



**HAL**  
open science

# Exploiting Tasks Descriptions to Assess and Ensure Effectiveness during the programming of interactive Java software

Célia Martinie, David Navarre, Philippe Palanque

► **To cite this version:**

Célia Martinie, David Navarre, Philippe Palanque. Exploiting Tasks Descriptions to Assess and Ensure Effectiveness during the programming of interactive Java software. Tutoriels of 10th ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS 2018), 2018. hal-04338194

**HAL Id: hal-04338194**

**<https://hal.science/hal-04338194v1>**

Submitted on 12 Dec 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Public Domain

# Exploiting Tasks Descriptions to Assess and Ensure Effectiveness during the programming of interactive Java software

Célia Martinie, David Navarre, Philippe Palanque  
ICS-IRIT, University Paul Sabatier – Toulouse III  
Toulouse, France  
{lastname}@irit.fr

## ABSTRACT

Ensuring the effectiveness factor of usability consists in ensuring that the application allows users to reach their goals and perform their tasks [4]. The interactive application shall implement all of the needed functionalities and these functionalities shall be accessible when needed (e.g. made available to the user in an order that is compatible with the one needed to reach the goals). Ensuring the effectiveness of an interactive application thus requires an explicit description of users' goals and tasks. Task models is one of the very few means for explicitly and exhaustively describing user tasks at design time. The only mean for ensuring that this need is met requires an explicit description of users' goals and tasks. How to understand these goals and tasks and how to demonstrate that an interactive application supports them all is the main objective of this tutorial.

## CCS CONCEPTS

- **Human-centered computing** → **Human computer interaction (HCI); Interactive Systems and Tools;**

## Author Keywords

Interactive applications, task modeling.

## OVERVIEW OF THE TUTORIAL

Ensuring the effectiveness of an interactive application also requires to verify that each user actions are feasible with the interactive application (e.g. being able to click on a button that is enabled at the appropriate time) and this verification should be possible along the whole development process. On the completion of the tutorial, attendees will be able to:

- understand and navigate through HAMSTERS task models [3]
- annotate parts of Java code to tag them as related to interactive user tasks
- identify and set correspondences between

Paste the appropriate copyright/license statement here. ACM now supports three different publication options:

- **ACM copyright:** ACM holds the copyright on the work. This is the historical approach.
- **License:** The author(s) retain copyright, but ACM receives an exclusive publication license.
- **Open Access:** The author(s) wish to pay for the work to be open access. The additional fee must be paid to ACM.

This text field is large enough to hold the appropriate release statement assuming it is single-spaced in Times New Roman 8-point font. Please do not change or modify the size of this text box.

Each submission will be assigned a DOI string to be included here.

interactive tasks (in task models) and lines of code (of the interactive software)

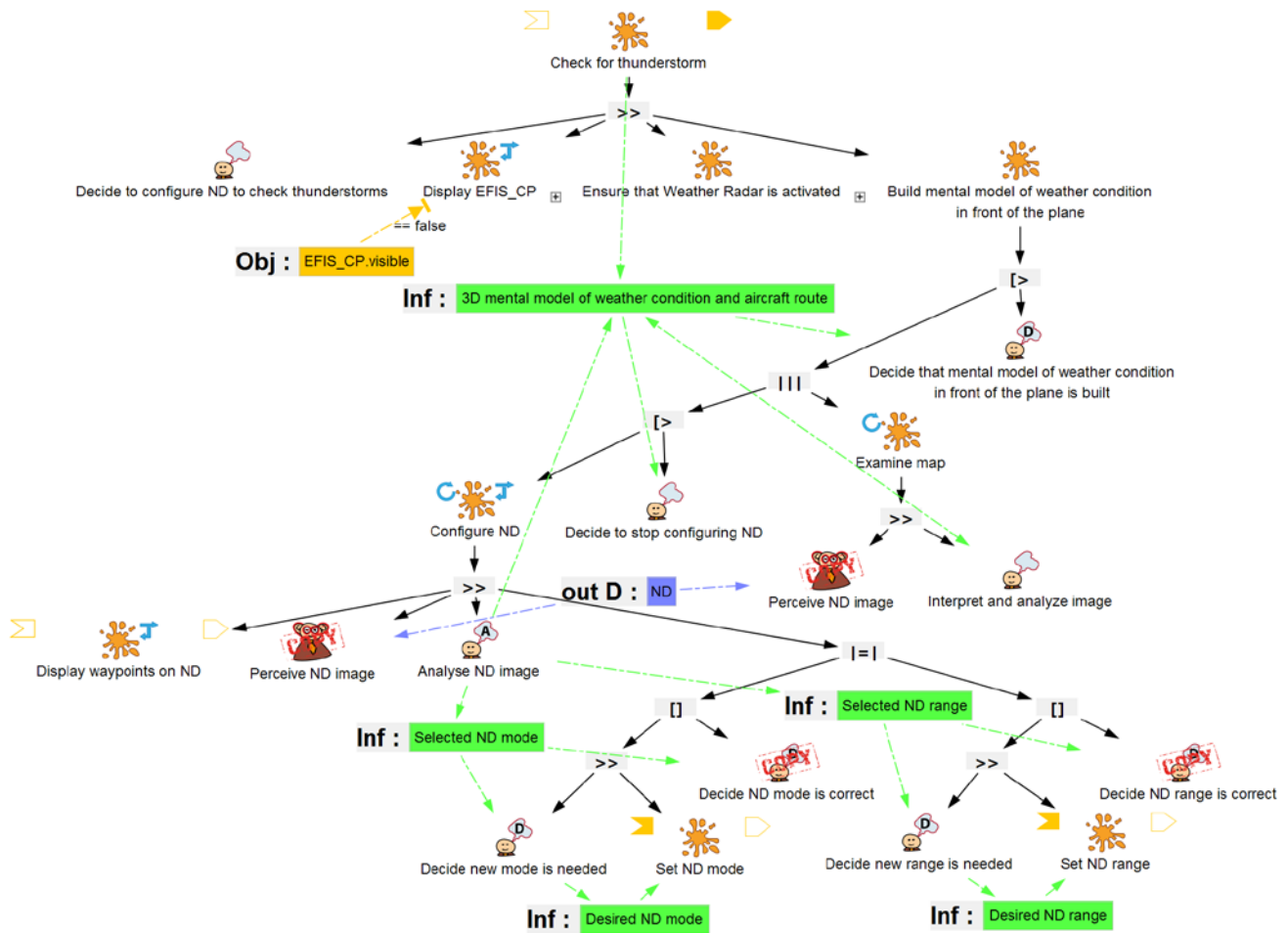
- assess compatibility between a task model and an interactive application (via co-execution of interactive Java applications and HAMSTERS task models)
- ensure (via modification of either task models or software code) compatibility between a task model and an interactive application
- setup and develop a Java-HAMSTERS project using the TOUCAN environment [1]. This point is critical as tutorial attendees will then be able to use content learnt in the tutorial for other interactive software (than the ones used in the tutorial)
- use HAMSTERS tool to modify users' goals and tasks

## DESCRIPTION AND CONTENT

Following the presentation of a set of small examples (of increasing level of complexity), the tutorial will provide an interactive hands-on exercise extracted from an industrial case study (Flight Control Unit in an aircraft cockpit, see Fig. 1).



Figure 1. Screenshot of the prototype of the FCU Software



**Figure 2. Task model of the pilot's task of checking for thunderstorm on the flight path (using the FCU Software application)**

This case study will be used as a support for presenting and explaining:

- How the effectiveness factor of usability can be checked using task models and interactive software application
- How a description of user interactive task in a task model may be connected to lines of codes in an interactive software application
- How an Integrated Development environment may support the co-development of an interactive application and its associated task models

Fig. 2 presents an excerpt of the pilot tasks (described with HAMSTERS).

#### AUDIENCE AND PREREQUISITES

This tutorial is open to researchers, practitioners, educators and students with an experience in software programming (Object Oriented and Java preferably). An experience in task analysis and task modeling is preferable but not mandatory.

#### RESOURCES PROVIDED TO ATTENDEES

The TOUCAN IDE (Integrated Development Environment) enables developers and HCI specialists to edit

correspondences between interactive user tasks and event sources and renderers in software applications. This support is achieved through the automatic extraction of interactive input and output tasks in the HAMSTERS task models and the automatic extraction of event sources and renderers from annotated applications using Java technology. TOUCAN editor also presents a view of the correspondence coverage, thus allowing developers to check the completeness of the defined correspondences. Once the correspondence between interactive tasks and event sources and renderers is completed, the TOUCAN tool provides three different means for the co-execution of the interactive application and its associated task models: task model driven co-execution, system driven co-execution and scenario driven co-execution.

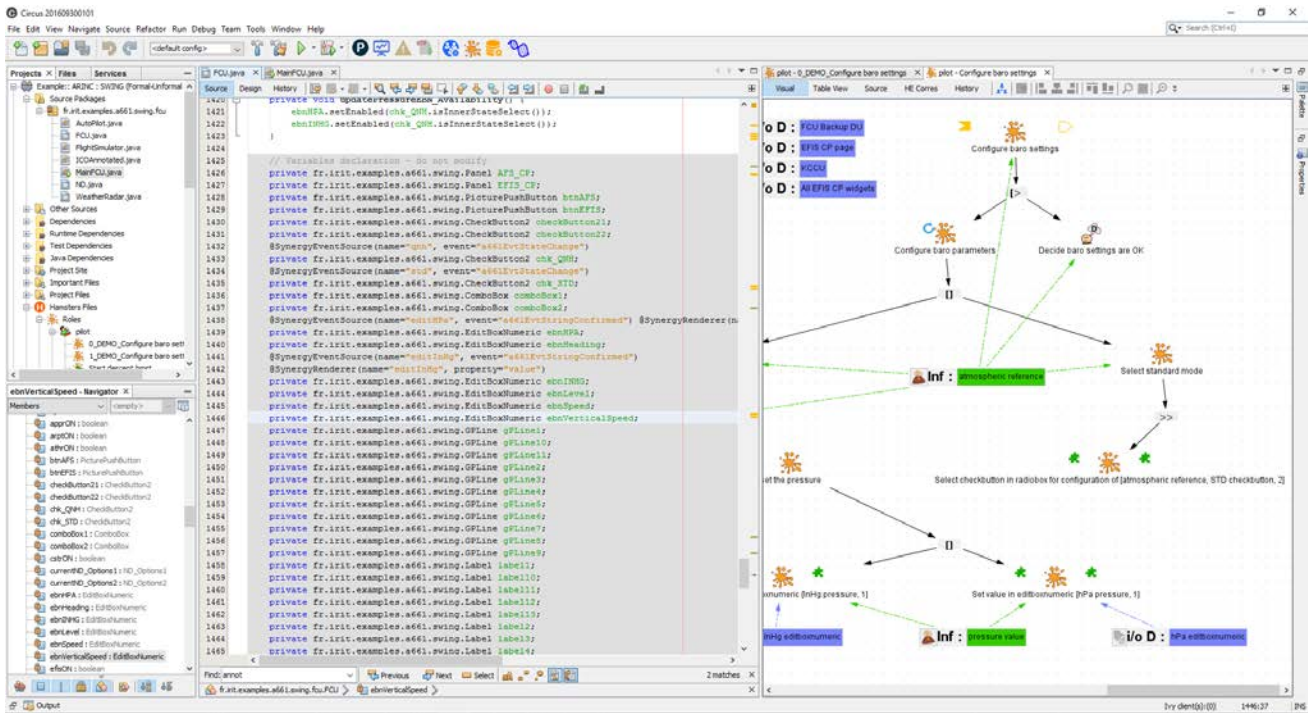


Figure 3. Screenshot of the TOUCAN environment

Each attendee of the course will receive a USB key containing and/or link to download:

- HAMSTERS task modelling tool
- TOUCAN IDE
- A set of prepared examples (sample TOUCAN projects) that will be carefully prepared for attendees to start with
- The course notes (slides and a 30-page textual documentation on how to use HAMSTERS tool)
- A video demonstrating the use of the TOUCAN and the use of the HAMSTERS tool

## ORGANIZERS

### Célia Martinie

is Assistant Professor in Computer Science at University of Toulouse 3. She has been working on task modeling techniques for the design and development of interactive systems since the beginning of her PhD in 2009. Prior to that, she worked in the mobile industry (Motorola) during 8 years, and has contributed to the design and development of user interfaces for mobile devices. She is the principal investigator of the projects related to the design and development of the HAMSTERS notation and tools. She applied the task modeling approaches to a variety of systems including satellite ground segments, interactive cockpits of large civil aircrafts and air traffic control workstations.

### David Navarre

is Lecturer in Computer Science and adjunct director of the Computer Science department at the University Toulouse 1. He has been working since 1998 on notations and tools for

the specification, prototyping, validation and implementation of Safety Critical Interactive Systems. He designed and developed the software APIs that enable synergistic mapping and execution of task descriptions with interactive applications. He has been teaching interactive software engineering for 15 years.

### Philippe Palanque

is Professor in Computer Science at University of Toulouse 3. He has been teaching HCI and task engineering classes for 20 years and is head of the Interactive Critical Systems group at the Institut de Recherche en Informatique de Toulouse (IRIT) in France. Since the late 80s he has been working on the development and application of formal description techniques for interactive system. He has worked on research projects to improve interactive Ground Segment Systems at the Centre National d'Études Spatiales (CNES) for more than 10 years and is also involved in the development of software architectures and user interface modeling for interactive cockpits in large civil aircraft (funded by Airbus). He is also involved in the research network HALA! (Higher Automation Levels in Aviation) funded by SESAR program which targets at building the future European air traffic management system. The main driver of Philippe's research over the last 20 years has been to address in an even way Usability, Safety and Dependability in order to build trustable safety critical interactive systems. As for conferences he is a member of the program committee of conferences in these domains such as SAFECOMP 2013 (32nd conference on Computer Safety, Reliability and Security), DSN 2014 (44th conference on Dependable

Systems and Networks), EICS 2014 (21st annual conference on Engineering Interactive Computing Systems) and was co-chair of CHI 2014 (32nd conference on Human Factors in Computing Systems) and research papers co-chair of INTERACT 2015.

#### REFERENCES

1. Célia Martinie, David Navarre, Philippe Palanque, Eric Barboni, and Alexandre Canny. 2018. TOUCAN: An IDE Supporting the Development of Effective Interactive Java Applications. In Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS '18). ACM, New York, NY, USA, Article 4, 7 pages. DOI: <https://doi.org/10.1145/3220134.3220136>
2. Martinie, C., Navarre, D., Palanque, P., Fayollas, C. A generic tool supported framework for coupling task models and interactive applications. In Proc. of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS'15). ACM, New York, NY, USA, 244-253.
3. Martinie C., Palanque P., Winckler M. Structuring and Composition Mechanisms to Address Scalability Issues in Task Models. IFIP TC 13 INTERACT conference, (2011) 589-609, Springer Verlag.
4. ISO 9241-11: Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability (1998).