Contextual Knowledge and Proceduralized Context

P. BREZILLON and J.-Ch. POMEROL

LIP6, Case 169, University Paris 6, 4 place Jussieu, 75252 Paris Cedex 05, France
Tel: +33 1 44 27 70 08 - Fax: +33 1 44 27 70 00
E-mail: {Patrick.Brezillon, Jean-Charles.Pomerol}@lip6.fr

Abstract

Based on our experience in the development of an interactive system for the control of subway lines, we set forward some notions about various forms of context and about context sharing. In effect, context plays a crucial role especially for incident solving on a subway line. The analysis of operators' behavior doing incident solving proved that context takes three different forms, namely external and contextual knowledge and proceduralized context. Moreover, the paper discusses the dynamics between these three kinds of context.

Keywords: Context, Contextual knowledge, Proceduralized context, external knowledge.

Introduction

Many authors raised the issue of context use in interactive systems (Bainbridge 1997; Hollnagel 1993). From our practical experience and a recent implemented system for the interactive control of a subway line, we propose some ideas about context. More precisely, we focus on the problem of the move from contextual knowledge to proceduralized context when the focus of attention moves.

Even before acquiring contextual knowledge, the designer faces the representation problem and a problem of frontier (Brézillon and Pomerol 1999.) The second problem arises because it is unlikely to represent everything that may influence a system, including contextual data with very small or rare influence. The designer can be tempted to exclude weakly influential information but he must capture all the knowledge that may be consequential for operations.

In the paper quoted above, we illustrated our ideas by refering to the SART project (French acronym for support system for traffic control) that aims at developing an intelligent decision support system to help the operators who control a subway line to solve incidents that occur on the line. An extended presentation of SART can be found in the paper of Brézillon *et al.* (1997).

This paper is organized as follows. In Section 2, we review the different views on context in the literature and present and illustrate our approach of context, pointing out three kinds of context. Section 3 discusses the dynamics between these three kinds of context, and context sharing between agents.

Different kinds of context[1]

Various views about context in the literature

There is already an abundant literature on context. From a review of the literature, Brézillon (1999) shows that context plays an important role in many domains, especially for activities such as foreseeing context changes, explaining unanticipated events and helping to handle them, and aiding to focus attention. Adopting an engineering view, we define context as the set of all the knowledge that can be evoked by a human being facing a situation, assuming that he has an unlimited time to think about it.

Moreover, context possesses a time dimension that raises some problems in modeling. Some have suggested that context is related to the interactions among agents, as opposed to context as a fixed concept relative to a particular problem or application domain (Maskery and Meads 1992). This means that without interacting agents, there would be no context. Context appears as a shared knowledge space. However, each entity involved in an interaction has its own context, which may or may not be consistent with some parts of the contexts of the others.

In a knowledge engineering setting, it has been argued (Grant 1992), that the term 'context' has some features in common with scripts (Schank and Riesbeck 1989), frames or schemata as developed in human cognition. The context here is a candidate for something that is stored in long-term memory, and recalled as a whole, as a viable unit of a task appropriate to some step in a decision making.

Contextual knowledge and proceduralized context

In fact, it is difficult to define the concept of context without considering the people involved in a situation because, at first glance, context involves knowledge that is not explicit. This 'explicitness' depends on the actors. Some common knowledge is implicit but well-known, for example the fact that it is easier to organize emergency operations in a station than in a tunnel. When the reasoning yields this type of knowledge, it is easily

[1] Information on the SART project is available at http://www.lip6.fr/SART

proceduralized and becomes an implicit part of the reasoning that can be elicited by knowledge engineers and finally included in the operation model.

The second fact is that each person involved uses a large amount of knowledge, different from one person to another, to picture the situation. We can define the **contextual knowledge** as all the knowledge that is relevant and can be mobilized to understand a given situated decision problem. By "situated" we mean in given, dated, well specified circumstances (Clancey 1991). Contextual knowledge is evoked by situations and events, and loosely tied to a task or a goal. However, when the task becomes more precise (e.g., an event occurs), a large part of this **contextual knowledge can be proceduralized** according to the current focus of the decision making. Although the contextual knowledge exists in theory, it is actually implicit and latent, and is not usable unless a goal (or an intention) emerges.

In our definition, contextual knowledge is a sub-part of the overall context (see Figure 1). Thus, the rest of the context, which is not relevant for a given situation, is called **external knowledge**.

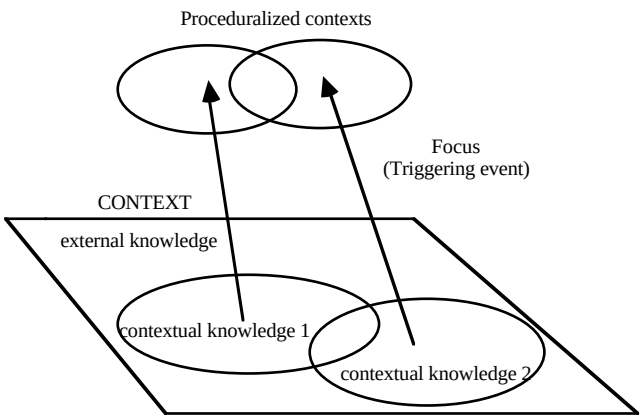

Figure 1: Different types of context

The proceduralized context is a part of the contextual knowledge that is invoked, structured and situated according to a given focus and which is common to the various people involved in decision making. The proceduralized context may be compiled but can generally be elicited with the usual techniques of knowledge acquisition.

At a given step of a decision making, one has: proceduralized context that is knowledge commonly known by the actors of the problem and directly (but tacitly) used for the problem solving, contextual knowledge that is knowledge not explicitly used but influencing the problem solving, and external knowledge that is knowledge having nothing to do with the current decision making step, but known by many actors of the problem.

There are some similar views in the literature. For example, Anderson's theories (1983, 1993) assume that knowledge is first acquired in a declarative form which encodes the basic facts and examples found in the instructions (our contextual knowledge). Once acquired, this knowledge is used by general problem-solving rules to create rules specific to a given context (our proceduralized context). There is another parallel with the *global-context* and *local-context* in (Carenini and Moore 1993). *Global-context* indicates the current topic under discussion. It contains the place in the dialogue history where this topic was begun. *Local-context* points to the most recent utterance. As regards Schank's theory, assuming that context is the set of all the possible stories, the case corresponds approximately to our proceduralized context while contextual knowledge is the set of paradigmatic cases.

In the SART application, context is the sum of all the knowledge, which is known by operators, regarding the controlling task at large. Contextual knowledge constitutes the context of the incident solving (and the operator's decision making). It is composed of all what the opeartor can learn or be informed in the control room:

- his colleagues that watch the operator and intervene from time to time by a question or a recall to the operator,
- information from outside (from the place of the incident and out of the subway), and
- the visual monitoring and control panel, which is visible by all operators in the control room, acts as a kind of shared memory.

At each step of the incident solving (and the operator's decision making), a part of the contextual knowledge is proceduralized. For example, at the information "Train too seriously damaged to be repaired", the piece of contextual knowledge "the damaged train may be repaired or evacuated" enters the proceduralized context to become "Train evacuation." As the incident solving progresses, new items of contextual knowledge are considered, and, thus the proceduralized context evolves.

Note that the incident solving is a collective solving of the incident relying on shared contextual knowledge. This leads to a kind of decontextualization of the whole process. Scott (1997) observe the same phenomenon with the globalization process.

A representation of proceduralized context

Brézillon *et al.* (1997) discuss of the mandatory procedures imposed on drivers. Procedures are acquired from operator's experience during similar incidents and fixed by the company. An implicit piece of knowledge is that travelers are safer in a station than in a tunnel. Such a piece of knowledge, which is not necessarily expressed, result in more or less proceduralized actions that are compiled as the proceduralized context (e.g., procedures.) Very often many pieces of proceduralized context are structured together in comprehensive knowledge about actions.

However, there is no procedure for complex incidents, but a set of procedures for solving parts of the incident. For example, when a train cannot move in a tunnel, there are procedures for traveler evacuation at the nearest station, to clear the damaged train by another train, etc. Some procedures are sequential, but others may be accomplish in any order compared to some ones. For example, when a train must push a damaged train, both trains must be empty but the order in which travelers of the two trains are evacuated is not important and depends mainly on the context in which are trains. What is important is that the two actions must be achieved. As a consequence, there are as many strategies for solving an incident as operators because cases that are similar in one context may be totally dissimilar in others, as already quoted by Tversky (1977).

A representation of contextual knowledge

Contextual knowledge implicitly delimits the resolution space. It is is always evoked by a task or, in our case, an event. Contextual knowledge does not focus on a task or on the achievement of a goal but is mobilized, even though it has not yet been proceduralized for use.

For instance, in a normal situation, the control operator faces the following concern:

F0: the 'normal' focus of attention is to see that schedules and intervals between trains are respected.

This task F0 can be regarded as routine and does not require special attention. Nevertheless, contextual knowledge about control is involved:

C0: the normal context associated with F0 involves:

k1: type of day (*e.g.*, working day or weekend),

k2: period of the day (morning, afternoon, evening),

k3: traffic state (rush hours, off-peak hours),

k4: the section load (very busy, few people),

All these pieces of knowledge describe the environment of the problem with which the following pieces of knowledge are associated:

k5: the interval between trains according to the situation,

k6: the stopping time in stations, etc.

Assume now that an incident occurs on the subway line; the pieces of knowledge k1 to k4 are (or should be) immediately invoked. This results in k5 and k6 being invoked too. All these pieces of contextual knowledge become a part of the proceduralized context in which the incident is resolved; contextual knowledge appears back-stage, whereas the proceduralized context is front-stage in the spotlights. It is noteworthy that, as far as engineering is concerned, only the proceduralized context matters, but contextual knowledge is necessary because this is the raw material from which proceduralized context is made.

Turner (1998) develops some similar ideas for the context-sensitive reasoning of underwater vehicles. Context is mainly considered as a way to cluster knowledge for efficiency search, for representing counter-factual or hypothetical situations, for circumscribing the effects of particular actions to particular situations, and for directing an agent's focus of attention to salient features of a situation.

Move between proceduralized context, contextual and external knowledge

We think that it is not only important to understand the dynamics of planning and action but also the dynamics of knowledge management. This is a twofold phenomenon that consists of focusing on some stimuli and, on moving contextual information from back-stage to front-stage. Interaction between agents appears to be a privileged way for moving a contextual knowledge into and out of the proceduralized context.

A Dynamics of the proceduralized context

At the interaction level, making context explicit enables knowledge to be shared with others. For example, in the case of "Sick traveler in a train," the change from the step "Answer the alarm signal" to "Stop at the next station" was surprising to us (as knowledge engineers). The triggering of the alarm signal implied for many years an immediate stop of the train, even in a tunnel because an alarm signal needed immediate attention. To explain the skipping of the action "Answer the alarm signal", the operator said that, based on company experience, they have decided to stop only at the next station for several reasons. Some reasons are easy to understand (*e.g.*, rescue is easier in a station than in a tunnel). Other reasons needed additional knowledge (*e.g.*, a traveler may be sensible to claustrophobia in a tunnel). Once we were able to gather all these reasons into a coherent picture, we understood the change.

Figure 2 represents how the proceduralized context is built from contextual knowledge during an interaction between two agents. The interaction context contains proceduralized pieces of knowledge in the focus of attention of the two agents. These pieces of knowledge are extracted from the contextual knowledge of each agent; they are structured jointly by both agents and result in a shared knowledge. Generally, the first utterance of an agent gives a rule such as 'Stop at the next station' if the alarm signal is triggered. Then on the request of the second agent, the first agent may add some pieces of knowledge related to his first utterance. If this knowledge chunk belongs to the common part of the contextual knowledge of the agents, the pieces are integrated into a mutually acceptable knowledge structure, and the knowledge structure may then be moved to the shared proceduralized context.

Thus, the proceduralized context contains all the pieces of knowledge that have been discussed and accepted (at least made compatible) by all the agents, and these pieces of proceduralized context then become part of the contextual knowledge of each agent. Even if they do not remain within the focus of the proceduralized context, they may be recalled later as a whole because shared. For example, "Stop at the next station" is such an example in

the solving of "Sick traveler in a train."

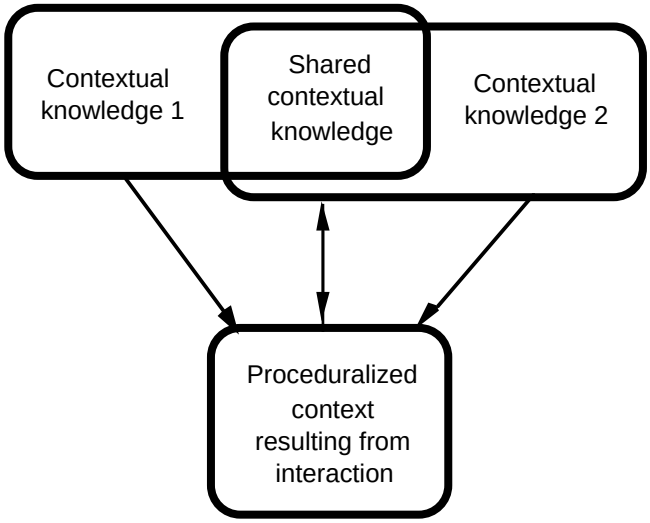

Figure 2: A representation of the interaction context

Dynamics of the decision making process and its contextual dependency

Operators solve an incident by choosing a scenario, which is a sequence of actions conditional on possible events. The choice of a scenario greatly relies on contextual knowledge. One operator said us: "When an incident is announced, I first look at the context in which the incident occurs." The reason is that operators want to have a clear idea of future events; the purpose of this look-ahead reasoning (Pomerol 1997) is to reduce, as far as possible, the uncertainty in the scenario. The problem for operators is that many scenarios are similar at the beginning and then diverge according to the context. Thus, a scenario is a sequence of actions intertwined with events that do not depend on the decision makers but that result in a limitation of their actions. For instance:

Focus of attention:

Removal of a damaged train from the line.

Contextual information: Level of activity on the line.

Action: Lead the damaged train:

- to the terminal if the line activity is low
- to the nearest secondary line if line activity is high.

As most of the contextual elements may intervene in several scenarios (*e.g.*, traffic activity, position of the next train), operators prefer to take them into account as soon as possible to get a general picture of the best path to choose. At this step, contextual knowledge is proceduralized and in the meantime operators postpone action. The main objective is to eliminate event nodes. By grouping together a set of actions in a macro-action, operators hope to make the following step easier. An example is given in (Brézillon and Pomerol 1998.) Macro-actions are a kind of compilation, originated from experience, of several actions. In this compilation, a part of the knowledge on each action becomes implicit in the proceduralized context.

Without entering the details, *macro-actions are a way to proceduralize contextual knowledge* and to introduce modularity in the diagnosis process by managing different modules accomplishing the same function in different ways according to the context. However, action postponement is not always possible, and it is preferable to look for pruning the decision tree in some situations (Brézillon, Pomerol and Saker 1998).

As said previously, RATP (the French metro company) has established procedures for incident solving on the basis of their experience. However, each operator develops his own practice to solve an incident, and one observes almost as many practices as operators for a given procedure because each operator tailors the procedure in order to take into account the current proceduralized context, which is particular and specific. This is observed in a number of companies it is a way of getting the result whatever the path followed. The validation of such unwritten rules is linked more to the result than to the procedure to reach it.

Conclusion

In this paper, we introduced three types of context by refering to the SART application for the subway control, we mainly focused on the decision making process during incident solving. We considered three parts in context, namely external and contextual knowledge, and proceduralized context. External knowledge is the part of context that has nothing to do with the incident solving at a given step of the incident solving. Contextual knowledge is knowledge that does not interven directly in the decision making process but constrains it. Procedurelized knowledge is a part of contextual knowledge that is structured to be used at a given step of the decision making.

Another result that we discussed in the paper is the dynamic among these three part of context when the decision making progresses from one step to the following one. Some pieces of proceduralized leave the interaction context to become shared contextual knowledge. Other pieces of contextual knowledge (and eventually from the external knowledge) enter the proceduralized context.

These two aspects, static description of context and the dynamic of context are modeled in our application at the level of the domain knowledge and included in the reasoning for incident solving.

We definitely believe that one cannot apprehend contextual issues in a static framework and that expliciting and sharing of contextual knowledge is key process for addressing and understanding context problem.

Acknowledgments: The SART project enters a convention between two universities, the University Paris 6 (France) and the Federal University of Rio de Janeiro (FURJ,

Brazil), a contract between the University Paris 6 the company of the subway in Paris (RATP), and another contract between the FURJ and Metrô that manages the subway in Rio de Janeiro. Grants are provided by COFECUB in France and CAPES in Brazil. We also thank J.-M. Sieur at RATP, and C. Gentile, L. Pasquier, I. Saker and M. Secron, Ph.D. students working in the SART project.

References

Anderson, J. R. 1983. *The architecture of cognition*. Cambridge, MA: Harvard University Press.

Anderson, J. R. 1993. *Rules of the Mind*. Hillsdale, NJ: Erlbaum

Bainbridge, L. 1997. The change in concepts needed to account for human behavior in complex dynamic tasks. *IEEE transactions on Systems, Man and Cybernetics*, 27, 3, 351-359.

Brézillon, P. 1999. Context in human-machine problem solving: A survey. *Knowledge Engineering Review*, to appear.

Brézillon, P., and Pomerol, J.-Ch. 1998. Using contextual information in decision making. In: Chapman & Hall. Widmeyer G., Berkeley D., Brezillon P. & Rajkovic V. Eds.: *Context-Sensitive Decision Support Systems*, pp. 158-173.

Brézillon, P., and Pomerol, J.-Ch. 1999. Contextual knowledge sharing and cooperation in intelligent assistant systems. *Le Travail Humain* (to appear).

Brézillon, P.; Gentile, C.; Saker, I.; and Secron, M. 1997. *SART: A system for supporting operators with contextual knowledge*. Paper presented at the *International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT-97)*. Rio de Janeiro, Brasil, Feb.

Brézillon , P.; Pomerol, J.-Ch.; and Saker, I. 1998. Contextual and contextualized knowledge and application in subway control. *International Journal of Human-Computer Studies*, 98, 357-373.

Carenini, G., and Moore, J. D. 1993. Generating explanations in context. *International Workshop on Intelligent User Interfaces*, Orlando, Florida.

Clancey, W.J. 1991. Bookreview of Israel Rosenfeld, The invention of memory : a new view of the brain. *Artificial Intelligence*, 50, 241-284.

Grant, A.S. 1992. *Mental models and everyday activities*. Paper presented at the *2nd Interdisciplinary Workshop on Mental Models*, Cambridge, UK, March, 94-102.

Hollnagel, E. ed. 1993. *Human Reliability Analysis, Context and Control*. London: Academic Press.

Maskery, H., and Meads, J. 1992. Context: In the eyes of users and in computer systems. *SIGCHI Bulletin*, 24, 2, 12-21.

Pomerol, J.-Ch. 1997. Artificial Intelligence and Human Decision Making. *European Journal of Operational Research*, *99*, 3-25.

Schank, R.C., and Riesbeck, C.K. eds. 1989. *Inside case-based reasoning*. Lawrence Erlbaum Associates, chap. 1-2.

Scott, C. 1997. The Proceduralization of Telecommunications Law: Adapting to Convergence. Work In Progress, 1997 (3), *The Journal of Information, Law and Technology, JILT*. <http://jilt.law.strath.ac.uk/jilt/wip/97_3scot/scott.TXT>

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

Tversky, A. 1977. Features of similarity, *Psychological Review* 84(4): 327--352.