

# DESIGN IN COLLABORATION: EXISTING TRENDS AND APPLICATION TO THE CASE OF CONFLICT HANDLING WITH CO<sup>2</sup>MED SOFTWARE

Muriel LOMBARD, Bertrand ROSE, Gabriel RIS

*CRAN Automatic Research Centre of Nancy, CNRS UMR 7039  
University Henri Poincaré, Nancy I, Faculty of Sciences, BP 239  
54506 Vandoeuvre-lès-Nancy - FRANCE  
Email:  [{firstname}.{name}@cran.uhp-nancy.fr](mailto:{firstname}.{name}@cran.uhp-nancy.fr)*

**Abstract:** New Organizational frameworks for companies dealing with product development are required for the new market rules as much as for the rising pressure and complexity in product and process development. Thus, the new stake is to examine new methods of leading the design activities. We examine tools and trends existing in collaborative design today and propose a repository and its software implementation for the most constrained case of collaboration in design: collaborative conflict handling.  
*Copyright © 2005 IFAC*

**Keywords:** Control remote collaboration, network of actors, collective work situations, CSCW tools, conflict handling.

## 1. INTRODUCTION

Project design management not only consists in allocating resources, but also in stimulating collaboration among the people involved in the project, in order to increase the performance of design teams. This paper presents an analysis of the type of collective work that could occur within the design process while focusing on the collaboration forms in order to insure a performing running of this activity. Various tools and trends existing in collaborative design are reviewed and lacks of these tools are highlighted. We then focus on the most constrained form of collaboration (design conflicts handling) in order to propose a repository to run a design conflict management process. The implementation of this repository in a web-based software solution will also be described.

## 2. VARIOUS TYPES OF COLLECTIVE WORK

What do we mean by collaboration? What are the varieties of collective work situations? It is established that actors of a project team up in order to enhance the group effectiveness with the aim of obtaining a sustainable gain. There seems to be a great deal of confusion about definitions. We remember here some definitions while highlighting the differences between them.

**Communication:** Commonly defined as « the action to pass on something to someone, to transfer a result ». It is the first level in the collaboration process

**Coordination:** Coordination is the process that aims to plan and schedule different tasks, and to distribute resources: coordination is the guideline for action (Legardeur et al. 2003). This refers to resources allocation needed for the various actors and the access facilitation of various information as well as the knowledge being able to help them to quickly reach their purpose.

**Co-operation** is defined in the Oxford English Dictionary as “to work together, act in conjunction [...] to co-operate for [...] mutual benefit” from the Latin term *co operari*. (Legardeur, et al. 2003) consider cooperation as an effective and concrete articulation among designers involved in a collective action (*working in practice towards a consensus end*). (Austin and Baldwin, 1991) emphasize on the fact that it is a work commonly performed, but also highlight that it must have a pre-established objective.

**Collaboration:** The act of working jointly with other persons. Collaboration implies a common view on the problem to be handled and require a shared environment in order to stock and share information. It is based on a mutual commitment of the participants to solve together a common problem (Dillenbourg et al., 96) in order to perform a common goal.

The distinction between the cooperation and collaboration notions does not seem to be necessarily obvious: both concepts enable to converge

collectively towards a common result while taking into account various parameters peculiar to each stakeholder. It is nevertheless crucial to note that it is in the way the tasks are divided that is important while separating the two notions. In cooperation, work is divided in independent sub-tasks, division in which each stakeholder is responsible for a part of the problem solving. Coordination steps in while assembling the various partial results (Dillenbourg, 1996). Collaboration refers in opposition to an activity synchronised and coordinated in order to build and maintain a shared view of a problem. (Mattessich and Monsey, 1992) highlights the fact that collaboration requires a greater commitment to a common goal than co-operation. Following the work of (Puleo, 2003), we synthesise the differences between the two notions in Table 1.

**Table 1 Differences between cooperation and collaboration (adapted from Puleo, 2003).**

<i>Cooperation</i>	<i>Collaboration</i>
Organisational goals not taken into account	Common goals are created.
Interaction with individuals on as an needed basis	Permanent interactions between stakeholders
Informal relationships- each organization functions separately	New organisational structure created with interrelated roles and division of labour.
No joint planning or goal setting.	One or more projects are undertaken for long term results.
Leadership is unilateral.	Leader of each part are fully committed. Leadership is dispersed and shared among participants.
Information conveyed as needed	Many levels of communication are created as it is viewed as a key element.
Resources kept separate	Resources are shared for a long term effort that is managed by the collaborative structure.

### 2.1. Collaborative work analysis

During a collaborative work, designer's tasks are performed in parallel and their results should be convergent to satisfy the design objectives. These objectives could be refined during the design project progress, so understanding collaborative design process to control it is important. Indeed, (Traum and Dillenbourg, 1998) emphasised that collaborative situations depend on whether participants are at similar levels of knowledge and ability, and whether they share common goals. They discussed the level of symmetry of action (participants able to and allow to perform the same tasks), knowledge (may be at a similar level, but not necessarily within the same field) and status within the community of collaboration.

Efficiency of collaborative work depends on the actor's capabilities to collaborate. It is therefore necessary to be able to pinpoint various kinds of situations according to a collaborative context and design objectives evolution. (Girard *et al.* 2003) have shown how it is possible to encourage collaboration thanks to an adapted collaboration form (Figure 1).

This taxonomy permits to evaluate the collaborative work according to the activity definition, the relationship freedom and the collaboration experiment of the actors. Consequently the project manager identifies on which characteristics he (she) could act to increase collaboration in order to satisfy design objectives. Nevertheless, Girard's taxonomy is generic and does not clearly take into account the shared knowledge during collaborative design process whereas it is very important to succeed in collaborative design activities (Robin *et al.*, 2004). Design is mainly a human activity and it is very complex to understand the activities carried out by designers (Gero, 1998). To increase design performances and consequently to satisfy customers' requirements and expectations of enterprises, there is a need to adapt the recruiting of the designers to their context of work and to gather them in a real network.

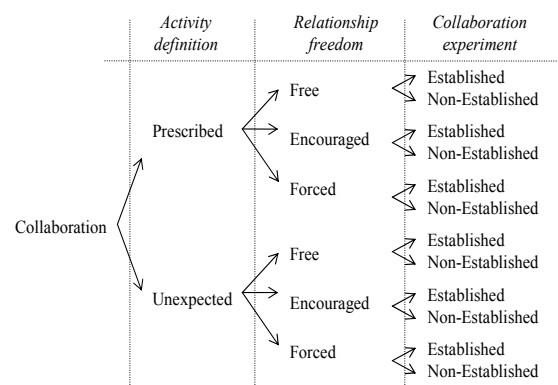


Fig.1. Collaboration taxonomy (Girard *et al.*, 2003)

### 2.2. A network of designers

Following the model defined by (Håkansson and Johanson., 1992) in Industrial Marketing, the design framework can be seen as a network gathering the elements "actors", "activities" and "resources". It is possible to distinguish between an actor network, an activity network and a resource network (*ibid.*). These three networks are strongly related to each other by four forces:

- Functional interdependence: actors, activities and resources form a system where heterogeneous demands are satisfied by heterogeneous resources.
- Power structure: on the basis of control of activities and resources there are important power relations between the actors.
- Knowledge structure: the design of activities as well as the use of resources is bound together by the knowledge and experiences actors.
- Intertemporal dependence: the network is a product of its history in terms of all memories, investments in relationships, knowledge, routines, etc.

(Håkansson and Snehota, 1995) gave a framework to describe how a relationship can be analyzed through

its individual substance layers: actor bonds, activity links and resource ties. The introduction of such a differentiation in the language facilitates the communication about these issues and totally fit in order to map the relationships within the “design teams”. In the same time, as far as more and more stakeholders are involved in the integrated design process, more and more knowledge is then shared and taken into account to develop the products. The Information System is supposed to collect and capitalize this knowledge by collecting the various data more or less structured and organised. One of the rare consensuses in the knowledge management domain is that knowledge is now perceived as an organisational and production asset, a valuable patrimony to be managed and thus there is a need for tools and methods assisting this management (Gandon, 2002), especially in a collaborative context. Some existing solutions are reviewed in the following section.

### 3. TOOLS AND TRENDS EXISTING IN COLLABORATIVE WORK

Indeed, if we go through the tools dedicated to collaborative work currently existing on the market, functions offered by CSCW tools (Computer Supported Cooperative Work) (fig.2.) are organised according to three dimensions of a collaborative activity (Salber, 1995):

- a communication dimension which allows direct exchange of knowledge between collaborating actors,
- a coordination dimension, which defines rules of interaction between actors themselves in a shared work space,
- a production space which allows production of shared objects, and the management of access to these shared objects.

Nevertheless, some trends exist within this framework in order to enhance the tools to fit the users' conditions.

#### 3.1. Awareness

When people work in “face to face” situations, the participants work to establish and maintain a shared understanding defined as common base (Clarck, 1996). When people work in a situation of collaboration but in a distributed context, resources used during the interactions are altered (Tang, 1991). So, the notion of “group awareness”, initially defined by (Dourish and Belloti, 1992) as being « the understanding of the activities of the others, allowing giving a context to its own activity » seems important to be implemented in a CSCW solution. It is therefore of primary importance to provide relevant communication means to overcome this lack.

#### 3.2. Malleability, flexibility and nomadism

We can also quote research works interested in integrating the concepts of malleability and flexibility

in collaborative systems, such as the DARE system, proposed by (Bourguin *et al.*, 2001). The concept of malleability within groupwares is the fact of proposing certain degree of flexibility and adaptability in the use of this type of software. Other works focus on objects sharing in distributed applications, such as (David *et al.*, 2003) which propose a concept of capillary CSCW by integrating the notion of “nomadism” (capacity of accessing to a particular knowledge from a mobile post, on any platform, which can connect and disconnect to different information sources).

#### 3.3. Lacks of these tools

Many research works focused on development of strong collaborative tools. As tackled by (Klein *et al.*, 2000), workflow systems currently provide little support for adaptive processes. Most do not allow one to modify a process model once it has started executing. Besides, most of CSCW tools focus on communication features (messaging) and coordination (approval forms, workflow tools, videoconference tools) but few of them are interested in collaboration among actors. Regarding this feature, besides shared documentary bases, market tools propose forums to facilitate exchange among actors. Such places do not formalize inter-skills relationships and do not guide the actors towards the setting up of a common knowledge project. As mentioned by (Nam *et al.*, 1999), the development of computer support for collaborative design as now to focus on ways of improving the interaction between collaborating designers in order to establish a suitable environment for sharing design information and knowledge.

A gap between CSCW literature and software proposals and an integrated Product-Process-Organisation solution really exists (Monplaisir, 1999). A global repository for integrated product, process and organisation is therefore requested in order to specify some reliable and relevant software solutions for collaborative product design.

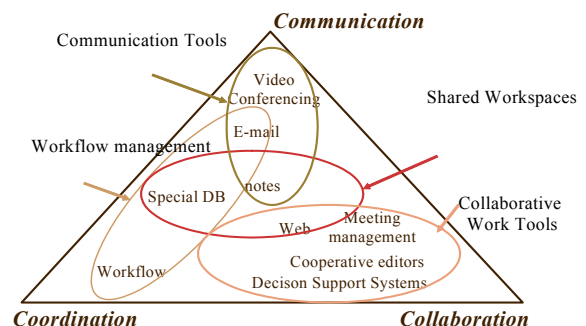


Fig. 2. Communication, coordination, collaboration and associated tools

### 4. PROPOSAL OF A REPOSITORY FOR CONFLICT HANDLING

We directed our works on the case of the most forced collaboration form (corresponding to the case of prescribed and forced collaboration in (Girard *et al.*, 2003). Based on the model and the dynamic protocol

established in (Rose *et al.*, 2004), we propose a repository for the collaborative design of products (figure 3). We expose here the software solution set up on the basis of these requirements, to capitalize the exchanged knowledge and an infrastructure allowing these exchanges of knowledge in his definition of organizational memory). Referring to the propositions of (Simone 1996) concerning the organizational memory, we identified the following objectives for

our repository Conflict management in a collaborative design repository:

- Adapt or innovate from the capitalized knowledge,
- Increase the collaboration,
- Manage the renewals of staff,
- Process the critical situations.

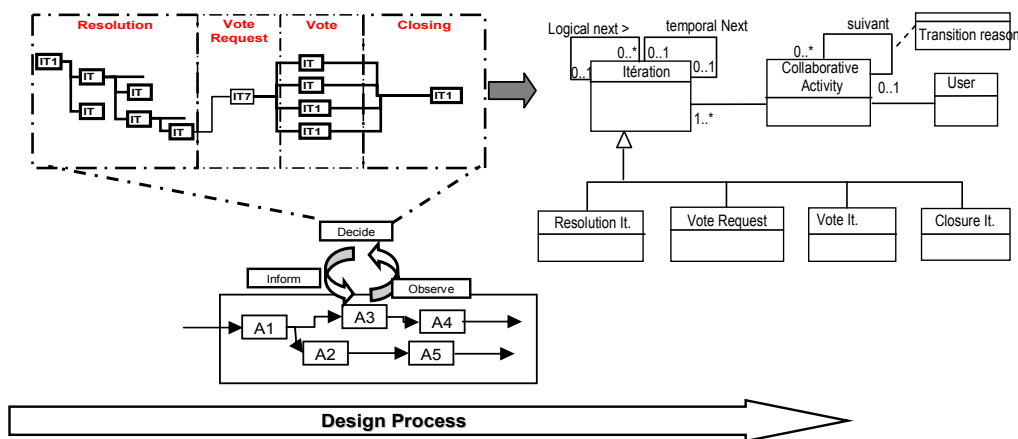


Fig. 3. Dynamic and static view of the conflict handling process

To have a good communication is implied by these objectives. They also underline the dynamic and adaptive characteristics inherent to this collaborative repository. The created repository has to rely on a common and shared meaning, but must nevertheless be able to adapt itself to the current context. This section presents a software solution able to answer these various needs in the case of conflicts management: the CO<sup>2</sup>MED software.

#### 4.1. Specifications of the repository

Collaborative design process consists of several stages alternating between definitions of new solutions and adapting old solutions. Based on previous work in the conflict management field, we apply the cybernetic loop protocol on this situation (Rose *et al.*, 2004). We can therefore highlight the *Observation* phase of the conflict, a *Decision* stage about the solution to be adopted and an *Information* to be brought to the involved actors (forwarding of the solution). In order to keep trace of the conflict resolution process (in terms of actions chaining and knowledge exchanged), we define an UML class diagram built around the class called “**Collaboration Activity**” (as a specialisation of an activity of the design process) to refer to the various resolution processes held to resolve different conflicts. Each “**Collaboration Activity**” is then composed of a set of “**Iteration**” classes. The “**Collaboration Activity**” class is instantiated to initialize the conflict resolution process. The “**Iteration**” class can handle several actions depending on the phase of the decision-process (arguing the actors’ choices, negotiating the proposed solutions, concluding the conflict...). The “**Iteration**” class was therefore specialized into three sub-classes in order to facilitate the capitalization of

the conflict management process by keeping trace of the different intervention.

### 5. IMPLEMENTATION OF THE REPOSITORY IN A SOFTWARE SOLUTION: CO<sup>2</sup>MED

We implement the proposed specifications jointly with our industrial partner Alstom Power Conversion in a software solution in order to experiment it in a real industrial case.

#### 5.1. Software architecture

In order to answer to the constraints of distributed design and to insure flexibility within the use of the software, we chose a web solution to implement the repository previously presented. It is therefore developed in PHP language, run on MySQL databases and is powered by an Apache Web Server. The ontologies are set up in PROTÉGÉ-2000 Software and are exported in an XML format in order to be rebuilt in an XML tree in the software.

#### 5.2. Functionalities implemented

The functionalities developed from the specification previously presented are:

- A personalized and secured access to the software (Users system management),
- A structured communication media (in fields explanation, justification solutions) to be able to exchange the various knowledge used to achieve the process of resolution,
- The management of collaboration activities and iterations,
- A vote system,
- A storage of the iterations facility,

- A Performance Indicators system dedicated to the human resources management (in order to build a reliable and competent network while recruiting the ad-hoc partners) and statistics.
- Domain ontologies integration as control lever for indexing met conflicts (these ontologies, easily changeable and upgradable can fit to the conflict handled with a relative flexibility).
- An inference system to help the actors to set up solutions adapted from the ones evoked in the past and proposed via the ontology.

We used our software prototype during the resolution of a conflict discovered during the design of a stator at Alstom Power Conversion Nancy. This stator was a part of a windmill generator. The global design process of this stator is detailed in (Rose *et al.*, 2004). A major constraint of this project concerned the bulk of the engine to be realized. In this paragraph we present the use of CO<sup>2</sup>MED for this conflict resolution.

#### Creation of the collaborative task:

This part consists in giving a title to the new collaborative task and in affiliating it to one of the elements of the ontology. This action is realized by the actor having discovered the conflict, namely, in our case, the expert in mechanical calculation who was in charge of verifying the design of the engine. Here, the problem concerns the sheet steel stator. The outside diameter is out of the range of acceptable tolerances. The stake is here to affiliate the conflict with the component from the “Product” ontology on which the conflict appeared. A second referencing with an item of the “Conflict” ontology can be also realized to characterize in a more precise way the type of met conflict. In our case, the conflict concerned the sheet steel stator and we considered a conflict of resonance on the stator.

#### Initialization of the conflict:

Once the activity was created, it is necessary to initialize it by creating a first iteration in which the problem is explained. If information is missing, this will be asked by the concerned users in the following. In order to run this process, it is beforehand necessary to have chosen the actors to subscribe to the process of resolution of conflict, among those declared in the data base. The consultation of the page relative to the Performance indicators can be a major asset to define the most competent and dynamic actors to achieve the conflict resolution. This is realized via another page available via the menu. For the studied example, the persons urged to resolve the conflict were:

- An actor from the Study Office,
- An actor of the electric calculations,
- An expert in mechanical calculations,
- The final customer.

#### Popularization and mediation of the conflict:

Once the collaborative task was initialized, the users previously selected can intervene by answering to the iterations already uttered via explanations or justifications (Popularization of the conflict). During these exchanges, one of the users can take the initiative to create a new solution which can be

discussed by the consortium (Mediation). The Human/Machine Interface enables us realizing these exchanges is proposed in figure 3. In our example, the various actors discussed the problem causes. The identified cause was the disregard of a calculation instruction. Several solutions were envisaged to resolve the conflict. After various iterations of Popularization / Mediation, the actor from the Study Office suggested integrating the cooling system of the engine (originally outside of the stator) within this one; by enabling the coolant liquid transporting tubes to pass through every stator sheet. To help the actors in their interventions, the software is provided with an inference function allowing them to consult the various exchanges having taken place during the resolution of previous conflicts of the same type through the use of ontology.

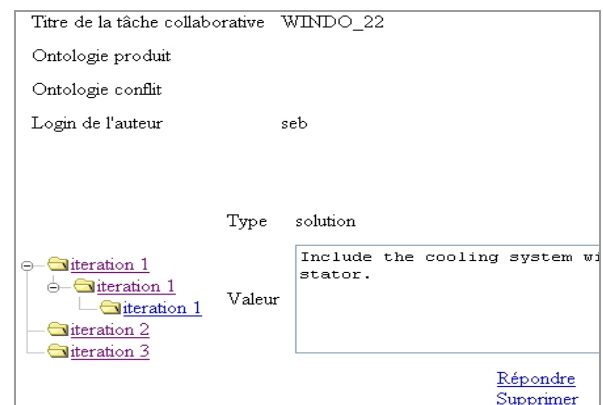


Fig. 4. Communication formalisation in CO<sup>2</sup>MED.

#### Vote demand:

At the end of a predefined duration  $D_j$ , in case of no convergence between the emitted solutions, a vote request message is sent to the various participants, meaning them the time which is granted to validate their choice from the proposed solutions.

#### Vote:

Each subscribed actor must be identified to be able to vote and then chooses a solution among those previously uttered during the various previous iterations and valid his choice.

When the actor sees that no solution is relevant, he can decide not to vote. The vote must be handled within the allocated timeframe defined by the vote demand iteration and must answer to various constraints:

**Constraint of participations:** a minimum quorum of voters is requested to validate it.

**Constraint of votes:** a minimum quorum of explained votes is requested by a proposed solution in order to be adopted.

#### Closing iteration:

The instantiation of this iteration occurs in three different situations:

- Further to the consensus obtained by explicit support of all the actors to a given solution before the deadline of the popularization / mediation phase. So the subscribed actors unanimously gave their favorable opinion for a given solution in filling the attribute "solution" of an iteration (further to the proposition of this solution by one

of them). In this case of friendly resolution, the actors responsible for the collaborative activity can automatically instantiate a closing iteration without launching the vote stage.

- Further to the consensus stemming from various iterations of vote emitted by each subscriber. If the constraints of participation are respected, the field "Solution" of the iteration is then automatically filled by the system with the solution to be used.
- In the case of vote process failure due to a lack of participants (voters' quorum not reached) or by non-consensus (quorum of votes not reached), the project manager decides on the solution to be used and manually fills the attribute "Solution" of the closing iteration.

In the studied example, the vote having unanimously been won by the solution suggesting integrating the cooling system within the stator, it was this solution which was adopted. The conflict is closed by a button, the software capitalizing the various exchanges and the solutions evoked during this resolution in an arborescence of iterations (Fig. 4).

## 6. CONCLUSION

The product design actors today evolve in a context of strong interactions and sharing of knowledge. Conflict management therefore appears in this context as the most constrained form of collaboration. This article presented the different forms of collective work situations and the various tools existing to support collaboration and particularly collaboration in design. We highlighted the lacks of these solutions and the need for a reliable repository to run performing design activities. We proposed static and dynamic specifications of the CO<sup>2</sup>MED software solutions in order to handle conflict occurring in design. This work is embedded within the IPPOP project (IPPOP, 2001) whose aim is to provide a complete demonstrator integrating the product, process and organisation perspectives of design.

## REFERENCES

Austin, A.E. and Baldwin, R.G. (1991), Faculty Collaboration :Enhancing the quality of Scholarship and Teaching, Washington DC University report.

Bourguin, G. Derycke, A. Tarby, J.C. (2001) Beyond the Interface: Co-evolution inside Interactive Systems—A proposal founded on Activity Theory. Proceedings of IHM-HCI 2001 conference, Lille, France, 10-14 September.

Clark, H.H., (1996). *Using Language*. Cambridge University Press, NewYork.

David, B. Chalon R., Vaisman, G., Delotte, O.,(2003) Capillary CSCW, In *Human-Computer Interaction Theory and Practice (Stephanidis C., Jacko J., ed.) Vol. 2*, LEA, London.

Dillenbourg, P., Baker, M., Blaye, C. O'Malley, (1996), The evolution of research on collaborative learning, in (Spada and Reiman

(Eds) *Learning in Humans and Machine : Towards an interdisciplinary learning science*.

Dourish, P., Bellotti, V., (1992), Awareness and Coordination in shared Workspaces, in Proc. Of ACM Conference on CSCW, Toronto, Canada.

Gandon F., (2002), *Ontology Engineering: a Survey and a Return on Experience*,INRIA report RR-4396, team ACACIA, Sophia Antipolis.

Gero J.S., (1998) An approach to the analysis of design protocols, *Design studies* **19 (1)** 21-61.

Girard Ph., Robin V., Barandiaran D., (2003), Analysis of collaboration for design coordination, 10th ISPE International Conference on Concurrent Engineering: Research and Applications. CE'03, Madeira, Portugal.

Håkansson, H. and Johanson, J. (1992), *A Model of Industrial Networks*, Routledge.

Håkansson, H. and Snehota (1995), I., *Developing Relationships in Business Networks*, London: International Thomson Business Press.

IPPOP (2001) Integration of Product-Process-Organization for engineering Performance improvement <http://www.opencascade.org/IPPOP>

Klein M., Dellarocas C., Bernstein A., 2000, Introduction to the Special Issue on Adaptive Workflow Systems, *CSCW Journal*, **Vol 9 issue 3**, pp 265-267.

Legardeur J., Merlo, C., Franchisteguy, I., Bareigts C., (2003) Coordination et Coopération dans les processus de conception, AIP Priméca La plagne.

Mattessich, P.W. and Monsey, B.R. (1992) Collaboration: What Makes It Work, Amherst H. Wilder Foundation, St. Paul, MN.

Monplaisir, L., (1999), An integrated CSCW architecture for integrated product/process design and development, *Robotics and Computer-Integrated Manufacturing* **Vol 15**, 145-153.

Nam T. K., Wright D, (2001) The development and evaluation of Syc03D: a real-time collaborative 3D CAD system, *Design Studies* **Vol 22, Issue6**, pp 557-582.

Puleo, M (2003) It's not just Summer reading anymore, NMRLS IDYLS Conference, 4 /09.

Robin V., Rose B., Girard Ph., Lombard M., (2004), Management of engineering design process in collaborative situation. 14th International CIRP design seminar, May 16-18, Cairo, Egypt.

Rose B., Lombard M., Gzara L., Lossent L., Claudon P., Leboeuf P. (2004) Industrial Case of Product Design: Towards Specifications of a Collaborative Design Tool, e-challenge 2004, Vienna, 27-29 October.

Salber, D., (1995) De l'interaction homme-machine individuelle aux systèmes multi-utilisateurs. PhD. dissertation, University of Grenoble,France.

Simone C., (1996), *Collective Memory from Everyday Work*. In Y. Waern.

Tang, J.C., (1991). Findings from observational studies of collaborative work. *International Journal of Man-Machine Studies*, **Vol 34**.

Traum, D.R. Dillenbourg. P. (1998), Towards a Normative Model of Grounding in Collaboration. In working notes, ESSLLI-98.