

# Operational Knowledge Representation for Practical Decision Making

Jean-Charles POMEROL, Patrick BREZILLON  
and Laurent PASQUIER  
*LIP6, Case 169, University Paris 6*  
*Jean-Charles.Pomerol@lip6.fr*

## Abstract

*In various industrial fields, the operators use pre-designed procedures either to solve problems or for troubleshooting. In the Parisian subway, such procedures exist since 1900. However, these procedures are not always exactly suited to the case at hand, and the operators generally prefer to customize a solution than to rely on fixed procedures. For the design of an “intelligent” assistant system aimed at supporting operators’ decision in subway control, we have modeled operators’ activity and know-how. As a result, we introduce the contextual graph model, which appears as a simple solution to describe and manage operational decision making.*

## 1. Introduction

In high technical and heavily dynamical regulation processes, operators who are responsible for the process control have to rapidly react. If an incident occurs, they have only few minutes to forge a representation of the issue, gather information on the situation, analyze the incident and undertake the correcting actions. To ease their job, many companies have established fixed procedures. Initially, general procedures have been designed to provide operators with a secure reference for incident solving. However, these general procedures forget the contextual dimension of the case at hand. Nowadays, companies are diversifying these procedures by introducing increasingly contextual considerations. This operation multiplies and specializes the available procedures for each type of incident. This rises the question of representing and managing these very numerous, contextual-dependent procedures.

Actually, very often the operators prefer to replan their action in real time rather than to rely on the procedures based on company's experience, this is due to two main reasons. Firstly, the selected procedure is not always perfectly adapted to the situation at hand and can lead to improper actions or sub-optimal incident resolution

strategies. Secondly, if the operator relies on a procedure, he can miss some important facts and notice them too late to adequately solve the incident. Procedures are then used as frames to construct a genuine strategy tailored to the specificities of a given situation. Such practices are based on operational knowledge and are shared by the operators. This well-known phenomenon was studied by Leplat [14] who distinguished between the prescribed and the effective tasks. The former is the task conceived by the “method office” of the company, and the latter is the effective task which is executed by the employees. The effective task corresponds to the operational knowledge of the workers. This knowledge results from their experience and social interaction, this phenomenon is known after Brown and Duguid [4] under the name of community of practice, see also Wenger [30].

For the design of precise procedures, it seems important to gather operators' practices and to analyze their operational knowledge. Using computers, we can manage large databases and data structures, and thus design decision support systems based on real practices. The SART project [3, 18] aims at the design and development of such a decision support system for the operators in charge of the control of a subway line. This project is based on the interaction between the operator and the system and will ease their mutual comprehension. For this purpose, we tried to mimic the system reasoning on the operators' one. Thus, we needed to analyze the operational knowledge used by operators and to record it in an adapted structure which can be easily understood by operators and efficiently used by the computer.

This paper presents the results obtained on the basis described in Brézillon and Pomerol [2]. We present, in Section 2, general information about the application field (the Parisian subway and its control) and our results about activity analysis. We present our model of representation of the operational knowledge in Section 3. This representation is based on decision trees and we explain the structural modification we introduced to adapt these decision trees to our particular domain.

## 2. Operational knowledge

### 2.1. Parisian subway representations

The Parisian subway is a dense underground railway network and a high technical transportation system. A line can be represented according to several points of view:

1. One can see it as a succession of **stations** and **interstations** (interstations are the rail track portions between two successive stations). This is mainly the travelers' view of the line.
2. A second viewpoint is based on electrical supplies. The power needed for all the trains on the line is too high to be supplied by a unique **rectifier sub-station**, thus, several rectifier sub-stations power the line. This permits to divide the track into several **sections**, each of them being powered by at least one sub-station, so that each section is independent from the others from an electrical point of view. For practical reasons, each section is also made of **sub-sections** to reduce the impact of an incident on the traffic.
3. There is a third view of the line according to the organization of the regulation [11]. Two main classes of operators regulate a line: first, the **local control point** agents (LCP agents) who manage the trains and their departure time, second, the **centered control room** (PCC in French) agents, which we call here operators, who are responsible of the traffic supervision and incident solving. The two classes of agents are working at different places and communicate by telephones. Operators at PCC can also communicate with train drivers, station managers and exploitation supervisor (himself connected to emergency services).

Operators alternatively use these three descriptions. When the issue principally concerns traveler transportation, the first description is preponderant. When a technical problem or a located long-lasting incident occurs, operators choose sectioning and sub-sectioning to limit the impact of the issue on the residual traffic. If trains on the line are delayed by the incident, operators communicate the information to the end station (third aspect). Most incidents need a combination of the three representations to be solved: the operators keep in mind that (1) travelers' security is an important contextual factor that constrains incident solving, (2) the consequences of an incident should perturb as few as possible the traffic, and (3) appropriate actions (*e.g.* train redirection) is intended to maintain as far as possible the

regularity of the transport, but interfere with the end station agents' work.

### 2.2. Parisian subway control organization

For each line, there is a principal terminus and one or more secondary terminuses, each of them having a local control point controlling the departure of the trains. The LCP agents have also to manage the traffic in the terminus area, to choose which train will start, to order the departure according to the theoretical timetable and to adapt it to the actual conditions.

A particularity of the Parisian subway is that the PCCs of all the lines (except the new line called METEOR which is entirely automatic) are in the same room, so that the operators of several lines can solve together an incident. Moreover, the operators are organized in several teams and two turnovers are defined, one based on 7 days, the other on 10 days. Thus, all operators rapidly share each new experience. The operators form a community of practice [4] or a community of interaction [16]. This is important to explain the construction of a shared knowledge and a collective resolution of the problem. Hereafter, we consider the problems from the viewpoint of the operators. The PCC is remote from the line and the only information available to operators is indirectly available by telephones and displayed on a synoptic panel of the line. Operators have to mentally construct a model of the local situation to make their decision in the spirit of what Clancey [7] proposed for diagnosis seen as the building of a situation-specific model.

The operator of each subway line is assigned for one part of the day to the corresponding **control console** that allows to cut the power in any section, to stop trains at each station, to communicate with train drivers with high frequency telephones, terminus agents, station agents, local operators or exploitation supervisor with automatic telephones. The operator faces a large **synoptic display** (called **TCO** at the RATP) representing the line, the sectioning and sub-sectioning, the stations and the train position. The TCO also controls some commands such as energy commands and switching commands.

Thus, the actions available to operators concern mainly train regulation (delaying), train redirection, section and sub-section power cutting and supply, or waiting for information, event or action from local or external agents. The operators play an important role as two-way communication channel, dispatching information from local agents (drivers, station agents, end station agents or local-situated executives) to exploitation

and line executives. All the available information is relayed by the operators. As a consequence, operators are the coordinators of all the people solving the incident and the managers of the necessary resources.

When an incident occurs, the operator responsible of the concerned line becomes the “incident manager” and the operators who may possibly help him, are the assessors. The incident manager stays at the control console, responding to phones, controlling the trains and making decision. Assessors help him on several points: for all incidents needing line power control, an assessor stays at the TCO to cut or reestablish the power on sections or sub-sections and for possible train redirections; for more important incidents, a second assessor observes the activity, advises the incident manager and take notes on the resolution (the time of the actions and events, the train number and location of train redirections...). When the incident is solved, the incident manager writes a report containing incident description and the actions undertaken to solve it.

### 3. Operational knowledge representation

Operational practices are difficult to model: firstly they are numerous, secondly they are often implicit within the community of practice [4] and strongly linked one to each other and thirdly the main distinction among them is the context in which these practices are applied. Moreover these practices are often dynamically constrained in sequences of actions. To gather and study operational practices, we record the incidents as a set of characteristics, including context description and the action sequence applied to solve them. With this feedstock, we constructed an adapted representation to collect and organize this type of knowledge for reuse purpose. In this section, we present the models underlying our representation.

#### 3.1. Importance of the contextual dimension and of the dynamics of context

Recently, for the sake of engineering applications, we have tried to understand and model the role of context in reasoning [22]. To do so, we worked on the control of a subway line [19]. Starting from these observations, we defined **contextual knowledge** as the part of the context that is relevant in a given situation for a given operator [2]. This may be seen as the subset of the context in which the operator can find every chunk of knowledge for reasoning about or interpreting and explaining the situation. The complementary part in the context is called external context.

Brézillon and Pomerol defined also the **proceduralized context** as the proceduralized part of the contextual knowledge, *which is considered explicitly, with causal and consequential links, at a given step of the problem solving* [2]. Indeed, the operators do not consider at each moment all the contextual pieces of knowledge during an incident solving. The proceduralized context is the part of the contextual knowledge on which the operators are focusing on at a given step of the incident solving (see Figure 1). This notion is relative to each operator, to the current situation and to the moment at which the operator is working. The proceduralized status of a chunk of context is not permanent since a contextual piece of knowledge is proceduralized when the operator focuses on it and goes back to its initial status of back-stage contextual knowledge when it is no longer mobilized in the reasoning. The process of proceduralization is close to the idea of “externalization” [16]. Externalization describes the way how a mobilization of some tacit knowledge changes it into explicit one.

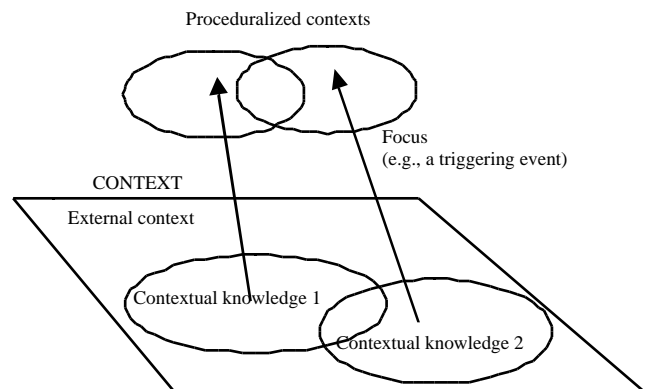


Figure 1. Three types of context

#### 3.2. Computer science approaches

Artificial intelligence develops, for several years, formalisms for operational knowledge representation. Here we present some approaches. In an expert system-like representation, knowledge is gathered as production rules. These rules are pieces of knowledge of the form “if *preconditions* then *conclusions*.” They are recorded in large rule bases difficult to update. The rules are structured chunks of knowledge, which are easily understood (by the domain experts). However, the lack of structure of the rule-base impedes the comprehension (even for the experts of the domain) and the maintenance of the knowledge.

Some works have been done on rule-bases structuring, namely on the splitting of the rule bases into several rule

packets, each containing a subset of rules applied to solve a specific sub-problem [2]. Clancey [7] proposed to add screening clauses to the precondition part of the rules so that they are activated only in some kind of context, this amount to add in the preconditions of the rules some clauses constraining the triggering to a certain context. This is burdensome because the designers must anticipate all the possible contexts to define the preconditions of the rules. Moreover each rules describes a part or a whole context without any reference to the dynamics of the situation which is fundamental to understand the situation.

The decision tree approach [23] tries to represent the decision step by step. This is obtained by the presence of two types of nodes: the event nodes and the decision nodes. At an event node, paths are separated according to all the possible realization of an event on which the decision maker has no influence. On a decision node, the person makes a choice. This approach might be a way to structure rule bases. For each new element analyzed in the preconditions, a new event node is created. For each new value of an existing contextual element, a new branch is created, and so on. Rule after rule, a tree is constructed. The leaves give the rule conclusions. The main problem with this structure is the combinatorial explosion. The number of leaves is an exponential function of the depth of the tree. The addition of a contextual element may easily double the size of the tree.

Bayesian networks are composed of nodes, representing random variables, and links representing the causal relations between the different random variables [9, 20]. Each node is associated to a table giving the distribution of probabilities of the corresponding random variable according to the values of the random variables of which it depends on. The influence diagrams introduce decision nodes into Bayesian networks [15, 17]. These networks are possible solutions to limit the combinatorial and information explosions in decision trees with probabilities. However, both approaches, Bayesian networks and influence diagrams, necessitate, to be handled, some information about the probabilistic dependence between the different random variables and are anyway limited to a reasonable number of variables.

Another interesting approach is the Case-Based Reasoning (CBR), which is a kind of analogy reasoning [10]. To solve a current issue, one selects the most similar problem in a problem base and one adapts the solution to the problem at hand. Note that instead of adapting prior solutions, Leake [12] proposes as an interesting alternative to store and reuse trace of how those solutions were derived. The main advantage of this reasoning is its great power of generalization and its

maintenance. However, it fails to provide explanations on the obtained solution.

### 3.3. Representing contextual information

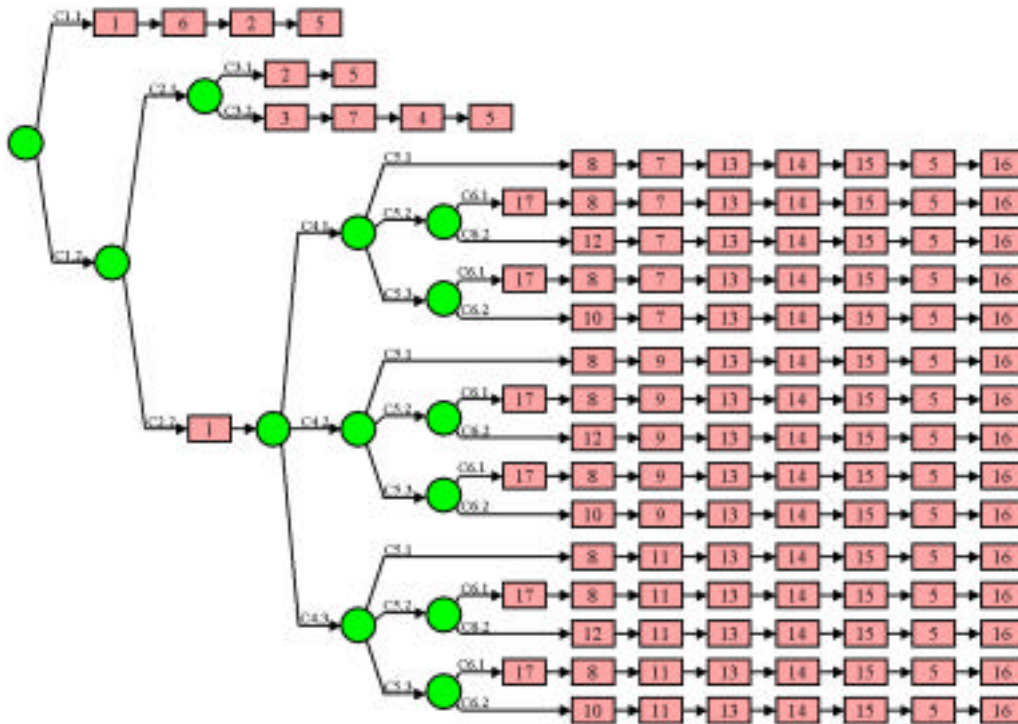
None of the above representations give us satisfaction and sticks to the operators' reasoning which is more or less a mixture of rules, decision trees and case-based reasoning. It has been recognized for a while that rules are not sufficient to capture the nature of cognition (*e.g.* [1]). Here we found again that the operators use some deductive rules simultaneously with fixed encoded schemes. This is the reason why we started to thoroughly study the operators' practice. This study proved that the operators are very sensitive to the context of their action. They try to diagnose, as soon as possible, the real state of the system by recognizing and anticipating sequences of events. As such it is reminiscent to scenario thinking. For this reason we started from the notion of decision tree.

We arrive now to a main point of our representation. Each event at an event node carries on a part of the uncertainty of the situation. For example, C1 in Figure 2, means that either an immediate repair would be possible or not (see Annex 1). There are two ways to manage this uncertainty either assess some probabilities for each events, which is the usual view in decision theory or consider that, anyway, the two events are possible depending on the circumstances. *Our representation must provide an answer for any possible circumstances whatever the probability is.* The problem of an operator is to get an adapted answer as soon as he knows what the situation is. Thus, the main problem is to diagnose the exact situations according to the contextual information reaching the operators. For this reason we considered that each branch, actually describes a contextual knowledge, which becomes more and more accurate as long as the branch is followed. It is not a question of nature bet, all the situations that may happen and have to be represented whatever their probabilities are. Thus, we are no longer interested by the probability of a branch but by the possibility to determine, as soon as possible, on which path the operator is, to determine what is the next action to undertake. In other words, we come back to the original idea of Savage [24], namely that each nature state (a sequence of events is a nature state) describes a state of the world, or using our words, a context for action. We must also stress that the operator has no choice of the path and therefore no optimal choice, because the path is dictated by the context of the incident.

At this step there is no probabilities on the events, the main purpose is to describe, with the maximum of parsimony, all the possible contexts in which the

decision has to be made. For example, the branch with {C12, C21, C32} describes a context in which there is no immediate repair possible, but enough motor coach power is still available and a steep incline is ahead. For this reason we will talk hereafter of context nodes instead of event nodes and of context instead of nature's state. The action postponement at the leaves observed in Figures 2 amounts to relating each decision to a state of nature, here the context described in the branch. Thus, our representation tends to stick to operators' behaviour. In some cases, especially at the beginning of the tree,

decision making under uncertainty would probably, if possible, be interesting, but by trying to gather as much as possible relevant information before action, the operators endeavour to make decision under certainty. This means that when undertaking an action in the tree they consider that, due to the contextual information they got, the state of nature between the root and the action undertaken is the true state of nature.



**Figure 2.** Decision tree representing the official procedure for “lack of train power” incident

Thus, we adopted a tree representation, made of two types of elements: the **actions**, which are directives to do an action, and the **contextual nodes**, which select a branch depending on the knowledge about the current context. Figure 2 shows the decision tree representation of the official procedure for “train lack of power” solving (the meaning of the boxes is not important, see Annex 1 and [17]), it suffices to say that the rectangular boxes are actions and the circles, contextual nodes.

In summary, our representation is inspired by decision trees, but mainly differs on two points. Firstly, our trees have no chance nodes but contextual nodes. Secondly, there are no probabilities. This representation shows

several important characteristics that have some consequences on the size and structure of our tree:

1. As said in section 2.2, operators use many contextual elements to perform their choice. This lead to a large number of practical strategies, even for the same incident. This multiplies the number of branches and the tree grows rapidly.
2. The operators prefer to gather a maximum of information before making their decision. This behavior postpones most of the actions to the end of the branches of the tree [2]. This observation is very close to the observation of Watson and Perrera [29] who consider a hierarchical case representation

that holds more general contextual features at its top and specific building elements at its leaves.

3. The operators privilege actions allowing to reach common intermediary situations. Thus, they can reuse common strategies to clear the incident. Graphically, the tip sequence of actions is often repeated from one branch to another.
4. Several action sequences are executed in the same order in different situations (paths).
5. Some actions could be done in different order, but must precede a given one. For example, before attaching two trains, both have to be emptied, but the order in which they are emptied does not matter (partial order on the actions).

The large expansion of the tree structure does not easily permit to represent highly-contextual decision making in complex applications. In the next section, we explain the modification we have done, based on the characteristics discussed above, to obtain a manageable structure for representing operational knowledge on incident solving on subway lines.

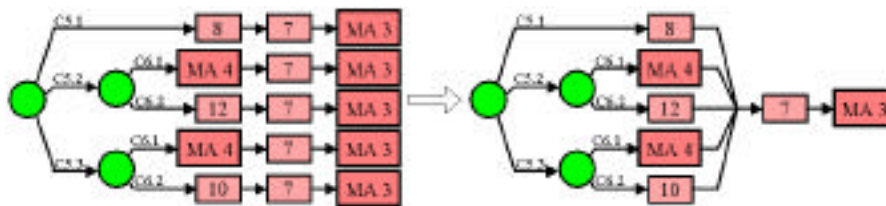
### 3.4. From contextual trees to contextual graphs

First, we can reduce the number of objects in the structure by replacing repeated sub-sequences of actions (characteristic 3 and 4 above) by a single object called **macro-action**. The choice made for defining the different macro-actions is based on common sub-procedures known by the operators, such as “linking trains,” “return to end-station without travelers”... The principle of the replacing is explained on Figure 3.



**Figure 3.** From a sequence of actions to a macro-action

This replacement simplifies the lecture of the tree, but does not reduce the structural expansion of the tree. Secondly, relying on the previous observation #3, we can merge the branches of the tree as soon as the sequence leading to the end of the incident are similar, as shown in Figure 4.



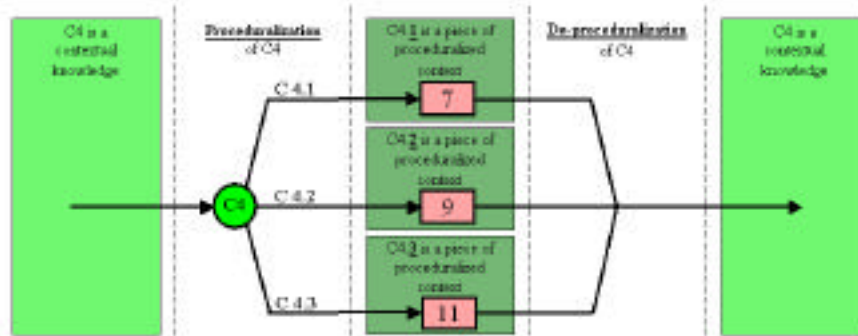
**Figure 4.** From tree to graph

Cognitively speaking, this amount to use a scarcity principle that leads the operators to try to reuse well-known procedures as soon as possible. This operation has several main consequences on the structure of the representation and on the meaning of the model.

1. We no longer face a tree but a graph. This graph is oriented without circuits, with exactly one root and one goal because operators have only one goal (clear the incident and go back to normal exploitation) and the branches express only different strategies, depending on the context, to achieve this goal. The graph structure moreover allows extending of the representation.
2. The size of the structure is now under control and the consideration of a new contextual element will add

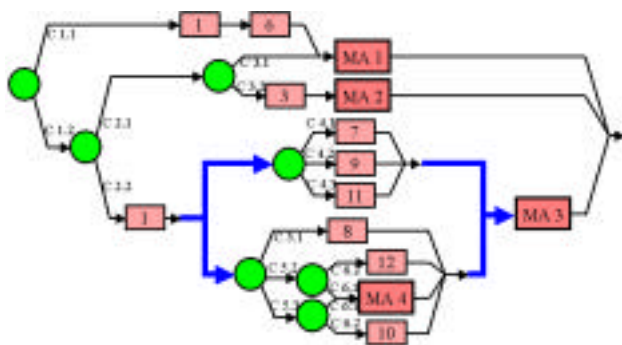
some elements in the graph, but not increase drastically its size.

3. The change of the structure introduces a dynamics comparable to the dynamics of the change between proceduralized and contextual knowledge. Indeed, when two branches are merged, it means that the undertaken actions led to a common situation from different contexts. The contextual elements attached to the different branches are proceduralized at the diverging node. They stay in this state for the different action sequences, because they intervene in the branch decisions. Finally, they are de-proceduralized when the branches are merged. By this way we explicitly express the life duration of the contextual elements (Figure 5).



**Figure 5.** Proceduralization and de-proceduralization

Even in a part of a subgraph it may happen that the actions are only partially ordered (see the example above of two trains on the same line which have to be cleared of the travelers whatever the order of the operations).



**Figure 6.** Contextual graph representing the official procedure for “lack of train power” incident represented in Figure 2.

We need to represent this issue; this is why we introduced the **temporal branching** symbols to represent action sequences that can be done in different order (characteristics # 5). This symbol is made of two parts: a divergent branching and a convergent branching; the parallel branches wear the temporally independent decision blocs. Finally we obtain the following structure (Figure 6) that we call **contextual graph**.

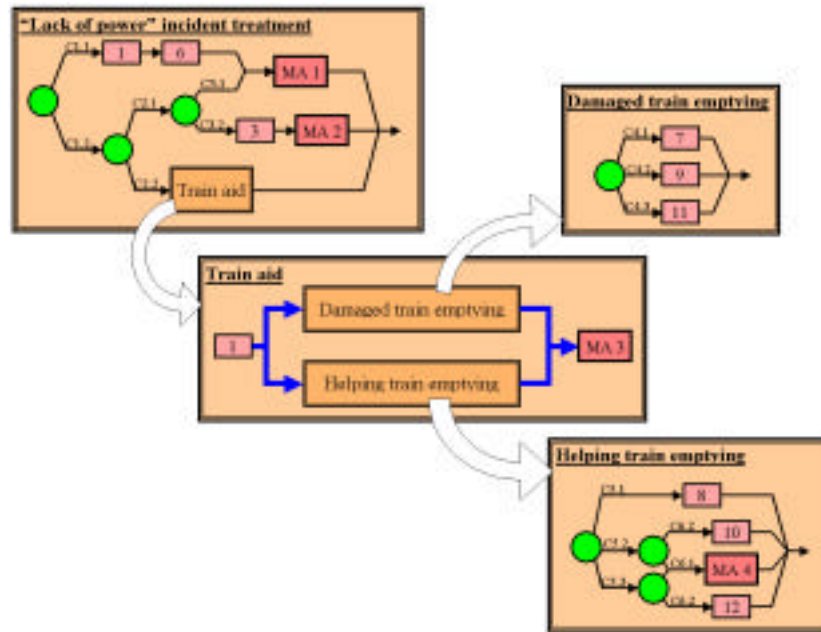
This structure is called “contextual graph” to recall that it makes it explicit the context and its dynamics for decision-making. This representation is more compact than trees and seems to be well accepted by the operators, it simply represent the succession of actions to carry out in order to solve an incident; the different possible paths express the possible strategies according to the context.

The idea to use the dynamics of scenarios for modeling practices is somehow reminiscent to various scenario-based analyses in particular in strategic planning [13, 28], design [5], software engineering, especially in

requirements engineering (e.g. [8, 21, 27]). In all these contexts, as in our case, what is important is not the probability but the exhaustivity: all the possible paths must have been envisioned.

In the contextual graphs that we have built for several incident types, we note that some parts of the different graphs are identical. Analyzing these sub-graphs we exhibit that they complete a common sub-goal (for example, when a train lacks of power and can not restart alone, or when a train has no more brake, both trains need to be helped by another train). Such a representation by contextual graphs/sub-graphs (Figure 7) is very similar to the generic tasks proposed by Chandrasekaran [6]. The difference is that this tasks are not generic in the sense that they can be combined to lead to more integrated tasks they are rather elementary or atomic tasks that can only add or followed each other in any order.

Each isolated sub-graphs, associated with a name (operators know the corresponding procedures and agree on a name) are also more or less reminiscent to a script [25, 26] but they moreover include the dynamics of proceduralization. These structures can be reused and adapted for another actions. For example in Figure 7 the sub-procedure “*Helping train clearing*” is derived from the sub-procedure “*Damaged train clearing*” and adapted by the introduction of the fact that an available train may run to the next station, if this station is free, to evacuate its travelers in better conditions. In our system the sub-graphs are the elementary chunks of reasoning stored and reminded to the operators in case of incident. The adaptation to the context is made by the choice of the path. Some flexibility is left to the operators by the adaptation of the elementary actions contained in the graph, because the actions are generally defined by their result rather than by a detailed modus operandi.



**Figure 7.** Set of contextual graphs used while "lack of train power" incident resolution

## 4. Conclusion

Our representation by contextual graphs is inspired from practice. We observed that contextual information matters more than probabilities to make decision in an operational setting. Thus we adapted the notion of decision tree to take into account contextual representation and its dynamics. We simplified the tree representation, thanks to two notions, namely macro-action and temporal branching. Going one step beyond the computer representation of reasoning on the basis of context and contextual graph, we have pointed out that some sub-graphs represent usual procedures. These sub-graphs, beyond the fact that they give a simple computer representation of reasoning, have a deep meaning for operators and are significant, even drawn out of the context of an incident. We thus are able to propose a set of interrelated contextual graphs that incorporate the notion of context in any problem solving (e.g. an incident solving) and represent the dynamics of the proceduralization de-proceduralization process.

### Acknowledgments:

RATP and COFECUB in France and CAPES in Brazil provide grants. We also thank J.-M. Sieur at RATP, C. Zanarelli, and all the other members working on the SART project.

## 5. References

- [1] Anderson J.R., *Rules of the mind*, Lawrence Erlbaum, Hillsdale N.J., 1993.
- [2] Brézillon, P. and Pomerol, J.-C., Using contextual information in decision making. *Context Sensitive Decision Support Systems*, Berkeley, D., Widmeyer G., Brézillon P. and Rajkovic V. (Eds.), Chapman et al., London., 1998, 158-173.
- [3] Brézillon P., Gentile C., Saker I., and Secron M., SART : A system for supporting operators with contextual knowledge. *First International and Interdisciplinary Conference On Modelling and Using Context (CONTEXT'97)*, Federal University of Rio de Janeiro, 1997, 209-222 (also available at <http://www-poleia.lip6.fr/~brezil/Pages2/CONTEXT-97/index.html>).
- [4] Brown J.S. and Duguid P., Organizational learning and communities of practice : towards a unified view of working, learning and organization, *Organization Science* 2 (1), 1991, 40-57.
- [5] Carroll J. M., *Scenario-based design*, Wiley, New-York, 1995.
- [6] Chandrasekaran B., Johnson T.R. and Smith J.W., Task-structure analysis for knowledge modeling. *Communications of the ACM*, 35 (9), 1992, 124-137.
- [7] Clancey W.J., Tutoring rules for guiding a case method dialogue. *International Journal of Man-Machine Studies*, 11(1), 1983, 25-49.
- [8] Jarke M., Rolland C., Sutcliffe A., Dömges R. Eds., *The nature of requirements Engineering*, Shaker Verlag, Aachen, 1999.
- [9] Jensen F., *An introduction to Bayesian networks*. UCL Press, London, 1996.
- [10] Kolodner J., *Case-Based Reasoning*, Morgan Kaufman, San Francisco, 1993.



- [11] **V. Laville and C. Zanarelli** La communication comme indicateur structurant de l'activité: illustration dans une situation de régulation de métro. 2000, submitted.
- [12] **Leake D.B.**, Case-based reasoning: Experiences, lessons, and future directions. Chapter I: *CBR in context: The present and future*. Menlo Park: AAAI Press/MIT Press, 1996.
- [13] **Leemhuis J.P.**, Using scenarios to develop strategies, *Long Range Planning* 18, 1985, 30-37.
- [14] **Leplat J.**, Erreur humaine, fiabilité humaine dans le travail. In: A. Colin (Ed.), *Collection Universitaire*, 1985, 100-120.
- [15] **Neapolitan R.**, *Probabilistic reasoning in expert systems*. John Wiley & Sons, New York, 1990.
- [16] **Nonaka I.**, A dynamic theory of organizational knowledge creation, *Organization Science* 5, 1994, 14-37.
- [17] **Oliver R. and Smith J.**, *Influence Diagrams, Belief Nets and Decision analysis*. Wiley Series in Probability and Mathematical Statistics, John Wiley & Sons, New York, 1990.
- [18] **Pasquier L.**, Raisonnements basés sur le contexte: Contextes procéduralisés, graphes contextuels et schèmes d'action. Research Report LIP6 N.2000-010, 2000.
- [19] **Pasquier L., Brézillon P. and Pomerol J.-Ch.**, Context and decision graphs in incident management on a subway line. Modeling and Using Context (CONTEXT-99). In: *Lecture Notes in Artificial Intelligence*, N° 1688, Springer Verlag, 1999, 499-502.
- [20] **Pearl J.**, *Probabilistic reasoning in intelligent systems*. Morgan Kaufmann publishers, San Mateo, California, 1988.
- [21] **Pomerol J.-Ch.**, Scenario development and practical decision making under uncertainty, Application to requirements engineering, *Requirements engineering* 3, 1998, 174-181.
- [22] **Pomerol J.-Ch. and Brézillon P.**, Dynamics between contextual knowledge and proceduralized context. Modeling and Using Context (CONTEXT-99). In: *Lecture Notes in Artificial Intelligence*, N° 1688, Springer Verlag, 1999, 284-295.
- [23] **Raiffa H.**, *Decision Analysis*, Mac Graw Hill, 1968.
- [24] **Savage L.J.**, *The foundations of statistics*, John Wiley and Sons, 1954.
- [25] **Schank R.C.**, *Dynamic memory, a theory of learning in computers and people*. Cambridge University Press, 1982.
- [26] **Schank R.C. and Alberson R.P.**, *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*, L. Erlbaum, Hillsdale, New-York N.J, 1977.
- [27] **Sutcliffe A.**, Scenario-based requirement analysis, *Requirements engineering* 3, 1998, 48-65.
- [28] **Wack P.**, Scenarios: Uncharted Waters ahead, *Harvard Business Review* 63 (5), 1985, 72-89.
- [29] **Watson I. and Perrera S.**, A hierarchical case representation using context guided retrieval. *Knowledge Based Systems Journal* (forthcoming), 2000.
- [30] **Wenger E.**, *Communities of practice: learning, meaning and identity*, Cambridge University press, Cambridge, UK, 1998.

## Annex 1

Some Actions (A) and Contexts (C) in case of incident on a metro line.

Actions	
1	Residual traffic regulation
2	Damaged train continues with travelers
3	Damaged train continues with travelers until a steep incline
4	Damaged train restarts without travelers
5	Stable damaged train at end station
6	Repair damage
7	Exit of the travelers out of the damaged train
8	Exit of the travelers out of next train
9	Exit of the travelers out of damaged train via available cars
10	Exit of the travelers out of next train via available cars
11	Exit of the travelers out of damaged train via track
12	Exit of the travelers out of next train via track
13	Next train joins damaged train
14	Link both trains
15	Convoy return to end station
16	Disassemble convoy
17	Next train goes to next station

Contexts	
C11	Immediate repair possible
C12	Immediate repair impossible
C21	Enough motor coaches available
C22	Not enough motor coaches available
C31	No steep incline between damaged train and end station
C32	Presence of steep incline until end station

C41	Damaged train at station
C42	Damaged train under tunnel
C43	Damaged train partially at station

C51	Next train at station
C52	Next train under tunnel
C53	Next train partially at station

C61	Presence of a station between damaged train and next train
C62	No station between both trains

Macro-actions		Actions lists
MA 1	Damaged train continues service	Actions 2 and 5
MA 2	Damaged train stops service	Actions 7, 4 and 5
MA 3	Make a convoy with damaged train and next train	Actions 13, 14, 15, 5 and 16
MA 4	Empty next train at a station	Actions 17 and 8