Collaboration in Requirements Engineering Process

Jacqueline Konaté[1, 2], Abd-El Kader Sahraoui[1, 3]

[1]*LAAS-CNRS: Laboratoire d'Analyse et d'Architecture des Systèmes, 7 Avenue du Colonel Roche, France, {konate,sahraoui}@laas.fr*

[2]*Université Paul Sabatier y of Venus, 118 Route de Narbonne –F-31062 Toulouse cedex 9, France*

[3]*IUT Blagnac Université de Toulouse le Mirail, 1 place Georges Brassens 31703 Blagnac, France*

Abstract

The paper addresses the merging the collaboration process in systems engineering context. The work is limited to the approach in development a collaborative working environment application free. For such objective, the systems engineering concepts and paradigm are adapted to any systems development. For better determining this problem, the processes of products development according to system approach are presented. After this presentation, some references to existing works in collaboration area, and finally the proposal of this article. The solution suggested by our approach is a solution-based process. These processes are defined from engineering processes already identified. That means that our approach separates engineering processes and collaboration processes. The standard EIA-632 is taken as our study case. We illustrated it with an example by using SysML notation

Keywords

Collaborative processes engineering, systems engineering, EIA-632.

1 Introduction

Collaboration is an old concept and practice which is currently the subject of active research in using actual enabling information technology. However, it appears normal to raise following question: why so much of references nowadays to collaboration notion? Indeed, the current economic context encourages the companies with fusion-acquisitions, with the externalisation of many activities. Consequently, the companies are forced with better communicating with their customers, their partners, their suppliers, their subsidiary companies, to produce more quickly and at lower cost.

Current collaboration is doing rather in an ad hoc way, i.e. there are no clearly defined processes governing interaction between actors implied in development of a system. This leads to design tools dedicated to collaboration which present many weaknesses; effectively, much of tools which are masterpieces from a technical viewpoint represent a failure because simply they do not take into account collaborative dimension. Furthermore, those tools are continuously in development because of improvement of technology that supports them; they are technology related. Compared to what precedes, this paper proposes an approach which in fact leaves the analysis of a set of engineering processes in particular those of standard EIA-632 [EIA-632 1998], to define processes so-called collaboration processes. This study on collaboration is placed on products development or systems engineering level. The multidisciplinary character of system engineering justifies the relevance of its selection. Indeed, an efficient complex products development requires effective collaboration between various competences.

The continuation of this document has following organization:

Section 2 presents a kind of state of the art about collaboration and systems engineering. Our approach is exposed in sections 3. Section 4 shows an example to illustrate our approach. Finally, conclusions and perspectives are outlined in latest section.

2 State of the Art: Collaboration and Systems Engineering

This section presents some related works in collaboration and systems engineering area and then, it make link between these two areas.

2.1 Some Existing Work about collaboration

In literature, the terms collaboration, coordination and cooperation are often invoked together. The difference between cooperation and collaboration is not often perceptible so that they are sometimes confused. Our objective is not to study the three concepts, but it is interesting to give some definitions about them although this study relates only to collaboration. Collaboration are also social issues addressed by some research such as [Joseph 1994] which approaches requirements engineering as the reconciliation of social and technical issues. De Terssac and Maggi [Terssac, Maggi, 1996] think that cooperation is a mean to overpass individual limitations. There are others definitions of cooperation in [Schmidt, Bannon, 1992], [Soubie 1998].

Collaboration is defined by [Dillenbourg, Baker, Blaye, O'Malley, 1996] as a mutual implication of the participants in an effort of coordination in order to solve a problem arising. The paper [Menachoff, Son, 2003] sees collaboration as a framework allowing to arrange various cooperation forms between independent organizations. Agents implied in collaboration must have a common objective which can be partial or total, they must also have communication means and capacities to decompose problem into sub-problems.

According [Den, Dean, Kolfschoten, Chakrapani, 2006], what is more important is the explicit design and management of collaboration processes integrated with the development and deployment of collaboration technologies to support these collaboration processes.

2.2 Systems Engineering and EIA-632 Standard

Collaboration being a vast concept which includes many fields, for better studying collaboration, it is better to choose a precise field in order to address it. Thus we were interested in collaboration within the framework of the product development following system approach. However, researches on collaboration in fields such as organisational field [Health, Larrick, Klayman, 1998], the communication area [Walker, 2006] and so on are in full effervescence. Our study is located side of engineering systems. This study on collaboration intends for the products development. Here, a product is a system having a life cycle following steps: Conceptualization, Systems design, Components realization, System integration, Transfer toward exploitation, Exploitation, Disposal.

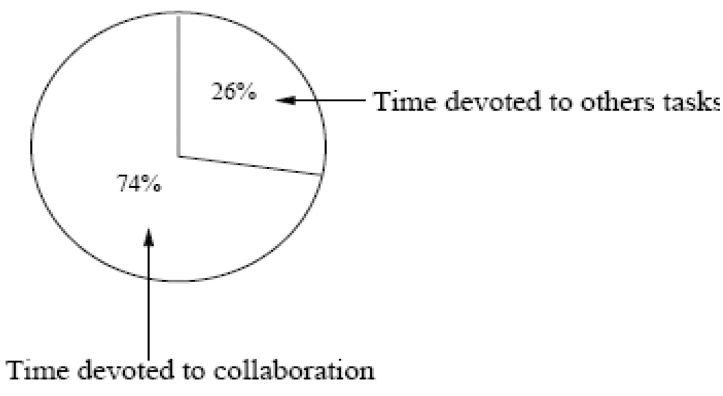

Figure 1: Time devoted to products development by engineers. (From [PTC 2007])

At each step of this life cycle, collaboration intervenes. Thus, a study shows that collaboration takes an important place in product development [PTC 2007] as depicted by figure 1. Collaboration increasingly occupies a considerable part in the products development, its improvement becomes a great factor of productivity.

Because of the current complexity of the systems, they are not built any more like formerly. Systems engineering defines a set of rules and processes according which systems must be developed. Our work aims to define collaboration by defining activities in term of processes. Experience shows that implementing technology alone seldom is the answer to effective and efficient collaboration. On the basis of these remarks, efforts are to be led to the level of the definition of collaboration processes rather than on tools. With this intention, we were interested in the processes of standard EIA-632 [EIA-632 1998]. This standard comprises 13 processes which apply all along product life cycle. Since all the stages of this life cycle does not concern us. Our approach targets the two design processes, the two realization processes and the four validation technical processes. The five isolated processes are processes which come under the organisational and logistic field that exceeds the framework of the context in which we work. Systems engineering is based on an iterative process that starts with (1) understanding a problem before you attempt to solve it, then (2) examining alternative solutions (do not jump to a point design), and then (3) verify that the selected solution is correct before continuing the definition activities or proceeding to the next problem. Each process is a succession of activities. According to connected activities, the processes differ. In the above engineering processes, there is a sequence of engineering tasks. Concretely, these tasks say what it is necessary to do with each stage of product development. For example, at design step it is necessary to define the requirements and to define the solution. To achieve these tasks which can be complex (made up of sub-tasks) or elementary, actors must collaborate. Indeed, engineering processes show that there is collaboration through the arrows, but they do not define it. It is in the light of these observations that this article was fixed as objective the development of collaboration processes starting from engineering processes. This paper is focused on these collaboration processes definition. Following section (section 3) is devoted to describe our approach.

3 Toward a System Engineering Collaborative Working Environment (SEC-WE) Application free

3.1 Introduction

We consider the starting point to what is presented in the figure 2 in (*a*). It's a situation where engineering and collaboration processes are merged and where it is difficult to clearly distinguish them. In some cases, if the system complexity allows it, the engineering process can be carried out by only one person. However, once the engineering of a system implies several people, it becomes necessary to separate the two aspects, i.e. collaboration and engineering. This last remark led us on vision illustrated in (*b*). After analysis about problem and attempts to dissociate engineering and collaboration processes, we noted that the two areas aren't completely disjoint. At this moment, we begin to adopt the vision highlighted by (*c*). That means collaboration has signification only it is taken in a particular context which is work to be realized. It is true that collaboration is built up around engineering project and there are bonds between them. This paper addresses the part of collaboration.

In the literature, one speaks much about collaboration, but without fixing the limits of the two fields in order to clearly express relations between them. On the basis of this remark, we carried out the separation of the two aspects by considering at the beginning that they were disjoint. The engineering processes are already known and standardized such as EIA-632. Thus we will not define them because we consider that they are already known and accepted by all [EIA-632 1998]. However, the processes of collaboration which are associated to them need to be identified and defined independently them. This new separation of problems leads us to focus on the part of the collaboration not coupled with engineering. It is not collaboration in the broad sense, but collaboration applying to engineering system. So the awaited processes of collaboration must be applicable to any standards of systems engineering.

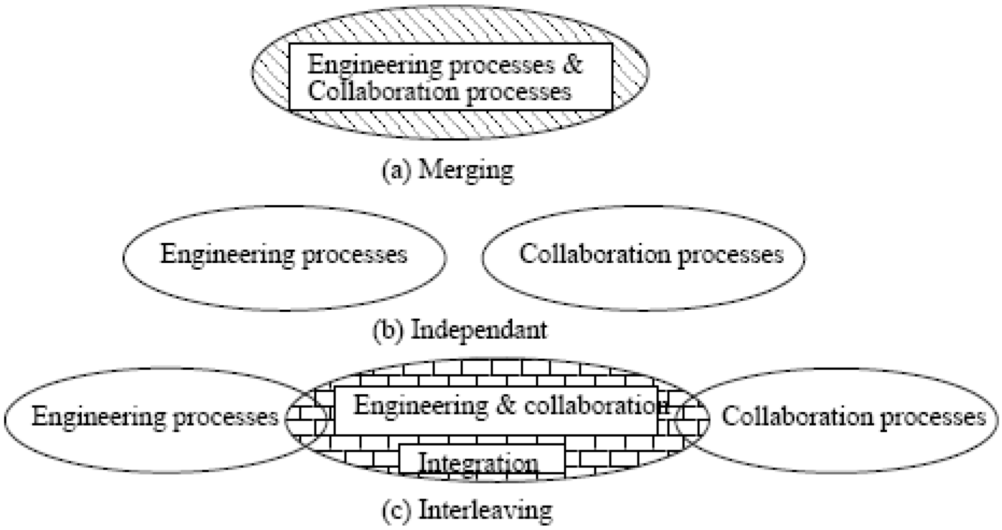

Figure 2: Problem statement

3.2 Collaboration model

Our vision of collaboration can be summarized in following formal notation. Here, collaboration is a function such as:

Collaboration: (***PROC$_I$***, ***RUL***) X ***PROC$_C$*** –> ***RESULT***

Where:

PROC$_I$ represents a set of engineering processes and ***PROC$_{EIA-632}$*** includes ***PROC$_I$***, ***PROC$_{EIA-632}$*** represents the processes of standard EIA-632, *RUL* is the set of rules which govern collaboration, ***PROC$_C$*** is collaboration processes.

This definition is the basis of our approach; indeed, we see collaboration in engineering area as a function using engineering processes with constraints to respect and collaboration processes which describe different interactions between participants and the data flow. Here, ***PROC$_I$*** is one process of engineering processes of EIA-632. Rules allowing efficient collaboration are defined using constraints to which actors, companies and activities are submitted. For more details on these rules, please refer to section 3.3. ***RUL*** and ***PROC$_C$*** are the unknown ones of above definition are ***RUL*** and ***PROC$_C$***. Following subsections are devoted to determine them.

3.3 Collaboration Rules: *RUL*

Like any process, collaboration processes are submitted to constraints relating to actors, activities and used resources. These constraints are taken account by collaboration rules definition. For example, rules linked to actors are submitted to constraints of disponibility, competencies, confidentiality, etc. Collaboration activities depend on the control structure of engineering processes and the rules relating to collaboration resources are under availability, right of property, etc. The number of rules can increase exponentially if we don't pay attention to the constraints we wish to take into account. For example, let two actors ***A*** and ***B*** working together to accomplish task ***T***. Here are some:

- ***A*** and ***B*** must have same goal, i.e. implement ***T***,
- ***A*** and ***B*** must be competent to implement ***T***,
- ***A*** and ***B*** must be available for ***T*** implementation,
- Needed resources to implement ***T*** must be available,
- Collaboration activities for ***T*** implementation depend on the structure of control of engineering process associated with ***T***,
- …

En short, we can identify three types of constraints: actors' constraints, activity constraints and resources constraints. A well definition of collaboration rules is essential because it's a basis of well-defined collaboration processes and it allows specify possible assumptions on these processes.

3.4 Collaboration Processes

We define a collaboration process in two parts. First, we establish its structure of control. Secondly, we define interactions for collaboration processes.

3.4.1 Control Structure of Collaborative Processes

This part is focus on the rule relating to activity constraints. The interactions and activities sequence are in the center of a collaboration model. In our approach, the control structure of collaboration process isn't identical to engineering process control structure. However, the first depends on the second. The control structure defines the partial order execution of activities.

3.4.2 Interactions

From preceding description, we can deduce following model which is a model for collaboration for the standard EIA-632. As described in the section 2, this standard distinguishes three types of actors: developer, acquirer and other stakeholder of system. We also kept the same actors as collaboration processes actors. These actors can indicate an individual or a work team. Figure 3 in page 5 is our collaboration model. We can note that collaboration is made around a set of tasks. Collaboration process gets inputs and gives outputs; it is also submitted to three types of constraints which conduct to collaboration rules defined in section 3.3. This model is applied to requirements engineering with SysML [Sandford, Anlan, Rick, 2006] in section 4.

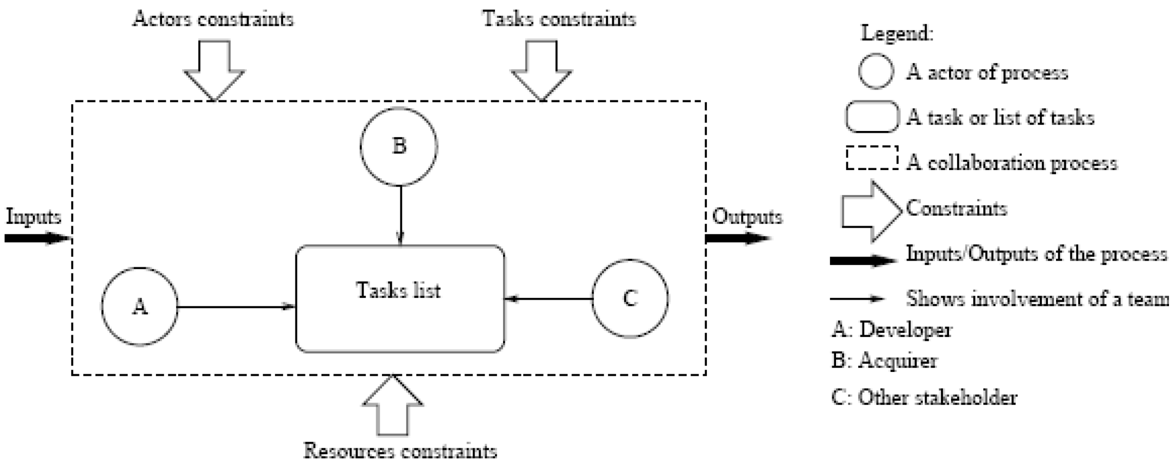

Figure 3: Collaboration model for EIA-632

4 Application: Collaboration Processes for Requirements Engineering

4.1 Modelling Collaborative Actors

Here, we illustrate our approach with requirements definition processes. Figure 4 highlights '*Collaboration process for requirements engineering*' ***P*** with its four sub-processes ***P1***, ***P2***, ***P3***, ***P4***. These sub-processes represent respectively '*Collaboration process for identification and collect of needs*', '*Collaboration process for needs transformation into technical requirements*', '*Collaboration process for requirements validation*', '*Collaboration process to record requirements*'. Actors involving in processes are indicated by arrows leaving the actors and directed towards the tasks of the process. The arrows directed to the bottom show the data flow (inputs and outputs). In particular, inputs are needs from acquirers' team (***AT***) and other stakeholder team (***OST***). These needs are identified and collected in collaboration of requirements definition team (***RET***). A set of needs resulting from this first process (***P2***) is used as input for second sub-process (***P2***) to be transformed into technical requirements. In third step (***P3***), technical requirements are validated by all actors and they are finally recorded by requirements definition team (***P2***). The result of the process ***P*** is a requirements document. We use formal modelling language SysML to model process (***P***). SysML which is a language inspired of UML and aims to allow systems modelling within for the needs of engineering systems. We show interactions between participants in collaboration and data exchanges.

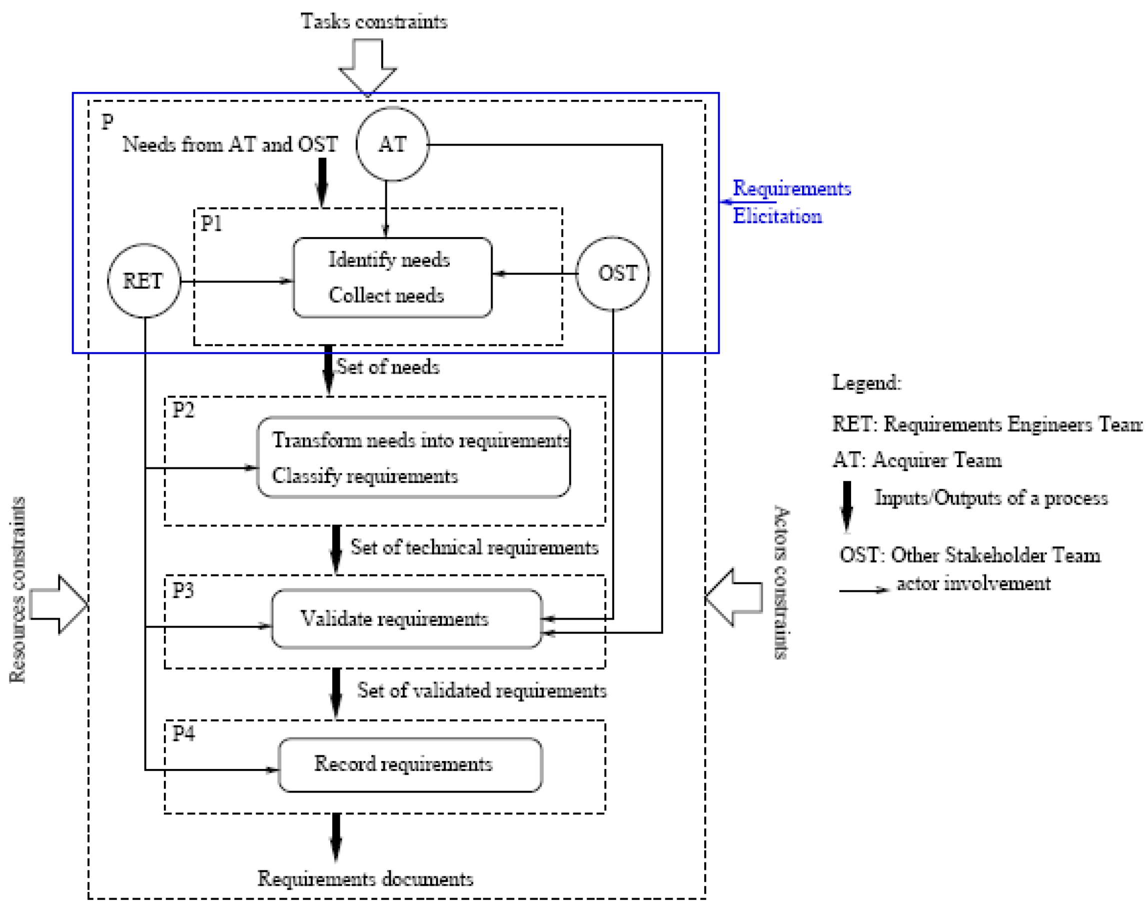

Figure 4: Requirements definition process

4.2 Interaction Model with SysML

The figure 5 in (a) shows actors and activities they must accomplish in requirements engineering process. We use SysML for our work perspectives because we are in systems engineering area.

In picture 5 in (b), we use interaction diagram to describe how actors interact in collaborative requirements definition process. Like use case diagram, this interaction diagram is a high level diagram because it does not shows all details of requirements engineering. But, it simply allows to understand the process on hand.

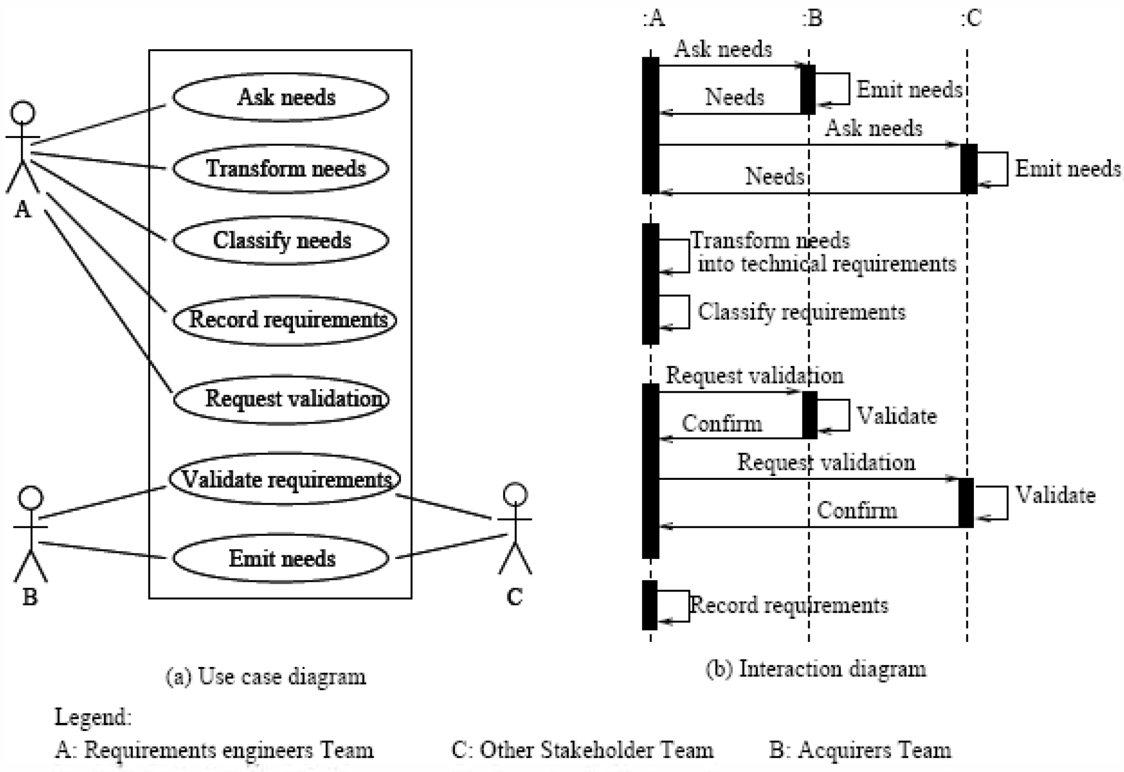

Figure 5: High level use case and interaction diagram for requirements definition process

4.3 Towards an Elicitation Collaborative Framework

We believe that our approach is promising and this is why we wish to refine it in a next work. This work will be centred on modelling collaboration processes in order to develop a tool for collaborative requirements engineering. We focus our work on the initial sub process of the requirements definition process: *requirements elicitation process*. The paper [Coulin, Sahraoui, 2005] presents a meta-model in requirements elicitation process. This approach takes advantage of benefits gained from using facilitated collaborative workshops whereby all relevant stakeholders can cooperatively contribute to results, and the combination of complementary techniques used to support the main activities, as well as within the actual requirements elicitation environment. Since their work offers possibility to instantiate requirements elicitation, we project to inspire us from their approach in order to instantiate collaboration for requirements elicitation.

5 Conclusion

This article suggests an approach-based process. This approach is focused on a good definition of collaboration processes which is guarantee for a good design of a tool. For this intention, we separate concerns by considering engineering processes are different from collaboration processes. This separation led us to define collaboration as a function using EIA-632's engineering processes, some collaboration rules and collaboration processes. The result of this function depends on process on hand, in particular a requirements document if we are in requirements definition context. The contribution of this article is to have defined the collaboration processes starting from engineering process and to have modelled them. It is also that makes the originality of our approach compared with others.

References

Coulin, C.; Sahraoui, A.E.K.: A meta-model based guided approach to collaborative requirements elicitation for software systems development. Rapport LAAS No05386. 18th International Conference on Software and Systems Engineering and their Applications (ICSSEA'2005), 29 November - 1er December 2005, 6p

Den, Hengst M.; Dean, D.L.; Kolfschoten, G.; Chakrapani, A.: Assessing the quality of collaborative processes. System Sciences, 2006. HICSS '06. Proceedings of the 39th Annual Hawaii International Conference on Volume 1, 04-07 Jan. 2006 Page(s):16b - 16b

Dillenbourg, P.; Baker, M.; Blaye, A.; O'Malley C.: The evolution of research on collaborative learning. In E. Spada & P. Reinman (Eds.), *Learning in Humans and Machine: Towards an interdisciplinary learning science*. Oxford: Elsevier. (pp. 189-211)

Electronic Industries Alliance: EIA-632: Processes for systems engineering. ANSI/EIA-632-1998 Approved: January 7, 1999.

Heath C.; Larrick R. P.; Klayman J.: Cognitive repairs, How organizational practices can compensate for individual shortcomings. Research in Organizational Behavior, (20):1–38, 1998.

Joseph A, Goguen: Requirements engineering as the reconciliation of social and technical issues. pages 165–199, 1994.

Menachoff, D. A.; Son, B. G.: The truth about collaboration. Chief logistic officer-Penton Media, pages 6–12, 2003.

Parametric Technology Corporation (PTC): Conception d'un système de collaboration. WWW page. http://www.ptc.com, accessed Jan. 2007.

Sandford, Friedenthal; Anlan, Moore; Rick Steiner: OMG Systems Modeling Language (SysML). Published and used by INCOSE and affiliated societies, 11 July 2006.

Schmidt; Bannon: Taking CSCW seriously. In Computer Supported Cooperative Work (CSCW), Springer Netherlands Volume 1, Numbers 1-2 / March, 1992, pages 7-40

Soubie ,J. L.: Modelling in Cooperative knowledge based systems. Third International Conference on the Design of Cooperative Systems COOP'98, Cannes, France. INRIA, France, p. 45–48, May 1998.

Terssac (de) G. ; MAggi B. : Autonomie et conception. In TERSSAC (de) G. et FRIEDBERG E. (Eds), Coopération et conception, Toulouse, Editions Octarès, 1996.

Walker, K.: Collaborative power: Collaborative Processes and Semantic Emergence of Power. 3 rd International Conference on Critical Management Studies. Stream: Communication and Collaboration, 2006.