MDDAUI 2010 Workshop Report

Jan Van den Bergh
Hasselt University - tUL - IBBT
Expertise Centre for Digital Media
Jan.VandenBergh@uhasselt.be

Gerrit Meixner
DFKI
Gerrit.Meixner@dfki.de

Stefan Sauer
University of Paderborn
s-lab – Software Quality Lab
sauer@s-lab.upb.de

ABSTRACT
The workshop on Model-Driven Development of Advanced User Interfaces is a forum of multidisciplinary discussion on how to integrate model-driven development with the more informal methods of user-centered design and development of user interfaces. Starting point of the discussion were the tools, models, methods and experiences of the workshop participants. This report presents the overall aims of the workshop and presents the results of the discussion groups.

Author Keywords
Model-driven development, user-centered design, models, workshop report.

ACM Classification Keywords
D.2.2. Design Tools and Techniques (User Interfaces), H5.2. User Interfaces (User-centered design).

INTRODUCTION
The workshop on Model-Driven Development of Advanced User Interfaces (MDDAUI) was the fifth edition of this workshop series, organized for the first time together with the CHI conference. Previous editions were organized at MODELS and IUI conferences. More information on the background and motivation for this workshop can be found in the CHI 2010 Extended Abstracts [5].

MDDAUI 2010 focused on challenges, opportunities, practical problems, and proposed solutions to increase the usability and user experience of user interfaces created with a model-driven development approach. A highly interactive format was used to foster discussion between participants. All participants, except the organizers, were selected based on papers, which were reviewed by the program committee.

After a short introduction, the thirteen accepted papers were presented in three blocks. Each block consisted of four or five seven-minute presentations followed by a short discussion to identify potential points for more elaborate discussions. The resulting set of potential discussion points were grouped in a collaborative effort by all participants into three discussion topics, which were later discussed in three separate discussion groups. Thus, the identified topics reflect important issues in this field from the viewpoints of the workshop participants.

Proceedings of the 5th International Workshop on Model Driven Development of Advanced User Interfaces (MDDAUI 2010): Bridging between User Experience and UI Engineering, organized at the 28th ACM Conference on Human Factors in Computing Systems (CHI 2010), Atlanta, Georgia, USA, April 10, 2010.

Discussion Topic	Discussion Points
Integrate knowledge from other fields	• HCI patterns • Cognitive models • HCI and usability guidelines and standards on knowledge representation • User experience and usability
Multi-device and multi-modal interaction generation	• New common reference framework? • Different abstractions? • Cost versus usability • Dynamic distribution
Development processes	• End-user development • Customization • Iteration and prototyping • ISO 13407 [3] • Standards about development processes

Table 1 Discussion topics and discussion points

Table 1 shows an overview of the discussion topics and the individual discussion points that were identified. Some of the points evolved as a result of merging several initial discussion points raised during the presentation blocks.

In the remainder of this workshop report, the results of the different discussion groups are summarized, followed by a discussion on how we created and presented the workshop

poster, which was also created in a collaborative effort of several workshop participants.

GROUP DISCUSSIONS

The group discussions formed a major part of the workshop. They started before lunch and ended just before the end of the workshop, leaving enough room for the plenary presentation of the results of the group discussions and a few short closing remarks.

Integrate knowledge from other fields

Out of practical considerations the discussion in this discussion group focused on the integration of knowledge from cognitive sciences to complement the knowledge already available from the engineering disciplines. The central goal of integrating this knowledge is to improve usability of user interfaces that are generated by a model-driven development approach.

Starting point of the discussion were the models and abstraction levels of the (revised) reference framework for plastic user interfaces [2] (Cameleon reference framework). Different contributions to these models from the cognitive sciences community were proposed for specific abstraction levels. The saliency model could e.g. be used to test final user interfaces (as presented by Jeremiah Still in the workshop) and provide feedback to the concrete user interface model. It was however indicated that a significant amount of research is still necessary to establish this capability. Cognitive workload models were identified as a potential candidate to enable transition from concrete user interfaces to abstract user interfaces. The participants of the discussion group believed that the latter could also benefit from an inclusion of the (relative) importance of its components in the model. Finally, the concepts and tasks layer of the Cameleon reference framework could include knowledge from the GOMS model (Goals, Operators Methods Selection rules) [1] for task analysis. Support for indicating task frequency was also considered important.

Besides the models, HCI patterns, guidelines and standards were identified as important containers of knowledge to support transitions between the different abstraction levels. Patterns (also discussed in the presentation of Andreas Wolff) could assist in transitioning between all abstraction levels, while standards (including guidelines) were mostly considered beneficial in the transition between final and concrete user interfaces.

HCI patterns promise to solve the shortcomings of standards and guidelines by representing the design knowledge in a machine-readable and reusable form. By specifying “when, how and why” they enable automatic processing of the represented design information and are therefore suitable for the model-based generation of user interfaces. For using HCI patterns in a model-based development process several problems should be addressed in the future e.g. lack of formalization, lack of organization and the lack of effective tool support.

Multi-device and multi-modal interaction generation

This discussion group focused on how to create good user experience for user interfaces that can (semi-) automatically adapt to multiple devices and support multi-modal interaction at runtime. Consistency was identified to be an important factor for good user experience (or rather usability, as argued by some participants?). Consistency has different aspects that need to be balanced: consistency within the user interface of a single application on a single device, consistency with user interfaces of other applications on the same device and consistency between the user interfaces of the same application on different devices. Depending on the kind of applications and the desired user experience, different kinds of consistency may be more important.

Another discussion point was centered upon the question: "How to ensure consistency?". Different approaches were discussed which would be suitable for different kinds of consistency. Guidelines were considered important, especially to ensure consistency between user interfaces of different applications on the same device, but they could also support consistency within an application's user interface. The usage of one or more common models (e.g. at the concepts and task level in the Cameleon reference framework [2]) for the user interfaces of an application on multiple platforms was raised as another potential means to ensure consistency. Device-specific models were, however, considered necessary to generate usable user interfaces.

In order to benefit from common models for ensuring consistency, the existence of both mappings and automated transformations was considered important. Mappings are important for logging and explaining links between the different models, while transformations support automation in the generation of models (and mappings). A last thing that was prevalent during the discussion was the importance of explaining why certain guidelines and/or transformations are present. Guidelines and transformations should become transparent to other stakeholders and be accomplished, e.g., by descriptions of the rationale.

Development processes

A large part of the discussion in the third discussion group was centered on the role of models in software engineering and user-centered design processes. First, a set of models and sub-models was gathered (see Table 2). The discussion then moved to different properties of these models within the development process. Three central questions were examined: "How are models used?", "At which level of abstraction are they?" and “How are the models related?”

The former question led to the distinction between design time models and runtime models. For design time models, tool support (user interface and languages) was considered a major area for future work, while models at runtime could be used for simulation (interpreting and executing) and analysis, and could play a major role in reverse engineering of UI models from interactive systems. The discussion

group members agreed: "Design-time models can ideally be used or instrumented as runtime models (or they should at least be related)."

Model	Sub-models
User model	• Preference, knowledge • Perception, cognition, (motor) action • Behavior, learning
User activity model	• Task • Scenario • Role • Workflow (procedures, cooperation)
Collaboration model	• Social behavior
User interface model	• Guidelines (e.g. ergonomic rules) • Dialogue model • Presentation model (input, output) • Platform model (CPU, description of device) • Interaction model (interaction techniques, widgets, gadgets, UI components, modalities)

Table 2 Models and sub-models

The discussion about abstraction levels started off with the observation that there are different views of abstractions. Among them, the Cameleon reference framework [2] and the Model-Driven Architecture [4] (MDA) were considered the most relevant. A mapping between their respective levels of abstraction was documented, while remarking the different notions of platform: Platform for the Cameleon reference framework refers to "a class of devices with similar interaction resources"; for MDA it refers to a certain combination of software and hardware. When reviewing the models, it was observed that most of the models did not fit completely in neither the Cameleon reference framework nor MDA; some extensions are required to either of them. The user activity model only fitted the concepts and tasks level of the Cameleon reference framework.

Finally, end-user development was also discussed. First, domain experts were considered as end-users of models, further distinguishing models for them from (user) models of them. Models for end-users were considered to support the active participation of end-users in the development of the user interface. The activities for which models are suitable were listed as design, program and customize.

In order to establish these goals, several requirements were listed: The models should be understandable, have the "right" level of abstraction and show the correspondence of concepts. Furthermore, tool support on different levels as well as domain-specific languages are also required for end-user development.

The last discussion group also touched the topic of guidelines and standards in user interface development. To ensure consistency and a high degree of usability across several user interfaces, standards and guidelines should be used during the user interface development process. For automatically considering standards (e.g. ISO 9241, VDI/VDE 3850) during the user interface generation process, e.g. for automatic verification in the concrete user interface layer, standards and guidelines have to be available in a formal notation. One possible solution could be the integration of the knowledge of standards and guidelines in knowledge bases. To reach this goal, much work still has to be done.

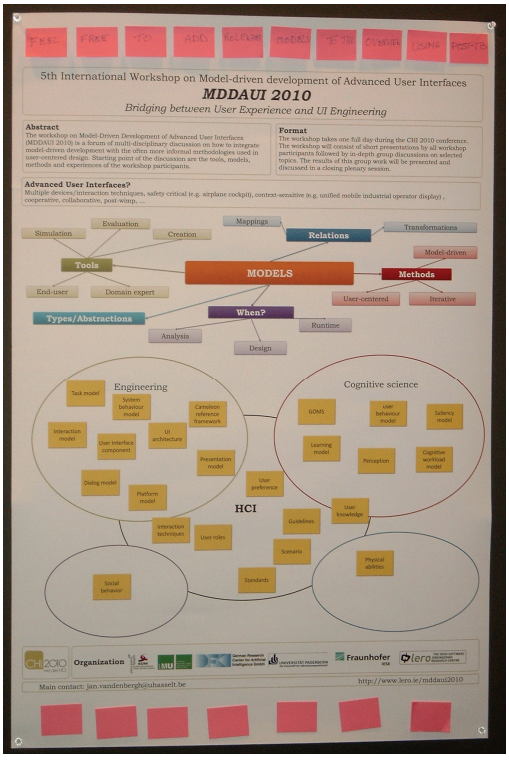

Figure 1 The "interactive" workshop poster

WORKSHOP POSTER

After the workshop, a poster was prepared in cooperation with several workshop participants. It was designed as an "interactive poster" inviting the viewers of the poster at the CHI conference to add artifacts. The poster consists of three major parts, as can be seen in Figure 1. The top part documents the goals and the format of the workshop, the middle part consists of a mind map that illustrates the topics

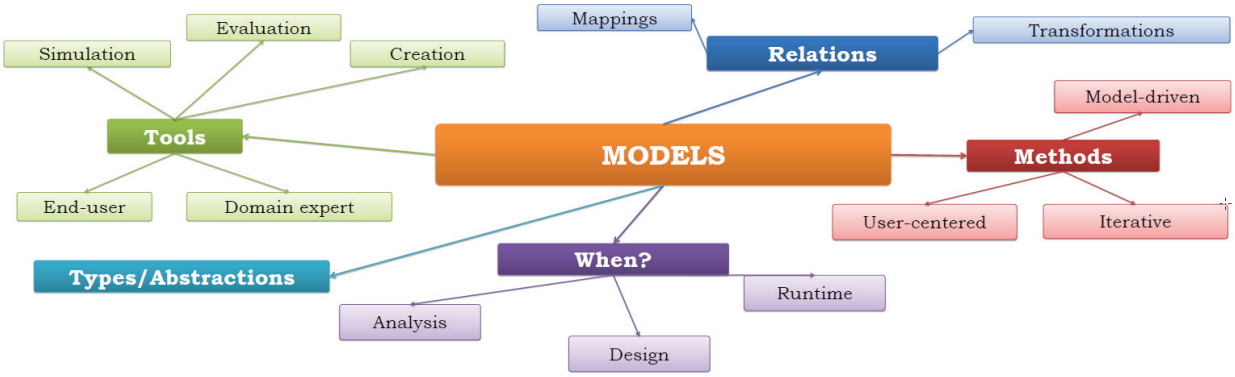

Figure 2 Mind map of the topics discussed during the workshop

that were discussed in the workshop, and the bottom part gives an overview of the different models and information artifacts that were considered from the different disciplines in the creation of user interfaces. The models were represented in a way similar to yellow post-it notes, while pink post-its invited viewers of the poster to contribute other models and information artifacts. In this way some more information artifacts were collected during CHI 2010.

Figure 2 gives a detailed view of the elaborated mind map that structures the area of model-driven user interface development. The main aspects identified around the central concept "models" are relationships (transformations and mappings), the main characteristics of the development methods (model-driven, iterative, or user-centered), the software development phase when models are applied (analysis, design, or runtime), the different scopes of tools (creation, evaluation, simulation, end-user, or domain expert), and the models' types and abstraction levels. The latter were elaborated further in the bottom part of the poster, based on the models in Table 2, but also including a set of other models as discussed in the previous section.

DISCUSSION

The fifth edition of MDDAUI, organized the first time at the CHI conference, shifts the workshop further towards a better integration with user interface design and HCI. According to the conference guidelines, the number of participants was restricted and emphasis was put on the discussions. Thus, paper presentations were kept really short, giving us much time for comprehensive and fruitful discussions, including both highly respected and well established experts and novices to the field and/or the scientific community.

The workshop also highlighted a number of challenges for the community and the workshop organizers:

- Development of a reference framework that better captures the needs of model-driven development of user interfaces providing good user experience.
- Incorporation of often relatively informal knowledge (such as HCI patterns, guidelines and standards) from non-engineering fields into models, mappings and transformations to support better user experience.
- Convince domain experts ─ both horizontal, (such as designers, information architects and usability experts) and vertical (such as health, finance and transportation) ─ of the benefits of participating in the MDDAUI community.

ACKNOWLEDGMENTS

We thank the CHI organizers for providing the opportunity to organize MDDAUI 2010 together with CHI 2010 as well as the student volunteers and other people from the CHI organization and program committees. Special thanks also go to the institutes and organizations supporting the participants and organizers to attend and organize the workshop. Last but not least, many thanks to all MDDAUI 2010 participants who helped to make this workshop a success.

REFERENCES

1. S.K. Card, T. P. Moran, and A. Newell. The psychology of human-computer interaction. Hillsdale, NJ: Lawrence Erlbaum Associates, 1983.
2. G. Calvary, J. Coutaz, D. Thevenin, Q. Limbourg, N. Souchon, L. Bouillon, and J. Vanderdonckt. Plasticity of user interfaces: a revised reference framework. In: First International Workshop on Task Models and Diagrams for User Interface Design TAMODIA2002, pages 127–134, July 18–19, 2002.
3. International Standards Organization. ISO 13407 – Human-centred design processes for interactive systems. Geneva, Switzerland 1999.
4. J. Miller and J. Mukerji. MDA guide version 1.0.1. http://www.omg.org/docs/omg/03-06-01.pdf, 2003.
5. J. Van den Bergh, G. Meixner, K. Breiner, A. Pleuss, S. Sauer, and H. Hussmann. Model-driven development of advanced user interfaces. In: CHI 2010 Extended Abstracts, pages 4429–4432, 2010.