

---

# Tool Support for Task Analysis

**Gerrit Meixner**

University of Kaiserslautern  
Gottlieb-Daimler-Str. 42  
67663 Kaiserslautern, Germany  
meixner@mv.uni-kl.de

**Nancy Thiels**

German Research Center for Artificial Intelligence (DFKI)  
Trippstadterstr. 122  
67663 Kaiserslautern, Germany  
nancy.thiels@dfki.de

**Abstract**

In this paper our user-centered model based user interface development process (MBUID) will be introduced with a clear focus on the analysis phase. Getting task information from users is a challenging problem, which is even more complex for user interface designer, because they have only little knowledge about task modeling. Well-known analysis methods will be combined by the support of an analysis tool, which helps entering, structuring, evaluating and exporting user information, e.g. tasks, preferences and structural mental models. All information is stored in a project database where developers can access information during every phase in the development process. Information is classified into different groups, where every group has different export mechanisms. Results from the analysis phase are used in further phases, thus there is no media break anymore. Future User interface description languages (UIDL) have then a clear starting point for developing user-centered user interfaces.

**Keywords**

User-centered design, user-interface design, MBUID, task models, information processing, useML

**ACM Classification Keywords**

H5.2. Information interfaces and presentation (e.g., HCI): User Interfaces

---

Copyright is held by the author/owner(s).

CHI 2007, April 28 – May 3, 2007, San Jose, USA

ACM 1-xxxxxxxxxxxxxxxxxx.

## Introduction

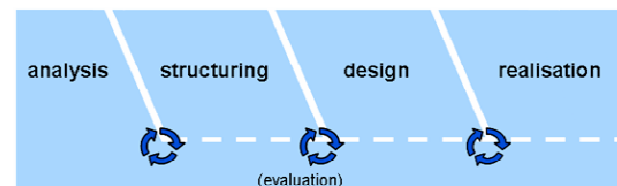
In recent years user interface (UI) developers have recognized that future users and their tasks are the most important factor for the success of a UI. For developing user-centered software new processes had to be developed. Normally a user-centered design process starts with the analysis of user tasks [1]. These tasks can be modeled as a task model with the notations of e.g. Useware Markup Language (useML) [2] or ConcurTaskTree (CTT) [3]. Task models are a very useful starting point for the development of further models in MBUID and help to guarantee a user-centered design [4]. Different approaches exist which generate the presentation model out of the task model [5]. A User Interface Description Languages (UIDL) is one method to cover the requirements of a presentation model. Today, most UIDL are based upon XML and many tools exist which help developers to support parts of MBUID, e.g. TERESA [6]. One of the main problems in developing task models still remains – how to get the knowledge of a domain expert into MBUID [1]. In this paper one possible software supported solution is shown.

This position paper is structured as follows: the following section gives a short introduction to the user-centered MBUID process with focus to the analysis of user tasks and requirements. Then the developed analysis tool will be illustrated. Finally some directions for further work are discussed.

## Useware Engineering Process

The level of acceptance and efficiency of a modern user interface are not at last determined by the ease of use. System development has been advanced by the Useware engineering process [8]. This process is made

up of the following phases: analysis, structural design, design, implementation and evaluation (see Fig. 1). The individual phases are not to be viewed in isolation from one another, but rather, as overlapping. The evaluation, as a continuation of the analysis, parallels the whole process [7].



**Figure. 1** Useware engineering process

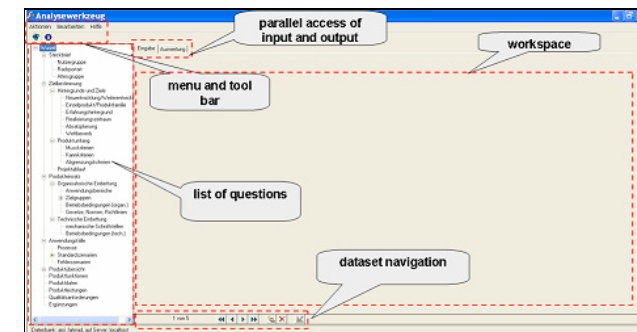
The primary considerations in this evolutionary process are always the requirements and needs of the user for whom the user interface is being developed. This is the only guarantee for an efficient use of the system. Thereby different methods can be used to gather the relevant data: First of all user interviews are best methods for this purpose, as they can be designed very flexible and a huge amount of different information can be obtained. In contrast to that, questionnaires are static and require much development time beforehand. Therefore their usage in this context is rare. To survey mental models, card sorting provides a valuable basis for the following structuring phase. After the data survey the evaluation takes place while using three different data categories: quantitative, qualitative and structural. The latter results from the mental models of questioned users. The other two are obtained by different questioned methods during the analysis: qualitative data by single user statements; quantitative data by univocal answers. Structural data and resulting

from that task models form the basis to derive a usage model within the next phase. This structuring phase is characterised by useML [2] which consists of use objects and elementary use objects, i.e. "change", "release", "select", "enter" and "inform". These objects describe actions, operations and activities which were obtained by the analysis phase. With these use objects and five elementary use objects it is possible to develop the whole structure for the final user interface. The result of the structural design phase is a platform-independent model which provides the foundation for the following design phase. Within this phase the use concept for the selected hardware platform is prepared on the basis of the conducted analysis and the use model developed. Furthermore, a global navigation concept, a draft of the related usage structure as well as a proposed layout is developed. The result of the design phase is a specific layout of the user interface system. The layout focuses on standard user tasks and the usage structure. Simultaneous evaluation during all of the formerly mentioned phases enables users to track and assess the development progress at all times on the basis of structures or prototypes. Therefore timely response to desired changes or modifications is possible. The evaluation includes user surveys to determine the validity of the results of structuring and designing.

### Analysis tool support

Through the multiplicity of the gathered information in the analysis phase (see chapter Useware Engineering Process), it is only possible to get an overview by the use of an electronic data storage. The developed analysis tool "useDATA" (see Fig. 2) presented serves thereby the creation of a central, development-specific database which guarantees above all the complete

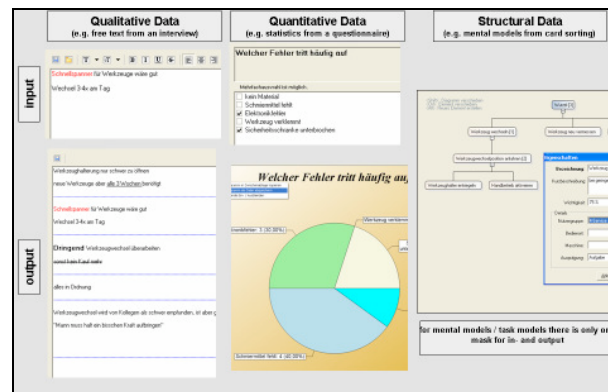
supply of all data resulted during the development process. The developer is able to use the tool for preparing, admission and passing data to further phases of the development process.



**Figure 2.** Screenshot of the analysis tool "useDATA"

The collection and evaluation of data can be handled parallel with the analysis tool. At any time it is possible to examine and concretize partial results (see Fig. 2). The collection and evaluation of data within "useDATA" takes place for each data category (see Fig. 3).

"useDATA" includes export mechanism for presentation and documentation data as well for task models (see Fig. 3). The developer has the possibility to export qualitative data, i.e. text (in Rich Text Format), quantitative data, i.e. diagrams (as Portable Network Graphics) and structural data, i.e. task models (useML). Qualitative and quantitative data can be used for presentation and documentation of acquired information during the analysis phase. Structural data in terms of task models can be enriched in further development phases.



**Figure 3.** Input and output of different data categories

### Conclusions and future work

In this paper the Useware engineering process with focus to the analysis phase was introduced, as well as the “useDATA” tool. “useDATA” was tested in several industrial projects and helped to develop task models. It has improved the structured analysis and reduced the amount of work for entering, structuring, evaluating and exporting data to further phases of the MBUID process. Especially future UIDL may benefit from a solid data base of user information. Thus “useDATA” bridges a part of the gap between research in MBUID and practical work.

At the moment a XML-based language for the storage of information is developed. The relational data base storage concept will be replaced by the Useware Data Description Language (useDDL). Developers can easily view, edit and exchange analysis data and don’t need to extract data out of the data base. Future versions of

“useDATA” should be multilingual and another feature could be the integration of an export mechanism of task models to CTT.

### References

- [1] Tam, R.C., Mulsby, D. and Puerta, A.R. *U-TEL: A Tool for Eliciting User Task Models from Domain Experts*. In *Proc. IUI 1998*, pp. 77-80.
- [2] Reuther, A. *useML – Systematische Entwicklung von Maschinenbediensystemen mit XML*. no. 8 in *Fortschritt-Berichte pak*, University of Kaiserslautern, 2003.
- [3] Paternò, F. *Model-based Design and Evaluation of Interactive Application*. Springer Verlag, 1999.
- [4] Johnson, P., Johnson, H. and Wilson, S. *Rapid Prototyping of User Interfaces driven by Task Models*. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, John Carroll (ed.), John Wiley & Sons. pp. 209-246, 1995.
- [5] Clerckx, T. and Coninx, K. *Integrating Task Models in Automatic User Interface Generation*. Technical Report TR-LUC-EDM-0302, EDM/LUC Diepenbeek, Belgium, 2003.
- [6] Mori, G., Paternò F. and Santoro, C. *Design and Development of Multidevice User Interfaces through Multiple Logical Descriptions*. In: *IEEE Press: IEEE Transactions on Software Engineering*. vol. 30, no. 8, 2004, pp. 507-520.
- [7] Bödcher, A. *Methodische Nutzungskontext-Analyse als Grundlage eines strukturierten USEWARE-Engineering-Prozesses*. no. 14 in *Fortschritt-Berichte pak*, University of Kaiserslautern, 2007.
- [8] Zühlke, D. *Model-based Development of User Interfaces: A New Paradigm in Useware Engineering*. In: *Proc. 10<sup>th</sup> IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human-Machine-Systems 2007*.