



HAL
open science

Embedding explicit representation of cyber-physical elements in task models

Racim Fahssi, Célia Martinie, Philippe Palanque

► **To cite this version:**

Racim Fahssi, Célia Martinie, Philippe Palanque. Embedding explicit representation of cyber-physical elements in task models. IEEE International Conference on Systems, Man and Cybernetics (SMC 2016), Oct 2016, Budapest, Hungary. pp. 1969-1974. hal-01782561

HAL Id: hal-01782561

<https://hal.science/hal-01782561>

Submitted on 2 May 2018

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.



Open Archive TOULOUSE Archive Ouverte (OATAO)

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

This is an author-deposited version published in : <http://oatao.univ-toulouse.fr/>
Eprints ID : 18941

The contribution was presented at SMC 2016 : <http://www.smc2016.org/>

To link to this article URL :
<http://dx.doi.org/10.1109/SMC.2016.7844528>

To cite this version : Fahssi, Racim Mehdi and Martinie De Almeida, Celia and Palanque, Philippe *Embedding explicit representation of cyber-physical elements in task models*. (2016) In: IEEE International Conference on Systems, Man and Cybernetics (SMC 2016), 9 October 2016 - 12 October 2016 (Budapest, Hungary).

Any correspondence concerning this service should be sent to the repository administrator: staff-oatao@listes-diff.inp-toulouse.fr

Embedding Explicit Representation of Cyber-Physical Elements in Task Models

Racim Fahssi*, Célia Martinie*, Philippe Palanque*

* IRT – Université Paul Sabatier Toulouse 3, Toulouse, France
{fahssi, martinie, palanque}@irit.fr

Abstract— User interfaces for the command and control of transportation and navigation systems, such as aircraft cockpits, usually integrate several types of interaction elements: physical, hardware or software. Within these cyber-physical environments, operators have to complete their tasks manipulating these different types of elements. However, task description notations do not take into account physical and hardware aspects beyond manipulation of input devices such as mouse and keyboard. This paper identifies generic aspects of cyber-physical interactive systems and proposes extensions to operators’ tasks description techniques, to capture them. We argue that representing cyber-physical elements explicitly and systematically in task models contribute to the design and development of usable and reliable transportation systems. These extensions are integrated within the tool-supported notation called HAMSTERS and are illustrated on a case study from the avionics domain.

I. INTRODUCTION

User interfaces for the command and control of transportation and navigation systems, such as aircraft cockpits, usually integrate several types of interaction elements: physical, hardware or software. Within these cyber-physical environments, operators have to complete their tasks manipulating these different types of elements. During these activities, the performance of the users is impacted by a) the positions (within the working environment) of the cyber-physical elements composing the command and control environment and b) the time and frequency of usage of the cyber-physical elements. Task analysis and modeling provide support for ensuring compatibility between human activities and user interfaces. Task models are a very powerful artefact for describing users’ goals and users’ activity and contain numerous information extremely useful for designing usable interactive application. Indeed, task models is one of the very few means for ensuring effectiveness of the application i.e. that the application allows users to reach their goals and perform their tasks and this is nowadays identified in section 1302 of CS 25 [5]. We argue that representing cyber-physical elements explicitly and systematically in task models contribute to the design and development of usable and reliable transportation systems. Indeed, those tasks descriptions can be used for ensuring that the command and control system allows operators to perform all the required tasks and to assess the coverage of a training program with respect to operational procedures [11].

Based on the analysis of existing work on representation of cyber-physical elements associated to user tasks, this article presents a set of extensions to existing task

modeling technique and tools. These extensions are integrated within the tool-supported notation called HAMSTERS and are illustrated on a case study from the avionics domain.

The remainder of the paper is structured as follows. Section 2 presents the specificities of cyber-physical systems and how these aspects have been so far integrated in tasks descriptions. Section 3 is dedicated to the extensions of HAMSTERS notation to encompass cyber-physical systems aspects. Section 4 presents the application of the notation to a concrete example on an aircraft cockpit. Section 5 summarizes the contributions, concludes the paper and highlights future work.

II. CYBER-PHYSICAL SYSTEMS AND RELATED WORK

This section first presents the main aspects of cyber-physical systems and then provides an overview of existing work on operators’ tasks representations.

A. Main characteristics of cyber –physical systems for task descriptions

The main characteristic of cyber-physical systems (CPS) is that they integrate the physical world and specific computing systems. Within such a context, operators’ work involves manipulation of input and output devices of the computing systems (such as mouse, keyboards, display units ...) together with physical elements such as seats, knobs, levers ... To describe the operators within such a context it is necessary to describe in details:

- The work environment (possibly using a 3D representation of it),
- Each physical element having an impact on operators activities (could be static displays, physical levers ...)
- The entire interactive computing systems including user interfaces, interaction techniques, input and output devices ...

The explicit representation of all these elements will make it possible for analysts to reason about fatigue (for instance computing the quantity of movements the operators have to perform), perception, motor and cognitive loads as well as input and output articulatory distances as introduced in [17].

B. Representing cyber-physical elements in task models

Some of the existing task modeling notations actually provide support for describing objects that are required to accomplish a task and/or that are manipulated during the accomplishment of a task. For instance, using CTT notation it is possible to associate to a task [16]. However, it cannot be described as a standalone artefact and is not

graphically represented in the model which makes it impossible to represent the fact that the same object is used in several tasks. HAMSTERS notation overcome this issue by providing support to describe objects as standalone artefacts and thus to reuse and connect them to multiple tasks [12]. In addition, the concept of objects is refined in HAMSTERS and several types of objects can be described (physical objects, software objects...) which is very important for CPS as highlighted above.

C. Connecting task descriptions and User Interfaces descriptions

There are three main approaches and objectives for making explicit the representation of the operators' tasks and interactive systems: use of tasks descriptions for assessment of performance of the operator interacting with a given interactive system, generation of interactive systems from tasks descriptions and validation of conformity between tasks descriptions and interactive systems.

- Approaches targeting at the assessment of usability of User Interfaces (such as Card et al [4]) propose techniques to predict user time performances when accomplishing interactive tasks with a graphical user interface. The GOMS family of techniques provide extended capabilities for usability assessment such as functionality coverage and consistency and procedure learning time predictions [9]. In particular, the CogTool environment [18] (supporting GOMS technique), provides support for predicting performance using sequences of tasks to be assessed and their associated graphical components (2D layout UI sketches). In these approaches, interactive tasks are associated to UI components, but they are neither dedicated to describe and simulate the full set of possible tasks, nor to represent the operators' work environment. None of the work presented in that section address the workspace description and its connection and impact on operators activities.
- The generation paradigm (gathered under the term of model driven development of User Interfaces (UI)) use tasks descriptions as input artefacts. For instance, the CAMELEON framework [3] provides support for the design and development based on tasks and domain models and several approaches are based on this philosophy. Manca et al. propose a solution to handle objects in preconditions during the generation of the UI [10]. Tran et al. propose a framework taking as input task, context and domain models to generate the UI [19]. In these approaches, as the UI is generated from the task models, there is a one-way connection between tasks and UI components. The main drawbacks are that it is difficult to integrate design considerations and craft knowledge in such processes ending up with stereotyped user interfaces far away (in terms of design and interaction techniques) from leading edge applications.
- The HAMSTERS (Human-centered Assessment and Modeling to Support Task Engineering for Resilient Systems) notation has initially been designed to provide support for ensuring consistency, coherence

and conformity between user tasks and interactive system at the model level [1]. It has then been further enhanced and now encompasses notation elements such as a wide range of specialized tasks types, data and knowledge explicit representations, device descriptions, genotypes and phenotypes of errors, collaborative tasks among others. However, the work environment aspects and the physical control and display elements cannot be described using dedicated primitives, preventing its use for large-scale cyber-physical systems.

D. Connecting cyber-physical representations and task descriptions

In the application domain of smart environments, techniques for providing support to human activities have been coined recently. Fisher et al. have proposed a tool supported technique for activity recognition in 3D environments [6]. It uses scene templates and user activities description to recognize, thanks to real time scene synthesis, what a user is doing. This work actually uses correspondences between 3D objects and user tasks. However, this work does not target neither complete description of user tasks, nor the design of interactive systems. In the opposite way, Forbrig et al [8] target the design of smart environments and, for this purpose, use task models as well as 2D layout representations of the users' environment. In this work, only user roles are associated to a 2D localization.

III. EXTENDING A TASK MODELING TECHNIQUE TO ADDRESS CYBER-PHYSICAL SYSTEMS ASPECTS

This section presents the extensions that have been added to the HAMSTERS notation and its associated tool in order to provide support for systematic representation of cyber-physical elements involved in the performance of operators tasks.

A. Overview of HAMSTERS









The HAMSTERS notation enables structuring users' goals and sub-goals into a hierarchical tasks tree in which qualitative temporal relationship amongst tasks are described by operators [15]. The output of this decomposition/refinement is a graphical tree of nodes that can be tasks or temporal operators. Tasks can be of several types (depicted in **Table 1**) and contain information such as a name, information details, and criticality level.

It is important to note that only the single user high-level task types are presented here but they can be further refined. For instance, the cognitive tasks can be refined in Analysis and Decision tasks [13] and collaborative activities can be refined in several task types [12]. Temporal operators (described in [13]) are used to represent temporal relationships between sub-goals and between activities.

HAMSTERS descriptive power goes beyond most other task modeling notations particularly by providing detailed means for describing data that is required and manipulated in order to accomplish tasks [14].

The content of the task models that are produced with the HAMSTERS notation depends on the analysis that is intended to be performed with them. The level of details of the description of an action can be very high. For example, to describe the task of withdrawing money from an Automated Teller Machine (ATM), the following action can be represented in a task model: to enter the pin code of the card. It can be represented as an interactive task (labelled “enter pin code”). In more detail, it can be represented with a sequence of perceptual, motor and interactive tasks, in order to describe that the user must perceive the key that will have to be pressed, then to move her/his finger on the key and press it down before the interactive task occurs. If the aim of the task modeling is to analyze the different types of interactions needed to withdraw money, the models are only required to contain description of interactive tasks. If the aim of the task modeling is to analyze possible human errors, the models are required to contain the precise description of all the types of human actions required to accomplish the task.

Table 1. Task types in HAMSTERS

Task type	Icons in HAMSTERS task model
Abstract task	 Abstract task
User task	    User task Perceptive task Motor task Cognitive task
Interactive task	   Interactive input Interactive output Interactive input/output

The HAMSTERS notation is supported by a CASE tool for edition and simulation of models. This tool has been introduced in order to provide support for task system integration at the tool level [12]. The HAMSTERS tool and notation also provides support for structuring a large number and complex set of tasks introducing the mechanism of subroutines, sub-models and components [7]. Such structuring mechanisms allow describing large and complex activities by means of task models. These structuring mechanisms enable designers breaking down task model into several ones that can be reused in the same or different task models.

B. Extensions to HAMSTERS

The main extensions added to HAMSTERS aim at covering the various aspects of the cyber-physical systems introduced in section II.A.

The main components of these extensions are:

- The explicit representation of a 3D version of the work environment
- The explicit positioning of the operator within this environment
- The explicit description of physical elements within this environment

Other aspects such as location of software application on computing systems, input and output devices and objects (both software and physical ones) were already explicitly accounted for in HAMSTERS.

Such extensions that allow complete representation of the cyber-physical elements will allow designers to integrate all these elements in the design and assessment activities

of the workplace. For instance, it is possible to represent the fact some information on a physical device are readable by the operator while standing and not while being seated.

The fact that the transportation system is moving in a given coordinate system is purposely not taken into account in HAMSTERS. Indeed, even though the operator might be able to look through the windshield outside the work environment such information is considered as being displayed on the windshield which is thus considered as a dynamic physical display. This modeling approach has proved very efficient as models are nearly identical when describing operators' tasks being executed within a real-world environment or within a simulator.

IV. ILLUSTRATIVE EXAMPLE FROM AN AVIONICS CASE STUDY

The presented example has been extracted from Standard Operating Procedures (SOPs) [1] in the avionics application domain. SOPs consist of inspections, preparations, and normal procedures. In our case study, we focused on the preliminary cockpit preparation procedure, and in particular, on the sub-section that is dedicated to Aircraft power-up.

The flight crew performs the preliminary cockpit preparation to ensure that all required safety checks are completed. The safety checks aim at preventing the inadvertent operations on aircraft systems, identifying danger to the aircraft, and danger to the personnel. The objectives of the preliminary cockpit preparation are:

- To ensure that all safety checks are performed before applying electrical power.
- To check the liquid levels i.e. oil, hydraulic and oxygen pressure.
- To check the position of surface control levers e.g. slats/flaps, parking brake.

A. Cyber-physical elements required for preliminary cockpit preparation

To perform the preliminary cockpit preparation, the crew member uses and interacts with the following cyber-physical elements: the overhead panel, the ECAM (Electronic Centralized Aircraft Monitor) and the pedestal. The Overhead panel (see disc 1 in Figure 1) is located in the ceiling, it contains the majority of the systems-related controls (e.g. fuel, hydraulics, pressurization and electrical)



Figure 1. Overview of the A350 cockpit

order to power up the aircraft:

- Perform a general inspection of the aircraft (abstract task “General” in Figure 3)
- Check that the both Engine 1 and 2 levers are in OFF position (abstract task “ENG” in Figure 3), using the ECAM control panel (input/output device “I/O D: ECAM CP” in Figure 3)
- Check that the landing gear lever is in DOWN position (abstract task “L/G” in Figure 3), using the lever (input/output device “I/O D: L/G (lever)” in Figure 3)
- Check that all Air Data Inertial Reference System (ADIRS) knob are in NAV position (abstract task “ADIRS” in Figure 3)
- Check cockpit lights (abstract task “COCKPIT LIGHTS” in Figure 3)

A subset of the activities that have to be led to charge the batteries (abstract task “Charge batteries” in Figure 3) is presented in Figure 4. If voltage level of at least one battery is under 25V, the crew member has to charge the batteries. For that purpose, s/he first perceive that the label “AVAIL” is displayed on button EXT2 (see disc 3

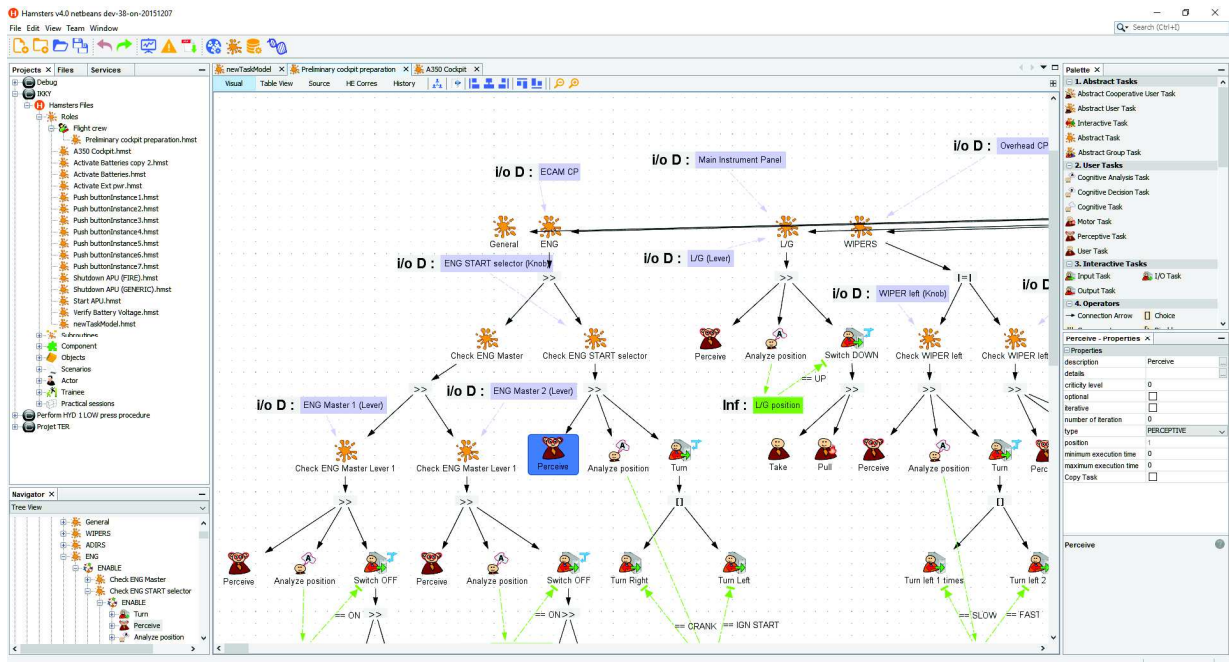


Figure 5. Screenshot of HAMSTERS (frame for editing task models)

- Check that the both WIPERS knob are in OFF position (abstract task “WIPERS” in Figure 3)
 - Check the batteries voltage (abstract task “ELEC” in Figure 3), using the Overhead panel (input/output device “I/O D: Overhead CP” in Figure 3)
- in Figure 2), which corresponds to perception tasks “Perceive” in subtree “Switch on EXT2” in Figure 4. Then, s/he analyze that the label “AVAIL” is on (cognitive analysis task “Analyse AVAIL light” in Figure 4). S/he then push the “EXT2” button (interactive input

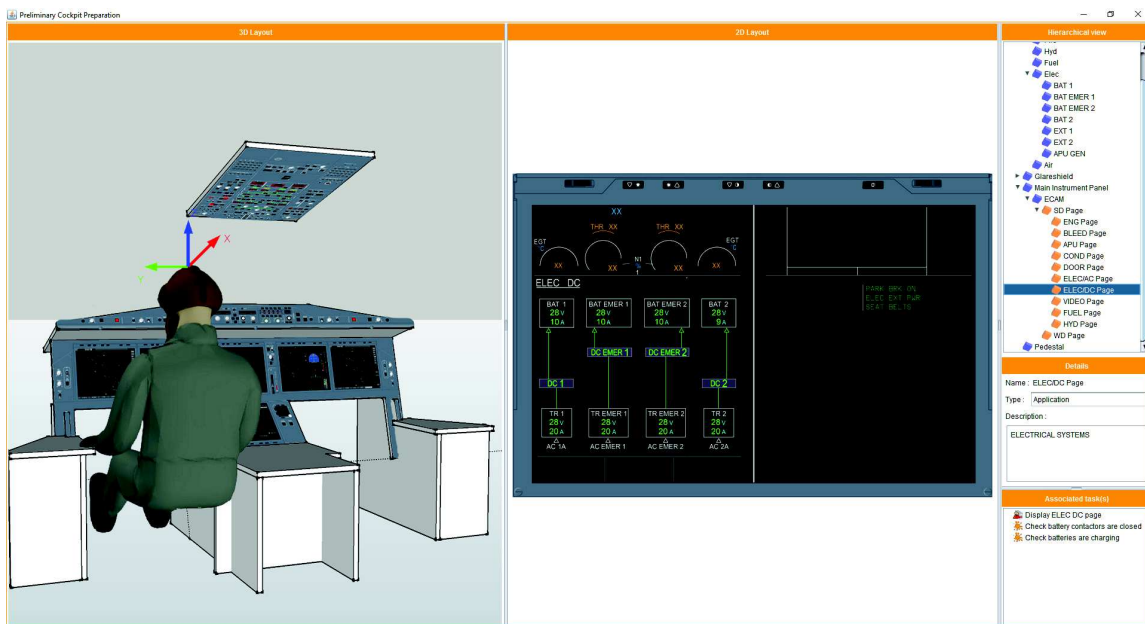


Figure 6. Screenshot of HAMSTERS (frame for visualization of 3D and 2D models with associated tasks)

task “push” in Figure 4). The AVAIL light then turns off at the same time that the ON light turns on (interactive output tasks “AVAIL light is off” and “ON light is on” under the “|||” concurrent temporal operator in Figure 4). The crew member then repeats the same operations with button EXT1 (disc 2 in Figure 2). Using the SD page of the ECAM, s/he then checks that the batteries are charging (user and interactive tasks under the abstract task “check ELEC DC page” in Figure 4).

C. Representation of the cyber-physical elements of the APU within HAMSTERS

Figure 5 and Figure 6 presents the HAMSTERS modeling environment which provides support for editing and simulating task models (Figure 5). Figure 6 depicts the extensions that have been added to integrate 3D models of the user environment, as well as 2D layout models of the interactive software applications that are part of the 3D environment. Figure 6 shows the set of frames and panels that are used in HAMSTERS to visualize cyber-physical elements. In this example, the two left frames display the 3D and 2D layouts that are associated with the task model "preliminary cockpit preparation" (depicted in Figure 3). In the left part of Figure 6, the 3D layout panel can be used to manipulate the 3D model (rotation, zoom in/out). In this illustrative example, this frame displays the cockpit of an Airbus A350: the overhead panel is located above the crew member, and the ECAM is located in the upper right in front of the crew member. The top right panel contains a hierarchical description of the devices and interactive software elements that are related to the displayed 3D and 2D layouts. In this hierarchical view panel, when the item "ELEC / DC page" is selected, the 2D layout frame is updated and displays the layout of the "ELEC / DC page" in the ECAM device, and at the same time the 3D layout frame displays a view that is zoomed in and centered on the selected device. The selection of a device or interactive software element also refreshes the bottom right panel named "Details" by displaying more information about the selected item. And the panel "Associated tasks" displays a list of all the tasks that require this selected item to be executed. In our example, we have three tasks that related to the item "ELEC / DC page" (“Display ELEC/DC page”, “Check battery contactors are closed”, “Check batteries are charging”) and these tasks are depicted in the task model "Charging Batteries" (in Figure 9).

V. DISCUSSION, CONCLUSION AND FUTURE WORK

This paper has presented how a notation for operators’ tasks descriptions could be extended in order to represent explicitly and exhaustively specific aspects of cyber-physical systems. It extends current state of the art in that domain by positioning the operator within the work environment and by connecting operators’ actions to a 3D representation of that environment. These extensions have been used in the context of operations in an aircraft cockpit highlighting how they have been integrated in the HAMSTERS too. Future work include description of multi-user and collaborative activities. For instance, the

activity of one operator might be hindered by the activity of another one. Such aspects will have to be integrated in HAMSTERS exploiting its ability to describe collaborative work [12].

REFERENCES

- [1] Airbus. A350 Flight Crew Operating Manual. 2015.
- [2] E. Barboni, J-F Ladry, D. Navarre, P. Palanque & M. Winckler, “Beyond modelling: an integrated environment supporting co-execution of tasks and systems models”. In Proc. of ACM SIGCHI symp. on EICS (pp. 165-174), 2010.
- [3] G. Calvary, J. Coutaz, D. Thevenin, Q. Limbourg, L. Bouillon, J. Vanderdonck. “A Unifying Reference Framework for multi-target user interfaces”. *Interacting with Computers* 15(3): 289-308 (2003).
- [4] S. K. Card, T. P. Moran, and A. Newell. 1980. “The keystroke-level model for user performance time with interactive systems”. *Commun. ACM* 23, 7 (July), 396 – 410.
- [5] CS-25 – Amendment 17 - Certification Specifications and Acceptable Means of Compliance for Large Aeroplanes. EASA, 2015
- [6] M. Fisher, M. Savva, Y. Li, P. Hanrahan, and M. Nießner, “Activity-centric scene synthesis for functional 3D scene modeling”. *ACM Trans. Graph.* 34, 6, Article 179 (October 2015), 13 pages.
- [7] P. Forbrig, C. Martinie, P. Palanque, M. Winckler, & R. Fahssi, “Rapid task-models development using sub-models, sub-routines and generic components”. In *Human-Centered Software Engineering* (pp. 144-163). Springer Berlin Heidelberg, 2014.
- [8] P. Forbrig, M. Zaki, G. Buchholz. “Models as a Starting Point of Software Development for Smart Environments”. *HCI* (21) 2014: 15-24.
- [9] B. E. John, D. E. Kieras. “Using GOMS for User Interface Design and Evaluation: Which Technique?” *ACM Trans. Comput.-Hum. Interact.* 3(4): 287-319 (1996)
- [10] M. Manca, F. Paternò, C. Santoro, and L. D. Spano. “Considering task pre-conditions in model-based user interface design and generation”. In Proc. of the 2014 ACM SIGCHI symposium on Engineering interactive computing systems (EICS '14), 149-154.
- [11] C. Martinie, P. Palanque, D. Navarre, M. Winckler, E. Poupart. “Model-based training: an approach supporting operability of critical interactive systems”. In Proc. Of EICS 2011: 53-62
- [12] C. Martinie, E. Barboni, D. Navarre, P. Palanque, R. Fahssi, E. Poupart, E. Cubero-Castan, “Multi-models-based engineering of collaborative systems: application to collision avoidance operations for spacecraft”. In Proc. of EICS 2014, pp. 85-94.
- [13] C. Martinie, P. Palanque, E. Barboni, & M. Ragosta, “Task-model based assessment of automation levels: application to space ground segments”. In *Systems, Man, and Cybernetics (SMC), IEEE International Conference on* (pp. 3267-3273). IEEE, 2011.
- [14] C. Martinie, P. Palanque, M. Ragosta, and R. Fahssi. “Extending procedural task models by systematic explicit integration of objects, knowledge and information“. Proc. of the 31st European Conf. on Cognitive Ergonomics (ECCE '13). ACM DL.
- [15] C. Martinie, P. Palanque, M. Winckler. “Structuring and Composition Mechanisms to Address Scalability Issues in Task Models”. *IFIP TC13 Human Computer Interaction 2011 (INTERACT)*, pp. 134-152. Springer-Verlag.
- [16] G. Mori, F. Paternò, C. Santoro: “CTTE: Support for Developing and Analyzing Task Models for Interactive System Design”. *IEEE Trans. Software Eng.* 28(8): 797-813 (2002).
- [17] D. A. Norman. 1986 *Cognitive engineering*. In *User centered system design* (ed. D. A. Norman & S. W. Draper, S. W). Hillsdale, NJ: Lawrence Erlbaum Associates.
- [18] L-H. Teo, B. John, and M. Blackmon. “CogTool-Explorer: a model of goal-directed user exploration that considers information layout”. In Proc. of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12). ACM, New York, 2479-2488.
- [19] V. Tran, J. Vanderdonck, M. Kolp, S. Faulkner. “Generating User Interface from Task, User and Domain Models”, *Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services*, 2009. CENTRIC '09, pp.19-26.