

# Towards an assistance for selecting methods and models for interactive software design or re-engineering within complex organisation: Application case of a CPOE software

Stéphanie Bernonville, Christophe Kolski, Marie-Catherine Beuscart-Zephir

▶ **To cite this version:**

Stéphanie Bernonville, Christophe Kolski, Marie-Catherine Beuscart-Zephir. Towards an assistance for selecting methods and models for interactive software design or re-engineering within complex organisation: Application case of a CPOE software. Second IEEE International Conference on Digital Information Management, Oct 2007, Lyon, France. pp.597-602, 10.1109/ICDIM.2007.4444290 . hal-03451215

**HAL Id: hal-03451215**
**https://uphf.hal.science/hal-03451215**

Submitted on 11 Jul 2022

**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.



Distributed under a Creative Commons Attribution 4.0 International License

# Towards an Assistance for Selecting Methods and Models for Interactive Software Design or Re-engineering within Complex Organisation: Application Case of a CPOE Software

S. Bernonville
*LAMIH-UMR 8530, Univ. of Valenciennes, Le Mont Houy, F-50313 Valenciennes cedex 9, France*
*Stephanie.bernonville@univ-valenciennes.fr*

C. Kolski
*LAMIH-UMR 8530, Univ. of Valenciennes, Le Mont Houy, F-50313 Valenciennes cedex 9, France*
*Christophe.kolski@univ-valenciennes.fr*

M.C. Beuscart-Zéphir
*EVALAB-EA 2694, Faculté de Médicine, 1 place de Verdun, F-59045 Lille, France*
*mcbeuscart@univ-lille2.fr*

## Abstract

*The hospital activities display all of the characteristics of a very complex sociotechnic organisation. Consequently, the design of Computerized Physician Order Entry (CPOE) softwares which support this type of activity is very difficult. The Study of the procedure of these projects shows communication information problems between project partners. This article proposes the principles of an approach, based on Software Engineering (SE) and Human-Computer Interaction (HCI) methods and models, allowing improving the transmission of information during design or re-engineering projects within complex organisation.*

## 1. Introduction

The hospital activities display all of the characteristics of a very complex sociotechnic organisation [1]: security issue, communication between actors, information management... CPOE software which supports ordering-dispensing-administration medication process appears more and more. These softwares attempt to improve the work situations and to reduce medication errors (ex: to avoid recopying errors or to carry out automatic checking of drug interactions) [2]. However, implementing such softwares is very difficult. Design or re-engineering projects of CPOE softwares involve the participation of different partners from specifics domains such as final users, organisation specialists, ergonomists, managers, designers, developers, graphic designers… We observed procedures used for of designing and re-engineering CPOE softwares and more particularly the exchanges between ergonomists and designers / developers. Table 1 shows different problems identified with theirs consequences. These problems often come back in design or re-engineering projects, more particularly for taking into account Humans Factors in design or re-engineering [3] or for using specific methods and models by each partners [4]. To attempt to solve these problems, we propose an approach based on SE and HCI methods and models to help software design or re-engineering within complex organisation. The purpose of this approach is to create models which can be used as common work supports between project partners. It proposes modelling solutions using existing methods and models, adapting them and creating new methods if necessary [5]. The purpose of the obtained supports is to improve the transmission of relevant information within project. Figure 1 shows common work supports goals for each partner.

In this article, the approach bases will be explained in a first part that is the needs definition and the concepts presentation. Then, an example applied to a CPOE software case will be presented in a second part, that is the creation of models with our approach and the results obtained.

## 2. The approach bases

### 2.1. Needs definition

In this part, needs which we plan to take into account in our approach, are defined. These needs attempt to answer to problems concerning information transmissions within design or re-engineering projects within complex organisation. For problem 1 (see table 1), the defined need is to propose modelling solution for difficult notions to be represented by ergonomists. For problem 2 (see table 1), the defined need is to have a common representation between all projects partners. For

problem 3 (see table 1), the defined need is to propose modelling solution for missing notions to be represented by ergonomists and designers / developers. For problem 4 (see table 1), the defined need is to show relevant results from activity analysis and software analysis (i.e. organisation changes from new software integration…). We need to adopt an ecological approach [6] allowing ergonomists and designers / developers to achieve naturally the expected results. This ecological approach must correspond to ergonomists and designers / developers habits. Finally, we need to ensure the project documentation in order to store acquired knowledge and to reuse common work supports in other projects. Needs being defined, we are going to see now the approach organisation.

Table 1. The different problems identified with theirs consequences

| n° | Problems identified | Consequences |
|---|---|---|
| 1 | The existing methods and models used by ergonomists are limited to translate data from activity analysis and software (existing or in the course of design) analysis within complex organisation | The ergonomists have difficulties to transmit relevant information helping software design or re-engineering |
| 2 | The ergonomists and the designers / developers come from specific domains. They use specific methods and models | The representations of analysed work situations are different according to ergonomists and designers / developers |
| 3 | - Information transmitted by the designers / developers to the ergonomists can be inadequate to understand the analysis context, mocks-up or prototypes proposed<br><br>- Information transmitted by the ergonomists to the designers / developers can be inadequate to understand recommendations proposed | - Software evaluation is difficult to the ergonomists<br><br>- The software design or modifications do not correspond to the expectations |
| 4 | Human factors are not taken into account in software design or re-engineering | The designed software does not correspond to the final user activity. |

## 2.2. Proposal

**2.2.1. Needs for SE and HCI methods and models to obtain common work supports.** To obtain common work support between project partners, SE and HCI methods and models (with some specific methods and models from ergonomics) have been utilized. Designers / developers use usually these techniques to construct the software units and to model organisational or technical aspects. There are a lot of SE and HCI methods and models taking into account specifics elements according to different points of view: cartesian methods allowing hierarchical decomposition of process (i.e. : SADT) [7]; systemic methods allowing to exchange and utilize information modelling within organisation (i.e. : MERISE) [8]; object-oriented approaches using UML (www.omg.org); Petri Nets allowing to model dynamic aspects of system [10]. Several tools supporting these techniques have been created as Computer Aided Software Engineering (CASE). They propose modelling tools supporting specific languages and methods (i.e.: ArgoUML is a CASE supporting UML language). Other CASE combine modelling tools and development tools. SE and HCI methods and models are known to have a good potential concerning organisation analysis [12]. They can help with some adaptations, in filling in methods and models limits and lacks usually used by ergonomists to translate data issued from activity analysis within complex organisation and software analysis [13]. We used SE and HCI methods and models according two ways: (1) Usual methods and models to keep partners project customary procedures and (2) Adapted and combined methods and models to complex organisation analysis and software analysis. The second use completes the first to offer ergonomists new modelling solutions. The following part presents in details the approach concepts allowing partners project to obtain common work supports.

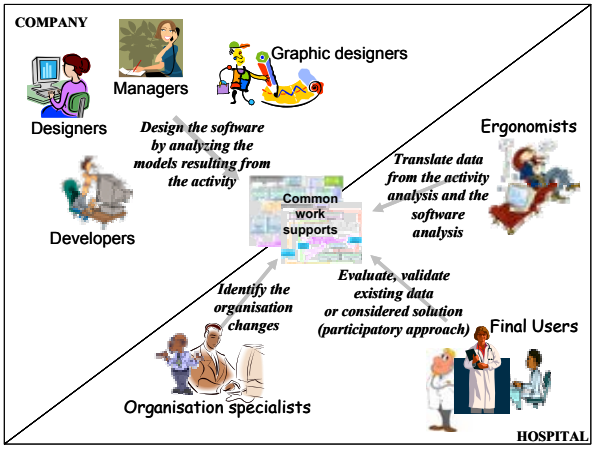



Figure. 1. Goals of common work supports

**2.2.2. The approach concepts.** We defined six concepts to construct our approach:

**Profiles**: For the moment, we distinguish the "Human-Computer Interaction specialist" profile, the « designer / developers » profile and the « organisation specialist» profile because their modelling objectives are very different.

**Modelling solutions**: We distinguish several modelling solutions: usual modelling solutions for each profile and adapted and combined modelling solutions issued from existing methods and models (for the moment, these modelling solutions concern only the "HCI" profile, particularly ergonomists)). These solutions are made up of SE and HCI methods and models (with some very specific to ergonomics). We added some methods and models well known in the SE, HCI and ergonomics domain as UML, SADT, Petri Nets, MAD [14]… Finally these methods and models have been classified in three categories: (1) HCI methods and models, (2) generic methods, models and languages and (3) organisational methods and models. Each category is made up of methods and models allowing taking into account specific modelling needs (with some adaptations if necessary).

**Modelling choices**: we distinguish two categories (modelling choices corresponding to fields of each profile and adapted choices corresponding to « ergonomists » profile. These modelling choices have been defined by us, in order to identify the project partners modelling needs. For the first category, the terms issue from the literature. They correspond to the objectives of each usual modelling solution (i.e.: for the UML activity diagram, the modelling choice is « operation structure in actions »). For the second category, the terms have been adapted in order to be more explicit for the project partners such as ergonomists which they are not used to SE and HCI methods and models in their activity (i.e.: for the adapted UML activity diagram, the modelling choice is "tasks or actions chain for an actor").

**Modelling context identification**: we distinguish two categories: modelling context identification corresponding to the different steps of Human Factors lifecycle (i.e.: « work situation analysis », « evaluation »…) and modelling context identification corresponding to the different steps of interactive system development lifecycle (i.e.: "requirements analysis", "analysis", "design stage" stages). Each project partner belonging to a profile can identify the model that he/she wants to create.

**Graphics elements taken into account by each modelling solutions:** a set of graphics elements is assigned to each modelling solutions (i.e.: for the UML activity diagram, the graphic elements can be swimlanes, actions, flow, UML objects…). These graphics elements issue from the literature. They correspond to the modelling possibilities proposed by each methods and models of SE and HCI (with some very specific to ergonomics).

The possibilities for each category, previously mentioned have been drawn up in the form of lists. The following part describes the connection between these different concepts.

### 2.3. The approach dynamic structure

The figure 2 shows the connection between the concepts to come to common work supports. Referring to this figure, one can see that initial state is starting data corresponding to collected data to be modelled, issued from activity analysis, software analysis, design stage… The first step is the profile identification ("designers / developers", "organisation specialist" or "Human-Computer Interaction specialist"). Then, the step two is the identification of the modelling context which is different according to the profile. The user identifies their modelling choices. One or several modelling solutions are proposed. The data entry form with the graphics elements corresponding to the selected modelling solution displays. Finally, the user realises the model. A complement step can be used, the user can analyse the created model (The step principle is not described in this article). The final step is created model can be used as the common work support between project partners.

The concepts and the dynamic structure of the approach have been explained. Now we are going to see an application example of our approach for a CPOE software case.

## 3. Application example: case of a CPOE software

### 3.1. Context

We interested in a software which was the subject of a (re)design project. This software is a web application which was designed to replace an existing software. Its purposes are the ordering and the planning of medication orders in hospital. During this project, designers have been entrusted mocks-up to HCI specialists, which is more particularly an ergonomists team, in order to be analysed. The ergonomists general purpose was to analyse the potential integration of the future software within an existing organisation. We interested particularly in the acquisition information of the physician during the ordering activity. Firstly, we are going to see the ergonomists collected data issued from an activity analysis performed by ergonomists. Then, we will see how to obtain a common work support from the collected data, showing the results analysis.

### 3.2. Collected data

Thanks to the activity analysis, ergonomists collected necessary information for the physicians

to update their representation of the patient case. Ergonomists distinguished two categories of information among collected data: necessary information to sum up the patient case and detailed information to obtain more precisions about the patient. Ergonomists established a table (figure 3) with all collected information categorised by types (i.e. : patient information, patient history…) used by the physician and classified according the two categories previously mentioned (i.e. : the patient constants can be used in the both cases, the physician needs last constants to update his/her representation of the patient case (necessary information to sum up the patient case) and he/she needs to see all constants of the patient stay because he/she wants to have precisions to make a decision (detailed information)). Concerning the software analysis, ergonomists studied mocks-up and textual specifications supplied by designers.

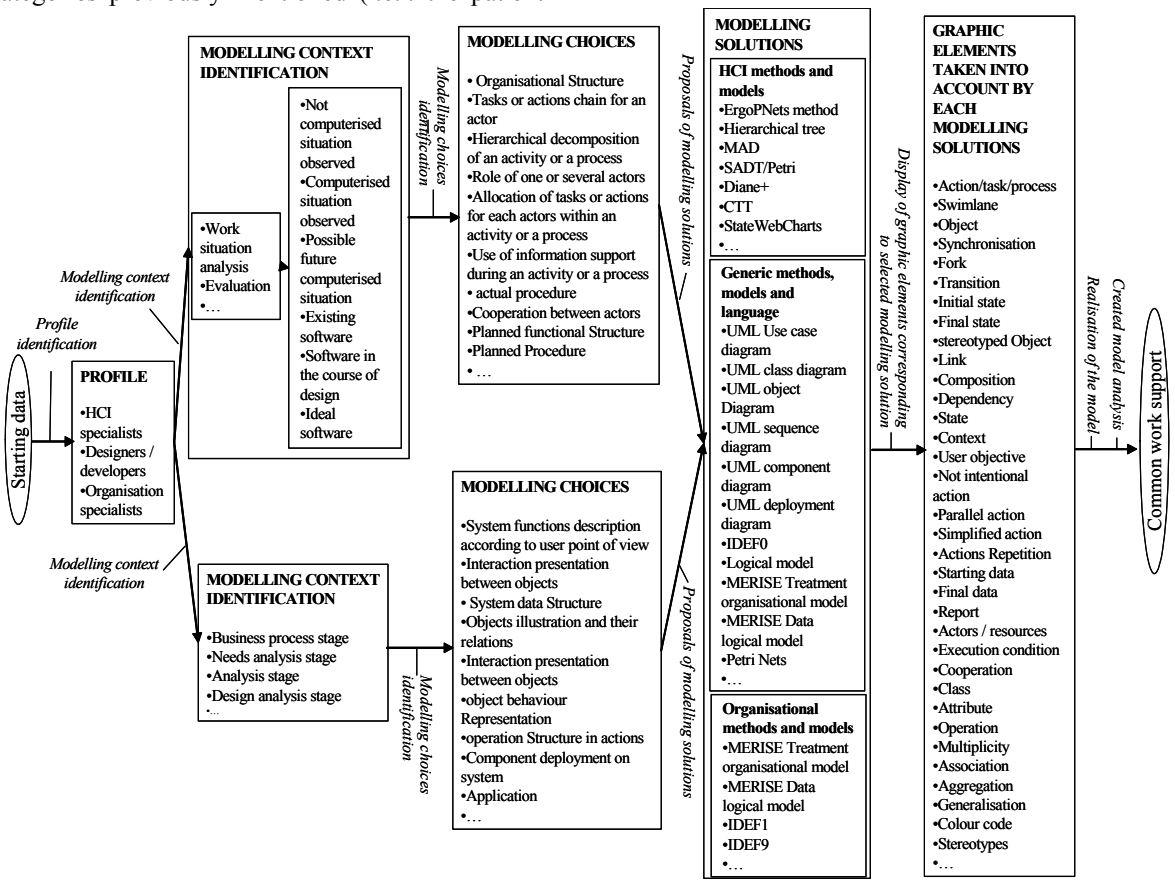



Figure. 2. Dynamic structure to come to a common work support

### 3.3. Approach application to obtain common work support

The ergonomists needed to show clearly the location of information used to the acquisition information of the physician during the ordering activity, planned by the future software, and to show the interactions planned by the future software, to reach these information. The final purpose was to see if the location and the interactions planned by the future software were compatible with physician activity observed. We applied our approach to answer to the ergonomists modelling needs. Referring to the figure 2, the principle is as follows. Starting data are the collected data previously defined in the part « data collected ». The identified profile is "Human-Computer Interaction" profile. The identified modelling context is "evaluation" and particularly "software in the course of design". The modelling choices are "information location used by the actor" and "browsing between screen pages". The modelling solution is the adapted UML State transition diagram. Indeed, we used the "state" element to represent each application screen page or window and to obtain an application view, and the "event" element to represent planned user actions to browse between each screen page. We modified the usual state description. The usual description is a rectangle with the state name, the entry events and exit events which allow to activate the state or to exit the state. The modified state description is a rectangle with the screen page name and the information contained in the screen

page. We added also the condition element from The UML activity diagram to represent conditional event. Finally the last step is the display of the graphic elements corresponding to the modelling solution (adapted UML State transition diagram) is displayed to create the models (i.e.: initial state, final state, event/action, screen page, window, condition, data, and acronym).

| Information categories | Necessary information to sum up the patient case | Detailed information to obtain more |
|---|---|---|
| **Patient Information** | | |
| First name, surname | × | × |
| Room number | × | × |
| Sex | × | × |
| Age | × | × |
| Usual Weight | × | × |
| Address and phone number | | × |
| Family situation | | × |
| Profession | | × |
| … | … | … |
| **Patient History** | | |
| Medical history | | × |
| Chirurgical history | | × |
| Family history | | × |
| … | … | … |

Figure. 3. Extract of table, defined by ergonomists and illustrated collected data

We obtained the model represented in figure 4. The screen pages are represented by a rectangle and the windows by an oval form. The events (or actions) are represented by oriented arrows and show the interaction between screen page and windows. The conditions are represented by a little rhombus. Finally the information location is represented by text located in rectangle and by acronyms corresponding to each categories of information defined by the ergonomists (i.e.: information patient is characterised by the acronym « IP »). To locate the information, one took each information used by the physician and one looked for if this information is represented in a screen page of the future software. If one found it, we wrote the information in the corresponding rectangle and one added the corresponding acronym. One added also the title of each screen page planned by the analysed software in order to the model is more explicit.

### 3.4. Created model analysis

The created model shows that information used by the physician to update his / her representation of the patient case is broken up within the software in the course of design at this moment. The physician must browse between the different screen pages to find necessary and detailed information. For example, to obtain the current patient treatment, the physician must go to the screen page « ordering ». Then if he / she would like to see the last patient constants, he / she must go to another screen page "patient folder" and click on a link to see the constants. While the activity analysis shows that the physician needs firstly for the last information on the patient (i.e.: last constants, last exams planned…). He/she consults in details other information if he/she needs more precisions (i.e.: all constants of the patient stay, medical history…).

The recommendation proposed by the ergonomists was to display a rundown of the patient case and to display links leading to other screen pages with detailed information; it had been represented with our approach by a complete analysis illustrating the possible problem and the recommendation proposed (for lack of place, it is not presented in this article).

## 4. Conclusion and perspectives

In this article, we explained the principles of an approach aiming to help the design or re-engineering of software within complex organisation. The purpose of this approach is to create models which can be used as common work supports between projects partners. It proposes usual modelling solutions (based on existing SE and HCI methods and models and a few very specific methods from ergonomics) to keep designers / developers and ergonomists customary procedures and adapted modelling solutions (based on combinations and adaptations of existing methods and models) to fill in methods and models limits and lacks usually used by ergonomists.

Our perspectives consist in assembling in the same assistance tool modelling solutions necessary to designers / developers and ergonomists, to facilitate information transmission between projects partners and to improve the human factors integration to help software design or re-engineering within complex organisation. This tool is in the course of design. Finally, in this tool, we plan also to propose a means to analyse created models proposing analysis criteria for each modelling solution used by ergonomists.

## 5. Acknowledgment

This research was supported financially by the FEDER, the *NPdC* Region (Miaou & Eucue projects), the Univ. Hospital of Lille, the French Ministry of Research (Rnts & Presc'Info projects).

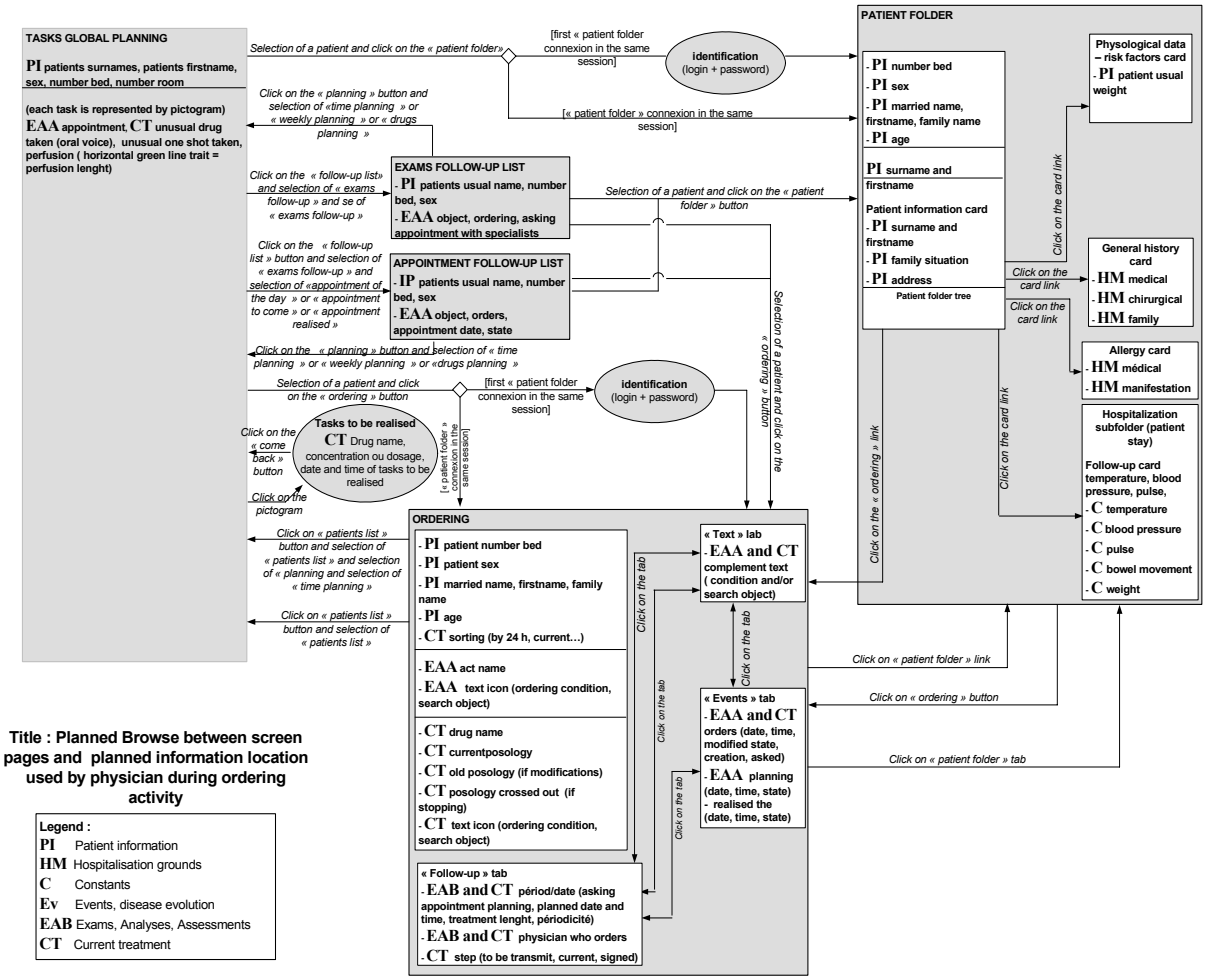



Figure. 4. Example of common work support.

## 6. References


[1] K.J. Vicente, *Cognitive Work Analysis*. Mahwah, NJ: Lawrence Erlbaum Associates, 1999.

[2] R. Kaushal, D.W. Bates, "Information technology and medication safety: what is the benefit?", Qual Saf Health Care, 2002, vol. 11, no. 4, pp. 261-265.

[3] M. C. Beuscart-Zéphir, P.Elkin, S.Pelayo, R.Beuscart, "The Human Factors Engineering approach to biomedical informatics projects : state of the art, results, benefits and challenges", IMIA Yearbook of medical informatics 2007, special issue, to be published.

[4] N.P.Nayak, D.Mrazek, D.R.Smith, "Analyzing and communicating usability data: Now that you have the data what do you do?", ACM SIGCHI Bulletin, 1995, vol. 27, pp. 22-30.

[5] S. Bernonville, C. Kolski, M.C. Beuscart-Zéphir, "Contribution and limits of UML models for task modelling in a complex organizational context: case studyin the healthcare domain", in Proc. the 5th IBIMA Conference, K.S. Soliman (Ed). Internet and Information Technology in Modern Organizations: Challenges & Answers, Cairo, 2005, pp. 119-127.

[6] J. J. Gibson, The Ecological Approach to Visual Perception. LEA, New Jersey, USA, 1979.

[7] D. Ross, "Structured Analysis: A Language for Communicating Ideas," *IEEE Transactions on Software Engineering*, 1977, vol. 3, pp. 16-34.

[8] H. Tardieu, A. Rochfeld, R. Colletti. *La méthode Merise*. Editions d'organisation, Paris, 1983.

[9] J. L. Peterson, *Petri Net Theory and the Modeling of Systems*. Prentice Hall, 1981.

[10] M. Helander et al., Handbook of Human-Computer Interaction, 2nd ed., Elsevier, Amsterdam, 1997.

[11] S. Bernonville, N. Leroy, C.Kolski, M. C Beuscart-Zéphir., « Explicit combination between Petri Nets and ergonomic criteria: basic principles of the ErgoPNets method", in Proc. the 25th EAM Conference, Valenciennes, France, 2006.

[14] D.Scapin, D. Pierret-Golbreich, (1989), "Towards a method for task description: MAD", in *Proc. the Work with display units Conference*, Montréal, Canada, 1989.