In: Johannsen, G. (Eds.): 8th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human Machine Systems 2001. Preprints , pp. 577-581, September 18-20, Kassel, VDI /VDE-GMA

FORMAL USABILITY EVALUATION OF INTERACTIVE SYSTEMS

Nico Hamacher[1], **Jörg Marrenbach**[2], **Karl-Friedrich Kraiss**[3]

[1]EMail: hamacher@techinfo.rwth-aachen.de
Tel.: +49-241-44026105
[2]EMail: marrenbach@techinfo.rwth-aachen.de
Tel.: +49-241-44026105
[3]EMail: kraiss@techinfo.rwth-aachen.de
Tel.: +49-241-44026101

Department of Technical Computer Science
Aachen University of Technology, Germany
Ahornstr. 55, D-52074 Aachen
Fax: +49-241-8888308

Abstract: This paper describes a new tool for the generation and analysis of normative user models based on the GOMS theory for the evaluation of interactive systems and the analysis of the usability. Furthermore, a comparison of user models and generated action protocols is facilitated. Additionally, design alternatives can be compared. The results of the analysis are visualised in various ways.
The introduced tool supports the system engineer in a considerable way to evaluate interactive systems and produces suitable analysis data as a base for decisions while the systems are developed.

Keywords: Evaluation, GOMS, usability, user model, interface design

1. INTRODUCTION

Today, usability engineering of interactive Systems is receiving increased attention. In general usability is determined by testing prototypes. The disadvantage of this approach is that these tests can be done only in late stages of development process. An early analysis of usability would be a significant advantage with regard to saving time and resources. In this paper the tool TREVIS (**T**ool for Rapid **Ev**aluation of **I**nteractive **S**ystems) is presented. TREVIS enables the design engineer to model the behaviour of an user while interacting with a device and derive usability measures from this simulation

2. DEVELOPMENT PROCESS

The development of systems can be divided into several phases. Figure 1 shows the V model as an example, which describes the development process including the different phases.
At the beginning, a requirement analysis collects the technical and environment requirements for the system. The system specification is the next phase, where the requirements are transferred into different formalisms as, e.g., Statecharts or SDL. Based on these formal specification the modules can be designed in a top down process. After the implementation these models can be integrated bottom up. Following the integrating phase the product development is terminated. Today, only in the integrating phases a system test can be performed, because only then a prototype is available for experimentation. Hence results from the analysis come to late to be included in the prediction process.

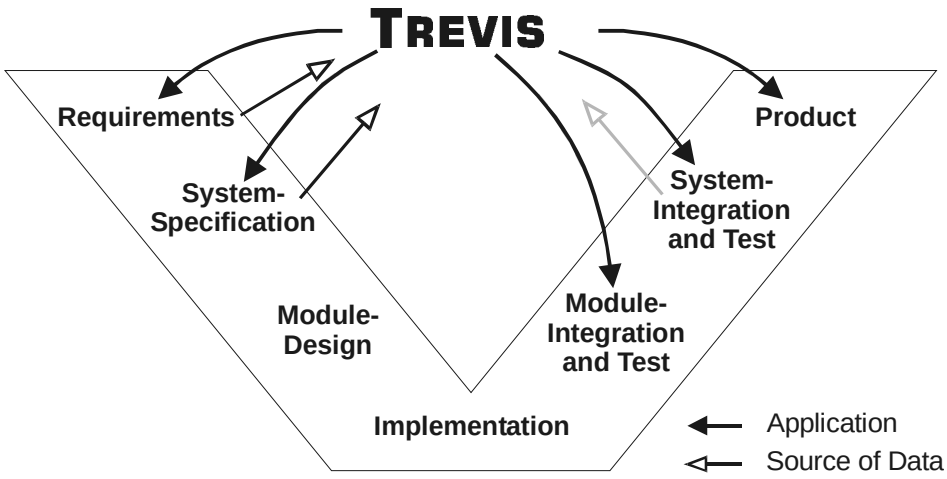

Fig. 1: V model describing the different phases of a development process.

3. USABILITY EVALUATION

One method of formal evaluation is developed by Card, Moran and Newell for modelling interactions between an user and an interactive system. It is called GOMS, which is an abbreviation for the components of the model: Goals, Operators, Methods and Selection rules. A GOMS model is also named user model. Over the years, some GOMS variants were introduced. The dialect with the most extensive analysis results is NGOMSL (Natural GOMS Language), first introduced by Kieras in 1988. An analysis based on NGOMSL generates qualitative as well as quantitative predictions, like execution and learning time. The execution time describes the time to reach the goal whereas the learning time specifies how much time an operator needs to learn the whole task.

Although the use of GOMS models is not very complicated, it is very tedious to build these models manually. Hence, a tool is needed, which integrates the GOMS theory in the development process and supports efficiently creation and analysis of user models.

4. SYSTEM ARCHITECTURE

For the evaluation of usability three criteria are considered essential. According to ISO 9241 part 11 these are *effectiveness*, *efficiency* and *satisfaction*. Effectiveness describes with which precision and speed a user can reach a goal or subgoal. Efficiency specifies the effectiveness in proportion to the required effort.

Mostly, a distinction between human efficiency, temporal efficiency and economic efficiency is made. Satisfaction is an indication of the acceptance of handling a device.

Regarding these criteria the tool TREVIS (**T**ool for **R**apid **Ev**aluation of **I**nteractive **S**ystems) is described that supports the synthesis and analysis of user models based on NGOMSL. As shown in figure 1, TREVIS is applicable in most of the stages of a development process.

TREVIS includes four main modules, depicted in figure 2:

- Based on the task sequences as one result of the requirements the user models can be created manually in the user model editor. The tool supports this process e.g. by offering a library for reusing components and a graphical editor. Moreover, user models can be considered in projects, where the project represents the interactive system and the user models describe the tasks which has to be done. In TREVIS a whole project management is also realised.
- The device model contains details about the inner works of the device. If a device model was created in an earlier stage of the development process, the task sequences can be generated semiautomatically using the device model converter. The user models can be created also from these sequences.
- In the analysis module different analyses are included, which depend on the development phase, where TREVIS will be used. These analyses are described in the next section.
- The user models contains the complete description of the procedural knowledge that the user has to know in order to perform tasks using the device. Hence, a handbook based on the user models can be created by the handbook generator. Therefore the hierarchical structure of the user models were used to build the sections, subsections and the topics. Furthermore, an index for the handbook is generated automatically, too.

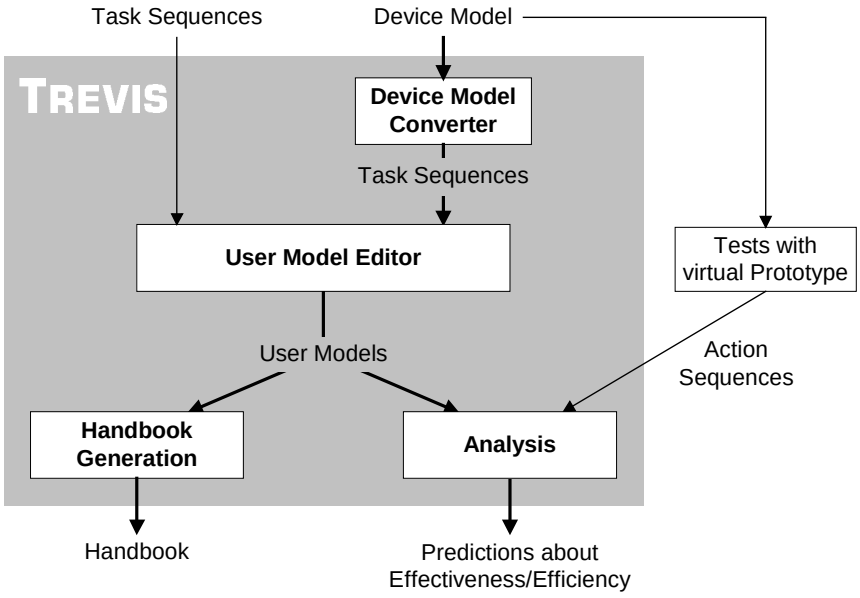

Fig. 2: Architecture of TREVIS

5. ANALYSIS RESULTS

The analysis module of TREVIS generates various Outcomes, which depends on the kind of analysis and the development phase, TREVIS will be used in. Four different analyses are realised:

- The user model analysis generates qualitative as well as quantitative predictions, like execution and learning time (as already described with NGOMSL).
- A comparison between different user models is implemented in the design analysis module, which can be used as a basis for design decisions. This comparison is realised by a suitable representation of the user model analysis of all included user models. Furthermore the used operators are diagrammed for each user model to compare the actions, which come into operation. Although this comparative presentation is a helpful functionality, no other tool includes it.
- In the action sequence analysis, action sequences resulting from testing a prototype can be imported and analysed. A grouping of different sequences is possible, e.g. to perform an analysis of significance or an analysis of variance. With this feature, TREVIS is also applicable in late stages of a development process as depicted in figure 1.
- A fourth method analyses these action sequences in comparison with the user models. This analysis shows the differences between the actions specified in the user models and the activities, the users performed while interacting with a prototype. According to that, this analysis can be understood as a comparison between a "perfect" user (described in the user model) and "real" user (specified in the action logs). Based on this analysis, predictions about the effectiveness and efficiency can be made. Additionally, various figures were calculated, e.g. action frequency, error rates, number of aborted action logs. These outcomes are displayed in various graphical and alphanumerical ways.

Figure 3 shows the user interface of the tool with the project management area on the left and the editing area on the right side. In the middle the different user models and action logs are shown.

6. APPLICATION

Two CD player prototypes were created to evaluate the tools features, with the user interfaces of the two players differing primarily in the number of buttons. Therefore, user models for the task "switch player on - activate programming mode - select tracks 1,6 and 10 - start program" for both prototypes were modelled.
These models were subjected to a user model analysis and a design analysis. The results of this analysis are shown in table 1.

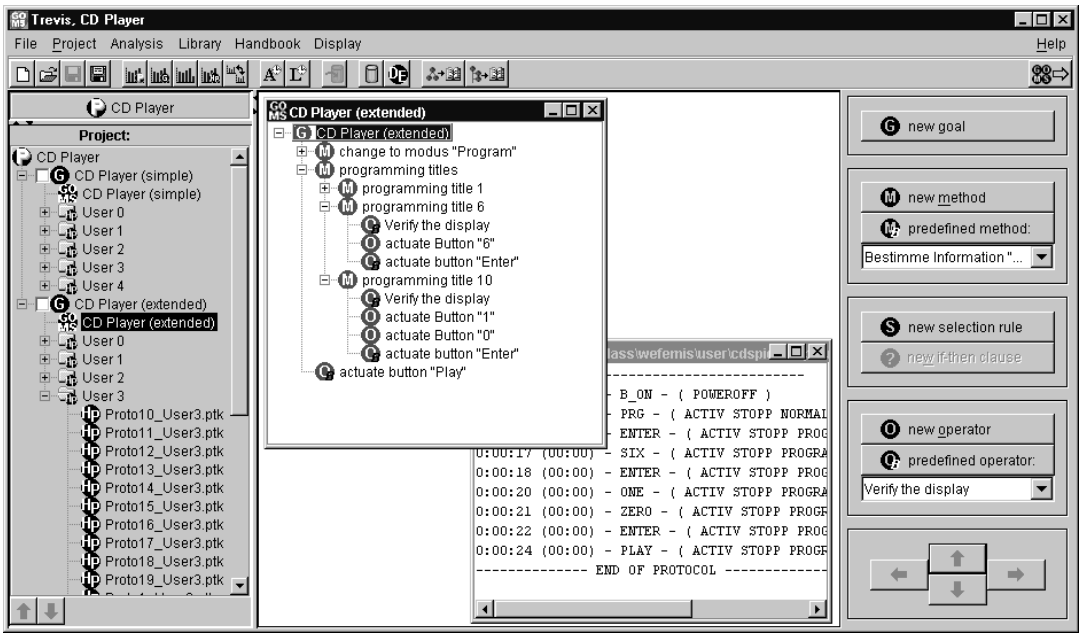

Fig. 3: The user interface of TREVIS with an exemplary project, user model, and action log.

Tab. 1: Results of the user model analyses.

	Simple Player	Extended Player
Execution time	22,5 sec	8,65 sec
No. of actions	22	10
Different actions	7	8
Action frequency (action/sec)	0,98	1,16
Learning time	3 min 41 sec	3 min 41 sec

The reason for the variation of the execution time prediction can be found in the extension of the buttons. The simple player has just the standard buttons (e.g. “play”, “stop”, “pause”) while the extended one carries a supplementary numeric keypad. On the one hand, the user can operate much faster with the extended player, on the other hand, there are more keys to learn. This fact shows in the learning time prediction.

Further on, several people interacted with the prototypes and action logs were generated, which were also analysed in the action log analysis.

Finally, for each CD player a comparison between the user model and the corresponding action logs was carried out.
This analysis algorithm indicates that of the 103/119 (simple/extended CD player) generated action logs only 83/99 could be used. The remaining logs describe aborted task operations or include actions which are not modelled in the user model. A list of the ignored action logs can be shown e.g. for further error analyses.
Another results of the comparison analysis are shown in table 2 and table 3.

Tab. 2: Results from comparison between user model of the simple CD player and action logs.

Simple CD player	TREVIS	Action logs (average)	Variance
Execution time	22,5 sec	23,43 sec	4,1 %
Actions	22	22,08	0,4 %
Action frequency (action/sec)	0,98	0,94	3,6 %

Tab. 3: Results from comparison between user model of the extended CD player and action logs.

Extended CD player	TREVIS	Action logs (average)	variance
Execution time	8,65 sec	9,41 sec	8,8 %
Actions	10	10,07	0,7 %
Action frequency (action/sec)	1,16	1,07	7,5 %

Two of the resulting diagrams are depicted in figure 4. The execution time calculated from the user models is shown as a line whereas the crosses specify the execution time of the action logs. It can be seen that most of the empirical times only vary up to 10% (grey region). Wrong actions done by the users are counted and diagrammed as columns.

Another tasks of other prototypes will be modelled to determine the quality of the tools analyses and to identify other useful parameters which can be calculated. But the results of the above example indicate that the tool is helpful to support the development engineer during the development process.

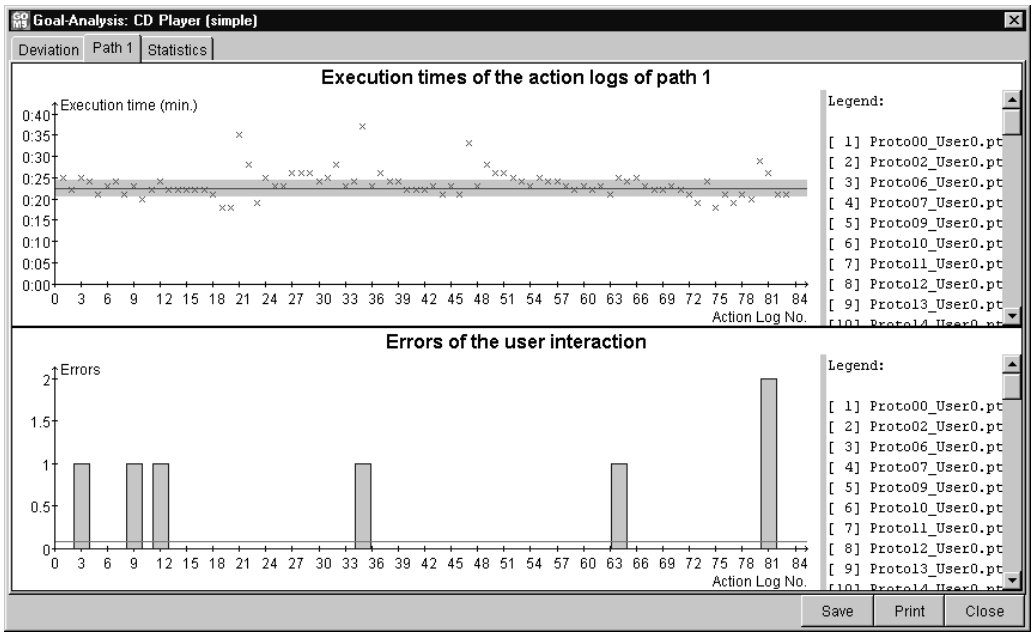

Fig. 4: Comparison of the analytical calculated and measured execution time (one prototype)

REFERENCES

Card, S., Moran, T. and Newell, A. (1983): *The psychology of human computer interaction.* Lawrence Erlbaum.

Hamacher, N. (2000): Entwicklung und Implementierung eines Werkzeugs zur Bewertung interaktiver Systeme basierend auf normativen Benutzermodellen. Department of Technical Computer Science, Diploma Theses, Aachen University of Technology.

ISO 9241-10 (1996): Ergonomic requirements for office work with visual display terminals - Dialogue principles. International Organisation for Standardisation, Genf.

ISO 9241-11 (1998): Ergonomic requirements for office work with visual display terminals - Guidance on usability. International Organization for Standardisation, Genf.

Kieras, D. (1988): Towards a practical GOMS model technology for user interface design. In: *Handbook of human-computer interaction* (M. Helander (ed.)), pp. 135-157, Amsterdam: North Holland.

Kraiss, K.-F. (1995): Modellierung von Mensch-Maschine Systemen . In: *Verläßlichkeit von Mensch- Maschine Systemen*, ZMMS - Spektrum (Willumeit H.-P. (Eds.)), **Volume 1**, pp. 15-35, Berlin

Marrenbach, J. (1999): Rapid Development and Evaluation of Interactive Systems. In: *Proceedings of the 5th ERCIM Workshop User Interfaces for All*, **Volume Report 74**, pp. 81-86, Dagstuhl

Marrenbach, J. (2001): Werkzeug-basierte Evaluierung der Benutzungsfreundlichkeit interaktiver Endgeräte mit normativen Benutzermodellen, Department of Technical Computer Science, Dissertation, Aachen University of Technology, (in print).