



Task Model-based Usability Evaluation for Smart Environments

Stefan Propp
Software Engineering Group
University of Rostock





Outline

1. Introduction

2. Usability Evaluation Method

2.1 Phase 1: Modelling

2.2 Phase 2: Test Planning

2.2 Phase 3: Test Execution

2.3 Phase 4: Analysis

3. User Guidance

4. Conclusion & Future Work





1. Introduction

2. Usability Evaluation Method

2.1 Phase 1: Modelling

2.2 Phase 2: Test Planning

2.2 Phase 3: Test Execution

2.3 Phase 4: Analysis

3. User Guidance

4. Conclusion & Future Work

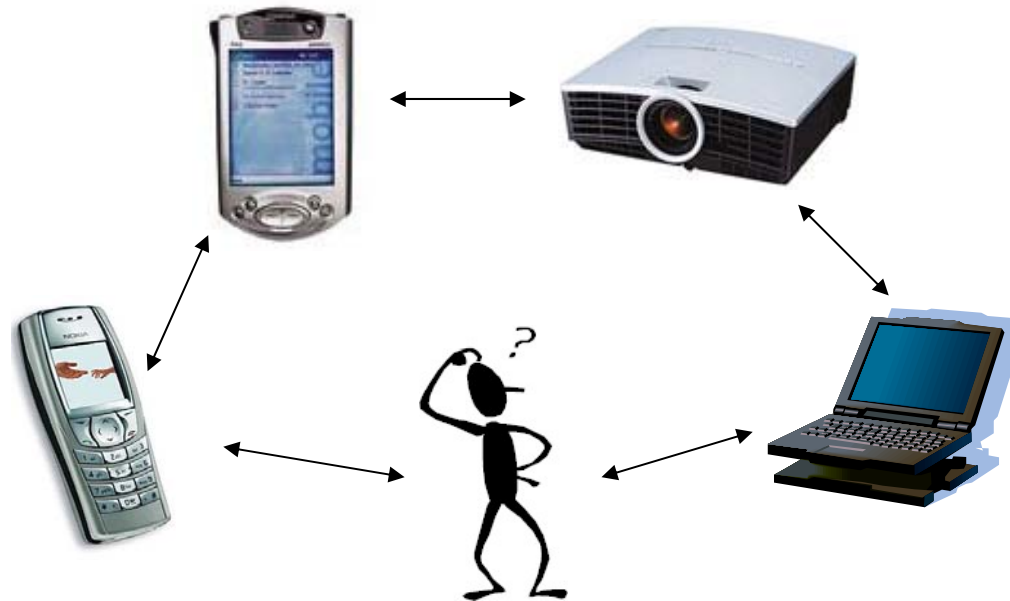




1. Introduction - Smart Environment

- Smart Environment
 - Combines everyday appliances and environments to form an ensemble
 - Individual features are composed to build more complex features

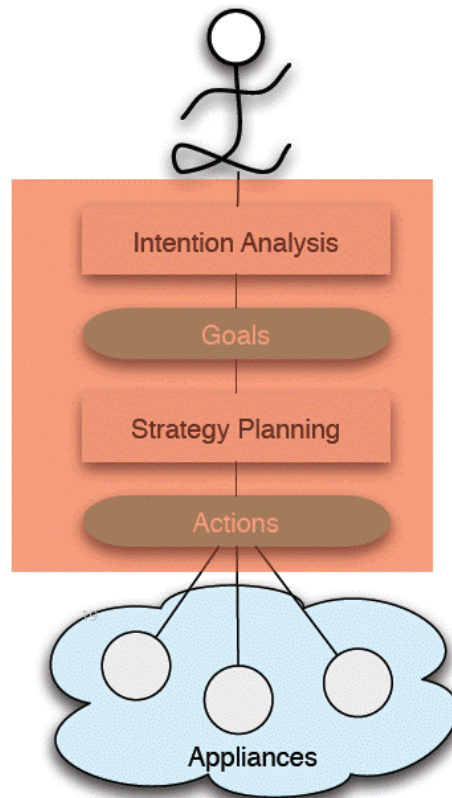
– Example:





1. Introduction - MuSAMA

- MuSAMA Project (14 PhD Students)
- Idea:





1. Introduction - Smart Environments

- **Characteristic of Human-Environment Interaction:**
[Shirehjini: A Multidimensional Classification Model for the Interaction in Reactive Media Rooms, 2007.]
 - Explicit vs. implicit initiative
 - Function vs. goal-based
 - Direct vs. dynamic device selection
 - Macros vs. dynamic strategy planning
 - Modalities (e.g. speech, gesture, ...)
 - Etc.
- **Problems concerning the Usability Evaluation:**
 - Users changing location → difficult to observe
 - Changing context influences the system behavior
 - Transitions between devices (one task, many devices)
(starting a task on one device and finishing it on another devices)
 - Cooperative work (one task, many users)
(accomplishing a task cooperatively)





1. Introduction

2. Usability Evaluation Method

2.1 Phase 1: Modelling

2.2 Phase 2: Test Planning

2.2 Phase 3: Test Execution

2.3 Phase 4: Analysis

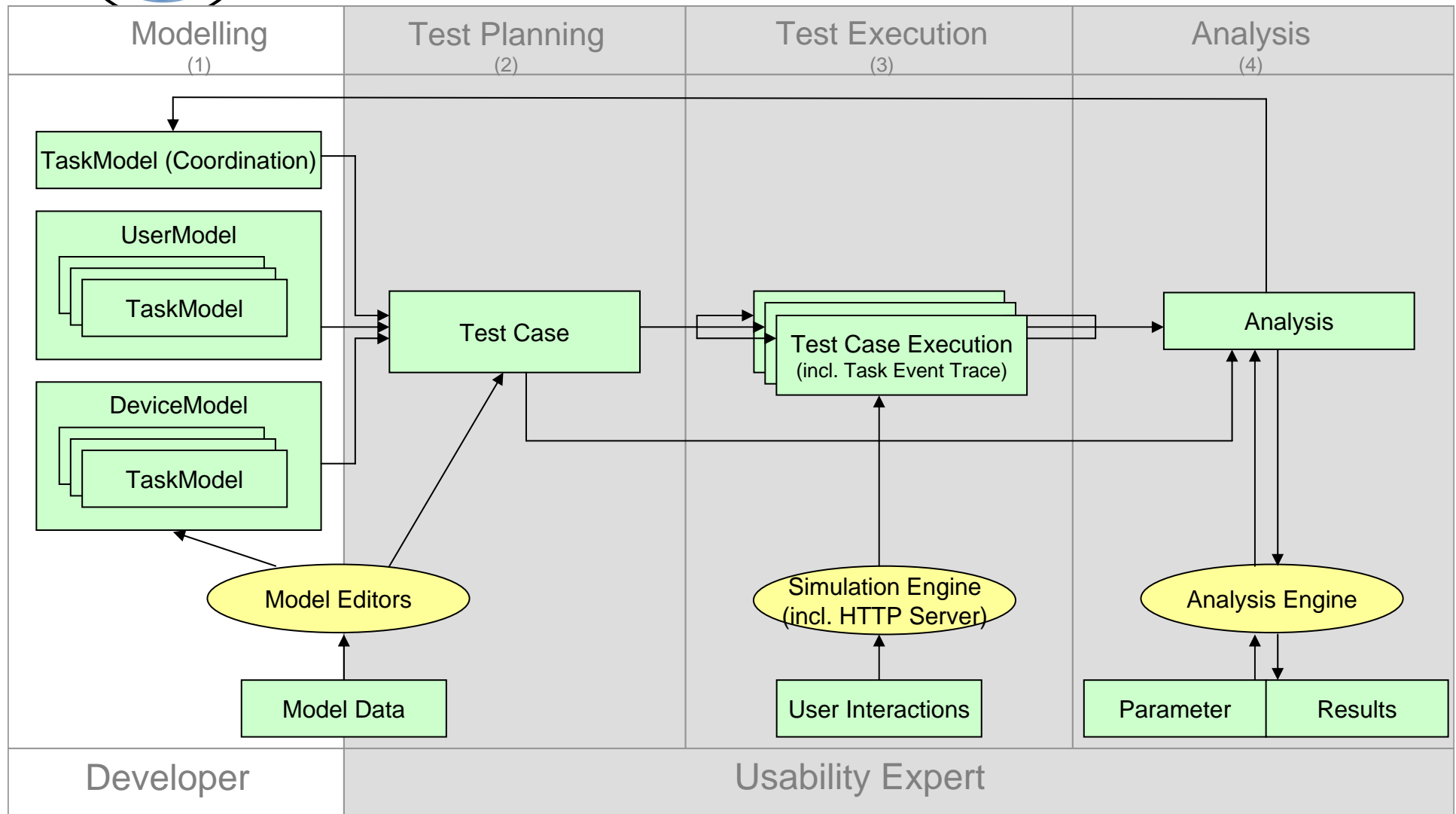
3. User Guidance

4. Conclusion & Future Work





2. Usability Evaluation Method





1. Introduction

2. Usability Evaluation Method

2.1 Phase 1: Modelling

2.2 Phase 2: Test Planning

2.2 Phase 3: Test Execution

2.3 Phase 4: Analysis

3. User Guidance

4. Conclusion & Future Work





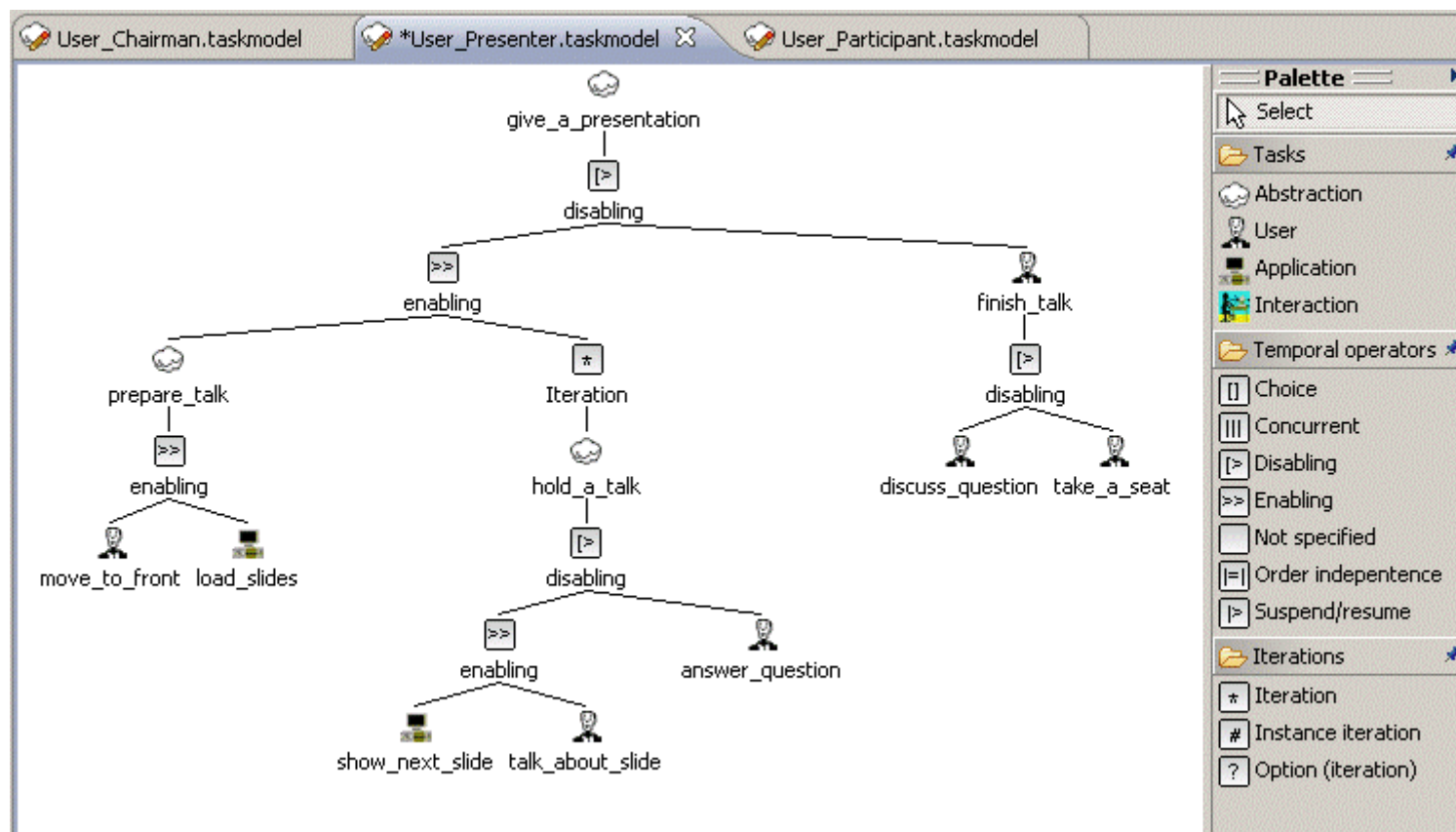
2.1 Modelling - Method

- Related work
 - Task Modelling (different notations: e.g. CTT, HTA, GOMS, ...)
[Limbourg, Vanderdonckt: Comparing Task Models for UI Design, 2003.]
 - Task Modelling for Smart Environments
 - Composing task model chunks to room task models
[Trapp, Schmettow: Consistency in use through Model based User Interface Development, CHI2006.]
 - Interpreting task models at runtime
[Feuerstack et al.: Prototyping of Multimodal Interactions for Smart Environments based on Task Models, AMI Workshop 2007.]
 - Modelling cooperative behavior with additional constraints
[Wurdel, Propp, Forbrig: HCI-Task Models and Smart Environments, HCIS 2008.]





2.1 Modelling – Tool





1. Introduction

2. Usability Evaluation Method

2.1 Phase 1: Modelling

2.2 Phase 2: Test Planning

2.2 Phase 3: Test Execution

2.3 Phase 4: Analysis

3. User Guidance

4. Conclusion & Future Work





2.2 Test Planing - Method

- „Usability Test Case“
 - Test plan with textual information
(e.g. purpose, environment description, evaluation measures)
[Rubin, J.: Handbook of usability testing. Wiley technical communication library, 1994.]
 - User and device models
(in CTT like notation)





2.2 Test Planning - Tool

The screenshot displays a task model editor with the following components:

- Task Model:** A hierarchical graph starting with 'give_a_presentation' (cloud icon). It branches into 'enabling' (double arrow icon) and 'disabling' (square with arrow icon). The 'enabling' branch leads to 'prepare_talk' (cloud icon), which further branches into 'enabling' (double arrow icon) leading to 'move_to_front' and 'load_slides' (laptop icon), and 'Iteration' (star icon) leading to 'hold_a_talk' (cloud icon). 'hold_a_talk' branches into 'enabling' (double arrow icon) leading to 'show_next_slide' (laptop icon) and 'talk_about_slide' (person icon), and 'disabling' (square with arrow icon) leading to 'answer_question' (person icon). The 'disabling' branch from 'give_a_presentation' leads to 'finish_talk' (person icon), which branches into 'discuss_question' and 'take_a_seat' (both person icons).
- Resource Set:** A tree view for 'Meeting_DGS.usabilit' showing a 'Resource Set' containing 'platform:/resource/Beispiele/m6c.exam'. Underneath is 'Usability Evaluation Meeting', which includes 'User Team Agenda', 'User Daniel', 'User Gregor', and 'User Stefan' (all with diamond icons). Below these are various devices and artifacts like 'Device Laptop of Daniel', 'Device Laptop of Gregor', 'Device Laptop of Stefan', 'Device Projector 1', 'Device ProjectionScreen 1', 'Device Ceiling Light', 'Artifact Door', 'Artifact right window', and 'Artifact left Window'.
- Palette:** A list of modeling elements including 'Select', 'Tasks', 'Abstraction', 'User', 'Application', 'Interaction', 'Temporal operators', 'Choice', 'Concurrent', 'Disabling', 'Enabling', 'Not specified', 'Order independence', and 'Suspend/resume'.
- Properties Window:** A table showing properties for the selected 'User Stefan' resource.

Property	Value
Comment	
Context Model	
Dialograph	
Domain Model	
Id	id12028387159661173
Link	
Name	Stefan
Pos X	
Pos Y	
Presentation	
Role	
Task Model	Task Model User_Presenter



1. Introduction

2. Usability Evaluation Method

2.1 Phase 1: Modelling

2.2 Phase 2: Test Planning

2.2 Phase 3: Test Execution

2.3 Phase 4: Analysis

3. User Guidance

4. Conclusion & Future Work





2.3 Test Execution – Method

Stages	early st. (Requirements analysis, Design)	later stages (Development, Deployment)
Test Object	models	running system
Process	interactively walk through the models (inspection, testing)	automatically: HTTP-connection to environment, manually: annotations of the expert (Testing)
Goal	validate models (inconsistent tasks or relations)	discover potential problems, for subsequent detailed analysis of videos etc.

- **Related Work:** [Klug: Computer Aided Observations of Complex Mobile Situations, CHI 2007.]





2.3 Test Execution - Tool

Meeting_DGS.usabilityevaluation

Resource Set

- platform:/resource/Beispiele/m6c.examples/usa
- Usability Evaluation Meeting
 - User Team Agenda
 - User Daniel
 - User Gregor
 - User Stefan
 - Device Laptop of Daniel
 - Device Laptop of Gregor
 - Device Laptop of Stefan
 - Device Projector 1
 - Device ProjectionScreen 1
 - Device Ceiling Light
 - Artifact Door
 - Artifact right window
 - Artifact left Window

*Meeting_DGS_20080229.usabilityevaluationsimulation

Team Agenda

- Meeting
 - Presentation
 - Presentation Daniel
 - Presentation Gregor
 - Presentation Stefan
 - Discussion

Gregor

- participate_in_a_presentation
 - prepare_to_participate
- * participate (10 iterations completed)
 - participate
 - sit_and_listen
 - take_notes
 - ask_a_question
 - discuss

Stefan

- give_a_presentation
 - prepare_talk
 - * hold_a_talk (4 iterations completed)
 - hold_a_talk
 - show_next_slide
 - talk_about_slide
 - answer_question
 - finish_talk
 - discuss_que
 - take_a_sea

Projector 1

- operate_projector
 - start
 - projection
 - show_video_signal
 - adjust_settings
 - finish
 - turn_off
 - cool_down

Run Task answer_question
Start Task answer_question
Stop Task answer_question
Crash Task answer_question
Forced Start Task answer_question
Forced Stop Task answer_question

Selection Parent List Tree Table »1



1. Introduction

2. Usability Evaluation Method

2.1 Phase 1: Modelling

2.2 Phase 2: Test Planning

2.2 Phase 3: Test Execution

2.3 Phase 4: Analysis

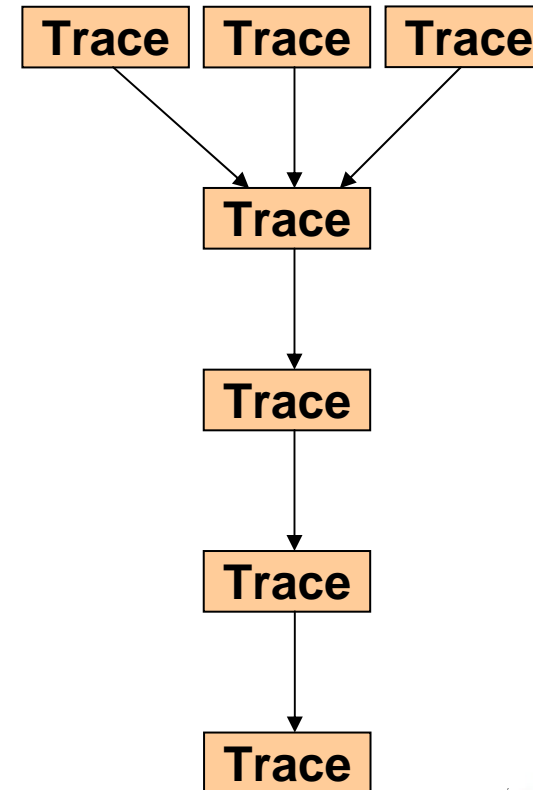
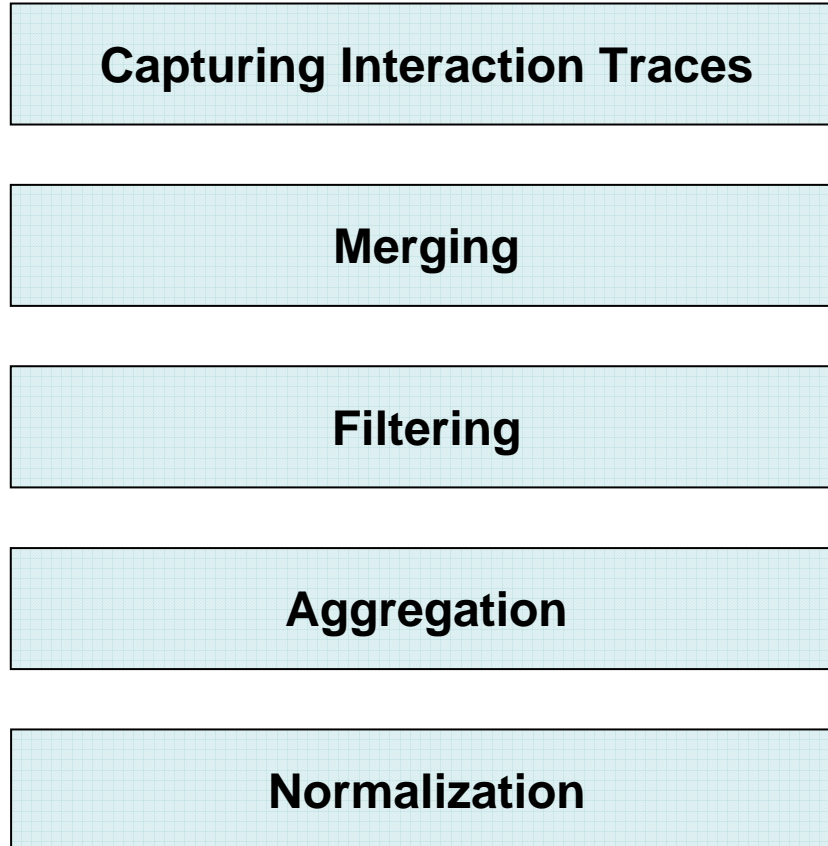
3. User Guidance

4. Conclusion & Future Work



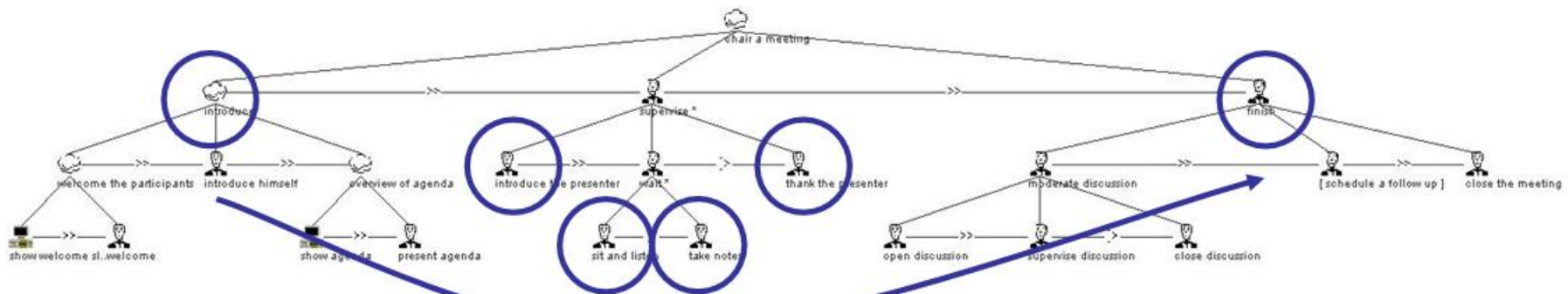
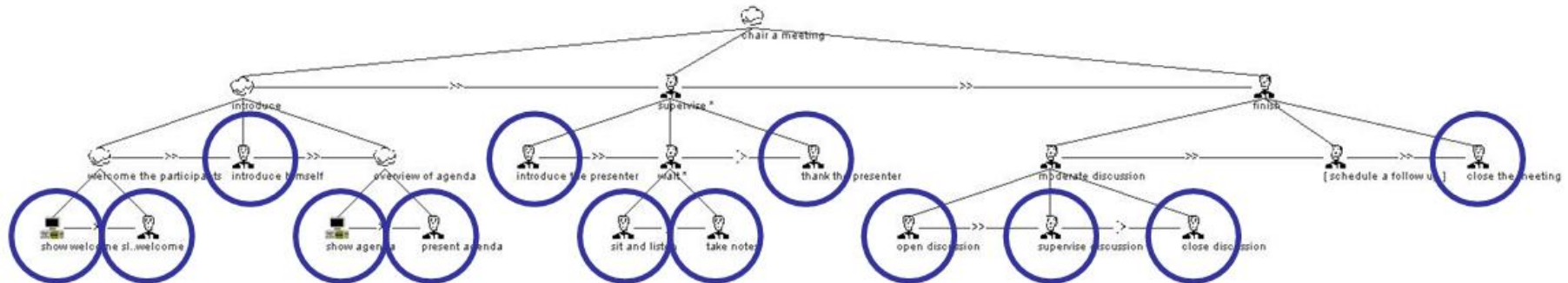


2.4 Analysis – Method (Pipeline)



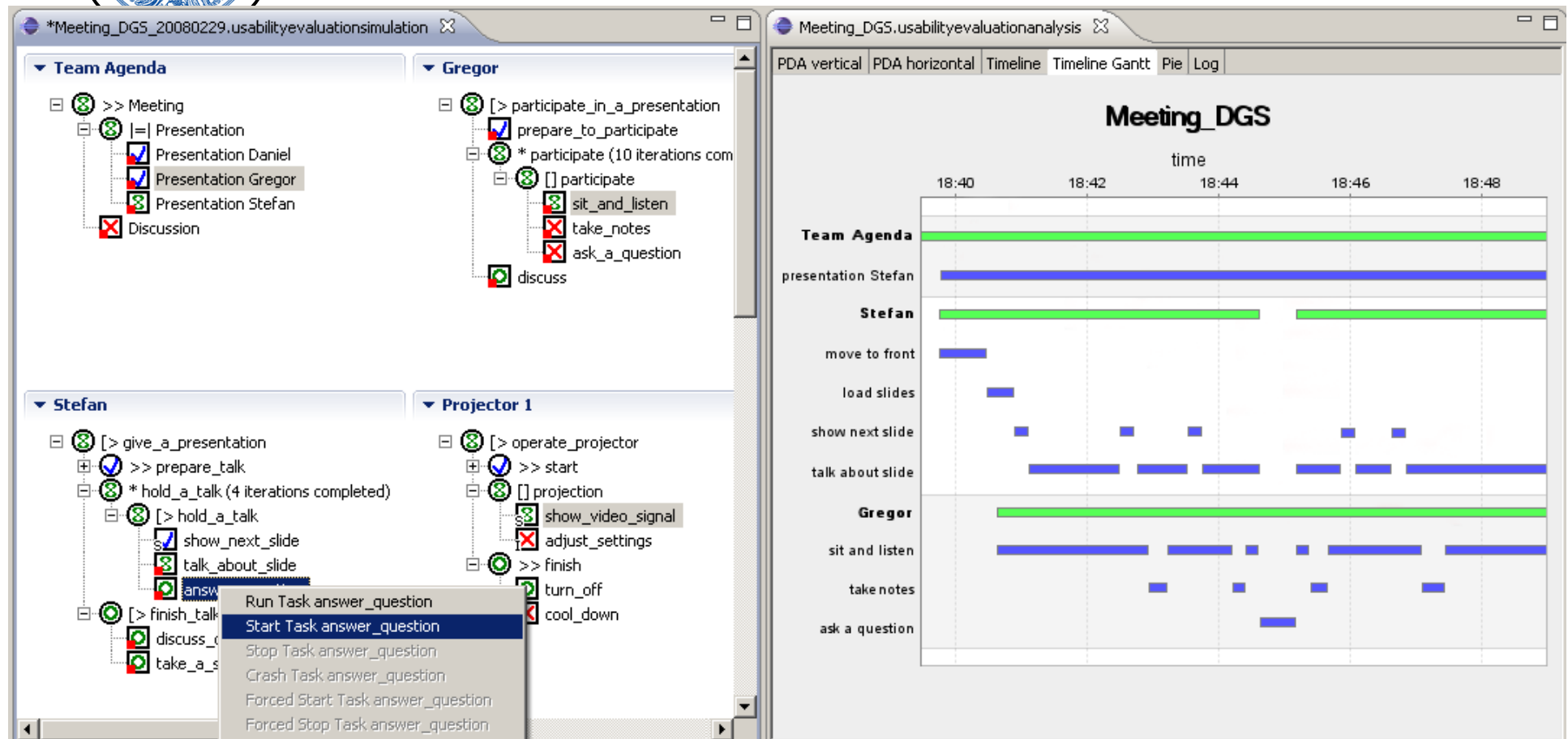


Semantic Lens





2.4 Analysis - Tool



- Related work:

- [Malý, I., Slavík, P.: Towards Visual Analysis of Usability Test Logs. Tamodia 2006.]
- [Paternò, Russino, Santoro: Remote evaluation of Mobile Applications. Tamodia 2007.]





2.4 Analysis - Method

- Support the process of error discovery and error decomposition: “
 - (1) imprecise sensor values (e.g. wrong location values),
 - (2) misinterpretations of sensor values (e.g. when applying a faulty user movement model to clean the raw sensor data),
 - (3) intention recognition errors (e.g. when predicting the wrong user task) and
 - (4) planning errors (e.g. when delivering the wrong functionality)“

[Wurdel, Propp, Forbrig: HCI-Task Models and Smart Environments, HCIS 2008.]





1. Introduction

2. Usability Evaluation Method

2.1 Phase 1: Modelling

2.2 Phase 2: Planning

2.2 Phase 3: Test Execution

2.3 Phase 4: Analysis

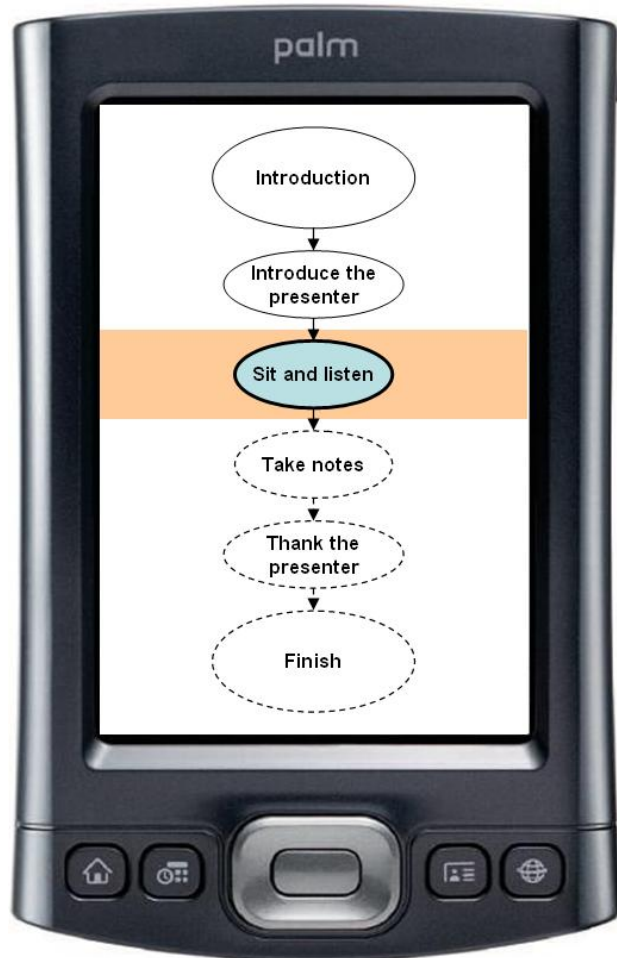
3. User Guidance

4. Conclusion & Future Work





3. User Guidance



- Idea: interactions already captured, can be further used for guidance
- Visualizing the current state of task fulfillment
 - History: captured interactions
 - Future: (1) task models include temporal relations, (2) further annotations for probabilities
[Giersich M., Forbrig P., Fuchs G., Kirste T., Reichart D., Schumann H.: Towards an integrated approach for task modeling and human behavior recognition. HCII 2007, vol. I, pp. 1109-1118, 2007.]
- Goal: visualize progress within the system transparently to improve user acceptance





1. Introduction

2. Usability Evaluation Method

2.1 Phase 1: Modelling

2.2 Phase 2: Planning

2.2 Phase 3: Test Execution

2.3 Phase 4: Analysis

3. User Guidance

4. Conclusion & Future Work





4. Conclusion & Future Work

- Usability Evaluation Method
 - Task-based approach of Usability Testing
 - Support of both: early and later development stages (simulation / execution)
 - Tool support integrates development and usability (conceptual and implementational)
- Future Work:
 - Incorporation of further Sensor values (Ubisense location detection, device states, ...)
 - Usability Test within our Smart Environment





? | !

