



# Contextual Interfaces for Operator-Simulator Interaction

Alexandre Kabil, Patrick Brézillon, Sébastien Kubicki

► **To cite this version:**

Alexandre Kabil, Patrick Brézillon, Sébastien Kubicki. Contextual Interfaces for Operator-Simulator Interaction. 9th International and Interdisciplinary Conference, CONTEXT 2015, Nov 2015, Lanarca, Cyprus. pp.483-488, 10.1007/978-3-319-25591-0\_36 . hal-01264390

**HAL Id: hal-01264390**

**<https://hal.archives-ouvertes.fr/hal-01264390>**

Submitted on 29 Jan 2016

**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.

# Contextual interfaces for operator-simulator interaction

Alexandre Kabil<sup>1</sup>, Patrick Brézillon<sup>2</sup>, and Sébastien Kubicki<sup>1</sup>

<sup>1</sup> ENI Brest, Lab-STICC UMR 6285, 29200 Brest, France

{kabil, kubicki}@eni.fr

<sup>2</sup> UPMC, LIP6, Paris

Patrick.Brezillon@lip6.fr

**Abstract.** We present the results of a study on the role of context in the mental representation that operators have of their task realization. This work is part of the ANR TACTIC project, which aims at proposing a migration of a simulator’s interface from PC (“click-simulation”) to tactile devices (“finger-simulation”).

**Keywords:** tactile devices, context, task realization, post-wimp interfaces, mental maps, smartphones

## 1 Introduction

In the current generation of computers with the triplet <screen, keyboard, mouse>, operators often confuse the simulator with its function, the simulation. Such click-based simulation gives the feeling to directly control the evolution of the simulation. The reason is that the operator’s mental representation results of a mixture of interpretation of the domain intertwined with an interpretation of the interface functioning. For example, by clicking on the pause button, the operator thinks to stop the simulation, while this action on the interface sends a command to the simulator that suspends the simulation. As a consequence, operator-simulator interaction is considered secondary to actions on the interface.

New technologies like tactile devices lead to new relationships between the operator and the simulator leading to finger-based simulation. However, the migration from the click-based simulation to the finger-based one supposes a change in the design to facilitate interfaces’ transfer while operator-simulator interaction stays identical. By focusing on the process that leads to an action (including the decision-making part), and not only the result of the action execution in an isolated way, it is possible to take into account the context in which the operator works effectively. By coupling context-awareness and specific interaction techniques, we can enhance the migration from an interface to another.

This paper is organized as follows. First, we will discuss the role of context in HCI (human-computer interaction), emphasizing how cognitive maps express mental representation of operators in their task realization. The next section

will introduce the modeling of operators reasoning and its interaction with a simulator. Finally, an interaction model linking gestural grammar, domain and interface actions and contextual graphs will be presented. This paper ends with a conclusion and research perspectives.

## 2 Context in HCI

### 2.1 Post-Wimp Interaction

Interaction is a *phenomenon* between a user and a computer that is controlled by the user interface running on the computer. Designing interaction rather than interfaces means that user interfaces are the means, not the end [1]. This supposes to combine and understand the context of use [2] with a special attention to the details of the interaction.

WIMP (Windows Icons Menu Pointer) for example is the most frequent interaction paradigm, used on every desktop systems. GUI (Graphical User Interfaces) are using this paradigm, which is fitted for a keyboard and a mouse usage. In order to enhance the naturalness of computer interaction, researchers proposed several approaches allowing users to go beyond WIMP interfaces. Post-WIMP is a generic term defining all interfaces that uses at least one non-WIMP control. Usually, Post-WIMP interfaces, such as tangible ones [3] takes into account context but only as a frame. The triplet <User, Platform, Environment> [4] is for instance used to constraint the whole application, by defining sets of parameters (the context of use).

This paper presents an extended context usage, by providing a model which matches specific interaction techniques to specific task-realization cases.

### 2.2 Contextual graphs

Brezillon [5] introduces the Contextual-Graphs (CxG) formalism for obtaining a uniform representation of elements of knowledge, reasoning and context. Contextual graphs are acyclic because of the time-directed representation and guarantee of algorithm termination. With a series-parallel structure, each contextual graph has exactly one root and one end node because the decision-making process starts in one state of affairs and ends in another state of affairs (generally with different solutions on the different paths) and the branches express only different contextually-dependent ways to achieve this goal. A contextual graph represents the realization of a task, and each path corresponds to a practice developed by an actor in a particular context.

The challenge to address concerns what operators are doing effectively, that is, their activity (and not their task). It is the well-known problem of distinction between task and activity [6], procedures and practices [7], logic of functioning and logic of use [8], etc. Making context explicit as contextual elements allows to consider all heterogeneous elements of context, which can be used for reasoning on scenarios.

In our research, we propose to use the contextual graph, which represents operators' behavior during their interaction with the simulator, as a "task-centered" modelling approach.

### 3 Task centered Approach

#### 3.1 Mental maps and contextual graphs

Expert's mental representation depends of their experience with the tasks attached to his role. This experience contains knowledge accumulated by the expert during his practical use of the domain knowledge along a number of task realizations in different contexts. The mental representation is a cognitive expression of the contextual knowledge related to the operator (the expert), the task at hand, the situation of the work, and the local environment in which resources are available. A cognitive map is a semi-structured expression of the mental representation that can be externalized, with classical knowledge-management tool.

The expert map corresponds to the selection of the part of the domain knowledge effectively used by participants during the realization of their tasks. As a result, the cognitive map gives a tree representation of the elements considered by participants. In terms of context, the expert map is a representation of the contextual knowledge, the part of the context that participants relate more or less directly to their task realization.

#### 3.2 Domain and Interface Actions

During task realization on a system, operators interact through an interface with a simulator that implements the simulation of a real-system evolution based on a model of this real system. Thus, the system is a model-based simulator and the interface is supposed to be part of the simulator. This is in contradiction with the goal of our project that is to allow interaction through different interfaces without changing the simulator. Confusion between simulation and simulator leads operators to assimilate specific actions of the domain (e.g. performing an action on a simulator item) with specific actions of the interface commands (e.g. stopping the simulation). We call the former *Domain actions* and the latter *Interface actions*. It leads to a serious cognitive problem because a unique domain action may be associated with several interface actions (each relevant in a specific context not made explicit), and operators would assimilate these interface actions to different domain actions for having the same information.

Thus, domain actions will be more easily associated with interface actions by taking into account the context of interaction, resulting in greater flexibility of the interface, not only with respect to the actor, but also with respect of the task realizations.

However, this approach does not describes users interaction, and we had to analyse interaction techniques that could be used on a specific Specific Command & Control (C2) system called SWORD <sup>3</sup>.

## 4 Unified Interaction Model

### 4.1 Gestural Grammar

In order to determine which interaction techniques should fit a C2 use, we took inspiration from similar systems, such as Disaster Management [9] and GIS [10]. We proposed an adapted ORBAT that allows a navigation through large hierarchies on tactile devices [11].

Through experimentations made with users on the SWORD C2 system, we managed to find interaction techniques that can be applied used when performing actions.

The table below sums up possible interaction techniques for C2 on a tactile mobile device:

Action	Possible Gesture
Unit Selection	<i>tap, swipe to select, tilt to select</i>
Contextual menu opening	<i>hold tap, double tap, Force Tap</i>
Map panning	<i>one-finger drag, two-finger drag, flick, Cyclopan</i>
Map zooming	<i>pinch Cyclozoom, Spiral zoom, tilt to zoom</i>

**Table 1.** Proposition of a Gestural Grammar for a C2 System

Merging "task-centered" and "platform-centered" approaches will enable us to facilitate users interaction with a tactile C2 system.

### 4.2 Interaction model

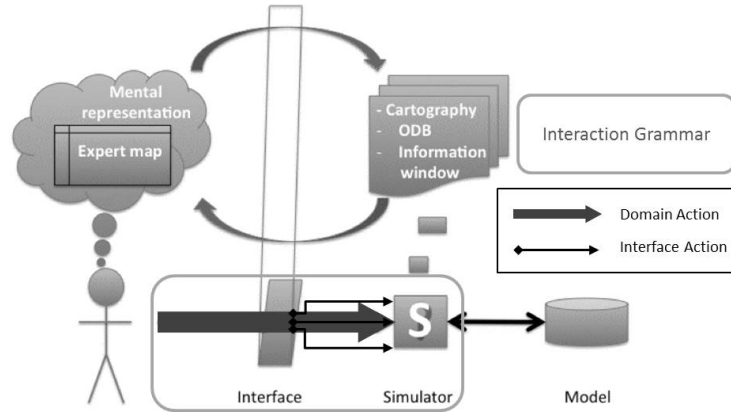
As described in the paragraph 3.2, Domain actions represents what users want to do whereas interface actions represents how they will interact.

Identifying different actions from operator's experiments help us to determine the ways in which the operator and the the simulator interact.

Figure 1 shows the two main changes in order to simplify operator's task realization with a simulator. The first one concerns a clear distinction of the interface with operator-simulator interaction. The consequence is the separation of domain actions and interface actions and a simple mechanism of translation between domain-actions and interface-actions by shifting the main problem of translation at the level of the exchange of interfaces. The second change is to use, on the one hand, the expert map as a concrete expression of the mental representation of the operator, and, on the other hand, to consider the sources

<sup>3</sup> <http://www.masagroup.net/products/masa-sword/>

of information used by the simulator as the expert map of the simulator. Thus, making compatible the expert maps of the operator and the simulator could be used to tailor the information presentation to operator in a task realization oriented way.



**Fig. 1.** A model of user-simulator interaction

We must take into account these issues in order to fit user's interaction to our model. By adapting the interaction and the visualization of the application, we could enhance its usability.

## 5 Conclusion

The main consequence of our work is flexibility of interfaces (say, with fingers movement in a touchscreen or mouse on PC). This approach proposes to adapt interface actions to domain actions, not to impose a translation of interface actions for each interface used. This opens the possibility to introduce easily news concepts for handling interfaces linked to new technologies. The operator-simulator interface then is physical as well as "cognitive". Thus, using "task-realization oriented" approach (or "oriented operator's behavior") for designing interfaces makes sense [5]. This translation of domain actions in interface actions is not the establishment of a simple lookup table because one must also take into account operator's preferences in this translation. Indeed, a path to explore is to identify the interface map of the operator similar to his expert map of the domain. This interface map is linked to the gestural grammar, as users adapt their interaction to the support.

This paper points out the distinction of domain actions and interface actions. A next step would be to make a model of the relationship between these two types of actions. A path to explore is to develop an interface map like an expert map can be developed in the domain. The operator makes an interface action

because he wants to perform a domain action. However, the links between these two types of action are not direct, and thus not natural for the operator.

## Acknowledgments

This research work was partially funded by the "Ministère de l'Enseignement Supérieur et de la Recherche", the "Conseil Général du Finistère", the "Brest Métropole Océane", and especially the "Agence Nationale de la Recherche" (TACTIC ANR Project, ANR-12-ASTR-0020). We want to thank Thibault Le Jehan, former master's degree student who worked on the Gestural Grammar.

## References

1. M. Beaudouin-Lafon, "Designing interaction, not interfaces," in *Proceedings of the Working Conference on Advanced Visual Interfaces, AVI '04*, (New York, NY, USA), pp. 15–22, ACM, 2004.
2. W. E. Mackay, "Which interaction technique works when?: Floating palettes, marking menus and toolglasses support different task strategies," in *Proceedings of the Working Conference on Advanced Visual Interfaces, AVI '02*, (New York, NY, USA), pp. 203–208, ACM, 2002.
3. S. Kubicki, S. Lepreux, and C. Kolski, "Distributed ui on interactive tabletops: Issues and context model," in *Distributed User Interfaces: Usability and Collaboration* (M. D. Lozano, J. A. Gallud, R. Tesoriero, and V. M. Penichet, eds.), HumanComputer Interaction Series, pp. 27–38, Springer London, 2013.
4. G. Calvary, J. Coutaz, D. Thevenin, Q. Limbourg, L. Bouillon, and J. Vanderdonck, "A unifying reference framework for multi-target user interfaces," *Interacting with Computers*, vol. 15, no. 3, pp. 289–308, 2003.
5. P. Brézillon, "Focusing on context in human-centered computing," *IEEE Intelligent Systems*, vol. 18, no. 3, pp. 62–66, 2003.
6. W. J. Clancey, "Simulating activities: Relating motives, deliberation, and attentive coordination," *Cognitive Systems Research*, vol. 3, no. 3, pp. 471–499, 2002.
7. P. Brézillon, "Task-realization models in contextual graphs," in *Modeling and Using Context*, pp. 55–68, Springer, 2005.
8. J. F. Richard, "Logique du fonctionnement et logique de l'utilisation," *Rapport de recherche INRIA*, 1983.
9. E. Artinger, T. Coskun, M. Schanzenbach, F. Echtler, S. Nestler, and G. Klinker, "Exploring multi-touch gestures for map interaction in mass casualty incidents," in *Workshop zur IT-Unterstützung von Rettungskraeften im Rahmen der GI-Jahrestagung Informatik*, 2011.
10. L. Ucuzal and A. Kopar, "Gis (geographic information systems) in ccis (command & control systems)," *Geographic Information, Conversion and Management Systems*, vol. 30, p. 6, 2010.
11. A. Kabil and K. Sebastien, "Richie: a step-by-step navigation widget to enhance broad hierarchy exploration on handheld tactile devices," in *Human-Computer Interaction. Mobile Interaction*, 2015.