Context-Awareness in Group Work: Three Case Studies

P. Brézillon*
M. R.S. Borges**
J. A. Pino***
J.-Ch. Pomerol****

LIP6, Université Pierre et Marie Curie, 8 rue du Capitaine Scott – 75015 Paris, France
*Email: Patrick.Brezillon@lip6.fr
****Email: Jean-Charles.Pomerol@lip6.fr

**NCE, Universidade Federal do Rio de Janeiro, CP 2324, 20001-970 Rio de Janeiro, Brazil
Email: mborges@nce.ufrj.br

***DCC, Universidad de Chile, Casilla 2777 Santiago, Chile
Email: jpino@dcc.uchile.cl

Abstract

There are advantages of using the concept of context in the area of Computer-Supported Cooperative Work. However, it was not the case until now. For example, never the term of awareness has been used in this area with an explicit association to context. We discuss in this paper the interest to make context explicit in three case studies in the domain of collaborative works. In particular, we propose a framework to understand how context and awareness are related together and to other concepts used in group works, such as user interface and storage. The proposed framework is used to consider groupware systems presented in this paper from the context perspective and to eventually obtain some insight on possible improvements for users.

Keywords

Decision making, awareness, context, groupware

1. INTRODUCTION

The terms context and awareness are used together in the area of CSCW, sometimes meaning similar ideas, often complementing each other, but also as quite different and conflicting concepts (Dourish, 2001). The concept of context by itself presents multiple definitions. One reason is that this concept is not perceived in the same way according to the domain (Brézillon, 1999; Dourish, 2001). Indeed a search on the web collects around one hundred and fifty definitions from about twenty different domains (Bazire et al., 2003). However a rapid analysis allows to point out that all definitions can be assembled around six questions: Who? What? When? How? Where? and Why?

Now, context is an active field of research, with is a series of interdisciplinary and international conferences on modeling and using context since 1997 (see also the web site of this community at http://context.umcs.maine.edu/). This series of conferences aims to deal with different aspects of the concept of context at the highest level of knowledge and reasoning. However, this approach rarely considers practical aspects of context in real-world applications such as collaborative work.

In parallel, one observes the emergence of a new community concerned with the design and development of context-aware systems and applications. In this approach the focus is on physical elements of the context as user's location, time, weather, etc. all information that can be drawn from sensors and directly used in applications. This branch is strongly related to what is called ubiquitous computing, pervasive computing, etc. However, users are only considered through rough and static models, and the dynamic of the context is limited to changes in the physical parameters. Moreover, one considers the context of one user, not the context of a collaborative work with several users.

When one reads articles in the CSCW forum, a number of issues appear related, directly or indirectly, to the concept of context. Awareness mechanisms and awareness information are the common terms to deal with context in groupware. The group memory also combines both context and content information, sometimes without an appropriate relationship between them. We believe that many of misunderstandings are caused by the lack of explicit recognition and representation of the notion of context and its explicit association with other

elements of groupware systems. This situation is particularly important for groupware where we must deal with several contexts at different granularities: The context of the group (why this group is constituted), the individual contexts of the members (e.g. their technical origins), and the context of the project (e.g. the product to be built by the team).

We claim that context and awareness should not be seen as separate concepts. Instead, they should be considered as paired to each other in group work application domains. Contextual knowledge describes a situation, the way in which to use elements in a group work, including the events that are treated by the group. Awareness is one of the groupware mechanisms aimed at providing context to group members. Consider a session in a cooperative editor, for example. The part of the text a team member is working on is contextual information that is made available to other members connected to the session through a user-interface item called telepointer. The telepointer is managed by an awareness mechanism that receives the information from the cursor sensor.

The paper is organized hereafter in the following way. Section 2 proposes a review of the context and awareness concepts, focusing on their usage for group work. Section 3 presents three case studies showing the interest of context-awareness in real-world applications. Section 4 ends the paper by a discussion on the lessons learned and the challenges now open by the explicit consideration of context.

2. CONTEXT-AWARENESS IN GROUP WORK

2.1 The notion of context

In real life, a context is a complex description of shared knowledge about physical, social, historical, or other circumstances within which an action or an event occurs. In order to understand fully many actions or events, it is necessary to have access to the relevant context in which actions must be executed and events appear,. For example, the comprehension of the action “opening a window" depends on what is referred: a real window or a window on a graphical user interface (Rittenbruch, 2002).

Three observations on context are relevant. First, our working definition of context is “Whatever does not intervene explicitly in a problem solving but constrains it” (Brézillon, 2003a). However, as the problem solving progresses, the context also evolves from one step of the problem solving to the following one.

Second, one distinguishes between the part of the context which is relevant for the current focus of attention, and the part which is not relevant for it (Brézillon and Pomerol, 1999). The latter part is called *external knowledge*. The former part is called *contextual knowledge* because it has strong connections with the current focus although not directly considered in the focus. According to the focus, a part of the *contextual knowledge* is extracted, assembled, and structured in a *proceduralized context* that is used in the current focus as a "chunk of knowledge" a la Schank. This proceduralization process obeys to the necessity of having a consistent explanatory framework for explaining and anticipating the results of a decision or an action.

Third, the granularity of context may vary (Brezillon, 2003b). In the groupware area, we discuss of the group context, the individual contexts and the project context. Thus, it is possible (1) to identify different types of context, and (2) to organize them in a two-dimension representation, vertically (i.e., depth first) from the more general (top) to the more specific (bottom) and horizontally (i.e., width first) as a heterogeneous set of individual contexts at each level. This aspect will be discussed in the case studies presented hereafter.

2.2 The notion of awareness

The concept of awareness has been widely used in CSCW research and applications. A well-known definition by Dourish and Bellotti (1992) states that awareness is a*n understanding of the activities of others, which provides a context for your own activity.* Furthermore, they say that this context *is used to ensure that individual contributions are relevant to the group’s activity as a whole, and to evaluate individual actions with respect to group goals and progress.* Therefore, it is easy to visualize that awareness is a key activity when the work is cooperatively done by a group of people in a computer-supported environment. Otherwise, there will not be actual joint work, but an incoherent set of isolated pieces.

Awareness on what co-workers have done or are doing is also very important (Gutwin et al., 1996). In some cases, the result of individual work needs to be known by the rest of the co-workers. In others, one needs *meta-information* or *aggregated meta-information* only. Consider the example of a distributed asynchronous discussion on a certain subject being done by a group of people. Every person should read each of the contributions of the other participants in the discussion. Moreover, it is important to provide the reader easy-to-grasp information on which contributions are *new* and which are *unread* from the last session the person logged in (meta-information). Furthermore, the person may also appreciate if the system tells him how many

contributions he has made and how that relates to the number of contributions provided by the other participants (aggregated meta-information). Note again that we find the privacy/awareness tradeoff, since some participants may not wish to have their number of contributions be summed up and presented to the rest of the group. In this case, perhaps a solution would be to provide just statistical aggregated meta-information, such as the average number of contributions provided by the participants. Collazos et al. (2003) have classified various types of awareness in the area of CSCW.

2.3 Context awareness

Context awareness is a fundamental concern of interaction. Much of the research in this area focuses on revealing environmental data that the system can acquire through sensors, including functional qualities of the space and quantitative interpersonal information such as presence of a person. Usually, context-awareness focuses on revealing external activities to the user. However, context is not limited to the immediate environment. Although environmental awareness is essential, it is also necessary to have self-awareness. Context awareness aims at covering both the user and the environment. Awareness supposes that one is able to transforms pieces of contextual knowledge into a proceduralized context at the current focus of attention. In Figure 1, we represent different types of contexts (group context, individual context and project context) at different granularity.

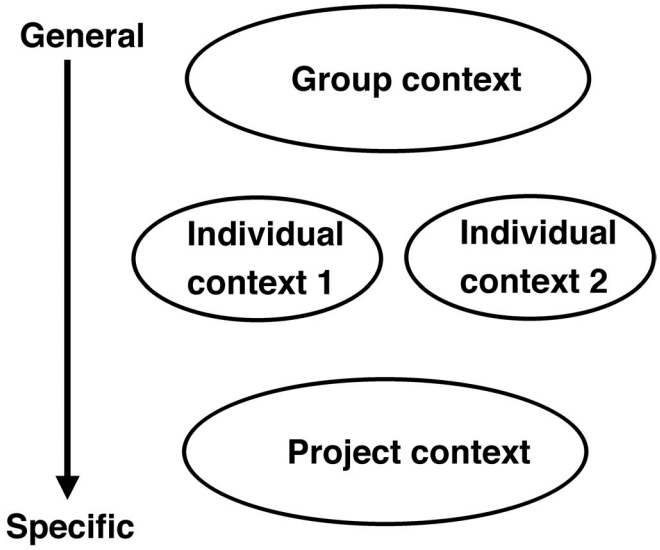

Figure 1**:** The different levels of contexts

In a group, there are two proceduralization processes along the different levels of contexts (see Figure 1). The first process concerns the transformation of the contextual knowledge in the group context into as many different proceduralized contexts as individual contexts. For example, in an European project, there are contextual elements such as the date of the next deliverable. This piece of contextual knowledge is transformed in a proceduralized context by each participant, according to their individual context (e.g. the country in which their work, proximity of holidays in their country, participation in the deliverable, time available until the deadline, etc.) The second process of proceduralization is between individual contexts and the project context on which all the participants work, and try to build jointly the product of a consortium. Here the goal is to assemble contextual knowledge from several sources in different domains into a unique proceduralized context at the product level (the system, platform, etc. to be built). Note that context at one level is not necessarily a subset of the context at the upper level. For example, Brézillon et al. (2003a) observe that in a newspaper firm, each member of the firm has his own, personal and strictly confidential address book. As a part of the individual context, the address book is not shared with others, and thus cannot be considered as an element of the group context.

2.4 Context awareness in groupware

We present in Figure 2 the framework that was proposed in (Brézillon et al., 2004). This section comes back only on our viewpoint on awareness. It essentially presents a *knowledge processing* procedure. Individually, people create knowledge, which is communicated to the rest of the group as well as being presented in a user interface and eventually stored. The *generation* step consists of a person providing information to the group. Of course, this information may be contents for the group's output or related information, such as questions,

suggestions, or procedural proposals. Some of this information is stored, according to pre-established conditions, e.g., "all contents information must be stored".

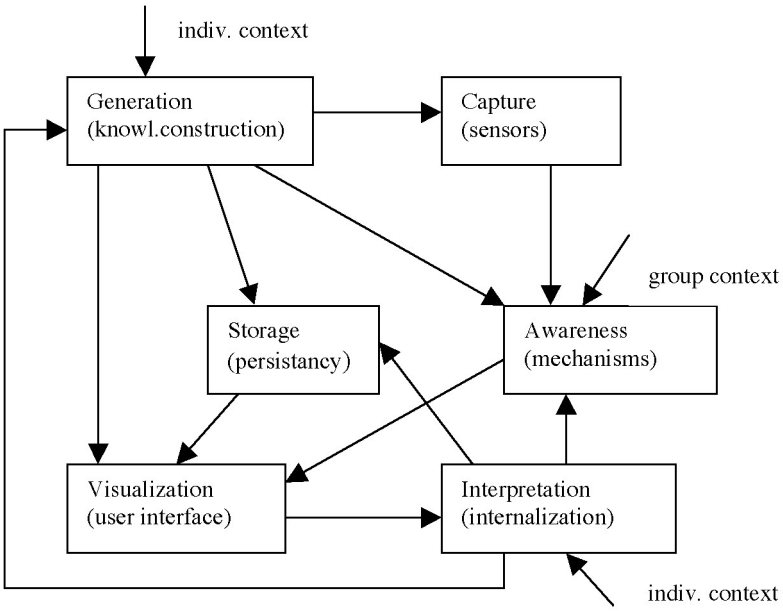

Figure 2**:** Contextual knowledge processing in group work

The *awareness* step is the way of providing information to other participants. There are several inputs to this process. The first is information from the generation step. An example would be a contribution just written by a group member. This information needs to be transformed in some way, perhaps summarized or filtered to make it available to other people. In fact, it takes into account the processing specifications given by individual users. Another type of input comes from the capture step; again, this information will probably be processed to avoid information overload. It also receives information from the storage step. This occurs, for example, when an agent decides to distribute a summary report on recent work in asynchronous systems. Finally, notice there is group context received as an input. This is needed as important information to process the rest of the inputs.

3. CONTEXT AWARENESS IN THREE CASE STUDIES

3.1 Introduction

We illustrate our view of context awareness in three applications in the groupware area: SisPro (Borges et al., 1999a; Naveiro et al., 2002) is a computer environment in project design; SISCO (Borges et al., 1999b) is a meeting preparation asynchronous system aimed at supporting the group discussion that occurs before an actual meeting; and CO2DE (Borges et al., 2003), a cooperative editor that supports multiple versions as a way to deal with conflicting views when building a diagram. These systems do not support context explicitly, although they use several contextual elements to support group work. The goal of this section is to analyze how context, at its several levels, is represented and used in these systems.

3.2 The SisPro project

The objective of this system is to ease collaborative activities (including negotiation) and learning processes, and the development of teamwork competence. Figure 3 presents the general architecture.

For supporting exchanges among participants, the SisPro environment proposes different workspaces that we assimilate to contexts. A first type of virtual-workspace is attached to each participant (individual context). These individual contexts allow participants to manage their specificity to ease communication between participants. A second type of workspace is attached to each project (project context) and is shared by all participants in the project. This workspace is the medium for the negotiation among the participants. The project context represents at each moment the project state that results from the consensus among the various participants. In contextual terms, participants draw from their individual context contextual-knowledge elements to negotiate with others in order to build a shared proceduralized context in the project context. This case study points out the movement between the different levels of context presented in Figure 1.

Each individual context benefits, on the one hand, from external specialist's sources of reference of the team member and, on the other hand, from the team context. The team project has a more general context than the individual contexts that explains why the members met in this project (such as an European project) where partial solutions are negotiated and are new knowledge for the specialist (as a proceduralized context at a previous step of the design). However, the individual contexts are relatively static with respect to the team project that possesses the dynamics required for the progress of the design.

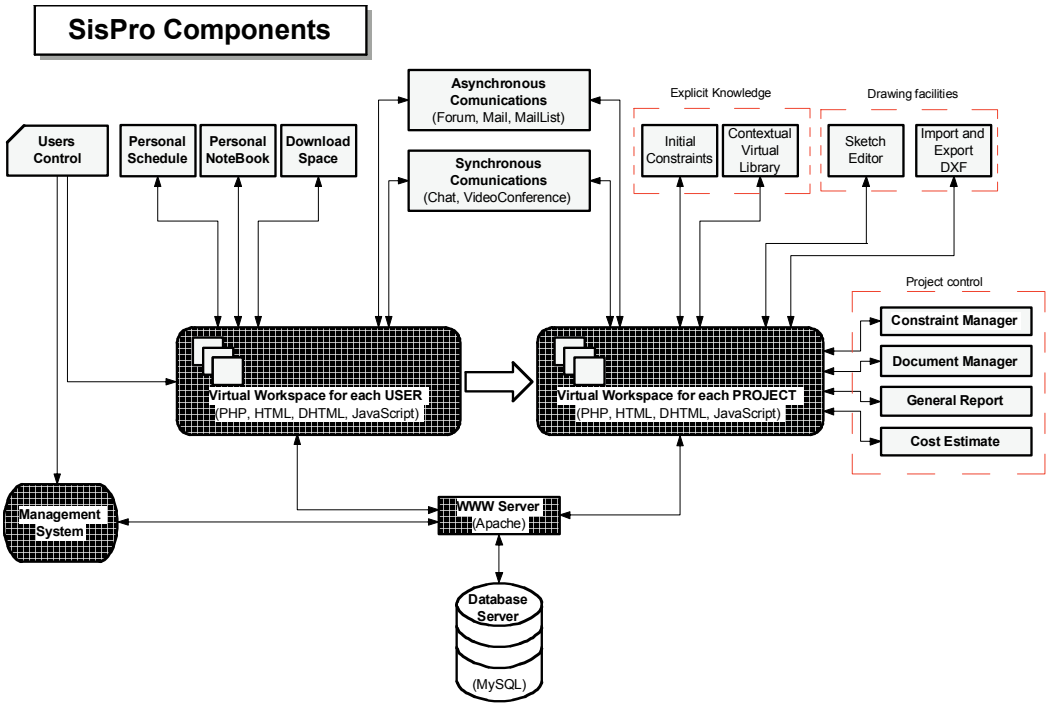

Figure 3**:** The SisPro architecture with the project context and the individual contexts

3.3 The SISCO Example

SISCO (Borges et al., 1999b) is a meeting preparation support system. The meeting participants are selected on the basis of their individual contexts. The selection is based on the contextual knowledge that each participant has about the meeting agenda items, as well as the diversity of individual contexts from which are drawn the contextual knowledge. The sharing of participants' contributions among group participants allows to reduce repetitions and also to increase the quality of the contributions by making explicit other participants' ideas and their connection to the current state of the project. This development of a shared context (such as the project context in Figure 1) promotes the internalization and idea generation processes. At this step, participants must reach a consensus, and this does not mean to have an identical view on the project, but compatible views: a shared context does not mean identical individual contexts.

SISCO provides persistency of contributions to the discussion as well as a support to each individual awareness of the discussion contents. Whenever a member logs in, the system generates a schematic view of the discussion contents, indicating what is new to her. This is a way to keep the contextual knowledge uniform among group members even when they are disconnected from the system during long periods. Therefore, the system provides awareness mechanisms to allow users to update their individual contexts with the group context represented by the set of all contributions. These awareness mechanisms support the transformation from contextual knowledge at one level to the proceduralized context at the level below.

The project context covers as much as possible the wide range of options and arguments related to the agenda items. During the discussion supported by SISCO using an IBIS-like argumentation model, most contributions are based on participants' individual context, thus the authorship provides some hints about the associated context. The system, however, also encourages participants to express views, which might not be of their own, but which are in pace with the group or the task context. In this way the system manages movement of information between the individual contexts and the project context.

Another form of supporting task context is through the definition of roles. When playing a role in SISCO, an individual is given a narrower context with specific awareness mechanisms. For example, the coordinator role

is provided with a *participameter*, a widget that informs the level of participation in the discussion (Borges and Pino, 1999). The participameter provides the coordinator with elements to decide on what to do when, for example, the level of participation in a certain item is low: remind people, promote discussion with some controversial statement or even drop the topic. Figure 4 illustrates the user interface of the participameter.

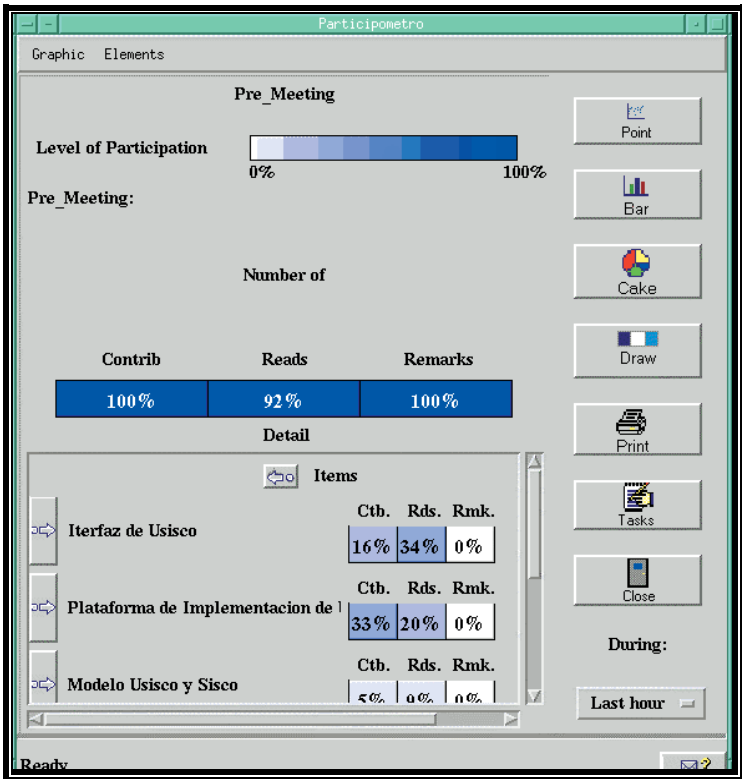

Figure 4: The participameter as an awareness widget

3.4 The CO2DE Example

The CO2DE editor allows to merge individual contexts into a single diagram by providing a synchronous cooperative edition facility and a WYSIWIS interface (see Figure 5), but the editor does not focus on asynchronous-interaction aspect especially. The diagram works as long as the latest group context that is the union of individual contexts (the notion of context is not explicitly treated by CO2DE). The resulting diagram is considered simply such as the result of a group work.

When conflicting views arise on elements of the diagram, most cooperative editors support users to reach a consensus by means of a communication mechanism, e.g., a chat. The goal of this communication mechanism is similar to the development of a shared context in order to make compatible all the interpretations of the participants and to allow the building of a unique proceduralized context. The resulting element is then expressed in the diagram associated with the corresponding discussion.

CO2DE has a particular approach to deal with conflicts. It allows several versions of the diagram to co-exist, and organizes them into a tree to associate each version to its origin, its alternative versions resulting from the conflict and its further decomposition originated from another conflict. In none of these cases, however, the system represents contextual information, for example, what was the conflict and which assumption a version was based on. This contextual information is kept within each individual context and is not saved by the system. If a person wants to understand the rationale behind the creation of a new version, he has to ask its creator.

During the elaboration of the diagram, several versions may co-exist. It is left to participants to solve the conflicts and express the resulting consensus in a single version. One may argue that this is similar to solve the conflicts as they arise. The CO2DE approach has the advantage of allowing users to represent their views in a more comprehensive format, since a single conflict in many cases involves several elements of the diagram. It is like discussing two or more options using the complete picture, instead of discussing element by element. Another advantage is the representation of the work evolution by means of a set of step refined versions.

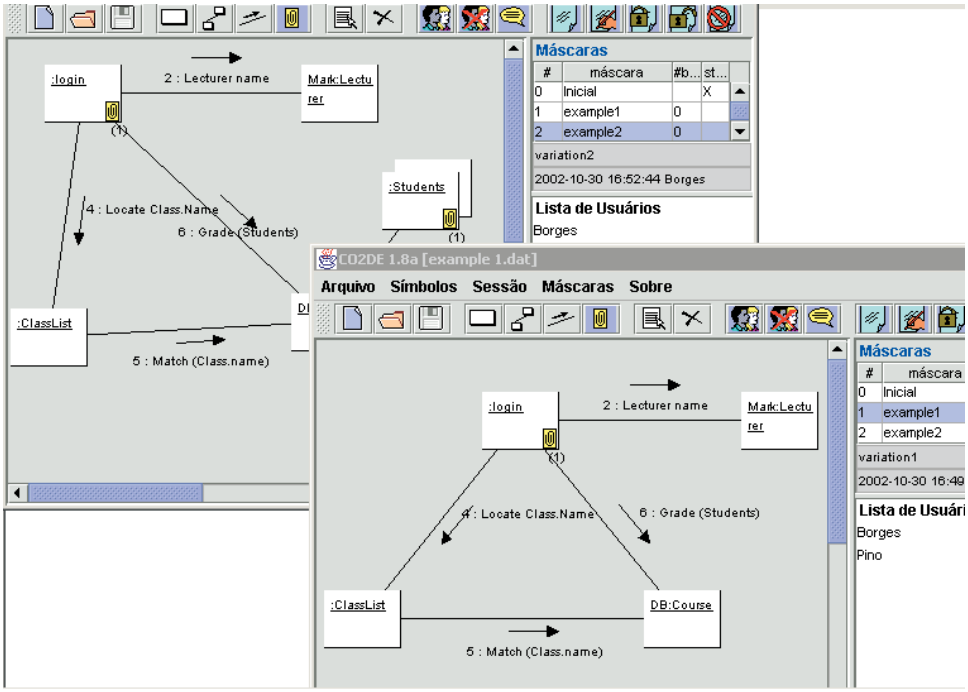

Figure 5: CO2DE user interface

4. CONCLUSION

Fields of context and CSCW largely developed independently. One could think this was not a good idea for groupware designers, who might have taken advantages of research on context. The framework presented in this paper is a first step to narrow that gap by relating the concepts of context and awareness to other terms widely used in real-world applications such as CSCW, user interfaces, automatic capture and storage.

The framework presents group work as a knowledge processing job with some machine supporting activities. This dataflow-type modeling is new, as well as the presentation of context as the knowledge flowing among processing activities. The framework can be applied to get some insight into some groupware designs. In particular, by considering context as knowledge to be applied during the group work, one can have a wider perspective than just focusing on the information provided to users by awareness mechanisms, as illustrated in the previous chapter. Many other groupware designs would probably be possible to analyze from this viewpoint.

Making context explicit is a way to remember, not only the way in which a solution was developed, but also the existing alternatives at the time of the solution building, the existing constraints, etc. Thus, by comparing the context at that time and the current context allows awareness.

A historical context consists of information about projects and tasks already completed. This information is important for the understanding of errors and successful approaches in previous projects to be used in current tasks. It can also be used out of the context of a project to provide insight into working practices and team cooperation. When we make the historical context available, users can access the real reasons on which a decision was made, for example.

During the progress of the building of the solution, one observes that the project context evolves jointly to the solution building, and even individual contexts of the participants are modified. Once all participants agree on a proceduralized context, this later becomes a piece of the solution construction, and each participant retains it in his individual context as a piece of the shared part of all individual contexts. For example, a specialist on a certain subject can propose a solution from his competence field. However, another specialist may introduce constraints which are clear from his own domain but not in the other field. In such a case, the first specialist will modify his context to take into account the fact that the pair (problem, solution) in her domain must be modified becoming the triple (problem, context, solution).

In a collaborative work, participants share experience with others. As a consequence, participants see their individual contexts enriched with contextual knowledge that will be shared with other individuals and found also in their individual contexts. This will lead later to more efficient interaction among them. For example, if

a group must meet regularly in the framework of a project, they will spend the first meeting to establish the frequency of the following meetings, the day of the week, the time, the location, etc. Once all participants have agreed (eventually after negotiation), all these pieces of contextual information will be shared, structured, and assembled together in a coherent whole (the proceduralized context), and they will finish the first meeting by saying only "see you next time" that will be a shared pointer towards a chunk of knowledge containing the date, the location, etc. of the next meeting.

The discussion of these three case studies in the light of context allows to point out different observations that could be the challenges of the next years:

- Different types of contexts must be identified at different levels of generality. However, this scale of generality does not mean that the context at one level is deducible from the individual contexts at the lower level. Each context needs a specific identification.
- The scale of generality can be used recursively. For example, the group context is also the individual context of the group interacting with other groups in an enterprise (a new context above the group context).
- A context contains more general knowledge than lower contexts, and there is a mechanism to transform contextual knowledge at one context into a proceduralized context at the level below. The identification of this situation allows to model knowledge and reasoning in a best way.
- A given context is like a system of rules (constraints) for identifying triggering events and for guiding behaviours in lower contexts. A context-aware system must have means to capture automatically (as much as possible) data, means to interpret that data to draw information from them, and learning capability to take advantage of its experience. This is a first step towards intelligent assistant systems.
- There is a dynamics in the context, and this dynamics corresponds to a movement between contextual knowledge and the proceduralized context. This dynamics is strongly related to the focus of interest and evolves with it.
- The proceduralized context built from the contextual knowledge at the upper level is explicitly used in the current focus of interest. Thus, it is important to consider jointly the focus of interest and its context of occurrence.
- The proceduralized context is used at a current step of the focus. After, it goes back to the contextual knowledge as a "chunk of knowledge" a la Schank. During this process memory organization is improved because knowledge is assembled with its context of use. This means that a system must tackle such an aspect to account for an evolution of users during their interaction, especially through computer-mediated means as for groupware.

REFERENCES

Bazire, M., Brézillon P. and Tijus, Ch. (2003) Eléments intervenants dans le contexte d'une activité finalisée. Symposium "Tâche, Activité et Contexte" (TAC). In: J.M.C. Bastien (Ed.), Actes des Deuxièmes Journées d'Etude en Psychologie Ergonomique – EPIQUE'2003, INRIA, pp. 281-286.

Borges, M.R.S. and Pino, J.A. (1999) Awareness Mechanisms for Coordination in Asynchronous CSCW, Proceedings of the 9th Workshop on Information Technologies and Systems (WITS '99), Charlotte, North Carolina, pp. 69-74.

Borges, M., Naveiro, R. and Souza Fillho, R. (1999a) SISPRO - A Computer Support System for conceptual design in architecture, *Proceedings of the 12th International Conference in Engineering Design*, Munich, Heurista.

Borges, M.R.S., Pino, J.A., Fuller, D. and Salgado, A.C. (1999b) Key issues in the design of an asynchronous system to support meeting preparation", *Decision Support Systems*, 27(3): 271-289.

Borges, M.R.S., Meire, A.P., Pino, J.A. (2003) An interface for supporting versioning in a cooperative editor, Proceedings of the 10th International Conference on Human - Computer Interaction, Crete, Greece (to appear).

Brézillon, P. (1999) Context in problem solving: A survey. *The Knowledge Engineering Review*, 14(1):1-34.

Brézillon, P. (2003a) Individual and team contexts in a design process. Proceedings of the 36th Hawaii International Conference on Systems Sciences, HICSS-36, Track "Collaboration Systems and Technology", R.H. Sprague (Ed.), Los Alamitos: IEEE, CD-Rom.

Brézillon, P. (2003b) Using context for Supporting Users Efficiently. Proceedings of the 36th Hawaii International Conference on Systems Sciences, HICSS-36, Track "Emerging Technologies", R.H. Sprague (Ed.), Los Alamitos: IEEE, CD-Rom.

Brézillon, P. (2003c) Représentation de pratiques dans le formalisme des graphes contextuels. In: J.M.C. Bastien (Ed.), Actes des Deuxièmes Journées d'Etude en Psychologie Ergonomique – EPIQUE'2003, INRIA, pp.263-268.

Brézillon, P. (2004) Representation of procedures and practices in contextual graphs. The *Knowledge Engineering Review* (to appear).

Brézillon, P. and Pomerol, J.-Ch. (1999) Contextual knowledge sharing and cooperation in intelligent assistant systems, PUF, *Le Travail Humain* 62 (3): 223-246.

Brézillon, P. and Marquois, E. (2003) Une approche centrée contexte de l'activité. Symposium "Tâche, Activité et Contexte" (TAC). In: J.M.C. Bastien (Ed.), Actes des Deuxièmes Journées d'Etude en Psychologie Ergonomique – EPIQUE'2003, INRIA, pp.263-268.

Brézillon P. and Pomerol J.-Ch. (2001) Is context a kind of collective tacit knowledge? European CSCW 2001 Workshop on Managing Tacit Knowledge. Bonn, Germany. M. Jacovi and A. Ribak (Eds.), pp. 23-29.

Brézillon, P., Adam, F. and Pomerol, J.-Ch. (2003a) Supporting complex decision making processes in organizations with collaborative applications - A case study. In: Favela, J. & Decouchant, D. (Eds.) Groupware: Design, Implementation, and Use. Lecture Notes in Computer Science 2806, Springer Verlag, pp. 261-276.

Brézillon, P., Pomerol, J.-Ch. and Pasquier, L. (2003b) Learning and explanation in a context-based representation: Application to incident solving on subway lines. In: R. Jain, A. Abraham, C. Faucher and J. van der Zwaag (Eds.) Innovations in Knowledge Engineering. International Series on Advanced Intelligence,

Brézillon, P., Borges, M., Pino, J.A. and Pomerol, J.-Ch. (2004) Context-based awareness in group work. Proceeding of the 17th International FLAIRS Conference, Invited Special Track "Modeling the Real World Through Context," Miami, Florida. (submitted October 2003).

Collazos, C., Guerrero, L.A. and Pino, J.A. (2003) Knowledge construction awareness. *Journal of Student Centered Learning*, 1(2): 77-86.

Dey, A.K., Salber, D. and Abowd, G.D. (2001) A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications, *Human-Computer Interaction*, 16 (2-4): 97-166.

Dourish, P. (2001) Seeking a Foundation for Context-Aware Computing. *Human-Computer Interaction*, 16(2-4).

Dourish, P. and Bellotti, V. (1992) Awareness and Coordination in Shared Workspaces. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work CSCW'92,* Toronto, Ontario, ACM Press, New York, pp.107-114

Gonzalez, A.J. and Ahlers, R., (1999) Context-based representation of intelligent behavior in training simulations. *Transactions of the Society for Computer Simulation International*, 15(4):153-166.

Gutwin, C., Roseman, M. and Greenberg, S. (1996) A usability study of awareness widgets in a shared workspace groupware system. *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW 96)*. ACM Press, New York; pp. 258-267.

Hudson, S. and Smith, I. (1996) Techniques for addressing fundamental privacy and disruption tradeoffs in awareness support systems. *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW 96)*. ACM Press, New York; pp. 248-257.

Moran, T.P. and Dourish, P. (2001) Introduction to the special issue on Context-Aware Computing. *Human-Computer Interaction*, 16(2-4): 87-94.

Naveiro, R.M., Brézillon, P. and Souza, F.R. (2002) Contextual knowledge in design: the SisPro project. Revue Document Electronique. "Espaces Numériques d'Information et de Coopération", C. Simone, N. Matta & B. Eynard (Eds.) Hermès, 5(3-4) : 115-134.

Pomerol J-Ch. and Brézillon P. (1999) Dynamics between contextual knowledge and proceduralized context, in *Modeling and Using Context* (CONTEXT-99), Lecture Notes in Artificial Intelligence n° 1688, Springer Verlag, pp. 284-295.

Rittenbruch, M. (2002) ATMOSPHERE: A Framework for Contextual Awareness, *International Journal of Human-Computer Interaction* 14 (2), 2002, pp. 159-180.

Turner, R M, (1998) Context-mediated behavior for intelligent agents" *International Journal of Human-Computer Studies*. Special Issue on Using Context in Applications, 48(3): 307--330.

Watson, R.T. (1998) Data Management. Databases and Organizations. John Wiley and Sons, Inc., Second edition, New York.

ACKNOWLEDGEMENTS

This work was partially supported by grants from: CNPq (Brazil) No. PROSUL AC-62, Fondecyt (Chile) No. 1040952.

COPYRIGHT