Model-based Useware Engineering

Gerrit Meixner
German Research Center for
Artificial Intelligence (DFKI)
Trippstadter Str. 122
D-67663 Kaiserslautern
+49 631 205-3707
Gerrit.Meixner@dfki.de

Statement of Interest

Considering the interaction with technical devices such as a computer or machine control panel, the users' perspective renders the classic division of these devices from components into hardware and software obsolete: the users actually interact with a subset and intersection of these hardware and software components that, in their entirety, make up the user interface. From this perspective, however, we need to question the common practice of letting hardware and software specialists develop user interfaces, because this way, we disregard the probably most important component of an interactive system—the user!

Moreover, in a highly competitive market bringing forth technically and functionally more and more equal devices, user-friendliness as an additional sales argument secures a competitive advantage [1]. To put stronger emphasis on the users' and customers' needs, wishes, working styles, requirements, and preferences, and to consider them right from the beginning in all phases of the device development process, the responsible professional organizations in Germany, i.e. GfA, GI, VDE-ITG, and VDI/VDE-GMA, have coined the term "Useware" for the before-mentioned subset and intersection of hardware and software. They have also defined a systematic Useware Engineering Process [2] which demands for a comprehensive user, task, and use context analysis preceding the actual development. In the further course of the Useware Engineering Process, interdisciplinary teams of, for example, computer scientists, mechanical engineers, psychologists, and designers, continue developing the respective device in close collaboration with the ordering customer and her clients by constantly providing prototypes even in the very early development phases, thereby facilitating a continuous, parallel evaluation.

Furthermore in the future, a broader range of paradigms will emerge, allowing for multi-modal interaction incorporating, for example, visual, acoustic and haptic input and output in parallel. But also the growing number of heterogeneous platforms and devices utilized complementarily—such as PC's, smartphones, PDA's etc.—demands for the development of congeneric user interfaces for a plethora of target platforms. The perseverative development effort for every single platform, single device or even single use context solution is way too high, so that a model-based approach for the (abstract) development of user interfaces (MBUID) appears to be favourable

To support the Useware Engineering Process and its phase's different XML-based languages were developed:

- Useware Data Description Language (useDDL) is a language for defining generic requirement catalogues and storing information from the task and requirements analysis as an instance of these different categories.
- Useware Markup Language (useML) is a notation for specifying enhanced task models in industrial environments. Originally developed in 2003 [3], useML was enhanced in 2009 with several aspects concerning temporal operators, conditions and optionality of tasks [4].

Furthermore other XML-based languages were integrated in the Useware Engineering Process:

- Dialogue and Interface Specification Language (DISL) has been developed at the University of Paderborn, Germany as a modeling language for platform- and modality-independent user interfaces for mobile devices [5] with the focus on scalability, reactivity, easy usability for developers, and low demands on processing power and memory consumption.
- User Interface Markup Language (UIML) is a declarative markup language for the description of user interfaces [6]. UIML separates presentation components (e.g. widgets and layout), dynamic behavior (e.g. state transitions), and the content of a user interface.

Tools like TAMaRA (Tasks, Activities, Models and Requirements Analyzer) or Udit (useML-editor), different renderer for DISL and UIML as well as transformation processes were developed or are currently being developed to support editing, transformation and rendering of the models. Figure 1 visualizes the relations of the Useware Engineering Process, the different languages and the different tools.

Furthermore our model-based tool chain is structured according to the different abstraction layers of the CAMELEON reference framework [7]:

- "Tasks & Concepts"-Layer is mapped onto the analysis and structuring phase. UseDDL is used for storing information from the task and requirements analysis and useML 2.0 is used for task modeling.
- "Abstract User Interface"-Layer is mapped onto a part of the design phase. DISL is used for modeling abstract modality independent user interfaces.
- "Concrete User Interface"-Layer is mapped onto the other part of the design phase. UIML with a generic (graphical) vocabulary [8] is used for modeling concrete graphical user interfaces.

- "Final User Interface"-Layer is mapped onto the realization phase. Programming languages like Java or C++ are used for modeling the final user interface.

Different (semi-)automatic transformation processes are used for transforming models and languages.

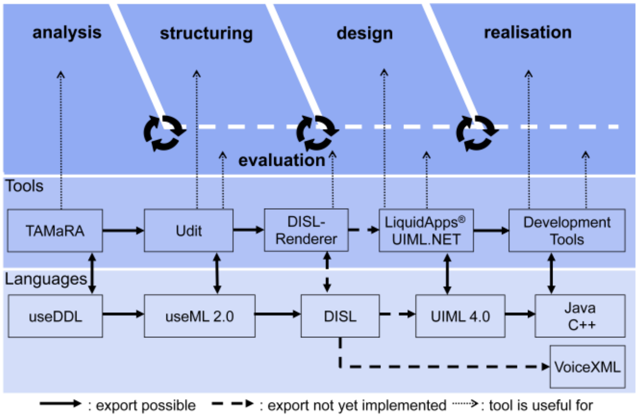

Figure 1: Supporting the Useware Engineering Process with a model-based tool chain.

As a member of the Incubator group "Model-based User Interfaces" I am especially interested in the different existing development approaches, models, notations and tools of the other participants. For several years now my main research interests is the linking of user-centered development processes (e.g. ISO 13407, Useware Engineering Process) with model-based development methodologies. Actually I am co-editor of the book "Model Driven Development of Advanced User Interfaces" by Springer and co-chair of the MDDAUI 2010 workshop at CHI 2010 in Atlanta, USA [9].

Actually we are conducting work in different areas of our tool-chain:

- Development of a transformation approach from DISL to UIML
- Development of a DISL-renderer for a Blackberry Bold 9000
- Extending UIML.NET to be compatible with the latest UIML 4.0 community draft
- Conducting research in the field of HCI patterns for integrating a (UIML) pattern language into our model-based development process [10]

1. REFERENCES

[1] Meixner, G.; Goerlich, D.; Boedcher, A.: Raising the Efficiency of the Use Context Analysis in Useware Engineering by Employing a Support Tool, Adjunct Proceedings of the 8th Asia-Pacific Conference on Computer-Human Interaction (APCHI) – Universal & Ubiquitous, Incheon, South Korea, 31-35, 2008.

[2] Zuehlke, D.; Thiels, N.: Useware engineering: a methodology for the development of user-friendly interfaces, in: Library Hi Tech, Vol. 26, No. 1, 2008.

[3] Meixner, G., Thiels, N.: Tool Support for Task Analysis, Workshop "User Interface Description Languages for Next Generation User Interfaces", 26th Annual CHI Conference on Human Factors in Computing Systems, Florence, Italy, 2008.

[4] Meixner, G.; Seissler, M.; Nahler, M.: Udit – A Graphical Editor For Task Models, Proc. of the 4th International Workshop on Model-Driven Development of Advanced User Interfaces (MDDAUI), Sanibel Island, USA, CEUR Workshop Proceedings Vol-439, 2009.

[5] Bleul, S.; Mueller, W.; Schaefer, R.: Multimodal Dialog Description for Mobile Devices. Workshop on Developing User Interfaces with XML: Advances on User Interface Description Languages, 2004.

[6] Organization for the Advancement of Structured Information Standards (OASIS). User interface markup language (UIML) version 4.0 committee draft. http://www.oasis-open.org/committees/download.php/28457/uiml-4.0-cd01.pdf, last accessed on 19 April 2010.

[7] Calvary, G.; Coutaz, J.; Thevenin, D. et al.: A Unifying Reference Framework for multi-target user interfaces. In: Interacting with Computers, 15, 3, 289-308, 2003.

[8] Plomp, C. J.; Mayora-Ibarra, O.: A Generic Widget Vocabulary for the Generation of Graphical and Speech-Driven User Interfaces. In: International Journal of Speech Technology, 5, 1, 39-48, 2002.

[9] 5th International Workshop on Model-Driven Development of Advanced User Interfaces, Atlanta, USA. http://www.lero.ie/mddaui2010/home.html, last accessed on 19 April 2010.

[10] 1st International Workshop on Pattern-Driven Engineering of Interactive Computing Systems, Berlin, Germany. http://www.zmmi.de/PEICS2010, last accessed on 19 April 2010.