Rationalisation of decision-making processes in design teams with a new formalism of Design Rationale

Lewkowicz Myriam, Zacklad Manuel

Tech-CICO

Université de Technologie de Troyes

12, rue Marie Curie BP 2060

10010 Troyes Cedex France

myriam.lewkowicz@univ-troyes.fr

manuel.zacklad@univ-troyes.fr

Abstract:

More and more frequently, the organisation of design fits into a project organisation where different designers have to cooperate with flexibility and reactivity. In order to help this cooperative design processes, we have to respond to new types of needs: a little formalised coordination that requires permanent mutual adjustment, the fact that members of the team are geographically distant, the difficulty to build a shared referential with design documents and technical and organisational decisions that structure the project. In order to meet these needs, asynchronous GroupWare is often recommended. However, these tools' efficiency depends on the models that structure the work processes and the knowledge exchanged. In this paper, we present ABRICo, a new Design Rationale formalism that permits the memorisation of cooperative decision-making processes in complex design situations. We will explain next how we intend to use this formalism to specify cooperative design tools usable via an Intranet for computing projects.

Keywords:

Design Rationale, Organisational memory, Knowledge Management, GroupWare

INTRODUCTION

Design activities are more and more frequently integrated in projects, which are transversal structures where all the required competencies are grouped to ensure progress of the project (Hatchuel, 1994). Designers' teams dedicated to the projects are built for a limited period. They are made of people who come from different areas, from other projects or even from outside the firm, and they can be located in different places. For the success of the project, it is necessary for these teams to have efficient tools for the sharing, the circulation and the capitalisation of information. The problematics that we are looking at here concerns the specification of a knowledge management tool for a mediated asynchronous cooperative design. The development of this kind of tool, named GroupWare (Ellis, 1991) benefits from CSCW studies (Bannon, 1991, Palmer, 1994, Schmidt, 1994)

In the projects that we have studied, we have brought to the fore the need of capitalisation and sharing of tacit knowledge used by the designers during the project in order to attain good coordination. We consider that a big part of this tacit knowledge corresponds to the know-how which allows people with differing expertise to coordinate themselves in the project during all the discussions and meetings. Capitalisation of knowledge used in cooperative design is the concern of Design Rationale field that provides methods for the formalisation of decision-making processes that occur in this context and in a general way of organisational learning.

Actually, knowledge and its management have became a concern because it is the key to competitiveness since it depends on an adaptive and innovative flexibility (Hatchuel, 1994).

For Nonaka (Nonaka, 1994), organisational learning depends on conversion between two forms of knowledge: tacit that is hard to formalise (know-how for example) and explicit that can be codified, formalised. Lastly, for Argyris and Schön (Argyris, 1978), as for Levitt and March (Levitt, 1988) and Nonaka (op.cit.), one can really talk about learning when the results are registered in the organisation memory and not only in the individual one.

For these two types of knowledge it would be desirable for the projects to have assistance systems for capitalisation. Tools dedicated to Electronic Management of Documents (EMD), widely used particularly for attestation of quality processes, answer the need for explicit knowledge memorisation. On the other hand, there still remains the of tacit knowledge representation that will be possible of course only at the end of a progressive process of explication.

If efficient tacit knowledge in the field of organisational learning has to be collective, we can rightfully think that meetings and discussions form favoured times where we could see this tacit knowledge coming to the fore and beginning to be explicated. This could explain the efficiency of Design Rationale methods(Moran, 1996) that, with the formalism that they suggest, catalyse the explication of the knowledge shared by the designers. This hypothesis will guide us through the specification of the tool that will help knowledge management tool that we want to develop to answer the needs of the designers' team which we will present below .

We are first going to present that design situations where one has to memorise a choice between different features of an artefact (Grudin, 1996) are *specific* and that there are other kinds of design situations that we shall call *complex* and for which classic Design Rationale methods cannot be used.

WHAT IS DESIGN RATIONALE ?

Generally

"Design Rationale" is used in many different senses, and no one of them is standard in this stage of research on the topic. We can quote (Moran, 1996) six senses:

- The expression of the reasons for the design of an artefact.
- The logical reasons given to justify a design artefact.
- A representation for recording the logical reasons for a design artefact.
- A method for designing, in which a DR notation is used as a process-facilitation tool for guiding the design process.

- Documentation of the reasons for the design of an artefact, the stages of the design process, and the history of the design and its context.
- An explanation of why a designed artefact (or some feature of an artefact) is the way it is.

Most of the current research is concerned with the representation of rationale, both its form and content. The aim is to produce a representation that can help others understand why the resulting design is the way it is, and to develop ways of reusing components of the design space to support further design and re-design. The goal is to find the most realistic and the least expensive way to memorise the reasoning, but it has to be structured and clear (Mac Lean, 1989). Therefore, things that have been memorised will be understandable and usable by someone who did not design the artefact and is trying to understand this artefact. Inasmuch as the main activity in meetings is the formulation and criticism of arguments, the Design Rationale goal is to develop schematic representations that will enable computers to support creation, evaluation, and modification of arguments. The main hypothesis is that when one makes the structure of the arguments explicit, the arguments will be built and explained more rigorously. Several methods have been developed. The difference between these methods is the formalism chosen to represent the arguments, the types of arguments that will be memorised, and the detail level it will be possible to obtain. We can quote DRL (Design Representation Language) (Lee, 1996), IBIS (Issue-Based Information Systems) (Yakemovic, 1993), and QOC (Questions, Options and Criteria)

(Mac Lean, 1996). We are now going to describe the last one, developed at the Rank Xerox EuroPARC (Cambridge). It enables one to represent, in a graphic design space, different solutions with their advantages and drawbacks.

QOC's example

QOC supports the representation of the design space around an artefact being produced. This design space is an explicit representation of alternative design options, and the reasons for choosing among those options. The figure below shows Questions that highlight key issues in the design space. Options can be thought of as "answers" to Questions. Criteria are the reasons that argue for or against possible Options. We represent relationships between Options and Criteria as relatively positive (solid line between Option and Criterion), or negative (dotted line). Options may spawn off consequent Questions which allow more detailed aspects of the design to be addressed.

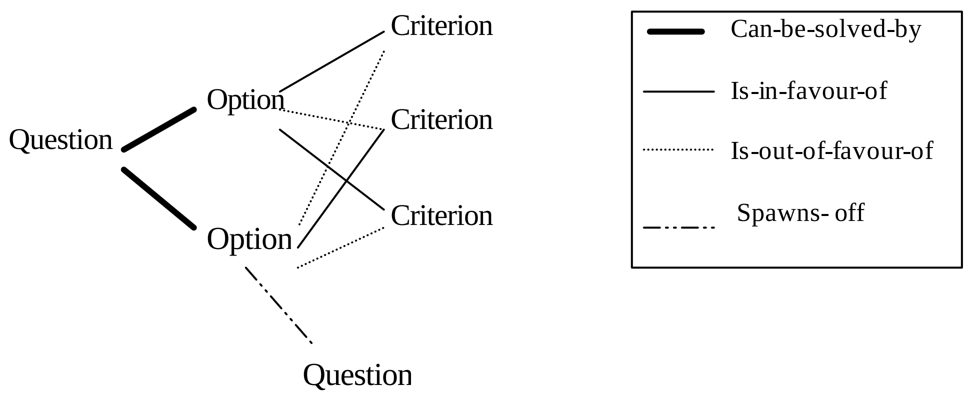

Figure 1 : QOC representation

Experimentation with the QOC method has been generally conclusive: Some researchers have successfully tested the capacity for the notation to represent a design process and used QOC as a technique for understanding designs through analysis and for characterizing some aspects of design activity (Mac Lean, 1996). In (Karsenty, 1996), the author presents the positive results of an empirical evaluation of DR documents (which were constructed using QOC), carried out with six experienced professional designers who were asked to understand and to assess a past design. The three studies presented in (Schum, 1996-b) and the two in (Schum, 1996-a).are interested in QOC structure as a cognitive tool to support individual and collaborative design in real time. The conclusion is that QOC provides most support when designers need to elaborate poorly understood design spaces in order to clarify the key Questions, Options and Criteria. However, certain difficulties have been identified: In (Schum, 1996-a), we can read difficulties encountered with QOC constructs: Firstly, Questions can be used by posing an extremely general Question or through a long series of Questions, each of them addresses the problem of the current iteration. Whereas, on the one hand, a very general Question offers no insight to someone else trying to understand the design, on the other, many different Questions increase too greatly the number of schemas, and they become unusable. Secondly, discrete Options were sometimes impossible to identify because the design of the final structure was treated as the evolution of one Option over time; the difference between each version was only a single, or a few fine details. Finally, designers sometimes had trouble precisely justifying their preferences for an Option. In the final analysis, the only

Criterion they were able to identify was "did it work ?" The difficulty is that for the purposes of QOC, this needs to be reexpressed as more focused Criteria which build a bridge from that goal to specific design features. In the studies presented in (Schum, 1996-b), designers found that sometimes, QOC was intrusive when they were trying to evaluate relatively well understood design spaces in which the main task was to satisfy multiple constraints through iterative testing and adjustment, until an optimal solution has been obtained.

We identified the main characteristics of situations for which it is possible to use existing Design Rationale formalisms:

- If the decision is the result of a choice between different alternatives that have been proposed in the meetings. Take an interface for example: there could be a choice between a narrow or a wide scroll bar, continuously appearing or not.
- If the time available to elaborate solutions is limited: punctual questions are raised and designers answer them in one or two meetings. Hence this time parameter seems inessential and does not appear in classic Design Rationale formalisms.

The actors' role in meetings is an element *not* taken into account in existing formalisms, because in the course of discussion these actors do not necessarily call upon an expertise that would be their own.

But all design situations do not respond to these characteristics. We have identified some situations that we shall call complex whose characteristics are:

- The final decision is constructed progressively and does not result from a choice between different alternatives.
- The time available to elaborate solutions is extended: it may be several months, and time is then a parameter to be taken into account.
- The actors' roles are important because their competencies are not interchangeable. Each actor as it were is an expert.

For these situations, it is impossible to use a classic Design Rationale formalism, primarily because these methods do not allow one to represent a solution's progressive evolution during meetings, but only the choice of one alternative among many.

Finally, when several alternatives may be proposed, classic DR formalisms like QOC are workable and are a useful representation of classic decision-making methods such as choosing or sorting out[1]. In this case, we could object on the one hand that the time over which propositions are elaborated and the order of arguments and criteria formulation do not appear, and on the other hand, that the roles of the actors who bring up issues are not mentioned.

[1] in (Hoc, 1989), we can read that these situations where one explore simultaneously several solutions (in-breadth-first research), are very costly in terms of work memory.

We are now going to first identify what complex situations are, and then we shall propose a way to formalise the discussions that lead to decision-making in these cases.

WHAT ARE COMPLEX DESIGN SITUATIONS ?

Uncertainty

As Hoc has pointed out (Hoc, 1989), problem-solving strategies are different according to knowledge and detail levels. In large design projects, designers face great uncertainty about the features of the solution they will adopt. The knowledge level is low at the beginning of the project. In order to reduce this uncertainty, designers adopt "depth-first" strategies, which lead them to explore the details of a solution, even if it means throwing the whole solution back into question, contrary to "in-breadth-first" design, where they examine several solutions' sketches among which they have to choose. In these complex situations, it is necessary to represent a solution's evolution over the course of a project by virtue of the increased knowledge level, and not the sorting out or the choice among several solutions whose features are known (Giard, 1996)[3].

[3] « one can say that the difference between a production process and a project process is that the aim of the first one is anticipated at the beginning of the process, when one have to involve in the second one to know if it could be brought to completion, and what will be the conclusion. Let's stress here on the

Time

As we have already stressed in introduction, complex design situations last for long periods. Then, it's necessary for the rationalisation of the process, to represent the progressive evolution of the solution in dating the different versions in order to restore the "phylogenesis". In some cases, the new propositions are diametrically opposed to the previous ones. In other cases, in contrary, they are a combination of different features of the previous propositions. Because the classical DR formalisms stress the logical relationships between the arguments, they put the elements of the discussion in an "a-temporal" space. So, they do not permit to catch the project dynamics. By introducing this dynamic dimension, we hope that we will give a better comprehension of the solution's evolution which will improve the ability to revise afterwards the solution. So, if some constraints[4] on the project are modified, it will be possible to identify when these constraints influenced the design, and we will be able to elaborate a new solution starting at this point, without having to reexamine elements of previous decisions.

---

fact that there is also uncertainty about the goal to achieve. The assertion that there is a priori a target does not mean that this target could be precisely and surely defined at the beginning.» (Giard, 1996)

[4] See the graph of the dynamical evolution of knowledge and freedom degrees of a project in (Midler, 1996)

Roles

Design Rationale's aim is to represent lines of argument that lead to decision-making in design situations. Thus, in complex situations, the actors can play quite different roles. Take a project meeting for example: we may have the owners who finance the project, the contractors who carry out the project, and among them a project manager, some designers, a quality manager... This diversity has an influence on the progress of design processes. Identifying the role of the actor who formulates an argument enables one to better comprehend the discussion because it allows a new criterion to appear that was not formulated in the meeting. This criterion is the influence of the source formulating an opinion. Indeed, an argument does not have the same weight, or even the same significance, if it is formulated by the project manager, the quality manager, or a designer for example. Classic formalisms of Design Rationale cannot account for one important element of complex design situations which is the roles' *diversity* because roles do not appear in these formalisms. So this will be an element to take into account for a rationalisation of the decision-making processes in complex collective design situations.

HOW TO ACCOUNT FOR COMPLEXITY, TIME AND ROLES : THE ABRICo SUGGESTION

When we analyse the meetings where it was impossible to use QOC, we find that they are made up of progress reports by the supplier and a control and/or a validation by the client. The decisions made are more the result of interactions between different actors in one or more meetings than the result of a sorting out or a choice of a solution after research initiated by a precise demand. In the meetings at which we were present, we have identified a goal to achieve, each party interpreting this goal, identifying functions to implement in order to achieve the goal, and seeking how to implement these functions. We shall now define these concepts and build the models that will describe decision-making processes with these concepts.

Common goal: the goal expresses a need to be satisfied. It can be identified by virtue of part of the contract: in this case it is equivalent to an objective to be achieved by one party. The goal can also be the result of an agreement between parties: for example the use of a particular method to solve a problem. This is a common element among the parties; it is the beginning of a work process in which the following categories will be found. A goal might be for example "to elaborate a procedure of needs analysis for producing and archiving documents".

One party's interpretation : the interpretation expresses the way in which an actor or a group of actors appropriates the goal and thinks about ways to attain it. It is a function to be implemented in order to achieve the goal. Each party has its own interpretation of the goal, and that's what will lead to discussion. An interpretation might be for example "analysing work processes".

One party's proposition : it is a proposition on how to implement the interpretation. Said interpretation may be that of the party who expresses the proposition or that of another party. In the latter case, the proposition is a reaction that responds to a solicitation from the other party. The proposition consists in showing how it is possible to achieve the functions that allow one to reach the goal. For example, a proposition might be a presentation of schemas.

Agreement between the parties : agreement is when a consensus on the proposition(s) is expressed with regard to the interpretation. Hence it is a common element among all the parties, which may occur in the course of the work process and which represents a decision when it closes the process. For example, an agreement might be the acceptance of certain schemas or the decision to use a particular tool.

Starting with the analysis categories presented above, we constructed a static model that identifies the types of links that exist between categories and a dynamic model that shows how they follow on from each other in the course of one or more meetings.

These models might represent two situations we encountered in the meetings and that seem to be ones generally encountered where collective work is involved. However we do not claim to be exhaustive and we could be led to enrich our modelling if we encounter a new situation. We have named this modelling ABRICo for the French words Accord (agreement), But (goal), pRoposition, Interpretation en Conception (during design).

In the static model, we show the relationships between the analysis categories, i.e. between goal, interpretation and proposition but also the relationships that might exist between two interpretations or two propositions. This model allows us to complete the definitions of the concepts given above (see figure 2). Applying this model to a decision-making process will allow better comprehension of the structure of this process.

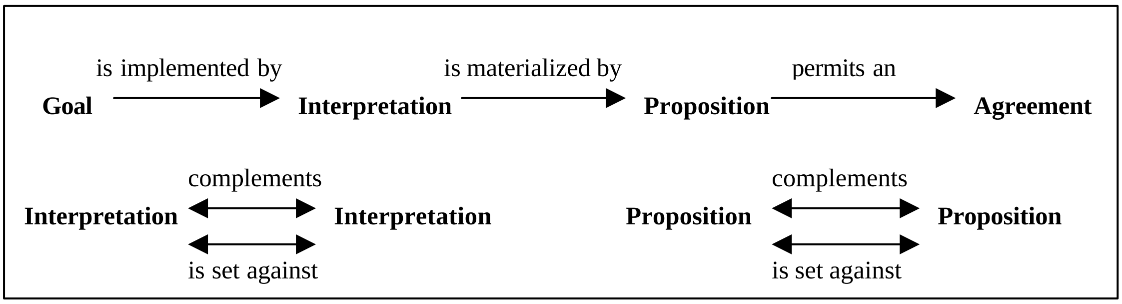

Figure 2: static model ABRICo

The dynamic model shows how the analysis categories are connected over time and thus follow on from each other. The decision-making process begins with the definition of a common goal, followed by a party's interpretation and its proposition, a proposition that leads to another party's interpretation.

We are now going to present an example of a complex design situation, and how we represent the design process with ABRICo.

AN EXAMPLE OF A COMPLEX DESIGN SITUATION

A needs analysis for a designers team working in the Information System department of a big firm has been run in the context of an exploratory project taking aim at providing a cooperative work tools for teams localised in different places. Among the teams chose to belong to this project, the one that we studied was composed of about fifteen designers. They were coming from the inside of the firm or were subcontractors recently recruited, and they were managed by two people, one with an administrative and relational management role and the other coordinating current design studies and checking their contents. The designers' role consisted in analysing what had to be implemented to answer the needs identified by the users and the owner. So they wrote studies notes, which could have some impact on the data model, the interface, the system's functionalities or the management rules. This project was planned to have two or three years' duration.

The team's needs that we have brought to the fore are related to following topics: formalisation of work processes, information classification, asynchronous communication from afar, nomadism, scheduling, and task management. We will not study here topics related to synchronous work and nomadism, in order to focus on needs related to the formalisation of work processes and information management. More precisely, designers complained about following aspects:

- Lack of a system of reference for documents.
- Lack of functional documentary (that is to say explaining the main functions, which the application program has to answer).
- Problems for accessing to actualised information.
- Lack of coordination leading to double tasks.
- Lack of criteria for efficiency evaluation of studies notes.
- Lacks of clarity in roles' distribution between the different departments taking part in the project and inside of the project.

If, as we previously said in introduction, if some of these needs can be satisfied by traditional EMD tools, others refer to problems of formalisation of collective design processes, whether it's the matter of roles, of validation criteria of a solution or of the memorisation of past design choices.

These needs, whether they come from a lack of formalisation as much in the activity coordination than in the work methods used, attest an insufficient degree of knowledge explication in the project. In accordance with what we said in previous chapters, these needs seem to be the concern of Design Rationale solutions, but not classical ones. We can actually identify the characteristics of complex design situations : (1) the solutions are not the result of a sort or a choice between several options but they are built step by step during long periods, (2) there are great time constraints because the development team is waiting for the designers studies to start the development, and finally, (3) the exchanged arguments have not the same weight or the same signification depending on the actor's role who formulate the (designer, project manager, quality engineer, …). We are then thinking that this team is confronted to complex collective design situations and that it could be more judicious to use ABRICo to specify a tool dedicated to this team which could assist knowledge management.

We are now going to present first the dynamic ABRICo representation (figure 4) to permit a better comprehension, and then we shall show the static one (figure 5). This example fits in the context of France Télécom launching new offers of service on the market: network services (call signal, conversation between three people…) and tariff services (several discount formulas). For marketing strategy and management reasons, Information System designers have constructed a modelling that considers all of these services as products in their own right.

In the new version of the application program that permits to take an order, the interface then proposes the access to these products by the *Product* item in the menu (cf. figure 3). Unfortunately, these services are rather perceived as intrinsic features of the rental than as products on their own rights by the users of the application program in the agencies. This representation is strengthened by the fact that, at the creation of a new rental, users are able to directly keyboard the most wanted products/services (cf. figure 3). Because of these divergent representations between designers and users, the latter do not search the products/services in the good item in the menu, that is what causes a bad use of the application program and multiple errors.

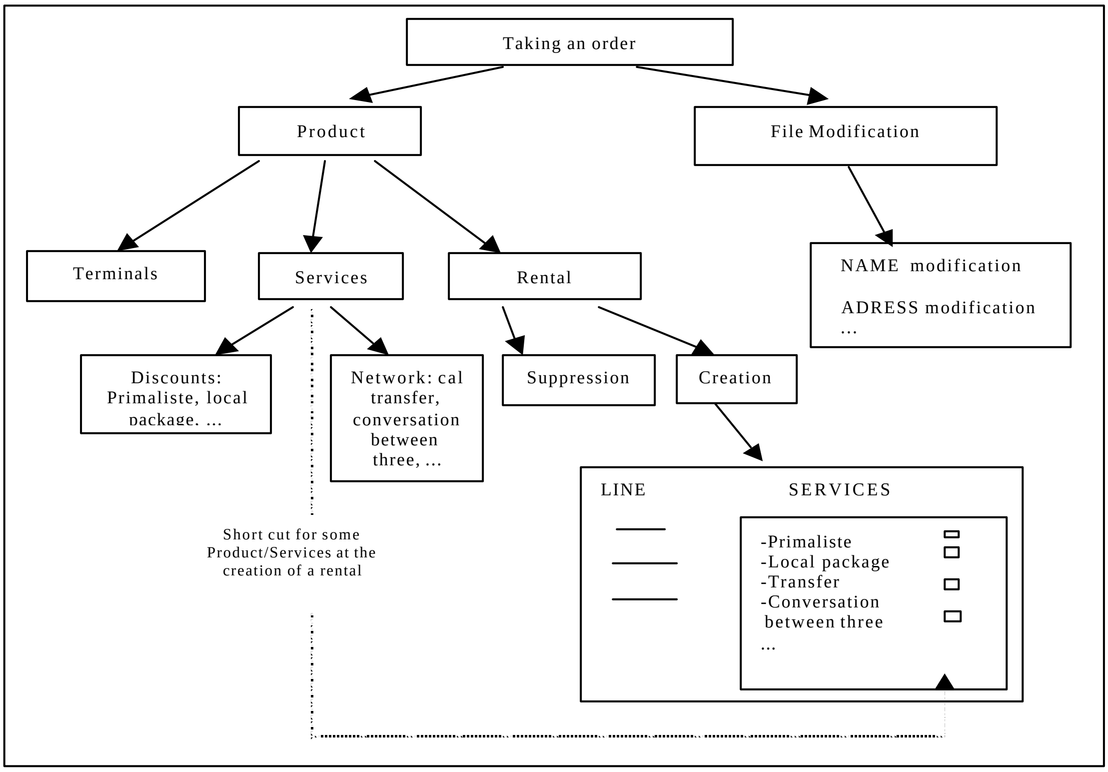

Figure 3: Scheme of a part of the user interface

Consequently to these mistakes and the complaints that they provoked, the owner who finances the project and the project manager who realises it, have met themselves on several occasions to decide which corrective actions could be done. The ABRICo schemes represented figure 4 and 5 are an artificial construction, inspired by real situations, corresponding to the example presented above.

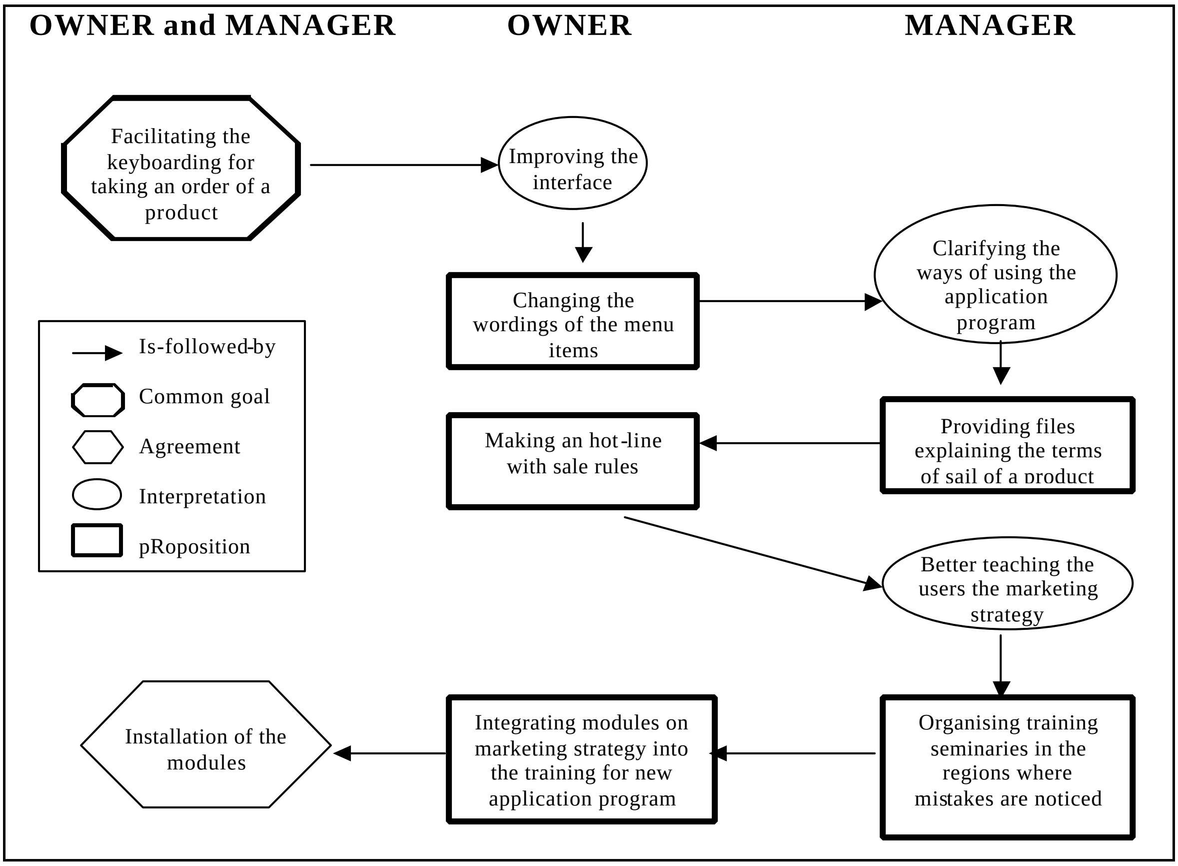

Figure 4: Dynamic representation of an example of a co-design process

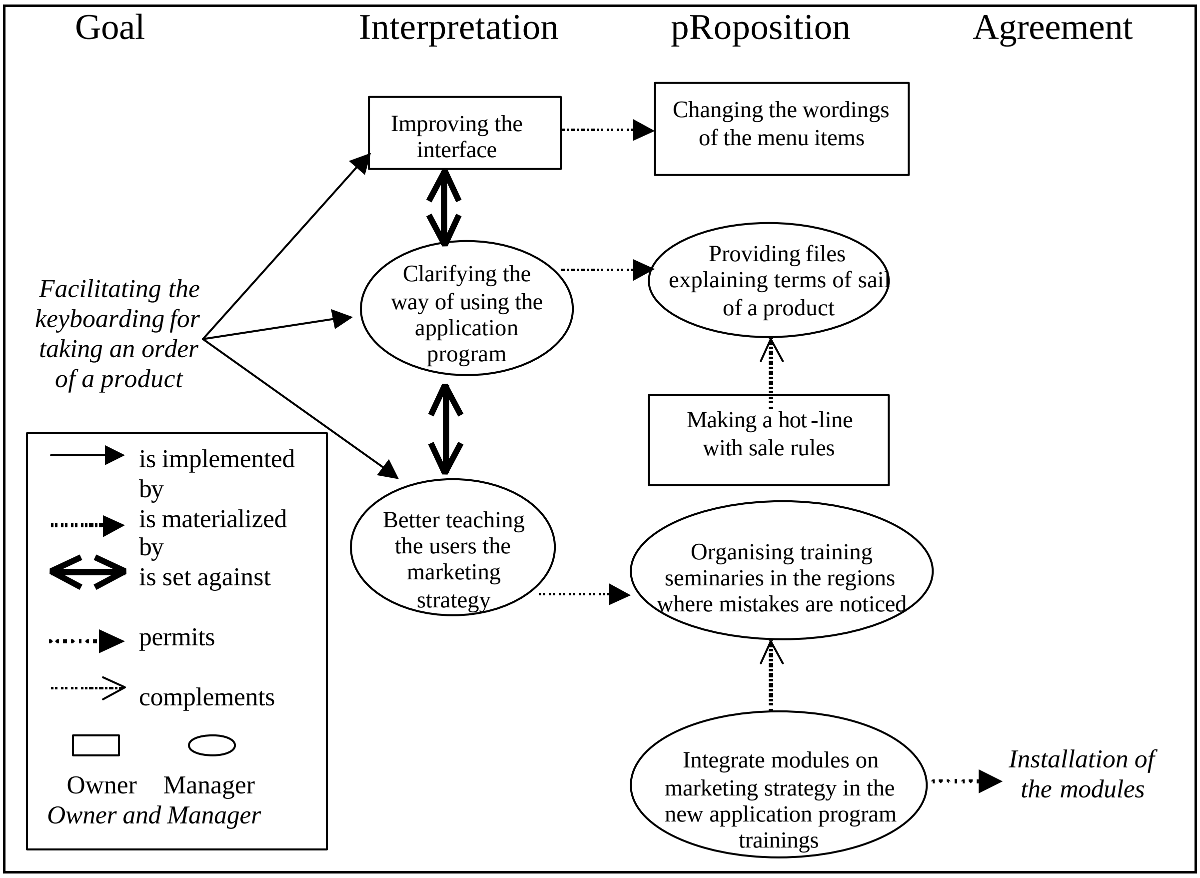

Figure 5: Static representation of an example of a co-design process

To experiment this formalism, we are going to install a GroupWare (that we are now developing in Lotus Notes Domino) accessible by the owner, the project manager and the development department (which realises what has been specified by the designers) via an Intranet. Our aim is to provide a tool that permits to structure and to memorise the discussions. The interventions will be classified in different views: by authors, by date, by abstraction level (goal, interpretation, proposition, and agreement). We will then be able to evaluate by interviewing the ABRICo users if it is easily usable and if it is efficient to represent exchanges. We are now going to present how a tool constructed on the basis of the ABRICo formalism answer the needs expressed by the designers team in term of formalisation and memorisation.

CONCLUSION

In conclusion we are going to present how, in the following of our work, we think that the installation of a GroupWare exploiting the ABRICo formalism could answer the designers needs :

Concerning the access to updated information and the coordination problems, the dynamic character of the formalism that gives for each argumentation category the one from it temporally ensues, allows a designer to know at any moment the last interpretations and propositions exchanged and the way they follow on from each other. That answers the need for updated information, which improves coordination and may avoid double tasks.

For the problems of the lack of clarity in the roles distribution and the lack of organisational coordination, the representation of the actors roles who formulate the argumentation categories obviously permits to know "who says what". That also permits to analyse, comprehend, indeed evaluate the relationships between the parties.

Concerning the lacks of functional documentation and criteria for efficiency evaluation studies notes, the ABRICo representation of the decision-making processes between the owner and the manager of the project may allow the designers to have a trace of all the functions that have to be implemented in the application program. The accessibility to the interpretations, which are at the origin of the propositions also, provides criteria for the evaluation of the adequacy of the proposed solutions to the functions. Likewise, by looking at past discussions on similar themes as the one on which he is working, a designer will be able to identify more precise criteria to evaluate the relevance of the functions that he is studying.

Finally, on a general point of view, the representation of argumentation taking place during collective design processes in the project and the representation of how past decisions were made, provides a structured memory of the project that will allow newcomers (from an other project or from the outside of the firm) to inform themselves and to be quickly integrated in the project.

We hope then that by broadening the Design Rationale to complex collective design situations thanks to ABRICo formalism, we will explicate a part of tacit knowledge exchanged during design discussions and we will inscribe them in the organisational memory through a GroupWare. In this way we fulfil the two conditions for organisational learning identified by Nonaka: conversion of tacit knowledge in explicit knowledge ("externalisation") and memorisation of these results.

REFERENCES

Argyris, 1978 - Argyris C., Schön D.A., *Organizational learning: a theory of action perspective*, Addison Wesley, Reading (Mass.), 1978

Bannon, 1991 – Bannon L.J., Schmidt K., *CSCW: Four characters in search of a context,* studies in CSCW- Theory, Practices and Design, Ed North Holland, Amsterdam, pp 3-16, 1991

Ellis, 1991 - Ellis C.A. et al., *GroupWare: some issues and experiences*, Communications of the ACM, 34, n°1, pp. 38-58, January 1991

Giard, 1994 - Giard V., Midler C., *Management et gestion de projet: bilan et perspectives*, 1994

Giard, 1996 - Giard V., Midler C., *Management et gestion de projet : bilan et perspectives*, http ://panoramix.univ-paris1.fr/GREGOR/96-11.html, 1996

Grudin, 1996 - Grudin J., *Evaluating opportunities for design capture*, in T.P. Moran & J.M. Carroll (Eds.), Design Rationale: Concepts, Techniques and use, Lawrence Erlbaum Associates, 1996

Hatchuel, 1994- Hatchuel A., Apprentissage collectif et activités de conception, *Revue Française de Gestion*, pp. 109-120, Juin-Juillet-Août 1994

Hoc, 1989- Hoc J.M. (Editor), *Cognitive Psychology of Planning (Computers and People)*, Hardcover, 1989

Karsenty, 1996- Karsenty L., *An Empirical Evaluation of Design Rationale Documents*, CHI 96, 13-18 Avril 1996

Lee (1996)- Lee J., Lai K.Y., *What's in Design Rationale*, in T.P. Moran & J.M. Carroll (Eds.), Design Rationale: Concepts, Techniques and use, Lawrence Erlbaum Associates, 1996

Levitt, 1988 – Levitt B., March J.G., Organizational learning, *Annual Review of Sociology*, 14, pp. 319-340, 1988

Mac Lean, 1989- Mac Lean, Allan, Young R.M., Moran T.P., *Design Rationale : The argument beyond the artefact,* in Proceedings of CHI '89, Austin Texas, April 30-May 4 1989 ACM Press, 1989

Mac Lean, 1996- MacLean A., Young R.M., Bellotti V.M.E., Moran P., *Questions, Options and Criteria : Elements of Design Space Analysis*, in Moran TH.P., Carroll J.M., *Design Rationale Concepts Techniques and Use*, Lawrence Erlbaum Associates, 1996

Midler, 1996- Midler C., *l'auto qui n'existait pas, management des projets et transformation de l'entreprise*, InterEditions, 1996

Moran, 1996- Moran TH.P., Carroll J.M., *Design Rationale Overview*, in Moran TH.P., Carroll J.M., *Design Rationale Concepts Techniques and Use*, Lawrence Erlbaum Associates, 1996

Nonaka, 1994 - Nonaka I., A dynamic theory of organizational knowledge creation, *Organization Science*/ Vol.5, N°1, pp 14-37, February 1994

Palmer, 1994 - Palmer J., Fields N. Guest editors introduction: computer-supported cooperative work, *IEEE Computer*, 27 (5) 15-18, 1994

Schmidt, 1994- Scmidt K., *The organization of cooperative work : beyond the « Leviathan » conception of the organization of cooperative work*, In Proceedings of CSCW '94 (October 22-26, Chapel Hill, North Carolina, USA), ACM, N.Y., 1994, pp. 101-112

Schum, 1996-a- Schum S.B., *Analysing the Usability of a Design Rationale Notation*, in Moran TH.P., Carroll J.M., *Design Rationale Concepts Techniques and Use*, Lawrence Erlbaum Associates, 1996

Schum, 1996-b- Schum S.B. et al., *Graphical Argumentation and Design Cognition,* Technical Report KMI-TR-25, Knowledge Media Institute, The Open University, U.K., 1996

Yakemovic, 1993- Yakemovic K.C.B., Conklin E.J., *Report on development project use of an Issue-Based Information System,* in Readings in Groupware and CSCW, pp 566-579, Morgan Kaufmannn Publishers, 1993