



HAL
open science

An HCI-enriched model for supporting human-machine systems design and evaluation

Christophe Kolski, Pierre Loslever

► To cite this version:

Christophe Kolski, Pierre Loslever. An HCI-enriched model for supporting human-machine systems design and evaluation. 7th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design and Evaluation of Man-Machine Systems, Sep 1998, Kyoto, Japan. hal-03411859

HAL Id: hal-03411859

<https://uphf.hal.science/hal-03411859v1>

Submitted on 14 Feb 2024

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.

AN HCI-ENRICHED MODEL FOR SUPPORTING HUMAN-MACHINE SYSTEMS DESIGN AND EVALUATION

Christophe Kolski, Pierre Loslever

LAMIH - URA CNRS 1775
University of Valenciennes, Le Mont Houy - B.P. 311
59304 Valenciennes Cedex, FRANCE
E-mail: kolski@univ-valenciennes.fr

Abstract: The main development models from Software Engineering are too general and largely insufficient in term of interactive systems development. One solution to this problem is the proposal to use so-called HCI-enriched models. Such a model, called ∇ (pronounced "nabla"), is presented in this paper. During a questionnaire-based evaluation, this model has been compared with four classical models (Waterfall, V, Spiral and Incremental models) and five other HCI-enriched models. The first results of this evaluation, led with 43 subjects are discussed in the paper. *Copyright ©1998 IFAC*

Keywords: methodology, human-machine interfaces, evaluation, design, software engineering

1. INTRODUCTION

System development methods (OMT, Merise, UML, SASD...) are, in most cases, based implicitly or explicitly on development models (particularly the Waterfall and Spiral models and their variants, sometimes the V and incremental ones): these models precise globally the different stages to follow for developing the system (see Boehm, 1981 or Sommerville, 1994). For instance, the figures 1 shows the well-known Waterfall model used in hundreds thousands of industrial projects. The main development models from Software Engineering are too general and largely insufficient in terms of interactive systems development, particularly for applications in which human errors inherent in using the human-computer interface(s) can lead to safety problems, and/or ecological and/or economic consequences (Rasmussen, 1986). A solution to this problem is a proposal to use so-called HCI-enriched models (i.e. "Human-Computer Interaction"-enriched models). Such models consist of combining the usual stages found in the main models (project orientation, specification, global design, detailed design, coding,

unitary tests, integration, validation...), with stages concerning more or less explicitly the interactive aspects of design and evaluation.

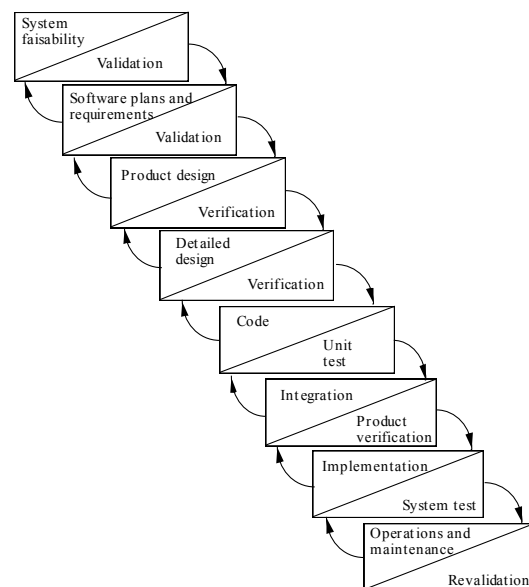


Fig. 1. Waterfall model (Boehm, 1981)

The notion of HCI-enriched model is not new. Several discussions and ideas concerning this notion exist in currently available literature (see for instance Hartson and Hix, 1989 ; Bass and Coutaz, 1991 ; Nielsen, 1992 ; Curtis and Hefley, 1994 or Collins, 1995). But no real and new model considered as a "Software Engineering model" has really been proposed. In this paper, a model called ∇ (pronounced "nabla") model, is presented. A questionnaire-based comparison of this model with four classical models and five other HCI-enriched models will also be explained. The first results are discussed.

2. GLOBAL PRESENTATION OF THE ∇ MODEL

The HCI-enriched ∇ model has been proposed by Kolski (1997 ; 1998). It concerns interactive system development. The first version is shown in figure 2. Its look is inspired by the well-known V model. Its objective is to locate the steps necessary to develop an interactive system, by distinguishing the HCI strictly speaking (left part of the model) from the application module(s) eventually accessible from the HCI (right part). One of its outstanding characteristics is to locate stages -that do not exist in the standard models - where the human factors have to be considered by the development team.

The first step is quite common in Software Engineering, and marks the beginning of the project by giving an orientation to the work, ie, realising objectives, project organization, constraints, and so on. Then the model emphasizes the importance of the analysis of the entire human-machine system during the project; this analysis deals more particularly with the system, the human tasks and the user(s). Modelling must be slanted towards:

- A real model corresponding to the current (existing or virtual) human-machine system, with its constraints, its strengths and weaknesses. Three cases can be considered. When the task consists of studying an already existing human-machine system to ultimately end with a new one, the modelling is of course carried out from the existing system. When the task consists of creating a new human-machine system from other already existing systems, the modelling is based on a synthesis of the data issued from each analysis. When there is no previously existing human-machine system and when the system is to be designed entirely from scratch, this model needs to be designed (Stammers *et al.*, 1990).
- A reference model corresponding to those of a human-machine system considered as ideal, by considering all the points of view and requirements

of the different human partners concerned with the planned human-machine system. This model must in particular list a set of criteria which must be abided by. The nature of these criteria can be extremely wide-ranging (safety of human beings, of the facilities, of the environment, production, software ergonomics, energy-savings...), following the considered application field.

By comparing progressively the two models during the analysis of the human-machine system, and by reaching compromises aimed at satisfying a maximum of criteria, the data must be sufficiently relevant for the specification of an interactive system. This system is, in turn, adapted to the users' informational requirements, and to the requirements in terms of cooperation modes between the user(s) and the application modules. Then, the task consists of specifying the HCI on the one hand, and specifying the identified application modules on the other hand. This set of specifications will have to be evaluated and validated from a socio-ergonomic point of view, so as to verify the relevance of the new solutions being integrated into the targeted human-machine system; indeed, in most cases this includes several human beings, inter-connected software and hardware packages. We will have to consider the collective aspects of the work, aspects which are generally neglected by development teams (Rasmussen *et al.*, 1991; Zorola *et al.*, 1995).

After the specification of the HCI and application modules, and in order to reach the coding stage, the preliminary and detailed design stages, respectively associated links in the V model with integration tests and unitary tests, are carried out in the usual way. In relation to the design stages, it is important to ergonomically evaluate and validate the components of the interactive system. With regard to this remark, it is necessary to note the importance of evaluation methods; some of them are useable at this level. Many evaluation methods already exist in current literature (see for instance Wilson and Corlett, 1996; Kolski, 1997). Note also that, as indicated in the ∇ model, the chaining-process between the stages is propitious to a prototyping approach (Boehm *et al.*, 1984; Lichter *et al.*, 1994).

As is the case in each existing model, the acknowledgment stage has been located. In order to insist on the problematic approach of an interactive system, we have chosen to split this stage up by symbolically distinguishing it into an HCI-oriented acknowledgment and an application modules-oriented one. These two stages should be minimized if the complete interactive system is to conform to the data issued from the modelling of the human-machine system, and if each solution has been effectively evaluated and validated.

Finally this cycle ends with another stage, quite common in Software Engineering: the exploitation and maintenance stage. Note that this stage could return to the orientation stage, in a loop. Indeed this could aim at the evaluation of the interactive system being used, together with improvement. Thus, our approach would consist of proceeding towards an

analysis of the human-machine system, leading to new specifications concerning the HCI and/or the application modules. Should these specifications be validated, the rest of the cycle could be viewed among the principles described previously. In conclusion, this development model could also be considered an evaluation and improvement model.

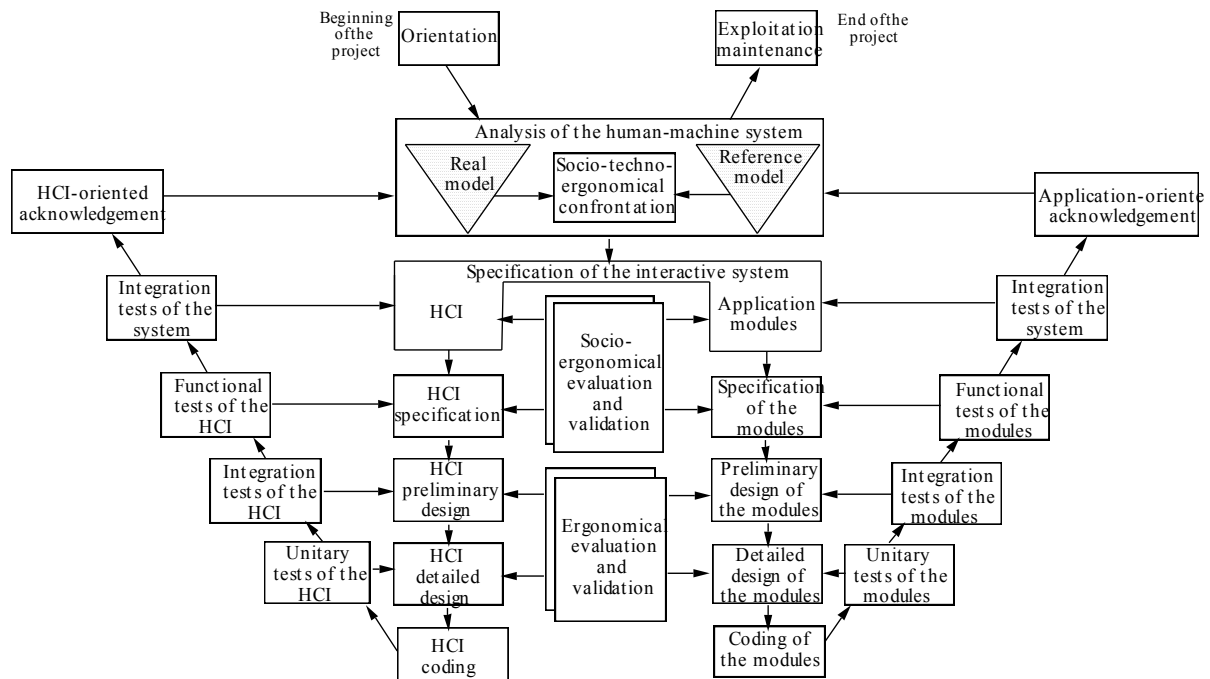


Fig. 2. ▽ model (Kolski, 1997; 1998)

Contrary to many of the main existing development models, the ▽ model has advantages, such as:

- importance being given to the human-machine system analysis and design,
- separation of interactive aspects from the application,
- importance being given to the system evaluation.

Such a model must provide interactive system developers with new ideas, and incite the developers to take into account human factors. This is why a questionnaire-based evaluation has been made, with the goal of comparing its impact with those provided by the classical models, and also by other HCI-enriched models.

3. QUESTIONNAIRE-BASED EVALUATION

Specific questionnaires have been given to 43 subjects. These subjects were academic students (with french diploma globally equivalent to Masters Degree), with primary experience in industrial and/or academic projects. These students came from three education types : (1) 6 ergonomists (5 academic years), (2) 15 computer scientists, with a specialization in human-computer interaction (5

academic years), (3) 22 computer scientists, without specialization (4 academic years). The subjects knew well the stages associated with the Waterfall, Spiral and V models; therefore they were able to understand each model.

For each question in the questionnaire, the subject had to answer with a value (if the answer was given without hesitation) or an interval in the contrary case (see figure 3). The subject could also comment on the answer given.

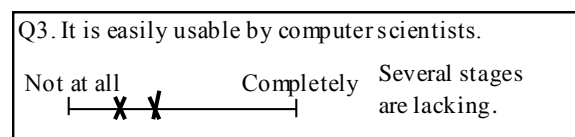


Fig. 3. Example of subject's answer

The questionnaire begins with a preliminary question: "QP : You feel concerned by the taking into account of human factors (requirements analysis, analysis of the human tasks the users have to perform, analysis of the users' characteristics, software ergonomics). It is a part of your job."

Ten models were successively displayed in the questionnaire without explanation (because a model must be self-explanatory). These consisted of four classical models (Spiral, Waterfall, V and Incremental models) and six HCI-enriched models: Producer model (James, 1991), the models of Valentin *et al.* (1993), Collins (1995), Curtis and Hefley (1994), the Star model (Hartson and Hix, 1989) and the ∇ (nabla) model. As examples, the model explained by Collins in his book is shown in figure 4, the model proposed by Curtis and Hefley in a paper is shown in figure 5; the Star model is shown in figure 6. Due to a lack of space the other models are not shown, but they can be easily found in currently available literature.

For each model, the subject had to answer the following questions:

- "Q1. This model takes well into account the stages in classical Software Engineering.
- Q2. It is sufficiently explicit concerning the development of the software interactive part.
- Q3. It can be used easily by computer scientists.
- Q4. It can be used easily by ergonomists.
- Q5. It can be used by mixed teams composed of computer scientists and ergonomists.
- Q6. It encourages teams to take into account human factors, and to integrate design ergonomics as early as in the first stages (specification and design stages).
- Q7. It encourages teams to take into account the human factors. However the ergonomics can intervene rather after the software realization, for correction and improvement.
- Q8. It allows a prototyping approach.
- Q9. It insists on the stage of analysis and modelling of the complete human-machine system in which the human-machine interface is envisaged.
- Q10. It allows identification of the difference between the HCI development and the application development.
- Q11. It simplifies the conduct of the project when the target software must be interactive; therefore it is useful to the project conductor.
- Q12. It must be taught during software Engineering lectures.
- Q13. It must be taught during the lectures concerning interactive software design and evaluation."

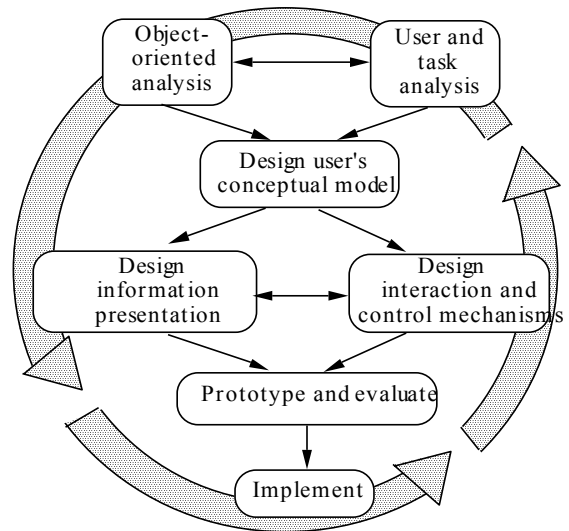


Fig. 4. Collins' model (1995)

USER INTERFACE ENGINEERING	DEVELOPMENT PHASE	SOFTWARE ENGINEERING
User & task analysis	Requirement analysis	Application design
Human vs. machine	Requirement allocation	Hardware vs. software
Dialogue design	Preliminary design	Architectural design
Screen design	Detailed design	Logical design
Coding	Implementation	Coding
Usability lab	Implementation testing	Unit % Integrating test
Contextual observation	System testing	System testing
Human performance	Optimization	Machine performance

Fig. 5. Curtis and Hefley' model (1994)

All the data were analysed. Initial results are given below.

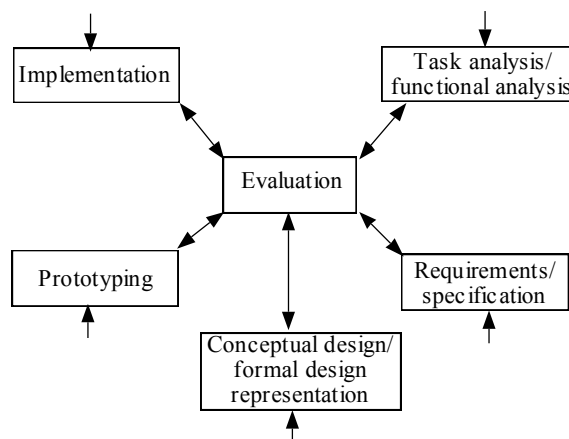


Fig. 6. Star life cycle (Hartson and Hix, 1989)

4. INITIAL RESULTS

With the presence of a multivariate aspect and responses through intervals, the statistical analysis is initially exploratory and global.

When observing the data roughly, one point becomes evident: only 10% of the responses are given with an interval; in fact few individuals do answer using an interval. Moreover, the use of intervals decreases with time (intervals are given for the first questions only). Thus, for this present and global analysis, the interval width is not considered (for the 10% of answers with a non-zero width, the values are summarized through the mean). A second point is that approximately 75% of individuals use the right part of the segment (from 2.5 to 5 cm) with the preliminary question i.e. they feel rather involved by the human factors in software design.

The next stage of the exploratory analysis aims to show the most discriminant questions, their relationships and the influence of the model and the education type. To achieve this aim, the principal component analysis (PCA) is used. The data are considered through a table where the columns correspond to the 13 questions and the rows the 430 combinations (s,i,m), *s* indicating the education type (*s*=1 (ergonomist), *s*=2 (computer scientist specialized in human-computer interaction) or *s*=3

(computer scientist)), *i* the individual within the education type and *m* the model (*m*= Waterfall, V, ..., nabl).

PCA yields two primary axes with very high relative inertia compared with the next axes (46% and 12% vs. 7%, 6%, etc.). Thus we only focus on the factor plane results (58%). The variables with the highest contribution to the position of the first axis are respectively Q13, Q2 and Q11. The first axis highlights rather high correlations between all 13 questions. Observations of the 430 combinations cloud showed high intra and inter-individual differences. Nevertheless, it was possible to see, roughly, subclouds according to some model levels. To assess the model factor effect on the questions, the 10 averaged profiles were computed and considered with a passive status in the PCA (they are projected as illustrative rows on the first factor plane without participating in the positioning of the axes). Such a procedure shows that models are organized into a hierarchy, figure 6; the highest (the best) responses being on the right side of the first factor axis and the lowest on the left side.

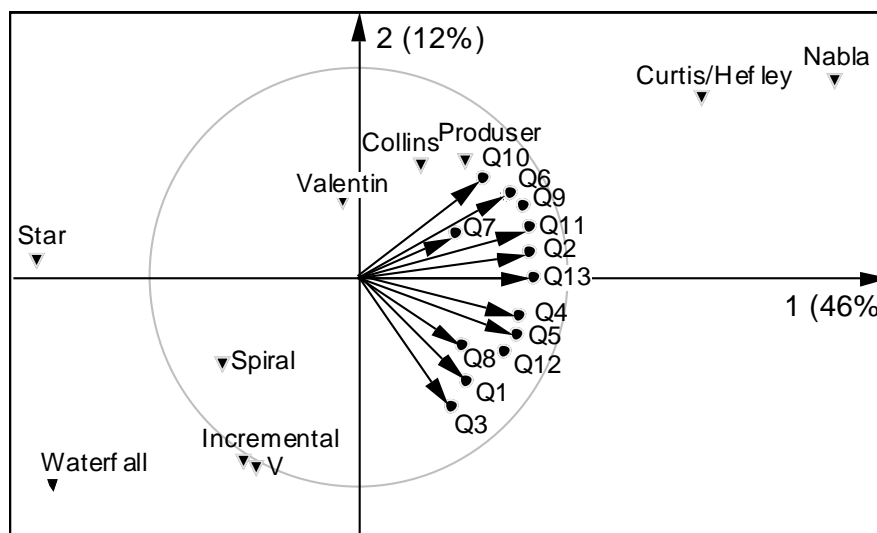


Fig. 7. Principal Component Analysis (PCA) of the questionnaire. Are represented on the first factor plane the 13 questions and 10 averaged points corresponding to the 10 models (from an integration over individuals and education types).

To go further with the questionnaire analysis, a confirmatory statistical approach based on the one way analysis of variance can be used. Most results are confirmed and more particularly the high influence of the model factor onto the most discriminant questions (Q13, Q2 and Q11) and a poor effect of the education type factor. For instance, the box plots of figures 7 and 8 show the influence of the model type factor and the absence of influence of the education type factor on question 13.

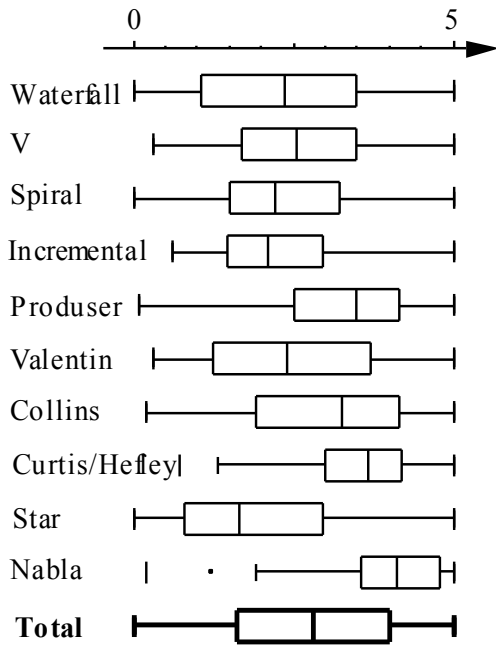


Fig. 8. Box plots of question 13 for the ten models

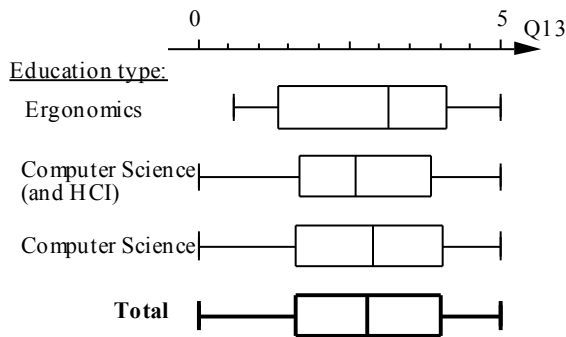


Fig. 9. Box plots of question 13 for the three education types

This first analysis shows clearly that the subjects can differentiate between the models. For interactive system development, the two most preferred are the ∇ (nabla) model and that of Curtis and Hefley, which are both HCI-enriched models. There is a high coherence between all the questions. The most discriminant one is question 13 (about the necessity to teach the model during lectures concerning interactive software design and evaluation).

5. CONCLUSION AND PERSPECTIVES

The HCI-enriched ∇ model has been explained in the paper. A first questionnaire-based evaluation shows a global positive impact with potential developers : computer scientists (specialized or not in human-machine interaction), as well as ergonomists. The statistical analysis can be enriched when studying more carefully the interval width and the connections between the variables.

This model could become a new theoretical and methodological framework for those researchers, concerned with the development of interactive systems.

Several research perspectives can also be envisaged. The most important of these would be validation of the model during real and/or simulated industrial projects and refining it over the next few years. Such validation is currently planned. Another perspective would be to improve existing development methods (SASD, UML, MERISE, and so on) by basing them on the stages proposed by the ∇ model.

ACKNOWLEDGEMENTS

The authors thank Joe Galway for considerable assistance in the translation of this paper.

REFERENCES

- Bass, L., and Coutaz, J. (1991). *Developing software for the user interface*. Addison-Wesley.
- Boehm, B.W. (1981). *Software Engineering Economics*. Englewoods Cliffs N.J. Prentice Hall.
- Boehm, B.W., Gray, T.E, and Seewaldt, T. (1984). Prototyping versus specifying: a multiproject experiment. *IEEE transactions on Software Engineering*, 10 (3), May.
- Collins, D. (1995). *Designing object-oriented user interfaces*. The Benjamin/Cummings Publishing Company, Inc, Redwood City.
- Curtis, B., and Hefley, B. (1994). A WIMP no more, the maturing of user interface engineering. *Interactions*, January, pp. 22-34.
- Hartson, H.R., and Hix, D. (1989). Towards empirically derived methodologies and tools for human-computer interface development. *International Journal of Man-Machine Studies*, 31, pp. 477-494.
- James, M.G. (1991). PRODUSER: PROcess for Developing USER Interfaces. In *Taking Software Design Seriously*, J. Karat (Ed.), Academic Press.
- Kolski, C. (1997). *Interfaces homme-machine, application aux systèmes industriels complexes*. Editions HERMES, Paris, january.
- Kolski, C. (1998). A call for answers around the proposition of an HCI-enriched model. *ACM SIGSOFT Software Engineering Notes*, 3, pp. 93-96.
- Nielsen, J. (1993). *Usability engineering*. Academic Press.
- Lichter, H., Schneider-Hufschmidt, M., and Zullighoven, H. (1994). Prototyping in industrial software projects, bridging the gap between theory and practice. *IEEE Transactions on Software Engineering*, 20 (11), pp. 825-832, november.

- Rasmussen, J. (1986). *Information processing and human-machine interaction, an approach to cognitive engineering*. Elsevier Science Publishing.
- Rasmussen, J., Brehmer, B., and Leplat, J. (Eds.) (1991). *Distributed decision making*. Chichester, J. Wiley.
- Sommerville, I. (1994). *Software engineering*. Fourth edition, Addison-Wesley.
- Stammers, R. B., Carey M. S., and Astley, J. A. (1990). Task analysis. In *Evaluation of human work. A Practical Ergonomics Methodology*, Wilson J. R. and Corlett E. N. (Eds.), Taylor & Francis, London.
- Valentin, A., Vallery, G., and Lucongsang, R. (1993). *L'évaluation ergonomique des logiciels, une démarche itérative de conception*. Montrouge, ANACT.
- Wilson, J.R. and Corlett, E.N. (eds.) (1996) *Evaluation of human works : a practical ergonomics methodology*, 2nd edition. Taylor & Francis, London.
- Zorola-Villarreal, R., Pavard, B., and Bastide, R. (1995). SIM-COOP: A tool to analyse and predict cooperation in complex environments, a case study: the introduction of a datalink between controllers and pilots. *Proceedings Fifth International Conference on Human-Machine Interaction and artificial Intelligence in Aerospace, HMI-AI-AS 95*, Toulouse, France, september.