supported by DARPA under contract N66001-96-C-8525.

REFERENCES

1. Puerta, A. R. A Model-Based Interface Development Environment. *IEEE Software*, (14) 4, July/August 1997, pp. 40-47.