Enriching Task Models with Usability and User Experience Evaluation Data

Regina Bernhaupt, Philippe Palanque, Dimitri Drouet, Célia Martinie

Enriching Task Models with Usability and User Experience Evaluation Data

Regina Bernhaupt[1,3(✉)], Philippe Palanque[1,2], Dimitri Drouet[3], and Celia Martinie[2]

[1] Department of Industrial Design, Eindhoven University of Technology, Eindhoven, the Netherlands
r.bernhaupt@tue.nl
[2] IRIT, ICS, Toulouse, France
{philippe.palanque,celia.martinie}@irit.fr
[3] ruwido, Neumarkt, Austria
Dimitri.Drouet@ruwido.com

Abstract. Evaluation results focusing on usability and user experience are often difficult to be taken into account during an iterative design process. This is due to the fact that evaluation exploits concrete artefacts (prototype or system) while design and development are based on more abstract descriptions such as task models or software models. As concrete data cannot be represented, eval-uation results are just discarded. This paper addresses the problem of discrep-ancy between abstract view of task models and concrete data produced in evaluations by first, describing the requirements for a task modelling notation:(a) representation of data for each individual participant, (b) representation of aggregated data for one evaluation as well as (c) several evaluations and (d) the need to visualize multi-dimensional data from the evaluation as well as the interactive system gathered during runtime. Second: by showing how the requirements were integrated in a task modelling tool. Using an example from an experimental evaluation possible usages of the tool are demonstrated.

Keywords:
Task models · User study · Usability · User experience Evaluation · Formal description

1 Introduction

There is a fundamental belief in human-computer interaction that an iterative design and development process leads to interactive systems with a higher usability and a better user experience compared to other forms of a more sequential design and development process [3]. In user centered processes by nature iterative [34], key to success is first, to focus on the user and try to understand what users want and need to do with a system and second, to evaluate the system from early on in the design and development process [3, 33].

Evaluation results focusing on usability and user experience are often difficult to be taken into account as evaluation exploits concrete artefacts (prototype or system) while design and development are based on more abstract descriptions such as task models or software models. The connection between what users are doing (their tasks) and how

https://doi.org/10.1007/978-3-030-05909-5_9

changes in the user interface or choices in the user interface affect the overall usability and user experience of the system are difficult to understand and describe. Main drawback is the missing support to connect results and findings from user evaluation studies to the description of what users are doing and errors users make [14].

To solve this problem task models and their associated tools can be used, to store and describe the data from the evaluation study, and combine the results from the real time evaluation study with the real time visualizations of the system. Task models bring several benefits when applied in the development process: they support the assessment of effectiveness (as sub-dimension of usability [20]) [10, 37], they can support the assessment of task complexity [15, 34, 41], help in the construction of training material [27] and they support the redesign of the system [8]. Task models that are enhanced with data from user studies can be used as shared artefacts by all the stakeholders of design, development, and evaluation to enable such a connection.

To cover all possible results from an evaluation study a notation language and the related tool must support the following aspects:

(1) Represent data from the actual use of the system during evaluation: Representing data from user studies can enhance task models, for example the task model can show frequencies of choice from the user for a certain option or activity.
(2) Support analysis on scenario-basis as well as on task model level: User study results on the other side are most often not connected to the overall user goals and how scenarios are representing the real work of the users: Supporting the analysis of result based on a tasks model can provide insights beyond standard (statistical) analysis formats and usability problem reports.
(3) Represent data from each individual user in the evaluation as well as from several studies at different iterative development stages: Results from one or several user studies with the same system are often not analyzed as data is stored independently, storing all results in relation to a task model can provide means to gain insights about how changes in the user interface affect usability and especially how iterations of a system affect the overall user experience over time.

To complete such an approach of extending a notation to describe user evaluation study results, a notation is presented, a tool enabling such functionality is described and using as example an experimental evaluation of a television user interface the possible usages of the tool are demonstrated.

2 Related Work: Task Models and Data from User Studies

Introduced by [35, 36], task models for describing interactive systems are used during early phases of the user-centered development cycle to gather, understand and analyze the behavior of users when interacting with the system. A task model can be as simple as a sentence in a word document (e.g. the user is withdrawing money from the ATM) or more specific using notations like ConcurTaskTrees [36] to describe users' activities.

2.1 A Brief Overview on Notations and Their Tools

There is a broad range of notations supporting the notation of users' tasks. Table 1 gives a summary opposing two example task notations and their related tools. ConcurTask Trees [32, 41] for example enable to describe tasks in detail, providing notations to describe low-level task elements (LTL) as well as sub-task levels (STL). Recent extensions include the ability to include even errors in this notation []. CTT offers a set of operators allowing to represent temporal relationships between tasks and do have the expressive power to represent collaboration. Compared to CTT, task notations like a hierarchical task analysis (HTA) do have less expressive power. HTA needs an additional algorithm (plan) to describe temporal relationships and does not support collaborative activity. The main goal of TKS was to describe knowledge and information needed to perform a task. The notation HAMSTERS allows precise description of the task model integrating the concepts of CTT, HTA and TKS. It provides refinements to these concepts like a user task can be more refined in HAMSTERS making explicit if a user task is cognitive (analysis/decision), perceptive or motoric. CTT has only a generic user task in its notation.

Table 1. Overview on notations and their expressive power x indicating full support (x) indicating partial support

Notations	LTL	STL	ERR	OP	COOP
CTT [32], [41]	x	x	(x)		
HTA	x	(x)	(x)		
TKS		x		x	
HAMSTERS	x	x	x	x	x

Notations for task models are typically supported by tools. For CTT the corresponding tool is called CTTe, for HTA normally paper-based approaches are used. For the HAMSTERS notation the tool is having the same name. For later stages, especially when performing evaluations that ask the user to perform a task, there are evaluation tools available. For example Morae [30] allows to represent each button-press a user was performing during the study. Table 2 details the abilities of such systems, especially the ability to incorporate data from the system model (SYS), data from the user study including system, evaluator, scenario, conditions (STUD) and the ability to represent properties beyond usability like dimensions or factors of user experience (UX) and the ability to enable different types of visualization that include data visualization of more than one user study (VIZ+).

The inclusion of evaluation data including continuous data like video (but no live evaluation data) has been proposed by Mori et al. [31]. This early work did not detail how such data can be analyzed related to the different scenarios that users can perform, nor how evaluation results from different users as well as different studies would be aggregated and visualized.

Table 2. Overview on tools and their expressive power X indicating full support (X) indicating partial support

	SYS	STUD	UX	VIS
HTA	No	No	No	No
Hamsters	X	X	X	X
Morae [30]			(X)	X

2.2 Data Produced During User Evaluations

The term user evaluations in this article is used to describe a broad set of methods, or combinations of methods, that can be used at all stages in the design and development cycle of an interactive system to understand or evaluate usability and/or user experience as software quality to inform, improve and enhance the next iteration of the system design and development [23]. They necessarily involve the end-user of the interactive system. User studies can be classified using the following dimensions [21]:

(a) The stage or *phase in the software design and development process* the user study is conducted in: user studies can be used at all stages in the development process to understand end-users motivations and goals when only having a paper prototype or to evaluate an existing full-fledged system to understand how to improve for example the effectiveness with the system. At early stages user studies will more likely focus on the identification of usability problems or understanding a user experience dimension like the type of emotion the user interface design shall support.

(b) Focus on behavioral or attitudinal data: Behavioral data describes what users are doing (most often recorded with a set of video cameras to identify users body position or how the user is interacting with the product), while attitudinal data refers to users thoughts, feelings or insights.

(c) Quantitative vs. Qualitative data: When focusing on quantitative data user studies can be performed with the scientific rigor of an experiment, while qualitative data, for example textual descriptions of usability problems, can be generated with a quick and simple usability test with only five users. Following conventions from statistics data is further categorized as nominal, ordinal, interval and ratio data [16].

(d) The majority of user studies asks the user to perform a predefined set of activities (scripted), but studies can also simply ask the user to interact with the product without any detailed instructions (natural) or observe current usages and non-usages of the system (non-usage or alternative system usage). Combinations of the three forms are common in studies, e.g. asking the user to freely discover how to interact with the system (e.g. the discovery of the product), followed by a set of fixed instructions on what activities to perform.

(e) User studies can be performed in the lab or in the field, both with the option of remote participation.

(f) In terms of time, user studies are limited to few hours or less, and thus provide only a snap-shot of an experience [22]. To cover long-term user experience user studies can be repeated over time, providing data that is showing how experiences change over days, weeks or months of usage [1].

3 User Studies and Task Modelling: Problem Description

The *focus* of this work is on usability and user experience studies that can be performed at all stages of the development cycle but at least *use a functional interactive prototype and have reached a stage where the task of the user can be analyzed and described*. They involve end-users and produce behavioral and attitudinal data that can be qualitative or quantitative. The focus is on studies that are prepared with an *explicit methodological design (scripted)* that involves the *usage of the product* or service, and can be conducted in the lab or the field. Usability studies typically are performed as one session of about an hour with the product, thus deliver a snapshot of an experience. Methods that do not involve the end-user like expert methods [7] or automatically performed evaluation methods are currently not considered.

3.1 Performing a User Evaluation

A user evaluation begins with the identification of the study goals, user groups and a script how the study is performed. It can include aspects like checklists on demographics for selecting participants in the study, planning and set-up procedures for the study (e.g. balancing of conditions/participant groups). Most common for the script of the user evaluation is to give the user a description of a specific situations e.g. “Assume you are just coming home from work and you are planning what you might watch on TV together with some friends in the evening” and a task e.g. “you are browsing your electronic program guide on TV to see what is airing at 20:30”. A script consists of a number of such tasks to be performed by each of the participants of the study. A set of *n* participants is conducting the study. Each individual user performing the set of prescripted tasks is called a “run” or study run.

Once all the *n* runs are performed, data is classified, analyzed and aggregated. The presentation of the results of such studies can vary from simple reports with lists of usability problems, to sophisticated statistical analysis of the data gathered during the study. It is thus important to understand what types of data are acquired with such studies, and how the data from studying a product can be used to inform the (re-) design of a system for the next iteration of the product.

3.2 Data Produced by User Evaluation and Analysis of Results

Evaluation studies produce additional information and data including *demographic data* or users *pre-knowledge*. To understand user experience dimensions questionnaire items or interview questions can include users preferences (e.g. questions like favorite brands bought and used), but also dimensions like users’ needs or values, or descriptions of users’ behavior.

While the user is interacting with the system behavioral and attitudinal data (data can be both qualitative and quantitative) is gathered. Data can be observed directly by the evaluator during the user evaluation or data can be analyzed after the study, most common is the usage of audio recording or video.

For *user experience,* data includes answers of the study participant to interview questions or remarks from the participant, ratings e.g. naturalness of interaction, stimulation, perceived challenge and necessary skills, personal involvement the user estimates (this can be a user describing involvement in the task, answers to an open or closed interview questions or simply a value to indicate a rating on a rating scale) [11]. Data for user experience dimensions like emotion or involvement can be measured by recording heart rate or skin conductance. Emotions can be detected in facial expressions analysis recorded video data (objective) or using simple ratings from the user indicating their emotional state (e.g. using EmoCards [12]).

To summarize: data from user studies can be classified in (a) qualitative vs quantitative data with data types including nominal, ordinal, interval and ratio, (b) continuous vs. discrete data (e.g. bio-physiological measurements during the whole study vs. naturalness rating on a discrete scale) (c) data associated to the overall system (e.g. the SUS questionnaire evaluating the whole system) vs. data associated to an atomic element (like a motoric movement of the user than cannot be further disaggregated) and (d) data associated to a single event or time stamp (t) or data associated to events over periods of time. For the visualization of these data types [40] it is important to state that obviously data can have various dimensions like one-dimensional data (the overall SUS value from the questionnaire ranging from 0 to 100), two-dimensional data (eye-tracking data represented as heat maps), three-dimensional data (manipulation of 3D objects in a user interface) and multi-dimensional (when representing changed precision of the user over a learning period in a 3D user interface).

For the analysis of gathered data in the user evaluation data a variety of approaches and methods from statistics is available. Most common is the representation of data using standard descriptive statistics including mean, standard deviation and/or variance and median and modus, frequency tables or correlations. Results from questionnaires like the SUS [9] or AttrakDiff [2] are values that are based on aggregated means which are normalized. Given the study had experimental conditions all types of inferential statistics for parametric and non-parametric data can be applied indicating significant differences between the conditions of the experiment or A/B testing.

3.3 How Task Models Relate to and Complement User Studies

In a classic iterative design and development process four phases are performed iteratively: analysis, design, implementation and evaluation [3]. Goal of such an iterative process is that findings and insights from the evaluation phase inform the analysis and design phase in a relevant way, which especially in Industry is very often not the case [5]. A detailed description on how to relate and complement task models with user studies (called PRENTAM) can be found in [3].

PRENTAM starts with a task analysis producing data that can be used for task modelling. Representation of the task models can be used to extract scenarios (or select tasks) that should be used during the evaluation study. This allows to check for completeness in terms of tasks for the user evaluation study or studies performed. Due to the task model it is also possible to forecast what type of data has to be collected: e.g. when the task models shows a user input it implies that the system (or system model) receives a value that should be registered during the study or when the task model describes a motor task variables like reaction time or video data on how the user was moving/behaving have to be recorded.

Once the study was performed the data produced is analyzed and data is prepared (e.g. preparing values like mean, minimum or maximum) to be ready for injection in the task model. Inserting the data collected during the study overcomes one of the main limitations of task models not to represent real usage data. With real usage data, task models can be used to reason about possible choices and how to enhance the usability of the system e.g. if it is discovered that the majority of users is performing a certain kind of error.

For the analysis aspect of data the storage of data related to the task model can help understand and investigate the relationship of user experience and tasks performed. Especially for such types of analysis it is important to be able to represent not only one but several user studies that are performed with the system and how associated software qualities like UX change of time. Once analysis of the data in the task models completed the final step in PRENTAM is to mend task and system models if necessary.

3.4 Need for Extensions in Task Models

To enable such injection of data into task models, any task modelling tool needs to be able to:

(1) represent the variety of user evaluation data described above for each participant of the user evaluation (each "run" of the study) including (a) qualitative/quantitative; (b) continuous vs. discrete; (c) from high level data related to tasks to atomic elements of a task model (not only for high level condensed information as proposed by [31]) and (d) represent changes over time.
(2) Enable storage of all aspects of a user evaluation study (participants, system(s) used in the study, scripted scenario the participant performs, conditions (if study is performed as an experiment).
(3) Allow storage of several user studies showing changes over time.
(4) Enable visualization of key aspects for all types of data. Examples for such a visualization can be the real usage of the system with frequencies of choice for options or the visualization of changes in the user experience over time (for example by highlighting these changes with color).

4 Representing Evaluation Data in HAMSTERS

This section presents the HAMSTERS notation and its eponym tool as well as the set of extensions that have been added to them in order to support usability evaluation. According to the sections above, these extensions are centered on the notion of scenarios that is the artefact connecting concrete elements such as prototypes to abstract representations such as user tasks and goals. While Sect. 4.1 presents an overview of HAMSTERS, Sect. 4.2 details the multiple extensions related to scenarios management that were added to HAMSTERS in order to better support usability evaluation activities. In a nutshell, these extensions concern both the interactive scenario browser and the set of tools to connect multiple data sources gathered during evaluation to those scenarios.

4.1 The HAMSTERS Tool and Its Notation

HAMSTERS [29] is a tool-supported graphical task modeling notation for representing human activities in a hierarchical and ordered manner. At the higher abstraction level, goals can be decomposed into sub-goals, which can in turn be decomposed into activities. The output of this decomposition is a graphical tree of nodes. Nodes can be tasks or temporal operators. Tasks can be of several types (see Fig. 1) and contain information such as a name, information details, and criticality level. Only the single user high-level task types are presented here but HAMSTERS has a variety of further task types available.

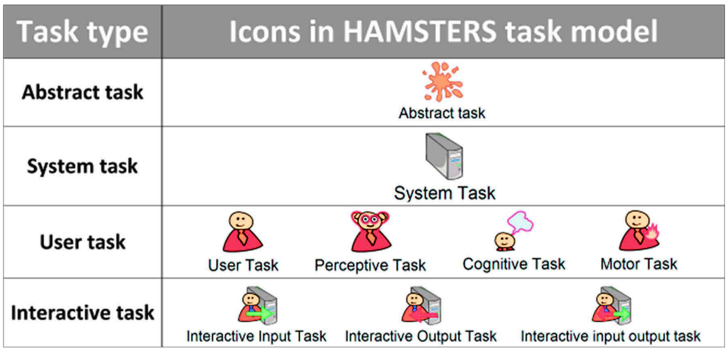

Fig. 1. High-level task types in HAMSTERS

Temporal operators are used to represent temporal ordered relationships between sub-goals and activities. Tasks can also be tagged by temporal properties to indicate whether or not they are iterative, optional or both. An illustrative example of this notation and the tool can be found in [27].

The HAMSTERS notation and tool provide support for describing and structuring a large number and complex set of tasks, introducing the mechanism of subroutines and generic components [17], and describing data that is required and manipulated in order to accomplish tasks [28]. Furthermore, as task models can be large, it is important to provide the analyst with computer-based tools for editing task models and for analyzing them. To this end, the HAMSTERS task modeling tool provides support for creating, editing, and simulating the execution of task models and can be connected to system models [24].

HAMSTERS provides support to record the steps of task models execution in **scenarios**. New scenarios can be added in HAMSTERS via the simulation panel, indicating the creation of a new scenario as a visual element that appears in the project explorer panel, on the left hand side of the HAMSTERS software environment. Figure 2 shows HAMSTERS with an active scenario representing the currently executed element highlighted (in green) and providing information about: (a) the current tasks that are available for execution (list in the upper part of the simulation panel in Fig. 2); (b) the scenario, i.e. the tasks that have been executed (list in the lower part of the simulation panel in Fig. 2) and (c) the tasks that are available for execution are highlighted in green in the task model (in the central part in Fig. 2).

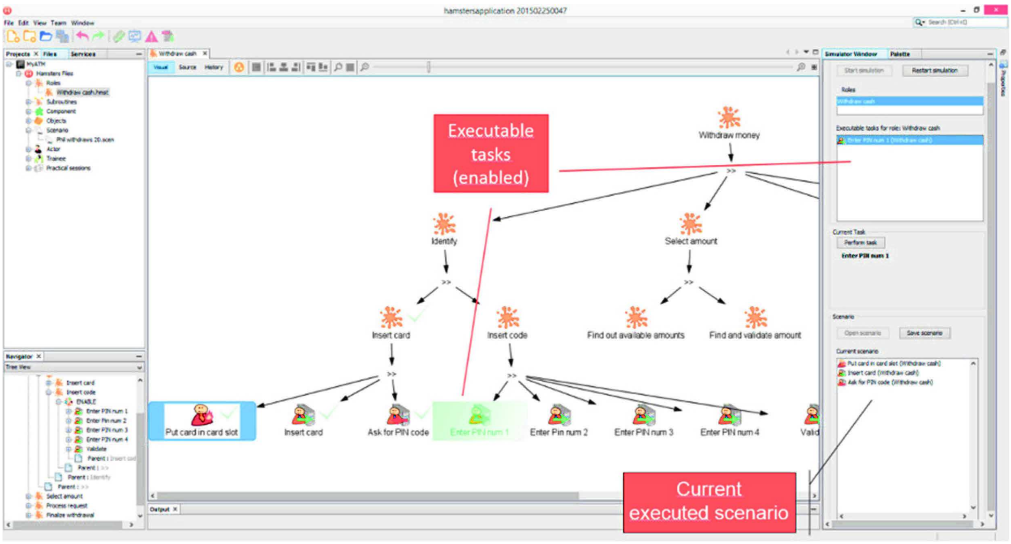

Fig. 2. Representation of executable and executed tasks during simulation

4.2 Extending Scenarios in HAMSTERS

To enable the representation of user evaluation data related to the depicted activities in the task models, HAMSTERS provides a support for executing and recording scenarios. The scenario part of the simulation panel (lower part in Fig. 2) allows browsing a scenario under construction and displays scenarios that have already been produced. However, only one scenario can be browsed at a time (in this task-model view) and the information presented are task sequence, type of task and object values that have been used during the execution of the scenario.

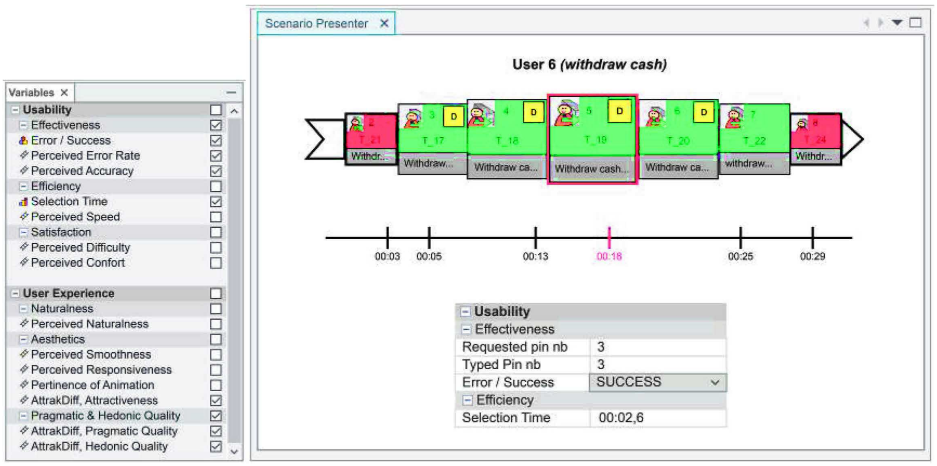

Fig. 3. The selection of variables to be displayed in the Scenario View (left) and the panel "Scenario Presenter" in HAMSTERS (right)

To display data from user studies, a dedicated panel provides support for visualization of a scenario under construction as well as of already existing scenarios. This panel is called the "Scenario Presenter" and is depicted in Fig. 3. Contrary to the model-oriented view in Fig. 2, several scenario can be visualized in this panel and for each presented scenario, two types of interactive visualizations are available a bar chart and a timeline.

The Bar Chart

The bar chart (illustrated in Fig. 3) presents the sequence of tasks that have been executed, from left to right. Each task is represented with a rectangle, displaying the following properties:

- Type of task, visually indicated by the color of the rectangle and by the icon in the top left part of the rectangle,
- task order in the scenario, visually indicated by the number displayed in the middle of the top part of the rectangle,
- task identifier, visually indicated by a code displayed in the middle of the rectangle,
- task short name, visually indicated by a text string, displayed in the lower part of the rectangle,
- data that has been eventually used by the user to accomplish the task, visually indicated by a yellow square in which a "D" is displayed (in the top right part of the rectangle).

The *Scenario Presenter* representation provides support for browsing a scenario and its sequence of tasks, as well as for interacting with them. Clicking on a task displays more information about its characteristics thanks to a dedicated pop-up window that contains details about the accomplished tasks (as well as data used or produced during the task if a "D" square is displayed in the task). As the scenario may contain a great number of tasks, this bar chart implements the fisheye interaction technique [18], enabling an overview of the whole scenario along with detail on demand. When the cursor is moved on a task, the area is centered around the cursor and then the targeted task as well as previous and next tasks are zoomed in.

The timeline (illustrated in Fig. 3) can present the start time, end time and duration of each task in the scenario. It also presents the duration of the scenario itself. Each scenario has an associated timeline that is located under the bar chart of the scenario.

The bar chart and the timeline representations are both interactive and are associated to each other for each scenario (see also Fig. 6, case study). For example, selecting a start time in the timeline will automatically position the center of the fisheye in the bar chart in the corresponding task. The panel *Scenario Presenter* provides support for comparing several scenarios, as several scenarios can be displayed one on top of each other.

4.3 Extensions to Scenarios and to Enrich Task Models

Filtering and Selection of Variables for the Scenario View

To enable the display of data gathered during user studies an interactive panel is available that allows selection of variables that shall be presented in the scenario view.

Figure 10 on the right hand side shows a sample view on such variables, currently displaying all variables for typical usability and user experience dimensions (Variables). A second panel called *Section Filtering* (not depicted) allows to select the type of tasks is displayed thus for exampling enabling the analysis of only cognitive tasks or only motor-tasks.

Representing System, Users, Scenarios/Conditions and Information
To represent information that is stemming from the user studies a set of additional information can be represented in the system (not depicted due to space constraints). A user evaluation is composed of a set of user evaluation runs that are performed in order to assess or evaluate the system. A run is made up of a set of information gathered while one user is performing a set of **planned actions (following the script)** on a given **system**. This set of planned actions is called a **scenario.** In case the user evaluation is based on an experimental set-up or is conducted as A/B testing each condition is represented as a different scenario (with a different system associated). A user evaluation is typically supervised by an evaluator. Activity of the evaluator like giving hints during a study can be represented in the task model and described as co-operative tasks if necessary.

For each condition/scenario HAMSTERS provides a set of most commonly used variables for usability and user experience (see Fig. 10, left, variable panel). In case additional variables currently not represented in HAMSTERS are necessary, the system allows to declare them by simply providing a name and scale (nominal, ordinal, interval, ratio, time, text or external data source). Currently as external data sources the integration of video is possible, the video is displayed below the time line. Furthermore, it is possible to connect HAMSTERS task models to an interactive application [26] and/or to systems' behavioral models [4]. These capability and the capability provided by the Circus environment (including HAMSTERS and Petshop) enables to co-execute all of the models and to monitor user actions and tasks in conjunction with events in the system (like waiting times due to system computation, keyboard presses or mouse clicks). The Circus environment provides support to record this data and to store it in HAMSTERS task models.

5 Demonstration of the Extensions

The tool and its extensions were for the user-centered design and development of an interactive television system. This section shows a small part of the overall project: goal is to demonstrate the usage of HAMSTERS extensions to represent data from the user evaluation.

The user interface prototype consists of a page with 12 tiles (4 columns, 3 rows), see Fig. 4 (left). During the experimentation dots appear pseudo randomized on the tiles. Users have to click on the corresponding area of the remote control to select the indicated tile. Correct selections are confirmed by displaying a green checkmark, incorrect selections with a red cross displayed on the item.

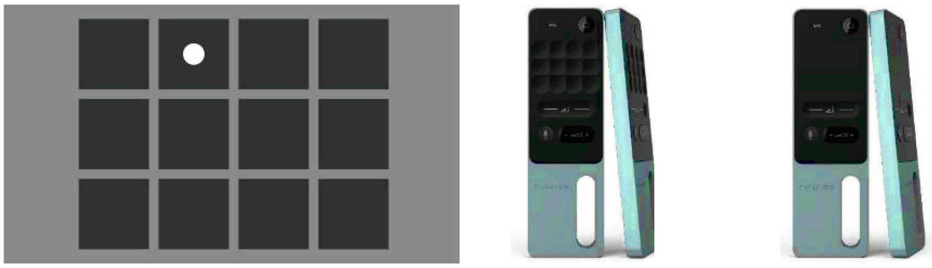

Fig. 4. The user interface and the remote controls with the touch interaction element with haptic landmarks integrated (middle) and without haptic landmarks (right)

Table 3. Dependent variables for usability and user experience dimensions shown to be not correlated in the domain of interactive TV [38]

Usability: Effectiveness	Number of errors; Accuracy [1..5]: very accurate to not accurate at all
Usability: Efficiency	Time to select a target
Usability: Satisfaction	Perceived difficulty [1..5]: very easy to very difficult; user comments on satisfaction [qualitative, text]; Comfort scale [1..5]: very accurate to not accurate at all
UX: Naturalness	Naturalness [1..5]: very natural, not natural at all
UX: Aesthetics	Smoothness [1..5]: very smooth to not smooth at all Responsiveness [1 to 5: very to not responsive Pertinence of animations [1..5]: very pertinent to not pertinent at all
UX: hedonic and pragmatic quality	Attrak Diff questionnaire sub-scales

The user study followed an experimental procedure with two independent variables: type of remote control and visual feedback or no visual feedback. Goal was to investigate if the haptic landmark would improve usability and user experience. Table 3 shows the type of data that was recorded and measured. In terms of UX the dimensions were derived from the domain specific IPTV-UX questionnaire [6].

5.1 Preparing and Performing the User Study: A Process Overview

Following the PRENTAM process [8] a task analysis was performed to identify task performed with the interactive TV system. Tasks were modelled using HAMSTERS including typical TV related tasks like selecting and changing a channel, changing volume or browsing the electronic program guide (EPG).

The process of setting up a user evaluation study based on task models, running the study and feeding back data into HAMSTERS is described in the following section in detail. The following step-by-step process demonstrates the general activities:

(1) **Task Analysis** identifying users task when interacting with TV
(2) **Task modelling** of major tasks performed with an iTV system with iterative refinements to represent even motor and cognitive tasks in the task models in HAMSTERS
(3) Setting up the user evaluation study: formulating the hypothesis for the study, preparing the **script (guidelines)** for the study on what activity each participant has to perform, developing and preparing the systems used for testing including the variations on the remote control.
(4) **Entering all study details**: Once the study details are finalized, information about system used (system with visual feedback, system without visual feedback, conditions (flat remote or remote with haptic landmarks), scenarios, participants and evaluator are entered into the HAMSTERS study set-up forms.
(5) **Declaring all dimensions and variables**: All dimensions and sub-dimensions evaluated (usability: efficiency, effectiveness, satisfaction; UX: aesthetics, naturalness, hedonic and pragmatic quality) are declared in HAMSTER. As the case study was the initial usage of HAMSTERS for such a study, all dimensions as well as variables were registered, in later stages the variables just have to be modified.
(6) **Preparing the scenarios**: Given the two independent variables, we had four conditions in the user study, thus four different scenarios were prepared.
(7) **Running the study:** The study was performed, with all participants in the study performing all four of the scenarios to have comparative measures.
(8) **Preparing and Inserting Data into Hamsters:** Based on all data recorded during the study, the data was checked for outliers, structured and inserted into the HAMSTERS tool, using the simple data entry form for scenarios and forms with varying attributes/variables for the different task types.

5.2 Associating User Study Results into HAMSTERS Scenarios

While performing the case study a variety of possible ways to enhance the integration of data was identified including automatic computation of frequencies and statistic tests or enabling users to freely add, manipulate and arrange data in data cells. But the goal of integrating user study results into HAMSTERS is not to provide functionalities of SPSS or Excel, but to support the analysis of user study results in combination with task models. The final mechanism thus simply supports manual entry and a simple import function.

To integrate user study results in HAMSTERS there are three levels of data:

- Low-level task elements: HAMSTERS provides only a basic data insertion form (see Fig. 5) that lists all currently defined variables in the system and associated task types (if applicable) for each run of a scenario. Data can be imported if strictly following the identified order of variables or is inserted manually.
- Level of activity (or sub-task): Ratings, judgements, feedback from the user, that is related to a certain activity is associated based on when in the scripted study the data was produced. Depending on the level in the task model (from abstract to concrete)

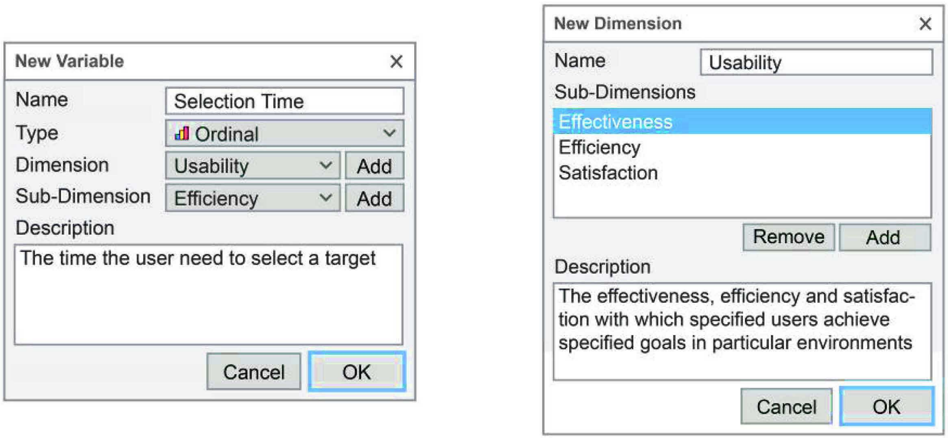

Fig. 5. Data declaration form (left) and dimension declaration form (right)

different forms are available to enter the data either as an additional variable or a standard attribute (see also Fig. 5).

- Data that is associated to abstract sub-tasks or on task or system level: the task model view can show high level data that is inserted in the overall view. This can be information from questionnaires like the SUS or AttrakDiff questionnaire that are typically used to evaluate the whole system.

5.3 Integrating and Representing User Study Results in HAMSTERS

Runs of the user study are represented using the Scenario Presenter View in Hamsters. Figure 6 shows a representation of results for user 6 performing a series of selection tasks with visual feedback given by the system (SelectTargetsFE). It shows averages for success rate, mean selection time, minimum and maximum selection time and average for the rates user experience (attractiveness) and results from the AttrakDiff [2] questionnaire.

Levels of data and their visualization:

(a) directly associated with low-level task element: Scenario Presenter, all values, time-line and video;
(b) Higher level tasks (e.g. for User 6 repeating 24 times the "selection an element task", showing mean, min/max, timeline for these summarized tasks) in the Scenario Presenter.
(c) Representing dimensions or factors like aesthetics, emotion, meaning and value, identification, stimulation, social connectedness in a task model using graphical representations like colors (see Fig. 6); representing choices, error frequencies, etc. using edges and adding information in the graph (see Fig. 6 for choice of an option).

The tool has been extended with a variety of visualization possibilities. There is a dedicated form that enables the user of the HAMSTERS tool to associate values to visualization formats accessible via the visible variables form (see Fig. 7 left). This way the user of HAMSTERS can select variables and the association of values for this variable to certain characteristics (e.g. color range for the Attractiveness Scale that was rated from [1 to 5] is represented in shades from green [1] to red [5].

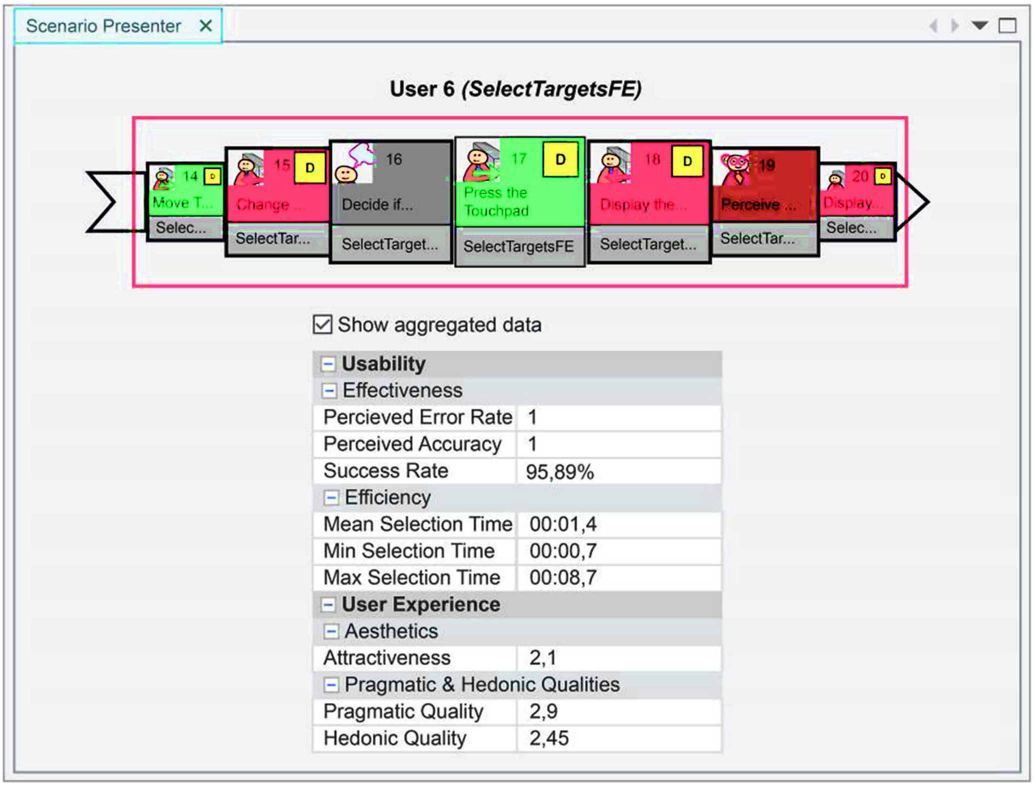

Fig. 6. Representation of user study results in the Scenario Presenter for a motor activity (Color figure online)

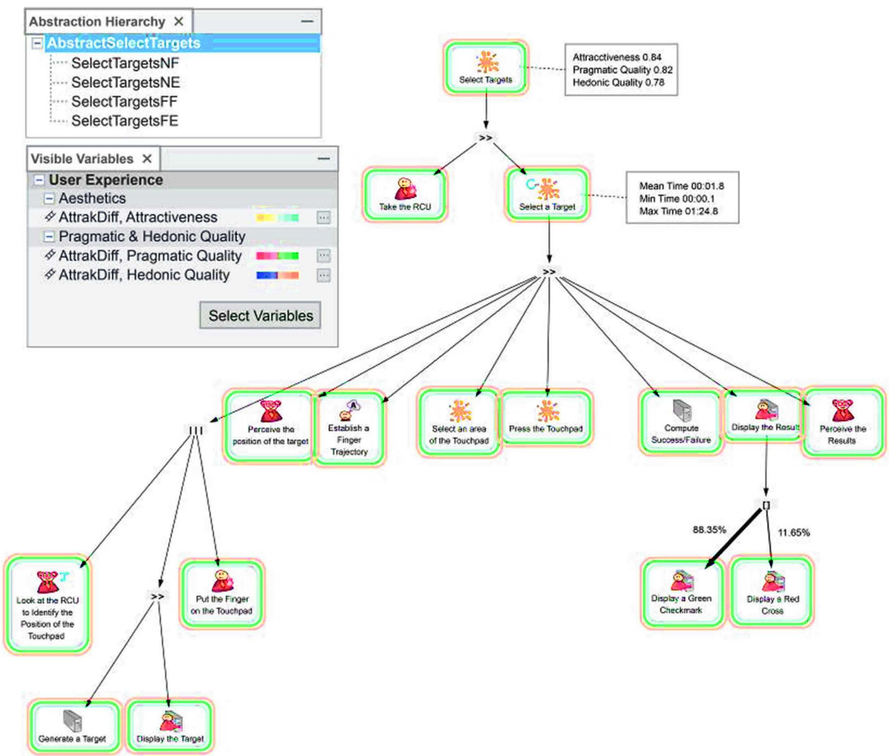

Fig. 7. Representation of user study result related to user experience in the task model indicating variations in the UX dimensions using color panels and representing frequencies of choice (%) on the connecting arrows to support usability analysis (Color figure online)

6 Discussion and Future Work

We proposed different ways to connect task models to interactive applications in the past such as the co-execution with annotated java code [26] or the connection with system models [35]. The connection of the task model with the real system is useful for designers to identify user interface objects and interactions that have to be amended (improved) but also the ones that should be used (or kept as templates) for other/similar designs as they work well. Identifying tasks and parts of the system that show a good usability and a clear contribution to an overall better user experience can help to inform not only the current system, but might also be used for similar systems or applications.

7 Future Work

For the near future the main goal is to investigate the representation of human errors in the task model thus we could make explicit where errors are expected (based on abstract knowledge of human errors) and assess whether they really occur or not. Designs could indeed overcome the presence of error precursors and prevent such errors from occurring. However, in order for designers to work on such aspects, they have to be made explicit and the task model is the only place to do so. HAMSTERS has already been extended [13] in order to represent in an explicit way errors following Hollnagel [19] and Reason [39] classifications. Future work will leverage on such descriptions to introduce results from user studies in the human error elements of task models. This would only require fine tuning of the process presented in [8] and the one on human error identification in [25].

References

Roto, V., et al.: All About UX: All UX Evaluation Methods, 17 October 2018. http://www.allaboutux.org/all-methods

Hassenzahl, M.: AttrakDiff – Fragebogen, 17 October 2018. www.attrakdiff.de

Baecker, R.M. (ed.): Readings in Human-computer Interaction: Toward the Year 2000. Morgan Kaufmann, San Francisco (1995)

Barboni, E., Ladry, J.-F., Navarre, D., Palanque, P., Winckler, M.: Beyond modelling: an integrated environment supporting co-execution of tasks and systems models. In: EICS 2010, pp. 165–174 (2010)

Bernhaupt, R., Manciet, F., Pirker, M.: User experience as a parameter to enhance automation acceptance: lessons from automating articulatory tasks. In: Proceedings of the 5th International Conference on Application and Theory of Automation in Command and Control Systems, pp. 140–150. ACM, New York (2015)

Bernhaupt, R., Pirker, M.: Evaluating user experience for interactive television: towards the development of a domain-specific user experience questionnaire. In: Kotzé, P., Marsden, G., Lindgaard, G., Wesson, J., Winckler, M. (eds.) INTERACT 2013, Part II. LNCS, vol. 8118, pp. 642–659. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40480-1_45

Bernhaupt, R., Navarre, D., Palanque, P., Winckler, M.: Model-based evaluation: a new way to support usability evaluation of multimodal interactive applications. In: Law, E.L.-C., Hvannberg, E.T., Cockton, G. (eds.) Maturing Usability. HIS, pp. 96–119. Springer, London (2008). https://doi.org/10.1007/978-1-84628-941-5_5

Bernhaupt, R., Palanque, P., Manciet, F., Martinie, C.: User-test results injection into task-based design process for the assessment and improvement of both usability and user experience. In: Bogdan, C., et al. (eds.) HCSE/HESSD 2016. LNCS, vol. 9856, pp. 56–72. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44902-9_5

Brooke, J.: SUS-A quick and dirty usability scale. In: Usability Evaluation in Industry, pp. 189–194 (1996)

Cockton, G., Woolrych, A.: Understanding inspection methods: lessons from an assessment of heuristic evaluation. In: Blandford, A., Vanderdonckt, J., Gray, P. (eds.) People and Computers XV—Interaction without Frontiers, pp. 171–191. Springer, London (2001). https://doi.org/10.1007/978-1-4471-0353-0_11

Desmet, P., Overbeeke, K., Tax, S.: Designing products with added emotional value: development and application of an approach for research through design. Des. J. **4**, 32–47 (2001)

Roto, V., et al.: Emo Cards, 17 October 2018. http://www.allaboutux.org/emocards

Fahssi, R., Martinie, C., Palanque, P.: Enhanced task modelling for systematic identification and explicit representation of human errors. In: Abascal, J., Barbosa, S., Fetter, M., Gross, T., Palanque, P., Winckler, M. (eds.) INTERACT 2015, Part IV. LNCS, vol. 9299, pp. 192– Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22723-8_16

Farenc, C., Palanque, P., Vanderdonckt, J.: User Interface evaluation: is it ever usable? In: Yuichiro, A., Katsuhiko, O., Hirohiko, M. (eds.) Advances in Human Factors/Ergonomics, vol. 20, pp. 329–334. Elsevier (1995)

Fayollas, C., et al.: An approach for assessing the impact of dependability on usability: application to interactive cockpits. In: Proceedings of the 2014 Tenth European Dependable Computing Conference, Washington, DC, USA, pp. 198–209. IEEE Computer Society (2014)

Field, A.: Discovering Statistics Using IBM SPSS Statistics. Sage Publication Ltd., London (2013)

Forbrig, P., Martinie, C., Palanque, P., Winckler, M., Fahssi, R.: Rapid task-models development using sub-models, sub-routines and generic components. In: Sauer, S., Bogdan, C., Forbrig, P., Bernhaupt, R., Winckler, M. (eds.) HCSE 2014. LNCS, vol. 8742, pp. 144– Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44811-3_9

Furnas, G.W.: Generalized fisheye views. In: Mantei, M., Orbeton, P. (eds.) Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 16–23. ACM, New York (1986)

Hollnagel, E.: Cognitive Reliability and Error Analysis Method. Elsevier Science, Oxford (1998)

ISO 9241-210 Ergonomics of Human-System Interaction Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems (2010)

Jones, M., Marsden, G.: Mobile Interaction Design. John Wiley & Sons, Chichester (2006)

Karapanos, E., Zimmerman, J., Forlizzi, J., Martens, J.-B.: Measuring the dynamics of remembered experience over time. Interact. Comput. **22**, 328–335 (2010)

Lazar, D.J., Feng, D.J.H., Hochheiser, D.H.: Research Methods in Human-Computer Interaction. John Wiley & Sons, New York (2010)

Martinie, C., et al.: Formal tasks and systems models as a tool for specifying and assessing automation designs. In: Proceedings of the 1st International Conference on Application and Theory of Automation in Command and Control Systems, pp. 50–59. IRIT Press, Toulouse (2011)

Martinie, C., Palanque, P., Fahssi, R., Blanquart, J.P., Fayollas, C.: Seguin: task model-based systematic analysis of both system failures and human errors. IEEE Trans. Hum. Mach. Syst. **46**, 243–254 (2016)
Martinie, C., Navarre, D., Palanque, P., Fayollas, C.: A generic tool-supported framework for coupling task models and interactive applications. In: Proceedings of the 7th ACM SIGCHI Symposium on Engineering Interactive Computing Systems, pp. 244–253. ACM, New York (2015)
Martinie, C., Palanque, P., Navarre, D., Winckler, M., Poupart, E.: Model-based training: an approach supporting operability of critical interactive systems. In: Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems, pp. 53–62. ACM, New York (2011)
Martinie, C., Palanque, P., Ragosta, M., Fahssi, R.: Extending procedural task models by systematic explicit integration of objects, knowledge and information. In: Proceedings of the 31st European Conference on Cognitive Ergonomics, pp. 23:1–23:10. ACM, New York (2013)
Martinie, C., Palanque, P., Winckler, M.: Structuring and composition mechanisms to address scalability issues in task models. In: Campos, P., Graham, N., Jorge, J., Nunes, N., Palanque, P., Winckler, M. (eds.) INTERACT 2011, Part III. LNCS, vol. 6948, pp. 589–609. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23765-2_40
Morae: Software, 17 October 2018. https://www.techsmith.com/morae.html
Mori, G., Paterno, F., Santoro, C.: CTTE: support for developing and analyzing task models for interactive system design. IEEE Trans. Softw. Eng. **28**, 797–813 (2002)
Nasa: Task Load Index, 17 October 2018. https://humansystems.arc.nasa.gov/groups/TLX/
Nielsen, J., Mack, R.L. (eds.): Usability Inspection Methods. John Wiley & Sons, Inc., New York (1994)
Norman, D.A., Draper, S.W.: User Centered System Design; New Perspectives on Human-Computer Interaction. L. Erlbaum Associates Inc., Hillsdale (1986)
Palanque, P., Bastide, R., Sengès, V.: Validating interactive system design through the verification of formal task and system models. In: Bass, L.J., Unger, C. (eds.) Engineering for Human-Computer Interaction, pp. 189–212. Springer, US (1996). https://doi.org/10.1007/978-0-387-34907-7_11
Paterno, F., Mancini, C., Meniconi, S.: ConcurTaskTrees: a diagrammatic notation for specifying task models. In: Howard, S., Hammond, J., Lindgaard, G. (eds.) Human-Computer Interaction, INTERACT '97. IFIP AICT, pp. 362–369. Springer, Boston (1997). https://doi.org/10.1007/978-0-387-35175-9_58
Pinelle, D., Gutwin, C., Greenberg, S.: Task analysis for groupware usability evaluation: modeling shared-workspace tasks with the mechanics of collaboration. ACM Trans. Comput. Hum. Interact. **10**, 281–311 (2003)
Pirker, M., Bernhaupt, R., Mirlacher, T.: Investigating usability and user experience as possible entry barriers for touch interaction in the living room. In: Proceedings of the 8th International Interactive Conference on Interactive TV&Video, pp. 145–154. ACM, New York (2010)
Reason, J.: Human Error. Cambridge University Press, Cambridge (1990)
Shneiderman, B.: The eyes have it: a task by data type taxonomy for information visualizations. In: Proceedings of the 1996 IEEE Symposium on Visual Languages, Washington, DC, USA, pp. 336–343. IEEE Computer Society (1996)
Swearngin, A., Cohen, M.B., John, B.E., Bellamy, R.K.E.: Human performance regression testing. In: Proceedings of the 2013 International Conference on Software Engineering, Piscataway, NJ, USA, pp. 152–161. IEEE Press (2013)